Software Requirement Analysis

1- Introduction:

1.1. The Purpose:

The purpose of this document is to build a Store Management system that can manage store's products, customers',employees' information, and can ease the processes of purchasing and store management.

1.2. Target Audience:

This project is a prototype for stores' management and purchasing system . It will be useful for both: customers seeking shopping and store's employees.

2- Overall Description

2.1. Product Perspective

It a store's online management and purchasing system that stores the following information:

- **Products Details**: It includes product's name and id, in addition to product's price and quantity inside the store.

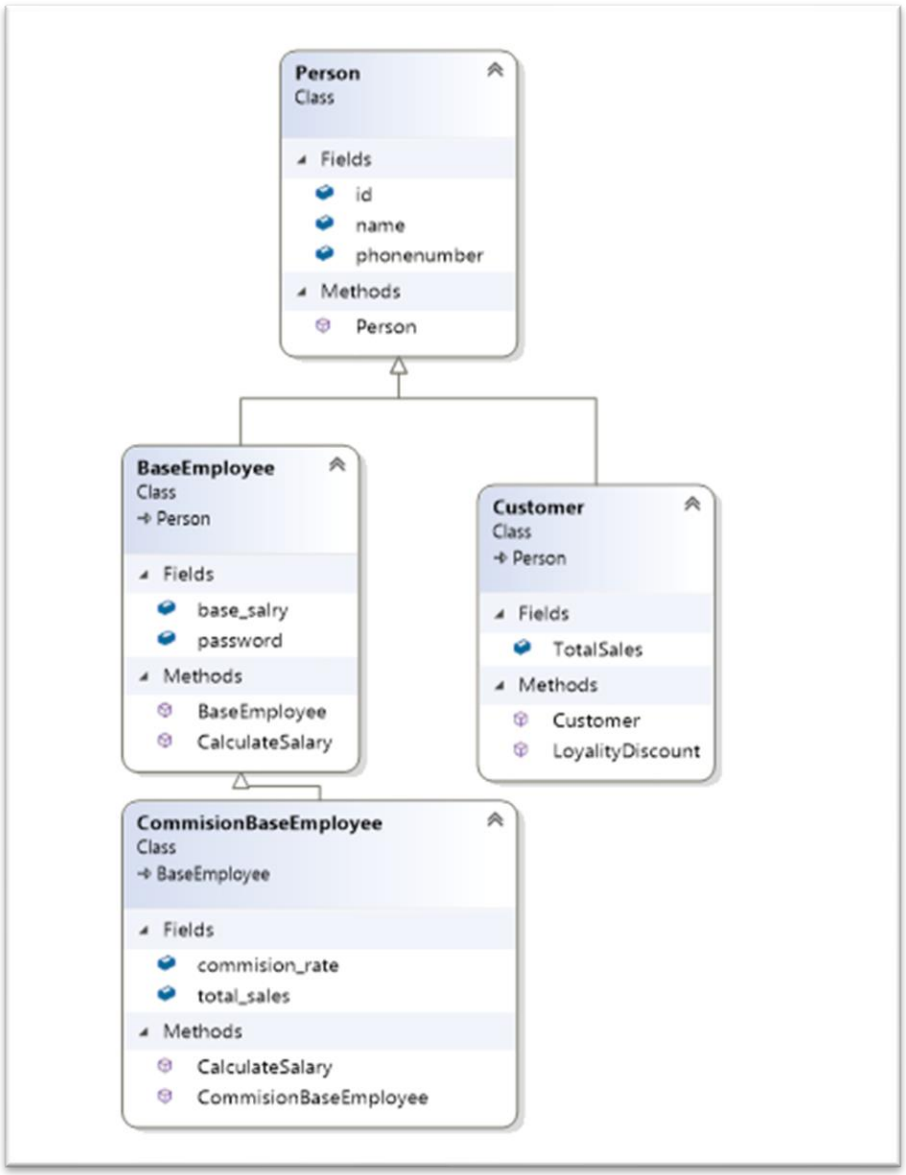
- **Customer Description**: This shows customer's personal information, such as customer's Name, and Phone Number.

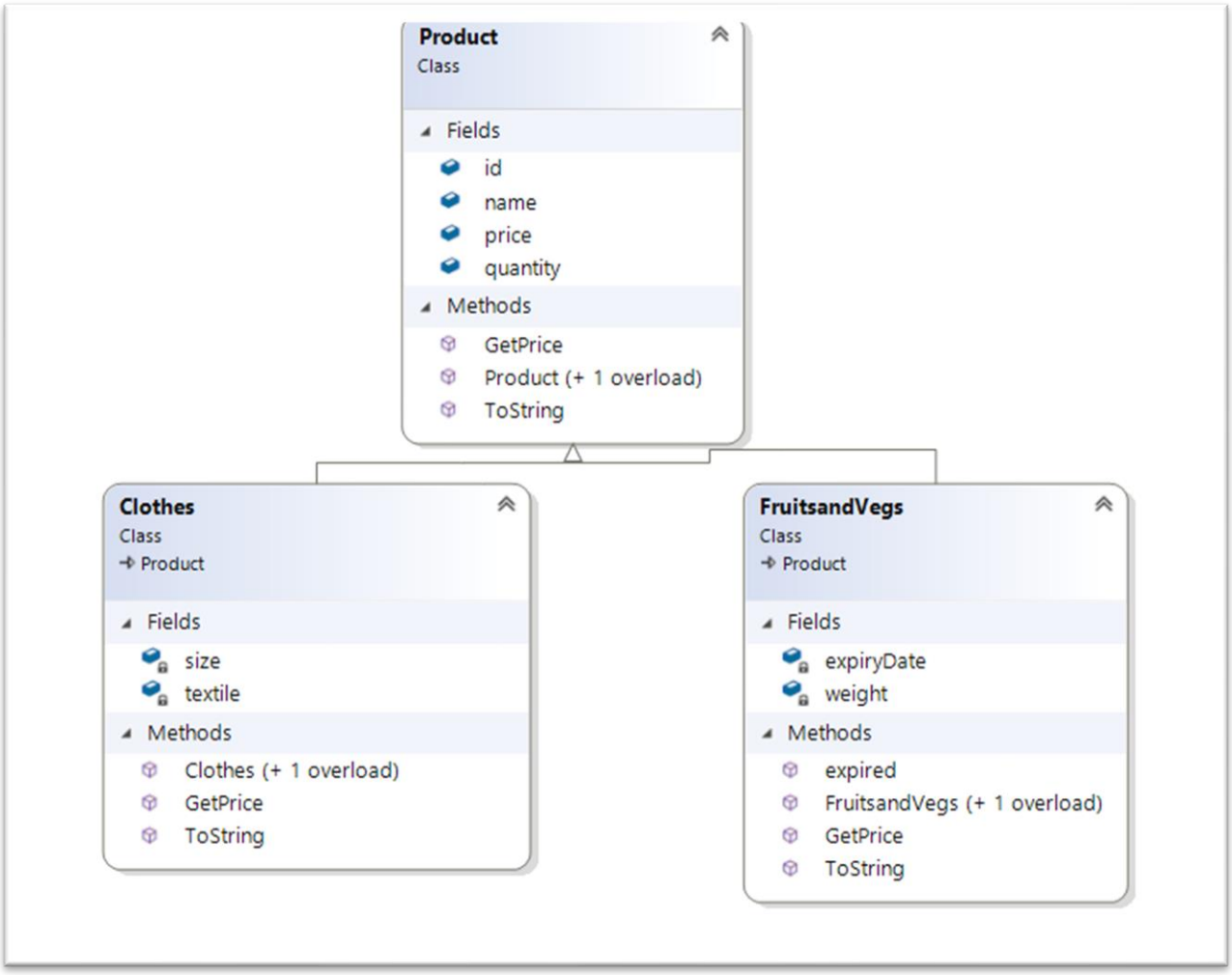
- **Employee Description: It shows the employee's personal information, in addition to his base/ commission-based salary.**

- **Bill description**: It includes the bill details: products purchased and the quantity and price for each product.

2.2. Product Features

The major features of Store's management and purchasing system are shown in the UML below:





BaseEmployee: -It is a class that refers to Employees who receive base salary.

-**CalculateSalary**(): Method that returns the base salary for each employee.

CommissionBaseEmployee:

-It is a class that refers to Employees who receive a commission-based salary.

-**total_sales**: refers to the total amount of products sold by the employee.

-**CalculateSalary**(): Method that returns the total salary for commission- based employee.

Customer:

- It is a class that refers to the customers seeking the Market for purchasing.
- **TotalSales:** refers to the total purchases of the customer.
- **LoyalityDiscount():** refers to the amount of discount the customer receives based on his total purchased products.

Products:

-it is a class that refers to the products available at the store.

- **GetPrice():** method that returns the price for each product.

FruitsandVegs:

- It is a class that refers to the products in the Fruits and vegetables section.
- expiryDate: refers to the date of expiry for each product.
- Weight: indicates the weight purchased from each item.
- **isexpired():** it returns whether the item is expired or not based on its expiry date.
- **GetPrice():** it returns the price of the purchased item based on its weight and the price/kilogram.

Clothes:

- it refers to the available products in the Clothing section.
- size: resembles the size of each clothing item.
- Textile: refers to the material from which the item is made.
- **GetPrice():** method that returns the price of each clothing item based on its size.

2.3. User Characteristics

Users of the system can retrieve every product's information. The system will support types of Users: Employee, and the Owner.

The Employee is able to do the following functions:

- Add new customers.
- Enter and confirm products for billing.

While the Owner can do the following functions:

- Adding new products to the system.
- Add/Remove employees.
- Update product details.
- Update employee's information.
- Add new customers.
- Enter and confirm products for billing.

2.4. Operating Environment

- client/ server system.
- Operating system: Windows.
- Database: SQL database.
- platform: C#.net

3- System Features

- **Description:**

The store's management and purchasing system maintains information on the available products as well as the customers, and Employees. Its main objective managing the products available and Employees' information in addition to providing customers with an easy and comfortable shopping experience.

- **Functional Requirements**

-Distributed Data base.

-Client/server System: it refers to the architecture or division of responsibilities , as the client is the application(front-end) and the server is the DBMS (back-end).

4- External Interface Requirements:

4.1. User Interfaces:

-Front-end software: C#.net version.

-Back-end software: SQL .

4.2. Hardware interfaces:

-Windows.

4.3. Software interfaces:

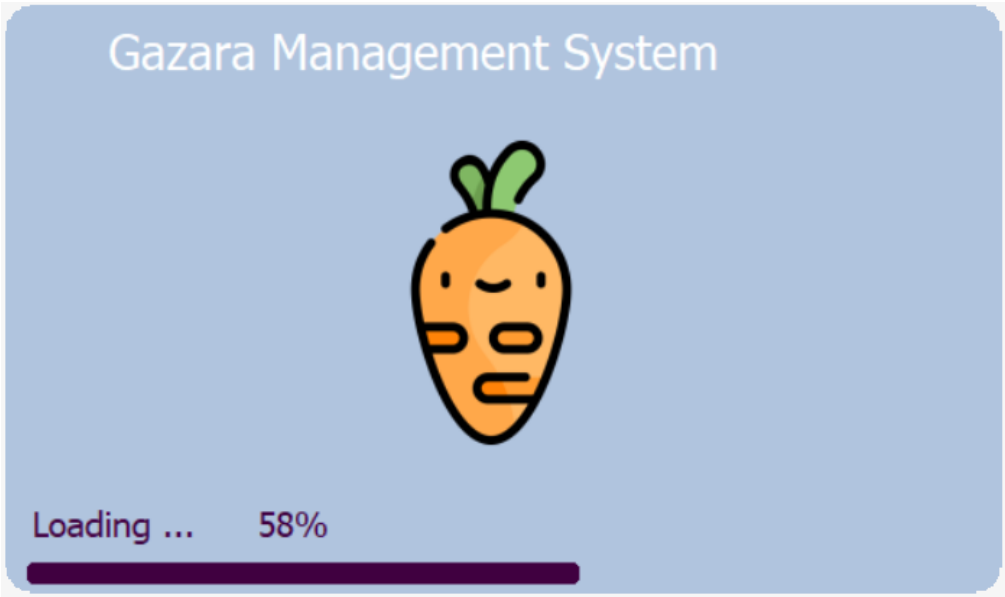
-Operating system: windows was chosen for its support and user friendliness.

- C#.net: This language has been chosen for its interactive support.

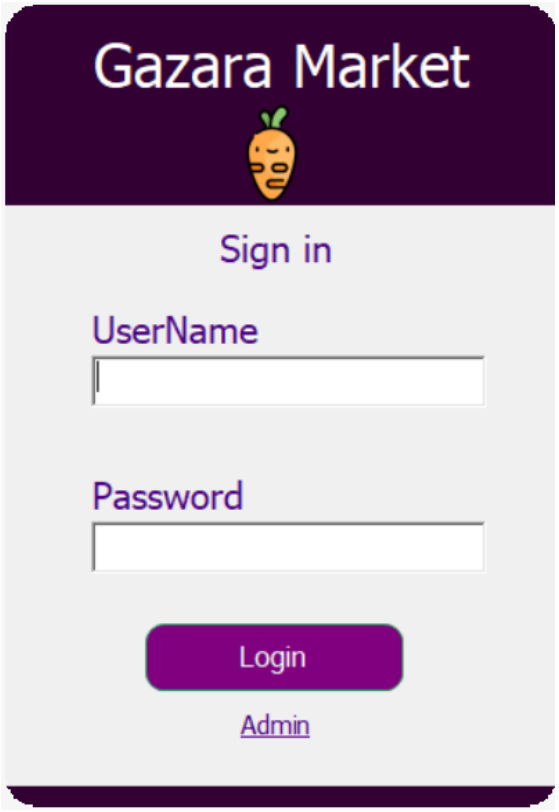
- Database: SQL database has been used to store products and customer details.

Interface Forms:

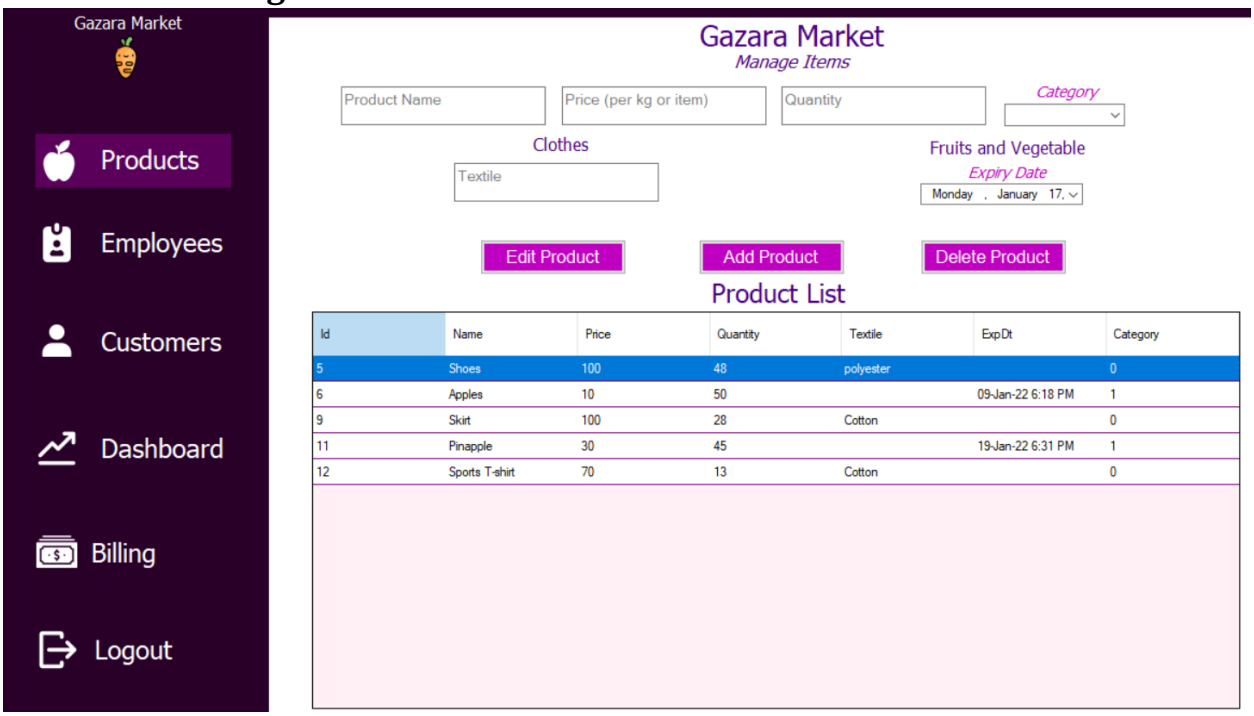
- **Loading Form:**



- **Login Form:**
  It views the login page to the system.

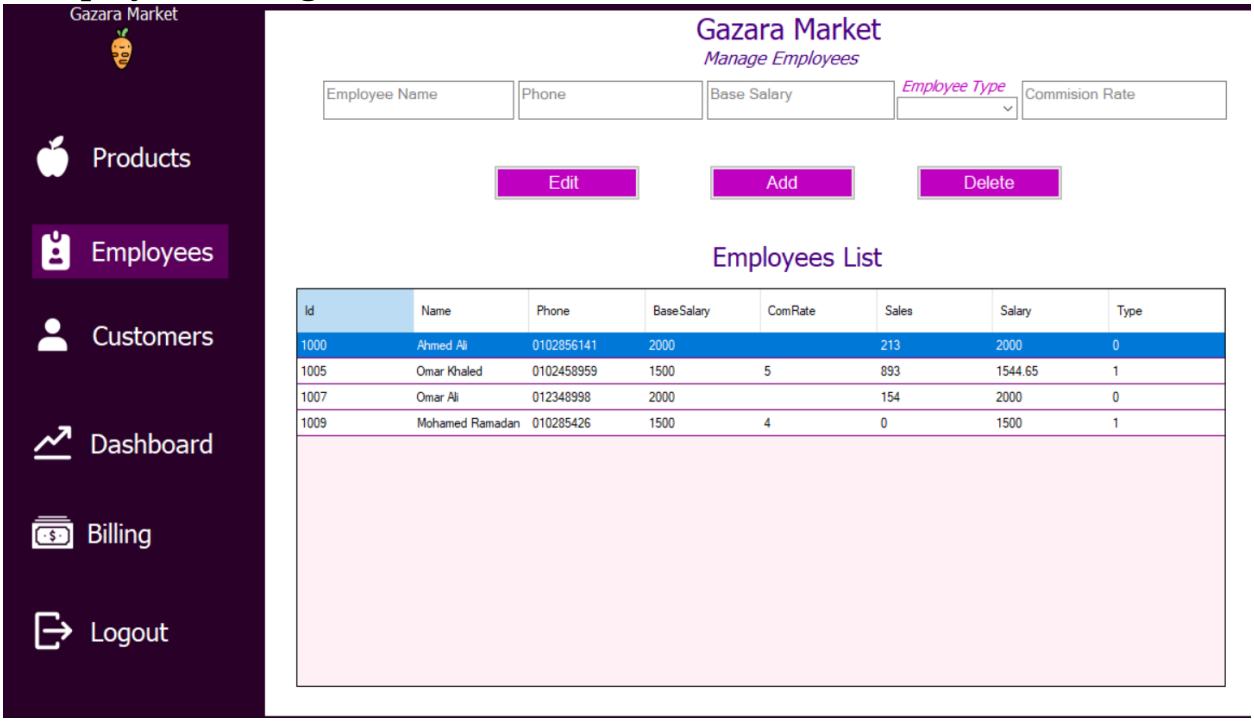


- Products management form:



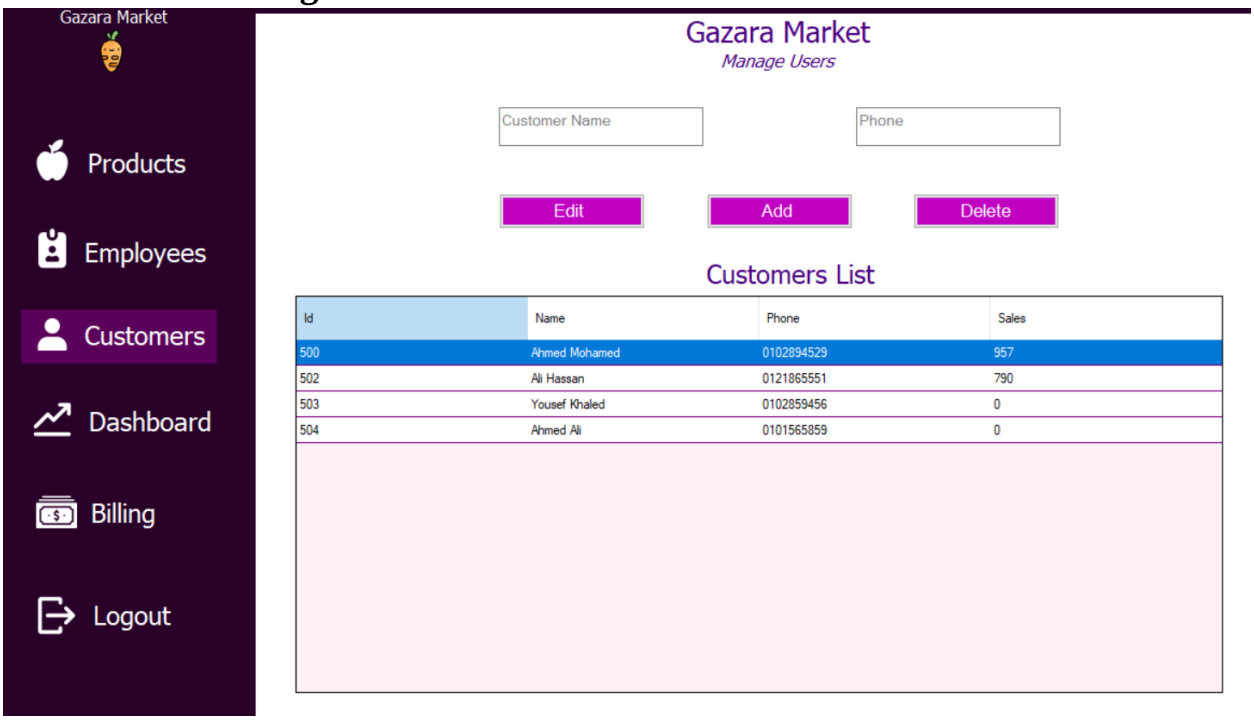
- It views the details of available products at the store.

- Employees Management form:



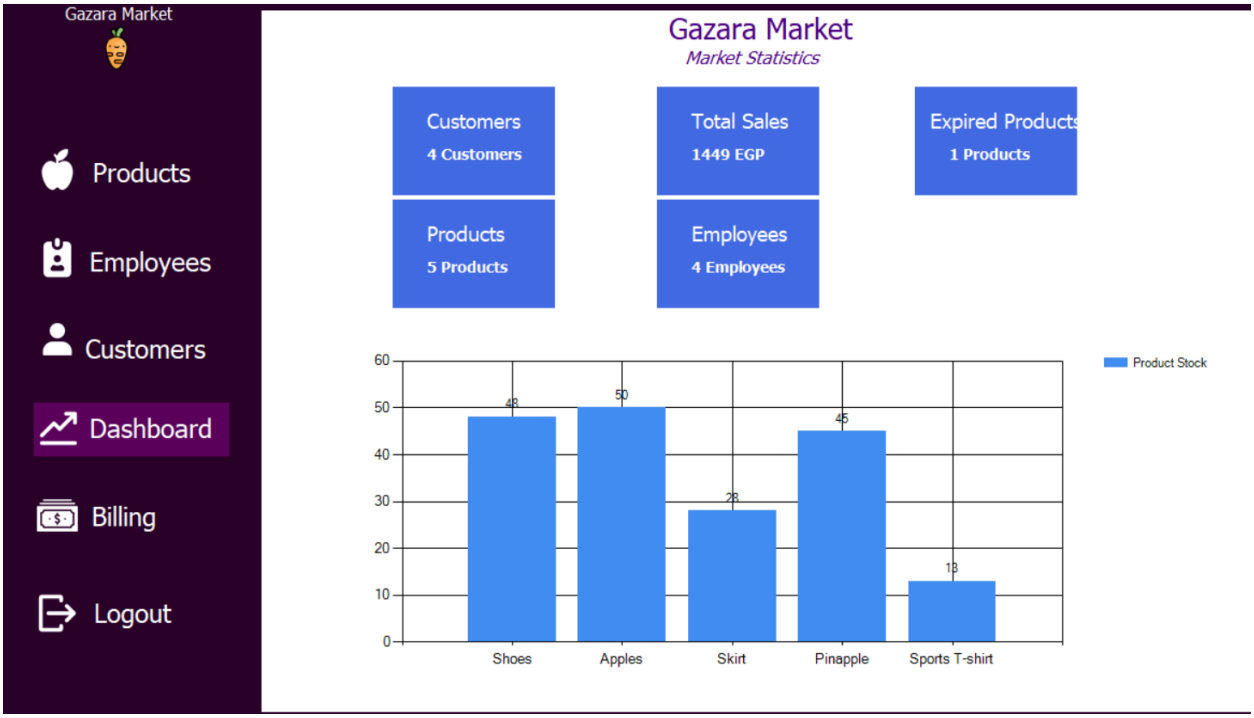
-It view the Employees personal information and base salaries.

- Customers Management form:



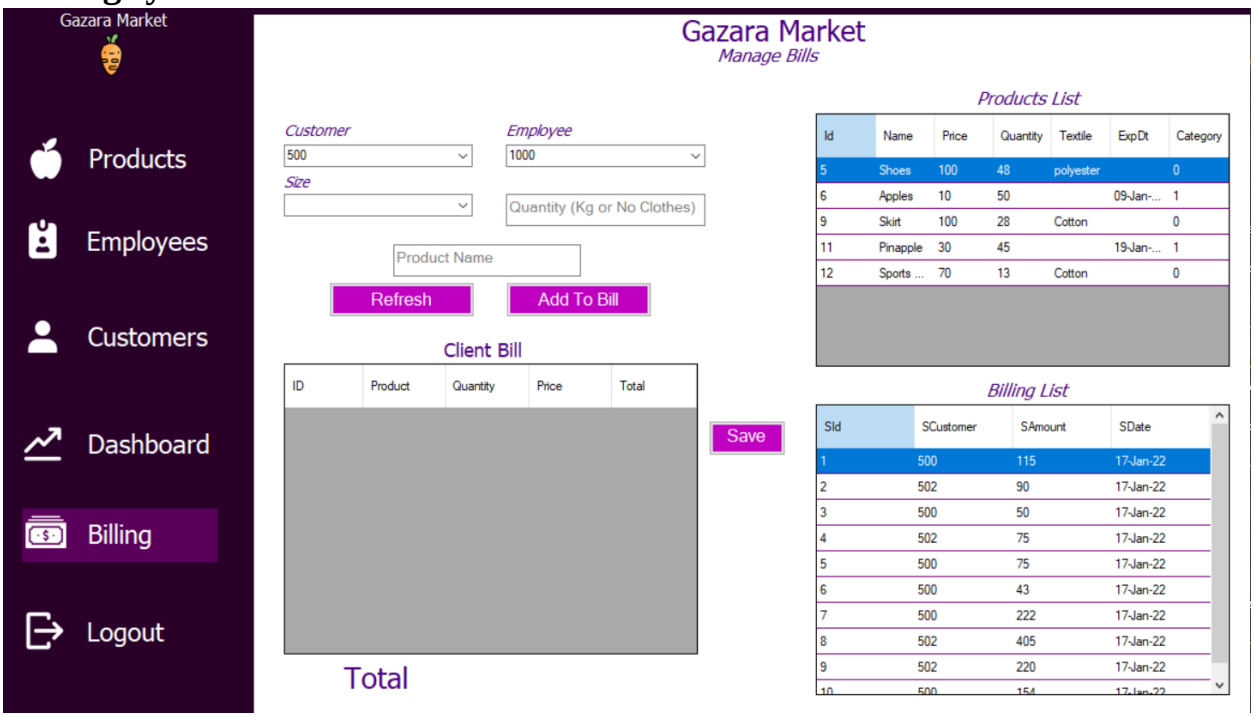
-It views the data of customer at the store.

-Store Dashboard form:



-It shows the summary of store's data: products, customers, and employees.

-Billing system form:



-It views the billing section of the system in addition to the invoice details.