

# Kullanım Kılavuzu

## 1.) Giriş (Ön Bilgilendirme)

- Akıllı Görev Yönetimi Sistemi, bireylerin ve ekiplerin görevlerini kolayca planlamalarını, takip etmelerini ve verimliliklerini analiz etmelerini sağlayan bir araçtır.

## 2.) Projenin Amacı ve Kapsamı

- Görevlerin etkin şekilde planlanması, önceliklendirilmesi ve takibi için geliştirilmiştir.
- Otomatik e-posta hatırlatmaları ve performans raporları sunar.
- Tüm görev değişiklikleri loglanır ve raporlanır.

## 3.) Sistem Gereksinimleri

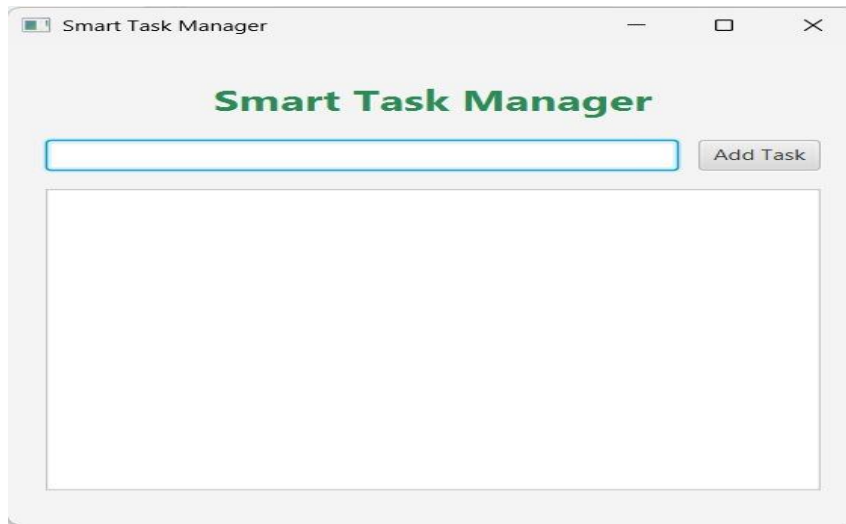
- Projeyi çalıştırmak için aşağıdaki yazılımların bilgisayarda kurulu olması gerekir:
  - Java JDK 11 veya üzeri (tavsiye edilen: JDK 17)
  - Apache Maven
  - Java destekleyen bir IDE

## 4.) Projeyi Çalıştırma Adımları

- **Depoyu Klonlayın:** git clone <https://github.com/Zeynep-2525/Akilli-Gorev-Sistemi.git>
- **Bağımlılıkları Yükleyin:** mvn clean install
- **Uygulamayı Başlatın:** mvn spring-boot:run
- **JavaFX Arayüzü:** javafx-maven-plugin ile ayrıca başlatılabilir.

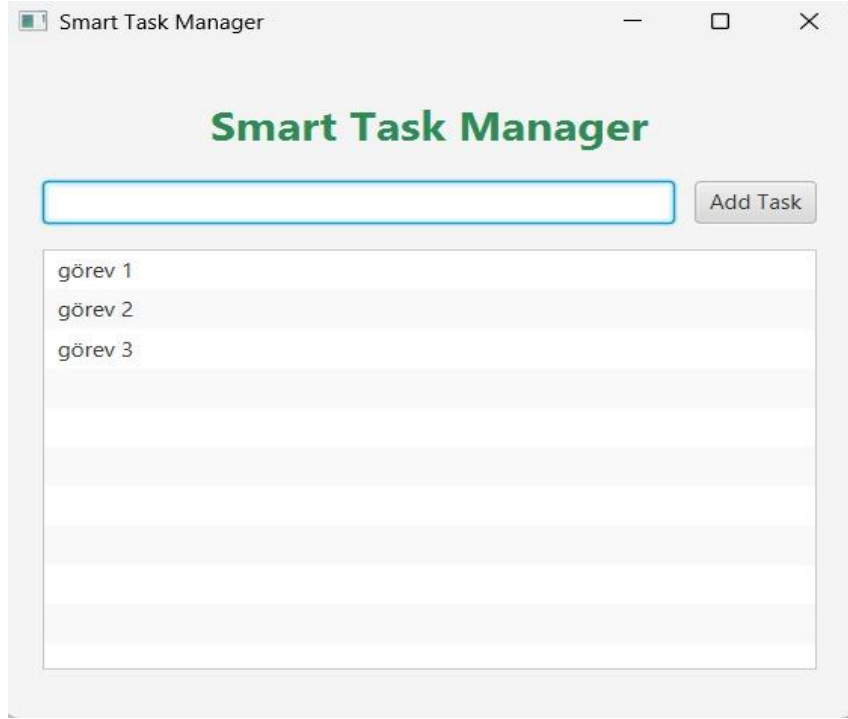
## 5.) Kullanıcı Arayüzü ve Temel İşlevler

### ➤ Yeni Görev Ekleme Ekranı



- "Add Task" butonuna tıklayın.
- Görev adını yazın.
- Enter tuşuna basarak veya "Kaydet" butonuna tıklayarak görevi ekleyin.

### ➤ Görev Eklendikten Sonra Ekran



- Tüm görevlerinizi bu ekranda liste halinde görebilirsiniz.
- Görevler "görev 1", "görev 2" şeklinde listelenir.

## 6.) API Kullanım Kılavuzu

### ➤ Görev Oluştur (Create Task)

- Yöntem: Post
- **URL:** /api/tasks
- **Header:** Content-Type: application/json
- Request Body:

```
{
  "title": "Yeni Görev",
  "description": "Görev açıklaması",
  "isCompleted": false,
  "userEmail": "kullanici@example.com",
  "priority": HIGH,
  "dueDate": "2025-05-20T00:00:00"
}
```

- Başarılı Yanıt:
  - 201 Created

```
{
  "id": 1,
  "title": "Ödev Yap",
  "description": "Matematik ödevini bitir",
  "isCompleted": false,
  "userEmail": "kullanici@example.com",
  "priority": HIGH,
  "dueDate": "2025-05-20T00:00:00"
}
```

## ➤ Tüm Görevleri Getir (Get All Tasks)

- **Yöntem:** GET
- **URL:** /api/tasks
- Başarılı Yanıt:
  - 200 OK

[

```
{
  "id": 1,
  "title": "Ödev Yap",
  "description": "Matematik ödevini bitir",
  "isCompleted": false,
  "priority": HIGH,
  "userEmail": "kullanici@example.com",
  "dueDate": "2025-05-20T00:00:00"
},
{
  "id": 2,
  "title": "Kitap Oku",
  "description": "Roman oku",
  "isCompleted": true,
  "priority": MEDIUM,
  "userEmail": "kullanici@example.com",
  "dueDate": "2025-05-15T00:00:00"
}
]
```

## ➤ ID ile Görev Getir (Get Task by ID)

- **Yöntem:** GET
- **URL:** /api/tasks/{id}
- **Örnek:** /api/tasks/1
- Başarılı Yanıt:

- 200 OK

```
{  
  "id": 1,  
  "title": "Ödev Yap",  
  "description": "Matematik ödevini bitir",  
  "isCompleted": false,  
  "priority": HIGH,  
  "userEmail": "kullanici@example.com",  
  "dueDate": "2025-05-20T00:00:00"  
}
```

### ➤ Görev Güncelle (Update Task)

- **Yöntem:** PUT
- **URL:** /api/tasks/{id}
- **Header:**
  - Content-Type: application/json
- **Request Body:**

```
{  
  "title": "Güncellenmiş Görev",  
  "description": "Görev açıklaması güncellendi",  
  "isCompleted": true,  
  "priority": MEDIUM,  
  "userEmail": "kullanici@example.com",  
  "dueDate": "2025-06-01T00:00:00"
```

```
}
```

- Başarılı Yanıt:
  - 200 OK

```
{  
  "id": 1,  
  "title": "Güncellenmiş Görev",  
  "description": "Görev açıklaması güncellendi",  
  "isCompleted": true,  
  "priority": MEDIUM,  
  "userEmail": "kullanici@example.com",  
  "dueDate": "2025-06-01T00:00:00"  
}
```

## ➤ Görev Sil (Delete Task)

- **Yöntem:** DELETE
- **URL:** /api/tasks/{id}
- **Örnek:** /api/tasks/1
- Başarılı Yanıt:
  - 204 No Content