

Online Restaurant order and Delivery System

Fall 2020

In this system, we are about to develop an on-line restaurant order and delivery system so that the restaurant can provide menus of food, customers browse and order the food from the menu, delivery people of the restaurant deliver the food.

In this system, there are three groups of users:

1. **Restaurant:**
 - a) At least two chefs who independently decide the menus;
 - b) At least two delivery people who compete for food delivery
 - c) The manager/superuser who processes customer registrations, handles customer compliments and complaints, hires/fires/raises or cuts pay for chef(s) and delivery people
2. **Customers:**
 - a) Registered customers who can browse/search, order and vote (lowest 1 star to highest 5 stars) food delivered (on food and delivery quality/manners individually); can start/participate in a discussion topic on cooks/dishes/delivery people.
 - b) VIP customers who spent more than \$500 or placed 50 orders as registered customers, whichever comes first, in addition to the actions of registered customers, they will receive 10% discount of their ordinary orders, have access to specially developed dishes, and their complaints/compliments are counted twice as important as ordinary ones.
3. **Surfers:**

Browse the menus and ratings only, can apply to be the registered customers with fixed amount of deposit money and checked by the manager.

*****All of the functionality specified for the users above has been implemented. Details are provided under each required system feature below.*****

System features:

1. Provide a GUI, *not necessarily web-based*, with pictures to show the components and descriptions of each dish and price; each registered customer/VIP has a password to login, when they log in, based on the history of

their prior choices, different registered customer/VIP will have different top 3 listing dishes. For new customers or surfers, the top 3 most popular (ordered most) dishes and top 3 highest rated dishes are listed on the page.

- Menu can be browsed by scrolling down or using the search bar. JPG images are associated with each item.
- Surfers can browse the menus and ratings by selecting the "I'm Hungry" option on the home page.
- If they try to order something, they will be asked to sign in. If they don't have an account, they can choose to sign up/apply to be registered customers. The manager then approves the customer on the Manage Customers page.
- VIP customers who spent more than \$500 or placed 50 orders as registered customers, whichever comes first, in addition to the actions of registered customers, they will receive 10% discount of their ordinary orders, have access to chef specials, and their complaints/compliments are counted twice as important as ordinary ones.
 - All of this is done on the backend and VIP customers see the chef specials on their Menu.

2. The chef whose dishes received consistently low ratings or 3 complaints, or no order at all for 3 days, will be demoted (less salary), a chef demoted twice is fired. Conversely, a chef whose dishes received high ratings or 3 compliments, will be promoted (higher salary). One compliment can be used to cancel one complaint. The delivery people are handled the same way.

- Hire = manager approves the employee on Manage Employees page.
Fire/raise/cut pay is automated in our backend server based on employees performance (for example, if they didn't do a job in 3 days, they get demoted) or based on the reviews they receive.

3. A customer can choose to 1) eat the food in the restaurant, 2) pick up the dishes by self or 3) by delivery. For 1) s/he need to fix the available time and seating in the restraint; for case 1) and 2) s/he can only complain/compliment the chef.

- These options are visible to the customer in their Cart. Currently, for #1, we linked to a 3rd party scheduling tool to book tables. For #2, the customer can see previous + current orders at the bottom of the customer dashboard. Orders that are marked as delivery not needed will only have the chefID filled in, hence customers can only leave reviews for chefs. For #3, the order will be marked as delivery needed in the backend, and thus the chefID and deliveryID will be filled in.
- Every delivery person gets to see all of the pending deliveries on the Deliveries page. They compete for food delivery there. This applies to any new delivery people as well.

4. Customers can file complaints/compliments to chef of the food s/he purchased and deliver person who delivered the dish or other customers who didn't behave in the discussion forums. Delivery person can complain/compliment customers s/he delivered dishes, all are handled by the manager. The complained person has the right to dispute the complaint, the manager made the final call to dismiss the complaint or convert it to one formal warning and inform the impacted parties. Customers/delivery people whose complaints are decided without merit by the manager will receive one warning.

- A chat popup available on nearly every page is used to discuss cooks/dishes/delivery people.
- Complaints and compliments can be posted by clicking the "Write A Review" button on the Reviews Received page. Delivery people and customers can do this. Clicking the Dispute button makes the review go away, so the manager can take another look at it and decide whether to approve it. Everything else is handled by the backend.

5. Registered customers having 3 warnings are de-registered. VIPs having 2 warnings are put back to registered customers (with warnings cleared). The warnings should be displayed in the personalized page when the customers log in.

- All of this is handled by the backend. Warnings appear on the Dashboard page.
- Get 3 warnings, and you're automatically signed out and the user account is added to the blacklist.

6. If the price of the order is more expensive than the deposited money in the account, the order is frozen until the customer put more money in the account.

- The deposited money amount is loaded in frontend. Customers then have the option to deposit more money. If price or order is more expensive than deposited money in account, the frontend stores the order in cart until the user adds sufficient balance. Once there is sufficient balance, the order is removed off of the cart and is available for chefs to begin processing.

7. Customers who are kicked out of the system or choose to quit the system will be handled by the manager: clear the deposit and close the account.

- Handled by the manager who blacklists the customer

8. The chef is the one who put in the description and keywords for people to search and browse. The average ratings for each food/dish by customers are available for all.

- Chefs independently decide the menus:
 - Each chef is able to create menu items and upload jpg images of the food.
 - Average rating is calculated and is displayed on the menu.

9. The manager keeps a taboo list of words, any customer who used those taboo words will receive one warning automatically and the words are replaced by *, a message with more than 3 taboo words are blocked automatically.**

- npm bad words is allowing us to keep a list of bad words and check if a bad word was used in the chat.
 - 1+ bad word(s) in 1 message = +1 warning, replace word with asterisks
 - >3 bad words in 1 message = delete message
 - Refresh the page to see the change

10. Each team comes up with a creativity feature of the system to make it more exciting, which is worth 10% of overall score of the final project. Details that are not found in this requirement list are up to your team's call: you fill in the details to your own liking.

- Sendgrid integration - when a customer or employee is approved by manager, an automated email with a beautiful template is sent to their email to welcome them to the Foodtopia family.
- Recipe Image upload functionality - instead of just getting images from the internet or local storage of repo, chefs are able to upload any image they want while creating a recipe and that image will be saved in our database which is then returned any time a menu item loads
- Beautiful User Interface - We used Material UI to built the user interface. Foodtopia's UI reflects the warmth that food brings to us. It designed to feel welcoming to any type of user that comes the website.
- Well optimized, structured, and scalable database - We used MongoDB Atlas which is a real time cloud database which allows multiple users to use the system at once. We used we put a lot of effort in how we structured the database. It's designed to be highly scalable. For instance the requirement was to have only two chefs and two delivery people but our database can handle any number of users.
- Chat discussion - instead of just creating simple page where users just post their discussion, we put a lot of effort to build discussion forum in a way that makes

you feel more connected to other users on Foodtopia where you can discuss about menu items, chefs, delivery people, or anything that you desire.

- Microservice architecture - we used a distributed microservice architecture (frontend on 1 server, backend on 1 server, email automation on 3rd server, db in 4th server) in order to prevent overloading of server.
- MERN stack - we used MongoDB, Express, React, and NodeJS to build our app. This technology stack is widely used in the industry which makes our app meet industry standards