

SMS Based Emerging Techniques for Monitoring and Controlling Android Mobiles

Deepak Kumar and Mohammed Abdul Qadeer

Abstract—This paper describes a software application for android mobile platform that discovers various excited applications of SMSs over its traditional text messaging application. It shows how various features in android mobiles can be automated by SMS. By this application user can perform various operations in its mobile even if mobile is very far from him, like by sending a single text message we can fetch and store our contact numbers, fetch our device's location, auto respond to the incoming messages, send SMS from our remote mobile, fetching SIM and mobile details used for GSM network. Convenience to the user, security and efficiency are main issues that are considered. This application makes the use of services like telephony, location based services (LBS) and native android applications.

Index Terms—Android, LBS, new SMS applications, telephony.

I. INTRODUCTION

SMS is very common and widely used way of communication. Now-a-days we do most of the communication with SMSs. Today in the era of technology we want most of the things to be automated. Imagine that it would be great if we could perform various functions in our mobile phones even if it is far from us or it could respond automatically like an intelligent device. So now this can be achieved by our SMS software application which is developed for android mobile platform. By using this application we can operate many functions via sending a SMS to the mobile phone which is far from us without interception of operator and in this way our android mobile phone will act as intelligent device. This application establish client-server relationship between mobile phones in which the mobile requesting operations by sending SMS will act as client and mobile serving those operations will act as server. Various operations that can be performed by this application are listed as storing and fetching contact numbers, fetching the device's location, sending SMS to other mobile phones through our remote mobile, auto responding to the incoming messages, finding and fetch the details about SIM and mobile like and we can also switch off our mobile via sending a SMS to it. This application makes the use of both traditional and advance technology like telephony and location based services (LBS). These services are also used in various applications but we are presenting them in very different way from there conventional use. There are some issues that have

gotten more attention like convenience to the user, security which is very important and necessary aspect of this application and efficiency. To use this software application we have some fixed formats of SMSs that are needed to be sending on the android mobile on which application is installed. These formats are discussed later sections in this paper.

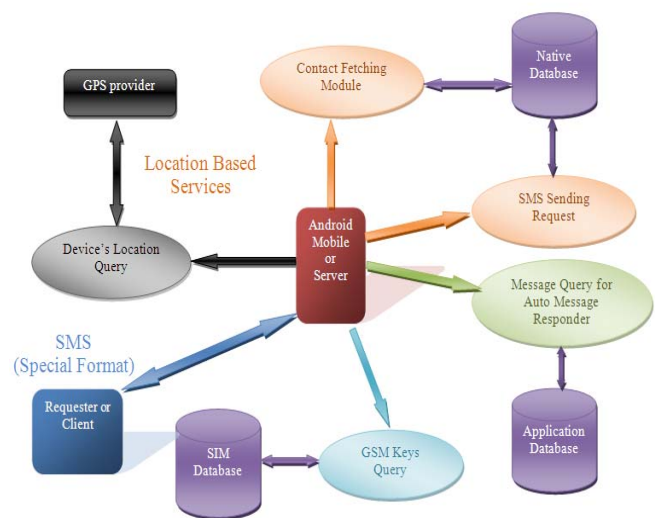


Fig. 1. Block diagram representing system

II. BACKGROUND

A. Brief Introduction to Android

Android is an open source software stack that includes the operating system, middleware, and key applications along with a set of API libraries for writing mobile applications that can shape the look, feel, and function of mobile handsets [1]. Mobile phones were largely closed environments built on proprietary operating systems that required proprietary development tools. The phones themselves often prioritized native applications over those written by third parties. This has introduced an artificial barrier for developers hoping to build on increasingly powerful mobile hardware. In Android, native and third-party applications are written using the same APIs and executed on the same run time. We are also making the use of native application database without any barrier. Android is backed by more than 40 OHA (Open Handset Alliances) members [2] and is surrounded by significant industry buzz. In market terms, the growth in portable devices is a worldwide phenomenon, with mobile-phone ownership outstripping computer ownership in many countries.

Manuscript received June 20, 2012; revised September 10, 2012.

The authors are with the Department of Computer Engineering, Zakir Husain College of Engineering and Technology, Aligarh Muslim University, Aligarh 202002, India (e-mail: deepakkumar@zhcet.ac.in, maqadeer@gmail.com).



Fig. 2. Android mobile phone

B. Telephony

Telephony is a general term that refers to the details surrounding electronic voice communications over telephone networks. Most fundamental components of the mobile phone are dialing out, receiving calls, sending and receiving text messages, and other related telephony services. The added bonus with Android is that all of these items are accessible to developers through simple to use APIs, and built-in applications that make use of intents and services. We can use the telephony support Android provides quite easily, and we can also combine it and embed it in our own application. So by embedding these APIs we have created these excited features.

C. Location Based Services

Given the recent advancement of mobile telecommunications technology and rapid diffusion of mobile devices, the importance of wired and wireless Internet services utilizing the past and present location information of users carrying mobile terminals with location tracking function is growing. LBS refer to value-added services that detect the location of the users using location detection technology and related applications. LBS is expected to play an essential role in creating value-added that utilizes wired and wireless Internet applications and location information, since these are very useful in various fields [3,4,5].

III. IMPLEMENTATION

This SMS software application is implemented on android platform using Android SDK 1.6. Since there are various functional module implemented here so we are discussing it one by one.

A. Detecting and Scanning Incoming SMS

For detecting the reception of SMS we have to register a broadcast receiver in our application manifest file statically. Broadcast receiver detects the broadcasted intents. If the intent's action received by the broadcast receiver is SMS_RECEIVED, it's mean that a SMS has been received. Now Broadcast receiver will invoke the background service class of our application even if our application is not in running state. The specialty of service class is that it runs in background without interrupting other applications. It does not have any user interface. Now entire received text message

will be passed to the service class where this SMS will be scanned one by one character and if any format defined by our application is matched with SMS than appropriate action will be performed.

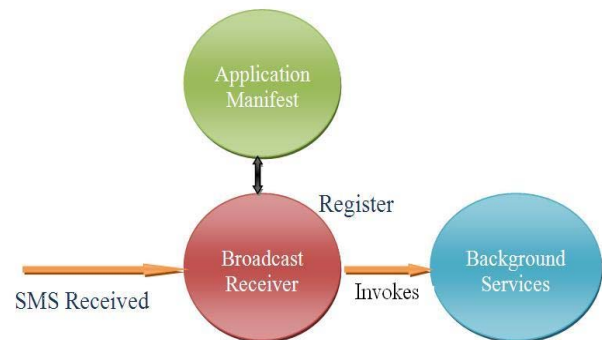


Fig. 3. Actions taking by a broadcast receiver.

B. Authentication Module

Since in this application we are accessing our personal details so security is very essential aspect in this software. That means only authenticated person can use the features offered by this software. Authentication is provided by a password. We can enter a new password in this application which can only be changed by us. We provided no way by which we can see that password. For fetching any information with the help of SMS we first need to know password. Password may be of any number of characters but personally suggest taking it more than 6 characters. All the SMS format that we have provided here stats with password except auto message responder. If service class finds the first keyword in incoming message as password then it further searches for other keywords for specific operation. For security purposes we suggest here to change password frequent this will we more secure and no one can track it.

C. Storing and Fetching the Contact Number

Suppose the case when we forget our mobile phone at home or in office and we want some important contact number immediately which is stored in our mobile phone. Then how we can get those contact number. This can be achieved by this application which must be installed on that phone. To fetch or store the contact number we must send a SMS in a fixed format on our mobile. To fetch a contact, message must be in format like

1) *Password fetch contact_name*
and to store a contact number , message must be in format like

2) *Password stored contact_name contact_number*

All these keywords must be separated by a single space. So when a text message is received and transferred to the service class. Service class than scan the message and separates out all the keywords if password matches to the password stored in the application database then next our application will make use of the native contact application database. This can be achieved by URI and content provider. A URI is a generic Uniform Resource Identifier, as defined by RFC 2396 [6], which include a scheme, an authority, and optionally a path. A Content provider is used in Android to share data between different applications. A code segment showing the fetching of contact number using URI and content provider is given

below

```
public String dataPath="content://contact"
final Uri data=Uri.parse(dataPath + "people/");
ContentResolver resolver= getContentResolver();
Cursor c=resolver.query(data, projection,People.NAME +
"=" + "" + contact_name.trim() + "", null, null);
```

After contact fetching application will automatically send back this contact number to the requester through a SMS. In the same contact can also be stored using URI and content provider.

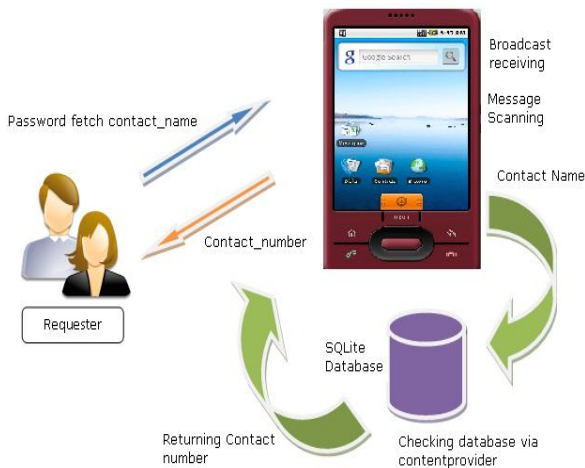


Fig. 4. Representation of fetching of a contact.

D. Fetching the Location of Your Mobile

We can also track the location of our mobile with this software. When we lost our mobile or stolen by someone, then in that case our SMS application software will be very useful because only by sending one SMS to our mobile it will automatically send its location back to us. To find the location of the mobile android uses location based services. The text message format for finding the location is

1) Password map location

When this format is detected by the service class the software understands that requester wants to find the location of this device. Location-based services are dependant on device hardware for finding the current location. Depending on the device, there may be several technologies that Android can use to determine the current location. Each technology, or GPS Location Provider, will offer different capabilities including power consumption, monetary cost, accuracy, and the ability to determine altitude, speed, or heading information. Access to the location-based services is handled using the Location manager system Service. To access the Location Manager [7], request an instance of the LOCATION_SERVICE using the getSystemService method. We can find the last location fix determined by a particular GPS Location Provider using the getLastKnownLocation method, passing in the name of the Location Provider. The following code segment shows how we are using LBS.

```
String ls=Context.LOCATION_SERVICE;
LocationManager
lm=(LocationManager)getSystemService(ls);
```

```
Location l=lm.getLastKnownLocation(LocationManager.
GPS_PROVIDER);
Geocoder g= new Geocoder(getApplicationContext(),
addresses=g.getFromLocation(l.getLatitude(),
l.getLongitude(),1);
if(addresses!=null)
{Address curAdd=addresses.get(0);
if(curAdd.getMaxAddressLineIndex()>0)
for(int i=0;i<curAdd.getMaxAddressLineIndex();i++)
sb.append(curAdd.getAddressLine(i));}
```

Here l. get Latitude and l. get Longitude will give the longitude and latitude of location of device. Geocoding lets us to translate between street address and latitude/longitude coordinates with the help of network provider lookup database. So we have created a geocoder to convert the latitude/longitude into the street address. It is called the reverse geocoding [4,5]. Then this complete street address is send back to the requester automatically by mobile.

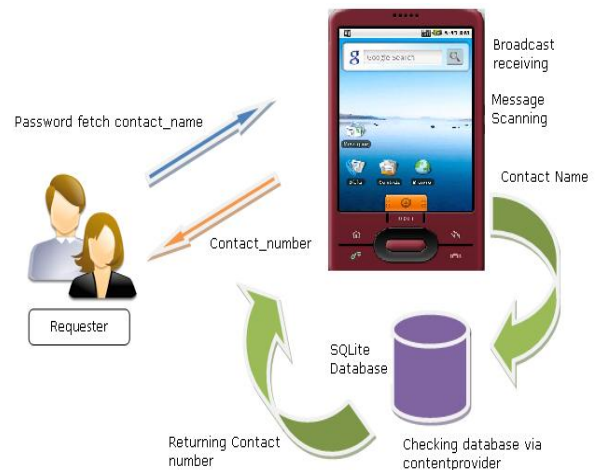


Fig. 5. Representation of fetching of location of mobile

E. Auto Responding to Incoming Text Messages

When we are busy or not in the condition of answering the incoming messages then our SMS software may be very useful. We have provided here the auto message responding facility in which our mobile automatically answers for message queries that are previously stored in it. We have provided here the query customization that means user can itself add the message query to which he wants to auto respond and their corresponding responses. He can simultaneously add more than one response for a single query. User can later set or change the state of response string to which he wants to send. Suppose user have added a query "where are you" then user can add "home" and "office" as his response and when he is at home he can set response as "home" or otherwise as "office" We have here also provided the facility of sending location with response string. All these message queries are store in SQLite database provided by android. Whenever a text message is received software looks up table of message queries in database to check whether it any query matches to the incoming message. If a match has occurred then it finds the corresponding response status and sends it back to the requester.



Fig. 6. Message query customization



Fig. 7. Response status setting

F. Sending SMS to Other Mobile Phones through Our Remote Mobile

Suppose when we have forgotten our mobile at home or office and now we want to send a very important message to our friend but his number is not available. Then with this software this can be achieved that is we can send a text message to our mobile along with contact number with a special format.

1) Password sendSMS contact_name text_message

All these keywords should be separated with the single space. Service class separates out all these keywords. Password is used for the user authentication. sendSMS keyword is used to identify that software has to find the text message from this SMS. Text message is the subset of the that incoming message of android mobile. Contact_name keyword is the name of the person to which message has to send. Now our software finds the contact number corresponding to that name using native application database. Now this text message will be sent to that person specified in message.

G. Determining GSM Details

GSM is a cellular telephone network. Devices communicate over radio waves and specified frequencies using the cell towers that are common across the landscape. This means the GSM standard has to define a few important things, like identities for devices and “cells,” along with all of the rules for making communications possible. The Android stack currently uses to support voice calls – and it's the most

widely used standard in the world across carriers and devices Android or otherwise. All GSM devices use a “SIM” card, or Subscriber Identity Module, to store all the important network and user settings like

1) Integrated circuit card ID (ICCID)

Unique number that identifies a SIM card (also known as a SIM Serial Number or SSN)

2) International mobile equipment identity (IMEI)

Unique number to identify a physical device (usually printed underneath the battery)

3) International mobile subscriber identity (IMSI)

Unique number to identify a subscriber (and the network that subscriber is on)

4) Location area Identity (LAI)

Unique number that identifies the region the device is in within a provider network.

5) Authentication key (Ki)

A 128 bit key used to authenticate a SIM card on this provider network.

We need to know that we are working with a GSM network, and then we need to be aware that we may come across terms like IMSI and IMEI which are stored on the SIM. Sometimes we need to know about any of these ID for security or other purposes. Our software can also determine these IDs with a SMS. Getting at this information is done with the TelephonyManager class [6], [7].

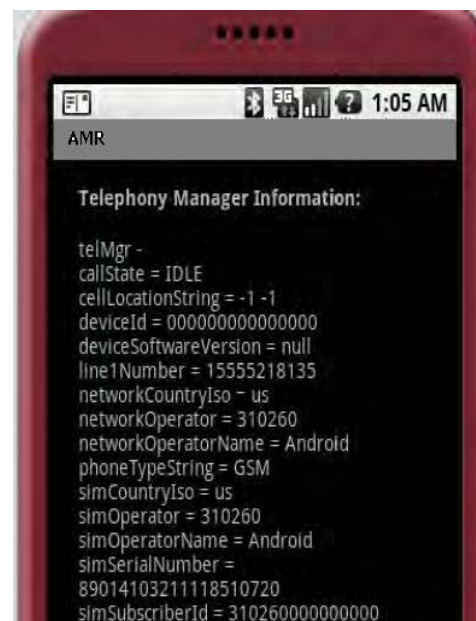


Fig. 8. Fetching GSM and telephony details

We also have format for fetching these details

Password IDs

like

Password IMEI LAI Ki

By the above format we can determine IMEI, LAI and Ki details with a single SMS.

IV. CONVENIENCE TO THE USER AND EFFICIENCY

We have obviously created this software to provide convenience to the user and increasing the use of SMSs. To use this software user does not require any special training.

Only he need to know the keywords used in this software and their special software. This is completely practical basis software and may provide a new direction to the use of SMS technology. It is not 100% efficient but we are trying to make it as efficient as possible. Its efficiency goes down like in case of auto message responding it matches query character by character. So mistake of a single character may fail it. But for the solution of such types of problem now we are also implementing data mining in this application so that it can understand the incoming message queries more efficiently.

V. CONCLUSION

We have implemented this SMS application with a thinking that new things are always appreciated. As per the articles referenced here we do not think that such types of new features have been yet implemented in any other SMS software application [8,9]. This software also provides an exited ways of using your mobile phones via SMS. In this application android phone acts as an intelligent device which performs many functions only by checking incoming text message format. It saves time of writing SMS when user is busy and it also helps in many other ways.

REFERENCES

- [1] R. Meier and Wrox, *Professional Android Application Development*, First Editon
- [2] Android. [Online]. Available: <http://www.android.com>
- [3] N. Park, *et al.*, "The security consideration and guideline for open LBS using XML security mechanism," ASTAP 04/FR08/EG.IS/06, 2004.
- [4] M. Tuisku, "Wireless java-enabled MIDP Devices as peers in grid infrastructure," *Helsinki Institute of Physics, ERN*.
- [5] H. Srivatsa, "Location-based services," *IBM Paper*, November 2002.
- [6] J. F. Jerome, *Android a programmer's guide*, MGH Publisher, Second Editon
- [7] Code. [Online]. Available: <http://google.com/android>
- [8] J. R. Cuellar, *Location Information Privacy. B.* 2002.
- [9] Srikaya (Ed.), *Geographic location in the internet*, Kluwer Academic Publishers, pp. 179–208.
- [10] S. Ravi, M. S. Chathish, and H. Prasanna, "WAP and SMS based emerging techniques for remote monitoring and control of a process plant," in *Proc.2004 7th International Conference on Signal Processing*.



Deepak Kumar was born in India in June 1988. He has completed polytechnic in computer engineering from University Polytechnic, Aligarh Muslim University, Aligarh India in 2007 and a graduate degree in Computer Engineering (class of 2011). His area of interest includes technological growth for humanity, heterogeneous computing and WMA. Mr. Kumar has two ACM, one IEEE and one international Publication.



Mohammed A. Qadeer (MIEEE, LMIETE, LMCSI) is an Asst. Professor with the Department of Computer Engineering, Aligarh Muslim University, India. Earlier, he was working with Cisco Systems Inc. as a Network Consulting Engineer with the Advanced Services division in the APAC region. He received his B.Sc. Engineering (Computer Engineering) from Aligarh Muslim University in 1996. He has an experience of 15~ years in the area of computer networks and systems. He served as a Technical Co-Chair for IEEE WOCN 2012, 2011, 2010, Technical Co-Chair IEEE AH-ICI 2012, 2011, International Steering Committee for ICACT 2012, 2011, 2010 and as TPC member for CCNC 2011, 2010, INMIC 2009, AH-ICI 2009, WIA 2009 and MMA 2009. He is on the editorial board of International Journal of Digital Multimedia Broadcasting. Established global and nationwide setups of Internet Service Providers (ISP), Internet Exchange Points (IXP), Internet Data Centre (IDC) and Content Delivery Networks (CDN) both from a Networks and Systems perspective. His areas of research are computer networks, wireless networks, mobile computing, next generation networks, IMS, LTE, WiMAX, 4G, WiBro etc.