**Sales App Thought Note**

### The “Sales App” project is a sales application that allows users to enter sales, view the top 5 deals, and see the total revenue generated. The front end of the application was built using technologies such as Bootstrap, Font-Awesome, React, React-Bootstrap, React-DOM, React-Redux, React-Router-DOM, Redux, and SweetAlert2. The backend was developed using Bcrypt, Cors, Dotenv, Express, Jsonwebtoken, Mongoose, and Nodemon. APIs were generated from Express.

The Sales App is a user-friendly application that provides functionality for users to log in and log out. Only registered users can add sales, view the top 5 sales, and see the total revenue generated. If a valid user is not logged in, the application restricts access to these features.

To install and run the project locally, Node.js and npm must be installed on your system. After cloning the repository, you can install dependencies using npm install and start the development server using npm start.

The frontend code is organized into a components folder (which contains reusable components), a pages folder (which contains different pages that users can visit), and a Redux folder (which stores the information of the valid user to log in). The backend code follows the MVC method to split functionalities into different folders: models (which contains all schema files and the DB connection file), server.js (which has the basic starting route), routes (which has extended routes from server.js), controllers (where all functionality of the code is written), and middleware (which has middleware functionality).

APIs were tested using Postman to ensure they were working properly.

The Sales App is functioning as expected. It allows users to enter sales data, view the top 5 sales, and see the total revenue generated. The login/logout functionality and user registration are also working perfectly.

The Sales App project was a great learning experience. Working with Redux was challenging but ultimately successful as I was able to store the user token in the Redux store.