



T.C. KIRKLARELİ ÜNİVERSİTESİ
TEKNİK BİLİMLER MESLEK YÜKSEKOKULU
BİLGİSAYAR PROGRAMCILIĞI BÖLÜMÜ
MOBİL PROGRAMLAMA DERSİ PROJE ÖDEVİ

Projeyi Veren:

Ad-Soyad: Nadir SUBAŞI

Projeyi Hazırlayan:

Ad-Soyad: Betül SAĞLIK

Numara:1247008036

İÇİNDEKİLER:

İÇİNDEKİLER.....	2
Kısaltmalar.....	3
1.Giriş.....	4
1.1.Kompleks UI ve Animasyon Nedir?.....	4
2.Avantajlar ve Dezavantajlar.....	5
3.Animasyon Türleri.....	6
4.Performans ve Optimizasyon.....	6
5.Easing Eğrileri ve Psikolojik Etkiler.....	7
6.Kullanıcı Kısıtlamalarına Duyarlılık.....	7
7.Flutter Animasyon Araçları ve Kullanım Alanları.....	8
7.1Kullanım Önerileri.....	9
8.Gelecek Trendler ve Yenilikler.....	10
9.Sonuç.....	10
10.Kaynakça.....	11

KISALTMALAR

UI : User Interface

UX :User Experience

FPS :Frames Per Second

SVG :Scalable Vector Graphics

AR :Augmented Reality

VR :Virtual Reality

AI :Artificial Intelligence

JSON : JavaScript Object Notation

GİRİŞ

Kompleks UI ve Animasyonlar

Kompleks UI ve Animasyon Nedir?

Kompleks kullanıcı arayüzü (UI), bir uygulamanın veya web sitesinin kullanıcıyla etkileşime geçtiği çok katmanlı, dinamik ve görsel olarak zengin arayüzleri ifade eder. Bu tür arayüzler yalnızca düğmeler ve metin alanlarından ibaret değildir; kaydırılabilir paneller, animasyonlu menüler, dinamik içerikler, etkileşimli öğeler ve kullanıcı davranışına göre değişebilen komponentler içerir. Günümüzde kullanıcıların beklentileri çok daha yüksek olduğundan, basit ve statik arayüzler artık yeterli değildir. İnsanlar, uygulamaların hızlı, sezgisel ve estetik olarak tatmin edici olmasını bekler.

Animasyonlar ise bu arayüzlerde hareket ve geçiş efektleri sağlayan görsel öğelerdir. Örneğin bir butona basıldığında büyüme efekti, menülerin yumuşak kayması veya sayfa geçişlerinde animasyonlu geçişler buna örnek teşkil eder. Animasyonlar sadece görsellik sağlamakla kalmaz; kullanıcıya işlemin başarılı olduğunu göstermek, dikkat edilmesi gereken alanları vurgulamak, uygulamanın yönlendirmesini kolaylaştırmak ve etkileşim hissini güçlendirmek için kritik öneme sahiptir. Modern uygulamalarda animasyonlar, aynı zamanda marka kimliğinin bir parçası hâline gelmiştir; doğru ve tutarlı animasyonlar uygulamayı daha profesyonel ve güvenilir gösterir.

Kompleks UI ve animasyonlar genellikle birlikte kullanılır. Animasyonlar kullanıcıya rehberlik eder, geri bildirim sağlar ve uygulamada geçirilen süreyi artırır. Örneğin bir e-ticaret uygulamasında ürün geçişleri animasyonlu yapılırken, kullanıcı hangi ürünün seçildiğini ve sepete eklenip eklenmediğini anında görebilir. Eğitim uygulamalarında ise animasyonlar, kullanıcıyı motive eder ve ilerlemeyi görselleştirir, böylece kullanıcı bağlılığı artar.

Günümüzde kompleks UI tasarımları yalnızca estetik değil, aynı zamanda erişilebilirlik ve performans odaklı olmak zorundadır. Hızlı ve akıcı animasyonlar, kullanıcı deneyimini geliştirirken, aşırı karmaşık veya optimize edilmemiş animasyonlar performans sorunlarına ve kullanıcı memnuniyetsizliğine yol açabilir. Bu nedenle modern arayüz tasarımında animasyonların kullanım sıklığı, hızı ve şekli dikkatle planlanmalı, kullanıcı tercihlerine göre özelleştirilebilir hâle getirilmelidir.

Özetle, kompleks UI ve animasyonlar, bir uygulamanın yalnızca “çalışması” değil, kullanıcıyla etkileşim kurması ve deneyimi zenginleştirmesi için kritik öneme sahiptir. Doğru tasarlandığında, bu kombinasyon hem estetik hem işlevsel başarıyı artırır, kullanıcıya rehberlik eder ve uygulamanın profesyonel görünmesini sağlar.

Kullanım Alanları

Kullanıcı Deneyimini İyileştirmek (UX):

Animasyonlar, kullanıcıya hangi işlemi yapması gerektiğini veya bir işlemin tamamlandığını görsel olarak göstermek için kullanılır. Örneğin, bir butona basıldığında renk değişimi veya hafif bir büyüme animasyonu, kullanıcıya işlemin algılandığını hissettirir. Formlar, onay ekranları ve yükleme animasyonları da bu kategoriye dahildir; örneğin bir form başarıyla gönderildiğinde animasyonlu bir onay işareti, kullanıcıya işlemin tamamlandığını bildirir.

Estetik ve Marka Değeri:

Akıcı, modern ve estetik arayüzler, uygulamanın profesyonel ve güvenilir görünmesini sağlar. Duolingo'daki ders tamamlama kutlama animasyonları, kullanıcıya hem eğlenceli bir deneyim sunar hem de markayı öne çıkarır. Sosyal medya uygulamalarında paylaşımların veya beğenilerin animasyonlu görünmesi, görsel zevki artırır ve uygulamanın marka değerini pekiştirir. E-ticaret uygulamalarında ürünlerin öne çıkan özelliklerinin animasyonla vurgulanması, kullanıcı algısını olumlu yönde etkiler.

Rehberlik ve Odak:

Animasyonlar, kullanıcıya dikkat etmesi gereken alanları veya önemli adımları vurgulamada yardımcı olur. Örneğin Instagram'daki yeni mesaj bildirimleri veya e-ticaret sitelerindeki ürün geçiş animasyonları kullanıcıyı yönlendirir ve odaklanmayı sağlar. Eğitim uygulamalarında yeni ders içeriği geldiğinde animasyonlu bir bildirim, kullanıcının dikkatini hızlıca çeker.

Etkileşim ve Bağlılık:

Dinamik ve eğlenceli arayüzler, kullanıcıların uygulamada daha uzun süre kalmasını sağlar. Spotify'da albüm kapağı animasyonu, kullanıcıyı uygulamada daha fazla vakit geçirmeye teşvik eder. Oyun uygulamalarında karakter hareketleri veya ödül animasyonları, kullanıcıyı motive eder ve sürekli etkileşimi artırır. Eğitim uygulamalarında görev tamamlamaya bağlı efektler, kullanıcı bağlılığını güçlendirir ve öğrenme sürecini daha eğlenceli hâle getirir.

Ek Kullanım Alanları:

- Formlar ve Onay İşlemleri:** Ödeme tamamlandığında animasyonlu onay simgesi.
- Navigasyon ve Geçişler:** Sayfa geçişlerinde kayma veya yumuşak animasyonlar.
- Bildirimler ve Uyarılar:** Yeni mesaj veya kritik uyarılar animasyonlu gösterim.
- Oyun ve Eğlence:** Karakter hareketleri, skor değişimleri, ödül animasyonları.
- E-ticaret:** Ürün tanıtımları ve öne çıkan özelliklerin animasyonla vurgulanması.
- Eğitim:** Görev tamamlamaları, ders ilerlemeleri veya ödül animasyonları.

Avantajlar ve Dezavantajlar

Avantajlar:

- Akıcı ve görsel olarak zengin arayüzler:** Kullanıcıya görsel bir deneyim sunar.
- Kullanıcı geri bildirimi:** Hangi adımın tamamlandığını net olarak gösterir.
- Marka değeri:** Modern ve dikkat çekici tasarımlar markayı güçlendirir.
- Etkileşim ve bağlılık:** Dinamik öğeler kullanıcıyı uygulamada tutar.

Dezavantajlar:

- Performans sorunları:** Çok animasyon veya karmaşık UI düşük donanımlı cihazlarda yavaşlamaya yol açabilir.
- Geliştirme zorluğu:** Karmaşık arayüzler zaman ve uzmanlık gerektirir.
- Karmaşıklık ve kafa karışıklığı:** Yanlış animasyonlar kullanıcıyı şaşırtabilir.
- Erişilebilirlik:** Görme engelli veya hareket hassasiyeti olan kullanıcılar için optimizasyon gerekebilir.

Animasyon Türleri

Geçiş (Transition) Animasyonları: Ekranlar arası geçişleri akıcı hâle getirir.

Örnek: Spotify albüm kapağı geçişleri.

Geri Bildirim (Feedback) Animasyonları: İşlemin başarılı veya hatalı olduğunu görsel olarak iletir.

Örnek: Buton renk değişimi, ilerleme çubuğu.

Odak ve Vurgu (Focus/Highlight) Animasyonları: Kullanıcının dikkat etmesi gereken öğeleri ön plana çıkarır.

Örnek: Yeni mesaj simgesi sallanması.

Süreç ve Progres (Progress/Loading) Animasyonları: Bekleme süresini gösterir ve sıkılmayı önler.

Örnek: Dönen yükleme ikonları.

Etkileşimli (Interactive) Animasyonlar: Kullanıcının doğrudan etkileşimi sağlar.

Örnek: Sürükle-bırak animasyonları.

Mikro Animasyonlar: Küçük ve hızlı geri bildirim sağlar.

Örnek: Buton basıldığında hafif büyüme.

Performans ve Optimizasyon

Performansı etkileyen faktörler:

- Aşırı widget kullanımı ve gereksiz animasyonlar
- Yüksek çözünürlüklü görseller ve SVG animasyonları
- Ani ve karmaşık animasyonlar
- Yanlış kod yapısı

Optimizasyon Yöntemleri:

- AnimatedBuilder ve CustomPainter:** Gereksiz rebuild'leri engeller.
- Lazy Loading ve görsel optimizasyon:** Sadece görünür alanı yükler.
- FPS kontrolü:** Animasyon kare hızını optimize eder.
- Modüler widget tasarımı:** Tekrar kullanılabilir parçalar performansı artırır.
- Flutter DevTools ile profiling:** Performans darboğazlarını tespit eder.

Ek Teknik Detaylar:

- Opaklık ve clipping optimizasyonu GPU yükünü azaltır.
- 60 FPS hedefi, animasyonların akıcı görünmesini sağlar.
- Aşırı optimizasyon, animasyonların etkisini azaltabilir; denge sağlanmalıdır.

Easing Eğrileri ve Psikolojik Etkiler

- Ease-in:** Başta yavaş, sonda hızlı.

Örnek: Menü açılırken başlangıçta yavaş.

- Ease-out:** Başta hızlı, sonda yavaş.

Örnek: Kapanış animasyonu daha yumuşak.

- Ease-in-out:** Başta ve sonda yavaş, ortada hızlı.

Örnek: Sayfa geçişleri, albüm kapağı animasyonu.

Kullanıcı Kısıtlamalarına Duyarlılık

- Reduce Motion ayarı ile hızlı animasyonları sınırlama
- Kullanıcı tercihlerine göre animasyonları kapatma/azaltma
- Alternatif görsel veya metinsel geri bildirim
- Renk körlüğü veya düşük kontrast farkları için optimizasyon

Flutter Animasyon Araçları ve Kullanım Alanları

Flutter, modern uygulamalarda akıcı ve etkileşimli kullanıcı arayüzleri oluşturmak için farklı animasyon araçları sunar. Bu araçlar, basitten karmaşığa, performans optimizasyonundan etkileşimli animasyonlara kadar geniş bir yelpazede kullanılabilir.

Uygulama Örneği: Spotify Now Playing Ekranının Flutter’da Modellenmesi

Bu ödev kapsamında, popüler bir uygulama olan Spotify’ın “Now Playing (Şu An Çalınan)” ekranı referans alınarak Flutter’da benzer bir kullanıcı arayüzü ve animasyon deneyimi oluşturulmaya çalışılmıştır. Spotify, akıcı geçişleri, yumuşak animasyonları ve dinamik etkileşimleriyle modern mobil tasarıma güçlü bir örnek oluşturmaktadır. Bu nedenle çalışma boyunca, uygulamanın en karakteristik animasyonlarından olan albüm kapağının yavaşça büyüüp küçülmesi, butonların yumuşak bir şekilde aktif hâle gelmesi ve arka plan geçişlerinin blur efektiyle sağlanması temel alınmıştır.

Flutter’da bu animasyonları gerçekleştirmek için AnimatedContainer, AnimatedOpacity, AnimatedBuilder, TweenAnimationBuilder ve Transform widget'ları kullanılmıştır. Ekrandaki sürekli hareket eden veya tekrarlanan animasyon efektleri için AnimationController ve CurvedAnimation yapıları tercih edilmiştir. Böylece Spotify ekranındaki akıcı, kesintisiz ve kullanıcı etkileşimine duyarlı animasyonların Flutter’da birebir olmasa da benzer mantıkla oluşturulabileceği gösterilmiştir.

1. AnimatedContainer

-Basit widget değişimleri için hızlı çözüm sağlar.

Örnek: Buton rengi, boyutu veya kenar yuvarlama değişimi.

Avantaj: Kod basit, hızlı ve performans dostu.

Sınırlama: Karmaşık animasyonlar ve kullanıcı etkileşimleri için yeterli değildir.

2. AnimatedBuilder

-Karmaşık ve performans gerektiren animasyonlar için idealdir.

-Kullanıcı etkileşimine bağlı animasyonlarda detaylı kontrol sağlar.

Örnek: Albüm kapağı döndürme, menü açılış animasyonları, özel geçiş efektleri.

Avantaj: Yalnızca değişen widget'ı rebuild eder, gereksiz kaynak kullanımını önler.

3. Lottie

-JSON tabanlı animasyonları Flutter projelerine entegre eder.

-Tasarımcıdan gelen animasyonları doğrudan kullanmaya uygundur.

Örnek: Yükleme ekranları, kutlama animasyonları, görev tamamlama geri bildirimleri.

Avantaj: Kod yazmadan etkileyici görseller ekleyebilir, hızlı prototipleme sağlar.

Sınırlama: Çok fazla veya yüksek çözünürlüklü animasyonlar performansı etkileyebilir; etkileşim sınırlıdır.

4. Rive

-Etkileşimli ve ileri düzey animasyonlar oluşturur; kullanıcı etkileşimine bağlı tetiklenebilir.

Örnek: Bir butona basıldığında karakter animasyonunun tetiklenmesi, interaktif simgeler veya menü animasyonları.

Avantaj: Yüksek etkileşimli animasyonlar oluşturulabilir, performans dostudur.

Sınırlama: Öğrenme eğrisi yüksektir, tasarım süreci biraz daha karmaşıktır.

Kullanım Önerileri

-Basit geri bildirimler ve widget değişimleri için AnimatedContainer,

-Performans gerektiren ve etkileşimli animasyonlar için AnimatedBuilder,

-Tasarımcıdan gelen hazır animasyonlar için Lottie,

-Kullanıcı etkileşimine bağlı dinamik animasyonlar için Rive tercih edilmelidir.

Projelerde birden fazla araç bir arada kullanılabilir; örneğin ana menü animasyonları 'AnimatedBuilder' ile, yükleme ekranı animasyonları 'Lottie' ile entegre edilebilir. Animasyonların FPS, render optimizasyonu ve mobil cihaz kaynak kullanımı açısından test edilmesi önerilir.

Örnek Kod:

AnimatedContainer(

duration: Duration(milliseconds: 300),

color: isPressed ? Colors.green : Colors.blue,

width: 200,

```
height: 50,  
child: Center(child: Text("Tamamla")),  
);
```

Kod Açıklaması:

Bu kodda basit bir animasyonlu buton örneği verilmiştir.

-Kullanıcı durumu (isPressed) değişirse, renk mavi -> yeşil (veya tersi) yumuşak bir şekilde animasyonla değişir.

-Boyut veya kenar yuvarlama gibi diğer özellikler değişseydi, onlar da animasyonlu olurdu.

-Avantajı: Kod basit ama görsel olarak etkileşimli ve dinamik bir UI sağlar.

Gelecek Trendler ve Yenilikler

-Mikro animasyonlar: Küçük ve hızlı geri bildirimler

-AR/VR ve AI destekli animasyonlar: Kişiselleştirilmiş arayüzler

-Kullanıcı testi ve UX ölçümleri: Etkili animasyonların belirlenmesi

-Kod ve tasarım entegrasyonu: Estetik ve işlevselliğin birlikte optimize edilmesi

-Performans karşılaştırmaları: Farklı araçların mobil uyumluluğu ve FPS optimizasyonu

Sonuç

Kompleks UI ve animasyonlar, modern mobil ve web uygulamalarında estetik ve işlevsel başarının temel taşlarıdır. Kullanıcıya rehberlik eder, geri bildirim sağlar ve uygulama ile etkileşimi güçlendirir. Örneğin, bir görevin tamamlandığını gösteren kutlama animasyonları veya sayfa geçişlerinde akıcı efektler, kullanıcıya işlemin algılandığını hissettirir ve uygulamada geçirilen süreyi artırır.

Performans ve erişilebilirlik optimizasyonu, tüm kullanıcılar için erişilebilir deneyim sağlar. Düşük donanımlı cihazlarda akıcı deneyim sunmak, gereksiz animasyonlardan kaçınmak, FPS kontrolü yapmak ve Opacity/Clip işlemlerini sınırlamak kritik öneme sahiptir. Hareket hassasiyeti olan kullanıcılar için animasyonları azaltma veya kapatma seçenekleri sunmak modern tasarımın kullanıcı odaklı bir parçasıdır.

Gelecekte AR/VR ve yapay zekâ destekli animasyonlar, mobil UI'yi daha etkileşimli ve kişiselleştirilmiş hâle getirecek; kullanıcı davranışına göre uyarlanan animasyonlar, deneyimi sezgisel ve akıcı hâle getirecek. Örneğin, alışveriş uygulamalarında kullanıcı

ilgisine göre ürünlerin animasyonlu şekilde vurgulanması veya eğitim uygulamalarında öğrencinin ilerlemesine göre kişiselleştirilmiş geri bildirim animasyonları kullanıcı deneyimini üst seviyeye taşıyacaktır.

Özetle, kompleks UI ve animasyonlar, modern uygulamaların estetik ve işlevsel başarısının ayrılmaz bir parçasıdır. Doğru tasarlandığında ve optimize edildiğinde, kullanıcı memnuniyetini artırır, etkileşimi yükseltir ve uygulamanın marka değerini güçlendirir.

KAYNAKÇA

<https://api.flutter.dev/flutter/widgets/AnimatedBuilder-class.html>

<https://flutter.dev/showcase/rive>

<https://rive.app>

<https://www.smashingmagazine.com>

<https://medium.com/flutter>