Inland Norway
University of
Applied Sciences

Inland School of Business and Social Sciences

Anders Nordby

System Thinking in Serious Games

PhD Thesis in Innovation in Services in the Public and Private Sectors
2023

Anders Nordby

System Thinking in Serious Games

PhD thesis

2023

Inland School of Business and Social Sciences
The Game School

Inland Norway
University of
Applied Sciences

Printed by: Flisa Trykkeri A/S

Place of publication: Eleverum

PhD Thesis in Innovation in Services in the Public and Private Sectors no. 29

ISBN printed version: 978-82-8380-461-4

ISBN digital version: 978-82-8380-462-1

ISSN printed version: 2535-6151

ISSN digital version: 2535-6143

Abstract

Many game designers and academics have pointed out that games are systems (Gee, 2007; Salen & Zimmerman, 2004). So, what would be more natural than using System Science or System Thinking to understand and develop them? That is, in short, what this PhD thesis is about, and is stated in the main research question: "*How can System Thinking be used to understand, design, develop, and document the process of Serious Games development?*" The PhD thesis discusses Serious Game development specifically for learning and training. However, that does not mean the results are not usable for entertainment games, it is just not the focus of this thesis.

System Thinking aims to identify, explore, and understand complexity in systems, and uses System Analysis and System Dynamics to do so. This gives System Thinking a place in Serious Game design throughout the design, implementation, and runtime phase of the game. The research also discusses how psychology can be integrated in this design, and what pedagogics should be used to implement the System Thinking into the game design. The research is conducted in three courses given to students at the Game School in HINN, Norway. The courses are well-established and have been running since 2006. The courses are revised every year, and new areas of System Thinking are included. The PhD thesis includes eight articles about these projects. Four of these are articles published in journals and are to be considered as part of the PhD thesis. The other four are conference articles and are to be considered as background information. The PhD thesis not only discusses what has been done so far, but also suggests areas of potential that can be developed in the years to come.

Sammendrag

Mange spilldesignere og akademikere har pekt på at spill er systemer (Gee,2007; Salen and Zimmerman, 2004), så hva kan være mer naturlig enn å bruke Systemtenkning for å forstå og utvikle dem? Dette er kort hva denne PhD vil forske på. Forskningsspørsmålet peker da også på dette: »Hvordan kan Systemtenkning brukes til å forstå, designe, utvikle og dokumentere prosessen rundt utvikling av seriøse spill»? PhD'n ser spesielt på seriøse spill som benyttes til læring og trening, men det betyr ikke at resultatene er brukbare til utvikling av underholdningsspill, det er bare ikke fokus.

Systemtenkning forsøker i identifisere, eksperimentere med og å forstå kompleksiteten i systemer gjennom å bruke Systemanalyse og Systemdynamikk. Dette gir systemtenkning en plass i utviklingen av seriøse spill gjennom hele designfasen, implementeringsfasen og spillfasen. PhD ser også på hvordan psykologi kan integreres i designet, og hvilken pedagogikk som bør benyttes for å implementere systemtenkningen i spilldesignet. Forkningen bygger på 3 prosjekter i kurs gitt til spillstudenter ved Spillskolen i Norge. Prosjektene har et langt tidsperspektiv, det første prosjektet går helt tilbake til 2006. Hvert år blir prosjektene evaluert og revidert og ofte inkluderes nye områder hvor vi bruker systemtenkning til spilldesign. PhD'n inkluderer 8 artikler skrevet om disse prosjektene, hvor 4 er journal artikler som anses som del an PhD'n, mens de fire andre er konferanseartikler som ses som bakgrunnsinformasjon. PhD'n ser ikke bare på og diskuterer hva som er gjort så langt i flere av studentenes kurs, men foreslår også potensialer som kan implementeres i de kommende år.

Preface

This journey has been long. The idea of enrolling in a PhD program was formed in 2006/07, and it led to writing an application within pedagogics and didactics at the University of Aarhus in 2008. The PhD thesis was based on a programming course and research project for game students that I started in 2006/7. The project is included in this thesis as Project 1 (Nordby & Karlsen, 2014; Nordby, 2015), and explores how to teach programming to arts students by teaching them game development in a problem-based class environment. In the period up to summer 2011, I completed all the PhD courses and finished the project. Unfortunately, my work situation changed, and I had to suspend the PhD thesis due to a lack of resources and other work commitments.

The new work situation also included teaching new courses. What I found most interesting was teaching System Thinking (Nordby et al., 2016, 2023). The course was started together with Professor Harald Sverdrup in the fall of 2012 in the Game School. It was introduced as a basic System Thinking course at first, but gradually became aimed more towards game design and development. In 2014, System Thinking was introduced into a course in gamification design and development (Nordby et al., 2023). In this course, the students used everything they had learned in the System Thinking course to design and develop gamifications. In the following years, I started writing articles on these courses and the projects undertaken.

In 2018/19, I was offered the opportunity to do a PhD thesis in System Thinking, with Harald Sverdrup as my supervisor. It was initially intended to take place at the University of Iceland, but things turned out differently, as Harald had received an offer to work with game studies at INN and started here in 2019. It was decided to put together the PhD thesis from six of my previously published articles, and to write two new ones. I was accepted as a student at INNs PhD education in innovation (INSEPP – Innovation in Services in the Public and Private Sector) in 2020.

I want to thank my supervisor Harald Ulrik Sverdrup and my institute leader Marit Berg Strandvik for the opportunity to do this PhD, and my family, Hilde and Vilde, for patience and support.

Table of contents

Dissertation articles

1. Teaching "hardcore science" to arts and design students: Reflecting on the development of a basic programming course (Nordby & Karlsen, 2014). Nordic Journal of Art and Research (peer-reviewed).
2. The art of gamification: Teaching sustainability and System Thinking by pervasive game development (Nordby et al., 2016). The Electronic Journal of e-learning (peer-reviewed).
3. System Thinking in Game Design (Nordby et al., 2023). Not yet published.
4. System Thinking in Gamification (Nordby et al., 2023). Will be published in 2023 in SN Computer Science. Springer Nature.

Appendix articles

1. Teaching "hardcore science" to arts and design students of the gaming generation: Reflecting on how to use game development to teach programming, System Thinking, mathematics, and physics (Nordby, 2015). Published in EDULEARN Barcelona proceedings.
2. The power of game development in learning: Why is game development good learning machines (Nordby, 2016). Published in Edulearn16 Barcelona proceedings.
3. "The Best School Day Ever" – teaching sustainability and System Thinking by pervasive game playing (Nordby et al., 2015). Published in EDULEARN Barcelona proceedings.
4. Teachers as learning game designers: Can elementary school teachers with no gaming development background really gamify their own teaching? (Nordby & Nordseth, 2016). Published in EDULEARN Barcelona proceedings.

Please note that text from these articles is sometimes used in the chapters about the project without further references.

Mission statement

Many game designers and academics –for example, James Paul Gee (2007) and Cathy Salen (Salen and Zimmerman, 2004) – have pointed out that games are systems. Applying System Thinking to games will help to identify these systems, explore the components on which they are built, and identify how they are connected to other systems, and the overall system purpose. This process can also be turned around; after brainstorming the game idea, System Thinking can be used to understand, design, develop, and document the game development. This PhD thesis will discuss just that, with a focus on Serious Games. The PhD thesis will mainly look at games for learning, which is a special category of Serious Games. The use of Serious Games for learning also incorporates learning methodologies (pedagogics and didactics) and psychology, which will also be discussed in the PhD thesis.

Research question

This leads us to the research question and goals for the PhD thesis:
How can System Thinking be used to understand, design, develop, and document the process of Serious Game development?

Goals and Scope

The main and secondary goals for the PhD thesis are:
To design and develop a framework that adopts methods, approaches, and tools provided by System Thinking, such as System Analysis and System Dynamics, to understand, design, develop, and document Serious Gaming development.

To discuss and develop methods for how learning methodologies and psychology can be integrated with Serious Game design and System Thinking to develop Serious Games.

The main goal concerns how System Thinking can be used to improve the Serious Game design and development process and to save development time by providing models that can be used to simulate the design and user experience. The secondary goal concerns how to use motivation psychology and learning methodologies to improve and strengthen the process without ruining the creative environment.

Scope

The PhD thesis will mainly concern the Serious Game design phase, with less emphasis on the simulation and implementation phase. Although this PhD thesis will mainly consider game design for Serious Games, this does not mean that the design principles described in the PhD thesis will not work for entertainment games – they certainly will, but they are not the focus of this thesis.

Also note that in this PhD thesis, System Thinking and System Science are interchangeable terms that largely mean the same thing. Our book *System Thinking, System Analysis and System Dynamics* (Sverdrup et al., 2012*)* defines System Science as the science of System Thinking, System Analysis and System Dynamics without going deeper into the definition of System Thinking.

Introduction to the subject

Serious Games

What are Serious Games?

The terms Serious Games and gamification are frequently used in game-based learning (GBL) and digital game-based learning (DGBL). While GBL and DGBL mostly involve the use of games in learning situations, gamification is about using game elements to make learning more fun. In his book *Serious Games*, Abt (1970) discusses the use of games beyond entertainment, and their potential for education, training, and problem-solving. More recent definitions define Serious Games as games that are designed for a primary purpose other than entertainment (Djaouti et al., 2015). The term "serious" often refers to industries such as defense, education, scientific exploration, healthcare, emergency management, city planning, engineering, and politics (Serious Games, 2019). Most of these fields involve simulations more than games, or more correctly, virtual interactive real-time simulations – a term that is defined below. The term "gamification" has appeared more recently and has a broader perspective than that of Serious Games. Gamification involves using game design elements in non-game contexts (Deterding, Dixon et al., 2011). The goal is to make boring learning activities more fun and engaging by introducing game mechanics that have been known to work with entertainment games into Serious Games. Serious Games and gamification can often be used together to create an even more convincing learning environment. *In this PhD thesis, the term Serious Games will include all of the above – that is, Serious Games, game-based learning, and gamification.*

The history of games and Serious Games

Games and gaming go back thousands of years, to the ancient human past. In his book *History of Social Games*, Radoff (2010) finds that games such as dice and backgammon were played in Persia, and Senet was played in Egypt, in 3100 BCE. Another example is Go (known as Weiqi in China or Baduk in Korea), which is a complex strategy game used to teach strategic thinking and planning, and dates from around the 6th century BCE (Wang et al., 2004). There are further examples on olde games, such as Ludus, a strategic board game (Rowland,1990), or Taijitu, a board game about balancing opposing forces, such as Yin-Yang (Wu, 1982). According to

Roger Smith's lecture on game history, games have always had a serious purpose, including fortune telling, religion, divination for weather, politics, accounting for crops, trade in animals, battle planning and gambling (Smith, 2009). With this in mind, it could be reasonable to assume that the history of Serious Games goes back as far as entertainment games.

The history of video games or digital games, however, has a much shorter time span, and only goes back to the 1950s/1960s. The first people to make video games were computer scientists, who created and played games such as Tic-Tac-Toe or Moon Landing on their mainframe computers. Gaming was not available to the public until the 1970s/1980s, with the introduction of handheld consoles and home computers. Along with them came joystick controllers, and Pong was one of the first globally popular games. Arcade games also gained popularity during this period, with larger-screened consoles placed in shopping malls. Mainframe computing was still popular in the 1970s, but was replaced at the start of the '80s by smaller computers such as Atari 2600, Commodore 64, Z3, Amiga, etc., which allowed people to play games at home on their TV screens. Then came a period of 8-bit consoles, followed by a period of 16-bit consoles. In the 1990s, 3D games, handheld consoles, and PC gaming started to emerge. In the late 1990s and the early 2000s, 32-bit and eventually 64-bit consoles began to appear. This was also the period in which the development of the smartphone became sophisticated enough for it to serve as a mobile gaming platform. Then came the first and second generations of game consoles from Nintendo, Sony, and Microsoft. At the time of writing, we are on the third generation of these consoles, and games can now be rented and streamed from cloud servers.

Traditional video game design and production was initially undertaken by engineers, programmers, and self-taught enthusiasts with little or no formal training in game design (Donovan, 2010; Kent, 2001). This changed toward the end of the 1990s, however, because the video game industry grew internationally and the cost of the productions rose, and there was a need for professional game development (Egenfeldt-Nielsen et al., 2008, pp. 15-21; Egenfeldt-Nielsen et al., 2013). Video game productions changed from small, one-man projects to large-scale projects with hundreds of employees. Developing video games became a major industry with specialized roles, such as programming, animation, and sound and game design, and the need for qualified personnel with formal education became evident. Video game design literature began to emerge. One of the first books, published in 1984, was *The Art of Computer Game Design* by Chris Crawford (Salen & Zimmerman, 2006). The literature focused on the ludic aspects of video games. Books such as *The Art of Game Design: A Book of Lenses* (Shell, 2008)*, Rules of Play* (Salen & Zimmerman, 2004), *Fundamentals of Game Design (Adams,* *2014),* and *Game Design: Theory & Practice* (Rouse, 2005) drew attention to interactivity, rules, mechanics, gameplay, and the development cycle. Game design continues to build heavily on traditional game development, developed gradually in parallel with computer development, often by enthusiasts.

The main foundation for the latest revolution in entertainment games, and the subsequent rise of Serious Games and gamification, is the developments that have taken place in computer technology. Simple 3D environments have quickly developed into controllable, virtual interactive real-time environments (VIRE) that trigger more and more senses. They start to feel alive and increasingly like the real world. Behind the scenes, more sophisticated programming enables more and more advanced art, technology pedagogics, and psychology to be integrated. The controllers and computer screens have also started to incorporate virtual or augmented reality glasses or controllers (VR/AR/XR), which provide even greater immersion. VIRE has also expanded into the internet in the form of online games and virtual worlds, creating a playground for the whole world, with access by players and peers around the globe. VIRE is the basis for all entertainment computer games and will also be the basis for Serious Games or gamifications. They are already seriously immersive and will quickly become even more so in line with computer developments.

The video game industry is huge. In 2018, the industry was worth around US$137.9 billion, which is larger than the music and movie industries combined (Newzoo & GamesIndustry.biz, 2018). More recent data from summer 2023 – for example, from the online source WePC (2023) – states that the video gaming industry was worth US$178.73 billion in 2021, which was an increase of 14.4% from 2020. Another source (Statista, 2023) estimates that revenue in the video games market will reach US$384.9 billion in 2023 and an annual growth rate of 7.89%, resulting in a projected market volume of US$521.6 billion by 2027. Although some of these figures are only estimates and the sources are not based on solid research, they paint a picture of a huge market that is growing fast.

Serious Games and learning

James Paul Gee (2010a) describes a game as follows: ‘All a video game is, is a set of problems that you must solve in order to win. The game itself does assessment all the time. No test is necessary when the game is completed to check whether the gamer has learned, it is proven throughout the game that he has.’ Pearce (2011) also points out that games have clear goals and contain the tools required to reach them. Challenges and obstacles and the learning needed along the way are easily accepted; games that are too easy are considered boring and not fun to play. Games, as Kapp (2012) has shown, can offer customized learning environments where the player can take chances and fail without severe consequences. Also, online games give opportunities for collaboration, peer learning, and competition with peers from all over the world.

Gee (2007) further discusses how good games are related to learning. He points out that games are systems from which the player can get a simplified top-down perspective, and then dive into the details later. Well-made games are pleasantly frustrating with the right learning intervals, and it is easy to see the training from a wider perspective and as a part of an overall goal. He points out how this can give a meaningful context for the training. Players can be creative and create their own styles – they are active agents, which is good for learning. The players can work in teams and take their own identity in a “community of practice” (Lave & Wenger, 1991), where the overall goal seems to be to play a game while the learning of school topics is “invisible”.

Games also integrate perception and action, which allows the player to immerse themselves in the game and to acquire an overall view before he manipulates at a distance. Learning through games can be “cross-disciplinary” and multimodal, and the assessment is continuous and immediate. Games contain beautiful graphics and moving characters, as well as sound and music. The characters that represent the player can often be customized to suit the player’s preferences. If the games are designed well, it is easy for the player to immerse themself in these interactive environments and go with the role play they suggest. An example of a Serious Game in education can be found in Article 1: IBM’s Robocode (Nelson, 2011; Long, 2007), where the students program intelligent tanks that have only one purpose: to create the most intelligent code, so that their tank can beat the other tanks. Another example is the Finnish game Lingoland (Rajaravivarma, 2005) that teaches liberal arts students programming by using numbers and word plays. The article also mentions a Danish game-based environment (see Andersen et al., 2003) that aims to teach arts students programming and can be brought into the classroom. Partti and Karlsen (2010) bring informal learning into the classroom and show how this is played out in a music community and implemented as part of a compulsory music education program (see also Partti & Westerlund, 2013). Some educators have, of course, also developed games that are especially suited for learning (Egenfeldt-Nielsen et al., 2013). More recent examples are SimCity Education, which is an educational version of the popular city-building simulation game, SimCity. It is used to teach urban planning, environmental science, and social studies (Steinkuehler, 2012). Minecraft Education Edition is a modified version of Minecraft, and is another example of a building game that is designed for educational settings. It can be used to teach various subjects, including history, geography, and coding (Birt & Hovorka, 2019).

Children and learning

Playing online games is also increasing rapidly among young people (Kzero, 2011). This generation of young people enter the school system with knowledge and experience gained from gameplaying.

Tapscott (2008) elaborates on what these children want from future employment in his book *The Net Generation*: 'They want to have fun. In fact, 58% of them say that having fun with a product or service is just as important as what that thing actually does. If you employ any of these people, realize that they also want to have fun at work. They want to collaborate and have relationships. They want innovation and creativity. They want speed. They want to customize everything … this group wants to do things their own way.' These wishes and needs are not often met when they enter the education system, because the courses are often designed in a more traditional way. Gee (2007) gives an example of his son, who jumps into a game without reading any manual and solves it in an inductive way by doing it. Gee tries to read the manual first, but it makes no sense to him until he has played the game for a while himself. This approach is about learning by doing (Dewey, 1916) and learning just in time (Gee, 2007). Prenski (2011) points out that these children process information more quickly, they have the ability to concentrate on many things at once, and they have sharper visual sensitivity and greater iconic understanding. They also have the awareness and ability to make connections between multiple sources of information ("they have hypertext minds"). They also have the ability to work in virtual teams and like to figure things out by trial and error. All in all, they have a play attitude and want to "just do it", and they have less tolerance for passive situations such as lectures and meetings and have an intolerance for things that do not pay off. These children meet each other on the internet in online games and in virtual worlds, and they are used to meeting people from all over the world. All the points above are important when discussing learning methodologies for these children.

Learning methodologies and pedagogics

These developments also mean that pedagogics that are used in gameplaying are already well-known to children from the gaming generation. Games often use some form of problem-based or case-based methodology, which are well-established pedagogical methodologies (Pettersen, 2005). It may reasonably be assumed that these pedagogical methodologies could also be appropriate as learning methodologies in schools. Pedagogics suggests that learning can also be achieved in a community by participating. One well-known theory concerning this is that of situated learning (Lave & Wenger, 1991). This kind of learning could also possibly take place in a virtual environment, which means that this theory is also of interest for game-based learning. The rapid development in computer technologies also provides new possibilities for virtual "learning by doing" (Dewey, 1916) to a much greater extent than before. This will create "just-in-time" learning events when the motivation is present, and the players want to solve a game problem (Gee, 2007).

Game design elements and psychology

When schools start to use Serious Games as part of learning, it will become necessary to ensure that it leads to better and correct learning. To achieve this requires the connection of gaming elements with motivation or learning psychology, to steer the learning in the right direction (Kapp, 2012).

An example of this could be the motivation theory called ARCS (Keller, 1987), which suggests that attention, relevance, confidence, and satisfaction are the key elements to create motivation. *Attention,* for example, can be gained by showing related examples the player is interested in, using conflict, surprise, or incongruity, and varying the delivery method regularly. Making the experience feel *relevant* can be achieved by describing the goals, how to reach them, and how they will help the player, and then modeling the results. The learner can be made *confident* by estimating the requirements and expectations, providing small possibilities for success along the way by creating small milestones, and providing feedback so the player feels that he is controlling the experiences. *Satisfaction* can be gained by providing positive encouragement along the way, allowing the player to apply their learning, maintaining consistent standards and measures of success for all learners, and of course by trying to tap into the learner's intrinsic motivation and not just use rewards. All triggering of the ARCS elements above is done with game elements such as feedback, reward structures, conflict, competition or cooperation, abstraction, storytelling, goals, aesthetics, flow, time expansion or compressing, etc. The ARCS feedback leads into System Thinking.

System Thinking history

The idea that the world is a system goes back a long way, to Plato (born 428 BC), who proposed the theory of forms and cosmos created out of the chaos of fire, water, air, and earth into the body of the universe, and to Aristoteles(born 384 BC), for whom form and matter could not exist without each other and without a purpose. In 1564, Galileo turned the telescope towards the night sky and described the system of the world. Descartes (born 1596) saw the world as a machine, and he was followed by Newton, who formed a system theory in his famous *Philosophie Naturalis Principia Mathematica* (1687), which started a movement to use mathematics as a language for what would become System Analysis.

It was not until 1961 that the term System Dynamics was established by Forrester at MIT. He described System Dynamics as a simplified model of the system and its feedback loops, which is used to simulate the system's response to various inputs over time. In this way, the model can be used to simulate the past, and when it works, it can be used to predict the future. In his work with industrial and urban dynamics, Forrester was a pioneer in many ways. His urban models focus mainly on the growth and decay of urban systems through economic systems such as housing, employment, and industry (Forrester, 1969). Many followed him, and the environmental movement raised transdisciplinary questions and problems that needed to be considered as a whole (Meadows, 1972; Meadows et al., 1992). System Dynamics was developed as a way to understand sustainable development. Much discussion was required in order to understand and analyze the feedback loops in the systems (Dørner, 1996; Senge, 1991; Sterman, 2000; Sverdrup et al., 2002). System Dynamics is usually described by stock flow diagrams (SFD), which show how stocks and flows behave in a system. Coyle (2000) and Maani and Cavana (2000) described how conceptual models can be taken one step further to become dynamic numerical models. However, quantitative models are complex and of limited use to communicate an understanding of the model to the user or client.

System Analysis examines systems and the interactions of elements within and between them. This is done by creating conceptual model structures with the help of *causal loop diagrams* (CLD), and ideally over a group modeling process (Randers, 1980; Vennix et al., 1992; Vennix, 1996; Andersen & Richardson, 1997; Vennix, 1999; Maani & Cavana, 2000; Sterman, 2000; Rouwette et al., 2002). System Analysis is further developed through group modeling (gathering stakeholders and discussing a common understanding by drawing a common CLD) to include flow charts to convey what flows through the systems, and various other tools that will be discussed later. System Analysis is solely a qualitative understanding of a system, and it can easily be taught to people without any technical background. System Analysis is also discussed quite a bit within System Dynamics circles and is found by many to be useful to convey an understanding of the more complex stock flow diagrams and the System Dynamics Modeling.

System Thinking history goes back almost as far as game history and received a great boost in the computer age. It should be a widely used tool, especially in the current age where "sustainable solutions" are so important. However, searching the System Thinking on the internet reveals very little. One might also think that System Thinking would play a strong role in game development, as games are built from systems. Surprisingly, this is not the case! This, however, is what this PhD thesis is all about.

Innovation

This PhD thesis will discuss how to integrate Serious Games, System Thinking, pedagogics and psychology to create learning environments. I view this PhD project as innovation, rather than invention. Innovation is about combining existing ideas and resources in a novel way (Schumpeter, 1934). As innovation mostly combines existing knowledge in new ways, it can be viewed as enlightening (Lundvall, 2010). My research combines known disciplines and topics in new ways, and aims to create scientific methods that can later be used to create a methodology for Serious Game development.

Games can be delivered in the form of software, tools, merchandise, and documents (goods), as well as via training courses and in-game services (services). I see this research project as a combination of goods and services innovation. A typical company that uses the results of the PhD project will develop Serious Games, or documentation and courses for other Serious Game development companies. As such, the research has a synthesis/integrative perspective on developing service innovation.

Grønroos (2001) defines service innovation as follows: 'as an activity or series of activities of a more or less intangible nature that normally, but not necessarily, take place in the interaction between the customer and service employees and/or physical resources or goods and/or systems of the service provider, which are provided as solutions to customer problems'. Edvardsson et al. (2005) suggest that: 'Services are like goods, but with the following additions: Intangibility, Heterogeneity, Inseparability, Perishability' (IHIP), which are all questioned, but still essential in the development aspects of the service innovation research field.

The innovation literature also discusses service systems, which concern framing service innovation as a system (Edvardsson & Tronvoll, 2013; Vargo & Lush, 2011). Vargo and Lush argue that: 'A system orientation is important to both academics and practitioners because it has different implications for understanding and applying principles of value co-creation, as is particularly essential in an increasingly interconnected, and thus increasingly dynamic, world.' Edvardsson and Tronvoll (2013) argue that understanding value creation is the key to understanding service innovation, and that value is created interactively in configurations of resources and actors. Companies do not offer service innovation, but rather develop and manage service systems to ensure that value is created for the customers. Vargo et al. (2010) call the value creation configuration "service systems", while Edvardson et al. (2011) argue that these systems are always embedded in social systems, where actors and their value creation are shaped by social interaction and the service system. I believe that service innovation can be linked to Serious Game development, in that they are both service innovation systems. It then follows that the System Thinking I use in this PhD thesis can also be used to describe service systems.

Another common feature is that service innovation systems can describe how learning or value creation is created in the systems. Edvardsson and Tronvoll (2013) describe how the conceptualization of service innovation is based on the interdependencies between resources and schemas that shape the actors and the value creation, although this research uses problem-based learning and situated learning/communities of practice to do the same.

Serious Game development builds on general and traditional game development, which is also an innovative process (and sometimes also an inventive process) resulting in a product, such as board games, or video games on computers or consoles, or in service innovation, such as online games that sell a product, as well as services connected to the game, such as accessories, game extensions, game currencies, etc. The innovation literature is of interest in this respect, as the processes in traditional game development are generally similar to the innovation processes described, for example, in Jean Hartley's chapter on public and private features of innovation (2013), or in Fuglesang et al.'s structured and unstructured approaches to innovation (2011, 2014).

However, I have not found the methodology of System Thinking, such as qualitative System Analysis in the form of CLD, flows and RBP (reference behavior pattern), or System Dynamics in the form of quantitative simulations, to be used in the innovation literature. I find that surprising, as I believe that this could enrich the service innovation field and make it more detailed, particularly in areas that consider how innovation is implemented or can be implemented in companies (as it does in the field of Serious Game design). However, to do this does not lie within the scope of this PhD thesis.

Knowledge gaps

There are several knowledge gaps in the above discussion. Below I have listed what is required to fill the gaps that I find most interesting:

- Create methods to be used to form a complete and serious methodology for implementing Serious Games for the new reality (children).
- Strengthen traditional game design or Serious Game design by introducing a scientific methodology (System Analysis) in the design phase.
- Simulate the design by using scientific and proven methodology (System Dynamics) before the implementation starts.
- Document the development process of Serious Games in a proven and scientific way.
- Create contributions to a theoretical fundament that can be used in innovation education.

This PhD thesis introduces System Thinking as a scientific methodology to fill these gaps in Serious Game design. System Analysis can be used in the game design processes to design, document, and understand how the game works, whilst System Dynamics concerns making a simplified quantitative model of the Serious Game and running simulations to see how it behaves over time. System Dynamics can also be used to test selected values and structures in various parts of the game design, for example various game balances between players and bosses, and the balance and effect of psychological motivation theories, etc. In addition, System Dynamics can be used to build simulations that can be used for learning. As this PhD thesis concerns innovation rather than invention, it might also shed some light on how to use System Thinking for general innovation, although this is not the focus of this PhD thesis.

Methodology

The main goal of my PhD thesis is to propose a new method to develop Serious Games by joining four field areas. These are Serious Game design, psychology, pedagogics and System Thinking. System Thinking will be the methodology that glues it all together. Pedagogy concerns identifying which learning methodologies to use and how to implement them. Serious Game design and psychology must work together to create play (the fun) and the motivation to learn. All the components are built on traditional methodologies for game design, psychology, pedagogics, and System Thinking. The PhD thesis does not aim to create a new methodology but is more concerned with developing new methods that can later be used to create a new methodology.

System Thinking – The main methodology

The thesis will use the terminology associated with System Thinking, System Analysis, and System Dynamics, as based on definitions from Senge (1990), Sterman (2000), Haraldsson (2004, 2005), and Sverdrup et al. (2019). The practical use of CLDs in examples throughout the thesis will use definitions from *Introduction to System Thinking and Causal Loop Diagrams* (Haraldsson, 2004). We define System Science as the science of System Thinking, System Analysis, and System Dynamics. System Science and System Thinking are mainly defined as the same thing. System Thinking is concerned with understanding causal relationships – how cause produces an effect, and how the effect may have feedback on the cause (Senge, 1990; Sterman, 2000; Sverdrup et al., 2019; Sverdrup & Svensson, 2004; Forrester, 1969). However, in recent years, there have been many definitions of System Thinking. The one that we usually refer to is by Senge (1990), who defines it as a discipline for seeing wholes, and a framework for seeing interrelationships rather than things, and for seeing patterns rather than static snapshots. In general, most definitions have a focus on the elements and connections within a system, rather than what a system actually is or does. A more recent definition that does this has been provided by Arnold and Wade (2015), and defines System Thinking as a set of synergetic analytical skills that are used to improve the capability to identify and understand systems, to predict their behaviors, and to devise modifications to them in order to produce desired effects. These skills work together as a system. System Analysis is a conceptualization upon which a qualitative understanding of the logic of the systems is built, whilst System Dynamics is concerned with building a quantitative model of the system that is used to simulate it. This requires the assignment of numerical values to all components in the system. System Thinking is traditionally defined as a three-step model, as shown in Figure 1 below.

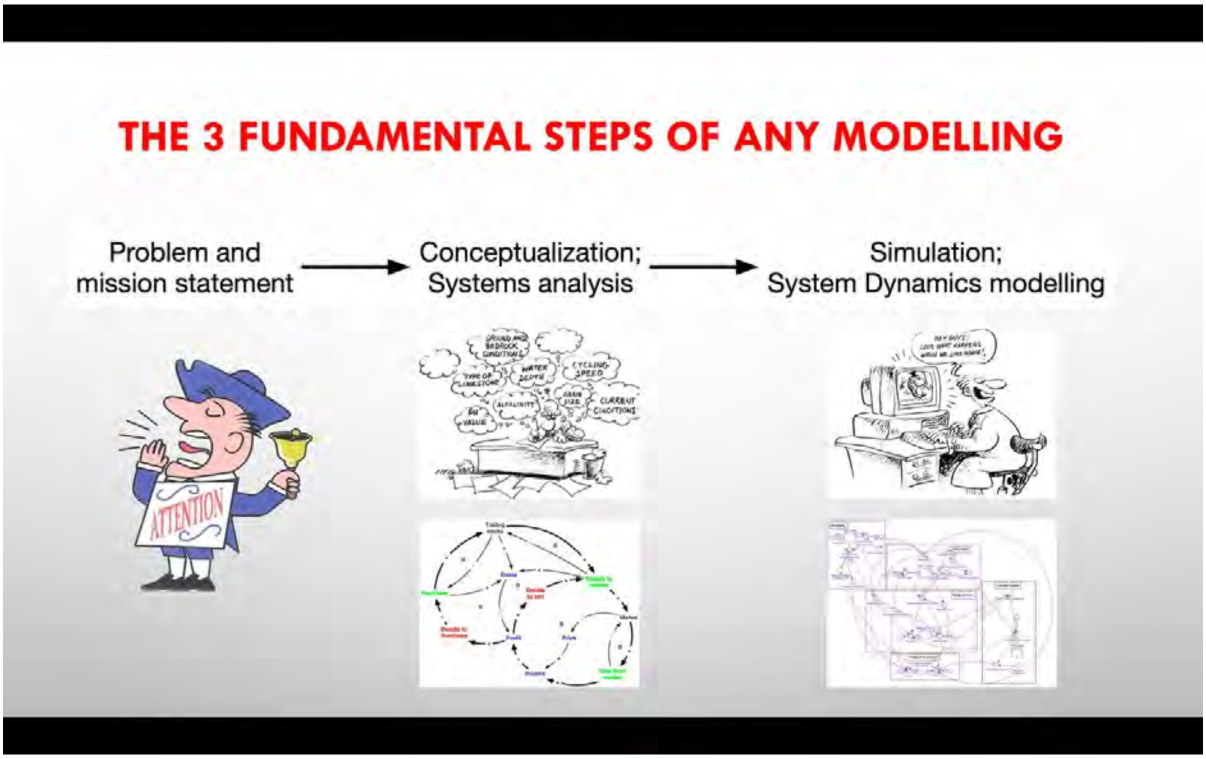


Figure 1. The traditional System Thinking model, as defined by Sverdrup et al. (2012). Mission -> conceptualization (System Analyses) -> simulation (System Dynamics).

The first step is to create a *problem and mission statement*. In this step, we brainstorm the project, identify the challenges/research questions, and define the goals. From this a mission statement is created.

In step two, the *System Analysis*, we conceptualize the problem. The boundaries are defined (what will and will not be included in our system), a clear question is defined that describes what will be studied, and a list of relevant project variables is created. The variables are often first sorted in a table, with items, actions, and controls. From this table we start by making a causal loop diagram (CLD) that connects the causes, effects, and feedback loops in each system. The CLD is the main tool in System Analysis. A CLD is a common language that conveys the logic of a system and helps us to understand the cause-and-effect relationships between the variables within a system. By observing the causes and effects, the kind of feedback that is at work within the system can be determined. After the CLD has been created, we make the connections between the systems. Flow charts (FC) that show what flows through the systems are also created. Finally, a Reference Behavior Pattern (RBP) is created, which shows how we think the system develops over time. System Analysis is a conceptual qualitative methodology that contains no information about values.

The third step is System Dynamics Modeling. The System Analysis gives a conceptual understanding of the system, but contains no numbers, and cannot tell us anything about how much of a cause is needed to give a certain effect. Nor can the System Analysis tell when the system balances. To obtain an insight into these issues, a System Dynamics Modeling of the system is performed. The modeling is quantitative and requires us to find numeric values and ranges for all parts of the system. Modeling is used to check the models in different scenarios. When the modeling reflects the design or the real world, it can be used to predict how the model will behave in different future scenarios. The software used to make models and do the simulations is ISEE Systems' Stella Architect.

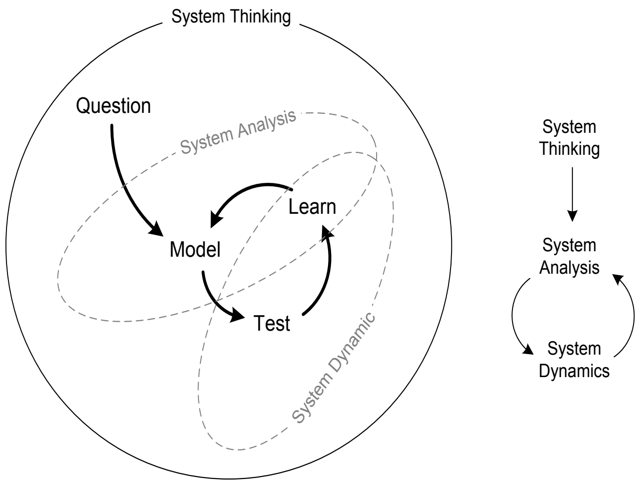


Figure 2. System Thinking, System Analysis, System Dynamics (Haraldsson, 2005; Sverdrup et al., 2019).

The framework – Addressing the gaps

System Thinking will be the scientific methodology or framework used to perform and document the design process (System Analysis) and to simulate the models (System Dynamics). The simulations will be checked against the design or the real-world processes to verify that the design meets the specifications and the goals, and fills the gaps. This is illustrated in the CLD below, starting on the left with the *Game proposals.* Two paths/arrows extend from that – one thick arrow that leads to the *Game goals* and another arrow that goes through *Conceptualization* and *Model implementation*. The first arrow represents the game goals we have specified or collected from the real-world system we want to simulate, and the other represents the design and simulation path built using System Thinking. In *Simulated outcome gap to goals* the two are compared, and the simulated output is compared to the game goals, to see whether our design works. Based on this comparison, we *Learn and revise the proposal* and, if necessary, we go back to *Conceptualization* or the *Game proposal* to revise and simulate again. This may need to be done many times before the design works as intended. Once we are happy with the result, we move to the top of the CLD and *Implement the game*. Again, we compare the implemented version against the goals, and, if necessary, we go back to the *Game proposal* and revise the game and perform new simulations. If we find any other *Issues* during the implementation, we do the same.

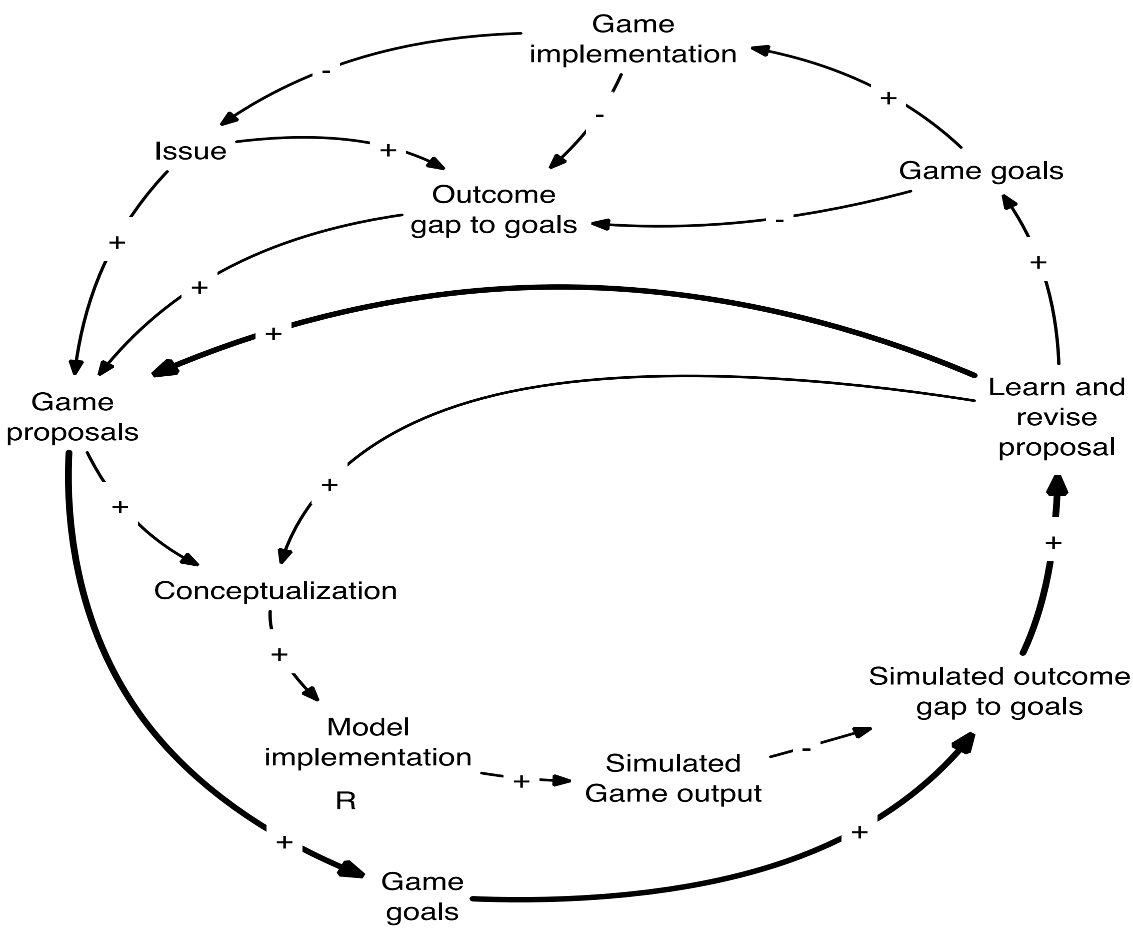


Figure 3. The point here is to illustrate how all games, or parts of the game, are simulated and tested before they are implemented. System Thinking can be used in many ways in the game design process – for example, in the design of the game logic, in the motivation psychology logic, in the learning methodology logic, and in the programming logic, etc.

Learning methodology

The learning methodology concerns the pedagogics that we implement and use in Serious Games, and are thus important for how well the players learn. Gee (2004, 2007, 2010b), Prensky (2011), and Shaffer (2007) argue that well-developed games are well-defined learning environments, which offer well-defined goals and the support to reach them. Continuous and immediate assessment is conducted between the goals and while playing the games. This is a problem-based learning (PBL) approach, which facilitates sequenced problem-solving, whereby each problem prepares the player for progressing to later parts of the game and, finally, to achieve the final game goal. PBL also facilitates team-based work, whereby problems are solved together as a team (Barrows & Tamblyn, 1980).

The book *The Fifth Discipline* by Peter Senge (1990) describes learning methodology through five disciplines that must work together to create a learning organization or team. Personal mastery, mental models, shared vision, and team learning are integrated through System Thinking to form a complete learning methodology. My aim is to adapt a similar learning methodology for this thesis. We also assume that members of a group that either develops or plays the Serious Game will learn by participating in and doing what the group does. This can be described as situated learning (Lave & Wenger, 1991; Wenger, 1998), and the group of participants is seen as a "community of practice". In the theory chapter, we will discuss the criteria for learning from this theory.

Gameplaying will most often be done in a virtual environment where the player plays with an avatar. This is "learning by doing" (Dewey, 1916). As the game progresses, it will become necessary to learn theory in order to progress and solve a task. This will happen when the new knowledge is necessary and when the player is motivated to solve the task. This is "learning just in time" and is discussed by James Paul Gee (Gee, 2007). The way in which these methodologies are used is mainly discussed in Article 1 in Project 1, but it remains the same for all three projects.

Game design and psychology

In most games, game elements such as goals, storytelling, and competition are used to create purpose, whilst feedback, replay, cooperation, and rewards maintain interest. Esthetics is important to make the virtual environments feel real and as close to reality as possible, and the right implementation of flow is used to make the gameplay fun and not boring or too hard. However, what makes entertainment games different from Serious Games is that many of the game design elements mentioned above are linked to psychology to ensure that the gaming motivates learning.

The PhD projects do that in different ways. In Project 1, the students made a game to learn programming, math, and physics. The motivation was in "making a game", as that was the goal of their education. While the students were striving to make something they loved, they were also forced to study school topics such as programming, math, physics, design, modeling, drawing, etc., along the way. All students had the same goals and rules, and cooperated and competed with other students, etc.

In Projects 2 and 3, the students created gamifications and Serious Games. The game created in Project 2 taught them System Thinking and sustainability, while Project 3 mainly taught them game design and gamification. I use the word "mainly" here because there were also many secondary topics that arose during game development, which the students then had to study. For the students, the motivation in Projects 2 and 3 was the same as in Project 1. However, in these projects, they were creating Serious Games, so the design needed to incorporate psychology to ensure that the players would learn what they were supposed to learn.

The students had to study psychology – or more specifically, motivation theory. We leaned on motivation theories from Lepper (1988), Malone (1981), and Self Determination Theory (SDT), as well as on Kapp's book on gamification (2012), which discusses both motivation theories and how to use them in relation to facts, concepts, rules, and procedures. With regard to serious and traditional game design, we leaned on books such as *A Book of Lenses* (Shell, 2008)*, Rules of Play* (Salen & Zimmerman, 2004), *Fundamentals of Game Design (*Adams, 2014*),* and *Game Design: Theory & Practice* (Rouse, 2005).

Service innovation

As INSEPP is where I will eventually submit my PhD thesis, and my PhD thesis is clearly about innovation, I will also consider some innovation theory. Although it will not be used as the main theoretical foundation, I will position my PhD thesis in the service innovation landscape and use relevant theory when possible.

How I worked with literature and theory

My interest in doing a PhD thesis started with the first project of the PhD thesis (Article 1) back in 2006-2012, where the focus was to improve the programming course for the game students and to lower the failure rate. There were two focuses: how to visualize the effect of the programming the students had to learn, and which pedagogics should be used the course. I started with the visualization focus, and read numerous books on how to visualize programming, such as *Macromedia Flash Mx 2004 Game Development* (Rhodes, 2004) and *Learning Actionscript 3.0* (Shupe et al., 2008). For the pedagogics, I began by reading *Kvalitetlæring I høgere utdanning* (Pettersen, 2005), which became the main source for the pedagogics for Project 1 in this PhD thesis. Following the first tests in 2006, we settled on problem-based learning (Barrows & Tamblyn, 1980) as the main pedagogic methodology for the project. At around the same time, I read James Paul Gee and David Williamson Shaffer's books on games and learning (Gee, 2004, 2007, 2010; Shaffer, 2007). Project 1 was clearly "learning by doing" (Dewey, 1916). Gee (2007) discussed how learning by doing created "learning just in time" for events in games and learning situations. Since then, "learning by doing" and "learning just in time" have been an important part of the learning methodologies in the projects. When article1 was written a few years later (2014), we decided to also include situated learning (Lave & Wenger, 1991) as a learning theory.

I have scanned many books and articles on the topics of Serious Games and gamification. At the start of the gamification course in 2012, I gave lectures based on Kevin Werbach's videos on gamification (Werbach, 2012). After a short while the course literature was changed to Karl Kapp's book *The Gamification of Learning and Instruction* (2012).and has since then been the most important source for gamification theory. This was also our initial source for motivation theories when we started to develop motivation CLDs by connecting game elements and motivation. Serious Game and gamification design also build on traditional game design, and several sources have been studied for this theme. Traditional game design has been taught in the Game School (INN) for 20 years, and the most important sources for this are *The Art of Computer Game Design* by Chris Crawford (Salen & Zimmerman, 2006), *The Art of Game Design: A Book of Lenses* (Shell, 2008), *Rules of Play* (Salen & Zimmerman, 2004), *Fundamentals of Game Design* (Adams, 2014) and *Game Design: Theory & Practice* (Rouse, 2005). However, Karl Kapp's aforementioned book on gamification (2012) has also been a frequently used source for me, as the game elements discussed there are presented more in a gamification perspective.

The main resource for System Thinking is our own book, *System Thinking, System Analysis and System Dynamics – The Methods Used to Find Out How the World Works and to Simulate What Could Happen with Models* (Sverdrup et al., 2022). This was initially released in 2012 but has been updated regularly since then. Other important sources for design of System Analysis and CLDs include *Introduction to System Thinking and Causal Loop Diagrams* by Hørdur V. Haraldsson (2004) and *Developing Methods for Modelling Procedures in System Analysis and System Dynamics* (Haraldsson, 2005). *The Fifth Discipline* (Senge, 1990) has also been a good source for System Thinking, as it has often been read by pedagogues.

The research design includes several research methods and could be described as practitioner inquiry as stance (Cochran-Smith & Lytle, 2009). Cochran-Smith and Lytle include several kinds of research, such as action research, teacher research, self-study, the scholarship of teaching, and using practice as a site for research. Practitioner inquiry intentionally blurs the boundaries between teaching, practice, and enquiry, and research data is collected systematically throughout the process. Of these research forms, action research has been the most used, and is defined as an interactive inquiry process that balances problem-solving actions implemented in a collaborative context with data-driven collaborative analysis or research to understand underlying causes, thereby enabling future predictions about personal and organizational change (Reason, 2001). The research design also draws on methods from System Dynamics theory, such as System Analysis, and causal loop wrapping and loop analysis, to understand System Dynamics (Senge, 1991; Sterman, 2000; Haraldsson & Sverdrup, 2004).

Theory

This chapter presents the case for a four-step approach for implementing System Thinking in Serious Gaming. This will be done by merging the previously mentioned concepts – i.e., System Thinking, Serious Gaming, gamification psychology and learning methods. The theory will be presented with a point of departure in the "learning loop" approach (Haraldsson & Sverdrup, 2004; Haraldsson, 2004, 2005; Sverdrup et al., 2012, 2019) and its evolution into implementation of Serious Gaming, showcased conceptually stepwise in CLDs.

System Thinking

System Thinking is the framework that is applied to game development. It has two steps: System Analysis and System Dynamics Modeling (Haraldsson, 2004, 2005; Sverdrup et al., 2012, 2019). The System Analysis starts after we have identified the systems and made a problem and mission statement. The System Dynamics Modeling starts after we have conducted a thorough System Analysis (Figure 1). When both have been completed, the implementation part starts. However, all steps are loops that feed back to the previous step. The whole development process can be shown in a learning loop (Figure 4).

System Analysis and the learning loop

Undertaking a complex project and solving it involves a learning process. When performing System Analysis, we usually start with the learning loop (Haraldsson, 2005). After brainstorming, the ideas are sorted into *Goals and work description*. We apply *Project knowledge* and create a *Conceptual model* of the project. The learning loop can take different forms in its basic loop, although testing, knowledge, and learning are always involved in the building of a basic conceptual understanding. Figure 4 shows the basic learning loop. Causes and effects are connected by arrows and form a loop. We can see that *Conceptual modeling* leads to *Conceptual testing*, which again leads to *Conceptual learning and insights.* An arrow where more of the cause results in more of the effect is marked with a plus (+) sign. In this loop, we only have + signs, which means we have a reinforcing loop. This is indicated with R1 (Figure 4). There is also another loop, *Conceptual testing*, which is connected to *Pool of knowledge and experience*, and pool of knowledge and experience feeds back to *Conceptual testing.* The idea here is that the more we test, the fewer questions we will have. More

Conceptual testing leads to fewer questions and less data in the *Pool of knowledge and experience*, which is indicated by a minus (–) sign. This forms a balancing loop, which is indicated with B1 (Figure 4). On the other hand, the feedback says that the more data we have in the *Pool of knowledge and experience*, the more testing we must do. This is indicated by a + sign. This loop has one negative and one positive arrow, which indicates that it is a balanced loop. The last two arrows, *Goals and work description* and *Project knowledge*, have positive arrows to *Conceptual modeling*, which means that the more descriptions and knowledge we have, the more modeling we can do.

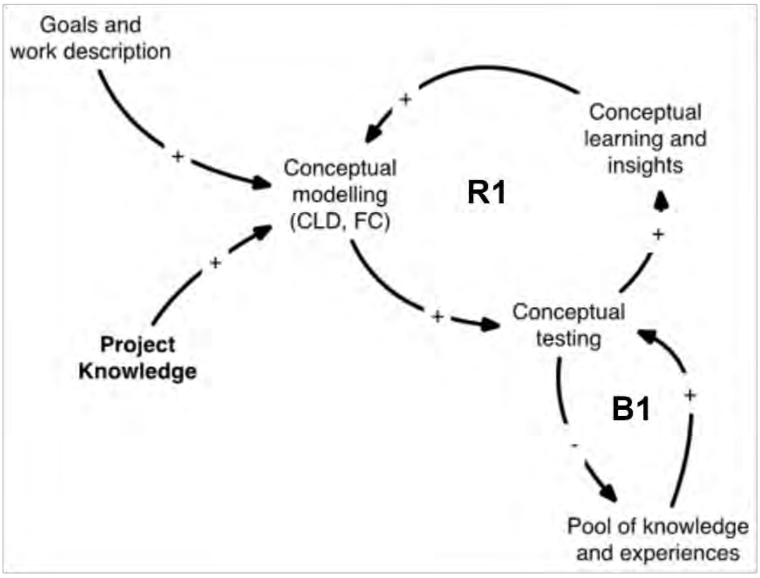


Figure 4. Simple learning loop CLD for System Analysis.

System Dynamics Modeling

System Dynamics Modeling (SD) is an important part of not only Serious Game design and development, but also the implementation of the virtual interactive real-time game engine. I will not show the complete SD learning loop here, but will instead give a simple example of how the SD works. The System Analysis CLDs will be implemented in SD Modeling software, such as Stella Architect by ISEE Systems. A simple example of a CLD and the matching Stella Architect modeling is shown in Figure 7 below. The example shows a stock or accumulator named *Effect*, which has an *Inflow* and an *Outflow*. The *Cause* is connected to the inflow of the *Effect*, and the *Effect* has feedback on the *Cause*. The *Cause* also leads to *Something else*, which again affects the *Outflow* of the *Effect*.

In the SD, we can test input values and balance the game. Once this has been done and the simulation works as planned, the Stella Architect code can be exported as pseudo code, translated to C++, and implemented in the game engine. The export pseudo code screen in Stella is shown in Figure 8 below. This approach has many advantages, such as modeling in real time directly in the game engine, optimizing the game during gameplay, and a higher modeling speed because the code is now native in C++. The modeling code can also be modified during gameplay, without having to return to Stella to run modeling again.

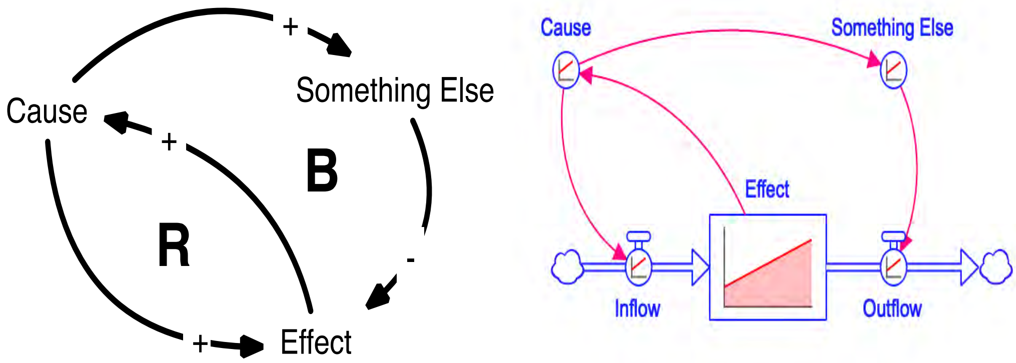


Figure 7. The System Analysis CLD and System Dynamic Modeling.

```
{ VERSION 1.6.2 }

{ INITIALIZATION EQUATIONS }
: s Boss_Health = 1000
: s Party_Health = 0
: c Boss_DPS = RANDOM(0,100)
: c Damage_dealer_DPS = RANDOM(0,50)
: c Healer_DPS = RANDOM(0,0)
: c Healing_Rate = RANDOM(0,30)
: c Tank_DPS = RANDOM(0,10)
: c Tanks = 2
: c Healers = 2
: c DPS = 2
: c Party_DRed = (Tanks*0.25 + DPS*0.1 + Healers*0.0)/(Tanks+DPS+Healers)
: f Party_Healing = IF(Party_Health = 0) THEN 0 ELSE (Healers*Healing_Rate)
: f Party_Initial_Health = PULSE((DPS*20 +Healers*10 +Tanks*30),0,0)
: f Boss_DMg = IF(Boss_Health = 0)THEN 0 ELSE(Boss_DPS-(Boss_DPS*Party_DRed))
: f Boss_Healing = 0
: c Boss_DRed = 0.2
: c Party_DPS = IF(Party_Health = 0) THEN 0 ELSE( DPS*Damage_dealer_DPS + Tanks*Tank_DPS + Healers*Healer_DPS)
: f party_Dmg = Party_DPS-(Party_DPS*Boss_DRed)
: c succes? = IF(Boss_Health = 0) THEN 1 ELSE(0)

{ RUNTIME EQUATIONS }
: s Boss_Health(t) = Boss_Health(t - dt) + (Boss_Healing - party_Dmg) * dt {NON-NEGATIVE}
: s Party_Health(t) = Party_Health(t - dt) + (Party_Healing + Party_Initial_Health - Boss_DMg) * dt {NON-NEGATIVE}
: c Boss_DPS = RANDOM(0,100)
: c Damage_dealer_DPS = RANDOM(0,50)
: c Healer_DPS = RANDOM(0,0)
: c Healing_Rate = RANDOM(0,30)
: c Tank_DPS = RANDOM(0,10)
: c Party_DRed = (Tanks*0.25 + DPS*0.1 + Healers*0.0)/(Tanks+DPS+Healers)
: f Party_Healing = IF(Party_Health = 0) THEN 0 ELSE (Healers*Healing_Rate) {UNIFLOW}
: f Party_Initial_Health = PULSE((DPS*20 +Healers*10 +Tanks*30),0,0) {UNIFLOW}
: f Boss_DMg = IF(Boss_Health = 0)THEN 0 ELSE(Boss_DPS-(Boss_DPS*Party_DRed)) {UNIFLOW}
: c Party_DPS = IF(Party_Health = 0) THEN 0 ELSE( DPS*Damage_dealer_DPS + Tanks*Tank_DPS + Healers*Healer_DPS)
: f party_Dmg = Party_DPS-(Party_DPS*Boss_DRed) {UNIFLOW}
: c succes? = IF(Boss_Health = 0) THEN 1 ELSE(0)

{ TIME SPECS }
STARTTIME=0
STOPTIME=12
DT=0.25
INTEGRATION=EULER
RUNMODE=NORMAL
PAUSEINTERVAL=0
{ The model has 19 (19) variables (array expansion in parens).
  In root model and 0 additional modules with 0 sectors.
  Stocks: 2 (2) Flows: 5 (5) Converters: 12 (12)
  Constants: 5 (5) Equations: 12 (12) Graphicals: 0 (0)
  }
```

Order: By Module | Order of execution | By Sector | A-Z | Keep flows with stocks | Export to File
Show: Units | Documentation | Annotations | Used by | Values | Export to Clipboard

Figure 8. An example of the Stella pseudo code that is exported to the game engine.

The System Dynamics Modeling loop

The next step is to add the learning loop for the System Dynamics Modeling (Haraldsson, 2005; Sverdrup et al., 2021). This loop is similar to the System Analysis loop but involves finding numbers or *Model parameters* and using these in the *Quantitative model*. As in System Analysis, we perform testing, but this time it is *Quantitative testing*. This provides us with *Quantitative learning*, which forms the loop R2 (Figure 5) and makes us go back to *Quantitative modeling*, change the values, and run the simulation again. We can also go all the way back to the conceptual System Analysis loop to improve on the design and perform more System Analysis. The loops show that we collect qualitative observations in the System Analysis loop, and quantitative observations in the System Dynamics loop, as indicated in loop B2. There we can observe that the quantitative testing is dependent upon the available observations and data. The B2 loop limits the quantitative testing and the ultimate learning from the quantitative modeling. The two loops are connected by the arrows from *Conceptual modeling* to *Quantitative modeling* and the feedback arrow from *Quantitative learning* to *Conceptual learning*. Together they form a third loop, R3, which connects the overarching learning process between the conceptual part and the quantitative part (R1+R2). We now have three reinforcing and two balanced loops in the CLD (see Figure 5).

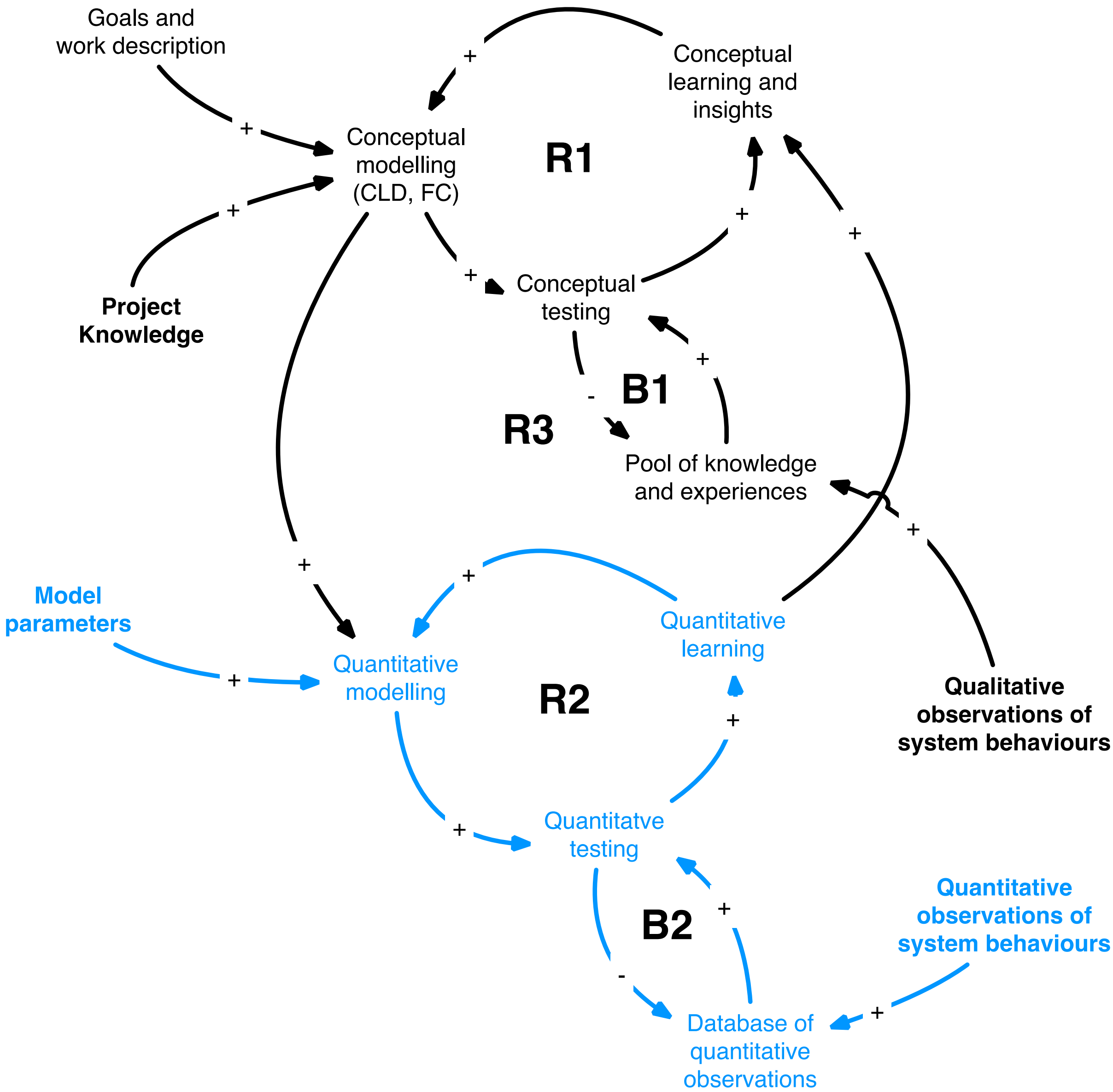


Figure 5. The System Analysis (black) and System Dynamics Modeling (blue) loops connected with a third loop (R3)

The System Game design learning loop

Together, the loops illustrated in Figure 5 constitute the System Thinking part of the learning loop for Serious Game design. However, we also need a loop for the implementation part of the Serious Game. This is shown in *red* in Figure 6 below. The implementation starts when we have learned enough from the other two loops – i.e., the collective behavior from R3.

Quantitative learning leads to where the game development actually begins – *Implement game*. This leads to *Design solutions and test* which again leads to *Challenges* that feed back into the System Thinking loops R3. In Figure 6, this loop is named R4. Together, the loops R3+R4 constitute the design process of the Serious Game implementation, which is only limited by B3 – *Pool of knowledge and experience*. Figure 6 is shown in a wider context in Figure 11, where skills, such as game design skills, psychology skills, art skills, programming skills, etc., are also added into the loop at the places where they are needed. All in all, both figure 6 and figure 11 show the learning loop for the entire Serious Game design process.

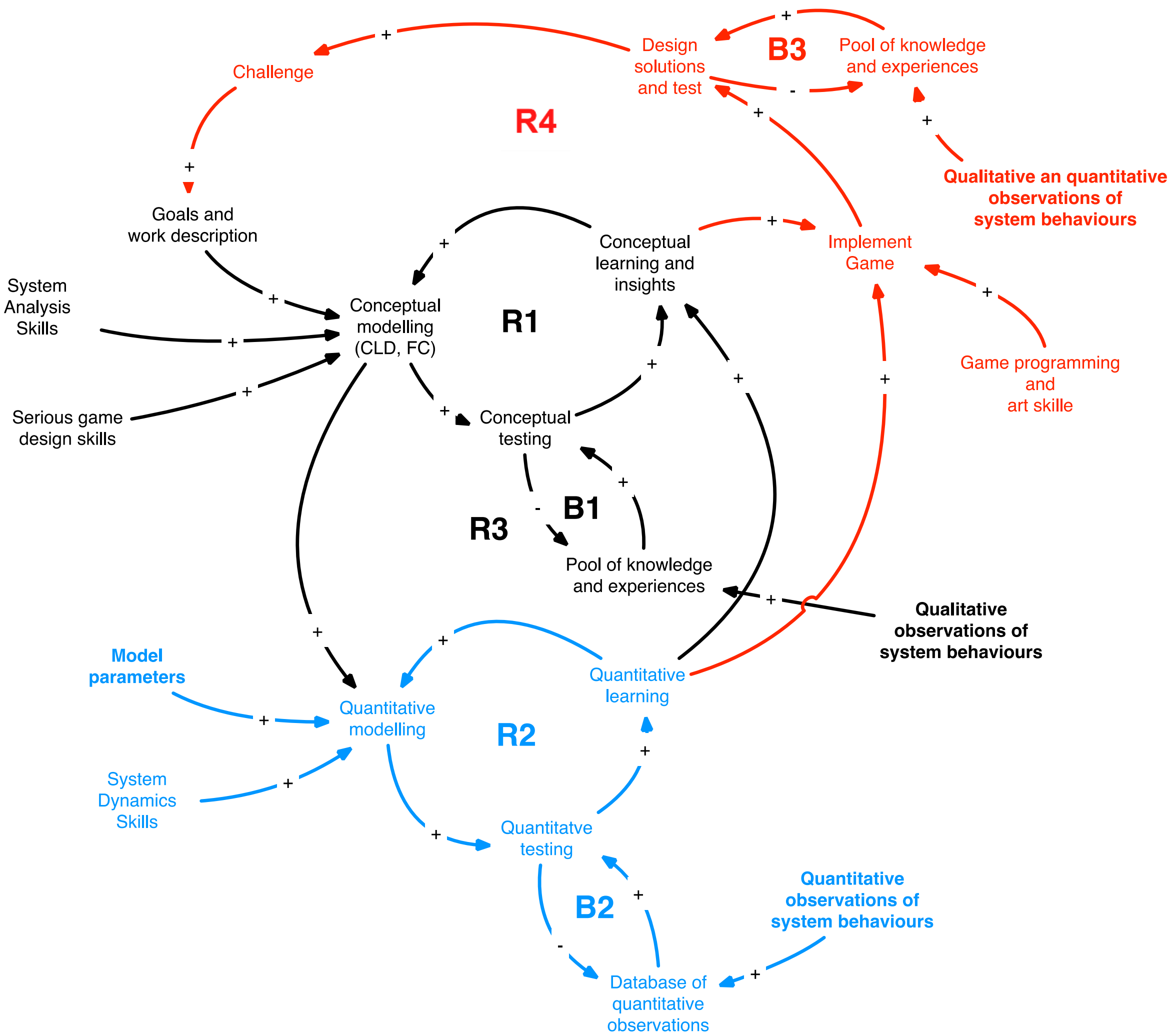


Figure 6. The Serious Games Learning Loop CLD with System Analysis (black), System

Dynamics Modeling (blue), and Serious Gaming implementation (red) learning loops. An implementation loop is added that feeds back to System Analysis.

Serious Gaming and implementation

The Fifth Discipline by Peter Senge (1990) describes five disciplines that need to work together to create a learning organization or team. Four of these disciplines – Personal Mastery, Mental Models, Shared Vision, and Team Learning – are integrated through System Thinking, which he names the fifth discipline. His book is well-known and constitutes a good way to discuss System Thinking and Serious Game design with people and academics who are not well-acquainted with either of these disciplines. This includes educationalists and psychologists, with whom it is important to communicate during the Serious Game design and development process. The disciplines that Senge presents and connects with System Thinking are described briefly below.

Personal Mastery

This discipline is about personal development – to continuously clarify and deepen our personal vision. It is also about focusing energy, the development of patience, and seeing reality objectively, which enable us to focus on achieving the things that matter most to us in life. This is possible through a commitment to lifelong learning.

Mental Models

Mental Models are images, assumptions, or generalizations that influence how we see the world and how we act on this. We must be willing to continuously rewrite these models so that they are "rewritten" in accordance with things newly learned.

Shared Vision

Shared Vision is about creating a genuine future goal or vision that can be shared among the group or organization. This is a vision that makes us learn because we want to, and not because we are told to.

Team Learning

In modern organizations, teams are the learning unit and not individuals. This means that organizations cannot learn unless teams can learn.

System Thinking

System Thinking is the conceptual framework that helps us to gain an overview of how parts fit together into a whole. It can be used to clarify systems, but also to clarify how several systems work together to become a bigger system.

Serious Game design

Serious Games are often simulations, such as flight simulators, training simulators, or medical simulations, where pedagogy or psychology is added to ensure good learning, and competition and reward elements are added to turn it into a game. Such simulations are useful learning tools in many areas, such as natural science, social science, economics, healthcare, and engineering. Serious Games can, however, be strengthened by adding gamification. Gamification is a relatively new, rapidly expanding, discipline that uses knowledge and research from other fields such as game design, pedagogy, and psychology to increase the learning potential in Serious Games. Karl Kapp defines gamification as 'using game-based mechanics, aesthetics and game thinking to engage people, motivate action, promote learning, and solve problems' (Kapp, 2012). The goal is to make boring learning activities more fun and engaging by introducing game mechanics into Serious Games. A collection of research on gamification shows that most studies of gamification find that it has positive effects on individuals (Hamari et al., 2014), although individual and contextual differences do exist (Koivisto et al., 2015). The steadily increasing quality of the virtual environment also leads to better immersion. Better and more sophisticated game engines allow the user to modify the games, and even build new games. An example of this can be seen in the game Doom, from 1993. This allows game-players to learn more about and participate in the game development process. Many players become modders (players that modify games and often make new games from the parts) and thereby become more knowledgeable and demanding, and game development spreads to a larger part of the game community. This might help to advance the video game industry and make it more innovative. However, Serious Games also build on traditional game design and game elements, which we will discuss briefly below.

Traditional game design

As discussed earlier, the process of creating games has developed from involving a single person to large teams with hundreds of people working for years to finalize the game. The traditional game design process is often visualized by the following diagram:

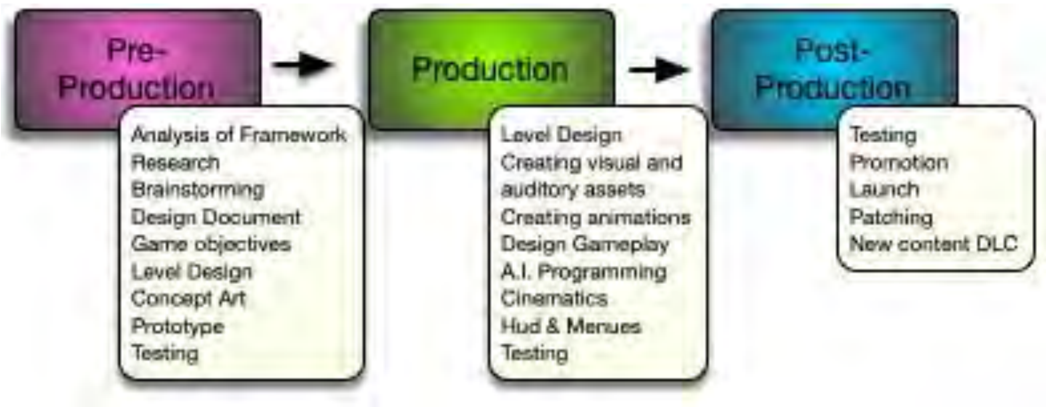


Figure 9. The traditional game design process.

The process is divided into three phases:

The pre-production phase is characterized by studying the prerequisites and brainstorming to come up with a great idea. However, not all ideas are useable, and the team needs to analyze the game objectives and undertake some concept art and level design (design of the game architecture), and even make a prototype, to test to whether the idea works. The production phase is costly, and worst designs are abandoned after the pre-production phase.

The production phase involves creating art and doing the programming for the game. The art element involves creating more detailed concept art, modeling assets and characters, texturing, animation, lighting, cinematics, etc., while the programming element concerns programming interactivity, movements, menus, and artificial intelligence (AI). Game design is also a strong component in this phase and is concerned with creating good gameplay. To be able to do that, a lot of testing is also required in this phase.

The post-production phase involves the promotion and launch of the game, but also ironing out bugs and further balancing the game. The latter often takes place through user testing. Downloadable content (DLC) is created and made available to enhance the game and fulfil user requests, and to keep them playing as long as possible.

Gameplay is often viewed as “the secret ingredient” that makes a game fun to play. Richard Rouse has put it this way: ‘A game’s gameplay is the degree and nature of the interactivity that the game includes, i.e. how players are able to interact with the game-world and how that game-world reacts to the choices players make’ (Rouse, 2005, xx). Gameplay is connected to interactivity, which is concerned with how the user interacts with the game. Furthermore, gameplay must focus on the choices given to the player and the ramifications of the chosen alternatives (Rouse 2005, xxi). Video games need to have systems that can handle the players’ inputs and outputs in a good way. As the video games hardware becomes more powerful and the gameplay becomes more advanced, the need to design the game systems in a systematic and scientific way increases. This is what we seek to remedy by introducing System Thinking in video game design.

Game and gamification elements

Game or gamification elements are the elements used to make the game. There are many versions, but below are a few from Karl Kapp’s book on gamification (2012):

- Abstractions of Concepts and Reality. Abstraction helps the user get an overview of a reality, event, or some idea, and makes it easier to understand the concept and leaves space to see causes and effects.
- Goals. The goals add purpose and focus, and enable feedback on progress, quality of play, and how close the player is to the goals. Goals also define rules, and Rules of Play (Salen et al., 2004) define Operational, Constitutive, Implicit, and Instructional rules.
- Conflict, Competition, or Cooperation – Game elements which is self-explanatory.
- Time. Time can be compressed and expanded to showcase special consequences or actions more quickly.
- Reward structures. Badges, Points, Rewards, Leaderboards, etc. Often given early in the game to trigger interest, and not kill intrinsic motivation that builds up later.
- Feedback. Feedback to the player can take many forms, including tactile, inviting, repeatable, coherent, continuous, emergent, balanced, surprising, etc.

- Flow. Games should not be too easy and not too hard, but should follow the flow channel in between.
- Storytelling. Games enable storytelling that can add characters, plot tension, and resolution. The Hero's Journey is a good model to help make the story.
- Esthetics. Art, beauty, and visual elements are part of every game, and are becoming better and more lifelike all the time. Regardless of the game, the esthetics play an important role.
- Replay/do over – a very important element of games that plays a huge role in learning. In games, failure is an option.

In order to trigger learning, Serious Games that are well-made should link these elements to psychology, as described below.

Gamification psychology

Gamification psychology concerns the creation of motivation. Motivation is normally divided into *intrinsic* and *extrinsic* motivation. Intrinsic motivation is when things are done for their own sake, for the enjoyment or learning that this gives, or for the sense of accomplishment that it creates (Lepper, 1988). Extrinsic motivation, on the other hand, concerns receiving a reward or avoiding punishment (Lepper, 1988). Whilst intrinsic motivation comes from within a person, extrinsic motivation is triggered externally. Motivational models describe how we can trigger extrinsic and intrinsic motivation. As an example, we will describe two models here and show how they can be linked to game elements. The ARCS model was developed for a classroom setting, and the self-determination theory was specially developed to describe elements that trigger intrinsic or inner motivation. There are of course many other models, such as Malone's theory of intrinsic motivation instruction (Malone, 1981), Lepper's instruction design principles for intrinsic motivation (Lepper, 1988), or the taxonomy of intrinsic motivation (Malone & Lepper, 1987). The descriptions of the models lean on those provided by Kapp (2012), as this book constitutes the main element of the students' curriculum in gamification, and thereby forms the basis for many of the student examples in this PhD thesis.

The ARCS model

This model is well-known within the field of instructional design and was developed by John Keller (Keller, 1987). The letters in the name represent Attention, Relevance, Confidence, and Satisfaction.

Attention involves making the learners interested in the content. The model focuses on doing this in several ways: perceptual arousal, which is about showing related examples; inquiry, which involves presenting problems to the learners that they are interested in solving; or variability, which involves maintaining interest through periodic variation of the content delivery methods.

Relevance can be established through goal orientation, matching the motives, or familiarity. Goal orientation involves making the learner aware of how they can benefit from the goal now and in the future, and illustrates the importance of achieving the goal. Matching the motives can be achieved by matching the motives of the learner with the motives of the instructors, which can include affiliation, risk-taking, power, and achievement, etc. Familiarity concerns showing how the learning is connected to the knowledge of the learner and, if possible, modeling the result of the learning.

Confidence – learners tend to be more motivated if they think they can achieve success (Kapp, 2012). This can take place by stating the learning requirements and expectations at the beginning, and by letting them know how much effort and time they will have to devote to the problem in order to succeed. Learners can also be given opportunities to experience success and to work through the problem via small milestones that they can master. It is also important to give feedback and personal reinforcement, to help make them feel they are in control.

Satisfaction refers to learners' confidence that learning has value and is worth the effort. This can be achieved by giving learners opportunities to successfully use what they have learned in real or simulated settings. Encouragement and reinforcement should be given throughout the learning process. It is important to maintain a sense of equity between the learners, and to maintain a standard measure of success.

The self-determination theory (SDT)

The self-determination theory is a macro theory that aims to explain inner motivation to perform a task or activity (Ryan et al., 2000). One important aspect of the theory is that it has several sub-theories, such as the cognitive valuation theory, which proposes 'that events and conditions that enhance a person's sense of autonomy and competence support intrinsic motivation, and that factors that diminish perceived autonomy or competence undermine intrinsic motivation' (Kapp, 2012).

The SDT has three important elements, of which the first is *Autonomy,* which concerns a person's sense of determining the outcome and being in control of their actions. The next is *Competence*, which is defined as a need for challenges and a sense of mastery, which according to the cognitive evaluation theory, supports intrinsic motivation. Acquiring a new skill or being appropriately challenged are examples of experiencing competence. The last element, *Relatedness*, is the sense of being connected to others. This can take place in many ways, such as by playing a game together in the real world or being in an online multiplayer game with others.

Kapp (2012) points to research that has shown that 'the psychological "pull" of games is largely due to their capacity to engender feelings of autonomy, competence, and relatedness, and that to the extent they do so they not only motivate further play, but also can be experienced as enhancing psychological wellness" (Ryan et al., 2006). The goal in Serious Games and gamification is to ensure that game elements trigger motivation factors. Figure 10 below shows an example CLD of this, based on the ARCS (red) and SDT (blue).

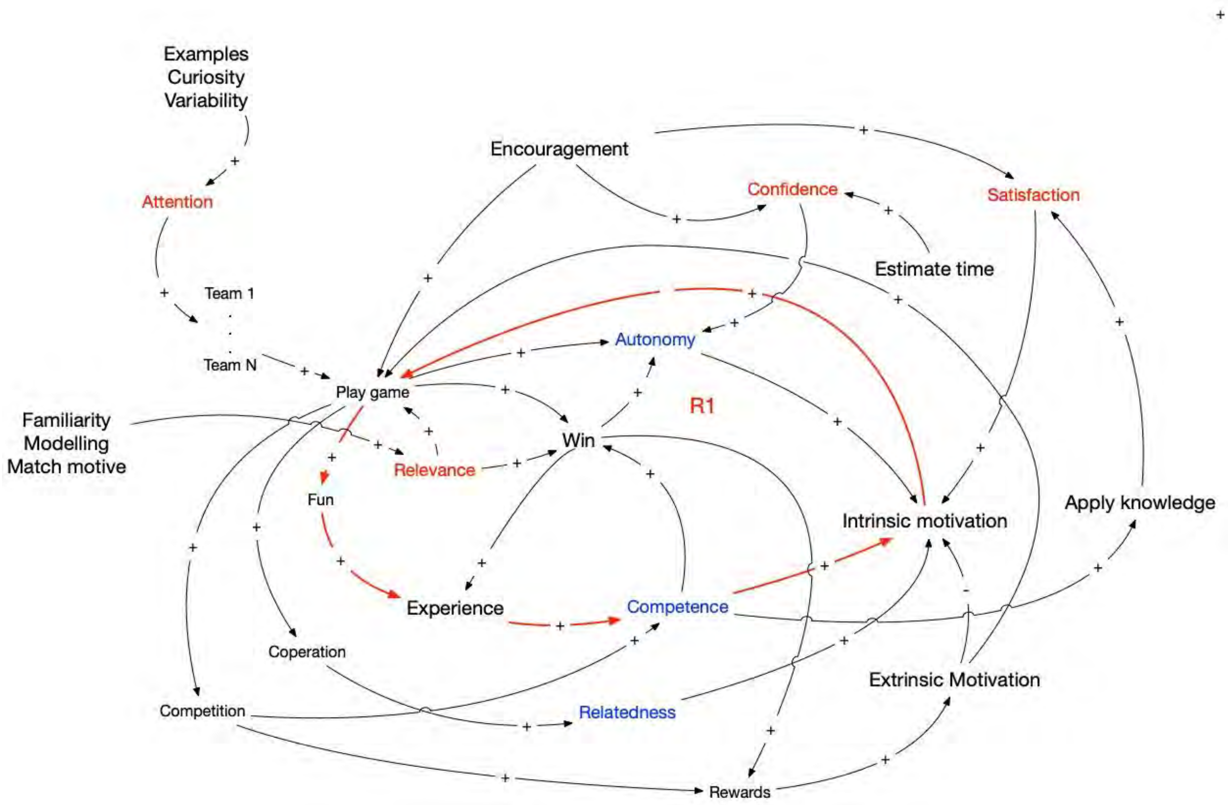


Figure 10. A CLD for game elements that can trigger elements of the ARCS and SDT motivation models.

Learning methodologies

Theoretically, the course design in all three projects in the thesis uses a problem-based approach (PBL) (Barrows, 1980), together with ideas found within sociocultural theories of situated learning (Lave & Wenger, 1991; Wenger, 1998), which are explained in more detail below.

Problem-based learning

For all projects in this thesis, the organization of the learning process shares the same basic elements, as they are loosely based on problem-based learning. In the first project, the programming course, this was a deliberate decision – we wanted to split the game development process into smaller problems that the students could mostly solve on their own. When we started teaching System Thinking in 2012, we had help from teachers from Lund University, who also took a problem-based approach in their teaching. It was then natural to select the same approach for the gamification course that started in 2014.

Problem-based learning was first implemented in medical schools in the late 1960s by Howard Barrows and his colleges (Barrows, 1980). It was originally implemented with small student groups of six to ten students, each group with a tutor who guided the students in self-study. Very little teaching was done by the tutor. The focus of these groups was to solve real-world problems related to the profession the students were about to study. However, PBL has since become established as an independent learning method, especially in higher education (Pettersen, 2005; Dochy et al., 2003). Leaning on Pettersen (Nordby & Karlsen, 2014), the principles of PBL could be described as follows:

1. The problems are based on practice-related descriptions and case studies.
2. Students receive support, assistance, and monitoring from a tutor.
3. Students engage in learning activities and develop ways of working and learning together with practical problem-solving and logical reasoning.
4. Students are autonomous in their learning and study activities.
5. The teaching, curriculum, and actual courses are organized so as to allow and maintain cross-disciplinary and inter-disciplinary approaches.
6. The study program that the students follow facilitates their early contact with future and possible clients/users, and authentic tasks and challenges are set to qualify the students for their future professional lives.

A PBL model will normally follow these seven steps (Maastricht University, 2013):

1. Clarify difficult terms.
2. Define the problem.
3. Brainstorm.
4. Take stock of and analyze the possible solutions provided in step 3.
5. Formulate learning objectives.
6. Self-study.
7. A post-discussion.

Steps 1 to 5 comprise the initial phase, in which the student groups define the learning goals of the given case. They first do this through brainstorming, and later sort and identify which theory is relevant to study to solve the case. The students perform steps 1 to 5 together as a group, but in step 6 they work individually with the study goals and implementation. In step 7, they meet again to discuss their experiences and the results from their individual work.

Situated learning

The situated learning theory (Lave & Wenger, 1991; Wenger, 1998) suggests that learning is integral to participation in "communities of practice". Although the theory is presented as an 'analytical viewpoint on learning, a way to understand learning' (Lave & Wenger, 1991, p. 40), it also presents ideas on participation and interaction that are worth considering. However, just grouping people together is not enough. Wenger lists 14 indicators (Wenger, 1998, p.73) for the community of practice, and emphasizes a few such as 'sustained mutual relationships – harmonious and conflictual […] shared ways of engaging in doing things together […] knowing what others know, what they can do, and how they can contribute to an enterprise […] mutually defining identities […] specific tools, representations, and other artefacts [and] local lore, shared stories, inside jokes, knowing laughter' (p. 125). The learning is considered to take place through shared undertakings, and can be accessed and facilitated through the relations between the newcomers and the old-timers in the community, and between its range of activities, identities, and artefacts. The key to participation is to gain access to everything that membership entails, where prolonged or vivid membership leads to a more mature or full participation.

The Serious Games learning loop

Figure 11 shows the learning loop CLD from Figure 6 in a wider context with the incorporation of game design, problem-based learning, situated learning, and psychology. This figure is where it all comes together. Although the loop structure is the same as in Figure 6, we see that the System Science/System Thinking design loops R3+R4 now also integrate System Thinking skills, Game design skills and Motivation psychology skills. This shows in more detail that System Science/System Thinking is more than a way to make a design drawing and simulation for the Serious Game design; it is also the language and a tool set to integrate different skill sets and disciplines and bind them all together. Situated learning and the community of practice bring together individual skills, team skills, and mental models (from the R2 modeling loop), and helps create a shared vision as described in *The Fifth Discipline* (Senge, 1990). Team learning and individual learning are enhanced by the problem-based learning approach, where the work and learning is done both individually and in groups. The learning in the R3 (R1+R2) and R4 loops is also enhanced through the established *Community of practice* and the *Problem-based learning* methodology.

As earlier, the balanced loops (B1, B2, and B3) regulate each loop (R1, R2, and R4), and this is the key to when to progress to the next loop or go back to previous loops, or when progress has been good enough to *Sell game*.

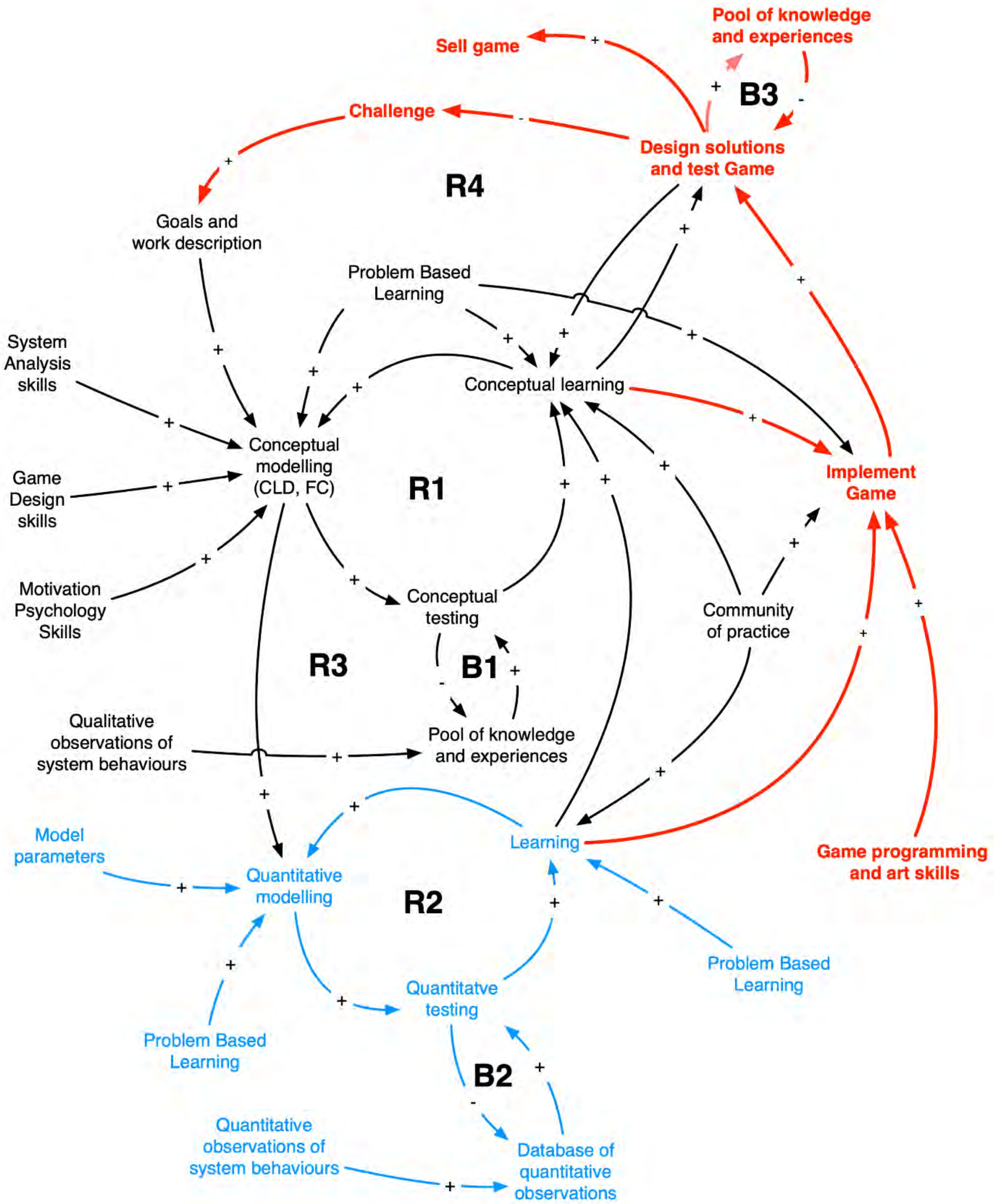


Figure 11. The Serious Games learning loop CLD includes game design, pedagogics, and psychology.

Service innovation and systems as part of the Serious Games learning loop

There are many definitions of service (eco-)systems. Spohrer et al. define them as 'value-co-creation configurations of people, technology, value propositions connecting internal and external service systems, and shared information (e.g., language, laws, measures, and methods)' (Spohrer et al., 2007). Maglio et al. see them as 'a configuration of people, technologies, and other resources that interact with other service systems to create mutual value' (Maglio et al., 2009), and Lusch and Vargo define this as a 'relatively self-contained, self-adjusting system of resource-integrating actors connected by shared institutional arrangements and mutual value creation through service exchange' (Lusch & Vargo, 2014).

To understand the service ecosystems, we can first refer to Giddens, who states that 'resources and institutions form inseparable, interdependent relationships, in continuous interaction through socialization'. Giddens argues that these interactions create a duality of structures in the value co-creation process (Giddens, 1979, 1984). Tronvoll explains service ecosystems as the duality of structures containing institutions (referring to institutionalized schemas of values, traditions, norms, and rules) and resources (resources used in the co-creation of value), where the duality consists of continuous interaction between institutions and resources that subsequently creates and recreates the service system aiming for value co-creation (Edvardsson et al., 2013). Tronvoll further links the service system to service innovation by designating it as an appropriate framing for studies, as it moves away from perspectives traditionally "rooted in technological product inventions". He then uses an "S-D logic" lens to conceptualize service innovation as a new and useful 'process of application of specialized competences (knowledge and skills) through deeds, processes, and performances for the benefit of another entity or the entity itself' (Vargo & Lusch, 2004), and with a business effect: 'the next practices of innovation must shift the focus away from products and services and onto experience environments – supported by a network of companies and customer communities – to co-create unique value for individual customers' (Prahalad & Ramaswamy, 2003, p. 12). S-D logic is defined by the five axioms (Vargo & Lusch, 2004, 2016):

Axiom 1/FP1: Service is the fundamental basis of exchange.

Axiom 2/FP6: Value is co-created by multiple actors, always including the beneficiary.

Axiom 3/FP9: All social and economic actors are resource integrators.

Axiom 4/FP10: Value is always uniquely and phenomenologically determined by the beneficiary.

Axiom 5/FP11: Value co-creation is coordinated through actor-generated institutions and institutional arrangements.

Edvardsson and Tronvoll (2013) then define service innovation as ‘changes in service system structure due to a new configuration of actors, resources and interactions, resulting in new practices that are valuable enough for the involved actors to make it sustainable’. Lusch & Vargo (2014) describes innovation as developing more effective value propositions for participating in beneficiaries’ resource-integrating, value-creating practices, through service.

Using this new definition of service system, he defines service innovation, when using an S-D logic lens and a service and social system foundation, as changes in structure sourced in either a new configuration of resources or a new set of schemas, resulting in new practices that are valuable for the actors in a specific context (Edvardsson & Tronvoll, 2013).

So which structures, systems, systemic features, processes, and outputs are elucidated or not considered in the service innovation literature concerning systems? Although I cannot find many components that are missing from the literature, I find that most of the components described lack details of both what is contained and how they are connected. For example, what is the cause-and-effect relationship between components in the service system? What flows through the system? How does the system develop over time? Etc. The System Analysis is actually missing from these studies.

When it comes to the strengths and weaknesses of these perspectives in my research project, I would say that, in many ways, the service innovation literature underpins my work and research. I can easily see Serious Game development, as described in my PhD thesis, as both goods and services innovation. I do however find that System Thinking is missing from the course literature as a scientific tool to clarify the details, and to systemize and test the overall qualitative connectivity and relational dependencies between actors, institutions, resources, and schema (values, traditions, norms, and rules).

As stated in the introduction, I find that the course literature contains little detail on how to implement service innovation in practice in companies. I can also see that it contains very little quantitative material that can be used in System Dynamic simulations to test the innovation implementation. However, both the literature and my PhD thesis indicate that elements of service innovation, such as research into pedagogical and psychological processes between actors inside the companies, need more detailed qualitative and quantitative research.

This is also the case for schema (values, traditions, norms, and rules). Schema could probably benefit from being merged with theories of situated learning and communities of practice (Lave & Venger, 1991). In Serious Game development in particular, we need more detailed research into how to achieve and measure motivation and learning to prove that the suggested processes work as intended. In general, quantitative research is needed in all aspects of innovation in order to simulate value outcomes on a computer. In the service innovation literature in general, a methodology for testing service innovation processes is also very rare, or non-existent.

The literature naturally has positions, controversies, and tensions, but I find them of little relevance to my PhD thesis project. For example, game development for online games often continues throughout the game's life cycle. I suggest that the game development process is a combination of product and service innovation, and in the case of online games, the service innovation often dominates. This indicates a synthesis or integrative view (in contrast to assimilation or demarcation) that emphasizes value and characterizes service innovation as multidimensional, with no dominant paradigm that encompasses the dynamics of innovation in all forms (technological and non-technological), and in both goods and services. Building on the Serious Games learning loop previously presented in Figure 11, Figure 12 below shows how systemic service innovation can be incorporated. The System Analysis part is a reinforcing loop, whereby the system is logically (or qualitatively) designed. It usually starts with a value proposition from the key actor that is worked on together with co-actors (partners and customers) but can also involve propositions made in crowdsourcing, competitions, or other similar activities in virtual apps or environments, or in real life. In the loop, the actors and network and the environment/social interaction (schema/structure or community of practice) influence the innovation design. The result is tested conceptually by observing the design. Lessons learned from this might lead to changes in the initial design, and thereby circle around the loop.

Next is the System Dynamics loop (figure 12), where the results of the System Analysis are simulated. This entails finding numerical values for the model parameters and running several scenarios to test the simulation model. When the model works according to the design, it can be used to predict results. Again, actors/network and social environment influence the tests, and the results might lead back to changes to the System Analyses, to improve the design. The last stage is implementation. This is where we create value for both customers and service providers (and network). When implementations are tested on the public they can lead to crowdsourcing, competitions, etc., but also a redesigning of the service innovation.

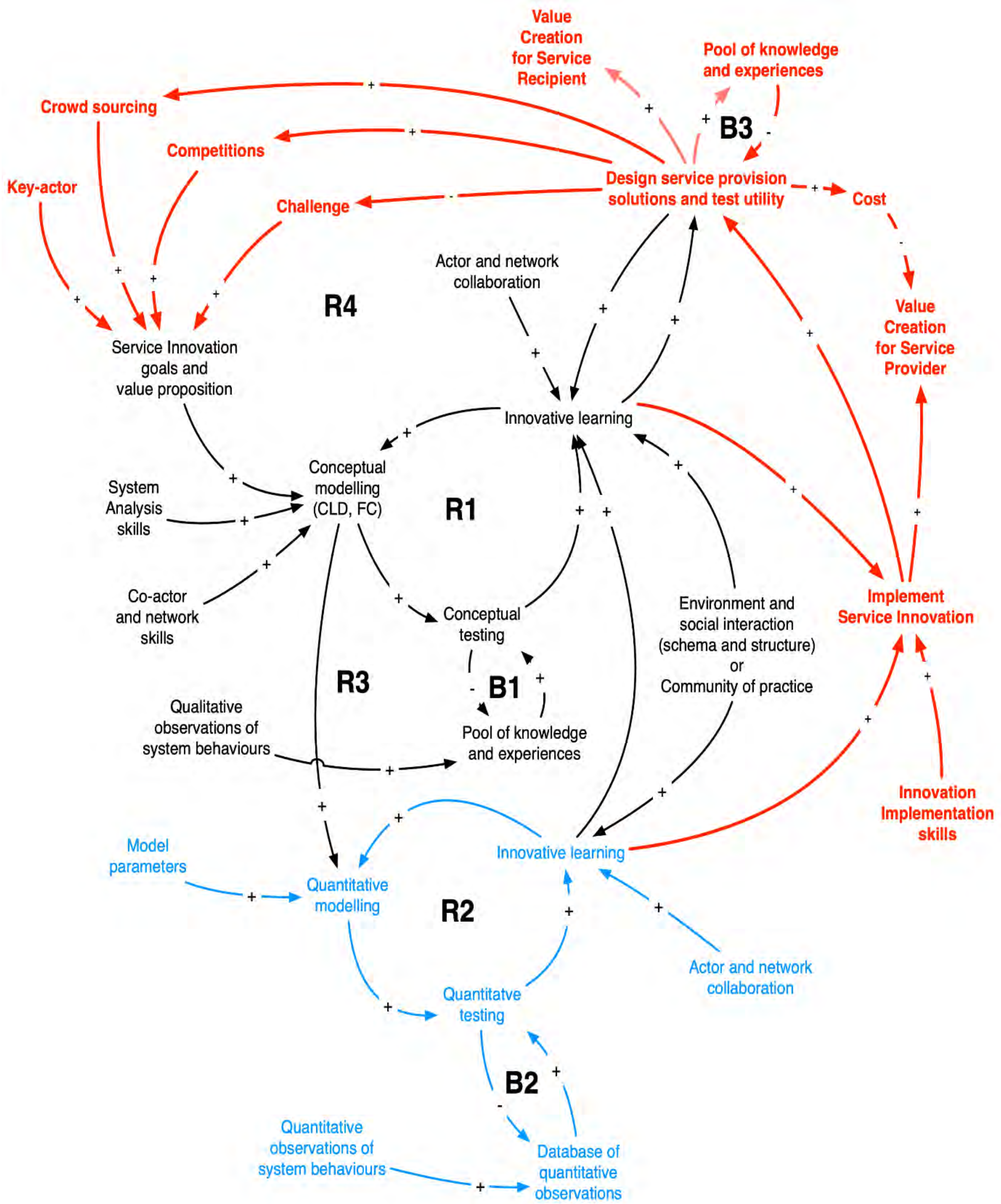


Figure 12. The Serious Games learning loop shown earlier, now incorporating service innovation.

Results

The dissertation articles and projects

This chapter will summarize the articles. There is a small overlap, as some projects are also mentioned and briefly described in other articles than those summarized under projects. The four dissertation articles are:

1. Teaching “hardcore science” to arts and design students: Reflecting on the development of a basic programming course (Nordby & Karlsen, 2014). Nordic Journal of Art and Research (peer-reviewed).
2. The art of gamification: Teaching sustainability and System Thinking by pervasive game development (Nordby et al. 2016). The Electronic Journal of e-learning (peer-reviewed).
3. System Thinking in Game Design (Nordby et al.). Not yet published.
4. System Thinking in Gamification (Nordby et al., 2023). Will be published in 2023 in SN Computer Science. Springer Nature.

The articles are about the following projects (the dates refer to the year(s) the project was done):

Project 1 (Article 1): Teaching “hardcore science” to university-level game students (2006 – 2011). Teaching programming, math, and physics to game students using game development, problem-based learning, and situated learning as the teaching methodology.

Project 2 (Article 2): System Thinking and pervasive gaming for teaching (2013)**.** Teaching System Thinking to game students, teachers, and children.

Project 3 (Articles 3 and 4): System Thinking in game and gamification design (2015–present). Teaching game and gamification design with System Thinking to game students.

The appendix articles

The appendix has four extra articles – two about Project 1 and two about Project 2. They are not part of the PhD thesis but are included to give a deeper insight and more examples of these two projects. All four are conference articles. They are not peer-reviewed but have given publication points.

Related to Project 1.

1. Teaching “hardcore science” to arts and design students of the gaming generation: Reflecting on how to use game development to teach programming, System Thinking, mathematics, and physics (Nordby, 2015). Published in EDULEARN Barcelona proceedings.
2. The power of game development in learning: Why is game development good learning machines (Nordby, 2016). Published in EDULEARN Barcelona proceedings.

Related to Project 2.

1. “The Best School Day Ever” – teaching sustainability and System Thinking by pervasive game playing (Nordby, Øygardslia, & Sverdrup, 2015). EDULEARN Barcelona proceedings.
2. Teachers as learning game designers: Can elementary school teachers with no gaming development background really gamify their own teaching? (Nordby & Nordseth, 2016). EDULEARN Barcelona proceedings.

Synthesis of articles

This chapter will give a summary of the three projects and the main articles written about them.

Article 1: Teaching "hardcore science" to arts and design students

Teaching "hardcore science" to arts and design students: Reflecting on the development of a basic programming course (Nordby & Karlsen, 2014). Nordic Journal of Art and Research (peer-reviewed). Published in 2014. (*Text from this article is used in this chapter without further references.)* The article is described in three parts: the first part introduces the project, the second part gives the introduction and the hypothesis, and the final part presents the results.

The project

The project was started in 2006 out of necessity – up to 50 percent of the students had failed the traditional programming courses in our newly created game education program. In 2006, I took over teaching the first programming courses aimed at solving the problem. At first, we did not have a clear view of what to do, so we started with the following changes, with the aim of experimenting with the details:

1. The programming environment was changed to Adobe Flash and we started to teach the students ActionScript instead of C++. The reasoning behind this was to make the programming more visual. Adobe Flash interpreted the ActionScript code on the fly, and the students could see visual results of their programming right away.
2. We wanted to allow the students to create a game as soon as possible. The reason was to lose no time in channeling their motivation to this topic, as the majority of the students on this course were game students, and this was what they had come here for! At first, we did not know what kind of game to make, or even whether it was possible to make a game in just one semester without any previous programming experience. However, we started experimenting, and tried out several different game types to see which one could fit the task.
3. We also began to investigate different pedagogical approaches, and gradually tested out several forms of case and problem-based learning (PBL) during the course in 2006/07.

During 2007/08, the project gradually became more formalized. We learned that a platform game would give the students the right balance of challenges to learn basic programming, and that problem-based learning (PBL) was suitable as the basic teaching methodology. Although these basic choices had been made, many details had to be fine-tuned during the following years. As mentioned above, almost all the students on the programming course were game students, coming from three different strands: Virtual Arts and Design, Animation, and Visual Simulation. Only Visual Simulation had students who were interested in programming, whilst the students from Virtual Arts and Design and Animation were mainly arts students. All the game students were in the first year of their studies. In addition, we occasionally had some third and fourth year teacher students joining the courses. On average, 70 students attended the course per term. Most of the students were aged 18-25.

Although the students were clearly part of the gaming generation, many of them did not understand why they had to learn programming. Even worse, there were no admission requirements for the game studies stipulating that they had to have advanced science classes in their upper secondary education. So, the decision to let them create a game was not only to teach them programming, but also to make them understand that "hard science" topics were necessary in game education programs to be able to create computer-based artwork. We found that the game development process gave opportunities to teach other science topics, such as math and physics, as events where they needed to learn this naturally popped up all the time. We did not place many restrictions on the platform game the students were to create, and in fact the only requirement was that it should contain one character who could walk, run, and jump, and would be controlled using the keyboard.

The PBL approach was inspired by similar projects in Finland (see Kinnunen & Malmi, 2005; Nuutila et al., 2005). We divided the students into groups in which they would work and collaborate throughout the course. However, one major difference was that each student had to make their own game. In line with traditional PBL, each group had its own tutor, whom we mainly recruited from higher-level students. We did not completely abandon traditional teaching; brief lectures were given on how to approach the PBL method, as well as how to approach the case. We also provided guided labs for students who needed this.

The platform game development project was divided into smaller cases or steps, in line with the PBL model. The actual number of cases varied from year to year, but an average of eight cases was found to work well. The typical order of the cases was important and is shown below:

1. Design the game and make a storyboard.
2. Design the characteristics of the main character of your game and then make a drawing of this character.
3. Divide the character into layers and animate its walking cycles and jumping abilities.
4. Connect the triggering of the walking cycles to the keyboard arrow keys.
5. Draw the ground of the game and make the main character follow this ground.
6. Make a second character (an enemy) and let it follow the main character.
7. Implement jumping for the main character and design platforms to jump on.
8. Make a starting screen with the necessary menus, as a well as screen to be shown upon winning or losing the game.

It proved important to design the cases to be as open-ended as possible, so that more skilled students could dive deeper into case details, instead of moving on to the next case and thereby moving through the planned steps of the course too quickly.

Introduction, questions, and hypothesis

The article discusses how young people's gameplaying experience should be considered when higher education is designed. In particular, it discusses how a programming class with game students from a broad background in both arts and technology can be designed.

Leaning on this knowledge about the gaming generation and the members' potential learning experience, the programming course was designed to better fit these students' needs. Most of the students who attended the course were typical game students of "the net generation" and had been playing games for years. However, they severely lacked the background skills and theory necessary to create advanced graphics and programming, as well as the knowledge of math and physics necessary to make the game. They often did not understand the need to learn these hardcore topics. This led to the following questions to guide our exploration: How can the "fun" experienced in gameplaying, modding, gamification, and online communities be brought into "boring" topics in higher education? How can gameplaying and game development be used to teach "hardcore" subjects such as programming (and its related math and physics) to arts and design students in a university course on computer programming? How can problem-solving, collaboration, learning by doing, learning just in time, and innovation be implemented as part of such a course?

This led to the hypothesis that game development, combined with the principles of problem-based learning, might create an environment that enhances the student's motivation *to learn programming, as well as the necessary operational and innovative skills and the background math and physics theory*.

Combined with more informal activities, such as making web pages and blogs and visiting online forums, the teaching was expected to proceed more efferently than with traditional teaching methods such as lectures. Traditional problem-based learning and the situated learning theory were introduced. The situated learning theory describes the class as a "community of practice" where "creating a game" serves as a common goal. In this environment, the tutors, as well as the students with previous experience from modding games, served as fuller members of the community. This was a big help when the rest of the students started as newcomers. We also believed that incorporating informal learning in the form of blogging and participating in online communities enhanced the meaningfulness and sense of belonging in the community. The article then presents the detailed design and experience of running the course, as featured in the above project description.

Results

Firstly, the assumption that "making a game" would become a common goal for most of the students was evident at an early stage. The class, and even more the student groups, were good examples of a vibrant "game development" community. The reasons for this could be many:

- Students were able to choose group members or collaborators themselves.
- Students shared tools, such as specific tools, representations and other artefacts.
- The fact that the task was steered towards actually "making a game" rather than learning "how to make a game" increased the motivation to solve the task.
- Students saw where the fragments they studied belonged in the bigger picture.
- Students gained an overview of how all of the specific topics taught fed into creating a larger academic, artistic, and practical context to which they needed to relate, in order to complete their game successfully.

The social environment in class and groups successfully created a "learning by doing" (Dewey, 1916) context, largely through the utilization of PBL, which forced the students to study theory when it was needed to solve the cases. Their theoretical learning happened "just in time" (Gee, 2007) when their motivation was high, because they wanted to solve a game development problem. Below, we provide a couple of examples of such game-related tasks.

When the students had designed their game characters and animated them, they wanted them to be able to jump. For this to happen, they needed to apply an upward force, but when they did so, the character disappeared out of the top of the screen. It was time to teach them about gravity and make them apply gravity to their character, to bring it back to "earth" (the ground). Very few students found it odd to talk about physics and gravity in this situation, and they quickly realized that they needed a solid theoretical foundation on forces and effects to implement a well-functioning solution.

When the students' characters started walking around the screen, the students needed to expand their knowledge of y and x-axes, coordinates, angles, and conversion between degrees and radians. They also eventually needed to learn trigonometry, to decompose movement into components. This is a theory that students usually struggle with when it is taught in the traditional abstract manner. In this context, however, the complex math and physics involved was regarded as comprehensible because it was tied to "concrete" game development challenges. In other words, the "fun" (Tapscott, 2008) and "motivation" aspects were evidently present throughout the course.

It was important to make the student groups creative, which required the groups to be as heterogeneous as possible (Darsø, 2001; Gee, 2007). The group sizes stabilized at 8-10 members after a while, which correlates with the theories both for creativity (Darsø, 2001) and problem-based learning (Pettersen, 2005). The student group tutors were carefully selected from among the best students from the previous year, and all of them put a lot of effort and energy into their work.

In the last years of the project, we increasingly encouraged the students to create good web pages and blogs to promote their ideas and games in online forums and communities. This gave the students arenas in which to construct and display their new-found identities as game developers (Partii & Karlsen, 2010), and gave valuable feedback on their work. In this way, the students established virtual communities of practice (Barab et al., 2004). Most of their blogs and Wiki pages were written in English, so the students were able to discuss and receive feedback and comments. The students were also encouraged to play games online or locally on their PCs, so they could experience for themselves how important emergence is and gain a sense of how the emergence of their games could be improved.

One thing that surprised us was how quickly the students took on identities as game developers. We believe this gave them credibility in online forums and game groups, and motivated them to visit these places and present their findings in the group and classroom.

According to their reports, online peers with knowledge of game development were greatly utilized in many situations.

The higher-level tutors also enjoyed their job. We did not often have to recruit them, as they came to us and wanted to be tutors for the next year's class. They also put a lot of effort into their work. We never paid the tutors for their work, but they obtained study points and a certificate for participating in the research project that they could use when they applied for further studies or work.

We discovered that the highly cross-disciplinary learning in this class was a bit confusing for the students. To help clarify this, we started the next programming course by summing up their learning from this course, and helped them to categorize their learning. It was important that this was done in the next class and not during the present class, because it tends to slow down creativity in the immersive community of practices.

What was also proved right was what critics of problem-based learning have pointed out: PBL does not fit all learners! We sought to remedy this by offering a lot of help, and we also held some lectures and labs for selected students, when necessary. However, the reorganization resulted in the failure rate dropping dramatically to almost zero!

The article ends with some concluding remarks on the students, some of whom followed the gaming path, whilst others followed "outbound trajectories" and chose to use their learned skills in other areas of the art and design world. Others pursued "insider trajectories" and became professional game developers (see, for example, Krillbite, n.d.; Moondrop, n.d.; Roofmoose, n.d.).

One challenge of the programming course was to keep up with the rapid development in the field and to continue to provide the students with fun that corresponded to their virtual and real-world realities. The article is intended to be a contribution showing that it is not just "what" you teach that matters; the "why" and, not least, the "how" of teaching also need to be in line with students' worlds for their educational experience to be considered meaningful.

Article 2: The art of gamification

The art of gamification: Teaching sustainability and System Thinking by pervasive game development (Nordby, Øygardslia, & Sverdrup, 2016). The Electronic Journal of e-learning (peer-reviewed). Published in 2016. *Text from this article is used in this chapter without further references.*

The project

System Thinking has been taught to game students at the Inland Norway University (INN) since 2012. We started by teaching them System Thinking, System Analysis and System Dynamics in a traditional way through lectures and giving them sustainability assignments (Meadows et al., 1972; Forrester, 1971; Senge, 1991; Sterman, 2000; Schlyter et al., 2012; Sverdrup et al., 2014; Haraldsson & Sverdrup, 2004). In 2013, we wanted to focus the course more on the game students' interests, and we decided to let them make *a pervasive game* to teach elementary school students about sustainability and System Thinking. The project was carried out during the three last weeks of their System Thinking course.

Pervasive games are a group of games that are mainly played in the real world but that contain digital clues or digital games. Examples of pervasive games can be urban adventure games, alternate reality games, street-smart sports, or similar. Although there are not so many examples, we believe that pervasive games can also be good as learning games. The way in which they are built allows for group collaboration, learning just in time, and creative problem-solving.

It was decided to make three digital games as part of a pervasive game. The students were divided into four groups, where three groups each made a digital game, and the fourth group designed the pervasive game. During the project development period, the students met every day to sum up recent developments and to discuss progress, new challenges, and how to solve them.

We decided to help the students by drawing basic CLDs for the sustainability of the three digital mini games. The CLD gave them a broad overview of the game systems, and they could select a part of this larger system for the game to cover. They then made new CLDs that described their game in more detail. The CLD for the pervasive game had to be developed by the students from scratch. During this period, we also held smaller System Thinking sessions to help the students who were using System Thinking in the design process. The students made System Dynamics models in Stella, and the mini games were made in Unity. The design period also contained several play tests, where they could test the whole system. During the design period, the students also had to maintain project plans and time estimates, and every student had to blog about the progress. These served as notes for the final report they had to write after the project was developed and the play tests with the children had been performed. Although the project was short and ad hoc, we tried to make it problem-based, in that we defined the steps as cases that were solved by the group.

We chose two classes of fifth and seventh graders to test the game. The aim was to play the game during the school day, which is essentially 5.5 hours. To sum up the results and study the children's motivation and learning, we created a diary that the children filled in after they had finished the game. At the end of the day, we also conducted a System Thinking session with the children, to see how much of the System Thinking they had understood.

Research data was collected from all parts of the project. The data was mainly collected from teachers' notes and photos from the game development process, as well as the play tests with the children, the student research reports that were written after the test with the children had been performed, students' exams and course evaluations, the children's 17-page diary, the results from the children's System Thinking session, and background data about the children obtained from their teacher.

The games

The overall pervasive game was based on the following curriculum goal: "Explain how production and consumption can destroy ecosystems and pollute earth, water, and air, and discuss how this can be prevented and repaired" (UDIR, 2014). On this basis, the students created a story about a refugee who had to flee his country because of famine and war caused by climate change. The three digital mini games were built on the following themes:

- The relationship between human consumption and the ecosystem (The Island Game)
- The ecological principles in the soil seen from an earthworm's perspective (The Earthworm Game)
- Showing children how refugees struggle and fight to survive (The Refugee Game)

The game had a well-written narrative that was told through actors, web pages, videos, digital and analogue clues, and the mini games. As far as possible, every aspect of the game had to be analyzed and designed using System Thinking.

The Pervasive Game

The Pervasive Game starts by showing the children a video in which a refugee speaks to them about the situation in his country and how he had to flee his country to survive. The children must then find his blog, where they can read about the situation in his country. The blog also asks them to investigate what happened to his country. The children are then divided into groups, and each group gets a backpack with some clues, a walkie-talkie, a map, and a notebook. They must then solve the clues and go to different locations around the school area to solve the game.

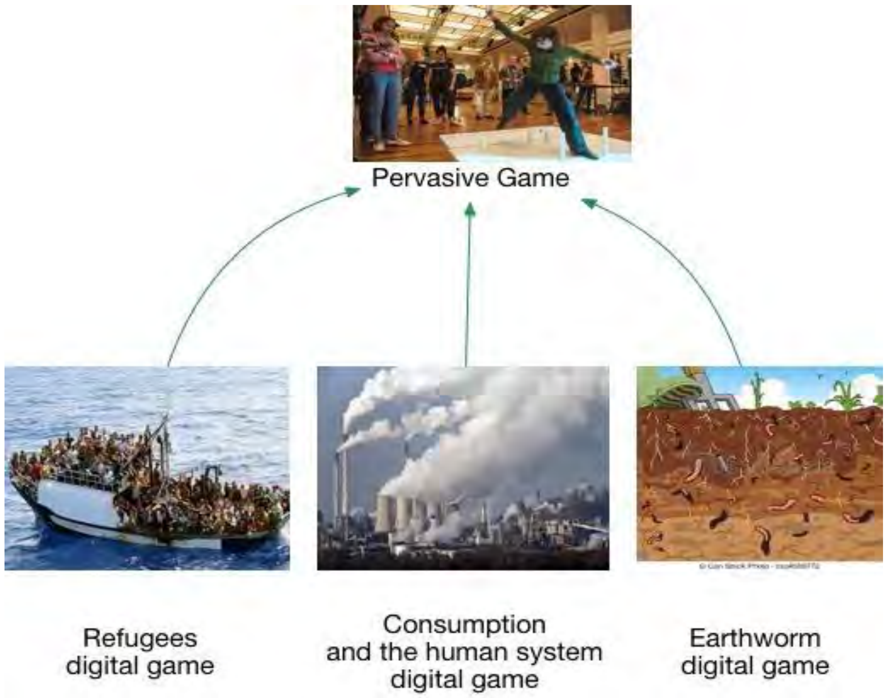


Figure 13. The Island/Earthworm/Refugee game must be solved to solve the overall Pervasive Game.

Some of the students have actor roles at different locations in the game, such as gardener, earthworm, locksmith, merchant, etc., and they all guide the children when they come to their locations. The game is implemented as a sandbox game, where the children can start out at any location and go through the locations in any sequence (if they get enough clues to proceed).

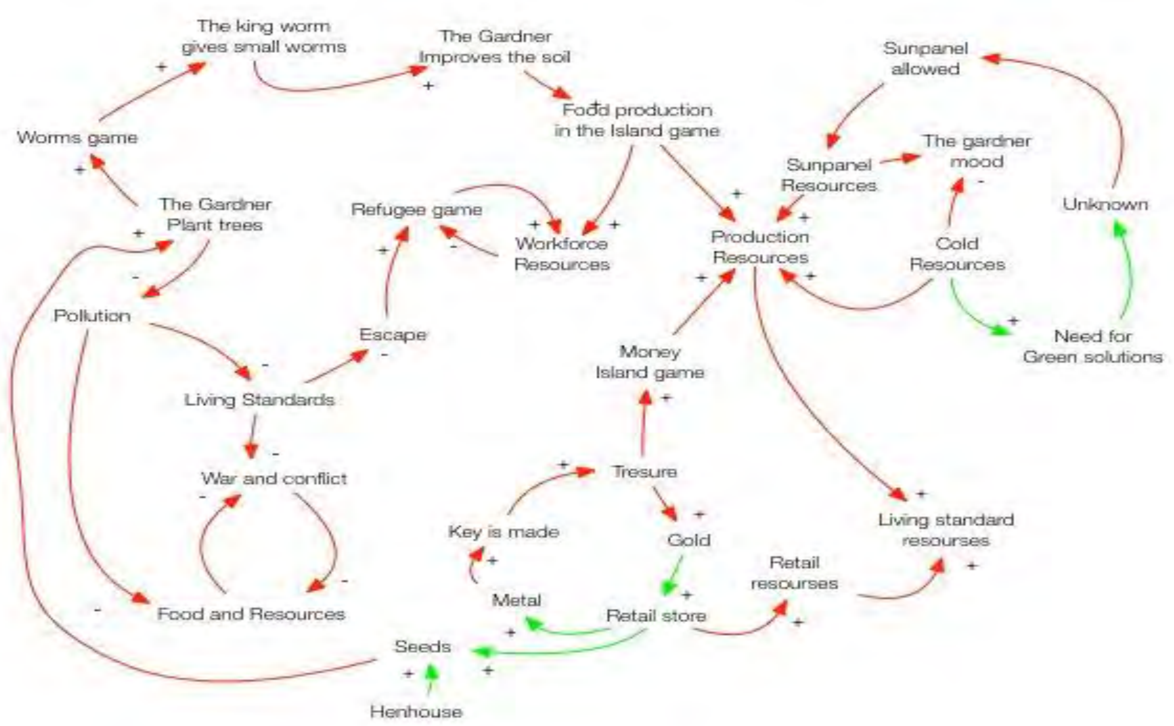


Figure 14. The pervasive game CLD as developed by the students. This shows the overall system's causal loop diagram, with all the small games inside the large game.

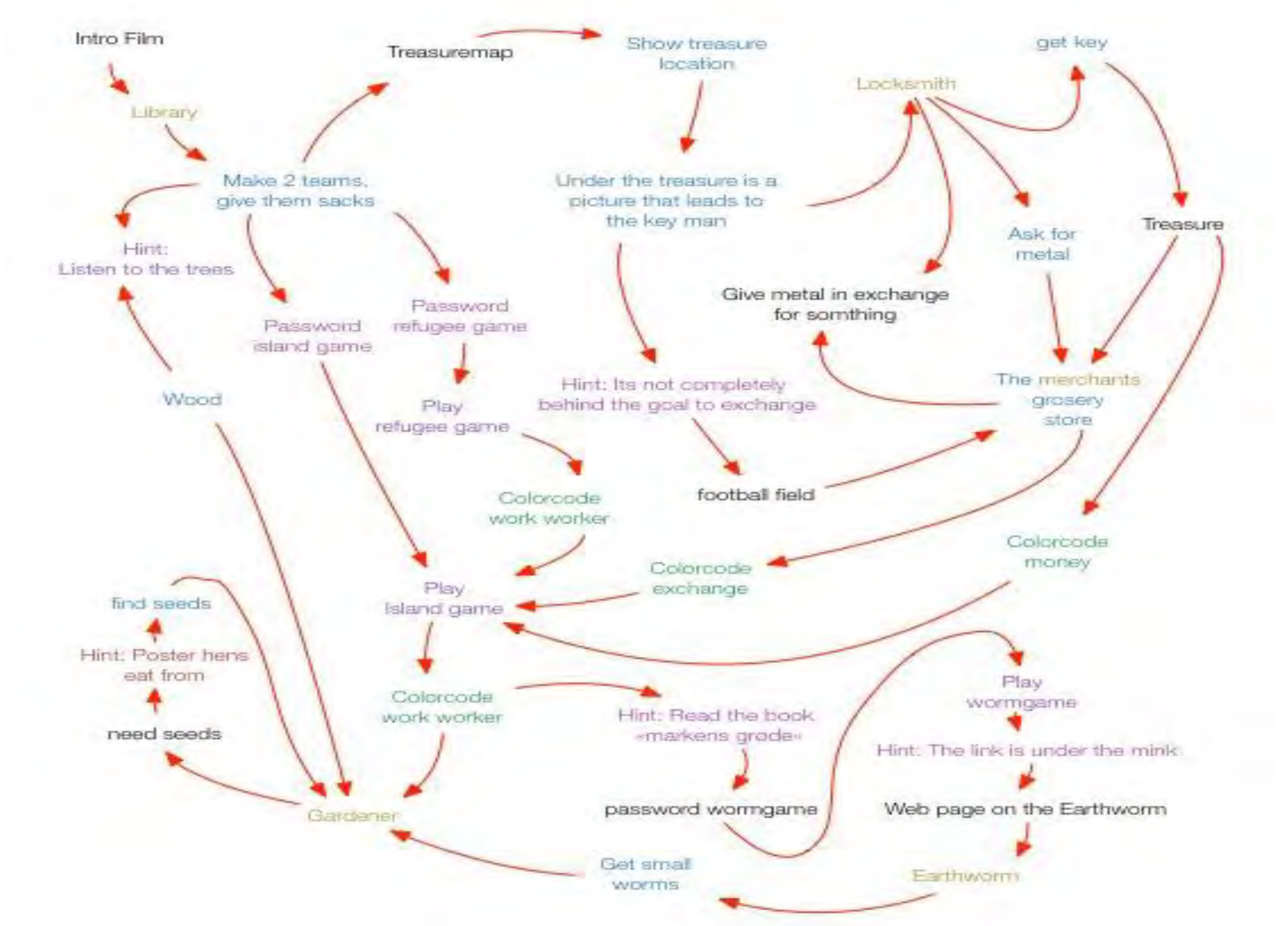


Figure 15. The game events map developed by the students.

The Island Game

In this game, the goal is to balance the resources of an island. The children can build houses to live in and factories to work in, locate various energy sources, fish, and cultivate plants. The key is to balance the resources, so that not too much CO_2 is generated, or too much waste produced. They can also let refugees come to the island and help develop it. The game encompasses many of the main aspects of sustainability, and shows them how they can do things in different ways to create more sustainable solutions.

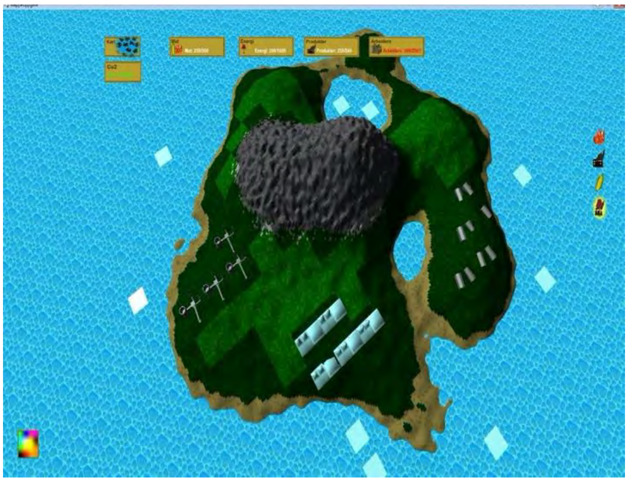


Figure 16. The Island Game

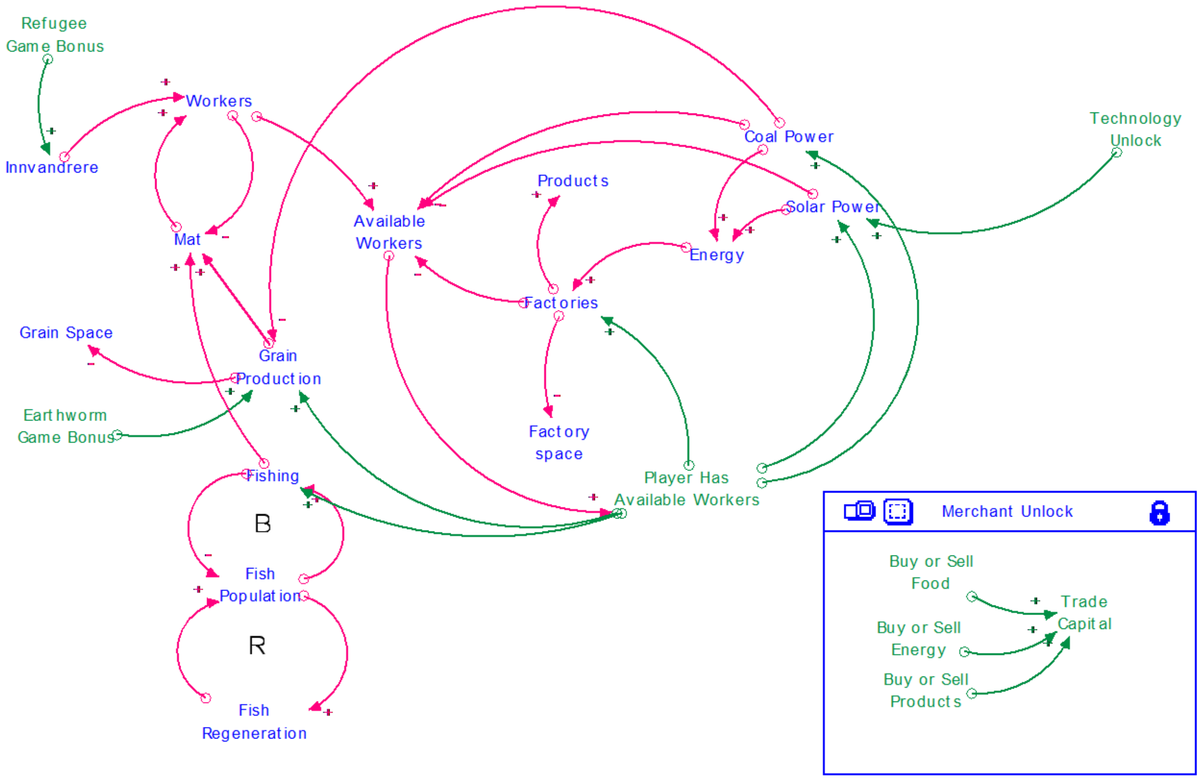


Figure 17. The Island Game CLD developed by the students.

The Earthworm Game

This game was made to show children how small details in nature can have a great impact on larger systems. The goal of the game is to optimize the harvest from a field of grain by moving earthworms to different locations and using as few pesticides as possible. The children learn how earthworms make the soil less compact, so that water and air can flow into the soil, thereby making things grow better and faster. The game also shows that the heavy machinery used in food production can make soil more compact, and that fertilizer kills earthworms. The game goes faster and faster and it becomes harder and harder to move the earthworms in time.

Figure 18. The Earthworm Game

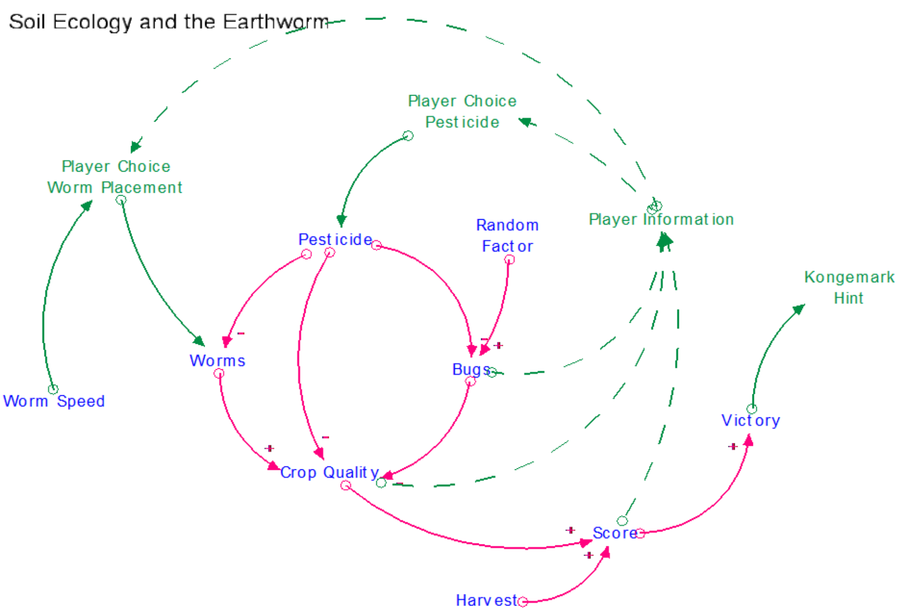


Figure 19. The Earthworm Game CLD developed by the students.

The Refugee Game

This game is more about feelings, and how it feels to be a refugee. The game lets the children be refugees who must flee their country, avoid the military and police, and collect enough food to survive along the way. If the refugee survives and arrives in a rich western country, they will apply for asylum. Their chance of being granted asylum is based on real numbers and statistics from that country.

Figure 20. The Refugee Game.

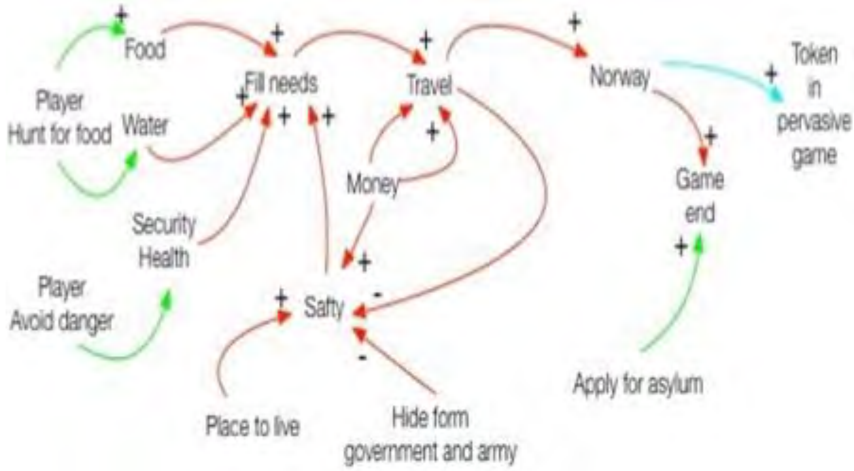


Figure 21. The Refugee Game CLD developed by the students.

The CLD shows how the refugee travels from their country to the country where they will apply for asylum. During the journey, they need to find food and water and avoid danger, in order to survive and seek asylum in the new country.

The aftermath

After the children had played the game, they sat down and worked on the 17-page diary for about an hour. The diary summarized the principles from the game in a way that related to the narratives of the game. It also contained a section with questions or assignments designed to give us an understanding of what and how much they had learned from playing the game. When the diary was completed, we gathered all the children in the library and conducted a

System Thinking session on the blackboard. One teacher asked the children questions, and another drew their answers on the blackboard in the form of a CLD. This resulted in a large CLD on the blackboard that surprised all of us, and perhaps especially the students that had studied System Thinking for weeks before they began this project.

Results from running the course

Both the student development class and the game day when the children played the game became vibrant communities of practice (Lave & Wenger, 1991; Wenger, 1998). In that respect, the gamification approach was successful in that it provided both the students and the children with a collaborative social environment in which they could practice according to their skills, and where the newcomers could learn from those who were more experienced. Both the environments and the community of practices successfully provided a “learning by doing” context. They shared a common goal: for the students to develop a game, and for the children to play and win a game. We believe that this was possible largely due to the PBL approach, where the theoretical learning appeared “just in time” (Gee, 2010).

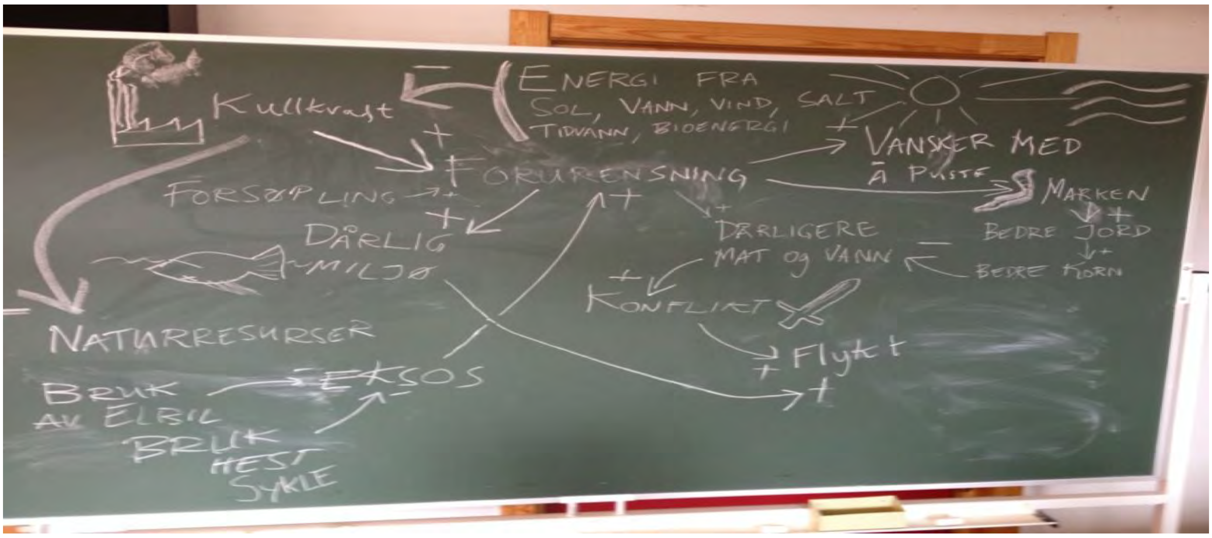


Figure 22. The blackboard CLD.

Some of the comments and impressions from the documentation collected during the project are summarized below. Note that no conclusive analysis was conducted, which was partly due to the long period of time since the project took place, and partly due to inadequate and lacking documentation.

The students

The students had fun while they designed the game and when they studied the children playing it. Here are some comments from their reports:

- “It has been great fun to participate in this project, with the best part being the play test with the fifth graders. I am satisfied with the whole project, and although it was hard to see how everything would turn out when we started the project, we managed to finish, and we finished with bravura.”
- “I see the pervasive game part of this course as an interesting experience where we learned a lot, and comments like ‘The best school day ever’, indicate that the fifth and seventh graders felt the same.”
- “This was a very fun and rewarding assignment to work with, and although we’re not actors and have not developed a game like this before, I felt it was a very successful project for us students, and I hope the teachers feel the same.”
- “To summarize, I’ll say this project has been a joy to work on. I’ve learned from it and gained many very nice memories. This has been nothing but fun!”

The students described the project as stealth learning, due to the short development cycle, but with a big reward at the end: the gameplaying by the children. From what they observed from the playing and the System Thinking session, it seemed that the children not only had fun but also learned something. This indicated that the gamification in the project had worked well. From the reports and the CLD made by the students, it seemed that they had learned at least as much as from previous classes about System Thinking, where they had used traditional teaching methods. However, the learning was not studied in detail, so we cannot say that the learning was better than from traditional teaching. We do, however, believe what the students said in their response – that learning by creating games is more fun and more thorough than traditional learning, due to the motivation that exists when “they learn by doing” and “just in time” (i.e., the need to learn pops up just when they need it (Gee, 2007)), so that making games was more fun. They also pointed out that watching the children was something to which they would tie their System Thinking learning and remember for a long time. Ideally, the students should make the CLD for their games themselves. This is something we will improve in future projects, by teaching them more about sustainability “just in time” during the development process for the pervasive game.

The time allocated to develop the game, which was only three weeks, was probably too short. The students wrote in their reports that more time would probably have resulted in better games.

Some students also pointed out in their reports that using CLD and simulations to check their game logic and debug the code proved to be very beneficial; they found the bugs much faster than with traditional tests. At that time (in 2013), we did not use System Thinking for this. Thus, they discovered another use of System Thinking in game development that was not pointed out to them.

While most of the students could see the value of using System Thinking in game development, some of them questioned the value of System Thinking in the small mini games. There was, however, a clear understanding that System Thinking would be a useful tool in the game design phase and for debugging.

The students also saw how the children completed the game much more quickly than they had anticipated. Fortunately, this turned out not to be a problem, since the diary had been developed in advance and the length of the System Thinking session was increased (note: the System Thinking session had not been well-designed in advance and was not meant be a big event). However, both turned out to be very interesting events, and the students wrote in their reports that they believed that these sessions added to the children's learning outcome. However, when done again, the game should be more difficult to solve. The students pointed out how impressed they were with the children's diligence and creativity, and the energy they put into solving the games, and the students believed that both the playing time and the difficulty could be increased without the children becoming bored.

Many of the students were also very impressed by how much faster than the students the fifth and seventh graders adopted the logic in System Thinking, and how quickly they saw the value of System Thinking in a sustainability context. Some students also pointed out that they believed System Thinking was a valuable tool for the children to learn.

The children

It was interesting to watch the children play the pervasive game. To us, it seemed that the pervasive game served as a creative playground where the children could use their creative skills and apply the lessons learned in practice in a creative way. Although digital games can often seem to be removed from reality, the pervasive game and the children's gameplay felt very real, as a playing experience for the children that was closely related to reality.

During the game, the children were taught sustainability and System Thinking by the actors, and after the game, we summed it all up in the diary and the System Thinking session. This seemed to be a good way of teaching; the children proved to be highly motivated by their experiences from the practical tasks they had solved during the game. This seemed to function as "stealth learning", as they learned quickly without noticing this, while running around playing the game.

From the teacher who knew the children, we learned that those who normally did not like school topics very much excelled at playing the game. They often progressed quickly, paid attention, and showed leadership skills. This shows that gameplaying in school can serve as adapted education, and can activate children who do not fit so easily into the traditional school system. We found that the game served as a great place in which to practice organizational skills. This applied particularly to the fifth graders, of whom a group leader rapidly emerged after they were given their backpack, and who divided the tasks between the members of the team.

Competition also seemed to be an important motivational factor for both the fifth and seventh graders. They soon established a leaderboard with scores on the blackboard and started to compete with the other group. Also, in the gameplaying, the groups were competing to finish the game first. The competition between the groups was a new discovery, as it had not originally been planned to be part of the game by the students.

The children also had great problem-solving skills. Although the walkie-talkies in the backpack were expected to be used to divide the group into smaller teams, so that they could solve the tasks more efficiently, they soon figured out how to "hack" into the other group's walkie-talkies and negotiate on information exchange with the other group. Some group members communicated in German so that the other group was not able to pick up any clues. The children also let the actors talk directly through the walkie-talkies and solve their problems.

We had the impression that the System Thinking session after the game was perceived as fun by the children. They participated very actively, and they soon identified components from playing the game and made connections between them. They were also able to quickly see the reinforcing or balancing factors. We had the impression that the children understood the System Thinking more quickly than the students had, and that it felt natural to them to think in terms of systems. Some of the students and teachers claimed that this session was the best part of the day. Some of the children also came to the library after the game and wanted to borrow books on sustainability to read more about how they could help to save the planet.

There were more differences than we expected between the fifth and seventh-grade pupils in terms of both their knowledge and how they played the game, even though they had the same competence aims. For example, while the fifth graders were highly enthusiastic throughout the game day, the seventh graders took some time to see the game idea as “cool” and fun. However, as soon as the children in the class who were seen as “cool” started to play the game, the others soon joined in (this was a comment from the children’s teacher, who knew them all well). The game design was probably more suitable for the fifth graders, who found the characters in the game fun. The seventh graders commented on them being a bit childish. However, the seventh graders seemed to like the digital games more and competed between themselves to solve the games as quickly as possible.

Article introduction

The article starts by discussing the many children who mod games these days, due to the many game engines made freely available to the public. During the modding process, the children learn about making art, animation, programming, game design, collaboration, and even pedagogy and psychology, when they strive to make their modded games fun. It then refers to Tapscott (2011) who talks about what these children want.

Learning designers have also seen this, among them Gee (2004, 2007, 2010) and Scaffer (2008), who argue that games can be powerful learning environments, and that these environments are about “learning by doing “(Dewey, 1916) and “learning just in time” (Gee, 2007). These learning environments can also be gamified, to bring back the motivation and fun (Detering et al., 2011). The article talks about the rapid advancements in technology in the past ten years, which allows children to “experience” and “solve real world problems” virtually. This takes place in advanced 3D simulations that are interactive, multimodal, and can also work via the Internet, and that enable children to share their experiences with other children from all over the world. This has never been possible before. By gamifying school topics in these environments, we can bring back the fun and motivation for learning, practice skills, perform automatic assessment, and create social environments where children can practice the theory they are about to learn in a fun way (Gee, 2010). There are many examples of such environments that are created as entertainment but “accidentally” also serve as learning environments, such as Minecraft, which allows children to create and build 3D environments, and Moviestar Planet, which teaches children about how to make movies, how to handle fame and fortune, and how to create identities. Also, virtual worlds such as Club Penguin, Habbo, or Stardoll can teach children about how to treat and take responsibility for their pets, or how to earn and save money. In all these environments where they meet, they also learn about collaboration, teamwork, and how to treat their peers. These environments have hundreds of millions of accounts (KZero, 2011) for children aged 5-15. They are also informal learning environments that could be integrated with "formal learning" in school classes.

Games are systems, and System Thinking goes hand in hand with game development. While System Thinking can be used to develop games, it can also show how systems are connected in the real world by causes and effects. System Thinking can also help children/students to understand advanced math without having to learn math the hard way. The article points to how System Thinking is part of K12 (the first 12 school years) in the USA, and that children who know this think more critically, express thoughts more clearly, and understand more complex problems (Lyneis & Stuntz, 2007).

The cross-disciplinary nature and importance of sustainability make it a good fit for "virtual environments", where teachers can create stories with which to teach children school topics. As such, this is a competence aim and goal in the curriculum, and it can naturally also lead developers and players to problems that force them to study subjects such as math and physics that are often considered "serious" and "boring" by children. Although developing digital games can be time-consuming and requires special skills, creating pervasive games is simpler, and developers can use real-world environments and common digital tools, the Internet, premade simple games, and virtual environments that already exist in schools. This project was developed by game students, but it could also have been developed directly by the teachers, as we shall see in examples from Article 5.

Scope and objectives

The scope and objective for this project is clearly two-fold: to teach the game students System Thinking by making games, and to teach children sustainability by gameplaying. The article focuses on the design part of the project and does not include any thorough data analysis of the project data. A working hypothesis for the project was that game development in combination with problem-based learning could be used to create a working environment that would enhance the student's motivation to learn System Thinking, but that would also make the learning proceed more efficiently than traditional learning. The course design was theoretically inspired by the sociocultural learning theory of situated learning (Lave & Wenger, 1991; Wenger, 1998, 2006).

The article dives into the theory behind problem-based learning and situated learning, and continues with an introduction to System Thinking and sustainability. It then explains the project design and the methodology, which aimed to bring together four major fields: System Thinking, sustainability, game development, and adaptive learning. The game design is described in the project description above. The research design is described as practitioner enquiry as stance (Cochran-Smith & Lytle, 2009). The System Dynamics theory diagram shows how data was collected from the children's debriefing, student reports and blogs, and the researchers' observations and notes.

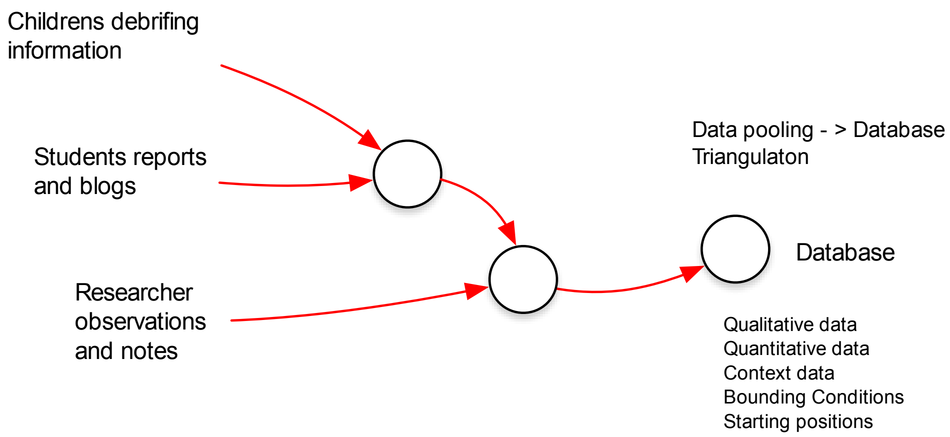


Figure 23. System Dynamics theory.

The article then considers the project description and discussion. In the discussion, the article is more concerned with how the students developed the game than how the children played the game. Most of the above comments on the students' game development of the project are taken from this article. The article states that 'without more thorough analyses of the research data in the project, it is hard to draw any conclusions. However, it also states that, from what the researchers have seen, they believe that the combination of System Analysis, sustainability, game development or gameplaying, and learning is a good one. It was quite clear from the reports and observations that both the schoolchildren and the students found the project to be fun, and the submitted reports, exams, and debriefing sessions all indicate that the learning outcomes for both were at least as good as for traditional methods. It is also believed, again without any evidence, that the learning was more thorough than would have been the case with traditional learning, due to the "learning by doing" aspect of the project.

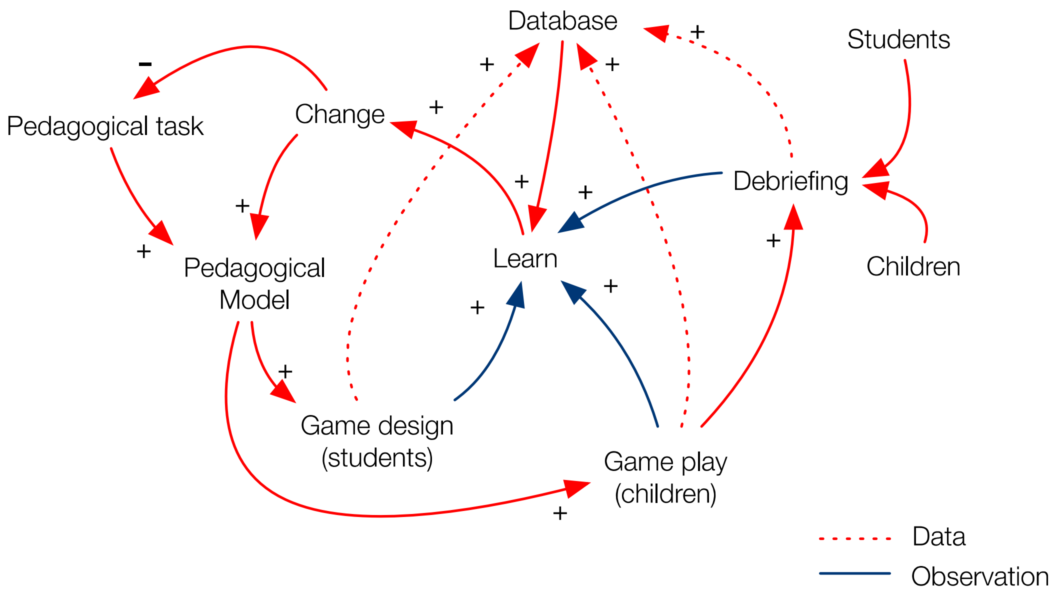


Figure 24. Researcher learning loop.

Another interesting fact was that the schools' interest in trying out this concept was exceptional. After sending out a one-page request to schools to implement the game just four weeks before we planned the actual test, we received answers from 15 schools, and more than 1,500 children wanted us to implement the game at their school. This indicates that many schools have seen the need for adaptive learning and a new teaching methodology.

Article 3: System Thinking in game design

System Thinking in game design (Nordby, Vibeto, & Sverdrup).

The article was mainly written in 2018 and was presented at the System Dynamics conference in Iceland in August 2018. It has since been significantly rewritten and will be published in 2023. *Text from these two articles is used in this chapter without further references.*

The project

This project mainly concerns game design. It is based on two courses in the Game School education program: Game Design in the second semester, and Games and System Thinking in the third semester. Although the Game Design course is not described at all in this PhD thesis (except for some perspectives on how to implement system identification and analysis in the discussion chapter), game design as a topic of the two other courses is described. Article 3 concerns how we introduce game design assignments in the Games and System Thinking course, which was originally a more traditional System Thinking course. As a result, the course has become more concerned with advanced game design, with a focus on using System Thinking in the game design process.

The Games and System Thinking course.

The Games and System Thinking course is basically a System Thinking course in which the students apply System Thinking and simulation to sustainability and game problems. The course has 15 assignments, of which the students undertake and deliver three to four as folder assignments. All assignment solutions are presented to all students in presentations and comment sessions in class. The assignments are problem-based, and the students work in groups with the teachers as tutors. In each assignment, the students solve a real-world task (concerning sustainability, etc.) or a game design task.

All real-world tasks are real-world problems that the students might encounter in their later working lives. Many of these tasks are cases that the students will gamify or simulate as a Serious Game. The game design tasks could be to balance challenges or to resolve more thorough game design challenges. Two examples of the latter tasks are given in this article.

The System Thinking approach in the course

The students first apply System Analysis to the problem by asking a question and identifying prerequisites, limitations, and borders, and then drawing flow charts and CLDs, and making

RBP or BBP. They then find numerical data for the problem, and create a Stella simulation of the problem where they simulate several scenarios. The final step is to write a report, in which they present the System Analysis and the simulation scenarios, and discuss what challenges they met and how they resolved them. The folder assignments have increasing complexity: the first has only one system, whilst the last has several systems that must be connected and interact. The compulsory assignments are intended to make the students work throughout the semester. These assignments mainly force the students to write blogs about their work, make compulsory presentations of their work in class, and submit parts of their work for comment by the teachers/tutors.

Introduction

The article starts by stating that games are systems, and points out how System Thinking is absent from the game design literature. This is misleading s most game design books, such as *Rules of Play* (Salen & Zimmerman, 2004), *Fundamentals of Game Design* (Adams, 2014) and *The Art of Game Design* (Schell, 2008), point out the systemic nature of games. Both metagaming, which involves out-of-game facts to influence the game, and the fact that modern games are increasingly becoming complex systems with different rule sets and modes that the players need to understand to make strategies to beat the games or human opponents (Tekinbas et al., 2014), mean that understanding System Analysis is also an important task for the players. Today it is possible to find videos, wikis, and forum posts that enable the systems in videogames to be broken down in order to master them (Wallach, 2019). So, both designing and playing games make it important to have a systemic understanding of the game – not only at a system level, but also at higher levels that require an understanding of how the systems are connected and interact. Modern online games are also constantly being updated, players need to work on them in teams, and the risk of events arising that the designers had not predicted increases. All this complexity makes it necessary for game designers to have well-proven tools in their game design toolbox. Many of these games showcase advanced systems that are interwoven. As sandbox and open world design become more widespread, the task of balancing sandbox and open world design to make good gameplay becomes more difficult (Doyle, 2015; Pramath, 2016).

Scope and objectives

The article shows two examples of how we, in our capacity as teachers from the Game School, use System Thinking to teach advanced game design. The article tries to shed some light on the following questions: Does System Thinking improve students' understanding of how games work, and can System Thinking in general improve game design?

Theory

The article presents System Thinking as a tool that can help game designers to design the game logics, but also to simulate game scenarios during the design process, as well as in the post-production phases where the game content is updated. Four phases are suggested in which System Thinking will be used. This is shown in Figure 25 (top diagram) below. The learning loop for the design is shown in Figure 25 (bottom diagram). It all starts with a vision of a game.

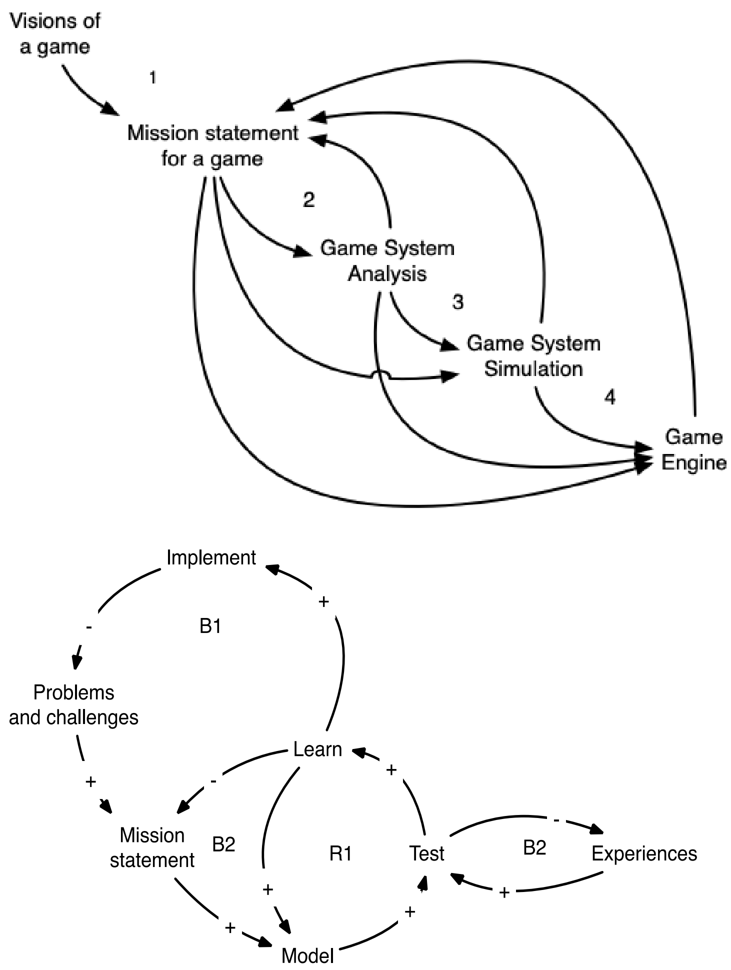


Figure 25. The overview of the game design process using System Thinking (top), and the learning loop (bottom).

1. Mission statement. Brainstorm and describe the game idea. The first step is free from any structuration tools.

2. System Analysis. Perform System Analysis of the game idea and design the qualitative model of the game. System Analysis uses tools described in the qualitative System Thinking methodology introduction, such as CLD, flow charts, and RBP.
3. System Dynamics Modeling. Simulate the model, test scenarios, and finally export the code to the game engine.
4. Game Engine. Do programming and art (and maybe art System Analysis) in the game engine. This is the game production phase and is not described in detail here.
5. The article also presents the four steps in the process in further detail in the diagram reproduced below. Here, both the four steps and the learning loop are shown together in the same diagram. Note that all steps have feedback loops, and all levels should be updated throughout the design process.

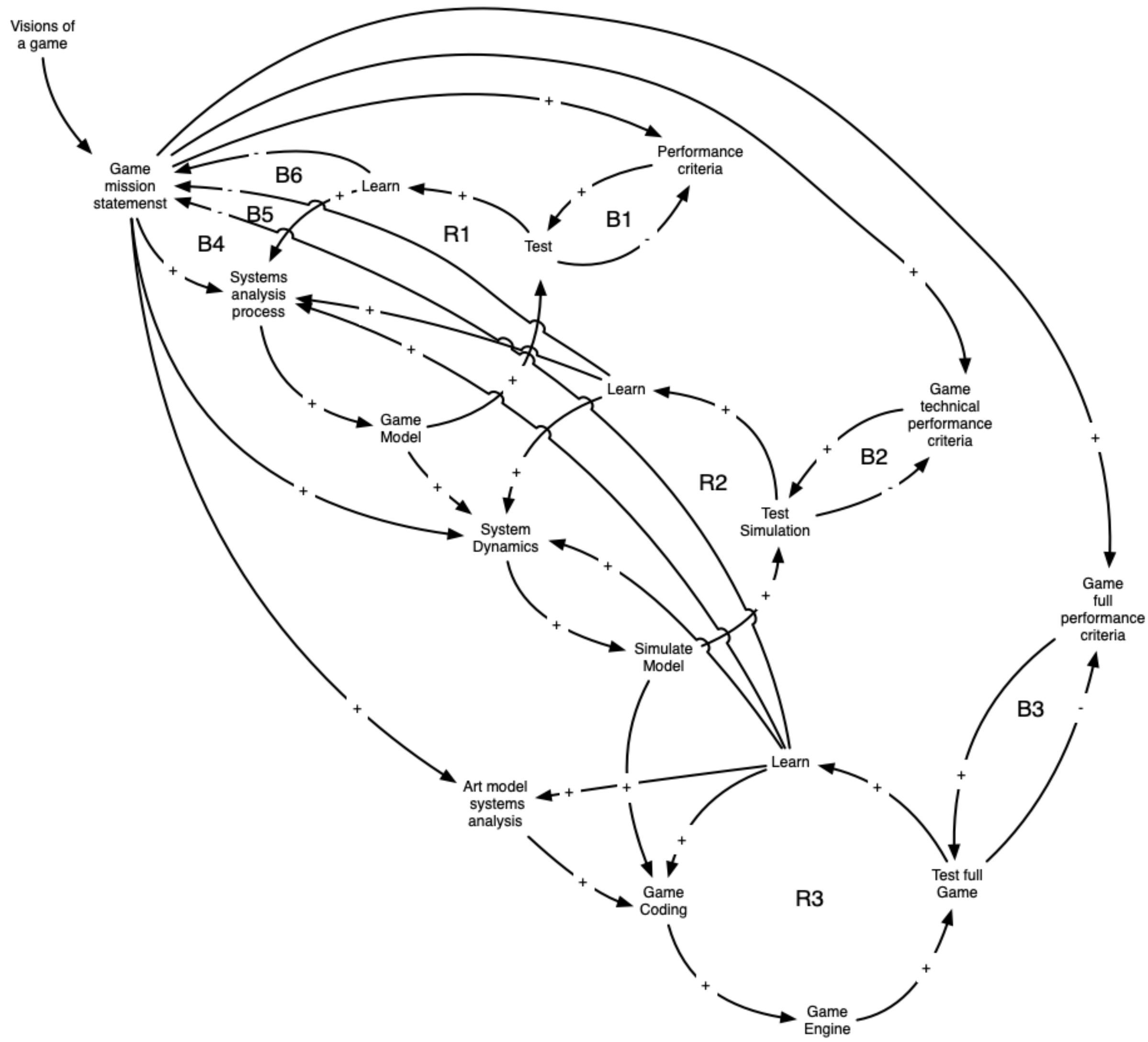


Figure 26. The game design process in more detail. Note that these CLDs are very similar to the CLDs presented in the theory chapter.

Case Studies

Below are two examples of the game-related assignments and how the students solved them.

Example 1

The first example – Tanks, Healer, Damage dealer – is an assignment that aims to make the students study how to balance the different elements in a boss fight. The example contains both the assignment text and a simple solution. This is a simple system, suitable for level 1 or 2 assignments. A dungeon party of 10 players (maximum number) is about to encounter the boss fight. Each of the classes has the following statistics:

Table 1. Tank, Healer, and Damage dealer health and aggro.

	Tank	Healer	Damage dealer
Damage rate	10	0	30
Healing rate	0	30	0
Health	30	10	20
Aggro rate	30	10	20
Damage reduction	25%	0%	10%

Table 2. Boss statistics in the game.

Damage	100
Health	1000
Healing Rate	0

Figure 27. Game balancing. The table was used first and then the causal loop diagram was derived from it.

Aggro is a term used by players to describe how non-player characters can be defined and implemented in many ways, and largely defines how the system is simulated. In this student solution, the Aggro is shown in the CLD as something that influences the Party Health but is not considered in the simulation. The boss has the statistics shown in Table 2. Figure 27 shows the Items, Actions, and Control table, and the CLD. The CLD is very simple and shows how the attacks from the Tanks and Damage dealers reduce the Boss's health, and how the Boss attacks reduce the Party Health. The Healers, Aggro, and the Boss attacks have an effect on how the Party Health develops over time.

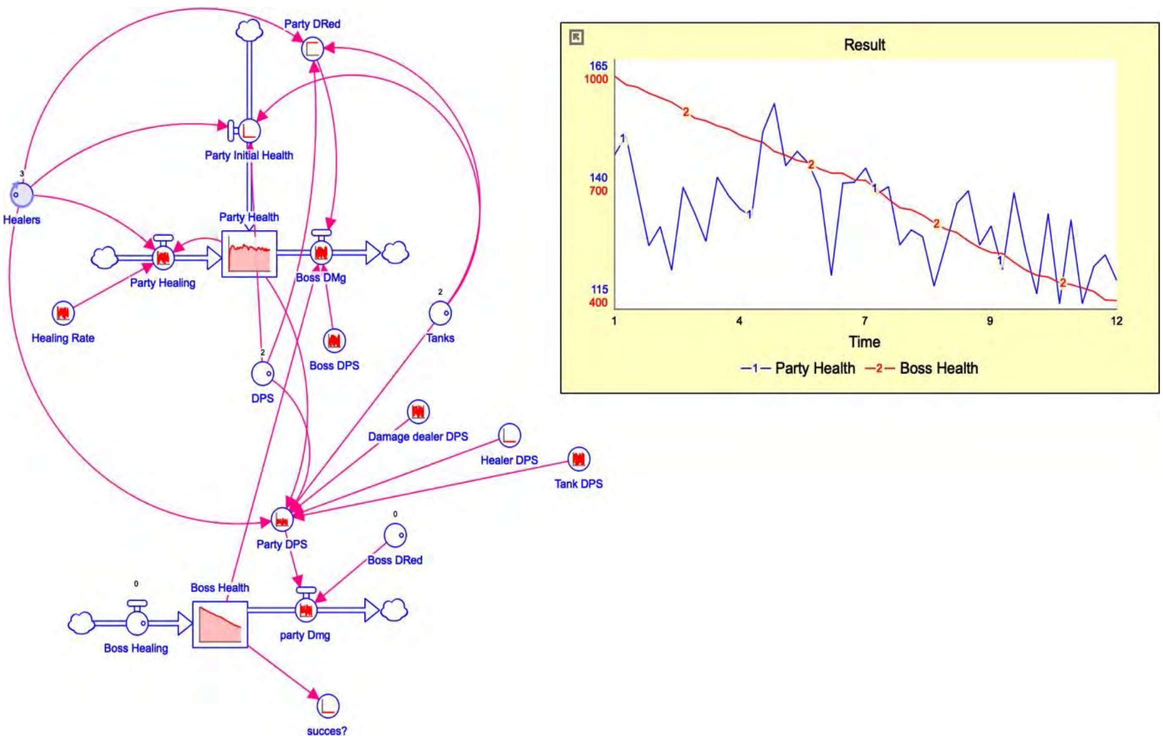


Figure 28. The simulation stella code example and an output.

The simulation has two stocks that store the Party Health and the Boss Health. They both have inflows that represent the healing rate that takes place over time (the Boss healing rate is 0), and outflows that represent the damage per time step. The Party damage done to the Boss is given by the number of players in each class, times the damage rate, minus the damage reduction. The Success is a Boolean that detects when or whether the Boss Health has reached 0, and the Boss is beaten.

Example 2

The second example – Trade Route Logistics – is a larger example consisting of two to three systems that must be connected and interact, suitable for a level 4 assignment. The assignment is about maximizing profit from trading cotton. Cottonbog, the trader's city where the cotton is produced, sells cotton to three nearby cities – Slagtown, Coalpeak, and Goldford. While the road to Slagtown is short and safe, the roads to the two other towns are longer and raided by bandits. The students need to create two systems, one for Cottonbog and one for the bandits, and connect them. The CLD also shows the cotton production system that is connected to the two other systems. There are 35 traders who travel to the cities and sell the cotton.

Table 3. Trade Route Logistics statistics

Town	Travel time (weeks)	Raid chance	Toll price per trip	Trader income per week	Trader income per trip	Gold earned per kg cotton
Slagtown	2	low	500	1000	1500	109.5
Coalpeak	8	medium	0	1000	8000	584
Goldford	4	high	0	1500	6000	438

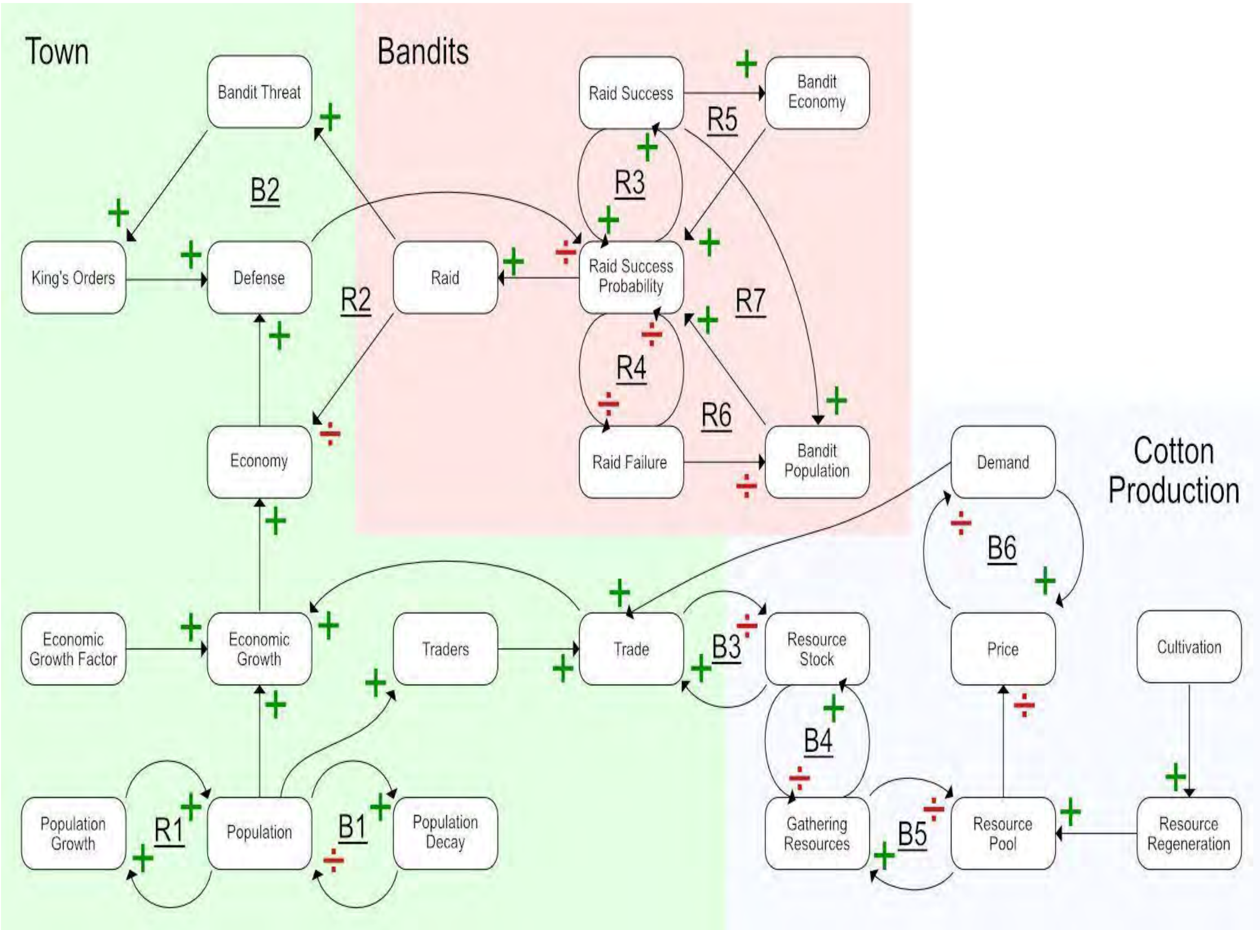


Figure 29. Trade Route Logistics CLD

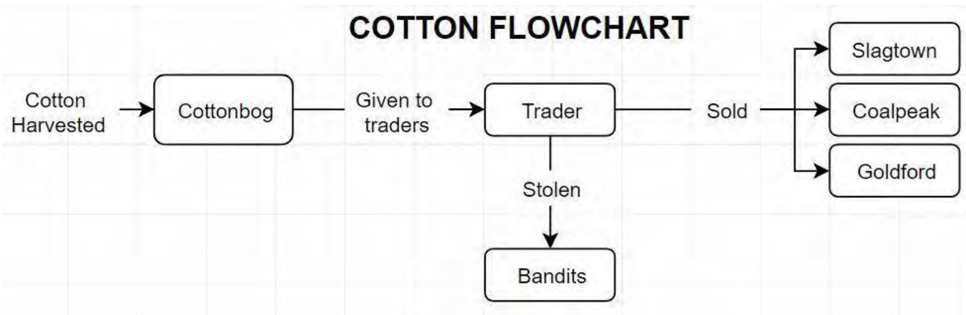


Figure 30. Cotton Flowchart

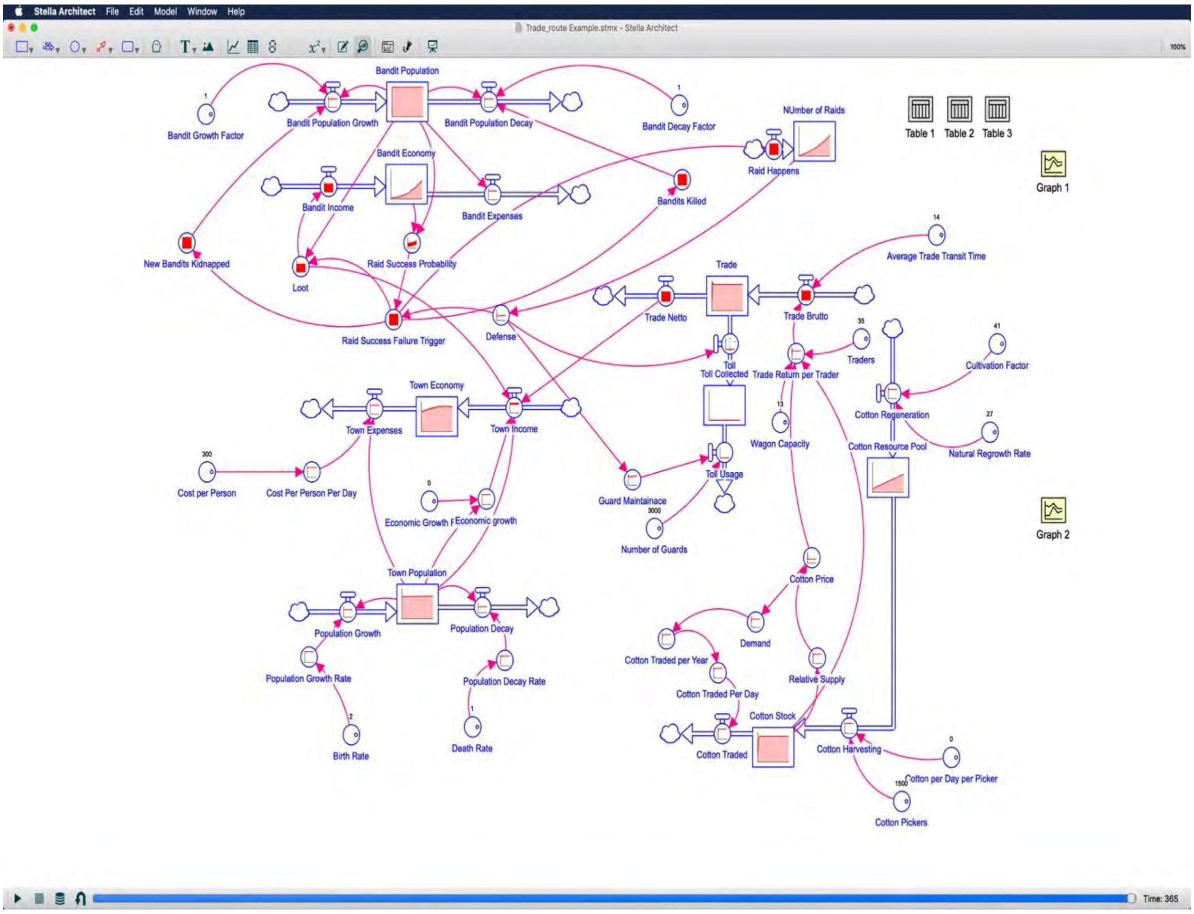


Figure 31. The STELLA diagram for a part of the game.

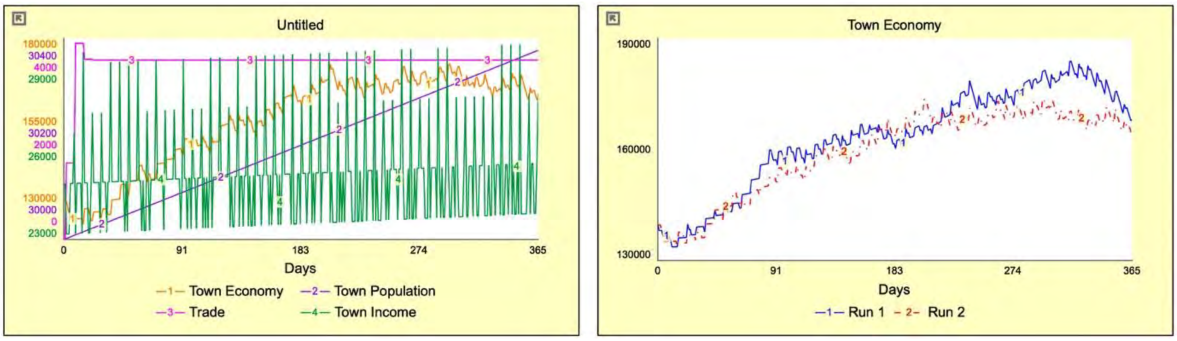


Figure 32. The flow charts from the System Analysis show what flows through the system – namely, cotton and gold. The first diagrams above show the town economy, the population, the trades, and the town economy. The second shows the model outputs from the same system economy, but from two different runs.

The CLD and the simulation have two population systems – one for Cottonbog and one for the bandits. These two population systems are connected to two different economic systems. Cottonbog's economic system is dependent on trading and raid success rate, whilst the bandits' system is only dependent on the raids. Both systems are naturally also dependent on how much cotton is produced. The raid success probability is random between 0 and 1 and corrected by modifiers based on the bandit population and economy. Raid success is also moderated by the defense against raids variable. With every raid, the defense is increased a little.

The article evaluates the students' performance and results in the following ways. Having to do System Analysis contributes to the students' general understanding of both System Thinking and game design. Simulations using System Dynamics Modeling force students to reflect on how the values in each task influence gameplay and even design. This also helps students to balance the game and provides mathematical equations they can use in the game engine. And the value of System Thinking of course increases with the complexity of the systems. In the end, the students also have a complete set of tools with which to explain their game to others. So, does System Thinking improve the understanding of game design? In general, we think it does.

Article 4: System Thinking in gamification

The gamification course is in the fourth semester, by which time the students already understand Game Design and System Thinking. They also know how to make games and have had projects in which they have done that.

Although game design and gamification design are not the same, gamification design largely builds on game design, as it uses game elements to make the gamified task more fun. Article 4 gives examples of assignments undertaken by the students in the gamification course. This course uses game design theory and System Thinking from both game design courses in the previous semesters and add its own flavor, as gamifications do not usually end up as games – the game design is just a tool to make the gamification task more fun. The article was mainly written in 2018 and was presented at the System Dynamics conference in Iceland in August 2018. They have since been significantly rewritten, and will be published in 2023. *Text from these two articles is used in this chapter without further references.*

The project

The students work in groups, with 3-5 students in each group. The gamification tasks have varied over the years. In 2017, the year of the first example, we gave the students the choice of selecting one of three tasks – The Galapagos dilemma, The blessing and pain of oil, and The water situation in the River of Jordan. Since 2018, however, we have allowed the students to define their own task freely within three areas: Learning – for example, in school, in hobbies or just training skills etc.; Health – for example, athletic training, eating healthier, exercising to become leaner, etc.; and Sustainability – for example, using less plastic, throwing away less food, using less oil-based fuels, etc. The second student example in the article is taken from this period. The course has three parts:

1. System Thinking. The students start by selecting the area for the gamification and brainstorm a gamification idea. Then they analyze the gamification idea using System Thinking. This is first done through System Analysis, where the students describe what flows through the system and then draw a CLD of the logics in the system. After the System Analysis is performed, they undertake System Dynamics simulation with a few scenarios to analyze and predict how the gamification will perform.
2. Gamification Design. In this part, the students perform the gamification design, and document and implement it in the prototype. In this part, the students work together with

the design and animation students to make a good design for their prototype. This collaboration also forces the gamification students to really think through all screens in their app, to be able to describe it and design them with the design/animation students. An important part of this phase is, of course, to design the gamification parts, which are methods to motivate, goals to expect, reward systems, and gamification elements, etc.

3. Report. Write a report on the process and the results of all parts of the gamification process. In the report, they must argue why their gamification will work, based on theory, since they cannot test it out in the real world (it's only a prototype).

Figure 34 shows the gamification design and development process learning loop. In the course, the development process stops after the prototype is developed. As we can see from the drawing, phases 1 and 2 of the student assignment described above are highly interactive.

Student example 1: Galapaseq

Galapaseq is a phone app for inhabitants and tourists in the Galapagos Islands. The Galapagos economy is largely based on tourists who come to the islands to see the wildlife, but due to increased demand, welfare, and increasing populations on the islands, fishing has grown so much that it now threatens some species to the point of endangerment. The idea of the app is to help the government to protect the environment by stopping illegal fishing, which they believe is a large element of overfishing. Although the government tries to catch illegal fishing boats, surveillance is expensive, and they need help to do so. The app allows users to take photos of boats they believe are fishing illegally and submit the photos to a database. The government goes through the photos and if an image helps them to catch an illegal fishing boat, they honor the app user with various rewards, such as discounts in shops, price reductions at hotels, etc. The main app screens and the app flow are shown in Figure 35.

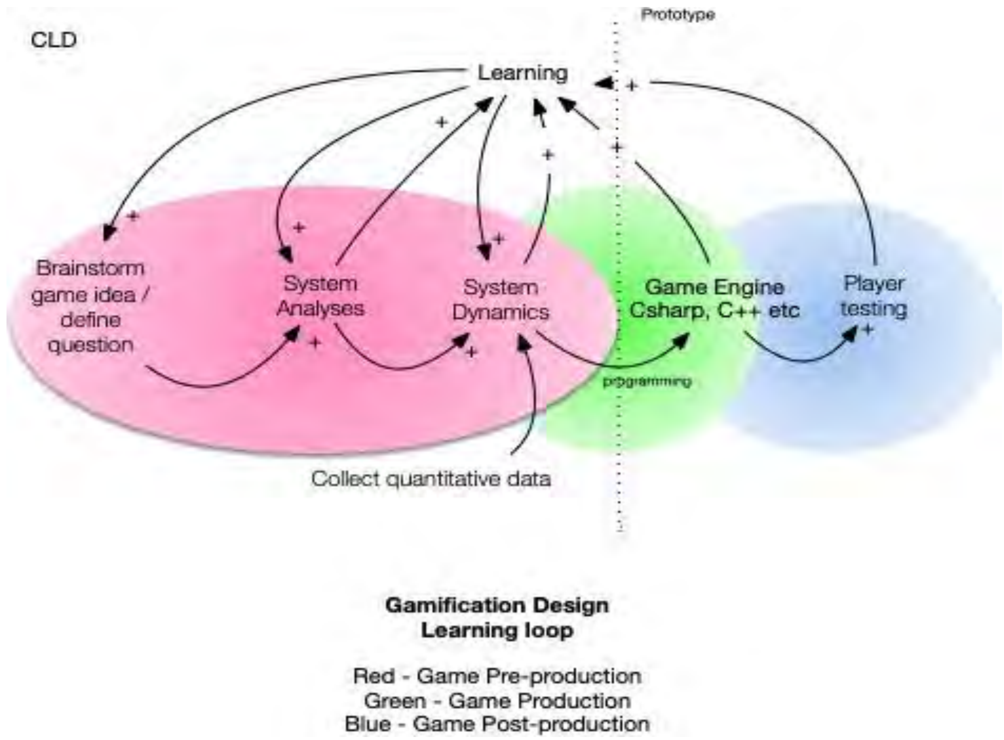


Figure 34. The learning loop as it was applied.

The "flow chart" or "course of action chart"

The most valuable asset of the Galapagos Islands is their unique nature, and most people living there are affected because this nature is the islands' most valuable source of income, in the form of tourism. The second largest source of income for the islands is fishing. The "flow chart" or "course of action chart" in Figure 36 below shows this balance. It is divided into two parts, and the students described it in the following way: 'This flow chart represents how we see the course of actions and effects that happens in the Galapagos. On the one hand, we have the protection and non-harmful exploitation of the Galapagos' ecosystem, and on the other, we have traditional harvest-style exploitation. Endangered species lead to pressure to protect them, which leads to protection laws (often meaning smaller fishing quotas), which leads to the conservation and nurturing of the biodiversity that attracts much of their tourism (Schep et al. 2014, 27-28). On the other hand, we have pressure from the fishermen for larger quotas and thereby more income from legal fishing. These two generally work against each other.'

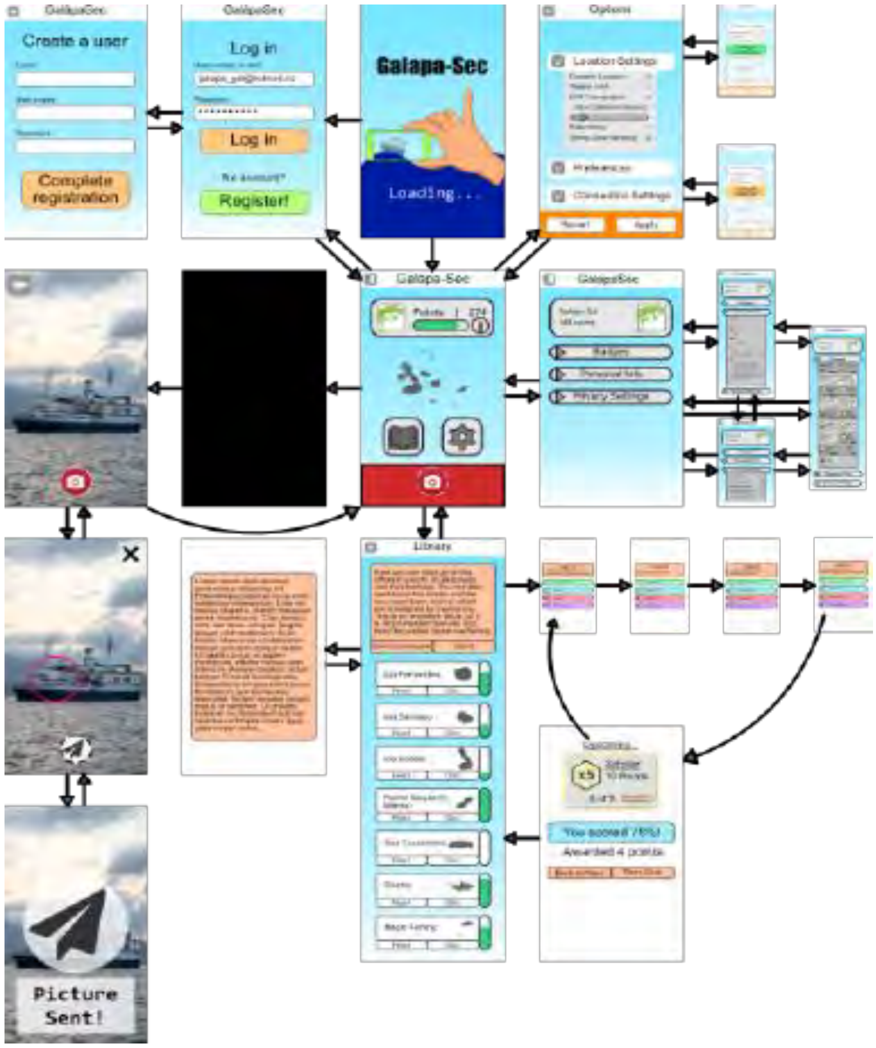


Figure 35. The main app screens and flow.

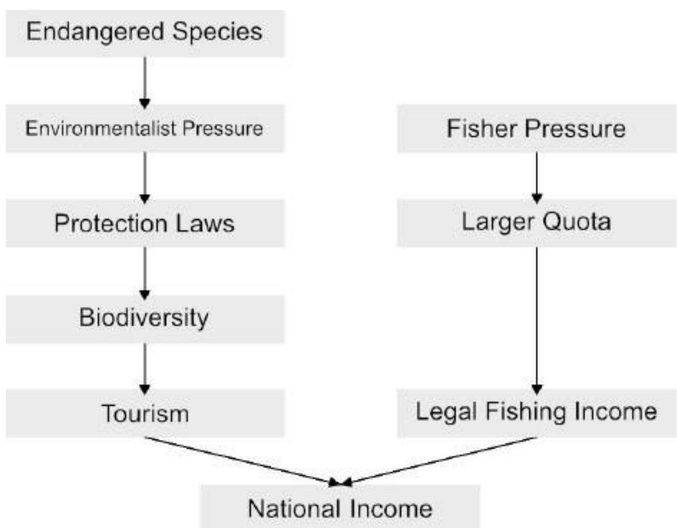


Figure 36. The "flow chart" or "course of action chart". This is not a proper flow chart but a map of events. The students frequently confuse the two.

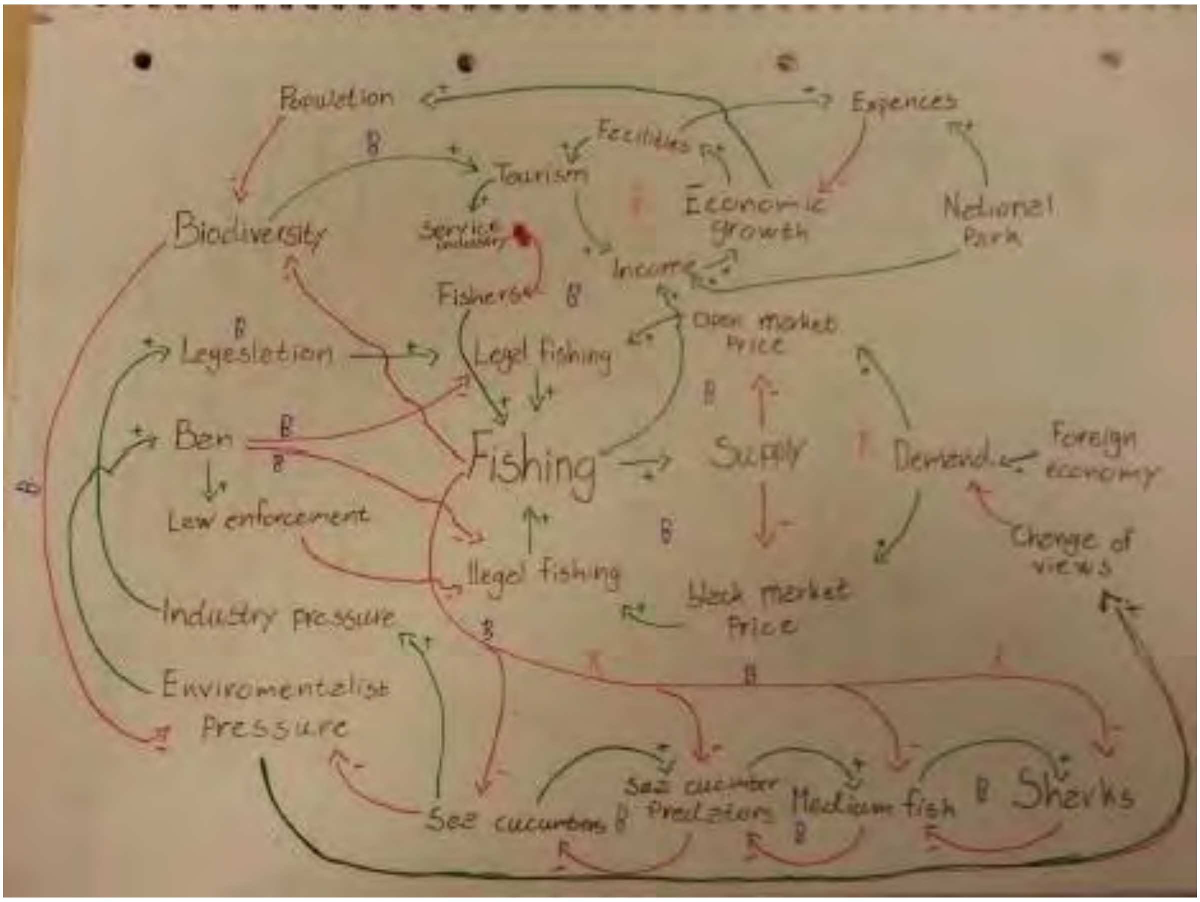


Figure 37. The students causal loop diagram for the game

The causal loop diagram (CLD)

The students' causal loop diagram (CLD) shows how fishing, the tourist industry, and the demand for fish affect biodiversity and economic growth. The green arrows represent growth, while the red arrows represent decline. The students have divided the fishing into four types of fish (from sea cucumber to shark), where sea cucumber is the species closest to extinction, which leads to pressure from environmental groups to stop fishing. Tourism leads to economic growth, but also to a larger population and more pressure on nature and biodiversity. Greater demand for fish leads to more fishing and illegal fishing, while environmentalists' pressure can endanger fishing's reputation and lead to less demand for the fish.

The System Dynamics Modeling

The students first tried to model the CLD with a larger system dynamics Stella model that included all parts of the CLD, but it proved hard to find numbers that could support the model. In the end, they made a more simplified simulation, shown below in Figure 37. This simplified version mainly concerns the Galapagos economy. In the model, the industry and activist pressures are merged, because the concern is only the strongest influence it has on legislation at any time. The motivation for the industry is the foreign demand, whilst the motivation for the activists is biodiversity. Legislation influences fishing and tourism, and the economy. With this model, the students then simulated four scenarios: the situation continues as usual; fishing is banned totally; the foreign demand decreases; and tourism declines. Even with this simplified simulation, the scenarios show how the economy of the Galapagos will react to changes in ecology. The scenarios are shown in Figure 38. In the examples, System Thinking serves as a scientific tool to understand and develop game design, and the System Analysis gives the students an overview of the systems and connections involved. System Dynamics Modeling allows the students to tweak the design before it is implemented.

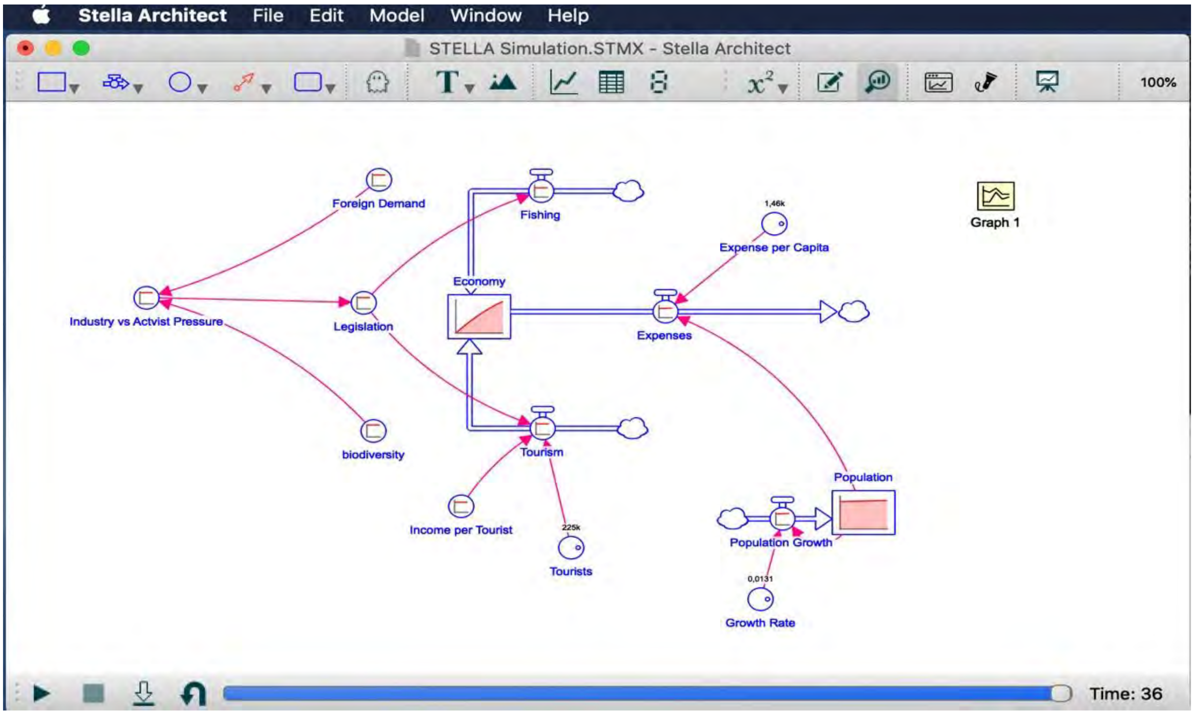


Figure 37. The Stella simulation.

The students claimed that working on the models and simulations provided them with an insight into the dynamics, and thereby a better understanding of the situation. Furthermore, the students also pointed out that, although the search for data and the material found did not give enough data to run a fully-fledged model, it did contain descriptions of the dynamics and cases that helped them to understand how it all worked together. In example 2, there are more systems, and the System Thinking can help giving a “fish tank” view to get a better overview of the game design.

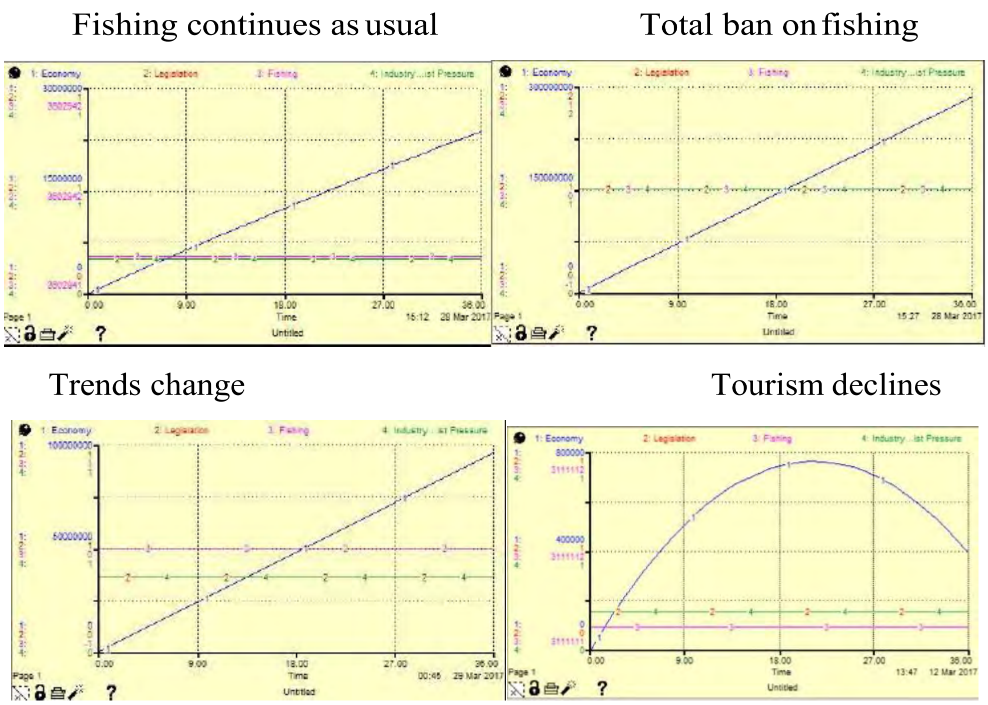


Figure 38. Scenarios investigated by the students.

Student example 2: Family manager. Examples of STELLA Architect outputs used by the students.

Student example 1: The family manager app

The family manager aims to teach families to create structure and implement positive habits in their lives. The app aims to make chores and regular boring tasks easy to administrate, and to create positive habits for the whole family over time. Examples of the tasks could include tidying up rooms, doing homework, walking the dog, cleaning the kitchen, etc. Defining the assignments is undertaken by the whole family together, and all family members are involved in performing them. Completion of the tasks is honored with small rewards, such as an ice cream or allowance, or as part of bigger rewards such as a family trip, etc. The students compared the app to a role play game, where the RPG requests were replaced with tasks, but they were also aware that gamification is not a game. The app can also help children to learn about finances, although this is not the core purpose of the app. Some of the screens from the app are shown in Figure 39.

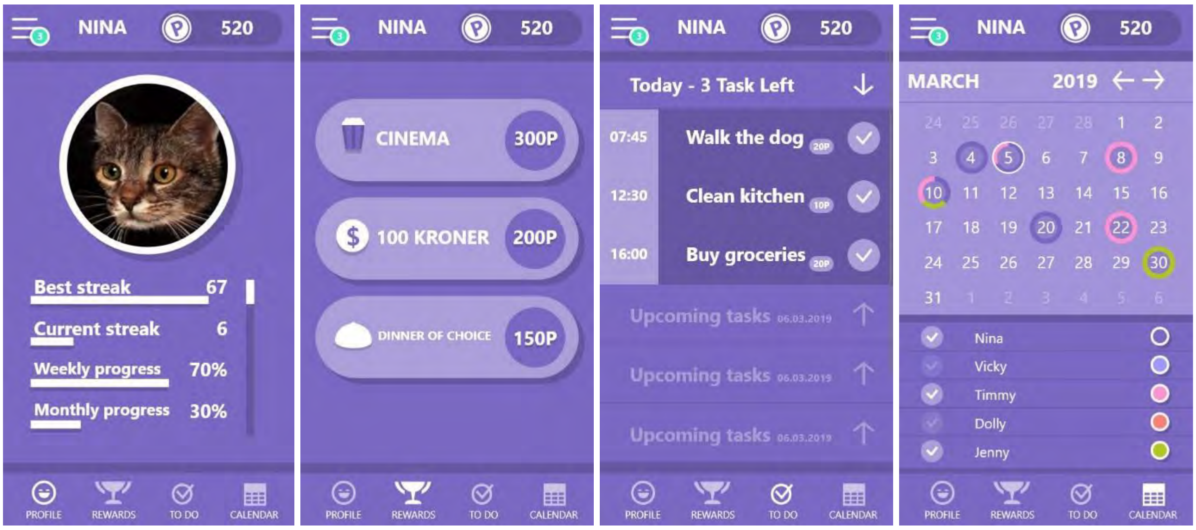


Figure 39. App screens as they appear.

The students' System Thinking part focuses on how to create the motivation, and builds on psychological models such as the ARCS model and the self-determination theory (SDT) (Kapp, 2012, p. 54), and occasionally Malone's theory of intrinsically motivating instruction (Malone, 1981, p. 27) and Lepper's instructional design for intrinsic motivation (Kapp, 2012, p. 57).

In the CLD, almost all loops visit motivation directly, and there are more reinforcing loops than balancing ones. The students see this as a good sign, as it points to increased motivation over time. Also, the CLD shows that most of the motivation goes into Intrinsic Motivation, which means inner and longer lasting motivation, which students believe will engage the users with the app. The students point out the importance of the user's sense of mastery and accomplishment, as well as of noticing the results in their environment. For the younger children, they also see learning as an important factor. The CLD also shows that Extrinsic Motivation is not forgotten and is given in the form of points (R2) and rewards (R1). The students see this as necessary to make the younger users want to use the app. However, they also point out that the goal is to increase the inner motivation over time. One way of doing this is to make sure the users are given more tasks than they can handle. The students also made flow charts as part of their System Analysis. These show how mastery, accomplishment, completing tasks, and learning add to intrinsic motivation, and thereby increase the gamification value. The students undertook a thorough System Dynamics Modeling, based on the CLD, which is arrayed so that multiple users can be simulated in parallel. In the model, the Task system is to create tasks and send them to the users. To complete these, the users need motivation, and if they are completed, the user gets rewards and points. However, completing the task also gives the user “psychological feelings” that lead to intrinsic motivation.

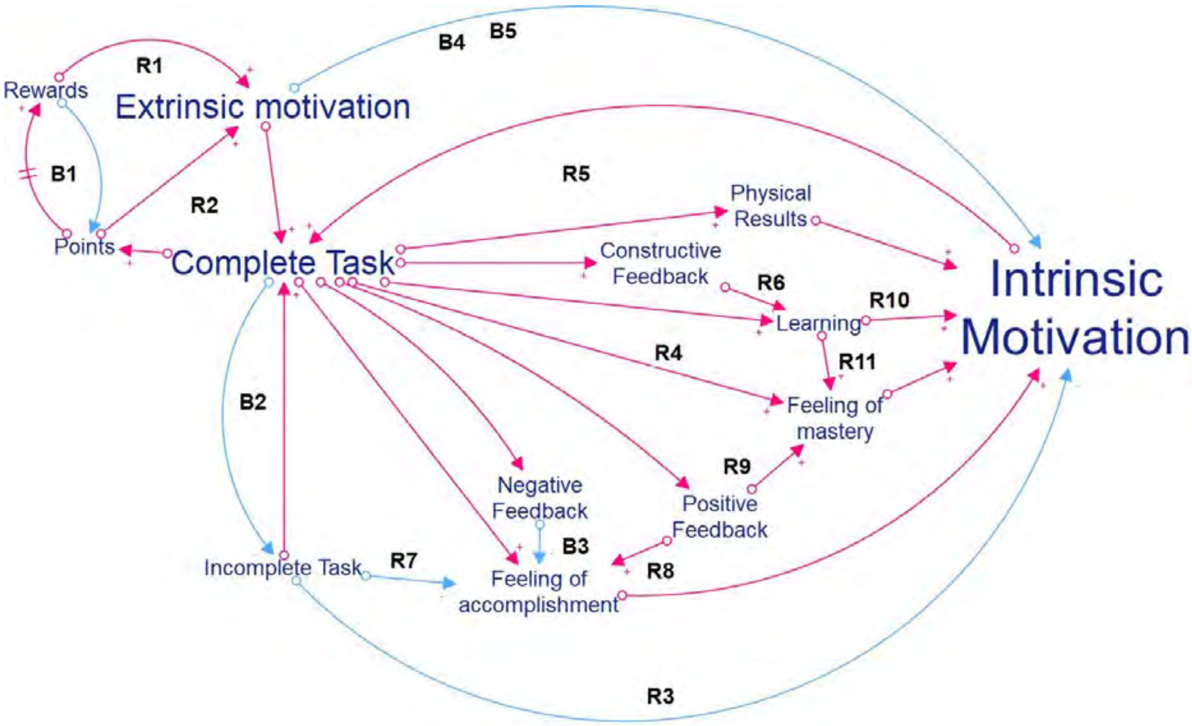


Figure 40. Motivation CLD

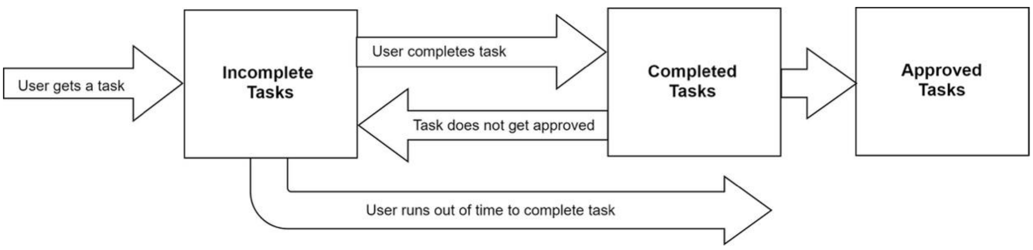


Figure 41. Tasks. A task enters the system when it is assigned to a user. The user either completes it or runs out of time to do so. If the time does not run out, it is put in the queue to be approved.

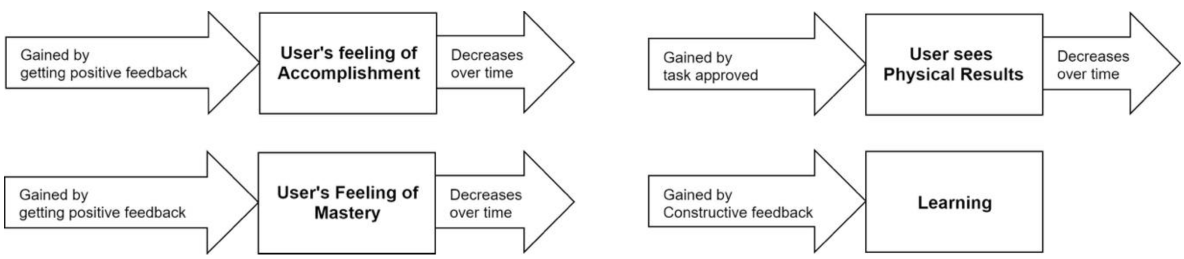


Figure 42. Users receive motivation from feeling a sense of accomplishment, from mastery, from learning, and from seeing results. The students assume that any motivation decreases over time, whilst learning does not decrease at all.

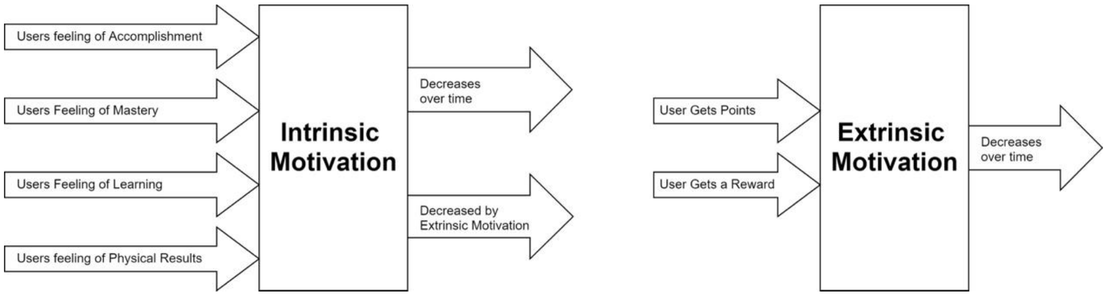


Figure 43. The final flow chart shows how mastery, accomplishment, learning, and physical results are likely to affect intrinsic motivation, whilst points and rewards influence extrinsic motivation.

A formula is used to calculate whether the task will be approved or not, based on expectations that are set according to the number of tasks approved and the skill of the user. The students ran the simulation for a year with the whole family. The Stella Architect model is shown in Figure 44. For the simulation, the students created a fictional family with five members. The initial values for the family are listed in Table 4.

Initial values	Female (8)	Male (10)	Female (12)	Mom	Dad
Skill	2.1	4.19	5.06	8.92	8.53
Intrinsic Motivation	10	20	2	50	50
Extrinsic Motivation	50	30	10	10	10
Base Amount of Tasks	2	2	3	5	5
Base Constructive Feedback Percentage	70%	70%	70%	0%	0%

Table 4. The initial values for the five family members.

The scenario results are shown in Figure 45. The students point out that the most important result is how intrinsic motivation increases constantly for all members of the family. They see this as good, and as a goal they hoped to achieve. Extrinsic motivation will also balance out for all family members, which they interpret to mean that the family will eventually be driven more by intrinsic than extrinsic motivation, and that all members of the family will find lasting motivation to maintain a structured life. Below, we show the results from running the simulation as described.

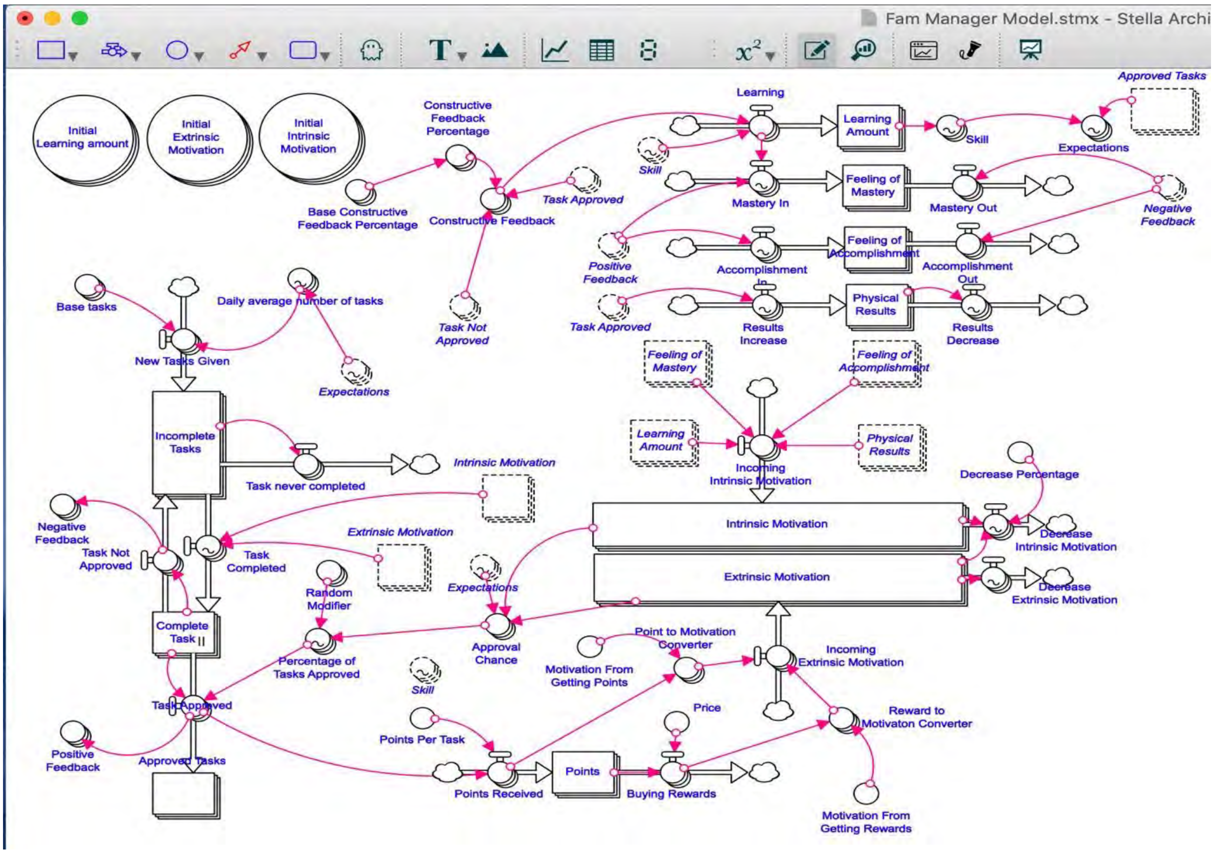


Figure 44. The Stella Architect Model.

The students also ran the simulations two more times, with the constructive feedback percentage set to 0 percent and 100 percent, to show the importance of constructive feedback. This shows that the children have noticeably less intrinsic motivation when they receive little constructive feedback. Just as importantly, this simulation also showed that their skills barely increased, and the users saw a visible drop in task approval chance.

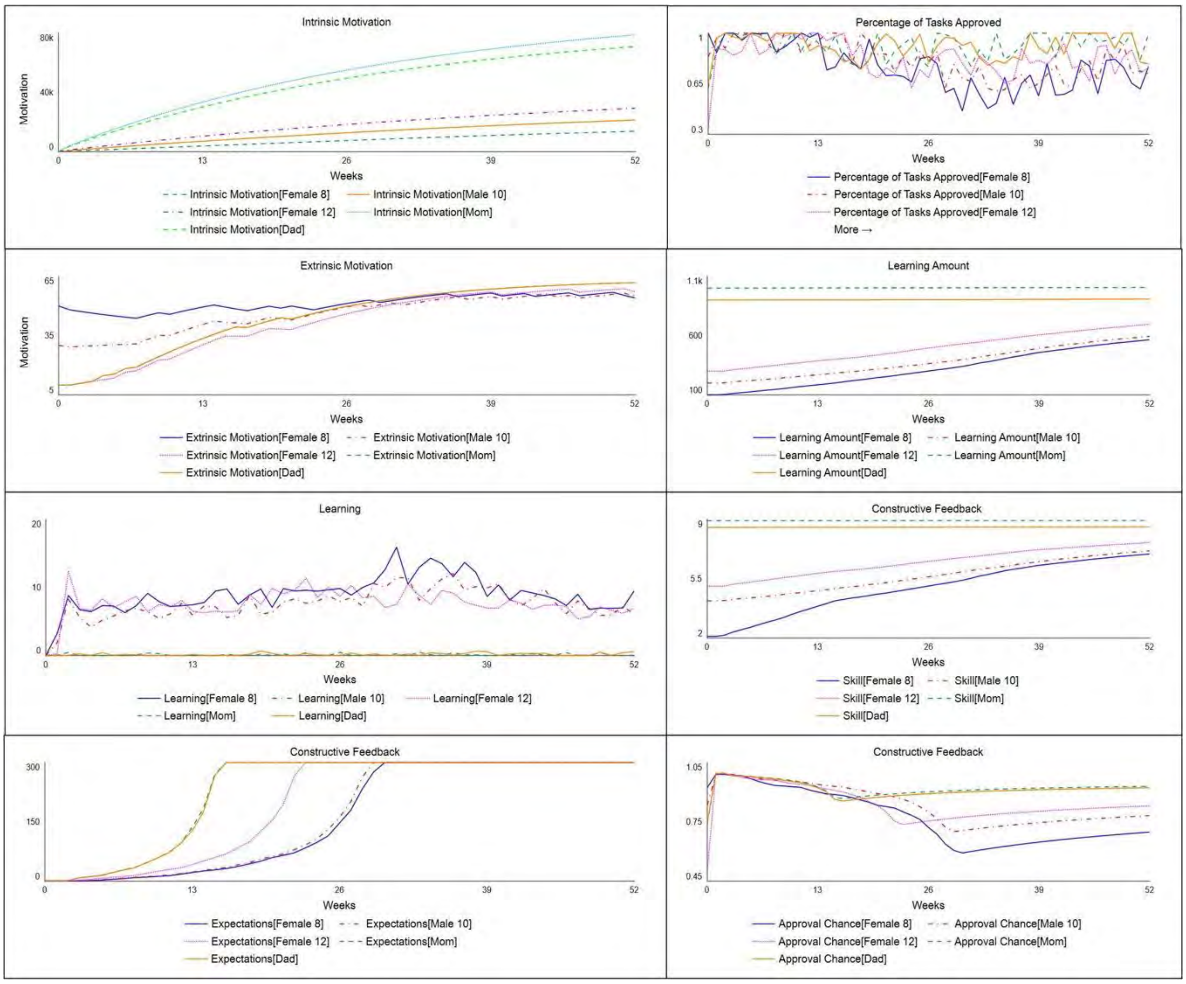


Figure 45. Stella simulation with 100 percent constructive feedback.

They also point out that the app would be 'less efficient without a message system that gives the users constructive feedback, together with giving the power of choice and the feeling of mastery'. The article points out that the students' examples are not complete, and that System Thinking can be used in gamifications in many more ways – for example:

1. To give a more complete understanding of the real-world system gamification builds on.
2. To give a broader and more thorough understanding of how the players learn and reach their goals.
3. To allow experimentation with balances, testing, server loads, marketing, distribution, debugging, etc.
4. As with the game design in Article 3, the students report that System Thinking makes them dig deeper into the design, and makes them pay more attention to how the game mechanics works.

By deploying psychology and pedagogics, the students also learn to ground their design rather than just use common sense, and the students are engaged in scientifically based reasoning rather than wishful thinking. System Dynamics Modeling is quantitative and offers a better understanding of the value ranges of the in and out data from the gamification, and if the students simulate the gamification fully, the dynamic simulation can be used to design the programming code for the gamification application.

An eagle's overview

This paragraph summarizes what is common for the projects, how the games are used in the various projects and project parts, how System Thinking is integrated into the projects, and, finally, what elements of learning methodology are common for the projects.

To what types of Serious Games do the projects belong?

All articles are about Serious Games for learning or education. Although the main parts of the articles discuss Serious Games for higher education or adult education, there are also examples of games that are made for and played in youth education.

How are games used in the projects?

The three projects give different perspectives on how gaming can be used as a serious tool in teaching.

Project 1 uses gaming in two ways. Firstly, it allows the students to develop games that teach them programming, math, and physics. Secondly, it uses gamification as a tool to make the class fun, in that it enables the students, groups, the class, and online peers, to compete to make the best game(s). The first project shows how game development can be used as a tool and a common goal for the class to learn hard science. The gamification added to this class is an example of how competition between students, groups, and peers on the Internet can be used to make the class more fun and interesting.

Project 2 also uses game development as a tool. The students create a pervasive game and three digital games that are aimed at teaching children sustainability and System Thinking. This project has a gamification part too, but it differs from Project 1 in that the goal is to be able to teach a topic to the children in a fun way. This project adds to the first, in that it gives an example of how gameplaying by the children is also gamification that makes serious topics more fun to learn.

Project 3 is also about game development and design but goes far deeper into game design than Projects 1 and 2. Article 3 presents game assignment examples from a System Thinking class. In this class, the students work on assignments to use System Thinking to understand and solve real-world problems, game balancing, testing, and game design. In all these tasks, the students learn how to use System Analysis and System Dynamics Modeling to understand, simulate, and test their game design. The course started as a basic course to learn System Thinking, but has evolved more and more toward being an advanced course in how to use System Thinking in game design. Although they still learn the basics of System Thinking and simulation, the game design processes are always discussed where relevant during presentations and guidance of assignments. The course is in the third semester, which means that the students have already learned basic game design and development, which makes this class an advanced game design class. *Article 4* is about gamification design, with examples from a class in gamification. In this class, the students learn basic gamification design. As the class is in the fourth semester, after the students have learned System Thinking, the students are expected to use System Thinking and simulation in all parts of the gamification design, such as the learning loop, the overall game design, the psychology, and the game balancing and testing, etc. All in all, Project 3 mixes game and gamification design with System Thinking at a deep level, and can be viewed as a form of advanced gamification design, where game and gamification design are in focus and System Thinking is a tool to understand and develop games and gamifications.

How is System Thinking used in the projects?

Projects 2 and 3 (and, to some extent, also Project 1) give examples of how System Thinking can be used as a tool to understand, design, document, and simulate/develop the projects.

Project 1 took place before we started teaching and using System Thinking in Game School, and it gives few examples of how System Thinking could be used in the project. However, the System Thinking principles are also present in Project 1. This can be seen from the learning loop in Figure 46 below. However, there is no doubt that System Thinking could have been used to a greater extent in this project if it had been part of the game education program at that time. For example, students could have been taught System Analysis at a very early stage of the course and used it in their game design in the form of CLD and flow charts. System Analysis could also have been used to design the learning process for the students and, could have been used as a tool by the tutors and teachers, as shown in the examples below (Figures 46 and 47).

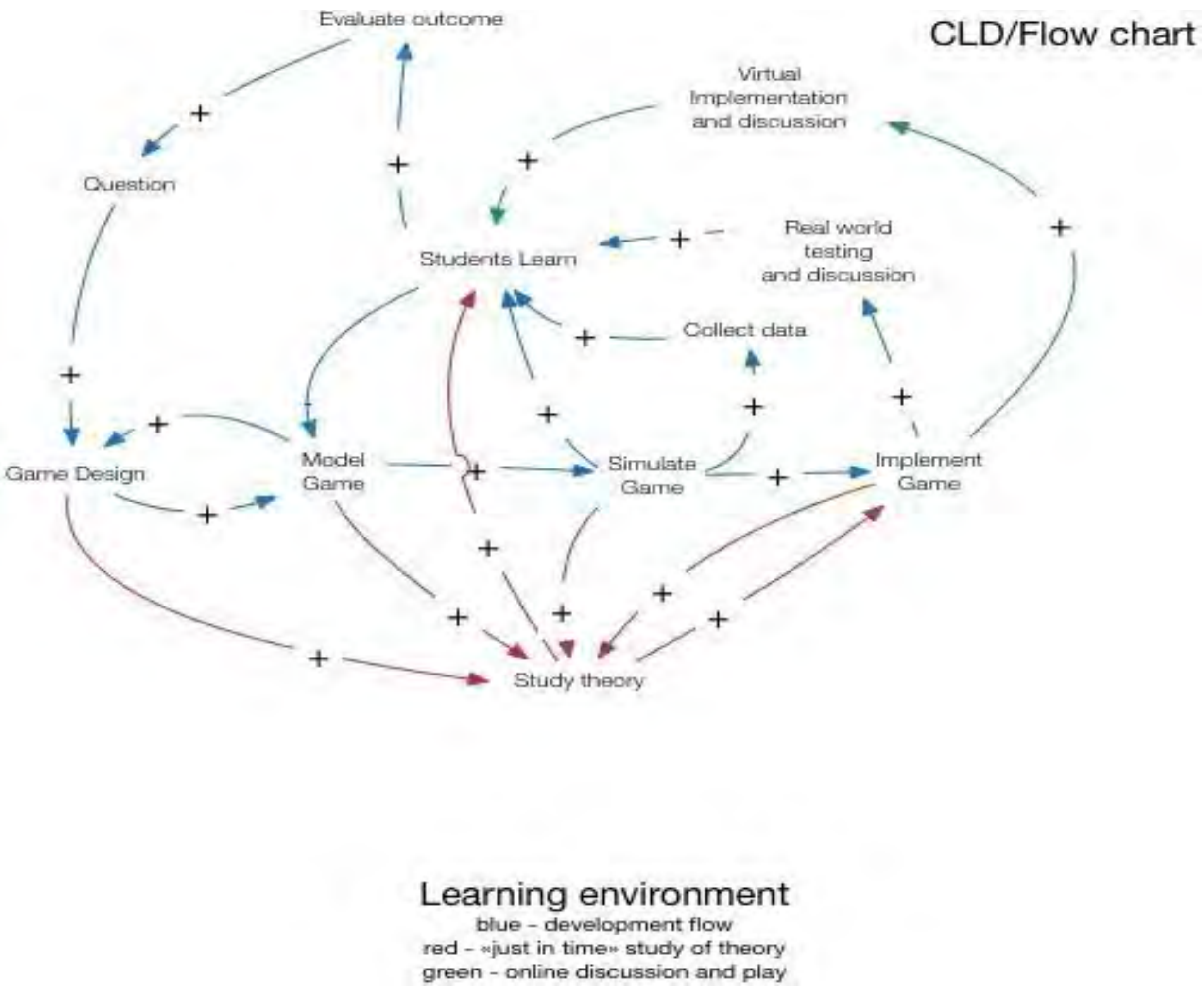


Figure 46. Learning loop for Project 1 (from Article 2 for Project 1 in Appendix 1).

Project 2 was conducted in a three to four-week period at the end of a third semester course in Games and System Thinking, which at that time was basically a traditional course in System Analysis and System Dynamic Modeling. The students had basic knowledge of System Thinking. They also had knowledge of game design and game development from the first year of their studies. The project – to make a pervasive game to teach fifth and seventh graders System Thinking and sustainability – was quite a challenge. Sustainability was only introduced to them through the System Thinking assignments in this class, so they needed help to understand this. When brainstorming the task with them, it was decided to create three digital games: The Island Game, The Earthworm Game, and The Refugee Game (see descriptions of these below). To help the students understand the basic problems in these games, we made three causal loop diagrams (one for each game) to explain how causes and effects were connected in the games. Their first System Thinking task was to read and understand these.

The students then created their own game CLD and flow charts for both the pervasive game and the three digital games, which they would later use in the report describing the game. There was also a third System Thinking challenge in this project: the children also drew a CLD after playing the game, to show how everything they had learned through the game was connected. This proved to be a great success, and the children really learned the principles of drawing CLDs quickly and showed the students that causal thinking concerning causes and effects was kind of natural to the children. The learning loop for the project is shown below.

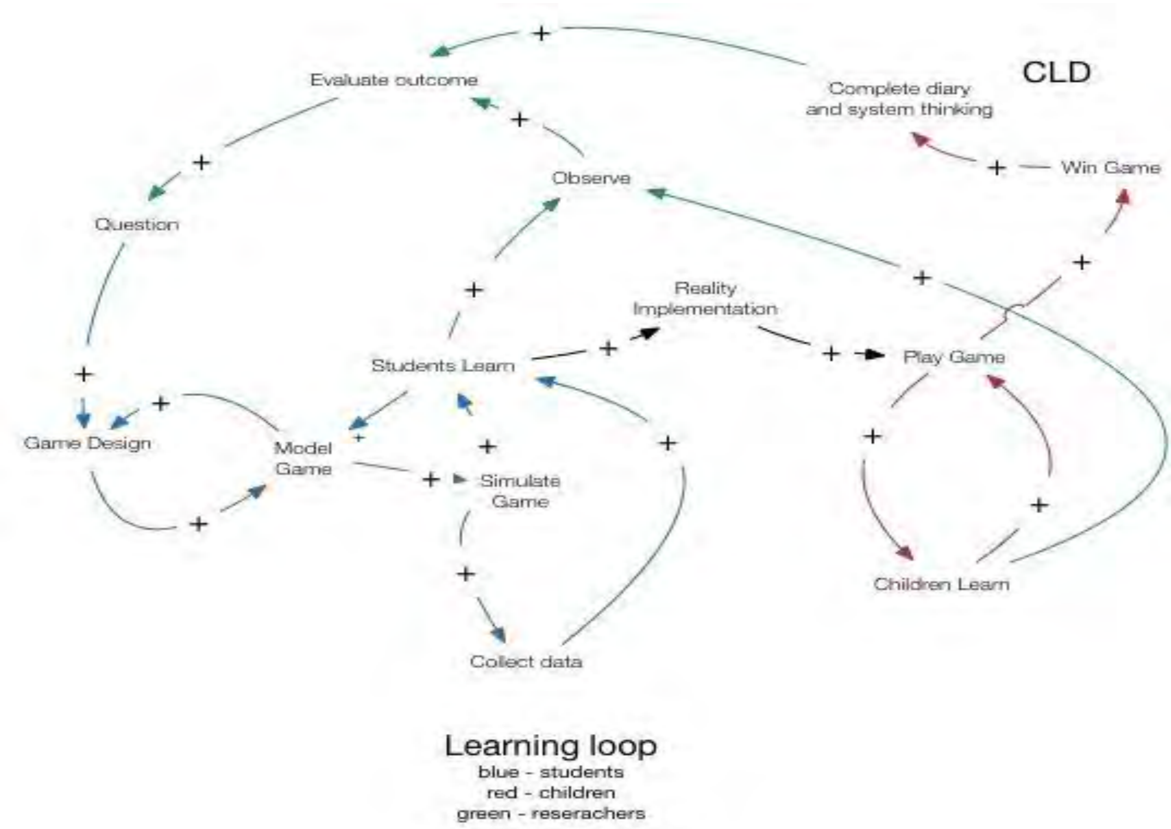


Figure 47. Project 2 learning loop (from Article 2 for Project 2 in Appendix 1).

Project 3, as described in the last paragraph, uses System Thinking integrated with game and gamification design at a deep level. Article 3 gives examples of using System Thinking in the following areas:

- Balancing players and enemies in game design.
- Balancing player groups and guilds in online games.
- System analysis of the game design using CLD and flowcharts.

- System Dynamics Modeling of the designed games and the game balances to debug and tune the games.

Article 4 discusses gamification design. Gamification design uses elements from game design and System Thinking. However, gamification doesn't usually result in a game, but rather an app or description that can be used to make traditional learning or changes of habit more fun. Motivational psychology is an important factor, and the System Analysis or simulations for gamification design must also include psychological design in the gamification. Article 4 gives one example of this, where psychological motivation theories are simulated to balance the gamification.
Summarizing how System Thinking is used in the three projects gives us the following list of tasks in which System Thinking is used:

- In the game and gamification design, as a tool to gain an overview of the design and clarify how rules, players, and scenarios are connected.
- In the game and gamification development, balancing of characters, groups, and scenarios.
- In gamification/Serious Game development, to design the connection between motivation psychology and game elements.
- In game and gamification testing, to debug the game/gamification.

In addition to this, many smaller tasks in the game can be understood better by using System Thinking and System Dynamics Modeling to research the specific task.

Which learning methodology is used in the projects?

The learning methodology was about the same for all three projects.

In Project 1, we started out by introducing visualization, and we started to use Flash and ActionScript as the programming environment. The idea was to make the programming more visual, by allowing the students to immediately see what the programming code did to the graphics. For example, when the students started programming, they had already designed characters they wanted to move, and their programming immediately gave them a visual result for the movement of their characters. Even though Adobe Flash was discontinued by Adobe, the visualization thought was carried on in the new C++ code environments. This has been possible in game engines such as Unity and Unreal.

Next, problem-based learning was introduced. The game development in the first project was divided into cases that the students solved in groups, with higher-level students as tutors. Although we started out using the traditional Maastricht model, we quickly found it to be too strict and modified it slightly. The most important changes were that, although the students worked in groups, each student made their own games. The seven steps also proved to be too detailed, and quickly evolved into two meetings: one before the work was done, and one after. However, after these changes were made, problem-based learning was retained throughout the life of Project 1. Problem-based learning was also part of the methodology for Project 2. During the development of the game, the students worked in groups (one group for each game), with the teachers as the tutors. In this project, the students solved the game together, which is more like the traditional problem-based methodology. The children who played the game also solved problems that build on each other in groups, with the actors (students) as tutors. Project 3 also used a problem-based methodology. In the Games and System Thinking class, the students solved several problems in groups, with the teachers as the tutors. The problems gradually became more difficult. In the gamification class, the students solved one gamification development project in groups, with the teachers as tutors. The project was divided into cases. This class is much like the programming class in Project 1.

All projects have a shared vision, in that they use game development or gameplaying as the tool to learn hard science. Project 1 grounds this shared vision in a community of practice (Lave & Wenger, 1991) – the students develop a game while they learn programming, math, and physics. The shared vision is continued in Project 2. The students that create the game have a shared vision: they want to teach children System Thinking and sustainability through gameplaying, while they are learning System Thinking. However, the children that play their game have solving the game as a shared vision while they learn System Thinking and sustainability. Project 3 has a shared vision too – in the Games and System Thinking class, the students solve problems to make them good game designers while they learn System Thinking. In the gamification class, the shared vision is to make a good gamification, and they learn how to do that by applying System Thinking, game design, and psychology.

In all three projects, the students clearly do something. In Projects 1 and 2, the students make games, and in Project 2, the children play the game, and learn. In Project 3, they also do something when they solve System Thinking problems or make gamifications. All projects are learning by doing (Dewey, 1916).

Learning by doing also creates events for learning when the motivation is at its highest (Gee, 2007). If planned carefully, these learning events will both be at the right time and have the right content. In Project 1, one of the biggest challenges that we worked on throughout the project was to ensure the right learning intervals between the cases. They should not be too big, as then the students would give up, and not too small either, as then the students would get bored. This also constantly challenged us in terms of the sequence of the learning events. In Project 2, which was an ad hoc project made in three to four weeks, we also put a lot of effort into planning the learning events for the children. We did a fairly good job on the events, in that the children solved all of them, but we also underestimated the children in a big way, as they were much smarter than we had thought, and they solved the game in half the time that we planned. In Project 3, the learning events are the cases in System Thinking in the System Thinking course, and the gamification case design in the gamification class. In the System Thinking course, we inherited the case design for the sustainability tasks from System Thinking courses at other universities, such as Lund, Stockholm, and Iceland. The game cases are new, however, and we experimented with where to place them, to achieve the best flow. The gamification course was modified several times over the years, to ensure the right flow and the right content of the cases, to give the right learning intervals.

In all projects, we worked to make the cases open-ended, so that the “clever” students could dig deeper and not just move on to the next case. We learned that this was a smart approach at an early stage of Project 1 – we needed the good students to help the others, and this gave them an opportunity not only to do that, but also to show off how good they were, by digging deeper into the case. During the years after my PhD thesis in pedagogics was stopped and we gradually increased the System Thinking, the teaching methodology thinking was gradually moved more toward the System Thinking methodology described in *The Fifth Discipline* (Senge, 1991). As not much of this was written in the articles included in the PhD thesis, I have chosen to describe this as something that we want to do in the future and will discuss it briefly in the Discussion below.

Discussion

The way in which System Thinking is used to make Serious Games has developed quite a bit since it was introduced in the game education programs and the Game School in 2014. The *Serious Games Learning Loop*, as described in the theory part in Figures 6 and 11, is the culmination of the research and experience from the introduction of System Thinking in practice. The *Serious Games Learning Loop* is also the blueprint for formulating the steps for the practical implementation of the theory. Figure 48 shows the proposed *Serious Game Development Cycle*, illustrated in four phases from Article 3 from 2018. The four phases are: 1) vision of a game and mission statement, where the game idea is brainstormed, 2) the System Analysis loop (R1), where CLDs, flow charts and RBP are used to design a qualitative model of the game, 3) the System Dynamics Modeling loop (R2), where the model is simulated and the code is exported to the game engine, and finally 4) the Game Production Phase loop (R3), where the art, psychology, pedagogics, and game logics are programmed in the game engine and made into the final game. The final phase is clearly based on the learning and design of the previous phases.

Figure 11 adds to Figure 48 by showing how System Science/System Thinking, together with situated and problem-based learning, bring together individual skills, team skills, and mental models, and help to create a shared vision (Senge, 1990).

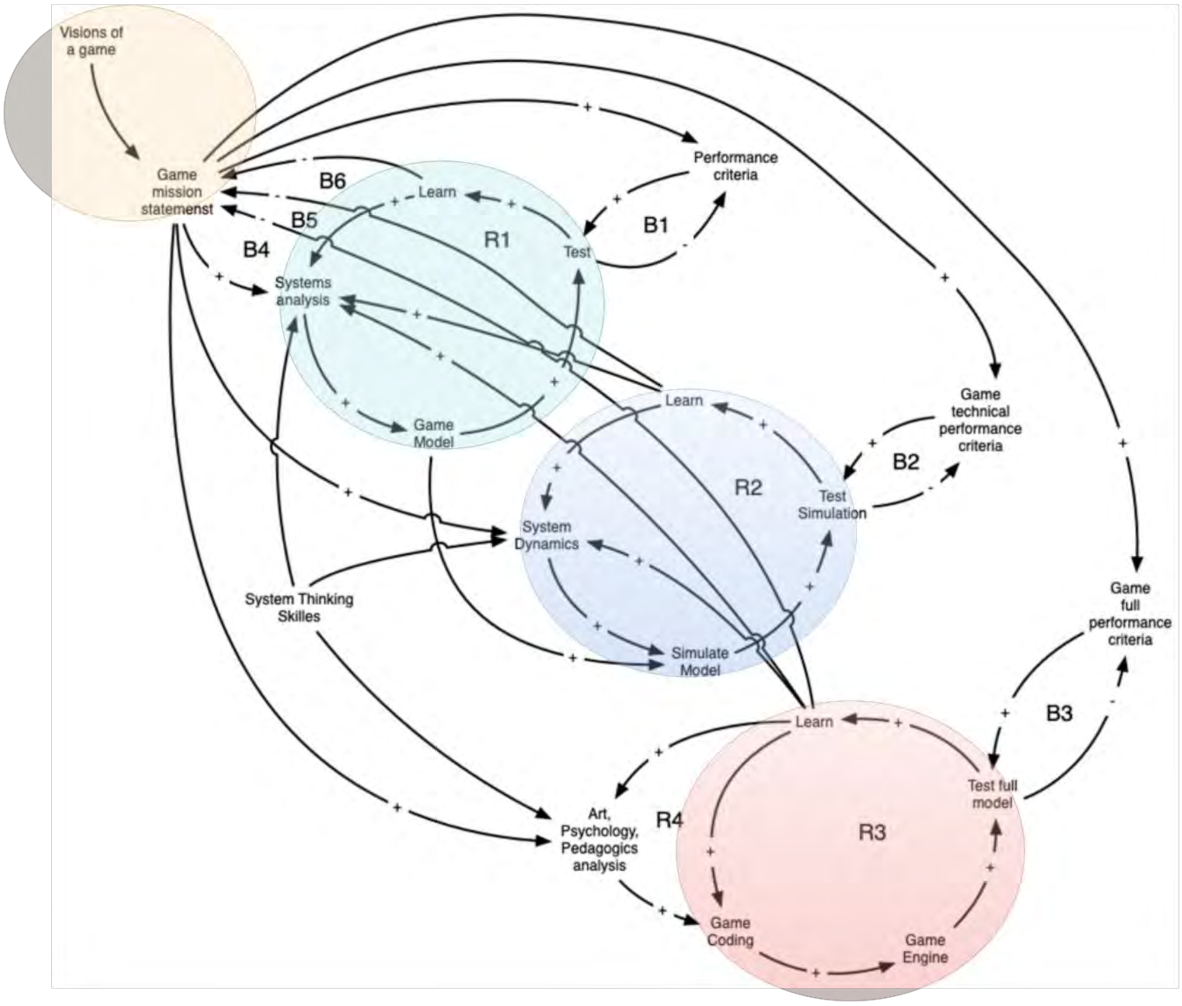


Figure 48. The Serious Games Development Cycle.

The *Serious Game Development Cycle* needs to be placed into a practical framework in a stepwise approach. Below, this is presented in two frameworks for how to implement System Thinking in Serious Game development. Framework 1 sums up how to understand, design, develop, and document the process of Serious Gaming development. Framework 2 summarizes the learning methodologies used to derive the best effect from System Thinking. We will then discuss some possible enhancements for the coming years, before finally considering wider visions for Serious Games.

Framework 1: Steps to follow for System Thinking in game and gamification design and development

1. *Study the source*.
   - By examining a brief or description that describes the task.
   - By detecting systems, components, and connections in the real-world system that are used as the starting point. This is often how to start when making gamifications.
2. *Brainstorm* to find a great game or gamification idea.
3. *Analyze* the data from the brainstorming and structure it. Ask a clear question and prepare a *Mission Statement* for what is to be achieved.
4. *Conceptualize – do the System Analysis* – by defining and sorting the components into a table of items, actions, and controls, and then follow the described process in the theory chapter on System Thinking.
   - It is important, however, that when undertaking System Analysis of the game, the CLD must contain all inputs and outputs in the game. The inputs will have to include all user influence and relevant setup parameters. The output will include all important output parameters necessary to be able to test the game.
   - Three types of CLD are currently made in System Analysis:
     - A CLD learning loop that shows how players learn in the Serious Game.
     - A CLD of the game that describes how the parts of the game influence each other.
     - In gamification, the students also create a CLD that connects motivation psychology and game elements (based on the motivation models such as ARCS, SDT, Malone, or Lepper, etc.)
5. *Simulate – do System Dynamic Modeling.* This is a quantitative process, and numerical data for all parts of the system will have to be collected. The simulations will usually force redesign, and it is then very important to keep the analyses and simulation updated throughout the design process. This enables testing in all three design phases (pre-production, production, and post-production) before it can even be tested by people. Doing this and setting up many different scenarios can save time and money, and, even more importantly, it will be possible to use the simulation data or even the simulation equations in the game or game engine.
6. *Write the simulation code for the game engine.* In Stella Architect, it is possible to export pseudo code and sequencing for the simulation, so that the simulation can be re-created in

programming code in the game engine. This enables the simulations to be run at any time during the game, to create new data sets.

7. *Make the gamified user interface with AR and VR.*

Framework 2: Steps to follow to set up the learning methodology

The result of the implementation depends on having found an interesting, shared vision that can give the learners interesting problems to solve. It must be possible to work in groups, and the Internet should be available most of the time. A problem-based learning approach will get you started (Barrows & Tamblyn, 1980; Maastricht University, 2013).

1. *Make a shared vision* for the class. Examples can be to make a game, to build or explore something, or to solve a mystery, etc.
2. *Define a set of learning goals* for the class from the overall competence aims or curriculum.
3. *Make a set of interesting problems* from the learning goals (cases) that build on each other. In each case, make learning events that force the students to study theory. The learning events should arise in a natural way, so that the motivation to make the game work pushes their theoretical studies. Make sure the learning curve is not too steep or too easy. At this point, it should also be outlined how System Thinking should be used in each case.
4. *Divide the class into groups*. The group can serve as a discussion group only, when individual members make their own game, or they can make the game together. For each case, the students do the following:
    - The group meets to brainstorm and do System Thinking, define the learning goals, and make a schedule for the work in this case. They also discuss how System Thinking should be used in their own games.
    - The students work on their own or together to solve the case.
    - Meet again to sum up and discuss the results. This is also when the System Thinking is discussed and updated.
5. *Encourage the use of the Internet* to find forums and peers where the students can present and discuss their project and get feedback, at individual, group, and class levels. This integrates informal learning environments into the classroom. Both the groups and

the individual members should be able to take on identities they can use online. This underpins both Team Learning and Personal Mastery (see below).

6. *Encourage and facilitate competition* between the students, the groups, and online in forums. In this way, you can gamify the class. Make sure you balance intrinsic motivation and extrinsic motivation, and don't allow the latter to kill the former (Kapp, 2012).
7. *Sum up and classify the learning*. Game-based learning is ad hoc, and the students/children will need help to structure and classify their learning.

Methodological discussions

There are certain things that we have had to modify or change in the methods we have developed and used over the years.

Changes and modifications related to problem-based learning (PBL) and pedagogics have been discussed elsewhere (Nordby & Karlsen, 2014; Nordby, 2015; Nordby et al., 2016). For example, the students found that the Maastricht model (Maastricht University, 2013) was too strict in our problem-based learning sessions and had to be adjusted. Instead of seven steps, it was modified to involve two meetings. Further, we did not always allow the PBL groups to work together to find and develop a solution. In some projects (e.g., Project 1), each student develops their own game, but we allow the group to be a place for them to discuss common problems and their approaches and solutions. We learned that it was important to make the tasks open-ended in order to keep all group members working on the same case and help each other. If possible, cases should also be designed so that they can be solved in different ways. Higher-level students can be used as tutors. It also proved hard to find the "right" learning interval – not too hard and not too easy – between different cases. Finally, always using PBL is not right for all students, so we made sure to include some workshops, and even lectures. We suggest six to ten students in each group, to maximize creativity (Darsø, 2001).

Some students will need more attention, and their learning styles are different, so ordinary teaching methods should not be abandoned entirely. If you think that some students need an ordinary lecture, then give it to them. You will probably receive more attention and a different response than you are used to, because it is given just in time, and the students will want to continue developing their game. Teaching the theoretical science while they are about to code it for their game can really be more interesting, effective, and rewarding.

Situated learning needs a common goal to make a community of practice and most of the 14 indicators described by Lave & Wenger (1991) and Wenger (1998) should be in place. We believe they were, although we didn't discuss every indicator in detail (Nordby & Karlsen, 2014). Maybe doing more research on the indicator would have proven to give even better results.

In the course in Games and System Thinking, we have been making game-related tasks to relate to the game students. These are described in the final two PhD thesis articles – System Thinking in Game Design, and System Thinking in Gamification (Nordby et al, 2023) – and have proven to be very popular, which probably also helps learning. Over time, we have also developed certain requirements for CLDs that are needed to make Serious Games, such as learning loop CLDs, and CLDs to link game elements to motivation psychology (Nordby et al., 2015, 2016). It has not, however, been necessary to modify basic CLD design to do this. We have also learned that it takes time for the students to learn to spot the systems in games, and we have therefore introduced System Thinking in the first study year, where they already start looking for the systems in their first game design course. In general, I have yet to find a situation where System Thinking does not work. However, as students have also pointed out, there can be cases where System Analysis alone is enough (Nordby et al., 2015) without a need for System Dynamics. There can also be situations where doing System Thinking is too much work compared to just solving the issue without doing System Thinking. A good example of the latter is the project described by Nordby et al. (2015, 2016), where the students engage in thorough System Thinking on the big pervasive game, but not on the small games that are part of the big games (Nordby et al., 2015, 2016).

Looking back at the history of the projects until 2019, a clear pedagogical development can be seen to have taken place. It is easy to draw a comparison between Senge's (1991) five disciplines and the methodology that we used when we started out. Right from the start, we introduced a shared vision in the game development undertaken by the students. Their goal was clearly to make a game. It was also a competition to make the best game. The team learning took place in the problem-based learning groups. However, in contrast to traditional PBL groups, where they make a game together, each student made their own game. This ensured a form of personal mastery, as every student had to personally learn to make a game. We can also argue that the vibrant situated learning community of practice (Lave & Wegner, 1991) that we created in the class could "unearth" the students' mental models of the world and hold them up to rigorous scrutiny (Senge, 1991, p. 8).

In 2012, we started to teach System Thinking, and the fifth discipline entered the scene; this made it possible to weave everything together. This is what the thesis is all about. System Thinking was always there in the design, hidden in the texts, but it had to be lifted out, visualized, and simulated using System Thinking, System Analysis, and System Dynamics.

Visions and the wider perspectives

As we see it today, there are a few things that we will work on and add in the years to come. Below is a list of six points:

1. System detection, component detection, connections, and basic System Analysis.
2. Game and gamification design, and user interface.
3. Technical extensions to make better user interfaces, faster code, and stable distribution.
4. Project management, marketing, and documentation.
5. Game programming.
6. Agent programming.

1 System detection, component detection, connections, and basic Systems Analysis

Systems are everywhere, including in nature and in games. James Paul Gee points out that games are systems (Gee, 2007). However , it takes time to get used to seeing the systems around us, and it takes even more time to make it a habit to analyze and understand them by looking for their components, and understanding how they are connected and where the feedback loops are.

In the game education program, students start to learn game design at an early stage, and large elements of this topic are studies of good or popular games. This is a good place to start when teaching students to look for and detect systems. For every game they study, there should also be discussion of where the systems are, what components they are built from, and how these work together. Basic qualitative System Analysis can also be taught early in the game education program, or to game or gamification design teams. This is a simple language that can be taught to anyone, and it does not require any previous knowledge. We suggest that teaching of basic System Analysis should also start with the first game design course. It should then be used for students to discuss and understand systems that are found in games they analyze, as well as in their own designs.

In the same way, System Thinking should also be taught to teachers, so that teachers start teaching children System Thinking in elementary school. We already do this to some extent, as part of the teacher course we teach in the fields of games, animation, and learning. However, System Thinking for teachers should be extended, so that they also have time to understand this enough to see how important it is for the children to learn. As an example, the groundwork for this has already been laid in the K12 education in the US.

2 Game and gamification design, and user interface

The use of System Thinking in game and gamification design is an area that has got more focus every year since 2014. However, we can still see new areas where we should give it even more focus,

One area requiring constant work is to make the assignments relevant. This can be done by introducing new assignments every year that are relevant to the current world situation, such as new and relevant cases on climate change or virus attacks, etc. Furthermore, students should find the cases to be relevant for them, which can be done by asking them to make interesting, gamified user interfaces using VR or AR. Both these tasks are naturally also very relevant "to the world", and may not only "get the message across" but also make the public want to learn about it.

In broadening the perspective of the assignment, we see an increasing need to strengthen the student's perspective on and understanding of "how the world works". This topic should be taught more explicitly in the existing courses, but also be extended to a broader overview at master level.

We can also see that System Thinking needs more "space" in the education programs. This is not possible, however, for the bachelor program, in the form of new courses, because the bachelor program needs all the courses and topics that are already there. However, some changes and extensions can be made, such as the extension of system detection in the game design described above.

The System Thinking should be extended to master level, however, and we plan to do this in several ways for our new master programs. We will not consider that here, but we are generally making the bachelor program courses in System Thinking and gamification more technical (see the paragraph on the technical aspects), while much of the existing design and content of the course in System Thinking and gamification has been moved to the master program. The main benefit of this is that we gain more time to make System Thinking more thorough and a cornerstone of the game education programs, with a focus on Serious Games and gamification.

3 Technical extensions to make better user interfaces, faster code, and stable distribution

Although Stella Architect is a good tool for System Dynamics Modeling, it is not very fast when the systems become large. We therefore see a need to write the simulations from scratch in a dedicated programming language, such as C++ or C sharp, or to export the results from Stella, and use the pseudo code to build a native application.

Furthermore, we need to put the simulation code in the game code, to be able to rerun the simulations inside the game if necessary, and finally, to be able to extend the simulations/Serious Games with a game interface or an AVR or AI application.

Many Serious Games or gamifications need to work online. System Thinking and System Dynamics Modeling are good tools to simulate and test distributed systems and server loads, etc. By keeping an updated simulation of the game current – not only throughout the game design period but also after the game has been released – game designers will be able to simulate how user loads affect their design at runtime load. This could save both time and money. As stated above, we see these technical extensions of the education program as belonging to the bachelor program, and we will already change the courses to include this next year. This is a big change, however, and it will make the courses more focused on the technical aspects and less focused on the broader perspectives of System Thinking. We will not lose these aspects, however, but rather move them to the master program.

4 Project management, marketing, and documentation

System Thinking can also be used for project management, marketing, and documentation. Project management and marketing will be part of the new master program. Documentation on how to use and maintain gamification or Serious Games will be integrated more closely into both the bachelor and the master programs.

5 Game programming

The System Analysis for the game will also be the basis for programming Serious Games. It is very important that the programmers stick to the design described by the System Analysis, as well as frequently checking that the code also does this. If changes must be made, they should be made not only to the programming but also to the entire design, including System Analysis and System Dynamics. If necessary, new simulations should also be carried out.

6 Agent Programming

System Analysis and System Dynamics can also be of help in designing and simulating game agents and artificial intelligence (AI). System Analysis is important in the design process, whilst System Dynamics Modeling can be a key resource for testing parts of the AI design before implementing it in the game engine. Here, code can also be exported from the simulations for use in the game engine. Although the above list gives an overview of the smaller enhancements in order to further extend the use of System Thinking and teaching methodology, it does not provide any long-term vision of where this work might take us. To view the wider picture, we will have to dig deeper into what game technology really is, and what we can do with it. In overall terms, it is clear that the rapid development of computer technology has brought us to this point, and forms the basis for the wider visions.

Virtual interactive worlds

When we first started Game School back in 2003, we already had a focus on programming. We knew that we needed good programmers to be able to build games. From that period, I remember how DigiPen – a school in Vancouver, British Columbia, that also worked with game development – named their technical education program *Virtual Interactive Real-time Simulations (VIRS)*. I still remember thinking, “Wow, this really describes what we will make.” That name really nailed what we are doing, and still does. We are making models and simulations of the real or fantasy worlds. These simulations are made with high-quality graphics and sounds that move in real time, and not only can we look at them, but we can also walk into them by using avatars. When we do this, they can feel real and evoke some of the same emotions we have when we do things in the real world. They can also be interactive, which means they can respond to what we do, when programmed correctly. Virtual reality and augmented reality (VR and AR, respectively, or often together called XR) also add to this feeling. Both have been around for many years, but it is only in recent years that VIRS has become fast enough to support them.

So, now we can make virtual interactive real-time simulations, and what is the big deal about that? Think about what these simulations can be: they can be worlds with any content, like our real world, or like fantasy worlds; and they are virtual, which means they are not limited to a physical place. They can be everywhere, and anyone in the world that has a computer, a tablet, or a phone can enter them with an avatar. For a few years now, we have seen them as online games or virtual worlds on the screen, but more recently we have also been able to wear AR or VR glasses, which make it feel as though we are there; and doing something in these worlds, such as riding a rollercoaster, gives us the same exhilarating feelings as we would experience in a real amusement park.

These simulations can be filled with any content – as places where we just meet and do things together, as places where we build things, alone or together (Minecraft, Sims, Second Life, etc.), or they can be places where we play games, alone (any console or PC game) or together online (WOW, Anarchy online, Eve online, etc.). They can be places where children meet friends and talk, and our friends can now be from any place or country in the world (Club Penguin, Start Doll, etc.).

They could also be gamifications or Serious Games that enable us to learn things. The important point is that, to be able to build these places, we need all the skills we learn in school. Project 1 is a good example. We need to learn not only programming to make games, but also math and physics. If the building takes place online, this might require us to cooperate with people from other countries, which could teach us languages, cooperation, teamwork, etc. In many instances, we would have to write descriptions, notes, etc. This also requires artistic skills, such as design and drawing, animations, sound design, etc. There is no topic left that we will not need, and the need arises while we are doing things, as in games. It will be learning by doing and just in time.

This is where the visions become bigger, as we can use RIS to learn school topics by doing things and by having fun. We can explore everything we learn, with no fear of hurting ourselves or damaging something. And we can do this with anyone from any part of the world. So where does System Thinking fit in?

System Thinking

The development of these virtual worlds and simulations has so far mainly been undertaken by large teams of people with different skills. For extensive online games, these teams can be very large, with hundreds of people, and their work continues for many years. These teams need professional, proven scientific methodology to organize, systemize, and simulate their work. That is what System Thinking is. System Analysis is a set of tools that enables them to analyze a system in the world, or in their virtual worlds, to spot the systems and the components, and to see how they work together as a whole. This constitutes the manual and drawings for the virtual world. System Dynamics Modeling is a simulation of the world. This enables us to test how well our design will work before we build it. We can test different scenarios and components and see how well our design behaves and predict how well it will behave in the future. We are developing System Thinking into a methodology that can be used to design, develop, and test RIS environments in a scientific way.

Conclusion

The PhD thesis has discussed and suggested frameworks that adopt methods, approaches, and tools provided by System Thinking in order to understand, design, develop, and document Serious Gaming development. This is illustrated in the Serious Games Development Cycle in Figure 48 and the frameworks presented in the discussion chapter. The Serious Gaming Learning Loop in Figure 11 also shows how System Science/System Thinking is also a language, a toolset, and a fifth discipline to connect all parts of the Serious Game development together. Through this work, I hope to have shed some light on both how interesting virtual Serious Games and simulations are for learning, and how the development can be improved and strengthened by adopting a more scientific approach and methodology such as System Thinking.

Virtual simulations and Serious Games are largely made possible by the rapid advances in game technology. The combination of programming, art, and animations has made possible simulations and virtual environments that can look and feel very much like the real world. The recent addition of other disciplines such as psychology and pedagogics can make these simulations into goal-oriented learning systems, and System Thinking can turn the development process into well-defined science. So, what can we expect to see in the future, and what are the visions?

- Serious Games can revolutionize the school system. Finally, we have technology that enables children and students to practice learning by doing, which Dewey had already stated was the best form of learning back in the early 19th century.
- We can revolutionize any training by making simulators where skills and reactions can be trained by trial and error without consequences. Flight simulators are the classic example, as we can learn to fly planes and crash as often as necessary, without any costs or damage. It is easy to see many other applications, and the military is already using them to teach people to pilot tanks and shoot cannons, and applications are also used in healthcare, together with VR.
- Internet-based simulations and Serious Games can enable people from all over the world to work and learn together. There are already many examples of online games where this happens informally, such as WOW and Second Life for adults, or Minecraft, Stardoll, and Club Penguin for kids.

- We can make learning fun by making gamifications, and we can adapt it to the new generations.
- And of course, we can make the game students highly skilled and sought-after. Or perhaps we already have, as the industry has already discovered that we train the best programmers, as they also understand System Thinking and how to use math and physics when needed.

Interestingly, although it is fun to play games, and they can motivate learning, it is not gaming that creates the big visions – this requires combining scientific disciplines with the fast-advancing technology, art, and animation to make worldwide, cross-disciplinary, and virtual interactive communication arenas, experiences, and learning environments that feel real. Once we have made that possible, we can add gameplay to make it even more fun. As a final vision, we can perhaps quote Jane McGonigal: ‘Games can save the world’ (Wall Street Journal, 2022; McGonigal, 2023).

References

Abt, C. (1970). *Serious games*. New York: Viking.

Adams, E. (2014). *Fundamentals of game design* (3 ed.). Berkeley, Calif.: New Riders.

Andersen, D. F., Richardson, & George P. (1997). Scripts for group model building: A previous version of this paper was presented at the 1994 International System Dynamics Conference, Stirling, Scotland, 11-15 July 1994. *System Dynamics Review*, *13*(2): 107-129

Andersen, P. B., Bennedsen, J., Brandorff, S., Caspersen, M. E., & Mosegaard, J. (2003). Teaching programming to liberal arts students – A narrative media approach. *ACM SIGCSE Bulletin – Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education*, *35*(3), 109-113.

Arnold, R.D., & Wade, J.P. (2015). A definition of systems thinking: A systems approach. *Procedia Computer Science*. Volume 44, Issue C, 2015, 669-678

BankMyCell (2023). https://www.bankmycell.com/blog/how-many-people-play-video-games

Barab, S. A., Kling, R., & Gray, J. H. (2004). *Designing for virtual communities in the service of learning.* Cambridge: Cambridge University Press.

Barrows, H. S., & Tamblyn, R. (1980). *Problem-based learning: An approach to medical education.* New York: Springer.

Birt, J., & Hovorka, D. (2019). What we know about digital games for learning: A systematic review and meta-analysis of K-12 research. *Review of Educational Research*, *89*(6), 785-820

Churchland, P.S. (1986). *Neuophilosophy: Towards a unified science of the mind/brain*. Cambridge, MA:MIT Press.

Cochran-Smith, M., & Lythle, S.L. (2009). *Inquiry as stance. Practitioners research for the next generation.* New York: Teachers College Press

Coyle, G. (2000). Qualitative and quantitative modelling in system dynamics: Some research questions. *System Dynamics Review*, Volume 16, Issue 3 Fall 2000, 225-244.

Damasio, A.R. (1994). *Decartes' error: Emotion, reason and the human brain.* New York: Avon

Darsø, L. (2001). *Innovation in the making.* Frederiksberg: Samfundslitteratur.

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining "gamification". *MindTrek '11. Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (pp. 9-15).

Deterding, S., Sicart, M., Nacke, L., O'Hara, K. & Dixon, D. (2011). Gamification: Using game design elements in non-gaming contexts. In: *The Gamification Research Network. CHI 2011 Workshop Gamification: Using Game Design Elements in Non-Game Contexts*. http://gamification-research.org/wp-content/uploads/2011/04/CHI_2011_Gamification_Workshop.pdf

Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

Djaouti, D., Alvarez, J., & Jessel, J.-P. (2019) Classifying Serious Games: The G/P/S model Retrieved from: (http://www.ludoscience.com/files/ressources/classifying_serious_games.pdf)

Dochy, F., Segers, M., van den Bossche, P., & Gijbels, D. (2003). Effects of problem based learning: A meta-analysis. *Learning and Instruction*, *13*(5), 533-568.

Donovan, T. (2010). *Replay: The history of video games*. East Sussex, England: Yellow Ant.

Doyle, N. (2015). Why more open world games is a bad thing. Retrieved from http://gamemoir.com/more-open-world-games-bad-thing/

Dørner, D. (1996). *The logic of failure: recognizing and avoiding error in complex situations*. Perseus Books,

Edvardsson, B., & Tronvoll, B. (2013). A new conceptualization of service innovation grounded in S-D logic and service systems. *International Journal of Quality & Service Sciences*, *5*(1), 19-31.

Edvardsson, B., Enquist, B., & Johnston, R. (2005). Cocreating customer value through hyperreality in the prepurchase service experience. *Journal of Service Research*, Vol. 8, 149-61.

Edvardsson, B., Tronvoll, B., & Gruber, T. (2013). Expanding understanding of service exchange and value co-creation. *Journal of the Academy of Marketing Science*, Vol. 39, 327-39.

Education Database Online. (2010). *American households playing games*. Retrieved from http://www.onlineeducation.net/videogame

Egenfeldt-Nielsen, S., Smith, J. H., & Tosca, S. P. (2008). *Understanding video games: The essential introduction*. New York: Routledge.

Egenfeldt-Nielsen, S., Smith, J. H., & Tosca, S. P. (2013). *Understanding video games. The essential introduction* (2nd ed.). New York: Routledge.

Forrester, Jay W. (1969). *Urban Dynamics.* Pegasus Communications. ISBN 978-1-883823-39-9.

Fuglsang, L., & Sørensen, F. (2011). The balance between bricolage and innovation: Management dilemmas in sustainable public innovation. *Service Industries Journal*, *31*(4), 581–595.

Fuglsang, L. & Rønning, R. (2014). Public service sector innovation in context (p.218-233). In Fuglsang L., Rønning R., & Enquist B.: *Framing innovation in public service sectors.* New York, Routledge

Gee, J. P. (2004). *Situated language and learning: A critique of traditional schooling*. London: Routledge.

Gee, J. P. (2007). *Good video games and good learning: Collected essays on video games, learning and literacy.* New York: Peter Lang International Academic Publishers.

Gee, J. P. (2010a*).* Design, learning, and experience [Video file]. Retrieved from http://vimeo.com/16254768

Gee, J. P. (2010b). *New digital media and learning as an emerging area and "worked examples" as one way forward*. Cambridge, MA: MIT Press.

Giddens, A. (1979). *Central problems in social theory: Action, structure and contradiction in social analysis.* Macmillan, London.

Giddens, A. (1984). *The constitution of society: Outline of the theory of structuration*. Polity Press, Cambridge.

Grönroos, C. (2001). The perceived service quality concept – a mistake? *Managing Service Quality: An International Journal*, Vol. 11 No. 3, pp. 150-152.

Hartley, J. (2013). Public and private features of innovation (p.44-59). In Osborne S.P., & Brown L. *Handbook of innovation in public services.* Cheltenham, E. Elgar

Hamari, J., Koivisto, J., & Sarsa, H. (2014). Does Gamification Work? – A Literature Review of Empirical Studies on Gamification. In *proceedings of the 47th Hawaii International Conference on System Sciences,* Hawaii, USA, January 6-9, 2014.

Hamari, J., Huotari, K., & Tolvanen, J. (2015). Gamification and economics. In S.P. Walz, & S. Deterding (Eds.). *The gameful world: Approaches, issues, applications*. Cambridge, MA, London: MIT Press.

Haraldsson, H.V. (2004). *Introduction to systems thinking and causal loops diagrams*. Reports on ecology and environmental engineering. Report 1:2004. 49pp

Haraldsson, H. (2005). *Developing methods for modelling procedures in system analysis and system dynamics*. Lund University.

Haraldsson, H.V., Sverdrup, H. (2004). *Finding simplicity in complexity in biogeochemical modelling*. J. Wiley and Sons Ltd., Chichester, pp. 211-223.

Kapp, K. M. (2012). *The gamification of learning and instruction: Game-based methods and strategies for training and education*. San Francisco: John Wiley & Sons.

Kinnunen, P., & Malmi, L. (2005). Problems in problem-based learning – Experiences, analysis and lessons learned on an introductory programming course. *Informatics in Education*, *4*(2), 193-214.

Kolb, D.A. (1983). *Experiential learning: Experience as the source of learning and development*. Prentice Hall

Krillbite. (n.d.). *Current projects.* Retrieved from http://www.krillbite.com/index.php

Keller, J. (1987). Development and use of the ARCS model of instructional design. *Journal of Instructional Development*.

Kent, S.L. (2001). *The ultimate history of video games: from Pong to Pokémon and beyond – The story behind the craze that touched our lives and changed the world* (1st ed.). Ashland, Kentucky: Prima Games.

Koca, D. (2020). REPORT – *Applied Systems Science Education and Capacity Building Programme.* EIT Climate-KIC. 2018. *Transformation, In Time*. EIT Climate-KIC Strategy 2019–2022. https://www.climatekic.org/wp-content/uploads/2018/12/Transformation-in-time.pdf

KZero. (2011). *Universe chart Q4 2011: Avg user age 10 to 15.* Retrieved from http://www.kzero.co.uk/blog/universe-chart-q4-2011-avg-user-age-10-15/

Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

Lepper, M.R. (1988). Motivational considerations in the study of instruction. Cognition and Instruction, 5 (4), **289–309**. https:// https://doi.org/10.1207/s1532690xci0504_3 Abstract Comments on the work of S. B. Nolen (see record 1989-31135-001).

Long, J. (2007). Just for fun: Using programming games in software programming training and education – A field study of IBM Robocode community. *Journal of Information Technology Education*, *6*, 279-290.

Lundvall, B.-Å. (2010). *National systems of innovation: Toward a theory of innovation and interactive learning*. Anthem press.

Lusch, R.F., & Vargo, S. L. (2014). *Service-dominant logic: Premises, perspectives, possibilities*. Cambridge, UK: Cambridge University Press.

Lyneis D., & Stuntz, L. (2007). System Dynamics in K-12 Education: Lessons learned. *The creative Learning Exchange*, v17, n2, spring 2008.

Maastricht University. (2013). *Problem based learning preparatory website.* Retrieved from http://www.umpblprep.nl/

Maglio, P.P., Vargo, S.L., Caswell, N., & Spohrer, J. (2009). The service system is the basic abstraction of service science Springer-Verlag. DOI 10.1007/s10257-008-0105-1

Maani, K.E., & Cavana, R.Y. (2000). *System thinking and modelling, understanding change and complexity*. Auckland, Prentice Hall.

McGonigal, J. (2023). You Found Me, https://janemcgonigal.com/

Meadows, D.H., Meadows, D.L., Randers, J., & Behrens, W. (1972). *Limits to growth*. Universe Books, New York.

Meadows, D.H., Meadows, D.L., & Randers, J. (1992). *Beyond the limits: Global collapse or sustainable future*. Earthscan Publications Ltd., ISBN 9781853831317

Moondrop. (n.d.). *Main page.* Retrieved from http://moondrop.no/

Malone, T. (1981). Toward a theory of intrinsically motivating instruction. *Cognitive* Science.

Malone, T.W., & Lepper, M.R. (1987). Making learning fun: A taxonomy of intrinsic motivations for learning. In R.E. Snow & M.J Farr (Eds.), *Aptitude, learning, and instruction volume 3: Conative and affective process analyses* (pp. 223-253). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Nelson, M. (2011). *Robocode*. [Java], IBM.

Newzoo & GamesIndustry (2018) Newzoo data shared in GamesIndustry.biz in *Numbers infographic* 2018 edition. Retrieved from: https://www.gamesindustry.biz/global-games-market-to-hit-usd137-9-billion-this-year-newzoo.

Nordby, A., & Karlsen, S. (2014). Teaching 'hardcore science' to arts and design students: Reflections on the development of a basic programming course. *InFormation – Nordic Journal of Art and Research* 2014; Volume 3.(2) s. 129-142

Nordby, A., Øygardslia, K., & Sverdrup, H.U. (2016). The art of gamification: Teaching sustainability and system thinking by pervasive game development. *The Electronic Journal of e-Learning*.

Nordby, A., Vibeto, H., Mobbs, S., & Sverdrup, H.U. (2023). System Thinking in Gamification. *NS Computer Science*. Springer Nature.

Nordby, A., Vibeto, H, Sverdrup, H.U. (2023). System thinking in game design. Not published yet.

Nordby, A. (2015). Teaching “hardcore science” to arts and design students of the gaming generation: Reflecting on how to use game development to teach programming, system thinking, mathematics and physics. Published in EDULEARN15 Barcelona proceedings.

Nordby, A. (2016). The power of game development in learning: Why is game development good learning machines. Published in EDULEARN16 Barcelona proceedings.

Nordby, A., Øygardslia, K., & Sverdrup, H.U. (2015). The best school day ever – Teaching sustainability and system thinking by pervasive game playing. Published in EDULEARN15 Barcelona proceedings.

Nordby, A., & Nordseth, H. (2016). Teachers as learning game designers: Can elementary school teachers with no gaming development background really gamify their own teaching? Published in EDULEARN16 Barcelona proceedings.

Nuutila, E., Törmä, S., & Malmi, L. (2005). PBL and computer programming – The seven steps method with adaptations. *Computer Science Education*, *15*(2), 123-142.

Partti, H., & Karlsen, S. (2010). Reconceptualising musical learning: New media, identity and community in music education. *Music Education Research*, *12*(4), pp. 369-382.

Partti, H., & Westerlund, H. (2013). Envisioning collaborative composing in music education: Learning and negotiation of meaning in operabyyou.com. *British Journal of Music Education*, *30*(2), 207-222.

Pearce, K. (2011). *The ‘gamification’ of education.* Retrieved from http://blogs.canoe.ca/canoetech/signs-of-the-times/72001/

Pettersen, R.C. (2005). *Kvalitetslæring i høgere utdanning: Innføring i problem- og praksisbasert didaktikk.* Oslo: Universitetsforlaget.

Prensky, M. (2011). Computer games and learning: Digital game-based learning. In Raessens, J., & Goldstein, J. (Eds.). *Handbook of computer game studies* (pp. 97-122). Cambridge, MA: MIT Press.

Prahalad, C.K., & Ramaswamy, V. (2003). Co-creation experiences: The next practice in value creation. *Journal of Interactive Marketing*, Volume 18, Issue 3, p5-14

Pramath. (2016). How do developers create massive open worlds for exploratipon? Retrieved from:https://gamingbolt.com/how-do-developers-create-massive-open-worlds-for-exploration.

Radoff, J. (May 2010), History of Social Games, Retrieved from: http://www.radoff.com/blog/2010/05/20/history-of-social-games/

Rajaravivarma, R. (2005). A game-based approach for teaching the introductory programming course. *ACM SIGCSE Bulletin*, *37*(4), 98-102.

Randers, J. (1980). *Elements of the System Dynamic Method*. Cambridge, Productivity Press.

Reason, P., & Bradbury, H. (2001) *Handbook of action research: Participative inquiry and practice*. Sage Publications, Thousand Oaks.

Roofmoose. (n.d.). *Factsheet.* Retrieved from http://roofmoose.com/press/

Rowland, I. (1990). Latrunculi and Ludus Latrunculorum. *Phoenix*, 44(4), 285-301.

Rouwette, E. A. J. A., Vennix J. A. M., et al. (2002). Group model building effectiveness: A review of assessment studies. *System Dynamics Review*, *18*(1): 5-45.

Ryan, R.M., Rigby, C.S., & Przybylski, A. (2006). The motivational pull of video games: A self-determination theory approach. *Motivation and Emotion*, *30*, 347–364.

Ryan, R.M., & Deci, E.L. (2000). Self-determination theory and the facilitation of intrinsic motivation, social development, and well-being. *American Psychologist*, *55*, 68–78.

Rhodes, G. (2004). *Macromedia Flash Mx 2004 Game Development*. Charles River Media, ISBN 1584503092, 9781584503095

Rouse, R. (2005). *Game design: Theory & practice*. Plano, Tex.: Wordware publishing.

Salen, K., & Zimmerman, E. (2004). *Rules of play: Game design fundamentals*. Cambridge, Mass.: MIT Press.

Salen, K., & Zimmerman, E. (2006). Preface. In K. Salen & E. Zimmerman (Eds.), *The game design reader: A rules of play anthology* (pp. xvii-xxvi). Cambridge, Mass.: MIT Press.

Schell, J. (2008). *The art of game design: A book of lenses*. Amsterdam: Elsevier.

Schumpeter, J.A. (1934). *The theory of economic development: An inquiry into profits, capital, credit, interest, and the business cycle*. Harvard University Press, Cambridge, MA.

Schlyter, P., Stjernquist, I., & Sverdrup, H. (2012). Handling complex environmental issues – Formal group modelling as a deliberative platform at the Science-Policy-Democracy interface, *Proceedings of the 30th International Conference of the System Dynamics Society,* Switzerland.

Serious Games (2019) (http://cs.gmu.edu/~gaia/SeriousGames/index.html). cs.gmu.edu. Retrieved June 14 2019

Senge, P. (1990). *The fifth discipline, the art and practice of the learning organization*. Century Business, New York.

Shupe, R., & Rosser, Z. (2008). Learning Actionscript 3.0. Amazon, ISBN 978-1449390174

Smith, R. (2009). *A history of games from dice to computer, Gametech 09, March 10-11 2009*, Orlando, FL

Shaffer, D.W. (2007). *How computer games help children learn.* New York: Palgrave.

Spohrer, J., Maglio, P.P., Bailey, J., & Gruhl, D. (2007). Steps toward a science of service systems. *Computer*, Vol. 40, pp. 71-7.

Statista. (2023). https://www.statista.com/outlook/dmo/digital-media/video-games/worldwide

Steinkuehler, C., Squire, K., & Barab, S. (2012). Using Digital Games to Learn in and out of School: What Makes a Game a Learning Game? In Proceedings of the 6th European Conference on Games Based Learning (Vol. 1, p. 658). Academic Conferences International Limited.

Sterman, J.D. (2000). *Business dynamics, system thinking and modeling for a complex world*. Irwin McGraw-Hill, New York.

Sverdrup H. (Ed.), Haraldsson, H., Olafsdottir, A.H., Belyazid, S., Svensson, M., & Nordby, A. (2022). *System thinking, system analysis and system dynamics: The methods used to find out how the world works and then simulate what would* *happen*. 6th revised edition. Hamar, Norway. Oplanske Bokforlag ISBN 978-82-7518-281-2.

Sverdrup, H., & Svensson, M. (2002). *Defining sustainability. In: Developing principles for sustainable forestry: Results from a research program in southern Sweden*. Sverdrup, H. and Stjernquist, I. (Eds.) Managing Forest Ecosystems 5:21-32 Kluwer Academic Publishers, Amsterdam.

Sverdrup, H., & Svensson, M. (2004). Defining the concept of sustainability, a matter of systems analysis. In: M. Olsson, & G. Sjöstedt (Eds.); *Revealing complex structures – Challenges for Swedish systems analysis*, 122-142. Kluwer Academic Publishers.

Sverdrup H. (Ed.), Haraldsson, H., Koca, D., & Belyazid, S. (2014). *System thinking, system analysis and system dynamics: Modelling procedures for communicating insight*

Sverdrup H. (Ed.), Haraldsson, H., Olafsdottir, A.H., Belyazid, S., & Svensson, M. (2019). *System thinking, system analysis and system dynamics: Find out how the world works and then simulate what would happen*. 3rd revised edition. Háskolaprent Reykjavik. 310 pages. ISBN 978-9935-24-425-3.

Tapscott, D. (2008). *Grown up digital: How the net generation is changing your world.* New York: McGraw-Hill.

Tapscott, D. (2011). The net generation is the smartest generation. *Interview Daily Education online database.* (2010).

Teknibas, K.S., Gresalfi, M.,Peppler, K., & Santo,R.(2014*). Gaming the system: Designing with Gamestar mechanic.* London: The MIT Press.

UDIR. (2014). http://data.udir.no/kl06/SAF1-03.pdf?lang=http://data.udir.no /kl06/ eng

Vargo, S.L., & Lusch, R.F. (2011). It's all B2B. . . and beyond: Toward a systems perspective of the market. *Industrial Marketing Management*, Vol. 40, pp. 181-7.

Vargo, S.L., Lusch, R.F., Akaka, M.A., & Yi, H. (2010), The service-dominant logic of marketing: A review and assessment. *Review of Marketing Research*, Vol. 6, pp. 125-67.

Vargo, S.L, & Lusch, R.F. (2004). Evolving to a new dominant logic for marketing. *Journal of Marketing*, *68*(January), 1-17.

Vargo, S.L, & Lusch, R.F (2016). Institutions and axioms: An extension and update of service-dominant logic. *Journal of the Academy of Marketing Science*, 1-19.

Vennix, J.A.M. (1996). *Group model building*. New York, Wiley.

Vennix, J.A.M. (1999). Group model-building: tackling messy problems. *System Dynamics Review* 15(4): 379-401.

Vennix, J.A.M., & Andersen, D., et al. (1992). Model-building for group decision support: Issues and alternatives in knowledge elicitation. *European Journal of Operational Research* 59(1): 14-28.

Wall Street Journal. (2022). Jane McGonigal Believes Games Can Change the World. https://www.wsj.com/articles/jane-mcgonigal-believes-games-can-change-the-world-11647620651

Wallach, O. (2019). Who Builds Video Game Wikis? Retrieved from https://www.fanbyte.com/features/who-builds-video-game-wikis/

Wang, Z., Cherrier, H., & Zhou, W. (2004). Go and Gomoku on the web. *Artificial Intelligence*, *157*(1-2), 343-358.

Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

Wenger, E. (2006). *Learning for a small planet: A research agenda.* Retrieved from: http://learning-affordances.wikispaces.com/file/detail/Learning+for+a+small+planet.doc

Werbach. (2012). Video series on Gamification. Coursera Videos: https://www.youtube.com/watch?v=rzihbzxC_6Y

WePC. (2023). www.wepc.com. Refers to sources like Statista, NewZoo, Juniper Research 2021, Statista / DFC Intelligence; 2021

Wu, C. (1982). The Taijitu in ancient Chinese thought. *Journal of Chinese Philosophy*, *9*(4), 401-41.

Dissertation articles

1

Article 1:
Teaching ‘hardcore science’ to arts and design students: Reflections on the development of a basic programming course

Anders Nordby & Sidsel Karlsen (2014)

InFormation, Nordic Journal of Art and Research

InFormation

Nordic Journal of Art and Research

ISSN: 1893-2479 www.artandresearch.info

Teaching 'hardcore science' to arts and design students: Reflections on the development of a basic programming course

Anders Nordby & Sidsel Karlsen[1]

Hedmark University College

Abstract: This article reports on the longitudinal development of a computer-programming course designed to meet the needs of students who enrolled in a specific higher education game development programme during the period of 2006 to 2010. The students came from three different arts and design-related strands of the programme, and had that in common that very few had taken advanced science classes as part of their upper-secondary education. This again meant that they were rather poorly equipped for learning computer programming, which they needed to master in order to tap into the full potential of the interactive and creative processes which their computers allowed for. Consequently, the programming course was designed in a way that allowed the students to practically engage in creating a computer game alongside being taught the actual programming skills, as well as the mathematics and physics needed in order to efficiently utilise those skills. A working hypothesis for the project was that if the responsible teachers were able to run the course in a way that cohered with the principles of problem-based learning, this would create an environment which would enhance the students' motivation to learn basic programming as well as the operative and innovation skills needed for fulfilling the course requirements. In addition, ideas developed within the field of situated learning constituted theoretical points of departure for developing the course. The article describes the practical and theoretical points of departure for developing the programming course and reflects on the experiences made from running it. Summing up, the authors conclude that the *why* and *how* of teaching needs to be in line with students' worlds in order for educational experiences to be considered as meaningful.

Keywords: arts and design students; computer programming; gamification; problem-based learning; situated learning.

Introduction: The gaming generation

During the past 20 years, videogames have become increasingly popular, in fact to such an extent that, in 2010, as many as 65 percent of all American households admitted to its members playing such

[1] Faculty of Education and Natural Sciences, Hedmark University College, Box 400, N-2418 Elverum. E-mail: anders.nordby@hihm.no, sidsel.karlsen@hihm.no.

games (Education Database Online, 2010). Online games and communities are widely played and extensively used by young people, and to a rapidly expanding degree. This same generation of people enters our educational systems with the experiences and knowledge they have gained through playing. In this article, we take this phenomenon as a point of departure for investigating how students' gaming backgrounds can be taken into account when designing higher education. More specifically, we look into the longitudinal development of a specific course on computer programming designed to meet the needs of the arts and design students who enrolled in the Game Development Programme at Hedmark University College during the period 2006 to 2010.

So, if anything, what characterises the young gamers? Interviewed about his book on the net generation (see Tapscott, 2008). Tapscott elaborates on what the younger internet users demand from life, education and future employment, and claims that the concept of *fun* is crucial to their participation and motivation: "They want to have fun. In fact, 58 percent of them say that having fun with a product or service is just as important as what that thing actually does. If you employ any of these people, realize that they also want to have fun at work. They want to collaborate and have relationships. They want innovation and creativity. They want speed. They want to customize everything … this group wants to do things their own way" (Davis, 2008). When the members of the gaming generation enrol in higher education they may have similar needs and wishes without those necessarily being met, among other things because educations and courses are designed in quite traditional ways.

Gaming and the aspect of learning

Needless to say, the facts and claims exhibited above have not gone unnoticed by researchers and learning designers. For example, Gee (2004, 2007, 2010b), Prensky (2011) and Shaffer (2007) argue that well-developed games may be used as learning environments. Also, they point out that gamers normally learn in inductive ways, by doing, and not deductively by reading the written manual before engaging in the game. Gee (2010b) describes how he himself tried to do the latter, but found himself sadly unequipped for the actual playing. Then, he tried to do it the other way around – playing the game for a while and then reading the manual – with the consequence that the manual appeared as both lucid and instructive. In other ways, he engaged in the game as the average underage gamer does: he started to play the game and then moved on to the written instructions whenever that was necessary for solving particular challenges that occurred along the road. Such an approach might be labelled *learning by doing* (Dewey, 1916) or *learning just in time* (Gee, 2007).

Video or internet-based games are typically multimodal. The player's visual environments consist of movable graphics – often presented as splendidly and meticulously painted interior or exterior views – as well as an animated character that represents the player and which can often be transformed, to some extent, in accordance with the player's personal preferences. In addition, the environment has audible features, such as a more general soundscape and particular sounds and musics that trigger and underpin the player's actions and experiences. Designed well, these features enable the player to immerse herself completely in the game and seemingly forget the fact that she is engaging in tasks that require large amounts of focus and actual learning. As Vangsnes (2009) reminds us, this immersion often also involves role-play, which may enhance the gamer's experiences of learning in additional ways.

The player's learning may be understood as cross-disciplinary as well as multimodal. Since games often cover many topics, the player will learn them all at once and in an integrated fashion. Moreover, games offer continuous and immediate assessment, which contribute to the learning, and with the evaluation procedures built in as part of the playing. In a keynote speech at the Meaningful Play 2010 conference, Gee (2010a) puts it this way: "All a video game is, is a set of problems that you must solve in order to win. The game itself does assessment all the time. No test is necessary when the game is completed to check whether the gamer has learned, it is proven throughout the game that he has."

Pearce (2011) points to some further key elements concerning games and learning. First, every good game has a clear goal and provides the tools necessary for reaching that goal. Since an easy game is considered boring, obstacles and challenges – and their related areas of learning – are obligatory components if a game should be perceived as 'fun to play'. Second, online games provide fruitful environments for collaboration, peer learning and competition. Third, as Kapp (2012) has shown, because most games provide different levels customized for players with various degrees of competence, the games may also offer personalized learning experiences and environments in which the player has the freedom to take chances and to fail without this having severe consequences for playing the game as such.

Recognising that games most often encompass features that require learning in one form or the other, some educationalists have sought to make use of these characteristics and develop educational games (Egenfeldt-Nielsen, Smith & Tosca, 2013) out of what traditionally have been thought of as 'boring topics'. This approach, sometimes named *gamification* [1], is defined as "using game-based mechanics, aesthetics and game thinking to engage people, motivate action, promote learning and solve problems" (The Learning Circuits Blog, 2012), and may be introduced in traditionally non-gaming environments, such as schools and universities, in order to create dynamic and competitive learning environments for students. Since most students nowadays come with game-related experiences of 'doing and learning before reading', which in many ways contradict the traditional approach to knowledge acquisition found within schools, aiming to transfer some of the learning-enhancing features of games into the educational sphere seems like a fruitful approach in order to strengthen students' motivation and build on their already acquired learning strategies.

With this knowledge about the gaming generation and its members' potential learning experiences in mind, the aforementioned computer programming course at Hedmark University College was developed so that it would better fit the students' needs and correspond more efficiently with their previously acquired skills and knowledge. Most of the students who enrolled in the course belonged to the gaming generation and had, as such, many years' experience of playing in video- or online-based multimodal, collaborative and fun learning environments. However, when it came to *creating* such or similar environments and artworks, and thereby tapping into the full potential for artistic and interactive processes that their computer-based tools allowed for, the students were sorely lacking in relevant knowledge and skills related to computer programming, as well as in knowledge of advanced mathematics and physics needed to efficiently perform such programming. Moreover, they did not even consider it interesting or important to learn any of these 'hardcore' topics. Describing how the computer-programming course developers accommodated to the situation at hand, and thereby laying out the theoretical framework, rationale and workings of this specific course, the following questions came to guide our explorations: How can the 'fun' experienced in gameplaying, modding, gamification and online communities be brought into so-called 'boring' topics in higher education? How can gameplaying and game development be used in order to teach 'hardcore' subjects like programming (and its related mathematics and physics) to art and design students in a university course on computer programming? How can problem solving, collaboration, learning by doing, learning just in time and innovation be implemented as part of such a course?

Gamification in education – a brief overview

Using games as a part of formal schooling is nothing new; in fact games have been employed in education generally as well as in the teaching of natural sciences specifically for quite a while, and gamification has even been put to work in courses on basic programming in other universities. A well-known and still ongoing project in this respect is IBM's *Robocode* (Nelson, 2011; see also Long, 2007) in which the students create 'intelligent' tanks which have only one purpose, namely to beat other tanks and win the game. This particular game is designed to teach students the basics of natural science phenomena and presents them with challenges related to, for example, physics, mathematics and artificial intelligence. Each time the students change the programming code or try out a new theory in order to overcome the obstacles created by other tanks, they receive immediate visual feedback from observing the alterations in their own tank's movement and capabilities. Another relevant example is a Finnish game (Rajaravivarma, 2005) created with the purpose of enhancing students' learning of special programming skills through the programming of simple numbers and wordplays. Furthermore, a Danish game-based environment exists (see Andersen, Bennedsen, Brandorff, Caspersen & Mosegaard, 2003), which is formed to teach multimedia students with an artistic background basic computer programming skills. With respect to the informal features of web-based learning, Partti and Karlsen (2010) show how these are played out in an online music community, with a special emphasis on peer learning and reviewing, sharing of information and learning through experimentation. They also discuss how such internet-related learning strategies can be implemented as part of compulsory music education (see also Partti & Westerlund, 2013). Despite the subject areas being vastly different, we find these contributions to be highly relevant and adaptable to a course on computer programming, encouraging the students to learn as much as possible from each other instead of providing a traditional teacher-led learning environment.

Theoretical points of departure for developing the course

On the basis of the above, a working hypothesis for the project of developing the course in computer programming at Hedmark University College was that if the teachers responsible for the course were to be able to run it in a way that combined student tasks of actual game development with teaching and facilitating founded on the principles of problem-based learning (PBL), this would create a working environment which would enhance the students' motivation to learn basic programming as well as their operative and innovation skills in this particular area. Furthermore, their learning was expected to proceed more efficiently in such an environment than was the case when the course was run using more traditional teaching methods, such as lectures. In addition to the game development tasks and the PBL teaching methodology, another prerequisite of the course was that it should incorporate more informal online activities such as creating web pages and blogs and participating in online communities and forums into the formal sphere of the university context. The aim of this approach was to maintain a sense of 'fun' as well as linking students' outside-of-school experiences to their educational reality. Theoretically, the course design was informed and inspired by the already mentioned PBL-approach as well as by ideas found within sociocultural theories of situated learning (Lave & Wenger, 1991; Wenger, 1998). In the following we will aim to explicate these theoretical frameworks and their contributions to the course design.

Problem-based learning

The concept of problem-based learning (PBL) originates from medical school, and was first implemented in the late 1960s by Howard Barrows and his colleagues (Barrows, 1980). Originally, it was a model for facilitating student-centred learning in small groups, ideally six to 10 persons, led by a tutor whose task it was to guide the students, more so than to teach them, and the focus of the group was to solve real-world problems related to the professional world which the students aimed to enter. Since its beginning in the 1960s, PBL has established itself as an independent pedagogical concept or tool in higher education. Leaning on Pettersen (2005; see also Dochy, Segers, van den Bossche & Gijbels, 2003) the present-day understanding and principles of PBL could be described as follows:

1. The problems build on practice-related descriptions, case reports and case studies
2. Students – both individually and in groups – receive support, assistance and monitoring from a tutor
3. Students will engage in learning activities and develop ways of working and learning as well as strategies for studying along with practical problem solving and logical reasoning
4. Students have autonomy of their learning and study activities and emphasis is put on self-regulation and student autonomy
5. The teaching, the curriculum and the actual courses are organized in ways that allow and maintain cross- and interdisciplinary approaches
6. The study programme that the students follow facilitates their early contact with future and possible clients/users, and authentic tasks and challenges are given which qualify for the students' future professional lives

In addition to following the above principles, a teacher planning a course adhering to the PBL model will normally facilitate for the students to follow these seven steps (Maastricht University, 2013):

1. Clarify difficult terms
2. Define the problem
3. Brainstorm
4. Take stock of and analyze the possible solutions provided in step three
5. Formulate learning objectives
6. Self-study
7. A post-discussion

Step one to five points to the initial phase where the tutored groups of students define the learning goals of the given case through brainstorming, and identify which theory it is relevant to study for solving the case. Step six represents the implementation phase where the actual work with the case is done, while step seven designates the stage of summing up the results and discussing one's experiences from the work accomplished.

Situated learning

As mentioned above, the field of *situated learning* and the idea that learning is integral to participation in communities of practice (Lave & Wenger, 1991; Wenger, 1998) also informed the design of the programming course. Underlining that their theory is meant as an "analytical viewpoint on learning, a way to understand learning" (Lave & Wenger, 1991, p. 40) and not as a set of prescriptions for how to facilitate learning, not to speak of designing education, the authors still present ideas worthwhile pondering for teachers and university faculties who wish to design learning environments while taking the participatory and interactional aspects of knowledge creation into account. According to Lave and Wenger, learning in a community of practice happens by legitimate peripheral participation, a position that is accessible to each and every one of the community's members simply by them "being located in the social world" (p. 36) which the community constitutes and affords, and partaking in its ongoing activities. However, just grouping people together does not automatically form a community of practice. One of its core characteristics is the "mutual engagement of [its] participants" (Wenger, 1998, p.73) centred around a joint enterprise and exhibiting a shared repertoire of, for example, actions, artefacts, stories and tools. Listing in all 14 indicators that a community of practice has been formed, Wenger emphasises, among other things, "sustained mutual relationships – harmonious and conflictual … shared ways of engaging in doing things together … knowing what others know, what they can do, and how they can contribute to an enterprise … mutually defining identities … specific tools, representations, and other artifacts [and] local lore, shared stories, inside jokes, knowing laughter" (p. 125). Since learning is considered to happen in and through shared and concrete undertakings, it can be accessed (and even facilitated) by considering the relations between the newcomers and the old-timers of a particular community as well as between its range of activities, identities and artefacts. The key to participation for a newcomer is to gain access to the community of practice in question "and all that membership entails" (p. 100), and while there exists no such thing as a centre or core of a learning community, prolonged membership and vivid participation may lead to the more mature stage of "*full participation*" (p. 37, italics in original). In the particular context of the programming course, the tutors employed (see below) and those of the students who had previous experience of creating computer games would typically act as more full members than the rest of the students – at least to begin with – and those with less experience would start the work from the newcomer's position.

For educators who wish that their students should grow and reach the full member-level of proficiency in relation to, for example, participation in the community of game designers and developers, it is crucial to take into account the intimate connections between practice, community, identity and meaning and acknowledge learning, not first and foremost as acquisition of knowledge, but as doing, belonging, becoming, and as experience (Wenger, 1998, p. 5). If knowledge is seen as "a matter of competence with respect to valued enterprises" (p. 4) – in this case the enterprise of creating computer games – information about what kinds of knowledge and skills are valued *within* this specific enterprise or profession is crucial for meaningful participation and future success. Moreover, considering Wenger's notion of *trajectories*, which addresses how participants move between, participate in and learn "both within and across communities of practice" (p. 154) as well as how the identity-work of "constant becoming" (p. 154) is key to forming such paths of learning, designing a course which allows the students to bring in their outside-of-university activities and identities, such as those of bloggers, gamers and participants of online communities, will most likely add to the meaningfulness and the feeling of belonging.

The design: Rationale and run-through of the basic computer-programming course

The course in basic computer programming at the Game Development Programme at Hedmark University College was implemented as a one-semester course, starting 2006. The first year served as a trial year, in which all the practical aspects of the course were tested and evaluated; for example which software [2] and game types to use, what kind of cases to present the students with, and how to mix the PBL-aspects of the course with lectures and lab work in efficient ways.

The students attending the course over the period covered in this article (2006 to 2010) mainly came from three different strands of the overarching programme in game development, namely from Virtual Arts and Design, Animation as well as Visual Simulation. They were all in their first year of studies. In addition, third- or fourth-year teacher education students for whom the course was optional and offered as one of several available specializations, would occasionally enrol. Typically, around 70 students attended the course each year, their age ranging from 18 to 25 years, mainly. As such, the students were typical examples of the gaming generation described in the beginning of this article.

The main reason for teaching computer programming to art and design students was, as mentioned above, to enable them to tap into the full potential for interactive creative processes which their computers allowed for, and which were seen as fruitful for the kind of artistic work that they engaged in. A big challenge, however, was that the students did not consider it interesting or beneficial to learn programming or any natural science topics for that matter. Very few of the students had taken advanced science classes as part of their upper-secondary education, which meant that the mathematics and physics needed in order to create a functioning game had to be learned alongside the other elements of the course. Although the main goal was to teach the students computer programming, we also wanted the students to experience why an understanding of certain 'hard science' topics was needed in digital arts education for them to be able to create the advanced computer-based artworks that they aimed for.

Before the implementation of the game-creating approach started in 2006, programming, mathematics and physics had been taught in more traditional ways. However, the general examination statistics of the university college showed that approximately 50 percent of the students failed, and consequent informal talks with the students indicated quite a lot of frustration with the teaching methods employed. Hence, new ideas for the design of this particular part of the students' education were needed. Gamification was much approved as an educational motivator around the time when the basic computer-programming course was first designed, and a decision was made to use this as one possible road of teaching the necessary skills. As mentioned above, during the trial year many different game types were tested until the course developers landed on a platform game which seemed to contain all the elements and challenges needed to justify the teaching of what the staff knew would be the skills and knowledge about basic programming, mathematics and physics that would be required in the students' future professional life.

Apart from some basic elements of a platform game, such as a main character that could be controlled from the keyboard and who should be able to walk, run and jump, the students were given very few specific guidelines or restrictions on how to design the game. The course developers implemented the PBL approach in a similar way to what had previously been done in Finland (see Kinnunen & Malmi, 2005; Nuutila, Törmä & Malmi, 2005). The students were divided into groups in which they were to work in collaboration throughout the course. Mostly, but not always, the groups were formed according to suggestions made by the students themselves about who they would like to work with. Within the frames of these groups each individual student was responsible for creating and completing his or her own game. In line with the PBL thinking, each group had their own tutor, which would typically be higher-grade students employed for this job specifically and with the necessary experience. Although the course was built around specific cases and tasks to be solved, lectures and guided lab work were not entirely abandoned as a way to facilitate the students' learning. Starting the course, short lectures were given on how to approach the PBL working method as well as the cases, and in order to prevent the students from becoming stuck while working on their projects, guided lab work was implemented in-between the case periods. Following the PBL model, the game-development project was divided into smaller tasks or steps that were presented as problem-based cases. While the actual number of cases would vary somewhat from year to year, a typical order of tasks would be:

1. Design the game and make a storyboard
2. Design the characteristics of the main character of your game and then make a drawing of this character
3. Divide the character into layers, animate its walking cycles and jumping abilities
4. Connect the triggering of the walking cycles to the keyboard arrow keys [3]
5. Draw the ground of the game and make the main character follow this ground
6. Make a second character (an enemy) and let it follow the main character
7. Implement jump for the main character and design platforms to jump on
8. Make a starting screen with the necessary menus as a well as screens to be shown when winning or losing the game [4]

The cases were deliberately designed as open-ended so that eager students could develop them beyond the basic requirements without proceeding to the next case and thereby moving too fast through the planned steps of the course.

During the last part of the course, the student groups used Wiki pages to present their individual projects. In addition, each student wrote a blog after the completion of every step or case about how the game they had chosen to create progressed. The blog tasks were obligatory and they were also part of the final report that all the students submitted at the end of the course. Chat tools and an internet-based forum were used throughout the project to facilitate collaboration among the group-members in-between physical meetings. This way, the students had the chance to develop both place-based and virtual communities of practice (Barab, Kling & Gray, 2004), something that was believed to enhance their overall experiences of learning.

Most of the Wiki pages and blogs were written in English so that the students could have feedback from the international community of gamers and game makers. The students were also encouraged and enabled to seek for online game communities in which they could participate, study and play games similar to the ones they aimed to create themselves as well as present their own projects, including their technical and artistic solutions, and discuss them with the other community members. Finally, the students were told to play their own games as they developed and in this way explore the 'play' aspects embedded in them as well as reflect on the emergence of the games and how they could be made more immersive.

An absolutely necessary tool throughout the course was each student's own PC and its accompanying software. To enable all the students both to create the games as well as play them later on without too many technical difficulties, they were offered a beneficial leasing deal that gave them access to powerful computers and upgraded software.

Experiences from running the course: Practical solutions and theoretical afterthoughts

In this section we aim to describe some experiences gained from running the computer-programming course as well as relate them to our theoretical points of departure as laid out above. Our discussion is based on the fact that one of us – Anders Nordby – was responsible for developing, implementing and running the actual course over the period 2006 to 2010. Furthermore, although this is not an empirical article, but rather a theorized description of a longitudinal course-development process, what follows is also partly based on the experiences of reading through a large amount of feedback material collected from the students and the tutors who participated in the course over the given time frame [5].

First of all, it quickly became evident when following and evaluating the course that 'making a game' in fact came to constitute a common goal for most of the participating students, despite the fact that they arrived with vastly different backgrounds and were enrolled in several and quite different study programmes. As such, the entire class, and perhaps even more so the student groups, made up vibrant communities of practice (Lave & Wenger, 1991; Wenger, 1998) which engaged in this task as their primary joint enterprise. Hence, the gamification approach was largely successful in that it provided the students with a collaborative social environment in which they could participate according to their level of knowledge and skills – in other words, more-or-less 'peripheral' or 'full' (Lave & Wenger, 1991, p. 37) – and where the newcomers to the field of game-making could learn from those considered more experienced, whether that be other students or the tutors. It is of course hard to tell the exact reasons for the student groups actually developing into communities of practice (as can be seen above, this can by no means be expected to happen automatically just by grouping students together). However, one fairly qualified guess would be that this development was afforded by the fact that the students – as a general rule – were able to choose their own collaborators. Thereby they could also select persons with whom they shared other features crucial for the formation of a community, for example they could know approximately "what they [could] do, and how they [could] contribute" (p. 125) and they would probably also have "shared stories" (p. 125) already prior to embarking on the making of the games. Another potentially community-developing factor could be that the students in fact did share quite a lot of "specific tools, representations, and other artifacts" (p. 125) given to them during the run of the course, and in order to complete the given task they had to figure out how to utilise these resources *together*, thereby strengthening the mutual commitment and engagement of the groups. The fact that the task was steered towards actually *making a game* and not towards learning *how to make a game* (ref. Folkestad, 2006) [6] also seemed to enhance the students' motivation towards solving the task as well as give them insights not just into the particular and minuscule operations and pieces of knowledge needed for completing it; they also came to see how these 'fragments' belonged to the bigger picture. Furthermore, they developed an overview of how the specific topics taught all fed into creating a larger academic, artistic and practical context to which they needed to relate in order to complete their game in a successful way.

The social environment and communities of practice created through the course also successfully provided a 'learning by doing' (Dewey, 1916) context, largely through the utilization of the PBL approach, which forced the students to learn in an integrated way since the theory they had to seek out was always connected to practical tasks and operations embedded in the problem-based cases. As such, their theoretical learning always happened 'just in time' (Gee, 2007) for them to solve actual problems, and not in a row of lectures preceding the application of the factual knowledge. As examples of practical, game-related tasks given that effectively led to students developing knowledge in hardcore physics and mathematics, we here choose to emphasize two.

First, when the students had finished their game characters and started the process of animating them, one of the features they needed to add was giving their characters the ability to jump. For this to happen, they first needed to apply a force that allowed an upward movement. However, without making any modifications to that force, the character would simply disappear out of the top of the screen. In other words, this was the right time to learn about gravity, since applying a gravitating force would move the character back to the ground. Still, not just any kind of gravitation would do, since the aim was to make the character jump in a credible and convincing way with body-movements similar to those a jumping human or animal would exhibit in real life. Faced with this situation, very few students found it odd to read and talk about physics and gravity, and they usually realized quite quickly that they needed a solid theoretical foundation for understanding the forces and effects of gravity in order to be able to implement a well-working solution to the task at hand.

Second, when reaching the stage of the game development in which they should implement their characters' walking cycles and moves around the screen, the students needed to expand their knowledge about x- and y-axes, coordinates, angles, and conversion between degrees and radians. Eventually, they also had to learn trigonometry to be able to divide the movements into components. Usually, this is a kind of theory that students struggle to understand if it is taught separately and on an abstract level; however faced with practical problems to solve in order to make their games work, the rather complex physics and mathematics once again became comprehensible because it was to be applied to something very concrete and something which it mattered greatly to the students to be able to complete. In other words, the 'fun' (Tapscott, 2008) and 'motivation' aspects were evidently present all along the course.

As mentioned above, through the gamification and PBL-based approach, the students were provided with one or more communities of practice in which they could develop their knowledge and skills. Not just the different modes of participation or the fact that the focus was on actually making a game contributed to this learning environment. So too did the composition of the working groups, the access to 'qualified helpers' as well as a wide-ranging (and explicitly encouraged throughout the course) contact with other, relevant communities of practice composed of people with similar interests, knowledge and skills. The latter opportunity enabled the students to develop *boundary trajectories* and thereby "linking communities of practice" (p. 154) as well as experiences and knowledge acquired in different learning contexts.

In order to accommodate groups who could work together creatively, the teachers sometimes interfered with the group composition so as to make them as heterogeneous as possible (Darsø, 2000; Gee, 2007). After a trial period in which different group sizes were tried out, the groups stabilized on eight to 10 students, which seemed to ease many of the organizational aspects. At the same time, this group size correlates with what the theories suggest both when it comes to maximising of creativity (Darsø, 2001) and PBL learning (Pettersen, 2005). The heterogeneity approach encompassed variables such as study programme, interests, gender and age. In addition to the existing knowledge and skills within each student group, the groups were, as already mentioned, provided with a tutor, who represented a very significant 'old-timer' (Lave & Wenger, 1991) with respect to the actual game task. The tutors were carefully selected among the best students from previous years, and all of them put much effort and energy into their work.

Related to the already-mentioned concept of trajectories, Wenger (1998) also emphasizes the necessity for communities of practice to be engaged with similar communities (see above) in order to expand their practice-related knowledge, and coins the notion of *brokers* to denote individuals who "make new connections across communities of practice, enable coordination, and – if they are good brokers – open new possibilities for meaning" (p. 109). Likewise, he reminds us that people may have (or aspire to) community "*multimembership*" (p. 149) as well as a need to negotiate "local ways of belonging [and doing] to broader constellations and of manifesting broader styles and discourses" (p. 149). In the course development described in this article – and especially during the last two years of the period attended to – these aspects were facilitated and nurtured through the students being encouraged to create web pages, blogs and promote their ideas and games in online communities and on web-based forums in which gaming and game development constituted the core point of interest for the members. Those of the students who belonged to such constellations beforehand, then had an opportunity to enact their multimembership and act as brokers between the university college-based game-creating communities and the web-based ditto. Furthermore, these outwardly directed "boundary trajectories" (p. 154) provided the students with valuable input and information from outsiders who were connected to international discourses on gaming. The input could range from practical and concrete help with getting the technical details right to broader discussions about what constituted a good, playable and interesting game from a gamer's perspective. The web pages and blogs also gave the students an arena for constructing and displaying – in a very visible way – their (often) newfound identities as game-developers [7]. Hence, these tools integrated the practice-, trajectory- and identity-aspects of situated learning in a very fruitful way, and might have contributed to the students experiencing their learning both as "belonging," "doing" and "becoming" (see Wenger, 1998, p. 5).

Concluding remarks

In the beginning of this article, we asked, among other things, how gameplaying and game development could be used in order to teach 'hardcore' subjects like programming, mathematics and physics to art and design students in a university course on computer programming. Furthermore, we wondered how problem solving, collaboration, learning by doing, learning just in time and innovation could be implemented as part of such a course. Hopefully, the previous sections have provided the reader with some tentative answers and theorized practical solutions to these questions. Going back to Wenger's (1998) notion of trajectories, not all students enrolled in the programming course remained in the game-developer sphere. Rather, some of them entered "outbound trajectories" (p. 155) and chose to implement the knowledge and skills gained through the course in other parts of the world of virtual art and design. Other students, however, remained on the inbound trajectories afforded by the course and chose to even enter "insider trajectories" (p. 154), further participating in "new events, [meeting] new demands, [contributing] new inventions" (p. 154) and continued to renegotiate their identities as game-developers. Some of these students have since embarked on a course as professionals within the gaming industry (see for example Krillbite, n.d.; Moondrop, n.d.; Roofmoose, n.d.).

One of the main challenges of running the programming course is of course to be able to keep up with the rapid development of the field and to continue to provide new gaming and net generations (Tapscott, 2008) of students with fun and motivational educational experiences that correspond with *their* virtual and real-world realities. As such, this article is meant as a contribution to keeping the field afloat, showing that it is not just *what* you teach that matters. The *why* and, not least, the *how* of teaching also needs to be in line with students' worlds in order for them to consider their educational experiences as meaningful.

Endnotes

[1] While we have chosen to relate to a definition of gamification that specifically incorporates the element of learning, we are well aware that differing and even competing definitions of the term exist. For example, in Deterding, Dixon, Khaled and Nacke's (2011) investigations into the historical origins of the term, gamification is taken to refer to the use of game design elements in non-game contexts more generally.

[2] An important premise for choosing the software in which the game should be developed was that the students should be able to have a visual representation of all programming actions at any given time. Consequently, Adobe® Flash® was chosen since it allows the developer to carry out the object-oriented programming, the graphic design, the animation and the implementation of the game soundscape using only one tool, and at the same it provides an environment where the results of all actions can be viewed immediately. This was considered crucial for the work in the PBL groups (see below), where each session started with the students showing each other how their particular game had developed since their last meeting and commenting on each other's work. Furthermore the immediate visual and graphic feedback of the programming was seen as a way to connect the more theoretically oriented programming work to the course participants' identities as students of art and design.

[3] The actual programming tasks started here; this fourth case would typically last double the length of the three preceding ones.

[4] Additional cases nine and 10 were sometimes used, typically involving more advanced programming options, like the implementation of several game levels, more characters and more advanced character functionality et cetera.

[5] As part of a bigger action research project, a vast amount of quantitative and qualitative data has been collected which concerns the students' experiences of participation and learning within the frames of this particular course. Among other things, the data includes students' and tutors' reports after completing each case; students' reports after finishing the course; blogs created by the students during the run of the course; and a questionnaire distributed among all the students who participated in the course during the period of 2006-2010. While the findings from this study are not the concern of this article, the task of reading through and analyzing the student data forms an additional interpretative backdrop for our reflections in this section. Since the research was carried out within the frames of the researchers' own institution and among students which one of the researchers also met in the capacities of teacher and supervisor, the project was conducted with a special awareness of the ethical concerns that come with such situations (Brydon-Miller, 2008).

[6] In an article on formal and informal learning situations, practices and ways of learning related to music, Folkestad (2006) makes a distinction between *learning how to play music* versus *playing music*, claiming that the latter constitutes an important part of an informal learning situation or practice, and is often also imbued with a lot more ownership and agency on the learner's side than the former, more formally oriented approach. We believe this to be true not only for learning situations involving music.

[7] Similarly, Partti and Karlsen (2010) show how online music communities may provide an arena for people to construct and maintain their musical identities as for example musicians and composers.

On the contributors

Anders Nordby is educated in electronics, computer science and pedagogy, and he currently works as an assistant professor at Hedmark University College at IgA, a section in the Department of Fine Arts and Computer Science which focuses on interactivity, gamification and arts. His work is mainly focused towards games and learning, system thinking and sustainability, and recent writings centre on how gamification might be utilised in order to teach systems science and sustainability. Nordby teaches typical game design topics, such as general game design, 3D design, animation, sound production and programming, as well as pedagogy.

Sidsel Karlsen is educated as a music educator and classical singer and she currently works as professor of music education at Hedmark University College in Norway as well as docent at the University of the Arts Helsinki (Sibelius Academy) in Finland. She has published widely in Scandinavian and international research journals and has contributed to a number of international anthologies, such as *Sociology and music education* and *Collaborative learning in higher music education*. Her research interests span, among other things, the field of multicultural music education, the relations between music, identity, learning and new media, as well as the social and cultural significance of music festivals.

References

Andersen, P. B., Bennedsen, J., Brandorff, S., Caspersen, M. E., Mosegaard, J. (2003). Teaching programming to liberal arts students – a narrative media approach. *ACM SIGCSE Bulletin – Proceedings of the 8th Annual Conference on Innovation and Technology in Computer Science Education*, *35*(3), 109-113.

Barab, S. A., Kling, R., & Gray, J. H. (2004). *Designing for virtual communities in the service of learning.* Cambridge: Cambridge University Press.

Barrows, H. S., & Tamblyn, R. (1980). *Problem-based learning: An approach to medical education.* New York: Springer.

Brydon-Miller, M. (2008). Ethics and action research: Deepening our commitment to principles of social justice and redefining systems of democratic practice. In Reason, P., & Bradbury, H. (Eds.), *The SAGE handbook of action research* (pp. 199-210). London: Sage Publications.

Darsø, L. (2001). *Innovation in the making.* Frederiksberg: Samfundslitteratur.

Davis, E. (2008). *Don Tapscott discusses the "net" generation.* Retrieved from http://blog.shop.org/2008/09/16/don-tapscott-discusses-the-net-generation/

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining "gamification". *MindTrek '11. Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments* (pp. 9-15).

Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

Dochy, F., Segers, M., van den Bossche, P., & Gijbels, D. (2003). Effects of problem based learning: A meta-analysis. *Learning and Instruction*, *13*(5), 533-568.

Education Database Online. (2010). *American households playing games*. Retrieved from http://www.onlineeducation.net/videogame

Egenfeldt-Nielsen, S., Smith, J. H., Tosca, S. P. (2013). *Understanding video games. The essential introduction* (2nd ed.). New York: Routledge.

Folkestad, G. (2006). Formal and informal situations or practices *vs* formal and informal ways of learning. *British Journal of Music Education*, *23*(2), 135-145.

Gee, J. P. (2004). *Situated language and learning: A critique of traditional schooling*. London: Routledge.

Gee, J. P. (2007). *Good video games and good learning: Collected essays on video games, learning and literacy.* New York: Peter Lang International Academic Publishers.

Gee, J. P. (2010a) Design, learning, and experience [Video file]. Retrieved from http://vimeo.com/16254768

Gee, J. P. (2010b). *New digital media and learning as an emerging area and "worked examples" as one way forward*. Cambridge, MA: MIT Press.

Kapp, K. M. (2012). *The gamification of learning and instruction: Game-based methods and strategies for training and education*. San Francisco: John Wiley & Sons.

Kinnunen, P., & Malmi, L. (2005). Problems in problem-based learning – experiences, analysis and lessons learned on an introductory programming course. *Informatics in Education*, *4*(2), 193-214.

Krillbite. (n.d.). *Current projects.* Retrieved from http://www.krillbite.com/index.php

Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

Long, J. (2007). Just for fun: Using programming games in software programming training and education – a field study of IBM Robocode Community. *Journal of Information Technology Education*, *6*, 279-290.

Maastricht University. (2013). *Problem based learning preparatory website.* Retrieved from http://www.umpblprep.nl/

Moondrop. (n.d.). *Main page.* Retrieved from http://moondrop.no/

Nelson, M. (2001). *Robocode*. [Java], IBM.

Nuutila, E., Törmä, S., & Malmi, L. (2005). PBL and computer programming – the seven steps method with adaptations. *Computer Science Education*, *15*(2), 123-142.

Partti, H., & Karlsen, S. (2010). Reconceptualising musical learning: new media, identity and community in music education. *Music Education Research, 12*(4), pp. 369-382.

Partti, H., & Westerlund, H. (2013). Envisioning collaborative composing in music education: Learning and negotiation of meaning in operabyyou.com. *British Journal of Music Education*, *30*(2), 207-222.

Pearce, K. (2011). *The 'Gamification' of education.* Retrieved from http://blogs.canoe.ca/canoetech/signs-of-the-times/72001/

Pettersen, R. C. (2005). *Kvalitetslæring i høgere utdanning: Innføring i problem- og praksisbasert didaktikk.* Oslo: Universitetsforlaget.

Prensky, M. (2011). Computer games and learning: Digital game-based learning. In Raessens, J., & Goldstein, J. (Eds.), *Handbook of computer game studies* (pp. 97-122). Cambridge, MA: MIT Press.

Rajaravivarma, R. (2005). A game-based approach for teaching the introductory programming course. *ACM SIGCSE Bulletin*, *37*(4), 98-102.

Roofmoose. (n.d.). *Factsheet.* Retrieved from http://roofmoose.com/press/

Shaffer, D. W. (2007). *How computer games help children learn.* New York: Palgrave.

Tapscott, D. (2008). *Grown up digital: How the net generation is changing your world.* New York: McGraw-Hill.

The Learning Circuits Blog. (2012). *What is gamification? And why it matters to L&D professionals.* Retrieved from http://learningcircuits.blogspot.no/2012/01/what-is-gamification-and-why-it-matters.html

Vangsnes, V. (2009). A dramaturgic perspective: seeing digital role-plays as drama and theatre. What are the epistemological and pedagogical consequences? *Nordic Journal of Digital Literacy*, *1*, 20-34.

Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

2

Article 2:
The art of Gamification; Teaching Sustainability and System Thinking by Pervasive Game Development

Anders Nordby, Kristine Øygardslia, Ulrik Sverdrup and Harald Sverdrup (2016)
The Electronic Journal of e-Learning

The art of Gamification; Teaching Sustainability and System Thinking by Pervasive Game Development

Anders Nordby[1], Kristine Øygardslia[2], Ulrik Sverdrup[3] and Harald Sverdrup[4]
[1]Department of Fine Arts and Computer Science,Hedmark University College, Norway
[2]Institute for digital media and computer science, Norges Tekniske og Naturvitenskaplige Universitet, På Toppen av Haugen, Trondhjem, Norge
[3]Lund University, Sweden
[4]Industrial Engineering, University of Iceland, Iceland
anders.norby@hihm.no
kristine@oygardslia.com
ulrik.sverdrup@gmail.com
hus@hi.is

Abstract: In 2013 Hedmark University College conducted a research project where students from a game development project/study program developed and tested a Pervasive Game for learning as part of a class in System Thinking. The overall game goal was to teach Sustainability through System Thinking, and to give the students a real world experience with their game;. It was tested on 5th and 7th graders in elementary school, spending one school day in each of the classes. This article focuses on the design of the project: how the game was developed, how the children played it and how research was designed and data collected.

Keywords: Gamification, game development, pervasive games, games and learning, pedagogy, system thinking, sustainability

1 Introduction

Making and modifying games have been a large part of children's and young people's lives in recent years due to the many game engines made available to the public for free. During the process of modifying games, and even making totally new games, the children learn about making graphics, animations, programming, game design, system thinking, collaboration, creativity and even pedagogy and psychology when they strive to make their games fun.

Tapscott (2011), the writer of the book on The Net Generation, has some thoughts on what this generation wants. He claims they want to have fun and that 58% of them say that having fun with a product or service is just as important as what that thing actually does. He further claims that if you employ any of these people, you must realize that they also want to have fun at work. They want to collaborate and have relationships. They want innovation and creativity. They want speed. They want to customize everything. Whether it's designing their own t-shirts on Threadless.com, or selecting which widgets to put on their desktop, this group wants to do things their own way.

Of course, these claims and facts have not gone by unnoticed by researchers and learning designers. Researchers like Gee (2004, 2007, 2010) and Schaffer (2008) argue that games can be a powerful learning environment. They point out that when young kids are playing or modifying games while having fun, learning comes naturally, while in traditional schools the children have to study theoretically and accumulate knowledge for a long time before they can practice. Gee describes how he experienced learning in games with his son, and how he tried to read the game manual first and found he didn't understand the game at all. Then he tried the other way around, playing the game first, and then reading the manual. And then everything was very clear. That's what kids do - play first and then read the manual to understand what is necessary to know to solve the game. This is "learning by doing" (Dewey, 1916) and "learning just in time" (Gee, 2007).

Using games for learning is basically about gamification: applying game design and game technology to "serious" learning to bring back the motivation and fun (Deterding et al. 2011). However, although the word "gamification" is quite new, games have always been seen as a useful tool to increase fun and motivation in

Reference this paper as Nordby A, Øygardslia K, Sverdrup U and Sverdrup H, "The art of Gamification; Teaching Sustainability and System Thinking by Pervasive Game Development" *The Electronic Journal of e-Learning Volume 14 Issue 3 2016*, (pp152-168) available online at www.ejel.org

learning. During the last 10 years we have seen rapid advances in digital technology that allow children to interactively "experience" and solve "real world problems" virtually in simulations that contains advanced 3D environments, animations, pictures, film, and sound simultaneously. Furthermore, these environments work in real-time over the Internet and let the children share their experience with other children from all over the world in a way that has never before been possible. By gamifying school topics in these virtual environments we can not only bring back the fun factor and create motivation, but also drill skills, do assessment automatically and create virtual social societies where the children can practice their theory in a fun way (Gee, 2010). Examples of such environments that "accidentally" also work as virtual learning environments are many; for example Minecraft, which teaches young people to design and build 3D environments, Moviestar Planet, which teaches the children how to make movies, how to create new identities or how to handle fame and fortune, or virtual worlds like Club Penquin, Habbo or Stardoll, which teach young children responsibility for their pets, or how to earn and save money. In all these environments the players learn collaboration, teamwork and how to handle peer comments/reviewing from kids all over the world. Today these games and virtual environments have hundreds of millions of users who are from 5-15 years old (KZero, 2011), and they serve as an informal learning environment for children of all ages. Gamification is also about formalizing these informal "learning environments", and using them beneficially in traditional learning and class environments.

System thinking goes hand in hand with game development; games are systems. Through games and system thinking the students and children can learn how events in the real world are connected and influence each other in a causal way. System thinking can also be used to construct games and teach the students how different parts of the game interact and perform. It can even let the students construct and simulate systems that would usually require a deep mathematical understanding of differential equations without knowing the mathematics. In the K12 education in the US system thinking is an important topic; recent research shows that children that are taught System thinking think more critically, express thoughts more clearly and understand more complex problems than children who don't learn system thinking (Lyneis and Stuntz, 2007).

Sustainability is an increasingly important topic in schools, and its cross-disciplinary nature involves many of the important topics in school. Topics about sustainability are therefore usable as "environments" for teachers to embed stories that involve many important school topics at the same time. While it in itself is a competency aim and a goal in the curriculum, it naturally leads both game developers and players into problems that force them to learn mathematics or physics, which for many children are seen as "serious" and "boring" topics.

While developing digital games that incorporate all the above is very time consuming and requires special skills teachers usually don't have, pervasive games is about creating simple games in real-world environments that use common digital tools, internet, pre-made simple games and virtual environments that schools and teachers already have access to in schools. In our project the development of the pervasive game was done by students studying game development, but this type of game is so simple to make that ordinary schoolteachers could equally well have made it. So the project then also served as a learning project for the teachers involved.

2 Scope and Objectives

There are clearly two groups of learners in this project: the students who make the game and the children who play the game. The main goal for the children was to learn sustainability and to understand how system thinking could be used to see how things in the world were connected. The main goal of the game then was to motivate the children to learn the sustainability and system thinking content in the game as well as trigger their interest to learn more about the topics the game presented to them.

The main goal for the students was to learn system thinking, and the game development task and the game playing done with the children was implemented to motivate them in this.

This article mainly focuses on the design of the project: the theory foundation, the game design and the research design and data collected. The article does not contain a thorough data analysis of the learning outcome or the motivation, but has a discussion in the end where we loosely discuss observations and thoughts during the implementation of the project.

2.1 Theoretical points of departure

A working hypothesis for the project was that the combination of game development, in this case pervasive game development, and teaching and facilitating based on the principles of problem-based learning (PBL), would create a working environment which would enhance the students' motivation to learn system thinking and enhance their operative and innovation skills in this particular area. Furthermore, their learning was expected to proceed more efficiently in such an environment than in traditional teaching environments, such as lectures. In addition to the game development tasks, another prerequisite of the course was that the students should experience the children playing their game to get a feel for how emergent their game was compared to commercial games for entertainment, and how the children understood the sustainability issues presented in the game and their links through system thinking. This approach would hopefully maintain a sense of 'fun' as well as linking students' outside-of-school experiences to their educational reality.

The course design was then theoretically inspired by the ideas found in the sociocultural theories of situated learning (Lave and Wenger, 1991; Wenger, 1998; Wenger, 2006), but also by cognitive theories such as experiential learning (Kolb, 1983) and the more modern approach of situated cognition studies suggested by James Paul Gee (Gee, 2010). Situated cognition theories bring in theory from other disciplines such as physics, psychology, brain research etc., and emphasize the conviction that thinking is connected and that it changes across actual situations, where thinking is not a process of applying abstract generalizations, definitions and rules, but rather dynamic images tied to our perception of the world and our bodies, feelings and internal states (Churchland, 1986; Damasio, 1994). Cognitive activities are tied to experiences of goal-oriented action in the material and social world and we think at our best abilities when we use previous experiences to prepare ourselves for future action. Gee (Gee, 2010) uses so-called connectionist or networked computers that search for and store patterns from the outside world as an analogy; humans look for patterns in their experience, and as they grow and their experience accumulates, they will discover deeper and more subtle patterns, which will help them predict what might happen in the future when they act to accomplish their goals.

Situated learning builds on the idea that participation in a community of practice with a common content, goal and mutual engagement stimulates and facilitates learning. In our project the game development class or the game playing community constitutes such a community of practice. While the students or children are busy designing or playing a game they learn system thinking or sustainability, almost without noticing. The main impact of the situated learning theories to the system thinking course-design came through acknowledging the situatedness of knowledge and the need to create learning situations in which theoretical ways of knowing were deeply connected to complex practices in which the students were expected and wished to participate.

Problem-based learning originates from the medical school in the 1960s and was first implemented by Howard Barrows and colleagues (Barrows 1980). The students were organized in small groups, usually 6-10 persons led by a tutor. The focus of the groups was to solve problems from the real world they were about to enter, and to study the theory they needed to solve these task themselves. The tutors' roles were more supervisors and facilitators than a traditional teacher. Problem-based learning has since then become an independent pedagogical concept often used in all levels of education. The principles of problem-based learning can be described as follows (Pettersen, 2005, p. 127):

- The study builds on practice-related case descriptions, reports and studies
- The students – both individually and in groups – receive support, assistance and monitoring from a tutor
- Students should develop practical ways of working and develop learning strategies for studying along with practical problem solving and logical reasoning
- Students have the responsibility for their own learning in collaboration with their tutor, with the emphasis on self-regulation and student autonomy
- The teaching, the curriculum and the actual courses are organized in ways that maintain cross- and interdisciplinary approaches
- The study program should facilitate authentic tasks and challenges, which qualify for the students' future professional lives.

The PBL model will in addition usually follow the steps below (based on the 7-step Maastricht model (Maastricht University, 2013) where the first 5 steps are put into one):

- Brainstorm and analyze the problem and define goals and learning objectives.
- Self-study period

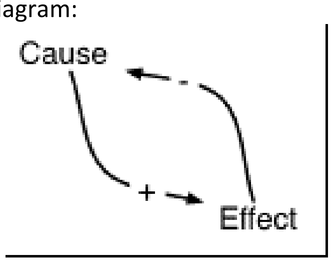
- Debriefing and post discussion.

Step 1 is the initial phase where the students clarify the problem, define the learning goals and identify the theory they will study to solve the problem. Step 2 is the phase where they work alone, while step 3 is the presentation and the debriefing session.

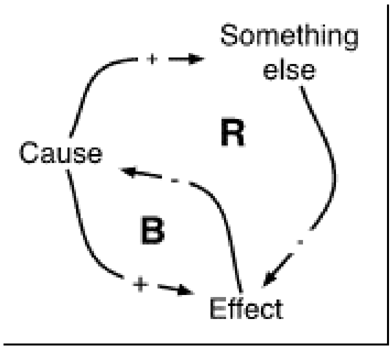
2.2 System thinking and system analysis, a short primer

System Thinking is a common concept for understanding how causal relationships and feedback work in an everyday problem. It has two parts: System Analyses and System Dynamics. System analyses are an easy-to-understand language that describes the connection and causalities in the system, while System Dynamics is the numerical simulation of the system. System Analyses includes group modeling, where we ask the initial questions of the problem and create a mental model structure, using Causal Loop Diagrams. While System Dynamics goes beyond the scope of this paper, we have included a simple primer to understand the basics of System Analyses below.

The basic method in this study is system analysis using causal loop diagrams as defined by Senge (1990), Sterman (2000), Haraldsson and Sverdrup (2004) and Sverdrup and Svensson (2002a,b) Because of the unsustainable path of world society outlined in the introduction, the scope of this study is to investigate a commercial activity that is recognized as unsustainable at present, and explore how to transform it to a sustainable activity. We will further investigate how that activity connects to society and policy planning. It is outside the scope of this study to create numerical simulations models; the discussion remains at the qualitative causal level. Our working hypothesis is that a free market needs both freedom to operate but also distinct and clear rules of engagement, as well as the provision of a market arena. We postulate that the state, business and customer each have necessary roles to play to make a sustainable system out of these essential components. The problem is analyzed using system analysis methods, and clarified using causal loop diagrams. The main tool is called the causal loop diagram:



What this says is that the CAUSE is the causality creating the effect. And the plus on the arrow says that the more cause we have, the more effect we get. It is not sufficient that CAUSE and EFFECT are correlated; there must be a real casualization. The next key issue is, does EFFECT have any feedback on CAUSE? If it does, we need to draw an arrow, and add a + if more of the EFFECT gives more CAUSE or a – if more EFFECT gives less CAUSE.



When this is done, we ask again, is there something else that is affected? Normally there is. And then we draw an arrow from CAUSE to SOMETHING ELSE. And we ask more or less and put the – or the +. And if SOMETHING ELSE has any effect on the EFFECT parameter we draw a signed arrow there too. It could be as shown in the figure above. In the Causal Loop Diagram (CLD) we have two closed loops, one called B and one called R. If we walk a round the loop called R, starting at CAUSE, then because of an even number of -, an increase in CAUSE will come back and cause more increase in CAUSE. We call this a Reinforcing loop (R). Another way to see this reinforcing loop is as a positive feedback loop where the increase of "Something Else" leads to an indirect

 ISSN 1479-439X

increase to “Cause” through the mediation of “Effect”, and only because a decrease in “Effect” leads to an increase in “Cause”. In the other loop, there is an increase in CAUSE which will cause an increase in EFFECT, but an increase in EFFECT will cause a decrease in CAUSE. And increase comes back as a decrease; it is a Balancing loop (B) (Sterman 2000, Senge 1990, Haraldsson and Sverdrup 2004, Haraldsson 2004, Sverdrup et al., 2014).

2.3 A short definition of sustainability

There are different ways to define sustainability, and some even make a point of saying it is not definable. We think there are basically two ways: the difficult one with all the details, and the simple one. The simple definition is quite old; it was given by the emperor Augustus of the Roman Empire in relationship to the engineering of the future Imperial Roman road network. He defined a sustainable plan to be “a plan that could be followed forever, without ruining the functions of the Empire”. He thus recommended the road construction to be done according to principles that would allow them to be used and kept up “forever”. In a vision of a sustainable society, we would like to have a similar vision. There are several names for sustainability out there, and we need to consider which of those are adequate, inadequate, sufficient or necessary. We should discuss the following central concepts:

- Sustainable society
- Sustainable growth
- Sustainable development

In the items below, each one of these will be assessed and commented on. **Sustainable society**. This is a society that can go on for as long as we can reasonably foresee. It is not dependent on growth, but may persist and prevail on a steady level. It is achievable under certain conditions. In a sustainable society there will be growth within the sustainability limits, but also de-growth of what is in excess of the sustainability limits. Growth and de-growth will be in a long-term balance with each other, like waves that rise and sink, as at sea. Overall, the resource use stays within the sustainability limits. There are sustainability limits for the biophysical system, the social system and their interface - the economic system. Sustainability in all aspects will be required for a sustainable system, as defined in the Tripple Bottom Line (John Elkington, 1997), which is emblematic to the importance of system thinking in problem areas of Sustainability. **Sustainable growth** is more problematic. There is no real consensus on how this concept is to be uniquely understood. Sustainable growth was the focus of the Bruntland commission, and very useful in getting the discussion started and focusing on the fact that the present civilization is not sustainable. The Bruntland Commission defined sustainable growth as “the growth that sustains the needs of the present without compromising the welfare of future generations”. It allows for perpetual growth in a finite world, and does not deal with several goal conflicts built into the definition. The present generation should have everything they want and the future generations will have that too. The Bruntland definition was important because it made the necessity of sustainability research evident and pointed towards the need to come up with solutions. But the definition itself is no more than a starting point. It is not sufficiently stringent, it is flavored by political correctness, it allows unlimited growth. However, sustained mass growth forever is a thermodynamic impossibility, and thus a dangerously unsustainable approach. **Sustainable development** is about developing within the sustainability frames that exist for society. It implies that there are quantifiable limits to physical consumption and to material use losses, limits to natural system acceptable damage, and that development must be understood under such conditions. Sustainable development implies development within the sustainability boundaries. It may mean material and energy consumption contraction and convergence, and for societies in resource overshoot, contraction for all. It means that for some situations, we may be wise to consider supplying sufficiency for the many before affluence for the few. Sustainable development concerns not only physical aspects, but also involves development of the social sphere and of society’s structures (Costanza and Daly 1992, Costanza et al., 1992, Sverdrup and Svensson 2002, 2004, EU 2008, 2014).

3 Project Design

The methodology should bring together the 4 major fields into one unity - System thinking, Sustainability, Game development and Adaptive learning - and describe how this was designed and consepted.

The frame for the project was a 6-weeks full-time course in system thinking for second year undergraduate students in a bachelor course in Game Technology and Simulation in the game technology studies at Hedmark University College. In the first 3 semesters of their education these students had studied game design, 3D modeling, game programming, animation and project-based game development, and had the basic knowledge in game production. System thinking and Sustainability were new to them and were taught through projects in this class. So was the basic theory for making the pervasive game.

The first half of the course was used to teach them system thinking, system analyses and system dynamics in a traditional way through lectures and sustainability assignments (Meadows.et al., 1972; Forrester, 1971; Senge, 1990; Sterman, 2000; Schlyter et al. 2012; Sverdrup et al. 2014; Haraldsson and Sverdrup, 2004). The last part of the course (3 weeks) was used to teach and develop the various aspects of the pervasive game.

During a brainstorming process with the whole class, it was decided that the game should be based on core curriculum goals from elementary school, and that the pervasive game should include three digital mini-games also developed on themes from the core curriculum. It should be possible to play the game in one school day, which essentially means 5,5 hours.

The students were divided into four groups with three to six students in each group. The largest group was responsible for the overarching pervasive game, while the three smaller groups made the digital mini-games. Every day in the development period, the class had a meeting to discuss the overall game design, the development progression, and how to solve challenges that arose during the design- and development process.

Due to the short development time for the game and the students' relatively little practice in system thinking, we decided to provide the students with basic system CLDs (Causal Loop Diagrams) of the mini-games. These CLDs gave them a broad overview of the system for which they were to create the games. The students would have to pick a part of these large systems and modify the CLDs so they reflected their own game design. The CLD for the main pervasive game had to be developed from scratch. Several system thinking sessions were also held in this period, which ensured that the students really used system thinking in their game development, and the game became consistent in terms of the sustainability content. The system models were programmed in Stella and the three digital mini-games in the Unity game engine. To preserve the sustainable solutions simulated in Stella, equations and results from Stella were programmed directly into Unity.

The students had to perform two internal design reviews. These were basically playtests with the aim of revealing and mending flaws in the game design. The students should conduct the first playtest alone, while the second was more thorough and used SurveyMonkey to plan, define and store the results.

Project management was to be in focus every day too; each group was to continually maintain project plans and time estimates, and every student was required to write personal blogs from the development every day. These blogs served as notes for the mandatory research report each student had to file after the course and project were finished. This report should focus on their own learning, motivation, creativity and collaboration skills as well as their reflections on the system thinking used in the game. They were also to write about the implementation phase and the game play done by the children.

4 Methods and Research design

The Research design includes several research methods and could be described as Practitioner Inquiry as stance (Cochran-Smith & Lytle, 2009). According to Cochran-Smith and Lytle practitioner includes action research, teacher research, self-study, the scholarship of teaching and using practice as a site for research. Practitioner inquiry intentionally blurs boundaries between teaching, practice and inquiry and research data is systematically collected throughout the process. The design also draws on methods from system dynamics theory such as system analyses and causal loop wrapping and loop analyses in order to understand system dynamics (Senge, 1990; Sterman, 2000; Haraldsson and Sverdrup, 2004). System analysis is used to map causalities involved in the processes studied. These causal loop diagrams constitute knowledge-maps for the system, and these are iteratively tested against data, experiences and qualitative information in a "learning loop mode" as illustrated in Figure 1. The system analysis process becomes an iterative adapting learning process (Senge 1990). When non-researcher are present, such as stakeholders or students, then these are included in the process, the term for this is an adaptive social learning process, a powerful participatory pedagogical tool.

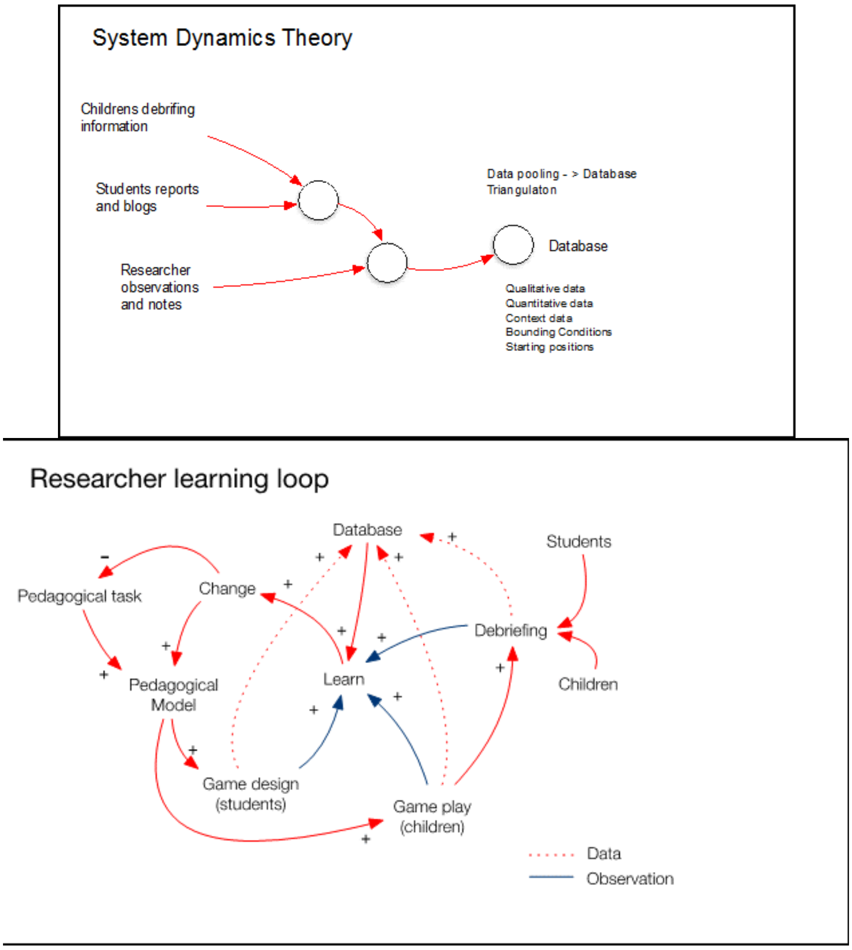


Figure 1: System Dynamics Theory and the Researcher learning loop for the project.

Research data were collected from all parts of the project. The researchers took notes and pictures during the game development process with the students as well as in the game playing with the children. They also made meeting resumes from the many meetings during the game development process.

The students supplied research data in the form of their report (se above). They also had a 4-hour system thinking exam after the course which hopefully would give us some data about their learning. Finally they filled in a thorough course evaluation where they could anonymously give their views on the course.

The children filled in a 17-page diary that summed up learning from game experience, and had an hour-long system thinking session in the end of the day that was documented thoroughly by the researchers. The children's teacher, who knew the children well, also observed the game playing and gave the research team valuable background information about the children playing the game.

5 The game

As mentioned above it was decided that the theme of the game should be based on core curriculum goals from elementary school, and we selected *"explain how production and consumption can destroy ecosystems and pollute earth, water and air, and discuss how this can be prevented and repaired"* to be the main theme of the game.

Further, the game should be cross mediated, meaning that it should contain a well-written narrative that should be told through actors, webpages, videos, clues on different locations and the digital mini-games.

Taking the core curriculum as inspiration the students decided to implement the following digital mini- games:

- a digital game showing the relationship between human consumption and the ecosystem, and how we can prevent and repair the pollution of earth, water, and air.
- a digital game showing refugees what they can expect their lives will be like as refugees.
- a game showing ecological principles from the perspective of an earthworm.

To create a coherent and logical structure, both the digital games and the overarching pervasive game should be designed and analyzed using system thinking. The picture below shows an overview of the project.

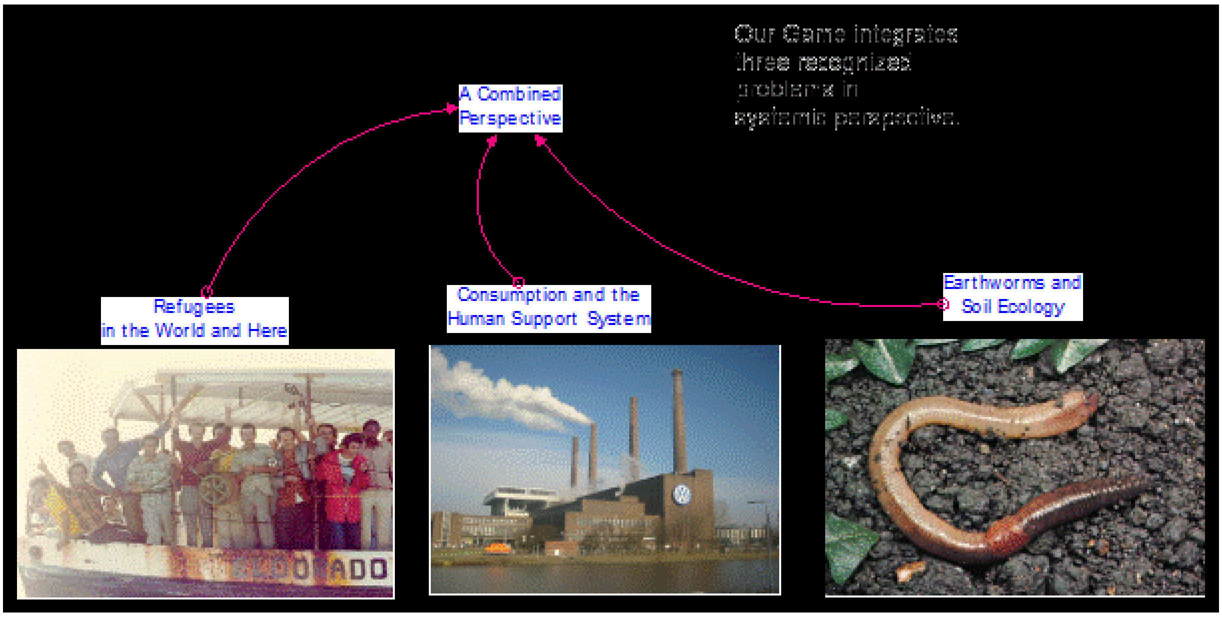


Figure 2: The overall pervasive game.

The picture shows how the *overall pervasive game* includes 3 digital games; *the island game*, *the refugee game* and *the earthworm game*. Below, the 3 digital games and the overall pervasive game are described in more detail.

6 The Island game

The Island game focus was to show the relationship between human consumption and the ecosystem, and how we can prevent and repair the pollution of earth, air and water. The goal was to build, balance and trade the recourses of an island. It was designed to teach the children about the balance in nature; how we need factories and products, but also that we can make things in a more sustainable way than we currently do. We showed them that we need to think about where food comes from, that production and consumption create waste and CO_2, and that this waste can ruin nature and make living there in these conditions unpleasant or impossible. Hopefully the children will learn that an island can prosper if they have fugitives come to their place, but also that the fugitives will need the island to adjust to their new situation.

 ISSN 1479-439X

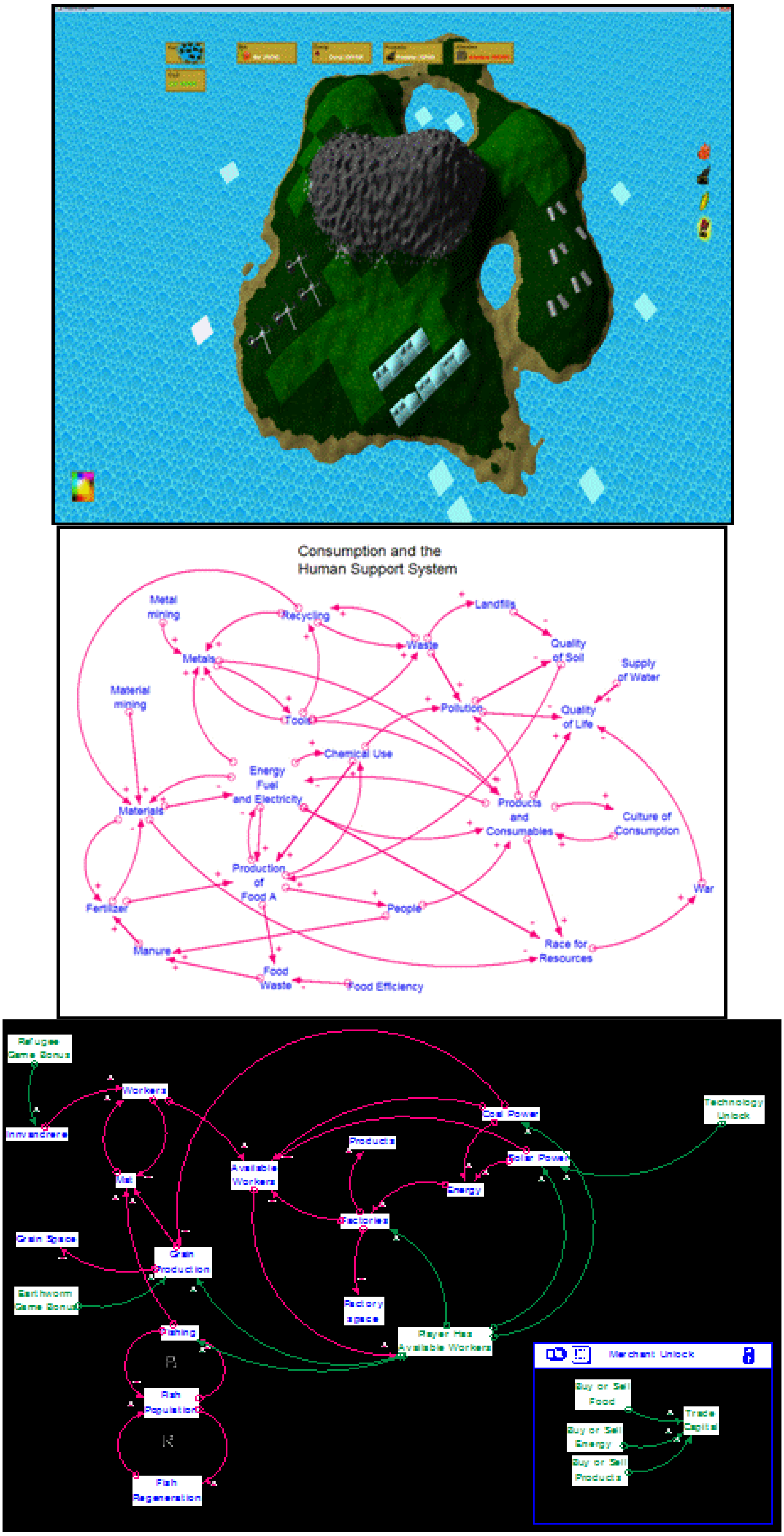


Figure 3: The Island game screen, the initial CLD and the students CLD.

The CLD shows how an increasing population leads to the production of more products and consumables, and to more pollution and waste. The amount and quality of water and food suffers and leads a race for resources and even war. Mining of materials and the production of fertilizers can increase the production of food but they require more mining of metals. Recycling can also increase the availability of minerals and metals, but increases the amount of waste.

6.1.1 Gameplay

When the game starts the players have many workers, but very little food to give them. The children will have to build factories and fishing boats and farms to feed the workers. They will quickly run out of workers, and to gain more of them they will have to play the Fugitive Game where, upon completion, they get a color code that will give them more workers. But, with more workers they need more food, and they have to go play the Earthworms game to be able to grow food more efficiently. They now have food and workers and can start building factories. However, factories and power plants pump out harmful chemicals and CO_2, so they have to find a way to stop the CO_2 level from getting too high. The solution is finding some money in one of the other game storylines. They can also trade goods and services with the actors to earn more money. When they have collected enough money they get a code for solar panels and a code for trade. They will then be able to build factories that run on solar power and use half of the energy without exceeding the dangerous CO2 level. Finally, they can use the extra food they have to trade with other islands, making their island balanced and prospering.

6.2 The Earthworm game

While the previous game showed the larger picture, depicting how different components in the world were linked together, we also wanted to include a game focusing on the small details. This was to show that small things in nature also had an impact on the larger system. We therefore included an assignment to make a game showing soil ecology and fertility from the perspective of an earthworm. From that perspective we decided to include a digital game showing ecological principles from the perspective of an earthworm.

The core goal of the Earthworm game is to get the best possible harvest. The player can achieve that by moving earthworms to the different parts of the soil (see picture below) and use as little pesticide as possible.

ISSN 1479-439X

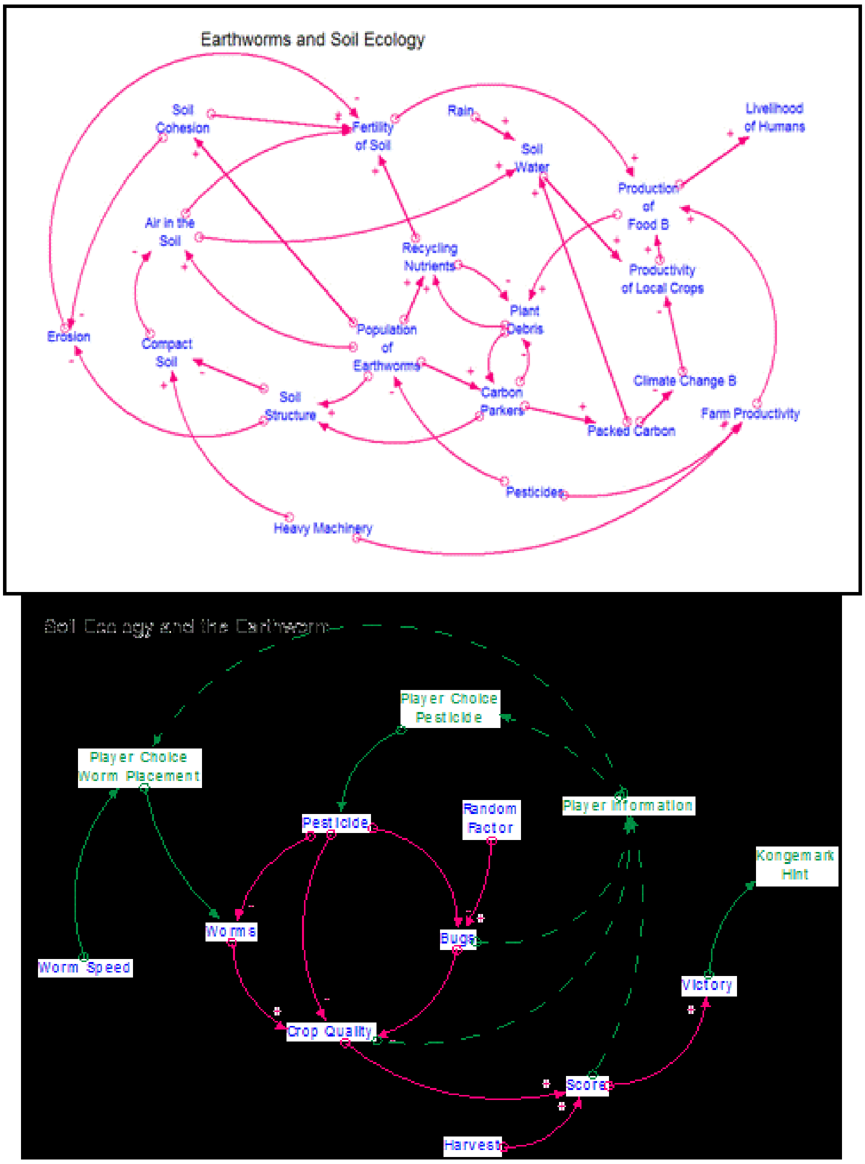


Figure 4: The Worm Game screen, the initial CLD and the students CLD.

The CLDs shows how a greater population of Earthworms gives a less compact soil structure and lets air, water and nutrients flow into the soil and give the soil better fertility. The farm productivity may increase both with the use of heavy machinery and pesticides and lead to a larger food production, but the CLD also shows how the heavy machinery leads to a more compact soil and that fertilizer kills the earthworms. Lastly the CLD shows how soil with a healthy population of earth worms leads to a better soil structure, which is less exposed to erosion.

The gameplay is very simple- it's all about making the corn grow as fast as possible. To do that the children will have to move the earthworms to the parts of the soil that need it the most, and keep the amount of pesticides as low as possible (to avoid killing the worms). The more worms you have in the soil, the better becomes the crop. If bugs come to eat your crop, you may have to use pesticide to not lose the crop, but the more you use of them, the more you kill the worms, which in turn impacts your crop. The game goes faster and faster which makes it harder and harder to place the worms in the right place.

6.3 The Refugee game

The Refugee game is a bit different from the other two games in that it doesn't necessarily focus on the larger system, but more about *identification* and understanding how it feels to be a refugee.

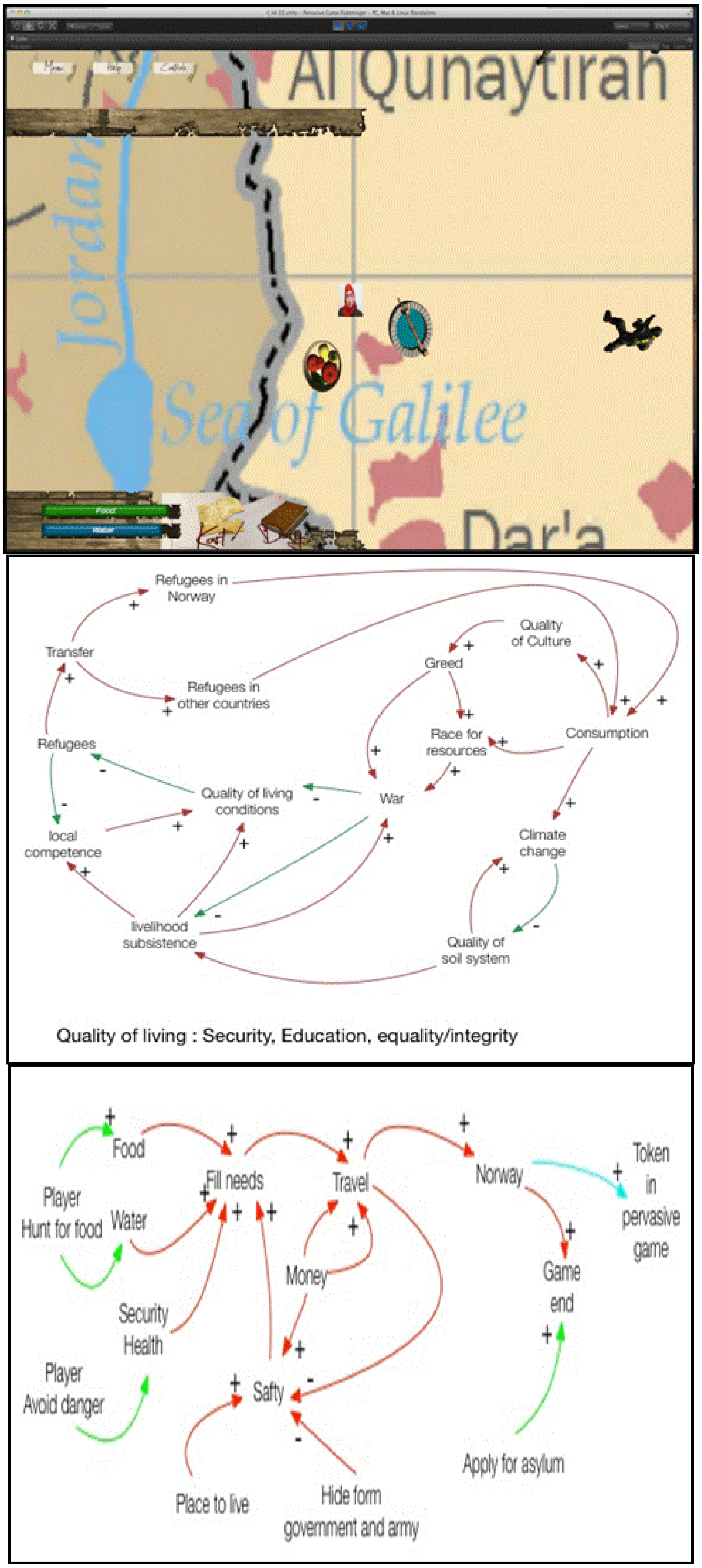


Figure 5: The Refugee game screen, the original CLD and the student CLD.

The CLD shows how poor quality of living and war leads to more refugees and how more refugees leads to higher consumption, greed and a race for resources, possibly war, climate change, and lower quality of the soil system and livelihood subsidence.

The gameplay in this game is also very simple; the children have to avoid danger and collect enough food to survive during their escape from their war-torn land to another country to apply for asylum. When they survive and get to a new country their application for asylum is tried with regards to the probability of getting asylum in that country (based on the real numbers). The children very often felt it was unjustified that their applications for asylum in the new country were not accepted after all their struggles and hardships.

6.4 The Pervasive Game

The fourth and largest student group were to create the overarching pervasive game. The focus here was on making a good story that integrated the smaller digital games in a consistent way.

The game started with a video of a refugee who had to flee his country because of famine and war as a consequence of pollution and climate change. The children find his video blog, and a request to find out what has happened to the environment in his country and how pollution and climate change there can be prevented. The players will then be divided into teams, and each team will receive a backpack with a pair of walkie-talkies, riddles and clues, a treasure map, computer passwords and a notebook. They will then have to solve the clues at different locations around the school, and talk to actors about sustainability-related subjects to be able to proceed in the game.

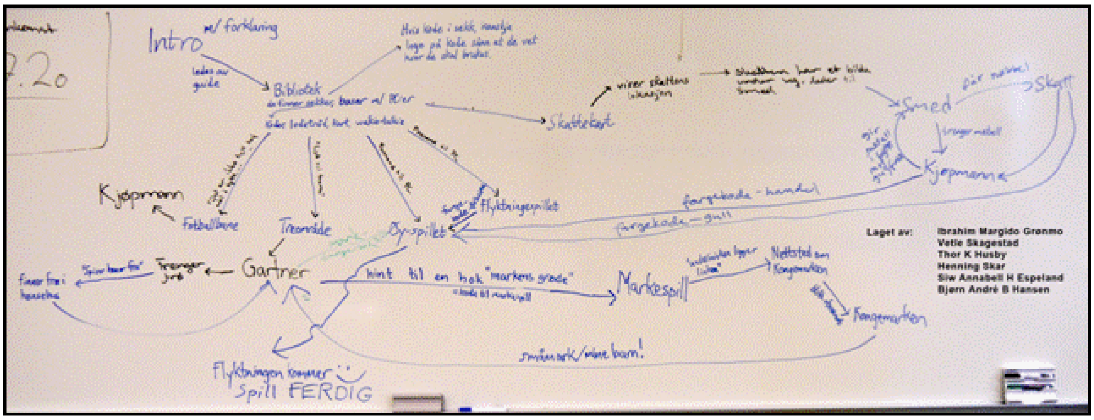


Figure 6: Blackboard sketch of the final game.

6.4.1 Pervasive game actors

The game contains several characters, and the students took the roles of actors. These characters were:

- The gardener: The gardener talks to the children about ecology, biological diversity and pollution. The children will help him plant vegetables and in return he will give them new quests related to trading.
- The King of Earth Worms: According to the legends, this mythical creature is about the size of a man residing in the area around the school. This character talked to the children about the importance of the small creatures in the ecosystem.
- The locksmith: This character will help the children create a key for the treasure chest they had found, by using recycled metal. This character talked to the pupils about the importance of recycling.
- The merchant: The merchant will give the players trade-related quests, and teach them about fair-trade and resources.
- The guides: Each team was appointed one student guide that they could request help from if needed.
- The refugee: The refugee will first appear in the video that is shown to the children, and will return towards the end of the game to talk to the players and discuss their experiences.

Below are a flow chart and the CLD for the game. The CLD shows how the mini-games are integrated with each other and the pervasive game. For example, the Worm game has to be played and won to have food resources for the island game, and the Refugee game has to be resolved to have a workforce for the factories and the power plants, also in the Island game. The treasure in the Pervasive game has to be found to be able to buy things in the Island game. The CLD also shows how the pollution, war and food/resources are the bases for the refugee escape, and how earthworms are needed to make food grow.

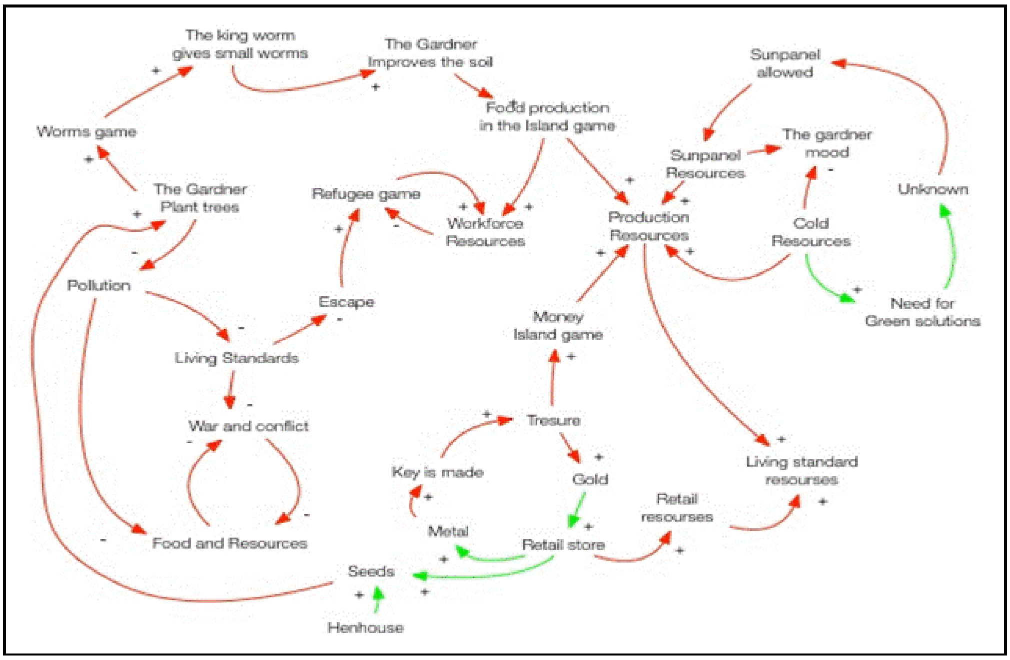


Figure 7: The Pervasive game CLD is shown as developed by the students. This shows the overall system causal loop diagram with all the small games inside the large.

An action flow chart is shown in Figure 8.

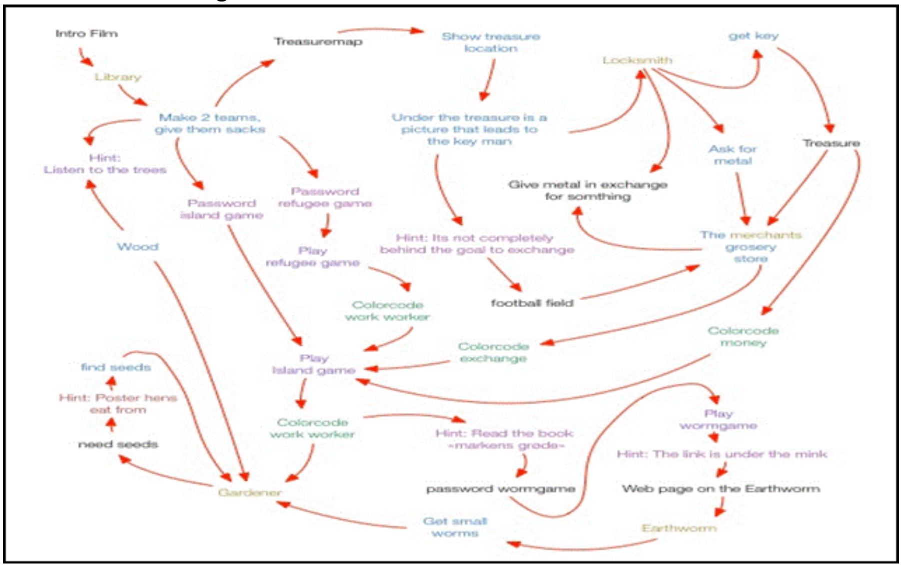


Figure 8: Game flow chart. The different action is color coded in the following way: Yellow – Actors, Green – The codes needed to solve the game, Magneta – Hints and passwords, Purple– Mini-games, Dark blue – Locations and actions.

6.5 The aftermath

After the game was completed, the children were given the refugee's diary. This 17-page diary has two functions. First, it was a way to summarize the main principles they had learned from the game in a way that was closely related to the narrative of the game. Secondly, the diary included several written assignments that the pupils have to solve, related to the game they have played.

After the children had filled in the diary they were gathered in the classroom to conduct a system thinking debriefing session. This session was led by one of the teachers who asked the children questions about the different parts of the game, and how they were connected. Another teacher drew a system CLD on the blackboard from the children's answers and made connections and signs in the way the children wanted them.

7 Discussion

This section discusses briefly some of the thoughts after the game playing in the elementary school.

Firstly**,** the energy, enthusiasm and creativity the children put into playing the game was impressive. They exhausted the assigned student helpers in 30 minutes, and the children basically solved the game that was meant to last the whole day in 2,5 hours. It was very clear that they considered the game playing fun; one child later enthusiastically called the day the "best school day ever". We could also see some of the same enthusiasm during the work with the diary and the system thinking session and some children went to the library to borrow books to read more about the things they had learned in the game.

The diary was designed in a way so that we should be able to check how much of the sustainability-related topics from the game they had learned, and also if they understood the related connections between them. All the children filled in this diary, and when we later studied them we found that the majority of the children summed up their sustainability knowledge in a thorough and good way, which indicates that there was a good learning outcome from the game.

The system thinking debriefing session enabled us to check further how much of the connectivity between the events they had understood. Within an hour the blackboard contained a fairly complex CLD, which impressed both the students and the research group. This leads us to believe that games are a good way to teach system thinking. We believe the system thinking sessions could be even longer and more thorough without boring the children.

The children's elementary school teacher pointed out that the children who normally didn't like the traditional school teaching excelled while playing the game. They paid attention, they progressed quickly, and they showed leadership skills. This shows that teaching through pervasive game playing in school also can help to activate those children who do not so easily fit into the traditional school system.

Observations indicated that the students also had fun both when they developed the game and when they observed how the children played it. They also wrote the same in their reports; below are some of their comments:

- "It has been very fun to participate in this project, with the best part being the playtest with the 5th graders. I am satisfied with the whole project, and although it was hard to see how everything would turn out when we started the project, we managed to finish, and we finished with bravura."
- "I see the pervasive game part of this course as an interesting experience where we learned a lot, and comments like "The best school day ever", indicate that the 5th- and 7th graders felt the same".
- "This was an enjoyable and rewarding assignment to work with, and although we're neither actors or have developed a game like this before, I felt it was a very successful project for us students, and I hope the teachers feel the same".
- "To summarize, I'll say this project has been a joy to work on. I've both learned from it, and gained many very nice memories. This has been nothing but fun!"

They also describe the project as stealth learning due to the short development frame, but with a big reward at the end - the implementation of the games for the children. Since both the fun factor and the learning outcome for the children seemed to be good, the game playing in itself indicated that the gamification of this project worked well. Regarding the learning outcome for the students – to learn system thinking, the CLDs made by the students' during the game development, the reports and the exam indicated that that they had learned at least as much as previous classes in system thinking done with traditional teaching methods. However, we didn't compare the students' learning in this project in detail with traditional teaching, so we can not say the learning outcome in this setting were better than doing it the traditional way. We believe, and the students also say so in their reports, that learning through making games is more fun and more thorough than traditional learning due to 'learning by doing', and 'learning just in time'; the need to learn pops up when they need it for the game. And of course, the implementation with the children is an event they will remember for a long time and tie their learning to this event.

Although the overall project was successful, the game development part had some limitations.

We found the students didn't have enough experience in system thinking and sustainability to make their own CLDs on the games. The overall CLDs we made for the mini-games helped, but ideally they should make all CLDs themselves. We believe more time should be used on teaching them system thinking and sustainability 'just in time' during the development process of the pervasive game.

The time to develop the games was probably too short. This caused some frustration and several of the students felt that they could have made the game better with some additional time. Several students also pointed out that using CLDs and simulations to check game logic and debug the code proved to be very beneficial; they found the bugs much faster than with traditional tests. Thus they found another use of system thinking in game development not explicitly pointed out to them.

While the students saw a clear need for system thinking in sustainability, they questioned the use of system thinking when developing the small mini-games. The systemic challenges were just too small to "waste" time on developing system CLDs or simulations. However, it was a clear understanding that if the games grew more complex, the system thinking would be a very useful tool both in the game design phase and as a debugging tool.

The game was also solved faster than we expected by the children. This was not a major problem because the time we could use on the diary and the session with System Thinking was increased. This turned out to be both interesting and effective and probably added to the children's learning. However, when we do this next time, the games should be more difficult to solve. The diligence, creativity and energy the children put into solving these games were truly amazing, and we believe the playing time and level of difficulty could easily be increased without them being bored or exhausted.

8 Conclusions

Without more thorough analyses of the research data in the project, it is hard to draw any conclusions. However, we believe that the combination of system analysis, sustainability, game development or game playing and learning is a good one. Its quite clear that both the school children and the students found the project fun, and the submitted reports, exam and debriefing sessions all indicate that the learning outcome for both were at least as good as with traditional methods. As stated above, we also believe the learning process is more thorough than traditional learning due to the "learning by doing" aspect of the project.

The interest from the schools in trying out this concept was also exceptional; after sending out a one-page request to schools to implement the game only 4 weeks before we planned the actual test, we got answers from 15 schools and more than 1500 children wanting us to implement the game in their school. Bearing this in mind, we will most definitely implement the project again

References

Barrows, H. S., & Tamblyn, R. (1980). Problem-based learning: An approach to medical education. New York: Springer.

Churchland, P.S. (1986). Neuophilosophy: Towards a Unified Science of the Mind/Brain., Cambridge, MA:MIT Press.

Cochra-Smith M. & Lythle S.L. (2009). Inquiry as Stance. Practitioners Research for the next generation, New York: Teachers College Press

Costanza, R., Daly, H.E.. Natural capital and sustainable development. Conservation Biology 1992; 6:1-10

Costanza, R., Daily, G and Ehrlich, P. 1992. Population, sustainability and the earths carrying capacity. A framework for estimating population sizes and lifestyles that could be sustained without undermining future generations. Journal of Bioscience 1-19.

Damasio, A.R. (1994). Decartes' Error: Emotion, Reason and the Human Brain, New York: Avon

Deterding, S., Dixon, D., Khaled, R., & Nacke, L. (2011). From game design elements to gamefulness: Defining "gamification". MindTrek '11. Proceedings of the 15th International Academic MindTrek Conference: Envisioning Future Media Environments (pp. 9-15).

Dewey, J. (1916). Democracy and education: An introduction to the philosophy of education. New York: Macmillan.

Elkington, J. (1997). Cannibals with forks – Triple bottom line of 21st century business. Stoney Creek, CT: New Society Publishers.

EU sustainability policies 2008. Annexes to impact assessments guidelines. Revised version 2008. Public website with downloadable policy documents. http://ec.europa.eu/smart-regulation/impact/commission_guidelines/commission_guidelines_en.htm http://ec.europa.eu/smart-regulation/impact/consultation/docs/ia_guidelines_draft_annexes_final_en.pdf. Accessed 2008, 2010, 2013, 2014, 2015.

EU sustainability policies 2014. WEB public sites: http://ec.europa.eu/environment/eussd/, http://europa.eu/legislation_summaries/environment/sustainable_development/l28117_en.htm, http://www.bmub.bund.de/en/topics/europe-international/europe-and-environment/eu-strategy-for-sustainable-development/, http://www.sd-network.eu/?k=basics%20of%20SD%20strategies

Forrester, J. (1971). World Dynamics, Pegasus Communications, Waltham MA.

Gee, J. P. (2004). Situated language and learning: A critique of traditional schooling. London: Routledge.

Gee, J. P. (2007). Good Video Games and Good Learning: Collected Essays on Video Games, Learning and Literacy (New Literacies and Digital Epistemologies) .

Gee, J. P. (2010). New digital media and learning as an emerging area and "worked examples" as one way forward. Cambridge, Mass., The MIT Press.

Haraldsson, H. V. (2004). Introduction to systems thinking and causal loops diagrams. Reports in ecology and environmental engineering. Report 1:2004. 49pp

Haraldsson, H.V., Sverdrup, H. (2004). Finding Simplicity in Complexity in Biogeochemical Modelling. J. Wiley and Sons Ltd., Chichester, pp. 211-223.

Kolb, D. A. (1983). Experiential Learning: Experience as the Source of Learning and Development. Prentice Hall

KZero. (2011). Universe chart Q4 2011: Avg User Age 10 to 15. Retrieved from http://www.kzero.co.uk/blog/universe-chart-q4-2011-avg-user-age-10-15/

Lave, J., & Wenger, E. (1991). Situated learning. Legitimate peripheral participation. Cambridge: Cambridge University Press.

Lyneis D., Stuntz L. (2007). System Dynamics in K-12 Education:Lessons learned. The creative Learning Exchange, v17, n2, spring 2008.

Maastricht University. (2013). Problem based learning preparatory website. Retrieved from http://www.umpblprep.nl/

Meadows, D.H., Meadows, D.L., Randers, J. , Behrens W. (1972). Limits to Growth, Universe Books, New York.

Pettersen, R. C. (2005). Kvalitetslæring i høgere utdanning: Innføring i problem- og praksisbasert didaktikk [eng: Quality learning in higher education: A primer in problem- and practise based didactics]. Oslo: Universitetsforlaget, Retrived from: http://www.universitetsforlaget.no/nettbutikk/kvalitetslaering-i-hoyere-utdanning-uf.html.

Schlyter, P., Stjernquist, I., Sverdrup, H., (2012). Handling complex environmental issues—Formal group modelling as a deliberative platform at the Science-Policy-Democracy interface, Proceedings of the 30th International Conference of the System Dynamics Society, Switzerland.

Senge, P., (1990). The fifth discipline, the art and practice of the learning organisation, Century Business, New York.

Shaffer, D. W. (2008). How Computer Games Help Children Learn.

Sverdrup H. (Ed.), Haraldsson, H., Koca, D., Belyazid, S. 2014 System Thinking, System Analysis and System Dynamics: Modelling Procedures for Communicating Insight and Understanding by Using Adaptive Learning in Engineering for Sustainability. Háskolaprent Reykjavik. 310pp.

Sverdrup, H., Svensson, M. 2002 Defining sustainability. In: Developing principles for sustainable forestry, Results from a research program in southern Sweden. Sverdrup, H. and Stjernquist, I. (Eds.) Managing Forest Ecosystems 5:21-32 Kluwer Academic Publishers, Amsterdam.

Sverdrup, H., Svensson, M. 2004. Defining the concept of sustainability, a matter of systems analysis. In: M. Olsson; G. Sjöstedt (Eds.); Revealing Complex Structures -- Challenges for Swedish Systems Analysis, 122-142. Kluwer Academic Publishers.

Sterman, J.D., (2000). Business Dynamics, System Thinking and Modeling for a Complex World, Irwin McGraw-Hill, New York.

Tapscott, D., (2011). The Net Generation is the smartest generation, Interview Daily Education online database (2010).

Wenger, E. (1998). Communities of practice: Learning, meaning, and identity. Cambridge: Cambridge University Press.

Wenger, E. (2006). Learning for a small planet: A research agenda. Retrieved from http://learning-affordances.wikispaces.com/file/detail/Learning+for+a+small+planet.doc

3

Article 3:
System Thinking in Game Design

Anders Nordby, Håvard Vibeto, Harald U. Sverdrup (2023)
Not published yet

Status:

Article not published yet. Will be submitted to System Dynamics Review in 2023.

https://systemdynamics.org/resources-old/system-dynamics-review/

System Thinking in Game Design

Anders Nordby, Håvard Vibeto, Harald U. Sverdrup
Department of Game Development – The Game School, Inland Norway University of Applied Sciences, Holsetgaten 31, NO-2301 Hamar, NORWAY

Abstract
This article describes how system thinking can be used in game development and game design. We use examples from an advanced game design course taught in the Game School at the Inland Norway University of Applied Sciences to demonstrate which elements of game development can benefit from this methodology. We discuss the course design and how students have used system analyses and system dynamics to design games and to balance game assets and resources by simulating game outcomes.

Keywords: game design, game development, games and learning, system thinking, system dynamic simulation

1. Introduction

Salen and Zimmerman (2004) point out that all games are intrinsically systemic, meaning that all games can be understood as systems. Gee (2007) argues that people learn skills, strategies, and ideas best when they see how they fit into a larger system. Games systematic core are therefore good at helping players see and understand how subsystems fit into an overall system and thus facilitate learning and system thinking (Gee, 2007). Ernst Adams emphasizes that games are complex systems that consist of many parts, and that when these parts are put together, games can display unpredictable behaviour, resulting in emergence (Adams, 2014). Systems in games provide context for interaction, rules and behaviours that players can explore, manipulate and immerse themselves in. Although one can argue that games have an inherently robust systemic quality, especially modern videogames, this is not often discussed explicitly in the game design literature. Game design in practice tends to focus on how to design the ludic elements of a game -- its core gameplay mechanics, rules and goals, and the corresponding aesthetic qualities that make a successful game experience (Schell, 2008). Many modern videogame genres have become complex systems of varying rule sets and modes that the player needs to interact with and understand to successfully play the game. The player needs to understand and predict how the rules will react to their input and then develop different strategies to beat the game or human opponents (Tekinbas et al., 2014). Some genres offer massive, simulated worlds inhabited by NPCs (non-player characters) that each offer many different goals and interactions to the player. An understanding of system thinking when playing these games has in many ways become essential if players are to get the most from their game sessions. Indeed, today players can find videos, wikis, forum posts and other materials online that break down the many systems in videogames in order to successfully master them (Wallach, 2019).

Certain games and genres require metagaming -- the player must understand how the different inner workings of game systems function to play them successfully. Metagaming involves many different elements that affect the game, but are not in the game (Mora-Cantallops & Sicilia, 2018). This use of out-of-game information or resources to affect the player's in-game decisions may include community standards, statistics or datamining. Metagaming often focuses on the strongest, most effective strategies currently available in a game because today's videogames are in constant flux, and each patch or new downloadable content (DLC) can change a game's rules and systems (Blake, 2017; Peterson, 2016).

Designing and playing videogames both demand that participants develop a systemic understanding of the game with regard to not only the behaviour of the individual components of a system, but also the way the system operates and interacts with other systems within the game. As videogames are interactive, players must not only understand a game's systems, they must also decide the best way to intervene to change and manipulate these systems as well as understanding the connections between systems. This means that game designers must make game systems intelligible and interactive, while also ensuring players can have agency over the systems without breaking them (Tekinbas et al., 2014).

As videogames have become increasingly diverse and advanced, design trends have emerged that system thinking can help explain. The growth in sandbox and open world videogames that create massive, interactive worlds that players can use for their own pleasure gives rise to a number of design problems. Many of these games showcase advanced systems on both macro and micro levels that are interwoven and which are supposed to give the player a feeling of meaningful play and agency. As sandbox and open world design become more widespread in the videogame industry, the task of balancing all these systems to create consistently good gameplay and experiences becomes more difficult (Doyle, 2015; Pramath, 2016). The same balancing problems can be seen in another dominant videogame trend, namely, the big multiplayer games that focus on a "games-as-a-service" business model. Games-as-a-service means that a game is constantly updated and monetized with new content that encourages players to keep playing and paying. Games-as-a-service can be found inside the MOBA (multiplayer online battle arena), FPS (first-person shooter), and MMOG (massively multiplayer online games) genres. All create online competitive modes where players compete yet also need to cooperate within teams. A lot of these games have many different heroes, abilities, and weapons that must be balanced against one another. Examples of these kind of games are League of Legends, Dota 2, Overwatch, The Division 2, Destiny 2, and World of Warcraft. Since many of these games are constantly being updated and new content introduced, there is always a chance that changes can alter the game in ways the game designers did not intend – and infuriate the player base. System dynamics simulations can help avoid these problems.

Finally, yet importantly, traditional game design often uses a linear approach when designing games (A causes B causes C causes D, etc.). This is understandable and what our brain wants to do, if not forced to think differently. System thinking, with its causal loop diagrams, does just that -- it forces game designers to discover non-linear feedback loops in games systems. (A causes B but B also has a feedback loop back to A, etc.). The real world is rarely linear. Feedback loops are everywhere, and good game design requires the game-world to be as complex and interactive as the real world.

Seen in this context, it is only natural that the game design process uses system thinking to develop the game. System thinking is a methodology used to understand how causal relationships and feedback loops function in everyday problems. It is comprised of two parts: system analyses and system dynamics. System analysis uses language and a set of diagrams to describe connections and casualties in a system. System dynamics is the numerical simulation of a system.

2. Scope and Objectives

We are lecturers at the Game School at the Inland Norway University of Applied Sciences, Norway. In this article, we discuss how we have changed the focus of a course in system thinking for game students from traditional system thinking to advanced game design. Students are given assignments where they must use system thinking to balance and further understand games and game design. The course, *Games and System Thinking*, is the follow up course to a more traditional course in game design and has been taught in the technical bachelor of the Game School since 2012. The article will discuss two examples of assignments given to the students during the course. Although the implementation of system thinking in the course is a work in progress, we will try to shed some light on two central questions in our current research: First, does system thinking improve students' understanding of how games work? Second, can system thinking in general improve game design?

3. Methodology

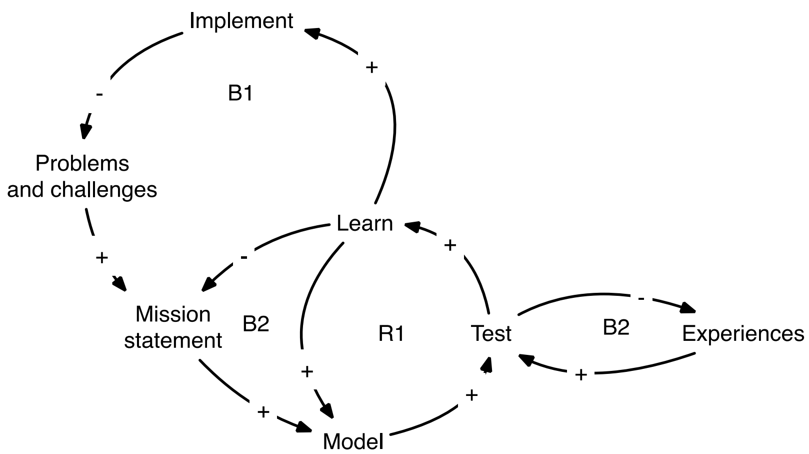
3.1 System Thinking

The main tool in system analysis is the causal loop diagram (CLD) as defined by Senge (1990), Sterman

(2000), Haraldsson and Sverdrup (2004), and Sverdrup et al. (2020). A CLD describe how variables in a system are connected and influence each other. These variables can be causes or effects and are connected by arrows and a label that describes how the effect develops in relation to the cause. A plus symbol indicates that both cause and effect develop in the same direction (e.g., more cause creates more effect). A minus symbol indicates that they diverge from one another (e.g., more cause creates less effect). Arrows can also indicate feedback – that is, the effect can result in changes to the cause. In addition to CLDs, system analysis uses flow charts to show what and how things flow through the system, and reference behaviour patterns (RPB) that show how the system develops over time (not shown here). Figure 1 shows an example of a CLD that describes how the system analysis works. All start with problems and challenges that are brainstormed and developed into a mission statement. The mission statement is then developed further into a model, which is tested. During the testing students learn and gain experience that is used to create better tests (the B2 loop). Learning often leads to corrections in the mission statement or model (the B2 branch). When learning is good enough, the game developers can implement the system, which usually leads to fewer problems and challenges to solve. The R indicates that the loop is reinforcing or always increasing, while the B indicates a balanced loop, meaning it will stabilize on a particular value.

Figure 1

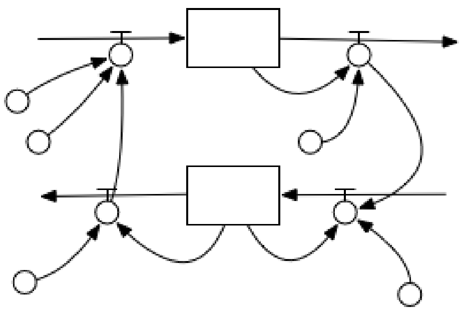
CLD Learning Loop



During the system dynamics component of system thinking, the numerical simulation of the system is developed. This is usually done with simulation software and involves two steps. First, a translation of the CLD using a simulation software, often through a stock and flow diagram (Figure 2), and then research to find appropriate and realistic numerical values or ranges for all variables in the system. The stock-flow diagram in Figure 1 shows how the CLD can be implemented through stocks and flows in a simulation. In a large systems simulation, finding the correct values can often be difficult and time-consuming. The translation from the CLD to the simulation software involves defining time variables and steps, establishing which values should be accumulated, which values should be used as input and output, and how mathematical functions should be defined in order to affect the system in the desired way. When the model is finished, different scenarios are simulated to confirm the model works (often by checking it against the real world). When the simulation aligns with the real-world scenario, game designers can use it to simulate how the system will behave with different inputs. In game design, it can be useful to randomize input values to simulate different playstyles and to run many simulations to learn which one results in the best gameplay.

Figure 2

Stock and Flow Diagram.



Note: The boxes are stocks, and the arrows indicate a flow.

Simulation data, mathematical equations, or pseudo code can be exported from the simulator (Stella Architect). Although data from simulations can be imported directly into the game engine (Unreal or Unity) and used when needed, pseudo code offers more flexibility if it is translated into the game engine programming language (C, C++, Java, etc.) and run directly in the game engine because it allows the students to rerun simulations in the game engine whenever the conditions of the original simulations change. Mathematical equations are the basis for the game design and can be used directly in the game programming.

3.2 Learning Methodology

The *Games and Systems Thinking* course uses a problem-based (PBL) learning approach. Students are divided into groups with a tutor and the development process of the game is divided into steps. In each step, the student groups decide what is necessary to study in order to solve the challenge. Each step prepares the player for the next step until the game is done or they go on to the next game. Problem-based learning was first implemented in medical schools in the late 1960s by Howard Barrows and his colleagues (Barrows, 1980), and has since established itself as an independent learning method, especially in higher education (Pettersen, 2005), (Dochy et al., 2003).

The game development process is also seen as a “community of practice” where relevant school topics pop up as students try to create a game. This suggests a closer look at learning methodologies such as situated learning (Lave & Wenger, 1991; Wenger, 1998), “learning by doing” (Dewey, 1916), and “learning just in time” (Gee, 2007), but these are not discussed further in this article.

3.3 Research Methodology

Course designers and instructors use of system thinking as a tool for game design is quite new. Because of this we expect to have to work further to refine the course design. We see the course as a research project with action research as the research methodology. Action research is an interactive inquiry process that balances problem-solving actions implemented in a collaborative context with data-driven collaborative analysis or research to understand underlying causes, thereby enabling future predictions about personal and organizational change (Reason, 2001). Throughout the course, we collect notes, feedback from students, and reports and results from each assignment. Every year, we examine this data to see how the course can be improved and what information it provides about how system thinking functions in game design situations.

4. Theory

4.1 Introduction to State-of-the-Art Game Design

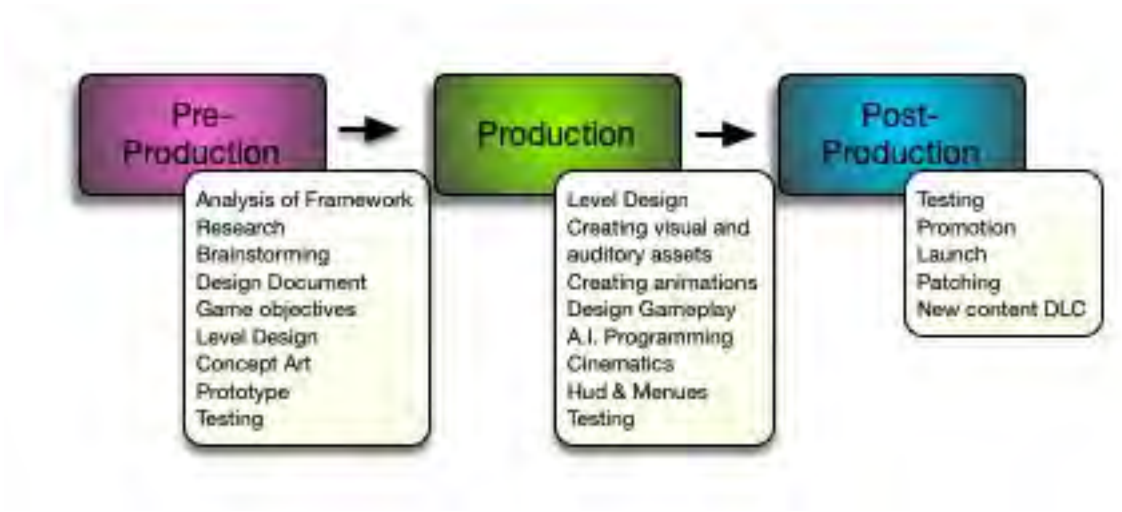
The need to have textbooks on game design and to foster an academic understanding of how to develop a good game became important as the videogame industry has grown. In its infancy, videogame production was the domain of engineers, programmers, and self-taught enthusiasts with no formal training in game design (Donovan, 2010; Kent, 2001). At the end of the 1990s, this changed. The videogame industry was growing internationally, production costs were rising, and there was a need to professionalize game development (Egenfeldt-Nielsen et al., 2008). Videogame development grew from being one-person projects to large-scale productions that employed hundreds. Budgets increased, development phases were standardized, and specialized roles related to work tasks such as sound, programming, animation, and game design were developed. Videogames had become so big, expensive, and specialized that the gaming industry needed qualified professionals with formal education, and higher education institutions needed to educate professionals in a number of fields in game development. In response, game design literature emerged, with a goal of explaining how to create high quality videogames. One of the first books on videogame design was published in 1984 (Salen & Zimmerman, 2006). As the years passed and industry grew, more and more books on game design – with the main focus on videogames – appeared. Game design literature clearly emphasized the ludic aspects of videogames. Attention was given to rules, interactivity, core mechanics, gameplay, the development cycle, and commercial questions (Shell, 2014; Salen & Zimmerman, 2004; Adams, 2014; Rouse, 2005).

For the most part, this literature highlighted gameplay as a key element unique to videogames and that also needed to be the main focus of games. Gameplay is often viewed as the secret ingredient that makes games fun to play. But what characterizes gameplay? Richard Rouse, for example, has argued: "A game's gameplay is the degree and nature of the interactivity that the game includes, i.e., how players are able to interact with the game-world and how that game-world reacts to the choices players make." (Rouse, 2005). Rouse links gameplay to a videogame's interactivity. Accordingly, game design must focus on determining the choices available to players and considering the ramifications of the preferred alternative (Rouse, 2005). This means that videogames need to produce systems that can handle the input and output of the players' choices. As videogame engines become more advanced and hardware more powerful, videogames need to develop scientific methods to guide and help the designer in making good design choices.

However, many game design publications are written by veteran game designers who do not necessarily have a research or academic background. This often limits the game design literature because it is mostly based on subjective experiences and includes little research or methods from other disciplines. As a result, in this literature, practical, technical, and ludic questions in game design easily eclipse more theoretical examinations of how to use tools and methods to help balance games (see Figure 3). This is something the Game School at the Inland Norway University of Applied Sciences has tried to remedy – by introducing system thinking into game design education. Even though system thinking has not been addressed in game design books, some game designers and game researchers have used **Unified Modeling Language** (**UML**) to map specific gameplay elements with the goal of assisting the balancing and testing of features and systems. Adams and Dormans (2012) have developed a form of UML called Machinations to help designers and students of game design create, document, simulate, and test (primarily) the internal economy of a game (Adams & Dormans, 2012).

Figure 3

Traditional Game Design Development Structure



4.2 Where Does System Thinking Fit in the Game Design Process?

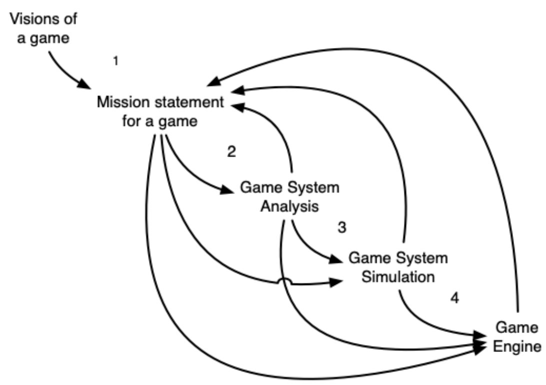
When using system thinking in game design, the process is approximately as follows:

- ***0.*** Vision of a game
- ***1.*** Mission statement; brainstorm and describe the game idea
- ***2.*** System analysis -> Complete system analysis of the game idea and design a qualitative model of the game
- ***3.*** System dynamics -> Simulate the model, test scenarios and export the code to the game engine
- 4. Game engine -> Develop programming and art (and maybe art system analysis) within the game engine

Unlike the traditional gaming model illustrated in Figure 3, in a system thinking developement process, all steps take place in preproduction. However, because production and postproduction might lead to changes in system analysis, simulations, or even the game idea or mission statement, feedback is an integral component of preproduction. This is shown in Figure 4 below.

Figure 4

Overview of the Game Design Process Using System Thinking



The process of adding system thinking tools to the game design process has usually been divided into the four steps shown in Figure 4. In this process the first step is brainstorming the game idea outside the constraints of any structuration tools. It is important to make sure creativity flows freely and is unhindered in this phase. As there are many descriptions in the literature on how to do this, we will not discuss this step here.

The second step in a system thinking approach is system analysis. This process, where elements from brainstorming the game idea are structured logically and qualitatively, uses the tools described in Section 3.1 introduction, such as CLDs, flow charts, and RBPs. An analysis component does exist in traditional game design, but only proven system analysis tools provide a solid scientific fundament on which to build the game.

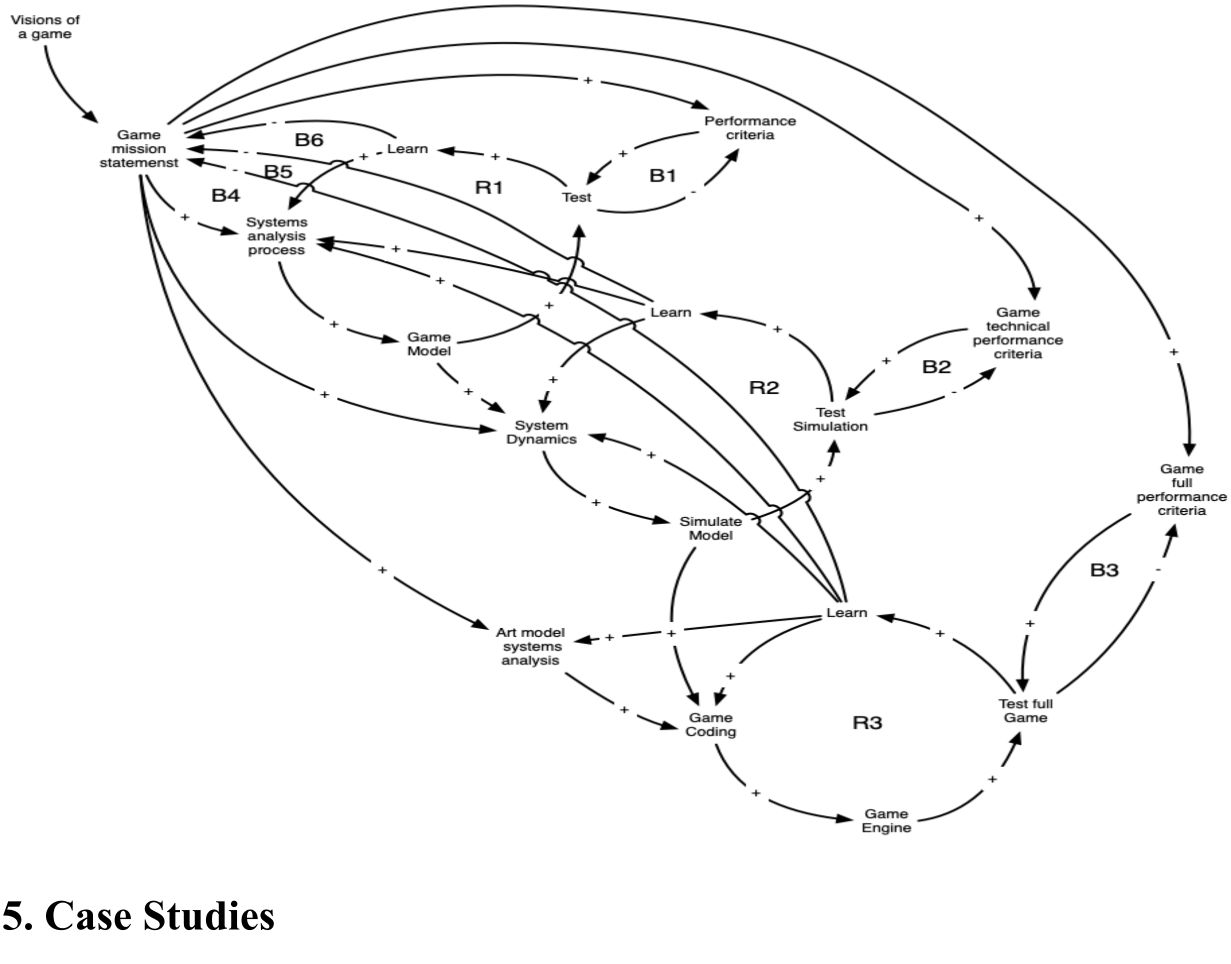
The third step, system dynamics, is where the game is simulated. This process involves designing stock-flow diagrams, finding values for the variables, and implementing the model in the simulation software. The simulation is then tested using scenario-based datasets. When it works as intended, the code is exported into the game engine (see more details on this below), creating a prototype of the gameplay. This step, too, happens in the preproduction phase. Assigning numbers can be an exacting process; often the developers must make qualified guesses based on the system analysis of the game idea. However, the process forces them to dig deeply into their game idea and game design documents to develop a quantitative understanding of how their game systems should work.

The fourth step is to make everything work within the game engine and, in the production phase, to add graphics. As this step is not very different from traditional game design, it will not be dealt with here.

Figure 5 illustrates this process in more detail. Note that all steps have feedback loops on all levels, which can entail revisiting and making changes at any level of the design. It is important that all steps are continually updated throughout the design process for the system analysis to always reflect the current version of the game engine.

Figure 5

The Game Design Process Using System Thinking



5. Case Studies

The game design and system thinking course at The Game School is centred on system thinking and includes game balancing and game design assignments. As the course is part of a broader gaming curriculum, as educators we want students to see the relevance of system thinking to advanced game design. During the course, students solve four assignments of increasing difficulty out of a total pool of 15 assignments/tasks. The first assignment is solved individually, but the next assignments are solved in groups that increase in size with each subsequent assignment. Since there are many different tasks and the students do not solve all of them, all finished assignments are presented in class so all students can discuss them and learn from them.

The task development follows the model given in the theory chapter. For the system dynamics simulations, the students use ISEE Systems' Stella Architect. When they have finished their designs and simulated a few scenarios, they present them to the class and teachers for feedback. This process usually forces them to correct their designs and run new simulations. The students then submit their revised designs along with a case report in which they present the final design and discuss the development process, the different scenarios they explored, and their progress and challenges along the way. As the assignments are folder-based, students can alter and improve their case solutions throughout the semester based on other cases they have solved. All these iterations force them to dig deeply into using system thinking in their game designs.

The next section of this paper presents two examples and two possible solutions to game design cases or assignments. The examples are from level 2 and level 4 in the class, where students are working in groups and cases have increased in difficulty. The first examples cover only one system, but in the second, several systems must be connected and work together in-game.

5.1 Example 1. Tank, Healer, Damage Dealer

The first example is a dungeon crawler game where the objective is to simulate different combinations of healers, tanks, and damage dealers. The assignment can be solved by making just one system and it serves well as an assignment at a moderately difficult level. It is open-ended in the sense that gameplay values – see Table X – can be interpreted in several ways. Aggro is one such example. Aggro is a term that describes how an NPC determines which player to attack first in an encounter. A player that has lost aggro during gameplay often cannot withstand attacks from the NPC, meaning that the game is often lost. In this case, different choices in how aggro is implemented will result in very different solutions – and the first time students were assigned this problem, every group interpreted and implemented aggro differently. The example in Table X shows only the simplest solution, based on World of Warcraft's aggro implementation, where only the player with the highest aggro receives damage. A dungeon party of 10 players (the maximum amount) is about to encounter the boss fight. Each of the classes have the following statistics.

Table X

Table Title Goes Here

	Tank	Healer	Damage Dealer
Damage rate	10	0	30
Healing rate	0	30	0
Health	30	10	20
Aggro rate	30	10	20
Damage reduction*	25%	0%	10%

The asterisk indicates the percentage of damage subtracted from the total damage dealt to the character. The boss has the following statistics.

Table Y

Table Title Goes Here

Damage	100

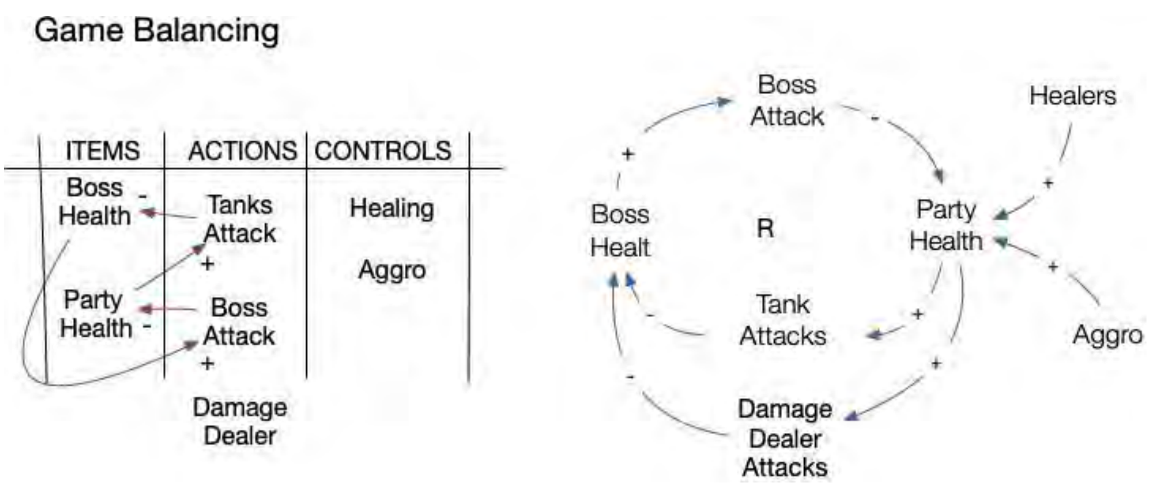
Health	1000
Healing rate	0

The assignment gives the students the following list of tasks:

1. Perform system analysis and create CLDs, flow charts and RBP for the system
2. Create simulations of the current balancing of the dungeon
3. Assign a limited random range of damage and healing instead of the constant shown in the table. Run the simulation 100 times and tweak the ranges to bring the probability for survival to 60%-80%
4. Create a more difficult version of the dungeon, where the survival rate is 10%-30%, varying the number of players and the boss damage reduction

Figure 7 shows the stage of system analysis at which students have sorted the variables in a table with items, actions, and controls and have developed this further into a CLD. Figure 8 shows how the simulation can be solved. In this example, there are two stocks (the square boxes with curves inside) – player health, which represent party health for all players, and boss health – the enemy's health. Inflows in both stocks represent how much they heal per time unit, and outflow represents how much health they lose per time unit. Player health also has another inflow that represents how much players heal during the fight, which again is dependent on the number of Healers in the party and the healing resources available. Party DPS calculates how much damage the Boss receives and is dependent on the number of players in each class and random factors set in the damage dealer DPS, healer DPS, and tank DPS. The damage dealt by the Boss is also random and is set in the boss DPS and multiplied with a value set in damage reduction (Dred). Damage reduction moderates the damage before it is applied to parties or the boss.

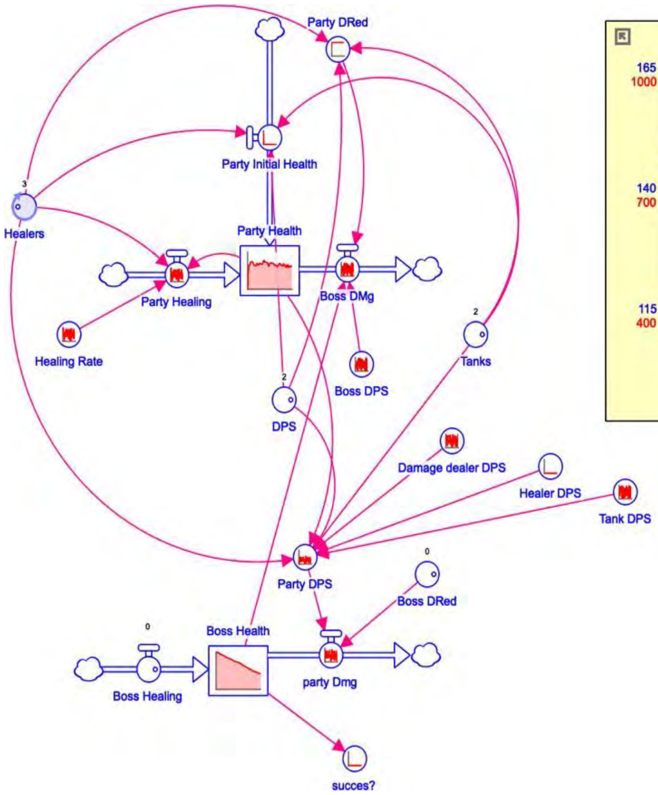
Figure 7

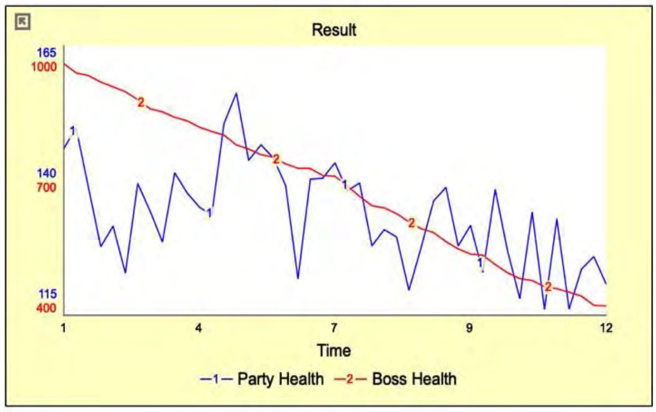


Note: The figure shows the CLD (to the left) and the resources used in the CLD sorted by items, actions, and controls.

Figure 8

The Stella System Dynamic Simulation





Note: Results from this example will be further discussed in 5.3 and the discussion section (part 6).

5.2 Example 2. Trade Route Logistics (Level 3 or 4)

The second example is a part of a bigger simulation game with several systems that must be connected. The game's goal is maximizing profit from cotton production for a whole town (Cottonbog) and choosing the right combination of trade routes to minimize loss from bandit attacks. In this example, town traders sell cotton to three nearby villages, Slagtown, Coalpeak, and Goldford, and bring back gold, food, and other commodities. Currently, the town of Cottonbog has 35 active traders. The road to Slagtown is short and safe. It is guarded by the local garrison and has a very low chance of bandit raids. The road to Coalpeak is long and tiresome, but relatively safe. The road to Goldford is medium in length but there is a greater chance of raids. Each route presents different task values, outlined in Table Z.

Table Z

Task Values and Trade Logistics in Cottonbog Scenario

Town	Travel time (weeks)	Raid chance	Toll price per trip	Trader income per week	Trader income per trip	Gold earned per kg cotton
Slagtown	2	low	500	1000	1500	109.5
Coalpeak	8	medium	0	1000	8000	584
Goldford	4	high	0	1500	6000	438

The assignment asks the students to do the following:

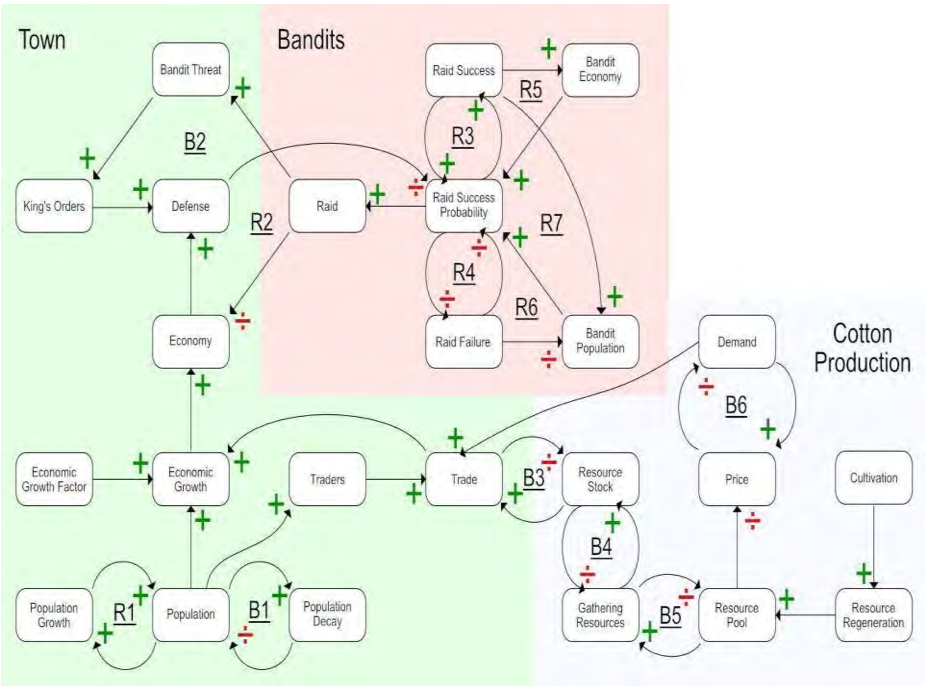
1. Create CLDs, flow charts, and RBPs
2. Create a model of Cottonbog's population, economy dynamics and trading, and cotton production. Run the simulation a number of times (e.g., 100) to establish "the best" trading routes
3. Introduce the bandits as a new system and connect it to the Cottonbog system
4. The king wants to make the roads to Coalpeak and Goldford safer and introduces a tax. Simulate

the tax rate and time needed to do this

5. The king wants to further increase the trade in the region and decrees that Cottonbog needs to trade for 10000 gold per week. Find a solution to this problem and implement it in the system. In this task, the students create a model with three systems: one for cotton production, one for the bandits that raid the traders, and one for Cottonbog. Two things flow through the system: cotton and gold.

The CLD and flow charts are shown in figures 9 and 10. The Stella simulation contains a standard population system for Cottonbog that is dependent on cotton production and trade.

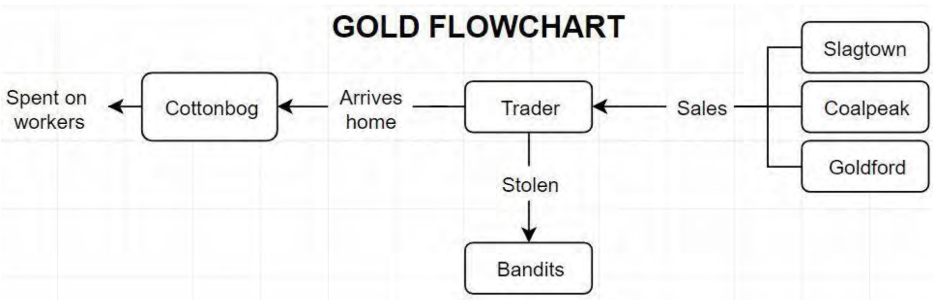
Figure 9 *CLD From System Analysis and Three Interconnected Systems*

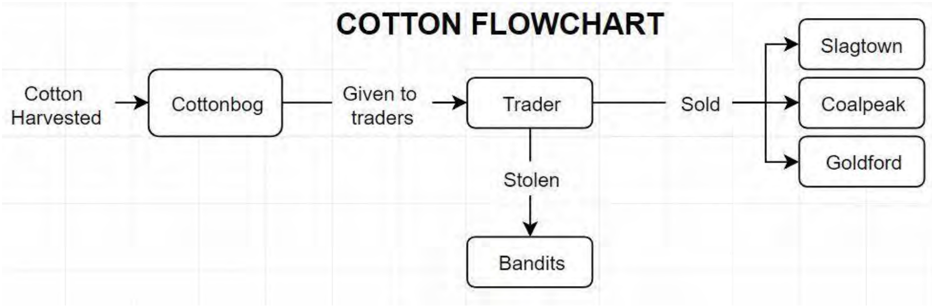


Note: Green (Town) shows the Cottonbog system, pink shows the bandits, and light blue shows cotton production. For example, bandit raids take money from Cottonbog and, consequently, the town's defense is weakened. An increase in raids also creates a bigger threat and increases the likelihood the king will give orders to secure the road.

Figure 10

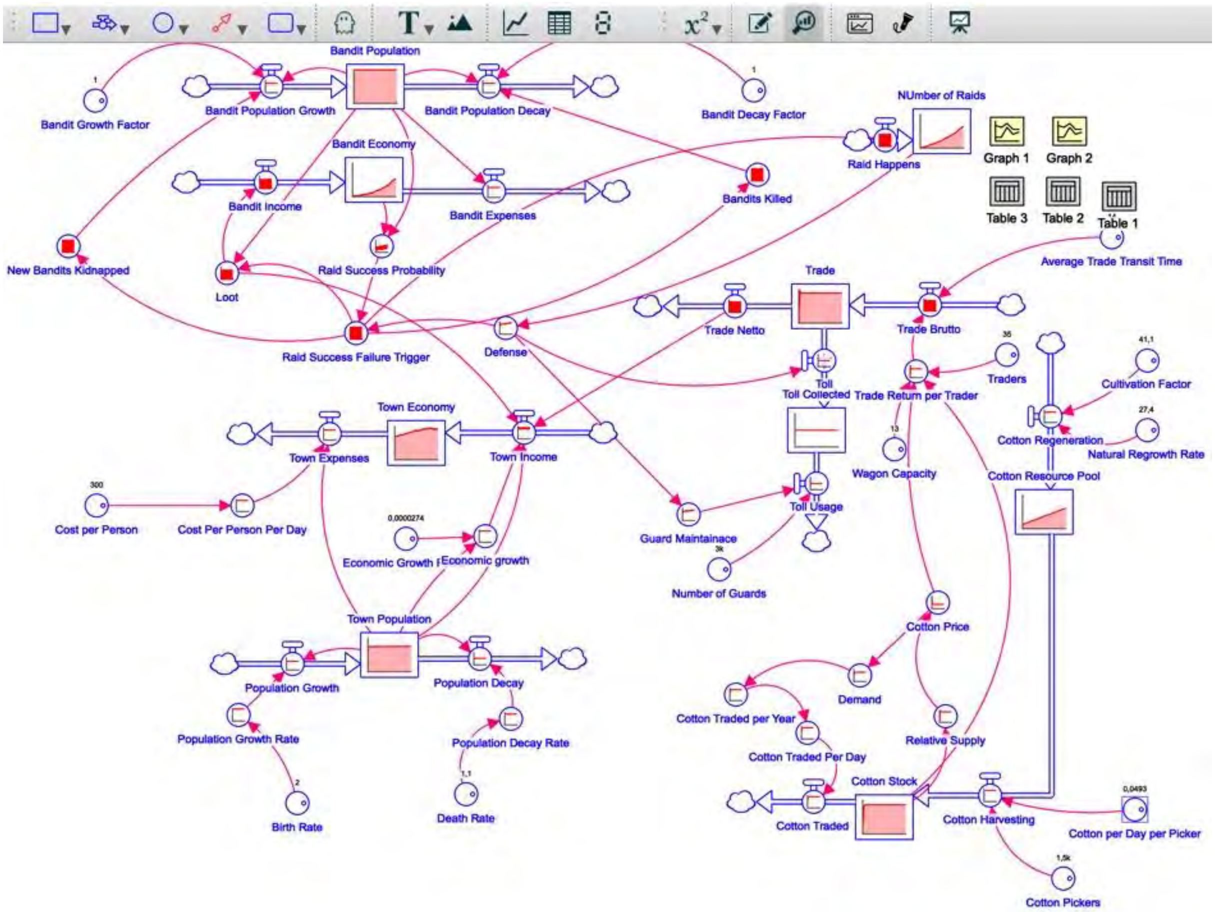
Resource Flowcharts From System Analysis





Note: The flow charts show what flows through the system – namely, cotton and gold.

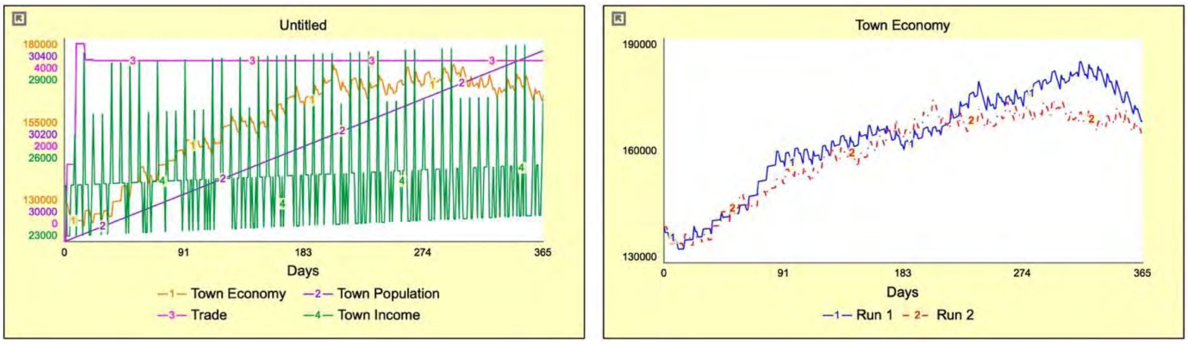
The Bandits system (on the upper left in the Stella model in Figure 11) differs from the Bandit population, and the economy improves only when a raid is successful. However, raids are risky because players often lose bandits during a raid. The number of raids and the success rate are random functions influenced by the number of bandits, the danger of the route, and a defense factor. This assignment is quite large and in many ways resembles how a game idea is developed and its difficulty balanced in creating good gameplay experience. As with the other example, the system analysis students apply is all about understanding how the game idea can be divided into systems, what each system will do, its boundaries, and how the systems can be connected. RBP allows students to analytically predict how the systems will behave over time, and what to expect from the simulations. The model, and the system dynamics simulation, make it possible to simulate different scenarios to build an understanding of how a game behaves under different conditions. The simulations enable the game designer to see how the game design functions when it is played by showing how player agency affects every system in the game and how each system reacts, and if these reactions create the desired gameplay experiences.



Note: In all accumulators (square boxes), curves indicate how the accumulator develops over time.

Figure 11

Stella Model of all Three Systems



Note: The Stella model contains all the three systems in the CLD. The yellow diagrams show system parameters and how they develop over time.

5.3 Critical Evaluation of Student Performance

The system thinking used in the first assignment highlights the usefulness in balancing the game mechanics in a dungeon crawler where different players, with different attributes, encounter an NCP. These encounters can be advanced, with many different systems connected to different player classes, abilities, items, and equipment. Because the simulations in Stella are well-suited to simulating the full range of variables, designers can scale up and down scenarios according to how advanced a game they want to run. This can be done in the initial game design process in the preproduction phase, or in the production or postproduction phase after feedback from playtesting.

In general, system analysis contributes to the students' general understanding of both system thinking and game design. Further, system dynamics and its simulations force students to reflect on how the values in a given task influence gameplay and even design (such as the aggro in example 1, which can be defined in many ways). It also helps students balance the game and provides mathematical equations they can code and use in the game engine. Value of system thinking of course increases with the complexity of the systems.

The three systems in example 2 train students to see how systems influence each other. This assignment extends students' thinking around system dynamics because it involves two completely different societies (town and bandits) that behave differently yet influence each other.

In our experience as educators, introducing system thinking into the game development process forces students to consider more thoroughly what type of gameplay experiences they want in their games and how to implement them, and how a game behaves and changes when parameters are varied. In addition, forcing students to update both system analysis and system dynamics each time changes in the game engine are implemented gives them a complete set of tools to explain their game to others, such as other team members, investors, and sub-contractors.

6. Discussion and Lessons Learned

In the two assignments presented in this article, system thinking is used as a proven academic tool to further understand and develop advanced game design. System analysis and system dynamics methodologies allow students to get a good overview of the systems underpinning their game concept. With these tools, the game concept can be broken into different systems, making it easier to see how the game difficulties will fluctuate and behave, how different systems will interact with each other, and if the gameplay loops create the desired player experience. The students can therefore tweak different aspects of the game idea until it approaches the concept they envisioned. They can take the data that their system thinking as a whole has produced and improve the game engine prototype they will playtest, rather than iterate on the game engine in the production phase.

By using system dynamics simulations, students can explore a scenario many times while varying the parameters without developing the game – and gain a more thorough and detailed understanding of how different solutions behave in their games. Over our three years of instruction in game design using system thinking, the class discussions, presentations, and reports have shown that these methodologies provide students a more comprehensive understanding of the problems involved in game design. In the first example, we found this evident in the students' deeper and more thorough discussion and understanding of aggro, healing, and health in a boss encounter and the flows and outcomes it produces. As simulations run in Stella can be stored, students can examine prior results, and therefore can more easily balance the solution they choose.

Class reports also indicate that students using system thinking tools developed a more comprehensive understanding of game balance, as their simulations showed them how small changes in game parameters can have a sizeable impact on game systems -- and ultimately affect player experience. This leads us to the conclusion that the system dynamics simulations students used in their assignments resulted in their more thoroughly understanding how to balance systems in a game and make meaningful gameplay encounters. As a result, students acquired an overall fuller understanding of how games work. Students do still need a basic understanding of traditional game design as a foundation, but system dynamics helps them dive deeper into the game design. The best student groups took the assignment further than needed and experimented by making "smarter healers" through implementing a more sophisticated aggro system. As educators, we learned that it is important to make the cases or assignments open-ended so students can take them as far as they can.

The second example presented above teaches students the same lessons and knowledge as the first, but with a more complicated game design; in this case, they must calibrate – and understand – three systems. While this second assignment is small when compared to a fully developed commercial game, approaching it through systems thinking can help provide an overview of the game (or parts of the game) that make it easier for students to grasp the overall behaviour of the gameplay experience without having to consider every detail of the game at once (Gee, 2007).

Normally, game design calls for a lot of playtesting. Many game design books assert that playtesting by professionals and amateurs alike is the best way for developer to get information about a game's different components (for example, gameplay and meaningful play). In our classroom assignments, system dynamic simulations saved time otherwise spent on playtesting by letting students "test" and document the balance of the game prior to user tests.

System dynamic simulations in Stella also give game designers differential equation-based pseudo code that can be implemented into the game engine with little effort. This, too, makes it easier for the game designers to quickly build a more balanced core version of the game in a game engine that can be playtested.

In general, all students enjoyed the assignments and found them relevant for their education and their game designs (explicitly pointed out by the in the mid-way evaluation). Adams and Dormans (2012) argue that game mechanics, meaning the actions and effects a player can make in a game, are harder to talk about than any other aspect of game design. They are difficult to prototype and test because they often are systemic in nature, meaning the actions, resources, and reactions must be coded, or the designer must use a spreadsheet to get a feel of how each function and relates to the other. This process is not particularly fast or intuitive and neither does it give designers a good understanding of how the game mechanics interact or scale with rising difficulty and increasing challenges. Using system thinking in game design helps solve these difficulties.

So, does system thinking improve the understanding of game design? In general, we think it does. System analysis helps designers understand how cause and effect and feedback loops are connected in their game, and RBP allows them to reflect on how their systems and loops behave over time. System dynamics greatly improves the ability to conduct thorough, detailed testing and balancing of the game before playtesting is carried out. It also makes it possible for the developer to transfer the simulations over to the game engine and execute them whenever the game requires, down to each frame.

7. Conclusion

System thinking is a well-tested and proven tool for analysing, understanding, and testing systems. As Salen and Zimmermann (2004), Gee (2007), and Adams (2014) have pointed out, games are systems; system thinking and game design would appear to be a good match. As game design educators, we have discovered that system thinking encourages students to dig deeper into their designs, which always is a good thing. Further use and testing will, we believe, reveal new areas where system thinking can be used in game design. As we discuss, game development can also be a great learning environment for traditional school subjects. All in all, we can recommend using system thinking for game design and believe it could become the predominant methodology in future game and gamification designs.

8 References

Adams, E. (2014). Fundamentals of game design (3 ed.). Berkeley, Calif.: New Riders.

Adams,E.and Dormans,J.(2012).Gamemechanics:advancedgamedesign.Berkeley,Calif:New Riders.

Blake, B. (2017). The Future of Gaming: Free DLCs and Paid Microtransactions. Retrieved from https://gamerant.com/future-gaming-dlc-microtransactions/

Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

Donovan, T. (2010). *Replay: The history of video games*. East Sussex, England: Yellow Ant.

Doyle, N. (2015). Why more open world games is a bad thing. Retrieved from http://gamemoir.com/more-open-world-games-bad-thing/

Egenfeldt-Nielsen, S., Smith, J. H., & Tosca, S. P. (2008). Understanding video games: the essential introduction. New York:Routledge.

Gee, J.P. 2007. Good video games and good learning: Collected essays on video games, learning and literacy. New York: Peter Lang International Academic Publishers.

Haraldsson, H.V. 2004. Introduction to systems thinking and causal loops diagrams. Reports in ecology and environmental engineering. Report 1:2004. 49pp

Handbook of action research: participative inquiry and practice. Reason, Peter, Bradbury, Hilary. London: SAGE. 2001. ISBN 978-0761966456. OCLC 50303325.

Kapp,K.M.2012.TheGamificationofLearningandInstruction:Game-BasedMethodsand Strategies for Training and Education. New York; Wiley.

Kent, S. L. (2001). The ultimate history of video games: from Pong to Pokémon and beyond--the story behind the craze that touched our lives and changed the world (1st ed. ed.). Roseville, Calif.: Prima.

Krapp, A. 2002. Structural and dynamic aspects of interest development: theoretical considerations from an ontogenetic perspective. *Learning and Instruction, 12*(4), 383-409. doi:10.1016/S0959-4752(01)00011-1

Lave, J., and Wenger, E. 1991. *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

Maastricht University 2013. *Problem based learning preparatory website.* Retrieved from http://www.umpblprep.nl/

Mora-Cantallops, M., and Sicilia, M.-Á. (2018). MOBA games: A literature review. *Entertainment Computing, 26*, 128-138. doi:10.1016/j.entcom.2018.02.005

Mäyrä, F. 2008. *An introduction to game studies: Games in culture*. Los Angeles, Calif.: Sage.

Nordby, A., and Karlsen, S. 2014. Teaching 'hardcore science' to arts and design students: Reflections on the development of a basic programming course. *InFormation - Nordic Journal of Art and Research* 2014; Volume 3.(2) p. 129-142

Peterson, S. 2016 The Marketing Power Of DLC. Retrieved from https://www.alistdaily.com/digital/marketing-power-dlc/

Pramath.2016HowDoDevelopersCreateMassiveOpenWorldsForExploration?Retrieved from https://gamingbolt.com/how-do-developers-create-massive-open-worlds-for-exploration

Rouse, R. (2005). *Game design: theory & practice*. Plano, Tex.: Wordware publishing.

Salen,K.,and Zimmerman,E.(2004).Rulesofplay:Gamedesign fundamentals.Cambridge,Mass.:MIT Press.

Schell, J. (2008). *The art of game design: A book of lenses*. Amsterdam: Elsevier.

Schiefele, U. (1991). Interest, learning and motivation. Educational Psychologist, 26(3 – 4), 299 – 323.

Tekinbas,K.S.,Gresalfi,M.,Peppler,K.,Santo,R.2014.*Gamingthesystem:designingwith Gamestar mechanic*. London: The MIT Press.

Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

Wallach, O. (2019). Who Builds Video Game Wikis? Retrieved from https://www.fanbyte.com/features/who-builds-video-game-wikis/

4

Article 4:
System Thinking in Gamification

Anders Nordby, Håvard Vibeto, Sophie Mobbs, Harald U. Sverdrup (2023)
SN Computer Science

Status:

Will be published in SN Computer Science in 2023.

https://www.springer.com/journal/42979?gclid=Cj0KCQiA7bucBhCeARIsAIOwr-_TwKxd53VI5TeV2323TInALaBMzFMPnWMWx7J1KHxPrG1S93mWOhQaAocREALw_wcB

System Thinking in Gamification

Anders Nordby[1], Håvard Vibeto, Sophie Mobbs, Harald U. Sverdrup
Department of Game Development – The Game School Inland Norway University of Applied Sciences, Holsetgaten 31, NO-2301 Hamar, NORWAY

[1]anders.nordby@inn.no

Abstract

Introduction: People spend a lot of time and energy playing videogames (Kapp, 2012), and as a result gamification has grown from a buzzword into a discipline. Since 2012, the authors have experimented with System Thinking as a methodology for developing gamification and will present examples in this article.

Scope and objectives: The primary objectives are to study how **System Thinking** can be used to understand, design, and develop **Gamifications**.

Methodology: This is an observational case study that gives examples of how students (i) use System Thinking to understand and clarify the gamification case using system analysis and (ii) use system dynamics to simulate cases and predict user responses. Students begin system analysis once the gamification idea is developed and their goals and the case parameters are established, and it includes making Casual Loop Diagrams, Flow Charts and Reference Behavior Patterns. Students then find and experiment with numerical data for the case and use system dynamics to simulate the gamification and predict the user results. The **pedagogy** is problem-based and grounded in traditional problem based learning and situated learning.

Discussion and Results: This article shows how System Thinking allows students and professionals to develop a deeper and more tangible understanding of the research materials and presumptions they have when engaging any given gamification scenario. System Thinking also provides tools to test research material and hypotheses in a more structured, manageable, and palpable way.

Conclusion: Although we have discovered several ways system thinking can benefit gamification design, the research has also revealed new areas where system thinking could be explored further.

Keywords, Gamification, Serious Games, Game Development, Games and Learning, Pedagogy, System Thinking.

1. Introduction

Gamification stems from the idea that game mechanics and game thinking can be utilized for goals other than creating entertainment and fun. Games employ structures, tasks and rewards that can be used outside games to change human behavior and motivation and to help reach non-trivial goals. Gamification can create new ways to engage users, target new user groups, and motivate them to achieve goals they have or did not know they had (Burke, 2014). The goal of gamification is to promote learning, engagement, motivation and change behavior in a positive way (Kapp, 2012). The idea of gamification has been fueled by the proliferation of smartphones, social media, the internet of things and the popularity of videogames.

The culture of the gamer has now permeated every age group; people spend a lot of time playing videogames, on many different devices (Kapp, 2012) and the vocabulary of videogames is familiar to a big portion of the population. Yet gamification is a relatively new field of study and as a discipline can be strengthened with scientific methods that can help design and test the development of different gamifications applications.

Designers must be able to predict how the user will react to and is motivated by the different components of the gamification. As gamification uses game-based mechanics, thinking, and aesthetics, the user must engage with a system that is going to provoke extrinsic rewards and lasting intrinsic motivation that will result in some kind quantifiable outcome (Kapp, 2012).

System thinking can help predict and simulate the multitude of outcomes and reactions the gamification can create. It can give more insights into individual cases, and to some extent it can even estimate how motivation and learning will develop when someone is using the gamification. It can also help target the type of motivation that needs to be stimulated to reach an application's goals.

1.1 Gamification

Using different engagement models and reward systems to create learning, motivation and behavior changes is nothing new – it has been done for decades in schools, sports, businesses, military, and the advertising industry. However, unlike these earlier applications, gamification fuses game thinking, game culture, and digital technology and new ways of learning, engagement and motivation emerge. The word gamification is often considered to have been coined by Nick Pelling in 2002 (Burke, 2014, p.5). But it was Jane McGonigal's 2011 book, *Reality is broken*, that popularized the idea of gamification as a tool (McGonigal 2011). So, what defines gamification? One short definition is that gamification is the use of game design elements in non-game contexts (Deterding, Dixon, Khaled & Nacke, 2011). Another more elaborate definition asserts that "Gamification is using game- based mechanics, aesthetics and game thinking to engage people, motivate action, promote learning, and solve problems" (Kapp 2012, p.10). The goal of a gamification is to make activities fun and engaging by introducing game mechanics, but also to use the engagement people have for games and play in areas other than entertainment.

In gamification one can motivate through rewards (extrinsic motivation), but also through fostering inner motivation that can change the user's behavior permanently (intrinsic motivation) and enable the user to reach goals she could not reach on her own (Kapp 2012, p.1-25). As a growing discipline, gamification uses knowledge and research from other fields, including game design, pedagogy and didactics, psychology, and interaction design. But as a new discipline, it also needs to develop methods to design gamification application. System Thinking can be a useful tool for developing the basic knowledge of game design – understanding the different components of a game as well as the formal theories and definitions of games – that underpins successful gamification design, and thus facilitates the creation of games that give players the desired game experience (Mäyrä, 2008, pp. 36-37).

1.2 State of the art in Gamification Design

Gamification design still often builds on a traditional game design development process, and although game design has moved beyond the days of simply adding rewards to induce player motivation, the design process is often linear (fig 0 below). A gamification product often begins with the desire to enhance a service or product in order to reach a specific objective by the owners. Usually, understanding how to reach these objectives comes from analyzing user behavior and wishe. This in turn informs how the developers translate these objectives into appropriate game-design elements that are compiled into the final gamification product or service (Blohm & Leimeister, 2013). Further, even though pedagogics and psychology may be part of the design process neither is usually woven into the gamification design with any scientifically proven methodology. We address this with our project. We use System Thinking to weave game design, psychology, pedagogics and technology together through System Analysis and simulation and show how feedback loops are an essential part of the design process. The innovation in our project is, in other words, not in the invention of new tools, but in the use of existing tools in new ways.

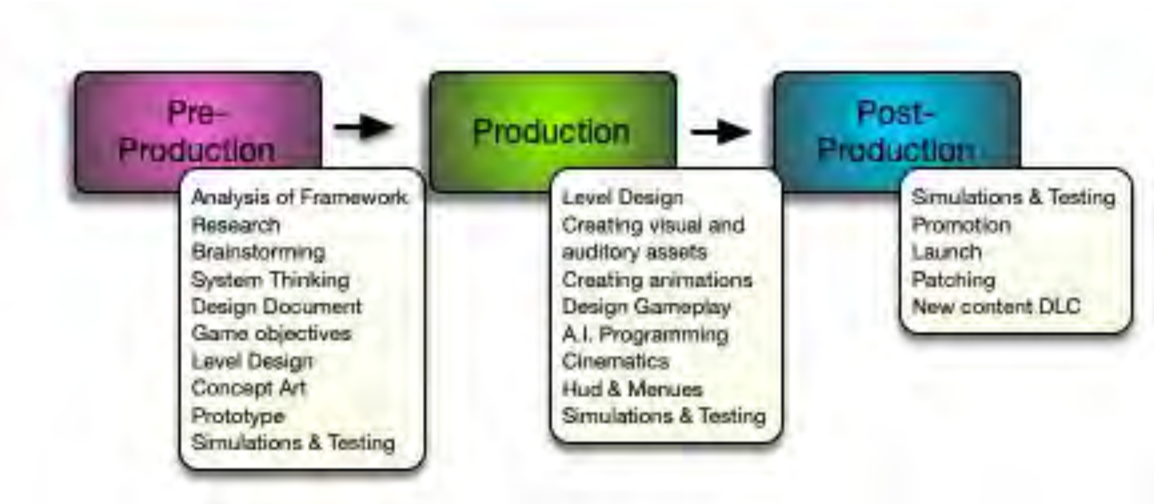


Fig. 0. Traditional Game design.

1.3 Knowledge gaps and hypothesis

We believe there are several knowledge gaps in the field of gamification development. Firstly, we believe that the design of gamification can be strengthened scientifically by introducing well-known and proven scientific methodologies into the preproduction phase. Secondly, connecting gamification to motivational psychology can strengthen the credibility of the design and its goals in this phase (Kapp, 2012). Thirdly, we hope that being able to simulate the use and interaction of the gamification before starting the production phase can save development time and money.

2. Scope and Objectives

This article discusses how we use System Thinking in several ways in connection to gamification. Two student projects from course that focuses on the preproduction phase of gamification development will be given as examples. In this course, students do not create a finished gamification application, but rather a prototype based on psychology, system thinking analysis and dynamics, and specific design choices. The article will therefore not discuss the production and postproduction phases of the gamification app in any depth.

2.1 Research questions

The article discusses how System Thinking can be used to design, develop, and document the process of gamification. The research question is: "How can System Thinking be used to design**,** develop and document the process of **gamification**"?

2.2 Data collection and analysis

The research methodology used in student projects is Action Research. Both the research and the data collection are situational, practical, systematic, and cyclical, and the result of each cycle is constructivist in its implementation (Efron & Ravid, 2020). The collected data are student assignments and reports, meeting resumes, teacher notes from student presentations, and so on. These data are analyzed and discussed both during the course and afterwards and form the basis for immediate changes or justifications of the course design and implementation. Whenever possible, system analysis is used to clarify and make sense of the results. The use of system thinking and action research has over the years spread into several courses and topics/disciplines in the game school such as gamification, game design, programming, and others and today is a form of collaborative action research.

3. Theory

3.1 System Thinking

System Thinking is a concept and language that helps to substantiate and explore how causal relationships and feedback work within a system. It has two components: system analyses and system dynamics simulations. System analysis is a qualitative way of describing connections, casualties, and feedback in a system, while system dynamics is a numerical simulation of the system. System analysis includes group modelling – stakeholders' interests and connections are mapped by finding shared questions for the problem, drawing flowcharts and making a Causal Loop Diagram (CLD). The CLD is the most important tool and shows causes, effects and feedback in the system and how they are connected. An example of a CLD is shown in Figure 1.

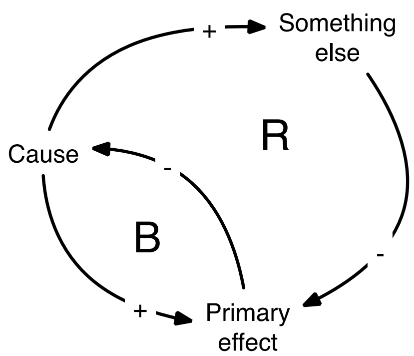


Figure 1. An example of a causal loop diagram (CLD).

This drawing (Fig 1.) shows a simple CLD in which a cause produces an effect. An arrow links the cause to the effect. The plus sign (+) indicates that the cause increases the effect. The effect provides feedback on the cause, which is illustrated with an arrow leading from the effect back to the cause. The minus sign (-) here indicates that more of the effect will weaken the cause. The system has two loops: balancing (B) and reinforcing (R).

System dynamics is a numeric simulation of the system analysis results. In our project, we use STELLA Architect for the system dynamics simulation. Figure 2 shows the simulation of the CLD in Fig 1.

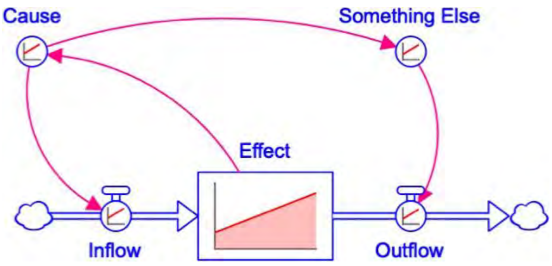


Figure 2. Realization of the causal loop diagram.

The effect influences the inflow through the feedback loop to the cause. The cause also influences the value of the effect through the outflow. More about systems System Thinking can be found in Senge (1990), Sterman (2000), Haraldsson and Sverdrup (2004) and Sverdrup and Svensson (2002a,b).

3.2. Gamification psychology

Gamification is about creating motivation. It normally begins with extrinsic motivation, while having with the goal of creating permanent intrinsic motivation. Proven and tested psychological models exist that can be used to better understand how extrinsic and intrinsic motivation are created – the student examples discussed here focus on the ARCS model and the Self Determination Theory (SDT) (Kapp, 2012, p.54), but also occasionally uses other theories as Malone (Malone, 1981, p.27) and Lepper (Kapp, 2012, p.57). As the ARCS and SDT models are the most widely used in the student examples, we will briefly introduce them below, building on Kapp's explanation of the model (2012).

The ARCS model is a four-factor model developed by John Keller (Keller, 1987). ARCS stands for Attention, Relevance, Confidence, and Satisfaction. Many of these elements have applications for gamification and motivation and can be applied to various aspects of gamification and game-based learning. *Attention* addresses gaining a learner's interest in the content through various means. Perceptual arousal draws learner attention through specific, relatable examples. Inquiry arousal stimulates curiosity through questions or challenges posed to the learner. Role play or hands-on experience also fall under this rubric. *Relevance* can be established by using goal orientation that describes how the goal will help the learner by illustrating the importance of reaching the goal. (Kapp, 2012). Similarly, the learner gets *Confidence* when they achieve success. *Satisfaction* is about the learning having value and being worth continued effort. That is, the learners should be given the opportunity to successfully apply their new knowledge and skills in a real or simulated setting so they can *see* what they have learned being applied. Additionally, Variability is about varying the delivery method periodically to maintain the learner's attention (Kapp, 2012). Positive encouragement and reinforcement keep them motivated throughout the learning process – it is important to try to tap into the intrinsic and not only the extrinsic motivation of the learners.

According to Self Determination Theory (SDT), human motivation to perform a task or an activity is internally driven as opposed to the externally driven. SDT can be used to describe how and why motivation is facilitated or undermined in diverse human activities such as sports, education, healthcare, work, and religion. The theory also proposes that events and conditions that enhance a person's sense of autonomy and competence support intrinsic motivation and, on the other hand, factors that diminish perceived autonomy or competence undermine intrinsic motivation. SDT focuses on three elements: *autonomy, competence* and *relatedness*.

3.3. Case methodology

In our classes, students constructs their gamification projects within one of three categories: Learning, Health, and Sustainability. The project assignment is divided into three parts: 1) project analyses through System Thinking, 2) design of the gamification prototype and 3) writing the project report. Continued progress on the projects is fostered by requiring students to present their work to the class at regular intervals, and by receiving comments and feedbacks from peers and teachers. The students also regularly write blogs detailing their progress and seek out feedback and advice in online forums and from peers.

Teaching methodology in the gamification course is loosely based on Problem Based Learning (PBL). PBL originates from the novel instruction model implemented in medical education in the late 1960s by Howard Barrows and his colleagues (Barrows, 1980). PBL was originally conceived as a student-centered learning model in which students solved real-world problems in groups. This approach is also our focus; we want our students to learn to solve real-world problems and to practice and methodology they are taught. During this phase they also find and study the theory they need to solve the problem. Working in groups assures that individual students don't get stuck. However, to ease the challenges a bit, and to help students that need more guidance, our course also includes lectures and workshops in psychological concepts relevant to gamification studies (Kapp, 2012).

The gamification development process in the class and the groups is also *situated learning*. That is, the learning is integral to participation in a *community of practice* that has the common goal of learning to create a gamification (Lave & Wenger, 1991), (Wenger, 1998). According to Lave and Wenger, as instructors we can expect that participation and mutual engagement in the class will also trigger interest in related school topics. This interest can then trigger intrinsic motivation.

We also encourage online communication in relevant forums. This allows students to bring their outside-of-university activities and identities into their work, whether this is as bloggers, gamers, or participants of online communities. We think this adds important and relevant informal learning into the classroom.

Finally, our gamification class lets the students *experience learning* – they are not simply being told what to learn, because their gamification development is *learning by doing* (Dewey, 1916). Students are also motivated to study related theory when they need to solve a practical gamification or system thinking challenge, which connects theoretical learning with solving practical tasks. This is le*arning just in time* (Gee, 2007).

4. The case study

Below we will give two examples of the design of a gamification application that the student groups made during our course. The first example focuses largely on the overall situation the gamification tries to improve, while the second example addresses the motivation the gamification tries to create. Each example uses System Thinking differently: the first has a broader focus and a less detailed perspective on motivation, while the second dives directly into various psychological motivation theories.

4.1. Project analyses through System Thinking

The first thing students do after brainstorming a good idea is to undertake system analysis. System analysis starts with establishing a clear and precise question about what their application will do. They then define the project's parameters and objectives. Making CLDs and flow charts helps them define the systems, feedback loops, and flows in their gamification. Finally, they draw Reference Behavior Patterns (RPB), which give them an idea of how the system develops over time.

The next step is to make a simulation of their gamification in a simulation tool, such as STELLA, and collect numerical data for all the variables. This can be a challenging task, and quite often the students won't find precise data and must make qualified guesses based on their research. This requires them to dig deep into their model to develop a thorough understanding of how it all fits together.

The students then design scenarios that simulate system behavior over time, and from that they predict as closely as possible how the gamification will perform given different variables and inputs. This is usually done by simulating the motivation of their users through extrinsic rewards and intrinsic motivation. It is often hard to find quantified data on motivation or learning. In a commercial context, this would be solved by sending questionnaires to users, but the students rarely have time or resources to do this in our educational setting. Instead, they have to make qualified guesses by googling and searching. These predictions are less reliable, but the guiding principles and the methodology for gathering feedback remain valid.

After the system analyses and system dynamics sessions are completed, the students present their findings in the class to get feedback from classmates and teachers.

4.2. Design the gamification application

The next step is to design the gamification application. The design should closely build upon the findings from the system analyses and system dynamics exploration. The app should use relevant game mechanisms, aesthetics, and rewards to influence the users in the desired way. Students should also aim to use different technology platforms, such as wearables, sensors, tracking devices in their design. They must also write a gamification design document, which should describe the gamification concept in every detail. The project requires user experience to be the focus and the design to be player-centric, meaning the user's goals and motivations need to be mapped and designed, so that gameplay motivate the players to achieve their goals and outcomes (Burke, 2014, p.89-125). Students need to explain in detail what goals and outcomes are to be achieved by users, and what changes in behavior, attitudes, or knowledge they hope to elicit from the user. The document also needs to go into detail about the target group and their motivations for change, extrinsic and intrinsic rewards, game mechanics and tasks, reward systems, descriptions of technology, aesthetics, game environment, and budgets. Students make a prototype of the app in Invision or Adobe XD but are also allowed to use game engines such as Unity or Unreal for prototyping. When all is done, they present their results in the class to get feedback.

4.3. Writing the project report

The final assignment of the gamification project is to write a research report. In this report the group discusses the implementation, challenges and how they overcame them. Students must anchor their report in theory discussed in syllabus literature or other books or articles they found necessary in developing the gamification. The report must discuss and reflect upon System Thinking, their own research, the simulation they conducted and the pedagogy and psychology they used. They are expected to discuss and explain why their gamification should work as intended, and why the game design and game tools they chose will work. Last but not least, the report must discuss how the group has worked together to achieve its goals, and outline who was responsible for the various components of the project development, simulation and implementation, and who authored each section of the report.

4.4. Student examples

Student example 1: Galapágos

This first example deals with the Galápagos Islands in Ecuador, where economic growth and tourism are threatening indigenous wildlife. This example is taken from the systems thinking teach-text by Sverdrup et al., (2019). In Galápagos, the key economic resource is the natural environment, the health of which directly affects the Galápagos economy. Preserving nature and native species through ecotourism and wildlife conservation is therefore very important.

Some species on the Galápagos Islands have been overfished to the point of endangerment, which has the potential to harm the islands' tourist industry. The students were asked to find a solution that could protect both the environmental and economic interests of the Galápagos Islands through a gamified app. They developed a game that supports law enforcement efforts to patrol and catch illegal fishers around Galápagos, which is expensive and difficult. Players take pictures of suspicious fishing activities and upload their pictures along with their GPS coordinates.

Figure 3 shows the screens and flow in the application. The left side of the flowchart in fig 4 represents the traditional harvest model. When for example fish are endangered, fishing quotas are usually reduced. Lower quotas and protection restrictions help species conservation and preserve biodiversity, which in turn supports tourism (Schep et al, 2014, pages 27-28). However, as the right side of fig 4 shows, lower quotas also result in pressure from the fishers.

In the CLD shown in Figure 5, red lines represent a declining effect (-) and green lines represent an increasing effect (+). The students point out that the case is a combination of ecosystem and economy: In Galapagos the two most important sources of income are the fishing and the tourism. They are shown in the CLD. We can see that the tourism is reliant on the biodiversity, and the fishing resources is divided into four parts: sea cucumbers, sea cucumber predators, medium fish and sharks. The CLD shows how the economic growth affects the biological biodiversity, which leads to pressure by environmentalists, more legislation, less fishing, and law enforcement. Since most of the harvested marine life are exported to other countries the fishers are reliant on external demand, the students decided to put foreign economy as the main effector of the demand for fish in the CLD. This demand then directly affects both the legal and black-market price which then affects the amount of legal and illegal fishing. Therefore, indirectly most fishing is controlled by foreign demand.

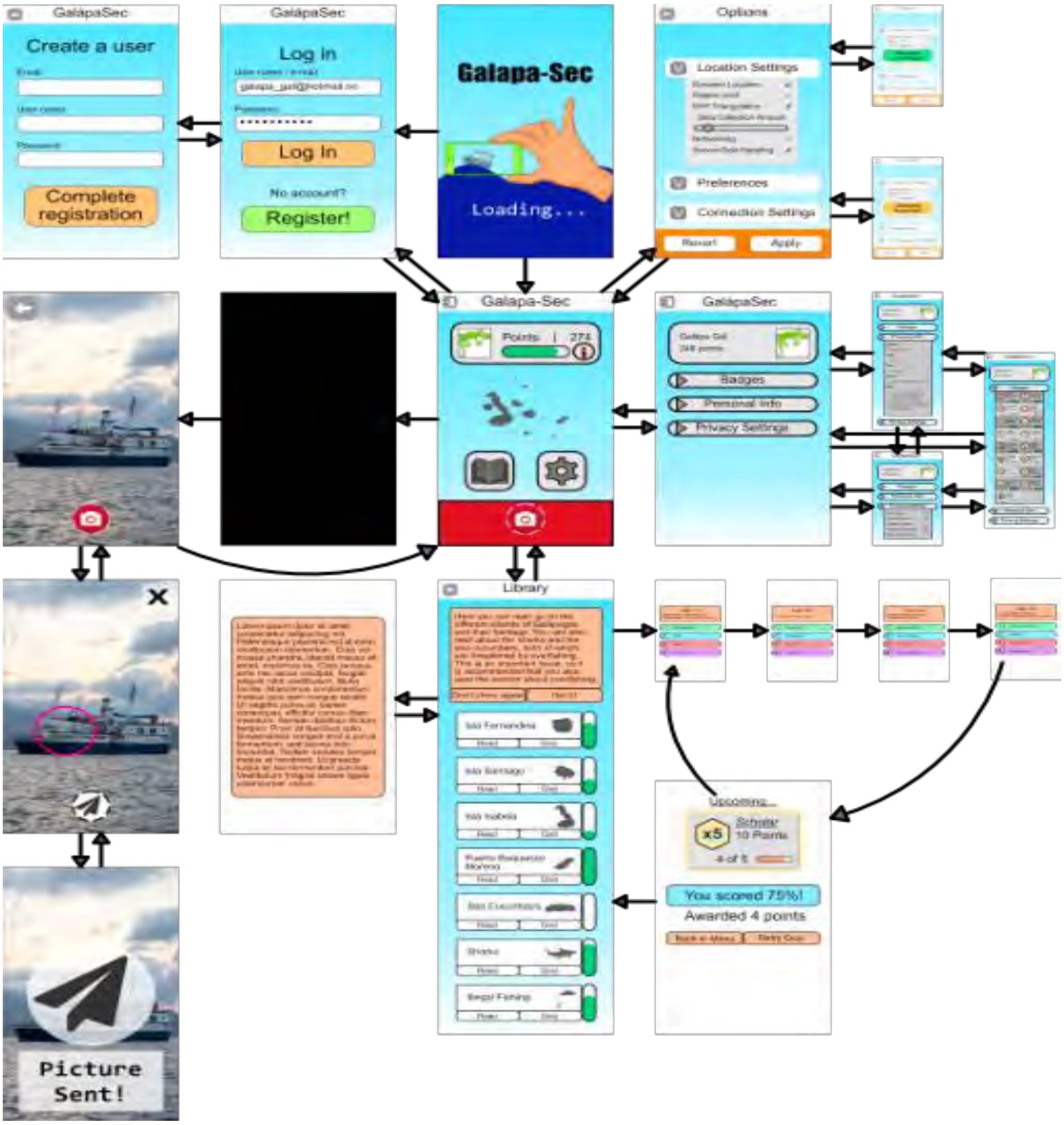


Fig. 3. Phone application flow and screens

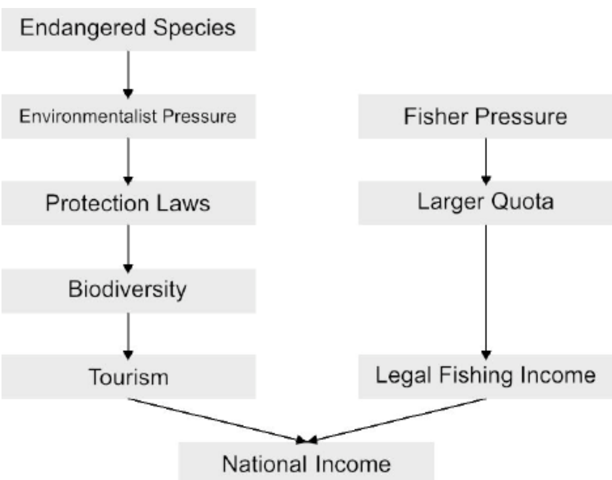


Fig. 4. Event flow chart for the modelling process.

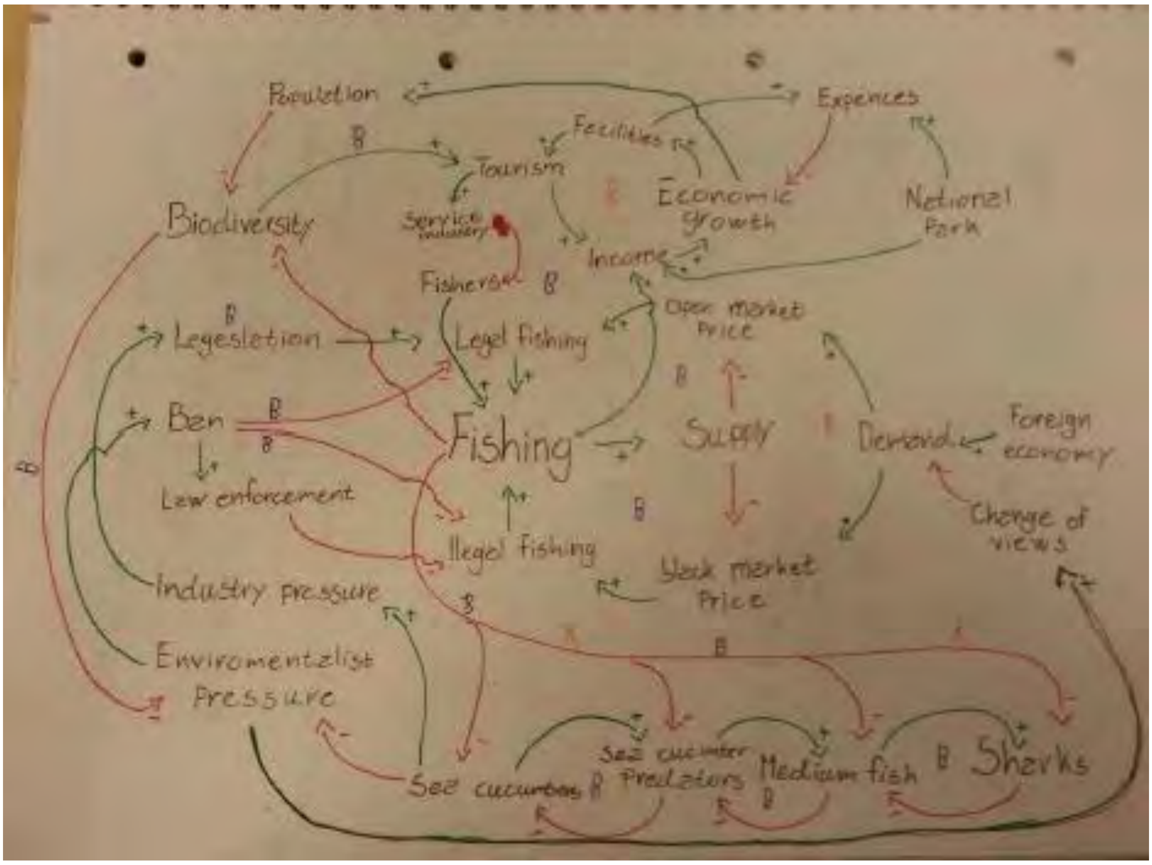


Fig. 5. Casual Loop Diagram (CLD).

The students then made a simulation that covered the whole ecosystem, with links to the dynamics of fishing and to biodiversity and tourism. Unfortunately, it proved impossible to find enough data to run this bigger model, so they made a simpler model that addressed only the economy. In this model, the categories Industry and Activist Pressure are merged as they are only concerned with the strongest influence on Legislation at any time. Foreign Demand is the motivation for Industry, and biodiversity is the motivation for Activists. Legislation then influences Fishing and Tourists and, finally, Economy. This simplified economic model is shown in Figure 6.

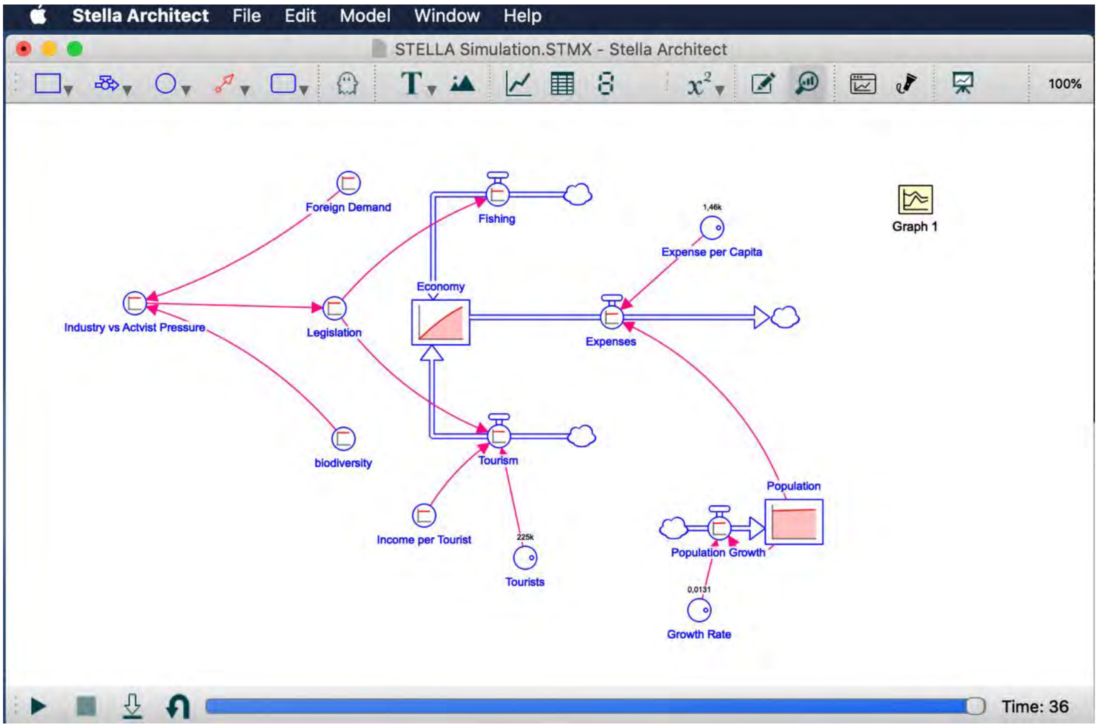
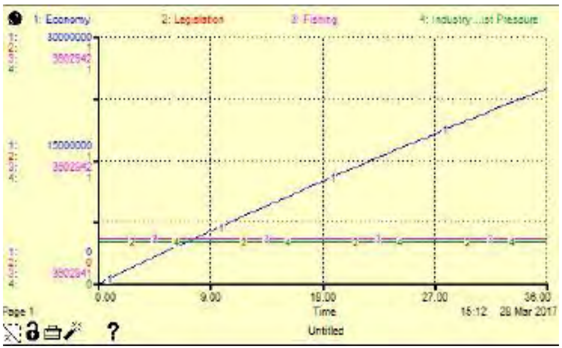
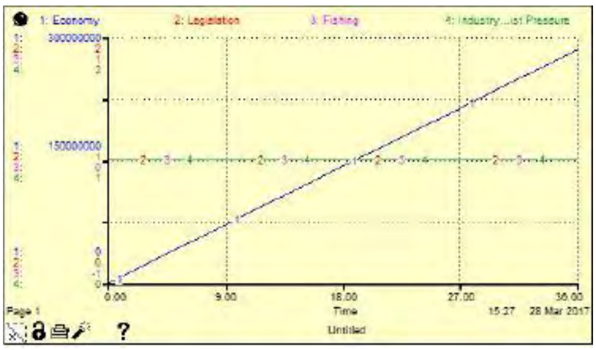


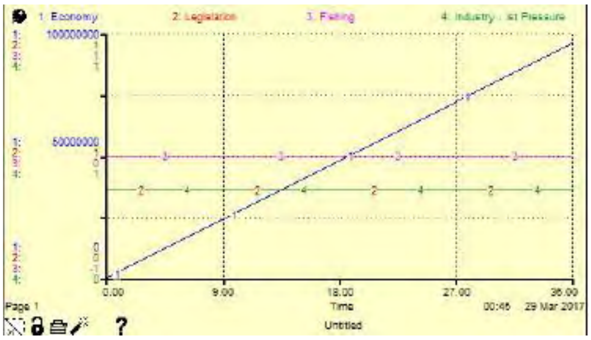
Fig. 6. Stella simulation interface as interpreted by the students.



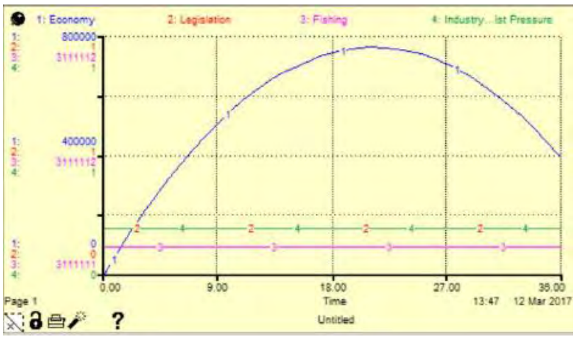
7.1 Fishing continues as usual



7.2 Ban on fishing



7.3 Chinese trends change



7.4 Tourism declines

The students simulated four scenarios (Figs 7.1-7.4):

1. Situation continues as usual
2. Fishing is banned
3. Foreign demand decreases
4. Tourism declines

Although the simulation overall is largely a simplification due to the lack of data, the scenarios show how the Galapágos economy could react to drastic changes in ecology. The students said that working on the models and simulations has provided them with an insight into the dynamics, and thus a better understanding, of the Galapágos economy. The students also explained that although the search for data and the material available didn't provide enough data to run a fully-fledged model, the data set they obtained included descriptions of the dynamics and cases that helped them understand how the key factors in the economy work together. They concluded that the most beneficial alternative path for Galapágos is to disband the legal industry fishing and focus entirely on tourism and other minimally harmful activities.

This leads us to the next example, which is more about the motivation models presented earlier in the article.

Student example 2: Family Manager

This app is aimed at families and focuses on teaching children the importance of positive habits and how to maintain structure in their lives by doing chores. Parents create assignments for the kids to complete which then earn points. Points can be used on rewards, and the family decides together what the tasks and rewards will be. For children, typical tasks are cleaning their room or doing their homework, while a typical reward may be an ice cream or an allowance. In addition, the family can establish collective goals, like a family trip. The app can also teach children about personal finance. Although the students are well aware their app is not a game, they compare the game mechanics in the app with a role-playing game, where the RPG quests are replaced with the individual and family tasks. Figure 8 shows 4 screens from the app.

Figure 8: App Design - Main Page

The students' System Thinking in this example primarily focuses on motivation. They split the motivation into intrinsic and extrinsic motivation, and the app tries as far as possible to trigger intrinsic motivation. These students use the psychological ARCS and the SDT models (Kapp, 2012) but also occasionally also other theories such as Malone's Theory of Intrinsically Motivating Instruction (Malone, 1981) and Lepper's Instructional Design Principles for Intrinsic Motivation (Kapp, 2012).

The CLD from the System Analysis below gives an overview of the system. Nearly all loops address motivation directly, and there are more reinforcing loops than balancing ones. The students see this as a good sign, it points to an increase in motivation over time. Secondly, they point out that intrinsic motivation increases the more the users engage with the app. They also point out that it is important the results are visible for the users to feel accomplishment and mastery. The CLD also shows that incomplete tasks and too much extrinsic motivation can decrease the motivation and must be monitored. We can also see that too many rewards will have a negative effect on the intrinsic motivation (Kapp 2012).

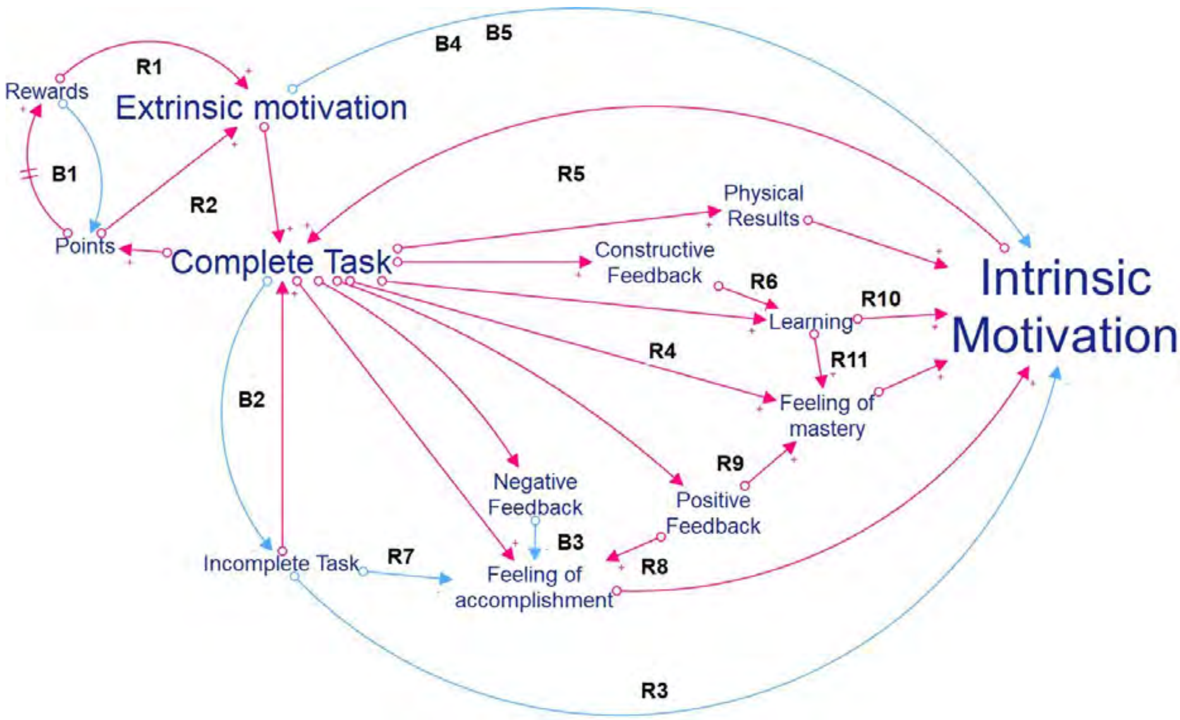


Figure 9. CLD for the psychological motivation model applied in the app.

Flow charts are important to understand how a gamification system works and are an integrated part of a System Analysis. Below are a few examples of the students flow charts.

The students first show a few examples on how Tasks flow through the system. We then look at the flow of motivation (Figures 10-12).

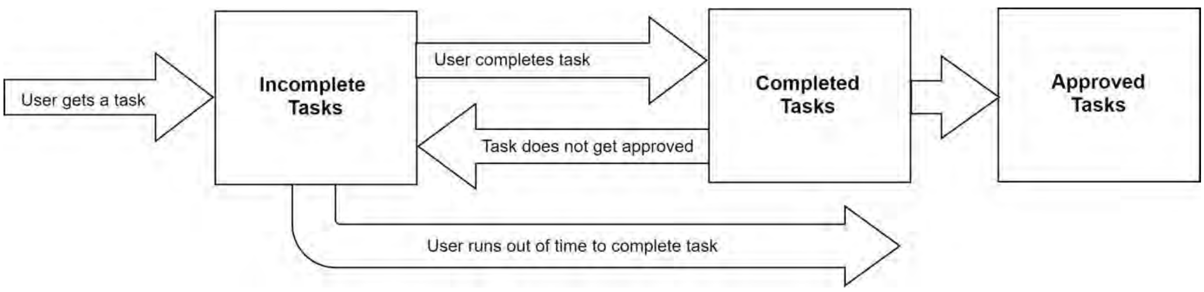


Fig. 10. Tasks. A task enters the system when it is assigned to a user. The user either completes it or runs out of time to do so. If time doesn't run out, the task is put in the queue to be approved.

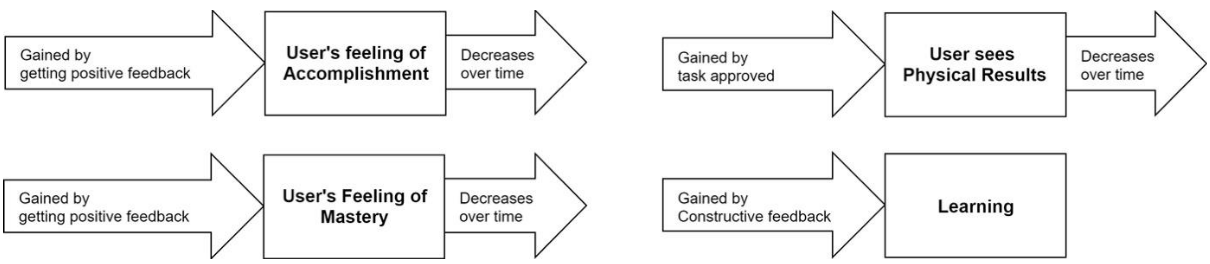


Fig. 11. Users are motivated by feeling accomplishment and mastery, learning and seeing results. Over time motivation decreases. The students assume that learning, unlike

motivation, can't decrease over time.

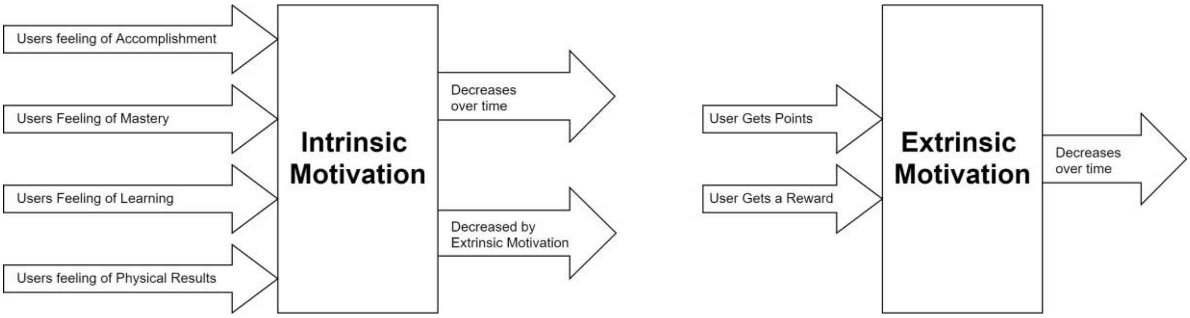


Fig 12. Motivation. This flow chart shows how mastery, accomplishment, learning and physical results will affect intrinsic motivation, while points and rewards influences extrinsic motivation.

The CLD and flow charts are the basis of the system dynamics simulations. The simulations were created in ISEE Systems Stella Architect and included several scenarios (Fig 13).

The simulation in fig 13 is best understood by looking at the CLD in fig 9 and the flow charts in fig 12. To the left is the task manager that handles tasks from when they are introduced until they are approved. Note that everything in the simulation is arrayed so that each family member has their own simulation. The psychological parameters related to intrinsic motivation are shown on the top right, while the same parameters linked to extrinsic motivation are shown on the lower right. The combined intrinsic and extrinsic motivation points for each family member are in the middle of the screen.

The initial values for the simulation are shown in Table 1. We see in this example that the parents have higher skills than the children, and that skill level increases with the age. The next line shows the initial values for intrinsic and extrinsic motivation. The Base Constructive Feedback Percentage indicates when to give constructive feedback. There is also a random chance of +-10% added to the initial value – for example, solving 60-80% of the tasks gives the kids constructive feedback.

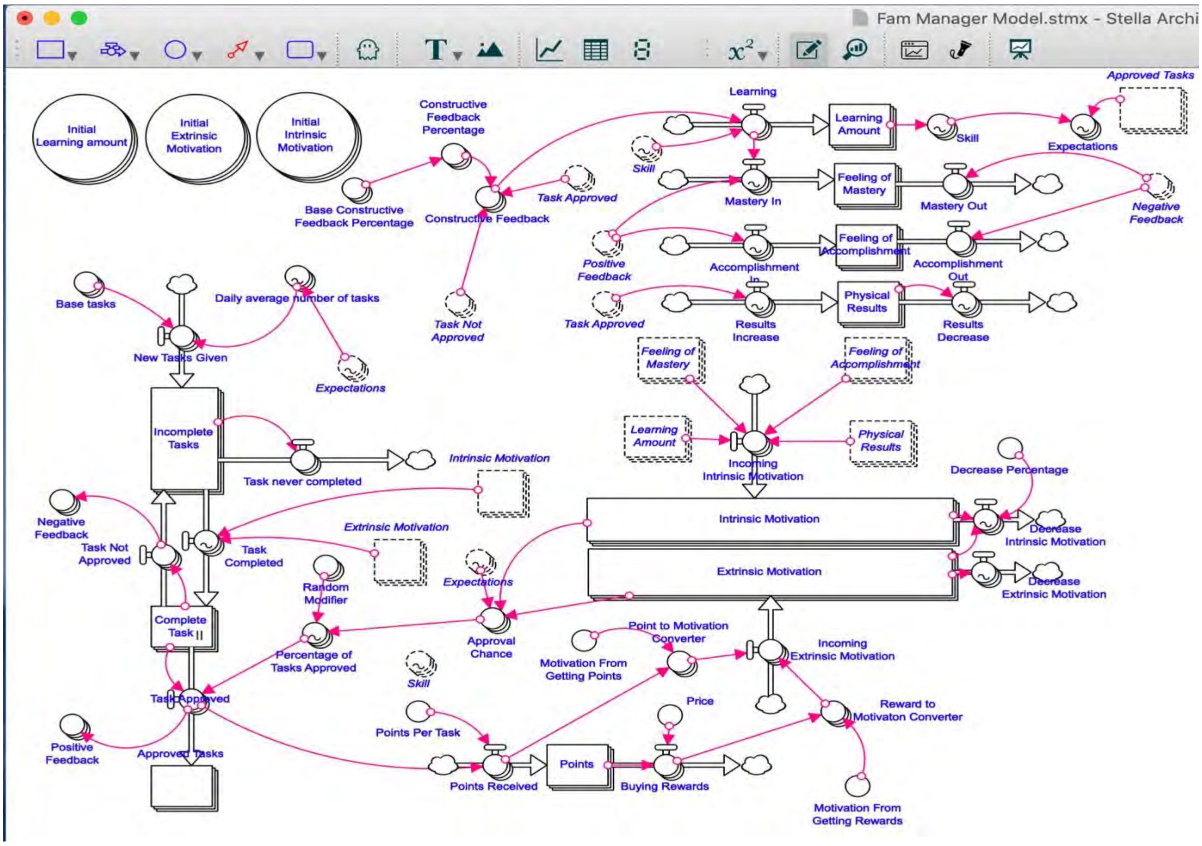


Fig 13. The System Dynamics STELLA Architect model interface.

Initial values	Female (8)	Male (10)	Female (12)	Mom	Dad
Skill	2.1	4.19	5.06	8.92	8.53
Intrinsic Motivation	10	20	2	50	50
Extrinsic Motivation	50	30	10	10	10
Base Amount of Tasks	2	2	3	5	5
Base Constructive Feedback Percentage	70%	70%	70%	0%	0%

Table 1. Overview of the parameter settings in the model.

The students simulated three scenarios. Each scenario runs for a whole year. The scenario in Fig 14 uses the initial values in Table 1. Here the intrinsic motivation increases steadily for all members of the family, which is exactly what the students hoped to achieve. We can see that the extrinsic motivation moves towards the same constant value for all family members. The students point out that as this doesn't increase indefinitely, at some point intrinsic motivation will take over for extrinsic motivation. They further assume that the app helps both kids and adults to "find a lasting motivation to keep a structured life."

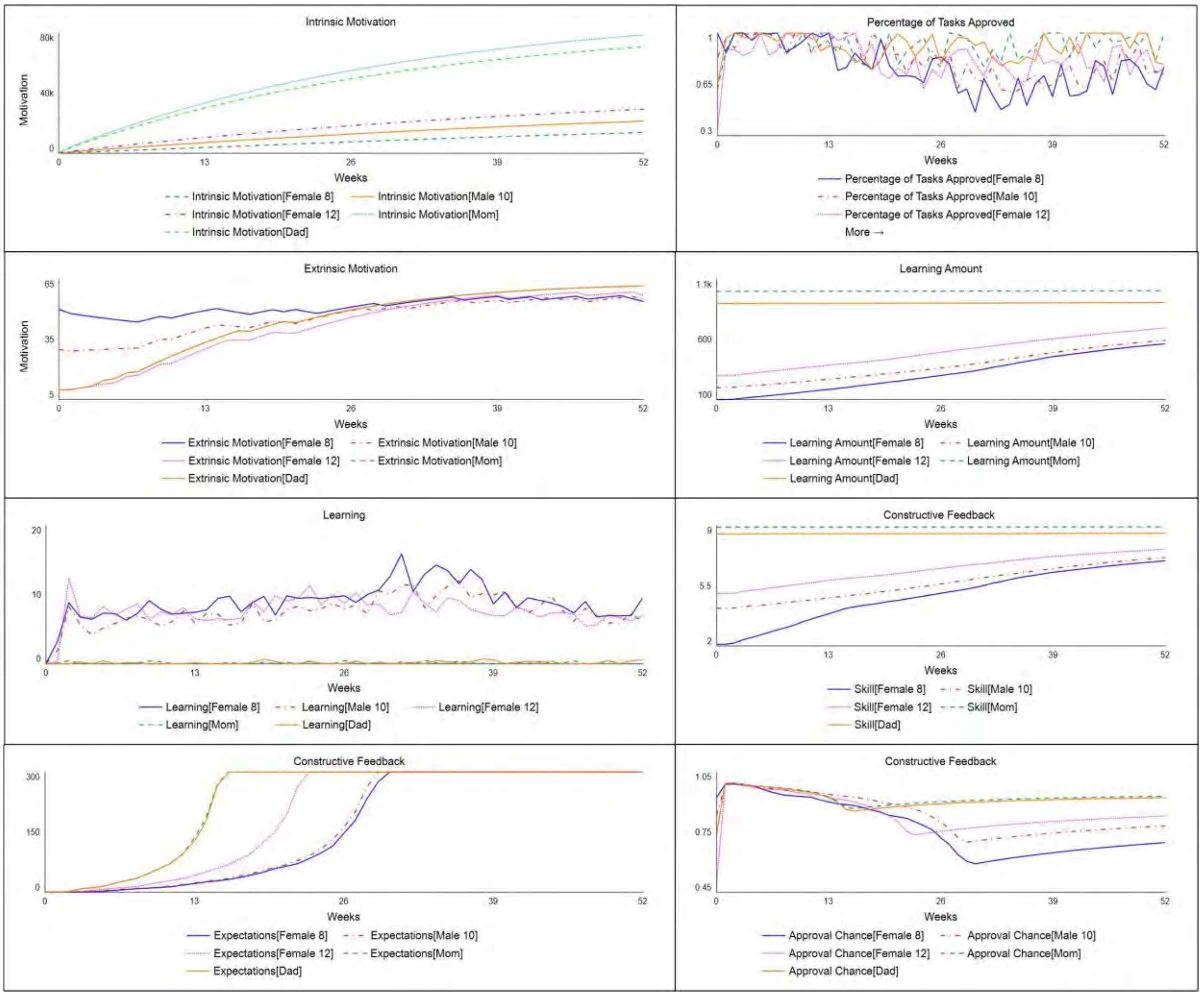


Fig. 14. *These System Dynamics Simulation diagrams shows how Motivation, Learning, Tasks Approved and Constructive Feedback develops for 5 family members.*

In the two other scenarios (not shown), the students repeated the simulation, but with the constructive feedback set to a range of 0 to 100%. The scenarios show that children's intrinsic motivation increases to a noticeably lesser degree when little constructive feedback is given. Just as importantly, children's skills barely increase when nobody tells them how they can improve. The students point out that the message system in the application is very important. It gives children power to choose and a feeling of mastery; without this feature, the system was much less effective, which is precisely what self-determination theory finds (Kapp 2012). The students articulated this observation in their report: "This proves that the usage of SDT in our app is the oil that keep the gears turning".

Discussion

Building a gamification can be an intangible and complex process. First, the real world system the gamification is built on must be understood thoroughly. Next, research must consider the gamification's users and how they will engage with it. Additionally, developers must ground their understanding of how the app motivates and teaches the users in a methodology such as psychology. A learning loop CLD must also be made so the learning process is clear, and goals must be set so user learning can be verified. The examples we discuss show that System Thinking gives students tools to acquire a more tangible understanding of design principles and practices in their gamification scenarios. With System Thinking, students also have the means to test their hypothesis in a more structured, manageable, and tangible way.

However, the student examples here are by no means complete or conclusive. There are many other ways System Thinking can be used in gamification. For example, before the gamification design process is even started, system analysis can provide a more thorough way to understand the real-world systems it builds on, or aspects of the real world that the gamification will affect. During the app design process, System Thinking can be enhance students' understanding of how players learn or reach their goals by making a learning loop for the design. Later in the process, System Thinking can be used to experiment with game balances, testing and debugging, and of course aspects of marketing, distribution, and server loads.

There is also constant development in where and how system thinking is used in gamification design. We believe this is because system thinking forces us to dig deeper into the gamification. Our students have often pointed out that merely engaging system thinking around their gamification makes them dig deeper into the details and create more detailed solutions. The system analyses provide them a deeper qualitative and logical understanding of how cause and effect function in gamification systems, and how the gamification app will perform and create user motivation. In this regard, system analysis makes it easier to see, for example, where to insert game elements that create intrinsic motivation. Additionally, this mode of thinking leads students to pay more attention to how the game mechanics work logically in gamifications and to discover if there are hidden feedback loops. All in all, a system analysis forces students to think more closely about the details of the design, the feedback loops, and how the various gamification systems work together to achieve their goals.

Gamifications will invariably include deploying psychology, pedagogics, and what the game designers consider common sense. The foremost issue here facing developers is grounding the gamification design, the motivation, and the in-game learning in psychological and pedagogical proven theories, rather than merely in so-called common sense. When students use psychological models like ARCS, Self Determination Theory, and similar psychological theories they have a more tangible base on which to ground their gamification design. Yet theories like these are often hard to understand and implement. Breaking them down into causes and effects in the CLDs that are an aspect of system analysis gives students a better understanding of the theories as well as providing solid and useful data and predictions about the design and behavior for the app's users. However, it is hard to find numbers or statistics on these theories that can be used in a system dynamics simulation of the gamification. These types of data would typically have to be found through user queries, interviews, and observations, as well as real-life user tests of the prototype gamification. Psychology and pedagogics are important not only in gamification but also in games; hopefully more research data and statistics in these areas will become in the next few years.

As the examples above show, it is necessary to supervise the students so they engage in scientifically based reasoning rather than wishful thinking. For example, some student diagrams shows that training and intrinsic motivation decrease over time. This is a certainly possible outcome, but the claim needs scientific foundation to be considered valid. In the same way, a statement that the need for extrinsic motivation will decrease because intrinsic motivation increases is based only on wishful thinking and would need to be backed up by scientific research in a real situation. However, students' simulation principles remain valid, and that is what counts in the classroom context. Since gamification is intangible and complex and involves a lot of lofty goals, it is easy for the systemic and straightforward nature of both system analysis and system dynamics to lead students to make claims that become self-fulfilling prophecies based on assumptions more than scientific facts and theories.

System dynamics simulation, the other component of system thinking, is quantitative by nature and offers a better understanding of the value ranges of the in-and-out data from the gamification. The system dynamic simulations in Stella are a valuable tool for students because they provide a real-time opportunity to numerically test different scenarios developed in system analysis. The quantitative output produced makes it easier for students to understand, theorize and predict how the messy real-life usage of a gamification can behave. Even the simplified version of simulation in Example 1, where the students lacked data, gives a better view of the dynamics in the situation. System dynamics simulations are, additionally, helpful tools for predicting the application's behavior.

If designers simulate the gamification fully and include all input and output parameters available, the dynamic simulation can be used to design the actual programming code for the gamification application. Code from Stella can be exported as pseudo code, which can then be translated into any programming language. This a very powerful feature – making a system dynamics simulation in Stella is easier than writing program code from scratch, and programmers can execute the simulation code directly in the game engine. A simulation in Stella can also save user testing time, and with repeated simulations students will better understand how the application will work with real users.

Developers can still write all simulation code from scratch based on the system analysis. This approach gives developers more flexibility and control but is also more difficult and requires more work, and in this scenario the system analysis remains very important as it serves as the complete logic design drawing for the programming design of the gamification.

Results and Conclusions

We believe the student examples discussed above answer our research question: How can System Thinking be used to design**,** develop, and document the process of **gamification**? The examples show how students undertake system analysis by making CLDs for gamification designs (Fig.5), and they illustrate the link between game elements and motivation (Fig 9). The examples also provide flow charts of everything that flows through the gamification (Fig. 4 and 10-12). Lastly, they simulate the system analysis quantitatively through the use of system dynamics (fig 6 and 13).

However, as discussed above, this assignment has also revealed new perspectives on the ways that system analysis and system dynamics can be applied in developing gamifications. Using system dynamics can create a better understanding of the real-world system developers will gamify, and building simplified models or "fish tanks" (Gee, 2007) of a case can improve understanding of how the system functions in general. Using system dynamics simulations to thoroughly test a gamification can reduce time and money spent on user testing. Undertaking different testing scenarios of the simulation will not only help iron out bugs and logical inconsistencies, but also aid in balancing game elements such as challenges, bots, or tools to match different user responses. Trying out different parameters to test balancing is an area where the results can produce large amounts of data to measure and accommodate different users and user skill levels. Lastly, System Thinking can offer meaningful insight into predicting shifts in markets, user numbers, server loads, update schedules, and other aspects of the game design production phase.

In sum, through our classroom instruction, not only have we have discovered several, beneficial ways to implement system thinking into gamification design, this process has a forced us to dig deeper into individual design projects in ways that yield new information about new areas of gamification research that can benefit from the application of system thinking.

References

Barrows, H.S., & Tamblyn, R. (1980). *Problem-based learning: An approach to medical education.* New York: Springer.

Blohm, I., & Leimeister, J. M. (2013). Gamification: Design of IT-Based Enhancing Services for Motivational Support and Behavioral Change. Business & information systems engineering, 5(4), 275-278. https://doi.org/10.1007/s12599-013-0273-5

Burke, B. (2014). *Gamify: How Gamification Motivates People to Do Extraordinary Things*. Brookline: Brookline, US: Taylor and Francis.

Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

Deterding, S., Sicart, M., Nacke, L., O'Hara, K. and Dixon, D. (2011). Gamification: Using Game Design Elements in Non-Gaming Contexts, In: *The Gamification Research Network. CHI 2011 Workshop Gamification: Using Game Design Elements in Non-Game Contexts*. http://gamification-research.org/wp-content/uploads/2011/04/CHI_2011_Gamification_Workshop.pdf

Dohn, N.B. (2011). Situational Interest of High School Students Who Visit an Aquarium. **Science Education**, *95*(2), 337-357. doi:10.1002/sce.20425

Efron, S. &, Ravid, R. Action Research in Education, a practical guide. EPUB Edition ISBN: 9781462541669; Kindle Edition ISBN: 9781462541645

Gee, J.P. (2007). *Good video games and good learning: Collected essays on video games, learning and literacy.* New York: Peter Lang International Academic Publishers.

Haraldsson, H.V., Sverdrup, H.U. 2004 Finding Simplicity in complexity in biogeochemical modelling. In: J. Wainwright and M. Mulligan (Eds.), **Environmental Modelling**. Wiley, New York, 211-223.

Kapp, K.M. (2012). *The gamification of learning and instruction: game-based methods and strategies for training and education*. San Francisco, California: Pfeiffer an imprint of Wiley.

Keller, J. (1987). Development and use of the ARCS model of instructional design. *Journal of Instructional Development*. *10*(3). 2-10

Kolb, D.A. (1983). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice Hall

Krapp, A. (2002). Structural and dynamic aspects of interest development: theoretical considerations from an ontogenetic perspective. **Learning and Instruction**, *12*(4), 383-409. doi:10.1016/S0959-4752(01)00011-1

Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge: University Press.

McGonigal, J. (2011). *Reality is broken: why games make us better and how they can change the world*. London: Jonathan Cape.

Mäyrä, F. (2008). *An introduction to game studies: Games in culture*. Los Angeles, Calif.: Sage.

Nordby, A., Øygardslia, K., Sverdrup, U., Sverdrup, H., 2015f. "The best schoolday ever" – Teaching sustainability and systems thinking by pervasive game playing. In: **EDULearn** 15:689-698. ISBN 978-84-606-8243-1, ISSN 2340-1117. 7th Conference on education and new learning technology, 6-8th July 2015

Nordby, A., Øygardslia, K., Sverdrup, U., Sverdrup, H., 2016. The art of gamification: Teaching sustainability and systems thinking by pervasive game development. **Electronic Journal of ELearning (EJEL)**. 15:689-698. ISBN 978-84-606-8243-1

Rheinberg, F. (2008). Intrinsic motivation and flow. In J. Heckhausen & H. Heckhausen (Eds.), Motivation as action (pp. 323 – 348). New York: Cambridge University Press.

Schiefele, U. (1991). Interest, learning and motivation. **Educational Psychologist**, 26(3 – 4), 299 – 323.

Schlyter, P., Stjernquist, I., Sverdrup, H. 2013 Handling Complex Environmental Issues – Formal Group Modelling as a Deliberative Platform at the Science-Policy-Democracy Interface. In; Schwanninger, M., Husemann, E., and Lane, D., Proceedings of the 30th International Conference of the System Dynamics Society, St. Gallen, Switzerland, July 22–26, 2012 Model-based Management. University of St. Gallen, Switzerland; Systems Dynamics Society. 4:3525-3538. ISBN: 9781622764143 Curran Associates, Inc.

Senge, P., (1991). The fifth discipline, the art and practice of the learning organisation, Century Business, New York.

Sverdrup, H., Svensson, M. 2004. Defining the concept of sustainability, a matter of systems analysis. In: M. Olsson; G. Sjöstedt (Eds.); *Revealing Complex Structures -- Challenges for Swedish Systems Analysis*, 122-142. Kluwer Academic Publishers.

Sverdrup H. (Ed.), Haraldsson, H., Olafsdottir, A.H., Belyazid, S., Svensson, M., 2019 System Thinking, System Analysis and System Dynamics: Find out how the world works and then simulate what would happen. 3rd revised edition. Háskolaprent Reykjavik. 310 pages. ISBN 978-9935-24-425-3.

Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge; University Press.

Appendix

1

TEACHING 'HARDCORE SCIENCE' TO ARTS AND DESIGN STUDENTS OF THE GAMING GENERATION: REFLECTION ON HOW TO USE GAME DEVELOPMENT TO TEACH PROGRAMMING, SYSTEM THINKING, MATHEMATICS AND PHYSICS

Anders Nordby[1]

Hedmark University College (NORWAY)

Abstract

This article reports on the development of two courses designed for students in higher education game development programs during the period of 2006 to 2015. The students in the course had in common that very few had taken advanced science classes as part of their upper-secondary education, which meant they were poorly equipped to learn the programming, mathematics and physics they needed to master game development. Consequently, both courses were designed in a way so that the students could practically engage in creating a computer game and being taught the science they needed when they needed it to solve a problem in the game. A working hypothesis for the project was that if we were able to run the course in a way that cohered with the principles of problem-based learning, this would create an environment which would enhance the students' motivation to learn basic science. In addition, ideas developed within the field of situated and experiential learning constituted theoretical points of departure for developing the course.

In the second example we also taught mathematics and physics by first developing the system thinking and the system dynamics, then program the output from the dynamic simulation, and finally dive into mathematics syntax to explain what they just have done. This is learning by doing too – they address the theory when they need it in the game.

Overall, the programming course design functioned successfully in merging the practical task of game development with increasing the students' mathematics and physics knowledge as well as their programming skills. In addition, the students had access to collaborative social environments both in class and online, which worked as creativity-enhancing communities of practice.

Keywords: arts and design students; computer programming; mathematics; physics; science education; gamification; problem-based learning; situated learning

1 INTRODUCTION

The fast technology development in the recent years have made it possible to create very real looking games as well as brought back the focus of the 90ties virtual reality boom. This draws more and more people to virtual worlds and online games due to the increased realism and immersion. Statistics shows that the worldwide video game industry grew by 9% in 2013 and will reach $86 billion by 2016 [1]. Out of this, Europe constitutes almost $20 billion. Much of this gaming is done on mobile devices; 91% of all people on earth have a mobile phone and 80% of time of the time on these is spent inside apps or games [1]

This technology development have also made it possible for 'ordinary' people without special skills in art or programming to make their own games, and distribute them to the whole world on the fast growing pad and phone networks (iOS and Android). Games (apps) are cheap on these devices, and more and more developers make their games modifiable so the players can create their own versions of the games, or even make brand new ones. Big companies like Microsoft and Electronic Arts have had apps for game development in their portfolio for years, and more and more game developers and networks open up their game engines for 3rd party game development and modding.

These new game developers, often called the ‘gaming generation’, knows a lot about game development, virtual arts, animation and programing. They start early in life [2] so that much of this knowledge is there already in elementary school. They are creative and they are used to collaborate, practice and modify to learn new things. When they need to learn something new to solve a task, they find the necessary theory online and collaborate with other children from all over the world in online communities.

Some teachers and researchers starts to see how much knowledge these kids possess in this area, and the potential this knowledge have when it is used as a vehicles to teach traditional school topics.

This article focus and gives examples on how game development in particular can be used to teach traditional school topics, and also how it can help to bring informal online forums and peers into the classroom.

2 GAME DEVELOPMENT AND THE ASPECT OF LEARNING

Marc Prensky [3] argues that the “net generation” or “game generation ” is different from other generations. Due to their heavy use of Internet and digital media not only their way of thinking, but also their brain have changed. Table 1 below gives an overview of what cognitive styles have changed and how. These changes have far reaching consequences for how they learn, and the education system they are learning in.

	Cognitive style changes	Outcomes
1.	Twitch speed vs. conventional speed	Quicker processing of information
2.	Parallel processing vs. linear processing	Ability to concentrate on many tracks at once
3.	Graphics first vs. text first	Sharper visual sensitivity, increase in iconic understanding
4.	Random access vs. step-by-step	Awareness and ability to make connections between multiple sources of information (“hypertext minds”)
5.	Connected vs. standalone	Ability to work in virtual teams. Being connected is a necessity.
6.	Active vs. passive	Figuring things out by trial and error. Less tolerance for passive situations such as lectures and meetings. “Just do it” approach.
7.	Play vs. work	Play attitude enters “real” world
8.	Payoff vs. patience	Intolerance for things that do not pay off.
9.	Fantasy vs. reality	Tendency to combine fantasy and reality in the design of work places.
10.	Technology-as-friend vs. technology-as-foe	Perceiving access to new technologies as a natural element of environment (‘birthright’).

Table1. from M. Prensky, 2007, pp.51-65, Cognitive style changes attributed to Games Generation[28]

Prensky point out that these kids “requires multiple streams of information, prefer inductive reasoning, want frequent and quick interactions with content and have exceptional visual literacy skills – characteristic’s that are all matched in digital game based learning” [4]. They tend to make meaning in an active way and play roles and participate, and they want to have fun while doing it. In interviews about his book “The Net Generation”, Tapscott [5] elaborates on what these young kids demand from life, education and future employment; “They want to have fun. In fact, 58% of them say that having fun with a product or service is just as important as what that thing actually does. If you employ any of these people, realize that they also want to have fun at work. They want to collaborate and have relationships. They want innovation and creativity. They want speed. They want to customize everything … this group wants to do things their own way” [6].

Most children consider game development and game playing fun, especially if used in “boring” school environments. When learning environments are gamified in a good way, they can make learning fun and turn the classroom into a ‘playroom’ where learning events appears naturally. This is an environment where the children are “learning by doing” [7], and the need to study traditional topics pops up “just in time” [8]. These environments can also naturally incorporate informal learning environments like internet forums and discussion groups that children are confident with from their informal learning, and allow for more active and creative teaching.

This is Gamification of the learning environment. The Learning Circuits Blog defines Gamification as “using game-based mechanics, aesthetics and game thinking to engage people, motivate action, promote learning and solve problems” [9]. In the table below Gee points out what these environments are particularly good at:

Principles of learning	What good games achieve
Co-design: learners feel like active agents not passive recipients	Games are interactive. Players’ experience is created by their actions and decisions made in the course of playing
Customize: different learners have different learning styles	Games are designed to allow different styles of playing and different ways of handling challenges
Identity: taking on a new and valued identity intensifies commitment	Games offer players identity that *triggers a deep investment on the part of player*
Manipulation and distributed knowledge: perception and action are inter-connected. Humans feel empowered when they can manipulate tools at a distance and extend their area of effectiveness.	Games involve action at a distance and a possibility of manipulating the world’s objects and use them as tools for achieving objectives
Well-Ordered Problems: problem spaces should be designed in the way that leads to hypotheses that work well	Earlier parts of good games prepare players for progressing throughout later parts of the games
Pleasantly frustrating: learning works best when the level of challenge is accurate. It cannot be neither too high nor too low.	Good games adjust challenge in a rewarding way
Cycle of expertise: repeating the cycles of practicing skills leads to expertise	Good games support the cycle of expertise in the way that challenges constructively routinized skills
Information “On Demand” and “Just in Time”: verbal information is used best when it is given when it can be put to use and when it is needed	Players do not need to read the manual to start playing but can use it as a reference at any moment

Fish tanks: it easier to approach complex world through simplified models, so called fish tanks, where elements and relations between them are more transparent	Good games offer player simplified models (fish tanks) in forms of tutorials or initial levels
Sandboxes: people learn better in situations that resemble real world but are less risky (like sandboxes)	In good games the cost of taking risk, trying new hypotheses is never too high contrary to schools which may be too punishing
Skills as strategies: people develop skills best when they perceive their sets as strategies leading to accomplishing goals	In a good game players use skill packages to achieve their goals. This helps them see skills not as separate entities but as strategies
System thinking: people learn best when they understand how things fit into a larger meaningful whole.	Games allow players to develop good intuitions about the rules, effective actions and the way the game environment functions as a whole.
Meaning as action image: people tend to think through experience rather than general definitions and logical principles. Meanings are tied to perception and action in the world	Games provide perceptual simulations that make ground for situated learning.

Table 2. J.P. Gee, 2007, pp. 28-43. Principles of learning built into good games[28]

Gee points out that system thinking have a natural place in these game environments because games are systems, and because people learn best when they understand how things fit into a larger meaningful whole. System thinking is about understanding the overall system of a problem and the realization that the resolution of the problem involves to grasp and involve the whole. To find out what the whole is System thinking uses Causal Loop Diagrams (CLD) and flow charts that shows the flow, the causal connection and the feedback loops in the system (see below for further definition of system thinking). Fig 3 shows a simple example of a CLD/Flow chart that illustrate how the learning in a gamified environment is connected:

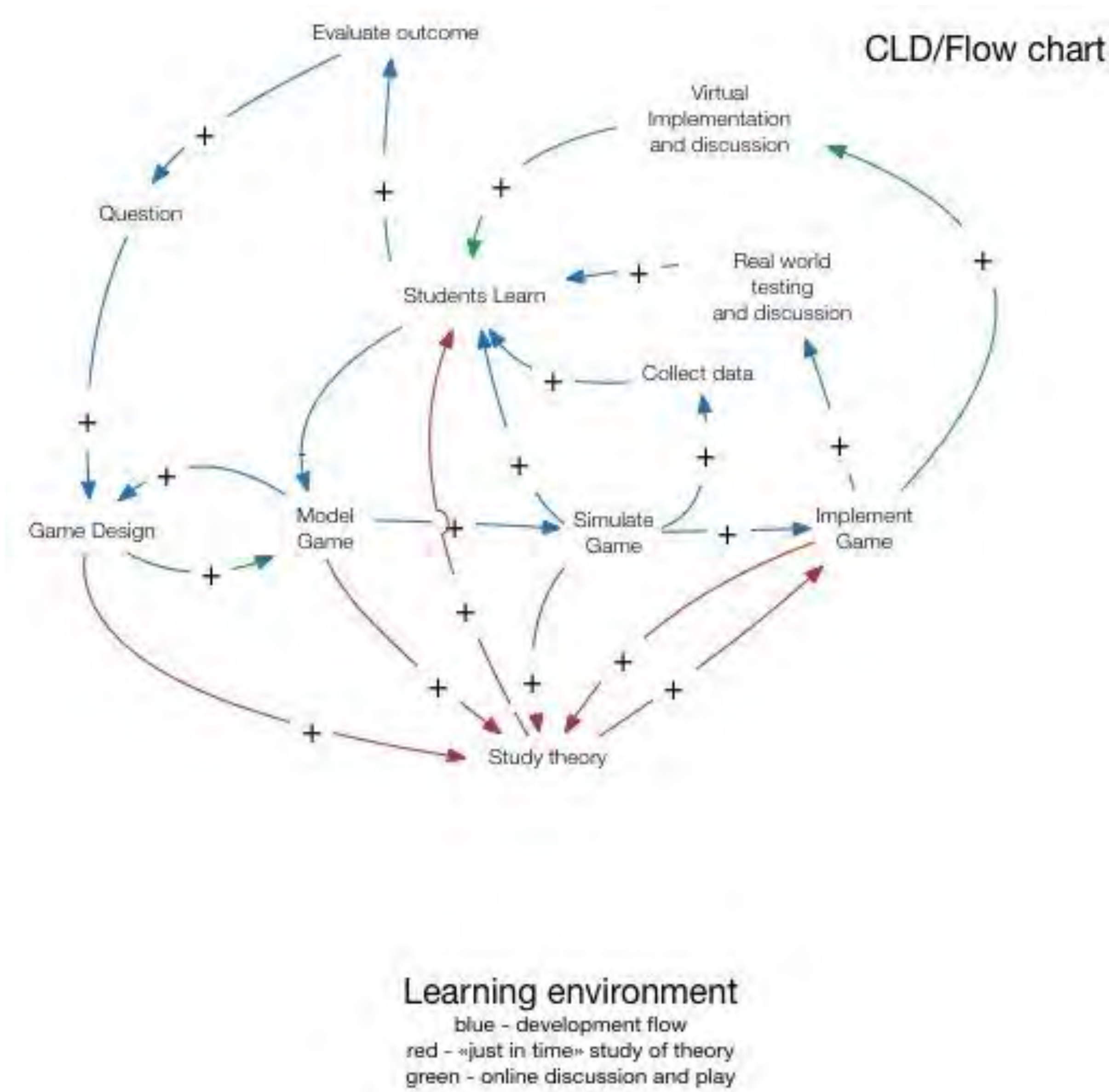


Fig 3. CLD and Flow chart of the game development flow. The CLD shows how informal learning environments are incorporated into the classroom and how study of theory is braided into the game development process.

3 THEORETICAL POINTS OF DEPARTURE

Gamification draws upon basic psychology; Kewin Werback [10] points out that gamification draws upon cognitivism to create extrinsic motivation by using for example points, badges and leaderboards as rewards, but also that gamification create environments on its own that can trigger intrinsic motivation. In these environments the player can experience the whole spectrum of an action, from making a decision to facing the result of the action, and the player internalize knowledge by reframing their mental representation with what they experience in the game. This frames the gamification activity within the theory of Constructivism.

When doing game development, which is gamification, the students learn by making a game instead of learning how to make a game. This is experiential learning [12]. The environment the students work in (the groups, the class or the online communities) works as a community of practices, where the common goal is to make the game and not just learns school topics. This frames the game development within the theory of situated learning [13].

Game development can be problem based on several levels; first the development cycle is made problem based in itself, and secondly while developing the game problems and challenges pop up naturally which has to be solved to make the game work. Originally Problem-based learning (PBL) was used in medical school in the late 1960s by Howard Barrows and his colleagues [14], where It was used as a model for facilitating student-centered learning groups, led by a tutor whose task it was to guide the students rather than teach them, and the focus of the group was to solve problems related to the professional world which the students aimed to enter. Gaming students doing game development is a similar situation, but instead of solving real world problems they create a game. The solution of the problems is also the same; they will have to study theory to solve the problem.

At last, game development is also about social collaboration and interaction both in student groups and class and in online forums and games. This calls upon Vygotsky's Cultural-historical psychological theories [11] and the Zone of Proximity. New learning is at it best when it builds on previous experience and the previous experience is not outside the Zone of Proximity. J.P. Gee take his thoughts further in his Situated Cognition Studies where he compare the bran with computers and computer networks that store whole experiences and where new experiences refines the old ones a bit. This is what we do when using game development as a vehicle for learning; we build on the knowledge the kids have from informal learning and build on and refine that knowledge with theoretical school knowledge.

4 DESIGN

Below are two examples where game development is used to teach traditional science topics such as programming, system thinking, mathematics and physics. The first example is a programming class for 1st year art and design students in upper secondary school, and the second are a course teaching system thinking to 2nd year game design students, also in upper secondary school. Although they teach different topics and the students are on different level, they both use game development as motivator, and share many of the same design properties.

4.1 The Programming courses (2006-2010)

The students attending this course had just started out their education and came from 3 different bachelors in animation, arts and game design. Typically, around 70 students attended the course. Most students were between 18 to 25 years old, and were typical examples of the game generation described in the beginning of this article.

The main goal of the course was to tech the students' computer programming, but we also wanted them to experience why basic understanding of 'hardcore science' topics was needed in digital arts education, for them to be able to create the advanced computer-based artworks that they aimed for.

During a trail year (2006) we experimented with different kind of games, and found out that making a basic platform game would give the students the right balance of challenges. This type of game would also be possible to implement in a freshmen course, and would require a moderate level of programming, mathematic and physic skills.

We wanted to use a programming environment that gave visual feedback to the students right away - an environment where the programming, mathematics and physics would show up visually as behavior or environment changes in the game graphics. At the time Adobe Flash was a highly popular tool to make games and animation movies, and we found that this environment also satisfied our needs for visual feedback as well as giving us a good object oriented programming environment.

We divided the game into 6-10 smaller problem based tasks or cases that was given to the students as assignments over the semester. Each case were1-2 week long, dependent on how difficult the case was.

We wanted all cases to be open-ended so that diligent and clever students could develop them beyond the basic requirements without proceeding to the next case.

A typical order of the cases would be:

1. Design the game and make a storyboard
2. Design the characteristics of the main character of the game, and make drawings of this character in Flash
3. Divide the character into layers, animate its walking cycles and jumping abilities
4. Connect the triggering of the walking cycles to the keyboard arrow keys (here the programming starts).
5. Draw the ground of the game and make the main character follow the ground
6. Make a second character (an enemy) and let it follow the main character
7. Implement jump for the main character and design platforms to jump on
8. Make a start screen with the necessary menus, and screens to be shown when the player win or lose the game

The students were divided into groups, with 6-10 students in each group. Each groups was given a tutor from a higher-level class. The groups met twice in the case period, each time about 3 hours. First time was in the start of case period, when they needed to define the learning goals and the theory they needed to study to solve the case. After this meeting they separated, studied the theory individually, and then completed the case in the best way they could. After that they met again and discussed the challenges the case had given them, and how they had solved them.

We choose the Maastericht model with 7 steps [15] as the teaching model. The first 5 steps in this model; clarify difficult terms, define the problem, brainstorm, analyse and formulate the learning objectives was done in the first meeting. Step 6 was self-study period, and step 7 was the post-discussion and sum-up meeting in the group.

Many opportunities to teach mathematics and physics popped up naturally during the cases. For example, when the students had finished their game characters and animated them, they needed to make them jump. They usually did this by applying a force upwards, which made the characters disappear out of top of the screen. To “fix” this they had to study theory about how forces worked in nature, and especially how the gravity force drag everything back down to “earth”. They had to dig deep into this theory; natural looking character jumps and walks required a good understanding of different forces, gravity and friction. Very few of the students found it odd to study forces and gravity in this situation.

Another example was when the students needed to implement their characters' walking cycles and moves around the screen. Then they needed understand about functions, x- and y-axes, coordinates, angles, and conversion between degrees and radians. Eventually, they also had to learn trigonometry to be able to divide the movements into horizontal and vertical components. Students often struggle to understand this kind of theory if it is taught on an abstract level. Here, however, the physics and mathematics became comprehensible because they needed the knowledge to make their game work. In other words, the 'fun' and 'motivation' aspects of the game development made the students see the 'hardcore science' as elements in the overall game development process, and because of that they were willing to put much effort into understanding it.

Throughout the course the students had to present their games online in wiki pages and blogs. This enabled them to present and discuss the project in online forums and discuss their problems and solutions with online peers. Most wiki pages and blogs were written in English. This was not a requirement, but for most students it was a natural thing and something they had done for years.

4.2 The System thinking courses (2012-2015)

This example is from two more experimental courses that not yet has found its final form, but it is presented here because we aim at the same as in the programming course; to teach 'hardcore science to arts and design students. In the first course the students learn system thinking in a more traditional way by studying real world sustainability problems. The second course builds on the first, and in this course the students actually develop games using system thinking and simulation as a base for the game design. The games are then simulations of real world problems, but presented in a graphically interesting and simplified way, so the players easily gasp the real world problem or challenge.

Before we start describing the course, lets define System thinking a bit more thorough: System thinking is about understanding the overall system of a problem and the realization that the resolution of the problem involves to grasp and involve the whole. In order to do this systems analysis and systems dynamics simulations is used [16], [17] ,[18]. Systems analysis is the process finding the causal links in the system and finding when the linkages constitute the feedback loops. This is often facilitated by using what is called the Group Modeling Process, where the stakeholders in the problem works with the systems analyst to map the system in an iterative adaptive learning mode [19], [20]. Systems Dynamics and Simulation is to use the system maps, such as mind maps, variables classifications, causal loop diagrams and flowcharts to create numerical simulation models. These models are used to quantitatively explore system behavior and sensitivity to change, and also used in iterative mode for future predictions and development of policies for future actions. Systems dynamics is often used in engineering, economics, research and management [21], [17], [22].

We choose to give the students sustainability related tasks mainly because system thinking is essential in understanding this kind of problems, and because we wanted the students to learn how they as game developers can contribute to the understanding of the sustainability challenges in the world by conveying them in an interesting and fun graphical way. We gave the students many different tasks to solve - two years ago we let them develop and test a pervasive game about sustainability in elementary school. The pervasive game in itself was a treasure hunt in the school area. During the hunt they had to solve puzzles and meet actors that played gardener, traders, refugees and even earthworms. The hunt also involved solving three digital games; one about human consumption, pollution and the ecosystem, one about the earthworm and soil ecology and the last about the refugee situation in the world. This project is more detailed explained in article "The best school day ever" [23]. Another example is this year's course where the students made digital games only - the first about the Galapagos islands and the balance between truism and wildlife diversity, the second about the Jordan River water conflict, and the third about the blessing and pain of oil.

The design and framing of the course is in many ways similar to the programming course described above. The goal is to make a game, the course is problem based and the students work in groups and have higher year students as tutors. The class and the student groups are smaller; the class is typically between 12 and 20 students and groups size typically 3-5 students. We use problem based learning here too, but have less focus on the Maastericht model.

The complete game is divided into 3 overall cases:

1. Make a the system thinking and simulation part of the problem
2. Design the game
3. Implement the game. This part is huge and are divided into smaller cases.

The implementation is by far the longest in time and we usually divide the work they do in this period into sub tasks similar to the ones described in the programming class, although on a higher technical level. These tasks are often different in each group and dependent on what task they solve. The students regularly present their partial developments for the whole class and the teachers, and get feedback and suggestions on how to improve their work. I this project the whole class also work as a group (in addition to the smaller development groups).

There are many opportunities in these classes to present theory. In the first class where we teach system thinking, all kinds of physics and mathematics pop up, dependent on what problems they simulate. Details on this will be presented in a later article.

In the second class the students first make a CLD and flow charts and then make a numerical simulation of the problem in a modeling and simulation software tool. We use STELLA for that which is a well-proven software simulation environment made by IBM. During the development of the simulation they have to find numerical values for all the parts, and there are always opportunities in this phase to teach them the underpinning mathematics and physics. The simulation gives output in form of diagrams, curves and a numeric listing of variables, tests and loops in the simulation. At this point, they need to learn how to program this into their game. While this can be done without understanding the underlying mathematics and physics, they are usually very motivated at this point to learn the underlying theory. For example, the Stella listing contains the differential equations used by the system simulation, so it's a good time to relate this to mathematical theory on derivation and show them how the programming code for the differential equations is linked to mathematical language, both in continuous and discrete domains. At this time they know how the system and the simulation work; they have built it themselves, and they really want to understand how this makes the game work. Often they have also translated the Stella listing to programming code at this time, and they want to change values directly in the game to tune the game performance. This is much faster to do when they understand the underlying mathematics; they don't have to go back and simulate it all again. Again, they are motivated to learn the science, and put much effort into studying it. We also see that it is easier to teach them the theoretical mathematics after they have programmed it and seen how it works. Its “learning just in time”; they need the understanding to implement their game, and are highly motivated because they find making the game fun.

Up to 2015 some mathematics have been taught to the students in traditional mathematics course during in first year of they studies, but these courses have the same problems we see in traditional math classes these days; many students hate them, they don't understand why they have to learn mathematics, and a high percentage, often more that 50%, fails. The idea is that we next year teach the underlying theory in this course instead of in the math courses.

In this projects the students also write weekly blogs and make web sites that at all times reflect their development. They also use online forums in their work, and it gives opportunities to bring the informal learning and online peers into the classroom in the same way as in the programming course.

5. EXPERIENCES FROM RUNNING THE COURSES

As mentioned above the students in these classes documented their learning process in wiki pages and blogs. They also write reports at the end of the projects. The tutors also document their work and write reports for each case and after the class are completed. In the case of the pervasive game that was played in 5th and 7th grade classes, the children submitted diaries that later was used by the researchers to study the learning and motivation. This documentation together with observations made by the researchers has been the basis for the comments given below.

Very early on we saw that both the whole class and the student groups worked as communities of practice, where the common goal was to create a game [13], [25]. The game development approach was successful in creating a collaborative social community where the students could contribute, and the newcomers could learn from the more experienced. Comments made by the students and tutors in the documentation indicate that the common goal and the belonging in the community greatly enhanced the motivation of the students.

This community of practice also effectively worked as an environment for “learning by doing” and “learning just in time”. The students were motivated to study the science needed to solve the tasks that popped up naturally. According to their comments, many even found it fun to learn science this way and understood why they had to learn it.

The students also often took on an identity as game developers. This gave them credibility in online forums, and motivated them to use their experience from online learning also in the classroom. According to comments in the student blogs and final reports from the tutors, Internet forum peers were used heavily in many situations.

Another and more unexpected finding was how much the higher-grade students used as tutors liked to be tutors. Very often we didn’t have to recruit them, they came to us and wanted to be the next year tutors, and all of them put much effort and energy into their work. It is also interesting to note that we never paid the tutors for their work; we gave ETCS points and certificates on participation in the research project.

The groups where put together as heterogeneously as possible to make them as creative as possible [26]. The group sizes varied, but stabilized after a while on a size of 8-10 students in each group. This group size correlates with what the theory suggested both when it comes to maximising of creativity [26] and PBL learning [27].

It is also interesting to note that while the actual main goal of the projects was to teach programming and system thinking along with necessary mathematics and physics, the students looked at the course as game development and upon themselves as game developers.

From the student comments and documentation it was clear that most students had fun doing these projects, and they also put a lot of effort into them. It is also clear that they learned both the wanted school topics as will as other things such as collaboration, project management, language etc. during the project.

What we found though, is that the learning in these projects are a bit ad hoc, and that they needed help to structure and categorize the theory after the course was finished. We did this this in the next course, but it could be a good thing to make this part of the course. However, it should be after the game is done and not during the course, or it would slow down the creative flow and immersive community of practice.

We also found here what other critics have pointed to about problem based learning and self-studies, that it doesn't fit all kinds of students. We remedied this by providing much help (we were many teachers and tutors) and even traditional lectures and labs were done for selected groups when needed.

6. Conclusion

Overall we found that game development together with problem based learning methods worked as expected and gave good results. Very few students failed this class, and most students got an understanding of why we wanted to teach them science in an art class. However, the classes were highly cross-disciplinary and often overlapped with what they learned in other classes, which requires that teachers of different classes and disciplines cooperate. The method is also a bit chaotic, and the sequence of the topics taught might differ from classic teaching.

With these challenges in in mind we can highly recommend this way of teaching students, and we see few problems in using the same approach in lower lever classes.

References

[1] 2014 Global Gaming Stats: Who's Playing What, and Why? Posted by Lisa Galarneau on January 16, 2014 in Trends and Statistics - http://www.bigfishgames.com/blog/category/trendsand-statstics/

[2] KZero. (2011). *Universe chart Q4 2011: Avg User Age 10 to 15.* Retrieved from http://www.kzero.co.uk/blog/universe-chart-q4-2011-avg-user-age-10-15/

[3] Prensky M. (2007), *Digital Game-Based Learning*, Paragon House, the United States

[4] Van Eck R. (2006), Digital Game-Based Learning: It's Not Just Digital Natives Who Are Restless…*EDUCASE Review, vol.41, no*

[5] Tapscott, D. (2008). *Grown up digital: How the net generation is changing your world.* New York: McGraw-Hill.

[6] Davis, E. (2008). *Don Tapscott discusses the "net" generation.* Retrieved from http://blog.shop.org/2008/09/16/don-tapscott-discusses-the-net-generation/

[7] Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

[8] Gee, J. P. (2007). *Good video games and good learning: Collected essays on video games, learning and literacy.* New York: Peter Lang International Academic Publishers.

[9] The Learning Circuits Blog. (2012). *What is gamification? And why it matters to L&D professionals.* Retrieved from http://learningcircuits.blogspot.no/2012/01/what-is-gamification-and-why-it-matters.html

[10] Gamification (2015) Cousera Course, Kewin Werback, 2015

[11] Morgan, A. (2009, July 28). What is "Scaffolding" and the "ZPD"? Retrieved October 13, 2014.

[12] Kolb, D. A. (1983). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice Hall

[13] Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

[14] Barrows, H. S., & Tamblyn, R. (1980). *Problem-based learning: An approach to medical education.* New York: Springer.

[15] Maastricht University. (2013). *Problem based learning preparatory website.* Retrieved from http://www.umpblprep.nl/

[16] Sverdrup, H., Svensson, M. 2002 Defining sustainability. In: Developing principles for sustainable forestry, Results from a research program in southern Sweden. Sverdrup, H. and Stjernquist, I. (Eds.) *Managing Forest Ecosystems* 5:21-32 Kluwer Academic Publishers, Amsterdam.

[17] Senge, P., (1990). The fifth discipline, the art and practice of the learning organisation, Century Business, New York.

[18] Sterman, J.D., (2000). Business Dynamics, System Thinking and Modeling for a Complex World, Irwin McGraw-Hill, New York.

[19] Haraldsson, H. V. Introduction to systems thinking and causal loops diagrams. Reports in ecology and environmental engineering. Report 1:2004. 49pp

[20] Haraldsson, H.V., Sverdrup, H. (2004). Finding Simplicity in Complexity in Biogeochemical Modelling. J. Wiley and Sons Ltd., Chichester, pp. 211-223.

[21] Haraldsson, H.,, Sverdrup, H., Belyazid, Holmqvist J., Gramstad, R., The tyranny of small steps: A reoccurring behaviour in management, Journal of Systems Research and Behavioural Science 24, 1-19.

[22] Meadows, D.H., Meadows, D.L., Randers, J. , Behrens W. (1972). Limits to Growth, Universe Books, New York.

[23] "The best school day ever" (Nordby et al, 2015). EDULEARN15, Barcelona 2015 Procedings.

[24] Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

[25] Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

[26] Darsø, L. (2001). *Innovation in the making.* Frederiksberg: Samfundslitteratur.

[27] Pettersen, R. C. (2005). *Kvalitetslæring i høgere utdanning: Innføring i problem- og praksisbasert didaktikk.* Oslo: Universitetsforlaget.

[28] Godejord. B (2014) Digital Game-Based Learning: Synergy between Pedagogy and Engagement. Nord University.

2

THE POWER OF GAME DEVELOPMENT IN LEARNING: WHY IS GAME DEVELOPMENT GOOD LEARNING MACHINES?

Anders Nordby[1]

Hedmark University of Applied Sciences (NORWAY)

Abstract

Many researchers, for example J.P. Gee, have been pointing out for years that game playing can give good learning. When school topics is Gamified we tap into the most important and established learning theories such as, Problem Based Learning, Situated Learning, Social Identity, Experiential Learning (learning by doing and learning just in time) and more. While this alone argues for using game playing in learning, we can add to this list.

These days, children not only play games, but also modify them from early age, and this can be beneficial also in school. Letting the children do their own game development broadens the learning perspective even further and opens up new possibilities. Game development underpins creativity and let the children work with arts, design, animation and storytelling. It also develops technical skills such as programming, system thinking and requires basic understanding of mathematics and physics. And teaching the theoretical science while they are about to code it for their game is way more interesting and effective. And of course, it does so while the children have fun – the children hardly notice the learning because they are making games.

Game development also invites informal learning into the classroom. If the children present their game ideas on the Internet and in game forums, they also get new highly skilled peers, from all over the world. This will also teach them language and cooperation, digital skills and competition, tradition and culture… Internet these days reflects all aspects of the real world.

The children build on skills they already have [1] - most of them have been playing games or developed them in their spare time or even before they started in school. Which is mandatory for good learning [2].

In Hedmark University of Applied Sciences (HUAS) we have used game development in teaching for almost 10 years, both for game students and teachers, and this presentation will give some examples on how to implement this in practice.

HUAS has a complete game development tracks since 2003, and is now one of the biggest and most complete game educations in Scandinavia with 250 game students. HUAS also have various teacher educations and have the latest years also given teacher education in using and development games for learning.

Keywords: Learning, Game Development, Gamification, Games and Learning, Pervasive games, Digital Games, Concurrent Design.

1 INTRODUCTION

Minecraft from Mojang is one of the most popular and most talked about online game or virtual world these days. According to Gamespot [3], Minecraft reached 14.3 million sales and 100 million users in February 2014. Microsoft then bought Mojang in September the same year, and later in October announced that over 60 million copies were sold on all platforms. 18 millions were sold on the PC alone, which makes it the best selling PC game ever. Minecraft is all about building and making games, and the numbers above illustrates how popular such virtual online games can be.

This interest for building and game development is not new; we have seen high sales figures in pc or console games like Sim City or Civilization where players develop cities and societies, or Spore where players design new species and develop them over time. Then came online games or virtual worlds like Secondlife, Wow, Club Penguin, Stardoll etc. Lastly there is a long history of game engines that was firstly developed as engines for specific game titles, but later was developed into engines the public could modify and use to make new games (Valve, Microsoft, Unity, Unreal etc). All of these virtual environment and games has drawn lots of young people to game development, and taught them game development skills like programming, design, arts creation and much more.

While all of these games were very popular, they never reached the fast growth we have seen with minecraft. The reason can be many, but surly the simplicity and how fast you can start playing is an important factor. Minecraft has only one building block, cubes. What we first would think is a limitation has proven to be a force; younger people find it utterly exiting to build large structures from this simple tool. Further, the technology to implement virtual reality (VR) is finally here; in minecraft you can soon use glasses that give the players a 180-degree vision, and gloves or cameras that follow your movements in the real world and translate it into movements in the game. This greatly enhances the immersion.

Having all these young kids with new technical, artistic and social skills, it would be logical to have a closer look at how we can use these to create better learning in schools. How can we design building environments in traditional school topics, so that these skills can be utilized and the kids put the same energy into learning this as they do in gaming? This article take a closer look at this, and outlines a way to implement game development to learn traditional school topics.

Before we do that though, we need a closer look at what these kids really want, and how their game playing can be utilized for learning.

2 THE NET GENERATION

Marc Prensky [1] argues that the “net generation” or “game generation ” is different from other generations. He claims that due to their heavy use of Internet and digital media not only their way of thinking, but also their brain have changed. Prensky point out that these kids, and then the kids that builds in minecraft these days “requires multiple streams of information, prefer inductive reasoning, want frequent and quick interactions with content and have exceptional visual literacy skills – characteristic’s that are all matched in digital game based learning”. They tend to make meaning in an active way, work in teams and play roles and participate, and they combines fantasy and reality and want to have fun while doing it. Also Tapscott [5] elaborates on what these young kids demand from life, education and future employment in interviews about his book “The Net Generation”; “They want to have fun. In fact, 58% of them say that having fun with a product or service is just as important as what that thing actually does. If you employ any of these people, realize that they also want to have fun at work. They want to collaborate and have relationships. They want innovation and creativity. They want speed. They want to customize everything … this group wants to do things their own way".

These kids have a top down approach to learning and they just dive in and use trial and error to solve the challenges. In a learning situation they will be good at making sense from multiple streams of information, but will have little patience for things that don't pay off right away. When they learn they like working with others, also outside their class; being connected to the world is a must. But most of all, learning must be fun.

These are the kids that the schools expect to sit still and patiently listen to lectures on theory they cant use for years, and accumulate knowledge step bye step in just one topic at a time.

We have seen from minecraft that having fun, don't mean inactivity of just meaningless play. Building things is also seen as fun, as long as they can do it in the same way they play games. But to build stuff often requires more knowledge than just playing games. They need to know more about how things work when they want to construct them instead of just using or modifying them; they need to know how physics, mathematics, programming affect and move things as well as how design, art, animation etc. is used to make the environment look good and immersive. Where they build the things is less important; they can have as much fun building in the real world as in a virtual world, provided they can do it their way.

However, there are a few things the real world can't provide in the same way a virtual environment can. Firstly, a virtual environment can combine a mix of real world behavior with fantasy behavior. The physics engine in the virtual worlds can be altered to give a different response to behavior than the real world does. For example, gravity can be changed so that the environment behaves as it does on the moon or another planet. Or the environment can be changed to look different from the real world. Only imagination is the limitation. Gee [2] points out this this is important; these kids want to be able to mix fantasy with reality.

Secondly, creative solutions can be tried out without being afraid of being hurt or take any risks. Only the virtual environment will be damaged, what they do have no consequences in the real world.

Thirdly, you can take on new identities when you talk with peers outside the classroom, or if you are inside a virtual world you can let your avatar communicate directly with the peer's avatars. Your peers knows nothing about your real looks or abilities/disabilities, they will learn to know you from the inside. This enables you to take on any personality you want and gives you the possibility to appear as what you aspire to be, and not what you seem to be from your looks or from what your friends know about your background.

All these "extras" obviously have an effect on the learning. You also get a 'second chance' as often as you like to contribute in a team with the things you are really good at, and are not assessed only by school topics.

So far we have just addressed the kids and what an online virtual environment can give them. Lets take a step further and look at what gaming and game development can add to this and how it relates to learning.

2 GAMES AND LEARNING

Games have been played at all times, but only in the last one or two decades we have been able to play them in a virtual environment. Games are about play, they often mix reality with fantasy, and they are considered fun - just what the doctor ordered for the net generation.

In his book about games and learning, Gee [2] discuss how good games relates to learning. In the following I have recaptured some relevant points from Gees discussion, and made some additional comments to relate them to this context or discussion:

1. Good games are systems and people often learn best when they first understand how things fit into a larger meaningful whole. That you can also simplify the models so that elements and relations are more transparent, adds to this. To start with the whole in a simplified way is the top down approach we talked about earlier; first get the overall view, then dive into the details later.
2. Good games are a set of well-ordered problems that build on each other, and learning events pops up when a problem has to be solved. A good game provide a step-by-step approach to learning in form of related problems that build on each other and leads to the overall goal.
3. Good game is pleasantly frustrating. In a learning context, pleasantly frustrating means that the learning intervals are right; not too hard and not too easy.
4. Good games let the player see skills in a bigger perspective; it is easy to see why they are needed. Training skills is a necessary evil in learning too, and to be able to see the training as a part of a whole, or a part of the game, will make it more meaningful to do such training.
5. Games let the players feel like active agents. The players can participate, be creative and create and modify in their own style.
6. Good games let the players work in teams and take on the identity they want. The players can take on any identity related to the game, and work as a part of a team.
7. Good games are a like a ‘community of practice’; the overall goal seems to be to create a game, while the learning of school topics are ’invisible’ and connected to this goal.
8. Games integrate perception and action. The players can immerse into the game, get an overall view and manipulate things at a distance.

If we build on this when we design a game development class, we should not only achieve a fun and interesting environment for learning, but also be able to design the learning so that the students that create the game learn what we want them to learn (read relevant school topics and goals).

4 GAME DEVELOPMENT AND LEARNING

To make a good ‘game environment’ for teaching we need to define set of interesting problems founded in the curriculum, sequenced them so that the player can and want to solve them, and reward them so they continue to all of them is solved. It must be possible to work in teams, it must be possible to take on different identities, and the world/Internet must be available most of the time.

A Problem Based Learning (PBL) [6] [7] approach will get us started; we can divide the game development process into smaller well-ordered problems that can be solved sequentially to reach the final game goal. PBL also accommodate team-based work where the problems are discussed and solved together.

Below is a suggestion to how to approach this process, based on experience from the examples given in the next chapter:

1. Find the learning goals you want to teach from the overall competency aims or curriculum. There should be both some overall goals and some sub goals.
2. Make a set of problems from the goals. If necessary, sequence them so that each problem builds on the previous one. Make sure the learning curve is not to steep. The problems should be open ended so clever students can take each task to a higher level without having to go to the next task. If possible, design the tasks so they can be solved in different ways.
3. Divide the class into teams or groups that work together. To make the groups as creative as possible, we suggest dividing the class into heterogeneous groups with 6-10 students in each group [8]. As tutors for these groups we suggest to use higher-level students. The groups can serve as a discussion group only if each student makes their own game, or as a development group where the members make a game together.
4. Open the classroom to the world and encourage the class to use the net. This integrates informal learning environments into the classroom. The students should be encouraged to use the net to find facts and theory when they need it. They should also be encouraged to present their solutions and get feedback on their work and progress.
5. Encourage completion between students, groups and online in forums. By letting the students or student groups compete both between themselves and as a group or class on internet you introduce more game elements and more fun

6. Let the students design the game and test it on another class.

7. Sum up the experience with the students and help them structure their new knowledge.

There are many game types that could be used for this. In the examples below on game development classes, we have used both digital games and pervasive games.

It is important that we design the class so that the students can be creative right away, and don’t have to read manuals or study topics before they start. They will have to do that in time too, but not before they start!

If the games developed are not only digital, but for example also include pervasive games, a number of addition skills is needed, like acting, developing costumes, designing locations and clues in the real world etc.

Game development can also invite informal learning into the classroom. If the students present their game ideas on the Internet and in game forums, they also meet new highly skilled peers from all over the world. This can teach them language and cooperation, digital skills and competition, tradition and culture etc. since Internet these days reflects all aspects of the real world.

Ordinary teaching methods should not be abandoned entirely. Some students will need more attention than others, and their learning styles are different. If you think they or just some need an ordinary lecture, give it to them. You will get an attention and response that is different from what you are used too, because it is given just in time and the students are eager to implement their game Teaching the theoretical science while they are about to code it for their game is way more interesting and effective.

3 THE THEORETICAL PERSPECTIVE

The main goal of game development in class is not only to teach the children specific school topics, but also make them interested in learning more about these topics. Interest is primarily defined as a phenomena that emerges from an individual`s interaction with the environment [9. Leaning on Dohn [10], Interest can be defined as a positively charged experience that relates to an activity [11]. In today's research there is primarily focus on two types of interest; situational and individual interest. Situational interest is about stimuli from the environment, which may or may not last over time [9]. Individual interest is more stable motivation that is associated with increased knowledge, value or positive feelings [12]. Situational interest focuses on individuals' response to external factors that my give interest in a certain context, while personal interest focuses on the individuals experience built from enduring preferences. These two types of interest are not in any way dichotomous, but they can influence each other's development. When a student is interested in a certain learning situation, it is most likely an interaction between both types of interest. Situational interest can over time have powerful influence over the students' personal interest in a topic [12].

The students Interest in game development will then depend on both on the social environment (the class as well as and activity in online forums or virtual worlds) and the students' personal interest in game development. We can expect that the social environment will make a difference on the personal interest in game development, but more important, also on the personal interest in the topics the student learn, and how much energy the student put into studying them.

The game development process itself serves as a "community of practice" [14] [15] that invisibly triggers interest in school topics disguised as game development, while they strive to reach a common goal; to learn to create a game. They do this by experience [16], game development is "learning by doing" [17], and they study theory only when they need to solve a game development problem, "which is learning just in time" [2]. With open-ended cases the game development process can accommodate different learning styles [18]. If the game development progress is designed so that it is "pleasantly frustrating" with the right learning intervals, Vygotskys' [19] approximation zone will not be to big, and maintain the players lust to try again if he fails. A Problem based learning approach facilitates sequenced problem solving where each problem prepares the player for progressing to later parts of the games and finally the final game goal. PBL also facilitates team-based work where the problems are solved together as a team [6].

Teaching through game development then taps into many of the mot widely used learning theories such as Problem Based Learning, Situated Learning, Experiential Learning, Identity Construction etc. which in itself should indicate that it s a valuable tool for creating Interest and good learning.

5 HOW CAN GAME DEVELOPMENT BE IMPLEMENTED IN SCHOOL? SOME EXAMPLES

In Hedmark University of Applied Sciences (HUAS) we have used game development in teaching for almost 10 years, both for game students and teachers. Here are some examples on how and where it has been used I practice:

The Programming courses (2006-2011)

The students attending this course had just started out their education and came from 3 different bachelors in IgA (Interactivity, games and Arts); Animation, Arts and Game Technology. Typically, around 70 students attended the course. Most students were between 18 to 25 years old, and were typical examples of the net generation described in the beginning of this article.

So we gave the students a simple task they loved - to create a game! Our main goal however, was to tech them computer programming and give them an idea of why 'hardcore science' topics was needed in a digital arts education.

We wanted to utilize a programming environment that gave visual feedback to the students right away - an environment where the programming, mathematics and physics would show up visually as behavior or environment changes in the game graphics. At the time Adobe Flash was a highly popular tool to make games and animation movies, and we found that this environment also satisfied our needs for visual feedback as well as giving us a good object oriented programming environment.

We divided the game development process into 6-10 smaller problem based cases that was given to the students as sequenced assignments over the semester. Each case were1-2 week long, dependent on how difficult the case was. All the cases were designed open-ended so that diligent and clever students could develop them beyond the basic requirements without proceeding to the next case. We started with cases to design, draw and animate the characters, and then moved on to programming cases. When they reached the programming cases they were very motivated to make their design move, which required programming.

The students were divided into groups, with 6-10 students in each group. Each groups was given a tutor from a higher-level class. The groups met twice in the case period, each time about 3 hours. First time was in the start of case period, when they needed to define the learning goals and the theory they needed to study to solve the case. After this meeting they separated, studied the theory individually, and then completed the case in the best way they could. After that they met again and discussed the challenges the case had given them, and how they had solved them.

We choose the Maasterischt model with 7 steps [7] as the teaching model. The first 5 steps in this model; clarify difficult terms, define the problem, brainstorm, analyse and formulate the learning objectives was done in the first meeting. Step 6 was self-study period, and step 7 was the post-discussion and sum-up meeting in the group.

Many opportunities to teach mathematics and physics popped up naturally during the cases. For example, when the students had finished their game characters and animated them, they needed to make them jump. They usually did this by applying a force upwards, which made the characters disappear out of top of the screen. To "fix" this they had to study theory about how forces worked in nature, and especially how the gravity force drag everything back down to "earth". They had to dig deep into this theory; natural looking character jumps and walks required a good understanding of different forces, gravity and friction. Very few of the students found it odd to study forces and gravity in this situation. Another example was when the students needed to implement their characters' walking cycles and moves around the screen. Then they needed understand about functions, x- and y-axes, coordinates, angles, and conversion between degrees and radians. Eventually, they also had to learn trigonometry to be able to divide the movements into horizontal and vertical components. Students often struggle to understand this kind of theory if it is taught on an abstract level. Here, however, the physics and mathematics became comprehensible because they needed the knowledge to make their game work. In other words, the ‘fun’ and ‘motivation’ aspects of the game development made the students see the ‘hardcore science’ as elements in the overall game development process, and because of that they were willing to put much effort into understanding it.

Throughout the course the students had to present their games online in wiki pages and blogs. This enabled them to present and discuss the project in online forums, and discuss their problems and solutions with online peers. Most wiki pages and blogs were written in English. This was not a requirement, but for most students it was a natural thing and something they had done for years.

Research data was collected from notes, tutor meetings, student reports and interviews. All the data are not yet fully interpreted, that will be done through Grounded Theory and presented in new papers. For further details about this project, see article [20] [21].

4.2 The System thinking and Gamification courses (2012-now)

This project is ongoing and is about teaching mathematics and physics through system thinking and gamification. In the first course the students learn systems thinking and simulation doing 9 sustainability related projects. In all these projects they implement the differential equations generated by the simulation engine using Euler in C# in Unity game engine where they also make simple visual applications to see how the differential equations work.

The Gamification course follows the system-thinking course, and in this course the students gamify a larger sustainability project using the knowledge from the previous course. This class is manly focused on game development (in unity so far, but minecraft has been discussed for the next years course).

In both courses we use a problem based learning approach as outlined earlier in the article. More details on these projects can be found in [20]

The Pervasive game project (2013)

This project was done in 2013 and was conducted in an IgA class on system thinking, where they developed a pervasive game to learning sustainability. The pervasive game also included 3 digital games. We wanted the game to be tested in a real environment, so we invited elementary schools in the district to volunteer. The response was massive, 15 schools called within two days, and we selected Tangen elementary school for the project. The game was tested in one day each for the 5th grade and the 7th grade classes. After the game we let the children fill in a diary that was about what they have learned, and lastly we conducted a one-hour system thinking session for the whole class. These two events plus notes from the research team and reports written by the students served as research data. These data is not analyzed yet, but will be in a future article through grounded theory.

The pervasive game was based on competency aims in the Norwegian curriculum related to sustainability. We wanted both the game students and the children to understand how they could support a long-term ecological balance, and to teach them about ecosystems, pollution, the social aspects of sustainability and point out how sustainable systems could be used to live wisely. We also wanted to apply system thinking, collaboration and problem-based learning (PBL) both in the game development process done by the students, and later when the children was playing the game.

By observing and analyzing motivation and learning when the game was created and later when it was played, we hoped to gain new knowledge on how good pervasive games was suited for learning applications. We also wanted to measure other outcomes such as collaboration skills, creative problem solving, and even physical activity. Finally, we wanted to examine how well the children were able to understand systems thinking, and how important systems thinking was in their understanding of the sustainability issues presented in the game. More details on this project can be found in [22].

The Teacher courses in Games, Animations and Learning (2013-now)

These courses were given to teacher students that were in their last year of their teacher education (4th year). They then had a thorough teacher and pedagogics background, but hardly any digital experience or even less experience in creating games. The challenge here then was to teach them what they needed during their game development process.

We threw them into deep water right away; the assignment was to develop a game for their own education. We used the same Problem based model as described above; the development process was divided into problem-based steps and the students were divided into working groups. Before we started we also let them develop simple web pages and Facebook groups where they could document the game development process

Since the teachers were familiar with pedagogics and learning goals, the first part of the development process went smoothly. The brainstorming process for game ideas also went good; they had a lot of ideas for the games. However, they did not know much about developing games, so at this point we needed to teach them some game design theory. We also in this step showed them how to design a storyboard and how to apply system thinking in their game design. These classes were a mix of lectures and working sessions where the groups and the supervisor worked together to help them develop their own games. In this way they managed to develop pretty interesting pervasive games.

Previously to these sessions we there had been classes in making Kodu games, and many of the teacher groups choose to develop Kodu games as a part of their pervasive games. During the whole process they blogged and documented their developments. When they had developed their games they took it to their practice or work to test them in their classes. Pretty much all of them worked well and the children had fun playing them.

I general we had no more challenges developing games with the teacher students than we had with the game students. They did great games too, and teaching them about game design for the pervasive games and even the digital games was not hard. Of course the technical level of the game development was much lower than for the game students, but the tools they used proved to create be very useful and easy, and the children loved to play them. Examples on games and implementations are given in [23]

6 CONCLUSIONS

In all our projects in game development for learning during the last 10 year, game development has proven to be a valuable tool to create interest in the school topics. This is not so surprising, considered the interest we see for building in online forums and virtual worlds.

There are some challenges though, specially in tuning the cases so that they get the right learning intervals. It took us several interactions to get this right. It is also hard to synchronize the game development process with the learning process; game development processes tend to naturally sequence what the developers need to learn. Further, these types of projects are also a bit chaotic; we found that the students needed help on structuring what they learned after the game is done.

Games are also very cross disciplinary; there are very few topics game development or game playing will not touch. Therefore it can also be possible to involve a whole class in just one game development process; some do art stuff like design, animation and digital effects, while others do technical stuff like programming, system thinking, math and physics. If the game is a pervasive game, it will also need actors, administrators, helpers etc.

If you want to use game development in learning we encourage you to read [20] to [23] more closely to get some more details on what challenges we struggled with along the way.

We can assure you though, game development and testing in a class situation is fun, and as a teacher you will be amazed about how creative these young kids can be.

REFERENCES

[1] Prensky M. (2007), *Digital Game-Based Learning*, Paragon House, the United States

[2] Gee, J. P. (2007). *Good video games and good learning: Collected essays on video games, learning and literacy.* New York: Peter Lang International Academic Publishers.

[3] Gamespot: Minecraft passes 100 million registered users, 14.3 million sales on PC (2014, Feb 26), Retrieved from: http://www.gamespot.com/articles/with-minecraft-microsoft-can-build-broader-horizon/1100-6422330/

[4] Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

[5] Tapscott, D. (2008). *Grown up digital: How the net generation is changing your world.* New York: McGraw-Hill.

[6] Barrows, H. S., & Tamblyn, R. (1980). *Problem-based learning: An approach to medical education.* New York: Springer.

[7] Maastricht University. (2013). *Problem based learning preparatory website.* Retrieved from http://www.umpblprep.nl/

[8] Darsø, L. (2001). *Innovation in the making.* Frederiksberg: Samfundslitteratur.

[9] Krapp, A. (2002). Structural and dynamic aspects of interest development: Theoretical considerations from anontogenetic perspective. Learning and Instruction, 12, 383 – 409.

[10] Dohn et al.(2010), Situational Interest of High School Students Who Visit an Aquarium, DOI 10.1002/sce.20425, Published online 18 October 2010 in Wiley Online Library (wileyonlinelibrary.com).

[11] Rheinberg, F. (2008). Intrinsic motivation and flow. In J. Heckhausen & H. Heckhausen (Eds.), Motivation as action (pp. 323 – 348). New York: Cambridge University Press.

[12] Schiefele, U. (1991). Interest, learning and motivation. Educational Psychologist, 26(3 – 4), 299 – 323.

[14] Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

[15] Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

[16] Kolb, D. A. (1983). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice Hall

[17] Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

[18] Overview of Learning styles (2016, May 16), Retrieved from: http://www.learning-styles-online.com/overview/

[19] Vygotsky, L. (1978). "Interaction between learning and development." (pp. 79-91). In Mind in Society. (Trans. M. Cole). Cambridge, MA: Harvard University Press.

[20] Nordby, A. (2015) Teaching ‘hardcore science’ to arts and design students from the gaming generation - Reflection on how to use game development to teach programming, system thinking, mathematics and physics. Proceedings EDULEARN15, Barcelona.

[21] Nordby, A., Karlsen, S. (2014). Teaching ‘hardcore science’ to arts and design students: Reflections on the development of a basic programming course. *InFormation - Nordic Journal of Art and Research* 2014 ;Volum 3.(2) s. 129-142

[22] ”The best school day ever’ – Teaching Sustainability and System Thinking by Pervasive Game playing” (Nordby, Øygardslia, Sverdrup)

[23] TEACHERS AS LEARNING GAME DESIGNERS: Can elementary school teachers with no background in game design really Gamify their own teaching?

3

“THE BEST SCHOOL DAY EVER” - TEACHING SUSTAINABILITY AND SYSTEM THINKING BY PERVASIVE GAME PLAYING.

Anders Nordby[1], **Kristine Øygardslia[2]**, **Ulrik Sverdrup[3]**, **Harald U. Sverdrup[4]**
[1]*Hedmark University College (NORWAY)*
[2]Norwegian University of Science and Technology *(NORWAY),*
[3,4]Lund University (SWEDEN)

ABSTRACT

“The best school day ever” was a comment from one of the children in 5th grade elementary school after a day of pervasive game playing in school. The children hardly noticed the learning when playing, but tests after the game reveled thorough knowledge acquiring in Sustainability and System Thinking. The task was simple; game development students from a university course in game design and *system thinking* were asked to developed and test a *pervasive game* for teaching *sustainability* and *systems thinking* to children in elementary school. The overall game goal was to “sneak in” learning while the children were having fun. This paper focus on two aspects of the project; firstly the development process done by the university game students, and secondly the game days in the elementary school where children played the game.

Pervasive games (PG) are an emerging genre of games, blending the real world and games in new and exciting ways. PG has many properties that make them suitable for learning. They do, for example, facilitate collaboration, learning just in time and creative problem solving. In addition, they provide a natural environment for the problems that should be solved and the theory the players have to study. *System thinking* is about discovering causal connections between different variables, and understanding how different components in the world are connected to make a system. *Sustainability* is about the capacity of Earth to endure human life, and is an important part of the Norwegian elementary school curriculum. The learning goals were taken from the Norwegian Core Curriculum and Subject Curricula.

Keywords: Gamification, Game Development, Pervasive Games, Games and Learning, Pedagogy, System Thinking, Sustainability.

1 INTRODUCTION

Playing computer games is one of the favorite activities for children and youth in Western countries. Research from Statistics Norway shows that 60% of the boys and 40% of the girls aged 12-16 played computer games on a given day [1], and research from Entertainment Software Rating Board (ESRB) indicates that the trends are the same in the rest of the Western world [2]. Lately, researchers such as James Paul Gee and Katie Salen have shown that computer games have characteristics that make them excellent learning platforms [3]. Slowly this is also discovered by the education systems in many countries including Norway [4].

Although the age of the average gamer is increasing, many teachers and parents understand little of what computer games are, and how children play them. There are several ways to define what a game is, but this is elegantly explained by Sicart: «Playing games is interacting with systems that have been created with the intention of encouraging their users to perform a number of actions to reach some predefined goals in pleasurable or engaging ways» [5]. This is an accurate definition on many levels: games are about *interacting with systems*, reaching predefined *goals,* and having *fun* while doing it. This also links games to systems thinking, and indicate that the game itself is a system.

While most of the research on games and learning has focused on computer games, there is another genre of games that is very interesting to look at. These games are called *pervasive games,* and are games that blend digital games and the real world in new and interesting ways. Similarly to games, there are many ways to define what pervasive games are. Lee Sheldon proposes the following definition: “A pervasive game is a game where the line between the game and reality blurs to the point where it is difficult to distinguish one from the other" [6]. However, it is important to note that pervasive games consists of a wide range of different types of games that blend games and reality in different ways. They can vary significantly in nature, from Urban Adventure Games where the goal is to learn about a city in a new way by solving puzzles based on GPS coordinates, Street Smart Sports, which are physical adaptations of digital games, like PacMan, or Alternate Reality Games, focusing on an online narrative and collaboration [7]. Jesse Schell has a tight definition of Alternate Reality Games, capturing much of the essence of the genre: "An ARG is a type of pervasive game that uses the real world as a platform, often involving multiple media and game elements, to tell a story that may be affected by participants' ideas or actions".

Alternate Reality Games have been used for marketing purposes for a long time. One example is the game "Why So Serious", which was used to promote Batman: The Dark Knight, praised as being one of the most successful marketing campaigns of our time. In this game, more than 10 million players from more than 75 different countries collaborated to reach a common goal, by finding and sharing information, helping each other solve tasks, and participating in real life events [8].

Although pervasive games have been used a lot for marketing purposes, they also inhibit characteristics that will make them well suited for learning purposes. As seen in the previous example, many of the games facilitate collaboration to a large extent. When retrieving and interpreting information, learning events occurs *just in time,* and the learning process occurs in a natural context, instead of in a constructed environment.

System thinking is not only useful in game development. It is also an important tool to understand sustainability, and a good tool to learn logic reasoning and enhance the understanding of how parts work together as a whole. Because of this we wanted to include systems thinking in both the development process of the games, and as a learning aim for the children who played the game. While system thinking is an important topic in the American K12-educations, it is yet included in the Norwegian school curriculum. Since games are system too, it is interesting to see how well games and systems thinking used together facilitate learning and education*,* or personal and cultural maturation [9].

Both in game playing and in system thinking the learning is done by solving problems, so Problem Based Learning (PBL) is also a cornerstone in the project.

2 SCOPE AND OBJECTIVES

In this research project we wanted to develop a pervasive game for learning, based on competency aims in the Norwegian curriculum related to sustainability. We wanted both the game students and the children to understand how they could support a long-term ecological balance, and to teach them about ecosystems, pollution, the social aspects of sustainability and point out how sustainable systems could be used to live wisely. We also wanted to apply system thinking, collaboration and problem-based learning (PBL) both in the game development process done by the students, and later when the children was playing the game.

By observing and analyzing motivation and learning when the game was created and later when it was played, we hoped to gain new knowledge on how good pervasive games was suited for learning applications. We also wanted to measure other outcomes such as collaboration skills, creative problem solving, and even physical activity. Finally, we wanted to examine how well the children were able to understand systems thinking, and how important systems thinking was in their understanding of the sustainability issues presented in the game.

3 THEORETICAL POINT OF DEPARTURE

Both the game developed by the students and the game playing done by the children is gamification, and consequently both share the same theoretical foundation. Kewin Werback [10] points out that gamification draws upon cognitivism to create extrinsic motivation by using for example rewards, but also that gamification create environments on its own that can trigger intrinsic motivation. In these environments the player can experience the whole spectrum of an action, from making a decision to facing the result of the action, and the player internalize knowledge by reframing their mental representation with what they experience in the game. This frames the gamification activity within the theory of Constructivism.

In both settings the learning would be experiential – the students or children will learn through experience, and will have to reflect on what they do when they develop or play the game. This frames the game development and the game playing within the experiential learning theories [10]. Further the game development class or the game playing serves as a community of practice where the students or children participate and work towards a common goal (developing the game or solving the game), which frames the project within situated learning [11], [12], [13].

4 CASE DESCRIPTION

The work started by finding competency aims for elementary school [15]. The Norwegian curriculum for elementary school is divided into competency aims, and several grades often share a set of competency aims. We did, for example, find that 5th – 7th grade share the curriculum for their subjects, such as social studies and natural science. We decided on a set of competency aims related to sustainability with the core competency aim to make the children able to: *“explain how production and consumption can destroy ecosystems, and pollute earth, water and air, and to discuss how this can be prevented and repaired”.*
In addition, we decided on a set of secondary competency aims that the game should include as well, trying to get a balance between the different components of sustainability, including the social dimension:

- Explain the connection between natural resources, industries, residences and lifestyles
- Explain how we in Norway use resources from other places in the world
- Understand the refugee situation in the world, and explain why some people flee form their country. Discuss how it feels to come to a foreign country as a refugee
- Retrieve and use relevant information from texts about natural sciences from different media

After the goals were selected the students decided to develop and include three digital games in the overall pervasive game. These games were as follows:

- A digital game showing the relationship between human consumption and the ecosystem, and how we can prevent and repair the pollution of earth, water, and air. (The Island Game).
- A digital game showing refugees what it is to be like to be a refugee (The Refugee Game).
- A digital game showing ecological principles from the perspective of an earthworm (The Earthworm Game).

We wanted the pervasive game to have a well-developed narrative and interesting characters, and include elements such as websites, clues in text and images, videos, and consistent game design. In addition, the smaller digital games should be included in a coherent way. The story, clues, characters and gameplay should all be aimed at teaching sustainability related content, and the game should be possible to play in one school day, which essentially means 5-6 hours. To create a coherent and logical structure, both the smaller games and the overarching pervasive game should be designed and analyzed using systems thinking.

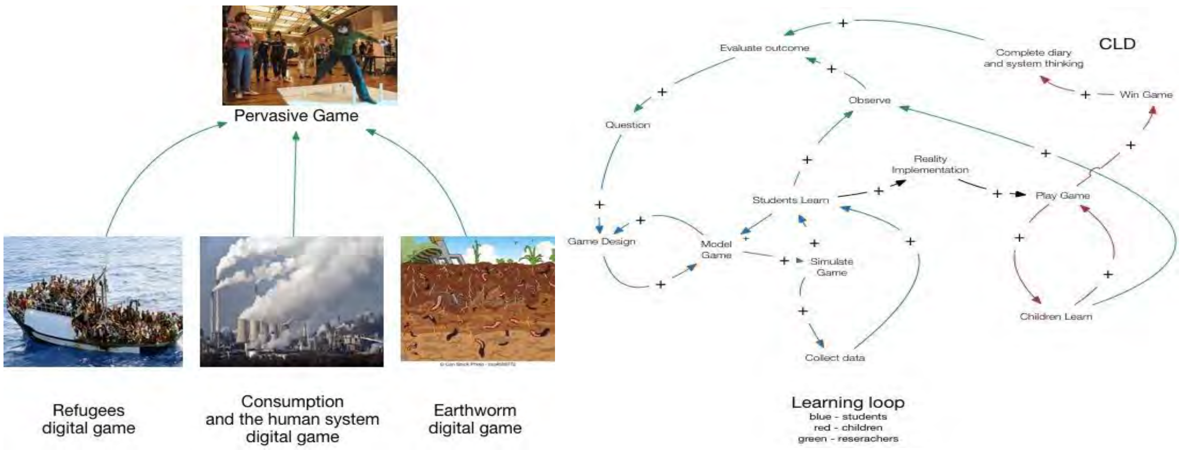


Figure 1: The pervasive game and a Casual Loop Diagram (CLD) of the settings for the game.

4.1 The Pervasive Game

The pervasive game was the overall game the children would play in the real world; in this case inside their school and in the outside schoolyard and local community.

The story starts with a refugee that had to flee from his country because of famine and war as consequences of pollution and climate change. The children find his video blog, and are given the task to find out what has happened to the environment in his country, and how the pollution and climate change could be prevented. The children were divided into teams, and each team received a backpack with a pair of walkie-talkies, some riddles and clues, a treasure map, a note with computer passwords and a notebook. They then had to solve clues on different locations inside the school and in the schoolyard and talk to actors about sustainability-related issues to be able to proceed in the game.

Some of the game development students proved to have talent as actors and played the game characters. One example of a character is the gardener that talks with the children about ecology, biological diversity and pollution. The children help him plant vegetables, and in return he will give them new quests related to trading. Another character, the King of earthworms was, according to local legends a mythical creature. He is about the size of a man and is residing in the area around the school. This “worm” talks to the children about the importance of the small creatures in the ecosystem. The locksmith helps the children create a key for the treasure chest they will find, using recycled metal, and talks to the children about the importance of recycling. The merchant will give the players trade-related quests, and teach them about fair-trade and resources. The refugee shown in the video that starts the game will return towards the end of the game to talk to the players, and discuss their experiences.

All children teams were appointed one guide from the students that would help them play the game, but the children fast exhausted their guide, and proved to be very good at solving the challenges on their own.

Below are a flow chart and the CLD for the game.

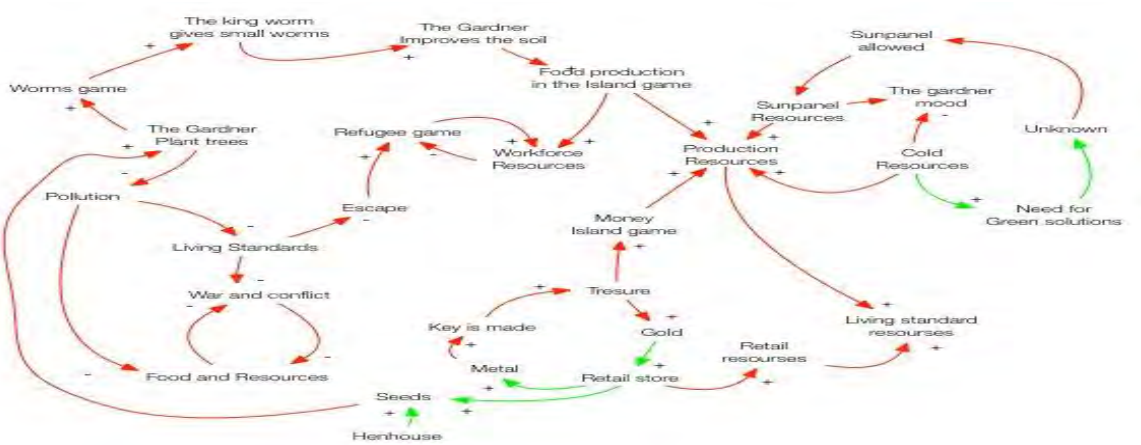
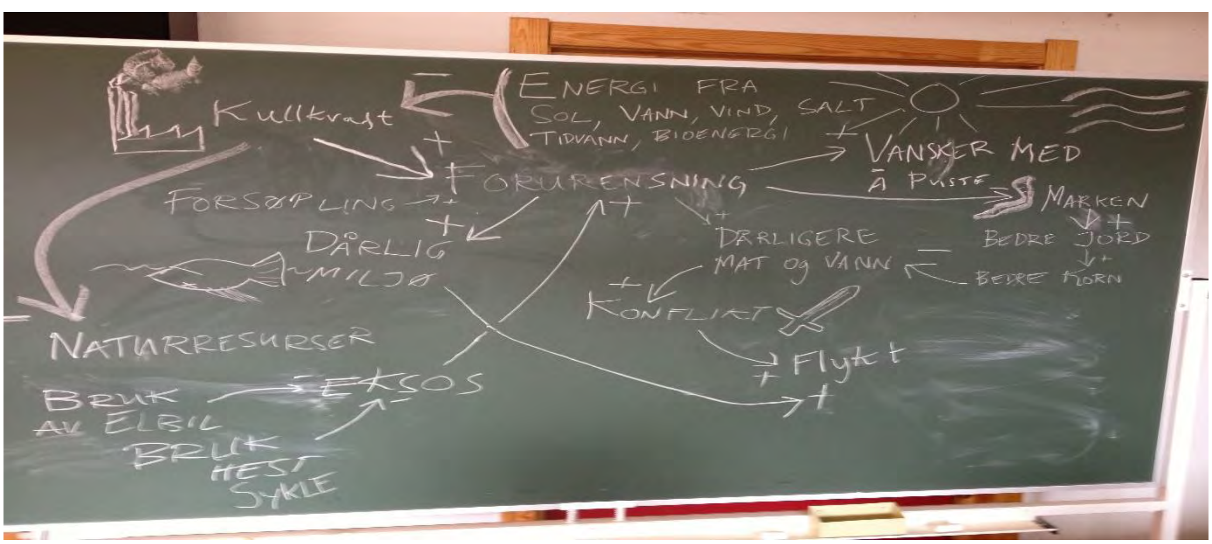


Figure 2. The Pervasive game CLD and float diagram developed by the students.

The CLD gives an overall perspective on the pervasive game, showing how the Island game is dependent on the other small digital games and what the children will have to do to finally find the treasure. For example, the earthworm game has to be played and won to get food resources, and the refugee game has to be resolved to get a workforce for the fabric and the power plants. Before anything can be bought in the island game, the children will have to find the treasure to get money. The CLD also shows how the pollution, war and food/resources are the basis for the refugee escape, and how earthworms are needed to make food grow.

When the game was completed, the children were given a copy of the refugee's diary. This diary had two functions; firstly it was a way to summarize what they had learned in the game and organize it in a more systematic way, and secondly to measure what the children had learned by giving them small assignments.

Finally, a system thinking session was conducted with the whole class. A teacher asked questions about how things they had learned in the game was connected, while another teacher put the answers from the children on the blackboard in form of a CLD. The children helped putting arrows and signs in the diagram. A pic of the blackboard is included below:



4.2 Human consumption, pollution and the ecosystem (The Island game)

This game encompasses many of the main aspects of sustainability. The focus of the game is on the relationship between human consumption and the ecosystem, and how they can prevent pollution of earth, air and water and live responsibly and become ethical consumers. A natural consequence of this is how societies deal with existing economic, social and ecological challenges; a key part in the education for responsible living [16]. This game utilizes systems thinking to a high degree to show both the students and the children how the ecosystems and human consumption are linked together.

The goal of the game is to build, balance and trade the recourses of an island. It is designed to teach the children about the balance in the nature; how we need factories and products, but that we can make them in a more sustainable way than we currently do. It teaches them that we need to think about where food comes from, that the production and consumption create waste and CO_2, and that the waste can ruin nature and make living very unpleasant. Finally the children see how an island can prosper if they have fugitives come to their place, but also how the island will need to adjust to the fugitives which also need food and homes.

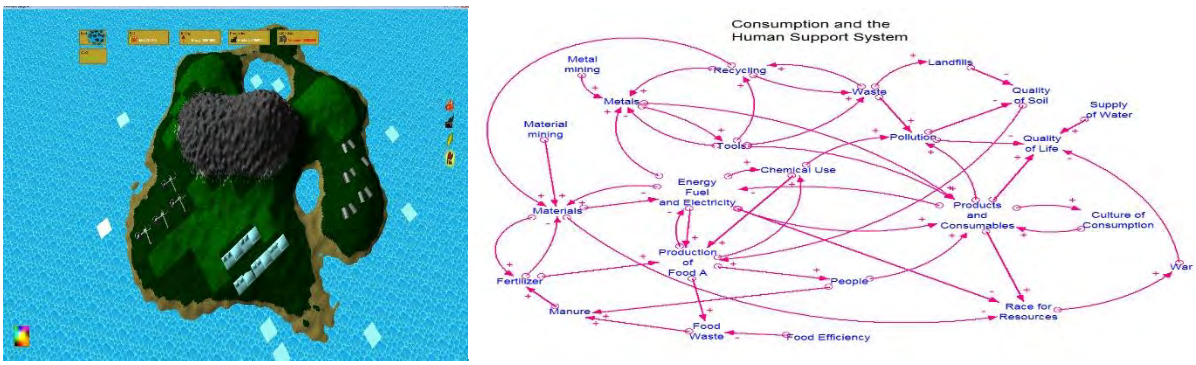


Fig 3. The Island game screen and the CLD

The CLD shows how an increasing population leads to the production of more products and consumables, and to more pollution and waste. The amount and quality of water and food will suffer, and can lead to a race to find new resources and even to war. Mining of materials and the production of fertilizers can increase the production of food, but they require more mining of metals. Recycling can also increase the availability of minerals and metals, but increases the amount of waste.

4.3 The ecological principles of the soil from the perspective of an earthworm (the earthworm game).

While the previous game showed the larger picture, depicting how different components in the world were linked together, *the earthworm game* focus on how small things in nature also can have an impact on the larger system.

The core goal of the earthworm game is to get the best possible harvest, which the player can achieve by moving earthworms to the different parts of the soil (see picture below) and use as little pesticides as possible.

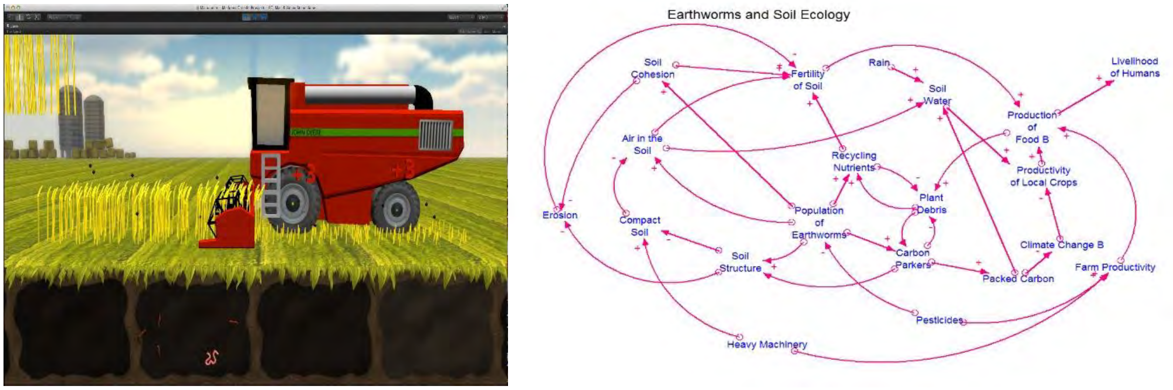


Figure 4. Shows the earthworm game screen and the game CLD.

The CLDs shows how a bigger population of earthworms gives a less compact soil structure and let air, water and nutrients flow into the soil and give the soil better fertility. The farm productivity increases both with the use of heavy machinery and pesticides and leads to a larger food production, but the CLD also shows how the heavy machinery leads to a more compact soil and that fertilizer kills the earthworms. Lastly the CLD shows how soil with a healthy population of earth worms leads to a better soil structure that is less exposed to erosion.

4.4 The refugee situation in the world (the refugee game)

This game focuses on how it feels to be a refugee. The only goal in the game is to avoid police and military and collect enough food on the flight from your country towards a new one. If you survive you can apply for asylum in the new country. The chance to get asylum is based on statistics from the real world.

Research has shown that computer games have a special ability to foster empathy and understanding, since the players not only watch a situation, but are able to experience it themselves by playing through it [17]. This game was based on the competency aim of being able to "*discuss what it can be like to come to a foreign country as a refugee".*

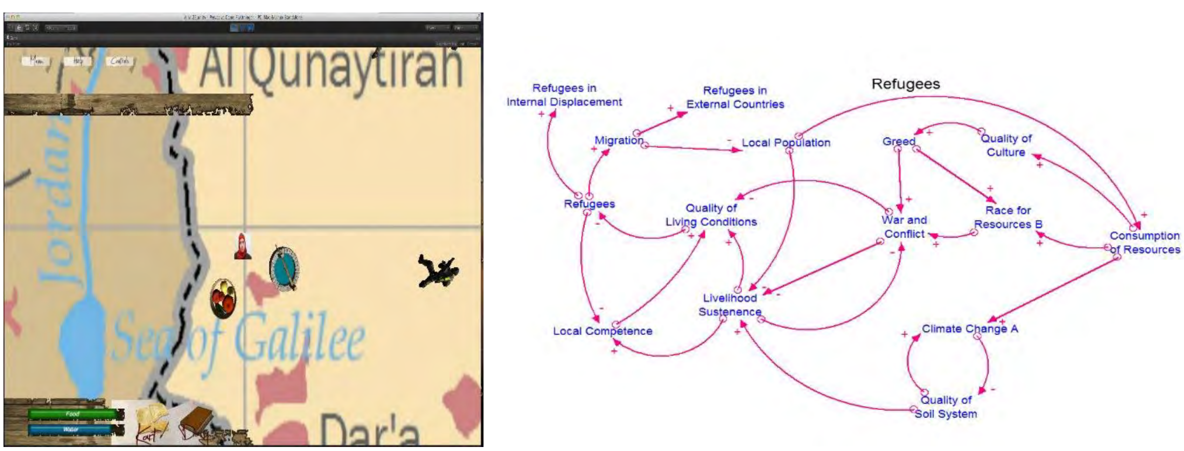


Figure 5: Refuge game screen and CLD

The CLD shows how quality of living and war leads to more refugees, and how more refugees leads to higher consumption, greed and a race for resources, possibly war, climate change, and lower quality of the soil system and livelihood subsidence.

5 DISCUSSIONS

First of all, it quickly became evident that both game development done by the students and the game playing done by the children gave them a common goal, to develop a game in the student case, and to win a game in the children's case. The development class or the game day became vibrant communities of practice [11], [12]. Hence, the gamification approach was highly successful in that it provided both the students and the children with a collaborative social environment in which they could practice according to their skills, and where the newcomers could learn from the more experienced.
This social environment and community of practice also successfully provided a "learning by doing" [18] context largely because of the PBL approach, where the theoretical learning appeared "just in time" [3].

5.1 Comments on the game development class

It is clear from reading the student reports and blogs that they enjoyed participating in this research project. Some of their comments include:

- "It has been very fun to participate in this project, with the best part being the playtest with the 5th graders. I am satisfied with the whole project, and although it was hard to see how everything would turn out when we started the project, we managed to finish, and we finished with bravura."
- "I see the pervasive game part of this course as an interesting experience where we learned a lot, and comments like "The best school day ever", indicates that the 5th- and 7th graders felt the same".
- "This was a very fun and rewarding assignment to work with, and although we're neither actors or have developed a game like this before, I felt it was a very successful project for us students, and I hope the teachers feel the same".
- "To summarize, I'll say this project has been a joy to work on. I've both learned from it, and gained many very nice memories. This has been nothing but fun!"

The students was very impressed by how fast 5th-graders and 7th-graders was adopting the system thinking, and they saw system thinking as a very valuable tool for the children. They also saw the value in system thinking in the game development process, but a few of them pointed out that it maybe was overkill in this small project. An interesting observation for both teachers and students was how much faster the children adopted the system thinking than the students did, and how fast they saw the value it had in the sustainability context. More details and reflections on learning by game development can be found in [19] and [20].

5.2 Comments on the elementary school game playing

The pervasive game served as a creative playground where the children could test and use their creative skills and apply the lessons into practice in a creative way. While the digital games often feel as being off-reality, the pervasive game felt very real, making the learning experience for the children feel more related to reality.

We found that it was beneficial to have small lectures in sustainability and system thinking during the game, and longer sum ups after the game was played (in this project the actors did the teaching during the game, while the diary and system thinking is examples on lessons after the game). I both cases the children proved to be highly motivated having the experiences from the practical tasks they solved during the game fresh in mind.

From our observations and conversations with the children, it seemed like the game functioned as "stealth learning": They learned without noticing while they were having fun.

The children's teacher told us that the children that normally didn't like the traditional school teaching excelled while playing the game. They paid attention, they progressed quickly, and they showed leadership skills. This shows that teaching through pervasive game playing in school also can help to activate children that not so easily fit into the traditional school system.

After playing the game, some of the children came to the library to borrow books on sustainability to learn more about what they could do themselves to save the planet.

Competition seemed to be an important motivational factor for both the 5th-graders and the 7th-graders. When playing the digital games, they wrote their high scores on the blackboard, and competed to beat each other's high scores. Competition also seemed to be a motivational factor when playing the overall game, since all the teams was competing to finish the game as fast as possible. The competition aspect wasn't originally a part of the game design.

We found that the game promoted organizational skills especially well. This was especially apparent among the 5th graders. When they got their backpack, someone usually took the leader role, and divided the different tasks among the rest of the team.

We also saw examples of creative problem solving skills. As expected, the walkie-talkies were very popular, and used in creative ways. While they were expected to use the walkie-talkies for dividing the teams so that they could solve the tasks more efficiently, they soon figured out how to "hack" into the other team's walkie-talkies. They then started negotiating on information exchange. In addition, there were two girls on one of the 5th-grader teams who both were able to speak German. They started to speak only German to each other, so that the other team could not understand what they were saying, and thus keep the relevant information to themselves. The walkie-talkies were also used in a creative manner to solve the different tasks. In one instance, the pupils were supposed to solve a quest by trading items and information between a merchant and gardener, who were actors played by our students. To solve this quest in the most efficient manner, they decided to give one walkie-talkie to each of the characters, so that they could solve the problem themselves.

The children were actively participating in the systems thinking part after the game. They were fast able to identify the different components based on what they had learned from playing the game, and to see the enforcing and balancing factors. In many ways, the pupils understood the systems thinking better and faster than many grown-up students did. Some of the research assistants claimed that this was the most successful part of the day.

Although 5th and 7th graders share the same curriculum competency aims, there were more differences between the 5th and 7th graders than expected, both in terms of knowledge about the topic, and how they played the game. While the ten-year-old 5th graders were highly enthusiastic from the very beginning, the twelve-year-old 7th-graders were a bit more skeptical in the beginning. The reason might be that they were in an age group where playing games were not perceived as being "cool" anymore. But after a while, it became apparent that when the children in the class that were perceived as the most popular really enjoyed the game, it was OK for the other children to get immersed in the game too.

The overarching game was probably more suited for the 5th-graders. They found the characters in the game especially amusing, and enjoyed talking to them and getting to know them. For the 7th graders, some of the pupils seemed to think that the characters were a bit too childish. However, the 7th-graders seemed to love the digital games to an even larger extent than the 5th-graders.

6 CONCLUSION

We conclude that pervasive games to teach systems thinking and sustainability are a good match; the project was a success. Both the students developing the game and the children playing it had fun. Even if it was a success, some factors could be improved:

- A more thorough preparation in systems thinking, sustainability and pedagogics would benefit the students.
- More time is needed to develop the pervasive and digital games.
- A bit more time for debriefing the children after the game would be beneficial to help them structure the perceived knowledge the got through game playing.

- We believe we could have added more traditional lecturing during the game without interrupting the creative flow too much.

For us the whole project was an interesting learning experience that we think is beneficial to work on further to make it into a well-proven concept to enhance creativity and motivate learning for both students and children in the school system.

REFERENCES

[1] Kulturdepartementet, (2008). *Dataspill.* Retrieved on http://www.regjeringen.no/nb/dep/kud/dok/regpubl/stmeld/2007-2008/stmeld-nr-14-2007-2008- html?id=502808

[2] ESRB, (2013). *Video Game Industry Statistics*. Retrieved on http://www.esrb.org/about/images/vidGames04.png

[3] Gee, J. P. (2010). New digital media and learning as an emerging area and "worked examples" as one way forward. Cambridge, Mass., The MIT Press.

[4] Det Kongelige Fornyings- og Administrasjons-departement, (2007). Retrieved from https://www.regjeringen.no/en/the-government/previous-governments/ministries-and-offices/government-secretariats-and-ministries-s/ministries-since-1814/ministry-of-government-administration-an-2/id449386/

[5] Sicart, M. (2009). *The Ethics of Computer Games.* Cambridge, Mass., The MIT Press.

[6] Sheldon, L. (2011). *The Multiplayer Classroom.* Delmar Gengage Learning.

[7] Montola, M., Stenros, J., Waern, A. 2009. *Pervasive Games.* Focal Press US. Norgesuniversitetet. 2013. *Spillbasert Læring.* Retreieved on http://norgesuniversitetet.no/prosjekt/spillbasert-laering

[8] 42Entertainment. 2013 *Why So Serious? The Dark Knight.* Retrieved on http://www.42entertainment.com/work/whysoserious

[9] Vettenranta, S., & Erstad, O. (2007). *Mediedanning og mediepedagogikk : fra digital begeistring til kritisk dømmekraft*. Oslo: Gyldendal akademisk.

[10] Kolb, D. A. (1983). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice Hall

[11] Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

[12] Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

[13] Wenger, E. (2006). Communities of practice: a brief introduction. [Online] retrieved 5 August 2011 from Ewenger.com

[14] Wadsworth, B. J. (2003). *Piaget's Theory of Cognitive and Affective Development: Foundations of Constructivism,* Allyn & Bacon

[15] Utdanningsdirektoratet. (2013). *Læreplaner.* Retrieved on http://www.udir.no/Lareplaner/

[16] PERL. (2013). *Concepts and Values.* Retrieved on http://www.perlprojects.org/content/view/full/18699

[17] Belman, J., Flanagan, F. (2010). Designing Games to Foster Empathy. *Cognitive Technology, 14*, 5-15.

[18] Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

[19] Nordby, A. (2015) Teaching 'hardcore science' to arts and design students from the gaming generation - Reflection on how to use game development to teach programming, system thinking, mathematics and physics. Proceedings EDULEARN15, Barcelona.

[20] Nordby, A., Karlsen, S. (2014). Teaching 'hardcore science' to arts and design students: Reflections on the development of a basic programming course. *InFormation - Nordic Journal of Art and Research* 2014 ;Volum 3.(2) s. 129-142

[21] Rothstein, B. Social Trust and Honesty in Government: A Causal Mechanism Approach, in: Kornai, J., Rothstein, B., Rose-Ackeman, S., (Eds): *Creating Social Trust in Post-Socialist Transition*. New York: Macmillan, 2004

4

TEACHERS AS LEARNING GAME DESIGNERS:

CAN ELEMENTARY SCHOOL TEACHERS WITH NO BACKGROUND REALLY GAMIFY THEIR OWN TEACHING?

Anders Nordby[1], Hugo Nordseth[2]

[1]Hedmark University of Applied Sciences (NORWAY)

[2]Nord University (NORWAY)

Abstract

Yes, they can! This presentation gives some examples from a project where ordinary elementary school teacher's design and implement games for their own teaching. Their assignment was simple - develop and implement a pervasive game that can be played in the school and the local community during one school day. The game should be based on the Norwegian national curriculum goals, and should include digital clues, videos, animations and digital games, as well as a well-written story that connected the game to the curriculum and the learning goals. Many of the teachers made the digital games themselves in Kodu (Microsoft) or similar game apps for iOS or Android. Some teachers even made collaboration with other classes in schools thousands of kilometers away. We will show that what the children learn from the game is very cross-disciplinary and can be connected to other topics and learning goal than the main one, including history, language, geography, gymnastics, science (mathematics and physics), digital competence etc.

The project was a collaboration projects between two Norwegian teacher educations in Game Based Learning. It was implemented between 2012 and 2015, and the classes for the teachers were implemented several times during this period. In HUC the course included teacher students are in their 4th educational year as well as experienced teachers that had worked in elementary school for years. The games were implemented and tested in their classes or in their educational practice. The course required no previous knowledge in game design or digital skills.

Keywords: Gamification, Games and Learning, Pervasive games, Digital Games, Concurrent Design.

1 INTRODUCTION

In 2012 we implemented a project in Pervasive gaming for learning in a game production track at Hedmark University of Applied Sciences (HUAS). In this project the students developed a pervasive game for learning, based on competency aims in the Norwegian curriculum related to sustainability. The game goal was to show the children how they could support a long-term ecological balance, and to teach them about ecosystems, pollution, the social aspects of sustainability, and point out how sustainable systems could be used to live wisely. We wanted to apply system thinking, collaboration and problem-based learning (PBL), both in the game development process done by the students, and later when the children was playing the game.

The story of the pervasive game starts with a refugee that had to flee from his country because of famine and war, as consequences of pollution and climate change. The children find his video blog, and are given the task to find out what has happened to the environment in his country, and how the pollution and climate change could be prevented.

The students also developed three digital games as part of the overall pervasive games; one about the refugee situation, one about the ecological balance of an island and one about the soil ecology see from an earthworm perspective. The complete pervasive game played out within the children's school area.

To play the game, the children were divided into teams, and each team received a backpack with a pair of walkie-talkies, riddles and clues, a treasure map, a note with computer passwords and a notebook. They then had to solve clues to find the different locations in the school or in the schoolyard, where they could talk to actors about sustainability-related issues to be able to proceed in the game.

The was also a sum up after the game had been played; a 25-page diary was filled in by the children to sum up and organized their learning experience, and we did a system thinking session to discover how things they learned about in the game were connected. More details about this project can be found in [1].

To be able test the game we sent out a request to test the game to a list of schools in the local community. The response was big; within hours we had response from 15 schools that wanted to implement the project on more 1500 children. At the time we only had capacity to implement the project in one of the schools, but we had to promise to come back to the other schools with similar projects.

At the same time we were about to start a project in Game Based Learning together with another Norwegian teacher educations - Nord University (NU). Because of the large response to the game track project described above, we decided to also include pervasive games for learning in this project, and to test how capable the teacher students participating in the project was to implement such games for their own education. During this project period (2013-2015) we tested this approach several times. All tests were done by the teacher students, which included both inexperienced young students and older more experienced teacher that had been working in Norwegian schools for years and wanted to refresh their education. The games were implemented and tested in their classes or in their educational practice. The courses required no previous knowledge in game design or digital skills.

The project research goal was then to find out if the teachers were able to implement games for their teaching without a background in game design.

2 THEORY

Pervasive games are a broad category of games that is mainly played in the real world, but includes clues, hints, texts, pictures, videos or even small digital games in the digital world. The user must use or solve all of these digital hints/clues or games to solve the complete game. Two examples of pervasive games can be Alternate Reality Games where players cooperate to share and understand information presented on the Internet or other digital media, or Urban Adventure Games where the goal for example could be to experience a town in a new way.

Pervasive game development contains creative and immersive processes similar in many ways to game playing. It is about making games out of non-gaming environments, which these days often are labeled as Gamification. Gamification is a broad categorization that has many definitions, for example "using game-based mechanics, aesthetics and game thinking to engage people, motivate action, promote learning and solve problems" [2]. As such, game development shares the same basic learning theory fundament as Gamification does. Werback [3] points out how gamification is rooted in cognitive psychology and use points, badges and leaderboards to create intrinsic and extrinsic motivation. Gamification is more than this though; it can be seen as a community of practice where the students can work together towards a common goal in a problem based way, while learning events pops up naturally. At each learning event, players are motivated to solve the task because let them move further in the game, and rarely find it odd to study the theory to solve the task at hand. A community of practice is situated learning [4] [5], and the fact that the students "make a pervasive game" instead of "learning how to make a pervasive game" is experiential learning [6]. The students are "learning by doing" [7] while the learning events pops up naturally "just in time" [8].

We have labeled the creative process that is carried out *before* the actual pervasive game development process the 'pre production' in our documentation. In the pre production, the students work in groups to brainstorm their game ideas. Darsø [9] labels this process the 'preject' and describes how this process is different from the rest of the project in that it should be a process balancing between chaos and structure to make the process creative. She discusses how assignment of roles to participants in such processes (the innovation gardener, jester, conceptualizer and challenger) can help maintain the creative potential in the group, and that creative groups should be put together as heterogeneous as possible, and ideally contain 6-10 people.

3 SCOPE

The scope of this article is to give some examples on pervasive games created by teachers with no gaming background, and to show that teachers are able to make pervasive and digital games for their education.

4 METHOD

For this project we used the same principles as in the student project described in the Introduction. The assignment was simple; make a pervasive game for your own teaching that can be played by the children in one school day. Below we have given a list of the tasks in the game development process:

1. The teachers first examine the competence aims in the Norwegian curriculum, and find goals that are aimed at what they want to teach to the students.
2. Then they brainstorm to find some good ideas for a game. The brainstorming is a chaotic process where the just note the ideas that pop up. The ideas is then organized and developed before they agree on the one they wanted to proceed with.
3. At this point the students have to study some game design to be able to create a game from the game idea. This is done through teaching and self-studies.
4. They then make sketches of the game environments, defines characters and makes a storyboard.
5. Since they made a pervasive game they also needed to create digital clues, videos, and even digital games to support their game.
6. They also needed to find out what kind of actors they needed and what roles they needed to play.
7. Last but not least they had to make material to test the children's learning after the game was finished.

The process described above was usually carried using problem-based learning [10]; the students are divided into groups and solve each step by brainstorming, systemizing, studying and summing up [11]. The teachers will guide them through this process.

There was no prerequisite that the students should include digital games in their design, but many did anyway, but with simpler development tools than those used by game students. A popular choice of game engine to produce digital games was KODU [12] . Kodu is developed by Microsoft, and is an environment to make digital games visually. The teachers develop the game and the playground using a number of pre made assets and characters. They can then code their game through a visual programming tool. The teachers found this tool both easy and fun to learn and to use. Kodu contains many examples, which makes it easy for the teachers to understand how games are made.

Some of the pervasive games was made using a couple of new digital tools for making treasure hunt games. These tools are called Seppo and Turfhunt.

Seppo is a Finnish tool, which let you make games in a web page using a map to place the game tasks. Each tasks includes multiple-choice questions, and is creative in that the players can include text, picture, video or sound in their answers. The players (students/children) must also play the game in the web page. After login they are given the map the teachers created with the locations for the tasks, where they solve the challenges or questions using an iPad, a tablet or a smartphone. The teachers, which follows the player's progression, gives points for the answers on the fly. More details on this game development tool are found in [13].

Turfhunt is an Icelandic produced App where you make the games by downloading a map of the game area and the game locations. Together with the download map you also include the multiple-choice questions that the players should answer in each task/location. The game is played by downloading the App to a smartphone, iPad or tablet. When you log on to the game you get the map and tasks. Their location is always monitored by the GPS and shown on the player's map. When the game is started, the location of the first challenge pops up on their map, and they have to move close to it to get the first task. When the first task is solved they can move on to the next. More details on this game development tool is found in [14].

The teacher students in the project had a thorough background in pedagogy and teaching, but no knowledge about game development or pervasive games, so we had to teach them how to do this. This was either done before the development process started or during the process as a part of the PBL process.

To teach the teachers more about game design and implementation, some game projects added a second session that included game students; third year bachelor student from the games track was asked to improve on the teacher's games. In this session the teacher students worked as facilitators. This was done through a brainstorming session; the game students were organized into groups of six students, and the groups were put together as heterogeneous as possible to make them as creative as possible [9].

Research methods and data collection

In the projects the researchers and the students worked together to design the game, solve the challenges and implement the game ideas. This kind of research is classified as ethnographic design [15].

Some of the projects were done as case studies [16], and some was taken further into problem based learning [10][11]; the development process for the games was divided into several cases and then presented to the students as problems. Each case started with a discussion where the students clarified the problem and found out what to study to be able to solve the case. They then have an individual study period where they studied theory and solve their part of the case. Then the group meets again to sum up and to implement the results from the study period into the game.

The game development process was documented through notes, meeting resumes, pictures, reports, web pages and blogs written by the students during the project. In most cases the research process could be described as action research [17], where the sum up phase was used to improve the next case. Further, each project gave experience and influenced the design of the next project, which also is action research.

In some projects the students collected data and observations from the implementation, and then reflected on the learning outcome in their reports. In other projects, like the "The mystery of Agatha Hansen" (see below), data were collected during both the development phase and the playing both by the teacher students, the game students, and the researchers in form of notes and reports.

5 RESULTS AND EXAMPLES

Most of the games the teachers made were tested in school environments and proved to be very popular. Below are some examples:

Operation Overload - an expanded treasure hunt

This game was developed by two students, which after school was finished ended in different parts of Norway (1000 km apart). The game was implemented after they finished their education as a distant cooperation project between their schools. The game was developed as a pervasive game - a treasure hunt.

The game was played as follows - while the children is watching a boring school video, the video is suddenly interrupted by a message - OPERATION OVERLOAD - that tells they have to meet in the football field outside the school right away.

There the class is divided into 3 teams named BRAVO, CHARLIE, FOXTROT (the names are connected to the task they will solve) and each team is assigned a teacher as a guide. They first find a coffin that is locked with a padlock with a number code. It is possible to solve the number code by solving a case on a sheet of paper beside the coffin. When they unlock the coffin they find a Skype address and a set of coordinates. They have to go back to the base and search Internet to find where the coordinates are, and then log into Skype to find out who is in the other end. In the other end there is a class 1000 km away. They are solving another game, and the coordinates the children were given have to be used by the other class to find the next game challenge. In a similar way the other class has some coordinates for this class. Both classes then need to go to the place the coordinates indicate, solve a task, and then get new information they will have to discuss with the other class to find the next challenge. Later on in the game they get both QR codes and maps they will have to discuss with the other class to get further in the game. In the last coffin they get a paper with applications to fill in to get money to travel and see the other class. This is the reward for the game.

The Aftermath: After the game the teachers had made a Kahoot quiz [18] that summed up the learning in the game. This quiz will also function as a log for the children as well as sum up for the teachers to improve the game next time.

The game was based on competence aims in mathematics, natural science, gymnastics, Norwegian, English and Social science but they also learned other things like digital skills, cooperation etc. The theoretical learning was put in the tasks they had to solve and discuss with the other class.

Two pervasive history games based on Kodu.

Both games combine the digital game Kodu with a pervasive game.

The overall game goal of the first game is to find a word that will be a result from questions and puzzles from the game. In this game the children start with Kodu. The Kodu game has 3 levels and in each level the children has to solve several tasks to get a code that leads them to some place in the real world, where they have to solve the next task. After solving the real world task, they get a code for the next level i Kodu. All in all there are 7 levels in the game, 3 in Kodu, 3 in the real world, and one boss level at the end.

In the first Kodu level the children has to collect a number of things to get the code, in the second level they have to win a racing game to get the code, while the third level they need to win an air-hockey/pong kind of game to get the code. In the real world they have to solve tasks in gymnastics, orientation, coordination, and memory recognition. However, they also get questions in history and social science throughout the game and they all need to be researched to find the final puzzle word.

The game is founded on learning goals for Greek roman societies (history), social science and gymnastics, but also here they will learn cooperation, digital skills, competition etc.

The other game is somewhat similar, the children have to find knowledge capsules (they pop up as an instruction when bumped into) in an open Kodu world and then solve them. There are 6 such capsules in 3 topics; Norwegian, English and Mathematics. The overall story is that the Kodus (the avatar the children play in the KODU game) is from outer space and has landed on earth to collect information about the humans. The children will help the Kodus to collect information, and when done they get the key to a boos game level where they can find a spaceship to travel to the Kodus planet.

"The mystery of Agatha Hanssen" **- Urban Adventure Game**

The goals of this game is to introduce the players to Namsos town during the Second World War, to teach the players war history, and to let them feel how it was to live in a bombed and burned out city. The local museum served as the base for the game, and the museum curator introduced the game by presenting documents written by Agatha Hanssen. These documents revealed that she had secretly hidden something from the Germans in several locations in the town. These things had to be found for the players to solve the puzzle.

The players were divided into groups, and each group was given an Ipad and a web site address where they could report their findings.

The overall game goal was to explore and discover what have happened to Agatha Hanssen in those locations in Namsos by reading her letters, her diary and by using the city maps and hints from the game. The time frame for the game was one school day.

After the teachers had designed the game it was presented to game students that were asked to improve upon the game design. The students were introduced to the original game concept of Agatha Hanssen, and then given some time to brainstorm the project for new ideas. During that process the students took on different roles such as Game designers, Producer, Programmers, Visual artists, Quality Assurance, Work estimation/economy etc. All ideas from each group were recorded in a shared Google document. After the brainstorming session each group analyzed their document to find the best ideas that was possible to implement in time. A final session was then done with all the students and teachers to select the final ideas. The results are presented in the result section below.

We found that the game students were less afraid to use digital tools to vary the game play. This is possibly because the teachers are not "digital natives" and a bit scared of the digital domain. We think this will change in new generations of teachers. Form the sessions with the game designers the students suggested, among other things, that the game could use a digital game board or map on the iPad where they used GPS tracking to see other players, and that the players could collect points and puzzle pieces and thereby give a digital view of the others players level, or scores in the game. They also suggested that the teachers expanded letters into puzzle games and used QR codes to get data from the locations in the game. All in all the game students found more uses for Pad than the teachers did, maybe because of their digital background. In addition to this, some of the groups had ideas for how to improve the gameplay in general. However, the teacher's story was good and was not improved upon much.

Keiser Resir **- Alternative Reality Game**

The learning goal of this game is to teach the children about recycling of waste like paper, bottles, plastic and general waste. The game is made for children in K4-5. The introduction of the game is a video with Keiser Resir, who threatens the dean to take down the green environment flag in the school, followed by another video where the dean asks the children to help to save the flag by doing recycling tasks. The children are then divided into groups with 4-5 children i each group. They start by inspecting trash baskets, where they will find trash that doesn't belong in the basket. This has to be reported to the teacher and corrected. In the next tasks they are introduced to QR-codes at different places at the school, which links to digital games and videos with recycling issues. The children then have to play these games or watch these videos to get to the next task in the game. The last task is to make Christmas decorations using old light bulbs. When that is done the game is concluded with a short celebration; they have saved the green environment flag in their school!

The *Liberation of the language teacher* - A treasure hunt game

The purpose of the game is to work with different assignments in “Nynorsk” (a secondary variant of the Norwegian language) through digital tools, and to win the race to free their kidnapped teacher. The game starts when detective Kambestad enter the classroom and show the students a video with their kidnapped teacher. In this video their teacher is begging the students to solve several assignments from the kidnapper to free him. The video also introduces them to the first task.

Detective Kambestad then present the game, divides the students into groups, and gives the children the necessary digital equipment and tools, such as a flash drive, smart phones with Internet connection, PC’s and codelock. The groups have to write a short report in ‘nynorsk’ to detective Kambestad after they have finished a task and want to go on to the to the next one. Detective Kambestad also helps the children to find the clues, provided that all questions and dialog is done in ‘Nynorsk’.

The Briefcase Mystery and The Biscuit Thief - two treasure hunts games in Turfhunt

The purpose of the briefcase mystery is to introduce digital maps and work with mathematical challenges for children in K5. The class had 20 students that were divided into five groups. The game started by giving the group leaders an envelope with the assignment and the key to the premade Turfhunt game. The game has six different tasks, each with a multiple-choice questionnaire. A correct answer to this gives 10 points, while 5 points is given for each GPS-location they visit at the map. When all but one task is solved, the students have to use their previous answers to solve a code to open a briefcase given to them in the last task. The group score must be at least 70 points before they are given the briefcase task. Some of the groups might have to do extra tasks to reach this 70 points limit. This game is scheduled to take about two hours, and the content of the briefcase is case candy for each group that are able to find the code.

The biscuit thief game is a treasure hunt to learn about the local village. In the game the players use digital maps and groupwise problem solving. The target group is children in K5-K7. The class is divided into groups with 3-4 children. The game starts with a plenum introducing for all groups to a secret mission. The story is about an accounting company in the neighbor building, which is frequently visited by a biscuit thief. Unfortunately the surveillance camera is stolen, but it is possible to find some clues by playing the Turfhunt game. The young “detectives” finds messages at the local store, the newest house in the village and by visiting an old inhabitant. Some clues and hidden messages are also hidden in origami sculptures. The key issue is to identify a specific link to a YouTube video from the surveillance camera where the thief presents herself and solve the problem. The prize for the “detectives” is of course biscuits, and the moral of this story is to change habits from eating biscuits to eating fruit. The game plan is estimated to take a school day.

6. DISCUSSION

The examples above were a small selection from more than 40 games the teachers have produced over the two-year period. The examples clearly show that teachers are able to implement their own games for teaching.

Also, in all implementations the children/players had fun.

To be able to say something clearer about the learning outcome, the research material will have to be studied in more detail. This certainly will be done sometimes in the near future, and presented in new articles. However, some thoughts about the game projects are given below.

An important advantage of pervasive games is that we don’t need special equipment; the teachers can use what the school already own. The game can also be played inside the school or in the existing campus without many modifications.

Pervasive games are also relatively fast and easy to make, and the teachers can make the assignments, clues and applications themselves. They can use available digital films, letters, diaries, digital maps, videos, email, smartphones, tablets, laptops, Smartboards, QR-codes/scanner and GPS, walkie-talkies etc.

In both schools involved in the projects the teachers (and game students, when involved) also did the acting and prepared all the scenery and locations before the game started.

Most children also have smartphones these days. They are computers and can be used for digital game playing, goggling, visiting forums or groups etc. A good idea could be to group the children so that at least one smartphone is an asset of each group.

We can, based on the examples seen here highly recommend that teachers try out games for learning in their teaching.

REFERENCE LIST

[1] "The best school day ever" (Nordby et al, 2015). EDULEARN15, Barcelona 2015 Procedings.

[2] The Learning Circuits Blog. (2012). *What is gamification? And why it matters to L&D professionals.* Retrieved from http://learningcircuits.blogspot.no/2012/01/what-is-gamification-and-why-it-matters.html

[3] Gamification (2015) Cousera Course, Kewin Werback, 2015

[4] Lave, J., & Wenger, E. (1991). *Situated learning. Legitimate peripheral participation.* Cambridge: Cambridge University Press.

[5] Wenger, E. (1998). *Communities of practice: Learning, meaning, and identity.* Cambridge: Cambridge University Press.

[6] Kolb, D. A. (1983). *Experiential Learning: Experience as the Source of Learning and Development*. Prentice Hall

[7] Dewey, J. (1916). *Democracy and education: An introduction to the philosophy of education.* New York: Macmillan.

[8] Gee, J. P. (2007). *Good video games and good learning: Collected essays on video games, learning and literacy.* New York: Peter Lang International Academic Publishers.

[9] Darsø, L. (2001). *Innovation in the making.* Frederiksberg: Samfundslitteratur.

[10] Barrows, H. S., & Tamblyn, R. (1980). *Problem-based learning: An approach to medical education.* New York: Springer.

[11] Maastricht University. (2013). *Problem based learning preparatory website.* Retrieved from http://www.umpblprep.nl/

[12] Kodu game lab community (2016, mai 15). Retrived from: http://www.kodugamelab.com

[13] Seppo. Teach with a game. In a fun and easy way (2016, mai 15). Retrieved from: http://www.seppo.io/en/

[14] Turfhunt (2016, mai 15). Retrieved from: https://locatify.com/turfhunt/

[15] Askheim, O.G.A. and Grenness, T. (2008) *Kvalitative metoder for markedsføring og organisasjonsfag*. Universitetsforlaget, Oslo, Norway.

[16] Pettersen, R. C. (2005). *Kvalitetslæring i høgere utdanning: Innføring i problem- og praksisbasert didaktikk.* Oslo: Universitetsforlaget.

[17] Reason & Bradbury, *Handbook of Action Research*. London: Sage, 2001

[18] Kahoot, *Make Learning awesome.(2016, mai 15).* Retrived from: https://getkahoot.com

Inland Norway
University of
Applied Sciences

Many game designers and academics have pointed out that games are systems. So, what would be more natural than using System Science or System Thinking to understand and develop them? That is, in short, what this PhD thesis is about, and is stated in the main research question: *"How can System Thinking be used to understand, design, develop, and document the process of Serious Games development?"* The PhD thesis discusses Serious Game development specifically for learning and training. However, that does not mean the results are not usable for entertainment games, it is just not the focus of this thesis.

System Thinking aims to identify, explore, and understand complexity in systems, and uses System Analysis and System Dynamics to do so. This gives System Thinking a place in Serious Game design throughout the design, implementation, and runtime phase of the game. The research also discusses how psychology can be integrated in this design, and what pedagogics should be used to implement the System Thinking into the game design. The research is conducted in three courses given to students at the Game School in HINN, Norway. The courses are well-established and have been running since 2006. The courses are revised every year, and new areas of System Thinking are included. The PhD thesis includes eight articles about these projects. Four of these are articles published in journals and are to be considered as part of the PhD thesis. The other four are conference articles and are to be considered as background information. The PhD thesis not only discusses what has been done so far, but also suggests areas of potential that can be developed in the years to come.

Anders Nordby • System Thinking in Serious Games • 2023