BSc in Software Development

Year 3

COMP07030 Software Design Project

Platform Game

G00311853

*Darren Fitzpatrick*

Contents

[Introduction 3](#_Toc447884727)

[Architecture of the solution 4](#_Toc447884728)

[Reasons I designed my project this way 5](#_Toc447884729)

[Technologies used 6](#_Toc447884730)

[Problems Encountered/Solved 7](#_Toc447884731)

[Conclusions 8](#_Toc447884732)

[Recommendations 8](#_Toc447884733)

Student Number: G00311853

Student Name: Darren Fitzpatrick

Supervisor: Gerard Harrison

GitHub Link: https://github.com/DarrenFitz/DarrenProject2016

# Introduction

For my final year project I developed a platform game. It is a user interactive game that relies on the arrow and jump keys to navigate around the screen. The player spawns in the game and has the challenge of reaching the door at the end of each level. There are multiple obstacles such as enemies, walls, lava, and bullets that stand in your way. The game has coins you collect along the way, which total up throughout the game.

This game was created multiple sprites, scripts, objects, and room. Code was done in the object and script files in order to create command and actions such as move, jump, and collide with other objects. This was all done using the “Game Maker Language”.

**Aims**

* Create a fun, simple, and appealing interactive game that provides the user with a challenge.

**Objectives**

* Become familiar with how to use Game Maker Studio.
* Learn as much as I can about Game Maker Language.
* Try and utilise my current programming knowledge and apply it to this project.
* Come away with a better understanding of what goes into developing games.

**Scope of Work Done**

* Design an overall visually pleasing game.
* Include as many feature as I can in the game.
* Add character and enemy animations.
* Add a score system for additional user enjoyment.
* Ensure the game is navigational (e.g. A Menu)

# Architecture of the solution

The structure involved in creating a GameMaker game includes multiple folder with different items in each. For my game I used the following folders Sprites, Sounds, Backgrounds, Scripts, Fonts, Objects, and Rooms.

**Sprites** – These are generally visual representations of objects within the game you are creating. A sprite is either a single image or a set of image that when played one after another to look like an animation. This is what I used to make my player and enemies appear to be moving.

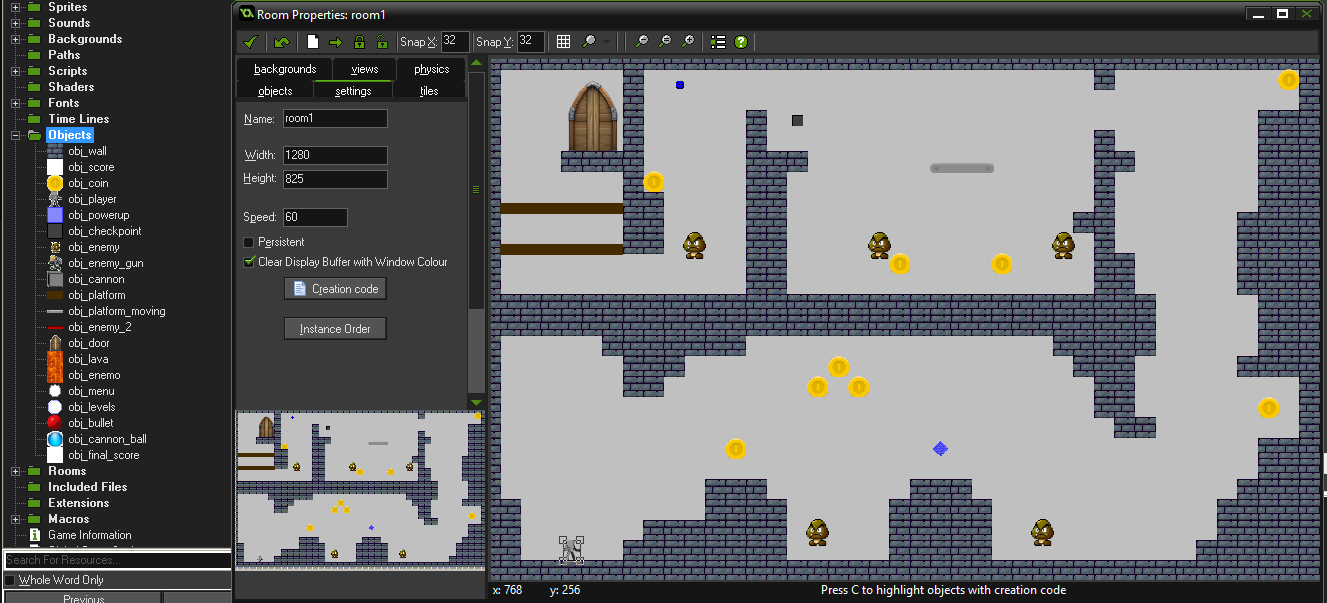
**Sounds** – These are sounds as either \*.wav or \*.mp3 that can be called to be used at any point in the game. I used a coin sound every time my player collided/collected a coin and ding sound every time I killed a Goomba.

**Backgrounds** – A large static image that can be used to make the page/level more engaging. I used a background in the total score page. And for the menu.

**Scripts** – A script is a place where you can write code for a specific action and call it later. I used a script for the player’s death, the menu, and the level selection.

**Fonts** – It is a particular font/style of text that can created to the user’s specification and called later to be used anywhere in the game where text is required. I used this in my final score page and used the font “Jokerman”.

**Objects** – In a game made with GameMaker Studio, the characters, monsters, balls, etc... are all instances copied from a basic object template that is found in the resource tree and then placed in a room. So when we talk about something affecting or changing an instance, we mean that one particular copy of an object in a room is being affected while all the rest are not, but when we talk about affecting or changing an object we mean that anything we do to it will be reflected in all the instances created from that point on too. So, the object is the template for the instance, and the instance is what we place in a room to make our game.



**Rooms** – Rooms in GameMaker Studio are where everything happens in your game and you must have at least one room in any game for it to run, but in general you will need many more. I used multiple rooms, one for each level and then also one each to display the menu, levels, and final score. Also a room to initialise global variables before game begins.

# Reasons why I designed my project this way

When it comes to making a game, design is one of the most important parts. You must take in consideration that it is going to be the selling point of the game. Features are important but if the design is poor the game will not succeed. There were many aspects that that went into my design:

* Animation
* Coin/Checkpoint Layout
* Manoeuvrability
* Efficiency
* Practicality

With the animation, I designed the game to have a couple of enemies and the main player move as if they weren’t just an image. The main player moved according to how to the user pressed the arrow keys. I made it look realistic when it jumped and when it fell. The enemies marched and looked in the correct direction as they moved.

The coin/checkpoint layout was also very important. The coins are an interactive feature and therefore they must be placed in a location that required the players to have a little difficulty to reach or collect. If the coins are too easy the game is not interesting and not as replayable. This way it makes it a challenge for you to beat your previous high score. With checkpoints they need to be placed after the player made decent progress but not too far into the game where it will be unrealistic to reach. I generally placed them before or after a difficult obstacle.

Manoeuvrability relates to having a path to get to the end of the level. Insuring that finishing the level actually possible. Taking care that that nothing will block or impede the task at hand. The only way to make sure this happens is to complete the level myself.

Efficiency was achieved by using the objects and having multiple instances of that object in each room. There is no need to create multiple objects. One object is used but is has multiple instances of it active throughout the game. For example there are multiple goombas in this game throughout each level, but each all of them are just instances of that one object, which saves a lot of time and is very efficient.

For practicality reasons I added a select level option in the menu of the game. This enabled me to pick any of the levels in the game and spawn there without having to complete to the levels before it. This feature also be accessed to change levels mid game by pressing “m” for the menu. This design feature is extremely useful for testing too.

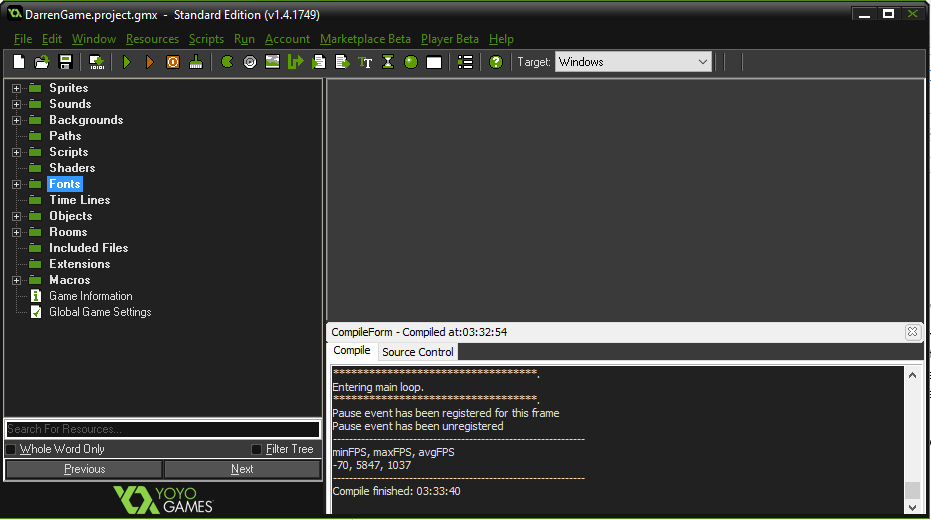
# 

# Technologies used

For this project I used to a new program and a new programming language that I have never used before called:

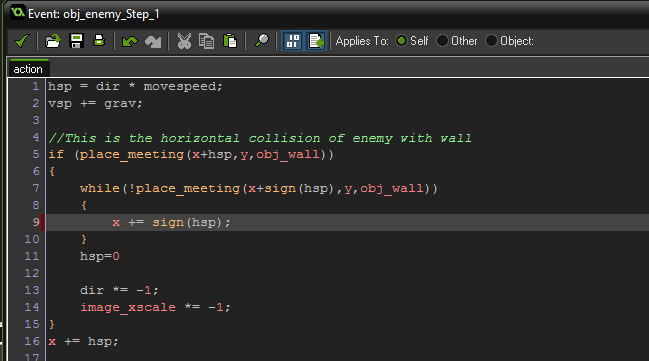
* GameMaker Studio 1.4
* GML (GameMaker Language)

I chose GameMaker because it was a program that I was not very familiar with and thought it would be a great opportunity to learn something new. I was not familiar with the layout of the program and it took some time to get use to, but was fairly simple once I did. GameMaker studio allowed me to create anything I imagined relatively easy. It was very suitable for what I needed. It utilised sprites and objects which were necessary for my game. This program has a very good website that seemed to help explain and show effectively how to use the programme. All the code functions were on their website or if you pressed f1 a list of them appeared. I used this site when looking for any function that I need to know.



The straightforward, comprehensible layout made it easy to use and organise the objects as I was developing the game. I found that this helped me save time in the long run.

The programming language I used was also completely new to me. It was called GameMaker Language. It wasn’t as hard to learn and was really quite intuitive. Having already been familiar with programming in other languages allow me to understand the concept very quickly. The website was also very easy to consult and get the answer I was looking for in seconds. GameMaker Studio also had a sort of intelliSense that was very practical when I was learning the language and also to help speed up the typing process.



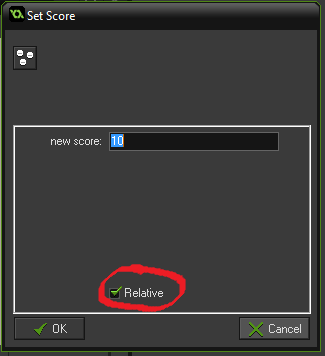
As you can see from the picture the code is relatively straight forward. There were a lot of websites that could aid me if I needed help, this is another reason I chose this program. Getting stuck was not too bad due to information about GameMaker widely available online.

For this project I could have used the auto generated event action buttons but decide to stick to coding for the most part, as it was more challenging and allowed me more freedom when it came to giving the player and objects commands.

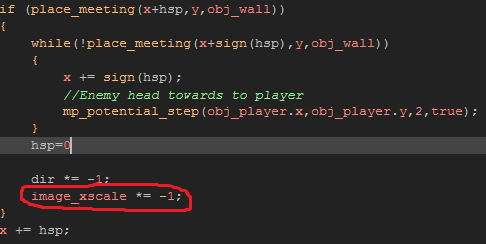
# Problems Encountered/Solved

One of the problems I encountered when I was developing the game was that when my player collected the coins, as he navigated through the level, if he died and respawned at the check point the coins would respawn too. This was a big game fault because this meant the player could take advantage and keep dying and respawning to collect unlimited number of coins. This was happening because I had the checkpoint to spawn and restart the whole level when the player died. What I had to fix it was to only respawn the player at that checkpoint as opposed to restarting the game.

The next problem I encountered was I couldn’t get my score to continue as I went into each level. The score I gained from collecting the coins was erased either leaving the room or if I died when playing that level. It meant that I couldn’t have a final score at the end of the game to display. What I had to do to fix it was scrap the idea of doing this with code, because I couldn’t find a way to make the score relative. I had to use the predefined event action to do this because then I could set the score value as relative, which meant the score would continue through each room.



Another little problem that I had, that was actually relatively easy to fix, was when I used the goombas animation to march around the level. The idea was that the animation would make them look like they were walking back and forth on the screen. The problem is the animation was playing in a forwards motion when the character was supposed to be walking in the other direction. To fix this, I made it so that when the goomba hit the wall the images/sprites would flip. The command “image\_xscale” flips the image. Its default value is 1. So when the goomba collides with a wall multiple “image\_scale” by -1 to flip it.



# Conclusions

This Project was an overall good experience. I felt I learned a lot about an area I have always wanted to delve into. I feel more confident that I would be able to a make more complicated and advanced game app with similar software now. I never realised how much or what went into a simple game like this. It has taught me a lot about animation, collisions between objects and how frames apply to a games.

The project was at times demanding and stressful, but at the same time manageable. It required my to meet a deadline and this was where I learned the most. I had to plan and stay ahead of schedule week by week. A lot of the time went into choosing the project, what software to use, what the game would do and how it would look. This help me improve on my time management skills and general organisation skills. Having enough time to figure out everything out was a crucial element to this project and it was a valuable learning process.

Working on a new programme with unfamiliar code on my own has forced me to figure things out independently. It without doubt helped me to improve on many areas like my problem solving ability, and my intuitive thinking skills. My communication skills were also exercised with meetings with my mentor to talk about my project. Overall it was very beneficial learning experience and I feel what I picked up will definitely be applied into future assignments and projects.

# Recommendations

For my project if I were to do it differently I would probably make the game a continuous runner instead of having to wind your way to the end. Make the game one long room that you traverse through, similar to the Mario games super bros. I feel it would make the game a little easier to manage. Also I would change the background image and walls to more appealing colours and or themes. A grass or outdoor levels would have look more pleasing.

Things I would do to improve the game would me to give the player the more abilities, such as slide or duck, which would be very useful against enemies with bullets. More abilities to open up more options for different obstacles or enemies. Having the player shoot would be a great aspect that could be implemented later in the game. Utilising the mouse click event to fire and kill enemies and add an overall more compelling gaming experience

More levels would also be an interesting addition. Having more levels means, long fun and more of a chance to add new exciting features. The addition of levels would have made the game more like a role playing game. Making it so there is more opportunity for a story as they progress level by level.

Lastly I think a high scoreboard should be added to display the top score player along with their name. This would be a nice interactive feature to make the game more playable. Trying to beat the new high score would be an awesome addition.