

## Kullanıcı Bazlı Görev Takip Uygulaması (Task Manager)

Birden fazla kullanıcının sisteme giriş yapabildiği,  
her kullanıcının **yalnızca kendi görevlerini** görebildiği basit bir görev takip uygulaması  
geliştirmeniz beklenmektedir.

### 1. Kullanıcı Yapısı

#### Kullanıcı (User)

Sistemde birden fazla kullanıcı bulunur.

Örnek alanlar:

- id
- username
- password

Güvenlik beklentisi yoktur.

JWT, OAuth, Spring Security gibi ileri seviye çözümler **zorunlu değildir**.

---

### 2. Giriş (Login) Davranışı

- Kullanıcı username ve password ile giriş yapar
  - Başarılı giriş sonrası kullanıcı kendi görevlerini görür
  - Giriş yapan kullanıcının bilgisi frontend tarafında tutulabilir
  - Backend, gelen isteklerde hangi kullanıcının işlem yaptığını bilmelidir
- 

### 3. Görev (Task) Yapısı

Her görev bir kullanıcıya aittir.

Örnek alanlar:

- id
- title (zorunlu)
- description (opsiyonel)
- status (NEW, IN\_PROGRESS, DONE)
- createdDate

- user (görevin sahibi)
- 

## 4. Backend Gereksinimleri (Spring Boot)

### Kullanılabilecek Teknolojiler

- Spring Boot
  - Spring Data JPA
  - H2 veya PostgreSQL (verileri veri tabanı üzerinde tutmayı bekliyoruz)
  - Maven
- 

### API Beklentileri

#### Kullanıcı

- Kullanıcı giriş işlemi yapılabilmelidir

#### Görevler

- Giriş yapan kullanıcı **sadece kendi görevlerini** listeleyebilmelidir
  - Giriş yapan kullanıcı kendi adına görev ekleyebilmelidir
  - Kullanıcı sadece **kendi görevlerini** güncelleyebilmelidir
  - Kullanıcı sadece **kendi görevlerini** silebilmelidir
- 

### Backend Tarafında Beklenenler

- Controller / Service / Repository katman ayrimı
  - Basit validation kullanımı (ör. boş title kabul edilmemesi)
  - Anlamlı HTTP status code'lar
  - Temiz ve okunabilir kod
- 

## 5. Frontend Gereksinimleri (Angular + PrimeNG)

### Kullanılabilecek Teknolojiler

- Angular
  - PrimeNG
-

## **6. Frontend Ekranları**

### **6.1 Login Ekranı**

- Username ve password alanları
  - Başarılı giriş sonrası görev listesi ekranına yönlendirme
- 

### **6.2 Görev Listesi**

- Sadece giriş yapan kullanıcıya ait görevler gösterilir
  - Görevler tablo şeklinde listelenir
  - Görev silme işlemi yapılabilir
  - Status'a göre filtreleme yapılabilir
- 

### **6.3 Görev Ekle / Güncelle**

- Yeni görev eklenebilir
  - Mevcut görev güncellenebilir
  - Title alanı zorunludur
  - Status seçilebilir
  - Form validation uygulanmalıdır
- 

## **10. Frontend Tarafında Beklenenler**

- API çağrıları Angular Service üzerinden yapılmalıdır
  - Component ve Service ayrımı yapılmalıdır
  - Temel routing kullanılabilir
  - Kullanıcıya basit başarı / hata mesajları gösterilebilir
- 

## **11. Dokümantasyon (Zorunlu)**

Proje içerisinde bir **README.md** dosyası bulunmalıdır.

README dosyasında aşağıdaki başlıklar yer almalıdır:

- Proje nasıl çalıştırılır?
  - Kullanılan teknolojiler
  - Karşılaşılan zorluklar
  - Vakit olsaydı yapılabilecek geliştirmeler
- 

## **12. Bonus (Zorunlu Değildir)**

- Status'a göre renkli gösterim
- Sayfalama (pagination)
- Görev için son tarih (due date) eklenmesi
- Admin kullanıcı(bütün görevleri gören, değişiklik yapabilen, istediği kişiye iş atayabilen bir kullanıcı)