10.Hafta Ders Raporu – Batuhan Şengül

# Post Processing

Projeye post proceesing eklemek için. A screenshot of a computer program

Description automatically generated

Window -> package manager seçilir.

A screenshot of a computer

Description automatically generated

Packages sekmesinde unity registiry seçilir. Arama kısmına istenilen paketin adı yazılır ve install tuşuna basılarak projeye dahil edilir.

A screenshot of a computer

Description automatically generated

Yeni boş bir oyun nesnesi oluşturulur ve adı “Post Proccess Manager” olarak ayarlanır.

A screenshot of a computer

Description automatically generated

Oluşturulan oyun nesnesine Post-Process Volume bileşeni eklenir. Layer’ı yeni oluşturulan “PostProcess” olarak ayarlanır.

Add effect tuşunua basılarak istenilen efektler eklenebilir. Bu örnektel Bloom ve Color Grading efektleri eklenmiştir.

Eklenen efektlerin detayları isteğe bağlı olarak ayarlanır.

## Bloom

A screenshot of a computer

Description automatically generated

Oyun için bu şekilde düzenlenmiştir.

## Color Grading

A screenshot of a computer

Description automatically generated

Oyun için bu şekilde düzenlenmiştir.

## Main Camera

A screenshot of a computer

Description automatically generated

Oyunun görüntü aldığı kameranın bu yapılan efektlerden etkilenmesi için main camera oyun nesnesine; Post-Process Layer bileşeni eklenir Layer ayarları düzenlenir ve efektlerimiz oyuz üzerinde etki alır.

A screenshot of a video game

Description automatically generated

Post-Process efektlerinden sonra oyun

A video game screen with a rocket and text

Description automatically generated

Post-Process efektleri olmadan oyun

# Sesler

Sesler için iki tane script oluşturulmuştur biri ses sınıfının genel özelliklerini tutan bir Sound.cs base sınıfı diğeri de seslerin genel olarak oynatılmasını durdurulmasını vb. durumları kontrol eden bir Singleton yapısında bir AudioManager.cs scripti.

## Sound.cs

A screenshot of a computer program

Description automatically generated

Bir ses nesnesinin basit özelliklerini tutan sınıf. Detaylı olarak açıklamayı düşünülmemektedir çünkü işin inekliğini yaptım ve uzun vadede kullanımıma uyun bir yapı oluşturdum.

## AudioManager.cs

### Değişkenler

A screen shot of a computer

Description automatically generated

### Awake() Fonksiyonu

A computer screen shot of a program code

Description automatically generated

Singelton yapısı kurulur. Ses yöneticisindeki tüm sesler oluşturulur, hierchyde gerkli yerlere eklenirler.

### Start() Fonksiyonu

A screen shot of a computer

Description automatically generated

Oyunun ana müziği başlatılır ve sahne yüklemelerini izleyen bir dinleyici atanır.

### OnDestroy() Fonksiyonu

A screen shot of a computer

Description automatically generated

Başlangıçta oluşturulan dinleyici nesnenin yok olmasına karşın kaldırılır. (Memory leak olasın diye)

### OnPlayerLoaded(GameObject newPlayer) Fonksiyonu

A screen shot of a computer program

Description automatically generated

Oyuncu oyun nesnesi oluşturulduğunda çalışr oyuncunun üzerinde olacak sesler atamasını sağlar. Örneğin ateş etme, zıplama gibi oyuncunun kendisinin çıkartabileceği sesler.

### AttachSoundToObject(Sound s, GameObject targetObject) Fonksiyonu

A computer screen with text and images

Description automatically generated

Eğer ses/sesler belirli bir nesnede oynatılmak isteniyorsa (örn. oyuncu) ses kaynağı bileşeninin ilgili nesnede oluşturulmasını sağlayan fonksiyon.

### Play(string sound) Fonksiyonu

A computer code on a black background

Description automatically generated

Adına göre istenilen sesi çalmamızı sağlayan fonksiyon. Inspector üzerinden yapılan ayarlamalarla sesin hızına ve ya hacmine de rastgelelik bu fonksiyonda uygulanır.

### PlayAtLocation(string sound, Vector3 pos,float spatialBlendValue,float minRangeValue, float maxRangeValue)

A screen shot of a computer program

Description automatically generated

Bu fonksiyon da istediğimiz sesi, oyun uzayında istenilen noktada oynatmamıza olanak tanır. Bu fonksiyon patlamalarda kullanılmıştır. Bu sesler 3d ayarlanarak oyuna daha dinamiklik katmıştır.

## Oyun İçi Sesler

 Yeni bir oyun nesnesi oluşturulup adı “AudioManager” olarak düzenlenmiştir. İçerisine “AudioManager.cs” scripti eklenmiştir.

A screenshot of a computer

Description automatically generatedYanda görüleceği üzere script bu şekilde gözükmektedir bu inspector görünümü ayrıca özel editor kodları kullanılark yapılmıştır detaylı bilgiler için github reposuna bakabilirsiniz. Sounds sekmesi altında gerekli sesler oluşturulur. Ve ayarları isteğe uygun şekilde yapılır.

### Arkaplan Müziği

A screen shot of a computer

Description automatically generated

AudioManager.cs içerisinde ana müzik başlatılır.

### Lazer

A computer screen shot of a program code

Description automatically generated

Playermovement.cs içindeki ateş etme fonksiyonunun olduğu yere bu eklenen satır ile ses oynatılır.

### Patlama Sesi

A screen shot of a computer program

Description automatically generated

Enemy.cs scripti içinde düşmanlar yok edildiğinde çağırılan fonksiyonda yukarıdaki gibi 3 boyutlu olacak şekilde gerekli parametreler verilerek patlama sesi eklenir.

### Bonus Yakalama Sesi

A computer screen shot of a program code

Description automatically generated

PowerUp.cs scripti içerisinde oyuncuyla temasa geçildiğinde çalışan kod içine, ilgili sesi oynatacak kısım eklenmiştir.

# Kaynakça

<https://docs.unity3d.com/Manual/>

<https://docs.unity3d.com/Packages/com.unity.ugui@1.0/manual/UIBasicLayout.html>

<https://chatgpt.com/>

# Proje Kodu ve Github Repo

Kod: <https://github.com/bathuchan/btu-gameprogramming-BatuhanSengul/tree/main/Reports/10.Hafta>

Proje Repo: <https://github.com/bathuchan/btu-gameprogramming-BatuhanSengul>

# Hazırlayan

Batuhan Şengül – 20360859008- bathu.sengul@gmail.com