



# BİL440 YZ Destekli Yazılım Geliştirme

2025-26 Güz, Final Projesi

## AI-Augmented Software Lifecycle Project

**Son teslim tarihi Final :** 13/01/2026 23:59 Sisteme son yüklemeye  
14/01/2026 Çarşamba günü - Duyurulacak saatlerde Sunum/demo gösterim (YÜZ YÜZE)

**Bütünleme :** 27/01/2026 23:59

Bu final projesinin amacı, öğrencilerin **büyük dil modelleri ve YZ kod asistanlarını** yazılım geliştirme yaşam döngüsünün farklı aşamalarında **bilinçli, eleştirel ve belgelenmiş** biçimde kullanabilme becerilerini değerlendirmektir.

Bu proje, *sadece çalışan bir yazılım üretmeyi değil; YZ ile birlikte düşünülmüş, sorgulanmış ve gerekçelendirilmiş* bir geliştirme sürecini ölçmesi amaçlanmıştır. Bu proje ile öğrencinin;

- YZ ile *kod yazan* değil
- YZ ile *yazılım geliştiren*
- YZ'yi gerektiğiinde *reddedebilen*

bir mühendis refleksi kazanmış olması beklenmektedir.

Proje Öğrenci/öğrenciler (Labda çalışığınız 2 kişilik eşli-programlama grupları ya da labda grup arkadaşı olmayanlar için tek), aşağıda kısaca açıklanan üç yazılım probleminden birisini,

- Analiz
- Tasarım
- Geliştirme
- Test
- Değerlendirme

aşamalarının **tamamında en az iki farklı YZ kod asistanı** kullanarak geliştireceklerdir.

## YZ Araç Kullanımı (Zorunlu)

**En az iki farklı** YZ kod asistanı:

- GitHub Copilot
- ChatGPT
- Claude Code
- FireBase Studio
- Gemini Code Assist

Her aracın **hangi aşamada, neden ve nasıl** kullanıldığı açıkça belirtilmelidir. Eğer başka bir araç kullanılmak istenirse ders koordinatörü bilgilendirilmelidir.

## Tasarım Çıktıları

- Kullanıcı hikâyeleri (AI-assisted)
- En az **1 UML diyagramı**
- Mimari karar açıklamaları:
  - AI önerisi*
  - İnsan kararı*ayrımı net biçimde yapılmalıdır

## Geliştirme ve Versiyonlama

- Kodlar GitHub üzerinden teslim edilecektir
- Commit mesajlarında aşağıdaki etiketlerden biri yer almalıdır:
  - [AI-generated]
  - [AI-assisted]
  - [Human-written]

## Test ve Hata Ayıklama

- AI ile üretilmiş:
  - Unit testler
  - Edge-case senaryoları
- En az **1 yanlış veya eksik AI çıktısı** bilinçli olarak raporlanmalıdır

## 4. AI Decision Log (Zorunlu)

Her grup aşağıdaki formatta bir *AI Decision Log* teslim edecektir:

Aşama	Kullanılan YZ	YZ Önerisi	Nihai Karar	Gerekçe

Bu tablonun temel amacı; “Bu kodu AI yazdı” demek değildir. “Bu kararı neden AI’ya bıraktık / bırakmadık?” sorusuna cevap vermektedir.

## 5. Etik, Güvenlik ve Lisans Değerlendirmesi

Raporda mutlaka ayrı bir bölüm olarak:

- Kod lisansı riski
- Veri gizliliği
- Güvenlik açıkları
- AI hallucination örnekleri

kısa ama net biçimde tartışılmalıdır.

## Teslim Edilecekler

1. GitHub Repository (kod + README)
2. Teknik Rapor (8-10 sayfa)
3. AI Decision Log (rapora dahil edilmesi gereken tablo)
4. Final Sunumu (PDF / Keynote / Microsoft PowerPoint)

## Final Sunumu

- Süre: **10 dk sunum + 5 dk soru = Toplam 15 dakika**
- Sunumda özellikle beklenen:
  - AI'nin güçlü olduğu anlar
  - Yanlıltıcı olduğu anlar
  - İnsan müdahalesinin kritik olduğu noktalar

## Değerlendirme Rubriği

Kriter	Ağırlık
YZ kullanım derinliği ve çeşitliliği	%25
Eleştirel analiz ve farkındalık	%30
Yazılımın teknik kalitesi	%20
Test, güvenlik ve etik boyut	%15
Raporlama ve sunum	%10

## Yasaklar ve Net Kurallar

- YZ çıktıları **doğrudan kopyalanamaz**
- Açıklamasız AI-generated kod **puan kaybı** sebebidir
- "Prompt paylaşılmadan" yapılan AI iddiaları geçersizdir
- Kodun *neden çalıştığını* açıklayamayan grup başarılı sayılmaz



# Proje #1 – Akıllı Görev ve Öncelik Yönetim Sistemi

## Amaç:

Kullanıcıların görevlerini yalnızca listeleyen değil; görevler arasındaki **öncelik, zaman baskısı ve bağımlılık ilişkilerini** analiz eden web tabanlı bir görev yönetim sistemi geliştirilmesi.

## Temel Gereksinimler:

- Kullanıcıya özel görev alanı
- Görev ekleme, silme ve güncelleme
- Her görev için:
  - Öncelik seviyesi (düşük / orta / yüksek)
  - Tahmini süre
  - Son teslim tarihi
- Görevler arasında bağımlılık tanımlanabilmesi  
(bir görev tamamlanmadan diğerini başlayamaz)

## Zorunlu Akıllı Davranış:

- Sistem, mevcut görevler üzerinden:
  - Gecikme riski taşıyan görevleri tespit etmeli
  - Öncelik çıkışmalarını analiz etmelidir
- En az bir yeniden önceliklendirme veya planlama önerisi yapay zekâ desteğiyle üretilmelidir.

# Proje #2 – Akıllı Doküman Arama ve Özetteme Sistemi

## Amaç:

Kullanıcının yüklediği dokümanlar üzerinde **anlamsal arama** yapabilen ve içeriğten **bağlama duyarlı özetler** üretebilen web tabanlı bir sistem geliştirilmesi.

## Temel Gereksinimler:

- Kullanıcı tarafından doküman yükleme (PDF veya metin)
- Dokümanların sistem içinde saklanması
- Doküman içeriği üzerinde anahtar kelimeye dayalı arama

## Zorunlu Akıllı Davranış:

- Kullanıcının doğal dilde sorduğu bir soruya:

- Doküman içeriğine dayalı cevap üretimi
  - Cevabın hangi doküman(lar) a dayandığının belirtilmesi
- Her doküman için:
    - Kısa özet
    - İsteğe bağlı detaylı özet üretimi

En az bir **yanlış veya eksik yapay zekâ çıktısı** bilinçli biçimde raporlanmalıdır.

## Proje 3 – Event-Driven Akıllı Bildirim ve İzleme Sistemi

### Amaç:

Belirli olaylara (event) bağlı olarak çalışan, kural tabanlı ve genişletilebilir bir **akıllı bildirim sistemi** geliştirilmesi.

### Temel Gereksinimler:

- Kullanıcı tarafından olay (event) tanımlanabilmesi
- Olaylara bağlı tetikleyici kurallar oluşturulması
- Kurallar gerçekleştiğinde bildirim üretilmesi
- Üretilen bildirimlerin kayıt altına alınması

### Zorunlu Akıllı Davranış:

- Sistem, geçmiş olaylara bakarak:
  - Anormal durumları tespit etmeye çalışmalı
  - Gereksiz veya aşırı bildirimleri azaltmaya yönelik öneriler sunmalıdır
- En az bir kural veya eşik değeri yapay zekâ desteğiyle önerilmelidir.

## Teknoloji Seçimi Hakkında

- **Programlama dili, framework, veritabanı, mimari yaklaşım ve altyapı seçimi serbesttir.**
- Projeler web tabanlı olacak şekilde tasarlanmalıdır.
- Değerlendirme, kullanılan teknolojiden bağımsız olarak:
  - Temel gereksinimlerin karşılanması
  - Yapay zekânın bilinçli ve eleştirel kullanımı
  - Geliştirme sürecinin açıklanabilirliği üzerinden yapılacaktır.