```
npx create-react-app task-manager
cd task-manager
npm install express mongoose cors
npm install --save-dev concurrently nodemon
npm install tailwindcss autoprefixer postcss
npx tailwindcss init -p
{
  "name": "task-manager",
  "version": "0.1.0",
  "private": true,
  "dependencies": {
    "cors": "^2.8.5",
    "express": "^4.18.2",
    "mongoose": "^8.0.0",
    "react": "^18.2.0",
    "react-dom": "^18.2.0",
    "react-scripts": "5.0.1",
    "web-vitals": "^2.1.4"
  },
  "scripts": {
    "start": "react-scripts start",
    "build": "react-scripts build",
    "test": "react-scripts test",
```

```
    "eject": "react-scripts eject",
    "server": "nodemon server.js",
    "dev": "concurrently \"npm run server\" \"npm
start\""
  },
    "devDependencies": {
    "concurrently": "^8.2.1",
    "nodemon": "^3.0.1",
      "autoprefixer": "^10.4.16",
      "postcss": "^8.4.31",
      "tailwindcss": "^3.3.5"
  },
  "eslintConfig": {
    "extends": [
      "react-app",
      "react-app/jest"
    ]
  },
  "browserslist": {
    "production": [
      ">0.2%",
      "not dead",
```

```
      "not op_mini all"
    ],
    "development": [
      "last 1 chrome version",
      "last 1 firefox version",
      "last 1 safari version"
    ]
  }
}
/** @type {import('tailwindcss').Config} */
module.exports = {
  content: [
    "./src/**/*.{js,jsx,ts,tsx}",
  ],
  theme: {
    extend: {},
  },
  plugins: [],
}
@tailwind base;
@tailwind components;
@tailwind utilities;
const express = require('express');
```

```
const mongoose = require('mongoose');
const cors = require('cors');
const app = express();
const PORT = process.env.PORT || 5000;

app.use(cors());
app.use(express.json());

mongoose.connect('mongodb://127.0.0.1:
27017/taskmanager', { // Replace with your
MongoDB URI
    useNewUrlParser: true,
    useUnifiedTopology: true
}).then(() => console.log('Connected to
MongoDB'))
.catch(err => console.error('Could not connect
to MongoDB', err));

const taskSchema = new mongoose.Schema({
    title: String,
    description: String,
    priority: String,
    category: String
```

```
});

const Task = mongoose.model('Task',
taskSchema);

app.get('/tasks', async (req, res) => {
  try {
    const tasks = await Task.find();
    res.json(tasks);
  } catch (err) {
    res.status(500).json({ message:
err.message });
  }
});

app.post('/tasks', async (req, res) => {
  const task = new Task(req.body);
  try {
    const newTask = await task.save();
    res.status(201).json(newTask);
  } catch (err) {
    res.status(400).json({ message:
err.message });
```

```
  }
});

// Implement PUT and DELETE routes similarly

app.listen(PORT, () => console.log(`Server
started on port ${PORT}`));
import React, { useState, useEffect } from 'react';

function App() {
 const [tasks, setTasks] = useState([]);

 useEffect(() => {
  fetch('http://localhost:5000/tasks')
   .then(res => res.json())
   .then(data => setTasks(data));
 }, []);

 return (
  <div className="App">
   {/* Display Tasks */}
   {tasks.map(task => (
    <div key={task._id}>
```

```
      .then(res => res.json())
      .then(data => setTasks(data));
  }, []);

  return (
    <div className="App">
      {/* Display Tasks */}
      {tasks.map(task => (
        <div key={task._id}>
          {/* Display task details */}
        </div>
      ))}
    </div>
  );
}

export default App;
```