

MucizeOI API Endpoint Dokümantasyonu (Frontend Ekibi İçin)

Bu belge, "MucizeOI" hayvan sahiplenme platformunun frontend geliştiricileri için backend API sözleşmesini tanımlar.

Genel Bilgiler

Tüm istekler, API Gateway servisi üzerinden tek bir ana giriş noktasından yapılmalıdır.

Ana (Base) URL (Lokal): `http://localhost:8080`

API Versiyonu: `v1`

Güvenlik ve Yetkilendirme (Authentication)

API, JWT (JSON Web Token) kullanarak korunmaktadır.

- Public Endpoint'ler:** (`Public`) olarak işaretlenen endpoint'ler (örn: ilan listeleme, giriş yapma) herhangi bir yetkilendirme başlığı gerektirmez.
- Protected Endpoint'ler:** (`Protected`) olarak işaretlenen endpoint'ler (örn: ilan oluşturma, talep gönderme) geçerli bir Access Token gerektirir.

Kimlik Doğrulama Akışı

- Kullanıcı `POST /api/v1/auth/login` endpoint'ine email ve password gönderir.
- Başarılı girişte, API bir `accessToken` ve `refreshToken` döner.
- Frontend, tüm (`Protected`) isteklere `accessToken` 'i Authorization başlığına eklemelidir:
`Header: Authorization: Bearer <SizinAccessTokenBurada>`
- `accessToken` süresi dolduğunda (örn: 401 Unauthorized hatası alındığında), frontend `POST /api/v1/auth/refresh` endpoint'ine `refreshToken` 'ı göndererek yeni bir `accessToken` (ve `refreshToken`) alır.
- Çıkış yapmak için `POST /api/v1/auth/logout` çağrılır; bu, sunucudaki `refreshToken` 'ı geçersiz kılar.

Auth Servisi (Kullanıcı & Yetkilendirme)

Kullanıcı hesapları, giriş/çıkış ve token yönetiminden sorumludur.

`POST /api/v1/auth/register`

Açıklama: Yeni bir kullanıcı hesabı oluşturur. **Erişim:** (`Public`)

Request Body (JSON):

```
{
  "firstName": "Ahmet",
  "lastName": "Çelik",
  "email": "ahmet@mail.com",
  "password": "GucluBirSifre123",
  "phoneNumber": "5551234567"
}
```

Success Response (201 Created): Oluşturulan User objesini döner.

```
{
  "userId": 1,
  "firstName": "Ahmet",
  "lastName": "Çelik",
  "email": "ahmet@mail.com",
  "phoneNumber": "5551234567",
  "roleName": "ROLE_USER"
}
```

POST /api/v1/auth/login

Açıklama: Kullanıcı girişi yapar ve token'ları döndürür. **Erişim:** (Public)

Request Body (JSON):

```
{
  "email": "ahmet@mail.com",
  "password": "GucluBirSifre123"
}
```

Success Response (200 OK):

```
{
  "accessToken": "eyJhbGciOi...",
  "refreshToken": "1.1c9ca2..."
}
```

POST /api/v1/auth/refresh

Açıklama: Süresi dolmuş accessToken 'i yenilemek için kullanılır. **Erişim:** (Public)

Request Body (JSON):

```
{
  "refreshToken": "1.1c9ca2..."
}
```

}

Success Response (200 OK):

```
{
  "accessToken": "yeniAccessToken",
  "refreshToken": "1.71fbb0..."
}
```

POST /api/v1/auth/logout

Açıklama: Güvenli çıkış yapar. Sunucudaki refreshToken 'ı geçersiz kılar. **Erişim:** (Protected)

Success Response (204 No Content): Response body boş döner.

GET /api/v1/auth/me

Açıklama: Giriş yapmış olan kullanıcının kendi profil bilgilerini döndürür. **Erişim:** (Protected)

Success Response (200 OK): User objesi döner.

```
{
  "userId": 1,
  "firstName": "Ahmet",
  "lastName": "Çelik",
  "email": "ahmet@mail.com",
  "phoneNumber": "5551234567",
  "roleName": "ROLE_USER"
}
```

Listing Servisi (İlan & Meta Veri)

İlanların, ilan meta verilerinin (şehir, tür vb.) ve resim yüklemelerinin yönetiminden sorumludur.

GET /api/v1/listings

Açıklama: "Mevcut" durumdaki ilanları filtreli ve sayfalı olarak listeler. **Erişim:** (Public)

Query Parametreleri (Opsiyonel):

- page : Sayfa numarası (örn: 0)
- size : Sayfa boyutu (örn: 10)
- cityId : Şehir ID'si
- animalTypeId : Hayvan türü ID'si
- animalBreedId : Hayvan cinsi ID'si

- `sort` : Sıralama (örn: `createdAt,desc`)

Success Response (200 OK): Sayfalanmış Listing objeleri listesi döner.

POST `/api/v1/listings`

Açıklama: Yeni bir sahiplendirme ilanı oluşturur. Resim yükleme işlemi de burada yapılır. **Erişim:** (Protected)

Request Body: `multipart/form-data` (JSON değil!)

- `image` : Resim dosyası (örn: `kedi.jpg`)
- `title` : (String) İlan başlığı
- `description` : (String) İlan açıklaması
- `animalTypeId` : (Number) Hayvan tür ID'si
- `animalBreedId` : (Number) Hayvan cins ID'si
- `cityId` : (Number) Şehir ID'si
- `age` : (Number) Hayvanın yaşı
- `gender` : (String) "Dişi" veya "Erkek"

Success Response (201 Created): Oluşturulan Listing objesini döner (içerisinde `imageUrl` alanı dolu olarak).

GET `/api/v1/listings/{id}`

Açıklama: Tek bir ilanın detaylı bilgilerini döndürür. **Erişim:** (Public)

Path Parametresi:

- `id` : İstenen ilanın `listingId` değeri.

Success Response (200 OK): Listing objesi döner.

PUT `/api/v1/listings/{id}`

Açıklama: İlan sahibinin kendi ilanını güncellemesini sağlar. **Erişim:** (Protected)

Path Parametresi:

- `id` : Güncellenecek ilanın `listingId` değeri.

Request Body (JSON): Güncellenmesi istenen alanlar.

```
{
  "title": "Yeni Başlık",
  "description": "Güncellenmiş açıklama...",
  "age": 3
}
```

Success Response (200 OK): Güncellenmiş Listing objesini döner.

```
DELETE /api/v1/listings/{id}
```

Açıklama: İlan sahibinin (veya Admin'in) ilanı silmesini sağlar. **Erişim:** (Protected)

Path Parametresi:

- `id` : Silinecek ilanın `listingId` değeri.

Success Response (204 No Content): Response body boş döner.

```
GET /api/v1/meta/cities
```

Açıklama: Filtreleme için kullanılabilir tüm şehirleri listeler. **Erişim:** (Public) **Success Response (200 OK):** City objeleri dizisi döner.

```
GET /api/v1/meta/animal-types
```

Açıklama: Filtreleme için kullanılabilir tüm hayvan türlerini listeler. **Erişim:** (Public) **Success Response (200 OK):** AnimalType objeleri dizisi döner.

```
GET /api/v1/meta/animal-types/{typeId}/breeds
```

Açıklama: Belirli bir hayvan türüne ait (örn: Kedi) tüm cinsleri (Tekir, Siyam vb.) listeler. **Erişim:** (Public)

Path Parametresi:

- `typeId` : Cinsleri listelenecek türün `typeId` değeri.

Success Response (200 OK): AnimalBreed objeleri dizisi döner.

Request Servisi (Sahiplenme Talepleri)

Kullanıcıların ilanlara yaptığı sahiplenme başvurularını yönetir.

```
POST /api/v1/requests
```

Açıklama: Bir ilana sahiplenme talebi gönderir. **Erişim:** (Protected)

Request Body (JSON):

```
{
  "listingId": 77,
  "requestMessage": "Bu kediye çok sevdim, ona iyi bakabilirim..."
}
```

Not: Talep yapan `userId`, `Authorization header`'ındaki `token`'dan otomatik olarak alınacaktır.

Success Response (201 Created): Oluşturulan AdoptionRequest objesini döner.

```
GET /api/v1/requests/my-requests
```

Açıklama: Giriş yapmış kullanıcının gönderdiği tüm taleplerin listesini döndürür. **Erişim:** (Protected)

Success Response (200 OK): AdoptionRequest objeleri dizisi döner.

```
GET /api/v1/requests/my-listings-requests
```

Açıklama: Giriş yapmış kullanıcının ilanlarına gelen tüm taleplerin listesini döndürür. **Erişim:**

(Protected) **Success Response (200 OK):** AdoptionRequest objeleri dizisi döner.

```
POST /api/v1/requests/{id}/approve
```

Açıklama: İlan sahibinin, ilanına gelen bir talebi onaylamasını sağlar. Bu işlem, ilgili listing 'in durumunu da "Sahiplendirildi" olarak günceller. **Erişim:** (Protected)

Path Parametresi:

- `id` : Onaylanacak talebin `requestId` değeri.

Success Response (200 OK):

```
{
  "message": "Talep onaylandı ve ilan 'Sahiplendirildi' olarak güncellendi."
}
```

```
POST /api/v1/requests/{id}/reject
```

Açıklama: İlan sahibinin, ilanına gelen bir talebi reddetmesini sağlar. **Erişim:** (Protected)

Path Parametresi:

- `id` : Reddedilecek talebin `requestId` değeri.

Success Response (200 OK):

```
{
  "message": "Talep reddedildi."
}
```

Data Modelleri (CamelCase Referans)

Endpoint'lerde (Obje) olarak belirtilen temel veri yapılarının referanslarıdır.

User

Kullanıcı bilgilerini temsil eder (Not: `password` gibi hassas veriler asla response'a dahil edilmez).

```
{
  "userId": 1,
```

```
"firstName": "Ahmet",
"lastName": "Çelik",
"email": "ahmet@mail.com",
"phoneNumber": "5551234567",
"roleName": "ROLE_USER",
"createdAt": "2025-11-04T10:30:00Z"
}
```

Listing

Bir sahiplendirme ilanını temsil eder.

```
{
  "listingId": 77,
  "userId": 1,
  "title": "Tekir Kedi 'Duman' Yuva Arıyor",
  "description": "Çok uysal bir kedi...",
  "imageUrl": "[https://mucizeol-images.fra1.digitaloceanspaces.com/kedi-123.jpg](https://mucizeol-images.fra1.digitaloceanspaces.com/kedi-123.jpg)",
  "animalTypeId": 1,
  "animalBreedId": 5,
  "cityId": 34,
  "age": 2,
  "gender": "Erkek",
  "status": "Mevcut",
  "createdAt": "2025-11-05T14:00:00Z",
  "updatedAt": "2025-11-05T14:00:00Z"
}
```

AdoptionRequest

Bir sahiplenme talebini temsil eder.

```
{
  "requestId": 123,
  "userId": 2,
  "listingId": 77,
  "status": "Beklemede",
  "requestMessage": "Bu kediye çok sevdim...",
  "createdAt": "2025-11-06T09:15:00Z"
}
```

City

```
{
  "cityId": 34,
  "cityName": "İstanbul"
}
```

```
}
```

AnimalType

```
{
  "typeId": 1,
  "typeName": "Kedi"
}
```

AnimalBreed

```
{
  "breedId": 5,
  "typeId": 1,
  "breedName": "Tekir"
}
```

Hata Formatı

Tüm 4xx (Client) ve 5xx (Server) hataları aşağıdaki standart JSON yapısını izler.

```
{
  "timestamp": "2025-11-17T17:45:12.345Z",
  "path": "/api/v1/auth/login",
  "status": 401,
  "code": "AUTH.INVALID_CREDENTIALS",
  "message": "Geçersiz bilgiler",
  "errors": [
    { "field": "email", "message": "Geçerli bir email giriniz" }
  ]
}
```