

# KOCAELİ ÜNİVERSİTESİ

## BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

### YAZILIM LABORATUVARI-I

## PROJE II:

### WEB TABANLI SOSYAL KÜTÜPHANE PLATFORMU

Proje Teslim Tarihi: 30.11.2025

## 1. Giriş ve Proje Tanımı

### 1.1. Proje Tanımı

Kullanıcıların kendi kişisel kitap ve film kütüphanelerini oluşturabildiği, içerikleri puanlayıp yorumlayabileceği ve sosyal akış (feed) üzerinden paylaşım yapabildiği web tabanlı bir sosyal platformun geliştirilmesi hedeflenmektedir.

### 1.2. Temel Amaçlar

Bu projenin tamamlanmasıyla aşağıdaki yetkinliklerin kazanılması ve hedeflerin gerçekleştirilemesi amaçlanmaktadır:

- Teknik Beceri:** Modern web programlama (Frontend/Backend ayrımı), RESTful API mimarileri ve veritabanı tasarımları/yönetimi.
- Kullanıcı Deneyimi (UX):** Dinamik, kullanıcı dostu ve mobil uyumlu bir web arayüzü geliştirme.
- Entegrasyon:** Harici API'ler (TMDb, Google Books) ile platformun veri kalitesini ve zenginliğini artırma.
- Profesyonel Gelişim:** Ekip çalışması, sürüm kontrol sistemi (Git) kullanımı ve analitik problem çözme.

## 2. Proje Mimarisi

Sistem, modern bir web uygulamasının gerektirdiği şekilde, sunucu (Backend) ve istemci (Frontend) katmanları birbirinden ayrılmış bir mimari üzerine kurulacaktır.

### 2.1. Frontend: Sayfa Bazlı Fonksiyonel İsterler

Arayüz, aşağıda listelenen sayfaları ve fonksiyonları eksiksiz olarak içermelidir.

#### 2.1.1 Giriş / Kayıt Ekranları

- Kayıt Ol Formu:** Kullanıcı adı, e-posta, şifre ve şifre tekrarı alanlarını içermelidir.
- Giriş Yap Formu:** E-posta ve şifre alanlarını içermelidir.

- **Hata Yönetimi:** Başarısız kayıt (örn: "Bu e-posta zaten kullanımda") veya giriş (örn: "E-posta veya şifre hatalı") durumlarında kullanıcıya net hata mesajları gösterilmelidir.
- **Şifre Sıfırlama:** "Şifremi Unuttum" akışı (kullanıcının e-postasına sıfırlama linki gönderilmesi) bulunmalıdır.

## 2.1.2 Ana Sayfa (Sosyal Akış - Feed)

- Bu sayfa, platforma giriş yapan kullanıcının "zaman tüneli" (timeline) olmalıdır. Akış, takip edilen kullanıcıların en yeniden en eskiye doğru sıralanmış son aktivitelerinden oluşur.
- Akış, basit metin satırları yerine, her bir aktivite için zengin içerikli "Aktivite Kartı" (Activity Card) bileşenleri şeklinde tasarlmalıdır.

### 2.1.2.1 Aktivite Kartı Yapısı

- Her aktivite kartı, kullanıcıya eylem hakkında tam bağlamı (kim, ne, hangi içerik) görsellerle birlikte vermelidir. Bir kartın standart yapısı şu bileşenleri içermelidir:
  - **Üst Bilgi (Header):**
    - **Kullanıcı Avatarı:** Aktiviteyi yapan kullanıcının küçük profil resmi.
    - **Kullanıcı Adı:** Kullanıcının adı (Profil sayfasına linkli).
    - **Aksiyon Metni:** Yapılan eylemin kısa açıklaması (örn: "...bir filmi oyladı.", "...bir kitabı hakkında yorum yaptı.").
    - **Tarih:** Aktivitenin ne kadar süre önce yapıldığı (örn: "3 saat önce").
  - **Ana İçerik (Body):**
    - Bu bölüm, aktivitenin türüne göre (aşağıda detaylandırılmıştır) görsel olarak farklılık gösterir. Temel amaç, içeriğin posterini/kapağını metinden daha öncé plana çıkarmaktır.
  - **Alt Bilgi (Footer) / Etkileşim:**
    - Kullanıcıların bu aktiviteye (yorumun kendisine veya puanlamaya) tepki vermesini sağlayan "Beğen" veya "Yorum Yap" butonları.

### 2.1.2.2 Aktivite Türlerinin Görsel Gösterimi

Metin tabanlı "[Kullanıcı A], [Film B]..." formatı yerine, her aktivite türü kendi görsel bileşenine sahip olmalıdır:

- **Puanlama Aktivitesi ("Rating"):**
  - **Gösterim:** Kullanıcı bir içeriğe puan verdiğiinde, kartta o içeriğin posteri (film) veya kapağı (kitap) büyük bir şekilde gösterilmelidir.
  - **Görsel Bileşen:** Posterin/kapağın altında, kullanıcının verdiği puan görsel yıldız (★★★★☆) veya rakamsal (8/10) olarak net bir şekilde belirtilmelidir.

- **Yorumlama Aktivitesi ("Review"):**
  - **Gösterim:** İçerik posteri/kapağı gösterilmelidir.
  - **Görsel Bileşen:** Metnin tamamı akışta gösterilmemelidir. Yorumun sadece ilk birkaç cümlesi (örn: 150-200 karakter) bir alıntı (excerpt) olarak gösterilir.
  - **Etkileşim:** Alıntıının sonunda "...daha fazlasını oku" linki ile yorumun tamamının okunabileceği içerik detay sayfasına yönlendirme yapılmalıdır.
- **Alt Bilgi (Footer) / Etkileşim:**
  - Kullanıcıların bu aktiviteye (yorumun kendisine veya puanlamaya) tepki vermesini sağlayan "Beğen" veya "Yorum Yap" butonları.

### 2.1.2.3 Sayfalandırma (Pagination)

Akış, (performans açısından kritik öneme sahip) sayfalandırılmış bir yapıda sunulmalıdır.

- Başlangıçta son 10-15 aktivite yüklenmelidir.
- Kullanıcı sayfanın sonuna geldikçe "Sonsuz Kaydırma" (Infinite Scroll) mantığı ile veya bir "Daha Fazla Yükle" butonu ile bir sonraki sayfanın aktiviteleri (sonraki 10-15) yüklenmelidir.

### 2.1.3 Arama & Keşfet Sayfası

- **Arama Çubuğu:** Kitap/film adı ile arama yapmayı sağlayan bir arama motoru bulunmalıdır. Sonuçlar (kapak, başlık, yıl) listelenmeli ve "İçerik Detay Sayfası"na yönlendirmelidir.
- **Vitrin Modülleri:** Platformda "En Yüksek Puanlılar" ve "En Popülerler" (en çok yorumla/listeye eklenmeye sahip) içeriklerin listelendiği dinamik bölümler olmalıdır.
- **Gelişmiş Filtreleme:** Kullanıcıların içerikleri Türe, Yılı ve/veya Puana göre filtreleyerek keşfetmesini sağlayan bir arayüz sunulmalıdır.

### 2.1.4 İçerik Detay Sayfası

- **İçerik Künyesi:** Seçilen kitabın/filmin tüm meta verileri (kapak görseli, özet, yıl, süre/sayfa sayısı, yönetmen/yazar listesi, tür listesi) gösterilmelidir.
- **Platform Puanı:** İçeriğe verilmiş tüm kullanıcı puanlarının ortalaması (örn: 7.8/10) ve toplam oy (puan) sayısı net bir şekilde gösterilmelidir.
- **Kullanıcı Eylem Butonları:**
  - Kullanıcının 1-10 arası puan verebileceği (veya verdiği puanı güncelleyebileceği) bir puanlama bileşeni.
  - İçeriğin türüne göre ("kitap" veya "film") ilgili kütüphane butonları ("Okudum", "Okunacak" / "İzledim", "İzlenecek") bulunmalı. Bu butonlar içeriği listeye eklemeli/çıkarmalıdır.
  - "Özel Listeye Ekle" butonu (tıklandığında kullanıcının mevcut özel listelerini açan bir menü).

- **Yorumlar Bölümü:**

- O içeriğe yapılmış tüm kullanıcı yorumları (kullanıcı adı, yorum metni, tarih) listelenmelidir.
- Kullanıcının yeni bir yorum ekleyebileceği bir metin alanı ve "Gönder" butonu bulunmalıdır.
- Kullanıcılar, *sadece kendi* yorumlarını düzenleyebilmeli veya silebilmelidir.

## 2.1.5 Kullanıcı Profili Sayfası (Kütüphanem)

- Kullanıcının temel bilgileri (kullanıcı adı, avatar, biyografi) gösterilmelidir.

- **Profil Sahipliği Durumu:**

- Kullanıcı *kendi* profiline bakıyorsa: "Profili Düzenle" (avatar/biyografi güncelleme) ve "Yeni Özel Liste Oluştur" butonları görünür olmalıdır.
- Kullanıcı *başka birinin* profiline bakıyorsa: "Takip Et" / "Takipten Çık" butonu görünür olmalıdır.

- **Kütüphane (Sekmeli Yapı):** Kullanıcının "İzlediklerim", "İzlenecekler", "Okuduklarım", "Okunacaklar" listeleri ayrı sekmederde gösterilmelidir.

- **Özel Listeler:** Kullanıcının oluşturduğu özel koleksiyonlar (örn: "En İyi Bilimkurgu Filmlerim") listelenmelidir.

- **Son Aktiviteler:** Kullanıcının yaptığı son yorumların ve verdiği puanların bir listesi bulunmalıdır.

## 2.2. Backend: Teknik Mimarisi ve Sorumlulukları

Backend, arayüzün ihtiyaç duyduğu veriyi sağlayan ve iş mantığını yürüten API katmanıdır.

### 2.2.1 Veri Kaynağı: Harici API Entegrasyonu

- Platformdaki tüm film ve kitap meta verileri, aşağıda örnek olarak verilen bir servisten çekilmelidir. Manuel veri girişi yapılmayacaktır.

- **Filmler için Örnek Kaynak:** The Movie Database (TMDb):

- API Dokümanı: <https://developer.themoviedb.org/docs>
- Çekilecek Veriler: Film başlığı, özeti, yayın yılı, yönetmen, oyuncular, türler, kapak görseli URL'si.

- **Kitaplar için Örnek Kaynak:** Google Books API , Open Library API

- Google Books API Dokümanı:  
<https://developers.google.com/books/docs/v1/using>
- Open Library API Dokümanı:  
<https://openlibrary.org/developers/api>

- Çekilecek Veriler: Kitap başlığı, yazar(lar), açıklama, sayfa sayısı, kapak görseli URL'si.

### **3. Veritabanı Tasarımı**

Kullanılacak veritabanı modelinin, projenin tüm gereksinimlerini karşılayan, performans dengesini gözeten bir model olmalıdır.

Veritabanı için teknoloji seçimleriniz (PostgreSQL, MySQL, vb.), isimlendirme standartlarınız (örn: İngilizce vs. Türkçe tablo adları) veya belirli performans optimizasyonlarına yaklaşımınızda esneklik payınız bulunmaktadır.

Önemli olan, projenin fonksiyonel isterlerini eksiksiz karşılayan, tutarlı ve verimli bir veritabanı yapısı oluşturmaktır.

#### **Proje Teslimi**

- Rapor IEEE formatında (önceki yıllarda verilen formatta) 4 sayfa, akış diyagramı veya yalancı kod içeren, özet, giriş, yöntem, deneysel sonuçlar, sonuç ve kaynakça bölümünden oluşmalıdır. Raporda oluşturulan veri tabanına ait class diyagramları ve use case diyagramları bulunmalıdır.
- Dersin takibi projenin teslimi dahil edestek2.kocaeli.edu.tr sistemi üzerinden yapılacaktır. edestek2.kocaeli.edu.tr sitesinde belirtilen tarihten sonra getirilen projeler kabul edilmeyecektir.
- Proje ile ilgili sorular edestek2.kocaeli.edu.tr sitesindeki forum üzerinden Arş. Gör. Abdurrahman GÜN veya Arş. Gör. Kadir KESİMAL'a sorulabilir.
- Demo sırasında algoritma, geliştirdiğiniz kodun çeşitli kısımlarının ne amaçla yazıldığı ve geliştirme ortamı hakkında sorular sorulabilir.
- Kullandığınız herhangi bir satır kodu açıklamanız istenebilir.
- Proje grupları her öğretim kendi içinde olmak üzere en fazla iki kişiden oluşturulmalıdır.