UNIVERSITY OF TARTU
Institute of Computer Science
Computer Science Curriculum

Jelizaveta Kuznetsova

Interactive VR Art Gallery: On the Other Side of the Painting

Bachelor's Thesis (9 ECTS)

Supervisor: Ulrich Norbisrath, PhD

Kärt Summatavet, PhD

Tartu 2021

Interactive VR Art Gallery: On the Other Side of the Painting

Abstract:

The thesis describes the innovative concept of an interactive VR art gallery application „On the Other Side of The Painting“ and the development of its prototype. The solution aims to enhance the painting allowing the spectator to enter and interactively explore the world of painting but still return to the classic gallery format of displaying the painting itself as well as other paintings. This thesis covers an overview of similar applications, a detailed prototype application design and an evaluation with target group feedback. The virtual gallery prototype was created according to the basic principles of virtual reality applications and game design.

Keywords:

Virtual Reality, art, gallery, Unreal Engine 4, application, game design

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

Interaktiivne VR Kunstigalerii: Maali teisel poolel

Lühikokkuvõte:

Töös kirjeldatakse interaktiivse VR kunstigalerii rakenduse "Maali teisel poolel" uuenduslikku kontseptsiooni ja selle prototüübi arendamist. Lahenduse eesmärk on täiustada maali, mis võimaldaks pealtvaatajal siseneda ja interaktiivselt uurida maalimaailma, kuid siiski naasta nii selle maali kui ka teiste maalide eksponeerimise klassikalises galeriiformaadis. Käesolev töö hõlmab ülevaadet sarnastest rakendustest, detailset rakenduse prototüüpi disaini ja hindamist koos sihtrühma tagasisidega. Virtuaalse galerii prototüüp on loodud vastavalt virtuaalreaalsuse rakenduste ja mängudisaini põhiprintsiipidele.

Võtmesõnad:

Virtuaalreaalsus, kunst, galerii, Unreal Engine 4, rakendus, mängu disain

CERCS: P170 Arvutiteadus, arvutusmeetodid, süsteemid, juhtimine

Table of Contents

1. Introduction

Nowadays, technology has become intertwined with numerous areas of everyday life. Art is no exception. Some artists have switched from traditional forms of painting to digital formats, and some even joined game development as designers and asset creators. Museums worldwide try to attract more people with technological solutions such as logic, history, or book-themed games in VR. One example is the museum "Kalevipoja Koda" in Estonia which offers visitors a role-play VR game of Estonia's most known epic poem "Kalevipoeg"[1].

Virtual Reality, further VR, is an application that imitates real-life experience in an artificial environment. It is one of the most popular solutions with immersive techniques and enhanced experience. VR games are capable of capturing players' attention with beautiful scenes, exciting interactions with the world, or captivating stories. However, most importantly, they provide the participants with the freedom of movement. The most common use of VR today is in educational or entertaining games, and often museums use them as an additional means of knowledge delivery to main exhibitions. But what about art galleries and paintings? How could VR be helpful in attracting the attention of visitors?

The interest in this thesis topic emerged from the author's artistic background, to be more precise, her idea of entering the painting. Virtual Reality can potentially be one of the solutions to convey the author's vision. The main goal of this thesis is to introduce an idea and create a detailed prototype of the "Virtual Art Gallery: On the Other Side of the Painting" application, which will show more of the painting's world behind the pane. In the application, Virtual Reality themed interactive rooms were established based on some pre-created oil paintings of the artist. The room presents the world of the respective painting that looped in time, like living the exact moment of life that the painting captures. The artworks themselves present as portals to the world of the painting.

This work consists of seven chapters in total. The background chapter will introduce known solutions and applications used in museums and briefly discuss the Virtual Reality concept. The design chapter describes the preparation and design details considering game/application development cycles. The fourth – implementation - chapter will overview the detailed prototype creation with custom assets. The final testing chapter summarises conclusions from the feedback analysis and improvements with future possibilities. In the **Glossary** (**Appendix** I) are defined some terms used in this thesis. The personal illustrative materials used in this work are also provided in a larger format in the **Illustrations** (**Appendix** II). The project source code is accessible through the repository, with the installation guide in **Appendix** III. The users' testing feedback form can be found in **Appendix** IV, and footage of the testing session in **Appendix** V.

[1] "Kalevipoeg" https://www.kalevipojakoda.ee/en

2. Background

According to Jason Jerald [1] - "..**virtual reality** is defined to be a computer-generated digital environment that can be experienced and interacted with as if that environment were real.". Also, the environment may contain experience-enhancing elements that can be similar to or completely different from the real world. Virtual Reality (further abbreviated as VR) applications can be for educational, entertaining, or commercial purposes. VR games could captivate all attention with beautiful scenes, exciting interactions with the world, or breaking stories but, most importantly, give the freedom to move. It may also strongly affect users' ideology through the concepts of immersion and presence, creating an empathic connection to its users. The current usage of VR is games, and some museums use them to augment their main exhibitions.

2.1. Previous studies

More museums started engaging technologies to attract customers with different forms and genres of games in the last few years. The story, role-play, and quiz games are widely used in this situation because they entrain as much as educate visitors. As for forms, museums try to use different approaches as projections with feedback, widely known game consoles such as Xbox with Kinect, and reborn in 2016 Virtual Reality with new solutions. This work reviews Virtual Reality solutions as the world tries small steps further toward providing a base for the Metaverse (**Appendix** I) that will connect all social fields.

Virtual Reality is an application that allows experiencing the natural world through the artificial environment with the help of specific devices [1]. An application can be accessed everywhere with an internet connection if it is necessary. So, the applications of places will become more mobile and available to the masses. The world has been in pandemic lockdown and isolation for the last few years, so virtual solutions may support businesses and museums in the most challenging times. It is worth mentioning that masterpieces should be contained in a particular environment and kept within those conditions for preservation [2]. As Downing notes [2], it is one of the most significant challenges for museum administrators to preserve masterpieces yet attract more visitors. Setting up the VR gallery would not harm art pieces during transportation and could be shown to the public.

The availability of applications depends on the availability of devices. Such devices could be a personal computer and head-mounted display or mobile phones. Using devices would significantly affect the experience compared to real-world experiences, like the lower quality of pictures and the inability to smell or touch. The application might not yet fully replace real-world feelings; conversely, it could affect the perception of VR, as users may start to confuse virtual with real.

One group from Taiwan [3] with Dr. Chih-Long Lin, Dr. Si-Jing Chen and Prof. Dr. Rungtai Lin researched the efficacy of Virtual Reality in painting exhibitions at the beginning of 2019. They have compared desktop and Head Mounted Display (HMD) versus ordinary exhibitions. The exhibition was made with the Unity game engine, using 12 paintings provided by the amateur artist Lee S. [3]. The group experimented with the 3D-VRII headset provided by

KWorld that is connected to a phone, similarly to Samsung Gear VR[2]. According to this research, final results have shown that the visitors prefer to see pictures with their own eyes. The visitors have also pointed out the advantage of seeing paintings non-included in real-life exhibitions in a virtual gallery and the opportunity to see and choose the most exciting painting for them. Turning to technical aspects, most people did not like motion sickness created by a head-mounted display connected to a phone, so they preferred to use the desktop version in most cases. The group has pointed out the ability to create a situation to draw viewers' attention to specific details that may not be noticeable in the first place in ordinary conditions. However, visitors did not support that point of view. The researchers admitted that the virtual gallery still needs improvements to be as impressive as a physical one

In summary, Virtual Reality solutions so far cannot fully replace galleries. However, the mentioned solution attracts people with innovativeness to the exhibition. The stand-alone VR application is an excellent solution if there is no opportunity to visit a physical gallery or museum. It is also an excellent solution for amateur artists who cannot afford a full-scale gallery but still want to attract people's attention. It is also essential to allow people to choose which devices to use desktop, mobile, or headset. Moreover, it is crucial to implement a smoother VR experience to avoid motion sickness. Therefore, the author aims for a better experience with smoother locomotion as Meta Quest 2 performs better than the headset used in the research. Also, the author intends to advance into the interactions that users can perform in a gallery. The result will be a different experience from the results gained in the research.

2.2. Similar ideas

An association named "Estonian Virtual and Augmented Reality Association" was officially established recently in Estonia [4], which has gathered Estonian companies that develop Virtual Reality games and applications. One of them, Maru VR, has created a role-play VR game, "Son of Kalev's Journey to Hell – A VR Experience" (Illustration 1), based on Estonia's most known epic poem "Kalevipoeg", specially for the museum "Kalevipoja Koda" [5]. In the mentioned game, the visitor is in the Son of Kalev's role and repeats the same adventures described in the legend. This game's actions and story have a connection to the permanent exhibition. This game is still upgradable, bugs are being fixed, and new game levels are being developed. However, it is an expensive solution that museums can afford with support, which might pay off. While for changeable exhibitions, the situation is not the same.

[2] Samsung Gear VR

Illustration 1. VR Experience "Son of Kalev's Journey to Hell" video footage screenshot [5].

Art galleries change their exhibitions each season, so making the permanent game is an expensive one-time purchase. However, a similar game solution was also used in the exhibition "*Alice: Curiouser and Curiouser*" at the Victoria and Albert Museum (V&A), which took place in London starting from summer till the end of the year 2021 [6]. This exhibition was based on the famous novel "Alice in Wonderland" by Lewis Carroll, and the museum offered the game in addition to the regular exhibition. The role-playing game, "Curious Alice: the VR experience" (Illustration 2), was produced as part of the exhibition by Vive and then added to their metaverse project ViveArt [7]. HTC Vive is a serious and big company that produces Vive VR headsets, accessories, and software and can afford the creation of their own version of Metaverse, and therefore lend out and reuse or sell the created game.

Illustration 2. Curious Alice: the VR experience screenshot [6].

On the internet, solutions for ordinary people can also be found. Virtualmerse has created the stand-alone virtual gallery (Illustration 3) based on artist Barbara Modinger's paintings and is accessible through a web browser [8]. This gallery also can be accessed by VR headsets such as Oculus Quest 1. The exhibition is like a regular one, with hanging paintings on the wall and descriptions near them. The difference from the real one is the absence of people. CoSpaces [9] and Hubs Mozilla [10] offer the same solution, where anyone can create their art exhibition (Illustrations 4 and 5). Hubs Mozilla generally offers room creation in the Spoke 3D tool, customisable from default asset packages and put into collaborative spaces in Hubs, so it is not only about galleries. The concept is simple: create space, insert images and description info – the virtual exhibition is ready. These constructors require fewer human and data resources compared to VR games.

Illustration 3. Virtualmerse Barbara Modinger's art gallery screenshot [8].

Illustration 4. CoSpaces gallery example screenshot [9].

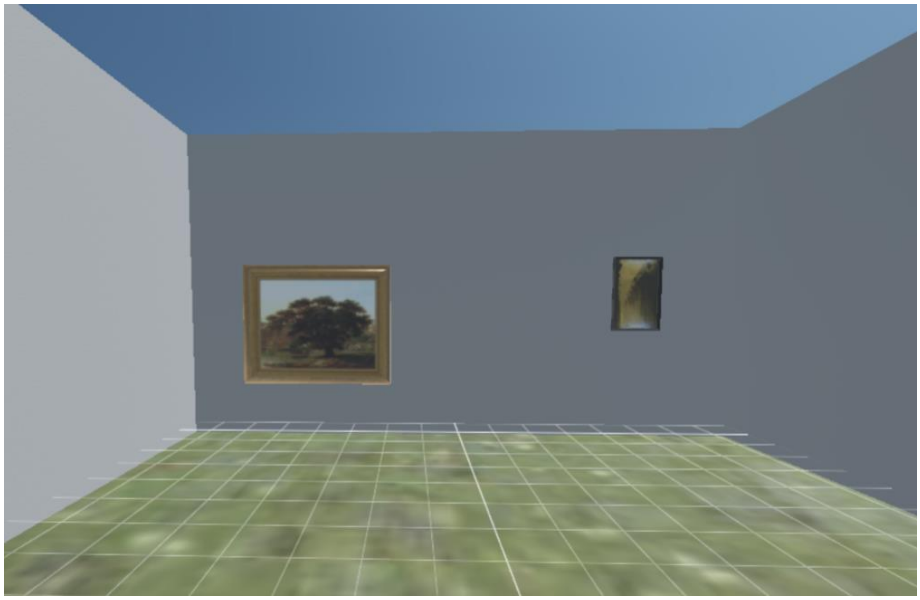


Illustration 5. Hubs Mozilla. Gallery made in Spoke.

The listed solutions are either expensive or imitate real-life exhibitions and miss the potential of interactivity that may be implemented in VR. Some solutions have advantages and disadvantages, such as the availability, customisation and low interactivity of Hubs and the interactivity with high expenses of “Curious Alice”, and restrictions have formed the concept of a middle solution. In this thesis, the advancement of the classic gallery is one of the main challenges.

3. Application Design

Currently, the most common VR usage is games, which could be educational, social or entertaining. Therefore, a classic Game Development Life Cycle, shortly GDLC (**Appendix** I), governs this thesis prototype's development management. The first stage of GDLC is an initiation that includes an overview of an idea, defining a target audience and team formation.

3.1. Initiation - Idea description

The idea came from the author's artistic background, developer skills, and the wish of seeing the painting from the inside, entering it, and seeing the painting's world how the artist sees it. As the author possesses arts and development skills, it allows enhancing the arts field with technologies. Virtual Reality seemed to be a promising technology to allow the realisation of the author's vision.

The main goal of this thesis is to describe the concept and design and develop a detailed prototype application of a virtual art gallery. The thesis aims to show the capabilities and potential of VR in the art field by designing a prototype and finding out if there is an interest in a target group. The solution brought in this thesis is just the beginning of a more significant project. The detailed application development is left into the future when VR could reach its full potential.

The application aims to show worlds created by artists in VR based on their paintings. Gallery application itself consists of two main parts. One of them is similar to an ordinary gallery with artworks displayed on the walls, and the second part is a room with the world of the artist's painting. The artworks themselves are the portals that lead inside the world of the paintings. These portals are slightly moving, showing that the world behind them is alive. By interacting with paintings, visitors transfer through the portal into another room where they can interact with different objects related to the painting. For example, the author uses her own paintings in this work, that listen in **Appendix** II.

To add such an interactive painting portal, the developer has to create interactive rooms based on a particular pre-created artist's painting. The artist decides how rooms will be organised and defines the entering - and exit points. Rooms are the paintings' worlds seen from the artist's perspective. They can show a slice of time that has been captured in a 3D room. Rooms are populated with static and interactive objects that simulate the real-world experience, like a bubble that will pop if it is touched, and some non-realistic, such as flying ghosts, that may go through the objects and give the spectator a thrill.

The envisioned virtual gallery application is aimed at this generation's artists to enhance their paintings. Furthermore, this is also for an amateur artist who wants to show the story behind their picture. In addition, art students may recreate a well-known masterpiece in 3D to create their portfolio [11]. Moreover, regular VR users are gamers with personal devices. Sometimes Estonian schools provide limited sessions for their students [12]. Therefore, they will most likely be the visitors of the virtual gallery and are viewed like personas for user stories.

In this thesis, the author presents as a developer, artist, and designer in one; therefore, team formation was set aside. For a full-scale game/application production, a development team would need numerous people such as artists-designers, sound designers, animators and managers.

3.2. Design Canvas

The second stage of GDLC is pre-production, where visuals of final goals define the further development and the framework is set up. Design documents, canvases, prototypes, and flow management are typically used for development management.

3.2.1. Design Canvas

Developers create a canvas, a logical structure and a well-visualised table as a guide that is updated and changed every production cycle. Design canvas (Table 1) summarises the goal, the target group, market research, feature list, commercial potential if such present, and briefly describes /shows the application style [13]. In this work, the author has paid more attention to some canvas elements, such as goal and market research, to understand the application vision better and described them earlier in chapters **2.1**, **2.2** and **3.1** and further in section 3.2.2**.**

<table>
<tr><td>Target Group:
Males and females at age 12 to 50
An interest in arts, Virtual Reality, and game development
Students and amateurs</td><td rowspan="2">Unique selling point:
Visit artists’ worlds on the other side of the painting.</td><td rowspan="2">Goal:
Create a room with pictures.
Pictures are portals to other worlds
Each world resembles the picture
3.1 Initiation - Idea description</td><td>Setting/Theme:
Art Gallery
Virtual Reality
Cultural
Educational</td></tr>
<tr><td>Market Research:
2.1 Previous studies
2.2 Similar ideas</td><td>Style:
3D
First player simulation
Painting imitation</td></tr>
<tr><td>Future options:
The ability for a user to modify the gallery in VR</td><td>Monetisation:
Not convenient in this scope</td><td>Feature List:
Smooth locomotion
Gallery Room
Rooms-paintings
Interactions
Custom assets</td><td>Mock-ups:
3.2.2 Prototyping – Paper Prototype</td></tr>
</table>

Table 1. The simplified version of Design Canvas [13].

3.2.2. Prototyping – Paper Prototype

A paper prototype is a simplified hand-drawn application version that allows goal visualisation. The application consists of two main parts: a gallery room with paintings and an interactive room of the painting (Illustration 6).

Numerous paintings can be displayed in the art gallery. Some of the paintings have interactive rooms implemented. Illustrations 6 show the painting from the view of visitors in the gallery room. By touching the painting, the visitor teleports to an interactive room.

When artists design a room, they also should come up with a spawning point: a point where a visitor ends up after teleportation. This point may resemble the same view the painting is showing. For example, the painting (Illustration 7) is a house window facing the street (Illustration 8). Stepping into the painting resembles stepping out of the window, for example, into the balcony. For this reason, the spawning point of the room (yellow circle in Illustration 8) is on a balcony.

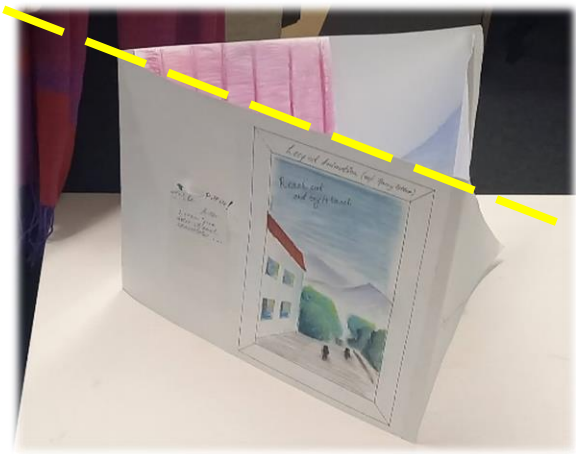


Illustration 6. A painting hung on the wall. Visitors only see the underline part.

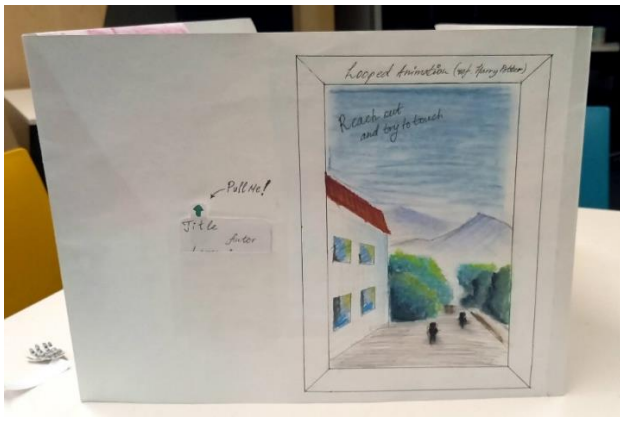


Illustration 7. Description of interactions and a closer look at the painting.

A visitor can walk around. The artist should also define the area within which visitors could walk. The walking area may limit to small or more significant areas; as with the upper example, the walking area could be restricted only to the balcony or allow the visitor to walk down the street. Also, the area should be restricted, considering the application's performance. A way back to the gallery room should be behind the spawning point (Illustration 9 blue rectangle). So if a room has many levels, there should always be a way back to the spawning point. For example, create a passageway to the rooftop to jump down to the balcony.

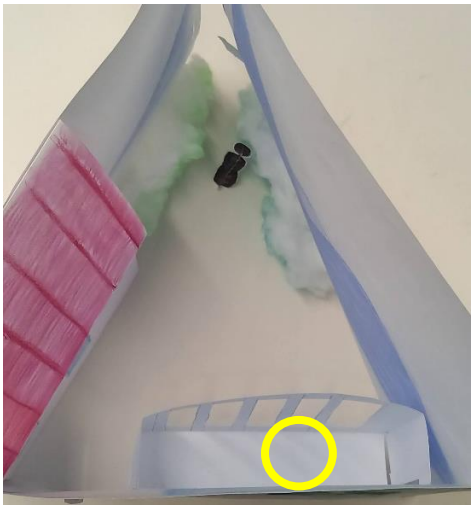


Illustration 8. The top view of the room.

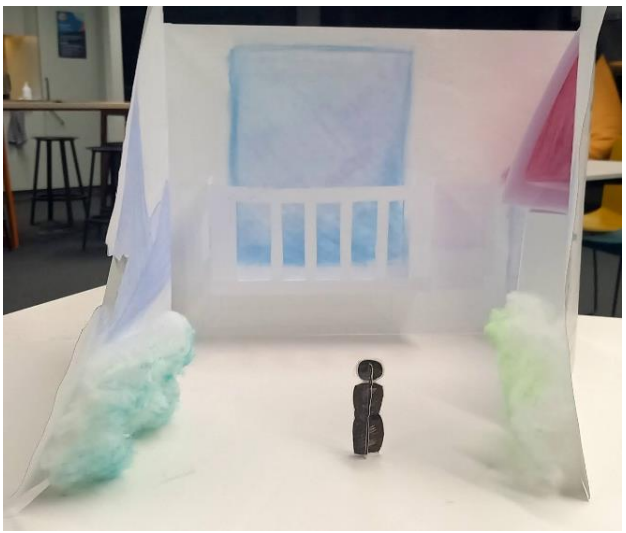


Illustration 9. Behind the spawning point.

For this project, the author chose the following paintings: the Night Library, Fire Flower, Space, Castle Swallow's Nest, and Mountains (Illustrations 10 -14 are listed in **Appendix** II). They differ in styles: abstraction, realism, pop art; - places: rooms, buildings, landscapes, space; - points of view: lower as a cat, upper as a bird and as a person. The main focus was on the library (Illustration 10.), the birds-view realistic room, and the fire painting (Illustration 11.), a person-view abstract space. Rooms based on these different paintings will show different room styles and configurations.

The library is closer to real-life experience than fire, so more detailed. The library painting show scenery from the bird's perspective; therefore, the spawning point should be aloft. The author put the Character's spawning point on the top of the bookshelf and added some floors to a room to make the way back to the spawning point from the first floor. The library painting shows the building; accordingly, the room's boundaries are walls. Therefore, the user can walk around the library's first floor, where are shadows, and on the second floor, on the top of the shelves.

On the other hand, the fire painting is viewed from the person's perspective and is an abstractive space type. For that reason, painting has only one plane and defines neither boundaries nor spawning point. The description of the rooms' creation process and asset development is handled further in chapter 4.2.

Illustration 10. The Night Library painting.

Illustration 11. Fire Flower painting.

3.2.3. Setting up the framework

Games and applications are developed inside frameworks. Assets are created with side tools or got/bought and then added to the framework. Alternatively, if the framework allows, assets are created inside it from the beginning. This project uses the following frameworks:

Unreal Engine 4 as the framework and Level Map asset design; Unreal Engine 4, further UE4 is a game engine, a framework for developing games and visual effects. UE4 creator, Epic Games, offers many useful plugins for VR development, implemented solutions, assets, and tutorials with best practices to make VR experience smoother. The base implementation is editable and customisable, so using this engine can save time. It is simple to use because it does not require technical knowledge of the C++ language. Instead of direct code writing, UE4 uses visual programming to compose code with nodes to develop small projects. UE4 is free-to-use, for no commercial purposes or with negligible revenue projects. This engine is popular for its ability to create advanced 3D games. At the moment of this project writing, the latest released engine's version was 4.27.

Blender for custom and complex assets with dynamic parts; Blender is a free 3D modelling and animation tool. At the moment of writing, the latest stable released version was 2.93. For development, default settings were used and additionally enabled the Node Wrangler add-on [14] to create procedural materials with nodes.

The University of Tartu has provided a Meta Quest 2 (Illustration 15) for this thesis and within the Virtual Reality course. The Quest 2's main advantage is that it is wireless and allows a user to move freely inside a Guardian Boundary (Illustration 15 blue line), which is customisable to the room [15]. To be able to try out custom projects, an Oculus/Meta developer account needs to be configured, and the developer mode needs to be enabled. During the first installation, Quest 2 requires installing the Oculus mobile application[3], which also provides a stream cast feature.

Alternatives for using UE4 were Godot[4] and Unity[5]; however, UE4 is the most artistic and advanced in 3D game development. Moreover, using other game engines would have required more time to learn these systems, as the author was more familiar with UE4. For Blender, there are no free analogues.

The Virtual Reality course provided the guide for the environment and framework configuration [16], both for Meta Quest 2 and multiple free engines (**Appendix** III). According to that guide, the author has set up the current project's environment. The author created a developer account, connected the Quest 2 to this, and enabled developer mode. Furthermore, configured preferences, build settings and plugins in Unreal Engine 4.27 needed for development and

[3] Oculus Mobile Application https://play.google.com/store/apps/details?id=com.oculus.twilight&hl=en&gl=US
[4] Godot https://godotengine.org/
[5] Unity https://unity.com/

connection to the Quest 2 headset. Also, connected the project to the Version Control System, Git and GitLab[6] platform as it allows to connect Large File Storage, LFS, for free and has templates for managing UE project versioning. U

The Virtual Reality template provided by UE4 formed the base of this prototype, as it already featured teleport locomotion, snap turning and grabbable/interactive objects. This leads to implementing the following main requirements: different types of smooth locomotion, rooms with painting-teleports, interactive or static custom objects to recreate the painting in depth, and level design.

Illustration 15. Meta Quest 2 Virtual Reality Set and its use [15].

[6] GitLab https://about.gitlab.com/

4. Implementation

The third stage of GDLC is production. The production phase is where the project idea and design are implemented. The development of Virtual Reality applications includes implementing in-room movement, objects and how a player can interact with them, and room layout.

The project templates of UE4 provide a ready-to-use default application. The Virtual Reality template contains one Level Map (Illustration 16) with two connected rooms, static and dynamic objects with default materials, and the first-person Character setup for VR application with three locomotion techniques and the ability to pick up dynamic objects.

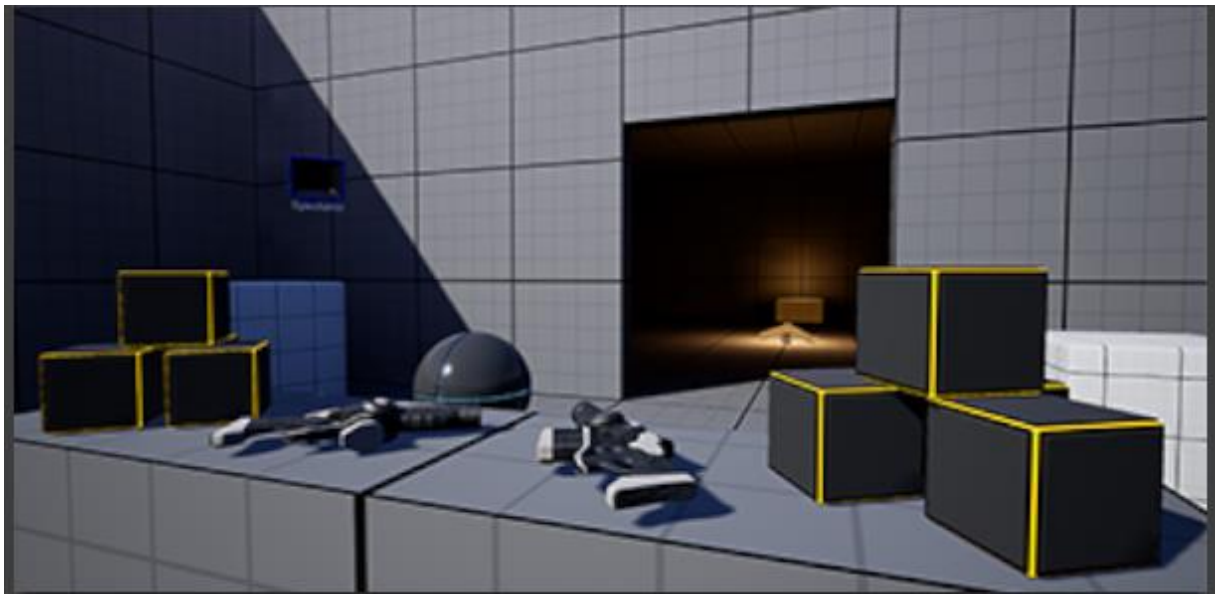


Illustration 16. Default Level Map from VR template.

4.1. Locomotion

Locomotion implementation is the most significant part of Virtual Reality application. The experience the user will gain using a Virtual Reality headset depends on how well the locomotion was done. Locomotion type is usually chosen depending on the environment in which it will be used and how it will affect the user's perception. Locomotion has numerous techniques, and researchers classified them into the following categories: posture, speed, energy-requiring, nausea causing, embodiment, intuitiveness, accessibility – and purposes: magical, presence breaking, off the ground and for video gamers [17]

In smaller areas, usage of faster types may cause nausea/motion-sickness as the environment around the user will change too fast. Faster types are more suitable for landscapes where the eye cannot detect how fast the movement is. Examples of faster types are elevator, teleportation and joystick. Slower types are more accurate and energy-consuming as they provide a smoother experience from locomotion. The closer-to-life examples of slower types are room-scale, grab and pull as pulling a rope, and walk-in-place.

Some locomotion techniques were already implemented in the project VR template: teleportation with preview, snap-turn and room position tracking supported by Meta Quest 2. Teleportation with a preview point of landing (Illustration 17) is a medium type of locomotion that rarely causes nausea. The application contains multiple room sets with different amounts of details. So to get a better understanding of what type of locomotion will suit the gallery, the author decided to add two more techniques: joystick and grab-n-pull.

Illustration 17. Teleportation locomotion usage, after press “A” appears following line with preview circle were to teleport.

4.1.1. Collision

In the template, locomotion is implemented in the VRPawn actor’s blueprint. The Pawn is the base class of objects that could be put into the scene. Although it is a physical representation of a player with location and rotation features, only the Character, a special type of Pawn, can move around because it comes with a CharacterMovementComponent, a CapsuleComponent that sets collision, and a SkeletalMeshComponent that can be visible in the game [18]. Therefore, before heading out to implement new locomotion types, it is necessary to implement room boundaries to stop a player from going through the walls (Illustration 18).

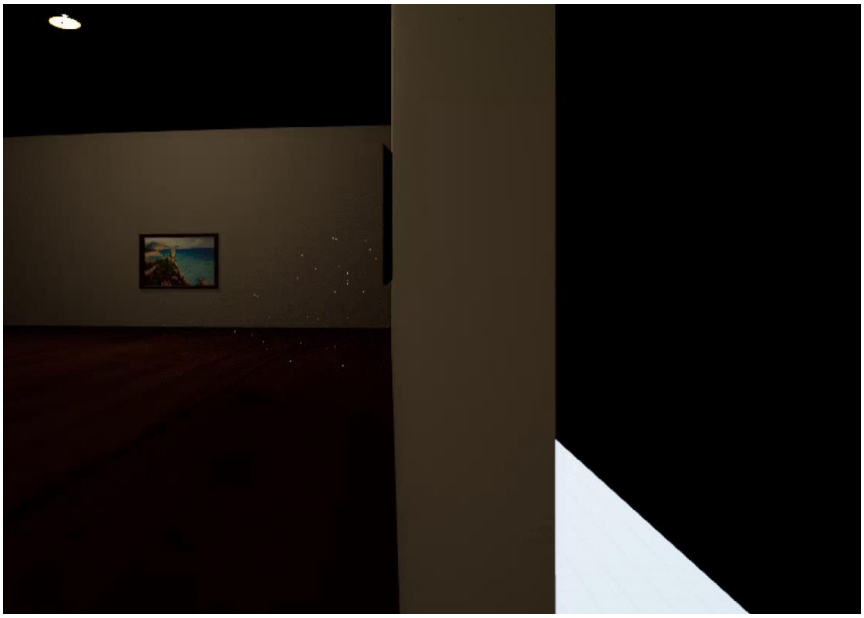


Illustration 18. Gallery Level Map. A Pawn with no collision inside the wall.

Firstly, in the VRPawn blueprint, the author changed the empty default scene root to the Scene component as a new root and class to Character. Now the player has a collision Capsule component and Character movement component (Illustration 19 side section). Then the new root to the bottom of the collision was adjusted (Illustration 19) to set the correct camera height based on the floor level. After that, custom functions were implemented to correct the collision height and bound together root and collision capsule positions. By doing so, the collision will always follow the camera, and the camera cannot go through the wall as a collision will block the static objects.

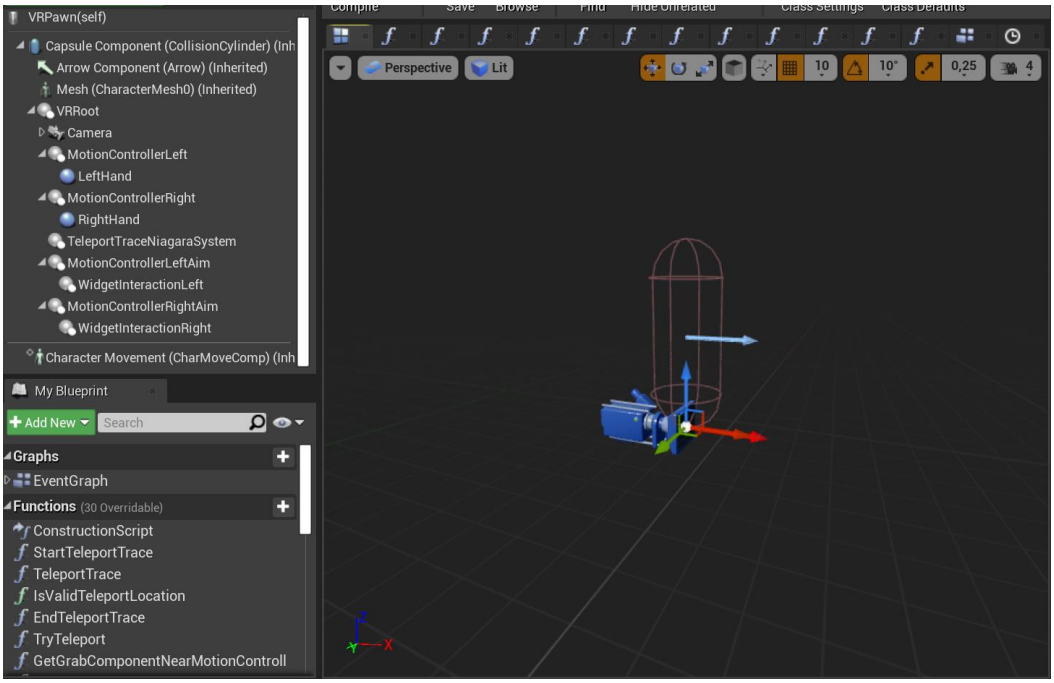


Illustration 19. VRPawn blueprint. In the right upper section - components and lower - global function. Camera at the bottom of Collision Capsule (red lines).

4.1.2. Joystick

The joystick technique is fast, frequently causing nausea, and a low energy-consuming type of locomotion better used in large spaces [17]. This technique is common for console gamers. The principle is simply pushing the controller's joystick and each frame while it is pushed, moving the player in the same direction at a certain distance.

Joystick locomotion requires the joystick input. However, teleportation was set on the right joystick input in the template. So the author changed the teleportation action trigger input to button "A" on the right controller and updated the input event node in the event graph of the VRPawn's blueprint. The joystick locomotion implementation consists of an input event and casting the mapped vector to the Character Movement Component. To not affect the movement in the game with the movement of the player's head, mapped the forward movement to the direction of the controller.

4.1.3. Grab-n-Pull

The grab and pull technique is slow and involves frequent hand moving; therefore, high energy consumption occurs [17]. Because a user controls locomotion input physically, this type does not cause nausea. The principle of actions: stretch out the hand, press the Grip button and on the hand backwards pull, update the player position, and release the button to fix the position. Moving direction is set to the point where the Grip button was pressed.

The VRPawn can pick dynamic objects, and this feature was set to the Grip button in the template. To leave this feature, the author added checks on button input if a pickable object is around. If not, then the custom event of locomotion made by the author starts. Firstly, it fixes the starting point of locomotion, the position of the controller where the Grip button was pressed. Then each frame, while the user presses the button, the event calculates the distance between the starting point and the current location of the controller and moves the Character Movement component accordingly at the same distance in the same direction of the starting point. On button release, the event stops.

These two locomotion techniques cover the little gallery and the bigger painting room spaces. With two default techniques, players choose locomotion according to a situation more suitable for them. This strategy will show the suitable locomotion for the extended gallery.

4.2. Level Design

Custom 3D objects were created with Blender to replicate the room interior and exported as FBX files to Unreal Engine 4. Some objects were imported from starter packs and the Epic Games asset store projects. The author has done models in Blender, some imported from default packs, and animated/customised them in the UE4 editor.

4.2.1. Gallery Room

A gallery is no matter what room where artists put on show their artworks. The Gallery Room is a simple rectangle room with hanging paintings that are still or animated. For this, a painting in the frame (Illustrations 20 and 21) was made in Blender. The materials of pictures were already added in the UE4 editor. Also, to make the visual difference between interactive

paintings, to teleports in UE4 were added some particle effects that resemble the painting's ambience: fire painting has ash particles (Illustration 22) and library dust particles (Illustration 23). Furthermore, with the dynamic material technique the blacking-out fading transition was made, which notifies the user that he has touched the painting.

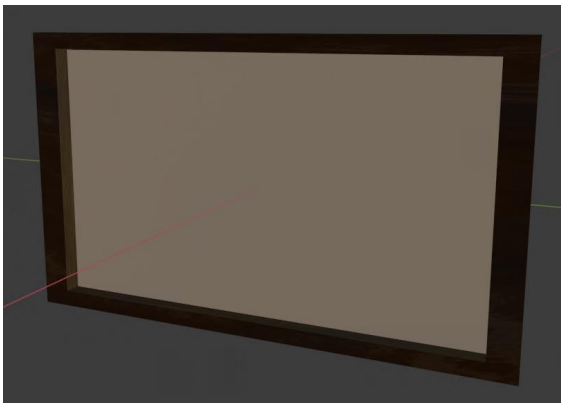


Illustration 20. Custom Blender painting model.

Illustration 21. Painting in Level Map with materials.

Illustration 22. Ash particles.

Illustration 23. Dust particles.

4.2.2. Library Room

The library painting room was difficult to create due to the number of objects. The lack of human resources influenced the creation process. Therefore, objects were grouped by behaviour and interactions with the player: living creatures that follow the player, ghosts who disappear by approaching, some interiors, and animations that create a library ambience.

The library in the painting is located in the old pavilion. The library building is a six-angled room with corner columns, three floors and a glass ceiling (Illustrations 24 and 25). The walls and ceiling were modelled in Blender. The floors with stairs were created in UE4 with Brushes for an opportunity to modify them on the go. The stairs were placed that way, so the user had to bow to get past the last stairs as the player was hitting the ceiling on steep staircases. This involvement puts the user to think about the surroundings.

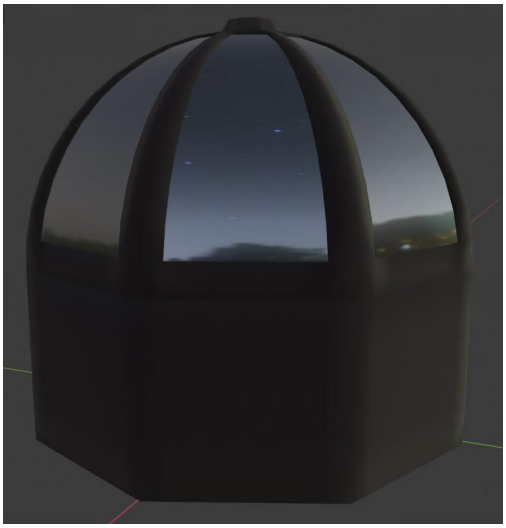


Illustration 24. Custom building: walls and ceiling.

Illustration 25. Floors and staircase.

In the room, in the centre, there are many bookshelves with books (Illustrations 26 and 27) made in Blender. The bookshelves were placed into the Level Map as a labyrinth (Illustration 28). An enormous amount of interactive books in one scene slowed down the performance. Therefore, they were added as static objects within the bookshelf. The lamp that the man in the painting is holding (Illustration 29) is styled aged antique, and illumination was added in the UE4.

Illustration 26. Custom bookshelf with books.

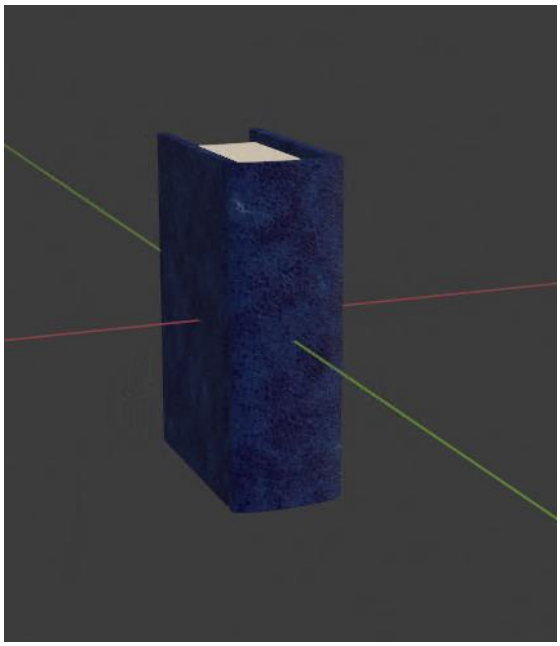


Illustration 27. Custom book.

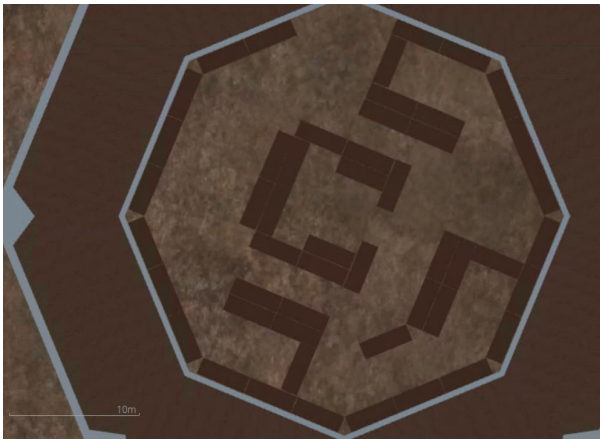


Illustration 28. Bookshelf placement in level.

Illustration 29. Static custom lamp**.**

The ceiling lamps (Illustration 30) and analogue materials are from the UE4 Mobile Starter Pack. Also, some nature assets were from the Unreal Engine marketplace from the free environmental asset pack "Procedural Nature Pack Vol.1" made by PurePoligons [19]. Animated trees created with splines were imported from this project to fill a visible room around the library building (Illustration 31).

Illustration 30. Default ceil lamp.

Illustration 31. Spline generated animated tree.

The dust effect was done with a particle emitter to create the ambience of an old room (Illustration 32). A dimmed volumetric fog effect was added to brighten the scene. The night sky is a sphere with inverted normals with the cubemap material of a night sky, generated in the 3D space tool provided by Rye Terrell [20] (Illustrations 33 and 34) and two simple spheres with emission effect as moons (Illustration 32).

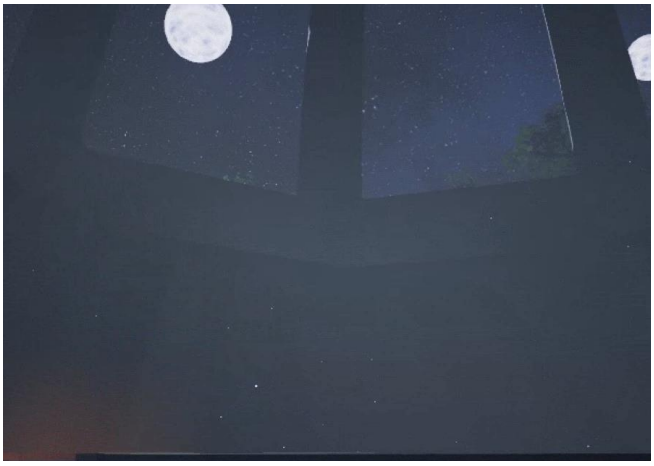


Illustration 32. Flying dust and fog effect and two moons.

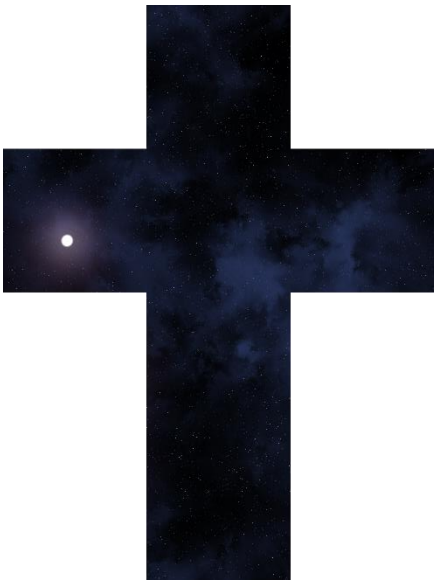


Illustration 34. Night sky cubemap.

Illustration 33. Night sky sphere with two moons

Man and woman shadow figures were styled in low-poly technique as only their silhouettes that will be visible from afar and disappear as the player approaches them close enough (Illustrations 35 - 37). That state is achieved with the use of the TriggerBox boundary. When a user enters the area covered by this box, the normalised distance between the ghosts and the user linearly affects the ghosts' opacity. The cat model is also done in a low-poly technique and has Skeletal Armature (Illustrations 38 - 40) for head rotation and tail wig animation. Animation is associated with the player's movement as the cat tracks the player.

Illustration 35. Custom Bender models of man and woman

Illustration 36. Custom man and woman ghosts in UE4 and from afar.

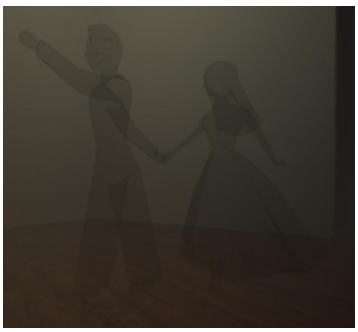


Illustration 37. Custom man and woman ghosts in a closer look during a game.

Illustration 38. Custom Bender cat models

Illustration 39. Custom cat in UE4 editor

Illustration 40. Custom cat during game

After entering the teleport, a player should occur on one of the bookshelves facing two moons and see ghosts, and the exit should be behind the player (Illustrations 41 and 42). The Character spawns at the top of the shelf, as the painting is viewed from the bird's perspective.

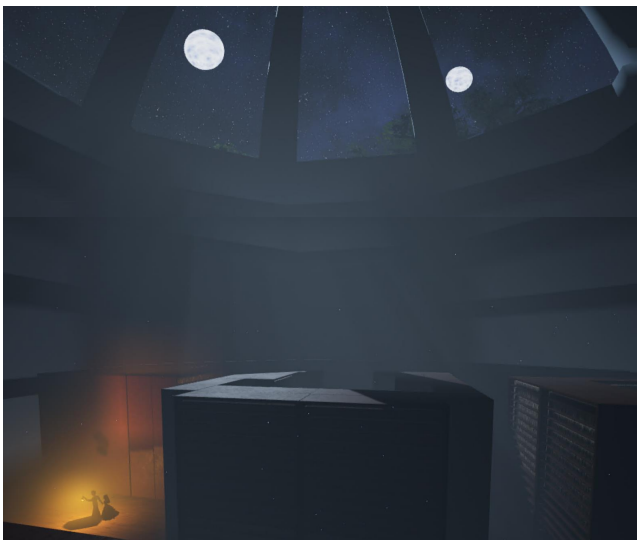


Illustration 41. A player sees this scene after Teleportation

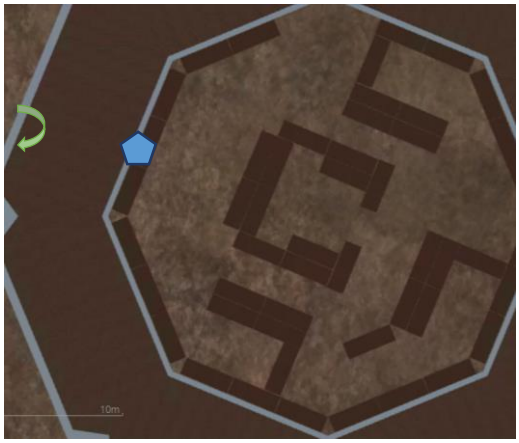


Illustration 42. Blue spawn point and green exit point placement in Level Map.

4.2.3. Fire Room

Fire Flower painting is a blazing fire. Therefore, the fire asset was generated from the particle system, where each particle is the default fire animation (Illustration 43). The system contains multiple fire leak animations and ash particles flying around. Also, spline generated particle effect as a blaze was added.

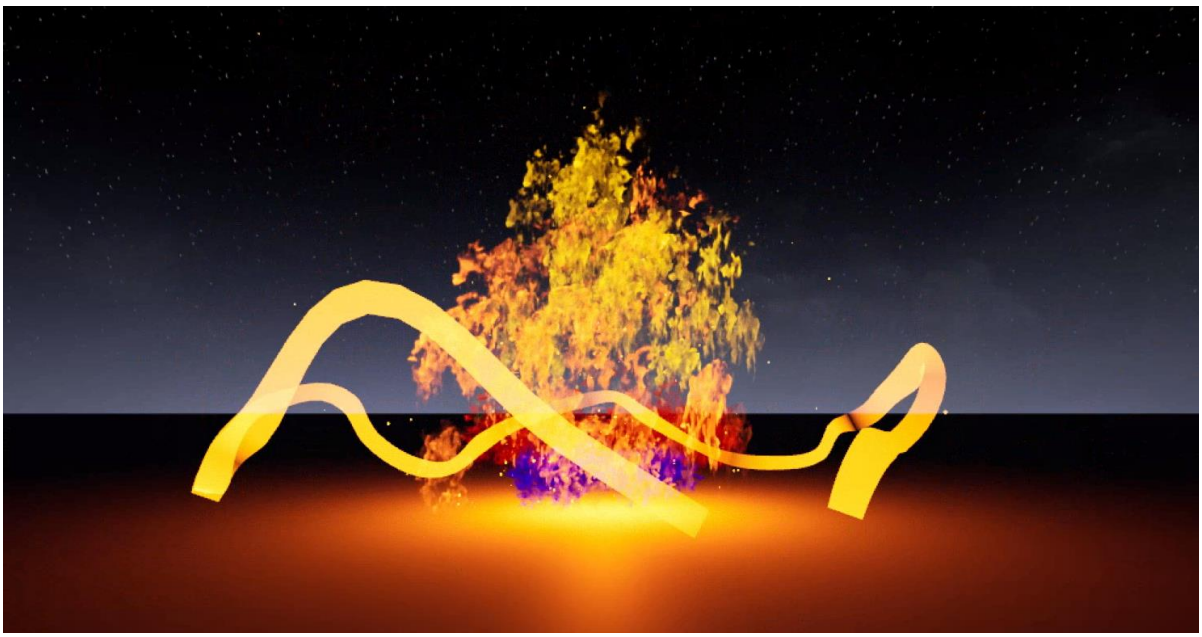


Illustration 43. Fire Animation with particle effect.

For all custom objects, the author had done custom procedural materials to recreate the oil paint style to objects with a node system in Blender. Unfortunately, the author found out too late that UE4 did not recognise these materials and had to find some closest analogues in starter packs. To pass the style of oil paintings to objects in scope author implemented a post-processing filter based on the Kuwahara filter[7] according to the tutorial on raywenderlich.com [21]. Then appeared that Meta Quest devices do not support post-process rendering. Therefore, the author failed to style the rooms.

As a result, paintings have formed into two 3D rooms with interactive and animated objects. The library room turned out to be spacious and contained different types of objects with interaction. The fire painting has been made simple with an enormous campfire animation and different effects. Although, the author failed to convey the oil painting style to rooms because of a lack of time and experience.

[7] Kuwahara filter https://en.wikipedia.org/wiki/Kuwahara_filter

5. Evaluation

Based on GDLC, after production, developers come to the testing phase, where developers search for bugs, exploits/glitches, and missing features and run on every chosen development platform. Also, testing with a user is essential for all software to have potential users' feedback. If something goes wrong, the team starts again from the pre-production step by analysing and complementing the game development document. The game was tested on students and some adults from the target audience to get the feedback on the design choices and whether the application implementation worked as expected for users.

5.1. Feedback gathering

The author conducted the testing sessions individually, mainly due to the lack of devices, and after testers gave their feedback using Google form (**Appendix** IV). The form is in two languages, English and Estonian. As Oculus mobile application provides a stream cast of what testers see, the author was able to give right away the instructions in some confusing situations. Also, with the testers' allowance, the author recorded a video of the testing session and, as a result, prepared short footage (**Appendix** V).

Virtual Reality is one of the newest technologies, and many testers, especially older testers, were unfamiliar with its usage. Therefore, at the start of the session, testers were informed how to use Meta Quest 2, about all three locomotion techniques, how to use them and how they are different. After a bit of practice, the author updated testers that they could try touching everything they saw. Testers knew that paintings were portals, although they did not know which ones. Then testers were asked to fill out the feedback form as the experience was fresh in memory.

5.2. Feedback analysis

Feedback was focused mainly on user experience in the VR gallery. In general, there were twelve questions, whereas nine were half- or fully open-ended questions, allowing testers to express their opinions on specific questions or about the experience in detail. To categorise the feedback based on the field of specialisation, the author distinguished the testers into four groups:

- artists – target group exhibiting their works;
- developers/designers - target audience knowing the concept of the application and noticing technical details; also could be in visitors' role;
- business people - may see the business potential and usage of the application;
- others, who represent as visitors.

In testing, ten testers participated from the target group. The testers rated the locomotion implementation. They all found that Grab-n-Pull locomotion was the most complicated to use because it was barely noticeable and had to move hands unreasonably often. Half of the respondents liked the Joystick locomotion the most because of the faster movement, despite all finding it the most motion-sickness-causing technique. In their opinion, the joystick was as easy to use as Teleportation locomotion, which was more sparing to the vestibular apparatus.

None of the testing team knew which paintings were teleports and which were not. To determine what attracts visitors' attention, the author asked the testers what painting they first noticed, which approached, and why. Most of the testers approached the first painting they saw as they entered the application: Mountains on the left, Castle in front and Cosmos on the right. Only a few, from the developers group, approached the Fire painting first because they noticed "bright fireflies". Therefore, the conclusion is that the particle effects did not work out as expected, attracting people's attention to paintings.

A few respondents, from the visitors group, found that the entered rooms did not fully match expectations about painting, like painting was brighter and more colourful. Nevertheless, most of those who answered the questions mentioned that they liked the ambience created in the rooms and the ability to try out impossible things, for example, "being on fire like witches from the Middle-Ages" and "jumping from shelf". Also, some testers were more impressed by the library room than the fire room. The main reason was that the library room is closer to real life than the fire abstraction aimed to create an emotional aspect. Therefore users could easily relate the room as real. The 3D recreations helped people understand the painting's meaning profoundly and the author's beliefs. "In the painting shelves did not feel enormous, but after entering I realised how tremendous they were", - pointed out one of the testers about the library room.

The testers reported that they liked the idea of an interactive gallery as something innovative and wanted to see more interpretations of the paintings. And not only presented in the gallery and author's other paintings but also famous masterpieces of grand artists like Vincent Van Gogh[8] and Ivan Aivazovsky[9]. Some testers also pointed out the lack of sounds and interactions with some objects in the scene, like the library books were static, which broke the presence of the experience.

The experience of the VR gallery in scope was positive, and artists have seen the opportunity and wish to create their galleries similarly. From a visitor's perspective, 3D rooms give a better and more meaningful overview of realistic paintings and abstract paintings' funny moods and emotional appeal. People associated with the business have pointed out monetisation possibilities such as free access and ordering custom assets for price.

5.3. Future Improvements

During testing, the testers found that when using teleportation locomotion in the library on the second floor, the user teleports to the first floor. Teleportation locomotion uses navigation mesh boundary volume for defining suitable places for teleportation. The problem was that this bound was automatically generated as soon as it was added to the Level. The recast mesh was not high enough to cover the second floor and was interrupted with artefacts on the stairs because of the steep rise and simple collision [22]. Therefore, it was frustrating to users that Teleportation was limited because they used it mainly in huge areas. Also, testers pointed out that bowing on the staircase is not intuitive and suggested adding the yellow-black striker on the ceiling.

[8] Vincent Van Gogh https://g.co/arts/C7qxCiMUUVPTi3xv8
[9] Ivan Aivazovsky https://g.co/arts/XokZhUPnyQJBorCRA

Testers also suggested continuing with this project. Besides adding new rooms, some testers would like to see more details in already existing rooms. One of the testers suggested including sculptures in the gallery that will move and interact with visitors.

The author sees the future of this application as a tool or sandbox-type game, enabling artists to create and edit rooms independently without developers' help. Therefore, it is necessary to implement the creative tool and cross-platform features to modify the gallery, access it on desktops and other VR devices, and expand the application usage. Moreover, the application could be elaborated with multi-player features in the future. Further, this project could be used for educational, scientific or commercial purposes.

The application focuses on the audience that uses it. Therefore, the application needs to work before any other feature or additional content. The application made during this thesis allows entering the world of the painting. That means the further steps of developing this project are locomotion fixes, oil paint style dynamic materials import, implement an editing module and multi-player mode to show the gallery. Those steps are necessary to allow artists to create their worlds behind the pane and expose them in the gallery open to everyone. All other suggested features may be implemented afterwards according to purposes.

6. Conclusion

This bachelor's thesis aimed to suggest a solution meant to enhance the experience of gallery paintings. As a result, the idea was explained and visualised by a detailed prototype and rated by testers. A prototype application of the interactive VR art gallery was designed and introduced to the target audience. Numerous locomotion and immersion techniques were used to enhance and supplement a virtual gallery experience. The most important of them are interactive rooms that convey the ambience of the painting to a new dimension.

It was essential to have the feedback and opinion from the target group. Some testers were unfamiliar with Virtual Reality games, although they have explored new possibilities enthusiastically. Ultimately, they had to complete a questionnaire that recorded the tester's background and overall tested application experience. The application testing session footage has been recorded. Generally, the testers enjoyed the experience, but various shortcomings made their experience uncomfortable. Some of these issues got fixed during this thesis, such as teleportation and collision bugs. However, most of them will be improved in the future, such as materials that match oil paint, sound assets, and locomotion overall fixes.

This project has numerous possibilities and fields for future development. In the author's opinion, the most significant features of the virtual gallery application are the creative tool, allowing artistic users to modify and customise interactive rooms, cross-platform access and multi-player mode. This thesis made it possible to enter the world behind the painting and made one step forward toward new opportunities

References

[1] Jason Jerald; The VR Book: Human-Centered Design for Virtual Reality; ACM Books; 2016

[2] Downing F.; Time and the Conservation of Paintings; 2005 https://www.buildingconservation.com/articles/conservofpaintings/conservofpaintings.htm (accessed 1.07.2022)

[3] Lin, Chih-Long; Chen, Si-Jing; Lin, Rungtai; Efficacy of Virtual Reality in Painting Art Exhibitions Appreciation, Clarivate Analytics Web of Science. Applied Science. 2020, 10(9), 3012; https://doi.org/10.3390/app10093012

[4] Estonian Virtual and Augmented Reality Association https://eevr.ee/estonian-vr-projects/ (accessed 1.07.2022)

[5] Maru VR, Tehtud Tööd https://maruvr.ee/tehtud-tood/ (accessed 1.07.2022)

[6] V&A; Curious Alice: the VR experience; 2021 https://www.vam.ac.uk/articles/curious-alice-the-vr-experience (accessed 1.07.2022)

[7] VIVERES, ViveArts https://www.vivearts.com/ (accessed 1.07.2022)

[8] Virtualmerse; Virtual Reality Art Gallery https://www.virtualmerse.com/blog/virtual-reality-art-gallery (accessed 1.07.2022)

[9] CoPpaces; How to Create a Virtual Reality Exhibition https://www.instructables.com/How-to-Create-a-Virtual-Reality-Exhibition/ (accessed 1.07.2022)

[10] Mozilla; Hubs; https://hubs.mozilla.com/ (accessed 1.07.2022)

[11] Lee I, 3D conception of Wassily Kandinsky's 'Orange' https://beljello.artstation.com/projects/mDeZOd?album_id=3434213 (accessed 1.07.2022)

[12] Futuclass. Education for schools in Virtual Reality https://futuclass.com/ (accessed 1.07.2022)

[13] Arvutiteaduse Instituut; Arvutimängude Loomine ja Disain; Design Document, 10.2021 https://courses.cs.ut.ee/2021/gamedev/fall/Main/DesignDocument (accessed 1.07.2022)

[14] Blender Manual, Adds-on, Node, Node Wrangler https://docs.blender.org/manual/en/2.93/addons/node/node_wrangler.html#node-wrangler (accessed 1.07.2022)

[15] Meta, Meta (Oculus) Quest 2 https://store.facebook.com/quest/products/quest-2/ (accessed 1.07.2022)

[16] The University of Tartu. Computer Graphics. Virtual Reality. VR Software and Project Setup. https://docs.google.com/document/d/1l0XKLSHwPn36urO8-Q5Mz9FxyD6NpLZZ2NTvu6k2AOQ/edit?usp=sharing (accessed 1.07.2022)

[17] Massimiliano Di Luca, Hasti Seifi, Simon Egan, Mar Gonzalez-Franco; Locomotion Vault: the Extra Mile in Analyzing VR Locomotion Techniques; Locomotion vault; 1-10. 2021, https://locomotionvault.github.io

[18] Epic Games; Unreal Engine 4.27.2 Documentation, Pawn https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/Framework/Pawn/ (accessed 10.07.2022)

[19] PurePolygons; Environments Produral Nature pack Vol.1; 22.07.2015 https://www.unrealengine.com/marketplace/en-US/product/procedural-nature-pack-vol (accessed 10.07.2022)

[20] Terrell R.; Space 3D, Generate procedural 3D space skyboxes; http://wwwtyro.github.io/space-3d/ (accessed 10.07.2022)

[21] raywenderlich.com, Tran T.; Unreal Engine 4 Paint Filter Tutorial; 1.05.2018 https://www.raywenderlich.com/100-unreal-engine-4-paint-filter-tutorial (accessed 10.07.2022)

[22] Epic Games; Unreal Engine 4.27.2 Documentation, Navigation system https://docs.unrealengine.com/4.27/en-US/InteractiveExperiences/ArtificialIntelligence/NavigationSystem/BasicNavigation/ (accessed 10.07.2022)

[23] Meta, What is the metaverse https://about.facebook.com/what-is-the-metaverse/ (accessed 10.07.2022)

[24] Stephenson, Neal (1992). Snow Crash (2003 paperback ed.). New York: Bantam Books. p. 24. ISBN 9780553898194, Accessed through Google books https://www.google.com/books/edition/Snow_Crash/RMd3GpIFxcUC?hl=en&gbpv=1&pg=PA24&printsec=frontcover . (accessed 10.07.2022)

[25] Irven K.; XR: VR, AR, MR—What's the Difference?; 31.10.2017 https://www.viget.com/articles/xr-vr-ar-mr-whats-the-difference/ (accessed 10.07.2022)

[26] Jaggo J.; Arvutimängude loomine ja disain: Pre-production; 2021 https://courses.cs.ut.ee/MTAT.03.263/2021_fall/uploads/Main/Lecture4 (accessed 10.07.2022)

[27] Blender https://www.blender.org/ (accessed 10.07.2022)

[28] Epic Games; Unreal Engine 4.27.2 Documentation https://docs.unrealengine.com/4.27/en-US (accessed 10.07.2022)

Appendix

I. Glossary

Conceptual terms

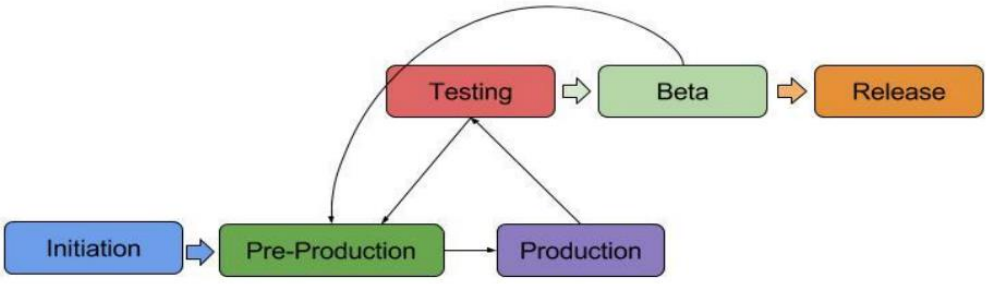
Art gallery - is a type of museum with themed exhibitions used to display or sell works of art.

Virtual Reality, shortly VR - according to Jason Jerald [1] - "..**virtual reality** is defined to be a computer-generated digital environment that can be experienced and interacted with as if that environment were real.". Also, the environment may contain experience-enhancing elements that can be similar to or completely different from the real world. Virtual Reality applications can be for educational, entertaining, or commercial purposes.

Metaverse – is an iteration of the internet as a single, social, universal and immersive virtual world accessible through technologies of Extended Reality. Originated from science fiction novel Snow Crash as "meta" and "universe". [23, 24]

Extended Reality, XR, stands for any form of computer-generated Reality: Virtual (V), Augmented (A) or Mixed (M). Simply X is a variable that can be any letter. [25]

The Game development life cycle, GDLC - According to the course "Game development and design" [26] is from of software development management. It usually consists of six phases that concentrate on game development and are different from the usual, including assets creation and team composition process. Drawing 1 is shown the flow of the cycle from initiation until release.



Drawing 1. GDLC phases.[26]

In the **initiation** stage, developers usually decide the idea of the game they will make, define a target audience, and compose the team based on needs.

Before actual development, the **pre-production** phase starts. The team creates a game development document, shortly GDD, by following which product is done. This stage includes (paper) prototypes and mock-ups to visualise the final goal and prepare frameworks and tools.

During the **production** phase, the development of visual assets proceeds, such as 3D models and animations, audio, level design, and source code.

After production comes the **testing** phase, where developers search for bugs, exploits/glitches, and missing features and run on every chosen for the development platform. At this stage, if something goes wrong team starts again from the pre-production step by analysing GDD and following its instructions.

The **beta** stage is provided by third parties such as publishers, reviewers, or potential buyers.

Post-launch, or **release** stage, involves finding bugs, creating content and feature upgrades, and communicating with users.

Technical terms

Application is a self-contained, comprehensive program written for the end-user.

Assets include graphics (characters, environments, logos), background music, special effects and sound effects that may be created with special tools or use existing assets.

Bug is an error, flaw or fault in computer software that causes it to produce an incorrect or unexpected result or to behave in unintended ways

Blueprinting is the concept of a node-based interface used to define object-oriented (OO) classes or objects.

Creative Suite is a family of professional multimedia creation and editing tools.

Demo - from demonstration - a showing of the merits of a product or service to a prospective consumer.

FBX is a file format to exchange 3D assets.

Feature is an intentional distinguishing characteristic of a software item (in performance, portability, or—especially—functionality)

Framework is the supporting structure of the platform, software, hardware, protocol, etc. For object-oriented systems, object classes provide the user or program with a group of interrelated functions.

Git is software for tracking changes in any set of files, usually used for coordinating work.

Locomotion is movement or the ability to move a player from one place to another.

Mock-up is a prototype, usually low-fidelity, such as paper illustrations, screenshots, or simple configurations of screens with limited interaction.

Prototype is a preliminary type, form, or instance of a system that serves as a model for later stages or the final, complete version of the system. Can be used for analysis, design and evaluation

Sandbox is a video game or part of a video game in which the player is not constrained to achieving specific goals and has a large degree of freedom to explore, interact with, or modify the game environment

Version Control Systems, VCS, is a system for the establishment and maintenance of baselines and the identification and control of changes to baselines that make it possible to return to the previous baseline

Unreal Engine terms

Actor is any object that can be placed on a level.

Blueprint is a node-based interface to create gameplay elements from within Unreal Editor.

Brush is an Actor that describes a 3D shape, such as a cube or a sphere. Can be placed into a level to define level geometry.

Character is a subclass of a Pawn Actor that is intended to be used as a player character. The Character subclass includes a collision setup, input bindings for bipedal movement, and additional code for player-controlled movement.

Class defines the behaviours and properties of a particular Actor or Object in Unreal Engine.

Component is a piece of functionality that can be added to an Actor.

Collision is invisible and used to prevent Actors from passing through them.

Event is a node that is called from gameplay code to begin execution of an individual network within the EventGraph

EventGraph of a Level Blueprint contains a node graph that uses events and function calls to perform actions in response to gameplay events

Emitter is the particle generator

First-person Character

Level is a gameplay area that the developer defines. Levels contain everything a player can see and interact with, such as geometry, Pawns, and Actors. Sometimes referred as **Map**

Mesh is a piece of geometry that consists of a set of polygons.

Objects are the most basic class in Unreal Engine - in other words, they act like building blocks.

Particles are small pieces/meshes spawned/animated with an emitter and done for visual effects such as dust, snow, beams and others.

Pawn is a subclass of Actor and serve as an in-game avatar or persona.

Spline is at least two vector points in 3D space connected by a line.

Trigger Box is a trigger Volume that is programmed to cause events when an Actor enters or exits it.

Volumes are bounded 3D spaces that have different uses based on the effects attached to them.

Devises

Desktop Computer is a personal computer designed for regular use at a single location on or near a desk due to its size and power requirements.

Meta (formerly Oculus) Quest 2 is an all-in-one VR, a Virtual Reality head-mounted display and controllers developed by Meta Platforms (formerly Facebook Inc.) [15].

Head-Mounted Display, HMD, is a display device, worn on the head or as part of a helmet, that has a small display optic in front of one or each eye.

Developing tools

Blender is the free and open-source 3D creation suite. “Blender is a public project hosted on blender.org, licensed as GNU GPL, and owned by its contributors. For that reason Blender is Free and Open Source software, forever.” is the vision of Blender [27].

Unreal Engine 4, UE4 [28], is a game engine, a software framework for game development, created by Epic Games. This engine uses for a development blueprint visual scripting system.

II. Illustrations

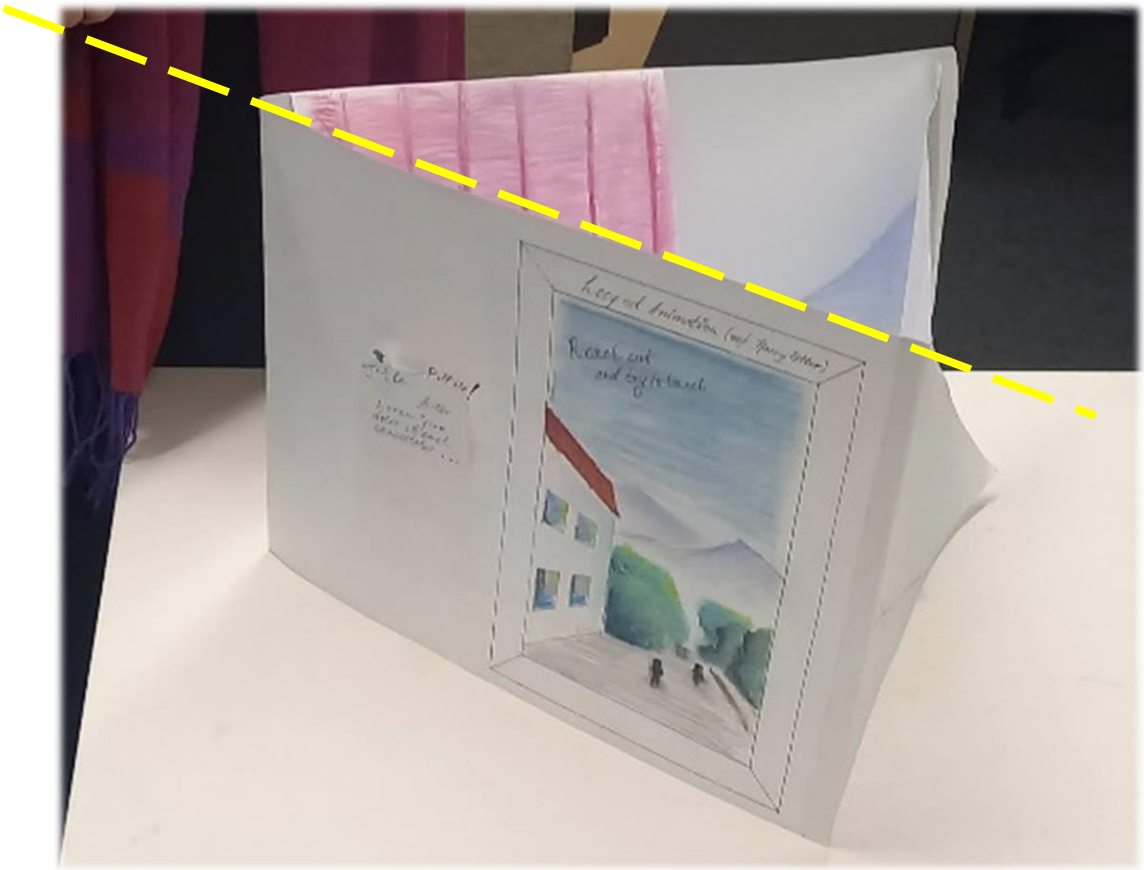


Illustration 6. Paper prototype. A painting hung on the wall. Visitors only see the underline part.

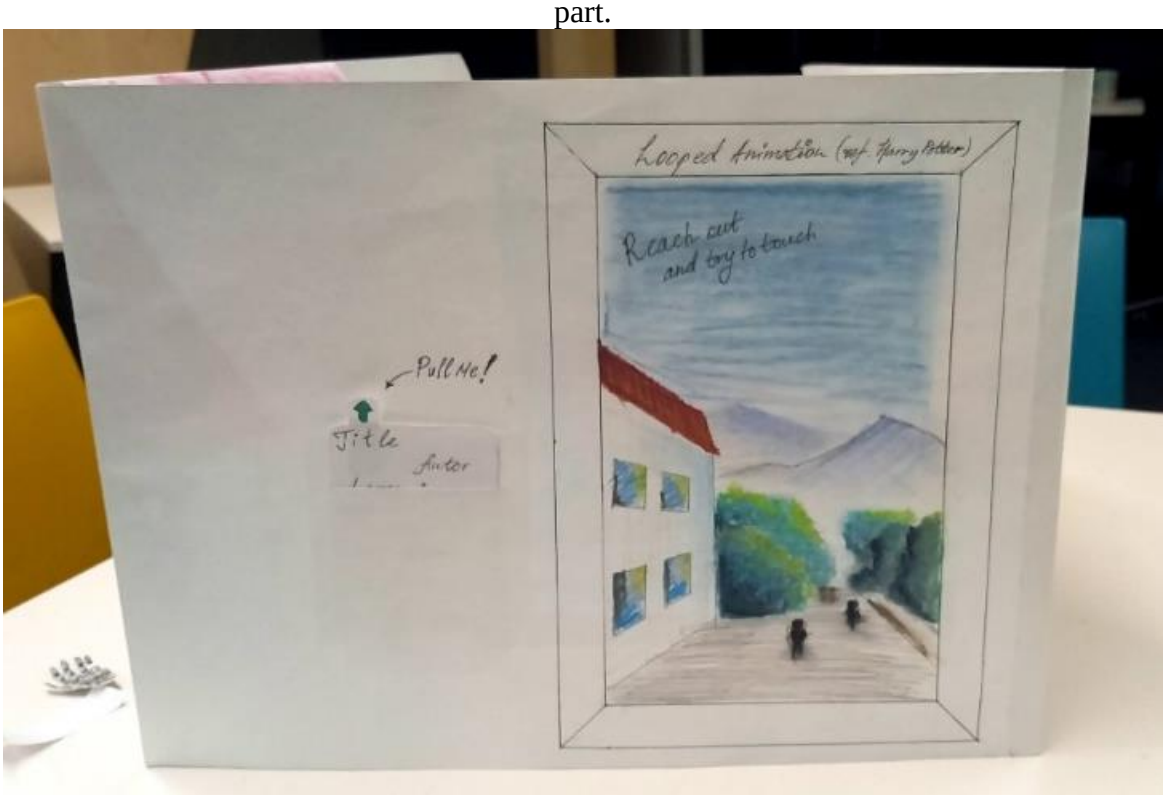


Illustration 7. Paper prototype. Description of interactions and a closer look at the painting.

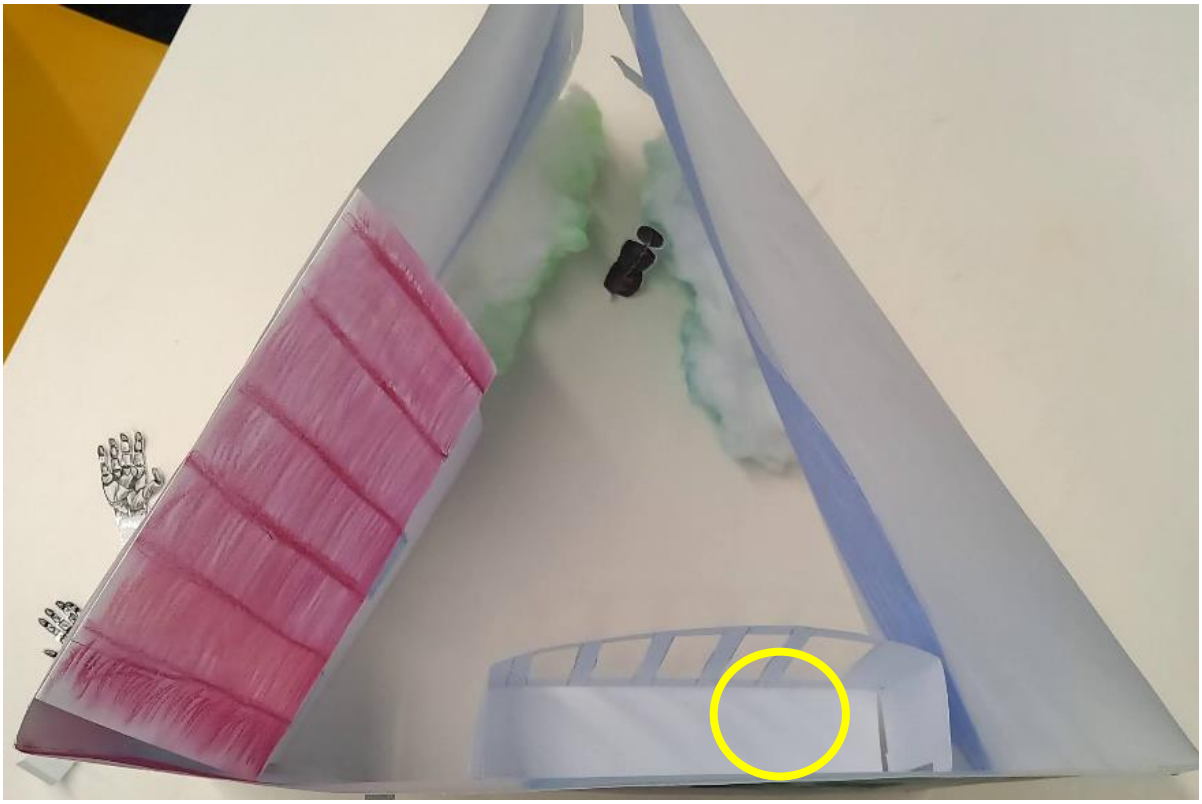


Illustration 8. Paper prototype.The top view of the room. Yellow circle spawning point

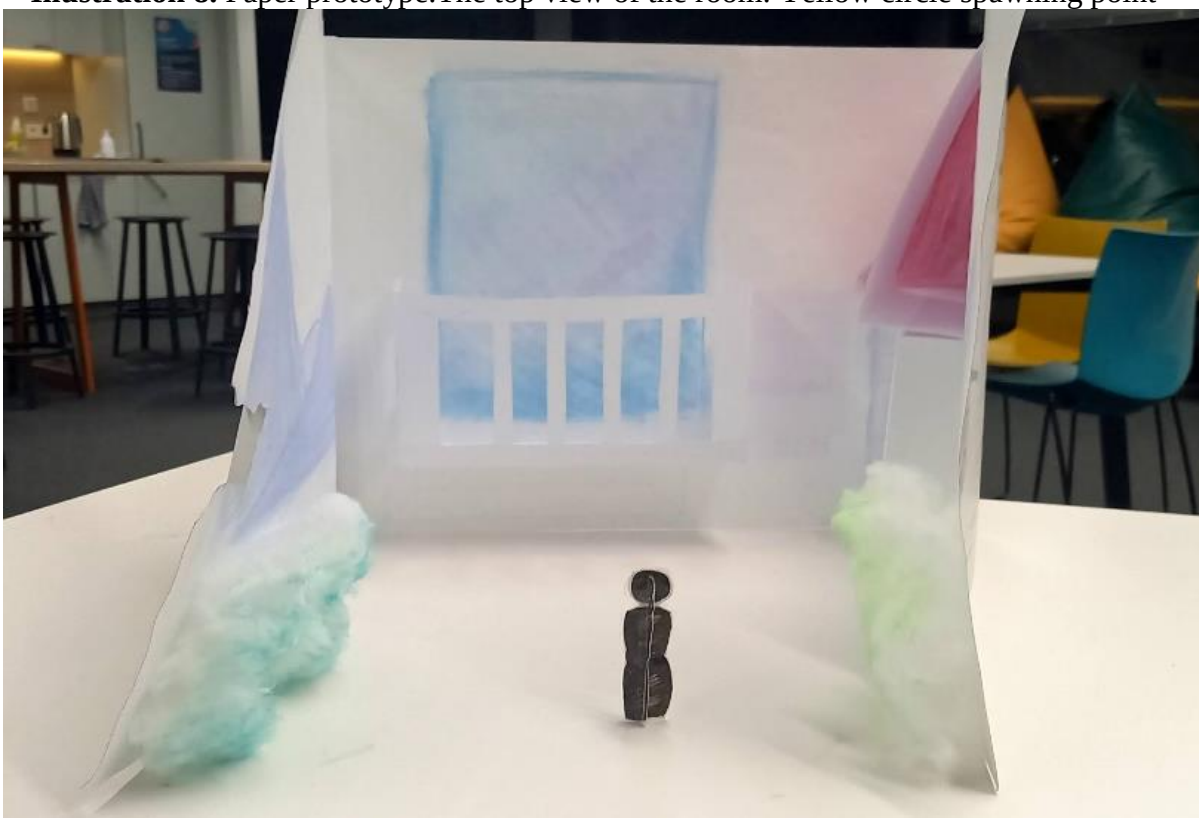


Illustration 9. Paper prototype. Behind the spawning point.

Illustration 10. The Night Library painting.

Illustration 11. Fire Flower painting

Illustration 12. Space painting.

Illustration 13. Swallow’s nest painting.

Illustration 14. Moutain painting

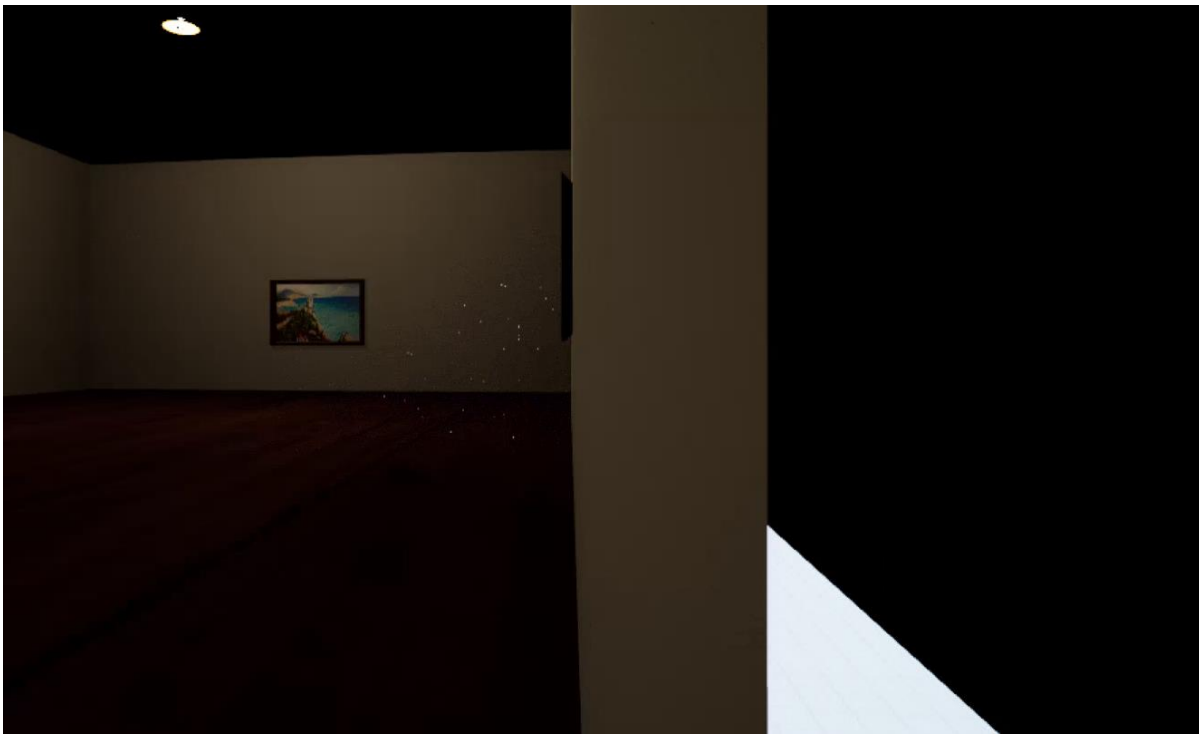


Illustration 18. Gallery Level Map. A Pawn with no collision inside the wall

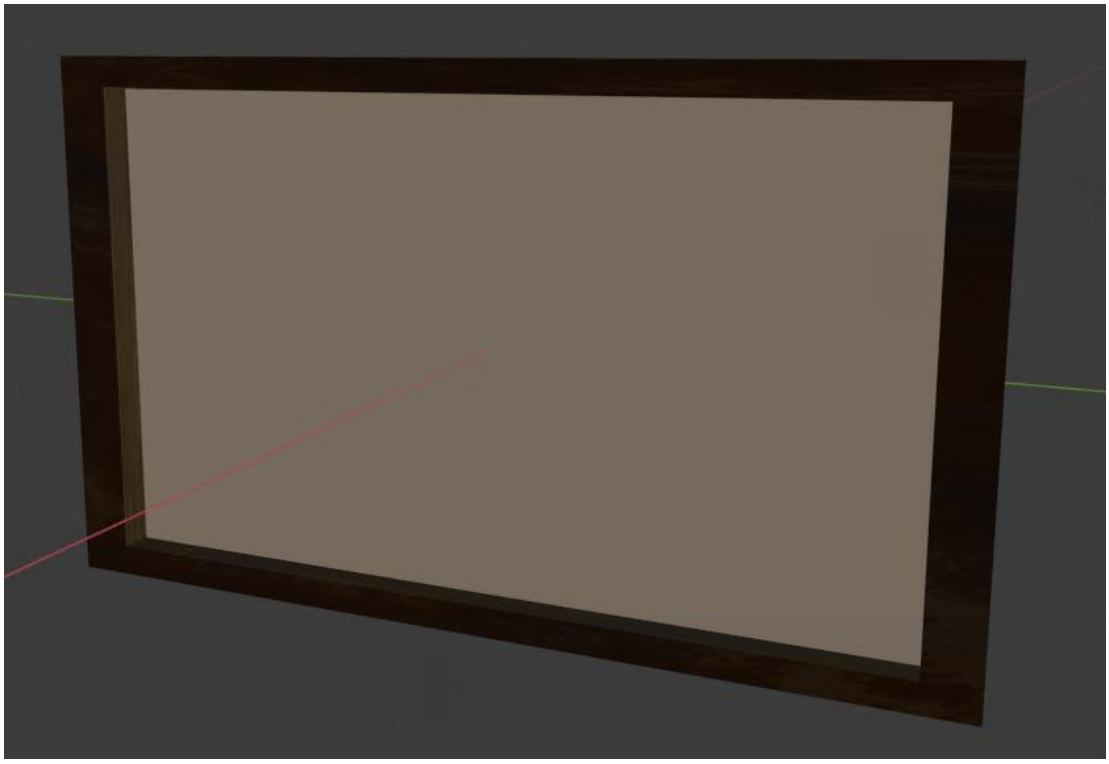


Illustration 20. Custom Blender painting model.

Illustration 21. Painting in Level Map with materials.

Illustration 22. Ash particles.

Illustration 23. Dust particles.

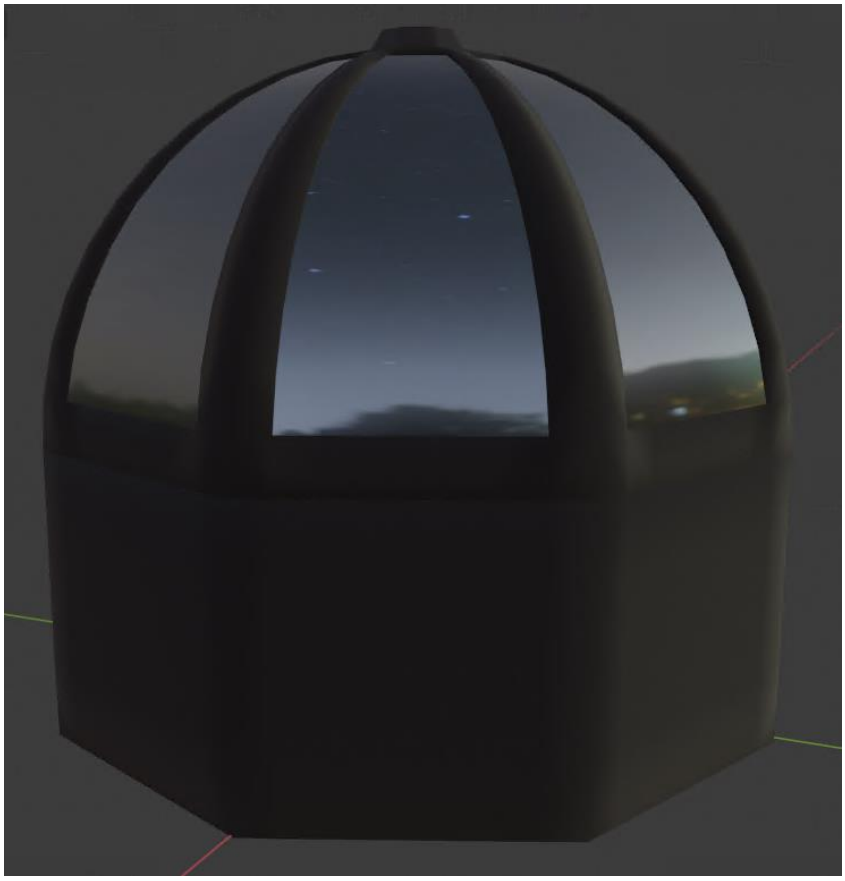


Illustration 24. Custom building: walls and ceiling.

Illustration 25. Floors and staircase.

Illustration 26. Custom bookshelf with books.

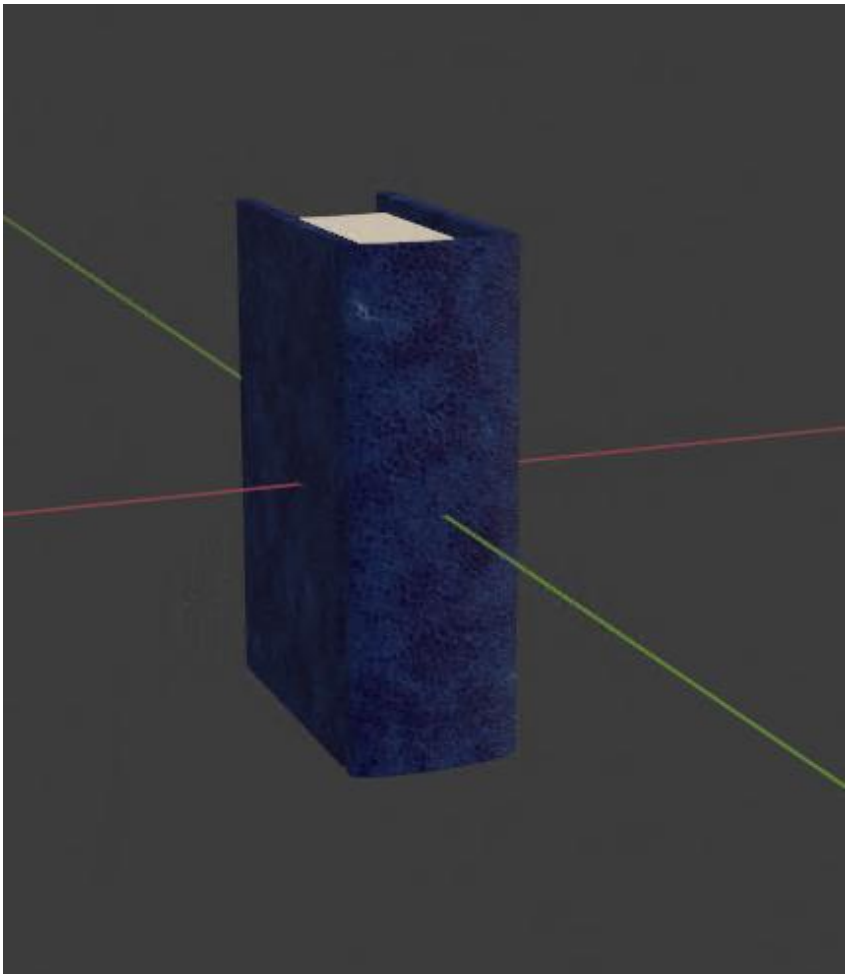


Illustration 27. Custom book.

Illustration 28. Bookshelf placement in level.

Illustration 29. Static custom lamp**.**

Illustration 30. Default ceil lamp

Illustration 31. Spline generated animated tree.

Illustration 32. Flying dust and fog effect

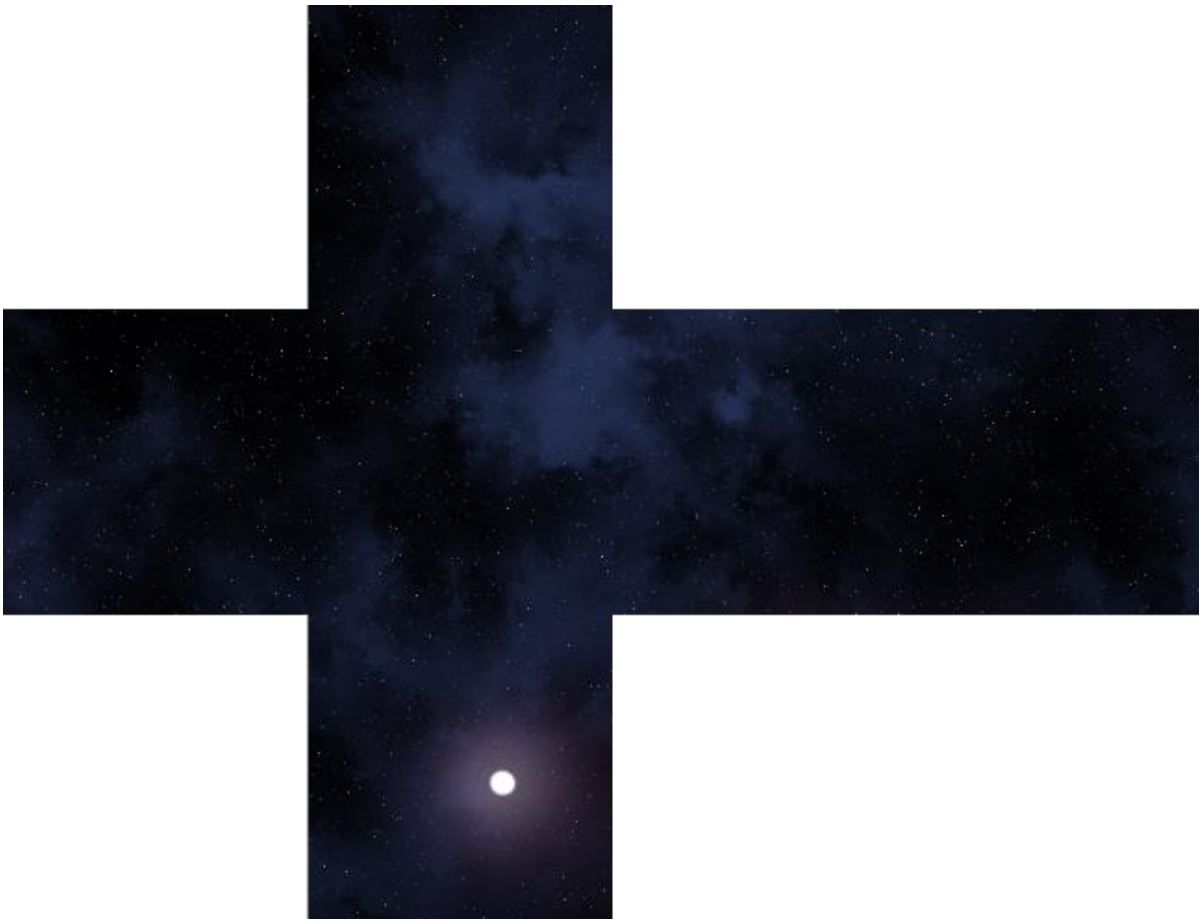


Illustration 34. Night sky cubemap

Illustration 33. Night sky sphere with two moons

Illustration 35. Custom Bender models of man and woman

Illustration 36. Custom man and woman ghosts in UE4 and from afar.

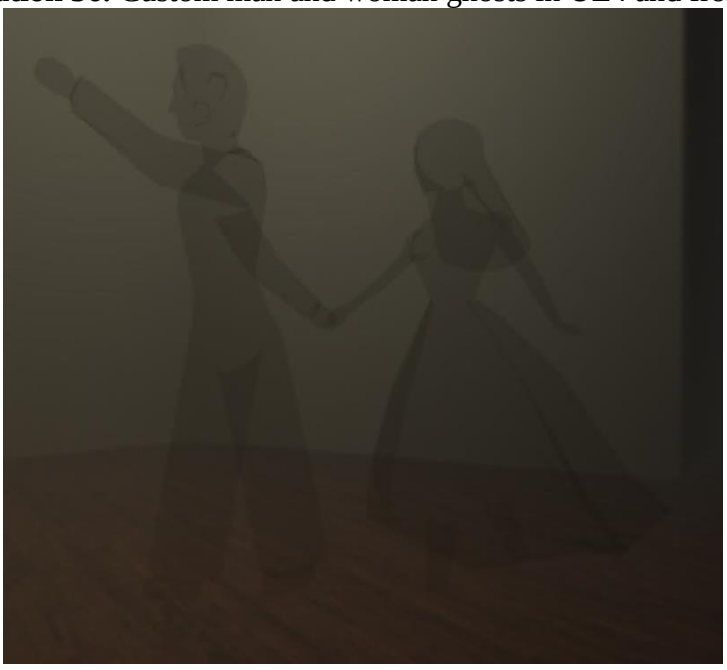


Illustration 37. Custom man and woman ghosts in a closer look during a game.

Illustration 38. Custom Bender cat models

Illustration 39. Custom cat in UE4 editor

Illustration 40. Custom cat during game

Illustration 41. A player sees this scene after Teleportation

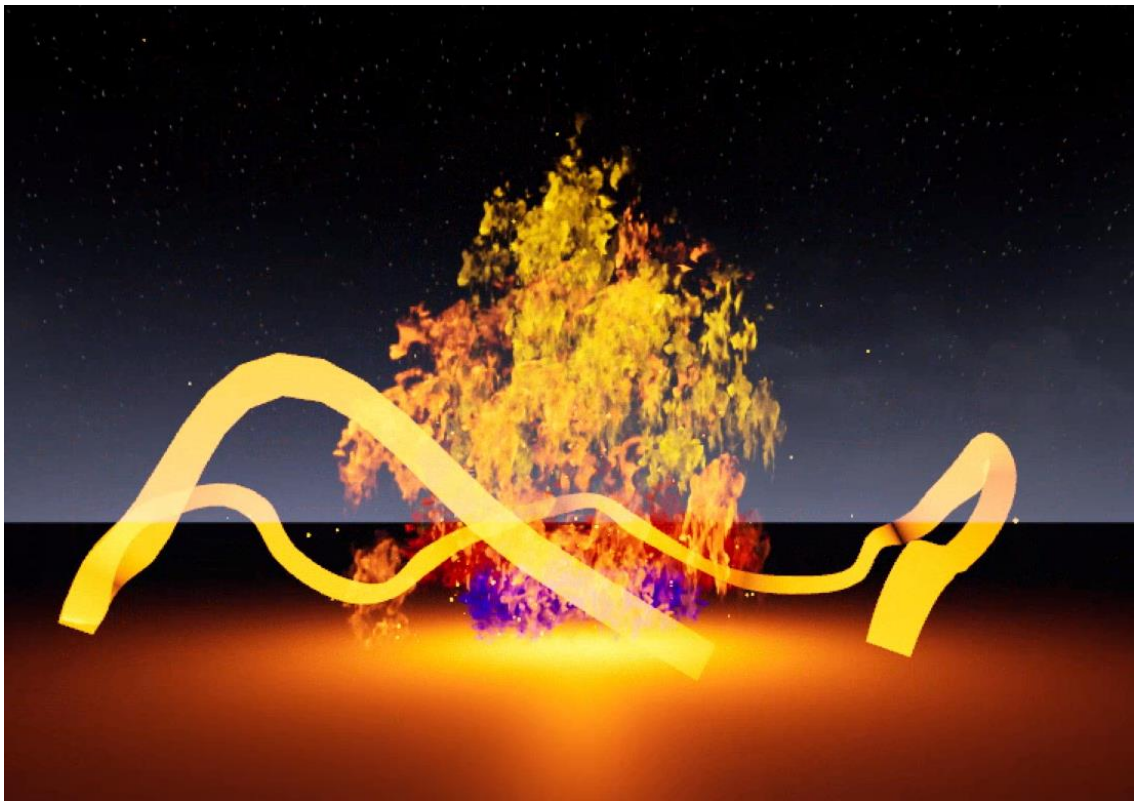


Illustration 43. Fire Animation with particle effect.

III. Repository and Installation guide

Repository with project: https://gitlab.com/jelizaveta_kuznetsova/bachelor-thesis-virtual-art-gallery

Installation Guide:

> **Oculus Dev Account and Quest setup**
>
> - Create a "verified" developer account here if you have not done so already
> - Also create an organization, the organization's specifics are unimportant when it comes to this course so don't worry about the name etc.
> - And Enable Development mode in the Mobile App for the specific Quest
> - Install Oculus ADB drivers on the PC
> - Oculus Developer Hub can also be a really valuable tool for debugging and managing Quest apps, Oculus SDKs, deploy over Wifi etc. -Make sure your ADB path matches the Androids Studio one in the settings! Something like: *C:\Users(UserName)\AppData\Local\Android\sdk\platform-tools*
> - Enable the USB Debugging Prompt in the Developer Settings Section on the Quest
> - Connect the Quest to the PC using the USB cable. In headset enable the USB Debugging Option
> - Set it to always allowed so that you don't have to agree to the prompt each time
> - You can check that the Quest is connected to the PC using the Oculus Developer Hubs Device Manager section or by typing adb devices into the Cmd or Powershell

source: The University of Tartu. Computer Graphics and Virtual Reality Lab. VR Software and Project Setup. https://docs.google.com/document/d/1l0XKLSHwPn36urO8-Q5Mz9FxyD6NpLZZ2NTvu6k2AOQ/edit?usp=sharin

Build and Package Projects Installation

1. From repository download the folder that is located in Build\Game\Android_ASTC\ to the destination where you would like your APK
2. Connect your Oculus device to your PC via USB
3. Check that the Quest is connected to the PC using the Oculus Developer Hubs Device Manager section
4. Under **App** section upload .apk file from downloaded folder
5. Launch project

Launching a Project Directly onto your Headset

You may also directly build and launch an application to your Oculus headset without saving the APK locally. For this you need to download whole project through git or zip file and prepare enviroment according to this guide under **Unreal Engine** section.

Unreal Engine (Spring 2022)

Software Setup

These Instructions are based on the Unreal Android SDK and NDK Setup and Quest Dev setup for Unreal from Oculus with notes from personal experience. This is done on Windows.

- Install Unreal Engine 4.27.2 from The Epic Games Launcher
    - It is also possible to compile the Oculus Branch of Unreal Engine from source. We will be sticking with the Official Unreal versions for this course's practice session, but for some specific cases the Oculus branch could offer some benefits such as having a newer version of the Oculus SDK and tools integrated, special optimizations etc.
- Install Visual Studio 2019, you can get it from the Older Downloads
    - When doing the setup check the C++ support option ...
    - This enables you to package the project for deployment on Quest
- Install Android Studio.
    - While the latest release works. Downloading the recommended 4.0 version from the Archive avoids having to do more tweaks later on.
    - Leave the SDK and JDK paths as default
    - Set up the components in the Android Studio SDK Manager. And don't forget to tick the Show Package Details flag in SDK Platforms and SDK Tools tab to install the specific versions of those components!
    - For UE 4.27.2 the following Android SDK components are needed/used:
        - Android SDK platform 29 (Android Q) in the SDK platforms Section.
            - No need for source or System Images!
        - SDK Build-Tools 30.0.2 though the default used is 28.0.3
            - If you are using a newer version of Android Studio remove the newest version of the Build tools 32.1.0-rc1 and 32.0.0 as some files are changed up there and your Unreal project will fail to build. There seems to be a workaround though.
        - Don’t Install NDK through Android Studio let the SetupAndoid.bat do it
        - Android SDK Command-line Tools (6.0) (Latest)
        - CMake 3.10.2.4988404
- Run SetupAndroid.bat from C:\Program Files\Epic Games\UE_4.27\Engine\Extras\Android or your equivalent Unreal Engine Install directory -If it fails (likely due to using newer version of the Android SDK components) you will need to modify it by changing the following line:

```
set SDKMANAGER=%STUDIO_SDK_PATH%\tools\bin\sdkmanager.bat
```

To

```
set SDKMANAGER=%STUDIO_SDK_PATH%\cmdline-tools\latest\bin\sdkmanager.bat
```

- Or get the modified .bat from here

- Might as well also run UpdateLinker.bat

Unreal Project Setup

- Select the VR template when creating your project
    - From 4.27 this is based on OpenXR, though the setup is still a bit green so expect changes and fixes in the future!
    - It has Forward Rendering enabled and some other VR specific optimizations set up already
    - First launch will have a long shader compile time (10s of minutes) if you have not launched a Forward Rendered project before

• In Project Settings I recommend that you check that the following options are selected (some of these optimizations/features can cause issues with other rendering features so it is good to keep them in mind)

- In Engine - Rendering
    - Mobile MSAA - 4x MSAA
    - If not using the Oculus VR plugin set the appropriate HMD Fixed Foveation Level
    - Instanced Stereo - ON
    - Forward Shading - ON
    - Mobile HDR - OFF (sadly have not found a working combination of settings to get this working on 4.27)
    - Mobile Multi-View - ON
    - Round Robin Occlusion Queries - ON
    - Auto Exposure - OFF
    - Anti-Aliasing Method - MSAA
- In Platforms - Android
    - Enable the Writing of Platform files
    - Accept the SDK License if it is not grayed out yet
    - Enable FullScreen Immersive on KitKat and above devices - True
    - Minimum SDK version - 23 or you could raise this to 27
    - Target SDK Version - 29 (might default to 28, that still works)
    - Switch the Arm Support target
        - Support Armv7 - OFF
        - Support Arm64 - ON
    - Only Support Vulkan Shaders and ASTC texture to reduce shader permutations and build times
        - Support Vulkan - On
        - Support OpenGL ES3.1 - OFF
        - Can also disable Include ETC2 textures for Multi Texture Formats, but we'll be only building the ASTC target anyway

• To get better access to Quest specific Features disable the Oculus OpeneXR plugin and enable the older OculusVR plugin

    - This will likely not be needed in future engine versions

    - It will offset the weapon grab rotation currently
    - Adds access to the ability to change target framerate (this is a BP node; others are also settings in the OculusVR Plugins Section of Project Settings), custom foveated rendering options, CPU and GPU performance levels etc.
- To see a better preview of how the visuals will look on Quest click on the Main Toolbar→Settings→Preview Rendering Level→Android Vulkan option to enable the Vulkan preview in the Editor viewport
    - This is a significant shader compilation wait time
- If you are getting some hard to debug errors a thing you can try is to delete all folders from the project other than the Config, Content and Source folders.

Deploying the project on Quest

...

- Launching the project on device:
    - Select the Drop down arrow next to the Launch button in the Toolbar and if you see the Quest Android Device click it to start the > packaging and launching process
        - If you do not, select ProjectLauncher option from the dropdown next to the Launch Button and find your Quest from there
        - After you have deployed once the Quest should become the default Launch option and you can just click the Launch Button itself
    - There is another significant wait here while the textures are prepared and shaders built
    - Once the process finishes you should see the
        - You can see the progress or errors in the Output Log which can be opened from Window→Developer tools→Output Log
    - You will see a Running YourAppName on Quest_2 …in the lower right corner of the Editor once the app has launched on device
        - Exit it from Quest or click Cancel to stop the app to be able to Launch again
- Once you have deployed ounce you can also find the app in the Quests Libraries Unknown Sources section

source: The University of Tartu. Computer Graphics and Virtual Reality Lab. VR Software and Project Setup. https://docs.google.com/document/d/1l0XKLSHwPn36urO8-Q5Mz9FxyD6NpLZZ2NTvu6k2AOQ/edit?usp=sharin

IV. Feedback form

Demo "Virtual Art Gallery Application" testing feedback

This survey was conducted by Jelizaveta Kuznetsova as part of bachelor's thesis. It aims to gather feedback on the demo application created by the author. Responding takes 5-10 minutes and answers are used only in the scope of the thesis.
Thank you!

Selle küsitluse viib läbi bakalaureusetöö osana Jelizaveta Kuznetsova. Selle eesmärk on koguda tagasisidet autori loodud demorakenduse kohta. Vastamine võtab aega 5-10 minutit ja vastuseid kasutatakse ainult lõputöös.
Aitäh!

* Required

1. Choose language *

Mark only one oval.

- ◯ Eesti *Skip to question 13*
- ◯ English *Skip to question 2*

English Version

2. Field of studies *

Mark only one oval.

- ◯ Technology
- ◯ Arts
- ◯ Buisness
- ◯ Other: _____

3. There were multiple locomotion solutions implemented. Which of the following statements applies to each type of movement?

Check all that apply.

	Joystick	Teleportation	Grab
Most complicated to use	☐	☐	☐
Easiest in use	☐	☐	☐
Liked the most	☐	☐	☐
Cuased the motion sickness	☐	☐	☐
Relieved the motion sickness	☐	☐	☐

4. In the first room (with hanging pictures), which painting did you approach first? *

Mark only one oval.

Mountains

Fire

Castle

Library

Space

5. Bethink, why did you choose especially this one, what has caught your eye? *

There were some paintings in the exhinition that have brought you to other worlds

6. Does the 3D world match your expectation/impression about painting? *

Mark only one oval.

- ◯ Yes
- ◯ No
- ◯ Maybe
- ◯ Other: _____

7. What do you like the most about paintings recreated in 3D rooms?

_____

_____

_____

_____

_____

8. What do you like the least about paintings recreated in 3D rooms?

_____

_____

_____

_____

_____

9. Please describe if and how the 3D recreation has affected your attitude towards painting

_____

_____

_____

_____

_____

10. Did the author manage to convey the mood of the painting in the 3D world? *

_____

There were other (ordinary) paintings in the exhibition.

11. What paintings would you like to experience in 3D, do not count those which you managed to enter?

Under other you may describe your own paintings or/and send files to author

Check all that apply.

- [ ] Mounatain
- [ ] Castle
- [ ] Space
- [ ] Other: _____

12. Comments and suggestions

_____

_____

_____

_____

_____

Eesti variant

13. Õppevaldkond *

Mark only one oval.

- ◯ Tehnoloogia
- ◯ Kunst
- ◯ Äri
- ◯ Other: _____

14. Demos rakendati mitmeid liikumislahendusi. Milline järgmistest väidetest kehtib liikumistüübi kohta?

Check all that apply.

	Joystick	Teleportation	Grab
Kõige raskem kasutuses	☐	☐	☐
Lihtsaim kasutuses	☐	☐	☐
Meeldis kõige rohkem	☐	☐	☐
Põhjustatud liikumishaigust	☐	☐	☐
Leevendanud liikumishaigust	☐	☐	☐

15. Esimeses toas (rippuvate piltidega), millisele maalile lähenesid esimesena *

Mark only one oval.

Mäed

Tuli

Loss

Raamatukogu

Kosmos

16. Mõtle järele, miks valisid just selle maali, mis on silma jäänud?

Nätusel esinesid mõned maalid, mis on toonud Teid teistesse maailmadesse.

17. Kas 3D-maailm vastab teie ootustele/muljele maalimise kohta?

Mark only one oval.

- ◯ Jah
- ◯ Ei
- ◯ Võibolla
- ◯ Other: _____

18. Mis Teile 3D-ruumides taasloodud maalide juures kõige rohkem meeldib?

_____

_____

_____

_____

_____

19. Mis Teile 3D-ruumides taasloodud maalide juures kõige vähem meeldib?

_____

_____

_____

_____

_____

20. Palun kirjeldage, kas ja kuidas on 3D-rekreatsioon mõjutanud Teie suhtumise maalisse

_____

_____

_____

_____

_____

21. Kas autoril õnnestus 3D-maailmas maali meeleolu edasi anda?

_____

Näitusel oli ka teisi (tavalisi) maale.

22. Milliseid maale sooviksite 3D-s kogeda, ärge arvestage neid, millele teil õnnestus siseneda?

Muu hulgas saate kirjeldada oma maale või/ja saata failid autorile

Check all that apply.

- [ ] Mäed
- [ ] Loss
- [ ] Kosmos
- [ ] Other: _____

23. Kommentaarid ja ettepanekud

_____

_____

_____

_____

_____

V. Video of Testing Session

YouTube link - https://youtu.be/Z0KZjvF6jkY

Google Drive link - bit.ly/3zFp32b
(https://drive.google.com/file/d/1KzdDvZjXxcJhFrhxLTVdD9isI9_w_7jt/view?usp=sharing)

VI. Licence

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

I, **Jelizaveta Kuznetsova**,

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 **Interactive VR Art Gallery: On The Other Side of The Painting;** supervised by Ulrich Norbisrath and Kärt Summatavet

2. I grant the University of Tartu the 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 from ***08/08/2022*** until the expiry of the term of copyright,

3. I am aware 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.

Jelizaveta Kuznetsova

08/08/2022