Final Scientific/Technical Report Template

HTU Store (POS)

FULL Stack-php Capstone project

Instructor:

Eng. Bishara Estephan

Course:

Full Stack Web Development-php

Made by: Hamza AbuQwaider

9991042474.ups@htu.edu.jo

Training period:

2-Oct-2022 to 2-Jan-2023



|  |  |  |
| --- | --- | --- |
| **number** | **Subject name** | **Page number** |
| 1 | **Chapter One: Introduction** | |
| 1.1 | Problem | 6 |
| 1.2 | Objectives | 6 |
| 1.3 | Tools and software used | 7 |
| 2 | **Chapter Two: System Life Cycle** | |
| 2.1 | System Analysis Goals | 9 |
| 2.2 | system design | 9 |
| 2.2.1 | System Participants | 10 |
| 2.2.1.1 | The primary webmaster | 10 |
| 2.2.1.2 | The Seller | 10 |
| 2.2.1.3 | The Procurement | 10 |
| 2.2.1.4 | The Accountant | 10 |
| 2.2.2 | Data Analysis | 11 |
| 2.2.3 | Entity relationship diagram | 11 |

|  |  |  |
| --- | --- | --- |
| 2.1 | System implementation | 13 |
| 2.2 | Integration & Testing | 21 |
| 2.3 | maintenance | 22 |
| 3 | **Chapter 3: Discussion & Conclusion** | |
| 3.1 | Discussion | 22 |
| 3.2 | Conclusion | 22 |
| 3.3 | Recommendation & Reference | 23 |

**List Of Tables:**

|  |  |  |
| --- | --- | --- |
| **Table Number** | **Description** | **Page** |
| 1 | Users | 19 |
| 2 | Transactions | 19 |
| 3 | Items | 19 |
| 4 | User Transaction | 20 |

**List Of Figures:**

|  |  |  |
| --- | --- | --- |
| **Figure Number** | **Description** | **Page** |
| 1 | System Lifecycle | 9 |
| 7 | Logging in to the website | 14 |
| 8 | Home Page | 15 |
| 9 | Selling Dashboard | 15 |
| 10 | Edit transaction | 16 |
| 11 | Items Page | 16 |
| 12 | Edit Item | 17 |
| 13 | Create Item | 17 |
| 14 | Accounts Page | 18 |
| 15 | Transaction Page | 19 |
| 16 | Transaction Edit Page | 19 |
| 17 | Create User Page | 20 |

|  |  |  |
| --- | --- | --- |
| 18 | Information User Page | 20 |
| 19 | Edit User Page | 21 |

1. Introduction

Point of Sale is an online deal the executives programming venture that serves the usefulness to Admin, sellers, accountant, and procurement. The system allows users to use according to their permissions given by the admin. This application Online and the website can be opened from anywhere in the world.

# Problem:

# HTU Store uses the manual method to store transactions and stock, this method is inefficient because it may increase the incidence of mistakes, takes a long time and consumes effort.

# Objectives

* Informative dashboard, a simple dashboard will contain the following information: Total sales, Total transactions, Total items number, Top five expensive items to buy, Total users.
* Selling dashboard: this dashboard contains selling form with item name and item

quantity. The seller can’t sell items that are out of stock. There should be a list in the

same page for all the transactions that has been made today by the current logged in

user. The seller can edit or delete each transaction on this list directly using action

buttons.

* Stock management page should list all items and provide all CRUD(Create, Read, Update, Delete) functionalities (through subpages) for each item. Each item should have cost, selling price, stock available quantity, created at, and updated at fields.
* Transactions management page should list all transactions and provide only Read,

Update, and Delete functionalities (through subpages) for each transaction. Each

transaction should have item id, quantity, total, created at, updated at.

* Users’ management page should list all users and provide all CRUD functionalities

(through subpages) for each user. Each user should have display name, username,

password, email, role, created on, and updated on.

* Add profile page for each user contains the user image and the user should be

able to update his/her information and image.

# Tools and software used

Technical Feasibility Our system is a web-based application and the main technologies and tools that are associated with this project are:

1. For Front-end we will use:
   * HTML (Hyper Text Markup Language)
   * CSS(Cascading Style Sheet)
   * Bootstrap
   * JavaScript and jQuery 2- For Back-end we will use:
   * PHP (Hypertext Preprocessor)
   * MySQL (Structured Query language)
   * Xampp Server
   * Ajax and Json
   * Rest APIs

System life cycle

Each system has a life cycle (System Development Life Cycle), through which the system is built and developed based on sequential stages, as shown in Figure 1:



Figure 1: System Lifecycle (SDLC)

# System Analysis Goals:

By performing a system analysis, we were able to accomplish several objectives, including:

* + - Identifying and understanding the current state of the system.
    - Determining the requirements for a new or updated system.
    - Identifying potential solutions and alternatives.
    - Evaluating the feasibility and cost-effectiveness of potential solutions.
    - Testing and evaluating the new or updated system.

# system design:

Identify the requirements: Based on the problem definition and analysis of the current system, the next step is to identify the specific requirements of the new system. These requirements may include functional requirements (what the system should do), performance requirements (how fast the system should operate), and non-functional requirements (such as security, reliability, and usability).

Design the new system: Once the requirements have been identified, the next step is to design the new system. This may involve creating a high-level design, which outlines the overall architecture of the system, and a detailed design, which specifies the individual components and their interactions.

Implement the system: The final step in the system design process is to implement the new system. This may involve writing code, building hardware components, and testing the system to ensure that it meets the requirements.

Overall, the goal of the system design process is to create a system that is efficient, effective, and meets the needs of the user.

# System Participants:

The system consists of four main parties: 1- Admin

1. Seller
2. Procurement
3. Accountant

# Admin:

The administrator has the authority to control all of the system:

* + - * 1. Modify and delete transactions from the account page.
        2. Adding, modifying and deleting system users.
        3. Adding, modifying and deleting items.
        4. Adding, modifying and deleting transactions.

# The Seller

The Seller has total authority over the Selling dashboard:

* + - * 1. Adding, modifying and deleting transactions.

# The Procurement

The Stocks dashboard is completely under the control of Procurement.

1- Adding, modifying and deleting items

# The Accountant

The Transaction Dashboard is completely under the control of Accountant.

* + - * 1. Modify and delete transactions from the account page.

# Data Analysis:

In a point-of-sale (POS) system, data analysis can be used to track sales, identify trends and patterns, and optimize inventory management. POS systems often generate a large amount of data, including information about customer purchases, product details, and sales staff performance. By analyzing this data, businesses can make informed decisions about marketing, pricing, and other key aspects of their operations.

* Create a database schema: which includes tables for storing information about items,

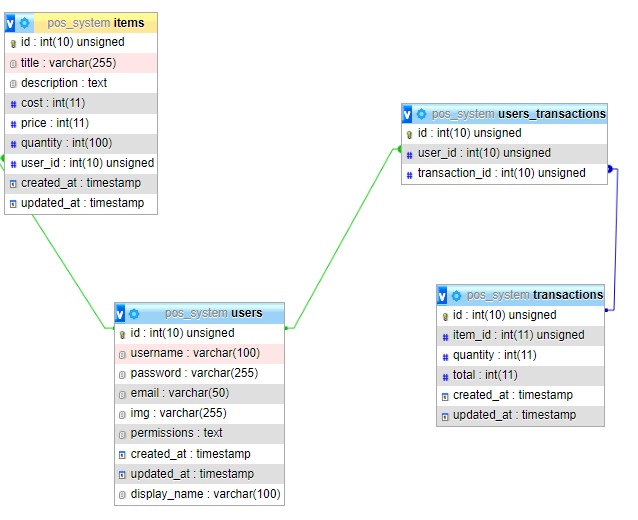
users, transactions, and any user-transactions.

* Create the models: The models in an MVC application represent the application's data and business logic. You could create models for products, customers, and transactions for a POS system.
* Plan the POS system's functionality: Determine the features and capabilities that you want the POS system to have. This could include inventory management, transaction processing, report generation, and other tasks.

# Entity relationship diagram

The entity relationship diagram is depicted in Figure 6.

Figure 6: Entity Relationship Diagram



Database and system page design:

The system includes the following primary tables:

Table 1: The Table Users

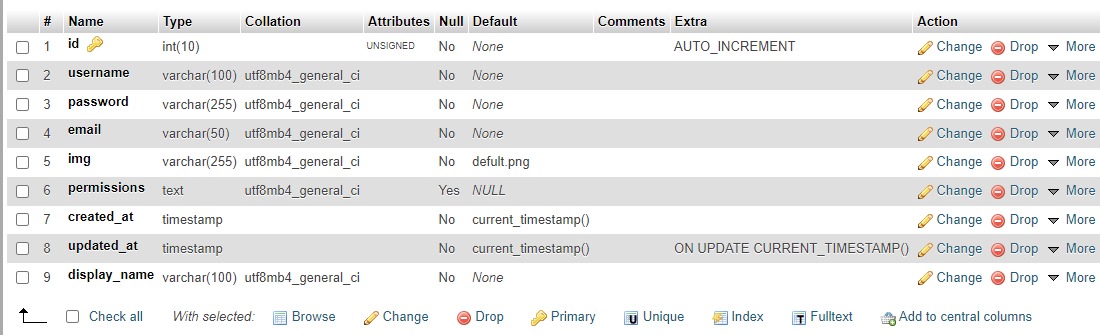


Table 2: The Table Transactions

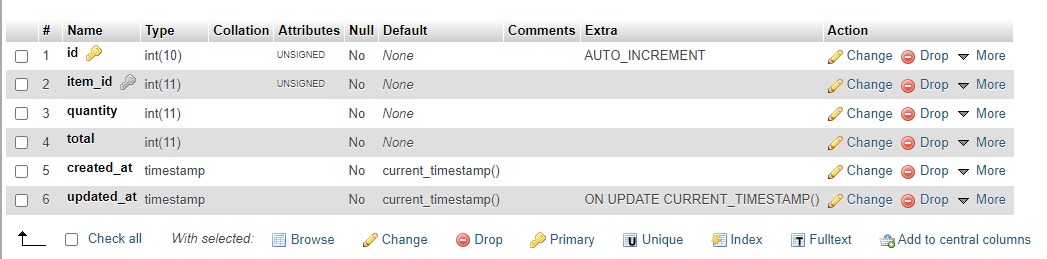


Table 3: The Table Items

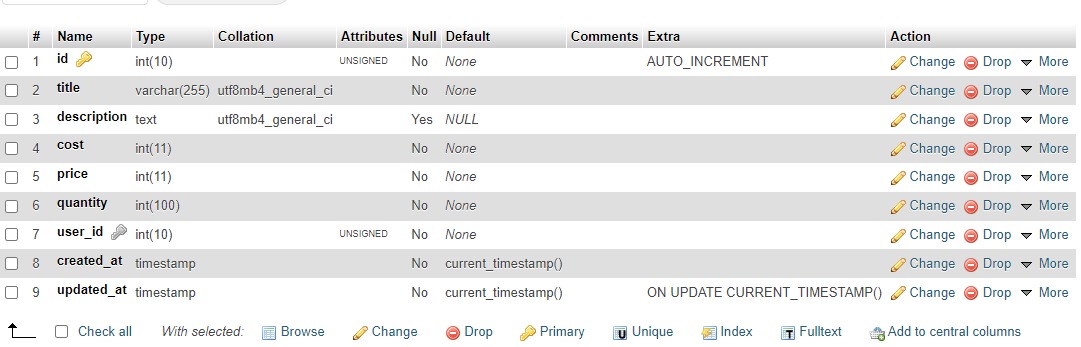
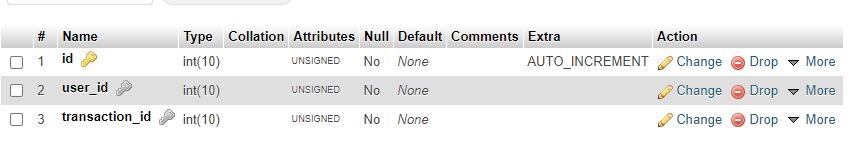


Table 4: Relationship table between the transaction and the user



# System implementation

After depending on the system's analysis and design stages, the implementation stage builds the whole system's components, including pages and lists, so that all of the system's pages are presented and the function of each page is explained separately. The website is meant to be easy and adaptable in order to suit the demands of all users.

Login page for the website:

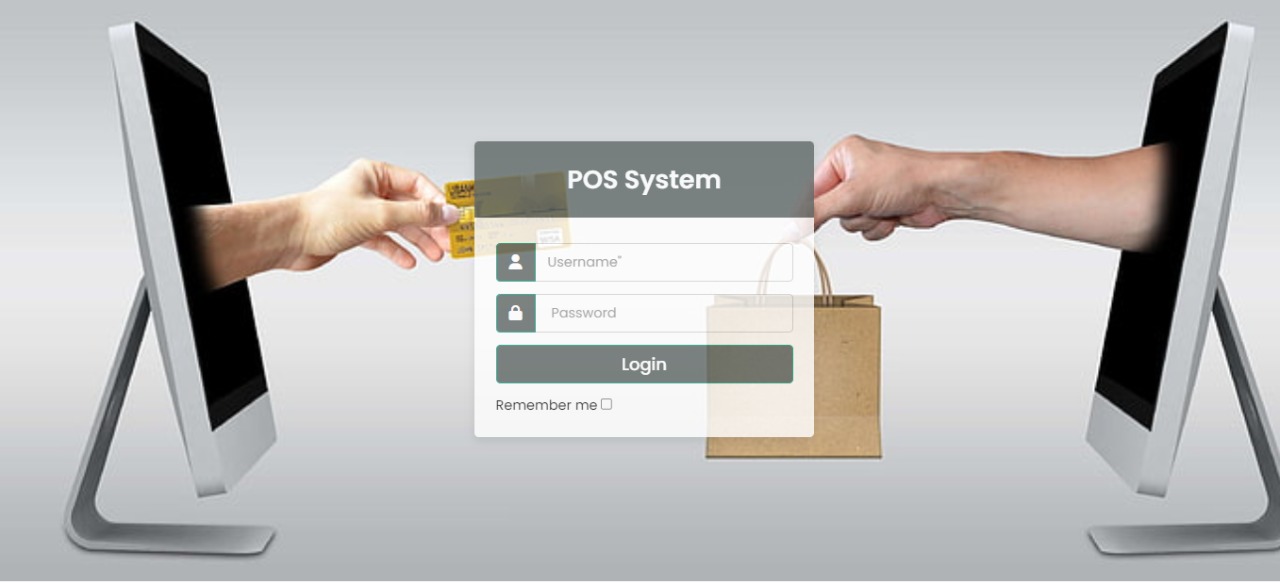


Figure 7: Logging in to the website

This screen displays a form with two components (username and Password), a login button, and a verification button (REMEMBER ME). Its purpose is to save the (User-id) when you return to the site. It is the initial page the user encounters. It was developed in a basic and adaptable manner, with no complication, and from here the user may log in to the site.

Home Page (Dashboard):

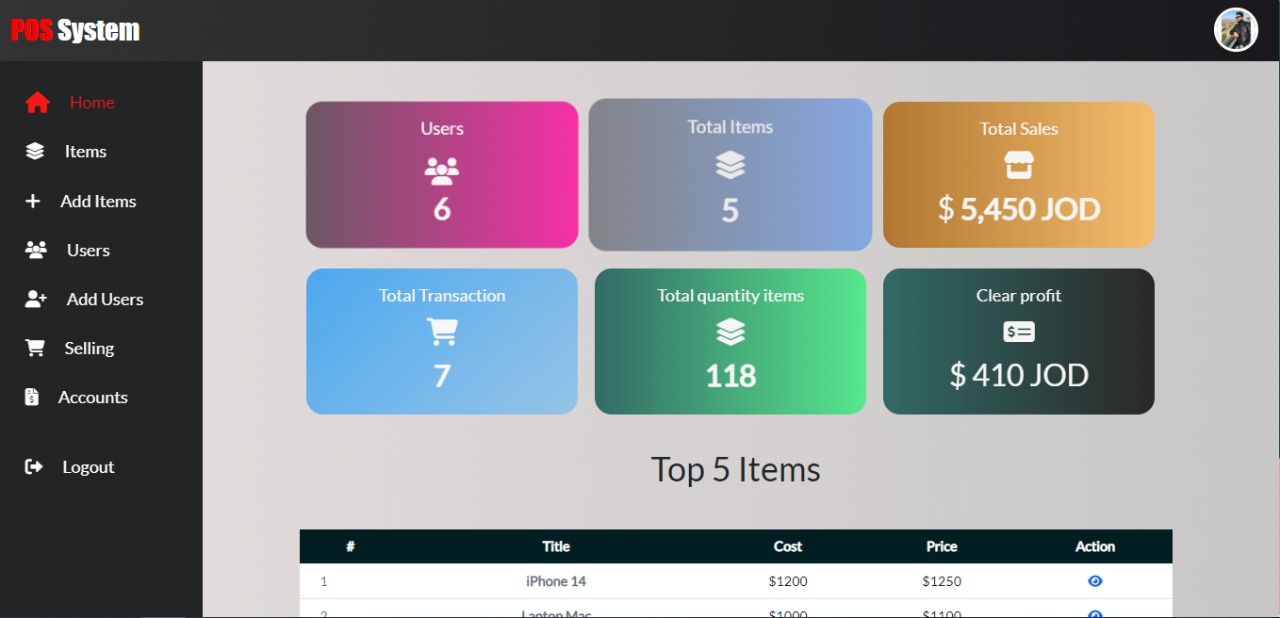


Figure 8: Home Page

The system's core categories and certain information are presented on the system's main page, such as: Total Sales, Total Transactions, Total Items, Total Users, Total Quantity, Total Profit , Top Five Expensive Items.

Selling Dashboard:

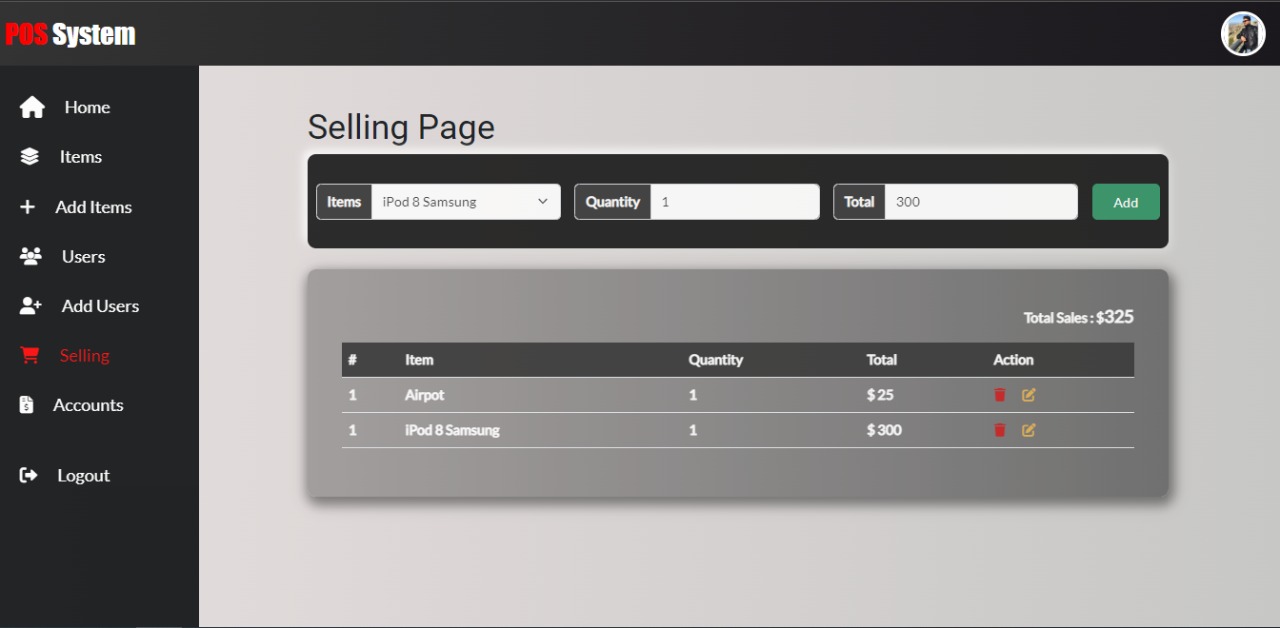


Figure 9: Selling Dashboard

This page lists all of the things in the store that are currently for sale. the seller can pick the quantity desired and hit the add button and show table from the bottom page which displays all of the sales made by the user today, and the button delete and update transactions.

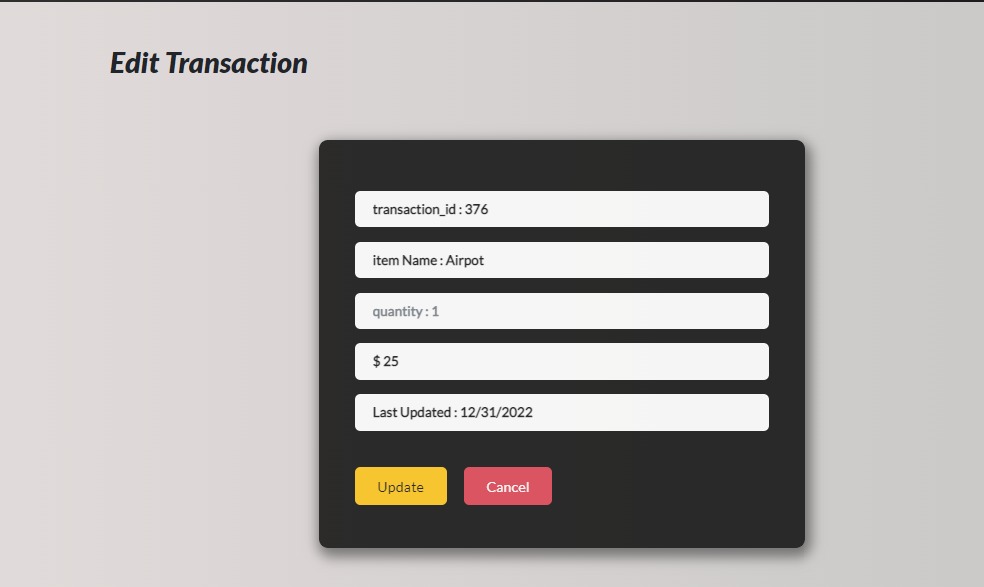


Figure 10: Edit Transaction

This page contains the information of transaction id and you can update this transaction quantity Either increase or decrease, and update item quantity from table item.

Items page:

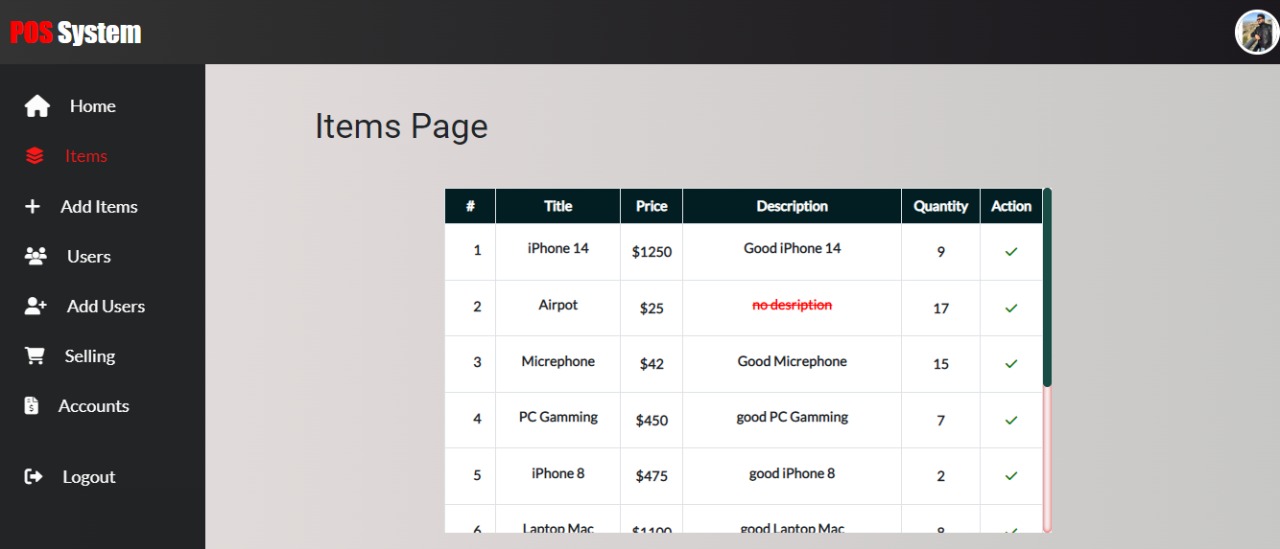


Figure 11: Items Page

The screen page allowing you to see how many things we have in stock as well as information about each product, such as cost, pricing, and quantity. There are also two buttons for each item, the first of which allows you to alter the product and the second of which allows you to delete the product.

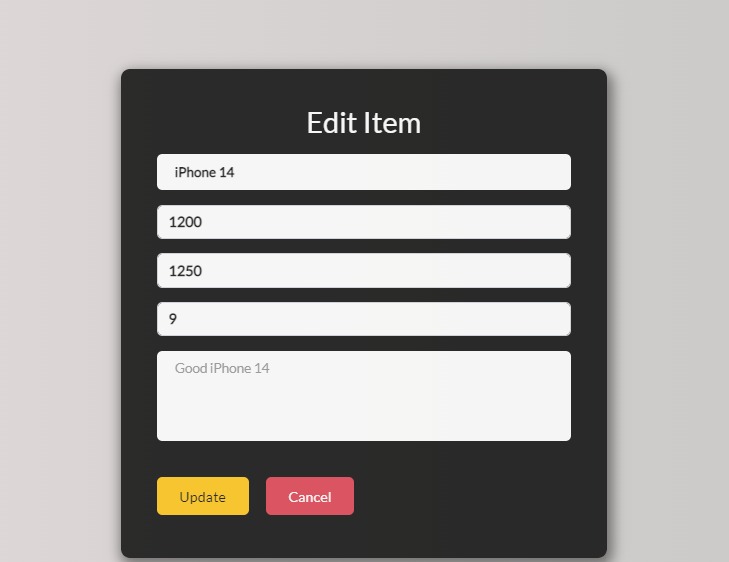
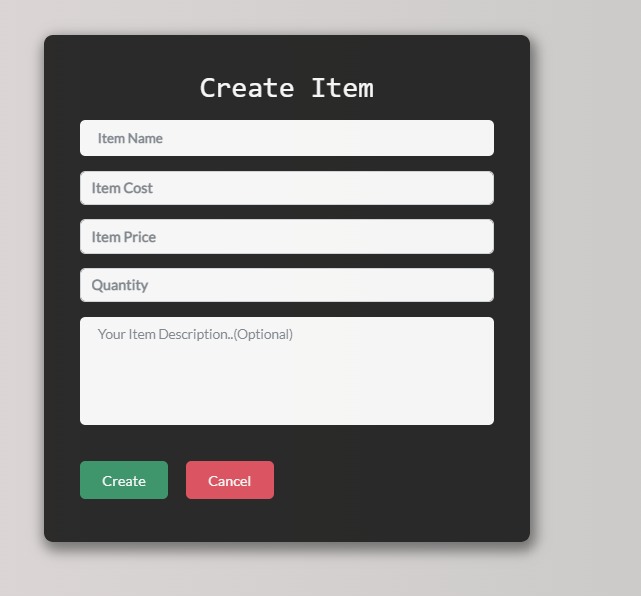


Figure 12: Edit Item

This page is about modifying the product, through which we can amend the product name, price, cost, or quantity, and modify the description.



.

Figure 13: Create Item

This screen page shows the form that can create new item by input title item and quantity and cost and price and the description is optional when you user needed added description to item.

Accounts Page:

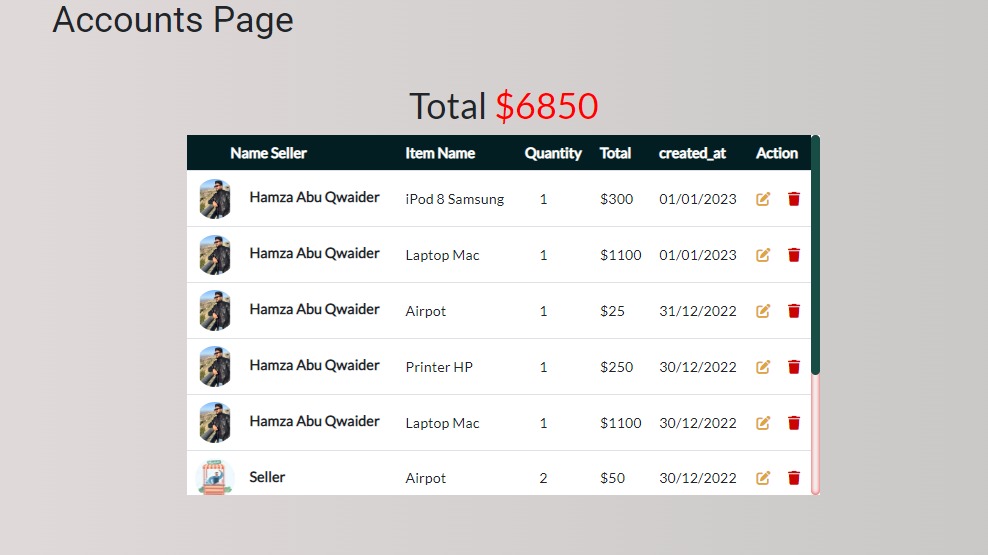


Figure 14: Accounts Page

This page is for the accountant to see all of the sales transactions that have occurred, as well as the name of the seller, the name of the product, and the amount sold by the seller. Each process also has two buttons, the first of which allows you to modify the sale process and the second of which allows you to remove the process.

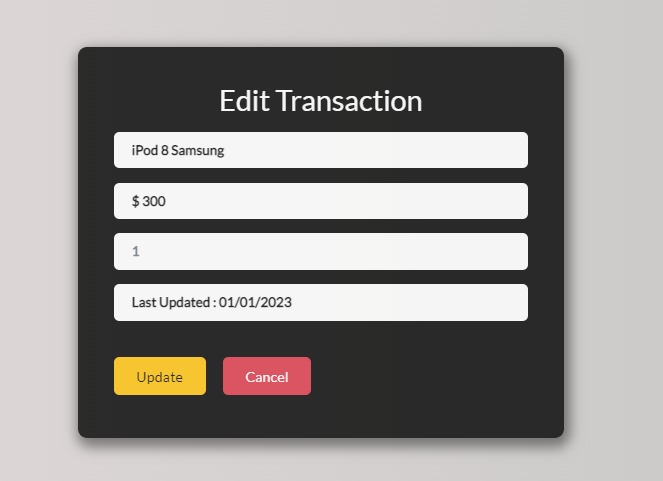


Figure 15: Transaction Edit Page

The accountant may adjust the quantity of a certain sale without changing the name or the

price item, and as soon as the amount changes, the total changes automatically.

Users Page:

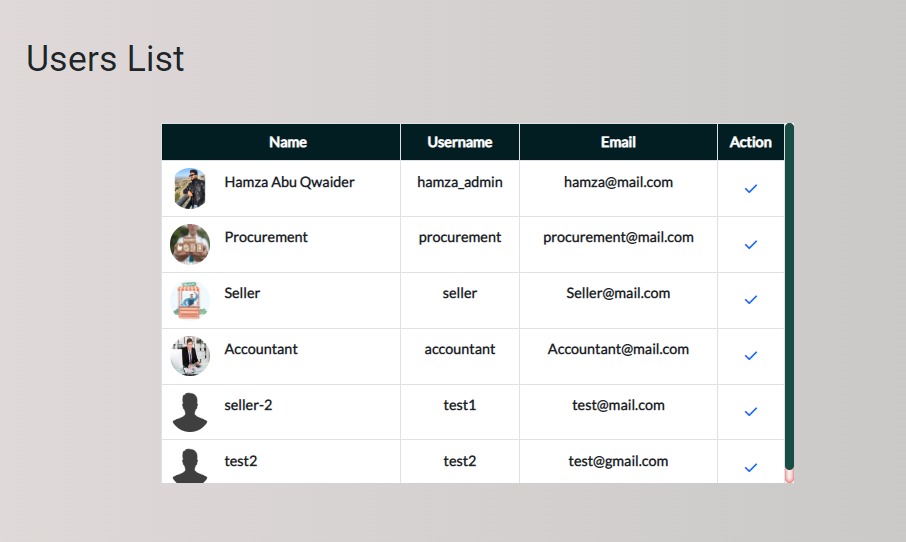


Figure 16: Users Page

This page is for the site management, and it allows them to see the number of users on the site as well as the powers of each user, as well as the check me button for each user, which provides particular information for each user.

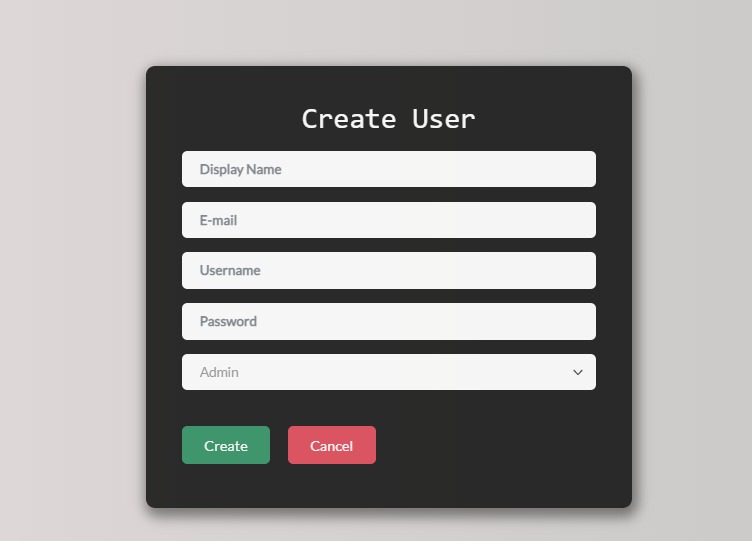


Figure 17: Create User Page

This page allows the site administrator to add users and delegate authority to them, such as an

accountant, seller, or admin, procurement.



Figure 18: Information User Page

Through this page, the system administrator may view the information for each user, and there are two buttons, the first to adjust the user's privileges, and the other to limit the user.



Figure 19: Edit User Page

This page is intended for the system administrator and the user but the user lacks the permission function, which allows the system administrator to update the user's information, his role in the system and even the user's photo. The user can only modify their personal information, such as their photo, name.

# Integration and Testing

Integration and testing are important phases in the software development life cycle (SDLC). The goal of integration testing is to ensure that the various components of a system work together as intended, while testing verifies that the system meets the specified requirements.

During the integration phase, individual components or modules that have been developed and tested separately are combined and tested as a group. This helps to identify any issues that may arise when the components are used together.

Testing is the process of evaluating a system or its component(s) with the intent to find whether it satisfies the specified requirements or not. There are various types of testing that can be performed during the SDLC, including unit testing, integration testing, system testing, and acceptance testing.

Unit testing is the testing of individual units or components of a system to verify that they are working as intended. Integration testing is the testing of a group of integrated components or modules to verify that they work together as intended. System testing is the testing of an entire system to verify that it meets the specified requirements. Acceptance testing is the testing of a system by the end-user to determine whether it is acceptable for use.

Overall, the integration and testing phases of the SDLC are important for ensuring the quality and reliability of a system. They help to identify and resolve any issues that may arise during development and ensure that the final product meets the specified requirements.

# Maintenance

The tasks necessary to guarantee that a software system continues to perform as intended are referred to as maintenance in software development. This can include bug fixes, new features, and hardware and software upgrades. Maintenance is an essential component of the software development life cycle (SDLC) since it helps to guarantee that the program stays dependable, efficient, and effective over time. The particular maintenance tasks carried out within the SDLC will be determined by the software system's requirements and the resources available to maintain it.

# Discussion:

I created the website based on the design pattern MVC in general project for Pos system Which I thought of as follows:

It tracks my inventory of items and commercial transactions that take place through the application, as well as an accounting system, adjusting the quantities I have, and displaying the most best-selling and most expensive commodities completely within the application.

At the beginning, tables were created through the use of phpMyAdmin and linked with the application. Within this data, only several tables were created, namely:

1. users
2. items
3. transactions
4. transactions-users

And give all powers to the admin and each user specific powers to enter and deal with data within the site.

A controller and a special model were created in each process, whether users, salesmen, accountants, or Items controllers.

# Conclusion

Finally, in order to effectively handle customer transactions, track sales and inventory, and provide real-time data to the administrator, HTU's store at King Abdullah Business Park requires the development of a Point of Sale (POS) system. The POS system will increase the store's efficiency, accuracy, and overall customer experience, resulting in increased profits and a stronger competitive advantage. The project's scope includes the creation of a web application that meets these requirements. The POS system will enable HTU to effectively manage its store and drive business success if the project objectives are met.

# Recommendation

The following recommendations can be made based on the project's objectives:

* + - To provide customers with a convenient checkout experience, we can integrate various payment methods into the system, such as cash, credit and debit cards, and mobile payments.
    - Put in place strong security measures to safeguard sensitive store data and prevent fraud and security breaches.
    - Maintain and update the POS system on a regular basis to ensure it remains reliable and effective in meeting the needs of the store.
    - Conduct user testing and collect feedback from employees and customers to continuously improve the POS system's functionality and usability.

# References

# <https://www.w3schools.com/>

# <https://www.phpmyadmin.net/>

# <https://www.google.com/>

# <https://www.php.net/>

# <https://www.youtube.com/>

# https://fonts.google.com/

# https://bootstrapdocs.com/