UNIVERSITY OF TARTU
Institute of Computer Science
Software Engineering Curriculum

Daniel Nael

Virtual Tour Game

Master's Thesis (30 ECTS)

Supervisor: Ulrich Norbisrath, PhD

Tartu 2022

Virtual Tour Game

Abstract:

This thesis gives an overview of the development of a web-based studying game called *Virtual Tour Game*. It was created for the University of Tartu's course *Evolution of Video Games.* It offers a way for students to combine entertainment with learning through a virtual tour of a video game museum. One of the main goals was to create a fully automatic system that would assess the student and synchronize the results from the game with the Moodle learning environment. This thesis also features the development of the next iteration of the studying game. The purpose was to upgrade the game with virtual reality support. The web and virtual reality versions of the game were designed regarding the general principles in the field of game design. The game design and implementation of both versions of the game are described. Playtesting was done on users to discover issues, evaluate the usability of the game and compare which version of the game was more beneficial. Regarding the positive feedback from the test results, the game will be developed further in the future.

Keywords:

Video Game, Educational Game, Studying Game, Virtual Reality, VR, Game Design

CERCS: P170 Computer science, numerical analysis, systems, control

Virtuaaltuuri mäng

Lühikokkuvõte:

Käesolev lõputöö annab ülevaate võrgupõhise õppemängu „Virtual Tour Game“ kohta. See loodi Tartu Ülikooli kursuse „Videomängude Arengulugu“ jaoks. Mäng pakub tudengitele võimalust kombineerida omavahel meelelahutus ja õppimine, kogedes virtuaaltuuri videomängude muuseumis. Üks peamistest eesmärkidest oli luua täisautomaatne süsteem, mis hindaks tudengit ja sünkroniseeriks õppetulemused mängust Moodle’i õppekeskkonnaga. Käesolev lõputöö sisaldab ülevaadet õppemängu järgmisest iteratsioonist. Eesmärk oli luua antud mäng virutaalreaalsuse toega. Mõlemad versioonid mängust disainiti järgides üldisi põhimõtteid mängudisaini valdkonnast. Lõputöös kirjeldatakse ka mängu disainimist ja implementeerimist. Mängu testiti kasutajate peal, et leida vigu, hinnata mängu kasutatavust ja võrrelda milline versioon mängust oli kasulikum. Arvestades testimisest saadud positiivset tagasisidet, arendatakse mängu tulevikus edasi.

Võtmesõnad:

Videomäng, Õppemäng, Virtuaalreaalsus, VR, Mängudisain

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine (automaatjuhtimis-teooria)

Table of Contents

1 Introduction

Video games have become quite popular as their market value worldwide has been increasing every year and is expected to grow[1]. People around the world spend a lot of time every day playing games for entertainment. Video games do not have to be just for amusement. Alternatively, they can have educational purposes too. Young children learn through playing, which can be possible for teenagers and adults.

This master's thesis aimed to create a video game titled *Virtual Tour Game* to offer the students of the University of Tartu's course Evolution of Video Games a possibility to learn through playing. Furthermore, the goal was that the game should automatically give points for the course grade while the player is playing it directly from their web browser.

Virtual Tour Game is a story-based adventure-type studying game that lets its players explore the virtual LVLup![2] museum located in Tallinn. The player must explore the museum, watch videos, read books, collect various items and talk with non-playable characters in the museum and answer their questions to gain points for the course. The intention was to make the experience as immersive as possible, which led to implementing the game's second version in *virtual reality* (VR), which has become trendy nowadays and could bring a whole new level of mesmerising. Both versions of the game pursue the main criterion of game design. A story module was created for this thesis to playtest the game and compare the virtual reality version to the web-based version in terms of usability, immersion, and what benefits do they add to studying.

Chapter 2 features the main background for understanding the design and implementation of the first version of *Virtual Tour Game*. It gives the definition of a video game, introduces the educational part of it, and features some similar experiences to *Virtual Tour Game*.

The focus of chapter 3 lies on the first version of *Virtual Tour Game*. There is a detailed description about its design, implementation, and testing results.

Chapter 4 is concentrated on the background of virtual reality and how it can connect to video games and education in general. Several notable experiences are brought out that helped form the design and implementation of *Virtual Tour Game* in virtual reality.

---

[1] https://www.statista.com/topics/868/video-games/#editorialPicks

[2] https://lvlup.ee/et/home/

Chapter 5, therefore, analyses the same subjects for the second version of the game that are discussed in chapter 3 with the addition of the comparison of the two versions of the game.

The Glossary in Appendix I defines the terms used in this thesis. The installing and running guides for both versions of the game are in Appendix II. The source code and asset files are in the repository (Appendix III).

2 Background

The aim of this chapter is to define the concepts video game and virtual tour to understand the main work of this thesis. It is done by discussing the key concepts of these topics with examples of similar content featured in *Virtual Tour Game*. Furthermore, it is relevant to apprehend how educational games differ from regular video games and why we need them.

2.1 Defining a Video Game

A video game is a form of entertainment that uses different technologies from computer science and arts to offer meaningful interactive experiences to its players. Firstly, it consists of *game mechanics* that are the main rules and features of a game. Secondly, a *story* is a particular order of the events that happen in the game. Thirdly, *aesthetics* interprets the appearance and the experience of the game. Finally, *technology* that makes every other element of a game mentioned before functional. The aesthetics, mechanics, and story are more visible to the player because it does not matter what kind of technology is used for the game. [1]

There is a wide variety of video game genres[3], and each one offers distinctive characteristics of its *gameplay*. For example, in adventure games, the player must solve puzzles (Illustration 1) and explore, but in real-time strategy games, the player must control armies to gather resources and destroy their enemies (Illustration 2). Despite this, it is possible that a video game can feature elements and ideas from two or more different genres [3].

[3] https://www.idtech.com/blog/different-types-of-video-game-genres

Illustration 1. *Firewatch*[4]

Illustration 2. *Age of Empires IV*[5]

As mentioned before, video games are primarily for entertainment purposes. However, they can improve various skills of a player or give new knowledge [4]. They can feature the intent to educate the players, therefore called studying or educational video games.

[4] http://www.firewatchgame.com/

[5] https://www.ageofempires.com/games/age-of-empires-iv/#game-gameplay

2.2 Education in Video Games

Video games can make learning more efficient [5]. Various video games primarily focus on education. This can be done by offering challenges to improve specific skills of a player (e.g., critical thinking skills) or teaching new concepts [6].

All kinds of e-learning content became more critical during the COVID-19 pandemic [7]. Schools had to switch to online learning. When keeping a social distance is important, it is beneficial to have experiences and medium to access from home. That is the main reason why the course *Evolution of Video Games* in University of Tartu was created, and for that, the video game featured in this thesis was developed [7]. The main design of the game used ideas from different existing video games and virtual tours.

2.3 Inspiration and Similar Works from Virtual Tours

It is a well-known truth that when creating a new kind of product in general, it is always relevant to see what already exists in the same field and what it is possible to learn from it. This idea also applies to video games and, moreover to *Virtual Tour Game*.

When we focus on *Virtual Tour Game*'s title, the keywords "virtual" implies that the game features a virtual tour. There are various sorts of virtual tours like tours featuring videos, 360° or tours using panoramas, tours using still photos and floor plan tours[6] which were researched about for the thesis.

The game of this thesis features a 360° image virtual environment like virtual tours created for places or businesses to let visitors explore the buildings or places before going there physically. For example, the virtual tour of Tallinn University[7] (Illustration 3) and the National Museum of the United States Air Force Virtual Tour[8] (Illustration 4) feature hotspots that show information about specific objects and to move to the next virtual room.

[6] https://www.techopedia.com/definition/4920/virtual-tour

[7] http://virtuaaltuur.tlu.ee/

[8] http://www.nmusafvirtualtour.com/

Illustration 3. Screenshot of the Tallinn University virtual tour

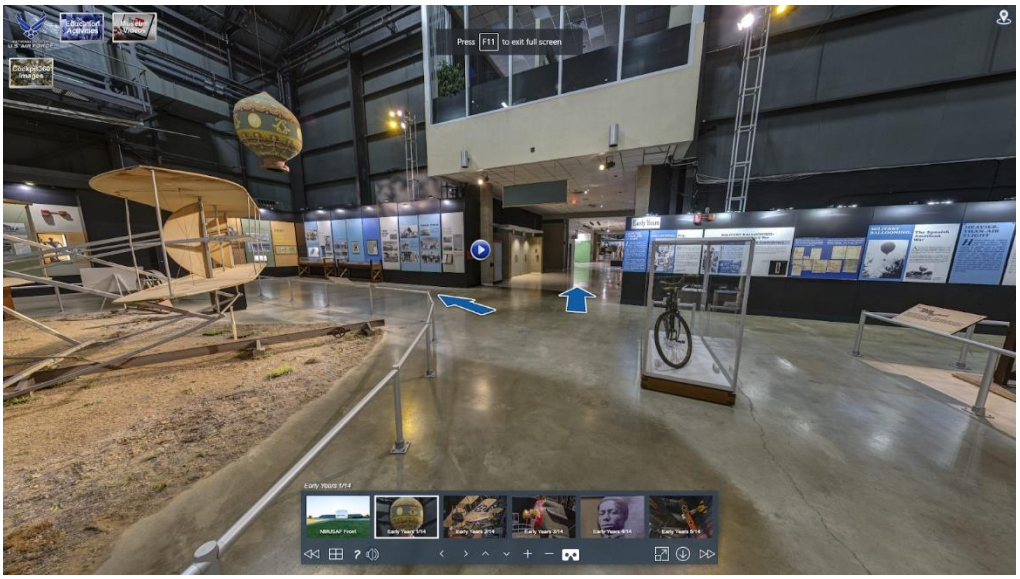

Illustration 4. Screenshot of National Museum of the United States Air Force virtual tour

The Louvre Museum in Paris also has a virtual tour[9] (Illustration 5) that has a map for the user to quickly move to next rooms was also decided to add in *Virtual Tour Game*. This makes it relatively "simple" to rush to the next room.

[9] https://petitegalerie.louvre.fr/visite-virtuelle/saison1/#/petite_galerie_7/

Illustration 5. Screenshot of the Louvre Museum virtual tour

Tallinn University is also similar to the game of this thesis because of the animated screen in the sound studio room of the tour (Illustration 6). *Virtual Tour Game* has several virtual screens that are animated or play videos directly from the YouTube platform.

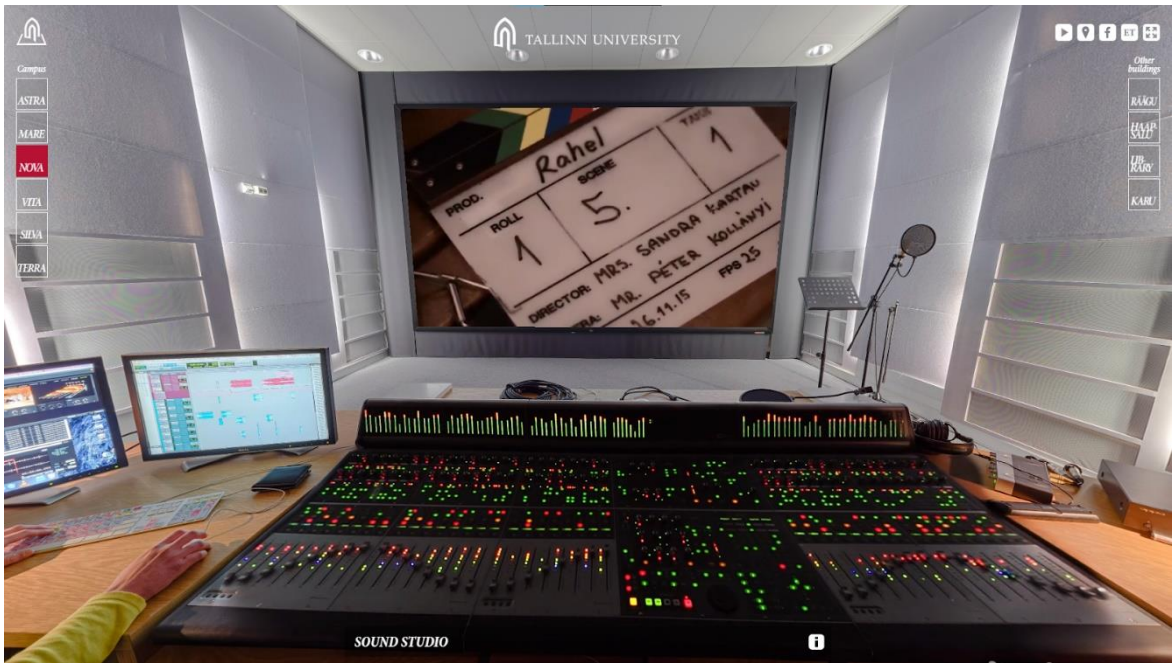

Illustration 6. Sound studio of Tallinn University virtual tour

2.4 Inspiration and Similar Works from Video Games

As mentioned, the "virtual tour" part of *Virtual Tour Game* features various ideas and elements from similar experiences. Different video games also inspired the title's "game" part.

Virtual Tour Game is a mixture between adventure and visual novel games, which means its main gameplay focuses on exploring, collecting items, and talking with different characters. Most notable inspirations and similar games for *Virtual Tour Game* are adventure games like *Firewatch* [10]and *RuneScape*[11], together with a visual novel game titled *Doki Doki Literature Club!*[12]

2.4.1 Firewatch

Firewatch (Illustration 7) is a 3D single-player first-person adventure game that takes place in the wilderness. The player has a mission to survive and experience a mysterious story. The main gameplay features controlling a character through the world, using a walkie-talkie to talk, collect items and solve puzzles. This game is noteworthy because of its dialogue system that feels immersive and has consequences in the game's world.

Illustration 7. Screenshot of the dialogue system of *Firewatch*

However, *Virtual Tour Game* does not offer a virtual tool like a walkie-talkie to communicate with non-playable characters nor a complex dialogue system where what the

[10] https://store.steampowered.com/app/383870/Firewatch/

[11] https://www.runescape.com/community

[12] https://store.steampowered.com/app/698780/Doki_Doki_Literature_Club/

player says impacts the game story. Still, it remains remarkable for the inspiration of the thesis game because it shows how well dialogues with characters can be done. Indeed, the game of this thesis impacts on the game based on what the player says to non-playable characters (NPC), but the story is smaller in terms of content and more linear.

2.4.2 RuneScape

RuneScape (Illustration 8) is a massively multiplayer online[13] adventure game. The game lets the player create their own character. It is the essential element of the world that needs to be continuously upgraded to pick up new adventures in the game.

Illustration 8. Screenshot of *Runescape*

Virtual Tour Game is similar to *RuneScape* because of the dialogue system and item-collecting game mechanic. Both games feature talking with non-playable characters. The non-playable characters ask the player to collect items or speak with other similar entities in the game's world. Though *RuneScape* and *Virtual Tour Game* look different, these mentioned game mechanics are almost identical in terms of idea and execution. Therefore, the "adventure" aspect of both games is similar.

[13] https://en.wikipedia.org/wiki/Massively_multiplayer_online_game

2.4.3 Doki Doki Literature Club!

Doki Doki Literature Club! (Illustration 9) is a paragon of the visual novel games that has also its impact on the *Virtual Tour Game*. This is since the game of this thesis features similar layout and functionality of the dialogue system where the non-playable character the player talks to is shown widely on screen, and with that, their text and options on what to say to the character. This is common to most visual novel type games.

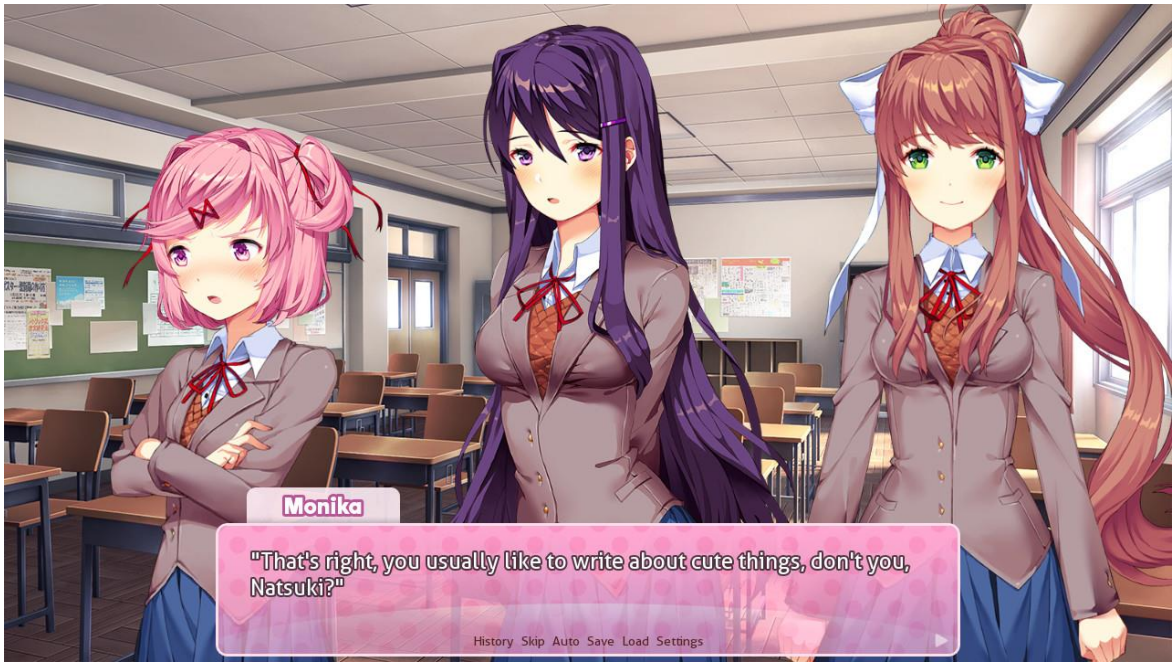

Illustration 9. Screenshot of *Doki Doki Literature Club!*[14]

The game mentioned in this chapter was a big inspiration when making *Virtual Tour Game*. Though, as mentioned before, they look different and do not feature virtual tours. However, their game mechanics are similar and helped form *Virtual Tour Game*.

[14] https://store.steampowered.com/app/698780/Doki_Doki_Literature_Club/.

3 Virtual Tour Game

This chapter aims to overview the first version of *Virtual Tour Game*. The overview includes how it was designed, and implemented, what technologies were used for it and how it works for real users.

3.1 Background

Virtual Tour Game is a 3D virtual tour text-based visual novel style adventure game. It takes place in the previous rooms (Pärnu mnt 154) of the virtual Interactive Video Game Museum LVLup![15] located in Tallinn. As stated in chapter 2.4 it is a mixture between adventure and a visual novel game where the player's main actions are exploring, collecting items, and talking with different characters of the game's world.

The main goal of the *Virtual Tour Game* (Illustration 10) was to offer an online study environment for the course Evolution of Video Games at the University of Tartu that differs from traditional ways of accessing learning materials and tests. For example, some courses at the University of Tartu highly rely on Moodle and its test system to assess a student's knowledge. In the *Evolution of Video Games* course, the main idea was to make learning more "fun" for the student by giving them the ability to play a game and at the same time learn about the fundamental concepts of video games and their history.

Illustration 10. Screenshot of *Virtual Tour Game*

[15] https://lvlup.ee/et/home/

3.2 Design

The design of Virtual Tour Game takes general principles of game design into account. The basic requirements for the game were:

1) The player must be able to tour the LVLup! museum virtually to explore it.
2) The player must be able to see study videos in the game.
3) The player must be able to read study materials in the game.
4) The player must be able to speak with non-playable characters through a dialogue system.
5) The player must be able to get points to assess the knowledge of the player.

These basic requirements were designed via an *iterative design* process. According to Zimmerman and Salen [2] a game is best designed when prototyping and playtesting are done regularly in the entire process. Therefore, it gives the opportunity to make changes if some feature of the game does not work as well as expected or if some new idea works better. This is the core meaning and workflow of the iterative design process and is important because it is not possible to theoretically assess if the design of the game works or not [2].

As mentioned before, the game of this thesis follows the main principles of game design. In chapter 2.1, it was mentioned one of the core elements of a game is called game mechanics, which are the rules and functionalities of a game. Different mechanics were designed for the *Virtual Tour Game*.

3.2.1 Rooms

The virtual museum consists of various rooms. There are a total of 14 rooms. Each room has its theme, and the player can move between them from the beginning of the game. The player “stands” in a fixed room position and cannot move continuously inside a room. This is also the case in similar virtual tour applications featured in chapter 2.3.

3.2.2 Movement

The primary way of movement in *Virtual Tour Game* is using hotspots (Illustration 11. The move hotspot) like in other similar applications. This allows the player to move from one room to another by clicking the hotspot with a mouse to move to the desired location. Additionally, when hovering into a hotspot, it shows the name of the room the hotspot takes the player after a mouse click.

Illustration 11. The move hotspot

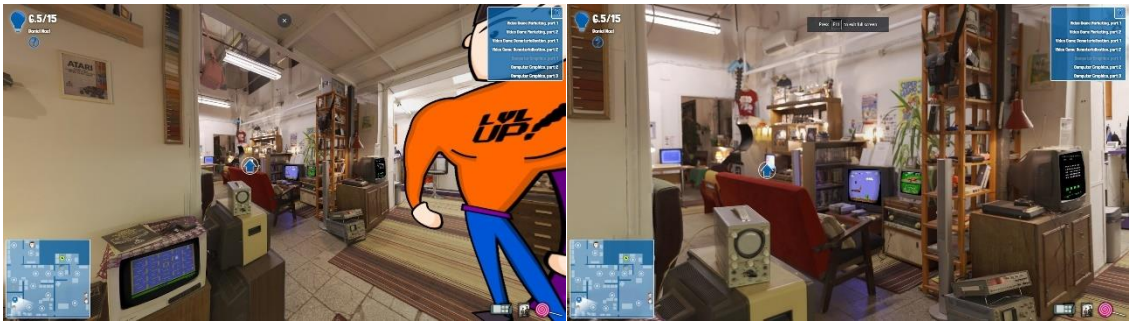

Illustration 12. Zooming in the panoramic image

The player could look around in a room by dragging the camera using a mouse. This is the most common way virtual tours manage the look of the user's camera. It is also possible to zoom (Illustration 12) into the panoramic image using the mouse scroll wheel to clearly see objects in the pictures.

3.2.3 Map

As mentioned in chapter 2.3 a virtual tour in the Louvre Museum in Paris used a map to show in which room they were located. The author designed and implemented a similar system for the game of this thesis (Illustration 13) but with the difference of showing the direction where the player looks. The map system in the game updates the looking visualisation continuously in real-time, whereas in the Louvre, it does not when the user rotates the view in the world.

Illustration 13. The map of the museum

Also, in the game, non-playable characters (in the game called guests of the museum) and locations that the player can interact with are shown.

3.2.4 Guests of the Museum

In the rooms of the virtual museum, there are 7 non-playable characters to which the player can talk to. They are called guests because, as their title says they are "visiting" the virtual museum.

When the player clicks on them, then the dialogue system is started, and the player can have a conversation with them. In the conversation mode, an image of a guest to who the player is talking is shown together with the text of the guest and choices of the player (Illustration 14).

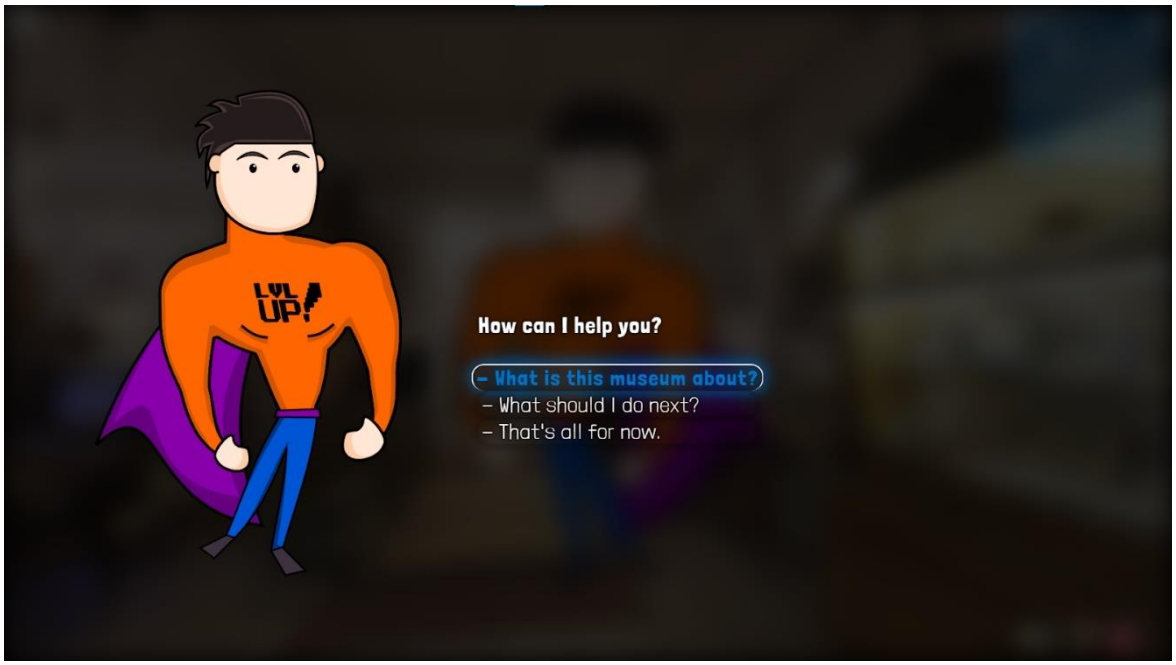

Illustration 14. The dialogue system of the *Virtual Tour Game*

The player's knowledge regarding specific topics is assessed throughout the game using the conversation between a guest and the player. A guest can ask questions from the player, which gives points if answered correctly (Illustration 15). All questions can be answered once. Therefore, the player cannot "cheat" by answering the same questions multiple times. The questions or text of a guest that can give points when answered are marked differently so that the player understands what is assessed in conversations.

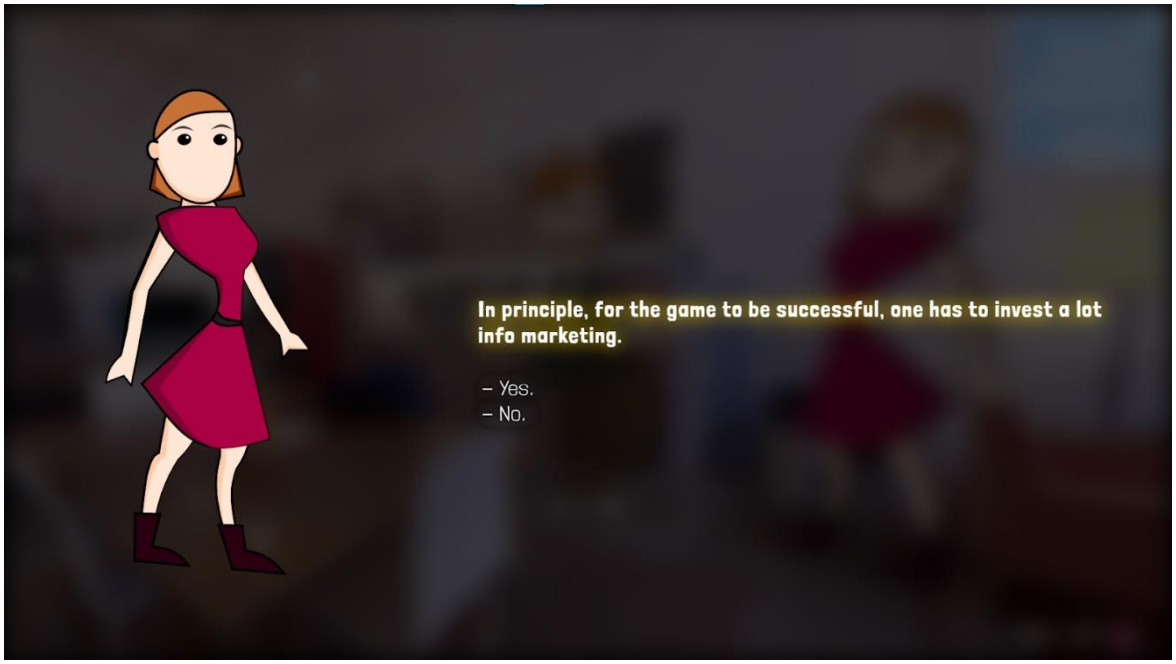

Illustration 15. The dialogue that can give points when answered correctly

Even though conversations are good for evaluating if a player knows the answer, they can also be good for introducing new knowledge to the player. However, the conversations in the current state of the game are more for checking the player's knowledge. There are interactive screens and books where the player can find material about different topics.

3.2.5 Interactive Screens

Illustration 16. Three interactive screens in the museum

The LVLup! Museum featured many television and computer screens. Therefore, as mentioned in chapter 2.3, it was possible to animate the screens to show content to the player (Illustration 16). There are two types of screens: a) animated screens that play a part of a video in loop, b) a video that also has controls (the player can pause and play it), sound and can be made full screen. The second option was used to show learning videos that were made for the course. This was an option for the player to study without reading any books.

3.2.6 Books

The other option for the player to study is to read the books. Throughout the museum, there are hotspots (Illustration 17) where the player can collect books which then appear in their book inventory. From the book inventory, it is possible to open a book (Illustration 17) any time in the game, excluding the time when they are speaking with a guest.

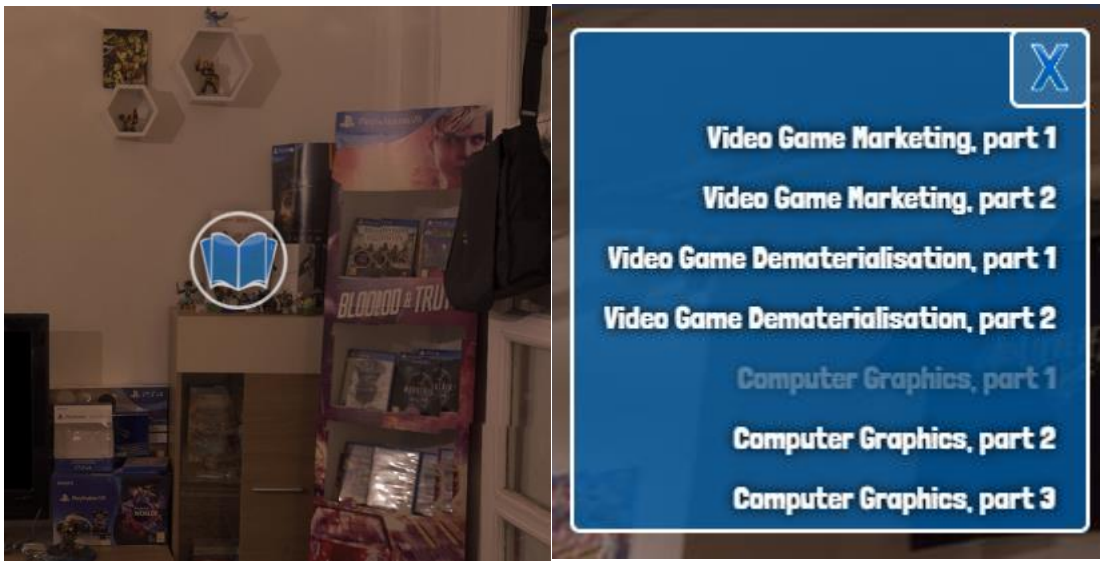

Illustration 17. Book hotspot (left) and inventory of books (right)

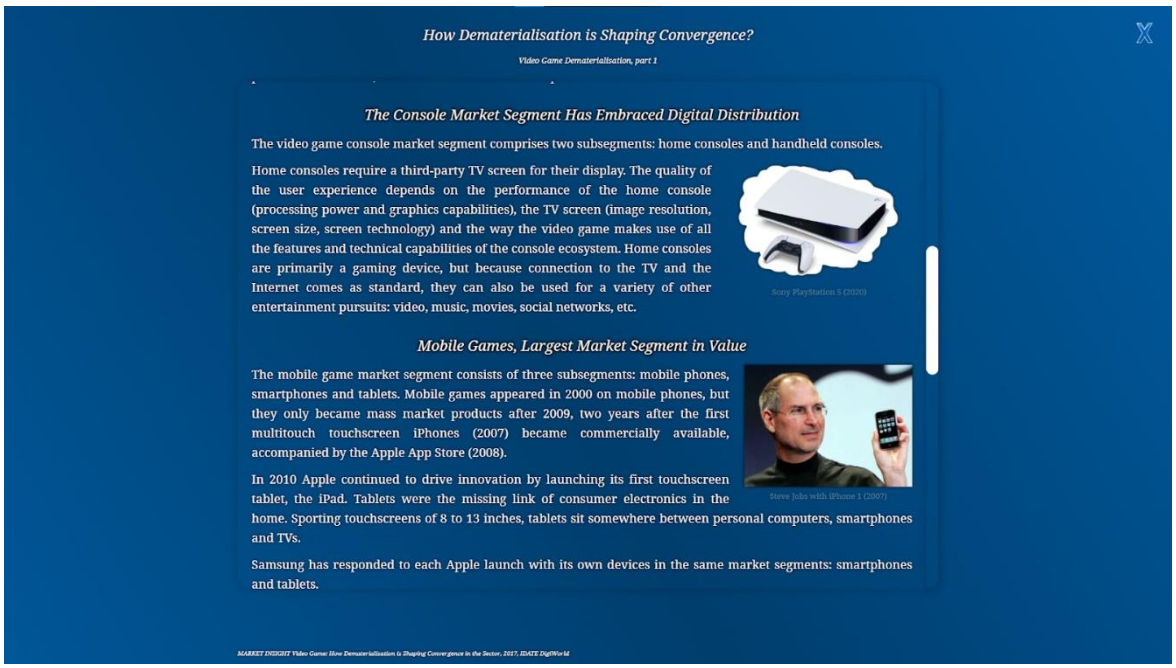

Illustration 18. Content of a book

There is a total of 7 books that can be found and read (Illustration 18) in the game. Every book has different topics connected to video games. Raimond-Hendrik Tunnel created most of the content for the books. The author stylized the graphical user interface part of a book.

3.2.7 Inventory

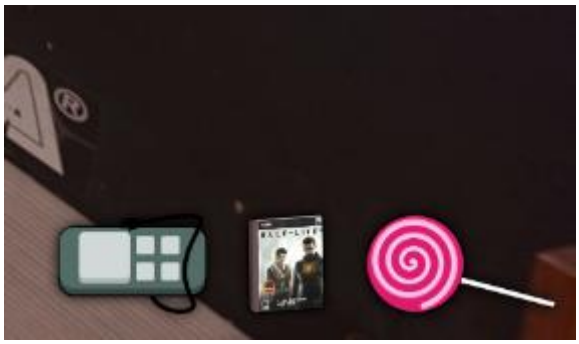

Illustration 19. Items in the inventory

Books are not the only things that can be collected in the museum. Due to the reason *Virtual Tour Game* being an adventure game, it features collectable items (Illustration 19) to give the player meaningful gameplay by finding them and bringing them to guests.

The majority of the features, together with the inventory system, needed a save and load system. Therefore, the author designed the game so that the player can exit the application, and after they open it again, all the items are still in the inventory they collected. As the game is a web application, the implementation of progress management required the use of an online database and implementing the connection between the game and it.

3.3 Implementation

The development of *Virtual Tour Game* started in the year 2020. Before implementing the primary systems, the author of this thesis did additional research to find out how to make a web-based video game connected with the Moodle environment. Chapter 3.3.1 focuses on the foundational technologies of the game. Chapters 3.3.2 - 3.3.4 give an overview of the most relevant implemented systems.

3.3.1 Technologies Used

The fact that the player had the requirement to play the game using a web browser made it clear that different web technologies had to be used. Therefore, the author used Hypertext

Markup Language[16] (HTML), JavaScript[17], and Cascading Style Sheets[18] (CSS) to create the game.

To fulfil the demand of virtually touring a museum also required an implementation. The decision was to find a JavaScript tool that already featured a setup for virtual tour handling (looking around in a panoramic image and changing those images via links, e.g., entering a different virtual room) to devote more design and development time to other systems and game design. Therefore, the tool had to be versatile and customizable. Finally, it was vital that the connection between Moodle and the game could be made so that all the game points the player earned went directly to Moodle.

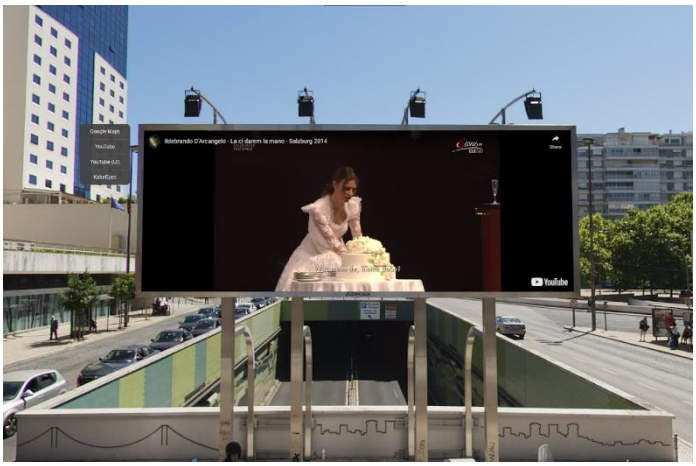

Illustration 20. Interactive screen of Marzipano

The author found different applications to make the game with. Most notable were H5P Virtual Tour[19] and Marzipano[20]. Both featured the opportunity to create a virtual tour and also add interactive content in it. At first, H5P seemed to be the most reasonable platform to pick as it was already integrated with Moodle. It is possible to add text, images, sound, videos, single-choice sets, interactive summaries, and connections to other scenes (rooms). However, the author could not find a way to develop further functional requirements because the demos[21] of Marzipano (Illustration 20) showed that it was feasible to create interactive content (e.g., interactive screens, videos). The author chose Marzipano to build the game with.

With Marzipano it is possible to show any kind of HTML content in hotspots. This does include the content specifically created for the hotspot or embedded content from different

[16] https://www.techopedia.com/definition/1892/hypertext-markup-language-html

[17] https://www.techopedia.com/definition/3929/javascript-js

[18] https://www.techopedia.com/definition/26268/cascading-style-sheet-css

[19] https://h5p.org/virtual-tour-360

[20] https://www.marzipano.net/

[21] https://www.marzipano.net/demos.html

web pages like for example, videos from YouTube. This is significant because it gives the opportunity to scale and add or remove content from the game. Although there is no direct integration between Moodle and Marzipano – the author of this thesis had to create it itself.

3.3.2 Moodle Authentication and Automatic Assessment

One of the main challenges in creating the game was making the connection between Moodle and Marzipano. The author used IMS Learning Tools Interoperability[22] (LTI) to authenticate the player from Moodle and send their earned points from the game to Moodle.

The implementation of the LTI connection consists of Tool Provider and Tool Consumer. Tool Consumer is the application that authenticates its users - in this case, Moodle. Tool Provider (*Virtual Tour Game*) uses the authentication of a Tool Consumer (Moodle) to also authenticate the users securely.

Firstly, it requires that Tool Consumer provides an URL that lets users access the Tool Provider. Secondly, to get to know from where the user came to Tool Provider a consumer key is used. Finally, to have secure messages between the two applications, a shared secret is used. Tool Provider needs to have scripts implemented to manage incoming connections from Tool Consumer. There it needs to be assessed if it is a valid LTI launch request which are signed with OAuth 1[23]. If all the required parameters are correct, then a user session can be established.

The author of this thesis implemented all the required steps to *Virtual Tour Game* using already implemented LTI class libraries written in PHP. This made it possible to use the provided *application programming interface* (API) to make the connection between Moodle and the game. The methods of the API automatically saved Moodle's user information to the database the game uses. The main flow of the authentication of the game consists of three steps:

1. The player logs in to Moodle.
2. The player clicks a link that opens *Virtual Tour Game*.
3. The player is authenticated in *Virtual Tour Game* and can start playing the game.

---

[22] https://www.imsglobal.org/recipe-making-lti-1-tool-providers

[23] https://oauth.net/1/

The data generated by LTI for the player was used in other data tables to connect the progress of the players and, for example what dialogues they have already answered.

3.3.3 Dialogue System

The author created the dialogue system of *Virtual Tour Game* using different technologies. Firstly, JavaScript was used for the main logic on how to visualise and connect dialogues in the back end of the application. In the front end, the SweetAlert2 library, together with Animate.css was used to visualise the content of a dialogue. Finally, a MySQL database is used to store all the dialogues.

The dialogue is represented by a single class. Every dialogue contains an id, URL to an image which is shown, title, text, answers to dialogue, sub dialogue references, points, and a boolean that reflects if the dialogue is gradable or not. The id is to identify and connect all dialogues. The URL shows which image to use to show what character is talking to the player.
Additionally, the title shows the name of the NPC. The text value holds the text the NPC says to the player. Answers to dialogue represent the answers of a dialogue that the player can choose between when talking to an NPC.

All content of the dialogues is fetched from the database at the start of the game. It is saved to a single dictionary type variable (Illustration 21). Every NPC and object in the game has one or many dialogues. Additionally, starting dialogue which shows which dialogue is loaded at first when the player clicks on an NPC or an object.

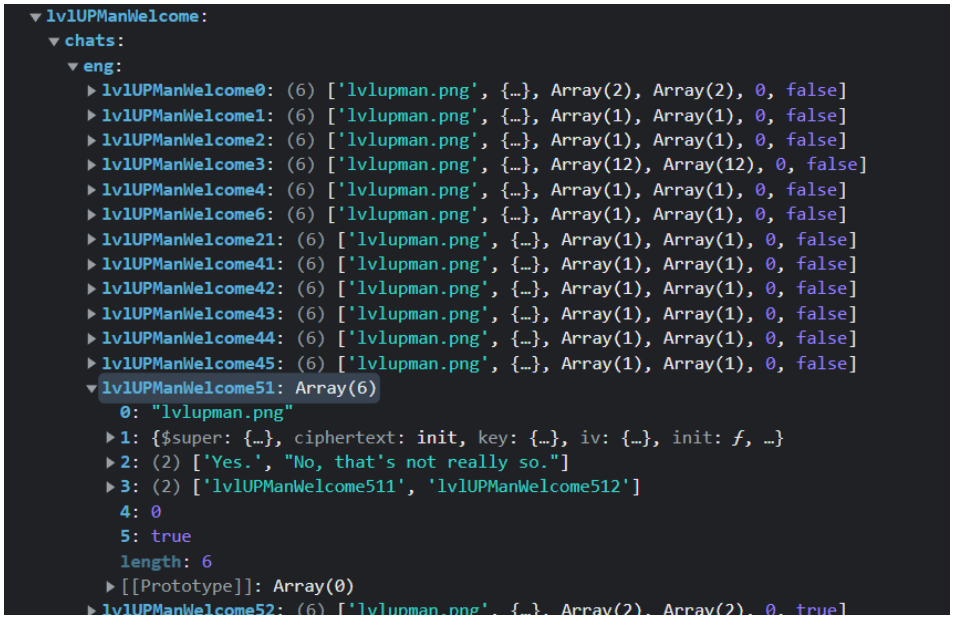

Illustration 21. Example of the dialogues dictionary object

Different constraints exist for dialogues. These constraints are for keeping track of what dialogues have already been answered and if certain actions are done in the game. For example, a dialogue features a constraint that checks if the player has collected a specific item. If they have collected it, then the dialogue will be shown. Otherwise, it is not shown.

Illustration 22. Example of constraints

As mentioned in chapter 3.2.4 the game features dialogues that give points to the player. Every dialogue is loaded from the database in the beginning of the game, but the points of the dialogues are not. This is due the reason to make the system more secure and so that the players cannot know which answer is right when inspecting the dialogues object in the web browsers console. When the player chooses an answer, then the dialogue id of what answer

is chosen is sent to the database. In the database it is checked if the specific dialogue gives points. If that is the case, then the points are added and saved to the player in the database and returned to the game (Illustration 23). Therefore, on the client side, it is not possible to see how many points a dialogue gives to the player. On the contrary, the necessary calculations are done server-side in the database.

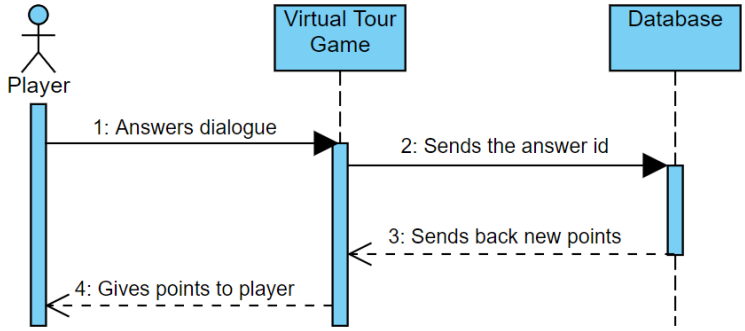

Illustration 23. Sequence diagram of a player getting points

As the majority of the gameplay focuses on the player talking to the NPCs, the dialogue system is one of the most important features of the game. However, dialogues cannot be started without interactive hotspots.

3.3.4 Interactive Hotspots

A hotspot is an element of a virtual tour that is for showing different content, moving between scenes, or having specific interaction, as mentioned in chapter 3.2. The hotspots can be placed everywhere in the 360° view. As mentioned in chapter 3.3.1 the system for visualising hotspots and placing them is implemented in Marzipano. Using the API, the author was able to create various hotspots and add desired content in them. The game features 4 types of hotspots: hotspots that play animated gifs and videos, hotspots for moving to the next rooms, hotspots that contain NPCs to see them and start talking with them and finally, hotspots that unlock items and books for the player when they click on it.

The interactive functionality and graphics were added to the hotspots by the author. Co-creator Raimond-Hendrik Tunnel also helped add content to hotspots by mostly finding and adding gifs to the interactive screens. Additionally, Raimond, together with Mark created studying materials for the *Evolution of Video Games* course that we also used in the *Virtual Tour Game*. Two modules of learning content, together with dialogues and books, were formed and added to the game to test if the game works on actual students.

3.4 The Testing

Playtesting is important for video games as this tells the developer if the game works and is amusing or not for the player. Additionally, it brings out the bugs and technical errors the game has. The playtesting of *Virtual Tour Game* was initially done by the creators of the game and the adjacent course throughout the development of the game. As mentioned in chapter 3.2 the design of the game followed an iterative design process which meant that after implementing certain designs to the game, they were tested and discussed if they worked for the game. For example, the author changed the dialogue GUI from that he implemented during the Computer Graphics Project course due it being not that usable and consistent with the whole style of the game (Illustration 24).

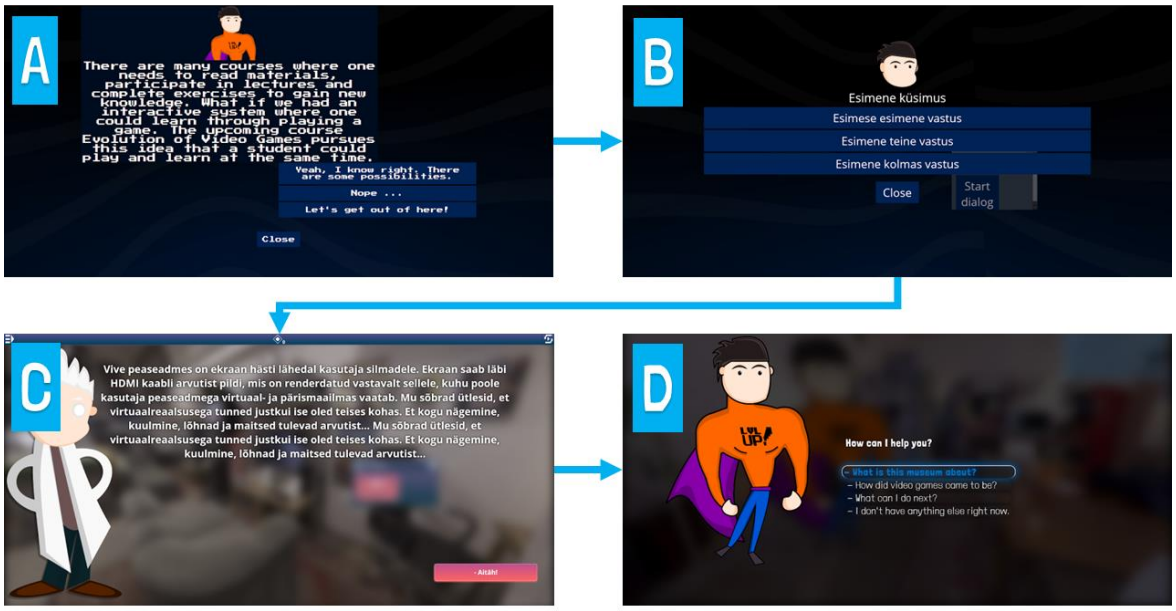

Illustration 24. Dialogue changes in iterations

Mark Muhhin [7] assessed in his master thesis if the *Virtual Tour Game* works on actual students. He collected feedback from the students on how pleased they were with the game and how it benefited their studies. Also, he compared how Moodle test results compared to the results of the game. The author of this thesis made it possible by saving the progress and the history of the player's actions to the database. That gave the opportunity to see how every dialogue in the game was answered by the students.

The results [7] were mainly positive, and the students were more motivated to learn through playing the game compared to doing regular Moodle tests. Most of them liked the idea of studying and playing the game. However, there were also comments about the opposing sides of the game. For example, it was brought out that searching for the objects in the game required a lot of work from the students. This was tedious for some of the students.

Illustration 25. Presenting the *Virtual Tour Game* at *MängudeÖÖ 2021*[24]

The game was also presented at the event *MängudeÖÖ 2021*[24] (Illustration 25), where the visitors could play the game. There the author gave an introduction about what the game is about and then inspected how the visitors play the game. The game received mostly positive feedback, and the visitors were pleased with the experience. They also pointed out some aspects that could be improved. For example, it would be rewarding if the players could also play different games inside the interactive screens of the game.

[24] https://cgvr.cs.ut.ee/games-at-mangudeoo-2021/

4 Virtual Reality

Developing a video game or any other application in virtual reality (VR) has its own unique aspects compared to creating common experiences that do not put the user directly in the virtual world [8]. Therefore, to understand the next step of *Virtual Tour Game* in VR, one must first get to know about the different facets of VR. This is the main reason for this chapter, as chapter 4.1 defines what virtual reality is, and chapter 4.2 gives an overview of the work that is related to the VR version of *Virtual Tour Game.*

4.1 Defining Virtual Reality

According to Jason Jerald [8] virtual reality is a digital environment that is created by a computer so the user can participate in it as if it truly existed in the real world. The goal would be the possibility that the user can be part of it by existing there and interacting with different objects in the virtual world. Nowadays, there are various experiences made in virtual reality that include movies, video games, simulations, etc. For example, one of the most popular video games made for virtual reality is *Beat Sabre*[25], where the player must use two swords to destroy blocks that are moving toward the player. Every block can be destroyed only by swinging the sword to the correct direction and using the same-coloured sword. Therefore, it gives the challenge to the player and requires the use of coordination to make an accurate hit. This game is a good example of how the player can be fully in a virtual world and interact with its objects.

Illustration 26. Screenshot of *Beat Sabre*[25]

[25] https://store.steampowered.com/app/620980/beat_saber/

Developing applications for VR is not a trivial task as it is demanding in terms of creativity and scientific approaches like testing how it works on users by collecting and analysing their feedback and interactions in the experience [8]. Therefore, it does not matter if the system built for an experience is technologically advanced. On the contrary, if it is not intuitive for the user or makes them feel unwell, the experience itself is already negative and needs to be improved. Due to that reason, the author of this thesis decided to take additional time to do research by playing different VR games and experiences to get inspiration for the next version of *Virtual Tour Game*.

4.2 Inspiration and Similar Works

Virtual reality has become quite popular nowadays, and its popularity is expected to grow even more in upcoming years [9]. Therefore, there are, as mentioned in the previous chapter, various experiences created for virtual reality. This chapter focuses on the relevant ones that are similar or helped to design and implement *Virtual Tour Game* in virtual reality.

4.2.1 The Lab

The Lab[26] is a 3D game made exclusively for virtual reality. There the player starts in a lab and can do different activities or explore different worlds. Several smaller games like fighting with robots and being an archer exist in the game. It can be more defined as a demonstration for what is possible to do with virtual reality. The playthrough of this game inspired the author of the game of this thesis. Playing the game can give a virtual reality developer basic knowledge how to design game mechanics in VR. Though *Virtual Tour Game* does not feature complex systems like in *The Lab*. It gave ideas to the author on how to place a graphical user interface and how movement and controls work in VR. The entire museum of *Virtual Tour Game* can

Illustration 27. Promotional image of *The Lab*[26]

[26] https://store.steampowered.com/app/450390/The_Lab/

be considered as a “lab” where the player must move around and interact with NPCs and objects.

4.2.2 The VR Museum of Fine Arts

The VR Museum of Fine Arts[27] (Illustration 28), on the other hand, is a game that has direct similarities with *Virtual Tour Game.* Firstly, the setting is also located in the museum, and it is possible to learn about history just like in the game of this thesis. The museum exhibits pursue an authentic look and feel, which is the same as the game of the thesis.

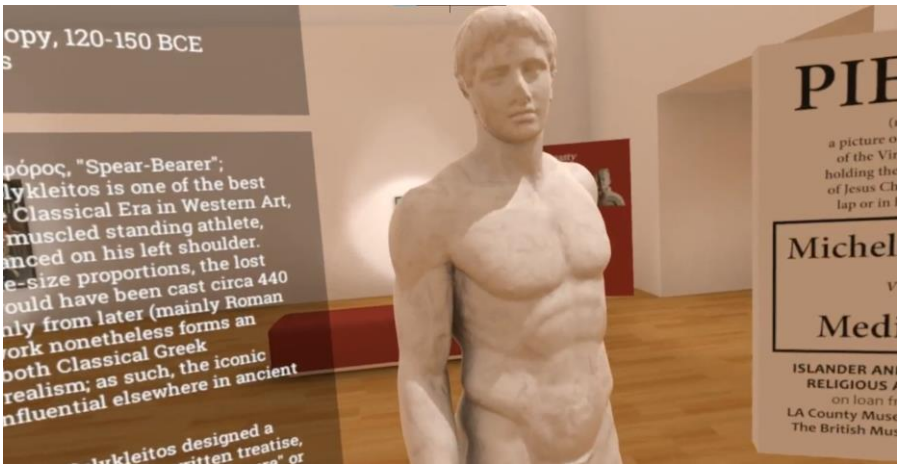

Illustration 28. Screenshot of *The VR Museum of Fine Art*

Illustration 29. The information sheet of the museum.

One of the ideas that the author got from this game was how to display the map and other useful information (Illustration 29) in the game. It used a sheet of information where the

[27] https://store.steampowered.com/app/515020/The_VR_Museum_of_Fine_Art/

mentioned elements were displayed. That inspired *Virtual Tour Game* to also use a GUI element that is like an information sheet.

The author played other games and experiences to get inspiration for *Virtual Tour Game* and to study aspects of virtual reality. As mentioned before, developing a VR experience is not a trivial task and requires effort. This was also the case with the *Virtual Tour Game* VR version.

5 Virtual Tour Game in Virtual Reality

As mentioned before, the next major improvement of *Virtual Tour Game* was to bring it to virtual reality. This chapter gives an overview of the virtual reality version of the *Virtual Tour Game*. In chapter 5.1 the background is given how and why the idea to bring the game to virtual reality was made. Chapter 5.2 discusses the main design points of the game with comparison with the web version, whereas chapter 5.3 talks about the most important parts of the implementation of the game. The final chapter 5.4 focuses on how the game works on actual players by introducing the test results gathered from playtesting the game.

5.1 Background

The idea and reason to start implementing the virtual reality version started after the *MängudeÖÖ 2021* event. There it was brought out that with virtual reality, we could do something more with the game. There was an idea that the immersion delivered by virtual reality by putting the player directly into the virtual world and allowing interaction with it could be more entertaining and, therefore, more beneficial in terms of studying.

The development of the game in virtual reality is relevant. The practicality of the application comes if there should happen in the world that restricts people to stay at home to avoid contacts between them. Therefore, it is necessary to have experiences that can be enjoyed virtually from home. However, the author of this thesis strongly recommends not replacing virtual reality experiences with the real world. Though, this is a subject that is not discussed further in the scope of this thesis. It is more important to focus on the technical and design part of the game.

5.2 Design

The principles of game design applied to the second version of the game also. The author used the iterative design as it was defined in chapter 3.2 again as it is beneficial for the result. Additionally, the principles that were discussed in chapter 4 were relevant while developing. Also, it was important to keep in mind similar experiences that were introduced in chapter 4.2.

Most requirements for the first version of the game that was introduced in chapter 3.2 also applied to the VR version. The goal was to have the basic gameplay mechanics from the first version to be implemented in the second version. Although, some of the features did not make the design list because they were not needed in the VR version to assess if the

whole application is an immersive experience for the player. Mostly those requirements like getting points and adding these to Moodle were needed for the *Evolution of Video Games* course.

5.2.1 Rooms

The author also designed the game to feature the same 14 rooms as in the earlier experience. Naturally, it now had to feature virtual reality support. Although, no significant changes were made in terms of placement of the rooms because there was no need. The only thing that the author changed about the rooms was the fact that the author wished to improve the quality of the panoramic images. Therefore, the author used a different set of pictures he took from the LVLup! museum initially and restitched them with Microsoft Image Composite Editor[28]. The same application was used to make the images for the first game also.

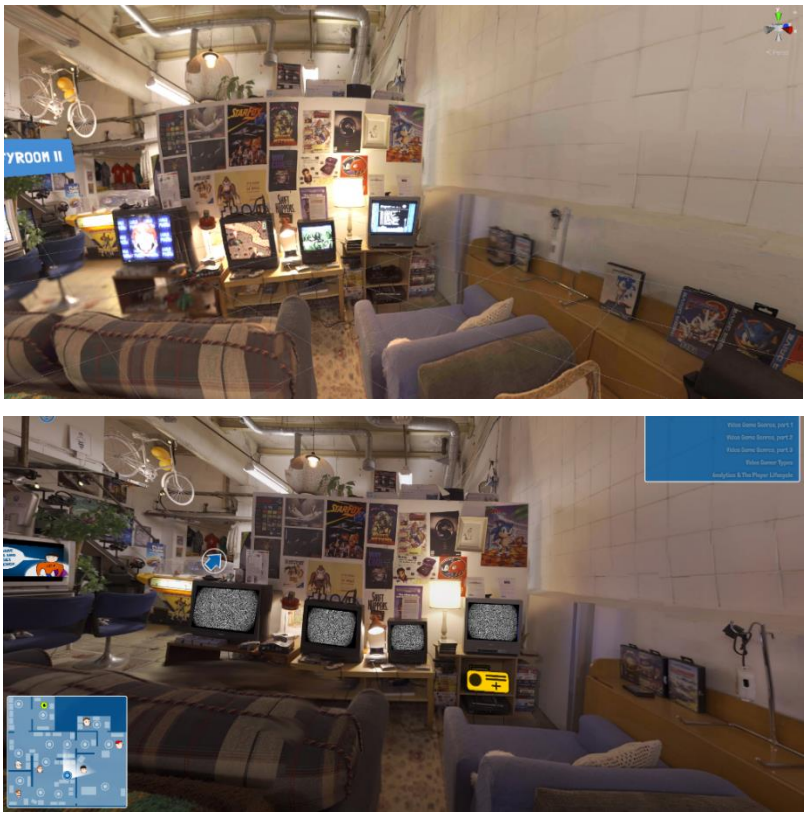

Illustration 30. New (up) an old (down) room comparison

[28] https://www.microsoft.com/en-us/research/project/image-composite-editor/

The author tried to improve the panoramas, but unfortunately, some of them lost quality. For example, Illustration 30 shows that the old image is sharper. Then again, the wall on the right became straighter, and the new lightning could give a better feeling for the player when moving to different rooms.

5.2.2 Movement

The movement of the second version of the game stayed the same regarding changing rooms. The player needs to point the VR controller to the hotspot of a specific room and then press a button to move on (Illustration 31).

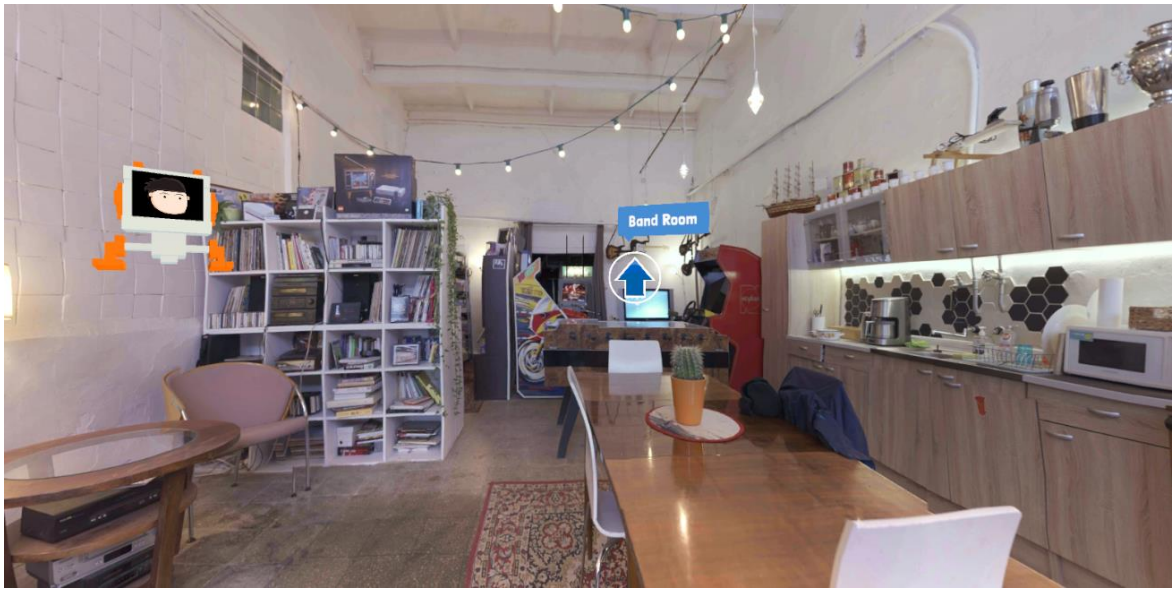

Illustration 31. Move hotspot in second version

Looking and moving around a room, however, is changed entirely. There is no need to drag the camera anymore as the player can just look at everything by rotating the camera with their head.

The author had to implement the zooming of the player's camera to allow them to look at items closely. If the player goes near objects in any direction, then the camera is zoomed in, and the player can see the objects and images zoomed in.

5.2.3 Map

The author implemented the map also in the game so that it also tracks the look view of the player. The location of the map is, however, changed. The map graphical user interface elements are now three-dimensional in the game world, not two-dimensional "stuck" to the

screen. Three-dimensional graphical user interfaces are more accessible in virtual reality [8]. The player can hold the map in their hand and move around similarly to the system mentioned in chapter 4.2.2 (Illustration 32).

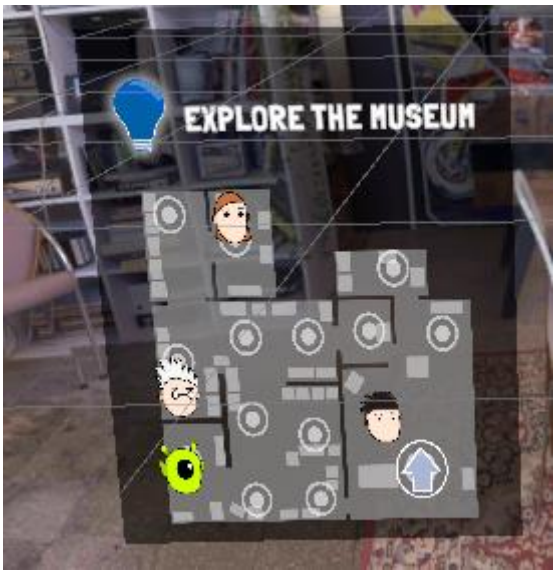

Illustration 32. The map of the museum in second version of the game

The map in this version does not include quick traveling when pressing to a certain. The author did not implement it due to making it necessary for the player to explore the museum and see all the guests.

5.2.4 Guests of the Museum

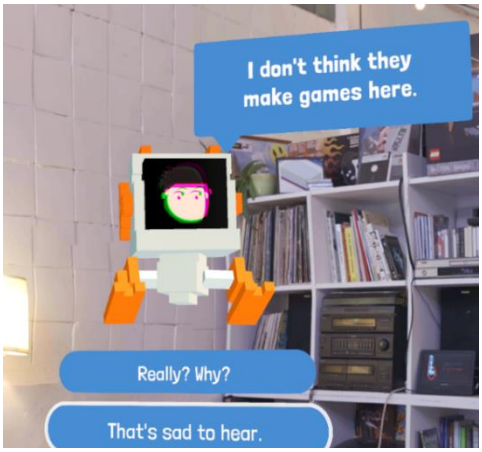

Illustration 33. LVLup! Man Bot

The significant improvement in the VR version of the game can be seen in the museum's guests. They are not anymore static 2D images. Instead, they are fully 3D modelled (by the author using Blender[29] and MagicaVoxel[30]) (Illustration 33) and animated and react with animations to the player's actions.

The dialogue system GUI is also changed (Illustration 33) due to the fact to make it three-dimensional and to be more part of the world. The

[29] https://www.blender.org/

[30] https://ephtracy.github.io/

author also animated its closing, changing, and opening to improve the aesthetics of the game. This is important because it also gives depth to the game and, therefore, could improve the immersion. The author animated the dialogue handling in the first version also. However, because the GUI of dialogue is part of the 3D world now, it could be aesthetically more pleasing. This is kept in mind in chapter 5.4 when discussing the test results.

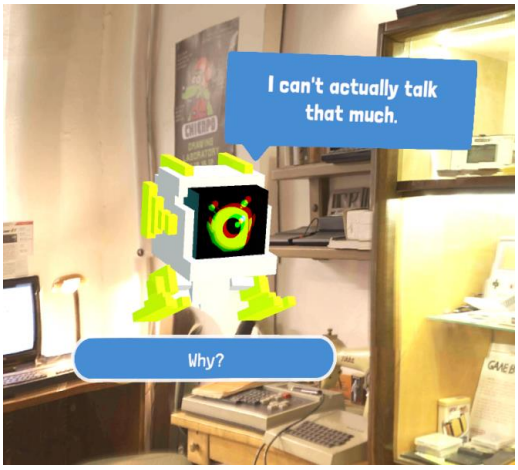

Illustration 34. Ufo Bot

The faces for the new guest models were from the original characters (Illustration 34) of the game to keep the continuity and at the same time bring new aspects to the game, not just only new dialogues of characters.

5.2.5 Dialogue System

The dialogue system in the game is designed similarly to the first game. Initially, the author had a plan to implement a dialogue system where the player had to take more action than just choosing the correct answers that lead to other dialogues through branching. The ludic dialogue system featured in the bachelor's thesis written by Frederik Raud [11] seemed at first relevant for the game. This meant that the dialogues were more intelligent, not only following the linear path that developer had implemented. Instead, be more flexible by choosing dialogues from all pre-written ones and therefore "talking" to the player. However, the author still made the traditional branching dialogue system to focus on offering exactly the story the author wants the player to experience due to the reason of the player studying specific topics.

Similar to the first version of the game, every dialogue has answers that the player could choose that lead to the next dialogue. The choices are shown or not using constraints that are kept in history. For example, if the player has gotten item A, then a constraint is saved and checked later in dialogue choices. When a choice requires that the player has item A, but they do not have it, then that choice is not shown to the player. Otherwise, they can see the choice and choose it.

The author designed and implemented an editor to change dialogues directly in the game's system quickly. With version one, there was no graphical user interface to write the

dialogues. Everything was written using only text, and connections between dialogues required the developer to know dialogue id values. Now it is possible to drag and drop dialogues and make connections between them (Illustration 35).

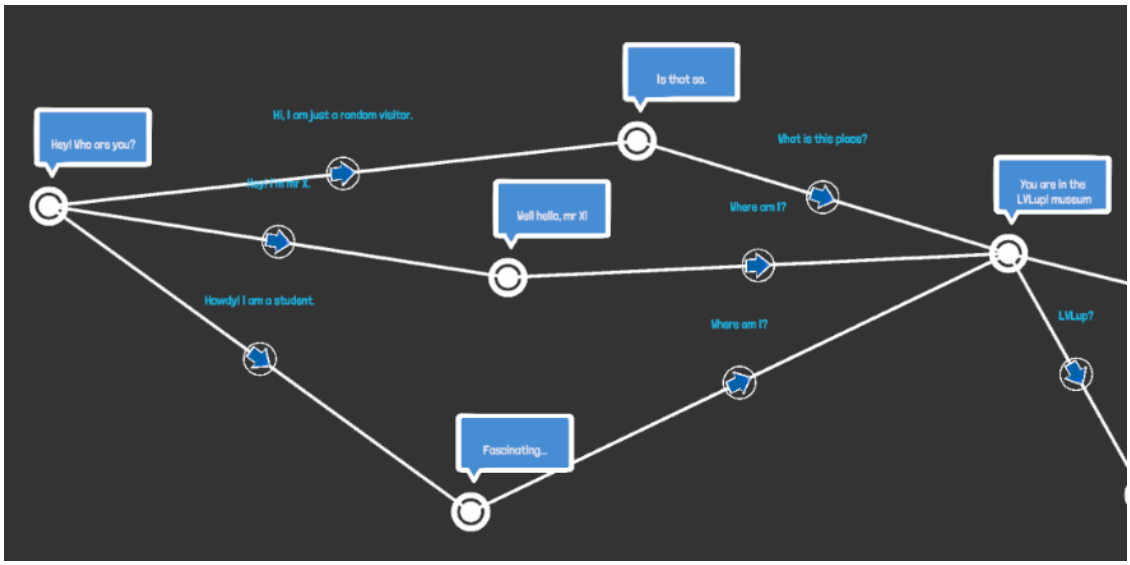

Illustration 35. Real-time screenshot from making a story from dialogues in the game

The player does not get any benefit because of this system – everything remains the same in terms of using the dialogue system. The author, however, can make the dialogues visually and see how everything is connected. Furthermore, the dialogues can be added to specific NPCs by just dragging and dropping them. As the system is built in game, it also is possible with some additional enhancements to use the system for debugging to exactly visualize what dialogue is an NPC currently using. Though it required additional implementation, it made the author's workflow regarding making dialogues more efficient.

5.3 Implementation

The implementation of the VR version turned out somewhat challenging for the author as it required knowledge about different new aspects of the platform. Chapter 5.3.1 elaborates why new technologies were introduced to the project. Chapters 5.3.2 – 5.3.4 give an overview of the most important implementations of the project.

5.3.1 Technologies Used

After some research, the author concluded that Marzipano was not the most practical tool to implement a video game in VR. Though it had the support for WebVR[31] that has the support for virtual reality, the decision was made to use a *game engine*. The reason behind this was due to the continuous support and advanced physics engines in game engines nowadays, which make the development and experimentation faster.

The author had three choices for development: Unity[32], Unreal Engine 4[33], and Godot[34], as these are the most popular game engines nowadays[35] that have VR support. Godot is the most lightweight of all of them, and it is an *open-source* program. However, it was not chosen because its community is smaller than Unity or Unreal Engine 4, which means in case of errors, there could not be that much support. Unreal Engine 4, on the other hand, is very popular game engine one could use to make games for nearly all platforms. However, the author felt that he could make the game more efficient with Unity. The conclusion came after trying out implementing a demo scene with Unreal Engine 4 and Unity.

The author chose C# for the scripting language to implement the next version of *Virtual Tour Game* from scratch. Unity features the Extended Reality (XR) Interaction Toolkit[36] to develop VR interactions in a game using a high-level interaction system that uses components. This made the development of the game faster in regard to time. Although, sometimes, it was a challenge for the author to understand how some specific systems work and can be customizable as it was the first time using this system.

Finally, because there exist different VR devices varying schemes of control [12] there was a need to choose the target device for the next version of the game. The target device for the game is Oculus Quest 2[37] (Illustration 36). It was chosen mainly because of two reasons. Firstly, the University of Tartu Institute of Computer Science had these devices ready for

---

[31] https://github.com/google/marzipano/tree/master/demos/webvr

[32] https://unity.com/

[33] https://www.unrealengine.com/en-US

[34] https://godotengine.org/

[35] https://www.incredibuild.com/blog/top-7-gaming-engines-you-should-consider-for-2020

[36] https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@0.9/manual/index.html

[37] https://store.facebook.com/quest/products/quest-2/

the students to develop on them during the writing of this thesis. Secondly, the author was fascinated by how effortless it was to use the device anywhere after having experience with HTC Vive Pro[38] and Oculus Rift[39].

Illustration 36. Oculus Quest 2 headset with controllers[40]

The chosen technologies were suitable for the game. During the development, there were not any technical limitations to the selected technologies. For example, no issues came with having all the panoramas in one scene at the same time.

[38] https://www.vive.com/us/product/vive-pro/

[39] https://en.wikipedia.org/wiki/Oculus_Rift

[40] https://www.techtastes.com/wp-content/uploads/2022/06/61tE7IcuLmL._SL1500_-500x500.jpg.webp

5.3.2 The Map

Illustration 37

Compared to the first version of the game, which had movement between rooms, movement in rooms, and a panoramic images rendering system, the second version needed implementations of those systems by the author. The author implemented one room as a 3D sphere object which contained the panoramic image as a texture. The player and every other object that was needed for the room are inside of their specific spheres.

The entire map of the game consists of the spheres (Illustration 37). Moving between rooms means traveling between those spheres. The author added an image on top of the player (Illustration 37) to show where the player is currently located. A special *camera* in the game world only sees and renders it on the texture of the map of the player that was mentioned in chapter 5.3.2.

5.3.3 Dialogue System

The most challenging task was to create the dialogue system that functions as it does in the first game but with a structure that can be easily expanded in the future. As mentioned in chapter 5.2.5, at first, the author wished to change the dialogue system entirely. However, that idea was not good for the game. So, the final implementation is similar to the system in the first version of the game. The author implemented software design patterns to keep the codebase maintainable.

There are four two main classes that are connected to the whole dialogue system when the player is talking to an NPC. Firstly, the Dialogue (Illustration 38) class itself represents a dialogue. It features an id, which gets automatically evaluated. It has a boolean signifying if the conversation with the NPC should start from the specific dialogue that has this value true. The dialogue class also has a choices list. The choices are the options that the player can choose when speaking to an NPC.

Secondly, the `DialogueHandler` class (Illustration 38) takes care of showing, hiding, and changing the dialogues. The dialogue handler itself is a Subject that can have Observers. This means the handler is using the observer pattern [13].

Additionally, as seen in Illustration 38, the dialogue handler has a state field as it derives from the Subject. This is the implementation of the state software design pattern [13], which is beneficial because when the player does actions, then there is no need to write a lot of if else statements for making the dialogue handler to react according to player's input and what kind of state it is in. Instead, it is now possible to give the dialogue handler new State classes that contain the functionality on how to react to the player's actions without making additional fields that hold in memory what kind of state the handler has.

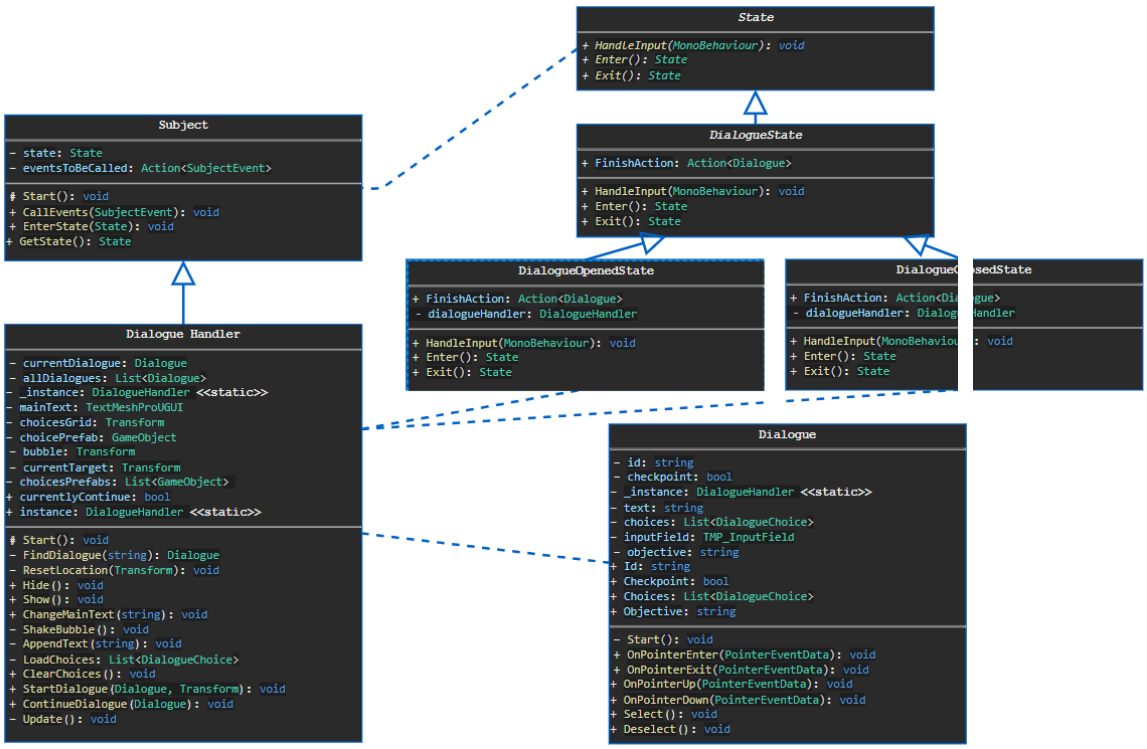

Illustration 38. The class diagram of the dialogue system

Currently only two states are implemented for the dialogue handler (Illustration 38). These are the `DialogueClosedState` and `DialogueOpenedState` that change according to when dialogue has closed or opened. Similarly, the player also has a component attached called `Tourer`, which derives from the Subject class, which means the state pattern applies to the player also. However, now there are not many states implementing it because there was no need for that. Although, this gave the possibility to extend the system in the future. It is relevant to mention because the author of the game pursued to implement software design patterns and to write code that can be altered in the future without a lot of effort. Keeping the code clean was also the goal of implementing the hotspot system.

5.3.4 Hotspot System

The Hotspot system in the second version of the game mainly relies on the XR Interaction Toolkit[41] made built in the Unity game engine. Every hotspot uses a component called `XRSimpleInteractable`[42] component (Illustration 39) that gets the input from XR Ray Interactor[43]. XR ray interactors are the components attached to the VR controllers. When the player points to the hotspot, a ray is visualized, and by pressing buttons from the controller, the player can interact with the hotspot. The author had to create his scripts with functionality that he added to the XR with simple interactable objects. Then, the custom events were called and executed after pointing and clicking on the hotspots. For example, teleporting between rooms works thanks to this, together with interacting with NPCs and their dialogues.

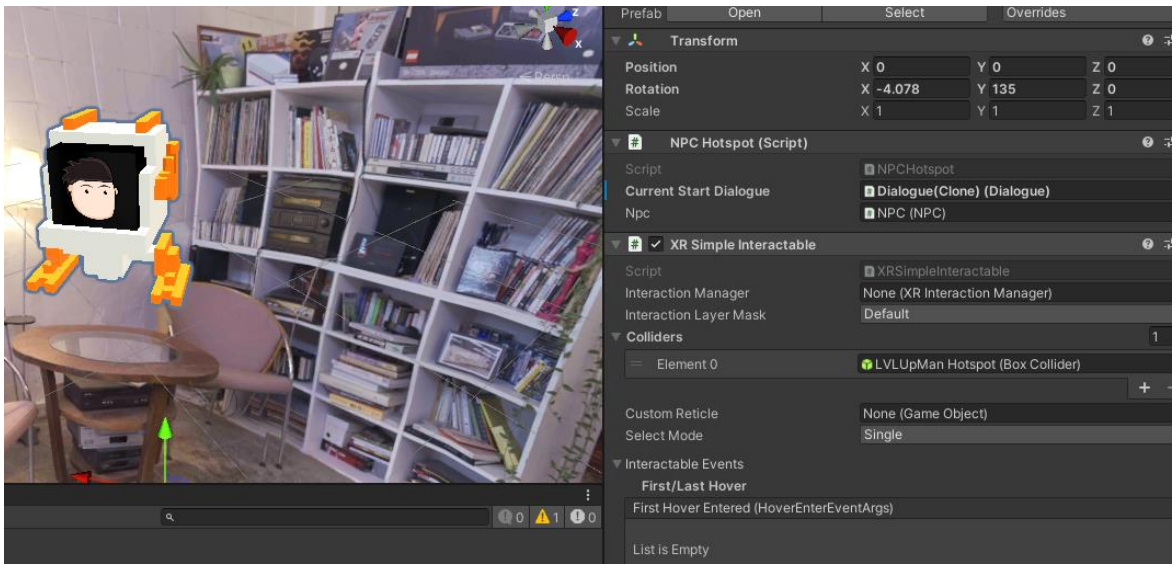

Illustration 39. XR Simple Interactable attached to the NPC hotspot.

Finally, when most of the systems were ready, the author tested the game and let other people experience it. This led to different results that are discussed in the following chapter.

[41] https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@0.9/manual/index.html

[42] https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.0/manual/xr-simple-interactable.html

[43] https://docs.unity3d.com/Packages/com.unity.xr.interaction.toolkit@2.0/manual/xr-ray-interactor.html

5.4 Testing

It is critical to playtest a video game to find out about all the problems it has and if the design and implementation of the game work [1]. Due to that, both versions of the *Virtual Tour Game* were tested on real users. Chapter 5.4.1 gives an overview of the methodology of the testing. Chapter 5.4.2 – 5.4.8 analyse the results gathered from the testing.

5.4.1 Methodology

There are two main goals regarding the testing of the *Virtual Tour Game*. Firstly, to find out if the design and implementation work for the virtual reality version. Secondly, to compare which version of the game was more entertaining for the player. Additionally, it was tested if the user interface and the application overall were working well in terms of usability. It is essential because when a piece of software is not usable, no one will use it [14].

It is enough to test with five users to find out the main shortcomings of an application [15]. The author of this thesis based the testing on that principle. However, the testing of the *Virtual Tour Game* VR version had four people. That means additional testing needs to be done in the future, but most of the problems with the software should still be discovered.

The average testing session length was about 40 minutes. It started with the playthrough of the first version of the game. The tester got 10 – 15 minutes to play the game without help. The author looked at how the tester managed to play and took notes when needed. After playing the game, the tester answered a questionnaire that applied to the game's first version. The primary categories of the questions were about the player to find out how experienced they are with playing video games. Other types included questions about the overall experience, game mechanics, user interface, and aesthetics. Finally, the testers were asked if they were likely to recommend the game to other people.

After successfully answering the first questionnaire, the author asked the testers to play the virtual reality version of the game again after 10 – 15 minutes. Then again when the playthrough was ended they were asked to answer the same questions but regarding the new version of the game. Both playthroughs were vital as they let us analyse which version the testers liked the most based on the test results.

5.4.2 Test Results of the Player Category

The first question of the testing was how often the testers play video games. Half of the testers play video games at least once a week. Only one person does not play video games regularly (Illustration 40).

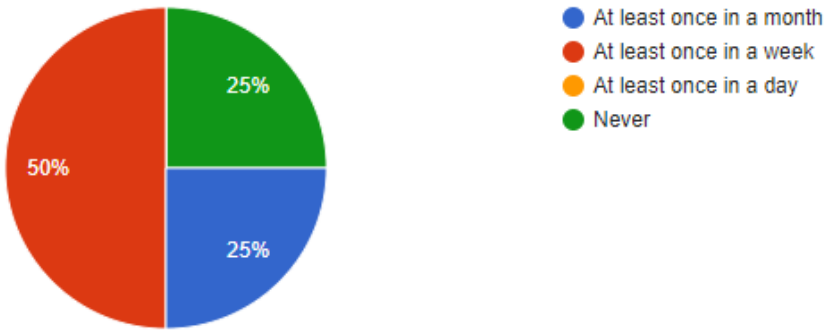

Illustration 40. How often do you play video games question responses

The most popular game genres among the testers are strategy and action/adventure games (Illustration 41). This is beneficial for testing the thesis game because it is also an adventure game, which means more ideas could come from the testers on how to improve the thesis game.

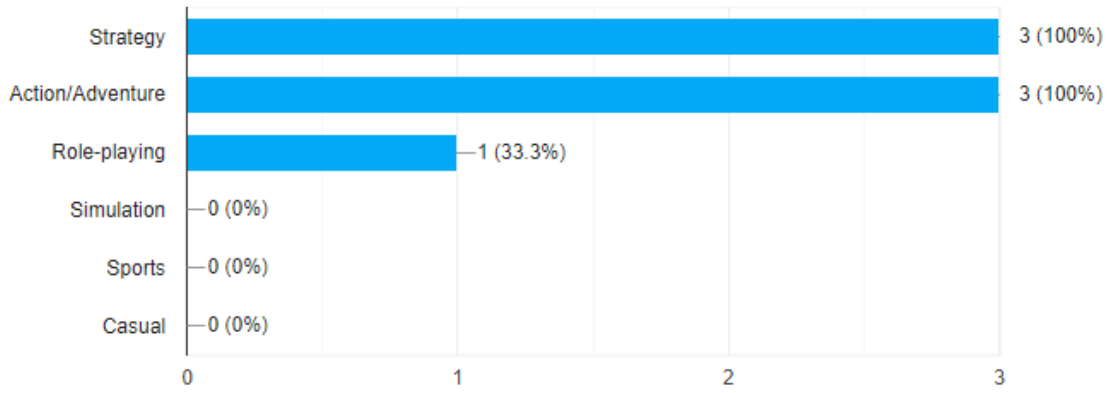

Illustration 41. What genres of computer games do you play question results

None of the testers play educational games regularly (Illustration 42). One tester has played, but it is not also their preference. The tester also mentioned the title of the game, which was *Brain Age: Train Your Brain in Minutes a Day!*[44]

Illustration 42. How often do you play educational game question results

This was the end of the questions from the first category. The primary intent was to "warm up" the testers for answering and to get the background of their contact with video games and educational video games in general. Based on the results, it seems in terms of experience, they are the "right" people to assess both versions of the game of this thesis, and their overall experience could benefit the testing by giving adequate feedback.

5.4.3 Test Results of the Overall Experience Category

The users had to rate the overall gameplay experience in terms of "fun" by scoring 1 to 6. Value 1 meant "awful" and 6 "perfect." The web version got an average score of 4.5 (Illustration 43). However, the VR version got an average score of 5.5 (Illustration 44). This means both the versions were fun for the testers, but the VR version was better. It was brought out that despite the fact the VR version had less content in it than the web one, it was still better in terms of fun. The testers did not like that they did not have time, for example, to read the books. The author noted during the test sessions that some of the testers were slightly under pressure when they started talking with the guests without knowing the correct answers.

[44] https://en.wikipedia.org/wiki/Brain_Age:_Train_Your_Brain_in_Minutes_a_Day%21

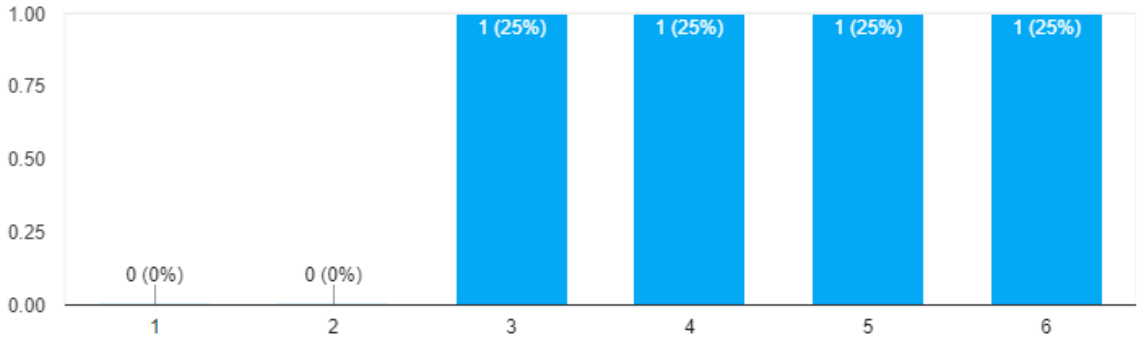

Illustration 43. Gameplay experience in terms of “fun” results (web version)

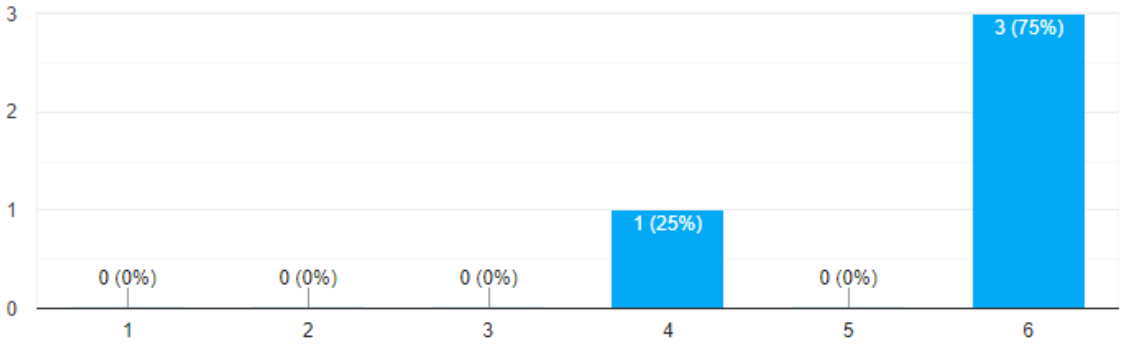

Illustration 44. Gameplay experience in terms of “fun” results (VR version)

The next question focused on how “educational” the game is. Again, the testers had to give the score from 1 to 6, where one meant “awful” and six “perfect.” The first version got an average score of 5.5 (Illustration 45) and the second one 4.5 (Illustration 46). That means the game is educational on both platforms, but the web has more content, making it more “educational.”

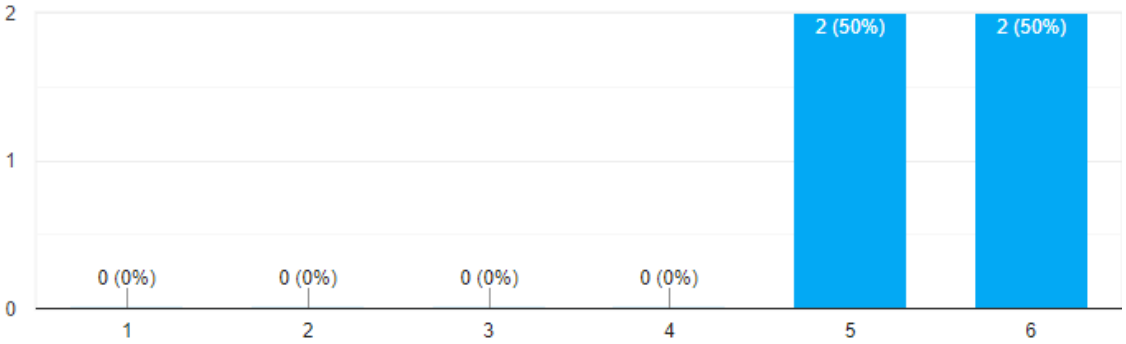

Illustration 45. Overall gameplay experience in terms of “educational” results (web version)

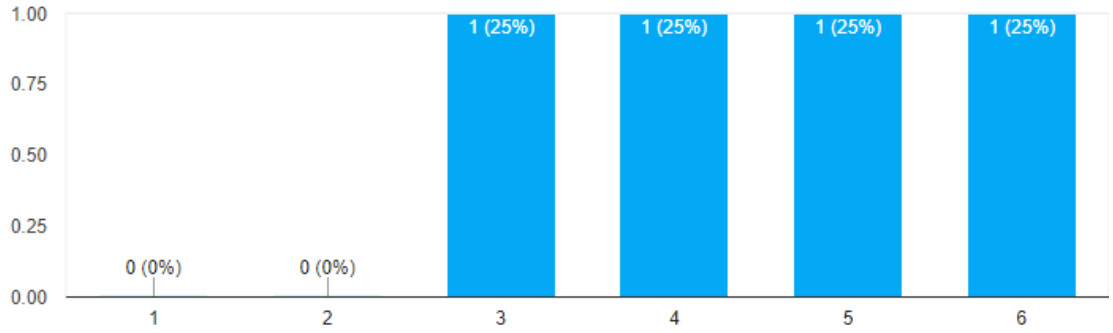

Illustration 46. Overall gameplay experience in terms of “educational” results (VR version)

The author also assessed how challenging the gameplay was for the testers. It was also assessed by letting the testers give a score from 1 (“impossible”) to 6 (“very easy”). The VR version was more accessible for the testers, with a score of 5.5 (Illustration 48). The web version got a score of 4.75. In general, both games were easy for the testers.

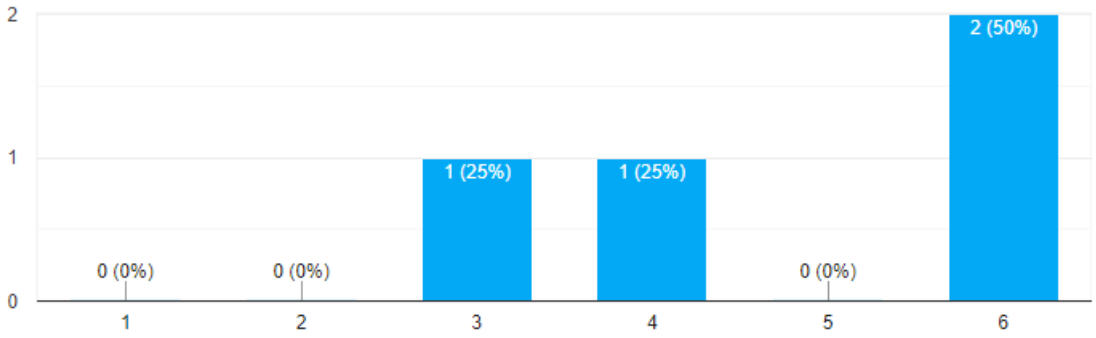

Illustration 47. Difficulty of the gameplay results (web version)

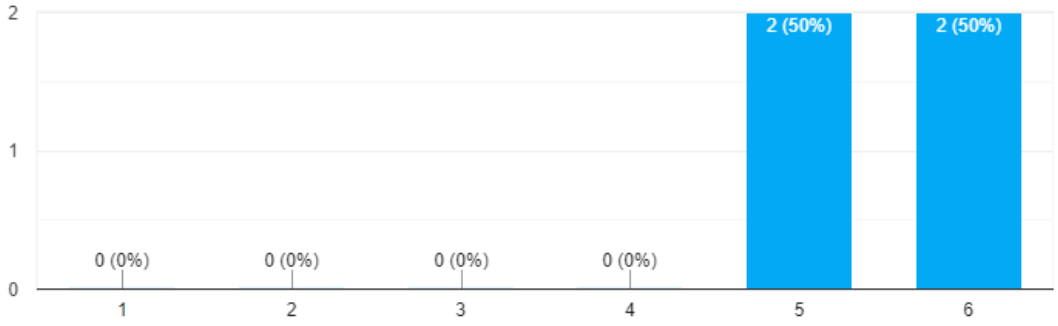

Illustration 48. Difficulty of the gameplay results (VR version)

The overall experience for the testers was positive for both games. The testers liked the museum and talking with the guests. Although, there was some negative feedback that the beginning of the game caused confusion and the conversations were too long. All the testers had different opinions on this. Therefore, it is critical to evaluate game mechanics to see where the shortcomings are of both versions of the game.

5.4.4 Test Results of the Game Mechanics Category

The next category included evaluating all the main gameplay mechanics from both games. The testers had to give a score from 1 ("awful") to 6 ("perfect") for every game mechanic in terms of how much value they added to the gameplay. In the end, say which one they liked most and which they did not.

Based on the results, the testers liked to move around more in the museum web version, with an average score of 5.75 (Illustration 49). However, the results for the VR version were also positive, with a score of 5.5 (Illustration 50).

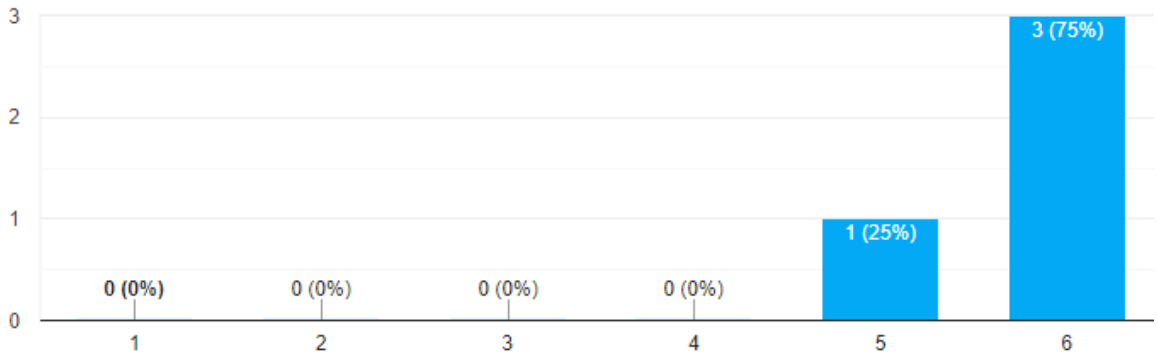

Illustration 49. Moving in the museum ratings (web version)

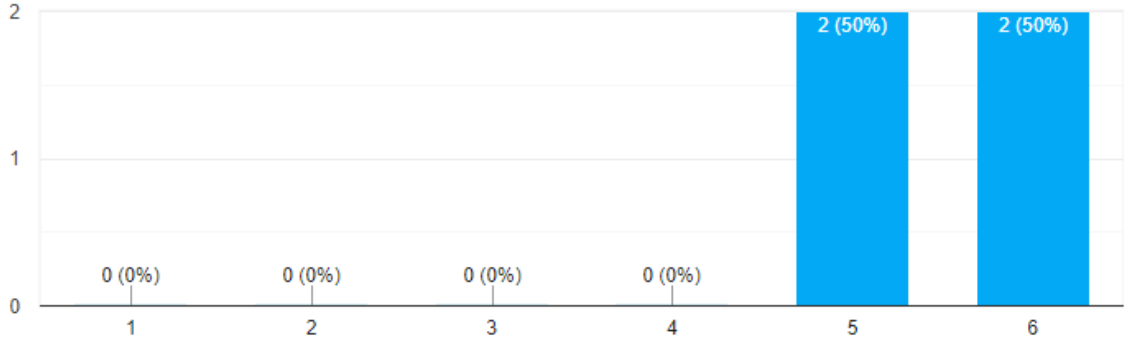

Illustration 50. Moving in the museum ratings (VR version)

Talking with the NPCs had more differences in the testing results between the two versions of the games. The first version got an average score of 5 (Illustration 51). The testers liked talking to NPCs in the VR mode with a score of 5.5 (Illustration 52). However, once more, the results are positive and do not have significant differences.

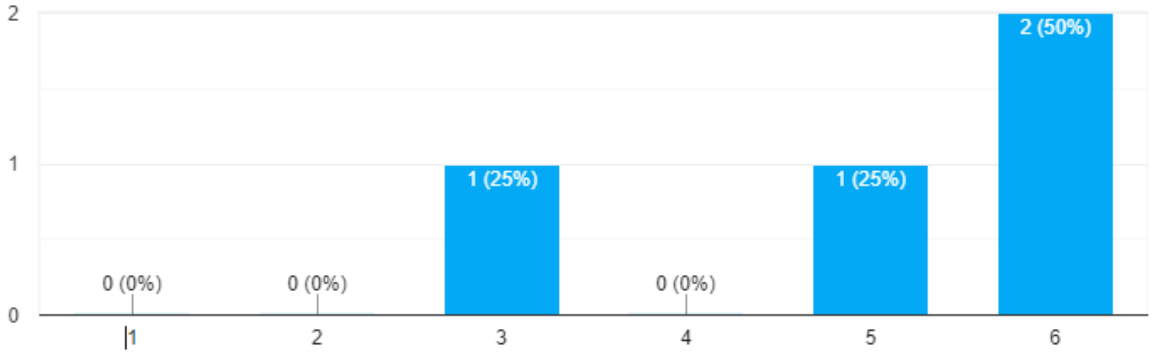

Illustration 51. Talking with NPCs ratings (web version)

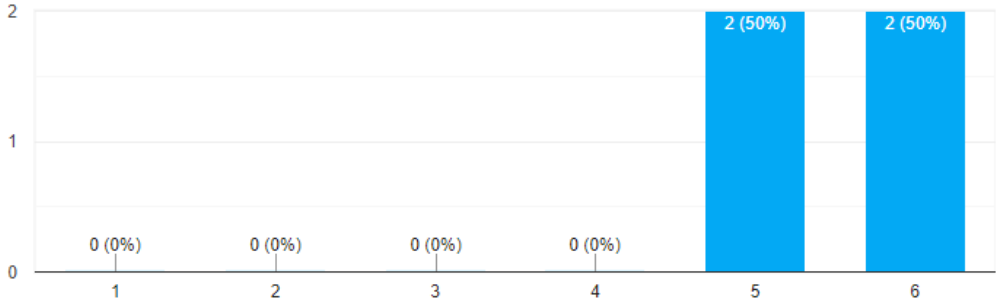

Illustration 52. Talking with NPCs ratings (VR version)

The next feature that was assessed was collecting items. The web version got an average score of 4.75 (Illustration 53). This time, the VR version was slightly higher, with a score of 5 (Illustration 54).

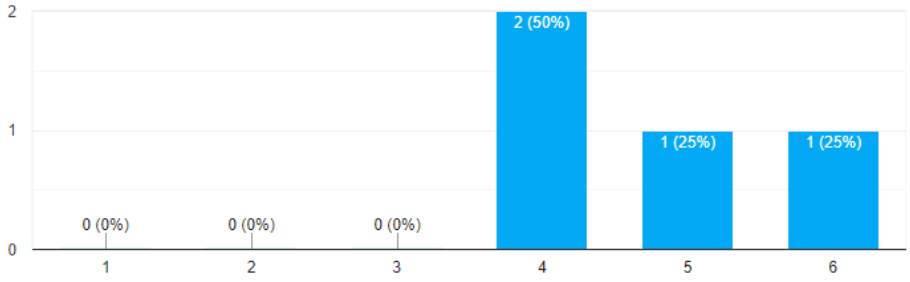

Illustration 53. Collecting items ratings (web version)

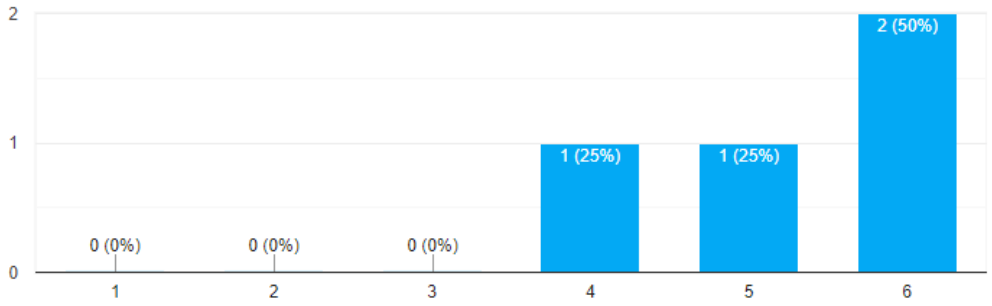

Illustration 54. Collecting items ratings (VR version)

The final evaluation of the game mechanics was the map feature. This time the average rating for the web version was 5.75 (Illustration 55). The VR version got a score of 4.25 (Illustration 56). This is due to the reason that it was not possible to change rooms by pressing on the points that were on the map.

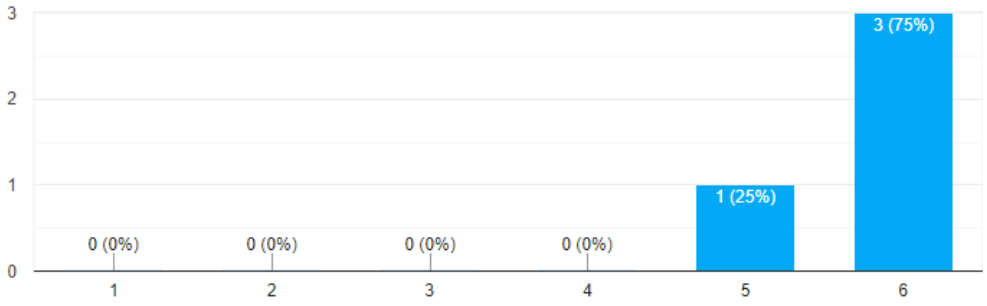

Illustration 55. The map feature (web version)

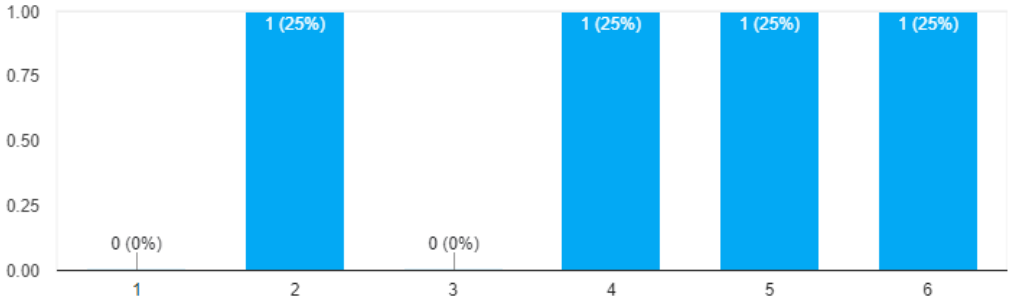

Illustration 56. The map feature (VR version)

Overall, the main gameplay mechanics added value to the gameplay based on the test results in both versions. Some adjustments need to be made with NPCs, as some testers said the conversations should be more meaningful. Everyone again had liked and disliked different

game mechanics. So, it cannot be said that some game mechanic is not working because overall, the average ratings were positive.

5.4.5 Test Results of the User Interface Category

The testers also had to evaluate the controls' comprehension for both games. Similar to the last assessments, they had to give a score from 1 ("very confusing") to 6 ("very intuitive"). The results were mainly positive. The web version got an average score of 5.75 (Illustration 57). At the same time, the comprehension of the controls was less intuitive in the virtual reality version than in the web version. The average score was 5 (Illustration 58).

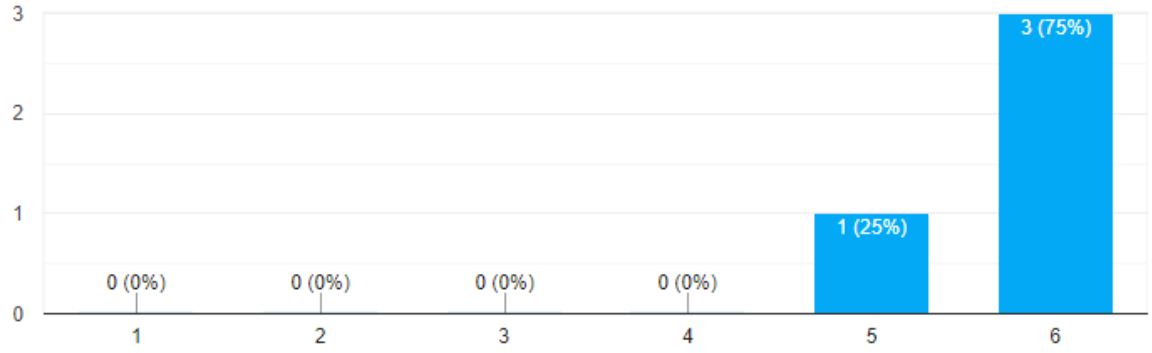

Illustration 57. Comprehension of the controls results (web version)

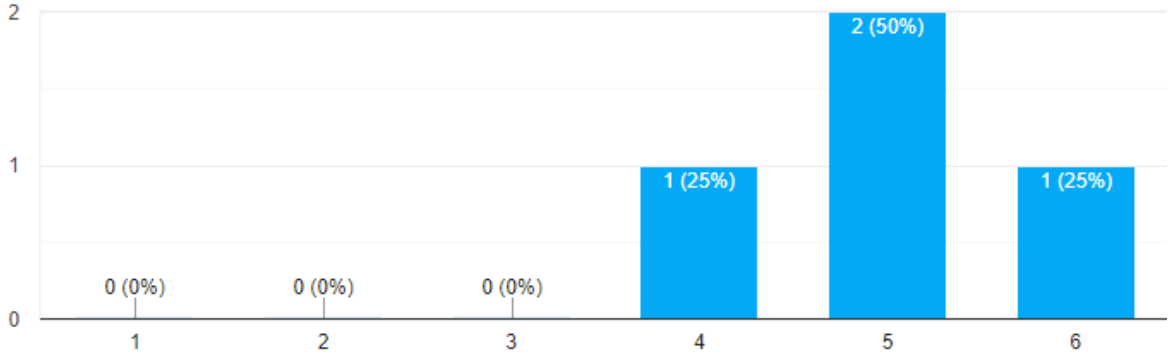

Illustration 58. Comprehension of the controls results (VR version)

There were no problems with the controls of the web version. Although with the VR version, the testers did not like the map teleportation feature did not work. One of the testers did not like starting and ending the talk with an NPC by just pointing the ray at them. After seeing how the user interface works for the testers, it was relevant to find out how they rate the GUI of both versions. Especially in terms of usability by giving the score from 1 ("awful")

to 6 (“perfect”). The web version was assessed with an average score of 5.75 (Illustration 59), whereas the VR version got a score of 5.5 (Illustration 60), which is positive. Considering the results, the GUIs of both versions of the game are usable.

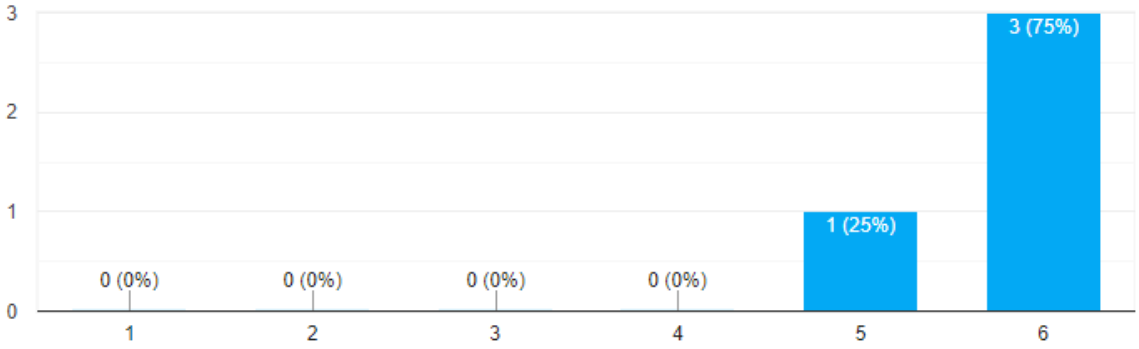

Illustration 59. GUI usability ratings (web version)

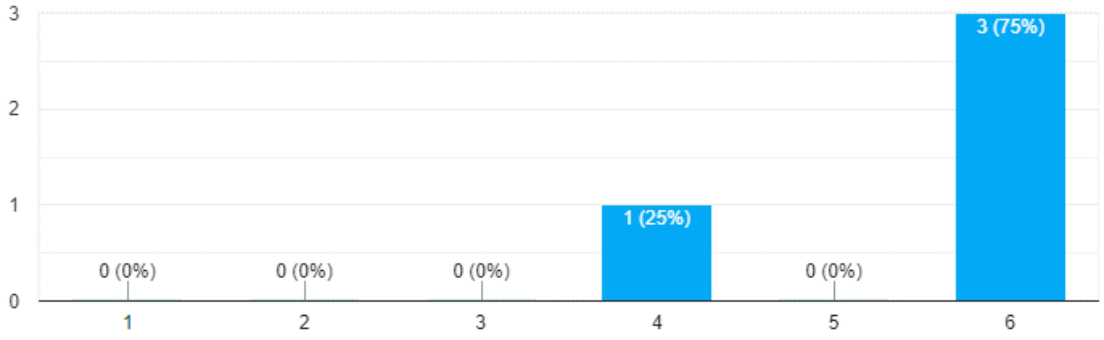

Illustration 60. GUI usability ratings (VR version)

The testers had to give a score from 1 (“very confusing”) to 6 (“very intuitive”) to assess the comprehension of the GUI. The web version got maximum points (Illustration 61). The virtual reality version also got an almost perfect score 5.75 (Illustration 62), which shows that there are no severe problems regarding GUI.

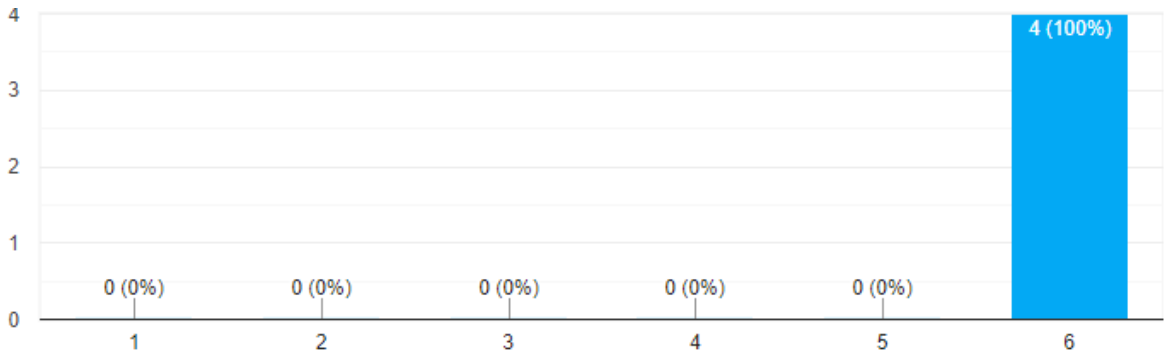

Illustration 61. GUI comprehension ratings (web version)

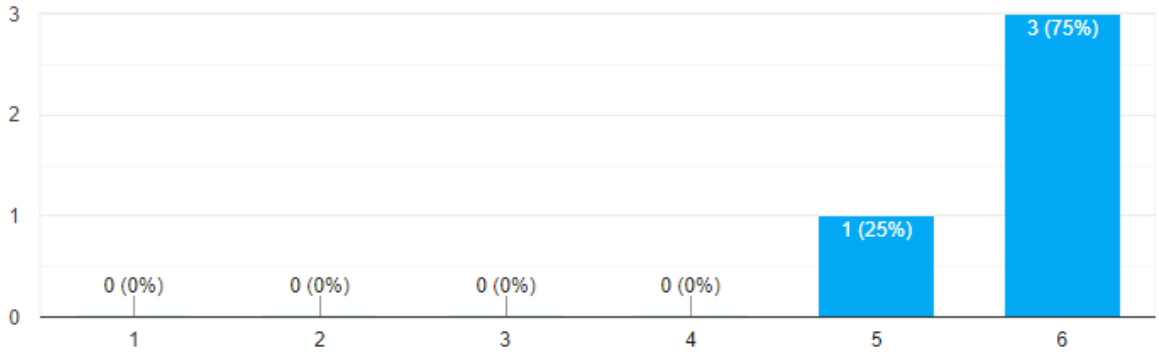

Illustration 62. GUI comprehension ratings (VR version)

The GUI part of both games work well based on the test results. However, the VR version needs some improvements on the aspects mentioned earlier. Video games need to be visually pleasing also besides being comprehensive.

5.4.6 Test Results of the Aesthetics Category

One of the last assessments was to find out how the testers like the appearance of the game. It is an essential characteristic of a video game. That is the reason why the testers needed to assess how both versions of the game look by evaluating it with a score 1 (“awful”) to 6 (“perfect).

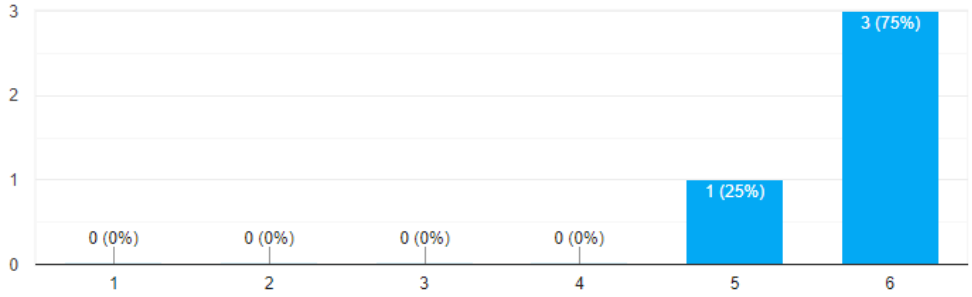

Illustration 63. The appearance ratings (web version)

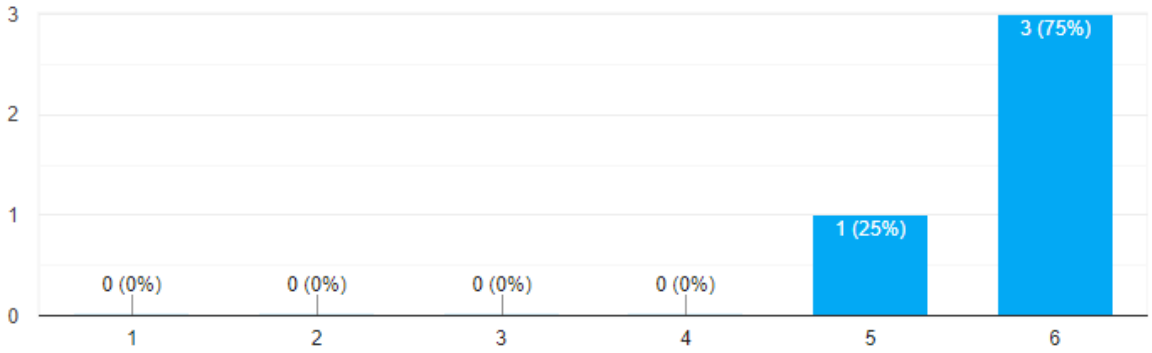

Illustration 64. The appearance ratings (VR version)

The web and VR version got the same number of points (5.75) in terms of appearance (Illustration 63, Illustration 64). This means that the testers like how the game looks and there are no major issues with this.

5.4.7 Test Results of the Conclusion Category

Finally, it was time to assess if the testers found the game attractive to show it to their friends. Therefore, lastly, they had to give the value 1 (very unlikely) to 6 (very likely) if they would show and introduce the game to their friends.

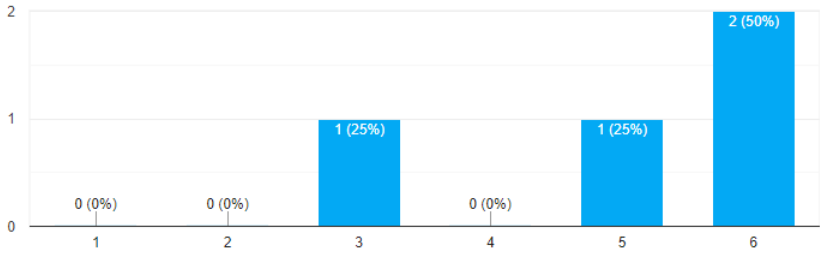

Illustration 65. Recommendation ratings (web version)

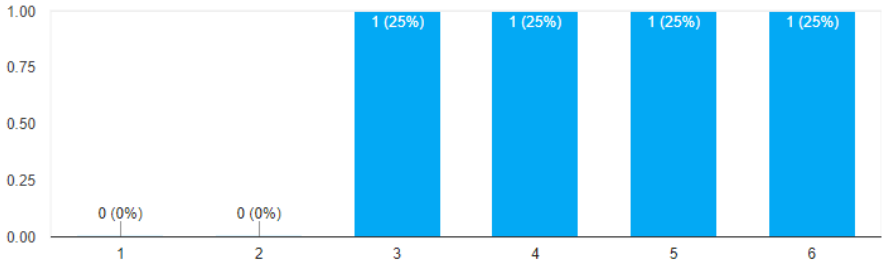

Illustration 66. Recommendation ratings (VR version)

Based on the results, the web version could be introduced to the friends of the testers more likely than the VR version. It could be because the first version offered more content to the player and did not have the problems that the VR version of the game had, like the map teleportation feature not working.

In conclusion, it can be said that both games had similar test results. Their design and implementations work. However, further development is needed to fix all the issues that came out during testing.

6 Conclusion

Video games consist of different elements called game mechanics, which are the rules and functionalities they include. It is expected that video games are typically designed and implemented to offer unique experiences and entertainment for the player. However, video games can also be beneficial in terms of education. These games are called educational or studying games, which can give the player the ability to learn while being entertained.

The result of this master's thesis features two developed versions of an educational game titled *Virtual Tour Game*. The first version was made for the University of Tartu's course *Evolution of Video Games* to replace the commonly used learning materials like Moodle tests with an interactive learning experience. The students of the course had an opportunity to be in the LVLup! Video Game Museum virtually and play while learning. They would also get points for the course directly from the game using a web browser.

Due to the positive feedback for the first version, it was decided to continue with the project. The goal of virtual reality is to take the user and put them in a virtual world so that it feels more real. The next iteration of *Virtual Tour Game* that was made for this thesis had the same goal of giving a more real-life feeling to the game. It was developed for the Oculus Quest 2 device.

Similar games were researched when developing both versions of the game in this thesis. They were mostly used as inspiration to make design choices using the iterative design process. While designing the game, principles of game design and virtual reality design field were followed.

Different technologies were researched to implement both versions of the game. The first version was made using completely different technologies from the second one. Software design patterns were used to create cohesive systems that can be developed further after the thesis.

Both versions were playtested to evaluate the quality of design and implementation. Moreover, they were compared to see which one was more beneficial to the tester. Based on the test results, the design and the implementation of the games was entertaining and educational. Issues found from the playtesting sessions are going to be fixed in the future when developing both games even further.

Special thanks go to supervisor Ulrich Norbisrath who did not give up and made suggestions to help improve the quality of the virtual reality version and this thesis. Many thanks go to Raimond-Hendrik Tunnel who is the original mind behind the first version of the game together with Mark Muhhin.

7 References

[1] Schell J. The Art of Game Design – A Book of Lenses. United States of America, Burlington: Morgan Kaufmann Publishers. 2008.

[2] Salen K, Zimmerman E. Rules of Play – Game Design Fundamentals. England, Massachusetts London: The MIT Press Cambridge. 2004.

[3] D. Nael, Icarus - a Real-Time Strategy Game in Space, Bachelor's thesis, University of Tartu, 2019.

[4] Barr M. Video Games Can Develop Graduate Skills in Higher Education Students: A Randomised Trial. Online Submission [Internet]. 2017 Jan 1 [cited 2022 Jul 6] ;113:86–97. Available from: https://search-ebscohost-com.ezproxy.ut-lib.ut.ee/login.aspx?drect=true&AuthType=ip,url,uid&db=eric&AN=ED608315&site=eds-live

[5] de Freitas S. Are Games Effective Learning Tools? A Review of Educational Games. EDUCATIONAL TECHNOLOGY & SOCIETY [Internet]. 2018 Apr 1 [cited 2022 Jul 6];21(2):74–84. Available from: https://search-ebscohost-com.ezproxy.ut-lib.ut.ee/login.aspx?drect=true&AuthType=ip,url,uid&db=edswss&AN=000429647500006&site=eds-live

[6] Milic S, Simeunovic V. Exploring e-learning critical success factors in digitally underdeveloped countries during the first wave of the COVID-19. INTERACTIVE LEARNING ENVIRONMENTS [Internet]. 2021 Oct 19 [cited 2022 Jun 18]; Available from: https://search-ebscohost-com.ezproxy.utlib.ut.ee/login.aspx?direct=true&AuthType=ip,url,uid&db=edswss&AN=000709231400001&site=eds-live

[7] M. Muhhin, Evolution of Video Games Online Course, Master's thesis, University of Tartu, 2021.

[8] Jerald J. The VR Book: Human-Centered Design for Virtual Reality. Association for Computing Machinery and Morgan & Claypool Publishers. 2016.

[9] STATISTA [Internet]. Available from: https://www.statista.com/topics/2532/virtual-reality-vr/#topicHeader__wrapper [cited 2022 Jul 24].

[10] S. Ussanov, Three-dimensional user interface design and development, Bachelor's thesis, University of Tartu, 2019.

[11] F. Raud, A Ludic Dialogue System for Interactive Fiction, Bachelor's thesis, University of Tartu, 2021.

[12] P. Paal, Virtual Reality Headset Study, Bachelor's thesis, University of Tartu, 2021.

[13] Nystrom R. Game Programming Patterns. [S.I.]: Genever Benning. 2014.

[14] Nielsen J. Usability 101: Introduction to Usability. 2012. https://www.nngroup.com/articles/usability-101-introduction-to-usability/ (03.05.2022)

[15] Nielsen J. Why You Only Need to Test With 5 Users. 2000. https://www.nngroup.com/articles/why-you-only-need-to-test-with-5-users/ (03.05.2022)

Appendix

I. Glossary

1. **Aesthetics** – a fundamental game element that interprets the appearance and the experience of the game [1].
2. **Application programming interface (API)** – a software intermediary that links two applications[45].
3. **Camera** – a camera in the game that can be rotated and moved to certain spots during gameplay[46].
4. **Caret** – the bar (or other symbol) marking the active editing point.
5. **Game engine** – an environment that is used to create the game software.
6. **Game mechanics** – a fundamental game element, consists of the main rules and features of a game [1].
7. **Iterative design** – a repeating designing process that takes into account playtesting results and prototyping [2].
8. **Open-source** – software with source code that has been made publicly available.[47]
9. **Playtesting** – a testing of the game by playing it (1)
10. **Story** – a fundamental game element, a particular order of the events that happen in the game [1].
11. **Technology** – makes the game mechanics, story and aesthetics in the game functional.
12. **Template** – a gauge, pattern, or mould, commonly a thin plate or board, used as a guide to the form of the work to be executed.
13. **Virtual reality** – a digital environment that is created by a computer so the user can participate in it as if it truly existed in the real world. (8)

[45] https://www.mulesoft.com/resources/api/what-is-an-api

[46] https://www.whatgamesare.com/2011/10/camera-comes-first-game-design.html

[47] https://www.oxfordlearnersdictionaries.com/definition/english/open-source?q=open-source

II. Launch Guide

To access first version of the Virtual Tour Game please write to daniel.nael@ut.ee.

To launch the *Virtual Tour Game* VR version, the apk file must be installed on the Oculus Quest 2.

Use this tutorial to install the game: https://vr-junkies.com/en/install-apk-files-on-oculus-quest-2/

III. Source Code

The source code and assets of Virtual Tour Game initial version are private. They can be accessed by writing to daniel.nael@ut.ee.

The source code and assets of Virtual Tour Game virtual reality version can be found at https://github.com/dannynt/virtual-tour-game-vr-next

IV. Accompanying Files

The accompanying files can be found from the repository https://github.com/dannynt/virtual-tour-game-vr-next.

The archive file containing the accompanying files has the following structure:

• /Build – the folder that contains the *Virtual Tour Game* VR version apk file.

• /Testing – the folder that contains questionnaire of test sessions and answers of the testers to it.

• vtg_demo.mp4 – a video demonstrating the first version of *Virtual Tour Game*.

• vtg_vr_demo.mp4 – a video demonstrating the second version of *Virtual Tour Game*.

V. License

Non-exclusive licence to reproduce the thesis and make the thesis public

I, **Daniel Nael**,

(*author's name*)

1. grant the University of Tartu a free permit (non-exclusive licence) to reproduce, for the purpose of preservation, including for adding to the DSpace digital archives until the expiry of the term of copyright, my thesis

Virtual Tour Game,

(*title of thesis*)

supervised by **Ulrich Norbisrath**

(*supervisor's name*)

2. I grant the University of Tartu a permit to make the thesis specified in point 1 available to the public via the web environment of the University of Tartu, including via the DSpace digital archives, under the Creative Commons licence CC BY NC ND 4.0, which allows, by giving appropriate credit to the author, to reproduce, distribute the work and communicate it to the public, and prohibits the creation of derivative works and any commercial use of the work until the expiry of the term of copyright.

3. I am aware of the fact that the author retains the rights specified in points 1 and 2.

4. I confirm that granting the non-exclusive licence does not infringe other persons' intellectual property rights or rights arising from the personal data protection legislation.

Daniel Nael
08/08/2022