Food Ordering cross platform (Android, IOS, Web Portal) Application:

Problem Statement:

- The purpose of this research is making an ordering food application based on Android, IOS, Web with Restaurant Profile, Restaurant Item, Setting Account, Order History, Browsing restaurant.
- The research method used in this research is waterflow model of System Development Life Cycle (SDLC) method with following phases: requirement definition, analyzing and determining the features needed in developing application and making the detail definition of each features, system and software design, designing the flow of developing application by using storyboard design, user experience design, Unified Modeling Language (UML) design, and database structure design, implementation an unit testing, making database and translating the result of designs to programming language code then doing unit testing, integration and System testing, integrating unit program to one unit system then doing system testing, operation and maintenance, operating the result of system testing and if any changes and reparations needed then the previous phases could be back.
- The conclusion of this research is to help customer in making order easily, to give detail information needed by customer, to help restaurant in receiving order, and to help courier while doing delivery

Technology Stack :

React-Native, React, Firebase, Yelp API, GooglePlacesAutoComplete API, React-native-navigation for navigation management, Expo CLI framework.

Explanation:

- Built Android, IOS and Web application Using react native cross platform technology.
- Using Google places API and the react native library GooglePlacesAutoComplete API, fetching the desired location for which user want to order food.
- Using Yelp API searching all the restaurant in that specific location and provide appropriate UI for fetching the restaurant with its rating and other information.
- Using Expo CLI framework manage the compilation for Android, IOS and for web application and run this app on Android mobile phone.
- Use the Firebase for storing all the details of an orders.

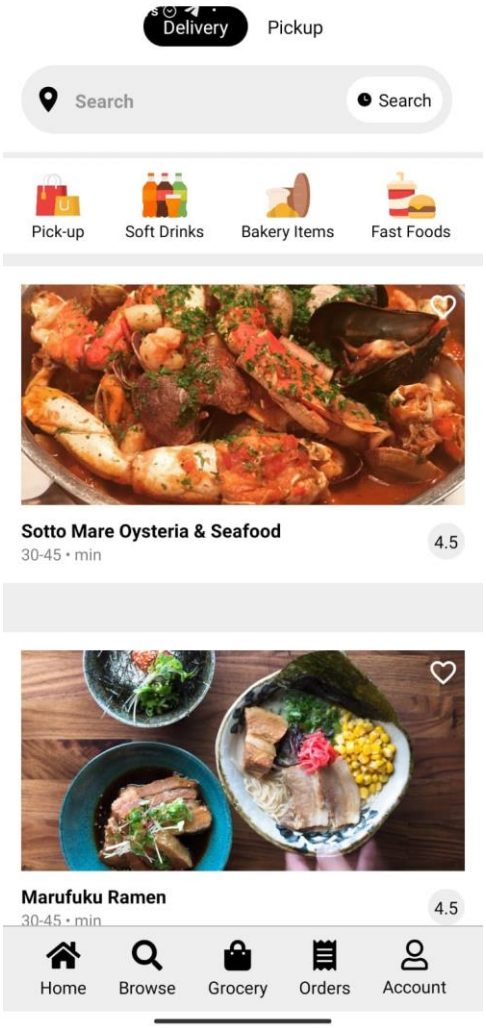
Folders and Files hierarchy with explanation:

Component folder contain two subfolders :

1: Home

2: RestaurantDetail

1.HOME



1.1 SearchBar.js :

This file is for searching the location for which user want to order food. In this component I use GooglePlacesAutoComplete API which help to find appropriate location to user with the help of autocomplete feature.

1.2.HeaderTab.js :

It shows two option Order and pickup. From them user select whatever they want to do.

1.3.RestaurantItem.js :

This component shows Restaurant detail like rating, time required to order, image of special item from restaurant etc.

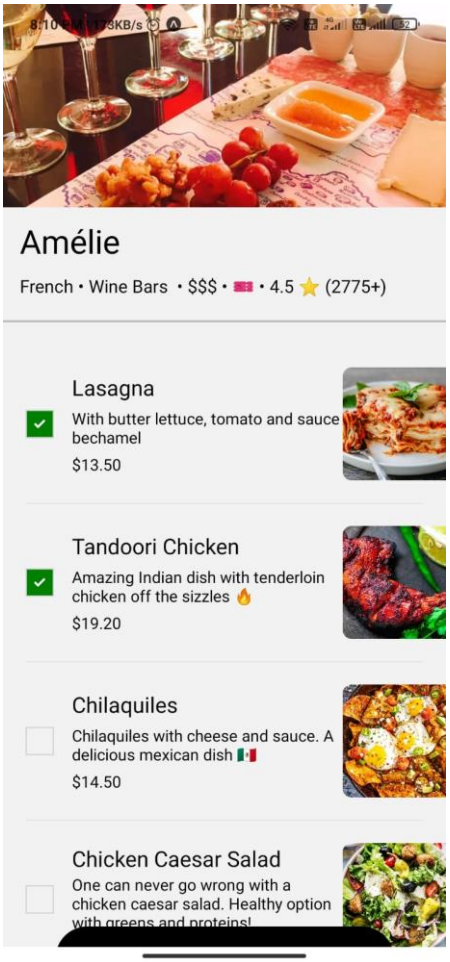
1.4.Cateory.js :

This component shows Pick-up, Soft Drinks, Bakery Items, Fast Foods. This is just for ease for the user to get fast food item.

1.5.BottomTab.js :

This component contains Home, Browse, grocery, Orders, Account which help to simplify UI.

2.restaurantDetail :



2.1: About.js :

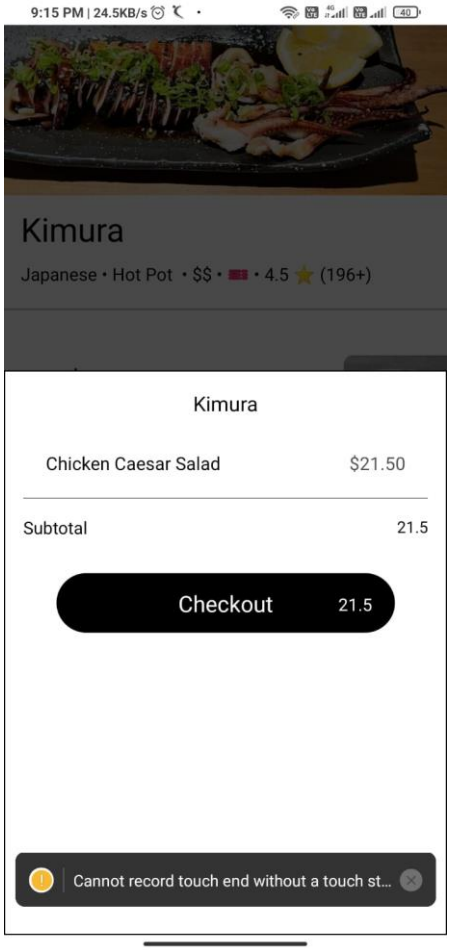
This component contains details about restaurant. When use click on specific restaurant this component is displaying.

2.2: MenuItem.js :

This component shows Menu Item from restaurant with description, image and pricing. When user select menu item then the item is added into cart.

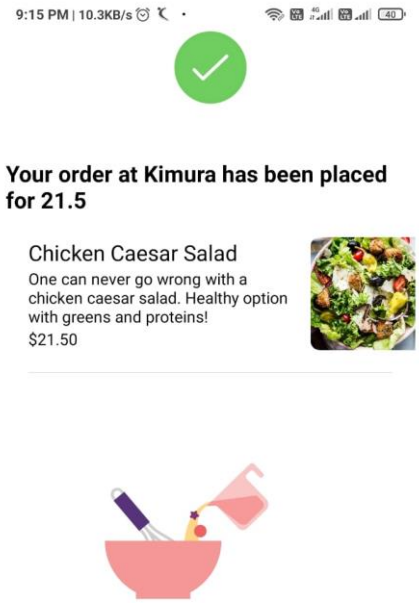
2.3: ViewCart.js

When user hit the cart button then the number of items with its price is calculated and shown to user.


9:15 PM | 24.5KB/s
Kimura
Japanese • Hot Pot • $$ • • 4.5 (196+)
Kimura
Chicken Caesar Salad
$21.50
Subtotal
21.5
Checkout
21.5
Cannot record touch end without a touch st...

2.4: orderItem.js

This component contains item name with price and the checkout button which confirms order.



Assets

It contains all the Images, icon and animation required for project.

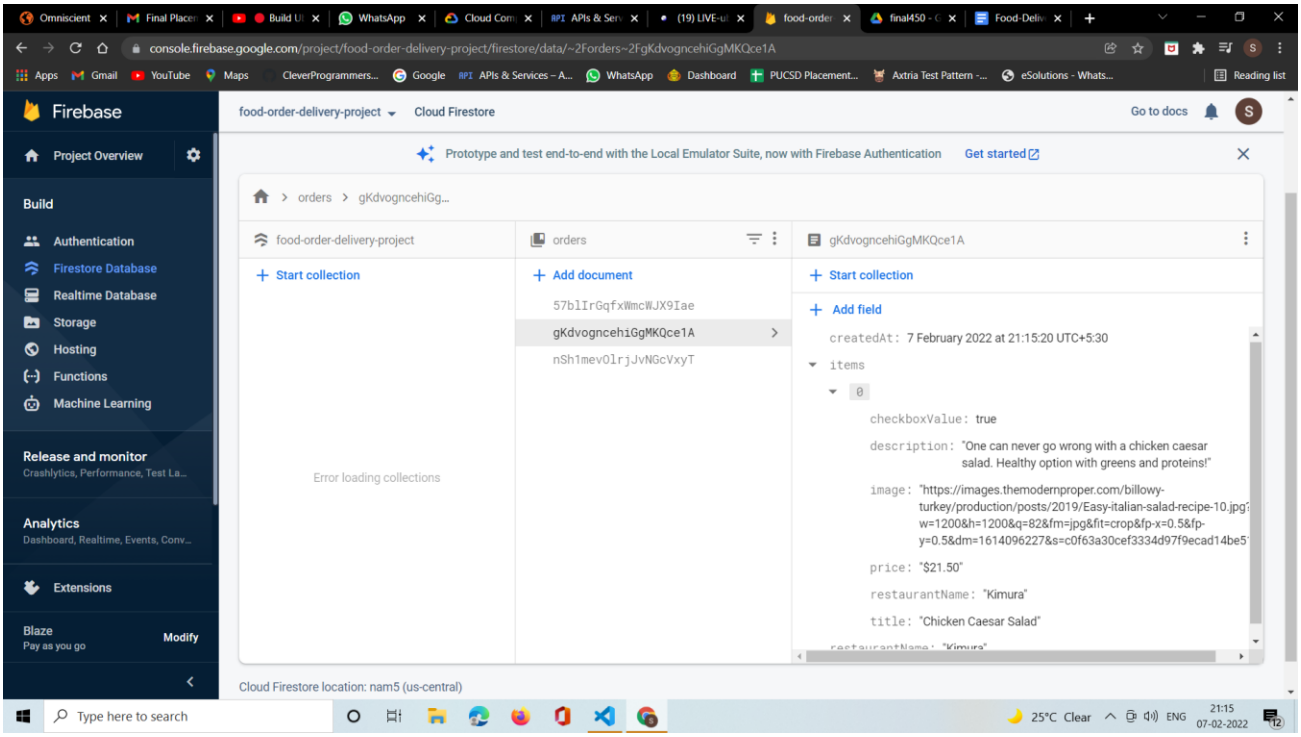
Screens

In screen folder there are three JS file which is screens on which navigation is done.

Home => restaurantDetail => orderCompleted

Firebase.js :

It contains all the configuration of firebase database.



Navigation.js :

It contains Screen Stack; it means it configure how our navigation works ex. Home => restaurantDetail => orderCompleted.

Redux:

For state management for example add to cart or remove to cart item.

🚀 How to use

- Install packages with `yarn` or `npm install`.
- Run `yarn start` to start the bundler.
- Open the project in a React runtime to try it:
    - iOS: Client iOS
    - Android: Client Android
    - Web: Any web browser

Adding Native Code

This project can be run from a web browser or the Expo client app. You may find that you want to add more native code later on. You can do this by ejecting the project and rebuilding it yourself.

- Run ***`yarn eject`*** to create the native projects.
- You can still run your project in the web browser or ***Expo client***, you just won't be able to access any new native modules you add.