Task Management Web Application

Abraham Gutierrez, Oluwatosin Omiteru, Simran Mogadala

Table of Contents

Problem Statement

These days managing tasks can be difficult for many people. People lose track of time, miss due dates for important tasks, and even miss out on important meetings. We expect that by individuals using this application, users will complete important tasks successfully and find it easy to use this application.

This task management application will help individuals organize their tasks. A person can create and manage their tasks through this application. Our goal is to make sure individuals stay organized during college and while working as well.

Solution Statement

This application uses an API, Frontend, and Backend which all works together so that the user can access the Task Management Application through the frontend.

The frontend is where the user can register or login to access their tasks. In the backend, the user's login information would be saved into the database.

Once a user is able to login, the user can

- Sort out tasks
- Delete tasks
- Write out new tasks
- Update tasks

Biggest Challenge

- Application starts at home page sometimes, instead of login screen
  - Needs to be reloaded
- Fetching Tasks
  - Deployment starts → Shows “Failed to fetch tasks”

Biggest Highlights

- Updating Tasks
  - Title
  - Description
  - Due Date
- Connection between MongoDB and API
  - Included CRUD operations

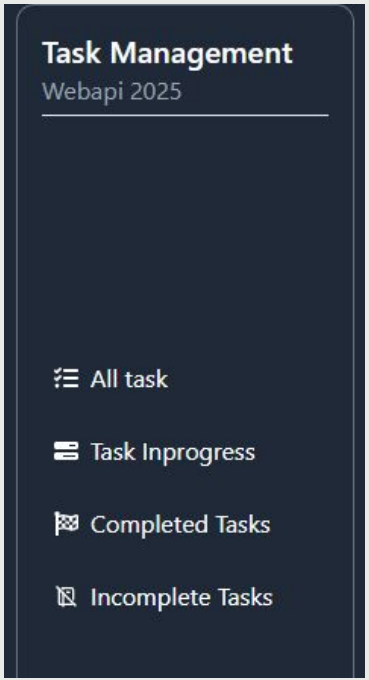
How it is Built

- Frontend
  - CSS files (App.css)
  - Render (App.test.js)
- Backend
  - SignUp and SignIn routes
  - JWT
  - User information (`username, email, password, createdate`)
  - Task information (`duedate, title, desc`)
- Running Application:
  - **`npm install → npm start`**
    - **`cd backend`**
    - **`cd frontend`**
      - Start development server through **localhost**

Technologies Used

User-Interface	React, React-Router, Tailwind CSS
API	Node.js, Javascript, JSON,
Database	MongoDB
Development Tools	VSCode, React-App, NPM

Task Management Tabs

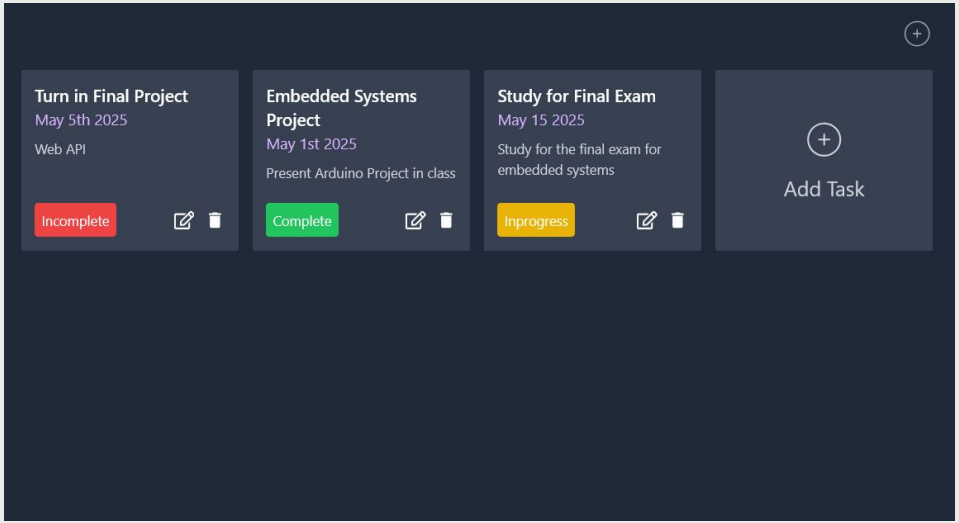


- Users can be able to see the **status** of their tasks
- The **sidebar** can be easily organized by showing tasks which are:
  - Completed
  - Incomplete
  - In Progress
  - Altogether

User SignUp and SignIn

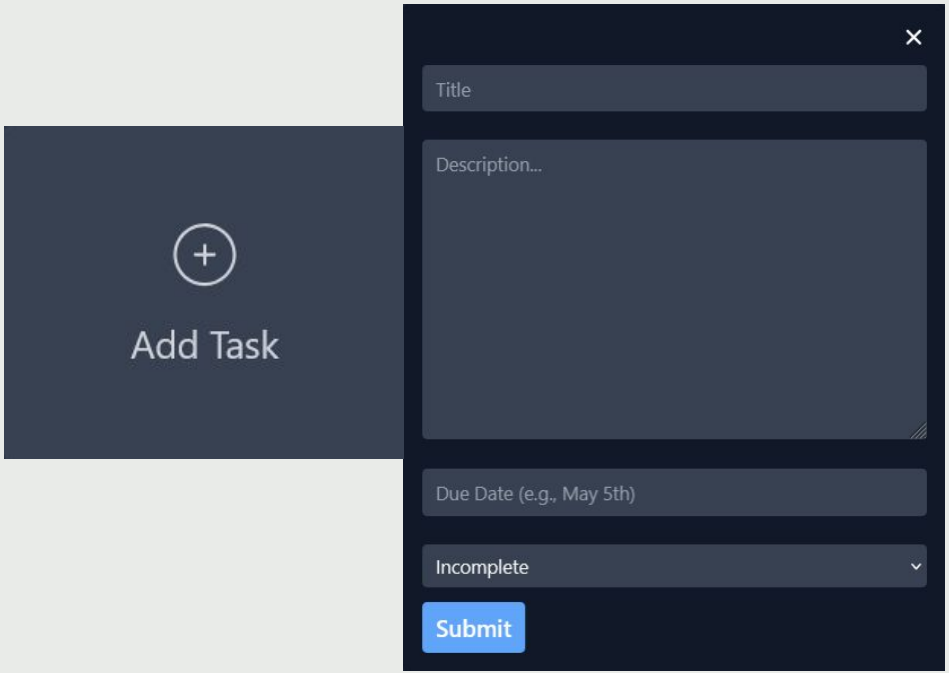
- Once a user enters login credentials, they are able to access their tasks
- SignUp
  - If someone who is new to the application will be able to create an account by:
    - Adding username
    - Enter email
    - Create password

Dashboard



- After logging in, the user is able to see all of their tasks in the dashboard.
- As mentioned before, they are able to organize tasks based on their status
- The user can also delete their tasks, update, and change their status
- The user can make new tasks from the dashboard

Create Tasks



- When a user creates a task, they press the the plus button
- This opens up a menu, where a the user can input a title, description, date, status, before submitting.
- Used when updating a task

DEMO!!

Future Development

- Important Tasks vs. Low Priority Tasks
  - People can mark which tasks are **time sensitive**
    - Needs to be completed immediately
    - Tasks that can be done later
- Visuals on Weekly and Monthly Progress
  - Show data or graphs on how a user is keeping up with tasks
- Flag missed tasks
  - If a task is missed, there will be a section for a user to notice
  - They can go back and complete the work/assignment
- Multi-Platform Implementation

Link to Repos

Github	https://github.com/ag0954/TaskManagement_APP.git