Platformer   
Cross Platform Development

Author: Thomas Lamb

# Change Log

Updates made to the document should be described below.

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Author | Date of change | Description |
| 0.0.0 | AIE | 31/08/2020 | Initial Template created |
| 0.7.0 | Thomas Lamb | 20/9/2020 | Basic platforms and movement made |
| 0.8.0 | Thomas Lamb | 24/9/2020 | Created android version |
| 0.9.0 | Thomas Lamb | 28/9/2020 | Fixed Ui scaling of menus |
| 1.0.0 | Thomas Lamb | 7/10/2020 | Finished builds and removed bugs. Playable. |
| 1.0.1 | Thomas Lamb | 13/10/2020 | Tweaked documentation to meet guidelines |
| 1.0.2 | Thomas Lamb | 17/10/2020 | Changed descriptions according to feedback |

Contents

[Change Log 1](#_Toc53490345)

[1.0 Development Environment 3](#_Toc53490346)

[1.1 Game Engine 3](#_Toc53490347)

[1.2 Source Control 3](#_Toc53490348)

[2.0 Game Overview 3](#_Toc53490349)

[2.1 Genre 3](#_Toc53490350)

[2.2 Camera Perspective and Movement 3](#_Toc53490351)

[2.3 Platform 3](#_Toc53490352)

[2.4 Installation Methods 3](#_Toc53490353)

[2.5 Technical Goals 3](#_Toc53490354)

[2.6 Game Objects and Logic 4](#_Toc53490355)

[3.0 Controls 4](#_Toc53490356)

[3.1 Windows / Web 4](#_Toc53490357)

[3.1 Android / Touch 5](#_Toc53490358)

[4.0 Mechanics 5](#_Toc53490359)

[4.1 Hazards 5](#_Toc53490360)

[5.0 Graphics 5](#_Toc53490361)

[5.1 ‘Mission’ / ‘Level’ structure 5](#_Toc53490362)

[6.0 Level design 6](#_Toc53490363)

[7.0 Interface 6](#_Toc53490364)

[8.0 Progress report and feedback Meeting Minutes 12](#_Toc53490365)

[Friday 4th September 12](#_Toc53490366)

[Wednesday 9th September 12](#_Toc53490367)

[Thursday 10th September 12](#_Toc53490368)

[Friday 11th September 13](#_Toc53490369)

# Development Environment

## Game Engine

Unity and version 2019.3.6f1

## Source Control

Link to GitHub repo - https://github.com/ThomasLambProgramming/CrossPlatformUnity

# Game Overview

Play as a cube and jump between moving platforms to reach the last gold platform to finish the game.

## Genre

Platformer

## Camera Perspective and Movement

3rd person camera perspective moves in accordance with the camera perspective with either wasd on the pc platform or joystick ui elements on the android platform, the camera can be moved left or right.

## Platform

The platforms available for this game is Android, webGL and windows, to deploy on each platform, simply open the unity project and export the build, the code is written so the game automatically switches ui and movement for each platform.

The platform concerns for development is to keep the UI for each platform scaling with the different resolutions and aspect ratios of both monitors and different android devices. To create movement that can be used by both a keyboard and touch controls.

## Installation Methods

Windows: users will have a exe installer to download, following the prompts that come up after opening the installer will have the game installed onto the users device.

Android: Users will have to download the apk and then run the apk to install the game onto their device.

WebGL: Users will follow a link to the game and all that is required is to have a browser that supports the WebGL platform such as chrome.

## Technical Goals

* Create working UI for windows and android
* Create moving platforms that can move in any direction
* Create limited camera movement
* Keep webGL build above 30fps
* Keep UI scaling correct

## Game Objects and Logic

Android UI: The android ui will have two separate joysticks, one for controlling the left right up down control of the player, the other will control the camera rotation. The ui will also have a jump button which when pressed will make the player jump. A pause button will be in the top left of the screen to stop the game whenever the player needs to.

Both the pause menu and fail menu will set the application timescale to 0 to “pause the game”.

Fail Menu: The fail menu will have a retry button which reloads the scene that is currently in use, a exit to main menu button, which will load scene 0, and a exit game button which will close the application.

Pause Menu: The pause menu will have a resume button to set the timescale back to 1 to continue the game, an exit to main menu button, which will load scene 0, and a exit game button which will close the application.

Main Menu: The main menu has a play button which loads the main game scene, and a exit game button which will exit the application.

Win Menu: the win menu has a text element that says, “You Win!” and a menu button to return to the main menu and a exit game button to stop the application.

Cube player: the player is a small cube which will take the player input to move and jump, with wasd on windows the player can move around the game and on android the player has joysticks to do the same movement, the player has a jump which when the velocity of the y axis reaches 0 additional gravity will be applied to simulate a more “snappy” movement. The player moves with a rigid body component and a box collider for the platform and ground registering.

Moving platforms: the moving platforms work with the unity animator to loop through a preset animation, so the platform follows a set path. They also have 2 box colliders, one to interact with the player to not fall through it and the other a trigger to parent the player so the platform can take the player with it without the player sliding off.

end platform: the end platform is just a scaled-up cube that has a tag with it of “Finish” so when the player collides with it the win menu is loaded.

Static platforms/Walls: the walls and the static platforms are just differently scaled cubes that have box colliders to the player can be either restricted or held off the ground.

Joysticks: the joysticks are a 3rd party asset to allow android users to have input for the game, they work the same as the unity axis input system by taking a measurement from the middle of how much the joystick has moved in the x and y axis and from there we can simulate the same movement as the wasd movement.

# Controls

## 3.1 Windows / Web

Windows: Wasd movement and q/e for camera movement left/right, esc for pause and space for jump.

## 3.1 Android / Touch

Android: joystick for player movement, joystick for camera movement, pause and jump buttons

# Mechanics

* Movement

The player can use wasd or joysticks to move forward backwards, left or right to traverse the game and go from platform to platform.

* **Jumping**

When the player presses space or uses the jump button the player will jump in the air, the player will remain in the air for longer when the key/button is held and when the player has reached the peak of the jump the player will have extra gravity applied to make the jumping feel more solid overall and faster.

* **Moving platforms**The platforms go along a set movement pattern, stopping when it reaches the end and start of its path, the player once getting on the platform cannot move with wasd when the platform is moving, this is to stop sliding on the platform, however the player can jump to regain control and get off the platform early before it stops again.

## Hazards

The entire floor other than platforms will cause the player to instantly fail and must restart the level.

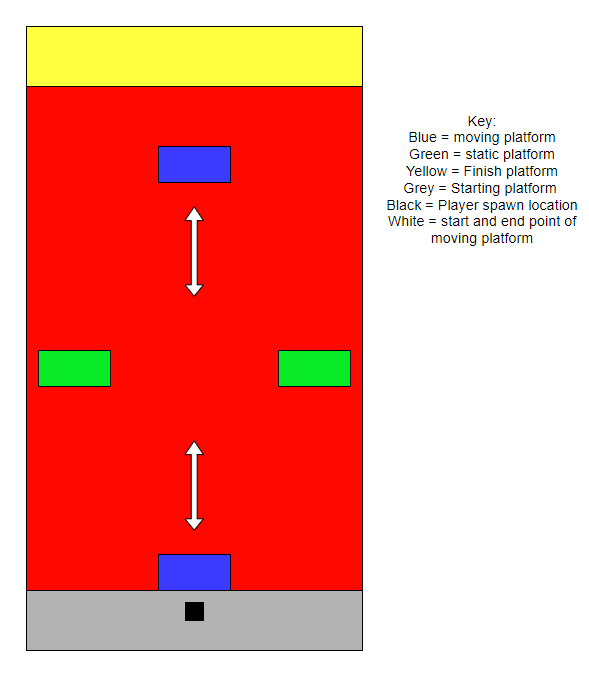
# Graphics

Basic graphics, low poly assets, 3rd person style camera. Simple materials from unity.

## ‘Mission’ / ‘Level’ structure

There is currently one level and you have to move forward towards the golden platform which is the finish.

# Level design



# Interface

The player on windows and android has a fail menu ui, and a pause menu ui, both of which are the same except the fail has a retry button and the pause has a continue button. (they are placed in the same position on the screen) both menus have a exit game and a return to main menu screen

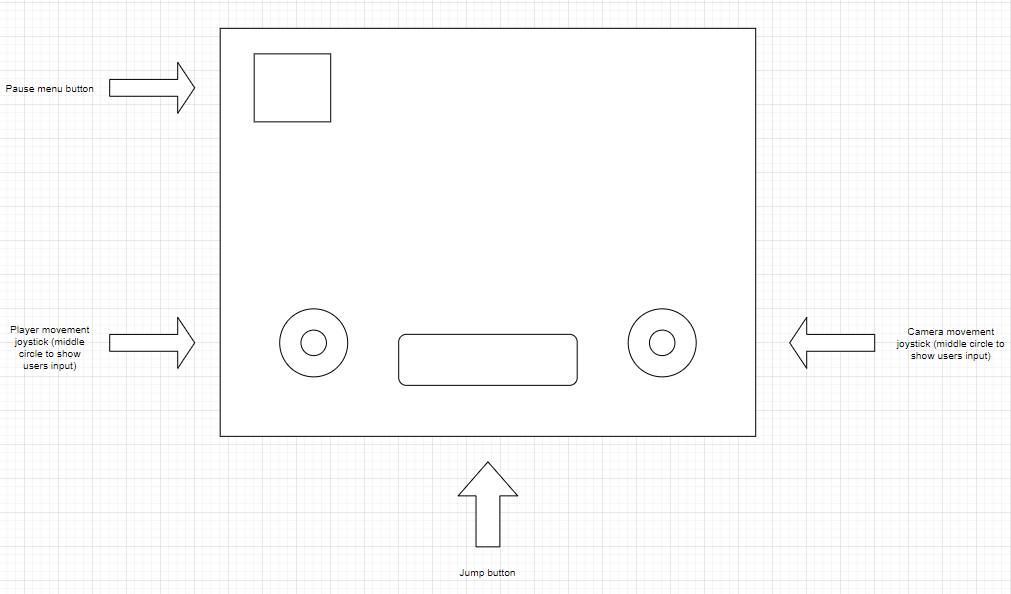
Android has its own main game ui with 2 joysticks on either side for the moving of the player and the camera, there is a jump button in the middle bottom of the screen for jumping and has a pause menu button in the top left of the screen.

The main menu has an exit game button and a play game button

The finish menu has a you win text, a main menu button and an exit game button.

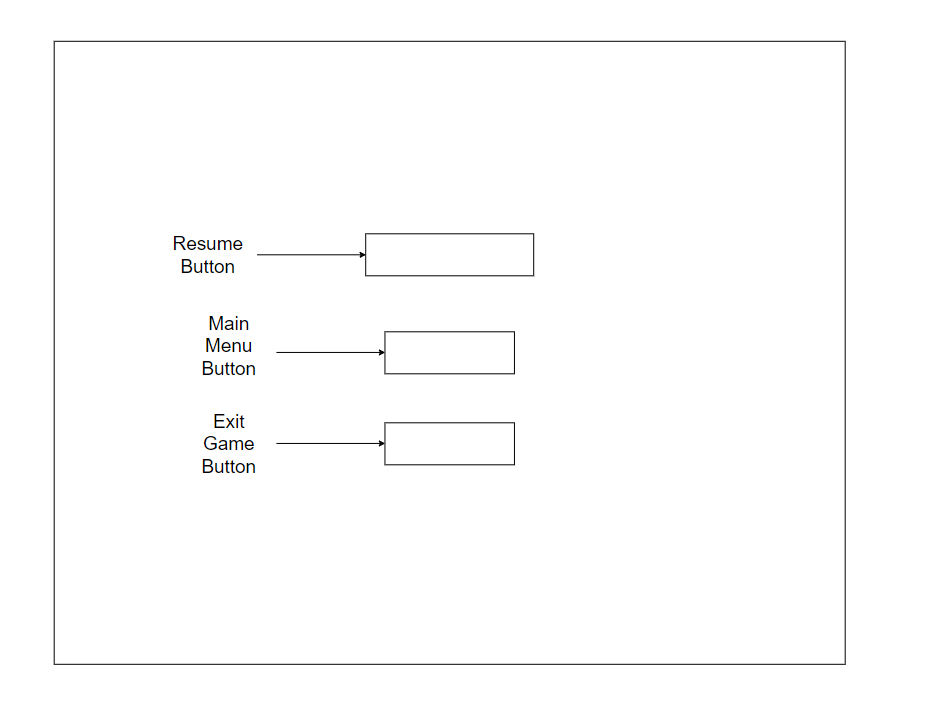
Android Main Game UI:

This UI will only be used for android, as it takes in touch controls which is only needed by android users.



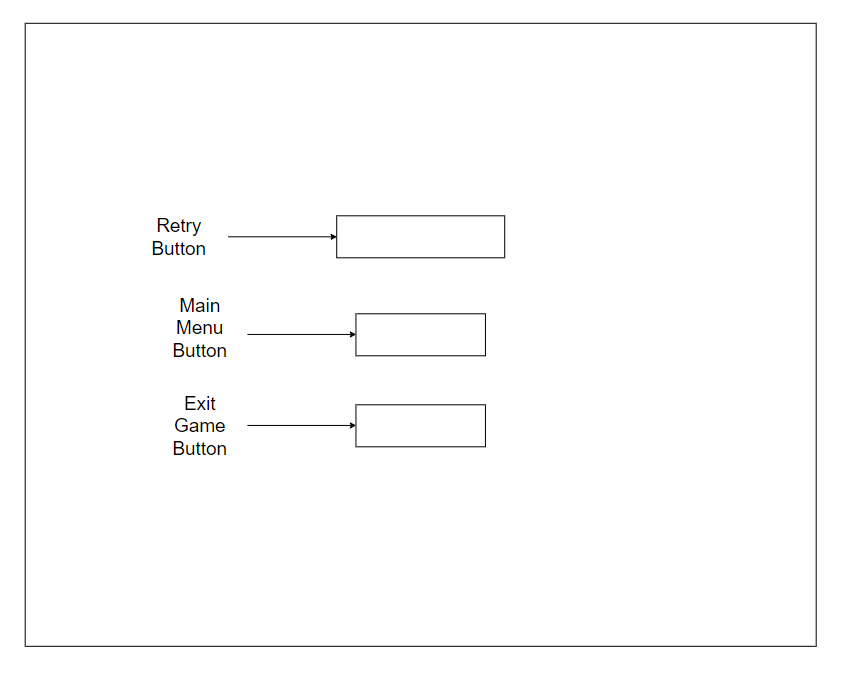
Pause menu:

The pause menu is transparent, the main game can be seen behind the buttons. Between platforms there is no difference as all buttons are useable on all platforms.



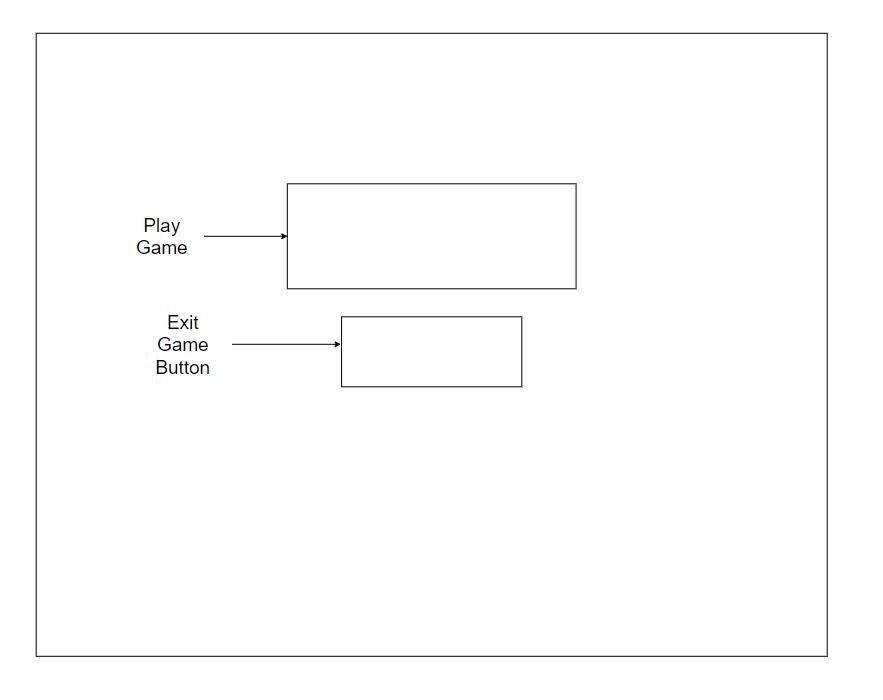
Fail Menu:

The fail menu is transparent and the players can see the main game behind the buttons. Between platforms there is no difference as all the buttons can be used on all platforms.



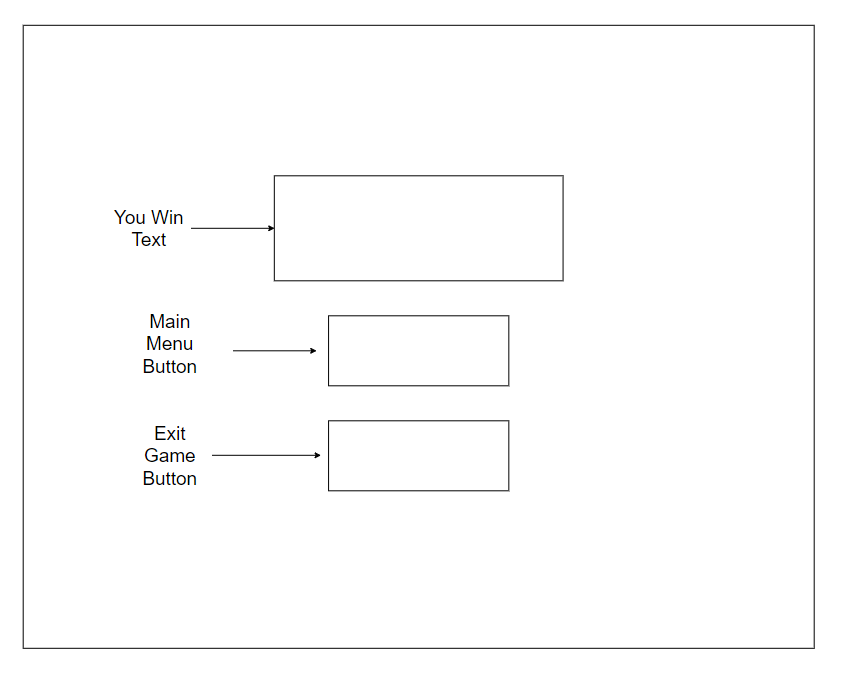
Main menu:

The main menu is simple as there is a start game button and a exit game button. Between platforms there is no difference.



Winning Screen:

The winning screen has a you win text, a main menu button and a exit game button. All platforms have the same layout as all buttons work on all platforms.



# Progress report and feedback Meeting Minutes

## Friday 4th September

Describe state of project

* Non Playable

Feedback from teacher and peers:

* Create something that is playable with movement or basic outline

Action Items:

* Create Movement
* Make UI

## Wednesday 9th September

Describe state of project

* Playable with basic scene

Feedback from teacher and peers:

* Better Movement
* Fix collisions with walls

Action Items:

* Change how collisions are detected
* Change movement

## Thursday 10th September

Describe state of project

* Playable with better collisions and movement

Feedback from teacher and peers:

* Make a more detailed level
* Add more platforms / obstacles

Action Items:

* Create moving platforms
* Make better UI which is simpler
* Build android version

## Friday 11th September

Describe what has been done since last time

* Created moving platforms.
* Made player move with them
* Added android support

Feedback from teacher and peers:

* Add a fail to the player so they have to use the platforms
* Make a winning menu

Action Items:

* Make the player be able to fail
* Bug fix small problems (glitchy collisions at certain points)
* Make webGl build and put it on the page