

Eisenhower Matrix Task Management Web App – Complete Project Guide

1. Understanding the Project

Goal: A web app that helps users organize their tasks using the Eisenhower Matrix — dividing tasks into 4 categories:

- Urgent + Important
- Not Urgent + Important
- Urgent + Not Important
- Not Urgent + Not Important

Core Features (MVP):

- User login/signup
- Add, edit, delete, and view tasks
- Categorize tasks automatically (based on urgency + importance)
- Display tasks in the 4 quadrants visually
- Option to drag & drop between quadrants (optional advanced step)

2. Suggested Tech Stack

- Frontend: React.js – Popular, component-based, easy to learn
- Styling: Tailwind CSS – Clean, modern UI quickly
- Backend: Node.js + Express.js – Simple for REST APIs
- Database: MongoDB (with Mongoose) – Flexible, beginner-friendly
- Authentication: JWT – Easy and secure login system
- Version Control: Git + GitHub
- Deployment: Frontend → Vercel/Netlify; Backend → Render/Railway/Cyclic

3. Step-by-Step Development Plan

- 1 Phase 1 – Setup: Install Node.js, Git, MongoDB; create /client and /server folders.
- 2 Phase 2 – Backend: Set up Express server, connect MongoDB, implement CRUD and JWT authentication.
- 3 Phase 3 – Frontend: Create React app, connect API with Axios, build Eisenhower Matrix UI using Tailwind.
- 4 Phase 4 – Polish & Testing: Add drag & drop, filters, validation, test with Postman.
- 5 Phase 5 – Deployment: Push to GitHub, deploy frontend and backend, use MongoDB Atlas for cloud DB.

4. Optional Advanced Features

- Add analytics (task completion stats).
- Integrate notifications or reminders.
- Dark mode toggle.
- Collaborative/shared task lists.
- Build a mobile app later using React Native.

5. Folder Structure Overview

```
project/ ■■■ client/ # React Frontend ■ ■■■ src/ ■ ■ ■■■ components/ ■ ■ ■■■  
pages/ ■ ■ ■■■ context/ ■ ■ ■■■ App.js ■■■ server/ # Node Backend ■■■ models/  
■■■ routes/ ■■■ controllers/ ■■■ config/ ■■■ server.js
```