A close up of a logo

Description automatically generated

Front end is written in ReactJS. There are 2 separate portals for normal users and administrators to use.

* User Portal: Users within the organization are able to access it to declare their health conditions as well as locations and people visited. Upon declaration form submission they will receive an instant response regarding based on the information submitted.
* Admin Portal: Administrators are able to access it to maintain static data such as locations, users etc. They can see all users’ submissions which have already been categorized and sorted automatically. For users showing suspected syndromes, administrators can send them notifications, contact them and generate contract tracing report of them.

Back end server is built on NodeJS. It listens to Rest API calls and send back responses asynchronously. It connects to a database running on IBM DB2 service and actively retrieve and write to the database. Both the NodeJS server and the database server are running on IBM Cloud which allows an agile build, test, deployment and release cycle.

Source Control: We use Git repository as source control tool.

Development: Build and start reactJS client side server. Build and start nodeJS server side server. Develop and see the outcome on the fly.

Test: Unit test cases are written to ensure each commit does not break any core functionality. Tests will be triggered automatically on Jenkins whenever a commit happens.

Build and Deploy: Jenkins will auto build and deploy a commit to UAT environment if all test cases are passed.

Release: Upon passing UAT test, a Jenkins job can be triggered manually to build and deploy a release version to Production server.