**Project Name:** “Task Manager”

**Description:**

This to-do list application is designed to simplify task management for students, faculty, and professionals. Whether you’re organizing academic tasks, personal projects, or daily activities, this app helps you stay organized, prioritize effectively, and manage deadlines with ease.

**Features**:

1. Task Management:

Create, edit, and delete tasks.

Set due dates and reminders for each task.

Organize tasks into different categories (e.g., work, personal, study).

2. Priority Levels:

Assign priority levels to each task (e.g., high, medium, low).

Use color-coded indicators to quickly identify the most important tasks.

3. Reminders and Notifications:

Set custom reminders for upcoming tasks.Receive notifications for tasks that are due soon or have expired.

4. Progress Tracking:

Mark tasks as completed to track your productivity.

View progress reports or summaries of completed vs. pending tasks.

5. Calendar Integration:

Sync tasks with your calendar to see an overview of all your deadlines and upcoming events.

Get a visual timeline of tasks spread across the days or weeks.

**Console-Based Code:**

import React, { useState, useEffect } from 'react';

import axios from 'axios';

function App() {

const [tasks, setTasks] = useState([]);

const [newTask, setNewTask] = useState("");

// Fetch tasks from the backend

useEffect(() => {

axios.get('http://localhost:5000/tasks')

.then(response => setTasks(response.data))

.catch(error => console.error(error));

}, []);

// Add a new task

const addTask = () => {

axios.post('http://localhost:5000/tasks', { task: newTask })

.then(response => {

setTasks([...tasks, response.data]);

setNewTask("");

})

.catch(error => console.error(error));

};

// Delete a task

const deleteTask = (id) => {

axios.delete(http://localhost:5000/tasks/${id})

.then(() => setTasks(tasks.filter(task => task.\_id !== id)))

.catch(error => console.error(error));

};

return (

<div>

<h1>To-Do List</h1>

<input

type="text"

value={newTask}

onChange={(e) => setNewTask(e.target.value)}

placeholder="New task"

/>

<button onClick={addTask}>Add Task</button>

<ul>

{tasks.map(task => (

<li key={task.\_id}>

{task.task}

<button onClick={() => deleteTask(task.\_id)}>Delete</button>

</li>

))}

</ul>

</div>

);

}

**OUTPUT:**

