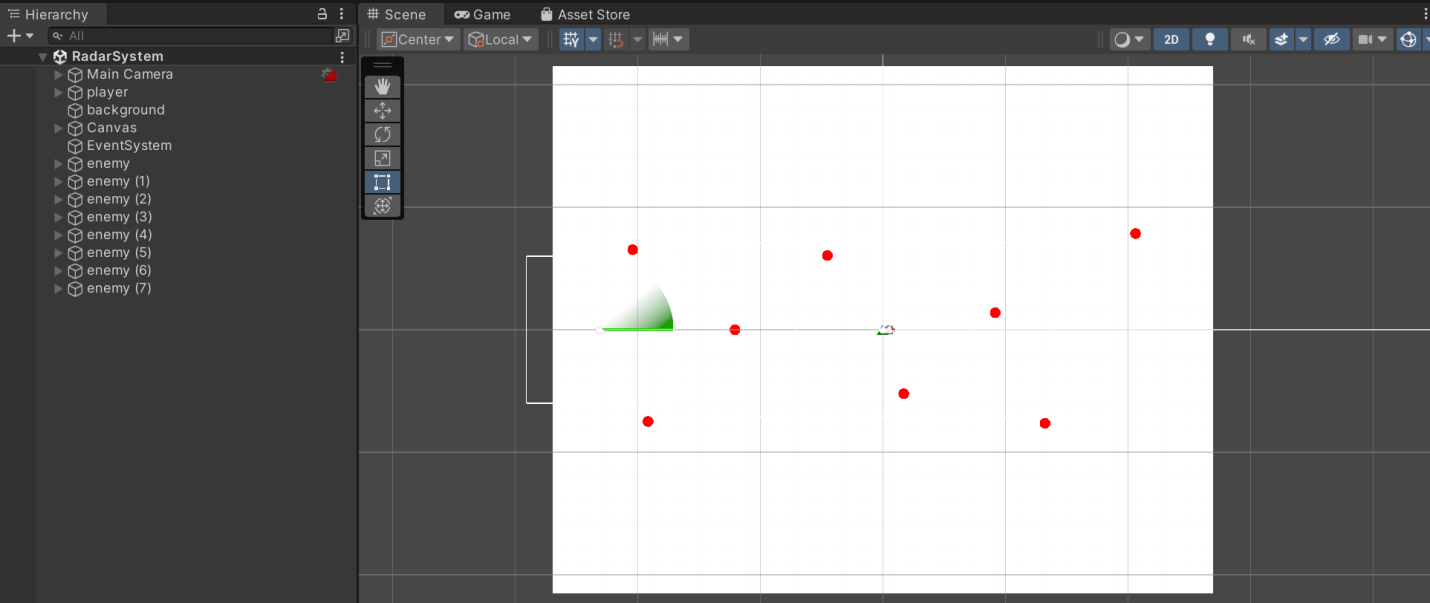
**Dominic Townsend**

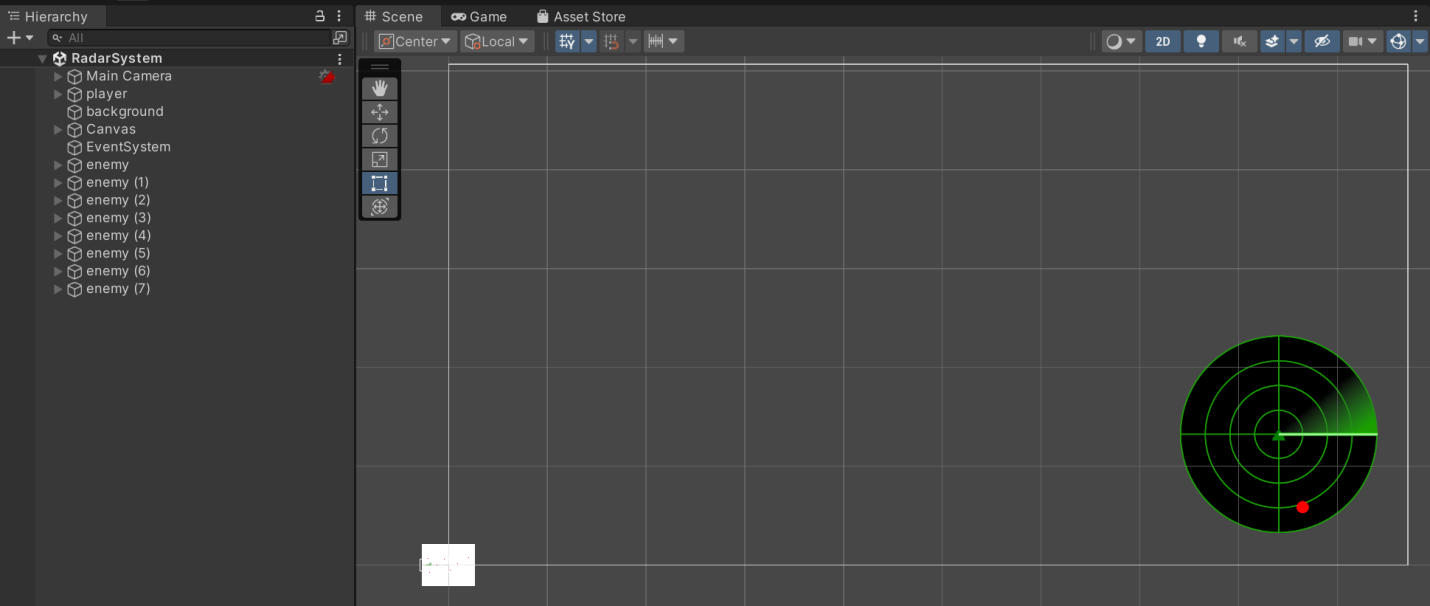
**Brief 1 – Radar**

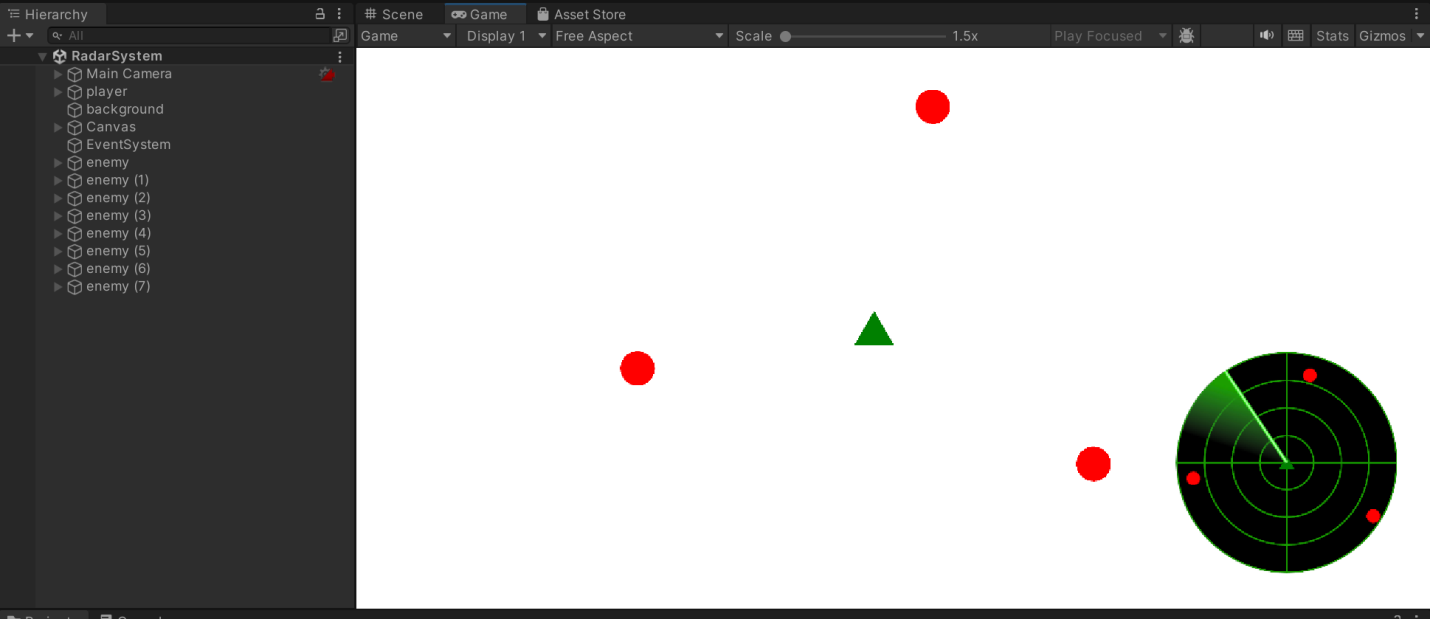
**Documentation**

**Overview:**

This project is designed to mimic a traditional radar system. It uses a secondary camera set as a user interface that can only see icons of the player and enemies, and a background based off a real radar system. A rotating line and trail appears in front of this (using a separate camera to properly play the animation) to tie the whole look together. The player can move in 8 directions (top-down, including diagonals), and is represented by a green triangle. The enemies, who are static in this example scene, are represented by red circles.







**Assets:**

**Radar Assets:**

* radar\_background.png – A circle with green lines that acts as the UI’s background.
* radar\_line.png – The line that sits on top of the background and rotates relative to its centre.
* radar\_trial.png – An addition to the line used stylistically to make it seem more like a traditional radar system.

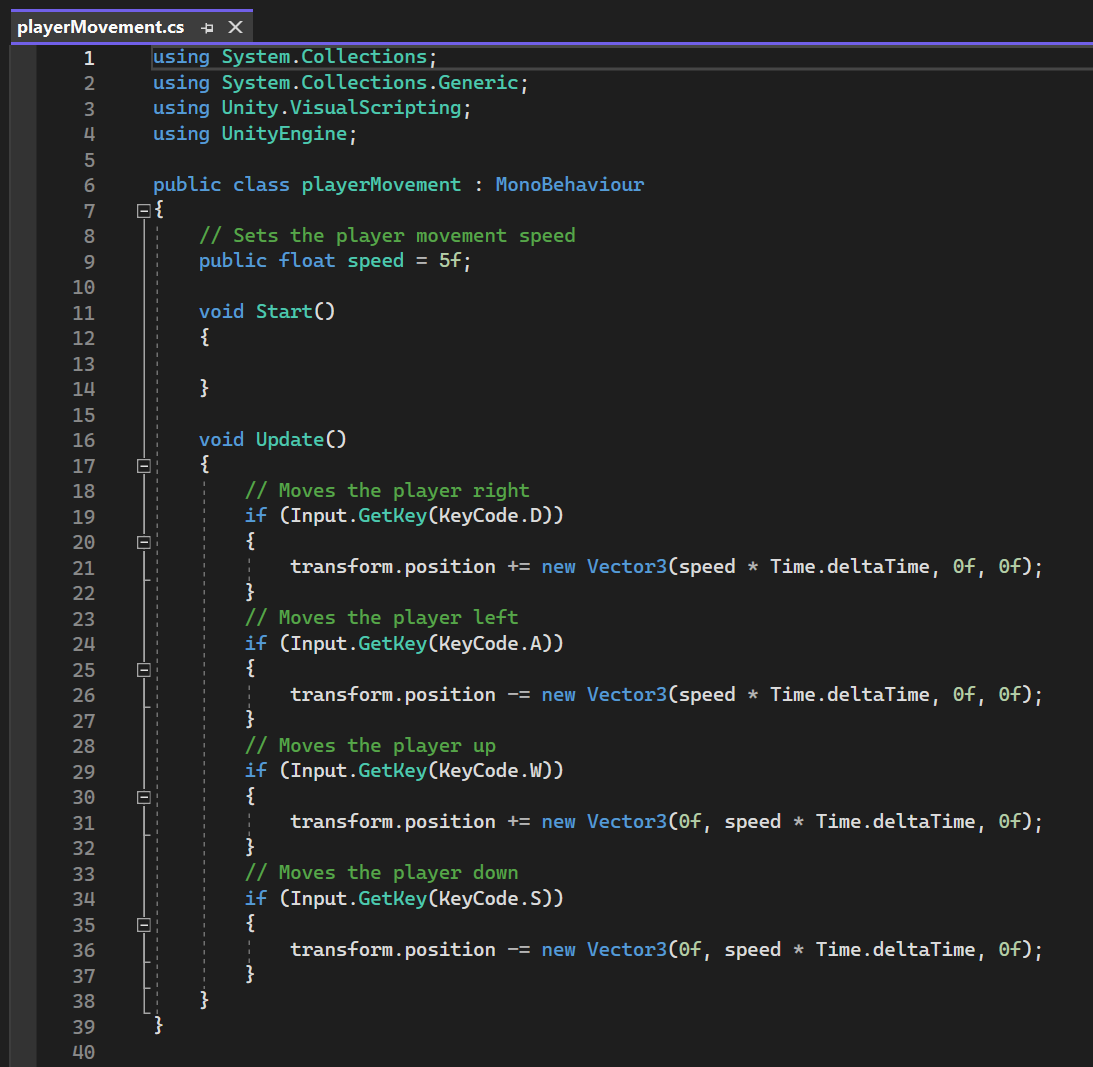
**Character Assets:**

* enemy.png – The red circle used to represent the enemies dotted around the map.
* player.png – The green triangle used to represent the controllable player.

**Scripts:**

**Script 1 – playerMovement.cs:**

This script allows the user to control the player and move it around the map. As with many top-down games, the WASD keys can be used to move the player, up, left, down and right respectively.



**Script 2 – radarContact.cs:**

This script rotates the line and trail of the radar clockwise to simulate what a real radar system would look like.

