



# **Sistem Uygulama Aşaması: Bilgisayar Mühendisliği Bakış Açısı**

Yazılım ve donanım entegrasyonunu içeren sistemlerin uygulanması aşamasında temel prensipleri, stratejileri ve önemli aşamaları ele alınacaktır.

**Öğr. Gör. Erhan AKAGÜNDÜZ**

# Sistem Geliştirme Yaklaşımları

## Şelale Modeli

Sıralı, adımlar halinde ilerleyen klasik yöntem.

## Çevik (Agile) Model

Esnek, iteratif, sürekli geri bildirim odaklı.

## Spiral Model

Risk odaklı, prototipleme ve geliştirme döngüleri.

# Sistem Tasarımı: Konsept ve Planlama







# Donanım ve Yazılım Entegrasyonu

## Donanım Seçimi

Performans, uyumluluk, maliyet faktörleri.

## Yazılım Geliştirme

Uygulama kodu yazma, test etme, hata ayıklama.

## Entegrasyon ve Test

Donanım ve yazılım bileşenlerini bir araya getirmek.

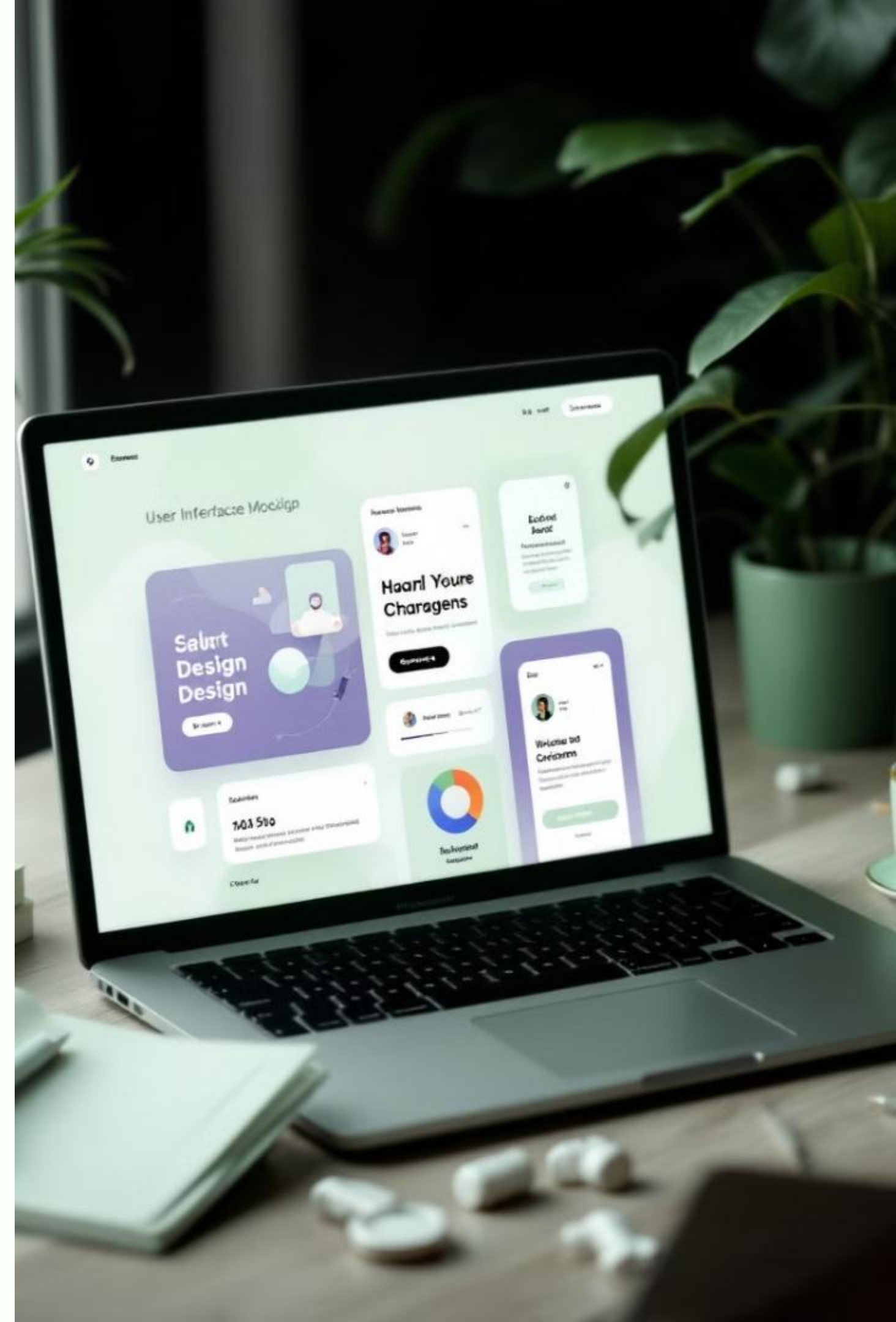
# Veri Tabanı ve Kullanıcı Ara Yüzü Tasarımı



Veri modeli, tablolar, ilişkiler, sorgu optimizasyonu.



Kullanıcı deneyimi odaklı tasarım, gezinti, etkileşim.





# Uygulama Geliştirme Metodolojileri

1

Suçluluk (Waterfall) modeli:  
Adımlar halinde ilerleme.

2

Çevik (Agile): Esnek, iteratif,  
hızlı geri bildirim.

3

Spiral Model: Risk odaklı, prototipleme döngüleri.



# Test ve Doğrulama Süreçleri

1

## Birim Testleri

Tekil kod parçalarının doğruluğunu test etme.

2

## Entegrasyon Testleri

Farklı bileşenlerin birlikte çalışabilirliğini kontrol etme.

3

## Sistem Testleri

Tüm sistemin işlevselliğini ve performansını değerlendirme.

4

## Kabul Testleri

Kullanıcıların sistemin beklentilere uygunluğunu onaylamaları.







# Olası Teknik Zorluklar ve Çözümler

## Uyumluluk Sorunları

Farklı bileşenler arasındaki uyumsuzluklar.

## Performans Sorunları

Yüksek iş yükü altında sistemin yavaşlaması.

## Güvenlik Açıkları

Yetkisiz erişim veya veri ihlalleri riski.



# Performans Optimizasyonu ve İyileştirme

1

## Profilleme

Sistemin darboğazlarını belirlemek.

2

## Kod Optimizasyonu

Daha verimli kod yazmak.

3

## Veri Tabanı Optimizasyonu

Sorgu performansını artırmak.

4

## Donanım Yükseltmesi

Daha güçlü işlemciler veya daha fazla bellek.



# Sistemin Sürdürülebilirliği ve Gelişimi



Sistemin güncelliğini korumak, yeni gereksinimlere uyum sağlamak ve güvenlik açıklarını gidermek için sürekli bakım, güncelleme ve geliştirme planlaması önemlidir. Bu aşamada çevik yaklaşımlar ve sürekli entegrasyon/teslim yöntemleri kullanışlıdır.