

Unity İle İlk Oyunum

Unity İle İlk Oyunum
Tamer GÜLSOY, Gamze GÜLSOY,
Nursultan BOSTAN, Merve ÇAKIR

Kapak Tasarımı
Tamer GÜLSOY

Mizanpaj
Tamer GÜLSOY

1. Basım, Ocak 2026

ISBN
978-625-5689-73-3

Yayınevi
Tenha Yayınları
Alpaslan mah., Besmele sok., no 10,
Nurşen apt., kat 4, daire 10,
Meram/Konya
05386685449
www.kitapincele.net

Baskı
Vadi Grafik Tasarım Ve Reklamcılık Ltd. Şti.
İvedik Org. San. 1420. Cad. No: 58/1
Yenimahalle/Ankara
Tel.: 0312 395 8571
Sertifika No: 47479

ÖNSÖZ

Değerli okurlar,

Elinizde tuttuğunuz bu çalışma, yalnızca bir yazılım eğitim kitabı değil, aynı zamanda dijital yaratıcılığa giden eğlenceli ve keşif dolu bir yolculuğun başlangıcıdır. Bugün, Unity 6.2 teknolojisinin sunduğu sınırsız imkanlar, teorik bilginin sınırlarını aşarak hayal gücünüzü somut, estetik ve işlevsel çıktılar üretmeye yönlendiriyor. Kitap boyunca yer alan dört ana proje, bu yaratıcı felsefenin teknik birer yansımasıdır.

Unutmayın, Unity editöründeki her bir bileşen ve yazdığınız her bir satır kod, sizin sınırsız potansiyelinizin bir kapısıdır. Bu kitapla edineceğiniz bilgi ve beceriler, sadece basit birer oyun projesi için değil, aynı zamanda dijital çağın sunduğu kariyer yollarında da size güçlü bir temel oluşturacaktır.

Kodun mantığıyla sanatın estetiğini birleştiren bu yaratıcı yolculukta hepinize başarılar dileriz. Unity'yi elinize alın ve kendi dijital sanat eserinizi yaratmaya başlayın!

İÇİNDEKİLER

Bölüm 1: Oyun Geliştirmeye Giriş

- Oyun geliştirme nedir?
- 2D ve 3D farkları
- Oyun türleri (Platformer, Puzzle, Arcade, Shooter, Runner vs.)
- Oyun geliştirmede hangi araçlar kullanılır?
- Unity'nin sektördeki yeri

Bölüm 2: Unity Kurulumu ve Ortam Hazırlığı

- Unity Hub nedir?
- Unity Hub indirme ve kurulum adımları
- Gerekli modüller:
 - **Editor sürümü seçimi (örnek: 2022 LTS)**
 - **Visual Studio / VS Code entegrasyonu**
 - **Android Build Support ekleme** (mobil geliştirmek isteyenler için)
- İlk proje oluşturma ("2D Template" seçimi)
- Proje klasör yapısının tanıtımı

Bölüm 3: Unity Arayüzünü Tanıyalım

- Scene, Game, Hierarchy, Inspector, Project ve Console panelleri
- Oyun nesneleri (GameObject) ve bileşenler (Component)
- Sahne kaydetme ve çalışma prensibi
- Oyun nesnesi oluşturma (sprite ekleme, zemin, arka plan)

Bölüm 4: C# Programlama Temelleri

- Unity'de script nasıl oluşturulur
- Start() ve Update() fonksiyonları
- Değişken türleri (int, float, bool, string)
- Operatörler (+, -, *, /, %)
- Debug.Log() ile çıktı alma
- Basit örnek: "Top zıpladığında sayacı artır"

Bölüm 5: Karar ve Döngü Yapıları

- if, else if, else
 - for, while, foreach
 - Koşul örnekleri (tuşa basıldığında hareket, skor artışı, zaman kontrolü)
 - Mini uygulama: Basit sayaç veya karakter hareket testi
-

Bölüm 6: Unity’de Bileşenlerle Çalışmak

- Rigidbody2D, Collider2D, Sprite Renderer nedir?
 - Prefab kavramı ve kullanım alanı
 - Nesne kopyalama, yok etme (Instantiate, Destroy)
 - Çarpışma tespiti (OnCollisionEnter2D, OnTriggerEnter2D)
-

Bölüm 7: Oyuncu (Player) Kontrolleri

- Klavye ile sağ-sol hareket
 - Zıplama mantığı
 - Yere temas kontrolü
 - Sprite animasyonları (Idle, Run, Jump)
 - Oyun alanı sınırlarını belirleme
-

Bölüm 8: Zemin ve Arka Plan Tasarımı

- Tilemap sistemine giriş
 - Arka plan (background) ekleme
 - Parallax efektleriyle derinlik hissi
 - Kamera karakteri takip etsin (Camera Follow Script)
-

Bölüm 9: Nesne Etkileşimi ve Skor Sistemi

- Coin toplama
 - Skor artışı
 - UI Text ile ekranda gösterim
 - Ses efekti ekleme
-

Bölüm 10: Düşman (Enemy) Sistemi

- Düşman hareketi (patrol mantığı)
 - Oyuncuya çarpınca can azaltma
 - Oyuncu öldüğünde “Game Over” ekranı
-

Bölüm 11: Basit Yapay Zeka (AI) Davranışları

- Takip eden düşman
 - Rastgele hareket
 - Devriye (ileri-geri yürüme)
 - Basit saldırı davranışı
-

Bölüm 12: Ses, Müzik ve Efektler

- AudioSource ve AudioClip ekleme
 - Arka plan müziği
 - Olay bazlı ses çalma
 - Particle System ile görsel efektler
-

Bölüm 13: Oyun Arayüzü (UI) Tasarımı

- Canvas sistemi
 - Ana menü, butonlar ve event'ler
 - Pause menüsü
 - Skor ve can barı UI'si
-

Bölüm 14: Seviye (Level) Sistemi

- Scene oluşturma ve geçiş (SceneManager)
 - Checkpoint ekleme
 - Oyun bitiş ekranı
-

Bölüm 15: Kayıt Sistemi

- PlayerPrefs ile skor ve seviye kaydı
 - “Devam Et” özelliği
 - En yüksek skor kaydı
-

Bölüm 16: Mini Oyun Uygulamaları (4 Proje)

Her biri 10–15 sayfalık kısa örneklerle:

1. **Flappy Bird Klonu** – Fizik temelleri
 2. **Prencesi Kurtar** – Karakter hareketi, Çarpışma ve Skor Sistemi
 3. **Uzay Savaşı (Shooter)** – Mermi üretimi
 4. **Labirent Oyunu** – Kamera ve hedef takibi
-

Bölüm 17: Yayınlama ve Optimizasyon

- Build Settings ayarları
 - FPS artırma ipuçları
 - Dosya boyutunu küçültme
 - Oyun test etme ve hata ayıklama
 - itch.io / Google Play yükleme süreci
-

Bölüm 1: Oyun Geliştirmeye Giriş

Bölümün Amacı

Bu bölümde okuyucu, oyun geliştirmenin temellerini öğrenecek, 2D-3D farklarını anlayacak ve oyun türlerini tanıyacaktır. Ayrıca Unity'nin neden oyun geliştirme dünyasında bu kadar popüler olduğunu kavrayacaktır.

1.1 Oyun Geliştirme Nedir?

- Oyun geliştirmenin tanımı
- Oyun geliştirmenin aşamaları:
 - **Fikir oluşturma**
 - **Tasarım (Game Design)**
 - **Görseller ve sesler**
 - **Kodlama ve prototipleme**
 - **Test ve hata ayıklama**
 - **Yayınlama**
- Oyun geliştirmenin sanatsal ve teknik yönleri
- Takım çalışması ve bağımsız (indie) oyun geliştirme

1.2 2D ve 3D Oyun Farkları

Özellik	2D Oyunlar	3D Oyunlar
Görsel boyut	Yalnızca X-Y ekseninde	X-Y-Z ekseninde
Kamera	Sabit veya yan görünüm	Serbest, dönebilen kamera
Zorluk seviyesi	Daha basit mekanikler	Karmaşık modeller ve fizik
Kullanım alanı	Mobil, retro, indie oyunlar	AAA oyunlar, simülasyonlar
Örnekler	<i>Super Mario Bros, Celeste</i>	<i>GTA V, Skyrim</i>

1.3 Oyun Türleri

Oyun dünyası çok çeşitli türlerden oluşur. Bazı popüler türler:

- **Platformer:** Engeller arasında ilerleme (örnek: *Super Mario*)
- **Puzzle:** Zeka ve mantık temelli oyunlar (*Cut the Rope*)
- **Arcade:** Kısa, tempolu oynanış (*Flappy Bird*)
- **Shooter:** Ateş etme odaklı oyunlar (*Enter the Gungeon*)
- **Runner:** Sonsuz koşu oyunları (*Subway Surfers*)
- **Adventure / RPG:** Hikaye tabanlı oyunlar (*Undertale*)
- **Simulation:** Gerçek dünya benzetimi (*Stardew Valley*)

✦ Bu kitapta özellikle “Platformer” türü 2D bir oyun geliştireceğiz.

1.4 Oyun Geliştirmede Kullanılan Araçlar

- **Unity:** En popüler oyun motorlarından biri, 2D ve 3D destekler
 - **Unreal Engine:** Daha çok 3D, yüksek grafik isteyen oyunlar için
 - **Godot:** Açık kaynaklı, hafif bir alternatif
 - **Construct / GameMaker:** Kod bilmeden 2D oyun yapmak için
 - **Blender:** 3D modelleme ve animasyon için
 - **Audacity:** Ses düzenleme
 - **Photoshop / GIMP / Krita:** Sprite (karakter, arka plan) çizimi
-

1.5 Unity’nin Sektördeki Yeri

- Unity, dünya çapında en yaygın kullanılan oyun motorlarından biridir.
- Mobil, PC, konsol, web ve hatta VR oyunları geliştirmek mümkündür.
- 2D oyun geliştirme konusunda en kolay öğrenilen motorlardan biridir.
- Geniş **Asset Store** desteği sayesinde görseller ve hazır sistemler kolayca bulunabilir.

- Unity, **C# programlama dili** ile çalışır (bu kitapta da C# kullanılacaktır).

1.6 Kısa Kontrol Listesi (Checklist)

[.....] Oyun geliştirme aşamalarını (fikir, tasarım, test vb.) biliyorum.

[.....] 2D ve 3D arasındaki eksen farkını anladım.

[.....] "Platformer" türünün ne olduğunu biliyorum.

[.....] Unity'nin neden C# kullandığını ve avantajlarını kavradım.



Bu Bölümün Sonunda Öğrenecekleriniz

- ✓ Oyun geliştirmenin temel aşamalarını
- ✓ 2D ve 3D oyun farklarını
- ✓ Farklı oyun türlerini tanımayı
- ✓ Oyun geliştirme araçlarını ve Unity'nin önemini

Bölüm 2: Unity Kurulumu ve Arayüzü Tanıma

Bölümün Amacı

Bu bölümde okuyucu: Unity Hub ve Unity Editor'ü indirip kurmayı, gerekli modülleri eklemeyi, yeni bir 2D proje oluşturmayı ve Unity Editor arayüzünü ayrıntılı olarak kullanmayı öğrenecek. Bölüm sonunda basit bir sahne kuracak ve ilk nesnesine küçük bir hareket scripti ekleyecek.

2.1 Unity Hub Nedir? — Kısa Tanıtım

Unity Hub, birden fazla Unity sürümünü ve projeyi yönetmek için kullanılan resmi bir araçtır. Hub ile:

- Editor sürümleri indirilebilir ve güncellenebilir.
- Projeler oluşturulur, açılır veya kaldırılır.
- Modüller (Android Build Support vb.) yüklenir.
- Unity hesabı ile oturum açılır ve lisans yönetilir.

Neden Hub kullanmalıyız?

Bir projeyi belirli bir Unity sürümü ile kilitlemek, ekip içinde tutarlılık sağlar. Ayrıca farklı projeler için farklı sürümler kullanmak gerektiğinde Hub hayat kurtarır.

2.2 Sistem Gereksinimleri (Kısa)

- **İşletim Sistemi:** Windows 10/11 (64-bit) veya macOS 10.15+
- **İşlemci:** SSE2 destekli çok çekirdekli işlemci
- **RAM:** 8 GB (16 GB önerilir)
- **Disk:** En az 10–20 GB boş alan (Editor + modüller için)
- **Ekran:** 1024×768 minimum

Not: Mobil geliştirme için ek SDK/NDK/Java gereksinimleri vardır (Android Build Support modülü ile birlikte gelir veya Hub üzerinden eklenir).