Jacob Pascoe

17828121

Multiplayer Game Development

Reflective Report

[GitHub Link](https://github.com/Pascoe007/MultiPlayer-GameClient)

Contents

[Introduction 2](#_Toc61356193)

[Implementation 2](#_Toc61356194)

[Critical Review 5](#_Toc61356195)

[Conclution 5](#_Toc61356196)

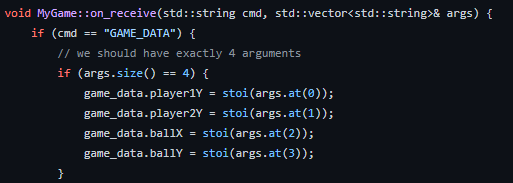
# Introduction

This is the report for CI628, this will cover the implementation of this project and a critical review of what has been completed over this project. The design of this game is Boris Johnson themed as it has pictures and sounds related to Boris Johnson. The idea for this theme was how much he has been in our press due to Covid-19 and being our prime minister. Below is a link to YouTube of game play of this project. This will show all the parts that have been added to the client and show that this game works.

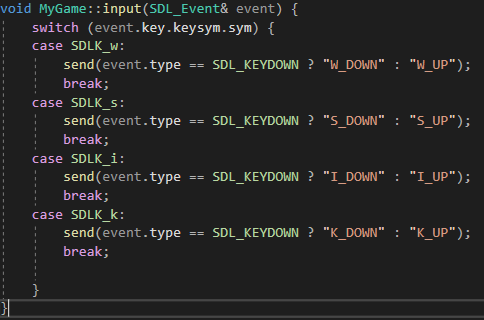
YouTube video

# Implementation

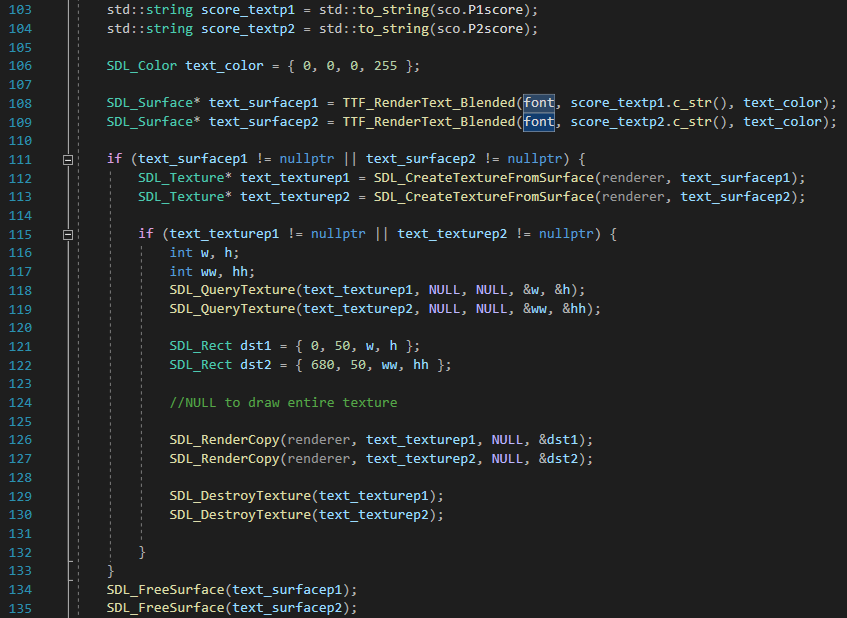
The plan for this project was to make a game client for the server provided for this course. As when the code was given to me, it was just a single block on the screen, I needed to add the player two paddle to the game. This was done by looking at the code that had been given to me. As the ball and player position is being sent from the server to the client, I needed to find the code that takes that data and converts it to usable data. As the server sends the data as a command lines with strings following that. This is handled in the code below. This code has been given to us:



These lines of code take the command line “GAME\_DATA” and breaks it in to smaller checks that can be applied throughout the code. In this case after game data there are 4 strings that hold the information for the player one and 2 y position and ball x and y position. Using the “GameData” struct in “MyGame.h” I applied the position for the player two to this Integer in the update function. This could then be used to render the bat on screen by making a cube on screen and setting the position to the player y value. This meant that I could have two paddles on screen that could be moved on the server. Following this, I added the ball using the small code as before. To make the paddles, I updated the paddle on the server side I had to add commands to send back to the server. This was done by making a switch statement and calling the send function to send the data from the client to the server. This was done with the code below. With this code added this means the server can receive four keyboard inputs done on the client side. The main part to stress about this part is that the client is not handling the movement of the paddles even though the client has keyboard inputs. It sends that data to the server then is applied to the game and is sent back to the client to update the paddles.

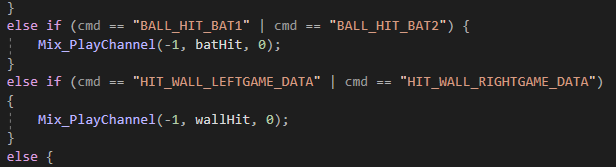


Adding the scores was the next part of this game. This would use the true type font (TTF) add on to SDL this makes it so you can render fonts into your game easily. With this you must find a font for your game in this case I used Arial. When getting the font on screen the first thing you have to do is initialise TTF this is done with one line of code which is “TTF\_INIT” then add an error message so you can see if anything goes wrong, all of this should be done before you run the game as you do not want to be loading anything while the game is running. Once the font has been loaded it needs to be passed to the “MyGame.h” script so you can render it. For rendering text, you will need a string in this case I converted an integer to a string that had been receive from the server. You will also need surface; this is made by using the font that was loaded earlier and the string also the text colour. This will then be used to make a texture. With this texture you can render it on screen. Below is the code I used for rendering the text on to screen. On line 134 and 135 of this code the surface is being freed, this has to been done otherwise the code would fill the user’s memory up as it will make a new surface every time render is called.



Once I had the basic game running, I wanted to add images to this project to make it look different to the basic game. This was done by adding images to the game. My theme for this game was Covid-19, so with this I added a picture of Boris Johnson head where the ball should be. I also added a background to the game. There were other images that I would have liked to add such as the covid-19 cell but could not find an image without the background. For this you must load the images in “Main.cpp” this is so you are not loading the image while the game is running. Once that’s is done, I pasted the data to the “MyGame” script. This would then create a texture from the surface that had been made when the image was loaded. This was then render on screen the with position the server sends for the ball. The background was loaded the same but was render with a fixed position. An issue that I ran into when coding this the ball was would be behind the background this was an easy fix as all I have to do was render the background first and everything after it.

The finial part to this game was to add music to it. This was done by first finding some sounds to play. For the music I found a parody of the Boris Johnson speech after the first lockdown and downloaded it. This was then added to the game by first loading the game and playing it when the game boots up. The next part was to add the sounds for when the ball hit the wall and bat. For this I found a sound board for Boris Johnson saying “yes” and “Ping-Pong”, I then did the same step for the music but passed it through to the game script. Then used what I learnt in moving the paddles and with the scores. This time used the commands “BALL\_HIT\_BAT” and “HIT\_WALL\_GAMEDATA” this would tell the client when the wall had been hit, one these where set up I used the code below to play the sounds once the ball had hit a wall or paddle.



# Critical Review

Over this Project I feel as there has been part of this project I have done well. The first part of this is the theme of the game, this in my opinion was done very well as there was an idea for this project to be a Boris Johnson themed pong game. With the images and sound this was done well, and it put the point across that that this is a parody game and is a joke. With this I fell there was way I could have done this better. My first plan to make this better was to add particles behind the ball of red dots, this was something that would have been easy to implement the issue with this was time. If I was to do this, I would make a class for all the particle variables and then send that to the rendering function of the code and have it follow the balls position.

Managing the game data for this game was another point I handled well as all of the data being received has all been handled and dealt with on the game client, this includes the player movement, ball position, wall hit, and paddle hit. With this I would have like to change a few things about the last two, as I had a few ideas about making it so when the ball hit one of the wall the ball position would go back to the middle of the screen this would stop the ball bouncing back to the back of the paddle and making it so the other player got two points when they should get one. This would have been done by editing the server code so once the ball hit on of the wall, it would be set to half of screen height and width. This would have been done in the “initPhysics” function in the server code. I also wanted to add movement to the paddle to make it, so it looks the paddle has been hit like in the server code. This would have been done in the on receive where the sound is played. This would also make the paddle rotate slightly when it was hit.

Another part I would have liked to add to this game was a pause menu as the serve has one, but I could not figure out how to do this in this project. As it does not seem like there is data being sent from the server, more of a lack of data being sent. As the server seemed to just stop sending data to the client, I could not figure out how to add a pause menu in to this game. Also sending data to the server I could not figure out as Java is not my strongest language. I think to add a pause menu to the client I would have to send a message to the client from the server and get it to trigger the pause menu when the button is hit.

The lack of data being sent to the client made another issue when testing this game as when I had loaded the game up, as it will not update the scores until the ball hits on the walls. This is because the game data of scores are not sent until then. If I was to change it, I would add code to the server to check when a client has connected to the local port and sent that data then, so that when the client boots up the scores are sent and update when the ball hits the wall. This would be the same for the rest of the game data, as another bug is when the server is paused no data is sent so if you boot the client up when the server is paused all the positions are set to 0. If I was to edit this, I would do the same as the score system. When a client connects to the port send over the last data that the server had.

# Conclusion

Overall, this project has improved my skills in editing the server with a client and developing it to create a new version of the game. As I found the main issue was time, I would improve this further by adding more images and sounds, alongside a pause screen in order to develop the project further. Despite this, I feel as though I have created a unique version of the game, with implementing my own features in order to give the game a modern, fun twist relevant to the current times.

### Estimated Grade

* Demo B+
* Idea A-
* Report A-
* Total B+