

# Django REST API Framework ile Proje Ödevi

---

## Giriş

Bu ödev, Django REST API framework kullanarak bir web uygulaması geliştirmeyi amaçlamaktadır. Proje kapsamında üç ana modül oluşturulacaktır: kullanıcı işlemleri, blog işlemleri ve blog öneri işlemleri. Kullanıcıların, JWT token kullanarak kimlik doğrulaması yapacakları bir login ekranı olacak ve giriş yaptıktan sonra yetkili oldukları API'lere istek gönderebileceklerdir.

## Gereksinimler

### 1. Login Ekranı:

- Kullanıcı adı ve şifre ile giriş yapılacaktır.
- Django'nun `authenticate` yapısı kullanılacaktır.
- Başarılı girişte JWT token dönecek.
- Başarısız girişte hata mesajı dönecek.

### 2. Modüller:

- Kullanıcı İşlemleri
- Blog İşlemleri
- Blog Öneri İşlemleri (Bonus)

### 3. Genel Kurallar:

- Tüm listeleme API'leri paginable olmalıdır.
- URL'ler kebab-case tipinde olmalıdır.
- API dokümantasyonu hazırlanmalıdır.
- Soft delete işlemleri kullanılmalıdır.

## Kullanıcı İşlemleri

### Gerekli API'ler

#### 1. Kullanıcı Listesi Getirme (GET)

URL: /api/get/kullanici-listesi

Açıklama: Sistemdeki tüm kullanıcıları listeleyecek.

## 2. Kullanıcı ID'ye Göre Kullanıcı Getirme (GET)

URL: /api/get/kullanici/{id}

Açıklama: Belirtilen ID'ye sahip kullanıcıyı getirecek.

## 3. Kullanıcı ID'ye Göre Kullanıcı Güncelleme (PUT)

URL: /api/update/kullanici/{id}

Açıklama: Belirtilen ID'ye sahip kullanıcıyı güncelleyecek.

## 4. Kullanıcı ID'ye Göre Kullanıcı Silme (Soft Delete) (DELETE)

URL: /api/delete/kullanici/{id}

Açıklama: Belirtilen ID'ye sahip kullanıcıyı soft delete yapacak.

## 5. Kullanıcı Oluşturma (POST)

URL: /api/create/kullanici

Açıklama: Gerekli Alanlar:

- Ad
- Soyad
- Şifre
- Şifre Tekrar
- Telefon Numarası (phone\_code ve phone olarak saklanmalıdır, örnek: +90 - 5452333333)

## 6. Kullanıcı Yetkileri Getirme (GET)

URL: /api/get/kullanici-yetkileri

Açıklama: Kullanıcı yetkilerini listeleyecek.

## 7. Kullanıcı ID'ye Göre Kullanıcı Yetkilendirme veya Yetki Kaldırma (PUT)

URL: /api/update/kullanici-yetki/{id}

Açıklama: Belirtilen ID'ye sahip kullanıcıya yetki ekleme veya kaldırma işlemi yapacak.

## Kullanıcı Yetkileri

- Blog ekleme yetkisi
- Blog silme yetkisi
- Diğer modüller için gerekli yetkiler

## Blog İşlemleri

### Gerekli API'ler

#### 1. Blogları Getirme (GET)

URL: /api/get/blog-listesi

Açıklama: Tüm blogları listeleyecek.

#### 2. Blog ID'ye Göre Blog Getirme (GET)

URL: /api/get/blog/{id}

Açıklama: Belirtilen ID'ye sahip blogu getirecek.

### 3. Blog ID'ye Göre Blog Silme (Soft Delete) (DELETE)

URL: /api/delete/blog/{id}

Açıklama: Belirtilen ID'ye sahip blogu soft delete yapacak.

### 4. Blog Kategorileri Getirme (GET)

URL: /api/get/blog-kategorileri

Açıklama: Blog kategorilerini veritabanından getirecek.

### 5. Blog Ekleme (POST)

URL: /api/create/blog

Açıklama: Gerekli Alanlar:

- Blog Kategorisi
- Blog Adı
- Blog Soyadı
- Blog Kısa Açıklaması
- Blog Uzun Açıklaması
- Blog Anahtar Kelimeler
- Kaydeden Kişi (veritabanına kayıt edilmeli)

## Blog Öneri İşlemleri (Bonus)

### Gerekli API'ler

#### 1. Blogları Getirme (GET)

URL: /api/get/blog-oneri-listesi

Açıklama: Tüm blog önerilerini listeleyecek.

#### 2. Blog ID'ye Göre Blog Getirme (GET)

URL: /api/get/blog-oneri/{id}

Açıklama: Belirtilen ID'ye sahip blog önerisini getirecek.

#### 3. Blog ID'ye Göre Blog Silme (Soft Delete) (DELETE)

URL: /api/delete/blog-oneri/{id}

Açıklama: Belirtilen ID'ye sahip blog önerisini soft delete yapacak.

#### 4. Blog Ekleme (POST)

URL: /api/create/blog-oneri

Açıklama: Gerekli Alanlar:

- Blog Kategorisi
- Blog Anahtar Kelimeler
- ChatGPT kullanılarak oluşturulan blog veritabanına ayrı bir şekilde kaydedilmeli.

## Blog Öneri Formu

- Blog Kategorisi: Teknoloji
- Blog Anahtar Kelimeler: Yazılım, React, Kod...

ChatGPT ile oluşturulan blog yazısı, mevcut blogların türüne uygun olarak detaylı şekilde hazırlanmalıdır. Örneğin, teknoloji alanında bir blog yazısı elektronik veya yazılım alanına ait olabilir. Yapay zeka kullanılarak oluşturulacak blog yazısının belirli kelimelerle ayırt edici özellikleri anlatılmalıdır.

## Çalışma Kuralları

- Blog listesi, blog öneri listesi ve kullanıcı listesi paginable yapıda olmalıdır.
  - Örnek paginable URL: `/api/get/blog-listesi?skip=1&take=10` (ilk 10 veriyi getirir).
  - Örnek paginable URL: `/api/get/blog-listesi?skip=2&take=10` (ikinci 10 veriyi getirir).

## Sonuç

Bu ödev, Django REST API framework kullanarak temel bir web uygulaması geliştirme ve bu süreçte JWT token ile kimlik doğrulama gibi modern web teknolojilerini kullanma becerilerinizi geliştirmeyi amaçlamaktadır. Herhangi bir sorunuz olursa, lütfen iletişime geçin. Başarılar!

## Son Teslim Tarihi

Ödevin son teslim tarihi 18 Ağustos 2024'tür.