**HCMC UNIVERSITY OF TECHNOLOGY AND EDUCATION**

**FACULTY FOR HIGH QUALITY TRAINING**

# **INFORMATION TECHNOLOGY**

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# 

**THE FIRST PROJECT REPORT**

**BUILDING AN APPLICATION GENERATE DATABASE SHEMA**

**LECTURER: Nguyễn Đức Khoan**

**STUDENT LIST:**

**1. Phạm Vũ Thiện 17110075**

1. **Trương Minh Quân 17110069**

# 

# **Ho Chi Minh City, November** **– 2019**

**THANK YOU**

In the first words of this end-to-end of "Project 1" 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 Mr. Khoan, lecturer at Ho Chi Minh City University of Technology and Education.

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, November, 2019

**Lecturer's evaluation**

*Ho Chi Minh City, November, 2019*

**Lecturer**

CONTENT

[Thanks 2](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719729)

[Lecturer’s evaluation 3](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719760)

[1. List of images 5](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719731)

[2. List of tables 5](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719732)

[3. Contents 6](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719733)

[3.1 Project Description 6](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719734)

[3.1.1 Objective 6](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719735)

[3.1.2 User benefits 6](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719736)

[3.1.3 Use case diagram 6](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719737)

[3.1.4 Use case description tables 7](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719738)

[3.2 Assignment of work 8](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719739)

[3.3 Design 9](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719747)

[3.3.1 Data access technology 9](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719748)

[3.3.2 Algorithm 9](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719748)

[3.3.2.1 Placing Datagridview location 9](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719748)

[3.3.2.2 Draw relationship ( arrow ) between tables 12](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719748)

[3.3.2.3 Check if Mouse Position is inside an arrow: 15](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719748)

[3.3.3 Class design 16](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719749)

[3.3.4 UI design 17](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719750)

[3.4 Installation and testing 18](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719751)

[3.5 Installation and testing 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719751)

[3.5.1 Evaluation of the project 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719751)

[3.5.1 For further implementation 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719751)

[3.5.1 Advantages 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719751)

[3.5.1 Disadvantages 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719751)

[4. Reference 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719759)

# **List of images**

[Image 1 - Use Case Diagram 6](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652034)

[Image 2 - Flowchart of placing datagridview location 10](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652035)

[Image 3 - Flowchart draw relationship between tables 12](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652035)

[Image 4 - Demo the table references 13](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652035)

Image 5 - Referenced table is above referencing table 13

[Image 6 - Referenced table is below referencing table 14](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652037)

[Image 7 - Referenced table is on the right of referencing table. 14](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652038)

[Image 8 - Referenced table is on the left of referencing table 14](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652040)

[Image 9 – Flowchart check mouse position is inside an arrow 15](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652035)

[Image 10 – Class Design 16](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652035)

[Image 11 - Design Login form 17](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652041)

[Image 12 – Design Generate Button and Panel 17](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652042)

[Image 13 - Design Table Information 17](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652053)

[Image 14 - Design Foreign Information Form 18](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

[Image 15 – Testing console 18](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

[Image 16 – Testing login 18](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

[Image 17 – Testing connect relation 19](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

[Image 18 – Testing move table 19](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

[Image 19 – Testing login server name 19](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

[Image 20 – Test add multiple database 19](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529652054)

**List of tables**

[Table 1 - Use case Login description 7](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719657)

[Table 2 - Use case Select description 7](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719658)

[Table 3 - Work assignment 8](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719659)

[Table 4 –Demo top left position 11](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719660)

[Table 5 – Demo assign next position 11](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719660)

[Table 6 – Demo assign position to the below pivot 11](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719660)

[Table 7 – Demo calculate items and Queue 12](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719660)

[Table 8 – Class design 16](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719660)

[Table 9 – UI design 17](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719661)

[Table 10 – Description of test case 16](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719662)

[Table 11 - Difficulties and Solutions 20](file:///C:\Users\Administrator\Desktop\%5bnhom2%5ddocument.docx#_Toc529719663)

# **Contents**

* 1. ***Project Description***

**3.1.1 Objective**

Design an application which helps user to generate database schema from Microsoft SQL Server in a visual graphic view.

Design user interface easy for using.

**3.1.2 User benefits**

You can quickly generate a Database into Database schema because it is a lightweight app.

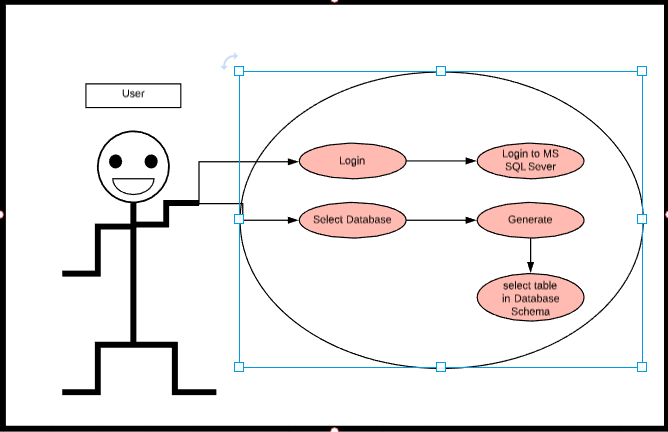
**3.1.3 Use case diagram**

Image 1 - Use Case Diagram

**3.1.4 Use case description tables**

Table 1 - Use case Login description

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Login | | |
| Description | Allows user to access the database management system | | |
| Actor | User | | |
| Business event | No. | Agent | System response |
|  | 1 | Access the application |  |
|  | 2 |  | Give login interface |
|  | 3 | Check the workstation information |  |
|  | 4 | Click “Login” button |  |
| Preconditions | Have installed DBMS Microsoft SQL Server | | |
| Condition affecting termination outcome | When connection succeeded and User click “Sign in” button  When connection failed and show error message | | |
|  |  | | |

Table 2 - Use case Select description

|  |  |  |  |
| --- | --- | --- | --- |
| Use case name | Generate | | |
| Description | Custom generate script by the following request from the user | | |
| Actor | User | | |
| Business event | No. | Agent | System response |
|  | 1 |  | Give Generate database name scrollbox |
|  | 2 |  | Load MetaData form DBMS to treeview – all Databases name |
|  | 3 | Select 1 database name to generate database schema |  |
|  | 4 | Click “Generate” Button to get the schema that user expect |  |
|  | 5 |  | Load schema shown in a Panel, User can see the results |
| Preconditions | Successfully connect to DBMS and show watch Database schema | | |

# 

# ***3.2*** ***Assignment of work***

Table 3 – Work assignment

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Student Name | General description of the work array students perform in the project. | Estimated percentage contribution |
| 1 | Trương Minh Quân Phạm Vũ Thiện | * Read how to use retrieve database schema. * Code test database schema information. * Design class * Language c# | Quân – 50% Thiện – 50% |
| 2 | Trương Minh Quân | * Create classes according to class design. * Assign properties’ value of all classes with data collected from GetSchema. * Print out properties’ value of all classes to console window using get() or ToString() method. | 100% |
| 3 | Trương Minh Quân Phạm Vũ Thiện | * Create login interface. * Fill data from a table object to Datagridview and view it on winform * Fill data from all table objects to Datagridview, view it on win form and adjust their position. | Quân – 65% Thiện – 35% |
| 4 | Trương Minh Quân Phạm Vũ Thiện | * Add features: * View detail information of a table by double click on datagridview using GetSchemaTable * Draw a line connect two tables which show constraint foreign key. | Quân – 70% Thiện 30% |
| 5 | Trương Minh Quân Phạm Vũ Thiện | * Add features: * Feature that allow user to hold and drag a table | Quân-50% Thiện-50% |
| 6 | Tương Minh Quân | * Add features: * Show Foreign Key By double click on arrow | 100% |
| 7 | Phạm Vũ Thiện | * Debug, test and optimize * Push on github | 100% |
| 8 | Trương Minh Quân Phạm Vũ Thiện | * Write report | Quân-50% Thiện-50% |

# ***3.3 Design***

***3.3.1 Data access technology*:**

We use ADO.NET to interface with Microsoft SQL Server, associate with databases, executing inquiries,…

ADO.NET is an information get to innovation from the Microsoft .NET Framework that gives correspondence among social and non-social frameworks through a typical arrangement of parts.

So as to utilize ADO.NET, we'll have to import two namespaces:

using System.Data;

using System.Data.SqlClient;

So as to build up an association with SQL Server, we utilize the SqlConnection object

SqlConnection conn= new SqlConnection(ConnectionString)

Opening an association is significant:

cn.Open();

this.cn.Close();

Association string normally taking this structure:

("Data Source={0};Initial Catalog={1};User ID={2};Password={3}",data\_source,this.database\_name, username,password)

So as to execute a SQL inquiry, or a put away strategy on the server, we utilize the SqlCommand object:

string selectQuery = "select ";

SqlCommand comm = new SqlCommand(query, conn);

***3.3.2 Algorithm*:**

**3.3.1.1 Placing Datagridview location:**

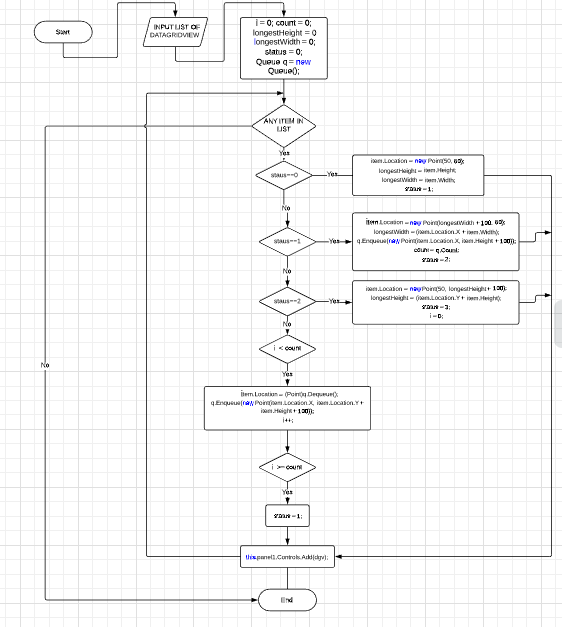
Purpose:

* + After filling in information of table objects to all Datagridview in List<Datagridview>, we need to assign location for all Datagridview.
  + Main goal of the algorithm is to use as much space of panel as possible.

Explanation:

* + Direction of placing Datagirdview from Top left to Down bottom
  + Flowchart:

Image 2 - Flowchart of placing datagridview location



* + Brief description:
* First assign the beginning position or pivot position which is the Top left position(X0, Y0) - status = 0

Table 4 – Demo top left position

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X0 | X1 | X2 | X3 | X4 |
| Y0 |  |  |  |  |  |
| Y1 |  |  |  |  |  |
| Y2 |  |  |  |  |  |
| Y3 |  |  |  |  |  |

* Second assign next position to the right of pivot position horizontally (X1, Y0) - status = 1

Table 5 – Demo assign next position

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X0 | X1 | X2 | X3 | X4 |
| Y0 |  |  |  |  |  |
| Y1 |  |  |  |  |  |
| Y2 |  |  |  |  |  |
| Y3 |  |  |  |  |  |

* Note: at this time, we calculate position right below the position we just assign (X1, Y1), and put it to Queue= {(X1, Y1)} then calculate the number of items in the Queue count = 1.
* Next assign position to the below pivot position vertically (X0, Y0=1) - status = 2

Table 6 – Demo assign position to the below pivot

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X0 | X1 | X2 | X3 | X4 |
| Y0 |  |  |  |  |  |
| Y1 |  |  |  |  |  |
| Y2 |  |  |  |  |  |
| Y3 |  |  |  |  |  |

* Next we can see that there is space between horizontal axis and vertical axis. Base on the count variable we have already calculate and all items in the Queue, we can fill this gap.

Table 7 – Demo calculate items and Queue

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | X0 | X1 | X2 | X3 | X4 |
| Y0 |  |  |  |  |  |
| Y1 |  |  |  |  |  |
| Y2 |  |  |  |  |  |
| Y3 |  |  |  |  |  |

* Note: at this time, we calculate position right below the position we just assign (X1, Y2), and put it to Queue= {(X1, Y2)}.
* This process will continue until there are no DataGridView to assign location.

**3.3.2.2 Draw relationship ( arrow ) between tables:**

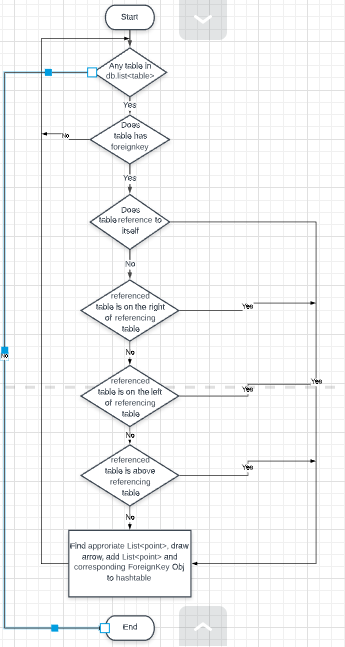
Purpose:

* + Illustrates relationship between referenced table and referencing table by drawing arrows.
  + The arrows must be clearly visible.

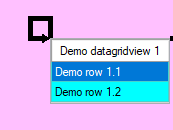
Explanation:

* Flowchart:

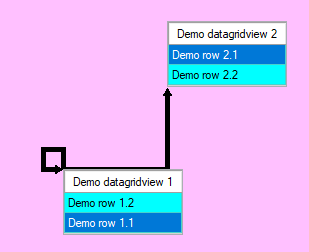
Image 3 - Flowchart draw relationship between tables



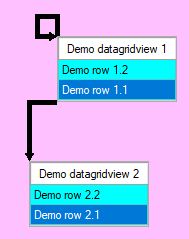
* Brief description: Arrows will be drawn depending on cases:
* Case 1: the table references to itself

Image 4 - Demo the table references

* Case 2: referenced table intersects referencing table vertically
* Case 2.1: referenced table is above referencing table:

Image 5 – Referenced table is above referencing table

* Case 2.2: referenced table is below referencing table

Image 6 - Referenced table is below referencing table

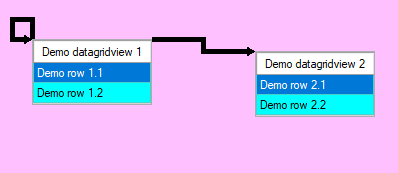
* Case 3: referenced table does not intersect referencing table vertically
* Case 3.1: referenced table is on the right of referencing table

Image 7 - Referenced table is on the right of referencing table

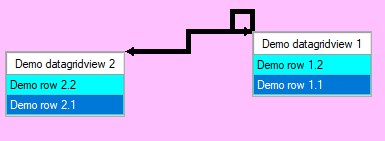
Case 3.2: referenced table is on the left of referencing table

Image 8 - Referenced table is on the left of referencing table

* Note: After draw an arrow hashtable will be added with an item which key is List<Points> that we use to draw and value is Foreign Key object related to referencing table object.

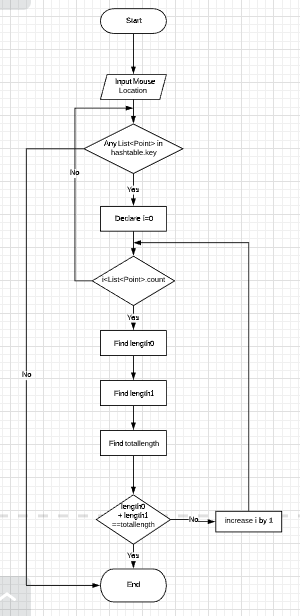
**3.3.2.3 Check if Mouse Position is inside an arrow:**

Purpose: to find out corresponding Foreign Key object when users double click on arrow:

Explanation:

* Flowchart:

Image 9 - Flowchart check mouse position is inside an arrow



* Brief description:
* An arrow is basically a list of points and a point and another following one in the list create section of an arrow.
* In order to find whether Mouse cursor is in a section or not we calculate the distance from mouse cursor to one point (length0) and the distance from mouse cursor to another following one (length1) and total length of that two points (totalLength). If length0 + length1 == totalLength Mouse cursor is inside that arrow.

***3.3.3 Class design*:**

Image 10 – Class Design

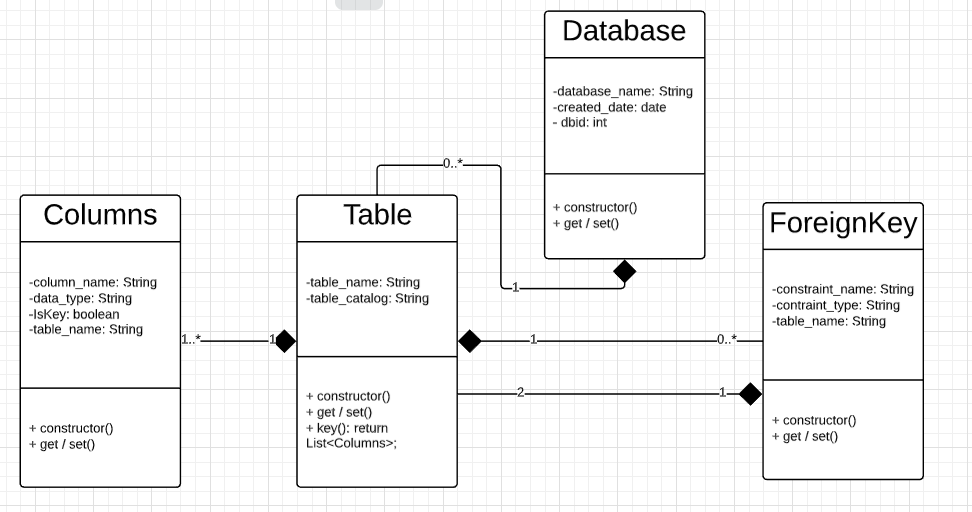


Table 8 – Class design

|  |  |  |
| --- | --- | --- |
| Number | Class name Inheritance or lead export from (specify name base class) – if we have | Purpose |
| 1 | Columns | Get multiple of columns and key values. |
| 2 | Database | Get id of table, database name and create date of that table. |
| 3 | ForeignKey | Each table will have foreign key and constraint name |
| 4 | Tables | This is used get tables, comlumns, foreignkeys. |

***3.3.4 UI design:***

Table 9 – UI design

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Screen/Dialog | Purpose | Describe the reason about the design |
| 1 | Image 11 – Design Login form | Login interface: Let user to login in the program | We need to design the login in other user can access into, if database have account the user can access throughout program |
| 2 | Image 12 – Design Generate Button and Panel | Generate Form: Use to read database schema | We use panel as the paper where we can read database schema |
| 3 | Image 13 – Design Table Information | Table information form: Knowing about the information in the table | We design this because whether or not user need to watch what was in table |
| 4 | Image 14 – Design Foreign Information Form | Foreign Key  Information Form:  We can now the relation ship of the foreign key | We use to watch the foreign key information as easy way |

# 

# ***3.4 Installation and testing***

Table 10 - Description of test cases

|  |  |  |  |
| --- | --- | --- | --- |
| Number | Case 1 | Purpose | Briefly explain the decision to make  situations and input data |
| 1 | Test get foregin keys for table using console. We use HR database.  Result: Can get the foreign key in each talbe  Image 15 – Testing console | We need to find foreign key for each table if we want to connect between two table | Because we need the figure out that table have a foreign key or not so we need to test if table have keys. We used HR database content a lot of table which have a lot of connection |
| 2 | Test normal login and create login form  Result: Successfully with normal login  Image 16 -Testing login | We need an login interface so I create an login form | We don’t have any input data here |
| 3 | Test connect relation and move table  Image 17 – Testing connect relation  Image 18 – Testing move table | This fate use to get connect between two tables and move table around the panel | I used HR database again because there have a lot of connection between table, more over we can see that I can move the table around the panel |
| 4 | Final test to add Sever name & DB Nam  Image 19 - Testing login server name    Image 20 – Test add multiple database | Final test to check whenever the project is bugged or not. | I use “Thien” as sever name and HR as database name to login to program. |

# ***3.5 Conclusion***

**3.5.1 Evaluation of the project**

Project assessment: this project has achieved about 90% listed goals.

Table 11 – Difficulties and Solutions

|  |  |
| --- | --- |
| Difficulties | Solution |
| Finding relationship between drawing arrow and ForeignKey Object | Using HashTable |
| Placing Datagridview which uses as much space of panel as possible | Apply Queues |
| Draw clear arrows | Listing all cases when placing Datagridview |

**3.5.2 For further implementation**

* Adding primary key icon
* Arrows have to illustrate referencing column and referenced column precisely

**3.5.3 Advantages**

* Show clear diagram according to database.
* View detail information of table.
* View Foreign Key information

**3.5.4 Disadvantages**

* Handling fail login slowly.
* Program starts to run slowly when database has about 20 tables.

1. **Reference**

* <https://docs.microsoft.com/en-us/dotnet/framework/data/adonet/getschema-and-schema-collections>
* <https://www.youtube.com/watch?v=3eIzmcTe0mg>
* <https://social.msdn.microsoft.com/Forums/vstudio/en-US/7efda165-f2da-46d2-8c25-f7b093d9e0d3/how-to-get-sql-server-instance-name?forum=csharpgeneral>
* <https://stackoverflow.com/questions/10550541/how-to-get-database-name-from-connection-string-using-sqlconnectionstringbuilder>