**Technical Assignment: Drone fight gameplay**

**Technical Architecture**

**Drone** **Turret** **Bullet**

**BulletRewind**  Records bullet’s position and rotation. References BulletBase class for Data

**TurretRewind**  Records turret’s health and alive status. References Turret class for Data

**DroneHealthRewind** Records and rewinds the health value of Drone. References the DroneHealthController for data

**RewindManager**  Exposes the InputKey, duration and cooldown for rewind. Has actions for Rewind key press and release

**BaseRewind - Abstract class** Has Rewind() and Record() as abstract methods. Handles the rewind and record in the fixed Update

**BulletData –** The bullet data holds various bullet’s damage, Speed, TurnSpeed & Duration in different scriptableObjects. This reduces unnecessary data duplicates in scene

**TurretMissile –** Enemy Turret’s bullet

**HomingMissile –** drone’s sub-weapon bullet

**Bullet –** drone’s primary bullet

**BulletBase – Abstract class** Base for all bullets and missiles, Handles most of the bullet functionalities. Has an abstract method which is implemented by inherited classes called InitializeBullet()

**Turret -**  Handles Initializing and Destroying of Turret. Talks to BulletBarrel class attched to barrel to shoot TurretMissiles. Extends from IHealth and handles the current health of turret

**TurretScanner -** Checks if player is in range and handles player lockdown and scan delay. Talks back to Turret class when Drone is scanned and locked for shooting

**IHealth - Interface** Implements TakeDamage(float) and OnHealthDepleted()

**DroneHealthController**  Extends from IHealth, Manages the current health of Drone

**DroneMovementController**Handles drone movement Inputs, movement, turn and dashing. Also takes care of Drone’s movement sound

**DroneShootingController** Handles drone shoot Inputs, Scan for sub weapon and Instructs the **BulletBarrels Class** to shoot (Instantiate bullets)

**DroneCameraController** Handles drone Mouse Axis movements and Camera rotation

**Rewind** **functionality**

**Managers & HUDs**

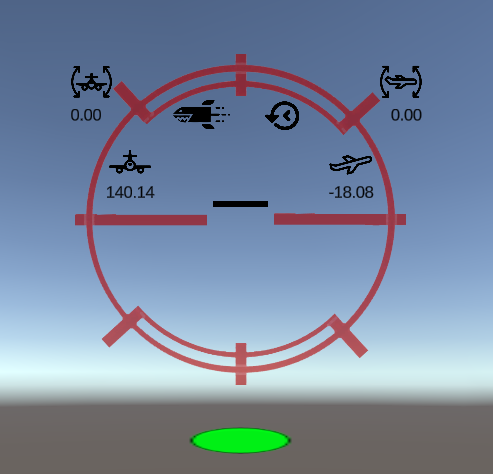
**UIDroneHUD**  - Shows the drone related data such as Health, altitude, pitch, roll, camera rotation, drone missile cooldown and rewind cooldown. Get all the needed data from DoneObject, DroneShootingController, RewindManager and DroneHealthController

**UIGameHUD**  - GameManager uses the UIGameHUD methods. Handles the turrent count, restart and Countdown functionalitys

**ObjectPoolManager**  - Instantiates and holds a list of objects that can be Spawned in game on request and added back to pool. All Bullets used are pooled initially

**LevelCreator**  - A grid based levelcreator that has a texture map and Color to map values. The pixel color in map texture is used to instantiate game objects

**GameManager**  - Handles the game-start and game-end scenario. Calls game start based on the ready status of other manager classes. Handles the GameHUD class

**Explaining Drone HUD: **

Rewind Ready Status

MissileReady Status

Drone’s Health

Cam tilt value

Altitude Value

Roll Value

Pitch Value

**Known Issues / bugs :**

1. The homing missile launches itself in the barrel’s forward with a force value(exposed), stays up in the air for 1second(exposed) and then seeks the target with a Turn speed and move speed(exposed). This seemed like a better option than directly going to the target in a beizier path.
2. The homing missile target(Turret) selected using laser pointer behind a building block mostly would be missed because of the homing missile hitting the block and destroying before reaching the Turret.
3. The PrimaryBullet (Bullet.Prefab) has an exposed bool called ‘TargetMousePointer’. This can be used to set the bullet’s target as a raycast point from the cursor instead of shooting straight from the barrel’s forward vector. This feature sometimes shoots the bullets in unexpected angles.
4. Although the Health rewind functionality works in the provided build. It sometimes does not work in the editor. Removing the ‘RewindManager’ from scene and creating a new gameobject with RewindManager script attached seems to fix this issue oddly.
5. The Rewind ‘Bullets’ and ‘TurretMissles’ sometimes stay in scene without moving after rewind complete. This eventually destroys itself after the alive duration is passed.
6. There is a small delay in Drone reacting to the input after using Dash functionality.