# **CS-355 Databases**

# **Fall 2019**

# **Project Proposal**

**DBMS for a Restaurants Chain**

**Laiba Fatima Khan (lk04067)**

**Muhammad Shahzain (ms03977)**

**Kabir Kumar (kk03925)**

# ***Submitted to***

**Dr. Ayaz ul Hassan Khan**

****

**Habib University, Karachi**

Table of Contents

[**About Restaurant Management System ………………………………………………….2**](#_Toc36022999)

1.[**Using Restaurant Management System ……………………………………………….3**](#_Toc36023000)

1.1 Login ………………………………………………………………………………………3

1.2 Main Menu………….………..……………………………………………………………3

1.3 Handling Orders………………………………………………………………………….4

1.3.1 Adding Orders………………………………………………………………………...4

1.3.2 Search Orders…………………………………………………………………………

1.4 Handling Staff

1.4.1 Adding Staff…………………………………………………………………………

1.4.2 Search Staff…………………………………………………………………………

1.5 Handling Food

1.5.1 Adding Food…………………………………………………………………………

1.5.2 Search Food…………………………………………………………………………

# 

# About Restaurant Management System

|  |  |
| --- | --- |
|  | A database driven software, the system can be used by restaurants of all sizes to keep a track of all the past and present staff employed and orders placed throughout its running. This data can also be modified and viewed through the interface of this system. The software also provides different levels of authorization under which data can be viewed or modified only by individuals with specified permissions. This utility will also be used to keep a track of all the changes made to the data and the person who made these changes.  The backend technology that maintains a database of all the information for this software is the SQL Server by Microsoft. The front end of the software, abstracting all the working from the end user, is constructed using C# and .NET framework.  The aim of this software is to relieve up-and-coming entrepreneurs from the stress of expensive management software and provide an easy and cost-effective solution. |

# Using Restaurant Management System

|  |  |
| --- | --- |
|  | This section is dedicated to introducing the user interface of the system and guide the user to using the system. |

# 1 Login

|  |  |
| --- | --- |
|  | Once the system has started, a window greets the user asking for the username and password. Through this, we enforce the authorization utility of the software. The window is shown in Figure 1. |

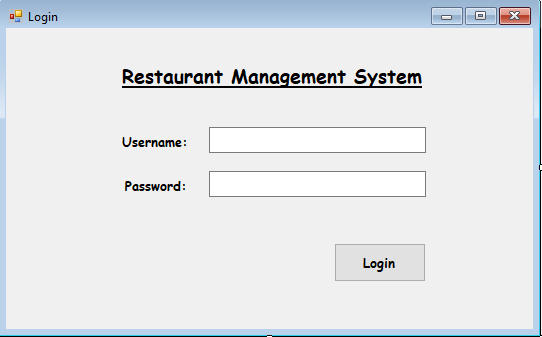


Figure 1: Login window

#### Figure 1: Login Window

# 2 Main Menu

|  |  |
| --- | --- |
|  | On logging in, the user will face a simple interface that shows the different abilities of the system. This window is shown in Figure 2. |

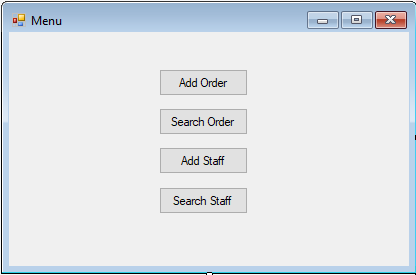


Figure 2: Menu window

# 3 Handling Orders

|  |  |
| --- | --- |
|  | This sub-section explores the interface provided by the system to handle orders. All the functionality related to orders, whether it may be adding orders or viewing information related to the orders, is described. |

# 1.3.1 Add Orders

|  |  |
| --- | --- |
|  | The interface to add orders is shown in Figure 3. As visible, adding an order will require the id of the staff member taking and serving the order. After that, the food items and their quantity included in the order are put in. These items are then displayed in the following box to give an overview of the order. A separate box is dedicated to inputting the details of the customer placing the order. |

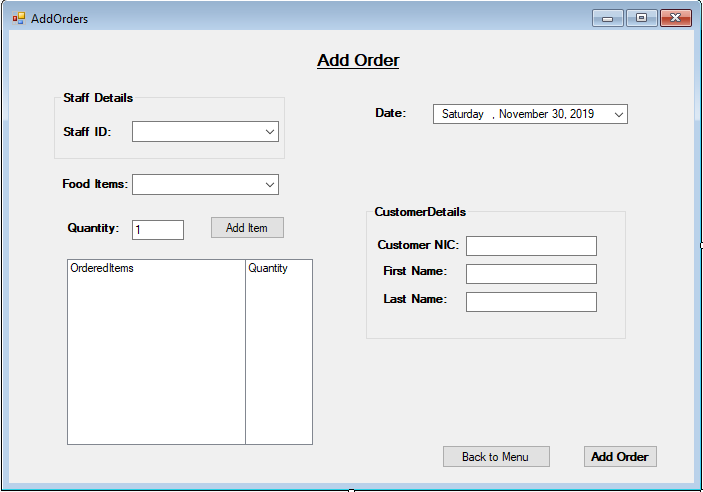


Figure 3: Add Order Window

# 1.3.2 Search Orders

|  |  |
| --- | --- |
|  | Figure 4 shows the window to search through the database of orders that is maintained by the system. There are four boxes to put in data related to an order and the relevant results are presented in the box below. It is not necessary for the user to put in all the details. The search functionality works as per whatever data the user provides.    Figure 4: Search Order Window |

# 1.4 Handling Staff

|  |  |
| --- | --- |
|  | This sub-section explores the interface provided by the system to handle the staff employed by the restaurant. This includes employees in all positions. All the functionality related to staff, whether it may be adding staff, deleting staff or viewing information related to them, is described. |

# 1.4.1 Add Staff

|  |  |
| --- | --- |
|  | The Add Staff window is used to add new employees to the database. All the data can be put in the text boxes as shown in Figure 5. The fields specified with an asterisk are mandatory fields, without which the system would not advance to adding the employee. The functionality to modify already present records is hoped to be achieved in the final product. |

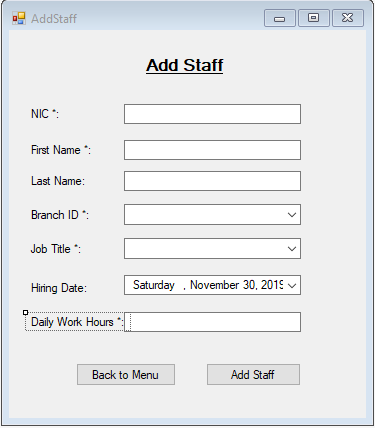


Figure 5: Add Staff Window

# 1.4.2 Search Staff

|  |  |
| --- | --- |
|  | The window to search the staff records is shown in Figure 6. This window works in similar fashion to the window in Figure 4, as the user does not have to necessarily put in all the fields and the search functionality will work with whatever is provided. This functionality accommodates the lapse of memory on the user’s side. |

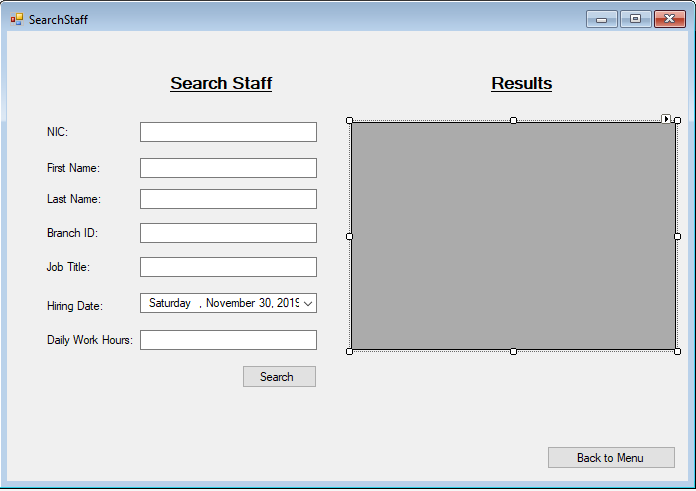


Figure 6: Search Staff Window

# 1.5 Handling Food

|  |  |
| --- | --- |
|  | This sub-section explores the interface provided by the system to handle the staff employed by the restaurant. This includes employees in all positions. All the functionality related to staff, whether it may be adding staff, deleting staff or viewing information related to them, is described. |

# 1.5.1 Add Food

|  |  |
| --- | --- |
|  | The Add Staff window is used to add new employees to the database. All the data can be put in the text boxes as shown in Figure 5. The fields specified with an asterisk are mandatory fields, without which the system would not advance to adding the employee. The functionality to modify already present records is hoped to be achieved in the final product. |

# 1.5.2 Search Food

|  |  |
| --- | --- |
|  | The window to search the staff records is shown in Figure 6. This window works in similar fashion to the window in Figure 4, as the user does not have to necessarily put in all the fields and the search functionality will work with whatever is provided. This functionality accommodates the lapse of memory on the user’s side. |