



Faculty of Computer Science and Information Technology

***ANDROID BASED TEXT MESSAGING FOR OPPORTUNISTIC NETWORK***

Lo Ka Hing

Bachelor of Computer Science with Honours  
(Network Computing)  
2014

**ANDROID BASED TEXT I**

**P. KHIDMAT MAKLUMAT AKADEMIK  
UNIMAS**

**PPORTUNISTIC NETWORK**



**1000288592**

**LO KA HING**

**This project is submitted in partial fulfilment of the  
requirements for the degree of  
Bachelor of Computer Science with Honours  
(Network Computing)**

**Faculty of Computer Science and Information Technology**

**UNIVERSITI MALAYSIA SARAWAK**

**2014**

# **APLIKASI BERMESEJ-TEKS ANDROID UNTUK RANGKAIAN OPORTUNIS**

**LO KA HING**

**Projek ini merupakan salah satu keperluan untuk  
Ijazah Sarjana Muda Sains Komputer dengan Kepujian  
(Pengkomputeraan Rangkaian)**

**Fakulti Sains Komputer dan Teknologi Maklumat**

**UNIVERSITI MALAYSIA SARAWAK**

**2014**

UNIVERSITI MALAYSIA SARAWAK

THESIS STATUS ENDORSEMENT FORM

ANDROID BASED TEXT MESSAGING FOR OPPORTUNISTIC NETWORK

ACADEMIC SESSION: 2013/2014

LO KA HING

hereby agree that this Thesis\* shall be kept at the Centre for Academic Information Services, Universiti Malaysia Sarawak, subject to the following terms and conditions:

1. The Thesis is solely owned by Universiti Malaysia Sarawak
2. The Centre for Academic Information Services is given full rights to produce copies for educational purposes only
3. The Centre for Academic Information Services is given full rights to do digitization in order to develop local content database
4. The Centre for Academic Information Services is given full rights to produce copies of this Thesis as part of its exchange item program between Higher Learning Institutions [ or for the purpose of interlibrary loan between HLI ]
5. \*\* Please tick ( √ )

CONFIDENTIAL (Contains classified information bounded by the OFFICIAL SECRETS ACT 1972)

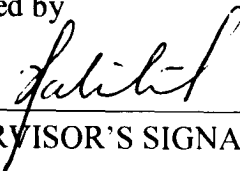
RESTRICTED (Contains restricted information as dictated by the body or organization where the research was conducted)

UNRESTRICTED



(AUTHOR'S SIGNATURE)

Validated by



(SUPERVISOR'S SIGNATURE)

Permanent Address:  
1853, Jalan 18/44,  
Taman Sri Serdang,  
43300 Seri Kembangan,  
Selangor

Date:

18-6-2014

Date:

18-06-2014

Note \* Thesis refers to PhD, Master, and Bachelor Degree

\*\* For Confidential or Restricted materials, please attach relevant documents from relevant organizations / authorities

## **Acknowledgement**

I would like to thanks to my final year project supervisor, Dr. Halikul bin Lenando for his guidance towards my project. Besides that, thanks to Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak for giving a chance to conduct final year project which will generate a great value in future. Thanks to all my friends and those who involved directly or indirectly in this final year project.

## **Table of Contents**

Acknowledgement.....	i
List of Figures .....	vi
List of Tables.....	viii
Abstract .....	ix
Abstrak .....	x
Chapter 1: Introduction .....	1
1.1 Overview .....	1
1.2 Problem Statement.....	2
1.3 Objectives .....	3
1.4 Brief methodology .....	3
1.5 Scope .....	4
1.6 Significant of Project.....	4
1.7 Project Outline.....	5
Chapter 2: Literature Reviews.....	6
2.1 Introduction .....	6
2.2 Android.....	6
2.3 Text Messaging.....	7
2.4 Opportunistic Network .....	8
2.5 Existing Communication Applications.....	10
2.5.1 WhatsApp.....	10

2.5.2	LINE.....	11
2.5.3	WeChat.....	12
2.5.4	Windows Live Messenger.....	12
2.5.5	ONEChat.....	13
2.5.6	Comparison of Text Messaging Applications.....	14
2.6	Summary.....	16
<b>Chapter 3: Methodology.....</b>		<b>17</b>
3.1	Introduction.....	17
3.2	Agile Development Methodology.....	17
3.3	Requirement Analysis.....	18
3.3.1	Hardware Requirement.....	20
3.3.2	Software Requirement.....	21
3.4	Designs.....	22
3.5	Summary.....	28
<b>Chapter 4: Implementation and Testing.....</b>		<b>29</b>
4.1	Introduction.....	29
4.2	Installation of Development Tools.....	29
4.3	Application's Implementation.....	29
4.3.1	Discovery Stage.....	30
4.3.2	Connection Establishment Stage.....	34
4.3.3	Data Transmission Stage.....	40

4.4.4	Termination Stage .....	44
4.4.5	Application's Local Storage.....	44
4.4	Application Testing .....	47
4.4.1	Functionality Test.....	47
4.4.2	Usability Test .....	56
4.4.3	Evaluation.....	57
4.5	Summary.....	57
Chapter 5: Conclusion and Future Works .....		58
5.1	Introduction .....	58
5.2	Objectives Achieved.....	58
5.3	Limitations.....	59
5.4	Future Works .....	60
5.5	Conclusion.....	61
References .....		62
Appendices .....		65
Appendix A: OppChat's Algorithm .....		65
Appendix B: OppChat.java Pseudocode .....		66
Appendix C: OppChatService.java Pseudocode .....		67
Appendix D: DeviceListActivity.java Pseudocode.....		69
Appendix E: methods in DatabaseAdapter.java .....		70
Appendix F: OppChat.java.....		71



Appendix G: OppChatService.java .....	77
Appendix H: DeviceListActivity.java .....	86
Appendix I: DatabaseAdapter.java.....	89

## List of Figures

Figure 3-1: Waterfall Model .....	17
Figure 3-2: Agile Development Methodology .....	18
Figure 3-3: Structure of a Node.....	22
Figure 3-4: OppChat's Messaging Life Cycle .....	23
Figure 3-5: OppChat's Use Case.....	24
Figure 3-6: OppChat's Activity Flow .....	25
Figure 3-7: Data Forwarding in OppNet.....	26
Figure 3-8: Database of OppChat.....	28
Figure 4-1: OppChat.java 's onCreate() method and onStart() method .....	31
Figure 4-2: ensureDiscoverable() method in OppChat.java.....	32
Figure 4-3: doDiscovery() method in DeviceListActivity.java .....	33
Figure 4-4: BroadcastReceiver in DeviceListActivity.java .....	33
Figure 4-5: connectDevice() method in OppChat.java .....	34
Figure 4-6: connect() method in OppChatService.java.....	35
Figure 4-7: constructor of ConnectThread in OppChatService.java.....	36
Figure 4-8: run() method in ConnectThread .....	37
Figure 4-9: run() method in AcceptThread .....	39
Figure 4-10: sendMessage() method in OppChat.java.....	40
Figure 4-11: run() method in ConnectedThread().....	41
Figure 4-12: else case of run() method in ConnectedThread .....	43
Figure 4-13: DbHelper class in DatabaseAdapter.java .....	45
Figure 4-14: Skeleton of DatabaseAdapter.java .....	46

Figure 4-15: A customized layout (with four buttons at bottom) for testing purpose .....	47
Figure 4-16: Bluetooth permission request .....	48
Figure 4-17: Action Bar of the OppChat.....	48
Figure 4-18: List of available devices .....	49
Figure 4-19: Bluetooth discoverable permission dialog .....	49
Figure 4-20: Brief information about the application .....	50
Figure 4-21: Status of the application on device.....	50
Figure 4-22: Conversation between Device A and Device B .....	51
Figure 4-23: Connection attempt failed .....	51
Figure 4-24: Message is stored.....	52
Figure 4-25: Message from out-of-ranged device.....	52
Figure 4-26: Terminated connection.....	53

## **List of Tables**

Table 1-1: Comparison of Text Messaging Applications .....	15
Table 4-1: Functionality Testing Results .....	53

## Abstract

*Android operating system is dominating the mobile device operating system. There are many text messaging applications are available to download from Google Play Store, and install on the device. The existence of text messaging applications is slowly replacing the usage of Short Message Service (SMS) as the applications can send and receive text messages without any charges. However, these text messaging applications are needed to connect to Internet in order to send and receive messages. As today's smartphones are able to create and join any wireless network, opportunistic network that involve a great mobility nodes will be able to reduce its difficulty to implement. Opportunistic network doesn't have a persistent end-to-end path between the source and destination. The data will be exchanged and transmitted by the opportunistic contact between intermediate nodes. In this project, an Android-based text messaging application is designed and developed, and expected to work in opportunistic network.*

## Abstrak

Sistem operasi Android semakin banyak digunakan sebagai sistem operasi kepada alat-alat bermudah alih. Banyak aplikasi bermesej-teks boleh dimuat turun dari Google Play Store dan dipasangkan dalam alat mudah alih. Kewujudan aplikasi bermesej-teks ini sedang menggantikan kegunaan servis bermesej-pendek (SMS) dengan secara tidak langsung. Hal sedemikian kerana kegunaan aplikasi bermesej-teks tidak dikenakan sebarang kos. Walau bagaimanapun, aplikasi-aplikasi bermesej-teks ini perlu disambungkan ke Internet supaya dapat mengirim dan menerima mesej. Memandangkan alat-alat bermudah alih pada hari ini mempunyai kebolehan mencipta dan menyambung kepada rangkaian tidak-berwayar, maka pelaksanaan rangkaian beroportunis (*Opportunistic Network*) yang amat memerlukan peralihan, boleh dipermudahkan. Rangkaian beroportunis tidak mempunyai perhubungan secara langsung antara nod sumber dengan nod destinasi yang berkekalan. Jadi, penghantaran data-data akan dilakukan melalui perhubungan nod-nod yang berantara sumber dengan destinasi. Dalam projek ini, aplikasi bermesej-teks yang berdasarkan sistem operasi Android akan direkakan dan diciptakan, supaya dapat digunakan dalam rangkaian beroportunis.

# **Chapter 1: Introduction**

## **1.1 Overview**

Android-based applications are common to everyone with the existence of smartphone. Social messaging applications like WeChat, WhatsApp and Line are famous as these are the top free downloaded Android social messaging apps from Google Play (Google Play Store, 2013) and been claimed as the best Android chat and messenger apps (Russell, 2013). These applications can send/receive messages to/from friends, which like Short Message Service (SMS).

However, all these applications are required internet connection such as Wi-Fi or phone's existing data plan. This project is aim to develop an application that based on opportunistic network environment, where a message able to send and receive from one to another. Opportunistic network may not have an end-to-end path between source and the destination. In an opportunistic network, there are several small networks scattered at different regions and an end-to-end path may only exist for a short period of time and unpredictable. If one of the intermediate nodes between the sender and receiver are not available, the message is yet able to reach to destination by opportunistic contact with another node, among the nodes from different regions. In this challenging environment, the main issue will be the routing and forwarding of the content. There are not many applications implement this type of network, not even Android's applications. As there are many smartphone users are using Android OS, the implementation of proposed application in the opportunistic network will be more suitable as increasing the number of mobile nodes.

## 1.2 Problem Statement

Today, existing text messaging applications are dependent on the Internet in order to send or receive message. Sometimes, users may find themselves in rural or urban areas where poor in communication coverage. Hence, they may not able to communicate by using the current text messaging. As today's mobile devices are able to support ad-hoc network with high mobility, by implementing the concept of opportunistic network, text messaging need not connect internet to send message to recipients. The message from sender may distribute to a mobile node that has the opportunity to meet with the receiver. Implementation of opportunistic network on text messaging may use the carry-store-forward fashion to send a message to recipient. But, the drawback of this implementation is causing a delay of the message been delivered to recipient compare with the other approach.

On the other hand, there are many Android text messaging (or social messaging) applications, in order to keep attract the users to continue using the applications, many features had been added and causing the applications' size larger and may not preferred by who would like to keep it in easy and simple way. Besides that, users may receive ads or activity announcements, which is annoying. Hence, in this project, the developed application will only has the basic functions like send and receive text message.



### **1.3 Objectives**

The objectives of this project outline as follows:

- To design a text messaging application that suit in opportunistic network environment.
- To develop an android-based text messaging application for opportunistic network.
- To investigate the text messaging implementation in opportunistic network.

### **1.4 Brief methodology**

In order to carry out the project, an agile approach is used throughout the whole project. Agile development cycle will have a high level planning and analysis to outline the scope of the project, and then undergo a series of iterations (Waters, 2011). Each iteration is covered the phase of analysis, design, develop and test.

At the analysis phase, any related information or new ideas gathered will have a deep study and analyzed. What kinds of tools are needed in development, what communication protocol is needed for the application, and how to implement the application in opportunistic network, will all decide in this phase.

The next phase is design and development, which come out the desire application's work flow and user interface. Once the design is done, development of the application will then proceed. In this project, Eclipse will be the main development tool as it is an opened source, can download and install the android-based libraries or packages on it.

At testing phase, developer will do debug and run the application by using Eclipse's emulator. The developed application will be tested with different scenarios in order to ensure the application can work well, and will be carried out by a group of users. Any errors found or feedback given will be recorded for enhancement.

Agile-approached is having iterations on these phases. As there is some new useful information or ideas, a new iteration will be occurs. Each of these iterations can make the project nearer to the goal.

## **1.5 Scope**

In this project, a simple android-based text messaging application is intended to be developed and investigate whether it can work in opportunistic network, where it only work in wireless network. This text messaging application is only developed for send and receives text messages, with one-to-one device communication.

## **1.6 Significant of Project**

The developed application can be used to communication (send/receive text message) between at least two devices. The application can work in infrastructure-less environment and without the need of connected to Internet.

## **1.7 Project Outline**

This project is organized into few chapters:

### **Chapter 1: Introduction**

This chapter is about the overview of the whole project. Through this chapter, problem statements, the objectives, scopes and limitations, and significant of the project will be clear.

### **Chapter 2: Literatures Review**

The backgrounds of the project, related works are described in this chapter. Comparison of the similar applications will be done as well.

### **Chapter 3: Methodology / Requirement Analysis and Design**

Analysis based on the project requirement will be carried out, suitable development method and tools will be chosen. Then, come out the application design.

### **Chapter 4: Implementation and Testing**

The implementation and test cases of the application that based on the design will be discussed.

### **Chapter 5: Conclusion and Future Works**

Finalize the project's finding and discuss the limitation and some needed future enhancements.

## **Chapter 2: Literature Reviews**

### **2.1 Introduction**

In this chapter, literature reviews of existing project field are presented. The purpose of this chapter gives fundamental knowledge regarding P2P android application. The review is about the android operating system, text messaging, opportunistic network and some overview of the existing text messengers. From all these reviews, a proper solution may be generated for the proposed project.

### **2.2 Android**

Android mobile operating system is founded by Google and released under the open source Apache License. Android's market share is now over 80% as in the third quarter of year 2013, which dominating the smartphone sales (Rosoff, 2013). Hashimi, Komatineni and MacLean (2010) states that Android is actually "features Linux-based operating system stack for managing devices, memory, and process". Programmers who familiar with Java programming language are able to adapt Android's development in a short period of time, as it supports most of the Java Platform, Standard Edition except the Abstract Window Toolkit (AWT) and Swing.

The latest platform version of Android is 4.4, version code named KitKat, with Application Programming Interface (API) of level 19 (Google Official Blog, 2013). Kashyap (2010) defines that API is "the interface implemented by an application which allows other application to communicate with it". Developers are usually looking for the API libraries and implement latest features for their application development. Android platform is a high-end

development framework, provided openness and affordability for user to create its own applications. Developers can create and publish their Android-based applications to Google-operated online Android Market; while users can browse for the applications and download to their smartphone. Android has less restriction as compared with other mobile operating system like iOS. Android can be customized on phone and able to install Android apps from other sites than official Google Play Store.

### **2.3 Text Messaging**

A simple way to define instant messaging (IM) is sending a text message to a recipient and the recipient receives the message instantly. Edney and Maximo (2007) define that IM is usually a text-based message communication between two or more people in real time by using a device, which always referred as *chat*. This type of communication existed since long time ago and yet, it's still popular, as it is faster and easier to use than email, or even SMS. To use IM, a user has to sign in the IM service, start a conversation by choosing one of the contact list, then wait for the recipient reply. IM is not only available on personal computer (PC), but web-based and even mobile devices. Today's IM is much more advanced than before, most of the IM able to send emoticons, voice message, file sharing, and even video call. However, most of the IM are need to connect to internet in order to operate.

Jennings et al. (2006) states that most of the IM are using client-server architecture, where clients (or users) have to be connected through servers in order to be able to communicate to others. There are two types of system architectures as for scaling the expansion number of users, which are symmetric and asymmetric. Every server in symmetric architecture will perform the same operations as a client started to establish a connection and

communicate. But the servers in asymmetric architecture will have its own specific role, such as user logging in, discovering other users, maintaining a chat room, or exchanging messages. On the other hand, some of the IM also provided some administrative and management functions. There are IM that have the features that managing user's contact list where user can groups the contacts into "favorite" or "block" list. Favorite list consist of the frequent contacted friends while Block list can prevent some friends from bothering but without the friends' knowing. Server will keep and maintain these lists as persistent state, so that clients will able to synchronize it when log in. Another common feature can be found in IM is the message's status. Users are able to know whether the message is delivered successfully or been read. But, there are users who don't like this feature as user may find out the receiver purposely not reply message even the message is delivered and been read.

## **2.4 Opportunistic Network**

Huang, Lan and Tsai (2008) defined that opportunistic network is "where a network contacts are intermittent or where link performance is highly variable or extreme". In such network, there is no guarantee an end-to-end path between the source and destination exist persistently. In order to make a communication in this network, the intermediate nodes may need to store and carry the data with the advantage of node mobility, until an opportunistic contact is met, then forward the data. This store-carry-forward fashion enables the data to be delivered from a region to another different region of a large network. Hence, each node in opportunistic network may play several roles: host, router or gateway. With the role host, the node must have enough storage to store and support transmission. The node will act as a

router, which store, carry and forward the data among the nodes in the same region. If the data need to send to different region, the node will then act as a gateway.

Opportunistic network is different with the traditional mobile ad-hoc network where the nodes are not necessary to aware about the network topology. As described above, the route taken by data to its destination is unpredictable, where any possible node can opportunistically be used as the next hop, and perhaps able to bring the data closer to its destination. There are some drawbacks by using this network, the data sent may experience additional delay of the delivery. So, the applications may need to tolerate with this implementation. Delay-tolerant network (DTN) and opportunistic network may cause some ambiguities as there are several concepts of opportunistic network are actually come from the studies of DTN (Pelusi, Passarella & Conti, 2006). DTN are assumed to know the Internet-like topologies, in which some links between gateways could be available just at certain times; while opportunistic network does not. The DTN architecture consists of a network of independent internets that located apart from each other form DTN regions, and interconnection among them is provided by the system of DTN gateways. This is why in DTN, points of possible disconnections are known and isolated at gateways.

Helgason, Yavuz, Kouyoumdjieva, Pajevic and Karlsson (2010) introduce a middleware architecture for a mobile peer-to-peer (P2P) content distribution system that allowing content dissemination between mobile nodes. In this architecture, the contents are able to transmit among nodes when within the coverage, opportunistically. This architecture is divided into three main components in this architecture: content structure, solicitation protocol and API. The contents are grouped into logical topics and allows for efficient matching of content lookups and downloadable via opportunistic contact of nodes. By having a protocol, the nodes are able to explore and discover the contents of the nodes, then

download. The API is to let the system services be accessed by applications via publish/subscribe interface.

## **2.5 Existing Communication Applications**

### **2.5.1 WhatsApp**

WhatsApp messenger is one of the top IM using in mobile devices all around the world. It is founded in year 2009, by two former employees of Yahoo! (WhatsApp Inc, 2013a). WhatsApp messenger is supported on iPhone, Android, Windows Phone, Sony and BlackBerry, even though they are using different kind of OS (WhatsApp Inc, 2013b). WhatsApp can send and receive text messages, images, video and audio messages when connected to Internet. Other than that, it can have a group chat and the ability of let users to know whether the message is been sent successfully or been read as well. There is no hidden cost of using WhatsApp, but it will be charged for \$0.99 annually after the first year of free service.

WhatsApp's simplicity is the attractive point for some users and text messaging is only their main concern. WhatsApp doesn't has provide any socialization or sharing features like LINE and WeChat. Hence, there will be no any events or advertisements approach to users, which may get rid of annoyances from users. Besides that, WhatsApp able to show is the user currently online or when was his/her last time accessed WhatsApp. However, WhatsApps is only able to interact with the person who are using WhatsApp and whose contact numbers are added in the device, users are not able to add new contact via WhatsApp. WhatsApp is only provides emoji in messaging, and doesn't has sticker like other text