

Node.js Proje Kampı - “TeamTask”

RestApi

Amaç

Bu proje çalışmasının amacı; TypeScript tabanlı Node.js ortamında **katmanlı mimari**, **REST API geliştirme**, **JWT tabanlı kimlik doğrulama**, **rol tabanlı yetkilendirme**, **Swagger dokümantasyonu**, ve **MongoDB veri modelleme** konularını gerçek bir takım projesi üzerinde uygulamalarıdır.

Her ekip, aynı projeyi bağımsız olarak geliştirecek, proje sonunda her ekip ödevler klasörü altında yapmış olduğu projeyi paylaşacaktır. Ekip sorumlusu proje sonunda ekip performans raporu sunacaktır.

Tarih: 09:00 – 24:00 (Toplam 15 Saat)

Proje Adı: TeamTask – Rol Tabanlı Proje Yönetim Sistemi – Rest Servisi

Teknolojiler: Node.js, Express, TypeScript, MongoDB, Mongoose, JWT, Swagger

Proje Tanımı

TeamTask, bir organizasyonun proje ve görevlerini yönetebilmesini sağlayan bir RESTful API uygulamasıdır. Sistemde 3 rol bulunur:

- **Admin:** Kullanıcı yönetimi yapabilir, rolleri atar.
- **Project Manager:** Proje oluşturabilir, proje görevlerini yönetir.
- **Developer:** Kendi görevlerini görüntüler ve günceller.

Proje Yapısı (Katmanlı Mimari)

```
src/
├── config/
│   └── db.ts
├── controllers/
│   ├── auth.controller.ts
│   ├── user.controller.ts
│   ├── project.controller.ts
│   └── task.controller.ts
├── middlewares/
│   ├── auth.middleware.ts
│   └── role.middleware.ts
├── models/
│   ├── user.model.ts
│   ├── project.model.ts
│   └── task.model.ts
├── routes/
│   ├── auth.routes.ts
│   ├── user.routes.ts
│   ├── project.routes.ts
│   └── task.routes.ts
├── utils/
│   └── jwt.ts
└── app.ts
```

Ana Modüller

Modül	Açıklama
Authentication (Auth)	JWT tabanlı login/register işlemleri
User Management	Kullanıcı CRUD + rol atama (Admin yetkisi)
Project Management	Proje CRUD işlemleri (Project Manager & Admin)
Task Management	Görev CRUD, statü değişimi, proje ilişkilendirme

Gereksinimler

1. Authentication

- POST /auth/register – Kullanıcı kaydı (Admin yetkisi)
- POST /auth/login – JWT token üretimi
- GET /auth/profile – Token doğrulama

2. User Management

- GET /users – Tüm kullanıcılar
- PATCH /users/:id/role – Rol güncelleme

3. Project Management

- POST /projects – Yeni proje oluşturma
- GET /projects – Projeleri listeleme
- PATCH /projects/:id – Güncelleme
- DELETE /projects/:id – Silme

4. Task Management

- POST /projects/:id/tasks – Görev ekleme
- GET /projects/:id/tasks – Görevleri listeleme
- PATCH /tasks/:id/status – Görev statü güncelleme

1. Grup (4 Kişi)

Grup Sorumlusu: Ufuk

Üyeler: Haktan, İsmail, Mustafa

Üye	Sorumlu Modül	Açıklama
Ufuk (Sorumlu)	Proje koordinasyonu, Auth modülü	Login/Register, JWT üretimi ve Swagger entegrasyonu
Haktan	User Management	Kullanıcı CRUD ve rol güncelleme işlemleri
İsmail	Project Management	Proje CRUD, Project Manager izinleri
Mustafa	Task Management	Görev CRUD, status değişimi, proje ilişkisi

2. Grup (3 Kişi)

Grup Sorumlusu: Ahmet

Üyeler: Mümine, Tuğba, Serhat

Üye	Sorumlu Modül	Açıklama
Ahmet (Sorumlu)	Auth & Proje Setup	JWT auth, proje başlatma, GitHub organizasyonu
Mümine	User Management	Rol tabanlı kullanıcı işlemleri
Tuğba	Project Management	Proje CRUD ve Controller yapısı
Serhat	Task Management	Görev CRUD, status güncelleme işlemleri

3. Grup (3 Kişi)

Grup Sorumlusu: Çağlar

Üyeler: Furkan, Ömer

Üye	Sorumlu Modül	Açıklama
Çağlar (Sorumlu)	Auth ve Swagger	JWT yapısı ve API dokümantasyonu
Furkan	Project Management	Proje CRUD işlemleri
Ömer	Task Management	Görev CRUD ve status yönetimi

Teslim Şartları

1. Her grup **ödevler klasörü altında örn: grup_1** klasörü ile projesini yükleyecektir.
 - Repo adı: TeamTask-Group_1
2. Her repo'da mutlaka bir README.md bulunmalıdır.
3. README dosyasında:
 - Proje açıklaması
 - Çalıştırma talimatları
 - Kullanılan teknolojiler
 - Swagger linki
yer almalıdır.
4. **Grup sorumlusu**, gün sonunda aşağıdaki bilgileri içeren bir raporu göndermelidir:
 - Ekip üyelerinin isimleri
 - Her üyenin geliştirdiği modül(ler)
 - Teknik katkı düzeyi
 - Ekip içi iletişim ve performans değerlendirmesi
 - Yaşanan sorunlar ve çözüm önerileri