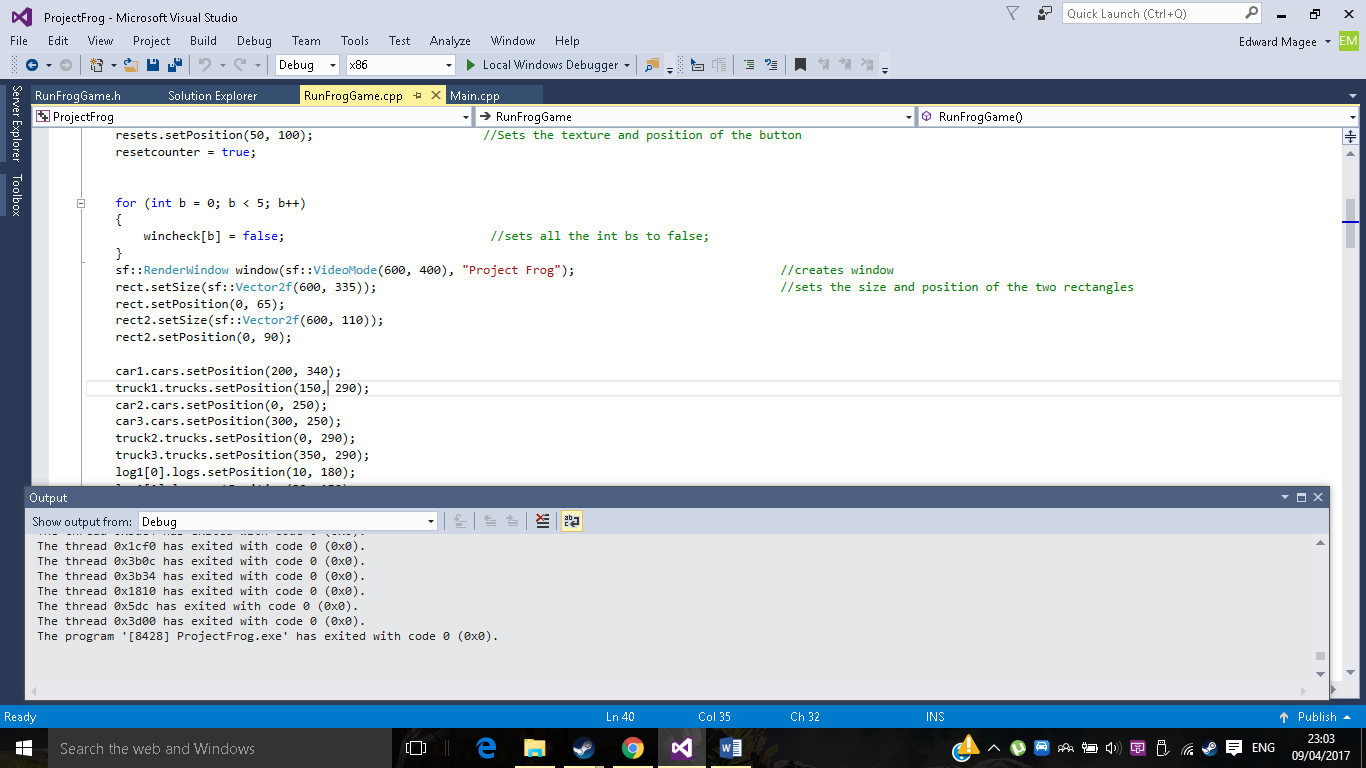
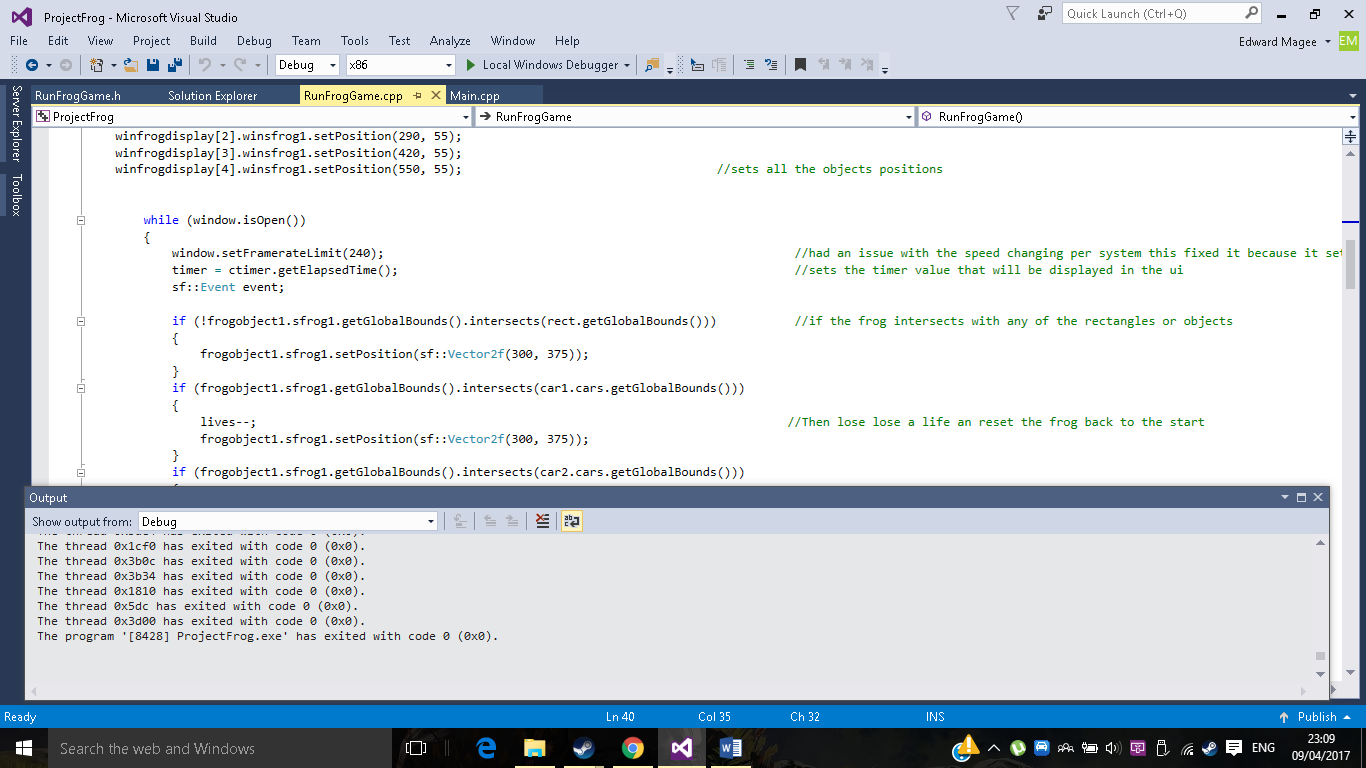
Project Frog (C++, SFML)

My aim is to create a Frogger style arcade game that will allow the player to take control of a frog trying to cross the street, there will be hazards such as water and traffic as well as logs to help aid the frog in crossing pass the water. When the frog reaches the end of the stage they will be reset back to the start, if they manage to collect all the docks they will be taken to the after menu where they can view their score and time this screen will also be displayed if they run out of lives.

Implementation

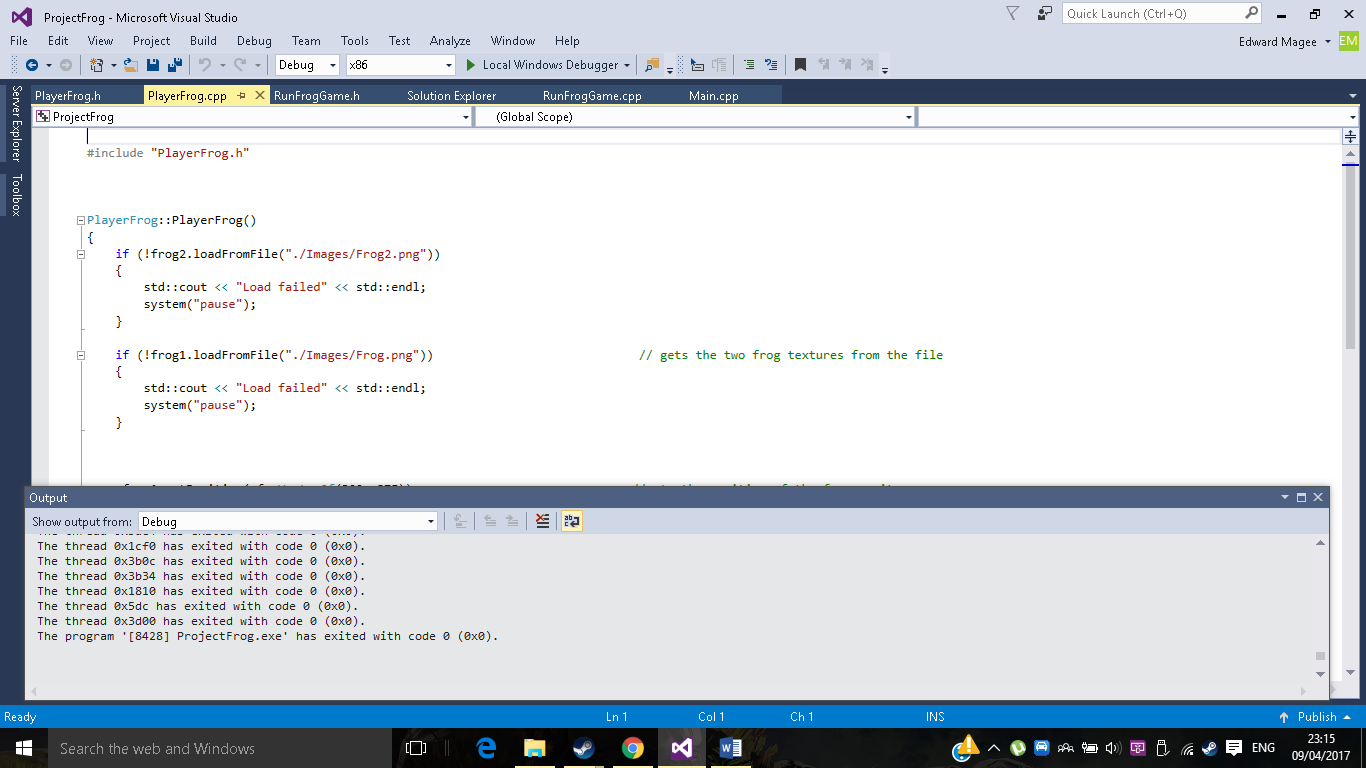
I firstly created a window that would become the game using the statement below.



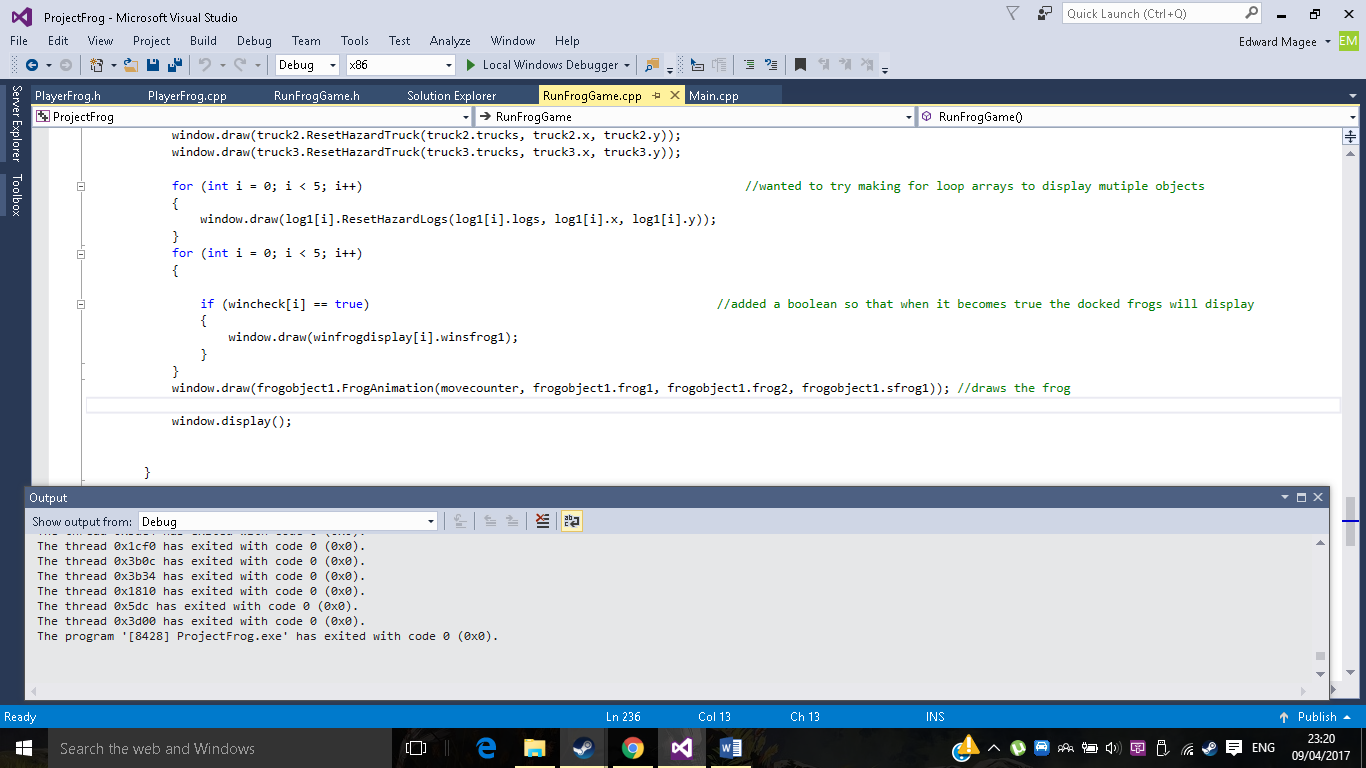
What this line is doing is creating a window called window and then it sets the x to 600 and the y to 400, then it finally gives it some text to display “Project Frog”.

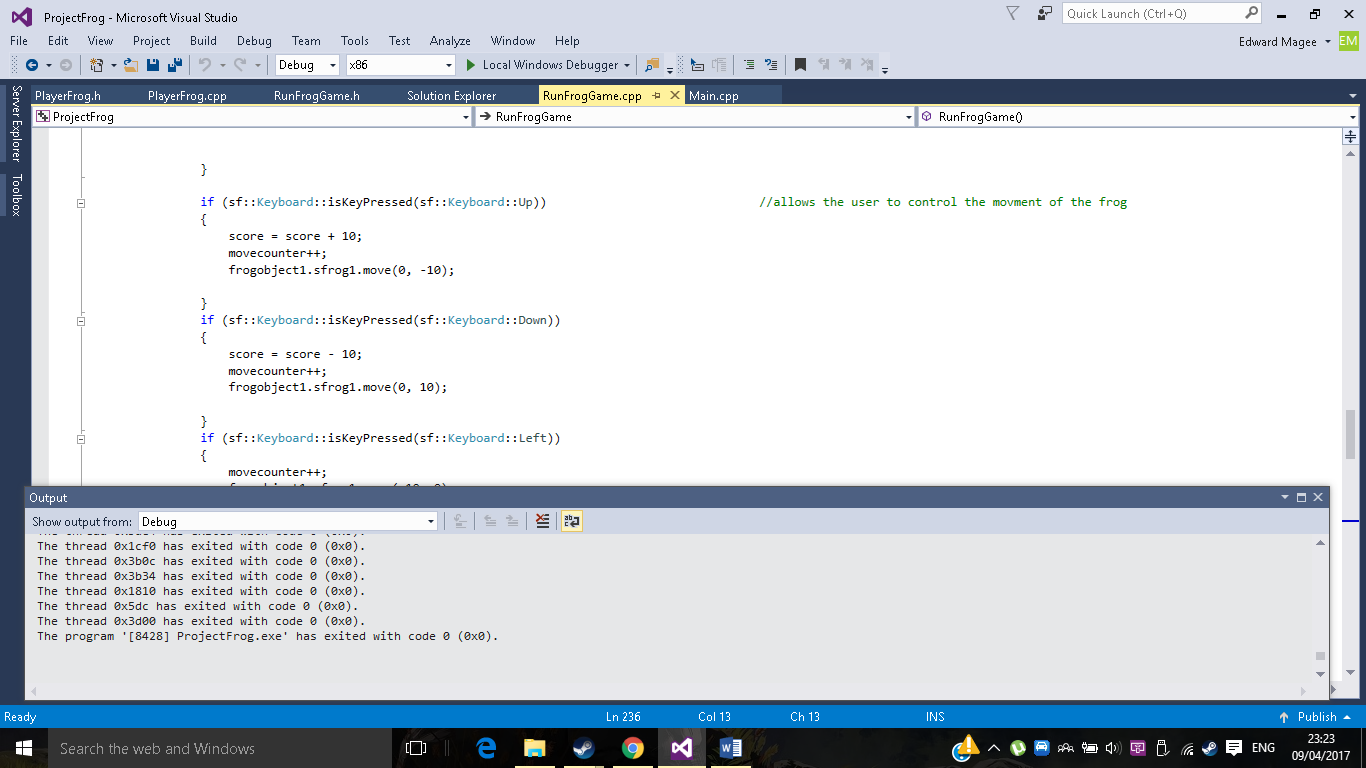
I then use this while loop to display the window that I had created before if the window is not open it becomes false and stops the loop.

I also used a poll event to allow the user to exit the window by clicking the x button.

The next thing I decided to do was create a class called PlayerFrog which will create a frog sprite for the game, first I got some graphic assets online and made them into png files.

Here the code is loading an image from the file and setting it to the texture if its false it will tell me load failed in the console.

Once the texture had the png file I could then set it to the sprite and then display that sprite using the below code, I had to create an object for it however and call the function and the parameters I used later to create a simple animation.

So I was able to display the frog but I now needed to get it moving what I did to do this was use move.

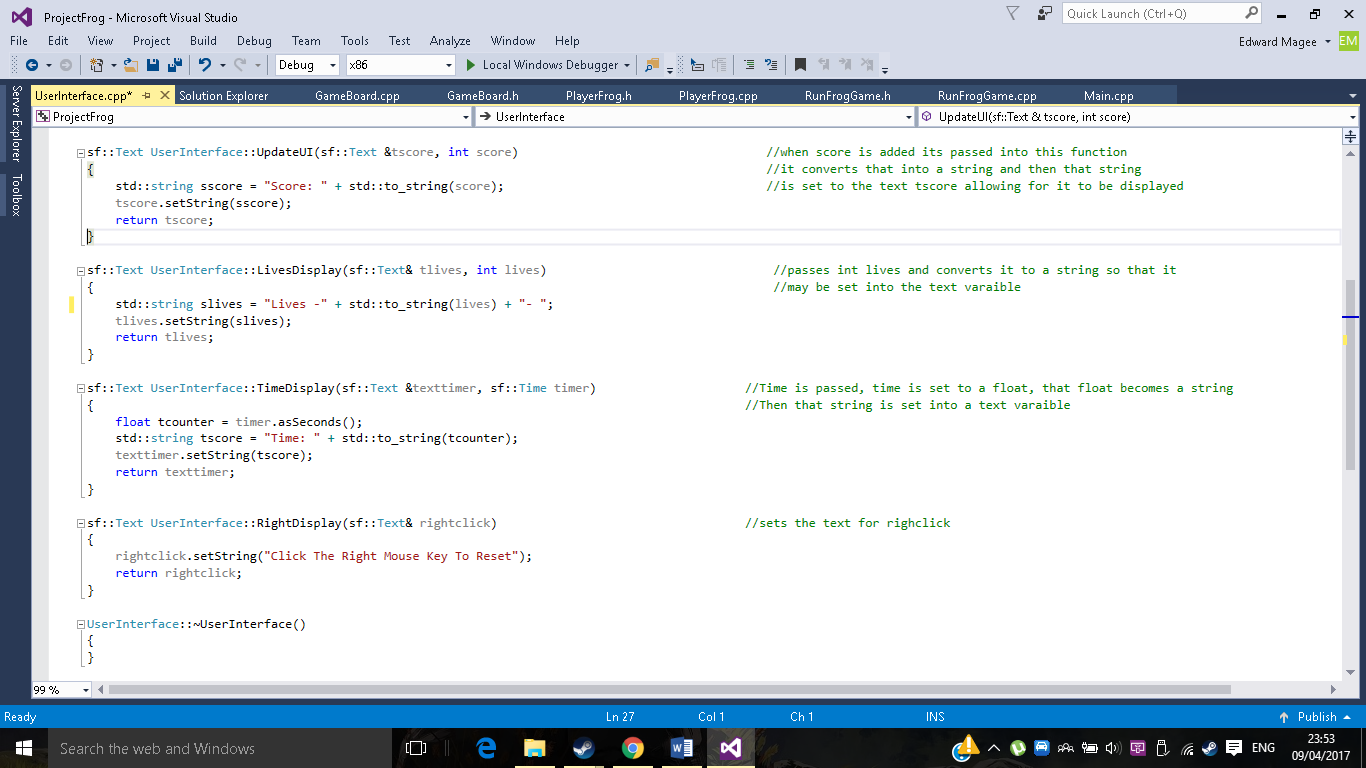
Here I used sf::keyboard when the up key is pressed it will do the following things such as add 10 to the score and move the frog -10 in the y axis.

I created three more for left, right and down but with down losing taking away 10 instead of adding 10 like up does.

So, by this point I had a frog moving around on a black screen, next I wanted to display the board so I created a class that contained it called GameBoard.

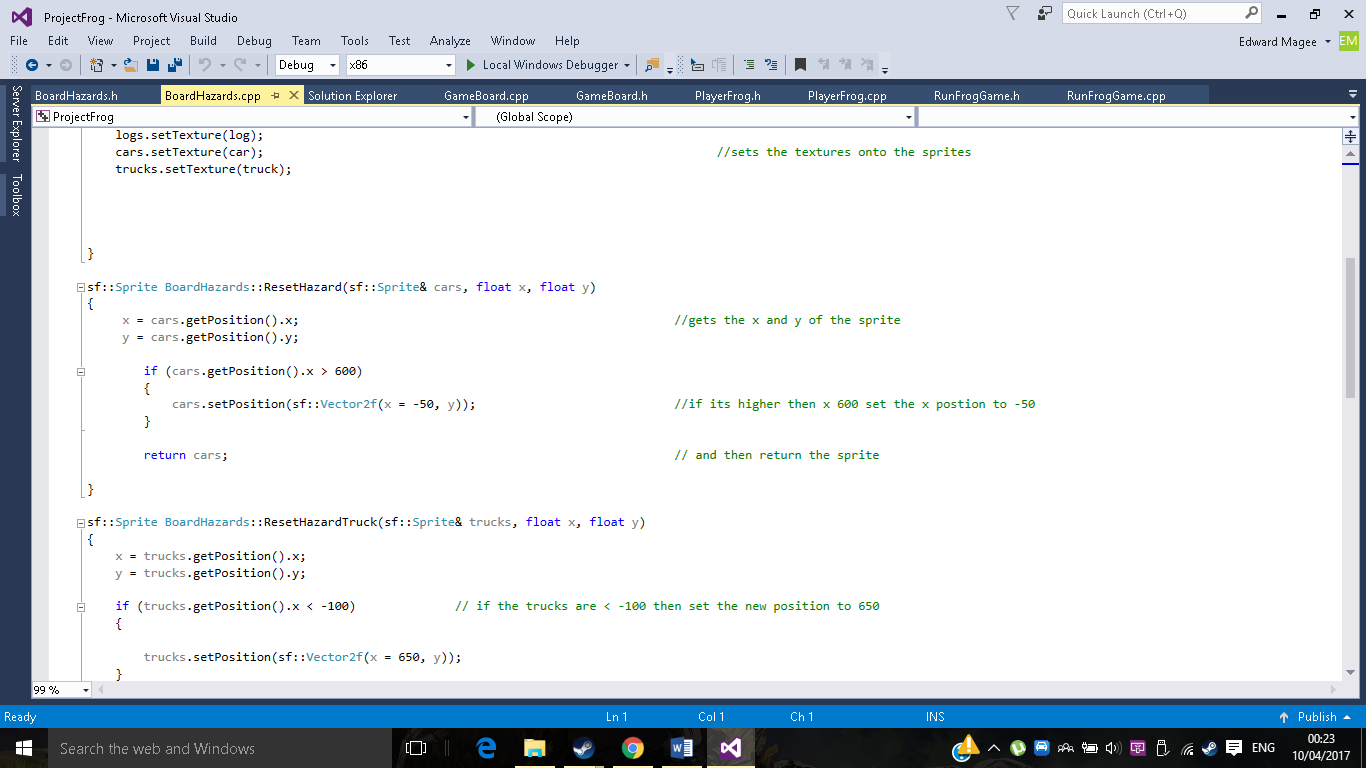
GameBoard is a class that pretty much just sets a texture to a sprite and gives it some position and scaling information.

When that was done, I had a frog and a map next I decided to create a user interface class that would display three things the time, score and lives.



Here is the function I used to convert an int into something displayable inside the window, first I passed it the int score and set it to a string, I then added it to another string called sscore and finally set tscore with sscore. For the others, I did something similar like timer I set to a float and then to a string. Once I set the position and used draw they were working fine on the window.

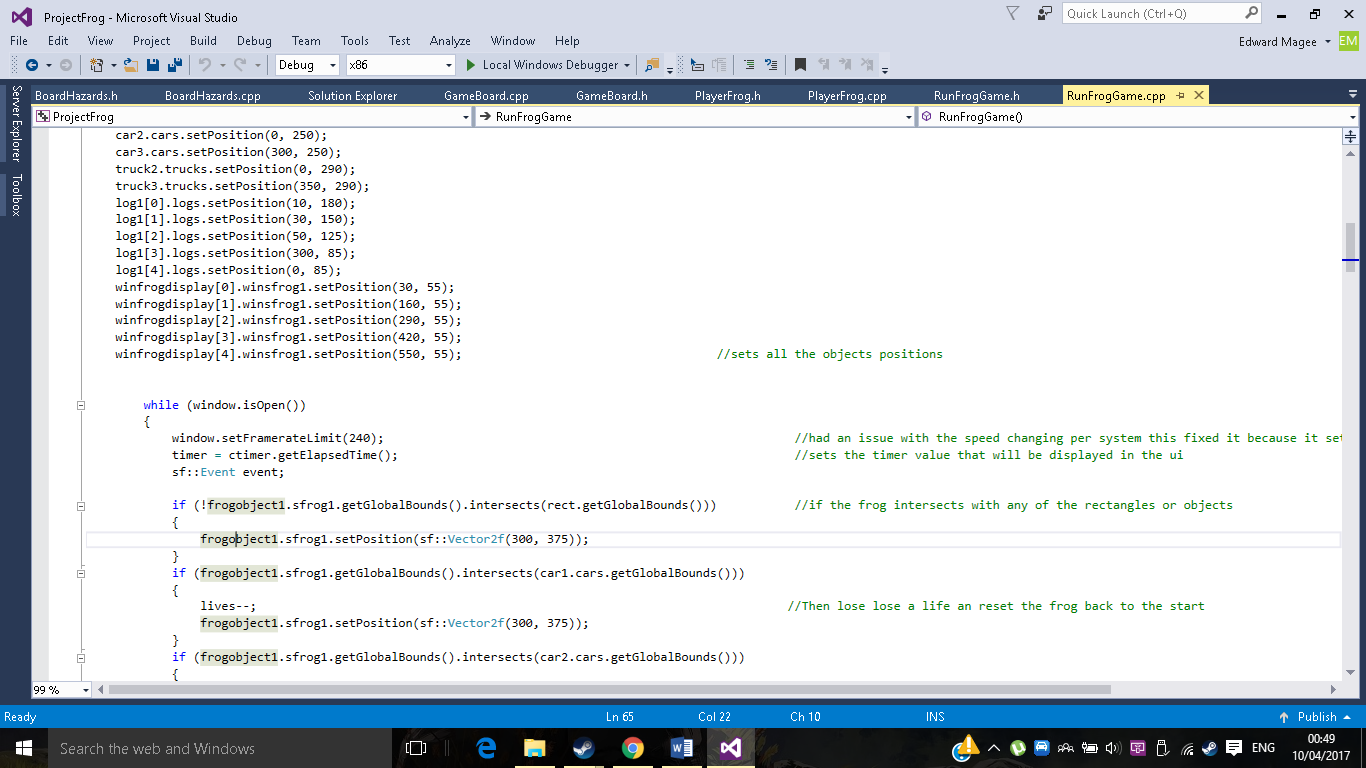
Then I created a hazards class were the vehicles and the logs were created, after displaying them I had to get them moving at first I tried time x speed == distance but it became too fast, so I tried using the move like the frog and this seemed to work, in till I used it at game labs and the same move error happened again. I believe this was because of me not setting the fps once I added that if they ran about the same on both systems. Another issue I had whilst getting the move working was a truncation warning because it was being converted from a double to a float which resulted in loss of data, it did not remove the error but became annoying, adding (0.6f, 0.0f) f to the end of it made the warning disappear.

With movable objects created I had to make sure they reset to the other side, I did this by creating a function which looks something like this.

This function gets the position of the sprite and sets it to a float called x and y.

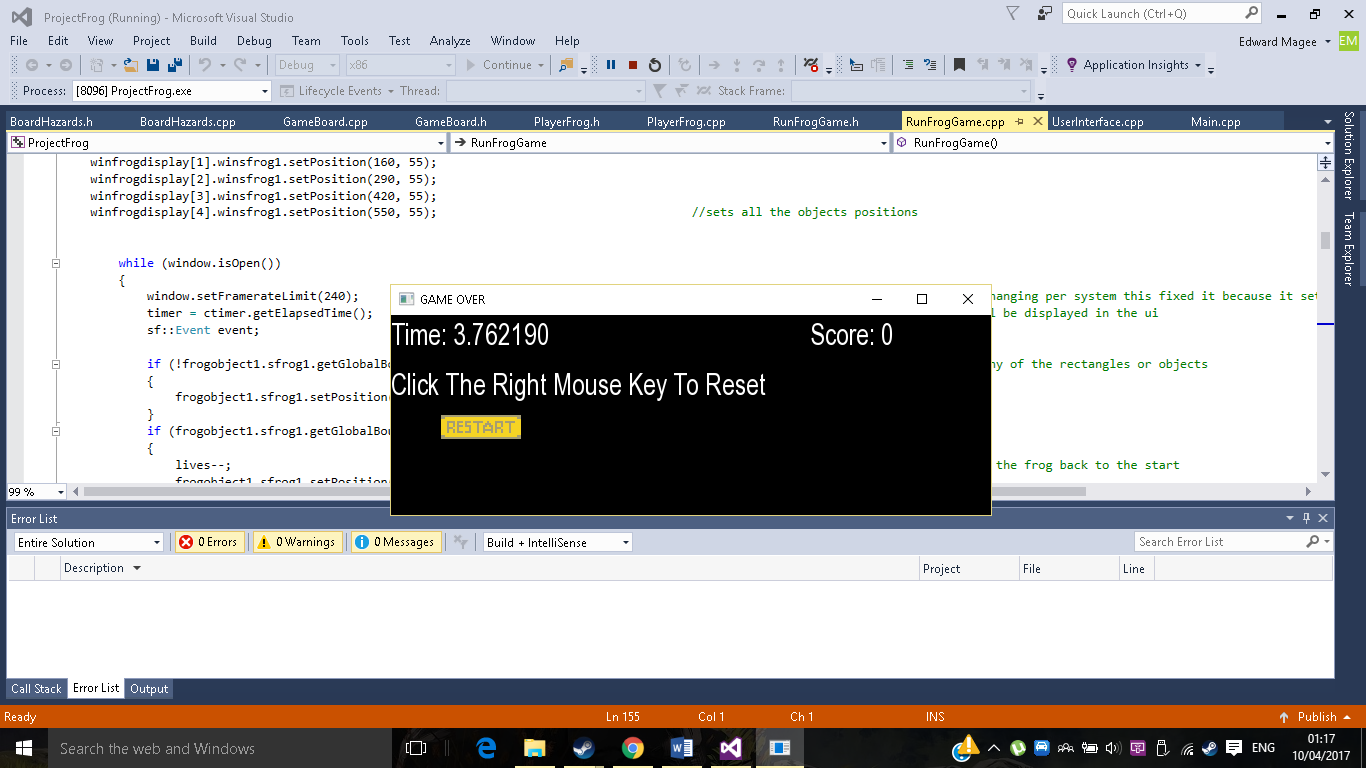
If the x value is higher than 600 then set the position to -50 and return cars and it will do it for every car that is created allowing it to be reset in the game.

Now there is a steady stream of traffic and logs moving but they’re not resetting the frog back to where the player starts.



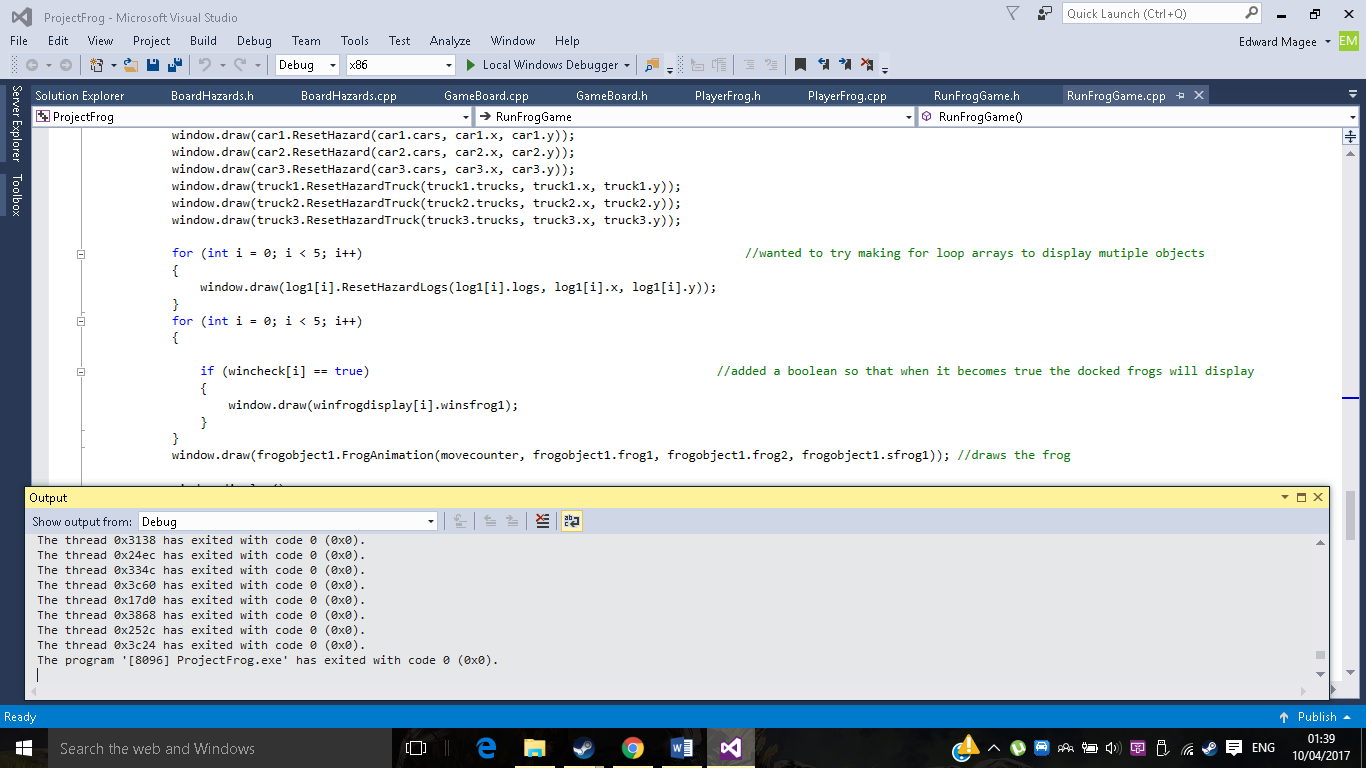
Here I created a rectangle called rect and used getGlobalBounds as well as intersect, when the frog is not in that rect it gets reset back to the start. I did something similar with the rest of them, when they intersect the global bounds they will be reset back to the starting position. The water however was a tad different I used a Boolean to check whether it was on a log if it was not it would see this as true and reset the frog back.

When that was finished the core of the game was done I had a frog that could move around the map, lose lives and get reset back, what annoyed me was that I could spam up and it did not matter whether the player lost lives. I decided to add a second window called window 2.

I did something like the first window but added a Boolean to allow the player to go back and between the two windows, it ended up looking like this

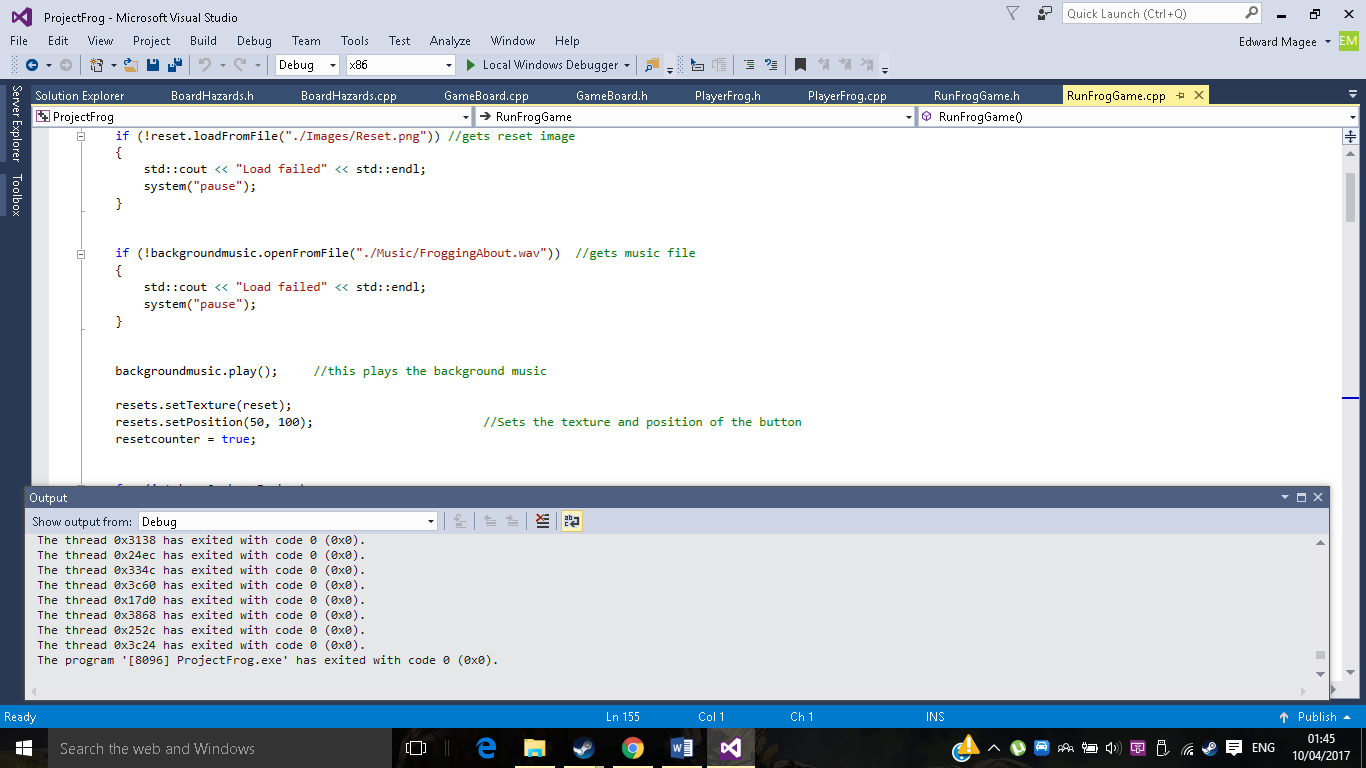
Here is the second window it displays the final score and how much play time the user put in and a reset option if they would like to take another go at the game.

I then added a small animation for the frog allowing it to switch between two textures by using a modulus every time it gets added by 1 it becomes either odd or even it will detect which one and switch accordingly.

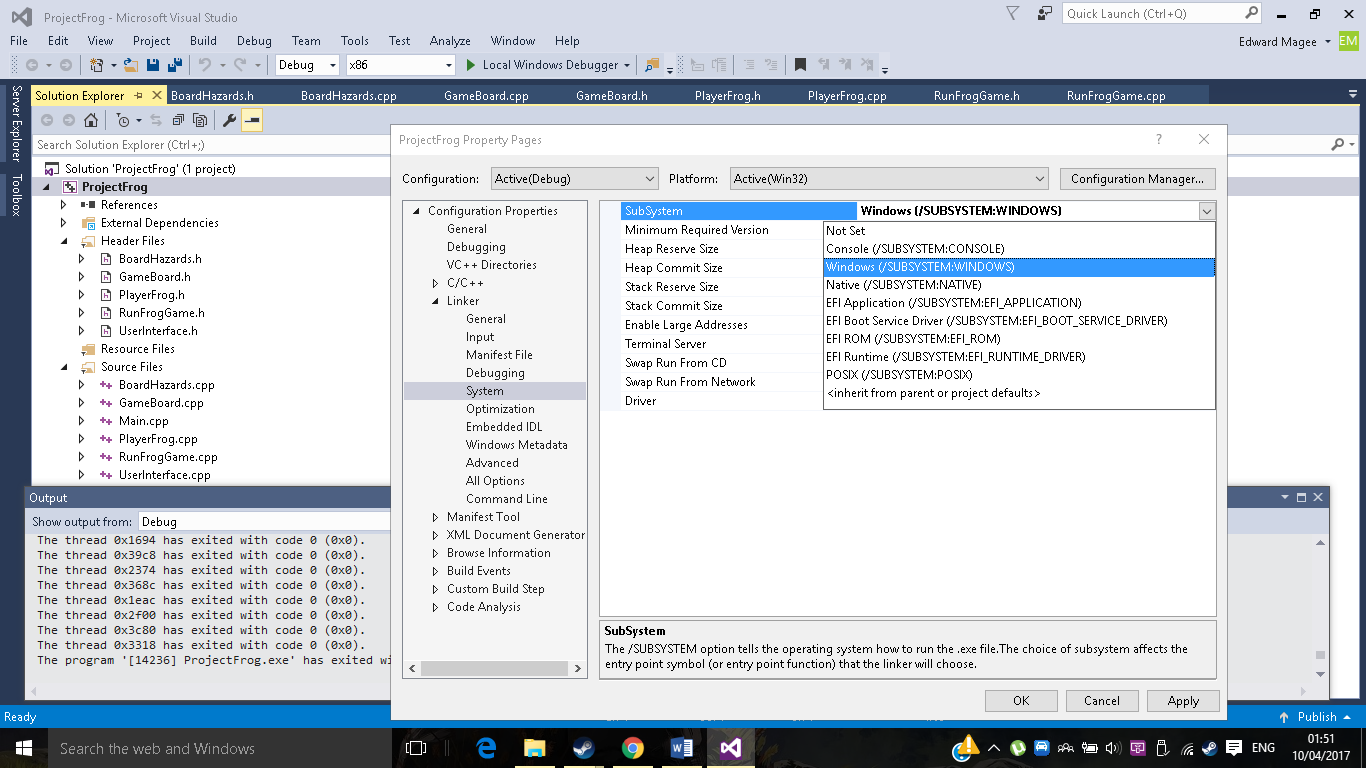
I wanted to add a win state for the player to get to so I created five docks for the user to get too, when the player arrives a frog will appear showing that they have docked there and will gain lives and score.

Here I decided to create a for loop that checks a Boolean array, it will check all 5 of them and if one becomes true it will display that frog in the position where the player had docked.

Now pretty much everything was done for the game the frog rest the cars and logs had collision, movement and resets but it felt to silent so I used the line.

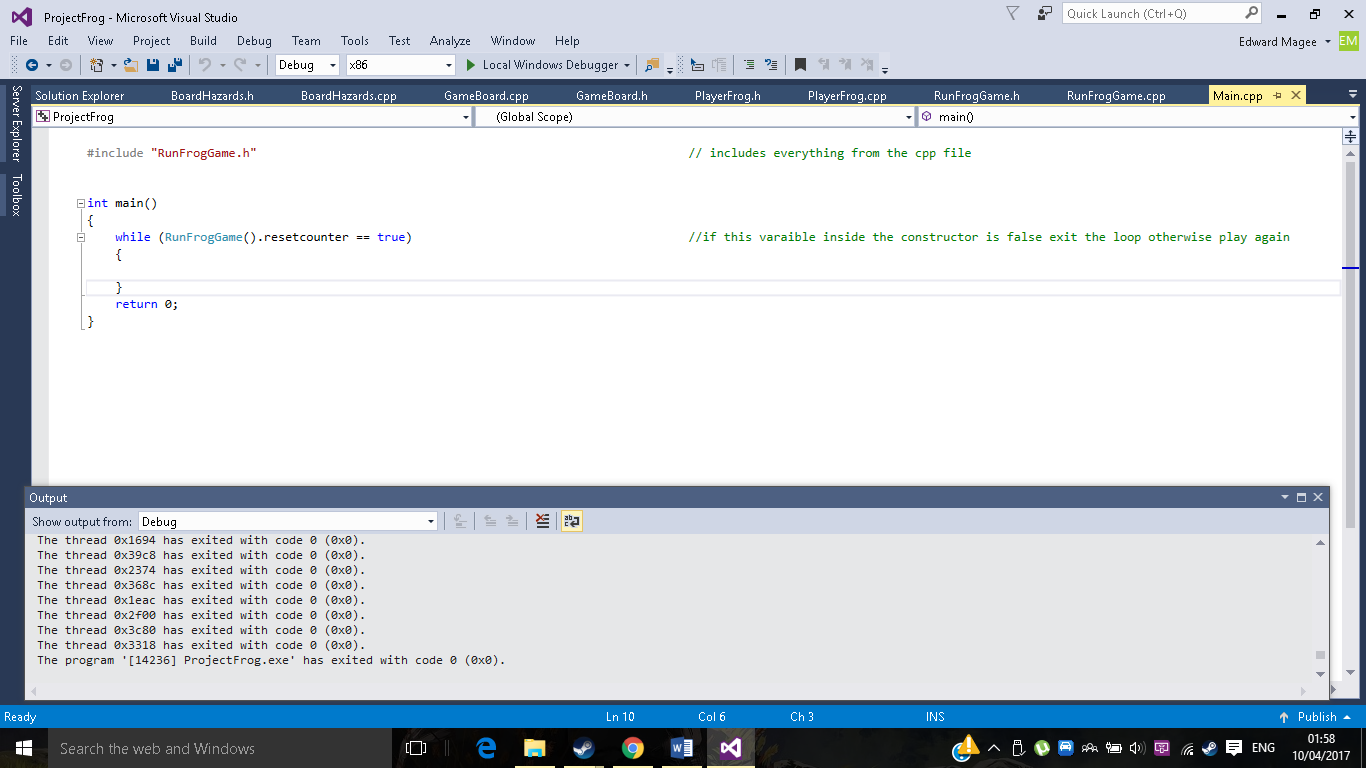


To add some music to the game, I created an sf music and set a wav file to it and when the game loads it plays the song that I set to it.

Final thing that was left was to remove the console, the user does not need that there when they are trying to play the game.

By setting the subsystem to windows it won’t display the console and only the game windows that the user will interact with.

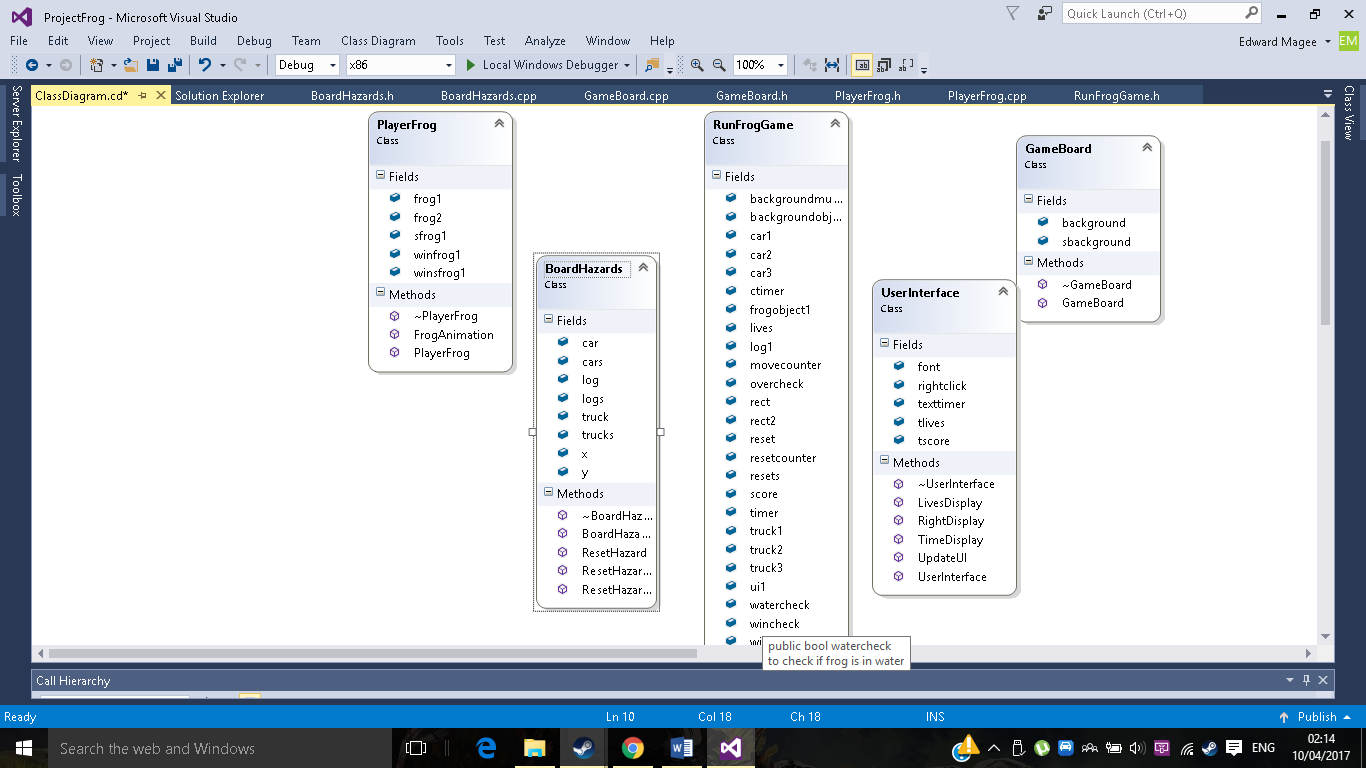
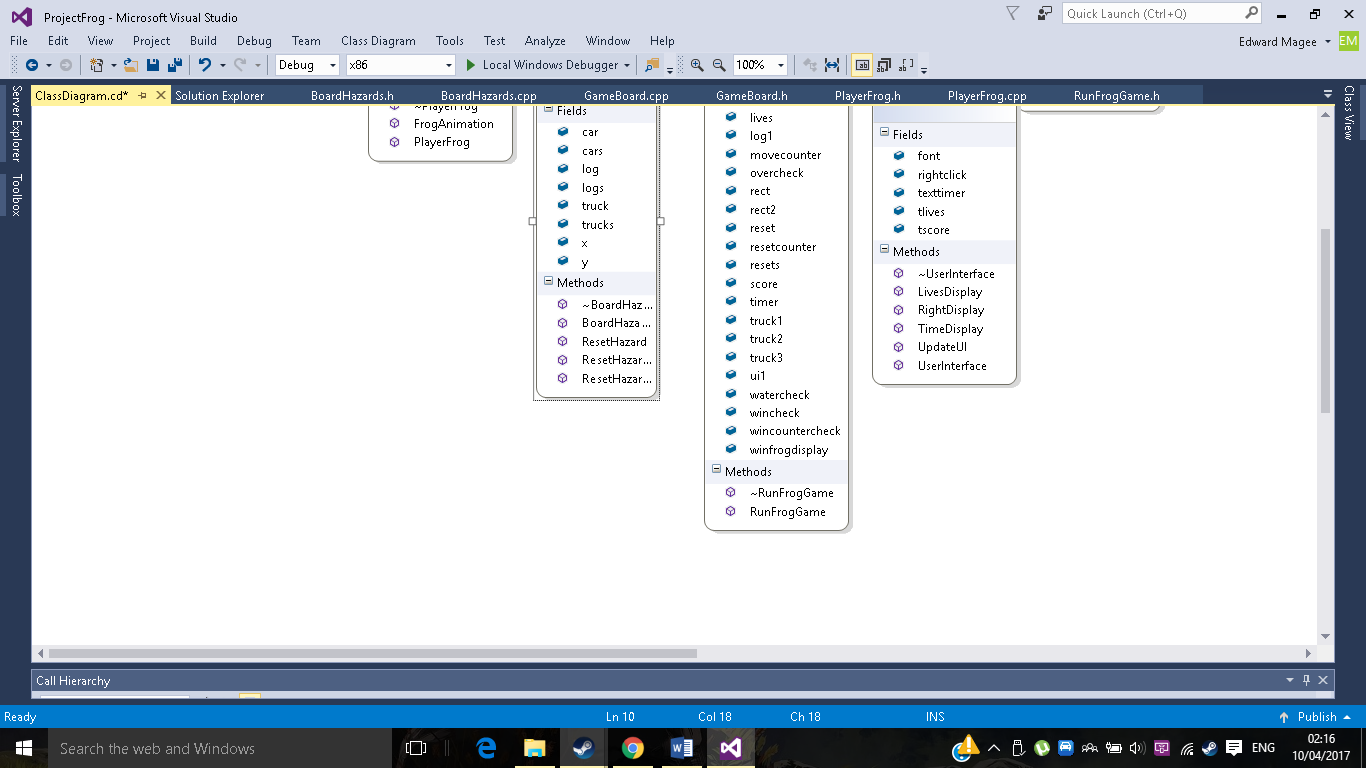
My Class structure

Originally everything was in the main, now it looks something like this.

This is all that is in the main, the RunFrogGame() is running the constructor that is the class RunFrogGame all the code that was in the main before is now in there. If it == false it will stop looping.

By making a new class and throwing all the code it their it now makes the main very tidy and it hardly has any code in it compared to before.

My User Interface class was originally gonna do more than it did before however it ended becoming a place to store all the texts I wanted to display and it made functions that turned ints into text and displayed them.



Here is the entire class structure all the other classes give data to the main class which runs the game.

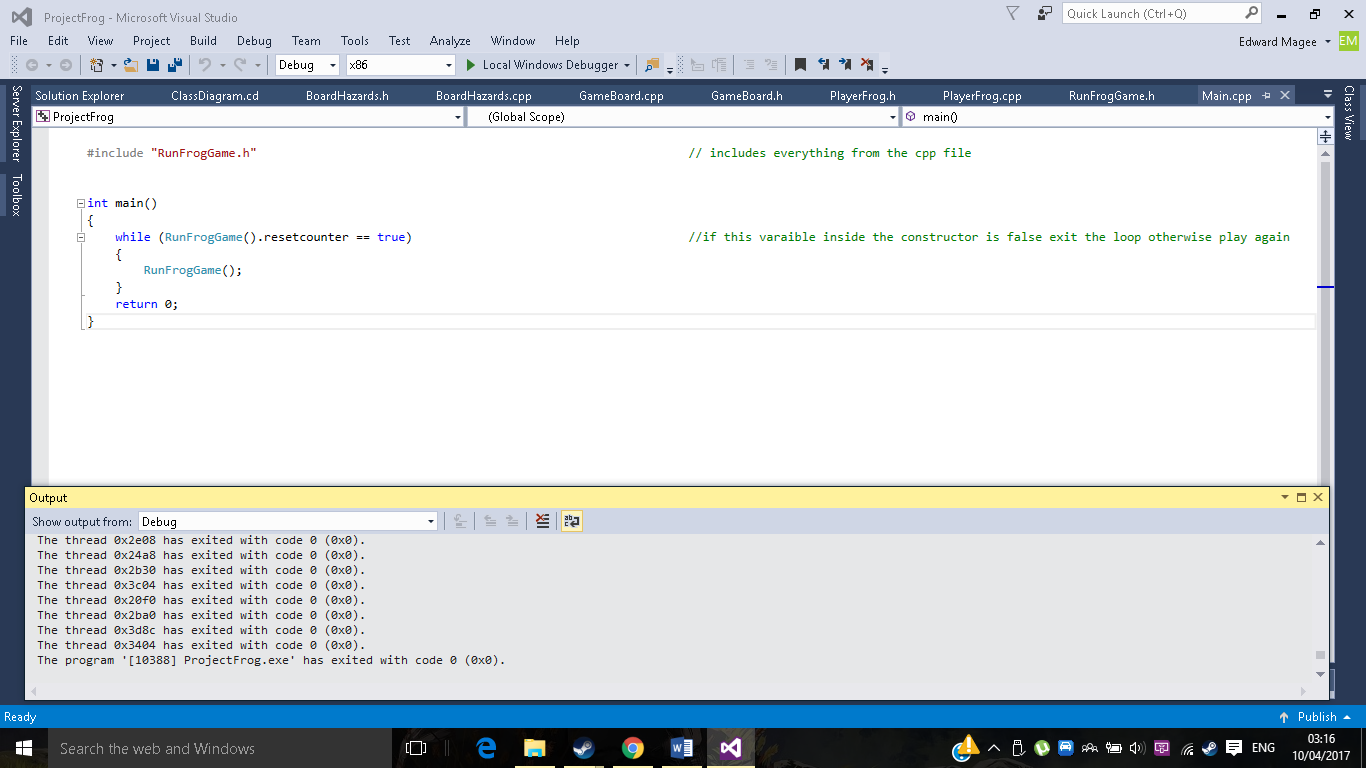
Testing

|  |  |  |  |
| --- | --- | --- | --- |
| Test | Expected Result | Actual Result | Comments if failed |
| Frog Movement | The frog should be able to move around | The frog successfully moved around | N/A |
| Car movement | The cars should be able to move along the x axis. | The cars successfully moved. | N/A |
| Truck Movement | The trucks should be able to move along the x axis. | The trucks successfully moved. | N/A |
| Log Movement | The logs should be able to move along the x axis. | The trucks successfully moved. | N/A |
| Water reset | When the frog hits the water, they should get reset. | Frog has been reset to original position. | N/A |
| Cars reset | When the frog hits the car, they should respawn with -1 health. | They lose 1 HP and are reset. | N/A |
| Truck reset | When the frog hits the truck, they should respawn with -1 health. | They lose 1 HP and are reset | N/A |
| Log safety | The player should be safe on the log | The player does not get reset on the log | N/A |
| Out of bounds | When the frog goes outside the map they should be reset | The frog does go back the starting position when out of bounds | N/A |
| Score | Score should increase and display | The score does increase and display on the window. | N/A |
| Lives | lives decrease when the frog collides with certain objects | Yes, the lives do decrease when the frog collides with certain objects. | N/A |
| Time | The time should go up and display | The time does go up and display | N/A |
| Car Reset | The car should go back to the other side | The car does reset along the road | N/A |
| Truck Reset | The car should go back to the other side | The car does reset along the road | N/A |
| Log Reset | The car should go back to the other side | The car does reset along the river | N/A |
| Docked frog | A frog should appear and reset the frog back | The frog does appear and sets the frog back | N/A |
| Moving with the logs | The frog should move with the logs | The frog does move with the logs | N/A |
| Window 2 | Window 2 should open when the criteria is met | Window 2 does open when the criteria is met | N/A |
| Window 2 Reset | When the user right clicks, it should open window 1 | When the user right clicks, it opens one | N/A |
| Window 2 close | The game should shut down when the x is clicked | Window 1 loads up | There is an error with the x button. |
| Window 1 score, time reset | When window 1 is re loaded the score and time should reset | The score and time do indeed reset | N/A |

Testing Error

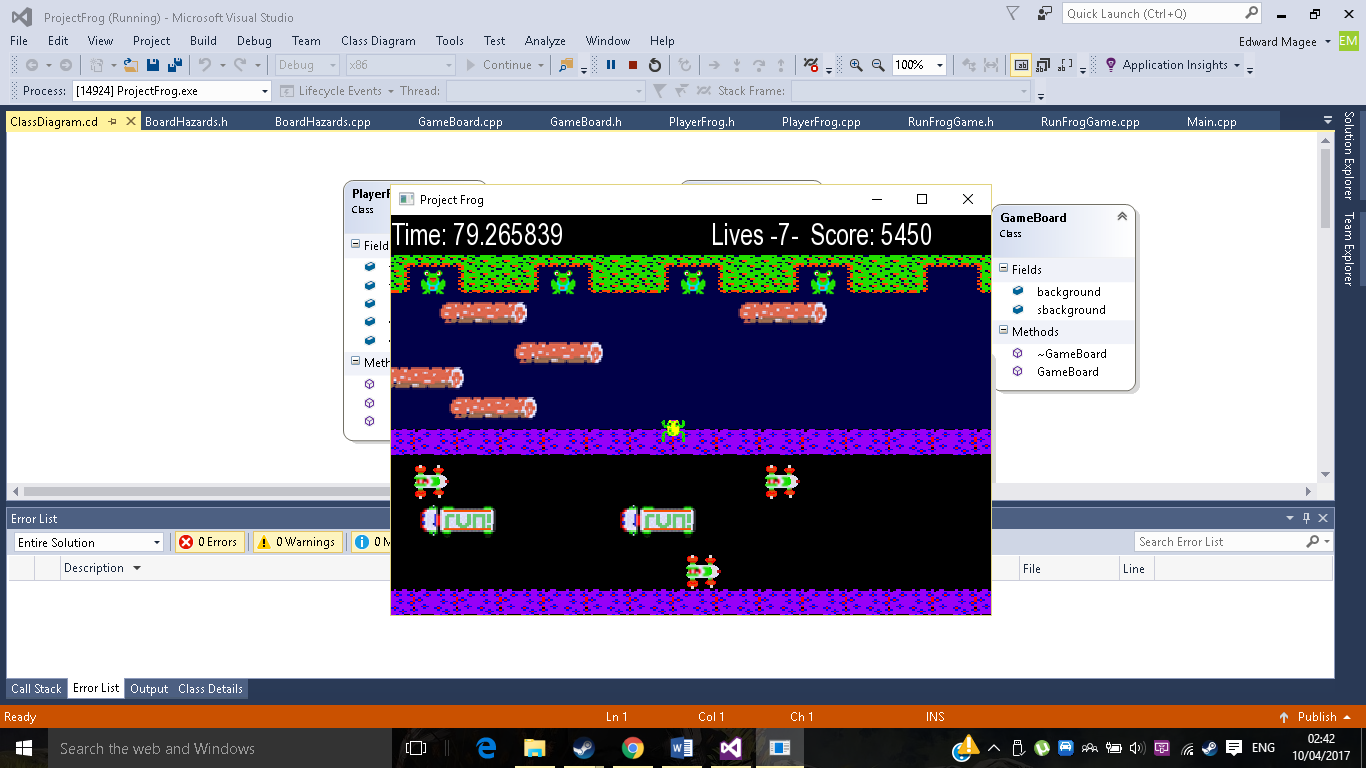
Trying to fix this error took me a while to realise what actually went wrong. Basically, when you play the game and then in window 2 click x it closes no problem. Do this after replaying it twice using the right click, it will replay the game. I thought it might have been an issue with the Booleans but if that was it, it would be rather odd it worked the first time and not the second.

I eventually found the issue:



It was running it twice, I thought while (RunFrogGame().resetcounter == true) was just checking if the reset counter was true and then doing the line underneath. It was doing this but it also ran the game a second time during the while loop which messed with the windows. After removing it worked fine. Now im left with empty brackets because it won’t loop without them.

Conclusion

Inconclusion I am quite happy with the outcome, it contains many of the desired features and the user can enjoy the game now. Although there are things I would still like to add, such things like adding a working reset button, when you click right anywhere t works but clicking left would create an issue with the x in the window edge, if I had more time I would have fixed this and got that button finished. Another thing I would have liked to add is multiple levels instead of getting a game over for collecting new frogs I wish there was new maps for the player to traverse, they could have included new hazards and bonuses.

Final Result: