On-line 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;

   Created chef's admin page for chefs to change the menus items.

   b)   at least two delivery people who compete for food delivery

   our application still has bugs, so it doesn't register the items in the database, so the delivery people can't handle the food delivery.

   c)   the manager/superuser who process customer registrations, handles customer compliments and complaints, hire/fire/raise or cut pay for chef(s) and deliver people

   Created manager/superuser's admin page for chefs to handle customer compliments and complaints in the discussion forum.

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 a discussion topic on cooks/dishes/deliver people.

   Created discussion forum and home page, menus for registered customer to browse,order

   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.

We didn't implement this part.

3.   Surfers: who can browse the menus and ratings only, can apply to be the registered customers with fixed amount of deposit money and checked by the manager.

Restricted surfer's power by django database, they can browse the menus and rating only.can register with a fixed amount of deposit money and checked by the manager.

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.

Implemented Django, javascript,html, and css for the gui, 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.

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.

The chef can be fired, demoted or promoted manually by the admins, using the admin pages created, admins can see the complaints in the discussion forum, but the rating system is not working.

3.   A customer can choose to 1) eat the food in the restraint, 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.

A customer can add items to cart and purchase it , but our application still has bugs, so doesn't register the items in the database.

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.

Customers can file complaints to the chef using the discussion forum, but it's not registered in the database. Manager can decide what to do.

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.

Didn't have this part done.

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 order won't be frozen

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.

Managers can close the accounts using the admin page.

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.

Created chef's admin page for chefs to change the menus items.Have rating for chefs but not for food.

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.

Didn't have this part done.

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.

Our home page picture changes over time, and also added social media buttons on the home page.