Interim Report: Evaluating the Impact of Phobia-Motivated Experiences delivered through a Virtual Environment

Submitted December 21, 2019, in partial fulfillment of
the conditions for the award of the degree **BSc Computer Science.**

Denis Stepanov
20027064

Supervised by Dr Dave Towey

School of Computer Science University of Nottingham Ningbo China

Abstract

Virtual Reality has a huge potential to help to evaluate and asses post-traumatic stress disorders (PTSD), delivered through Virtual Reality Environments (VREs). It helps to analyze the stress levels more safely and provide a possible solution or treatment for the user with a specific fear or phobia. One of the greatest advantages of Virtual Reality (VR) is that individuals know that the reality they are facing is virtual, but their body and mind act as if it was real. Therefore, it is easier to help healthcare professionals to understand the impact of PTSD on users without real-life exposure to stressful situations. However, background research shows that there are not many VR applications in the healthcare industry at the moment that are working closely with VR and fear evaluations. This project intends to provide a system where a healthcare professional could see the results of body sensors attached to the person (wearable technology) and analyze the impact of VR exposure to stressful situations. The sensors on the smartwatch should show the typical reaction to the fear, such as changing heart rate. The challenge is to provide a safe environment for the user who would be immersed in VRE of their fear and to access as much data as possible using wearable technology.

Contents

List of Figures

List of Tables

List of Abbreviations

CC BY Creative Commons Attribution License

ECG Electrocardiogram

FDA Food and Drug Administration

GVR Google Virtual Reality

HR Heart Rate

IDE Integrated Development Environment

iOS iPhone Operating System

macOS Macintosh Operating System

MR Metamorphic Relation

MT Metamorphic Testing

PTSD Post-Traumatic Stress Disorder

SDK Software Development Kit

VRET Virtual Reality Exposure Therapy

VRE Virtual Reality Environment

VR Virtual Reality

Chapter 1

Introduction

Since the release of the first VR headsets in the 1960s, VR technology has been evolving and progressing all over the world [4]. Now, there are more affordable headsets on the market, such as Google Daydream, Oculus Go and others, which interest many researchers, especially in the healthcare industry [5]. Considering that the first VR interactions were introduced more than 50 years ago, there is still a lot of potential to grow in terms of helping patients and doctors in the field. VR systems and Virtual Reality Environments (VREs) are becoming more common in the healthcare applications for training, educating professionals, and treating patients [6]. This chapter describes the motivation and background of this project, as well as provides the aims and objectives of the work.

1.1 Background

The technological changes in the healthcare industry significantly affected the education of healthcare professionals. A study shows that on average it takes up to 30 years in the field of study to get a job in the healthcare industry [7]. However, the advances in technology offer new innovative tools when it comes to education in the healthcare industry. VR is now used in various fields of medical treatment, including professional education and training, diagnostic assistance, rehabilitation and diverse mental treatments for patients [8].

Anatomy is one of the most significant fields of study in the healthcare industry [9]. For the past five years, almost 90% of the applications on VR were made to simulate surgery, treating patients' mental disorders and precisely diagnosing various mental diseases [10]. However, there are still not many applications that are present on the market that help to analyze the problem and the diseases. The study of brain activity has been one of the biggest interests in the VR healthcare industry for the past ten years [9]. Almost all of the brain activity research in VR is concentrated on human feelings that can be achieved virtually, such as fear or stress. A recent study shows that more than 90% of people have various phobias, and it is important to know how to analyze and treat them [9].

1.2 Motivation

Searching through the VR healthcare applications that are on the market now, it can be concluded that there are not many products available that help assessing personal fears. Limbix VR library shows that it is very important for the patient to completely immerse in the situation of fear and try to present it as realistically as possible [2]. There are several applications, such as Limbix [2] and C2phobia [1], which provide the necessary experiences, but none of them collect any personal data to evaluate the impact of the virtual phobia environment. The problem behind displaying phobias in VR is the lack of a proper system which would help healthcare professionals to analyze the world of phobias and to assist their patients according to the proper diagnoses.

There are products in the industry that are made specifically for data collection. Qardio [3] is the arm monitor that provides details for the users about their health, such as heart rate and blood pressure. It is an example of wearable technology connected with a phone application to display data. However, Qardio only functions as a standalone measurement tool and does not include any VR environment to be tested with. Therefore, it is worth investigating wearable technology combined with virtual reality environments as a way to evaluate the impact of phobia-motivated experiences.

1.3 Aims and Objectives

The main objective of this project is to create a system capable of providing VREs for the user with phobia(s) and deliver the patient's data to the healthcare professional(s) using wearable technology. It is not mandatory to obtain same hardware and software tools as stated in this project. Any VR headset, Apple Smart Watch and a mobile device with gyroscope, a sensor for detecting the orientation of the device, would be enough to replicate this project. The key objectives of this project are:

1. Investigate previous and current methods that are used to treat patients with phobia(s).
2. Design a comprehensive system of cooperating hardware and software tools to display VREs addressing phobia(s).
3. Develop a mobile device application communicating with wearable technology which sends and displays the patient's information to the healthcare professionals. Apply Metamorphic Testing (MT) [11] to VR application testing process.
4. Test the system on various software and hardware specifications, such as various headsets, mobile devices and smart watches.
5. Collect data and analyze the results that the application provides to evaluate the impact of phobia-motivated experiences.

1.4 Interim Report Outline

This chapter has summarized the VR history in the healthcare industry, provides general background on the related work and states aims and objectives of this project. Chapter 2 describes previous work in VR healthcare industry and gives specific details about what products are currently available on the market. Chapter 3 outlines software requirements specifications for this project and highlights the risks that could be taken during the implementation stage. Chapter 4 presents the ideas on how to combine the software and hardware tools to complete the project. Chapter 5 shows the results of implementation steps to create the VR application and first testing evaluations. Finally, Chapter 6 expresses the project management and reflections on the work that has been done.

Chapter 2

Background and Related Work

This chapter outlines major studies on VR exposure as a way to treat phobias, and wearable technology as a valid method to measure stress levels. This chapter also includes the latest products on the market related to VR healthcare industry. Additionally, this chapter explains the better characteristics of the project over previous work in related field.

2.1 VR Phobia Research

The idea of analyzing and treating PTSD using VR exposure started a few years ago with the basic implementation of virtual environments displaying various 3D images [6]. This section describes the potential advantages of VR exposure to stressful situations on patients and the risks that could be caused by it. It also argues on wearable technology becoming one of the valid tools to collect personal stress level data from the human body.

2.1.1 VR exposure as a Safe Method to Assess Stress

Considering the available therapies to treat PTSDs nowadays, there are two main approaches: Virtual Reality Exposure Therapy (VRET) and in-vivo therapy [12]. VR therapy is a well-known practice nowadays when it comes to assessing a person's stress level. The main idea behind it is to immerse a user into the VRE where they would face a stressful situation. VR exposure therapy is a complete opposite of in-vivo therapy, where the patients or users would be accompanied by a psychologist or any related-field healthcare professional. In-vivo therapy is time-consuming, expensive and requires a lot of personal background knowledge about the individual when it comes to phobia treatment. VR exposure is challenging that method and tries to provide a robust approach treating PTSDs at minimal cost and risk for the user.

VR therapy has been applied in many case studies, such as in fear of flying [13], nervousness in social situations, or social phobia [14] and acrophobia, a fear of heights. [15]. All participants went through VR therapy and had been immersed in specific situations where they were feeling under stress. After several sessions, it was confirmed that VR therapy lowered participants' stress levels and reduced anxiety. Participants stated the effectiveness of VR exposure therapy as a method to reduce stress levels when facing their phobias. For example, 93% of the people who were afraid of flying and went through VR exposure therapy had flown after several sessions without any problem [13]. Cases presented here and also various other studies indicate that VR exposure is an effective method to treat phobias and possibly replace expensive and time-consuming in-vivo therapy.

2.1.2 Social Phobia Study

Social anxiety is a disorder where a person faces a stressful situation just by being around other people. One of the studies [14] focuses on VR exposure application surveys and 3D images to help analyze and treat social phobia.

The users had to interact with several VR scenarios, such as walking down the street, entering a party or having a conversation in a group of people. By the end of all sessions, the participants noted that their social anxiety had disappeared or dropped to a minimum (decreased 75% on average), improving their way of life and changing the way they think. It is stated that, on average, only 21.29 minutes were needed to complete one session and 7 sessions in total to see the improvement. Participants also mentioned that the realism of the images was better than expected while no real-life footage was used to produce virtual reality environments.

2.1.3 Smart Watch as a Stress Measuring Tool

When it comes to the ways of measuring stress levels, there are several approaches, including body sensors, wearable technology, and invasive healthcare tools. Assessing fear levels is a complicated task because people react differently to stressful situations. There are many aspects to be taken into consideration, such as the measures that display the stress level, invasive or noninvasive technologies and the validity of the software and hardware that records the data. A recent study about human emotions indicates that a person's body temperature, blood pressure, and heart rates are changing, rising in all cases when a person is facing a stressful situation [16].

There are not many tools nowadays that would allow us to measure the stress levels with all metrics available. However, wearable technology is considered to be one of them. Disregarding the fact of not being able to record all: blood pressure, skin temperature and heart rate, the smartwatch industry focuses on one of those measures - the Heart Rate (HR). For example, a study on the validation of Apple Smart Watch shows that 90% of all the heartbeats that were taken during the testing stages were valid and correct, showing same results as industrial health rate monitors [17]. It stipulates that wearable technology nowadays can compete with industrial heart rate monitors on the open market as the main tool to measure heart conditions. Furthermore, choosing Smart Watch technology for this project is one of the innovative methods because it has never been used before with VR to collect personal data while the user is immersed in VREs (to author's knowledge).

2.2 Industry Research

As stated in Chapter 1, there are multiple applications on the market trying to combine VR technology with the healthcare industry and apply it to the patients' treatment, surgery simulations, and professional training. However, most of those applications lack the functionality to fully implement the product to evaluate the impact of VR experiences. This section describes the products in detail, their potential impact on the market and this project.

2.2.1 C2Care

C2Care [1] is one of the largest products on the market nowadays that helps treating phobias using VREs. Moreover, it provides VREs for functional disorders, eating disorders and addictions, such as tobacco, alcohol or gambling. According to the information at C2Phobia website [1], a part of C2Care, the current selection of VR content is very large and most of them are implemented using 3D technology (Figure 2.1).

Figure 2.1: C2Care Main Page [1]

To gain access for full content, the user has to pay for the subscription or buy a VR kit from C2Phobia directly. After the registration for the demo course, the user would be brought to their account page where it is possible to launch a VR environment for any specific phobia from the VR environments list (Figure 2.2). C2Phobia works with wired headsets, such as HTC Vive or Oculus Rift (Figure 2.3) and mobile headsets, such as Oculus Go and Samsung Gear VR.

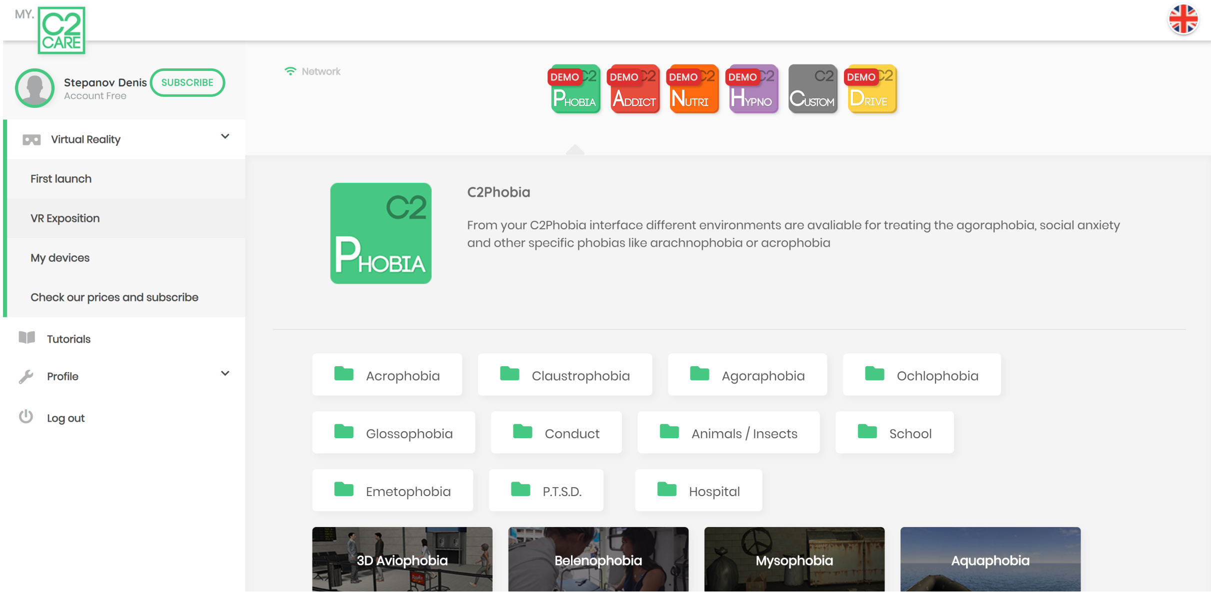

Figure 2.2: C2Phobia Environments Selection [1]

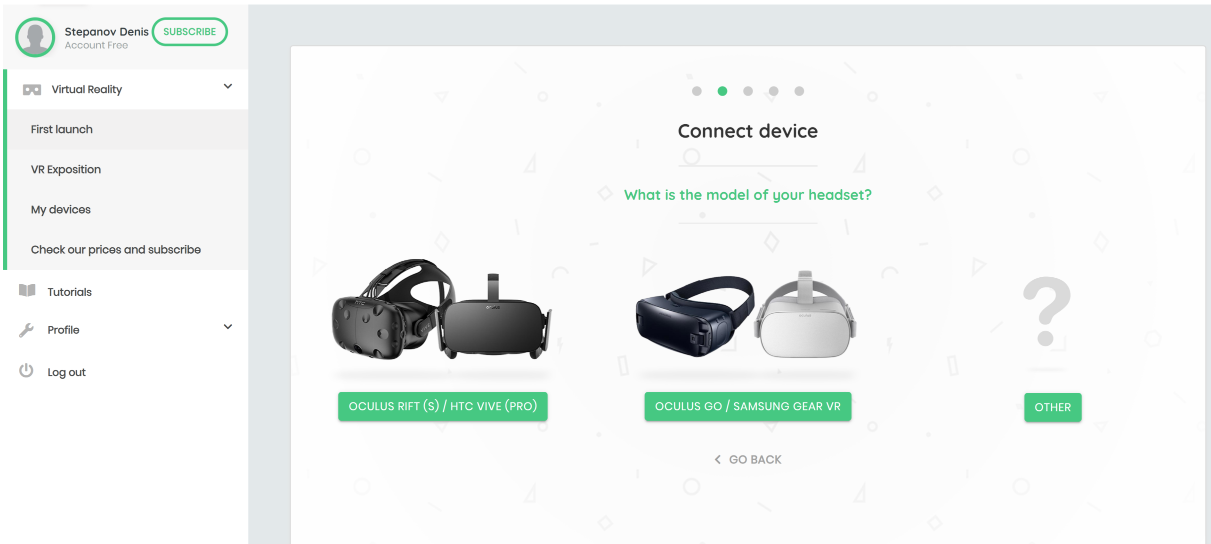

Figure 2.3: C2Care VR Headsets Selection [1]

C2Phobia is one of the largest Food and Drug Administration (FDA) approved systems on the market trying to implement VR technology in the healthcare industry to treat disorders, addictions, and phobias. However, C2Phobia does not collect any personal data nor notifies healthcare professionals about the sessions that the user has taken. C2Phobia works only as a VR environment player for displaying phobias, but it never evaluates the impact of the experience that the user has gained. Nevertheless, C2Phobia is still one of the first and largest products on the market that shows the combined technology working between the healthcare and VR industry.

2.2.2 Limbix

Limbix is a digital therapeutics for mental health [2]. It is a product approved by FDA for researching mental health issues using VR experiences. Limbix built its foundation from multiple pieces of research on mental trauma, phobias, and anxieties, providing a good example of VR technology in the healthcare industry. Limbix package includes: VR Kit (Figure 2.4), consisting of VR wireless headset, tablet, a docking station to charge and transport the equipment and access to Limbix application. However, Limbix is not cheap: it comes at 2000 US dollars per kit with all features included.

Figure 2.4: Limbix VR Kit [2]

The users can not register for the Limbix program as the website does not have a signup option. It is only possible to view the information on the website about the research, product pipelines, exposure therapy environments, and other functionality. Looking at the description of the software from Limbix website, I have noticed that they use real-world images, which is different from C2Phobia. Additionally, their VR Library has a lot of various environments available with extra features, such as changing attributes for height during VREs experiences or choosing a level of stress. Moreover, most of the VR environments come in multiple languages, which is important to have for users from different countries.

The downside of Limbix is that all of their VR content is hosted by YouTube [18] which is not accessible in various countries due to restriction policies. Furthermore, similar to C2Phobia, Limbix does not collect any personal data. Limbix kit is used only to play the VR environments as video files.

2.2.3 Qardio

Qardio [3] is one of the best modern ways to monitor users' health activity. Qardio specializes in measuring heart rate, electrocardiogram (ECG), respiratory rate and skin temperature. Qardio has different devices, such as QardioArm, QardioBase and Qardio-Core, each one for different use. All of them are focused on checking the user's heart activity and reporting it to the application on the phone. Qardio works for both iPhone Operating System (iOS) and Android, including wearable technology (Figure 2.5). It comes with an application for mobile phones with many functionalities, including storing, sharing and evaluating the data of the user in real-time.

Figure 2.5: QardioArm Kit [3]

Affordable modern monitoring technology comes with each device's price lower than 200 US dollars. The registration is free and all additional features are included. Qardio also has a special bundle for healthcare professionals with extra features which are approved by many administrations all over the world.

Qardio, Limbix, and C2Care are the three main products that are present in the healthcare market nowadays. They are efficient in what they do, but they are lacking some functionality, such as collection of personal data, live images, wearable technology or VR headsets for mobile phones. Uniting the features from the products above, I plan to design the system where all of the features combined would produce a sufficient method to evaluate the impact of phobia-motivated experiences using the VR environment.

Chapter 2 summarized the background information from the related work and available products on the VR healthcare market today. Next chapter introduces software requirements specifications for the project.

Chapter 3

Project Specification

This section states the main requirements for the project. There are two types of requirements: functional and non-functional. Functional requirements define what the system should do, while non-functional requirements display how the system works [19]. The software requirements below are only applicable for the implementation of the VR application and do not include any aspect of wearable technology development yet. The software requirements engineering process was completed based on common practices and general thoughts on mobile applications development. Sommerville [19] also talks about user specifications as a part of requirements engineering. However, the full list of requirements is not yet completed, thus some of the them could be modified, removed or replaced by other features of the software in the future.

3.1 Functional Requirements

1. Anybody can start the application and use it immediately.
2. The application is required to give the user various options from the first launch.
3. The application must be supported on the iOS platform (version 10.0 and higher).
4. The application must have a list of videos (VR Videos Gallery).
5. The application must include button interaction using gyroscope (Google VR Pointer [20]).
6. The application must have interactive buttons to change between the screens.
7. The application must provide the information or tips about VR and how to use it.
8. The application must inform the user about various phobia before selecting them from video VR gallery.
9. Quit option should be available for users in the main menu.

3.2 Non-Functional Requirements

Usability
The user interface should be intuitive and easy to use, where the buttons and text fields should be clear and readable. It means that the color of the buttons and the size of the text should be visible on a clear background. Any user should understand the meaning of each button, text field, or figure that are used in the application without any additional explanation. The buttons' names should be self-explanatory and lead to specific scenes or pages with the corresponding functionality. The usability could be measured by the average run time and pause time on each scene to see if the user has complications with understanding the context of the application.

Performance
The application should load within 5 seconds of clicking the icon in offline mode. The selection of any button or feature in the application should take at most 2 seconds to load. There should be no frozen scenes or stuck moments during video playback. The application should run on mobile devices with gyroscope that allow VR view (split screen). If there is a bug or an error with the application, it should be removed with a patch or a fix in the next application release.

Online Access
As VR video content could be hosted online, the user should have internet connection to access media files. However, it is not required to have an online access at all times as the main part of the application does not require access to the internet.

Operating Environment
The application should work on iOS-supported devices, exclusively iPhones, version 10.0 and higher.

Security
VR application should not ask or store any personal data. It would work only as a video player for the user. However, Smart Watch application will have an access to personal data, such as heart rate readings. The application should collect the data and send it to the mobile application of the healthcare professional with user's personal details. The details should include only general information, which could be name, weight, height, and age. None of the applications should have access to personal data unless the user gives permission to the application before the start.

Safety
The application should show safety tips for VR usage during the first run to provide a secure environment for the users. The application should prevent the start of any video unless the user gives consent and reads the safety tips.

3.3 Priority & Risk

During the software requirements engineering process, it is important to know which tasks need to be implemented first, or prioritized over the others. It is a common practice to evaluate all software specifications and place them in order of importance [19]. However, almost all requirements have their risks, a probability of loss during the development stage. Each requirement has a chance of a failure based on either time, resources or any other factor that the developer usually does not anticipate. Therefore, a developer should produce an estimate chart where they know what tasks should be completed first. The value of loss probability could be measured by common practices and previous work in the related field. For example, implementing the menu for mobile application on Android platform could take longer than iOS depending on the programming skills of the developer. According to the time estimates from previous work on building menus for Android application, the developer should check the timeline for the current project to build the menu for iOS. The developer should then calculate the probability of loss on building the menu, the chance that the task will not be completed on time.

Table 3.1 specifies what tasks are most important and chances of a specific task failure during the development stage. Priority is ordered from 1 to 4, where 1 is the most important and 4 is the least important. Loss Probability is ordered from 1% to 99%, where 1% is unlikely to be failed and 99% is most likely to be failed. None of the requirements can be lower than 1% because there is always a chance of failure in any specification. A requirement also can not be at 100% loss probability because then it is neglected from the start and does not count as a proper software specification.

ID	Name	Priority	Loss Probability
1	Application Start	1	5%
2	Menu	1	10%
3	iOS Support	1	10%
4	VR Gallery	2	30%
5	Buttons Interaction	3	10%
6	Return to Main Page	3	20%
7	Tips	4	30%
8	Video Descriptions	4	10%
9	Quit	4	5%

Table 3.1: Priorities and Risks of Functional Requirements

This chapter summarized software specifications requirements and showed the risks during the process of requirements engineering. Next chapter introduces the design of the project and tools that would be needed to complete it.

Chapter 4

Design

This chapter introduces the design of the project. It includes the descriptions of the software, developer tools, and the hardware that will be used to implement the applications.

At the early stages of the design process, it was decided to develop two applications for iOS-supported mobile devices. One of the applications would display the scenarios of stressful situations through videos in VR. The second application would collect data from the smartwatch and use online services to deliver the data to the responsible individual(s). The choice was based on the availability of the hardware at the moment and access to various services online on different platforms. Considering the cost options, latest news and availability of help resources online, the option to develop a mobile application for iOS as the main platform was confirmed.

Unity [21] was chosen as the main tool to create the VR application. Unity is a game engine which works as Integrated Development Environment (IDE) and helps to produce the prototype of the applications in a short amount of time using C Sharp (C#) scripting and Google VR Software Development Kit (SDK) libraries. While the application was under the first steps of the development, it was noticed that the application could easily be converted to the Android platform, using Unity services. Even though requirements specifications (Chapter 3) do not state Android as the main build platform, there is still the potential to develop the application for Android users.

There is no confirmation at the moment on the design for the second application to implement wearable technology functionality. It is still possible to merge two applications in one and run the healthcare services in the background of the VR app. However, not much design thought was put into Smart Watch application yet and it will be investigated more after the full completion of VR application development.

This chapter presents the software and hardware details, developer tools that will help to build the project. Next chapter shows the implementations steps to create the VR application.

Chapter 5

Implementation

This chapter specifies the implementation steps of the VR application development based on the software requirements specifications (Section 3.1 - 3.2) and design plans (Chapter 4). This chapter covers the hardware and software preparations, choice of environment and tools, features description and testing process.

5.1 Software & Hardware Preparations

Based on the selection of hardware and software from the design (Chapter 4), the choice was to implement the VR application on iOS mobile devices. An iPhone X was chosen as the main testing device, MacBook Air as the main machine for writing scripts and application compiling, and Google Daydream 2 as the VR headset to help interact with the application. It is worth noting that the VR headset is not required to build an app and only needed during the system testing phase. Moreover, any phone with the gyroscope, a sensor to detect the orientation of the device, can be used for the implementation of VR applications. It is also possible to run the application on the emulator, but for VR application development, testing on emulators is not recommended because emulators do not have a gyroscope.

For the software part, it was decided to use Unity tools and Google VR library because it has code samples and demos on how to build VR applications. Xcode [22] should be available on any macOS device as a free application from the App Store. It is also recommended to update all software and hardware to the latest version available before the start of the project. For additional information about run specifications and tools descriptions, refer to Appendix B.

5.2 Features & User Interface

VR application is managed by the head movement and the gyroscope on the mobile device. All interactions around the application must be done with the controller or focus function. Gaze [20] is a method of interacting with objects in VR using cursor focusing. A user looks at the object with the cursor that is located in the middle of the headset and can perform various actions with it (additional information about Gaze function is available in Appendix A). In the example of our application, the user will have four options at the application start: tips, VR video gallery, test video and quit (Figures 5.1 & 5.2).

Figure 5.1: User Interface PC View

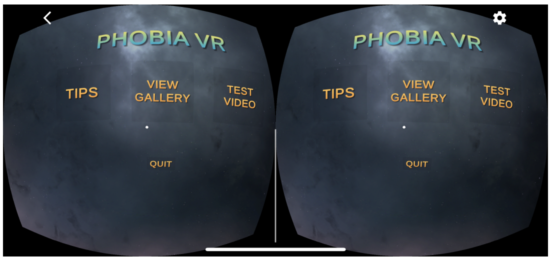

Figure 5.2: User Interface VR View

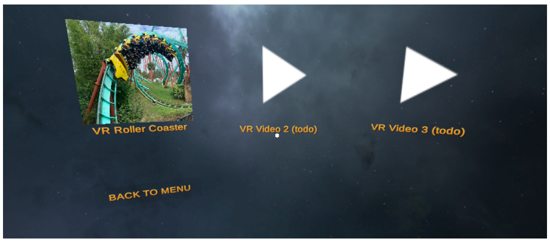

Figure 5.3: VR Video Gallery

Figure 5.4: Roller Coaster Video

The buttons on the screen are linked to C# scripts to switch the windows of the application. For example, the VR video gallery button will lead to the page which will consist of VR-ready videos (Figure 5.3). Test Video button will launch the video immediately (Figure 5.4).

Most functionality of the VR application is already implemented by now. The only tasks that are left to do are helpful VR tips for users' safety and expansion the video gallery.

5.3 Testing

As the main portion of the VR application was finished, I have started to test it on various headsets and mobile devices. The testing phase was divided into several subsections. Only some of them were initiated and included in the subsections below.

In the future, I plan to include Metamorphic Testing (MT) [11] to address an oracle problem in VR applications. "Oracle" is a method in software testing by which testers can determine whether the software under test is valid or not. However, oracle can be hard to apply or expensive in terms of software resources. Therefore, MT found its applications in modern software as a tool to alleviate the oracle problem by including metamorphic relations (MRs) [23]. MRs are properties of intended functionality of the software and must include multiple executions of the software.

To author's knowledge, there are no current applications of MT in VR industry, thus it is worth investigating and apply MT as a part of testing method for the project.

5.3.1 Testing on Different Headsets

The application was tested on an iPhone X with iOS version 13.2. Five headsets were used: Google Cardboard, Google Daydream 2, Merge VR, Park VR, and XiaoZhai Z6. The headsets were chosen based on different prices, online reviews in their price ranges and overall grade by companies in the VR industry.

Since the headsets that were used for testing only work as holders for mobile devices and do no interfere with the run of the application, it was decided to split the ranking in several categories: quality, comfort, accessibility and available controllers. The choice of categories and ranking was based purely on developer's opinion. Quality ranking is based on the headset lenses image quality. Comfort ranking is ordered from most comfortable headset to use while seated, standing and moving around. Accessibility category is showing the available screen size of the mobile device. Finally, the last category shows whether the headset has native controllers. Table 5.1 represents testing for all five headsets in all categories (1 - terrible, 2 - below average, 3 - average, 4 - above average, 5 - exceptional):

Headset	Quality	Comfort	Accessibility	Controllers
Google Cardboard	1	1	4 - 6	No
Google Daydream	4	4	Up to 6.2	Yes
Merge VR	5	3	5 - 6	No
Park VR	2	2	4 - 6	No
XiaoZhai Z6	4	5	Up to 6	No

Table 5.1: Headsets Testing

5.3.2 Testing on Media Files with Different Size

During the first round of the VR application testing, it was noticed that the mobile device does not handle large media files properly. The screen would get stuck at one point or the application would crash. Considering that it is a known bug in the program, it was decided to implement a Fault Based approach [24] as the next step of the testing process. Five iOS devices were used: iPhone X, iPhone 5s, iPhone 7, iPhone 7+, iPhone XR and iPhone 11. All devices had iOS version 13.2 except iPhone 5s had version 12.2 at the moment of testing. The tables below show various devices running the application with different size media files.

Phone	Video Freezes	App Crashes
iPhone 5s	✓	
iPhone 7		
iPhone 7+		
iPhone X		
iPhone XR		
iPhone 11		

Table 5.2: Video Size <100MB

Phone	Video Freezes	App Crashes
iPhone 5s	✓	✓
iPhone 7	✓	
iPhone 7+	✓	
iPhone X		
iPhone XR		
iPhone 11		

Table 5.3: Video Size <500MB

Phone	Video Freezes	App Crashes
iPhone 5s	✓	✓
iPhone 7	✓	
iPhone 7+	✓	
iPhone X	✓	
iPhone XR		
iPhone 11		

Table 5.4: Video Size 1GB - 2GB

Phone	Video Freezes	App Crashes
iPhone 5s	✓	✓
iPhone 7	✓	
iPhone 7+	✓	
iPhone X		✓
iPhone XR		
iPhone 11		

Table 5.5: Video Size >2GB

By the completion of the first test runs, it was concluded that the video files of large size are not handled well by mobile devices that are older than iPhone 7 (Release date: September 2016). The application has crashed on an iPhone 5s twice during the testing period. For future evaluations, I plan to use Unity's Profiler [25] to find the worst performance stages and fix them. The Profiler is a tool that shows performance information about the application. It can connect to devices on the network or devices connected to the machine to test how the application runs on the intended release platform.

Google Daydream 2 [26] is the best option for application development at this moment, as it is the best overall headset available and has SDKs from Google. XiaoZhai Z6 [27] is a valid runner-up and I plan to continue using it for future application development as a second testing option.

5.4 Media Licensing Issues

Considering the issue with the media file size during the run of the VR application, there was also another problem during the implementation. The video that was used during the testing process (Sections 5.3.1 - 5.3.2) was retrieved from YouTube [18] which was published under Creative Commons Attribution (CC BY 4.0) license [28]. The license allows distribution with several restrictions, such as indicating if changes were made to the content and giving appropriate credit to the author. However, YouTube itself has a strong policy about non-official downloads. Therefore, I will not download any videos from online sources for the final product of my project, but I will consider to film and edit the videos myself.

This chapter specified the implementation steps of the VR application development based on the software requirements specifications and design plans. Next chapter shows project management and reflections on the project development.

Chapter 6

Progress

This chapter discusses the progress made so far in the project as well as a critical assessment of that progress.

6.1 Project Management

Project development started with collecting the necessary hardware and software tools (Section 5.1). Then, the implementation of VR application to display phobia(s) and Smart Watch application to collect data. The development of the applications will take the most significant part of the project because it requires a lot of time to write code. Testing and data analyzing would also take a significant part of project development time considering the Ethics Committee approval time estimates.

The Gannt chart below (Figure 6.1) describes the project plan. Using the Waterfall methodology [29], I planned to conclude testing on each major stage of te software implementation. Some of the tasks could be combined due to the importance levels, risk management and wait time for other various tasks that do not depend on the developer, such as ethics forms approval. Moreover, the spread of the workload is put more on the first half of the year because of total number of credits in second semester. The number of classes and credits is more in the second semester than the first, thus I intend to finish most of the application implementation before the start of second semester. Currently, the project is under testing phase of VR application which was anticipated at the beginning of the year to be around similar time that was stated on the timeline.

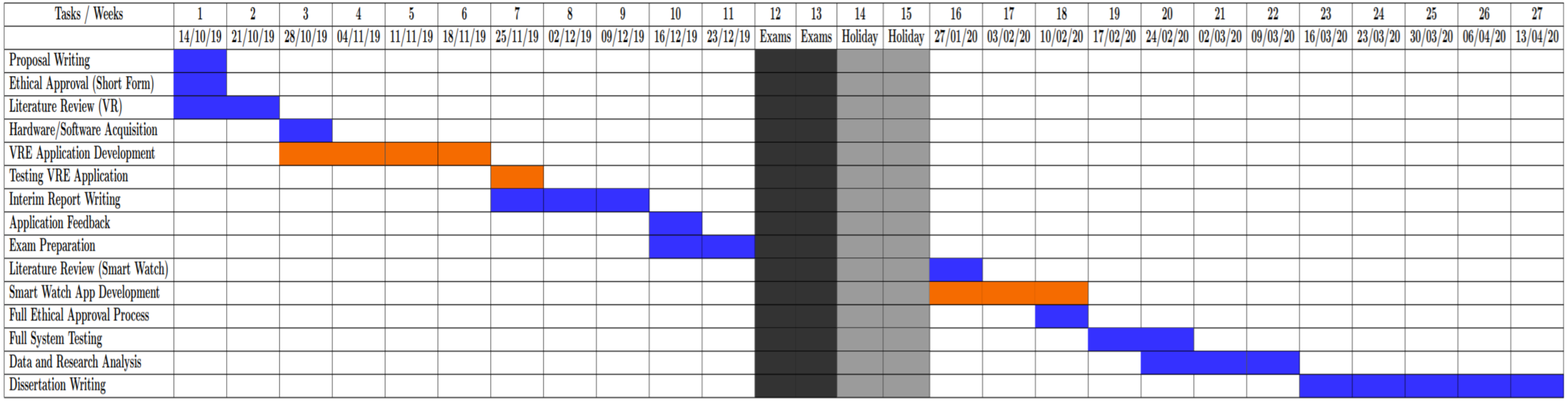

Figure 6.1: Original Timeline

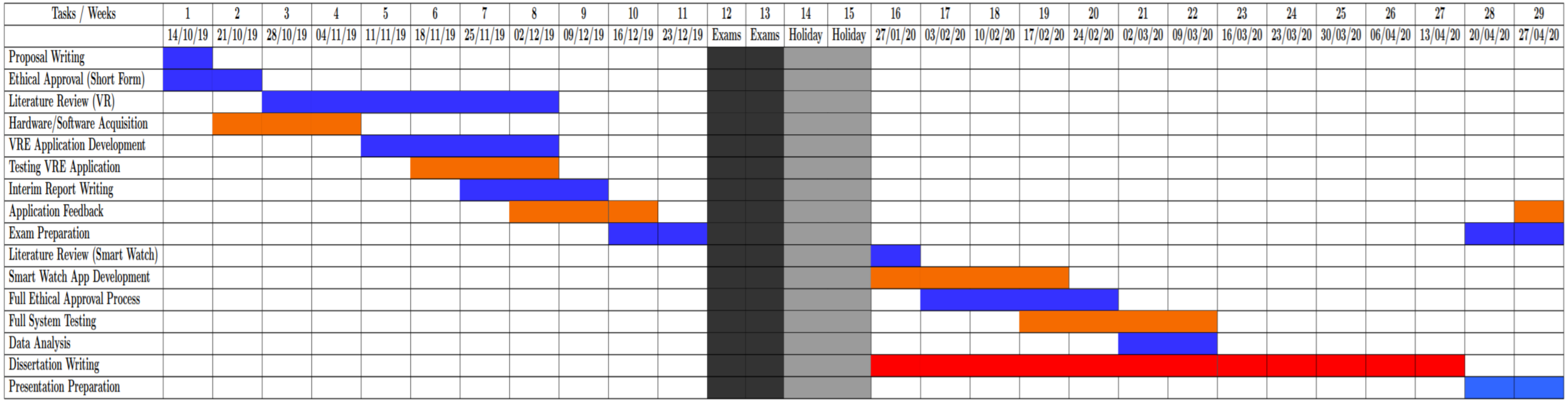

Figure 6.2: Updated Timeline

However, after ten weeks into project development, several changes were made. The waterfall methodology did not seem suited at the time and was replaced by Agile [30]. Most of the tasks ran in parallel with each other and did not depend on the completion of the processes before them. The implementation of the VR application was done alongside with testing on each phase of development. The dissertation writing part was not assigned properly before because most of the research progress was done during the first weeks and continued through the whole timeline of the project. Therefore, the restructured format and timeline were applied to the new chart with adjusted predictions for the second half of the year in Figure 6.2. "Presentation" row was added according to official timeline from the course page because it was missing from the original project management timeline.

6.2 Reflections

The scope of this project increased to include filming VR-ready videos for the VR application as well as learning a new programming language to ensure the functionality of the applications under a specific environment. These additions were made to increase the usability of the project and establish that there were no ethical conflicts involved. Additionally, the level of project difficulty was increased because of those functionalities. Therefore, required measures were taken, such as revised time management and change of software engineering methodology.

The speed of the project development is good and deadlines are strictly maintained by all deliverable submitted on time. The work that has been done during the first weeks covered most parts of the related field research and later followed through to the application development stage based on the information that was collected. However, considering the timeline that was set at the beginning of the project, some of the difficulties were met during software and hardware installation. The time that has been taken to solve the problems seriously affected the work and timeline of the project development. Nevertheless, at this critical halfway point, I think the project is standing strong on supporting its ideas. It is now important to continue maintaining the same pace for the next stages of project development and consider doing extra work on the next application based on wearable technology.

This chapter described the project management during the first half of the project and introduced new timeline with according changes based on work reflections.

References

[1] Rus-Calafell. C2Phobia - Treating phobias in virtual reality. `https://www.c2.care/en/c2phobia-treating-phobias-in-virtual-reality`. Accessed 24-October-2019.

[2] Limbix. About Limbix. `https://www.limbix.com/about`. Accessed 24-October-2019.

[3] Qardio. About Us. `https://www.getqardio.com/about-us`. Accessed 24-October-2019.

[4] Pietro Cipresso, Irene Alice Chicchi Giglioli, Mariano Alcañiz Raya, and Giuseppe Riva. The past, present, and future of virtual and augmented reality research: A network and cluster analysis of the literature. *Frontiers in Psychology*, 9, Jun 2018.

[5] Meixu Chen, Yize Jin, Todd Goodall, Xiangxu Yu, and Alan Bovik. Study of 3d virtual reality picture quality. *IEEE Journal of Selected Topics in Signal Processing*, 2019.

[6] Lan Li, Fei Yu, Dongquan Shi, Zongjun. Tian, Jiquan. Yang, Xingsong. Wang, and Qing Jiang. Application of virtual reality technology in clinical medicine. *American Journal of Translational Research*, 2017.

[7] Fabrizia Mantovani, Gianluca Castelnuovo, Andrea Gaggioli, and Giuseppe Riva. Virtual reality training for health-care professionals. *CyberPsychology & Behavior*, 6(4):389–395, 2003.

[8] Nikhil Ichalkaranje and Ajita Ichalkaranje. *Intelligent paradigms for assistive and preventive healthcare*. Springer, 2011.

[9] Kalpana Srivastava, Suprakash Chaudhury, and Ram Das. Virtual reality applications in mental health: Challenges and perspectives. *Industrial Psychiatry Journal*, 23(2):83, 2014.

[10] Cecília Sik Lányi. Virtual reality in healthcare. *Intelligent Paradigms for Assistive and Preventive Healthcare Studies in Computational Intelligence*, page 87–116, 2010.

[11] Tsong Yueh Chen, Fei-Ching Kuo, Huai Liu, Pak-Lok Poon, Dave Towey, and Zhi Quan Zhou. Metamorphic Testing: A Review of Challenges and Opportunities. *ACM Comput. Surv.*, 51(1):4:1–4:27, January 2018.

[12] Alexander Miloff, Philip Lindner, Peter Dafgård, Stefan Deak, Maria Garke, William Hamilton, Julia Heinsoo, Glenn Kristoffersson, Jonas Rafi, Kerstin Sindemark, and et al. Automated virtual reality exposure therapy for spider phobia vs. in-vivo one-session treatment: A randomized non-inferiority trial. *Behaviour Research and Therapy*, 118:130–140, 2019.

[13] Barbara Olasov Rothbaum, Larry Hodges, Samantha Smith, Jeong Hwan Lee, and Larry Price. A controlled study of virtual reality exposure therapy for the fear of flying. *Journal of Consulting and Clinical Psychology*, 68(6):1020–1026, 2000.

[14] Cristiane Gebara, Tito De Barros-Neto, Leticia Gertsenchtein, and Francisco Lotufo-Neto. Virtual reality exposure using three-dimensional images for the treatment of social phobia. *Revista Brasileira de Psiquiatria*, 38(1):24–29, 2015.

[15] Paul Emmelkamp and Lex Hulsbosch. Virtual reality treatment versus exposure in vivo: a comparative evaluation in acrophobia. *Behaviour Research and Therapy*, 40(5):509–516, 2002.

[16] Maria Teresa Valderas, Juan Bolea, Pablo Laguna, Montserrat Vallverdu, and Raquel Bailon. Human emotion recognition using heart rate variability analysis with spectral bands based on respiration. *2015 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, 2015.

[17] David Hernando, Surya Roca, Jorge Sancho, Álvaro Alesanco, and Raquel Bailón. Validation of the apple watch for heart rate variability measurements during relax and mental stress in healthy subjects. *Sensors*, 18(8):2619, Oct 2018.

[18] YouTube. Youtube. `https://www.youtube.com/`. Accessed: 2019-12-18.

[19] Ian Sommerville. *Requirements Engineering*, page 82–118. Pearson, 9 edition, 2016.

[20] Google Developers. Gvrreticlepointer prefab. `https://developers.google.com/vr/reference/unity/prefab/GvrReticlePointer`. Accessed: 2019-12-18.

[21] Unity. Unity Real-Time Development Platform. `https://unity.com/`. Accessed: 2019-11-15.

[22] Apple Inc. Xcode. `https://developer.apple.com/xcode/`. Accessed: 2019-12-18.

[23] Zhi Quan Zhou, De Huang, Zongyuan Yang, Haitao Huang, and Tsong Chen. Metamorphic Testing and Its Applications. In *8th International Symposium on Future Software Technology*, 10 2004.

[24] Bernhard Aichernig and Corrales Delgado. From faults via test purposes to test cases: On the fault-based testing of concurrent systems. *Fundamental Approaches to Software Engineering Lecture Notes in Computer Science*, page 324–338, 2006.

[25] Unity Technologies. Profiler overview. `https://docs.unity3d.com/Manual/Profiler.html`. Accessed: 2019-12-18.

[26] Google. Daydream. `https://vr.google.com/daydream/`. Accessed: 2019-12-18.

[27] XiaoZhai. XiaoZhai3 Z6 VR Headset. `https://www.xiaozhai360.com/z6/`. Accessed: 2019-12-19.

[28] Creative Commons. Creative commons license deed. `https://creativecommons.org/licenses/by/4.0/`. Accessed: 2019-12-18.

[29] Sundramoorthy Balaji and Sundararajan Murugaiyan. Waterfall vs. V-Model vs. Agile: A comparative study on SDLC. *International Journal of Information Technology and Business Management*, 2(1):26–30, 2012.

[30] David Parsons, Ramesh Lal, Hokyoung Ryu, and Manfred Lange. Software development methodologies, agile development and usability engineering. In *18th Australasian Conference on Information Systems*, 12 2007.

Appendix A

Gaze Function

The examples in Section 5.2 do not show the actual interaction with buttons. On the computer, it is always a click of the mouse or keyboard button, but in VR it is different. All interactions in VR are done by Gaze [20] or by using the controllers (Google Daydream has one, but Cardboard does not).

Gaze is a method of manipulating or interacting with any object in VR (button, figure, text, etc) just by looking at it. Gaze consists of mainly 3 things: Pointer, Timer and Image. Pointer is usually a pre-build package made by Google VR (GVR) and shows a white point (dot) always in the middle of the screen. Timer is the function that will count down the time until there is a call for any other function or action. Image is the figure that will be displayed when the timer is loading.

There were 3 options to implement the Gaze in my application:

1. Pointer - do not implement Timer or Image functions, just look at something and perform the action.

2. Pointer and Timer - do not add any animations to the functions, look at the object for some time and then do the assigned function.

3. Pointer, Timer and Image - perform complete, game style animation. Look at the something and see loading animation that is related to the timer.

The last was chosen to implement in the VR application - Pointer, Timer and Image. No relevant tutorial or documentation about Gaze in Unity was found online. The screenshot below shows the in-process action of pushing the button - half of the time has passed, green circles are indicating it. When the circle will be finished (full on-screen), the scene will change (Figure A.1). The script for Gaze is built in C# language and available in Section C.

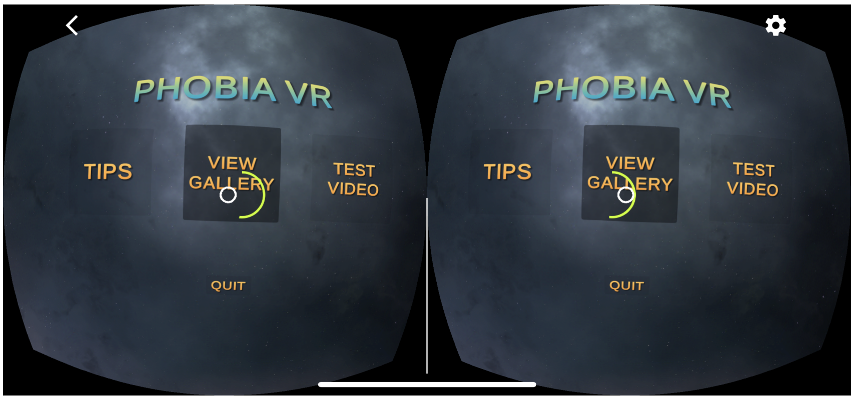

Figure A.1: GVR Menu Interaction

Appendix B

Software & Hardware Specification

Based on information from Section 5.3, it can be concluded which hardware and software is required to run the VR application. Assumed devices and tools for a smooth run of the app are listed in the Table B.1.

Hardware / Software	Version	Cost	Comments
iPhone X	iOS 13.2	$999 USD	
MacBook Air	macOS 10.15	$999 USD	
Google Daydream View 2	N/A	$99 USD	Not available anymore
Unity	2019.f.02.1	Free	
Xcode	11	Free	Download from AppStore
Apple Developer License	N/A	$99 USD	Required to create iOS apps

Table B.1: Hardware & Software Specifications

The versions of the hardware are minimal requirements and it is unknown whether the application would work with the versions lower than the ones in Table B.1. Furthermore, there are no other platform specifications available except iOS and macOS since the application was only tested on those platforms. It is planned to include support for other platforms after the end of final version release for iOS.

Appendix C

Code Samples

The code sample below is the script for Gaze function to initiate the timer and animation when the users focuses on a button or text field. It is written in C# programming language and works only when the user look directly at the object with the cursor in the middle of the headset.

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.UI;
using UnityEngine.Events;

public class GVRButton : MonoBehaviour
{
    public Image GVRImage;
    public UnityEvent GVRClick;
    public bool GVRStatus;
    public float TotalTime = 2;
    public float GVRTimer;
    // Update is called once per frame
    void Update()
    {
        if (GVRStatus)
        {
            GVRTimer += Time.deltaTime;
            //GVRImage.fillAmount += GVRTimer / TotalTime ;
            GVRImage.fillAmount += 0.01f;
        }
        if (GVRTimer > TotalTime)
        {
            GVRClick.Invoke();
        }
    }

    public void GVROn()
    {
        GVRStatus = true;
    }
```

```
    public void GVROff()
    {
        GVRStatus = false;
        GVRTimer = 0;
        GVRImage.fillAmount = 0;
    }
}
```

Listing C.1: C# Gaze Interaction

```
using System.Collections;
using System.Collections.Generic;
using UnityEngine;
using UnityEngine.SceneManagement;

public class SceneChanger : MonoBehaviour
{
    public void LoadScene(string sceneName)
    {
        SceneManager.LoadScene(sceneName);
    }
}
```

Listing C.2: Scene Changer