Project: Multi-Role Project Management System

Goal-

Build a robust REST API for a Project Management System where companies can manage users, projects, and tasks, with strict role-based access and multi-tenancy.

Tech Stack-

Node.js with Express.js

MongoDB (Mongoose ODM)

JWT (with refresh token support)

dotenv for environment config

Postman or Swagger for API documentation

Entities & Relationships

1. Company-name, domain

2. User-name, email, password (hashed), role (Admin | Manager | Member)

belongs to one company

3. Project-name, description, createdBy (User), companyId

4. Task-title, description, status (To Do | In Progress | Done)

assignedTo (User), projectId

Functional Requirements-

Multi-tenancy: Users can only access data from their own company.

Authentication: JWT-based login + refresh token.

Authorization:

a. Admin: Full access to all company data.

b. Manager: Can manage projects & tasks, but not users.

c. Member: Can view and update assigned tasks only.

CRUD Operations:

Companies: register & setup

Users: create (admin only), list (by company), login, refresh tokens

Projects: CRUD by Admin & Manager

Tasks: CRUD by Admin & Manager; Members can update their assigned tasks

Pagination on listing endpoints.

Search/Filter: Tasks by status and assignee.

Validation: Use Joi or express-validator.

Additional Requirements

Rate Limiting (per IP using express-rate-limit)

Centralized Error Handling middleware

Async/Await + Promises – avoid callbacks

Modular folder structure (routes, controllers, services, models)

Basic unit tests using Jest or Mocha for at least 2 modules (e.g., Auth and Project APIs)

Bonus-

Dockerize the app with docker-compose for MongoDB

Implement audit logging (e.g., user X updated task Y at time Z)

CI/CD pipeline setup with GitHub Actions or similar

Expected Deliverables

GitHub repo with:

README with setup instructions and API usage

Postman collection or Swagger UI link

.env.example

Code should be clean, readable, and modular

SUBMIT BY 10:00AM (12:02:2025)

SEND THE LINK ON MAIL.