PSZ 19:16 (Pind. 1/97)

UNIVERSITI TEKNOLOGI MALAYSIA

BORANG PENGESAHAN STATUS TESIS◆

JUDUL: **BLUETOOTH ANNOUNCEMENT SYSTEM**

SESI PENGAJIAN: SEMESTER II 2008/2009

Saya YEO TZE CHUNG
(HURUF BESAR)

mengaku membenarkan tesis (PSM/~~Sarjana/Doktor Falsafah~~)* ini disimpan di Perpustakaan Universiti Teknologi Malaysia dengan syarat-syarat kegunaan seperti berikut:

1. Tesis adalah hakmilik Universiti Teknologi Malaysia.
2. Perpustakaan Universiti Teknologi Malaysia dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan tesis ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. **Sila tandakan (✓)

☐	**SULIT**	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD**	(Mengandungi maklumat TERHAD yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☑	**TIDAK TERHAD**	

Disahkan oleh

(TANDATANGAN PENULIS)

Alamat Tetap:

564, lot 147 Kenyalang Park phase 2,
93300 Kuching Sarawak

Tarikh: 11 MAY 2009

(TANDATANGAN PENYELIA)

En Syed Othmawi b Abdul Rahman
NAMA PENYELIA

Tarikh: 11/5/09

CATATAN:
* Potong yang tidak berkenaan.
** Jika tesis ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali sebab dan tempoh tesis ini perlu dikelaskan sebagai SULIT atau TERHAD.
◆ Tesis dimaksudkan sebagai tesis bagi Ijazah Doktor Falsafah dan Sarjana secara penyelidikan, atau disertasi bagi pengajian secara kerja kursus dan penyelidikan, atau

“I declare that I have read this project and in my opinion this project report has satisfied the scope and quality for the award of the degree of Bachelor of Science (Computer Science)”.

Signature : _____

Name of Supervisor : En Syed Othmawi b Abdul Rahman

Date :_____

Bluetooth Announcement System

YEO TZE CHUNG

A project report submitted in partial fulfillment of the

requirements for the award of the degree of

Bachelor of Science (Computer Science)

Faculty of Computer Science and Information System

Universiti Teknologi Malaysia

MAY, 2009

DECLARATION

I hereby declare that this project entitled “Bluetooth Announcement System” is the result of my own research except as cited in the references**.**

Signature : _____

Name of Candidate : Yeo Tze Chung

Date :_____

DEDICATION

This thesis is dedicated to my beloved family.

ACKNOWLEDGEMENT

Firstly, I would like to thank my PSM supervisor, En. Othmawi, by guiding me along the PSM progress and provides a lot of valuable opinions and advices to me when I face difficulties during the project.

Besides that, I also appreciate the helps from all my friends and my family who giving their support and encouragement to me in order to complete this project. Without their encouragement, this project will probably cannot done on time.

Last but not least, I am grateful to all staffs and lecturers in Faculty of Computer Science and Information System, Universiti Teknologi Malaysia for their help and support at every step during this course of studies.

Abstract

First version of announcement system was traditional notice board. Which mean that the notice board is made of cork or soft plastic. Officer needed to update the information from time to time manually. Many universities still keep this traditional notice board because it is easy to develop and save cost of operation. After few years later, electronic notice board was developed to save the work of officer. Electronic notice board is more effective than traditional notice board because electronic notice board is a simple to use interface which allows an end user to quickly and easily update any information that they wish to pass on to visitors in real time. Scheduling capabilities are also included which allows a company to target specific information at a certain audience at a particular time. Nowadays after the fast development of Bluetooth technology, it is well use in Bluetooth marketing and Bluetooth advertising. Bluetooth is also use in announcement system for improve the effectiveness of announcement system. Hence the developing of Bluetooth announcement system can support the students life cycle is a personalized way.

Abstrak

Pada permulaanya, sistem pengumuan yang biasa digunakan oleh kalangan masyarakat adalah papan notis. Papan notis diguna secara meluas kerana kos yang diperlukan adalah sangat rendah. Malah ia sangat mudah untuk didapati dan dibangunkan. Semasa penggunaan papan notis, kerani perlu mengemaskini papan notis dari masa ke semasa. Selepas penggunaan papan notis beberapa tahun kemudian, papan notis elektronik dibangunkan untuk mengurangkan kerja kerani. Kerani hanya perlu mengemaskinikan notis di antarakamuka pengguna. Walaupun demikian, masih terdapat kelemahan yang didapati. Oleh itu, system pengumuman Teknologi Bluetooth dibangunkan untunk mengatasi masalah yang didapati.

Index

List of Table

List of Figure

Chapter 1

Introduction

1.1 Introduction

"Is my class canceled today?", "When and where is the PSM seminar?". These is typical questions junior students, freshman and senior students in Faculty of Science and Information System (FSKSM) will ask at the beginning of each new semester or even at the middle of semester. These typical question will pop out in their mind was due to the ineffective announcement system use in FSKSM. These situations tell us that message cannot separate around by using the current announcement system. Hence a brand new system needed to solve this problem.

We can use the advantages of current technology to solve this problem. As everyone knows, mobile is a very useful and common device. Most of us use

mobile to communicate with each others, through SMS, call, MMS and etc. After the development of Bluetooth technology, mobile is not only use for communication, but now mobile can be use to obtain information through Bluetooth. For example Bluetooth technology phone is widely use in advertising and marketing, moreover mobile Bluetooth is also use for providing useful information for tourist as well.

Hence, as a same concept a Bluetooth-based announcement system can be develop to upgrade the current announcement system in FSKSM. This system name Mobile Bluetooth Announcement System (M-BAS). M-BAS can support the students live cycle in a personalize way and helps the user on FSKSM to find and access information, which is interested and relevant to them. They only need a Bluetooth enabled mobile device, which would be either PDA or a mobile phone, walk into the Bluetooth coverage area and they can pull the require information from the server and general announcement will be push to students mobile.

1.2 Problems

The use of Bluetooth technology still in an early stage and different user are still experimenting with various format. However research and practice suggest that much can be achieved using simple technology that is already well established.

In the past, announcement system in FSKSM building had limited media option such as print media and electronic board. Targeting user through exiting system is still risky. As we can notice, when there is seminar or changes of class

schedule, there would be some student miss the seminar or class because they not read notice board from time to time.

Main notice board of FSKSM was located at ground floor and other mini notice board was located at the entrance of lift every floor. Normally, important notice or announcement will be stamp at the notice board for students at FSKSM to get information relevant to them. At the beginning of semester, the notice board was well organized and in order, but after come to the middle of semester, notice board was start getting disarray. Some more, FSKSM officer or staff need to update notice board manually, and stamp the announcement at the common area. Sometimes announcement need to stamp at notice board on every floor. This step is not efficient at all.

Besides, FSKSM building also lack of electronic notice board. Electronic notice board only located at second floor. Hence if students not pass through second floor they will not know the announcement posted on electronic notice board. Thus, a brand need system need to come out so that current problem can solve.

1.3 Purpose of the system

Based on the problem discuss above, the purpose of this study is to gain a solution to improve announcement system in FSKSM using Mobile Bluetooth Technology.

1.4 Project Objective

To achieve purpose of the project, some objective been identify as below:

(i) Make use of Bluetooth technology nowadays in student life cycle

(ii) "Message Undelivered" problem can be reduce

(iii) Provide convenience way for user to retrieve latest information.

1.5 Project scope

(i) Only public (student, staff and etc) who own Bluetooth mobile, PDA or laptop can receive the announcement.

(ii) This application/system maybe limited to certain model of mobile phone only.

(iii) Public can receive the information only when they switch on their Bluetooth device.

1.6 Summary

This system is developing to solve the current announcement system problem. Its help supports the students life cycle in a personalized way at any time at any location. M-BAS helps students on FSKSM to find and access information related to them.

Chapter 2

Literature Review

2.1 Introduction

The previous chapter provided the background and problem discussion of the area of this thesis. In this chapter we will review earlier studies about current system within our purpose area. The aim of this chapter is to provide relevant literature on uses of Bluetooth technology in campus announcement system.

2.2 Introduction to the current system

Current announcement system used in FSKSM was notice board and electronic board. A notice board (pin board, or bulletin board) are made of material such as cork or soft plastic material to facilitate additional and removal message. Notice board is commonly for people to leave message, for example, to advertise things for buy or sell, announce events, or provide information.

Main notice board was located at ground floor of FSKSM building, in front of lift entrance. There are others ten notice boards made of was placed at the same place to improved the efficiency of notice board. When there is notice or announcement needed to inform student, FSKSM officer will leave a message at notice board. Officer will keep on update the information from time to time.

While electronic notice board was located at second floor of FSKSM building. Several years ago, FSKSM had start using electronic notice board because of some advantages. For example, electronic notice boards can help FSKSM officer to keep all FSKSM user informed with all the news and announcement without relying on boring static notice board, because static notices nowadays is not a effective way of communicating. Besides, by using electronic notice board officer reduce their work by just update notice using computer in front of them. They no need to update notice by adding or remove notice at traditional notice board.

2.2.1 Analysis of current system

There are several problems occur when using the current system.

(i) The message display from electronic notice board isn't clearly sent across, as not much detail can be include in the notice.

(ii) The notice is dynamic and constantly changing, user will not wait in front of electronic notice board waiting for the second notice. Thus not all messages will send across user.

(iii) Electronic notice board don't look attractive, user tend not to look at them as they may think from their first impression that it doesn't seem very important.

(iv) It is time consuming when officer update information at notice board from time to time.

2.3 Introduction to the propose system

Nowadays there are several ways to communicate with a group of people. Like mention before from the previous section, we can broadcast a message to a group of people using email, SMS, electronic board or even notice board, but this is not a perfect way to broadcast a message to a group of people. As we can notice that if we broadcast using email or SMS sometimes we may not know the email address or mobile number of the group member. Furthermore, by using electronic

board or notice board, the broadcast message is not portable but by using this Bluetooth broadcasting system message broadcast to the group member is portable. Message receive can be store in mobile so that message can view anytime and anywhere.

This application is a small collection of Python scripts which will allows admin to build up a quickly Bluetooth advertisement or welcome server. Imagine one of the following situations:

(i) FSKSM is conduct a seminar and want to welcome every guest with a short Bluetooth message on their mobile phone.
(ii) Faculty want to remind every student to do their EPPP.
(iii) Time, date and venue of seminar is changed.

All of these situations can be realized with this application.

This application is a message queuing concept. Message can broadcast to anyone or an unique device which is identified through Bluetooth Mac address. The concept behind this application is as follow:

(i) Create broadcast message file – this file cab be a text file, an image file, an audio file or any others file type. But the condition is the file must be interpreted by the target mobile phone.
(ii) Add file to the message queue – specify a broadcast or target address and can set a timeout if necessary.
(iii) Toggle on Bluetooth connection on mobile phone and wait for the execution of the daemon
(iv) Message will be sent to mobile when connection established.

2.4 Internal working concept

This application is based on two main parts: the message queue utility and the daemon

Message queue utility:
Message queue utility is managed by add-message.py. add-message.py takes a defined input file and put this files into the queue.

Daemon:
Execute-server.py is the core of this application. It takes a defined CSV file, which includes the current Bluetooth devices in neighbour. After execution the daemon searches for outstanding messages of any Bluetooth devices which are defined in the CSV file. If a outstanding message is found, the daemon triggers an external script. This script will be send-ussp.sh.

2.5 Technology

2.5.1 Introduction to Bluetooth technology

Electronic devices connect to one another in a variety of ways: A cable

connects a computer's processing unit to a display, a data cable and a docking cradle connect a Personal Digital Assistant (PDA) or a cellular phone to a computer, radio waves connect a cordless phone to its base unit, an infrared beam connects a remote control to a television. The elaborate array of connectors among electronic devices cries out for a better solution. That's where Bluetooth comes in.

The Bluetooth wireless connectivity technology was originally started in 1994 by the Swedish phone equipment maker Ericsson as a way for mobile devices to communicate with each other at short ranges -- up to 30 feet, or 10 meters. In 1998, Ericsson, IBM, Intel, Nokia, and Toshiba formed the Bluetooth Special Interest Group consortium to develop a royalty-free, open specification for short-range wireless connectivity. Since then, more than 2000 companies have joined the Bluetooth SIG, including virtually all manufacturers of phone, computer, and PDA equipment. The specifications for the Bluetooth version 2 was released in 2004, though Bluetooth version 1.x (1.0, 1.1 and 1.2) are still more common in devices used today.

Each Bluetooth devices has a unique address, also called Bluetooth ID or device ID. This ID is usually not shown to users, as more user friendly customizable Bluetooth name is shown instead.

Bluetooth operates with radio waves using the 2.45 GHz band. This band is license free, and is therefore also used by others appliances such as microwaves ovens and baby monitors. To avoid interference with other devices, the Bluetooth protocol divides this band into 79 channels, each of them 1 MHz wide. A Bluetooth transmission changes between these channels up to 1600 times per second.

The transmission range of Bluetooth ranges depending on the power class of the implementation. It has three power classes, where class 2, with a range of

about 10 meters is the most common. The others are class 1, up to 100 meters, and the rarely used are class 3, which has a range of 10 cm, up to a maximum of 1 meter. The maximum speed of Bluetooth transfer is 723.1 kilobits/s in version 1.1 – 1.2 and 2.1 megabits/s in version 2.0

Devices with Bluetooth can usually set it in one of three modes: Off, hidden and discoverable. When set to discoverable, other devices will be able to see it when searching for devices, and can attempt connecting to it. When setting it to hidden, other devices will not be able to see it when they search for devices. Devices will have to know the Bluetooth ID in order to connect to a hidden device. This can be done by pairing the devices, by using a pass code which the users agree on. This has to be done when at least one of the devices is discoverable. When the code is recognized, the two devices will store the other device's Bluetooth ID, so that they can connect to each other even when they are set to hidden.

2.5.2 Bluetooth Network Topology

Bluetooth-enabled devices are organized in groups called piconets. A piconet consists of a master and up to seven active slaves. A master and a single slave use point-to-point communication; if there are multiple slaves, point-to-multipoint communication is used. A master unit is the device that initiates the communication. A device in one piconet can communicate to another device in another piconet, forming a scatternet, as depicted in Figure 1. Notice that a master in one piconet may be a slave in another piconet:

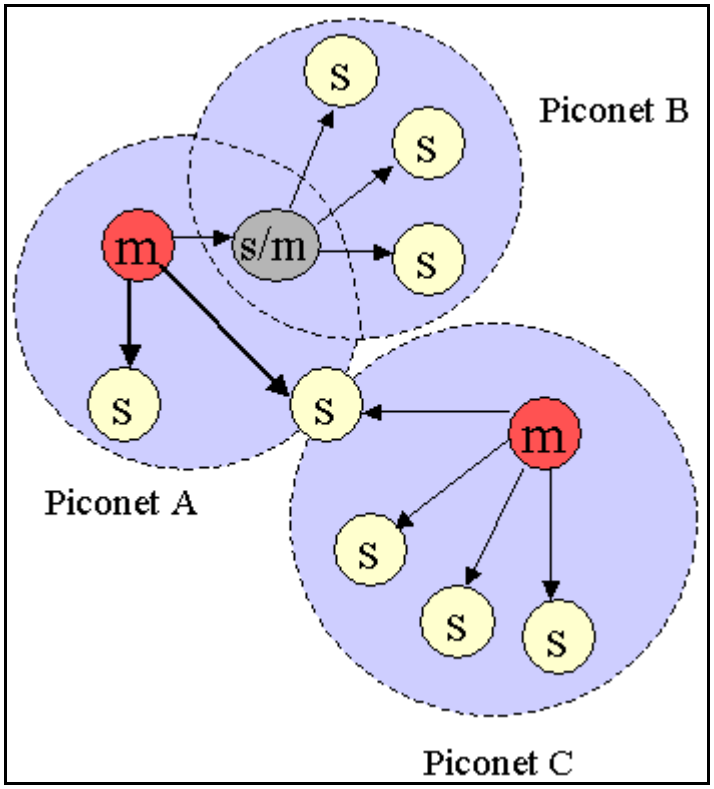


Figure 2.1: Scatternet Comprising Three Piconets

The normal duration of transmission is one slot, and a packet can last up to five time slots in length. In order to support full-duplex communications, Bluetooth uses a Time-Division Multiplexing (TDM) scheme, in which a master device always uses an even-numbered slot when it transmits, and a slave uses an odd-numbered slot.

Wireless application programming with J2ME and Bluetooth by Qusay H.Mahmoud from http://www.sundeveloper.com

2.5.3 Low-Power Operating Modes

Bluetooth defines provisions for three low-power operating modes in order to conserve battery life:

Sniff mode - a slave listens at a reduced level and doesn't take an active role in the piconet.

Hold mode – no data transmits, but its clock continues to operate, and a slave remains in synchronization with the master. The device is not an active member of the piconet, but it retains its active member address. Power requirements decrease as a device goes from sniff to hold.

Park mode - is like hold mode in that the slave is synchronized to the master but is not part of the traffic. In this mode, however, the slave doesn't retain its active member address. Power requirements decrease still further as a device goes from hold to park.

2.5.4 Bluetooth Protocol Stack

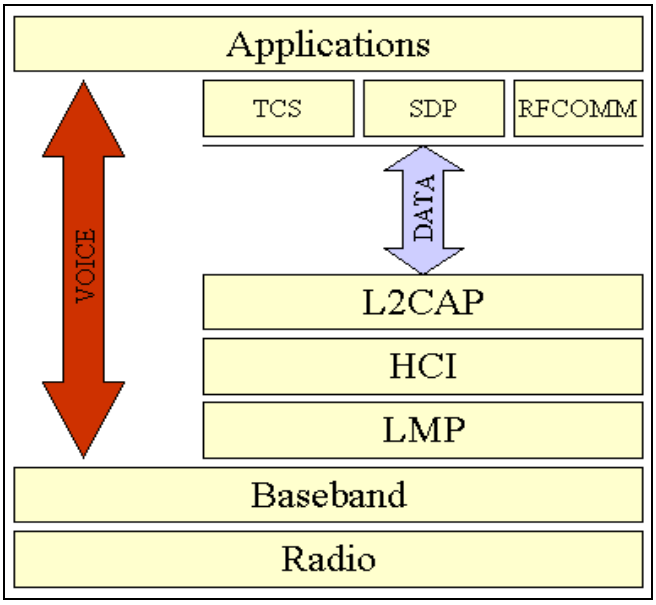


Figure 2.2: High-level view of the architecture of the Bluetooth protocol stack:

The responsibilities of the layers in this stack are as follows:

The radio layer - Physical wireless connection. Radio layer is use to avoid interference with other devices that communicate in the ISM band. 2.4GHz frequency band been divided by Bluetooth devices into 79 channels. Those channels is uses to spread spectrum hop from one channel to another, up to 1600 times a second. The standard wavelength range is 10 cm to 10 m, wavelength can be extended to 100m by increasing transmission power.

The baseband layer - responsible for controlling and sending data packets over the radio link. Baseband layer provides transmission channels for both data and voice. The baseband layer maintains Synchronous Connection-Oriented (SCO) links for voice and Asynchronous Connectionless (ACL) links for data. SCO

packets are never retransmitted but ACL packets are, to ensure data integrity.

SCO links - point-to-point symmetric connections, where time slots are reserved to guarantee timely transmission. A slave device is allowed to respond during the time slot immediately following an SCO transmission from the master. A master can support up to three SCO links to a single slave or to multiple slaves, and a single slave can support up to two SCO links to different slaves.

Data transmissions on ACL links, on the other hand, are established on a per-slot basis (using slots not reserved for SCO links). ACL links support point-to-multipoint transmissions. After an ACL transmission from the master, only a slave addressed specifically may respond during the next time slot; if no device is addressed, the message is treated as a broadcast.

The Link Manager Protocol (LMP) - uses the links set up by the baseband to establish connections and manage piconets. Responsibilities of the LMP also include authentication and security services, and monitoring of service quality.

The Host Controller Interface (HCI) - is the dividing line between software and hardware. The L2CAP and layers above it are currently implemented in software, and the LMP and lower layers are in hardware. The HCI is the driver interface for the physical bus that connects these two components. The HCI may not be required. The L2CAP may be accessed directly by the application, or through certain support protocols provided to ease the burden on application programmers.

The Logical Link Control - and Adaptation Protocol (L2CAP) receives application data and adapts it to the Bluetooth format. Quality of Service (QoS) parameters are exchanged at this layer.

[Adapted from http://developers.sun.com/mobility/midp/articles/Bluetooth1/]

2.5.5 Establishing a Network Connection

When a device is not connected to a piconet, it is in a standby mode. In this mode, the device listens for messages every 1.28 seconds over 32 hop frequencies. When one device wishes to establish a connection with another, it sends out 16 identical *page* messages on 16 hop frequencies. If the slave doesn't respond, the master retransmits the page message on the other 16 hop frequencies. If the master doesn't know the slave's address it must preceed the page message with an *inquiry* message, which requires an extra response from the slave unit. When the slave responds to the page message, the master can begin transmitting voice or data.

Steps device needed to establish connection with a nod:

1. Inquire: In a new environment, the device automatically initiates an inquiry to find an access point. All nearby access points respond with their addresses, and the device picks one.

2. Page: The paging procedure synchronizes the device with the access point.
Establish a link: The Link Manager Protocol establishes a link with the access point.

3. Discover services: The LMP uses the Service Discovery Protocol (SDP) to find out what services are available from the access point. Here we assume that the email service is available.

4. Create an L2CAP Channel: The LMP uses information obtained from the Service Discovery Protocol (SDP) to create an L2CAP channel to the access point. The application may use this channel directly or use a protocol like RFCOMM (Radio Frequency Communications Protocol) that might be running over L2CAP. RFCOMM emulates a serial line.

5. Create an RFCOMM channel: Depending on the needs of the application, an RFCOMM channel (or another channel) is created over the L2CAP channel. Creating an RFCOMM channel allows an existing application that works with serial ports to work with Bluetooth as well, without any modifications.

6. Authenticate: This is the only step that requires input from the user. If the access point requires authentication, it will send an authentication request, and the user will be prompted to enter a PIN to access the service. For security reasons, the PIN code itself is not sent over the wireless link, but rather a key generated from it.

7. Log in: If the devices use the Point-to-Point Protocol (PPP) over RFCOMM, a serial port is emulated, and Sally can log in to her email account.

8. Send and receive data: The email client and the access point now use standard network protocols like TCP/IP to send and receive data.

[Adapted from http://developers.sun.com/mobility/midp/articles/Bluetooth1/]

2.5.6 Why Choosing Bluetooth Technology

Using Bluetooth technology for broadcasting information allows for a way of broadcasting that is quite different from any other currently common broadcasting technologies. Bluetooth broadcasting systems can be used in situations where the other technologies are not suitable.

(i) Area broadcasting instead of group broadcasting

Unlike other digital channels for sending information, Bluetooth broadcasting allows for directing broadcasted information towards all people that are in the area, rather than a group of people. Bluetooth broadcasting technology can look as a short range radio or TV broadcast, where everyone can receive information broadcast as long as user have the receiver. The reason why we say short range is because Bluetooth technology nowadays still in limited range.

When compare with messaging service like SMS/MMS and email, the most noticeable difference is that these services require a list of know user in order to send messages. Thus Bluetooth broadcasting is more useful in this area instead of just sending message to know user only. Example, first year junior who just come to FSKSM. They be able to access the broadcast and receive information like FSKSM map and etc.

(ii) Portable Information

When sending information through email or putting it on the internet like e-learning, the information would only be accessible from a computer, which in case is not easily to be portable. If putting information on notice board/electronic board,

students have to write it down, if wish to view the information away from notice board.

However sending information through Bluetooth to mobile phone can be easily portable and viewed at any time.

(iii) Cost Free distribution

While SMS and MMS messages could fulfill the same purpose of Bluetooth broadcasting, but using SMS/MMS require extra free from service provider. Using Bluetooth has the additional advantages of being free to use. There is no need to pay for each message sent to users. It is completely free of charge.

2.5.7 Bluetooth Target Devices

For a Bluetooth broadcasting system, the main target for the Bluetooth will be devices that move within range of the broadcast, without necessary residing there. In most cases this will be mobile phones or PDAs. Bluetooth enable laptops could also possibly be a target.

(i) Mobile phones

Mobile phones are portable phones that communicate with other phones using wireless mobile phones networks. Original mobile phones is use for calls, most modern phones can also use to send and receive text and multimedia

messages, and many phones also embedded extra function like camera and music playback.

Due to the large variations in what is supported by different mobile phone models, it is difficult to say that what kind of file can be used in a broadcast targeted at mobile phones. Files can be transmitted through Bluetooth broadcast device to mobile phones is depend on the model of mobile phones.

(ii) PDAs(Personal Digital Assistants)

PDAs are handheld computers with a touch screen for user interaction. Original PDAs just mean to replace personal organizers, but nowadays PDAs can run vast number of different program, making them practically small personal computers.

The two most used operating systems for PDAs are palm OS and Windows Mobile. Both operating system support Bluetooth, but the way they handle it differs greatly:

Palm OS will receive any file via Bluetooth, but if no application set to handle the file receive file type, palm OS will state it as unknown file and discard it.

Windows Mobile also can receive any file via Bluetooth, if there are no application set to open the file, OS will put it at Temp folder. Leaving it up to user to open it and find the file manually.

(iii) Laptops

Laptops are portable computers, with a keyboard and a flat screen that can

be folded over the keyboard. While laptop used to be twice as expensive as stationary computer in the early days, the price has drop significantly, making them a lot more common for personal usage.

Laptops with Bluetooth can usually accept anything sent to them, allowing users to accept also uncommon file types which may be used by software they have installed. Sending files to laptops should therefore not be much of problems.

2.5.8 File types can be transmit using Bluetooth technology

Since both PDAs and mobile phone will have a large limitation in the type of files that can be sent, this has to be taken into consideration in a broadcasting application. Two ways to solve this problem is:

(i) send only file types that all mobile phones can handle
(ii) let the user choose what type of file to receive

File that can be transmitted via Bluetooth broadcasting as follow:

- Text[as txt files]
- Still images [as GIF or jpeg]
- Animated images [as animated GIF]
- Audio [as WAV, RMF, MP3, MP4]
- Video [as RM,3gp or MP4 files]
- Java application [as JAR files]

- This list contains most commonly supported file types on mobile phones.

Notes: Not all of the mentioned file will work on all mobile phones. In this System only text files will be use.

2.6 Linux Mint and Python language is choose instead of Windows and Java

2.6.1 Linux Mint

Linux Mint is an operating system mostly based on and compatible with Ubuntu. The design of the user interface is considerable different. A distinct user interface like simplified bootloader, desktop layout, theme, and the custom Mint menu. Besides, Linux Mint is strong focus on full functionality. This means that Linux Mint not require any additional, installations, plugin, expansion pack and etc. Plus Linux Mint had then Mint tools, which is a collection of system tools designed to make system management and administration easier for end user.

For example, Linux Mint operating system has build in Bluetooth stack implementation – Bluez. Bluez is the official Bluetooth stack for Linux and is used in Google's android OS. Its goal is to make an implementation of Bluetooth wireless standars specification for Linux. As 2006, the Bluez stack supports all core Bluetooth protocols and layers.

2.6.2 Python Language

Python is an interpreted, interactive, interactive, object-oriented programming language. Python actually is a language that most programmer like to use because of Python is similar in some ways of Perl and Java.

Python is an dynamic object-oriented programming language that can be compare with Java and Microsoft's .net-based language as a general purpose of software development. From statistic Python is the most rapidly growing open source programming language. According to infoworld its user based nearly doubled in 2004, and currently includes about of 14% of all programmers.

Python is being use in many sector, for example Python is use in world's largest exchange, forms the basis for high end newspaper websites, runs on million of cell phone, and is used in industries. Python is available for most of the operating system, including windows, UNIX, Linux, and Mac ox.
Strength of Python:

(i) Integration – integration of all types include Enterprise Application Integration (EAI), Python makes it easy to develop Web Service, calls directly to and from C, C++ or Java code via Python, provides powerful process control capabilities, implements all common internet protocol and data formats, processes XML and other markup language, can be embedded as a scripting language, and runs from the same bytes code on all modern operating systems.

(ii) Network intensive application – Python framework is suited to running large number of concurrent network, database, and inter–process communication links within the same process.

(iii) Web development – Python provides interfaces to most databases, powerful text processing and document processing facilities, and plays well with other web technologies.

(iv) Software testing – Python is come out with own testing unit

(v) Numeric and scientific application

(vi) Application scripting

(vi) Framework

(vii) Desktop development

2.6.2.1 The open source advantages

As all of know that Python is develop as an open source by thousand of programmers all around the world. Hence Python is very well designed, fast, robust, portable, scalable, and portable. With those strengths, Python often exceed the capabilities of comparable commercially available solution. The open source

license for Python allows unrestricted use, modification, and redistribution of the language or anything that is based on it, commercially or otherwise. Full source is available and there are no license costs.

2.6.2.2 Benefits of Python

Research from Python users showing that dramatic productivity benefits over development in other languages, typically with 10 times the development speed seen with C or C++ and a 2 times that seen with Java or Visual Basic.

Python boost is the result of these specific benefits:
(i) Extensible - it is easy to write your own C or C++ extension modules, and then load them into your Python code. Jython, the Java implementation of Python, seamlessly integrates Python and Java modules.

(ii) Powerful - Python isn't just a language. It comes with an extensive standard library of useful modules, including extensive support for email and other internet data formats, HTML, XML and other markup languages, http, ftp, and many other internet protocols, cryptography.

(iii) Portable - Python runs on most operating systems, including some embedded and special-purpose processors

(iiii) Scalable - The speed, power, and extensibility of Python combine to give you a development toolset that will scale effortlessly to the largest projects

2.6.2.3 Python at Google

At Google, Python is one of the 3 "official languages" alongside with C++ and Java. Official here means that Googlers are allowed to deploy these languages to production services. (Internally Google people use many technologies including PHP, C#, Ruby and Perl). Python is well suited to the engineering process at Google

Internally Google has been using Python 2.2. It has been hard for them to move forward to 2.3 or 2.4 to make them capabilities with others machines and language. Google will soon try to move to Python 2.4 and 2.6.

Where is Python used in google?

- The Google build system is written in Python. All of Google's corporate code is checked into a repository and the dependency and building of this code is managed by Python. Greg mentioned that to create code.google.com took about 100 lines of Python code. But since it has so many dependencies, the build system generated a 3 megabyte makefile for it!
- Packaging. Google has an internal packaging format like RPM. These packages are created using Python.
- Binary Data Pusher. This is the area where Alex Martelli is working, on optimizing pushing bits between thousands of servers
- Production servers. All monitoring, restarting and data collection functionality is done with Python
- Reporting. Logs are analyzed and reports are generated using Python.

- A few services including code.google.com and google groups. Most other front ends are in C++ (google.com) and Java (gmail). All web services are built on top of a highly optimizing http server wrapped with SWIG.

(Adapted from http://panela.blog-city.com/Python at Google (Greg Stein - SDForum))

2.6.3 Comparing Python and Java Side by side

Java	Python
Statically Typed In Java, all variable names (along with their types) must be explicitly declared. Attempting to assign an object of the wrong type to a variable name triggers a type exception. Java container objects (e.g. Vector and ArrayList) hold objects of the generic type Object, but cannot hold primitives such as int. To store an int in a Vector, you must first convert the int to an Integer. When you retrieve an object from a container, it doesn't remember its type, and must be explicitly cast to the desired type.	**Dynamically Typed** In Python, you never declare anything. An assignment statement binds a name to an object, and the object can be of any type. If a name is assigned to an object of one type, it may later be assigned to an object of a different type. That's what it means to say that Python is a dynamically typed language. Python container objects (e.g. lists and dictionaries) can hold objects of any type, including numbers and lists. When you retrieve an object from a container, it remembers its type, so no casting is required.

Table 2.1 show the comparing of Python language and Java

Example 1

Java	Python

public class HelloWorld
{
public static void main
(String[] args)
{
 	
System.out.println("Hello,
world!");
}
}
	print "Hello, world!"

Table 2.2 shows the different between the simple programming “Hello, world”

Example 2

Java	Python

int myCounter = 0;
String myString =
String.valueOf(myCounter);
if (myString.equals("0")) ...
	myCounter = 0
	myString = str(myCounter)
	if myString == "0": ...

// print the integers from 1 to
9
for (int i = 1; i < 10; i++) {
System.out.println(i);
}
	# print the integers from 1 to 9
	for i in range(1,10):
	print i

Table 2.3 shows the different between assign string and for loop between Java and Python

Example 3

Java	Python

public class Employee
{
    private String myEmployeeName;
    private int myTaxDeductions = 1;
    private String myMaritalStatus = "single";
 	
    //---constructor #1 ----
    public Employee(String EmployeName)
    {
        this(employeeName, 1);
    }
 	
    //---constructor #2 ----
    public Employee(String EmployeName, int taxDeductions)
    {
       this(employeeName, taxDeductions, "single");
    }
 	
    //---constructor #3 ----
    public Employee(String EmployeName,
           int taxDeductions,
           String maritalStatus)
    {
       this.employeeName   = employeeName;
       this.taxDeductions   = taxDeductions;
       this.maritalStatus   = maritalStatus;
    }
...
	class Employee():
	
	    def __init__(self,
	        employeeName,
	taxDeductions=1,
	maritalStatus="single"):
	
	        self.employeeName = employeeName
	        self.taxDeductions = taxDeductions
	        self.maritalStatus = maritalStatus
	...

Table2.4 shows the different between class implementation in Java and Python

2.6.4 SQLite

SQLite is a small C library that implements in SQL database engine. SQLite is a small database engine, its stand alone and doesn't require any special dependencies and its database itself is stored in a single file. Hence there is no confusing configuration involved, and SQLite is small and, in many applications, quite fast.

2.5 Summary

The aim of this chapter is to provide relevant literature in Bluetooth technology and Mobile broadcasting. The Following chapter will describe what method will be use and how we will conduct the system development.

Chapter 3

Methodology

3.1 Introduction

As state in wikipedia (http://en.wikipedia.org/wiki/Methodology)

Methodology also called manner is defined as

(i) "the analysis of the principles of methods, rules, and postulates employed by a discipline";

(ii) "the systematic study of methods that are, can be, or have been applied within a discipline"; or

(iii)"a particular procedure or set of procedures."

In software engineering environment there are many software process model available can be used for different software development. Hence, we must

choose the most suitable software process model for the software or application we want to develop.

As we know, there are several important phases or stages in developing an application such as the requirement gathering, phase, design, implementation and the validation phase. Thus, after doing research and consideration from all software process models, Rational Unified Process (RUP) will be used.

3.2 Methodology

Software process is a structured set of activities required to develop a software system in the following aspect: Specification, Design and Implementation, Validation and Evolution. While a software process model is an abstract representation of a process. It presents a description of a process from some particular perspective

Nowadays, there are some process model that are widely use by software engineer. For example, waterfall model, evolutionary development, component-based software, Rational Unified Process, spiral model and etc.

RUP was introduced in 1980s and become a division of IBM since 2003, and RUP is a process model that developed and maintained by the Rational Software Corporation. RUP provides several mechanisms, such as short- term iterations with well define goal at the end of each phases. RUP can provide clear view of all the development processes during the development process.

Phases included in RUP:

(i) Inception Process

During inception phase, scope of project will be determined by identify all the external entities that will interact with our proposed system. These entities named as actors which also need to define the interaction among the actor by creating a set of use case to represent the relationship.

Outcome of the phase: Requirement and scopes, a set of initial use cases, a project plan and some risk assessment.

(ii) Elaboration Phase

In this phase a deep view and understanding of our proposed system will be determined. Hence after the full understanding of the system, problem domain will be determined, produce a basic architecture design, develop the project plan, and consider the risk element during the development of the system.

An executable architecture prototype also will be produced in this phase. It is built in one or more iteration, depending on the scope define in the system.

Outcome of this phase: A full use case model, software architecture description, architecture prototype and development plan for the overall project

(iii) Construction Phase

After confirm the architecture design, use case will define at early phase will be convert into coding using suitable software and language. After coding been done, a set of test plan will be organized. All the requirement will be review to ensure that system meet the initial user requirement.

Outcome for this phase: The system and description of the current phase.

(iv) Transition Phase

Purpose of this phase is to transfer the product into the user community. Thus, bugs or error can be detected when the system is introduced to the user. In this phase demo version of system will be released.

Besides four phases mention above, RUP also included nine disciplines that guide you during the whole process. During each iterations we will alternate back

and fourth between activities of the discipline, performing each task to the extent needed at the time, to achieve the goal of the iteration.

(i) Business Modeling

In business modeling discipline, we need to understand the business of the organization, usually will fix it to the scope if the business that is relevant to the development of the system.

(ii) Requirements

In requirements discipline, we will drive a reason, a document, and agree upon of what we going to build and what we should not build in the system. This information is needed to build the system, verify the system, planning and manage the system.

(iii) Analysis and Design

In analysis and design discipline, we will analyze the requirement for the system to design a solution to be implemented, taking into consideration the requirements, constraint and all applicable standard and guidelines.

(iv) Implementation

In implementation discipline, we will transform the design into executable code and to perform a basic level of testing.

(v) Test

In test discipline, we will perform an objective evaluation to ensure the quality by finding and detect defect, error and validating the system work as objective fix earlier and verify that the requirement is met.

(vi) Deployment

In deployment discipline, alpha and beta version of system been created. System will install in the end user workplace.

(vii) Configuration and Change Management

In configuration and change management discipline, we will manage access to the project's work system. System will be tracking from time to time so that controlling and managing can be make on the system.

(viii) Project Management

In project management discipline, we will provide guidelines for managing of the system, handling the simultaneous update conflict. Besides, we also directing people, and coordinate with people and system outside the scope of the project to be sure that it is delivered on time and within the budget.

(ix) Environment

In environment discipline, the main goal is to support the rest of the effort in terms in ensuring that the proper process, guidance, and tools are available as needed.

3.3 Phases Involved in Problem Solving

As stated previously, in Rational Unified Process model, there are four phases that involved throughout the software process which are inception phase, elaboration phase, construction phase and the transition phase

3.3.1 Inception Phase

During the inception phase, the most important element is the requirements

gathering and scopes confirmation. We have to identify all the users who interact with the application and their relationship. We call the user as the actor and develop a set of use cases to clarify their relationship.

During the requirement gathering process, it is essential for us to start our process in our client side. Hence, to gather accurate information and requirement needed by client side or user, several researches are made. For example, several interview sections are made with staff of FSKSM and student in the faculty. During the interview, we can gather some needs from them and transform the requirement into a set of use cases.

Besides the staff authorities, the application user is also the main actor that interacts with this application. Thus, needs and expectation of user is in the consideration.

3.3.2 Elaboration Phase

As mention above, we are required to analyze the problem domain, design a basic architecture design and develop a project plan during this elaboration phase. In order to achieve these goals, it is necessary for us to fully understand the proposed application. Functional and non functional requirement, scopes should be clarified before the architectural model.

In the following phase, we will continue to complete the set of use cases that started in the inception phase. All the requirement and scopes will be finalized at this stage.

As we know that, there are plenty of language and ways to develop a Bluetooth push application. For example, Bluetooth application can be develop by using Java, C#, Python and etc. By referring to the needs that had been gathered, we can determine the best approach that should be used during the application development. In this phase the using programming language is been determined.

3.3.3 Construction Phase

Construction phase is mainly technical part in the software process. As the architecture design and all the use cases have been confirmed, we will start transfer the design part in to the coding and programming part.

Firstly, I do some research and find a lot of relevant tutorial about Python Bluetooth programming since Python Bluetooth programming is a new programming language for me. After such research I start the programming part of the application. In the early stage, Obexftp push is initially used. After doing some research, realized that USSH push is much better than Obexftp since USSH push doesn’t need and authentication from user which is more convenience for user.

After completing all the coding part, a test plan will constructed to test different aspect of the application. Functionality, non functionality, reliability, system performance are all aspect that should be tested using either the white-box testing or the black-box testing. During the testing process, we will analyze the suggestion provided by the application when given certain criteria. All the requirement that gathered in the inception phase will be reviewed to make sure the product meets all the needs from the user.

3.3.4 Transition Phase

The main objective of the transition phase is to transfer the product into the user community. After completing the project, application is carry out and try at a small community. This transition phase may include several iterations.

Once the application has been employed in the real environment, different type of bugs can be detected during the running time. Besides, it is essential to analyze the acceptance of the user for our application. This can be done by organizing a survey among the user who interact with the application and gather their overall satisfaction about this application. Their suggestion will be valuable for improving the product quality.

3.4 Hardware needed

Some basic hardware needed in developing this prototype. List of hardware need show as follow:

i. Type of device: Personal Computer

Processor: Pentium 233Mhz or above

Hard disk: greater than 1GB

RAM: 64MB or above

Cache Memory: 128KB

ii. Type of device: Bluetooth adapter

iii. Type of device: Mobile phone

3.5 Software needed

3.5.1 Specific software needed for the propose system development

i. Python

Programming language similar with Java , Perl and C++. Python is embedded in Linux Mint.

ii. Linux Mint

RedHat Linux Mint 5 is an enhanced and user-friendly desktop Linux distribution based on ubuntu 8.04

iii. Vmware

Vmware is use to running virtual os.

3.5.2 Specific package need in Linux Mint:

i. Ussp-push : Bluetooth push function and use to detect active Bluetooth device.

ii. SQLite3 : SQLite is a small C library that implements a self-contained SQL database engine. This article examines pysqlite, one of a number of libraries in Python, that is used as an interface with SQLite

iii. Bluez-utils : is a bluetooh package specific use for enable universal Bluetooth adapter.

3.6 Summary

This chapter is touch about the technology required to develop the propose system and why those technology is being choose. The next chapter will be mention about System design.

CHAPTER 4

SYSTEM DESIGN

4.1 Introduction

As state in the previous chapter, Unified Modeling Language (UML) that included in Rational Unified Process (RUP) will be use to represent the design for the application. Design phase will be separated into two phases that is Software Requirement Specification (SRS) and the Software Development Document (SDD). In SRS all the requirement gather will be transform into use cases. Actor will interact with the system and use case is the process or function that flow between the application and actor. Besides, Sequence diagram will produce for each of the use case diagram. Sequence diagram will use to demonstrate the behavior of the object in a use case. While SDD will focus on architecture design and database design.

4.2 Architecture design

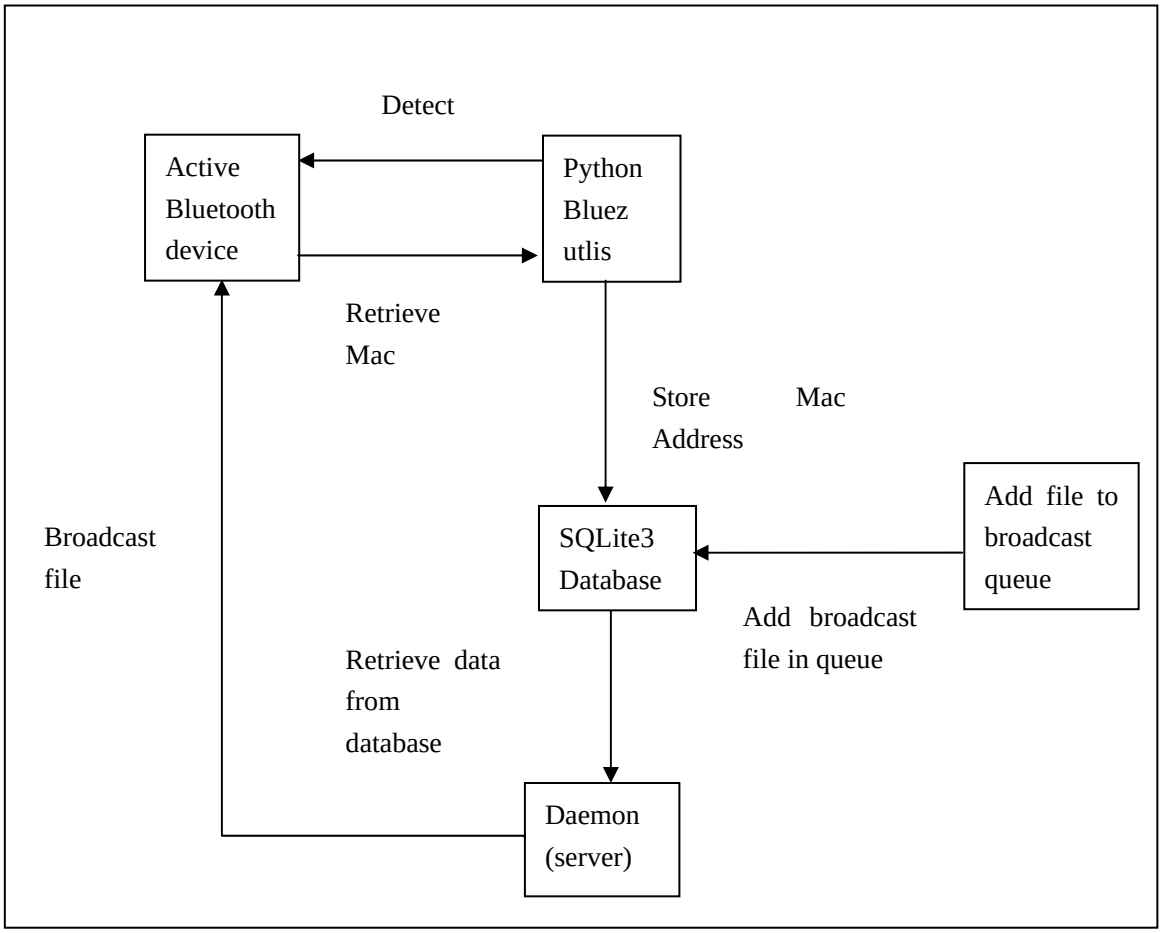


Figure 4.1: system architecture

Figure 4.1 Show the architecture design of the application. Firstly daemon will search for all the active Bluetooth devices around and store the Mac address into database. Secondly, Mac address will be compare with the Mac address in existing database to ensure that a single phone will not be broadcast twice. After that message or a file will be added to the queue waiting for broadcast and message will be broadcast when the daemon is activated.

4.3 System Modeling

Many different object-oriented modeling languages developed and led to the advent of UML. UML consists of a set of diagram use to represent the model of a system. This set of diagram include use case diagram, class diagram, object diagram, state diagram, sequence diagram, activity diagram, collaboration diagram, component diagram and deployment diagram. In this project, system will be model using use case diagram, sequence diagram and class diagram. Where Use Case Diagram is use to describe the functions of a system and its users, its include actor and use cases. While sequence diagram will represent the interaction between different objects in the system and class diagram is used to define the detail design of the system.

4.3.1 Use Case Diagram

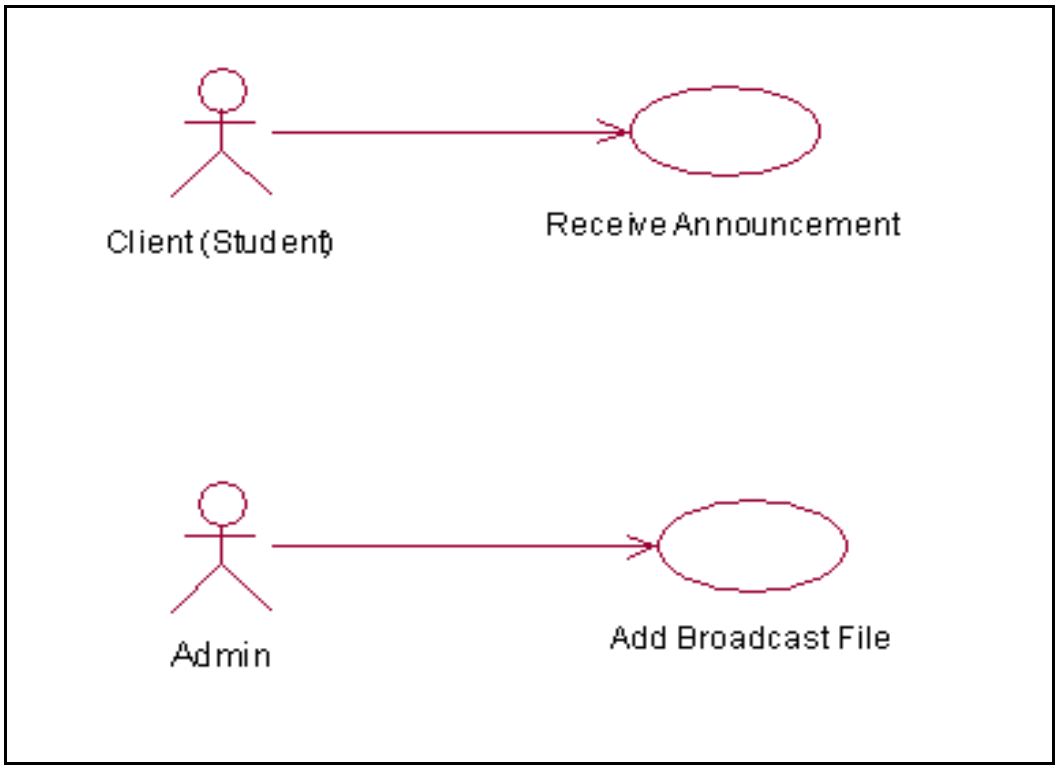


Figure 4.2: use case diagram for admin and user

Description about use cases will be discussed as below:

i. Receive Announcement:

This is process where user will receive general announcement from service node.

ii. Add broadcast file

Admin will insert new new broadcast file to the file system.

4.3.2 Actor Explanation

There are two actor involve in this system. That is FSKSM student and administrator. Both of the actor carry different role, function and accessibility.

i) Administrator

Administrator will be officer of FSKSM, which will manage the database. Admin can update, insert and delete data. Administrator will update the data inside the file system in Linux by adding or delete broadcast file in Linux.

ii) FSKSM Student

User in this system is FSKSM student, which they only need to switch on their mobile bluetooh then message will broadcast to their mobile phone.

4.3 Sequence Diagram of the System

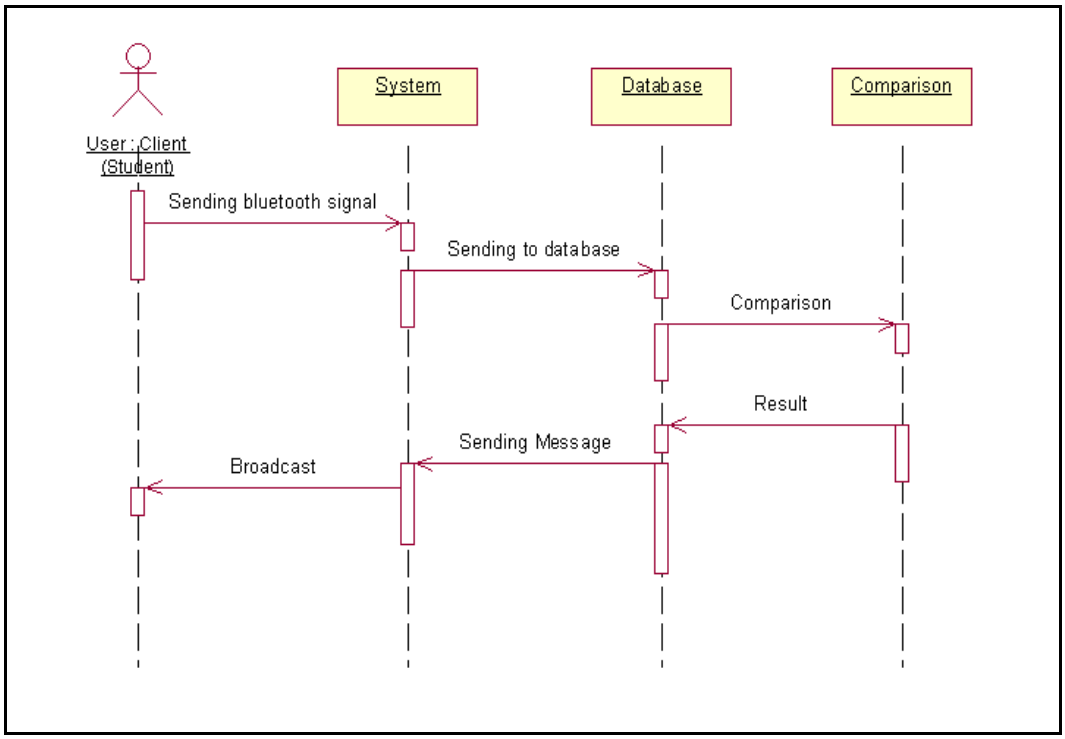


Figure 4.3: Sequence diagram for user to receive broadcast meesage

Once user enable Bluetooth on the mobile, Bluetooth signal will be send to the system, and Mac address will be retrieve and store in the database. After that a comparison process will carry on to make sure that this user not yet receive broadcast message, and the result will send back to the database and system will get the result from database and broadcast the message to the user.

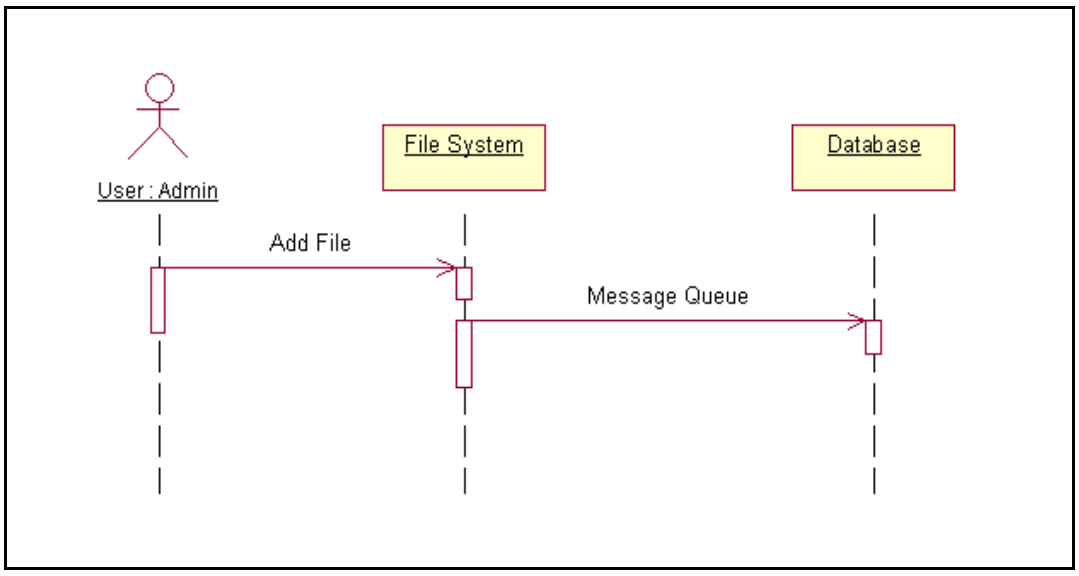


Figure 4.4: Sequence diagram for admin to add a broadcast file

In this process, admin will add the broadcast file in the file system in Linux.

4.4 Database design

Software selected to use in database development is Python SQLite3. The information included in database will be Active Bluetooth devices been detected.

Field name	Explanation	Type	Key Type
Mobile phone Mac address	Unique Id of a mobile phone	String	
Moblike phone name	Name set by mobile phone owner	String	

Table 4.1 Active Bluetooth device table

Active Bluetooth device table is store in Python SQLite3 database. When Bluetooth adapter search around for active Bluetooth devices, Mac address of active mobile phone will be collect and store in database. This database will be updated every 5 mins.

Field name	Explanation	Type	Key Type
receipient	Unique Id of a mobile phone or “*”	String	
Path_to_file	Path to the broadcast file	String	
date	Time message added to the database	Time	
end	time message will be expired	Time	

Table 4.2 Incoming_message

In this database, who will receive the broadcast message, the path of the file store, the date file is added in the message queue and when is the broadcast file will be discard is store in Incoming_message. Daemon will broadcast file according to this database.

Field name	Explanation	Type	Key Type
Message_id	Message id	int	
Recipients	Mac address of mobile pnone	String	
Date	Time message send	Time	
Status	File send or discard	int	

Table 4.3 Outgoing_message

This database is storing the data which recipients already broadcast with message, date message broadcast and the status of message.

4.5 Summary

This chapter is mention about the architecture design of the propose system and explain the use case and sequence diagram. Beside, database design also included. The next chapter will be mention about implementation of the system

Chapter 5

Implementation

5.1 Introduction

In this chapter, we will discuss about the implementation of the application. These included installation of the requirement software, and testing are debugging also carry out in this phase to make sure that this application can function well.

5.2 Development Environment

In this section, the installation of required package in Linux will be discussed.

5.2.1 Installation of required package in Linux

Package that need to be installed in developing the application was Python 2.5.2 , SQLite3, Sqlitebrower, ussp-push and bluez-utils.

Python 2.5.2 is programming language use to develop the application, Python is a powerful programming language which also support Bluetooth programming and Python is widely use in many programming field nowadays. SQLite3 is use as database backend, SQLite3 is choose because it is simple, fast and function well. Beside, Sqlitebroweser is a Linux programe which is use to create, edit and delete sqlite database. Ussp-push package is use to sending message to the mobile phone without any paring process, and active Bluetooth devices are discovered by hcitool. While, bluz-utils is a package which is needed for developing a Bluetooth application. This package is use for to detect most of the Bluetooth adapter sells in the market.

5.2.2 Installation of required software in windows

Software needed in windows are Rational Rose 2000, Microsoft Office 2007 and Microsoft project 2007. Rational Rose 2000 is used for design use case, and sequence diagram. While Microsoft Office 2007 is use for documentation of the application and Microso Project 2007 planning and generate Gantt chart for this project.

5.3 Coding

In this section, some important coding extract from the Linux script will discuss. Important codings included

1. Active Bluetooth devices searching
2. Comparing Mac address to ensure that each mobile phone will not send twice and broadcast message.
3. Add message to database.
4. Broadcast message

5.3.1 Active Bluetooth devices searching

In this script, a tools from ussp-push package will be call to search for the active Bluetooth devices around. After active devices is found and the Mac Address of the mobile is retrieve then will be straightly store in database. If the database file not found an error message will shown to the user. Last coding was the tools require for searching active Bluetooth devices and retrieve Mac address (hcitool scan). Coding to call the hcitools as follow:

```
DEVICE_CSV=$1

die()
{ printf "$@\n"
exit 1}
if [ ! -n "$1" ]
then
die "Expecting path to CSV file as first parameter"
fi
if [ ! -e `dirname $DEVICE_CSV` ]
then
die "Directory for storing CSV file does not exist.\n"
fi

hcitool scan | grep ":" | sed -e 's/\t\(.*\)\t\(.*\)/\1;\2/' > $DEVICE_CSV
```

Figure 5.1 Snippet coding for device searching active Bluetooth devices

5.3.2 Comparing Mac Address

The following coding is used to compare the new Mac address detected with the existing database. When a new Mac address is found then the Mac address will be store into a database file name btdWantTosend.csv. While the Mac address already exist in database previously will be discard and later after the Mac address already broadcast with file in database queue will be merge with existing database.

```
Fgrep -f /home/inamorato/Desktop/psm/data/dataHouse/dataMine.csv -v
/home/inamorato/Desktop/psm/data/bt-devices.csv >
/home/inamorato/Desktop/psm/data/btdWantTosend.csv

fgrep -f /home/inamorato/Desktop/psm/data/bt-devices.csv
/home/inamorato/Desktop/psm/data/dataHouse/dataMine.csv >
/home/inamorato/Desktop/psm/data/sameDevicesDetected.csv

cp /home/inamorato/Desktop/psm/data/btdWantTosend.csv
/home/inamorato/Desktop/psm/data/dataMerge/bt-devices3.csv

cp /home/inamorato/Desktop/psm/data/dataHouse/dataMine.csv
/home/inamorato/Desktop/psm/data/dataMerge/bt-devices2.csv

cat /home/inamorato/Desktop/psm/data/dataMerge/*.csv >
/home/inamorato/Desktop/psm/data/dataHouse/dataMine.csv
```

Figure 5.2 Snippet coding for compare and merge between two database

5.3.3 Add message to database

The following coding is use to add the message waiting queue in database. Firstly, application will check whether user is wish to broadcast file to all active Bluetooth devices or specific Bluetooth devices. Secondly, application will check whether the existence of file in the storage area, if the file not exit a error message will be raise. Finally, broadcast file will be added to the message queue.

```
    def add_broadcasted_file(self, path_to_file, recipient, expires_in_seconds = 0, move = 0):
                if recipient != "*":
                        p = re.compile('^([0-9a-f]{2}([:]|$)){6}$', re.IGNORECASE)

                        if p.match(recipient) == None:
                                raise Exception("Recipient can be '*' or a MAC address like '88:99:00:AA:BB:CC'")

            # move file to storage dir
                filename = os.path.basename(path_to_file)

                if os.path.exists(path_to_file) != 1:
                        raise Exception("File '" + path_to_file + "' does not exist")

                target_dir = self.config.get("data", "storage_dir")
                target_file =  target_dir + "/" + filename

                if os.path.exists(target_file) != 1:
                        raise Exception("Could not copy file '" + path_to_file + "' to '" + target_file + "'")

                # mask "ends" as NULL
                ends = None
                if expires_in_seconds > 0:
                        ends = datetime.datetime.fromtimestamp((time.time() + expires_in_seconds))

                cursor = self.conn.cursor()
                cursor.execute('INSERT INTO message_incoming (recipient, filename, date, ends) VALUES(?,?,?,?)', (recipient, filename, datetime.datetime.now(), ends))
                self.conn.commit()
```

Figure 5.3 Snippet coding for adding message into database

5.3.4 Broadcast message

```
def find_outstanding_messages(self):
        r = []

        rows = self.find_new_assigned_messages()

        for row in rows:
            if self.is_message_send(row[0]) == 0:
                r.append(row)
        return r
```

Figure 5.4 Snippet coding for finding outstanding message from database

Broadcast message is take place in SendWorker.py. Firstly, outstanding message or message waiting in message queue will be check.

```
def find_new_assigned_messages(self):
        cursor = self.conn.cursor()
        cursor.execute('SELECT rowid, filename FROM message_incoming WHERE ( ( (ends IS NULL) OR (julianday(ends) > julianday("now")) ) AND ( (recipient = "*") OR (recipient = ?) ) )', (self.mac,))
        return cursor
```

Figure 5.5 Snippet coding for retrieve message from database

Secondly, outstanding message will be retrieve from database

```
add_message_to_queue(self, message_id):
        cursor = self.conn.cursor()
        cursor.execute('INSERT INTO message_outgoing (message_incoming_id, recipient, date_try) VALUES(?,?,?)',(message_id, self.mac, datetime.datetime.now()))
        self.conn.commit()
        return cursor.lastrowid

   def set_status_of_message(self, id, is_accepted):
            self.dbg("Accepted message id '" + str(id) + "' to '" + self.mac + "': " + str(is_accepted))

            cursor = self.conn.cursor()
            cursor.execute('UPDATE message_outgoing SET date_finished = ?, is_accepted = ? WHERE rowid = ?', (datetime.datetime.now(), is_accepted, id))
            self.conn.commit()
```

Figure 5.6 Snippet coding for update the outgoing_message database

Table message_outgoing in database will be update.

```
def send_file_to_adapter(self, filename):
            path_to_file = self.config.get("data","storage_dir") + "/" + filename

            cmd = self.config.get("exec", "send_command")
            cmd = cmd.replace("$mac", self.mac)
            cmd = cmd.replace("$path_to_file", path_to_file)
            self.dbg("Executing '" + cmd + "': ")

            return os.system(cmd)
```

. **Figure 5.7** Snippet coding for broadcast message

And finally message will be send to the mobile phone.

5.4 Summary

As conclusion, this chapter discussed the implementation of the system. Implementation can be described as the realization system from the design and analysis phrases.

Chapter 6

Conclusion

6.1 Introduction

In this chapter we will come to the summary of this report. The existence of technology is to help and reduce the burden of human and so this application. Between, fast growth of wireless technology and high demand of Bluetooth application also support the existence of high technology application.

6.2 Achievements

Bluetooth announcement system is a system that acts as a media to broadcast all necessary announcements to the student in FSKSM. Throughput the

development process, Bluetooth push is implemented in order to broadcast message to mobile phone. By using usssp-push pairing is not required, means that user of mobile phone is no need to enter password for accepting the announcement. Once the application is activated, active Bluetooth devices will be detected and store inside database and will be broadcast with the announcement in the storage area. And using this application, a mobile phone will not be send twice event it is detected.

6.3 Limitation and Challenges

There is no hundred percent perfect system in the world, this system also cannot be excluded. The main problem of this system is that, this system is lack of User interface, which is not really convenience to admin. Admin need to key in command in Linux terminal.

Bluetooth service node cannot simply set up to send a wide variety of file types and hope that others will be able to receive the information. Thus, this inflexibility makes the usage areas for the Bluetooth broadcasting somewhat limited.

Besides, users don’t want to wait while their mobile device receives data. Having to wait for a connection request is even worst. Because most mobile phones and PDAs on the market today contain Bluetooth hardware that still adheres to the older Bluetooth 1.1 standard, there is no specific support for IEEE 802.11 and Bluetooth coexistence. Hence, this will be a problem during the implementation of

the system.

Furthermore, it’s a better should have intelligent message filtering. Nothing us more annoying for a user of system as receive unwanted information from time to time. Hence there should be come out with a intelligent message filter in the server to make sure that the information user receive is relevant to them.

Future expectation about this system will be security issues with the broadcast. For example, the system should prevent hacker to hacks the system through the Bluetooth connection or it may possible for attacker to send harmful files to people who are expecting a file from the service node. Hence a more secure security need to be produce.

6.4 Expectation

Expectation for this application is that hoping this application can fully function and fulfill the system requirement and scope. So that this application can help increase the efficiency of announcement system in FSKSM.

Besides, this application is also suitable to use in restaurant, supermarket and etc to broadcast their promotion to customer. For example, when a customer come in to a restaurant, he will receive the information for meal promotion in the week in .jpg format.

6.5 Summary

At the end of this project, an Bluetooth base application will be develop and trying to make it user friendly and implement in FSKSM in the future. Hope by using this application, some of the announcement will be solve. Hence, this is also telling us that Bluetooth is capable to use for broadcast to a group of people instead of transferring a file to a single Bluetooth device. Industry of Bluetooth is growing fast and is an evolution to the connectivity technology.

Reference

1. L. Aalto, N. G¨othlin, J. Korhonen, and T. Ojala. Bluetooth and wap push based location-aware mobile advertising system. In MobiSys ’04

2. G. Antoniou and F. van Harmelen. Web ontology language: Owl. In S. Staab and R. Studer, editors, Handbook on Ontologies in Information Systems. Springer-Verlag, 2003.

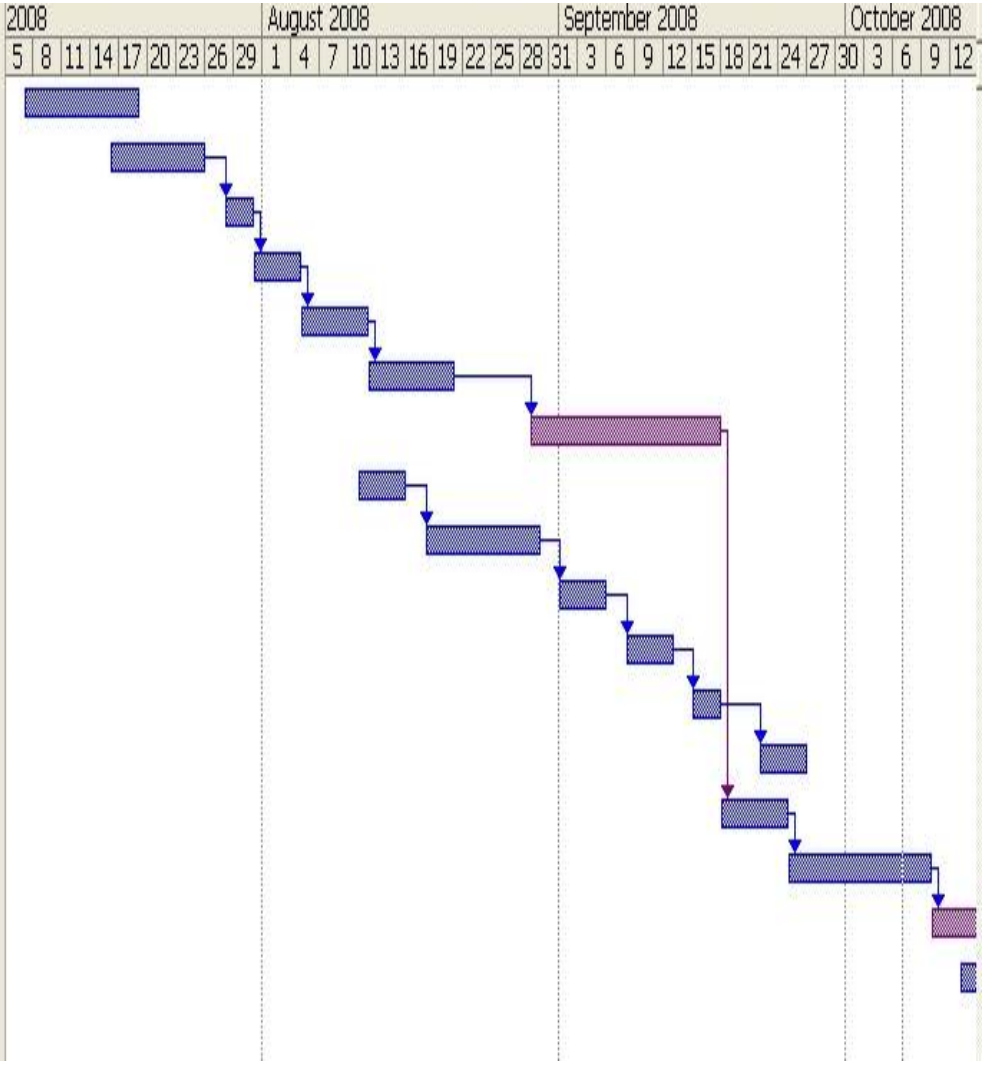
3. Mobile advertising case study of mindmatics and 12snap lokomobil, Hoang Nga Luong, 2007

4. Bluetooth and WAP Push Based Location_Aware Mobile Advertisng System, Lauri Aalto, Nicklas Gothlin, Jani Korhinen, Tino Ojala

Appendix A

Ghantt Chart PSM1

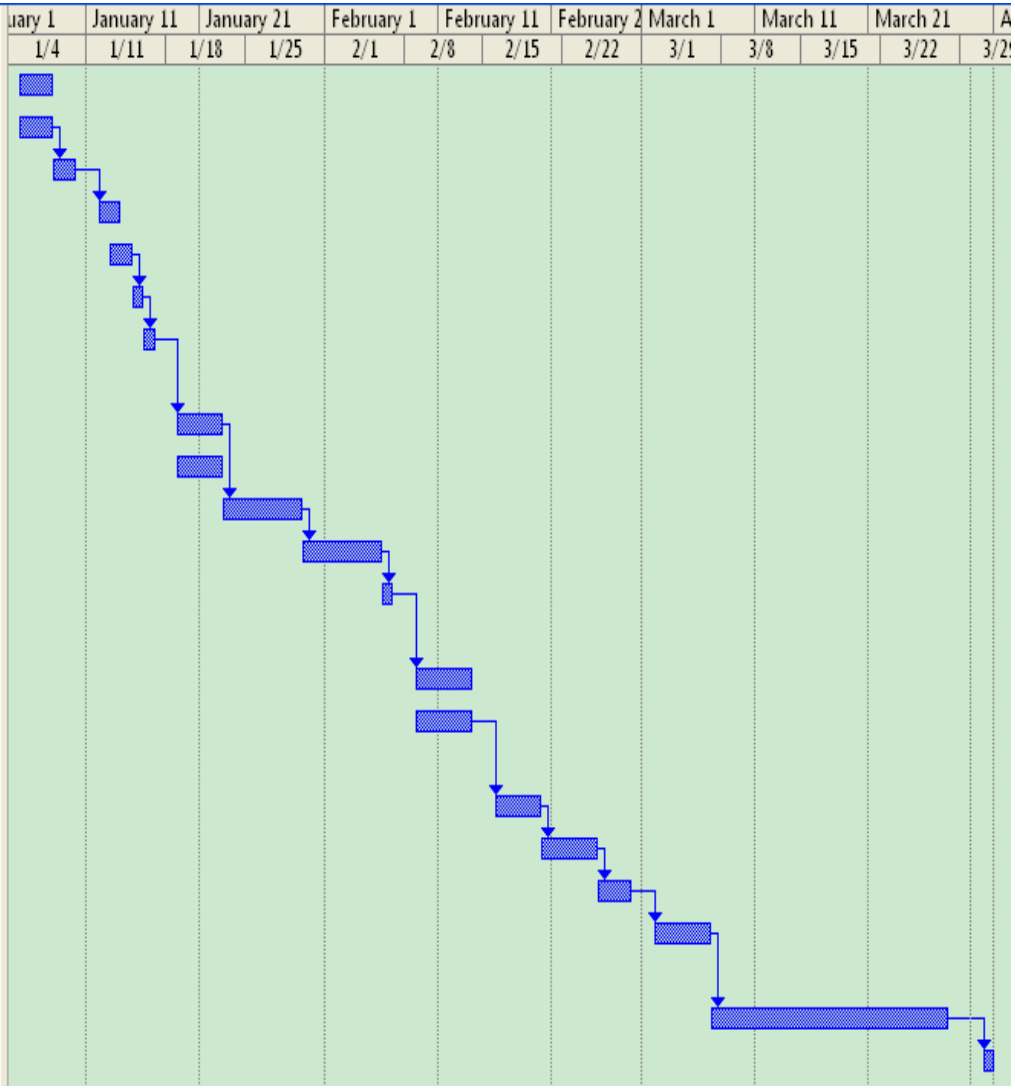
	Task Name	Start	Finish	Duration	Predi
1	Tajuk dan pemilihan Penyelia	Mon 7/7/08	Fri 7/18/08	10 days	
2	Cadangan dan penentuan Tajuk Project	Wed 7/16/08	Fri 7/25/08	8 days?	
3	Menunggu Pengesahan Tajuk	Mon 7/28/08	Wed 7/30/08	3 days?	2
4	Bincang Tajuk Dengan Penyelia	Thu 7/31/08	Mon 8/4/08	3 days?	3
5	Kajian Tentang Sistem Metodologi	Tue 8/5/08	Mon 8/11/08	5 days?	4
6	Kumpul maklumat Tentang Projek	Tue 8/12/08	Wed 8/20/08	7 days?	5
7	Penyedian Laporan	Fri 8/29/08	Wed 9/17/08	14 days?	6
8	Bab 1 Pengenalan	Mon 8/11/08	Fri 8/15/08	5 days?	
9	Bab 2 Kajian Literatur	Mon 8/18/08	Fri 8/29/08	10 days?	8
10	Bab 3 Metodologi	Mon 9/1/08	Fri 9/5/08	5 days?	9
11	Bab 4 Rekabentuk	Mon 9/8/08	Fri 9/12/08	5 days?	10
12	Bab 5 Implimentasi dan pengujian	Mon 9/15/08	Wed 9/17/08	3 days?	11
13	Hantar dan penyemakan Draf Laporan	Mon 9/22/08	Fri 9/26/08	5 days?	12
14	Pembetulan Draf Laporan	Thu 9/18/08	Wed 9/24/08	5 days?	7
15	pembetulan Laporan	Thu 9/25/08	Thu 10/9/08	11 days?	14
16	Penyerahan Laporan Lengkap Projek	Fri 10/10/08	Wed 10/15/08	4 days?	15
17	Pembentangan Projek	Mon 10/13/08	Thu 10/16/08	4 days?	



Appendix B

Ghantt Chart PSM2

	Task Name	Duration	Start	Finish
1	Install require software	3 days	Mon 1/5/09	Wed 1/7/09
2	install linux package	3 days	Mon 1/5/09	Wed 1/7/09
3	Finding python tutorial	2 days	Thu 1/8/09	Fri 1/9/09
4	studying the python tutorial	2 days	Mon 1/12/09	Tue 1/13/09
5	implement python command	2 days	Tue 1/13/09	Wed 1/14/09
6	finding linux bluetooth programming	1 day	Thu 1/15/09	Thu 1/15/09
7	implement python bluetooth programming	1 day	Fri 1/16/09	Fri 1/16/09
8				
9	programming linux ussh push script	4 days	Mon 1/19/09	Thu 1/22/09
10	coding hcitool to scan bluetooth devices	4 days	Mon 1/19/09	Thu 1/22/09
11	develop database	5 days	Fri 1/23/09	Thu 1/29/09
12	coding to compare to databse	5 days	Fri 1/30/09	Thu 2/5/09
13	combining all the module	1 day	Fri 2/6/09	Fri 2/6/09
14				
15	testing cronbtab scheduling	5 days	Mon 2/9/09	Fri 2/13/09
16	install crontab service	5 days	Mon 2/9/09	Fri 2/13/09
17				
18	System Testing	4 days	Mon 2/16/09	Thu 2/19/09
19	Improve The application Performance	3 days	Fri 2/20/09	Tue 2/24/09
20	Improve the application Performance	3 days	Wed 2/25/09	Fri 2/27/09
21	Add and Improve the application	5 days	Mon 3/2/09	Fri 3/6/09
22				
23	Document report psm2	16 days	Sat 3/7/09	Fri 3/27/09
24	Submission of Final Project Report	1 day	Tue 3/31/09	Tue 3/31/09



Appendix c

User Manual

Installation of Bluetooth announcement system in Linux Mint:

1.0 Requirement:

(1) Python2.4 or Python 2.5 (normally Linux Mint already embedded with Python)
(2) Sqlite3 binding fpr Python
(3) A working bluetooh environment on machine (hcitools and sdptool must be installed)
(4) Ussp-push must be install

1.1 Installation

(1) Logon into Linux Mint

Figure 1c. Linux login interface

(2) Open terminal in Linux Mint

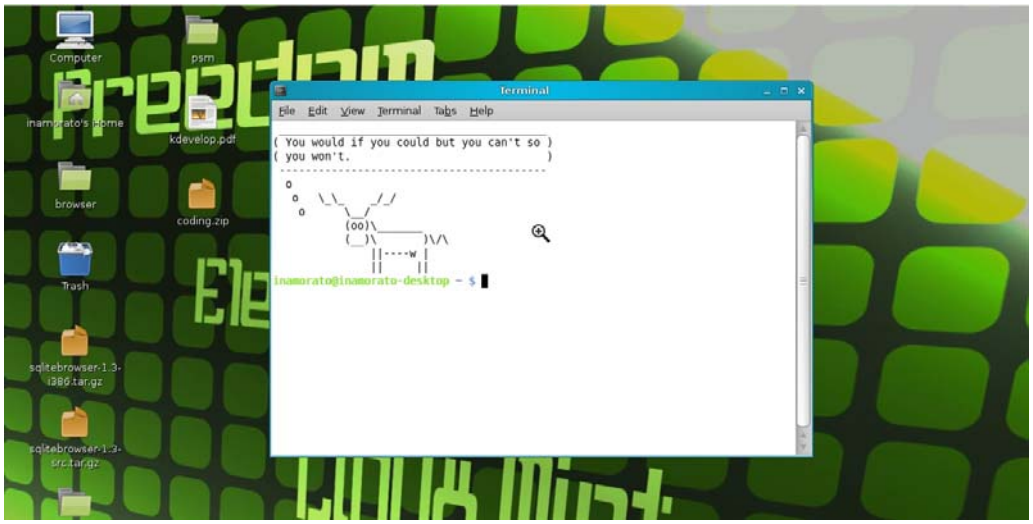


Figure 2c. Linux terminal

(3) Type in terminal: sudo apt-get install Python-pysqlite1.1

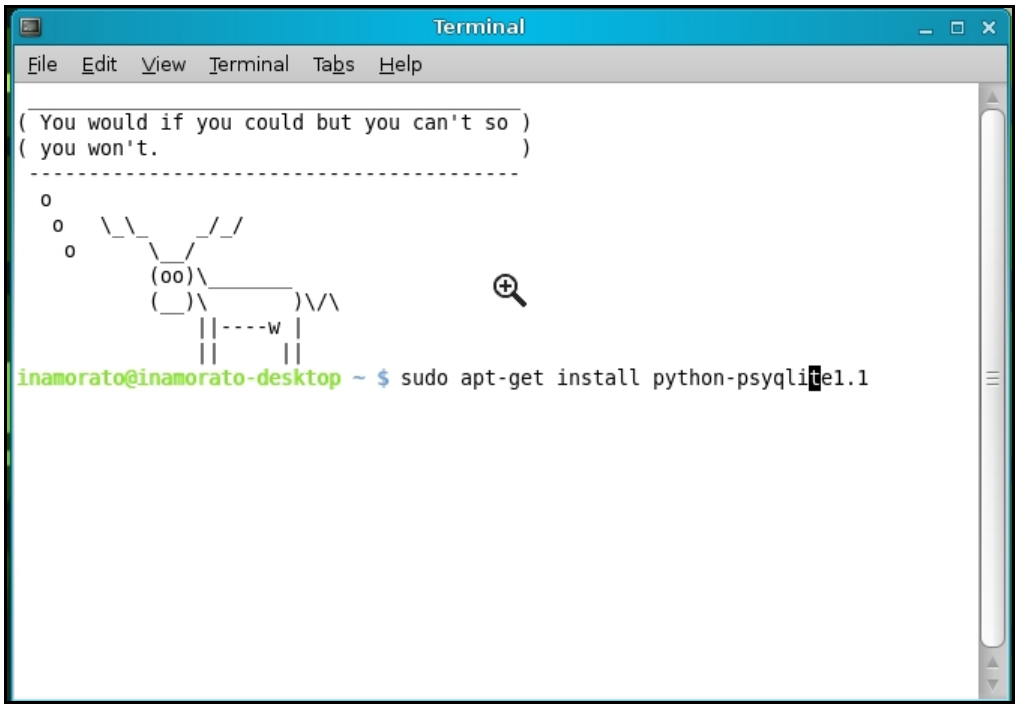


Figure 3c. Linux install Python-psyqlite1.1

(4) Type in terminal : sudo apt-get install bluez-utils

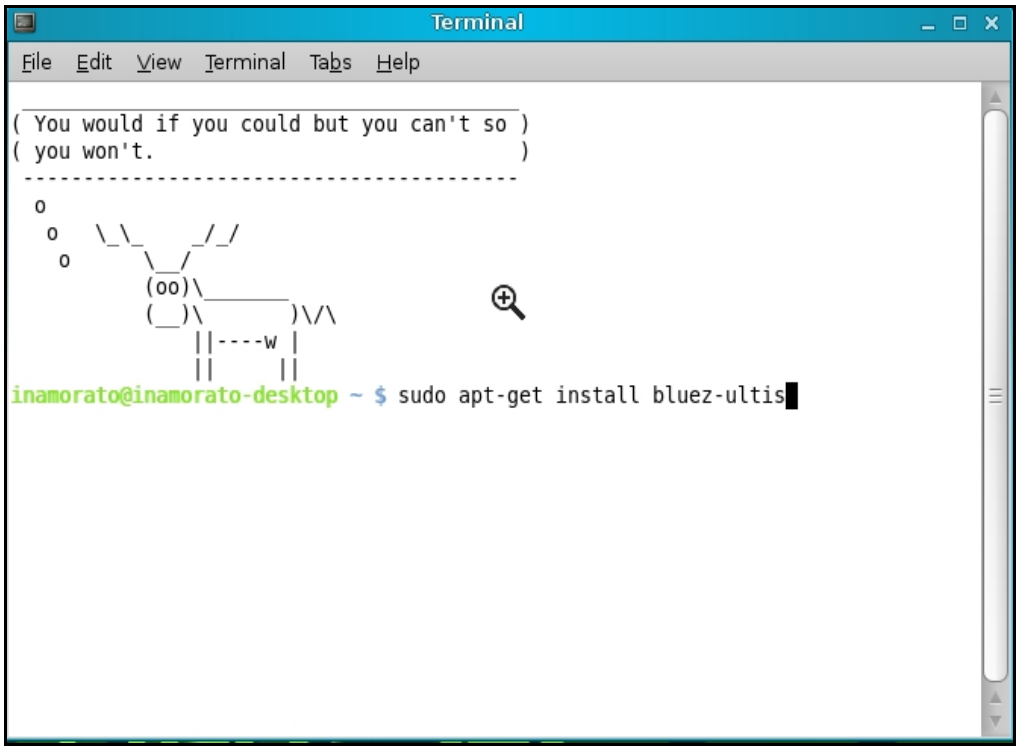


Figure 4c. Linux install bluez-utils

(5) Type in terminal: sudo apt-get install Obexftp ussp-push

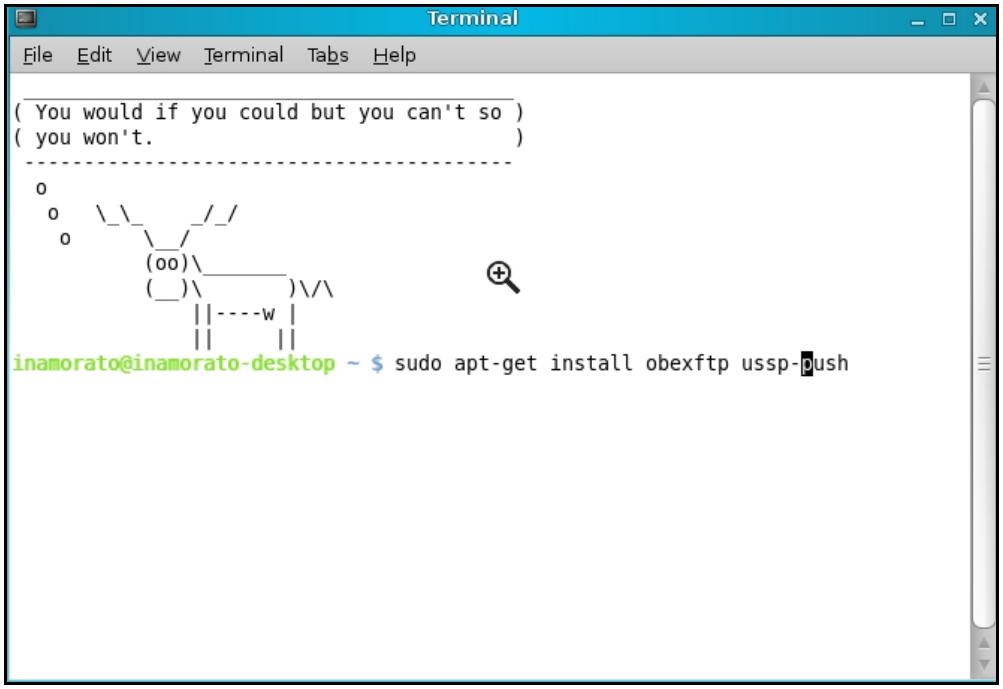


Figure 5c. Linux install Obexftp and ussp-push

All the three package will be install once enter is press. The following print screen show the succeed install package.

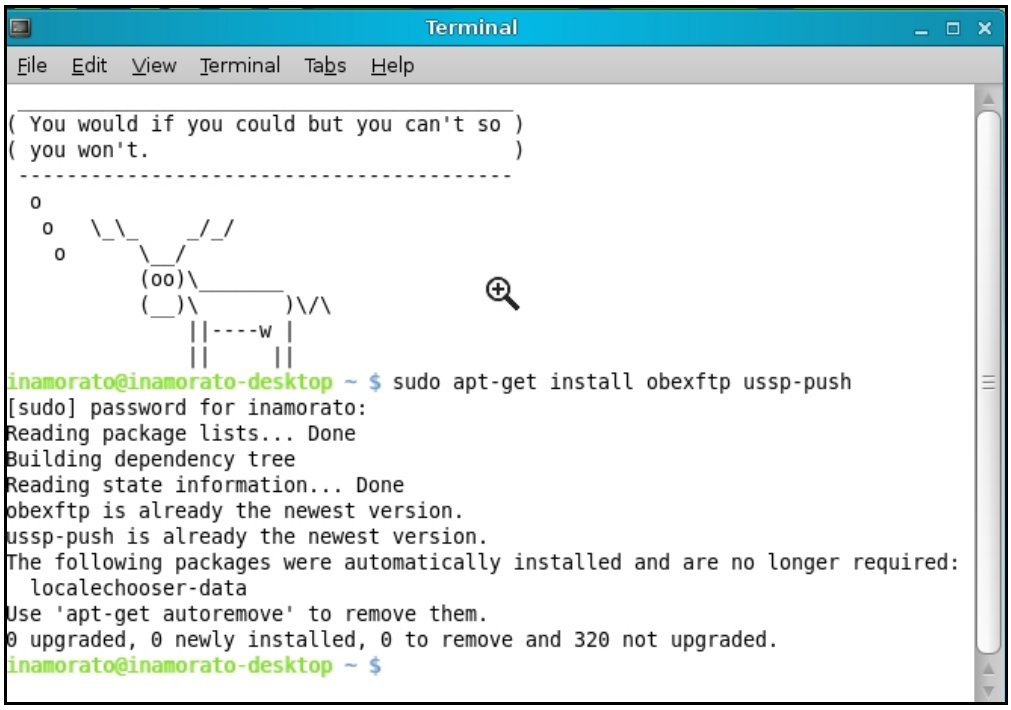


Figure 6c. Package install successful

(6) Copy the whole system folder a directory

(7) Move the path to the directory: for example directory is /home/inamorato/Desktop/psm

cd $INSTALL_DIR

cd /home/inamorato/Desktop/psm

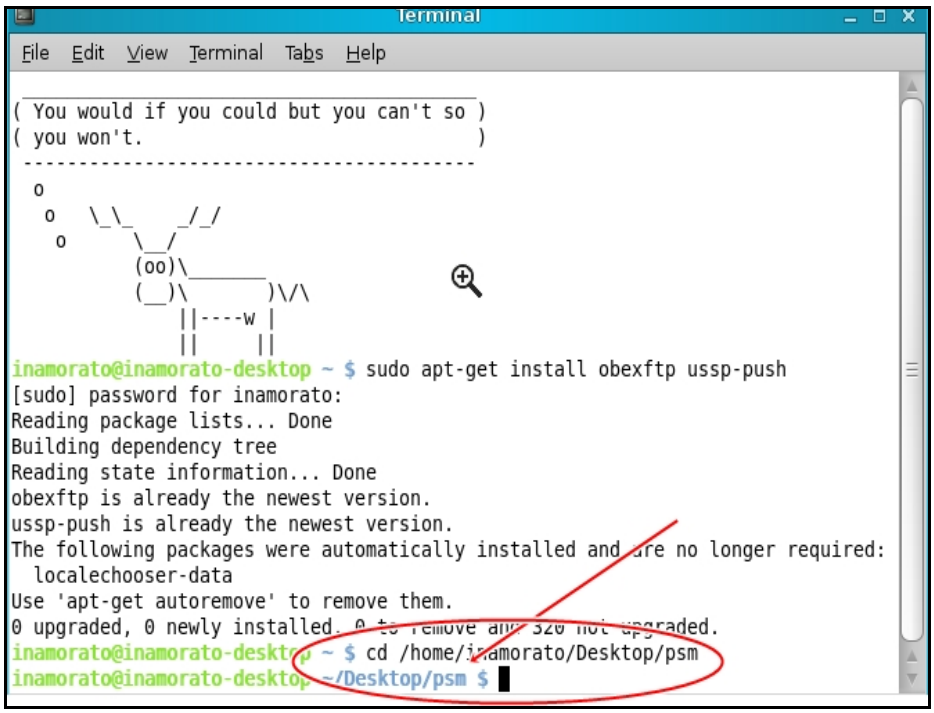


Figure 7c. move to the install directory

(8) Type in terminal:

chmod o+x execute-server.py add-message.py scripts/find-Bluetooth.sh scripts/send-ussp.sh scripts/send-Obexftp.sh

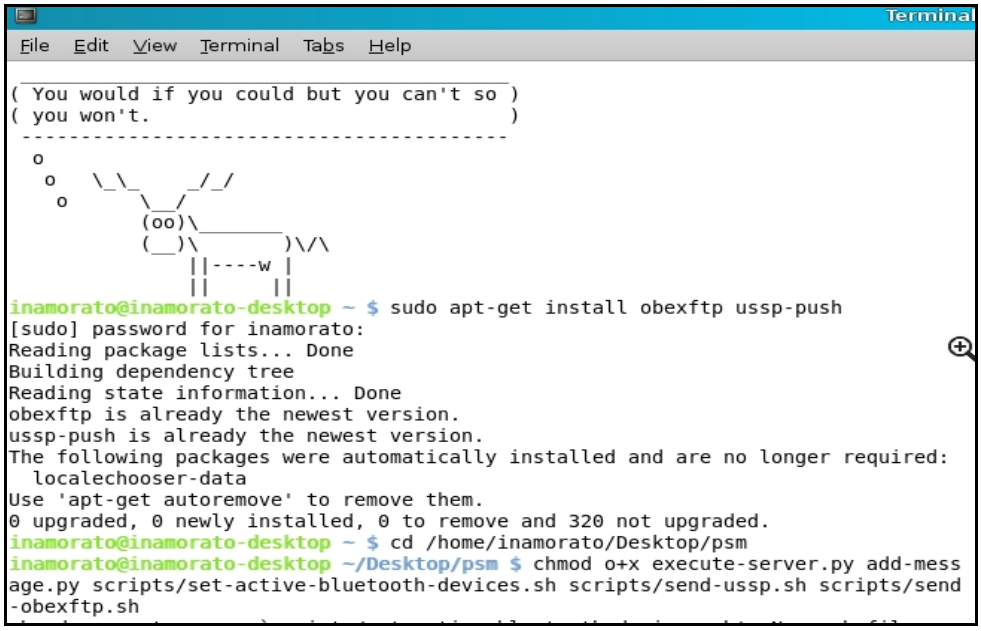


Figure 8c. Figure show command executing chmod

(9) Type in terminal: crontab –e

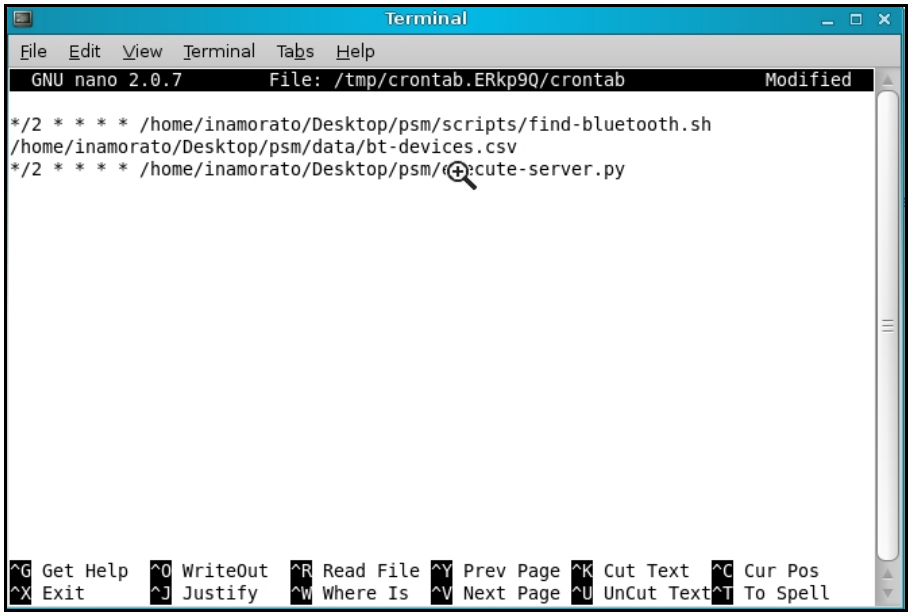


Figure 9c. crontab scheduling task

And type in the command as above print screen.

Admin use of application:

(1) Add broadcast file in install_dir/storage

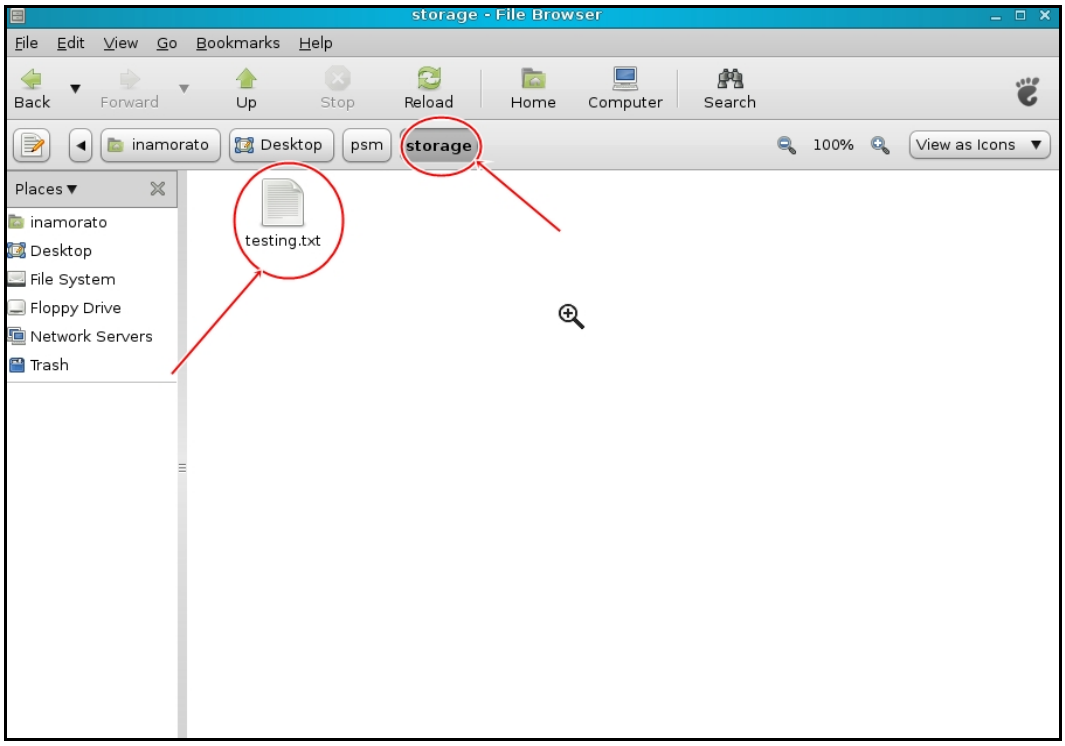


Figure 10c. Add file to the storage directory

(2) Type in terminal : $INSTALL_DIR/add-message.py -r “*” -f /tmp/testing.txt

(3) Type in terminal: $INSTALL_DIR/execute-server.py

(4) File will be broadcast

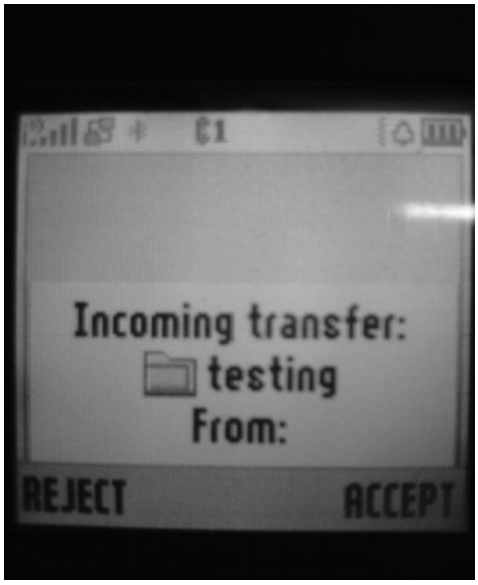


Figure 11c. File receive