Master of Technology Project Report

Contactless Menu Recommender system Installation and User Guide

Team Members

Mu Aohua - A0121924M

Lee Joon Hui Jeremy - A0048174A

Installation

Requirements

Node: v14.6.0 and above
Yarn: 1.22.4 and above
Browser: Latest Chrome or Firefox
OS: MacOS

DB setup

MongoDB installation
Clean installation without username and password

Neo4J installation
Create a new account with default username **neo4j** and with the password **asdf**

Seeding Neo4J Database
Run the 2 cypher scripts here in sequence

1. https://github.com/aohua/KG-food/blob/main/db/kg_food_db.cypher
2. https://github.com/aohua/KG-food/blob/main/db/kg_food_complementary.cypher

Backend setup

1. Start both MongoDB and Neo4J databases before running the actual backend
2. Access to the backend server code and execute the shell command, **run.sh**

```
a. > sh run.sh
```

Frontend setup

All the frontend code is under ***food-app-web-pwa*** folder.

You can access the deployed version here:
https://kg-food.web.app/

To run the project locally:

Install Node and Yarn:

1. brew install node
2. brew install yarn

Install dependencies:

1. cd food-app-web-pwa
2. yarn install

Start the dev server:

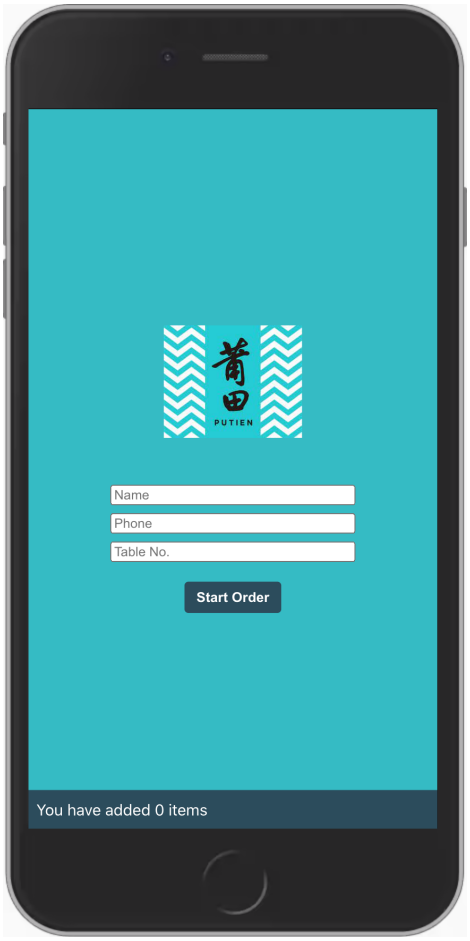
1. yarn start

You can now access the web app at:
http://localhost:3000/

Before you start to use the app, please make sure that you have already done the backend setup and have the backend server running at `http://127.0.0.1:5000`

User Guide

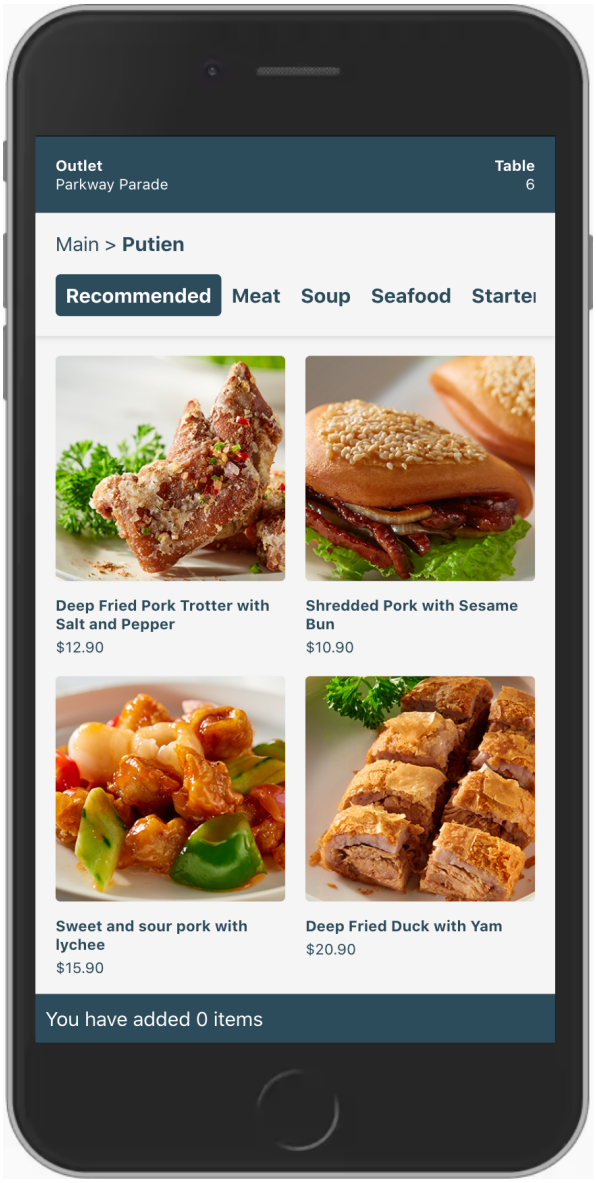
1. Open a modern browser(please ***don't*** use ***incognito*** or ***private*** mode) and go to the url: https://kg-food.web.app/ or http://localhost:3000/, if you are accessing the deployed version, at this point of time, the internet is no longer required, until you want to sync up with backend.



2. Fill in the details(Due to the limited time, we **didn't add any validation on the form**, please do fill the forms properly)

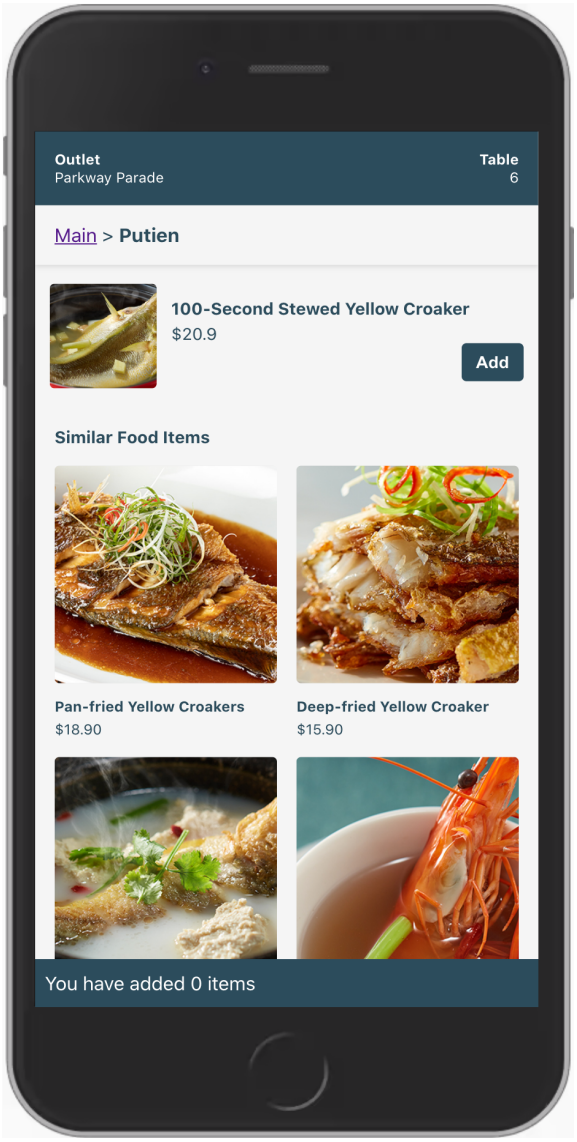
Popular Items Recommendation

3. Press start order, and you will see our recommended dishes, the recommendation section is based on the frequency of a dish that is purchased in the previous orders

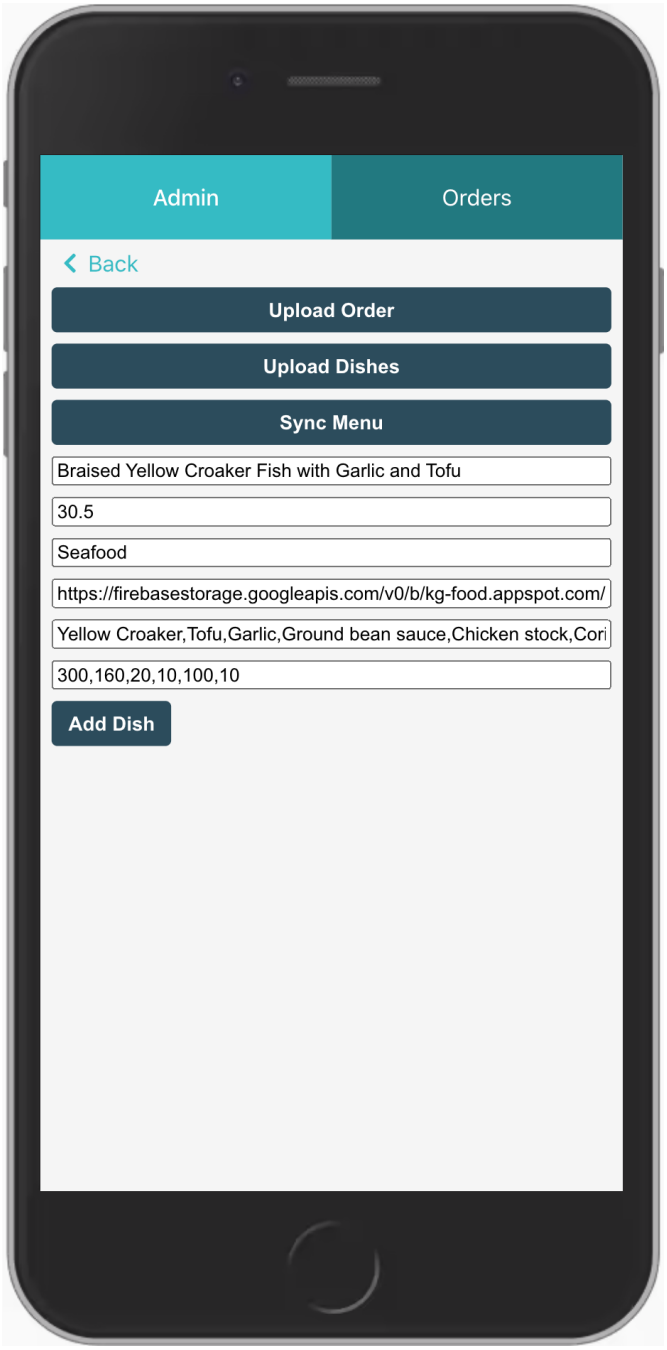


Similar Items Recommendation

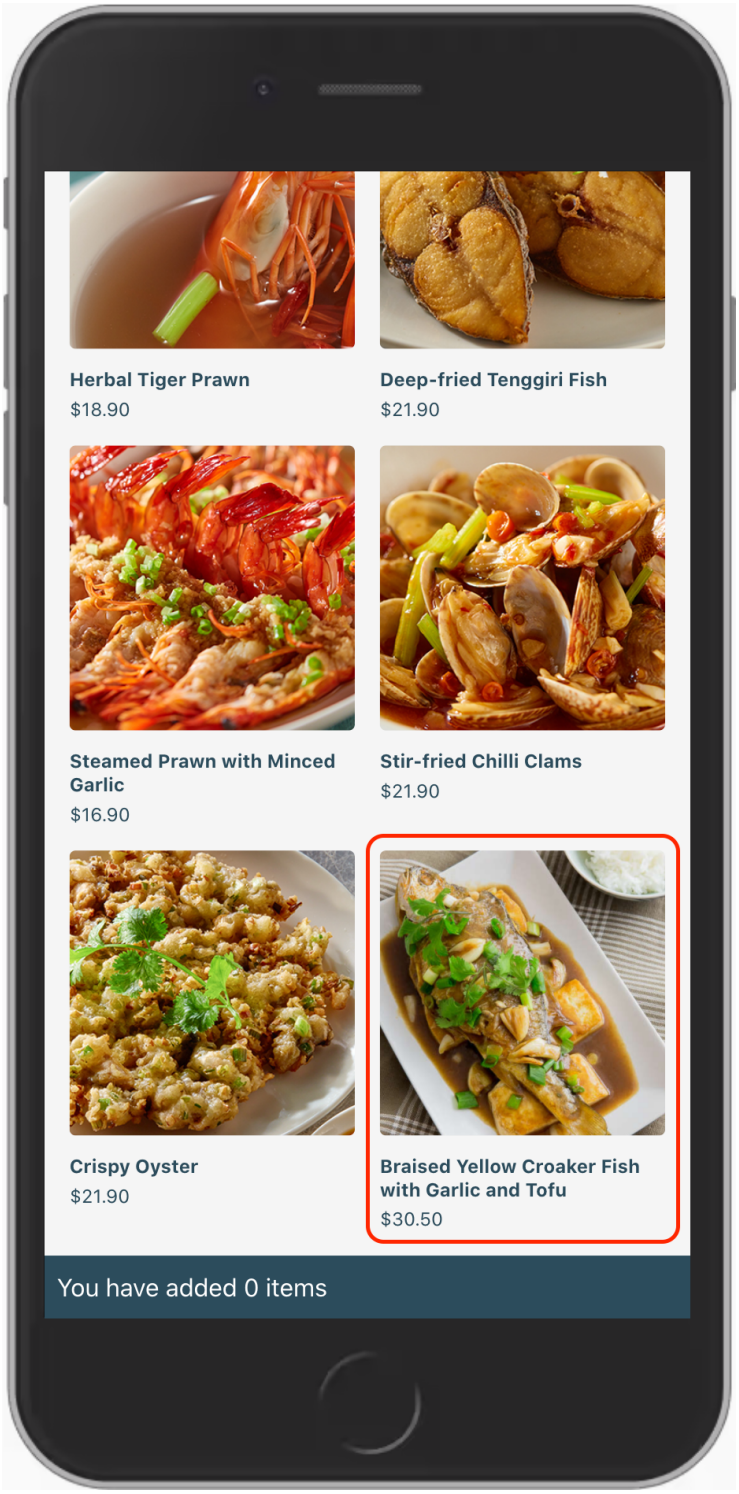
4. Because Putien has many yellow croakers dishes, let's use this dish as an example https://kg-food.web.app/food/123, under the **similar food items** section are the recommendation returned from the server, based on our KG for the dishes, frontend then sort it based on the tf-idf indexing similarity.



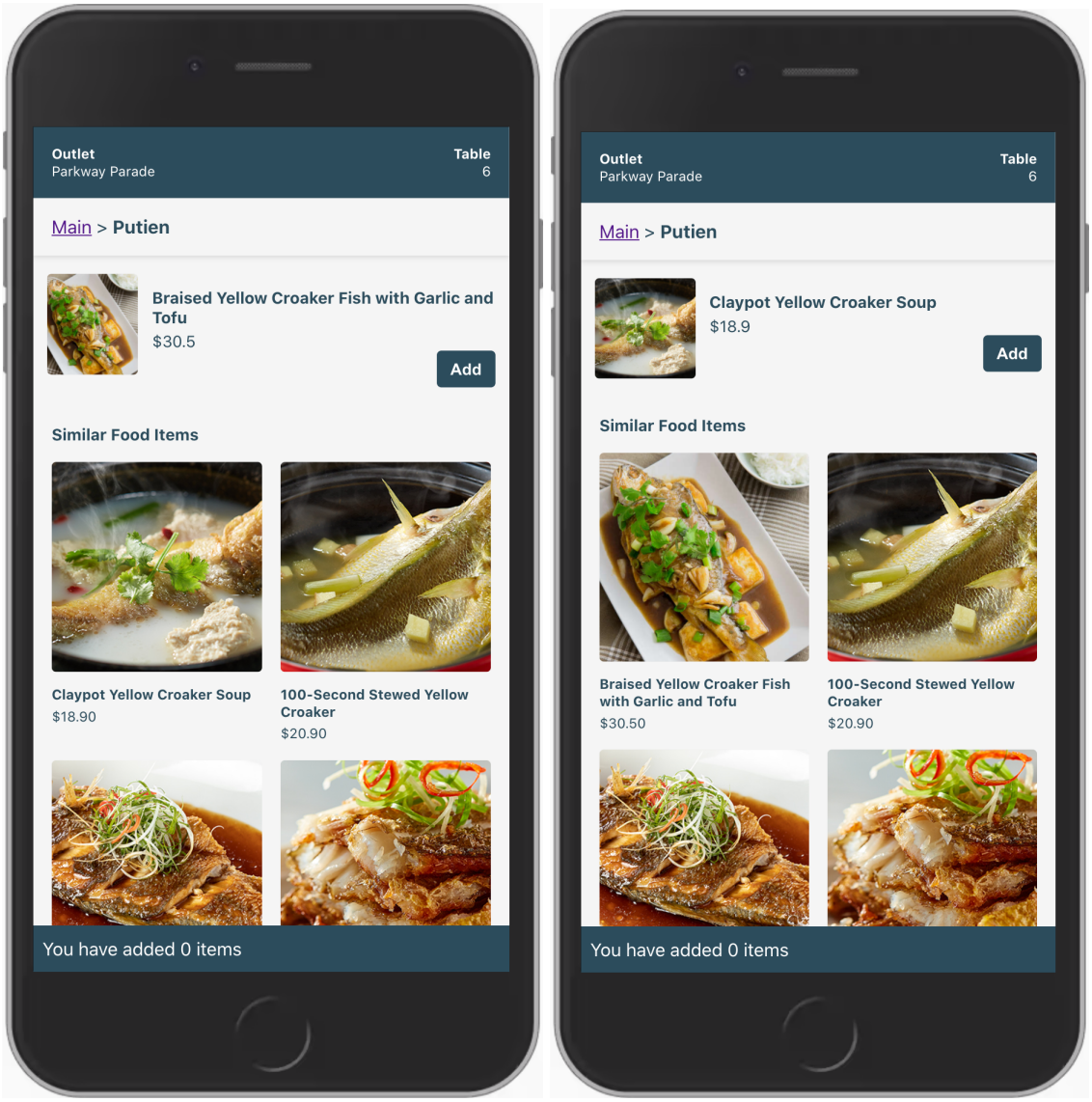
5. Long press the header, and you will be navigated to https://kg-food.web.app/admin, now try to add a new dish, I have **prefilled a new dish for your convenience**



6. After adding the new dish, go back to the menu https://kg-food.web.app/menu, you can find the new dish **under the seafood category**. Click on the dish or go to https://kg-food.web.app/food/165, to see the recommended items generated on the frontend.

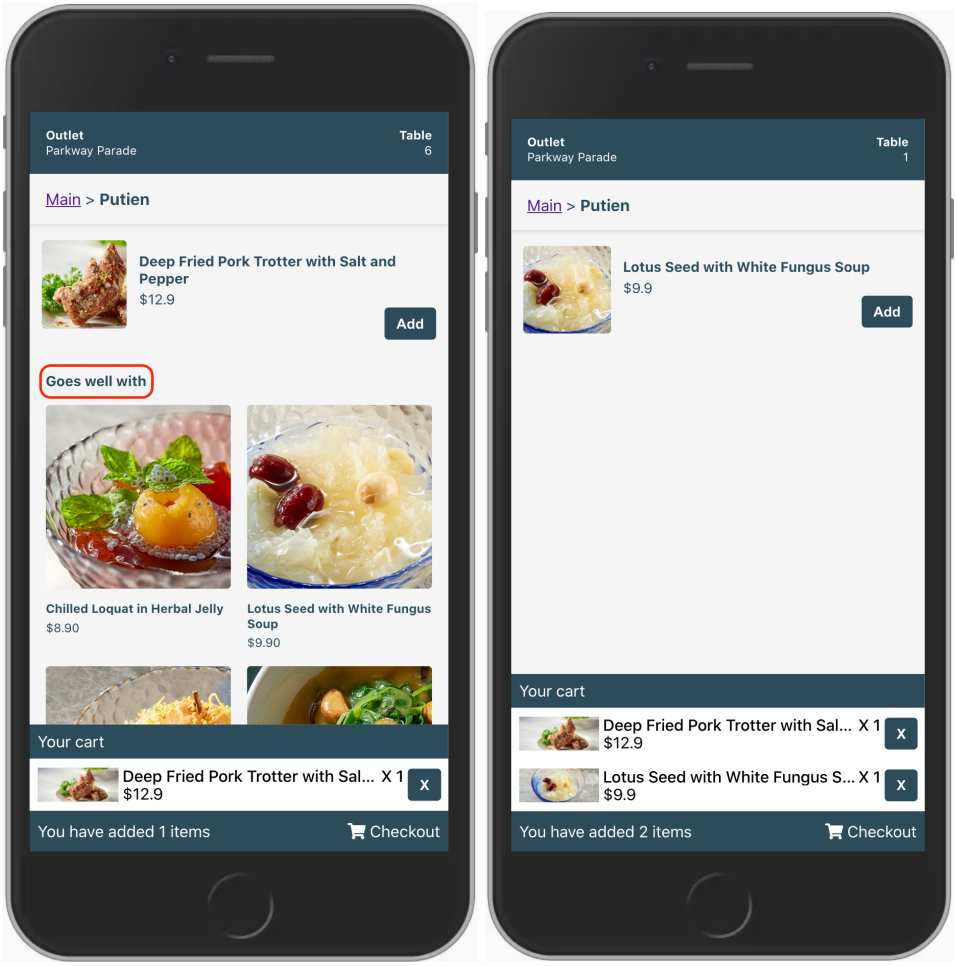


7. You will see that the new dish already has recommendations without sending any API to the server, the new dish is also added to other existing dishes' recommendations.

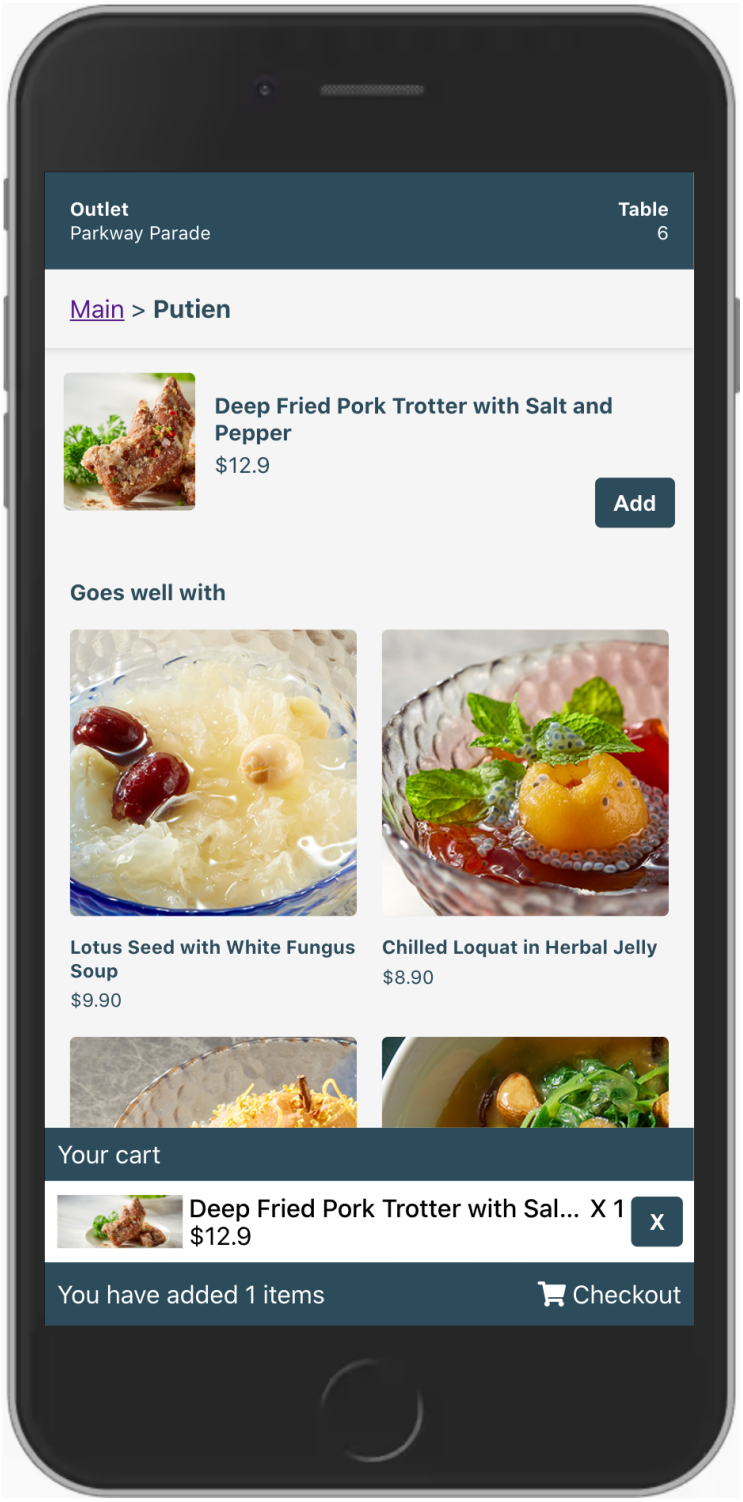


Complementary Items Recommendation

8. Now let's check Complementary item recommendation, go to https://kg-food.web.app/food/115, and add this item to the cart, you will see that the bottom section has changed from 'Similar Food Items' to 'Goes well with', which are a new set of recommendations returned from the server. Let's also add 'Lotus Seed with White Fungus Soup' to the cart and checkout. (Currently, the first 2 recommended items has the same ranking)

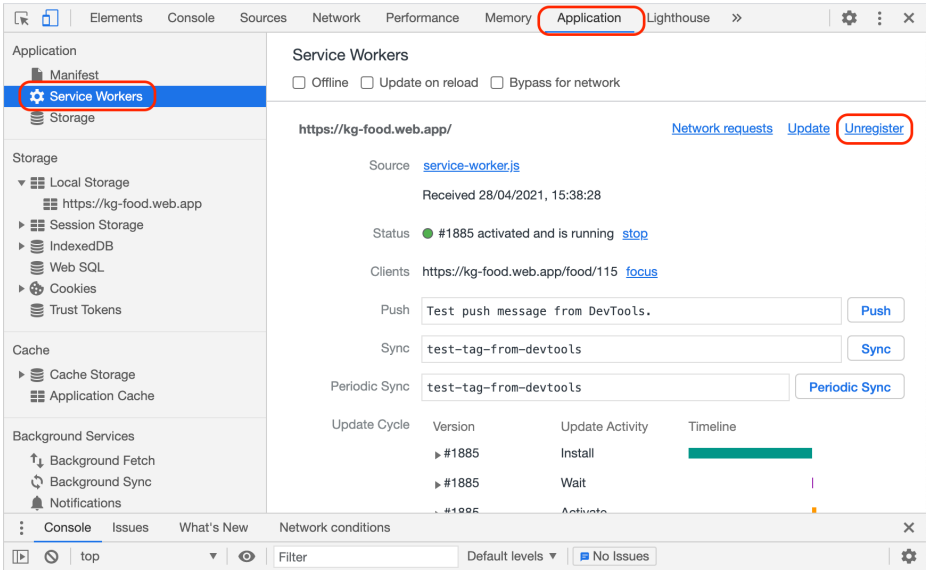


9. After checkout, now go back to https://kg-food.web.app/food/115, and add this dish to the cart again, now you will see that the ‘Lotus Seed with White Fungus Soup’ has become the first recommended item by the frontend model, it also works for the backend model, but you need to upload the client-side orders to the server to update the modal.

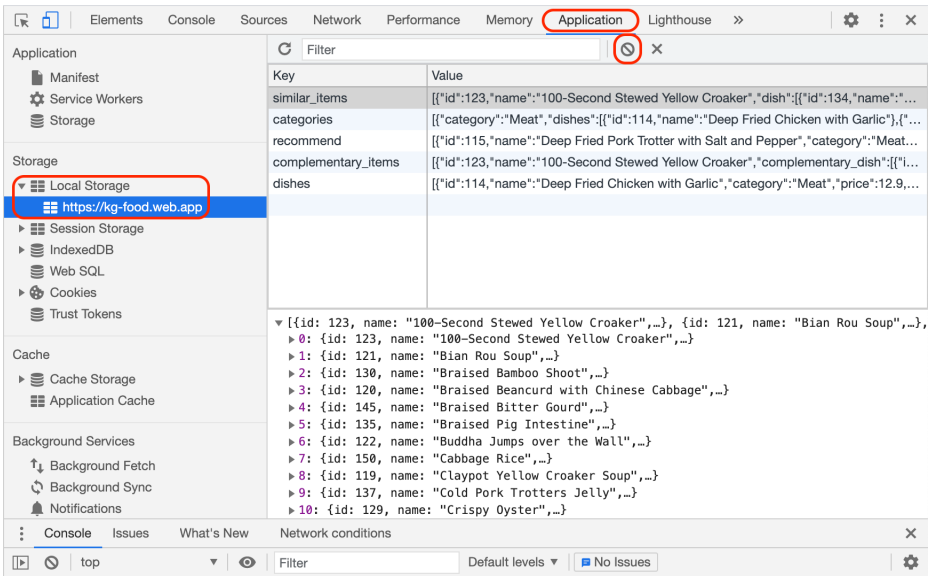


10. If the app cannot load, right-click on the page and open the ‘inspect’:

    1. **Unregister service worker**



2. **Clear local storage**



3. **Refresh the page**

We will provide more detailed instruction in the use case demo video.