Docker Container where Postgres/Mongo code is hosted running on port Z

Docker Container where NodeJS code is hosted running on port Y

Docker Container where ReactJS code is hosted running on port X

* React JS

There will be several components in react JS used for creating this incident management application

React JS components in combination of redux components can be used for the UI Crud operations, redux will help to maintain the state of the application

Few components that will be there in ReactJS

1. Create Incident component (will take all the UI inputs for creating incident)
2. Incident selector component where list of incidents and users will be shown in UI, where the admin can choose to assign the incident
3. A user login component
4. A dashboard component which is shown after user login
5. Admin dashboard component
6. User dashboard component
7. A component to acknowledge the incident for the user, it will me an auto refresh for the user based on configured time to see the new incoming incidents
8. A component to filter the incident based on the criteria’s such as date, incident type
9. A component to show the details about the incident and delete the incident

* Node JS

There will be an incident service which will be having methods exposed for Crud operations

There will be a user service which will maintain the list of incidents for the user, the user service will call the incident service to do the crud operations

There will be an admin service which will help to assign the incident to a user

There will be a service which will have all the login creds for the user and categorise them as admin or user