Gesture Based UI Project Design Document

by

Brendan Toolan & Antaine Ó Conghaile

Github Repo Link:
https://github.com/Antaine/TheMadMaze

Introduction

For our project for the module Gesture Based UI as part of the course of Software Development, Antaine Ó Conghaile and Brendan Toolan have decided to use ‘Voice recognition’.

Originally both of us were to make our gesture based project using the Myo armband hardware, however due to the outbreak of Covid-19 this meant that we both did not have access to the Myo hardware that would of have been needed to develop our original idea.

Both of us had to move back to each of own homes. So we decided to change up the project by using voice recognition as we would both have access to voice recognition programs on each of our laptops/computers. Our project is a game made using Unity and with the programming language ‘C#’.

We picked C# as it is easier to use with Unity as it is supported by it and Visual Studio 2017. The game will consist of the player trying to navigate through a maze with the end-goal being to reach the exit. Once the player gets to the end, he/she/they will advanced to the next level.

While the player goes through the maze, there will be obstacles for them to avoid. There will also be enemies that will try to catch the player in the maze. The player will then navigate through the maze using the built in voice recognition to enhance their movement to suddenly stop, jump or speed up.

Purpose of the application

The purpose of this project was to create a game that is able to recognise the voice commands the user would say aloud. The project is a game set in the maze where the user must find there way out of. They navigate through the maze like so using the key arrows on the keyboard or WASD.

However the user is able to use voice commands such as ‘Run’ so the player would be able to sprint through the maze and can also say the word jump so the player would be able to jump over any obstacles or anything in there way. The user will also be able to stop suddenly by saying pause with just there voice to activate it. There was a plan to add menu navigation with voice commands however we did not have time to add this feature.

Gestures Identified

Seeing that we would be using voice recognition for the gestures that will be used in the project it would be safe to assume that the gestures that would be used would be voice commands from the user to run the game.

We decided that as we had to change our idea for the project entirely due to the ongoing circumstance that is Covid-19, we would create a whole new project that would use voice recognition so the gestures that we would use would be voice based and available for testing in our own homes. The gestures that would be used in our project would voice commands that would let the user to pause the game, make the playable character jump and also make make the player move faster or sprint and stop suddenly.

Hardware Used in creating Application

Originally we were going to use the Myo Arm bands that were made available to the whole course to use. However due the Covid-19 pandemic we did not have access to the Myo hardware.

It was suggested to us that we use voice libraries that were available to us universally. Seeing that we are using Unity to create this project we would use the Microsoft Speech Engine for the development of our project. Compared to the Myo armbands that we planned to use, we had to completely change what our project would be.

Architecture for the Solution

he biggest and main problems that we faced doing this project was that we had to change what hardware we would be using for the project and also to change what the project would be as well.

In order to solve this issue, both of us kept in contact through the use of social media and Discord to discuss what were the options that were available to us during this time.

We also kept a list of what each person would do for the project and kept checking in on each other to see how we were progressing. We had no real trouble or issues when using the voice recognition that was available to us. However we both were using different operating systems so it was at times difficult for us as not all resources would be made available to us. As well our internet connection was not ideal making downloading updates slow our work rate.

Conclusions & Recommendations

In conclusion from doing this project for this module we learned that doing a project based on voice recognition was a challenge considering that we both had limited time to re-think our project and re-strategise how to do this from home.

If we were to do this again we would both get a head of doing this project immediately and ask to use the hardware we both wanted to use for it so we could start on it straight away and see if this hardware would be suitable for our idea.

But due to circumstance not within our control having to switch what hardware to use for this has proven a bit difficult for us seeing that we had to learn about using voice recognition and also had to re-work what our project would be with all things considering.

Upon reflection on the work we had done with this project we were both surprised what we had accomplished and also said to each other that if we were do this again we would use voice recognition as the basic of our project again as it is easy to use and understand.

References

- **https://www.youtube.com/watch?v=-igoV67B5h8**
- **https://www.youtube.com/watch?v=VnG2gOKV9dw**
- **https://www.youtube.com/watch?v=b4oqOdBCy3c**
- **https://docs.microsoft.com/en-us/windows/mixed-reality/voice-input-in-unity**
- **https://lightbuzz.com/speech-recognition-unity/**
- **https://docs.microsoft.com/en-us/archive/msdn-magazine/2014/december/voice-recognition-speech-recognition-with-net-desktop-applications**
- **https://yellrobot.com/speech-recognition-in-video-games/**