

Assignment: Full-Stack Task Management App with Admin Dashboard (MERN Stack)

Objective

Develop a **Task Management Web Application** using the **MERN stack (MongoDB, Express.js, React.js, Node.js)**. The application should support **role-based functionality** where:

- **Admin** can create, assign, edit, and delete tasks.
- **Users** can create and manage their own tasks.
- Users **cannot edit or delete admin-assigned tasks** but can update their status.
- Implement a **Start/Pause timer** for tasks (only one active task per user at a time).

Requirements

Tech Stack

- **Frontend:** React.js (Tailwind)
- **Backend:** Node.js + Express.js
- **Database:** MongoDB (with Mongoose)
- **Authentication:** JWT
- **State Management (optional):** Redux or React Context

Roles & Permissions

- **Admin:**
 - Create, assign, edit, and delete tasks.
 - View a dashboard with task/user statistics.
 - Can assign tasks to any user.
- **User:**
 - View assigned and self-created tasks.
 - Can create and edit/delete **their own tasks only**.
 - Can **update task status** (To Do, In Progress, Completed).
 - Can **Start/Pause a task timer** – only one task can be active at a time.

Functional Requirements

Authentication

- JWT-based authentication.
- Role-based access control (Admin/User).

Task Management

Users can:

- View tasks assigned to them or created by them.
- Create and edit/delete their own tasks.
- Update task status (in progress, completed).
- Start/Pause a task timer — only one task can be active at any time.

Admin can:

- View, create, edit, and delete all tasks.
- Assign tasks to any user.
- View dashboard metrics:
 - Total tasks (admin + users)
 - Completed tasks
 - Total users

Task Timer Logic

- When a user clicks "Start" on a task:
 - If another task is active, it should be paused.
 - Track time using `startedAt`, `pausedAt`, and `totalTimeSpent`.
- When "Pause" is clicked:
 - Save the time difference to `totalTimeSpent`.

Additional Features

- Push notifications for:
 - Task assigned to a user
 - Task completed by user (admin notified)
- Filters (by status: completed, in-progress, overdue)
- Pagination for task lists
- Responsive UI/UX with clean layout
- Testing: Include unit or integration tests

Deliverables

- Full source code on GitHub (public or private with access)
- README file with:
 - Setup instructions
 - API endpoints
 - Admin login (optional seed data)
- Postman collection (if API-first)
- Bonus: Link to a deployed version (Render, Vercel, Netlify)

Timeline

- **Estimated Time:** 3–5 days
- Submit when ready or as per discussed deadline

Attached the Figma Link For the UI/UX

<https://www.figma.com/proto/LXbtuHi5B2KTLymdxnNbhd/Test-Project---Laravel?page-id=0%3A1&node-id=2003-899&node-type=canvas&viewport=1705%2C639%2C0.23&t=GITUNTRkE4JHsgh1-1&scaling=min-zoom&content-scaling=fixed>