Lappeenranta University of Technology
School of Industrial Engineering and Management
PERCCOM Master Program

Maria Victoria Palacin Silva

GREEN ASPECTS STUDY IN GAME DEVELOPMENT

Examiners: D.Sc. Jussi Kasurinen
Professor Jari Porras

Supervisors: D.Sc. Jussi Kasurinen

This thesis is prepared as part of an European Erasmus Mundus programme PERCCOM - Pervasive Computing & COMmunications for sustainable development.

Co-funded by the Erasmus+ Programme of the European Union

This thesis has been accepted by partner institutions of the consortium (cf. UDL-DAJ, n°1524, 2012 PERCCOM agreement).

Successful defense of this thesis is obligatory for graduation with the following national diplomas:

- Master in Master in Complex Systems Engineering (University of Lorraine)
- Master in Pervasive Computing and Computers for sustainable development (Lulea University of Technology)
- Master of Science in Technology (Lappeenranta University of Technology)

ABSTRACT

Lappeenranta University of Technology
School of Industrial Engineering and Management
PERCCOM Master Program

Maria Victoria Palacin Silva

Green Aspects Study in Game Development

Master's Thesis

79 Pages, 17 Figures, 14 Tables, 1 Appendix

Examiners: D.Sc. Jussi Kasurinen
Professor Jari Porras

Keywords: software industry, game development, software engineering, sustainability migration processes.

Context: Game development has become increasingly important in the software industry, but this importance has not affected the way software engineering approaches and methodologies manage the differences they have with game development. Similarly, software engineering does not fully support sustainability practices, causing this element to often not be considered or even known as a requirement for a development lifecycle. **Goal:** The aim of this thesis is to study the mode in which games are developed, and the involved sustainable aspects and the relevant concerns regarding the migration processes. **Method**: A quantitative study was conducted, gathering 33 answers of game professionals from four continents, from administrative (25%) and technical oriented positions (75%). **Results**: Three trends were observed: 1) Agile process models are used, 2) major concerns for mobile development and digital marketing, 3) minor concerns for eco-impact elements and certain development phases such as testing and crunch time development. **Conclusion**: Traditional Software engineering would require a major change on its processes and models to fit with modern agile development, game development approaches and sustainable requirements.

ACKNOWLEDGEMENTS

There were times I really doubted I would ever finish this document. I lost and found my motivation many times... Finally, seems my work is done, and I would like to express my special gratitude to people…special people that were key during this process:

Special thanks to the PERCCOM Selection committee, and to the fine host institutions: University of Lorraine, Lappeenranta University of Technology, ITMO University and Luleå University of Technology, for this unforgettable and fruitful two years of studies.

All my gratitude to D.Sc. Jussi Kasurinen for all the time, the excellent guidance and the great attitude, during this research process.

Thanks to Lappeenranta University of Technology, Prof. Jari Porras, Susanna Koponen and Suvi Tiainen for hosting this Erasmus mundus master with such professionalism and efficacy.

Gracias mamá, papá y Piero por todo el amor, apoyo y gran animo que siempre me brindan, rompiendo todas las barreras de la distancia. Son lo mejor que tengo en mi vida.

Gracias Ville Myllynpää, por toda la buena vibra y ánimos durante este proceso.

Great thanks to my proof readers: Morgan Oats, Maike Schmidt, Sumeet Thombre, Peter Miklos and Anna Fuksa. Without you this document might not be fully readable

This two years were a wonderful journey which I had the pleasure to share with incredible people from all over the world. Thank you so much for everything: Dimocro (Vlad and Maike), Dorine, Mohaimen, Zainie, Ramya, Vitalii, Khoi, Alex, Fisayo, Fia, Rohan, Iqbal, Stefanos, Princess and Chandra.

TABLE OF CONTENTS

LIST OF SYMBOLS AND ABBREVIATIONS

AI	Artificial Intelligence
BFR	Brominated flame retardants
CEO	Chief Executive Office
CMMI	Capability Maturity Model Integration
ESA	Entertainment Software Association
EU	European Union
LD	Level Design
LUT	Lappeenranta University of Technology
MMO	Massive Multiplayer Online
MMOGs	Massively Multiplayer Online Games
PERCCOM	Pervasive Computing and Communications for Sustainable Development4
MIT	Massachusetts Institute of Technology
PVC	Polyvinyl chloride
QA	Quality Assurance
RoHs	Restriction of the use of certain hazardous substances in electrical and electronic equipment
RQ	Research Question
SE	Software Engineering
SOCES	Software Development in Creative Ecosystems
SWEBOK	Software Engineering Body of Knowledge
UAT	User Acceptance Testing
US	United States
US$	United States dollar
XP	Extreme Programing
UAT	User Acceptance Testing
LD	Level Design
AI	Artificial Intelligence
RE	Requirements Engineering
QA	Quality Assurance

1 INTRODUCTION

The video game industry has grown progressively since 1970, becoming one of the fastest growing sectors worldwide, a pioneer technology, and a significant part of the modern software development, with revenues three times higher than software retail in 2012 (Nayak 2013). This industry is characterized by a high degree of innovation and dynamics, turning into not only a simple way of entertainment and social interaction for all ages and genders, but also a medium to train students, soldiers and medical professionals (Murphy-Hill, Zimmermann and Nagappan 2014). Overall numbers (Entertainment Software Association, 2014) reveal that: 52% of game players are male and 48% women, who are on average, 31 years old. In addition up to 62% of game players play frequently with other people (either in person or online).

However, due to the agile and creative nature of video game development, their practices and methods are highly iterative and do not strictly meet with the traditional software engineering (SE) standards and practices (Murphy-Hill, Zimmermann and Nagappan 2014). Nevertheless, the differences between SE and games development are not exclusive; it seems that traditional SE does not fully support other fields such as sustainability (Penzenstadler 2013).

The goal of this thesis is to study the mode in which games are developed, the involved sustainable aspects and relevant concerns regarding the migration processes. A quantitative study was conducted, gathering 33 answers of game professionals from Canada, USA, Finland, Sweden, Australia, Russia, Germany, Ecuador, Spain, and France among other countries, from four continents. From our respondents: 36% were developers, testers or other technically-oriented employees, 14% were artists, musicians, graphics designers or from other artistically-oriented positions, 25% were project managers, lead designers or part of other project-level management position, 18% were from upper management levels and finally 7% were belonging to any other marketing, administration or related position. The applied research methods involved: 1) (Kitchenham, et al., 2002) frequentist approach, 2) descriptive statistics detailed by (Fink, 2013) and Kendall's tau statistics, to analyze the data.

This study is part of the Pervasive Computing and Communications for Sustainable Development (PERCCOM) program study topics. The objectives of this joint master degree are: (1) Address educational challenges and attract students according to the new market demand expressed by OECD and European Commission reports. (2) Synergize the strengths, competence and diverse aspects of education in sustainable networks, software and services, pervasive computing systems and communications, and develop a common platform of competence within the guidelines of the Bologna process. (3) Provide future Masters Students with competences, skills, and knowledge in computer communications, wireless networking, mobile technologies, SE, pervasive and distributed systems, and to make them aware of the impact that ICT makes on the environment and efficient use of resources. (4) Educate students in the direction of the "green digital charter" committing the European Cities to reduce emissions through Information and Communications Technologies. (5) Propose the new International Master degree with no currently available match at the international level filling the gap between ICT skills and environmental considerations.

This thesis report is structured as following:

- Chapter two contains general insights about video games industry.
- Chapter three explores definitions, gaps and limitations between SE, game development processes and sustainability.
- Chapter four covers the research questions, methods, and results of this study.
- Chapter five consists of the discussion of several authors and key findings.
- Chapter six includes the conclusions of this study.

2 VIDEO GAME INDUSTRY

This chapter presents an overview of the video games industry, describing its history, business insights, contemporary business models, and generic technical information about platforms, genres, and trends.

2.1 History

"Though the history of video games is a subsection of the history of computing, it is important to recognize how integrated this entertainment medium is to the evolution of computing hardware and software." (Ted 2014)

The history of games began in January 1947 with the first documented game, by the U.S. Patent #2 455 992 by (Goldsmith, Grove and Ray Mann 1948). This was a game inspired by radar displays from World War I and designed to be played with a cathode ray tube. During the 1950-1970 decades, games had some extra features arriving to the market such as: displays, multiple players, home consoles, commercial games, and university innovations like Checkers and Tic-tac-toe by the Massachusetts Institute of Technology (MIT) (Winter 1996). Through the course of the 1970s, a second generation of consoles with a variety of arcade games was born, which is the reason this decade is now known(became known) as the golden age of arcade games. Consequently, in the course of the 1980s, new games genres became popular, such as: action adventure, role-playing, fighting, and racing, among others; additionally, important hardware evolutions emerged, which caused as a result the arrival of the third generation of gaming consoles by Nintendo.

The 1990s was the decade of innovation and maturity in the video game industry, with important architectural hardware evolutions such as: 32-bit, 64-bit and 128-bit new processor capacities, which were integrated then in the fourth and fifth console generations, and caused a rise of 3D graphics and further CDs to arrive as a greater storage medium opportunity for software and games (Ted 2014). Also, the mobile gaming sector emerged with Nokia installing the Snake game onto its phones, causing this practice to become a trend for all mobile manufacturers around the globe (Nokia 2009).

After the year 2000, a rise in online games led Massive Multiplayer Online Games (MMO's) to become the leading force in PC gaming, while in the game-console work the emphasis was on hardware: add-on devices and motion control gadgets, such as the Wii remote or Xbox-Kinect, and their inventions were a dominant trend. Likewise, during this decade the sixth, seventh, and eighth generation of consoles emerged. Finally, a new gaming genre appeared: the casual social gaming with: Wii Sports, The Sims, and Farmville applications, which became very popular around the world (Berg, 2010).

Cloud computing in 2010 met with games, with the apparition of a few services and projects offering cloud computational power to render the video games in order to reduce the load for the end user, increasing the games' performance. (PricewaterhouseCoopers 2012)

2.2 Business Insights

The video game industry is considered one of the fastest-growing components of the international media sector (Bilton 2011), and it has established itself as an important contributor to the global entertainment economy (Marchand and Hennig-Thurau 2013) and a significant part of the modern software development, with revenues three times higher than software retail in 2012 (Nayak 2013). This sub section describes the games industry's business relevance through its revenues, consumers, and contemporary business models.

2.2.1 Revenues

The rise of video game popularity caused an evident continuous rise in its revenues since 1970 period, *Table 1* reflects its actual global revenues and a short term forecast by 2016, represented in United States dollar (US$). However, (Newzoo, 2013). However, it outlines that only 15% of the global population generates 74% of the worldwide game revenues.

Table 1: Video Game Industry Worldwide Revenue Comparison (PricewaterhouseCoopers 2012) ; (Wikia 2014)

Year	1970	1995	2011	2016
Revenue (US$)	40 Million	29.32 Billion	58.7 Billion	83.0 Billion

2.2.2 Consumers

Video games attract a wide range of consumers across the world: the average contemporary game player is a 31 years old person, with a gender distribution of 52% males and 48% females. Further, up to 62% gamers play frequently with other people (either in person or online), and this has resulted in an increase in casual and social gaming on wireless devices (such as smartphones) and online environments by 55% from 2012 - 2013. (Entertainment Software Association 2014).

Regarding the consumers' preferences, up to 47% of the gamers prefer social games and about 53% of gamers in US acquire their games in digital format. In addition, three years experienced frequent gamers tend to reduce their time on activities such as watching TV (-48%), going to movies (-47%), and watching movies at home (-47%) in order to spend more time gaming (Entertainment Software Association 2014). These are causing a rise in video game digital advertisments (PricewaterhouseCoopers 2012).

2.2.3 Business Models

Following the major business models of video games among companies (Lee, 2013):

1. **Packaged Game Software Sales:** It is the oldest and most used business strategy for selling games, and consists of a software package which has game content (that customers can play anywhere, often from 10 to 30 hours). The way this model works is simple and efficient: players purchase the initial license for games which have a diverse medium matched devices such as Game Consoles, PCs or Smartphones and

then get the service forever. Also this model is easy to combine with other business model approaches, such as micro transactions inside games.

However, comparing the yearly reports from Entertainment Software Association, a decline in sales of this type of games can be spotted since 2008 period - this might be due to the multiple modern channels available to buy games, and the rise of free-to-play models. Yet this business model still leads on the game industry.

2. **Subscription:** It is not a new model among companies, but it is very popular among developers because it generates constant revenue and engage players to be disposed to pay. In this model gamers pay a monthly service fee, which represents a big opportunity for continuous profit, when the game has large audience. An exceptional example of success with this business model goes to "World of Warcraft", which recorded the largest active subscribers in game history (12 million), charging to each of them 14USD monthly.

   Nevertheless, developing a game with such a model requires a big investment on the game development (which provides real time support for many players) and added investment for maintenance (servers, help desks, contents updating). Therefore, the life span of these kind of games is expected to be longer than the packaged ones. Also, the subscription games include other small business model approaches such as micro transactions and free trials of features.

   Free-To-Play: This model is the result of intense competition between companies to attract players. The target market is mostly casual players on social networks and mobile apps (casual gaming is reinforced by this model). The revenue in this model comes from the players buying in-game services with real money, while other sources of profit are ads, freemium features, and virtual goods. Further, in a free-game model many players are not always willing to pay, but they still play an important role due to the most defining element of this model being the large number of players in which to interact with, therefore, even when the total number of players are not generating direct profits their existence is still beneficial.

2.3 Value Chain

The game development industry, as other fields, has interconnected layers between its elements to make game existence possible. This chapter explores the traditional and online value chain for games and its components.

2.3.1 Traditional Value Chain

Traditionally, the game industry value chain (Figure 1) has five main components: 1) Developers whom represent the talent layer designing and developing games. 2) Publishers that are responsible for licensing the rights and the concept on which the game is to be based. 3) The distributor who is in charge of marketing the game, handling packaging and transport, and in some cases, providing user support. 4) Retailers that commercialize games, such as counter trading, net trading (via downloads or post mail) and online gaming (example browser-based games). 5) End Users/Consumers or Customers whom buy and play the games based on their given options such as hardware available, game products preferences by genres and interfaces, and the online/offline availabilities (Norway Ministry of Culture 2008).

Figure 1: Traditional Value Chain (European Games Developer Federation, 2010)

2.3.2 Online Value Chain

When digital distribution and online elements are highly involved with games the traditional value chain (Figure 1) is reduced to three components: 1) Developers 2) Distributors, which are online stores such as Google Apps, Apple Store or Windows Phone store that allow developers to upload their games, and 3) Consumers. This value chain (Figure 2) has been forecasted to become the largest category by 2016 due to its fast-growing tendency.

Figure 2: Online Value Chain (European Games Developer Federation, 2010)

2.4 Operation and trends

The game industry is highly dynamic and innovative, so that in many cases hardware and software advancements are applied to games before other domains - this has caused games to cover a wide variety of genres and are supported by multiple platforms (Ampatzoglou & Stamelos, 2010). This section will explore those game components and forecasted industry trends.

2.4.1 Platforms

In technology, the term platform refers to a specific combination between certain computer hardware and software which allow software systems to operate (FOLDOC, 1992). In the contemporary game industry there are five main platforms (Edge Staff, 2007) with different levels of popularity:

1. **Arcade**: Includes a playing surface that can be manipulated by the player: this kind of platform was very popular, from 1970 to 1990. Popular examples of this type of game's platform are: Dance Revolution, Pac-Man and Time Crisis. Though Arcade games are not the most popular gaming platform anymore, they are still generating revenues worldwide, especially in Japanese and Chinese industries where arcade platforms are still widely spread among cities (Edge Staff, 2007).

2. **Console:** Term generally referred for a video game console. This platform consists of functions on a computational processor with powerful graphical features attached with joysticks or other controllers, and aimed to display and play games (FOLDOC, 2014). It represents one of the most popular platforms for gaming. Widespread examples are PlayStation, Xbox and Wii families of game consoles.

3. **Mobile:** Consists of mobile devices such as smartphones, tablets or other wireless gadgets with a thin processor able to run a game: they range from a basic phone such as old Nokia phones running the snake game, to very modern devices which can support online games with social features. Currently this platform is widely popular and the most rapidly increasing area of business.

4. **Online**: Involves the use of Internet to play a game and the generation of player-to-player interactions. This platform promotes cross-platform interactions through different browsers, mobile devices, PCs, and consoles (Järvinen, 2008). As an advantage, the games deployed on this platform can reach large audiences and open opportunities for digital distribution of contents (PricewaterhouseCoopers, 2012).

5. **PC Game:** Implicates the use of a general purpose computer to play a game either installed or in online mode.

PricewaterCoopers forecasted that global console, online, and mobile games will continue expanding at 2.1%, 13.3%, and 10.1% annual rates until 2016. On the other hand, this report highlights a 1.9% anual decrease on sales for PC game platforms and estimates that online and wireless games will replace console games as the largest gaming category by 2016. (PricewaterhouseCoopers, 2012)

2.4.2 Genres

Among many classification dimensions, grouping games by genre is one of the ways to categorize them. This classification takes into account common gaming features such as style or set of characteristics. The ESA reports define a list of super-genres for games based on data from the NPD consumer research firm. This list includes:

1. Action games.
2. Shooter games.
3. Sport Games games.
4. Role-Playing games.
5. Adventure games

6. Racing games.
7. Strategy games.
8. Family Entertenaiment games.
9. Casual games.
10. Arcade games.
11. Fighting games.
12. Flight games
13. Other game compilations.

However, game genre defintions are still under a theoretical debate due to the vast amount of approaches for classifing them concerning their features and its compliance with the genre theory (Clearwater, 2001).

2.4.3 Trends

Five key trends for the video game market from 2012 to 2016 are reported by (Newzoo, 2013): 1) an increase in gamers acquiring more screens to play – the number of gamers playing with two screens has doubled since 2007. 2) A tendency to try games before buying, as in the free-to-play model. 3) New business models which balance the value for consumers and profits for the developers/publishers respecting the free gaming environment. 4) developers/publishers aiming to engage gamers for as long as possible, providing games as a service. 5) A global market place inclusion since online connectivity becomes the game market into a global playground; emerging markets should be a part of any game company's strategy.

3 SOFTWARE ENGINEERING, GAME DEVELOPMENT, AND SUSTAINABILITY. IS THERE SOMETHING IN COMMON?

Chapter three focuses on provide an overview of Software Engineering (SE), games development processes, and sustainable software engineering. This chapter also intends to describe the essential differences and gaps between traditional software engineering and other related areas such as game development, and sustainability.

3.1 Software Engineering

SE is the application of engineering to software using a systematic approach to the development, operation, maintenance, and re-engineering of software products (ISO/IEC and IEEE Computer Society 2014). A systematic approach of SE for software development is fulfilled by different specialized methodologies which are used to structure, plan, and control software development processes, following a specific life cycle with clear phases, iterations, outputs and responsibilities (ACM 2006). Due to the existence of several specific software development methodologies, organizations and industries must analyze which approach or framework fits the best to their ultimate goal and development culture.

3.1.1 Development Phases

Despite the differences among software development methodologies and lifecycles, there are four main phases of software development which intend to support the software development activities through its whole lifespan: 1) analysis, 2) design, 3) implementation and 4) testing. Each phase is strongly dependenct upon the others (Burback, 1998).

The analysis phase is the "what" phase, and its focus is the system's requirements definition, ignoring how these requirements will be accomplished (Burback, 1998). This phase ensures business consistency and accuracy through two informational components: Information Gathering and Requirements Analysis. However, this phase has a high dependence on the methodology and lifecycle chosen (Langer, 2008). Also, risks and strategic offers for risk mitigation should be identified (Azarian, 2013) during this period.

Through the **design phase** the requirements will be broken down and studied in detail in order to be able to forecast the project's timeline and estimate the level of effort and amount of resources needed. (Azarian, 2013). This phase is the "how"' stage (Burback, 1998) and has, as a result, the software architecture specifications (what programming language, database vendor to use, how to report results, what network communications technologies or topologies should be implemented). Design is perhaps the most iterative activity in software development which often iterates with analysis where questions and suggestions from designers can raise issues about alternatives not considered during the analysis stage. (Langer, 2008).

The **implementation phase** is focused on the building of components either from scratch or composition (Burback, 1998), through tasks which are broken down into release efforts so the application can be completed in separated parts and the client can preview what has been done during the process (Azarian 2013). All the necessary steps to accomplish the creation of the application are done during this stage (Langer, 2008). However, the implementation phase deals with major issues of quality, performance, baselines, and debugging. The end deliverable at this stage is the product itself. (Burback, 1998)

Quality is a distinguishing attribute of a system indicating the degree of excellence (Burback, 1998). The intersection between development and quality (Langer, 2008) lays on the t**esting phase**, which consists of testing all the functionalities of the application (Azarian, 2013). Testing is performed iteratively as issues are found, corrected, and retested. The last and critical testing activity is User Acceptance Testing (UAT), which is performed by the client. (Azarian, 2013)

3.1.2 Standards

SE has general, internationally accepted practices, which set a baseline for all industries that want to focus their efforts on software development in order to ensure quality, efficiency, and requirements compliance. The following list includes some of the most common and widely applied international SE standards, process models and certifications:

Table 2: Some of the most common SE-related standards

Title	Type	Description
Software Engineering Body of Knowledge (SWEBOK) ISO/IEC TR 19759:2005	Guide and Standard	Specifies the required body of knowledge and recommend practices for SE. (IEEE, 2014).
Capability Maturity Model Integration (CMMI)	Best Practices Model	*"Is a process improvement model that can be adapted to solve any performance issue at any level of an organization"* (Carnegie Mellon University, 2014). In addition, it is based on the best practice cases of the industries and has independent assessments to grade process definition compliances. It does not guarantee the quality of the end result.
ISO 9000	Standard Model	Sets out the criteria for a quality management system for manufacturing and service industries. (ISO, 2008). Focuses on the formality of processes, methods, and monitoring processes, therefore does not guarantee the quality of the end result, but certifies the formal order of processes in an organization.
ISO/IEC 15504	Standard Model	Reference model, and for the maturity models in software development, relating all the business management practices in an organization. (ISO , 2004). Dedicated to setting clear processes to manage, control, and monitor software development, then compare them with the organization's reality, and as a result identifies the weaknesses, strengths and opportunities for a software development organization.

3.2 Video Game Development

Video game development is a progressively influential part of software development. In this context each game, as each software product, must be managed, analyzed, designed, programmed, tested, and delivered. However, the roles in game development teams are much more different than in traditional software development teams, requiring more artistic and creative members which might perform two or more roles at the same time (Bates, 2004). The contemporary game development process is much more complex than decades ago - this has been generated by the change in team size and the increasing load on coding. Despite these characteristics, game development still requires a high number of iterations with a short analysis phase, and long design/creative phases (Redavid & Adil , 2011). This section focuses on how video games are developed and which elements are involved in this process.

3.2.1 Traditional Game Development team

Assigning job titles and tasks to each person in a team when developing games might vary constantly due to creative and dynamic reasons. Still, each game 1) must be managed, analyzed, designed, programmed, tested, and delivered; 2) need code, art, sound, and music, and 3) should be tested. These tasks are usually performed in a practical way, and the same person can take different roles or simultaneous ones (Bates, 2004). Following are the required teams which often participate in a traditional game development process:

Design team: Group consisting of game designer, lead designer, level designer, writer, or script writer (Sicart, 2007). This team is responsible for launching the game's original blueprint through the creation of design documents which include details about the gameplay mechanisms, game's movie, dialogues, and level designs (LD). This team also performs supportive activities for achieving marketing and sales goals, building the official game website, and creating assets and resources such as demos (Bates, 2004). Nonetheless, everyone on the project can have an effect on the design before it's completed.

Programming team: Group often formed by: lead programmer engineer or tools programmer, graphics programmer, artificial intelligence (AI) programmer, and multiplayer networking programmer (Sicart, 2007). This team addresses all technical aspects involved in the game development such as the selection of the architectures, delivery platforms, special features, technical implications, and most importantly to accomplish the imagined design through coding (Bates, 2004).

Visual arts team: Often formed by 3D model builder, 2D concept artist, 3D cut scene artist, 3D character builder or animator, level builder, art director, and art technician (Sicart, 2007). This team has the responsibility to create all visual art assets, which are the main characteristics judged in a game. The members of this team have high impact in phases like game design, and face complex issues when it comes to selecting tools that fit well with the creative needs for animations and special effects (Bates, 2004).

Audio team: This team according to (Sicart, 2007) can be composed of a sound engineer, a composer, and/or an audio engineer: they hold the responsibility to perform the art-sound and sound effects involved in the desired game.

Testing team: Testing plays a vital role in the game development in order to ensure the quality of the final product. This responsibility belongs to a specific team which members are usually a test lead and testers. Their main goals are to ensure that the game works, is fun/user friendly, and that it makes sense. In order to achieve these goals, this team performs the following main activities: 1) elaborate a test plan, 2) provide rapid feedbacks to the programmers, and finally, 3) identify incidents and risks (Bates, 2004).

Production team: Crew consisting of a producer, project manager, lead tester, game tester, and quality assurance responsible (Sicart, 2007). The main goals of this team are: 1) sell the game, 2) align the development to the company's goals, 3) track the status of progress and 4) manage the risks involved in the project in order to assure the quality of the product result. However, (Bates, 2004) points out that this activity can be performed by responsible personnel internally or externally depending on the company's choice.

3.2.2 Development phases

Despite game development being considered as part of software development with special creativity and media requirements, formal software engineering methods are poorly used for games development because they are not fully suitable for these types of projects (Bates, 2004). Still, it is a software engineering intersected field (Ampatzoglou and Stamelos 2010), sharing common problems and challenges (Petrillo et al. 2008; Petrillo et al. 2009; Petrillo and Pimienta 2010) which makes SE methods a potential medium to learn for game development and to deal with its issues (Redavid & Adil , 2011). Following are the nine phases of game development lifecycle described by (Schultz, et al., 2005):

Concept Development: This phase marks the beginning of a games development from the moment the idea appears to the moment the preproduction starts: usually the team that work in this stage is small and has part time team members. (Redavid & Adil , 2011). The main goal of this phase is define what the game is about, and the principal outputs are: a high game concept, a pitch document, and a concept document. In addition, during this phase the major gameplay elements and art concepts such as the game genre, features, story, and appearance are defined (Bates, 2004).

Preproduction: The goal of this phase is to perform the game design, set the production path and project plan, and release an internal prototype. This phase ends with the delivery of a game prototype which is a piece of software that shows how fun and functional the game is. Also during this period, the software engineer or project leader tries to identify, address, and reduce/eliminate problems in the software development effort before they cause costly problems. (Redavid & Adil , 2011). However, this phase is usually not funded by any producer for independent games (Bates, 2004).

Development: This phase is also known as the production stage, and it is the main phase of game development. Usually all the programming activities are performed here. Also, the art-related teams release their respective assets, sounds, stories. The game development phase is likely to last from six months to two years (Bates, 2004).

Some practices from SE can improve the efficiency during this stage, such as: nominate functional leads for sub-systems, and use basic UML schemas, such as using cases to develop a static design of the game. Although, the same SE practices might be dangerous when applied in very small projects because of their time limitations (Redavid & Adil , 2011).

(Redavid and Adil 2011; Kasurinen 2012) Highlight the importance of the testing process in game development. The testing process for video games is a "*black box*" of processes, test cases, and time limitations, which make it different from the traditional testing for software. However, as the testing process across the whole development becomes better controlled and measured, the product quality increases significantly along with the efficiency of the involved teams. During the following 3 phases quality is the goal, thus tracking all the defects and bugs is a concurrent activity which is improved when a clear quality assurance plan is presented in advance, including estimations from each team about expected bugs and critical sectors, hence, the quality team can work towards a measurable quality objective in order to get the game released (Redavid & Adil , 2011).

Alpha: By this stage, all the major components of the game have to be completed and it should be possible to play the game almost completely. The focus of this phase is the rapid feedback to programmers from fast testing and bugs fixing (Bates, 2004), although the definition of alpha might vary from company to company (Schultz, et al., 2005).

Beta: This stage marks the end of the development work and implies that all the outputs are merged completely, although, some bug-fixing activities are still performed during this stage (Schultz, et al., 2005). The goal of this phase is to deliver a stable and fun game, still, testing remains a focus, and the crunch time (last period of time before the publishing of the game) is performed during this phase (Bates, 2004).

Code Freeze: This phase involves "freezing the code", thus no changes are permitted to the code after this point. All the work is released to a master disk, which is then used for additional testing (Schultz, et al., 2005). The test of this phase is measured comparing the

accomplished quality with the quality assurance plan set at the beginning of the development project (Redavid & Adil , 2011).

Release to Manufacture: If the development of the game has been done correctly, once it is released, profits will being appearing, meeting the expectations of the game developer company. This is not strictly a development phase, but it is crucial because it implicates the business part of a game. Thus, setting clear financial expectations can help to have a clear measurement at this stage (Redavid & Adil , 2011).

Patch: Maintaining the video game is the key activity during this phase (Redavid & Adil , 2011): a common practice among gaming companies is to release patches for their games once they are in use. This is not strictly related to errors during the development process, but also to hardware combinations that players might have. Nowadays a patch usually contains readjustments to specific issues and content updates for the game such as maps and levels (Schultz, et al., 2005).

Upgrade: This phase involves the creation of additional content aimed to improve the original game, generate more profit, and provides further engagement for the gamers (Schultz, et al., 2005).

3.2.3 Game development methods

"A development method is a systematized procedure to achieve the goal of producing a working product within budget and schedule" (Sicart, 2007)

Waterfall method: Is a formal and class method for software development. In the variant for games development once the design document is done, an activity of "*waterfalling*" is performed. This task implies the division of functionalities and assets, and then assigns them to respective teams. This method requires a significant amount of time dedicated to front-end activities and functionality definitions, therefore, it brings a late implementation of mechanisms and levels. (Sicart, 2007). The main issue with this method is its difficulty to reverse (Flood, 2003).

Agile methods: Highly iterative methods, which are not documentation-centric, where the production is divided into small cycles, focuses on the most crucial features: at the beginning of each cycle the whole team meets and sets clear objectives. At the end of each cycle, meetings with the client are performed in order to showcase the product. (Sicart, 2007). These methods support team dynamics and different team cycles through daily meetings. Scrum, rapid prototype modeling, and extreme programming (XP) are the most followed methodologies in the games industry (Godoy & Barbosa, 2010; Bates, 2004).

Unified Development Process: Traditional SE method, focused on the requirements analysis in order to convert those requirements into functional software components. It requires effort on: document as use cases, game concepts, and assets definition (Sicart, 2007).

3.3 Sustainability

Sustainable Development = "Meet the needs of the present without compromising the ability of future generations to satisfy their own needs" (United Nations World Commission on Environment and Development 1987)

The (United Nations, 2005) set sustainable development goals for the following decades based on three pillars of sustainability: 1) economic development, 2) social development and 3) environmental protection. Those pillars are not exclusive and can be mutually reinforcing (Figure 2). These basic elements in the definition of sustainability have served as basis of several standards and certifications systems for various industries such as food production (Manning, et al., 2012). Sustainability implies balancing local and global efforts in a responsible, proactive decision-making, and innovative process that would reduce negative impact and preserve the balance between ecological resilience, economic prosperity, political justice, and cultural diversity to ensure a desirable planet for all species now and in the future (Magee, et al., 2013).

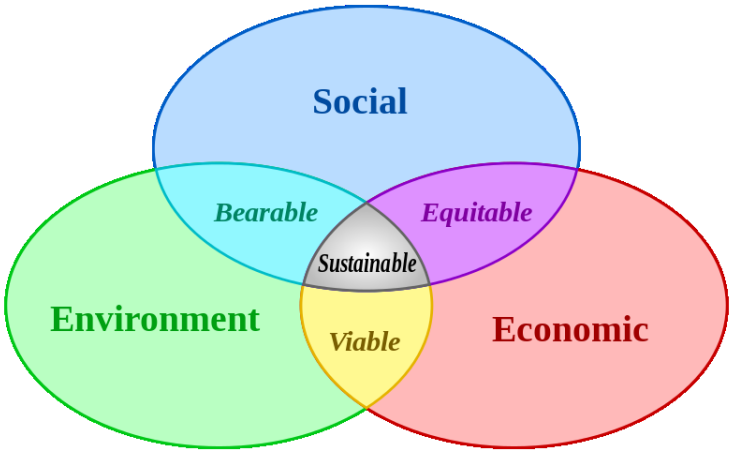

Figure 2: Sustainability pillars and intersections (Adams, 2006).

(Dahl, 1995) points out that sustainability is deeply related to societies, economies, and the world itself. Thus it is dynamic and requires simple defintions of dimensions and indicators despising the complexity and uncertainty, in order to be understandable and matter to all societies.

3.3.1 ICT and Sustainability

Today's world has a new important agenda: tackling environmental issues and adopting environmentally sound practices in all industries. In this context, ICT might be the biggest opportunity the world has to drive efficiency across the economy and deliver emission savings (The Climate Group, 2008). The ICT growing rate is incredibly high and that means its emissions (energy consumption) and effects (electronics manufacture) are increasing rapidly. However, the ICT industry can produce more benefits from its own growth in emissions by enabling other industries to reduce their emissions (Fujitsu, 2012). The Smart 2020 report, claims that ICT could reduce approximately 15% of the emissions in 2020, which can be translated into approximately 600 billion Euros of savings.

ICT is an inseparable part of modern business and societies. This implies, a greater ICT carbon based Generation, as well. Furthermore, ICT usage has different levels of effects (Figure 3) on the environment, societies, and businesses which can lead up to the sustainable triangle (Unhelkar, 2011; Jain, 2011; Erdmann, et al., 2004; Plepys, 2002). However, whether ICT will help us to move towards a more efficient and sustainable society, or to increase resource consumption and emissions, is still only little understood and very complex

(Zapico, 2012). The following sections detail the two approaches of using ICT to achieve sustainability.

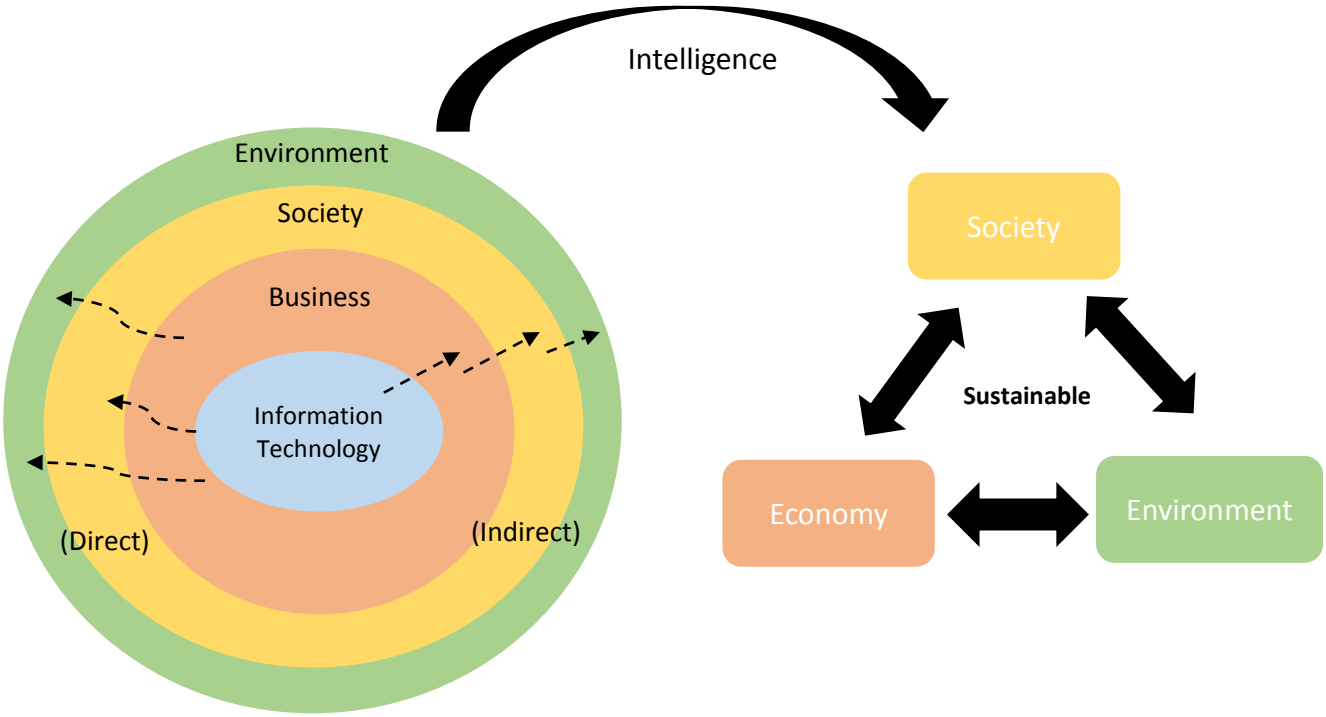

Figure 3: ICT influence in Business, Society and Enviroment leading up to the sustainability triangle (Unhelkar, 2011).

3.3.1.1 Green IT

*"Green IT refers to environmentally sound IT" (*Murugesan, 2008*)*

Green IT or Green computing refers to the study and practice of enviromental practices to reduce power and enviromental waste during the designing, manufacturing, using, and disposing of computers, servers, and associated subsystems (Murugesan, 2008; Hewlett-Packard, 2015).

Green ICT focus is on the following areas (Murugesan, 2008):

- Design for environmental sustainability.
- Energy-efficient computing.
- Power management.
- Data center design, layout, and location.

- Server virtualization.
- Responsible disposal and recycling.
- Regulatory compliance.
- Green metrics, assessment tools, and methodology.
- Environment-related risk mitigation.
- Use of renewable energy sources.
- Eco-labeling of IT products.

(Andreopoulou, 2012) defines four dimensions of the contribution of green ICT to sustainability:

- Reduction of energy consumption/carbon footprint while production and usage towards low carbon economy
- Rise of environmental awareness with information diffusion, training and education
- Effective communication for environmental projects and networks
- Sustainable environmental governance.

However, despite many enviroment-related decisions which are taken in specific development points of a system's architecture. The overall understanding of the effect of those decisions and the impact of ICT in the whole business is still complex. The architect should see the big picture impact and all the pieces fitting together productively. Due to this, a holisc design is a must for green ICT systems, in order to comprehensively and effectively address the effects of ICT on the enviroment and the following paths (Microsoft, 2008; Murugesan, 2008):

a) *Green use of ICT systems***:** Reduction or optimization of its energy consumption.
b) *Green disposal of ICT systems:* Renovate and reuse old ICT components and properly recycle unwanted electronic equipment.
c) *Green design of ICT systems:* Design energy efficient and enviromental standard compliant components, computers, servers, and cooling equipment.
d) *Green manufacturing of ICT systems:* Minimize or eliminate the enviromental impact of electronics manufacture.

"Adopting green IT practices offers businesses and individuals financial and other benefits." (Murugesan, 2008)

Green ICT carryies along several benefits such as: better energy effiency which can be translated into economical savings, new competitive landscapes, taxes and regulations complying, new reseach measuring tools, and grid computing enviroments(Harris, 2008; Murugesan, 2008). Furthermore, (Unhelkar, 2011) defines four emcompasing layers of green ICT that could support the vision of a enterprise(Figure 4).

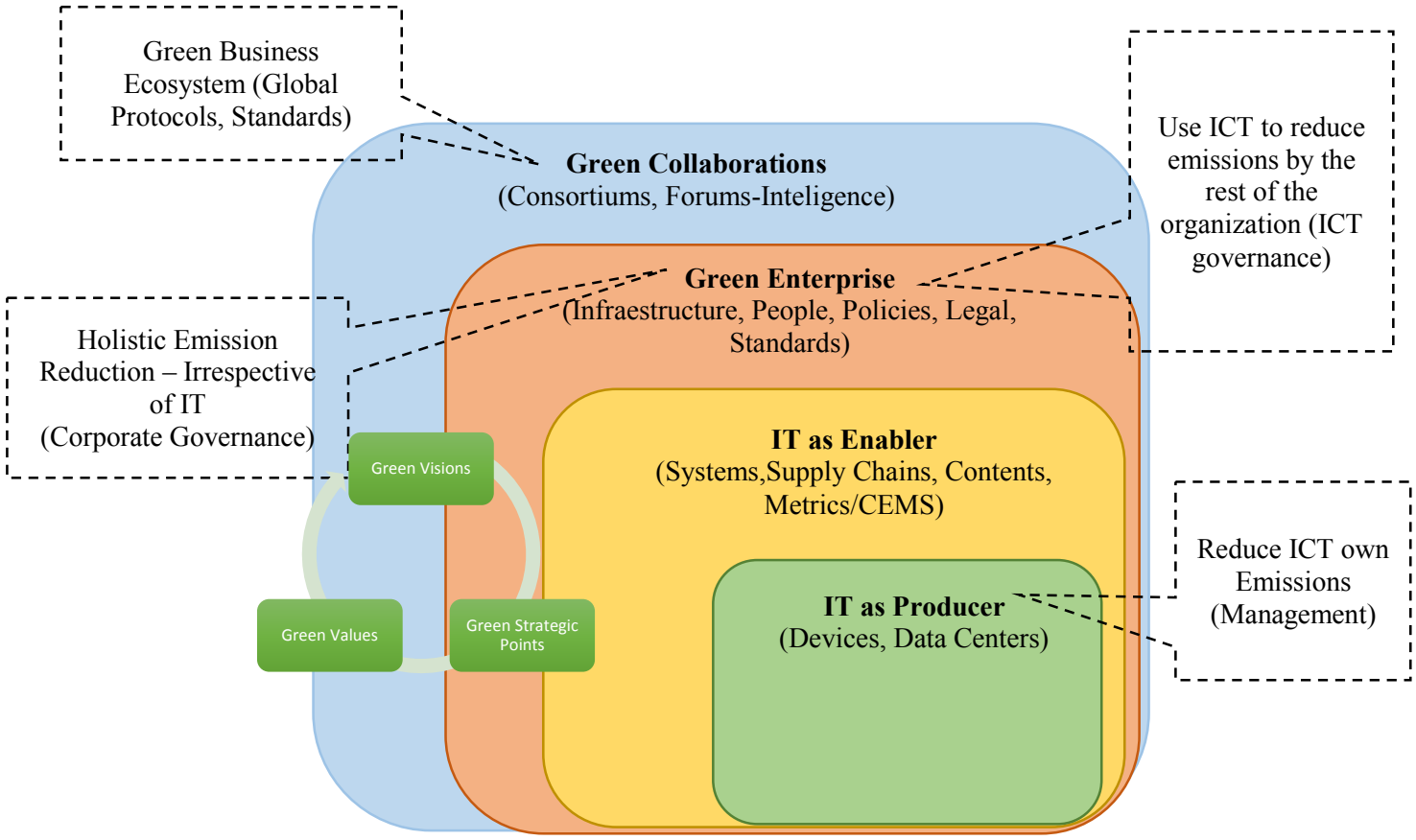

Figure 4: Envision of a green enterprise beyond green ICT (Unhelkar, 2011).

3.3.1.3 ICT for Greening

ICT for greening or ICT for sustainable growth refers to the utilization of informatics in the interest of the natural environment and the natural resources regarding sustainability and sustainable development (Andreopoulou, 2012). Such as using IT for green education, for collective actions, and for spreading ideas (Zapico, 2012).

Six policy areas have been selected as priorities by (Ernst and Young, 2011): 1) Energy Efficiency of the ICT Sector (greening of ICT), 2) Smart Sustainable Cities (greening with ICT), 3) Energy Efficient Buildings (greening with ICT), 4) Smart Grids (greening with ICT), 5) Water Management (greening with ICT), 6) Climate Change Management (greening with ICT). (The Climate Group, 2008) defines four major sectors where ICT can enable sustainabiliy: 1) Smart motor systems, 2) Smart logistics, 3) Smart buildings and 4) Smart grids.

3.4 Sustainable Software Engineering

Contrary to the common assumption that software is "environmentally friendly" simply because it is virtual, the processes and methods used to develop, maintain and deploy software do have an environmental, social and economic footprint (Albertao, Xiao and Tian 2010).

Sustainability and sustainable development have become increasingly important concerns over the past decades. Software systems strongly affect our daily lives. Thus, supporting sustainability in software engineering explicitly would impact the process of making our planet greener in the long run and improving our societies, our economies, as well as our environment (SE4S, 2014). Although, there is no common definition for sustainable software engineering yet, engineers are already approaching practically specific topics that are related to a sustainable impact such as: green IT, efficient algorithms, smart grids, agile practices, and knowledge management (Penzenstadler 2012). In addition, (SE4S 2014; Albertao, Xiao and Tian 2010) points out that a focus on requirements engineering (RE) and quality assurance (QA), are key elements to improve the sustainability performance in software-related projects.

A generic sustainability model, was proposed by (SE4S 2014), which aims at supporting requirement engineers to: 1) to analyze their projects according to the different dimensions of sustainability, 2) instantiate concrete goals for the project, and 3) choose actions for improvement. In addition, SE4S claims that sustainability is part of the non-functional requirements in a project.

(Albertao, Xiao and Tian 2010) proposed a framework with specific metrics to measure the sustainability performance of software projects. The metrics used to asses each property, were taken from the finding of the Urban Water Management Platform (UWMP), a software project developed by IBM research. This technical report, recommends analyzing and assessing specific properties, within the three sustainable dimensions (Economy, Environment, and Society), in three development phases, as detailed in the following list:

1. Development-related properties
    a. Modifiability
    b. Reusability
    c. Portability
    d. Supportability
2. Usage-related properties
    a. Performance
    b. Dependability
    c. Usability
    d. Accessibility
3. Process-related properties
    a. Predictability
    b. Efficiency
    c. Project's Footprint

However, most of these metrics have neither a good nor a bad result. They aim at being informative and at being used as basis for continuous improvement (Albertao, Xiao and Tian 2010).

(Penzenstadler 2013) claims that the sustainability aspects in software engineering are heavily present in the following project phases: 1) Development process, 2) Maintenance process, 3) System production and 4) System usage, because of their impact for human, economic, and environmental dimensions. Yet, there is not a specific body of knowledge of sustainability in software engineering which can provide specific guidance to fulfill all sustainability aspects from a development perspective. However, engineers already empirically approach topics related to sustainability. Still, there is a lack of a common and tangible definition of the concept of sustainability in the discipline (Penzenstadler 2012).

3.5 Software engineering gaps and limitations towards games development and sustainability

In the software indusry, games are becoming a progressively influential area, because of their massive impact and global revenues (Nayak 2013). However, despite their rise in importance, there is a gap in models and methodologies that support game development from a SE perspective (Ampatzoglou & Stamelos 2010, Kasurinen and Laine 2012). In larger scale, the lack of SE research in game development implies that 1) SE methods have been strictly developed and framed for software development, and that 2) to train game developers, educators and companies should focus on developing creative skills along with engineering skills (Murphy-Hill, Zimmermann and Nagappan 2014).

It has been claimed that games have significant differences from "traditional" software development. (Murphy-Hill, Zimmermann and Nagappan 2014; Stacy and Nandhakumar 2009; Baba and Tschang 2001). However, game development is not a fully related creative industry (Tschang 2005) but rather a software engineering intersected field (Ampatzoglou and Stamelos 2010), sharing common problems and challenges with it (Petrillo et al. 2008; Petrillo et al. 2009; Petrillo and Pimienta 2010). In this context, there is a need of SE methods that support modern game development processes (Kasurinen and Laine 2012; Murphy-Hill, Zimmermann and Nagappan 2014).

(Murphy-Hill, Zimmermann and Nagappan 2014) details the essential differences between ~~the~~ game development and software development: 1) a lack of strict functional requirements, 2) a small design phase based on experience, expertise, and emotions, 3) less automated testing processes, 4) delayed modules' maintenance (for non-cloud games), and 5) highly evolved configuration management techniques (due to high number of assets). As a result, game development is often an unpredictable and highly iterative and creative process, which makes the agile methodologies a close fit, thus encouraging the game industry to largely apply agile practices in their work (Kultima and Alha 2009).

The inclusion of SE techniques in games development is not widely spread among game developing companies, due to a gap between the traditional SE methods, their documentation-centric approaches, and the rapid iterative, not documentation-centric and highly creative game development processes. A gap of knowledge, the lack of tools, processes, methods from SE that can be tailored and implemented into the young video game industry (Laine, 2012).

Green IT and IT for Greening are concepts that have been raising attention in the modern IT industry (The Climate Group 2008). However, it seems that traditional SE still do not support sustainabiliy in software projects (Penzenstadler 2013). Still, the recent apparition of green software engineering approaches~~,~~ might represent an opportunity for games development. This approach focuses/there approaches focus on product-specific processes that can be easily assessed due to its/their effect on the society, the economy, the environment, and the software development itself. (Green Software Engineering., 2014). For instance, the following four sustainable principles which (Penzenstadler 2013) catalogs, could address and improve determined common issues in game development described by (Murphy-Hill, et al., 2014) in the list below, by: 1) a responsible use of ecological, human and financial resources, 2) continuous monitoring of quality and knowledge management, 3) using Green IT principles and sustainable produced hardware components and 4) having a responsible impact in society, economy, and ecology. In addition, sustainability is not supported by traditional SE methods, such as game development processes.

Games issues detailed by (Murphy-Hill, et al., 2014):

1. “Architectural debt” from a poor design phase, which affects the lifespan of the game
2. Undisclosed details about how agile process integrate specific software engineering practices
3. High number of code parts which are thrown away instead of being reused
4. Maintenance delay for non-cloud games (the game is only maintained if it is successful)
5. Development physically demanding characterized by long hours of work
6. Suboptimal effects from games testing such as motion sickness

4 RESEARCH QUESTIONS AND METHODOLOGY

In this chapter, the research questions are described and the applied research methodology and data collection processes are detailed. Small discussions and reasoning behind the selection of the applied research approaches are held.

4.1 Research Problem and Questions

Developing games is a broad task that requires advanced creative and technical knowledge in a dynamic and agile context (Godoy & Barbosa, 2010; Petrillo & Pimienta, 2010). Therefore, establishing proper development practices that can ensure the efficient use of resources and metrics during a game development lifecycle is important. Yet, it seems as though games have significant differences from "traditional" software development (Murphy-Hill, et al, 2014; Stacey & Nandhakumar, 2009; Baba & Tschang, 2001; Ampatzoglou & Stamelos, 2010). But still, share common problems and challenges with software engineering (Petrillo et al. 2008; Petrillo et al. 2009; Petrillo & Pimienta, 2010). However, very little is known about issues affecting the game industry (Godoy & Barbosa, 2010). In addition, migrating games to different platforms is becoming a modern trend (Furini, 2007), which requires that game developing companies perform more efficient and rapid processes, reusing as many assets as possible (Murphy-Hill, et al., 2014) and minimizing the time of development. However, the current models and methodologies that guide this process for game developing have a gap of knowledge and do not fully adapt to the peculiarities of game development (Godoy & Barbosa, 2010; Kasurinen & Laine, 2012).

In addition, (SE4S, 2014; Penzenstadler, 2013) state that traditional SE does not support sustainability. Supporting sustainability in software engineering would explicitly impact the process of making our planet greener in the long run and improving our societies, economies as well as our environment (SE4S, 2014). Despite the lack of support from SE to sustainability, engineers are already approaching practical topics such Green IT or IT for Greening, but still lack a common tangible definition for sustainability in their field (Penzenstadler, 2013; Penzenstadler, et al., 2012; Christen & Schmidt, 2012; Albertao, 2004). Consequently, a need for a body of knowledge with clear practices for RE and QA

and a common definition of sustainability in SE is required to bring together the best practices from software development that can be easily measured in terms of economic, environmental, and social impact (SE4S, 2014; Albertao, et al., 2010; Penzenstadler, 2012).

This thesis, studies the intersection between games development and sustainability and focuses on how game developers approach sustainability while doing their work and what are their definitions, opinions, practices, and priorities regarding this matter.

The (Kitchenham, et al., 2002) **explorative approach** was the chosen method in order to approach the research problem "What are the costs and requirements imposed during a video games migration process to a new platform on the game developing organizations and identify the most expensive, work-intensive and possible green components related to this activity?". In order to accomplish this approach, the problem was divided into a group of research questions (RQs), which were addressed through a quantitative survey study (See Table 3).

Table 3: Research Questions (RQ), Goals and Survey Structure

Research Question (RQ)		**Goal**	**Survey Section**
RQ1: What are the main trends among game developer companies?		Identify main platform development trends	Section 1: Basic Information
RQ2: How concerned are game developer companies about green aspects?	RQ2.1: Relation between role and opinion about eco-impact factors	Identify the green concerns and relations in game developer companies	Section 2: Green Aspects and Marketing
	RQ2.2: Relation between company age and opinion about eco-impact factors		
	RQ2.3: Relation between role and opinion about green activities involvement opinion		
	RQ2.4: Relation between company age and opinion about green activities involvement opinion		

<table>
<tr><td rowspan="4">RQ3: What are the characteristics of game developer companies regarding their software engineering methods?</td><td>RQ3.1: How many companies use methodologies?</td><td rowspan="4">Identify the framework of software development companies</td><td rowspan="4">Section 4: Migration Process and Development Work</td></tr>
<tr><td>RQ3.2: What are the most common development methodologies?</td></tr>
<tr><td>RQ3.3: How mature are their processes?</td></tr>
<tr><td>RQ3.4: What are the most intensive phases?</td></tr>
<tr><td rowspan="3">RQ4: How experienced are game developer companies with software migration processes?</td><td>RQ4.1: How -many companies have migration experience?</td><td rowspan="7">Explore the components of the migration process and identify key factors</td><td rowspan="7">Section 4: Migration Process and Development Work</td></tr>
<tr><td>RQ4.2: What is the relation between company age and software migration experience?</td></tr>
<tr><td>RQ 4.3 What is the relation between methodology use and migration experience?</td></tr>
<tr><td rowspan="3">RQ5: How is a software migration process in game developing companies?</td><td>RQ5.1: How long a migration process takes in average?</td></tr>
<tr><td>RQ5.2: What is the relation between company age and the time a migration process takes?</td></tr>
<tr><td>RQ5.3: What is the representation of a migration process?</td></tr>
<tr><td colspan="2">RQ6: What are the most intensive/decisive factors for a software migration in game developer companies?</td></tr>
</table>

4.2 Research Methods

In order to approach the RQs empirical research guidelines from (Kitchenham, et al., 2002) and quantitative survey methods according to (Fink, 2013) were applied. The main key points were the following three: 1) general information overviews, 2) software development processes and 3) companies' concerns.

4.2.1 Quantitative Study

A quantitative study focuses on collecting numerical data and generalizing it through groups of persons. Its methods emphasize on objective measurements and numerical analysis of data gathered through polls, questionnaires or surveys. (University of Southern California, 2013) . According to (Kitchenham, et al., 2002), the survey method is a proper method to collect data as part of an empirical research which gathers information from a standardized sample of individuals related to software engineering activities.

Surveys are information collection methods used to describe, compare, or explain individual and social knowledge, feelings, values, preferences, and behavior. There are two types of surveys: Self-administered (mailed or online) and Interview (By phone or in person). (Fink, 2013). For this research, a **self-administered structured and online survey,** was applied. This means that the survey was accessed and completed online using any internet connected device, and the respondents were responsible of this activity on their own, without personal help.

(Fink, 2013) Affirms that surveyors prefer online surveys and that respondents are becoming more used to them. In addition, Fink details some of the advantages and disadvantages attached to an online survey such as advantages: 1) Worldwide information is obtained immediately ("real time"). 2) It can provide the respondent with explanations of unfamiliar words and help him with difficult questions. 3) It is easy to send many reminders. 4) It is easy to process data because the response can automatically be downloaded to a spreadsheet data, analysis package or database. The disadvantages include: 1) the surveyor needs reliable e-mail addresses. 2) The respondent must have reliable internet access. 3) Questionnaires

may look different in different browsers. 4) There is no method for selecting random samples from general e-mail addresses.

In addition, in order to comply with **ethics privacy and confidentiality** (Fink, 2013), a preliminary section was added to the survey which contained: 1) Explanation of data storing procedures and 2) An offer to answer inquiries.

4.2.2 Design Methods

The selected design method was the **cross-sectional** defined by (Fink, 2013) because data was collected only at a single point of time. The individuals are considered as **unit of observation** (UO) since the survey allowed receiving multiple respondents in an organization (Kitchenham, et al., 2002). **The survey design** followed a structured organization in order to approach the RQ detailed in Table 3. In addition, the detail of **questions design** (Fink, 2013) included in each section of the survey is given in Table 4.

Table 4: Survey Design, question design detail

Structure Section	Number of Questions	Question Design Type
Section 1: Basic Information	4	Ordinal scale question.
		Multiple options with ordinal order.
		Closed question.
Section 2: Green Aspects and Marketing	4	Rank order scale question.
		Closed question.
		Numeric discrete scale.
Section 3: Game Business and Product Design	5	This section is not focus of this thesis study, since it belongs to another research work.

Section 4: Migration Process and Development Work	7	Semi open question.
		Rank order scale question.
		Constant sum question.
		Closed question.

4.2.3 Sampling and Data Collection

Probabilistic random sampling methods described by (Fink, 2013) were used. Table 5 resumes all the methods and details used for the data collection.

Table 5: Survey methods

Method	Detail
Survey method	Online
Design method	Cross-sectional
Number of sample groups	1
Number of survey sections	4
Time duration	1 month (From March 2014 to April 2014)
Selection method	Random Sampling
Sample requirements	Game developer companies, with some experience publishing at least one games
Survey administration	Via webpool tool from Lappeenranta University of Technology (LUT)
Processing the data	Data is automatically entered from survey to database via webpool.
Survey distribution	Invitations to fill the survey to a random sample: 1) Via Emails, 2) Via Twitter. Invitation to collaborate to SOCES game developer partner companies.
UOs Answers collected	28
UOs contacted (times form opened)	514

4.2.4 Data Analysis

The results of the data analysis are presented in the section 4.3 and they follow the frequentist approach detailed by (Kitchenham, et al., 2002), and the methods described by (Fink, 2013) which include descriptive statistics with averages, summaries, cross tabulations, and correlations. Kendall's tau correlation co-efficient is a statistic method used to quantify the relation or association between two measured variables. Where correlations are classified as follows: small (τ=.10), medium (τ=.30) and large (τ=.50). The reason Kendall's tau statistic was used lays on its tendency to be more accurate in small samples (Lösch, 2006). The responses were analyzed using Excel 2013 and Matlab R2013a.

The independent variables of the study were: surveyed role, company's age and development methodology. **The dependent variables** of the study were: development platform, opinion about eco-factors, and production of marketing items, green activities involvement, processes maturity, project phase's consumption, migration developments experience, migration development process maturity, and migration development driving factors.

Figure 5 resumes the sources of the 33 UOs' answers that were collected. These responses were distributed between five different roles in the organizations. In addition, the 75% of total answers have as source positions directly related to the game development phases such as technically-oriented, artistically-oriented and project-level management employees, and 25% from organization level and administrative related positions.

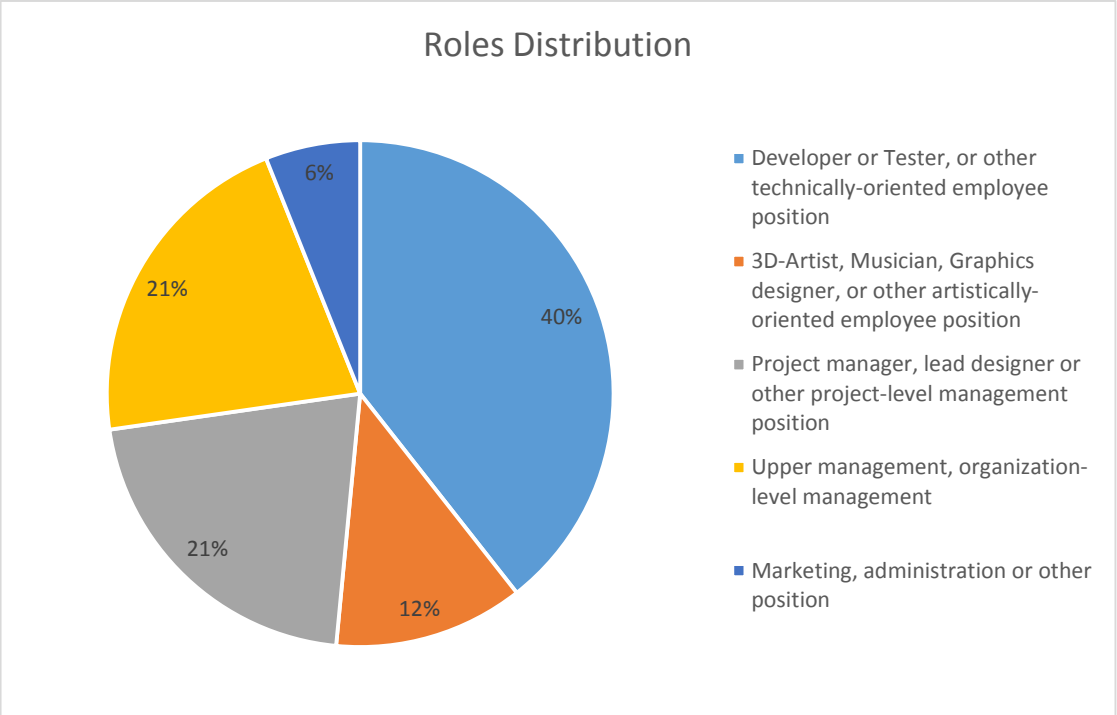

Figure 5: Roles

4.3 RESULTS

In this chapter, the cross-section survey results in which 33 UOs participated are described, organized by the research questions order. Two submissions from the role "administration or other", were rejected from the results analysis since these positions did not have professional tasks related to the development processes or business management tasks.

4.3.1 RQ1: What are the main trends among software developer companies?

There are **two main trends** among the surveyed UOs: 1) Mobile as a preferred development platform, 2) Digitalization of their marketing.

From Figure 6 it is possible to observe that the main development platform among companies is mobile, gathering companies with fresh experience on the market (less than 2 years), companies which have between two and five years of experience, to companies with more than ten years of experience. This finding complies with the forecasted growth of this industry provided by (PricewaterhouseCoopers 2013; Forbes 2014).

Figure 7 suggests that 94% of UOs do not perform hardware developments but 6% do. This represents a trend to use the already spread hardware components instead of the creation of new ones. Furthermore, this is the result of the main trend of development platforms showed in Figure 16 which is mobile platforms development such as for tablets and smartphones.

Regarding marketing in game developing companies (Figure 8), the 87% of UOs do not produce physical marketing items anymore while the 13% of them still produce articles such as toys, trinkets, clothes, posters, stickers, and others. This trend is common for all entertainment and media companies where digital content became the best components on their marketing channel (PricewaterhouseCoopers, 2013).

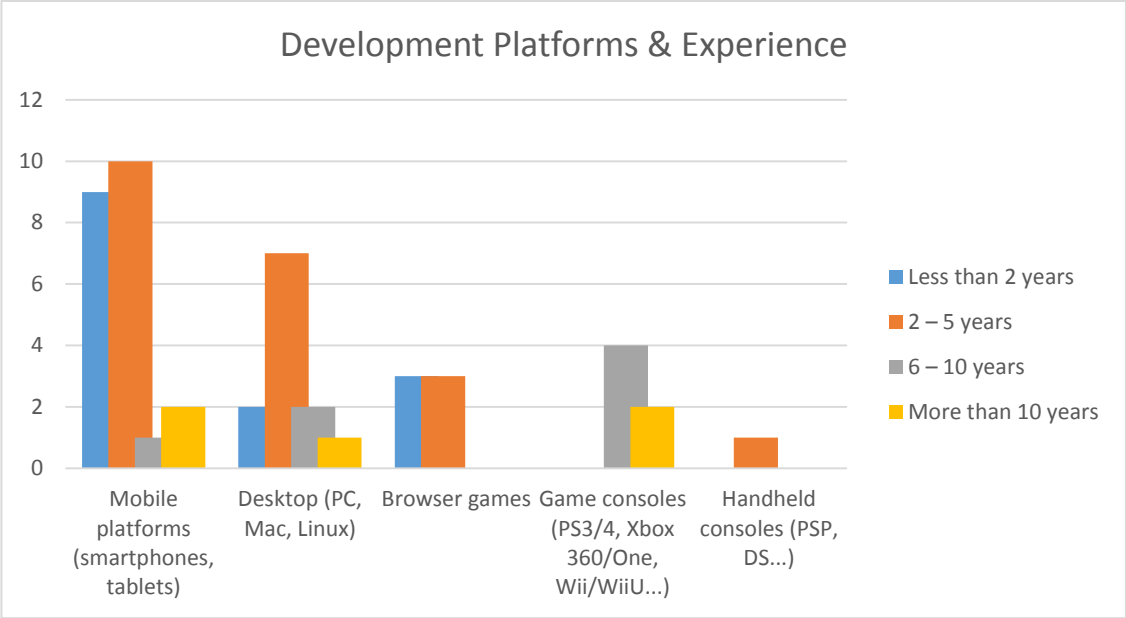

Figure 6: Development Platforms and Companies Age

(N=31)

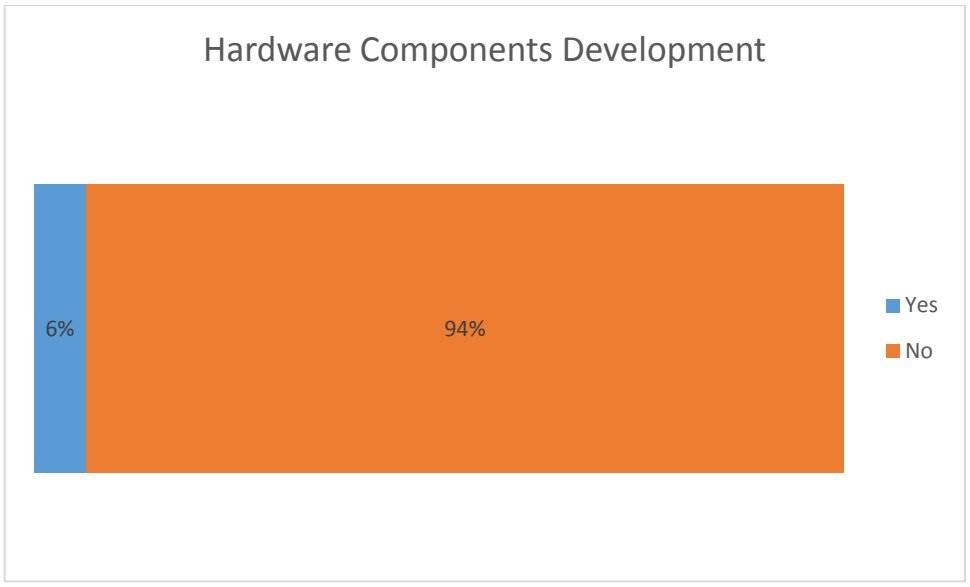

Figure 7: Hardware Components Development

(N=31)

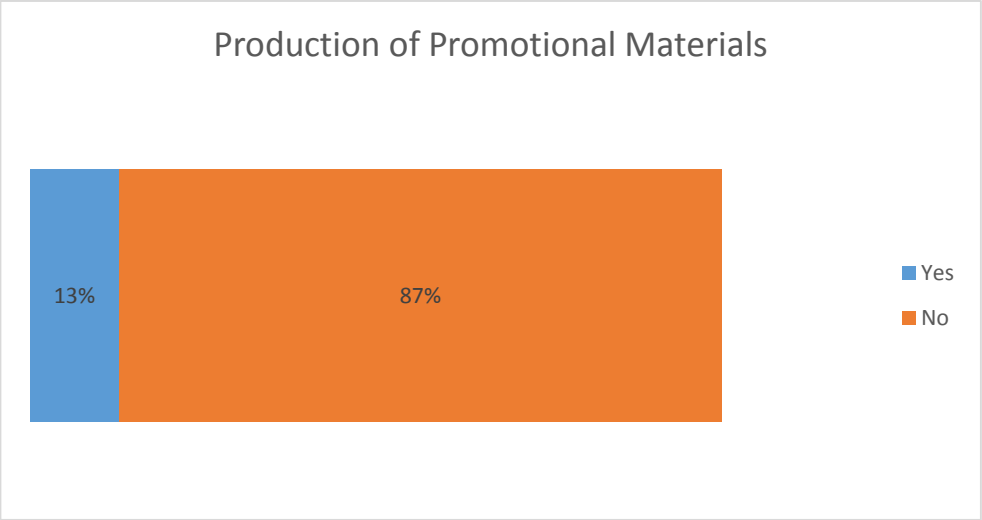

Figure 8: Marketing Items

(N=31)

4.3.2 RQ2: How concerned are game developer companies about green aspects?

Game developing organizations seem not likely to involve neither recognize eco-impact factors (e.g. code reusability, energy-efficient programing, social awareness impact, reuse or repurpose of old hardware components or marketing materials or support to legacy systems) in their daily work (Figure 9). In all observed eco-impact categories, the average rate of involvement was between 1.4 and 3.5 (where 1= No involvement, 5= Focus area), code reusability obtained the highest rank with 3.5 average rate. However, the lack of a common definition of sustainability in the field (Penzenstadler 2013) might have caused an unclear understanding of eco-impact factors in this specific section.

Table 6: Eco-Impact factors importance

(N=31)

Question	Average rate
Based on your experiences or observations, rate the importance of eco-impact factors (such as reusability of components, energy efficiency, long term support, social impact)	2.3

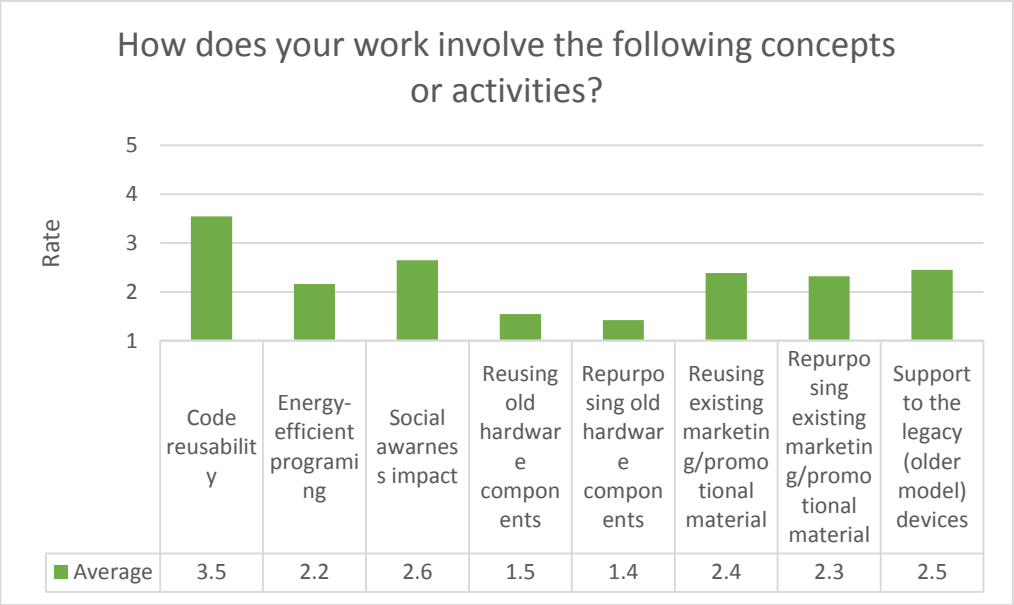

Figure 9: Green activities involvement

(N=31)

RQ2.1: Relation between role and opinion about eco-impact factors

There was found a low correlation (τ=.10)-between the role and the opinion about eco-impact factors (see Table 7). The found coefficient was negative, meaning that, the higher the role of a person goes, the lower becomes the appreciation for eco-impact factors.

Table 7: Correlation between role and eco-impact factors opinion

Kendall's tau correlation (N=31)

Right-Tailed Correlation Tests

Correlated variables		**Coefficient**	**Correlation**
Role	Opinion about eco-impact factors	-0.1002	Low Correlation

RQ2.2: Relation between company age and opinion about eco-impact factors

In Table 8, it is possible to see that there is a low correlation (τ=.10) between the company's age and the opinion about eco-impact factors (see Table 6), meaning that, as a company's age increases, its appreciation for eco-impact factors declines.

Table 8: Correlation between company's age and eco-impact factors opinion

Kendall's tau correlation (N=31)

Right-Tailed Correlation Tests

Correlated variables		Coefficient	Correlation
Company's Age	Opinion about eco-impact factors	-0.1237	Low Correlation

RQ2.3: Relation between the role and opinion about green activities involvement opinion

Through right-tailed correlation tests, the relations between the role and the eight green activities involvement (Figure 10) were explored. Table 9 presents the findings where low positive correlations were found in three green activities involvement, meaning that the higher the role of a person goes, his/her preferences for involving the following activivies decreases: 1) Repurposing old hardware components, 2) Reusing existing marketing/promotional material, and 3) Repurposing existing marketing/promotional material. This adheres to the findings from (Murphy-Hill, Zimmermann and Nagappan 2014; Godoy & Barbosa, 2010) which report that in game organizations, non-engineering management often does not respect software engineering's good practices involvement, because those activities have no immediate impact, and when those activities are introduced into game development it normally comes from engineers.

Table 9: Correlation between role and green activities involvement opinion

Kendall's tau correlation (N=31)

Right-Tailed Correlation Tests

Correlated variables		Coefficient	Correlation
Role	Code reusability	-0.0029	No Correlation
Role	Energy-efficient programming	-0.0777	No Correlation
Role	Social awareness impact	0.0338	No Correlation

Role	Reusing old hardware components	0.0378	No Correlation
Role	Repurposing old hardware components	0.1792	Low Correlation
Role	Reusing existing marketing/promotional material	0.2109	Low Correlation
Role	Repurposing existing marketing/promotional material	0.1308	Low Correlation
Role	Support to the legacy (older model) devices	0.0906	No Correlation

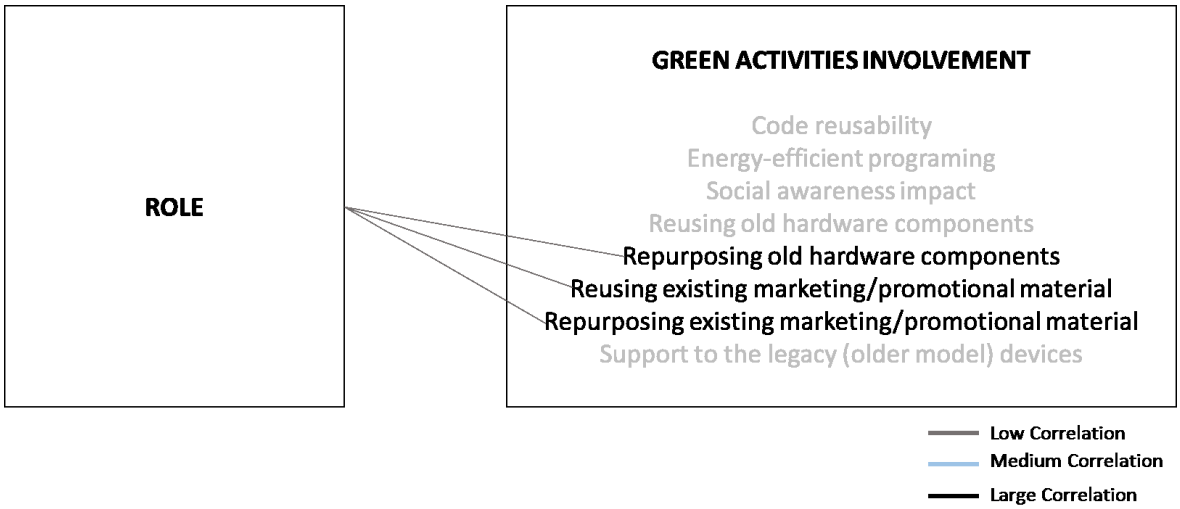

Figure 10: Correlation between role and green activities involvement opinion

Kendall's tau correlation (N=31)

Right-Tailed Correlation Tests

RQ2.4: Relation between company age and opinion about green activities involvement opinion

Through right-tailed correlation tests the relations between the company's age and the eight green activities involvement (Figure 11) were explored. Table 10 presents the findings, low (τ=.10) and large negative (τ=.50). Correlations where found in five green activities involvement.

As company's age increases, their willingness for 1) Reusing existing marketing/promotional material and 2) Repurposing existing marketing/promotional material falls (large negative correlation coefficients). This might be related to the modern preference to use digital channels for marketing purposes. On the other hand, old companies have a slightly lower appreciation for 1) reusing old hardware components and 2) performing social awareness impacts (low negative correlations).

Table 10: Correlation between company's age and green activities involvement opinion

Kendall's tau correlation (N=28)

Right-Tailed Correlation Tests

Correlated variables		**Coefficient**	**Correlation**
Company's Age	Code reusability	0.0411	No Correlation
Company's Age	Energy-efficient programming	-0.0369	No Correlation
Company's Age	Social awareness impact	-0.142	Low Correlation
Company's Age	Reusing old hardware components	-0.0813	No Correlation
Company's Age	Repurposing old hardware components	-0.1883	Low Correlation

Company's Age	Reusing existing marketing/promotional material	-0.5428	Large Correlation
Company's Age	Repurposing existing marketing/promotional material	-0.5057	Large Correlation
Company's Age	Support to the legacy (older model) devices	-0.0525	No Correlation

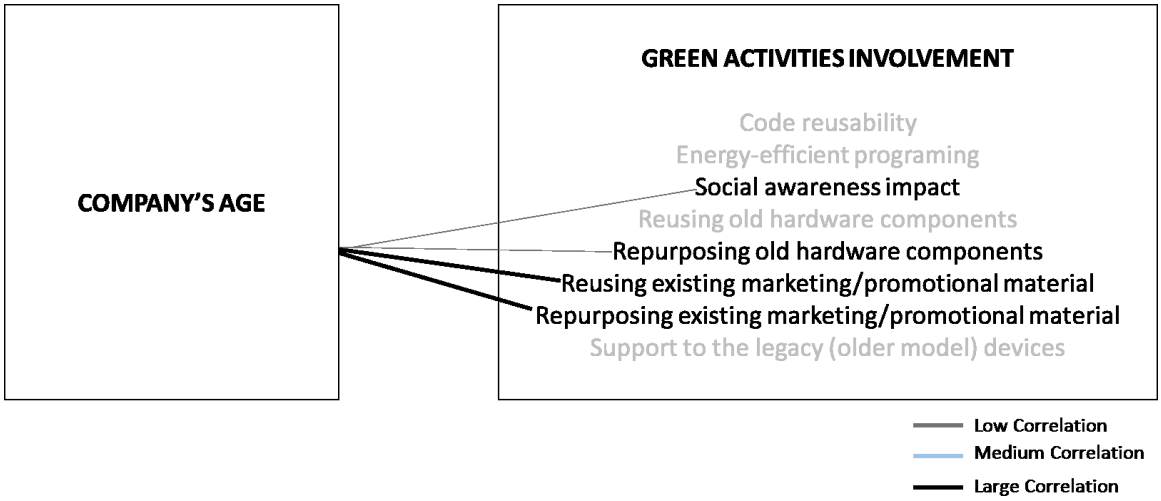

Figure 11: Correlation between company's age and green activities involvement opinion

Kendall's tau correlation (N=28)

Right-Tailed Correlation Tests

4.3.3 RQ3: What are the characteristics of game development companies regarding their software engineering methods?

In order to answer the research question number three, four sub-research questions were formulated (RQ3.1, RQ3.2, RQ3.3, RQ3.4) to find the average number of companies following a development methodology and to find which are their preferred methodologies, how they describe their processes, according to maturity levels (Carnegie Mellon University, 2014) and finally, which phases the UOs identify as the most work intensive during a game development project.

The main findings of this section were:

- Only 39% of UOs follow a development methodology.
- Scrum is the most common agile methodology followed by game developing companies.
- Most of the UOs describe their processes as *"Processes are often reactive (We only react to problems that actually happen, and only prepare for the most probable problems.)"*
- The UO's identify four phases as the most work intensive ones: Programming, Design development, Art/Audio Production, and Levels Creation.

RQ3.1: How many companies use methodologies?

According to the analysed data, 61% of total UOs do not follow any systematic development methodology. Consequently, 39% of the UOs follow a software development methodology, which could be characterized as "systematic". Out of those organizations, 67% (26% of the total) identified "Scrum" or "Partial Scrum" as their method, while other 33% (13% of the total) recalled "Prototyping" and "Other agile".

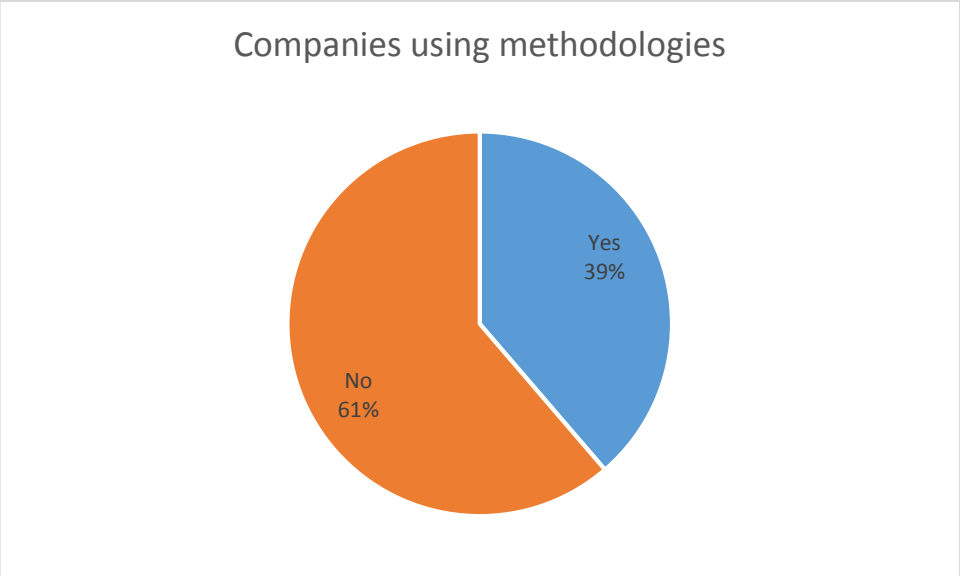

Figure 16: Software development methodology followed by UOs

(N=31)

RQ3.2: What are the most common development methodologies?

The 39% of OUs that reported they follow some development methodology, also provided information about which kinds they use. Table 11 resumes their responses where a 75% reported following A and B methodologies, 17% of UOs described that they use C and D methodologies, finally, another 8% of UOs follows E methodology.

Table 11: Development methodologies

(N=12)

ID	**Methodology**
A	Scrum
B	Partially Scrum
C	Prototyping
D	Agile Methods, Fast prototyping
E	Agile software development

RQ3.3: How mature are their processes?

UOs were asked to define their processes according to a list of definitions based on the CMMI maturity levels descriptions (Carnegie Mellon University, 2014). From Figure 12, according to the average rate each process description received, most of the UOs define their processes as "*Often reactive (We only react to problems that actually happen, and only prepare for the most probable problems.)"* which represents the level one in the maturity scale of CMMI. UOs also reported that their way of working is far from a product design which does not change much during the development, after the initial product design is completed which complies with the highly iterative and dynamic nature of video game industry.

Figure 12: Processes maturity definition

(N=31)

RQ3.4: What are the most intensive phases?

As can be observed in Figure 13, UOs rated four phases as the most intensive of development according to their experience on games development. Those phases are: prototype development, programming, art/audio production, and alpha/beta. This finding can be related to the technical intensity of those stages. On the other hand, the UOs rated the following two as the less intensive phases: testing, gold master/crunch time. This finding could be related to the maturity level of the majority of UOs which is not focused on metrics, quality improvement, or long term support because of the mitigation of problems that appear during the development phases.

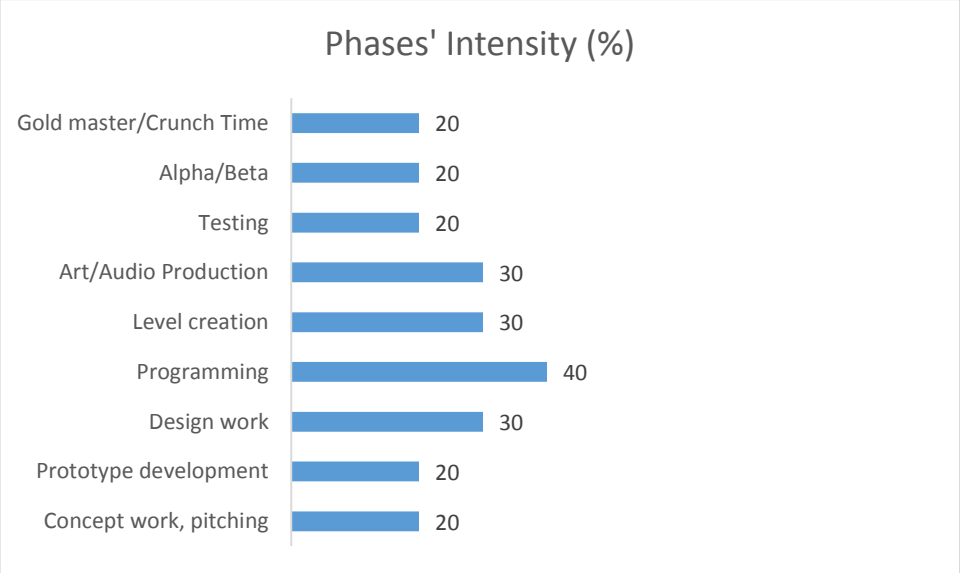

Figure 13: Development phase's intensity
(N=31)

4.3.4 RQ4: What is the experience of game developing companies regarding software migration processes?

The research question number four was focused on the study of software migration processes and analyzes their experience on migration developments and whether or not is related to the company's age and the methodology use. In order to accomplish this, three sub research questions were formulated (RQ4.1, RQ4.2, RQ4.3). This question did not take into account companies which are specialized on doing migration work for game developing companies.

The main findings of this section were:

- Only 32% of UOs has experience processes related to game migration.
- The migration experience is correlated to the company's age.
- The migration experience is correlated to the use of one methodology.

RQ4.1: How many companies have migration experience?

In Figure 14, only 32% of UOs reported to have experience with games platform migration, while 68% of them have no experience regarding this topic. In RQ4.2 and RQ4.3 the relation between this finding, the company's age and the use of methodology is studied.

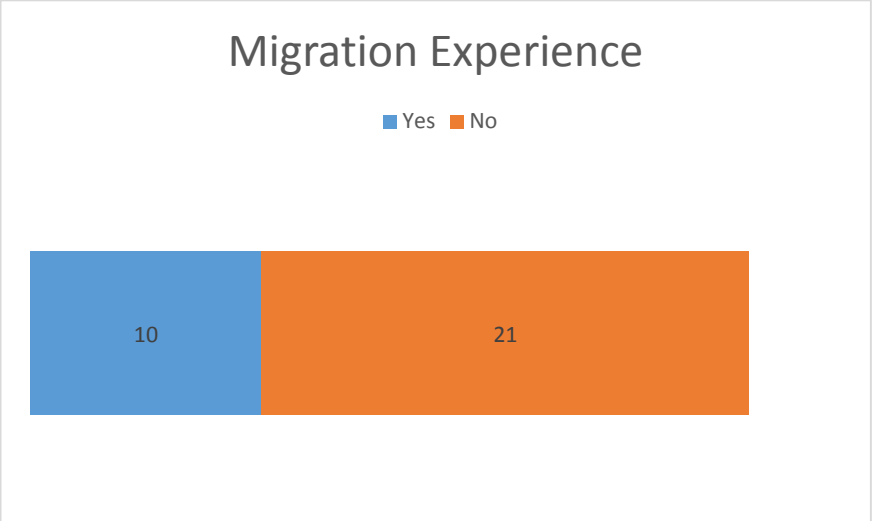

Figure 14: Migration development experience
(N=31)

RQ4.2: What is the relation between company age and software migration experience?

Through right-tailed correlation tests, the relation between the company's age and the migration experience was explored. Table 12 displays the correlation coefficient found in the UOs sample. This statistical value represents a low (τ=.10)/ medium (τ=.30) correlation. In addition, that correlation coefficient was negative, meaning that old companies are less likely to have migration experience.

Table 12: Correlation between company's age and migration experience

Kendall's tau correlation (N=31)

Right-Tailed Correlation Tests

Correlated variables		**Coefficient**	**Correlation**
Company's age	Migration experience	-0.2739	Low/Medium Correlation

RQ 4.3 What is the relation between methodology use and migration experience?

The use of a methodology is correlated to the migration experience in a game developer organization, with a positive low coefficient, which indicates that companies which follow any sort of development methodology have migration experience (Table 13).

Table 13: Correlation between company's methodology use and migration experience

Kendall's tau correlation (N=31)

Right-Tailed Correlation Tests

Correlated variables		**Coefficient**	**Correlation**
Methodology use	Migration experience	0.0175	No Correlation

4.3.5 RQ5: How is a software migration in game developing companies?

The research question number five was focused on the exploration of how a software migration processes is accomplished in the UOs, studying their duration, processes representation, and the possible correlations between the company age (which was found as a correlated variable, see Table 12) and the duration of a migration process. In order to accomplish this, three sub research questions were formulated (RQ5.1, RQ5.2, RQ5.3).

The main findings of this section were:

- 42% of UOs reports that an average migration process takes less than two months.
- The company's age is correlated to the migration experience
- The majority of UOs considers the following two descriptions as a proper representation of their migration process: *"Our migration process was informal and/or agile"* and/or *"The final costs were what we expected or estimated them to be"*.

RQ5.1: How long does a migration process take in average?

As can be observed in Figure 15, 42% of UOs report that their migration process takes in average less than two months while 58% of them take between six months to one year and half to accomplish this task. However, there were 0% of UOs that report a migration development that takes a longer period than one and half years.

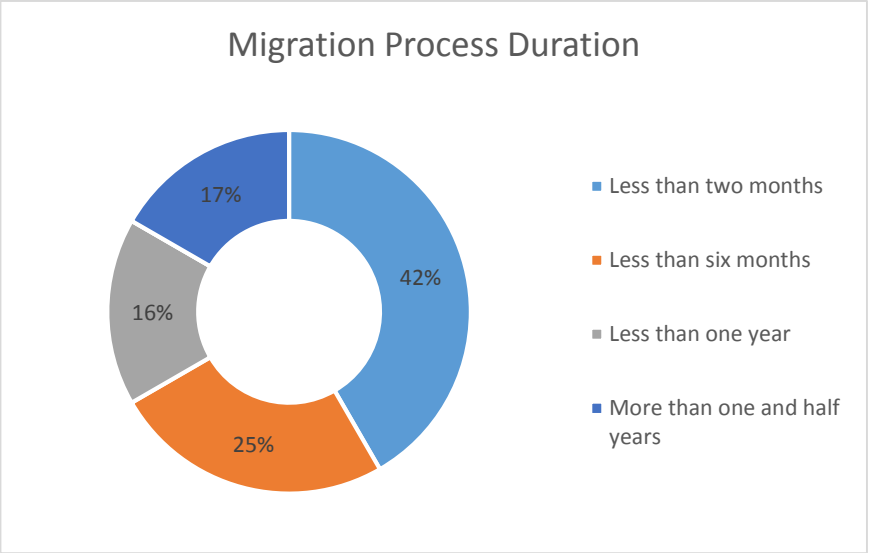

Figure 15: Migration development duration

(N=12)

RQ5.2: What is the relation between company age and the time a migration process takes?

Right-tailed correlation tests were used to explore the relation between the company's age and the migration process duration. Table 14 displays the correlation coefficient found in the UOs sample. This statistical coefficient represents a low positive correlation (τ=.10). This indicates that older companies are less likely to have a long migration process. In contrast, younger companies are more expected to have longer migration developments.

Table 14: Correlation between company's methodology use and migration experience

Kendall's tau correlation (N=12)

Right-Tailed Correlation Tests

Correlated variables		**Coefficient**	**Correlation**
Company's age	Migration Process Duration	0.1372	Low Correlation

RQ5.3: What is the representation of a migration process?

In Figure 16, UOs sample rated two descriptions as the closest representation of their migration development process: *"Our migration process was informal and/or agile"* and *"The final costs were what we expected or estimated them to be"*. In contrast, UOs sample rated the following description as the less close representation of their migration development process: *"We were able to keep metrics on the migration progress"*.

This finding could be related to the maturity level of the majority of UOs which is not focused on metrics, quality improvement or long term support because of the mitigation of problems that appear during the development phases. This complies with the finding of the research question 3.4 (RQ3.4).

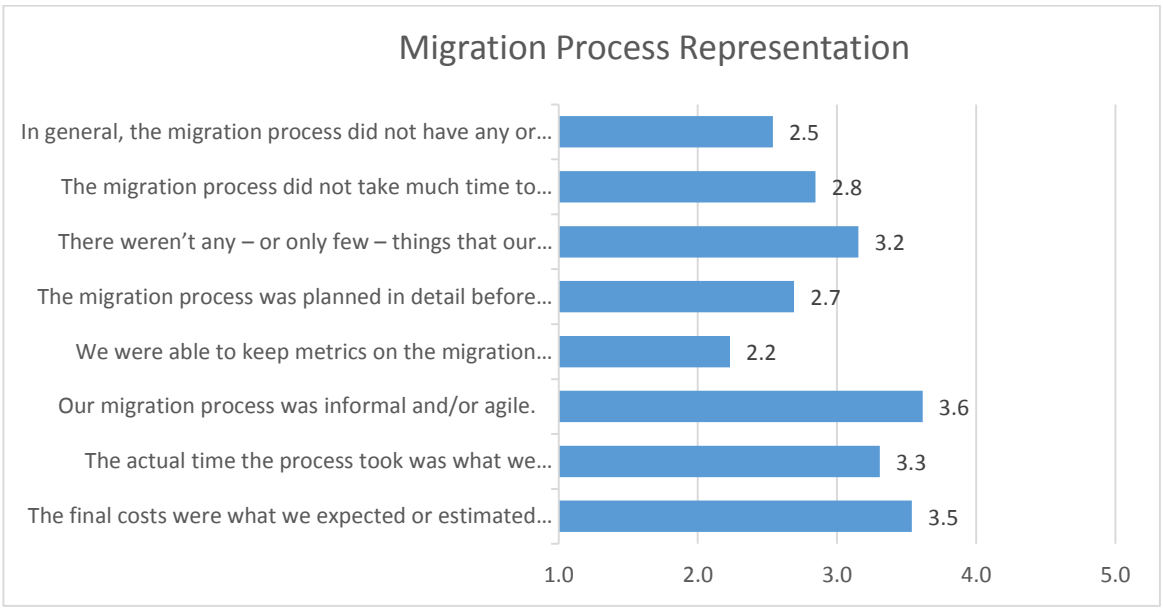

Figure 16: Migration development processes representation

(N=12)

4.3.6 RQ6: What are the most intensive factors for a software migration in a game developing company?

UOs were asked to rank the importance of nine factors into a decision of perform a game migration development. The findings represent the most intensive factors for a software migration process. The results are given in Figure 17 where a slant for five factors related to profits, investment costs (in terms of time and money), and effort (in terms of new or reusable content) can be observed. On the other hand, the UOs sample does not consider market forecasts as major influence onto a decision for migration.

Finally, the 2% of the sample reported that the eco-impact is relevant to go for migration development. However, some eco-impact factors (green activities) are already positively engaged in the role and company's age as it was showed in the RQ2 results (Figure 11). However, the lack of a common definition of sustainability in the field (Penzenstadler 2013), might have caused an unclear understanding of eco-impact factors in this specific section.

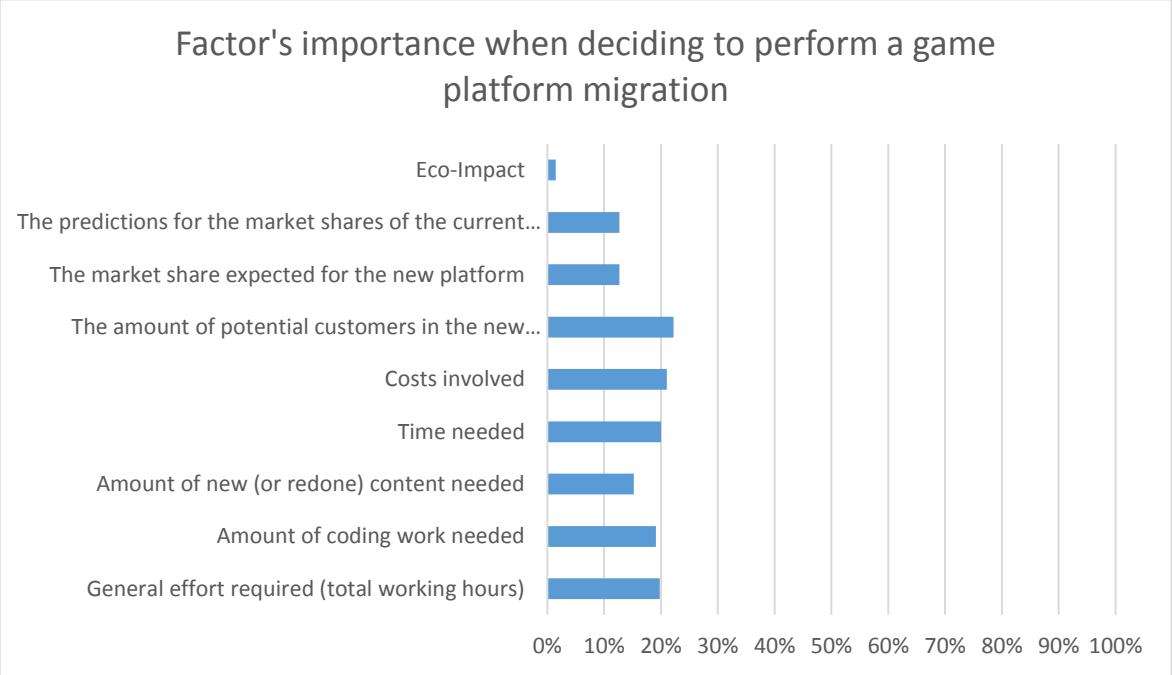

Figure 17: Intensive Factors for a software migration

(N=12)

5 DISCUSSION

The aim of this thesis report is to study the mode in which games are developed, the involved sustainable aspects, and the relevant concerns regarding migration processes. In order to achieve this goal, six research questions were formulated and included in a quantitative survey. This section will focus on discussing how the findings of each research question of this work are related with previous studies and theories.

Research Question 1: What are the main trends among game developer companies?

Based on the results presented in the previous section, there are two major trends in the game industry: 1) Mobile development as a preferred platform. 2) Digital marketing.

Trend 1, adheres to the current trend in ICT global markets where mobile devices have gained high levels of popularity and caused the contemporary society to be tagged with a "M" (Mobile) (ZTE Coporation, 2014). As a result, the mobile gaming market is progressively growing and is expected to replace console games as the largest gaming category by 2016 (PricewaterhouseCoopers, 2012). Furthermore, (Furini, 2007; Soh & Tan, 2008) describe a need for mobile games to be connected to multiplayer modes under very reliable internet connections. In this context, mobile game developers face the challenge to create multiplatform applications that can perform in a wide variety of different thin devices, which means that making a game available to the entire mobile market is very hard to achieve (Furini, 2007).

Trend 2 complies with forecasts from (PricewaterhouseeCoopers, 2013; Forbes, 2014), meaning that game industry is also part of the digital marketing revolution and a main driver towards the rise of mobile platforms where, digital content is the best component on their marketing channel (PricewaterhouseCoopers 2013). Moreover, digital marketing plays an important role as a source of profit for gaming companies among different business models (Lee 2013).

Research Question 2: How concerned are game developer companies about sustainability aspects?

"Sust*ainability is not supported by traditional software engineering methods"*
(Penzenstadler, 2013).

Regarding sustainable aspects in the game industry, the results of this study suggest that the game developers do not recognize what sustainability means in their industry and their minor concerns were eco-impact elements such as reusability of components, energy efficiency, long term support, and social impact. Thus, the UOs consider that they do not involve sustainable activities in their daily actions. Paradoxically, they recognized some practical definitions of green IT, such as design for reusability, in the activities they perform. This complies with the claim from (Penzenstadler, 2013; Penzenstadler, et al., 2012; Christen & Schmidt, 2012; Albertao, 2004), meaning that there is a lack of a common and tangible definition of sustainability in software disciplines. However, engineers already approach empirically topics related to sustainability such as green IT, efficient algorithms, code recycling, smart grids, agile practices and knowledge management. (Albertao, et al., 2010) claim that clear development, and measuring practices of sustainability for software projects would incrementally improve the sustainability of end products, creating a potential business advantage.

Another observation on sustainability concerns from the data analysis was that people are the most defining factor to either support or not support sustainability. This has also been explicitly described by (Andreopoulou, 2012). In the results, the priority green factors got from the surveys was rather low (two out of five), which seems to be related to specific upper management roles and company's age which has been also found by (Godoy & Barbosa, 2010; Murphy-Hill, et al., 2014) reports. They further find that a non-engineering management usually does not respect important engineering activities because those activities have no immediate effect. However, there are specific green practices such as code reusability which are used by developers to generate savings and profits from computational performance, energy consumption, time, and effort. This has also been spotted by (Murphy-Hill, et al., 2014) study: code recyling is already included frequently in games development

through reuse of code between subsequent game releases and reuse of game enginees or core frameworks. Furthermore, ICT efficiency alone cannot produce sustainability but rather a combination of effiency and sufficient strategies (created by people) that promote enviromentally friendly innovations that can unleash the ICT potential to support sustainability (Hilty, et al., 2011). Yet, sustainable software engineering and green informatics cannot substitute people and their behaviors which happens to be the most critical factor when dealing with environmental protection (Andreopoulou, 2012).

Systematic software reuse is a key business strategy that software managers can employ to dramatically improve their software development processes, to decrease time-to-market and costs, and to improve product quality (Griss, 1993).

Research Question 3: What are the characteristics of game developer companies regarding their software engineering methods?

"The software engineering process in video game development is not clearly understood" (Godoy & Barbosa, 2010)

The data analysis of this study suggests that only 39% of the sample follow SE methodologies. At a first glance, the practice of following a development method is not very common. However, choosing to use of agile methodologies in the games industry has become increasingly common (Godoy & Barbosa, 2010; Murphy-Hill, et al, 2014; Ampatzoglou & Stamelos, 2010; Godoy & Barbosa, 2010) because the focus of agile methods is on the product rather than on the processes used and because they can be adapted to the team reality and to the peculiarities of game development (Godoy & Barbosa, 2010). Consequently, agile, simple, highly iterative models and methodologies, which are discipline agnostic such as Scrum, are currently best option for computer game development processes (Musil, et al., 2010; Kettunen, et al., 2010; Petrillo & Pimienta, 2010; Clinton , 2010).

However, it appears that games have significant differences from "traditional" software development (Murphy-Hill, et al, 2014; Stacey & Nandhakumar, 2009; Baba & Tschang,

2001; Ampatzoglou & Stamelos, 2010). Still, they share common problems and challenges with software engineering (Petrillo et al. 2008; Petrillo et al. 2009; Petrillo & Pimienta, 2010). Yet, very little is known about issues affecting the game industry (Godoy & Barbosa, 2010). Therefore, SE and games development are intersected fields (Ampatzoglou & Stamelos, 2010) and the games industry is not a fully related to the creative industry (Tschang, 2005). In addition, (Ampatzoglou & Stamelos, 2010) after carrying a literature review study on software engineering research for games development, concluded that software engineering for computer games is a research domain that has doubled its activity during the last 5 years. Still, there is a lack of SE methods or literature that fully support game development processes (Kasurinen & Laine, 2012; Kasurinen, Laine and Smolander 2013).

According to the results, the majority of the surveyed UOs described their development process as: "*Processes often reactive (We only react to problems that actually happen, and only prepare for the most probable problems.)"*. Designing games is not a systematic but rather an ad-hoc process where ideas flow and where no strictly defined processes are used as previously found by (Kutima & Alha, 2009; Petrillo, et al., 2009; Flynt & Salem, 2005). In addition, four development phases were ranked as the most work-intensive: 1) Programming, 2) Design work, 3) Level Creation and 4) Art/Audio production, which implies that most their development process is full of intensive work. Consequently, long hours of work are required and might create "architectural debts" as result due to a poor planning phase (Murphy-Hill, et al., 2014; Godoy & Barbosa, 2010).

Research Question 4: How experienced are game developer companies with software migration processes?

Only 32% of the surveyed UOs reported to have experience with migration of a game to more than one platform. These results suggest that a need to port applications to several platforms (mostly mobile ones) is important and that it is key in the success of a game (Furini, 2007; Soh & Tan, 2008). However, this is a very recent trend and a challenge for game developer companies. Very little is known about issues such as migration expertise and their effect on the game industry (Petrillo et al. 2009; Petrillo et al. 2009).

Despite the early stage of the art in migration development methods for computer games, the term "code/tool reuse" starts to appear and attract game developers as a method to reduce the amount of wasted code between releases and repeated work. Some current examples of "code or tool reuse" take place during migration or upgrading processes and when multiple developing organizations reuse a core framework through a game engine (Murphy-Hill, et al., 2014).

Research Question 5: How is a software migration process in game developing companies?

According to the study results, the majority of UO's the following two descriptions as a proper representation of their migration process: *"The final costs were what we expected or estimated them to be"* and/or *"Our migration process was informal and/or agile"*. This specific perception characteristic was also described by (Strandén, 2012). As result, this might lead to organizations which are satisfied with their current tools and methods.

In addition, the data analysis with Kendall's correlations suggests that the software migration experience is not related to whether or not the organization follows a methodology. This finding possibly relates to an unstructured development process (Strandén, 2012). However, a low/medium link with the company's age was spotted, meaning that the older the companies are the more likely they are to have migration experience in games development. This adheres to previous studies (Callele, et al., 2005) which found a strong correlation between game companies' experience and their ability to identify issues at every level of development. This would suggest that old companies' migration experience is not related to the use of a methodology but rather to an experienced team with adjusted practices to the peculiarities of game development (Godoy & Barbosa, 2010).

Research Question 6: What are the most intensive/decisive factors for a software migration in game developer companies?

The survey results suggest that, when a company decide to either perform or not perform a software migration, the top three decisive factors are: 1) the amount of potential customers in the new platform, 2) the costs involved, and 3) the time and effort required, as previously found by (Vanhala & Kasurinen, 2014). In contrast, either sustainability matters or market forecasts have an influence when deciding whether or not to perform a migration development. The sustainability influence is possibly related to a lack of a common understanding of the concept of sustainability in software-related disciplines. Still, engineers in the field recognize their practical applications (Penzenstadler 2013; Penzenstadler, et al., 2012; Albertao, 2004; Christen & Schmidt, 2012). Furthermore, due to the highly innovative and creative nature of the games industry (Godoy & Barbosa, 2010; Murphy-Hill, et al., 2014; Ampatzoglou & Stamelos, 2010; Bates, 2004), the market forecasts cannot be accurate about which game will be the next hit and in which platform.

Limitations of the study

1. **Researcher's bias:** A researcher carried its preconceptions, the author selected the search results to be included. In addition, the author had neither a previous deep understanding of multiplatform development, nor an understanding of software engineering methods for games or the merge of those concepts with sustainability. This was minimized during the research work with the data analysis and literature review.

2. **Sample limitation:** Due to the applied contact methods (websites, social media, business contacts, and existing university partners), this study ensures that the respondents were professional game developers but there is no feasible method to measure that this study is not representing the start-up game industry or the game industry in general, since most of the collected answers belong to less than five year old organizations. Nevertheless, the current sample has a large variety of answers in terms of: 1) answers from positions related to the development of different game platforms and 2) from several countries in four continents.

3. **Methodological relevance:** As defined by (Kitchenham et al., 2002), surveys can be categorized into two types according to their design: exploratory studies, from which only weak conclusions can be drawn, and confirmatory studies, from which strong conclusions can be drawn. This study belongs to the category of exploratory, observational, and cross-sectional studies since the ultimate objective was to explore the importance of the different development processes and their possible intersection with sustainability.

4. **Statistical Relevance:** The amount of collected answers (which is 33) poses a threat to the validity of this study. This number is relatively small and its statistical relevance is difficult to corroborate. Still, there are also studies for example by (Iivari, 1996), which claims that this amount of answers is enough if the data is analyzed correctly. Consequently, this study avoided a group of threads for collecting and analyzing by collecting information on the organizational profiles and ensuring

the collected data was not localized to any specific geographical location or sub-domain of game industry.

5. **Exclusion vulnerability:** This study did not take into account companies which perform migration work as a subcontractor for the game developer organizations.

6 CONCLUSIONS

In general, the game development and software development share a number of common features, challenges, and issues. According to this study, the game development differs from software development in a sense that it tends to be unstructured or follows some agile methodology adapted to their development peculiarities. Yet, it provides results for companies profitable enough to continue with this sort of organization. However, this can be significantly improved with a simple, practical, and highly iterative SE framework that can support the creative requirements and late changes in game development without missing essential planning phases.

The UOs reported several issues with their development process and their migration processes. These issues might be mostly generated by the lack of SE methodologies that fit game development and its peculiarities, causing them to not be able to keep clear measurements during a project, since constant changes lead to wasted resources. If an organization has not solved their issues with measuring and controlling the reactivity of their processes, it is no wonder that the companies do not plan, control, measure, or improve sustainability aspects such as ICT efficiency, business healthy growth and better working environments, either.

The topic of sustainability is a current trend in many industries. To support sustainability, software and games development need to create a better and common understanding of the definition of what sustainability means in their field and how it can be achieved practically. Being sustainable is no longer an extra feature for an organization but a competitive advantage in the market. The inclusion of the green activities can open opportunities for new efficient practices, markets, and a better use of resources which can create an increase of revenues for any sort of organization.

People are the main element of societies, companies, and game development. Therefore, people are the most defining factor to either support or to not sustainability. ICT efficiency alone cannot produce sustainability but a combination of effiency and sufficient strategies (created by people) that properly repespect defined enviromental practices and friendly innovations can unleash the ICT potential to fully support sustainability.

REFERENCES

ACM, 2006. *ACM.* [Online]
Available at: http://computingcareers.acm.org/?page_id=12
[Accessed 14 July 2014].

Adams, W. M., 2006. *The Future of Sustainability: Re-thinking Environment and Development in the Twenty-first Century ,* Cambridge: s.n.

Albertao, F., 2004. *Sustainable Software Engineering,* Silicon Valley: Carnegie Mellon University.

Albertao, F., Jing, X. & Chunhua, T., 2010. *Measuring the Sustainability Performance of Software Projects,* Beijing: IBM Research Division.

Ampatzoglou, A. & Stamelos, I., 2010. Software engineering research for computer games: A systematic review. *Information and Software Technology,* Volume 52, pp. 889-901.

Andreopoulou, Z., 2012. Green Informatics: ICT for Green and Sustainability. *Agricultural Informatics,* III(2), pp. 1-8.

Azarian, I., 2013. *SEGUE Technologies.* [Online]
Available at: http://www.seguetech.com/blog/2013/07/30/key-phases-software-development-projects
[Accessed 18 July 2014].

Baba, Y. & Tschang, T., 2001. Product Development in Japanese TV game software: The case of an innovative game. *International Journal of Innovation Management,* V(4), pp. 487-515.

Bates, B., 2004. *Game Design.* Boston: Thomson Course Technology.

Berg, H. A., 2010. *The Computer Game Industry,* Trondheim: Norwegian University of Science and Technology.

Bilton, N., 2011. *The New York Times.* [Online]
Available at: http://bits.blogs.nytimes.com/2011/07/05/video-game-industry-continues-major-growth-gartner-says/?_php=true&_type=blogs&_r=0
[Accessed 10 July 2014].

Bronk, C., Lingamneni, A. & Palem, K., 2010. *Innovation for Sustainability in Information and Communication Technologies (ICT),* Houston: s.n.

Burback, R., 1998. *Standford University Info Lab.* [Online]
Available at: http://infolab.stanford.edu/~burback/watersluice/node4.html
[Accessed 18 July 2014].

Callele, D., Neufeld, E. & Schneider, K., 2005. *Requirements Engineering and the Creative Process in the Video Game Industry.* Saksatchewan, Canada, IEEE International Conference on Requirements Engineering.

Carnegie Mellon University, 2014. *CMMI Institute.* [Online]
Available at: http://whatis.cmmiinstitute.com/what-is-cmmi
[Accessed 17 July 2014].

Chang , T., 2010. Gaming Will Save Us All. *Communications of the ACM,* pp. 22-24.

Cheng, K., 2011. *Casual gaming,* Amsterdam: s.n.

Christen, M. & Schmidt, S., 2012. A Formal Framework for Conceptions of Sustainability – a Theoretical Contribution to the Discourse in Sustainable Development. *Sustainable Development,* XX(6), pp. 400-410.

Clearwater, D. A., 2001. What Defines Video Game Genre? Thinking about Genre Study after the Great Divide. *The Journal of the Canadian Game Studies Association,* pp. 29-49.

Clinton , K., 2010. *Gamasutra.* [Online]
Available at:
http://www.gamasutra.com/view/feature/132683/the_state_of_agile_in_the_game_.php?print=1
[Accessed 25 October 2014].
Dahl, A. L., 1995. *TOWARDS INDICATORS OF SUSTAINABILITY.* Wuppertal, United Nations Environment Programme .
Delforge, P. & Horowitz, N., 2014. *The Latest-Generation Video Game Consoles: How Much Energy Do They Waste When You're Not Playing?,* Washington: Natural resources Defense council.
EagerFish, 2010. *EagerFIsh.* [Online]
Available at: http://eagerfish.eu/international-gaming-and-mobile-app-club-in-estonia/
[Accessed 13 July 2014].
Edge Staff, 2007. *Edge.* [Online]
Available at: http://www.edge-online.com/features/30-defining-moments-gaming/
[Accessed 13 July 2014].
Entertainment Software Association, 2014. *Essential Facts About the Computer and Video Game Industry,* United States: Entertainment Software Association.
Erdmann, L., Hilty, L., Goodman, J. & Arnfalk , P., 2004. *The future impact of ICTs on environmental sustainability,* Spain: Institute for Prospective Technological Studies & European Commission Join Research Center.
European Games Developer Federation, 2010. *Information Society Policy Research Unit.* [Online]
Available at:
http://is.jrc.ec.europa.eu/pages/ISG/PREDICT/games/documents/2MBehrmannEGDFgeneralcontext.pdf
[Accessed 13 July 2014].
European Parliament & Council of the European Union, 2003. Directive 2002/95/EC of the European Parliament and of the Council of 27 January 2003 on the restriction of the use of certain hazardous substances in electrical and electronic equipment. *Official Journal L 037,* pp. 19-23.
Fink, A., 2013. *How to Conduct Surveys: A step-by-step Guide.* United States of America: SAGE Publications.
Flood, K., 2003. *GameDev.* [Online]
Available at: http://www.gamedev.net/page/resources/_/technical/general-programming/game-unified-process-r1940
[Accessed 28 July 2014].
Flynt, J. & Salem, O., 2005. *Software Engineering for Game Developers.* 1st ed. United States: Thomson Course Technology PTR.
FOLDOC, 1992. *FOLDOC: Free On-Line Dictionary of Computing.* [Online]
Available at: http://foldoc.org/platform
[Accessed 10 October 2014].
FOLDOC, 2014. *FOLDOC: Free On-Line Dictionary of Computing.* [Online]
Available at: http://foldoc.org/console
[Accessed 10 October 2014].
Forbes, 2014. *6 Predictions About The State of Digital Marketing In 2015.* [Online]
Available at: http://www.forbes.com/sites/sujanpatel/2014/11/05/6-predictions-about-the-

state-of-digital-marketing-in-2015/
[Accessed 9 February 2015].
Fujitsu, 2012. *GREEN IT: THE GLOBAL BENCHMARK,* s.l.: Fujitsu.
Furini, M., 2007. *MOBILE GAMES: WHAT TO EXPECT IN THE NEAR FUTURE.* Alessandria, Italy, GAMEON Conference.
Gartner, 2013. *Gartner Inc.*. [Online]
Available at: http://www.gartner.com/newsroom/id/2614915
[Accessed 2014 July 2013].
Gavalas, D. & Economou, D., 2011. Development Platforms for Mobile Applications: Status and Trends. *IEEE Software,* pp. 77-86.
Godoy, A., 2010. *Game-Scrum: An Approach to Agile Game Development.* Florianópolis, Proceedings of SBGames 2010.
Godoy, A. & Barbosa, E. F., 2010. *Game-Scrum: An Approach to Agile Game Development.* Sao Carlos, Computing Track, pp. 292-295.
Goldsmith, T. T., Grove, C. & Ray Mann, E., 1948. *Cathode-Ray Tube Amusement Device.* United States, Patent No. 2 455 992.
Green Software Engineering., 2014. *GreenSoft.* [Online]
Available at: http://www.green-software-engineering.de/en/reference-model.html
[Accessed 30 July 2014].
Greenpeace, 2008. *Playing Dirty: Analysis of hazardous chemicals and materials in games console components,* Amsterdam: Greenpeace.International.
Griss, M. L., 1993. Software reuse: From library to factory. *IBM Systems Journal, vol 32,* pp. 548-566.
Harris, J., 2008. *Green Computing and Green IT Best Practices on Regulations and Industry Initiatives, Virtualization, Power Management, Materials Recycling and Telecommuting.* London, UK: Emereo Publishing .
Hewlett-Packard, 2015. *Five Steps to Green Computing.* [Online]
Available at: http://h20426.www2.hp.com/program/carepack/pdf/5steps.pdf
[Accessed 03 02 2015].
Hilty, L., Lohmann, W. & Huang, E., 2011. Sustainability and ICT – An overview of the field. *notizie di POLITEIA,* XXVII(104), pp. 13-28.
IDATE Research, 2012. *IDATE.* [Online]
Available at:
http://www.idate.org/2009/pages/download.php?id=1598&idl=21&t=f_telech_etudes&fic=VideoGameWatch2013_Factsheet.pdf&repertoire=fiche_etude/M13201SRDB_WorldVideoGameMarket
[Accessed 13 July 2014].
IEEE, 2014. *SWEBOK V3.0,* New Jersey: IEEE.
Iivari, J., 1996. Why are CASE Tools Not Used?. *Communications of the ACM,* Volume 39, pp. 94-103.
ISO , 2004. *ISO.* [Online]
Available at: https://www.iso.org/obp/ui/#iso:std:iso-iec:15504:-1:ed-1:v1:en
[Accessed 17 July 2014].
ISO/IEC & IEEE Computer Society , 2014. *SEVOCAV.* [Online]
Available at: http://pascal.computer.org/sev_display/index.action
[Accessed 14 July 2014].

ISO, 2008. *ISO.* [Online]
Available at: http://www.iso.org/iso/home/standards/management-standards/iso_9000.htm
[Accessed 17 July 2014].
Jain, H., 2011. Green ICT organizational implementations and workplace relationships. In: B. Unhelkar, ed. *Handbook of Research in green ICT: Technical, Business and Social Perspectives.* Hershey, PA, USA: IGI Global, pp. 290-299.
Järvinen, A., 2008. *Games without Frontiers,* Tampere: University of Tampere.
Juul, J., 2010. *A casual revolution : reinventing video games and their players.* United States of America: The MIT Press.
Kasurinen, J., 2012. *Software Organizations and Test Process Development,* Lappeenranta: s.n.
Kasurinen, J. & Laine, R., 2012. *Games from the Viewpoint of Software Engineering,* Lappeenranta: s.n.
Kasurinen, J., Laine, . R. & Smolander, K., 2013. *How applicable is ISO/IEC 29110 in Game Software Development?,* Lappeenranta: s.n.
Kettunen, V., Kasurinen, J., Taipale, O. & Smolander, K., 2010. *A Study on Agility and Testing Processes in Software Organizations.* Trento, Italy, ISSTA'10, ACM .
Kitchenham, B. A. et al., 2002. Preliminary Guidelines for Empirical Research in Software Engineering. *IEEE TRANSACTIONS ON SOFTWARE ENGINEERING, VOL. 28, NO. 8,* pp. 721-734.
Kultima, A. & Alha, K., 2010. *Hopefully Everything I'm Doing Has to Do with Innovation.* Tampere, Games Innovations Conference.
Laine, R., 2012. *Video Game - From Software to Experience,* Lappeenranta: Lappeenranta University of Technology.
Langer, A. M., 2008. *Analysis and Design of Information Systems.* London: Springer-Verlag.
Lee, R., 2013. *Business Models and Strategies in the Video Game industry: an analysis of Activision-Blizzard and Electronic Arts,* Massachusetts : Massachusetts Institute of Technology.
Lösch, A., 2006. Combining Quantitative Methods and. *Libri, 2006, vol. 56 ,* pp. 133-144.
Magee, L. et al., 2013. Reframing social sustainability reporting: towards an engaged approach. *Environ Dev Sustain 15,* p. 225–243.
Manning , S., Boons , F., Von Hagen , O. & Reinecke , J., 2012. National Contexts Matter: The Co-Evolution of Sustainability Standards in Global Value Chains. *Ecological Economics,* pp. Vol. 83 197-209.
Marchand, A. & Hennig-Thurau, T., 2013. Value Creation in the Video Game Industry: Industry Economics, Consumer Benefits, and Research Opportunities. *Journal of Interactive Marketing ,* pp. vol 27, pp. 141 - 157.
Marchand, A. & Thorsten , H.-T., 2013. Value Creation in the Video Game Industry: Industry Economics, Consumer Benefits, and Research Opportunities. *Journal of Interactive Marketing,* pp. 141-157.
Microsoft, 2008. Green Computing. *The Architecture Journal,* Issue 18, pp. 1-41.
Murphy-Hill, E., Zimmermann, T. & Nagappan , N., 2014. *Cowboys, Ankle Sprains, and Keepers of Quality: How Is Video Game Development Different from Software Development?.* Hyderabad, India, ACM.
Murugesan , S., 2008. Harnessing Green IT: Principles and Practices. *IT Pro, IEEE Computer Society,* Volume January/February, pp. 24-33.

Musil, J., Schweda, A., Winkler, D. & Biffl, S., 2010. *A Survey on a State of the Practice in Video Game Development ,* Vienna, Austria: Institute of Software Technology and Interactive Systems .
Natural Resources Defense Council, 2008. *Improving the Energy Efficiency of Video Game Consoles,* New York: Natural Resources Defense Council.
Newzoo, 2013. *The Global Games Market: Key Facts & Insights On the Global Games Market 2012-2016,* Amsterdam: Newzoo.
Nokia, 2009. *Nokia Corporate.* [Online]
Available at:
http://wayback.archive.org/web/20090209232201/http://www.nokia.com/A4303014
[Accessed 13 July 2014].
Norway Ministry of Culture , 2008. *Ministry of Culture.* [Online]
Available at: http://www.regjeringen.no/en/dep/kud/documents/draft-resolutions-bills-and-white-papers/white-papers-/2007-2008/report-no-14-2007-2008-to-the-storting/3.html?id=518817
[Accessed 12 July 2014].
Norway Ministry of Culture, 2007. *Norway Ministry of Culture.* [Online]
Available at: http://www.regjeringen.no/en/dep/kud/documents/draft-resolutions-bills-and-white-papers/white-papers-/2007-2008/report-no-14-2007-2008-to-the-storting/3.html?id=518817
[Accessed 12 July 2014].
Penzenstadler, B., 2013. *Towards a Definition of Sustainability in and for Software Engineering,* München: s.n.
Penzenstadler, B., Bauer, V., Calero, C. & Franch, X., 2012. *Sustainability in Software Engineering: A Systematic Literature Review,* München, La Mancha, Catalunya: s.n.
Petrillo, F. & Pimenta, M., 2010. *Is agility out there?: agile practices in game development.* New York, ACM, pp. 9-15.
Petrillo, F., Pimenta, M., Trindade, F. & Dietrich, C., 2008. *Houston, we have a problem...: a survey of actual problems in computer games development.* New York, ACM, pp. 707 - 711.
Petrillo, F., Pimienta, M., Trindade, M. & Dietrich, C., 2009. What Went Wrong? A Survey of Problems in Game Development. *ACM Computers in Entertainment,* pp. Vol. 7, No. 1, Article 13.
Plepys, A., 2002. The grey side of ICT. *Enviromental Impact Assesment Review,* Issue 22, pp. 509-523.
PricewaterhouseCoopers, 2012. *Global entertainment and media outlook: 2012–2016,* s.l.: PricewaterhouseCoopers (PwC).
PricewaterhouseCoopers, 2013. *Game changer: A new kind of value chain for entertainment and media companies,* United States: PricewaterhouseCoopers US.
Redavid, C. & Adil , F., 2011. *An Overview of Game Testing Techniques,* Västerås: s.n.
Schultz, C. P., Bryant, R. & Langdell, T., 2005. *Game Testing all in One.* United States of America: Thomson Couse Technology.
SE4S, 2014. *SE4S: Software Engineering for Sustainability.* [Online]
Available at: http://se4s.ics.uci.edu/
[Accessed 19 July 2014].
Sicart, M., 2007. *IT University of Copenhagen.* [Online]
Available at: http://www.itu.dk/courses/MSU/F2007/sicart9feb.pdf
[Accessed 28 July 2014].

Sicart, M., 2007. *IT University of Copenhagen.* [Online]
Available at: http://www.itu.dk/courses/MSU/F2007/sicart9feb.pdf
[Accessed 28 July 2014].
Soh, J. & Tan, B., 2008. Mobile Gaming. *Communications of the ACM,* 51(3), pp. 35-39.
Strandén, J.-P., 2012. *Technical framework in game development startups,* Lappeenranta: Lappeenrannan teknillinen yliopisto.
Ted , S., 2014. *The History Of Computing Project.* [Online]
Available at: http://www.thocp.net/software/games/games.htm
[Accessed 13 July 2014].
The Computer Language Company Inc, 2014. *PC Magazine.* [Online]
Available at: http://www.pcmag.com/encyclopedia/term/53848/video-game-console
[Accessed 10 July 2014].
Tschang, T., 2005. Videogames As Interactive Experiential Products And Their Manner Of Development. *International Journal of Innovation Management,* IX(01), pp. 103-131.
Tyni, H., Sotamaa, O. & Toivonen, S., 2011. *Howdy Pardner!: On free-to-play, sociability and rhythm design inFrontierVille,* Tampere: s.n.
Unhelkar, B., 2011. *Green IT Strategies and Applications.* United States of America: Taylor & Francis Group.
United Nations, 2005. *2005 World Summit Outcome ,* s.l.: s.n.
University of Southern California, 2013. *USC Libraries.* [Online]
Available at: http://libguides.usc.edu/content.php?pid=83009&sid=615867
[Accessed 15 July 2014].
Vanhala, E. & Kasurinen, J., 2014. *The role of business model and its elements in computer game start-ups.* Switzerland, Springer International Publishing.
VGChartz, 2014. *Video Games, Charts, Articles, News, Reviews, Community, Forums at The VGChartz Network.* [Online]
Available at: http://www.vgchartz.com/
[Accessed 14 July 2014].
Wikia, 2014. *Video Game Sales Wiki.* [Online]
Available at: http://vgsales.wikia.com/wiki/Video_game_industry
[Accessed 10 July 2014].
Winter, D., 1996. *PONG-Story.* [Online]
Available at: http://www.pong-story.com/intro.htm
[Accessed 13 July 2014].
Zapico, J., 2012. ICT and Environmental Sustainability, Friend or Foe?. *Information Technologies & International Development,* 8(3), pp. 99-101.
ZTE Coporation, 2014. *ICT Development Trends (2014): Embracing the Era of Mobile-ICT,* s.l.: ZTE.

APPENDIX 1.

Survey on Game Platform Migration, Environmental and Economic Aspects

Survey on Game Platform Migration, Enviromental and Economic Aspects

This survey is conducted by Lappeenranta University of Technology with PERCCOM graduate school. All collected data will be handled and stored confidentially, and no data will be released for the use of third party, with the exception of anonymized survey results as research results. From these published results, it will be impossible to deduct any information on names, products or answers from any individual responder.

More information available from Dr. Jussi Kasurinen (jussi.kasurinen@lut.fi).

Basic Information

1. Please indicate your current role in your organization: *

- ○ Developer or Tester, or other technically-oriented employee position
- ○ 3D-Artist, Musician, Graphics designer, or other artistically-oriented employee position
- ○ Project manager, lead designer or other project-level management position
- ○ Upper management, organization-level management
- ○ Marketing, administration or other position

2. In general, for how long have your company been involved in game development? *

- ○ Less than 2 years
- ○ 2 – 5 years
- ○ 6 – 10 years
- ○ More than 10 years

3. What are the main platforms you develop your games for? (Please select maximum of two platforms.) *

- ☐ Mobile platforms (smartphones, tablets)
- ☐ Desktop (PC, Mac, Linux)
- ☐ Browser games
- ☐ Game consoles (PS3/4, Xbox 360/One, Wii/WiiU...)
- ☐ Handheld consoles (PSP, DS...)
- ☐ Other:

APPENDIX 1. (continue)

4. Does your organization develop hardware components (controllers, consoles, devices, sensors, others) as a part of your game product? *

- ○ Yes
- ○ No

Green Aspects and Marketing

5. Based on your experiences or observations, rate the importance of eco-impact factors (such as reusability of components, energy efficiency, long term support, social impact) in a game development project: (1-5 scale, 5: Very important, 1: Not relevant)

○ 1 ○ 2 ○ 3 ○ 4 ○ 5

6. Does your organization use promotional items (toys, trinkets, clothes, posters, stickers, others) in the marketing work of your product? *

- ○ Yes
- ○ No

7. If you answered yes to the last question, please estimate what types of products you annually give away as promotion, and approximate on amounts (hundreds, thousands, tens of thousands).

Example: Toys: around 500; Keychains: ~800: Posters: around 15000; Demo CDs: around 2000.

8. How does your work involve the following concepts or activities? (1-5, 1: No involvement, 5: Focus area)

	1	2	3	4	5
Code reusability	○	○	○	○	○
Energy-efficient programing	○	○	○	○	○
Social awarness impact	○	○	○	○	○
Reusing old hardware components	○	○	○	○	○
Repurposing old hardware components	○	○	○	○	○
Reusing existing marketing/promotional material	○	○	○	○	○
Repurposing existing marketing/promotional material	○	○	○	○	○
Support to the legacy (older model) devices	○	○	○	○	○

Game Business and Product Design

APPENDIX 1. (continues)

9. Please rate the following aspects in your organization according to your own experiences:

	Close to this	Leaning towards this	Neutral	Leaning towards this	Close to this	
Game product design is driven by economic factors	○	○	○	○	○	Game product design is driven by creative factors
Design relies on quickly built prototypes	○	○	○	○	○	Design relies on systematic development
Game design concept are based on individuals	○	○	○	○	○	Game design concept are based on group effort
Design is ad-hoc process	○	○	○	○	○	Design has a formal process
External sources (publishers, funders) can dictate our designs	○	○	○	○	○	External sources have no power to affect designs

10. How important are the following roles for your organization? (1 = not important, 5 = very important)

	1	2	3	4	5
Developers, who do programming	○	○	○	○	○
Artistist, who develop graphics	○	○	○	○	○
Designers, who design content	○	○	○	○	○
Business, who manage business aspects	○	○	○	○	○
Musicians, who do sound work	○	○	○	○	○
Testers, who test products	○	○	○	○	○
Marketers, who manage marketing aspects	○	○	○	○	○
Field specialists, who have domain experience	○	○	○	○	○
Researchers, who work with new technology	○	○	○	○	○
Administration, who manage day-to-day activities	○	○	○	○	○
Hardware specialists, who have technological experience	○	○	○	○	○

11. How important are the following revenue sources for your organization? (1 = not important, 5 = very important)

	1	2	3	4	5
Funding organizations	○	○	○	○	○
Venture capitalists	○	○	○	○	○
Own savings	○	○	○	○	○
Daily job	○	○	○	○	○
Partners	○	○	○	○	○
Selling games	○	○	○	○	○
In-app purchasing	○	○	○	○	○
Advertisements in games	○	○	○	○	○
Outsourcing work for someone else	○	○	○	○	○

12. In your opinion, please select the three most important business aspects for game organization:

APPENDIX 1. (continues)

- [ ] Customer relationship
- [ ] Customer segment
- [ ] Financing
- [ ] Human capital
- [ ] Innovation and creative process
- [ ] Key activities (e.g. programming, designing, etc.)
- [ ] Key partners (e.g. publisher, musician, etc.)
- [ ] Key resources (e.g. office, computers, etc.)
- [ ] Marketing

13. Please select the three most important skills for a person who you would hire to work for your organization:

- [ ] Communication skills
- [ ] Ability to learn while doing
- [ ] Experience
- [ ] Understanding of personal talents
- [ ] Independent worker
- [ ] No need to closely administer
- [ ] Team working skills
- [ ] Outspoken
- [ ] Intelligent
- [ ] Trustworthy
- [ ] Artistic talent
- [ ] Ability to create production-quality content
- [ ] Good business sense
- [ ] Management skills
- [ ] Understanding of hardware
- [ ] Understanding of scientific improvements

Migration Process and Development Work

14. Does your organization follow some software development methodology such as Scrum (even partial or unofficial following is counted)?

- ○ Yes, please specify: [ ]
- ○ No

APPENDIX 1. (continues)

15. How closely these definitions fit to your organization's way of developing games (0-5 scale, 5 very closely, 0 very badly):

	1	2	3	4	5
Focused on processes improvement (We collect data to learn from our earlier project experiences)	○	○	○	○	○
Processes measured and controlled (We try to ensure that we get get what we want by measuring progress.)	○	○	○	○	○
Processes documented, standardized and proactive (We (or our client) want to know what has happened, so we document things.)	○	○	○	○	○
Processes are often reactive (We only react to problems that actually happen, and only prepare for the most probable problems.)	○	○	○	○	○
Processes reactive and often unpredictable (Since there may be large changes during the process, we focus on daily problems.)	○	○	○	○	○
Our product design does not change much during the development, after the initial product design is completed.	○	○	○	○	○

16. Please rate on average, how much time you spend in your typical project with the different project phases:

Add to the scale so that the total is 100 %

	0-10%	11-20%	21-30%	31-40%	41-50%	51-60%	61-70%	71-80%	81-90%	91-100%
Concept work, pitching	○	○	○	○	○	○	○	○	○	○
Prototype development	○	○	○	○	○	○	○	○	○	○
Design work	○	○	○	○	○	○	○	○	○	○
Programming	○	○	○	○	○	○	○	○	○	○
Level creation	○	○	○	○	○	○	○	○	○	○
Art/Audio Production	○	○	○	○	○	○	○	○	○	○
Testing	○	○	○	○	○	○	○	○	○	○
Alpha/Beta	○	○	○	○	○	○	○	○	○	○
Gold master/Crunch Time	○	○	○	○	○	○	○	○	○	○

17. Has your organization released a game, which was later ported to another platform or console generation?

If your answer is "No", you can skip the rest of the questions.

○ Yes

○ No

APPENDIX 1. (continues)

18. Based on your experience, how long does a migration process for your typical product take from one platform to another?

- ○ Less than two months
- ○ Less than six months
- ○ Less than one year
- ○ Less than one and half years
- ○ More than one and half years

19. How well do you consider the following items to represent migration process from one platform to another (1-5, 5 very closely, 1 very badly):

	1	2	3	4	5
The final costs were what we expected or estimated them to be.	○	○	○	○	○
The actual time the process took was what we expected or estimated it to be.	○	○	○	○	○
Our migration process was informal and/or agile.	○	○	○	○	○
We were able to keep metrics on the migration progress.	○	○	○	○	○
The migration process was planned in detail before implementation.	○	○	○	○	○
There weren't any – or only few – things that our organization had to learn before or during the migration work.	○	○	○	○	○
The migration process did not take much time to complete.	○	○	○	○	○
In general, the migration process did not have any or only a few unexpected issues.	○	○	○	○	○

20. When thinking about the game migration process and all the activities involved, what percent represents the importance of the following factors in the decision to migrate a game to a new platform?

Please add percentages so that the total is 100%.

General effort required (total working hours)	
Amount of coding work needed	
Amount of new (or redone) content needed	
Time needed	
Costs involved	
The amount of potential customers in the new platform	
The market share expected for the new platform	
The predictions for the market shares of the current platforms	
Eco-Impact	

Summa on yhtä kuin 0