Project Description:

Create a simple **Task Management System** where users can:

1. View a list of tasks.

2. Add a new task.

3. Mark a task as completed.

4. Delete a task.

Requirements:

1. **Backend**:

    o   Use **Node.js** with **Express**.

    o   Use a database (SQLite, MongoDB, or any other) to store tasks.

    o   Implement the following API endpoints:

        ▪   GET /tasks: Retrieve all tasks.

        ▪   POST /tasks: Add a new task.

        ▪   PUT /tasks/:id: Update a task (e.g., mark it as completed).

        ▪   DELETE /tasks/:id: Delete a task.

    o   Tasks should have the following fields:

        ▪   id (unique identifier)

        ▪   title (string)

        ▪   description (string, optional)

        ▪   completed (boolean, default: false)

        ▪   created_at (timestamp)

2. **Frontend**:

    o   Use **React**.

    o   Create a user-friendly interface to interact with the backend API.

    o   Implement the following features:

        ▪   Display a list of tasks.

        ▪   A form to add a new task.

        ▪   Buttons to mark tasks as completed and delete tasks.

    o   Use any CSS framework (e.g., Bootstrap, Material-UI) or plain CSS for styling.

3. **Bonus Points**:
   o Use **TypeScript**.
   o Add validation (e.g., ensure the title is not empty when adding a task).
   o Implement sorting or filtering options (e.g., show completed tasks separately).
   o Deploy the project on a platform like Vercel, Netlify, or Heroku.

Submission Guidelines:

1. Push your code to a **public GitHub repository**.
2. Include a README.md file with the following:
   o Project description.
   o Steps to run the project locally.
   o Any assumptions or decisions made during development.
3. Share the GitHub repository link with us.

Evaluation Criteria:

1. **Functionality**:
   o Does the project meet the requirements?
   o Are all endpoints working correctly?
2. **Code Quality**:
   o Is the code well-organized and readable?
   o Are best practices followed?
3. **UI/UX**:
   o Is the frontend intuitive and visually appealing?
4. **Extra Features**:
   o Bonus points for any additional features implemented.

Timeframe:

You have **2-3 days** to complete this task. Please plan your time accordingly.

We look forward to seeing your work. Good luck!