NTNU – Trondheim
Norwegian University of
Science and Technology

TDT4900 - Computer Science, Master's Thesis

EXPLORING COLLABORATION IN VIDEO GAMES FOR CHILDREN

Game Prototype and Experiment

David Hovind

supervised by
Alf Inge Wang

June 2016

Abstract

Children today, are exposed to video games from an early age in the form of tablets, smart phones and computers. Social interaction is a big part of why we play video games and is a way for us to socialize with friends and strangers. This thesis seeks to explore these areas by creating a video game, focusing on social interaction to stimulate collaboration children.

The main research goals of this thesis was to create a game prototype that focuses on social interaction and collaboration, conduct an experiment by testing the game with children, and to study the technologies and process involved in video game development. Reaching these goals was achieved through a literature study on video game technologies and concepts, the design and creation of a video game prototype, and by analysing the results from a play test that was performed at Buvik School.

From these methods, the central findings were that having different roles for the players in the game created a dependency between the them, which enhanced the collaboration. Puzzle elements, and the unique gameplay elements in the game prototype were a great way to encourage special interaction, as the children needed to communicate in order to succeed in the game. The play testing was a success and all the children enjoyed playing a collaborative game with focus on social interaction. Additionally, the children saw the importance of working together in order to succeed. Furthermore, the process of video game development is greatly aided by tools like game engines that handle common video game related tasks.

Research presented in this thesis is useful for aspiring and qualified video game developers, as it explores video game technologies and concepts, and the process of video game development. The game prototype created, its design, and the results of the experiment serves as empirical data and a source of inspiration for research and video game development. The results show that using collaboration in video games has potential for further research, and could possibly be used to help teach children valuable skills with regards to social interaction and collaboration.

Sammendrag

Barn blir i dag introdusert til spill i en veldig ung alder, i form av smart-telefoner, tablets, og datamaskiner. Sosial interaksjon spiller en stor rolle i hvorfor vi spiller spill, og er en måte vi bruker for å sosialisere med venner og fremmede. Denne masteroppgaven vil utforske disse områdene ved å utvikle et spil som fokuserer på sosial interaksjon for å oppfordre til samarbeid blant barn.

Hoved-forskningsmålene i denne masteroppgaven gikk ut på å lage en spillprototype som fokuserer på sosial interaksjon og samarbeid, å utføre et eksperiment hvor spillet ble testet med barn, og å utforske hvilke teknologier som er tilgjengelig for, og prosessen bak, spillutvikling. For å nå disse målene ble det gjennomført en litteraturstudie for å se på teknologier og aspekter ved spillutvikling, en spillprototype ble designet og utviklet, og en analyse av observasjoner og resultater fra testingen av spillprototypen ble gjennomført.

Disse metodene ble brukt for å komme frem til hovedfunnene til denne oppgaven. Det å la spillerne ha forskjellige roller, skapte et bånd hvor de ble avhengige av hverandre, noe som drev samarbeidet i spillet. Puslespill elementer, og de unike spillelementene i spillprototypen, var med på oppfordre til sosial interaksjon blant barna, hvor de måtte kommunisere for å komme seg videre i spillet. Spilltestingen var en suksess, alle barna syntes det var gøy å spille et spill hvor samarbeid var i fokus, og de forstod viktigheten med å samarbeide i spillet. Spillutvikling er en komplisert prosess, men det finnes gode verktøy som spillmotorer for å hjelpe med ordinære spillrelaterte oppgaver.

Denne masteroppgaven er nyttig for både ambisiøse og erfarne spillutviklere, siden den utforsker spillteknologier og konsepter, og gir innblikk i prosessen for spillutvikling. Spillprototypen, dens design, og resultatene fra eksperimentet tilbyr en kilde til inspirasjon og empirisk data for forskning og spillutvikling. Resultatene viser at det å bruke samarbeid som fokus i spill har potensiale for videre forskning og kan potensielt brukes som et hjelpemiddel for å lære barn viktige ferdigheter som samarbeid og sosial interaksjon.

Preface

This report presents the work done for my master's thesis at the Department of Computer and Information Technology at the Norwegian University of Science and Technology. The work was done during the spring of 2016 and concludes my master's degree in Computer Science.

I would like to thank Bente Sandø and the faculty and pupils of Buvik Skole SFO (Buvik School after school program) for letting us preform a user test with the children there. The children were all well behaved and provided great feedback, directly and indirectly, which helped with essential data for the evaluation of the game prototype.

I would also like to thank Kjell R. Elstad, fellow student and friend with whom I wrote the foundation that this project is based on.

Last but not least, I would like to thank my supervisor, Alf Inge Wang, for providing great feedback, interesting ideas and views, and for helping me with the execution of the user testing. Your guidance and expertise was an invaluable resource for the work done in this project.

Trondheim, the 17th of June, 2016

David Johan Hovind

Contents

List of Figures

List of Tables

Acronyms

AI Artificial Intelligence. 56, 57

API Application Programming Interface. 23, 93

AR Augmented Reality. 30, 33, 35

D-pad Directional pad. 20

GQM Goal Question Metric. 7–9

IDE Integrated Development Environment. 41, 46, 91

IR infrared. 23, 26

N64 Nintendo 64. 20, 21, 33

NES Nintendo Entertainment System. 20

PS1 Sony PlayStation 1. 20

PS3 Sony PlayStation 3. 17, 18

PS4 Sony PlayStation 4. 16, 17

RTS Real Time Strategy. 21, 22

SNES Super Nintendo Entertainment System. 20

VR Virtual Reality. 30, 33, 35, 39, 66

Wii Nintendo Wii. 14, 15, 18, 22

WiiU Nintendo WiiU. 15

Xbox 360 Microsoft Xbox 360. 17, 18, 23

Xbox One Microsoft Xbox One. 15–17

PART I

Introduction and Methodology

This part introduces the background and context for the thesis, as well as describing the methodology that went into answering the research questions presented.

Chapter 1

Introduction

The video game industry is one of the biggest entertainment industries in the world, and it has become a big part of people's lives, especially children, who are exposed to technologies like smart phones and tablets from an early age.

Video games come in all shapes and sizes, from being played on computers, video game consoles, smart phones, and even virtual reality headsets. There are also different types of video game developers ranging from big companies creating AAA-titles to small independent game companies and even individuals creating independent video games.

Today's technology makes it so that individuals and small teams can create video games that can reach millions of gamers online, without investing much more than just time. Big video game companies usually stick to well proven video game concepts and interaction devices to reach as many people as possible and to generate the most revenue. Independent developers can on the other hand generally be more creative and innovative when creating video games, and create games that use unconventional interaction devices or game concepts.

Proprietary input devices like motion controls for video game consoles mostly use standard communication protocols like USB or Bluetooth, which can be used to interface with a computer with the help of official and unofficial driver software.

Social interaction is a big part of video games, with most games including a multiplayer mode, either local or online. Gaming can be a way to socialize with strangers to create new friendships, or with friends to strengthen (or ruin) friendships. Moreover, social interaction in games can also be used to stimulate and teach children collaboration, and this is an area this project will explore. Potentially, social games can be used in schools to train children to improve their collaboration skills, which later can boost their skills working in groups in projects

These are exciting areas for research that this thesis will delve into.

1.1 Context

This report is an extension of work done in a specialization project where I and Kjell R. Elstad explored different game technologies and concepts that goes into creating innovative video games. I was interested and continuing this work and to create a game prototype of one of the game concepts that were created in the specialization project [1].

The finished game prototype, and the full source code, as well as binaries for running the game on Windows, is available on GitHub at:
`https://github.com/Daveiac/Master`

1.2 Motivation

Video games have always been a big part in my life. I grew up playing video games and I still do on a daily basis. Having played a lot of video games, I am excited to specialize in a project where I can create my own video game and explore different game technologies and concepts.

With the evolution of game technology video games have become accessible for anyone, and children are more exposed to video games from early childhood. I want to create a video game for children with the focus on collaboration, and encourages social interaction between the players. I want to expose children to a new kind of gameplay, where the players are dependent on each other in order to succeed.

1.3 Problem Description

The following problem description is based on the an earlier version given for the master's thesis:

> In this project, the goal is to prototype an innovative game and test this game on users. The game prototype shall be a collaborative game made for children, where the emphasis is on encouraging social interaction and stimulating collaboration. Innovation can be made in the type of gameplay the game provides, how it combines various game genres, what technology is used to control or play the game, how the social interaction between players is supported, and the purpose of the game.
>
> The first phase of the project will include a study of game technologies, game concepts and interaction devices, the second phase will be

to implement one or more prototype, and the third phase will be to evaluate the concept through play-testing on real users.

This master's thesis will include a literature study, the creation of a game prototype, and the play testing of this prototype on real users.

1.4 Structure

The structure of this thesis is divided into five parts which roughly correspond to the order of phases that this project went through.

Part I: Introduction and Methodology The introduction provides the context and motivation behind the project, the problem statement and the structure of this report.

Part II: Literature Study The literature study presents a background of the evolution of game technology, various interaction devices, game immersion, and game development technologies.

Part III: Design and Implementation The design and implementation part describes the design, architecture and implementation of the game prototype.

Part IV: Experimentation The experimentation part provides details of the planning, execution, and results of the play testing that was performed.

Part V: Evaluation and Conclusion The final part concludes the thesis with an evaluation of the research, a conclusion, and a list of areas for further work.

Chapter 2

Methodology

The research method used in this project is based on the Goal Question Metric (GQM) approach [2], where firstly a research goal is defined (conceptual level), secondly the goal is decomposed into research questions (operational level), and lastly the research questions are described by a set of metrics that answers the research questions (quantitative level). Through the GQM approach three main research goals will be specified which represents the areas that will be investigated and drive the research through this thesis.

2.1 Research Goals

The following research goals were developed using the template for research goals of the GQM framework. This thesis focuses on three different aspects of creating a collaborative video game for children. The focus on the first goal is the creation of the game prototype. The motivation behind the second goal is to observe and capture children's experience with the game prototype. The third and final goal seeks to discover what goes into game development and what tools are available for an independent developer.

RG1: The purpose of this project is to create a collaborative game prototype which promotes collaboration and social interaction from the point of view of a game designer in the context of video game design.

RG2: The purpose of this project is to see how children experience a collaborative game where the focus is on collaboration and social interaction from point of view of the children in the context of playing video games.

RG3: The purpose of this project is to investigate what tools are available to create video games from the point of view of an independent developer in the context of video game design.

2.2 Research Questions

The research goals are broken down into research question according to the GQM framework.

2.2.1 Research Goal 1: Collaborative Game

These research questions focuses on the creation of a video game that stimulates collaboration and encourages social interaction between players. when creating the game prototype it is important to find out how the game can be designed, and what different game aspects can encourage social interaction and collaboration.

RQ1.1: What game mechanics are suitable for stimulating collaboration between players?

With this question, the intention is to discover various game mechanics that can will engage both players and encourage collaboration.

RQ1.2: How can the game be designed to encourage social interaction between players?

The focus of this question is to find out how the game can use game mechanics and controls to encourage social interaction.

2.2.2 Research Goal 2: Children's Experience

The intention of these research questions is to study find out how children will experience a video game where social interaction and collaboration is encouraged. A play test of the game prototype will be performed and these research questions will help guide the methods for observation and analysis of the testing.

RQ2.1: Is a collaborative game that encourages social interaction engaging for children?

This question is meant to gauge children's interest for a collaborative game that encourages social interaction.

RQ2.2: How does the children's gender or game experience affect their experience of collaborative game with focus on collaboration and social interaction?

The intention behind this question is to determine if the gender or game experience of the children have an effect on their experience the game prototype.

2.2.3 Research Goal 3: Game Development

These last research questions focuses on the development of a video game and what tools are available for independent video game developers. The research and process that goes into creating the game prototype will be documented to get a better view of the process and technology available for an independent developer.

RQ3.1 What tools are available for developing video games as an independent developer.

The objective of this question is to discover what tools are available for developing video games as an independent developer.

RQ3.2 Can proprietary controllers made for game consoles be used to develop games for the PC?

The purpose of this question is to determine if proprietary game controllers made for game consoles can be used when developing games for the PC.

2.3 Process

This section explains the different methods used in this project to be able to answer the research questions. The GQM model advises the use of quantitative data, but most of the questions will be answered with qualitative data gathered from the research, development and experience. This is due to the openness of the research questions. Table 2.3 gives an overview of the metrics used in answering the research questions.

Table 2.1: overview of the metrics for the research questions

Question	**Metrics**
RQ1.1	Development of game prototype. Observations and results from play test.
RQ1.2	Development of game prototype. Observations and results from play test.
RQ2.1	Observations and analysis from the play test. Results and analysis of the questionnaire.
RQ2.2	Observations and analysis from the play test. Results and analysis of the questionnaire.
RQ3.1	Literature study.
RQ3.2	Experience with development of the game prototype.

2.3.1 Literature Study

A literature study will be conducted which will explore different aspects of video game development. This includes first, an evolution of game technology to provide context and background for the technologies and concepts used in the video game industry. Second, interaction devices to give an insight into how player can interact with video games and to find interesting interaction devices to be used with the game prototype. Third, immersion in order to be able to create an immersive game and also to help with the questionnaire. Last, game development technologies in order to understand what options are available and to answer research question RQ3.1, but will also be used to gain information to help build the game prototype and how to evaluate it. The literature study can be found in Part II.

2.3.2 Game Prototype

A game prototype will be created to be able to test how children will respond to a collaborative game. This game prototype will feature two players with different roles that need to work together to reach the goal. The focus of the game prototype will be to encourage social interaction between the players. The prototype will be developed using tools, interaction devices, and concepts based on the literature study. The focus of the development will be to create a prototype that is easily modifiable to add new features, and the gameplay must be simple enough for children to understand and play. The development of the game prototype will be the main way of answering the research questions RQ1.1, RQ1.2, and RQ3.2. Details about the implementation and design of the game prototype can be found in Part III.

2.3.3 Play Testing, Data Collection and Analysis

A play testing session with a group of children will be conducted, accompanied by a questionnaire that will help with the evaluation of the game prototype. A questionnaire was chosen to ask all the participants the same questions in order to get answers in a quantitative form that can easily be analysed. The observations from the play testing and results from the questionnaire will primarily be used in the answering of research questions RQ2.1 and RQ2.2, but will also help to answer RQ1.1 and 1.2. More details about the play testing in Part IV, and a more detailed description of the questionnaire can be found in Section 10.2.

PART II

Literature Study

This part describes an evolution of game technology, different interaction devices, immersion in video games, and an overview of video game development technologies.

Chapter 3

Evolution of Game Technology

This chapter will explore the background of different video game technologies and how the industry has evolved. Games are being played by different kinds of people all over the world, and games can now be developed by a single person and reach these people through the internet. This chapter serves to give some information on how this came to be. The majority of the research presented here was conducted for the specialization project [1], and is based on work done by Overmars [3].

3.1 Casual Gaming

Video games had been primarily for the dedicated gamer, but during the 2000s there was a rise in casual games. With more households having faster internet connections and more people using computers, video games became more accessible to people that were not traditionally gamers. Many of the games were played in the browser and was financed by ad revenue alone. A key example is the highly successful game Bejeweled (2001). Another trend was games on Facebook, like Farmville (2009) which had over 80 million users in 2012.

With the introduction of the iPhone (2007), the mobile gaming market started to blossom. The most important factor was the App Store which allowed every developer to create games and sell them through Apples store. This made it possible for individuals and small teams of developers to develop games that could reach millions of users. The amount of content grew and games were sold for USD 1; some games were even free and supported by ad revenue. Apple got some competition in 2008 when the android mobile operating system hit the market. With phones created by Samsung HTC and Sony, android quickly became popular and even had its own application market similar to the App Store, called Android Market [4]. The most popular game for the smart phones is Angry birds (pictured in Figure 3.1), which had been downloaded over three billion times as of July 2015

[5]. One of the reasons for its big success is its freemium model, where the game is initially free, but supports in-app purchases and displays advertisements.

Figure 3.1: Angry Birds, the most downloaded game for smart phones

3.2 Motion Controls

Nintendo was the company that pioneered motion controls for the consoles. They chose to create a less powerful console, the Nintendo Wii (Wii) in 2006, and catered to a more casual audience with cartoonish graphics, and games geared toward children. They also created a new controller, the Wiimote, that looked like a remote that could track the player's arm and hand movements (more about this controller in Section 4.2.1). The most popular game was Wii Sports, shown in Figure 3.2 that featured a collection of sports games that used the motion controller to play. Nintendo was more successful than the other consoles and had sold over 100 million consoles worldwide as of June 2014 [6].

Microsoft and Sony saw the success of the Wii and created their own motion control systems. Microsoft created a camera system that tracked the full body of the player, called the Kinect in 2010. Sony made a device similar to the Wiimote in 2010, called the Playstaton Move. The Move is a handheld remote like controller

with a coloured light at the end and a camera to track the controller in 3D space. More information about the motion control devices in Chapter 4.

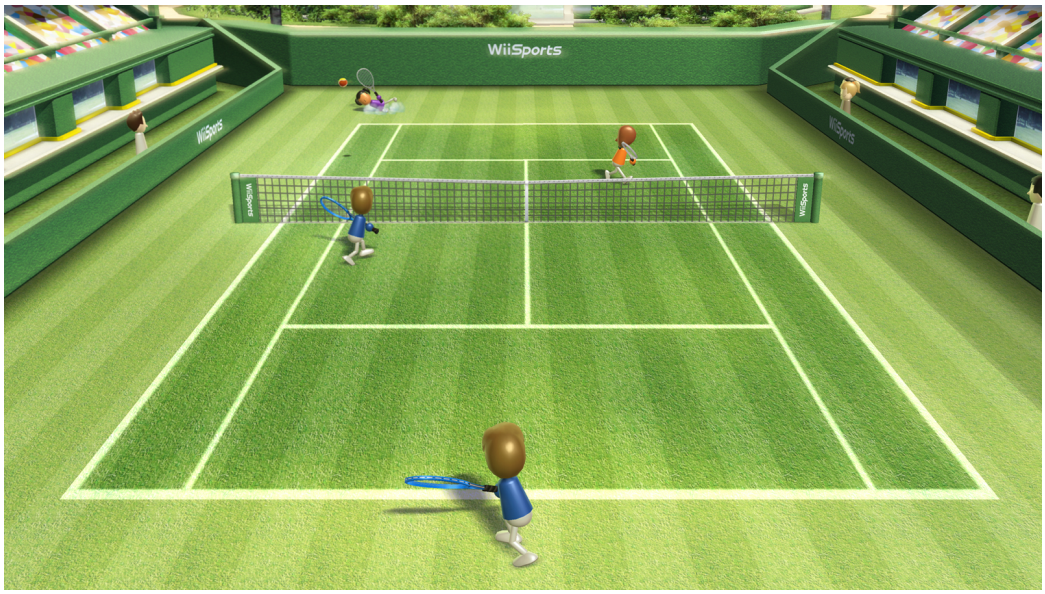

Figure 3.2: Wii Sports, a popular motion controlled game

3.3 Current Generation of Consoles

Nintendo started what is considered the current generation of consoles with the Nintendo WiiU (WiiU) as a successor to the Wii in 2012. The WiiU features a tablet like gamepad with a 6 inch touch screen, motion sensors, and traditional buttons and joysticks [7]. Unlike the Wii, the WiiU does have HD graphics, but it is still the least powerful of the current generation of consoles [8]. Nintendo's primary target audience is still children, and Nintendo has created a line of physical game accessories called amiibo. Amiibo, illustrated in Figure 3.3, are toys like figures and cards that can interact with the games and provide special functionality like an extra character, special items, and new adventures [9].

Microsoft and Sony released their consoles at the same time for the 2013 holiday season. Microsoft's Microsoft Xbox One (Xbox One) focused on creating an all-in-one home entertainment experience, hence the name. The Xbox One can rent films, or play live television using an interactive TV guide. The Xbox One also features Kinect 2.0, a more powerful version of the motion control system for Xbox. Two of the top games for the Xbox One at launch was Call of Duty Ghosts (pictured in Figure 3.4) and Forza Motorsport 5 [10].

Figure 3.3: An amiibo figure interacting with the Wii U gamepad

Sony focused on the games themselves and created a powerful console. The Sony PlayStation 4 (PS4) has features like game streaming, which makes it possible to stream games from a PS4 to a phone or tablet. The PS4 also has support for their motion control system Move. Sony also focused on social interaction by being able to share screen shots and game clips with a share button on the controller [11]. Sony continued one of their top game series with Killzone Shadow Fall at launch [12].

The Xbox One and the PS4 have a lot of similar features. They both have the support for game streaming to Twitch and Ustream, and can record game clips as you play. Both consoles also support third party applications such as Netflix and YouTube. The consoles are becoming more about entertainment than just about the games. Gamers want to have access to the internet from their couch without having to use the computer. Downloadable games is also a popular feature, where

all the games are available as a digital download instead of buying a game disc [8]. Even though the Xbox One and PS4 support the motion control systems from the previous generation, they never really reached the success of the WiiMote. In recent years these motion control system have started to fade in popularity, but some niche areas still remain, such as fitness, dance, and party games [13–15].

Figure 3.4: Call of Duty Ghosts, a popular launch title for the Xbox One and PlayStation 4

3.4 Digital Distribution

Downloading the games from the internet via the built in store of the console was something that was introduced with the Microsoft Xbox 360 (Xbox 360) and Sony PlayStation 3 (PS3), however not all titles were available for digital download. With PS4 and Xbox One though, all disc titles are also available for digital download. This makes buying games less of a hassle, because you don't have to go to a physical store and you can buy it whenever you want. It is also possible to pre-order and pre download titles so that they are ready to play on the release date [16, 17].

Buying games digitally was something that existed on the PC, and the majority of PC gamers download and play games today. A popular platform for buying games digitally on the PC is Steam. Steam was created by Valve in 2002 to easily

be able to update their games [18]. It then went on to become the biggest digital distribution platform for games on the PC [19].

Digital distribution of games does not only allow for big game companies to release games; it allows for independent game developers to create and share their games with the world. These types of games are called indie games and have been around since the dawn of the computer. Indie games are often more innovative than games from big companies because they have less constraints and pressure from publishers to create best selling games. Modern indie games that was distributed digitally, became popular at the later half of the 2000s with Cave Story (2004) being one of the first. Cave story was initially released as freeware for the PC and quickly went viral. Xbox 360, PS3, and the Wii started to feature indie games in their online stores. Probably most notably was the Xbox Live Arcade together with the framework XNA created by Microsoft to release games on their Xbox 360. Some very popular games came out of that like Braid, Super Meat Boy, and Castle Crashers [20]. Steam also housed indie games in their store and even released a platform called Greenlight where developers can showcase their games and the community gets to decide which games get accepted into Steam [21].

3.5 Summary

This chapter have looked at how the current state of the video game industry came to be as it is today. Games are being played by a multitude of people and on all kinds of platforms and casual gaming became popular. Motion controls have created a whole new way to interact with video games and the three major video game console companies have their own form of motion controllers. Video game consoles have become more of a home entertainment device with the ability to watch movies and television, and have also become more social with more options for connecting and sharing content with friends. In recent years there has been a shift towards digital distribution which has enabled independent game developers to create games for the masses.

It is interesting to see the shift towards more casual gaming and the fact that independent developers can create games for the newest game consoles and the PC. This is helpful for the project because the technology has evolved in such a way to make it possible for a single developer to create a video game to be played on the PC or the newest game consoles. In this project the most relevant findings from this chapter include how I can utilize casual game concepts which will both be familiar and entertaining to the players.

Chapter 4

Interaction Devices

An important part of playing a game is being able to interact with it. The motivation for this chapter is to discover various interaction devices that could be used with the game prototype. This chapter will provide insight into what types of interaction devices can be used to interact with video games. The research in this chapter is based on the research conducted for the specialization project [1]

4.1 Conventional Interaction Devices

Conventional interaction devices are interaction devices that have been the most commonly used for video games.

4.1.1 Joystick

With the introduction of game consoles came the need for a more standardized device for interacting with games. The versatile joystick has been an important part of fulfilling this need. A joystick, along with one or two buttons, was normal for the early consoles. Atari 2600 is one such example, shown in Figure 4.1. Later it has become a small, but important part of most controllers as a thumbstick.

4.1.2 Gamepad

As games became more complex, requirements for the controller also increased. More buttons and new elements were added and the joystick eventually reappeared as one or more thumbsticks on the gamepad. This section describes how the gamepad has changed throughout the years and how it has changed the way games are played. A paper written by Cummings [22] on the subject is the primary source for this section.

Figure 4.1: The joystick used for the Atari 2600

Nintendo Entertainment System (NES) was one of the first systems to use the gamepad. It had two buttons and a Directional pad (D-pad). One of the primary advantages of a joystick is in its ability to give analogue feedback. The joysticks used in game consoles did not have this feature at the time, nor did they need analogue input, so the D-pad could provide the same accuracy of input with less movement.

While the D-pad provides sufficient input for 2D games, it is not sufficient for 3D games. With the development of Super Mario 64 for Nintendo's Nintendo 64 (N64) console, they needed a controls scheme that could control the protagonist in a 3D environment. Through collaboration between hardware and software developers, the controller for the N64 was created. The controller, seen in Figure 4.2 (bottom left), had an analogue joystick for controlling Mario in the 3D environment. Separate buttons were used to manually adjust the camera angle. By collaborating they could create a good control scheme that worked smoothly with the game. As this was the first use of an analogue joystick for this purpose, the collaboration was a very useful tool for creating a solid system. A second thumbstick has since become a common part of the gamepad after it was introduced for the Sony PlayStation 1 (PS1). Four different gamepads for Nintendo consoles are shown in Figure 4.2. These show the changes from the NES through the Super Nintendo Entertainment System (SNES) and N64 to the GameCube.

Figure 4.2: The gamepads for four generations of Nintendo consoles. NES (top left), SNES (top right), N64 (bottom left), and GameCube (bottom right)

Haptic feedback, often referred to as force feedback in the context of video games, is another technology added to the gamepad. First in the Rumble Pak for the N64 which was an addition to the controller to make it shake. Force feedback, in the form of shaking the gamepad, has since become common in console controllers. This extra form of feedback can, among other things, tell the player that they have crashed or have been hit, as it did in the N64 game Starfox 64.

Analogue input is not limited to joysticks. A common use of analogue buttons on gamepads are the triggers. An analogue trigger can for instance be used as the accelerator in a racing or driving game.

4.1.3 Keyboard and Mouse

The quantity of keys on the keyboard and later also the precision of the mouse as a pointing device gave new options for how games could be played. New genres and games followed. Among these were Real Time Strategy (RTS) games like Command & Conquer and point-and-click games like Monkey Island.

A good example of how well the keyboard and mouse setup can be for some genres, is that the control scheme for the RTS genre has stayed relatively unchanged since the beginning. An example of a keyboard and mouse can be seen in Figure 4.3

Figure 4.3: Keyboard and Mouse

4.2 Motion Control Devices

Motion controls devices became popular in the mid to late 2000s and are supported by the current generation of video game consoles. This section describes the motion control systems for the three major game consoles.

4.2.1 Nintendo Wiimote

When Nintendo released their Wii console in 2006 [3] their new controller, the Wiimote, did not resemble the ones used by their competitors. The controller was in shape to a television remote, but more technically complex in functionality. Before the release of Wii, controllers and games were getting increasingly complex. By making the controller for their new home console simpler, Nintendo aimed for a different market than their competitors: new and casual gamers. Quite contradictory to the predicted downfall of Nintendo after the release, Wii became the most popular console of the generation. The revolutionary controller, along with a cheap price tag, may have been the reason for this [3].

Figure 4.4: Wiimote

Although the Wiimote can be used as a gamepad by holding it sideways, this is not its primary function. The next five sections describe how it differs from a typical gamepad and how it changes the way games can be played.

For more information about how the technology of the Wiimote and how it can function as a convenient device for 3D interfaces, see [23, 24].

4.2.2 Microsoft Kinect

Microsoft Kinect is an input device for Microsoft's Xbox consoles. It was first introduced for the Xbox 360 and had a camera, an infrared (IR) depth sensor, and a multi-microphone array. Using this technology it can track players, motion, and gestures in 3D space. Microsoft has also provided an Application Programming Interface (API) for PC which makes it possible to use the Kinect for other purposes like research and even combining it with other technologies to create new innovations [25].

One advantage of the Kinect is that it can encourage physical activity for the

Figure 4.5: Kinect

players. A disadvantage of the Kinect is that it requires open space to prevent damage to players and surroundings.

4.2.3 PlayStation Move

The PlayStation Move has similarities to the Wiimote described in Section 4.2.1, but has different features and functions differently. On top of the controller there is a ball which can light up in different colours. A PlayStation Eye USB Camera can track the position of the controllers by tracking these balls. Distance to the camera can be calculated using the apparent size of the ball. The controller also has one analogue trigger and eight buttons, along with accelerometers and gyroscopes for tracking 3D position. Figure 4.6 shows the controller and camera.

4.3 Specialized Controllers

Controllers made specifically for a single game or a specific genre of games is not exclusive to arcades. They have been created for both PC games and home consoles too. Some constructed to supplement or improve an existing game, others designed as a critical part of a game.

4.3.1 Driving games

Racing games is a popular genre and there are many other genres that involves driving. A steering wheel provides intuitive controls to these types of games. The

Figure 4.6: PlayStation Move and PlayStation Eye

addition of pedals further realism to the driving experience. In some cases, even a controller for manual shifting is possible.

These types of controllers may also have additional buttons and other inputs to navigate menus, as seen in Figure 4.7.

4.3.2 Flight Simulators

An interesting example of specialized controllers is the, sometimes very complex, setups made by flight simulator enthusiasts. It is possible to get a yoke and from there keys, switches, and other instrument panels can be added.

4.3.3 Light Guns

Light gun controllers are based on a camera and a source of light. The point of light is used to determine where the light gun is pointed. This is similar to how the WiiMote works, where the source of light is the Sensorbar. It is worth noting

Figure 4.7: Pedals and steering wheel with buttons for navigating menus.

that the Sensorbar has multiple IR LEDs used to provide orientation in addition to position. For the older game Duck Hunt [26], seen in Figure 4.8, the light source is the television screen. The entire screen turns black, except for a small dot located where the player should be aiming, for a short time.

4.3.4 Rhythm Games

A very common category of controllers created as a critical part of a game are the ones used in rhythm games. Instruments that are simplified versions of the real thing or in some cases almost lifelike. Dance mats and motion tracking. Even microphones can be used as a controller for rhythm games.

Common examples are games that simulate the experience of playing in a band. To play these, special controllers made to look like instruments are required. Figure 4.9 shows an example of such instruments used for games in the Rock Band

Figure 4.8: Screenshot from Duck Hunt

franchise. Dancing is another common example, where the position of the player's feet on the ground typically is the input for the game. More recent games has also used cameras or motion detection controllers.

4.4 Summary

There is a multitude of ways of interacting with a video game, this chapter has explored the most common devices such as keyboards and gamepads, and also more specialized game controllers like light guns and plastic instruments. Using different interaction devices provides different experiences for the player and can be a big factor in the enjoyment and immersion in a video game.

This information will prove useful when deciding on what interaction devices to use for the game prototype to be developed. Most relevant for this project would

Figure 4.9: Instruments from the Rock Band franchise

be to use a gamepad, or a motion control device, like the PlayStation Move, to create a different game dynamic which could be engaging for children to use.

Chapter 5

Immersion

This chapter takes a look at different aspects of game immersion and is based on research done in the specialization project [1]. There are many different parts required to make the player immersed in a game. At first glance, immersion usually involves a mix of visuals and audio and for console games, the use of haptic feedback is common. Other measures to keep the player immersed become apparent when looking deeper. One of these is through the setting and storyline of a game, and there are different ways for a storyline to unfold. If a game does not keep up with the standard set by the other game aspects, the suspense of disbelief may be broken [3]. Social interaction is a big part of why we play video game, people use video games as a way to spend time with friends.

Immersion is important when creating a game, and the information gathered here will be important when designing and developing the game prototype. The social interaction section will be especially useful when creating the game prototype, because it is one of the core concepts for this project. Concepts from the Gameflow framework will also be important when creating questions for the questionnaire, because it is a useful framework for evaluating immersion.

5.1 GameFlow and Game Enjoyment

A player who does not enjoy a game will not continue to play it. This is where GameFlow becomes important, as a player who is experiencing gameflow is enjoying the game. The term GameFlow was introduced by Sweetser [27] to create a model for player enjoyment in video games. It is based on the term flow from Csikszentmihalyi, which is described as “so gratifying that people are willing to do it for its own sake, with little concern for what they will get out of it, even when it is difficult or dangerous” [28].

The GameFlow model consists of eight core elements - concentration, challenge, skills, control, clear goals, feedback, immersion, and social interaction.

Each of the elements have criteria that relate to Csikszentmihalyi's elements of flow. The GameFlow model was found as a useful tool for reviewing games and identifying issues that could affect game enjoyment. The core elements of the GameFlow model are described by Sweetser [27] as follows:

Concentration: Games should require concentration and the player should be able to concentrate on the game without being distracted from tasks that they want or need to concentrate on.

Challenge: Games should be sufficiently challenging and match the players skill level, and the level of challenge should increase as the player progresses through the game.

Player Skills: Games must support skill development and mastery, and should allow for players to develop their skills through playing the game.

Control: Players should feel a sense of control of their actions and movement in the game world, and players should be able to recover from errors.

Clear Goals: Games should have clear goals presented at appropriate times, where overriding goals of the game should be presented early.

Feedback: Players must receive feedback from the game in form of progress towards their goals, feedback on immediate actions, and players should always know their status or score.

Immersion: Players should experience deep, but effortless involvement in the game through becoming less aware of their surroundings and less worried about everyday life.

Social Interaction: Games should support social interaction between players and encourage competition and cooperation between players.

Malone [29] categorizes the essential characteristics that makes games fun into three categories: challenge, fantasy, and curiosity. On the subject of challenge, it is important for a game to have a goal and the outcome of whether or not the goal is achieved should be uncertain. Creation of good goals for a game can come from different approaches. Combining the challenge with fantasy will make the game more interesting. Conveying the fantasy to the player can be achieved by the use of graphics (including Augmented Reality (AR) or Virtual Reality (VR)), audio, haptic feedback, and through the setting and storyline of the game. Creating the fantasy can be done through different means, including a scripted or emergent approach. Curiosity is a driving force to learn and explore. The game should give the player the sensory and cognitive stimuli to keep playing, which can be implemented using the topics described in the next sections in different ways.

5.2 Graphics

As graphics hardware has improved through the years, this aspect of games has seen significant changes since the early days of video games. From simple black and white two-dimensional games to almost photo realistic three-dimensional games. In more recent years, the technology for virtual and augmented reality has also improved and become more easily available.

Although the technology for graphically intensive games is available, it is not always used, even in more recent games. Some games strive for realism, while others may aim for a more stylish or simple look. Among the factors that may influence this decision is the genre, target audience, and budget. There are also considerations to be taken regarding accessibility for colour blind or visually impaired players.

There are different reasons for not choosing a realistic visual style. One such reason is a lack of realism in other aspects of the game. Another reason could be to complement other aspects of the game. Ōkami [30], a game based on Japanese storytelling, uses a visual style described as ”sumi-e / cel-shaded / 3D” [31], which complements the story and setting of the game. See Figure 5.1 for an example of the graphics from Ōkami.

Figure 5.1: The graphical style of Ōkami

5.3 Audio

Another important aspect of a game is the audio design. Music sets the tone and intensity for the moment. Audio can also give feedback to actions performed by the player. From the sounds and music, the player can get a hint of what is about to happen. According to Huiberts [32], sound can help the player get immersed in the game, but it can also have the opposite effect if it is not used properly. It is also worth noting that a player might be deaf or hard of hearing, which brings up the subject of accessibility in this regard too.

In some games the audio can be a central part of the gameplay, as it is in the game 140 [33], described by Double Fine as "A musical platformer with super tight puzzle design and a striking audiovisual presentation." [34]. The elements in the game is controlled by the music which, as the name might suggest, is playing at 140 beats per minute. Each element has it's own, recognizable sound that plays to the beat of the music, which means that the player can know the nature of the upcoming obstacles and challenges before they appear on screen. Figure 5.2 shows a screenshot from the game 140.

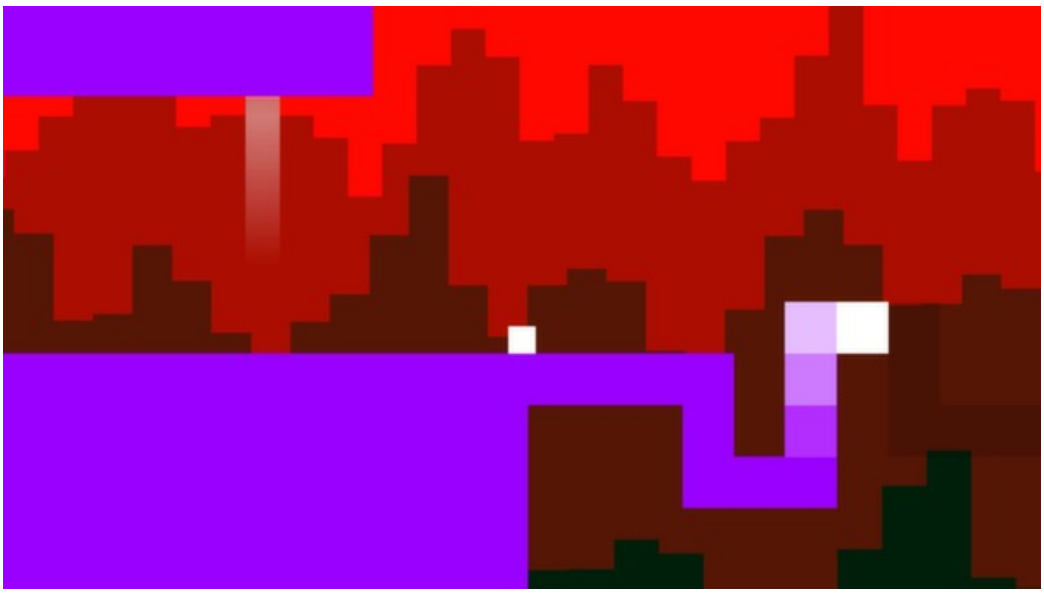

Figure 5.2: Screenshot of the game 140

While audio is one of many aspects in most games, there exists games based solely on sound. The graphics are very simple or completely non-existent. Interaction with the game may be achieved through a normal gamepad or with specialized controllers. An advantage of such games is that they may be as available for a vi-

sually impaired player as a sighted one. BlindSide [35] is one such game where the player navigates a 3D world with nothing but sound. Played on an iOS device, the game uses the top, middle, and bottom of the touch screen along with gyroscopes to control the game. The graphics in the game is a very simple interface for the three zones of interaction on the touch screen.

5.4 Haptic Feedback

The purpose of haptic feedback is to give the player some sort of physical feedback to in-game events. In the context of games, force feedback is another common name [22]. While this can provide new dimension to the game, some may not enjoy it. It is not uncommon for games to provide an option to turn it off.

One common example of physical feedback is vibrating the gamepad. For the N64, this was achieved with the Rumble Pak accessory. It has become a common part of most game controllers in later console generations [22]. Another example of force feedback in interaction devices is steering wheels for driving games. The feeling of steering a car can be simulated by resisting, to some degree, the force asserted by the player on the wheel to turn it.

5.5 Virtual and Augmented Reality

AR is the concept of augmenting the real world with virtual elements, while VR is the concept of showing the user an entirely virtual reality. While VR often is restricted by the need for the user to wear a head mounted display, VR also has the option of using a mobile device. Virtual reality headsets has seen an increase in popularity lately and are expected to really hit its prime in 2016, with devices like Occulous Rift, Microsoft Hololens, and HTC Vive coming. The price point is still a little steep to be present in every home just yet, but it will be interesting to see its progress and evolution [36]

There are many ways of interacting with these systems. One way is to use a camera or sensor based system to track movement and recognize gestures performed by the user. Another is to use specialized input devices, as is possible for HTC Vive [37]. Combining multiple methods of input is another option.

Hover Junkers is a game developed by StressLevelZero entirely for VR [38]. The game tracks the players movement and the player interacts with the world using VR controllers. The player has limited movement within a virtual ship that maps to the movement in the real world, but the ship can move and navigate the much larger virtual world. By doing this, the game can have a large world and still allow the player to play in a smaller room. Figure 5.3 shows a screenshot of a video where a Hover Junkers developer plays the game [39].

Figure 5.3: Hover Junkers gameplay

5.6 Setting and Storyline

A game based only on a core concept may not in itself hold many hours of gameplay. By giving the player a setting to the game and a storyline to play, they may find extra motivation to continue playing and give them a greater interest and immersion in the game [3]. This can be achieved by adding an overarching story, smaller stories, character progression, a game world, and other elements outside the core game concept.

5.7 Social Interaction

Social interaction is a part of the flow in Sweetser's GameFlow model [27], but it can often interrupt the immersion in a game world by providing a link to the real world. However, social interaction plays a big part in video games and some people play games for the social interaction alone. Lazzaro [40] mentions social interaction as one of the four keys to emotion in their article on why we play video games. Many players get their enjoyment out of a game by playing with other people inside, and outside of the game. Players may even play games that they do not like, just to spend time with their friends. Some players use video games as a mechanism for social interaction. Multiplayer games are best at creating social interaction and games with both cooperative and competitive elements offer a wider

variety of emotions in the players. Wang showed how social interaction could be used in a game based student response system, Kahoot!, to better engage students in a lecture [41]. They were able to improve the simple gameplay of a quiz format by including elements that encouraged social interaction among the students.

5.8 Summary

In this chapter some of the different aspects that affect immersion in games have been discussed.

GameFlow is a model made by Sweetser [27] for measuring player enjoyment in video games, and describes eight main elements that has to do with experiencing flow when playing video games. Malone [29] explains game enjoyment with three categories that captures the main characteristics of what makes games fun to play.

Graphics provide an important interface to most games, although some games without a graphical interface exists. Audio can be used both for ambiance and as an essential aspect of a game. Giving the player a form of haptic feedback may also give a stronger sense of immersion. AR and VR provide a whole new way to get immersed in video games by taking up most of, in not all of your vision. This is an interesting area of video games and it will be interesting to see if this technology gains popularity in the masses. Game content is another important aspect. Creating a setting and storyline in a game has a significant influence on immersion and motivation to play the game.

Social interaction plays a big part in why we play video games. People even play games they dislike just to play with their friends. Competition and cooperation create emotions in players, which players use as a mechanism for social interaction.

Immersion is an important part to keep in mind when developing video games and all of the different aspects described here will come into play when creating the game prototype. In this project, th most important aspects learned from this chapter include GameFlow, as it seems like a good tool for reviewing a game, and could be a good model to base the questionnaire on. Social interaction, and how this affects games, is another important aspect, since this is a core concept of this project.

Chapter 6

Game Development Technologies

This chapter will take a look into what technologies are available for independent video game developers in order to create video games. The main topics include game engines, software tools like version control and programming languages, and game resources such as audio and graphics.

6.1 Game Engines

A game engine is a tool to help with the development of a video game. Game exist to abstract away complicated tasks that are common in video games, like physics simulation, visual rendering, graphical user interfaces, and input. By using a game engine, more time can be spent focusing on the gameplay and elements that make up the actual game, such as game design, story, and how different objects interact in the game world. This is especially useful for independent developers which may not have the resources or expertise to create all of the components for a video game from scratch [42].

The benefit of using a game engine is that a lot of the hard work has already been done, usually by someone with expertise, and the game engine has been tried and tested before. However, using a game engine can also be limiting, in the fact that the developer does not have full control over how specific things are handled and has to rely on using the constructs of the game engine in order to achieve the desired outcome. Game engines can also homogenize games, by using the same game engines games often have a similar feel to them in terms of movement and physics interaction [43].

Two of the most popular game engines are Unreal Engine and Unity. Both of these engines have big communities and free entry into creating video games [44]. The two following sections will take a look into the features and limitations of these game engines.

6.1.1 Unreal Engine 4

Unreal Engine 4 is a game engine created by Epic Games which is known for its powerful visual graphics engine. It supports dynamic lighting and shadows, photo realistic shaders, and has a particle system that can handle millions of particles at once. The built-in editor makes it easy to debug and test the game during development with features like Hot Reload and Instant Game Preview. Hot Reload makes it possible to update the game code while playing, and Instant Game Preview is an in-editor view of the game that can be launched to test or preview any part of the game. Figure 6.1 shows the standard view of the Unreal Engine 4 editor.

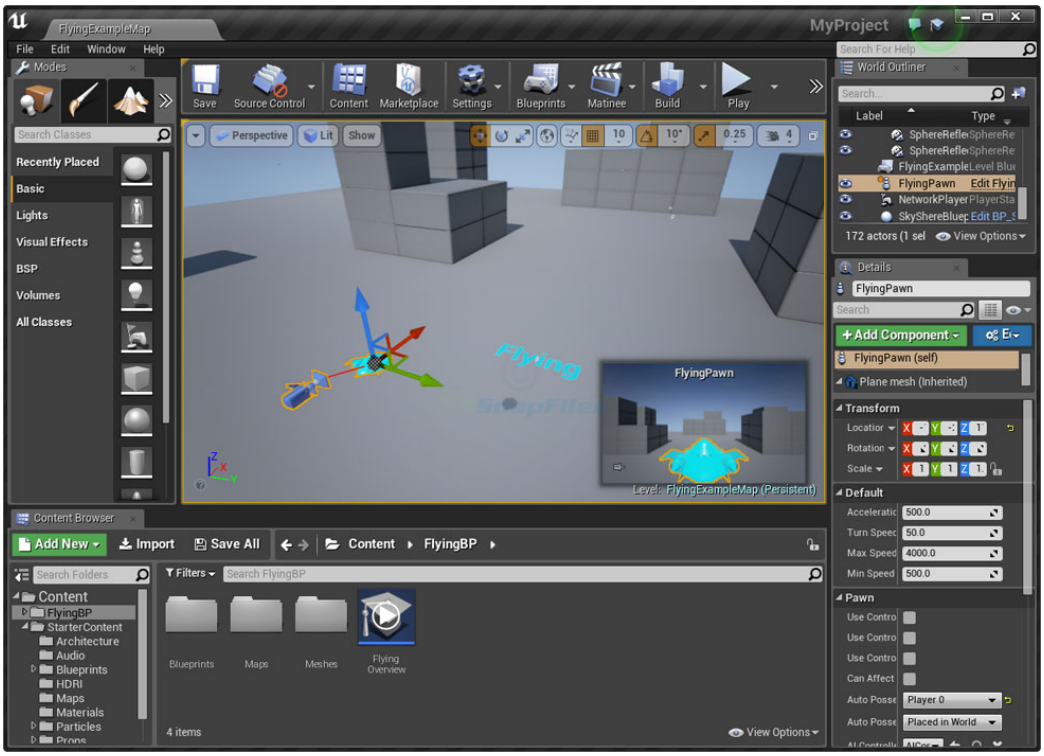

Figure 6.1: The Unreal Engine 4 editor interface

The programming language used when creating games for Unreal Engine 4 is C++, which the engine itself is created with, but it also supports visual scripting which is called Blueprints. Blueprints are meant to be easy to use by anyone, and is a way to create quick prototypes and build playable content without writing code. The full C++ source code for the engine is also made available, which enables customization of the Unreal Engine subsystems, including physics, networking, and the editor itself.

Other notable features of the Unreal Enigne 4 include Artificial Intelligence implementation, support for VR, and integration with middleware technologies, like NVIDIA PhysX, AutoDesk Gameware, and Oculus VR. Unreal Engine is free to use, with a 5% royalty on product revenue after the first USD 3 000 for commercial products. More information about the features and licensing of Unreal Engine 4 can be found on their website [45].

6.1.2 Unity3d 5

Unity3d 5, created by Unity Technologies, focuses on being an easy to use, but powerful game engine. It has support for multiple platforms including desktops, consoles and mobile platforms. Unity has a strong community and is popular among independent developers, especially for creating mobile games. Unity was the world's leading engine for developing mobile games in 2012 [46]. A view of the Unity3D 5 interface is illustrated in Figure 6.2.

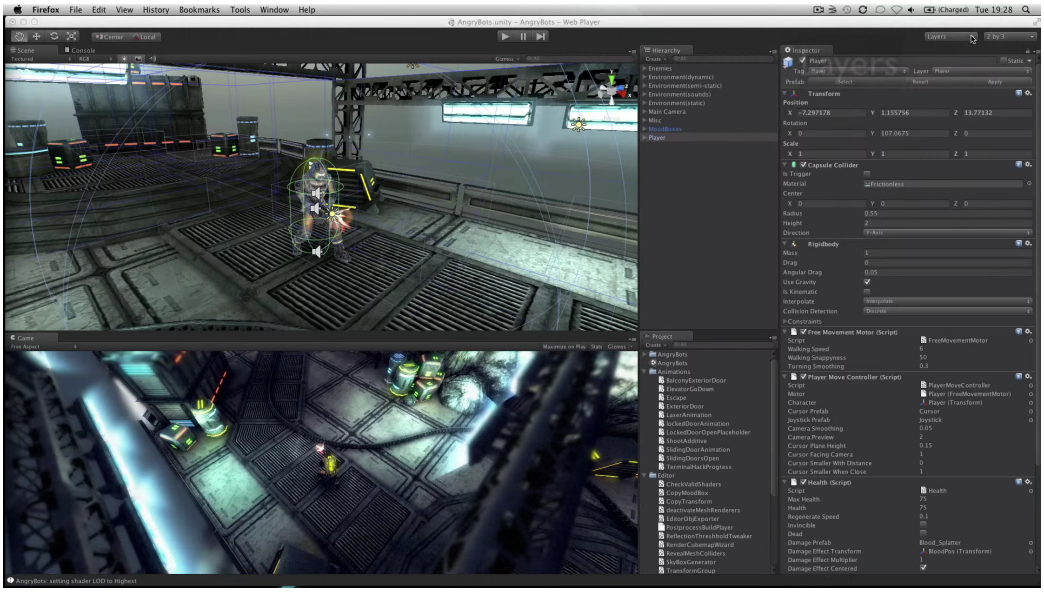

Figure 6.2: The Unity3D 5 editor interface

The Unity editor is made to be extendible tool, and has its own store with over 1 700 free and paid extensions to enhance the editor. These extensions consist of game assets like textures and audio, starter projects, particle systems, and shaders. The editor has a Play Mode, that supports rapid interactive editing, with pausing and frame by frame stepping for easy debugging. Games can also be viewed from the perspective of specific platforms in order to easily test and develop for different

platforms simultaneously. In terms of programming languages, Unity supports C# and UnityScript. UnityScript is a custom language specific for unity and is based on JavaScript. [47, 48]

There are two main versions of Unity3d 5, Personal Edition and Professional Edition, neither of which have any royalty fees. Personal edition is a free alternative made for individuals and small teams with full access to the game engine and its features, at the cost of having a Unity splash screen at the start of the game. There is also a limit to how much revenue and funding that can be received while still using the Personal Edition. Professional Edition has additional features, like cloud services for building and game analysis, and prioritized bug reporting. The price for Professional edition is USD 75 per month, or a perpetual licence for USD 1 500. There are also other alternatives for enterprise and educational editions. The source code for the engine can be licensed for users of the professional or enterprise editions. [49]

6.1.3 Other Engines

There are other feature rich game engines out there for both corporations and individual developers. Licensing for game engines have traditionally been expensive, but they have become more affordable in the later years. In May 2016, CryTek cut their licensing fees completely and released the full source, as well as a pay what you want licensing model without royalties, for their game engine CryEngine [50]. CryEngine is known for producing games with beautiful graphics like the Crysis games. Hero Engine is an engine for creating MMO games with support for seamless transition between maps and a robust AI system. It is quite expensive though with a starting price of USD 99 per year, and 30% of revenue as royalties [51]. GameMaker:Studio is a simple and straight forward engine created by YoYo Games for creating 2D games. It prides itself on being easy to use, and the engine also features a free version [52]. Wave Engine is a relatively new game engine, released in 2013 by Wave Corporation, that focuses on mobile gaming. It is free to use with no royalties, except the obligation to include their logo in the splash screen of the game [53].

Some game engines are not available for licensing though, and are exclusive to the companies that created them. Examples of this include Rage Engine, by Rockstar used for Grand Theft Auto IV, Luminous Engine by Square Enix used for Final Fantasy XIII, and Anvil by Ubisoft used for Assassins Creed [54–56]. An overview of game engines available for independent developers can be found on IndieDB's website [57].

6.2 Game Resources

In addition to game engines, there are other tools and resources needed in order to create and distribute video games. This section gives an overview on what resources are available for independent develops to aid in game making. This section is based on an overview by Pixelprospector, a site that offers information and lists of resources related to these topics [58].

6.2.1 Software Tools

Other than game engines, important software tools include Integrated Development Environment (IDE), project management software, and version control systems. The IDE and programming language used is usually dependent on the game engine chosen. An IDE is a tool for writing source code for the game and normally contain a compiler and a debugger. IDEs are meant to assist the developer in writing code and increasing productivity by providing easy management and editing of the code base for the game. Project management software is useful for managing the development of the software by providing tools for keeping track of tasks and resources for the project. This kind of software allows for easy management of requirements and can help manage which developers are working on different features.

Another important software tool is version control. A version control system records changes to files, in order to recall different versions of the files later. It enables useful features like reverting files to a previous state, reverting the whole project to a previous state, and comparing changes over time. It is also useful if the code somehow gets lost or broken [59].

6.2.2 Graphics and Audio

Graphics and audio, referred to as game assets, are important for the look and feel of a video game and is a central part in creating immersion. Graphics is considered anything visual from 3D textures and models to 2D pixel sprites and backgrounds. Audio is background music as well as sound effects of the game. Getting game assets for a game is usually done in three ways: creating game assets using software, hiring a graphical or sound designer to do it, or acquire pre-made free or paid game assets. The method for acquiring game assets is dependent on the game developers' expertise and resources.

6.2.3 Distribution

The distribution of a video game can be done in different ways, this section only covers digital distribution. Games can be directly distributed via a payment processor, or indirectly distributed via a digital game store. Direct distribution are usually done on the developers website, or on a site provided by a payment processor, where the developer has control of the presentation and distribution of the game. Payment processors are often used to handle the payment itself, they can host the game and offers various payment options like credit card, PayPal, and Google Checkout. Payment processors often take a fee or a percentage of revenue for handling the payment, but some payment processors are free, like itch.io [60].

Distribution can also be done via a digital store, like Steam, GOG, or Humble Store. The main advantage of using a store is that it provides marketing and is a way for players to discover new games.

6.3 Summary

Game development technologies have come a long way to support independent developers. Feature rich game engines like Unity3d 5 and Unreal Engine 4 are available for anyone, and are used by both independent developers and big game companies. There are even game engines which specializes in different genres, and game engines which cater to people with various programming experience.

Game resources is also an important aspect to the development of video games. Software tools like version control systems are freely available for anyone to use and help keeping track of changes over time. Game assets can be acquired through paid or free channels on the internet, and there are online resources for hiring designers. Digital distribution makes it easy for an independent developer to publish their game online either via payment processors or via online game stores like Steam.

For this project, a game engine like Unity3D or Unreal Engine 4 would be helpful for creating the game prototype. Access to free game assets can give life to the game prototype, and a version control system would be helpful in order to provide a backup and to keep track of the changes in the game.

PART III

Design and Implementation

This part describes the design and implementation of the video game prototype that was developed for this thesis.

Chapter 7

Design

This chapter covers the design of the video game prototype that was created. It will provide insight into the chosen technologies for the game, the concept of the game, the requirements, and the gameplay elements of which the game is comprised.

7.1 Chosen Technologies

This section describes the technologies that were chosen for the development of the game prototype.

7.1.1 Interaction Devices

The game will use a PlayStation Move motion controller and a gamepad to control the characters in the game. These interaction devices were chosen with the game concept in mind, more in Section 7.2.2, and supports a design where the players have different roles and will create a unique dynamic between the players. PlayStation Move controller will control the hand that can manipulate the game world, this controller is suitable for representing the players hand with precision and provides buttons for manipulating the environment. The gamepad will control the character that can interact with the game world, and provides a familiar interaction device for a platformer game. The players will have entirely different functions in the game and must use these functions to work together. The PlayStation Move was also chosen to see how a proprietary game controller for a console can be used in a PC game.

7.1.2 Game Engine

The game will be developed using the Unreal Engine 4 as the game engine. This ease the development by handling common video game tasks such as rendering,

collision, and physics. Unreal Engine 4 has a plugin called Paper2D which helps in the creation of 2D games. There is also an unofficial plugin for the PlayStation Move controller to make it easy to use with games developed in Unreal Engine 4. Unreal Engine 4 is a popular game engine with a big community and good documentation online. It also uses C++ as a programming language that the author is familiar with, which is one of the main reasons that it was chosen over Unity3D 5 that has similar features to Unreal Engine 4.

7.1.3 Software Tools

Microsoft's Visual Studio 2015 will be used as the IDE for the development, as it is the default IDE for Unreal Engine 4. The project will also use Git as version control, arguably the most popular version control systems today. A version control system will help keep a track of the changes throughout the development, and serve as a safety net in case the files get lost or corrupted locally.

7.1.4 Game Assets

The game prototype will use free online game resources for graphics and audio. This is chosen to see if a game can be created using free resources, and because the game is supposed to be a prototype with limited monetary resources. The graphics used in the game prototype are created by Kenny Land [61] and are provided with a Creative Commons CC0 1.0 licence, which means that they are free to use for personal and commercial projects. The background music for the prototype is by TomCat Carty [62] and is provided as freeware. Most of the sound effects come from Sonnis.com's Game Audio Bundle [63], and some from JewelBeat.com [64], both of which allow for use in personal or commercial products for free.

7.1.5 Summary

The game prototype will use a gamepad and the PlayStation Move controller to control the game. The game prototype will use the Unreal Engine 4 as the game engine for development. In addition to this, Microsoft Visual Studio 2015 will be used along with Git to aid with the programming, managing of the source code, and will keep a history and online backup of the source code. The game's graphics and audio will come from free online assets.

7.2 Game Concept

The main motivation behind the game concept is to have two players working together using two different controllers to navigate through various challenges.

This is meant to stimulate collaboration between players.

7.2.1 Initial Concept

The initial concept of the game was created in the specialization project with the idea of an action platformer with two players that have different roles [1]. Player one controls a game character using a traditional gamepad, while player two controls the game world using a motion control device. The players have to cooperate and help each other to progress through the game and complete the different levels of the game. Figure 7.1 shows the original game concept with one player using a gamepad to control a character and the other player using hand gestures.

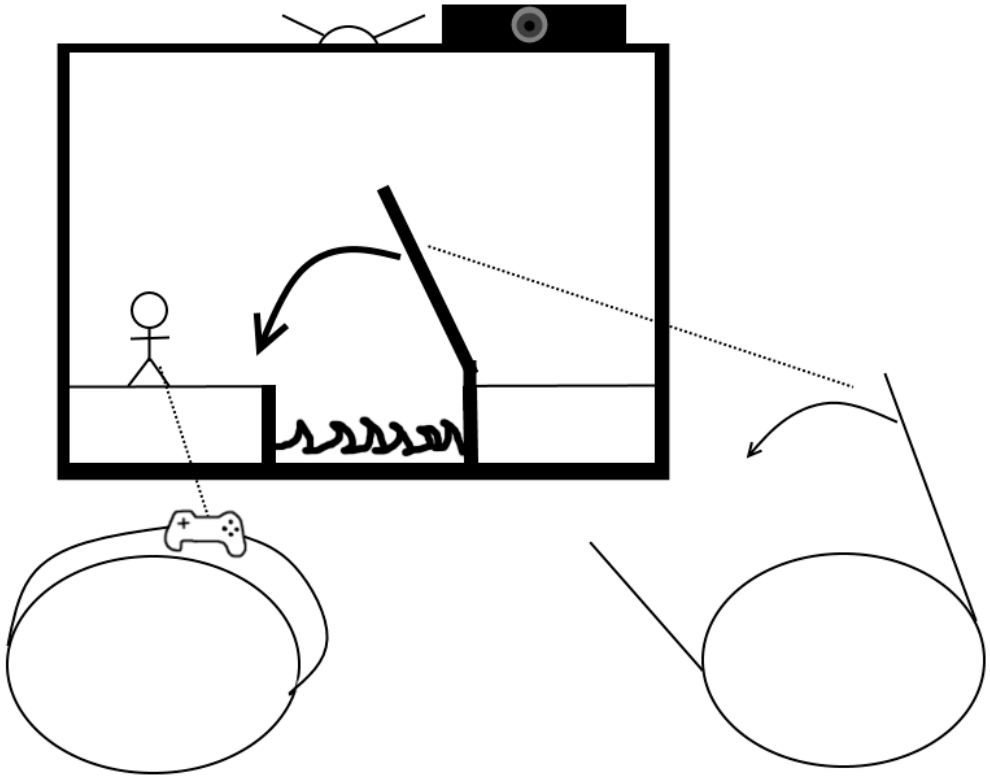

Figure 7.1: Early concept sketch: One player lowers a bridge using hand gestures to help the other player cross the water

7.2.2 Final Prototype

The core concept of the game is cooperation; neither of the players should be able to complete a level without the cooperation of the other player. Both players

should be equally engaged, but have different roles and tasks during the game. The original concept experienced an evolution throughout the development period, detailed in Section 9.1, and the final concept of the game prototype took shape.

The game is a puzzle platformer that has different levels where the players have to cooperate to reach the goal. Player one has to reach the goal of each level, but cannot do so because there are obstacles in the way that can only be manipulated by player two. In order to unlock the goal the character needs to collect a certain number of coins on that level. To be able to reach some places, player two needs to place blocks that the character can stand on, creating bridges and towers. There are enemies in the game that will chase the character and hurt them, but the players cannot hurt the enemies. The player controlling the hand can lure enemies away by using bait, allowing the player to access areas guarded by enemies. Player two can also activate switches that can control moving platforms, or make coloured boxes appear and disappear. Figure 7.2 shows a part of the first level where the hand must activate coloured switches in order to remove the corresponding coloured boxes that block the players movement.

The game prototype has a total of five levels with varying degrees of difficulty and different puzzles to solve.

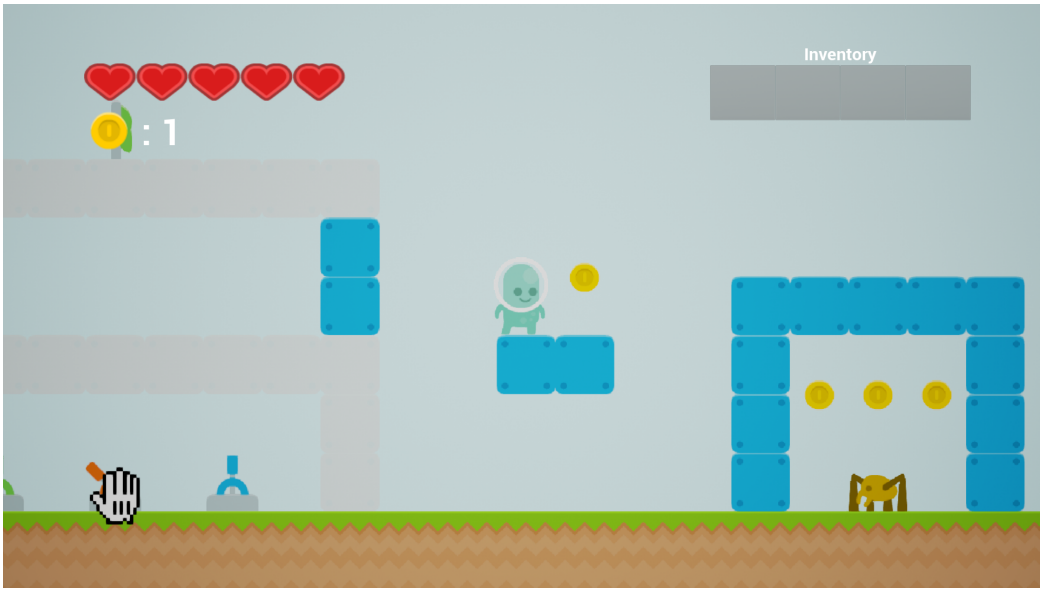

Figure 7.2: A puzzle with coloured switches that control the existence of corresponding boxes

7.3 Requirements

Because the game prototype is a creative initiative, there were no pre determined requirements. The requirements presented here are meant to make sure that the prototype adheres to the core concepts of the game and allows for interesting gameplay that encourages social interaction and stimulates collaboration between players. Table 7.1 contains a list of the requirements for the game prototype. These requirements will be helpful when creating and testing the prototype to ensure that they are met.

Table 7.1: The requirements for the game prototype

ID	Description
	Functional requirements
RQ1	The game must support two players
RQ2	The two players must have different roles in the game
RQ3	Player one controls a character that can interact with the game world
RQ4	Player two contols a hand that can manipulate the game world
RQ5	The character will be controlled by a gamepad
RQ6	The hand will be controlled by the PlayStation Move
RQ7	The game must support multiple levels
RQ8	All levels must have a visible goal that leads to the next level
RQ9	The player must collect coins in order to unlock the goal
	Usability requirements
RQ10	All levels should require both players involvement
RQ11	The two players should be equally engaged
RQ12	The game should have different challenging game obstacles
RQ13	All levels should be possible to complete

7.4 Gameplay Elements

This section describes the different gameplay elements that make up the game prototype.

7.4.1 Character

The character is represented by an alien avatar, illustrated in Figure 7.3, which can move around and interact with the game world. It is controlled by player one, and its only controls are left-right movement and jumping. The character may not harm enemies, nor interact with elements that can manipulate the world, like switches

Figure 7.3: The visual representation of the character

or bait. The primary objective of the character is to collect coins and reach the exit of the level while dodging enemies, pitfalls, and spikes to stay healthy. When the character loses all of its health, the character dies and the level is restarted.

7.4.2 Hand

Figure 7.4: The visual representation of the hand

Shown in Figure 7.4 is the hand, which is a powerful entity that can manipulate the game world. The hand can move up, down, left, and right, and is not affected by gravity or obstacles in the game world like walls and enemies. By using switches and items, the hand can manipulate the game world in order to provide a safe passage for the character.

7.4.3 Springboard

Figure 7.5: The springboard is ready to launch the player.

The springboard is pictured in Figure 7.5 and will launch the character high up in the air whenever it jumps on the springboard. This can be used to reach places that is otherwise to high for the player to reach. The spring board will only launch players and not enemies or items.

7.4.4 Enemies

Figure 7.6: The snail(left) and the spider(right) will chase the character.

There are two different enemies in the game: the spider and the snail, both pictured in Figure 7.6. The snail and the spider will both chase the character when it gets close to them, but the spider has a higher movement speed than the snail. If the enemies reach the character they will damage the character and cause it to be knocked back. The enemies cannot be killed by the character or hand directly, but they can be lured off a cliff. The spiders can be lured away by the hand by using bait, or lured off a cliff by making it follow the player off.

7.4.5 Spikes

Figure 7.7: Spikes are harmful for the character to jump on

Spikes can be seen in Figure 7.7 and is harmful for the character. When the character hits the spikes from above, it will take damage and be knocked up in the air.

7.4.6 Items

Items are building blocks that can be picked up by the character and then placed in the world by the hand. Figure 7.8 shows a building block item. The items are found throughout a level and it must first be picked up by the character before the hand can place it in the world. When an item is picked up, it goes into an inventory, displayed in the top right of the screen, which the hand can then click on to grab the item to be placed. The character can not stand on the items unless

Figure 7.8: A grass building block item

they have been placed by the hand, and the hand may not pick up the item. This is done do create a dependency between the character and the hand. After the item has been placed it can be grabbed again by the hand in order to reduce the penalty for erroneous placement. Multiple items can be used together in order to create a bridge or a tower for the character to reach otherwise unreachable places.

7.4.7 Bait

Figure 7.9: Bait can be used to lure spiders

The bait, pictured in Figure 7.9, is used for luring spiders away from the character in order for the character to reach an area guarded by a spider. Bait works like items where the character picks the bait up from the world, the bait gets added to the inventory, and the hand can then place the bait out in the world. When a spider gets close to a bait, it will run to the bait and start eating it, ignoring the player. When the spider finishes eating the bait, it will continue chasing the character if it is close. The bait can be placed in the world by the hand to immobilize the spider for a short period, or the bait can be held by the hand to make the spider chase the hand instead of the player.

7.4.8 Moving Platforms

Moving platforms, shown in Figure 7.10, are platforms that can be set in motion or stopped by a switch. The character can stand on the platform and move along with it, but only the hand can start and stop the movement of the platform. The switch to turn the platform on or off can either be attached to the platform itself, or it could be stationary, attached to the floor or a wall.

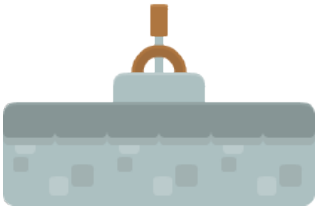

Figure 7.10: A moving platform with the switch attached

7.4.9 Coloured Blocks

Figure 7.11: A basic opaque colour block

Coloured blocks work similar to the moving platforms in that they are also controlled by switches. A basic colour block is shown in Figure 7.11, and a level with multiple colour blocks can be seen in Figure 7.2. The colour blocks can be activated and deactivated using a switch, which makes them appear or disappear in the game world. When a colour block is deactivated it becomes transparent and does not block the movement of the character. However, when the colour block is activated, it becomes opaque and blocks the character movement. Activating and deactivating the coloured blocks can create bridges, open doors, and trap enemies, but it can also make the player fall or even trap the player by blocking its movement.

7.5 Summary

The game prototype contains different gameplay elements that are meant to encourage cooperation between the players. In order to complete the levels, the players need to work together to solve the puzzles involved. The main objective of the character is to run and jump around while collecting coins, picking up items for the hand, and avoiding enemies in order to reach the goal. The character can not reach the goal though without the hand placing items, activating moving platforms and colour boxes, and luring enemies away with bait. Figure 7.12 shows the character using items placed by the hand in order to reach a ledge and avoid an enemy.

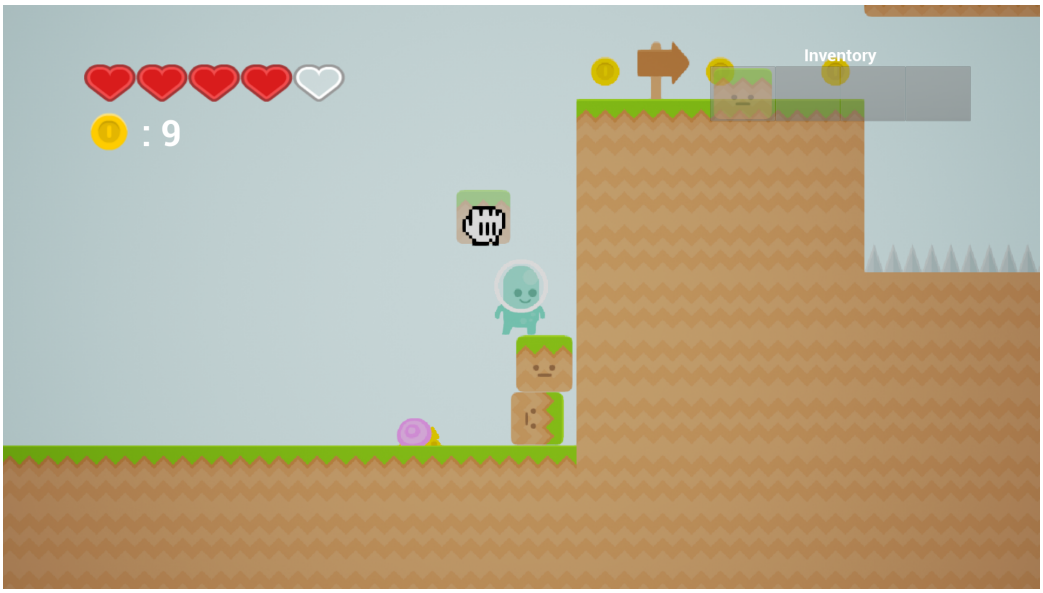

Figure 7.12: The character and hand must work together to reach the ledge

Chapter 8

Architecture

This chapter describes the software architecture of the game prototype. It contains architectural drivers, patterns, views and tactics.

8.1 Architectural Drivers

This section will describe the motivation behind the architecture, the expected life time of the game prototype, and limitations for the implementation.

8.1.1 Motivation

The game prototype would be played by children, so the game controls and gameplay had to be easy to understand. The author also wants the game to be easy to develop for, so the game had to be easy to modify and add new features. This set the focus for the quality attributes usability and modifiability. With high usability it should be easy for the children to enjoy the game without being frustrated by the controls or gameplay elements. Focusing on modifiability it should be easy for the developer to add new features, like enemies, items, and levels.

8.1.2 Expected Life Time

The game prototype will be developed for a play testing session during the course of this project, after which the development will end. The game will not be ready for release and will be a purely gameplay prototype without a story, menus or other common non-gameplay elements. It should be possible for the author or someone else that wants to continue the project to continue working on it after the termination of this master thesis. Information on possibilities for future work is detailed in Chapter 15.

8.1.3 Limitations

The game will be developed using Unreal Engine 4, so the project will be limited to use of the underlying architecture of that engine. Included in the engine is a plugin called Paper2D that will be used, which helps with the creation of 2D games. The author has no previous experience with Unreal Engine, which presented challenges and opportunities to develop new skills.

8.2 Unreal Architecture

The Unreal Engine uses a modular architecture where the engine is comprised of different modules for elements like networking, shaders, and physics. In a projects settings, the developer can define the different modules that are needed. When creating a game, the developer will mostly be working with the gameplay framework of the engine. The following sections describe the gameplay framework. The information presented is taken from the Unreal Engine's online documentation [65].

8.2.1 Players and Entities

All visual objects in the game world, like players, enemies, and items all share the base class Actor. An Actor is basically any object that can be placed into a level, they support basic 3D transformations, such as translation, rotation and scale. The Pawn class is the base class for all Actors that can be controlled by players or Artificial Intelligence (AI), and are set up to easily accept input. Character is the class for representing a humanoid-style Pawn, which has a component for movement and collision detection.

8.2.2 Control and Input

There are two main ways of controlling a Pawn, by using a PlayerController or an AIController. Both classes share the base class Controller. The Controller is responsible for directing the Pawn which is referred to as possessing a Pawn. The PlayerController is the interface between the Pawn and the player, it handles the input from the player and issues commands to the Pawn. The AIController class responds to input from the game environment and uses AI logic to command the Pawn.

8.2.3 Display

There is a HUD class which is the base class for displaying elements overlaid on the screen. Every PlayerController has an instance of the HUD class which

draws their individual viewport. The Playercontroller also has an instance of a PlayerCameraManager class, which represents the view of the player.

8.2.4 Game Rules

The rules of the game are defined and implemented in a GameMode class. A game should implement functions or variables that set or enforce game rules in a subclass of the GameMode class. There can only be one game mode at any given time, and in the context of networking games, the GameMode will only be present on the server. GameState contains the state of the game, which can include the score, location of chess pieces in a chess game, or what tasks have been completed. There is also a class called PlayerState for keeping track of the state of a player in the game. Things like player name, experience level or score are data that would be present in the PlayerState.

8.2.5 Game Framework Overview

Figure 8.1 shows how the gameplay classes interact with each other. The game is made up of a GameMode which sets and enforces the game rules and a GameState which describes the state of the game. Players are represented by PlayerControllers which can possess Pawns in order to control them. Pawns can also be controlled via an AIController that controls the Pawn using AI. The PlayerController provides the player with input controls for controlling the pawn, a HUD for displaying an overlay on the screen, and a PlayerCameraManager for handling the players view in the game world.

8.3 Game Architecture and Views

The architecture for the game is based on the Unreal Engines architecture and uses C++ classes and Blueprints. All classes and Blueprints inherit from a class in the Unreal Engine framework. Figure 8.3 shows the inheritance between classes and illustrates how the games architecture is linked with the architecture of Unreal Engine. The classes are organized into directories which form different groups of classes. The following sections will explain these groups in greater detail. The folder structure of the C++ game source is illustrated in Figure 8.2.

8.3.1 World Objects

The world objects are static objects that exist in the game world that the character or the hand can interact with. The classes that are grouped as world objects are Coin, Goal, MovingPlatform, Spikes, SpringBoard, KillVolume, Switch, ColorSwitch

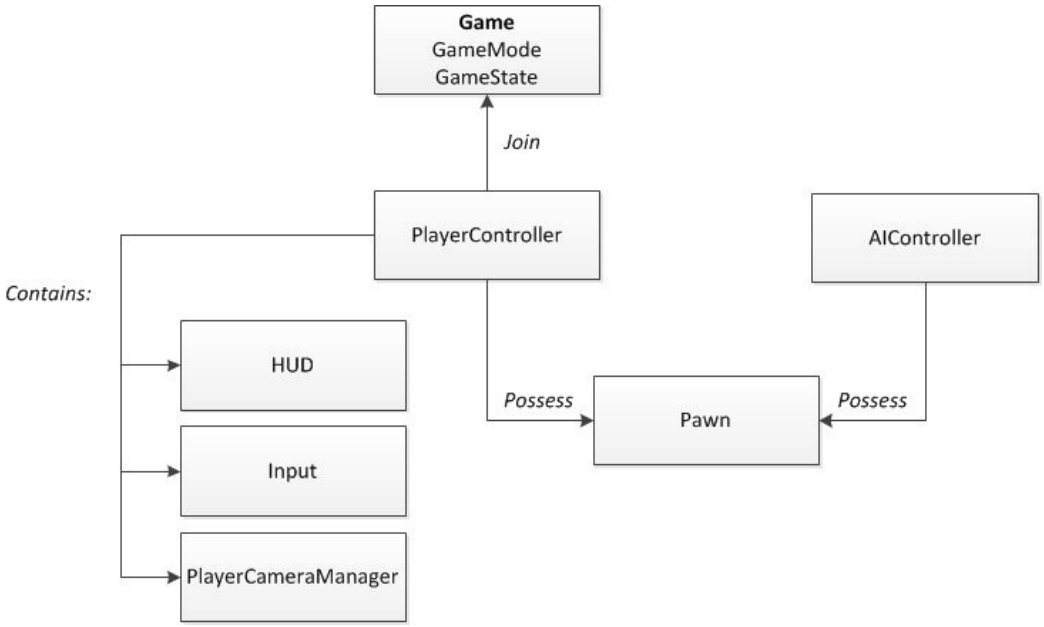

Figure 8.1: The relationship between the game framework classes [66]

and ColorBox. Most of these represent gameplay elements that are explained in Section 7.4. KillVolume is a trigger to kill the character when it moves out of bounds of the game world. Goal is the goal that the character has to reach in order to complete the level. ColorSwitch inherits from Switch in order to reuse functionality. The difference between the two types of switches is that the regular Switch controls one or more MovingPlatforms and the colour switches control the existence of coloured blocks.

8.3.2 Items

Items represent the items that the character can pick up and add to the inventory. There are currently only two classes of items: the Item class and the Bait class. The Item class is made in such a way that it is easy to subclass it and alter the functionality of a new type of item. The Bait class does this and only alters two functions from the Item class.

8.3.3 Game

The game category only consist of the MainGameMode class. Becase the game prototype does not have very complex rules or game states, and there is no need for networking, the functionality in the MainGameMode includes functionality

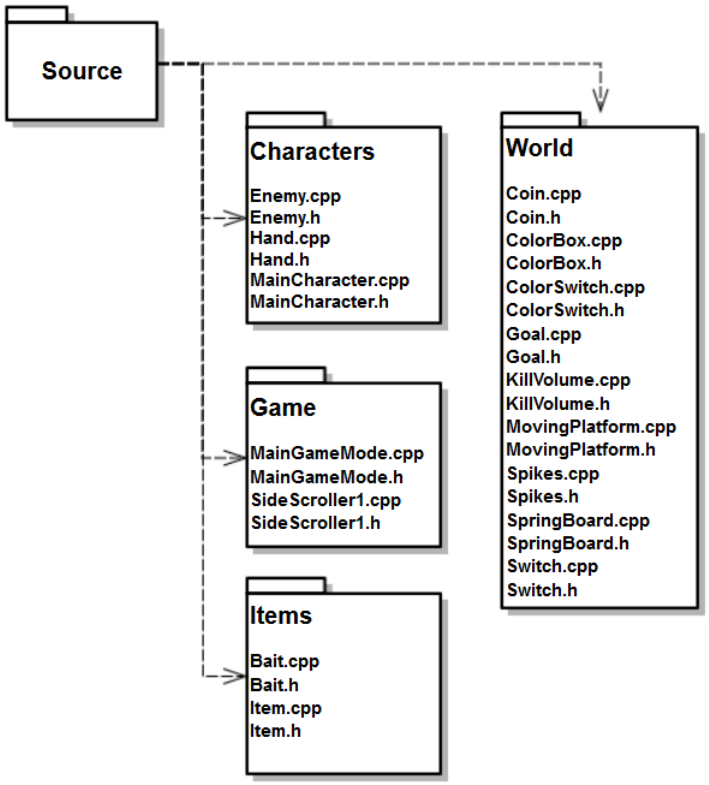

Figure 8.2: The folder structure of the source files

that normally belongs in the GameState. For the same reasons, there are also no PlayerStates. The MainGameMode handles the display of the HUD, keeps track of the score and inventory, and what item is currently grabbed by the hand.

8.3.4 Characters

The characters are the classes that move around in the game world and consist of the MainCharacter, Hand, and Enemy classes. These classes are sub-classes of PaperCharacter which is a specialization of the Character, found in the Paper2D plugin. Normally non-human controlled pawns are not Characters, but the enemies share similar movement and collision detection of the player, so they were made into characters.

8.3.5 Blueprints

The blueprints were used in such a way that they only changed a few attributes of their super-classes. All the blueprints, except for the HUD blueprints, were based on a C++ class where the majority of the functionality existed. Most of the additions of the Blueprints were minor, like using a different sprite, or changing the movement speed of an enemy. Blueprints were convenient for this because of the ease of changing attributes while testing the game. Using Blueprints helps with the modifiability, by making it easy to change attributes and appearance of the different game elements.

8.4 Summary

The motivation for the architecture is to focus on high usability and modifiability. Usability is achieved in part by using the Unreal Engine to handle controller input, physics, collision and rendering which makes for a smooth gameplay experience. The other part of usability is to make sure that the gameplay elements implemented are fun, yet challenging. Modifiability is ensured by making the gameplay elements easily extendible through C++ classes of Blueprints. The Unreal Engine's gameplay framework also promotes modifiability by using a modular system that increases cohesion.

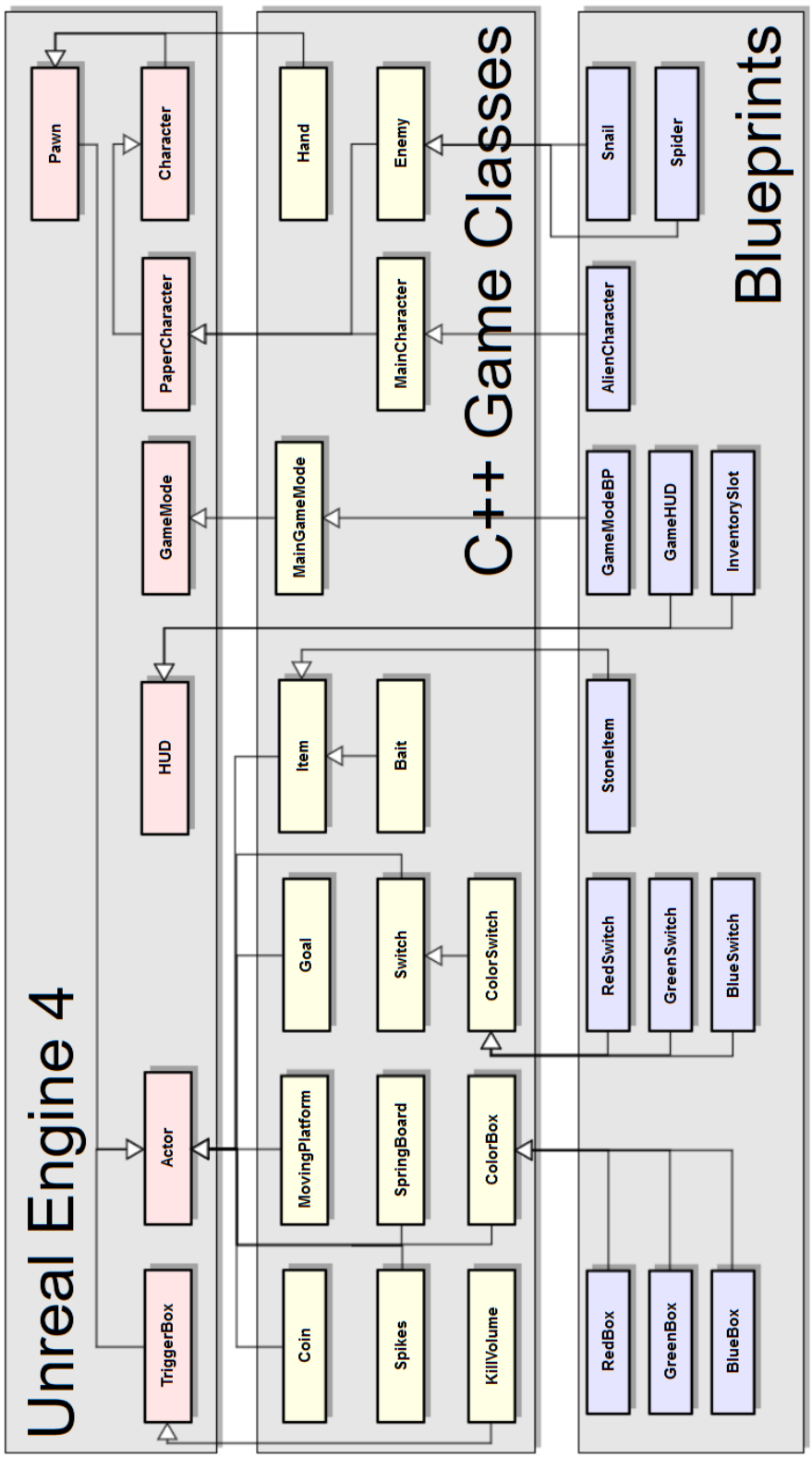

Figure 8.3: Class diagram for all the classes of the game prototype

Chapter 9

Implementation

This chapter explains the development process of the game prototype, the testing that was done, and hindrances during development.

9.1 Development Process

The development of the game prototype started on the 1st of February and ended on the 11th of May, the date of the play testing. The overall process for development was a weekly cycle, where at the end of each cycle there was a meeting with the advisor. The purpose of the meetings was to discuss the work done during that week, and plan the work for the next week. The meetings usually started with a test or showcase of the state of the game prototype, followed by a discussion on what elements had been implemented from the previous meeting and what new features that were to be implemented next. This development process is similar to Scrum or Extreme Programming, but there were no customer requirements, but rather a list of features to be implemented each week. During development the focus was on the features that were decided upon in the weekly meeting prior.

The development can be split into four main phases which will be discussed in the following sections.

9.1.1 Early Phase

The early phase made up the bulk of February and was spent getting used to the Unreal Engine and getting core gameplay mechanics working. It started with creating a basic side scrolling game where a character could move around and jump on different platforms. Some core mechanics was implemented here, like having items that the character could pick up. The character was controlled by the keyboard, and it was possible to use the mouse to place items in the world. The inventory with a HUD was created during this phase and items that the character

picked up ended up in the inventory, that could be clicked with the mouse to select and placed in the world.

9.1.2 Main Gameplay Elements

After getting to know the Unreal Engine better, this next phase was during the weeks of March and focused on creating more gameplay elements. Most of the main gameplay elements were developed during this phase including coins, springboard, and spikes. A few different testing levels were created in order to properly test these new elements. Enemies were also introduced during this phase along with a HUD to display the character's health and number of coins collected. During this phase there was a graphical overhaul of the game prototype which helped with creativity and made it easier to develop new features.

9.1.3 Additional Features and Sound

With most of the gameplay elements implemented it was time to get the game ready for testing. This phase was during the first three weeks of April. The focus of this phase was to create more dependencies between the players, and having the necessary elements for creating interesting levels. The bait and coloured boxes were developed during this period to give the second player more interaction with the game world. Background music and sound effects were also added during this period.

9.1.4 Level Design and Play Test

Up until the middle of April there were only a few basic testing levels without much dynamic interaction between the players. The last phase started around the last two weeks of April and ended at the 11th of May with the play testing. The main purpose of this phase was to get the game to a playable condition with some interesting levels. The five main levels of the game prototype was designed in this phase, three smaller introductory levels, and two larger more challenging levels. A lot of time went into getting the PlayStation Move controller to work with the game, but it was abandoned due to not working satisfactory. Due to this the keyboard and mouse that had been used for internal testing was replaced with a gamepad for each player. The controls were basic with only one movement joystick and one action button for each player. More information about why the Move controller was abandoned can be found in Section 9.3, along with some other problems and limitations.

9.2 Testing

Testing was done throughout the development period by the author, and during the weekly meetings with the advisor. Unreal Engine made it easy to test the game from any point with their Play In Editor feature, which could be activated by the click of a button. Some test levels were created during development in order to test how the different gameplay elements interacted with the character and the hand. Some final testing was performed in order to confirm that the game prototype adhered to the requirements that was set. The results of this testing and what requirements are met by the game prototype, can be seen in Table 9.1.

The first four requirements, RQ1-RQ4, are met by the game prototype having support for two players that have separate controls.Requirements RQ5 and RQ6 has to do with the interaction devices that the game uses, RQ5 is met by using a gamepad, but RQ6 states that the game will use the PlayStation Move to control the hand, which it not true. The reason for this is detailed in section 9.3. Requirement RQ7, is met by supporting multiple levels, which the game does by using Unreal Engine's UWorlds. The requirements RQ8 and RQ9 are met by ensuring that all the levels have a goal, visualised by a flag, that can be unlocked by collecting a certain number of coins (adjustable per level).

The usability requirement are a bit more difficult to judge whether they have been met or not, but RQ10 and RQ11 are guaranteed to be met by the five levels designed for this prototype. The requirements RQ12 and RQ13 are strived for when designing the levels but it is hard to ensure equal engagement and challenging game obstacles, as this depends on the skills of the players.

9.3 Limitations and Obstacles

The development period was pretty straightforward without many obstacles or limitations. One limiting factor was that there was only one person developing and testing the game, besides the weekly meetings with the advisor. This made it hard to judge the possible interactions between two different players when developing. Some level elements might be trivial to overcome with just one player, but more challenging with two players and vice versa. Being only one person also meant that it was hard to control two controllers at once, so most of the testing was done using a mouse and keyboard.

Another obstacle was the creative aspect of developing a game. The levels and game elements had to be fun and challenging for both players, and the game was supposed to encourage social interaction and cooperation between the players. The weekly meeting with the advisor helped a lot with getting the game elements right by providing a second opinion and new ideas.

Table 9.1: Results of the testing to see if the requirements were met

ID	Description	Met
	Functional requirements	
RQ1	The game must support two players	Yes
RQ2	The two players must have different roles in the game	Yes
RQ3	Player one controls a character that can interact with the game world	Yes
RQ4	Player two contols a hand that can manipulate the game world	Yes
RQ5	The character will be controlled by a gamepad	Yes
RQ6	The hand will be controlled by a PlayStation Move controller	No
RQ7	The game must support multiple levels	Yes
RQ8	All levels must have a visible goal that leads to the next level	Yes
RQ9	The player must collect coins in order to unlock the goal	Yes
	Usability requirements	
RQ10	All levels should require both players involvement	Yes
RQ11	The two players should be equally engaged	Yes
RQ12	The game should have different challenging game obstacles	Yes
RQ13	All levels should be possible to complete	Yes

The game prototype was originally going to use the PlayStation Move controller for controlling the hand by the second player, but was later abandoned for not functioning properly. The Move had been originally tested to work with Unreal Engine during the first week of the project, so it was not seen as a major problem to implement later in the development. When the time come to get the Move controller integrated into the game some problems occurred. The plugin that was used to interface with the Move only provided a simple example to control an object on the screen. In order to get the satisfactory behaviour of the move controller in the game, access to the raw data from the camera tracker was needed in order to determine the position of the controller. This was impossible to do with this plugin without the use of a virtual reality headset, because the plugin was integrated with the VR part of Unreal Engine. The creator of the plugin was contacted in order to try to solve this problem, but the creator had discontinued support of the plugin in order to create a new system that was independent of Unreal Engine. After much trial and error, the Move controller was eventually abandoned in favour of using a gamepad for controlling the hand.

9.4 Summary

The development of the game prototype was an exciting and challenging experience. The weekly cycle of development worked well, and helped set intermediate goals throughout the development period. The Unreal Engine was very helpful, but took some getting used to in the beginning. Being a single developer creating a two-player game presented some challenges, but weekly meetings with the advisor helped combat those challenges. Even though the final prototype did not use the Move controller as planned, the play testing was still a success, and having similar controllers might even have helped the cooperation between the players. The game prototype was ready with five playable levels on the day of the play testing, which is documented in Chapter 11.

PART IV

Experiment

This part shows the preparation, results, and the analysis of the play testing that was performed.

Chapter 10

Planning

This chapter describes the planning that went into finding a suitable group that fit the target audience of the game prototype and the questionnaire that were developed to evaluate the results of the experimentation.

10.1 Preparations

Since this project and one of the research goals is aimed towards children, the play testing should be preformed with children. The natural place to look for children are in schools, so a local school was contacted to try to set up a play testing session with the school children.

Bente Sandø the leader of the after school programme for Buvik School was contacted by email to inquire about the possibility to preform the play testing with their pupils. The response was positive and the planning on how the play test would be executed was started.

10.2 Questionnaire

A questionnaire was developed in order to evaluate the results of the play testing. The questionnaire is based of the Gameflow framework [27], and the questionnaire created by Fu et al [67]. The questions are divided into groups corresponding to the Gameflow categories: Concentration, Goal Clarity, Feedback, Challenge, Autonomy, Immersion, Social Interaction, and Game Enjoyment. The final version of the questionnaire can be seen in Table 10.1. The questions were tailored towards children by simplifying them and making them easier to understand. Gameflow is discussed in Section 5.1. The questions were given with a four-point rating scale. This was chosen, as opposed to a five-point scale or greater, to make it easier for the children to respond, and easier for the examiners to interpret their answers.

Note: the questions were asked verbally to the children, in Norwegian, and their answers were logged by the interviewer. The original Norwegian questionnaire is included in Appendix 15.

Table 10.1: The questions from the questionnaire

ID	Question
	Concentration
Q1	I am concentrated when playing the game
	Goal Clarity
Q2	It was easy to understand the goal of the game
	Feedback
Q3	It is easy to understand when I do something wrong in the game
Q4	It is easy to understand when I do something correct in the game
Q5	I liked the graphics of the game
Q6	I liked the sounds of the game
	Challenge
Q7	I get bored when playing the game
Q8	The game was too easy
Q9	The game was adequately difficult
Q10	The game was too hard
Q11	My skill gradually improved when playing the game
	Autonomy
Q12	I felt in control of the movements of the player
Q13	I could recover from errors I made in the game
	Immersion
Q14	Time goes by fast when plaing the game
Q15	I become unaware of my surroundings when playing the game
	Social Interaction
Q16	I was competing with others to reach the furthest
Q17	It was important to communicate when playing the game
Q18	It was important to cooperate when playing the game
Q19	I cooperated well with the other player
Q20	The other player cooperated well with me
	Game Enjoyment
Q21	I enjoyed playing the game
Q22	I would like to play this game at home

Chapter 11

Play Testing

11.1 Plan

After creating the questionnaire and establishing the date and location for the play testing, the testing session was planned. The play test would occur from 2PM to 5PM on the 11th of May on Buvik School's after school program.

There were two interviewers: the author and the supervisor, which meant that two tests could be run simultaneously with a total of four children at once. The testing of the video game prototype was planned to be about 15 minutes in length with about five minutes of interviewing the children after, with the help of the questionnaire. This turns out to a total of 36 with maximum efficiency for the three hours planned.

This number is quite inflated, because it does not take into account setup and conclusion time, and the questionnaire time would be variable from child to child. However, this number was considered to be a maximum, but more realistically 20-25 participants were expected.

11.2 Execution

The setup for the play testing started at 2PM, in a medium sized room where two laptops were stationed with two controllers each. There were some minor issues in the beginning, where the laptops were not recognizing the controllers, and the USB ports on one of the laptops were not working. One laptop also did not have the most updated game prototype, so it had to be updated, but the issues were eventually solved. About 15-20 minutes went to organizing the room and solving the initial problems.

After fixing the starting problems, the children were sent in four at the time, two on each computer where they played for about 15 minutes each. After that the

children were interviewed, and the next set of children were sent in.

The children were explained the basic rules and controls of the game, and were then observed for how they would try to solve the different levels. If they got stuck for a while, they would be helped, but they were mostly without input aside from the occasional hint. Figure 11.1 shows the play testing on one of the laptops.

Figure 11.1: The children received some hints if they were struggling

A problem occurred with one of the laptops, where the game would start to drop frames and the players experienced considerable input lag. This started after the first pair of children and persisted for the rest of the testing. Therefore towards the middle of the testing this computer was deemed too unplayable to accurately represent a proper experience of the game prototype and was discontinued for the rest of the testing. The issue is believed to be an artefact of a failing graphics card.

Towards the end of the play testing session most of the children had been picked up from the after school program and the testing was terminated. There were in total about two hours of testing with a total number of 19 participants. The number is odd because one of the children was picked up in the middle of the play testing.

11.3 Summary

Despite the initial technical difficulties and one of the laptops failing, the play test was deemed a success. The children all seemed excited to play and were attentive and well behaved when both playing and answering questions. Aside from the one child that was picked up in the middle of playing, all of the children that participated answered all of the questions. There could have been better preparation before the test to identify the faulty laptop, and making sure that they were both updated with the latest version of the game.

Chapter 12

Results and Analysis

This chapter presents the results of the play testing session along with observations that were made and analysis of the results.

12.1 Results

Table 12.1 shows the disagreement and agreement of the questions that were asked. For simplicity's sake, agreement and strong agreement have been aggregated, likewise with disagreement and strong disagreement. More detailed information regarding the results, like statistical mean, variance and standard deviation of the questions from the questionnaire can be found in Appendix B. The results show that the children were generally very positive to the game and its features. The ages of the participants were ranging from six to nine years old with the composition being: six six year olds (32%), five seven year olds (26%), five eight year olds (26%), and three nine year olds (16%). Out of the total of 19 participants, 8 of them were female (42%). Over half of the children told that they had a lot of prior game experience (58%), seven said that they had some game experience (37%), and one told that they had no prior game experience (5%).

12.2 Observations

This section describes some general observations that were made during the play testing which were observed by the interviewers. The length of the questionnaire was an appropriate amount of questions, it could not have been any longer as some children seemed to lose their attention at the end. It was also important to explain the questions well, as some of them were not as easy to grasp for the children. Some questions also needed to be asked twice in order to get a sense of scale of agreement or disagreement. Questions like: "do you strongly agree or just

Table 12.1: The results from the questionnaire

ID	Question	Disagree	Agree
	Concentration		
Q1	I am concentrated when playing the game	0,00%	100,00%
	Goal Clarity		
Q2	It was easy to understand the goal of the game	47,37%	52,63%
	Feedback		
Q3	It is easy to understand when I do something wrong in the game	21,05%	78,95%
Q4	It is easy to understand when I do something good in the game	10,53%	89,47%
Q5	I liked the graphics of the game	0,00%	100,00%
Q6	I liked the sounds of the game	21,05%	78,95%
	Challenge		
Q7	I get bored when playing the game	89,47%	10,53%
Q8	The game was too easy	84,21%	15,79%
Q9	The game was adequately difficult	42,11%	57,89%
Q10	The game was too hard	63,16%	36,84%
Q11	My skill gradually improved when playing the game	10,53%	89,47%
	Autonomy		
Q12	I felt in control of the movements of the player	31,58%	68,42%
Q13	I could recover from errors I made in the game	31,58%	68,42%
	Immersion		
Q14	Time goes by fast when plaing the game	15,79%	84,21%
Q15	I become unaware of my surroundings when playing the game	10,53%	89,47%
	Social Interaction		
Q16	I was competing with others to reach the furthest	68,42%	31,58%
Q17	It was important to communicate when playing the game	0,00%	100,00%
Q18	It was important to cooperate when playing the game	0,00%	100,00%
Q19	I cooperated well with the other player	10,53%	89,47%
Q20	The other player cooperated well with me	10,53%	89,47%
	Game Enjoyment		
Q21	I enjoyed playing the game	0,00%	100,00%
Q22	I would like to play this game at home	0,00%	100,00%

slightly" were frequently asked when the detail was not provided in the initial response of the child.

The majority of the children were very cooperative with each other and there were a lot of communication between them on things like how to conquer the different obstacles and what to do next. Learning the controls of the alien and the hand character was the focus of the first level, but the movement for the alien character was more complex, which led to some players wanting to switch controllers. However on the following levels the tasks for the hand character were more complicated and the players often wanted to switch back to the hand character. The game was possibly a little too hard for the children that we tested, as some were struggling to beat the first level. Despite this, most of the children thought the game difficulty was adequate and all of the players thought that the game was fun and would like to play it at home.

12.3 Analysis

The results from the questionnaire will be analysed and grouped into the different Gameflow elements that the questionnaire is based on. Then some differences in the children's background will be discussed and how that affected the results.

12.3.1 Concentration

All the participants agreed that they were concentrated while playing the game. This corresponds to the observations that were made, everyone seemed to be focused on the game.

12.3.2 Goal Clarity

About half of the participants agreed that it was easy to understand the goal of the game. This number is lower than desired and might be due to the fact that the game had no overall purpose or story. This might also be due to the difficulty of the game being a little too challenging; there should have been some easier introductory levels to explain the game mechanics better.

12.3.3 Feedback

The feedback from the game to the players was appropriate - they knew when they did something wrong, and when they did something correct. The graphics and audio was very well received with the participants. There was an issue where two of the first participants played without sound, but this was shortly fixed.

12.3.4 Challenge

The challenge for the game was a little higher than desired. Around half of the participants found the challenge to be adequate, but over a third of the participants found the game too challenging. This was in part due to the first level being a little too difficult. There should have been more introductory levels for the players to learn the game mechanics. It is also hard to predict the skill level of children around that age, as some found the game to be very easy and struggled very little. It might also come down to time constraints. The participants were given about 15 minutes to play the game and almost everyone said that they got better at the game after playing for a while. There were also some issues with input lag that occurred for which negatively affected the difficulty of the game. This affected a total of six participants, where five of them reported that the game was too difficult. Both of the players that disagreed that their skill improved, was negatively affected by the lag.

12.3.5 Autonomy

About two thirds of the participants said that they felt in control of the player, and that it was possible to recover from errors. These numbers were negatively affected by the input lag that was experience on one of the laptops. Five of the six participants affected by the lag disagreed that they felt in control of the players movements. There was also a spot in the first level that was easy to get trapped in. The spot was hard to get out of and the and should be considered as a flaw in the level design.

12.3.6 Immersion

Most of the participants got immersed in the game and agreed that time went by quickly when playing the game. The majority also primarily focused on the game, and became unaware of their surroundings. Some participants however were interested in the other pair that was playing and got a little distracted by trying to see how far the other players had come. Overall the game seemed to grab the attention of the participants.

12.3.7 Social Interaction

The social interaction between the participants was good. They all agreed that it was important to communicate and cooperate when playing the game. The participants were eager to communicate, and provide directions and help to each other. One of the pairs that performed the best had a very direct form of communication by issuing commands to each other, like ”stand there” and ”grab that”. There were

also a lot of pointing to the screen to tell the other player where to stand or place a block and similar. Being a collaborative game it was not very surprising to see that most did not see the game as a competition. However, some participants said that they did feel like they were competing against others to get the furthest. This was also made apparent when some participants would ask us how far some other players had made it, and some even stayed to see how the other participants did.

12.3.8 Game Enjoyment

All of the participants strongly agreed that they enjoyed playing the game. Though it would seem that the game is very fun to play, it is important to look at it from a broader perspective. This was a special occasion, and the children may not get to play that much video games in the after school program. Everyone seemed excited to even get a chance to play the game. The participants would also like to play the game at home, and some even asked where the game could be bought.

12.3.9 Gender Differences

There were a total of eleven males and eight female participants. Table 12.2 shows the differences in responses between the male and female participants. Only the agreement is shown for simplicity, but because of the binary nature of the questionnaire, the disagreement is the opposite of the agreement. Only the questions that had a significant difference in the responses have been included (a difference greater than 10 percentage points).

Unfortunately, half of the female participants played on the laptop that had a considerable amount of input lag, which might skew the results some. Overall the males seemed to respond with a better understanding of the game, they agreed more that it was easy to understand the goal and easy to understand when they did something correct or wrong. The game sounds appealed more to the female participants, but two male participants played without sound, which skews the results a little. There was also a tendency where more of the female participants responded that the game was to hard compared to the males. All but two participants felt that their skill improved when playing the game, both of these participants were female and both experienced lag while playing, which negatively affected their responses. Similarly the majority of the male participants agreed that they felt in control of the movements, and could recover from errors, the majority of the females disagreed. This might also be attributed to lag. The sense of cooperation also seems to have been affected by lag as the two female participants that disagreed that the cooperation was good, both experienced lag and said that it was hard to cooperate. More of the males seemed to feel that time went by fast compared to the females, and were also more competitive with others to reach the furthest.

Table 12.2: The results showing the gender differences in the responses

		Male	**Female**
ID	**Question**	**Agree**	**Agree**
	Goal Clarity		
Q2	It was easy to understand the goal of the game	63,64%	37,50%
	Feedback		
Q3	It is easy to understand when I do something wrong in the game	90,91%	62,50%
Q4	It is easy to understand when I do something good in the game	100,00%	75,00%
Q6	I liked the sounds of the game	63,64%	100,00%
	Challenge		
Q7	I get bored when playing the game	18,18%	0,00%
Q8	The game was too easy	27,27%	0,00%
Q10	The game was too hard	27,27%	50,00%
Q11	My skill gradually improved when playing the game	100,00%	75,00%
	Autonomy		
Q12	I felt in control of the movements of the player	90,91%	37,50%
Q13	I could recover from errors I made in the game	90,91%	37,50%
	Immersion		
Q14	Time goes by fast when plaing the game	90,91%	75,00%
	Social Interaction		
Q16	I was competing with others to reach the furthest	36,36%	25,00%
Q19	I cooperated well with the other player	100,00%	75,00%
Q20	The other player cooperated well with me	100,00%	75,00%

12.3.10 Game Experience

Out of the total 19 participants, eleven responded that they had a lot of prior game experience, seven said that they had some game experience, and one said that they had no prior game experience. Table 12.3 shows the effect of game experience on the results. About half of the participants with a lot of gaming experience were male (55%male), and about half of the participants with some gaming experience were male (57% male). This means that the gender differences will not influence the results to a great degree. Because there was only one participant with no prior game experience their results will either be agree (100%) or disagree (0%), which is important to keep in mind when comparing the results.

Understandably the participants with more game experience found it easier

to understand the goal of the game. This could come from their familiarity with side scrollers. The participants with a lot of game experience seemed to enjoy the sounds of the game less, however both of the participants that played without sound stated that they had a lot of game experience. When disregarding those two participants the results are a lot more similar with a difference of only two percentage points. The challenge seemed to be more adequate for the participants with some game experience compared to the ones with a lot of game experience. One reason for this might be that their inexperience with video games might make them less able to judge the challenge of the game, and would incline them to judge the challenge as adequate. The differences in the results of Question 11, are attributed to lag, as the only participants that disagreed were affected by lag. The participants with a lot of game experience got more immersed while playing, and felt that time went by fast while playing, whereas only about half of the ones with some game experience agreed. This could indicate that the children with more game experience find it easier to get immersed in the game. Participants with a lot of game experience responded that they were competing with others more than the ones with only some game experience. This could be because they were more used to the competitive nature of video games. The with regards to the cooperation is attributed to lag, as the two participants that disagreed that the cooperation was good between the players said that it was hard to cooperate due to the lag.

12.3.11 Control Differences

The participants played two different figures, the hand and the character and there were a few differences in the responses between the two. ten players controlled the character and nine players controlled the hand. Table 12.4 shows the difference in the responses between the players.

Most of the results were similar, and this might be because of the pairwise testing. Since the players played in pairs of two the players had a similar game experience, and the pars seemed to respond in a similar fashion. Because the participants were asked simultaneously they might even have influenced each other with their choices. There were some minor differences however; the players that controlled the character found it easier to understand the goal of the game. This could be attributed to the fact that it was the character that had to reach the goal in each level - maybe the hand also should have had a goal to reach. The character was probably also more familiar to the participants as the concept of running around and collecting coins is a common video game concept. An interesting thing to note is that the participants that controlled the hand judged the difficulty as more adequate and less hard or easy. This could indicate that the difficulty for the character was a little to challenging for the children, and it might be a result of the hand not really being in any risk of dying.

12.4 Summary

The questionnaire provided a lot of information about the children's experience with the video game prototype. With regards to GameFlow, the game successfully grabbed the attention of the players and everyone agreed that they were concentrated while playing. The goal clarity was not as good and could be better explained through a story and with easier introductory levels. The feedback from the game to the players was appropriate and they had a good grasp on when they did something correct or wrong. The majority also enjoyed the graphics and sound of the game. Concerning challenge, the children experienced the challenge a little more difficult than desired. The challenge could have been eased by including more introductory levels, and letting the children play for longer. Overall the autonomy of the game was decent and the majority of the participants felt in control of the game. However, the autonomy, challenge, and to a degree social interaction were all affected by input lag on one of the laptops. Most of the participants got immersed in the game, with the children not focusing on their environment and having a feeling of time going by quickly. The social interaction between the participants was good and they all agreed that it was important to communicate and cooperate. All of the children also enjoyed the game and would like to play it in their homes.

Some differences in the responses could also be seen when considering the gender and prior game experience of the children, and also what player they were controlling. The male participants seemed to have a better understanding of the game and found the challenge to be easier compared to the female participants. The males also got more immersed in the game and felt that time went by quickly. Participants with more game experience found it easier to understand the goal of the game and were more easily immersed in the game. Game experience also seemed to have an effect on competitiveness, where the players with more game experience were more competitively inclined. The results from the participants that controlled the hand and the ones that controlled the character were very similar due to the pairwise testing. However, a difference was that the participants controlling the hand found it harder to understand the goal of the game, but thought the challenge was more adequate compared to the participants controlling the character.

Table 12.3: The results showing effect of game experience on the responses

	Game Experience	**A lot**	**Some**	**None**
ID	**Question**	**Agree**	**Agree**	**Agree**
	Goal Clarity			
Q2	It was easy to understand the goal of the game	63,64%	42,86%	0,00%
	Feedback			
Q3	It is easy to understand when I do something wrong in the game	63,64%	100,00%	100,00%
Q6	I liked the sounds of the game	72,73%	85,71%	100,00%
	Challenge			
Q8	The game was too easy	18,18%	14,29%	0,00%
Q9	The game was adequately difficult	45,45%	71,43%	100,00%
Q10	The game was too hard	36,36%	28,57%	100,00%
Q11	My skill gradually improved when playing the game	100,00%	71,43%	100,00%
	Immersion			
Q14	Time goes by fast when plaing the game	100,00%	57,14%	100,00%
Q15	I become unaware of my surroundings when playing the game	81,82%	100,00%	100,00%
	Social Interaction			
Q16	I was competing with others to reach the furthest	36,36%	14,29%	100,00%
Q19	I cooperated well with the other player	81,82%	100,00%	100,00%
Q20	The other player cooperated well with me	81,82%	100,00%	100,00%

Table 12.4: The results showing the difference in responses between the players

	Character/Hand	**Character**	**Hand**
ID	**Question**	**Agree**	**Agree**
	Goal Clarity		
Q2	It was easy to understand the goal of the game	60,00%	44,44%
	Challenge		
Q8	The game was too easy	20,00%	11,11%
Q9	The game was adequately difficult	50,00%	66,67%
Q10	The game was too hard	40,00%	33,33%

PART V

Evaluation and Conclusion

This part evaluates the research done in this project, as well as providing a conclusion of the thesis, and offers insight into possible areas for further work.

Chapter 13

Evaluation

This chapter will evaluate the research done in this project, as well as an evaluation of the game prototype, an evaluation of the technology used, and an evaluation of the project as a whole.

13.1 Evaluation of Research and Results

The research goals will be revisited and the research questions will be answered through information gathered through the literature study, the experience from developing the game prototype and the results gathered from the play testing and questionnaire.

13.1.1 Research Goal 1: Collaboration

The first research goal was all about the collaboration and social interaction between players and was defined with the accompanied research questions as:

RG1: The purpose of this project is to create a collaborative game prototype which promotes collaboration and social interaction from the point of view of a game designer in the context of video game design.

RQ1.1: What game mechanics are suitable for stimulating collaboration between players?

RQ1.2: How can the game be designed to encourage social interaction between players?

From the game prototype (details in Chapter 7) and from the experiment (details in Chapter 11) it was clear that having the two players play different roles with different functions had a big impact on the collaboration between the players. With the

players having different functions a dependency was created between the players as they could not do everything by themselves. The function of having the character being able to pick up items that the hand could then place in the world created a bond between the players where they could help each other out. Dangerous elements like spikes and enemies that threatened the character were another way to create a bond between the character and the hand - the hand could protect the character from danger by blocking spikes with items and luring the enemies away with bait. This created more social interaction between the players where they would request or offer help to the other player. The puzzle aspect of the game also encouraged social interaction, as the players would think out loud and give directions to each other in order to solve the puzzles. A particularly interesting game element were the different switches that controlled coloured boxes and moving platforms. The players would try out the different switches, which was one of the key elements to solving the puzzles. The players would often discuss which switches were the correct ones to activate to solve the puzzle.

13.1.2 Research Goal 2: Children's Experience

The second research goal focused on the children's experience with the game prototype and how they would react to a collaborative game - it was defined with two research questions as follows:

RG2: The purpose of this project is to see how children experience a collaborative game where the focus is on collaboration and social interaction from point of view of the children in the context of playing video games.

RQ2.1: Is a collaborative game that encourages social interaction engaging for children?

RQ2.2: How does the children's gender or game experience affect their experience of collaborative game with focus on collaboration and social interaction?

The results from the questionnaire were analysed in Chapter 12, and shows that the game prototype were engaging for the children that participated in the play testing. Everyone saw the importance in collaboration and communication for progressing in the game. The questionnaire was created with Sweetser's [27] GameFlow model in mind for better review of the game prototype. The results from the GameFlow analysis shows that the game prototype was good at capturing the concentration of the children, the feedback from the game was good, the children got easily immersed in the game, and the social interaction was good. The GameFlow analysis also uncovered that the game prototype's areas for improvement were in goal clarity, challenge, and autonomy.

With regards to the gender and game experience of the children, a few differences were noticed. The males found the game to be less challenging and showed a greater understanding of the game compared to the females by understanding the goal of the game more easily, and recognizing when they did things correct or wrong in the game. The male participants also seemed to get more immersed in the game by experiencing time going by fast, and found the game less challenging to a greater degree than the female participants. The males were also more competitively inclined despite the collaborative nature of the game. There were similar results for the children's prior game experience. Children with more game experience found it easier to understand the goal of the game, were more competitively inclined, and got more immersed in the game. However the participants with only some prior game experience found the game to be more adequately challenging.

13.1.3 Research Goal 3: Game Development

The third and final research goal was intended to explore the various technologies available for an independent video game developer, described with research questions as:

RG3: The purpose of this study is to investigate what tools are available to create video games from the point of view of an independent developer in the context of video game design.

RQ3.1 What tools are available for developing video games as an independent developer.

RQ3.2 Can proprietary controllers made for game consoles be used to develop games for the PC?

There are many technologies that support the independent video game developer in creating video games, as discribed in Chapter 6, with the most important being the game engine. Game engines lessen the burden of the game developer by handling common video game functionality like rendering, physics, lighting, and networking. Two of the most popular game engines are Unreal Engine 4 and Unity3D 5 - these game engines have big online communities that can provide help and information. The entry level for these game engines are free so any aspiring game developer can test their skills in video game development without monetary risk. There are also free software tools available like IDEs for managing and writing source code, and version control systems for keeping track of the games history and for preventing loss and corruption of code. Game assets such as sound and graphics are also available online with both free and paid options, there are even possibilities for hiring a graphic or sound designer online. The assets for the game prototype in this project come from free online resources, details in Section 7.1.

From the experience of developing the game prototype, using proprietary controllers made for consoles on the PC is not a straightforward process. With some of the controllers not providing official APIs it can be difficult to get the controllers to work in the desired environment. This was experienced while trying to get the PlayStation Move controller to work with Unreal Engine 4, more details in Section 9.3. The Move controller was abandoned because of complications when integrating it in the game prototype.

13.2 Evaluation of the Game Prototype

The game prototype was a success at stimulating collaboration and encouraging social interaction and all of the children enjoyed the gameplay. The idea of the players having two different roles was well received by the children and created a unique dynamic that that made the players depend on each other. The other elements were also good for enhancing the gameplay and making it a game that the children easily got immersed in.

However, there are also some room for improvement as made apparent by the results of the questionnaire. The challenge of the game was not consistent and was a little too difficult for the children. The goal of the game was not that apparent either. The game should have had some more introductory levels to teach the players the basics, which could have helped teach the players the mechanics of the game better and made the goal more apparent. There could also have been a story to the game, but this was omitted due to time constraints. The level design could also have been a little better and more adjusted to the skill level of the children, especially the first level. There was a spot in the first level where the players would get stuck and had a hard time getting out of.

The game is very open and could support a multitude of new gameplay elements. There are some enhancements that could have been made to make the hand more active like having the hand being in danger at times and the character had to rescue them - the way it is now the stress level for the hand is low compared to the character. For example there could have been some ghosts chasing the hand, and only the character could scare them away. There are currently also only two different items that the hand can grab and place in the world, other possible ideas for items include an anvil that can be dropped on enemies to kill them, ladders, suspension bridges, or ropes for the character to get around the world easier, or maybe a vehicle for the player to ride or fly in to cross big gaps or float across lava.

Children was the primary target audience for the game, but the gameplay elements can be combined to create more difficult levels that are challenging even for experienced adult gamers by leaving less room for error and punishing mistakes

more. The graphics and audio could also be changed to change its look and feel. For example the game can be made more gloomy and dark to appeal to an older audience that enjoys horror style video games.

13.3 Evaluation of Technology

Unreal Engine 4 was decently easy to get into in the beginning. The gameplay framework was logical, and having the C++ code classes inherit from classes in the framework made it easy to start creating gameplay elements. The only problem was that a big part of the online tutorials for Unreal Engine 4 included or was entirely based on Blueprints. Blueprints a visual scripting language that can be used together with or in place of C++ classes. In the beginning, Blueprints seemed like a useless addition for an experienced programmer, as everything could be done using C++, and was considered a tool for inexperienced programmers. Even though Blueprints are useful for inexperienced programmers it can also be used by experienced programmers to create quick prototypes of features, and can also be used to change attributes of classes to create different variants.

There were some initial problems getting the PlayStation Move controller connected to the computer, but it was eventually achieved through an unofficial API. A plugin for Unreal Engine that was based on this API was discovered and tested working with the Move controller early on in the project and the technology seemed promising. However, when the time came to integrate the move controller to the game, there was no way of accession the raw positional data of the move controller on the screen, which made it hard to implement without much complications. More time should have been invested into trying to get the Move controller working, but since the plugin was tested and looked functional in the beginning of the project, it was assumed to provide the needed functionality.

13.4 Evaluation of Project

Working on a big project like this was a challenging and educational experience. To work on this project alone was both freeing in the sense of having the sole responsibility, but it was also a little frightening to face this challenge alone. The weekly development cycle was a good way of conducting the development because it broke down the project into intermediate chunks that could be completed on a week by week basis. The weekly meeting with the advisor was invaluable for feedback and direction on what to do next, and for keeping up the motivation for the project.

To work with children was a new and enlightening experience that posed some challenges. It was hard to judge the skill level of the children, which in turn made

it hard to create appropriately challenging levels. The levels were made with an increasing difficulty, but it could have started a little lower in the challenge spectrum. It was impressing to see how well the children were able to communicate and work together to solve the levels, and they were all genuinely excited to be able to play. All of the children were well behaved and no problems occurred during testing, other that technical issues. The children would also come with unexpected feedback - some of the children asked the name of the game and also the name of the character, or what planet they were on. One child said that they were competing with the spider when asked if they felt that they competed with the other players. Thoughts and statements like that made the experience more fun and rewarding, and provoked some interesting new perspectives.

Chapter 14

Conclusion

This thesis started with the intention of creating a collaborative video game for children that encourages social interaction. The goals for the research was to create a game prototype, test this prototype with children, observe and analyse the results from the testing, and to study the technologies and process involved in video game development. This was achieved through a literature study; the description of the design and implementation of the game prototype; a presentation of observations, results and analysis of the play testing; and finally an evaluation to answer the research questions for the thesis.

14.1 Findings

The literature study creates a foundation of knowledge on which the game prototype and method for evaluation of the play test is built. This is done by describing the current state of video game technology and concepts, exploring interaction device like the PlayStation Move, and aspects of immersion, like social interaction and the importance of graphics and audio, and discovering GameFlow, a model for describing game enjoyment. In the literature study, important technologies for video game development was discovered, such as the game engine Unreal Engine 4 that the game prototype was created with.

The design of the game prototype describes the concepts and game elements that stimulates collaboration and encourages social interaction. Having two players with different roles created a dependency between them where they had to work together in order to succeed. The puzzle elements worked well in encouraging social interaction among the players, where the children would think out loud and inquire and instruct the other player on how to solve the puzzles. There was a unique relationship between the character and the hand where the character would pick up items for the hand, which in turn could be used to aid the character and progress through the levels.

An experiment was performed in the form of a play test at Buvik School's after school activity program, where 19 pupils participated. Through observations from the play test and analysis of the results from the questionnaire, the game prototype was found successful at stimulating collaboration and encouraging social interaction between the children. There was a lot of communication between the children while playing, in order to progress through the levels of the game. All of the children responded that it was important to communicate and collaborate while playing the game. The game grabbed the attention of the children and made them concentrate on the game while playing, making them immersed in the game world.

14.2 Limitations

Although the game prototype and play testing was a success, there are some limitations to this research. The game prototype was developed with Unreal Engine 4 and is limited to use of the underlying architecture of that game engine. Additionally, there was only one person developing the game prototype, which led to some challenges in the creative aspect of development, complications in the testing for a game with two players, and difficulties in predicting the skill levels of the children. However, this was remedied by weekly meetings with the advisor, which provided a second point of view and guidance. Moreover, the experiment was done with only 19 participants, which could impact the reliability of the results. The questions were asked to players in pairs, so the responses could potentially be affected by the other player, or by bias in the interpretation from the examiners. The play test was also a special occasion for the children and might influence their excitement and enjoyment of the game. Furthermore, this project was also restricted on time and resources, which limited the scope and opportunities for research and experimentation.

14.3 Relevance of Thesis

In this thesis, the literature study is helpful for aspiring and qualified video game creators, especially independent developers, to broaden their knowledge of video game technologies, interaction devices, aspects of video game immersion, and the resources available for video game development. Additionally, the game prototype and its design serves as a source for inspiration and demonstration for a video game that stimulates collaboration and encourages social interaction that can be used in research and video game development. Analysis of the results and observations from the play testing provides empirical evidence and quantitative data that solidifies the claims made in this thesis, which can be used in further research.

Chapter 15

Further Work

This final chapter brings the thesis to a close by providing a list of areas for further work.

- Improving the game prototype by addressing the issues that were revealed after the testing. As made apparent by the analysis of the results, the game was lacking in some areas, for instance in providing an adequate challenge for the children and making the goal of the game clear.

- Adding new items and features to the game prototype. The game prototype could be expanded with new features and game elements to make it into a complete game. Adding common video game features like menus, a story, and a narrative are also a great way to enhance the game experience.

- Creating custom graphics and audio for the game. The game prototype uses free online assets for the graphics and audio, but other options are available. By working with graphical and sound designers the game experienced could be even more enhanced and custom tailored.

- Trying out various interaction devices to see how the dynamics of the game and social interaction are affected. The game prototype was originally going to use a PlayStation Move controller for one of the players; it would have been interesting to see how different interaction devices would affect the children's experience with the game.

- Conducting further research on how collaboration in video games can be used to improve children's collaboration and interaction skills in other areas, such as working in group projects. This would be an exciting area for research, and could potentially be used by schools to aid in teaching the children how to work together.

- Exploring various platforms like mobile gaming. Game engines make it easy to develop a game for multiple platforms simultaneously, and creating a mobile version of the game could provide a different experience and allow for new and innovative ways of interacting with the game elements.
- Conducting more experiments with larger groups of children. More experimentation could be done to confirm or challenge the findings in this thesis, and to discover new ways in which to stimulate collaboration.
- Exploring other target audiences, for example adults, and conducting experiments. This thesis has focused on collaboration between children, but the game could also be modified to appeal to adults, as collaboration and social interaction can be a powerful motivator of video games.
- Release and distribute the game. The game is in a prototype stage right now, but it could be further developed into a full fledged game and released to the public. This study mentions ways of distributing games digitally and the process of releasing and distributing a game would be an valuable experience.

Appendices

Appendix A

Norwegian Questionnaire

	Alder	___ år			
	Kjønn	gutt ☐	jente ☐		
	Har du spilt mye spill før?	aldri ☐	litt ☐	mye ☐	
ID	Spørsmål	Uenig	Litt enig	Litt enig	Enig
	Konsentrasjon				
1	Jeg konsentrerer meg når jeg spiller spillet	☐	☐	☐	☐
	Klare mål				
2	Det var lett å forstå hva jeg skal gjøre i spillet	☐	☐	☐	☐
	Informasjon				
3	Det er lett å vite når jeg gjør noe galt i spillet	☐	☐	☐	☐
4	Det er lett å vite når jeg gjør noe bra i spillet	☐	☐	☐	☐
5	Jeg likte hvordan spillet så ut	☐	☐	☐	☐
6	Jeg likte lydene i spillet	☐	☐	☐	☐
	Utfordring				
7	Jeg kjeder meg når jeg spiller spillet	☐	☐	☐	☐
8	Spillet var for lett	☐	☐	☐	☐
9	Spillet var passe vanskelig	☐	☐	☐	☐
10	Spillet var for vanskelig	☐	☐	☐	☐
11	Jeg ble bedre i spillet etterhvert som jeg spilte	☐	☐	☐	☐
	Kontroll				
12	Jeg hadde kontroll over spillerens bevegelser	☐	☐	☐	☐
13	Jeg kunne komme meg videre selv om jeg gjorde noe feil	☐	☐	☐	☐
	Innlevelse				
14	Jeg synes at tiden går fort når jeg spiller spillet	☐	☐	☐	☐
15	Jeg tenkte ikke på det som var rundt meg når jeg spilte spillet	☐	☐	☐	☐
	Sosial interaksjon				
16	Jeg konkurerte med andre om å komme lengst	☐	☐	☐	☐
17	Det var viktig å snakke sammen for å klare spillet	☐	☐	☐	☐
18	Det er viktig å samarbeide for å klare spillet	☐	☐	☐	☐
19	Jeg samarbeidet godt med den andre spilleren	☐	☐	☐	☐
20	Den andre spilleren samarbeidet med meg	☐	☐	☐	☐
	Underholdning				
21	Det var gøy å spille	☐	☐	☐	☐
22	Jeg kunne tenkt meg å spille dette spillet hjemme	☐	☐	☐	☐

Appendix B

Statistical Results

Question	Mean	Median	Variance	ST.Dev
Q1	3,6842105263	4	0,2280701754	0,4775669329
Q2	2,5789473684	3	1,701754386	1,3045130839
Q3	3,1578947368	3	1,0292397661	1,014514547
Q4	3,4210526316	4	0,701754386	0,8377078166
Q5	3,9473684211	4	0,0526315789	0,2294157339
Q6	3,2631578947	4	1,0935672515	1,045737659
Q7	1,3684210526	1	0,9122807018	0,9551338659
Q8	1,4736842105	1	1,0409356725	1,0202625508
Q9	2,5789473684	3	1,5906432749	1,2612070706
Q10	2,1052631579	2	1,5438596491	1,2425214884
Q11	3,5263157895	4	0,9298245614	0,9642741111
Q12	3,0526315789	4	1,3859649123	1,1772701102
Q13	2,8421052632	3	1,5847953216	1,2588865404
Q14	3,2631578947	4	1,2046783626	1,0975784084
Q15	3,6842105263	4	0,6725146199	0,8200698872
Q16	1,7894736842	1	1,2865497076	1,1342617456
Q17	3,8947368421	4	0,0994152047	0,3153017676
Q18	3,9473684211	4	0,0526315789	0,2294157339
Q19	3,6842105263	4	0,4502923977	0,6710382982
Q20	3,7894736842	4	0,3976608187	0,6306035353
Q21	4	4	0	0
Q22	4	4	0	0

Bibliography

[1] D. Hovind and K. R. Elstad, "Alternative games - innovations in video games and game technologies," tech. rep., Department of Computer and Information Science - Norwegian University of Science and Technology, 12 2015.

[2] V. R. Basili, "Software modeling and measurement: the goal/question/metric paradigm," 1992.

[3] M. Overmars, "A brief history of computer games." http://www.cs.uu.nl/docs/vakken/b2go/literature/history_of_games.pdf, 01 2012.

[4] Android, "Android history," 11 2015. https://www.android.com/history/, accessed 2015-11-26.

[5] A. Robertson, "'angry birds 2' arrives 6 years and 3 billion downloads after first game," 07 2015. http://www.forbes.com/sites/andyrobertson/2015/07/16/angry-birds-2/, accessed 2015-11-26.

[6] L. Nintendo Co., "Consolidated sales transition by region," 06 2014. https://www.nintendo.co.jp/ir/library/historical_data/pdf/consolidated_sales_e1406.pdf, accessed 2015-12-07.

[7] Nintendo, "Wii u features," 12 2015. http://www.nintendo.com/wiiu/features/, accessed 2015-12-07.

[8] IGN, "Ps4 vs. xbox one vs. wii u comparison chart," 06 2015. http://www.ign.com/wikis/xbox-one/PS4_vs._Xbox_One_vs._Wii_U_Comparison_Chart, accessed 2015-12-07.

[9] Nintendo, "What is amiibo?," 12 2015. http://www.nintendo.com/amiibo/what-is-amiibo, accessed 2015-12-07.

[10] T. Walker, "Xbox one: 'the ultimate all-in-one home entertainment system': Microsoft finally unveils its latest console," 05 2013. http://www.independent.co.uk/life-style/gadgets-and-tech/

news/xbox-one-the-ultimate-all-in-one-home-entertainment-system-microsoft-finally-unveils-its-latest-8626177.html, accessed 2015-12-08.

[11] S. C. E. Inc., "Sony computer entertainment inc. introduces playstation4 (ps4)," 02 2013. http://www.scei.co.jp/corporate/release/130221a_e.html, accessed 2015-12-08.

[12] T. Amini, "All 24 of the ps4's launch games, reviewed," 11 2013. http://kotaku.com/all-24-of-the-ps4s-launch-games-reviewed-1464421841, accessed 2015-12-08.

[13] G. Lehri, "Is the ps move dead?," 01 2015. http://www.gameondaily.com/ps-move-dead/, accessed 2015-12-15.

[14] A. Carlson, "A kinect-less xbox one signals the death of motion gaming," 05 2014. http://www.hardcoregamer.com/2014/05/19/a-kinect-less-xbox-one-signals-the-death-of-motion-gaming/85360/, accessed 2015-12-15.

[15] ScienceBeta, "Wii vs. playstation 3 vs. xbox 360: Home fitness gaming consoles compared," 08 2014. http://sciencebeta.com/wii-vs-playstation-3-vs-xbox-360-home-fitness-gaming-consoles-compared/, accessed 2015-12-15.

[16] Microsoft, "Buying and using digital games on xbox one faq," 12 2015. support.xbox.com/en-US/xbox-one/games/digital-game-faq, accessed 2015-12-08.

[17] S. C. Entertainment, "Playstation store faq's," 12 2015. https://support.us.playstation.com/articles/en_US/KC_Article/PlayStation-Store-FAQ-s, accessed 2015-12-08.

[18] R. Stanton, "Full steam ahead: How valve's platform just gets hotter," 12 2015. support.xbox.com/en-US/xbox-one/games/digital-game-faq, accessed 2015-12-08.

[19] K. Graft, "Stardock reveals impulse, steam market share estimates," 11 2009. http://www.gamasutra.com/php-bin/news_index.php?story=26158, accessed 2015-12-08.

[20] J. Matulef, "Xbox 360 at 10: How xbla changed the industry forever," 11 2015. http://www.eurogamer.net/articles/2015-11-27-xbox-360-at-10-how-xbla-changed-the-industry-forever, accessed 2015-12-08.

[21] W. Watlington, "A short history of indie games, and a look into the future," 07 2015. `http://updownright.com/2015/03/07/a-short-history-of-indie-games-and-a-look-into-the-future/`, accessed 2015-12-08.

[22] A. H. Cummings, "The evolution of game controllers and control schemes and their effect on their games," in *The 17th Annual University of Southampton Multimedia Systems Conference*, vol. 21, 2007.

[23] J. C. Lee, "Hacking the nintendo wii remote," *Pervasive Computing, IEEE*, vol. 7, no. 3, pp. 39–45, 2008.

[24] C. Wingrave, B. Williamson, P. D. Varcholik, J. Rose, A. Miller, E. Charbonneau, J. Bott, and J. J. LaViola Jr, "The wiimote and beyond: Spatially convenient devices for 3d user interfaces," *Computer Graphics and Applications, IEEE*, vol. 30, no. 2, pp. 71–85, 2010.

[25] Microsoft, "Meet kinect for windows," 12 2015. `https://dev.windows.com/en-us/kinect`, accessed 2015-12-15.

[26] Nintendo, "Duck hunt." Game for NES released in 1984 (JP).

[27] P. Sweetser and P. Wyeth, "Gameflow: a model for evaluating player enjoyment in games," *Computers in Entertainment (CIE)*, vol. 3, no. 3, pp. 3–3, 2005.

[28] M. Csikszentmihalyi and M. Csikzentmihaly, *Flow: The psychology of optimal experience*, vol. 41. HarperPerennial New York, 1991.

[29] T. W. Malone, "What makes things fun to learn? heuristics for designing instructional computer games," in *Proceedings of the 3rd ACM SIGSMALL symposium and the first SIGPC symposium on Small systems*, pp. 162–169, ACM, 1980.

[30] Capcom, "Ōkami." Game for Sony Playstation 2 (2006). Ported to Nintendo Wii (2007). Rereleased HD version for Sony Playstation 3 (2012).

[31] A. Donovan, H. Cho, C. Magnifico, and J. H. Lee, "Pretty as a pixel: Issues and challenges in developing a controlled vocabulary for video game visual styles," in *Proceedings of the 13th ACM/IEEE-CS joint conference on Digital libraries*, pp. 413–414, ACM, 2013.

[32] S. Huiberts, *Captivating sound the role of audio for immersion in computer games*. PhD thesis, University of Portsmouth, 2010.

[33] Carlsen Games, "140." Game for PC released 16 Oct, 2013. http://game140.com, accessed 2015-12-08.

[34] Double Fine Productions, "140 recommendation." http://store.steampowered.com/curator/6857089-Double-Fine-Productions/?appid=242820, accessed 2015-12-08.

[35] epicycle, "BlindSide," 2012. http://www.blindsidegame.com/, accessed 2015-12-11.

[36] D. M. Plasencia, "One step beyond virtual reality: connecting past and future developments," *XRDS: Crossroads, The ACM Magazine for Students*, vol. 22, no. 1, pp. 18–23, 2015.

[37] HTC and Valve, "HTC Vive." http://htcvive.com, accessed 2015-12-08.

[38] StressLevelZero, "Hover junkers." http://www.hoverjunkers.com/, accessed 2015-12-08.

[39] StressLevelZero, "Hover junkers - large room multiplayer (video)." https://www.youtube.com/watch?v=Ai4MCmZz28o, accessed 2015-12-08.

[40] N. Lazzaro, "Why we play games: Four keys to more emotion without story," 2004. http://www.xeodesign.com/whyweplaygames/xeodesign_whyweplaygames.pdf, accessed 2016-06-09.

[41] A. I. Wang, "The wear out effect of a game-based student response system," *Computers & Education*, vol. 82, pp. 217–227, 2015.

[42] J. Ward, "What is a game engine?," 04 2008. http://www.gamecareerguide.com/features/529/what_is_a_game_.php, accessed 2016-06-06.

[43] M. Enger, "Game engines: How do they work?," 06 2013. http://www.giantbomb.com/profile/michaelenger/blog/game-engines-how-do-they-work/101529/, accessed 2016-06-06.

[44] W. Malik, "Top 5 3d game engines for indie game developers," 02 2016. http://www.useoftechnology.com/top-5-3d-game-engines-indie-game-developers/, accessed 2016-06-06.

[45] E. Games, "Unreal engine features," 06 2016. https://www.unrealengine.com/unreal-engine-4, accessed 2016-06-06.

[46] D. Bratcher, "Unity games sweep app store awards," 02 2013. http://blogs.unity3d.com/2013/02/19/unity-games-sweep-app-store-awards/, accessed 2016-06-06.

[47] U. Technologies, "A feature-rich and highly flexible editor," 06 2016. http://unity3d.com/unity/editor, accessed 2016-06-06.

[48] U. Technologies, "Creating and using scripts," 06 2016. http://docs.unity3d.com/Manual/CreatingAndUsingScripts.html, accessed 2016-06-06.

[49] U. Technologies, "Get unity," 06 2016. http://unity3d.com/get-unity, accessed 2016-06-06.

[50] J. Papadopoulos, "Crytek open sources cryengine, available for download to everyone," 05 2016. http://www.dsogaming.com/news/crytek-open-sources-cryengine-available-download-everyone/, accessed 2016-06-07.

[51] I. Fabrik, "Licensing options," 06 2016. http://www.heroengine.com/heroengine/licensing-options/, accessed 2016-06-07.

[52] Y. Games, "Gamemaker:studio," 06 2016. http://www.yoyogames.com/, accessed 2016-06-07.

[53] W. Corporation, "Wave engine," 06 2016. https://waveengine.net/, accessed 2016-06-07.

[54] U. Staff, "Grand theft auto iv," 04 2008. http://www.1up.com/previews/grand-theft-auto-iv_3, accessed 2016-06-07.

[55] R. Makuzawa, "Former luminous engine lead talks transition from final fantasy versus xiii to final fantasy xv," 12 2015. http://www.novacrystallis.com/2015/12/former-luminous-engine-lead-talks-transition-from-final-fantasy-versus-xiii-to-final-fantasy-xv/, accessed 2016-06-07.

[56] T. Bayer, "Assassin's creed 2: Engine detailed," 05 2009. http://www.pcgameshardware.com/aid,684874/Assassins-Creed-2-Engine-detailed/News/, accessed 2016-06-07.

[57] I. DB, "Game engines," 06 2016. http://www.indiedb.com/engines?sort=ranktoday-asc, accessed 2016-06-07.

[58] PixelProspector, "Indie resources," 06 2016. http://www.pixelprospector.com/indie-resources, accessed 2016-06-07.

[59] Git, "Getting started - about version control," 06 2016. https://git-scm.com/book/en/v2/Getting-Started-About-Version-Control, accessed 2016-06-07.

[60] itch Corp, "Create a custom page to instantly distribute or sell your indie games," 06 2016. https://itch.io/developers, accessed 2016-06-07.

[61] K. Land, "Platformer pack redux," 2016. http://www.kenney.nl/assets/platformer-pack-redux, accessed 2016-06-13.

[62] T. Carty, "Space drifter," 2015. http://www.flashkit.com/loops/Ambient/Ambient/space-drifter-150305121007.html, accessed 2016-06-13.

[63] Sonnis.com, "The license – #gameaudiogdc bundle," 2016. http://www.sonniss.com/gdc-bundle-license/, accessed 2016-06-13.

[64] JewelBeat.com, "Free sound effects," 2016. http://www.jewelbeat.com/free/free-sound-effects-1.php, accessed 2016-06-13.

[65] E. Games, "Unreal engine 4 documentation," 06 2016. https://docs.unrealengine.com/latest/INT/, accessed 2016-06-07.

[66] E. Games, "Gameplay framework quick reference," 06 2016. https://docs.unrealengine.com/latest/INT/Gameplay/Framework/QuickReference/, accessed 2016-06-07.

[67] F.-L. Fu, R.-C. Su, and S.-C. Yu, "Egameflow: A scale to measure learners' enjoyment of e-learning games," *Computers & Education*, vol. 52, no. 1, pp. 101–112, 2009.