

Read Me

Setup and Installation:

Install Node.js and npm: If not already installed, download and install Node.js, which includes npm.

Install Dependencies: Open the terminal, navigate to both my-web-app and my-frontend directories, and run npm install in each to install all necessary dependencies.

Set Up MongoDB: Ensure MongoDB is up and running. You can use MongoDB Atlas for a cloud-based database solution.

Update MongoDB Connection String: In your backend configuration file, update the MongoDB connection string to point to your running MongoDB instance.

Start Backend: Run npm run start:backend from the my-web-app directory to start the backend server. By default, it will run on port 9000.

Start Frontend: Run npm run start:frontend from the my-web-app directory to start the frontend React app. By default, it will run on port 3000.

Run Both: Optionally, you can run both frontend and backend simultaneously by running npm run start:both from the my-web-app directory.

How to Use the App?

Access the App: Open a browser and navigate to <http://localhost:3000> to interact with the frontend.

Signup: Use the Signup option to create a new user account. The credentials will be stored in MongoDB.

Login: Once signed up, use the Login option to access the application using the credentials you provided during signup.

Add and View Data: Post login, you will see forms to add data. All data will be stored in MongoDB.

Features:

- Create projects by filling out the necessary information.
- Formulate teams and name them.

- Add members to created teams.
- Draft user stories for projects.
- View existing user stories.
- Prioritize user stories in the backlog.
- Create Teams
- Pick user stories for development.
- Create task lists for chosen user stories.
- Monitor the status of tasks.
- View Teams: Click on the '**View Team**' button to see a list of all teams that have been created.
- View Projects: Click on the '**View Projects**' button to see a tabular list of all the projects, along with relevant details like Project Name, Description, Product Owner, Manager, and Team.

Additional Features:

Add Team Members

- Fetching created Teams.
- Picking a team.
- Fetching registered users.
- Adding selected user(s) to a team.

Remove Team Members

- Fetching created Teams.
- Picking a team.
- Fetching users in the selected team.
- Removing selected user(s) from a team.

Create User Story

- Fetching created projects.
- Picking a project.
- Adding a description and priority to the user story.

View User Stories

- Fetching created projects.
- Picking a project.
- Viewing user stories and their priorities in a table, with the option to delete.

Assign User Stories:

- Access the page for assigning user stories to team members.
- View unassigned user stories of all projects assigned to your teams in tabular form.
- Each row should have an "assign button" enabling a team member to assign a user story to themselves. This assignment is stored in the Assigned table.

Create Tasks:

- Access the page for creating tasks for an assigned user story.
- View the list of assigned user stories in a dropdown.
- Click on a particular user story.
- Enter the task description.
- Assign a status from a dropdown (New (Default), In Progress, Awaiting Confirmation, Completed).
- Add the task to the Tasklist table.

View Tasks:

- Access the page to view the list of all tasks for user stories assigned to you.
- View the list of tasks in tabular form.
- Each row should display the status of the task in a dropdown with the status as the selected value.
- You can change the status of the task from the status dropdown.

New Features Implemented

Home Page for logged-in users, displaying:

- Teams the user is a part of.
- Projects assigned to the user's teams.
- User stories assigned to the user.
- "In Progress" and "Awaiting Confirmation" tasks assigned to the user.

Detail Pages:

- Team: Name and members list.
- Project: Name, description, product owner, manager, and team.
- User Story: Project name, description, and priority.
- Task: User story description, task description, and status.