

GNG 1103

Design Project User and Product Manual

Empathy Tool

Submitted by:

Team Speakeasy

Noe Kapuscinsky, 300239719

Natalia Garcia Hernandez, 300236141

Raissa Rurangwa, 300233899

Nicholas Perras, 0300264902

Graydon Oka, 0300249709 >

<TEAM MEMBER 5, STUDENT NUMBER>

April 13th, 2022

University of Ottawa

Table of Contents

Table of Contents	ii
List of Figures	iv
List of Tables	vi
List of Acronyms and Glossary	vii
1 Introduction	1
2 Overview	2
2.1 Conventions	2
2.2 Cautions & Warnings	4
3 Getting started	5
3.1 Set-up Considerations	6
3.2 User Access Considerations	7
3.3 Accessing the System	7
3.4 System Organization & Navigation	8
3.5 Exiting the System	12
4 Using the System	13
4.1 <Given Function/Feature>	4
4.1.1 <Given Sub-Function/Sub-Feature>	4
5 Troubleshooting & Support	15
5.1 Error Messages or Behaviors	15
5.2 Special Considerations	15
5.3 Maintenance	15
5.4 Support	15

6	Product Documentation	17
6.1	<Subsystem 1 of prototype>	18
6.1.1	BOM (Bill of Materials)	18
6.1.2	Equipment list	6
6.1.3	Instructions.....	21
6.2	Testing & Validation.....	41
7	Conclusions and Recommendations for Future Work	43
8	Bibliography	44
	APPENDICES	46
	APPENDIX I: Design Files	46
	APPENDIX II: Other Appendices	11

List of Figures

Figure 1: Interaction Scene of Prototype	3
Figure 2: NPC	3
Figure 3: Storyline of the Final Prototype	4
Figure 4: Additional Start Function to Enable VR	5
Figure 5: VR Device Detection for VR Enabled Inputs	6
Figure 6: Main Menu	8
Figure 7: About Page	8
Figure 8: User POV When Entering the Experience	9
Figure 9: Classroom Environment	9
Figure 10: Hallway Environment	10
Figure 11: Outdoor Environment	10
Figure 12: Interaction Scene of Prototype	11
Figure 13: Storyline of the Final Prototype	12
Figure 14: Unity Creation Set Up	22
Figure 15: Adding an Image	22
Figure 16: Creating an Empty	23
Figure 17: Image Background	24
Figure 18: Creating a Button	25
Figure 19: Creating a Script	26
Figure 20: GameMenu Script	27
Figure 21: Setting up The Buttons	28
Figure 22: Button Aesthetics	29

Figure 23: About Page Script.....	30
Figure 24: Inserting a Terrain	31
Figure 25: Modifying Terrain	31
Figure 26: First Person Controller	32
Figure 27: Inserting Low Poly Nature Props	33
Figure 28: Inserting the Classroom.....	34
Figure 29: Toru Prefabs	36
Figure 30: NPC Script.....	37
Figure 31: Player Script	37
Figure 32: DialogueManager Script Part 1	38
Figure 33: DialogueManager Script Part 2	38
Figure 34: Creating an NPC Archive.....	39
Figure 35: Creating a Dialogue.....	40
Figure 36: Script Settings.....	41

List of Tables

Table 1. Acronyms.....	vii
Table 2. Glossary	vii
Table 3. Referenced Documents	46

List of Acronyms and Glossary

Table 1. Acronyms

Acronym	Definition
BOM	Bill of materials
EDI	Equality, Diversity and Inclusion
NPC	Non-Player Character
POV	Point of View
VR	Virtual Reality

1 Introduction

This User and Product Manual (UPM) collects the information necessary for users to effectively use virtual reality and for prototype documentation.

2 Overview

The world has always been faced with issues surrounding discrimination for a variety of reasons. Equity consists of people having access to equal opportunities and benefits, by removing biases and prejudices. Diversity is the inclusion of people from a range of different social, genders, ethnic backgrounds, and sexual orientations. Inclusion ensures an environment where everyone feels equally welcomed and valued. To achieve a more EDI lifestyle and culture, it is necessary to first understand the adversities experienced by underrepresented groups and people affected by biased and systemic barriers – such as, women, racialized minorities, Indigenous Peoples, people with disabilities and members of LGBTQIA+ communities–. Systemic barriers are practices that unfairly discriminate belittled groups and prevent them from receiving equal access to resources and opportunities. These barriers are mostly described as unwitting to those who set them because it has been considered normal for a long time, but to those that are affected, they tend to experience lasting negative impacts on their careers, their mental and/or physical health. Furthermore, when choosing an appropriate approach to mitigate discrimination and systemic barriers, to be aware about the increase in discrimination towards people belonging to multiple marginalised groups, also known as intersectionality.

Our virtual reality tool aims to help users to develop a further understanding and awareness of the importance of EDI. In other words, the virtual reality tool intends to teach users empathy towards minority and marginalized groups to address discrimination issues and avoid future ones. The tool needs to showcase accurate experiences, convey a strong and powerful message and be user interactive. The previous is required to create an experience as immersive as possible for the message to get across and to ignite feelings within the user. Assuming the first part is achieved, it is important for the tool to portray a message that is based on real events to showcase and create empathy towards accurate experiences.

Our tool highlights the topic of linguistic discrimination. This is a type of discrimination that is often overlooked and lacks representation in EDI training tools, despite Canada being an official bilingual country as well as uOttawa being a bilingual university –relevant since the students of uOttawa are our main audience–; furthermore, Canada welcomes over hundreds of thousands of immigrants each year which may be subject of this type of discrimination, therefore, it is important to develop a safe and welcoming environment for them through EDI training –our immersive tool–. Additionally, our immersive tool is made up of short scenes to keep the audience engage and for the user to easily to follow along. Each scene was created through some of the team members experiences, family members, surveys and a compilation of TikTok videos that are relevant to the topic. In addition, our final story prototype has gone through over 25 user tests that have resulted in very positive outcomes. From all the testing we have done on the various prototypes of the story we can confidently say we have a story that has high levels of accuracy, relatability and interactivity, and most importantly that conveys a strong and powerful message.

Final prototype:

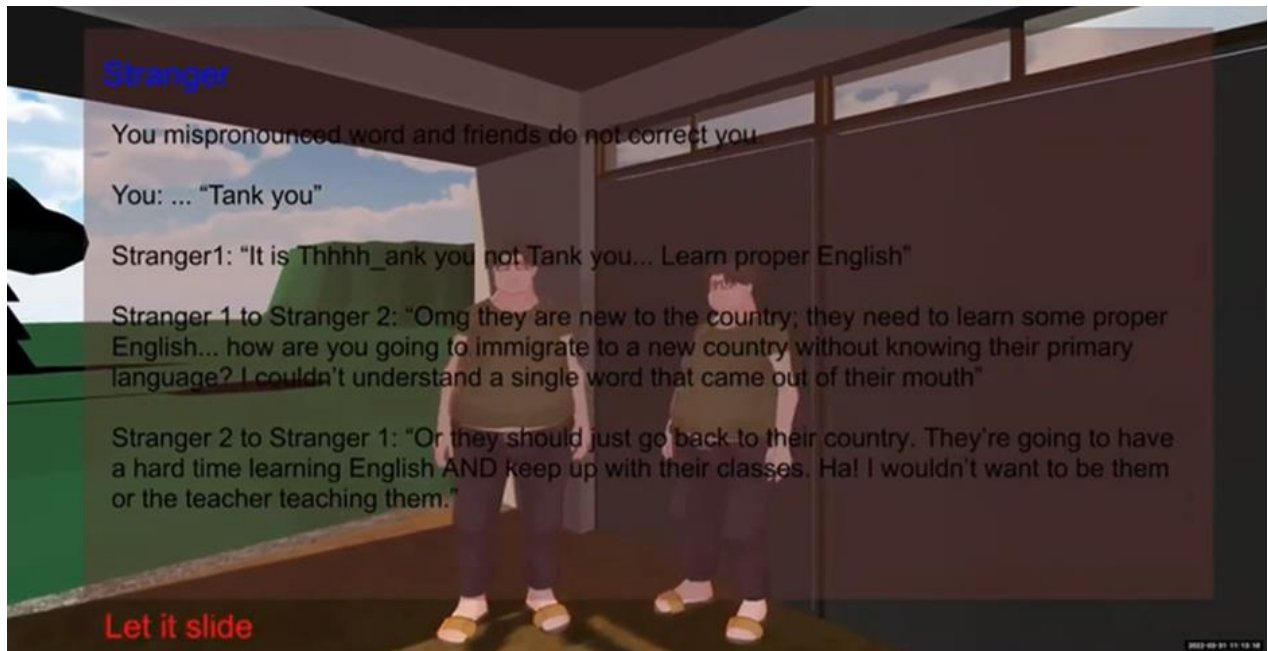


Figure 1: Interaction Scene of Prototype



Figure 2: NPC

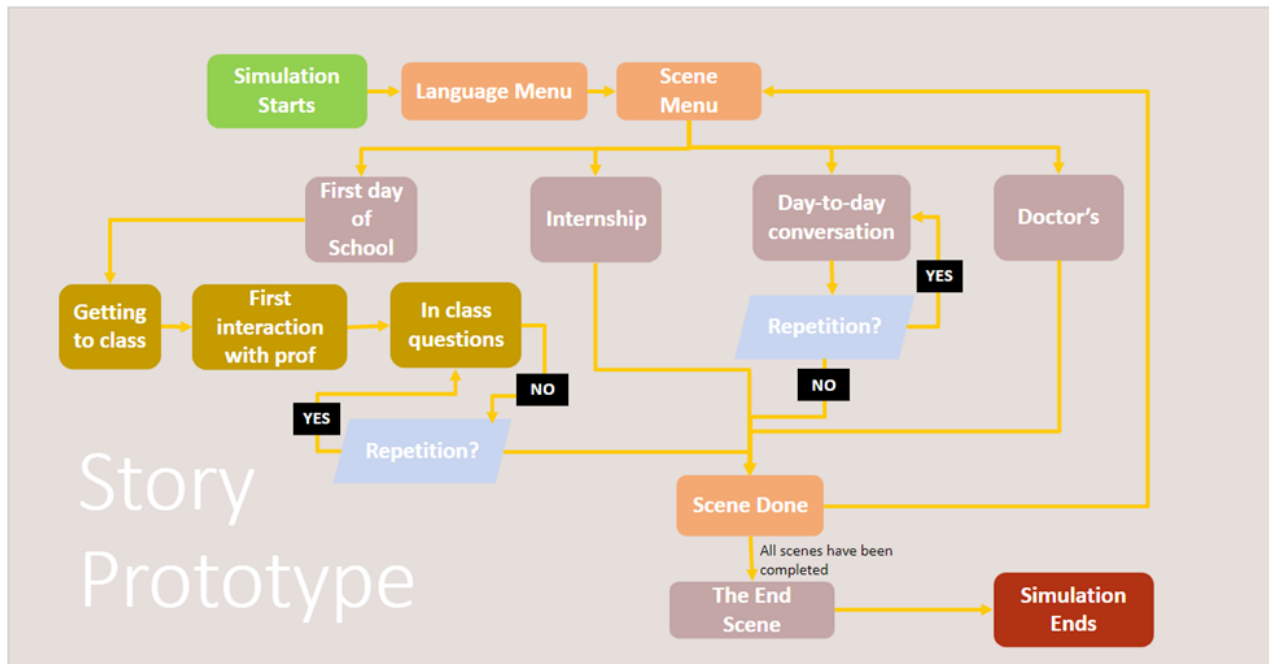


Figure 3: Storyline of the Final Prototype

The final prototype consists of a computer desktop version of a VR experience with potential to be converted into a VR tool; the previous was developed through Unity. As of right now, the experience consists of an explanatory menu of importance of EDI training, as well as a walkthrough of the experience. Once inside, the user can access the different story scenes by coming into range with the different NPCs throughout the environment.

2.1 Cautions & Warnings

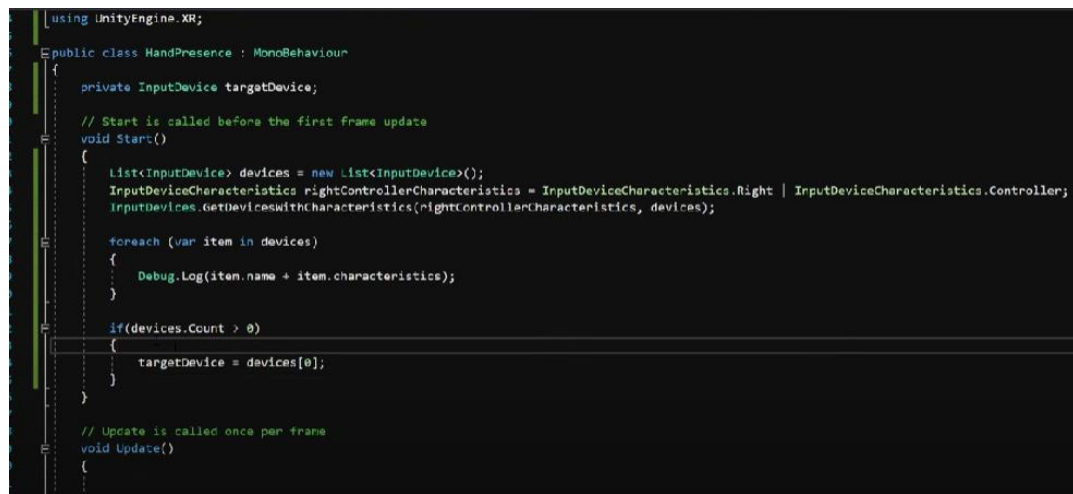
This immersive experience uses a VR headset which can cause motion sickness which may not appear for a few hours after the use of the headset. Furthermore, the scenarios depicted in this product showcase the struggles faced by people that are unfamiliar with the language of the area they reside in or people with heavy accents. This content can be hard to experience and therefore we warn users that their perceptions maybe altered by this experience.

3 Getting started

The final prototype developed by this group was a computer desk top version as the group was never able to access the VR equipment. However, since unity was used to develop this environment, it can easily be edited to function as a VR tool. Therefore, this initiation will be explained for a laptop but the version showcased in this document can easily be edited to VR using this video <https://www.youtube.com/watch?v=VdT0zMeggTQ> to edit the scripts found in the appendices.

The design of this product was done by making the NPC's "scriptable objects" which simply means that only one script needs to be modified which will then be applied to all the NPC's. The script that would need to be modified is shown in the documentation section and is labelled as "DialogueManager". The steps to make these changes are as follows:

1. Enter the desired script in Unity (Dialogue Manager), the script will need to import unity's XR engine by typing "using UnityEngine.XR;" in the very first line of the script above "using System.Collections;"
2. Input the following lines of code into the "void start()" function without deleting the other code in there. This will add the VR headset devices to the list of devices which can then be used to control the experience.



```
using UnityEngine.XR;

public class HandPresence : MonoBehaviour

{
    private InputDevice targetDevice;

    // Start is called before the first frame update
    void Start()
    {
        List<InputDevice> devices = new List<InputDevice>();
        InputDeviceCharacteristics rightControllerCharacteristics = InputDeviceCharacteristics.Right | InputDeviceCharacteristics.Controller;
        InputDevices.GetDevicesWithCharacteristics(rightControllerCharacteristics, devices);

        foreach (var item in devices)
        {
            Debug.Log(item.name + item.characteristics);
        }

        if(devices.Count > 0)
        {
            targetDevice = devices[0];
        }
    }

    // Update is called once per frame
    void Update()
    {
    }
}
```

Figure 4: Additional Start Function to Enable VR

3. Next, read through to find what keys are currently implemented in this script. When doing this, you should find the following keys used in the code to make the game function as "KeyCode.Button".
 - a. KeyCode.Escape
 - b. KeyCode.E
 - c. KeyCode.Return
 - d. Special case: GetAxis("Mouse ScrollWheel")

4. Once the keys have been Identified, you can make the following changes to make the environment VR compatible.
 - a. KeyCode.Escape →
targetDevice.TryGetFeatureValue(CommonUsages.secondaryButton, out bool secondaryButtonValue;)
 - b. KeyCode.E →
targetDevice.TryGetFeatureValue(CommonUsages.triggerButton, out float triggerValue;)
 - c. KeyCode.Return →
targetDevice.TryGetFeatureValue(CommonUsages.primaryButton, out bool primaryButtonValue;)
 - d. Special case: GetAxis("Mouse ScrollWheel") →
targetDevice.TryGetFeatureValue(CommonUsages.primary2DAxis, out Vector2 primary2DAxisValue;)
5. The next step is to replace the "key words" for the keyboard with those of the VR controllers. For example any if statement containing a keyboard command can be changed as follows.
 - a. `if (Input.GetKeyDown(KeyCode.Escape))` → `if (secondaryButtonValue);`

```
// Update is called once per frame
void Update()
{
    targetDevice.TryGetFeatureValue(CommonUsages.primaryButton, out bool primaryButtonValue);
    if (primaryButtonValue)
        Debug.Log("Pressing Primary Button");

    targetDevice.TryGetFeatureValue(CommonUsages.trigger, out float triggerValue);
    if (triggerValue > 0.1f)
        Debug.Log("Trigger pressed " + triggerValue);

    targetDevice.TryGetFeatureValue(CommonUsages.primary2DAxis, out Vector2 primary2DAxisValue);
    if (primary2DAxisValue != Vector2.zero)
        Debug.Log("Primary Touchpad " + primary2DAxisValue);
}
```

Figure 5: VR Device Detection for VR Enabled Inputs

6. Finally, this will need to be repeated in the Player, GameMenu, and About scripts. This can be done in the same way as this script.

Note: for the game to function properly, the Player script must use the opposite controller from the DialogueManager script. In this case, the right controller was used in the DialogueManager script so the left controller would be used for the player.

3.1 Configuration Considerations

As previously explained, this is a virtual immersive experience which means that a user would require a device that is compatible with the product. The main device to use would be either a laptop,

or a VR headset that can be paired with either a computer or modern cell phone. When using a computer, all the inputs would be done as described above using the computer's keyboard. However, if the user modified the product for VR use, all the inputs would be made using the controllers.

3.2 User Access Considerations

At the start of the development of this EDI tool, our main target audience was students from uOttawa. However, as we continued the design process of the story – the tool aspect that determines the target audience the most – became more open ended to reach a larger audience. We touch on experiences such as first day of school, day-to-day conversations, internship/work interviews and going to the doctor's, all of which are and/or have been experienced by most people. These common experiences work as a steppingstone to create a more relatable environment for the user and therefore provide a more immersive tool.

The main restriction of the system is the lack of visibility access, since the tool showcases experiences only through text. To address this restriction a voice setting/gadget could be added.

3.3 Accessing/setting-up the System

Since this product is a virtual experience, the set up for this product would consist of the following.

1. Acquiring appropriate equipment (computer, phone, VR head set).
2. Entering the virtual environment in the same way you would access any other application on the selected device. (On a computer and/or phone this would consist of just clicking on the application to run it.
3. Once the application is open, the user can follow the following instructions to get started.

Upon first entering the experience, the user will be brought to a main menu. The menu consists of three buttons that would allow the user to do one of three things. The first button would allow the user to enter the experience and play out the scenarios and interactions available in this product. The second button allows the user to access the "About" page which is a page that gives the user an overall description of the product and why it is important. The last button on the start menu is also available on the About page and this is the exit button which allows the user to shut down the experience. Furthermore, the "About" page also has a button that would allow the user to enter the experience or return to the main menu.

Empathy Tool



Figure 6: Main Menu

This is a project created by Natalia Garcia Hernandez, Noe Kapuscinsky, Graydon Oka, Nicholas Perras, and Raissa Rurangwa for the Winter 2022 GNG1103's Design Day.

The purpose of this project was to create a VR experience that would allow its user to experience life in the shoe's of someone else in order to gain a more empathetic understanding of certain lifestyles or challenges.

Language is an extremely important aspect of life in general. It allows us to communicate ideas, express oneself, and build relationships. It is therefore paramount to understand that other people may have trouble communicating, especially if english or french is not their first language. This is also particularly important as inclusivity is a major goal at the University of Ottawa.

We hope you enjoy as much as you are informed.



Figure 7: About Page

3.4 System Organization & Navigation

Once the user has entered the experience, they will be able to navigate the environment using key board arrows. The arrows can be used in the most common way where the “up” arrow moves the player forwards, the “down” arrow moves them back and so on. When modified to VR, the player would be using VR controllers, designed by oculus, with slightly different commands. The user would be navigating the experienced based on what commands are inserted as replacement for the computer commands. The user will also need to use the trackpad or computer mouse to control which direction is “forwards”. In the VR version, this would be done when the users moves their head and “looks around the environment”



Figure 8: User POV When Entering the Experience



Figure 9: Classroom Environment



Figure 10: Hallway Environment

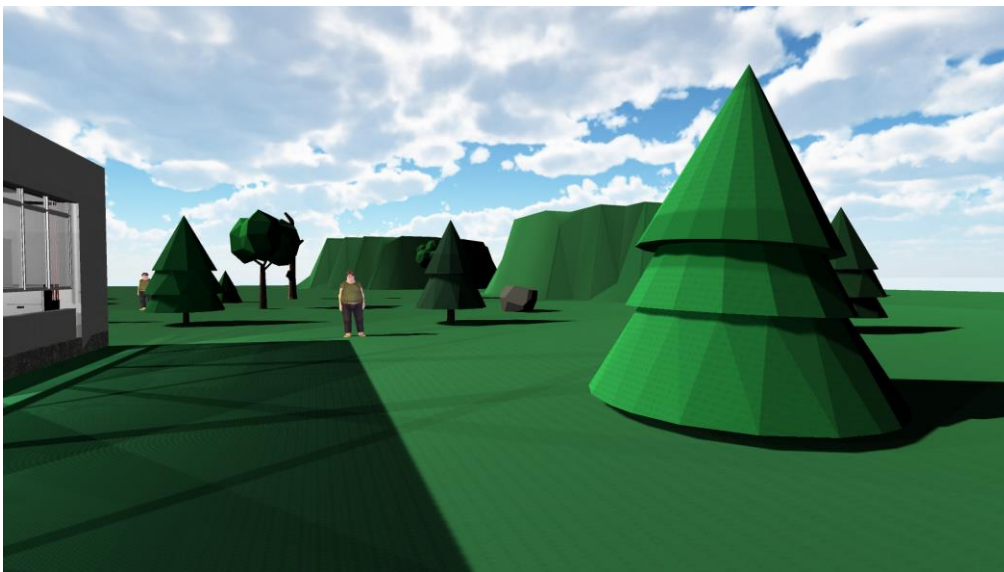


Figure 11: Outdoor Environment

Next, once the user is feeling more comfortable with the navigating the environment, the user will be able to interact with any NPC they choose by clicking the button “E” on their device. In the case of an Oculus VR controller, the user would be selecting a different button based on the updated programming. Once the interaction has started, the user can scroll through their response to a given scenario. They will be able to scroll using the “scroll” function on their mouse. On the other hand, once this is updated for VR, the user will be able to use two buttons to scroll up and down through the responses. To select a response, the user can press the “enter” key which will

show them the outcome of their choice. The user will then be able to scroll through the other possible responses to see how their response affects the overall outcome. This is shown in both figures 1 and 12.

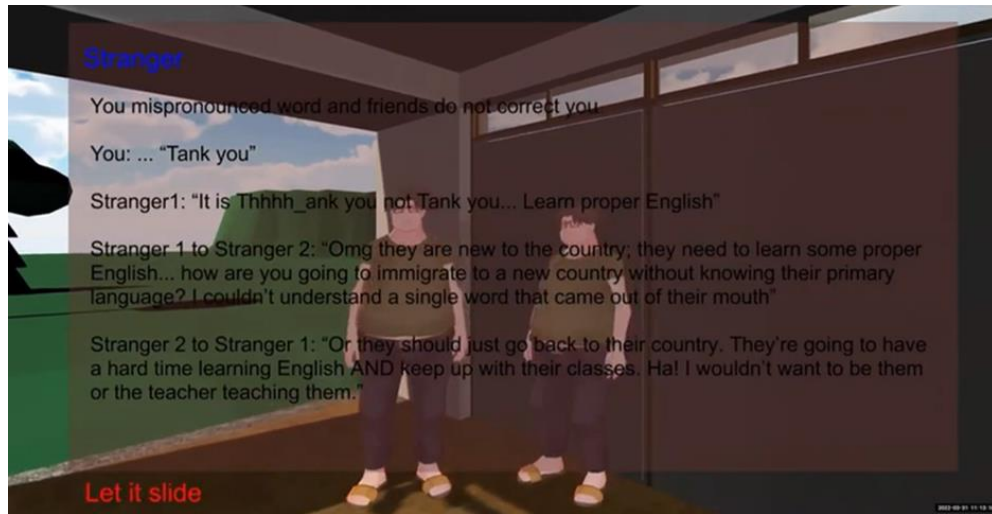


Figure 12: Interaction Scene of Prototype

This image shows an interaction that would take place. The blue writing represents the person the user interacts with. This ranges from strangers, friends, professors and more. Next, the dark text will represent either the scenario or the outcome of the interaction once a response has been selected. Finally, the red writing at the bottom represents the different responses the user can provide to the given situation. Each scenario in this experience and the outcomes of their responses are depicted in the next flow chart. This flow chart is shown both above as figure 3 and below as figure 13.

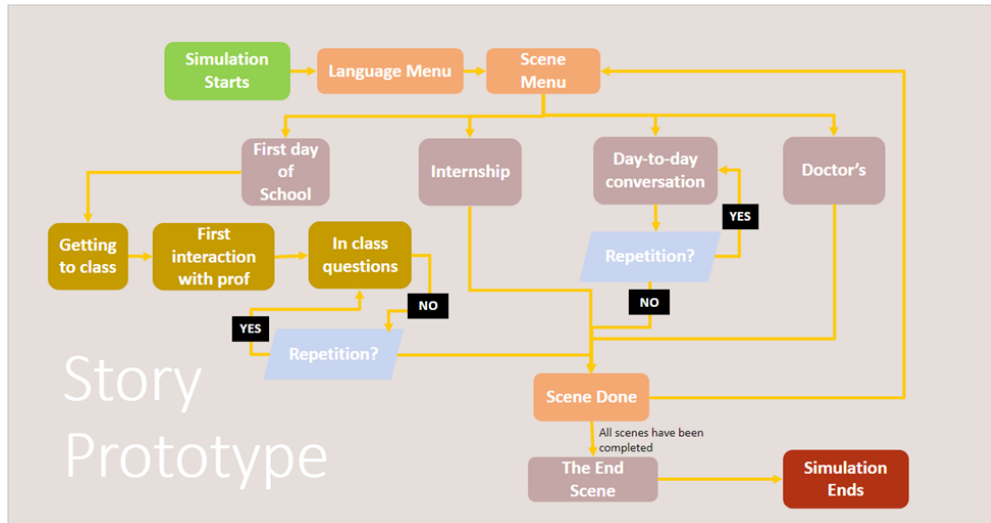


Figure 13: Storyline of the Final Prototype

3.5 Exiting the System

Finally, when the user is ready to exit the game, they can simply press the “escape” key which will shut down the experience. This would end their time in the product, however, they would be able to re-enter in the same way they did the first time.

4 Using the System

In this part of the document, the user can find all commands and inputs required to successfully use the final design of this project. This includes everything from entering the environment, navigating it, interacting with NPC's, exiting the game, and Menu/About page navigation.

4.1 Entering the Environment

This section of the document will provide explicit instructions on entering and navigating the VR environment used in this product.

4.1.1 Game Menu/ About Page

Upon first entering the experience, the user will be brought to a main menu. The menu consists of three buttons that would allow the user to do one of three things. The first button would allow the user to enter the experience and play out the scenarios and interactions available in this product. The second button allows the user to access the “About” page which is a page that gives the user an overall description of the product and why it is important. The last button on the start menu is also available on the About page and this is the exit button which allows the user to shut down the experience. Furthermore, the “About” page also has a button that would allow the user to enter the experience or return to the main menu.

4.1.2 Navigating in the Environment

Once the user has entered the experience, they will be able to navigate the environment using key board arrows. The arrows can be used in the most common way where the “up” arrow moves the player forwards, the “down” arrow moves them back and so on. When modified to VR, the player would be using VR controllers, designed by oculus, with slightly different commands. The user would be navigating the experienced based on what commands are inserted as replacement for the computer commands (this document recommends the use of the left-handed controllers 2D-axis). The user will also need to use the trackpad or computer mouse to control which direction is “forwards”. In the VR version, this would be done when the users move their head and “looks around the environment”.

4.2 Interacting in the Environment

This section will give the user explicit instructions on how to interact with NPC's the VR environment of the product.

4.2.1 Beginning an Interaction

As described above, the user will be able to start interacting with an NPC by pressing the “E” key on a computer or using the trigger of the right-hand controller on a VR set up. Once this has been done, if the user is within range, the NPC will prompt the user with a scenario.

4.2.2 Selecting a Player Response

Once the user has been prompted with a scenario, they will be able to respond in multiple different ways. This can be done by scrolling with the mouse or in the VR case, scrolling with the primary 2D-axis (Joystick) on the right-hand VR controller. Typically, the user should expect at least 2 different ways of responding however, some interactions contain more. The user can continuously scroll through the possible response and view the different outcomes associated with each one. To select a response, the player will use the “enter” key if using the laptop version or the primary button on the right-hand controller for the VR version. Once the user has selected an answer, the outcome of the situation will be depicted, and the user will be able to select a different response if they chose.

4.2.3 Exiting an Interaction

When the user is ready to exit an interaction, they may use the same command to start the interaction to exit. For the case of a computer, the user would simply press on the “E” key and for VR the user would press the trigger button.

4.3 Exiting the Environment

Finally, when the user is ready to exit the game, they can simply press the “escape” key (or for VR configured the way described in this document, the secondary button of the right-hand controller) which will shut down the experience. This would end their time in the product;+z however, they would be able to re-enter in the same way they did the first time.

5 Troubleshooting & Support

This section of the document covers the recovery and error correction procedures to ensure that the Virtual Reality experience is operational. This includes different error messages / behaviors, special considerations, maintenance and support.

5.1 Error Messages or Behaviors

The virtual reality experience we've created does not have many errors that could happen. If there are some issues it will most likely be in the programming of the game or the app itself. There may be issues where the program isn't running properly or doesn't run at all.

5.2 Special Considerations

The software produced is not a complicated piece. However, issues may persist. Since this product was designed with VR in mind, there may be calibration errors with the software and the VR equipment. If the VR controls are unresponsive or do not act appropriately, restart the program. If errors persist after restarting the program, the VR equipment may be at fault and should be reset or recalibrated as per their instructions. Similarly, if none of the interactions appear to be activating, reboot the program and/or your system. If errors occur after an update to Unity, then there may be a compatibility error. Please refer to section 5.3, Maintenance, for more details on how to solve and prevent this problem. Please also check whether your device and/or operating system can run Unity, as there may be a compatibility error there.

5.3 Maintenance

Because this product is entirely virtual, unique maintenance is required. Although not a part of the design, the hardware that the software is running on or uses (computer, VR headset, controllers, keyboards, etc.) should be maintained as per their own instructions, and there is no extra or alternative care from the use of this software. This software uses Unity version 2019.4.35f1 and should be compatible with every version of Unity up to and including the most recent version (as of April 13th, 2022). However, future versions of Unity may cause compatibility issues that are not foreseeable and therefore it is highly recommended to create a backup of a functional version before every update. If a non-functioning update was performed without any backups, attempt to perform a rollback to the previous version of Unity. Since no physical copy of this software exists, it is recommended to create a personal physical copy, to upload the software to a cloud, and/or to save the software to multiple HDDs and/or SSDs. This is because HDDs and SSDs can suffer from memory leaks, which may make the software unusable. By having a physical copy, having it saved on a cloud service, and by having it saved to multiple storage devices, it minimizes possibility of losing the software entirely.

5.4 Support

When using the virtual reality experience, one may encounter different problems that they will need help to fix or have questions for the developers. For example, if the game code has

problems, you can contact Nicholas Perras with his email address: nperr007@uottawa.ca or his phone number: 613-799-1779. If you are having problems with the story and have any questions/feedback, you can contact Raissa Rurangwa or Natalia Garcia Hernandez with their email addresses: rrural04@uottawa.ca, ngarc046@uottawa.ca or their phone numbers: 613-5010-513, +52 983 137 4052. If you have any questions revolving the application of the game please contact Noe Kapuscinsky at his email address: nkapu024@uottawa.ca or his phone number: 587-434-5116. If you have any general questions please contact Graydon Oka at his email address: goka014@uottawa.ca or his phone number: 613-850-5880.

6 Product Documentation

This section will focus on the development of the story itself as well as the unity aspects of the prototype. It will break down the steps that were taken and requirements needed to obtain the final prototype presented in this manual. It will also touch on the considerations, and assumptions made along the way.

6.1 Story aspects of prototype

As mentioned before, the first step was fully understanding linguistic discrimination, we read articles and listened to podcast of experts on the subject to create a deeper understanding. Once we had a grasp of the theory, we moved to recollect real-life experiences to create an accurate storyline the user can create empathy with. The recollection of the previous was the result of firsthand experiences of some team members, interviews to close friends and family members, 15 survey response through Instagram from immigrants that moved to Canada, Canadians with immigrant parents, multilingual People, and Quebecers, and the compilation of 50+ Tik Tok videos of people sharing their stories in relation to Linguistic discrimination - hashtags that were used to find the relevant videos to our study case were: #bilingualproblems, #bilingualstruggles, #mispronouncingwords, #bilinguallife, #accent, #multilingual, etc.-. Through our case study, we were able to distinguish the most prominent linguistic discrimination experiences and highlight them in our storyline.

Through the case study just mentioned, we were able to distinguish the most prominent linguistic discrimination experiences and highlight them in our storyline which was the next step for the story development. With the base of the storyline, we were able to develop scenes and dialogues to accurately portray the different cases of discrimination whether direct or indirect discrimination.

Finally, we used the Tests that are mentioned in section 6.1.1 Testing & Validation to tweak and adapt the different phases of the story to improve it and match our initial goal as much as possible. After we implemented the feedback and needed changes we obtained our final prototype, a story that has high levels of accuracy, relatability and interactivity, and most importantly that conveys a strong and powerful message.

6.1.1 Testing & Validation

Our team started testing the storyline by sharing it in a Facebook group with 326 members of which the majority are the targeted audience to test accuracy and the relatability of the storyline. We asked people to like or comment if they have experienced one or more situations described in the storyline. In addition, we encouraged them to give us extra feedback to adapt or add to the storyline. We considered the comments and thoughts to adapt and/or add to the storyline before developing the scenes and dialogues.

The second and final step of testing was via text through Instagram, Facebook and Snapchat. We sent it to a large variety of people. The test pool included people that have been linguistically discriminated – those who do not speak the Native Language of the area they are in and those who do but have a strong accent -, people that have not been linguistically discriminated. Once they complete, they responded a set of questions in relation to interactivity, relatability and story impact:

- "1. Have you been linguistically discriminated against before? If so, have you experienced one or more situations described?
2. From 1 to 5 how would you rate the level of interaction you had (5 being best)?
3. Did something happen on the storyline that you had never thought could happen in real life? If so, what?
4. Any comments on the plot line?"

6.2 Unity aspects of prototype

6.2.1 BOM

The following image includes all equipment used for the tool, each of their cost and an overall cost –sum of equipment costs- to create the tool.

Budgeting						
Item	Description	Prototype	Quantity	Cost \$	Extended Cost \$	Link
Standard Assets	Unity Asset	1, 2, 3	1	0	0	https://assetstore.unity.com/packages/essential/asset-packs/standard-assets-for-unity-2018-4-32351

HDR Sky	Unity Asset	1, 2, 3	1	0	0	https://assetstore.unity.com/top-assets/top-free
6 Free Textures	Unity Asset	1, 2, 3	1	0	0	https://assetstore.unity.com/packages/2d/textures-materials/nature/6-free-textures-90953
Low Poly Nature Pack	Unity Asset	2, 3	1	0	0	https://assetstore.unity.com/packages/3d/environments/landscapes/low-poly-nature-props-pack-131853#publisher
Classroom	Unity Asset	2, 3	1	9	0.45	https://assetstore.unity.com/packages/3d/environments/urban/classroom-75007
Unity	VR development program	1, 2, 3	1	0	0	https://unity3d.com/get-unity/download
VR Body	Unity Asset	3	1	15	0.75	https://assetstore.unity.com/packages/tools/character/vr-body-79337

Basic NPC	Unity Asset	1, 2	1	0	0	https://assetstore.unity.com/packages/3d/characters/humanoids/npc-character-prototype-series-132051
Toru NPC	Unity Asset	3	1	20	1	https://assetstore.unity.com/packages/3d/characters/humanoids/humans/toru-110909#content
VR headset	VR hardware	1, 2, 3	1	0	0	Provided by the maker lab
VR controllers	VR hardware	1, 2, 3	1	0	0	Provided by the maker lab
Total Cost \$	46.2					

Budgeting	
Item	Link
Basic Motions	https://assetstore.unity.com/packages/3d/animations/basic-motions-free-154271
Joystick pack	https://assetstore.unity.com/packages/tools/input-management/joystick-pack-107631
3D Game Kit	https://assetstore.unity.com/packages/templates/tutorials/3d-game-kit-115747
HDR Sky	https://assetstore.unity.com/top-assets/top-free
6 Free Textures	https://assetstore.unity.com/packages/2d/textures-materials/nature/6-free-textures-90953
Modular First Person Controller	https://assetstore.unity.com/packages/3d/characters/modular-first-person-controller-189884
Everyday Motion Pack	https://assetstore.unity.com/packages/3d/animations/everyday-motion-pack-free-115067
Classroom	https://assetstore.unity.com/packages/3d/environments/urban/classroom-75007
Unity	https://unity3d.com/get-unity/download
VR Body	https://assetstore.unity.com/packages/tools/camera/vr-body-79337
Basic NPC	https://assetstore.unity.com/packages/3d/characters/humanoids/npc-character-prototype-series-132051
Toru NPC	https://assetstore.unity.com/packages/3d/characters/humanoids/humans/toru-110909#content
VR headset	Provided by the maker lab
VR controllers	Provided by the maker lab

The Appendix section contains the link to the excel of the BOM file, from here links for all the things that were used can be accessed.

6.2.2 Instructions

This product was made using unity 2019.4.35f1 (64 bit) and therefore in order to recreate this you will need to download unity from the following link:

<https://unity.com/download>

Once this is downloaded, you will also need to instal the unity assets from the BOM in section 6.2.1. Next, you can open the unity hub and create a new project. Once this is done, the next steps can be followed to recreate the game.

1. Name the first seen “MainMenu”.
2. For organization purposes under assets in the project teab, create a folder called “Scenes” and move the MainMenu Scene into this folder.
3. Next, create 2 more scenes called “About” and “Classroom” then add them to the scenes folder. This should look something like figure 14 shown below without the “interaction” and “SampleScene” scenes.

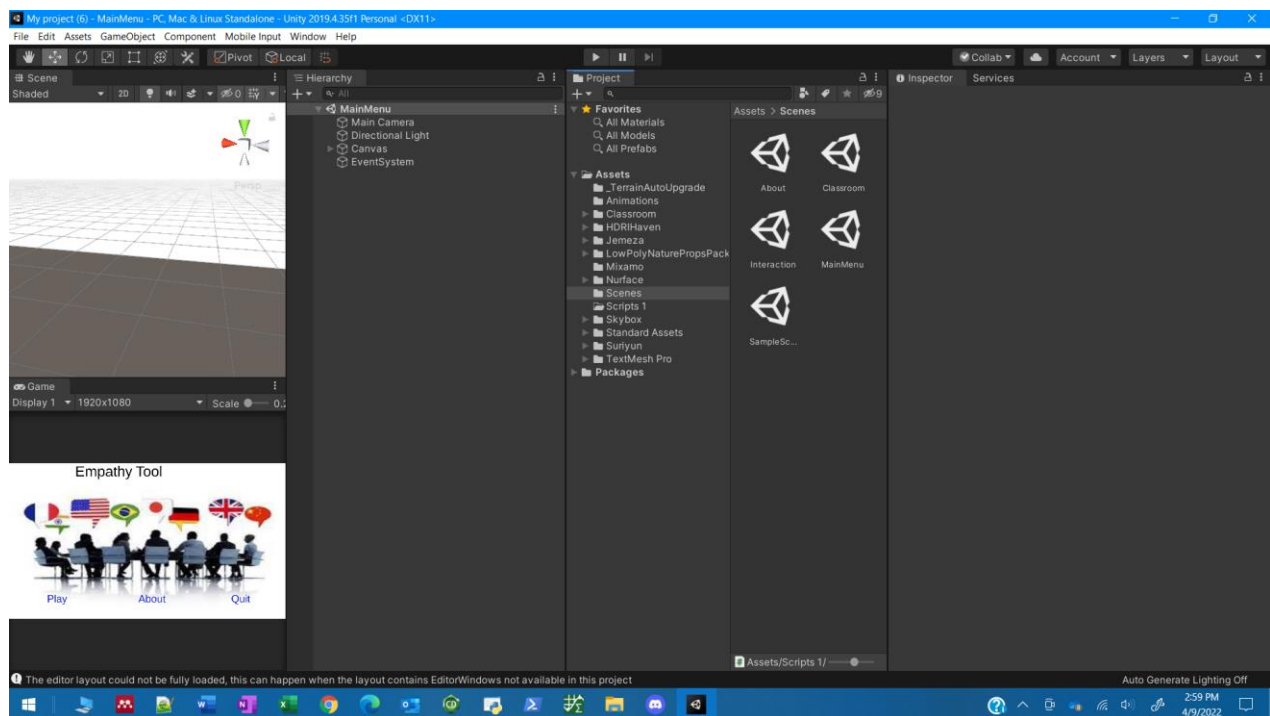


Figure 14: Unity Creation Set Up

Main Menu

1. Open the scene called “MainMenu” and go to the Game object Menu, and select Image under UI as shown in figure 15.

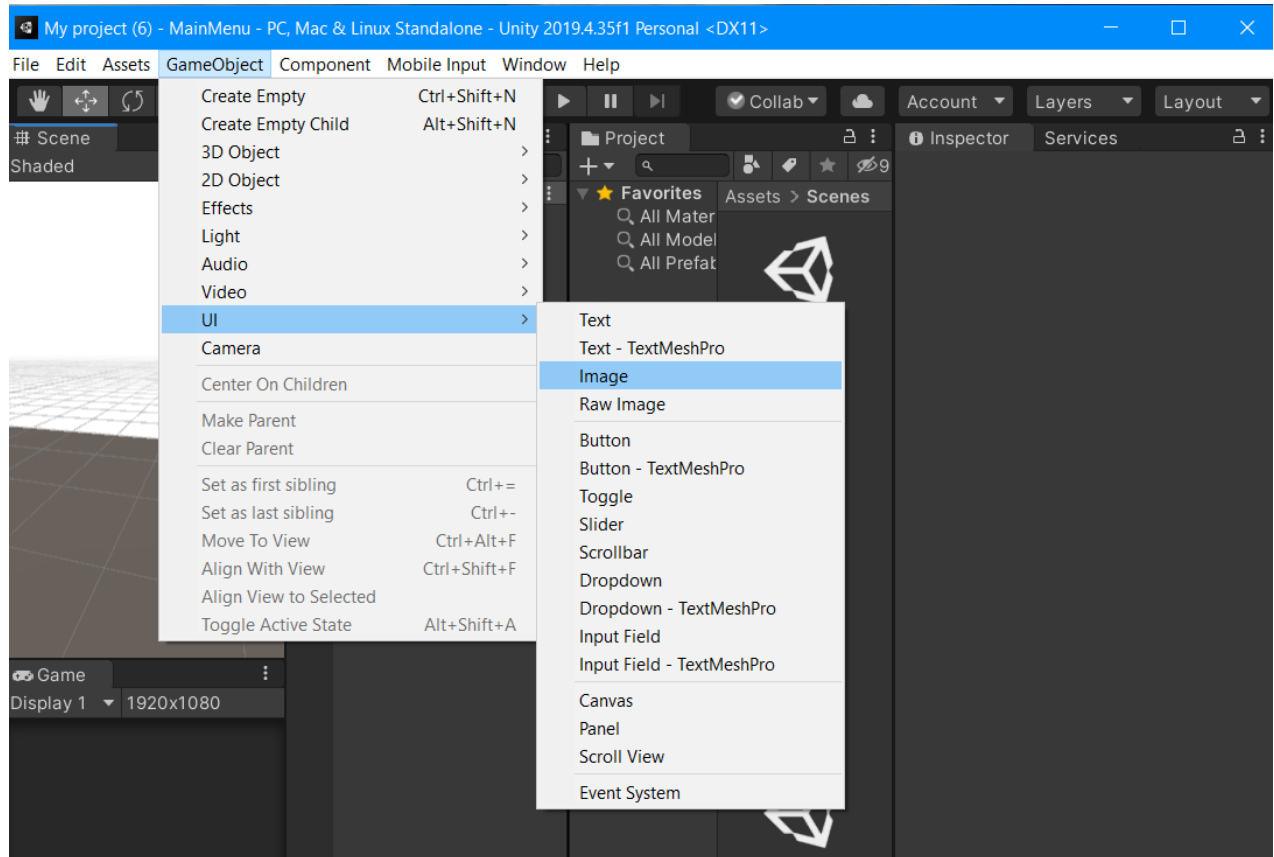


Figure 15: Adding an Image

Once the image has been added, you can fit it to the players view by pressing on the image in the game view and dragging the corners. Next, the users image of choice can be imported in to unity (drag and drop from desk top into the assets folder) and then added as the source image by clicking on the “source image” tab under the “Image” object in the Hierarchy window as shown by figure 16.

2. Next, the user can right click on the canvas in the Hierarchy window and press on create empty and name the empty object “GameMenu” as shown in figure 16.

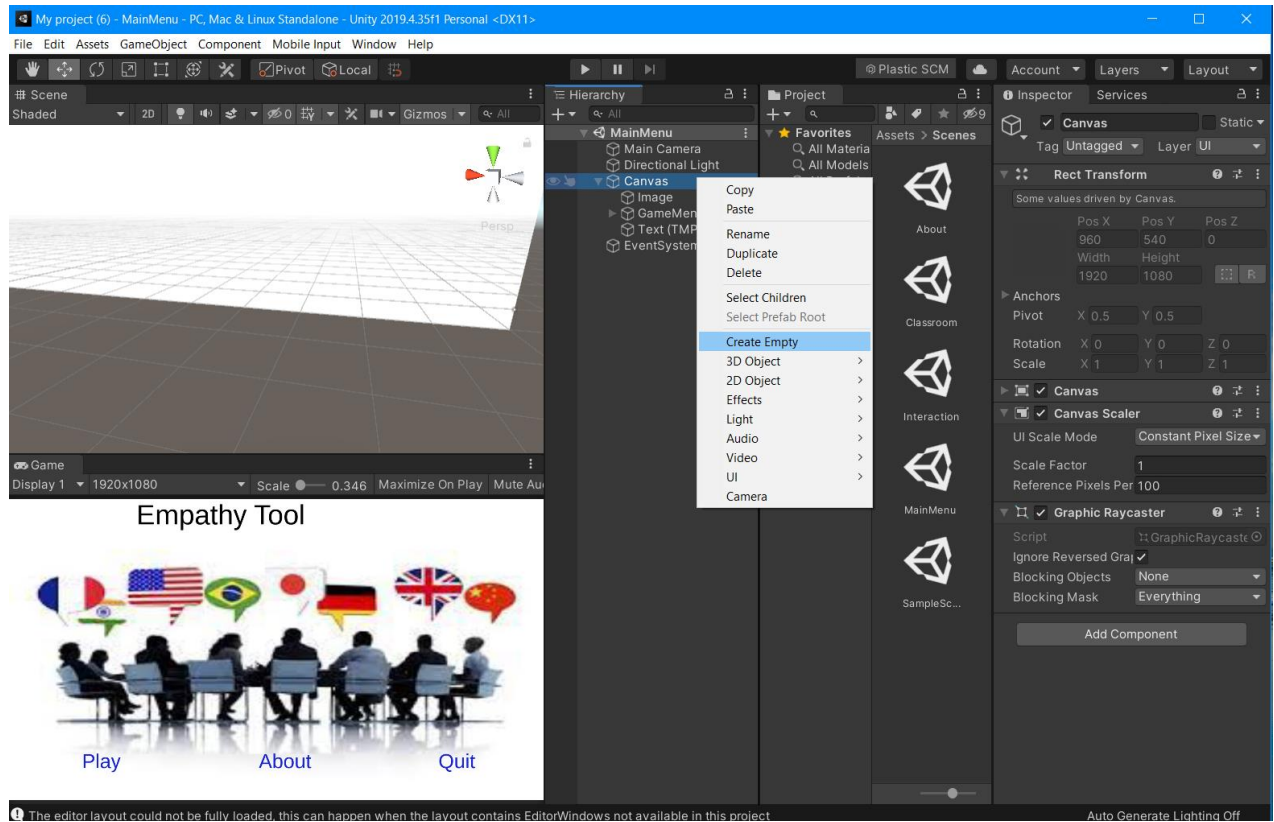


Figure 16: Creating an Empty

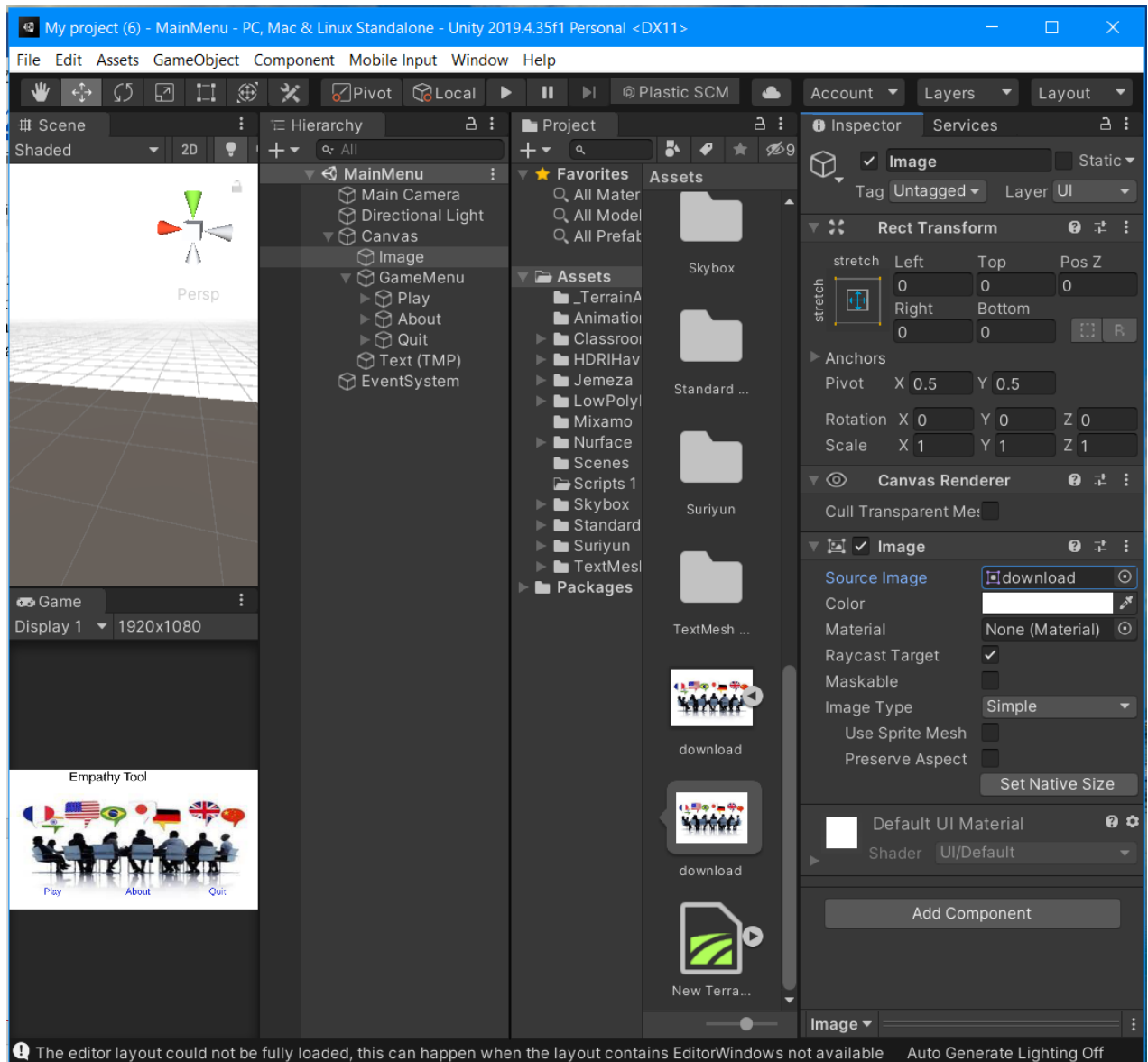


Figure 17: Image Background

3. The user can then create buttons called “Play”, “About”, and “quit” by right clicking on game menu and inserting a button as shown in figure 17.

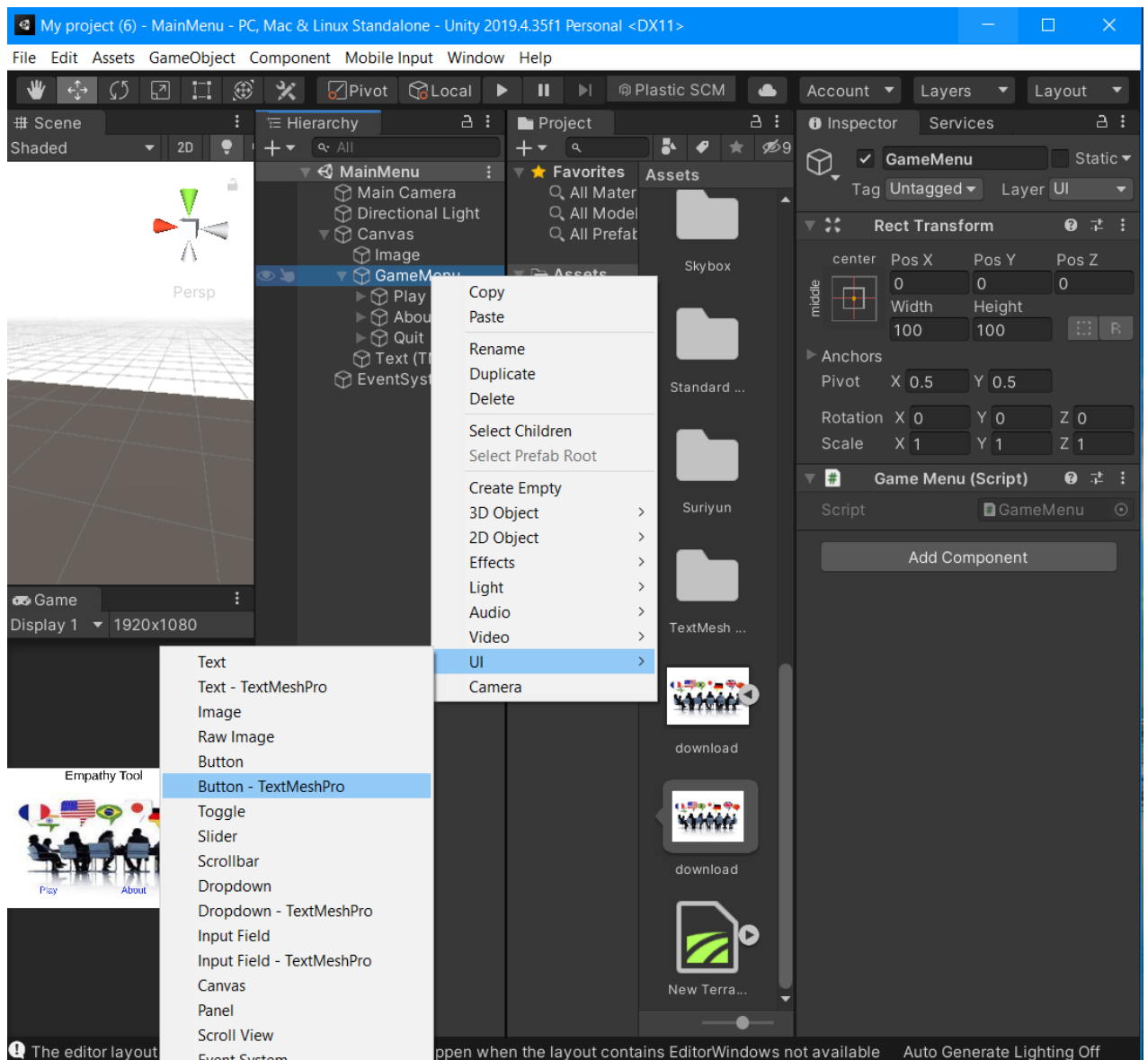


Figure 18: Creating a Button

4. You will then need to create a script for the buttons under GameMenu. You can do this by selecting “Add component” → “New Script” as shown in figures 18. Then, you can edit the script into the one shown in figure 19 by double clicking on the script.

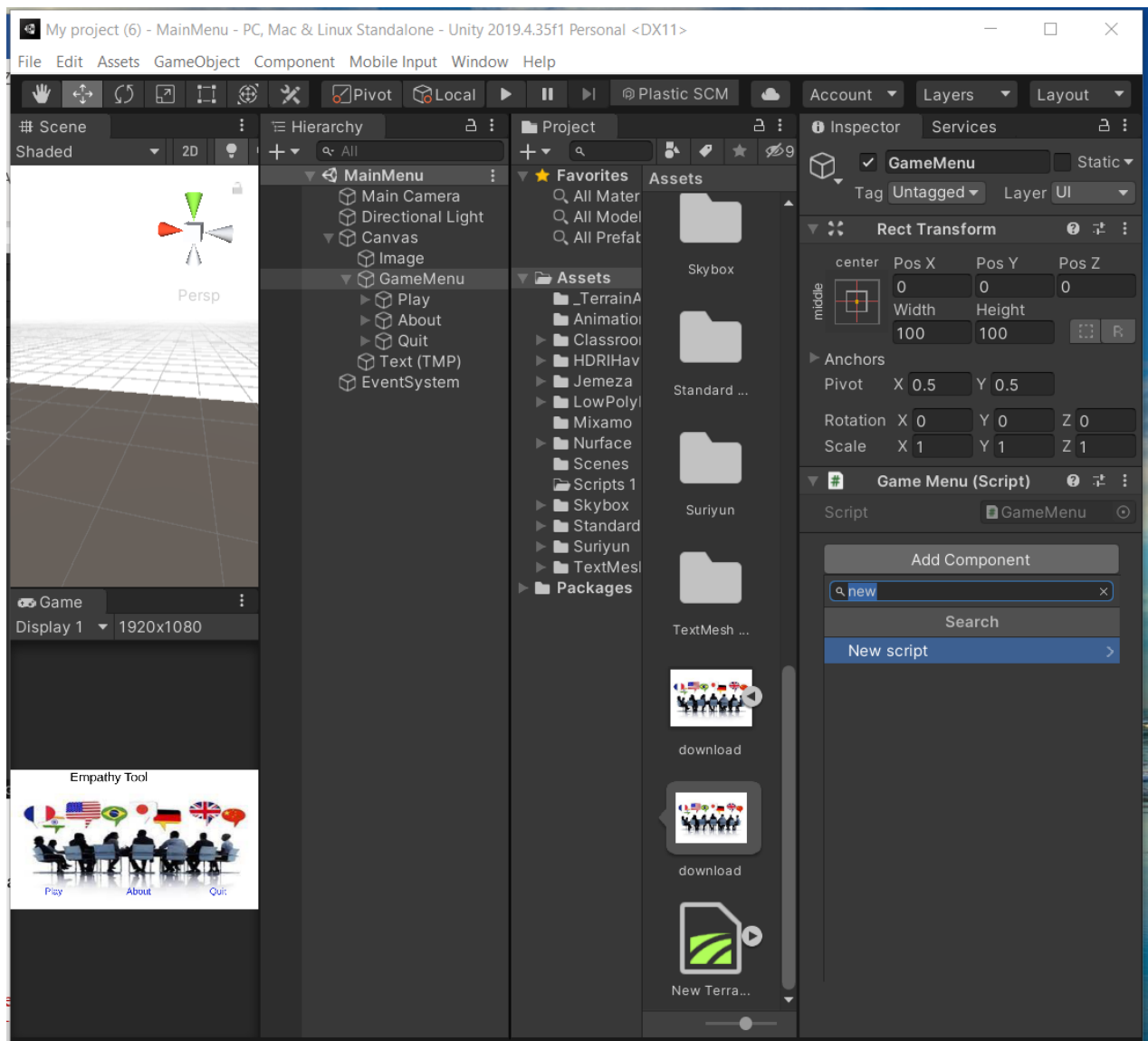


Figure 19: Creating a Script

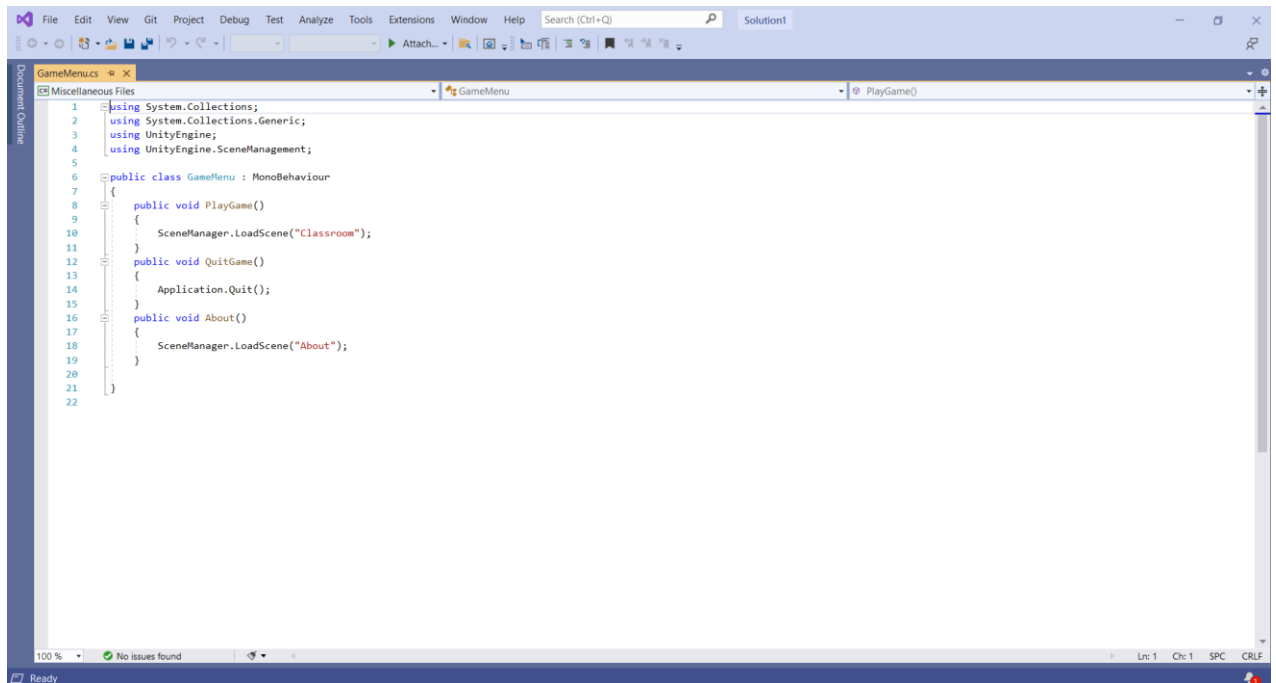


Figure 20: GameMenu Script

5. Once the script is saved, return to unity. Navigate to the inspector window of the play button and add an “On Click()” and select the GameMenu script. Once this is done, there will be another selection to make, select “GameMenu.PlayGame”. As shown in figure 20.

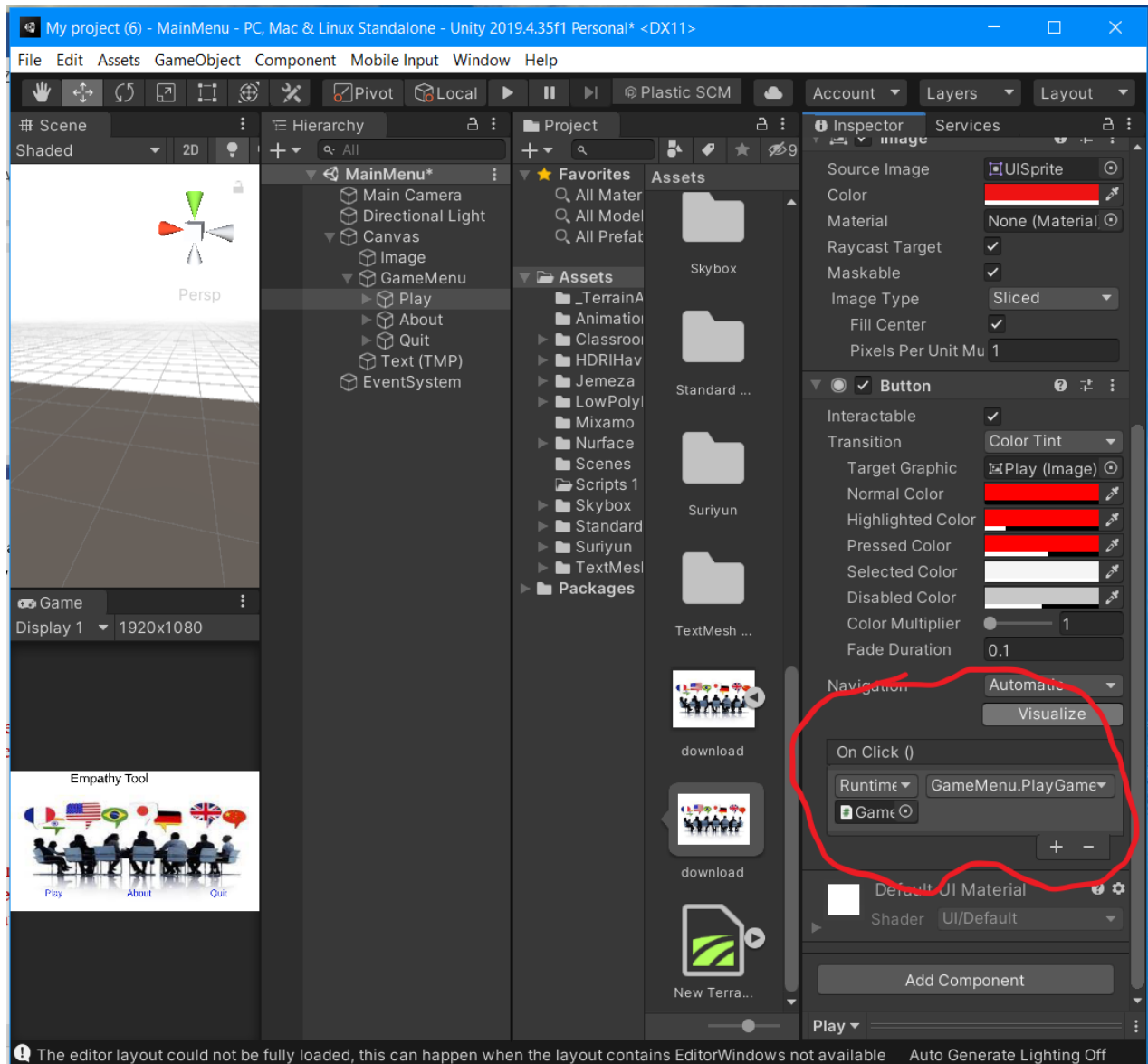


Figure 21: Setting up The Buttons

Repeat this process for the other buttons but replacing the “GameMenu.PlayGame” with “GameMenu.About” and “GameMenu.QuitGame”.

6. You can style the buttons by using the “color bars” in the inspector window which are displayed in figure 21.

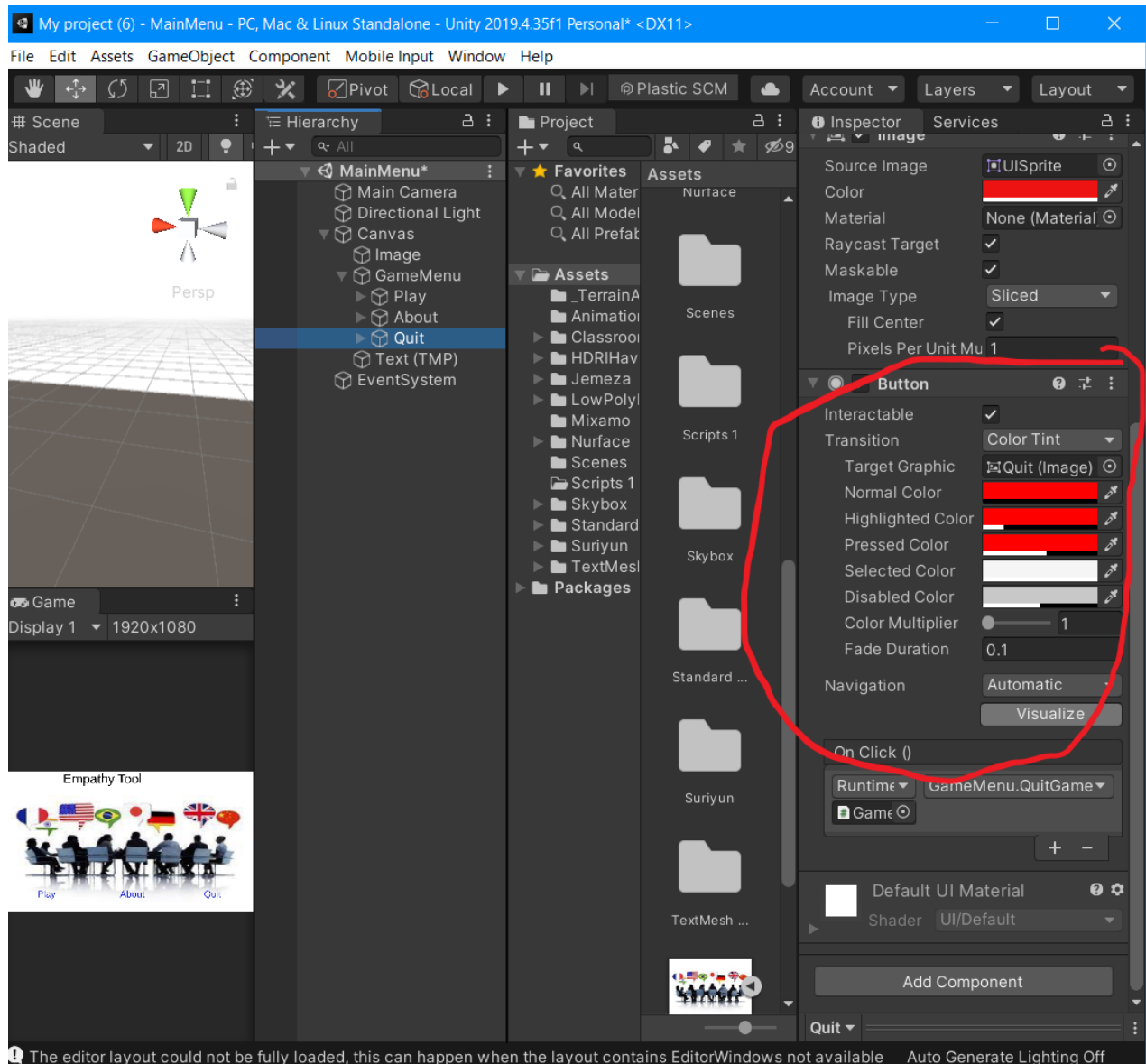


Figure 22: Botton Aesthetics

The menu scene is now complete.

About Page

1. Navigate to the about page and set up an image background in the same way as the game menu scene.
2. Next, create an empty object (refer to figure 16) called "BackButtons". Along with three buttons called "Back to Menu" , "Quit_2" , "Play_2" as done in figure 18.

- Next create a script on the empty object “GameButtons” as shown in figure 19 and call it “About”. Add the following code to the script.

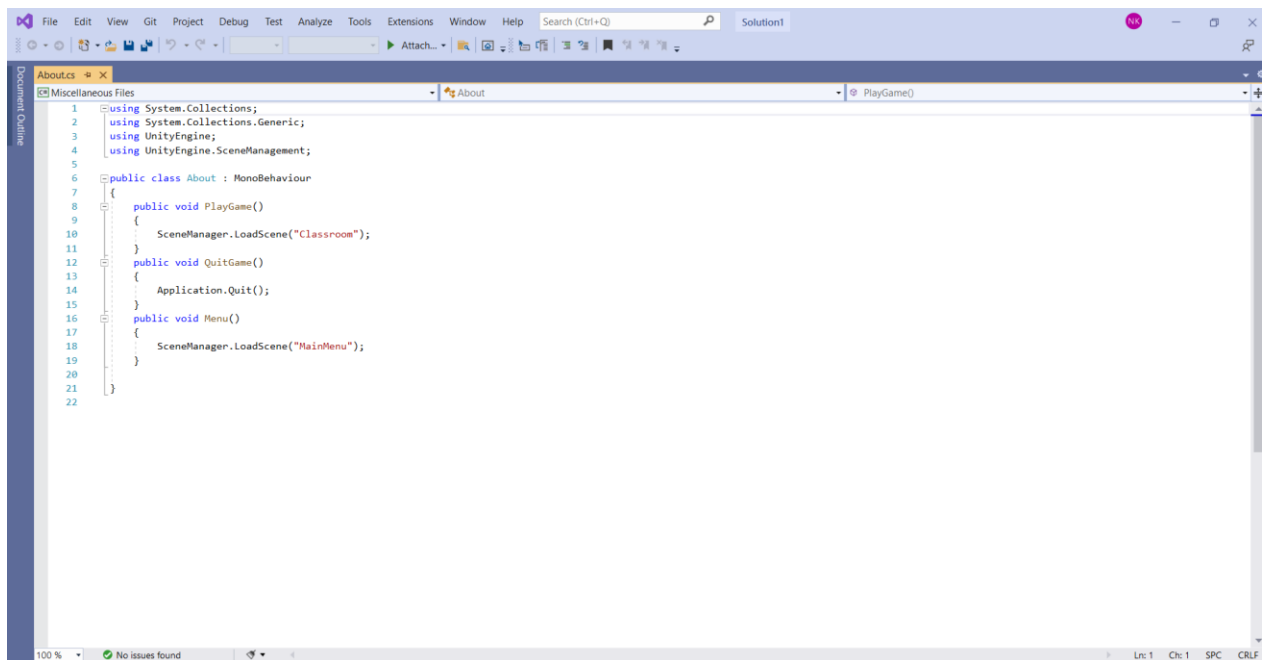


Figure 23: About Page Script

- Assign the commands from the script to each button as shown in figure 21. Finally, you can style the buttons as shown in figure 22.

Classroom Scene

- First, navigate to the “Classroom” scene and create a new terrain object as shown in figure 24.

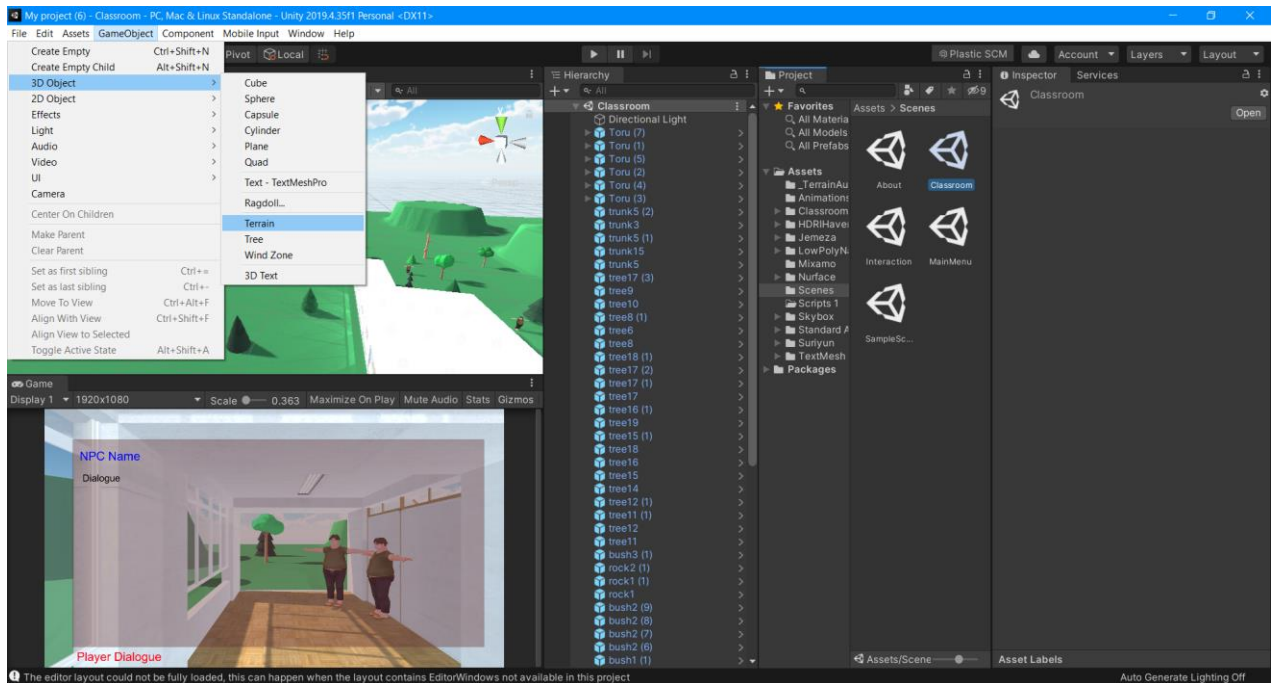


Figure 24: Inserting a Terrain

- Once the terrain has been created, the user can modify it by following figure 25.

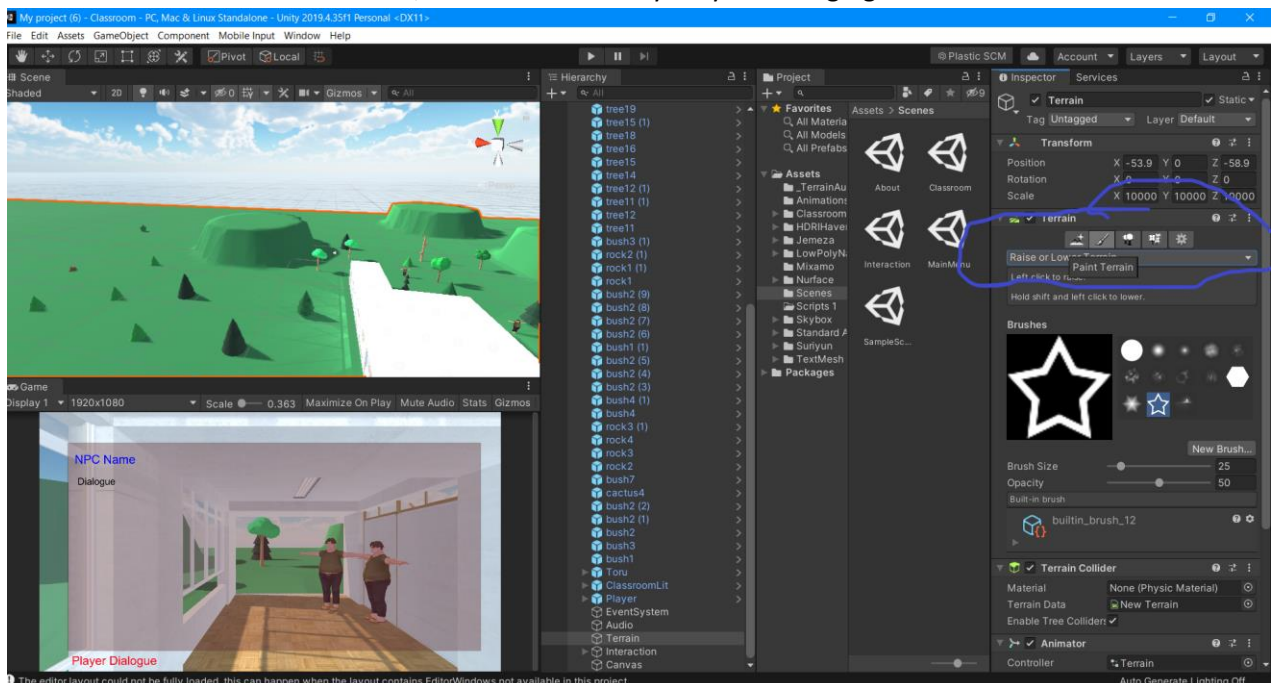


Figure 25: Modifying Terrain

- Using the Unity website, Download and import the standard assets package from unity. Once this is done, there should be a “standard assets” folder under assets. Import the “RigidBodyFPSController” prefab into the scene (make sure to delete the “Main Camera” from the Hierarchy window).

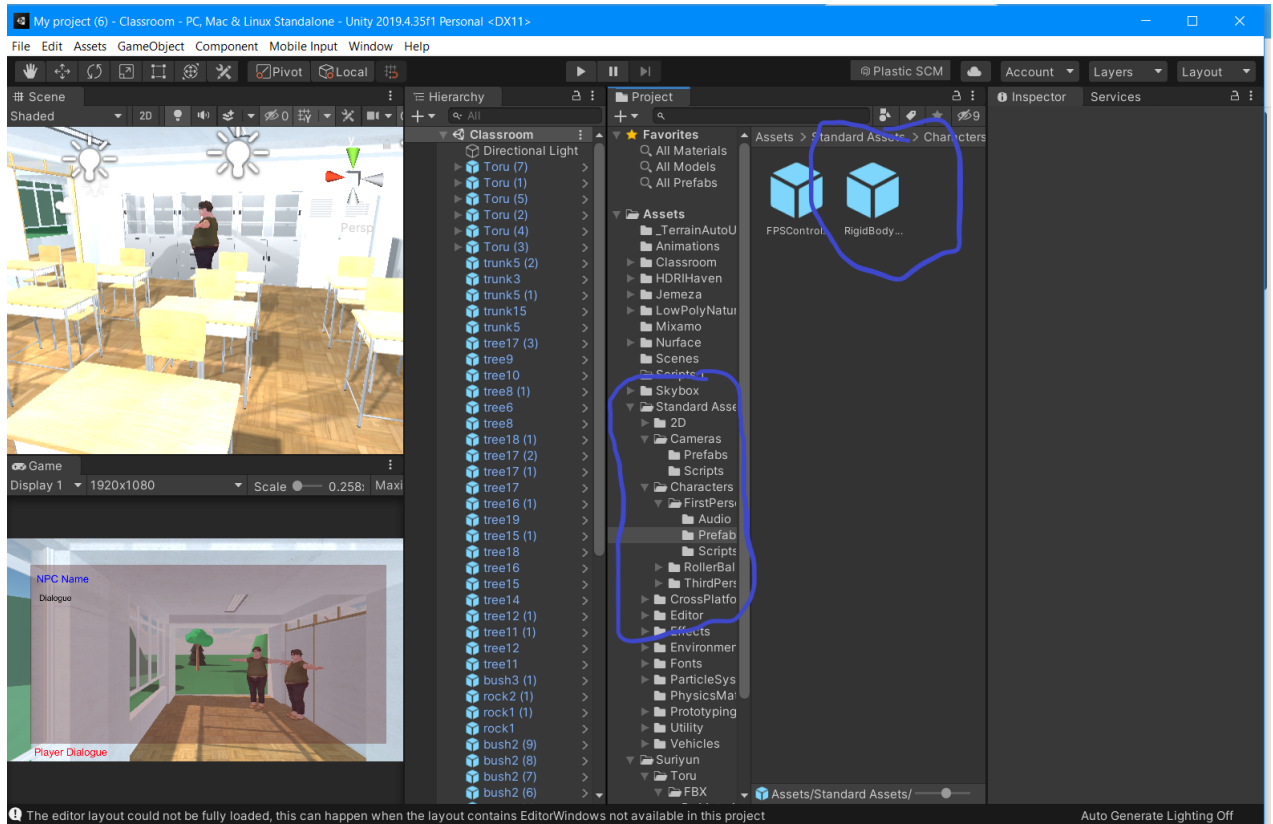


Figure 26: First Person Controller

Once this is brought into the scene, equip it with the “Player” Script found in figure 31.

Note: If developing the VR version, you would instead use the “VR Body” asset in place of the standard assets FPS.

- Once the first-person controller has been imported in, import all the assets from the BOM by following the links in the BOM. Then, go to the Assets > LowPolyNaturePropsPack > Prefabs folder and insert the items you want by dragging and dropping.

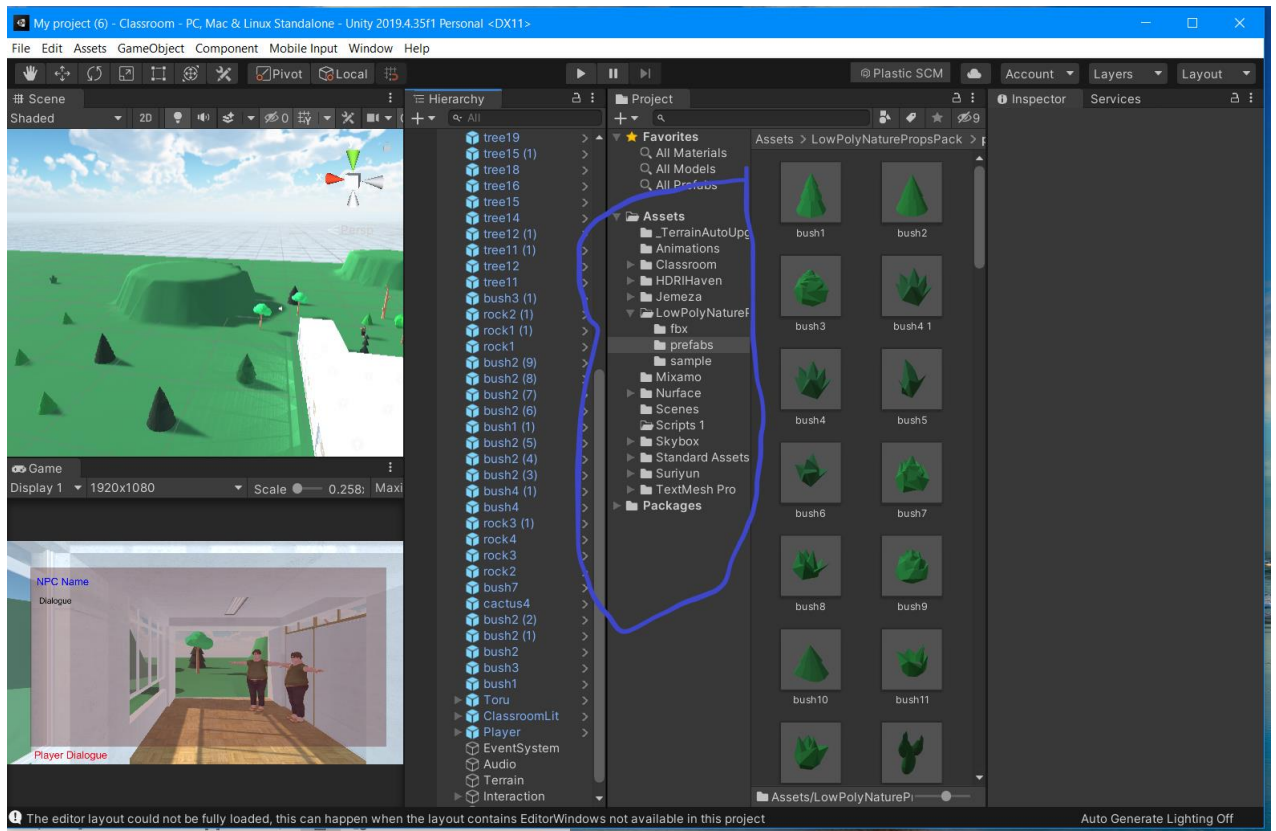


Figure 27: Inserting Low Poly Nature Props

5. Once the nature features have been added, The same thing can be done for the classroom asset by going to the Assets > Classroom > Prefabs and dragging the “Classroom Lit” Prefab in the the scene.

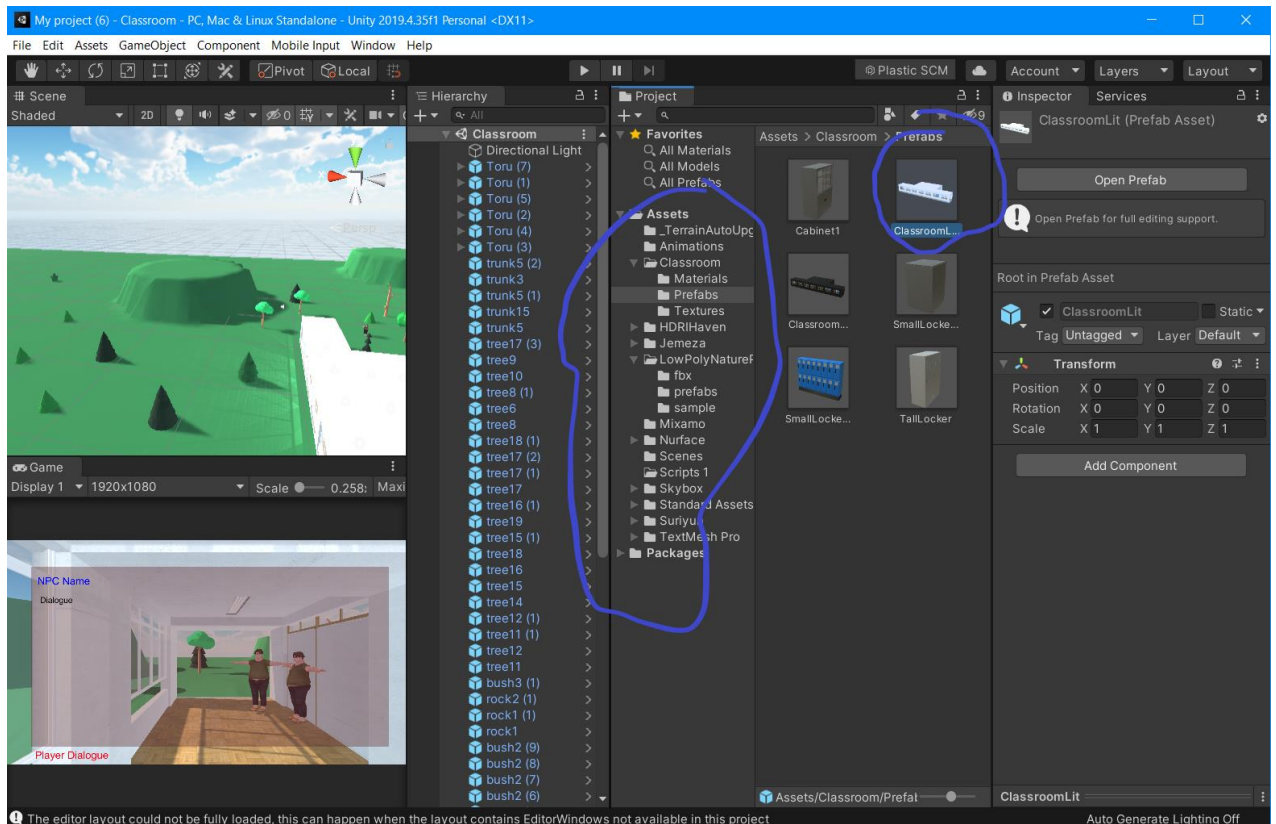
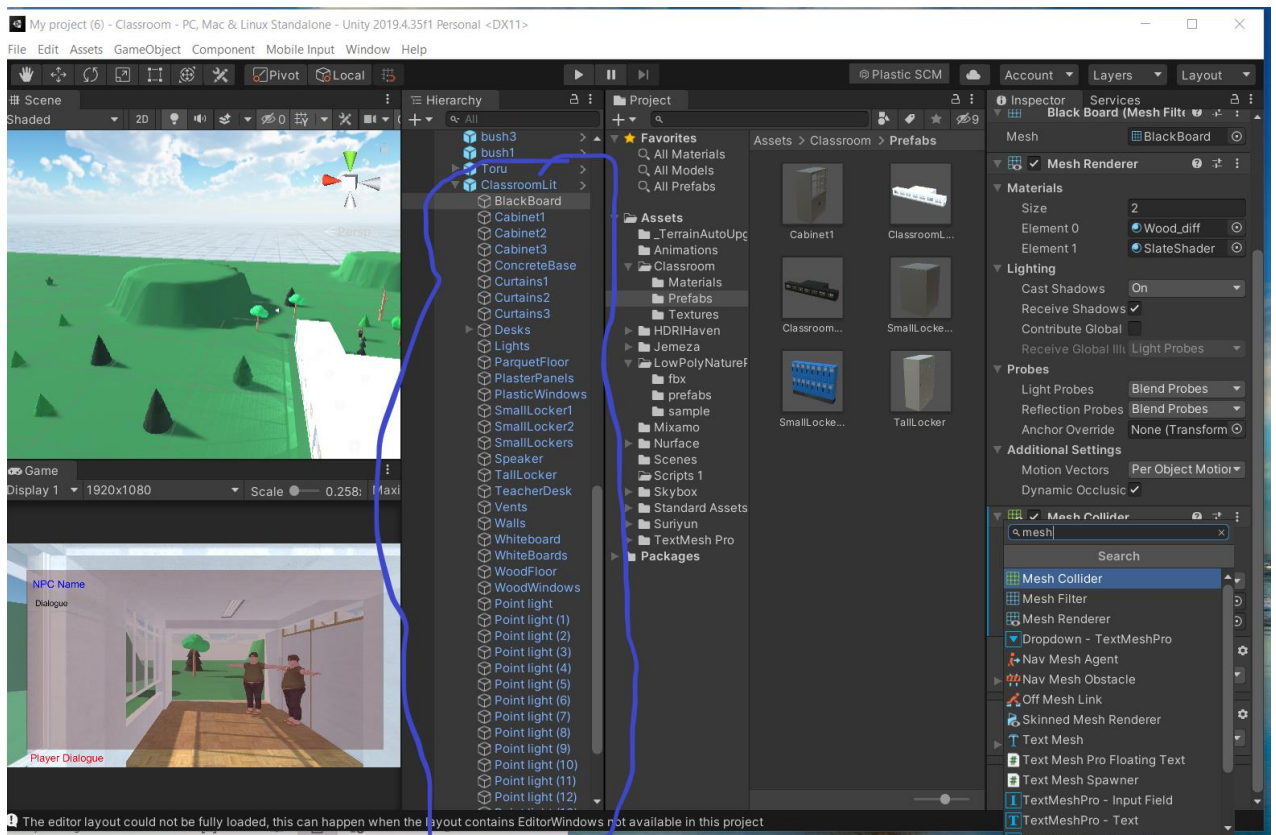


Figure 28: Inserting the Classroom

Once the classroom is inserted, a mesh collider must be added to every component of the prefab to prevent phasing. This is shown in the following figure.



This must be done for each subcomponent of the classroom prefab.

6. The final thing to import at the moment is the “Toru” NPC. You can create as many as you want by dragging the prefab from the Assets > Suriyum > Toru > Prefab folder.

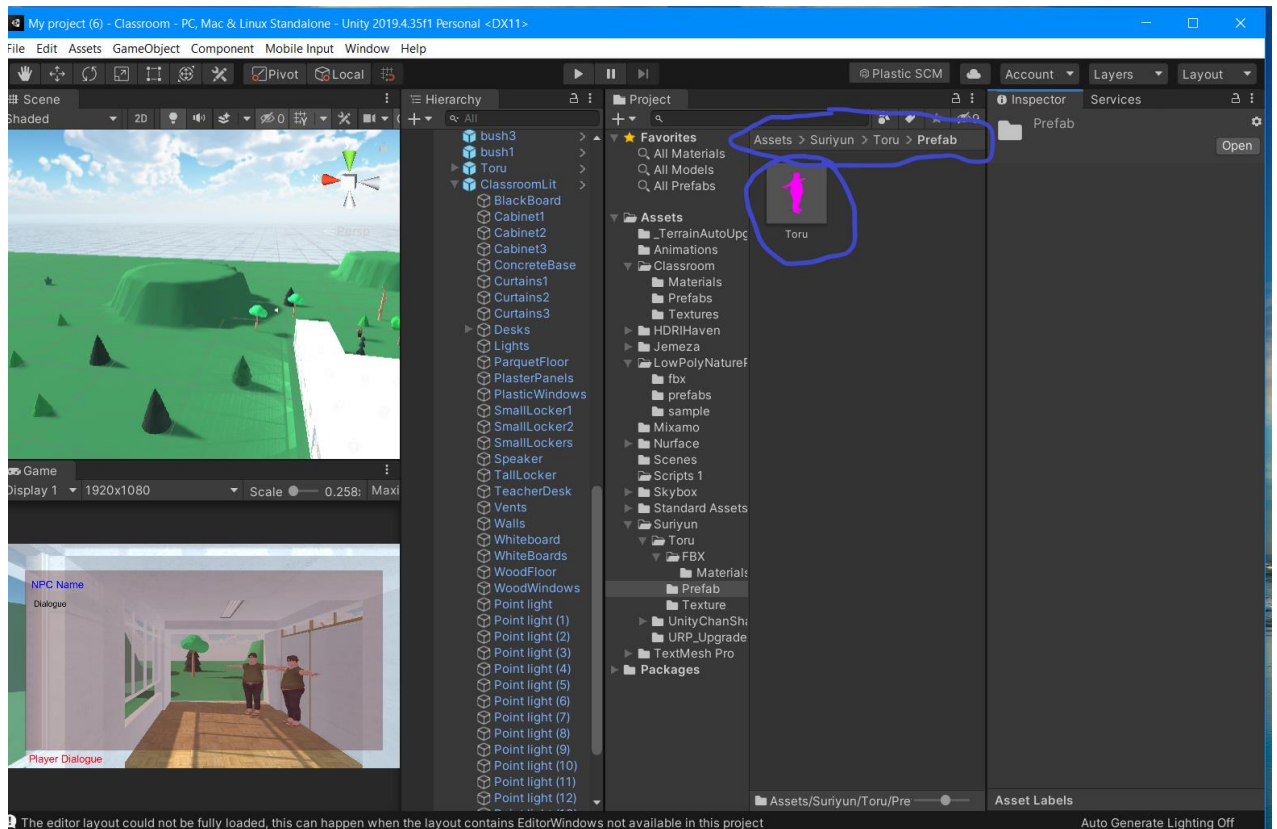


Figure 29: Toru Prefabs

Any NPC animations can be added by following the following video tutorial.

<https://www.youtube.com/watch?v=9H0aJhKSIEQ>

- Now that the Prefabs have all be put into the environment, the scripts that would control the interactions of the NPC's. The first script is called "NPC" Followed by "Player" and Finally "DialogueManager". The scripts are displayed bellow in figures 29-32.

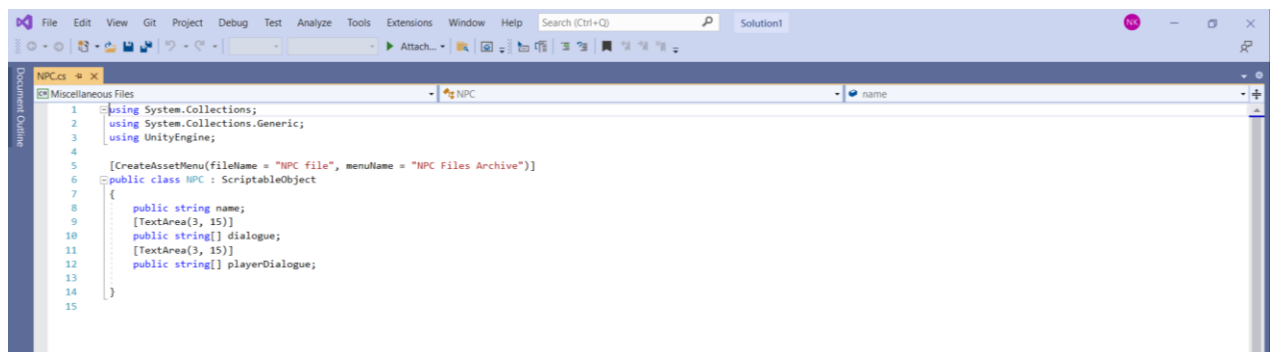
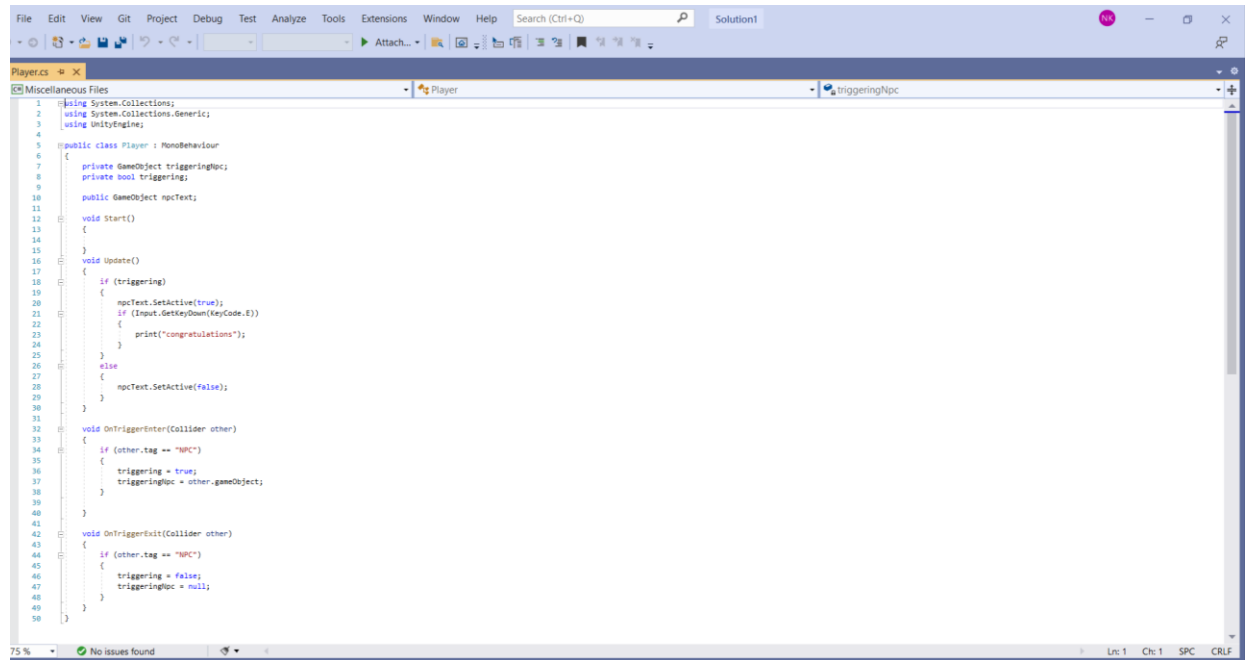


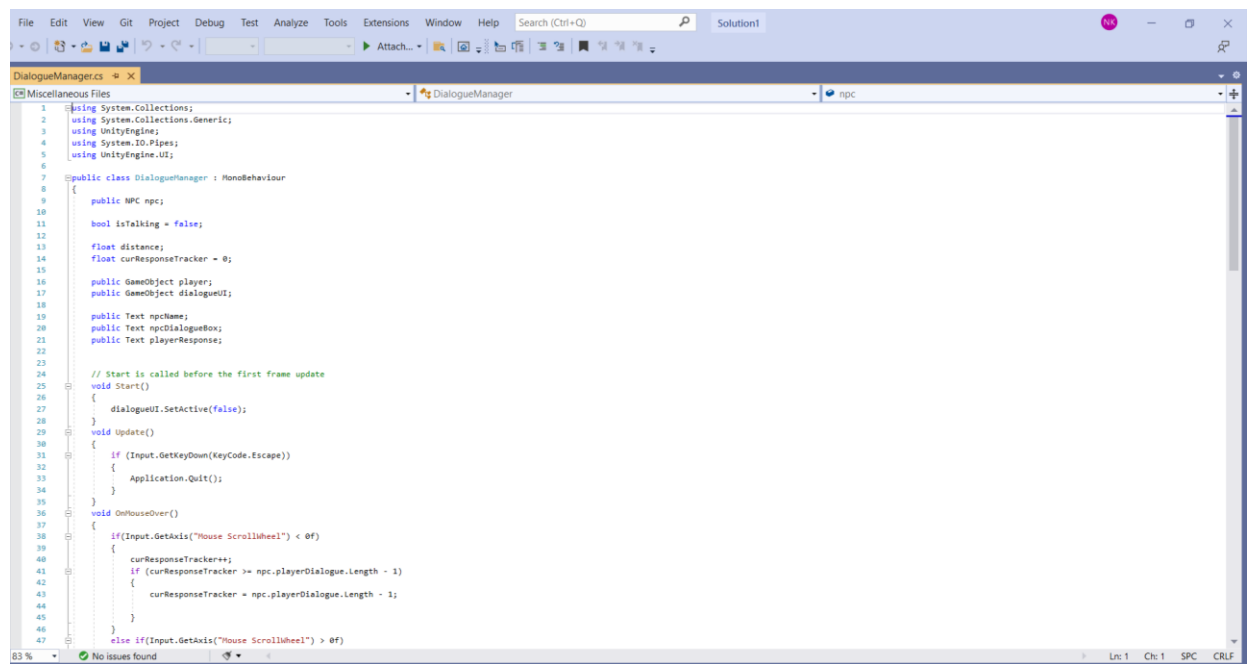
Figure 30: NPC Script



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4
5 public class Player : MonoBehaviour
6 {
7     private GameObject triggeringNpc;
8     private bool triggering;
9
10    public GameObject npcText;
11
12    void Start()
13    {
14    }
15
16    void Update()
17    {
18        if (triggering)
19        {
20            npcText.SetActive(true);
21            if (Input.GetKeyDown(KeyCode.E))
22            {
23                print("congratulations");
24            }
25        }
26        else
27        {
28            npcText.SetActive(false);
29        }
30    }
31
32    void OnTriggerEnter(Collider other)
33    {
34        if (other.tag == "NPC")
35        {
36            triggering = true;
37            triggeringNpc = other.gameObject;
38        }
39    }
40
41    void OnTriggerExit(Collider other)
42    {
43        if (other.tag == "NPC")
44        {
45            triggering = false;
46            triggeringNpc = null;
47        }
48    }
49 }
50
```

75 % No issues found Lrn: 1 Ch: 1 SPC CRLF

Figure 31: Player Script



```
1 using System.Collections;
2 using System.Collections.Generic;
3 using UnityEngine;
4 using System.IO.Pipes;
5 using UnityEngine.UI;
6
7 public class DialogueManager : MonoBehaviour
8 {
9     public NPC npc;
10
11     bool isTalking = false;
12
13     float distance;
14     float curResponseTracker = 0;
15
16     public GameObject player;
17     public GameObject dialogueUI;
18
19     public Text npcName;
20     public Text npcDialogueBox;
21     public Text playerResponse;
22
23
24     // Start is called before the first frame update
25     void Start()
26     {
27         dialogueUI.SetActive(false);
28     }
29
30     void Update()
31     {
32         if (Input.GetKeyDown(KeyCode.Escape))
33         {
34             Application.Quit();
35         }
36     }
37
38     void OnMouseOver()
39     {
40         if (Input.GetAxis("Mouse ScrollWheel") < 0f)
41         {
42             curResponseTracker++;
43             if (curResponseTracker >= npc.playerDialogue.Length - 1)
44             {
45                 curResponseTracker = npc.playerDialogue.Length - 1;
46             }
47         }
48         else if (Input.GetAxis("Mouse ScrollWheel") > 0f)
49         {
50             curResponseTracker--;
51             if (curResponseTracker < 0)
52             {
53                 curResponseTracker = 0;
54             }
55         }
56     }
57 }
58
```

83 % No issues found Lrn: 1 Ch: 1 SPC CRLF

Figure 32: DialogueManager Script Part 1

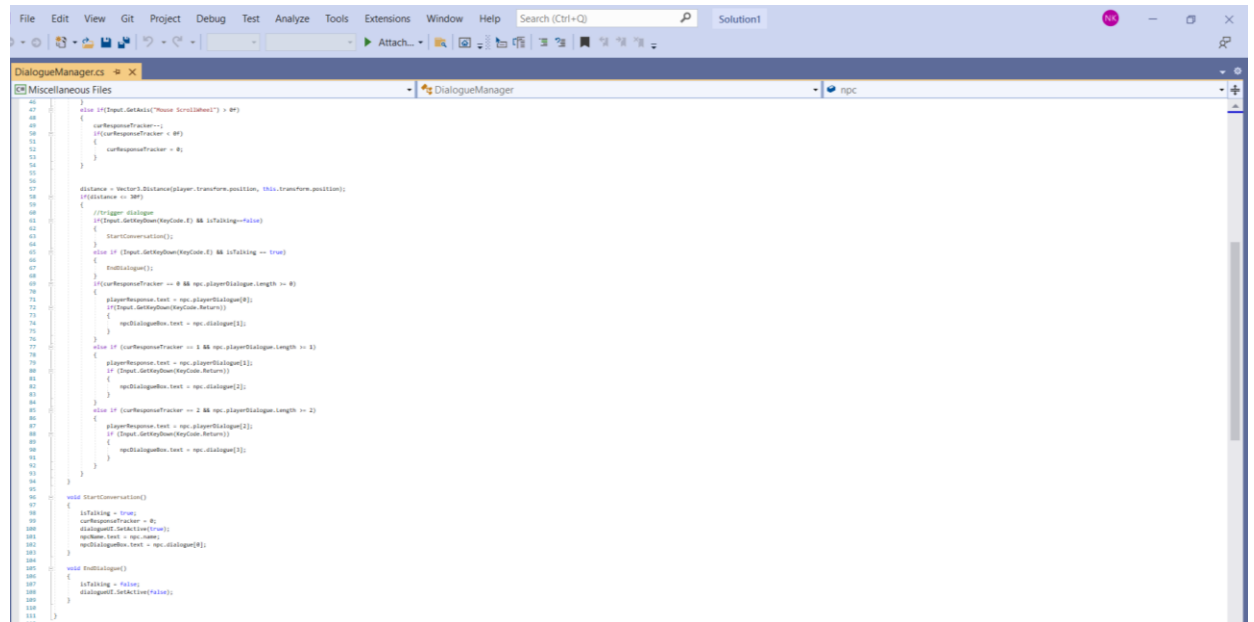


Figure 33: DialogueManager Script Part 2

8. Create a scripts folder under assets, and move all the created scripts into this folder. Then, right click in the folder and find “create > NPC Files Archive” and name it “Friend”.

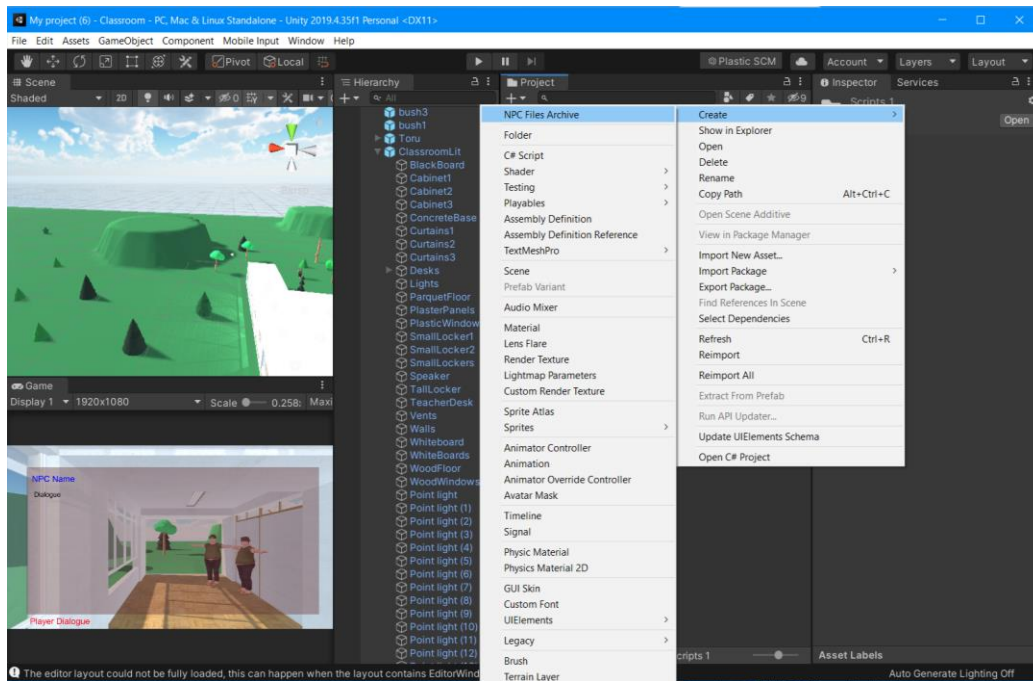


Figure 34: Creating an NPC Archive

9. Once the archive is created, you can create a dialogue by manipulating the text found in the following diagram.

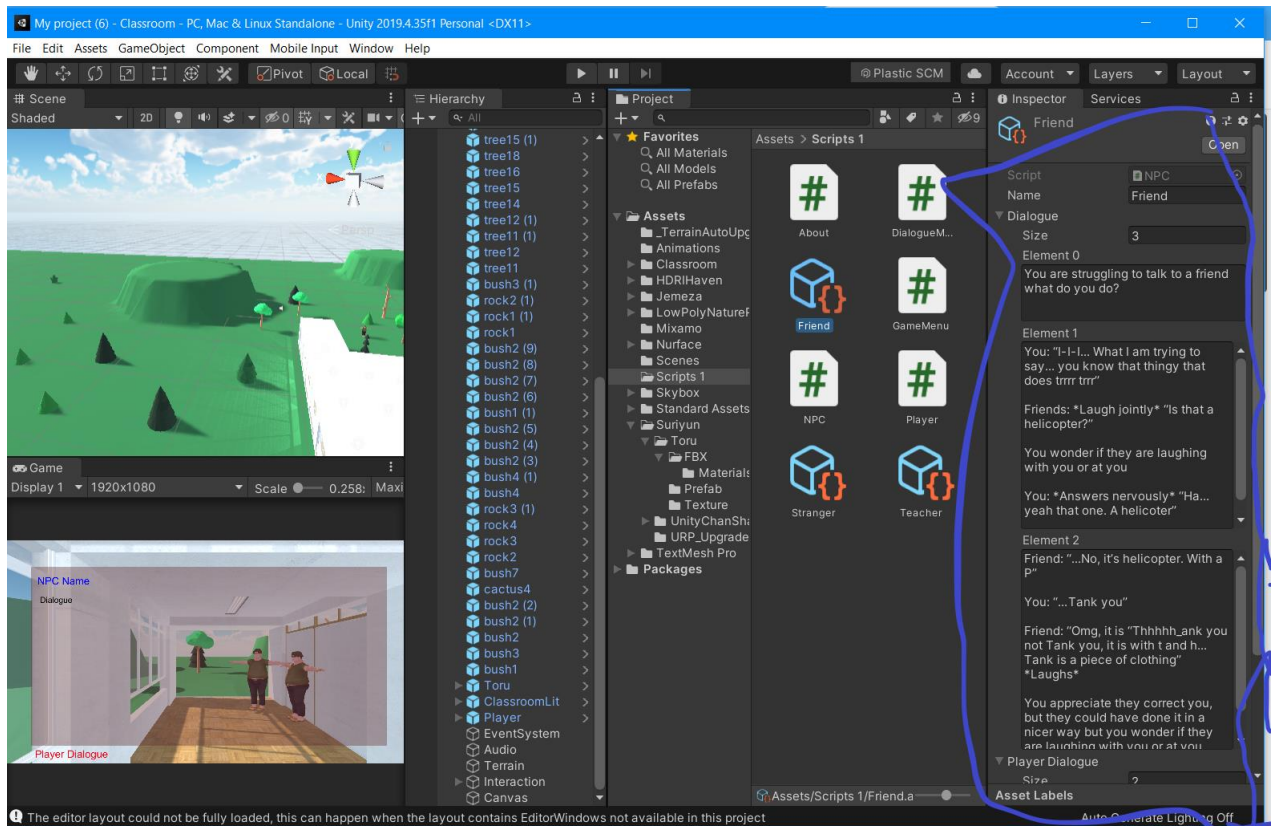


Figure 35: Creating a Dialogue

10. Finally, the NPC that you would like labelled as “Friend” for the scene would equip the dialogue manager script and select the following settings.

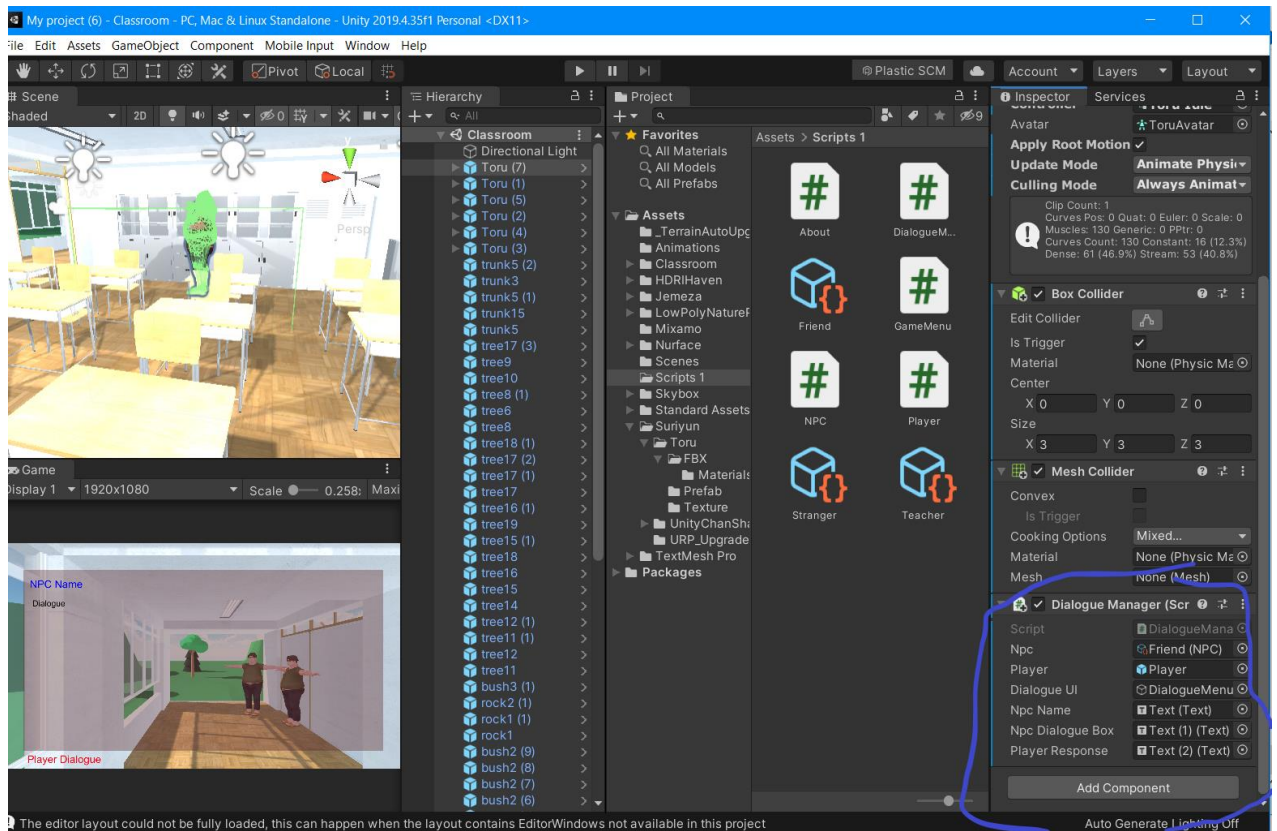


Figure 36: Script Settings

11. Repeat this process with every NPC and Dialogue as you please, there could be more interactions than shown in the prototype.
12. The virtual environment should now be ready as in our final prototype.

6.2.3 Testing & Validation

For the Unity aspect of the project, testing was done constantly. With how coding for Unity works, every part must be tested alongside its creation. Coding can be fickle, and thus it is necessary to receive immediate feedback to prevent more code from being built upon broken code. The testing was also especially simple. With more physical projects, there is an emphasis on collecting numerical data, graphing results, and making multiple prototypes to measure comparative strengths and weaknesses. However, with Unity the primary subject being tested is whether the code performs as intended or not, and no numerical data was needed for the product. For example, the creation of the menu required a scene with usable buttons and so the tests were conducted at once after both the scene met the requirements component-wise and the code was written and integrated. The test was for the sole purpose of finding if the buttons sent the user to the correct scenes or not, as well as if the buttons could be selected. No other data was needed, nor could more data aid the project.

As a result of the high number of tests and the lack of viewable data collected from the tests, it is difficult to show the amount of testing and what was done during the tests.

Although the final product was designed for only 10 to 20 minutes of usage before fresh content runs out, indefinite usage is possible. The program should run until it is exited out of or the system that is running it crashes. However, it is to be noted that prolonged use of VR equipment may cause health problems and thus it is best to consult its user manual for more information on its prolonged usage.

7 Conclusions and Recommendations for Future Work

Our group encountered numerous demanding situations, we overcame obstacles and we also received plenty of new and interesting realistic competencies in the course of working on our assignment that we may additionally use in our professions and/or our future courses. We had quite a few assignment-related thoughts/ideas that failed to get implemented, however, we simply had to prioritize what we change we wanted to see in our society, which became the product's functioning. We additionally realized the importance of our assignment's presentation. We found out that self-assurance goes an extended manner and that the judges would consider in our proposal if we did. We, as a group, discovered that for a project to become the change we wanted to see in our society, we had to learn as teammates to be on the same page and with that being accomplished, we knew our combined ideas would turn into the project that would affect positively our society. These small details we learned throughout preparing and working on our project will assist us in being higher lecturers and engineers in the future.

8 Bibliography

- Benefit Studio. (2016). *Project empathy*. Project Empathy. Retrieved January 29, 2022, from <https://www.projectempathyvr.com/>
- Best Practices in Equity, Diversity and Inclusion in Research*. (2021, June 22). Government of Canada. Retrieved January 28, 2022, from <https://www.sshrc-crsh.gc.ca/funding-financement/nfrf-fnfr/edi-eng.aspx>
- Diversity inclusion training in virtual reality*. Equal Reality. (2021, September 10). Retrieved January 29, 2022, from <https://equalreality.com/>
- Stanford University. (2020). *1,000 cut journey*. VHIL. Retrieved January 29, 2022, from <https://stanfordvr.com/1000cut/>
- Creating an Equitable, Diverse and Inclusive Research Environment: A Best Practices Guide for Recruitment, Hiring and Retention*. (2021, November 17). Government Canada. Retrieved January 28, 2022, from https://www.chairs-chaieres.gc.ca/program-programme/equity-equite/best_practices-pratiques_examplaires-eng.aspx
- The 360 Guy. “The Ultimate VR Headset Comparison Table: Every VR Headset Compared.” *360 Camera Reviews and Guides*, 13 May 2021, <https://www.threesixtycameras.com/vr-headset-comparison-table/>.
- Playstation. “Tech Specs.” *PlayStation*, <https://www.playstation.com/en-ca/ps-vr/tech-specs/>.
- Miami University. (n.d.). *Diversity and inclusion training through virtual reality*. Diversity and Inclusion Training Through Virtual Reality - Miami University. Retrieved February 5, 2022, from <http://www.miamioh.edu/it-services/initiatives-and-projects/student-tech-grant/proposals-fy20/fsb-diversity-training/index.html#:~:text=%2484%2C990%20for%20Customized%20VR%20Solution,%2419%2C900%20annual%20support%20and%20consultation>
- NPC interaction / C# tutorials in Unity*. Youtube. (n.d.). Retrieved March 7, 2022, from <https://www.youtube.com/watch?v=-3SdEUhlDzY>
- Setup: How to create a flexible dialogue system - youtube. <https://www.youtube.com/watch?v=RfLCzDzkvb0> (accessed Mar 28, 2022).

APPENDICES

9 APPENDIX I: Design Files

Table 2. Referenced Documents

Document Name	Document Location and/or URL	Issuance Date
Storyline (Dialogue)	Speakeasy Project Team 2 3 MakerRepo (makerepo.com)	March 15, 2022
BOM	<ul style="list-style-type: none">• Speakeasy Project Team 2 3 MakerRepo (makerepo.com)• Budgeting.xlsx	February 20, 2022
Unity code	Speakeasy Project Team 2 3 MakerRepo (makerepo.com)	March 15, 2022
Story baseline	Speakeasy Project Team 2 3 MakerRepo (makerepo.com)	March 4, 2022