ARYAMAN MISHRA

19BCE1027

CAFETERIA ORDERING SYSTEM

LAB 3

1.Functional Requirements

Every component of the cafeteria ordering system provides a level of isolation between the end user and the database.It allows the employees(user) to interact with the ordering system through a user interface for a much appealing experience,particularly for the non-technical users working at ABC pvt Ltd.The system is designed to handle users taking any actions outside the limits of the legacy server.It is essential to enuumerate exactly which functions will be presented to users and how they are divided into components.

Users of the cafeteria ordering system, namely cafeteria customers, must be provided the following functionality:

1)Create an account(or created first-hand by the employers).

2)Manage their account.

3)Log in to the system.

4)Navigate the cafeteria’s menu.

5)Select an item from the menu.

6)Customize options for a selected item.

7)Add an item to their current order.

8)Review their current order.

9)Remove an item/remove all items from their current order.

10)Provide delivery and payment details.

11)Place an order.

12)Receive confirmation in the form of an order number.

As the goal of the system is making the process of ordering food more comfortable,the system is restricted to the maintenance of food and ensures that it reaches the employees in due time. With the exceptions of account creation and management, will be used every time a customer places an order.

The menu management system will be available only to cafeteria employees and will, as the name suggests, allow them to manage the menu that is displayed to users of the cafeteria ordering system. The functions afforded by the menu management system provide user with the ability to, using a graphical interface:

1)Add a new/update/delete vendor to/from the menu.

2)Add a new/update/delete food category to/from the menu.

3)Add a new/update/delete food item to/from the menu.

4)Add a new/update/delete option for a given food item.

5)Update price for a given food item.

6)Update default options for a given food item.

7)Update additional information (description, photo, etc.) for a given food item.

It is anticipated that the functionality provided by this component will be one of the first things noted by the cafeteria user, as they will have to go through it to configure their menu, etc. before beginning to actually take orders. Once everything is initially configured, however, this component will likely be the least used, as menu updates generally do not occur with great frequency.

Of the three components, the order retrieval system is functionally the simplest. Like the menu management system, it is designed to be used only by cafeteria employees(chef,manager,deliverer) and provides the following functions:

1)Retrieve new orders from the database.

2)Display the orders in an easily readable, graphical way.

3)Mark an order as having been processed and remove it from the list of active orders.

2.Non Functional Requirements

The non-functional requirements for the Cafeteria Ordering System are limited,given the simplicity of the system’s working and that it is a public domain non-commercial management system.

1)Security and Privacy

Account details will be protected in the legacy server.Personnel and Food files in the central host repository will be encrypted such that contents are not visible by any other means through the management system.This is to prevent users on the central host computer,including administrative users,cannot see private information.

2)Performance

The product will be based on local server.It will take an initial loading time.The performance would depend on hardware components of both parties-front-end and backend.Payment system will enable the secure environment mode on user’s devices.The system will have different entries in database for users/employees.

3)Licensing

None required.

4)Legal,Copyright and Other Notices

All rights reserved by ABC PVT Limited.

5)Applicable Standards

It will be as per industry standard.

3.Ambuigities

1)Number of employees using them at a time.

2)Unknown time regarding the session.

3)Uncertainity regarding whether person is ordering food items for self or for his group of friends while using a single account.

4.Inconsistencies

1)A customer's experience on one channel feels different and unconnected from their path via another channel.

2)Moving from one channel to another should be intuitive and everything from menus to buttons and other interfaces should be structured in the same manner.

5.Incompleteness

1)The system may still crash on huge load of employees signing in at the same time.

2)2 customers may order the same thing which has only 1 stock left and the order may get generated but one employees won’t be provided with the desired food.