

Reflective Statement

What have I achieved?

Developing my Hit a Moving Target brief I have created an AI system capable of shooting moving targets using their velocity and projectile speed to calculate a direction it should fire to successfully hit their target. To test and demonstrate its functionality I constructed a top-down mini-game inside a 2D unity project; importing free tank assets I found online, keeping note of my progress via a production diary to look back on and assist me with future projects.

Starting, I created a 2D Unity scene, making use of square sprite I had generated in the engine I laid out a play area including barriers and obstacles with colliders, colouring the sprites to stand out from each other, then placing down two tanks and their accessory sprites I found online via craftpix.net, setting their Pixel Per Unit values to correctly size them relative to the grid, using empty game objects as pivot points for the guns, finally turning the tanks into prefabs for the Unity Package.

Next, I began coding a script responsible for managing the player movement utilizing user keyboard input, producing a system that resembled tank movement, rotating the tank with horizontal input, and moving it forward and back with vertical input.

Following, I created a script for the enemy AI system, keeping it simple I focused on getting basic aim mechanics working, first calculating the direction to aim towards using the player's position, and employing a function called in the update method to keep its aim, resulting in the enemy tank's gun following the player tank's movements, but continuing to keep aim with obstacles in the line of sight. This led me to incorporate further intelligence into the AI using a function to check if there was a clear line of sight to the player, thus preventing the enemy tank from persistently aiming in the player's direction as desired, incorporating smoothing to the gun's turn behaviour to prevent it appearing to suddenly change aim direction once it gains sight of the player.

After this, I imported a bullet sprite provided in the free asset pack into the scene to set up as projectile prefab for the tanks to shoot, producing and attaching a script responsible for handling behaviour on its collision, using this to trigger an explosion effect animation again I found in the free asset pack when it hits a collider.

Leading on, I modified the tank prefabs adding on empty game objects serving as positions to shoot projectiles from, then I included some more attributes to the enemy AI script for managing the projectile speed and fire rate, integrating functionality to fire projectiles aiming with a clear line of sight to the target player.

Finally, to meet the brief requirements I needed to advance my AI system to give it the capability to hit a moving target. Searching online I found a code snippet of a function capable of predicting the future position of a moving target; using the target's velocity and its bullet projectile speed to make its calculation. After tweaking the snippet to be compatible with my code the AI system, at last, met the brief requirements; successfully hitting a moving target.

As a bonus I threw in extra elements to my project, I created and attached another script to the player tank to introduce aiming and shooting from the player with controls utilizing mouse input, recycling code I had already developed in my AI script. Furthermore, I combined a health system into the mix, I began constructing a health bar with again the square sprite I produced in unity; later joining it to my tank prefabs, formed a new script to administer the health system with properties to manage total health. To make use of the added health system I modified the bullet script to deal damage to an object that it can find the health system assigned to, i.e., the enemy and player tanks. Moreover, I included consumables the player could pick up by driving over to provide them with a health bonus, achieving this by using another sprite supplied in the free asset pack, attaching a box collider as a trigger, establishing a further script to handle the trigger behaviour featuring properties to control the minimum and maximum health the health bonus would provide.

In addition to that, I obtained a second free asset pack from craftpix.net containing tiles and sprites I applied to my scene to spruce it up a little, including a football sprite I imported to bounce around the play area and block incoming projectiles.

What have I learnt?

Throughout building this brief I have discovered additional transform and Vector2 methods and attributes that I can use to direct game objects axis; for example, using 'transform.up' and a calculated direction vector to aim an object at a target, and making use of empty game objects to scale child objects from specific points, e.g., a health bar scaling down from right to left. A deeper understanding of Rigidbody, for instance, altering its traits to work in a top-down setting, and finally knowledge of code used to hit a moving target; utilizing velocity and projectile speed to calculate an effective aim.

What would I aim to do differently next time?

If I were to reproduce this brief or make something similar in the future, I could aim to make the system and its scripts more universal to work in other projects, and/or with auxiliary scripts, namely opening possibilities to perform custom actions, to give an example, when a tank is at low health or destroyed, or a when an AI has caught line of sight of a target. In general, I would like to produce a level-based game next time to demonstrate further capabilities.

How has what I have created assisted me through my studies?

To conclude, the experience I have obtained and the production diary I have written will be put to good use in my future studies, specifically projects involving C# and unity components I have used, and systems I have established and throughout this brief.