

El proyecto contiene una distribución de carpetas con el nombre de cada desafío.

En la carpeta "basededatos" podrán encontrar un modelo de datos realizado con MySQL Workbench, se incluyo un png exportado desde el mismo programa mencionado anteriormente para que puedan revisarlo de una manera mas rápida. Se incluye un archivo .sql con el nombre "toctoc_bd_test.sql" en el cual se encuentra el script para poder cargar la base de datos en MariaDB 10.4.17 además se incluye un archivo con el nombre "consultas.sql" donde se incluyen 3 consultas para 3 diferentes casos.

En la carpeta "javascript" en la cual se encuentran 3 archivos, 2 de los cuales contienen las funciones solicitadas, el tercer archivo es un "package.json" con el cual podrán ejecutar las funciones con los siguientes comandos:

- npm run run:search
- npm run run:sort

En la carpeta "nodejs" podrán encontrar el servicio de backend desarrollado con Express.js, manipula la base de datos SQLite 3 a través de Sequelize.js el cual es un ORM para la manipulación de información, tiene configurado EventEmitter para gestionar los eventos del agente, a su vez este ultimo fue construido con las dependencias "OS" y "GETMAC" para obtener la información del host. Para poder ejecutar el proyecto es necesario ejecutar los siguientes comandos:

- npm install
- npm run start

Este ultimo comando ejecutara la aplicación y creara en base a los modelos la base de datos si esta no existe.

Cuando se inicia la aplicación se levanta el servicio de express.js luego se comprueba si la base de datos se encuentra conectada, es en este momento donde crea la base de datos si esta no existe, posteriormente a través de EventEmitter el agente se conecta y empieza a actualizar la información en la base de datos con la memoria disponible que tiene el dispositivo cada un segundo. Este ultimo proceso se detiene automáticamente después del tiempo x en minutos que se encuentra definido en un archivo de configuración, para la prueba viene preconfigurado con 30 minutos.

El servicio de express.js cuenta con 2 endpoints funcionales:

- /get/resource/host este se encarga de traer la información del host (id, nombre, macaddress y memoria total del dispositivo)
- /get/resource/memory/:id donde :id vendría a ser el parámetro id del host por lo cual es dependiente de la primera consulta, este método traerá los últimos 30 registros agrupados por minuto, esto quiere decir que cada registro nuevo que se añade cada segundo será agrupado en la consulta para ser devuelto. Retornando el id del ultimo registro que se tomo en la agrupación, la cantidad de memoria disponible, y la fecha en formato string

En la carpeta “reactjs” podrán encontrar el servicio de frontend el cual fue desarrollado con React.js, incluye las dependencias axios para la llamada al API, Bootstrap para tener un estilo predefinido, chart.js para poder mostrar los gráficos solicitados, redux y saga, para poder manipular el estado de la aplicación de mejor manera, y hacer las llamadas al API a mas bajo nivel.

La aplicación se encuentra desarrollada con el principio de estructura atómica, actualmente en la aplicación solo hay 1 pantalla llamada “HomeScreen”, esta se encuentra dividida en “container” y “template” donde en el primero se encuentra la lógica de la pantalla y en el segundo su visualización recibiendo los datos del container por spread operator.

Se utiliza React.js como componentes funcionales acompañados de sus hooks, de estos se utilizan useEffect y useState, cuando HomeScreenContainer es renderizado se llama a la acción getHost de redux, lo cual desencadena la llamada al api /get/resource/host cuando esta información es traída se desencadena la acción de traer el historial de memoria desde el api, al tener esta información y solo si esto es correcto se muestra la gráfica y la información en pantalla. Podrán revisarlo en el código adjunto.

Para poder ejecutar el proyecto deben ejecutar los siguientes comandos:

- npm install
- npm run start

La prueba no fue realizada en su totalidad, sino que se desarrollo hasta el punto donde ustedes puedan evaluar lo que se hacer, espero puedan revisarlo y que sea de su agrado.