LOGBOOK 8

Team 1 : Hand Ball

Artist :

* Melanjutkan pekerjaan membuat elemen-elemen UI yang lain
* Hasil ada di github

Programmer :

Enemies Behavior

using System.Collections;

using System.Collections.Generic;

using System.Linq;

using UnityEngine;

public class EnemiesBehavior : MonoBehaviour {

//GameManager gameManager;

private float lastShotTime;

GameObject target;

public GameObject bulletObject;

BulletPrefabs bulletPrefab;

EnemiesPrefabs enemiesPrefabs;

float interval;

float healthPoint;

public float EnemyHP

{

get { return healthPoint; }

set { healthPoint = value; }

}

// Use this for initialization

void Start () {

//gameManager = GameObject.Find("Game Manager").GetComponent<GameManager>();

target = GameObject.Find("Player");

enemiesPrefabs = gameObject.GetComponent<EnemiesPrefabs>();

interval = enemiesPrefabs.CurrentType.fireRate;

bulletPrefab = enemiesPrefabs.CurrentType.bullets;

//lastShotTime = Time.time;

healthPoint = enemiesPrefabs.CurrentType.health;

InvokeRepeating("Shoot", 1f, interval);

}

// Update is called once per frame

void Update () {

Vector3 direction = gameObject.transform.position - target.transform.position;

// gameObject.transform.rotation = Quaternion.AngleAxis(Mathf.Atan2(direction.y, direction.x) \* 180 / Mathf.PI,

// new Vector3(0, 0, 1));

if (EnemyHP <= 0)

{

GameManager.listEnemies.Remove(this.gameObject);

Destroy(this.gameObject);

}

}

void Shoot ()

{

Collider2D target = this.target.GetComponent<Collider2D>();

Vector3 startPosition = gameObject.transform.position;

Vector3 targetPosition = target.transform.position;

startPosition.z = bulletPrefab.transform.position.z;

targetPosition.z = bulletPrefab.transform.position.z;

GameObject newBullet = (GameObject)Instantiate(bulletObject);

newBullet.GetComponent<BulletPrefabs>().setCurrentType(bulletPercentage()); ;

newBullet.transform.position = startPosition;

BulletBehavior bulletComp = newBullet.GetComponent<BulletBehavior>();

bulletComp.target = target.gameObject;

bulletComp.startPosition = startPosition;

bulletComp.targetPosition = targetPosition;

bulletComp.circleCollider2D = newBullet.GetComponent<CircleCollider2D>();

bulletComp.circleCollider2D.radius = bulletObject.GetComponent<BulletPrefabs>().CurrentType.radius;

}

int bulletPercentage()

{

List<float> mBulletA = enemiesPrefabs.CurrentType.bulletsPercentage.ToList<float>();

mBulletA.Sort();

int index = 0;

float random = Random.value;

for (int i = 0; i < mBulletA.Count; i++)

{

if (random > mBulletA[i]/100)

index = i;

}

return index;

}

private void OnCollisionEnter2D(Collision2D collision)

{

if (collision.gameObject.tag == "Bullets")

{

if (collision.gameObject.GetComponent<BulletBehavior>().canDamage)

{

healthPoint -= collision.gameObject.GetComponent<BulletBehavior>().damage;

Destroy(collision.gameObject);

}

}

}

}

Game Design :

* Game Stage Flow Details  
  File in : ../Game Design Document/Game Stage Design – Chapter 1
* Game Monetization Plan

File in : ../Game Design Document/Game Monetization Model

To do :

* Chapter 1 Stages progressive story.
* Game Monetization Details.
* Game Publishing Plan.