Scripting Documentation

Summary

AircraftMovement

Place it on the player Prefab, it handle the player movement,
members:
-rotationSpeed: is the speed at the player can rotate (default: 30)
-maxAngle: the angle max that player can rotate (default: 90)
-tiltAngle: the angle at the player roll(rotate on X) when the input axis horizontal is at is max (default: 15)

AreaBoundaries

Place one on the scene it will define the GameArea as in the GameDesign Document about Enemy Spawn, It will draw a cube but it will be visible only in the editor
members:
-EditorColor
-m_minX / minX
-m_maxX / maxX
-m_minZ / minZ
-m_maxZ / maxZ

    methods:
+GetCenter: Return a position with a X between min and max and a Z equal to minZ
+GetWidth: Return the width of the area

Bullet

Place it on the prefab of a bullet, it will be call by the weapon
members:
-LifeTime : the delay before the autodestruction of the bullet (default: 5)

methods:
+Shoot(float bulletSpeed, float bulletDamages, string targetTag): Shoot the bullet with and set trhe speed, damages and the target tag

CameraFollowTarget

Place it on the Main Camera, it will allow this camera to follow the player movement
members:
-pitch = (default: 30)
-distance: (default: 20)
-cameraSpeed: (default: 1)
-rotationSpeed: (default: 2)

Enemy

Place It on an Enemy Prefab, this will allow to create an enemy capable to move toward player and inflict damage, also it require Health Script to handle the life of this enemy
members:
-Speed = 5: This is the movement Speed of the enemy,
-Damages = 5: This is the amount of damage it will do if it touche the player
-DeathParticle: the particle to play on death (optional)

EnemySpawn

This Script will makes randomly spawn enemies on the Spawn Area (check the document about Spawn)
members:
-spawnables: The list of Enemies prefab with their probability (see RandomSelector)
-SpawnDistance = 50: the distance of the spawn area from the player
-SpawnFrequency = 2: the coolDown between two spawn
methods:
+StartSpawn: Start the spawning of the enemies, call it at the start of the game
+StopSpawn: Stop the spawning of the enemies, call it at end the of the game in case of gameover or win

Health

This Script allows to take damage, its used by the playerController script and the Enemy Script

members:

-maxHealth = 100: the health max of the object

-onDamaged: an event that is call when object take damage

-onHealed: an event that is call when object is healed

-onDie: an event that is call when the life point are at Zero

methods:

+GetLife: Return the current life of this Entity

+Kill: Kill this Entity

+TakeDamage(float damages): Decrease the life with an amount of damages

+Heal(float heal): Increase the life with an amount of heal

HurtOnTouch

This Script allows to make damages to other Entity

members:

-onHurt: an event that is call when this object collide with an Entity (ex: Player or Enemy) and have made damages

methods:

+SetTarget(string targetTag): Define wich kind of Entity this object can hurt with a Tag

+SetTarget(float damages): Define how much damage this object can make

LootOnDie

This script allow the object to drop something when it die, the object should have a Health Script

members:

-probabilityToDrop: the probability to drop something on death

-Lootables: the list of possible loot (see RandomSelector)

Pickup

This script manage a single pickup and allow the player to catch it, the pickup will move toward the player and call the onPickedUp event when it will touch the player, its use by UpgradePickup script

members:

-onPickedUp:

PlayerController

This script should help to manage all the script on the player, but for now it can only help to detect when the player die and trigger a gameover,

edit: now it also place the player at a position on the AreaBoundaries

RandomSelector

This Script manage a list of selectable objet and their probatility ot's now used by EnemySpawn and LootonDie

members:

-onHurt: an event that is call when this object collide with an Entity (ex: Player or Enemy) and have made damages

methods:

+SetTarget(string targetTag): Define wich kind of Entity this object can hurt with a Tag

+SetTarget(float damages): Define how much damage this object can make

Selectable

A simple struct to represent a Selectable used by RandomSelector

members:

   -gameObject: a prefab to select

   -probability : the probability for this Object to be selected

Upgrade

Its represent an Upgrade Object

-upgradeName: use it if you need to differentiate an upgrade, for Expanple to display it separatly in the UI

members:

-formula: type of formula to use Flat will simply add value a,d PErcent will add a percent equal to value

-upgradeType: define wich stat will be upgraded

-value:

-temporary: define if the upgrade is permanent or temporary

-duration: if the upgrade is temporary, this define the duration of the power up

UpgradeHandler

Manage the upgrades and the stats of the Weapon

members:

-onStatsChange: An event that will be trigger when the stat change, you can use it to detect when update your UI).

methods:

+UpgradeWeapons(List upgrades): Call this method with a list of upgrades that you want to add on the Weapon

+GetCurrentStat: Return the current stat of the Weapon (see WeaponsStat)

+GetTemporaryUpgrades: Return the list of all the Temporary Upgrades on the Weapon

+GetPermanentUpgrades: Return the list of all the Permanent Upgrades on the Weapon

UpgradePickup

It can contains a list of upgrade that will be apply on Picked up, it require a pickup script

members:

-UpgradeList: the list of upgrades to add

Weapons

Handle the Weapons of the aircraft

members:

-OutPuts where the bullet will out

-BulletPrefab

-InitialStats the weaponsStats at the start of the game

WeaponStats

members:

-coolDown

-bulletSpeed

-bulletDamages