

Animals Merge

Geliştirici: Abdullah Enes Karapınar – 205541059, Yunus Emre Özen – 215542001

Üniversite / Fakülte: [Yazılım Mühendisliği - Teknoloji Fakültesi]

Ders: Mobil Uygulama Geliştirme

Geliştirme Süresi: 8 hafta (Nisan 2025)

Kullanılan Araç: Unity (2D template)

Programlama Dili: C#

Hedef Platform: Android 8.0 ve üzeri

Veri Tabanı: Kullanılmadı (local kayıt: PlayerPrefs)

Oyun Türü: Puzzle / Merge (Birleştirme)

Görsel Tarz: Sevimli 2D sprite bazlı

Kitle: Geniş kitle (çocuklar, gençler, yetişkinler)

APK Boyutu: ~92MB

Yayın Türü: Offline (internet gerektirmez)

Proje Özeti ve Genel Bilgilendirme

Animals Merge, mobil platformlarda çalışmak üzere Unity kullanılarak geliştirilen 2D tabanlı bir bulmaca (merge/puzzle) oyunudur. Oyun, oyuncuya rastgele hayvanlar vererek bu hayvanları belirli bir kutu düzeni içerisinde yerleştirmesini ve aynı türden hayvanları birleştirerek daha üst seviyeye çıkarmasını hedefler. Bu mekanik, hem stratejik düşünmeyi hem de alan yönetimini ön plana çıkarır.

Oyunda toplam 11 farklı hayvan seviyesi vardır. Oyuncu, kurbağayla başladığı bu evrim sürecinde tavşan, ördek, köpek, domuz gibi hayvanları birleştirerek en yüksek seviye olan panda'ya ulaşmaya çalışır. Oyun süresince kazanılan altınlar markette kullanılabilir. Bu altınlar, kullanıcıya görsel özelleştirme seçenekleri sunar (kostüm, efekt vs.).

Oyun tamamen offline çalışır. Herhangi bir sunucuya ya da arka uç yapısına bağlı değildir. Veriler cihazın local storage'ında (Unity PlayerPrefs aracılığıyla) saklanır. Bu, veri güvenliği açısından kullanıcının verilerinin yalnızca cihazında kalmasını sağlar. Özellikle çocuk kullanıcılar için reklam veya internet bağlantısı gerektirmemesi büyük avantaj sağlar.

Uygulamanın amacı sadece eğlenceli bir oyun sunmak değil, aynı zamanda geliştiricilere oyun tasarımı, kullanıcı deneyimi, veri yönetimi ve oyun içi ekonomi sistemlerini nasıl yöneteceklerini öğretmektir.

SWOT Analizi

Güçlü Yönler (Strengths):

- Basit ve Etkili Oynanış: Öğrenmesi kolay, ustalaşması zor bir sistem sayesinde oyuncu hemen adapte olabilir.
- Görsel Çekicilik: Hayvan temalı sevimli grafikler her yaştan kullanıcıyı cezbetmektedir.
- Offline Yapı: İnternete ihtiyaç duymadan çalışması, ulaşılabilirliği artırır.
- Düşük Sistem Gereksinimi: Performans dostu yapısıyla düşük donanımlı cihazlarda bile sorunsuz çalışır.

Zayıf Yönler (Weaknesses):

- Tekdüze Yapı: Görev sistemi veya seviye geçişleri olmaması bazı kullanıcılar için tekrar hissi yaratabilir.
- Sosyal Etkileşim Eksikliği: Skor paylaşımı, arkadaş listesi veya çevrimiçi skor tablosu gibi sosyal mekanizmalar bulunmamaktadır.
- İleri Seviye İçerik Eksikliği: Uzun süreli oyuncular için yeni içerik ihtiyacı doğabilir.

Fırsatlar (Opportunities):

- Market Genişlemesi: Yeni kostümler, arka planlar, hayvan sesleri ile özelleştirme artırılabilir.
- Reklam ve Satın Alma Entegrasyonu: Uygulama içi satın alma ve ödüllü reklam sistemiyle gelir modeli oluşturulabilir.
- Global Rekabet: Firebase ya da benzeri servislerle online skor tabloları eklenerek kullanıcılar arası rekabet sağlanabilir.
- Etkinlikler ve Turnuvalar: Haftalık görevler, etkinlikler ve skor bazlı ödülleri kullanıcı etkileşimi ve bağlılığı artırılabilir.

Tehditler (Threats):

- Pazar Rekabeti: Mobil oyun pazarında benzer türde çok fazla oyun bulunmaktadır.
- İlgi Kaybı: Görev ve içerik eksikliği nedeniyle kullanıcı zamanla ilgisini kaybedebilir.
- Teknolojik Uyumluluk: Yeni Android sürümleri ile uyumluluk sorunu yaşanabilir.

UML Diyagramları

Kullanım Durumu (Use Case)

Bu diyagramda oyuncunun sistemle olan temel etkileşimleri gösterilir.

- Oyuncu → Oyunu başlatır
- Oyuncu → Hayvanı kutuya bırakır
- Oyuncu → Marketten kostüm satın alır
- Sistem → Aynı hayvanları algılar ve birleştirir
- Sistem → Puan ve altın günceller
- Sistem → Kutu dolarsa oyunu sona erdirir
- Sistem → Verileri cihazda kaydeder (PlayerPrefs)

Sınıf Diyagramı (Class Diagram)

Animal

- Attributes: seviye, sprite, pozisyon
- Methods: Birles(), YukariCik()

GameManager

- Attributes: skor, altin, aktifHayvan
- Methods: YeniHayvanUret(), SkorGuncelle(), OyunBitti()

MergeController

- Görevi: Aynı hayvanları algılayarak birleştirme fonksiyonlarını yönetir.

UIManager

- Görevi: Skor, altın, market ve oyun içi menülerin görsel yönetimi.

SaveSystem

- Methods: VeriKaydet(), VeriYukle() – PlayerPrefs ile çalışır.

Aktivite Diyagramı (Activity Diagram)

1. Oyun başlatılır
2. Oyuncuya yeni hayvan sunulur
3. Oyuncu hayvanı kutuya bırakır
4. Sistem kutudaki konumu kontrol eder
5. Aynı hayvan varsa birleştirme gerçekleşir
6. Yeni hayvan oluşturulur
7. Skor ve altın güncellenir
8. Kutu doluluk kontrolü yapılır
9. Oyun bitmişse skor gösterilir
10. Oyun devam ediyorsa yeni hayvan verilir

Teknik Detaylar ve Oyunun Akışı

Animals Merge oyunu, Unity oyun motoru kullanılarak C# diliyle geliştirilmiştir. Unity'nin komponent tabanlı mimarisi sayesinde oyun mantığı, kullanıcı arayüzü, fiziksel etkileşimler ve animasyon sistemleri birbirinden bağımsız ve sürdürülebilir bir yapı ile inşa edilmiştir.

Genel Yapı:

- Unity sürümü kullanılmıştır.
- 2D oyun şablonu üzerinden inşa edilmiştir.
- Proje üç ana sahneden oluşmaktadır: Ana Menü, Oyun Sahnesi ve Market Sahnesi.
- Her sahnede Canvas sistemi ile UI öğeleri yönetilmiştir.

Kodlama Yaklaşımı:

- OOP (Nesne Yönelimli Programlama) prensipleri uygulanmıştır.
- Component-based mimari ile her objeye ayrı davranışlar atanmıştır.
- ScriptableObject yapıları ile hayvan verileri merkezi olarak tanımlanmıştır.
- Tek Sorumluluk Prensibi (Single Responsibility) korunarak her script yalnızca bir işlev üstlenmiştir.

Oyunun Akışı:

1. Oyuncu uygulamayı başlattığında Ana Menü sahnesi açılır.
2. "Başla" butonuna tıkladığında oyun sahnesine geçilir.
3. Rastgele 4 hayvan oluşturulur ve kullanıcıya sunulur.
4. Kullanıcı bu hayvanları kutuya bırakır.
5. Eğer aynı türden 2 hayvan yan yana veya üst üste gelirse otomatik birleşme gerçekleşir.
6. Birleşme sonucu skor ve altın artar.
7. Kullanıcı kutunun dolmasını engelleyerek mümkün olan en uzun süre boyunca oyunu sürdürmeye çalışır.
8. Kutu dolarsa oyun biter ve skor/altın bilgisi gösterilir.
9. Tüm veriler PlayerPrefs ile local olarak kaydedilir.

Bu yapı sayesinde kullanıcı, yüksek performansla çalışan, sade ama tatmin edici bir oyun deneyimi yaşar. Ayrıca geliştirici tarafında da kodun sürdürülebilirliği ve yeniden kullanılabilirliği yüksek tutulmuştur.

Market Sistemi ve Altın Ekonomisi

Oyun içi ekonomi, oyuncunun ilerlemesini görsel olarak özelleştirebilmesini ve harcadığı sürenin karşılığını hissetmesini sağlamak amacıyla kurulmuştur. Her başarılı hayvan birleşimi sonucunda kullanıcıya belirli miktarda altın verilir. Bu altınlar oyun içi markette harcanabilir.

Altın Kazanımı:

- Her birleşme +1 ila +5 altın arasında ödül sağlar.
- Zincirleme birleşmeler ekstra bonus getirir.
- Günlük girişler ileride ödül sistemine dönüştürülebilir.

Market Sistemi:

- Kullanıcı market ekranında farklı kostümler görebilir.
- Kostümler yalnızca görsel amaçlıdır (pay-to-win yok).
- Satın alınan kostümler anında aktif hale gelir.
- Oyuncunun toplam altın bakiyesi market ekranının üst kısmında gösterilir.
- Satın alma işlemi sonrasında bir "başarı animasyonu" oynatılır.

Gelecekteki Market Genişletmeleri:

- Arka plan temaları (mevsimler, uzay, gece-gündüz)
- Özel hayvan efektleri (parlayan, dumanlı vb.)
- Altınla değil yalnızca başarı ile kazanılan nadir içerikler

Bu sistem sayesinde oyuncunun elde ettiği skorlar ve birleşmeler yalnızca bir sayı değil, aynı zamanda kişisel ilerlemesinin görsel bir karşılığı olur. Bu da oyun bağlılığını ve tekrar oynanabilirliği artırır.

Test Süreci ve Sonuç Kısmı

Projenin son aşamasında çeşitli cihazlar ve kullanıcı grupları ile test süreci gerçekleştirilmiştir. Bu süreç boyunca oyun içi hatalar, kullanıcı alışkanlıkları ve performans gözlemleri sistematik olarak raporlanmıştır.

Test Edilen Cihazlar:

- Samsung Galaxy A52 (Android 12)
- Xiaomi Redmi Note 10 (Android 11)
- Huawei P30 Lite (Android 10)
- Android Studio Emulator (Pixel 4 – API 29)

Yapılan Teknik Testler:

- FPS ölçümleri (ortalama: 58-60 FPS)
- Bellek tüketimi (maks. 220 MB)
- Birleşme gecikmesi (ortalama: 30-40 ms)
- Market işlem hızı (maks. 100 ms)

Gözlemlenen Hatalar ve Çözümleri:

- Bazen hayvanlar birleşme sonrası görünmüyordu → Sprite Renderer yeniden çağrıldı
- Altın verisi zaman zaman kaydedilmiyordu → PlayerPrefs save on quit yapıldı
- Market'te aynı kostüm iki kez alınabiliyordu → Satın alınan içerikler disable edildi

Kullanıcı Geri Bildirimleri:

- "Oyun çok sade ve anlaşılır."
- "Hayvanlar çok sevimli, kız kardeşim bile bırakmadan oynadı."
- "Birleşme efekti çok tatmin edici."
- "Marketin büyümesini isteriz."

Geliştirme Süreci Aşamaları

Animals Merge projesinin geliştirilmesi, planlı ve modüler bir yaklaşımla gerçekleştirilmiştir. Projenin ilk aşamasında oyun fikri netleştirilmiş, hedef kitle belirlenmiş ve temel mekanikler tasarlanmıştır. Bu süreçler proje planlamasında temel yol haritası olarak kullanılmıştır. Aşağıda, geliştirme sürecinin haftalara göre aşamalı dökümü yer almaktadır:

1. ve 2. Hafta – Fikir ve Planlama:

- Oyun türü ve teması belirlendi (Bulmaca / Merge).
- Oyun mekanikleri ve temel işleyiş kurgulandı.
- Hayvanlar sıralaması (kurbağa → panda) oluşturuldu.
- Unity 2D template projesi başlatıldı.

3. ve 4. Hafta – Mekanik Geliştirme:

- Hayvan nesneleri oluşturuldu.
- Kutu sistemi (ızgara yapısı) geliştirildi.
- Sürükle-bırak fonksiyonelliği eklendi.
- Birleşme kontrolleri ve skor sistemi yazıldı.

5. ve 6. Hafta – UI ve Market:

- Ana menü, oyun ekranı ve market tasarımları tamamlandı.
- Altın kazanma ve marketten alışveriş sistemi geliştirildi.
- Kostüm sistemi eklendi.
- PlayerPrefs ile local veri kaydetme kodlandı.

7. ve 8. Hafta – Test ve Son Rötuşlar:

- Cihaz testleri ve performans ölçümleri yapıldı.
- Hatalar giderildi, son kullanıcı testleri uygulandı.
- Dokümantasyon süreci başlatıldı.
- APK dosyası oluşturuldu.
- Bu süreçte agile yaklaşıma benzer biçimde her hafta geri bildirimlerle iteratif olarak güncellemeler yapılmıştır.

Olası Geliřtirmeler ve İleri Sürüm Planları

Animals Merge oyunu, ilk sürümde temel özelliklere sahip basit bir yapı üzerine kurulmuştur. Ancak geliştirilebilirlik gözetilerek esnek bir sistem mimarisi ile yazılmıştır. Bu sayede ilerleyen sürümlerde oyun birçok yeni özelliğe kolaylıkla kavuşturulabilir.

Planlanan Yeni Özellikler:

- Günlük Görevler: Her gün tamamlanabilecek mini görevlerle ödüller kazanma.
- Online Skor Tabloları: Firebase ile global sıralamalar.
- Multiplayer Modu: Gerçek zamanlı rakiplerle kutu yönetimi.
- Yeni Hayvanlar: Panda sonrası mitolojik hayvanlar (Ejderha, Anka kuşu vs.)
- Zamanlı Mod: 60 saniyede en fazla puanı toplama.
- Görev Defteri: Seviye bazlı hedeflerin olduğu modül.

Geliştirme Ortaklığı ve Yayın Planı:

Gelecekte bu projenin Google Play Store üzerinden yayınlanması hedeflenmektedir. Bunun için reklam sistemi, kullanıcı gizliliği politikası, veri yedekleme gibi destekleyici yapıların entegre edilmesi planlanmaktadır. Ayrıca içerik üreticileriyle işbirliği yapılarak oyunun tanıtımı artırılabilir.

Hazırlayanlar

Abdullah Enes Karapınar - 205541059

Yunus Emre Özen - 215542001