IoT Assisted Food Donation and Waste Management System

🧑‍💻 Technologies

FrontEnd
⚛️ React.js
🧩 TailwindCss
🗺️ Google map Api

Backend
💚 Node.js
🛣️ Express.js
🍃 MongoDB
📩 Nodemailer (To send the email)
🖼️ Cloudinary (To store the event images)
🔐 JsonWebToken (For Authentication)

- **Features**

The System consists of three types of Users:

1. Admins
2. Donors
3. Agent

Admins: They control all the activities and accept/reject donations and select agents.
Donors: They are the driving users of the application who donate food.
Agents: They are responsible for collecting food from homes of Food donors.

Sign up, login and logout functionalities.

- **Donor Features:**

1. Donors submit requests for food donations along with some basic information.
2. Donors' requests for donations can be approved or denied, and they can simply Monitor the features.
3. Donors are able to examine their currently unfinished donations (if any).
4. Donor can update their profile.

- **Admin Features:**

1. Admin receive all the requests made by donors.
2. Admin can accept or reject the donation requests depending upon the details provided by a donor.
3. If accepted, admin can assign an agent to a donation for collecting donation from the donor's home.
4. Admin can view all the pending donations.
5. Admin can view all the donations that they have received.
6. Admin can also view all the agents in the application.
7. Admin can update their profile.

- **Agent Features:**

1. Agents will receive notifications from admins to collect food from donor's homes.
2. Agents can mark their collection upon collection of food from donor's home.
3. Agents can also view all those food donations which have been collected by them previously.
4. Agents can update their profile.

Getting Started with Create React App and Redux

This project was bootstrapped with Create React App, using the Redux and Redux Toolkit template.

Available Scripts
In the project directory, you can run:

npm start
Runs the app in the development mode.
Open http://localhost:3000 to view it in your browser.

The page will reload when you make changes.
You may also see any lint errors in the console.

npm test
Launches the test runner in the interactive watch mode.
See the section about running tests for more information.

npm run build
Builds the app for production to the build folder.
It correctly bundles React in production mode and optimizes the build for the best performance.

The build is minified and the filenames include the hashes.
Your app is ready to be deployed!

See the section about deployment for more information.

npm run eject
Note: this is a one-way operation. Once you eject, you can't go back!

If you aren't satisfied with the build tool and configuration choices, you can eject at any time. This command will remove the single build dependency from your project.

Instead, it will copy all the configuration files and the transitive dependencies (webpack, Babel, ESLint, etc) right into your project so you have full control over them. All of the commands except eject will still work, but they will point to the copied scripts so you can tweak them. At this point you're on your own.

You don't have to ever use eject. The curated feature set is suitable for small and middle deployments, and you shouldn't feel obligated to use this feature. However we understand that this tool wouldn't be useful if you couldn't customize it when you are ready for it.

Learn More

You can learn more in the Create React App documentation.

To learn React, check out the React documentation.