# Developer Diary

**Game Designer:** (Unidentified Classmate) **Game Developer:** Katelyn Graham (G00377785)

## First Thoughts

I have been asked to develop a game called “Formula: The Climb”, designed by the customer. After reading his design document I have an understanding what he is looking for in this game. There are some parts of his game that I feel he has not described in detail fully so I will have to get in contact with him so I can produce a game to the best of my abilities that he has planned.

Graphical user interface, text

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generatedGraphical user interface, application

Description automatically generatedA picture containing graphical user interface

Description automatically generatedA picture containing application

Description automatically generatedGraphical user interface

Description automatically generatedI started with doing up more graphical menu screens so I could picture clearer as to what the customer is looking for as they had only done up rough hand drawn sketches.

Pause Screen Win Race Screen Lose Race Screen

Menu Screen The Paddock Obstacle Information

These are the screens I envision when reading the design document. The changes I feel that should be made are:

* Menu Screen: As this is the first screen, there is no need for an exit game button option. Instead I feel this option should lead to an Information Screen about the possible obstacles the player will encounter.
* The Paddock/Obstacle Info Screens: These screens are fine, but we will need a back button option to get us back to the main menu screen or pause game screen, depending on where they were accessed from.
* Win/Lose Screens: These Screens are fine, but I feel that the buttons should lead back to the main menu screen and the player can start the game from there. If not, then there should be an option to go back to the main screen as well as start game again, as the player might want to access the paddock or obstacle information.
* Pause Game Screen: In this screen there should be an option to Resume Game option as there is no way to get back to the game. In the design document, it says that this screen can be accessed by the “Esc” key on the keyboard. I think that there should be an on-screen option as well to pause the game, as keyboard keys can get jammed and some users may not be able to access their Esc button. Also there is nowhere that lets the user know that the “Esc” key will pause the game.

Diagram

Description automatically generated with medium confidence

To get a feel for the movement through the screens I created a simple user interface

The red is the changes I would make.

## Assets

In the Design Document the customer did not specify any assets from the unity store. After searching the store myself I decided on assets that I felt where most suited to the customers description and game theme. I was unable to find assets for car debris so instead I have used tyre stacks and barriers as these things are found in racing. I used the following assets to develop the game:

1. Player
   * <https://assetstore.unity.com/packages/3d/vehicles/land/racing-cars-pack-1-2195>
2. Obstacles
   * <https://assetstore.unity.com/packages/3d/environments/roadways/cartoon-race-track-oval-175061>
   * <https://assetstore.unity.com/packages/3d/vehicles/land/racing-cars-pack-1-2195>
3. Environment
   * <https://assetstore.unity.com/packages/3d/environments/roadways/cartoon-race-track-oval-175061>
   * <https://assetstore.unity.com/packages/3d/environments/urban/simple-city-pack-plain-100348>
4. Collectibles
   * <https://assetstore.unity.com/packages/3d/high-detail-1970-s-supercar-wheel-92408>
5. Audio
   * <https://assetstore.unity.com/packages/audio/sound-fx/transportation/rotary-x8-free-engine-sound-pack-106119>
   * <https://assetstore.unity.com/packages/audio/sound-fx/shooting-sound-177096>
   * <https://assetstore.unity.com/packages/audio/music/track-day-racing-soundtrack-199016>
   * <https://assetstore.unity.com/packages/audio/sound-fx/transportation/i6-german-free-engine-sound-pack-106037>

## Developing Process

* I started by making a simple environment and getting basic player movement working, to do this I created the PlayerMove script. I added forward movement but also, I added left and right movement.
* For the left and right movement I created the LevelBoundary script. The player can only move left and right by 3. This will be the width of the playable platform. The left and right movement can be controlled by both the A and D key and the left and right.
* I have added the car asset to the player object. I adjusted the box collider around the player car. I made sure the movement looked right before moving on.
* I started to design a simple environment using different assets. Once I had one of these environments designed, I duplicated it twice and changed them both slightly. I added obstacles to these sections, although they do nothing yet.
* I created the GenerateLevel script so it will randomly place these sections to form a continuous platform. In this script it contains a GameObject array. It randomly picks one of the three sections and then adds it to the array placing them every 50 positions. Each section is a length of 50 positions.
* I started adding the collectible items, these are the Pirelli tyres. I made these triggerable. I created the CollectableControl script. I added a canvas to the game scene, I added a raw image for the border and a text box, this displays the amount of health collected
* Since the customer wants a max collectable amount of 3, I needed to make sure any health collected over the 3 would not affect the value. For this I created the HealthCollect script, in this there is a simple if statement. I also added sound effects to this script.
* To start the game I want it to look like a 3, 2, 1 countdown, I created the Starter script and I added texts to the canvas and added an animator to each. I have a 3, 2, 1 and a go text. All have a different animation to them which rotates them on the x axis, so they fade out. I also added a fade in animation, so the screen goes from pitch black to visible. I also added in sound effects, so it sounds like a race countdown. Until this is over the player cannot move left or right.
* I created the BigObstacle script. I decided to get the bigger obstacles working first and then I can use this script later to develop the script for smaller obstacles. I created different sizes of obstacles and added the assets to them, so they look like cars, barriers and tyre stacks. I added the BigObstacle script to all the obstacles, set the is Trigger on. If I can get obstacles working under the BigObstacle script, then it should not be too hard to change them to small obstacle scripts later. In this script once the obstacle and player collide, the player stops moving.
* I added the main camera to the player object so is follows the car during the game. When the player crashes, I want to have an animation that will play to let the player know they have crashed. For this, the animation I made the z position of the camera move so it bounces forward a bit and changed the z rotation so that is wabbles from side to side. When the player crashes this animation, and a crash sound is played.
* Like the start-up animation of the game I wanted a sequence of events to be played when the player had run into an obstacle. For this I created the EndRunSequence script. Initially I was just going to display a Game Over and then fade out but reading back over the customers design document I decided to add two different screens. I will need to create a distance count to decided between what screen is displayed at the end. I will display the win screen if the player beats the high score.
* I implement a distance script called LevelDistance. For this I used a coroutine. In this it just adds one to distance after one second wait period, it displays the distance to the game canvas.
* With the distance now implemented I went back to the end-run sequence. I added an if statement. This just plays different coroutines depending on if the player beats the current high score or not.
* I tested the game out and tweaked different aspects like speed, so it looked good in the game. I now want to add a difference between big and small obstacles. I copied the contents of the big obstacle script into SmallObstacles. This uses the health count to indicate whether it will minus one from the health count or if the count is 0 it will end the game.
* After implementing the small obstacles. I tested the game more. This is when I realised, I needed a way to implement levels. For this I decided that I would have subgroups within each section called “F3”, “F2” and “F1”. F3 had little obstacles but more health collectables, F2 increased the number of obstacles but decreased the health collectables. F1 had the most obstacles but little health collectables. Using a new script called ActiveObjects, I used this, and the player move script to decide which subsections are activated at different times. To do this I used the distance ran number. F3 is used between 0-100, then the next section, F2 starts between 100-250 and then from 250 onwards it will show the F1 subgroup.
* I decided to add more sections to the game, I now have 8 sections that will create the platform.
* After testing the workings of the game I then just needed get a main menu scene, a pause scene and to get the end scenes and win scene buttons working. The main menu and end scenes where simple to get going but when I tried to get a pause scene going, I found it hard to then resume the game, so instead of going to a completely new game I decided to just show the pause options in the game scene. This meant then to pause the game all I had to do was put the time scale to 0. To resume I then just put it back to 1.
* To get buttons working and movement through my game I had to create the MainMenu script. This just allows exit game, play game, quit application and a lot more functions work.
* I did some final tweaking to my game and tidying up. I ran a set of tests that all passed. I built and ran the game some more to make sure it was all working fine.

## Table Description automatically generatedTable Description automatically generatedTesting Plan

Table

Description automatically generated