Project: Restaurant Ordering and Pickup Management System (https://ritual.co Inspired)
Course: CMPT-315 Web Application Development
Instructor: Chen Liang MEng BASc

Introduction

In this project, your team is tasked with implementing a **Restaurant Ordering and Pickup Management System** inspired by https://ritual.co. Using Python/Django or JavaScript/Node.js, and React.JS to implement this system, your system will emulate the core functionalities of Ritual, facilitating easy ordering from local restaurants and efficient pickup. This project will let you demonstrate the skills you have learned throughout the course. The general idea behind the system is to allow users to place food orders, for restaurants to confirm orders, and assign a pickup time window. Your group of 4 will have 6 weeks to complete this project.

Getting Started

The Restaurant Ordering and Pickup Management System you are to develop should be able to be used by any restaurant in Canada. The required features and objectives are below.

Project Objectives:

- Create a **Python/Django** or **JavaScript/Node.js** Backend Rest API
- Create a **React** Frontend Application that communicates with the backend to perform the database operations.
- Design a customer user interface that allows the customers to perform various Food Order operations.
- Design a restaurant order interface that allows managers of the restaurant to conduct various administrative operations.

Required Features:

- **Customer Database:** Maintain a list of all registered customers.
- **Restaurant Database:** Maintain a list of all registered restaurants.
    - A minimum of three restaurants is required
- **Menu Management:** Each restaurant has its menu and inventory.
    - A minimum of 24 menu items across all restaurants
    - Allow restaurants to mark specific items as ‘sold-out’ and it will not appear on the customer ordering page.
- **Ordering System:** Customers can place orders at any restaurant.
    - Customers cannot place orders with any menu item that is ‘sold-out’.
    - Customers cannot place orders with menu items across multiple restaurants in 1 order.
- **Pickup Scheduling:** Customers can schedule pickups at their convenience.
    - If the customer is scheduling for immediate pickup, give the restaurant manager the option to add a predicted pickup time.
- **Order Management for Restaurants:** Outlets can manage, process, and update the status of orders.

        - Status: ‘ordered’ (this is when the customer first places the order)
            - Done by the customer
        - Status: ‘in-progress’ (this is when the restaurant has accepted the order)
            - Done by the restaurant manager
        - Status: ‘awaiting-pickup’ (this is when the order is ready for pickup)
            - Done by the restaurant manager
        - Status: ‘completed’ (this is when the customer has picked up the order)
            - Done by either the restaurant manager or the customer

Bonus features

- **Special Offers:** Implement special offers and discounts.
- **Loyalty Program:** Implement a loyalty program for frequent customers per restaurant.
- **Payment:** Implement a payment system via paypal to accept the payment for the order amount total.

Client Interface Objectives:

- **Menu Browsing:** Ability to view detailed menus from different restaurants.
    - Dynamic updates of menu items based on ‘sold-out’ status.
- **Order Placement:** Functionality for placing orders with specific menu items.
    - Available food items will have a related image depicting it (For example, a restaurant that has available french fries will have a picture of french fries)
- **Order History and Tracking:** Customers can view their past orders and track current orders.
    - See the order status requirement above.
    - See the pickup time requirement above.
- **Search and Filter:** Search for restaurant name or specific dish name.

Restaurant Manager Interface Objectives:

- **Menu Management:** Add, update, or remove menu items.
    - The ability to mark specific items as ‘sold-out’ and it will not appear on the customer ordering page.
- **Order Processing:** View and manage incoming orders and their statuses.
    - See the order status requirement above.
    - See the pickup time requirement above.
- **Analytics Dashboard:** View insights on sales, popular items and busiest pickup time by hour.
- **Custom Feature:** Develop a unique feature that enhances the restaurant pickup experience. (this function is up to your group to create)

Github

It is recommended to use a version control system for this project. With a group of 4, it can be difficult to work on the project simultaneously, and with a version control system such as github, we can alleviate this. If you need to become more familiar with github, please refer to the below tutorials/links. Usage of Github is not required for the project but is highly recommended.

- A useful Github cheat sheet https://education.github.com/git-cheat-sheet-education.pdf
- Another useful Github cheat sheet https://rogerdudler.github.io/git-guide/files/git_cheat_sheet.pdf

Database Architecture

You are welcome to use either a MySQL or MongoDB database. It is recommended to finalize the database before starting to work on the backend.

Grading

While working on the project, you will have two different milestones to complete and one final milestone demo where you will demo your final project.

Milestones

Milestone 1: March 7-8

- Database implemented
- The backend application can communicate with the database (create, read, update, delete)
- Demo during lab that your backend can communicate with the database with multiple different functionalities (adding a menu item to an order, placing an order, updating the status of an order are all good examples)

Milestone 2: March 21-22

- The React frontend should be able to interact with the backend to do a few operations.
- For example: placing an order, adding menu items to the order, scheduling pickup time, or updating the order status are all valid options.
- Must be able to perform more than two operations from the React frontend.
- Demo your React frontend interacting with the backend during lab

Milestone 3: April 4-5

- Demo for the Final Project