In partial fulfillment of the requirement for
System Development
420-940-VA section 05808
Vanier College

Implementation & Client Comments

7th Deliverable

Customer Name:
Delhi Kabab

Team Name:
Green Team

Team Leader:
Lei Wang

Team members:

I, Lei Wang, ID: 2295046, certify that I contributed to this deliverable

I, Bing Bai, student ID# 2295011, certify that I contributed to this deliverable

I, Shayekh Tarafdar, student ID# 2295035, certify that I contributed to this deliverable

I, Md Ibrahim Ullah, student ID# 2295025, certify that I contributed to this deliverable

I, Bermet Doolbekova, student ID# 2295017, certify that I contributed to this deliverable

I, Mark Benedict Muyot, student ID# 2295022, certify that I contributed to this deliverable

Date of Submission:
November 09, 2022

TABLE OF CONTENTS

1. STATEMENT OF PREVIOUS WORK EXPERIENCE

In this final deliverable, the team worked on the implementation of the inventory management application for Delhi Kabab. C# was used for this project and the team were also able to apply some of the programming concepts learned from other programming languages and tools applied from previous projects.

Table 1: Prior projects related to this deliverable

Projects Made	**Tools Used**	**Contributor/s**
Cars2Go Database	Microsoft SQL Server	Everyone
Simulation program	Java and Java Swing	Ibrahim and Mark
Online Grocery	Java, Spring Boot Framework, and Apache Maven	Bing Bai
Web Search Engine	Java	Shayekh
Hotel Reservation	Java	Lei and Bermet

2. EXECUTIVE OVERVIEW

In this deliverable, the team incorporated all the information acquired form all the previous deliverables into their implementation of Delhi Kabab's inventory management system. The team was able to develop the application using C# and Visual Studio. Although not all functions or features were implemented due to time constraint, the team was able to at least present to the client a running application with all core functions in it. The client was able to comment and give feedback regarding the application which can be seen in the succeeding sections of this deliverable. A summary description of the client and final business problem addressed were also included in the documentation.

In addition, the team discussed various decisions made regarding the design and implementation, security, and future work that could be done to finish or improve the system. The team was also able to provide revised user interface indicating the changes made from the prototype to the implementation **(see Appendix A)** and a list and screenshots showing which parts of the user stories were implemented **(see Appendix B)**.

3. DESCRIPTION OF THE CLIENT AND BUSINESS PROBLEM

Description of the client

The Team's client, Delhi Kabab, is an Indian restaurant located in Brossard. It is a cozy, sit-down restaurant plating tandoori chicken, spicy curries, and traditional Indian dishes (Delhi Kebab, n.d.). This fine Indian cuisine also serves variety of French food and various delicious desserts. The business is being operated by two partners with 6 employees. This business has been operating for more than 5 years. The restaurant is open everyday from 4:00PM till 9:00PM, they have their own delivery service, and they also are using 3rd party delivery services such as Doordash, SkipTheDishes, and Uber Eats**.**

Business Problem

Delhi Kabab has an inefficient system in monitoring their inventories. The restaurant is using a notepad to list necessary items to be ordered such as ingredients for their food preparations, cleaning items, toilet supplies etc. According to the client, they often lose their list and sometimes forget to update the list of ordered items in their system. In consequence, the restaurant's inventory record often produces inconsistent information. This type of system leads to increase in spoilage and loss if inventories like ingredients are not tracked and monitored properly. The system also takes a lot of time tracking and reviewing their inventory. There is no way to view a comprehensive report easily, such as the ingredient supply level.

Customer satisfaction is sometimes affected by the inconsistencies in their inventory management system. There are times where customers want to order something, but the food is not available because some ingredients are missing. This leaves unsatisfied customers which could affect the bottom line of the business.

Lastly when it comes to security, their excel file for tracking inventories is not encrypted or password protected. This can be easily access and modify by anyone within the organization. The owner realized that this type of system can lead to pilferage without him knowing it. Someone can just manipulate the stock level and steal some inventories used in the restaurant.

4. REVISED SUMMARY DESCRIPTION OF THE SYSTEM

The inventory management system developed for Delhi Kabab has the following functions. The user must first enter login credentials to be able to use the application. The team determined that there will be two user types or roles using the system. First role is the administrator, who also happens to be the restaurant's owner. The administrator will basically have full access privilege in the system. The second type of user is the guest who can basically just view the list of items in the system. The guest user cannot do any modification within the system. In case a user forgets his/her username, the system will have a username retrieval mechanism that when a user clicks this function then the username will be sent to his/her email. There will also be a forget password feature in case the user forgets his/her password

When it comes to user account management, the administrator may perform CRUD operations such as adding new users, viewing user profiles, deleting user profiles and updating user profiles including password, email, contact, etc. The admin can also perform CRUD operations in managing the items in their inventory such as adding an item that they want to track, display or view list of all items, delete or remove an item form the system, and update an item’s description such as expiration date, brand name, stock level, etc.

The system will record all the items they are tracking, and the application will include a filter feature to let users search for an item by category or by name. The user can also sort the items by price. This will give the restaurant a way to quickly monitor and see their items and its stock level.

Not only that the system can provide a list of items that the restaurant currently has but the system will also be capable of generating list of expired items and items that are running low in stock.

In addition, our client also wanted to add a category and supplier management function that would help the admin do CRUD operations such as adding, updating, and deleting a category or supplier in the system. The client also added that a help function would also be useful to add in this project. These new features were added to the team's user story and user story map.

5. CLIENT'S COMMENT

The client was very happy to see the application that we were able to implement. The team was able to do a big improvement especially when it comes to the UI. This was possible because of the help and tips that were provided by our professor, Mr. Lebensold, from deliverable 5. Client was satisfied but since most of the functionalities were not implemented yet such as password retrieval and username, error handling, etc., the client is looking forward on having the complete version of the project. The team showed a side-by-side comparison of the old and new designs with incomplete implementation to the client.

Sample design comparison
(Old vs New)

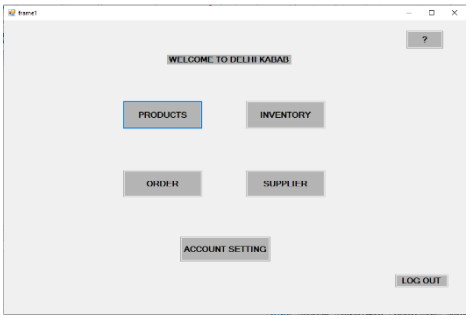


Figure 1 Welcome Page (admin)-old

Figure 3 Welcome page(admin) -new

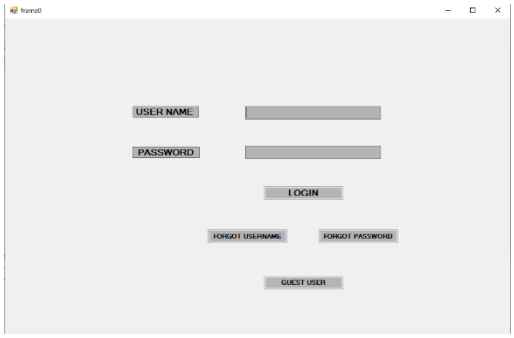


Figure 2 Login (old)

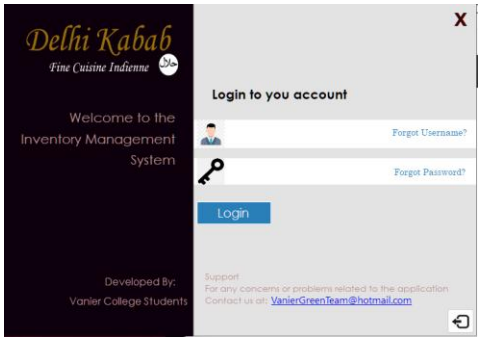


Figure 4 Login (new)

6. DESIGN AND IMPLEMENTATION

In terms of the design, the team decided to make a major revision in terms of UI. The team's approach was to design a minimalist UI to improve the readability of the application. The team also implemented a color scheme for admin and guest main panel to create distinction for the type of users. Icons that are easily understood were also used for the buttons to increase familiarity in using the application.

In terms of implementation, all major functions were implemented such as CRUD operations in managing inventory. Some functions such as notification, username, and password retrieval, etc. were not yet implemented. For database, all tables were normalized so that it won't create data anomalies and redundancies in the future. The team also identified 5 entities for the database.

The team used C# since it is the best programming language when developing desktop application in Windows. The team also used Visual Studio as in IDE and injected fontawesome.sharp dependency in the application to improve the aesthetics of the program. The team planned to use WPF but ended up using WinForm because it is lighter than WPF. WPF on the other is more complicated to use and heavy on the system. The team used Microsoft SQL Server since we are using a relational database model. Another advantage of using Microsoft SQL Server has excellent data retrieval, highly secured, and easy to install.

7. DESCRIPTION OF CURRENT SECURITY MEASURES

The team implemented security measures to safeguard confidential information and maintain data integrity of the inventory management system of Delhi Kabab. Unlike with their old system, the system implemented user roles within the system and limit the privilege or access of the guest users to mostly just viewing. In this way, the admin is the only one who can do modification and view essential information within the system. Limiting the access and access privilege will safeguard data integrity of the system and minimize data manipulation.

A login function was also implemented for the system to be notified if the user is an Admin or guest. This will authenticate the profile of the user and be given the proper access within the system. This will ensure that no unauthorized user would be accessing the system.

The team also made sure that all the libraries and dependencies injected were free from vulnerabilities or possible security risks. For the database, the team used Microsoft SQL Server since it has some reliable built-in security features.

8. FUTURE WORK

UI Improvement

In terms of UI, the team and client were very satisfied with the recent improvements done in the program. The team is planning on migrating the system using UWP framework from Winform since UWP is much newer and it will make the UI more dynamic. The team will improve the responsiveness of the application so that it will be compatible in different screen size but for now the screen size is locked and not sizable. The team is also planning to not use grid tables for future work.

Unimplemented User Stories and Functions

Out of 55 user stories the team was not able to implement 22 user stories.

Unimplemented User Stories			
ID	**As an <type of user/ persona>**	**I want <goal/ objective> User story**	**So that <benefit/result/some reason>**
3	Administrator	to have a forget password,	I can easily reset it when I forget my password.
4	Administrator	to be able to retrieve username,	I can retrieve my username if I forget it.
6	Administrator	to be able to update password,	I can change my password.
7	Administrator	to have a delete user account,	I can manage the users in the system.
8	Administrator	to have an update user profile,	I can update user profile if there are any changes.
14	Administrator	to get notification when a particular item has low stock level,	I would be aware if an item is running low.

15	Administrator	to see list of low-level items,	I can easily monitor those items that are running low.
16	Administrator	to get notification when a particular item is about to expire,	I can remove them right away from the stockroom or use them right away.
27	Administrator	to be able to print list of expired items,	I can use this document in removing those items from our stockroom.
28	Administrator	to be able print list of items in the system,	I can use this document for monitoring and tracking purposes.
53	Administrator	to have a help function,	I can easily check the system's documentation if I need it.
34	Guest	to be able to search item,	I can easily track an item that is being monitored
35	Guest	to be able to view items being monitored,	I can easily manage our inventory.
36	Guest	to see quantity of each item in the inventory list,	I can easily monitor stock level of our inventory list.
37	Guest	to get notification when a particular item has low stock level,	I would be aware if an item is running low.
38	Guest	to see list of low-level items,	I can easily monitor those items that are running low.
39	Guest	to get notification when a particular item is about to expire,	I can remove them right away from the stockroom or use them right away.
43	Guest	to see items based on their category,	I would have an efficient way for checking an item in a list.
55	Guest	to have a help function,	I can easily check the system's documentation if I need it.

Recommendations Regarding Future Security Measures to be Taken

The team is planning to implement an MVVM design to add an additional layer of security within the program. The team also suggest implementing a change password feature that

will prompt the user every 6 months. Another recommendation is to maintain the firewall of the client's system always on and run a virus/spyware scan every 6 months. Lastly, the team recommended to create a backup file.

Recommendations regarding unit and integration test strategies to be used.

The team will implement MsTest framework since it supports unit testing in C#. The team will test every method implemented to identify early problems in the application. The team will perform integration test to check database connectivity.

Acceptance Test Plan

Purpose

The following is a set of recommendations that our team suggested with regards to testing the application. The purpose of these recommendations is to be able to verify that the application conforms to the initial exceptions of the project and to ensure that everything is working as expected.

Who will be testing the application?

The person testing the application will be the only user of it, which is the owner of Delhi Kabab, Mr. Masundra Chandra.

Testing environment

The environment in which the testing will take place will primarily be on Delhi Kabab's system

What type of testing do we recommend and why?

User Acceptance Tests (UATS) will be done. This will make sure software is functioning properly and meets client's requirements. This will also determine if weather any features have been overlooked or having any issues. This will help determine future work that is needed to be done for this project.

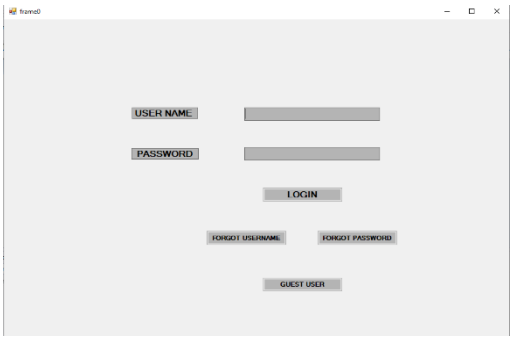
Below are some user acceptance tests the team generated. To check the whole list of UAT please see Deliverable 4.

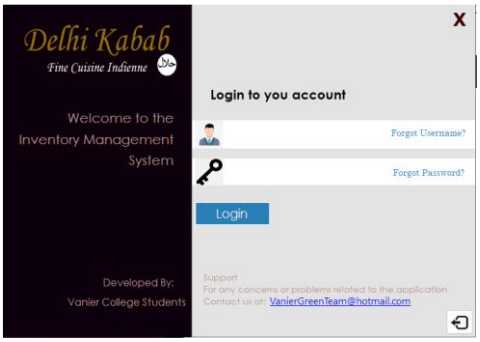
ID	User Stories	Function	Acceptance criteria
1	**As** an [admin], **I want** [to have a login] **so that** [I'll have a secured application].	Login	**Scenario 1**:Admin enters the correct login credentials**.** **Given** [That the user is an administrator]**,** **when** [the user enters the correct login credentials]**,** **then** [The system will give the access to the user].
			Scenario 2: Admin enters the wrong login credentials**.** **Given** [That the user is an administrator]**,** **when** [the user enter the wrong login credentials]**,** **then** [The system will not give access to the user] **And** [the system displays an error message "user name or password is incorrect"].
2	**As** an [admin], **I want** [to have different access level] **so that** [the system will have additional level of security].	Access level	**Scenario 1**: Admin can access all features**.** **Given** [The administrator enters login credentials]**,** **when** [the admin access in to the system]**,** **then** [The system will give full access to the user].
			Scenario 2: Guest user can access the limited features**.** **Given** [the user enters as a guest]**,** **when** [the user access the system]**,** **then** [The system will give limited access to the user].
3	**As an** [Admin], **I want** [to have a forget password] **so that** [I can easily reset it when I forget my password].	Forget Password	**Scenario 1**: An admin user forgot his password **Given** [That the user is an administrator]**,** **when** [the user enters the wrong password]**,** **then[**the system display a error message**]** **And** [The system send a link to the administrator email address upon clicking on forget password option]. **And** [The user changes his password using this link]. **And** [the database will remove the old password]

			And [replace it with the new password] **And** [display a success message]
			Scenario 2: A guest user tries to access login **Given** [That the user is a guest user]**,** **when** [the user enters the wrong password]**,** **then** [The system sends a link to the administrator email address]. **And** [The user can not change the password].
4	**As an** [Admin], **I want** [to be able to retrieve username] **so that** [I can retrieve my username if I forget it].	Forget Username	**Scenario 1**: An admin user forgot his username **Given** [That the user is an administrator]**,** **when** [the user tries to change the username using forget username option]**,** **then** [the system will sent a link to the user email].
			Scenario 2: A guest user try to access login **Given that** [That the user is a guest user]**,** **when** [the user tries to change the username using forget username option]**,** **then** [the system will sent a link to the admin email]. **and** [the system will not change username]
5	**As** [an admin], **I want** [to be able to add users] **so that** [I can create multiple users].	Add Users	**Scenario 1**:An administrator can add another Admin account **Given** [That the user is an administrator]**,** **when** [the user log in to the system]**,** **then[**the admin selects create admin account**],** **and** [the system record the information and create another admin account]. **and** [a successful message will be displayed]**.**
			Scenario 2: A guest user can not add another admin account **Given** [That the user is a guest user]**,** **when** [the user tries to add another account]**,** **then** [the system will display an error message] **and** [the new user account would not be created]

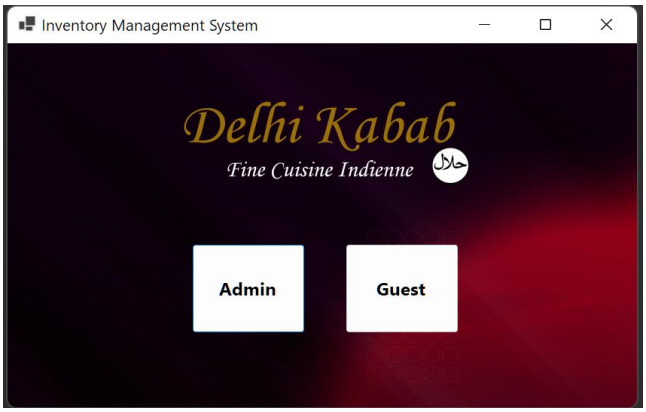
9. APPENDIX A: REVISE UI VS PROTOTYPE

The team made a major revision with the UI vs their prototype. Almost all pages were changed and improved. These comparisons between the prototype and the final version are shown below.

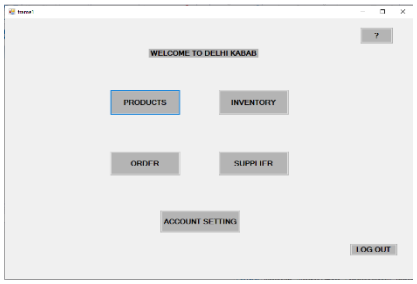


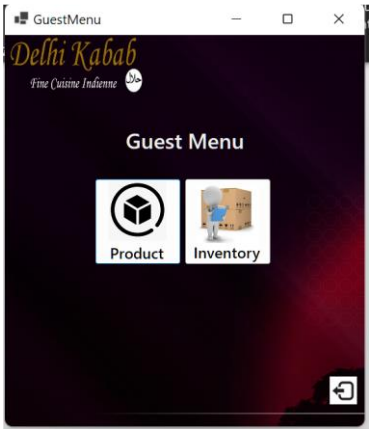


Revised login page was improved. A left panel was added for the logo and title of the application. The group developed the application was mentioned at the bottom of the panel. The right section of the login contains the login textbox and the login button. Button appears more visible than the previous design. The layout looks more professional and easier to understand.

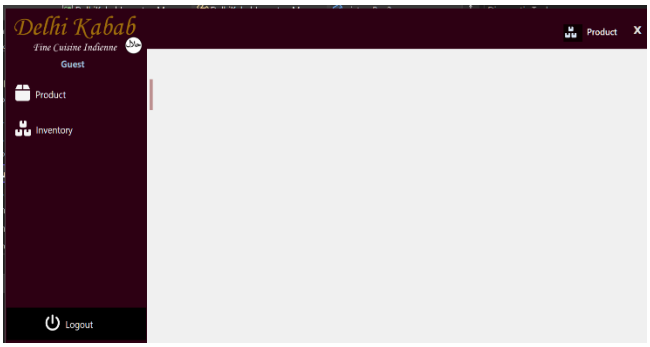


A welcome page was added to choose which type of user will be using the system.

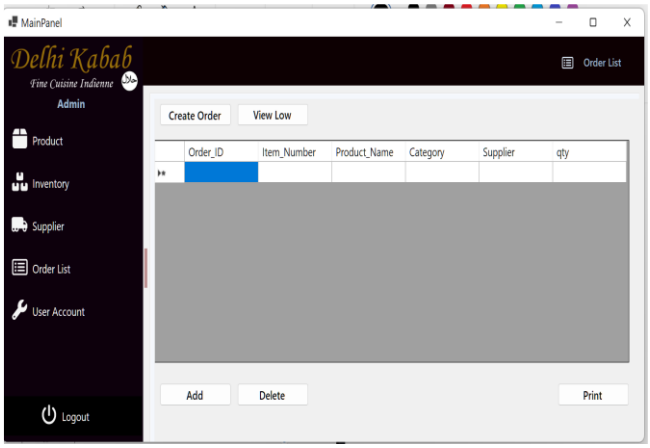




Admin and Guest main menu were seperated. The team created a menu using buttons and icons to give familiarity to the users while navigating this page. The background image of Delhi Kabab's website was used as a theme in both main menus.

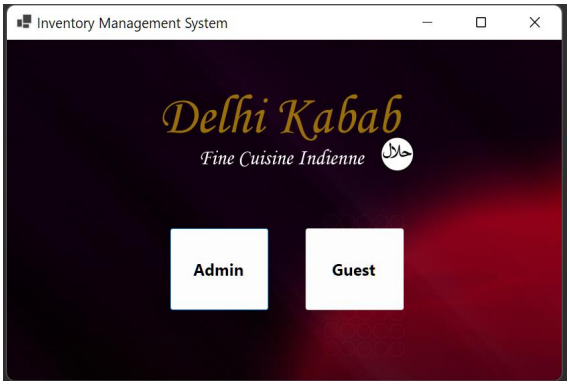


These two main panel were added in the new UI. The admin panel to the left contains all the functions of the admin while guest main panel to the right contains all the functions for guest user. The team used different colors to create distinction of what type of user will be accessing these page. The icon and text at the upper right corner of these pages wil change dynamically depending on what function the user will select at the left section of these pages. A small rectangular shaped panel that is in color rose will dynamically move alongside tha selected function to remind user where currently they are.

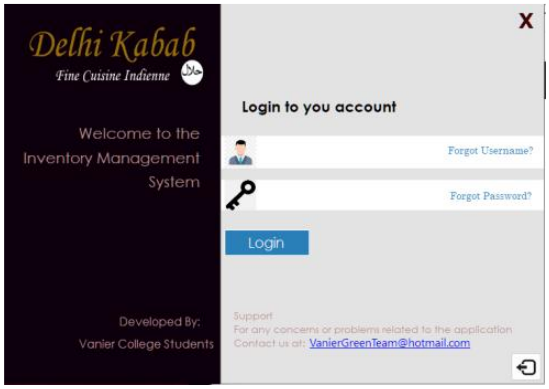


Grid Tables will pop up inside the right panel of the new page instead of using soley the grid page. This will make it appear dynamic.

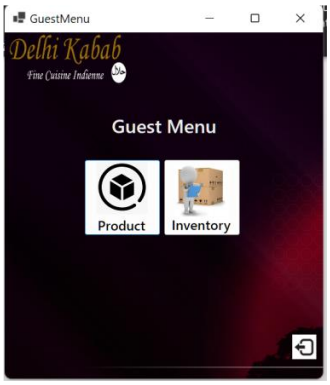
10. APPENDIX B: USER STORY IMPLIMENTED WITH UI SCREENSHOT



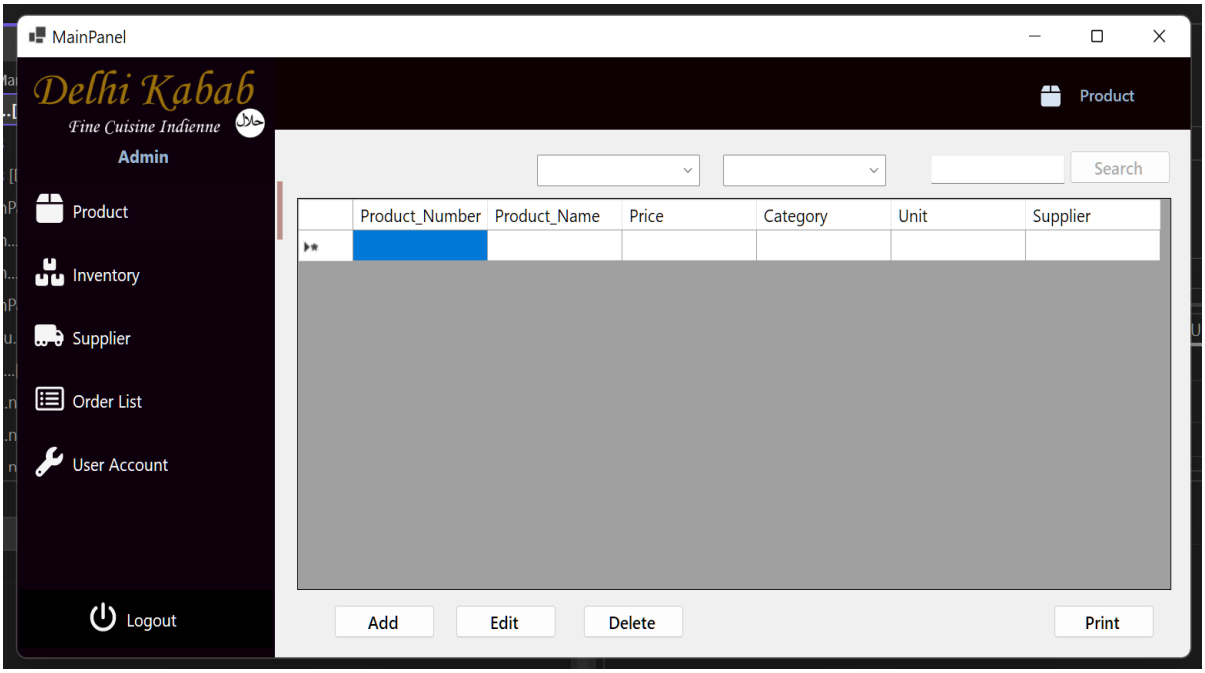
ID	As an <type of user/ persona>	I want <goal/ objective> User story	So that <benefit/result/some reason>
ID 2	Admin	to have different access levels,	the system will have additional level of security.



ID 1	Admin	to have login,	I can login.
ID 3	Admin	to have a forget password,	I can easily reset it when I forget my password.
ID 4	Admin	to be able to retrieve username,	I can retrieve my username if I forget it.



Admin and Guest main menu were newly added UI's.



The rest of the CRUD operations for managing account and items in the inventory could be seen in this panel depending on the button the user will select on the left side of the screen. Below are the user stories implemented for this page.

5	Administrator	to be able to add user,	I can create multiple users.
6	Administrator	to be able to update password,	I can change my password.
7	Administrator	to have a delete user account,	I can manage the users in the system.
8	Administrator	to have an update user profile,	I can update user profile if there are any changes.
9	Administrator	to be able to add item in the system,	I can track and monitor items.
10	Administrator	to be able to delete item,	I can delete an item that is no longer monitored or tracked in our inventory.

11	Administrator	to be able to view item being monitored,	I can easily manage our inventory.
12	Administrator	to be able to update item,	I can change or update item's description if needed.
13	Administrator	to see quantity of each item in the inventory list,	I can easily monitor stock level of our inventory list.
17	Administrator	to be able to update expiration date of an item,	I can easily adjust expiration date when a wrong date is recorded.
18	Administrator	to view list of expired items,	I can remove them from the stockroom.
19	Administrator	to remove expired items from the list,	the system stays up to date.
53	Administrator	to have a help function,	I can easily check the system's documentation if I need it.
34	Guest	to be able to search item,	I can easily track an item that is being monitored
35	Guest	to be able to view items being monitored,	I can easily manage our inventory.
36	Guest	to see quantity of each item in the inventory list,	I can easily monitor stock level of our inventory list.
40	Guest	to view list of expired items,	I can remove them from the stockroom.
41	Guest	to remove expired items from the list,	the system stays up to date.