

T.C.
TRAKYA ÜNİVERSİTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

2D PLATFORM SAVAŞ OYUNU

PROJE GRUBU

1221602078 Eralp ÇİNKİLİÇ
1221602014 Özgür KÖSE
1221602074 Nurullah ÖZGENÇ
1221602060 Mustafa KANER
1221602094 Murat KARABEYAZ

EDİRNE-2024

İÇİNDEKİLER

İÇİNDEKİLER	ii
BÖLÜM 1.....	2
GİRİŞ	2
BÖLÜM 2.....	4
PROJEDE KULLANILAN TEKNOLOJİLER.....	4
BÖLÜM 3.....	9
PROJE YAPISI.....	9

BÖLÜM 1

GİRİŞ

İnsanların eğlenmesini, kafasını dağıtmasını ve vakit öldürmesini sağlamak amacıyla tasarlanmış bir projedir.

Projemizde üç farklı seviye bulunmaktadır. Bunun yanı sıra 4 farklı kontrol edilebilir oyuncu; 4 farklı düşman da bulunmakta bu düşmanlar kalkanlı, okçu, şifacı, çağrıcı düşman, çağrıcı düşmanın çağırdığı ateş topu ve bir adet de güçlü Düşman bulunmaktadır. Oyunumuzda güçlendirici nesneler var bunların yanı sıra kullanıcı arayüzü kısmında oyun giriş ekranı, seviye seçme ekranı, ölüm ekranı, can barı, karakter seçim ekranı bulunmaktadır.

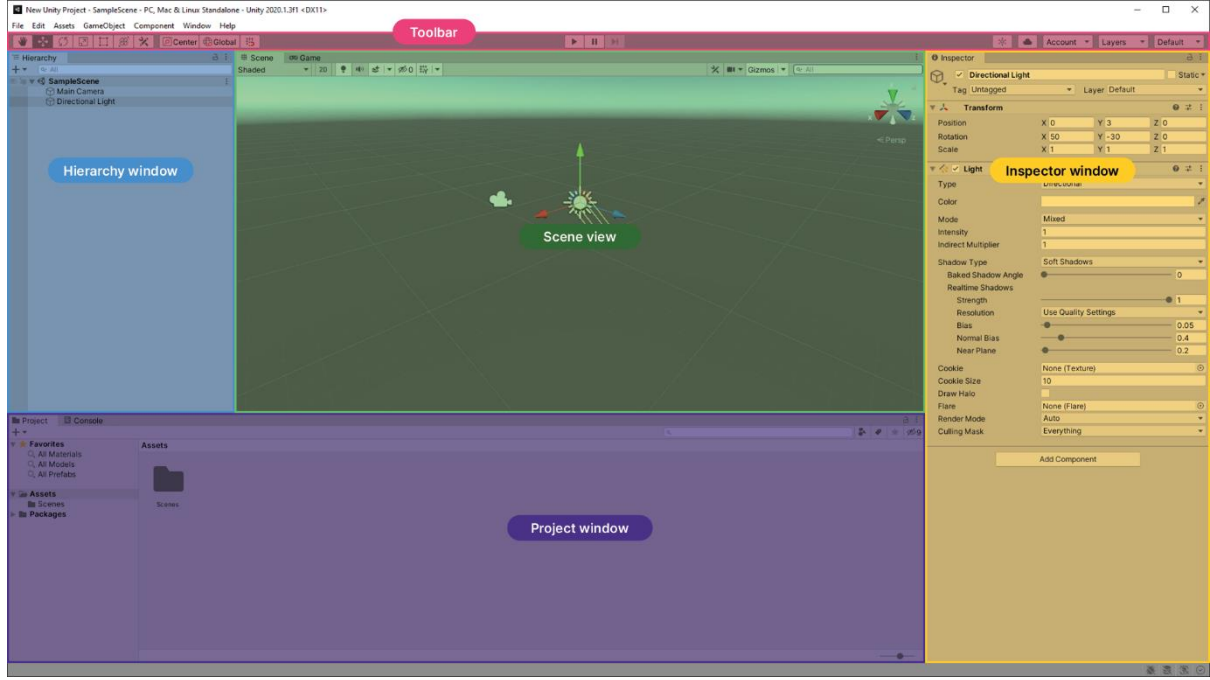
Projemizin çalıştığı sistem Windows' dur.

Projemiz yediden yetmişe eğlenmeyi bilen her kesim için uygundur. Kullanıcı sayısı bir olup kullanıcımız tarafından oyuncumuzun hareketi ve atakları kontrol edilir.

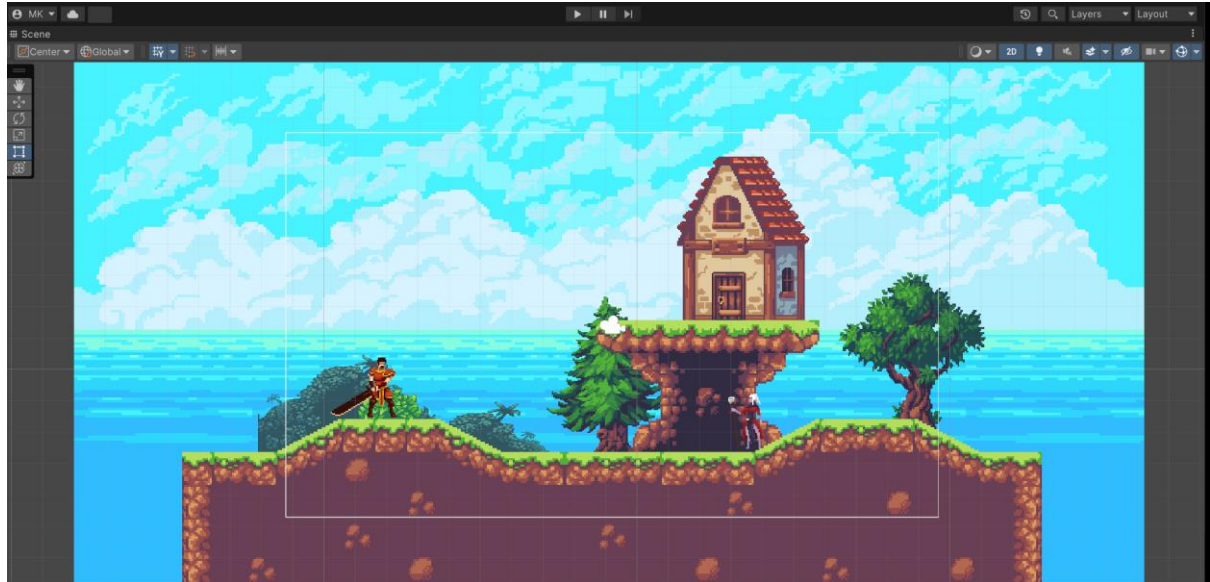
BÖLÜM 2

PROJEDE KULLANILAN ARAÇLAR TEKNOLOJİLER

1)Unity Editor Ve Scene View: Unity Editor, oyun sahnelerini oluşturmak ve düzenlemek için kullanılır. Scene View, oyununuzu düzenlerken nesneleri yerleştirmek için kullanılır.

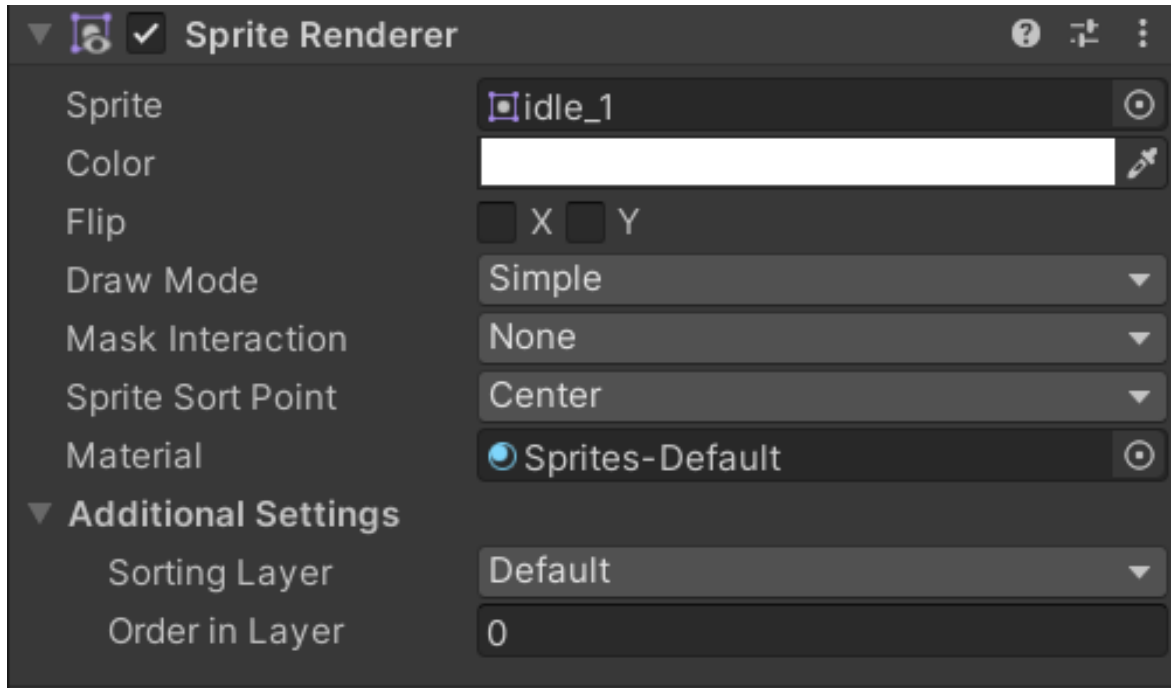


Şekil 1.1. Unity Editor Arayüzü



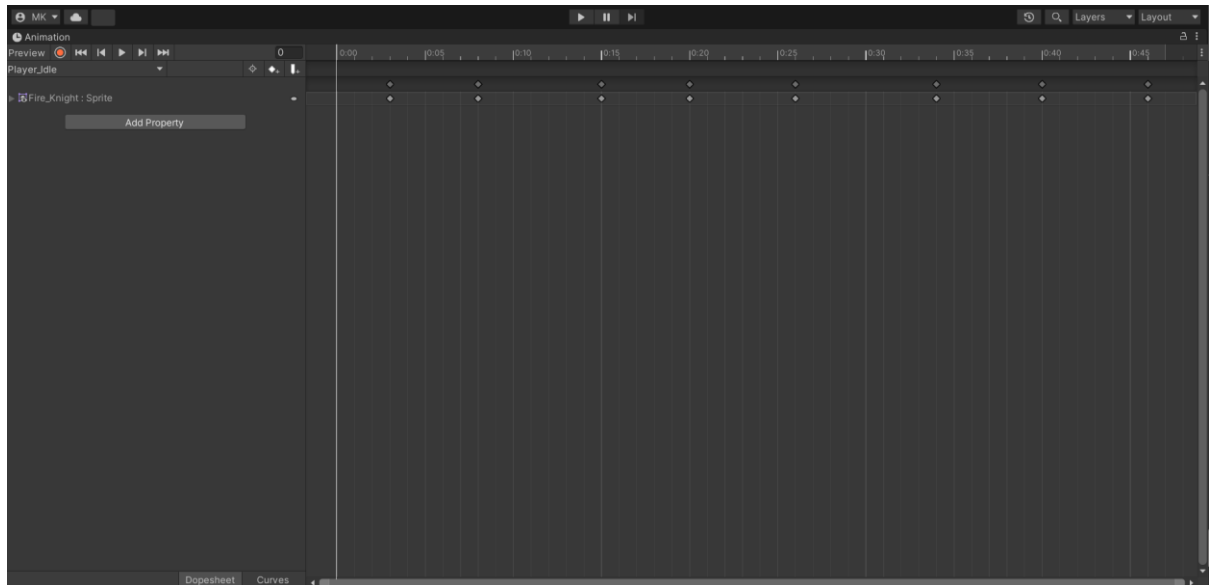
Şekil 1.2. Scene View

2)Sprite Renderer: PNG formatındaki tasarımların unity içerisinde oyun objesi haline getirilerek sahnedeki yerinin ve boyutunun ayarlanmasını sağlıyor.

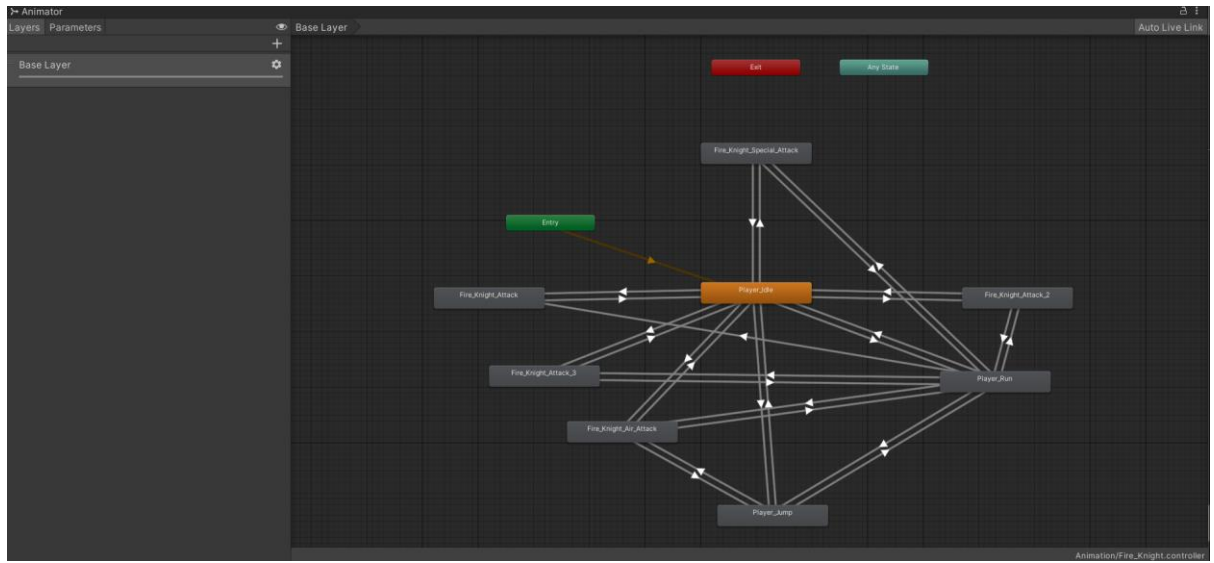


Şekil 2.1. Sprite Renderer Ekranı

3)Animator Ve Animation: Animation, Animasyonları oluşturmak ve kontrol etmek amacıyla kullanılır. Animator ise nesne animasyonlarının parametrelerle kod tarafından kontrolünü sağlar.

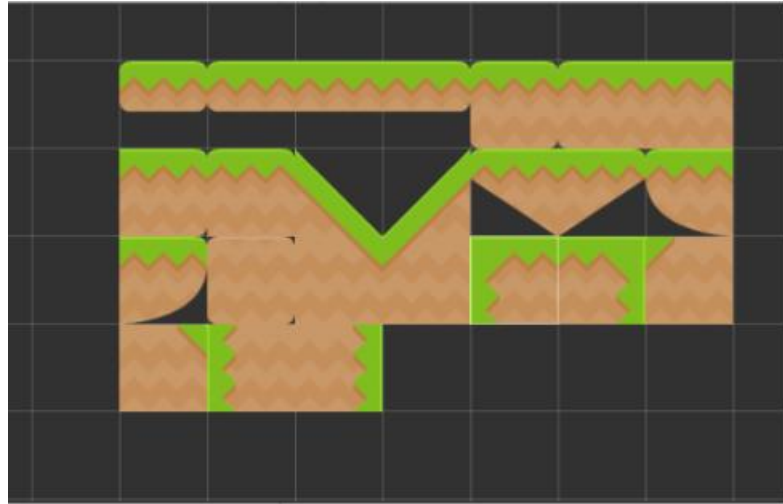


Şekil.3.1. Animation Ekranı



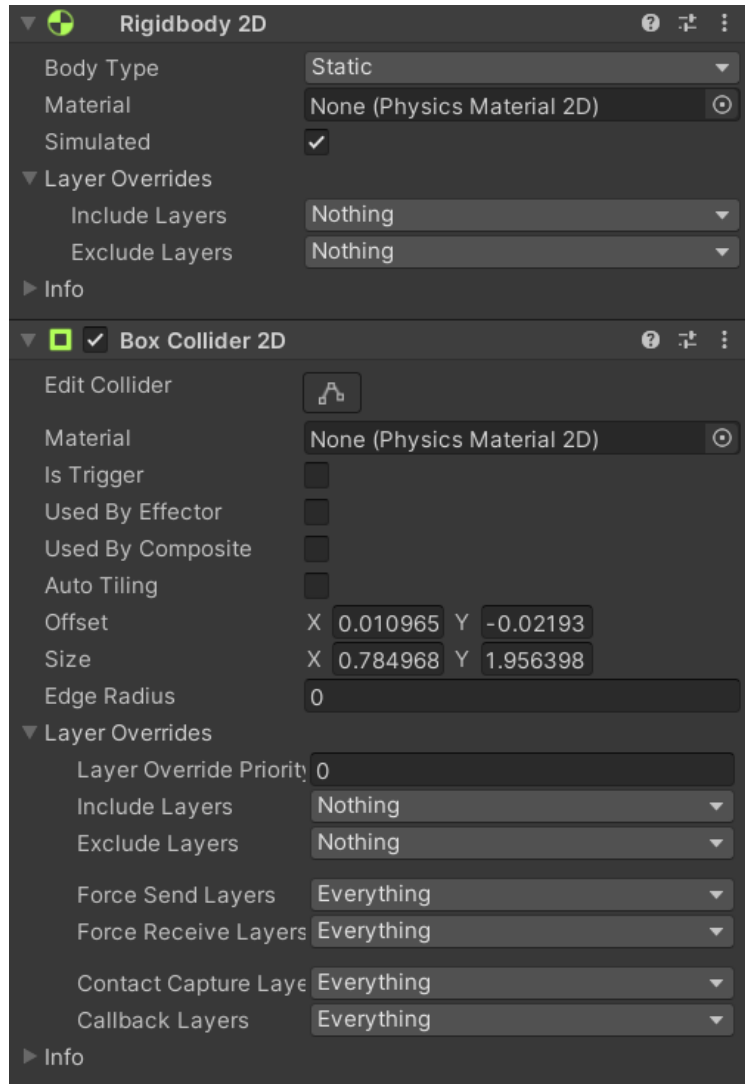
Şekil.3.2. Animator Ekranı

4) **Tilemap:** Haritaları kolayca oluşturmaya yarar



Şekil.4.1. Tilemap

5)Physics 2D: Unity, 2D fizik motorunu içerir. Rigidbody2D, Collider2D gibi fiziksel bileşenleri nesnelere kazandırır.



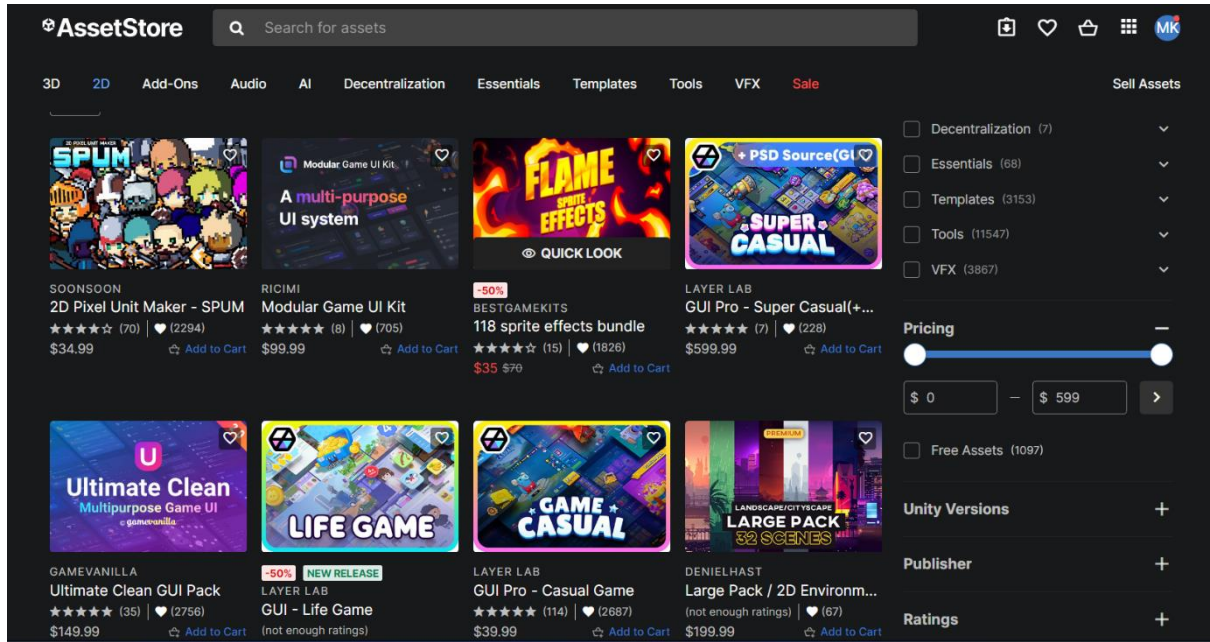
Şekil.5.1. Physics2D

6) **C# Scriptleri:** Oyunun mekaniklerini ve kontrollerini eklememizi sağlar.



Şekil.6.1. Scripts

7) **Unity Asset Store:** Farklı ihtiyaçlar için hazır karakter, arka plan vb. bulabileceğimiz bir mağazadır.



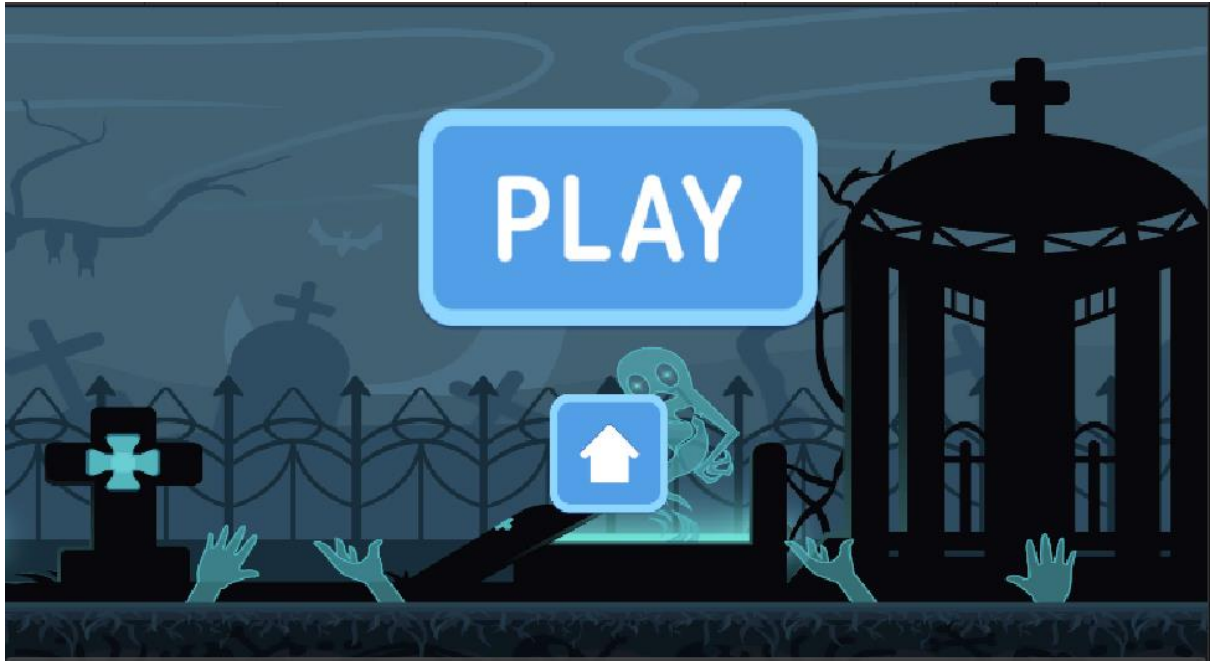
Şekil.7.1. Assets Store

BÖLÜM 3

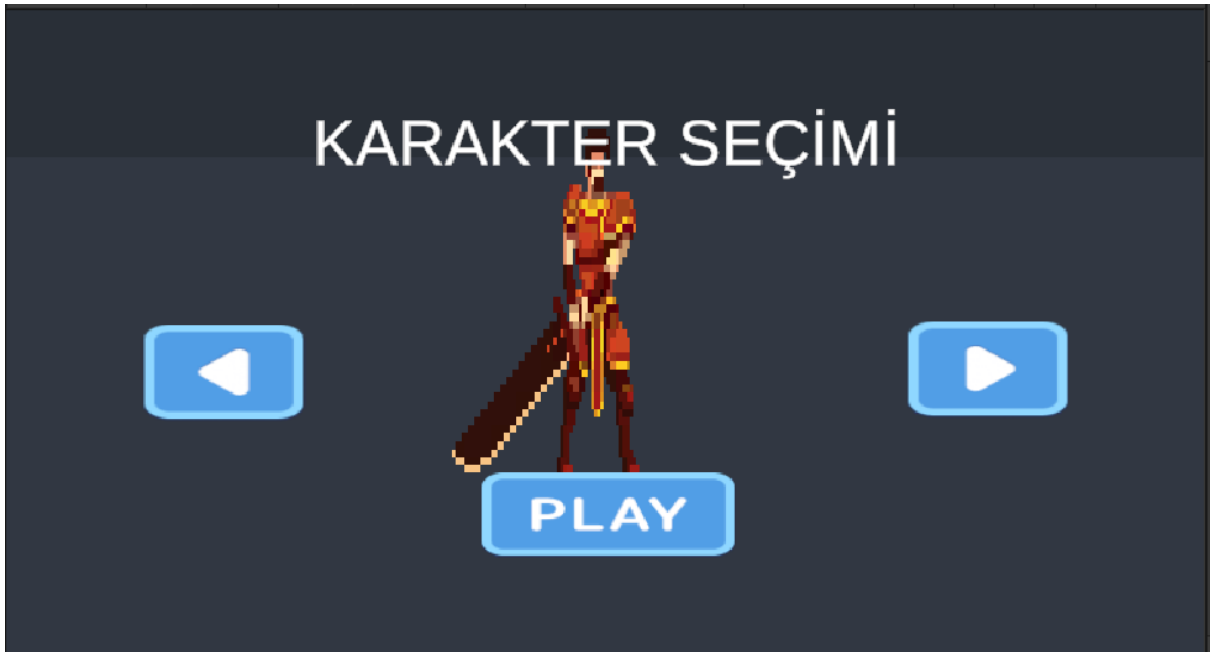
PROJE YAPISI VE ÇALIŞMASI

Her bir oyuncu nesnemizi kapsayan bir playerMovement aynı zamanda her birinin ayrı ayrı Combat scriptleri mevcuttur. Düşman kısmında ise 4 adet düşmanımızın her birinin bir adet DusmanAI ve enemy isimlerinde scriptler mevcuttur.Enemy scripti hasar alma,ölüm metotları ve sağlık değişkenlerini tutan bir script iken DusmanAI scripti karakter salınım hareketi(eğer karakter hareket ediyorsa),hasar verme,çağrılacak bir nesne yaratma metodunu içermektedir.Bunların dışında oyuna çağrıldığı zaman dahil olan nesnelerin de scriptleri vardır(FireBall,Arrow,EnemyArrow vb.).Bunlar dışında kullanıcı arayüz kısmında ise kısaca hangi tuşun hangi ekranı açacağını ve ne işe yarayacağını yazdığımız scriptler mevcuttur.

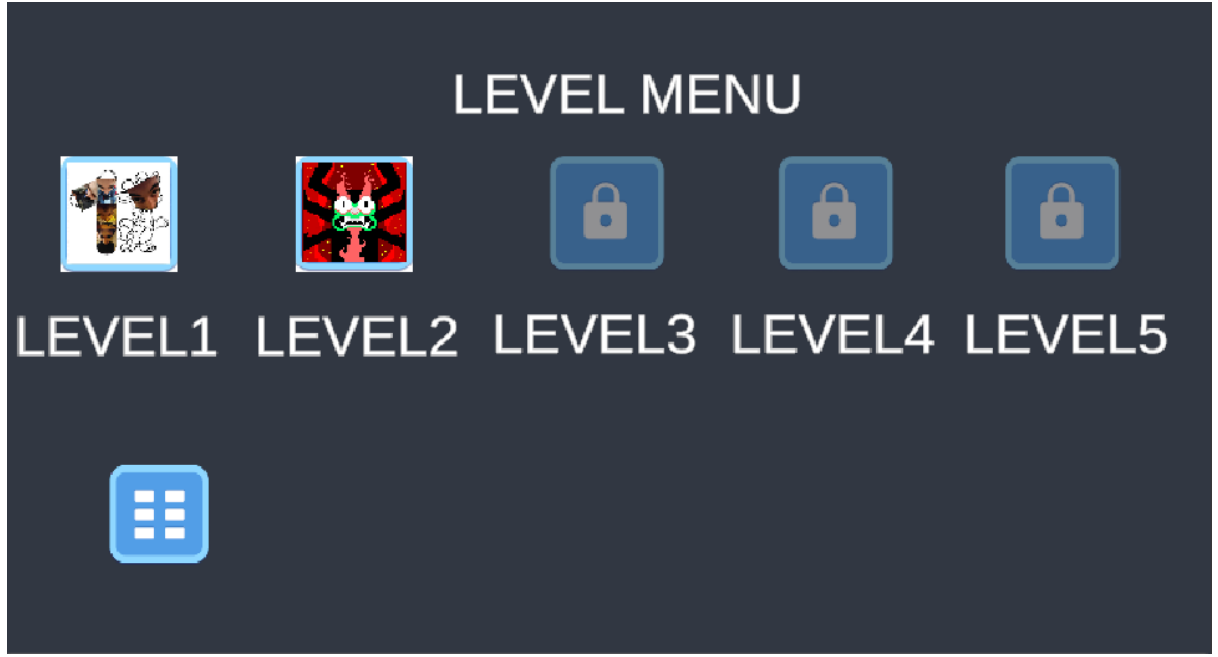
Projenin çalışmasında ise ilk etapta giriş ekranı geliyor.Play tuşuna basıldığında karakter seçim ekranı gelmekte bu ekrandan da seçmek istediğimiz karakteri oklarla sağa sola giderek bulup play tuşuna basınca seviye seçme ekranına devam ediyor.Seviye seçme ekranımızda istediğimiz seviyeye tıklıyoruz ardından o seviye açılıyor ve oynamaya başlıyoruz.Seviye ekranımızda çeşitli düşmanlar, bazı düşmanlardan belirli olasılıklarla düşme oranına sahip yüzükler,ana karakterin can barı gibi nesneler içermektedir.



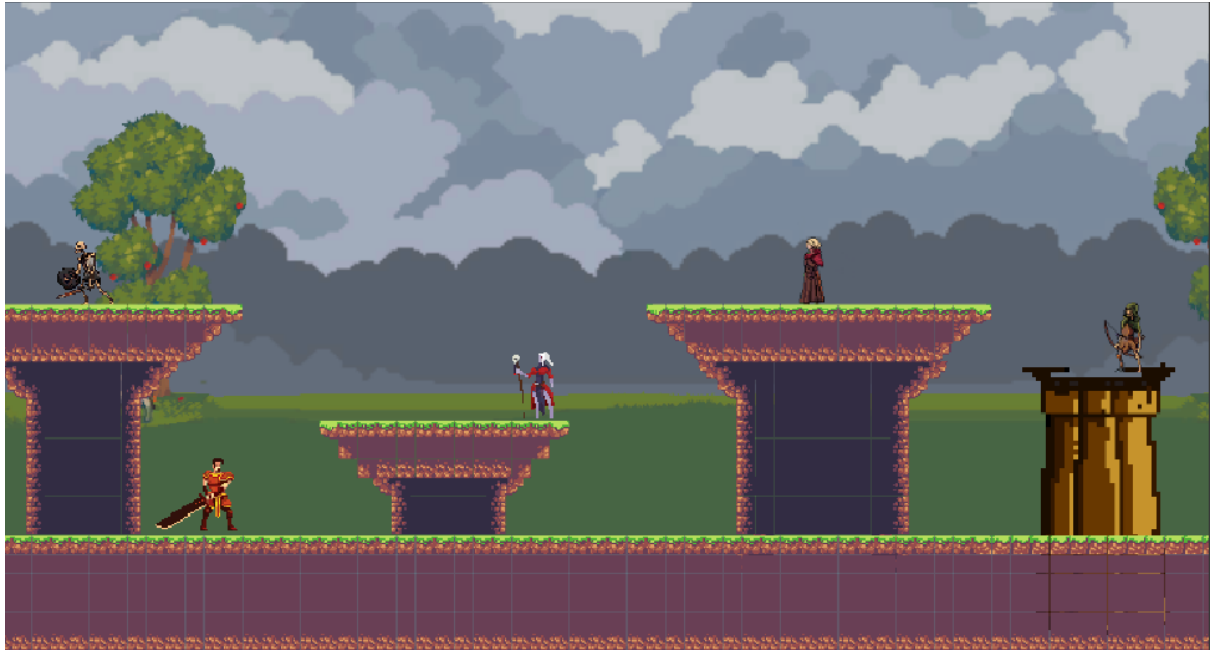
Şekil.3.1.Giriş Ekranı



Şekil.3.2.Karakter Seçim Ekranı



Şekil.3.3.Level Seçim Ekranı



Şekil.3.4.Level Ekranı

Nesneye yönelik ilkelerden kalıtımı, Monobehaviour base classından doğrudan classlar oluşturarak kullandık.

Oyuncular Combat scriptlerinde hasar alma,4 adet hasar verme scripti,ölüm methodlarını içermektedir. PlayerMovement ortak scriptinde oyuncunun zıplama ve hareket kodlaması mevcut.CharacterController diye bir scriptimiz daha mevcut oyuncu için. Düşman kısmında ise DusmanAI ve enemy scriptleri var.DusmanAI scriptinde hareket,iyileştirme,atak ve çağırma gibi durumlar yazmakta(hareket,çağırma,iyileştirme durumu düşmandan düşmana değişiklik göstermektedir).Bir de Fireball ve EnemyArrow scriptleri vardır.Bu scriptlerde Fireball 5.düşmanın çağırdığı ateş topu olup içeriğinde hasar alma,Oyuncuyu takip etme,ölüm ve atak(patlama) metodlarını içermektedir. EnemyArrow scripti ise belli hızda oyuncuyu belirtilen kapsama alanının içine girdikten sonra belli saniye sonra ateş etmeye başlayan scripttir.Bir de boss scripti mevcuttur.Bu da BossAI ve Boss diye iki adet script mevcut.Boss scripti ölüm ve hasar alma metotlarına sahip iken BossAI scripti hareket,oyuncuyu göredüğü zaman daha hızlı gitmesini sağlayan EnemyFollow metotunu ve atak metotunu içerir.GirisEkranı scriptinde play butonu karakter seçim ekranına gönderir.Karakter seçim ekranında oklar karakterler arası gezinmeye yararken Play butonu ise seviye seçme ekranına geçmemizi sağlar.Seviye seçme ekranında bulunan sayılar seviyeleri temsil ederken bu butonlara basılığında basılan sayının seviyesi açılır. Ardından seviye ekranı gelir seviye ekranında öldüğümüzde ise GameOver sahnesi açılır. Restart ve Main menu diye iki buton olur restart butonu seviye seçme ekranına dönmemizi sağlarken main menu giriş ekranına döner.

1 başvuru

```
IEnumerator SpecialAttackDelay()
```

```
{
```

```
    yield return new WaitForSeconds(0.75f);
```

```
    GameObject beam = Instantiate(beamPrefab, beamPoint.position, beamPoint.rotation);
```

```
    Collider2D[] hitEnemies = Physics2D.OverlapAreaAll(beamArea1.position, beamArea2.position, enemyLayer);
```

```
    foreach (var enemy in hitEnemies)
```

```
    {
```

```
        if (enemy.GetComponent<enemy1>() != null) { enemy.GetComponent<enemy1>().TakeDamage(beamDamage); }
```

```
        if (enemy.GetComponent<enemy2>() != null) { enemy.GetComponent<enemy2>().TakeDamage(beamDamage); }
```

```
        if (enemy.GetComponent<enemy3>() != null) { enemy.GetComponent<enemy3>().TakeDamage(beamDamage); }
```

```
        if (enemy.GetComponent<enemy4>() != null) { enemy.GetComponent<enemy4>().TakeDamage(beamDamage); }
```

```
        if (enemy.GetComponent<enemy5>() != null) { enemy.GetComponent<enemy5>().TakeDamage(beamDamage); }
```

```
        if (enemy.GetComponent<FireBall>() != null) { enemy.GetComponent<FireBall>().TakeDamage(beamDamage); }
```

```
        if (enemy.GetComponent<Boss>() != null) { enemy.GetComponent<Boss>().TakeDamage(beamDamage); }
```

```
    }
```

```
    Destroy(beam, 0.5f);
```

```
}
```

Şekil.3.5.SpecialAttackDelay

O an sahnede hangi düşman varsa ona ulti atmayı yarayan kod aynı zamanda diğer ataklarda buna benzer yapıda olduğundan bence önemli bir kod.

```

1 başvuru
void Heal1()
{
    timer1 += Time.deltaTime;
    if (timer1 >= 5)
    {
        if (H1.currentHealth > 0 && H1.currentHealth < H1.maxHealth)
        {
            //animator.SetBool("Heal", true);
            animator.SetTrigger("Heal");
            H1.currentHealth += 25;
            timer1 = 0;
        }
    }
}
1 başvuru

```

Şekil.3.6.Heal1

Bu kodda diğer düşmanlardan ziyade sağlık vermeye yarayan düşman olduğu için farklı olduğundan hoşuma giden script budur. Kodun bu kısmı sahnede düşman1 varsa Heal1 metodunu çalıştırıyor ve bu kod 5 saniye sonra bir eğer sağlığı maksimum sağlığa eşit değilse H1'in mevcut 25 canına ekliyor. Bu kodun önemli olmasının en büyük yanı kaynaklarda vs. çok fazla bulunmuyor aynı zamanda atak olayının dışında bir şey yapıyor olması.

```

using UnityEngine.SceneManagement;

Unity Betiği (2 varlık başvurusu) | 0 başvuru
public class LevelMenu : MonoBehaviour
{
    public string mainMenuname = "girisEkran";
    public string level1name = "level1";
    public string level2name = "level2";
    public string level3name = "level3";
    0 başvuru
    public void LoadMainMenu()
    {
        SceneManager.LoadScene("girisEkran");
    }

    0 başvuru
    public void LoadLevel1()
    {
        SceneManager.LoadScene(level1name);
    }

    0 başvuru
    public void LoadLevel2()
    {
        SceneManager.LoadScene(level2name);
    }

    0 başvuru
    public void LoadLevel3()
    {
        SceneManager.LoadScene("level3");
    }
}

```

Şekil.3.7.LevelMenu

Bu kodda butonlarla sahnelere erişimi SceneManager kütüphanesi LevelMenu ekranından level1,level2,main menu ekranlarına geçişini sağlıyor.Bu kodu seçme sebebim temel anlamda tüm UI scriptleri buna benzer şekilde oluşmasından sebeptir.

KAYNAKLAR

Bliss assets <https://prinbles.itch.io/bliss> 7 Mayıs tarihinde adresinden erişildi.

Background assets <https://craftpix.net/freebies/free-halloween-2d-game-backgrounds/> 3 Mayıs tarihinde adresinden erişildi.

Tileset assets <https://assetstore.unity.com/packages/2d/characters/sunny-land-103349> 1 Mayıs tarihinde adresinden erişildi.

Boss assets <https://chierit.itch.io/boss-demon-slime?download> 7 Mayıs tarihinde adresinden erişildi.

Button assets <https://prinbles.itch.io/silent> 13 Mayıs tarihinde adresinden erişildi.

Wind Hashashin assets <https://chierit.itch.io/elementals-wind-hashashin> 2 Mayıs tarihinde adresinden erişildi.

Water Priestess assets <https://chierit.itch.io/elementals-water-priestess> 2 Mayıs tarihinde adresinden erişildi.

Fire Knight assets <https://chierit.itch.io/elementals-fire-knight> 2 Mayıs tarihinde adresinden erişildi.

Leaf Ranger assets <https://chierit.itch.io/elementals-leaf-ranger> 2 Mayıs tarihinde adresinden erişildi.

Archer,Shieldman assets <https://craftpix.net/freebies/free-skeleton-pixel-art-sprite-sheets/> 29 Nisan tarihinde adresinden erişildi.

Spawner assets <https://luizmelo.itch.io/evil-wizard-3> 13 Mayıs tarihinde adresinden erişildi.

Fireball assets <https://craftpix.net/freebies/free-fantasy-enemies-pixel-art-sprite-pack/> 14 Mayıs tarihinde adresinden erişildi.

Healer assets <https://craftpix.net/freebies/free-wizard-sprite-sheets-pixel-art/?num=1&count=1496&sq=wizard%202d&pos=4> 1 Mayıs tarihinde adresinden erişildi.

Enemy scripts
<https://www.youtube.com/playlist?list=PLnanKIXWfsieWvMXwGF2TWwYYYWYHPaNC>

18 Mart tarihinde adresinden erişildi

Healthbar canvasına https://www.youtube.com/watch?v=BLfNP4Sc_iA&t=835s

15 Mayıs tarihinde adresinden erişildi.

Character selection system https://www.youtube.com/watch?v=_xjszfXBZeo 12 Mayıs tarihinde adresinden erişildi.

Character Load <https://www.youtube.com/watch?v=3qIRgICRoeA> 12 Mayıs tarihinde adresinden erişildi.

How to make 2D game in unity
<https://www.youtube.com/playlist?list=PLPV2KyIb3jR6TFcFuzI2bB7TMNIIbPbKMQ>

20 Nisan tarihinde adresinden erişildi.

How to make explosion in unity <https://www.youtube.com/watch?v=k4hr7-7ysCY&t=8s> 14 Mayıs tarihinde adresinden erişildi.

2D enemy shooting unity tutorial <https://www.youtube.com/watch?v=--u20SaCCow&t=176s> 12 Mayıs tarihinde adresinden erişildi.

Unity Kursuna <http://udemy.com/course/unity-ile-sifirdan-profesyonellige-3d-2d-oyun-yapimi/> 27 Nisan tarihinden adresine erişildi.