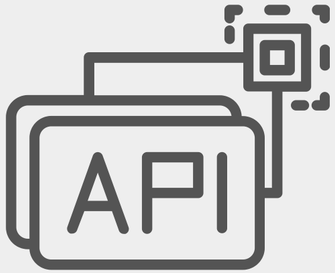
ToDo List API PROJESİ

ToDoListAPI, kullanıcıların yapılacaklar listesini yönetmelerine olanak tanıyan bir RESTful API'dir. Bu API, kullanıcıların görev ekleme, güncelleme, silme ve listeleme işlemlerini kolaylıkla yapabilmelerini sağlar.

Proje Amacı

Kullanıcıların günlük görevlerini yönetmelerine yardımcı olacak bir API. Görev **ekleme**, **güncelleme**, **silme** ve **listeleme** özelliklerini içerir.

Kullanılan Teknolojiler



- **Backend:** Java 17 ve Spring Boot kullanılarak geliştirilmiştir.
- **Veritabanı:** PostgreSQL ile veri depolama ve yönetimi sağlanmıştır.
- **API Dokümantasyonu**: Swagger (Springdoc OpenAPI) kullanılarak API dokümantasyonu oluşturulmuştur.
- **Diğer Araçlar:**
  - ModelMapper: DTO dönüşümleri için kullanılmıştır.

Proje Yapısı
Katmanlı Mimari

1. **CONTROLLER KATMANI:**
   - API endpoint'lerini tanımlar.
   - Gelen istekleri alır ve iş mantığını çalıştırmak için Service katmanına yönlendirir.
   - Kullanıcı dostu yanıtlar döndürür.
2. **SERVICE KATMANI:**
   - İş mantığını içerir.
   - Veritabanı işlemleri için Repository katmanıyla iletişim kurar.
   - DTO (Data Transfer Object) dönüşümleri ve diğer işlemler burada yapılır.
3. **REPOSITORY KATMANI:**
   - Veritabanıyla doğrudan iletişim kurar.
   - CRUD işlemleri için JPA/Hibernate yöntemlerini sağlar.

4. **ENTITY KATMANI:**
   - Veritabanı tablolarını temsil eden sınıfları içerir.

5. DTO KATMANI:
   - Entity'lerin direkt olarak dış dünyaya açılmasını engeller

6.EXCEPTION KATMANI
   - Proje genelindeki hata yönetimini sağlar.
   - Özel hata sınıfları ve global exception handler içerir.

7.CONFIG KATMANI
   - Swagger yapılandırması

API Özellikleri

- **GÖREV YÖNETIMI:**
  - Yeni görev ekleme
  - Mevcut görevi güncelleme
  - Görev silme
  - Tüm görevleri listeleme
  - ID'ye göre görev getirme
- **DOKÜMANTASYON**:
  - Swagger UI ile etkileşimli API dokümantasyonu sağlanmıştır.
- **HATA YÖNETIMI:**
  - Global exception handling ile özelleştirilmiş BaseException ve ErrorMessage sınıfları ilehata yöneltilmiştir.

API Endpoints Tablosu

HTTP Metodu	Endpoint	Açıklama	Girdi Parametreleri
GET	`/api/todos`	Tüm görevleri listeleme	Yok
GET	`/api/todos/{id}`	ID'ye göre görev getirme	`id` (Path)
POST	`/api/todos`	Yeni görev oluşturma	Gövde (JSON)
PUT	`/api/todos/{id}`	Mevcut bir görevi güncelleme	`id` (Path), Gövde
DELETE	`/api/todos/{id}`	Görev silme	`id` (Path)

Hata Yönetimleri

GET

Code	Details

400
Undocumented
	Error: response status is 400
	Response body
	{
	"status": 400,
	"exception": {
	"hostName": null,
	"path": "/api/todos/4777",
	"createTime": "2025-01-11T10:31:16.250+00:00",
	"message": "Kayıt Bulunamadı : Kayıt bulunamadı: ID 4777"
	}
	}

DELETE

Code	Details

400
Undocumented
	Error: response status is 400
	Response body
	{
	"status": 400,
	"exception": {
	"hostName": null,
	"path": "/api/todos/20854",
	"createTime": "2025-01-11T11:19:52.280+00:00",
	"message": "Kayıt Bulunamadı : Silinecek kayıt bulunamadı: ID 20854"
	}
	}

POST

Code	Details

400
Undocumented
	Error: response status is 400
	Response body
	{
	"status": 400,
	"exception": {
	"hostName": null,
	"path": "/api/todos",
	"createTime": "2025-01-11T10:19:21.909+00:00",
	"message": "Genel Bir Hata Oluştu : Aynı başlığa sahip bir Todo

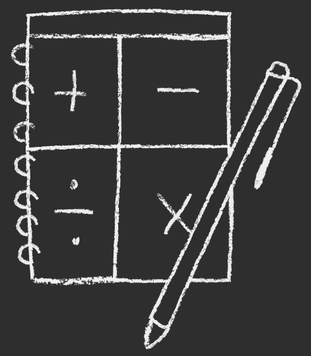
PUT

400
Undocumented

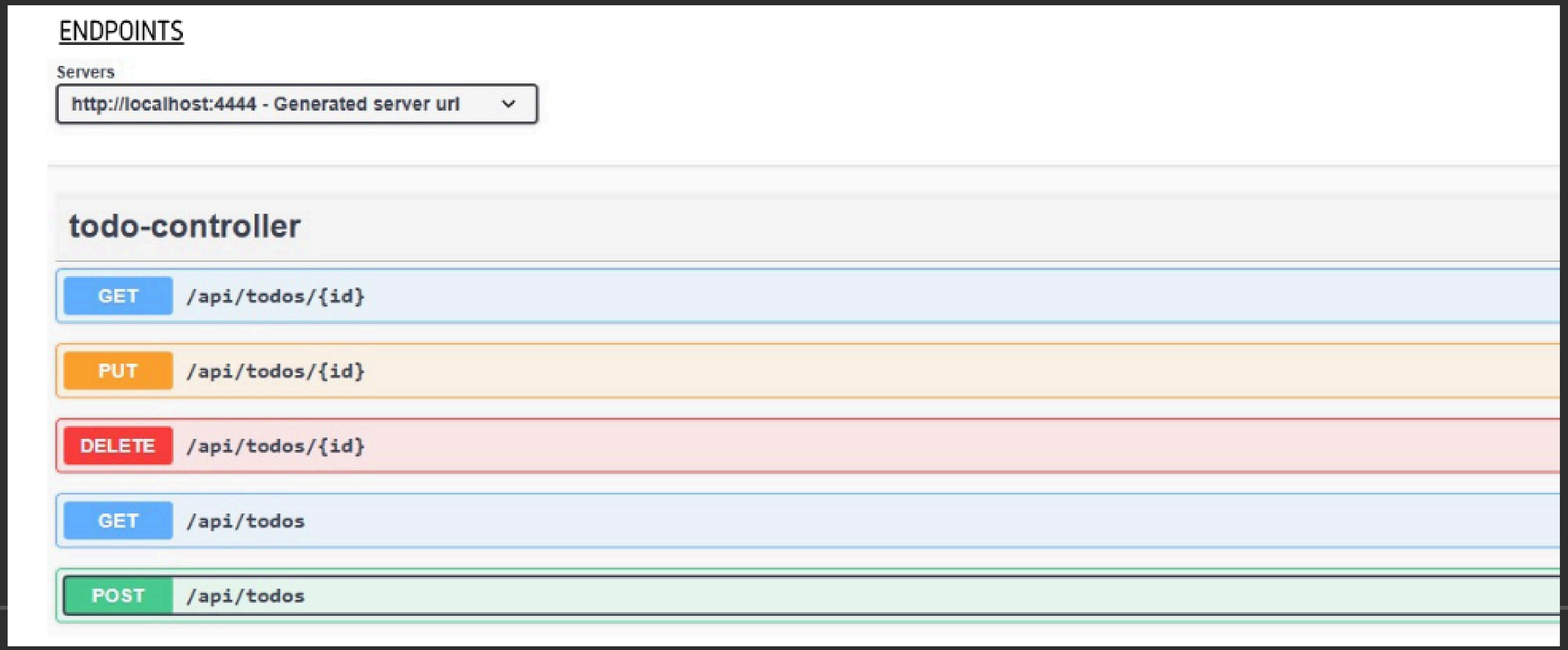
Error: response status is 400

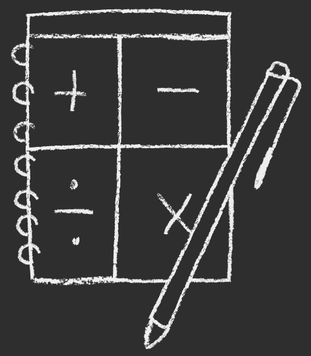
Response body

{
"status": 400,
"exception": {
"hostName": null,
"path": "/api/todos/788",
"createTime": "2025-01-11T10:28:25.036+00:00",
"message": "Kayıt Bulunamadı : Güncellenecek kayıt bulunamadı: ID 788"
}
}



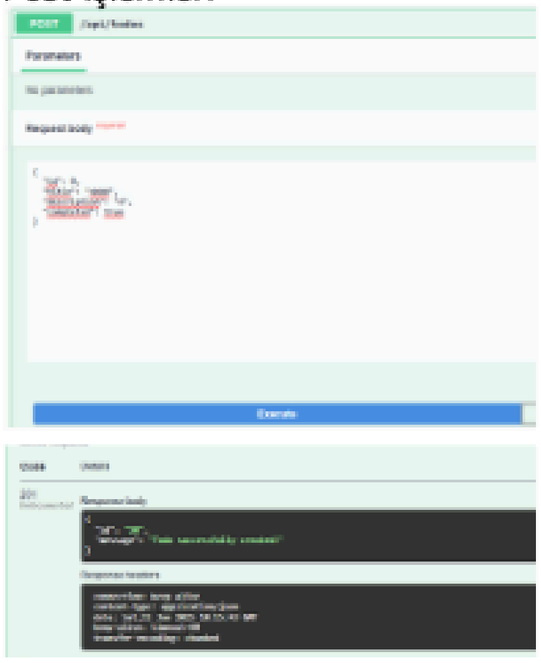
Swagger Ekran Görüntüleri



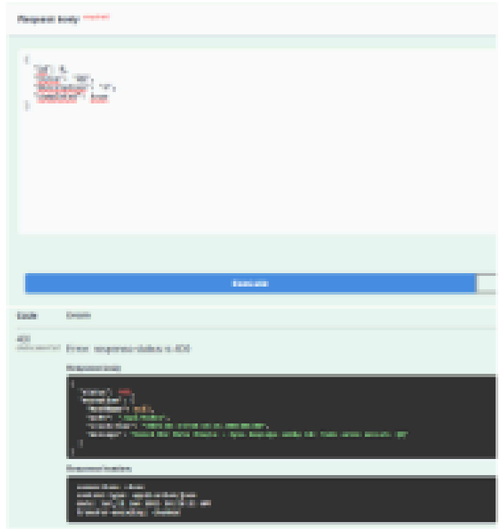


Swagger Ekran Görüntüleri

Post İşlemleri



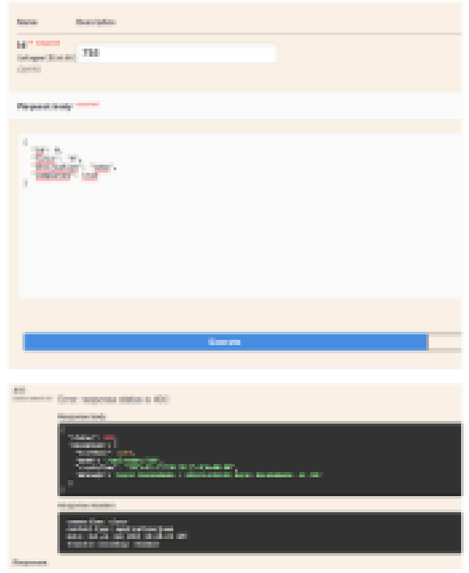
Başarılı



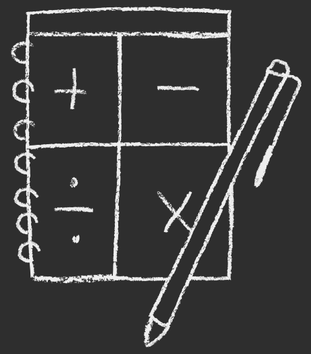
Başarısız

Put İşlemleri

Başarılı

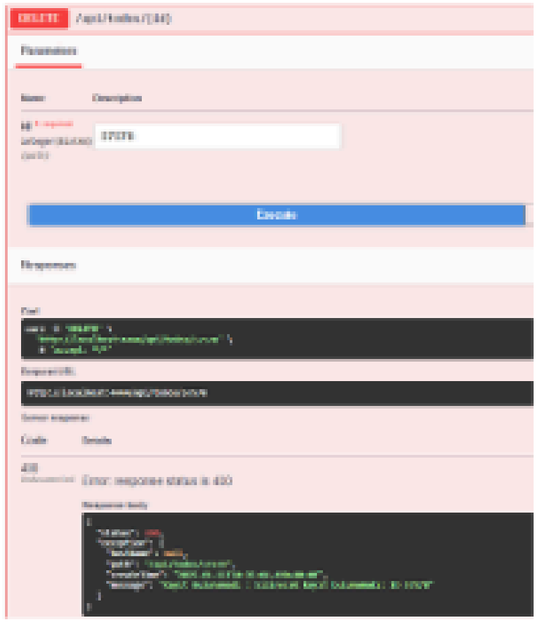


Başarısız

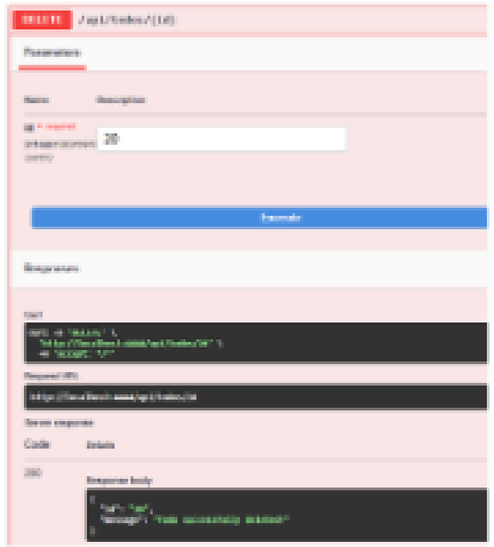


Swagger Ekran Görüntüleri

Delete İşlemleri

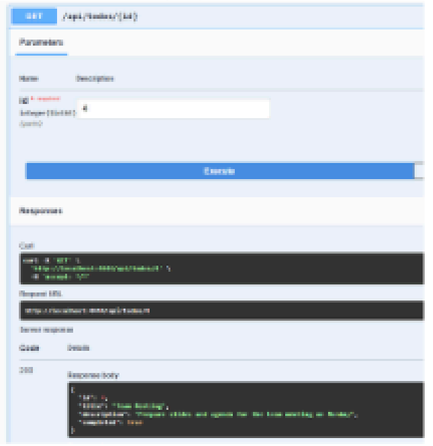


Başarılı

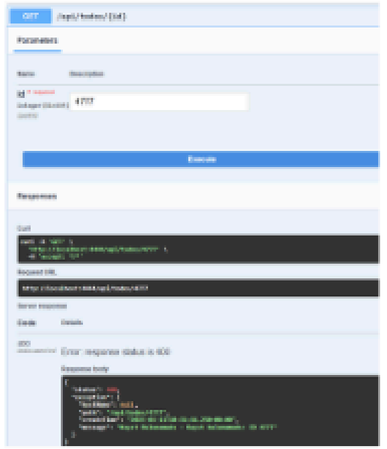


Başarısız

Get İşlemleri (/id)

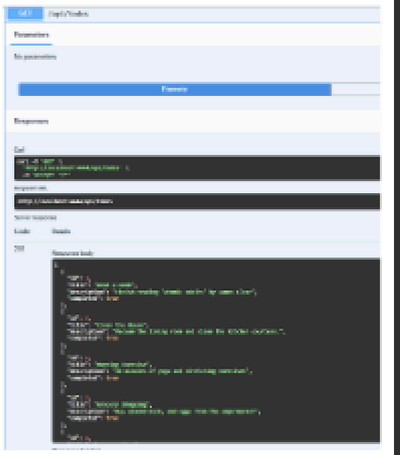


Başarılı



Başarısız

Get İşlemleri



teşekkürler.

- CEYLAN YILDIRIM