## **REQUIREMENTS**

### Cafeteria Management System:

**Canteen Management System project is a desktop application which is developed in C platform.**

### Research

**Ageing:**

* Mecca, known for being the Prophet Muhammad’s birthplace and Islam’s holiest city, played a key role in the history of the cafe. People believe that it is the location of the world’s very first coffee shop.
* The evolution and history of the cafe has had a significance in daily life of people around the world.
* The first café is said to have opened in 1550 in Constantinople, during the 17th century cafés opened in [Italy](https://www.britannica.com/place/Italy), [France](https://www.britannica.com/place/France), [Germany](https://www.britannica.com/place/Germany), and [England](https://www.britannica.com/place/England).
* Existing systems require
  + more man power
  + Paper work
  + Needs manual calculations

### System definition:

Cafeteria management system is specially designed for calculating total bill in a cafe and adding ordered item’s record. This system elaborates basic concept for storing and generating ordered item’s detail. In this system, staff can sign up as a system admin, He/she can have full access to the system for maintaining daily records.

The whole project is designed in ‘C’ language and different variables and strings have been used for the development of this project. It is easy to operate and understand by users. There is no any error and warning contents in the project. The design is so simple that user should not find it difficult to use and navigate.

### SWOT Analysis:

Strengths:

* It tracks all the information of cafeteria
* Increases the efficiency of managing the cafeteria
* Manage the information of the products
* Manage the sales records
* Editing, adding and updating of records is improved which results in proper resource management of cafeteria
* Integrate all the records
* Work becomes very speedy

Weakness:

* Since the system is developed in C, reports cannot be sent automatically to email or website
* Orders are taken in offline mode, online ordering is not developed

Opportunities:

* We can printer in future
* More advanced software for Management

Threats:

* Memory issues

### Requirements:

|  |  |
| --- | --- |
| **ID** | **DESCRIPTION** |
| H\_1 | Visual Studios or Code blocks |
| H\_2 | Required standard libraries |
| L\_1 | Design various blocks or functions |
| L\_2 | Create product details |
| L\_3 | Generate bills |
| L\_4 | Maintain sales records |
| L\_5 | Validation for user input |

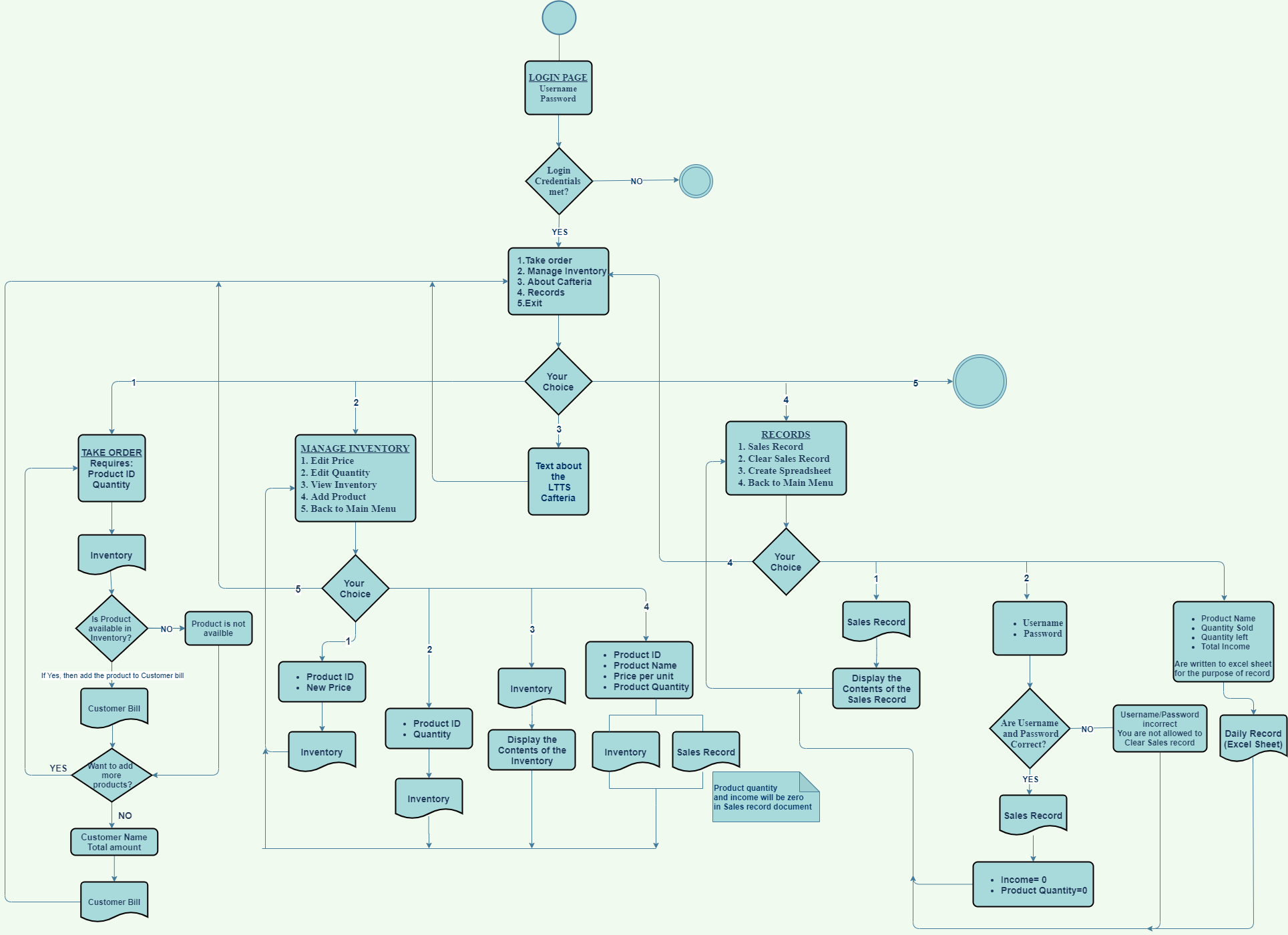
## **DESIGN**

Features:

* Login page
* The system helps the user to manage the inventory.
* The user can add new products, increase the price or quantity for a product.
* Taking order from the customer
* Generates bill for the customer with fields like customer name, name of the products ordered, quantity of each product and the total payable by the customer
* Each time a customer bill is generated, sales record gets updated
* The sales record includes the products that are sold and their quantity
* When the quantity of a product is zero then the system notifies the user that the product is no more available since there is no stock
* There is an excel sheet which records the stock in the inventory, sales for the day and the total income earned

Functions:

* Take Order
  + Add products as per the ID to the customer bill
  + Generates a bill (file) for the customer
* Management
  + View Inventory
  + Edit price
  + Edit quantity
  + Add product
  + Delete product
* Records
  + Sales record
  + Clear sales record
  + Update the Spreadsheet



**FIGURE**: Activity Diagram

**TEST PLANS**:

* Functional testing of the entire system as a whole
* Verification on login details
* Testing the reports with all its functionality
* Peak Load Test: It determines whether the system will handle the volume of activities that occur when the system is at the peak of its processing demand. Test the system by activating all terminals at the same time
* Storage Testing: It determines the capacity of the system to store transaction data on a disk or in other files
* Performance time Testing: It determines the length of time used by the system to process transaction data. This test is conducted prior to implementation to determine how long it takes to get a response to an inquiry, make a backup copy of a file, or send a transmission and get a response
* Recovery Testing: This testing determines the ability of user to recover data or re-start system after failure
* Procedure Testing: It determines the clarity of documentation on operation and uses of system by having users do exactly what manuals request.
* Human factors Testing: It determines how users will use the system when processing data or preparing reports