

**UNIVERSITI TEKNIKAL MALAYSIA MELAKA**  
FAKULTI TEKNOLOGI MAKLUMAT DAN KOMUNIKASI

**WORKSHOP 1**

**REPORT**

|  |  |
| --- | --- |
| **Name** | CHEOK LI LI |
| **Matric Numbers** | B031310127 |
| **Course** | Bachelor of Computer Science (Software Development) |
| **Project Title** | Food Ordering System ( FOS ) |
| **Cluster** | BITS/S1G2 |
| **Supervisor** | EN. MUHAMMAD SURYANATA |
| **Assessor** | EN. YAHYA IBRAHIM |

**Table Of Content**

**Table Of Content------------------------------------------------------------------------------------------------------1**

**Chapter 1: Introduction---------------------------------------------------------------------------------------------2**

1.0 Project Background----------------------------------------------------------------------------------2

1.1 Problem Statement-----------------------------------------------------------------------------------2

1.2 Objectives---------------------------------------------------------------------------------------------2

1.2 Scope------------------------------------------------------------------------------------------------2-4

**Chapter 2: Analysis--------------------------------------------------------------------------------------------------4**

2.0 Comparison-------------------------------------------------------------------------------------------4

2.1 Problem Description---------------------------------------------------------------------------------5

2.2 Problem Decomposition-----------------------------------------------------------------------------5

2.3 Structure Chart----------------------------------------------------------------------------------------6

**Chapter 3: Design----------------------------------------------------------------------------------------------------7**

3.0 Data Model (ERD) ----------------------------------------------------------------------------------7

3.1 Data Dictionary------------------------------------------------------------------------------------7-8

3.2 Data Flow Diagram (DFD) ---------------------------------------------------------------------8-14

3.3 Flow Chart---------------------------------------------------------------------------------------15-19

3.4 Pseudo Code-------------------------------------------------------------------------------------20-22

**Chapter 4: Implementation-----------------------------------------------------------------------------------23-31**

**Chapter 5: Conclusion----------------------------------------------------------------------------------------------32**

**Chapter 1: Introduction**

**1.0 Project Background**

Food Ordering System (FOS) is a database system which accelerates customer orders. In most cases, the company has problem with the manual food ordering which ordered by their staff. Most of the company receives the complaint about their order is written wrongly by the staff. This will spoil the image of the restaurant. Therefore, the computerized ordering system is needed as this system help to provide a better service for customers. The Food Ordering System is used by staff (waiter/waitress), kitchen employees, cashier and manager.

A Food Ordering System is designed and used in the restaurant. The main purpose of this project developed was to substitute from manual method of food order to a computerized system. This system is necessary to be used in restaurant in order to eliminate problems that may occur during the ordering process. Furthermore, it might to provide better customer service than manual.

**1.1 Problem Statement**

First of all, waiters take a long time to drop the customer’s order to the cashier and kitchen. Sometimes, the foods or drinks that ordered by customer is not available because of some reason, and the staff need to keep updating from the kitchen by asking the kitchen employees or manager every time. This problem causes the waiter serves their food or drinks late, besides extend the customer’s waiting time, but also influence the customer’s mood. In addition, the staffs need more paper to take the order. The paper that contains customer’s order is easily missing.

**1.2 Project Objective**

Based on the problem stated above, the objectives of FOS project are :

1. To develop computerized food ordering system in restaurant
2. To store all the customer’s order in the system
3. To easy the waiters know about whether the food is available or unavailable

**1.3 Project Scope**

FOS is using Visual Basic language and Mysql Database for implementation.

This system consist of 4 modules, there are Manager Module, Kitchen Module, Waiter Module and Cashier Module.

-> Manager Module

* Login
* Manage Staff Login Account
* Add New Staff Login Account
* Delete Staff Login Account
* Update Current Staff Login Account
* Manage Menu
* Add New Menu
* Delete Menu
* Update Current Menu
* Manage Table
* Add New Table
* Delete Table
* Update Current Table
* Generate Daily Sales Report
* View Daily Sales Report
* Print Daily Sales Report
* Generate The Best Selling Menu Information
* View The Best Selling Menu Information

-> Kitchen Module

* Login
* Update Menu Status
* View Order Detail

-> Waiter Module

* Login
* View Order
* Manage Order
* Add New Order
* Update Current Order
* Delete Order (Cannot delete the order after save)

-> Cashier Module

* Login
* View Order
* Payment
* Manage Order
* Add New Order
* Update Current Order
* Delete Order (Cannot delete the order after save)

**Chapter 2: Analysis**

**2.1 Comparison**

|  |  |  |
| --- | --- | --- |
| Difference between current and new system | Current Food Ordering system in WaterLily Restaurant | New Modified Food Ordering System (FOS) in WaterLily Restaurant |
| Login Account with different username and password | No username input and all staff access with the same password | Every staff have their own login account |
| Print Customer Bill | Yes, but the cashier ID is unknown. | Yes, who order the menu and who paid the bill is recorded. |
| Print Daily Sales Report | No | Yes |
| Generate The Best Selling Menu Report | No | Yes, with a bar graph. |
| Customer Order | The detail like the staff ID (who serves the customer) is unknown. | Record the waiter ID into database |
| Menu Status | No, waiter have to ask frequently in a day | Yes, each menu has its status. |

**2.1 Problem Description**

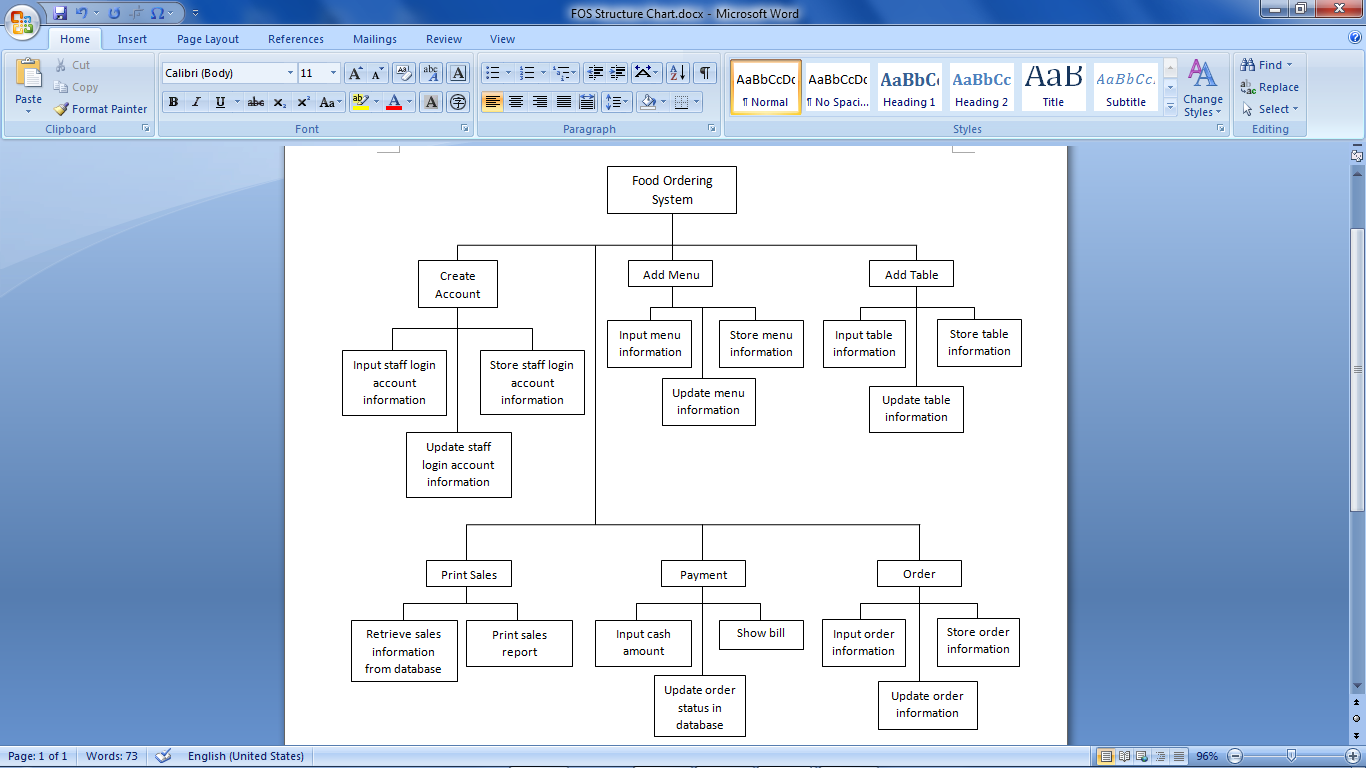
At the end of each day the restaurant manager need to compute manually the daily sales of the restaurant, and determine the best selling menu for each day. Besides that the manager is required to establish login’s account for each staff in order to enable them using the system. Each on-duty staff (waiter/waitress) has to capture customer’s order from each table. Other than that, each on-duty staff (cashier) has to calculate the total order’s amount for each table. Finally the manager is required to generate a bar graph to show the distribution of the best selling menu for each day.

**2.2 Problem Decomposition**

The problem is divided into several parts as follows:

1. Establish login’s account for each staff
2. Manage menu description from hardcopy menu
3. Create table description according to the restaurant’s table
4. Compute and generate daily sales of restaurant
5. Determine the best selling menu for each day
6. Capture order for each table
7. Calculate total order’s amount for each table

**2.3 Structure Chart**



**Chapter 3: Design**

**3.0 Data Model (ERD)**



**Table above show the data dictionary for the ERD.**

**3.1 Data Dictionary**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Entity** | **Entity Description** | **Column Name** | **Column Description** | **Data Type** | **Length** | **Primary Key** | **Nullable** |
| Staff | Store the Staff Information for Login to the system | Staff\_ID | ID for each staff | Char | 6 | / |  |
| Staff\_Name | Name for each staff | Varchar | 45 |  |  |
| Staff\_Position | Position for each staff | Varchar | 10 |  |  |
| Password | Password for staff login | Int | 11 |  |  |
| Table | Store the Table Information in the restaurant | Table\_ID | ID for each table | Char | 3 | / |  |
| Capacity | Number of seat in each table | Int | 11 |  |  |
| Table\_Status | Status for each table (Available or Unavailable) | Varchar | 15 |  |  |
| Menu | Store the Restaurant Menu Information | Menu\_ID | ID for each food/drink/dessert in menu | Char | 5 | / |  |
| Menu\_Name | Name for each food/drink/dessert in menu | Varchar | 45 |  |  |
| Unit\_Price | Price per serving for each food/drink/dessert in menu | Decimal | (10,2) |  |  |
| Menu\_Status | Status for each food/drink/dessert in menu (Available or Unavailable) | Varchar | 15 |  |  |
| Order | Store the Customer Order Information | Order\_ID | ID for each order | Int | 11 | / |  |
| Date | Current Date for the current order | Date | - |  |  |
| Time\_In | Current time for the current order | Time | - |  |  |
| Order\_Status | Status for each order (Paid/Unpaid) | Varchar | 10 |  |  |
| Amount | Amount for each order | Decimal | (10,2) |  | / |
| Order\_Detail | Store the Customer Order Product Quantity | Order\_Detail\_ID | Autogenerate number in database | Int | 11 | / |  |
| Order\_Quantity | Quantity for each food/drink/dessert | Int | 11 |  |  |
| Order\_Menu\_Status | Current Status for each ordered food/drink/dessert (Waiting or Done) | Varchar | 7 |  |  |

**3.2 Data Flow Diagram**

*Context Diagram*



*Level 0 Diagram*



*Level 1 Diagram : Process 1 – Login*



*Level 1 Diagram : Process 2 – Manage Staff Login Account*



*Level 1 Diagram : Process 3 – Manage Menu*



*Level 1 Diagram : Process 4 – Manage Table*



*Level 1 Diagram : Process 5 – View Daily Best Selling Menu*



*Level 1 Diagram : Process 6 – Print Daily Sales Report*



*Level 1 Diagram : Process 7 – Manage Customer Order*



*Level 1 Diagram : Process 8 – Payment*



|  |  |  |
| --- | --- | --- |
| **Data Name** | **Data Description** | **Data Structure** |
| Best Selling Menu info | Best Selling Menu information | Date + Menu\_ID + Menu\_Name + Table\_ID + Order\_ID + Order\_Quantity |
| Bill | Customer bill | - |
| Calculated bill | Return balance if any | - |
| Cash amount | Cash amount of customer | - |
| Customer order | Information of each Customer order | Table\_ID + Order\_ID + Staff\_ID + Date + Time\_In + Menu\_ID + Order\_Quantity |
| Daily Sales info | Daily Sales information | Date + Menu\_ID + Menu\_Name + Unit\_Price + Order\_Quantity |
| List of order info | A list of current order information | Menu\_ID + Order\_Quantity |
| List of order menu info | A list of current order menu information | Table\_ID + Staff\_ID + Date + Time\_In |
| Login info | Login information | Staff\_ID + Password |
| Login info validation | Check whether the username and password match to database | Staff\_ID + Password |
| Login Status | Status for login to the system | Boolean |
| Login Success/Fail Message | Message about either login success or fail | "OK" |
| Menu info | Information of each food/drink/dessert in menu | Menu\_ID + Menu\_Name + Unit\_Price + Menu\_Status |
| New Cus order info | Information for new customer order | Table\_ID + Order\_ID + Staff\_ID + Date + Time\_In + Menu\_ID + Order\_Quantity |
| New menu info | Information for new menu | Menu\_ID + Menu\_Name + Unit\_Price + Menu\_Status |
| New staff info | Information for new staff | Staff\_ID + Staff\_Name + Staff\_Position + Password |
| New table info | Information for new table | Table\_ID + Capacity + Table\_Status |
| Order info | Information of each | Time\_In + Table\_ID + Order\_ID + Menu\_Name + Order\_Quantity |
| Order\_Detail info / Order menu info | Information of each ordered | Menu\_ID + Menu\_Name + Order\_Quantity |
| Saved Cus order info | Saved information of current order | Table\_ID + Staff\_ID + Date + Time\_In |
| Saved menu info | Stored menu information | Menu\_ID + Menu\_Name + Unit\_Price + Menu\_Status |
| Saved order menu info | Saved information of current ordered menu | Menu\_ID + Order\_Quantity |
| Saved staff info | Stored staff information | Staff\_ID + Staff\_Name + Staff\_Position + Password |
| Saved table info | Stored table information | Table\_ID + Capacity + Table\_Status |
| Staff info | Information of each staff | Staff\_ID + Staff\_Name + Staff\_Position + Password |
| Table info | Information of each table in restaurant | Table\_ID + Capacity + Table\_Status |
| Updated menu info | Updated information for current menu | (Menu\_ID) + (Menu\_Name) + (Unit\_Price) + (Menu\_Status) |
| Updated staff info | Updated information for current staff | (Staff\_Name) + (Password) |
| Updated table info | Updated information for current table | (Table\_ID) + (Capacity) + (Table\_Status) |
| Updated table status | Updated table status for current using table | Table\_Status |
| Validated Cus order info | Checked customer order information | Table\_ID + Order\_ID |
| Validated menu info | Check whether the Menu ID is duplicated or not | Menu\_ID |
| Validated new staff info | Check whether the Staff ID is duplicated or not | Staff\_ID |
| Validated Order ID | Generate the new Order ID | Order\_ID |
| Validated table info | Check whether the Table ID is duplicated or not | Table\_ID |
| Verified username,password | Return username and password | Staff\_ID + Password |

**Table above show the data dictionary for the DFD.**

**3.3 Flow Chart**

*Start Menu*



*Login\_Page.vb*



*Manager\_Use.vb*



*KitchenDisplay.vb*



*KitchenLogin.vb*



*Kitchen\_Use.vb*



*WaiterCashier\_Use.vb*



*Order.vb*



*Order\_Product.vb*



*Table\_info.vb*



**3.4 Pseudo Code**

1.0 Start

2.0 Call Login\_Page()

2.1 Validate username and password

2.2 If username and password correct

2.2.1 Display “Username and password are wrong.”

2.3 Else

2.3.1 If Staff\_Position = “Manager”

2.3.1.1 Call Manager\_Use()

2.3.2 Else if Staff\_Position = “Waiter” or Staff\_Position = “Cashier”

2.3.2.1 Call WaiterCashier\_Use()

2.3.3 Else if Staff\_Position = “Kitchen”

2.3.3.1 Call KitchenDisplay()

2.3.4 Else

2.3.4.1 Display “You have no assign any position. Please contact your manager.”

3.0 Call Manager\_Use()

3.1 Validate Staff info, Menu info, Table info

3.2 Display Staff info, Menu info, Table info

3.3 If Date Clicked

3.3.1 Display Best Selling Report

3.4 Else if Report Clicked

3.4.1 If Date Clicked

3.4.1.1 Display Daily Sales Report

3.4.1.2 Print Daily Sales Report

4.0 Call WaiterCashier\_Use()

4.1 If NewOrder Clicked

4.1.1 Call Order()

4.2 Else

4.2.1 Validate Curr\_Table\_ID

4.2.2 If Load Clicked

4.2.2.1 Call Order()

4.2.3 Else

4.2.3.1 Validate Cash\_Amount

4.3 Calculate Change

4.3.1 If Cash\_Amount > Total\_Due

4.3.1.1 Display Change

4.3.1.2 Print Bill

4.3.2 Else if Cash\_Amount = Total\_Due

4.3.2.1 Display Change

4.3.2.2 Print Bill

4.3.3 Else if Cash\_Amount < Total\_Due

4.3.3.1 Display “Not enough cash.”

5.0 Call KitchenDisplay()

5.1 View Order\_Detail info

5.2 If Update click

5.2.1 Call KitchenLogin()

6.0 Call KitchenLogin()

6.1 Validate username and password

6.2 If username and password correct

6.2.1 Display “Username and password are wrong.”

6.3 Else

6.3.1 If Staff\_Position = “Kitchen”

6.3.1.1 Call Kitchen\_Use()

7.0 Call Kitchen\_Use()

7.1 Update Menu\_Status

7.2 Display “Data Updated.”

8.0 Call Order()

8.1 Validate order info

8.1.1 Call Table\_info()

8.1.2 Call Order\_Product()

8.2 Calculate Subprice, Subtotal, GovTax, SvcCharge, Total

8.3 Display Subprice, Subtotal, GovTax, SvcCharge, Total

9.0 Call Table\_info()

9.1 Return Table\_ID

9.2 If Ok Clicked

7.2.1 Call Order()

10.0 Call Order\_Product()

11.1 Calculate Subprice

11.2 Return Menu\_ID, Order\_Quantity, Subprice

11.2 If Ok Clicked

8.2.1 Call Order()

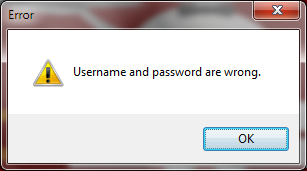
12.0 End

**Chapter 4: Implementation**

1. Main page is the User Login page: Different user login to different page.



When username and password entered are wrong:

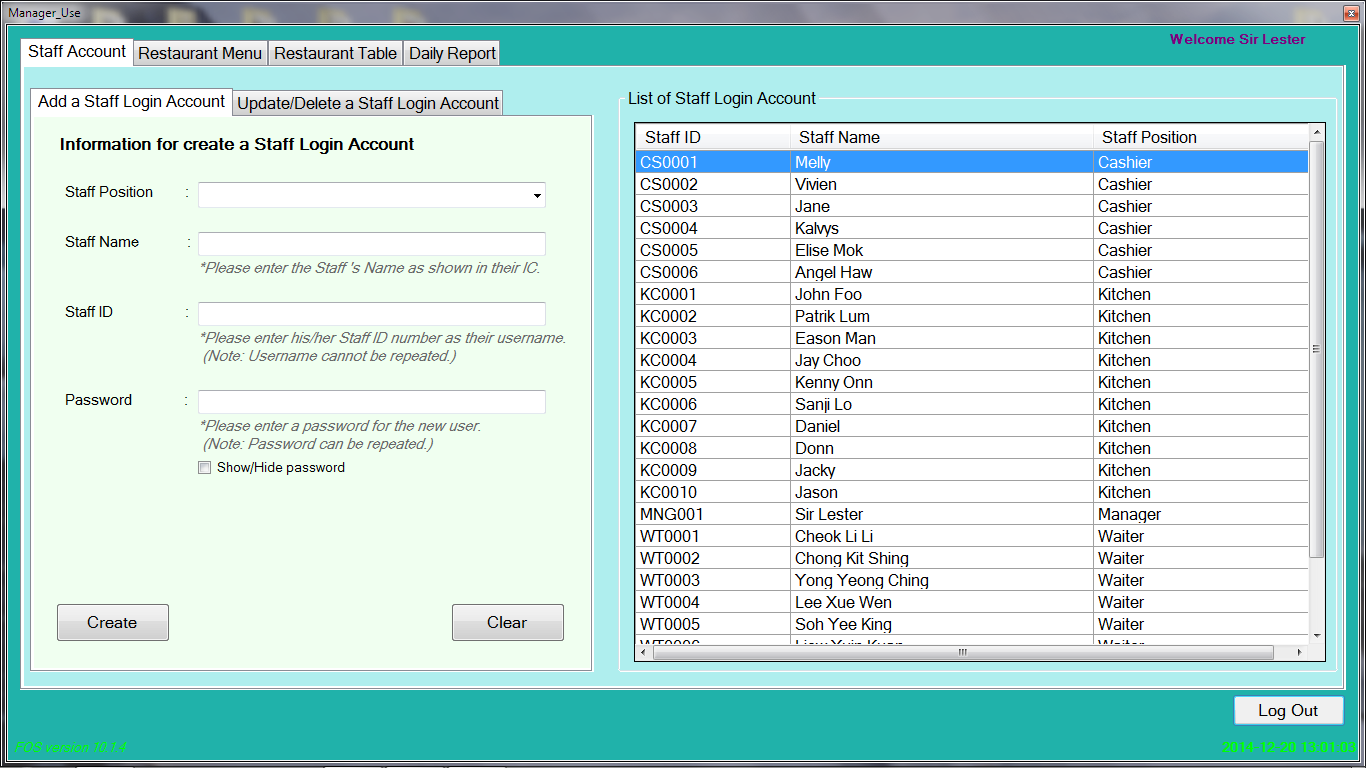


2. After Restaurant Manager Login:

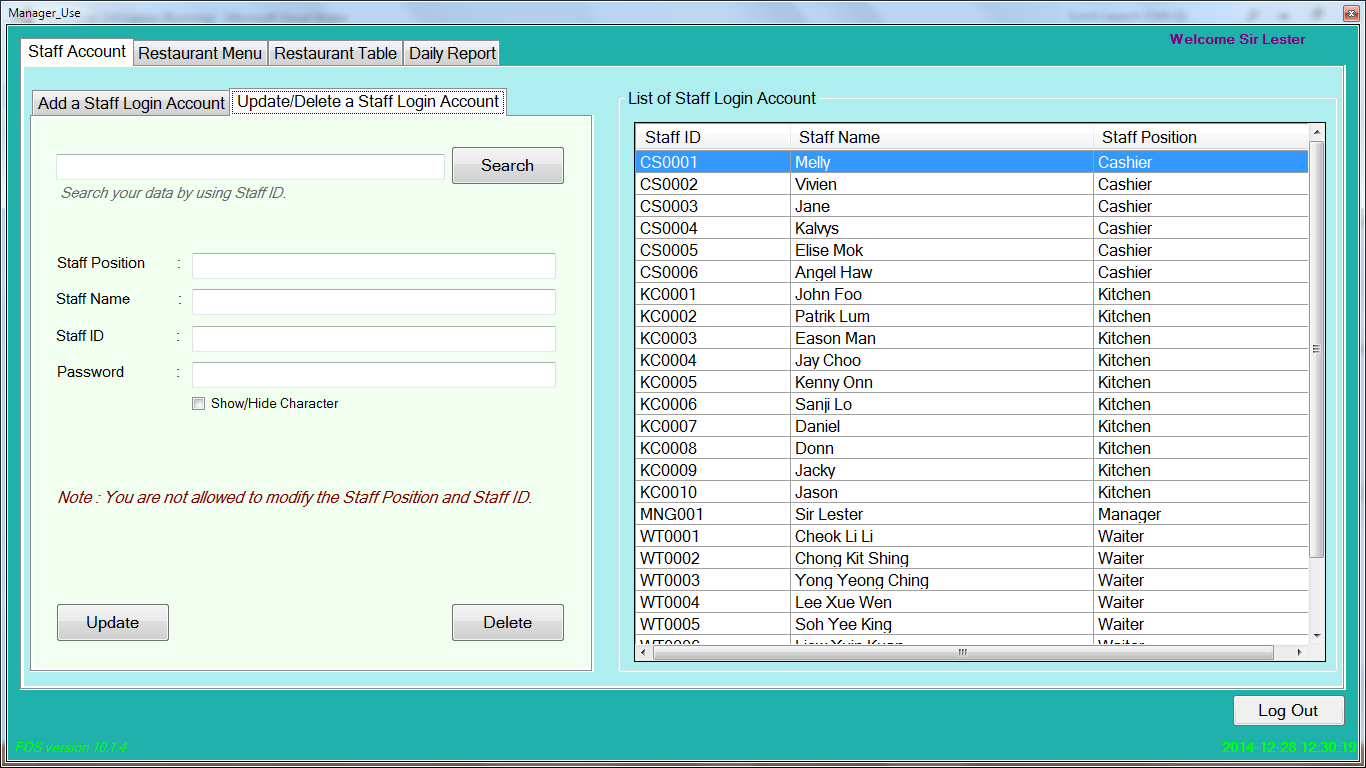
Note:

The pages after manager login contains 4 tab pages, there are Staff Account, Restaurant Menu, Restaurant Table and Daily Report.

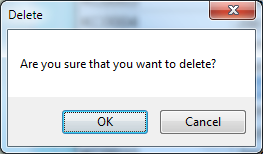
A) Staff Account: Record and show all the Staff Login Account Information.



The above page is for proceed a work about adding a new staff login account for this system.

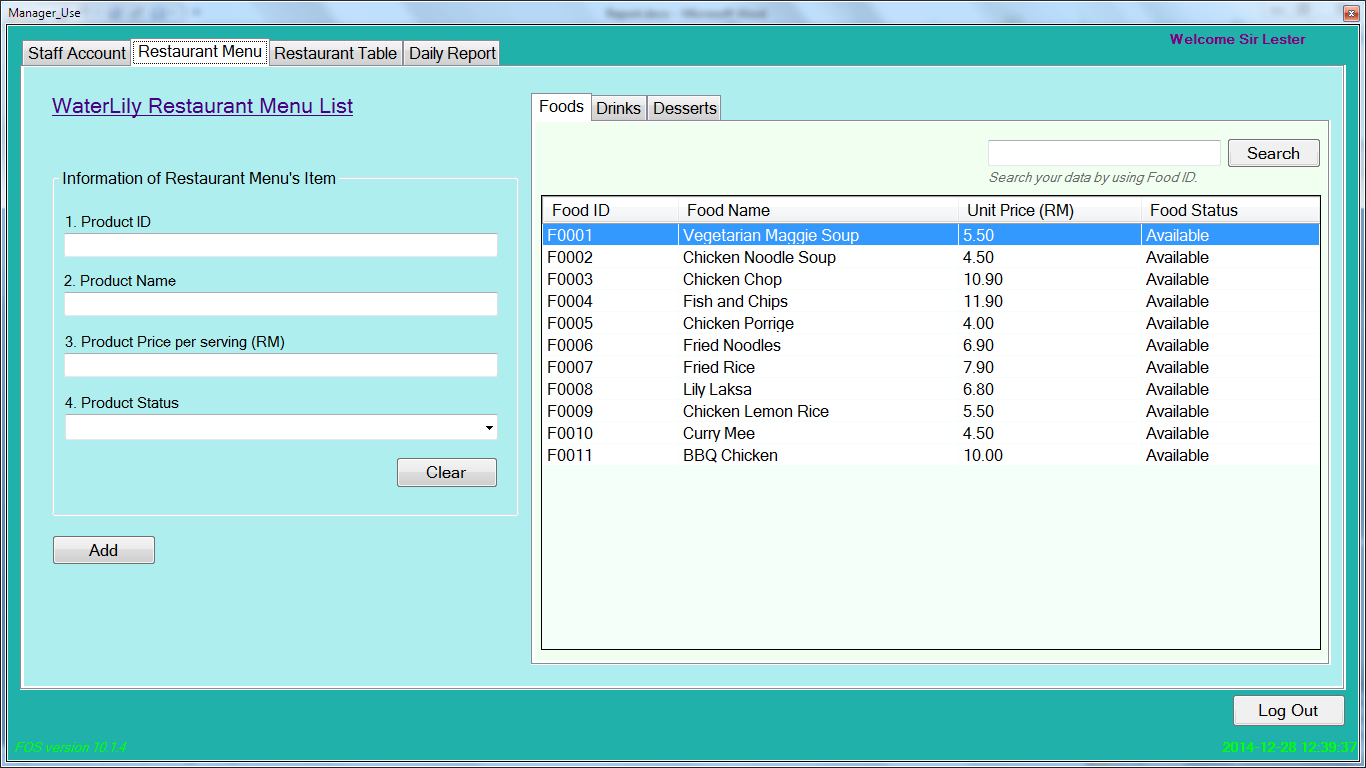


While the above page is for carry out a work about update or delete a current staff login account.



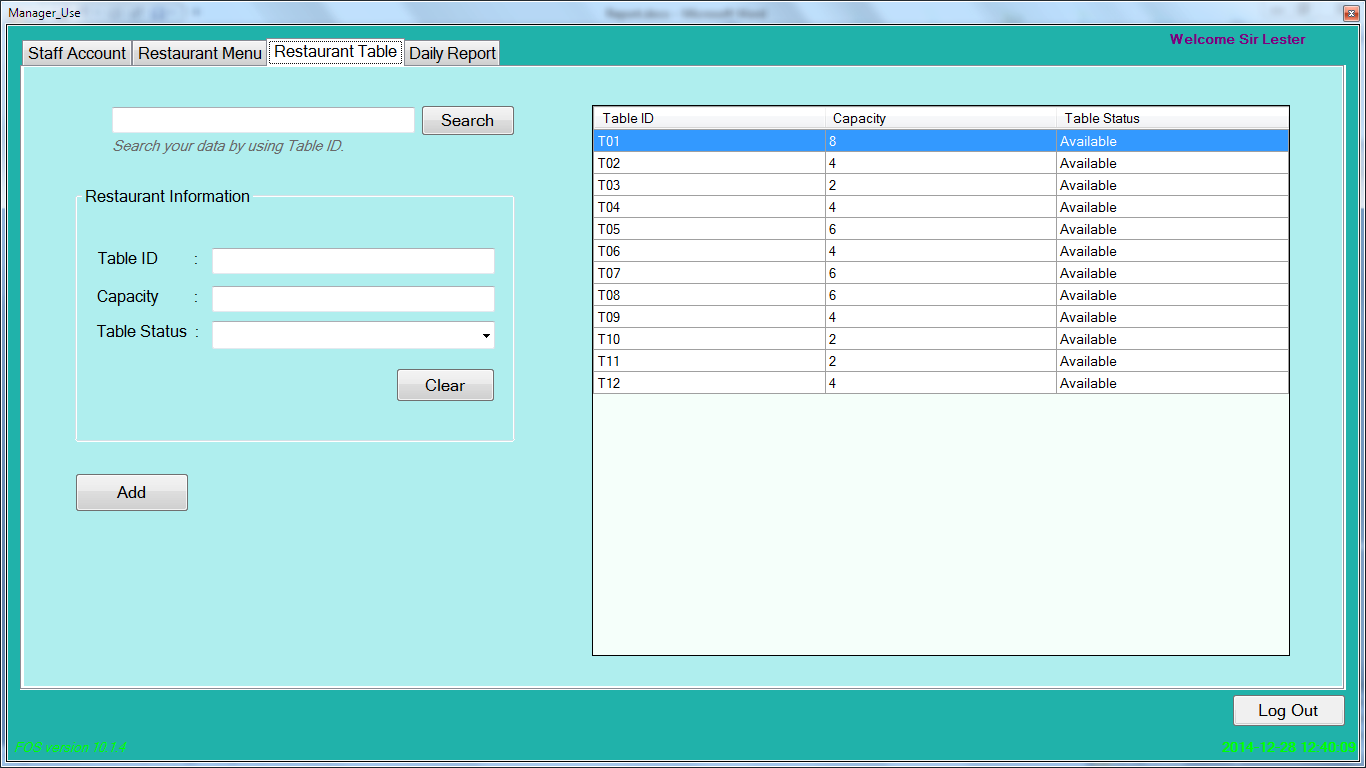
A confirmation message will pop out when “Delete” button in the 1st three tab pages is clicked.

B) Restaurant Menu: Record and show all the Restaurant Menu Information.



The “Update” and “Delete” button only show when the “Search” button click.

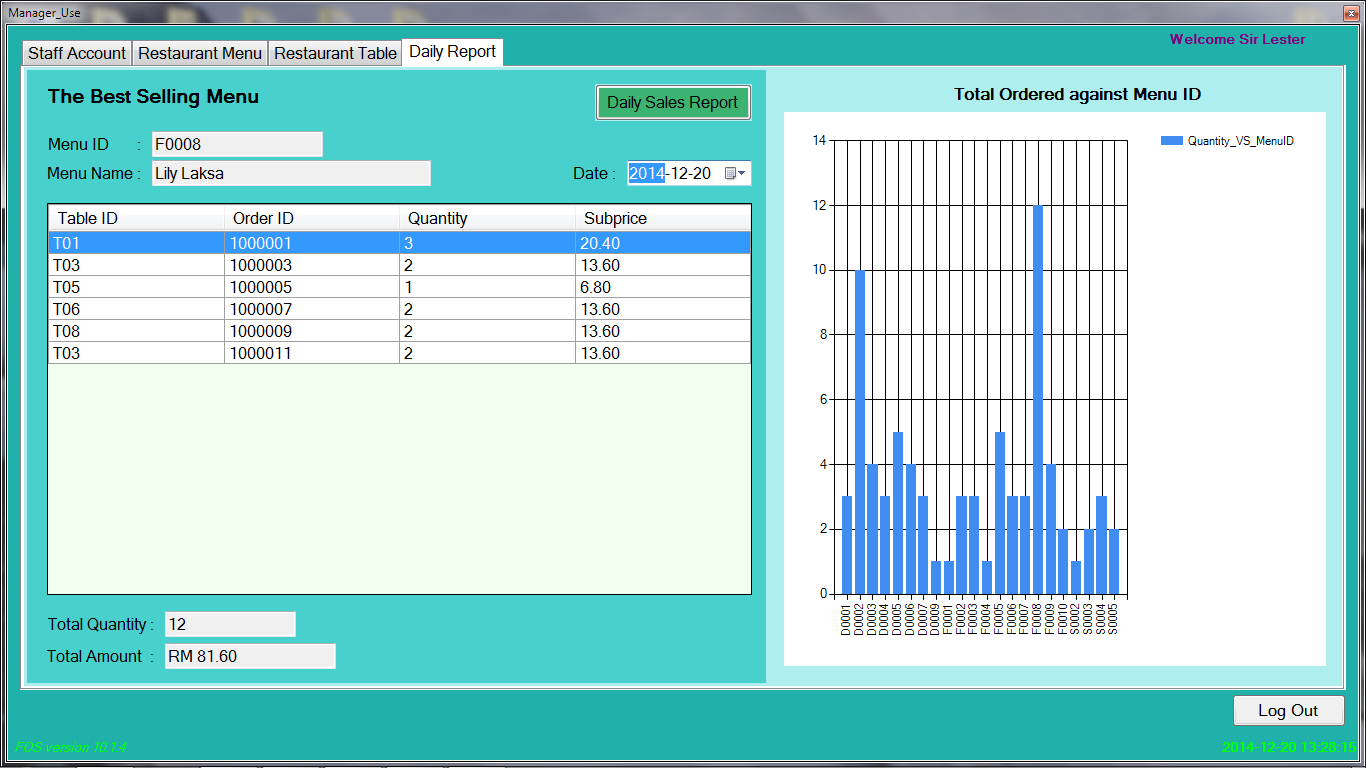
C) Restaurant Table: Record and show all the Restaurant Table Information.



The “Update” and “Delete” button only show when the “Search” button click.

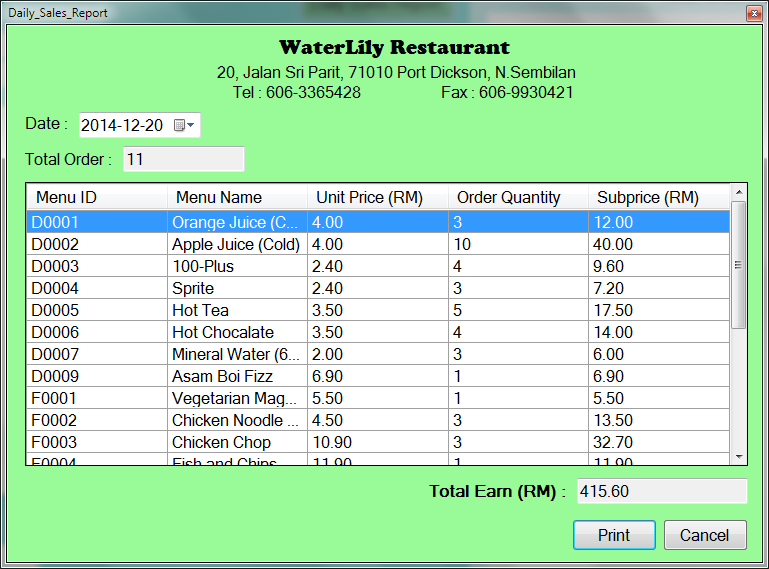
D) Daily Report: Record the information of The Best Selling Menu and Daily Sales Report.

Below show The Best Selling Menu Report:

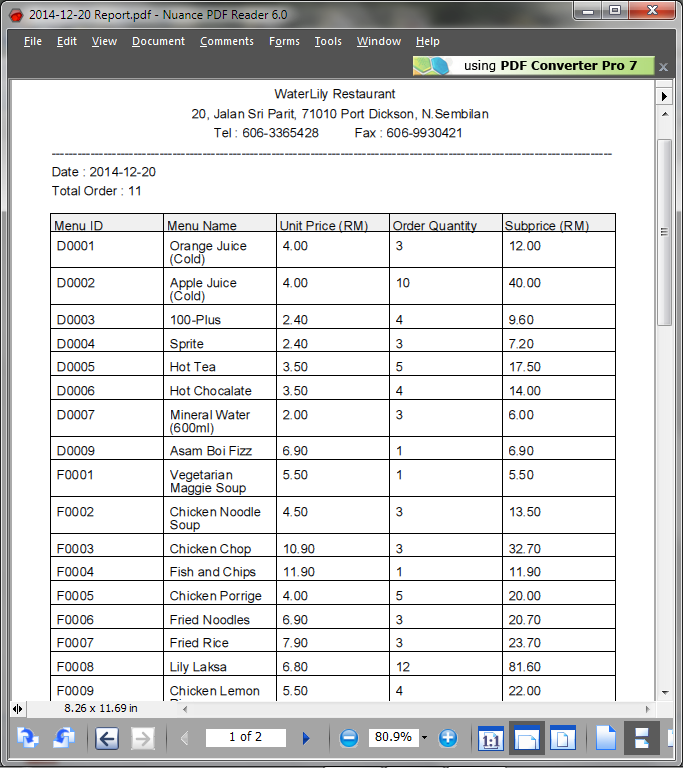


This only show the result for a current date, print is not provided for this report.

Below show the Daily Sales Report:



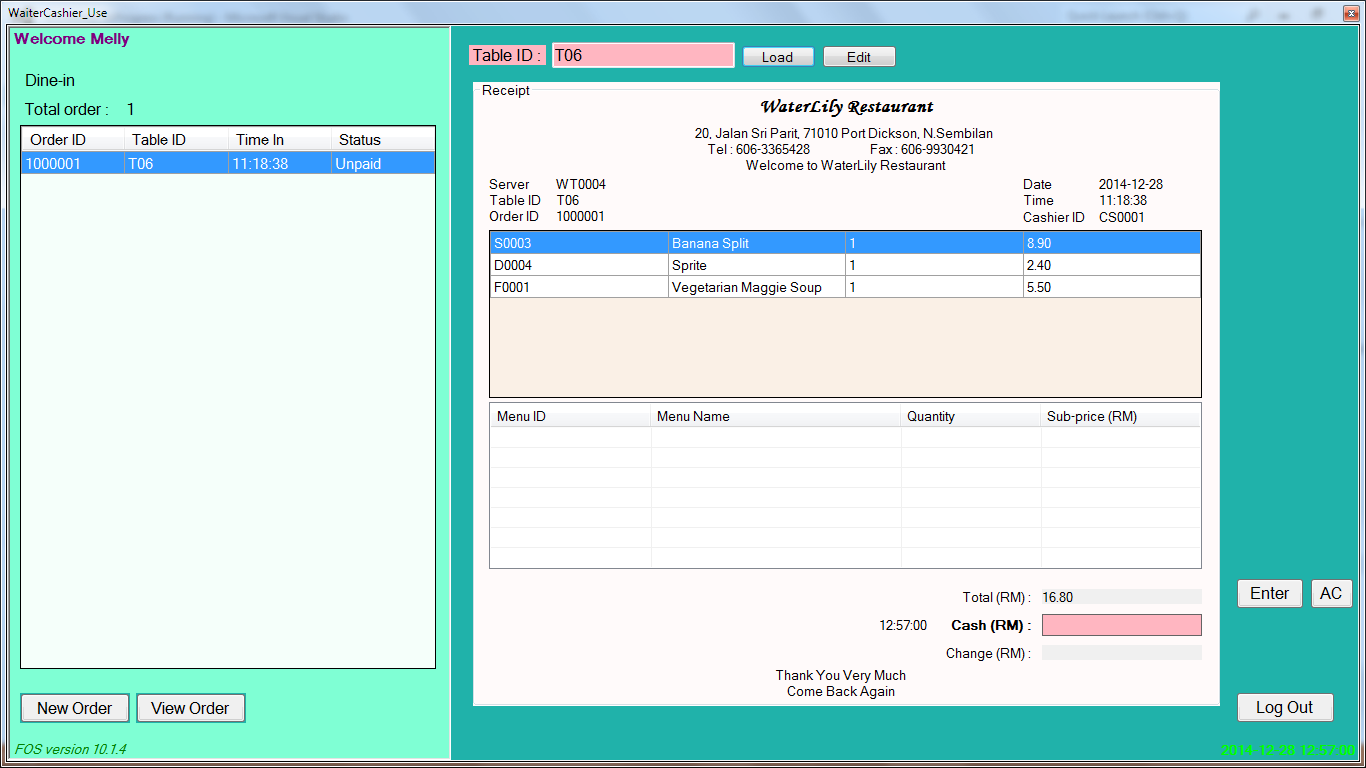
While the “Print” button is clicked, the Daily Sales Report will be printed to pdf file:



3. After Waiter and Cashier login, they are to the same page, but Waiter cannot make payment for customer.

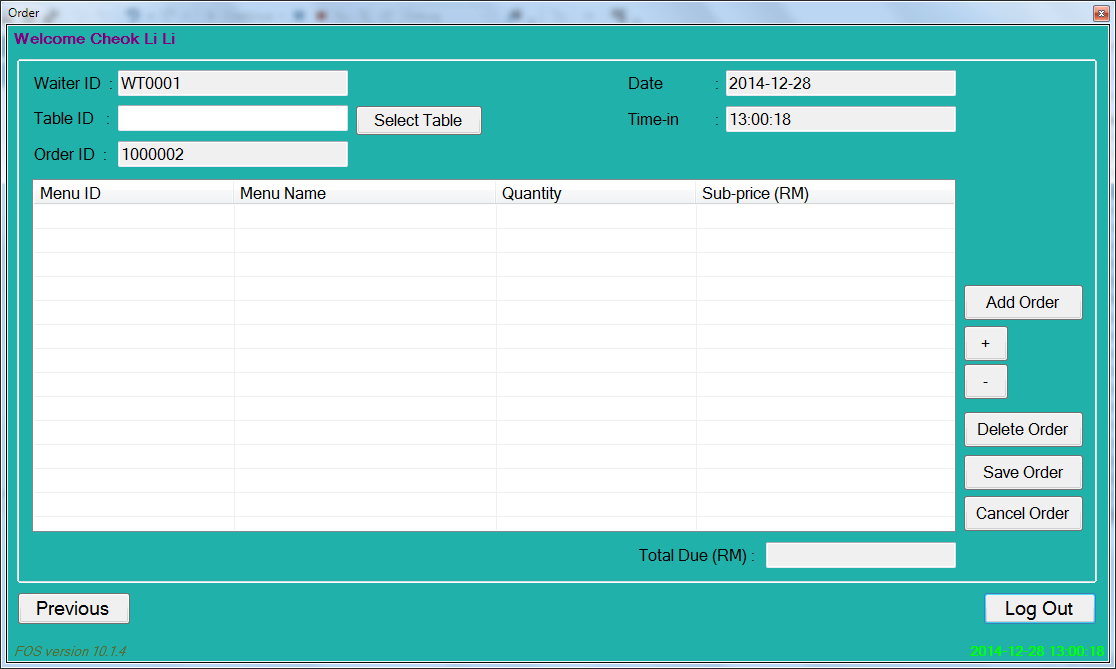
Note: The “Enter” and “AC” buttons are not show after waiter login.

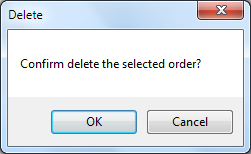
(AC = All Clear, Clear all the value in the Cash textbox)



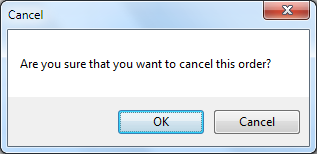
The “Edit” button is for add the new foods/drinks/desserts to the current order.

Below show the order form for waiter order for customer:



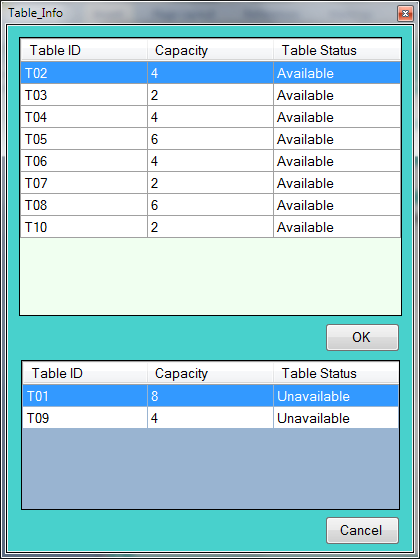


A confirmation message will pop out when “Delete Order” button is clicked.

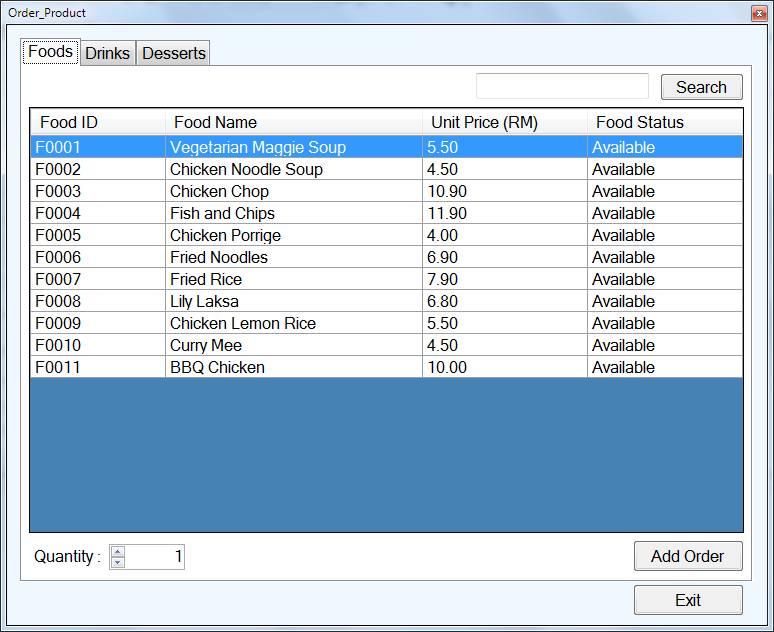


A confirmation message will pop out when “Cancel Order” button is clicked.

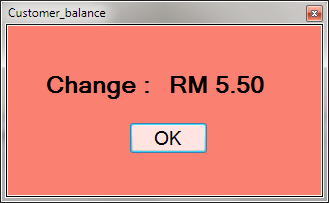
After “Select Table” button is clicked, choose a Table before order for foods/drinks/desserts:



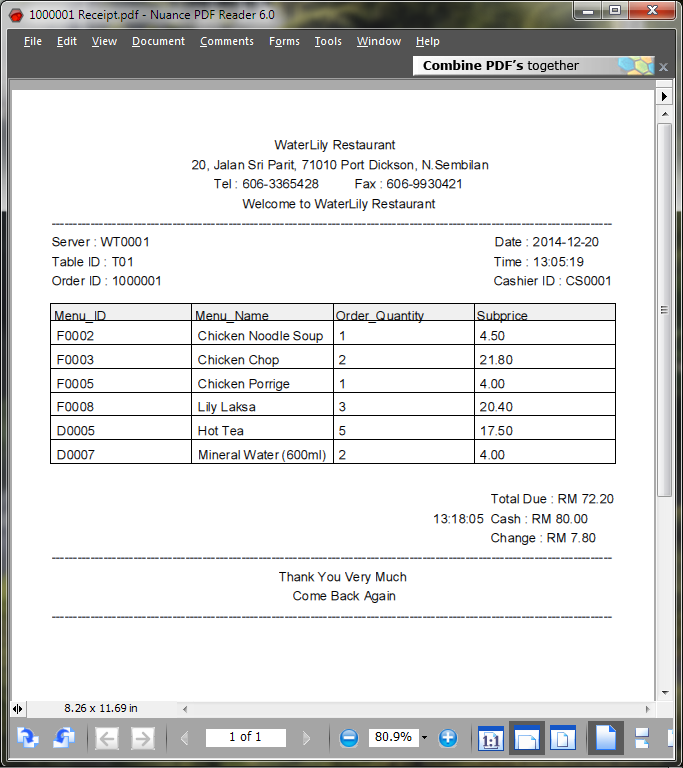
After “Add Order” button is clciked, choose Menu(foods/drinks/desserts) when order:



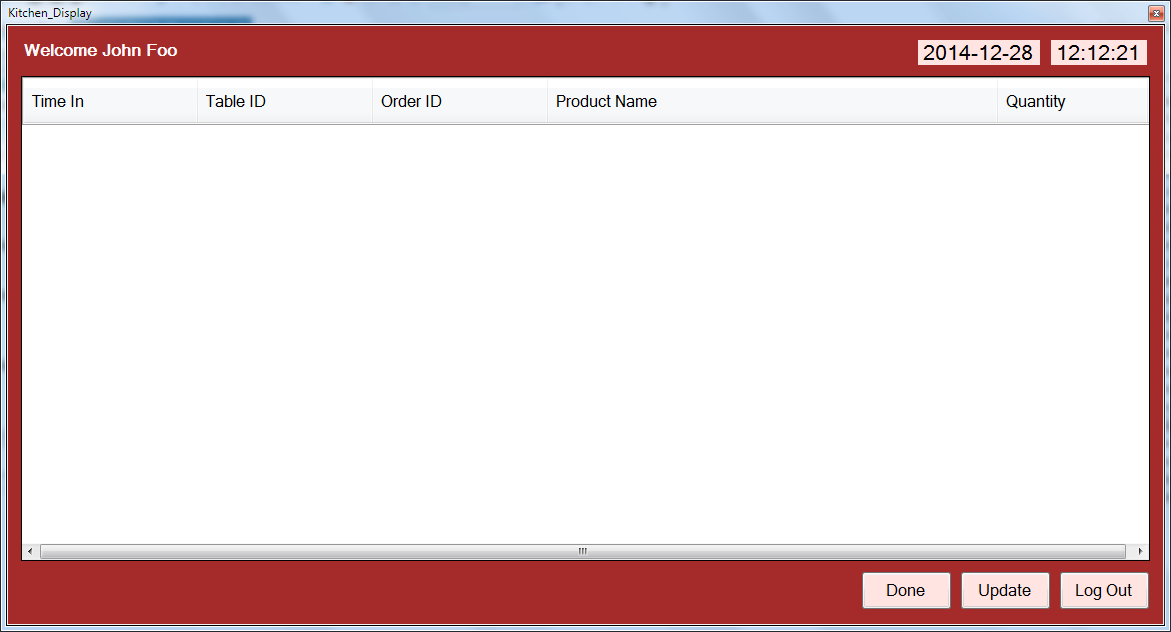
For Cashier, after “Enter” button is clicked, a message for the current customer change will show:



At the same time, customer bill is print to pdf file:

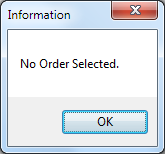


4. After Kitchen Login:



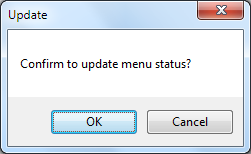
The above page shows the detail for the current ordered foods/drinks/desserts.

An order must be selected before press “Done” button, otherwise a message will pop out informed that “No order selected.”



If an order selected from the table, and click “Done” button , there will be a DingDong sound to inform the waiter about that there have an ordered foods/drinks/desserts is finished cook, and the selected order will be remove from the table.

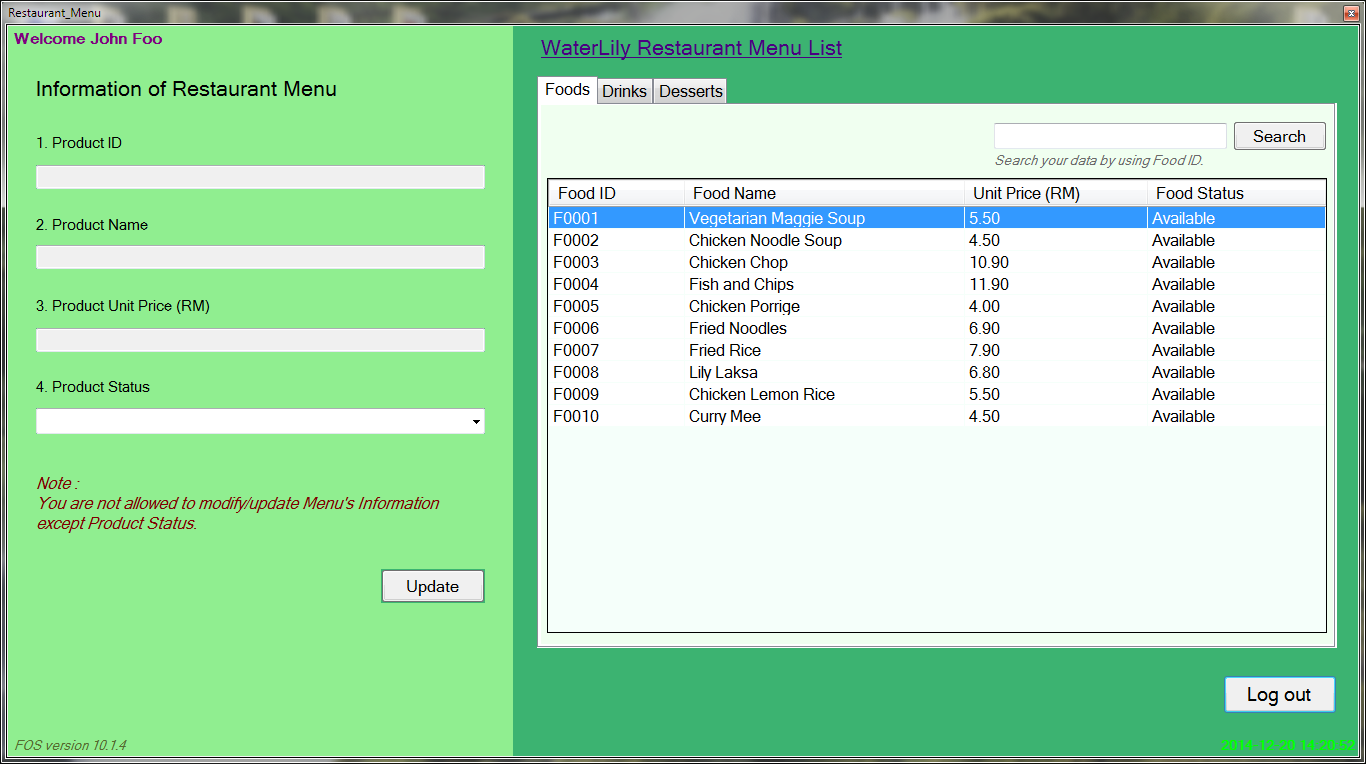
While the “Update” button will give them this page:



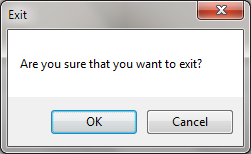
A login page will show for the second login,



After Login, the below page will show to update the product status (Available/Unavailable).



5. Logout



**Chapter 5: Conclusion**

In short, Food Ordering system is completed created and produced within the given time. There have some problem while designing and implementing the program, such as FOS interface design. The big problem that often encountered is visual basic coding. While there has error on the coding, this problem has been solved by searching and watching video on YouTube, or finding the help from others like course mate, lecturer and senior who are familiar to this language.

For overall, many parts of this Food Ordering System need to improve. Besides, the sales report of the restaurant is designed for daily only, the system need to contain also the weekly, monthly and yearly sales report to view more detail restaurant sales. Furthermore, the login system is less security because the user can login to the system by many times without giving any warning message like “Your account has been locked. Please be waiting for 30 seconds to login the system again.” or lock the current user account. If can, this FOS system is suggested to upgrade to a mobile application for the ordering part.