FACULTY OF MATHEMATICS AND PHYSICS
Charles University

MASTER THESIS

Shivam Sharma

Simulation of Symptoms of Mental Health Disorders in Virtual Reality

Department of Software and Computer Science Education

Supervisor of the Master Thesis:	Mgr. et Mgr. Iveta Hocko Fajnerová, Ph.D.
Study programme:	Visual Computing and Game Development
Study branch:	Computer Game development (IVVPA)

Prague 2023

I declare that I carried out this master thesis independently, and only with the cited sources, literature and other professional sources. It has not been used to obtain another or the same degree.

I understand that my work relates to the rights and obligations under the Act No. 121/2000 Sb., the Copyright Act, as amended, in particular the fact that the Charles University has the right to conclude a license agreement on the use of this work as a school work pursuant to Section 60 subsection 1 of the Copyright Act.

In .... date ....

Author's signature

I would like to begin by thanking my supervisor, Mgr. et Mgr. Iveta Hocko Fajnerová, Ph.D. without whom I would not be able to pursue such an unconventional topic for my master's thesis. She has been exceptionally patient as a mentor and her subject expertise has been pivotal for the development of this project. I am grateful to my parents and my sister for their love, support, and sacrifices in ensuring that I received the best possible education. I bow down to the divine for giving me strength and perseverance. I am thankful to my family and friends for always believing in me. And lastly, a big thank you to Arya, and Jyoti for always being my support system.

I dedicate this project to all the people who suffer from mental health disorders. The struggle of trying to explain what you're going through, the feeling of detachment, and being misunderstood by those around you. This project is a small contribution towards a world that is more empathetic, kind, and sensitive to those with mental health disorders.

Title: Simulation of Symptoms of Mental Health Disorders in Virtual Reality

Author: Shivam Sharma

Department: Department of Software and Computer Science Education

Supervisor: Mgr. et Mgr. Iveta Hocko Fajnerová, Ph.D., Center for Virtual Reality Research in Mental Health and Neuroscience,Department of Psychiatry and Medical Psychology of the Third Faculty of Medicine Charles University and NIMH

Abstract: The target of the thesis is to develop a software tool for the simulation of various symptoms of selected mental health disorders (e.g. depression, schizophrenia, etc.) using virtual reality (VR). This software will be applied as an educational tool to provide insight and increase understanding of symptoms in students studying psychiatry, clinical psychology or general medicine. Software development will be closely monitored by experts (psychiatrist, clinical psychologist) to ensure the quality and relevance of the simulation in means of the represented symptomatology.

Keywords: Mental Health Disorders, Virtual Reality, Learning Tools, Symptom simulation

Contents

Introduction **4**

1 Theoretical Background **7**

1.1 Mental Health Disorders 7

1.2 Bipolar Disorder 8

1.2.1 Symptoms of Bipolar Disorder 8

1.3 Schizophrenia 13

1.3.1 Symptoms of Schizophrenia 13

1.4 Computer Simulation 14

1.5 Virtual Reality 15

1.5.1 Immersion and Presence 15

1.5.2 VR and Cybersickness 16

1.6 Simulation in Virtual Reality 17

1.6.1 Simulating Mental Health Disorders in VR 17

2 Project Setup **19**

2.1 Hardware 19

2.1.1 Oculus Quest 2 19

2.2 Software 21

2.2.1 Game Engine 22

2.2.2 VR Software Development Kit 22

3 Analysis **24**

3.1 Simulating the Symptoms 25

3.1.1 Inner Speech 25

3.1.2 Locomotion 27

3.1.3 Visual Changes 28

3.1.4 Behavioural Changes 31

3.1.5 Visual Hallucinations 33

3.1.6 Audio Hallucinations . . . 35
3.1.7 Loss of Focus . . . 36

4 Implementation **37**
4.1 Unity Project Setup . . . 37
4.1.1 Folder Hierarchy . . . 38
4.2 Application Flow . . . 39
4.2.1 Landing Scene . . . 39
4.2.2 Day Scene . . . 40
4.2.3 Night Scene . . . 40
4.3 Architecture . . . 41
4.3.1 Settings Screen . . . 42
4.4 The Symptoms Class . . . 45
4.4.1 SymptomsSO - Configuring the Symptoms . . . 46
4.5 Extending the Symptoms Class . . . 47
4.5.1 Inner Speech . . . 47
4.5.2 Audio Symptoms . . . 49
4.5.3 Visual Symptoms . . . 50
4.6 Other Symptoms . . . 52

5 The Pilot Experiment **53**
5.1 Aim . . . 53
5.2 Target Group . . . 53
5.3 Study Design . . . 54
5.3.1 System Usability Scale . . . 55
5.3.2 Effectiveness of Individual Symptom Mechanics . . . 56
5.3.3 General Feedback . . . 57
5.4 Limitations . . . 57

6 The Result **58**
6.1 Studying the SUS results . . . 58
6.2 Evaluating Individual Symptom Mechanic . . . 61

6.3 Takeaways from the General Feedback 63

Further Scope **64**

Conclusion **65**

Bibliography **66**

List of Abbreviations **71**

Introduction

Virtual Reality has pushed the boundaries of immersive experience through its capability of producing life-like simulations. It enables anyone to participate in infinite experiences that would otherwise remain inaccessible. Although Virtual Reality has been used in a few fields such as some sectors in the military since the 1970s, technological advances have recently made the accessibility of VR affordable. Widely thought to be a breakthrough technology in the gaming industry, it has now also emerged as a powerful tool in many fields like education, medicine, tourism, entertainment and more. Today there is a strong global interest in widening the scope of VR applicability. This is because of the limitless expanse of subject areas that it can cover, the ease of accessibility it provides, and the benefits of active participation that the user can get.

This project and subsequent research are inspired by the way VR is creating new learning opportunities for medical professionals. Virtual simulation has been demonstrated to be an effective pedagogy that supports learning outcomes. Research has shown that virtual simulations can improve knowledge retention, clinical reasoning, and student satisfaction with learning [27].

The focus of this project is particularly on Mental Health Disorders. Far more than any other type of illness, mental disorders are subject to negative judgements and stigmatization. Many patients not only have to cope with the often devastating effects of their illness but also suffer from social exclusion and prejudices [28]. Stigma usually stems from ignorance, prejudice, or fear. There is also a general lack of awareness when it comes to mental disorders. This is owing to the fact that these problems often remain "all in the head". This implies that an onlooker would not be able to empathise with an individual who has some kind of mental disorder.

The ideal outcome is to use VR technology to illuminate the symptoms that a patient with mental health disorders would experience. This would become a learning tool for psychiatry and clinical psychology students as they learn to treat patients with a correct understanding of the ailment in question. It can also help cast away the prevailing prejudice and ignorance around mental health disorders.

VR simulation-based educational tools are not a new thing in this digital age. They have already been put to use in medicine, including psychiatry and clinical psychology. Studies have been carried out using VR-based mental disorder simulation but those are, most of the time, pre-recorded 3D videos. This has certain drawbacks including but not limited to:

- They offer little to no interactivity to the user.
- Personalising the experience for a particular user would be highly redundant as a new copy would be generated every time it is personalised for a user.
- Making even the slightest modifications to the experience means, rendering the 3D video again, which might involve filming the entire video all over again.

The goal of this project is to create a mental health disorder simulation, which is not only in VR but at the same time follows a video game approach. This would allow the user to control the character, regulate, and adjust the symptom parameters and have the first-hand experience of the disorder in focus. This can be easily extended to accommodate other disorders in the future as well as make changes to already existing simulations.

Thesis structure

The thesis is divided into six chapters. Chapter 1 briefly explains the theory and tries to explain all the major terminology that has been used throughout the thesis. This includes terms like VR, Computer Simulation, Mental Health Disorders, and associated symptomatology. Chapter 2 is mostly about setting up the project. The Hardware and software choices, the thoughts and reasoning behind the decisions as well as the pros and cons of the said choices. Chapter 3 analyses the symptomatology and the ideas, brainstorming, and decisions to bring those symptoms to the simulation. Chapter 4 touches upon the technical side of everything discussed in the preceding chapter, and how the ideas are implemented using the game engine. Chapter 5 outlines the experiment and Chapter 6 describes the results. The final section of the thesis is dedicated to the summary and the further scope of the research.

1. Theoretical Background

This chapter focuses on establishing the key terminology that will be used in the project. To ensure a thorough understanding of the subsequent sections of the thesis, it is crucial for readers to familiarize themselves with terms such as Virtual Reality, Bipolar Disorder, and Simulation. The chapter aims to provide clear and concise explanations of these terms, presenting them in a manner that is both accessible and informative without delving into excessive technical detail.

1.1 Mental Health Disorders

A mental health disorder, also sometimes referred to as mental illness, psychological disorder, and psychiatric disorder, is a disorder associated with the mental functioning of an individual. In the latest edition of the American Psychiatric Association's diagnostic manual, the DSM-5, it is defined as,

> A mental disorder is a syndrome characterized by clinically significant disturbance in an individual's cognition, emotion regulation, or behavior that reflects a dysfunction in the psychological, biological, or development processes underlying mental functioning. Mental disorders are usually associated with significant distress or disability in social, occupational, or other important activities. An expectable or culturally approved response to a common stressor or loss, such as the death of a loved one, is not a mental disorder. Socially deviant behavior (e.g. political, religious, or sexual) and conflicts that are primarily between the individual and society are not mental disorders unless the deviance or conflict results from a dysfunction in the individual, as described above. [3, p. 20–21]

The latest report by WHO mentions that in the year 2019, 1 in every eight people, or 970 million people worldwide, had some sort of mental disorder. And this number has drastically increased after the world went into a lockdown in 2020 [35]. Another study carried out in the Czech Republic during the Covid-19 pandemic found that there was an alarming rise in mental disorders such as Anxiety and Depression, while the risk of suicide also rose to three times the figures from 2017 [36]. Although modern medicine does make effective treatment possible, there is still a lot of social stigma and discrimination that patients have to face. This makes it so much harder for the patients and, at the same time, prevents people who feel they need help from getting a diagnosis. There are various types of mental disorders based on severity, physiological, and psychological symptoms. The focus of this thesis would be to implement symptom mechanics that are typical to mental illness, modify their characteristics, and combine them to mimic different disorders such as Bipolar Disorder (both mania and depressive episode), Psychosis, Depression etc.

1.2 Bipolar Disorder

In 2019, 40 million people were living with Bipolar Disorder. The nature of the symptoms that Bipolar Disorder entails, puts the patients at an increased risk of suicide [35]. This makes Bipolar Disorder a severe yet common illness. Symptoms of Bipolar disorder usually come in phases, which are discussed in detail below.

1.2.1 Symptoms of Bipolar Disorder

This section further discusses the symptoms experienced during the two phases of Bipolar Disorder. The symptoms and their description are described after thorough discussion with the experts, as well as carefully going through the relevant literature. The two phases of Bipolar Disorder are — Mania and Depressive Episode.

Mania

DSM-5 defines Mania as "A distinct period of abnormally and persistently elevated, expansive, or irritable mood and abnormally and persistently increased goal-directed activity or energy, lasting at least one week and present most of the day, nearly every day (or any duration if hospitalization is necessary)"[3, p. 124]. Following are the major symptoms associated with Mania (These are carefully studied from resources like DSM-5 manual[3, p. 124-125], NIHM[24], NHS etc. and briefly explained):

- **Increase of energy:** A person experiences a significant increase in their energy levels, leading them to pursue more challenging and ambitious tasks. Additionally, a sense of unpredictability characterizes their actions.

- **Faster Psycho Motor Tempo:** Psychomotor tempo refers to the speed and the coordination of an individual's physical movements, speech, and thought processes. A person may exhibit a significant increase in psychomotor activity in the form of restlessness, faster movements, and fidgeting.

- **Increased Self Esteem:** The person often overestimates their abilities and becomes ambitious. There is an increased sense of self-esteem.

- **Racing Thoughts:** The person is often bombarded with thoughts that vary drastically, and it is very hard to keep track of them. The person may feel overwhelmed by the sheer volume and speed of their thoughts, making it challenging to focus on one idea or complete a task effectively.

- **Pressured speech:** The direct consequence of racing thoughts is pressured speech. The person tends to speak a lot and quickly, sometimes jumping from one topic to another, in an attempt to keep up with their thoughts. This can make it difficult for others to follow the conversation or fully understand what is being said.

- **Decreased Need for Sleep:** The person seems to be able to function on significantly fewer hours of sleep. Three hours is more than enough, and often time they go long periods without any sleep at all.

- **Decreased Need to Eat:** Another major bodily need that the person overlooks is the need to eat. The person doesn't feel hungry and functions on a bare minimum diet. This can result in unintentional weight loss, and it can give rise to other health-related problems.

- **Impulsive judgement:** The sense of judgement becomes impaired, and the person tends to make impulsive decisions or conclusions. This often results in risky behaviours and bad decisions that may cause problems in the future.

- **Excessive Spending:** One kind of bad decision is to spend impulsively on things that the user can't afford, often putting them under huge debts.
- **Psychotic symptoms:** In severe cases of mania, a person may have delusions about their abilities (this also relates to increased self-esteem) and transcendental tasks (goal-direction). Sometimes the person also experiences hallucinations.
- **Distractibility:** The person gets distracted very easily. This may result in incessant switching between tasks and leaving the previously taken tasks unfinished. Attention is drawn to unimportant or irrelevant external stimuli too easily. This can also hinder the person from paying attention to what others are saying, rendering them unable to have meaningful conversations.
- **Extreme goal-direction:** The increased self-esteem and the inflated sense of their own abilities make the person set goals and targets which are way beyond their reach. The individual becomes intensely focused on pursuing specific goals, with excessive drive and determination.

Depressive Episode

The manic phase is followed by a period of normalcy but often times the depressive episodes. This includes abnormally low mood, energy, and activity levels lasting for at least two weeks. When going through this phase, the person loses any kind of motivation, becomes highly anti-social and loses interest/doesn't find any joy in most of the activities. The person may become suicidal and requires special attention and care. The major symptoms associated with the Depressive Episode include(as per DSM-5 manual[3, p. 125-126], NIHM[24]):

- **Depressed mood:** The person's energy is very low. There is a continuous feeling of sadness, emptiness, and hopelessness, and the person doesn't find purpose in anything.

- **Anger:** The person becomes very irritable. They become angry and might lose their temper at very minor things. This causes them to grow distant from family and friends, and the quality of their social life may deteriorate drastically.
- **Loss of Interest:** The person loses interest in most of the day-to-day activities. Activities that they previously enjoyed also don't bring joy any more. The person feels drained to engage in any sort of activity.
- **Guilt:** The person may have feelings of worthlessness and self-doubt, which can lead to a sense of guilt. They may feel guilty about things they have done in the past, sometimes even for the things of the least significance.
- **Hyper/Hyposomnia:** During the depressive episode, a person may experience either hypersomnia (excessive sleep) or hyposomnia (reduced sleep) or both. In cases of hypersomnia, the person may feel an overwhelming sense of fatigue and may sleep for prolonged periods of time. Whereas, in hyposomnia, the person may have difficulty falling asleep, staying asleep, or early morning awakenings. This can lead to feelings of restlessness, irritability, and fatigue during the day.
- **Change in Appetite, and Weight:** A person may experience a drastic change in their appetite that also has significant impacts on their physical and emotional well-being. The most noticeable change is gaining or reducing a lot of weight. In cases of appetite loss, the person may experience a reduced desire to eat, resulting in unintended weight loss and other physical health problems. Appetite gain, on the other hand, can lead to overeating, which can result in weight gain and associated health risks.
- **Fatigue:** The person feels extremely exhausted all the time. No amount of sleep helps with relieving the exhaustion. The exhaustion further fuels a lack of motivation and interest in daily activities. This impairs the person's ability to complete tasks, engage in social activities, and maintain relationships.

- **Concentration Impairment:** Concentration Impairment is another major trait of Bipolar Depression. The person may experience impairments in their ability to concentrate on tasks, remember information, and make decisions, which can significantly impact their daily functioning.

- **Self-loathing:** Self-loathing as a trait of Bipolar Depression that works in tandem with self-guilt. The person may form negative perceptions about themselves, they have feelings of guilt and a sense of hopelessness about their future. This can severely impact their sense of self-worth and self-esteem, contributing to low confidence and social withdrawal.

- **Thoughts about death:** Thoughts about death, which include suicide ideation, is a potentially life-threatening trait of depression. A person may experience a range of thoughts related to death, from fleeting thoughts of death or dying to more persistent thoughts of suicide. The continuous cycle of guilt and self-loathing contributes to the desire of the person to end their life.

- **Psychomotor Agitation or Retardation:** This trait is mostly related to the movement of body parts and thought processes. It can occur in two different ways, i.e. Psychomotor Agitation — excessive, unintentional, and purposeless physical movement or restlessness. The person may exhibit behaviours such as fidgeting, pacing, wringing their hands, or making repetitive motions. And Psychomotor retardation — slowed-down physical movements, speech, and thought processes. The person may struggle with starting or finishing tasks, exhibit slower-than-usual movements, and speak in a sluggish or monotone manner.

The many symptoms can be responsible for flaring up each other [19]. This creates a never-ending cycle that makes it difficult for people to carry out even simple day-to-day tasks. They start questioning themselves and the decisions they take or took in the past. As a result, their self-worth greatly diminishes, and they may feel more distant from their loved ones.

1.3 Schizophrenia

Schizophrenia is a serious psychotic disorder. It is explained by NIHM as,

> A serious mental illness that affects how a person thinks, feels, and behaves. People with schizophrenia may seem like they have lost touch with reality, which can be distressing for them and for their family and friends. The symptoms of schizophrenia can make it difficult to participate in usual, everyday activities, but effective treatments are available. Many people who receive treatment can engage in school or work, achieve independence, and enjoy personal relationships [25].

1.3.1 Symptoms of Schizophrenia

Symptoms of Schizophrenia (as discussed in NIHM[25], DSM-5 Manual [3, p. 99-100]) are:

- **Delusions:** This includes having strong beliefs about things which are not true or real. For instance, a delusional person may believe that random people are conspiring against them, or that they are being followed in public.
- **Hallucinations:** When an individual has a perception of something involving smell, taste, vision, or hearing, but that thing is not actually there, it is called a hallucination.
- **Disorganized Speech:** Disorganised speech involves the inability to complete sentences, speech derailment, and making illogical statements.
- **Catatonic Behaviour:** Catatonia is one of the most severe symptoms of Schizophrenia. Catatonia can range from disorganised behaviour to motor abnormalities which can result in a state of immobility, Mutism, repetitive behaviour, staring etc.
- **Thought disorder:** The individual is unable to follow the train of thoughts, organise them into something meaningful and use them to form meaningful conversations. This results in stopping mid-sentences, speech derailment, and making up words or sentences that don't have any meaning.

- **Negative symptoms:** Negative symptoms involve diminished emotional expression, withdrawal from social, withdrawal from activities that the individual once enjoyed, loss of interest, and avolition.

- **Cognitive symptoms:** Cognitive symptoms cause issues related to memory, attention, and concentration. The individual is unable to remember things, form memories, or participate in conversations.

People with mental disorders require love, support, and empathy, as these contribute to diminishing stigma and fostering a more inclusive and accepting atmosphere. Being treated kindly may result in reduced feelings of isolation and improved self-esteem for the affected person. Moreover, an empathetic approach from close ones can enhance treatment adherence and encourage open discussions about mental health challenges. Ultimately, empowering those with mental health disorders to effectively manage their condition and lead more satisfying lives.

1.4 Computer Simulation

A computer simulation can be defined as "the process of designing a model of a real or imagined system and conducting experiments with that model for the purpose either of understanding the behaviour of the system or of evaluating various strategies for the operation of the system" [30]. It is the process of virtually representing a real-world system or process using computers. Most of the time, these are done to study systems which are hard or impossible to create in real life. It enables the researchers to study the behaviour of these systems, perform tests and analyse their future outcomes. Due to the cost-effectiveness, and ease of implementation, simulations have an extremely wide set of applications ranging from studying the stock market to training pilots. Simulations are also being actively used in the field of health sciences to study the human body, microbes and various kinds of diseases.

1.5 Virtual Reality

Virtual Reality is defined in Encyclopedia Britannica as,

> The use of computer modeling and simulation that enables a person to interact with an artificial three-dimensional (3-D) visual or other sensory environment. VR applications immerse the user in a computer-generated environment that simulates reality through the use of interactive devices, which send and receive information and are worn as goggles, headsets, gloves, or body suits. In a typical VR format, a user wearing a helmet with a stereoscopic screen views animated images of a simulated environment [20].

The headset or the Head Mounted Display (HMD) surrounds the user's head and covers the eyes, cutting visual contact from the physical world. This enables the user to experience whatever environment the VR generates from a first-person perspective. This can make the virtual experience very real, plausible and immersive if implemented properly. The rate at which VR technology is changing is unbelievable. A decade ago, it was impossible to fathom the things that VR is capable of at present. It has entered all sectors ranging from studying medicine to visualising architecture to virtual museums, and is here to stay.

1.5.1 Immersion and Presence

Discussion about VR is complete without explaining the two very important properties associated with it. These properties are:

- **Immersion:** Immersion is a property of a media system that, with the help of various technological elements, tries to stimulate the different senses and create experiences that feel real. High-Def film theatres, VR goggles, and surround sound systems are some of the advanced technology that is used in today's world to make the media experience immersive. In the case of VR, high-pixel density screens for each lens, field-of-view, frame rate, and hand tracking are some ways to make the VR experience immersive and real-world like [21][31].

- **Presence:** Presence or Psycho-Presence in terms of VR is a physical and cognitive sense of being present inside the virtual environment even though it is not real. Psycho-presence is a direct result of the immersion of a system. The better the immersion of a system, the better the stimuli to the senses are, which in turn elevates the psycho-presence of the individual [21][31].

Immersion and Presence are indicators of the quality of the VR experience, but they can also give rise to one of the major drawbacks of VR, cybersickness.

1.5.2 VR and Cybersickness

Cybersickness can be thought of as a motion sickness that results from using simulation systems or Virtual Reality representations[16]. Motion sickness is typically triggered by low-frequency vertical, lateral, angular, rotary motion, or virtual stimulator motion, to which an individual has not adapted. Sine qua non for developing motion sickness is when the brain receives conflicting information from different sensors about real body movements or virtual environment. The principal sensors are the eyes, the vestibular apparatus, and proprioceptive receptors.[18].

A common scenario where one could experience motion sickness is during a car ride. Reading a book while moving in a car can cause motion sickness. This happens because the eyes and the ears send conflicting information to the brain. As per the eyes, the user is stationary, but the inner ears sense the twists and turns of the ride. When an individual uses a VR device with an HMD, the exact opposite of the car scenario happens. The brain gets the notion of the changing environment from the eyes, but the inner ears suggest that the body is at rest. These conflicting signals cause the user to feel sick.

As mentioned above, motion sickness is mostly caused by the motion to which the body hasn't adapted. Therefore, prolonged usage of VR can make the user adapt to the motion or the sense of motion and may help relieve the symptoms of motion sickness. Other steps that may help curb motion sickness in VR experiences include changing the movement mode to Teleport instead of continuous motion and snap turning instead of continuous turning. More about this has been discussed in Section 3.1.2.

1.6 Simulation in Virtual Reality

As discussed, Simulation and Virtual Reality can both prove to be powerful learning tools. They provide a more intuitive and natural way to look at and understand things. However, combining these two can take their benefits to a whole new level. VR enables the first-hand experience of the virtual environment, combining this with simulations makes them far more real and immersive and guarantees a much better understanding when used as an educational tool.

1.6.1 Simulating Mental Health Disorders in VR

As mentioned earlier, simulations in virtual reality are being used effectively as an educational tool in the field of medicine for a long time now [6]. However, this approach has also been extended to the field of mental health. This includes setups where students get to experience the disorders from a first-person perspective (also the agenda of this thesis) or interact with a virtual patient with the said disorders. This approach can be beneficial in ways including:

- There is no time restriction, the simulation can be played again and again to study the subtleties or revisit things that might be missed in the initial attempts.
- In the case of actual patients, the efficacy of the demonstration largely depends on how willing the patient is. But this is not the case with VR simulations.
- It can be used for mock clinical/ psychiatry interviews.

This has already been tried and tested by several reputed institutes with promising results. Studies such as *A virtual environment to simulate the experience of psychosis* [6], *Testing the efficacy of a virtual reality-based simulation in enhancing users' knowledge, attitudes, and empathy relating to psychosis* [9], and *Virtual Patients for Clinical Therapist Skills Training*[15] have proven the immense potential that VR has as an educational tool.

2. Project Setup

This chapter will outline the key elements involved in hardware, software, and framework setup. It will also shed light on the choices made while selecting those elements, as well as the thought that went into it. The chapter will also discuss the pros and cons of the said elements.

2.1 Hardware

As mentioned before, VR development requires dedicated hardware that is capable of playing VR media. Ever since Meta (formerly Facebook) announced its plans to invest significantly in VR, a lot of big players have launched their own version of VR headsets. They are coming in all shapes, sizes, and price points, each with its advantages and drawbacks. Some of the most popular VR devices include Oculus Quest 2 by Meta, PlayStation VR by Sony and HTC Vive. For the scope of this project, Oculus Quest 2 was deemed to be the ideal device.

2.1.1 Oculus Quest 2

Oculus Quest 2 is a VR headset developed by Oculus, a subsidiary of Meta. It is the successor to the original Oculus Quest Headset. Some of the key features that made it ideal for this project are:

- **Standalone Capabilities:** Oculus Quest 2 is a standalone VR headset. This means that it doesn't require a powerful PC to power it. It has its own processing unit and can work independently. This makes it highly portable and easy to use.

- **Powerful Hardware:** Oculus Quest 2 comes with a Qualcomm Snapdragon XR2 SoC, 6 GB Ram and 128 GB or 256 GB storage options. This makes developing and deploying high-fidelity 3D experiences on this device very easy.

- **Display:** The Quest 2 has a per-eye resolution of 1832×1920, a significant increase in pixel density when compared to its predecessor. This makes the visual experience far more immersive.

It also has several advantages over other major VR headsets, including:

- **Price:** Oculus Quest 2 is a mid-ranged VR headset, making it far more affordable compared to other high-end headsets like Vive Pro or Quest Pro. This makes it ideal for students who intend to learn VR development or people who want to try VR as a hobby.
- **Setup Requirements:** Oculus Quest 2 is a standalone VR device, and it doesn't require an external PC for functioning, so pretty much everyone can use it as it is. This is why Quest 2 appeals to a wider audience.
- **Tracking and Controllers:** Devices like HTC Vive Pro use external trackers to gather positional data of the player. These equipments are heavy and not very easy to set up in a regular household. Oculus Quest 2 solves this problem with inside-out tracking using its built-in cameras, eliminating the need for external sensors. It is very easy to use, even for the least experienced users.

But, there are also some drawbacks that Oculus Quest 2 possesses. They are very minor compared to all the features it brings to the table, but they are still worth mentioning.

- **Meta Account Requirement:** Meta services are integrated into the core of Oculus Quest 2 OS. A Meta account is required for the proper functioning of the device, as well as developing apps for the Oculus headset. The user is given the option to create an unmerged Oculus account or merge their Facebook account and use it for development. This has led to concerns about privacy for both end users and developers.
- **Comfort:** The Oculus Quest 2 fits the battery as well as the SoC in the HMD. This shifts the entire balance of the headset towards the front. At the

rear, it uses a stretchable nylon strap to support the HMD. This can become very uncomfortable for the user over long periods of usage. However, to address this problem, there are a lot of third-party straps available that replace the original strap with a more comfortable version, that has the battery at the back. This distributes the weight more evenly, resulting in a more comfortable and better fit.

- **IPD Settings:** Interpupillary distance or IPD is the distance between the centre of the pupils. The glasses and VR devices that we wear over our eyes need to account for this for clearer viewing. Oculus Quest 2 has three presets for IPD instead of a continuous slider, which gives more control. This can lead to clarity issues for users who don't have IPD values similar to the presets.

For the development of the software for this VR project, an HP Victus laptop was used. It is a high-end machine with RTX 30 series GPU and Windows OS. This ensured seamless development and testing of the project, without the need to build and deploy the app to the VR headset every time a new feature needed to be tested. This saved a significant amount of time during the development and also made debugging quick and easy.

2.2 Software

Making a game or an interactive simulation can be a very tricky process. There are multiple aspects to a game that needs to be set up before one can program even the most basic feature of the game. Based on the type of the game being developed, this can include Physics and collision detection, input, lighting, graphics, and rendering. The developer also has to worry about the platform they are building and may need to make substantial changes if they wish to port their game to a different platform in the future. This section discusses various software tools, libraries, and frameworks that greatly assist and simplify the process of game development.

2.2.1 Game Engine

Arm Limited defines a game engine as a "software development environment, also referred to as a "game architecture" or "game framework," with settings and configurations that optimize and simplify the development of video games across a variety of programming languages" [1].

Game Engines can make things easier and faster for a developer, as they come equipped with a collection of tools and libraries that work as a framework on top of which the developers can build their game. All the key aspects — lighting, graphics and rendering, physics, and cross-platform support are some of the features that a game engine may provide out of the box.

Unreal Engine[33] and Unity 3D[32] are two of the most popular game engines that are used today. Based on the use case, both of these may present their own advantages and disadvantages. This can influence the decision of the developer while choosing the right game engine for their game.

This project is made with Unity 3D as the game engine. Developed by Unity Technologies, Unity is a cross-platform game engine that was first introduced in 2005. It holds a major market share among game engines based on the number of users and offers support for both 2D and 3D development. Unity comes with all the major libraries built in, making the game development process more efficient. It boasts a vast and highly active online community, ensuring easy access to support. Additionally, Unity offers comprehensive resources for creating VR experiences.

2.2.2 VR Software Development Kit

VR development adds another layer of complexity to an already complicated process of game development. Like game engines help to streamline the process of game development, VR Software Development Kits (SDKs) provide the additional setup required to make VR games and experiences. It handles the VR Camera setup, input from the hand controllers, HMD, and other sensors (if any).

A lot of industry-leading companies have their own VR SDKs, including Oculus Interaction SDK by Oculus, XR Interaction Toolkit by Unity Technologies, and Mixed Reality Toolkit (MRTK) by Microsoft. All these come with their own set of features and use cases.

This project was started with Unity's own XR Interaction Toolkit, as it is platform-independent and the same app could be used with multiple supported VR devices. It was later decided to make the UI inside the experience, hand controlled to make the simulation more immersive. At the time of development, Unity XR toolkit did not support hand tracking and hence, Unity XR Toolkit SDK was swapped with Oculus Interaction SDK. Oculus has one major drawback, that it only supports development targeted at the Oculus Quest devices. For this very reason, it is very likely that the project would switch back to the Unity XR toolkit once it introduces hand tracking. This would ensure that the project runs on other VR devices as well.

3. Analysis

The first chapter gives an idea of how the symptoms of a mental health disorder come in all shapes and forms. They can range from being purely physiological, like racing heartbeat, to purely psychological, for instance, excessive feelings of guilt to being somewhat of a mix of the two, such as troubled speech resulting from racing thoughts.

The recipe for a good simulation is the ability to make it as precise as its real counterpart. Simulating a mental health disorder and reproducing its symptoms virtually pose multiple major challenges. The biggest challenge is posed by the limitations of the current technology. Current hardware can only stimulate the audiovisual senses of the user. Users can experience modern-day cinema through their eyes and ears only. There is no way to touch, taste or smell the elements of the film. Although many kinds of hacks make an appearance occasionally, they are nowhere close to the real experience. This means that while developing mental disorder experiences for today's devices, the development is limited to the symptoms that involve audiovisual stimuli in some or the other way.

But as mentioned above, the symptoms come in all shapes and forms. Therefore, in order to accommodate the complex symptomatology, several workarounds, hacks, and creative approaches had to be used throughout the development of this project. This chapter describes the thinking and the brainstorming that went into figuring out methods to simulate these symptoms. The format of this chapter, for the most part, consists of an explanation of how the symptoms are simulated in VR. While this chapter mostly describes the approach to simulate the symptoms at a non-technical level, a more low-level explanation touching upon the technical implementation of the symptoms would be provided in the succeeding chapter.

3.1 Simulating the Symptoms

While doing the research for the thesis, it was realised that how various mental health disorders can share multiple symptoms at times. For instance, many disorders can have racing thoughts, hallucinations, or loss of appetite as key symptoms. This motivated the project development in such a way that the framework for the simulation, once complete, could be easily extended to accommodate other mental health disorders. The techniques used to simulate the symptoms have also been implemented, keeping reusability and scalability in mind. The next section would describe several techniques that were adopted to effectively show the symptoms. For the technical explanation of the same, refer to Section 4.5.

3.1.1 Inner Speech

Inner speech, also known as self-talk, internal monologue, or inner discourse, can be defined as an internalized, self-directed dialogue. It loosely translates to the inaudible voice in the back of your head. It is usually attached to the person's consciousness. Inner speech is a continuous process that goes on when we are conscious. This involves critical thinking, problem-solving, contemplation, forming impressions, reacting to emotions, and subvocalization (reading in one's head) [22].

Inner speech often mirrors a person's thoughts, emotions, and feelings. It is very relevant to mental health disorders as well. The change in mood, thoughts, and emotions can also be seen in the inner speech of the affected individual.

Being able to replicate the Inner Speech and thoughts of a person was one of the major goals of the simulation. AudioSources have been used in an attempt to simulate the Inner Speech. AudioSources are pre-built components provided by Unity Engine that are used to play sounds in the environment. The various parameters of the audio clip played by the AudioSource, such as the volume, pitch, and speed, can be controlled programmatically. Pre-recorded audio clips are provided to the AudioSource, which are then selected at random and played to mimic the thought process of an individual. Audio clips are categorised on the basis of various moods, and each mood category has multiple audio clips. For instance, an overwhelming feeling of joy, which is a symptom of Mania, makes a person extremely happy and makes them feel high about themselves. Audio clips recorded with dialogues corresponding to the energetic mood can be grouped together, chosen at random, and played to replicate the symptom.

Below are the various modifications made to the AudioSource to make it resemble the Inner Speech.

- **Same Position as HMD:** The AudioSource is made to have the same position in the world as the HMD of the VR Headset. The idea behind this is that they have the effect that the sound is coming from within the head.
- **Reflecting the Mood:** Keeping Bipolar Disorder in mind, various emotions that are relevant to the symptomatology, were selected. These included — *Happy, Energetic, Motivated, Guilty, Sad,* and *Suicidal.* Dialogues are recorded corresponding to those moods. For instance, in mania, the dominant moods are *Happiness, High Energy,* and *High Motivation.* Attention is paid to making sure that the tone of the dialogue expresses the emotion that the dialogue corresponds to. The same voice is used to record all the dialogues of Inner Speech, in contrast to Audio Hallucinations, which use different sets of voices. To avoid hiring a voice artist to record the dialogues, Speech Synthesis by ElevenLabs was used [17]. Speech Synthesis is an AI-powered Text-To-Speech software that automatically identifies the tone of the text and generates the audio with desired emotions.
- **First Person Mode of Speech:** Since the internal monologue always has the first-person form of speech, the dialogues are phrased in the first-person speech as well. This makes it more legible as well as helps in differentiating it from the external voices such as the Audio Hallucinations.

3.1.2 Locomotion

Locomotion in a virtual environment refers to the system that allows the player to control their virtual character and explore the virtual world. It incorporates the translation as well as the rotation/ turning of the character. When a person is suffering from a mental disorder, one of the key physiological symptoms is the change in physical activity of that person. Very restless motion during Mania and lethargy during Depressive Phase are some examples of how the disorder shows through physical activity.

Locomotion is a very effective way of mimicking the physical activity of a person with some mental disorder in the virtual world. Locomotion has certain parameters, such as walk speed, turn speed, snap angle etc., that can be modified programmatically. There are several modes in VR through which the character can be controlled, including physically walking/ turning, using the controller joysticks, or teleportation. Although we can't control the physical mode of locomotion programmatically, we can control the other modes, i.e., continuous locomotion using joysticks and discrete locomotion – teleportation and snap turn, and use them to replicate the symptoms. The parameters have been explained below.

- **Walk Speed:** This parameter regulates how fast the character translates when the continuous mode of movement has been selected. It can be increased to a very high value to mimic restlessness in Mania or decreased to a low value to depict lethargy or the unwillingness to perform any physical activity during Depressive Phase.

- **Turn Speed:** Other than the translation, the user can also steer the player in different directions. This enables the player to traverse the walking plane effectively. Turn Speed controls the rate at which the player can change direction. High Turn Speed, again, can be used to mimic Mania and low Turn Speed, Depression.

VR is known to induce nausea, and one of the biggest reasons for it is continuous movement inside the virtual environment. To curb this, a more discrete alternative is used, which is called teleportation.

- **Teleport Range:** Teleportation is the other mode of translation from one point to another. Instead of continuous movement, the player is teleported to the point they select. Teleport Range can be increased or decreased as per the need of the symptom. High range means the player can teleport farther, while lower range restricts the teleport distance.

- **Snap Angle:** Snap Turn is the teleport analogous to Continuous Turn. Instead of continuously turning, the player turns at fixed angles.

The values of the parameters of the teleport/snap-based locomotion can also be changed to mimic movement-related symptoms.

3.1.3 Visual Changes

A very significant change that an individual with a mental disorder may experience is how they perceive the external world through their eyes. For instance, for people with depression, the outside world may appear dull with a lack of colours and very monochrome. Whereas, in mania, everything may appear very bright and overly saturated in colours. Unity provides an effective solution to mimic visual changes with the help of a process called Post-Processing.

As per Electronic Arts,"Post-Processing techniques are used to change a rendered image as a last step before the presentation and include, but are not limited to, operations such as change of saturation or contrast, and also more advanced effects like depth-of-field and tone mapping" [8]. Following are some of the post-processing parameters that have been used to simulate the visual changes:

- **Bloom:** The Bloom effect can be seen as an artefact in the images produced by the cameras in the real world. The effect appears as if the light is leaking from the edges of a luminous object or a bright area (Figure: 3.1), thereby creating a bright light effect that can be a bit extreme for the camera [10].

This effect can be used to make the lights appear extra bright and could be utilised to mimic vision-related symptoms of mania, where everything appears extra bright.

Figure 3.1: The Bloom Effect

- **Vignette:** In the field of photography and imaging, vignette refers to the darkening or decolorising of the edges of an image (Figure: 3.2). This can be done by improper lens hood, thick or stacked filters, and secondary filters. Vignette is also used as an effect in photography to draw focus towards the centre of the image [14]. By controlling the radius of the vignette effect, two very important symptoms can be simulated. These are tunnel vision and loss of consciousness. These both can be present in many mental disorders, including anxiety and panic attacks.

- **Chromatic Aberration:** Chromatic Aberration is an artefact in photography that results from the inability of the camera lens to converge all the colours to the same point. This results in a rainbow-like effect (Figure: 3.3) at the edges separating lighter and darker objects. The Chromatic Aberration effect in Post-Processing replicates this artefact and is often used for artistic effects [11]. Chromatic Aberration can be used to mimic the effects of blurred vision, astigmatism, and intoxication.

Figure 3.2: The Vignette Effect

- **Motion Blur:** As the name suggests, Motion Blur is a Post-Processing effect that replicates the blurring of the image (Figure: 3.4) resulting from sudden movement of the camera or when the movement of the object is faster than the shutter speed of the camera. This can also be generated by increasing the camera's exposure time to longer durations. Motion Blur is used in many racing and fast-paced games to add smoothness and make things appear faster[13]. Motion blur is a very powerful tool to mimic dizziness, a very common symptom associated with panic attacks, anxiety, and depression.

- **Color Grading:** Color Grading is used in photography to change the color composition of the final image. It is similar to applying filters to the images on a smartphone [12]. Color Grading comes equipped with various parameters, but one major parameter that was used actively in the project is Saturation. Saturation controls the intensity of all the colours. It controls how vivid and vibrant the colours appear on the screen (Figure: 3.5). This property can be used to control how colourful or dull the environment looks to the user.

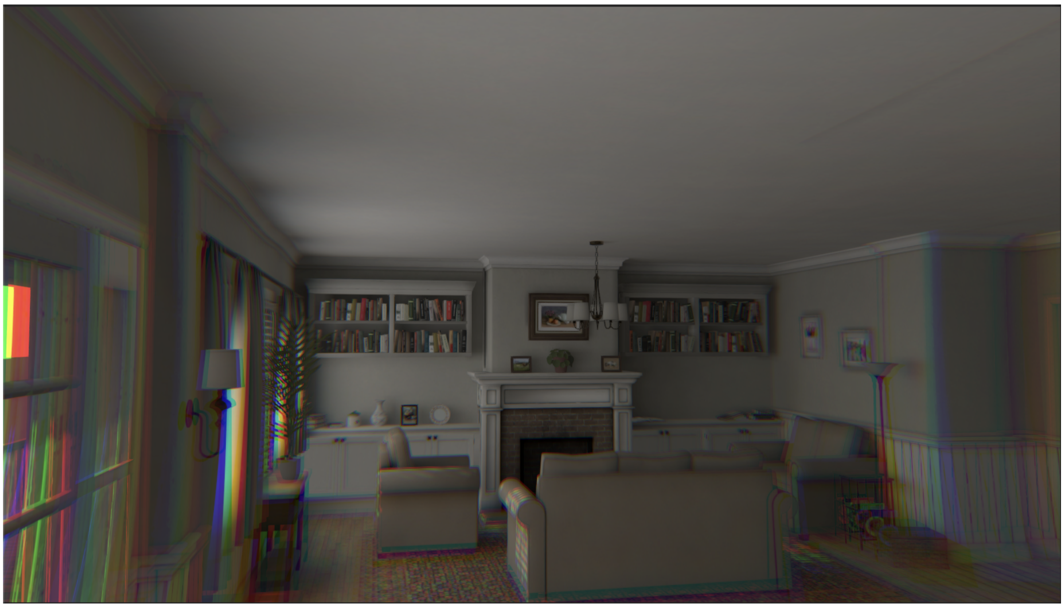

Figure 3.3: Chromatic Aberration

3.1.4 Behavioural Changes

So far, the symptoms discussed in this chapter had one or the other kind of audiovisual aspect associated with them. But as mentioned earlier, not all the symptoms of mental health disorders are limited to sound and visuals. For instance, how would one go about simulating something like sleep deprivation or excessive sleeping? This is where the idea of adding a personal device to the simulation came into being.

In this digital era, smartphones have become an integral part of an adult's life. They are constantly getting better at keeping accurate track of an individual's day-to-day activities. From keeping a record of the finances to the number of hours of sleep the user had, they can come in quite handy to get an idea of the user's behavioural pattern.

Hence, a virtual personal device has been added to the experience, and it includes various apps, like in a smartphone. These apps offered a new way to simulate specific tricky symptoms that were proving challenging to replicate using the previously mentioned audiovisual methods. It should be noted that hardcoded values have been fed to the apps as per the requirement of the symptoms, and in no way, the experience tracks any kind of real-life data.

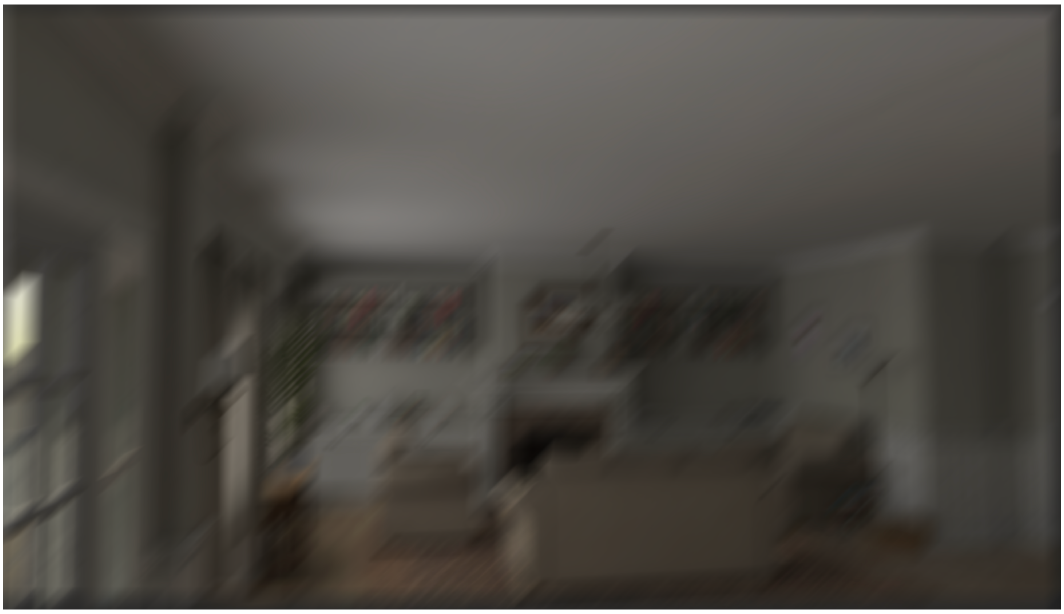

Figure 3.4: Motion Blur in Action

Some apps added for this purpose include:

- **Sleep:** The *Sleep* app in the experience is fed data such as – total sleep, in-bed duration, time of going to bed, time of waking up, percentage of awake time, percentage of REM sleep, percentage of deep sleep, and percentage of light sleep. It also shows the change in the sleep duration compared to the previous week (Figure: 3.6). The values of these parameters for a healthily functioning human are already known and are modified accordingly. For instance, if the person has insomnia, the value of sleep duration decreases significantly and similarly, the time awake in bed increases.

- **Health:** The *Health* shows some common yet important health metrics. These include weight, average heart rate, and weekly calorie intake. These parameters can help simulate abnormal changes in weight, loss of appetite and stress levels. Again, the normal levels of these parameters are well-known and can be modified as per the symptom. For instance, to show loss of appetite, weekly calories count can be reduced along with the change in weight.

- **Bank:** The *Bank* app provides an idea of the spending pattern of the individual using different kinds of monthly expenditures such as – Eating

Figure 3.5: An example of how Saturation can control the colour intensity of the scene.

Out, Groceries, Shopping, and Electronics. It also tells about the change in the expenditure compared to the previous month. A symptom of Mania is reckless spending, which can be simulated by increasing all the values of this app to big numbers.

- **Notifications:** The *Notification* app tells about the number of missed calls, messages, emails, and other notifications. A high number of unattended notifications would imply that the person is avoiding socialising, whereas a low number would imply normalcy or hyper-social behaviour.

Whenever the user opens any of these apps, the Inner Speech dialogue changes to match what the user would feel looking at the stats in the app.

3.1.5 Visual Hallucinations

Visual Hallucinations are another symptom that is present in a majority of schizoaffective mental health disorders. It can include – seeing things move when they are actually still, things that are not present, light flashes and colours etc.

SpriteRenderer and, ParticleSystems have been used to simulate visual hallucinations. The SpriteRenderer component displays 2D graphics, also known as sprites, on the screen. The choice of using 2D graphics over 3D models was made

Figure 3.6: The UI of the Sleep app showing the vital stats.

because it is significantly lighter in terms of calculations. This allows for multiple sprites in the environment at any given moment without impacting the performance at all. Moreover, making 3D models can be difficult and time-consuming.

ParticleSystem is another very powerful component that is used to create special effects like fire, smoke, fog etc. It usually spawns numerous particles of the same kind with certain variations to produce the mentioned effects as a whole. ParticleSystem has been used to simulate bright lights and flashes, which are often reported as symptoms of mental disorders.

Some steps taken to tweak the visual hallucinations include:

- To minimize the use of additional resources, generative AI tools such, as DALL-E [26] and Lexica [2] are used to obtain the required images for hallucinations.

- The SpriteRenderer and ParticleSystem include parameters such as colour, size, and transparency. Controlled Randomization of these parameters ensured variety in the results.

- Since the sprites are 2D and don't have thickness/ depth, the billboard effect is applied to all the sprites. The billboard effect ensured that all the sprites were facing the player.

- Shaders have also been used to give special effects to the sprites. Distort Shader is used to provide the sprites with a blurry, distorted effect, as the hallucinations are usually not very clear. Distort shader also added a wavy movement to the things it is applied to.

3.1.6 Audio Hallucinations

Another key symptom of a number of mental health disorders is hearing sounds and voices which are not actually present there. Like the Inner speech, Audio hallucinations have also been implemented using the AudioSource component. The following are the measures taken to refine the experience:

- **Varying Position:** Unlike Inner Speech which used just one AudioSource, Audio Hallucination uses multiple audio sources that keep on changing position w.r.t. the HMD to give a sense of plural entities talking to the individual.

- **Reflecting the Mood:** Various moods have been selected for the dialogues generated for the voices of the audio hallucination. These included — *Happy, Energetic, Motivated, Guilty, Sad,* and *Suicidal.* Dialogues are recorded corresponding to those moods. For instance, in mania, the dominant moods are *Happiness, High Energy,* and *High Motivation.* Attention has been paid to make sure that the tone of the dialogue expresses the emotion that the dialogue corresponds to. Several voices have been used to record all the dialogues used in the hallucination simulation. Similar to Inner Speech, Speech Synthesis by ElevenLabs was used for generating the audio files.

- **Second Person Mode of Speech:** To give a sense of voices talking to the individual, the dialogues have been recorded in the second person mode of speech. This helps to differentiate the hallucination from the Inner Speech and makes the user feel that he is being addressed by the voices.

3.1.7 Loss of Focus

Lastly, some other key symptoms that were on the agenda of this project included — distractibility, loss of focus, and zoning out. The approach used in an attempt to simulate these symptoms involved the use of ambient sounds. They are similar to Audio Hallucinations in some aspects and totally opposite in others. Multiple audio sources are used for ambient sounds as well. However, they don't use dialogues but various sounds — Whispers, Traffic, Construction, Beeps instead. They also differ in execution as they start with a very low volume that increases gradually to a maximum and at that point the user can hear nothing but these sounds. This can help to stress how zoning out or losing focus feels like.

4. Implementation

The preceding chapters have primarily focused on providing an overview of the theory, setup, and high-level explanation of how things are working behind the scenes. Consequently, this chapter serves as an ideal platform for delving into the detailed implementation of essential components within the project, which are crucial for the functioning of the simulation as a whole. The chapter discusses the unity project setup, application flow, provides a technical explanation of the symptom implementation, and other related aspects. The chapter concludes by examining the challenges and difficulties encountered when working with VR, along with the measures taken to overcome them.

4.1 Unity Project Setup

For the sake of reliability, it was decided to use Unity 2021 LTS. LTS builds are provided updates for a longer period of time, and extended support compared to regular builds. They are also more stable and less prone to bugs. Before the start of any Unity project, it is a great idea to do the project setup. Project Setup comprises a series of steps to tweak Unity settings as per the needs of the current project. This includes choosing the render pipeline, target platform, signing the application, and other project-specific settings.

A render pipeline is a series of steps and algorithms that a 3D virtual scene goes through, and the result is the 2D image that is displayed on the screen. For this project, Universal Render Pipeline (URP) has been used, which is one of the types of Unity's new Scriptable Render Pipeline (SRP). This choice was made because URP is lightweight compared to its counterpart. This is important since keeping a constant frame rate, which is greater than 70 FPS, is crucial for an ideal experience and also reduces motion sickness to some extent. But, at the same time, URP is powerful enough to support Post-Processing, which enabled the implementation of some key visual symptoms.

Building a project in VR also requires VR specific setup. This can include choosing the VR framework, target VR device-specific settings, optimisation settings, stereo rendering modes etc. As mentioned earlier, the project uses Oculus Framework, which may be later changed to Unity XR to support a larger number of VR Headsets.

4.1.1 Folder Hierarchy

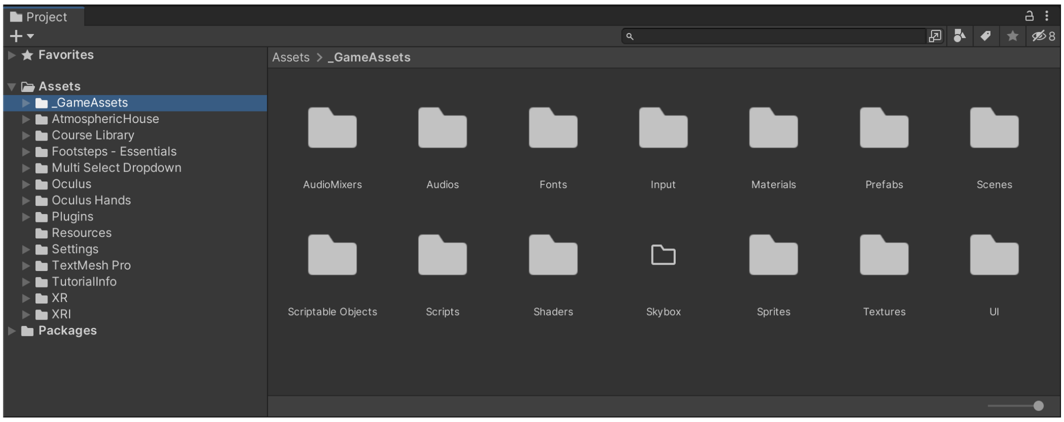

Figure 4.1: The folder structure of the *_GameAssets* Folder

This section briefly touches upon the folder structure of the project. The parent directory of the project is called *Assets*. Everything that is needed to be used or is referenced in the game in some form should be present in the *Assets* folder.

For the sake of clarity, the game art and assets have been separated from other files present in the project. All the files, assets, artwork, and scripts that were generated purely for the purpose of being used in this project are placed in *_GameAssets* folder. Figure: 4.1 shows the interior of the *_GameAssets* folder. The folder *Scripts* within the *_GameAssets* folder contains all the source code of the project. Similarly, all the other folders have assets specific to the experience with self-explanatory folder names.

The parent directory *Assets*, which also includes the *_GameAssets* folder, contains other important files responsible for the proper functioning of the game. This includes 3rd party plugins, framework files, demos, URP settings etc.

4.2 Application Flow

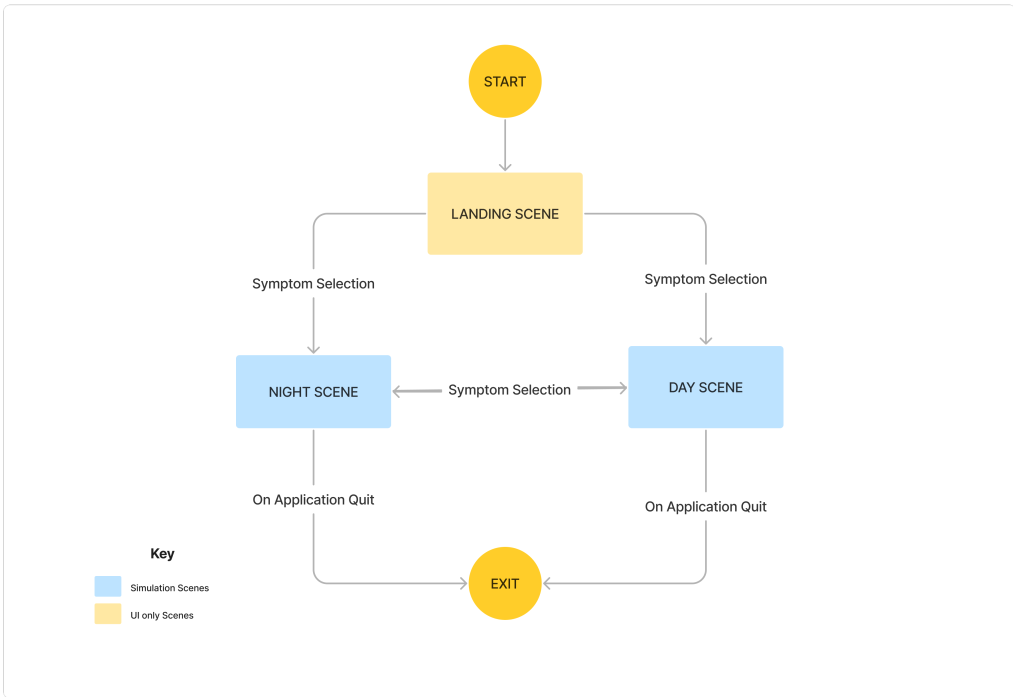

Figure 4.2: Application Flow

The experience consists of three scenes in total. As soon as the application launches, the users come across the *Landing Scene*. From there, based on the preset they chose, they either enter the *Day Scene* or the *Night Scene*. Figure: 4.2 tries to explain the application flow graphically while the detailed explanation of each of the scenes is as follows:

4.2.1 Landing Scene

This is the very first scene that comes up when the experience is launched. The *Landing* Scene is not a part of the simulation and doesn't simulate any symptoms in any manner. To minimize the stimulation of the user before the actual simulation starts, the *Landing* Scene consists of no 3D models. It is just a vast expanse of a light grey Skybox and a dark grey floor that stretches in every direction. This colour scheme has been used because shades of grey are not very strong on the eyes, and they make a great choice for background colours, hence, giving a neutral feel to the *Landing* Scene.

The personal device is enabled by default in the *Landing Scene* and is visible to the user at chest height and an elbow's distance. The experience supports hand tracking and the entire UI of the device can be controlled using the fingers like the real world. This enhances immersion to a great extent.

The device shows a static tutorial in the beginning, briefing the user about the controls. Once the user goes past the tutorial, the symptom settings screen comes up. The symptom setting screen has various tabs, including a symptom preset tab. Once the user applies the symptom settings, then based on the type of disorder, The *Day Scene* or the *Night Scene* is launched.

Both these scenes have the exact same environment. However, they do differ in terms of the house's interior and the lighting of the environment. The house model for these scenes has been imported from the Unity Asset Store [4].

4.2.2 Day Scene

The *Day Scene* has been associated with life, energy, and positivity. To add more to this association, the lighting of the *Day Scene* is very bright with ample light inside the house. The look and feel of the entire house gives the idea that the house is newly renovated, being appropriately maintained, and taken care of. This setting can be used to simulate the normal behaviour of an individual or something like Mania after tweaking the lighting settings.

4.2.3 Night Scene

The *Night Scene* used the exact same 3D models as the *Day Scene*, but it has been used to convey something opposite of what the former signifies. The *Night Scene* has been associated with a lack of energy, dullness, and emptiness. The lighting mostly consists of fluorescent tubes, which also add to the dull effect that has been aimed for. The look and feel of the house is very battered, which gives the idea that the house isn't being looked after actively and has not been renovated in a long time. The textures of the walls have been swapped with the dirty, worn-out variants that give the house a really shabby feel. This setting can be used to simulate the behaviour of a depressed individual or an individual suffering from psychosis.

4.3 Architecture

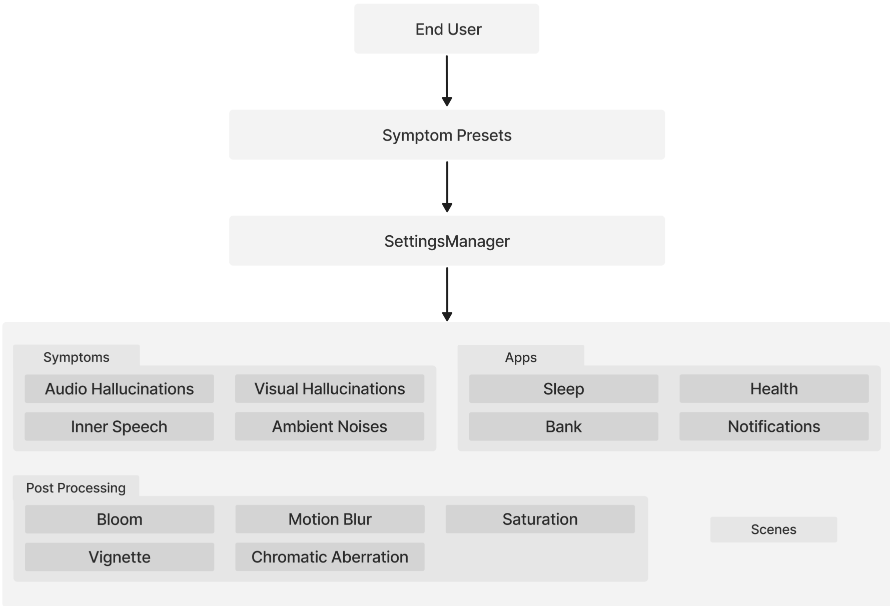

Figure 4.3: Application Overview

An essential thing considered while building this application was that it might be used by Psychologists and Psychiatrists at some point. This meant extending the application, including — adding, removing and editing symptom presets, changing audio-video files, and other tweaks should be intuitive and user-friendly even for someone with little to no experience with the Unity engine. For this exact reason, there has been extensive use of `ScriptableObjects` wherever necessary. Figure: 4.3 gives a brief overview of how different entities interact inside the application.

The end user interacts with the personal device that is present as a computer tablet inside the experience as mentioned before. Inside the tablet, there is a *Settings* icon that takes the user to the *Settings* screen. Through the *Settings* screen, the user can select pre-defined symptom presets or customise individual symptom values to their liking. The *Settings* screen is controlled by `SettingsManager`. Any time a symptom preset is chosen, or some new settings are applied, `SettingsManager` fires an `event`. All the other classes that are related to the simulation are subscribed to this `event` and they make the necessary changes to the variables to reflect the symptom settings on the *Settings* screen. This entire process will be discussed in the following sections in detail.

4.3.1 Settings Screen

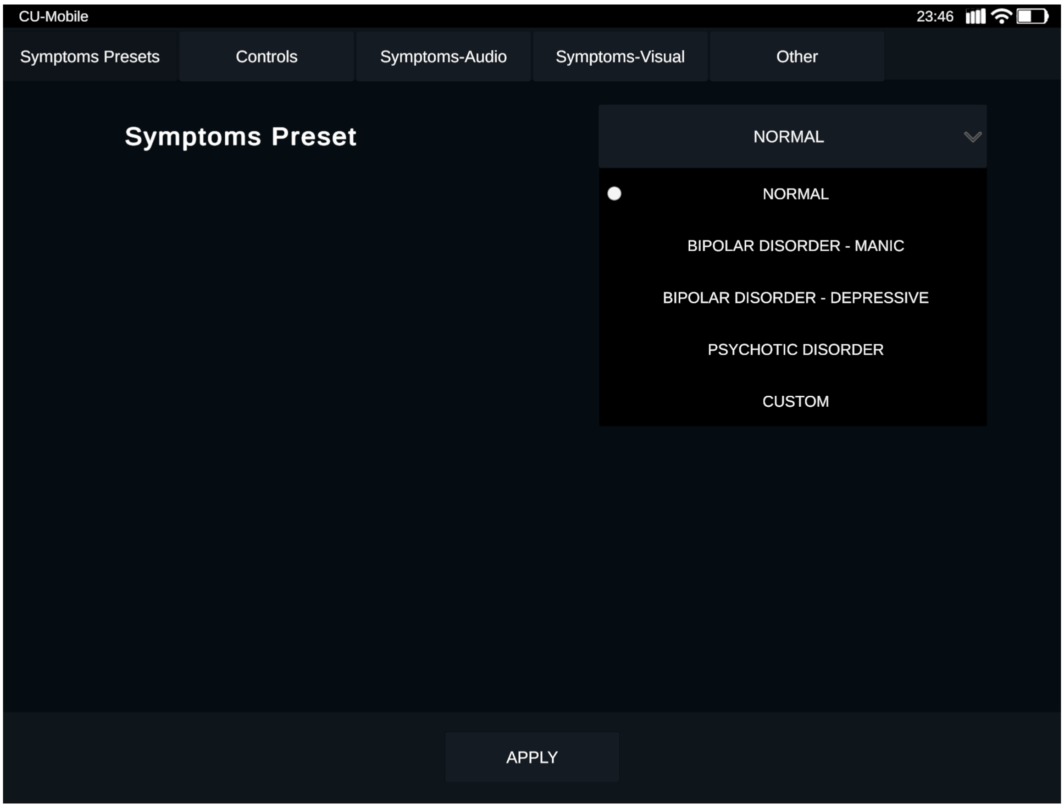

Figure 4.4: Symptoms Preset tab of the Settings Screen

The *Settings* screen Figure: 4.4) is responsible for all the symptom configurations, including enabling/ disabling the effects, regulating the intensities, editing the presets etc. The screen's layout has been designed like a regular video game settings menu. It has different sections for different categories of parameters. The *Settings* screen has five tabs — *Symptoms Presets, Controls, Symptoms-Audio,* *Symptoms-Visual,* and *Other*. Different shades of grey have been used for the majority of the settings UI. White and blue are the other colours that are a part of the colour scheme. These colours have been used because they are neutral, easy on the eyes, give off a professional feel, and hence, seemed ideal for an educational application.

Symptom Presets

Symptoms Presets is the first screen that the user comes across when the *Settings* screen shows up for the first time. It has a drop-down menu with different symptom presets to choose from. A fresh installation of the application includes these four presets: *Normal, Bipolar Disorder – Mania, Bipolar Disorder – Depression*, and *Psychosis*.

Each preset has a `ScriptableObject` file mapped to it that stores the configurational values relevant to that particular symptom. Once a preset is selected from the drop-down menu, these values are copied to all the various settings present on the *Settings* screen. Other than the presets that come out of the box with the application, there is a possibility to add new symptoms, edit them, and delete them. This would enable the experts to tweak the symptoms from the app itself without having to make a new build. All the above-mentioned processes are handled by `SettingsManager` class, which is explained in detail in the next section.

SettingsManager

`SettingsManager` is one of the most important classes that works as the mediator between the various Symptom classes and the UI. It is responsible for reflecting the changes in the settings on the UI, recording them, and relaying them to the respective symptoms classes. It is also responsible for saving, deleting, and editing user-defined symptom presets.

The API of `SettingsManager` does not include any `public` methods. The way it works is by having a reference to all the UI elements that are present on the *Settings* screen. All the methods that are responsible for synchronising the changes between the UI and the ScriptableObjects are subscribed to the `event OnValueChanged` of the respective UI elements. Once there is a change in the value of a particular UI element, the method subscribed to the event is called, which changes the values stored in the `ScriptableObject`.

As mentioned earlier, each symptom preset has a variable of the type `ScriptableObject SymptomSettingsSO` that holds all the data configuration values related to that preset, and there is a value present for each and every setting that is present in the settings. `SettingsManager` maintains a list of all the `SymptomSettingsSO`. Other than these, `SettingsManager` also has two more `SymptomSettingsSO` variables — `currentSetting`, and `savedSetting`.

It needs to be noted that the user can make any changes to the symptom settings, but they are not reflected till they press the *Apply* button present on the settings screen. In case they choose to press the *Cancel* button, all the changes made are reverted to previous values that were there before the user made those changes. This is ensured with the help of `currentSetting` and `savedSetting`. While the user is making the changes to the settings, they also get copied to the `currentSetting`, while the `savedSetting` contains the values from the preset *Normal* initially. Once the user is done making those changes, they can either cancel those changes or apply them. If the user presses the *Apply* button, the values from `currentSetting` are applied to `savedSetting`. At the same time, the `Action SettingsUpdated` is fired, this enables all the scripts subscribed to this event to apply the new settings value to their respective parameters. In case the user chooses to cancel the changes, the values from `savedSetting` are copied to `currentSetting`, these changes are further propagated to the UI, and it shows the reverted values on the UI. This should be noted that `currentSetting` is always in sync with the settings values on the UI. In case the user selects a preset, instead of changing one particular setting, the whole `SymptomSettingsSO` mapped to that particular preset is copied to `currentSetting`.

`SettingsManager` is also responsible for saving, deleting, and editing the user-defined symptom presets. Once any kind of change is made to the default presets, an asterisk (*) appears next to the name of the selected symptom preset to illustrate the change, along with a *Save As* button at the bottom of the screen. The user can then choose to save the modified preset under a new name, which is then added to the drop-down menu of presets. `SettingsManager` then takes the values in `currentSetting`, serialises them and saves them to the device storage as a text file. This addition is persistent, which means, that the new preset would be present there the next time the application is launched. If the user makes any changes to the user-defined presets, *Update* and *Delete* buttons also appear at the bottom, along with the *Save As* button. If the user wishes to update the changes made to the selected user-defined preset, they can press *Update* button. This copies the values stored in `currentSetting` to the `SymptomSettingsSO` mapped to the selected preset. This change is also updated in the text file stored in the device storage. Similarly, if they wish to delete a user-defined preset, they can do so by pressing the *Delete* button, which removes the preset from the preset dropdown as well as the device storage.

4.4 The Symptoms Class

Section 3.1 broadly classifies different kinds of symptoms and the various approaches taken to simulate them. There is one major characteristic that divides these symptoms into two different groups — Symptoms that remain persistent throughout the simulation and symptoms that change over time. The latter is the kind of symptoms that may increase or decrease in intensity, have a time duration, and may or may not get triggered. These similar attributes mean they could be extended from one parent class and then personalised as per their specifications. Hence, the `Symptoms` class. The `Symptoms` class is `Abstract` in nature, which allowed defining the common features that the symptoms extending this class would share. `Symptoms` has a reference to the `SymptomSettingsSO` `currentSetting` with the `protected` access modifier. This allows all the children classes to access the settings values and modify their respective simulations accordingly. The `protected` API includes `ShouldSimulate()`, which decides whether the simulation would run for that particular iteration or be skipped. The `public` API includes a `abstract` method `Simulate()`, which the children classes implement as per their requirements. Other `abstract` methods are `Stop()` — to stop the simulation after the current iteration is over, and `StopAbrupt()` — to stop the simulation immediately. `Symptoms` also has access to the `GameObject` *character* i.e. the model of the character in the game, the `Camera` *CharacterCamera* i.e. the VR camera, and the `AudioSource` *MainAudioSource*, the audio source responsible for the inner speech. The references to these are stored in `protected` properties and can be accessed by the classes extending `Symptoms`.

4.4.1 SymptomsSO - Configuring the Symptoms

As previously mentioned, emphasis is placed on making the application customisation user-friendly, even for individuals with limited technical expertise. `ScriptableObject`, as data containers, have been used extensively throughout the app to ensure this. A common characteristic of all the symptoms derived from the `Symptom` class is that they vary over time based on the parameters. These parameters come in various forms, from numerical ranges to transparency values to audio clips. Each symptom has some common parameters and some unique set of parameters, and it is important that each one of these parameters is easily customisable from the Unity editor without meddling with the source code.

To solve this problem, `ScriptableObject SymptomsSO` analogous to the class `Symptoms` is introduced. All the classes extending `Symptoms` store and reference data containers extending `SymptomsSO`. The values are then used to customise the parameters influencing the simulation. `SymptomsSO` like `Symptoms` is also `Abstract` and not instanced itself. The base class `SymptomsSO` has just one variable `probability`, which is a common parameter used by all `Symptoms` classes. Each `SymptomsSO` class can have multiple instances to store data for various settings of that particular `Symptom`. For instance, `AudioHallucinations` uses a list of `SoundsSO` derived from `SymptomsSO`. These `SoundsSO` vary based on the mood – Anger, Guilt, Happiness etc.

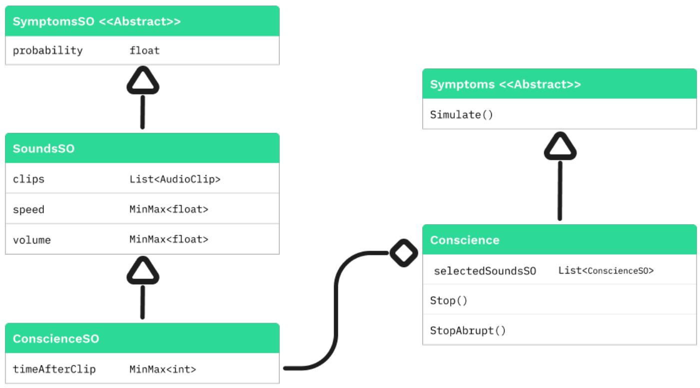

Figure 4.5: Relationship between Conscience and ConscienceSO

All the symptoms that are derived from the `Symptoms` class have been explained in the following section.

4.5 Extending the Symptoms Class

This section would discuss how all the different classes extend the `Symptoms` class. The classes have been grouped together based on their similarities.

4.5.1 Inner Speech

The class `Conscience` handles the simulation of Inner Speech. This class extends `Symptoms`. `Conscience` uses a list `List<ConscienceSO> selectedSoundsSO` where `ConscienceSO` is a scriptable object which extends `SoundsSO`.

`ConscienceSO` contains a variable `timeAfterClip`. This variable gives a

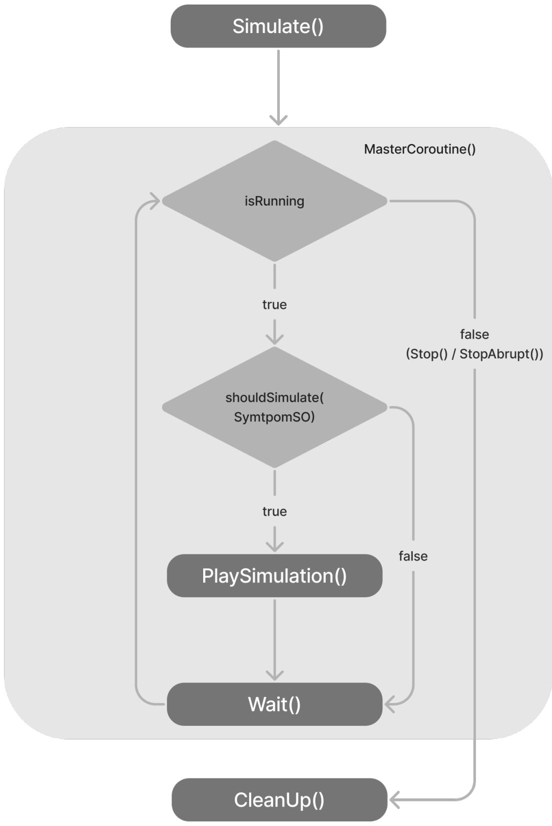

Figure 4.6: Order of execution of Conscience.cs

range from which a random value is picked that decides the time the coroutine would wait after playing an audio clip. `SoundsSO` contains the variables `clips`, `speed`, and `volume` to store the audio clips, speed range, and volume range, respectively. `SoundsSO` in turn extends `SymptomsSO`. The entire hierarchy has been explained in (Figure: 4.5). `selectedSoundsSO` contains multiple elements representing various moods that the Inner Speech can have based on the preset selected. The `Abstract` method `Simulate()` has been overridden and used to start the coroutine `IEnumerator MasterCoroutine()`. This coroutine loops indefinitely till the `Stop()` or `AbruptStop()` methods are not called (Figure: 4.6).

During each iteration, `MasterCoroutine()` chooses one of the elements of the type `ConscienceSO` from the list and, based on its probability, checks if it should play an audio mimicking the Inner Speech. If no, the loop waits for a certain time decided by the variable `timeAfterClip` before moving to the next iteration. If yes, `MasterCoroutine()` starts another coroutine in turn called `PlaySimulation (ConscienceSO conscienceSO)`. The selected `ConscienceSO` is provided as an argument to this coroutine.

`PlaySimulation()` adjusts the speed, volume, and chooses a clip from the range of values in `ConscienceSO` and calls `PlayOneShot()` on *MainAudioSource*. These ranges have been configured as per the characteristics of the mood it represents. A random value is selected from these ranges every iteration, which ensures variability and makes the Inner Speech more immersive.

4.5.2 Audio Symptoms

Voice Hallucinations

Voice Hallucinations have been implemented in a very similar way to Inner Speech. The class responsible for the handling of voice hallucinations is called `VoiceHallucinations`. The `GameObject` to which this script is attached also parents a set of `GameObject`. These `GameObject` have an audio source attached to them. The number of children `GameObject` determines the number of different audio sources that would play the hallucination clips. `VoiceHallucinations` maintains a list of all these `GameObject`. `VoiceHallucinations` uses a list `List<VoiceHallucinationSO> soundsSOs` where `VoiceHallucinationSO` is a scriptable object which extends `SoundsSO`. `VoiceHallucinations` contains the variables `pitch`, `distance`, and `timeAfterClip`. `VoiceHallucinations` overrides `Simulate()` in a very similar way by starting `MasterCoroutine()`. The only difference is that the indefinite loop inside `MasterCoroutine()` iterates through all the audio sources and starts `PlaySimulation()` on each one of them. Even `PlaySimulation()` is pretty much the same as `Conscience`. One key difference is that other than speed, volume, and clips (based on different moods), it also has extra parameters – distance and pitch that regulate the position of the audio source with respect to the HMD as well as the pitch.

This variation in the distance gives the impression that the source of the hallucination is moving. The change in the pitch distorts the audio and adds to the hallucination quality.

Ambient Sounds

`AmbientSounds` controls the functioning of ambient sounds that have been used to simulate losing focus in the experience. It, too, is very similar to other audio symptoms but still has its own characteristics. The biggest difference between this and Voice Hallucinations is that there is just one audio source, and that too is 2D in nature. This means that it doesn't have the effect of direction and depth, it is uniform for both ears and always has the same intensity no matter where the HMD device is. Another difference is that Ambient Sounds use audio clips of noises present in a regular environment — traffic, construction, beep, background chatter etc in contrast to the human dialogues used in Voice Hallucinations. `AmbientSounds` uses `AmbientSoundSO` that is extended from `SoundSO`. `AmbientSoundSO` stores a variable `activeTime` that decides the duration for the effect to play. `MasterCoroutine()` is pretty much the same for `AmbientSounds`, there is one major difference in `PlaySimulation()` though, instead of a random value for the volume, the volume starts at a particular level and gradually increases to it's maximum to mimic zoning-out. It stays at the maximum level for some time and then goes back to the initial value before turning off. A value is picked from the range of variable `activeTime`, which specifies for how long the audio clip plays the ambient sound clip.

4.5.3 Visual Symptoms

Hallucinations – Shapes

Shape hallucinations can be thought of as the visual analogy to audio hallucinations. The whole idea is exactly the same, but instead of playing the audio clips, the sprites are enabled/disabled with ranging parameters. Shape Hallucinations are controlled by the script called `ShapesHallucinations`. It uses `ShapesSO`, which also extends `SymptomsSO`. The parameters that are configured through `SymptomsSO` include the sprites of all the shapes that are used in the simulation, scale, distance, transparency of the sprites, and the time for which the sprite would be visible. The `GameObject` to which `ShapesHallucinations`, like Audio hallucinations, parents a number of `GameObject`. These `GameObject` are named *Shape* and have a `LookAt` script attached to them to ensure that the sprite is always facing the HMD. `GameObject` *Shape* further has two children – *Main*, and *Distort*. Both of these children have a `SpriteRenderer` component attached to them and are important for the functioning of the Distort Shader.

`MasterCoroutine()` is pretty much the same as `VoiceHallucinations`. It starts `PlaySimulation()` for each child with the `SpriteRenderer` component attached. It chooses all the parameters from `ShapesSO` at random, plug them into the `SpriteRenderer` component of the `GameObject` *Main* and displays the shape for the specified time. Shaders have been used to elevate the effects of visual hallucination. The two shaders used for the purpose are:

- **Distort Shader:** Distort Shader adds a wavy effect to the sprites. As mentioned earlier, two `GameObject` with `SpriteRenderer` work in sync for the Distort Shader to work properly. The `GameObject` *Main* use the Depth Shader (explained next) and renders the sprites normally. The `GameObject` *Distort* use the distort shader and is rendered on top of *Main* to give the desired wavy effect.

- **Depth Shader:** Depth Shader ensures that the sprite is always rendered on top of everything else and is not occluded by the geometry that is in front of the sprite in the environment.

Hallucinations – Bright Lights

`ParticlesHallucinations` is responsible for simulating little light spots spawning near the HMD using the `ParticleSystem` component. It uses `ParticlesSO` for configuration. `ParticlesSO` contains `Vector3 offset` that controls the position of the particles with respect to the HMD, and `MinMax<int> activeTime`, responsible for deciding the time for which the `ParticleSystem` remains active every iteration. `PlaySimulation()` is pretty straightforward as it simply enables the `ParticleSystem` and then disables it after the randomly chosen interval from `activeTime`.

4.6 Other Symptoms

This section is about the symptoms that do not extend the `Symptoms` class. They comprise — Scenes, Post-Processing, and behavioural apps (Sleep, Health, Bank, and Notifications). These are not covered extensively like the previous symptoms because their functioning is very straightforward. Each of these symptoms has a manager class responsible for applying the changes made to them in the `Settings` screen. The changes propagated by the event `SettingsUpdated` of `SettingsManager` are listened to by the manager, and it in turn applies them to these symptoms. For instance, if the scene is changed to *Day* in the settings, the event relays this to the `SceneLoader`, which loads the *Day* scene and unloads whatever the current scene is.

5. The Pilot Experiment

Having discussed the theoretical and technical details that are the cornerstones on which this project was created, the next step is to derive a verdict about its final usability. Therefore, this chapter is dedicated to outlining the process of the assessment design and testing that were undertaken to determine the final effectiveness of the VR project within an educational setting. In order to arrive at the final version of the software, it was essential to have undergone this rigorous procedure of testing to enable continuous improvement. The involvement of experts and students was enabled through a collaboration with Národní ústav duševního zdraví (NUDZ) [23] that represents the clinical teaching base for medical students of the 3rd Faculty of Medicine of Charles University and other master programs such as Psychology.

5.1 Aim

The aim of the study was to engage both experts and students in testing the software using VR Headsets, and then fill out a questionnaire about their experience. The questionnaire included questions encompassing the System Usability Scale or SUS (see Section 5.3.1), an assessment of individual symptom mechanics, and some open-ended questions for suggestions. The repeated feedback provided by some experts at every stage of the development was also essential in making reiterations in the software to include usability-related improvements, patching the bugs reported, and incorporating miscellaneous suggestions. The process was also vital to understand whether the software was effective in simulating symptoms with accuracy, as intended.

5.2 Target Group

For the purpose of this study, it was decided to bring both students and experts on board. The experts included both Psychiatrists and Psychologists. At the time of writing this thesis, a total of fourteen entries (six experts and eight students) have already been registered and processed. However, this number is expected to increase as the software undergoes more updates and refinements as a part of the future roadmap. In compiling results and making reiterations, data from all the entries were considered for the evaluation of the SUS score. But, while assessing the symptomatology relevance, only the answers from the experts were considered.

5.3 Study Design

Even though three experts were involved in the QA testing of the application at every stage of the development, only the assessment of the final prototypes will be reported here. The first rounds of the study were carried out in the month of April 2023 on the premises of NUDZ. This sample space involved only the students. The sample consisted of eight medical students of the 3rd Faculty of Medicine (Five females and three males between the age range 19-29, and a mean age of 23). The thesis supervisor and the thesis author were also present at the time to offer help and guidance required in application onboarding.

The assessment took place in one of the lecture halls at NUDZ. Students were asked to sit far apart from each other, to not cause any interference in each other's experience. Each student was provided with a VR headset. Each student was expected to use the application for at least twenty minutes. They were expected to test all the symptom presets at least once. They were free to experiment with the settings and other features of the application. They could also reach out for assistance in case they were stuck or could not understand something. In the end, they were asked to fill up the questionnaire (discussed in-depth in the upcoming sections).

Similarly, the experts also took part in the assessment, but due to different schedules, each one of them tested it at different times throughout the summer. The sample consisted of six experts, either psychiatrists or clinical psychologists (four females and two males between the age range 27-53 and a mean age of 38).

The names or any identity-relevant information were not recorded in the questionnaire or during the study. All the data gathered by the questionnaire meets GDPR compliance and in no way reveals any personal information about any of the subjects. The next section covers the different parts of the questionnaire in detail.

5.3.1 System Usability Scale

The System Usability Scale (SUS) is a simple, effective, and reliable tool used to measure the usability of a system. Invented by John Brooke in 1986, it is termed a "quick and dirty" usability test. SUS is composed of a questionnaire that includes ten questions. Each question is supposed to be answered using one of the five responses. These responses are – Strongly Disagree, Disagree, Neutral, Agree, Strongly Agree. SUS is widely used in testing all sorts of systems ranging from software and websites to hardware devices, and mobile phones [7] [34].

The set of ten questions that SUS includes are as follows:

	Strongly Disagree				**Strongly Agree**
1. I think that I would like to use this system frequently.	1	2	3	4	5
2. I found the system unnecessarily complex.	1	2	3	4	5
3. I thought the system was easy to use.	1	2	3	4	5
4. I think that I would need the support of a technical person to be able to use this system.	1	2	3	4	5
5. I found the various functions in this system were well integrated.	1	2	3	4	5
6. I thought there was too much inconsistency in this system.	1	2	3	4	5

7. I would imagine that most people would learn to use this system very quickly. | 1 | 2 | 3 | 4 | 5 |
8. I found the system very cumbersome to use. | 1 | 2 | 3 | 4 | 5 |
9. I felt very confident using the system. | 1 | 2 | 3 | 4 | 5 |
10. I needed to learn a lot of things before I could get going with this system. | 1 | 2 | 3 | 4 | 5 |

The following steps are taken to evaluate the SUS questionnaire data[34]:

- For the odd-numbered entries, 1 is subtracted from the score.
- For the even-numbered entries, the score is subtracted from 5.
- This converts the scale from 0 to 4 (0 being the lowest, and 4 being the highest.)
- The total is then multiplied by 2.5 to make the score range 0 – 100 from the initial range 0 – 40.

Attention must be paid to the fact that despite being in the range of 0 – 100, these scores are not percentages and must not be seen as such while evaluating the data. There are several methods to evaluate the results such as normalising the final score to obtain a percentile rank which is calculated based on already available SUS results from different studies. Another approach is to assess the SUS score against various scales, which is discussed in more detail in Section 6.1. Furthermore, SUS is not diagnostic and should be best used as a tool to classify the ease of use of the system in question [34].

5.3.2 Effectiveness of Individual Symptom Mechanics

The later part of the questionnaire addresses each symptom mechanic individually and asks the users to rate its effectiveness on a scale very similar to SUS – A scale ranging from 1 to 5 where 1 implies Not Effective and 5 implies Highly Effective.

To further test if the users were able to identify the various symptoms mechanics implemented in the game, there is an open-ended question asking them to list out all the symptoms, typical of the mental disorder in question, they experienced in the simulation.

5.3.3 General Feedback

The final part of the questionnaire is an open-ended question asking users to provide their overall impression of the application, feedback, and new things they would like to see in the simulation.

5.4 Limitations

While the questionnaire addresses the usability of the system, symptom mechanics, and the overall experience, there are no formal arrangements in place to assess immersion, presence, and cybersickness. These might be adopted in the latter, more refined versions of the study.

Other than the questionnaire, general feedback from the supervisor and other experts was accommodated at every step of the development of the application. This process has been instrumental in moulding the software to its present, refined version, along with refined onboarding instructions for first-time users.

6. The Result

This chapter marks the end of this thesis project, presenting the findings of the data gathered from the questionnaire. Even though the questionnaire has been explained to some extent in Section 5.3, it has also been provided as additional material with the thesis report. This includes the raw data gathered from the survey and its modified extensions. The chapter delves deeper into the data insights and tries to draw a conclusion.

The target group used Oculus Quest 2 as the testing device. Even though no test was designed to assess cybersickness caused by the experience, there were no complaints regarding it by the users.

6.1 Studying the SUS results

The first part of the questionnaire was devoted to the System Usability Scale. This section tries to explain the SUS score and then assesses it on various scales designed for it. Refer to Section 5.3.1 to know more about System Usability Scale, the set of questions involved, and the steps to calculate the SUS score.

The data gathered from the SUS test has been plotted as a box plot, as seen in Figure: 6.1. Each dot represents a user score. The *mean* value of the data turned out to be 80.89, with a *median* of 83.75, *q1* of 75, and *q2* value of 90. The height of the box or the *Inter-quartile range* was 15, i.e. 75 – 90. This suggests that 50 per cent of the scores lie in this range, and 75 per cent of the scores are above 75. The data looks promising, but there are still a lot of questions that need to be addressed – What should be the ideal score? Is 80.89 good enough? SUS has been in use for over three decades with thousands of responses. This helped the experts in the field to create a sort of benchmarking where the results of the SUS survey can be compared with the other results in the database. Various scales were also introduced to give the score a rating that was easily understood by anyone without worrying about the numbers [5].

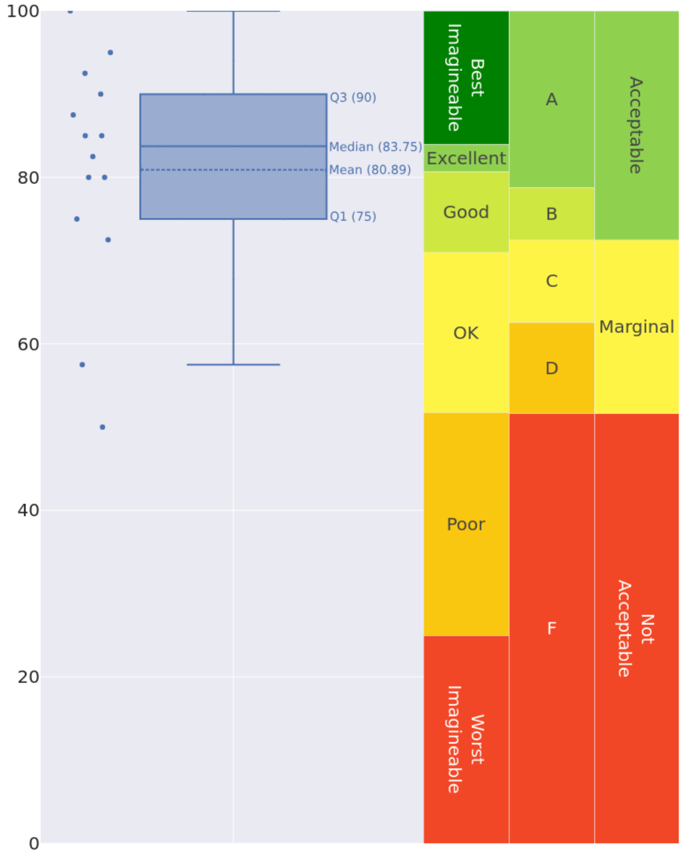

Figure 6.1: System Usability Scale Score of the Application against different scales

Following are the different scales on which the SUS has been measured. Each scale has been divided into smaller segments based on the guide by Jeff Sauro, a statistical analyst and UX expert [29]:

- **Percentile:** The mean of the SUS score is taken and normalised to obtain a percentile rank with respect to the other SUS scores. As per the scale developed by Sauro, the score at the 50th percentile is 68. This means any score above 68 is considered above average, and below 68 is considered below average. A mean score of 80.89 makes this application fall under the 90 - 95 percentile, which means it fares better than 90 - 95 percent SUS scores in the database.

- **Adjective Scale:** The second type of scale associates adjective words to different scale ranges, including — Best Imaginable, Excellent, Good, OK, Poor, and Worst Imaginable. As per Sauro, *Good* is just above average at

71, *Worst Imaginable* is anything below 25, and *Best Imaginable* is anything above 95. With a mean score of 80.89, the application gets the *Excellent* classification.

- **Grade Scale:** Grade Scale resembles the grading system commonly used in schools, where an A is the best possible score and an F means fail. C indicates the average mark. The application scores A on the Grading Scale.

- **Acceptability Scale:** The last scale included classifies the SUS percentile in terms of acceptability. The three classifications are — Acceptable, Marginal, and Not Acceptable. Suaro classifies anything above 70 as acceptable, anything below 50 as unacceptable, and the middle portion is termed marginal. On this scale as well, the application does well with an Acceptable rating.

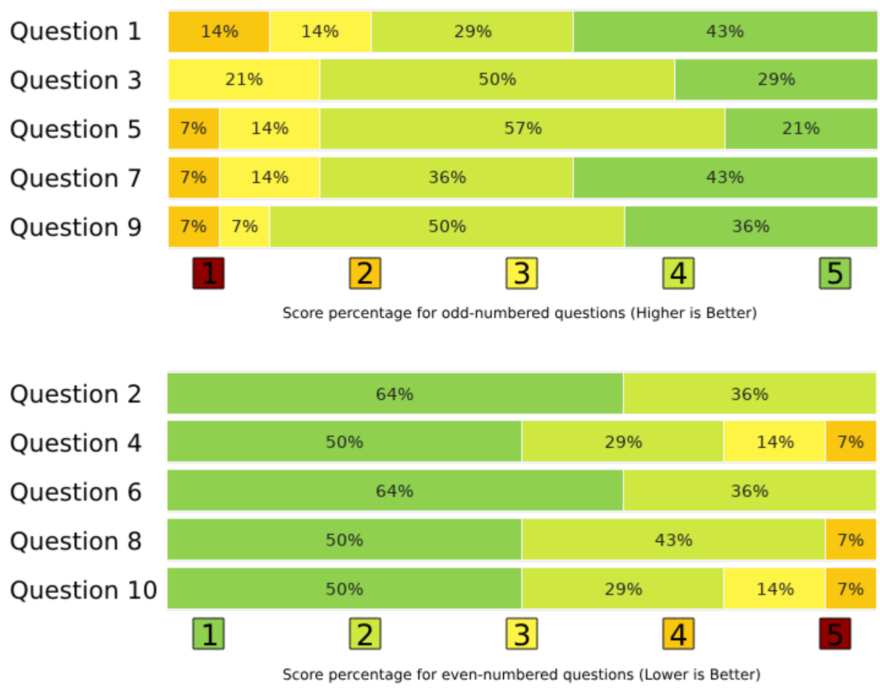

Figure 6.2: Five-Point scale percentage of each SUS question

A second plot has been created to examine the performance of individual SUS questions. The responses of the users have been divided based on the five-point scale, and their percentages have been plotted. The way the SUS questionnaire was designed, a higher score is optimal for the odd-numbered questions, and for the even-numbered questions, a lower score is better. (Figure: 6.2) segregates even and odd-numbered questions into groups, and it can be seen that for the former group, a scoring of 4 (agree, pale green) and 5 (strongly agree, green) is dominant. The lowest score received in the first half is 2 (disagree, orange). Question 1 has the highest percentage of 2s scored. The overall sentiment of the users for the odd-numbered questions was above neutral. The latter part of the figure depicts the even-numbered questions. In this case, a score of 1 (strongly disagree, green) and 2 (disagree, pale green) are the most prevalent. The lowest score received is 4 (agree, orange) and Questions 4,8 and 10 have the same percentage of 4s scored. The overall sentiment of the users for the even-numbered questions was below neutral. Results from both groups point in favour of the usability of the application.

6.2 Evaluating Individual Symptom Mechanic

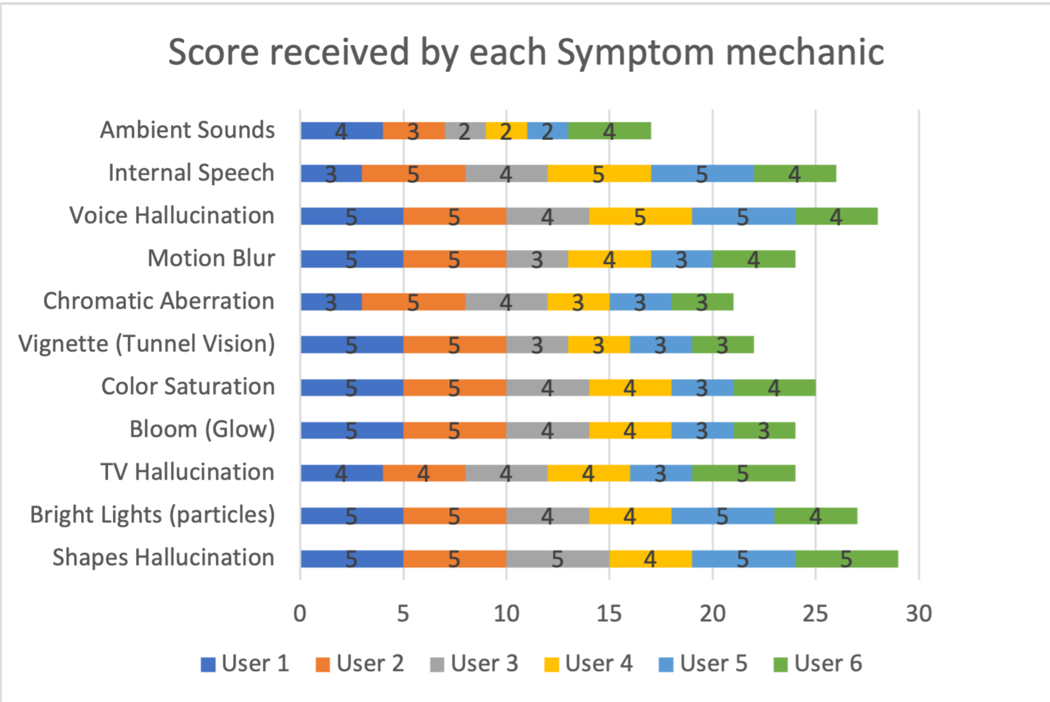

Figure 6.3: Scoring for Individual Symptom Mechanic

The later part of the questionnaire focused on the symptomatology, and the respective mechanics in the simulation to simulate them. Since this section concerned symptomatology simulated by the application, only the data entries of the experts (six in total) were taken into consideration. Similar to the SUS scale, the users were asked to rate each symptom mechanic on a scale of 1 – 5 (higher is better). The results for each symptom were added up and compiled, as can be seen in Figure: 6.3. As per the figure, the mechanic *Ambient Sounds* scored the lowest, with a total score amounting to 17 out of 30. On the other hand, the mechanics *Internal Speech, Voice Hallucinations, Shapes Hallucinations,* and *Bright Lights* fared pretty well, with each one of them scoring above 25. *Shapes Hallucinations* was liked the most, with a near-perfect score of 29. Other than *Ambient Sounds*, all the other mechanics scored above 20. Clearly, the next steps would be to study this feedback in addition to the open-ended answers and refine the mechanics even further.

Another important metric that has been studied with the help of the assessment is the different symptom mechanics the users could identify on their own while inside the simulation without any external help or guidance. This is key to identifying the mechanics that are evident and the mechanics that go unnoticed during the simulation. The users were asked to list out all the symptoms typical for the selected mental disorder preset. Since the responses were in text form, they were scanned for keywords representing any of the symptoms and a rough estimation data chart was created. Figure: 6.4 plots a clustered bubble chart of the data obtained. The radius of the circles is proportional to the number of users that reported the particular symptom. From the figure, it is evident that mechanics involving the environment, and the lighting were noticed the most, along with *Inner Speech*, and *Audio-Visual Hallucinations*. The mechanics that depended on the use of the personal device that concerned the behavioural symptoms (sleep, diet, weight etc.) have been underreported. This implies that the users are not able to fully understand or use the personal device as a part of the simulation. The next steps would include enhancing the UX of the personal device, more audio clips for the Inner Speech concerning the behavioural symptoms, environmental visualisations, and additional onboarding steps to ensure no mechanic goes unnoticed or underreported.

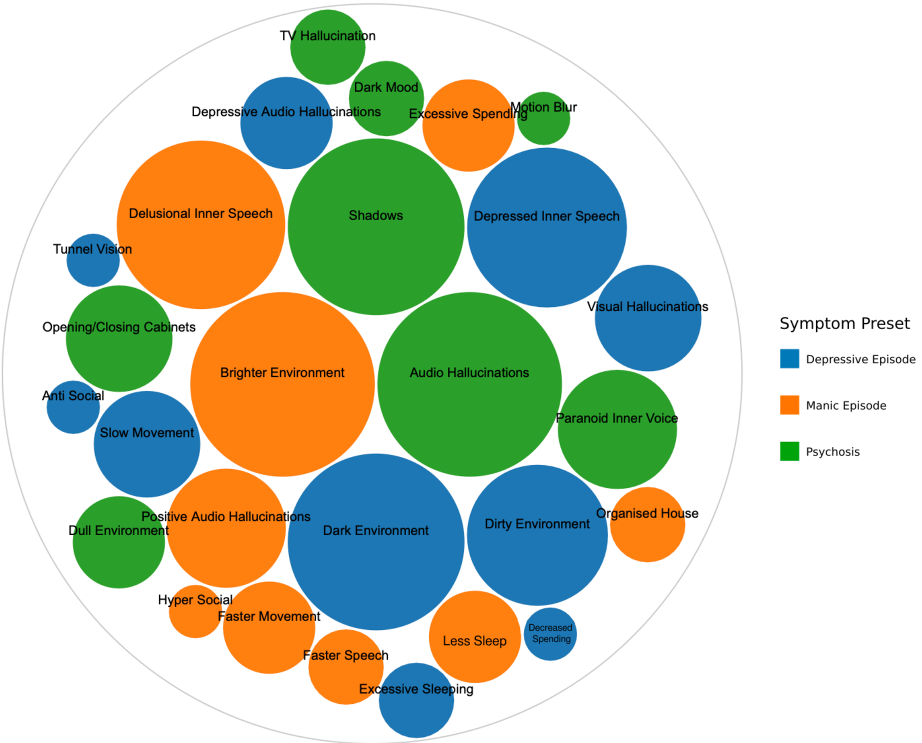

Figure 6.4: Visualisation of different symptom mechanics of each symptom preset that the users identified

6.3 Takeaways from the General Feedback

The last question regarding general feedback received different kinds of responses, touching upon all kinds of topics. Overall feedback for the application remained positive, but there were suggestions involved. Some users also reported a change in their own mood as they used the simulation, a positive sign pointing towards the immersion of the experience.

A common complaint that some users had was not being able to distinguish between the Inner Speech and Audio Hallucinations effectively. Other suggestions included the addition of more dialogues, moods, environmental props etc.

All the data from the questionnaire proved to be very insightful and would help in shaping the future roadmap of the development of the project. This has been discussed in more detail in the next chapter.

Further Scope

The overall positive feedback that the project received is an indication that it is headed in the right direction. This also serves as a huge motivation for future work and maintenance of the project. The first course of action from here would, of course, be to study the feedback from the questionnaire thoroughly, point out the shortcomings of the project, address the most critical ones, and patch them.

Since the ultimate aim of the project is to be used as an educational tool to study mental disorders, localisation is another major feature that is a part of the future roadmap. There is a plan to add support for global languages (starting with Czech), as language complications should not be a hindrance in the path of education.

One thing that the project is clearly missing in its current form is the human touch. Being a thesis project with very limited resources, most of the art and assets for the project were either improvised using AI tools or not created in the best quality, including the voice recordings used for Inner Speech and Hallucinations. This is another major thing that has been planned for the project; to replace the artwork and the assets with better, higher-quality versions.

The personal device used in the simulation to show the behavioural changes is also very static and generic. It, too, has a lot of scope for improvement. There can be settings added to make it more personalised so that when the user uses it, they don't just see made-up stats but something in the proximity of their real-life stats and can relate with the behavioural changes on a higher level.

VR is a technological domain that is evolving every day, and that too very rapidly. There are things possible in VR today that were unimaginable ten years ago, and this is going to be highly likely in the next ten years as well. The next steps would be to keep in touch with the evolving VR tech and brainstorm ideas to implement more and more symptom mechanics and symptom presets, while at the same time polishing the pre-existing ones.

Conclusion

The topic of Mental Health still remains taboo in the society. There isn't enough awareness, which gives rise to stigma, prejudice, and ignorance towards people suffering from mental disorders. This, in turn, may prevent people from seeking help. A significant number of Mental Health cases don't get medical treatment at all. However, all this can be turned around with awareness in society and a better understanding of these conditions. Modern medicine is evolving at an exceptional rate. The application of VR in the field of medicine is one big example. Various research studies have also been carried out to test the efficacy of VR as a tool for studying mental disorders, and they all seem quite promising. This project was one such attempt at simulating mental health disorders in VR, with future prospects of it being used as an educational tool for the training of medical professionals.

The aim of the project was to build a configurable VR simulation in contrast to the pre-recorded VR videos[6] that are used in the studies. The project tried to identify various symptoms that are prevalent in mental disorders, implemented mechanics to simulate those, and then provided the settings to configure those mechanics and club them with other mechanics to mimic various mental disorders such as Bipolar Disorder (both phases including Mania and Depressive Episodes), and Schizophrenia.

A study involving the students and experts from NUDZ was set up to study the application further. The study involved testing the application and filling up a questionnaire at the end. The questionnaire sought to assess the application on the basis of system usability, quality of individual symptom mechanics, and the overall simulation in general. System Usability Scale was used to assess the usability, and the results were quite promising. Individual symptom mechanics also received neutral to positive responses from the group. Further steps would include studying the feedback, incorporating necessary changes, and maintaining the project as more advances are made in the field of VR.

Bibliography

[1] Arm Limited. Gaming engines. https://www.arm.com/glossary/gaming-engines. Accessed: June 13, 2023.

[2] Lexica Art. Lexica: An ai-powered art generator. URL https://lexica.art/. Generative AI used for generating image assets.

[3] American Psychiatric Association. and American Psychiatric Association. *Diagnostic and statistical manual of mental disorders : DSM-5*. American Psychiatric Association Arlington, VA, 2013.

[4] Atmospheric House (Modular). URL https://assetstore.unity.com/packages/3d/environments/urban/atmospheric-house-modular-192712. House asset used in the simultation.

[5] Aaron Bangor, Philip Kortum, and James Miller. Determining what individual sus scores mean: Adding an adjective rating scale. *Journal of usability studies*, 4(3):114–123, 2009.

[6] Jasmine Elizabeth Banks, Jennifer Tichon, Geoffery Ericksson, Sean Ivermee, Kevin Burrage, and Peter Yellowlees. A virtual environment to simulate the experience of psychosis. *Digital Image Computing, Techniques and Applications (DICTA)*, pages 869–878, 2003.

[7] John Brooke. Sus: A quick and dirty usability scale. *Usability Eval. Ind.*, 189, 11 1995.

[8] Electronic Arts Inc. (EA). Advanced Real-time Post-Processing. https://www.ea.com/frostbite/news/advanced-real-time-post-processing-using-gpgpu-techniques#:~:text=Post%2Dprocessing%20techniques%20are%20used,of%2Dfield%20and%20tone%20mapping. Accessed: June 16, 2023.

[9] Nicholas J. Formosa, Ben W. Morrison, Geoffrey Hill, and Daniel Stone. Testing the efficacy of a virtual reality-based simulation in enhancing users' knowledge, attitudes, and empathy relating to psychosis. *Australian Journal of Psychology*, 70(1):57–65, 2018. doi: 10.1111/ajpy.12167. URL https://doi.org/10.1111/ajpy.12167.

[10] Unity Technologies Inc. Bloom. https://docs.unity3d.com/2017.3/Documentation/Manual/PostProcessing-Bloom.html, . Accessed: June 16, 2023.

[11] Unity Technologies Inc. Chromatic Aberration. https://docs.unity3d.com/2017.3/Documentation/Manual/PostProcessing-ChromaticAberration.html, . Accessed: June 16, 2023.

[12] Unity Technologies Inc. Color Grading. https://docs.unity3d.com/560/Documentation/Manual/PostProcessing-ColorGrading.html, . Accessed: June 16, 2023.

[13] Unity Technologies Inc. Motion Blur. https://docs.unity3d.com/2017.3/Documentation/Manual/PostProcessing-MotionBlur.html, . Accessed: June 16, 2023.

[14] Unity Technologies Inc. Vignette. https://docs.unity3d.com/2017.3/Documentation/Manual/PostProcessing-Vignette.html, . Accessed: June 16, 2023.

[15] Patrick Kenny, Thomas D. Parsons, Jonathan Gratch, Anton Leuski, and Albert A. Rizzo. Virtual patients for clinical therapist skills training. pages 197–210, 2007.

[16] Andreas Koch, Ingolf Cascorbi, Martin Westhofen, Manuel Dafotakis, Sebastian Klapa, and Johann Peter Kuhtz-Buschbeck. The Neurophysiology and Treatment of Motion Sickness. *Dtsch Arztebl International*, 115(41):687–696,

2018. doi: 10.3238/arztebl.2018.0687. URL https://www.aerzteblatt.de/int/article.asp?id=201532.

[17] Eleven Labs. Eleven labs: Generative ai text to speech & voice cloning. URL https://beta.elevenlabs.io/. Generative AI used for generating audio assets.

[18] Alexander Kc Leung and Kam Lun Hon. Motion sickness: an overview. *Drugs in Context*, 8, 2019. doi: 10.7573/dic.2019-9-4.

[19] Boaz Levy and Emily Manove. Functional outcome in bipolar disorder: The big picture. *Depression research and treatment*, 2012:949248, 01 2012. doi: 10.1155/2012/949248.

[20] Henry E. Lowood. Virtual reality. *Encyclopedia Britannica*, June 2023. URL https://www.britannica.com/technology/virtual-reality. accessed: 2023-07-18.

[21] Gustavo Hernández Melgarejo. Vr: immersion and a sense of presence, 2023. URL https://tecscience.tec.mx/en/science-communication/virtual-reality-immersion-and-a-sense-of-presence/.

[22] Richard Nordquist. Inner speech, aug 2020. URL https://www.thoughtco.com/what-is-inner-speech-1691070.

[23] NUDZ. Národní ústav duševního zdraví. URL https://www.nudz.cz. Accessed: July 13, 2023.

[24] National Institute of Mental Health. Bipolar disorder, 2023. URL https://www.nimh.nih.gov/health/topics/bipolar-disorder.

[25] National Institute of Mental Health. Schizophrenia, 2023. URL https://www.nimh.nih.gov/health/topics/schizophrenia.

[26] OpenAI. Dall-e 2: An image generator trained to create images from text. URL https://openai.com/dall-e-2/. Generative AI used for generating image assets.

[27] José Miguel Padilha, Paulo Puga Machado, Ana Ribeiro, José Ramos, and Patrício Costa. Clinical virtual simulation in nursing education: Randomized controlled trial. *J. Med. Internet Res.*, 21(3):e11529, March 2019.

[28] Wulf Rössler. The stigma of mental disorders: A millennia-long history of social exclusion and prejudices. *EMBO Reports*, 17(9):1250–1253, 2016. doi: 10.15252/embr.201643041.

[29] Jeff Sauro. Interpret sus score, September 2018. URL `https://measuringu.com/interpret-sus-score/`.

[30] R.E. Shannon. Introduction to the art and science of simulation. In *1998 Winter Simulation Conference. Proceedings (Cat. No.98CH36274)*, volume 1, pages 7–14 vol.1, 1998. doi: 10.1109/WSC.1998.744892.

[31] Mel Slater. Immersion and the illusion of presence in virtual reality. *British Journal of Psychology*, 109(3):431–433, 2018. doi: https://doi.org/10.1111/bjop.12305. URL `https://bpspsychub.onlinelibrary.wiley.com/doi/abs/10.1111/bjop.12305`.

[32] Unity. URL `https://unity.com`. Link to the official website of Unity3D Engine.

[33] Unreal Engine. URL `https://www.unrealengine.com/en-US`. Link to the official website of Unreal Engine.

[34] Usability.gov. System Usability Scale. `https://www.usability.gov/how-to-and-tools/methods/system-usability-scale.html`. Accessed: July 14, 2023.

[35] WHO. Mental disorders — who.int. `https://www.who.int/news-room/fact-sheets/detail/mental-disorders#:~:text=In%202019%2C%201%20in%20every,of%20the%20COVID%2D19%20pandemic.`, Jun 2022.

[36] P. Winkler, T. Formanek, K. Mlada, A. Kagstrom, Z. Mohrova, P. Mohr, and L. Csemy. Increase in prevalence of current mental disorders in the

context of covid-19: analysis of repeated nationwide cross-sectional surveys. *Epidemiology and Psychiatric Sciences*, 29:e173, 2020. doi: 10.1017/S2045796020000888.

List of Abbreviations

1. **2D** - Two Dimensional
2. **3D** - Three Dimensional
3. **AI** - Artificial Intelligence
4. **API** - Application Programming Interface
5. **BD** - Bipolar Disorder
6. **GB** - Gigabyte
7. **GPU** - Graphics Processing Unit
8. **HMD** - Head Mounted Display
9. **IPD** - Interpupillary Distance
10. **LTS** - Long Term Support
11. **MRTK** - Microsoft Mixed Reality Toolkit
12. **NUDZ** - Národní Ústav Duševního Zdraví
13. **OS** - Operating System
14. **PC** - Personal Computer
15. **REM** - Rapid Eye Movement
16. **RTX** - Ray Tracing Texel eXtreme
17. **SDK** - Software Development Kit
18. **SRP** - Standard Render Pipeline
19. **SUS** - System Usability Scale
20. **SoC** - System on a chip

21. **URP** - Universal Render Pipeline

22. **VR** - Virtual Reality

23. **WHO** - World Health Organisation