

GNG1103  
**Final Design Report**

# **The Ottawa Hospital Virtual Reality Treatment Simulation**

Submitted by

**GNG 1103-B, TEAM B11**

Adi Makkar, 300060213

Kerollos Guerguis, 300121743

Andrea Boulanger, 300143278

Luke Marshall, 300077329

Alison Nandram, 300056006

12/12/2019

University of Ottawa

## **Abstract**

---

Patients are always looking for ways to make their time spent at the hospital more comforting and relaxing, yet they only have so much they could do while spending most of their time just lying on the bed during the actual operation/scan being performed on their body. The Virtual Reality Treatment Solution changes their lives by providing the patients of the Ottawa hospital by providing them with a virtual environment of their treatments before they physically go through the treatment process. Our hypothesis was the videos we provide would be comforting and would provide the patients with a very interactive and calming scene which would want them to go through the treatment process and instead of being frightened about it they would have a clear idea of what is happening to their bodies. The experimental and prototyping results showed that the group's hypothesis was a very much what the group wanted as the final output and created a strong basis for us to finish our project as planned.

## Table of Contents

---

Abstract .....	i
Table of Contents .....	ii
List of Figures .....	iii
List of Tables .....	iv
List of Acronyms .....	v
1 Introduction.....	6
1.1 Product Development.....	<b>Error! Bookmark not defined.</b>
1.1.1 Unity .....	<b>Error! Bookmark not defined.</b>
2 Need Identification and Product Specification Process .....	7
3 Conceptual Designs .....	11
4 Project Plan, Execution, Tracking & Bill of Materials.....	15
5 Prototyping, Testing and Customer Validation. ....	16
6 Final Solution.....	24
7 Conclusions and Recommendations for Future Work .....	29
8 Bibliography .....	30

**List of Figures**

---

Figure 1. Concepts for Design.....14

Figure 2. The code involved in Prototype I.....17

Figure 3. Result on Unity for Prototype I.....19

Figure 4. The code involved in Prototype II for buttons.....20

Figure 5. Result for one of the Stereoscopic videos for Prototype II.....20

Figure 6. Results for Buttons for Prototype II.....21

Figure 7. Initial Screen.....22

Figure 8. Code which prompts the videos to run on unity.....22

Figure 9. Stopping Criteria for Testing.....23

Figure 10. Final Initial Screen.....24

Figure 11. Final First Screen.....25

Figure 12. Final Second Screen.....25

Figure 13. Final Third Screen.....26

Figure 14. Final code for buttons.....27

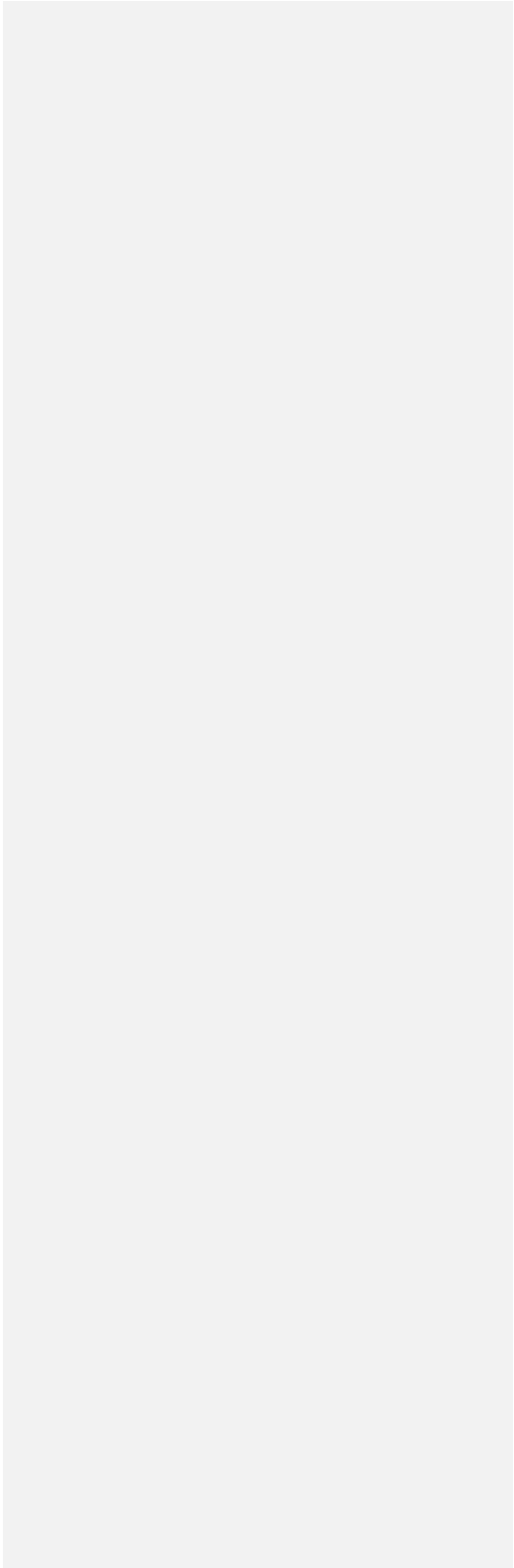
Figure 15. Final code for prompting the videos.....28

**List of Tables**

---

Table 1: Benchmarking for Designs ..... 9

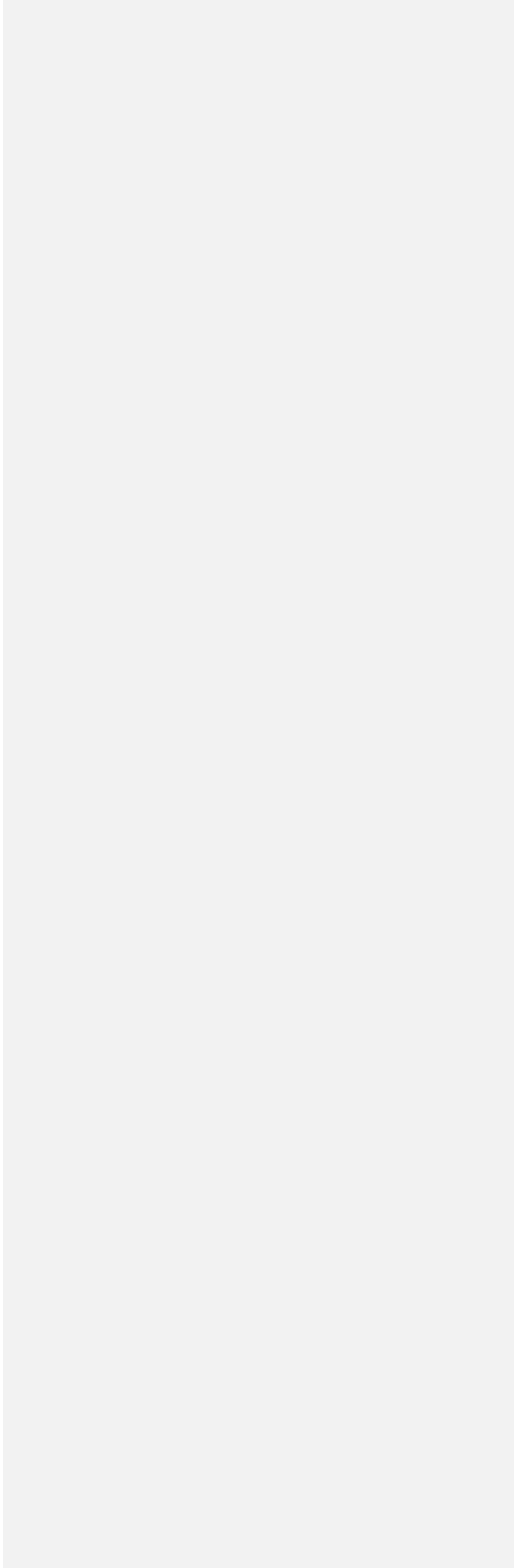
Table 2: Target Specifications of the final product.....10



## List of Acronyms

---

Acronym	Definition
VR	Virtual Reality
OVR	Oculus Virtual Reality
GUI	Graphic User Interface



# **1 Introduction**

The purpose of this project was to provide budding engineers with the experience of being in a team, learn more about team dynamics, and team management. While learning new software such as 'Unity' the goal was to provide the Ottawa hospital patients with a VR platform through which they could experience a version of their treatment before they went through the process physically. This report presents with a step by step process of how the project progressed and what things were learnt through that course.

## **1.1 Product Development**

It is frequent for the patients at a hospital undergoing treatments to be overwhelmed about the procedures they are about to go for. Therefore, the need arises for future engineers to step up and create a VR platform for patients which provides them with the same experience before they go through the process. The platform can be accessed with very minimal restrictions such as the Ottawa hospital being equipped with a Steam VR and the patient being able to monitor their way through. The solution provides the user with a very interactive experience on Unity while being user-friendly, cost effective, and making sure that it calms the patient instead of overwhelming them.

### **1.1.1 Unity**

The platform was created using an interactive software called Unity. This interactive software provides the user to create wide variety of experiences for their clients.

## 2 Need Identification and Product Specification Process

The need arises for a VR platform for the Ottawa Hospital where patients can explore treatment solutions prior to receiving treatment on the Steam VR. The process included constructing a problem statement (as stated above), talking amongst the group and formulating ideas and ways to solve the problem, functional and non-functional requirements, benchmarking, setting up metrics, and focusing on creating target specifications which set up the basis of the whole design process, prototyping, testing, and iterating process.

The process began when the group got together and jotted down a few ideas which could result in a possible solution to this problem. Few of them include,

- Hospital VR asset creating a more realistic experience.
- Trees environment with floating buttons to provide a more calming approach.
- A system which acts like a tv in the patients' living room.

After debating about the best of these three ideas, the group decided that the project should be more calming and interactive and give the user or the patient with a more calming approach of the trees unlike others focusing on making it more of realistic hospital scene.

Functional requirements played an important role in devising a plan for the project. It was decided that the program will use video footage taken with two 180-degree video cameras and a VR headset to create a virtual environment that would simulate cancer treatment and diagnosis procedures. The virtual reality program would be compatible with the Oculus Quest and the Oculus Rift virtual reality headset which was later changed to Steam VR, as this is the type of equipment



the client has. The program would have an interface that allows users to choose between different virtual reality scenarios provided, as well as contain an option to have all scenarios play end to end, so the hospital can have several options to suit its diverse clientele. The program would allow users to pause and play the video as well as rewind and move the video forward. The program would be bilingual (in English and French) and the hospital should be able to add subtitles and audio prompts for user accessibility. Lastly, the program would have a disclaimer saying that the video contains static and that while patients may look around in the simulation, they must remain still during actual procedures to give patients a better idea of what to expect during treatment.

On the other hand, the non-functional requirements included ideas for future component such as the program would be designed to read in text inputted by the user, features such as subtitles for hearing impaired patients and values of the patient's orientation would be added by the hospital if required. In order to optimize the experience as well as prevent disorientation of the patient, patients will be in the required position during the VR session. The program will be designed with a navigation menu where an option for a child friendly version of the recording could be put in to make the program able to accommodate younger users. Additionally, options for making the program available on other platforms and an option to use stereo audio would also be made available. Lastly, options for storing files on the cloud as opposed to the headset hard drive would be explored.

Since this was a first-year class engineering project, the constraints included but were limited to a budget of \$100 or less, individuals of age nine and older should be able to use the program, the design must be able to accommodate video footage taken by two 180-degree videos,

and the size of the hard drive, which would affect the size of the program since there would need to be large allocated memory just for the input videos.

After having worked on other components of the design process; the benchmarking, the metrics, and target specification played a very crucial role in solidifying the ideas and our design process,

Company	VR Vision [4]	Stanford Children’s Hospital [3]	VR Health [2]	BC Children's Hospital [1]
Cost	Not specified	Not specified	Not specified	Not specified
Compatible with Oculus Rift	Yes, as well as HTC Vive Focus Plus and Pico Neo	Yes, the project was developed to be used on the Oculus	Yes, the company is in partnership with Oculus	Yes
Compatible with Oculus Quest	Yes, as well as HTC Vive Pro Eye and Pimax 5K Plus	Yes, the project was developed to be used on the Oculus	Yes, the company is in partnership with Oculus	Yes
End to end/selectable options	Yes	Yes	Yes	Not specified
180 Degree Video Display	Yes - as well as high quality video for anti nausea	Yes, there was a complete recreation of the operational experience.	Not specified	360 degree video
Interface to Select Different VR Scenarios	Not specified	Not specified	Not specified	No

*Table 1. Benchmarking for Designs.*

Benchmarking resulted in giving ideas that were very similar to each other and resulted in having the similar numbers (when benchmarked) which made us decide to compile all the ideas into one. Before we move onto the Target Specifications, the final product was made for Steam VR and as seen wasn't included in the Benchmarking and the Target Specifications table (as seen following) because we did not think about it before we went through the process.

The target specifications finally set us on the track which led to a successful project,

Design Specifications	Relation (=, < or >)	Value	Units	Verification Method
Takes input from two 180 degree video cameras	=	Yes	N/A	Test
Has a user interface with selectable options	>	2	Possible user modes	Analysis
Has pause, play, skip and rewind buttons	=	Yes	N/A	Analysis
Can be modified by hospital	=	Yes	N/A	Test
Compatible with Oculus Rift	=	Yes	N/A	Test
Compatible with Oculus Quest	=	Yes	N/A	Test

*Table 2. Target Specifications of the final product.*

All the necessities came in handy and made sure we steered along the write path.

### 3 Conceptual Designs

As a group we came up with many ideas and are mentioned below,

Allison's Ideas:

1) Select language → Select between an end-to-end mode or further select between individual scenes → play clips with pause, play skip and rewind buttons

If there is a lot of movement detected, movement prompt flashes

2) Select language (activates subtitles for all screens) → click to proceed with tour → have all clips play (with standard pause, play, skip and rewind button) and additional button to skip complete clips

3) Select language (activates subtitles for all screens) → Warning screen informing patients that it is ok to move around in simulation but not in tour, click to proceed with tour → have all clips play (with standard pause, play, skip and rewind button) and additional button to skip complete clips

Kerollos's Ideas:

1) A default controller free mode, that plays everything with both French and English subtitles that play end to end. As well as a controller on mode that allows you to pick and choose

2) Nature scene select language → Pick and Watch or Pre-selected. Will pause if user stands up. Uses controller to pause, play, rewind, skip.

3) Hospital waiting room → Language selection → Controller off → Preselected video  
Stops when patient sits up → Controller on → Preselected video → Pick and watch  
Uses controller to pause, play, skip & rewind

Adi's Ideas:

- 1) A movie scene, select language, pick and watch or pre-selected, lasts until their operation has been completed.
- 2) Something like a tv focused specifically based around the surgery/operation they are going through.
- 3) A research-based video on the procedure there are going through.

Luke's Ideas:

- 1) Language is selected before VR experience starts. Video begins when patient sits down and ends when patient sits up
- 2) First asks for language then asks for automatic or controlled. If the patient selects controlled they will be able to pause, rewind, and fast word the video at will. If they chose automatic the video will begin when they sit down and end when they get up.
- 3) Have patient select between English and French. Then have patient chose which scene they want. The patient can then choose if they want to have control of the video or if they want it to run automatically.

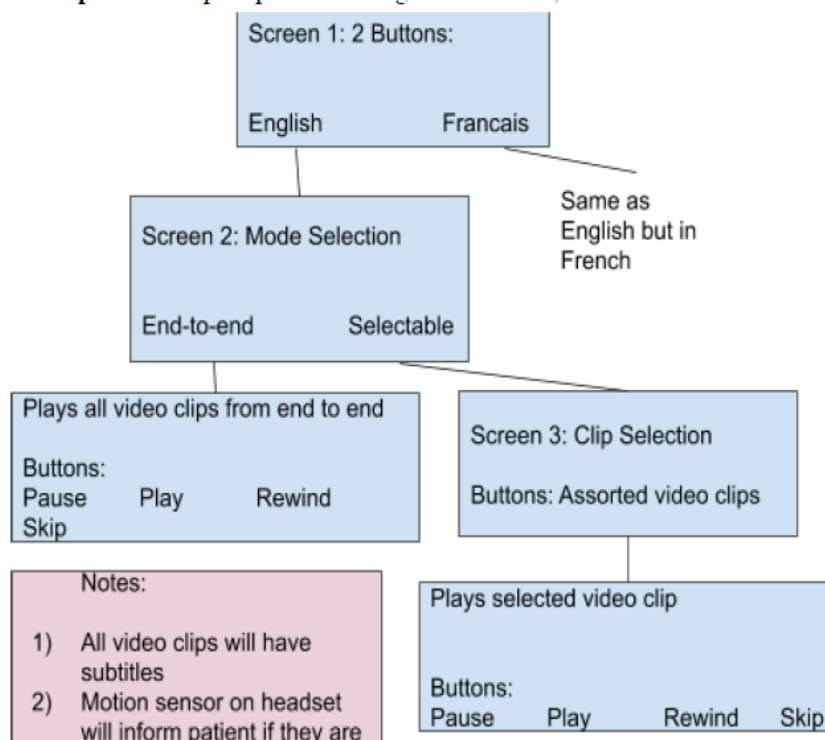
Andrea's Ideas:

- 1) Patient observes a peaceful scene on their screen while supervisor makes selections on an external screen. The external screen first asks for a language choice, then, for a video to be selected and finally, for a mode (controllers off or on) to be selected. If controllers are on, patient has a controller to play, pause and rewind video. If not, video plays end-to-end.
- 2) Patient has controller and navigates through menu independently. First screen displays the hospital waiting room and asks for language choice. Patient can then choose to watch

the whole pre-selected video at once or can choose clips from the video to watch. In each case, patient has controller to play, pause and rewind.

3) Patient has controller. Screen 1 asks for language choice (peaceful or waiting room background decided by supervisor depending on patient). Screen 2 displays instructions and disclaimers about video to follow (mentions motion sensors and that video will pause automatically if patient stands up). When patient clicks 'continue' button, preselected video plays. Patient has controller to play, pause and rewind.

**Concept 1:** Concept explained through a flow chart,



**Concept 2:** Concept explained through a flow chart,

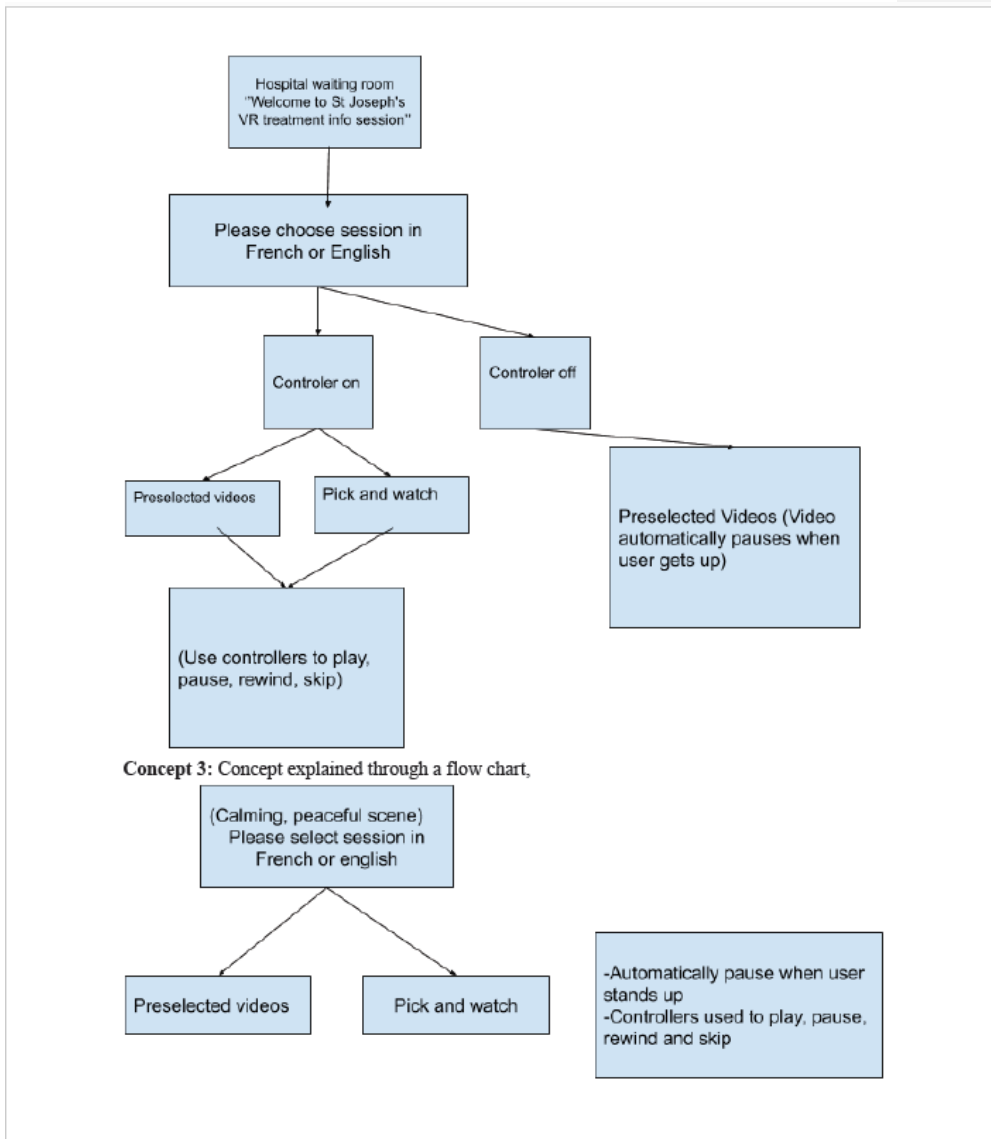


Figure 1. Concepts for the Designs.

## 4 Project Plan, Execution, Tracking & Bill of Materials

We intended on using no money and didn't ended up buying any assets at all.

Talking about the project plan and testing procedures, we devised a plan wherein every person would do a task and the other person would take over but eventually things weren't going well so we had to change it all up and Adi (the team lead) led everything by delegating everything, working on deliverables, Trello, Gantt chart, and assisting with prototype III & its testing. Alison and Andrea were responsible for making sure the videos run perfectly fine and can be prompted by the buttons. Kerollos who was also very much involved in the process created all the basic buttons and assisted in the testing process while his aid Luke made sure the buttons were good to go.



## 5 Prototyping, Testing and Customer Validation.

The actual prototyping, testing, and iterating process began as quickly as possible. For prototype I the group focused on devising a test plan while developing the prototype. The prototype itself was very small and in its preliminary steps. The prototype included a nature screen with buttons put on which helped the patients choose if they wish to start/stop the video. The test itself was going to be a representation of how usable and valuable our product was to the client/end user. As a group, the general objective was learning and adapting. Going into the details of the prototype the graphics included a simple nature scene as seen below, three text objects, and an oculus main camera from the OVR suite. The software identified pick ray of unity controller using the OVR suite and was used to identify text GUIs in the frame as seen below. For example, if one of the text objects was selected by the pick ray, it changed the text prompts from the chosen language to choose the video type. Since this was our very first prototype, we were trying to get used to the fact that if a basic prototype is functional with the patients or not, and if this being the basis of our final project would work comfortably for all the patients. During the testing procedures it was found that that OVR suite wouldn't be able to be tested at the school and everything had to be switched at the very last second.

Commented [AM1]:

```

C:\Users\User1\Documents\TOHProject\Assets\FrenchScript.cs
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class FrenchScript : MonoBehaviour
6 {
7     // Start is called before the first frame update
8     void Start()
9     {
10
11     }
12
13     // Update is called once per frame
14     void Update()
15     {
16         leftData.worldSpaceRay = new Ray (rayTransform.position, rayTransform.forward);
17         //leftData.worldSpaceRay is the pick ray of the controller, I don't know how to add that
18         if (controllerIsPresent && VRRaycaster.Instance != null) {
19
20             GameObject label1 = GameObject.Find("PleaseSelectLanguage");
21             GameObject label2 = GameObject.Find("English");
22             GameObject label3 = GameObject.Find("French");
23
24             text1 = label1.GetComponent<GUIText>();
25             text2 = label2.GetComponent<GUIText>();
26             text3 = label3.GetComponent<GUIText>();
27             //Please select language --> Start right away, pick and watch
28             text1.text = "Commencer Maintenant";
29             text2.text = "Choisir";
30             text3.text = "";
31             //Start right away is selected --> roll video
32             //pick and watch is selected -->pan over clips

```

Figure 2. The code involved in Prototype I.

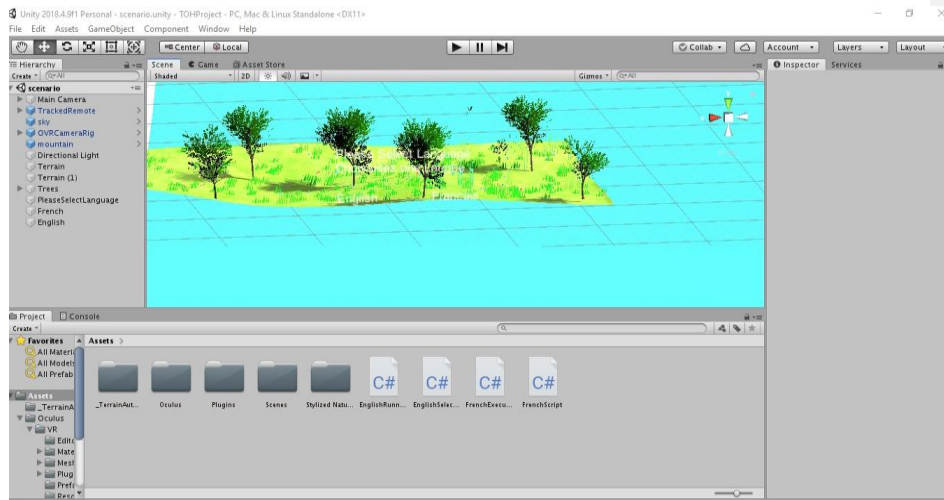


Figure 3. Result on Unity.

Since we could not test Oculus Rift the first prototype had to be completely scrapped and changed for the Steam VR. Prototype I had a lot of malfunctions and therefore we decided to scrap the oculus software and go ahead with this prototype which had a basic code and could work with every equipment available at the hospital. The prototype II included a stereoscopic screen with buttons put on which helps the patients choose if they wish to start/stop the video (as seen below). Since this was our second prototype now, we were getting used to the fact that if the prototype is functional with the patients or not. For the testing process, the file for the prototype, including the codes, buttons, and stereoscopic videos were uploaded to unity (also shown below). The work was divided where two people in the group worked on buttons and two worked on stereoscopic videos, so we made sure that they all bonded together, and we got the base setup for prototype II. A steam VR was setup in order to run the machine. And then in the end the whole function was set up the way it would be built for an actual patient.

```

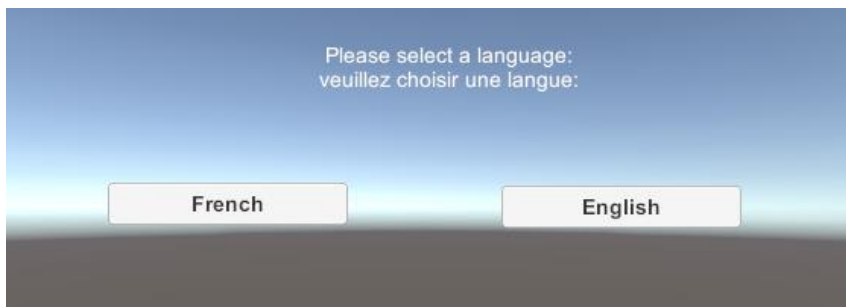
1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using UnityEngine;
5  using UnityEngine.EventSystems;
6  using Valve.VR;
7  public class VRinput : BaseInputModule
8  {
9
10     public Camera m_Camera;
11     public SteamVR_Input_Sources m_TargetSource;
12     public SteamVR_Action_Boolean m_ClickAction;
13
14     private GameObject m_CurrentObject = null;
15     private PointerEventData m_Data = null;
16     private List<RaycastResult> m_RaycastResultCache;
17     private GameObject pointerUpHandler;
18
19     protected override void Awake()
20     {
21         base.Awake();
22
23         m_Data = new PointerEventData(eventSystem);
24     }
25
26     public override void Process()
27     {
28         m_Data.Reset();
29         m_Data.position = new Vector2(m_Camera.pixelWidth / 2, m_Camera.pixelHeight / 2);
30
31         eventSystem.RaycastAll(m_Data, m_RaycastResultCache);
32         m_Data.pointerCurrentRaycast = FindFirstRaycast(m_RaycastResultCache);
33         m_CurrentObject = m_Data.pointerCurrentRaycast.gameObject;
34
35         m_RaycastResultCache.Clear();
36
37         HandlePointerExitAndEnter(m_Data, m_CurrentObject);
38
39         if (m_ClickAction.GetStateDown(m_TargetSource))
40         {
41             ProcessPress(m_Data);
42         }
43         if (m_ClickAction.GetStateUp(m_TargetSource))
44         {
45             ProcessRelease(m_Data);
46         }
47     }

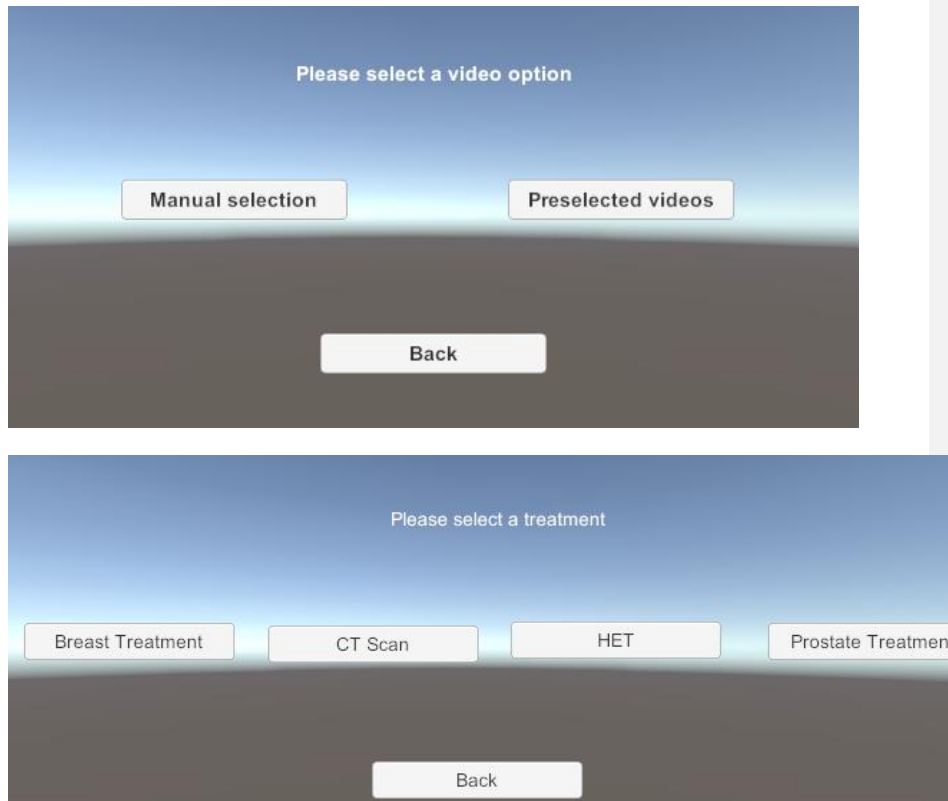
```

Figure 4. The code involved in Prototype II for buttons.



*Figure 5. Result for one of the Stereoscopic videos for Prototype II.*





*Figure 6. Results for Buttons for Prototype II.*

Moving onto the final prototype and pretty much the final product with a few bug fixes. This prototype was the final prototype which included everything at a functional level. We stuck to the idea of using Steam VR and turned out perfectly fine. Now, the prototype included a nature screen with buttons put on which helps the patients choose if they wish to start/stop the video which was already in the second prototype. We also had functional buttons showing manual and pre-recorded videos, moving forward with that we had substitution of buttons ready for the pre-recorded videos. We incorporated another set of code which prompted the videos to run on unity.

Since this is was last prototype, we were wrapping up on a project and gained a lot of experience working on unity.



Figure 7. Initial Screen.

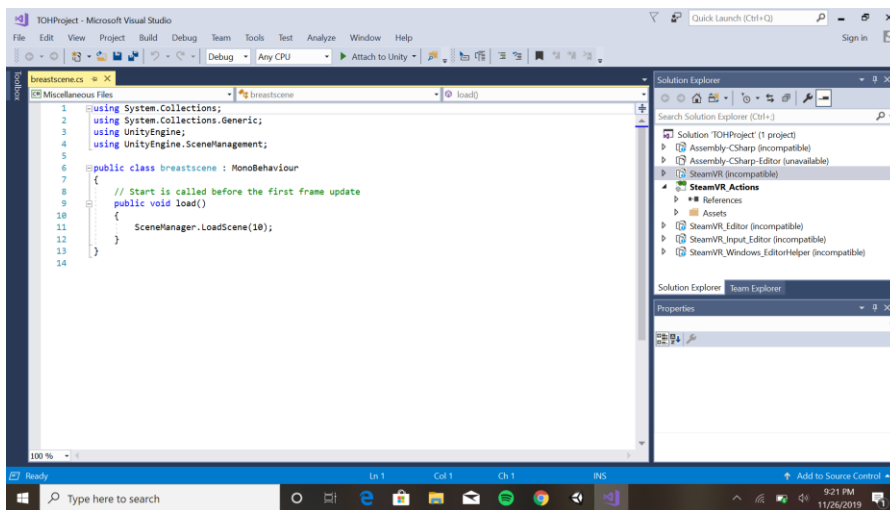


Figure 8. Code which prompts the videos to run on unity.

For testing procedures, the same testing criteria as seen in the second prototype was followed and stopping criteria (shown below) was followed carefully,

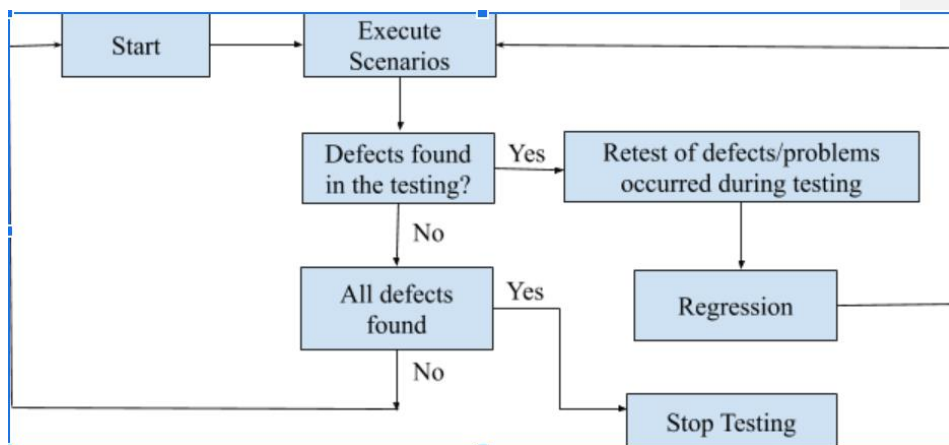


Figure 9. Stopping Criteria used for Testing.

In the end all the prototypes and testing came together and provided us with the final product. When the product was showed to the client he really loved it and made sure we followed on the client feedback which was also very positive.



## 6 Final Solution

The final solution included a VR platform with an opening screen of an appealing scenery full of trees. The buttons on the running game are put on there by the code below. When the button is clicked the first actual screen shows up with the buttons reading French or English. Moving on when the buttons are prompted, the buttons reading manual selection, pre-selected video, or back, and manual selection is pressed we finally reach the for different treatment solutions. This is where things get interesting and interactive. Using another set of code, the desired video is prompted, and it starts running the system. This entire process can be seen below,



*Figure 10. Final Initial Screen.*

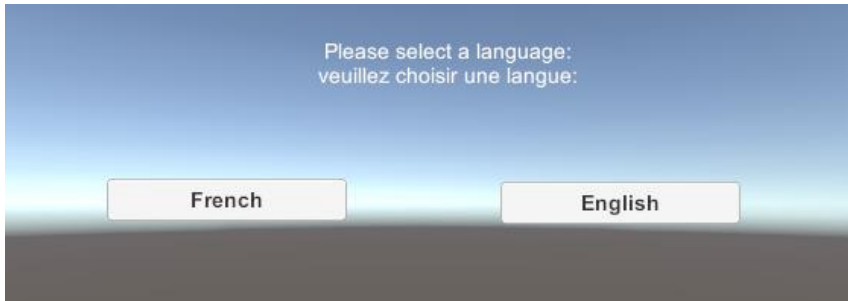


Figure 11. Final first screen.

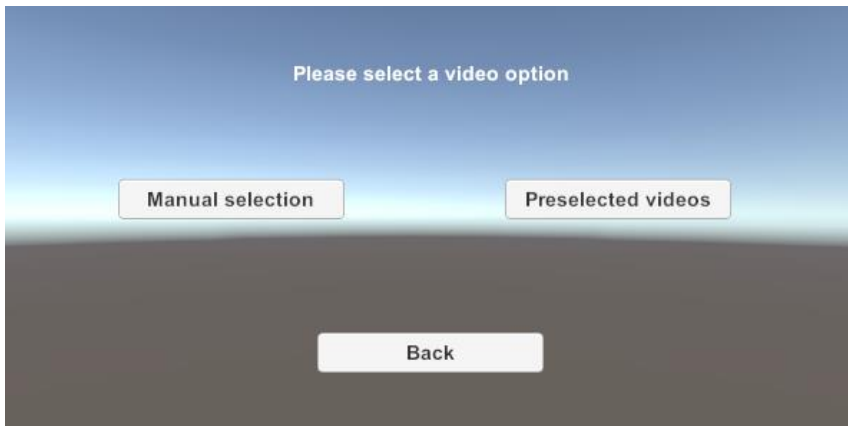


Figure 12. Final second screen.



*Figure 13. Final third screen.*

```

1  using System;
2  using System.Collections;
3  using System.Collections.Generic;
4  using UnityEngine;
5  using UnityEngine.EventSystems;
6  using Valve.VR;
7  public class VRinput : BaseInputModule
8  {
9
10     public Camera m_Camera;
11     public SteamVR_Input_Sources m_TargetSource;
12     public SteamVR_Action_Boolean m_ClickAction;
13
14     private GameObject m_CurrentObject = null;
15     private PointerEventData m_Data = null;
16     private List<RaycastResult> m_RaycastResultCache;
17     private GameObject pointerUpHandler;
18
19     protected override void Awake()
20     {
21         base.Awake();
22
23         m_Data = new PointerEventData(eventSystem);
24     }
25
26     public override void Process()
27     {
28         m_Data.Reset();
29         m_Data.position = new Vector2(m_Camera.pixelWidth / 2, m_Camera.pixelHeight / 2);
30
31         eventSystem.RaycastAll(m_Data, m_RaycastResultCache);
32         m_Data.pointerCurrentRaycast = FindFirstRaycast(m_RaycastResultCache);
33         m_CurrentObject = m_Data.pointerCurrentRaycast.gameObject;
34
35         m_RaycastResultCache.Clear();
36
37         HandlePointerExitAndEnter(m_Data, m_CurrentObject);
38
39         if (m_ClickAction.GetStateDown(m_TargetSource))
40         {
41             ProcessPress(m_Data);
42         }
43         if (m_ClickAction.GetStateUp(m_TargetSource))
44         {
45             ProcessRelease(m_Data);
46         }
47     }

```

Figure 14. Final code for Buttons.

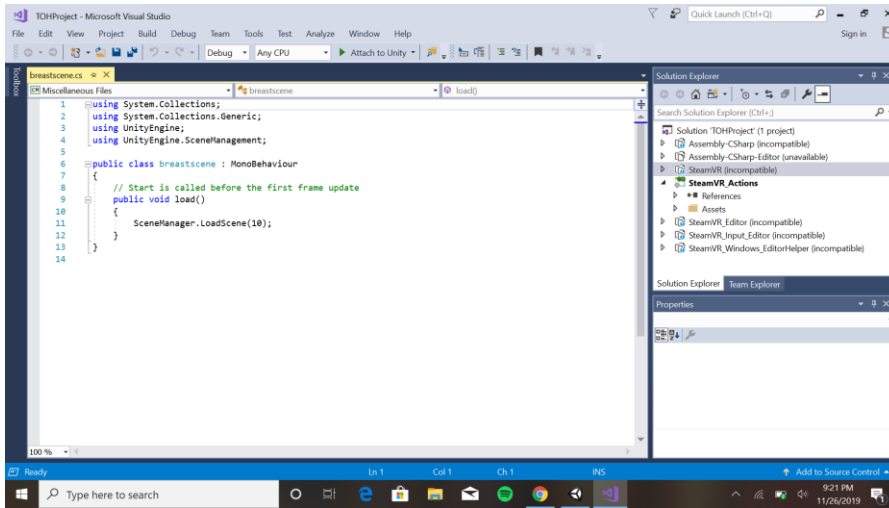


Figure 15. Final code for prompting videos.

## 7 Conclusions and Recommendations for Future Work

In conclusion, although we went with a very simple design it covers all the points and requirements for the project and is very easy to use. Keeping in mind this project is very cost effective since no money was required for any of the assets used, it provides an immersive experience at a cheap rate. It was a very long diligent driven project and was fruitful when we made sure we hit all the metrics and target specifications we had. The need for the VR were raised due to problems of the patients of the Ottawa hospital which eventually were fixed and hopefully provide the patients of the Ottawa hospital with what they were looking for. For future work, we are going to focus making the game more informative, visually appealing, and also make sure to incorporate play, pause, and rewind buttons in the game mode itself.

## 8 Bibliography

1. "How virtual reality in the ER can help calm patients - and make crying kids smile | CBC News," CBCnews, 09-Mar-2019. [Online]. Available: <https://www.cbc.ca/news/canada/british-columbia/b-c-hospitals-using-virtual-reality-to-help-kids-cope-with-painful-procedures-1.5046092>. [Accessed: 12-Dec-2019].
2. J. Kite-Powell, "See How This Company Uses Virtual Reality To Change Patient Healthcare," Forbes, 30-Sep-2018. [Online]. Available: <https://www.forbes.com/sites/jenniferhicks/2018/09/30/see-how-this-company-uses-virtual-reality-to-change-patient-healthcare/#56da378e455e>. [Accessed: 12-Dec-2019].
3. "Project Brave Heart - Stanford Children's Health," Project Brave Heart - Stanford Children's Health. [Online]. Available: <https://www.stanfordchildrens.org/en/innovation/virtual-reality/anxiety-research>. [Accessed: 12-Dec-2019]
4. "Virtual Reality Healthcare Solutions," VR Vision Group. [Online]. Available: [https://vrvisiongroup.com/healthcare-solutions/?gclid=EAIaIQobChMIOPyPybPv5AIVB56fCh0MZQOheAAAYASAAEgIKrvD\\_BwE](https://vrvisiongroup.com/healthcare-solutions/?gclid=EAIaIQobChMIOPyPybPv5AIVB56fCh0MZQOheAAAYASAAEgIKrvD_BwE). [Accessed: 12-Dec-2019].