## **Sprint 1**

## **Level Generation [RWMPC22-2]**

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-28>

**Description:** Random level generation that generates up or down in a 2d plane

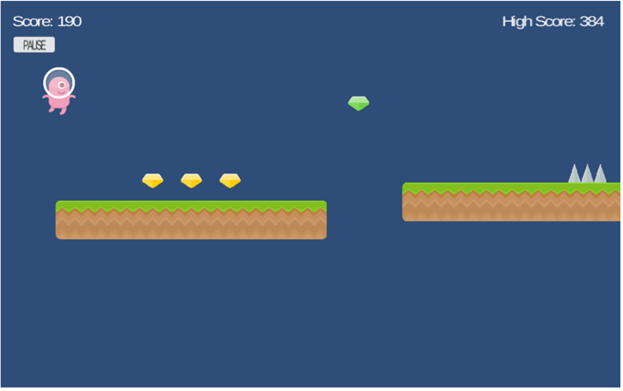


Figure 1: A Example of a random level generation

**Conditions of Satisfaction:**

**1.** Adding chunks to the bottom of the screen. Chunks are square shapes

**2.** Adding tiles onto the chunk at random amounts to simulate a randomly generated map

**3.** When the chunks are off the screen and aren't in use, destroy them to save resources

**4.** Add a hole obstacle by not drawing tiles above a few chunks in succession. Hole will be randomly generated

**5.** record what the top tile of each chunk is to potentially spawn an obstacle on top

## **Jumping Enemy [RWMPC22-3]**

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-30>

**Description:** An enemy that jumps every couple of seconds

**Conditions of Satisfaction:**

1. An enemy that spawns in the level generator

2. An enemy that jumps after X seconds

3. The player loses a life after they hit the enemy

**Steps to Test:**

1. The enemy spawns.

2. The enemy jumps.

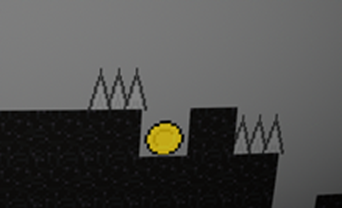
3. Check if enemy is above original y value.

Spawn the obstacles that can kill player [RWMPC22-4]

**Link to**

**JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-4>

**Description:** Obstacle is spawning in front of the player on top of the freshly generated map



**Conditions of Satisfaction:**

1. Obstacles

spawn in the right side of the screen

2. Obstacles

spawn in the same position as the map generates(same Y axis)

3.

Obstacles take out player life if collided

## 

## **Coin Pickup [RWMPC22-20]**

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-20>

**Description:** Player is able to collect coins to gain score

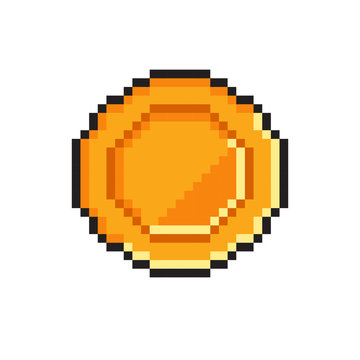
****

Fig. Example coin sprite

**Conditions of Satisfaction:**

1. Create trigger with player.

2. Add Pickup Sprite with collision box

3. Add score

4. Add UI to display Score

## 

## **Sprint 2**

## **Restart Game [RWMPC22-27]**

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-27>

**Description:** Player is able to restart game once all lives are lost.



Fig. representing a restart button

**Conditions of Satisfaction:**

1. Create Restart Button in UI.

2.Make Restart button visible and clickable once player dies

3.On button click the game is restarted

**Steps to Test:**

1. Player is killed
2. Game is restarted

3. Wait 0.1 seconds

4. Check if core is reset to 0

## **Bounding Box [RWMPC22-28]**

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-28>

**Description:** A bounding box that resets the player’s position to the centre of the screen.

**Conditions of Satisfaction:**

1. When the player goes off screen, they respawn at the centre of the screen.
2. The player loses a life when they go off screen.
3. If the player has no lives, they get a game over.

**Steps to Test:**

1. Spawn player

2. Spawn walls

3. Set player to walls

4. Wait 0.1 seconds

5. Check if player position has changed

## **Save and display score [RWMPC22-29]**

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-29>

**Description:** A high score that is displayed to the player at the end of the game



Figure 1: A Example of a high score screen.

**Conditions of Satisfaction:**

1. Add UI for Highscore and display it under the current score

2. Save players highest score for the session.

3. Display Highscore and the current run score after players death

**Steps to Test:**

1. Save high score of any value,
2. Get all saved high scores,
3. Check saved high score against last in list

PlayTesting [RWMPC22-30]

**Link to**

**JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-30>

**Description:** A player presses “L” key and is being sent to the browser form

that will display certain information and will ask player some questions about

his experience in the game

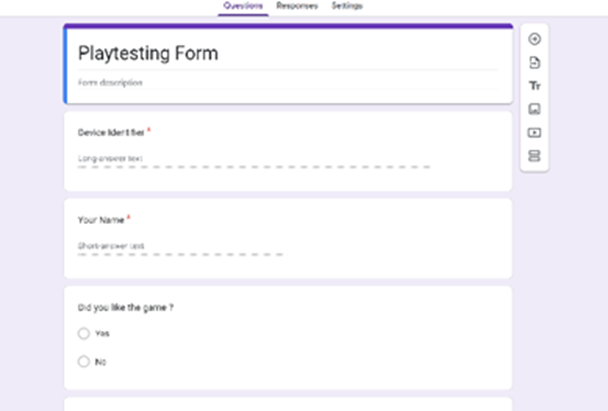


Figure 1: A simple sketch of a the form

**Conditions**

**of Satisfaction:**

1.

When the “L” key is pressed, opens browser and displays the form

2.

Certain details are being recorded such as death counter and sent

to form

3.

Certain details such as death count is being displayed on the form.

Distance Tracker [RWMPC22-40]

**Link to JIRA:**<https://jira.itcarlow.ie/browse/RWMPC22-40>

**Description:** the distance counter increases as the player progresses. This counter will then be used as a part of the final score.

**Conditions of Satisfaction:**

• The Distance counter increases while the player is progressing through the level.

• Distance travelled will be shown on the screen during the level

• counter continuously rises until the player loses.

• Value then gets saved and is used later for the highscore screen

Distanace counter increases over time(2 points)

**Condition(s) of Satisfaction**:

1. The Distance counter increases while the player is progressing through the level.

**Steps to Test:**

• set Distance value as zero.

• start the level.

• Yield to the Unity engine for 0.1 seconds.

• check that the new distance value is greater than the old distance value.

Distanace counter stops increasing upon death(2 points)

**Condition(s) of Satisfaction**:

3. counter continously rises until the player loses.

**Steps to Test:**

• set Distance value as zero.

• set the player lives to 0

• start the level.

• Yield to the Unity engine for 0.1 seconds.

• check that the new distance value is equal to the old distance value.