**MINISTRY OF EDUCATION AND TRAINING**

**UNIVERSITY OF TECHNOLOGY AND EDUCATION**

# **FACULTY OF INFORMATION TECHNOLOGY**

# **--------------------**

# **SUBJECT PROJECT**

**DATABASE MANAGEMENT SYSTEM**

**RESTAURANT MANAGEMENT SOFTWARE**

**STUDENT LIST:**

1. **NGUYỄN MẠNH TIẾN 17145370**
2. **NGUYỄN HOÀNG TRƯỜNG MINH**

**LECTURER: Lê Thị Châu**

# **Ho Chi Minh City** **– 2019**

**MINISTRY OF EDUCATION AND TRAINING**

**UNIVERSITY OF TECHNOLOGY AND EDUCATION**

# **FACULTY OF INFORMATION TECHNOLOGY**

# **--------------------**

# **SUBJECT PROJECT**

**DATABASE MANAGEMENT SYSTEM**

**RESTAURANT MANAGEMENT SOFTWARE**

**STUDENT LIST:**

1. **NGUYỄN MẠNH TIẾN 17145370**
2. **NGUYỄN HOÀNG TRƯỜNG MINH**

**LECTURER: Lê Thị Châu**

# **Ho Chi Minh City** **– 2019**

**Lecturer's evaluation**

*Ho Chi Minh City, Jun, 2019*

**Lecturer**

**THANK YOU**

In the first words of this end-to-end of "Database Management System" report, we would like to give our sincere thanks and gratitude to all those who have supported and provided us with knowledge and spirituality during project implementation.

We would like to express our thankful to Ms. Le Thi Chau, lecturer at Ho Chi Minh City University of Technology and Education, and our classmates that helped us throughout the Database Management System course.

Since knowledge is unlimited but ours acknowledge is limited, this program might contain unexpected out-coming. We would like to receive feedback from you in order that we could have more experience to implement our project better.

Best regards!

Ho Chi Minh City, Jun, 2019

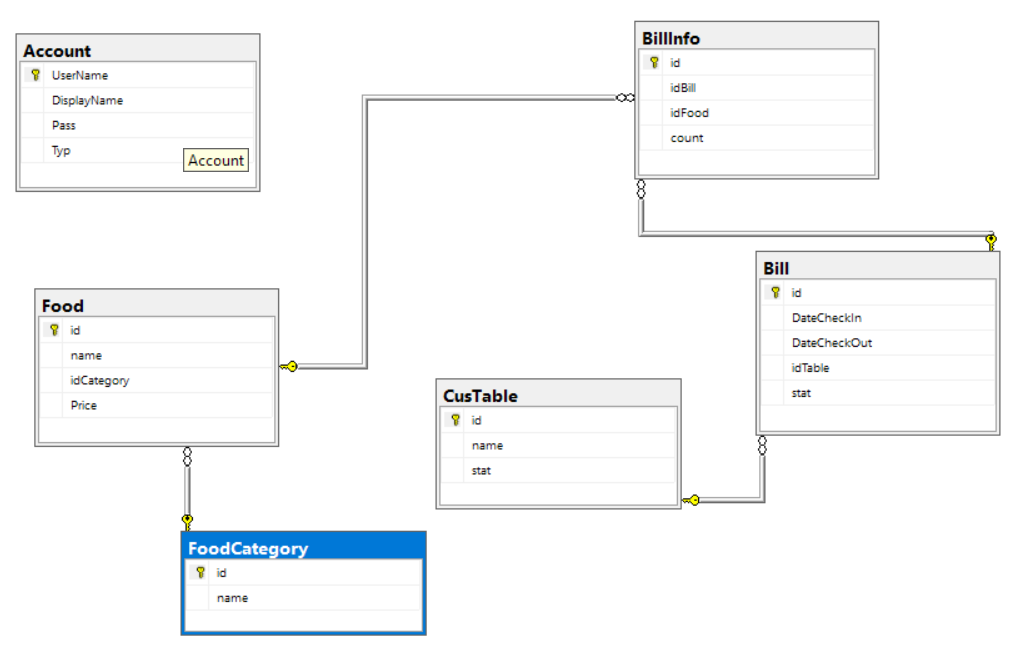
Team TĐTT

Leader

**Chapter 1: Introduction/ Main Idea**

It is very difficult to manage a chain of tables and dishes that customers call, so the group decided to create a cafe management software to make it easier to manage the tables in the restaurant as well as the statistics.

And the most important reasons in creating this project is : the project is a mixture of everything we have learnt so far in Object-Oriented Programming course and Database Management System that is : **Class and Object , C# Winform and its tool, SQL programmabilities : Trigger, Function, Stored Procedure, Index,…**

**Chapter 2: Database Design and Diagram**

**2.1 Table Design**

|  |  |  |
| --- | --- | --- |
| Number | Table Name | Store information of |
| 1 | Account | Account Table |
| 2 | Food | Food details |
| 3 | Table | Table details |
| 4 | Bill | Bill details |
| 5 | foodCategory | Category of food |
| 6 | BillInfo | Detail content of each bill |

Account Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Datafield name | explain | Datatype | Key |
| 1 | Username | Username of user | Int(32) | Primary key |
| 2 | DisplayName | Name for displaying | Nvarchar(50) |  |
| 3 | Pass | Password of the account | Nvarchar(50) |  |
| 4 | Typ | Role of user | Int(32) |  |

Food table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Datafield name | Explain | datatype | key |
| 1 | id | Id of each food | Int(32) | Primary key |
| 2 | name | Name of food | Nvarchar(50) |  |
| 3 | idCategory | Category | Int(32) | Foreign Key |
| 4 | Price | Price of the food | float |  |

CusTable table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Datafield name | Explain | datatype | key |
| 1 | id | Id of each table | Int(32) | Primary key |
| 2 | name | Name of each table | Nvarchar(50) |  |
| 3 | stat | Current status | Nvarchar(50) |  |

FoodCategory table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Datafield name | Explain | datatype | key |
| 1 | id | Id of category | Int(32) | Primary key |
| 2 | name | Name of category | Nvarchar(50) |  |

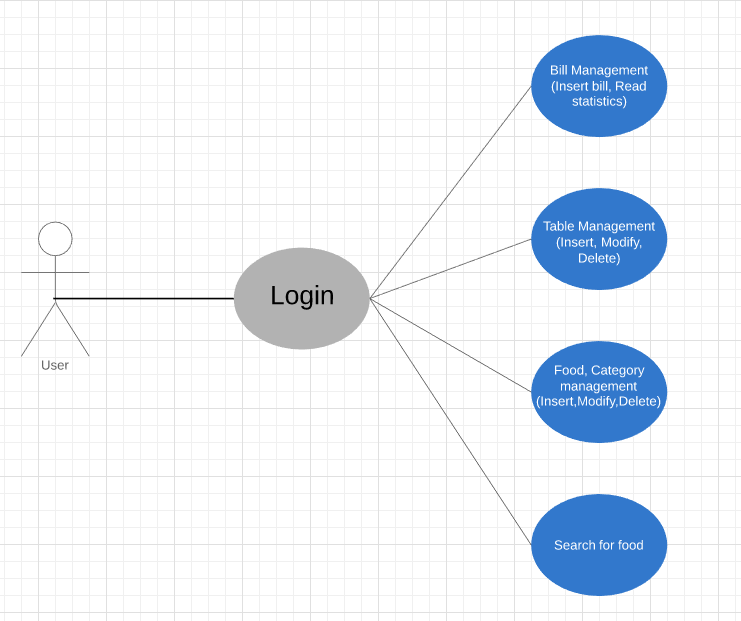
Bill Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Datafield name | Explain | datatype | key |
| 1 | id | Id of the bill | Int(32) | Primary key |
| 2 | DateCheckIn | Check in time | DateTime |  |
| 3 | DateCheckOut | Check out time | DateTime |  |
| 4 | idTable | Id of the table | Int(32) | Foreign Key |
| 5 | stat | Status | Int(32) |  |
| 6 | totalPrice | Total price of the bill | Float |  |

BillInfo Table

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Datafield name | Explain | datatype | key |
| 1 | id | Id of each bill info | Int(32) | Primary key |
| 2 | idBill | Id of bill | Int(32) | Foreign Key |
| 3 | idFood | Id of food | Int(32) | Foreign Key |
| 4 | count | Number of food | Int(32) |  |

**2.2 : Usecase Diagram**

****

**2.3 : Programmability**

**2.3.1 Stored Procedure**

|  |  |  |
| --- | --- | --- |
| **Number** | **Name of Stored Procedure** | **Function** |
| 1 | dbo.USP\_Login | Check for login information and handle authentication |
| 2 | dbo.USP\_Menu | Display and calculate price for the order. |
| 3 | dbo.USP\_InsertBill | Insert Bill when check out |
| 4 | dbo.USP\_InsertBillInfo | Insert Bill Information when check out |
| 5 | dbo.USP\_GetTableList | Get the list of table |
| 6 | dbo.USP\_GetListBillByDate | Get the list of bill which had checked out by date |

**2.3.2 Trigger**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Name of trigger** | **Apply on** | **Function** |
| 1 | UTG\_UpdateBill | Bill Table | Update the status of table which has a bill updating. |
| 2 | UTG\_UpdateBillInfo | BillInfo Table | Update the status of table when that table’s bill information is insert/update |
| 3 | UTG\_DeleteFoodCategory | FoodCategory Table | Delete all the food of a category when that category is deleted |

**2.3.3 Function**

|  |  |  |
| --- | --- | --- |
| **Number** | **Name of Function** | **Function** |
| 1 | dbo.NormalizeString | Change Vietnamese words into non-symbol. |

**2.3.4 Constraint**

|  |  |  |  |
| --- | --- | --- | --- |
| **Number** | **Name of constraint** | **Apply on** | **Function** |
| 1 | DF\_\_Account\_\_Typ | Account Table | The type of account can only be 1 or 0 (admin or user) |
| 2 | DF\_\_Account\_\_Display | Account Table | Automatically set an account display name is “User” when a new acount is inserted without any detail for it’s role |
| 3 | DF\_\_Bill\_\_DateCheckIn | Bill Table | Automatically set the check in date is the current day’s date time when a bill is inserted. |
| 4 | DF\_\_BillInfo\_\_count | Bill Info | Automatically set the count property of a bill’s information to 0 when a new one is inserted. |
| 5 | DF\_\_CusTable\_\_stat | CusTable Table | Set default status value of a table is “Empty”. |
| 6 | DF\_\_Food\_\_name | Food Table | Set default name of a food is “No name” when a food is inserted without name property. |
| 7 | DF\_\_Food\_\_Price | Food Table | Set default price of a food is 0 when it’s price is not inserted. |

**CHAPTER 3.ADO.NET**

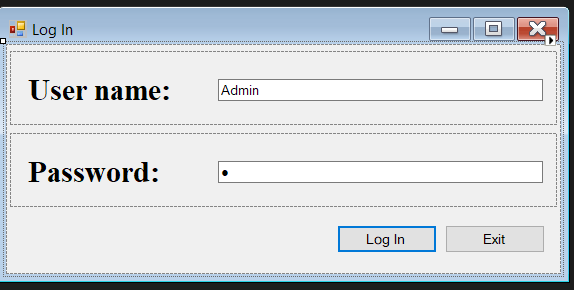
DO.NET is similar to ADO, the main difference is that ADO.NET is a disconnected, disconnected data architecture (Disconnected Data Architecture). With this architecture, data is received from the database and stored on the user's machine cache. Users can manipulate data they receive and only connect to the database when they need to change data streams or request new data.

ADO.NET connects to the database to retrieve data and connects back to update data when users change them. Most applications use a lot of time to read and display data, so ADO.NET has provided a subset of unconnected data objects for applications so that users can read and display them without connecting to the database.

These disconnected objects work similarly to Web applications.

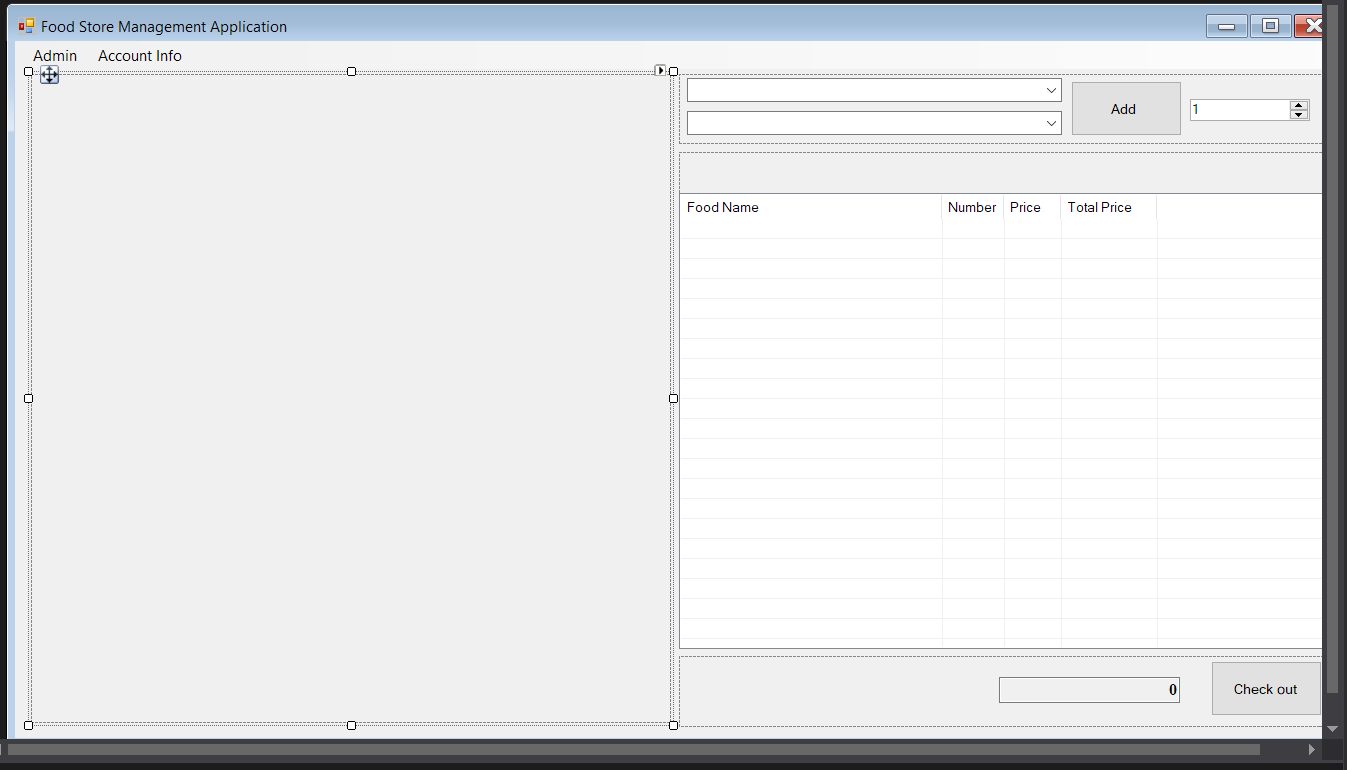
**3.1.Design**

**3.1.1 Login**



* This form will show in the first when user want to use this app thay must enter the ‘username’ and the ‘password’ correct in the database before.

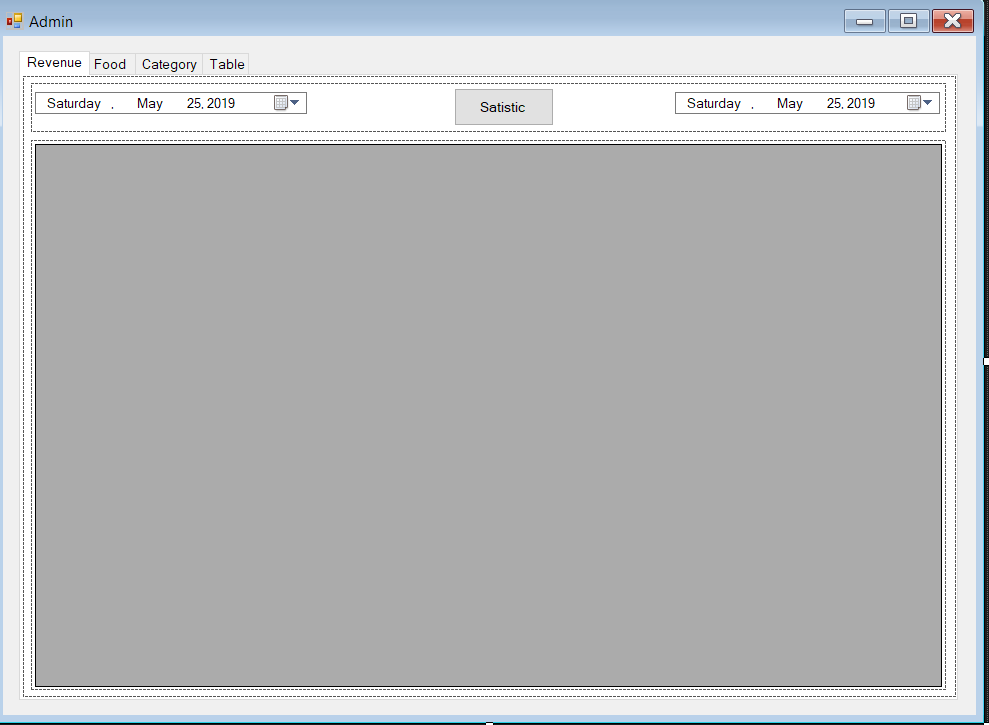
**3.1.2. Table**



* This form will show when you enter after Login
* The form have the table of all in the store on the left, and the bill is on the right, there also have the add mean ‘AddFood’ and the ‘CheckOut’ will show your information of your Bill.
* There have a menu to access the admin (revengue, food, category and table)

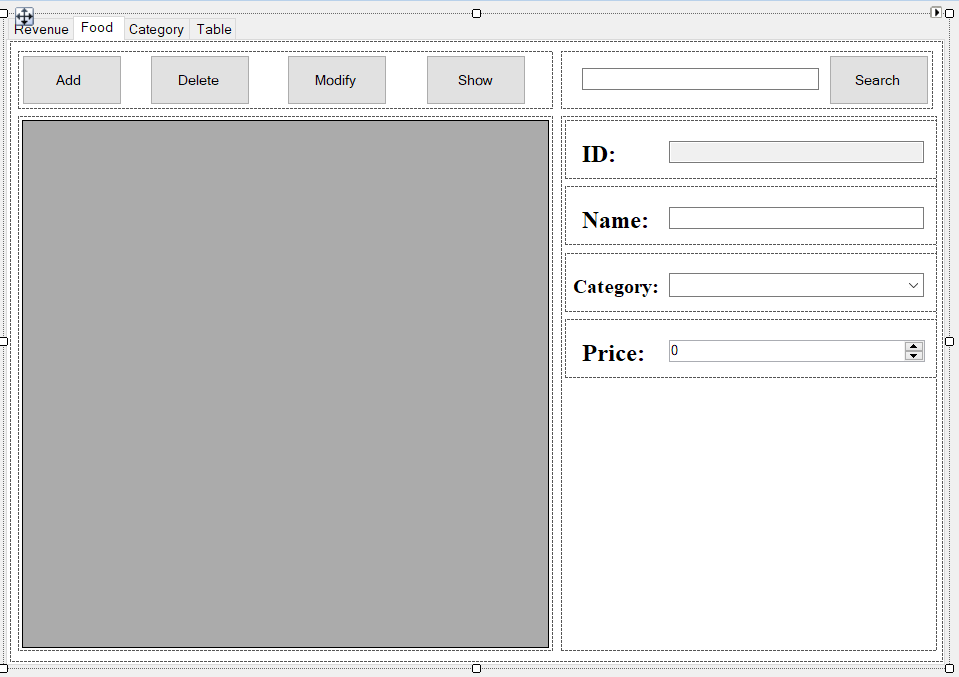
**3.1.3. Account**

**3.1.3.1. Revenue**

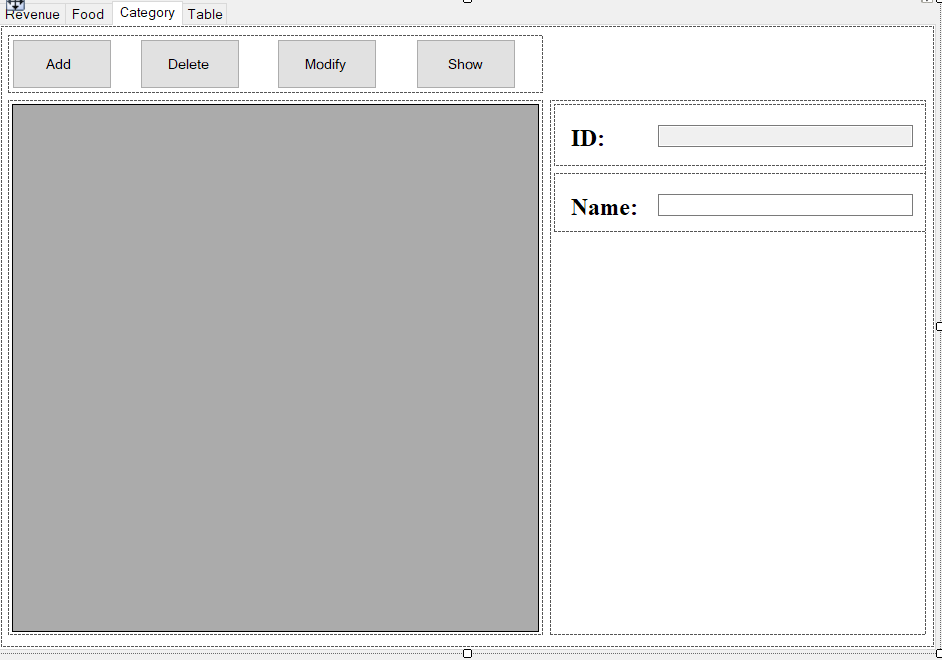


* This table will show revenue by month according to the way you choose on the datetimepicker
* The datalistview will take the data from database.

**3.1.3.2. Food**

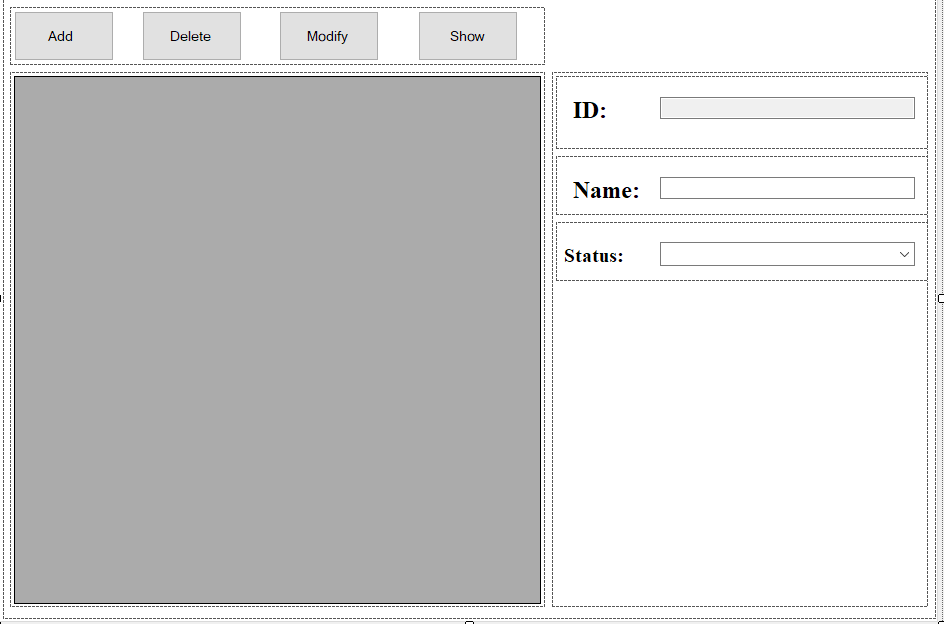


* This table will show food from database and show it on the datagridview, you can search the food, add, delete or modify it and then you can click show to show it

**3.1.3.3.Categorty**

* This show category from the database you can add and modify or delete mỏe category.

**3.1.3.4. Table**



* This show table from the database you can add and modify or delete more table.
* The status will make you know the table is empty or full.

**Chapter 4 :** **Installation and testing**

|  |  |  |
| --- | --- | --- |
|  | **Situation** | **Purpose** |
| 1 | Input: data from database  Output: show data to datagridview | Check the program can assign the database yes or not |
| 2 | Input: the data edit, delete the data, add the data  Output: change of data when adding, editing or deleting data | Check the code make sure that user can edit, add and del the data, and the data must change when user do that |
| 3 | Input: Login information  Output: Authenticated user can start using the program | Authentication |
| 4 | Input: Checkout with bill information  Output: Handle the bill, insert details into BillInfo Table | Handle the bill when a table is checked out |
| 5 | Deployed the application | Make sure that the application can work perfectly on the computer |

**CHAPTER 6. Conclude**

**6.1 Difficulties and solution**

|  |  |
| --- | --- |
| **Difficulties** | **Solution** |
| How to grant privileges for each kind of user (admin, normal user) when sign in. | * Checking the role property of user when authenticating. |
| Grating privileges for admin and user in database | * Still haven’t figured out |

**6.2. For further implementation**

1. Grant privilege for admin differrent from user.
2. Make the theme more beautiful and easier for user to use

**6.3. Advantages**

1. Simple easy-to-use software for managing a small restaurant
2. Know more about how to solve the problem

**6.4. Limit**

1. The IU is not good-looking

**Chapter 7: Task distribution**

|  |  |  |
| --- | --- | --- |
|  | Name | assignment |
| 1 | Nguyễn Mạnh Tiến | Login (user authentication), CheckOut, Statistics, UI, Write Report, Design database and code Triggers, Stored Procedures and Functions. |
| 2 | Nguyễn Hoàng Trường Minh | Food, Table management (insert, modify, delete), Create database connection class, UI, Search operation. |