Point and click project

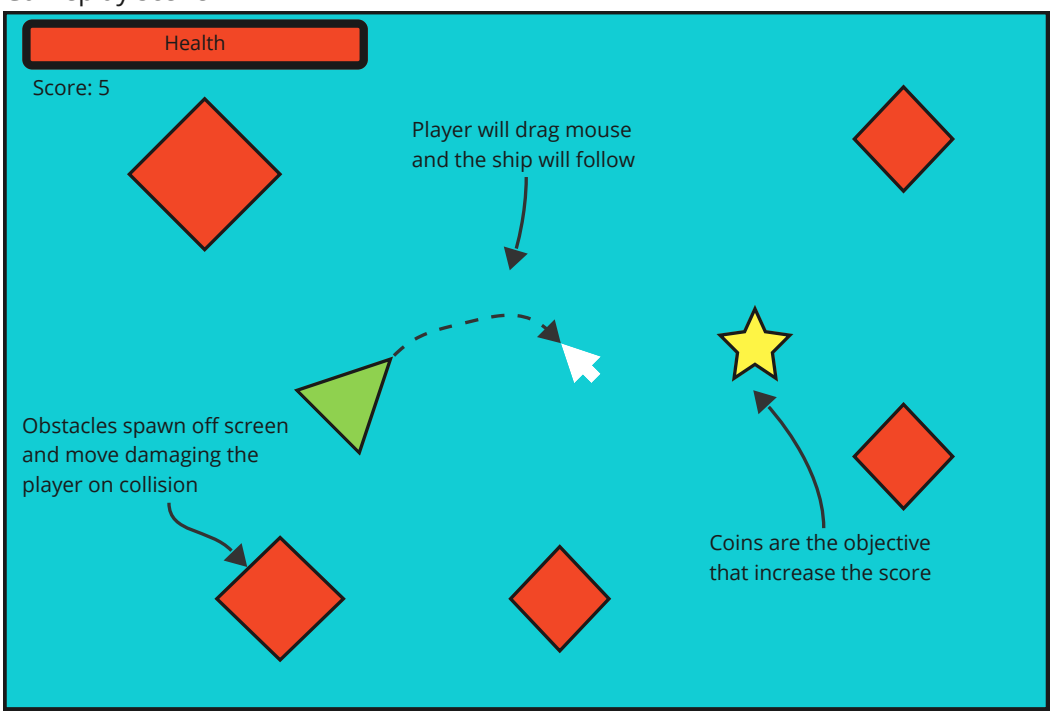
Joe Place 991717262

The player will control a ship that will follow the mouse and go about collecting coins and avoiding rocks
This navigation will happen with the player dragging the mouse around the screen controlling the ship.
The game will feature a title gameplay and game over screen

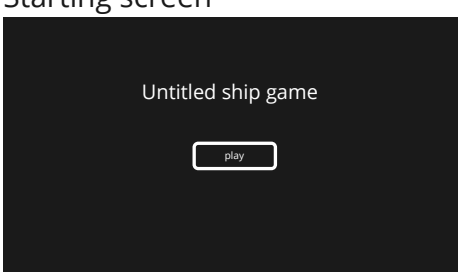
Functionality

The player object will move behind the mouse at a distance constantly
The player will be able to dash increasing there movement and changing their appearance
The coin will increase the player score which will be displayed to the HUD
The obstacles will spawn off screen and slowly move across the screen damaging the player
UI will track health coins and player speed

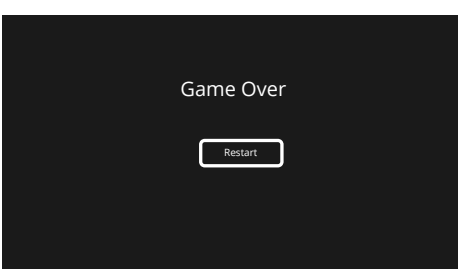
Gameplay scene



Starting screen



Game Over screen



Player Object/Ship

The player object will be a kinematic rigid body with colliders to detect what it collides with

The player will have health that is reduced by obstacles
The player will have an animation clip for being destroyed and damaged

The movement of the player will be handled in **fixedUpdate()** where the equation **Atan2()** will take the mouse position and the ships position and get the angle between the two points using trigonometry. The ship will then rotate towards it and move its position closer towards it.

The player will have to be able to record the mouses position at all times and constantly update. This can be handled in an array of **Vector2** using the position retrieved with **Camera.main.ScreenToWorldPoint();** which will then be added to the list.

The list will need a threshold of when to retrieve a new position to have a delay between its positions. This can be used by the **Distance()** function between the previous recorded position and the mouses current position

The player will have its own take damage function and detect if it has contacted a game object with the tag "Obstacle"
If it collides with the coin object it will call the coins function to increase the score and destroy the coin.

Players movement ability.

When the player presses the mouse key there speed will increase as well as change color using Lerp and an animation curve to add juice to the experience

problem solving
The rotation of the object was detecting the first mouse position. It would then rotate towards the position and ignore any further inputs from the mouse position.

To test i set the object to move with the mouse which discovered that it was rotating properly but to a single index in the list. The solution was changing the index to be the most recent updated position which allowed for fluid movement

Movement issue of ONMOuseDown not responding. This is due to the OnMouseDown interacting with the collider of the object so if the player clicks off of the object it does not activate. Switched to input.keypressed(keycode.mouse(0));

Obstacles

The Obstacles will be kinematic rigid bodies that spawn and destroy themselves after a set time

The obstacle will need to randomize its position and rotation for what angle it will spawn at moving in that direction

the size of the object will also be randomized to have variety of obstacles the player can avoid

The collision of the object will happen only to the player where it will send a message to inflict damage to the player object