ARCHITECTURE

The CBD Design project is built using a RESTful Architecture (MERN Stack) with the Front end being a Single Page Application developed using React and Redux that connects to a REST API in the backend built using NodeJS. The data for the project is stored in a MongoDB Database.

We adopted a stateless approach by using a REST API for loose coupling and easier scalability. Since the project involved rapid prototyping with the design of the dashboard changing constantly, we wanted to keep the server code separate and untouched as we played around with different designs on the UI and deployed several versions parallelly.

Currently the dashboard is hosted in a single virtual box (Ubuntu 18.04 LTS) at **cbme.usask.ca**. The machine has a Nginx server running to host the webpage and the same server also acts as a reverse proxy for the Node Server (calls made **to cbme.usask.ca/api** are routed to the backend node server running on port 8081). We also have a mongodB server that stores the data and is accessible only locally to the Node Server. The Nginx server is secured using a certificate issued by certbot (letsencrypt.org) that refreshes automatically every three months. By using a reverse proxy, we only need to secure the Nginx server which automatically also secures the backend API that it reverse proxies for.

DATABASE

Our database **RCM-CBD** has three collections (or tables) **USERS, RECORDS and NARRATIVES.**

USER TABLE

This table is for storing user information and for authentication and providing additional information regarding residents. Entries in this table can be updated by admin users through the admin tab from the UI. We use an access type entry in the user table to distinguish users, It can be: admin, program director, competence committee member, academic advisor or resident. The access type of a user determines what pages they are able see on the UI. Access to the dashboard is controlled through the CAS (central authentication system) interface maintained by the University of Saskatchewan. Users are redirected to our dashboard after they login successfully and the CAS system provides us with their NSID (unique six-character code). We check for a corresponding entry for that NSID in our user database and then issue a JWT (JSON web token) for access to our dashboard.

RECORD TABLE

This table holds all the EPA observation records and is updated when new data files are uploaded for residents. This data is stored in pretty much the same format that the royal college portal exports. During data upload, records are tagged with a couple of new fields such as “program”, “rotation_schedule” and “academic_year” based on the resident for whom the data is being uploaded for. These tags are used in processing the data rapidly for the program evaluation dashboard.

SERVER

The server is built using expressJS on NodeJS and is fairly rudimentary and only does two major things. Firstly, it provides access control by validating the tickets issued by the authentication service and issues JSON web tokens and secondly it acts as a relay to get the required data from the mongodB serve. The server thread once launched is maintained by the pm2 process manager.

WEB GUI

The UI is built as a single page application using React and Redux. We use webpack a module bundler for creating a production build with all our web files bundled together as static assets.

React is used for all the DOM manipulations and routing is done by react-router. Each component on the page is built as an individual react component while the application logic is centralized using Redux. The styling is done in sass and the several sass files are compiled by webpack into a single css file. Apart from the main dependencies we also use a couple of other libraries such as moment (date manipulation), lodash (array handling), axios (http requests) and d3 (vector graphic visualizations).

We use a custom shell script to currently deploy our code when an update is made.

All our source code is version controlled using git and is available on GitHub for free under an MIT license

https://github.com/kiranbandi/cbd-dashboard-ui
https://github.com/kiranbandi/cbd-dashboard-server