



**Cairo University**  
**Faculty of Graduate Studies for Statistical**  
**Research**



**Easy Invoice Application (Group 130)**

**Supervisor**

**Dr. Tarek Ali**

**Team Members:**

|                           |           |
|---------------------------|-----------|
| Ahmed Gamal Sallam        | 202201314 |
| Kahlawy Saber Fathy       | 202201347 |
| Waled Ibrahim Mohey Eldin | 202202040 |
| Ahmed Gomaa Shazly        | 202201069 |
| Mohamed Mohamed Selim     | 202201271 |

**Problem & Idea (Lecture 1 page 4):**

With Easy invoicing application, Tax and Customs Authority (ZATCA) mandating e-invoicing for taxpayers in the Kingdom of Saudi Arabia effective from December 4, 2021, you will need software that helps you stay compliant with the new rules. is e-invoicing compliant and designed to adapt to GAZT regulations.

The objective of this Software Requirements Specification (SRS) document is to define the necessary features and requirements for the development of a billing system with integrated e-invoicing capabilities. The system aims to simplify and automate the billing process by providing a centralized platform for managing all invoicing activities. The e-invoicing integration will allow for faster invoice creation and delivery, reducing the likelihood of errors and improving overall efficiency.

The billing system will include features such as automatic payment reminders, invoice tracking, and customizable invoice templates. The system will enable users to generate invoices quickly and easily, customize invoices according to their preferences, and manage invoices efficiently. Additionally, the system will allow for the management of payments and provide users with insights into their billing and payment history.

### **Solution depended on MVP (skate) (Lecture 1)**

A simple application as a point of sale for itinerant sellers that offers the following solutions:

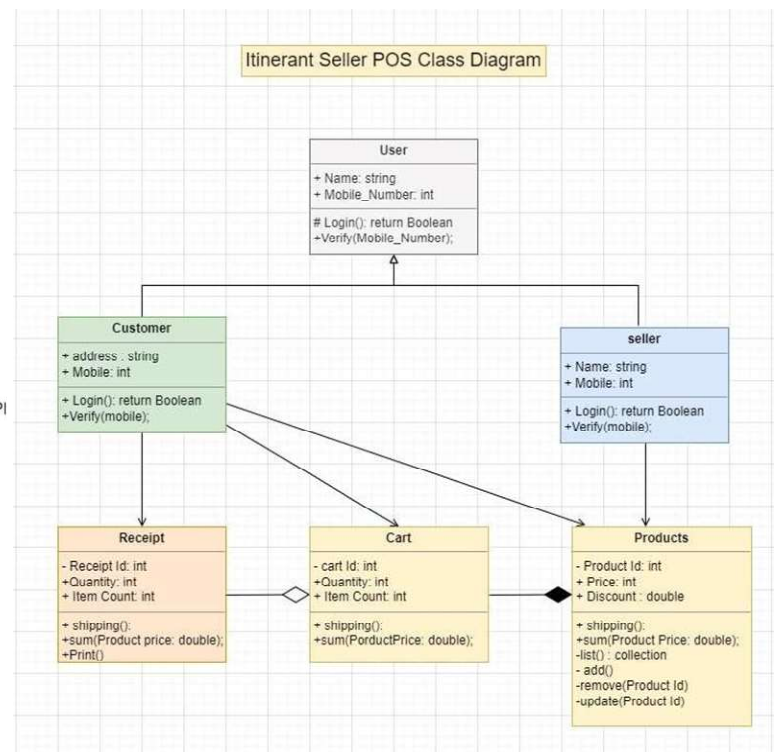
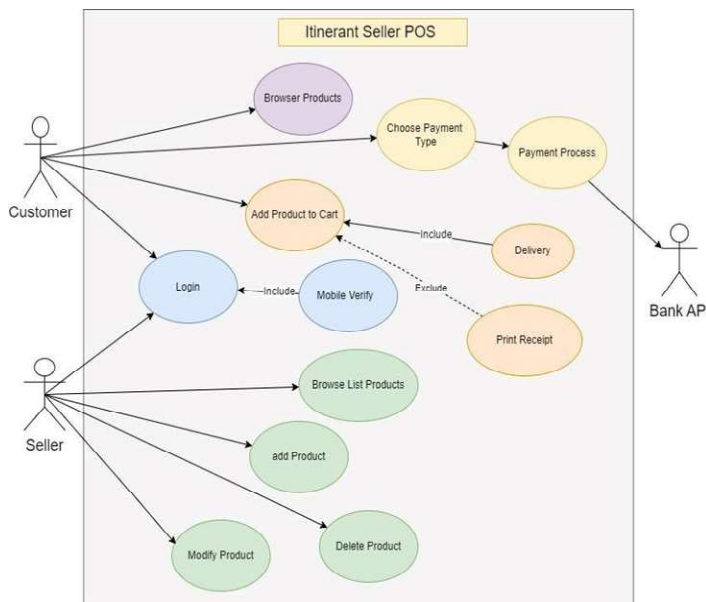
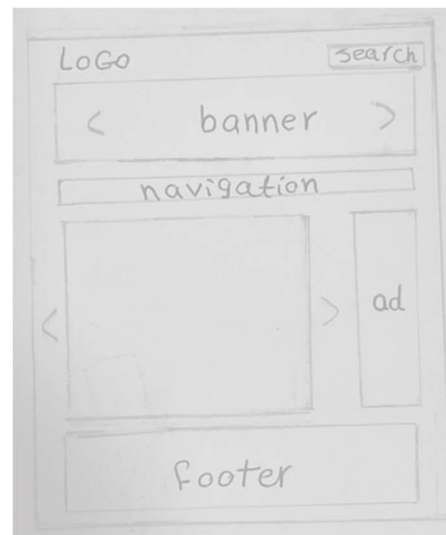
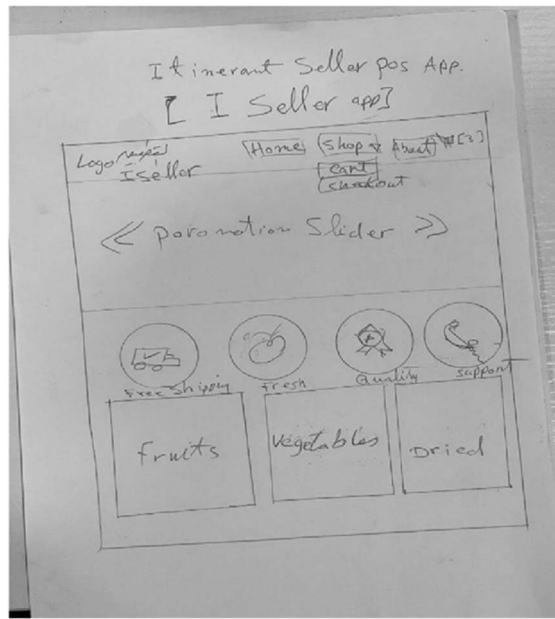
- Easy use app for various mobile devices, whether they are mobile phones, tablets, etc.
- Energy Source: You can use the mobile and a power bank only to use the application.
- Selling Online: add the feature for selling online
- Points of sale: where it makes accounting operations and issues invoices to the customer.



### **Kick of meeting (Lecture 3 page 4)**

We scheduling a meeting through Microsoft Teams to discuss the problem and find a solution to it, define the stockholders of the project, the roles, and responsibilities of the team members, and set a timeline for the implementation of the project, and the meetings and documented on hard pager.

### Sketch (Prototype): (Lecture 3 page 22)



### Technologies (Lecture 5 start from page 8)

- Frontend Technologies and Language: HTML, CSS, JavaScript.
- Backend Technologies: Asp.net core
- Backend Language: C#

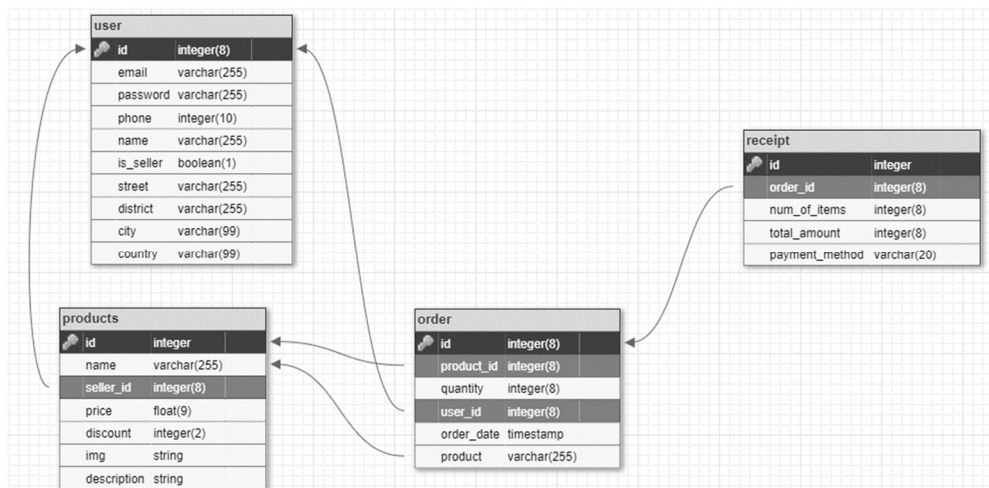
```

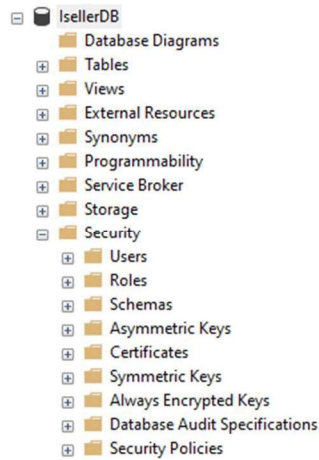
class cart {
private:
    int productId;
    string productName;
    double price;
    double discount;
public:
    void customer();
    void administrator();
    void receipt();
    void items();
    void list();
    void add();
    void edit();
    void rem();
};

for (int i = 0; i < c; i++) {
    data.open("Pdata.txt", ios::in);
    data >> productId >> productName >> price >> discount;
    while (!data.eof())
    {
        if (productId == arrId[i]) {
            amount = price * arrQty[i];
            dis = (amount * discount / 100);
            total = amount - dis;
            cout << "\n" << productId << "\t\t" << productName
        }
        data >> productId >> productName >> price >> discount;
    }
    data.close();
}

```

- Microsoft SQL Server





```
CREATE TABLE [products] (
    id integer NOT NULL,
    name varchar(255) NOT NULL,
    seller_id integer(8) NOT NULL,
    price float(9) NOT NULL,
    discount integer(2) NOT NULL,
    img string NOT NULL,
    description string NOT NULL,
    CONSTRAINT [PK_PRODUCTS] PRIMARY KEY CLUSTERED
    (
        [id] ASC
    ) WITH (IGNORE_DUP_KEY = OFF)
```

### Personas (Lecture 3 page 18)

- Customers
- Sellers

### Methodologies (Lecture 2 page 17)

- We use Agile – Scrum methodology

### Features of Tool I Seller App (MVP) (Lecture 3 page 12)

A) Use Cases:

| <u>ID</u> | <u>Feature</u>   |
|-----------|--|
| 1         | As a guest, I can create an account, manage my personal information and access my order history                  |
| 2         | As a member, I can manage my personal information and access my order history                                    |
| 3         | As a member, I can add products to my cart, view my cart contents, and modify quantities or remove items         |
| 4         | As a member, I can leave feedback on products and services, and view ratings and reviews.                        |
| 5         | As a member, I can access customer service resources, such as a help center, FAQs, and support tickets.          |
| 6         | As a member, I can create and manage promotions, discounts, and coupon codes.                                    |
| 7         | As admin, I can provides insight into key performance indicators, such as sales, traffic, and customer behavior. |





# Cairo University

## Faculty of Graduate Studies for Statistical Research



|    |  |
|----|--|
| 8  | As admin, I can ensures that the online shop meets industry standards for security, privacy, and compliance with regulations such as GDPR and PCI DSS. |
| 9  | As admin, I can guides members through the process of selecting a shipping method, entering payment information, and confirming their order            |
| 10 | As admin, I can provides tools for managing orders, including tracking, cancellations, refunds, and returns.   |

### **Test Cases (Lecture 6 Page 17)**

- 1- Validate that the seller should have an account to login
  - 2- Validate that the client should have an account to login
  - 3- Validate that user can add his location
  - 4- Validate that system saves the user location
  - 5- Validate that System has a list of products
  - 6- Validate that seller can update the available product per day
  - 7- Validate that seller can update the products price
  - 8- Validate that seller receive the products status after selling
  - 9- Validate that seller receive a calculation on purchased products
  - 10- Validate that seller receive a calculation of his profit
  - 11- Validate that seller receive a calculation of his selling per month
  - 12- Validate that seller can view the review of his products
  - 13- Validate that System display the available products per day in the identified location with their price
  - 14- Validate that client account can select from the available products
- Per day & location
- 15- Validate that client order the selected products the identified location
  - 16- Validate that client receive an invoice of purchased products
  - 17- Validate that client can add review on his order.
  - 18- Validate that Admin can edit System user types
  - 19- Validate that Admin add System products



# **Cairo University**

## **Faculty of Graduate Studies for Statistical Research**



- 20- Validate that Admin Update System products
- 21- Validate that Admin determine products price limit
- 22- Validate that Admin can view products reviews
- 23- Validate that Admin can add review on seller's accounts



# Cairo University

## Faculty of Graduate Studies for Statistical Research



### References

1. The official ASP.NET Core documentation: <https://docs.microsoft.com/en-us/aspnet/core/>
2. The official ASP.NET Core GitHub repository: <https://github.com/aspnet/AspNetCore>
3. "Pro ASP.NET Core 3" by Adam Freeman and Andrew Lock:  
<https://www.apress.com/us/book/9781484255078>
4. "Building Web Applications with ASP.NET Core and MongoDB" by Rajeev Sakhuja:  
<https://www.packtpub.com/application-development/building-web-applications-aspnet-core-and-mongodb>
5. "Learn ASP.NET Core 3.1" by Kudvenkat:  
[https://www.youtube.com/playlist?list=PL6n9fhu94yhVkdruSLaQsfERmL\\_Jh4XmU](https://www.youtube.com/playlist?list=PL6n9fhu94yhVkdruSLaQsfERmL_Jh4XmU)
6. ASP.NET Core Community on Stack Overflow:  
<https://stackoverflow.com/questions/tagged/asp.net-core>