Criterion A: Planning

Current scenario

My client runs a bakery and she records all the sales manually in a diary. This is a very cumbersome method for doing her work because she has to write the records for every cake that she has sold. Also, these records are not in a chronological order as they're simply placed one after the other. One day, my client was discussing the drawbacks of her current records holding system and told me that she wanted a software so that she could see everything she wanted at once instead of searching through a diary[1]. I thought that I could take this up for my Computer Science IA since it was within the scope of what I can do for my IA and also, the problem being faced by her is an ideal topic for the solution. Therefore, I decided to go up to my computer science teacher and ask her if I could take up the problem for my Computer Science IA. She agreed and I decided to go further with this topic and planned the first interview with my client, the baker.

Rationale for the proposed solution

The product will be able to allow my client, the owner of the business to maintain a complete track of her sales: daily and monthly along with a comparison of the sales of a particular month with the previous month's sales. The annual sales will be shown as a graph and statistics such as months with the maximum sale and months with the least sale. It would show her the stock status of the ingredients and notify with a popup if the stock is low. It would enable her to see the entire menu and make changes to it such as add, remove or modify items.

---

[1] Refer to Appendix A: Initial Interview

The final bill should for each of the items would contain:

- Name and address of the customer.

- The name of the item bought, quantity and delivery date.

- The total bill amount.

I have decided to use Java for my IA because of the following reasons:

- It is platform independent which means that my client can use the software on her phone, laptop or tablet seamlessly.

- The solution would be user friendly due to strong graphical user interfaces such as buttons, drop-down menus, etc.

- Java is object oriented which means that I can create a modular and easy to maintain software which can be tweaked in the future quite easily.

In order to store the information, I decided to use MySQL because of the following reasons:

- It does not require an internet connection

- Through the use of PhpMyAdmin, its strong GUI allows for easy maintenance of data in tabular format.

Word count:447

Success Criteria

   a. The user will be able to login using the allotted username and password and the program will redirect the user according to the user's role namely employee or admin.

   b. The admin will be able to see the following:

     I. The number of employees.

     II. The entire menu.

     III. Month wise sales records.

     IV. The orders for the day

     V. Stock

   c. The admin will be able to **add**:

     I. Employees

     II. Items to the menu

     III. Orders

     IV. Stock

   d. The admin will be able to **update** details for existing:

     I. Employees

     II. Items in the menu

     III. Orders

   e. The admin will be able **delete** Items from the menu

   f. Once the admin **delivers** an order, the program should keep track of the past orders.

   g. The admin will be able to search tables according to the following parameters:

   1. Employee table:

     I. Serial No.

     II. Name

     III. Username

     IV. Status

   2. Menu table:

     I. Serial No.

     II. Name of the item

     III. Price

     IV. Quantity

3. Past orders table:
     I. Serial No.
     II. Name of the customer
     III. Name of the item purchased
     IV. Price
     V. Quantity
     VI. Date range

4. Pending orders table:
     I. Serial No.
     II. Name of the customer
     III. Name of the item purchased
     IV. Price
     V. Quantity

h. The admin will be able to view a graph showing the monthly sales.

i. The admin will be able to print out a menu card and receipt for an order.

j. The admin will receive a notification if the stock is low.

k. The employee will be able to **see** but not **edit**:
     I. The orders for the day
     II. Items in the menu
     III. Stock