**Project Name: Curry Leaves**

**Type: Web Application**

**Software Used: React JS, Node JS**

**Summary**

Curry Leaves is an Online food Delivery application built using MERN stack. React JS for client-side front end and Express JS (A back-end web application framework for Node JS). Mongodb Atlas Database is used to store menu, user data, order history. The application features secure user signup, login system, add to cart, Order history functionalities. Once user log in they can view the menu and place order. Menu contains various dishes with their name, image, brief description and price. User can specify quantity of dishes and add to the cart. User can view the cart and place order. My orders page will display all the previous orders placed by that user. The application also features user authentication and validation functions. User passwords will be encrypted before stored in database. This is done by JSON web Token (JWT). After a user sign in to the application, the application then assigns JWT to that user. Subsequent requests by the user will include the assigned JWT. This token tells the server what routes, services, and resources the user is allowed to access. Server uses bcryptjs which is module that enables storing passwords as hashed passwords instead of plain text which will add security layer and prevent user credentials being compromised in case someone got access to database.

**Project Structure**

project structure is organized in a way that separates the backend and frontend code. It follows the MVC (Model-View-Controller) architecture pattern.

* **server Directory**

The server directory contains the backend code of the web application. The backend code is built using Node.js and Express.js. The server directory contains the following directories and files.

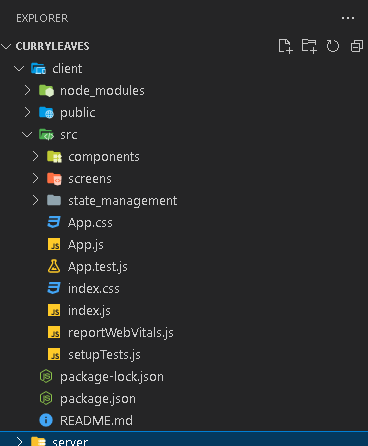
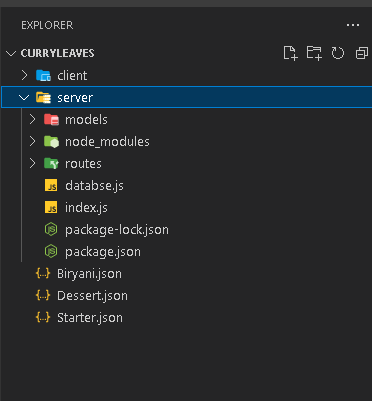
* + config directory: This directory contains the configuration files of the back-end application.
  + models directory: The Models directory contains the database Schema and functions that interact with database.
  + routes directory: This directory contains various routes acts as bridge between front end and back end. Routes will handle the request from the front end and send the response to those requests.
  + Index.js file: This file will be entry point of the back end.
* **client directory**

The client directory contains the front-end code of the application. It is built using React JS. The client directory contains following files and directory.

* + Public directory: This directory contains the HTML file that is displayed in the browser.
  + src directory: This directory contains source code of the front-end application and contains following files and directories.
    - * components: This directory contains all the reusable components of the application.
      * screens: This directory contain all the JSX files which will display various web pages (e.g., login page, home page etc)
      * state\_management: This directory contains a file which will handle various states across components.
      * App.js file: This file is the main file of the application which renders the components.
      * Index.js: This file is entry point of the front-end application.

**How to run the application locally**

* **Pre-requisite:** 
  + Mongodb Atlas is used as database for the project. Mongodb Atlas is Cloud based database. You need to have an account in mongodb website **(**[**https://www.mongodb.com/atlas/database**](https://www.mongodb.com/atlas/database)**).**
  + Active Internet connection.
  + Install node JS in your machine **(** [**https://nodejs.org/en/download**](https://nodejs.org/en/download) **)**
* Download Project to your local machine from this git repository.[**https://github.com/Vinayakumar18/CurryLeaves.git**](https://github.com/Vinayakumar18/CurryLeaves.git)
* Open the Project folder in your IDE (VS Code). Goto CurryLeaves folder.
* Open 2 terminals. Goto client folder in one terminal and server folder in another
* Type ‘**npm install** ‘ command in both terminal. This will install all the needed dependencies for client and server.
* At this point your project structure will look like this.

****

* Goto your Mongodb Atlas Dashboard and click Build Databases. And create a cluster Note down Username and Password.
* Near cluster0 click on connect. And copy the connection string.
* In the server folder goto database.js file. Here in code 2 paste the copied connection string to mongoURI variable. In mongoURI string replace <password> with password you have noted earlier.
* In server terminal type ‘ **npx nodemon index.js** ‘ This will start the back end server and connect to your mongodb Atlas. You will get Database connected message in terminal if connection is successful. Keep the server running.
* In Client terminal type ‘ **npm start**’ this will run client.
* Come back to Atlas dashboard and go to cluster0>collections and create Database remember the name of the database. Refresh once and you will see empty collections (e.g., starters, biryanis, desserts, users, orders)
* Now we have to import menu data to the database for that Open new command line type ‘**mongoimport --uri mongodb+srv://demo:<PASSWORD>@cluster0.x8i7wpf.mongodb.net/<DATABASE> --collection <COLLECTION> --jsonArray --file <FILENAME>**’

(replace <password> with earlier noted password. <DATABASE> with name of the database created. <COLLECTION> with name of the collection (to import data to starters collection you have give exact collection name) <FILENAME> with path to the json file you need to import). This is how it looks

**<mongoimport --uri mongodb+srv://demo:demo@cluster0.x8i7wpf.mongodb.net/demo--collection starters --jsonArray --file “C:\Users\Vinay\Desktop\Demo\CurryLeaves-main\CurryLeaves\Starter.json”>**

* Similarly import Starter, Biryani, Dessert.json files to their respective collections
* Refresh Atlas Database and see collections are imported.
* Now all the settings are done and web application can operate.

**How to Use the Application**

* Since all the requirements are met and both client and servers are running, we can use the application.
* First, we need to Sign up and login to the home page of the application.
* We can see login page and new user button and by clicking button we will goto sign up page.
* After creating user, we can now login using correct credentials.

(Note: When user is created. The user data will be stored in ‘users’ collection in the Atlas Database)

* We can see All the dishes reflected in the home page. We can specify Quantity and add to cart using Add button
* Navigate to Cart page to see the Cart with Total Amount and Checkout button
* Clicking On Checkout button will place the Order. And alert message will be displayed notifying estimated Delivery Time
* My Orders page will contain all the Orders history made by that particular user

(Note: In Database “orders” collection will store all the orders made by different user. In My Order page it will only display the logged in users order history using JWT)