Dig in! - Local Food Ordering

Scripting and Computing Environments (CSE505) Project

Overview

The Dig in! portal will allow customers to order food from a local restaurant. They will be able to choose from the menu of the cafeteria and order during the specified timings of the restaurant. The customer can also specify details on how the food is to be prepared. The cafeteria will be able to update their menu whenever necessary and also decline orders if they feel so. The method of payment will only be Cash on Delivery.

Technologies Allowed

HTML5, CSS (Bootstrap/Any framework) and JavaScript/jQuery/Angular2 for the Front-End. SQLite/MySQL as the Database of choice. Flask (Python) for the Server Side.

Expected Features

1) User Registration/Login
    - Create an account
    - Log in to the system.
    - Manage the account
2) List of Restaurants on homepage
    - Navigate the restaurant's menu.
    - Search a restaurant from the homepage
3) Ordering module
    - Add an item to their current order.
    - Review their current order.
    - Remove an item/remove all items from their current order.
    - Place an order.
    - Receive confirmation in the form of an order number.
    - View order placed.
4) Restaurant Backend Module
    - Retrieve new orders from the database.
    - Display all the orders on an user-friendly page

Desired Requirements

Goal 1. The system has a user-friendly user interface.

1. A particular user of the system has no difficulty in reading the text on the display.
2. The system is navigable through intuition.
3. Menu choices are presented in form of buttons, which contain text as well as little pictures illustrating the choice for better understanding.

Goal 2. System takes order from the customer as per his/her choice.

1. Scenario 2.1. User may want to make up his own order by selecting dishes.

Goal 3. The system calculates and displays the final bill based on the placed order.

1. System calculates final bill based on the quantity of the items multiplied by their unit price topped up by the applicable taxes if any.
2. User is given the option to either pay for the order or revise the order.

Goal 4. System handles the payment for the user-defined order.

1. User decides to pay cash and system asks user to enter cash in the slot.
2. The system prints out receipt containing a token number, details of the order, bill and the payment method with a terminal message (Thank you visit again or Store address).

Goal 5. System offers the store manager to check orders

1. Store manager decides to check all the pending orders.
2. Calculate total revenue from all the orders.

Bonus marks for any creative features. Use your imagination and ofcourse order online to understand the process :)