

## Slime Attack!

A forced collaboration game across four iPads designed for children in primary school

Bachelor's thesis in computer science and engineering

ADAM ANDREASSON  
VIKTOR DJUKIC DIMANDER  
DANIEL ILLIPE  
PATRIKS OLSON  
MARCUS RANDEVIK  
VIKTOR SVENSSON



BACHELOR'S THESIS 2018:05:26

## Slime Attack!

A forced collaboration game across four iPads designed for children  
in primary school

Adam Andreasson  
Viktor Djukic Dimander  
Daniel Illipe  
Patrik Olson  
Marcus Randevik  
Viktor Svensson



**CHALMERS**  
UNIVERSITY OF TECHNOLOGY

Department of Computer Science and Engineering  
CHALMERS UNIVERSITY OF TECHNOLOGY  
Gothenburg, Sweden 2018

## **Slime Attack!**

A forced collaboration game across four iPads designed for children in primary school

ADAM ANDREASSON  
VIKTOR DJUKIC DIMANDER  
DANIEL ILLIPE  
PATRIK OLSON  
MARCUS RANDEVIK  
VIKTOR SVENSSON

© ADAM ANDREASSON, 2018.  
© VIKTOR DJUKIC DIMANDER, 2018.  
© DANIEL ILLIPE, 2018.  
© PATRIK OLSON, 2018.  
© MARCUS RANDEVIK, 2018.  
© VIKTOR SVENSSON, 2018.

Supervisor: Olof Torgersson, Chalmers University of Technology  
Examiner: Arne Linde, Niklas Broberg, Chalmers University of Technology

The Authors grants to Chalmers University of Technology and University of Gothenburg the non-exclusive right to publish the Work electronically and in a non-commercial purpose make it accessible on the Internet. The Author warrants that he/she is the author to the Work, and warrants that the Work does not contain text, pictures or other material that violates copyright law. The Author shall, when transferring the rights of the Work to a third party (for example a publisher or a company), acknowledge the third party about this agreement. If the Author has signed a copyright agreement with a third party regarding the Work, the Author warrants hereby that he/she has obtained any necessary permission from this third party to let Chalmers University of Technology and University of Gothenburg store the Work electronically and make it accessible on the Internet.

Bachelor's Thesis 2018:05:26  
Department of Computer Science and Engineering  
Chalmers University of Technology  
SE-412 96 Gothenburg  
Telephone +46 31 772 1000

Cover: Visualization of a Slime Attack! game in progress.

Typeset in L<sup>A</sup>T<sub>E</sub>X  
Gothenburg, Sweden 2018

## **Slime Attack!**

A forced collaboration game across four iPads designed for children in primary school

ADAM ANDREASSON

VIKTOR DJUKIC DIMANDER

DANIEL ILLIPE

PATRIK OLSON

MARCUS RANDEVIK

VIKTOR SVENSSON

Department of Computer Science and Engineering

Chalmers University of Technology

## **Abstract**

The increased usage of tablets in recent years has caused significant changes throughout society. This change can also be seen in schools where tablets are now incorporated throughout primary education. Though, tablets are often used individually causing a lack of collaboration, which is an essential part of education.

This paper describes the development process of the mobile game Slime Attack! and how it encourages young children aged 7-9 to cooperate while playing. Slime Attack! is a castle defence game, played on four iPads placed adjacent to each other, collectively forming a large playing field.

Initially, a prestudy was carried through, mainly focusing on interaction design theory. Another important basis came from interviews concluded with the intended user group. The gained knowledge from the prestudy provided a foundation for the development to further build upon. This foundation, summarized in a number of strict requirements, together with the narrow scope of the project were thereafter used to form the final game concept.

Slime Attack! was shown to encourage collaboration amongst children by fueling a discussion of strategies concerning the expansion of their outpost, and also in supporting each other when the amount of enemies increased. We also discuss flaws in the prototype, including the children's initial competitive playing style, and issues with the user interface that slowed down the onset of collaboration.

Finally, we put forth a number of possible changes to the game that would address the major flaws we found in our tests.

Keywords: iPad, forced collaboration, co-located, mobile game, multi-player game.

---

## **Slime Attack!**

A forced collaboration game across four iPads designed for children in primary school

ADAM ANDREASSON

VIKTOR DJUKIC DIMANDER

DANIEL ILLIPE

PATRIK OLSON

MARCUS RANDEVIK

VIKTOR SVENSSON

Department of Computer Science and Engineering

Chalmers University of Technology

## **Sammandrag**

De senaste årens ökade användning av surfplattor och mobiltelefoner har orsakat stora förändringar i samhället. Denna förändring syns även i skolan där enheterna numera används i stor utsträckning i grundskoleutbildningen. Dock används ofta surfplattorna individuellt och uppmuntrar inte samarbete vilket är en viktig del av utbildningen.

Den här rapporten beskriver utvecklingsprocessen av spelet Slime Attack! och hur det uppmuntrar barn i åldrarna 7-9 att samarbeta. Slime Attack! är ett castle defence som spelas på flera intelligande iPads vilka tillsammans formar en stor spelplan.

Inledningsvis genomfördes en förstudie som huvudsakligen fokuserade på teori inom interaktionsdesign. En annan viktig del var intervjuer som utfördes på den avsedda användargruppen. Den sammanlagda kunskapen från förstudien lade grunden för fortsatt utveckling av projektet. Den här grunden, bestående av olika krav tillsammans med den smala projektomfattningen, användes därefter för att forma det slutgiltiga spelkonceptet.

Slime Attack! visade sig uppmuntra samarbete mellan barn både genom att de tillsammans började diskutera hur expansionen av basen skulle gå till, men också genom att hjälpa varandra när antalet fiender ökade. Barnens instinktiva reaktion till spelet var dock till en börja ofta tävlingsinriktad och svårigheter med att förmedla vikten av byggnadsfunktionen uppmärksammades.

Slutligen föreslår vi ett antal möjliga förändringar i spelet som hade eliminerat de stora problemen vi iakttog vid test-tillfället.

Nyckelord: iPad, forced collaboration, co-located, mobile game, multi-player game.



# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
1.1	Purpose of project . . . . .	1
1.2	Task . . . . .	2
1.3	Social and ethical aspects . . . . .	2
1.4	Scope . . . . .	3
<b>2</b>	<b>Tools and methods for game development</b>	<b>5</b>
2.1	Semi-structured interviews . . . . .	5
2.2	User tests . . . . .	6
2.3	Agile software development . . . . .	6
2.4	Development platform . . . . .	7
2.5	Choice of communication technology . . . . .	7
2.6	Secondary development tools . . . . .	8
<b>3</b>	<b>Prestudy</b>	<b>9</b>
3.1	Designing for cooperation . . . . .	9
3.2	Designing for children . . . . .	10
3.3	Investigation of existing games . . . . .	10
3.3.1	Games with an academic background . . . . .	11
3.3.2	Games on the market . . . . .	11
3.4	Interviews at Johannebergskolan . . . . .	12
<b>4</b>	<b>Development of Slime Attack!</b>	<b>14</b>
4.1	Creating a game concept . . . . .	14
4.1.1	Requirements for game concept . . . . .	14
4.1.2	Brainstorming . . . . .	15
4.1.3	Concept evaluation . . . . .	15
4.2	Concept development . . . . .	16
4.2.1	Determining the target to defend . . . . .	17
4.2.2	Building for active progression . . . . .	18
4.2.3	Encouraging interaction across devices . . . . .	19
4.3	User interface, graphical assets and sound . . . . .	21
4.3.1	Enemies . . . . .	21
<b>5</b>	<b>Results</b>	<b>23</b>
5.1	Game concept . . . . .	23



---

5.2	Breakdown of a game session . . . . .	23
5.2.1	Setting up a connection . . . . .	24
5.2.2	Organizing the playing field . . . . .	25
5.2.3	The game . . . . .	25
5.2.3.1	Wave . . . . .	26
5.2.3.2	Buy time . . . . .	27
5.2.4	Starting over . . . . .	28
5.3	Implementation details . . . . .	29
5.3.1	Network communication . . . . .	29
5.3.1.1	Connecting the devices . . . . .	29
5.3.1.2	Synchronizing game state . . . . .	29
5.3.2	Making enemy waves interesting . . . . .	30
5.3.3	Balancing the economy . . . . .	31
5.3.4	Determining enemy targets . . . . .	31
5.3.5	Function of the solar panel . . . . .	31
5.4	Results from user tests . . . . .	32
<b>6</b>	<b>Discussion</b>	<b>34</b>
6.1	Lessons from user testing . . . . .	34
6.2	Suggested improvements . . . . .	35
6.2.1	User interface and user experience . . . . .	35
6.2.2	Improved structures . . . . .	37
6.2.3	Bugs and performance issues . . . . .	37
6.2.4	Game tutorial . . . . .	38
6.3	Evaluation of the process . . . . .	38
6.4	Choice of device communication . . . . .	39
6.5	Development platform . . . . .	39
6.6	Outcome of social and ethical aspects . . . . .	40
6.7	Future research . . . . .	40
<b>7</b>	<b>Conclusion</b>	<b>42</b>
	<b>Bibliography</b>	<b>43</b>
<b>A</b>	<b>Letter of consent</b>	<b>I</b>
<b>B</b>	<b>Prestudy Interviews</b>	<b>II</b>
<b>C</b>	<b>User testing transcript</b>	<b>VI</b>

# 1

## Introduction

The rise of cheap and readily available technology has caused rapid change throughout society in the past decade. Small computers in the forms of tablets and smartphones have seen an exponential growth in sales, where in the most recent years almost a billion smartphones have been sold [1]. The effects of an increased availability of smartphones and their usage in everyday life can be attributed to the phenomena of a “mobile bubble”, where people are enclosed in the interaction with their own device [2].

An increased usage of computers can also be seen in education, especially in the Swedish primary and high school where a large portion of the students have access to either a tablet or a laptop [3]. Although tablets serve as a viable tool in education, many of the applications developed for usage with tablets are designed to be used by a single person [4]. Because education includes not only logical skills such as math, but also soft skills such as collaboration [5], an application that could encourage and help develop collaborative skills in children could have some potential benefit.

Hand held and mobile devices, such as tablets, have been shown to be a competent tool for encouraging collaboration [6]. Their compact size and multi-touch screen interface allows for users to easily share devices without the struggle of only having a single input. Previous work has been conducted exploring the development of collaborative games for cognitively impaired children, where games are played in a co-located setting across several tablets [7, 8, 9]. This is an interesting type of game concept as it allows for the usage of mobile devices as a mean of encouraging collaboration without creating a mobile bubble. We intend to extend this concept by developing for and with children of average cognitive abilities. More specifically children in the ages of 7-9 are of interest as this is the age where most develop the ability to collaborate and think in the perspective of others [10].

### 1.1 Purpose of project

The purpose of the project is to create a game for young children in the age range 7-9 that forces collaboration between players in addition to provide a way of using tablets devices without creating a mobile bubble. The game is to be played across several tablet devices placed next to each other forming a large playing field. The purpose of this paper is to describe the development process and motivate the various

design decisions that were made.

## 1.2 Task

The task of the project was to design, create and implement a game that fits the criteria put forth in the purpose. More specifically, this game is played across several devices placed beside each other collectively forming a large playing field. A handful of the design challenges associated with the project includes not only designing for children and forcing collaboration, but also technical aspects such as a multi-screen interface and seamless device communication.

Applications designed to be used by children need to accommodate for the lack of knowledge and experience compared to adults. Children of young age have, for example, not yet fully developed the ability to read and thus cannot comprehend instructions in the form of long texts.

Because collaboration is an essential part of the game, the design of the game should make it difficult or preferably impossible to be completed without the inclusion of several users. This collaboration should ideally be verbal, as the players communicate over common decisions, in addition to the active interaction throughout the game session. The game design should reward the team as a whole and not punish the individual players for their mistakes.

As the game is played across several devices and by users placed at varying positions, it is important to ensure the interface is consistent from all angles. Elements of interaction need to be placed at location which can be reached by the intended users with ease despite the large screen area.

The usage of several devices also requires there to be a network connection in order to sync updates within the game. This connection should be easy to set up for the users, reliable once set up and provide acceptable performance in terms of delay and bandwidth. Choosing a network technology should be based on the available technology in the schools whilst still keeping performance as a priority.

## 1.3 Social and ethical aspects

Although the goal of the project was to develop a game which encourages collaboration, thus bringing positive effects, the use of tablets and games may have some negative social and ethical effects. These are aspects that need to be considered throughout the development in order to minimize any potential negative impact.

Perhaps most apparent is the limited number of players who may attend at any given moment due to the fact that a device may typically only be used by a single person. Limiting the number of players in the game introduces the risk of some children being left out of the group. For any children that may already feel excluded, the inability to play the game with others may strengthen that feeling. As a result, it is

preferable that either the design of the game allows for a variable number of people, even when the number of devices is fixed, or that the session is supervised by an adult who may dictate groups.

Another important aspect is the potential of game addiction. As of 2018, the World Health Organization (WHO) now includes gaming disorder in the 11th Revision of the International Classification of Diseases [11]. Gaming disorder is defined a recurrent gaming behavior manifested by: impaired control over gaming (e.g duration, frequency, termination); increased priority of gaming compared to other personal activities; and the continuation or escalation of gaming in spite of negative consequences [12]. Developing a game which is to be used as a part of education for children, an environment not typically populated by many digital games, does increase the exposure of games. In order to avoid the risk of giving rise to an game addiction, leading to a gaming disorder, game sessions should be designed to be short, limited in scope and not provide a way of grinding to earn in game advantages [13].

Finally, video game violence and its link to violent behavior is an issue that is being actively researched upon. The American Psychological Association (APA) adopted a resolution in 2015 that concluded the exposure of video game violence to be a significant risk factor of aggressive behaviour [14]. Though, APA notes that there is a lack of research for children under the age of 10. Research has also shown that the link may not be of significant size and could even be the result selection bias [15]. Conclusively, the game should not contain more violence than necessary in order to reduce the potential risk of increasing aggressive behavior.

## 1.4 Scope

The intended target audience of the game was young children between seven and nine years old in the Swedish primary education. As such, the game was developed in Swedish with no other languages necessary. The Apple iPad has the largest market share compared to competing tablets and operating systems in Sweden[16]. Because of this, the game was only required to function properly on the iPad.

As a core part of the game is to force collaboration, no effort was made to adapt the game to fewer than two devices. Additional restrictions have been made to have the game only properly function on four devices as these form a quadratic playing field.

The initial set up of the game session, i.e. the placement of the tablets in the correct positions, is an interface and interaction design issue that has been studied and solved in various ways [4, 8, 7, 9]. Our intent was not to further develop this concept but instead use already existing solutions.

The game was implemented using already existing tools for game development. As such, no effort was made to create tools and/or methods making future development of similar games easier. In addition, the report does not aim to present in depth technical solutions but instead focus on general implementation details.

As the project was limited in time, testing of the final product was conducted only

on a single class. Because of this, the findings can not be considered to be general but rather a glimpse of what might be.

# 2

## Tools and methods for game development

In the previous chapter, tablets and collaborative games were introduced as a potential tool for increasing the collaboration between children in education. In this chapter, the methods used in order to guide the design process of Slime Attack! as well the technical tools for implementation are described. Also included is a comparison of available network technologies for cross device communication.

### 2.1 Semi-structured interviews

Semi-structured interviews were performed with a mixed school class at Johannebergskolan consisting of children aged 7-9. A semi-structured interview is based on a script of questions written prior to the occasion, but new questions are created as interesting threads are picked up by the interviewer [17]. This allows for a more exploratory course of action and flexibility depending on the interviewees answers, in comparison to a structured interview where instead more specific questions may be answered [17]. Kvale compares this to “traveling” versus “mining”, where the interviewer either travels with the interviewee through their answers or mines for specific information [18]. This was chosen over a fully structured interview as our goal was not to gather specific data but to explore how children view and experience games in general. A letter of consent (appendix A) was sent out to the parents of the class through their teacher prior to the interviews. The children were interviewed in a room next to their classroom, and brought in as pairs to feel more confident in comparison to being interviewed alone. Due to the sensitivity of working with children, the interviews were not recorded, and as only one member of our work group was available at the time no full transcriptions were possible to write down. However notes (appendix B) about their answers and thoughts on each question were written down. These were then used as material to further refine and iterate our concept.

## 2.2 User tests

As the first playable version of the game took shape, we needed to test it in order to observe whether the children would play it the way we intended. The same set of children who participated in the interviews were once again included. Three groups of four were selected by the teacher in order to create a mix of students not necessarily being regular playmates. The group was brought in to a separate room and allowed to sit down around a table prepared with four devices (iPads). Each group got to finish 3 game sessions in total, which usually took about 30 minutes. No instructions were given at the start of the test. If a group did not understand how to proceed at a stage, they were given brief instructions. If a player accidentally exited the game by pressing the “home button” on their device they were allowed to start a game alone, as they cannot rejoin the ongoing session. If a group had played for some time and did not explore a large aspect of the game, hints were given to them by us. The groups were observed by two members of our group, one of which who took notes (appendix C) and one who supervised the tests.

## 2.3 Agile software development

During the project, an agile development approach was used. Agile software development describes a large set of practices for software development focused on delivering continuous value through short cycles. These practises emerged from the values and principles described in the agile manifesto [19]. The differences between agile development and the more traditional sequential development process, such as waterfall [20], is the much shorter cycles between releases. In a waterfall process, each step of the development is treated separably and will not begin until the previous one has been completed. As such, requirements and detailed specifications are established first, followed by implementation, testing, and the final delivery. In contrast, agile development focuses on completing each of the development steps continuously in smaller periods of time, known as *incremental development*. This allows for a continuous change of product specifications during the entirety of the development.

This decision was made mainly due to the fact that none of the team members had any prior experience developing a game designed for children, and even less so, a game played across several devices. Because of this, knowledge would need to be gathered during the entirety of the project making it impossible to create rigorous specifications beforehand. Several team members have had previous experience with the Scrum framework [21] and as such, it was the agile method of choice.

## 2.4 Development platform

The project is aimed towards modern tablets and should be built on a platform that supports these devices. Several application and game development platforms, engines and frameworks were evaluated with the goal of finding one that would best aid the development of the project.

When developing games, it greatly speeds up the development process to build on top of an existing game engine that handles basic concepts such as physics and graphical rendering. We researched a number of game engines with support for tablets and encountered Unity. Unity is a game engine that can compile code to different platforms, including IOS, Android, GNU/Linux and Windows. The latter two for desktops. This is beneficial since everyone in the group can deploy the game to his own platform and do not need to have the tables at hand for testing.

Unity has high level API tools for networking, graphics and physics which are all thoroughly documented online [22]. It also has an easy to use graphical interface, which makes it possible to create quick prototypes without a lot of previous experience. Out of the different frameworks that were evaluated, Unity appeared to be the easiest to learn and having the best graphical performance and was therefore chosen to be the development platform for this project.

## 2.5 Choice of communication technology

Before deciding the method of communication between the tablets, three different options were examined: Bluetooth, local network over Wi-Fi, and remote servers. The preferred requirements were low latency and automatic discovery. Another factor taken into consideration was the environment in which the game will be played, which in this case was a primary school.

Comparison of technologies			
Factor	Wi-Fi	Remote server	Bluetooth
Latency	Low	High	Low
Reliability	High	Low	High
Ease of use	Medium	Low	Medium
Complexity	Medium	Low	High
Ease of testing	High	Medium	Low

**Table 2.1:** A comparison of different communication technologies for connecting the tablets

A total of five factors were considered when choosing the technology. *Ease of use* refers to the complexity that players face when setting up a game session. *Complexity*



refers to the work load as implementation of the technology is done by the developer. *Ease of testing* refers to the steps that have to be taken in order to test or debug the game while developing. The complete comparison can be found in table 2.1.

A remote server, while easy to set up for a developer, would be complex to connect to for the players. It also loses out to Bluetooth and Wi-Fi in all other factors. Latency is much higher, which can be considered the main factor as to why a remote server is out of the question. Low latency is key to create a seamless experience across the devices.

Bluetooth is a more promising technology, but the latency and ease of use are both on par with Wi-Fi. It is quite complex to implement, since our choice of development platform (Unity) does not support Bluetooth communication without plug-ins. Testing for Bluetooth is also more difficult than a network approach, since multi-device tests can only be run on Bluetooth devices.

With these factors in mind, and essentially by process of elimination, the technology chosen was local network communication using Wi-Fi. The technology allows for automatic host discovery using network-wide broadcasts. When used successfully, this can eliminate the need for the player to actively create a “lobby” and invite players, or connect to a specific remote server. The devices can communicate in the background and automatically designate one of them as the server, which then broadcasts its presence to the other tablets and tells them where to connect.

The only additional assumption that has to be made when using Wi-Fi is that all devices are connected to the same network. In our environment, a primary school, this is very likely to be the case.

## 2.6 Secondary development tools

While the majority of the game development was done in the Unity software, a number of additional tools were used to assist in the development. For writing the C# scripts used by Unity, Microsoft Visual Studio was used [23]. In prototyping, sketching and texturing, Adobe Photoshop was used [24]. Finally, since the game uses 3-dimensional objects, 3D modelling was done in Blender [25]. The reason for choosing Blender is that it is a powerful free open-source program that has been around for 20 years, and has a large online knowledge base [26].

Git was used as version control system and to keep the code synced between the group members [27]. The repository was remotely hosted on GitHub to allow for easy remote work [28]. GitKraken, a GUI Git client was used to simplify tasks that could be difficult to do from the command line, like merging branches [29].

# 3

## Prestudy

A prestudy was conducted in order to gather necessary knowledge about designing a collaborative game for children. The insight gained from this prestudy is later used to determine requirements for the final game concept. The chapter begins by presenting guidelines found in literature of how to design both for cooperation as well as for children. Following is an investigation of already existing collaborative games both from academia and commercially available. Finally the findings of the interviews performed at Johannebergskolan are presented.

### 3.1 Designing for cooperation

According to Bergström, there are three key ingredients needed to create a “group feeling” amongst players when designing table top role playing games, motivating them to work and perform as a team rather than individuals [30]. The three ingredients; “boundary to outsiders”, “mutual focus attention”, and “shared mood” are mainly derived from non-digital games, but they are still all reasonable to consider when creating a digital cooperation based game supposed to be played locally.

A boundary to outsiders marks out who is part of the team and who is not, and is important for team integrity [30]. Bergström explains this boundary to be partly tied to the rules of the game and the players’ knowledge of them, where a more complex rule set enforces the boundary, keeping outsiders from participating, while simpler rules make it easier for newcomers to join in.

A mutual focus attention is achieved by minimizing the “backstage work” (“backstage work” being keeping up with e.g. points and character changes in table top games) [30]. This is a lot easier when developing a digital game, as the computer in many cases can do these calculations automatically, but maximizing the time players concentrate on the same issues to enable cooperation is still important when developing gameplay mechanics.

Concerning the last ingredient, “shared mood”, Bergström writes “The more the players are focused on the game activity, the greater the push towards a common mood, which in turn, leads to greater focus” connecting shared mood to mutual focus [30]. Though he later points out that this process is far from automatic, as two participants can perceive the same activity differently, therefore developing

different moods (thus emphasizing the importance of designing towards creating a common mood).

When deciding the strictness of game rules for a cooperative game it is important not to set rules that are too strict. Doing so may result in inferior social interaction due to the players inability to tweak the game depending on situation and personal interests. An example of this would be when there is an inexperienced player in the group and the group may want to change certain game rules to fit those conditions. If the rules on the other hand are too loose the players may initially be confused, and overwhelmed with the responsibility of creating their own rules.[31]

## 3.2 Designing for children

The study of Computer Child Interaction (CCI) is a field of research within the broader Human Computer Interaction (HCI) [32]. As the name suggest, CCI concerns the design, implementation and evaluation of computer interfaces used by children. According to [32], the key differences between CCI and HCI are the rate of change in children, adult involvement in the children interaction, context of usage, as well as the cultural and social norms that determine what is good and brings value to children.

The design of computer systems meant to be used by children has to account for the rate of change, meaning the children's ability to quickly adapt to new knowledge. In [32], it is mentioned how the design of certain computer systems were a problem for children to use only five years ago but is used today without much difficulty. An example of this is the process of searching for information where kids now are accustomed to how to use the Google search engine to their benefit.

Another set of guidelines, proposed in [33], are now considered fundamental to the design of children centered computer systems. Although some of the guidelines concern domain specific issues within the design of construction kits, most of the guidelines are applicable to design in general. Out of the proposed guidelines, [33] considers "Design for Designers" to be the one of most importance. *Design for designers* is the idea of enriching learning experiences by allowing the user to actively design and create things. This creation should not follow strict instruction but instead allow for open-ended design, allowing the kids to learn the underlying idea.

## 3.3 Investigation of existing games

An investigation of already existing games within the genre was concluded in order to gather ideas on game concepts which could be of use when later creating one of our own. The games were analyzed using the gathered information both in regard to the target audience as well as to cooperation as a game design.

### 3.3.1 Games with an academic background

As mentioned in chapter 1, three previous research projects have been conducted exploring the development of co-located collaboration games across tablet devices. Although these were developed for children in the Swedish compulsory school for pupils with learning disabilities and as such had a different target audience, they provided some insight regarding which type of game concepts contributed mostly to the collaboration of the players. Following is a short description of each of the games:

- **CirKva**[7]: A logic based game where the goal is to move different characters across the tablet to specific locations. The characters are controlled by dragging them around using one's finger. The paths of the characters may be hindered by walls which are overcome by moving the characters to certain portals causing them to teleport to another location.
- **Prisma**[8]: A puzzle game where each player controls the laser beam of a spaceship. The goal is to direct these beams to target elements in the form of keys. The beams are primarily controlled by rotating the stationary spaceships but may also be further directed by movable mirrors.
- **LuTa**[9]: A board game with the goal of moving a ball through a labyrinth. The movement of the ball is controlled by tilting the board using a total of four buttons placed in the corners of the screens. Each player is in control of a single button causing the movement on the ball to be dependent on more than one player.

User testing of both CirKva and Prisma revealed that certain players understood the game faster and, as a result, dictated what others should do. In some cases these faster players even reached across devices in order to complete the action of their peers. Slower thinking players were notably frustrated and the situation had to be solved by the supervisors.

We believed the cause of the problem to be the lack of forced progression. As both of these games progressed only by the actions of the players, pace was entirely dictated by the players themselves. This meant that any difference in game knowledge between the players had to be neglected by forcing the faster solvers to wait for the slower solvers, which in terms could cause frustration for both slower and faster players. Although collaboration includes having to understand the differences in knowledge between one another, a game where there is a constant progression, i.e. a game played in real time, could lead to a more balanced collaboration.

### 3.3.2 Games on the market

A brief investigation was also made on the wider game market to see what kind of genre lends itself well to co-op (cooperative) games.

In total, five popular co-op games were dissected into their primary cooperative

qualities.

- **Keep Talking and Nobody Explodes**[34]: A game in which one person has the game in front of them (main player), and the other players only have a paper of instructions. The main player has to describe a bomb to the other players, while they read out instructions on how to defuse it.
- **Overcooked**[35]: The players are kitchen workers in a busy restaurant, moving plates, dishes and cooking food in order to serve all the guests.
- **Portal 2**[36]: A first-person two-player game in which a puzzle has to be solved to escape a maze. Both players can place portals to move objects or themselves around the maze.
- **Payday 2**[37]: A first-person game in which the players are criminals planning and executing a coup on banks or other valuable buildings.
- **Left 4 Dead 2**[38]: A first-person zombie survival game, in which four players work together to defend against incoming zombies.

Each of these games have interesting cooperative aspects. Our main focus in this investigation is to see which games *force collaboration* onto the players, and which simply have a cooperative aspect (encourages it). We find that all games are easier to play cooperatively, but only one (Portal 2) is impossible to play alone. The puzzles in Portal 2 are made in a way that requires both players to use their portals.

Keep Talking and Nobody Explodes, Payday 2 and Left 4 Dead 2 can be played by a single expert player, since there is nothing forcing other players to engage in the game.

Another interesting aspect is the *availability of information*. Only in Overcooked and Payday 2 is there a overhead map or shared view of the game to all players. The other games require players to communicate what information they are given on their respective screens.

Yet another aspect that can be considered is the pace of the game. All games but Portal 2 allow players to build a strategy in the beginning, but the game is rapid action after the players decide to start the round (i.e. a battle against time or enemies). This allows for both *strategic collaboration* and *impulsive collaboration* in the game.

In Portal 2, there is no outer threat to the players. It is a slow puzzle game, where the players may take their time to discuss how to solve the puzzle.

### 3.4 Interviews at Johannebergskolan

Semi-structured interviews performed with the pupils at Johannebergsskolan showed a particular interest among the children for games based on using your creativity, and building freely. Every pair interviewed mentioned either Minecraft[39] or Block craft 3D[40] as games they regularly play. Other creative games were also mentioned

frequently by the children, such as painting and puzzling games. One of the children mentioned Bloons Tower Defence[41] as a game they do not like, due to the repetitiveness of the gamepla without much action from the player.

All the children had their own iPad provided by the school, and as of such had basic understanding on how to use mobile devices. Six out of the ten children interviewed also had their own mobile phones.

The opinion about violence in video games was clearly mixed, with some children being very positive and others refusing to play games where you hurt others. When interviewing the teacher she stated that most of the boys find violent games interesting, but our interviews showed mixed opinions across all genders. This finding suggests that the intention of reducing or removing violence (as presented in section 1.3) will make the game accessible for more players.

Interviewing the teacher also revealed the need to practice cooperation among the children. She usually arranges the groups they are supposed to work in at school, and have to avoid putting non-cooperative children in the same group. She prefers letting them work in pairs as this usually causes less conflict than larger groups.

# 4

## Development of Slime Attack!

This chapter presents the development process of our game, Slime Attack! from the formation of requirements to the final concept. Reasoning behind several key concepts within the game were a large part of the development, and are featured heavily in the chapter.

### 4.1 Creating a game concept

The first task of the game development was naturally to establish a basic idea of what the game is supposed to be and how it will be played. Knowledge gained from the prestudy along with the intention of using several devices to encourage collaboration were used to form the requirements.

#### 4.1.1 Requirements for game concept

In order to begin to evaluate any game concept, certain requirements had to be set in order to ensure that the final product fulfilled the purpose of the project. In addition to specific game concept requirements, the limited scope of the project set the cornerstone. Evidently, the game has to be developed and fully functional within four months with only four programmers and no artists. With this in mind, here follows the final set of requirements:

- **Forcing collaboration:** The game needs to be difficult (or impossible) to complete without the contribution and collaboration of several players. This is essential as the purpose of the game itself is to force the collaboration of the players.
- **Active game progression:** An external threat from either time or enemies that requires players to continually stay active. Our prestudy showed that any game where the pace is dictated by the players themselves runs the risk of having a small set of players controlling the action of others.
- **Consistent from several angles:** As the game is played by four players, located at different locations around the devices, elements within the game need to be understood and accessible regardless of viewing angle.

- **Appropriate for target audience:** The game must not be too complex or information dense. It should also not contain elements of violence towards lifelike enemies which may otherwise typically be found in popular games. Favorably, the game would be rated PEGI-3 by the PEGI organization[42].
- **Level of required physical proximity:** The game should encourage the players to remain physically close. A game that could be played without the players necessarily staying close to each other fails to fulfill the purpose of using tablet devices without creating a mobile bubble.
- **Fun to play:** Ultimately, in order for the game to have the potential of being used outside of academic purposes, it must be fun for the players.

### 4.1.2 Brainstorming

Following the completion of the requirements, ideas needed to be conceived and evaluated against the requirements. To do this, a brainstorming session was held. Brainstorming is mentioned by Tracy Fullerton and Christopher Swain as an effective method of idea creating [43]. Though, in order to be effective, it is important that the session follows a set of best practises. These include: not discarding any ideas early on, quantity over quality, visualizing the ideas and limiting the length of session.

Several ideas were generated during the brainstorming session. After discarding game concepts which contained basic flaws, e.g. lack of collaboration, a total of four game concepts were used for further development and evaluation. These were as follows:

- **Snake** A version of the now classical game Snake made to work across several devices. Players either control the snake only on their own screen or only in a single direction.
- **Castle Defense** The screens make up a field with a base in the middle. Each player defends the base from enemies on a single screen. The mechanics of defending could be of various types such as shooting at or throwing enemies away.
- **Fast engine repairment** Similar in concept to Keep Talking and Nobody Explodes mentioned in section 3.3.2. Instead of the goal being to defuse a bomb, players would instead try and repair an engine with the time constraint being a angry customer.
- **Role-playing** The players each choose a character/class and have to combine their abilities to overcome obstacles in a fictional world.

### 4.1.3 Concept evaluation

After developing each of the concepts, each of them were evaluated according to the previously listed requirements.



Snake was considered early on as the rules of collaboration were distinct and perhaps even previously known by the children, thus eliminating the need of learning a new game which may otherwise postpone any collaboration. Snake, being an information-sparse game viewed from above, would also require no changes to make it suitable for viewing from different angles. Ultimately, this concept was discarded due to the fact that a single player could potentially reach across the devices and control the action of others.

The fast engine repairment would follow much the same idea as *Keep Talking and Nobody Explodes* (mentioned in section 3.3.2). When played correctly, collaboration would be intense and a vital part of success as none of the players has all the information. However, playing correctly also means that the information displayed on each device would need to be kept secret from the other players. This means that the goal of creating a unified playing field would be impossible. The game concept did also not strictly force the players to be co-located as the only means of communication were speech and could therefore be replaced with voice communication over the internet or telephone.

A role playing game would have enabled the kids to control characters in a virtual world. The characters would feature different skill sets vital for the completion of a common goal. Both *Payday 2* and *Left 4 Dead 2* (mentioned in section 3.3.2) feature some element of role playing as the players take on different characters. Actively giving players different roles forces collaboration effectively as you can easily provide challenges that require a combination of the characters' abilities to overcome. However, these types of games typically require extensive development as there are not only simple elements and mechanics but large amounts of content that need to be created. This was considered to be unfeasible with the given time constraint.

Castle defence is at this stage the only game concept left. The game concept could be adapted to several devices by showing a top-down view of the playing field. Enemies would move across devices thus forcing the players to be co-located in order to see the entire map. Early discussion on game mechanics regarding the defence of the base included tossing enemies away by swiping them with one's finger or to shoot them using cannons placed within the base. These mechanics, in combination with a great number of enemies would essentially make it impossible for a single player to control the action of others. The concept is similar to that of *Left 4 Dead 2* (discussed in 3.3.2), where players co-operate in killing zombies charging towards them. The sheer amount of enemies is virtually impossible to face by one player, and requires active participation by more players. This active variation of castle defence would also deal with the issue of repetitive game play without much action (as mentioned during the interview in section 3.4).

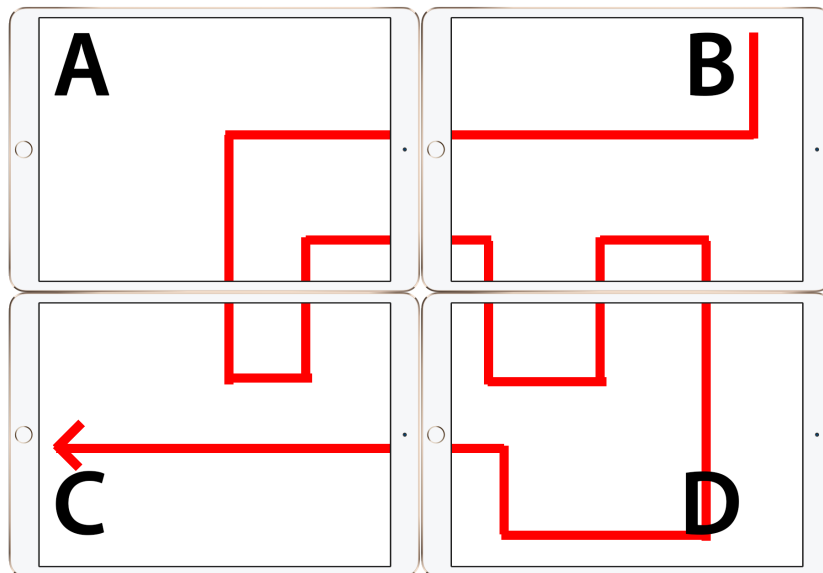
## 4.2 Concept development

The castle defense concept is an extension of the tower defense (TD) genre of video games. Games in this genre generally have a focus on strategy and base building.

The game is often a type of turn-based strategy game, divided into *waves* where enemies appear with pauses in between (*buy time*), where the player may buy upgrades. An example of a typical TD game is Bloons Tower Defense 4 [41].

### 4.2.1 Determining the target to defend

Implementing a typical TD game is not favourable when building for collaboration. In the typical TD game, enemies move in a maze towards a given goal. In our case, this means that the enemies would start on one iPad and move across the devices when laid out on a surface. While the playing field does get more complex over time, the game is restricted to one device in the beginning, which means one player will get to dictate the game. In figure 4.1, the player on iPad B would get to dictate the game for the first few rounds, while the other players remain inactive.



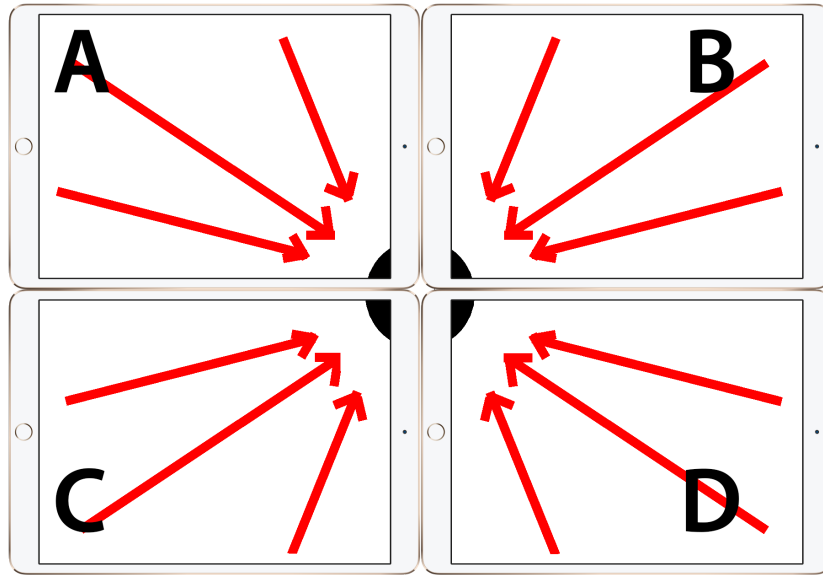
**Figure 4.1:** Traditional TD maze as seen on our screen constellation.

In order to invite all players to take part in the game from the beginning, an initial problem posed was where to place the enemies' target. A suggested solution is to place the target equally on all four devices, and let the enemies move towards this shared target. The obvious location to put this is in the center of the device constellation, since this contains the point shared by all devices. This center target is from here on referred to as *the base*.

Placing the target in the center of the playing field does not, however, solve the problem of enemies appearing only on one screen, thus providing one player with more action than the other players. Our suggested solution is to let enemies appear on the edge of all devices, and move towards the base.

This approach creates a symmetric playing field, which favours no single player with

more action than the others (see figure 4.2).



**Figure 4.2:** Symmetric playing field with target in center.

### 4.2.2 Building for active progression

The next issue that had to be examined was the pace and intensity of the game. As stated in our concept requirements, we would like to have active game progression. The TD genre has the player partake during the wave to a varying degree, ranging from completely passive gameplay to mandatory active participation.

Given our requirements, the concept being developed should be closer to the active end of the spectrum, with players having to continuously interact with the game during waves. In fact, had the game instead been on the passive end, it could be played by a single player. The forced active participation is what shapes the requirement for multiple players.

Instead of building a traditional TD game, we decided to go closer to the approach that can be seen in *Defend Your Castle*[44], where the player defends a stationary target (or base) from approaching enemies. The game relies heavily on the player killing enemies instead of an artificial agent, such as the monkeys seen in *Bloons Tower Defense*.

Having the players toss away the enemies from the target, similar to *Defend Your Castle*, gives the game a strong element of forced participation.

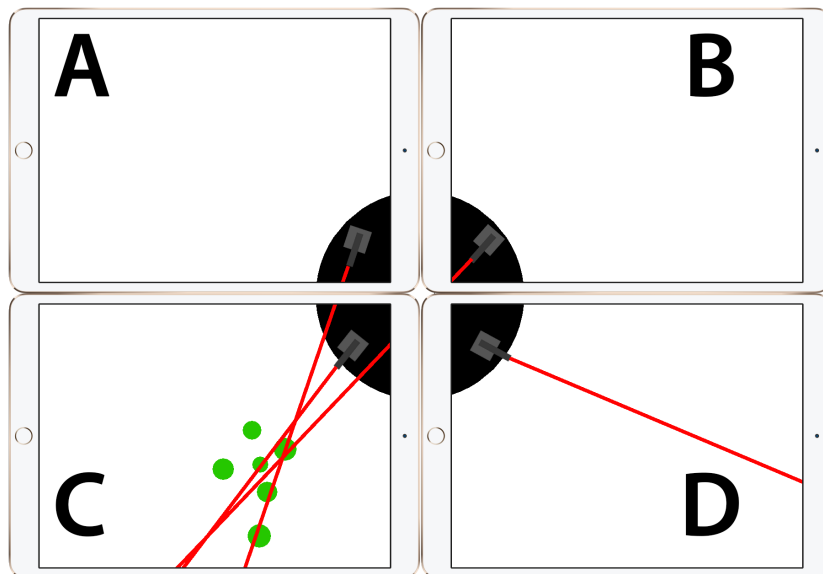
In further elaboration, we decided against the tossing of enemies, since the gameplay may feel monotone after some time, and the weapon upgrade path is not very clear. A different option that we decided to implement in its place was the laser cannons placed on the center base. These cannons can be upgraded, which adds to the progression of the game.

### 4.2.3 Encouraging interaction across devices

With the concept that we have developed so far, we have created a game that distributes tasks equally among the players and thus should see no one player taking control of the game. However, the changes that we have made to the basic TD concept have introduced a new problem. Since all device screens essentially look the same, and the enemies only move from one side of the screen to the target (in one of the corners), we are facing the problem of a low level of required physical proximity. By this, we mean that the game could be played without the tablets being placed in the constellation. Each player could sit with only their device and not share any of the information with other players, because the concept does not encourage it.

There are a number of possible solutions to this problem. One possible solution is to distribute enemies unevenly on the playing field, and let the players send resources or weapons to one another. This could be done by making the base more complex. We decided against this, since we would like to keep interactions in the center down to a minimum, because it may disturb the visibility of the playing field.

A spontaneous, and relatively simple, solution that we ended up implementing is to allow player cannons to face the other screens and shoot into another player's view. By randomly spreading the intensity of enemies on the playing field, players with low amount of action on their screen can instead turn the cannon to another screen. In figure 4.3, player A and B are helping player C by shooting the enemies on his screen.



**Figure 4.3:** Players can shoot across screens.

Another problem we saw with our symmetric playing field was the uniform movement of enemies. An enemy that appears on screen A will remain on screen A until shot. Having enemies move across screens will create even more incentive to keep the screens in the desired constellation.

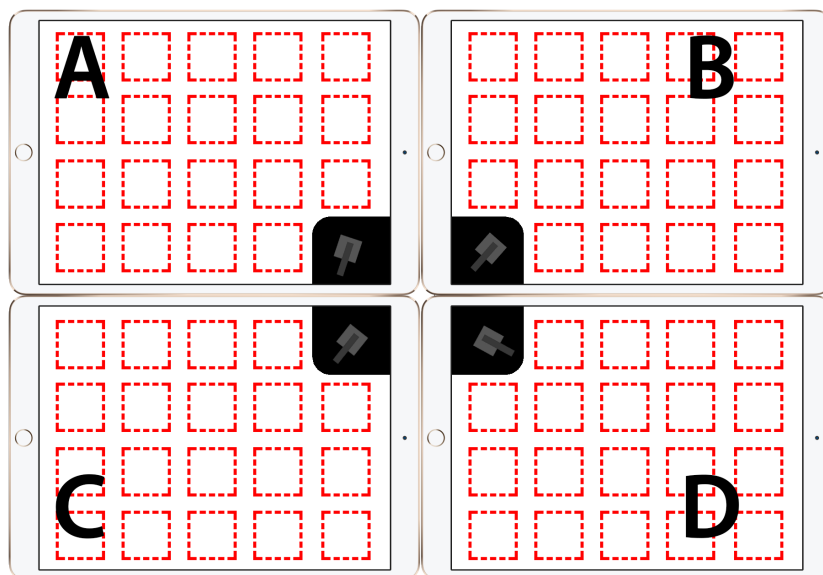
At this point, a number of possible solutions were discussed, including both moving the location of the base and reintroducing the maze pattern. However, this would also reintroduce the problems that we solved earlier.

The best solution we found to this problem was to greatly simplify the idea of the base. Previously, we mentioned the fact that upgrades are purchased between waves. These upgrades were simple attributes that improved the defense and attack properties of the base. Instead of buying upgrades to the base, we changed the concept to be focused on buying new *structures*. The base would remain the static, large structure in the center, while other smaller buildings can be built around the playing field. These structures give the players more powerful weapons and act as traps for enemies.

By letting enemies target a random structure instead of always targeting the base, we saw a much more unpredictable pattern of movement.

Solving this problem by implementing structures essentially killed two birds with one stone. By letting players build structures freely, we are also able to fulfill the “Design for designers” concept as presented in section 3.2.

As seen in both the literature study as well as the interviews, “designing for designers”, or designing to enable creativity, is an effective way of stimulating children’s desire to create. The lack of these elements in our concept was concerning, and it became clear that in order to keep the players’ interest over time, the game had to enable the players to create something of their own when progressing. This further motivates reevaluating the concept of the base, making it something the players build up over time rather than something they have from the start. Eventually this ended up in the idea of a grid system in which the players can build structures (figure 4.4).



**Figure 4.4:** Structures can be constructed on a grid.

Adding player-built structures to the concept was thought to give the players a feeling of creating their own base, and due to free placement in the grid also allowed every base to be unique to one play session. The structures were not made to stop incoming enemies from getting to the base, but still allowed for strategic placement due to A. the structures themselves having different benefits and B. the structures sharing the attention of enemies with the main base, thus enabling players to draw enemies away from the main base by placing structures far away, but at the same time leaving those structures very vulnerable.

### 4.3 User interface, graphical assets and sound

Due to the small scope of the project, the vast majority of time was put into developing a desirable game concept, rather than sound effects and graphical assets. Our assumption is that these can be kept relatively simple without negatively affecting the game.

Music and audio effects that were used in the game were gathered from the free section of the Unity Asset Store [45].

Three dimensional models and textures were, with few exceptions, created within Unity. All textures are simple one-colored materials.

The user interface was kept very simple, due to the multiple-angle constraint. A busy interface can also be overwhelming to a new user. The only interface visible throughout the game session is the counter on every device showing how much currency the players have gathered.

#### 4.3.1 Enemies

Some consideration was put into what the enemies should look and act like. By our requirements, we would like to avoid anthropomorphism, i.e. the players seeing enemies as human or assigning a value to their lives. Humans and animals were quickly discarded as possible enemies.

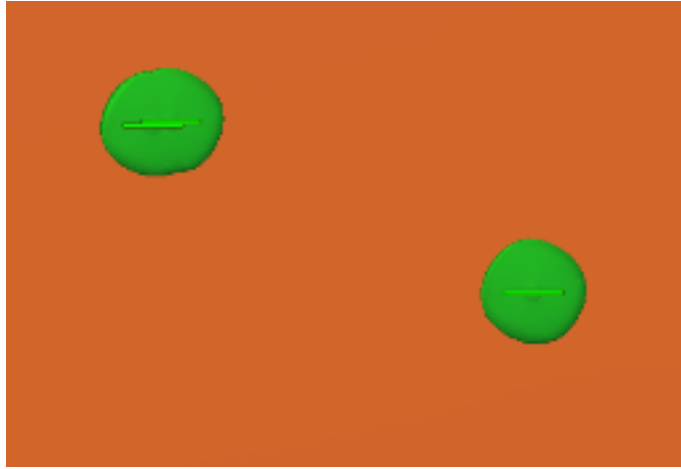
A compromise also had to be made as to how complicated we would make the enemy models. Not only would a more complex model require more time to produce, but it would also have an impact on the game performance, especially as the amount of enemies increases.

Our solution was to make the enemies *slimes*. These simple, green pebble shapes can be interpreted by the players as bacteria, aliens or whatever their imagination tells them.

The slimes were modeled and animated in Blender before being imported into the Unity project.

Enemies also have a *health bar* visible above them, indicating how much health they

have. The bar gets smaller as enemies take damage.



**Figure 4.5:** Three enemies or "slimes" as seen in game.

# 5

## Results

This chapter presents the design and implementation of the final version of the Slime attack! based on the requirements presented in section 4.1.1 and following the discussion of game elements provided throughout chapter 4. Some observations from user tests performed at Johannebergskolan are also presented briefly while the full transcript can be found in appendix C.

### 5.1 Game concept

The game Slime Attack! takes place in the future, around a human outpost on an alien planet. At the center of the playing field is a base, which houses the outpost's defenses. Four laser cannons are situated on top of the base, each controlled by one of the four players.

The goal in Slime Attack! is to survive for as long as possible. Slimes will keep appearing outside the visible area with the simple goal of destroying the human outpost. The players have to kill the slimes and are rewarded with the currency "unobtanium" for doing so. This currency can be used to improve the outpost.

### 5.2 Breakdown of a game session

Each game session can be broken down into four distinct stages.



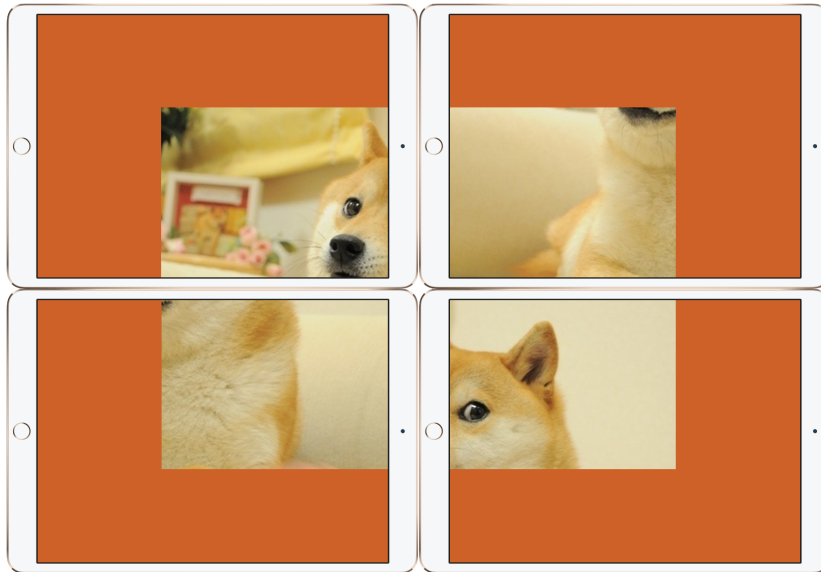
### 5.2.1 Setting up a connection



**Figure 5.1:** The first screen the players will see. A single button is present, when pressed will launch the set up of a session.

As the players start the game, they are presented with a simple screen (fig 5.1). There's only one button available, which starts a search for other players on the same network. After all four players have pressed this button, the setup is handled seamlessly.

## 5.2.2 Organizing the playing field



**Figure 5.2:** The puzzle stage of the game. Players need to arrange the tablets according to the puzzle on the screens.

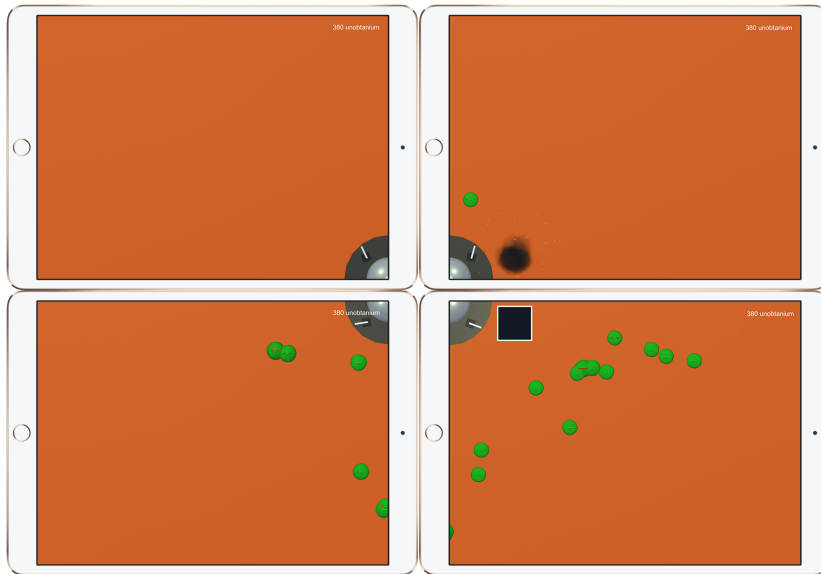
While the technical implementation handles network setup without the players intervening, the devices cannot know the physical position relative to one another. Without this organizing stage, the playing field would appear scrambled, as seen in figure 5.2.

At this point, the players have to physically move the iPads so that the picture of the dog is assembled correctly. Once this is done, each player taps their screen and the game commences.

## 5.2.3 The game

As discussed in 4.2, a tower defense game is generally divided into *waves* and *buy time*. This is also the case with Slime Attack!.

### 5.2.3.1 Wave



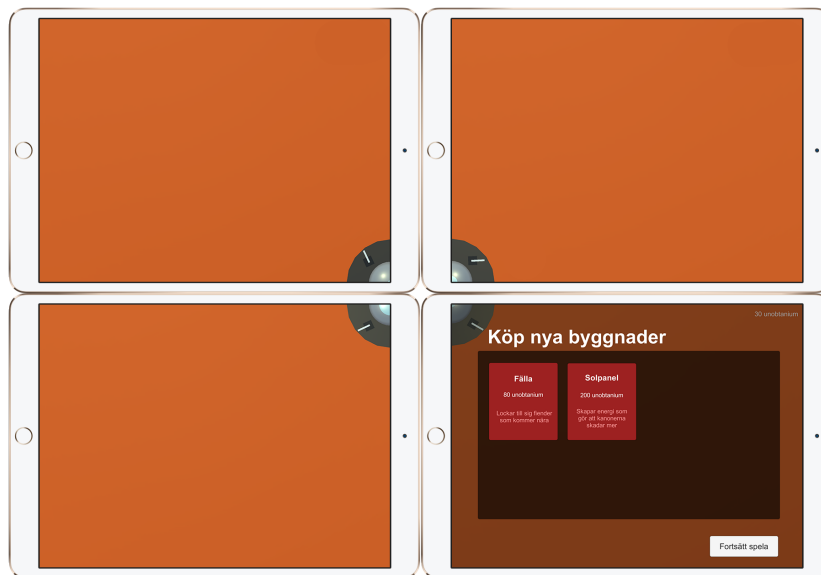
**Figure 5.3:** The game during a wave. Enemies (green) are seen attacking the base in the middle. A single solar panel is found near the base on the bottom-right.

A wave lasts between 15-30 seconds, depending on how quickly the enemies are killed. During a wave, the players can only fire the cannons by tapping or holding down on their screen. The player can tap anywhere on the screen, and the cannon will fire towards that point.

The amount of enemies, and the distribution across the playing field, is determined by the round number. After the players finish a wave, the round number increases. This means that there will be more enemies in every wave as the game progresses.

Killing an enemy will reward the players with a form of currency, that can be used to buy upgrades.

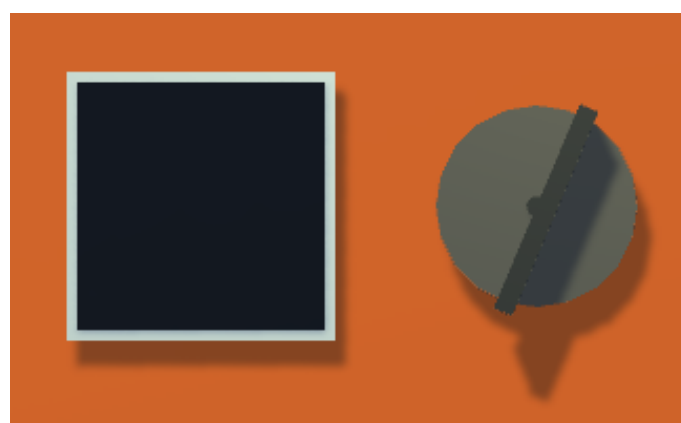
### 5.2.3.2 Buy time



**Figure 5.4:** The game during buy time. A interface allowing players to purchase buildings is seen on the bottom-right device.

As the last enemy is killed, the wave ends and buy time commences. There are no enemies on the screen during buy time, and it lasts for as long as the players choose. A shop screen is opened on one of the devices, the designated “shopper’s iPad” (bottom right in figure 5.4).

There are two upgrades that the players can purchase during buy time, the solar panel and the trap.

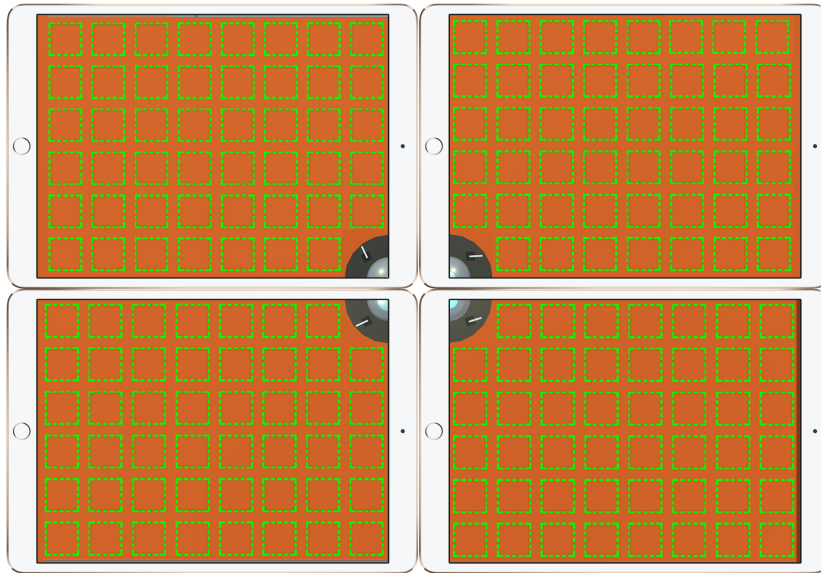


**Figure 5.5:** The two structures that can be purchased, a solar panel generating energy(left) and a trap luring enemies (left).

The trap (right hand side in fig 5.5) is the cheaper upgrade. The purpose is to bait

enemies into targeting it, instead of valuable structures. It has a relatively high amount of health, but will eventually be destroyed by the enemies that walk into it.

The solar panel (left hand side in fig 5.5) costs almost three times as much as the trap, so it can only be purchased in later rounds. Solar panels generate energy, which makes the laser cannons more powerful thus making it easier to defend against the enemies. However, solar panels are very vulnerable and easily destroyed by enemies.



**Figure 5.6:** The building grid. Building purchased during the buy time can only be placed in the designated squares.

After purchasing a new structure, the players are presented with the grid view of the playing field (figure 5.6). Pressing a square in the grid will place the newly bought structure at that location.

When the players feel ready, they may press the bottom right button to enter the next wave.

#### 5.2.4 Starting over

All good things must come to an end, and this game is no exception. As the waves get more difficult, the amount of enemies will eventually be overwhelming. The base at the center of the playing field has a limited amount of health, which decreases as enemies collide with it. When the health reaches zero, the game ends.

At the end of the game, all buildings explode and the players are presented with a simple “game over” screen and a “restart” button. Pressing this will clear the game and restart a game session. Notice that the players do not need to set up the connection or redo the puzzle, since this data is saved independent of game sessions.

## 5.3 Implementation details

In this section, we go into details about how the game is structured from a technical standpoint and motivate some of our design choices. A few unique solutions will be discussed in depth, while basic concepts will only be mentioned briefly.

### 5.3.1 Network communication

The largest problem that had to be solved in our implementation was the network communication between devices, both in configuring the connection and keeping a reliable connection throughout the session. As concluded in section 2.5, Wi-Fi is the best technology to use. However, Wi-Fi functions within a normal LAN, which means we still have to deal with technicalities such as servers, clients, ports and IP addresses. Preferably, all of this will be hidden from the players.

#### 5.3.1.1 Connecting the devices

Given the target audience, we made the conscious design choice that connecting the tablets has to be straightforward. The devices should negotiate in the background and decide which one of them becomes the server that the other clients can connect to. This is a known problem in academia called *leadership election* with many possible solutions [46]. The way we solved this was by creating a component we call the Host Finder. When a player taps “play”, the device will activate this component. The Host Finder begins by listening for broadcasts from existing servers on the local area network. If there is no existing server to join, the Host Finder will instead create a server and continuously broadcast its existence to other devices on the network. As soon as a player joins, the Host Finder has finished its job and will wait for a total of four players to join. On the contrary, if no players join, it will go back to searching for other servers again. Both the “searching for servers” and “becoming a server” stages last for a random period of time. The random duration is to avoid an overlap where devices constantly switch between server and client at the exact same time. With a random duration, the Host Finders on each device will eventually diverge and connect to one another.

#### 5.3.1.2 Synchronizing game state

As discussed in section 2.4, Unity abstracts away from low level network concepts and provides high level communication APIs for us to utilize. Unity has network components that handle synchronization of objects between devices. These components can also communicate over different types of network channels - either reliable or unreliable, the former requiring more system resources than the latter. At first, all communication was handled over a single reliable network channel that ensured arrival of the messages. Later on in the development as the game concept matured,

we realized that this wasn't feasible. Each player had a cannon that could shoot up to 10 times per second. These shots could be directed to any other screen, and as such they were sent to all the other devices for potential rendering. Similarly, the position of each moving enemy in the game was synchronized over this channel several times per second. This resulted in severe jitter in the movement of the enemies, and ultimately lag and freezing when a lot was going on at once. Some optimization was required to make the game playable.

We set it up to use two separate network channels - one reliable and one unreliable. The reliable channel was reserved for important messages, such as spawning a new enemy. We also made a custom component that only synchronized the velocity of the enemies instead of constantly synchronizing their position. As for the unreliable network channel, this was used for things that were deemed less important. An example of this is the rendering of the lasers from the player-controlled cannons. Even if a few shots were to disappear between the server and a client, the players would be none the wiser. Hit detection is calculated on the client side and sent to the server on the reliable channel, ensuring that the damage is still inflicted.

### 5.3.2 Making enemy waves interesting

A few issues had to be solved when building the enemy spawn system. Firstly, the enemies should be reasonably evenly spread out on the playing field so that each player feels that they are contributing. Secondly, the amount of enemies per wave should increase as the game progresses. Thirdly, the enemies should sometimes be concentrated to one area, in order to encourage the players to assist one another.

The final model of enemy spawning that we implemented can be described as a function that has one input variable: the number of the round.

The function first calculates the amount of enemies that should spawn in the round, using the following formula:

$$n = 2 + 9e^{0.3r}$$

Where  $r$  is the current round number, and  $n$  is the number of enemies. Notice the exponential nature of the equation, which means that the amount of enemies will increase greatly as the players progress through the game. In the first round, there are 14 enemies total, while in the eighth round, there are over 100 enemies.

After determining the amount of enemies, the spawn function builds groups of enemies that we call swarms. These swarms are a number of enemies located very closely together. Swarms are what create the unpredictability as the game progresses, and they require collaboration to be killed.

Swarms are very rare in the beginning of the game, but get more common as the game progresses. The size of a swarm is determined randomly, and the upper limit increases as the level increases. For example, in round 8 there is a 10% chance that a massive swarm of 70 enemies spawns.

After dividing the total amount of enemies into swarms, the function scatters them randomly within a certain minimum and maximum radius from the base.

The exponential nature of both total amount of enemies and swarm size makes the game extremely difficult once the players get to higher levels. This means that the players need to conceive a strategy and build upgrades in order to have a chance at completing later rounds. It also functions as a limiter to the length of a play session. With a slower increase in the amount of enemies, sessions could potentially last for hours.

### 5.3.3 Balancing the economy

In order for the game progression to feel natural, the economy has to be reasonably structured. We decided on the simple approach of rewarding the players with a fixed 10 points when an enemy was killed. Thus, the amount of money gathered in a round was quite predictable.

Knowing the amount of money that the players can gather in a round, this could be used to set the pricing of the upgrades. The solar panels are very good, but not necessary in the early game. The pricing was thus set to be higher than the amount of money that could be gathered in round 1, making them impossible to buy instantly (at 200 unobtanium).

The traps do not impact the game as much as the solar panels, unless there are a great amount of traps on the playing field. They can be priced rather cheaply (80 unobtanium).

### 5.3.4 Determining enemy targets

When an enemy spawns, it's not inherently clear where it should be heading. As discussed in 4.2.3, the goal is to make the enemy movement more interesting than just moving towards the center base (which ends the game when destroyed). Each structure is thus assigned a weight. When an enemy spawns, its target is determined randomly based on the the relative weight of all structures.

A trap is very likely to be targeted, with double the weight of the base. A solar panel is not very likely to be targeted, with only half the weight of the base. The vulnerability and high cost of the solar panel means that any enemies approaching it have to be killed regardless.

### 5.3.5 Function of the solar panel

The solar panels produce energy. The energy is stored in the base, and visualized by the blue sphere in the middle of the base. The sphere grows and shrinks depending on how much energy is available. Energy is used by laser cannons. Each shot will slightly decrease the energy in storage. The more energy is in storage, the more



damage shots will do to enemies. With no energy, a cannon requires 10 shots to kill an enemy, while it only requires 2 shots with full energy.

## 5.4 Results from user tests

The performed user tests clearly highlighted some of the major issues with the current game design, but also showed a lot of positive results. The results from testing with the three groups were:

All groups quickly solved the puzzle to organize the screens, but only one group intuitively understood how to start the game when the puzzle was in order.

All groups played their first session without using or reflecting over the buy time, quickly pressing the “continue” button as fast as they saw it. Eager to shoot more enemies they did not take any time to read descriptions for buildings or explore functionality further than shooting and aiming with the cannons. As the waves got more difficult, most groups started exploring functions during buy time to look for solutions, trying out building structures, though not always taking their time to read what the structures actually do. One group played two sessions without figuring out how to build structures, clearly highlighting the issue of communicating the importance of the buy time.

When the groups started building constructions however, they progressed further and started discussing where and what to build. During the first session, the two groups who discovered the building function were initially confused about who got to build and when, and competed about pressing first when the grid showed, so they would get the building on their side. During the second and third session however, both of these groups developed rules within the group controlling who got to build and when, one group taking turns between players whenever they had enough resources to build.

In an interesting example, one group always saved up enough resources to buy one structure for each player at the same time. When structures got destroyed on one players side, the groups discussed whether or not this player were allowed an extra replacement building, creating rules about responsibility over your own buildings as well as shared responsibility within the group to help each other out.

The first two groups also discussed and tried out different strategies for where to place structures, trying out building them far away, close to the base, as well as building only traps on one device and solar panels on all others. Between waves, pros and cons of each strategy were discussed within the group. Remarkably, during the last session the groups would spend about as much time, if not more, during buy time as during waves.

Even further, at the end of the session, one group had realized that solar panels are best built close to the base, while traps should be built far away from the base. One player suggested that solar panels are necessary to kill all the enemies and should be well protected, so the panels should be placed as far away from the enemies as

possible, i.e. next to the base. As this strategy was suggested, the other players quickly agreed and this was the new strategy of the group.

All groups quickly understood how to shoot across to the other player's devices, and the initial reactions were often competitive, with players trying to "steal" kills from each other when they ran out of slime on their own device. This resulted in some players pulling their devices out of formation to avoid other shooting onto their device. When the waves got harder they instead started helping each other out, calling out for help when needed and shooting across devices whenever they had time to spare. All players who initially removed their devices from formation later returned in order to be able to help and get help from others.

During the first session, players did not realize that they all had to protect the base. Notably, they used terms such as "your base is about to be destroyed". Eventually, some players would come to the realization that they would all lose if one player lost, and this helped spur on the cooperative aspects.

Cooperation was especially clear during the more difficult swarm waves, where all players had to help out on one device in order to survive. At later stages, players let their own buildings get destroyed as their firepower was needed to save the base from swarms on another players screen, recognizing the greater threat.

On one occasion a player accidentally exited the ongoing session by pressing the home button on their device. They were then allowed to start a session by themselves, playing on only one device. One of the three remaining players in the original session wanted to try playing alone as well and deliberately pressed the home button and started a session alone as well. Interestingly, both players quickly got bored and started calling out to the others to let them join again. They quickly realized how difficult it was to defend the base alone.

Concerning entertainment, during the first session (especially the beginning when the waves are easy) some players expressed boredom, and got annoyed when they ran out of slime. This changed however as they discovered the usefulness of buy time and their mindset became more team based, and over the course of each test all players became more and more involved and emotionally invested in the sessions. Towards the end of each test, all players were dedicated to making it through as many waves as possible together and were seemingly having fun, expressing joy as they defeated waves and heavily engaging in discussions concerning strategies during buy time.

Another interesting observation was that the players seemingly all thought about the best strategies as they played, and presented their ideas during buy time. Even the observably more introverted and generally quiet players would raise their voice and share their thoughts on how to distribute the gathered resources.

# 6

## Discussion

This chapter brings up a discussion of how well the final version of Slime Attack! based on the findings of the user test and the further improvements that could be made. Reflections on development process as well as the choice of development platform are also presented. Finally, suggestions of future research within the field of collaborative games are mentioned.

### 6.1 Lessons from user testing

The difficulty of the game is crucial to force cooperation and create a team mindset. In the early stages, and especially when the players did not know the purpose, their instincts lead them to a single player mindset. It's only when the difficulty increases that the players realize they have to help each other.

Shared resources allow the players to develop their own rules within the group to delegate resources. The most obvious example is that the players quickly realize that the money is a shared resource, and they buy things together. They did not have a clear understanding of what the buildings do, but they instinctively wanted to save the buildings on their screen, and they were just in how they divided the resources.

Interestingly, the players did not seem to realize that the structures would reward everyone equally, so everyone should be concerned with saving the structures, regardless of whose device they are located on. Instead, they formed the mindset of each player only benefiting from their "own" structures, which seems to have caused the careful distribution of new structures to keep everything fair.

The uneven distribution of enemies encourages players to develop social norms of responsibility within the group, governing when you are supposed to help others. We observed that the enemy spawn patterns worked as intended. At times, players would have no action on their screen, so they helped their teammates instead. When a player had a lot of enemies on their screen, they would call out for help from the others. This became especially clear in later rounds, when larger swarms of enemies form. It's simply impossible for one player to kill a large swarm without help. The further into the game, the less likely it is that a group would survive without cooperation, as intended.

Cooperation within the group is a great stimulant for player entertainment. From our observations, the apparent instinct was to play the game as any single player game. When played like this, the only stimulant is killing enemies that enter the screen. As the players realized that it's much easier to cooperate, the group were more focused on helping each other and building strategies, essentially rating the shooting of enemies as a chore, rather than the main goal. The goal instead seems to have shifted into building a better base and surviving the coming round.

The lack of “correct” places to build structures allow for exploration of strategies. The players are free to place the structures wherever they wish to on the playing field. However, the solar panel and the trap serve two very different purposes. As the players realized what the structures did, they begun trying different strategies. The group that explored this concept in most detail realized the optimal placement of buildings, much to our surprise. Their reasoning behind why they built the structures where they did was not clear, but it seemed as if all players had the same instinct, and they did not discuss in depth as to why this was the best strategy.

## **6.2 Suggested improvements**

Following the outcome of the user tests, there are a number of changes that can be made to the game to vastly improve any future sessions.

### **6.2.1 User interface and user experience**

The one area that we feel could have vastly improved the user tests is the UI/UX. The most obvious part that has to be changed is the shopping menu, seen in fig 6.1. As we built the game, the menu was built to look essentially as any game menu does, without much further thought. What we did not realize is that the target audience see the menu in a completely different light. Our suspicion is that the screen looks to them as a full-page ad that is found in most free mobile games. There is a lot of text visible, which could be too overwhelming for the children. In all test groups, their instinct was to look for the “continue” button and press it as fast as possible.



**Figure 6.1:** A close-up of the shopping menu during buy time.

Our suggested improvements to the menu is to add pictures of the structures to the menu, instead of the description. The function that the description fills would be reduced to a simple attribute. For example, the solar panel would feature a “+attack” attribute. We would also make the money indicator more clearly visible during buy time, and highlight structures that the players can afford.

Another part of the UI that could easily be improved is the puzzle stage. In our prototype, it is far from obvious that each player has to tap their screen to start the game after the puzzle has been solved. A simple text or animation would solve this problem.

It’s also not obvious to the players that the base is one unit with shared health across all devices, and that they all suffer when an enemy collides with the base. Our suggestion is to add a clearly visible effect across the devices that the base takes damage. A simple sound and a flash in colors would indicate that the base takes damage.

As for the money, the implementation worked quite well. Players quickly understood that they all had a shared economy, and they could all read the money count visible on every device. However, our initial specification asked for an interface that can be read from any angle, and the simple text does not fulfill this requirement. A suggested simple solution is to flip the text on the upper two iPads in the constellation. Another solution that was discussed during development was to include a physical bank in the playing field that displayed the amount of money, instead of showing the amount on each screen.

### 6.2.2 Improved structures

The main issue we found with the structures was that it was not clear to the players what effect they gave. The solar panel is quite complex in the current implementation. There are a lot of subtleties that were not noticed by the players, such as the growing and shrinking energy sphere in the base and the changing sounds of the lasers depending on energy charge. The concept of “energy” could probably be simplified for the target demographic. Solar panels could simply directly add attack damage to the laser cannons.

The trap structure worked as expected. Players understood the idea of a trap, and built them accordingly. The one improvement we see is to slightly change the traps to attract all enemies that within a certain radius from it. This would open up even more possibilities for strategies in later game rounds.

We also envision even more structures that could be built to enhance the late game even further. One suggestion is the anti-matter factory, which gradually creates anti-matter, used to kill a large amount of enemies within a certain area. This would open up for more discussion and strategy during the waves. The players would have to decide when and where to use this special weapon.

Another suggested structure is the turret, which is an autonomous cannon that shoots any enemies nearby. This would make the game a lot easier in later rounds. However, it also has the possibility of making the game easier to play without cooperation, so the balance would be important.

### 6.2.3 Bugs and performance issues

An issue that has been observed at rare times is when four clients start the game at once. When two devices each turn into a server simultaneously and broadcast their existence, unfortunate timing with the broadcasts can cause the remaining two clients to connect to separate servers. This in turn halts the Host Finder on each device and the four players never get connected to each other. The issue could be resolved by randomly closing the connection if only one client is connected to a server.

Another connectivity issue is when a player leaves the game, either by accidentally pressing the home button or the application crashing, and later rejoins. This event causes the newly rejoined player to be placed on top of another player resulting in one area of the playing field to be inaccessible. This problem could be solved by storing the device id and its camera position on the playing field thus making it possible to determine where the player should be placed.

The use of two separate network channels greatly improved performance overall. However, there are still performance issues in the game. As the game progresses and the amount of enemies increase, the devices have a hard time keeping up and the same jitter we observed with the initial design becomes noticeable once more. A possible solution is to mostly increase the size (and health) of enemies instead of

exponentially increasing the enemy count each round.

### 6.2.4 Game tutorial

A common way to introduce new players to a game is through a tutorial, that explains how to interact with the game. A tutorial prior to starting the game would solve a lot of the problems we faced, such as players not realizing they can press anywhere on the screen to aim the cannon, or the purpose of the buy time. One of the previously developed collaborative games, Prisma, included a tutorial level explaining the game mechanics and key concepts before introducing the players to collaborative play. This style of tutorial is sometimes referred to as “the tutorial room” [47]. Using a tutorial room does however conflict with some of the best practices for tutorials such as short burst of just-in-time instructions and gradual fade in to free play [48]. A better alternative would be to incorporate small visual cues as new concepts are introduced.

What we found during user tests was a gradual learning curve as the players explored the functions of the game. In fact, the game is inherently built with slow waves in the beginning and more action as it progresses, as well as the gradual availability of new structures. This naturally gives the players time to learn the game while it is relatively simple without requiring a tutorial. By the point the players reach more complex gameplay, they have already assessed the basics.

The argument could also be made that a tutorial would obstruct the collaborative nature of the game, since new players would learn from the instructions instead of from their peers. The natural learning curve that exists in the current implementation allows the players to learn the game functions together, and develops the group mentality. This concept is far from new, and is referred to as the “No tutorial method” by Paul Suddaby[47]. The method can be very difficult to implement, but we believe that the simple nature of the game lends our concept very well to this type of tutorial. The issues that a tutorial addresses could essentially be resolved by improving the UI/UX of the game, as discussed in section 6.2.1.

## 6.3 Evaluation of the process

As mentioned in section 2.3, the project has used an Agile development approach because of the initial lack of domain knowledge. In hindsight, this is a choice which have most likely had a positive effect on the outcome of the project. Short iterations has allowed for quick changes in regard to both the feedback gained form play tests conducted during the development as well as to suggestions from our supervisor. The short iterations also made sure that regular reviews were set up concerning both specific game behavior and progress towards the overall goal of the project. This in terms helped the project to stay on course and not pursue goals outside of the defined purpose.

Another helpful practice was one of the Scrum specific artifacts, namely user stories (see section 2.3). Due to the fact that none of the project members were from the target audience of the game, functionality could not necessarily be adapted to the experience and knowledge of those who developed it. User stories therefore served as a tool for addressing issues and functionality in term of what the actual intended user needs and expects.

The limitations of Scrum was clear in the initial part of the project where focus was on trying to gather as much information as possible. Defining small and discrete goals was at this point a difficult task to do as the amount of information and the possible outcomes where impossible to predict.

Another aspect which proved to be challenging was the collaboration with the school class. Getting in contact with a school and a class of children ready for collaboration was a tedious process which involved several failed attempts. Once a connection had been established, considerable scheduling issues due to the intense workload experienced by the teacher caused further delays.

In the initial plans, two tests, in addition to the interviews, were to be conducted as a measure to gain insight from the intended users and adapt the game accordingly. Though, due to the previous mentioned issues regarding collaboration, this had to later be changed. As a replacement, we ourselves played the game in order to identify larger issues concerning the overall game concept and design. These tests helped the ongoing development but were ultimately not as useful as the late final test of the game on the intended users.

## 6.4 Choice of device communication

From our testing, it appears that communicating solely over LAN using Wi-Fi was a sensible decision. We did not notice any direct issues in the environments we tested, which included mobile hotspots, primary school and university networks. In theory however, issues could arise in larger networks where devices can be assigned to separate subnets or virtual LANs that hinder communication between the devices.

## 6.5 Development platform

Choosing Unity as our main development platform proved to be a sensible decision. Not only did the development process speed up immensely by using an existing game engine with support for tablets, but it also simplified taking care of emerging problems thanks to the user-friendliness of the platform. Unity certainly is the ideal platform for smaller projects created by smaller teams with previous scripting experience.

Blender turned out to be quite complex and a lot of time was needed to understand enough to be able to utilize it the way we desired. The relatively long amount of



time spent learning Blender functions should probably have been spent elsewhere, considering the simplicity of what we wanted to achieve. In the end, Blender modeling was only used to create the slime model, with its moving and hit animations. Even though there may exist simpler 3D modeling software that presumably would be enough for what we wanted to accomplish the end result still turned out to be sufficient.

## 6.6 Outcome of social and ethical aspects

The development of Slime Attack! was not a project without any potential side effects. As mentioned in section 1.3, problems such as exclusion and gaming addiction are issues of social and ethical aspect that the design and usage of our game need to account for. The outcome of these aspects were evaluated both during the test of the game with the target audience as well as throughout the development process. Although the full extent of these issues need to be established over time, certain aspects could be evaluated in the scope of the project.

Exclusion, in the form of certain children being left out from playing the game with the rest of the group, was an aspect that the controlled environment of the final test could not assess. Such analysis would require a longer study where the game is incorporated in the education itself. However, exclusion in the form of certain children being left out of the collaboration whilst playing the game was an aspect that could be assessed. The final test clearly showed that all players were actively involved and contributed to the success of group. Even players who initially were somewhat silent came to raise their voice and ideas concerning strategical decisions.

Game addiction on the other hand were primarily addressed by not having elements rewarding a grinding behaviour. By design, Slime Attack! did not contain any way for the player to gain an advantage (such as better laser or structures) in forthcoming game sessions by increasing the amount of time played. The only advantage of increased play time is the general increase of skill. Though, given the simple mechanics of the game, could be seen as rather small increase. Our test also showed that the maximum length of a game session was around 20 minutes which is short enough to not intervene with other school activities.

## 6.7 Future research

As mentioned in section 6.1, the results of the user test of Slime Attack! showed positive results both in regard to collaboration as well as the outcome of negative aspects. However, it is important to notice that these tests were limited in scope as the testing was only performed during a single event. As such, the findings presented should not be considered to be generally applicable but instead more of a glimpse on the more general effects.

A natural continuation of our work would be to use Slime Attack! as a part of

normal education for a selected class. By employing the game in the education, longitudinal effects of a usage within a natural environment could be studied. This would also allow the further study of exclusion as an effect of the game due to the fact that the children themselves may have to choose the groups of play.

Previous research on the effects of introducing tablets in other parts of education has shown that the students perception of increased performance is not a guarantee for improved results [49]. As such, in order to motivate the future development and research of collaborative games, the link between collaboration in games and their effects on the increase in collaboration among players outside of the game need to be established.

# 7

## Conclusion

The purpose of this project was to design, create and implement a game for children in the ages of 7-9 that forced collaboration between the players. The game should be played across several devices in order to facilitate the usage of tablets outside of single user interaction.

The resulting game is called Slime Attack!. Slime Attack! is a tower defense inspired game where players are defending their base from incoming enemies using both laser cannons and player built structures. Collaboration is encouraged through the movement of the enemies across the screens and through the strategic decisions involved in the purchase and placement of new structures.

User testing showed that the game, whilst at first being played as a single player game, eventually encouraged the players to collaborate. Additionally, the test revealed areas that further development could improve upon. These include a better interface when buying structures as well as a clearer introduction to new players.

Issues encountered during the project mainly involves the set up of, and continued collaboration with a school class. Several failed attempts of getting in contact with a devoted teacher lead to the postponing of user tests. This, in terms, forced the development of Slime Attack! to initially rely on tests conducted on users who were not of the target audience.

The project enables future research studying the long term effects of using Slime Attack! as a part of education, both in regard to positive outcomes (increased collaboration) and to any eventual negative outcomes such as gaming addiction. Further research is also needed within the domain of collaborative games to establish whether the usage of these applications actually increase collaboration outside the narrow domain of mobile gaming.

# Bibliography

- [1] Gartner, “Number of smartphones sold to end users worldwide from 2007 to 2016 (in million units),” [Online]. Available: <https://www-statista-com.proxy.lib.chalmers.se/statistics/263437/global-smartphone-sales-to-end-users-since-2007/>, 2016, accessed on: Feb 15, 2018.
- [2] S. Lundgren and O. Torgersson, “Bursting the mobile bubble,” unpublished.
- [3] Skolverket, “It-användning och it-kompetens i skolan,” [Online]. Available: <https://www.skolverket.se/publikationer?id=3617>, 2016, accessed on: Feb 16, 2018.
- [4] W. Barendregt, P. Börjesson, E. Eriksson, and O. Torgersson, “Stringforce: A forced collaborative interaction game for special education,” in *Proceedings of the 2017 Conference on Interaction Design and Children*, ser. IDC '17. New York, NY, USA: ACM, 2017, pp. 713–716. [Online]. Available: <http://doi.acm.org/10.1145/3078072.3091987>
- [5] Skolverket, “Läroplan för grundskolan samt för förskoleklassen och fritidshemmet,” [Online]. Available: [https://www.skolverket.se/laroplaner-amnen-och-kurser/grundskoleutbildning/grundskola/laroplan#anchor\\_5](https://www.skolverket.se/laroplaner-amnen-och-kurser/grundskoleutbildning/grundskola/laroplan#anchor_5), 2017, accessed on: Feb 24, 2018.
- [6] R. L. Mandryk, K. M. Inkpen, M. Bilezikjian, S. R. Klemmer, and J. A. Landay, “Supporting children’s collaboration across handheld computers,” in *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, ser. CHI EA '01. New York, NY, USA: ACM, 2001, pp. 255–256. [Online]. Available: <http://doi.acm.org/10.1145/634067.634219>
- [7] A. Alvmo, A. Belfrage, A. Henriksson, E. Hromic, L. Lipkin, and M. Lönn, “Cirkva - ett spel som sträcker sig över flera ipads, anpassat för barn med kognitiva funktionsnedsättningar,” BSc Thesis, Department of Computer Science and Engineering, Chalmers University of Technology, 2016.
- [8] E. Axel, A. Berneryd, N. Rosengren, H. Tran, and M. Tran, “Prisma ett pekskärm-baserat samarbetspel för barn med kognitiva funktionshinder,” BSc Thesis, Department of Computer Science and Engineering, Chalmers University of Technology, 2017, 67.
- [9] J. Carlshede, E. Karlkvist, A. Kjellgren, J. Linder, H. Rönström, and O. Trens, “Luta: A collaborative game designed for children with cognitive impairments, using multiple tablets to create a unified playing field,” BSc Thesis, Department

- of Applied Information Technology, Chalmers University of Technology, 2017.
- [10] S. McLeod, “Preoperational stage,” [Online]. Available: <https://www.simplypsychology.org/preoperational.html>, 2014, accessed on: Feb 20, 2018.
- [11] World Health Organization, “Gaming disorder,” [Online]. Available: <http://www.who.int/features/qa/gaming-disorder/en/>, 2018, accessed on: Jan 25, 2018.
- [12] World Health Organisation, “Icd-11 (mortality and morbidity statistics): 6c51 gaming disorder),” [Online]. Available: <https://icd.who.int/dev11/l-m/en#/http://id.who.int/icd/entity/1448597234>, 2018, accessed on: Jan 25, 2018.
- [13] J. Hilgard, C. Engelhardt, and B. Bartholow, “Individual differences in motives, preferences, and pathology in video games: the gaming attitudes, motives, and experiences scales (games),” *Frontiers in Psychology*, vol. 4, p. 608, 2013. [Online]. Available: <https://www.frontiersin.org/article/10.3389/fpsyg.2013.00608>
- [14] American Psychological Association, “Resolution on violent video games,” [Online]. Available: <http://www.apa.org/about/policy/violent-video-games.aspx>, accessed on: May 28, 2018.
- [15] W. DeCamp, “Impersonal agencies of communication: Comparing the effects of video games and other risk factors on violence,” *Psychology of Popular Media Culture*, vol. 4, no. 2, pp. 296–304, 7 2015, [Online]. Available: <http://psycnet.apa.org/record/2014-24371-001>, accessed on: May 30, 2018.
- [16] NORDICOM, “Share of children who have access to tablets in sweden in selected years from 2011 to 2016,” [Online]. Available: <https://www.statista.com/statistics/690928/children-s-access-to-tablets-in-sweden/>, 2016, accessed on: Feb 15, 2018.
- [17] Å. Nilsson, Å. Ericson, and P. Törlind, *Design: process och metod*. Lund, Sverige: Studentlitteratur, 2015, accessed on Feb 8, 2018. [Online]. Available: <https://books.google.se/books?id=XhzGjwEACAAJ>
- [18] S. Kvale, *Den kvalitativa forskningsintervjun : Steinar Kvale : översättning Sven-Erik Torhell*. Lund, Sverige: Studenlitteratur, 1997, accessed on Feb 8, 2018. [Online]. Available: <https://books.google.se/books?id=IU49HwAACAAJ>
- [19] Agile Manifesto, “Principles behind the agile manifesto,” [Online]. Available: <http://agilemanifesto.org/principles.html>, 2001, accessed on: Jan 19, 2018.
- [20] Tutorialspoint, “Sdlc - waterfall model,” [Online]. Available: [https://www.tutorialspoint.com/sdlc/sdlc\\_waterfall\\_model.htm](https://www.tutorialspoint.com/sdlc/sdlc_waterfall_model.htm), 2018, accessed on: Mar 25, 2018.
- [21] J. S. Ken Schwaber, “The scrum guide,” [Online]. Available: <http://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-US.pdf>, 2017, accessed on: Jan 19, 2018.
- [22] Unity Technologies, “Unity user manual (2017.3),” [Online]. Available: <https://>

- [//docs.unity3d.com/Manual/index.html](https://docs.unity3d.com/Manual/index.html), 2017, accessed on: Feb 27, 2018.
- [23] Microsoft Corporation, “Visual studio ide,” [Online]. Available: <https://www.visualstudio.com/vs/>, 2018, accessed on: May 8, 2018.
- [24] Adobe Systems Inc., “Adobe photoshop,” [Online]. Available: <https://www.adobe.com/products/photoshop.html>, 2018, accessed on: May 8, 2018.
- [25] Blender Foundation, “About - blender.org,” [Online]. Available: <https://www.blender.org/about>, 2018, accessed on: May 8, 2018.
- [26] B. Veldhuizen, “Happy birthday, blender!” [Online]. Available: <https://www.blendernation.com/2014/01/02/happy-birthday-blender>, 2014, accessed on: May 8, 2018.
- [27] S. Chacon, “Git,” [Online]. Available: <https://git-scm.com/>, 2018, accessed on: May 8, 2018.
- [28] Github Inc., “Github,” [Online]. Available: <https://github.com/about>, accessed on: May 8, 2018.
- [29] Axosoft LLC, “Best git client 2018,” [Online]. Available: <https://www.gitkraken.com/git-client>, accessed on: May 8, 2018.
- [30] K. J. Bergström, “Playing for togetherness,” Ph.D. dissertation, Department of Applied Information Technology, University of Gothenburg, 2012, [Online]. Available: <http://hdl.handle.net/2077/29072>.
- [31] I. R. M. G. B. M. Yan Xu, Evan Barba, “Chores are fun: Understanding social play in board games for digital tabletop game design,” 2011.
- [32] J. C. Read and M. M. Bekker, “The nature of child computer interaction,” in *Proceedings of the 25th BCS Conference on Human-Computer Interaction*, ser. BCS-HCI '11. Swinton, UK, UK: British Computer Society, 2011, pp. 163–170. [Online]. Available: <http://dl.acm.org/citation.cfm?id=2305316.2305348>
- [33] M. Resnick and B. Silverman, “Some reflections on designing construction kits for kids,” in *Proceedings of the 2005 Conference on Interaction Design and Children*, ser. IDC '05. New York, NY, USA: ACM, 2005, pp. 117–122. [Online]. Available: <http://doi.acm.org/10.1145/1109540.1109556>
- [34] Steam, “Keep talking and nobody explodes,” [Online]. Available: [https://store.steampowered.com/app/341800/Keep\\_Talking\\_and\\_Nobody\\_Explodes/](https://store.steampowered.com/app/341800/Keep_Talking_and_Nobody_Explodes/), 2015, accessed on: Feb 12, 2018.
- [35] —, “Overcooked,” [Online]. Available: <https://store.steampowered.com/app/448510/Overcooked/>, 2016, accessed on: Feb 12, 2018.
- [36] —, “Portal 2,” [Online]. Available: [https://store.steampowered.com/app/620/Portal\\_2/](https://store.steampowered.com/app/620/Portal_2/), 2011, accessed on: Feb 12, 2018.
- [37] —, “Payday 2,” [Online]. Available: [https://store.steampowered.com/app/218620/PAYDAY\\_2/](https://store.steampowered.com/app/218620/PAYDAY_2/), 2013, accessed on: Feb 12, 2018.

- 
- [38] ———, “Left 4 dead 2,” [Online]. Available: [https://store.steampowered.com/app/550/Left\\_4\\_Dead\\_2/](https://store.steampowered.com/app/550/Left_4_Dead_2/), 2009, accessed on: Feb 12, 2018.
- [39] Mojang, “Minecraft,” [Online]. Available: <https://minecraft.net/>, 2011, accessed on: Mar 18, 2018.
- [40] Fun Games for Free, “Block craft 3d,” [Online]. Available: <https://play.google.com/store/apps/details?id=com.fungames.blockcraft/>, 2015, accessed on: Mar 18, 2018.
- [41] AppSpy, “Bloons td 4 review,” [Online]. Available: <http://www.appspy.com/review/4080/bloons-td-4>, 2011, accessed on: Feb 11, 2018.
- [42] Pan European Game Information, “What do the labels mean?” [Online]. Available: <https://pegi.info/page/what-do-labels-mean>, 2017, accessed on: Feb 20, 2018.
- [43] T. Fullerton and C. Swain, “Chapter 6 - conceptualization,” in *Game Design Workshop*, 2nd ed., ser. Gama Network Series, T. Fullerton and C. Swain, Eds. Boston: Morgan Kaufmann, 2008, pp. 148 – 174. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/B9780240809748500130>
- [44] Imagine Games Network, “Defend your castle review,” [Online]. Available: <http://www.ign.com/articles/2008/05/12/defend-your-castle-review>, 2008, accessed on: Feb 11, 2018.
- [45] Unity Technologies, “Unity asset store,” [Online]. Available: <https://assetstore.unity.com/>, 2018, accessed on: Feb 20, 2018.
- [46] T. W. Kim, E. H. Kim, J. K. Kim, and T. Y. Kim, “A leader election algorithm in a distributed computing system,” in *Proceedings of the Fifth IEEE Computer Society Workshop on Future Trends of Distributed Computing Systems*, Aug 1995, pp. 481–485, [Online]. Available: <https://ieeexplore.ieee.org/document/525021/>, accessed on: May 12, 2018.
- [47] P. Suddaby, “The many ways to show the player how it’s done with in-game tutorials,” [Online]. Available: <https://gamedevelopment.tutsplus.com/tutorials/the-many-ways-to-show-the-player-how-its-done-with-in-game-tutorials--gamedev-400>, 2012, accessed on: April 25, 2018.
- [48] A. Shannon, A. Boyce, C. Gadwal, and T. Barnes, “Effective practices in game tutorial systems,” in *Proceedings of the 8th International Conference on the Foundations of Digital Games*, ser. FDG ’13. California, USA: SASDG, 2013, pp. 338–345. [Online]. Available: [http://www.fdg2013.org/program/papers/paper44\\_shannon\\_etal.pdf](http://www.fdg2013.org/program/papers/paper44_shannon_etal.pdf)
- [49] H. Montrieux, T. Schellens, J. Van Landeghem, and T. Mouton, “Introducing tablet devices during mathematics education: ’is it really a magical learning tool?’,” *PONTE*, vol. 72, no. 7, pp. 393–410, 2016. [Online]. Available: <http://dx.doi.org/10.21506/j.ponte.2016.7.30>

# A

## Letter of consent

Missivbrev - Kandidatarbete "Four in One" Chalmers Tekniska Högskola

Vi är sex studenter från Chalmers tekniska högskola som nu läser tredje året och därmed utför vårt kandidatarbete, till vilket vi valt att arbeta med spelutveckling för barn i en utbildningsmiljö.

Syftet med detta arbete är att utveckla ett mobilspel som uppmuntrar barn till att engagera sig socialt med varandra och utvecklar deras samarbetsförmåga, samtidigt som dom har roligt tillsammans. Vår målgrupp är barn i åldrarna 7-9, och för att bättre förstå oss på hur dessa tänker och vad de anser roligt önskar vi att ditt barn får delta i en intervju som berör dessa områden.

Frågorna under intervjun kommer att kretsa kring nuvarande spelvanor, teknikvana, samt sociala vanor. Frågor kommer att ställas både till individuella barn såväl som öppet till gruppen och ett mindre moment idegenerering kommer utföras i form av en övning där barnen själva får hitta på ett spel.

Dokumentering av intervjun sker skriftligt och varken ljud eller videoinspelning kommer utföras. Resultaten kommer att användas som underlag för idegenerering av vårt spelkoncept såväl som analysunderlag vid senare koncept-iterering. En transkribering av intervjun kan komma att finnas redovisad som bilaga tillsammans med den slutrapport vårt arbete producerar, vilken kommer publiceras under Chalmers tekniska högskola. Intervju-resultaten kommer enbart att användas akademiskt.

För frågor och funderingar är du välkommen att kontakta mig (Viktor) på [dimander@student.chalmers.se](mailto:dimander@student.chalmers.se)

Hälsningar kandidatarbetsgruppen "Four in one"

Viktor Djukic Dimander

Adam Andreasson

Viktor Svensson

Marcus Randevik

Patrik Olson

Daniel Illipe



# B

## Prestudy Interviews

### Interview script

Do you have a mobile phone?

Do you play mobile games?

What games?

Which one is your favourite?

Why? What do you like about it?

Are there any mobile games that you really don't like?

Do you ever play video games (any kind) with your friends?

What do you usually do during schoolbreak?

### Answers:

#### Pair 1:

Do you have a mobile phone?

1:yes      2:yes

Do you play mobile games?

1:yes      2:yes, sometimes

What games?

1:Minecraft, Blockcraft 3d      2:Sandbox, Unicorn

Which one is your favourite?

1:Minecraft      2:Blockcraft 3d

Why? What do you like about it?

free, can build things

Are there any mobile games that you really don't like?

1:Cookie clicker      2:Fidget spinner

Do you ever play video games (any kind) with your friends?

1:Play minecraft at LANs      2:understands terms like "network" and "wifi"

What do you usually do during schoolbreak?

1:Pokemon      2:Football, "Pretend to be the penguin"

Other notes: Every member of the class have their own ipad Have to leave their phone with the teacher when they arrive at school

### **Pair 2:**

Do you have a mobile phone?

1:No      2:No

Do you play mobile games?

1:yes      2:yes

What games?

1:A game containing many games, don't remember name

2:Blockcraft, Flatheros, Robblox, Knifehit

Which one is your favourite?

1:Blockcraft      2:Blockcraft

Why? What do you like about it?

1:building      2:building big castles

Are there any mobile games that you really don't like?

1:mathfight      2:pizzaplace

Do you ever play video games (any kind) with your friends? 1:No      2:No

What do you usually do during schoolbreak?

1:Fotball      2:Climb, slides

Other notes: "Almost all games are violence" (negative tone)

### **Pair 3:**

Do you have a mobile phone?

1:no      2:yes

Do you play mobile games?

1:no      2:yes

What games?

2:Assassin, Pixel guns, Batman, Zombie, Clash of Clans, Minecraft

Which one is your favourite?

2:don't know, likes games with dragons

Why? What do you like about it?

2:Don't know

Are there any mobile games that you really don't like?

2: Lego, childish, boring. Games that are too hard

Do you ever play video games (any kind) with your friends?

2: Roblox, Minecraft

What do you usually do during schoolbreak? -

Other notes: Child nr 2 very positive of violence in games

#### **Pair 4:**

Do you have a mobile phone?

1: yes      2: yes

Do you play mobile games?

1: yes      2: yes

What games?

1: Sandbox, piano tiles 2, pokemon go      2: Jocke och Jonnas games

Which one is your favourite?

1: Pokemon go      2: jocke och jonna räddar julen

Why? What do you like about it?

1: you can run around      2: Jumping on platforms

Are there any mobile games that you really don't like?

1: Alfons Åberg      2: Radioapan

Do you ever play video games (any kind) with your friends?

1: Pokemon go      2: no

What do you usually do during schoolbreak?

1: Run around      2: Jumping rope

Other notes: One plays likes board games, mentions yenga, Trivial pursuit, memory and chess

#### **Pair 5:**

Do you have a mobile phone?

1: yes      2: no

Do you play mobile games?

1: yes      2: yes

What games?

1: Minecraft, Sandbox, Slingkong      2: Painting games and puzzle games

Which one is your favourite?

1: One that looks like Minecraft, cant remember name      2: MovieStarPlanet

Why? What do you like about it?

1:Obstacle courses      2:Clothes and pets

Are there any mobile games that you really don't like?

1:Bloons      2:Cargames

Do you ever play video games (any kind) with your friends?

1:-      2:-

What do you usually do during schoolbreak?

1:-      2:-

Other notes: Nr.1 doesn't like bloons tower defence due to the repetitiveness, "you complete a wave but what do you get out of it?"

### **Teacher:**

**How do you work with cooperation today?** Often in pairs. Trouble with shy children. Children take different roles within a group. Arranges groups herself, needs to keep certain children away from each other.

**How do the children handle tasks together?** Some cannot work together at all. The children needs practice in cooperating. Difficult to get shy kids to participate.

**What is the general opinion on violent games?** Boys find violent games interesting.

# C

## User testing transcript

### Test 1

Spelarna får ipadsen med startrutan utplacerade på ett bord, klickar ganska snabbt på hitta spelare. Alla enheter hittar varandra och dem lägger ihop pusslet. Förstod dock ej till en början att man sedan behövde klicka på alla skärmar för att spelet skulle starta.

I like slime”

“dela med er”

“DU SKJUTER IN PÅ MIN!!”

“du skjuter på mig”

Spelarna förstår väldigt lite när de kommer till köprutan och klickar nästan omedelbart bort den. Separerar nu ipadsen för att enklare kunna spela själv.

“har inte du kommit till nästa nivå?”

Svårt att förstå samarbetsperspektivet och blir något uträkade när det inte finns nåt kvar på egen skärm.

“skjut till mig, hjälp mig!” "förstör inte min fyrkant"

Visar återigen inget intresse av att läsa texterna för byggnader utan klickar bara vidare för att snabbt kunna börja spela.

“hjälp mig!!!”

Börjar hjälpa varandra genom att på måfå skjuta över till andra enheter.

“man måste skynda sig”

Lägger tillbaks ipads i formation igen för att enklare kunna skjuta över till andra enheter. Tror dock att spelet handlar om vem som är snabbast.

“ser ut som små ärtor. ärtspelet!”

Förstår inte vad byggnaderna är eller gör men bryr sig om dom på så sätt att de är villiga att försvara bygganderna.

“jag skickar till dig” “slimeklumpar mot alien”

## Test 2

Spelarna förstår likt tidigare test att alla behöver klicka på hitta spelare men denna gången lyckas bara 3 av 4 enheter att hitta varandra. Vi får hjälpa till att starta om spelet varvid det sedan fungerar utan problem.

“jag kan skjuta genom davids ipad”  
“jag skjuter dina slimes david!!”  
“hörrni man kan ha dom lite för sig själv också”  
“det här va ju ändå ganska kul, på ett sätt”

Förstår inte köprutan och klickar bort denna för att snabbt få nya fiender.

“jag vill inte ha nån hjälp” - sitter med sin egen ipad  
“kan man inte vinna?”  
“JAG SKA HA DEN”  
“är det nån som vill köpa?”  
“nu har alla en sak”  
“jag fick ingen ” “haha!”

Tänker ej på var dem placerar byggander och tävlar om att ha flest på sin egen skärm. Tror att man måste hålla på kanonen för att sikta, istället för att enbart klicka på fienden.

“vänta! först ska vi bestämma!”  
“vi sparar tills alla kan köpa varsin”

Turas nu om att bygga för att balansera så att fienden möter fällor på alla skärmar. Vissa av spelarna vill hellre ha sin ipad nära sig och inte nödvändigtvis bredvid dem andra, detta för att enklare kunna sikta.

“Nej jag vill inte samarbeta!”, (slänger sig sen in i samarbetet två rundor senare)  
ansvarar för sina egna saker  
“ursäkta maja men jag måste avbryta dig, för DIN SVÄRM”  
“Maja, ta fällan nu. David, din tur”  
“jag vill också spela själv!” , (går ur spelet)  
“jag vill spela med eeer” (ångrar sig, tröttnar på att spela själv)  
“det här är mitt favoritspel!!”

Diskuterar vid det här laget strategier tillsammans mellan rundorna och försvarar sig mot större svärmer genom att rikta alla kanoner mot en ipad. tacklar “svärmer” tillsammans

### Test 3

trycker direkt på forsätt spela när dom ser knappen

“vadå köp nya byggnader??”

läser inte vad byggnaderna gör lär sig med tiden diskuterar var det är smartast att placera byggnader