

## **LOGIN MODULE:**

Installed node.js on my computer:

npm install

For development:

npm run dev

For production:

npm start

For running the backend-code that is server.js:

Node server.js

For the tests file

npm init -y

npm install express npm install --save-dev jest supertest

Conncted the backend file to tests file

To run the tests:

npm test

npm test -- --coverage

This will output the coverage of the the files.

Note: If you want to run this module, remove: 'sanjanas' from both package json and lock json

## **VOLUNTEER MATCHING MODULE:**

Installed node.js on my computer:

npm install

For development:

npm run dev

For production:

npm start

For running the backend-code that is server.js:

Node server.js

For the tests file

npm init -y

npm install express npm install --save-dev jest supertest

Conncted the backend file to tests file

To run the tests:

npm test

npm test -- --coverage

This will output the coverage of the the files.

Note: If you want to run this module, remove: 'sanjana' from both package json and lock json

## **VOLUNTEER HISTORY AND NOTIFICATION PAGE:**

Installation

Prerequisites

Node.js (v16+)

npm (v7+)

npm init -y

Install dependencies

npm install express cors

Run the Backend server

node notify\_backend.js

or

node volunteer\_history\_backend.js

Testing

The project uses Jest and Supertest for testing.

Install Dependencies

```
npm install --save-dev jest supertest
```

Run All Tests:

```
npm test
```

You should create the test folder and put test files in the folder.

And If you want to run this module, remove: 'shiv' from both package json and lock json

## **EVENT MANAGEMENT MODULE**

Prerequisites

- Node.js (v14 or later)
- NPM (Node Package Manager)

Installation Instructions

Install the necessary Node.js packages by running:

```
npm install
```

The project dependencies include:

- Express.js, Body-parser, EJS, Express-validator and Cors

Dev dependencies include:

- Jest, Supertest, Nodemon, NYC and Dotenv

Running the Project

- npm serverV.cjs

It will display:

- Server is running on <http://localhost:3000>

Copy and paste the link into any web browser to open the webpage.

To stop the server, press Control + C.

Running Tests:

To run the unit and integration tests:

- npm test

This will execute all the test

To get test coverage enter

- npm test -- --coverage

Access the Application:

Once the server is running, you can access the form at:

<http://localhost:3000>

## Configuration Details

The project uses a .env file to manage environment variables. You can configure the PORT variable inside .env (default is 3000 if not provided).

Example .env file:

- PORT=3000

## Troubleshooting Tips

Port Conflict: If the default port 3000 is already in use, then will kill and process in 3000 and start server again.

- lsof -i :3000
- kill -9 <PID>
- node server.cjs

## USER PROFILE MANAGEMENT:

- Prerequisites: node.js and install npm
- Installations done: npm, express, multer, cors, jest – coverage
- Run the backend server: npm backend.js
- Run the VPM.html, enter info to form and view the saved data by copy pasting [‘http://localhost:4000/volunteer’](http://localhost:4000/volunteer) from VPM.html’s script section. The server run can be cancelled by the shortcut ctrl+c
- Run the test: npm test
- To get test coverage: npm test – coverage