Python Advanced Project

Report

INTERNET SERVICE INFORMATION MANAGEMENT SYSTEM

Group 12

May, 2021

USTH

VIETNAM FRANCE UNIVERSITY

Author:

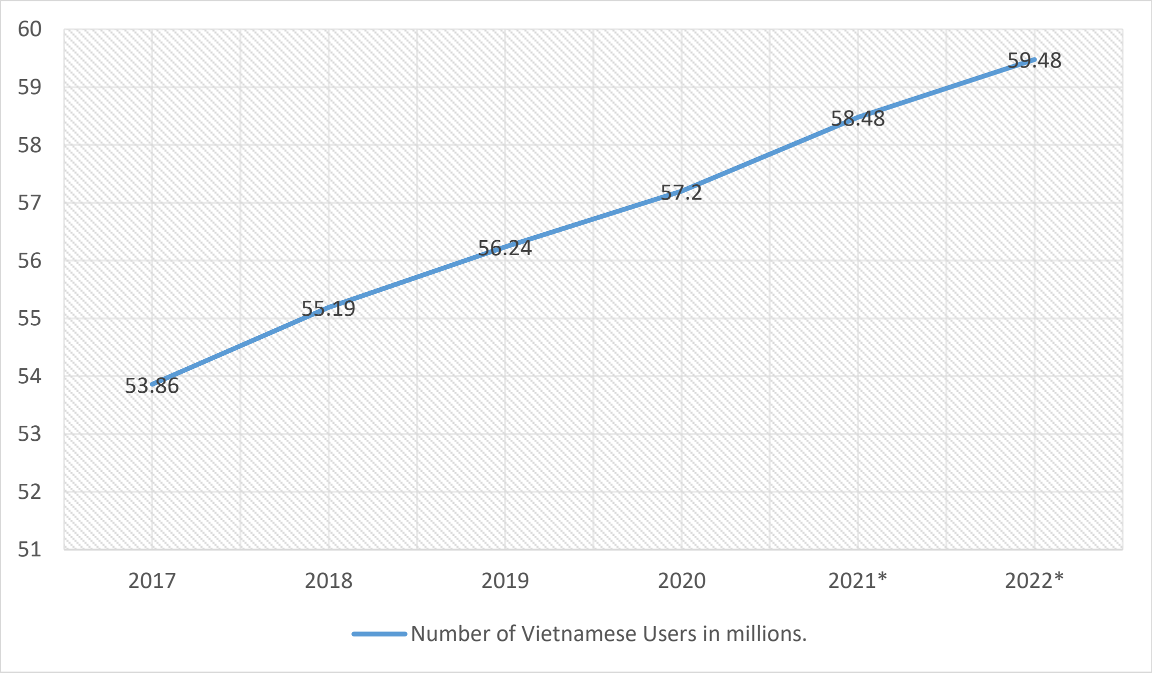
1. Nguyen Viet Nhan – BI10-132
2. Do Dinh Phuc – BI10-137
3. Tran Trung Hieu – BI10-058
4. Phi Nguyen Hai Minh – BI10-120
5. Do Thanh Hieu – BI10-059

Table of Contents:

1. Introduction:

1.1 Overview:

- These days, the demands of healthcare are increasing swiftly. This leads to a problem that hospitals have to be under lot of pressure in managing medical records, so that we need a better database management and efficiency.
- In our project, we are going to provide a database management which is more simple, and more efficiency to improve the ways hospitals manage their medical records nowadays. We have developed some kind of functions, for example doctors can view details about patients, medicine is prescribed,... and patients can also look for doctors' informations in a most comfortable way.



* : Estimated statistics.

- Let's have a look at the chart above. This is the statistics for the number of Vietnamese Users (in millions unit), and it can be clearly seen that there is an increased for approximately 110% each year. With a wider look, according to vnetwork.vn, 4.72 billions is the amount of Internet Users all over the world, which could be equivalent to about 60.1% of the World's Population. And this is the origin of the problem!

1.2 Why We Need This Program:

- HydraISP, Splynx or Cynox, which are "friendly" called Billing Softwares, are currently the most popular for Internet Services Providers in order to manage the service they are giving to customers. Both of them have the same disadvantages:

- They are extremely expensive! It is a huge obstruction for providers with small & micro business scale.
- They are difficult to use for the first-time staffs. Companies using these softwares have to spend a lot of time for training their staffs to get use to the softwares.
- They are still overload in high – user – access time, although companies have to pay lots of money for authorities!

- Our team cannot solve all these problems for a short time period, so we decide to solve the economic problem first! The reason why we chose this problem is the trend of users. Users, this day, seem to give up with big Internet Service Providers (ISPs) and choose the small providers, which currently have better services with cheaper prices. What the small providers need is a service management software which is easy to use, reliable, and cheap!
- Understanding this situation, we decided to build a half – professional service management software that is suitable for the Small & Micro Internet Service Providers. It is easy to use, Reliable and of course, a cheap price!

1.3 What This Program Do?:

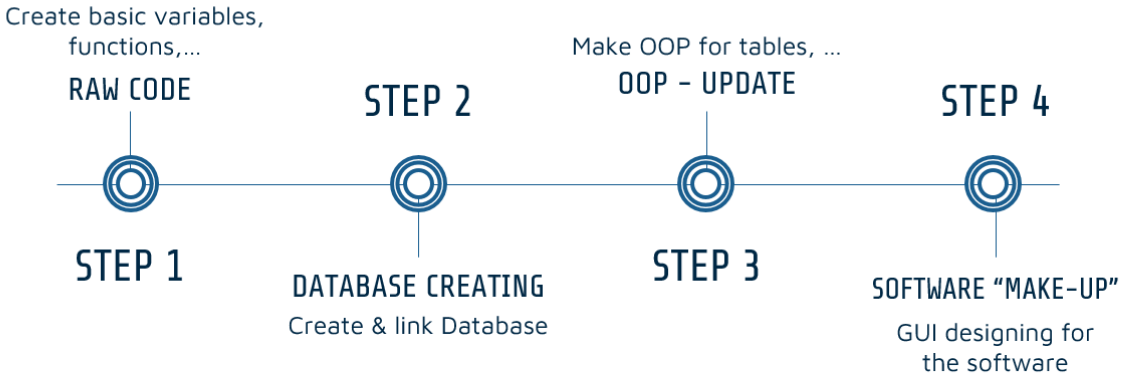
This program will be run by the network staff to stored the information of the Customer, Data plan and calculate, export the Bill to Customer. Also we can Add, Update, Remove, Search, and Insert Customer or Data plan or Bill to easier to managed.

1.4 Software Used To Develop Program:

- OS : Windows 10, MacOS
- Pycharm
- MySQL Workbench
- MySQL Connector : Connector/Python
- MySQL Server

2. Our Process(how we create it ?):

2.1 Overview Process:



2.2 Structures:

❖ **Functions ideas:**

- We want to create a program in which the input of a customer, dataplan or bill must be defined by one Object (stored in OOP class). Therefore, we list & define Objects and required attributes, then make a Object py class file which have getter and setter function.
- Next, we create & connect Python with MySQL Database and affiliated with Object,fundamental manipulation with MySQL : Create, Read, Update and Delete (CRUD), create function to calculate the total amount for Customers, create function to issue bills for Customers, and last is create functions to show information.
- Code in Python and some functions are linked with SQL syntax and server.
- In CURD Functions, we used "if/else" for checking value of variables if they were empty or not.

Py class:

Main.py:

- Fist idea we want to separate 3 file :
- Input.py: storing input function :get the user input.
  - Add_Customer(), Add_Dataplan(), Add_Bill() : to add a new customer (dataplan, bill) with their information in database .
  - Update_Customer(), Update_Dataplan(), Update_Bill() : to edit and update information of a customer(dataplan,bill) in database.
- Output.py storing output function which interact with user input and show informations (information of customer,dataplan, bill and table, get bill fuction)
  - Search_Customer(), Search_Dataplan(), Search _Bill() : to search a customer (dataplan, bill) and show all informations of it in database.
  - Remove_Customer(), Remove_Dataplan(), Remove _Bill() : to remove a customer (dataplan, bill) which we added in database
  - Calculate_Total_Amount(): Check the Customer ID and Dataplan, get the Duration and Price values of this entity to calculate the total price using the syntax Total_Amount = Duration * Price
  - Write_Customer_Input() : create a file txt. Then get all informations of customer table, bill table, dataplan table in database and write its into a file txt. And we'll print file.txt for customer (get bill)
  - Insert_Table_Customer(),Insert_Table_dataPlan(), Insert_Table_Bill(): to get all information of 3 table customer, table bill, table dataplan and return it (we use for show tables in GUI)
- Main.py: storing GUI code and manage the running of the program. (We ll talk about GUI later)
- Because we have a problem name circular import so we decided to combine all of them into one file Main.py

code_create_table_mysql.py:

- Create tables in database function
- CreateDataBaseTableCustomer(), CreateDataBaseTableDataplan(), CreateDataBaseTableBill() : create 3 table and attributes of it (define the data type and data size) in database.

Class "Customer" ("Customer.py") :

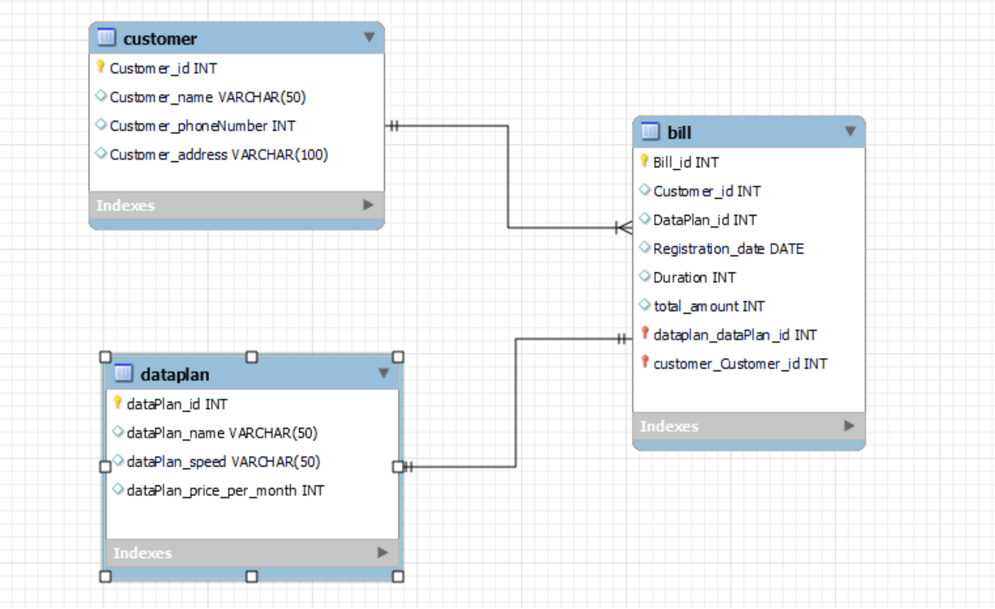
- init(): To store the Customer into the individual objects.
- Function get() will return the value of Customer table that we needed to do anything else.
- Function set() will change the old Customer value to new value we entered
- All values of Customer objects are set to Private

- Class “Dataplan” (“Dataplan.py”) :
  - init(): To store the Dataplan into the individual objects.
  - Function get() will return the value of DataPlan table that we needed to do anything else.
  - Function set() will change the old DataPlan value to new value we entered
  - All values of Dataplan objects are set to Private

- Class “Dataplan” (“Bill.py”):
  - init(): To store the Bill into the individual objects.
  - Function get() will return the value of Bill table that we needed to do anything else.
  - Function set() will change the old Bill value to new value we entered
  - All values of bill objects are set to Private

- **Modules:**
  - tkinter.messagebox : Access to standard Tk Dialog Boxes.
  - tkinter.ttk : Provide access to the Tk themed widget set.
  - tkinter import * : Provide access basic syntaxs for create GUI (title, button, label,…)
  - mysql.connector: Database driver to process data in MySQL from Python
- **Packages:**
  - “Domain” :
    - Storing 3 classes files (“Customer.py”, “Bill.py”, “Dataplan.py”).
    - “___init__.py” : Just a empty file, appear when creating pakage.

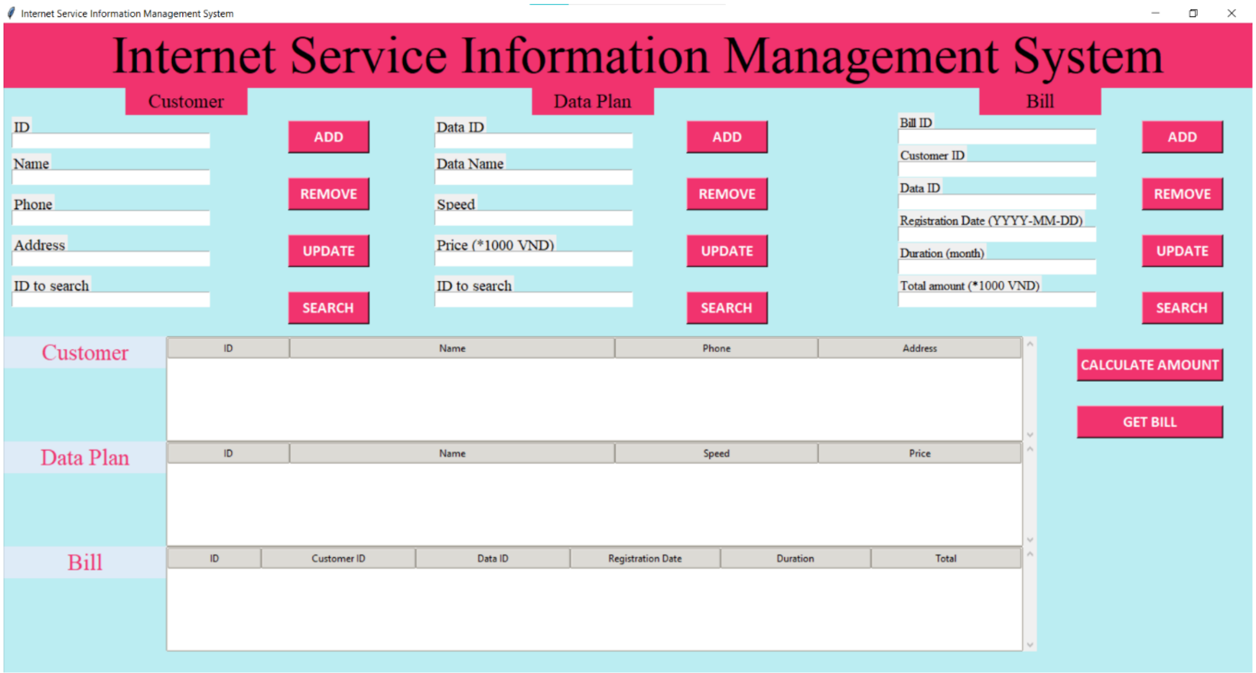
2.3 Database Diagram :



- In this **Diagram,** we have a three entities with their Attributes , beside attributes, we can see their data type (like INT, varchar,…) and data size (like 50 , 100).
- the Relationship:
  - the Relationship between customer and bill is one – many, because the one customer can have many bills.
  - And one – one is the Relationship between dataplan and bill because the one dataplan only can appear in one bill.

- In Bill table, dataplan_dataPlan_id and customer_Customer_id appear when we draw the relationship in MySQL workbench, they are the same Customer_id and DataPlan_id we created but have foreign keys relationship (we will update Customer_id and DataPlan_id later).

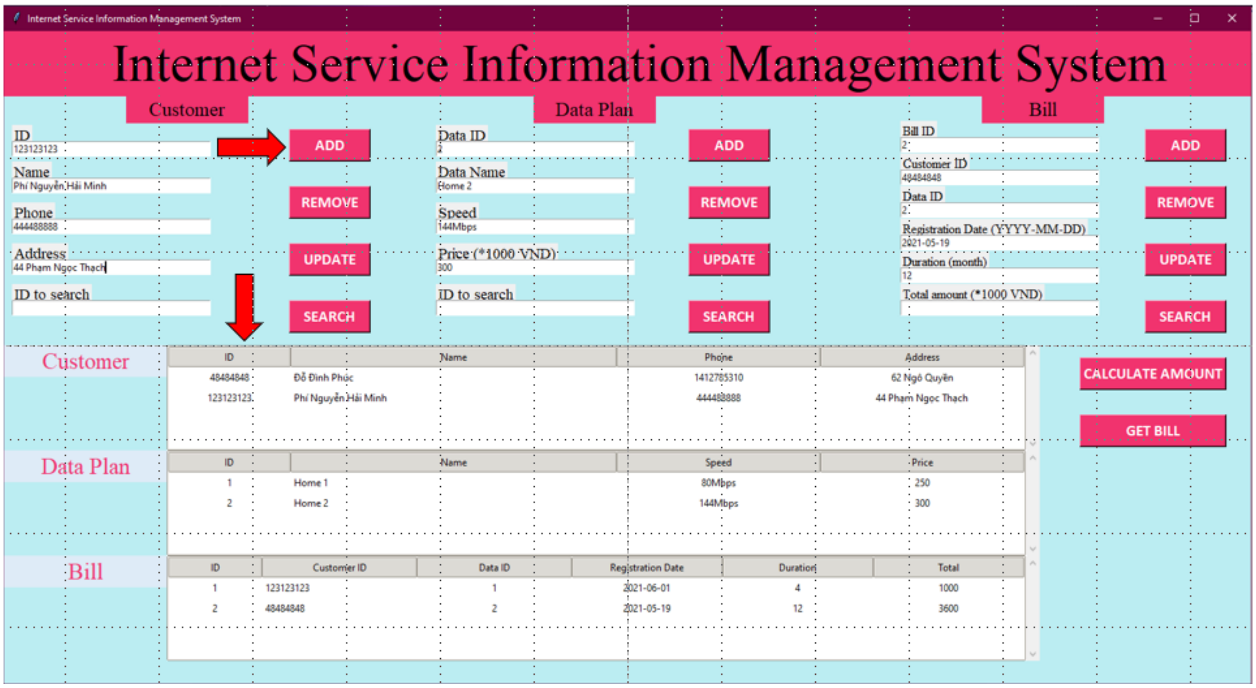
2.4 Graphic User Interface:



- We create 3 sectors for entering information of “Customer”, “Data Plan” and “Bill”.
- Inside each sector is the spaces where you fill the suitable information.
- We add the buttons that have the functions like adding, removing, updating and searching on the right of each sector.
- In “Bill” sector we have two more buttons:
    - “calculate amount”: to calculate the total price of a customer depending on the duration of data plan that they choose.
    - “get bill”: to get bill.
- 3 boards are where show the information filled above.

3. DEMO:

3.1 *ADD* button: Adding new Customer/Data Plan or Bill into the Customer board, Data Plan board or Bill board.



ID
123123123

Name
Phí Nguyễn Hải Minh

Phone
444488888

Address
44 Phạm Ngọc Thạch

ID to search

Data ID
2

Data Name
Home 2

Speed
144Mbps

Price (*1000 VND)
300

ID to search

Bill ID
2

Customer ID
48484848

Data ID
2

Registration Date (YYYY-MM-DD)
2021-05-19

Duration (month)
12

Total amount (*1000 VND)

Customer

ID	Name	Phone	Address
48484848	Đỗ Đình Phúc	1412785310	62 Ngô Quyền
123123123	Phí Nguyễn Hải Minh	444488888	44 Phạm Ngọc Thạch

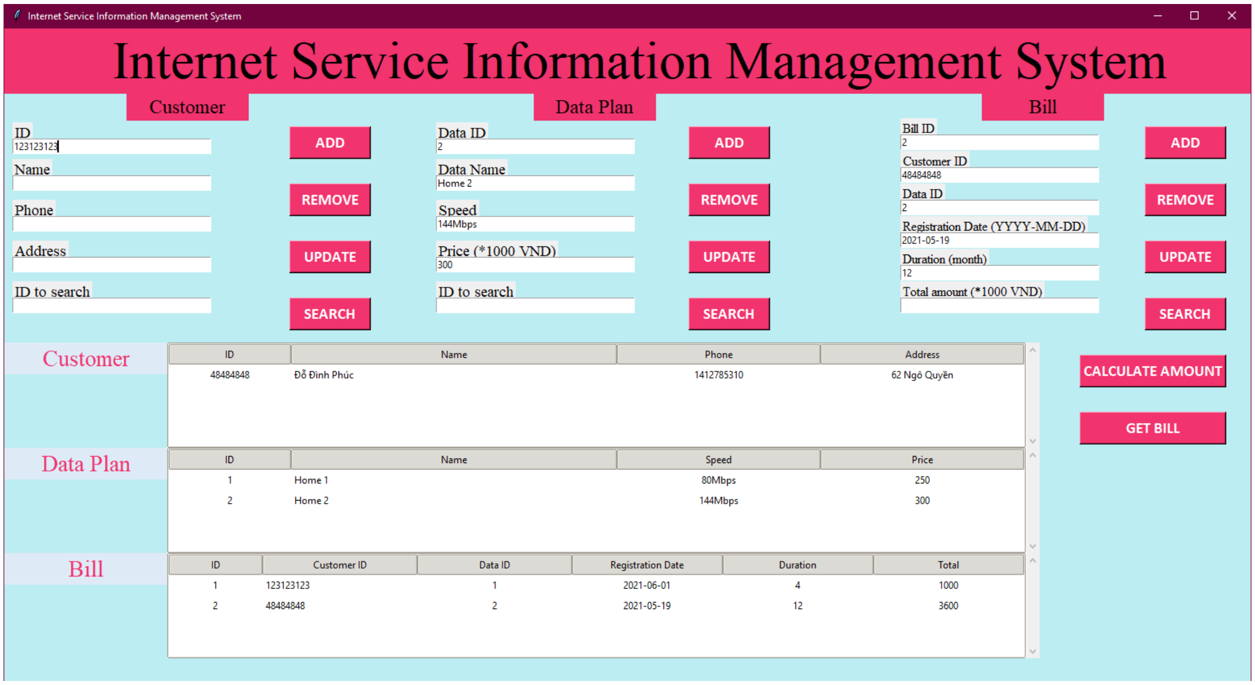
Data Plan

ID	Name	Speed	Price
1	Home 1	80Mbps	250
2	Home 2	144Mbps	300

Bill

ID	Customer ID	Data ID	Registration Date	Duration	Total
1	123123123	1	2021-06-01	4	1000
2	48484848	2	2021-05-19	12	3600

3.2 *REMOVE* button: Removing a Customer, Data Plan or Bill by using their IDs (The Customer Phí Nguyễn Hải Minh who has ID: 123123123 is already removed)



ID
123123123

Name

Phone

Address

ID to search

Data ID
2

Data Name
Home 2

Speed
144Mbps

Price (*1000 VND)
300

ID to search

Bill ID
2

Customer ID
48484848

Data ID
2

Registration Date (YYYY-MM-DD)
2021-05-19

Duration (month)
12

Total amount (*1000 VND)

Customer

ID	Name	Phone	Address
48484848	Đỗ Đình Phúc	1412785310	62 Ngô Quyền

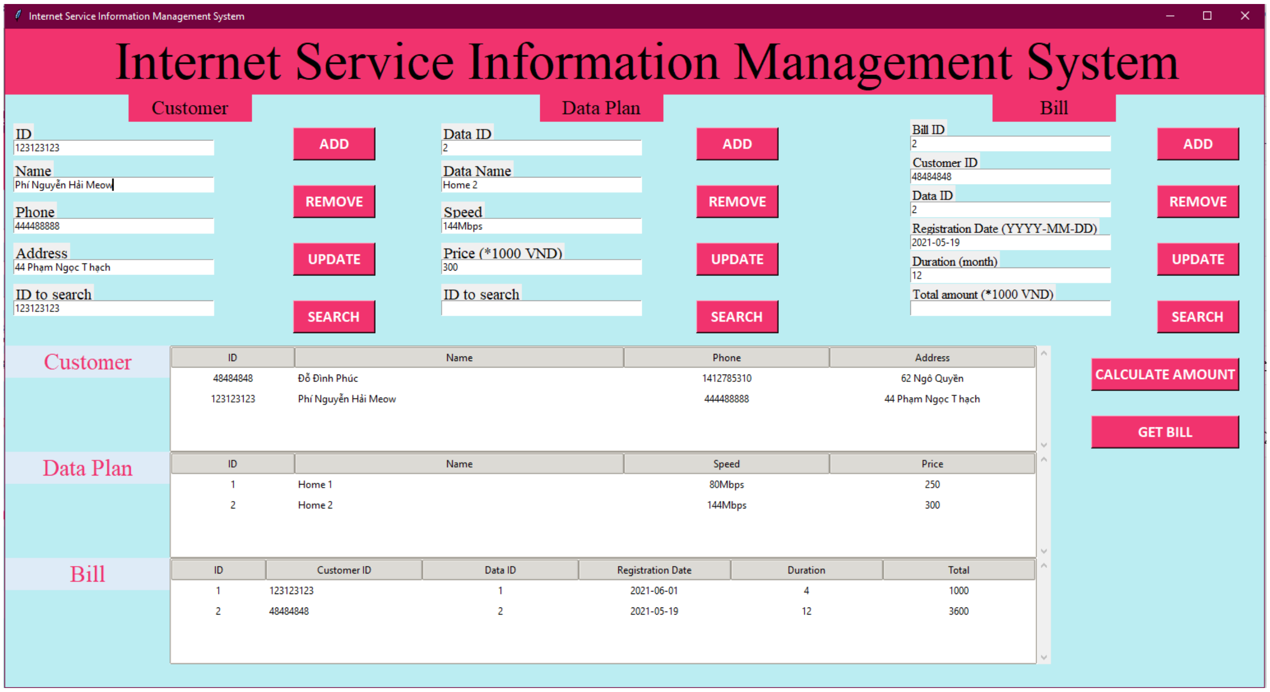
Data Plan

ID	Name	Speed	Price
1	Home 1	80Mbps	250
2	Home 2	144Mbps	300

Bill

ID	Customer ID	Data ID	Registration Date	Duration	Total
1	123123123	1	2021-06-01	4	1000
2	48484848	2	2021-05-19	12	3600

3.3 *UPDATE* button: Updating the new information for the Customer, Data Plan or Bill (Example: Name of customer Phí Nguyễn Hải Minh is changed to Phí Nguyễn Hải Meow)



Internet Service Information Management System

ID
123123123
Name
Phí Nguyễn Hải Meow
Phone
444488888
Address
44 Phạm Ngọc T hạch
ID to search
123123123

Data ID
2
Data Name
Home 2
Speed
144Mbps
Price (*1000 VND)
300
ID to search

Bill ID
2
Customer ID
48484848
Data ID
2
Registration Date (YYYY-MM-DD)
2021-05-19
Duration (month)
12
Total amount (*1000 VND)

Customer

ID	Name	Phone	Address
48484848	Đỗ Đình Phúc	1412785310	62 Ngô Quyền
123123123	Phí Nguyễn Hải Meow	444488888	44 Phạm Ngọc T hạch

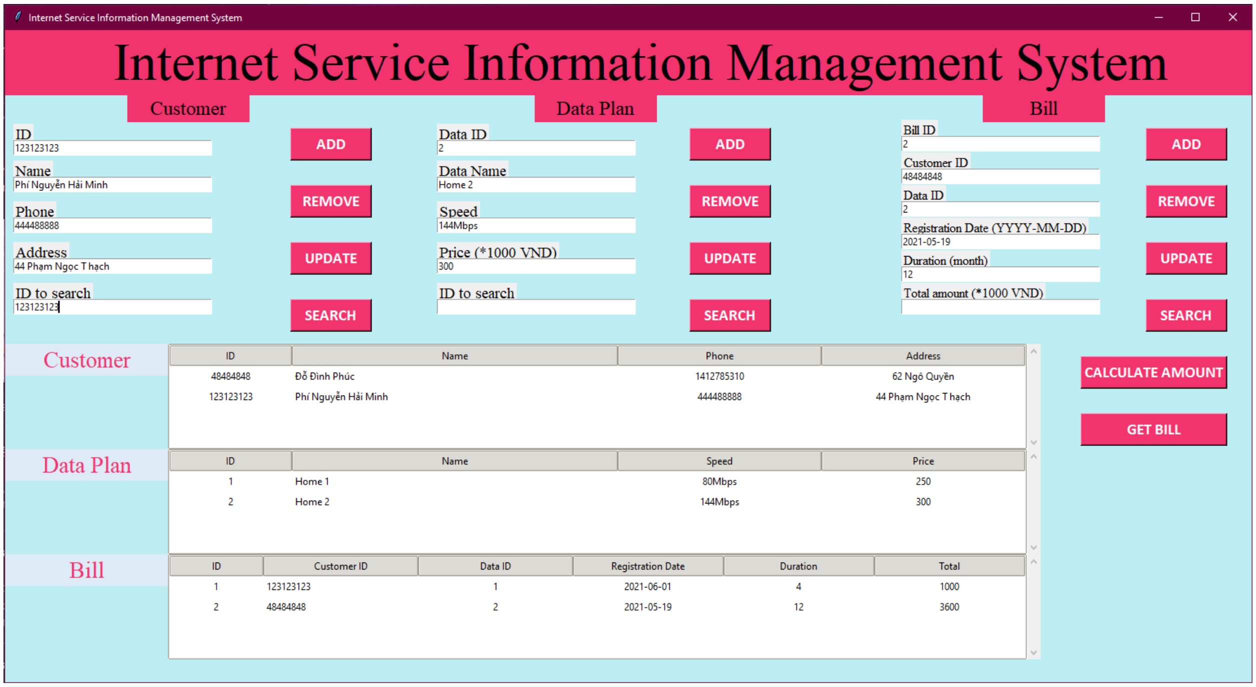
Data Plan

ID	Name	Speed	Price
1	Home 1	80Mbps	250
2	Home 2	144Mbps	300

Bill

ID	Customer ID	Data ID	Registration Date	Duration	Total
1	123123123	1	2021-06-01	4	1000
2	48484848	2	2021-05-19	12	3600

3.4 *SEARCH* button: Searching a specific Customer, Data Plan or Bill by using their IDs and “ID to search” (the white spaces like ID, Name, Phone and Adress are filled the information of the Customer having ID:123123123)



Internet Service Information Management System

ID
123123123
Name
Phí Nguyễn Hải Minh
Phone
444488888
Address
44 Phạm Ngọc T hạch
ID to search
123123123

Data ID
2
Data Name
Home 2
Speed
144Mbps
Price (*1000 VND)
300
ID to search

Bill ID
2
Customer ID
48484848
Data ID
2
Registration Date (YYYY-MM-DD)
2021-05-19
Duration (month)
12
Total amount (*1000 VND)

Customer

ID	Name	Phone	Address
48484848	Đỗ Đình Phúc	1412785310	62 Ngô Quyền
123123123	Phí Nguyễn Hải Minh	444488888	44 Phạm Ngọc T hạch

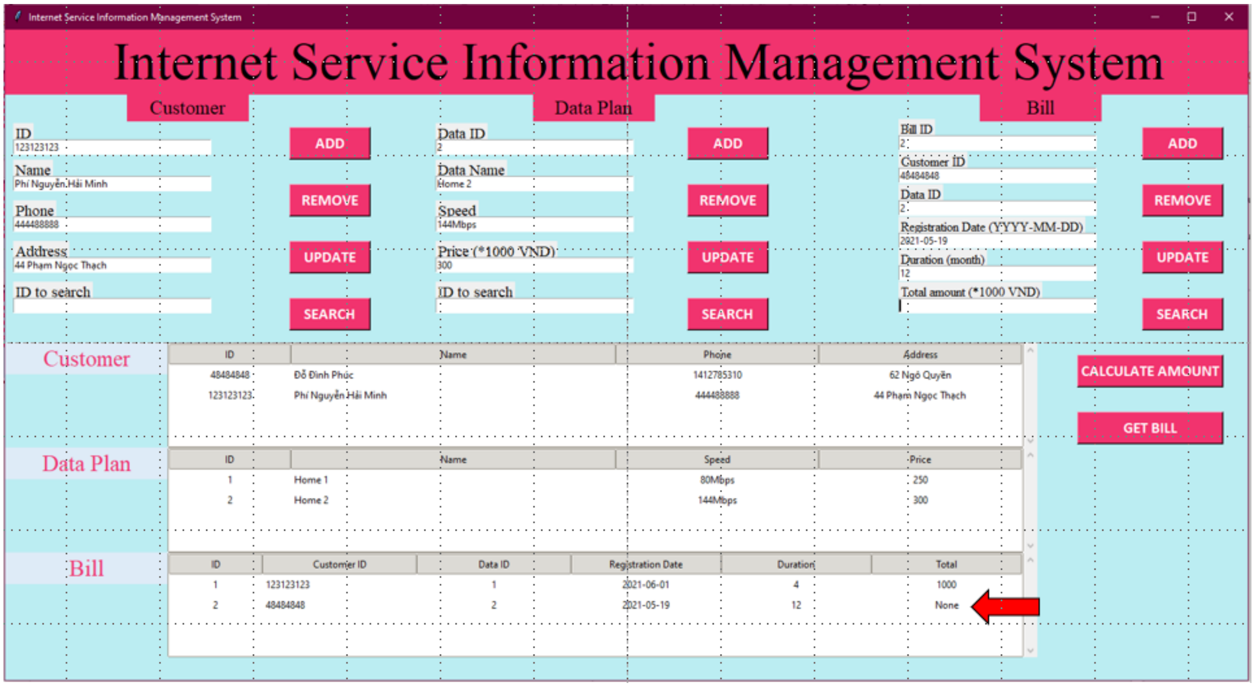
Data Plan

ID	Name	Speed	Price
1	Home 1	80Mbps	250
2	Home 2	144Mbps	300

Bill

ID	Customer ID	Data ID	Registration Date	Duration	Total
1	123123123	1	2021-06-01	4	1000
2	48484848	2	2021-05-19	12	3600

3.5 *CALCULATE AMOUNT* button: Calculating the total price (Price*Duration) that Customer must pay.



ID: 123123123
Name: Phí Nguyễn Hải Minh
Phone: 444488888
Address: 44 Phạm Ngọc Thạch
ID to search:

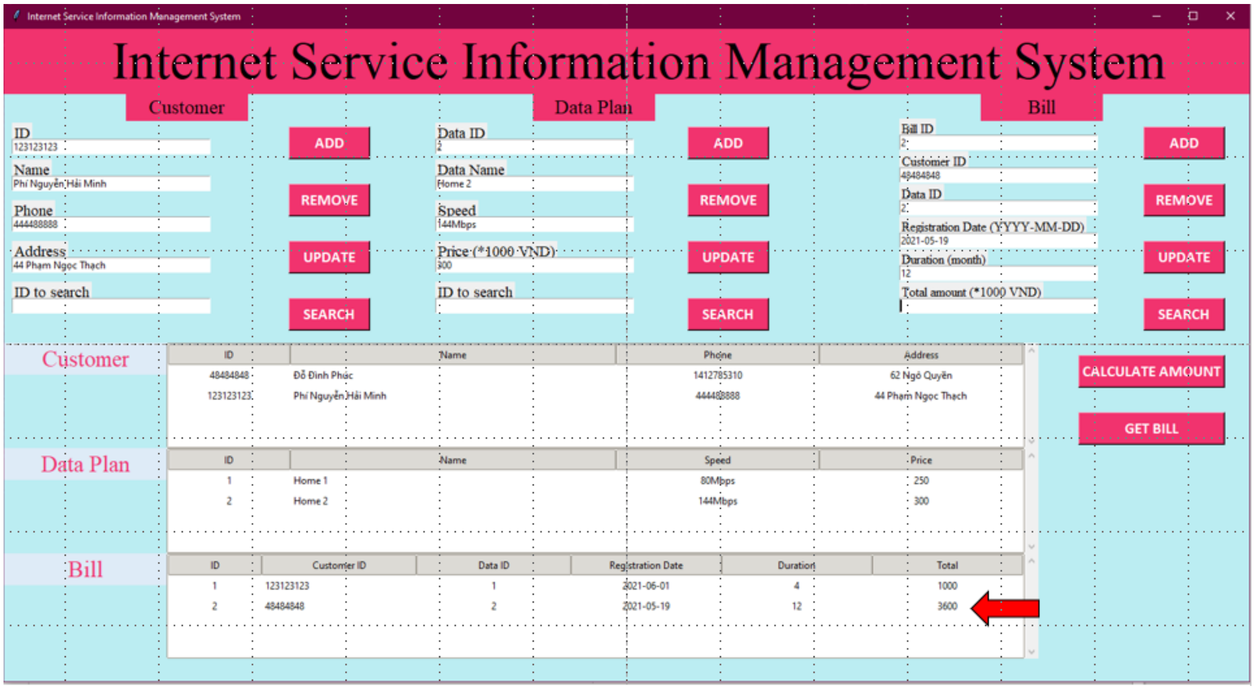
Data ID: 2
Data Name: Home 2
Speed: 144Mbps
Price (*1000 VND): 300
ID to search:

Bill ID: 2
Customer ID: 48484848
Data ID: 2
Registration Date (YYYY-MM-DD): 2021-05-19
Duration (month): 12
Total amount (*1000 VND):

ID	Name	Phone	Address
48484848	Đỗ Đình Phúc	1412785310	62 Ngô Quyền
123123123	Phí Nguyễn Hải Minh	444488888	44 Phạm Ngọc Thạch

ID	Name	Speed	Price
1	Home 1	80Mbps	250
2	Home 2	144Mbps	300

ID	Customer ID	Data ID	Registration Date	Duration	Total
1	123123123	1	2021-06-01	4	1000
2	48484848	2	2021-05-19	12	None



ID: 123123123
Name: Phí Nguyễn Hải Minh
Phone: 444488888
Address: 44 Phạm Ngọc Thạch
ID to search:

Data ID: 2
Data Name: Home 2
Speed: 144Mbps
Price (*1000 VND): 300
ID to search:

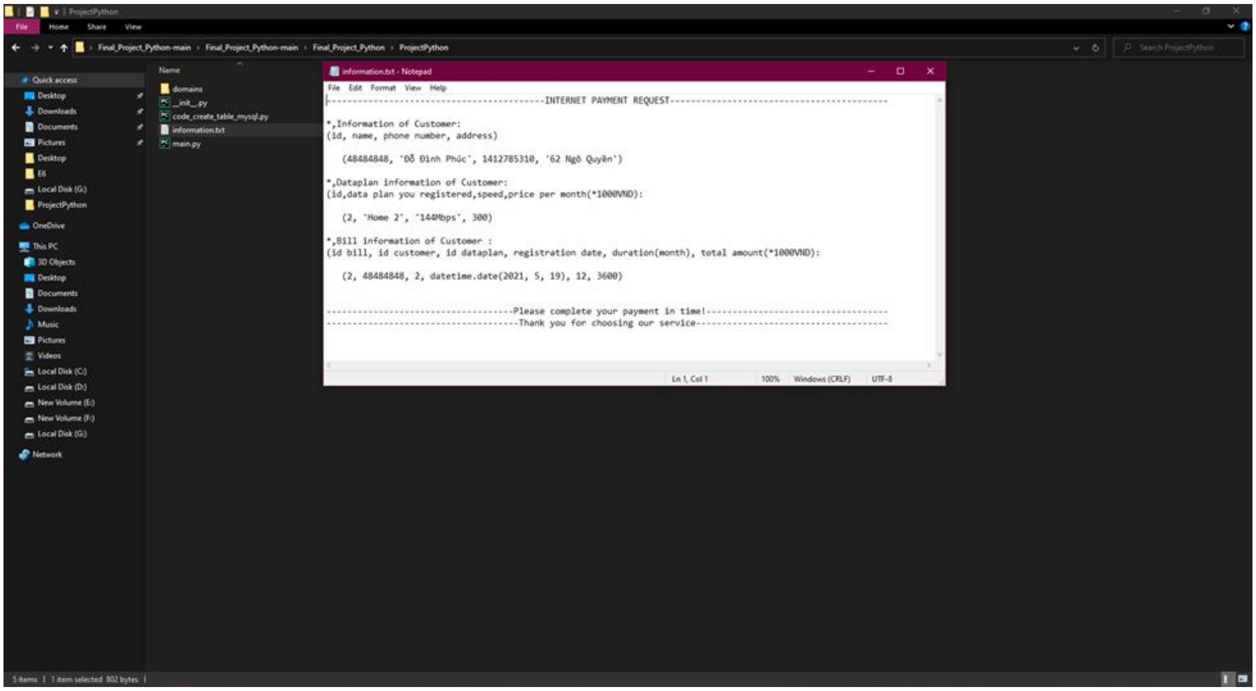
Bill ID: 2
Customer ID: 48484848
Data ID: 2
Registration Date (YYYY-MM-DD): 2021-05-19
Duration (month): 12
Total amount (*1000 VND):

ID	Name	Phone	Address
48484848	Đỗ Đình Phúc	1412785310	62 Ngô Quyền
123123123	Phí Nguyễn Hải Minh	444488888	44 Phạm Ngọc Thạch

ID	Name	Speed	Price
1	Home 1	80Mbps	250
2	Home 2	144Mbps	300

ID	Customer ID	Data ID	Registration Date	Duration	Total
1	123123123	1	2021-06-01	4	1000
2	48484848	2	2021-05-19	12	3600

3.6 *GET BILL* button: Exporting the Bill to .txt file


information.txt - Notepad
File Edit Format View Help
Final_Project_Python-main › Final_Project_Python-main › Final_Project_Python › ProjectPython
domains
__init__.py
code_create_table_mysql.py
information.txt
main.py
Ln 1, Col 1
100%
Windows (CRLF)
UTF-8

```
-----------------------------------------INTERNET PAYMENT REQUEST-----------------------------------------

*,Information of Customer:
(id, name, phone number, address)

   (48484848, 'Đỗ Đình Phúc', 1412785310, '62 Ngô Quyền')

*,Dataplan information of Customer:
(id,data plan you registered,speed,price per month(*1000VND):

   (2, 'Home 2', '144Mbps', 300)

*,Bill information of Customer :
(id bill, id customer, id dataplan, registration date, duration(month), total amount(*1000VND):

   (2, 48484848, 2, datetime.date(2021, 5, 19), 12, 3600)


-----------------------------------Please complete your payment in time!-----------------------------------
------------------------------------Thank you for choosing our service-------------------------------------
```

4. Conclusion:

- The project is done, after being tested by our Customers, we have plans to improve our software due to the requested information from Customers. We will also upgrade our servers and update the database in order to improve our software to be as good as the customers' expectation:
    - Search Customer and Dataplan by name
    - Separate file main into 3 file : input, output, main for maintain
    - ……….