ABSTRACT
Context: Software projects often have four objectives; to produce the required functionality, in budget and in schedule, with acceptable quality. That statement may be true for most of the ordinary software development projects, but are these objectives enough for game development, where creativity and artistic aspects have a major role? Goal: We analyze how game developing organizations test their products, what are their main test objectives and how they perceive themselves in the software business. Method: We interviewed seven game development teams from different companies and studied how they test their products with grounded theory approach. Results: Our results suggest that game developers focus on soft values such as game content or user experience, instead of more traditional objectives such as reliability or efficiency. Conclusions: Game developers have similar, but not fully comparable to software industry, set of priorities in their software testing and quality assurance approaches.

Categories and Subject Descriptors
D.2.9 [Software Engineering]: Management – Software quality assurance (SQA)

General Terms
Management, Human Factors, Verification.

Keywords
Test process, Game development, Test methods, Empirical study

1. INTRODUCTION
The general objectives of software projects are to produce software, which fulfills the requirements, has acceptable quality, is completed within budget and is released in time [1]. In software testing, the general objective is in verification and validation of the developed product; testing if the product is built as designed (verification), and that the product fulfills the purpose it was created for (validation) [2].

Fundamentally, video game development is a form of software development, although with several added requirements like visual presentation, artistic aspects and creative design [3, 4]. In this sense, game developing organizations should be able to apply mostly the same principles as organizations in other areas of software engineering. However, this is not necessarily true, as there are studies in the game development context that challenge the applicability of software engineering principles. For example, on design methods [5], technical frameworks and development techniques [3], and even on the required technical skills [3, 6] there are studies which discuss differences between games and software.

Testing processes in the software business can suffer from various problems, such as resource shortages, limited cooperation with the customers, lack of management involvement or even simply from the lack of experienced testing personnel [1, 2, 7, 8]. These problems can be addressed by focusing the testing work on certain aspects [9], such as security, stability or correctness. But how do the additional features of game products, such as artistic aspects, sound design or simply testing that the potential users like emotionally the developed game affect the test process and test plans?

In this paper we study how game developing organizations test their products, what their main test methods are and what objectives they try to achieve with the test process. Although game development is also software development, it has several unusual features such as the drive for novelty factors, creativity and artistic impression. In this sense it differs from the conventional view to software engineering (for example as defined in ISO/IEC 12207-2008 [10]), where measurable quality aspects, such as performance, reliability and security are valued. From this difference follows that a purist approach of verifying and validating functionality and infrastructure may not be sufficient. Motivated by this understanding, our study focuses on the questions How game developing organizations test their products? and Does game testing differ from the testing of conventional software products?.

This study is also a continuation study of our earlier empirical studies in software testing and game development. In these studies, our research group has made similar observations in software organizations to understand how they develop their test plans [11] or what quality aspects they emphasize in their test processes [12]. On game development, our earlier work has discussed, for example, how software engineering process models fit to the game developing organizations [13] or what types of software infrastructure the game developers usually apply [14].

The rest of the paper is constructed as follows; Section 2 provides an overview on the related research in game development from the viewpoint of software engineering and testing. Section 3 presents the data collection and analysis method, while Section 4 presents
the results from the data analysis, main findings and their implications. Section 5 discusses these results and the potential threats to the study validity and finally, Section 6 closes the paper with conclusions.

2. RELATED RESEARCH
Testing work is always focused on certain aspects of the software product, as complete testing is not only very expensive but also practically impossible in all real life cases [8]. In practice most organizations select and prioritize their test cases based on the preferred quality or at least to test the most critical components of the software system [9]. Organizations may simply lack resources for testing. They may have as low as fifteen percent of the testing resources they need and still produce acceptable end-product quality, which also means that the lack of resources may go unnoticed by the upper management [15]. There are several reasons for difficulties in the test process. They may include lack of management support, lack of user involvement or lack of any resource, such as time or money [7].

In game development, software development work is a major activity, as video games consist of software. However, in addition to software development, the game industry has also other concerns. For example Peltoniemi [16] has analyzed game industry and compared it to other areas of software development. He concluded that in general game software development and their processes are not very mature. Peltoniemi also argued that game development differs from the conventional software industry in its relatively stable hardware platform and the extensive demand for creativity. Because of these differences, the established methods and practices from the software engineering discipline may not be directly applicable to game development.

However, Kultima & Alha [17] have also observed game industry from the viewpoint of creativity and process management, and concluded that the game industry has been moving from the “creative chaos” and ad-hoc practices towards process thinking. In some cases the hallmarks of game development work – innovation and creativity – have been reduced to publicity acts; in larger organizations market-oriented thinking and brand management are the leading factors. Innovation and creativity are still required in the development of new products, but the game industry as a whole is becoming more organized. Defined and constrained processes are taking place over “chaotic” approaches.

The application of software engineering practices in the design phases of game development has been discussed by Kultima [5]. This study found that defined practices often resulted to unsatisfactory results or did not work properly. Similarly, Blow [3] has expressed the concern that the current software development techniques do not support the game developing community very well. For example, the development suites created for software development do not support well game-specific features and game developers use them only because nothing better is available.

Based on these concerns, it seems that game developing organizations resemble normal software organizations, but they do not necessarily apply the same approach in all of their activities. The game development industry has its own priorities, which sometimes differ from the traditional software engineering industry [6]. For example, Kanode and Haddad [4] list audio engineering, multimedia asset management and game play design as areas that the conventional software development does not take into account. In addition, Callele et al. [18] also include user interactivity and artistic aspects to the list of game development-specific considerations.

It is apparent that the game industry has practices and considerations that differ from the rest of the software industry. For example, the emphasis on game play design and user interactivity are important to games but not so much to conventional software products. This warrants a question how do these aspects affect the test process? Software testing in general has been studied widely, but similarly as in game design [5], systematic testing techniques may prove unsatisfactory or miss the actual objectives when testing games. On the other hand, game developers may use software industry-based testing practices simply as there are no better alternatives, similarly as observed by Blow [3] in other development techniques.

3. RESEARCH METHOD
Software development, including the design, development and testing of a commercial product is a complex phenomenon, with several stakeholders and various methods for accomplishing similar objectives. Acknowledging this we decided to apply empirical qualitative analysis, and use the grounded theory method [19, 21,22] to observe and analyze the testing methods and testing objectives of game developing organizations.

The Grounded Theory approach was first defined by Glaser and Strauss [20]. Later this approach has divided to two distinctive schools, defined by Barney Glaser (Glaserian method) [19] which promotes passive observation and non-intrusive data gathering, and by Strauss and Corbin (Straussian method) [21], which promote a systematic data collection and codification approach [20]. Due to the relatively large number of observed organizations for a qualitative study, we decided to apply the Strauss-Corbin approach. In theory building we followed the principles and guidelines derived from studies by Eisenhardt [23], Paré and Elam [24] and Klein and Myers [25].

3.1 Data collection
The initial population and population criteria were based on theoretical sampling, where the objective was to gather different types of game developing organizations to the study population. The population ranges from small startup companies making their first products to large and medium-sized established game companies with extensive product libraries. In the largest organization, several hundred people contributed to the product development by directly working in the organization or by producing content via partnership, whereas the smallest organization was a startup that employed only three people.

All study organizations were professional software producers, who made games as their main activity and source of income. The organizations varied (Table 1) from newly started mobile game developers to developers of browser-based, PC and console games. Their distribution channels included platform-specific internet stores, general digital distribution, and physical boxed copies sold at game stores. One organization also developed browser-based games with their own distribution site and store. All interviewed organizations were located in South-Eastern Finland, and their sizes varied from very small to medium-sized companies [26].

We carried out 27 interview sessions with game development professionals in four interview rounds (Table 2). In one
organization we were able to arrange only three interview sessions due to schedule conflicts, but in this organization the round two related topics were discussed with the interviewees of the latter rounds. In another organization we were able to arrange only one interview before they had to withdraw from the study; this organization and their interview data was discarded. The final sample of the interview rounds consisted of seven organizations selected from our research partners and supplemented with volunteering organizations by researchers to achieve a heterogeneous group of different target audiences, development platforms and organizational backgrounds. The interviews were collected during the spring, summer and fall of 2012 by seven researchers from two research laboratories. The interviews were mostly in Finnish, to encourage discussion and comments from the interviewees over the interview themes and topics. For this publication we translated the relevant quotations to English when necessary.

The objective of the interviews was to gain an insight into the game organizations and understand what game developers considered their most important test objectives, how these organizations tested their games and, if possible, identify the differences between the conventional software testing work and game testing. In addition to the testing-specific objectives, the interviews also collected data on other software engineering-related topics, such as quality aspects, development and technical frameworks. The list of interview themes included questions related to development methods, test processes, quality requirements, test phases, applied tools, outsourcing and design principles. This enabled us to form the whole picture of the case organizations. The complete theme-based question lists are available at http://www2.it.lut.fi/project/SOCES/. A reference list of the themes of the data collection rounds is also presented in Table 2.

The interviews contained semi-structured questions, and the sessions were tape-recorded for later qualitative analysis. An interview lasted in average about one hour. The interviews were arranged face-to-face with one or two organization participants and one or two researchers present.

The first round interviewees were project managers, who were either responsible for the entire development process of one product, or one development phase for all of the products. The decision to interview project managers during the first round was based on the objective to understand the organizational processes and to gain understanding of the operational level of software development in the case organizations. Another reason was that

<table>
<thead>
<tr>
<th>Organization</th>
<th>Release platforms</th>
<th>Production team size¹</th>
<th>Intended distribution channels for products</th>
<th>Number of released games</th>
<th>Age of the organization</th>
</tr>
</thead>
<tbody>
<tr>
<td>Case A</td>
<td>PC², game consoles²</td>
<td>Large</td>
<td>Digital distribution, boxed copies</td>
<td>More than 10 released products</td>
<td>More than 5 years</td>
</tr>
<tr>
<td>Case B</td>
<td>Mobile platforms</td>
<td>Small</td>
<td>Platform store</td>
<td>Less than 5 released products</td>
<td>Less than 3 years</td>
</tr>
<tr>
<td>Case C</td>
<td>Game consoles², PC²</td>
<td>Large</td>
<td>Boxed copies, digital distribution</td>
<td>More than 10 released products</td>
<td>More than 5 years</td>
</tr>
<tr>
<td>Case D</td>
<td>Mobile platforms, PC²</td>
<td>Medium</td>
<td>Platform store, digital distribution</td>
<td>Less than 5 released products</td>
<td>Less than 3 years</td>
</tr>
<tr>
<td>Case E</td>
<td>Mobile platforms</td>
<td>Small</td>
<td>Platform store</td>
<td>Less than 5 released products</td>
<td>Less than 3 years</td>
</tr>
<tr>
<td>Case F</td>
<td>PC²</td>
<td>Medium</td>
<td>Digital distribution</td>
<td>Developing first commercial product</td>
<td>Less than year</td>
</tr>
<tr>
<td>Case G</td>
<td>Browser games</td>
<td>Small</td>
<td>Separate service site</td>
<td>Developing first commercial product</td>
<td>Less than year</td>
</tr>
</tbody>
</table>

¹Number of people contributing working hours to the released product, size by SME definitions [26]
²Releases for this platform usually also include publishing contract or funding partners.

<table>
<thead>
<tr>
<th>Interviews</th>
<th>Interviewee role</th>
<th>Description</th>
<th>Main themes of the interviews</th>
</tr>
</thead>
<tbody>
<tr>
<td>Qualitative interview with 7 organizations</td>
<td>Team leader or project manager</td>
<td>The interviewee is responsible for management of development of one product, or one phase for all products.</td>
<td>Development process, test process, quality, outsourcing, development tools, organizational aspects.</td>
</tr>
<tr>
<td>Qualitative interview with 6+1* organizations</td>
<td>Developer or tester</td>
<td>The interviewee was responsible for development tasks, preferably also with responsibilities in the testing activities.</td>
<td>Development process, test process, development tools, development methods, quality.</td>
</tr>
<tr>
<td>Qualitative interview with 7 organizations</td>
<td>Upper management or owner</td>
<td>The interviewee was from the upper management, or business owner with an active role in the organization.</td>
<td>Organization, quality, marketing, innovation and design process, development process.</td>
</tr>
<tr>
<td>Qualitative interview with 7 organizations</td>
<td>Lead designer or Art designer</td>
<td>The interviewee was a game designer, or managerial level person with the ability to affect the product design.</td>
<td>Development process, design and innovation, testing, quality</td>
</tr>
</tbody>
</table>

*topics discussed at later rounds with the other representatives
we wanted to test whether our earlier observations and experiences from the general software industry [11, 12] would be applicable in the game industry context.

The interviewees in the second round were developers or programmers that directly contributed to the game software product and had experience with the technical details of the developed product and, if possible, had substantial testing-related tasks. The second round interview topics were heavily focused towards programming techniques, process activities, testing work and applied development tools.

In the third round, the focus of the interviews was to collect more general data on the company beyond the software work and focus on the managerial and business aspects. During this round additional themes such as marketing, innovation and financing were collected to better understand the context in which the game industry operates and how game business in general works.

In the fourth round, the focus was on the creative aspects of the game development. During this round the interviewed employees were game designers or management-level personnel with the ability to affect the final designs of the product by deciding which features were added or dropped from the final version. In addition to design, interview topics in this round discussed change management, design-phase development activities and external influences.

3.2 Data Analysis
The grounded theory method contains three data analysis steps: open coding, where categories and their related codes are extracted from the data; axial coding, where connections between the categories and codes are identified; and selective coding, where the core category is identified and described [21].

The objective of the open coding is to identify leads in the data and classify the observations. This process was started with “seed categories” [27] such as “process problems”, “development process details”, “test methods”, that contained essential and known phenomena based on the prior knowledge and experiences. In our case, the seed categories were derived from our prior studies on general software industry and from known issues identified in the previous studies, and also used in the main themes of the data collection questionnaires.

In open coding, the classified, codified, observations are organized into categories. At the end of the open coding, the number of codes was 172 codes in 14 categories, identifying 1572 observations from over 1400 minutes of recorded interview data. During coding, new categories appear and merge because of new information that surfaces during the analysis of the interview data by constant comparison between the observations and existing codification. For example, our initial concept of collecting data on applied standards as a separate category was revised as the interview data proved that the organizations in general were not as process-oriented or highly organized as originally thought. This led to merge between different categories such as quality requirements and test process planning. Additionally, some concepts like industry-specific development activities were given their own categories.

The objective of the axial coding, which starts when the categories start to emerge and runs somewhat parallel with the open coding [21], is to further develop the categories by looking for causal conditions or any kind of connections between the categories. To advance to this phase, the categories and their related observations have to become stable enough to allow transition of focus from individual observations to developing the relationships between larger concepts. For example, codes such as “Design process: First design compared to final product”, “Development process: Quality objectives” and “Test process: Decision makers” form a chain of evidence for observing the organizational design practices and the way the product design matures during the development process. The objective of axial coding is to identify these connections, and compare these inter-category relationships in different organizational contexts.

The third phase of grounded analysis, selective coding, is used to identify the most essential phenomenon, the core category [21] and relate it to the surrounding categories. As based on [21], the core category is sometimes one of the existing categories, and at other times no single category is broad or influential enough to cover the central phenomenon. In this study, the core category Main test objectives in game development resulted from summarizing the analysis categories that discussed the official test objectives of the organizations, what sort of testing activities the organization actually did and what sort of testing the interviewees considered valuable. We categorized these testing objectives to three different focus areas, which were named game mechanics, technical aspects and user experience and observed that the most important items were more related to user experience and usability than to the usual software quality aspects, such as reliability or functionality. To further elaborate on this topic, we included to the study several other categories that are related to the differences between game and general software development. For software developers, the correctness of the developed system is more important, whereas game developers view software work more as a means to expression. We adjusted the study to include issues that explained testing activities and especially how the game developers decided the scope of their testing. All observations divided to categories are summarized in Table 3.

4. RESULTS
In this section we introduce the categories created in the analysis, present the main findings we made on the testing work in game development and discuss the implications of the findings.

4.1 Categories
The core category, Main test objectives in game development, describes the main testing objectives in the interviewed organizations. We identified three areas of objectives. User experience denotes items such as fun factor, user impressions or satisfaction over using the game product. Technical aspects denote the focus on items such as functionality, stability or any other technical quality aspect of the product. Game mechanics denotes the focus on game contents, such as game rules, the balance between different features and mechanics, and overall content design.

The category Used test methods describes the most important methods the organization uses to test their products. In most organizations the test methods focused on usability testing, where easiness of use and learning were tested, or on explorative testing, where testing was conducted simply by using the software. In three organizations there were also separate pre-designed test sets. In four organizations programmers did also extensive unit testing activities during the development.
Table 3: Findings and related categories from the case organizations

<table>
<thead>
<tr>
<th>Case A</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
<th>F</th>
<th>G</th>
</tr>
</thead>
<tbody>
<tr>
<td>Main testing objective in game development</td>
<td>User experience, technical aspects</td>
<td>User experience, technical aspects</td>
<td>Technical aspects, game mechanics, user experience</td>
<td>Game mechanics, user experience, technical aspects</td>
<td>Technical aspects, user experience</td>
<td>Game mechanics, technical aspects</td>
</tr>
<tr>
<td>Used test methods</td>
<td>Usability testing, tasked tests, explorative testing, external testing</td>
<td>Unit testing, integration testing, usability testing</td>
<td>Usability testing, tasked tests, explorative testing, external testing</td>
<td>Usability, unit testing, explorative testing, external testing</td>
<td>Usability testing, explorative testing, market survey on game characteristics</td>
<td>Tasked tests, unit testing, Unit testing, explorative testing</td>
</tr>
<tr>
<td>Test plan for the product</td>
<td>Pre-defined by QA lead</td>
<td>Lead designer - made decisions</td>
<td>Pre-defined by QA lead</td>
<td>Management - made decisions</td>
<td>Team-made decisions</td>
<td>Pre-defined by management</td>
</tr>
<tr>
<td>Development approach</td>
<td>Defined agile, SCRUM</td>
<td>Plan-driven</td>
<td>Defined agile, SCRUM</td>
<td>Undefined agile</td>
<td>Undefined agile</td>
<td>Defined agile, SCRUM</td>
</tr>
<tr>
<td>Test process effect on product design</td>
<td>Large, able to affect features</td>
<td>Medium, able to affect features</td>
<td>Large, may cause major changes</td>
<td>Small, some changes possible</td>
<td>Large, may cause major changes</td>
<td>Large, may cause major changes</td>
</tr>
<tr>
<td>Amount of testing resources</td>
<td>Average, 50%</td>
<td>Good, 85%</td>
<td>Average, 50%</td>
<td>Low, 75%</td>
<td>Good, 70%</td>
<td>First product</td>
</tr>
<tr>
<td>Type of product</td>
<td>PC, game consoles</td>
<td>Mobile devices</td>
<td>Game consoles, mobile</td>
<td>Mobile devices, PC</td>
<td>Mobile devices</td>
<td>PC, mobile devices</td>
</tr>
<tr>
<td>Product Size</td>
<td>Large</td>
<td>Small</td>
<td>Large</td>
<td>Medium</td>
<td>Small</td>
<td>Medium</td>
</tr>
<tr>
<td>Organizational background</td>
<td>Media</td>
<td>Academy</td>
<td>Media</td>
<td>Software Industry</td>
<td>Academy</td>
<td>Software Industry</td>
</tr>
<tr>
<td>Creative work vs. Software work</td>
<td>Creative</td>
<td>Creative</td>
<td>Leans to creative</td>
<td>Divided</td>
<td>Divided</td>
<td>Leans to creative</td>
</tr>
</tbody>
</table>

The category Test plan for the product defines the way the organizations make their test plan. In three organizations the test plan is made before the actual testing work starts by a QA leader or by management and updated if necessary. In the rest of the organizations the test plan is based on the decisions made during the test process made by the lead designer, management or the development team itself. The separate pre-defined test plan was used only in the large and in one of the medium-sized organizations. In three organizations there was a separate person making the testing-related decisions. In the rest of the organizations, the decision makers were either from upper management or communal decisions were made within the development team.

The category Development approach defines what kind of a development process the organization applies. The applied scale varies from undefined agile to plan-driven. In undefined agile the organization uses basically an open approach, with little documentation and planning. In defined agile the organization follows the principles of an established and defined agile practice, and in plan-driven the organization follows a more conventional design-first approach.

The category Test process effect on product design defines how much influence the test process has on the final design of the game product. The scale of the effect has the scale of three steps. In Small effect, the test results may change some minor details or small things in the software product to fix found problems. In Medium, test results may cause large changes to existing features, or product to drop or add minor features. In Large, test results affect every aspect of the game, and may even cause changes to the main features, addition of new main features, changes to the game theme, or even cause the development team to drop the entire game from development.

The category Amount of test resources defines how much testing resources the organization considers to have from the optimum they consider themselves to need. For example, if an organization has three testers and they consider themselves to need four, they...
have 75 percent of resources. In addition to personnel, things like time, money or availability of testing tools affect this self-defined optimum. In two organizations that were making their first commercial product, the interviewees considered themselves unable to assess their resources effectively or said that they are yet to acquire actual resources to conduct large-scale testing.

The category Type of product summarizes the typical target platform of the organization. For example, PC represents strict PC-targeted games, while game consoles denote that the organization develops games for some of the game console platforms. Mobile devices denote both smart phone and tablet systems, while browser games indicates that the game is offered as an applet or a service in the Internet, playable with a browser.

The category Product Size describes the size [26] of a typical game product developed by the organization. In this assessment, in addition to the people employed by the organization, also the producers of the bought content such as models, sound effects or game art are also included in the assessment.

In the category Organizational background describes the origin of the organization. Media indicates that most of the original members have background or education in TV, advertisement industry or print media. Academia indicates that the members are teachers, researchers or students of games or software-related academia. Software Industry indicates that the founding members used to be professional non-game software developers.

The last category Creative work vs. Software work indicates the point-of-view to the development – if the work is considered creative and artistic with software development tasks, or software development work with creative and artistic tasks. The applied scale was “Creative”, “Leans to creative”, “Divided”, “Leans to software work” and “Software work”.

4.2 Findings

4.2.1 Most game developers consider their work to be creative work, where software development is only a means of expression.

Many of the game developers do not consider themselves strictly as working in the software business. In all case organizations even developers doing programming and testing considered game development as creative work rather than normal software engineering. In fact, the most technically oriented people were project managers; two out of the seven managers considered game development to be as much software work as any other type of software development. In addition, in all organizations except in Case C, the game designers considered game products to be art and not just software constructs.

“Definitely creative work with mandatory programming tasks”...“The final package is much more important than some technical detail” - Case E developer

“Game development is creative work, where software development is only a miniscule slice of everything that has to be done.” - Case A, Project Manager

“I think it is creative work, and I'd venture a guess that our programmers also think so.” - Project manager, Case D

“Games are just interactive art, fully comparable to movies and music, just a more complicated combination.” - Case B Project manager

“Compared to the traditional software business, making games requires a vast amount of passion and motivation.” - Case F project manager

“It is creative work. The programming part perhaps is not the most creative part...” "...but creativity definitely is the first thing that comes to my mind." -Case C Developer

However, this consideration was not completely universal. In Cases C, D and E there were also interviewees that considered their work to be mostly software business. In their perspective the software development work was the actual requirement to create a game; without programming skills there would be no game product, but even without a competent artist you could create at least something. However, most of the technically minded also acknowledged that their opinion was in minority.

“Well, you can always mimic artistic styles, but you still have to do the actual programming work.” - Case E Project manager

“I think it is more software work than creative things, but I also know that people here disagree.” - Case C project manager

4.2.2 In game development, the test process has a larger influence on the product than in the conventional software development, as late changes are allowed and even expected.

Unlike in the software domain, where late changes to the product are generally considered detrimental to quality, in game development late changes are usually expected and allowed [2, 27]. All of the case organizations allowed at least some changes to their products based on the testing results. In Cases A, C, E, F and G changes based on the test results were often large revisions. Even major features could be changed based on the testing results. This was mostly caused by the decision to test everything with a test audience. In all organizations the features which the audience did not like would be cut, revised or redesigned.

“You can plan for a large number of things, but ultimately the final decision is made when actual people try out the idea.” - Designer, Case E

“It [testing work] does affect and it should affect [design].” - Designer, Case C

“I usually make a change to the game mechanics, ask people to test it, and tell me if they think it was for better or worse.” - Case G, Designer

“Testing really affects [the product features]” - Case A Designer

Even in Case D, where the interviewees considered it unlikely that major features would change after testing, testing affected the final product. In their organization constant testing was done to assure that the designed feature worked as intended.

“You have to constantly test things; even if something works perfectly in your head, the implementation may not work at all.” - Case D Developer

4.2.3 Test management is more ad hoc in game development than in conventional software development.

In all interviewed organizations, the number of predefined test cases was considered to be either zero or at most up to one third of the actual test cases. However, even if the organizations did not
have strict test plans, they did apply quality control measurements. For example, Cases A and C had quality assurance leaders for their products, ensuring that the product content achieved at least an acceptable level of quality. In Cases B, D and F similar decisions were made by the lead designer (Case B) or by the upper management (Cases D and F). In other organizations the quality control was based on the team-made decisions on what was acceptable to be included in the actual product.

"Our fundamental rule is that nothing that the QA leader has not signed off leaves the studio." - Case A, Project manager

However, even in Case A the largest and most influential testing activity was considered to be usability testing, with a test group formed from the target demographics and other volunteers.

"In the end most of our testing is just people using our product." - Case A, Project manager

This sentiment was present in most organizations. The applied test approaches relied heavily in the usability tests with users from the target demographic, not on the documented test plans or predefined test cases. Case E went even a bit further, adding a small market survey to their play test sessions.

"We do not have any systematic [testing], it’s mostly the latest version which we then give out for people to try out." - Case E, Project manager

"We have two people dedicated to testing during development"..."Otherwise, we employ [people] to do ad hoc testing" - Case D Project Manager

"We test what we feel like the best things to do, as long as it does not take too much time" - Project manager, Case B

"We do play tests with people and conduct surveys. We have a small questionnaire for them." - Case E, Developer

4.2.4 Game organizations do not focus on the same testing areas than conventional software organizations.

Game organizations seem to focus on “soft” testing areas, such as user experience, usability or game mechanics over more technical aspects such as stability, security or maintainability. In four case organizations (C, D, F and G) one of the main test objectives was in game mechanics, and in five organizations (A, B, C, D and E) user experience was mentioned. One explanation for the relatively low priority of technical aspects could be that game organizations tend to outsource their technical development, for example by buying a third party game engine. In the population, only Cases B and G used their own engine, while the rest of the cases used a commercially available product. However, this division did not affect the main objective of the testing process much; Cases B and G were still promoting user experience and balance over technical quality.

"The user experience has to be the best possible." - Case B, Designer

In game products the user experience is more important than in the conventional software development. Based on studies on test case selection principles [29-31], conventional software organizations focus on the implementation of the required features and their technical platform. In many software organizations pre-planned test cases are the only things that are tested and the explorative testing approach is not widely used. In game organizations the focus is on playability, usability, game rules and “fun factor”, not on the technical aspects of the product. The technical aspects are taken into account, but only to the degree of ensuring that the problems do not start to hurt the product and company reputation.

"Players are quite tolerant with technical stuff."..."Of course you should not have so many bugs that it hurts sales" - Case A, Project manager

"Our main focus on testing is to ensure that the game itself is in balance."..."Of course it also has to be fun" - Case D Project manager

Some interviewees had a more technology-oriented view. In cases C and F the interviewees also considered the technical functionality to be important, although the Case F interviewee added that technical functionality and usability are close to each other in testing priorities.

"Most important is that the game works correctly without problems in all of the platforms we are going to release it for" - Case C Project manager

“Same as with other software, you have to take out the critical problems while also ensuring the usability.” - Case F, Project manager

4.2.5 Game organizations have sufficient test resources even if the approach to testing does not have highly organized processes.

In our earlier survey [13] conducted with the general software industry, the average amount of test resources in software organizations was 70 percent (75 median) of their self-defined optimum. The same average was observed in our case organizations, with 50 percent of resources being the lowest assessed amount. It seems that even if some of the game developers were not very organized in their testing activities, they have at least an adequate amount of resources to do the testing work. Additionally, with the exception of Case D, the game developers considered themselves to have enough test resources to get everything done.

"I’d say we have something like 70 [percent] of the resources we planned for." -Case E developer

"We have not been doing very strong testing work"..."I think we have something around 75 [percent of test resources]" - Developer, Case D

["Around 90 percent availability?"]..."Yes, if we are doing traditional 2D on top of the mobile platform." - Developer, Case B

"We are somewhere around 50 percent.” – Developer, Case C

Based on these observations it is evident that game developers have at least adequate testing resources and that they have invested in the testing work. Even if the study average was the same as with conventional software developers, the difference in test resourcing was that all the interviewed game developers had invested in testing activities. In our earlier studies of conventional software developers [12, 14] there were organizations with the mindset “release now, fix problems later”. There were also cases with a tendency to leave to testing whatever resources there were
left after development. In this study, all the game developers had at least some priority in the testing-specific activities.

4.3 Implications
The common theme of the findings was that the game developing organizations did not consider themselves to be software developers but rather organizations using software systems as a means of expression. This was reflected especially when discussing of creativity, and it also affects the test process objectives and applied test methods. Even though all organizations mentioned technical aspects to be important for testing, most of the organizations promoted usability or explorative testing as their most important test method. Four cases (A, B, D and G) considered user experience and game mechanics more important than technical aspects. This indicates that game developers concentrate more on soft aspects, especially when compared to conventional software organizations [12]. Additionally, many of the smaller organizations (Cases B, D, E and G) did not use managed test plans; they relied heavily on usability and explorative testing (D, E, G) or on a game designer that was able to manage the test process to reach a satisfactory result (Case B).

Another major difference between the test process activities between software and game developers was in the attitudes towards late changes. Late changes in software projects are usually considered detrimental to the overall quality [8, 9]. In all the organizations the test process had at least some influence on the game contents if testing revealed problems or revision needs. In medium and large-sized Cases A, C and F, the changes could be large and they could even change major features of the game. In Case A this was especially interesting as the organization was very experienced in game development, was highly organized with SCRUM as their development approach, and still considered test-based late changes possible, even necessary for game development. Another interesting observation was that none of the interviewed organizations considered explorative or non-document-based test approaches an unfeasible or unproductive activity. In fact, most organizations (Cases B, D, E, G) did not use managed test plans; they relied heavily on usability and explorative testing (D, E, G) or on a game designer that was able to manage the test process to reach a satisfactory result (Case B). In Case A and C were also only organizations with experience on game console publications, and even if the console releases have rather rigid acceptance testing requirements, they still considered usability testing and user experience (especially Case A) more important than technical testing work.

In Case A the interviewees were also most confident that their work was more related to the creative and artistic aspects than software work. Based on the findings on the testing objectives and testing practices, testing in game developing organizations can be summarized as follows: Game developing organizations have in testing a strong emphasis on soft values such as usability or user experience to the degree that even major changes may occur late in the development based on the test results. Game developing organizations focus on testing activities, which involve the following aspects: actual playtests for testing the usability of the interface, testing the contents for fun factor and artistic presentation and user-based testing to examine the learning and difficulty curve of the game. Technical quality as understood in non-games software development is often present, but only as a secondary concern.

Based on these observations, it can be therefore argued that game projects have five objectives; produce required functionalities, in budget and in schedule, with acceptable quality, and the result has to appeal emotionally to the customers. In all interviewed organizations user experience and game mechanics were mentioned, and unlike in the earlier considerations with conventional software organizations [for example 1, 8, 12], things like fun factor or game mechanics were promoted over technical concepts like reliability, compatibility and efficiency. The summary of these findings is also listed in Table 4.

5. DISCUSSION
It seems that game developers have more focus on soft aspects of testing than on technical ones. However, almost all case organizations (Cases A, C, D, E and F) had outsourced their game engine development and currently apply a customized third party game engine in their products. The application of a third party engine does not mean that the development is not technically challenging. Even if the organizations acquired the game engine from a third party, they still did extensive and complex programming work to their products. When considering all the case organizations, the amount of technical work done within the development team does not correlate with the test objectives; even in the organizations that created their own game engines (Cases B and G) the testing focus was on the game mechanics and user experience over technical solutions. The different priorities of game development are also apparent when considering the mindset of the interviewees. Only three people out of the 28 interviewed professional developers considered game development to be primarily software work, with additional creative and artistic components, whereas twelve considered game development to be mostly creative work with software development only as a means for expression.

Game organizations do software testing similarly than conventional software organizations, although their priorities are different. The interviewed organizations did not seem to have any resource shortages for testing: none of the game developers reported less-than-adequate amount of resources. The amount of pre-planned testing was low in the observed organizations, but

<table>
<thead>
<tr>
<th>Table 4: Summary of the observations</th>
</tr>
</thead>
<tbody>
<tr>
<td>Observed testing behavior of game-developing organizations</td>
</tr>
<tr>
<td>-Focus on testing the product with end users, technical reviews a secondary concern.</td>
</tr>
<tr>
<td>-Focus on usability, learnability and user experience over technical functionality and technical quality.</td>
</tr>
<tr>
<td>-Testing is used to steer the product design work; fun factor one of the tested aspects along with technical requirements.</td>
</tr>
<tr>
<td>-Late changes expected, development process constructed to allow late changes to the product design.</td>
</tr>
<tr>
<td>-Test process has a higher importance than with traditional products, since most things need testing with real users.</td>
</tr>
<tr>
<td>-Usability testing, explorative testing and variations of unit testing the most commonly applied test methods.</td>
</tr>
<tr>
<td>-Moderate to good amount of resources dedicated to testing work; the average is the same as in traditional software development, but no organization reported lack of resources.</td>
</tr>
</tbody>
</table>
this can be explained by the higher emphasis on customer-based feedback. Testing has also more influence on the product design when the testing work focuses on the player experience, “fun factor” and game mechanics. It is then understandable that test results are used to fine-tune the product to cater to the target customers as closely as possible. This was voiced by all case organizations when discussing the main objectives of testing activities, even though the extent the features could be redesigned varied between the case organizations.

There are threats that should be acknowledged when addressing the validity of qualitative research [32-34]. For example, reliability and validity in qualitative research are not the same as in quantitative research, so they should be explained in more detail to put this study into a context [32]. These threats to qualitative studies have been discussed by Robson [33], who lists three main threats to validity; reactivity (the interference of the researcher’s presence), researcher bias, and respondent bias. Additionally, Whittimore et al. [34] lists integrity, authenticity, credibility and criticality as the most important attributes that are universally required for a qualitative study, and added with secondary features like explicitness or thoroughness, depending on the observed phenomena. Whittimore et al. also lists and categorizes several validity concerns, and presents a list to assess the validity of different criteria of a qualitative study. In their conclusion, the most important aspect of presenting validity is to present enough evidence for the reader to accept that the researchers truly understand the phenomena they are observing, “know what they claim to know”.

All interviewed organizations were small and medium sized game organizations, located in Southeastern Finland. This can be considered a threat, as some underlying phenomena might have been caused by the geographical proximity, such as local governance or support from the local educational institutions. However, the game business is international and all our organizations aimed their products at international markets. We also have a reason to believe that the Finnish game industry is advanced, and therefore representative for the study, with many global success stories from different platforms such as console systems (for example Alan Wake, Flatout) to mobile devices (for example Angry Birds, Clash of Clans). The threats to the study validity were addressed in many design decisions for data collection and analysis. To avoid the researcher bias, the interviews were conducted by a group of seven researchers, and several researchers participated in the data analysis to enable observer triangulation. Respondent bias was avoided by selecting organizations representing different maturities, sizes and target platforms. In any case, qualitative study results are always context-sensitive, and outside the scope of the study they should be regarded as recommendations or considerations.

6. CONCLUSIONS
We observed seven game industry software organizations by interviewing 27 game development professionals. Our objective was to understand how these organizations test their software, what the main test objectives of game developing organizations are and if the game testing differs from software testing.

The results indicate that game developers have somewhat different priorities in testing than software developers. In game development, the test process results have a larger influence on the actual end-product, and the testing work itself focuses heavily into the soft aspects of the product, such as internal mechanics, game rule balance and user experience. The game organizations have enough resources to conduct technical testing work, but they choose explorative testing and usability testing over more pre-planned approaches to fine-tune the user experience. Technical concerns are also taken into account, but they are usually a secondary concern even among developers and testers, who do most of the technical software development tasks. Overall, in game development projects the objective is to produce required functionality, in budget and in schedule, with acceptable quality, and the resulting product has to appeal emotionally to the customers.

These results can be used to help new organizations in the game business to focus their testing effort, or as a starting point on process improvement activities in game organizations. Future research should identify the most applicable test process models and activities, which could be useful to game organizations. Another option would be to collect more data on testing in game-developing organizations and draft a framework for the testing process based on the study findings. This framework could then be tested for general applicability in the game industry context.

7. ACKNOWLEDGMENTS
This study was supported by the European Union Regional Development Fund projects number A31814, “Kaakon Peliklusterit” administered by the Council of Southern Karelia and number A32139 “Game Cluster” administered by the Council of Päijät-Häme, and by the organizations funding the related research projects.

8. REFERENCES