Service Manual

GOVERNMENT
PORTAL

CONTENTS

1 RULES AND POLICES FOR ENGINEERS

- All engineers must follow the instructions on this service manual for any sort of technical inspection, service upgrade and or maintenance.
- All system functions contain commented details about parameters and functional code, any updates to existing or new functions should be commented likewise for future operations.
- At any moment shall not an engineer place or reverse engineer code which may damage, outflow, or provide super user privileges to a user including the engineer and any code that is deprecated should be commented and removed from operations.
- Tangible components can only be operator with supervision of a head engineer, these sections are commented within the application classes.
- All engineers must follow governing law of the country and implement only allowed services and data collection should strictly follow government requirements.
- Any rules here and Government enforced laws should not be broken by any engineer at any moment.

2 ENVIRONMENT SETUP

Software repository should be cloned from the GitHub Repository, engineers should change active branch to **developer** before proceeding with work.

Requirements:

- Active Git CLI/ Client installed
- Typescript supported IDE (Recommends Microsoft Visual Studio Code)
- Active Network Connection
- NodeJS and npm/yarn package manager
- Authorised Service Letter and Manual

Step 1: Open an integrated terminal for the working project, the terminal should be opened from the project's main directory where the **package.json** file is present.

Step 2: Depending on the package manager currently present on the system proceed with project installations: **npm install / yarn install**, wait for the installation to complete. (This may take several minutes)

Step 3: Once the installation is completed, execute command **npm start / yarn start** this will run the project locally. Wait for the project to compile and open host **http://localhost:4242/** from the browser.

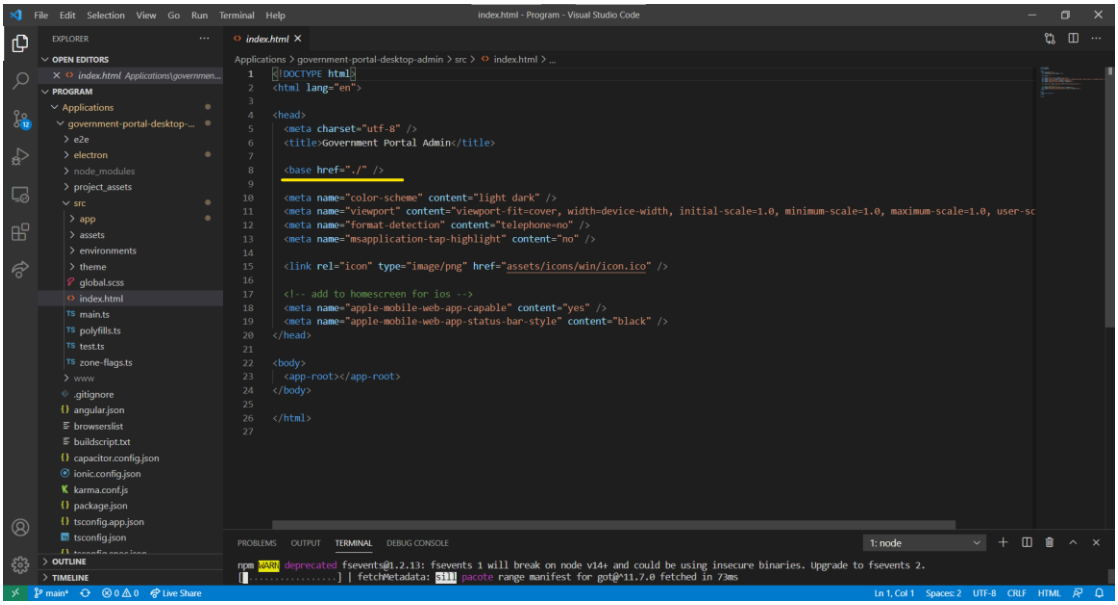
Notes: For gracefully exiting the server after service, press **ctrl+c**.

3 SOFTWARE APPLICATION SERVICING

Please navigate to /Program/Applications in the cloned repository to access software applications of the Government Portal System.

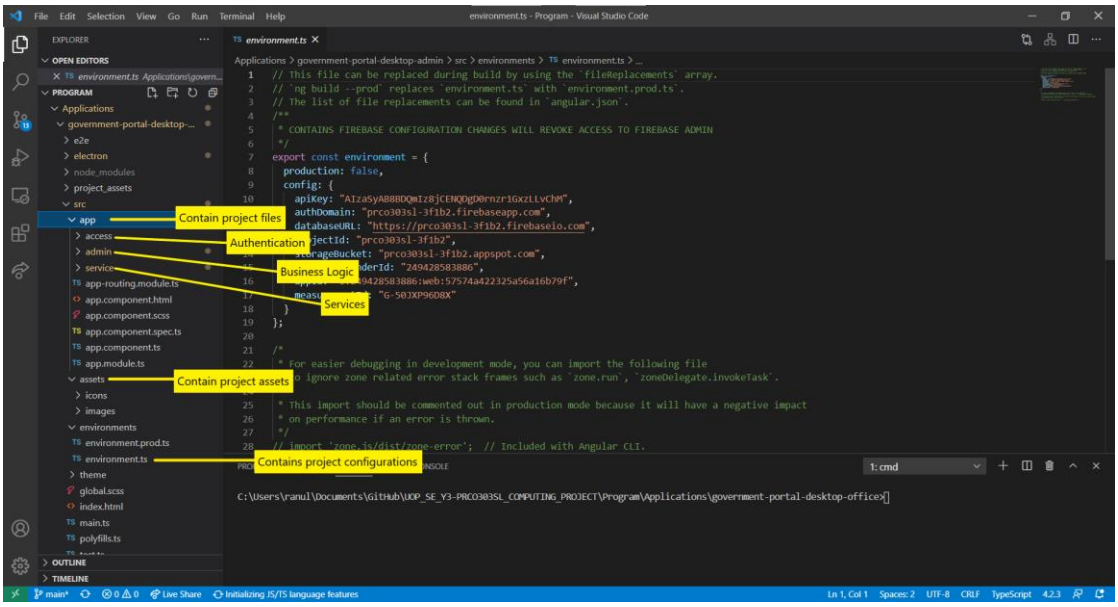
3.1 SERVICING AND UPGRADING DESKTOP APPS

For testing the application, make sure the `<base href="./"/>` in the index.html file is changed to `<base href="/" />` after the application testing is completed. Revert this process for distribution.



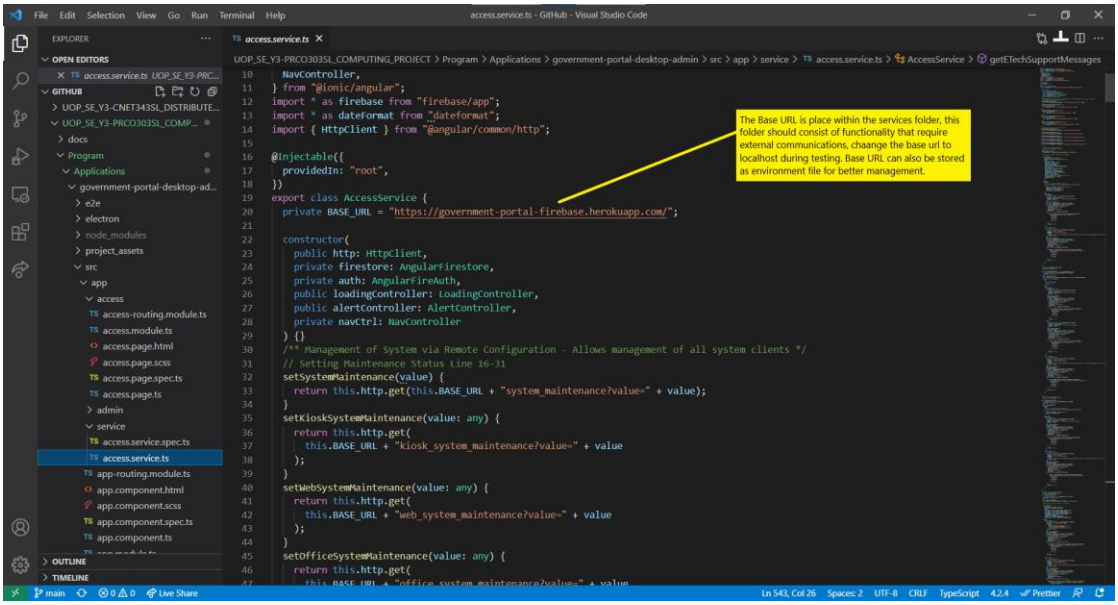
3.1.1 SOFTWARE PROJECT STRUCTURE

The project is modular is design and allows easier maintenance, follow the instructions below.



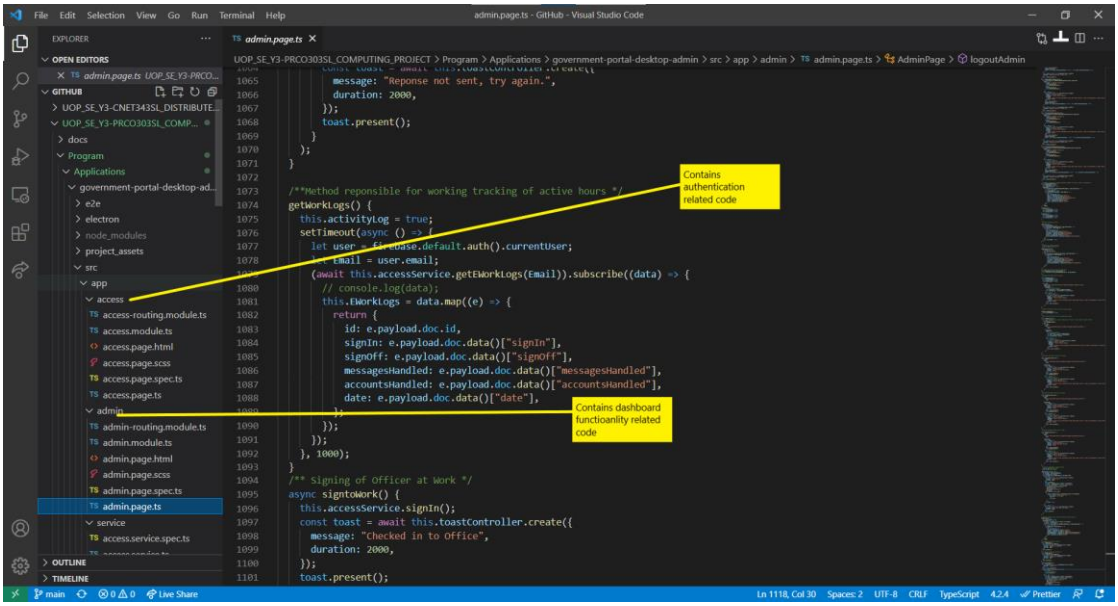
3.1.2 SOFTWARE SERVICE MANAGEMENT

The project is storing all services in the **services** folder, any future updates should follow the same procedure.



3.1.3 SINGLE PAGE APPLICATIONS

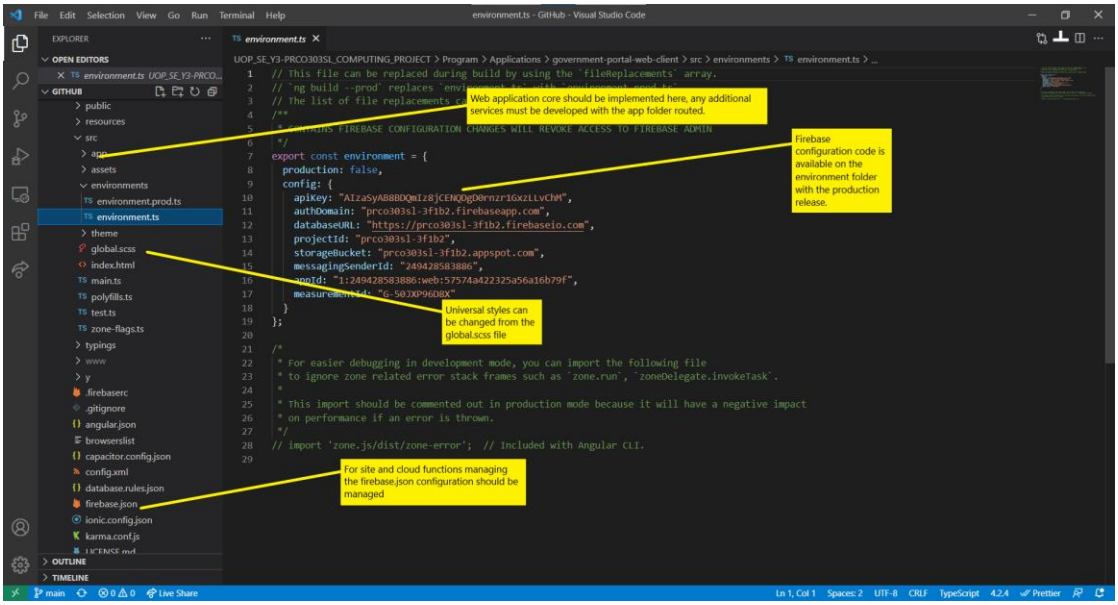
The project is following DRY, SOLID principles for developing Single Page Applications (SPAs) these terminologies should be followed to maintain both scalability and performance.



3.2 GOVERNMENT PORTAL WEB

3.2.1 SOFTWARE PROJECT STRUCTURE

The project is modular is design and allows easier maintenance, follow the instructions below.



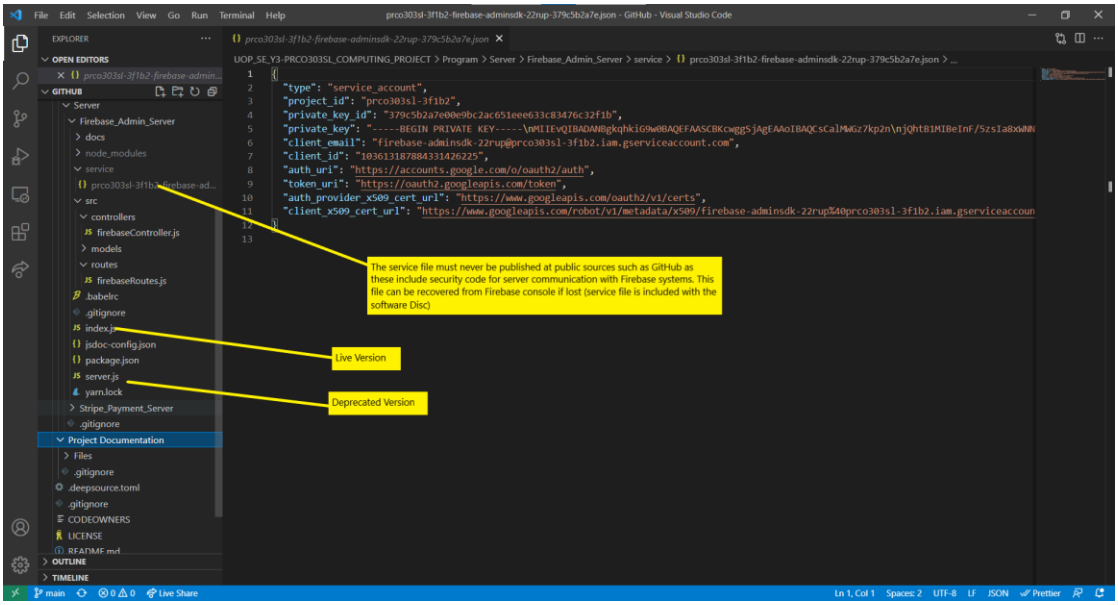
yarn start will compile and run the project; base routes are not required to be changed for the web app as with desktop counterparts.

4 SERVER APPLICATION SERVICING

Please navigate to /Program/Server in the cloned repository to access software applications of the Government Portal System.

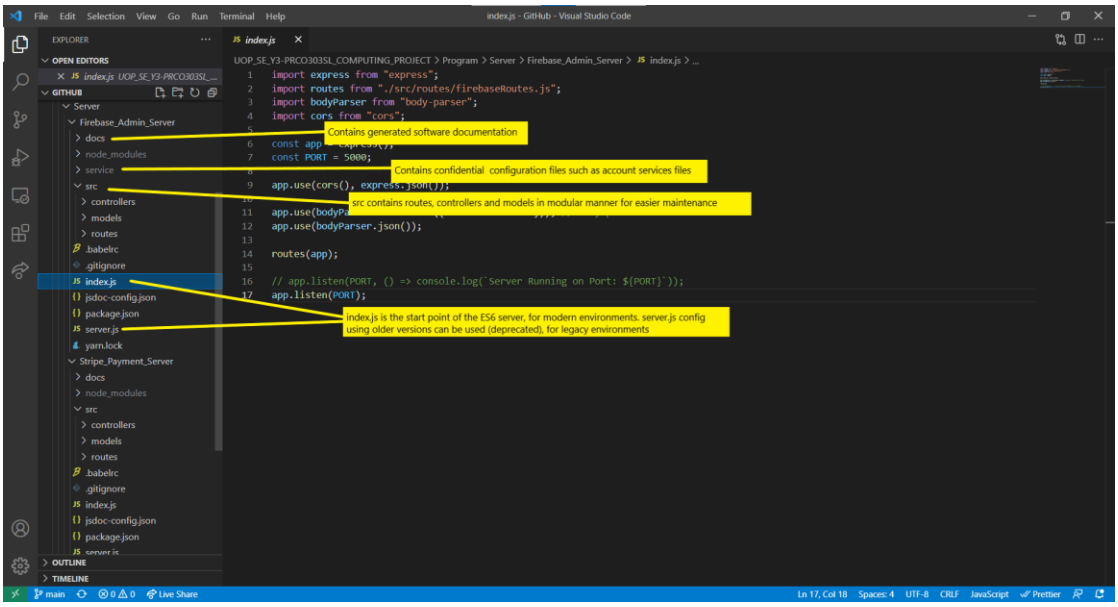
4.1 SERVICING AND UPGRADING

For testing the server, first install the server with **yarn install** and to run the server use **yarn v2 (live) yarn v1 (deprecated)** currently v1 is deprecated but can be used on legacy environments that does not support ES6.



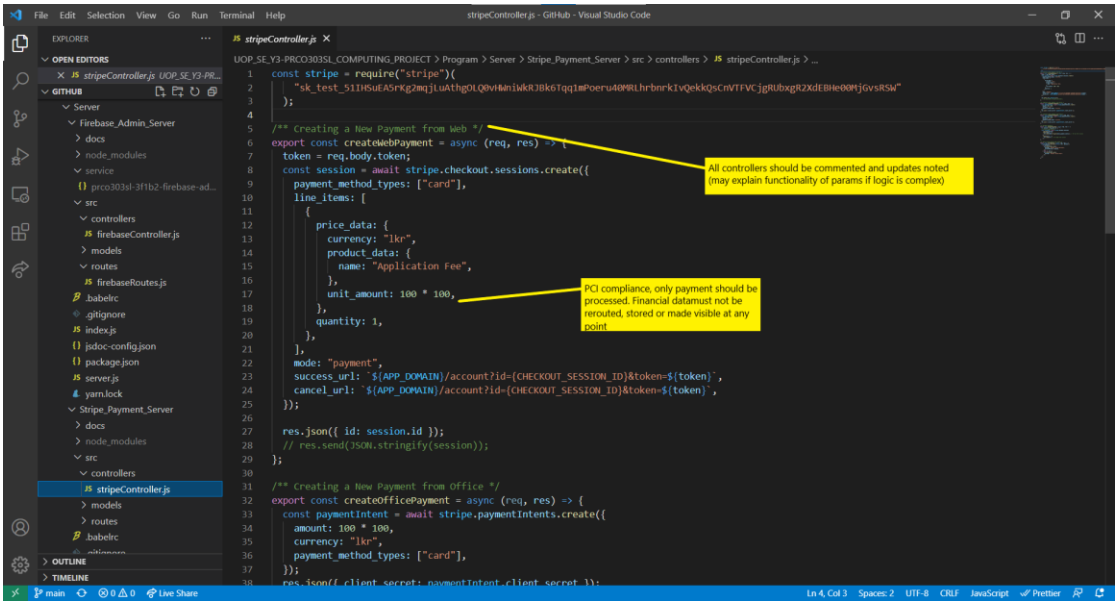
4.1.1 SERVER PROJECT STRUCTURE

The project is modular is design and allows easier maintenance, follow the instructions below.



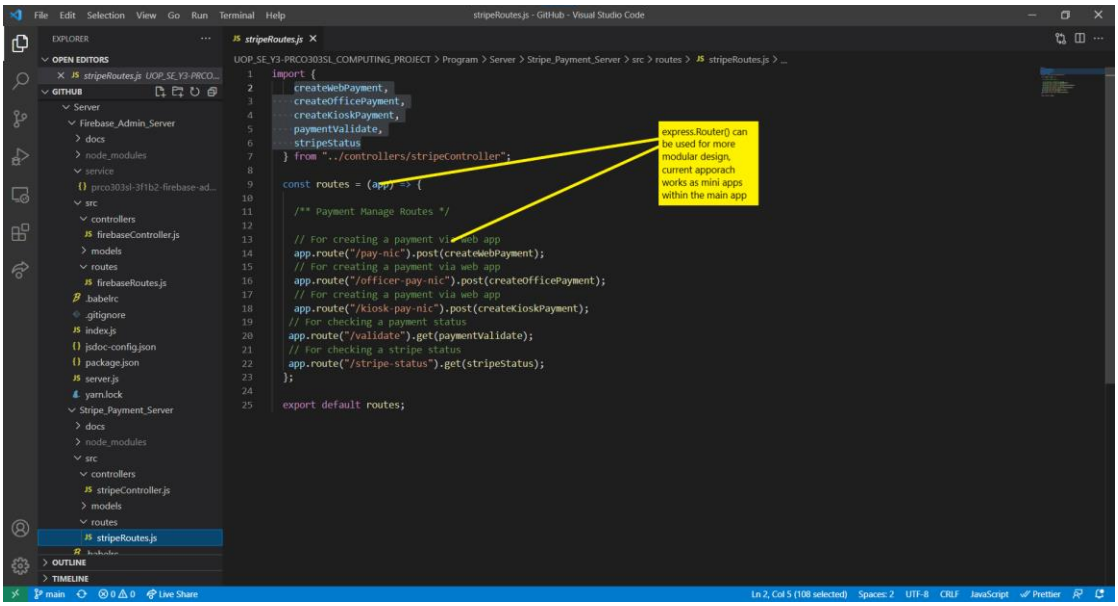
4.1.2 PCI COMPLIANCE

PCI compliance, only payments should be processed. Financial data must never be rerouted, stored, or made visible at any point during any upgrade. The stripe secret code should be only kept within the controller or environment file.



4.1.3 MODULAR ROUTES

Modular routes can be used if required under the main app to act as mini apps.



~ END OF SERVICE MANUAL | UPDATE WITH EACH SUITE VERSION RELEASE ~