Things for readMe:

* Tell user they need to set up permissions on their AWS account
* Tests need to be shareable events
* Testing Compatibility
  + Dependencies
  + Runtime compatibility
* Things to login to AWS accont
  + RoleARN
  + RoleSessionName
* We have:
  + History Log
  + Error Log
  + Settings
    - To change username, password or ARN
  + List of layers
    - Select multiple functions to add/remove layer to/from
  + List of functions
  + Login page
    - Required username, password and ARN

Things we learned

* Cant use forEach with async/await
* General
  + Set up multiple edge cases
    - If one failed, don’t want it to kick out of the middleware, want to display each array message, while allowing working functions to process
* TypeScript
  + Best practices for typescript typing – Do we follow implicit typing or explicitly type our variables
  + Typescript typing AWS Commands
  + Import vs require
  + Tsconfig.json setup, allowJs needed to be removed
  + Adding ‘.default’ on layerController imports in routers
  + Variable expecting result from AWS, giving errors because return could be AssumeRoleCommand or undefined. Putting in conditional to prevent undefined did not work
  + Forced to use ANY
    - Insanely nested objects
  + How to type nested objects
  + Changing requires to imports
  + Promise<void> for return next() – is this the write way or should it be more specific?
* Hiding DB URI so its not visible on Github
* Jest
  + Setting up account for testing purposes?
  + Cant test without having accurate ARNS and layer/function Names
  + Mock testing?
  + Frontend/Backend testing which to do
  + Attaching a cookie for specific tests
  + Deciding how specific expects should be
  + Frontend results res.data vs jest results response.body
* When adding h1 tag to history or error log, quick popup on unformatted data then good data pops in
* Cookies
  + With credentials in the axios call
* AWS
  + Making too many fetch requests, AWS was throttle us, put conditional into useEffect to only run if certain state set
  + Never made fetch requests faster – AWS is slow
    - Generate Loading bar
* Cors/Axios
  + app.use(cors({ origin: 'http://localhost:8080', credentials: true }));
* Never solved the problem of making
* Had to use setTimeOut for 5 seconds in testDependencies ,
* Images
  + Server static files to get image to display
  + Because we’re not in prod or bundled we have not yet had to host a file and pull it down
* Grafana/Visualizer – thought we’d use, did not need
* Webpack
  + Setting use TS files vs JS files