



**SCHOOL OF QUANTITATIVE SCIENCES  
UUM COLLEGE OF ARTS AND SCIENCES  
UNIVERSITI UTARA MALAYSIA**

**SQIT3043 DATA MODELING AND DATABASE  
GROUP A  
FIRST SEMESTER SESSION 2021/2022 (A211)**

**GROUP PROJECT**

Submitted to:

**DR. MAZLAN BIN MOHD SAPPRI**

Prepared by:

<b>MATRIC NUMBER</b>	<b>NAME</b>	<b>I/C NUMBER</b>
277378	TAN HUI LING	000317-01-0136
277381	ENG JINQ KAE	000317-10-1329
278054	CHIN HAO QIAN	000531-07-0447
279817	CHEAH RUI MING	001206-07-0106
280027	LEE JUAN	010101-10-1185

**Table of Contents**

No	Content	Pages
<b>1.0</b>	Introduction	2
<b>2.0</b>	Problem Statement	4
<b>3.0</b>	Project Objectives	6
<b>4.0</b>	Project Scope	7
<b>5.0</b>	Difference Between Proposal and Group Project	9
	5.1 Entities	9
	5.2 Attributes	10
	5.3 Relationships	11
<b>6.0</b>	Data Model	13
<b>7.0</b>	Flow Chart	16
<b>8.0</b>	Database	17
	8.1 Database Table	17
	8.2 Input Data Testing	20
	8.3 Relationships	25
	8.4 Normalization	26
<b>9.0</b>	Interface and Coding	34
<b>10.0</b>	Conclusion	123
<b>11.0</b>	Reference	124

## 1.0 Introduction

UUM Cafe Restaurant is a western food restaurant founded by a group of undergraduate students from the School of Quantitative Sciences in 2021. This restaurant is located in Universiti Utara Malaysia (UUM), Sintok, Kedah and sells more than 30 menus of western food, desserts and beverages such as escargot, Bolognese, churros, tacos, coffee and milk tea, just to name a few. We decided to start this business because we hope that all UUM residents have a chance to enjoy delicious western food anywhere inside UUM. Therefore, other than dine in, we offer online ordering for customers to order on their mobile devices, which will then have their food delivered to them by delivery boys too. Besides, we will create our own database system, which will record UUM residents' details and orders. If UUM residents dine in or use our delivery service more than 10 times, they will automatically become our loyal customers and enjoy 8% discount rewards. Other than that, our database system also records our sales and employee information to aid our top management to manage employees easily and systematically. Lastly, we will always update our latest information on our Facebook and Instagram accounts.

We decided to develop a database system for UUM Cafe restaurant because there are many problems we might face if we use traditional methods to manage and store data. Firstly, it will waste many resources as existing systems use heaps of paperwork such as bills, at the same time it brings upon a problem of occupying large amounts of space in our office areas. Thirdly, UUM Cafe restaurant is also facing the problem of lacking human resources to manage and store our paperwork. Besides, it will take a lot of time to find our data if we do not use a database system. Lastly, recording a customer's data using traditional methods may lead to the customer's data not being updated. A new database system allows us to edit our customer's data by just logging into the system and clicking on the edit button.

Next, UUM Cafe restaurant uses an object-based data model to describe and manipulate our data, relationships between data, and constraints on the data. In addition, this data model is used to make the data more understandable because it uses concepts such as entities, attributes and relationships. This database system will be used by our top management and staff to record business transactions, collect sales, prepare and present our business report in a proper way. The database system and Windows Form Application for our business will be developed using Microsoft Visual Basic and Microsoft Access. Inside our object-based data model, we have 11

entities needed to represent our database. Firstly, we must have a Customer as our first entity. At the same time, we have the CafeMenu as our second entity, which will record our food and beverage provided. Order, Discount, and Payment also as a distinct object in our business is to be represented in the database. There are Staff and DeliveryBoy as our entities because we need some workers to run everyday activities. Moreover, we also need an Area as our entity to represent which area or what address we deliver our food and beverage to. Then, we have Attendance and ManagementPage as our entity to record duty time and revenue.

To develop a new database system for UUM Cafe restaurant, first we need to have a database planning to make sure the stages of the database system development life cycle is to be realized as efficiently and effectively as possible. Secondly, we need to give a system definition which describes the scope and boundaries of the database system and the major user views. After that, we need to collect and analyse information about parts of the UUM Cafe restaurant. Subsequently, we will design our database. This is the process of creating a design that will support the UUM Cafe restaurant's mission statement and mission objectives for the required database system. After we design our database, we will design our application. In this process, the user interface and the application programs that use and process the database are designed. Our next step is prototyping. We will be building a working model of the database and this process allows users to use it to identify the features of the system that work well or are inadequate. The next step will be implementation. This process' main activity is creating the physical database definitions and the application programs. For the data conversion and loading process, we will be transferring any existing data into the new database and converting any existing applications to run on the new database. Next, we will run the database system with intent of finding errors then fix all the errors and bugs that we have found. The last step is operation and maintenance. This is the process of mentoring and maintaining the database system following installation. The monitoring process will continue throughout the life of the database system.

## 2.0 Problem Statement

In this current COVID-19 pandemic, managing and storing in the UUM Cafe restaurant is less efficient, plus there is no social distancing between customers while ordering food. UUM Cafe has not developed an application or system for receiving orders from customers, storing and checking sales records and other data. UUM Cafe restaurant found that the existing system faces five problems.

First, the UUM Cafe restaurant faces the problem of wasting resources. Existing system from many paperwork such as cash bills, invoices, documents, etc. Hence, the business needs to expand a lot on this aspect. The problems will increase the cost of resources and reduce the business's profit. Second, the UUM Cafe restaurant faces the problem of occupied space. The document files occupy a big space and are untidy. When the business wants to prepare the new room to store the documents, it needs to increase the rental cost. Third, the UUM Cafe restaurant faces the problem of needing more human resources. Existing systems have the issue that manual systems need a lot of human resources. More human resources are required to manage the massive piles of files in handling forms and redundancy. Finding data, writing it down, dealing with customers, taking orders, processing orders and so on all require many human resources. Fourth, the UUM Cafe restaurant faces the problem of finding information that is troublesome and time-consuming. There are far too many documents to keep track of the customer's records. Even though the files have been organized, the manual method can result in errors, such as incorrectly arranging the files, making it difficult to locate records, and having insufficient storage space for the files, making it time-consuming to find files from one shelf to the next. Fifth, the UUM Cafe restaurant faces the problem of the customer's data not being updated. The personal information of the customer did not key in the computer data. The problems will be wasting the time to update customer's personal information. It shows that the business is outdated.

An application was developed to facilitate customers for dine in, take away, and delivery at UUM Cafe restaurant. It will reduce the risk of infection in this Covid-19 pandemic and maintain social distancing between customers. Database information systems help staff to store the large numbers and data of records. The database is one of the software that is easier to handle by someone and saves more time to store, record the data, find the information and reduce the human resources. Database information systems will control data redundancy, and it will avoid

occupied space. Database will be constantly updated to ensure that it is not outdated and has improved security.

### **3.0 Project Objectives**

The objective for the *UUM Cafe restaurant* in terms of establishing a database to support the business is as following:

1. To design the database system to track down the business transaction.
2. To record, modify, delete, and search business transactions by using a database information system.
3. To prepare the business report in the way by using a database information system.

## 4.0 Project Scope

This software is designed to ease the business operation of UUM Cafe by lessening all the extra paperwork. Getting customers information, taking orders, delivering the foods if requested and others. This software will be launched in UUM Cafe first, then it will soon be spread to the whole country. Our potential customer is every cafe owner, or even restaurant owner. As our software also involves the delivery boys, this is the trend in the market nowadays, and also the advantage of our software. Hence, the size of this system will be getting bigger, but maintained in one software. When logged in to the application, the manager, staff, and customers will receive different interfaces and functions. In focus, this software is to receive orders from customers, record customers' details, make the customers pay easier, and delivery boys can send the foods and beverages.

The end users of this software are the customers of the cafe, and the cafe owners, who buy this software, are just the middle user of this software and act as the admin. Hence, the users can be the customers of the cafe, the managers, the staff and the delivery boys. The customers will be able to check the menu and make an order or a reservation. They can find the needed information faster. This software will be accessed online to reach real-time programming so that the users mentioned above can check the data online quickly. They can also update their details and information and these changes will be automatically updated as it is accessed online. Besides, this can also lessen the human resources and avoid errors made by the staff.

This software will be developed by Microsoft Visual Basic to design the application, and also Microsoft Access to store the data. The Microsoft Visual Basic contains many necessary functions to develop an information system that can handle different requests by the customers. This makes the capability of real-time programming that we have targeted. It can design the interface of a software easily and it tells the error in the software in a very specific way. The developer and engineer can check and fix the errors quickly. Besides, Microsoft Access will also contain many tables to store a variety of data. It can also connect the relationship between all entities so each of the tables is related. Microsoft Visual Basic is suitable to integrate with different database systems, for instance Microsoft Access. These two programmes can collaborate well to develop a huge information system software. Eventually, the customer and staff will be able to use the fully developed application.

This software will take 3 months to develop the database and application, and 2 weeks of testing. The cost of this software is estimated at RM10,000 to buy the Microsoft Visual Basic and Microsoft Access. There will be 1 project manager, 2 database engineers and 2 software engineers to build this application.

## 5.0 Difference Between Proposal and Group Project

Although we always used our proposal as our guideline, there are still some changes we made when doing our group project. Therefore, we need to clarify what we have changed to reduce confusion.

### 5.1 Entities

First, the number of entities in our proposal is 10. However, we have 11 entities in our group project. Besides the number of entities are different, most of the entities are different from our proposal. In our proposal, we have Customer, CafeMenu, Order, Discount, Payment, Staff, DeliveryBoy, Area, Attendance and ManagementPage but entities in our group project are Customer, CustomerAccount, CustomerOrder, Feedback, Payment, PromoList, PromoPayment, Rider, RiderAccount, Staff and StaffAccount. There are only 3 entities that are the same as the proposal, which are Customer, Payment and Staff.

Proposal	Group Project
<ol style="list-style-type: none"> <li>1. Customer</li> <li>2. CafeMenu</li> <li>3. Order</li> <li>4. Discount</li> <li>5. Payment</li> <li>6. Staff</li> <li>7. DeliveryBoy</li> <li>8. Area</li> <li>9. Attendance</li> <li>10. ManagementPage</li> </ol>	<ol style="list-style-type: none"> <li>1. Customer</li> <li>2. CustomerAccount</li> <li>3. CustomerOrder</li> <li>4. Feedback</li> <li>5. Payment</li> <li>6. PromoList</li> <li>7. PromoPayment</li> <li>8. Rider</li> <li>9. RiderAccount</li> <li>10. Staff</li> <li>11. SatffAccount</li> </ol>

## 5.2 Attributes

Although the remaining 3 entities are the same as the proposal, the attributes are different in the group project. First, for the Customer, we have 8 attributes, which are CustomerID, CustomerUsername, CustomerName, CustomerPassword, CustomerHP, CustomerAddress, CustomerStatus and CustomerFeedback. Now in the group project, there are only 6 attributes, which are CustomerID, Cu\_Name, Cu\_Email, Cu\_HP, Reg\_Date and Cu\_Status. Secondly, we have only 6 attributes inside Payment, which are PaymentID, ServiceTax, DeliveryFee, NetPrice, PaymentType and OrderID. However, we have 17 attributes inside the entity in the group project, which are Payment ID, Customer\_ID, Cu\_HP, Promo\_Code, Order\_Type, Payment\_SubTotal, Delivery\_Fee, Payment\_Total, Payment\_Type, Payment\_Date, Payment\_Time, Order\_Status, Cu\_Remark, Cu\_Office\_Inasis, Cu\_Block\_Room, Rider\_ID and Delivered\_Time. Lastly, the Staff entity has StaffID, StaffUsername, StaffPosition, StaffName, StaffPassword, StaffHP as attributes in the proposal, but StaffID, Staff\_Name, Staff\_Email, Staff\_HP in the group project.

### 5.3 Relationships

Since most of the entities and attributes are different from the proposal, therefore the relationships are different. The figure below shows the relationships in the proposal and the group project.

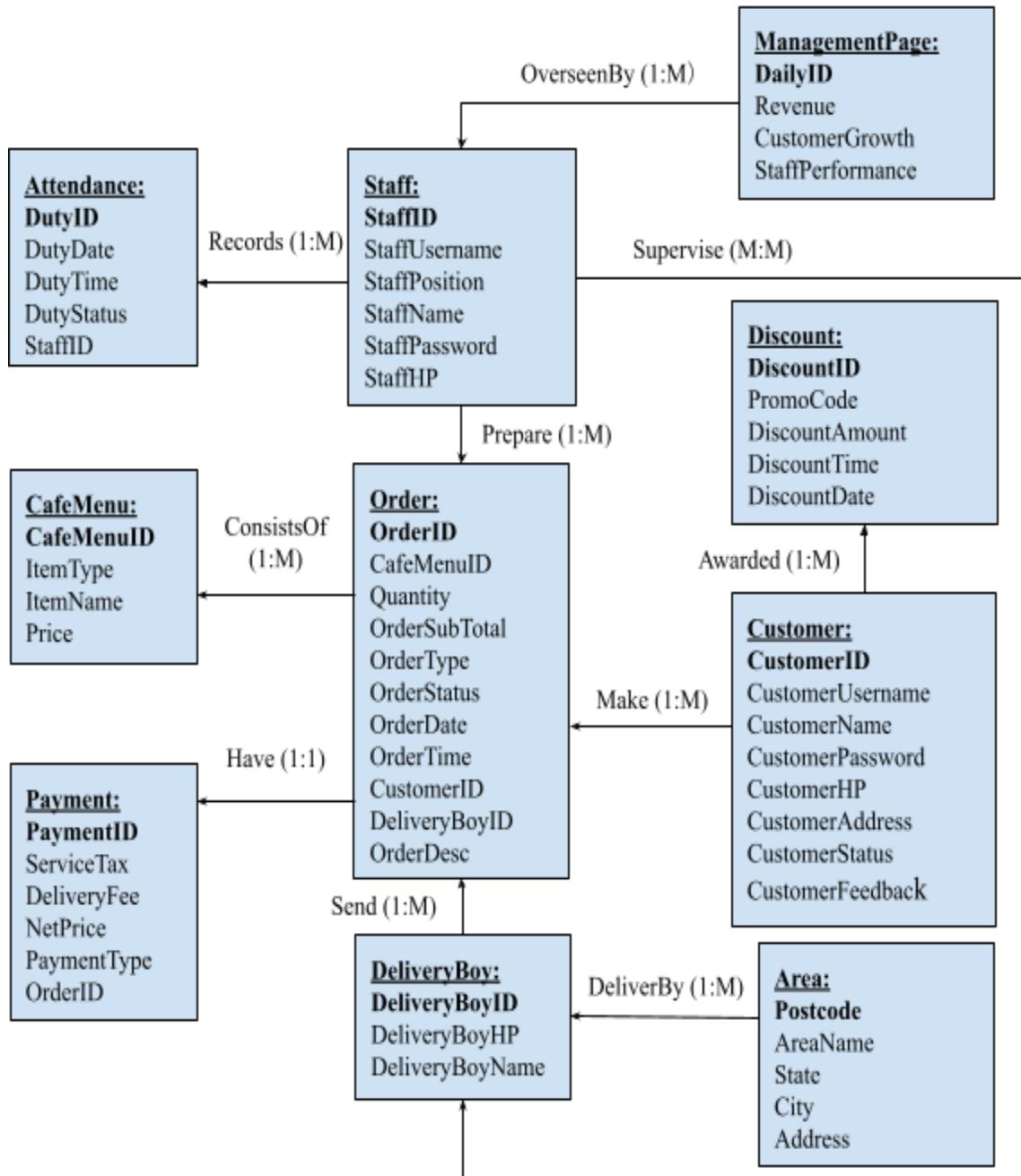


Figure 5.3.1: Relationships in the proposal

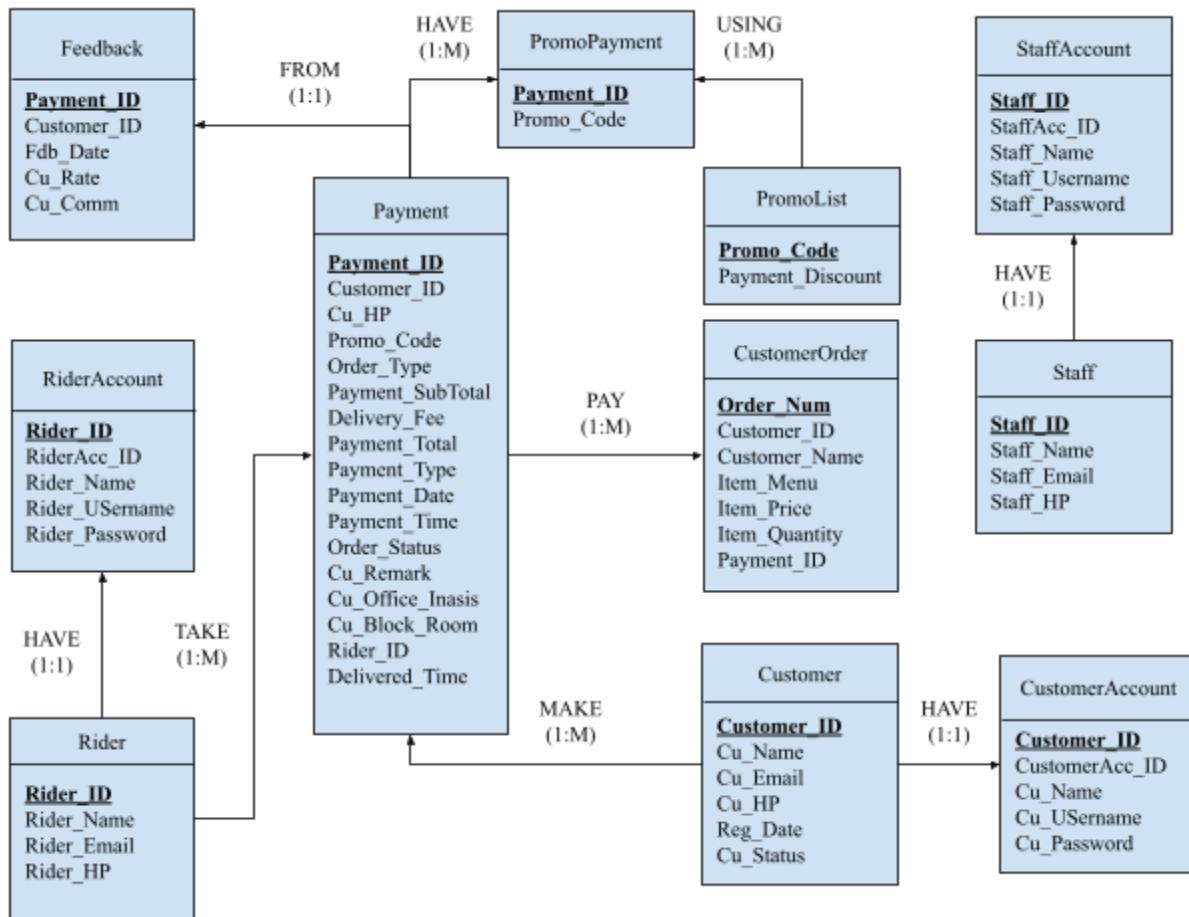


Figure 5.3.2: Relationships in the group project

## 6.0 Data Model

Customer (**CustomerID**, Cu\_Name, Cu\_Email, Cu\_HP, Reg\_Date, Cu\_Status)

CustomerAccount(**CustomerID**, CustomerAcc\_ID, Cu\_Name, Cu\_Username, Cu\_Password)

CustomerOrder(**Order\_Num**, Customer\_ID, Customer\_Name, Item\_Menu, Item\_Price, Item\_Quantity, Payment\_ID)

Feedback(**Payment\_ID**, Customer\_ID, Fdb\_Date, Cu\_Rate, Cu\_Comn)

Payment(**Payment\_ID**, Customer\_ID, Cu\_HP, Promo\_Code, Order\_Type, Payment\_SubTotal, Delivery\_Fee, Payment\_Total, Payment\_Type, Payment\_Date, Payment\_Time, Order\_Status, Cu\_Remark, Cu\_Office\_Inasis, Cu\_Block\_Room, Rider\_ID, Delivered\_Time)

PromoList(**Promo\_Code**, Payment\_Discount)

PromoPayment(**Payment\_ID**, Promo\_Code)

Rider(**Rider\_ID**, Rider\_Name, Rider\_Email, Rider\_HP)

RiderAccount(**Rider\_ID**, RiderAcc\_ID, Rider\_Name, Rider\_Username, Rider\_Password)

Staff(**Staff\_ID**, Staff\_Name, Staff\_Email, Staff\_HP)

StaffAccount(**Staff\_ID**, StaffAcc\_ID, Staff\_Name, Staff\_Username, Staff\_Password)

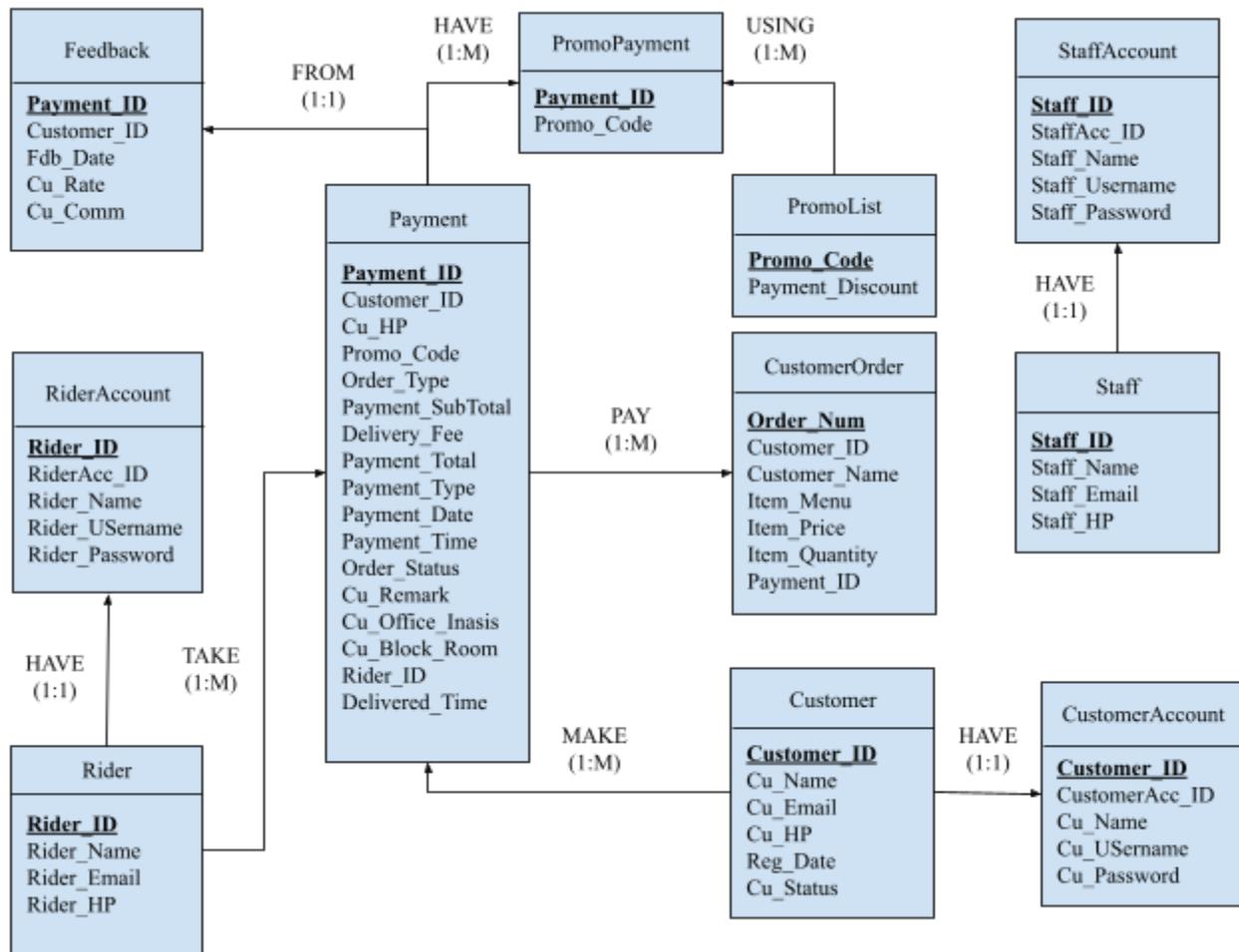


Figure 6.0.1: Data Model in the group project

Below lists the relationship between each entity and their justifications:

1. The relation between Customer and CustomerAccount is a one-to-one relation, because one customer only can have one account only.
2. The relation between Customer and Payment is a one-to-many relation, because one customer can make many payments.
3. The relation between Payment and CustomerOrder is a one-to-many relation, because one payment pays from many orders.
4. The relation between Feedback and Payment is a one-to-one relation, because one feedback only can from one payment only.

5. The relation between Payment and PromoPayment is a one-to-many relation, because one payment can have many promo codes.
6. The relation between PromoList and PromoPayment is a one-to-many relation, because one promo code from PromoList can be for many payments.
7. The relation between Rider and Payment is a one-to-many relation, because one rider can take many payments which are ordered by the customer.
8. The relation between RiderAccount and Rider is a one-to-one relation, because one rider can have one account only.
9. The relation between Staff and StaffAccount is a one-to-one relation, because one staff only can have one account only.

## 7.0 Flow Chart

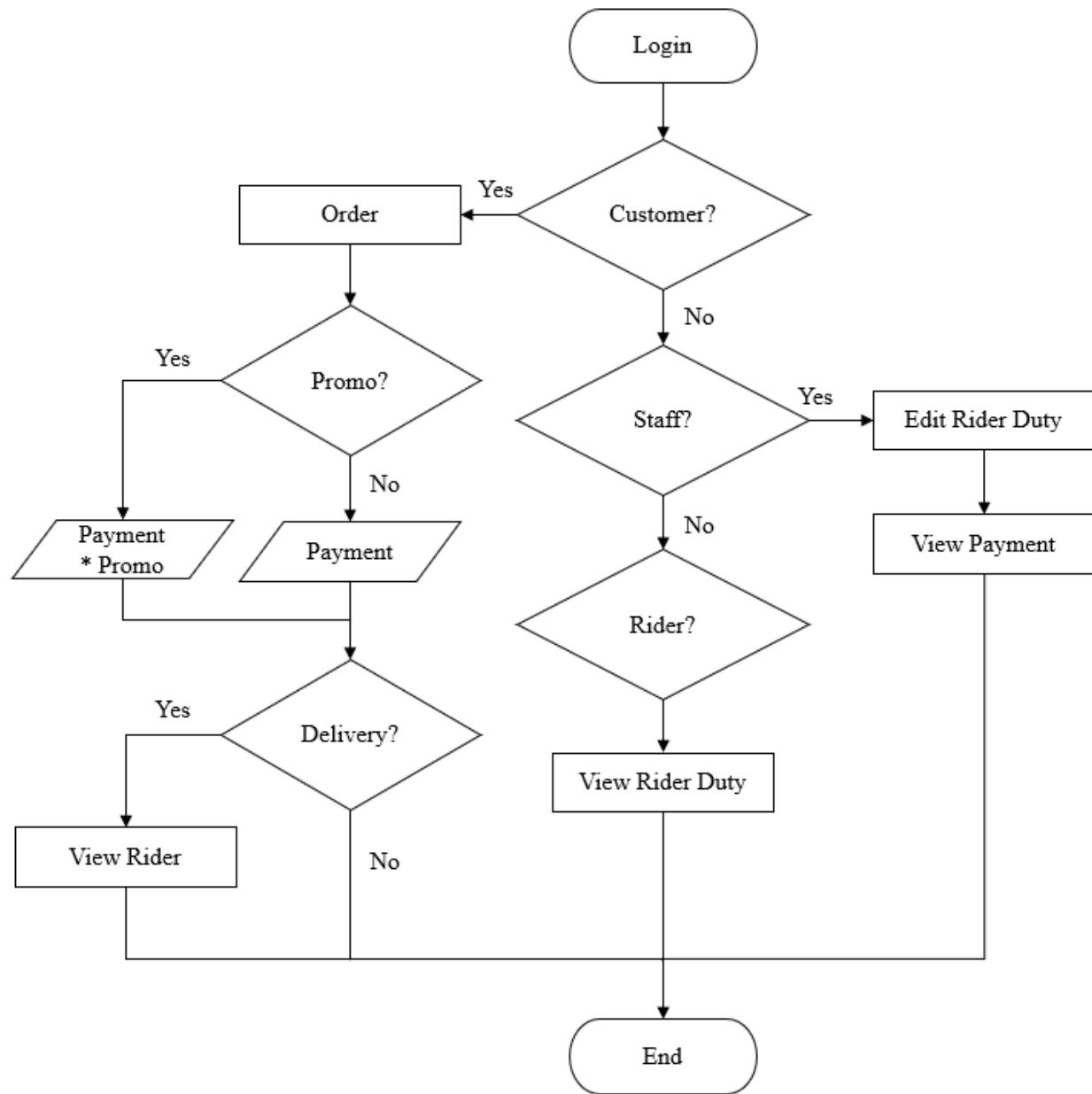


Figure 7.0.1: Flow Chart in the group project

## 8.0 Database

### 8.1 Database Table

In the database system for *UUM Cafe restaurant*, there are 11 relation tables created named **Table Customer**, **Table CustomerAccount**, **Table CustomerOrder**, **Table Feedback**, **Table Payment**, **Table PromoList**, **Table PromoPayment**, **Table Rider**, **Table RiderAccount**, **Table Staff** and **Table StaffAccount**. The design of the tables is shown in *Figure 7.1.1 to 7.1.9* below:

	Field Name	Data Type
1	Customer_ID	Short Text
	Cu_Name	Short Text
	Cu_Email	Short Text
	Cu_HP	Short Text
	Reg_Date	Date/Time
	Cu_Status	Short Text

Figure 8.1.1: Table Customer

	Field Name	Data Type
1	Customer_ID	Short Text
	CustomerAcc_ID	Short Text
	Cu_Name	Short Text
	Cu_Username	Short Text
	Cu_Password	Short Text

Figure 8.1.2: Table CustomerAccount

	Field Name	Data Type
1	Order_Num	AutoNumber
	Customer_ID	Short Text
	Customer_Name	Short Text
	Item_Menu	Short Text
	Item_Price	Currency
	Item_Quantity	Number
	Payment_ID	Short Text

Figure 8.1.3: Table CustomerOrder

	Field Name	Data Type
1	Payment_ID	Short Text
	Customer_ID	Short Text
	Fdb_Date	Date/Time
	Cu_Rate	Short Text
	Cu_Comm	Short Text

Figure 8.1.4: Table Feedback

	Field Name	Data Type
1	Payment_ID	Short Text
	Customer_ID	Short Text
	Cu_HP	Short Text
	Promo_Code	Short Text
	Order_Type	Short Text
	Payment_SubTotal	Currency
	Delivery_Fee	Currency
	Payment_Total	Currency
	Payment_Type	Short Text
	Payment_Date	Date/Time
	Payment_Time	Date/Time
	Order_Status	Short Text
	Cu_Remark	Short Text
	Cu_Office_Inasis	Short Text
	Cu_Block_Room	Short Text
	Rider_ID	Short Text
	Delivered_Time	Date/Time

Figure 8.1.5: Table Payment

	Field Name	Data Type
1	Promo_Code	Short Text
	Payment_Discount	Short Text

Figure 8.1.6: Table PromoList

	Field Name	Data Type
1	Payment_ID	Short Text
2	Promo_Code	Short Text

Figure 8.1.7: Table PromoPayment

	Field Name	Data Type
1	Rider_ID	Short Text
2	Rider_Name	Short Text
3	Rider_Email	Short Text
4	Rider_HP	Short Text

Figure 8.1.8: Table Rider

	Field Name	Data Type
1	Rider_ID	Short Text
2	RiderAcc_ID	Short Text
3	Rider_Name	Short Text
4	Rider_Username	Short Text
5	Rider_Password	Short Text

Figure 8.1.9: Table RiderAccount

	Field Name	Data Type
1	Staff_ID	Short Text
2	Staff_Name	Short Text
3	Staff_Email	Short Text
4	Staff_HP	Short Text

Figure 8.1.10: Table Staff

	Field Name	Data Type
1	Staff_ID	Short Text
2	StaffAcc_ID	Short Text
3	Staff_Name	Short Text
4	Staff_Username	Short Text
5	Staff_Password	Short Text

Figure 8.1.11: Table StaffAccount

## 8.2 Input Data Testing

Input data testing is a simulated test system.

An example of input is shown in *Figure 8.2.1* below, where the table concerned is **Table Customer**:

	Customer_ID	Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
1	CU0001	Vieem A/L Hrithik Roshan	vieem@uum.edu.my	017-4712253	02/07/2021	Normal
2	CU0002	Aditya Chopra A/P Iqra	adityachopra@uum.edu.my	010-5581095	03/07/2021	Normal
3	CU0003	Tan Wei Xiong	tanweixiong@uum.edu.my	018-7743052	08/09/2021	Normal
4	CU0004	Ghadi bin Hadrami	ghadi@uum.edu.my	012-1425367	12/10/2021	Normal
5	CU0005	Ali Akmal	aliakmal@uum.edu.my	014-8596123	12/01/2022	Normal
6	CU0006	Zhang San	zhangsan@uum.edu.my	012-7485963	09/01/2022	Normal

Figure 8.2.1: Customer Information Input Data Testing

An example of input is shown in *Figure 8.2.2* below, where the table concerned is **Table CustomerAccount**:

	Customer_ID	CustomerAcc_ID	Cu_Name	Cu_Username	Cu_Password
1	CU0001	CUA001	Vieem A/L Hrithik Roshan	Cookies1243	1g2g9erE8
2	CU0002	CUA002	Aditya Chopra A/P Iqra	Square2597	gef5dgc4ere
3	CU0003	CUA003	Tan Wei Xiong	WX1993	WeXiong1993
4	CU0004	CUA004	Ghadi bin Hadrami	Ghadi1111	d2d5fF6efe
5	CU0005	CUA005	Ali Akmal	aliakmal15	AliAkmal123
6	CU0006	CUA006	Zhang San	zhangsan4567	zhangsan1234@

Figure 8.2.2: Customer Account Information Input Data Testing

An example of input is shown in *Figure 8.2.3* below, where the table concerned is **Table CustomerOrder**:

Order_Num	Customer_ID	Customer_Name	Item_Menu	Item_Price	Item_Quantity	Payment_ID
ORD0001	CU0001	Vieem A/L Hrithik Roshan	Green Tea	RM3.90	1	P0001
ORD0003	CU0001	Vieem A/L Hrithik Roshan	Chicken Burger	RM9.90	3	P0001
ORD0005	CU0001	Vieem A/L Hrithik Roshan	Mushroom Soup	RM5.90	2	P0001
ORD0006	CU0002	Aditya Chopra A/P Iqra	Chocolate Cake	RM5.90	2	P0002
ORD0007	CU0002	Aditya Chopra A/P Iqra	Milk Tea	RM3.90	1	P0002
ORD0008	CU0002	Aditya Chopra A/P Iqra	French Fries	RM5.90	2	P0002
ORD0009	CU0003	Tan Wei Xiong	Oriental Chicken Chop	RM10.90	3	P0003
ORD0010	CU0003	Tan Wei Xiong	Lemon Tea	RM3.90	1	P0003
ORD0011	CU0004	Ghadi bin Hadrami	Green Tea	RM3.90	5	P0004
ORD0012	CU0004	Ghadi bin Hadrami	Spicy Chicken Burger	RM10.90	2	P0004
ORD0013	CU0004	Ghadi bin Hadrami	Spaghetti Bolognese	RM3.90	3	P0004
ORD0014	CU0005	Ali Akmal	Milk Tea	RM3.90	1	P0005
ORD0015	CU0005	Ali Akmal	Potato Wedges	RM5.90	2	P0005
ORD0016	CU0005	Ali Akmal	Spaghetti Carbonara with Mushroom	RM9.90	1	P0005
ORD0017	CU0005	Ali Akmal	Grilled Chicken with Mushroom Sauce	RM12.30	1	P0005
ORD0018	CU0005	Ali Akmal	Tiramisu	RM6.90	1	P0005
ORD0019	CU0001	Vieem A/L Hrithik Roshan	Potato Soup	RM5.90	2	P0006
ORD0020	CU0001	Vieem A/L Hrithik Roshan	Crispy Fish Burger	RM9.90	6	P0006
ORD0021	CU0005	Ali Akmal	Milk Tea	RM3.90	3	P0007
ORD0022	CU0005	Ali Akmal	Grilled Chicken with Black Pepper Sauce	RM12.30	1	P0007
ORD0023	CU0005	Ali Akmal	Chicken Soup	RM6.90	2	P0007
ORD0024	CU0006	Zhang San	Spaghetti Aglio Olio Peperoncino	RM9.90	1	P0008
ORD0025	CU0006	Zhang San	Cheesecake	RM6.90	2	P0008
ORD0040	CU0006	Zhang San	Milk Tea	RM11.70	3	P0010
ORD0041	CU0006	Zhang San	Milk Tea	RM7.80	2	P0011
ORD0042	CU0006	Zhang San	Green Tea	RM3.90	1	P0011
ORD0043	CU0006	Zhang San	French Fries	RM5.90	1	P0011
ORD0044	CU0006	Zhang San	Milk Tea	RM7.80	2	P0015
ORD0045	CU0006	Zhang San	Milk Tea	RM11.70	3	P0016
ORD0046	CU0006	Zhang San	Milk Tea	RM7.80	2	P0018
ORD0047	CU0006	Zhang San	French Fries	RM11.80	2	P0018
ORD0052	CU0006	Zhang San	Chicken Burger	RM29.70	3	P0013
ORD0053	CU0006	Zhang San	Spicy Chicken Burger	RM32.70	3	P0013
ORD0054	CU0006	Zhang San	Oriental Chicken Chop	RM21.80	2	P0013
ORD0055	CU0006	Zhang San	Milk Tea	RM7.80	2	P0014
ORD0056	CU0006	Zhang San	Green Tea	RM7.80	2	P0014
ORD0057	CU0006	Zhang San	Milk Tea	RM7.80	2	P0009
ORD0058	CU0006	Zhang San	Green Tea	RM3.90	1	P0009
ORD0059	CU0006	Zhang San	Milk Tea	RM7.80	2	P0012
ORD0060	CU0006	Zhang San	French Fries	RM11.80	2	P0012
ORD0061	CU0006	Zhang San	Milk Tea	RM7.80	2	P0017
ORD0062	CU0006	Zhang San	French Fries	RM5.90	1	P0017
ORD0063	CU0006	Zhang San	Mushroom Soup	RM11.80	2	P0017
ORD0064	CU0006	Zhang San	Milk Tea	RM11.70	3	P0019
ORD0065	CU0006	Zhang San	Mushroom Soup	RM23.60	4	P0019
ORD0066	CU0006	Zhang San	Spaghetti Bolognese	RM19.80	2	P0019
ORD0067	CU0006	Zhang San	Milk Tea	RM7.80	2	P0020
ORD0072	CU0006	Zhang San	Milk Tea	RM11.70	3	P0021
ORD0073	CU0006	Zhang San	French Fries	RM17.70	3	P0021
ORD0074	CU0006	Zhang San	Milk Tea	RM7.80	2	P0022
ORD0075	CU0006	Zhang San	Green Tea	RM3.90	1	P0022
ORD0076	CU0006	Zhang San	Milk Tea	RM3.90	1	P0023

Figure 8.2.3: Customer Order Information Input Data Testing

An example of input is shown in Figure 8.2.4 below, where the table concerned is **Table Feedback**:

	Payment_ID	Customer_ID	Fdb_Date	Cu_Rate	Cu_Comm
[+]	P0001	CU0001	29/12/2021	2	Paper bag is broken
[+]	P0002	CU0002	27/01/2022	4	Food is delicious
[+]	P0003	CU0003	24/12/2021	5	Responsive
[+]	P0008	CU0006	17/01/2022	1	Cold
[+]	P0018	CU0006	17/01/2022	5	Delicious

Figure 8.2.4: Feedback Information Input Data Testing

An example of input is shown in *Figure 8.2.5* below, where the table concerned is **Table Payment**:

Payment	Payment_ID	Customer_ID	Cu_HP	Promo_Code	Order_Type	Payment_SubTotal	Delivery_Fee	Payment_Total	Payment_Type	Payment_Date	Payment_Time	Order_Status	Cu_Remark	Cu_Office_Inasis	Cu_Block_Room	Rider_ID	Delivered_Time
[+]	P0001	CU0001	017-4712233	NEWREG	Dine In	RM45.40	RM0.00	RM22.70	Cash	31/12/2021	10:03:00 AM	Food completed	N/A	N/A	N/A		
[+]	P0002	CU0002	010-5581095	NEWYEAR	Take Away	RM27.50	RM0.00	RM17.88	Online Banking	31/12/2021	12:06:33 PM	Kitchen preparing	N/A	N/A	N/A		
[+]	P0003	CU0003	018-774052	EATINGDEALS	Delivery	RM36.60	RM1.00	RM30.08	DuitNow QR	31/12/2021	01:14:25 PM	Rider on the way	N/A	HEP	A201	R0001	
[+]	P0004	CU0004	012-1425367	N/A	Dine In	RM53.00	RM0.00	RM53.00	Cash	31/12/2021	06:45:12 PM	Kitchen preparing	N/A	N/A	N/A		
[+]	P0005	CU0005	014-8596123	N/A	Delivery	RM44.00	RM1.00	RM45.80	Cash	31/12/2021	07:35:41 PM	Kitchen preparing	N/A	Petronas	4G205	R0003	
[+]	P0006	CU0006	012-7485956	N/A	Delivery	RM71.20	RM1.00	RM72.20	Online Banking	11/01/2022	01:00:00 PM	Rider on the way	N/A	TM	6D311	R0004	
[+]	P0007	CU0007	012-7485956	N/A	Dine In	RM37.80	RM0.00	RM37.80	Cash	14/01/2022	08:46:38 PM	Kitchen preparing	Milk tea less sugar	N/A	N/A		
[+]	P0008	CU0008	012-7485956	N/A	Delivery	RM23.70	RM1.00	RM24.70	Cash	15/01/2022	01:35:50 PM	Kitchen preparing	N/A	Grantt	SG105	R0003	03:38:34 PM
[+]	P0009	CU0009	012-7485956	N/A	Delivery	RM10.70	RM0.00	RM11.70	Cash	21/01/2022	02:00:00 PM	Delivery	N/A	N/A	N/A		
[+]	P0010	CU0006	012-7485953	NEWREG	Delivery	RM11.70	RM1.00	RM12.70	Cash	17/01/2022	06:31:53 PM	Kitchen preparing	Less sugar	SOS	3011		
[+]	P0011	CU0006	012-7485953	N/A	Delivery	RM17.69	RM1.00	RM18.69	DuitNow QR	17/01/2022	06:52:54 PM	Delivery	N/A	Mamalat	6D501	R0005	
[+]	P0012	CU0006	012-7485953	N/A	Delivery	RM19.60	RM0.00	RM19.60	Cash	21/01/2022	02:25:27 PM	Delivery	N/A	N/A	N/A		
[+]	P0013	CU0006	012-7485953	N/A	Delivery	RM84.20	RM0.00	RM84.20	RM0.00	20/01/2022	11:08:05 AM		N/A	N/A	N/A		
[+]	P0014					RM0.00	RM0.00	RM0.00									
[+]	P0015	CU0006	012-7485953	N/A	Dine In	RM7.80	RM0.00	RM7.80	Cash	17/01/2022	06:57:18 PM	Dine In	N/A	N/A	N/A		
[+]	P0016	CU0006	012-7485953	N/A	Dine In	RM11.70	RM0.00	RM11.70	Cash	17/01/2022	06:59:59 PM	Dine In	N/A	N/A	N/A		
[+]	P0017	CU0006	012-7485953	N/A	Delivery	RM25.50	RM0.00	RM25.50	Cash	21/01/2022	02:29:35 PM	Delivery	N/A	N/A	N/A		
[+]	P0018	CU0006	012-7485953	N/A	Dine In	RM19.60	RM0.00	RM19.60	Cash	17/01/2022	07:01:48 PM	Dine In	N/A	N/A	N/A		
[+]	P0019	CU0006	012-7485953	N/A	Delivery	RM55.10	RM0.00	RM55.10	DuitNow QR	21/01/2022	02:33:05 PM	Delivery	N/A	N/A	N/A		
[+]	P0020	CU0006	012-7485953	N/A	Delivery	RM7.80	RM0.00	RM7.80	Cash	21/01/2022	02:34:53 PM	Delivery	N/A	N/A	N/A		
[+]	P0021					RM0.00	RM0.00	RM0.00									
[+]	P0022	CU0006	012-7485953	N/A	Dine In	RM11.70	RM0.00	RM11.70	Cash	21/01/2022	02:49:17 PM	Dine In	N/A	N/A	N/A		
[+]	P0023	CU0006	012-7485953	N/A	Dine In	RM3.90	RM0.00	RM3.90	DuitNow QR	21/01/2022	03:00:18 PM	Dine In	N/A	N/A	N/A		

Figure 8.2.5: Payment Information Input Data Testing

An example of input is shown in *Figure 8.2.6* below, where the table concerned is **Table PromoList**:

PromoList	Promo_Code	Payment_Discount
[+]	EATINGDEALS	20
[+]	HAPPYMEAL	15
[+]	NEWREG	50
[+]	NEWYEAR	35
[+]	PICKUP	10

Figure 8.2.6: PrmoList Information Input Data Testing

An example of input is shown in *Figure 8.2.7* below, where the table concerned is **Table PromoPayment**:

Payment_ID	Promo_Code
P0001	NEWREG
P0002	NEWYEAR
P0003	EATINGDEALS
P0010	NEWREG

Figure 8.2.7: PromoPayment Information Input Data Testing

An example of input is shown in *Figure 8.2.8* below, where the table concerned is **Table Rider**:

Rider_ID	Rider_Name	Rider_Email	Rider_HP
R0001	Awwam bin Fahrezz	awwam@rider.uum.my	011-57896482
R0002	Braham A/L Kalyan	braham@rider.uum.my	011-15839682
R0003	Ng Wei Heng	ngweihong@rider.uum.my	017-2582076
R0004	Elhasiq bin Izdihaar	elhasiq@rider.uum.my	017-4729523
R0005	Andrew Tan Zhen Yong	andrewtanzhenyong@rider.uum.my	018-8382014
R0006	Aadi A/L Hamza	aadi@rider.uum.my	016-4582246
R0007	Jallal bin Khairun Nas	jallal@rider.uum.my	012-3567826
R0008	Lim Chin Yee	limchinnyee@rider.uum.my	019-9148062
R0009	Kabir A/L Agni	kabir@rider.uum.my	014-8573583

Figure 8.2.8: Rider Information Input Data Testing

An example of input is shown in *Figure 8.2.9* below, where the table concerned is **Table RiderAccount**:

Rider_ID	RiderAcc_ID	Rider_Name	Rider_Username	Rider_Password
R0001	RA0001	Awwam bin Fahrezz	awwam	awwamdb1
R0002	RA0002	Braham A/L Kalyan	braham	brahamdb2
R0003	RA0003	Ng Wei Heng	ngweihong	ngweihongdb3
R0004	RA0004	Elhasiq bin Izdihaar	elhasiq	elhasiqdb4
R0005	RA0005	Andrew Tan Zhen Yong	andrewtanzhenyong	andrewtanzhenyongdb5
R0006	RA0006	Aadi A/L Hamza	aadi	aadidb6
R0007	RA0007	Jallal bin Khairun Nas	jallal	jallaldb7
R0008	RA0008	Lim Chin Yee	limchinnyee	limchinyeedb8
R0009	RA0009	Kabir A/L Agni	kabir	kabirdb9

Figure 8.2.9: Rider Account Information Input Data Testing

An example of input is shown in *Figure 8.2.10* below, where the table concerned is **Table Staff**:

	Staff_ID	Staff_Name	Staff_Email	Staff_HP
+	ST0001	Abu al Khayr bin Malazi	abualkhayr@staff.uum.my	016-5434221
+	ST0002	Ooi Cheong Wai	ooicheongwao@staff.uum.my	012-4443210

*Figure 8.2.10: Staff Information Input Data Testing*

An example of input is shown in *Figure 8.2.11* below, where the table concerned is **Table StaffAccount**:

	Staff_ID	StaffAcc_ID	Staff_Name	Staff_Username	Staff_Password
+	ST0001	STA001	Abu al Khayr bin Malazi	abualkhayr	UUMCafe2021
+	ST0002	STA002	Ooi Cheong Wai	ooicheongwai	2021AGMCafe

*Figure 8.2.11: Staff Account Information Input Data Testing*

### 8.3 Relationships

Using the utility provided by *Microsoft Access 2016*, 9 relation tables are linked together. The relationship between the tables is shown in *Figure 8.3.1* below.

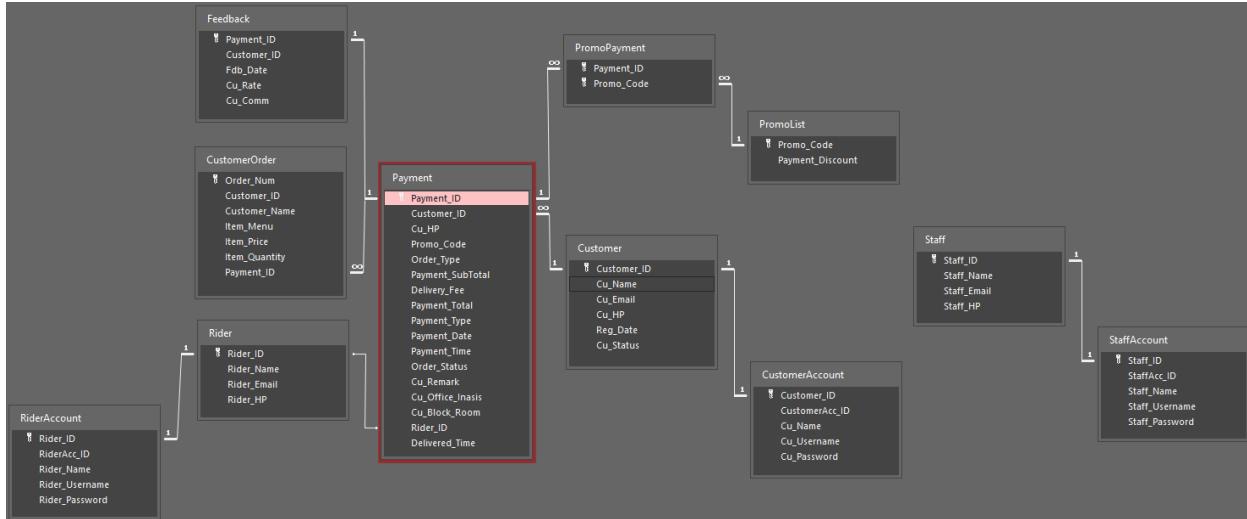


Figure 8.3.1: Relationship between Tables

## 8.4 Normalization

In the database system for *UUM Cafe restaurant*, there are only 9 tables from 11 relation tables named **Table Customer**, **Table CustomerAccount**, **Table Feedback**, **Table Payment**, **Table PromoList**, **Table Rider**, **Table RiderAccount**, **Table Staff** and **Table StaffAccount** are normalization according to the relationship. The example of the tables is shown in *Figure 8.4.1 to 8.4.9* below:

This is normalization for all the tables according to the relationship by using the database:

Customer																							
Customer_ID	Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status	Click to Add	Payment_ID	Cu_HP	Promo_Codi	Order_Type	Payment_SubTotal	Delivery_Fe	Payment_Tc	Payment_Type	Payment_Dx	Payment_Ti	Order_Status	Cu_Remark	Cu_Office_Ir	Cu_Block_Rc	Rider_ID	Delivered_T	
CU0001	Vleem A/L Hrithik Roshai vleem@uum.e	017-4712253	02/07/2021 Normal																				
# P0001	017-4712253	NEWREG	Dine In				RM45.40	RM0.00	RM22.70	Cash					31/12/2021	10:03:00 AM	Food complete	N/A	N/A	N/A			
# P0002	Aditya Chopra A/P Iqra	adityachopra@ 010-5581095	03/07/2021 Normal																				
# P0002	010-5581095	NEWYEAR	Take Away				RM27.50	RM0.00	RM17.88	Online Banking					31/12/2021	12:06:33 PM	Kitchen prepar	N/A	N/A	N/A			
# P0003	Tan Wei Xiong	tanweixiong@ 018-7743052	08/09/2021 Normal																				
# P0003	018-7743052	EATINGDEALS	Delivery				RM36.60	RM1.00	RM30.08	DuitNow QR					31/12/2021	01:14:25 PM	Rider on the w	N/A	HEP	A201	R0001		
# P0004	Ghadi bin Hadrami	ghadi@uum.e	012-1425367	12/10/2021 Normal																			
# P0004	012-1425367	N/A	Dine In				RM53.00	RM0.00	RM53.00	Cash					31/12/2021	06:45:12 PM	Kitchen prepar	N/A	N/A	N/A			
# P0005	Ali Akmal	aliakmal@uum	014-8596123	12/12/2022 Normal																			
# P0005	014-8596123	N/A	Delivery				RM44.80	RM1.00	RM45.80	Cash					31/12/2021	07:35:41 PM	Kitchen prepar	N/A	Petronas	4G205	R0003		
# P0006	Zhang San	zhangsan@uum	012-7485963	09/01/2022 Normal																			
# P0006	012-7485963	N/A	Dine In				RM37.80	RM0.00	RM37.80	Cash					14/01/2022	08:46:38 PM	Kitchen prepar	Milk tea less s	N/A	N/A			
# P0007	012-7485963	N/A	Delivery				RM23.70	RM1.00	RM24.70	Cash					15/01/2022	01:35:50 PM	Kitchen prepar	N/A	Grantt	5G105	R0003	03:38:34 PM	
# P0008	012-7485963	N/A	Delivery				RM11.70	RM0.00	RM11.70	Cash					21/01/2022	02:20:01 PM	Delivery	N/A	N/A	N/A			
# P0009	012-7485963	N/A	Delivery				RM13.60	RM0.00	RM19.60	Cash					21/01/2022	02:25:27 PM	Delivery	N/A	N/A	N/A			
# P0010	012-7485963	N/A	Delivery				RM25.50	RM0.00	RM25.50	Cash					21/01/2022	02:29:35 PM	Delivery	N/A	N/A	N/A			
# P0011	012-7485963	N/A	Delivery				RM55.10	RM0.00	RM55.10	DuitNow QR					21/01/2022	02:33:05 PM	Delivery	N/A	N/A	N/A			
# P0012	012-7485963	N/A	Delivery				RM7.80	RM0.00	RM7.80	Cash					21/01/2022	02:34:53 PM	Delivery	N/A	N/A	N/A			
# P0013	012-7485963	N/A	Dine In				RM11.70	RM0.00	RM11.70	Cash					21/01/2022	02:49:17 PM	Dine In	N/A	N/A	N/A			
# P0014	012-7485963	N/A	Dine In				RM3.90	RM0.00	RM3.90	DuitNow QR					21/01/2022	03:00:18 PM	Dine In	N/A	N/A	N/A			
# P0015	012-7485963	N/A	Delivery				RM84.20	RM0.00	RM84.20	Cash					20/01/2022	11:08:05 AM	N/A	N/A	N/A	N/A			
# P0016	012-7485963	N/A	Delivery				RM11.70	RM1.00	RM12.70	Cash					17/01/2022	06:31:53 PM	Kitchen prepar	Less sugar	SQS	3011			
# P0017	012-7485963	N/A	Delivery				RM17.60	RM1.00	RM18.60	DuitNow QR					17/01/2022	06:52:54 PM	Delivery	N/A	Muamalat	6D501	R0005		
# P0018	012-7485963	N/A	Dine In				RM7.80	RM0.00	RM7.80	Cash					17/01/2022	06:57:18 PM	Dine In	N/A	N/A	N/A			
# P0019	012-7485963	N/A	Dine In				RM11.70	RM0.00	RM11.70	Cash					17/01/2022	06:59:59 PM	Dine In	N/A	N/A	N/A			
# P0020	012-7485963	N/A	Delivery				RM19.60	RM0.00	RM19.60	Cash					17/01/2022	07:01:48 PM	Dine In	N/A	N/A	N/A			
# P0021	012-7485963	N/A	Delivery				RM71.20	RM1.00	RM72.20	Online Banking					11/01/2022	01:00:00 PM	Rider on the w	N/A	TM	6D311	R0004		
# P0022	012-7485963	N/A	Delivery				RM0.00	RM0.00	RM0.00														

Figure 8.4.1: Normalization for Table Customer

CustomerAccount						
Customer_ID	CustomerAc	Cu_Name	Cu_Username	Cu_Password	Click to Add	
CU0001	CUA001	Vieem A/L Hrithik Roshar	Cookies1243	1g2g9erE8		
		Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
		+ Vieem A/L Hrithik Roshar	vieem@uum.e	017-4712253	02/07/2021	Normal
*						
CU0002	CUA002	Aditya Chopra A/P Iqra	Square2597	gef5dgf4ere		
		Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
		+ Aditya Chopra A/P Iqra	adityachopra@	010-5581095	03/07/2021	Normal
*						
CU0003	CUA003	Tan Wei Xiong	WX1993	WeXiong1993		
		Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
		+ Tan Wei Xiong	tanweixiong@	018-7743052	08/09/2021	Normal
*						
CU0004	CUA004	Ghadi bin Hadrami	Ghadi1111	d2d5ff6efe		
		Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
		+ Ghadi bin Hadrami	ghadi@uum.e	012-1425367	12/10/2021	Normal
*						
CU0005	CUA005	Ali Akmal	aliakmal15	AliAkmal123		
		Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
		+ Ali Akmal	aliakmal@uum	014-8596123	12/01/2022	Normal
*						
CU0006	CUA006	Zhang San	zhangsan4567	zhangsan1234@		
		Cu_Name	Cu_Email	Cu_HP	Reg_Date	Cu_Status
		+ Zhang San	zhangsan@uur	012-7485963	09/01/2022	Normal
*						

Figure 8.4.2: Normalization for Table CustomerAccount

Feedback														
Payment_ID	Customer_ID	Fdb_Date	Cu_Rate	Cu_Comm	Click to Add									
P0001	CU0001	29/12/2021	2	Paper bag is broken										
	Customer_ID	Cu_Hp	Promo_Cod	Order_Type	Payment_SubTotal	Delivery_Fe	Payment_Tc	Payment_Type	Payment_Ds	Payment_Ti	Order_Status	Cu_Remark	Cu_Office_Ir	Cu_Block_Rc
	+ CU0001	017-4712253	NEWREG	Dine In	RM45.40	RM0.00	RM22.70	Cash	31/12/2021	10:03:00 AM	Food complete	N/A	N/A	N/A
*					RM0.00	RM0.00	RM0.00							
P0002	CU0002	27/01/2022	4	Food is delicious										
	Customer_ID	Cu_Hp	Promo_Cod	Order_Type	Payment_SubTotal	Delivery_Fe	Payment_Tc	Payment_Type	Payment_Ds	Payment_Ti	Order_Status	Cu_Remark	Cu_Office_Ir	Cu_Block_Rc
	+ CU0002	010-5581095	NEWYEAR	Take Away	RM27.50	RM0.00	RM17.88	Online Banking	31/12/2021	12:06:33 PM	Kitchen prepar	N/A	N/A	N/A
*					RM0.00	RM0.00	RM0.00							
P0003	CU0003	24/12/2021	5	Responsive										
	Customer_ID	Cu_Hp	Promo_Cod	Order_Type	Payment_SubTotal	Delivery_Fe	Payment_Tc	Payment_Type	Payment_Ds	Payment_Ti	Order_Status	Cu_Remark	Cu_Office_Ir	Cu_Block_Rc
	+ CU0003	018-7743052	EATINGDEALS	Delivery	RM36.60	RM1.00	RM30.08	DuitNow QR	31/12/2021	01:14:25 PM	Rider on the w	N/A	HEP	A201
*					RM0.00	RM0.00	RM0.00							
P0008	CU0006	17/01/2022	1	Cold										
	Customer_ID	Cu_Hp	Promo_Cod	Order_Type	Payment_SubTotal	Delivery_Fe	Payment_Tc	Payment_Type	Payment_Ds	Payment_Ti	Order_Status	Cu_Remark	Cu_Office_Ir	Cu_Block_Rc
	+ CU0006	012-7485963	N/A	Delivery	RM23.70	RM1.00	RM24.70	Cash	15/01/2022	01:35:50 PM	Kitchen prepar	N/A	Grantt	5G105
*					RM0.00	RM0.00	RM0.00							
P0018	CU0006	17/01/2022	5	Delicious										
	Customer_ID	Cu_Hp	Promo_Cod	Order_Type	Payment_SubTotal	Delivery_Fe	Payment_Tc	Payment_Type	Payment_Ds	Payment_Ti	Order_Status	Cu_Remark	Cu_Office_Ir	Cu_Block_Rc
	+ CU0006	012-7485963	N/A	Dine In	RM19.60	RM0.00	RM19.60	Cash	17/01/2022	07:01:48 PM	Dine In	N/A	N/A	N/A
*					RM0.00	RM0.00	RM0.00							

Figure 8.4.3: Normalization for Table Feedback

# SQIT3043(A) Group Project

Payment_ID	Customer_ID	Cust_HP	Promo_Code	Order_Type	Payment_SubTotal	Delivery_Fee	Payment_Percentage	Total	Payment_Type	Payment_Dates	Payment_Time	Order_Status	Cust_Note	Cust_Office_Shop	Cust_Block_Report	Rider_ID	Delivered_Time	
P0003	CU0003	018-7343023	EATINGDEALS	Delivery	RM16.60	RM0.00	RM16.60	RM16.60	DuitNow QR	31/12/2021	01:14:25 PM	Rider on the way	N/A	HEP	A291	80003		
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00009	CU0003	Tan Wei Xiong	Oriental Chicken Chop	RM10.90	3													
OR00010	CU0003	Tan Wei Xiong	Lemon Tea	RM1.90	1													
(New)																		
P0004	CU0005	012-1425367	N/A	Dine In	RM53.00	RM0.00	RM53.00	RM53.00	Cash	31/12/2021	06:45:12 PM	Kitchen preparing	N/A	N/A	N/A	N/A		
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00011	CU0004	Ghais bin Hadrami	Green Tea	RM1.90	5													
OR00012	CU0004	Ghais bin Hadrami	Spicy Chicken Burger	RM10.90	2													
OR00013	CU0004	Ghais bin Hadrami	Spaghetti Bolognese	RM10.90	3													
(New)																		
P0005	CU0005	014-8596123	N/A	Delivery	RM44.80	RM1.00	RM45.80	RM45.80	Cash	31/12/2021	07:35:41 PM	Kitchen preparing	N/A	Petronas	4G205	R0003		
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00014	CU0005	All Ahmal	Milk Tea	RM1.90	3													
OR00015	CU0005	All Ahmal	Potato Wedges	RM5.90	2													
OR00016	CU0005	All Ahmal	Spaghetti Carbonara with Mushroom	RM9.90	1													
OR00017	CU0005	All Ahmal	Grilled Chicken with Mushroom	RM12.30	1													
OR00018	CU0005	All Ahmal	Tiramisu	RM6.00	0													
(New)																		
P0006	CU0005	014-7485963	N/A	Delivery	RM71.20	RM1.00	RM72.20	RM72.20	Online Banking	11/01/2022	01:00:00 PM	Rider on the way	N/A	TM	6D311	R0004		
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00019	CU0005	Vineem A/R, Hritikk Roshan	Potato Salad	RM1.90	0													
OR00020	CU0005	Vineem A/R, Hritikk Roshan	Crispy Fish Burger	RM9.90	6													
(New)																		
P0007	CU0004	012-7485963	N/A	Dine In	RM17.80	RM0.00	RM17.80	RM17.80	Cash	14/01/2022	08:46:38 PM	Kitchen preparing	Milk tea less sugar	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00021	CU0005	All Ahmal	Milk Tea	RM1.90	3													
OR00022	CU0005	All Ahmal	Grilled Chicken with Black Pepper Sauce	RM12.30	1													
OR00023	CU0005	All Ahmal	Chicken Soup	RM6.90	2													
(New)																		
P0008	CU0004	012-7485963	N/A	Delivery	RM23.70	RM1.00	RM24.70	RM24.70	Cash	15/01/2022	01:35:50 PM	Kitchen preparing	N/A	Granttt	5G105	R0003	03:38:34 PM	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00024	CU0006	Zhang San	Spaghetti Aglio Olio Peperoncino	RM9.90	1													
OR00025	CU0006	Zhang San	Cheesecake	RM6.90	2													
(New)																		
P0009	CU0005	012-7485963	N/A	Delivery	RM11.70	RM0.00	RM11.70	RM11.70	Cash	21/01/2022	02:20:01 PM	Delivery	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00026	CU0006	Zhang San	Milk Tea	RM1.90	2													
OR00027	CU0006	Zhang San	Green Tea	RM3.90	1													
(New)																		
P0010	CU0005	012-7485963	NEWREG	Delivery	RM11.70	RM1.00	RM12.70	RM12.70	Cash	17/01/2022	06:31:53 PM	Kitchen preparing	Less sugar	SGS	3011			
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00040	CU0006	Zhang San	Milk Tea	RM11.70	3													
(New)																		
P0011	CU0005	012-7485963	N/A	Delivery	RM17.60	RM1.00	RM18.60	RM18.60	DuitNow QR	17/01/2022	06:52:54 PM	Delivery	N/A	Muamalat	6D501	R0005		
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00041	CU0006	Zhang San	Milk Tea	RM2.80	2													
OR00042	CU0006	Zhang San	Green Tea	RM3.90	1													
OR00043	CU0006	Zhang San	French Fries	RM5.90	1													
(New)																		
P0012	CU0005	012-7485963	N/A	Delivery	RM19.60	RM0.00	RM19.60	RM19.60	Cash	21/01/2022	02:25:27 PM	Delivery	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00044	CU0006	Zhang San	Milk Tea	RM1.90	2													
OR00045	CU0006	Zhang San	Chicken Burger	RM25.70	3													
OR00053	CU0006	Zhang San	Spicy Chicken Burger	RM32.70	3													
OR00054	CU0006	Zhang San	Oriental Chicken Chop	RM21.80	2													
(New)																		
P0014	CU0006	012-7485963	N/A	Delivery	RM40.00	RM0.00	RM40.00	RM40.00	RMO.00									
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00055	CU0006	Zhang San	Milk Tea	RM1.90	2													
OR00056	CU0006	Zhang San	Green Tea	RM2.80	2													
(New)																		
P0015	CU0006	012-7485963	N/A	Dine In	RM7.80	RM0.00	RM7.80	RM7.80	Cash	17/01/2022	06:57:18 PM	Dine In	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00044	CU0006	Zhang San	Milk Tea	RM1.90	2													
(New)																		
P0016	CU0006	012-7485963	N/A	Dine In	RM17.00	RM0.00	RM17.00	RM17.00	Cash	17/01/2022	06:59:59 PM	Dine In	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00045	CU0006	Zhang San	Milk Tea	RM11.70	3													
(New)																		
P0017	CU0005	012-7485963	N/A	Delivery	RM25.50	RM0.00	RM25.50	RM25.50	Cash	21/01/2022	02:29:35 PM	Delivery	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00061	CU0006	Zhang San	Milk Tea	RM7.80	2													
OR00062	CU0006	Zhang San	French Fries	RM5.90	1													
OR00063	CU0006	Zhang San	Mushroom Soup	RM11.80	2													
(New)																		
P0018	CU0005	012-7485963	N/A	Dine In	RM19.60	RM0.00	RM19.60	RM19.60	Cash	17/01/2022	07:01:48 PM	Dine In	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00046	CU0006	Zhang San	Milk Tea	RM1.90	2													
OR00047	CU0006	Zhang San	French Fries	RM11.80	2													
(New)																		
P0019	CU0006	012-7485963	N/A	Delivery	RM55.10	RM0.00	RM55.10	RM55.10	DuitNow QR	21/01/2022	02:33:05 PM	Delivery	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00064	CU0006	Zhang San	Milk Tea	RM11.70	3													
OR00065	CU0006	Zhang San	Mushroom Soup	RM23.50	4													
OR00066	CU0006	Zhang San	Spaghetti Bolognese	RM19.50	2													
(New)																		
P0020	CU0005	012-7485963	N/A	Delivery	RM7.80	RM0.00	RM7.80	RM7.80	Cash	21/01/2022	02:34:53 PM	Delivery	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00067	CU0006	Zhang San	Milk Tea	RM7.80	2													
(New)																		
P0021	CU0006	012-7485963	N/A	Dine In	RM0.00	RM0.00	RM0.00	RM0.00										
Order_Num	Customer_ID	Customer_Name	Items_Menu	Item_Price	Item_Quantity	Click to Add												
OR00072	CU0006	Zhang San	Milk Tea	RM11.70	3													
OR00073	CU0006	Zhang San	French Fries	RM11.70	3													
(New)																		
P0022	CU0006	012-7485963	N/A	Dine In	RM11.70	RM0.00	RM11.70	RM11.70	Cash	21/01/2022	02:49:17 PM	Dine In	N/A	N/A	N/A	N/A	N/A	
Order_Num	Customer_ID																	

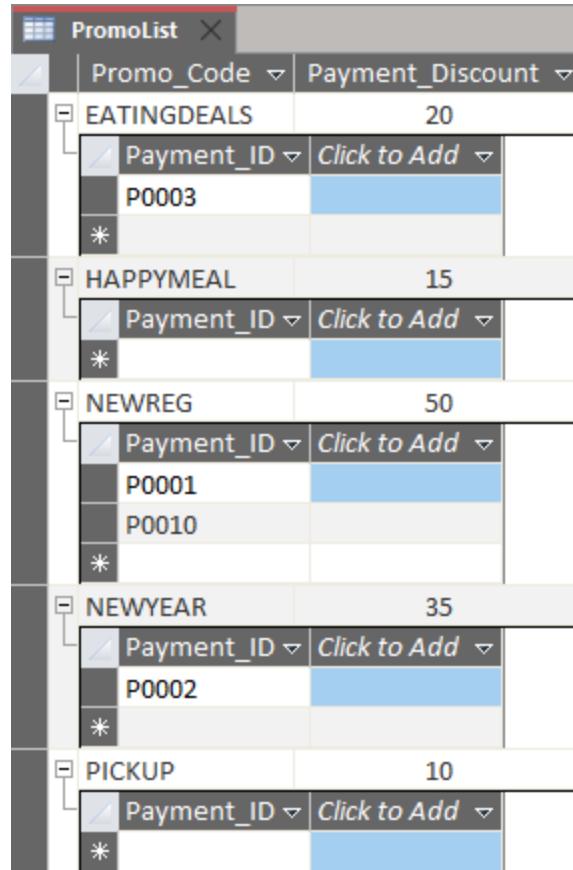


Figure 8.4.5: Normalization for Table PromoList

Rider	Rider_ID	Rider_Name	Rider_Email	Rider_HP	Promo_Code	Order_Type	Payment_SubTotal	Delivery_Fee	Payment_Total	Payment_Type	Payment_Date	Payment_Time	Order_Status	Cu_Remark	Cu_Office_Inasis	Cu_Block_Room	Delivered_Time
R0001	Awwam bin Fahrizzi	awwam@rider.uum.my	011-5789482														
	P0003	CU0003	018-7743052	EATINGDEALS		Delivery	RM36.00	RM1.00	RM30.08	DuitNow QR	31/12/2021	01:14:25 PM	Rider on the way	N/A	HEP	A201	
								RM0.00	RM0.00								
R0002	Braham A/L Kalyan	braham@rider.uum.my	011-15839682														
	P0008	CU0005	012-7485963	N/A		Delivery	RM23.70	RM1.00	RM24.70	Cash	15/01/2022	01:35:50 PM	Kitchen preparing	N/A	Grant	SG105	03:38:34 PM
								RM0.00	RM0.00								
R0003	Ng Wei Heng	ngweihong@rider.uum.my	017-2582076														
	P0005	CU0005	014-8596123	N/A		Delivery	RM44.80	RM1.00	RM45.80	Cash	31/12/2021	07:35:41 PM	Kitchen preparing	N/A	Petronas	4G205	
								RM0.00	RM0.00								
R0004	Elhasiq bin Izdihaar	elhasiq@rider.uum.my	017-4725923														
	P0006	CU0005	012-7485963	N/A		Delivery	RM71.20	RM1.00	RM72.20	Online Banking	11/01/2022	01:00:00 PM	Rider on the way	N/A	TM	60311	
								RM0.00	RM0.00								
R0005	Andrew Tan Zhen Yong	andrewtanztzhyong@rider.uum.my	018-8382014														
	P0011	CU0005	012-7485963	N/A		Delivery	RM17.60	RM1.00	RM18.60	DuitNow QR	17/01/2022	06:52:54 PM	Delivery	N/A	Muamalat	60501	
								RM0.00	RM0.00								
R0006	Aadi A/L Hamza	aadi@rider.uum.my	016-4582246														
	P0007	Jallal bin Kharun Nas	jallal@rider.uum.my	012-3567826													
								RM0.00	RM0.00								
R0008	Lim Chin Yee	limchinyyee@rider.uum.my	019-9148062														
	P0009	Kabir A/L Agni	kabir@rider.uum.my	014-8571583													
								RM0.00	RM0.00								

Figure 8.4.6: Normalization for Table Rider

Rider_ID	RiderAcc_ID	Rider_Name	Rider_Username	Rider_Password	
R0001	RA0001	Awwam bin Fahrezzi	awwam	awwamadb1	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Awwam bin Fahrezzi	awwam@rider.uum.my	011-57896482	
R0002	RA0002	Braham A/L Kalyan	braham	brahamadb2	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Braham A/L Kalyan	braham@rider.uum.my	011-15839682	
R0003	RA0003	Ng Wei Heng	ngweihong	ngweihongdb3	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Ng Wei Heng	ngweihong@rider.uum.my	017-2582076	
R0004	RA0004	Elhasiq bin Izdihaar	elhasiq	elhasiqdb4	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Elhasiq bin Izdihaar	elhasiq@rider.uum.my	017-4729523	
R0005	RA0005	Andrew Tan Zhen Yong	andrewtanzhenyong	andrewtanzhenyongdb5	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Andrew Tan Zhen Yong	andrewtanzhenyong@rider.uum.my	018-8382014	
R0006	RA0006	Aadi A/L Hamza	aadi	aadidb6	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Aadi A/L Hamza	aadi@rider.uum.my	016-4582246	
R0007	RA0007	Jallal bin Khairun Nas	jallal	jallaldb7	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Jallal bin Khairun Nas	jallal@rider.uum.my	012-3567826	
R0008	RA0008	Lim Chin Yee	limchinyyee	limchinyyeedb8	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Lim Chin Yee	limchinyyee@rider.uum.my	019-9148062	
R0009	RA0009	Kabir A/L Agni	kabir	kabirdb9	
		Rider_Name	Rider_Email	Rider_HP	Click to Add
*		+ Kabir A/L Agni	kabir@rider.uum.my	014-8573583	

Figure 8.4.7: Normalization for Table RiderAccount

Staff_ID	Staff_Name	Staff_Email	Staff_HP	Click to Add
ST0001	Abu al Khayr bin Malazi	abualkhayr@staff.uum.my	016-5434221	
StaffAcc_ID	Staff_Name	Staff_Username	Staff_Password	Click to Add
STA001	Abu al Khayr bin Malazi	abualkhayr	UUMCafe2021	
*				
ST0002	Ooi Cheong Wai	ooicheongwao@staff.uum.my	012-4443210	
StaffAcc_ID	Staff_Name	Staff_Username	Staff_Password	Click to Add
STA002	Ooi Cheong Wai	ooicheongwai	2021AGMCafe	
*				

Figure 8.4.8: Normalization for Table Staff

Staff_ID	StaffAcc_ID	Staff_Name	Staff_Username	Staff_Password
ST0001	STA001	Abu al Khayr bin Malazi	abualkhayr	UUMCafe2021
Staff_Name	Staff_Email	Staff_HP	Click to Add	
Abu al Khayr bin Malazi	abualkhayr@staff.uum.my	016-5434221		
*				
ST0002	STA002	Ooi Cheong Wai	ooicheongwai	2021AGMCafe
Staff_Name	Staff_Email	Staff_HP	Click to Add	
Ooi Cheong Wai	ooicheongwao@staff.uum.my	012-4443210		
*				

Figure 8.4.9: Normalization for Table StaffAccount

**One Of The Example For Normalization Customer Order**

# Customer Order - 1NF

Primary key							
Order_Num	Customer_ID	Customer_Name	Item_Menu	Item_Price	Item_Quantity	Payment_ID	
ORD0001	CU0001	Vieem A/L Hrithik Roshan	Green Tea	RM3.90	1	P0001	
ORD0003	CU0001	Vieem A/L Hrithik Roshan	Chicken Burger	RM9.90	3	P0001	
ORD0005	CU0001	Vieem A/L Hrithik Roshan	Mushroom Soup	RM5.90	2	P0001	
ORD0006	CU0002	Aditya Chopra A/P Iqra	Chocolate Cake	RM5.90	2	P0002	
ORD0007	CU0002	Aditya Chopra A/P Iqra	Milk Tea	RM3.90	1	P0002	
ORD0008	CU0002	Aditya Chopra A/P Iqra	French Fries	RM5.90	2	P0002	
ORD0009	CU0003	Tan Wei Xiong	Oriental Chicken Chop	RM10.90	3	P0003	
ORD0010	CU0003	Tan Wei Xiong	Lemon Tea	RM3.90	1	P0003	
ORD0011	CU0004	Ghadi bin Hadrami	Green Tea	RM3.90	5	P0004	
ORD0012	CU0004	Ghadi bin Hadrami	Spicy Chicken Burger	RM10.90	2	P0004	
ORD0013	CU0004	Ghadi bin Hadrami	Spaghetti Bolognese	RM3.90	3	P0004	
ORD0014	CU0005	Ali Akmal	Milk Tea	RM3.90	1	P0005	
ORD0015	CU0005	Ali Akmal	Potato Wedges	RM5.90	2	P0005	
ORD0016	CU0005	Ali Akmal	Spaghetti Carbonara with Mushroom	RM9.90	1	P0005	
ORD0017	CU0005	Ali Akmal	Grilled Chicken with Mushroom Sauce	RM12.30	1	P0005	
ORD0018	CU0005	Ali Akmal	Tiramisu	RM6.90	1	P0005	

## 2NF

Primary key

Customer_ID	Customer_Name
CU0001	Vieem A/L Hrithik Roshan
CU0002	Aditya Chopra A/P Iqra
CU0003	Tan Wei Xiong
CU0004	Ghadi bin Hadrami
CU0005	Ali Akmal

Order_Num	Item_Menu	Item_Price	Item_Quantity	Payment_ID
ORD0001	Green Tea	RM3.90	1	P0001
ORD0003	Chicken Burger	RM9.90	3	P0001
ORD0005	Mushroom Soup	RM5.90	2	P0001
ORD0006	Chocolate Cake	RM5.90	2	P0002
ORD0007	Milk Tea	RM3.90	1	P0002
ORD0008	French Fries	RM5.90	2	P0002
ORD0009	Oriental Chicken Chop	RM10.90	3	P0003
ORD0010	Lemon Tea	RM3.90	1	P0003
ORD0011	Green Tea	RM3.90	5	P0004
ORD0012	Spicy Chicken Burger	RM10.90	2	P0004
ORD0013	Spaghetti Bolognese	RM3.90	3	P0004
ORD0014	Milk Tea	RM3.90	1	P0005
ORD0015	Potato Wedges	RM5.90	2	P0005
ORD0016	Spaghetti Carbonara with Mushroom	RM9.90	1	P0005
ORD0017	Grilled Chicken with Mushroom Sauce	RM12.30	1	P0005
ORD0018	Tiramisu	RM6.90	1	P0005

Primary key Primary key

Customer_ID	Payment_ID
CU0001	P0001
CU0002	P0002
CU0003	P0003
CU0004	P0004
CU0005	P0005

## 3NF

Primary key

Customer_ID	Customer_Name
CU0001	Vieem A/L Hrithik Roshan
CU0002	Aditya Chopra A/P Iqra
CU0003	Tan Wei Xiong
CU0004	Ghadi bin Hadrami
CU0005	Ali Akmal

Order_Num	Item_Menu	Item_Price	Item_Quantity	Payment_ID
ORD0001	Green Tea	RM3.90	1	P0001
ORD0003	Chicken Burger	RM9.90	3	P0001
ORD0005	Mushroom Soup	RM5.90	2	P0001
ORD0006	Chocolate Cake	RM5.90	2	P0002
ORD0007	Milk Tea	RM3.90	1	P0002
ORD0008	French Fries	RM5.90	2	P0002
ORD0009	Oriental Chicken Chop	RM10.90	3	P0003
ORD0010	Lemon Tea	RM3.90	1	P0003
ORD0011	Green Tea	RM3.90	5	P0004
ORD0012	Spicy Chicken Burger	RM10.90	2	P0004
ORD0013	Spaghetti Bolognese	RM3.90	3	P0004
ORD0014	Milk Tea	RM3.90	1	P0005
ORD0015	Potato Wedges	RM5.90	2	P0005
ORD0016	Spaghetti Carbonara with Mushroom	RM9.90	1	P0005
ORD0017	Grilled Chicken with Mushroom Sauce	RM12.30	1	P0005
ORD0018	Tiramisu	RM6.90	1	P0005

Primary key Primary key

Customer_ID	Payment_ID
CU0001	P0001
CU0002	P0002
CU0003	P0003
CU0004	P0004
CU0005	P0005

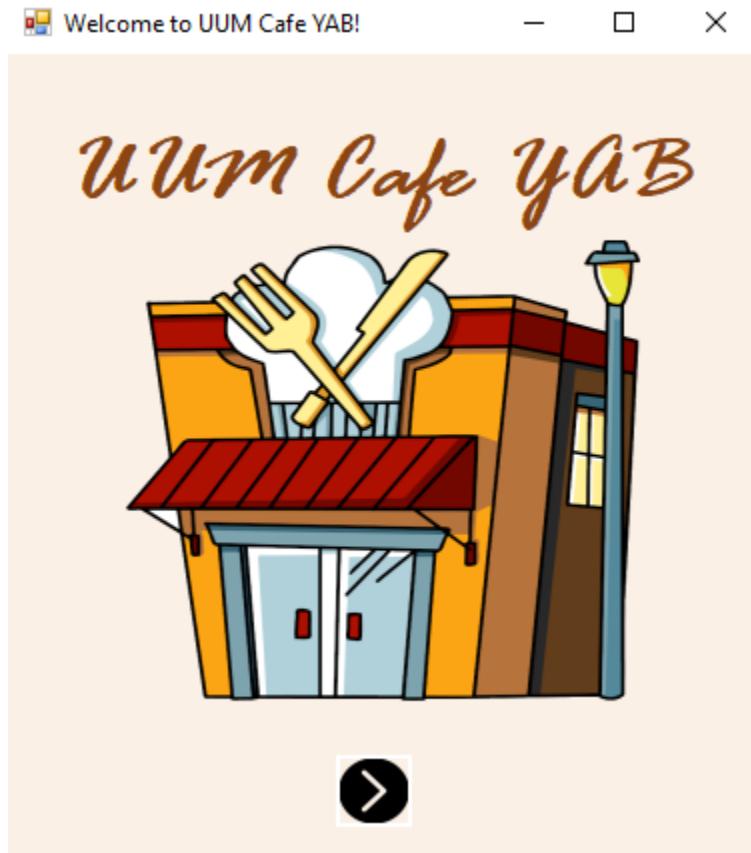
Primary key

Item_Menu	Item_Price
Green Tea	RM3.90
Chicken Burger	RM9.90
Mushroom Soup	RM5.90
Chocolate Cake	RM5.90
Milk Tea	RM3.90
French Fries	RM5.90
Oriental Chicken Chop	RM10.90
Lemon Tea	RM3.90
Spicy Chicken Burger	RM10.90
Spaghetti Bolognese	RM3.90
Potato Wedges	RM5.90
Spaghetti Carbonara with Mushroom	RM9.90
Grilled Chicken with Mushroom Sauce	RM12.30
Tiramisu	RM6.90

## 9.0 Interface and Coding

For this *UUM Cafe restaurant*, 11 forms are created to present records in the Visual Studio. The forms created are shown below:

**Form 1: frmWelcome**



### Coding

Public Class frmWelcome

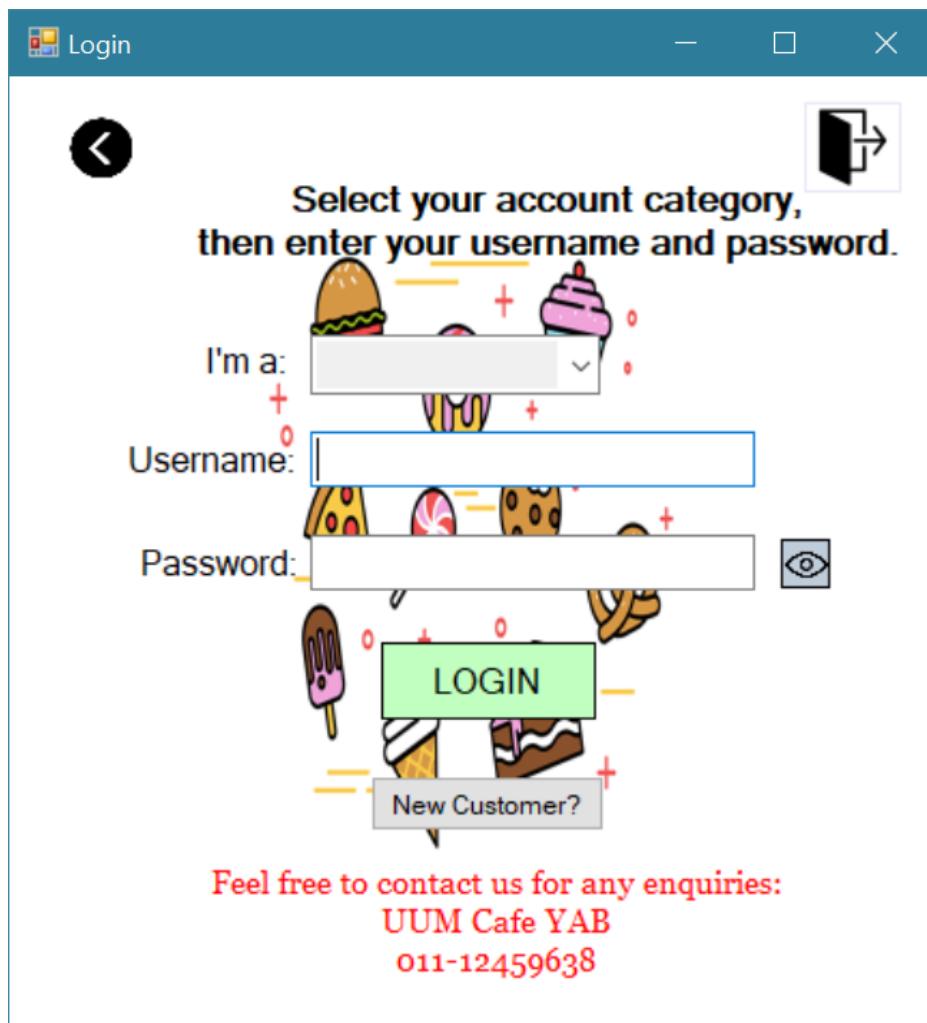
```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
    Me.Hide()
```

```
    frmLogin.Show()
```

```
End Sub
```

```
End Class
```

**Form 2: frmLogin**Coding

```
Imports System.Data.OleDb
```

```
Public Class frmLogin
```

```
    Dim con As New OleDbConnection
```

```
    Dim mouse_click As Integer
```

```
'Login
```

```
Private Sub cmdLogin_Click(sender As Object, e As EventArgs) Handles cmdLogin.Click
```

```
    mouse_click = mouse_click + 1
```

```
If mouse_click <= 3 Then
    con.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
    con.Open()

If txtComboBox.Text = "Customer" Then
    Dim strsql As String
    strsql = "select Customer_ID, Cu_Name , Cu_HP from Customer"
    Dim cmd As New OleDbCommand(strsql, con)
    Dim myreader As OleDbDataReader
    myreader = cmd.ExecuteReader
    myreader.Read()

    txtCustomerID.Text = myreader("Customer_ID")
    txtCustomerName.Text = myreader("Cu_Name")
    txtCuHP.Text = myreader("Cu_HP")

    Dim logincmd As OleDbCommand = New OleDbCommand("Select Cu_Username,
Cu_Password from CustomerAccount Where [Cu_Username]="" & txtLogin.Text &
""and[Cu_Password]="" & txtPassword.Text & "", con)
    Dim loginID As OleDbDataReader = logincmd.ExecuteReader
    If (loginID.Read() = True) Then

        Me.Hide()
        MsgBox("Login Succeeded!", MessageBoxIcon.Information, "UUM Cafe YAB")
        frmCustMainMenu.Show()
    Else
        MsgBox("Login Fails!", MessageBoxIcon.Exclamation, "UUM Cafe YAB")
    End If
```

```
End If
```

```
If txtComboBox.Text = "Staff" Then
```

```
    Dim strsql As String
```

```
    strsql = "select Staff_ID, Staff_Name from StaffAccount Where [Staff_Username]=""
```

```
& txtLogin.Text & "and[Staff_Password]="" & txtPassword.Text & """
```

```
    Dim cmd As New OleDbCommand(strsql, con)
```

```
    Dim myreader As OleDbDataReader
```

```
    myreader = cmd.ExecuteReader
```

```
    myreader.Read()
```

```
    txtStaffID.Text = myreader("Staff_ID")
```

```
    txtStaffName.Text = myreader("Staff_Name")
```

```
    Dim logincmd As OleDbCommand = New OleDbCommand("Select Staff_Username,  
Staff_Password from StaffAccount Where [Staff_Username]="" & txtLogin.Text &  
""and[Staff_Password]="" & txtPassword.Text & "", con)
```

```
    Dim loginID As OleDbDataReader = logincmd.ExecuteReader
```

```
    If (loginID.Read() = True) Then
```

```
        Me.Hide()
```

```
        MsgBox("Login Succeeded!", MessageBoxIcon.Information, "UUM Cafe YAB")
```

```
        frmStaff.Show()
```

```
    Else
```

```
        MsgBox("Login Fails!", MessageBoxIcon.Exclamation, "UUM Cafe YAB")
```

```
    End If
```

```
End If
```

```
If txtComboBox.Text = "Rider" Then
```

```
    Dim strsql As String
```

```
strsql = "select Rider_ID from RiderAccount Where [Rider_Username]="" &
txtLogin.Text & "and[Rider_Password]="" & txtPassword.Text & """
Dim cmd As New OleDbCommand(strsql, con)
Dim myreader As OleDbDataReader
myreader = cmd.ExecuteReader
myreader.Read()

txtRiderID.Text = myreader("Rider_ID")

Dim logincmd As OleDbCommand = New OleDbCommand("Select Rider_Username,
Rider_Password from RiderAccount Where [Rider_Username]="" & txtLogin.Text &
""and[Rider_Password]="" & txtPassword.Text & "", con)
Dim loginID As OleDbDataReader = logincmd.ExecuteReader
If (loginID.Read() = True) Then
    Me.Hide()
    MsgBox("Login Succeeded!", MessageBoxIcon.Information, "UUM Cafe YAB")
    frmRiderDuty.Show()
Else
    MsgBox("Login Fails!", MessageBoxIcon.Exclamation, "UUM Cafe YAB")
End If

End If
con.Close()

Else
    MsgBox("You have typed wrong 3 times, please contact us. 011-12459638. System is
closing now.", MessageBoxIcon.Exclamation, "UUM Cafe YAB")
    Me.Close()
End If
```

```
End Sub
```

```
'New Customer
```

```
Private Sub cmdNewCu_Click(sender As Object, e As EventArgs) Handles cmdNewCu.Click
    Me.Hide()
    frmNewCu.Show()
End Sub
```

```
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
```

```
    Me.Close()
    frmWelcome.Show()
End Sub
```

```
Private Sub Button2_Click(sender As Object, e As EventArgs) Handles Button2.Click
```

```
    If txtPassword.UseSystemPasswordChar = True Then
        txtPassword.UseSystemPasswordChar = False
    Else
        txtPassword.UseSystemPasswordChar = True
    End If
End Sub
```

```
Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
```

```
    Dim user_respond As Integer
    user_respond = MessageBox.Show("Are you sure to exit the application?", "EXIT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)
```

```
If user_respond = vbYes Then
```

```
    MessageBox.Show("Thank you for using our application." & vbCrLf & "We hope to
see you again!", "BYE~")
```

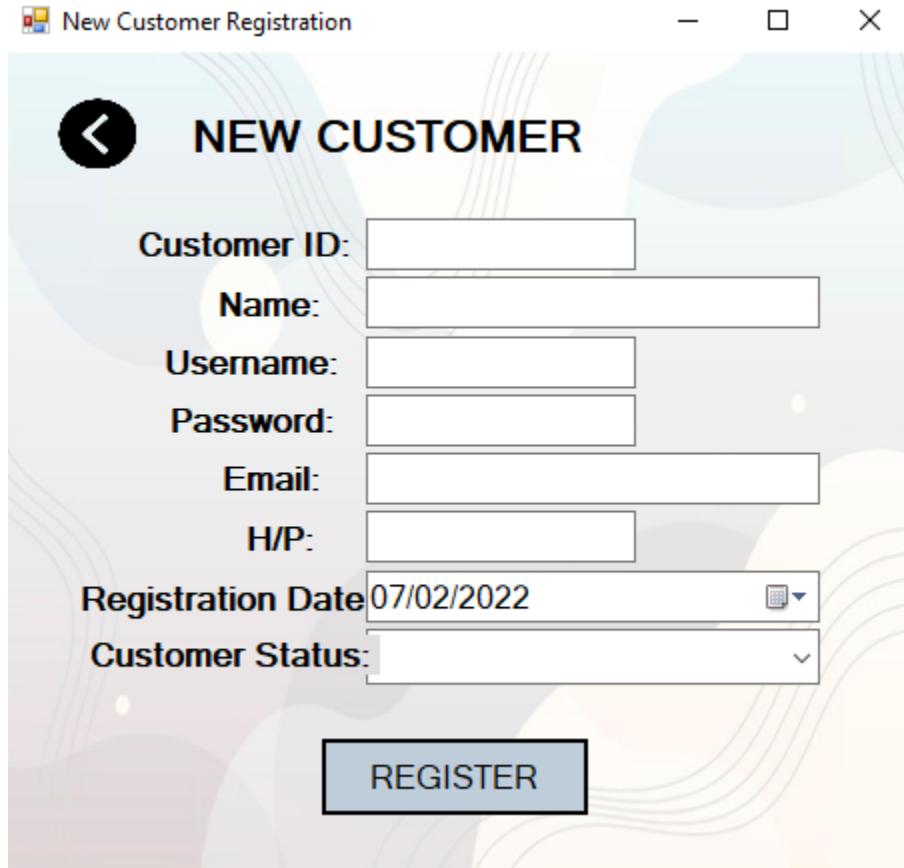
```
    Application.Exit()
```

```
End If
```

End Sub

End Class

### Form 3: frmNewCu



### Coding

Imports System.Data.OleDb

Public Class frmNewCu

Dim pro As String

Dim connstring As String

Dim command As String

Dim myconnection As OleDbConnection = New OleDbConnection

```

Private Sub cmdAddNew_Click(sender As Object, e As EventArgs) Handles
cmdAddNew.Click
    pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
    connstring = pro
    myconnection.ConnectionString = connstring
    myconnection.Open()

    command = "insert into Customer ([Customer_ID] , [Cu_Name], [Cu_Email] , [Cu_HP] ,
[Reg_Date] , [Cu_Status]) values ("" & txtCuID.Text & ""","" & txtCuName.Text & "","" &
txtCuEmail.Text & "","" & txtCuHP.Text & "","" & txtCuRegistration.Value & "","" &
txtCuStatus.Text & "")"

    Dim cmddetail As OleDbCommand = New OleDbCommand(command, myconnection)
    cmddetail.Parameters.Add(New OleDbParameter("Customer_ID", CType(txtCuID.Text,
String)))
    cmddetail.Parameters.Add(New OleDbParameter("Cu_Name", CType(txtCuName.Text,
String)))
    cmddetail.Parameters.Add(New OleDbParameter("Cu_Email", CType(txtCuEmail.Text,
String)))
    cmddetail.Parameters.Add(New OleDbParameter("Cu_HP", CType(txtCuHP.Text, String)))
    cmddetail.Parameters.Add(New OleDbParameter("Reg_Date",
CType(txtCuRegistration.Value, Date)))
    cmddetail.Parameters.Add(New OleDbParameter("Cu_Status", CType(txtCuStatus.Text,
String)))
    cmddetail.ExecuteNonQuery()

    command = "insert into CustomerAccount ([Customer_ID] , [Cu_Name], [Cu_Username]
, [Cu_Password]) values ("" & txtCuID.Text & ""","" & txtCuName.Text & "','" &
txtCuUsername.Text & "","" & txtCuPassword.Text & ")"

    Dim cmdcuacc As OleDbCommand = New OleDbCommand(command, myconnection)
    cmdcuacc.Parameters.Add(New OleDbParameter("Customer_ID", CType(txtCuID.Text,
String)))

```

```

cmdcuacc.Parameters.Add(New OleDbParameter("Cu_Name", CType(txtCuName.Text,
String)))
cmdcuacc.Parameters.Add(New OleDbParameter("Cu_Username",
CType(txtCuUsername.Text, String)))
cmdcuacc.Parameters.Add(New OleDbParameter("Cu_Password",
CType(txtCuPassword.Text, String)))
MsgBox("Register Successful!")
frmLogin.Show()

```

Try

```

cmdcuacc.ExecuteNonQuery()
cmdcuacc.Dispose()
myconnection.Close()
txtCuID.Clear()
txtCuName.Clear()
txtCuUsername.Clear()
txtCuPassword.Clear()
txtCuEmail.Clear()
txtCuHP.Clear()
txtCuStatus.Text = String.Empty
txtCuRegistration.Value = DateTime.Now

```

Catch ex As Exception

```
    MsgBox(ex.Message)
```

End Try

End Sub

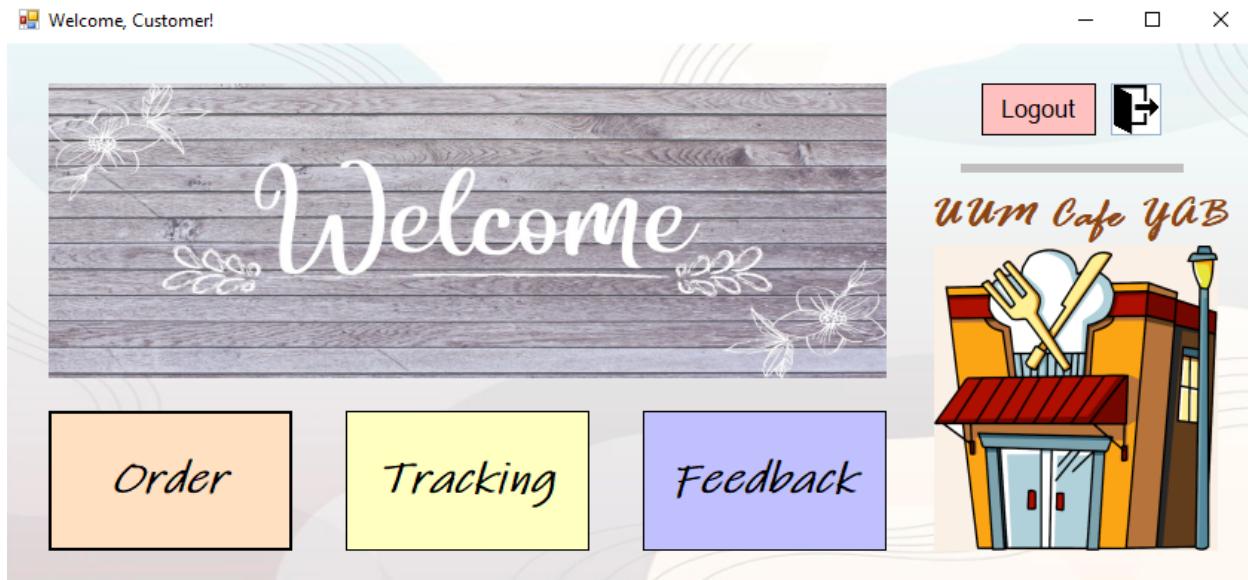
```

Private Sub PictureBox1_Click(sender As Object, e As EventArgs) Handles
PictureBox1.Click
Me.Close()
frmLogin.Show()
End Sub

```

End Class

#### Form 4: frmCustMainMenu



#### Coding

Public Class frmCustMainMenu

```
Private Sub cmdOrder_Click(sender As Object, e As EventArgs) Handles cmdOrder.Click
    Me.Hide()
    frmOrder.Show()
End Sub
```

```
Private Sub cmdTracking_Click(sender As Object, e As EventArgs) Handles
cmdTracking.Click
    Me.Hide()
    frmTracking.Show()
End Sub
```

```
Private Sub cmdFeedback_Click(sender As Object, e As EventArgs) Handles
cmdFeedback.Click
    Me.Hide()
```

```
frmFeedback.Show()
End Sub

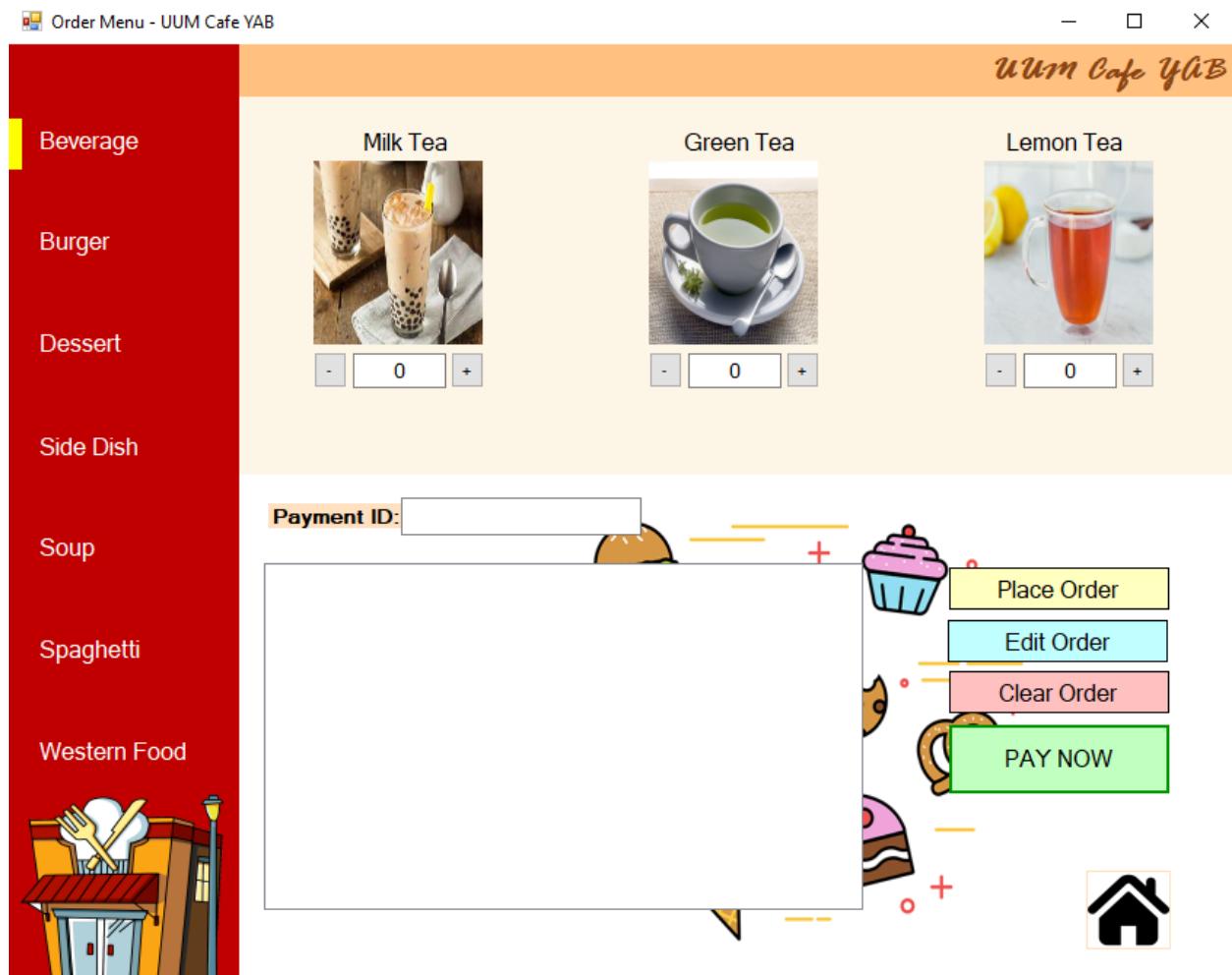
Private Sub cmdLogout_Click(sender As Object, e As EventArgs) Handles cmdLogout.Click
    Dim user_respond As Integer
    user_respond = MessageBox.Show("Are you sure to log out?", "LOGOUT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)

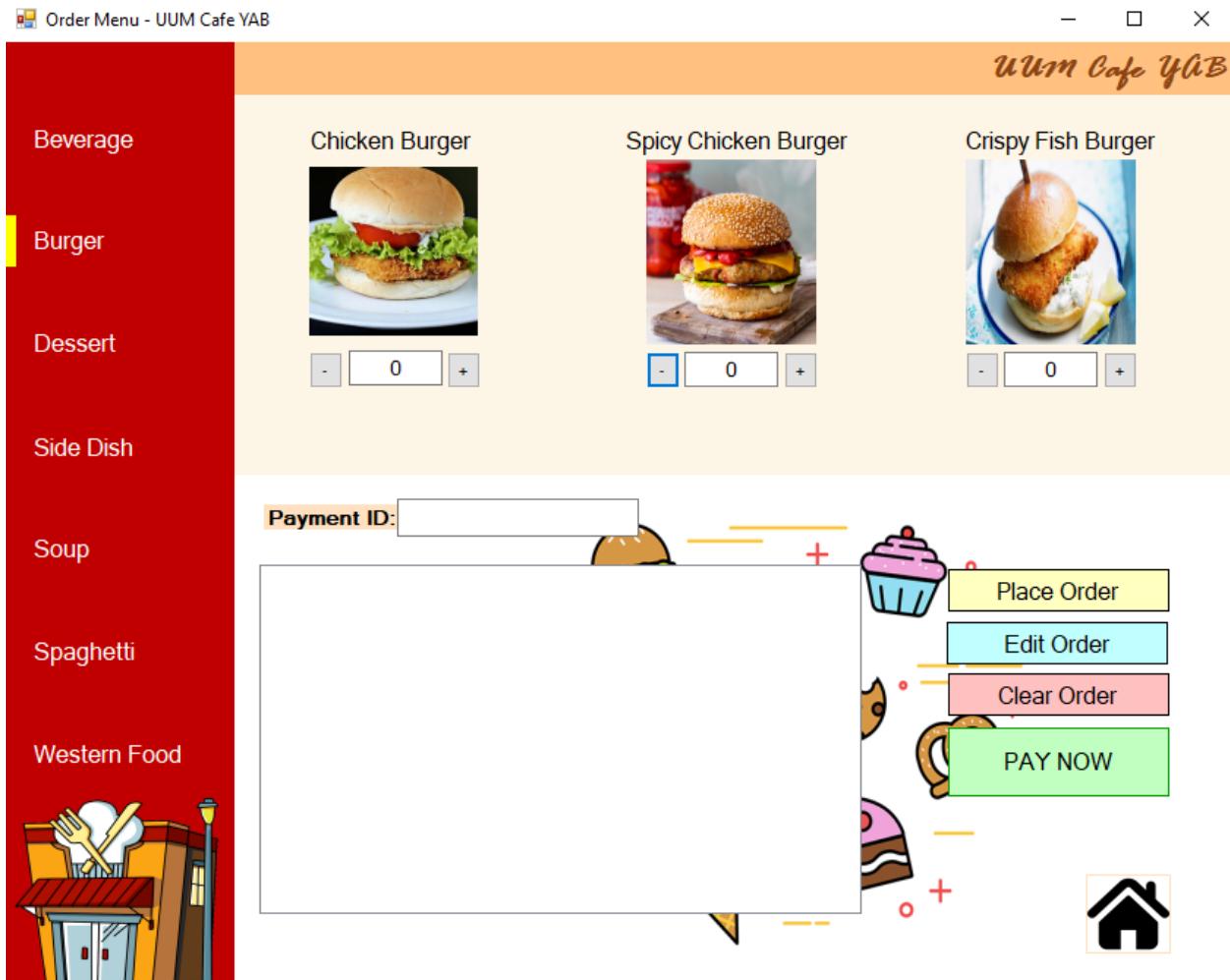
    If user_respond = vbYes Then
        Me.Hide()
        frmLogin.Show()
    End If
End Sub

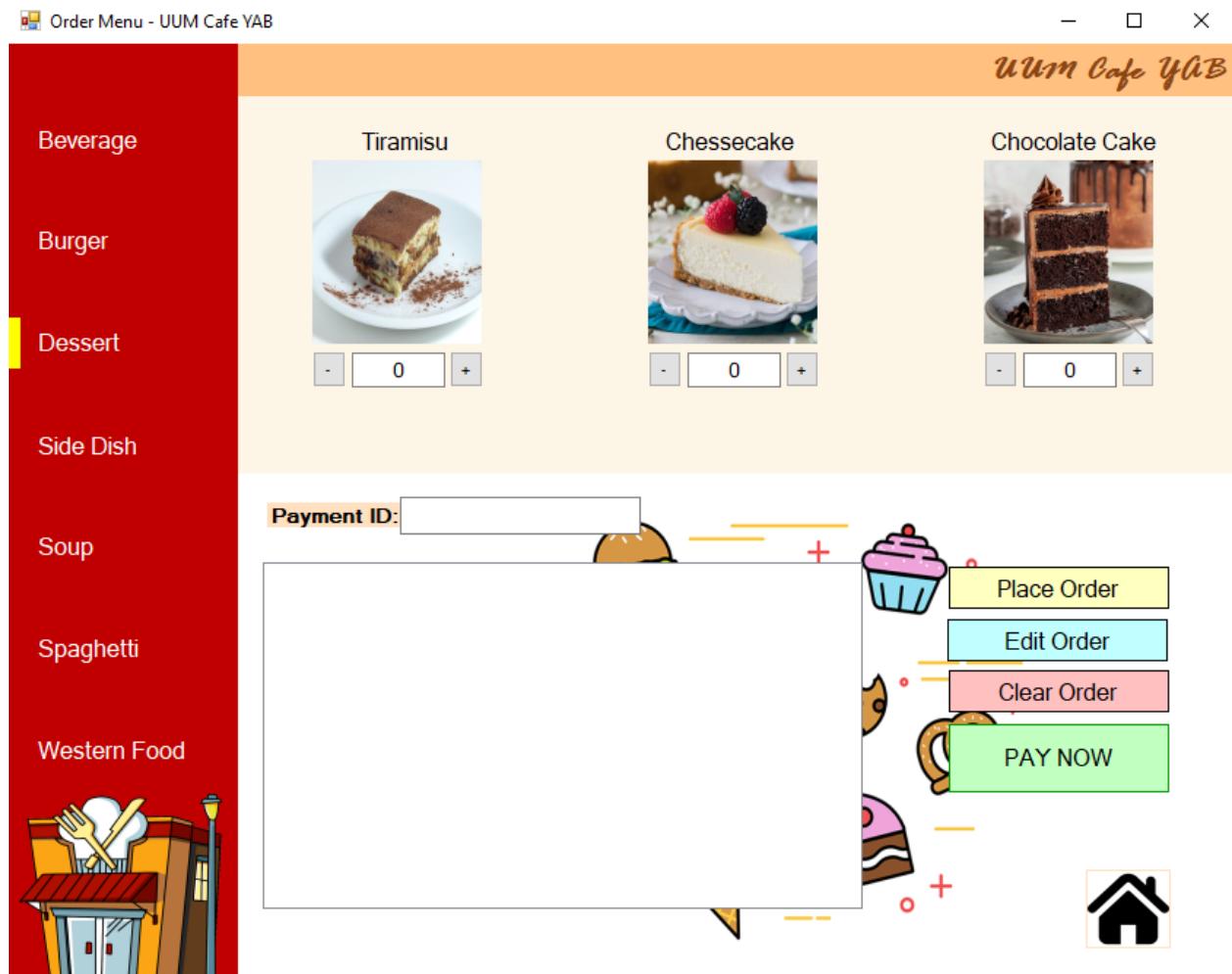
Private Sub Button1_Click(sender As Object, e As EventArgs) Handles Button1.Click
    Dim user_respond As Integer
    user_respond = MessageBox.Show("Are you sure to exit the application?", "EXIT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)

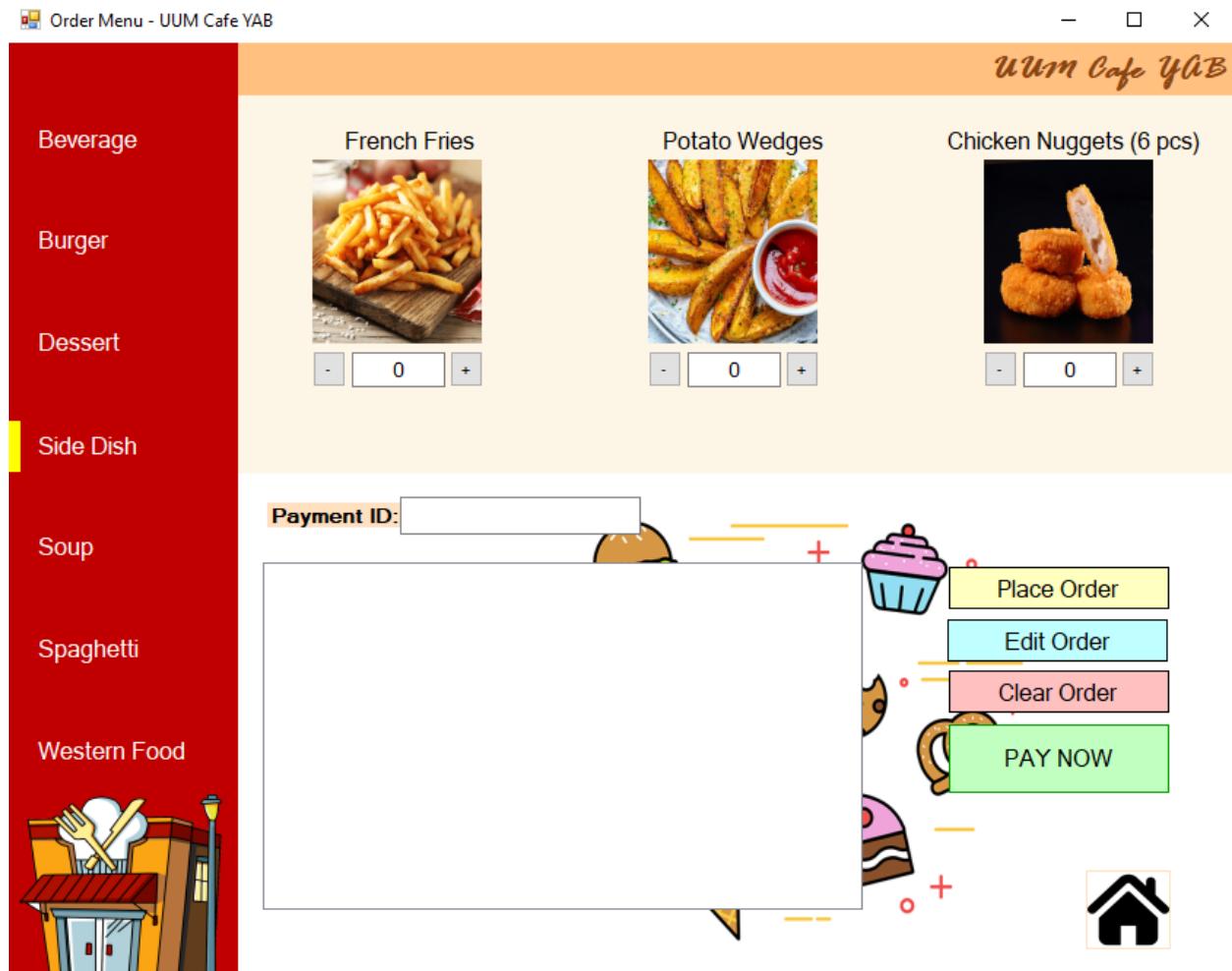
    If user_respond = vbYes Then
        MessageBox.Show("Thank you for using our application." & vbCrLf & "We hope to
see you again!", "BYE~")
        Application.Exit()
    End If
End Sub

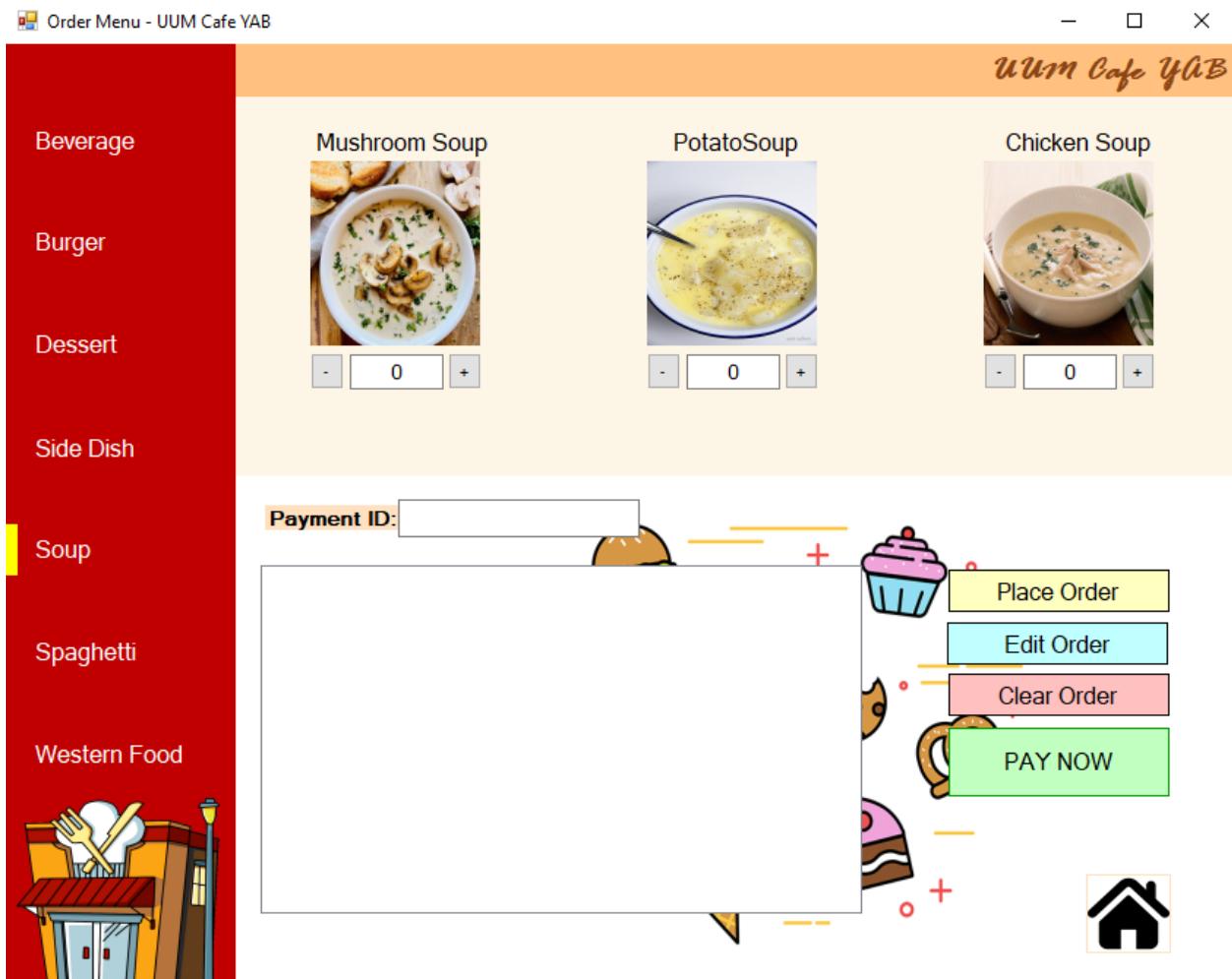
End Class
```

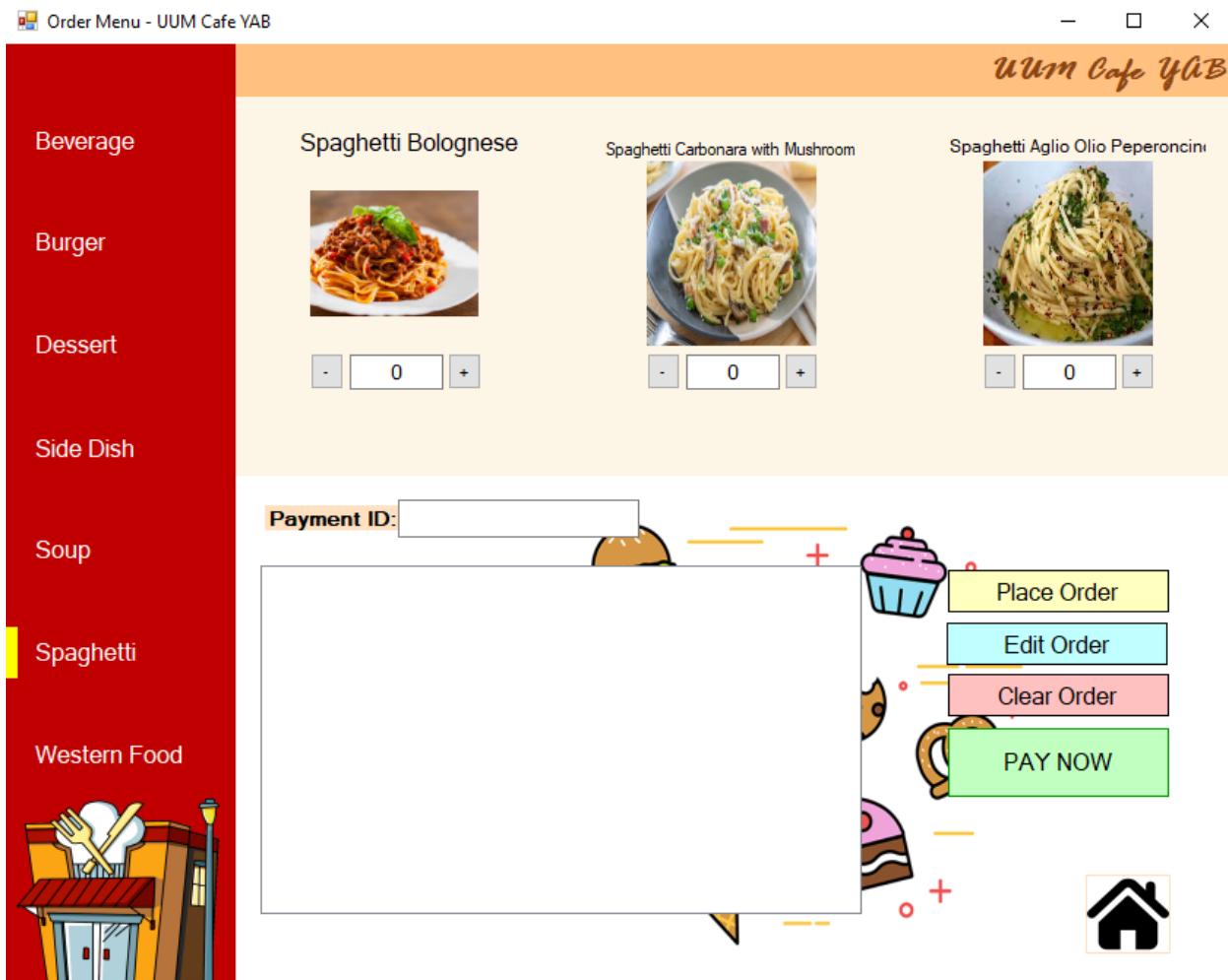
**Form 5: frmOrder**

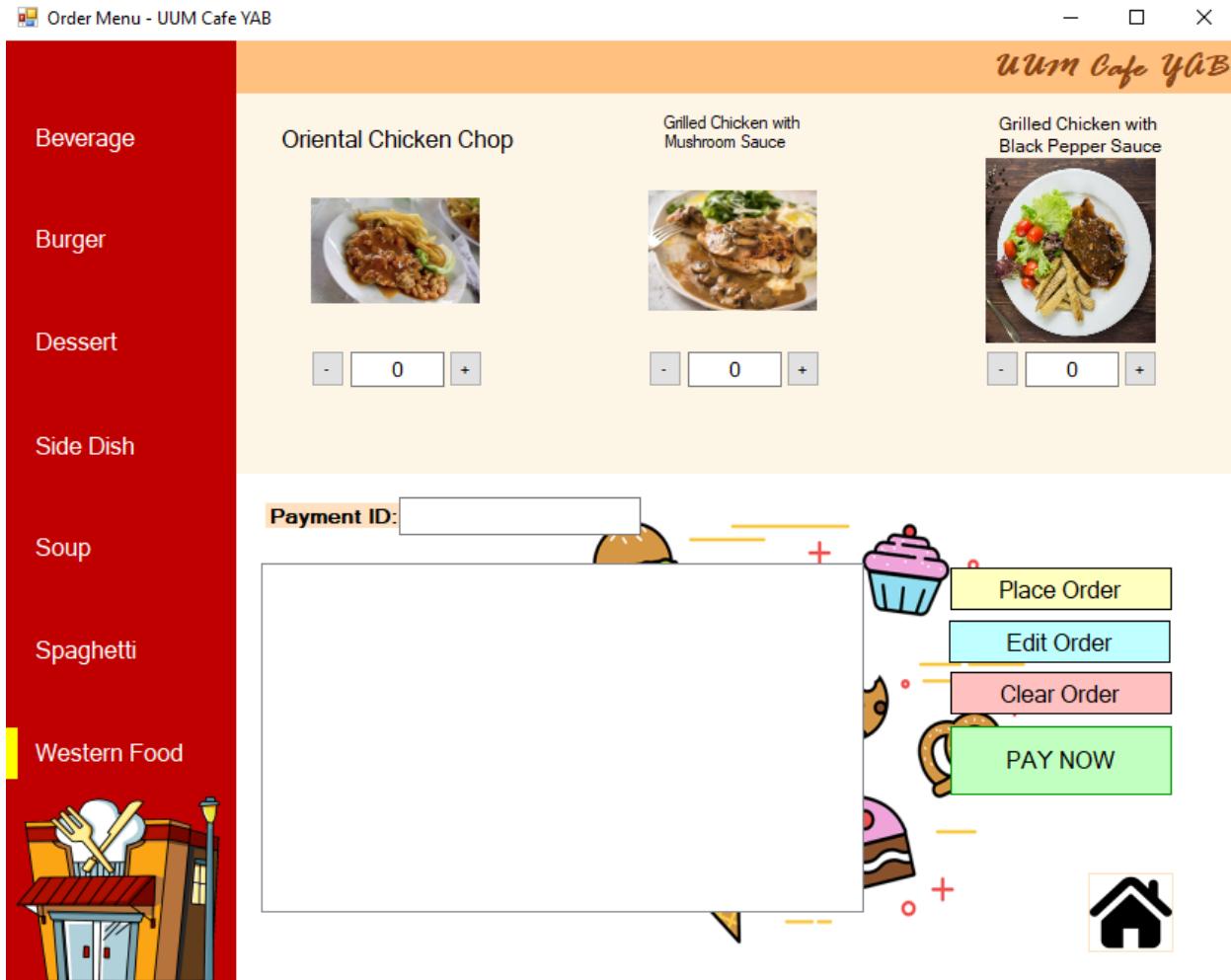












## Coding

```
Imports System.Data.OleDb
```

```
Public Class frmOrder
```

```
    Dim pro As String
```

```
    Dim connstring As String
```

```
    Dim commmand As String
```

```
    Dim myconnection As OleDbConnection = New OleDbConnection
```

```
Private Sub CustomerOrderBindingNavigatorSaveItem_Click(sender As Object, e As EventArgs) Handles CustomerOrderBindingNavigatorSaveItem.Click
```

```
    Me.Validate()
```

```
    Me.CustomerOrderBindingSource.EndEdit()
```

```
Me.TableAdapterManager.UpdateAll(Me.UUMCafeDatabase5DataSet)
```

```
End Sub
```

```
Private Sub frmOrder_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
'TODO: This line of code loads data into the 'UUMCafeDatabase5DataSet.CustomerOrder' table. You can move, or remove it, as needed.
```

```
Me.CustomerOrderTableAdapter.Fill(Me.UUMCafeDatabase5DataSet.CustomerOrder)
```

```
txtCustomerID.Text = frmLogin.txtCustomerID.Text
```

```
txtCustomerName.Text = frmLogin.txtCustomerName.Text
```

```
End Sub
```

```
'User view
```

```
Private Sub btnBeverage_Click(sender As Object, e As EventArgs) Handles
```

```
btnBeverage.Click
```

```
pnlOnButtonPosition.Height = btnBeverage.Height
```

```
pnlOnButtonPosition.Top = btnBeverage.Top
```

```
pnlBeverage.Visible = True
```

```
pnlBurger.Visible = False
```

```
pnlDessert.Visible = False
```

```
pnlSideDish.Visible = False
```

```
pnlSoup.Visible = False
```

```
pnlSpaghetti.Visible = False
```

```
pnlWesternFood.Visible = False
```

```
End Sub
```

```
Private Sub btnBurger_Click(sender As Object, e As EventArgs) Handles btnBurger.Click
```

```
pnlOnButtonPosition.Height = btnBurger.Height
```

```
pnlOnButtonPosition.Top = btnBurger.Top
```

```
pnlBeverage.Visible = False
```

```
pnlBurger.Visible = True  
pnlDessert.Visible = False  
pnlSideDish.Visible = False  
pnlSoup.Visible = False  
pnlSpaghetti.Visible = False  
pnlWesternFood.Visible = False
```

End Sub

```
Private Sub btnDessert_Click(sender As Object, e As EventArgs) Handles btnDessert.Click  
    pnlOnButtonPosition.Height = btnDessert.Height  
    pnlOnButtonPosition.Top = btnDessert.Top  
    pnlBeverage.Visible = False  
    pnlBurger.Visible = False  
    pnlDessert.Visible = True  
    pnlSideDish.Visible = False  
    pnlSoup.Visible = False  
    pnlSpaghetti.Visible = False  
    pnlWesternFood.Visible = False
```

End Sub

```
Private Sub btnSideDish_Click(sender As Object, e As EventArgs) Handles  
    btnSideDish.Click  
    pnlOnButtonPosition.Height = btnSideDish.Height  
    pnlOnButtonPosition.Top = btnSideDish.Top  
    pnlBeverage.Visible = False  
    pnlBurger.Visible = False  
    pnlDessert.Visible = False  
    pnlSideDish.Visible = True  
    pnlSoup.Visible = False  
    pnlSpaghetti.Visible = False  
    pnlWesternFood.Visible = False
```

End Sub

```
Private Sub btnSoup_Click(sender As Object, e As EventArgs) Handles btnSoup.Click
    pnlOnButtonPosition.Height = btnSoup.Height
    pnlOnButtonPosition.Top = btnSoup.Top
    pnlBeverage.Visible = False
    pnlBurger.Visible = False
    pnlDessert.Visible = False
    pnlSideDish.Visible = False
    pnlSoup.Visible = True
    pnlSpaghetti.Visible = False
    pnlWesternFood.Visible = False
```

End Sub

```
Private Sub btnSpaghetti_Click(sender As Object, e As EventArgs) Handles
btnSpaghetti.Click
    pnlOnButtonPosition.Height = btnSpaghetti.Height
    pnlOnButtonPosition.Top = btnSpaghetti.Top
    pnlBeverage.Visible = False
    pnlBurger.Visible = False
    pnlDessert.Visible = False
    pnlSideDish.Visible = False
    pnlSoup.Visible = False
    pnlSpaghetti.Visible = True
    pnlWesternFood.Visible = False
```

End Sub

```
Private Sub btnWesternFood_Click(sender As Object, e As EventArgs) Handles
btnWesternFood.Click
    pnlOnButtonPosition.Height = btnWesternFood.Height
    pnlOnButtonPosition.Top = btnWesternFood.Top
```

```
pnlBeverage.Visible = False  
pnlBurger.Visible = False  
pnlDessert.Visible = False  
pnlSideDish.Visible = False  
pnlSoup.Visible = False  
pnlSpaghetti.Visible = False  
pnlWesternFood.Visible = True  
End Sub
```

'Making order

```
Private Sub btnMilkTeaAdd_Click(sender As Object, e As EventArgs) Handles  
btnMilkTeaAdd.Click  
    txtMilkTeaAmount.Text += 1  
End Sub
```

```
Private Sub btnMilkTeaMinus_Click(sender As Object, e As EventArgs) Handles  
btnMilkTeaMinus.Click  
    txtMilkTeaAmount.Text -= 1  
    If txtMilkTeaAmount.Text <= 0 Then  
        txtMilkTeaAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnGreenTeaAdd_Click(sender As Object, e As EventArgs) Handles  
btnGreenTeaAdd.Click  
    txtGreenTeaAmount.Text += 1  
End Sub
```

```
Private Sub btnGreenTeaMinus_Click(sender As Object, e As EventArgs) Handles  
btnGreenTeaMinus.Click  
    txtGreenTeaAmount.Text -= 1
```

```
If txtGreenTeaAmount.Text <= 0 Then  
    txtGreenTeaAmount.Text = 0  
End If  
End Sub
```

```
Private Sub btnLemonTeaAdd_Click(sender As Object, e As EventArgs) Handles  
btnLemonTeaAdd.Click  
    txtLemonTeaAmount.Text += 1  
End Sub
```

```
Private Sub btnLemonTeaMinus_Click(sender As Object, e As EventArgs) Handles  
btnLemonTeaMinus.Click  
    txtLemonTeaAmount.Text -= 1  
    If txtLemonTeaAmount.Text <= 0 Then  
        txtLemonTeaAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnChickenBurgerAdd_Click(sender As Object, e As EventArgs) Handles  
btnChickenBurgerAdd.Click  
    txtChickenBurgerAmount.Text += 1  
End Sub
```

```
Private Sub btnChickenBurgerMinus_Click(sender As Object, e As EventArgs) Handles  
btnChickenBurgerMinus.Click  
    txtChickenBurgerAmount.Text -= 1  
    If txtChickenBurgerAmount.Text <= 0 Then  
        txtChickenBurgerAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnSpicyChickenBurgerAdd_Click(sender As Object, e As EventArgs) Handles  
btnSpicyChickenBurgerAdd.Click  
    txtSpicyChickenBurgerAmount.Text += 1  
End Sub
```

```
Private Sub btnSpicyChickenBurgerMinus_Click(sender As Object, e As EventArgs) Handles  
btnSpicyChickenBurgerMinus.Click  
    txtSpicyChickenBurgerAmount.Text -= 1  
    If txtSpicyChickenBurgerAmount.Text <= 0 Then  
        txtSpicyChickenBurgerAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnCrispyFishBurgerAdd_Click(sender As Object, e As EventArgs) Handles  
btnCrispyFishBurgerAdd.Click  
    txtCrispyFishBurgerAmount.Text += 1  
End Sub
```

```
Private Sub btnCrispyFishBurgerMinus_Click(sender As Object, e As EventArgs) Handles  
btnCrispyFishBurgerMinus.Click  
    txtCrispyFishBurgerAmount.Text -= 1  
    If txtCrispyFishBurgerAmount.Text <= 0 Then  
        txtCrispyFishBurgerAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnTiramisuAdd_Click(sender As Object, e As EventArgs) Handles  
btnTiramisuAdd.Click  
    txtTiramisuAmount.Text += 1  
End Sub
```

```
Private Sub btnTiramisuMinus_Click(sender As Object, e As EventArgs) Handles  
btnTiramisuMinus.Click  
    txtTiramisuAmount.Text -= 1  
    If txtTiramisuAmount.Text <= 0 Then  
        txtTiramisuAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnCheesecakeAdd_Click(sender As Object, e As EventArgs) Handles  
btnCheesecakeAdd.Click  
    txtCheesecakeAmount.Text += 1  
End Sub
```

```
Private Sub btnCheesecakeMinus_Click(sender As Object, e As EventArgs) Handles  
btnCheesecakeMinus.Click  
    txtCheesecakeAmount.Text -= 1  
    If txtCheesecakeAmount.Text <= 0 Then  
        txtCheesecakeAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnChocolateCakeAdd_Click(sender As Object, e As EventArgs) Handles  
btnChocolateCakeAdd.Click  
    txtChocolateCakeAmount.Text += 1  
End Sub
```

```
Private Sub btnChocolateCakeMinus_Click(sender As Object, e As EventArgs) Handles  
btnChocolateCakeMinus.Click  
    txtChocolateCakeAmount.Text -= 1  
    If txtChocolateCakeAmount.Text <= 0 Then  
        txtChocolateCakeAmount.Text = 0
```

End If

End Sub

```
Private Sub btnFrenchFriesAdd_Click(sender As Object, e As EventArgs) Handles  
btnFrenchFriesAdd.Click  
    txtFrenchFriesAmount.Text += 1  
End Sub
```

```
Private Sub btnFrenchFriesMinus_Click(sender As Object, e As EventArgs) Handles  
btnFrenchFriesMinus.Click  
    txtFrenchFriesAmount.Text -= 1  
    If txtFrenchFriesAmount.Text <= 0 Then  
        txtFrenchFriesAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnPotatoWedgesAdd_Click(sender As Object, e As EventArgs) Handles  
btnPotatoWedgesAdd.Click  
    txtPotatoWedgesAmount.Text += 1  
End Sub
```

```
Private Sub btnPotatoWedgesMinus_Click(sender As Object, e As EventArgs) Handles  
btnPotatoWedgesMinus.Click  
    txtPotatoWedgesAmount.Text -= 1  
    If txtPotatoWedgesAmount.Text <= 0 Then  
        txtPotatoWedgesAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnChickenNuggetAdd_Click(sender As Object, e As EventArgs) Handles  
btnChickenNuggetAdd.Click
```

```
txtChickenNuggetAmount.Text += 1  
End Sub
```

```
Private Sub btnChickenNuggetMinus_Click(sender As Object, e As EventArgs) Handles  
btnChickenNuggetMinus.Click  
    txtChickenNuggetAmount.Text -= 1  
    If txtChickenNuggetAmount.Text <= 0 Then  
        txtChickenNuggetAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnMushroomSoupAdd_Click(sender As Object, e As EventArgs) Handles  
btnMushroomSoupAdd.Click  
    txtMushroomSoupAmount.Text += 1  
End Sub
```

```
Private Sub btnMushroomSoupMinus_Click(sender As Object, e As EventArgs) Handles  
btnMushroomSoupMinus.Click  
    txtMushroomSoupAmount.Text -= 1  
    If txtMushroomSoupAmount.Text <= 0 Then  
        txtMushroomSoupAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnPotatoSoupAdd_Click(sender As Object, e As EventArgs) Handles  
btnPotatoSoupAdd.Click  
    txtPotatoSoupAmount.Text += 1  
End Sub
```

```
Private Sub btnPotatoSoupMinus_Click(sender As Object, e As EventArgs) Handles  
btnPotatoSoupMinus.Click
```

```
txtPotatoSoupAmount.Text -= 1  
If txtPotatoSoupAmount.Text <= 0 Then  
    txtPotatoSoupAmount.Text = 0  
End If  
End Sub
```

```
Private Sub btnChickenSoupAdd_Click(sender As Object, e As EventArgs) Handles  
btnChickenSoupAdd.Click  
    txtChickenSoupAmount.Text += 1  
End Sub
```

```
Private Sub btnChickenSoupMinus_Click(sender As Object, e As EventArgs) Handles  
btnChickenSoupMinus.Click  
    txtChickenSoupAmount.Text -= 1  
    If txtChickenSoupAmount.Text <= 0 Then  
        txtChickenSoupAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnSpaghettiBologneseAdd_Click(sender As Object, e As EventArgs) Handles  
btnSpaghettiBologneseAdd.Click  
    txtSpaghettiBologneseAmount.Text += 1  
End Sub
```

```
Private Sub btnSpaghettiBologneseMinus_Click(sender As Object, e As EventArgs) Handles  
btnSpaghettiBologneseMinus.Click  
    txtSpaghettiBologneseAmount.Text -= 1  
    If txtSpaghettiBologneseAmount.Text <= 0 Then  
        txtSpaghettiBologneseAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnSpaghettiCarbonaraMushroomAdd_Click(sender As Object, e As EventArgs)
Handles btnSpaghettiCarbonaraMushroomAdd.Click
    txtSpaghettiCarbonaraMushroomAmount.Text += 1
End Sub
```

```
Private Sub btnSpaghettiCarbonaraMushroomMinus_Click(sender As Object, e As EventArgs)
Handles btnSpaghettiCarbonaraMushroomMinus.Click
    txtSpaghettiCarbonaraMushroomAmount.Text -= 1
    If txtSpaghettiCarbonaraMushroomAmount.Text <= 0 Then
        txtSpaghettiCarbonaraMushroomAmount.Text = 0
    End If
End Sub
```

```
Private Sub btnSpaghettiAglioOlioAdd_Click(sender As Object, e As EventArgs)
Handles btnSpaghettiAglioOlioAdd.Click
    txtSpaghettiAglioOlioAmount.Text += 1
End Sub
```

```
Private Sub btnSpaghettiAglioOlioMinus_Click(sender As Object, e As EventArgs)
Handles btnSpaghettiAglioOlioMinus.Click
    txtSpaghettiAglioOlioAmount.Text -= 1
    If txtSpaghettiAglioOlioAmount.Text <= 0 Then
        txtSpaghettiAglioOlioAmount.Text = 0
    End If
End Sub
```

```
Private Sub btnChickenChopAdd_Click(sender As Object, e As EventArgs)
Handles btnChickenChopAdd.Click
    txtChickenChopAmount.Text += 1
End Sub
```

```
Private Sub btnChickenChopMinus_Click(sender As Object, e As EventArgs) Handles  
btnChickenChopMinus.Click  
    txtChickenChopAmount.Text -= 1  
    If txtChickenChopAmount.Text <= 0 Then  
        txtChickenChopAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnGrilledChickenMushroomAdd_Click(sender As Object, e As EventArgs)  
Handles btnGrilledChickenMushroomAdd.Click  
    txtGrilledChickenMushroomAmount.Text += 1  
End Sub
```

```
Private Sub btnGrilledChickenMushroomMinus_Click(sender As Object, e As EventArgs)  
Handles btnGrilledChickenMushroomMinus.Click  
    txtGrilledChickenMushroomAmount.Text -= 1  
    If txtGrilledChickenMushroomAmount.Text <= 0 Then  
        txtGrilledChickenMushroomAmount.Text = 0  
    End If  
End Sub
```

```
Private Sub btnGrilledChickenBlackPepperAdd_Click(sender As Object, e As EventArgs)  
Handles btnGrilledChickenBlackPepperAdd.Click  
    txtGrilledChickenBlackPepperAmount.Text += 1  
End Sub
```

```
Private Sub btnGrilledChickenBlackPepperMinus_Click(sender As Object, e As EventArgs)  
Handles btnGrilledChickenBlackPepperMinus.Click  
    txtGrilledChickenBlackPepperAmount.Text -= 1  
    If txtGrilledChickenBlackPepperAmount.Text <= 0 Then
```

```
txtGrilledChickenBlackPepperAmount.Text = 0
End If
End Sub

'Place order
Private Sub txtPlaceOrder_Click(sender As Object, e As EventArgs) Handles
btnPlaceOrder.Click

lstPlaceOrderCheck.Items.Clear()

Dim MilkTeaSubPrice As Single
Dim GreenTeaSubPrice As Single
Dim LemonTeaSubPrice As Single
Dim ChickenBurgerSubPrice As Single
Dim SpicyChickenBurgerSubPrice As Single
Dim CrispyFishBurgerSubPrice As Single
Dim TiramisuSubPrice As Single
Dim CheesecakeSubPrice As Single
Dim ChocolateCakeSubPrice As Single
Dim FrenchFriesSubPrice As Single
Dim PotatoWedgesSubPrice As Single
Dim ChickenNuggetSubPrice As Single
Dim MushroomSoupSubPrice As Single
Dim PotatoSoupSubPrice As Single
Dim ChickenSoupSubPrice As Single
Dim SpaghettiBologneseSubPrice As Single
Dim SpaghettiCarbonaraMushroomSubPrice As Single
Dim SpaghettiAglioOlioSubPrice As Single
Dim ChickenChopSubPrice As Single
Dim GrilledChickenMushroomSubPrice As Single
Dim GrilledChickenBlackPepperSubPrice As Single
```

## Dim TotalPrice As Single

```

MilkTeaSubPrice = Format(txtMilkTeaAmount.Text * 3.9, "0.00")
GreenTeaSubPrice = Format(txtGreenTeaAmount.Text * 3.9, "0.00")
LemonTeaSubPrice = Format(txtLemonTeaAmount.Text * 3.9, "0.00")
ChickenBurgerSubPrice = Format(txtChickenBurgerAmount.Text * 9.9, "0.00")
SpicyChickenBurgerSubPrice = Format(txtSpicyChickenBurgerAmount.Text * 10.9,
"0.00")
CrispyFishBurgerSubPrice = Format(txtCrispyFishBurgerAmount.Text * 9.9, "0.00")
TiramisuSubPrice = Format(txtTiramisuAmount.Text * 6.9, "0.00")
CheesecakeSubPrice = Format(txtCheesecakeAmount.Text * 6.9, "0.00")
ChocolateCakeSubPrice = Format(txtChocolateCakeAmount.Text * 5.9, "0.00")
FrenchFriesSubPrice = Format(txtFrenchFriesAmount.Text * 5.9, "0.00")
PotatoWedgesSubPrice = Format(txtPotatoWedgesAmount.Text * 5.9, "0.00")
ChickenNuggetSubPrice = Format(txtChickenNuggetAmount.Text * 6.9, "0.00")
MushroomSoupSubPrice = Format(txtMushroomSoupAmount.Text * 5.9, "0.00")
PotatoSoupSubPrice = Format(txtPotatoSoupAmount.Text * 5.9, "0.00")
ChickenSoupSubPrice = Format(txtChickenSoupAmount.Text * 6.9, "0.00")
SpaghettiBologneseSubPrice = Format(txtSpaghettiBologneseAmount.Text * 9.9, "0.00")
SpaghettiCarbonaraMushroomSubPrice =
Format(txtSpaghettiCarbonaraMushroomAmount.Text * 9.9, "0.00")
SpaghettiAglioOlioSubPrice = Format(txtSpaghettiAglioOlioAmount.Text * 9.9, "0.00")
ChickenChopSubPrice = Format(txtChickenChopAmount.Text * 10.9, "0.00")
GrilledChickenMushroomSubPrice = Format(txtGrilledChickenMushroomAmount.Text *
12.3, "0.00")
GrilledChickenBlackPepperSubPrice = Format(txtGrilledChickenBlackPepperAmount.Text
* 12.3, "0.00")

TotalPrice = Format(MilkTeaSubPrice + GreenTeaSubPrice + LemonTeaSubPrice +
ChickenBurgerSubPrice + SpicyChickenBurgerSubPrice + CrispyFishBurgerSubPrice +

```

TiramisuSubPrice + CheesecakeSubPrice + ChocolateCakeSubPrice + FrenchFriesSubPrice +  
 PotatoWedgesSubPrice + ChickenNuggetSubPrice + MushroomSoupSubPrice +  
 PotatoSoupSubPrice + ChickenSoupSubPrice + SpaghettiBologneseSubPrice +  
 SpaghettiCarbonaraMushroomSubPrice + SpaghettiAglioOlioSubPrice + ChickenChopSubPrice  
 + GrilledChickenMushroomSubPrice + GrilledChickenBlackPepperSubPrice, "0.00")

```

If (txtMilkTeaAmount.Text >= 1) Then
  lstPlaceOrderCheck.Items.Add(txtMilkTeaAmount.Text & " Milk Tea: RM" &
  MilkTeaSubPrice & "0")
End If

If (txtGreenTeaAmount.Text >= 1) Then
  lstPlaceOrderCheck.Items.Add(txtGreenTeaAmount.Text & " Green Tea: RM" &
  GreenTeaSubPrice & "0")
End If

If (txtLemonTeaAmount.Text >= 1) Then
  lstPlaceOrderCheck.Items.Add(txtLemonTeaAmount.Text & " Lemon Tea: RM" &
  LemonTeaSubPrice & "0")
End If

If (txtChickenBurgerAmount.Text >= 1) Then
  lstPlaceOrderCheck.Items.Add(txtChickenBurgerAmount.Text & " Chicken Burger: RM" &
  & ChickenBurgerSubPrice & "0")
End If

If (txtSpicyChickenBurgerAmount.Text >= 1) Then
  lstPlaceOrderCheck.Items.Add(txtSpicyChickenBurgerAmount.Text & " Spicy Chicken
  Burger: RM" & SpicyChickenBurgerSubPrice & "0")
End If

```

```
If (txtCrispyFishBurgerAmount.Text >= 1) Then
    lstPlaceOrderCheck.Items.Add(txtCrispyFishBurgerAmount.Text & " Crispy Fish
Burger: RM" & CrispyFishBurgerSubPrice & "0")
End If

If (txtTiramisuAmount.Text >= 1) Then
    lstPlaceOrderCheck.Items.Add(txtTiramisuAmount.Text & " Tiramisu: RM" &
TiramisuSubPrice & "0")
End If

If (txtCheesecakeAmount.Text >= 1) Then
    lstPlaceOrderCheck.Items.Add(txtCheesecakeAmount.Text & " Cheesecake: RM" &
CheesecakeSubPrice & "0")
End If

If (txtChocolateCakeAmount.Text >= 1) Then
    lstPlaceOrderCheck.Items.Add(txtChocolateCakeAmount.Text & " Chocolate Cake:
RM" & ChocolateCakeSubPrice & "0")
End If

If (txtFrenchFriesAmount.Text >= 1) Then
    lstPlaceOrderCheck.Items.Add(txtFrenchFriesAmount.Text & " French Fries: RM" &
FrenchFriesSubPrice & "0")
End If

If (txtPotatoWedgesAmount.Text >= 1) Then
    lstPlaceOrderCheck.Items.Add(txtPotatoWedgesAmount.Text & " Potato Wedges: RM"
& PotatoWedgesSubPrice & "0")
End If

If (txtChickenNuggetAmount.Text >= 1) Then
```

```
IstPlaceOrderCheck.Items.Add(txtChickenNuggetAmount.Text & " Chicken Nuggets  
(6pcs): RM" & ChickenNuggetSubPrice & "0")
```

```
End If
```

```
If (txtMushroomSoupAmount.Text >= 1) Then  
IstPlaceOrderCheck.Items.Add(txtMushroomSoupAmount.Text & " Mushroom Soup:  
RM" & MushroomSoupSubPrice & "0")
```

```
End If
```

```
If (txtPotatoSoupAmount.Text >= 1) Then  
IstPlaceOrderCheck.Items.Add(txtPotatoSoupAmount.Text & " Potato Soup: RM" &  
PotatoSoupSubPrice & "0")
```

```
End If
```

```
If (txtChickenSoupAmount.Text >= 1) Then  
IstPlaceOrderCheck.Items.Add(txtChickenSoupAmount.Text & " Chicken Soup: RM" &  
ChickenSoupSubPrice & "0")
```

```
End If
```

```
If (txtSpaghettiBologneseAmount.Text >= 1) Then  
IstPlaceOrderCheck.Items.Add(txtSpaghettiBologneseAmount.Text & " Spaghetti  
Bolognese: RM" & SpaghettiBologneseSubPrice & "0")
```

```
End If
```

```
If (txtSpaghettiCarbonaraMushroomAmount.Text >= 1) Then  
IstPlaceOrderCheck.Items.Add(txtSpaghettiCarbonaraMushroomAmount.Text &  
" Spaghetti Carbonara with Mushroom: RM" & SpaghettiCarbonaraMushroomSubPrice & "0")  
End If
```

```
If (txtSpaghettiAglioOlioAmount.Text >= 1) Then
```

```
IstPlaceOrderCheck.Items.Add(txtSpaghettiAglioOlioAmount.Text & " Spaghetti Aglio
Olio Peperoncino: RM" & SpaghettiAglioOlioSubPrice & "0")
```

```
End If
```

```
If (txtChickenChopAmount.Text >= 1) Then
    IstPlaceOrderCheck.Items.Add(txtChickenChopAmount.Text & " Oriental Chicken
Chop: RM" & ChickenChopSubPrice & "0")
```

```
End If
```

```
If (txtGrilledChickenMushroomAmount.Text >= 1) Then
    IstPlaceOrderCheck.Items.Add(txtGrilledChickenMushroomAmount.Text & " Grilled
Chicken with Mushroom Sauce: RM" & GrilledChickenMushroomSubPrice & "0")
```

```
End If
```

```
If (txtGrilledChickenBlackPepperAmount.Text >= 1) Then
    IstPlaceOrderCheck.Items.Add(txtGrilledChickenBlackPepperAmount.Text & " Grilled
Chicken with Black Pepper Sauce: RM" & GrilledChickenBlackPepperSubPrice & "0")
```

```
End If
```

```
IstPlaceOrderCheck.Items.Add("")
```

```
IstPlaceOrderCheck.Items.Add("-----")
```

```
IstPlaceOrderCheck.Items.Add("Your total is: RM" & TotalPrice & "0")
```

```
End Sub
```

```
'Edit Order
```

```
Private Sub btnEditOrder_Click(sender As Object, e As EventArgs) Handles
btnEditOrder.Click
```

```
IstPlaceOrderCheck.Items.Clear()
```

```
End Sub
```

'Clear Order

```
Private Sub btnClearOrder_Click(sender As Object, e As EventArgs) Handles  
btnClearOrder.Click  
    lstPlaceOrderCheck.Items.Clear()  
    txtMilkTeaAmount.Text = 0  
    txtGreenTeaAmount.Text = 0  
    txtLemonTeaAmount.Text = 0  
    txtChickenBurgerAmount.Text = 0  
    txtSpicyChickenBurgerAmount.Text = 0  
    txtCrispyFishBurgerAmount.Text = 0  
    txtTiramisuAmount.Text = 0  
    txtCheesecakeAmount.Text = 0  
    txtChocolateCakeAmount.Text = 0  
    txtFrenchFriesAmount.Text = 0  
    txtPotatoWedgesAmount.Text = 0  
    txtChickenNuggetAmount.Text = 0  
    txtMushroomSoupAmount.Text = 0  
    txtPotatoSoupAmount.Text = 0  
    txtChickenSoupAmount.Text = 0  
    txtSpaghettiBologneseAmount.Text = 0  
    txtSpaghettiCarbonaraMushroomAmount.Text = 0  
    txtSpaghettiAglioOlioAmount.Text = 0  
    txtChickenChopAmount.Text = 0  
    txtGrilledChickenMushroomAmount.Text = 0  
    txtGrilledChickenBlackPepperAmount.Text = 0  
End Sub
```

'Pay

```
Private Sub btnPayment_Click(sender As Object, e As EventArgs) Handles btnPayment.Click
```

```
If lstPlaceOrderCheck.Items.Count > 0 Then
```

```
pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access  
Database\UUMCafeDatabase5.accdb"  
  
connstring = pro  
myconnection.ConnectionString = connstring  
myconnection.Open()  
  
commmand = "insert into Payment (Payment_ID) values ("" & txtPaymentID.Text & "")"  
Dim cmdpayment As OleDbCommand = New OleDbCommand(commmand,  
myconnection)  
cmdpayment.Parameters.Add(New OleDbParameter("Payment_ID",  
CType(txtPaymentID.Text, String)))  
  
cmdpayment.ExecuteNonQuery()  
  
Dim MilkTeaSubPrice As Single  
Dim GreenTeaSubPrice As Single  
Dim LemonTeaSubPrice As Single  
Dim ChickenBurgerSubPrice As Single  
Dim SpicyChickenBurgerSubPrice As Single  
Dim CrispyFishBurgerSubPrice As Single  
Dim TiramisuSubPrice As Single  
Dim CheesecakeSubPrice As Single  
Dim ChocolateCakeSubPrice As Single  
Dim FrenchFriesSubPrice As Single  
Dim PotatoWedgesSubPrice As Single  
Dim ChickenNuggetSubPrice As Single  
Dim MushroomSoupSubPrice As Single  
Dim PotatoSoupSubPrice As Single  
Dim ChickenSoupSubPrice As Single  
Dim SpaghettiBologneseSubPrice As Single
```

Dim SpaghettiCarbonaraMushroomSubPrice As Single

Dim SpaghettiAglioOlioSubPrice As Single

Dim ChickenChopSubPrice As Single

Dim GrilledChickenMushroomSubPrice As Single

Dim GrilledChickenBlackPepperSubPrice As Single

Dim TotalPrice As Single

MilkTeaSubPrice = Format(txtMilkTeaAmount.Text \* 3.9, "0.00")

GreenTeaSubPrice = Format(txtGreenTeaAmount.Text \* 3.9, "0.00")

LemonTeaSubPrice = Format(txtLemonTeaAmount.Text \* 3.9, "0.00")

ChickenBurgerSubPrice = Format(txtChickenBurgerAmount.Text \* 9.9, "0.00")

SpicyChickenBurgerSubPrice = Format(txtSpicyChickenBurgerAmount.Text \* 10.9, "0.00")

CrispyFishBurgerSubPrice = Format(txtCrispyFishBurgerAmount.Text \* 9.9, "0.00")

TiramisuSubPrice = Format(txtTiramisuAmount.Text \* 6.9, "0.00")

CheesecakeSubPrice = Format(txtCheesecakeAmount.Text \* 6.9, "0.00")

ChocolateCakeSubPrice = Format(txtChocolateCakeAmount.Text \* 5.9, "0.00")

FrenchFriesSubPrice = Format(txtFrenchFriesAmount.Text \* 5.9, "0.00")

PotatoWedgesSubPrice = Format(txtPotatoWedgesAmount.Text \* 5.9, "0.00")

ChickenNuggetSubPrice = Format(txtChickenNuggetAmount.Text \* 6.9, "0.00")

MushroomSoupSubPrice = Format(txtMushroomSoupAmount.Text \* 5.9, "0.00")

PotatoSoupSubPrice = Format(txtPotatoSoupAmount.Text \* 5.9, "0.00")

ChickenSoupSubPrice = Format(txtChickenSoupAmount.Text \* 6.9, "0.00")

SpaghettiBologneseSubPrice = Format(txtSpaghettiBologneseAmount.Text \* 9.9, "0.00")

SpaghettiCarbonaraMushroomSubPrice =

Format(txtSpaghettiCarbonaraMushroomAmount.Text \* 9.9, "0.00")

SpaghettiAglioOlioSubPrice = Format(txtSpaghettiAglioOlioAmount.Text \* 9.9, "0.00")

ChickenChopSubPrice = Format(txtChickenChopAmount.Text \* 10.9, "0.00")

GrilledChickenMushroomSubPrice = Format(txtGrilledChickenMushroomAmount.Text

\* 12.3, "0.00")

```
GrilledChickenBlackPepperSubPrice =
Format(txtGrilledChickenBlackPepperAmount.Text * 12.3, "0.00")
```

```
TotalPrice = Format(MilkTeaSubPrice + GreenTeaSubPrice + LemonTeaSubPrice +
ChickenBurgerSubPrice + SpicyChickenBurgerSubPrice + CrispyFishBurgerSubPrice +
TiramisuSubPrice + CheesecakeSubPrice + ChocolateCakeSubPrice + FrenchFriesSubPrice +
PotatoWedgesSubPrice + ChickenNuggetSubPrice + MushroomSoupSubPrice +
PotatoSoupSubPrice + ChickenSoupSubPrice + SpaghettiBologneseSubPrice +
SpaghettiCarbonaraMushroomSubPrice + SpaghettiAglioOlioSubPrice + ChickenChopSubPrice
+ GrilledChickenMushroomSubPrice + GrilledChickenBlackPepperSubPrice, "0.00")
```

```
txtMilkTeaSubPrice.Text = MilkTeaSubPrice
txtGreenTeaSubPrice.Text = GreenTeaSubPrice
txtLemonTeaSubPrice.Text = LemonTeaSubPrice
txtChickenBurgerSubPrice.Text = ChickenBurgerSubPrice
txtSpicyChickenBurgerSubPrice.Text = SpicyChickenBurgerSubPrice
txtCrispyFishBurgerSubPrice.Text = CrispyFishBurgerSubPrice
txtTiramisuSubPrice.Text = TiramisuSubPrice
txtCheesecakeSubPrice.Text = CheesecakeSubPrice
txtChocolateCakeSubPrice.Text = ChocolateCakeSubPrice
txtFrenchFriesSubPrice.Text = FrenchFriesSubPrice
txtPotatoWedgesSubPrice.Text = PotatoWedgesSubPrice
txtChickenNuggetSubPrice.Text = ChickenNuggetSubPrice
txtMushroomSoupSubPrice.Text = MushroomSoupSubPrice
txtPotatoSoupSubPrice.Text = PotatoSoupSubPrice
txtChickenSoupSubPrice.Text = ChickenSoupSubPrice
txtSpaghettiBologneseSubPrice.Text = SpaghettiBologneseSubPrice
txtSpaghettiCarbonaraMushroomSubPrice.Text =
SpaghettiCarbonaraMushroomSubPrice
txtSpaghettiAglioOlioSubPrice.Text = SpaghettiAglioOlioSubPrice
txtChickenChopSubPrice.Text = ChickenChopSubPrice
```

```

txtGrilledChickenMushroomSubPrice.Text = GrilledChickenMushroomSubPrice
txtGrilledChickenBlackPepperSubPrice.Text = GrilledChickenBlackPepperSubPrice
txtTotalPrice.Text = TotalPrice
cmdpayment.Dispose()
myconnection.Close()

'Into CustomerOrder
pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
connstring = pro
myconnection.ConnectionString = connstring
myconnection.Open()

If (txtMilkTeaAmount.Text >= 1) Then
    commmand = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
& "','" & txtCustomerName.Text & "','" & "Milk Tea" & "','" & txtMilkTeaSubPrice.Text & "','" &
txtMilkTeaAmount.Text & "','" & txtPaymentID.Text & "') "
    Dim cmdordermilktea As OleDbCommand = New OleDbCommand(commmand,
myconnection)
    cmdordermilktea.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))
    cmdordermilktea.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))
    cmdordermilktea.Parameters.Add(New OleDbParameter("Item_Menu", CType("Milk
Tea", String)))
    cmdordermilktea.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtMilkTeaSubPrice.Text, String)))
    cmdordermilktea.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtMilkTeaAmount.Text, String)))

```

```

cmdordermilktea.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))

Dim cmdorder As OleDbCommand = New OleDbCommand(command,
 myconnection)

cmdorder.ExecuteNonQuery()

End If

If (txtGreenTeaAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
 [Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
 & "','" & txtCustomerName.Text & "','" & "Green Tea" & "','" & txtGreenTeaSubPrice.Text &
 "','" & txtGreenTeaAmount.Text & "','" & txtPaymentID.Text & "')"

    Dim cmdordergreentea As OleDbCommand = New OleDbCommand(command,
 myconnection)

    cmdordergreentea.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))

    cmdordergreentea.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))

    cmdordergreentea.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Green Tea", String)))

    cmdordergreentea.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtGreenTeaSubPrice.Text, String)))

    cmdordergreentea.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtGreenTeaAmount.Text, String)))

    cmdordergreentea.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
 myconnection)

    cmdorder.ExecuteNonQuery()

End If

```

```

If (txtLemonTeaAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Lemon Tea" & "" & txtLemonTeaSubPrice.Text &
"" & txtLemonTeaAmount.Text & "" & txtPaymentID.Text & ")"
    Dim cmdorderlemontea As OleDbCommand = New OleDbCommand(command,
myconnection)
    cmdorderlemontea.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))
    cmdorderlemontea.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))
    cmdorderlemontea.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Lemon Tea", String)))
    cmdorderlemontea.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtLemonTeaSubPrice.Text, String)))
    cmdorderlemontea.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtLemonTeaAmount.Text, String)))
    cmdorderlemontea.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))
    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
    cmdorder.ExecuteNonQuery()
End If

If (txtChickenBurgerAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Chicken Burger" & "" &
txtChickenBurgerSubPrice.Text & "" & txtChickenBurgerAmount.Text & "" &
txtPaymentID.Text & ")"

```

```

        Dim cmdorderchickenburger As OleDbCommand = New
        OleDbCommand(command, myconnection)
            cmdorderchickenburger.Parameters.Add(New OleDbParameter("Customer_ID",
        CType(txtCustomerID.Text, String)))
            cmdorderchickenburger.Parameters.Add(New OleDbParameter("Customer_Name",
        CType(txtCustomerName.Text, String)))
            cmdorderchickenburger.Parameters.Add(New OleDbParameter("Item_Menu",
        CType("Chicken Burger", String)))
            cmdorderchickenburger.Parameters.Add(New OleDbParameter("Item_Price",
        CType(txtChickenBurgerSubPrice.Text, String)))
            cmdorderchickenburger.Parameters.Add(New OleDbParameter("Item_Quantity",
        CType(txtChickenBurgerAmount.Text, String)))
            cmdorderchickenburger.Parameters.Add(New OleDbParameter("Payment_ID",
        CType(txtPaymentID.Text, String)))
        Dim cmdorder As OleDbCommand = New OleDbCommand(command,
        myconnection)
        cmdorder.ExecuteNonQuery()
    End If

    If (txtSpicyChickenBurgerAmount.Text >= 1) Then
        command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
        [Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
        & ""","" & txtCustomerName.Text & "","" & "Spicy Chicken Burger" & "","" &
        txtSpicyChickenBurgerSubPrice.Text & "","" & txtSpicyChickenBurgerAmount.Text & "","" &
        txtPaymentID.Text & ") "
        Dim cmdorderspicychickenburger As OleDbCommand = New
        OleDbCommand(command, myconnection)
        cmdorderspicychickenburger.Parameters.Add(New OleDbParameter("Customer_ID",
        CType(txtCustomerID.Text, String)))
        cmdorderspicychickenburger.Parameters.Add(New
        OleDbParameter("Customer_Name", CType(txtCustomerName.Text, String)))

```

```

    cmdorderspicychickenburger.Parameters.Add(New OleDbParameter("Item_Menu",
    CType("Spicy Chicken Burger", String)))

    cmdorderspicychickenburger.Parameters.Add(New OleDbParameter("Item_Price",
    CType(txtSpicyChickenBurgerSubPrice.Text, String)))

    cmdorderspicychickenburger.Parameters.Add(New OleDbParameter("Item_Quantity",
    CType(txtSpicyChickenBurgerAmount.Text, String)))

    cmdorderspicychickenburger.Parameters.Add(New OleDbParameter("Payment_ID",
    CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
    myconnection)

    cmdorder.ExecuteNonQuery()

End If

If (txtCrispyFishBurgerAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
    [Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
    & "','" & txtCustomerName.Text & "','" & "Crispy FIsh Burger" & "','" &
    txtCrispyFishBurgerSubPrice.Text & "','" & txtCrispyFishBurgerAmount.Text & "','" &
    txtPaymentID.Text & "')"

    Dim cmdordercrispyfishburger As OleDbCommand = New
    OleDbCommand(command, myconnection)

    cmdordercrispyfishburger.Parameters.Add(New OleDbParameter("Customer_ID",
    CType(txtCustomerID.Text, String)))

    cmdordercrispyfishburger.Parameters.Add(New OleDbParameter("Customer_Name",
    CType(txtCustomerName.Text, String)))

    cmdordercrispyfishburger.Parameters.Add(New OleDbParameter("Item_Menu",
    CType("Crispy Fish Burger", String)))

    cmdordercrispyfishburger.Parameters.Add(New OleDbParameter("Item_Price",
    CType(txtCrispyFishBurgerSubPrice.Text, String)))

    cmdordercrispyfishburger.Parameters.Add(New OleDbParameter("Item_Quantity",
    CType(txtCrispyFishBurgerAmount.Text, String)))

```

```

cmdordercrispyfishburger.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
 myconnection)

        cmdorder.ExecuteNonQuery()

    End If

If (txtTiramisuAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
 [Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
 & "','" & txtCustomerName.Text & "','" & "Tiramisu" & "','" & txtTiramisuSubPrice.Text & "','" 
& txtTiramisuAmount.Text & "','" & txtPaymentID.Text & "')"

    Dim cmdordertiramisu As OleDbCommand = New OleDbCommand(command,
 myconnection)

        cmdordertiramisu.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))

        cmdordertiramisu.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))

        cmdordertiramisu.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Tiramisu", String)))

        cmdordertiramisu.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtTiramisuSubPrice.Text, String)))

        cmdordertiramisu.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtTiramisuAmount.Text, String)))

        cmdordertiramisu.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
 myconnection)

        cmdorder.ExecuteNonQuery()

    End If

```

```

If (txtCheesecakeAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Cheesecake" & "" & txtCheesecakeSubPrice.Text
& "" & txtCheesecakeAmount.Text & "" & txtPaymentID.Text & ") "
    Dim cmdordercheesecake As OleDbCommand = New OleDbCommand(command,
myconnection)
    cmdordercheesecake.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))
    cmdordercheesecake.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))
    cmdordercheesecake.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Cheesecake", String)))
    cmdordercheesecake.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtCheesecakeSubPrice.Text, String)))
    cmdordercheesecake.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtCheesecakeAmount.Text, String)))
    cmdordercheesecake.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))
    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
    cmdorder.ExecuteNonQuery()
End If

If (txtChocolateCakeAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Chocolate Cake" & "" &
txtChocolateCakeSubPrice.Text & "" & txtChocolateCakeAmount.Text & "" &
txtPaymentID.Text & ") "

```

```

    Dim cmdorderchocolatecake As OleDbCommand = New
    OleDbCommand(command, myconnection)
        cmdorderchocolatecake.Parameters.Add(New OleDbParameter("Customer_ID",
        CType(txtCustomerID.Text, String)))
        cmdorderchocolatecake.Parameters.Add(New OleDbParameter("Customer_Name",
        CType(txtCustomerName.Text, String)))
        cmdorderchocolatecake.Parameters.Add(New OleDbParameter("Item_Menu",
        CType("Chocolate Cake", String)))
        cmdorderchocolatecake.Parameters.Add(New OleDbParameter("Item_Price",
        CType(txtChocolateCakeSubPrice.Text, String)))
        cmdorderchocolatecake.Parameters.Add(New OleDbParameter("Item_Quantity",
        CType(txtChocolateCakeAmount.Text, String)))
        cmdorderchocolatecake.Parameters.Add(New OleDbParameter("Payment_ID",
        CType(txtPaymentID.Text, String)))
    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
    myconnection)
        cmdorder.ExecuteNonQuery()
    End If

    If (txtFrenchFriesAmount.Text >= 1) Then
        command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
        [Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
        & ""","" & txtCustomerName.Text & ""","" & "French Fries" & ""","" & txtFrenchFriesSubPrice.Text
        & ""","" & txtFrenchFriesAmount.Text & ""","" & txtPaymentID.Text & ")"
        Dim cmdorderfrenchfries As OleDbCommand = New OleDbCommand(command,
        myconnection)
        cmdorderfrenchfries.Parameters.Add(New OleDbParameter("Customer_ID",
        CType(txtCustomerID.Text, String)))
        cmdorderfrenchfries.Parameters.Add(New OleDbParameter("Customer_Name",
        CType(txtCustomerName.Text, String)))

```

```

    cmdorderfrenchfries.Parameters.Add(New OleDbParameter("Item_Menu",
    CType("French Fries", String)))
        cmdorderfrenchfries.Parameters.Add(New OleDbParameter("Item_Price",
    CType(txtFrenchFriesSubPrice.Text, String)))
            cmdorderfrenchfries.Parameters.Add(New OleDbParameter("Item_Quantity",
    CType(txtFrenchFriesAmount.Text, String)))
                cmdorderfrenchfries.Parameters.Add(New OleDbParameter("Payment_ID",
    CType(txtPaymentID.Text, String)))

        Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
            cmdorder.ExecuteNonQuery()
        End If

If (txtPotatoWedgesAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
& "','" & txtCustomerName.Text & "','" & "Potawo Wedges" & "','" &
txtPotatoWedgesSubPrice.Text & "','" & txtPotatoWedgesAmount.Text & "','" &
txtPaymentID.Text & "')"

    Dim cmdorderpotatowedges As OleDbCommand = New
OleDbCommand(command, myconnection)
        cmdorderpotatowedges.Parameters.Add(New OleDbParameter("Customer_ID",
CType(txtCustomerID.Text, String)))
            cmdorderpotatowedges.Parameters.Add(New OleDbParameter("Customer_Name",
CType(txtCustomerName.Text, String)))
                cmdorderpotatowedges.Parameters.Add(New OleDbParameter("Item_Menu",
CType("Potato Wedges", String)))
                    cmdorderpotatowedges.Parameters.Add(New OleDbParameter("Item_Price",
CType(txtPotatoWedgesSubPrice.Text, String)))
                        cmdorderpotatowedges.Parameters.Add(New OleDbParameter("Item_Quantity",
CType(txtPotatoWedgesAmount.Text, String)))

```

```

cmdorder.Parameters.Add(New OleDbParameter("Payment_ID",
CType(txtPaymentID.Text, String)))

Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)

cmdorder.ExecuteNonQuery()

End If

If (txtChickenNuggetAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
& "','" & txtCustomerName.Text & "','" & "Chicken Nuggets (6 pcs)" & "','" &
txtChickenNuggetSubPrice.Text & "','" & txtChickenNuggetAmount.Text & "','" &
txtPaymentID.Text & "')"

    Dim cmdorderchickennugget As OleDbCommand = New
OleDbCommand(command, myconnection)

    cmdorderchickennugget.Parameters.Add(New OleDbParameter("Customer_ID",
CType(txtCustomerID.Text, String)))

    cmdorderchickennugget.Parameters.Add(New OleDbParameter("Customer_Name",
CType(txtCustomerName.Text, String)))

    cmdorderchickennugget.Parameters.Add(New OleDbParameter("Item_Menu",
CType("Chicken Nuggets (6pcs)", String)))

    cmdorderchickennugget.Parameters.Add(New OleDbParameter("Item_Price",
CType(txtChickenNuggetSubPrice.Text, String)))

    cmdorderchickennugget.Parameters.Add(New OleDbParameter("Item_Quantity",
CType(txtChickenNuggetAmount.Text, String)))

    cmdorderchickennugget.Parameters.Add(New OleDbParameter("Payment_ID",
CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)

    cmdorder.ExecuteNonQuery()

End If

```

```

If (txtMushroomSoupAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Mushroom Soup" & "" &
txtMushroomSoupSubPrice.Text & "" & txtMushroomSoupAmount.Text & "" &
txtPaymentID.Text & ")"
    Dim cmdordermushroomsoup As OleDbCommand = New
OleDbCommand(command, myconnection)
    cmdordermushroomsoup.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))
    cmdordermushroomsoup.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))
    cmdordermushroomsoup.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Mushroom Soup", String)))
    cmdordermushroomsoup.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtMushroomSoupSubPrice.Text, String)))
    cmdordermushroomsoup.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtMushroomSoupAmount.Text, String)))
    cmdordermushroomsoup.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))
    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
    cmdorder.ExecuteNonQuery()
End If

If (txtPotatoSoupAmount.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Potato Soup" & "" &
txtPotatoSoupSubPrice.Text & "" & txtPotatoSoupAmount.Text & "" &
txtPaymentID.Text & ")"

```

```

    Dim cmdorderpotatosoup As OleDbCommand = New OleDbCommand(command,
myconnection)

        cmdorderpotatosoup.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))

        cmdorderpotatosoup.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))

        cmdorderpotatosoup.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Potato Soup", String)))

        cmdorderpotatosoup.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtPotatoSoupSubPrice.Text, String)))

        cmdorderpotatosoup.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtPotatoSoupAmount.Text, String)))

        cmdorderpotatosoup.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)

    cmdorder.ExecuteNonQuery()

End If

If (txtChickenSoupAmount.Text >= 1) Then

    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& ""","" & txtCustomerName.Text & ""","" & "Chicken Soup" & ""","" &
txtChickenSoupSubPrice.Text & ""","" & txtChickenSoupAmount.Text & ""","" &
txtPaymentID.Text & "") "

    Dim cmdorderchickensoup As OleDbCommand = New OleDbCommand(command,
myconnection)

    cmdorderchickensoup.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))

    cmdorderchickensoup.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))

```

```

    cmdorderchickensoup.Parameters.Add(New OleDbParameter("Item_Menu",
    CType("Chicken Soup", String)))
        cmdorderchickensoup.Parameters.Add(New OleDbParameter("Item_Price",
    CType(txtChickenSoupSubPrice.Text, String)))
            cmdorderchickensoup.Parameters.Add(New OleDbParameter("Item_Quantity",
    CType(txtChickenSoupAmount.Text, String)))
                cmdorderchickensoup.Parameters.Add(New OleDbParameter("Payment_ID",
    CType(txtPaymentID.Text, String)))

        Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
            cmdorder.ExecuteNonQuery()
        End If

If (txtSpaghettiBologneseSubPrice.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
& "','" & txtCustomerName.Text & "','" & "Spaghetti Bolognese" & "','" &
txtSpaghettiBologneseSubPrice.Text & "','" & txtSpaghettiBologneseAmount.Text & "','" &
txtPaymentID.Text & "')"

    Dim cmdspaghettibolognese As OleDbCommand = New
OleDbCommand(command, myconnection)
        cmdspaghettibolognese.Parameters.Add(New OleDbParameter("Customer_ID",
CType(txtCustomerID.Text, String)))
            cmdspaghettibolognese.Parameters.Add(New OleDbParameter("Customer_Name",
CType(txtCustomerName.Text, String)))
                cmdspaghettibolognese.Parameters.Add(New OleDbParameter("Item_Menu",
CType("Spaghetti Bolognese", String)))
                    cmdspaghettibolognese.Parameters.Add(New OleDbParameter("Item_Price",
CType(txtSpaghettiBologneseSubPrice.Text, String)))
                        cmdspaghettibolognese.Parameters.Add(New OleDbParameter("Item_Quantity",
CType(txtSpaghettiBologneseAmount.Text, String)))

```

```

cmdspaghettibolognese.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
 myconnection)

        cmdorder.ExecuteNonQuery()

    End If

If (txtSpaghettiCarbonaraMushroomSubPrice.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
 [Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ('" & txtCustomerID.Text
 & "','" & txtCustomerName.Text & "','" & "Spaghetti Carbonara Mushroom" & "','" &
 txtSpaghettiCarbonaraMushroomSubPrice.Text & "','" &
 txtSpaghettiCarbonaraMushroomAmount.Text & "','" & txtPaymentID.Text & "')"

    Dim cmdspaghetticarbonaramushroom As OleDbCommand = New
 OleDbCommand(command, myconnection)

        cmdspaghetticarbonaramushroom.Parameters.Add(New
 OleDbParameter("Customer_ID", CType(txtCustomerID.Text, String)))

        cmdspaghetticarbonaramushroom.Parameters.Add(New
 OleDbParameter("Customer_Name", CType(txtCustomerName.Text, String)))

        cmdspaghetticarbonaramushroom.Parameters.Add(New
 OleDbParameter("Item_Menu", CType("Spaghetti Carbonara with Mushroom", String)))

        cmdspaghetticarbonaramushroom.Parameters.Add(New
 OleDbParameter("Item_Price", CType(txtSpaghettiCarbonaraMushroomSubPrice.Text, String)))

        cmdspaghetticarbonaramushroom.Parameters.Add(New
 OleDbParameter("Item_Quantity", CType(txtSpaghettiCarbonaraMushroomAmount.Text,
 String)))

        cmdspaghetticarbonaramushroom.Parameters.Add(New
 OleDbParameter("Payment_ID", CType(txtPaymentID.Text, String)))

    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
 myconnection)

        cmdorder.ExecuteNonQuery()

```

End If

If (txtSpaghettiAglioOlioSubPrice.Text >= 1) Then

```
    commmand = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Spaghetti Aglio Olio Peperoncino" & "" &
txtSpaghettiAglioOlioSubPrice.Text & "" & txtSpaghettiAglioOlioAmount.Text & "" &
txtPaymentID.Text & ") "
```

```
    Dim cmdspaghettiaglioolio As OleDbCommand = New OleDbCommand(commmand,
myconnection)
```

```
    cmdspaghettiaglioolio.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustomerID.Text, String)))
```

```
    cmdspaghettiaglioolio.Parameters.Add(New OleDbParameter("Customer_Name",
 CType(txtCustomerName.Text, String)))
```

```
    cmdspaghettiaglioolio.Parameters.Add(New OleDbParameter("Item_Menu",
 CType("Spaghetti Aglio Olio Peperoncino", String)))
```

```
    cmdspaghettiaglioolio.Parameters.Add(New OleDbParameter("Item_Price",
 CType(txtSpaghettiAglioOlioSubPrice.Text, String)))
```

```
    cmdspaghettiaglioolio.Parameters.Add(New OleDbParameter("Item_Quantity",
 CType(txtSpaghettiAglioOlioAmount.Text, String)))
```

```
    cmdspaghettiaglioolio.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))
```

```
    Dim cmdorder As OleDbCommand = New OleDbCommand(commmand,
myconnection)
```

```
    cmdorder.ExecuteNonQuery()
```

End If

If (txtChickenChopSubPrice.Text >= 1) Then

```
    commmand = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Oriental Chicken Chop" & "" &
```

```

txtChickenChopSubPrice.Text & "," & txtChickenChopAmount.Text & "," &
txtPaymentID.Text & ") "
    Dim cmdchickenchop As OleDbCommand = New OleDbCommand(command,
myconnection)
        cmdchickenchop.Parameters.Add(New OleDbParameter("Customer_ID",
CType(txtCustomerID.Text, String)))
        cmdchickenchop.Parameters.Add(New OleDbParameter("Customer_Name",
CType(txtCustomerName.Text, String)))
        cmdchickenchop.Parameters.Add(New OleDbParameter("Item_Menu",
CType("Oriental Chicken Chop", String)))
        cmdchickenchop.Parameters.Add(New OleDbParameter("Item_Price",
CType(txtChickenChopSubPrice.Text, String)))
        cmdchickenchop.Parameters.Add(New OleDbParameter("Item_Quantity",
CType(txtChickenChopAmount.Text, String)))
        cmdchickenchop.Parameters.Add(New OleDbParameter("Payment_ID",
CType(txtPaymentID.Text, String)))
    Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
        cmdorder.ExecuteNonQuery()
End If

If (txtGrilledChickenMushroomSubPrice.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values (" & txtCustomerID.Text
& "," & txtCustomerName.Text & "," & "Grilled Chicken with Mushroom Sauce" & "," &
txtGrilledChickenMushroomSubPrice.Text & "," & txtGrilledChickenMushroomAmount.Text
& "," & txtPaymentID.Text & ")"
    Dim cmdgrilledchickenmushroom As OleDbCommand = New
OleDbCommand(command, myconnection)
    cmdgrilledchickenmushroom.Parameters.Add(New OleDbParameter("Customer_ID",
CType(txtCustomerID.Text, String)))

```

```

cmdgrilledchickenmushroom.Parameters.Add(New
OleDbParameter("Customer_Name", CType(txtCustomerName.Text, String)))
    cmdgrilledchickenmushroom.Parameters.Add(New OleDbParameter("Item_Menu",
CType("Grilled Chicken with Mushroom Sauce", String)))
        cmdgrilledchickenmushroom.Parameters.Add(New OleDbParameter("Item_Price",
CType(txtGrilledChickenMushroomSubPrice.Text, String)))
            cmdgrilledchickenmushroom.Parameters.Add(New OleDbParameter("Item_Quantity",
CType(txtGrilledChickenMushroomAmount.Text, String)))
                cmdgrilledchickenmushroom.Parameters.Add(New OleDbParameter("Payment_ID",
CType(txtPaymentID.Text, String)))

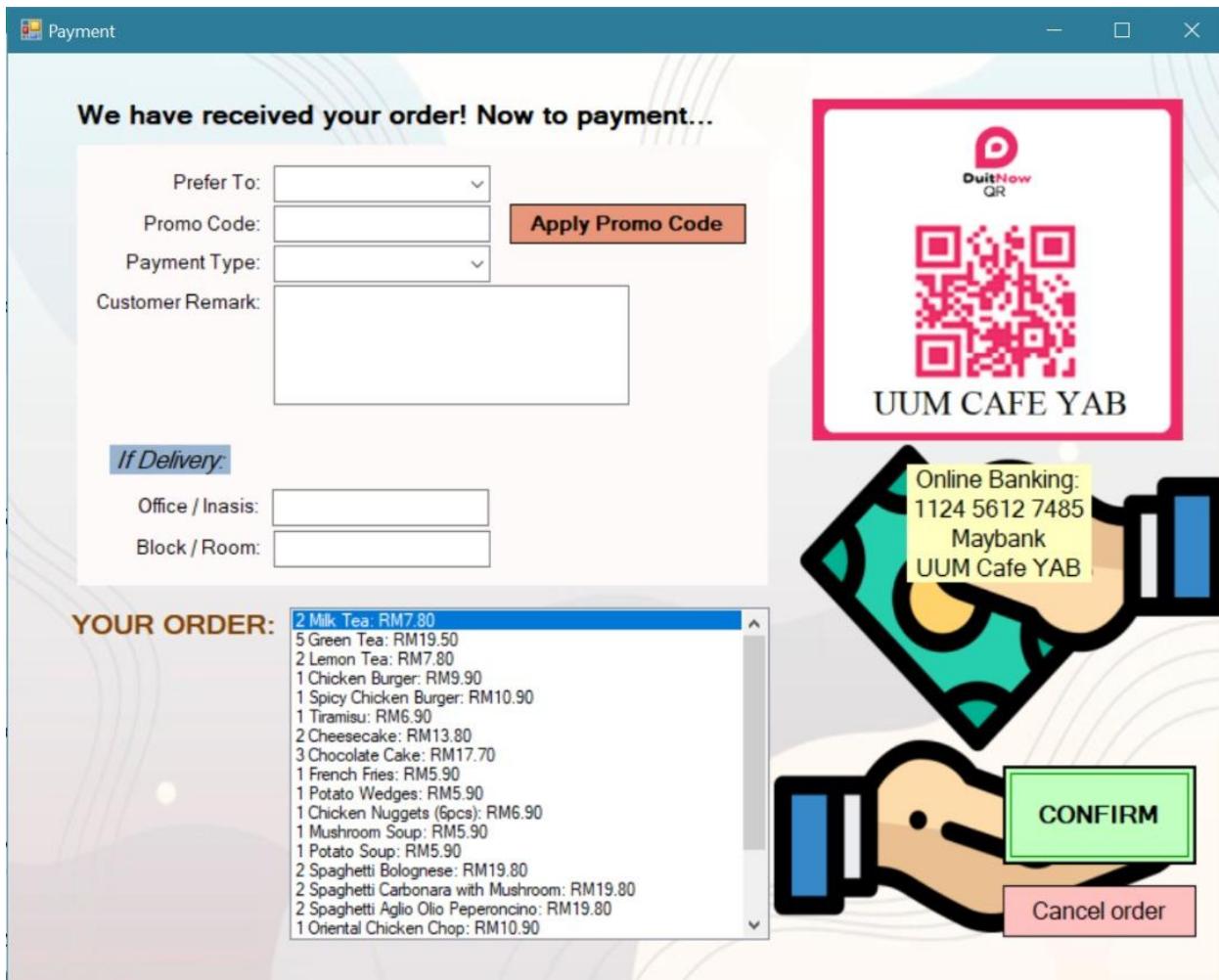
        Dim cmdorder As OleDbCommand = New OleDbCommand(command,
myconnection)
            cmdorder.ExecuteNonQuery()
        End If

If (txtGrilledChickenBlackPepperSubPrice.Text >= 1) Then
    command = "insert into CustomerOrder ([Customer_ID] , [Customer_Name] ,
[Item_Menu] , [Item_Price] , [Item_Quantity] , [Payment_ID]) values ("" & txtCustomerID.Text
& "" & txtCustomerName.Text & "" & "Grilled Chicken with Black Pepper Sauce" & "" &
txtGrilledChickenBlackPepperSubPrice.Text & "" &
txtGrilledChickenBlackPepperAmount.Text & "" & txtPaymentID.Text & ")"
    Dim cmdgrilledchickenblackpepper As OleDbCommand = New
OleDbCommand(command, myconnection)
        cmdgrilledchickenblackpepper.Parameters.Add(New OleDbParameter("Customer_ID",
CType(txtCustomerID.Text, String)))
            cmdgrilledchickenblackpepper.Parameters.Add(New
OleDbParameter("Customer_Name", CType(txtCustomerName.Text, String)))
                cmdgrilledchickenblackpepper.Parameters.Add(New OleDbParameter("Item_Menu",
CType("Grilled Chicken with Black Pepper Sauce", String)))
                    cmdgrilledchickenblackpepper.Parameters.Add(New OleDbParameter("Item_Price",
CType(txtGrilledChickenBlackPepperSubPrice.Text, String)))

```

```
cmdgrilledchickenblackpepper.Parameters.Add(New  
OleDbParameter("Item_Quantity", CType(txtGrilledChickenBlackPepperAmount.Text, String)))  
cmdgrilledchickenblackpepper.Parameters.Add(New OleDbParameter("Payment_ID",  
CType(txtPaymentID.Text, String)))  
Dim cmdorder As OleDbCommand = New OleDbCommand(command,  
myconnection)  
cmdorder.ExecuteNonQuery()  
End If  
  
myconnection.Close()  
Me.Hide()  
frmPayment.Show()  
  
Else  
    MsgBox("You need to place order to view your ordered food.",  
    MessageBoxIcon.Information, "UUM Cafe YAB")  
End If  
  
End Sub  
  
Private Sub cmdBack_Click(sender As Object, e As EventArgs) Handles cmdBack.Click  
    Me.Hide()  
    frmCustMainMenu.Show()  
End Sub  
End Class
```

## Form 6: frmPayment

Coding

```
Imports System.Data.OleDb
```

```
Public Class frmPayment
```

```
    Dim pro As String
```

```
    Dim connstring As String
```

```
    Dim command As String
```

```
    Dim myconnection As OleDbConnection = New OleDbConnection
```

```
Private Sub frmPayment_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    lstPlaceOrderCheck.DataSource = (frmOrder.lstPlaceOrderCheck.Items)
```

```
txtCustomerID.Text = frmOrder.txtCustomerID.Text  
txtPaymentID.Text = frmOrder.txtPaymentID.Text  
txtCuHp.Text = frmLogin.txtCuHP.Text  
txtSubTotal.Text = Val(frmOrder.txtTotalPrice.Text)  
dateOrderDate.Value = DateTime.Now  
dateOrderTime.Value = DateTime.Now  
End Sub
```

```
Private Sub cmdPromoCodeActivation_Click(sender As Object, e As EventArgs) Handles  
cmdPromoCodeActivation.Click
```

```
If txtOrderType.Text = "Delivery" Then  
    txtDeliveryFee.Text = Val(1.0)  
Else  
    txtDeliveryFee.Text = Val(0.00)  
End If
```

```
Dim SubTotalPrice As Single  
SubTotalPrice = Val(txtSubTotal.Text)
```

```
Dim DeliveryFee As Single  
DeliveryFee = Val(txtDeliveryFee.Text)
```

```
Dim TotalPrice As Single  
TotalPrice = Val(txtTotalPrice.Text)
```

```
TotalPrice = Format(SubTotalPrice + DeliveryFee, "0.00")
```

```
If txtPromoCode.Text = "NEWREG" Then  
    TotalPrice = TotalPrice * 0.5  
End If
```

```
If txtPromoCode.Text = "NEWYEAR" Then
```

```
    TotalPrice = TotalPrice * 0.65
```

```
End If
```

```
If txtPromoCode.Text = "EATINGDEALS" Then
```

```
    TotalPrice = TotalPrice * 0.8
```

```
End If
```

```
If txtPromoCode.Text = "HAPPYMEAL" Then
```

```
    TotalPrice = TotalPrice * 0.85
```

```
End If
```

```
If txtPromoCode.Text = "HAPPYMEAL" Then
```

```
    TotalPrice = TotalPrice * 0.85
```

```
End If
```

```
If txtPromoCode.Text = "PICKUP" Then
```

```
    TotalPrice = TotalPrice * 0.9
```

```
End If
```

```
txtFinalPrice1.Text = ("Promo Code: " & txtPromoCode.Text)
```

```
txtFinalPrice2.Text = ("Your final price is: RM" & TotalPrice)
```

```
txtTotalPrice.Text = TotalPrice
```

```
End Sub
```

```
Private Sub cmdConfirmPayment_Click(sender As Object, e As EventArgs) Handles  
cmdConfirmPayment.Click
```

```
If txtOrderType.SelectedIndex = -1 Then
```

```
    MsgBox("Are you prefer to dine in, take away, or delivering?",
```

```
    MessageBoxIcon.Question, "UUM Cafe YAB")
```

```
ElseIf txtPaymentType.SelectedIndex = -1 Then
    MsgBox("Please choose your payment type.", MessageBoxButtons.Question, "UUM Cafe
YAB")

Else
    If txtPromoCode.Text = String.Empty Then
        txtPromoCode.Text = "N/A"
    Else
        txtPromoCode.Text = txtPromoCode.Text
    End If

    If txtCuRemark.Text = String.Empty Then
        txtCuRemark.Text = "N/A"
    Else
        txtCuRemark.Text = txtCuRemark.Text
    End If

    If txtOfficeInasis.Text = String.Empty Then
        txtOfficeInasis.Text = "N/A"
    Else
        txtOfficeInasis.Text = txtOfficeInasis.Text
    End If

    If txtBlockRoom.Text = String.Empty Then
        txtBlockRoom.Text = "N/A"
    Else
        txtBlockRoom.Text = txtBlockRoom.Text
    End If
```

```
txtOrderStatus.Text = "Kitchen preparing"
```

```
Dim SubTotalPrice As Single
SubTotalPrice = Val(txtSubTotal.Text)
```

```
Dim DeliveryFee As Single
DeliveryFee = Val(txtDeliveryFee.Text)
```

```
If txtDeliveryFee.Text = String.Empty Then
    txtDeliveryFee.Text = 0
End If
```

```
Dim TotalPrice As Single
TotalPrice = Val(txtTotalPrice.Text)
```

```
TotalPrice = Format(SubTotalPrice + DeliveryFee, "0.00")
txtTotalPrice.Text = TotalPrice
```

```
pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
connstring = pro
myconnection.ConnectionString = connstring
myconnection.Open()
```

```
command = "UPDATE Payment SET [Customer_ID] = '" & txtCustomerID.Text & "' ,
[Cu_HP] = '" & txtCuHp.Text & "' , [Promo_Code] = '" & txtPromoCode.Text & "' ,
[Order_Type] = '" & txtOrderType.Text & "' , [Payment_SubTotal] = '" & Val(txtSubTotal.Text)
& "' , [Delivery_Fee] = '" & Val(txtDeliveryFee.Text) & "' , [Payment_Total] = '" &
Val(txtTotalPrice.Text) & "' , [Payment_Type] = '" & txtPaymentType.Text & "' ,
[Payment_Date] = '" & dateOrderDate.Value & "' , [Payment_Time] = '" & dateOrderTime.Value
```

```
& "" , [Cu_Remark] = "" & txtCuRemark.Text & "" , [Cu_Office_Inasis] = "" &
txtOfficeInasis.Text & "" , [Cu_Block_Room] = "" & txtBlockRoom.Text & "" , [Order_Status] =
"" & txtOrderType.Text & "" where [Payment_ID] = "" & txtPaymentID.Text & """
```

```
Dim cmd As OleDbCommand = New OleDbCommand(command, myconnection)
```

```
cmd.ExecuteNonQuery()
```

```
cmd.Dispose()
```

```
myconnection.Close()
```

```
txtCustomerID.Text = ""
```

```
txtPromoCode.Clear()
```

```
txtOrderType.Text = String.Empty
```

```
txtSubTotal.Clear()
```

```
txtDeliveryFee.Clear()
```

```
txtTotalPrice.Clear()
```

```
txtDeliveryFee.Clear()
```

```
txtPaymentType.Text = String.Empty
```

```
dateOrderDate.Value = DateTime.Now
```

```
dateOrderTime.Value = DateTime.Now
```

```
txtCuRemark.Clear()
```

```
MsgBox("Thank you! Your payment code is " & txtPaymentID.Text & ". Take note as
you reference later.", MessageBoxButtons.Information, "UUM Cafe YAB")
```

```
Me.Hide()
```

```
frmCustMainMenu.Show()
```

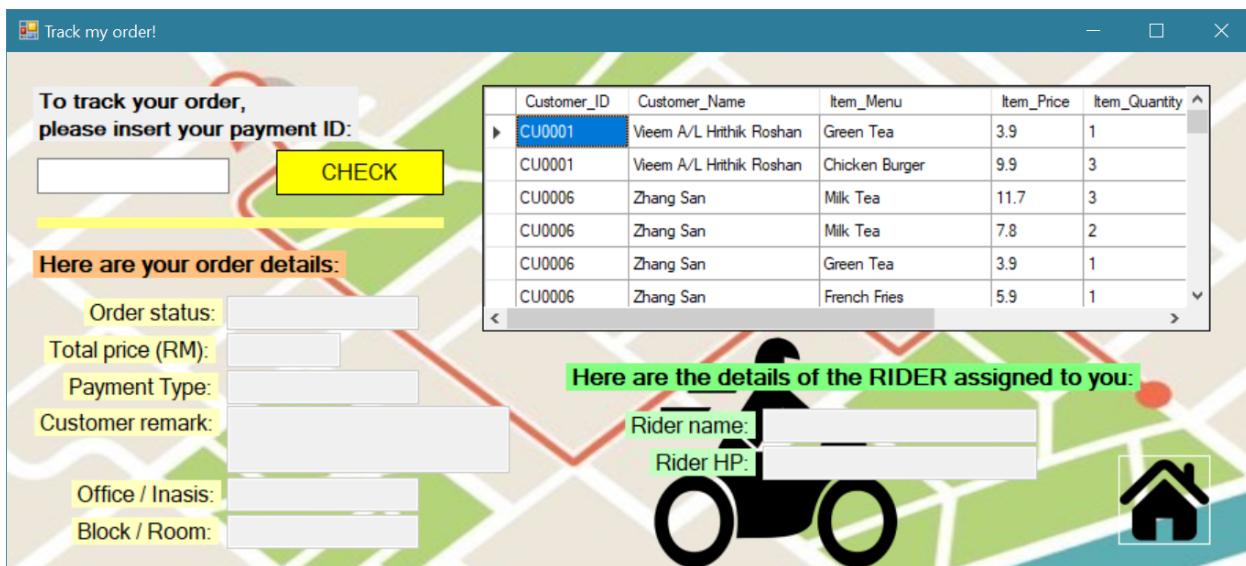
```
End If
```

```
End Sub
```

```
'Cancel order

Private Sub cmdBack_Click(sender As Object, e As EventArgs) Handles cmdBack.Click
    Dim confirm_cancel As Integer =
        MessageBox.Show("Are you sure you want to cancel this order?", "UUM Cafe YAB",
        MessageBoxButtons.YesNo)
    If confirm_cancel = DialogResult.Yes Then
        pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
        connstring = pro
        myconnection.ConnectionString = connstring
        myconnection.Open()
        command = "DELETE FROM Payment where [Payment_ID] = '" &
        txtPaymentID.Text & "'"
        Dim cmd As OleDbCommand = New OleDbCommand(command, myconnection)
        cmd.ExecuteNonQuery()
        cmd.Dispose()
        myconnection.Close()
        MsgBox("Your order: " & txtPaymentID.Text & " has been deleted.",
        MessageBoxIcon.Information, "UUM Cafe YAB")
        Me.Hide()
        frmCustMainMenu.Show()
    End If
End Sub

End Class
```

**Form 7: frmTracking**Coding

```
Imports System.Data.OleDb
```

```
Public Class frmTracking
```

```
    Dim pro As String
```

```
    Dim connstring As String
```

```
    Dim myconnection As OleDbConnection = New OleDbConnection
```

```
    Dim con As New OleDbConnection
```

```
Private Sub CustomerOrderBindingNavigatorSaveItem_Click(sender As Object, e As EventArgs) Handles CustomerOrderBindingNavigatorSaveItem.Click
```

```
    Me.Validate()
```

```
    Me.CustomerOrderBindingSource.EndEdit()
```

```
    Me.TableAdapterManager.UpdateAll(Me.UUMCafeDatabase5DataSet)
```

```
End Sub
```

```
Private Sub frmTracking_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

'TODO: This line of code loads data into the 'UUMCafeDatabase5DataSet.CustomerOrder' table. You can move, or remove it, as needed.

```
Me.CustomerOrderTableAdapter.Fill(Me.UUMCafeDatabase5DataSet.CustomerOrder)
```

End Sub

```
Private Sub cmdCheckTracking_Click(sender As Object, e As EventArgs) Handles  
cmdCheckTracking.Click
```

```
If txtPaymentID.Text = String.Empty Then  
    MsgBox("Please fill in your payment ID.", MessageBoxIcon.Information, "UUM Cafe  
YAB")
```

Else

```
txtOrderStatus.Text = ""  
txtTotalPrice.Text = ""  
txtPaymentType.Text = ""  
txtCuRemark.Text = ""  
txtOfficeInasis.Text = ""  
txtBlockRoom.Text = ""  
txtRiderID.Text = ""  
txtRiderName.Text = ""  
txtRiderHP.Text = ""
```

```
CustomerOrderBindingSource.Filter = "Payment_ID LIKE '" & txtPaymentID.Text & "'"  
con.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access  
Database\UUMCafeDatabase5.accdb"  
con.Open()
```

```
Dim strSQLnorder As String
```

```

strsqlnorider = "select [Order_Status] , [Payment_Total] , [Payment_Type] ,
[Cu_Remark] , [Cu_Office_Inasis] , [Cu_Block_Room] , [Rider_ID] from Payment where
[Payment_ID] = "" & txtPaymentID.Text & """

```

```

Dim cmdnorider As New OleDbCommand(strsqlnorider, con)
Dim myreadernorider As OleDbDataReader
myreadernorider = cmdnorider.ExecuteReader
myreadernorider.Read()

```

```

txtOrderStatus.Text = myreadernorider("Order_Status")
txtTotalPrice.Text = Format(myreadernorider("Payment_Total"), "0.00")
txtPaymentType.Text = myreadernorider("Payment_Type")
txtCuRemark.Text = myreadernorider("Cu_Remark")
txtOfficeInasis.Text = myreadernorider("Cu_Office_Inasis")
txtBlockRoom.Text = myreadernorider("Cu_Block_Room")

```

```

Dim strsqlorderrider As String
strsqlorderrider = "select [Rider_ID] from Payment where [Payment_ID] = "" &
txtPaymentID.Text & """
Dim cmdorderrider As New OleDbCommand(strsqlorderrider, con)
Dim myreaderorderrider As OleDbDataReader
myreaderorderrider = cmdorderrider.ExecuteReader
myreaderorderrider.Read()

```

```
txtRiderID.Text = myreaderorderrider("Rider_ID").ToString
```

```

If myreaderorderrider.IsDBNull(0) Then
    MsgBox("We are assigning rider for you, thanks for waiting.",_
    MessageBoxButtons.Information, "UUM Cafe YAB")
    myreaderorderrider.Close()
    con.Close()

```

Else

```
Dim strsql As String
strsql = "select [Order_Status] , [Payment_Total] , [Payment_Type] , [Cu_Remark] ,
[Cu_Office_Inasis] , [Cu_Block_Room] , [Rider_ID] from Payment where [Payment_ID] = "" &
txtPaymentID.Text & """
Dim cmd As New OleDbCommand(strsql, con)
Dim myreader As OleDbDataReader
myreader = cmd.ExecuteReader
myreader.Read()

txtOrderStatus.Text = myreader("Order_Status")
txtTotalPrice.Text = Format(myreader("Payment_Total"), "0.00")
txtPaymentType.Text = myreader("Payment_Type")
txtCuRemark.Text = myreader("Cu_Remark")
txtOfficeInasis.Text = myreader("Cu_Office_Inasis")
txtBlockRoom.Text = myreader("Cu_Block_Room")
txtRiderID.Text = myreader("Rider_ID")

If txtOfficeInasis.Text = "N/A" Then
    txtPaymentID.Text = ""
    txtOrderStatus.Text = ""
    txtTotalPrice.Text = ""
    txtPaymentType.Text = ""
    txtCuRemark.Text = ""
    txtOfficeInasis.Text = ""
    txtBlockRoom.Text = ""
    MsgBox("Please check again of your Payment ID, only delivery order can be
tracked.", MessageBoxButtons.Information, "UUM Cafe YAB")
    txtPaymentID.Focus()
```

```
myreader.Close()
con.Close()

Else

    'Check Rider
    Dim strsqlrider As String
    strsqlrider = "select [Rider_Name] , [Rider_HP] from Rider where [Rider_ID] = " &
txtRiderID.Text & ""

    Dim cmdrider As New OleDbCommand(strsqlrider, con)
    Dim myreaderrider As OleDbDataReader
    myreaderrider = cmdrider.ExecuteReader
    myreaderrider.Read()

    txtRiderName.Text = myreaderrider("Rider_Name")
    txtRiderHP.Text = myreaderrider("Rider_HP")
    myreaderrider.Close()
    con.Close()

End If

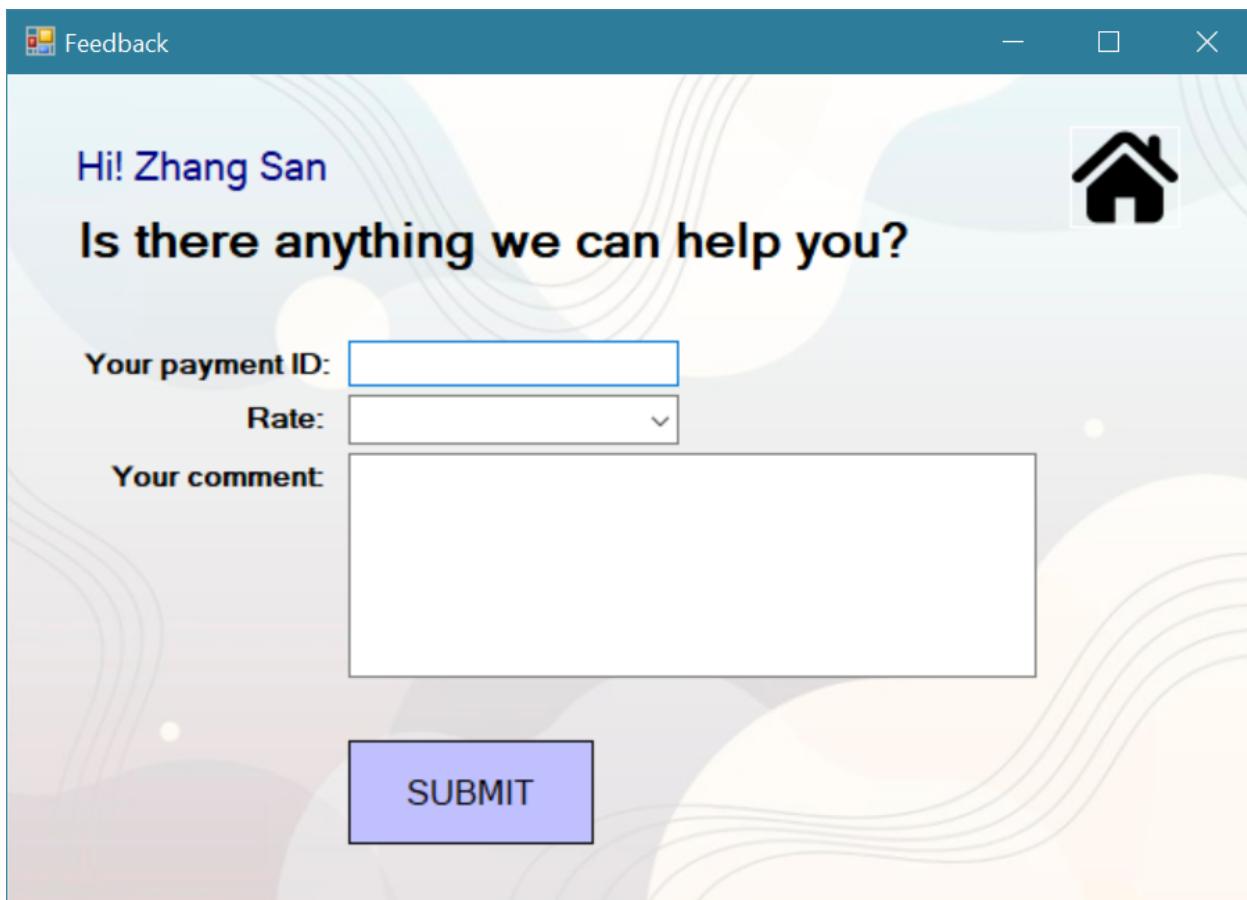
End If

End If

End Sub

Private Sub cmdBack_Click(sender As Object, e As EventArgs) Handles cmdBack.Click
    Me.Hide()
    frmCustMainMenu.Show()
End Sub

End Class
```

**Form 8: frmFeedback**Coding

```
Imports System.Data.OleDb
```

```
Public Class frmFeedback
```

```
    Dim pro As String
```

```
    Dim connstring As String
```

```
    Dim command As String
```

```
    Dim myconnection As OleDbConnection = New OleDbConnection
```

```
Private Sub frmFeedback_Load(sender As Object, e As EventArgs) Handles MyBase.Load
```

```
    txtCustName.Text = "Hi! " & (frmLogin.txtCustomerName.Text)
```

```
    txtCustID.Text = frmLogin.txtCustomerID.Text
```

```
End Sub
```

```

Private Sub cmdSubmit_Click(sender As Object, e As EventArgs) Handles cmdSubmit.Click
    If (txtPaymentID.Text = String.Empty) Or (txtRate.Text = String.Empty) Then
        MsgBox("Please fill in the relevant Payment ID or Rate.", MessageBoxIcon.Information,
        "UUM Cafe YAB")
    End If

    If (txtComment.Text = String.Empty) Then
        txtComment.Text = "N/A"
    End If

    pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
    connstring = pro
    myconnection.ConnectionString = connstring
    myconnection.Open()

    command = "insert into [Feedback] ([Payment_ID] , [Customer_ID] , [Fdb_Date] ,
[Cu_Rate] , [Cu_Comm]) values ('" & txtPaymentID.Text & "','" & txtCustID.Text & "','" &
dateFdbDate.Value & "','" & txtRate.Text & "','" & txtComment.Text & "')"
    Dim custfeedback As OleDbCommand = New OleDbCommand(command,
myconnection)
    custfeedback.Parameters.Add(New OleDbParameter("Payment_ID",
 CType(txtPaymentID.Text, String)))
    custfeedback.Parameters.Add(New OleDbParameter("Customer_ID",
 CType(txtCustID.Text, String)))
    custfeedback.Parameters.Add(New OleDbParameter("Fdb_Date",
 CType(dateFdbDate.Value = DateTime.Now, String)))
    custfeedback.Parameters.Add(New OleDbParameter("Cu_Rate", CType(txtRate.Text,
String)))

```

```
custfeedback.Parameters.Add(New OleDbParameter("Cu_Comm",
CType(txtComment.Text, String)))

custfeedback.ExecuteNonQuery()
custfeedback.Dispose()
myconnection.Close()

txtPaymentID.Clear()
txtRate.Text = String.Empty
txtComment.Clear()
dateFdbDate.Value = DateTime.Now

MsgBox("Thanks for your response!", MessageBoxButtons.Information, "UUM Cafe YAB")
Me.Hide()
frmCustMainMenu.Show()

End Sub

Private Sub cmdBack_Click(sender As Object, e As EventArgs) Handles cmdBack.Click
Me.Hide()
frmCustMainMenu.Show()
End Sub

End Class
```

## Form 9: frmStaff

**Staff**

Welcome back! Ooi Cheong Wai, ST0002      Uum Cafe YAB

**Customer Order**

**Rider Duty**

Order_Num	Customer_ID	Customer_Name	Item_Menu	Item_Price	Item_Quantity	Payl ^
1	CU0001	Vieem A/L Hrithik Roshan	Green Tea	3.9	1	P0001
3	CU0001	Vieem A/L Hrithik Roshan	Chicken Burger	9.9	3	P0001
40	CU0006	Zhang San	Milk Tea	11.7	3	P001
41	CU0006	Zhang San	Milk Tea	7.8	2	P001
42	CU0006	Zhang San	Green Tea	3.9	1	P001

<> < > >>      ADD EDIT DELETE

Payment ID: P0001      ORDER FILTER  
 Order Number: 1      FILTER  
 Customer ID: CU0001      FILTER  
 Customer Name: Vieem A/L Hrithik Roshan      FILTER  
 Item Menu: Green Tea      FILTER

RESET FILTER



**Logout** 

**Staff**

Welcome back! Ooi Cheong Wai, ST0002      Uum Cafe YAB

**Customer Order**

**Rider Duty**

Payment_ID	Promo_Code	Order_Type	Payment_SubTotal	Delivery_Fee	Payment_Total	Payment_Type ^
P0007	N/A	Dine In	37.8	0.0000	37.8	Cash
P0008	N/A	Delivery	23.7	1	24.7	Cash
P0014			0.0000	0.0000	0.0000	
P0009	N/A	Delivery	11.7	0.0000	11.7	Cash
P0012	N/A	Delivery	19.6	0.0000	19.6	Cash

<> < > >>

**Check for DELIVERY Type Order:**

FILTER DELIVERY  
 CLEAR DELIVERY FILTER

**Assign a RIDER to order:**

Which payment ID:  Assign a rider ID:   
 UPDATE



**Logout** 

Coding

Imports System.Data.OleDb

Public Class frmStaff

Dim pro As String

```

Dim connstring As String
Dim command As String
Dim myconnection As OleDbConnection = New OleDbConnection

Private Sub cmdStaffRecord_Click(sender As Object, e As EventArgs) Handles
cmdCustomerOrder.Click
    pnlOnPosition.Height = cmdCustomerOrder.Height
    pnlOnPosition.Top = cmdCustomerOrder.Top
    pnlCustomerOrder.Visible = True
    pnlRiderDuty.Visible = False
End Sub

Private Sub cmdRiderDuty_Click(sender As Object, e As EventArgs) Handles
cmdRiderDuty.Click
    pnlOnPosition.Height = cmdRiderDuty.Height
    pnlOnPosition.Top = cmdRiderDuty.Top
    pnlCustomerOrder.Visible = False
    pnlRiderDuty.Visible = True
End Sub

Private Sub frmStaff_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'TODO: This line of code loads data into the 'UUMCafeDatabase5DataSet.Payment' table.
    You can move, or remove it, as needed.
    Me.PaymentTableAdapter.Fill(Me.UUMCafeDatabase5DataSet.Payment)
    'TODO: This line of code loads data into the 'UUMCafeDatabase5DataSet.CustomerOrder'
    table. You can move, or remove it, as needed.
    Me.CustomerOrderTableAdapter.Fill(Me.UUMCafeDatabase5DataSet.CustomerOrder)

    txtStaffNameID.Text = (frmLogin.txtStaffName.Text) & ", " & (frmLogin.txtStaffID.Text)
End Sub

```

```
Private Sub CustomerOrderBindingNavigatorSaveItem_Click(sender As Object, e As EventArgs) Handles CustomerOrderBindingNavigatorSaveItem.Click
```

```
    Me.Validate()
```

```
    Me.CustomerOrderBindingSource.EndEdit()
```

```
    Me.TableAdapterManager.UpdateAll(Me.UUMCafeDatabase5DataSet)
```

```
End Sub
```

```
'Customer Order
```

```
Private Sub cmdFilterPaymentID_Click(sender As Object, e As EventArgs) Handles cmdFilterPaymentID.Click
```

```
    CustomerOrderBindingSource.Filter = "Payment_ID LIKE'" & txtPaymentID.Text & "'"
```

```
End Sub
```

```
Private Sub cmdFilterOrderNum_Click(sender As Object, e As EventArgs) Handles cmdFilterOrderNum.Click
```

```
    CustomerOrderBindingSource.Filter = "[Order_Num] ='" & txtOrderNum.Text & "'"
```

```
End Sub
```

```
Private Sub cmdFilterCustomerID_Click(sender As Object, e As EventArgs) Handles cmdFilterCustomerID.Click
```

```
    CustomerOrderBindingSource.Filter = "[Customer_ID] ='" & txtCustomerID.Text & "'"
```

```
End Sub
```

```
Private Sub cmdFilterCustomerName_Click(sender As Object, e As EventArgs) Handles cmdFilterCustomerName.Click
```

```
    CustomerOrderBindingSource.Filter = "[Customer_Name] ='" & txtCustomerName.Text & "'"
```

```
End Sub
```

```
Private Sub cmdFilterItemMenu_Click(sender As Object, e As EventArgs) Handles  
cmdFilterItemMenu.Click  
    CustomerOrderBindingSource.Filter = "[Item_Menu] = " & txtItemMenu.Text & """  
End Sub
```

```
Private Sub cmdClearFilter_Click(sender As Object, e As EventArgs) Handles  
cmdClearFilter.Click  
    CustomerOrderBindingSource.RemoveFilter()  
End Sub
```

```
Private Sub cmdMoveFirst1_Click(sender As Object, e As EventArgs) Handles  
cmdMoveFirst1.Click  
    CustomerOrderBindingSource.MoveFirst()  
End Sub
```

```
Private Sub cmdMovePrevious1_Click(sender As Object, e As EventArgs) Handles  
cmdMovePrevious1.Click  
    CustomerOrderBindingSource.MovePrevious()  
End Sub
```

```
Private Sub cmdMoveNext1_Click(sender As Object, e As EventArgs) Handles  
cmdMoveNext1.Click  
    CustomerOrderBindingSource.MoveNext()  
End Sub
```

```
Private Sub cmdMoveLast1_Click(sender As Object, e As EventArgs) Handles  
cmdMoveLast1.Click  
    CustomerOrderBindingSource.MoveLast()  
End Sub
```

```
Private Sub cmdAdd1_Click(sender As Object, e As EventArgs) Handles cmdAdd1.Click
```

```
If cmdAdd1.Text = "ADD" Then
    cmdMoveFirst1.Hide()
    cmdMovePrevious1.Hide()
    cmdMoveNext1.Hide()
    cmdMoveLast1.Hide()
    cmdEdit1.Hide()
    cmdDelete1.Hide()
    cmdClearFilter.Hide()
    cmdFilterPaymentID.Hide()
    cmdFilterOrderNum.Hide()
    cmdFilterCustomerID.Hide()
    cmdFilterCustomerName.Hide()
    cmdFilterItemMenu.Hide()
    cmdRiderDuty.Hide()
    CustomerOrderBindingSource.AddNew()
    cmdAdd1.Text = "SAVE NEW"

Else
    CustomerOrderBindingSource.EndEdit()
    TableAdapterManager.UpdateAll(UUMCafeDatabase5DataSet)
    CustomerOrderBindingSource.MoveFirst()
    cmdAdd1.Text = "ADD"
    cmdMoveFirst1.Show()
    cmdMovePrevious1.Show()
    cmdMoveNext1.Show()
    cmdMoveLast1.Show()
    cmdEdit1.Show()
    cmdDelete1.Show()
    cmdClearFilter.Show()
    cmdFilterPaymentID.Show()
    cmdFilterOrderNum.Show()
    cmdFilterCustomerID.Show()
```

```
cmdFilterCustomerName.Show()  
cmdFilterItemMenu.Show()  
cmdRiderDuty.Show()  
End If  
End Sub
```

```
Private Sub cmdEdit1_Click(sender As Object, e As EventArgs) Handles cmdEdit1.Click
```

```
If cmdEdit1.Text = "EDIT" Then  
    cmdMoveFirst1.Hide()  
    cmdMovePrevious1.Hide()  
    cmdMoveNext1.Hide()  
    cmdMoveLast1.Hide()  
    cmdAdd1.Hide()  
    cmdDelete1.Hide()  
    cmdClearFilter.Hide()  
    cmdFilterPaymentID.Hide()  
    cmdFilterOrderNum.Hide()  
    cmdFilterCustomerID.Hide()  
    cmdFilterCustomerName.Hide()  
    cmdFilterItemMenu.Hide()  
    cmdRiderDuty.Hide()  
    cmdEdit1.Text = "SAVE"
```

```
Else  
    CustomerOrderBindingSource.EndEdit()  
    TableAdapterManager.UpdateAll(UUMCafeDatabase5DataSet)  
    CustomerOrderBindingSource.MoveFirst()  
    cmdEdit1.Text = "EDIT"  
    cmdMoveFirst1.Show()  
    cmdMovePrevious1.Show()  
    cmdMoveNext1.Show()  
    cmdMoveLast1.Show()
```

```

cmdAdd1.Show()
cmdDelete1.Show()
cmdClearFilter.Show()
cmdFilterPaymentID.Show()
cmdFilterOrderNum.Show()
cmdFilterCustomerID.Show()
cmdFilterCustomerName.Show()
cmdFilterItemMenu.Show()
cmdRiderDuty.Show()

End If

End Sub

```

```

Private Sub cmdDelete1_Click(sender As Object, e As EventArgs) Handles cmdDelete1.Click
    Dim user_respond As Integer

    user_respond = MessageBox.Show("Are you sure to delete this order? Reminder that this
cannot be undone once deleted.", "DELETE?", MessageBoxButtons.YesNo,
MessageBoxIcon.Warning)

```

```

If user_respond = vbYes Then
    CustomerOrderBindingSource.RemoveCurrent()
    TableAdapterManager.UpdateAll(UUMCafeDatabase5DataSet)
End If

End Sub

```

```

'Rider Duty

Private Sub cmdDeliveryFilter_Click(sender As Object, e As EventArgs) Handles
cmdDeliveryFilter.Click
    PaymentBindingSource.Filter = "Order_Type LIKE '" & txtDelivery.Text & "'"
End Sub

```

```
Private Sub cmdClearDeliveryFilter_Click(sender As Object, e As EventArgs) Handles cmdClearDeliveryFilter.Click
    PaymentBindingSource.RemoveFilter()
End Sub

Private Sub cmdUpdate_Click(sender As Object, e As EventArgs) Handles cmdUpdate.Click
    If txtPaymentID2.Text = String.Empty Then
        MsgBox("Please choose a payment ID for rider to be assigned.", MessageBoxButtons.Information, "UUM Cafe YAB")
    Else
        pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access Database\UUMCafeDatabase5.accdb"
        connstring = pro
        myconnection.ConnectionString = connstring
        myconnection.Open()

        command = "UPDATE Payment SET [Rider_ID] = '" & txtRiderID.Text & "' where [Payment_ID] = '" & txtPaymentID2.Text & "'"

        Dim cmd As OleDbCommand = New OleDbCommand(command, myconnection)
        cmd.ExecuteNonQuery()
        cmd.Dispose()

        MsgBox("Done assign.", MessageBoxButtons.Information, "UUM Cafe YAB")

        myconnection.Close()
    End If
End Sub
```

```
Private Sub cmdLogout_Click(sender As Object, e As EventArgs) Handles cmdLogout.Click
    Dim user_respond As Integer
    user_respond = MessageBox.Show("Are you sure to log out?", "LOGOUT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)
```

```
If user_respond = vbYes Then
```

```
    Me.Close()
```

```
    frmLogin.Show()
```

```
End If
```

```
End Sub
```

```
Private Sub cmdExit_Click(sender As Object, e As EventArgs) Handles cmdExit.Click
```

```
    Dim user_respond As Integer
```

```
    user_respond = MessageBox.Show("Are you sure to exit the application?", "EXIT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)
```

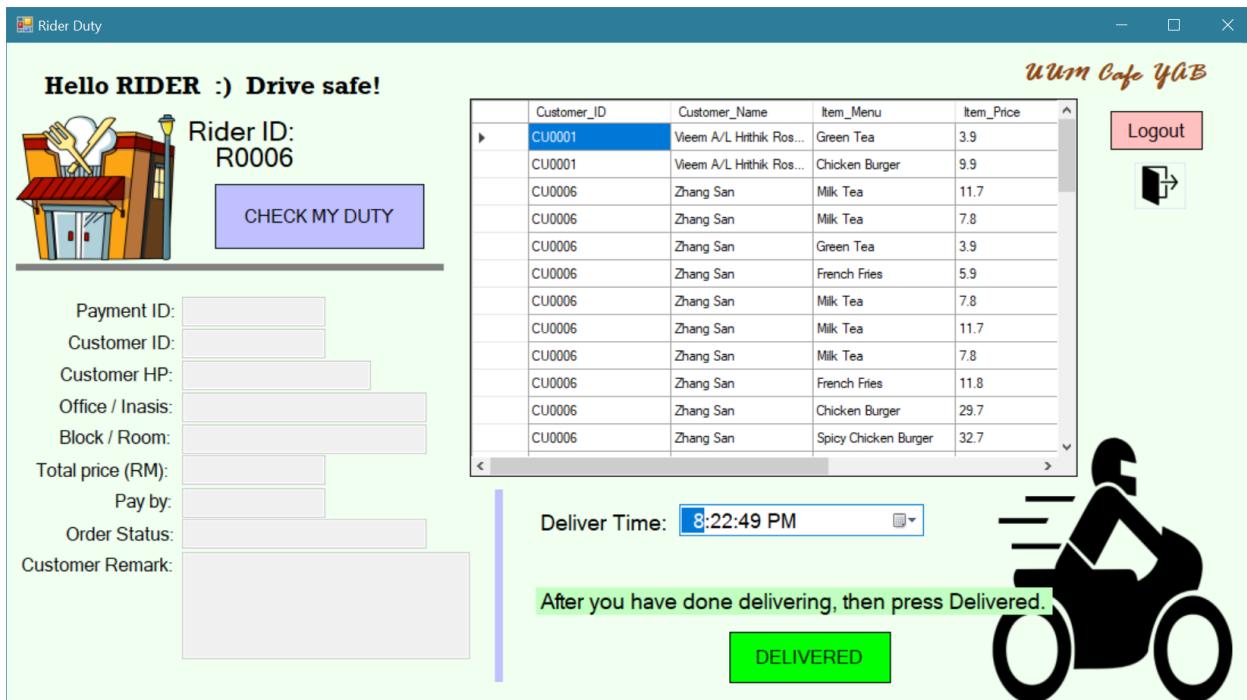
```
If user_respond = vbYes Then
```

```
    Application.Exit()
```

```
End If
```

```
End Sub
```

```
End Class
```

**Form 10: frmRiderDuty**Coding

```
Imports System.Data.OleDb
```

```
Public Class frmRiderDuty
```

```
    Dim pro As String
```

```
    Dim connstring As String
```

```
    Dim command As String
```

```
    Dim myconnection As OleDbConnection = New OleDbConnection
```

```
    Dim con As New OleDbConnection
```

```
Private Sub CustomerOrderBindingNavigatorSaveItem_Click(sender As Object, e As EventArgs) Handles CustomerOrderBindingNavigatorSaveItem.Click
```

```
    Me.Validate()
```

```
    Me.CustomerOrderBindingSource.EndEdit()
```

```
    Me.TableAdapterManager.UpdateAll(Me.UUMCafeDatabase5DataSet)
```

```
End Sub
```

```
Private Sub frmRiderDuty2_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    'TODO: This line of code loads data into the 'UUMCafeDatabase5DataSet.CustomerOrder'
    'table. You can move, or remove it, as needed.
```

```
    Me.CustomerOrderTableAdapter.Fill(Me.UUMCafeDatabase5DataSet.CustomerOrder)
```

```
    txtRiderID.Text = frmLogin.txtRiderID.Text
```

```
End Sub
```

```
Private Sub cmdCheckMyDuty_Click(sender As Object, e As EventArgs) Handles
cmdCheckMyDuty.Click
    con.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
    con.Open()
```

```
    Dim strsql As String
    strsql = "select [Payment_ID] , [Customer_ID] , [Cu_HP] , [Payment_Total] ,
[Payment_Type] , [Order_Status] , [Cu_Remark] , [Cu_Office_Inasis] , [Cu_Block_Room] from
Payment where [Rider_ID] = '" & txtRiderID.Text & "'"
    Dim cmd As New OleDbCommand(strsql, con)
    Dim myreader As OleDbDataReader
    myreader = cmd.ExecuteReader
    myreader.Read()
```

```
    txtPaymentID.Text = myreader("Payment_ID")
    txtCustomerID.Text = myreader("Customer_ID")
    txtCustomerHP.Text = myreader("Cu_HP")
    txtTotalPrice.Text = Format(myreader("Payment_Total"), "0.00")
    txtPaymentType.Text = myreader("Payment_Type")
    txtOrderStatus.Text = myreader("Order_Status")
```

```

txtCuRemark.Text = myreader("Cu_Remark")
txtOfficeInasis.Text = myreader("Cu_Office_Inasis")
txtBlockRoom.Text = myreader("Cu_Block_Room")

CustomerOrderBindingSource.Filter = "Payment_ID LIKE '" & txtPaymentID.Text & "'"

con.Close()

End Sub

```

```

Private Sub cmdDelivered_Click(sender As Object, e As EventArgs) Handles
cmdDelivered.Click

If txtPaymentID.Text = String.Empty Then
    MsgBox("Please check your duty first.", MessageBoxIcon.Information, "UUM Cafe
YAB")

Else
    con.ConnectionString = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access
Database\UUMCafeDatabase5.accdb"
    con.Open()

    Dim strSQL As String
    strSQL = "SELECT [Delivered_Time] FROM Payment WHERE [Payment_ID] = '" &
txtPaymentID.Text & "'"

    Dim cmd As New OleDbCommand(strSQL, con)
    Dim myreader As OleDbDataReader
    myreader = cmd.ExecuteReader()
    myreader.Read()

    dateDeliverTime.Value = myreader("Delivered_Time")

```

```
If myreader IsNot DBNull.Value Then
```

```
    dateDeliverTime.Value = DateTime.Now
```

```
    pro = "Provider=Microsoft.ACE.OLEDB.12.0;Data Source=D:\Access  
Database\UUMCafeDatabase5.accdb"
```

```
    connstring = pro
```

```
    myconnection.ConnectionString = connstring
```

```
    myconnection.Open()
```

```
    command = "UPDATE Payment SET [Delivered_Time] = '" &  
dateDeliverTime.Value & "' where [Payment_ID] = '" & txtPaymentID.Text & "'"
```

```
    Dim command As OleDbCommand = New OleDbCommand(command,  
myconnection)
```

```
    command.ExecuteNonQuery()
```

```
    myreader.Close()
```

```
    myconnection.Close()
```

```
    con.Close()
```

```
    MsgBox("Done delivering and updated in the system. Contact the staff if you have  
accidentally pressed this button.", MessageBoxIcon.Information, "UUM Cafe YAB")
```

```
Else
```

```
    MsgBox("You have delivered this payment, contact the staff if error.",  
MessageBoxIcon.Error, "UUM Cafe YAB")
```

```
    myreader.Close()
```

```
    con.Close()

End If

End If

End Sub

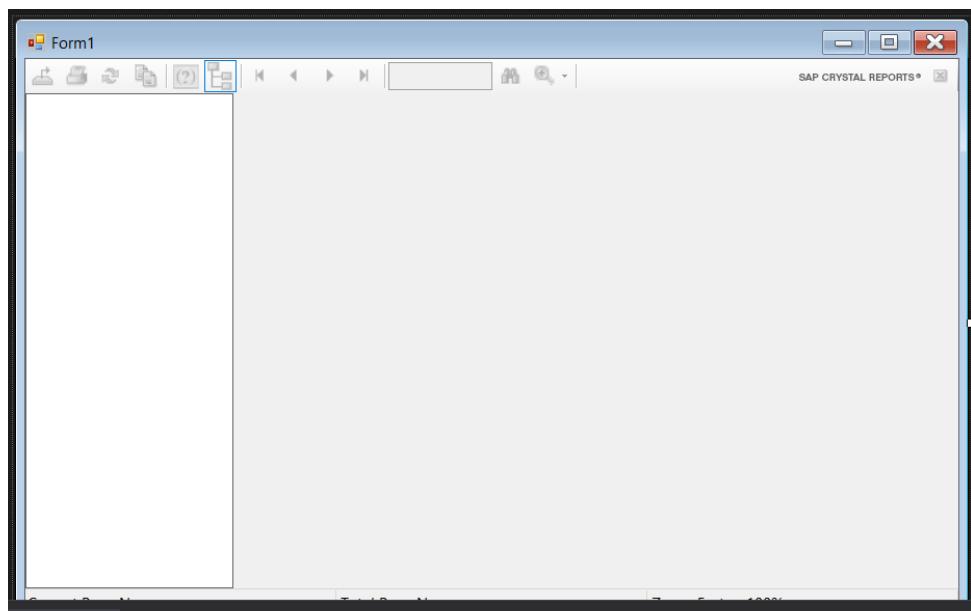
Private Sub cmdLogout_Click(sender As Object, e As EventArgs) Handles cmdLogout.Click
    Dim user_respond As Integer
    user_respond = MessageBox.Show("Are you sure to log out?", "LOGOUT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)

    If user_respond = vbYes Then
        Me.Close()
        frmLogin.Show()
    End If
End Sub

Private Sub Button3_Click(sender As Object, e As EventArgs) Handles Button3.Click
    Dim user_respond As Integer
    user_respond = MessageBox.Show("Are you sure to exit the application?", "EXIT?",
        MessageBoxButtons.YesNo, MessageBoxIcon.Warning)

    If user_respond = vbYes Then
        MessageBox.Show("Thank you for using our application." & vbCrLf & "We hope to
see you again!", "BYE~")
        Application.Exit()
    End If
End Sub

End Class
```

**Form 11: frmCrystalReport**A screenshot of the SAP Crystal Reports software interface, specifically the "Main Report" pane. The report structure is outlined on the left side. It includes:

- Section1 (Report Header )
- Section2 (Page Header )
  - Print Date
  - Payment\_ID Customer\_ID Promo\_Code Order\_Type Payment\_Total Payment\_Type Payment\_Date Payment\_Time
- Section3 (Details )
  - Payment\_ID Customer\_ID Promo\_Code Order\_Type Payment\_Total Payment\_Type Payment\_Date Payment\_Time
- Section4 (Report Footer )
- Section5 (Page Footer )

The "Main Report" tab is selected at the bottom of the interface.

The screenshot shows a Crystal Reports interface with a report titled "Main Report". The report displays a table of payment records. The table has the following columns: Payment Date, Payment Time, Payment Type, Payment Total, Order Type, Promo Code, Customer ID, and Payment ID. The data in the table is as follows:

Payment Date	Payment Time	Payment Type	Payment Total	Order Type	Promo Code	Customer ID	Payment ID
7/2/2022		Cash	RM 37.80	Dine In	N/A	CU0006	P0007
			RM 24.70	Delivery	N/A	CU0006	P0008
			RM 0.00				P0014
			RM 11.70	Delivery	N/A	CU0006	P0009
			RM 19.60	Delivery	N/A	CU0006	P0012
			RM 25.50	Delivery	N/A	CU0006	P0017

## Public Class Form1

```

Private Sub Form1_Load(sender As Object, e As EventArgs) Handles MyBase.Load
    Dim ds As New UUMCafeDatabase5DataSet
    Dim da As New UUMCafeDatabase5DataSetTableAdapters.PaymentTableAdapter
    da.Fill(ds.Payment)
    Dim rpt As New CrystalReport1
    rpt.SetDataSource(ds)
    CrystalReportViewer1.ReportSource = rpt
End Sub
End Class

```

## 10.0 Conclusion

Nowadays, all people are encouraged to stay at home to reduce the spread of the virus Covid-19. Therefore, the UUM Cafe restaurant decided to create a system for the customers to make an order in an online store and review our menus at home, for the staff to add, edit, and view customer information, orders, and payments and also for the manager with the information on revenue, graphs, market growth, and employee performance.

Microsoft Visual Basic and Microsoft Access are the applications that we use to create this system for the UUM Cafe restaurant. This system allows the user to add, edit, delete and filter information. After signing up, the user will log in using their username and password. There will be discount prizes and a promo code for loyal customers. This system has over 30 menus and daily sets for the user to choose from. Festival promotion will also be included in the system. When using their phone to place an order, users must fill in their contact number and address, and also select a payment method. We allow online orders, reservations for dine-in and take-out. We will also use delivery boys as couriers to deliver the meal to the user. This system will present the receipt to the customer when the customer has already paid it.

UUM Cafe restaurant can store all the records in the database automatically. We will be able to enter data into the system, which will then enable us to collect and store it. We can record, locate and update data more quickly anytime and anywhere. The cafe might be promoted more effectively, and we will be better equipped to negotiate with it. We can provide value to the business. We can improve the execution process, management, and strategic planning with this software. We can improve the intangible services we provide. We may improve the quality of the process and services provided by reading customers' feedback. We can save money by not wasting resources and dramatically reducing the number of people required to work physically. During the progress of the software, we may also limit the effects of data and project mismanagement. We can make the procedure update go more smoothly.

Here attached the link of final project presentation: <https://youtu.be/M8n9tU7gHRQ>

## 11.0 Reference

- Elmasri, R., & Navathe, S. B. (2011). *Database Systems Models, Languages, Design and Application Programming* (6 ed.). Boston: Pearson.
- Garcia-Molina, H., Ullman, J. D., & Widom, J. (2009). *Database Systems The Complete Book*. New Jersey: Prentice Hall.
- Connolly, T., & Begg, C. (2010). *Database Systems A practical Approach to Design, Implementation, and Management* (5 ed.). Boston: Pearson.
- Peckham, J., & Maryanski, F. (1988). Semantic data models. *ACM Comput. Surv.*, 20(3), 153-189