using System.Collections;

using System.Collections.Generic;

using UnityEngine;

using UnityEngine.UI;

public class BaseManager : MonoBehaviour

{

/// <summary>

/// This script is to manage the underlying behaviour of the main bases.

/// It includes the health, energy generation, and damage detection.

/// </summary>

public Slider healthBar; //Health bar for the Base

public Image healthBarFill; //Fill are on base health bar.

public GameObject energyBar; //Player's energy bar on UI

public bool isPlayer2; //This bool will be set to true everytime player2 spawns something

public float energyGenerationTime; //This float controls how long the base waits for before adding the regen amount to the players Energy UI bar.

public float energyGenerationAmount; //This float controls how much to increment the players Energy UI bar by

public int health;

private bool done = false;

private bool once = false;

private bool energy = false;

public float minionSpawnTime;

public GameObject player1Minion;

public GameObject player2Minion;

public Transform spawnPointP1;

public Transform spawnPointP2;

// Start is called before the first frame update

void Start()

{

healthBar.value = health;

healthBar.maxValue = health;

healthBar.minValue = 0;

}

// Update is called once per frame

void Update()

{

if(!once)

{

StartCoroutine(generateMinion());

once = true;

}

if(!energy)

{

StartCoroutine(generateEnergy());

energy = true;

}

if(healthBar.value==0 && !done)

{

done = true;

Debug.Log(this.gameObject.tag + " base destroyed!");

this.gameObject.SetActive(false);

once = true;

energy = true;

//call win/lose screen function or script.

}

}

public void OnTriggerEnter(Collider other)

{

if (isPlayer2)

{

if (other.tag == "Damage"||other.tag=="Player1")

{

Debug.Log("Hit by P1!");

}

}

else

{

if(other.tag == "Damage"||other.tag=="Player2")

{

Debug.Log("Hit by P2!");

}

}

}

private IEnumerator generateEnergy()

{

yield return new WaitForSeconds(energyGenerationTime);

//energyBar.addValue(energyGenerationAmount) The main UI system should have a helper function that increments the fed in amount to its slider.

energy = false;

}

private IEnumerator generateMinion()

{

yield return new WaitForSeconds(minionSpawnTime);

if (this.gameObject.tag == "Player1")

{

GameObject minion = GameObject.Instantiate(player1Minion,spawnPointP1);

minion.tag = "Player1";

}

else

{

GameObject minion = GameObject.Instantiate(player2Minion,spawnPointP2);

minion.tag = "Player2";

}

once = false;

}

public void TakeDamage(float damage)

{

Debug.Log(this.tag + " hit for " + damage);

healthBar.value -= damage;

if(healthBar.value>(0.65\*health))

{

healthBarFill.color = new Color(0,255f,28f,255f);

}

if (healthBar.value<=(0.65\*health))

{

healthBarFill.color =new Color(255,190,0,255);

}

if(healthBar.value<=(.30\*health))

{

healthBarFill.color = new Color(255, 95, 0, 255);

}

}

}