Game Design Document For ARGO 2016-17

[Introduction](#_v47k0vmljbq)

[Story](#_gskzq86yef0d)

[Aesthetics](#_tobda4j0y1jr)

[Technical Notes](#_9m422wa12ye7)

[Features](#_hm0scx4n71v9)

[Basic Player](#_1joqin3j2q7u)

[Conditions of Satisfaction](#_wx3nhvbthr7h)

[Basic World](#_y25cn1eqkqg1)

[Conditions of Satisfaction](#_kdh0qn98e120)

[Camera](#_tzx744gcihyh)

[Conditions of Satisfaction](#_adju0fso6pqy)

[Input](#_f7fpbtysj3ic)

[Conditions of Satisfaction](#_jog90ud9y146)

[Obstacles](#_gzmkntgm4ynm)

[Conditions of Satisfaction](#_rsj4jvpd1los)

[Advanced Player](#_a6chip68urrf)

[Conditions of Satisfaction](#_s1edc3pchg86)

[Advanced World](#_feo5vqa1gr12)

[Conditions of Satisfaction](#_xf5drkhutzr)

[Collisions](#_nu8vmeo5b1s7)

[Conditions of Satisfaction](#_h8h3lr7rbh8z)

[Shooting](#_5wkbht2k3xyi)

[Conditions of Satisfaction](#_s08ir1lmqmc)

[Basic Networking](#_ecoodry5my9d)

[Conditions of Satisfaction](#_hzto12ncrzgy)

[Basic AI](#_g8exa0u46a8r)

[Conditions of Satisfaction](#_pjktcjuu8sd9)

[Scenes](#_vnqcte5osyu4)

[Conditions of Satisfaction](#_56cmx24cpo38)

[Basic Gameplay](#_c5incdfemnpu)

[Conditions of Satisfaction](#_49puo4eayx6z)

[UI](#_og1rnm114qv4)

[Conditions of Satisfaction](#_bpbvkphfz5j6)

[Networking](#_ay1ye4qhtqt9)

[Conditions of Satisfaction](#_th7u0ek94rmm)

[Gameplay](#_pcsoy67c9spk)

[Conditions of Satisfaction](#_cgxprnv0txwq)

[AI](#_2wrdo3430x27)

[Conditions of Satisfaction](#_vfh6mty2m2e3)

[Audio](#_8it165j0c5gp)

[Conditions of Satisfaction](#_22999qczoqbd)

[Feature](#_myh2x1ecnaug)

[Conditions of Satisfaction](#_5x6uui1lfdar)

[Feature](#_g1u9ce35skn0)

[Conditions of Satisfaction](#_801ctvdnpm7t)

# Introduction

This GDD describes a game to be implemented in the Real World Modelling problem 4, ARGO.

# Story

On a distant world, 8 warriors race for the glory of their home planet.

# Aesthetics

The game is meant to create a sense of enjoyable, friendly competition between players.

The presentation is sleek and cartoonish to create a more child like feel.

# Technical Notes

In this project we use:

SDL2.

BOX2D.

SDL Image.

SDL TTF.

SDL Mixer.

SDL Net.

# Features

## [Basic Player](#_1joqin3j2q7u)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-1)

### Conditions of Satisfaction

1. The Player can move in 360 degrees.
2. The Player has a Maximum velocity of 5 units per second.
3. The Player is moved Up/Left/Down/Right using the W/A/S/D keys.

Manual Test:

* Use the W/A/S/D keys to move the player.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWcUZaVFZISVFYNGM)



## [Basic World](#_y25cn1eqkqg1)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-2)

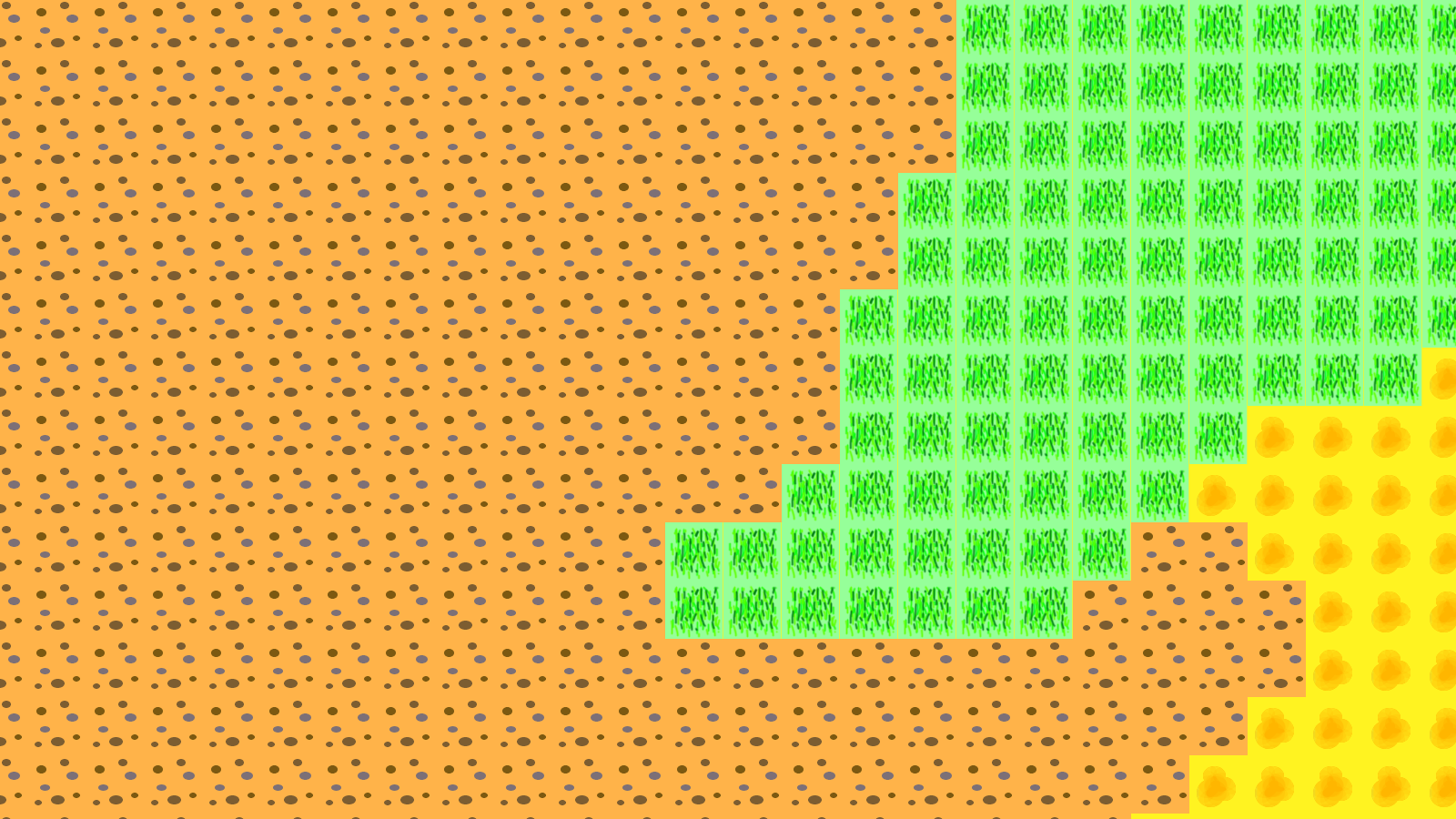
### Conditions of Satisfaction

1. The World will be rectangle/square in shape.
2. The World size will be tiled with temporary textures.
3. The World has a defined border.

Manual Test:

* Start the Application.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWd2xYZXdEeEM1ZEk)



## [Camera](#_tzx744gcihyh)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-3)

### Conditions of Satisfaction

1. The Camera will follow the player, keeping it center.

Manual Test:

* Use the W/A/S/D keys to move the player.
* The camera keeps the player centered.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWcUZaVFZISVFYNGM)



## [Input](#_f7fpbtysj3ic)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-4)

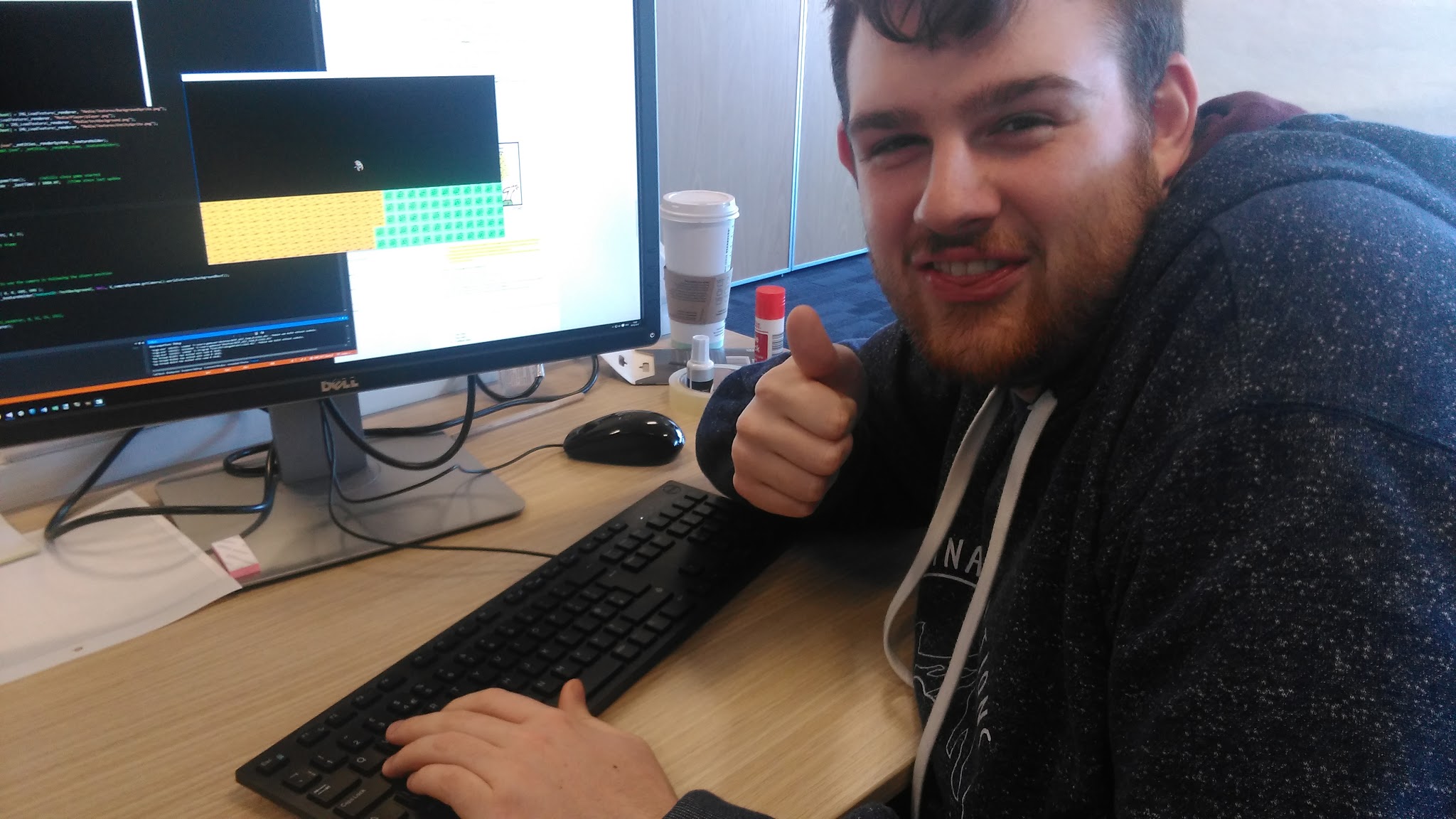
### Conditions of Satisfaction

1. Assign keys to commands, with an input type
2. When the key used matches the input type, the command assigned is activated.
3. The Player is moved Up/Left/Down/Right using the W/A/S/D keys.
4. The Application is closed by pressing Escape.

Manual Test:

* Use the W/A/S/D keys to move the player.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWcUZaVFZISVFYNGM)



## [Obstacles](#_gzmkntgm4ynm)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-19)

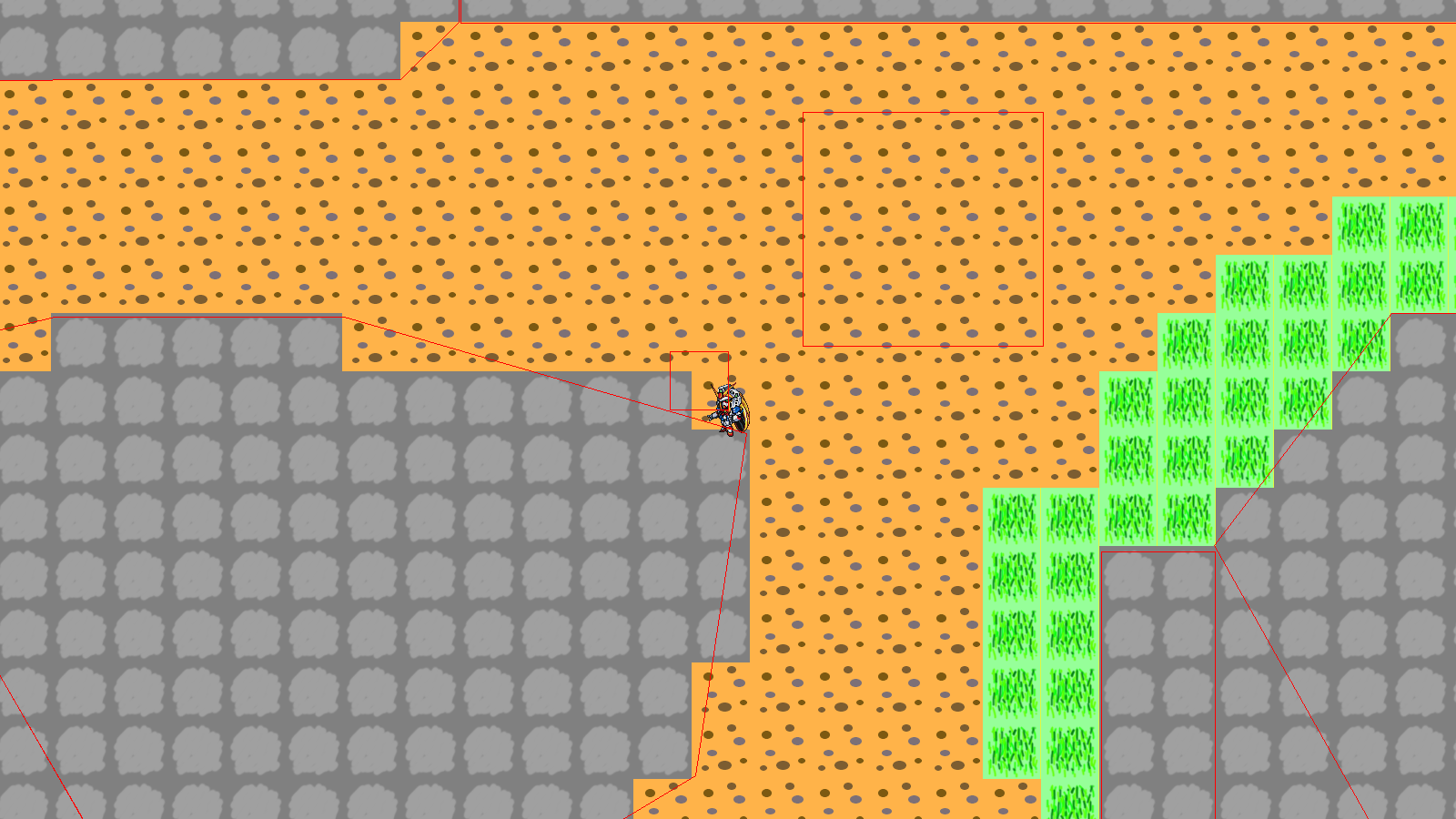
### Conditions of Satisfaction

1. The Player has a Box2D Dynamic Body.
2. The Player collides using Box2D collisions.
3. The walls are box2d polygons that collide with the player.

Manual Test:

* Move the Player using W/A/S/D.
* Collide the player with a Box2D object and the player will bounce appropriately.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWbWl4dGJGZmtkNjg)



## [Advanced Player](#_a6chip68urrf)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-20)

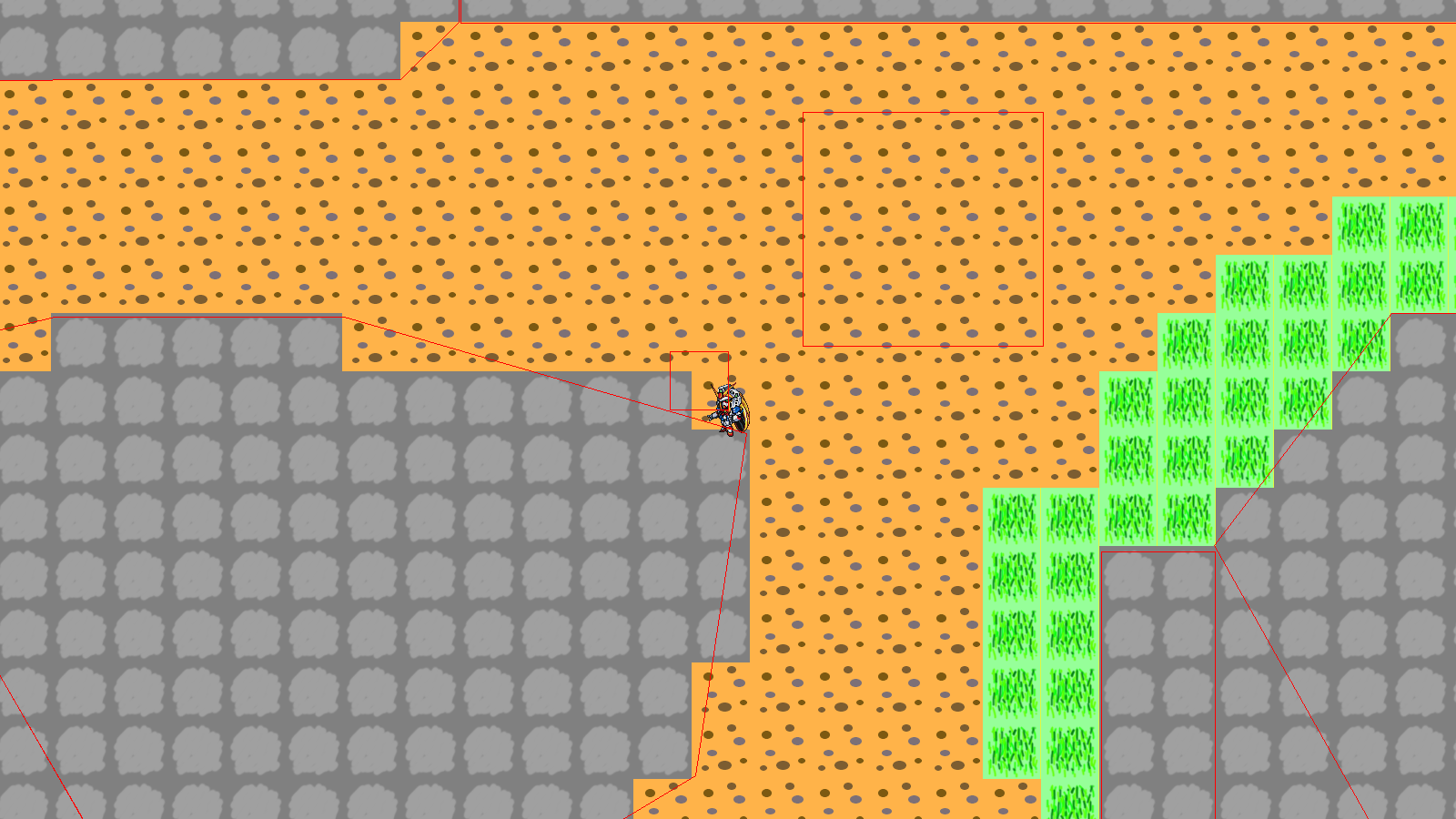
### Conditions of Satisfaction

1. The Player accelerates at a rate of 5 units per update.
2. The Players max velocity is 50 units.
3. The player decelerates by 5% per update.
4. The Player has a Box2D Dynamic Body.
5. The Player collides using Box2D collisions.

Manual Test:

* Move the Player using W/A/S/D and the player will accelerate.
* Stop using the keys and the Player will decelerate.
* Collide the player with a Box2D object and the player will stop moving that direction.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWbWl4dGJGZmtkNjg)



## [Advanced World](#_feo5vqa1gr12)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-21)

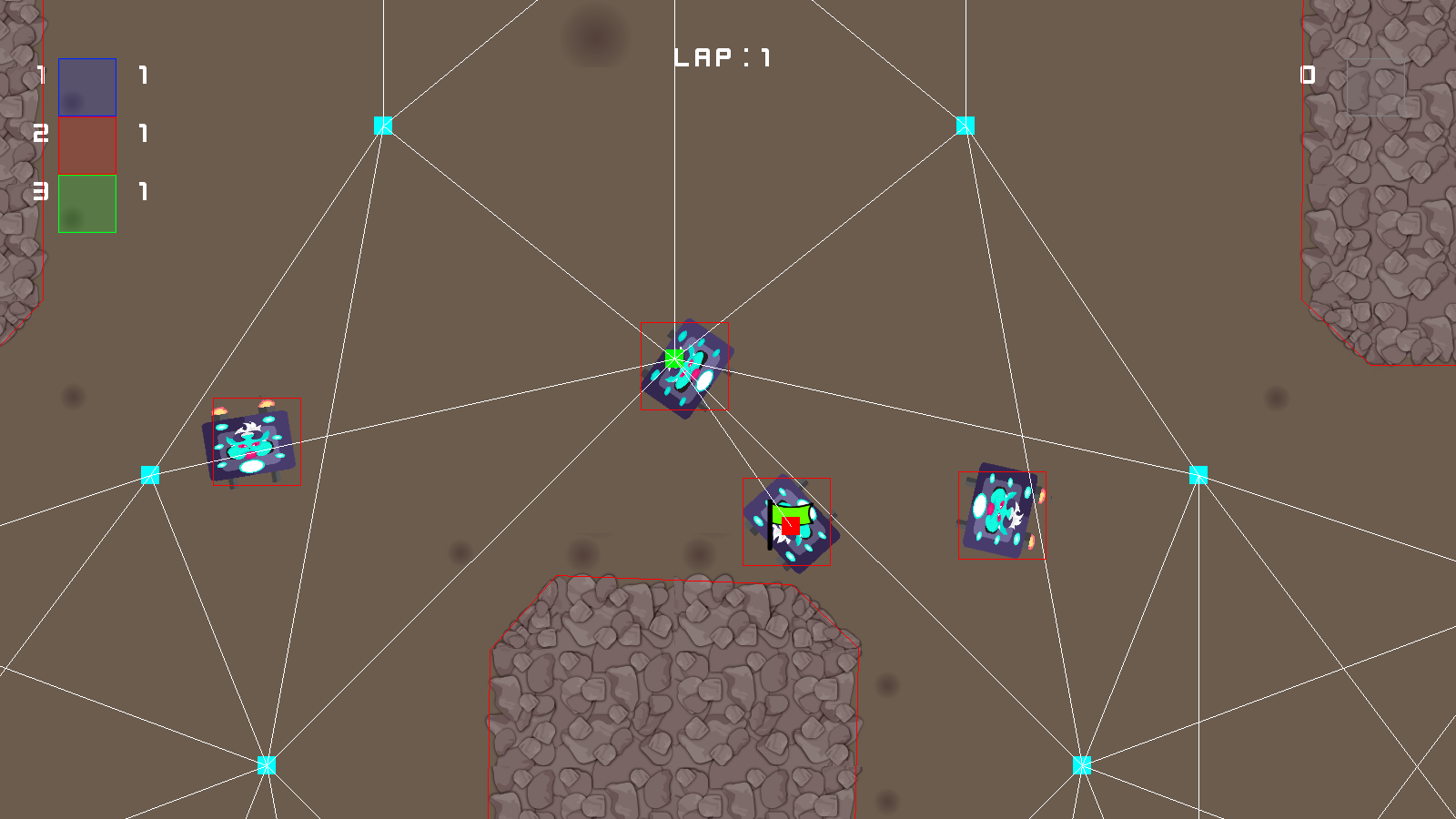
### Conditions of Satisfaction

1. Moving the Player moves the camera.
2. The Camera stops moving when the borders of the world are reached.
3. The Player collides with the bounds of the world.

Manual Test:

* Move the Player using W/A/S/D.
* The Camera will automatically follow the Player.
* Move the Player to the edge of the map, and the Camera will stop at the boundary.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWaTZPQXo4NzFVX28)



## [Collisions](#_nu8vmeo5b1s7)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-22)

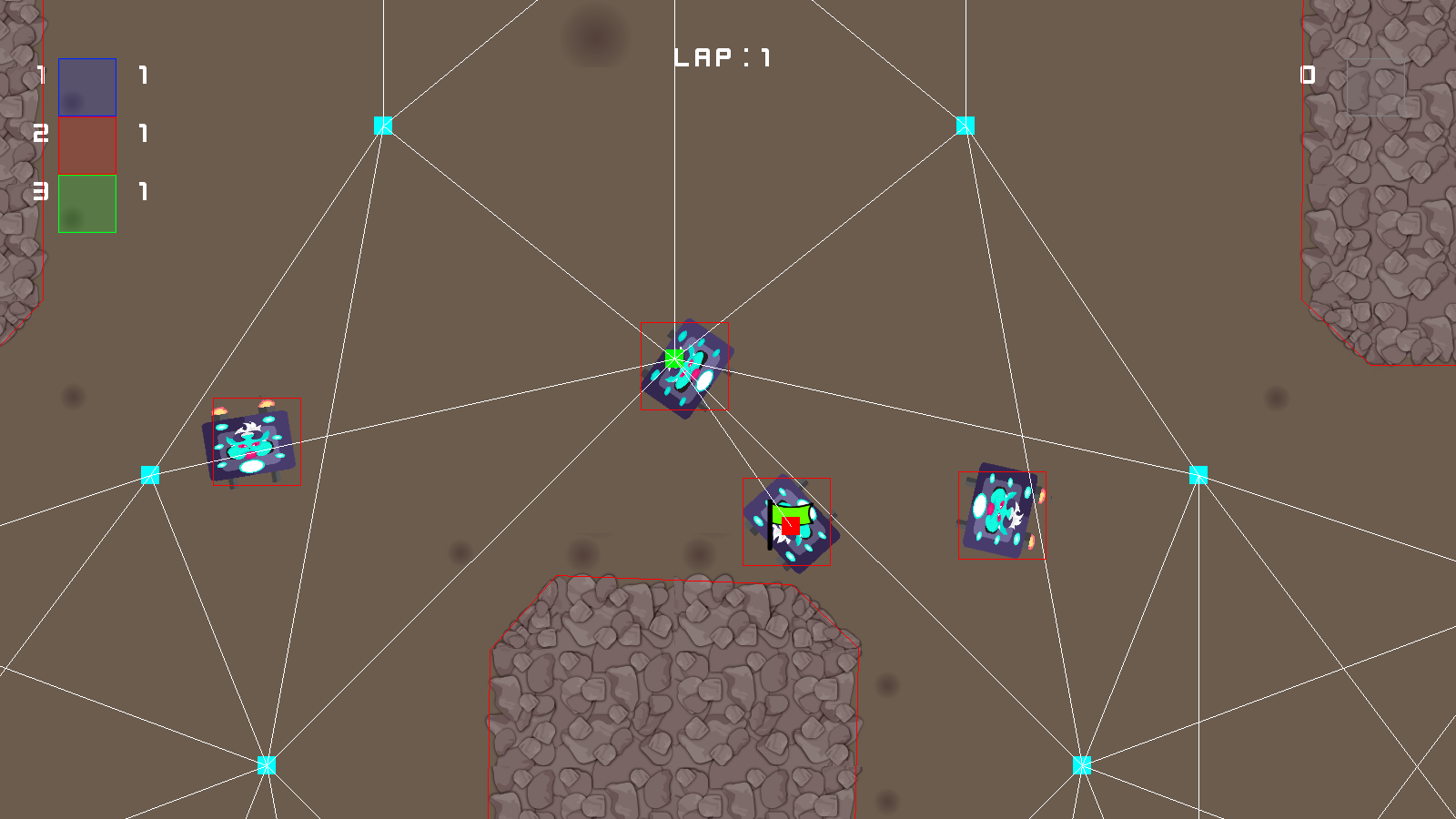
### Conditions of Satisfaction

1. Obstacles are made using Box2D bodies and polygons.
2. The Collision component runs Box2D collision detection.
3. Using Box2D, collisions boxes can be drawn for each object.
4. The Player will bump off walls and other players, as well as collide with checkpoints, the flag, and pick-ups.

Manual Test:

* Move the Player using W/A/S/D.
* Collide the player with a Box2D object and the player will bounce appropriately.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWaTZPQXo4NzFVX28)



## [Shooting](#_5wkbht2k3xyi)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-29)

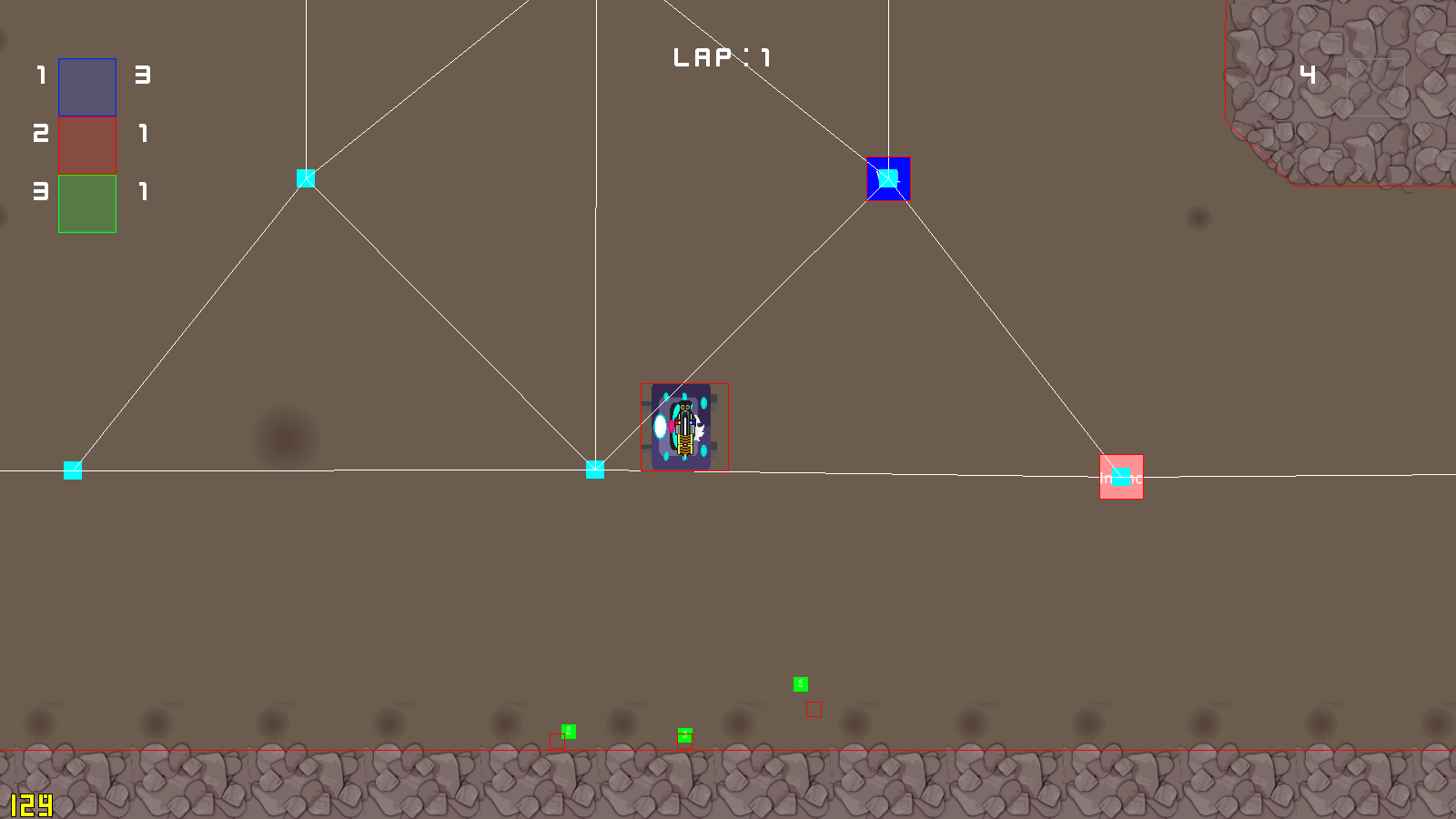
### Conditions of Satisfaction

1. The Player's Weapon position is set to the front of the player.
2. The Weapons rotation points forwards..
3. The Weapon fires bullets in the direction it is pointing.

Manual Test:

* Change the rotation of the Weapon by moving the mouse.
* Press space to fire.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWeW5wT0hCWXJmTVk)



## [Basic Networking](#_ecoodry5my9d)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-39)

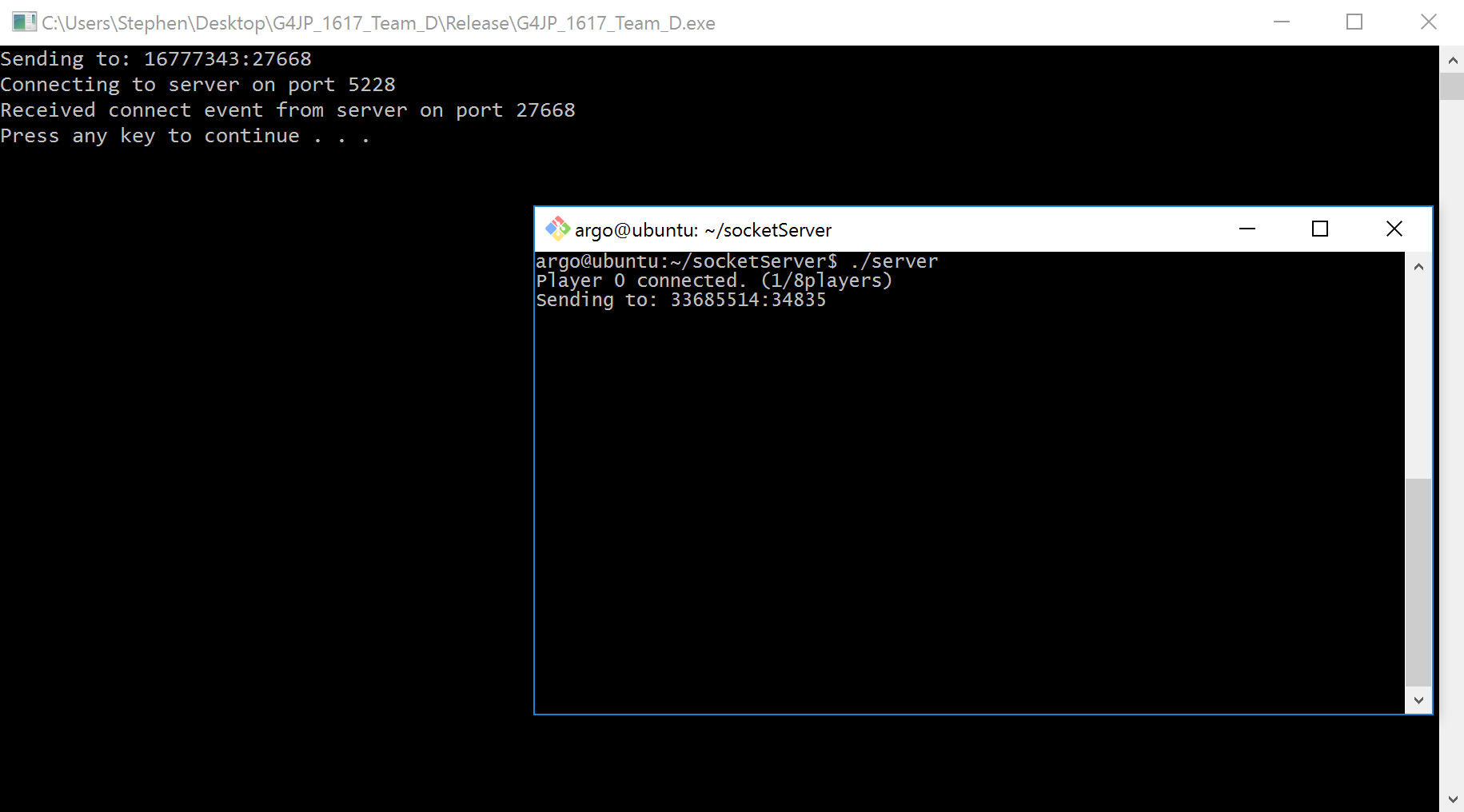
### Conditions of Satisfaction

1. SL Net allows the client to send a message to the server.
2. The Server echoes the message back to the client.

Manual Test:

* Enter a message to send.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWVGhaaU5tZmo1YTA) - (Video of connecting to Lobby on server)



## [Basic AI](#_g8exa0u46a8r)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-40)

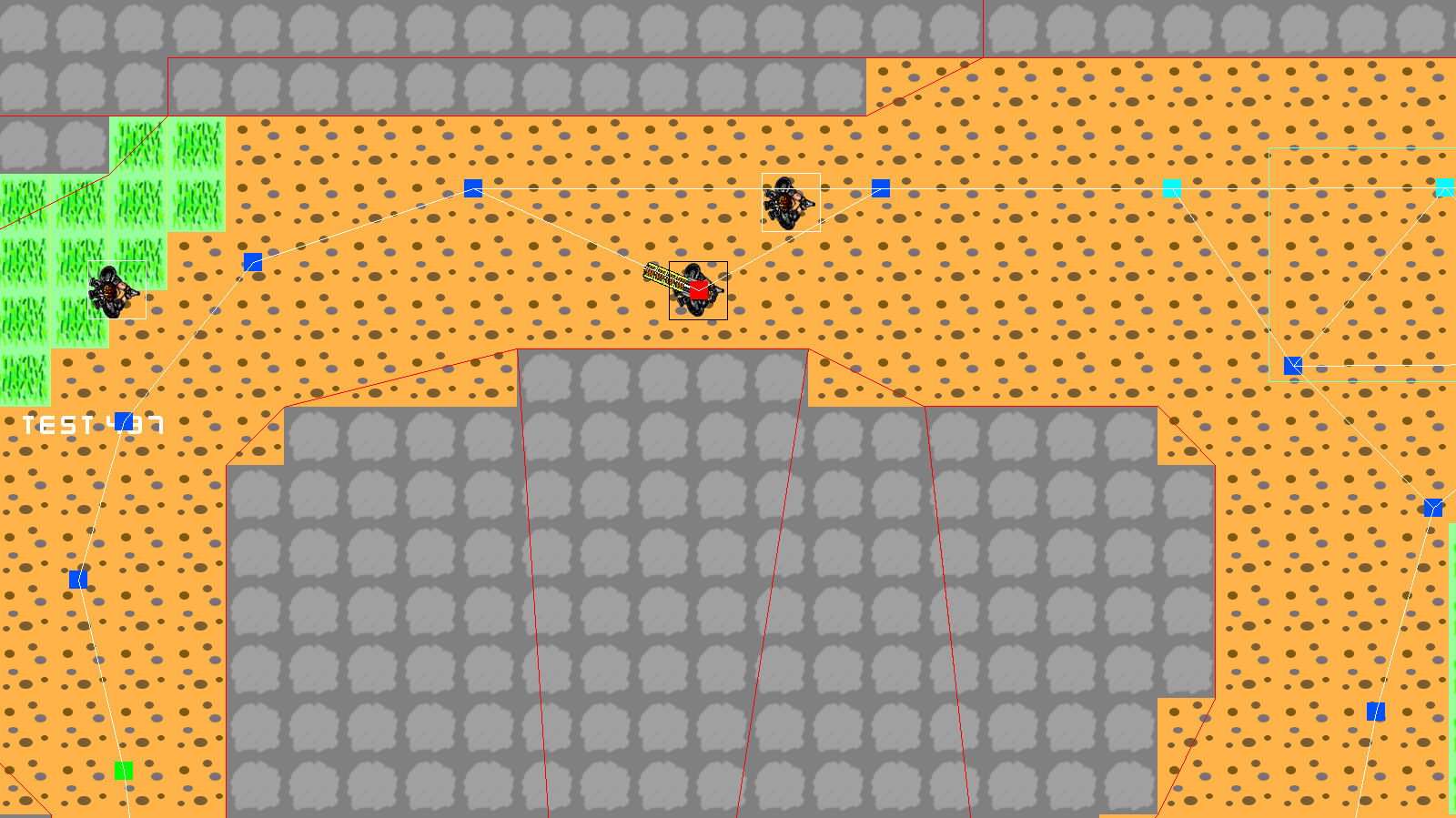
### Conditions of Satisfaction

1. Multiple Enemy entities are spawned.
2. The AI does an A Star search on waypoint nodes.
3. The AI searches for the Player, and moves towards that position.

Manual Test:

* Move the player using W/A/S/D.
* The AI will use A Star to find the Player's node.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWcGZjcDJkVlQ5VWM)



## [Scenes](#_vnqcte5osyu4)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-41)

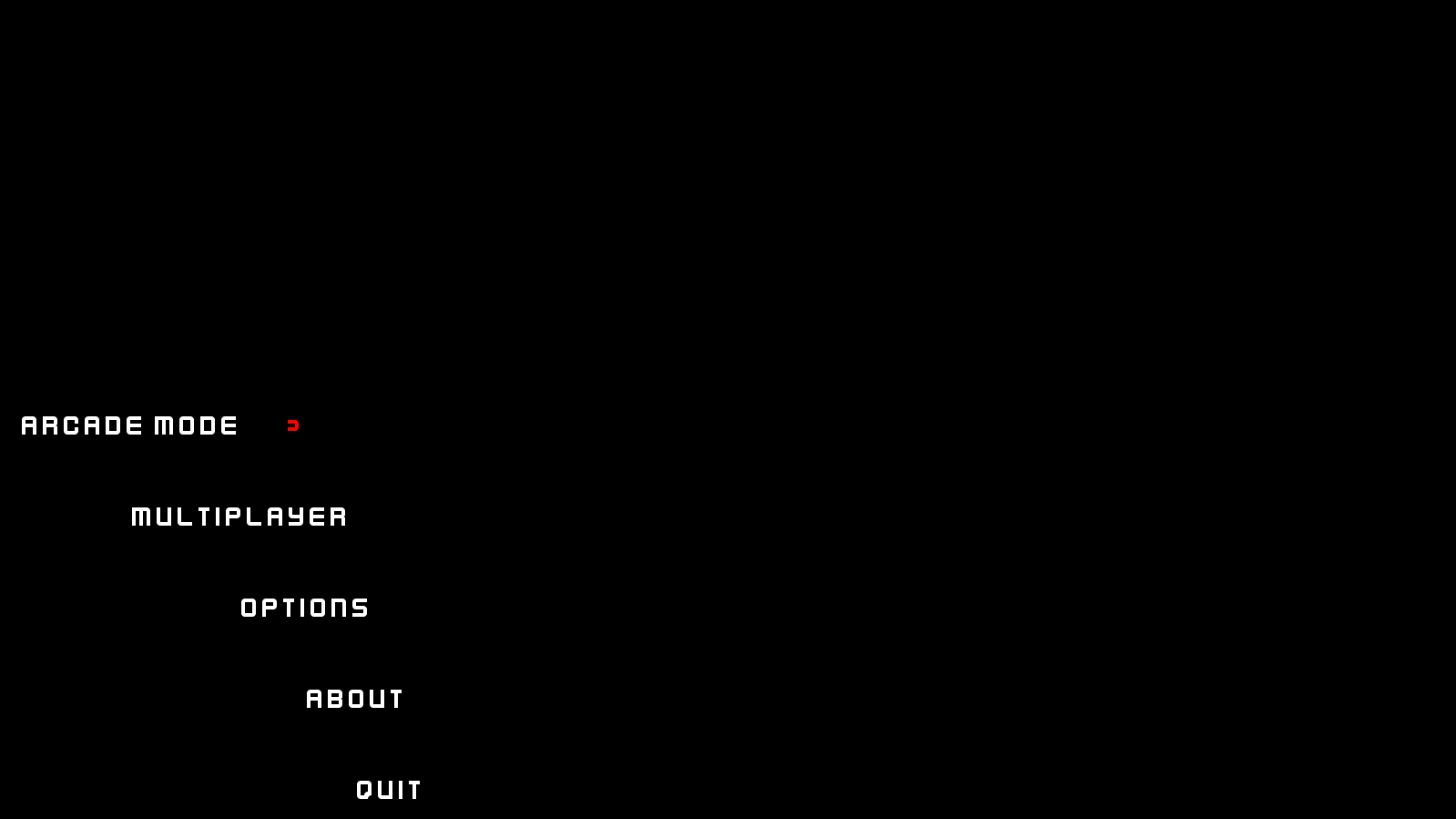
### Conditions of Satisfaction

1. The Main Menu is loaded at the start of the application.
2. The Menu is traversed using W and S.
3. Enter accesses the current highlighted option.
4. Backspace returns you to the previous scene.

Manual Test:

* Use W and S keys to traverse the Menu.
* Use Enter to select the highlighted field.
* Use Backspace to return to the Main Menu from any Scene.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWSmxlbXR1LXN6YnM)



## [Basic Gameplay](#_c5incdfemnpu)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-42)

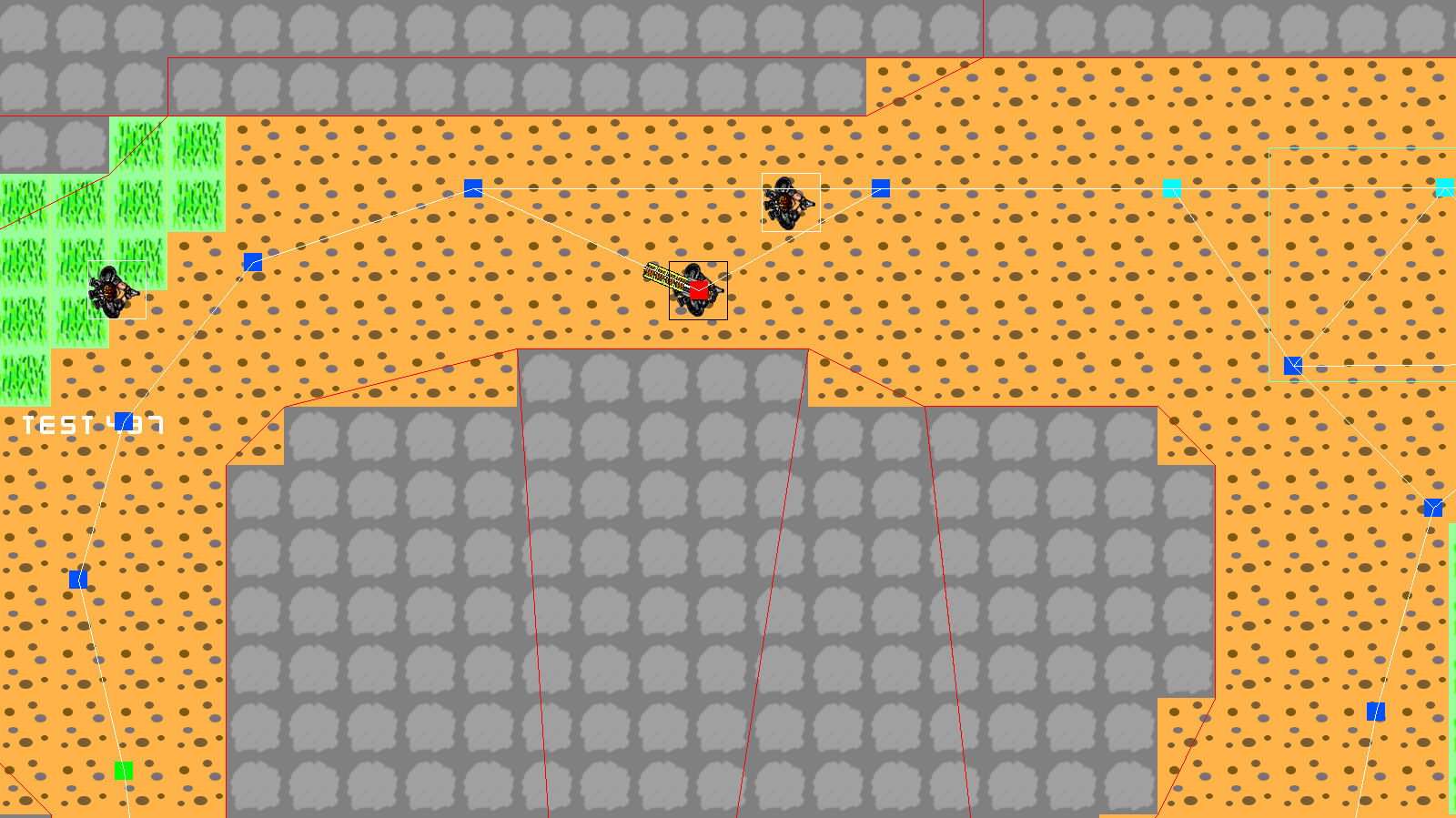
### Conditions of Satisfaction

1. The Player can move around the track.
2. The player and Enemies collide using Box2D.
3. The Enemies search for the Player.

Manual Test:

* Move the player using W/A/S/D.
* Hit off obstacles to see effects on movement.
* The AI will use A Star to find the Player's node.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWcGZjcDJkVlQ5VWM)



## [UI](#_og1rnm114qv4)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-61)

### Conditions of Satisfaction

1. The UI is displayed on screen during the Game Scene.
2. The UI updates with the state of the next checkpoint, the first 3 players in the race, weapon held and its ammo, and a mini map.

Manual Test:

* Move the player using W/A/S/D.
* Collect the Flag to display you are holding the flag.
* Hit a Checkpoint to display the next Checkpoint.
* Collect a Power-Up to display your current weapon.
* Move around the map to update the Mini-Map.

Video Link

Picture Image

## [Networking](#_ay1ye4qhtqt9)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-63)

### Conditions of Satisfaction

1. Player can enter the Multiplayer scene, and will see a list of available session with: current players in the session, and session ID.
2. If a new session is created by another Player, their list will also update.
3. A Player can create or join a session.
4. When the player enters a session they will see a list of Players within the session, identified by an ID number, also there is indication of which Player you are.
5. Players can ready up, and their name in the list will turn green, this will notify other Players of their status.
6. Once all Players are ready, a countdown from 3 begins, when it’s done, the Game begins with all Players joined.
7. As the Local Player moves, he sends his state to all other Players, that player will move on other players screens.

Manual Test:

* Use W/S keys to traverse menus.
* Enter the Multiplayer Scene by Pressing enter, or clicking on the text using the mouse.
* Traverse to the desired lobby session.
* You can also create a new lobby or refresh the lobby list, by selecting the options.
* When in the lobby, you can press a button to ready up.
* When all players are ready, the host of the lobby can start the game.

Video Link

Picture Image

## [Gameplay](#_pcsoy67c9spk)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-64)

### Conditions of Satisfaction

1. The Player and Enemies can move around the track.
2. The player and Enemies collide using Box2D.
3. The Enemies search for the Flag or Flag holder.
4. When the Flag holder hits the correct checkpoint, their designated checkpoint changes.
5. When an Entity bumps into the Flag holder, the Flag is dropped and the holder is staggered.
6. By hitting all checkpoints, an Entity does a lap.
7. The first Entity to do 3 laps wins the level.

Manual Test:

* Manual test

Video Link

Picture Image

## [AI](#_2wrdo3430x27)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-65)

### Conditions of Satisfaction

1. Multiple Enemy entities are spawned.
2. The AI does an A Star search on waypoint nodes.
3. The AI Entities will seek the Flag or Flag Holder, and moves towards that position.
4. If the AI Entity has a Weapon, it will fire at the Flag Holder.
5. If the AI entity has the Flag it will target the next Checkpoint in its lap.
6. The AI can predict the next checkpoint the flag holder is going to and go there first.
7. If the AI Entity completes 3 laps, it wins the game.

Manual Test:

* Move the player using W/A/S/D.
* Pick up the Flag.
* The AI Entities will seek the Player.
* AI Entities will attempt to get the Flag by bumping into the Player, or hitting them with a Weapon.
* If the AI has the Flag, it will avoid other Entities while attempting to finish the race.

Video Link

Picture Image

## [Audio](#_8it165j0c5gp)

[Jira Link](http://jira.itcarlow.ie:8080/browse/GJPTEAMD-66)

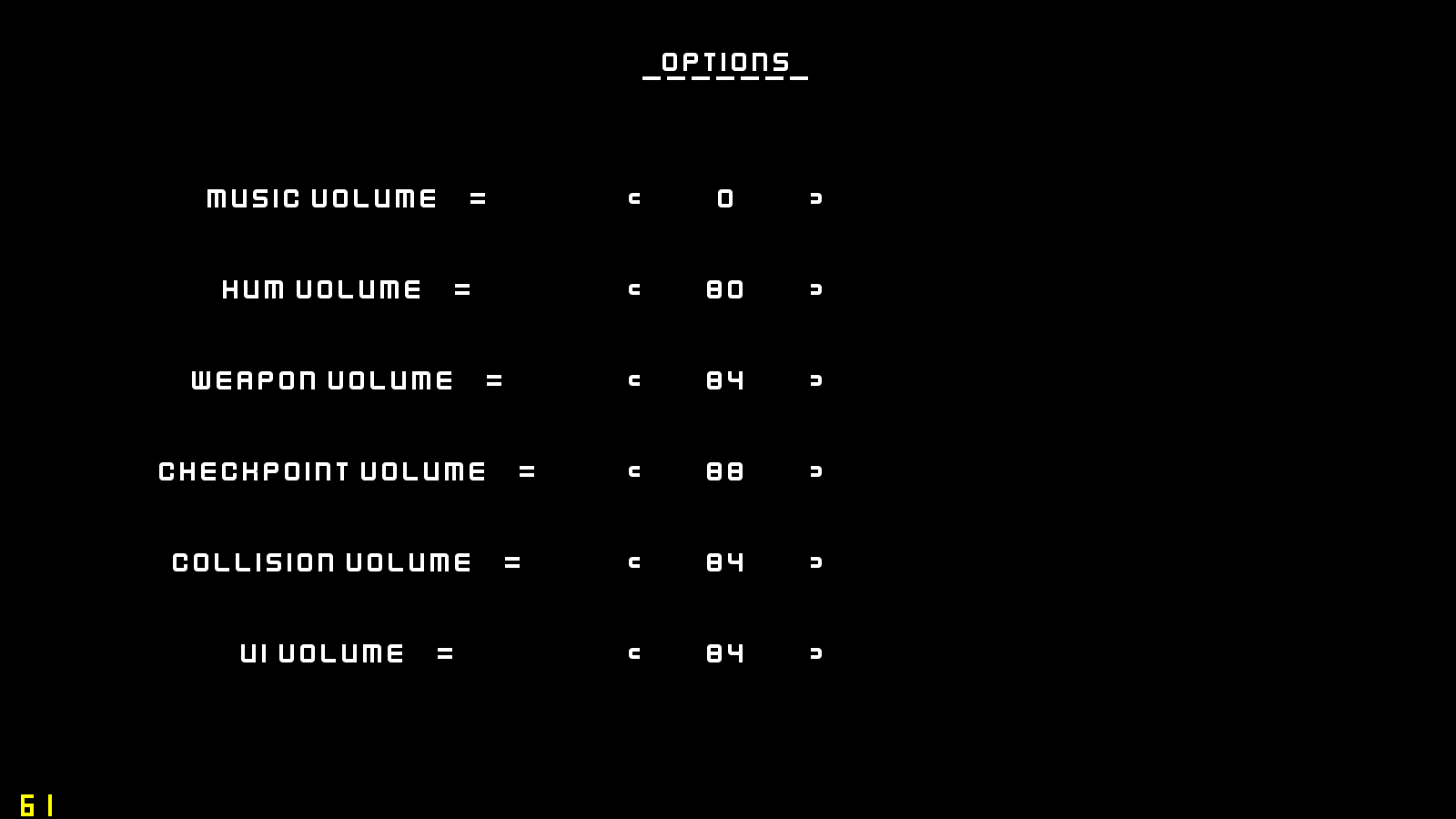
### Conditions of Satisfaction

1. Music will Play continuously throughout the game.
2. A hum Sound plays when the vehicles move.
3. A bullet sound plays when a Weapon is fired.
4. A sound will play indicating a bump occurred.
5. A sound will play to indicate hitting a checkpoint.
6. Sounds volume can be changed in the Options Scene.
7. Music can be turned off in the Options Scene.

Manual Test:

* Music will play when the Application is started, and change by level.
* Move the Player to hear the hum Sounds
* When a Weapon is fired the Bullet Sound will play.
* When an Entity collides with a wall a bump sound will play.
* When an Entity hits a checkpoint while holding a flag, an indicator sound will play.
* Going to the Options scene allows you to increase and decrease volume on each sound by using the mouse.

[Video Link](https://drive.google.com/open?id=0B3ggAWipk3TWbEd2WlZRYlQyeW8)



## [Animation](#_myh2x1ecnaug)

Jira Link

### Conditions of Satisfaction

1. Conditions of satisfaction

Manual Test:

* Manual test

Video Link

Picture Image

## [Level Creation](#_g1u9ce35skn0)

Jira Link

### Conditions of Satisfaction

1. Conditions of satisfaction

Manual Test:

* Manual test

Video Link

Picture Image