T.C. Fırat Üniversitesi
Bilgisayar Mühendisliği Bölümü

BMÜ460 – Oyun Programlama Dersi

3. Ödev Raporu

Ders Sorumlusu: Dr. Öğr. Üyesi Ahmet ÇINAR

Öğrenci İsim: Mert İNCİDELEN
Öğrenci No.: 170260101

ÖDEV TANIMI: *Ödevde oyun motoru, fizik motoru ve GPU kavramlarının açıklanması ve seçilen bir oyun motoru için basit bir oyunun geliştirilme aşamalarının izah edilmesi beklenmektedir.*

Oyun Motorları: Oyun motorları, oyun geliştirmek için gerekli tüm araçları ve özellikleri barındıran, oyun geliştirmeyi kolaylaştıran yazılımlardır. İki boyutlu ve üç boyutlu oyunlar yaratılabilmesine olanak sağlayan geliştirme ortamlarıdır. Unity gibi bazı oyun motorları oyunların mobil ve masaüstü çeşitli platformlar için geliştirilebilmesine imkân tanır.

Fizik Motorları: Fizik motorları, bilgisayar ortamında oluşturulan nesnelerin hareketlerinin, birbirileriyle etkileşimlerinin ve fiziksel olayların gerçek yaşamdaki fizik kuralları çerçevesinde gerçekleşebilmesini sağlayan yazılımlardır. Bilgisayar oyunlarında, fiziksel olayların ve nesne hareketlerinin gerçek yaşamda olması gerektiği gibi olmasını sağlayarak oyunlara gerçekçilik ve akla uygunluk kazandırmaktadır.

Grafik İşlemci Birimi (GPU): Grafik işlemci birimi, görüntü işleme amacıyla ihtiyaç duyulan çok sayıdaki hesaplama işlemlerinin gerçekleştirildiği birimdir. Ekran performansının yönetiminde ve hızında doğrudan rol oynayan, bir elektronik cihazda görüntülemek üzere iki ve üç boyutlu grafikler oluşturabilen özel olarak tasarlanmış elektronik devrelerdir.

Unity ile Basit Oyun Yapım Aşamaları

Oyun motorları, oyun görsellikleri ve işlevsellikleri için büyük kolaylıklar sağlamaktadır. Unity de bu oyun motorlarından bir tanesidir. Oyun geliştirme için ilk aşama oyun fikrinin, karakter ve senaryoların belirlenmesidir. Daha sonra oyun için gerekli görsel düzenlemelere geçilebilir. 2 boyutlu basit bir oyun yapımı için gereken aşamalar şöyledir:

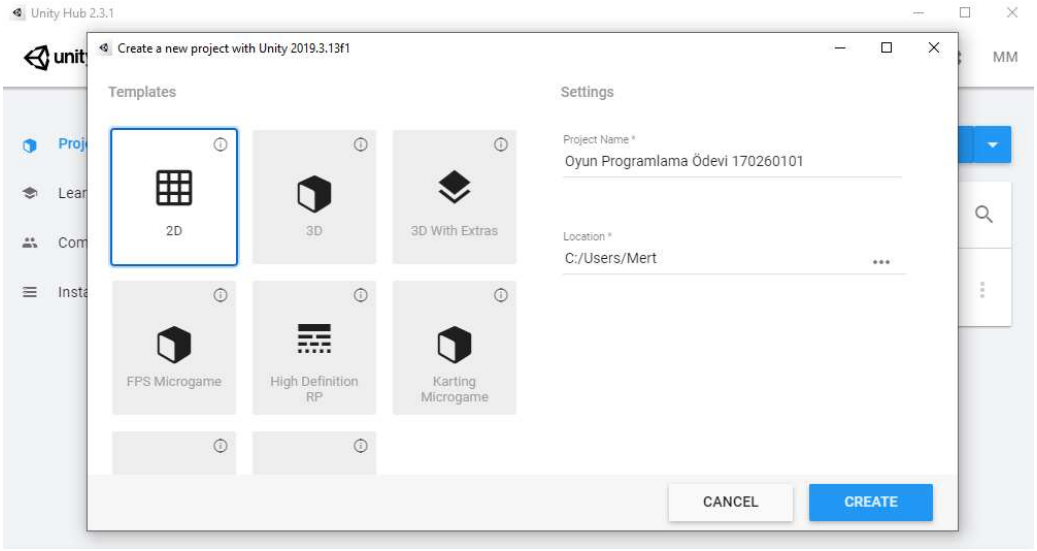
- **Varlıklar Belirlenir**

Oyunda kullanılacak olan karakterler, zemin gökyüzü ve benzeri dekorlar ve diğer tüm unsurlar için varlıklar (assets) oluşturulur. Varlıklar için hazır tasarımlar Unity'de ve internetteki birçok kaynaktan edinilebilir. Varlıkların görünümünün oyunun görsel tasarımı için doğrudan etkisi vardır.

Şekil 1: *Karakter Varlığı İçin Farklı Görünümlerden Oluşan Bir Set*

- **Proje Oluşturulur**

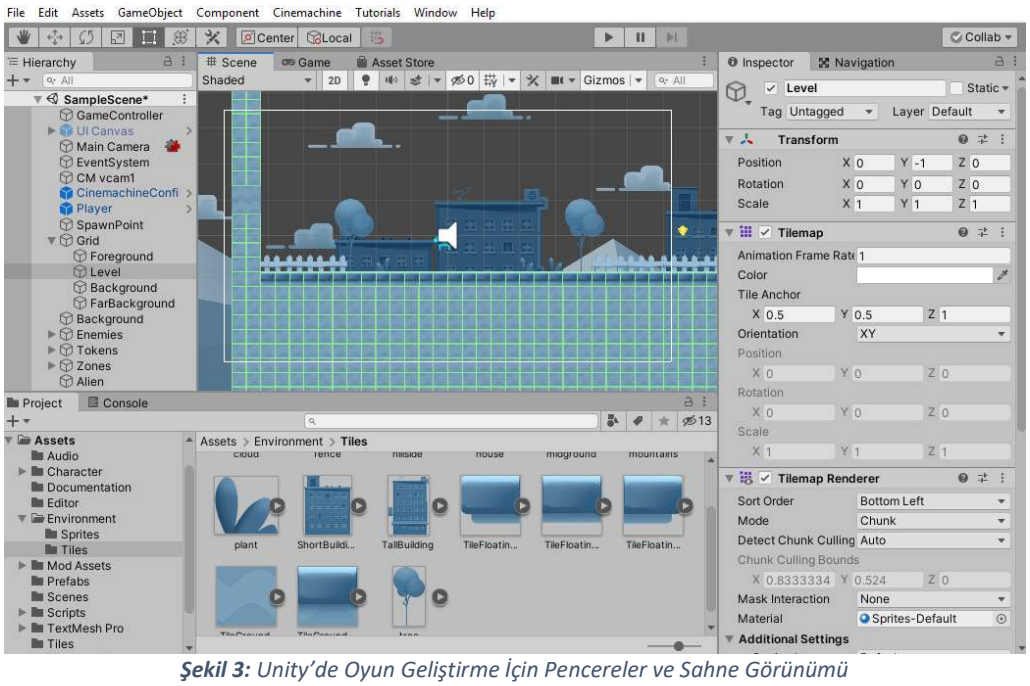
Oyunun tüm unsurlarının ve çalışmaların yer alacağı oyunun niteliklerine uygun iki boyutlu ya da üç boyutlu çalışma için bir proje oluşturulur. Basit bir oyun yapımı için 2D çalışmalarda sahne iki boyutlu uzaydan oluşmaktadır. Proje dosyalarında görsel materyaller ve kodlar yer almaktadır.



Şekil 2: *Unity'de Proje Oluşturma Ekranı*

- **Zemin, Dekorlar ve Diğer Varlıklar Eklenir**

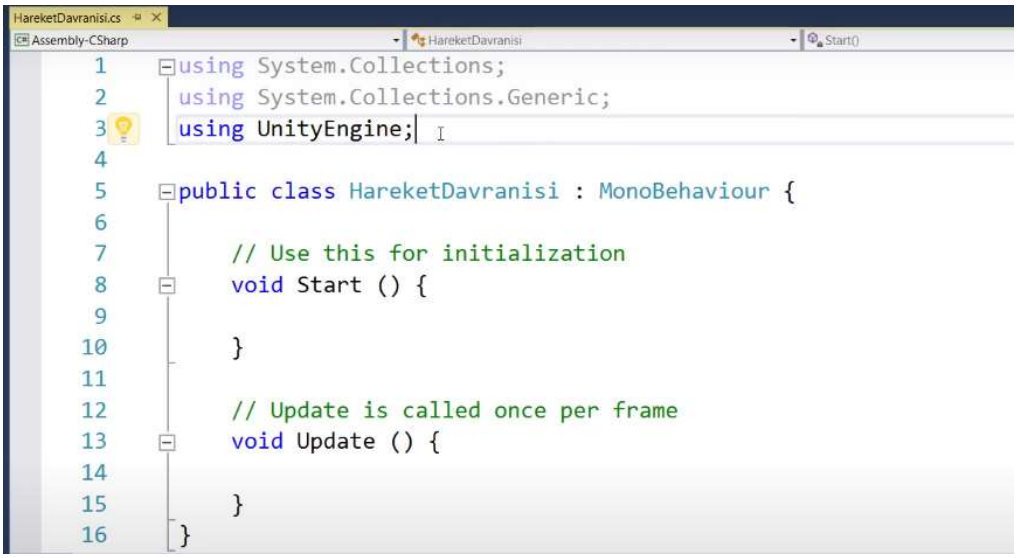
Unity'de bulunan, özel olarak tasarlanan veya internet üzerinden edinilebilen varlık (assets) tasarımları ile sahnedeki ızgaralara yerleştirilerek zemin ve çeşitli engeller oluşturulabilir. Görsellerin fonksiyonları, özellikleri ve boyutları gibi ayarlar oyun motorundaki pencereler yardımıyla kolaylıkla yapılabilir. Daha sonra oyun sahnesi için gerekli arka plân dekorları yine bu varlık tasarımları kullanılarak gerçekleştirilir. Oyunda yer alacak hareketli varlıklar da eklenir.



Şekil 3: *Unity'de Oyun Geliştirme İçin Pencereler ve Sahne Görünümü*

- **Karakterlerin Hareketleri, Diğer Fiziksel Olaylar ve İşlemler İçin Gerekli Kodlar Yazılır**

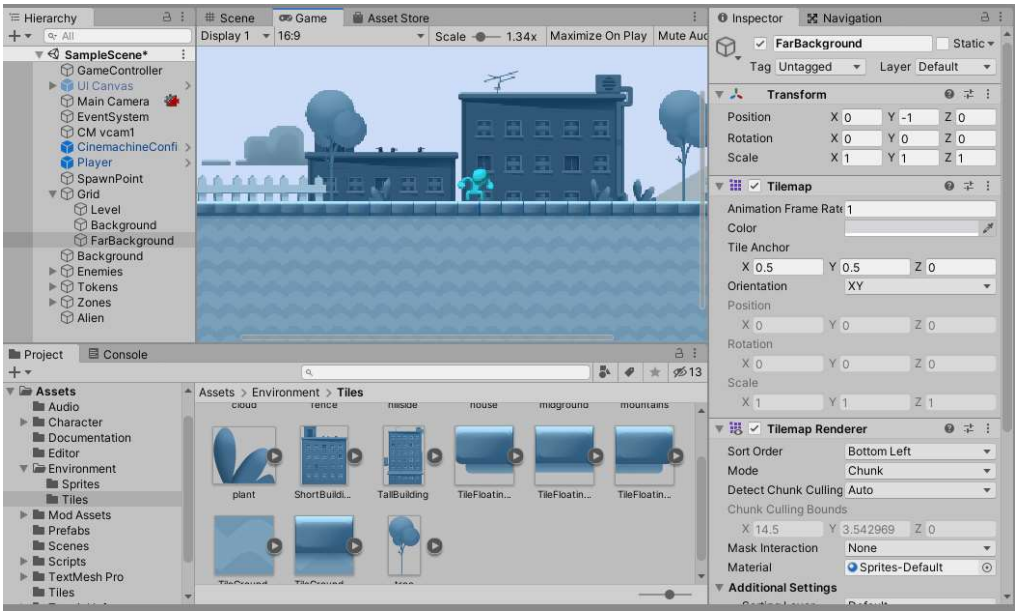
Görsel düzenlemeler ve sahne tasarımlarının ardından oyun içindeki olaylar, varlıkların fiziksel hareketleri ve tepkileri için gerekli kodlar yazılır. Programlama safhasında C# Programlama Dili kullanılabilir. Aşağıdaki şekilde örnek olarak gösterilen hareket davranışı için yazılacak kodlar için Start() bloğuna kurucu kodlar, Update() bloğuna hareket davranışı için frame başına döndürülecek gerekli kodlar yazılarak hareket davranışı için gerekli işlemler tanımlanmış olur.



Şekil 4: *Hareket Davranışının Kontrolü İçin Örnek Kod Ekranı*

- **Oyun, Yürütme Penceresi Yardımı ile Her Safhada Test Edilebilir ve Gözlemlenebilir**

Görsel düzenlemelerin nasıl göründüğü ve arka plânda yazılan kodların oyun üzerinde nasıl çalıştığı, oyun motorunda bulunan game penceresi yardımıyla başlatılarak test edilip gözlemlenebilir.



Şekil 5: *Unity'de Oyunun Anlık Görüntülenebildiği ve Oynanarak Test Edilebildiği Pencere Ekranı*

YARARLANILAN KAYNAKLAR

Ödev içeriğindeki ifadeler ve şekillerdeki ekran görünümleri özgündür. Ödeve hazırlıkta aşağıdaki kaynaklardan faydalanılmıştır.

[1] https://unity3d.com/what-is-a-game-engine

[2] https://interestingengineering.com/how-game-engines-work

[3] https://education.abc.net.au/home#!/media/2629019/what-is-a-physics-engine-

[4] https://www.computerhope.com/jargon/p/physics-engine.htm

[5] https://www.synopi.com/graphics-processing-unit-gpu/

[6] https://www.investopedia.com/terms/g/graphics-processing-unit-gpu.asp

[7] https://sanalkurs.net/unity-ile-2d-basit-platform-oyunu-yapimi-10803.html

[8] https://learn.unity.com/course/beginning-2d-game-development

[9] https://www.youtube.com/watch?v=o3r34p9ZBmY