Evaluating and processing changes for use of an early requirements engineering modelling and creativity tool in the gaming industry

Bachelor of Science Thesis in Software Engineering and Management

ILYA SHABALIN
OLIVER KJELLMAN

Department of Computer Science and Engineering
UNIVERSITY OF GOTHENBURG
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2017

Evaluating and processing changes for use of an early requirements engineering modelling and creativity tool in the gaming industry

ILYA SHABALIN
OLIVER LIDEBO KJELLMAN

Supervisor: Jennifer Horkoff, Staffan Björk
Examiner: Jan-Phillipp Steghöfer

University of Gothenburg
Chalmers University of Technology
Department of Computer Science and Engineering
SE-412 96 Göteborg
Sweden
Telephone + 46 (0)31-772 1000

Department of Computer Science and Engineering
UNIVERSITY OF GOTHENBURG
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2017

Evaluating and proposing changes for use of an early requirements engineering modeling and creativity tool in the gaming industry

Ilya Shabalin
Department of Software Engineering and Management
Gothenburg University
Gothenburg, Sweden
gusshail@student.gu.se

Oliver Lidebo Kjellman
Department of Software Engineering and Management
Gothenburg University
Gothenburg, Sweden
guskjeol@student.gu.se

Abstract— **Early requirements engineering(ERE), a process of developing ideas and requirements for software has not been widely used in the game development industry. Frameworks, such as iStar aiming at providing help with formulating models for visual representation of requirements might need to evolve to fit the needs of the game development industry. Tools which employ the iStar framework can be evaluated towards the current game development techniques. This thesis answers the research question of what techniques are used for Early Requirements engineering in gaming industry. It attempts to ascertain if current modeling tools are useful and what a goal-modeling tool and framework would need in order to be useful for the gaming industry.**

Keywords— **Early requirements engineering, game development, creativity techniques, Creative Leaf**

I. INTRODUCTION

Early requirements engineering (ERE) is one of the earliest stages of the software development lifecycle [1]. Errors in the early stages of development are frequent and very costly, which is why it is important to structure even the early processes that can otherwise be quite informal [11]. To structure early requirements with modeling a few frameworks have been developed, one notable example is goal-modeling with iStar [9]. IStar is a goal-modeling framework that uses actor-oriented modeling to help reasoning by using techniques for analysing models [9]. The goal of the frameworks is to structure initial needs and ideas of the product. For the purpose of this study the iStar framework will be used to investigate ERE process. There are also techniques for structuring and helping with creativity that can be combined with iStar. One tool that incorporates creativity techniques with goal-modeling is Creative Leaf [10]. Goal models can be used to more easily transition to the next stage of the development process [18]. Creativity techniques uses the structured requirements to support the user in coming up with more ideas through techniques like brainstorming.

Frameworks and creativity techniques have proven an asset in regular software development [19], but remains mostly unused in the game development industry. Games have a different type of requirements than regular software, placing a heavy emphasis on non-functional requirements. Non-functional requirements such as storyline and flow usually dominate requirements specification in game development [4]. Emotional responses and vague requirements such as “fun” are also requirements for games. It has been argued that game development can benefit from requirements engineering [4, 5]. Since ERE techniques are presumed to not be used as often in the gaming industry some improvements to the current frameworks to make it more fit for game development could be needed.

This thesis is a part of an ongoing study on the potential for early requirements modeling tools in game development. It will attempt to investigate the capabilities of early requirements engineering tools for game development. In addition, it will investigate what techniques, if any, are used in the gaming industry today. These topics will be discussed based on the modeling framework iStar, which is used for the process of requirement engineering. Model-driven development for games can be used to help create prototypes and support rapid game development [12, 20]. Models can help visualize needs and help facilitate communications, throughout the development process. Game development can be seen as a creative process and since IStar has functionality to help structure creativity, it is potentially valuable to use this framework while testing goal-modeling for the game industry. The thesis will use a modeling and creativity tool called Creative Leaf and analyse its effectiveness for the process of early requirements engineering for gaming industry. Creative Leaf uses creativity techniques, therefore it could be usable for the study the synergy between creativity tools and goal-modeling.

Existing modeling techniques are not specifically tailored for the gaming industry. There is a higher focus on non-functional requirements when developing video games than in normal software development. The implementation must satisfy non-functional requirements just like regular software. However, the game has non-functional requirements for cognitions and also there are non-functional requirements for emotions that should be evoked in the player for every part of the game [5].

The study will gauge if early requirements techniques, especially modeling and creativity techniques are useful in the gaming industry, and if not, what improvements to the existing frameworks can be proposed based on the feedback received from the game industry interviewees. The paper therefore will attempt to answer the following research questions:

RQ1: Are Early Requirements engineering techniques useful in a game industry context and if not, how can the established techniques be improved upon to make it compatible with current game development techniques?

RQ1.1: What techniques are used in the game industry to model and design for the early stages of development?

RQ1.2: How can goal-modeling in iStar with creativity techniques be improved to better fit game development?

The model above was made for a real-time strategy game. The model describes interaction between actors, which can be relevant for the game. In the beginning of the modeling, it is important to define actors. In this model the actors are company, player and game. Then, the game was analysed to detect more actors, such as e.g. kingdoms for the game in question. Actors, represented on the model as circles, are entities that aim at achieving goals with cooperation with other actors. Actors are usually related to each other with “is-a” or “participates-in” types of links. [9]. Some of the more important actors have goals set for them that they must achieve for the game to function. To be completed, these goals usually require soft goals to be completed as well. Achievement of the goals also require tasks to be executed. An Actor may need resources for performing a task. As can be seen on the figure #1, five actors were modelled. Company and player would have general goals, e.g. player would like to have fun and company would like to get fanbase and generate income. The game would require a number of resources, e.g. sounds and animation to function. Also, the game would need to generated resources in game world, so it would be playable. Player controlled kingdom would use this resources to defeat kingdom, controlled by computer. Overall, a model like this could help to understand what should be done for the game and gives a start for the game development.

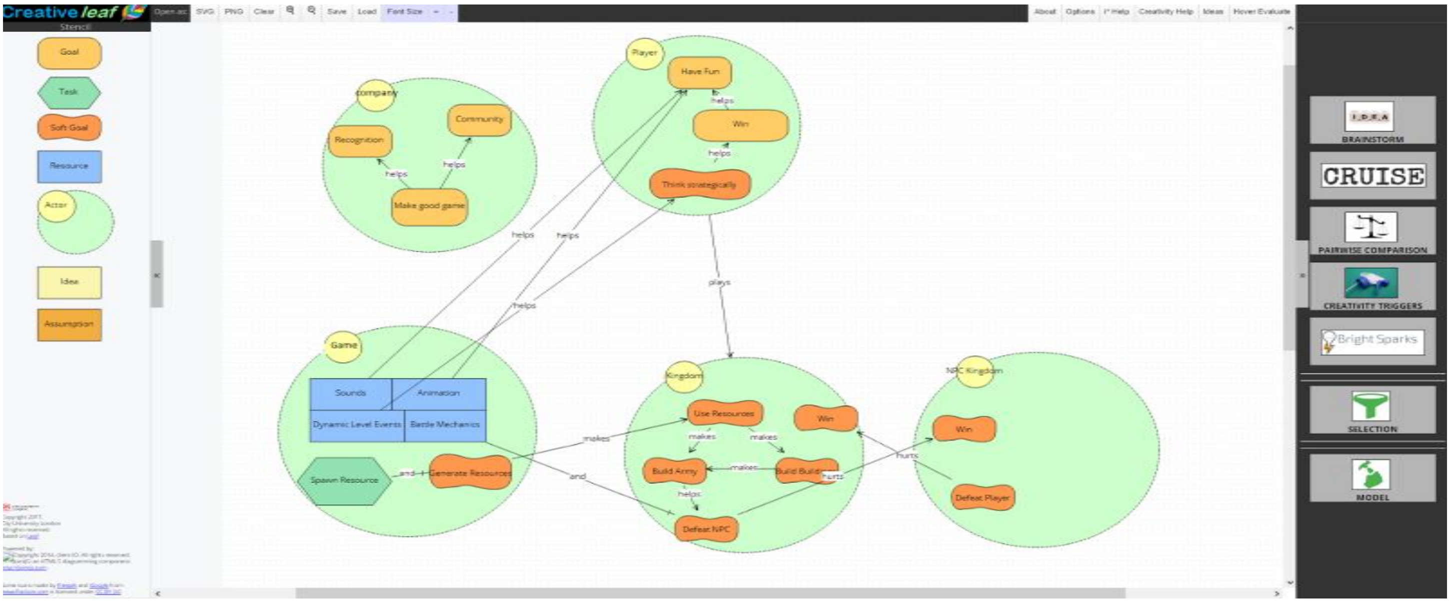


Fig. #1. *Model made in Creative Leaf*

II. Related Work

A. *Creative Leaf*

IStar is an actor oriented goal-modeling framework with goals, tasks, soft goals, resources and ideas, and different ways to connect them [9]. Creative Leaf is a tool that uses a combination of IStar goal-modeling with creativity techniques [10]. An example of a model, made in Creative Leaf in Figure #1.

B. *Literature Research*

Requirements state what a system is supposed to provide, modeling languages and techniques are used to help capture this information and represent it in a visual way. However, most of the techniques are intended for the later parts of requirements engineering (RE), which focuses on completeness and consistency of the requirements [1]. It is argued that the process of the early RE is also important for the system lifecycle, since it helps with successful development, deployment and evolution of the system [2]. Early requirements are described as the initial wishes of the customer. In early stages of RE, usually requirements are too ambitious, incomplete and are presented informally. Later phases of RE focuses more on completeness and consistency of requirements. modeling tools are supposed to help requirement engineers to make a visual representation of the requirements and make them precise and consistent so it can be passed on to the developers [1]. Early requirements engineering can benefit from creativity techniques. There exist a number of tools, which can be used to assist creative thinking [3]. These tools are supposed to help with creation of ideas and requirements at the early stages of the development.

Non-functional requirements in games can differ very much from traditional software. For example, the gaming industry is looking to provide entertainment for their customers so one of the requirements can be that the game should be fun [4]. Structuring non-functional requirements could help the process of the development, the nature of requirements in games need to place an emphasis on emotional reactions from the customer. Going from preproduction to production is where a lot of game projects fail and this is where good structured requirements from RE can make an impact [4, 5].

The process of game modeling can benefit from gameplay design patterns. These design patterns are an attempt to structurize game patterns and to improve the process for future game-development projects. Design pattern models consists of a structural framework, which describes game components and patterns which occur in game-play. The model of design patterns is used to analyse, design and compare games [6]. Gameplay design patterns are used for development of many parts of gameplay, e.g. dialogues and creation of non-playable characters [7, 8].

It is stated that game development benefits from the use of prototyping and iterations [14]. Prototyping is using the approximation of the features, artworks and concepts, to produce a working representation of the system [15]. These types of features could be modelled with the help of ERE techniques. It is argued that model driven game development can be used for games. These models can become the first building blocks of the games, benefiting game development, evolution and maintenance [12]. In addition, a combination of techniques, e.g. storyboards and scenarios can be used to develop software [16]. Storyboarding is a technique which can demonstrate system interface and how it can be used [21]. Storyboards are used for developing prototypes of games [17, 20]. However, research on how to apply the modeling tools on early stages of requirement engineering in the gaming industry is limited.

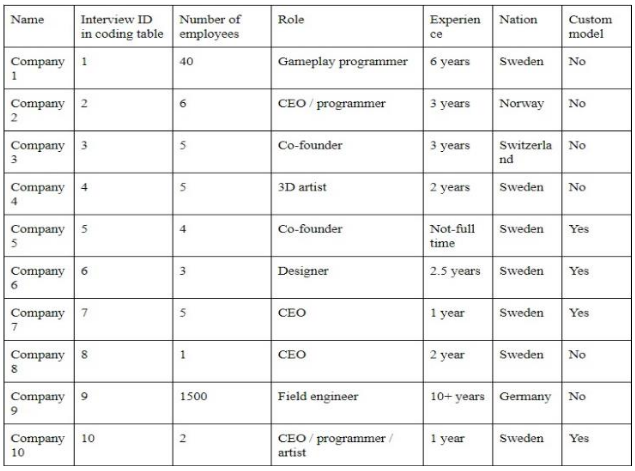
To help modeling, structuring and reasoning for organizational environments a goal and actor-oriented framework iStar was created. It is a modeling language that is used by different modeling tools which aim at modeling stakeholders' interests, model embedded systems as well as support analysis and design [1]. It is considered that using the iStar framework will make transitioning from early requirements to formal requirements easier and providing a high-level view of the models [13]. This framework was lately improved, providing user support and an updated set of core concepts. [9]

One of the existing tools for ERE is Creative Leaf. It is a tool, where users can model new software with the help of iStar, define requirements and use different creativity techniques to help structure creativity during the process of creating requirements and making goal models [10].

III. Research Methodology

A. Research strategy

The study is qualitative in nature and was carried out as an exploratory case study. Game development companies were approached for this study. Companies are briefly described in Section 3.1. Representatives of the companies were interviewed according to the interview protocol, which can be found in Appendix A. Two students from the Gothenburg University bachelor's software engineering and management program, who had prior game development experience, were used to refine this protocol. The pilot interviews were conducted so that it would be possible to estimate the time needed for each question and how long the discussion would last. Interviews were semi-structured, leaving room for open ended answers. A search was done to determine companies that could be used for this study. The mailing list of the companies, around the Gothenburg area was presented by the supervisors. In the e-mail, representatives of the companies were asked to perform a 30-40 minute interview. However, only 3 companies responded. Therefore, another way of finding interviewees was required. It was suggested by the supervisors that interviews can be completed over the Nordic Game 2017, a fair with game developers taking place on 17-19 May in Malmo, Sweden. In the end a total of ten companies were found and interviewed. It was promised that companies and representatives are anonymized. The companies are described in table #1.



Name	Interview ID in coding table	Number of employees	Role	Experience	Nation	Custom model
Company 1	1	40	Gameplay programmer	6 years	Sweden	No
Company 2	2	6	CEO / programmer	3 years	Norway	No
Company 3	3	5	Co-founder	3 years	Switzerland	No
Company 4	4	5	3D artist	2 years	Sweden	No
Company 5	5	4	Co-founder	Not-full time	Sweden	Yes
Company 6	6	3	Designer	2.5 years	Sweden	Yes
Company 7	7	5	CEO	1 year	Sweden	Yes
Company 8	8	1	CEO	2 year	Sweden	No
Company 9	9	1500	Field engineer	10+ years	Germany	No
Company 10	10	2	CEO / programmer / artist	1 year	Sweden	Yes

TABLE 1. DESCRIPTION OF THE COMPANIES

Overall, the interviewed companies are from 4 countries: Sweden, Germany, Norway and Switzerland. Eight of the companies are small 1-6 people companies. All of the people in the smaller companies have multiple roles, so a CEO can also be a programmer or artist.

People mostly from smaller companies were interviewed about how they come up with new game ideas, and if they use tools or techniques to help them structure their creative process. goal-modeling and creativity techniques were explained and demonstrated in the Creative Leaf tool for the interviewees to interact with. When they had seen the tool and the goal models and knew their use they were asked about their feedback on the tool and if they thought that it could be useful for them when developing games, and if not, what changes would they want to make it compatible with their current state of the art. Then the list of changes was made and presented to a four of the developers previously interviewed for a follow-up interview to evaluate the benefit of the changes. The companies for the follow-up interviews were chosen by a convenience sample because they were located in Gothenburg. The companies for the follow-ups have following ID's 5, 6, 7 and 10.

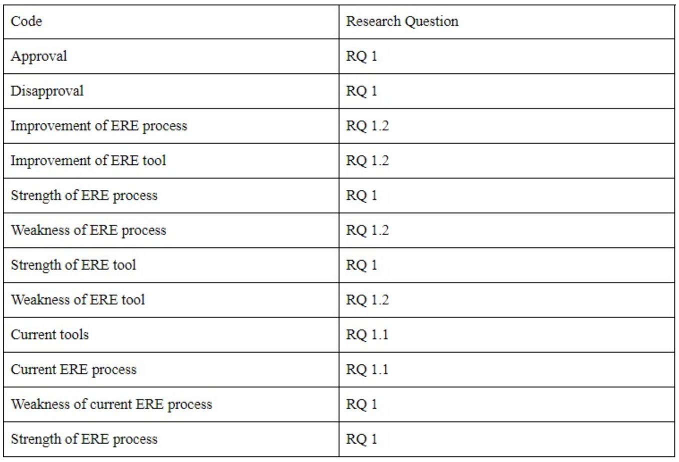
The first interview was with company number 5, a model which had high complexity was shown to the interviewee, but it was too hard to understand and another interview was requested with an easier to understand model. The model was customized as a simple model of a product from their company. The initial idea was to customize models for every company but in the end only the scheduled interviews received a customized model, the others all got the same model of a simplified real time strategy game. There were control questions to see if the interviewees understood the models.

For follow-up interviews the protocol was much simpler, as less questions required attention. The interview mostly consisted of an explanation of the changes and feedback regarding the changes. The usefulness of the tool was also in question and the interviewees were asked if they would use the tool with the proposed changes.

The protocol for initial interviews can be found in Appendix A and the protocol for follow-up interviews can be found in Appendix B.

B. Data Analysis

All the company interviews had their audio recorded and transcribed. The transcribed data was then put into coding tables [22]. The research questions were examined along with the transcriptions to establish a few codes and to try and relate the quotes back to the research questions. The relation between research question and codes can be seen in table #2.



Code	Research Question
Approval	RQ 1
Disapproval	RQ 1
Improvement of ERE process	RQ 1.2
Improvement of ERE tool	RQ 1.2
Strength of ERE process	RQ 1
Weakness of ERE process	RQ 1.2
Strength of ERE tool	RQ 1
Weakness of ERE tool	RQ 1.2
Current tools	RQ 1.1
Current ERE process	RQ 1.1
Weakness of current ERE process	RQ 1
Strength of ERE process	RQ 1

TABLE 2. CODES RELATION TO THE RESEARCH QUESTIONS

Current tools and current process are two codes that relate to RQ1.1, what the game developers are currently using to come up with ideas and structuring them in the early part of development. The next set of codes is pertaining to strengths and weaknesses of Creative Leaf and the iStar framework from the game developer's perspective. The code aims to find out if people in the industry think that ERE framework, like iStar are useful or not and also give some hints at what parts of it are useful and what is not. Some improvements could be implied from the weaknesses of the ERE tool and process. For the scope of the study ERE process is analysed with use of iStar framework. Suggested improvements to the tool and process were also coded for the more concrete improvements suggested by the interviewees. Lastly the strengths and weakness of their ERE processes were to be coded to understand how ERE techniques can be compatible with or substitute to their current techniques but the data for this was either non-existent or told for a process further down the development cycle.

During the coding process some quotes were discovered that could best be described as approval or disapproval without mentioning a strength or weakness and thus was coded as approval and disapproval. This information can potentially be valuable to answer if the tool and process are useful. A margin note was added on every quote to break it down and more easily relate it to other quotes.

Based on the suggested improvements, concept for changes to the tool was made in an image editor and loosely documented. Follow up interviews were conducted to see what the interviewees thought of the changes and if they would use the tool if they were implemented. These follow-up interviews were recorded and transcribed as well. The transcripts were coded for approval or disapproval of the changes and the usefulness of the tool. The code "no longer ERE" emerged during the coding process, because some of the interviewees now thought it was more of a task management tool.

C. Validity threats

Construct validity: The concept of usefulness is hard to measure. If some part of the tools or processes tested receive positive remarks it can be counted as partly useful. Just a positive remark is not enough to deem the tool as useful. If the tool and process receive negative or neutral remarks it will be regarded as not useful. The ultimate measure of usefulness is if the person in question would actually use the tool and process in a real context. Changes meant to be improvements could also be seen as detriments if the added value is lower than the added complexity. The concept of suggested improvements will be evaluated by some of the game developers, to ascertain the usefulness of the improvements to the tool and process.

Internal validity: Since there are two researchers findings can be discussed and analysed from two different perspectives, neither has any stake in the tool or framework and can thus view the findings objectively. The interview protocols were not followed rigorously because the questions would have already been answered due to the open-endedness of the previous questions. It is redundant to ask a question that has already been answered. These answers were mostly about their role in the industry and information about their company, points in the interview prior to the gathering of feedback on the tool and process.

External validity: The sampling was done by convenience-sampling and thus any available game developers were interviewed. 8/10 interviews were conducted on small European game development studios ranging from 1 to 5 people. Out of these, 4 were conducted appointments and the rest were conducted ad-hoc during Nordic Game 2017 which is a fair for game developers. The interviews from the fair are still valuable, even if the interviews were slightly rushed in comparison, because all topics were touched and the models were understood. All the other interviewees could still show that they somewhat understood the models because they were questioned about it, but 20 minute interviews are probably not enough to understand goal modelling in iStar fully. The findings can only be generalized for small game development companies and not for the game industry at large. For the follow-up interviews only the 4 people from the Gothenburg Game Incubator were selected because they were the only accessible follow-up interviewees. Some people might have given more approval statements to be polite, however before the interviews it was stated that the interviewers do not have any stake in the tool or framework.

Reliability: The first interview was conducted with a complex model that would not be suited for a first time viewer or user of the framework and tool. An easier to understand model was requested and made. The first four interviews had very basic models modeled after one of their games.

IV. Results

A. Analysis

This section will provide the findings of the qualitative research done on the process of Early Requirements Engineering in gaming industry. The analysis was done in three parts. In the first one, coding of the interviews was performed. In the second part, the discussion of the coding results was presented and changes which can be made for Creative Leaf tool will be discussed. In the later part, evaluation of the follow-up interviews will be done to evaluate the response to changes suggested. Coding and transcribing procedure was described in section III, subsections c and d. The coding for first part was done from ten interviews, following the interview protocol, described in the Appendix A. The coding procedure for third part was made from four follow-up interviews, following interview protocol, which can be found in the Appendix B. Transcripts for interviews can be found in Appendixes F and G.

The data from the ten interviews was categorized and sorted in the coding table which can be found in Appendix C.

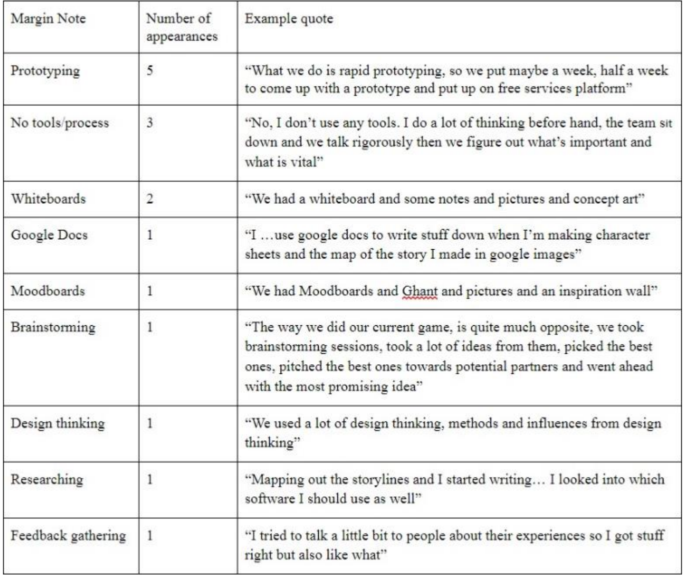
After performing this qualitative research, it was possible to analyse the process of the Requirements Engineering in the Gaming Industry. Following common topics were detected throughout the interview and are worth discussing:

Current Early Requirements Engineering process and tools in gaming industry.

As demonstrated in Appendix C, game designing companies are using tools and practices at the stage of the early requirements engineering. Five companies, Interview #3, 4, 6, 7 and 10 perform prototyping process at the early stages of the requirements engineering. It can be seen in three interviews (#6, 7, 8) that companies gather feedback and try to further elaborate ideas to verify if the ideas will fit into market need. Some tools are being used, such as whiteboards and mood boards for discussion of the ideas, and Google docs for the storage of the information, however none of the tools are specifically tailored for early requirements engineering. Also, two companies with ID 9 and 10 are not using any specific tools. However, in interview #1, a feeling that "actual idea of requirements capturing is quite alien to the industry" was described.

Overall, according to interviews #1, 4, 6, 7, 8 and 10 that companies perform some sort of requirements engineering, however there is no specific tool or process that all the companies follow. Next sections will focus on the weaknesses and strengths of the proposed process and tool, such as Creative Leaf.

To answer RQ 1.1, for the most part no techniques are used explicitly for ERE in the industry and the process is very ad-hoc, but some techniques for early development like brainstorming, design thinking and Mood boards. Other techniques for trying to validate their ideas exist, like feedback gathering and prototyping. This is summarised in table #3.



Margin Note	Number of appearances	Example quote
Prototyping	5	“What we do is rapid prototyping, so we put maybe a week, half a week to come up with a prototype and put up on free services platform”
No tools/process	3	“No, I don’t use any tools. I do a lot of thinking before hand, the team sit down and we talk rigorously then we figure out what’s important and what is vital”
Whiteboards	2	“We had a whiteboard and some notes and pictures and concept art”
Google Docs	1	“I …use google docs to write stuff down when I’m making character sheets and the map of the story I made in google images”
Moodboards	1	“We had Moodboards and Ghant and pictures and an inspiration wall”
Brainstorming	1	“The way we did our current game, is quite much opposite, we took brainstorming sessions, took a lot of ideas from them, picked the best ones, pitched the best ones towards potential partners and went ahead with the most promising idea”
Design thinking	1	“We used a lot of design thinking, methods and influences from design thinking”
Researching	1	“Mapping out the storylines and I started writing… I looked into which software I should use as well”
Feedback gathering	1	“I tried to talk a little bit to people about their experiences so I got stuff right but also like what”

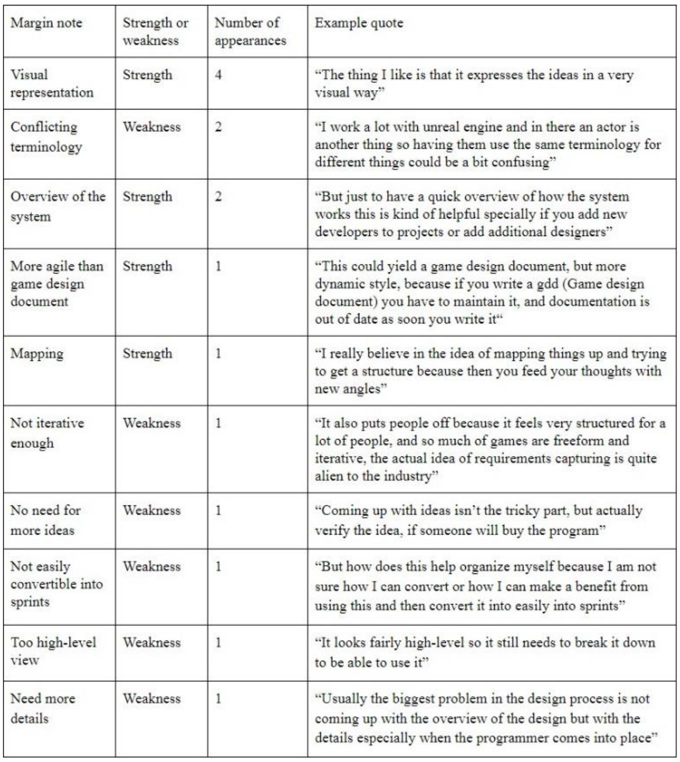
TABLE # 3. PROCESSES AND TOOLS USED IN THE GAME DEVELOPMENT INDUSTRY

Strengths and weaknesses of the ERE process.

After describing the early requirements process, during the interviews companies discussed features of iStar and Creative Leaf. Companies thought that visual representation can help to communicate ideas throughout the development teams (interviews #1-4). Also, they thought that the framework helps map new ideas to see the bigger picture and an overview of the system (interviews #5 and 9). In addition, according to interview #3, this way of representing requirements by using a model is more expressive than a game design document, which is a document that describes the game.

However, interviewees expressed that it was difficult to understand the terminology suggested by iStar framework. One of the biggest confusions was with the term task, where it was difficult to distinguish between tasks which can be modeled in iStar and tasks that are defined in agile management tools, e.g. Trello or Pivotal Tracker. In addition, there were some individual negative remarks, pointing out that the models need more details (interview #9), difficulty in conversion to sprints (interview #7), being not iterative enough (interview #1).

To partly answer the main research question of the thesis, some teams think that there is potential for it to be useful as a means of communication and a visual overview of the system’s main functionality. However it is not perceived to be agile enough. The perception of strengths are more concentrated than were the visual nature of the model is almost the only strength, weaknesses but a lot of weaknesses could be identified. More details are summarized in table #4.



Margin note	Strength or weakness	Number of appearances	Example quote
Visual representation	Strength	4	“The thing I like is that it expresses the ideas in a very visual way”
Conflicting terminology	Weakness	2	“I work a lot with unreal engine and in there an actor is another thing so having them use the same terminology for different things could be a bit confusing”
Overview of the system	Strength	2	“But just to have a quick overview of how the system works this is kind of helpful specially if you add new developers to projects or add additional designers”
More agile than game design document	Strength	1	“This could yield a game design document, but more dynamic style, because if you write a gdd (Game design document) you have to maintain it, and documentation is out of date as soon you write it“
Mapping	Strength	1	“I really believe in the idea of mapping things up and trying to get a structure because then you feed your thoughts with new angles”
Not iterative enough	Weakness	1	“It also puts people off because it feels very structured for a lot of people, and so much of games are freeform and iterative, the actual idea of requirements capturing is quite alien to the industry”
No need for more ideas	Weakness	1	“Coming up with ideas isn’t the tricky part, but actually verify the idea, if someone will buy the program”
Not easily convertible into sprints	Weakness	1	“But how does this help organize myself because I am not sure how I can convert or how I can make a benefit from using this and then convert it into easily into sprints”
Too high-level view	Weakness	1	“It looks fairly high-level so it still needs to break it down to be able to use it”
Need more details	Weakness	1	“Usually the biggest problem in the design process is not coming up with the overview of the design but with the details especially when the programmer comes into place”

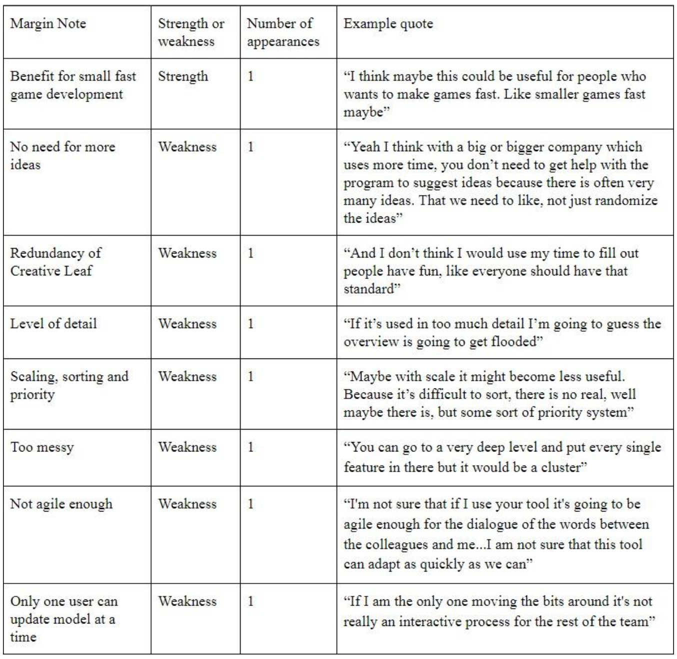
TABLE # 4. STRENGTHS AND WEAKNESSES OF THE ERE PROCESS

Strength and weaknesses of the ERE tool.

Throughout the interviews, the functionality of Creative Leaf was described to the interviewees. As described in the interview protocol, the question was asked if this kind of tool can be used for game development. As a response, strong and weak points of the tool were evaluated. A respondent in interview #2 speculated that this tool will be “useful for people who want to make games fast”. However, in general, a number of responses outlined weaknesses of Creative Leaf. In gaming industry, according to interview #2, there is no need to generate new ideas, since “there are often many ideas”. However, instead of using a tool to get ideas there should be some way to evaluate which ideas are more valuable. General, throughout the interviews, creativity techniques were perceived negative or indifferent.

Another problem of the tool, according to the interviewees is that is looks too complicated and isn’t agile enough. It was suggested that due to the deep level of detail, the model can “get flooded” (interviews #4, 7). And as it was described in interview #10, that ideas throughout the discussion alter quickly and interviewee #10 described a worry that “the tool wouldn’t be able to adapt as quickly”.

To yet again partly answer research question 1, some responses speculate at how it could be useful for other teams than their own teams, but most of the interviewees seem to think that there are many weaknesses, some want more detail and some think that the model might get flooded. The creativity techniques doesn't seem very necessary as the only comments about them were negative and that they don't need more ideas, just ways to verify them. Overall, details are summarized in table #5.



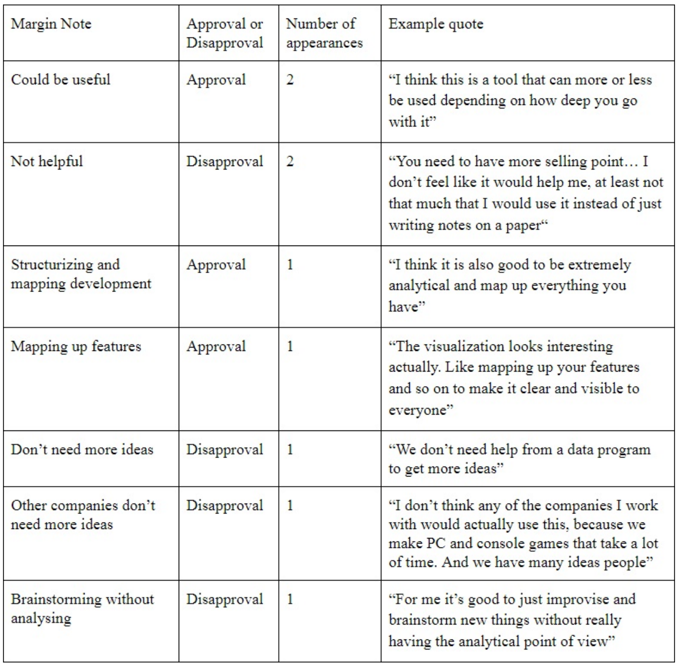
Margin Note	Strength or weakness	Number of appearances	Example quote
Benefit for small fast game development	Strength	1	"I think maybe this could be useful for people who wants to make games fast. Like smaller games fast maybe"
No need for more ideas	Weakness	1	"Yeah I think with a big or bigger company which uses more time, you don't need to get help with the program to suggest ideas because there is often very many ideas. That we need to like, not just randomize the ideas"
Redundancy of Creative Leaf	Weakness	1	"And I don't think I would use my time to fill out people have fun, like everyone should have that standard"
Level of detail	Weakness	1	"If it's used in too much detail I'm going to guess the overview is going to get flooded"
Scaling, sorting and priority	Weakness	1	"Maybe with scale it might become less useful. Because it's difficult to sort, there is no real, well maybe there is, but some sort of priority system"
Too messy	Weakness	1	"You can go to a very deep level and put every single feature in there but it would be a cluster"
Not agile enough	Weakness	1	"I'm not sure that if I use your tool it's going to be agile enough for the dialogue of the words between the colleagues and me...I am not sure that this tool can adapt as quickly as we can"
Only one user can update model at a time	Weakness	1	"If I am the only one moving the bits around it's not really an interactive process for the rest of the team"

TABLE # 5. STRENGTHS AND WEAKNESSES OF THE ERE TOOL

Approval and Disapproval of the ERE tool.

Overall, the response from the interviews towards Creative Leaf was slightly negative/neutral. Some of the interviewees had difficulties in evaluating if the tool can be useful or not. In interviews #3, 4, 5, 8 the tool was evaluated as useful, e.g. for mapping up the game and features (interviews #5 and 8). However, in interviews #2, 5, 6, 7 a feeling that Creative Leaf is not helpful enough was expressed. It was mentioned that companies does not require help from a tool to get new ideas and that usage of a tool consumes time, which can be invested in brainstorming and it would slow down the development process (interview #5).

General approvals and disapprovals can match against the usefulness in RQ 1. ERE can not be viewed as overly useful in a game development context since there are more negative remarks than positive remarks, only an overwhelmingly positive result can be generalized as ERE being useful and that is not the case.



Margin Note	Approval or Disapproval	Number of appearances	Example quote
Could be useful	Approval	2	"I think this is a tool that can more or less be used depending on how deep you go with it"
Not helpful	Disapproval	2	"You need to have more selling point… I don't feel like it would help me, at least not that much that I would use it instead of just writing notes on a paper"
Structurizing and mapping development	Approval	1	"I think it is also good to be extremely analytical and map up everything you have"
Mapping up features	Approval	1	"The visualization looks interesting actually. Like mapping up your features and so on to make it clear and visible to everyone"
Don't need more ideas	Disapproval	1	"We don't need help from a data program to get more ideas"
Other companies don't need more ideas	Disapproval	1	"I don't think any of the companies I work with would actually use this, because we make PC and console games that take a lot of time. And we have many ideas people"
Brainstorming without analysing	Disapproval	1	"For me it's good to just improvise and brainstorm new things without really having the analytical point of view"

TABLE # 6. EVALUATION OF ERE TOOL

Changes for the ERE tool and process

To make the tool and the iStar framework more appealing to the game developers, the following changes were suggested. These changes are based on the feedback received in the interviewees. As seen in interview #4, a prioritization function for features would "help a lot". Also, the terminology should be more specific (interview #5). Some developers wanted a way to connect to other tools, e.g. scrum tools like Trello or Pivotal Tracker (interviews #3,7) and importing other sources, such as images or mood boards (interview #8). A need for description boxes, which would pop-up to describe the ideas behind nodes was described in interviews #3, 8 and 9. A way to have multiple viewpoints, which would promote agility of the tool was described in interview #1. A need to show the core of the game in a better way was expressed in interview #2. It would be quite valuable to verify the ideas of the developers to "find some players, some people that ... are actually interested in this" (interviews #6 and 7). In addition, to make this tool more agile, a collaborative function should exist, so that people can work remotely and as a group on the same model.

This section attempts to answer RQ 1.2 by listing a few of the features that were suggested by the game developers. The tool can be improved by adding e.g. collaboration and support for agile task managers. IStar as a framework can be improved by adding e.g. support for images and therefore also mood boards, layers for different development backgrounds and less conflicting terminology. Overall, the table with proposed changes can be found in Appendix D and are discussed in the section V.

These ideas and some other findings are described, linked to the interviews and evaluated into a list of proposed changes for the ERE process and the tool, which can be seen in the next section.

V. Discussion

A. Proposed changes for the ERE process and the tool

This section focuses on a set of possible improvements that can be suggested to improve usability, productivity and attitude towards the usage of ERE processes and a tool like Creative Leaf for game developers. A list of changes that can be made to the Creative Leaf can be seen below. In addition, a concept was made to graphically show the proposed changes. More detailed list of suggestion can be seen in Appendix D.

1. Reducing a confusion of the terms of iStar format. Throughout the interviews, respondents were confused with the terminology of iStar, such as ERE task vs Trello task and iStar actor vs actor of Unreal Engine. As a suggestion, a discussion with creators of the iStar framework can be made, and a solution can be found by changing the terminology in iStar to minimize conflicts in terminology with game development.

2. The iStar framework could also benefit from more flexible description of arrows and descriptions, interviewee 5 thought that the terminology wasn't descriptive enough, the term wasn't descriptive enough. Adding description boxes that can be hidden could be used even for the arrows.

3. Changing position of creativity techniques. Throughout the interviews, one of the respondents said that there is no need for more ideas (interview #2, 6 and 7) also no support or praise was given to the creativity techniques when shown, therefore, it was decided to compress the space used for creativity tools, instead of it having a button on the top of the screen, instead of the black space, which will make creativity techniques appear on the side of the screen. This can be seen in the figure #2.

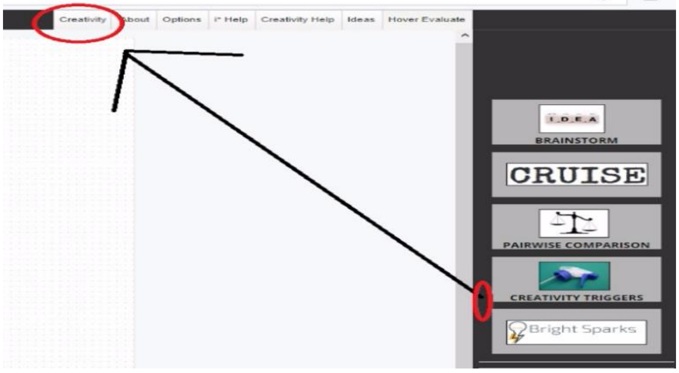


Fig. #2. *Moving Creativity tools to the top of the screen*

4. Make Creative Leaf collaborative, so that more than one person can work at a time. A good example for this is the Google drive tool Draw.io. This can be substantiated with a need from interviewee #10. And this would make the tool more agile, as it can be more easily used for the agile development, and more accessible.

5. Adding extra button, symmetric to delete button, which will can hide the internal contents of an actor, and show contents if clicked again. Interviewee #4 suggests that the model can become "flooded". This might also lead to better scaling and reduce the amount of mess in the model. This change was not concretely suggested by game developers, but implied through proposed weaknesses. Concept can be seen in the figure #3.

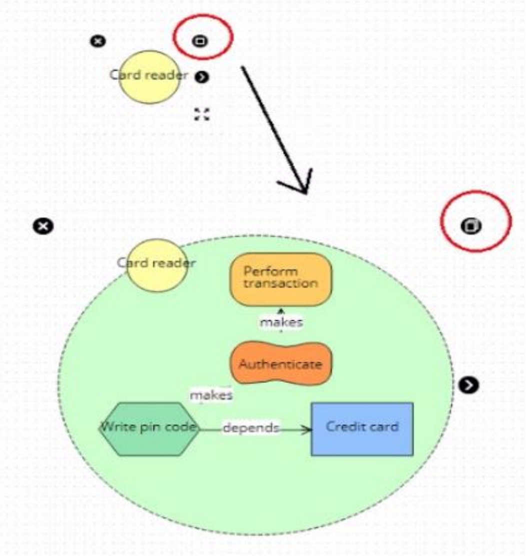


Fig. #3. *Moving Creativity tools to the top of the screen*

6. Prioritization. In Creative Leaf priorities for ideas already exist, however, it would be beneficial to replicate it to all nodes. Also, instead of the three buttons, it can be suggested to put one button on the top of the node, which when pressed would give ability to change priority from 1 to 5 with a scrollbar. Interviewee #4 thought that prioritization would help a lot. The change from three buttons to a 1 to 5 system is not substantiated in the interviews. Improvement number 9, the description box would possibly obstruct the 3 button design. This can be seen in the figure #4.

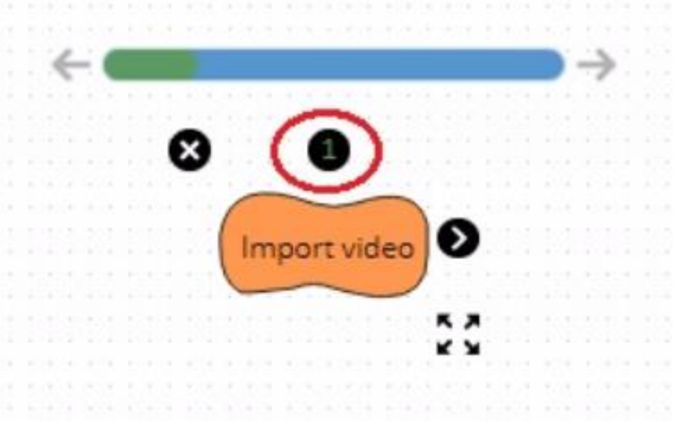


Fig. #4. *Priority button for the nodes.*

7. Adding custom text to the model outside the nodes. As was described in interview #5, it should be possible to add a headline to the model, which would improve the overall understanding of the model. Therefore, it is suggested to add an ability to put text in fonts, sizes and colors on the model. Texts and fonts can be more flexible than just the ability to add a big headline.

8. Ability to link between layers and tags, and layers/tags which can be targeted for specific worker, for example the manager would benefit from complete picture, however the designer of programmer would benefit from seeing information only relevant to him/her. Interview #1 and 5 suggests alternate adding viewpoints, but to be able to link between them, it is easier to implement a layering structure. The proposed layering structure can be seen in the figure #5.

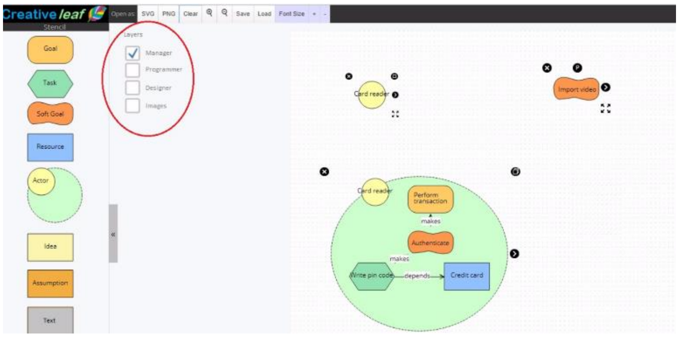


Fig. #5. *Layering structure.*

9. Ability to add a description boxes to all the nodes. Nodes could be minimized and maximized by a double click and could contain valuable information. Three of the interviewees with ID's # 3, 9 and 10 suggested this.

10. A Trello integration module. A concept where an integration between the modeling tool and a scrum tool e.g. Trello was made. It would be possible to select start date, deadline, priority and acceptance criteria. A module would be able to translate tasks and soft goals into Trello roadmaps.

Interviewee #7 suggested an easier way to transition from model to sprints. This module can be seen in the figure #6.

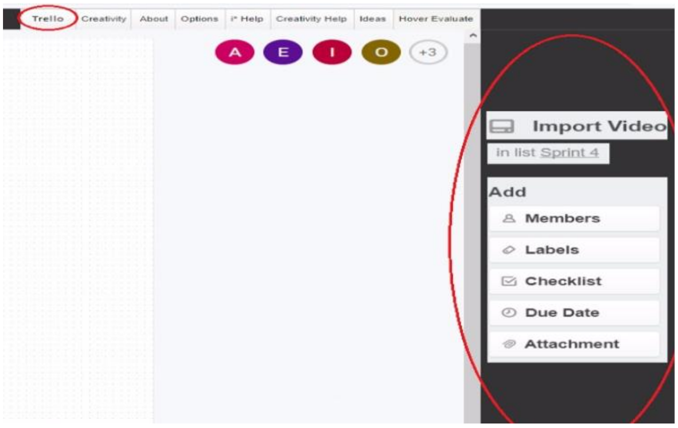


Fig. #6. *Trello layer.*

11. Support for verification of ideas were suggested by interviewees #6 and 7 . This concept can be discussed in a future study, as it's a complex and broad theme, which would require some research and brainstorming of ideas. It can in part be done via goal model analysis but this needs to be tested in a further study. [23]

12. Image layer. Importing images and the possibility to put them in a layer, which can be shown or hidden. This was suggested in interview #8.

Overall, the full concept of changes can be seen in the figure #7.

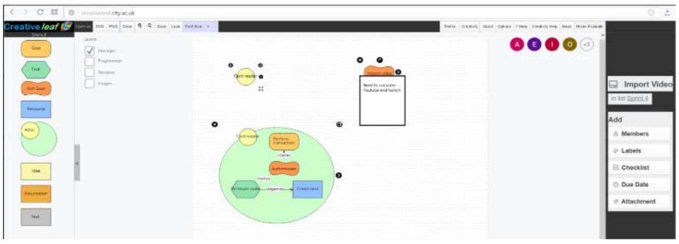


Fig. #7. *Full concept.*

This set of changes was described to interviewees #5, 6, 7, 10 in the follow-up interviews. Section V.C will focus on the responses from the respondents about these changes.

B. Discussion of changes for ERE process and tool.

The suggested improvements do not necessarily only make the tool better for game developers. Some of the changes could be beneficial for any user of goal-modeling tools. Suggestions #1 and 8 from section V.A can be seen as slightly more beneficial for game developers. The first one looks at the terminology of iStar and there is conflicting terminology that would only apply for game developers i.e. actor in Unreal Engine vs actor in iStar. Suggestion number 8 could be viewed as slightly more beneficial for game developers as the game development teams consist of people with vastly different backgrounds e.g. story writers and programmers.

Suggestions 3 and 12 can be seen as a change aimed mostly at game developers. Suggestion #3 from section V.A would decrease the focus given to creativity techniques because game developers shared that coming up with ideas isn't hard for them, which can be seen in e.g. interview #6. Suggestion #12 can be seen as especially tailored for game developers because of the graphical nature of games and the need to have aesthetics incorporated into their models. Also mood boards are images and can be imported as well.

The other changes could be beneficial for anyone using the tool or process. However the added detail from description boxes and new layers and the addition of a task manager connection could make one argue that the tool is no longer for the earliest stage of development and it has now become no longer tailored specifically for ERE.

C. Evaluation of changes.

After creating the list of the improvements, follow-up interviews with four of the original subjects were conducted. The results were coded according to the coding procedure, which can be found in Appendix B and the coding is included in the Appendix E.

After describing the changes to the subjects of the follow-up interviews, the goal was to find out if the interviewees approve or disapprove the changes and if the proposed updates to the tool would convince game developers to use ERE process and tools. In addition, throughout the interviews the functionality of the tool in the eyes of some of the developers shifted from ERE to project managerial.

Perception of changes

The proposed addition of the Trello tool was perceived positively from the developers, with 3 out 4 subject praising concept for transitioning into product backlog (interviews #1, 2 and 4). Layering also received approval from interviewee, since "you can flip from one (layer) to another, and that's way easier to understand". Proposed changes also, according to interviews #1 and 3 would detect hidden workloads and make models more simplified. However, most of the respondents think that changes would not give enough benefit and still it would be easier and better to just use a whiteboard.

To answer research question 1.2 interviewees are most positive towards the possibility of transition into to agile task management tools, such Trello. The interviewees are also positive towards the inclusion of images and layers. Collaboration also received praise. These are the only changes which are seen as most important, thus they can be seen as improvements.

Usefulness of the tool

One of the questions asked during the follow-up interviews was if the tool could be hypothetically used at the start of the new project. Interviewees #3 and 4 responded that the tool can help with communication and graphical representation and communication in the team, and subject #3 expressed that he would be willing to use the tool next time starting a new project. However, the general response was slightly negative, due to belief that tool would not provide enough benefit and would be difficult to learn and introduce to their teams.

To answer RQ1, it can be seen that ERE techniques as of now is not necessarily useful in the game development industry context, at least not for smaller companies. This is due to the nature of game development, since game developers believe that imposition of the tool and framework would not provide enough benefit and would be difficult to adopt into the company process. Attempts can be made to improve the established techniques by tailoring them for game terminology and usage of images and layers. An ERE tool could benefit from being collaborative and providing support for transition from the idea stage to later stages via agile task managers. The ERE techniques could be improved to make them compatible with some game development techniques such as mood board and most likely prototyping.

VI. Conclusion and future work

A. Conclusion

The purpose of this thesis was to investigate whether or not early requirements engineering would be useful for game development and to find out what game developers use instead and if it's possible to propose changes to make it more useful. Most of the companies interviewed could see a limited benefit in using the tool but would rather continue to use their techniques which mostly consisted sketching ideas on whiteboards and using post it notes for early ideas and then building quick prototypes to validate those ideas against the market. Creativity techniques were received with indifferent or negative remarks. Improvements were suggested to goal-modeling tools which make them collaborative and provide support with agile task managers, and to allow for more detail without making it too messy. When confronted with concept for these changes they could see that it was more valuable now, but generally most would still not use it. Only one out of ten interviewees would consider using it to facilitate communications in the teams and would be willing to try it out, and then only in its improved form.

B. Future work

This research raised some possible topics for future work. One of the topics was raised at the data gathering process. Some interviewees suggested creating a tool, which would be able to validate the ideas. The main problem with ideas in the start of projects were that developers have difficulties to detect which ideas would get positive response on the market and would generate income. Structuring their ideas and coming up with them was not perceived as a problem by the game developers. Therefore, a study of how ideas can be better evaluated and sorted can with the help of goal-modeling be done. The topic can include creation of an algorithm or an artificial network which would analyse current trends and other data on the internet which can be useful for analysing the ideas.

A study like this could be replicated with bigger game development companies to compare the response and the process of the ERE in big vs small companies.

The ongoing project that this thesis is part of might implement and evaluate some of the changes proposed in this thesis.

Overall, a follow-up study to create a new framework, based on iStar, specifically customized for game development could be conducted instead of changing the current iStar to encompass all kinds of software development.

VII. ACKNOLEDGMENT

Researchers would like to thank Gothenburg University for providing funding for tickets for Nordic Game 2017, Jennifer Horkoff and Staffan Björk for supervision and support throughout the project. In addition, we would like to thanks all the interviewees, the organizators of Nordic Game 2017 and Gaming Incubator in Gothenburg.

REFERENCES

[1] E. S. K. Yu, Towards modelling and reasoning support for early-phase requirements engineering. Proceedings of the Third IEEE International Symposium, 1997 pp. 226-235.

[2] J. A. Bubenko, Challenges in Requirements Engineering. Proceedings of the 2nd IEEE International Symposium on Requirements Engineering, 1995.

[3] A. D. P. Carvalho et al, Tools for creativity. Proceedings of the Third IEEE International Symposium on Requirements Engineering, 2012

[4] D. Callele et al. “Requirements Engineering and the Creative Process in the Video Game Industry”. Proceedings of the 13th IEEE International Conference on Requirements Engineering, 2005

[5] D. Callele et al. “Emotional Requirements in Video Games”. Proceedings of the 2006 14th IEEE International Requirements Engineering Conference, 2006

[6] Björk, S., Lundgren, S. & Holopainen, J. “Game Design Patterns”. Copier, M. & Raessens, J. (Eds.) (2003) Level Up - Proceedings of Digital Games Research Conference 2003, Utrecht, The Netherlands, 4-6 November 2003

[7] Brusk, J. & Björk, S. Gameplay Design Patterns for Game Dialogues. Paper presentation at DiGRA 2009: Breaking New Ground: Innovation in Games, Play, Practice and Theory. London, UK, 2009

[8] Lankoski, P. & Björk, S. Gameplay Design Patterns for Believable Non-Player Characters. Paper presentation at DiGRA 2007, Tokyo, Japan, 2007

[9] F. Dalpiaz, X. Franch, & J. Horkoff, iStar 2.0 Language Guide. Cornell University, USA, 2016

[10] J. Horkoff and N. Maiden, Creative Leaf: A Creative iStar Modeling Tool. City University London, UK, 2016

[11] A. Fuxman, L. Liu, J. Mylopoulos, M. Pistore, M. Roveri, P. Traverso. Specifying and Analyzing Early Requirements in Tropos, 2004

[12] Emanuel Montero Reyno and José Á Carsí Cubel. Automatic prototyping in model-driven game development. Comput. Entertain. 7, 2, 2009.

[13] E. Dubois, E. Yu & M. Petit. ."From early to late formal requirements: a process-control case study," Proceedings Ninth International Workshop on Software Specification and Design, Ise-Shima, pp. 34-42, 1998

[14] C. M. Kanode & H. M. Haddad. "Software Engineering Challenges in Game Development," 2009 Sixth International Conference on Information Technology: New Generations, Las Vegas, NV, pp. 260-265, 2009

[15] Tracy Fullerton , Christopher Swain , Steven Hoffman, Game Design Workshop: Designing, Prototyping, and Playtesting Games, CMP Books, 2004

[16] Colin Venters, Rob Procter, Alistair Sutcliffe, John McNaught, Oscar de Bruijn, Iain Buchan, Sarah Thew, "Experience in e-Science Requirements Engineering", 2013 21st IEEE International Requirements Engineering Conference (RE), vol. 00, no. , pp. 277-282, 2008

[17] Jean Lee Tan, Dion Hoe-Lian Goh, Rebecca P. Ang, Vivien S. Huan, Participatory evaluation of an educational game for social skills acquisition, Computers & Education, Volume 64, Pages 70-80, 2013.

[18] Horkoff, Jennifer, et al. “Using goal models downstream: a systematic roadmap and literature review”. International Journal of Information System Modeling and Design (IJISMD), vol. 6.2, pp. 1-42, 2015

[19] Yu, E., Giorgini, P., Maiden, N., Mylopoulos, J. (eds): Social Modeling for Requirements Engineering. MIT Press, Cambridge, 2011

[20] K. M. Cooper and C. S. Longstreet, “Towards model-driven game engineering for serious educational games: Tailored use cases for game requirements,” in Computer Games (CGAMES), 2012 17th International Conference on, pp. 208–212, IEEE, 2012.

[21] Khai N. Truong, Gillian R. Hayes, and Gregory D. AbowdStoryboarding: an empirical determination of best practices and effective guidelines. In Proceedings of the 6th conference on Designing Interactive systems (DIS '06). ACM, New York, NY, USA, 12-21, 2006

[22] C. B. Seaman, "Qualitative methods in empirical studies of software engineering", IEEE Transactions on Software Engineering, vol. 25, no. 4, pp. 557-572, 1999.

[23] J. Horkoff, E. Yu, "Interactive Goal Model Analysis For Early Requirements Engineering", Requir. Eng., vol. 21.1. pp. 29-61, 2016.

Appendix A.

Interview protocol

Interview procedure:

Background on the interviewee: (max. 5-6 minutes)

1. How long have you been working with gaming industry?
2. How long have you been working with current company and what role do you have in game development

If possible and appropriate, describe working experience with creating games

Requirements gathering: max 5-6 minutes

1. Is the way you come up with requirements and early ideas for a game the same for each game, what process do you follow?
2. Is there any supplementary tools that you use for that process?
3. What are the strengths and weaknesses with those processes and tools?

Creative leaf: max 25 minutes

Describe Creative Leaf. Show Creative leaf. Describe main functionality and features tool provides.

1. Describe the tool, and the techniques. + modeling framework.
2. Show the tool with a preloaded goal model. And describe it.
3. Expand on the model a bit and use one creativity technique.
4. Is the model more or less understandable? Is anything missing in the model?
5. Do you think the tool might be useful or not for game development? Why or why not?
6. What changes would you want in order to make it more useful for game development?
7. Any other feedback?

Appendix B.

Follow-up interview protocol

Interviewee: *Fill in the name here*
Interviewer: Ilya Shabalin and Oliver Kjellman

Interview procedure:

Interview overall: max 10 minutes

We made some concept for changes based on our interviews, we will now show them to you.

1. What do you think about them?
2. Which of those changes would have the most impact?
3. Would you now use the tool, if not what else could be improved to make it more useful for you?

Appendix C.

Table #6.

Code	Margin Note	Quote	ID
Approval	Could be useful	“But yes this kind of tool could be used”	3
Approval	Could be useful	“I think this is a tool that can more or less be used depending on how deep you go with it”	4

Approval	Structurizing and mapping development	“I think it is also good to be extremely analytical and map up everything you have”	5
Approval	Mapping up features	“The visualization looks interesting actually. Like mapping up your features and so on to make it clear and visible to everyone”	8
Disapproval	Don’t need more ideas	“We don’t need help from a data program to get more ideas”	2
Disapproval	Other companies don’t need more ideas	“I don’t think any of the companies I work with would actually use this, because we make PC and console games that take a lot of time. And we have many ideas people”	2
Disapproval	Brainstorming without analysing	“For me it’s good to just improvise and brainstorm new things without really having the analytical point of view”	5
Disapproval	Not helpful	“You need to have more selling point… I don’t feel like it would help me, at least not that much that I would use it instead of just writing notes on a paper“	6
Disapproval	Not helpful	“I’ll have a hard time seeing the value of using it instead of using a whiteboard.”	7
Improvement of ERE process	Priority	“I think being able to prioritize would help a lot”	4
Improvement of ERE process	Connection between nodes	“Starting from the centre which is the customer and building out would make sense so maybe another way of thinking how to connect the nodes would be interesting”	4
Improvement of ERE process	More specific terminology	“Also, take note the vocabulary, place, help, help is very general. Maybe, you could use more specific language, in what way does it help. So it is not just help”	5
Improvements of ERE tool	Multiple viewpoints and more agility	“Yeah and then somehow link them together so that they lead or direct so the publishers and producers can look at the whole picture. Those are the things you need to think about I think, being able to update it and who would look at it”	1
Improvements of ERE tool	Better way to represent the core of the game	“But you could see the core game mechanics in this, then I could use it not for making the game but to showing what my ideas are to others. Not to get my ideas from it”… “I would like to be able to better see the core of the game”	2
Improvements of ERE tool	Description boxes	“The thing that would be missing is descriptions, opening up a box and see why you need voice acting or ideas about it, like people you want voice acting”	3
Improvements of ERE tool	Connection to other tools. High to low transition help.	“Something that would be really cool, is to link this to the bugtracker, attack would be linked to the feature task in jira or utrack that actually ask the software developers to create this. Implementation from the high level design to the low level tasks, it would be cool if you could link it”	3
Improvements of ERE tool	Tag system	“Maybe by using a tag system or something similar”	4

Improvements of ERE tool	Customer need	“You could use the customer as a starting point, you put out a note the customer wants this and that, and based on that you make the game loop and this part of the loop fulfil that, and that part fulfil this”	4
Improvement of ERE tool	Headline	“I think it would be the headline saying what is it here we’re going to solve”	5
Improvement of ERE tool	Verification of ideas	“Yeah, for our company we had a million ideas, and the main thing that affected our decision was if we could find some players, some people that were actually interested in this”	6
Improvement of ERE tool	Verifying the market need	“The main issue is what potential customers thinks and to be able to test those towards customers, you need to make a big prototype, a virtual slice and it takes a lot of time”	7
Improvement of ERE tool	Help transitioning from high to low level	“The transition is the hard part because I need to now go back, find my photos of whiteboards and documents and try to transition those into next step and that is the tricky parts. If you can help with that step that will be a value proposition”	7
Improvement of ERE tool	Transitions into sprints	“If that could become easier, from documents to sprint then it would be valuable”	7
Improvement of ERE tool	Description boxes	“I think it (right clicking to get a description) would be really helpful like how does everything work together like how was the flow”	8
Improvement of ERE tool	Importing images	“If it would be possible to like expand on something, then I would like images and so on”	8
Improvement of ERE tool	Redirect to image library	“Double click and you would be redirected to a library which will have different images”	8
Improvement of ERE tool	Possibility of adding Moodboard	“Add a mood board, same for sounds … the path finding or visualization of the path”	8
Improvement of ERE tool	Description boxes	“The problem is you need to show all the functionalities in order for it to be useful but of course you don’t have to go into details. So if you have the ability to double click something and open another graph this will be good”	9
Improvement of ERE tool	Cooperation with colleagues online	“I would like a collaborative function where me and my colleagues can use it simultaneously. Drag and move and add. Color code each of us”	10
Improvement of ERE tool	Graphics	“Tools can work but why isn't there more flair given to the actual graphics. You only have one chance to make a first impression, if it's pretty it's more likely to hook me”	10
Strength of ERE process	Visual representation helps communication	“Something like this I can see the benefit of having something very visual because it will help communicating between different people”	1
Strength of ERE process	Visual representation helps communication	“I could use this if I was presenting my idea to other people before it was built and needed a visual way to show it. But then you have to design this and have a designer to make it”	2
Strength of ERE process	Visual, More agile than Game design document	“Custom icons can help you visualize and help document what you have, this could yield a game design document, but more the dynamic style, because if you write a GDD (Game design document) you have to maintain it, and documentation is out of date as soon you write it“	3

Strength of ERE process	More expressive than Game design document	“We try to keep them (Game design documents) small and mostly high level and this kind of tool could help in being more expressive in that sense”	3
Strength of ERE process	Visual representation	“The thing I like is that it expresses the ideas in a very visual way”	3
Strength of ERE process	Visual representation	“I think it’s a very interesting way of looking at it this visual sort of node based, it makes sense when you are planning”	4
Strength of ERE process	Mapping	“This model could perhaps help map up all the different tasks and you could go on with the discussion... Maybe that’s the good way to apply the model, like start from the concrete example for a problem and see if you can help solve it in some way”	5
Strength of ERE process	Seeing bigger picture	“When you are developing something, you don’t necessary have that big picture, the overview. So it’s easy to miss out on certain things. But if you have everything on a map and you label it then you get a structure”	5
Strength of ERE process	Mapping gives new ideas	“I really believe in the idea of mapping things up and trying to get a structure because then you feed your thoughts with new angles”	5
Strength of ERE process	Overview of the system	“But just to have a quick overview of how the system works this is kind of helpful specially if you add new developers to projects or add additional designers”	9
Weakness of ERE process	Not iterative enough	“And something like this, it can be really helpful, but it also puts people off because it feels very structured for a lot of people, and so much of games are freeform and iterative, the actual idea of requirements capturing is quite alien to the industry”	1
Weakness of ERE process	Conflicting terminology	“Not really but I work alot with unreal engine and in there an actor is another thing so having them use the same terminology for different things could be a bit confusing. It’s a very minor thing”	1
Weakness of ERE process	Terminology confusion	“What do you mean by task?”	5
Weakness of ERE process	Don’t need more ideas	“Coming up with ideas isn’t the tricky part, but actually verify the idea, if someone will buy the program”	6
Weakness of ERE process	Not easily convertible into sprints	“But how does this help organize myself because I am not sure how I can convert or how I can make a benefit from using this and then convert it into easily into sprints”	7
Weakness of ERE process	Too High-level view	“It looks fairly high-level so it still needs to break it down to be able to use it”	8
Weakness of ERE process	Needs more details	“Usually the biggest problem in the design process is not coming up with the overview of the design but with the details especially when the programmer comes into place”	9
Strength of ERE tool	Benefit for small fast game development	“I think maybe this could be useful for people who wants to make games fast. Like smaller games fast maybe”	2

Weakness of ERE tool	No need for more ideas	“Yeah I think with a big or bigger company which uses more time, you don’t need to get help with the program to suggest ideas because there is often very many ideas. That we need to like, not just randomize the ideas”	2
Weakness of ERE tool	Redundancy of Creative Leaf	“And I don’t think I would use my time to fill out people have fun, like everyone should have that standard”	2
Weakness of ERE tool	Level of detail	“If it’s used in too much detail I’m going to guess the overview is going to get flooded”	4
Weakness of ERE tool	Scaling, sorting and priority	“Maybe with scale it might become less useful. Because it’s difficult to sort, there is no real, well maybe there is, but some sort of priority system”	4
Weakness of ERE tool	Too messy	“You can go to a very deep level and put every single feature in there but it would be a clusterfuck”	7
Weakness of ERE tool	Not agile enough	“I'm not sure that if I use your tool it's going to be agile enough for the dialogue of the words between the colleagues and me. And after the end of a long talk we usually end up somewhere radically different and I am not sure that this tool can adapt as quickly as we can”	10
Weakness of ERE tool	Only one user can update model at a time	“If I am the only one moving the bits around it's not really an interactive process for the rest of the team”	10
Current tools	Moodboards, inspiration wall	“We had Moodboards and Ghant and pictures and an inspiration wall”	2
Current tools	Whiteboard	“I don’t know, whiteboards”	4
Current tools	Whiteboards, pictures and concept art	“We had a whiteboard and some notes and pictures and concept art”	6
Current tools	Google	“I …use google docs to write stuff down when I’m making character sheets and the map of the story I made in google images”	8
Current tools	No tools	“I cannot remember us using any tool like this, any third party tool”	9
Current tools	No tools	“No, I don’t use any tools. I do a lot of thinking before hand, the team sit down and we talk rigorously then we figure out what’s important and what is vital”	10
Current ERE process	Lack of process, defining the vision	“No, that's more of a definition of vision that happens”	1
Current ERE process	Prototyping	“Our process is that we start with a prototype we just want to try out ideas, we have concepts we want to explore and make a prototype, we gather ideas for how to expand it and how to polish it, what kind of theme, and we start development and creation from there”	3
Current ERE process	Prototyping	“What we do is rapid prototyping so we put maybe a week, half a week to come up with a prototype and put up on free services platform”	4
Current ERE process	Market verification of ideas and prototypes	“We… had some suggestions, prototypes, ideas and such. We didn’t have a very controlled process, we sat and talked pros and cons about them and then we went for one and tried to verify it against the market”	6

Current ERE process	Elaborate ideas, prototypes	“The first one is where someone had an idea for a very long time they have been contemplating how to do it, why to do it, it’s been long brewing and it can be quite obvious that it’s an idea that we want to pursue for a prototype”	7
Current ERE process	Brainstorming	“The way we did our current game, is quite much opposite, we took brainstorming sessions, took a lot of ideas from them, picked the best ones, pitched the best ones towards potential partners and went ahead with the most promising idea”	7
Current ERE process	Design thinking	“We used a lot of design thinking, methods and influences from design thinking”	7
Current ERE process	Researching procedure	“Mapping out the storylines and I started writing… I looked into which software I should use as well”	8
Current ERE process	Feedback gathering	“I tried to talk a little bit to people about their experiences so I got stuff right but also like what”	8
Current ERE process	Prototyping	“No, I don’t use any tools. I do a lot of thinking before hand, the team sit down and we talk rigorously then we figure out what’s important and what is vital”	10
Weakness of current ERE process	People	“I think the most obvious weakness with that part of process is that it’s basically the people in the room. In our case, our company. In the large company, some kind of evaluation group that decides what is the good and bad idea.”	7

Appendix D.

Margin Note	Number of Appearances	Example quote
Description boxes	3	“I think right clicking to get a description would be really helpful, like how does everything work together like how was the flow”
High to low transition help	2	“The transition is the hard part because I need to now go back, find my photos of whiteboards and documents and try to transition those into next step and that is the tricky parts. If you can help with that step that will be a value proposition”
Verification of the customer need	1	“You could use the customer as a starting point, you put out a note the customer wants this and that, and based on that you make the game loop and this part of the loop fulfil that, and that part fulfil this”

Verification of the idea/market need	1	“Yeah, for our company we had a million ideas, and the main thing that affected our decision was if we could find some players, some people that were actually interested in this”
Connection between nodes	1	“Starting from the centre which is the customer and building out would make sense so maybe another way of thinking how to connect the nodes would be interesting”
More specific terminology	1	“Also, take note the vocabulary, place, help, help is very general. Maybe, you could use more specific language, in what way does it help. So it is not just help”
Multiple viewpoints and more agility	1	“Link them together so that they lead or direct so the publishers and producers can look at the whole picture. Those are the things you need to think about I think, being able to update it and who would look at it”
Better way to represent the core of the game	1	“But you could see the core game mechanics in this, then I could use it not for making the game but to showing what my ideas are to others. Not to get my ideas from it … I would like to be able to better see the core of the game”
Tag system	1	“Maybe by using a tag system or something similar”
Headline	1	“I think it would be the headline saying what is it here we’re going to solve”
Transitions into sprints	1	“If that could become easier, from documents to sprint then it would be valuable”
Importing images	1	“If it would be possible to like expand on something, then I would like images and so on”
Redirect to image library	1	“Double click and you would be redirected to a library which will have different images”
Possibility of adding Moodboard	1	“Add a mood board, same for sounds … the path finding or visualization of the path”
Graphics	1	“Tools can work but why isn't there more flair given to the actual graphics. You only have one chance to make a first impression, if it's pretty it's more likely to hook me”
Cooperation with colleagues online	1	“I would like a collaborative function where me and my colleagues can use it simultaneously. Drag and move and add. Color code each of us”

Appendix E.

Code	Margin Note	Quote	ID
Approval	Trackers	“Because last time we spoke, I expressed the need for integration with Pivotal Tracker or Trello, so that’s good”	7
Approval	List of components	“ I see a use for it, because it’s not uncommon that you plan something and try to figure out what I need to go to make this work, what components are needed here	7

		and try to scope it and it’s really difficult to cover the things and not forget components in the whole thing”	
Approval	Hidden workloads	“So for detecting those hidden workloads it would be useful if you could see and plan all the dependencies”	7
Approval	Similar to Google Docs	“It's like google docs benefits, that makes it useful. I don't really know how useful”	6
Approval	Trackers	“I like that you added so that you can make tasks that you have a task manager and you move it to a tracker”	6
Approval	More simplified	“I think it looks way more to understand now, so I like that, it’s simplified, compared to the version I saw before”	5
Approval	Layering	“I really like layer function and the graphic view, because now that I work like with game design and game development there is a lot of text, and you have them in a lot of places, and pdf and next you don’t have them in one place, but here you have them in layers, you can flip from one to another, and that’s way more easier to understand, so that’s a great feature”	5
Approval	Importing images	“I liked to add images, and just easily to throw in design”	5
Approval	Tool helpful for all workers	“Main benefit I think would be for a team, I am not a programmer, when I click this the programming box and you programmers put in data in a way that a manager like me can understand, the same goes for programmers I guess”	5
Approval	Happy about Trello	“They are an improvement in the sense that there is a clear roadmap on how the tool actually promotes better development practises later by putting things in this Trello thing”	10
Disapproval	Benefit not significant enough	“So I see benefit with your solution, but I don’t see overwhelming benefit”	7
Disapproval	Should be much better than whiteboard	So you need to have a same thing your program, whiteboard need to be s**t compared to your system.	7
Disapproval	New system	“I think there is a big barrier, when it comes to programs, … but when it comes to new tools, I need to have a pretty big improvement to transition and actually learn the tool, because I actually have no idea, how this works”	7
Disapproval	Can be used inefficiently	“There is a lot of ways to use your tool inefficiently, there are many ways you can f**k up this tool, creating too many layers or not understand what will be scrapped”	10
Useful	Tool useful for starting new project, Would try it.	(So if you had a tool, would you use it for your next project for example) “Sure, I mean would put all my data in it and see how it would change my view on a project and see, how the other team members, how we can more easily understand each other's viewpoints”	5
Useful	Communication	“ I think it’s a good forum or arena to facilitate communication”	5
Useful	Graphical	“I can really see the more meticulous person get use of this… It’s a good jumping off point, showing a graphical representation after a verbal assessment of the project, It’s free from people forgetting stuff and such.”	10

Not useful	Not more useful than sketching	“I just try to think how and what way would this be really useful compared to just sketching. I don't mean to be rude but I don't really now. I can't really see it useful”	6
Not useful	Easier to sketch on paper, Wouldn’t use it	(So you still wouldn't use it?)“No, i don't think so… I think maybe it's just me but it’s easier to just sketch on a paper like mind maps are.”	6
Not useful	Benefit not significant enough, Wouldn’t use it	“The program needs to add a significant benefit and the question is what are those benefits”	7
Not useful	Wouldn’t use the tool	“No because it’s fundamentally the same thing but better. “You want to have pizza?” “No I’m not in the mood for pizza”, ”But we made the pizza better” “I still don’t want to eat pizza”. It’s not that it’s not good or better, but it’s still what it is.”	10
Not useful	Extra work	“It’s just work”	10
Not useful	Too slow	“...thinking about what we are going to do, when we are done with that meeting, the last thing I want to do is to slow down, and do this.”	10
Not longer ERE	Task manager	“As i remember it the tool was used to help with ideas, now i get the feeling that is more like task manager”	6
Not longer ERE	Project overview	“So this is like a project overview?”	5

Appendix F.
Interview with Company 1.

Q:What company do you work for?
A: I work for Company 1 as a gameplay programmer.
Q:How many years of experience do you have?
A:Six in the gaming industry and a few more regular software engineering.
Q:And how much experience do you have with your current company?
A:Ah, six years.
Q:When you develop the games do you follow any procedures?
A:We do have various procedures built around agile principles
Q:What kind of extra tools do you use for this process
A:Jira, and extra bug tracking tools shared because we have external publishers and we need to share that with them so we use their systems.
Q:Do you have a process of early requirements engineering.
A:No. That’s more of a definition of vision that happens.
Q:Do you see any strengths and weaknesses in the processes and tools you use.
A:That’s really tricky since it’s such a messy process like, yeah there are definitely pros and cons we have problems getting the team working the tools we have but, it’s such a deep complicated topic.
Q:Our main concern is to develop a tool that will be helpful for people in the early stages of development. <Shows models> In this modeling tool you can model your company, your plans for what you want to get, good reviews, build a community, how do you get that, well you make a good game. You can also see it from the players perspective. Having fun, completing a game, this is like a multiplayer army vs army game. They want to have fun and you get a target audience basically and after that you can model your engine for example you can say what kind of resources you would have, say virtual reality, databases whatever. And then you can generate a task that will generate and help the players, and then you can model the game, what kind of soft goals you have and add the tasks. We are trying to see what is missing. To make this as effective and useful as possible because in the gaming industry people don’t use modeling a lot. What you get is what do game developers want to have, what is useful what do you like about it and what do you dislike?
A: I’m kind of a fan of visual, flowchart and state machines to show relationships between things, I found using that in games hard a lot of times. What you have is a lot of different people from different technical backgrounds. And the difficulty is often getting information in a way that people can understand. And something like this, it can be really helpful, but it also puts people off because it feels very structured for a lot of people, and so much of games are freeform and iterative, the actual idea of requirements capturing is quite alien to the industry. Previously I was working as a software engineer in the aerospace systems, with very very specific requirements and specific what to do and very structured. I’m not seeing that in games. A big chunk of it is that things change so often and so rapidly. So thats something to look into is the iterative nature to it and how quickly can you adapt. People have abandoned game design documents because they are so hard to maintain. Everything something changes someone has to go in and change it and once it stops being up to date it becomes harder to update it and it gradually gets left behind. Something like this I can see the benefit of having something very visual because it will help communicating between different people. But you probably want to focus down on iterating it quickly.
Q:This is an example of a very small scale.
A:Yeah exactly. The other thing to do is to work out who in the team this is for. Is it for programmers? Designers? For producers to talk to their team internally or people externally? And some different views. Some of the stuff here is like market focused and lot of people in the dev teams don’t want to get involved with

market things, that's more of a producer thing. This stuff is more of a design thing, and this is a tech things. And of course there are interdepencies but having everything in one graph might be abit too much. Might want to think about all the different viewpoints and angles with different needs.
Q: Yeah for example if we have 4 different tabs one is for the developer and such
A:Yeah and then somehow link them together so that they lead or direct so the publishers and producers can look at the whole picture. Those are the things you need to think about I think, being able to update it and who would look at it.
Q:Anything else?
A:Not really but I work alot with unreal engine and in there an actor is another thing so having them use the same terminology for different things could be a bit confusing. It’s a very minor thing.
Q:Okay, thank you

Interview with Company 2.

Q:Can you please introduce yourself and tell us what company you work for and how long you have been working with that company?
A:Yeah my name is *** I’m the CEO and programmer for Company 2. We have had the company from september 2014 so two and a half year.
Q:When you develop games how do you come up with the requirements for the game? And with requirements I mean what kind of value will it have, what things can you do in the game? What is the process of developing your game?
A:We started on the game when we were still in university where we took our bachelor degree, and we took inspiration from an old mod called warlock for warcraft 3. And we wanted to make it a cool stand alone. And we wanted to add more things, build upon the success.
Q:Did you follow any process or was the way you did it more random?
A:It was a school assignment so we had to write like a report and report to your teachers every week, so it was like a school process. I don’t know if we would have done it the same way if it wasn’t a school project.
Q:So you basically followed that idea from school and you built a company and then made the game?
A:We started the company after school and wanted to make more out of the game. We wanted to release it. But we used likeFhant
Q:Did you use any extra tools in the planning stage? Did you actually plan that much like follow a process for planning for how the game will look?
A:Well we had a designer who painted, we had moodboards and Fhant and pictures and an inspiration wall. Just so we could show the teacher.
Q:Did you like anything in particular out of these?
A:The mood board was fine.
Q:Did you like anything in particular about that, the moodboard, I’ve never heard of it.
A:Just take a lot of colors and plan. The colors you are going to use and put it on a white sheet and see how it looks.
Q:Alright, so our study is that we are trying to model a game from the perspective of requirements engineering. And here it’s basically in the early stages of the game. <shows model> This is a model of a game, it’s a multiplayer army versus army battle in virtual reality. So for example you can make a model for your own company, what are your ideas, what do you want to generate, you can say that you want to get good reviews or build a community for example, how you do this by making a good game. After that you can model what kind of players you want what is your target audience? Basically what they want.
A:Yeah we have a lot of reports I have 20 pages of reports about target audiences for when we applied for funding. And we have all of these things.
Q:If you have problems coming up with ideas there are a lot of creativity techniques. You can use extra tools like creativity triggers. <didn’t work> Anyways, They can help you come up with ideas. From a technical perspective what kind of resources you will have how they will help you, you can also “generate” tasks, let’s say that we need to generate a map for this game, it will use some engine for that and it will help people complete a match and we can see what kind of functionality the player will have so you will need to control an army by moving units. After that you can kill the opponent's units to win.
So we are trying to see if this is useful for gaming industry and if not, how can we improve this?
A:I think maybe this could be useful for people who wants to make games fast. Like smaller games fast maybe.
Q:That’s interesting because the people we interviewed from small startup companies they actually said that it’s too not complex but. It’s quite redundant because they don’t really plan that much.
A:Yeah I think with a big or bigger company which uses more time, you don’t need to get help with the program to suggest ideas because there is often very many ideas. That we need to like, not just randomize the ideas. But I think if it was a company with just random ideas put them out and put different levels maybe it could be useful.
Q:But if you would like to use this tool, what would you want to have inside it. What extra functionality would you like to have? So that it’s going to be useful for game developers?
A:I don’t know, like making a good game is always. You always need to do that. And I don’t think I would use my time to fill out people have fun, like everyone should have that standard. “This” is maybe the most helpful but I don’t think we would have that with our game since we don’t need any ideas, we don’t need help from a data program to get more ideas.
Q:This is a small perspective game, we also made a model for a big game like need for speed for example.
A:But I like graphs and stuff like that so if you add.
Q:When we have a big model, unfortunately we can’t load it now, it actually get’s quite complex, so we have a lot of actors which will actually connect with each other, do you think this is even useful for the gaming industry?
A: I don’t think any of the companies I work with would actually use this, because we make pc and console games that take alot of time. And we have many idea people.
Q:Would the people that come up with the ideas find this useful.
A:I don’t know, maybe I could use this if I was presenting my idea to other people before it was built and needed a visual way to show it. But then you have to design this and have a designer to make it. But you could see the core game mechanics in this, then I could use it not for making the game but to showing what my ideas are to others. Not to get my ideas from it.
Q: So if you want it as just a presentation tool do you want to change something?
A:Yeah I would like to be able to better see the core of the game.
Q:Thank you for your time.

Interview with Company 3.

Q:So first of all can you please introduce yourself and how long have you been working in the gaming industry?
A: Sure. So my name is *** I’m a member of a swiss studio Company 3, 2 developers 2 artists and one who is kind of in between. The studio was founded 2014 so it’s three years old and have released two mobile games and are now developing a pcs and console.
Q:The games that you talk about how do you come up with the early requirements, do you follow any process for that?

A:Our process is that we start with a prototype we just want to try out ideas, we have concepts we want to explore and make a prototype, we gather ideas for how to expand it and how to polish it, what kind of theme, and we start development and creation from there.
Q:Do you use any tools for building the prototype?
A:Unity
Q:More managerial.
A:Google drive to collaborate, trello to manage the tasks and bug trackers.
Q:Do you see any strengths or weaknesses in those tools.
A: Yeah they all have their uses, trello can't link issues and doesn't work as a bugtracker so we have to use a bugtracker too.
Q:So we are trying to see if modeling tools can be useful for gaming development, so what we do is model the games, we can see the company, the goals of the companies and players and how to get there <Shows the models>, the characters general idea and attributes to tell the story and defeat demons. Do you think you can see any usefulness for the gaming industry and if not how can we improve this.

A:From what I see here this is a mind map like xmind, freemind, but with custom icons. I think you could use mind mapping tools to get to this. Custom icons can help you visualize and help document what you have, this could yield a game design document, but more the dynamic style, because if you write a gdd you have to maintain it, and documentation is out of date as soon you write it.
Q:I heard that game design documents are like 20 pages and a small change and you have to change the whole thing.
A:We try to keep them small and mostly high level and this kind of tool could help in being more expressive in that sense.
Q: How can we improve this tool what do you think it's lacking?
A: To me I could have the same functionality from a mindmapping tool, the thing that would be missing is descriptions, opening up a box and see why you need voice acting or ideas about it, like people you want voice acting.
Q:You can do it through another process of soft goals and tasks, is it too complex. Is it too much to describe it as tasks and soft goals.
A: It makes me thinks of something else, something that would be really cool is to link this to the bugtracker, attack would be linked to the feature task in jira or utrack that actually ask the software developers to create this. Implementation from the high level design to the low level tasks, it would be cool if you could link it.
Q:Anything else?
A:The thing I like is that it expresses the ideas in a very visual way, that's cool. I'm a bit curious about the software stack for this.
Q:It's made by a university and I think it's made in javascript. We are trying to improve it.
A:But yes this kind of tool could be used.

Interview with Company 4.

Q:Can you introduce yourself and what company do you work for and how long have you been working within the gaming industry.
A: I'm a 3d artist at Company 4. It's an indie studio in *** in Skövde. Our company makes small experimental games so at the moment we are working at *** which is a starving artist simulator.You paint paintings, put them up in your gallery and there are customers coming criticizing your paintings. And I've been working with flamebait for a year and a half, two years maybe so I'm still pretty fresh.
Q:What game did you do first?
A:This game is actually our first game at our company. The first game I made was in a game project in university of skövde. Which was a complete fail.
Q:How do you come up with the initial requirements and ideas for your game, because they seem to be quite creative?
A:What we tend to look at is since we are a company with 0 budget and 0 anything we try to look at how we can get the game to market itself as much as possible without us doing much so. What we do is rapid prototyping so we put maybe a week, half a week to come up with a prototype and put up on free services such as gamejolt or where people can download it. And then we look at how much the game spreads. Passepartout for example had a pretty high virality factor. Let's players enjoy making paintings and people enjoy watching people make paintings so the game made sense. So that's our main goal to get a game with high spreadability so we don't really need to market aggressively ourselves.
Q:I see, did you follow any process when you were developing the prototypes?
A:We do have certain standard procedures like scrum meetings in the morning, Slack. What do you mean when you say process? Do you mean like pipeline process or communication or.
Q:Sort of, do you use any tools to structure the prototype?
A:Yeah, we use pivotal tracker which is awful in my opinion.
Q:Any drawing tools to visualize your prototypes? Like to draw it a bit.
A:I dunno, whiteboards.
Q:Okay
A:We are not at that level, we are pretty low tech when it comes to planning.
Q:Do you see any strengths or weaknesses of any particular tool?
A:The things that are nice about pivotal tracker is that it gives you an estimate of when about you should be done at the current pace you are working at. The problem is based on an arbitrary system of points. <Small talk about this> So the biggest issue is time estimates.
Q:Now I will introduce you to our tool, what we are basically trying to do here is that we look from both the perspective of the company and the player so first of all let's say that you will develop a new prototype, you are going to try to see what the player would like to do, this model is not for your game unfortunately we didn't have time to customize. But this is a turn based RPG with a nicely looking weird graphics. Or this is another model damn. It's basically an army fight in virtual reality. So people would like to compete in a match and they would like to have fun,the company wants to get good reviews and build a community to make a good game. After you prototype this you can decide what types of resources like virtual reality for this example and after that you can also see what kind of tasks you will have for example generate a map, generate resources to fight for and things like that. And inside here you can see what a person wants from the game. How to win, and the connections to get there. Do you think this could be useful for the gaming industry, especially for small or big companies?
A:I think this is a tool that can more or less be used depending on how deep you go with it. if it's used in too much detail I'm gonna guess the overview is going to get flooded.
Q:I can show you one of them, need for speed it's not in this folder damn.

A:So my gut reaction is that maybe with scale it might become less useful. Because it's difficult to sort, there is no real, well maybe there is, but some sort of priority system, like a way too -
Q:There is no priority system
A:It would be interesting for example these tasks are super important these needs to get done. Like MVP sort of level stuff. "This?" is good to have but we can cut it out. Maybe by using a tag system or something similar, "this" is a graphics related task and "this" is a design related task. Then maybe it might scale better if it's easier to sort. Yeah I think it's a very interesting way of looking at it this visual sort of node based, it makes sense when you are planning, when you base your

plan on sort of a game loop, so you can start with the game loop and then you can sort of build out from the game loop. What do we need to do for this part of the loop to work? And what tasks need to be done.
Q:Also in the process of developing the game and you need to come up with some extra ideas, you can use the creativity techniques for example brainstorming. Then the ideas will appear and the only thing you need to do is connect them. Also there are some more of them that can help you develop a thing or two. What else would you like to add to the tool, the general response is that it lacks usability. It doesn't have value, for example if we can connect this to value. I don't want to lead you but they are saying it has bad connection to the customer value for companies. So do you have any other ideas on how to improve this tool specifically?
A:Don't know really, I'm not a super planning master guy since I am very inexperienced, but I would, I think being able to prioritize would help alot, I feel the connection with the customer is, because this is like an in-house tool it's not meant for the public, so I don't know it depends on how you choose to use the tool, you could use the customer as a starting point, you put out a note the customer wants this and that, and based on that you make the game loop and this part of the loop fulfil that, and that part fulfil this. Starting from the centre which is the customer and building out would make sense so maybe another way of thinking how to connect the nodes would be interesting.
Q:Anything else?
A:I think this looks familiar what is it?
Q:It's istar and we are trying to see how it would work in the gaming industry.
A:Okay
Q:Thank you so much.

Interview with Company 5.

Q: So we actually thought how we can structure the project of Company 5 here so we can start it describing what resources we have on application. So we have, stories, sounds, and animations all of this will depend on the specific characters that you want to program or to show. In this case, our stories, sounds and animations will help people learn the new characters and you as a company and as a developer, you want to help people learn of course to interviews, and that will help you receive money or grants. As a user of this product, you want to learn characters and you want to have fun so you can also play single player or cooperation. We haven't really done this because we didn't really get exactly you'd like to have a co-op. Also, if you're doing the single player, animation and story will make you have fun while learning and also it will help you to learn the characters which is the ultimate goal of the person using this program. You can also listen to how the characters pronounce which will help you incase you're doing a cooperation, you will help others to learn and ultimately again it leads to the learning of new characters. So we're trying to see how can we make this useful and easy to learn for the person who has never seen this before, who is a game developer. A programmer can use this later on but our research is mostly on game developers. Because in here you're actually coming up with tasks, coming up with resources you need to have. Later on is one of the options you can show this to the developer but then it lacks a bit of like there's should be first step and second step. Do you have any feedback on this, how can we make this useful for the game developers?
A: You could turn the question around and ask the developers like what problem do you have. Oh, and the game developers says I have this problem. In my case, for example, I want to do a co-op game which teaches pronunciation and then you start with a question with a problem and then this model could perhaps help map up all the different tasks and you could go on with the discussion with the question. Maybe that's the good way to apply the model, like start from the concrete example for a problem and see if you can help solve it in some way.
Q: Maybe even one of the new boxes in here would have an idea or assumption or a problem and after that we can allocate resources to solve the problem for example. Because this is used more in early requirements management where people don't actually say we will have this problem. However, if people already worked for something similar they might outline the problems right away and they can locate resources and goals to resolve the problem.
A: You could also say that all game developers has a problem, earn money or get users at a very basic levels. Create game, little bit higher up and then it goes back and forth like we're at the development process. I really like the idea of structuring up all the values or whatever you have in the project, the goals and the resources. When you are developing something, you don't necessary have that big picture, the overview. So it's easy to miss out on certain things. But if you have everything on a map and you label it then you get a structure.
Q: Do you have them in specific order, like from top to bottom? The way that you think or is it just randomly.
A: No, it's randomly. We tend to think that these are the building blocks. From these you can make different types of games and then you have inputs of it. As a user, you can type with keyboard or you can write and speak. You can interact with your finger. You can make these in all kind of different ways to make a game.
Q: The whole framework from my perspective, lacks a bit of verb and actions structure. If we're going to describe what actions are we going to do, which will help him discover a story. But is it a good or a bad representation of an action for you or complicated instead of writing the useable top, create a task. Is it too complicated or not really? If we can just write top or speak, instead of creating an absolutely new task here, what is more convenient for the game developers?
A: What do you mean by task?
Q: We can have different task like the user would click top to listen. Like listen to story together since it is going to be co-op, help us learn new character. I'm kind of wondering if this is a good way to present this to game developers thru task which will help.
A: It depends what you want to do, as you see the animation soundtrack for example. You just listen to the sound of the animation, you don't see anything. You start to create images in your head, which reminds you of the story, and which reminds you of the character. So there can be one way audio, or the other one, pronunciation. I think we have to contextualize it more. If the question is better or not, it depends on what we're trying to solve here, what is it, what concrete questions do we have here. It's a way to map up the game structure but it's a little bit big.
Q: So this is the program right, how to learn pronunciation? We would have to think how to solve this problem and we input tasks and soft goals. Like the way how to get there, how to actually learn it. Maybe it would be listen to the sound and the correct pronunciation. Learn characters is very broad.
A: We can say that this means learn.
Q: If we're going to have specific tasks, then we need to break this down in almost seven different task, because the character radical pronunciation, animation. The user wants to use all of this to ultimately learn a new character. This model will be again more complicated the same as need for speed model that we did and showed you before.
A: If this is the goal, this is like the tools. The content on the left side and the actions on the right side.
Q: If we want to represent actions and content, we do have a resource which is a good thing but do you think it is representation of an actions, if we do it through tasks and softer goals or do we need this specific new structure which is gonna be called actions. Is it necessary?
A: The way I think is, what actions and resources, what do you have to play with when you design a game. For me it's good to just improvise and brainstorm new things without really having the analytical point of view. In the other side, I think it is also good to be extremely analytical and map up everything you have. Then comes platform. The single player and the co-op game. I guess that's what your model is trying to do, map everything up.
Q: Do you think it is useful for the game developers?
A: Yes, for sure. In a way that makes sense, I really believe in the idea of mapping things up and trying to get a structure because then you feed your thoughts with new angles. With that, you come up with new ideas and you can go back and just think of it and try to be creative for awhile and you try to go back to this model and see. I like the idea of going back and forth to different perspectives because it makes you more creative.
Q: Do you like or dislike specific parts of this? What would you like to change to make it more useful and what would you like to keep?

A: I think it would be the headline saying what is it here we're going to solve. Like over the whole thing, is it interaction design, is it a game? What in the game, is it the part of the game or is it the whole game? Because you can say we need to build a leveling structure so that can be a problem and then you map up things. But it could also be retain the users, how do we get the users from coming back to the game. Some kind of a headline that clearly states what is it that we're trying to solve.
Q: If you have like three different tasks. First task is to get the users, second is to retain the users maybe line up different task.
A: If you have one map for each problem then maybe it gets less congested. Also, take note the vocabulary, place, help, help is very general. Maybe, you could use more specific language, in what way does it help. So it is not just help.
Q: In the whole I* framework, they don't have that many verbs. I think for the gaming industry down we can explore to extend this to make it way easier for the requirement engineers to do this because otherwise if you look back, if you look at this model a day later, you will just see help and you might just forget what exactly does it do.
A: Another thing is, it's important that if you're going to use this. Don't do the analysis yourself from looking at the game. Leave all this blank and bring the developer and do it together from start. Because then, I the user of this, I will be engaged with this from the start and I will feel that I'm a part of it, like a good way to build engagement. Otherwise, it's like a lecture, you're trying to understand but you're not a creator yourself.
Q: The problem then would be the time for developers to gather.
A: Then use an example from their everyday life it becomes way more easy to understand.
Q: Any more final feedback for this?
A: No.

Interview with company 6.

Q:How long have you been working in the game industry?
A:About two years at least professionally.
Q:What did you do before?
A:Some hobby projects during summers and so.
Q:So how long have you been working with your current company and what role do you have?
A: From the start, almost two and a half years ago, we are actually three people and have a lot of stuff we need to do, but basically I am a graphic designer and a level designer and do some UI and administrational things.
Q:So when you make a game, like when you decide how it will be in the beginning, do you follow some kind of process?
A: We just sat there and talked and had some suggestions, prototypes, ideas and such. We didn't have a very controlled process, we sat and talked pros and cons about them and then we went for one and tried to verify it against the market, and it went well so we stuck with that one.
Q: is this your first game?
A: Yeah and it's a very big game. It's pretty risky but so far it's going good and we hope it continues that way.
Q:So for that process that you use, did you use any tools for it, not even whiteboards?
A:Well yeah we had a whiteboard and some notes and pictures and concept art.
Q:Since your game is quite complex was it really difficult to get everything from the start, how many extra meetings did you require to get everything.
A:For the first year we couldn't even play we just had to work on the engine, we use unreal but we need to implement a lot of features that doesn't exist there, so it took over a year to the core work before you could even see how it will look.
Q:Let's imagine in two years you are done with the game in case you want to do dlc how do you plan for it?
A: We already planned for it, we don't know if we will use them but we planned for them in the engine so they will be easy to implement.
Q:So the tools, what are the strengths and weaknesses of them?
A:Well I don't really know, I haven't really thought about whiteboards that way.
Q:Well so we have a tool called creative leaf, it's for helping you structure the early ideas and stuff, we are trying to see how we can develop the tool so that it will be useful for people who develop games, because right now it's not tailored for them but more for big software products so, we tried to find as much information we could on bannermen. So what we started from is the resources your game would have, yeah so these yellow ones are the goals you are trying to achieve, these are softer goals which you use to reach them in order to reach the goals. These are resources that are used to achieve goals. You can also plan what is your long term plan, so you make a game that will make you a recognizable company and build a community, you can also see the target audience here, what do they want they want to win. So after that you can develop and see how you can reach those goals, making people have fun and thinking strategically. So the player plays a kingdom, but you can model more, this is just a model for every strategy game, but would you say that something is missing from this?
A:Why is it better than just a whiteboard?
Q:Because it can translate to what programmers know and easily make user stories for agile teams and stuff like that, so do you have any suggestions? You can be brutal.
A:You need to have more selling point that actually explains why I should use this instead of a standard whiteboard. I don't feel like it would help me, atleast not that much that I would use it instead of just writing notes on a paper.
Q:So what would you want to make it better for you to use it, we also have some creativity techniques, brainstorming ideas and creative sparks where you put yourself in the mind of someone else, for how you come up with things. It could help you come up with ideas.
A:I think that the big issue when you want to design a game, coming up with ideas isn't the tricky part, but actually verify the idea, if someone will buy the program.
Q:So a market verification support?
A:Yeah, for our company we had a million ideas, and the main thing that affected our decision was if we could find some players, some people that where actually interested in this.
Q:Would it benefit from a money/time thing.
A:Such a tool would be very valuable but is it even doable?
Q:It will require different research, anyways thank you.

Interview with the Company 7.

Q: We are doing a study on the implementation of early requirements tool in gaming industry. We want to know for how long have you been working in the gaming industry?
A: About a year.
Q: The same company, or did you changed companies?
A: No, we've been together for a year now.

Q: How long have you been working with your current company? What role do you have?
A: I am the CEO
Q: What does that include for you?
A: We have 5 people in our company. In a company, there's a lot of administration to do outside the actual development process so I take care of everything, not development related that we need to do to move forward. Not everything, but my job is to make sure that the other guys are able to do their jobs without too much getting in the way. I do economics, administration, outside contacts.
Q: When you come up with a new game, what process do you follow during the earliest parts? What ideas do you follow?
A: There is basically two ways that a game or idea or a prototype for a game can come to life. The first one is where someone had an idea for a very long time they have been contemplating how to do it, why to do it, it's been long brewing and it can be quite obvious that it's an idea that we want to pursue for a prototype. The way we did our current game, is quite much opposite, we took brainstorming sessions, took a lot of ideas from them, picked the best ones, pitched the best ones towards potential partners and went ahead with the most promising idea.

Q: Is there any tool that you used for that process?
A: We do a lot of our in that state of development process you could say. We used a lot of design thinking, methods and influences from design thinking. Design thinking is pretty much a toolbox for coming up with ideas and have a large amount of creativity. Things like there are no bad ideas, everybody needs to be positive towards all ideas and a very important part of that is, there are no bad ideas because bad ideas could lead to good ideas. It's actually quite academic and we use them a lot and they work very well.
Q: What are the strengths and weaknesses for those?
A: I think the most obvious weakness with that part of process is that it's basically the people in the room. In our case, our company. In the large company, some kind of evaluation group that decides what is the good and bad idea. The optimal would be if you could take all those ideas and somehow prototype them or test them towards a target audience because it doesn't really matter what we think. The main issue is what potential customers thinks and to be able to test those towards customers, you need to make a big prototype, a virtual slice and it takes a lot of time. That's the greatest weakness, I think.
Q: We have a tool called Creative leaf and it's employing goal modeling. Do you know what that is? We checked your game briefly and we made a short model for it. ****Shows Model**** This is your company, these are the hard goals, this is what you want, get good reviews and build a community. As a player, he wants to have fun and be immersed and to complete a match. To achieve those, you build smaller goals like soft goals. These are resources to achieve goals as well and the tasks can be generated and you can just show the tasks to developers for example, in your game you want the ability to move, move units, you also need to generate the map, where the action would be. After that, you could see those tasks if achieved how will that lead into achieving your final goals. Is this understandable?
A: Yes, I think so.
Q: Do you think something else is missing from your game here?
A: It's a quite complex undertaking to try to map it but of course there are things missing. It depends on how deep you wanna go I guess. You can go to a very deep level and put every single feature in there but it would be a clusterfuck.
Q: There's also this creativity techniques on side here. Like for example, brainstorming and you write the ideas quickly and then you make them easier to model and then there is this one that let's you get into the mind of someone else, tries to follows on steps to see how that person would react to your software. Do you think this could be useful to the gaming industry?
A: Could you explain more about in which state is this software supposed to be used and exactly how?
Q: You use this in the early stages when you're thinking out the idea and you're trying to model it our more. So that it's easier to show people. It can also tell what kind of resources you want to have, what kind of ideas will the target audience have and what would you like to deliver to them and even outline some functionalities early on so you can actually pass on the developers small tasks.
A: Today I tried to organize what we do. We use services like trello and other things like that. This is a more broader sense for that because we can do things with this and put it into trello to take care of different things. But how does this help organize myself because I am not sure how I can convert or how I can make a benefit from using this and then convert it into easily into sprints. I guess that if it could be easily convertible from a broader perspective of the entire project into tasks that would actually help my work. As you said before, we have this documents were as, what is supposed to do, the scope and everything and then we try to keep parts of that and try to prioritize those tasks and put them into sprints. If that could become easier, from documents to sprint then it would be valuable.
Q: Currently it is not being used in the gaming industry, do you think there's an improvement to it that could be made?
A: I think it needs to focus on the value proposition for the actual user. For my perspective right now, I have this system here which could be very useful as a tool for visualizing the project and the scope of the project. On the other hand, if I don't have a direct connection towards what I need to plan in my sprints and the detail planning. I'll have a hard time seeing the value of using it instead of using a whiteboard. So if you're able to connect everything there you'll actually have a service that could be competitive with trello because you do everything they do plus this part, or in some way, connected to services like that. It could help the entire process rather than just one part of it.
Q: We think it's a bit more like an early requirements management that I guess that is quite a good feedback that we should move on from the early requirements.
A: Correcting stuff is the easy part. I can do it on a whiteboard, a paper, it doesn't matter. The transition is the hard part because I need to now go back, find my photos of whiteboards and documents and try to transition those into next step and that is the tricky parts. If you can help with that step that will be a value proposition.

Interview with Company 8

Q: Can you first introduce yourself? What company you've been working for and how long have you been working for the company?
A: The main thing that I'm doing right now is I'm an independent game developer at Company 8, my own company. I've been doing that for roughly 1 and half years, for that I was at paradox. Both other company and other companydevelopment studio.
Q: You have been working for how long in the gaming industry?
A: You can say 5 years. I did some freelancing gigs before that. 5 years like full time.
Q: What exactly experience do you have in creating games? Can you describe it a bit?
A: Right now I basically make everything in my company. Since I also freelance right now I have experience writing stories for others. When I was at paradox I work for QA so I start out as a tester then I moved on to manager position. So managing, testing and so on. I've seen a lot of development and narrative and QA.
Q: For the game that you're working on currently, how did you came up with the requirements and early ideas for that game?
A: I guess it started out with few topics that we talked about so I'm like, okay I have to talk about this to people so I just made like a list of stuff to talk about and started researching it and mapping it out. Basically, mapping out the story lines and I started writing, building assistance. Of course I looked into which software I should use as well. I tried to talk a little bit to people about their experiences so I got stuff right but also like what would you like to see in a game like this. I tried to talk to people about that.

Q: Do you use any extra tools that really help you?
A: I guess, I'd likely use google docs to write stuff down when I'm making character sheets, for the characters and the map of the story I made in google images.
Q: Now we can talk about the tool that we have. What kind of model would be better for you to show. There is a game called when you're gone. It's also basically telling the story of a person through a turn-based rpg. Unfortunately I don't have time to research the specific figures but I can make the models for your company. So basically what we're trying to do is to investigate if this kind of modeling requirements engineering is useful for gaming industry at all so I can quickly just describe you what's going on here. As a company we would like you to define your goals as a company. For example, you would like to intrigue people, you would like to get recognition, you want to connect to people because that game was quite touching it wasn't really a mainstream game, it's more like telling a story. I guess we can actually quite relate this to you. After that, you could also see what kind of resources would you have inside of the games. For example, if you have any beautiful story and beautiful pictures on this animation you can just describe it as a resource put it there and that will help you intrigue people for example. You can also decide what kind of resources you can manage and how they will help you, in process. After that here we modeled a player or is called sam in the game, he's supposed to tell you a story through his game. He's supposed to use his resources such as attributes to attach, to defeat the inner demons and voice will help to tell the story so you can attack and move. You need to basically win a battle, here you will have a key term and after that you can narrow down to specific tasks. So if you'll need defeat demons to win the battle, and if you're defeated, you cannot win the battle. We make models for quite different games, and we're trying to investigate if this is even useful. How can we improve this tool, so it's going to be more useful because we believe right now there is a gap in the gaming industry because people don't really use extra tools and we're trying to see if that's even useful.
A: The visualization looks interesting actually. Like mapping up your features and so on to make it clear and visible to everyone. Visualization tool looks cool, I'm just curious because it is high level, like attack is hard to design, something just based on a word attack and it doesn't tell me that much like going to more detail anyway.
Q: You can describe it as a software goal for example, right now we're actually quite lacking of the description of a task. For example would you think it's useful right click and then you have a big description.
A: Yes, exactly something like that. I think it would be really helpful like how does everything work together like how was the flow. But then I would actually need to make the attack what are the building stones of that, is it range or melee stuff like that.
Q: Right now, at the moment, is that we actually describe it as a software goal the problem is that we had a huge game that we tried to model games like need for speed triple A games and what we ended up is with amount of tasks and softer goals if we try to build everything down to the smallest bits. Anything else that catches your eyes, straight away?
A: You don't really use assumptions, right?
Q: The ideas would be generated from these activity tools and then you can have an assumption which will help you generate ideas. We are actually not focusing on creativity techniques at the moment. We are way more focusing on those 5 aspects. We might also need to add an extra block if you think about one. Right now you can have a goal, which is basically the goal of the game and you can have a task. What kind of resource the game would have and can be anything NPC. Assumption, I guess what we can do is that let's have we had unreal engine. That assumption would be that that engines stays top quality for the next 2 years.
A: If it would be possible to like expand on something, then I would like images and so on. Is that something that you've been considering.
Q: Not really, but now we are.
A: Because if I would want to use this as an actual game documents then I would definitely want to have images like references, that this is the visual style that we're going for.
Q: So, for example you would double click and you would be redirected to a library which will have different images.
A: Yes, something like that. Like being able to add a mood board, same for sounds. I guess that's kinda why when in documents stuff I usually do that as well. For example, this is quite the tracking systems or what the feature Is going to look like, the path finding or visualization of the path. In some cases that you would want to have reference even though it's not always just aesthetics like UI for example.
Q: Do you think it's even useful for the gaming development?
A: It's hard to tell just from a glance. It looks fairly simple so I'm kind of interested in seeing how "this" can tie into it as well. I have hard saying what I would be looking for in a tool like this. I guess it's interesting but it looks fairly high-level so it still needs to break it down to be able to use it, I guess that is my biggest feedback. Because otherwise I would still have documents in other places anyways.

Interview with company 9.

Q; Can you please say the name that you've been working in, what company?

A: I'm ***. I work as Engineer at Company 9 and besides that and I do lectures at University from time to time. Obviously teaching stuff about Company 9. So I can understand quite well the requirements or the situation of writing the thesis. I've also written my thesis about game development several years ago and I think I was one of the first students to graduate in game related subject in general. Like almost 10 years ago.
Q: What about game development? How much experience do you have?
A: Yeah, after University I shipped various games from small mobile games to video games. Successful mobile games scaled from 20 Million. I've worked on sacred 2 which is a AAA game in Germany, the budget was quite high, I know the number but I can't say. Besides that, we won some awards from time to time as well. But now at Unity, I'm being a Field Engineer at larger accounts in Germany and Nordics so that includes companies everyone knows and the likes will be the next games in Germany. So some really interesting accounts, and a lot of automotive customers are using Unity as well. So all the car makers, all the suppliers.
Q: Do you follow any process when you come up with requirements and early ideas for the games?
A: It has been really different. Right now of course I'm not involved in production anymore but back in the days we had various different approaches on the AAA productions of course there were strict processes. Even though the team was rather small like 30 people, it is really small. But at the smallest studios I've brought after these, the only process problem was that at some point there was some kind of prototype in the beginning and like a game design document was kind of drafted and after this, we usually approach the customer in terms of a publisher, so it's mostly work for hire. When the customer or the publisher was interested we would usually visit them to preach the project. Basically, then the actual planning and production starts which means lining out all the features and going quite into detail. We'll be able to add the numbers to it. In terms of, how much time, which type of people are needed to allocate resources and then calculate the costs. That is usually the approach, I don't know in detail how this works for larger triple A projects because I joined after the production started.
Q: In all these processes, did you used any supplemental tools? Anything in particular?
A: We usually did most of the stuff in either on tools or the classic stuff like Excel. I know that some teams in Germany successfully uses stuff that Navigo was doing, it's call RC drafts, I could see drafts that's quite popular especially students use this a lot. Apparently, I cannot remember us using any tool like this, any third party tool.
Q: Do you think anything in particular that you like about this tool, or dislike?
A: Well, myself didn't used a R2C draft But yet it's hard to differ. I think it's probably competitor of your tool anyways, I'm not sure if you know about this.
Q: So what you're trying to do is structure the process of early requirements engineering in games? I forgot the University came up with this tool where you can develop goals, tasks, account and resources and use actors. Right now we have models for different games from small companies to big ones. We have EA games model somewhere. So basically what we have in here is that company can outline what are the plans for the game and say that they want to get good reviews and

build a community for example and to do that they need to make a good game. Or for example some more abstract game, the ones that could get recognition and they needed to do that by intriguing people because they have quite some interesting visuals, it's quite unusual type of graphics that they used, creative. So what they have to do Is basically, they wanted to intrigue people not, just to get money. We can also define what kind of target audience looks like, what it wants to have, it wants to have fun, it wants to be immersed, it wants to complete the match then you can also outline a little bit more from technical perspective you can say what kind of resource you can have. For example, virtual reality working, working with some engines or some art style, sounds, anything that you want to have here. You can also generate a lot of tasks, which you can later on send to the developers and then you can also outline how the game play will look like. For example you need to move units to control army, if you control army while you kill the opponents army. So trying to see here is that, what is this tool if it's used of a gaming industry or not and it's not used full how can we improve this so it's gonna be useful.
A: It's really high level, well it depends on who should use this. It looks more like a strategic tool and but the thing is producers and game designers, if they do some kind of planning in such tool, like a visual tool, it's mostly focused on balancing stuff or general features and it will be quite a huge graph anyways so a lot of stuff can be done so it's like move, control and kill and wins is super super high level. There will be hundreds of elements and different combinations and parameters for basically ensure that the game play works fair depending on how you choose. Usually there's many ways to achieve one goal, usually there are multiple goals as well. Basically it means deciding which paths to go and which goal to follow and you have to balance how difficult each different path is and how rewarding each goals and there's probably like one supergoal who everyone will call it winning and losing. But of course there are different types of games so there can be cooperative games so people can have different goals or you might have 2 goals but they are conflicting. For example, you might like have a goal to win together but there are separate goal to kill your cooperative person as well.
Q: There was a game like that that we modeled, one where person is fighting an npc or fighting in co-op, so basically there is a way to have many different actors, players killing players or players killing npcs. If the npc defeats the player the player can not win. So it's kind of doable, but I also see your point that there is no priority.
A: The second thing that I probably should mention is that this is like a tool to make concept, to build like a strategy or a design. Usually the biggest problem in the design process is not coming up with the overview of the design but with the details especially when the programmer comes into place because this program does not have much value, it just shows roughly what is possible but every programmer needs super crazy amount of details and in order to implement something and we have for the management of the producing side, you need these details in order for the program not to say like how much effort and cost it is.
Q: So for example if you have right click and there is gonna be a new window saying that this would roughly cost X amount of dollars to generate X amount of players and the description for the developer.
A: It might be helpful but of course having a feature like have fun and immersed is more kind of concepts but you need processes really. You need this happens then this happens then this happens. And it's very very important to model the player interaction with the world looks like so this is really called features but of course there also features when the world interacts with itself and does stuff automatically, this is not just for immersion but also for game play so say trees can grow themselves or there might be animals moving around killing each other if you go hunting or if you kind of use trees as wood resource the player of course kind of like farms the trees, or kills the animals so this is like a feature but of course the world has own functionality. But then again for program I just have the information like, kill the animals is not enough so you need a lot of details and let's say you have a gun so you need to be able to aim, be able to walk, you need to be able to shoot somehow and depending if it's a multi-player game, it needs to be synchronized somehow with other players and then when you shoot something, you need to range, it's like the shot needs some actual moves like velocity or if it's instant, no velocity. Of Course you need to damage and target needs support system and you need a collision system so you can collide and it gets quite complicated very soon. So the functionalities is important for any.

Q: The model for need for speed only needed to generate enemies and the whole process, the whole model became extremely complex. The people, the general expression, that model is like many. So we're trying to do is find the balance of good enough information but not only complicated stuff.
A: The problem is you need to show all the functionalities in order for it to be useful but of course you don't have to go into details. So if you have the ability to double click something and open another graph this will be good. (Like a statechart)
Q: Anything else that you think that they held process of more than this is useful for gaming?
A; Yeah, I think it is specially for like game design-related stuff so a programmer might probably not use it so much but you might possibly get something to model from a game designer and then try to understand it come back to the designer and say that I need ten times more detail but of course it is good for an overview. You will have a design document anyway like several hundred pages or even more depending on the complexity that outlines all the features in detail and has like a lot of graphs and text. But just to have a quick overview of how the system works this is kind of helpful specially if you add new developers to projects or add additional designers. Maybe even if you have people doing scripts, and they should probably know stuff like immersion when they work on something related to completing a match they should have some information on what the goal is or you should have fun and be immersed and player related things. This can be quite helpful but I think when the tool is flexible enough it's up to the team to decide how to nail this stuff. Of course specially at Unity know the more flexible the tool is, the more chance you have to screw yourself up and the more fixed the workflow is the less different things you can obviously do but also low other risk that your design can be really bad.
Q: Thank you so much.

Interview with company 10.

Q: How long have you been working in the gaming industry?
A: I've been working I guess about a year but previously I studied for 4 years.
Q: What did you studied?
A: I studied generic game development I focused most on programming and design. I had a two year art school degree.
Q: How long have you been working with your current company? What role do you have?
A: I am the CEO, the programmer and the artist, we've been working together for 2 years .
Q: The game that we focused with this discussion is the when you're gone.
A: We spent January, December we're working on other games for phone but we decided to scrap it, so we're working full time.
Q: When you come up for that game, do you follow the same process every time? What is needed in the game for it to be this game?
A: A framework, minimum viable product. How I arrived at that? Well in our case, our company is very story narrative driven. We often start with something that we want to tell and then we sculpt the game to allow us to tell that story and also be manageable with our team size. The framework and the scope is entirely locked to whether or not it allows us to tell the story and if we find that the story can be told within our means of production then the game can be made.
Q: So how do you figure out if it's plausible or not?
A: When I'm studying, there are not a lot of time to complete a game because they are course dependent but you do have a lot of time to work on small and unfinished prototypes like I rapidly got a grip on general. If I do all the programming myself, I could dictate whether or not I feel like I will be able to do something the things I did in this courses were both turn-based and placed on a board. So I could extract a lot of that into this game, the next game we'll be making is a 3D game, 3D third person platform and there are a lot more question marks getting because it's both going from 2D to 3D and scaling up the team to

one additional member. There's a lot of factors. In that case, we decided to keep the story or the narrative quite open but focused on a core theme. In this case, it's language and translation and if we can manage to fit our core mechanic into. I don't think I can fully answer your question.

Q: So you don't use any tools for processing in the beginning when you try to come up with how the game is going to be.

A: No, I don't use any tools. I do a lot of thinking before hand, the team sit down and we talk rigorously then we figure out what's important and what is vital to our enjoyment then we just try this out. When it comes to tools, there never any better than the person using them and I don't know how much I'll get off like tools initially comparing to just starting working on the very basics and trying to evaluate that versus so many unknowns.

Q: What strengths and weaknesses do you have. Is it dependent on the person using them. Would you say that? In our study we're trying to structure the process of development games.

A:I have been working with tools like that before and I want to do as little administrative things as possible.

Q: Do you know what goal modeling is?

A: No, I am not familiar.

Q: So it's basically a way to structure your early requirements. We are going to show you a model here that we made for how you think your game is. So here is the company and the actors these actors represent the game companies. <Shows model> <inaudible minute because of malfunction> We described the game aesthetic as intriguing, And the character here Sam, so we thought that the goal of the character Sam is to tell a story and to do that he has to battle enemies, and I'm not sure if he has attributes but in the model he does, and voice acting probably to tell the story. But of course you would make it a bit differently. But this kind of modeling is not being done in the gaming companies.

A:It reminds me of a mind map.

Q:Do you think the tool might be useful for game development?

A:Yeah sure sure, I believe that you should figure out a way that works for you. So that you know what to do, but it seems this tool is designed to scope. Based on what you told me. It's seems to be there to explain the game as a node based system, I believe that our game is quite simplistic it's like chess but a little push and pull, but a game like heroes of might and magic, has an immense number of interactions, mess with something over here and something changes over there, and it's not always clear how those systems work together and something like this could help to de-abstract it or show those mechanical layers. Yes something like that, sure I can see the purpose but of course I can see where it is just busy work depending on the person. No offense but I think that is true for every design tool or method, the larger the team, the more relevant these things become, in our team it's just me and my friends and we can't afford to not know what the others are thinking so we make sure everyday that we know.

Q:So what about those early sessions with your friends then, is something like this not applicable there?

A:It could be, but I'm not sure that if I use your tool it's going to be agile enough for the dialogue of the words between the colleagues and me. And after the end of a long talk we usually end up somewhere radically different and I am not sure that this tool can adapt as quickly as we can. So it might not be a tool that I would use for something like brainstorming more like something to use after a brainstorming session to structure it.

Q:We have this creativity technique for brainstorming. Where you can write down ideas and then you have a lot of them and you can accept them or reject them.

A: Maybe me using the word brainstorming was unfortunate because we don't sit and get ideas from the aether. I've never seen a good product come up from brainstorming, I'm not cool on the fly, it takes a lot of shower time or sleepy time to come up with cool ideas.

Q:Would you change something to make this more useful for game industry?

A:Yeah sure I'm no expert at these kinds of tools but if I were to use it I would like a collaborative function where me and my colleagues can use it simultaneously. Drag and move and add. Color code each of us. It would make it a more fun tool if nothing else. If I am the only one moving the bits around it's not really an interactive process for the rest of the team. I would not be able to use this tool to kick off a project.

Q: Any other feedback?

A:Tools can work but why isn't there more flair given to the actual graphics. If it snaps and makes it own molecular structure it might seem superfluous but now it reminds me of powerpoint presentations or mind maps which sucks the energy right out of me. You only have one chance to make a first impression, if it's pretty it's more likely to hook me.

Appendix G.
Interview with company 1.

Q: So we made concept for some changes, so we are going to show you. This is the tool now, not really it's just a concept. We have a collaborative function, and we are going to add that many people can work on it on the same time. Then, some layers, so you know who are interested in the things maybe and you can also add images to the layer, maybe that will help somehow.

A: So, if you click manager, you will see this?

Q: Yeah, if you click programmer, you will see this but more in depth, same for designer, but then it will be more graphically related. We are going to add buttons, so that you can minimize and you can fit more on the screen, so to reduce the space it takes up. If you double-click on the things you will be able to add description to them, so they won't be here all time and you can hide them up. There is a priority button, to add priorities so you can say how important those features are. From 1 to 5,basically like that, something similar to pivotal tracker, something like that. We also added a way to add text, also support for something like trello or pivotal tracker, where you could export your nodes into tasks, with generated function. So what do you think about these changes?

A: I think it looks way more to understand now, so I like that, it's simplified, compared to the version I saw before.

Q: Maybe it's because it's a smaller model maybe

A: But I think it's more cleaned up, which I like, but perhaps you can walk me through it in little bit more details? Now you talked about it in more general terms, if you talk more specifically, like for example what is this model?

Q: It's a card reader, so the goal for the cardreader is to perform the transaction, so one little goal it has to achieve to get there is to authenticate that it is actually correct and how does it do it, well you actually write a pin code, that is depending on a credit card, and you can perform a transaction.

A: So, the project would be to build this card reader?

Q: And it's also simplified, there would be more tasks. And if you want to see more tasks, you show them in another layer.

A: That's one of my comments, that I really like layer function and the graphic view, because now that I work like with game design and game development there is a lot of text, and you have them in a lot of places, and pdf and next you don't have them in one place, but here you have them in layers, you can flip from one to another, and that's way more easier to understand, so that's a great feature, I think. Also, I liked to add images, and just easily to throw in design, like now I want to do this and I tell or someone tells, why you don't tweak, so you have everything in one place, really interesting. So if you tell me what you have taken away from the first version?

Q: We used to have a creativity tool, it's still here, but we made it so if you press a button you will get that. General idea from other developers was that it's not that hard to come-up with ideas, it's more hard to verify the ideas, to connect them. The actual coming-up with ideas isn't that hard and that's what creativity is for.
A: Oh yeah, before you had arrows
Q: Yeah, they are still there and I guess we didn't change the functionality of arrows that much, they are still there
A: So this is like a project overview?
Q: Yeah, I click here (pointing at the minimize button on screen) and it opens up and you can see this how you want and if you click this …
A: What about other goals?
Q: Not it is only for actors right now. If you double-click you will get the pop-up with information and you can set-up priority if you want to.
A: So in your model these would be clickable, so if you click this, you would get it.
Q: Yeah. So if you had a tool, would you use it for your next project for example
A: Sure, I mean would put all my data in it and see how it would change my view on a project and see, how the other team members, how we can more easily understand each other's viewpoints, this is the main benefit I think would be for a team, I am not a programmer, when I click this the programming box and you programmers put in data in a way that a manager like me can understand, the same goes for programmers I guess, understanding my view, I think it's a good forum or arena to facilitate communication.

Interview with company 2.

Q: So we made some concept change, based on the interviews we got we're just going to show you a picture of them basically, so what do you think? So basically, we made different layers so you could have the things for more people. It is just a concept and then collaborative work so there could be a lot of people working on it at the same time also there would be descriptions, you double click on the things so you could write more things and lastly we have added trello support. It should be able to move everything into trello and sprints automatically. If you want to minimize it after it would do it by clicking also you could set priorities, these are like the soft goals and how they would be like tasks we are basically trying to see if the changes are viable. Which changes do you prefer to have? Which one do you think is redundant and would it make this tool better and usable for you?
A: As I remember it the tool was used to help with ideas, now i get the feeling that is more like task manager.
Q: It's more like you start with the early ideas and then you move on so you can go to the next stage. You can also have creativity tools if you want to then you can just press the button and got that response that is not really a lot people keep on saying that, they don't need ideas. One of the few plans, we have to think about it was to verify the ideas. One of this crazy idea is to just search this on twitter, just create some forum host and get that verification if you actually need that.
A: I think it's more useful when you can add tasks and have it like a manager but you can want it so everything could be in here from start to the end.
Q: That's not really the point. The point is, you start from this, when you have this whole game here, your whole idea and everything then you can transfer it to trello then you can work from there.
A: So you don't use this as a tracker?
Q: If you still want a tracker, you could still connect in here if you want. I'm not sure if that is the point. The point is to go from here to the next step easier.
A: I just try to think how and what way would this be really useful compared to just sketching. I don't mean to be rude but I don't really now. I can't really see it useful.
Q: So you still wouldn't use it?
A: No, I don't think so. Before you go over to the tracker, the stage before that. I think maybe it's just me but it's easier to just sketch on a paper like mind maps are. What's the benefit compared to making mind maps.
Q: The advantage of this is you don't actually have to draw mind maps to show them and you can save it and go back to it at a later date. With a mind map like on a whiteboard you can't always because this is digital. And also people can work like 5 to 6 people on things, you can do trello task incase you want, and you can put a lot of ideas here and just reject them or accept them.
A: So you can use this while you are on skype?
Its like google docs benefits.
Q: Yes
A: It's like google docs benefits that makes it useful. I don't really know how useful. But I just don't know how to make it better. Otherwise I like that you added so that you can make tasks that you have a task manager and you move it to a tracker.
Q: these things will be like a checklist for this task because you will need to write a pin code this functionality is to be able to do this right.
A:Yeah
Q:So nothing else to add?
A:I don't think so.
Q:Alright thank you.

Interview with company 3.

Q: Do you remember our tool from last time?
A: Yeap, pretty much
Q: So, we basically made some concept for some changes and we would like to show it to you. What we did was that we added collaborative function, that there can be many people on the same model. We also added layers, it's like viewpoints, so if you click them, things that are made there would appear for that layer and if you click again things will disappear. Also we made it so that you can minimize them, like actors, where things will be, so things get smaller, if it is too much of a cluster, it would free up some space. If you double-click on one of the things you will get a description box for it, where you can write further description.
And priorities to see how important this thing would be. We also added trello or pivotal tracker support that would make things go from this stage to auto-generate trello backlog. All of these would be the tasks and these would be tasks. It would make so it would be easier to transition from early stage to later stage. So what do you think?
A: I am trying to remember how it looked last time I saw it, but, absolutely it's an improvement. I am trying to get a grip of what it was…
Q: So the program was at the early stages to map up your thoughts and ideas and things like that. Syntax and everything is pretty much the same.
A: I am trying to see how I want to use it. Because last time we spoke, I expressed the need for integration with Pivotal Tracker or Trello, so that's good. I am just looking at our process and how it would be used. I figured that the point of the system is pretty much to prioritize tasks and to get overview of dependencies?

Q: First you make your ideas and you map them up how they would actually be in the actual program and you make dependencies and you make connections between things and the thing that we are going to add is that we make things into actual tasks that you can transition from your earlier stages to the later stage.

A: What do you mean by later stage? Next sprint?

Q: No, you do this before sprint or very early, when you come up with the concept for your game and stuff like that.

A: So, if we are to do this, we need to do this and this would be a feature of that (*pointing at a screen").

Q: I guess you can go back to if you want to in the later stages of development phase as well, but it's not the point of it, point of it is to start somewhere and have structure from the start.

A: But is it meant like to get overview of the tasks rather than detail planning them?

Q: Yeah, but you can add details, I mean this would be detailed planning, the tasks would lead into the goals. Resources and tasks lead into goals in how the program would work.

A: Ok, so it's rather a description of all the components in the project than the project planning itself. To get the understanding of the components and the parts needed to make everything work then. I see a use for it, because it's not uncommon that you plan something and try to figure out what I need to go to make this work, what components are needed here and try to scope it and it's really difficult to cover the things and not forget components in the whole thing. If let's say you plan a network application and you like, what do I need for this application, we need a database, we need to set up a server and a client and then you might forget about a lot of these things, for example you forgot your planning communication between database and a server, some components that are necessary and because of that it would be hard to project plan or rather make the scope real good. So for detecting those hidden workloads it would be useful if you could see and plan all the dependencies. How does the integration with pivotal tracker work?

Q: Since we are in the conceptual stage, it would be like your goals become bigger tasks and smaller things that you would need would become checklists or roadmap to that goal. So that would mean you would need to name them like in pivotal tracker or trello. In pivotal tracker there is priority tag, like a 1 and 2 and 3. That would be integrated into priority button. And you would use trello, priority would just create a tag and you could look of how high of a priority a task has.

A: When we are using the pivotal tracker, we do it strangely, because I think that we don't use it as intended, but we usually rank our, or place our points based on time invested, so we make a time evaluation, so how long would this task take and then, of course we need to divide them down into smaller tasks, we can't just say that we can that that off, because the task is too big, but we say that 1 point is 4 hours, so 2 points is our work day and that's how we usually handle things. Because it's great that I got that detection of all the dependencies and all components needed, but as I said if I have a small tasks that I need to convert into pivotal tracker and sprint, I would need some way to make the tasks much smaller, as we talked about, let's say we need a database, it wouldn't be one point it would be higher or maximum.

Q: Yeap, so the big goal would be an epic, in pivotal tracker I guess, and you make the smaller things for the actual database, as a smaller tasks here, like a small things you need for your database, like a sort function, I don't know and that would be a thing like this. On that thing you can work for a week, and actually completing the whole database is taking out longer time.

A: But you mean building into layers then, so I get the overview, so at my top level I have 10 tasks or ten components, the back-end, the front-end,network component, user-testing and when I click into them, I get zoomed into that particular area and can look into those things, is it like that you mean?

Q: Not yet, I would say. We do have layers. You can make your bigger tasks here and you can program the things you adding, much more detailed things.

A: What is the thesis of the project that you are doing?

Q: We are going to check how this kind of framework and tool would be used in the gaming industry or if it is not used, why and how we can improve the tool so it can be used.

A; But what is the end goal, what do you want to improve, what aspect of the development process?

Q: Early requirements engineering. We would like to see if this concept is used in gaming industry and if not how we can promote it, how can we change the framework.

A: So, it is more from the project lead standpoint, it's a tool for understanding the scope of the project? I can absolutely see it used for not missing things and understanding what context and what components project need of.

Q: So, if the tool would be changed like that, would you use it, hypothetically on your next project?

A: I think there is a big barrier, when it comes to programs, because you claim that, and you might be correct saying if you use this tool, you will improve on whatever, like your efficiency, your accuracy on prediction or whatever you are talking about, but when it comes to new tools, I need to have a pretty big improvement to transition and actually learn the tool, because I actually have no idea, how this works, I don't know what happens when I click here or here, ok, you explained things over there but I don't know the exact reaction to my input. To get into new system, I try to, we talk with my colleague like couple of weeks ago, and we got from the job and started to use, we swapped from whatever we were using before into SourceTree for git, and he was very reluctant on beginning of using it, it took us like two weeks to convince him that it is actually easier to use, so it exists something to get over to start using it, so what is the big improvement of using your software, compared to using whiteboard in my case?

Q:Because it's digital, so you can go back to it easily and a lot people can work on it at the same time, you can work not co-located. Also, since it can translate into trello and send stuff automatically, that's a transition between stages.

A: Then we have two things, one of the things, there are whiteboard online, so I could actually use a whiteboard online, but you are right that in transitional stages you need this one(tool), and if you think it can be made, it would be valuable, making project planning easier through skipping a step of something, we always need to discuss what will be in this sprint for example, or what we need in this project and then we would dim that down into tasks in trello or other tool. As I said, if tool would have that, it would be something good, it would get me over that bump of transitioning of new way of working with it, but I think, it is the most important question. The program needs to add a significant benefit and the question is what are those benefits. Focus on those benefits, rather than on things that are not benefitial compared to the other things.

Q: So, you would say that ideas, that we on the right side before are not benefit of this tool?

A:Like what?

Q:It used to be here some brainstorming tools, roleplaying tools to come up with ideas

A: If it saves me considerable amount of time, than … what was the brainstorming tool?

Q: We can go to the real tool then … it's like you press brainstorming, you write your ideas, what you want for your game I guess. When you have a lot of ideas, you can prioritize the ideas, you can prioritize the ideas or reject that and you can marked it as model, you can keep a lot of ideas on the side and then when it's actually modelled, press model, and you won't need it more, because you already have representation of that.

A: Yeah, I see benefit of that too, when you use whiteboard or things like that, a lot of people use post-it(wtf?) and you can do them in circles, move them around and have them in different colours. What is not possible with post-it is that … I am brainstorming, so I am talking loud ... I can't change the color of the post-it note on the go, but on the other hand I can just take a post-it and write the same thing, so I see benefit with your solution, but I don't see overwhelming benefit, so what I said before about learning and stuff.

Q: So there is a barrier, because you need to learn it and convince people to use it as well?

A: Yeah, so for example Word, let's say that I like writing with paper, and someone comes with the Word, then there are significant benefits, like if I write on paper, and like oh ****, I need to have a text here, on a paper I am f****d I need to start over, but in computer you can just copy it down and get it working. It's a significant benefit. Writing on paper is bad compared to writing on computer. So you need to have a same thing in your program, whiteboard need to be s**t compared to your system. If you'll manage that - people will use it.

Q: Ok, thank you

Interview with company 4.

Q: You remember our tool?
A: Yes
Q: We made some concept for changes based on the interviews we had. This is just concept art. This is collaboration also we added levels of detail, through layers. Like what people are supposed to watch it for bigger teams maybe. If you press one of the layers things for that layer will appear. And also we have added so you can minimize the actor boxes. And text boxes. We still have the creativity tools like we did last time, but we put them here and we are going to integrate some kind of trello or pivotal tracker so you can link it to what is in here.
A: How is that doable?
Q: The soft goals and goals would be tasks and the tasks and resources are the roadmap things how to get that task.
A: So it's dependant on the type of analysis you do when you start with it. Like if you don't do this whole thing it won't work properly because it will just be called "Authenticate" (Like the softgoal in the model)
Q: Yeah and you will also have to do some other things to them, like adding due dates. And if you double click on the soft goals, you will get description boxes and you can set their priority.
A: So like a star or not.
Q: For now it's something like a prio 1-5. And we would like to have support for every tool like trello and pivotal tracker. So what do you think about these changes?
A: Let's see, they are an improvement in the sense that there is a clear roadmap on how the tool actually promotes better development practises later by putting things in this trello thing. And as far as the user interface and being able to shrink these and add layers. But it still doesn't, I'm going to try and remove myself from this, because we've been through my opinion on the necessity of these kind of tools. But if you are a team that would like to use tools like this, the things you added make the tool better, comments and the ability to make things less cluttered. But for me personally I still don't understand how this is not just extra work, because I've never seen a project actually go from the idea stages to the product without changing itself 8 times over, and sometimes day-to-day stuff. That's my concern, how will this tool keep up with a conversation or the changing nature of a project, or am I mistaken, is it not supposed to be iterated upon.
Q: Yeah maybe you can just abandon this after the early stages, basically to start something off, and map it up and then move forward.
A: The only way to accurately assess whether it works, is like to demo it in a game jam or a real context. Actually try the features out, I have opinions that are solely based on my intuition, there is a lot of ways to use your tool inefficiently, there are many ways you can fuck up this tool, creating too many layers or not understand what will be scrapped. This tool for me can be used to cement ideas onces they are out of the talking stages, let's see if we can create a roadmap or systematic approach to what we are making. If you have talked about it and someone has kept loose notes, like when we are talking and maybe had a beer and thinking about what we are going to do, when we are done with that meeting, the last thing I want to do is to slow down, and do this. But that depends on how organized you are as a person, I can really see the more meticulous person get use of this. Because they don't have to keep everything in their head, and it could be good if you are a big team that later adds a member or two. It's a good jumping off point, showing a graphical representation after a verbal assessment of the project, It's free from people forgetting stuff and such. The ones that will benefit from this, is larger teams or smaller teams if they have a more systematic approach, which is rare, but of course it exists. I'm still in the impression that the presentation could use some work, it looks like an office app. There is no excitement factor too it, I don't know how to add it without sacrificing clarity.
Q: So you still wouldn't use it.
A: No because it's fundamentally the same thing but better. "You want to have pizza?" "No I'm not in the mood for pizza", "But we made the pizza better" "I still don't want to eat pizza".
It's not that it's not good or better, but it's still what it is. It's an organizational tool aimed towards a person who doesn't gain a lot from organizing things because it never sticks. I've done textform asset lists and things we have to do, and keeping up with that list even if it's just text. It's just work. But we are a small team so we can be very agile. So a larger team maybe they can't be as agile and a tool like this maybe drastically improves their performance, I can't say because I'm not in a big team and I don't have the experience.
Q: Thank you.