**NAMRIA**

**GSDD PERSONNEL TRACKING SYSTEM**

**OJT**

**VALDEZ, HASHIM JAHARA DOMINGUEZ**

**# Personnel Tracking System**

Tracking system for National Mapping and Resource Information Authority(NAMRIA) Geospatial Information Systems Branch(GISMB)/ Geospatial Systems Development Division(GSDD) employees.

> Keywords: Time in & time out and employee status updates

**## Development server**

This web app was built by ExpressJS, Angular 15, MSQL, and NodeJS. So we have a 2 server need to run in development. While 1 (Express) in deployment.

Run `ng serve` in root directory for a angular dev server. Navigate to `http://localhost:4200/`.

Enter `cd backend` to change location to backend files, and run `npm start` in express dev server.

The application will automatically reload if you change any of the source files.

**## Deployment**

Run `ng build` to build the project. The ExpressJS is the one we will deploy. The build artifacts will be stored in `backend/dist`

**## Environments**

Backend environment:

1. DB\_USER

2. DB\_PASSWORD

3. DB\_SERVER

4. DB

5. GMAIL\_USER\_AUTH

6. GMAIL\_PW\_AUTH

**## Running unit tests**

Run `ng test` to execute the unit tests via [Karma](https://karma-runner.github.io).

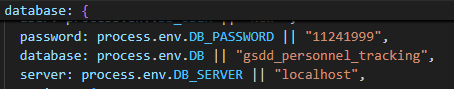
**## Running end-to-end tests**

Run `ng e2e` to execute the end-to-end tests via a platform of your choice. To use this command, you need to first add a package that implements end-to-end testing capabilities.

**## Further help**

To get more help on the Angular CLI use `ng help` or go check out the [Angular CLI Overview and Command Reference](https://angular.io/cli) page.

**#how to set the web**

First you need to check the config.js its located at the backend and look for the database and check if your database is in the config,js look like in the image

# And the second before running the web you need to go in the terminal and type [npm install --legacy-peer-deps](https://stackoverflow.com/questions/66239691/what-does-npm-install-legacy-peer-deps-do-exactly-when-is-it-recommended-wh)

After installing the npm in the project, you can start running the web by typing cd backend in the terminal and npm start to start the backend, and to start the front end, you need to create another terminal or slip the terminal where you create the backend and type "ng serve and wait to generate the [localhost](http://localhost), and remember the [localhost](http://localhost) is always like this: localhost:4200. It will end in 4200, not 80; the 80 is used for the API. And now you can access the web but you cannot login in the web so we will use the postman to request in for the account go to the postman and if you don’t have a postman in your desktop you can download it in the link that ill give to you https://www.postman.com/ and if you have an postman just go to the postman and follow this guild

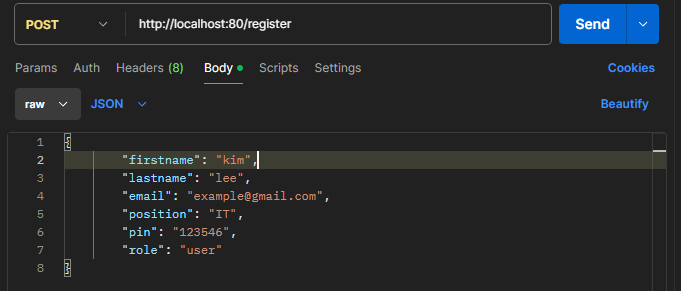
**Initialize user using postman**

If database has no data, you can initialize user using postman to create a new account. Then you can use initialize account to create accounts to user specifically.

**Pre-requisite:**

* ExpressJS must be running in backend. Run npm start at backend.
* Node\_modules is installed using npm install.
* Postman is installed in machine.

**Postman:**

* In backendroutes/account.routes.js. all route is in there so you can use it appropriately.
* In line 23.. post(‘/register’) is the route for inserting account in database.
* In line 25, we can see the fields that we need to fill up populate the new user.
* Open Postman. Set request to POST.
* In the URL, enter the port you’ve assigned. In default it is assigned in port 80. It should look like this:
* Then fo to Body tab and populate it like this:
* The role should be admin if you want to add accounts using the initialize account that you created and remember the roles are key sensitive do not use upper latter always small latter.
* And finally, click send.