

PedalStart - Task Management Application

Project Title: PedalStart - Task Management Application

Description

PedalStart is a task management application designed to help users create, view, edit, and delete tasks. The application consists of a frontend built with React and a backend powered by Node.js and Express, with data stored in a MongoDB database. This application demonstrates a full-stack implementation with RESTful APIs and a responsive user interface.

Links

- **Live Site Link:** <http://pedalstart.s3-website-us-east-1.amazonaws.com/>
- **GitHub Repository:** <https://github.com/Ac-11/PedalStart-TaskManagementApplication>
- **Backend API:** <http://18.205.252.53:5000/api/tasks>
- **Demo Video Link:** <https://drive.google.com/file/d/1q6sBr2eNhHectlb2G91SfgFeToEkcNBd/view>

Technologies Used

- Frontend: React, Bootstrap
- Backend: Node.js, Express
- Database: MongoDB
- Deployment: AWS EC2, S3, Netlify

Features

- Add new tasks with title, description, and due date.
- View a list of all tasks.
- Edit existing tasks.
- Delete tasks.
- Responsive design for mobile and desktop views.

Installation Instructions

Backend Setup

1. **Clone the repository:**

```
git clone https://github.com/Ac-11/PedalStart-TaskManagementApplication.git
cd PedalStart-TaskManagementApplication
```

2. **Install dependencies:**

```
npm install
```

3. **Set up environment variables:** Create a `.env` file in the root directory and add the following:

```
MONGODB_URI=mongodb://localhost:27017/your-database-name
PORT=5000
```

4. **Start the backend server:**

```
pm2 start server.js
pm2 startup
pm2 save
```

Frontend Setup

1. **Navigate to the client directory:**

```
cd client
```

2. **Install dependencies:**

```
npm install
```

3. **Change API routes:** In the `src` folder, update the API URLs in the components to point to your backend server's IP address. For example, change:

```
const response = await axios.get('http://localhost:5000/api/tasks');
```

to:

```
const response = await axios.get('http://<your-backend-ip>:5000/api/tasks');
```

4. **Build the frontend:**

```
npm run build
```

5. Serve the frontend:

```
pm2 serve build 3000 --spa
```

Deploying the Frontend on AWS S3

1. **Create a new S3 bucket:**
 - Name your bucket (e.g., `pedalstart`).
 - Enable static website hosting.
 - Upload the contents of the `build` folder to the S3 bucket.
2. **Set permissions:**
 - Go to the bucket permissions.
 - Add a bucket policy to make the contents publicly accessible.
3. **Access your site:**
 - Note the S3 bucket endpoint and access your site using the given URL.

Usage Instructions

1. **Adding a Task:**
 - Click on "Add New Task".
 - Fill in the Title, Description, and Due Date.
 - Click "Add Task".
 - If any field is empty, a browser pop-up will alert the user to fill in all fields.
2. **Viewing Tasks:**
 - The homepage displays a list of tasks.
 - Click "View" to see the details of a task.
3. **Editing a Task:**
 - Click "Edit" next to the task you want to update.
 - Modify the Title, Description, or Due Date.
 - Click "Update Task".
 - After successful update, you will be redirected to the task list page.
4. **Deleting a Task:**
 - Click "Delete" next to the task you want to remove.
 - Confirm the deletion by clicking "Yes".
 - After successful deletion, you will be redirected to the task list page.

Backend Changes

- **Database Connection:** Ensure the MongoDB connection string in `server.js` is set to your database:

```
mongoose.connect(process.env.MONGODB_URI, { useNewUrlParser: true, useUnifiedTopology: true });
```

- **Server Port:** Ensure the backend server listens on the port specified in the `.env` file:

```
const port = process.env.PORT || 5000;
app.listen(port, () => console.log(`Server running on port ${port}`));
```

Screenshots on desktop and phone preview

1. Task List:

PedalStart - Task Management Application

Task List

task - 1	View	Edit	Delete
task -3	View	Edit	Delete

Add New Task

Made by Abhishth_Chatterji@UPES



PedalStart - Task Management Application

Task List

task - 1	View	Edit	Delete
task -3	View	Edit	Delete

Add New Task

Made by Abhishth_Chatterji@UPES



2. **View Task:**

PedalStart - Task Management Application

View Task

task - 1
testing new features
Due Date: 6/28/2024

Made by Abhish Chatterji@UPES



PedalStart - Task Management Application

View Task

task - 1
testing new features
Due Date: 6/28/2024

Made by Abhish Chatterji@UPES



3. Edit Task:

PedalStart - Task Management Application

Edit Task

Title

task - 1

Description

testing new features

Due Date

28-06-2024



Update Task

Made by Abhishth_Chatterji@UPES



PedalStart - Task Management Application

Edit Task

Title

task - 1

Description

testing new features

Due Date

28-06-2024



Update Task

Made by Abhishth_Chatterji@UPES



4. Add Task:

PedalStart - Task Management Application

Add Task

Title

Title

Description

Description

Due Date

dd-mm-yyyy

Add Task

Made by Abhishth_Chatterji@UPES

[in](#)[github](#)

PedalStart - Task Management Application

Add Task

Title

Title

Description

Description

Due Date

dd-mm-yyyy

Add Task

Made by Abhishth_Chatterji@UPES

[in](#)[github](#)

5. Delete Task:

PedalStart - Task Management Application

Delete Task

Are you sure you want to delete this task?

Yes

No

Made by Abhishth_Chatterji@UPES



PedalStart - Task Management Application

Delete Task

Are you sure you want to delete this task?

Yes

No

Made by Abhishth_Chatterji@UPES



Additional Notes

Design Choices

- **Responsive Design:** The application is designed to be responsive, ensuring usability on both mobile and desktop devices.
- **User Experience:** Browser pop-ups are used for validation and confirmation to provide immediate feedback to the user.

Challenges Faced

- **API Integration:** Ensuring the frontend communicates effectively with the backend, especially when deployed on different servers.
- **Deployment:** Configuring the EC2 instance and S3 bucket for seamless deployment.

Future Improvements

- **Authentication:** Implement user authentication to secure the application.
- **Advanced Features:** Add features like task categorization, priority settings, and notifications.
- **Enhanced UI:** Improve the user interface with more interactive elements and animations.