

CHAPTER 5

RESULT AND ANALYSIS

5.0 Introduction

After completing all the process the project will be build and run. From the complete project we can see the result and analyze the result. This section will show the result of the Bluetooth Device scanner, GUI and the client side program.

5.1 Bluetooth Device Scanner

Figure below show the result of the Bluetooth device scanner program. The result will be compared to Network Chemistry Bluetooth Device Scanner to scan the nearby Bluetooth device.

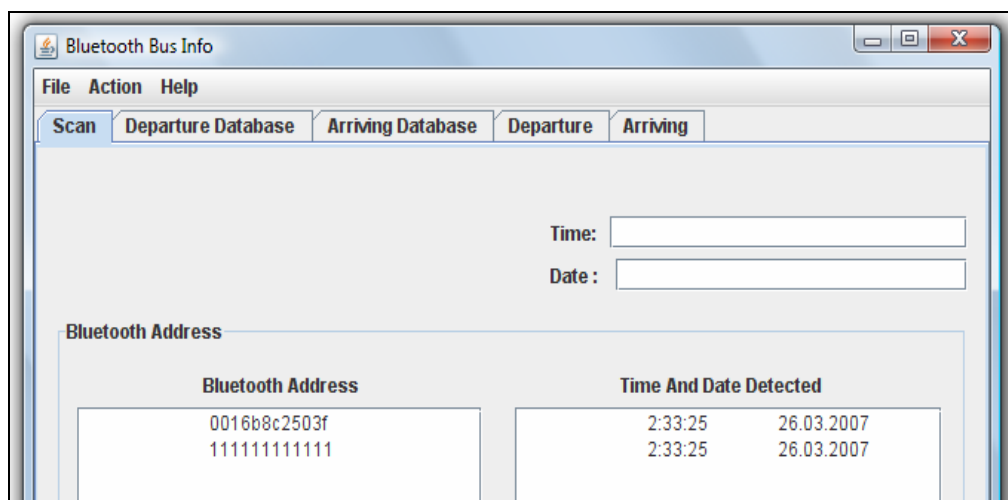


Figure 5.1: Bluetooth Device Scanner Result

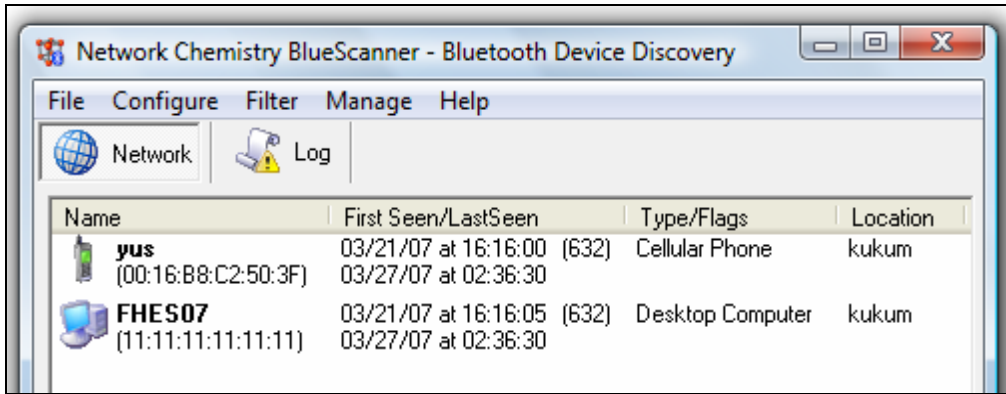


Figure 5.2: Network Chemistry Bluetooth Device Discovery Result

5.2 Database Graphical User Interface Result

The result for this section will be the output from the server side GUI. A couple of figure will be attached to show the result before and after a button is click.

5.2.1 Add data to database

- Before add data to the database

Departure Database

Add Edit Delete

Add Depart Data

Bluetooth ID :

Company :

Registration Number :

Destination :

Time Depart :

Platform :

Clear Field Add

BluetoothID	Company	Regno	Destination	Time Depart	Platform
000e6db1ac04	kukum	JHJ9981	Kangar	08:30	Platform 1
0012d13f4ac2	kukum	nbn9981	kubang gajah	08:00	platform 3
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2

Figure 5.3: Process to Add data to Database

- Fill the text field with new data

Departure Database

Add Edit Delete

Add Depart Data

Bluetooth ID : 123456789009

Company : kukum

Registration Number : afd9981

Destination : KWSP

Time Depart : 08:00

Platform : platform 2

Clear Field Add

Figure 5.4: Insert Data to Field

- The result after the Add button is clicked

Departure Database

Add Edit Delete

Add Depart Data

Bluetooth ID :

Company :

Registration Number :

Destination :

Time Depart :

Platform :

Clear Field Add

BluetoothID	Company	Regno	Destination	Time Depart	Platform
000e6db1ac04	kukum	JHJ9981	Kangar	08:30	Platform 1
0012d13f4ac2	kukum	nbn9981	kubang gajah	08:00	platform 3
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

New data location

Figure 5.5: Result After Add button is clicked

From the Figure 5.2, the data from the Figure 5.3 will be send to the database and the database will send it back to the table and display the result in the table. The new data is located in line 5 in the departure database table.

5.2.2 Edit the data in the database

If a data have the typing error or the administrator need to update the data the edit database tab will be used. Here is the demo to edit the data in the database.

- Let say the administrator want to edit the data at line 1 in the departure database table, fill the text field with a new data. The BluetoothID will remain and it can't be change because the Bluetooth ID is primary key to the departure database.

The screenshot shows a software interface titled "Departure Database". At the top, there are three tabs: "Add", "Edit", and "Delete", with "Edit" being the active tab. Below the tabs is a form titled "Edit Depart Data". The form contains six text input fields, each with a label and a value:

- Enter Bluetooth ID : 000e6db1ac04
- Edit Company : unimap
- Edit Registration No: re9981
- Edit Destination : KWSP
- Edit Time Depart : 08:00
- Edit Platform : platform 2

At the bottom right of the form are two buttons: "Clear Field" and "Update". Below the form is a table with the following data:

BluetoothID	Company	Regno	Destination	Time Depart	Platform
000e6db1ac04	kukum	JHJ9981	Kangar	08:30	Platform 1
0012d13f4ac2	kukum	nbn9981	kubang gajah	08:00	platform 3
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

Figure 5.6: Process to Edit Data from Database

- When the Update button is click we can note that the data in line 1 is change.

Departure Database

Add Edit Delete

Add Depart Data

Bluetooth ID :

Company :

Registration Number :

Destination :

Time Depart :

Platform :

Kukum was change to unimap

Clear Field Add

BluetoothID	Company	Regno	Destination	Time Depart	Platform
000e6db1ac04	unimap	re9981	KWSP	08:00	platform 2
0012d13f4ac2	kukum	nbn9981	kubang gajah	08:00	platform 3
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

Figure 5.7: Result after Update button is clicked

5.2.3 Delete data from database

When a data is not needed or already expired, the administrator can delete the data from the database. To delete a data in the database we need a unique text. In this project is the Bluetooth ID or registration number of the busses. The administrator can choose to delete data from database using the Bluetooth ID or registration number. The process will be show by Figure 4.7 below;

Departure Database

Add Edit Delete

Delete Depart Data By:

Bluetooth ID :

OR

Registration Number :

Clear Field Delete

BluetoothID	Company	Regno	Destination	Time Depart	Platform
000e6db1ac04	unimap	re9981	KWSP	08:00	platform 2
0012d13f4ac2	kukum	nbn9981	kubang gajah	08:00	platform 3
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

Figure 5.8: Delete data from database using Bluetooth ID

In this example, the administrator will delete a data using the Bluetooth ID 000e6db1ac04 from the database. Figure below show the result when the Delete button is clicked. Note that the data with a Bluetooth ID 000e6db1ac04 is deleted.

Departure Database

Add Edit Delete

Delete Depart Data By:

Bluetooth ID :

OR

Registration Number :

Clear Field Delete

BluetoothID	Company	Regno	Destination	Time Depart	Platform
0012d13f4ac2	kukum	nbn9981	kubang gajah	08:00	platform 3
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

Figure 5.9: Result After Delete button is clicked

Figure below show the process to delete data using the registration number. In this example, a data with the registration number mbg9981 will be delete form the database.

Departure Database

Add Edit Delete

Delete Depart Data By:

Bluetooth ID :

OR

Registration Number :

Clear Field Delete

BluetoothID	Company	Regno	Destination	Time Depart	Platform
001370285bb9	kukum	mbg9981	automart	07:45	Platform 2
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

Data to be deleted

Figure 5.10: Delete data from database using Registration Number

When the Delete button is clicked, note that the data with the registration number mbg9981 is deleted from the database

Departure Database

Add Edit Delete

Delete Depart Data By:

Bluetooth ID :

OR

Registration Number :

Clear Field Delete

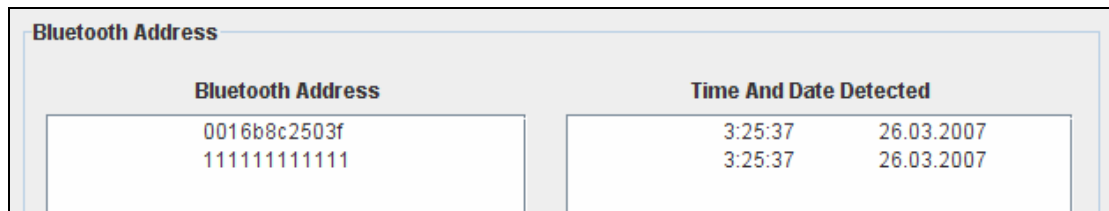
BluetoothID	Company	Regno	Destination	Time Depart	Platform
0016b8c2503f	kukum	bha9981	muhibah	08:00	platform 2
123456789009	kukum	afd9981	KWSP	08:00	platform 2

Figure 5.11: Result after Delete button is clicked

5.2.4 Bus information panel

This panel will show the data to be display to the screen. When the Bluetooth scanner is running, the information panel will automatically compare the discovered device Bluetooth address with the Bluetooth address in the database. If the discovered Bluetooth address is match with the Bluetooth address stored in the database, the panel will show the detail about the Bluetooth address in the information panel.

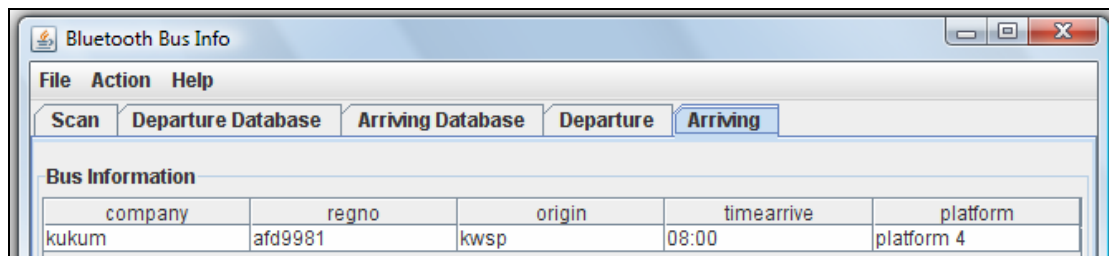
Figure below will show the result from the information panel compare to Bluetooth device scanner



The screenshot shows a window titled "Bluetooth Address" with two columns: "Bluetooth Address" and "Time And Date Detected".

Bluetooth Address	Time And Date Detected
0016b8c2503f	3:25:37 26.03.2007
111111111111	3:25:37 26.03.2007

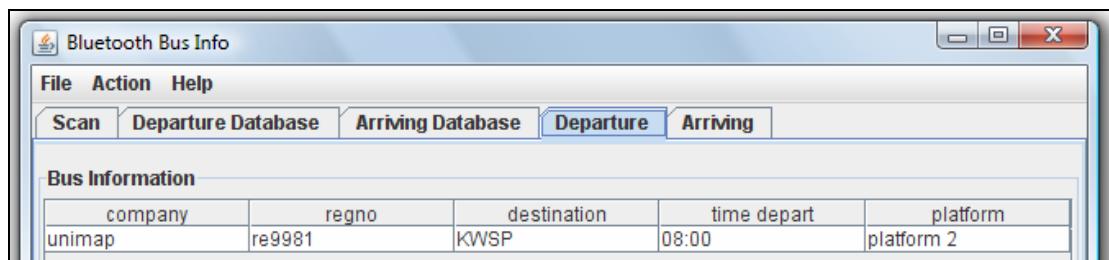
Figure 5.12(a): Bluetooth Device Scanner Result



The screenshot shows the "Bluetooth Bus Info" application window with the "Arriving" tab selected. The "Bus Information" table displays the following data:

company	regno	origin	timearrive	platform
kukum	afd9981	kwsp	08:00	platform 4

Figure 5.12(b): Result that appear at Arriving Information Panel



The screenshot shows the "Bluetooth Bus Info" application window with the "Departure" tab selected. The "Bus Information" table displays the following data:

company	regno	destination	time depart	platform
unimap	re9981	KWSP	08:00	platform 2

Figure 5.12(c): Result that appear at Departure Information Panel

5.3 Phone Side User Interface

To request information from server, users need to choose the departure or arriving information. To choose between departure information and arriving information, users can use navigation button shown below and press SELECT. The real button on phone will depend on model and manufacturer.



Figure 4.13: Navigation button in wireless toolkit emulator

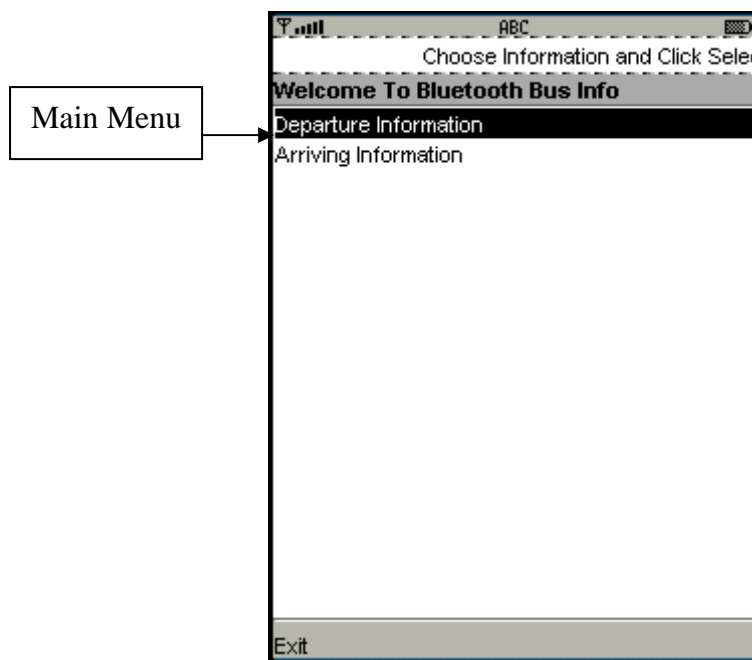


Figure 5.14: Main Menu in phone

Once the SELECT button is pressed, the GUI will jump to the next page depending on the chosen menu. In this demo, the next page will be the departure information form to request information about the departure journey from the server.

5.3.1 Send Request to the Server

To request departure information from server, users need to fill the text field with the company and destination of the busses. Figure below show the GUI to request departure information from server.

The image shows a mobile application interface for requesting departure information. The screen is titled "Departure Information" and has a status bar at the top with "ABC" and "Insert The Company". The form contains two text input fields: "Company" and "Destination". A "Back" button is located at the bottom left, and a "Send" button is at the bottom right. Three callout boxes are present: one on the left pointing to the top of the form area with the text "Instruction to fill the text field", one on the right pointing to the "Company" text field with the text "Company text field", and another on the right pointing to the "Destination" text field with the text "Destination text field".

Figure 5.15: Departure Information Form

To fill the text field with the data, users can use the button shown below. The method to fill the field is the same method to write the SMS, depending on the model and manufacturer. In the emulator, the method to fill the field can be using emulator button or keyboard.



Figure 5.16: Button layout for the emulator

After the request data is filled. The next step to send request to server is clicking the send button. Back button will return the page to previous menu.

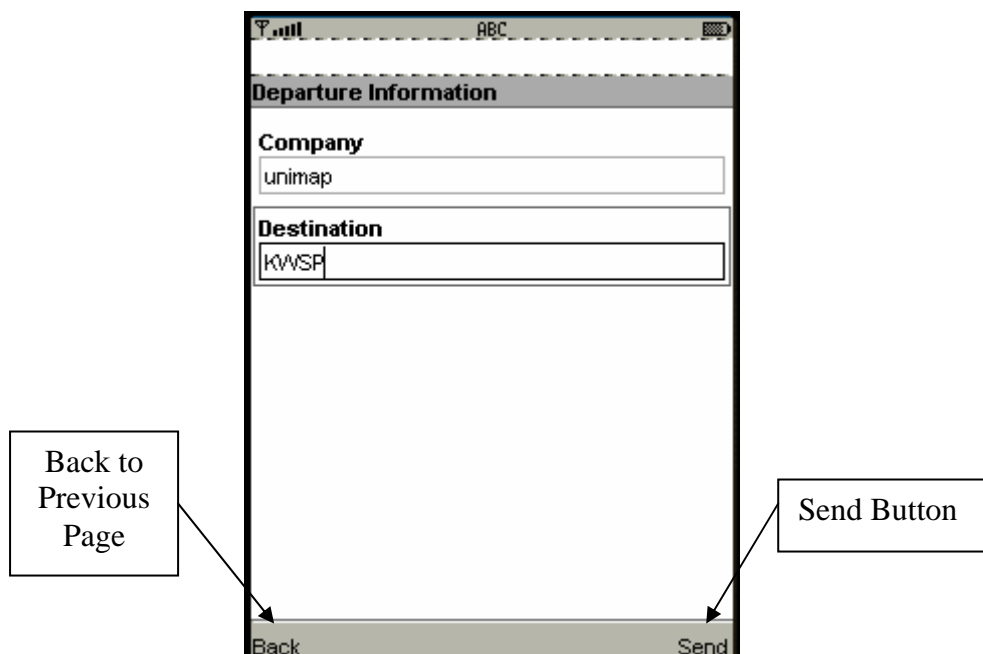


Figure 5.17: Departure Information Form with the request data.

5.3.2 Receive information from Server

If the client and server are connected, the screen on the phone will display the information from server. Figure below show the information of departure buses request by the users. Users can press Back to return to previous page or exit to terminate the program.

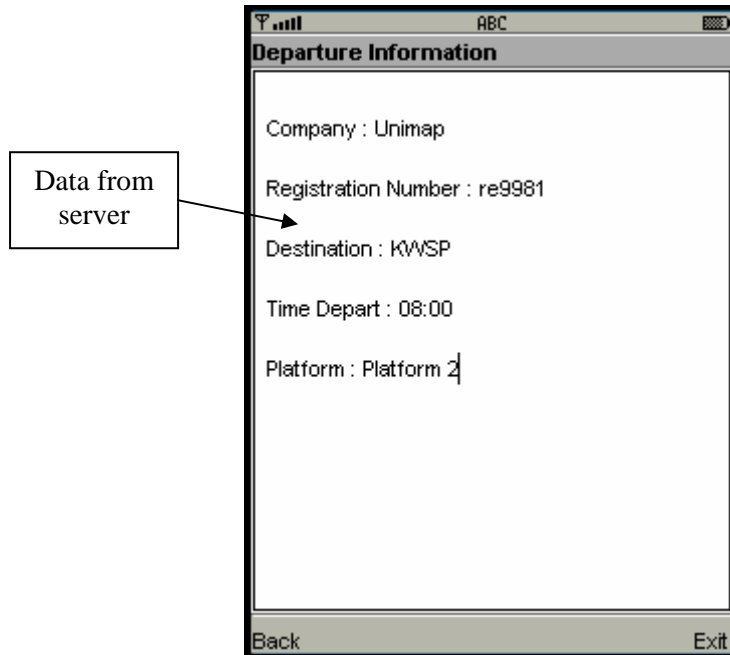


Figure 5.18: Result received from server

5.4 Discussion

After completing the design and developing this project, the program was tested using PC and 2 mobile phones that support Java Environment. First mobile phone will be Nokia 6600 and the other one is LG M6100. The result for mobile part in Section 4.3 came from Wireless Toolkit from Sun Microsystems. The reason why the result is just come from emulator is because the screen on the mobile phone cannot be printed and shown in the report. The emulator result should be working on mobile phone as long as the phones fulfil the requirement of the program.

The program tested on Nokia 6600 phone shows that the phone part can't get the data send by the server even when the emulator shows it does. Maybe the phone screen sizes are not compatible with program screen size. PC program can receive the phone request and it can be viewed.

The other major problem in this system is the running process of the system which decreasing after a long period. This happen because the scanning program will continuously scan for nearby device. That is the main reason why stand alone database is used. If the database is connected with external database, the result would be worse.

This project only works if the PC comes with integrated Bluetooth device or plug and play Bluetooth dongle. This limitation will become a burden to the user either to buy new PC or plug and play Bluetooth dongle. The system is not tested with plug and play dongle because the workstation used in this project come with integrated Bluetooth device. However, there is a solution for non plug and play dongle. The system can run in Linux or Mac Operating System (OS) which not need the plug and play dongle.

Other limitation for this project is the range of Bluetooth itself. The widest range for Bluetooth is 100m. This will be a great limitation for this project since the bus station area could be bigger than 100m. Solution for this problem is to extend the range of the Bluetooth range by creating another Bluetooth network around the bus station (piconet). There another project that try to extend the range of the Bluetooth, but the experiment is against the FCC rules and manufacturer right.

Bluetooth device discovery process is not real time process. The time to detect the nearby Bluetooth device may take several seconds and it will become longer if it runs in client program. The delay cannot be eliminated but it can be reduce. Overall study about Bluetooth connection show Bluetooth is still left behind by other wireless communications. But the next generations of Bluetooth will response to this limitation and create a solution to improve the Bluetooth connections.