AI College Phonebook and Automatic Notification

Project report submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology

in

Information Technology

By

Prashant Gill(131402)

Under the supervision of

Dr. Suman Saha

to

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology Waknaghat,
Solan-173234, Himachal Pradesh

CANDIDATE'S DECLARATION

I hereby declare that the work presented in this report entitled **"AI College Phonebook and Automatic Notification "** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Computer Science and Technology** in **Information Technology** submitted in the department of Information Technology**,** Jaypee University of Information Technology, Waknaghat is an authentic record of our own work carried out over a period from August 2016 to April 2017 under the supervision of **Dr. Suman Saha** Assistant Professor, Department of Computer Science And Engineering. The matter given in the report has not been submitted for the award of any other degree or diploma.

Prashant Gill (131402)

This is to certify that the above statement made by the candidate is true to the best of my knowledge.

(Supervisor Signature)
Dr. Suman Saha
Assistant Professor
Department of Computer Science And Engineering
Dated:

ACKNOWLEGEMENT

I am grateful and indebted to **Dr. Suman Saha**, Assistant professor, **Department of Computer Science and Engineering** for his help and advice in completion of this project report. I also express my deep sense of gratitude and appreciation to my guide for his constant supervision, inspiration and encouragement right from the beginning of this Project report. I also want to thank my parents and friends for their immense support and confidence upon me. I deem it a pleasant duty to place on record my sincere and heartfelt gratitude to my project guide for his long sightedness, wisdom and co-operation which helped me in tackling crucial aspects of the project in a very logical and practical way.

Prashant Gill
(131402)
Information Technology

TABLE OF CONTENTS

LIST OF ABBREVATIONS

SDK - Software Development Kit

APIs - Application Program Interfaces

GUI - Graphical User Interface

OHA - Open Handset Alliance

GPS – Global Positioning system

LRU – Last Recently Used

MHTML – Mobile HTML

QoS – Quality of Service

WAP – Web Application Protocol

CSD – Circuit Switched Data

OTA – Over-the-Air

LIST OF FIGURES

LIST OF TABLES

ABSTRACT

The title of this project is "AI College Phonebook and Automatic Notification". The principle reason for phonebook is to oversee telephone contact day by day operation effectively. This framework essentially index has been every now and again being used in our day by day life.

We usually observe index introduced in, cell phone and so forth. Phonebook Directory application gives the capacity to pursuit, see, and oversee sections in an index. Versatile Directory application ought to permit any supporter with a cell phone.

The query output information stacks specifically to the portable screen and gives the client alternative to CALL, SAVE,VIEW, END TO FRIEND or DISCARD. Portable Directory takes out the need to call whatever other call focus.

CHAPTER 1

INTRODUCTION

Main purpose of my project is to manage college students contact operations efficiently. College phonebook is used to provide search, view and manage entries in directory. According to my project the admin has right to CALL, ADD CONTACT, SAVE, DELETE, EDIT or send a group message. With the use of automation feature, the message for particular task send automatically to the student.

1.1 What is Android

Android is the most popular mobile platform of world. Android is an open source and Linux based working framework for cell phones, for example, cell phones and tablet PCs. In the main android was produced by the Open Handset Alliance driven by Google and different organizations. Android SDK provides necessary tools and API's for developing android applications using java language.

1.1.1 History of android

Android was created by the Open Handset Alliance driven by Google and different organizations. In 2007 Google discharged first beta variant of android SDK yet the main business rendition, android 1.0, was discharged in September 2008. Google declared the following android form on July 27, 2012.

1.1.2 Growth of Android

In early times of android, the growth rate is slow but after 2009 growth rate goes high as android was able to add new customers quickly, and this is shown in the below figure.

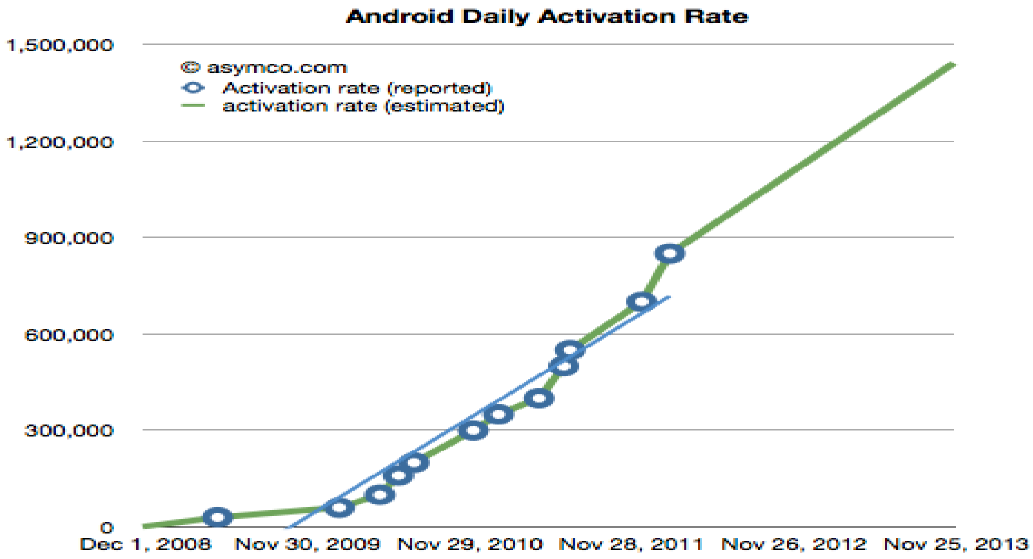


Fig. 1 – Growth of Android Users

1.1.3 Android Features

Android have great features, few of them are listed below.

- **Application framework –** enables reuse and replacement of components.
- **Beautiful UI** – android OS fundamental screen gives an excellent and associating UI.
- **Storage –** database, is used for the purpose of data storage.
- **Media support –** for audio, video, and image formats (JPEG, GIF, PNG).
- **Messaging –** used for SMS and MMS.
- **Multi-tasking -** utilized for bouncing starting with one undertaking then onto the next errand and in the meantime different applications can run all the while.
- **Web browser –** based on open source WebKit engine.

1.1.4 Android Architecture

Android architecture is divided into five parts:

1. Linux Kernel
2. Native Libraries
3. Android Runtime
4. Android Framework
5. Applications

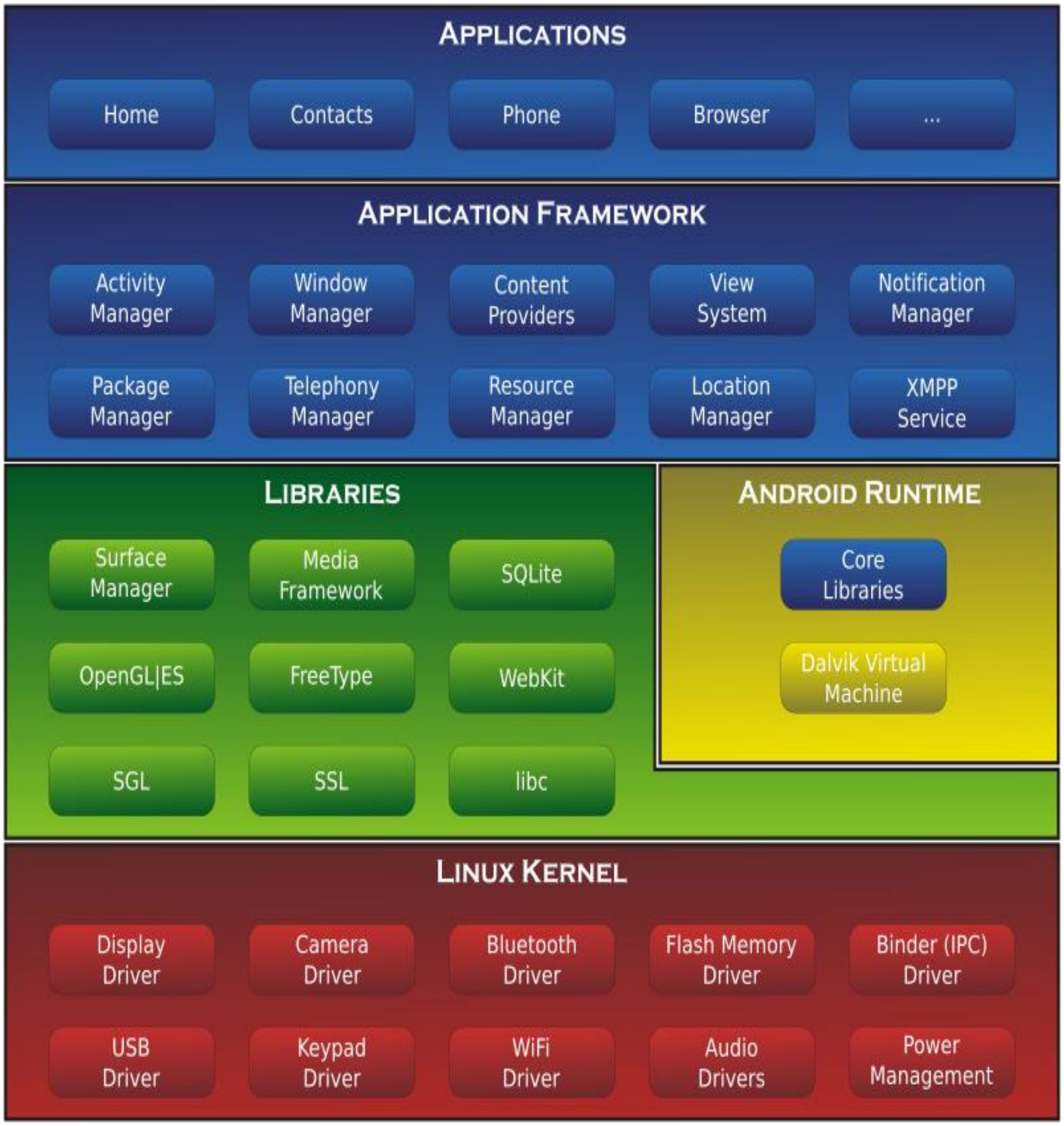


Fig. 2 Android Architecture Diagram

Linux Kernel

It is just like heart for man that mean, Linux Kernel is heart of android architecture. Linux Kernel is used for device drivers, power management and resource access.

Native Libraries

There are native libraries such as WebKit, SQLite, Media etc.

Android Runtime

There are core libraries and DVM in android runtime which is responsible for android application run.

Android Framework

Android framework includes android API’s.

Applications

Android applications find at the top layer. Examples are Contact Books, Settings etc.

1.2 Problem statement

Implementation of AI College Phonebook and Automatic Notification, in this application students will get information which is must for them.

1.3 Aim and objectives

The aim is to connect more students with the university, departments by sending the necessary message which is important for the students.

1.4 Methodology

In this project I simulate college phonebook using automation for sending message to students. Prime methodology for this project is automation of notification.

1.5 Organization of project report

In Chapter 1 I have examined about android essentials, the present development in this field, the basic difficulties being confronted by people in actualizing the android applications

.

In Chapter 2 I will give the essential wording about the distinctive research paper which are perused by me. I will give the statistical data points about various ideas I have examined in those exploration papers.

In Chapter 3 I will give a model of how the venture is done on the premise of improvements :-

- Analytical
- Experimental
- Statistical

In Chapter 4 I have given a legitimate examination of android utilization of college phonebook on the premise of programmed notice.

In Chapter 5 I have given the snapshots that I get from my project.

CHAPTER 2

LITERATURE SURVEY

Writing literature is intends to assess and translate the accessible research that is applicable to a specific research zone. The primary point of writing literature is to exhibit a reasonable assessment of the intrigued look into range by leading an appropriate strategy. The primary reason for my writing overview is to locate the significant writing about Android phonebook application and notification of use for the foundation consider.

The objective behind making this application was to connect more number of students with department and teachers through a mobile device. Now the problem is that on which platform I should implement this application, so I have choosed android platform for some reasons:

- Android is an open source platform
- It supports multifunction
- Provide good quality tools to make interactive application

According to the android website the android platform is divided into four core features as given in below figure

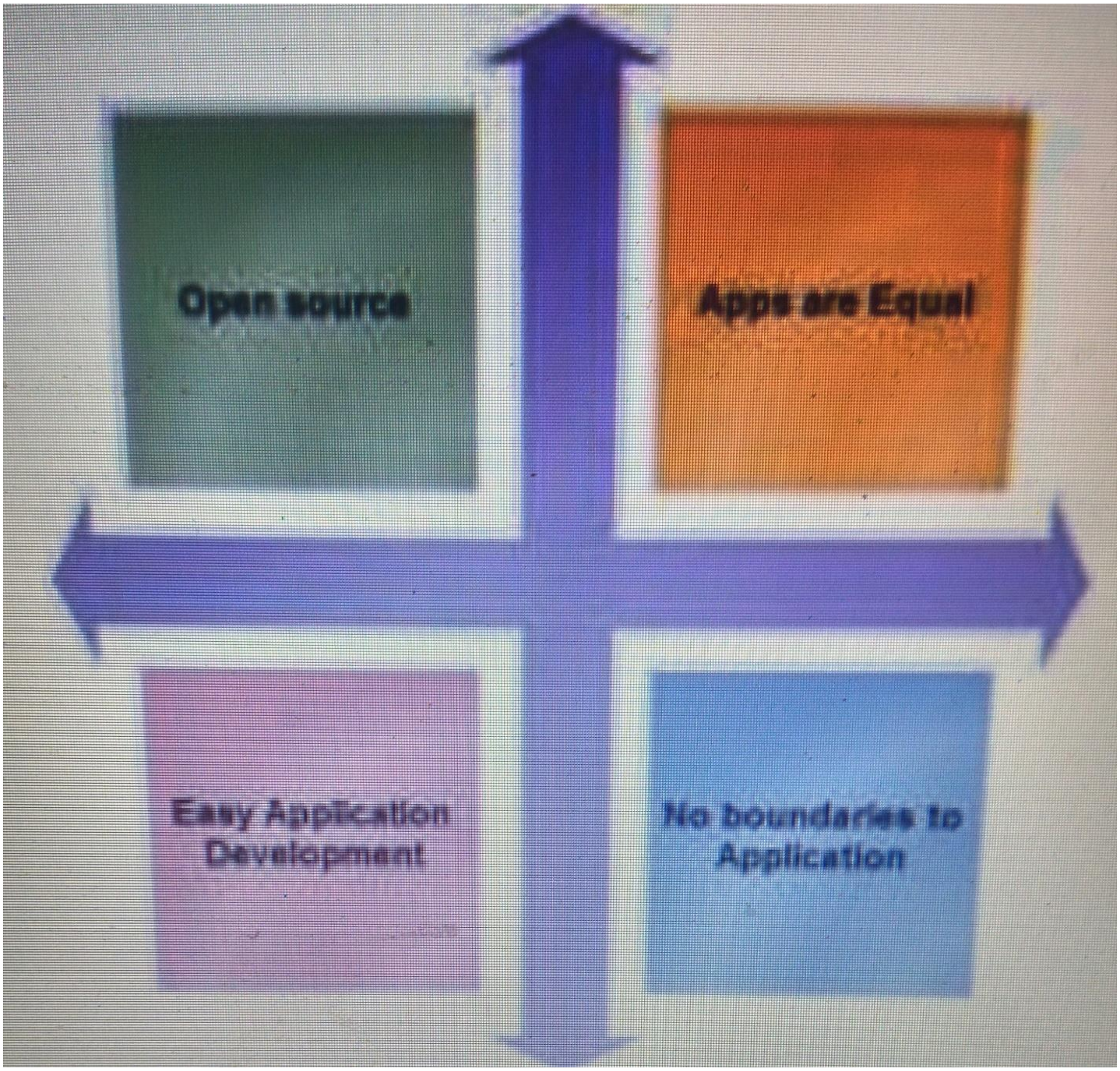


Fig. 3: Four core features of the android platform

2.1 Platform Preview

Android is the most popular mobile platform of world. Android is an open source and Linux based operating system for mobile devices such as smartphones and tablet computers. In the first android was developed by the Open Handset Alliance led by Google and other companies. Android SDK provides necessary tools and API’s for developing android applications using java language.

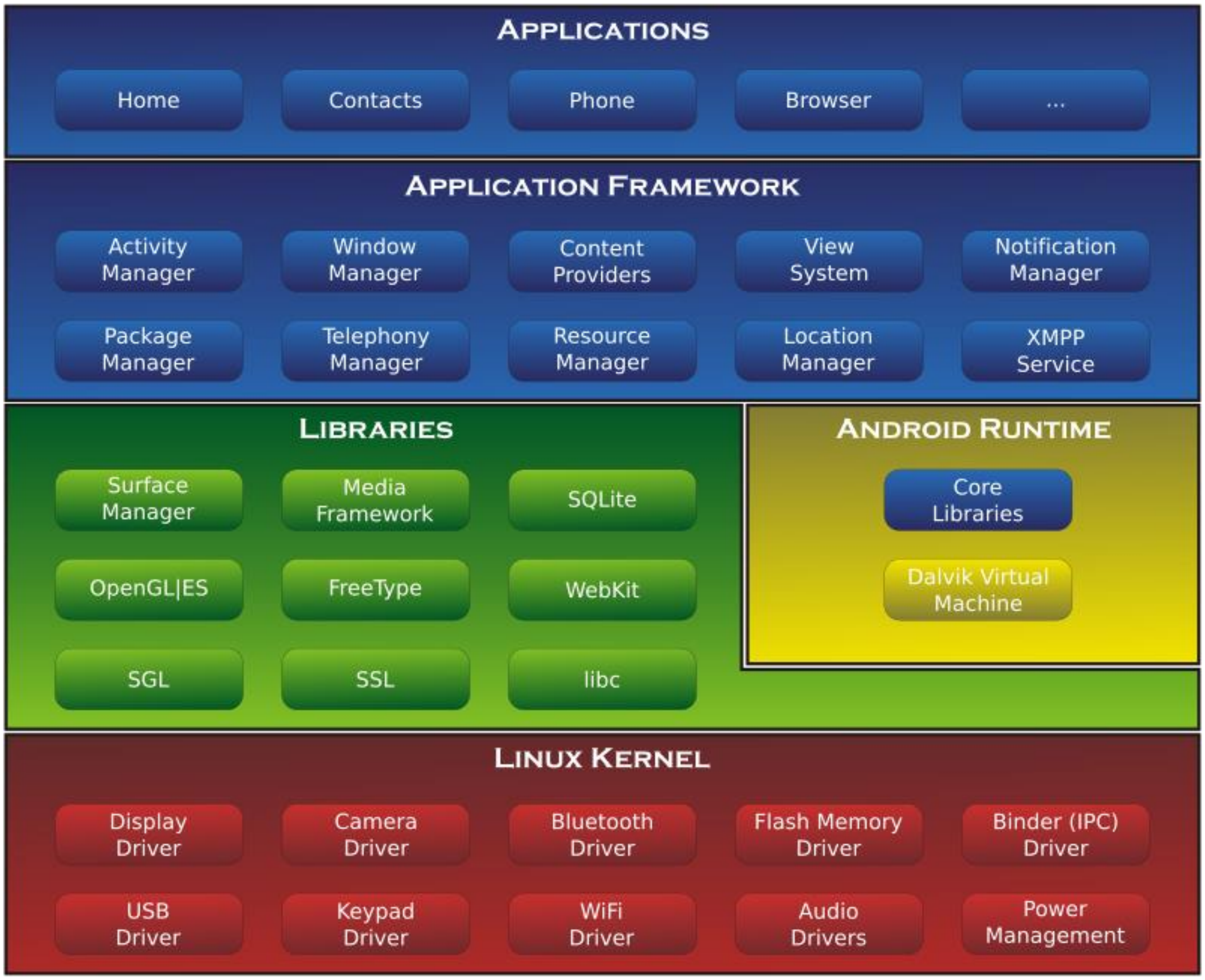


Fig.4 – Android application architecture

2.2 Application Framework

Developers have full access to the same framework APIs used by the core applications. The application architecture is designed to simplify the reuse of components; any application can publish its capabilities and any other application may then make use of those capabilities (subject to security constraints enforced by the framework). This same mechanism allows components to be replaced by the user.

Basic all applications is an arrangement of administrations and frameworks, including:

- A rich and extensible arrangement of Views that can be utilized to fabricate an application, including records, networks, content boxes, catches, and even an embeddable web program.

- Content Providers that enable applications to access data from other applications (such as Contacts), or to share their own data.

- A Resource Manager, giving access to non-code assets as restricted strings, illustrations records.

- A Notification Manager that empowers all applications to show custom cautions in the status bar

- An Activity Manager that manages the life cycle of applications and provides a common navigation backstack

2.3 Libraries

Android includes a set of C/C++ libraries used by various components of the Android system. These capabilities are exposed to developers through the Android application framework. Some of the core libraries are listed below:

- **System C library** - a BSD-inferred execution of the standard C framework library (libc), tuned for installed Linux-based gadgets.

- **Media Libraries** - in light of PacketVideo's Open CORE; "the libraries bolster playback and recording of numerous mainstream sound and video" groups, and in addition static picture documents, including MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG

- **Surface Manager** – oversees access to the show subsystem and consistently composites 2D and 3D realistic layers from different applications

- **LibWebCore** – a present day web program motor which powers both the Android program and an embeddable web see

- **SGL** - the underlying 2D graphics engine

- **3D libraries** – an execution in view of OpenGL ES 1.0 APIs; the libraries utilize either equipment 3D speeding up (where accessible) or the included, profoundly advanced 3D software rasterizer

- **Free Type -** bitmap and vector textual style rendering 3.3 Android Runtime

Android fuses a game plan of focus libraries that gives most by far of the value open in the inside libraries of the Java programming lingo. Every Android application continues running in its own technique, with its own specific event of the Dalvik virtual machine. Dalvik has been created so that a contraption can run different VMs capably. The Dalvik VM executes records in the Dalvik Executable (.dex) orchestrate which is enhanced for immaterial memory impression. The VM is enroll based, and runs classes accumulated by a Java tongue compiler that have been changed into the .dex mastermind by the included "dx" instrument. The Dalvik VM relies on upon the Linux piece for essential helpfulness, for instance, threading and low-level memory organization.

At a comparable level there is Android Runtime, where the crucial portion Dalvik Virtual Machine is found. It was formed especially for Android running in obliged condition, where the confined battery, CPU, memory and data stockpiling are the guideline issues. Android gives a consolidated instrument "dx", which changes over created byte code from .holder to .dex record, after this byte code ends up being altogether more capable to continue running on the little processors.

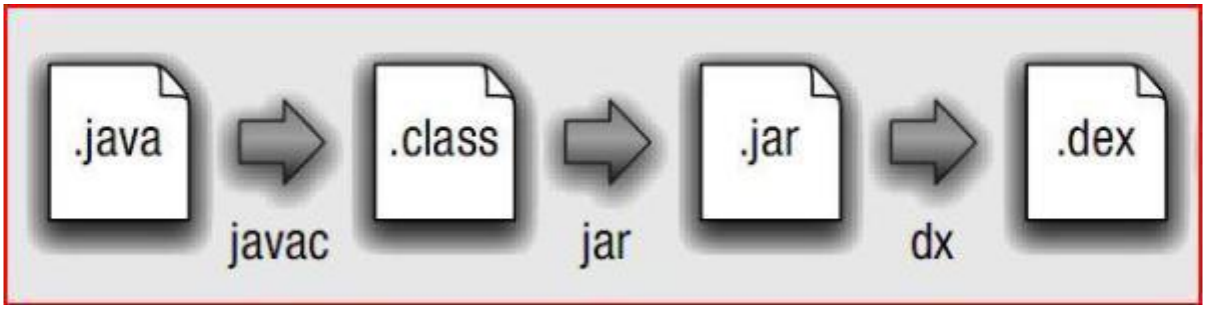


Fig.5 : Conversion from .java to .dex file

As the outcome, it is conceivable to have different examples of Dalvik virtual machine running on the single gadget in the meantime. The Core libraries are composed in Java dialect and contains of the accumulation classes, the utilities, IO and different devices.

2.4 Linux Kernel

Android Architecture relies on upon Linux 2.6 piece. It supervises security, memory organization, get ready organization, mastermind stack and other indispensable issues. Thusly, the customer should get Linux his PDA as the crucial working structure and present each one of the drivers required remembering the true objective to run it. Android gives the support to the Qualcomm MSM7K chipset family. For instance, the present segment tree reinforces Qualcomm MSM 7200A chipsets, however in the second half of 2008 we should see PDAs with stable adjustment Qualcomm MSM 7200, which joins critical segments :

1. WCDMA/HSUPA and EGPRS network support Bluetooth 1.2 and Wi-Fi support

2. Digital audio support for mp3 and other formats

3. Support for Linux and other third-party operating systems

4. Java hardware acceleration and support for Java applications

5. Qcamera up to 6.0 megapixels

2.5 Architecture for Secure Data Storage

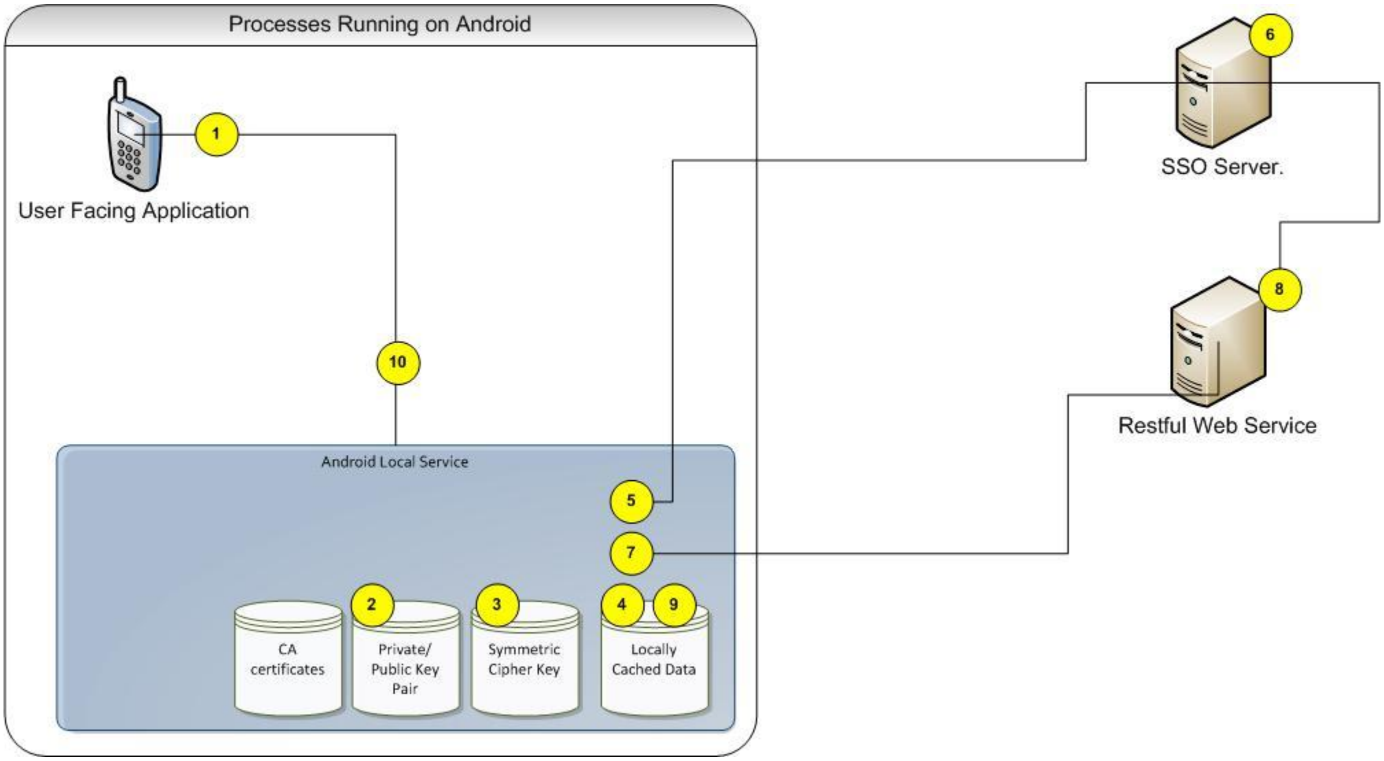


Fig. 6 - Architecture for secure data storage

Secure information stockpiling arrangement that could possibly be conveyed on Android. It is as appeared in the figure. Nonetheless, numerous inadequacies of the outline have been tended to. Extra security highlights will be exhibited toward the finish of the segment.

Using this figure , I have the following workflow:

1. The private key is utilized to unscramble the symmetric figure key. The symmetric figure key is used to scramble/decode any privately stored information. A solid symmetric figure like 3DES is utilized .

2. All data found in the local cache is encrypted with the symmetric cipher key defined in step #1 .

3. In the event that the asked for information is not privately stored or lapsed. We should speak with the SSO benefit again to have the capacity to get new information from the Restful web administrations. In any case, dissimilar to the engineering exhibited in area 2 of this report, we login to the SSO server utilizing an antagonistic test in view of the private key of the client. In that capacity, we login with the SSO framework utilizing open/private key foundation. The client name and the watchword are never sent over the system. The SSO framework can recognize the client in light of this test and returns a 496 piece alpha numeric token.

4. The tokens generated by the SSO system are set to automatically expire after a given period of time.

5. On social event of the SSO token. The Android establishment application can now talk with any Restful web benefits that stick to the same SSO alliance. Open/private key establishment is by the day's end used to setup an ensured correspondence station between the phone and the server. The presentations of the servers that host the web organizations are gotten from a comparable confirmation master that shipped with the phone .

6. On reception of a request, the SSO token is extracted from the request. The web service calls upon the SSO system to authorize the operation .

7. On social event of the data, the symmetric figure depicted in slug #1 above is used to encode the data before it accomplishes any close-by decided storeroom .

8. Data is returned to the user facing application.

2.6 Android Version History

Android version started with the release of the version Android beta in November 2007, but the first commercial version released in September 2008.

Android Version	**Name**	**Release Date**
Android 1.0	Alpha	Sep 2008
Android 1.1	Beta	Feb 2009
Android 1.5	Cupcake	April 2009
Android 1.6	Donut	Sep 2009
Android 2.0-2.1	Eclair	Oct 2009
Android 2.2-2.2.3	Froyo	May 2010
Android 2.3-2.3.7	Ginger Bread	Dec 2010
Android 3.0-3.2.6	Honey Comb	Feb 2011
Android 4.0-4.0.4	Ice Cream Sandwich	Oct 2011
Android 4.1-4.3.1	Jelly Bean	June 2012
Android 4.4-4.4.4	Kit Kat	Sep 2013
Android 5.0-5.1.1	Lollypop	Nov 2014
Android 6.0-6.0.1	Marshmallow	Oct 2015
Android 7.0-7.1.2	Nougat	Aug 2016

Table : 1 – Android Version

Now the next figure will show which version used how much:

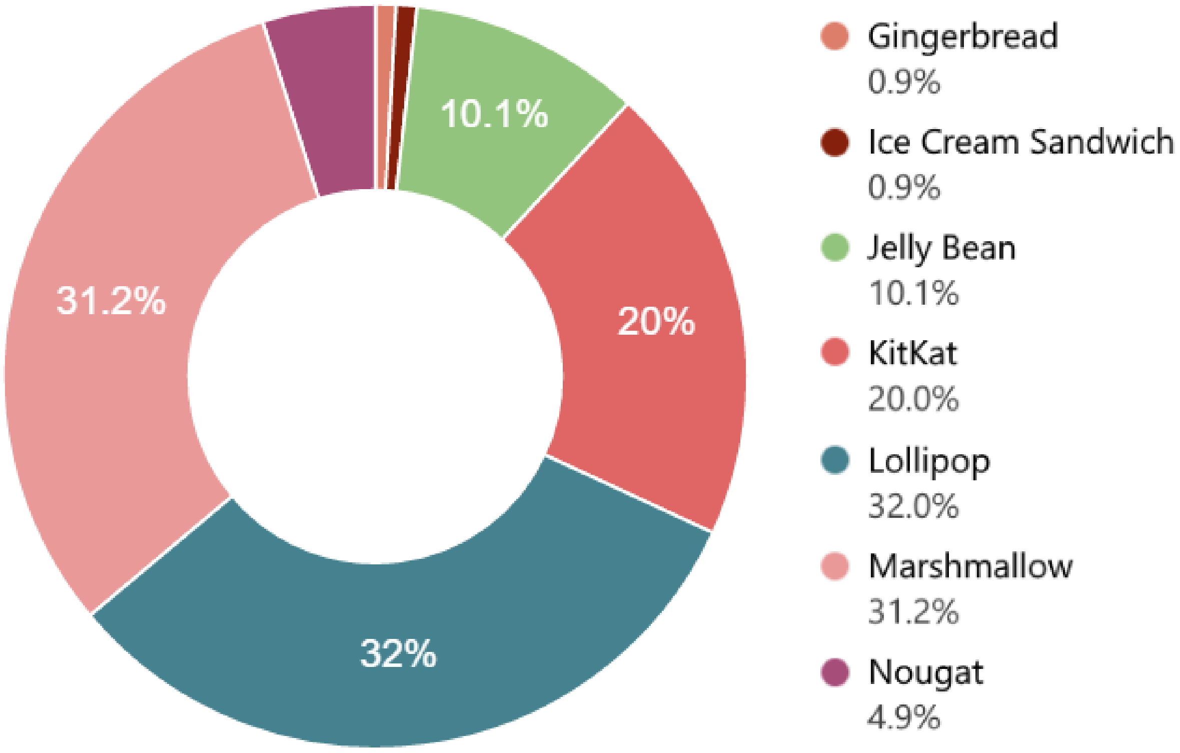


Fig. 7 - Android version use

CHAPTER 3

SYSTEM DEVELOPMENT

There are many ways for development of android applications. My project is based on android application so I have used Android Studio for development of my project. Phonebook directory application has the ability to search, view and manage entries. The search result gives options to the user like CALL, SAVE, SEND, VIEW, DISCARD etc. This system developed according to the android architecture whose basics given in chapter 2.

3.1 Project Requirement

For development of this application, I have used ANDROID as Front End and SQLite as Back End.

3.1.1 Front End

Android version - 4.3(Jelly Bean)

Android Studio – 2.1.2

Fig. 8 – Android Jelly Bean

Android studio is most usable and easy development tool for android applications that's why I have used it. Android SDK is the official IDE for development of android applications. Android SDK have a set of application programming API's.

Android SDK offers :

- Instant run
- Intelligent code editor
- Fast and rich emulator
- Flexible built system
- Optimized for all android devices
- Testing tools and frameworks are given
- Layout editor
- APK Analyzer

3.1.2 Back End

SQLite

Fig. 9 – SQLite diagram

SQLite is an library which implements a self-contained, serverless, zero-configuration, transactional SQL database engine. We can link SQLite engine statically or dynamically as per our application requirements.

SQLite contains database management classes which are used by application to manage its private database. If creating a content provider, than we have to use these classes to store content.

See Content Providers to take in the traditions for executing a substance supplier. See the Notepad Provider class in the Notepad test application in the SDK for a case of a substance supplier. Android ships with SQLite form 3.4.0.

My project allows the user to use this application for different purpose through different modules provided in application.

I have used SQLite because it offers :

- It does not require separate system to operate as it is serverless.
- It comes with zero-configuration, that means no administration required.
- It is stored in a single cross-platform disk file.
- It is very small and light weight.

3.2 Application Development

I have used layered approach for developing this android application.

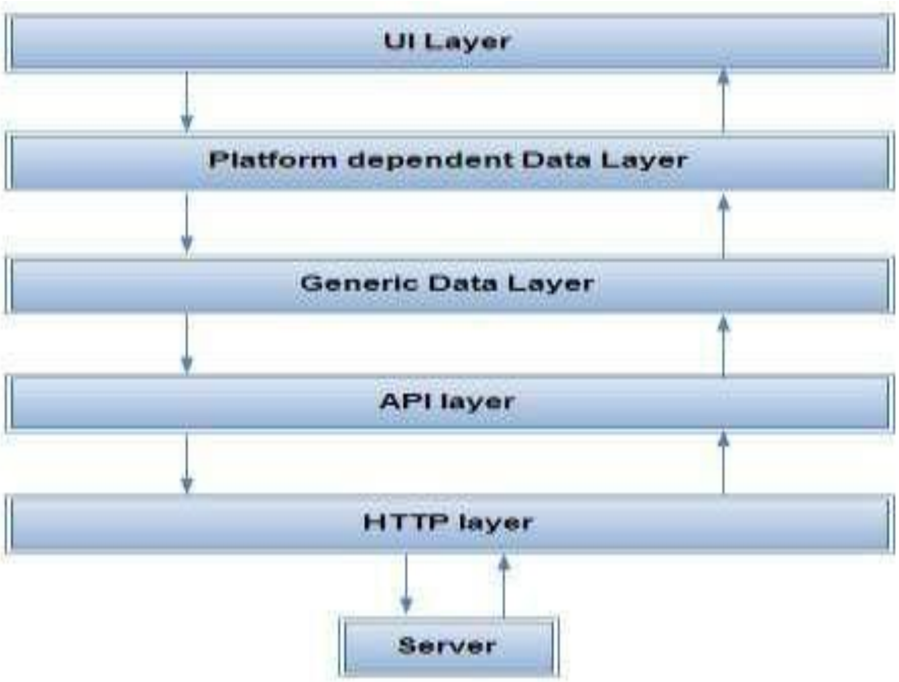


Fig. 10 – Layered Architecture

- HTTP layer is responsible for sending HTTP request to the server and receiving response back.
- API layer is used for parsing the response string.
- Generic data layer is used for designing business layers and implementing functionalities.
- Platform dependent data layer takes the data from API layer and use it.
- Last one, UI layer is used to show data to user and manages user interactions.

3.3 Algorithm for development of android application

I have used these steps for developing this application, which are given below :

1. Written in java language using Android SDK.

2. Java source file converted into java class file by java compiler.

3. The android SDK have a tool dx which is used to convert java class file into a .dex file.

4. The .dex file and the resources of project are packed into an .apk file.

5. This is how finally whole application created and deployed.

CHAPTER 4

PERFORMANCE ANALYSIS

4.1 Lifecycle of an Android Application

Much of the time, each Android application keeps running in its own Linux procedure. This procedure is made for the application when some of its code should be run, and will stay running until it is did not require anymore and the framework needs to recover its memory for use by different applications.

A critical and abnormal element of Android is that an application procedure's lifetime is not specifically controlled by the application itself. Rather, it is controlled by the framework through a mix of the parts of the application that the framework knows are running, how imperative these things are to the client, and how much general memory is accessible in the framework.

It is essential that application engineers see how diverse application segments (specifically Activity, Service, and IntentReceiver) affect the lifetime of the application's procedure.

Not utilizing these segments accurately can bring about the framework executing the application's process while it is doing imperative work.

A typical case of a procedure life-cycle bug is an IntentReceiver that begins a string when it gets an Intent in its on ReceiveIntent() technique, and after that profits from the capacity. When it gives back, the framework views that Intent Receiver as no longer dynamic, and in this manner its facilitating procedure did not require anymore (unless other application parts are dynamic in it). Hence, it might kill the procedure whenever to recover memory, ending the generated string that is running in it. The answer for this issue is to begin a Service from the Intent Receiver, so the framework realizes that there is as yet dynamic work being done simultaneously.

To figure out which procedures ought to be executed when low on memory, Android places them into a "significance chain of command" in light of the segments running in them and the condition of those segments. These are, all in order of importance:

1. A foreground procedure is one holding an Activity at the highest point of the screen that the client is associating with (its onResume () technique has been called) or an IntentReceiver that is as of now running (its onReceiveIntent () strategy is executing). There will just ever be a couple of such procedures in the framework, and these may be slaughtered if all else fails if memory is low to the point that not even these procedures can keep on running. By and large now the gadget has achieved a memory paging state, so this activity is required so as to keep the UI responsive.

2. A visible procedure is one holding an Activity that is unmistakable to the client on-screen yet not in the frontal area (its onPause() strategy has been called). This may happen, for instance, if the closer view movement has been shown with a discourse appearance that enables the past action to be seen behind it. Such a procedure is considered critical and won't be slaughtered unless doing as such is required to keep all closer view forms running.

3. An service procedure is one holding a Service that has been begun with the startService() strategy. In spite of the fact that these procedures are not specifically obvious to the client, they are by and large doing things that the client thinks about, (for example, foundation mp3 playback or foundation organize information transfer or download), so the framework will dependably keep such procedures running unless there is insufficient memory to hold all closer view and noticeable process.

4. A background procedure is one holding an Activity that is not right now unmistakable to the client (its onStop() technique has been called). These procedures have no immediate effect on the client encounter. If they execute their movement life cycle accurately (see Activity for more subtle elements), the framework can kill such procedures whenever to recover memory for one of the three past procedures sorts. More often than not there are large portions of these procedures running, so they are kept in a LRU rundown to guarantee the procedure that was most as of late observed by the client is the last to be murdered when running low on memory.

5. A void procedure is one that doesn't hold any dynamic application segments. The main motivation to keep such a procedure around is as a reserve to enhance startup time whenever a segment of its application needs to run. In that capacity, the framework will regularly murder these procedures with a specific end goal to adjust general framework assets between these purge stored forms and the fundamental portion reserves.

When choosing how to characterize a procedure, the framework picks the most critical level of the considerable number of segments right now dynamic all the while.

4.2 Additional Security Notes

1. People in general/private key match of the client is created straightforwardly on the handset at introduce time. In that capacity, the private key has never left the telephone nor has it been exchanged over any system.

2. The endorsement of the client should at any rate be enrolled once in the SSO application. This should be possible at introduce time of the handset application.

3. "Man-in-the-middle"38 assaults are impractical since the application is sent with the CA endorsement of the organization that will have the web administrations.

4. In the event that the gadget is lost, all the privately reserved information is totally incoherent. The symmetric key that encoded this information is additionally mixed up. People in general/private keys that are fundamental to the security engineering are ensured by a passphrase.

5. The passphrase is the weakest connection in the chain. On the off chance that the client enters an excessively straightforward secret key, get to could be picked up to the private key and thus the privately reserved information.

6. That being stated, it is conceivable to additionally extend this design to store the scrambled symmetric key on the server. Along these lines, regardless of the possibility that the passphrase of the private key is traded off, the privately stored information still can't be gotten to. This is on the grounds that the encoded solid symmetric figure key is put away on the server. When the passphrase has been broken, there has been plentiful time to report the stolen telephone and deny this key from this client account on the server. Besides, under this plan, the key put away on the server is still encoded. Regardless of the possibility that this key is caught in travel it is futile without the client's private key.

7. It is additionally conceivable to implement a solid watchword strategy specifically from the handset application.

8. Regardless of the possibility that this plan is altogether more secure than the past emphasis, to the client, the experience is the same. The client must enter a username and secret word to demonstrate his distinguish.

9. We could increase the engineering in yet another course. The nearby reserving framework could likewise require a SSO token and in this manner ask for approval from a SSO framework.

4.3 Security and Permissions in Android Application

Android is a multi-handle framework, where every application (and parts of the framework) keeps running in its own particular procedure. Most security amongst applications and the framework is implemented at the procedure level through standard Linux offices, for example, client and gathering IDs that are appointed to applications. Extra better grained security elements are given through a "consent" component that authorizes limitations on the particular operations that a specific procedure can perform.

Android cell phone stage will be more secure than Apple's iPhone or whatever other gadget over the long haul. There are a few arrangements these days to shield Google telephone from different assaults. One of them is security seller McAfee, an individual from Linux Mobile (LiMo) Foundation. This establishment joins specific organizations to build up an open cell phone programming stage. A large number of the organizations recorded in the LiMo Foundation have likewise progressed toward becoming individuals from the Open Handset Alliance (OHA).

Therefore, Linux secure coding practice ought to effectively be incorporated with the Android improvement prepare. Be that as it may, open stage has its own particular inconveniences, for example, source code powerlessness for dark cap programmers. In parallel with incredible open doors for versatile application designers, there is a desire for abuse and mischief. Stealthy Trojans covered up in energized pictures, specific infections gone from companion to companion, utilized for spying and data fraud, every one of these dangers will be dynamic for a long run.

Another answer for such assaults is SMobile Systems versatile bundle. Security Shield –an incorporated application that incorporates against infection, hostile to spam, firewall and other portable assurance is up and prepared to keep running on the Android working framework. Right now, the fundamental issue is accessibility for infections to act like an application and do things like dial telephone numbers, send instant messages or multi-media messages or make associations with the Internet amid ordinary gadget utilize. It is feasible for some individual to utilize the GPS highlight to track a man's area without their insight. Subsequently SMobile Systems is prepared to tell and square these protected cautions. In any case, in all actuality it is unrealistic to secure r cell phone or PC totally, as it interfaces with the web. What's more, neither the Android telephone nor different gadgets will turn out to be the exemption.

4.4 Development Tools

The Android SDK incorporates an assortment of custom apparatuses that help create portable applications on the Android stage. The most imperative of these are the Android Emulator and the Android Development Tools module for Eclipse, yet the SDK additionally incorporates an assortment of different devices for investigating, bundling, and introducing r applications on the emulator.

Android Emulator

A virtual cell phone that keeps running on PC utilize the emulator to configuration, investigate, and test r applications in a genuine Android run-time condition.

Dalvik Debug Monitor Service (ddms)

Coordinated with Dalvik, the Android stage's custom VM, this apparatus lets oversee forms on an emulator or gadget and helps with investigating. can utilize it to slaughter forms, select a particular procedure to troubleshoot, create follow information, see store and string data, take screenshots of the emulator or gadget, and the sky is the limit from there.

Android Debug Bridge (adb)

The adb instrument lets introduce application's .apk documents on an emulator or gadget and get to the emulator or gadget from a summon line. can likewise utilize it to connect a standard debugger to application code running on an Android emulator or gadget.

Android Asset Packaging Tool (aapt)

The aapt device lets make .apk documents containing the pairs and assets of Android applications.

Android Interface Description Language (aidl)

Aidl Lets produce code for an interprocess interface, for example, what an administration may utilize.

sqlite3

Included as an accommodation, this device lets get to the SQLite information records made and utilized by Android applications.

mksdcard

Makes a circle picture that can use with the emulator, to mimic the nearness of an outside capacity card, (for example, a SD card).

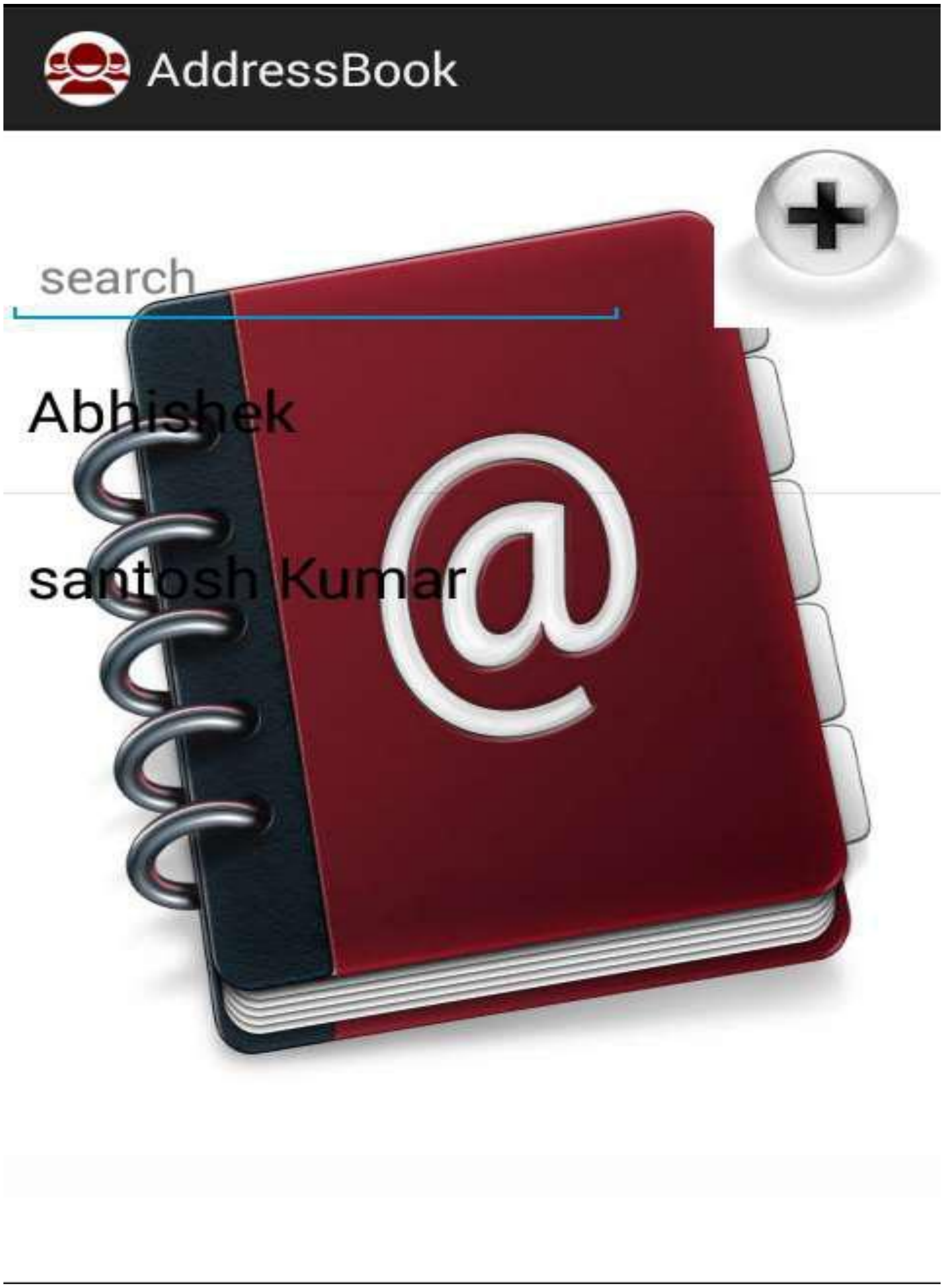
dx

The dx device reworks .class bytecode into Android bytecode (put away in .dex documents.)

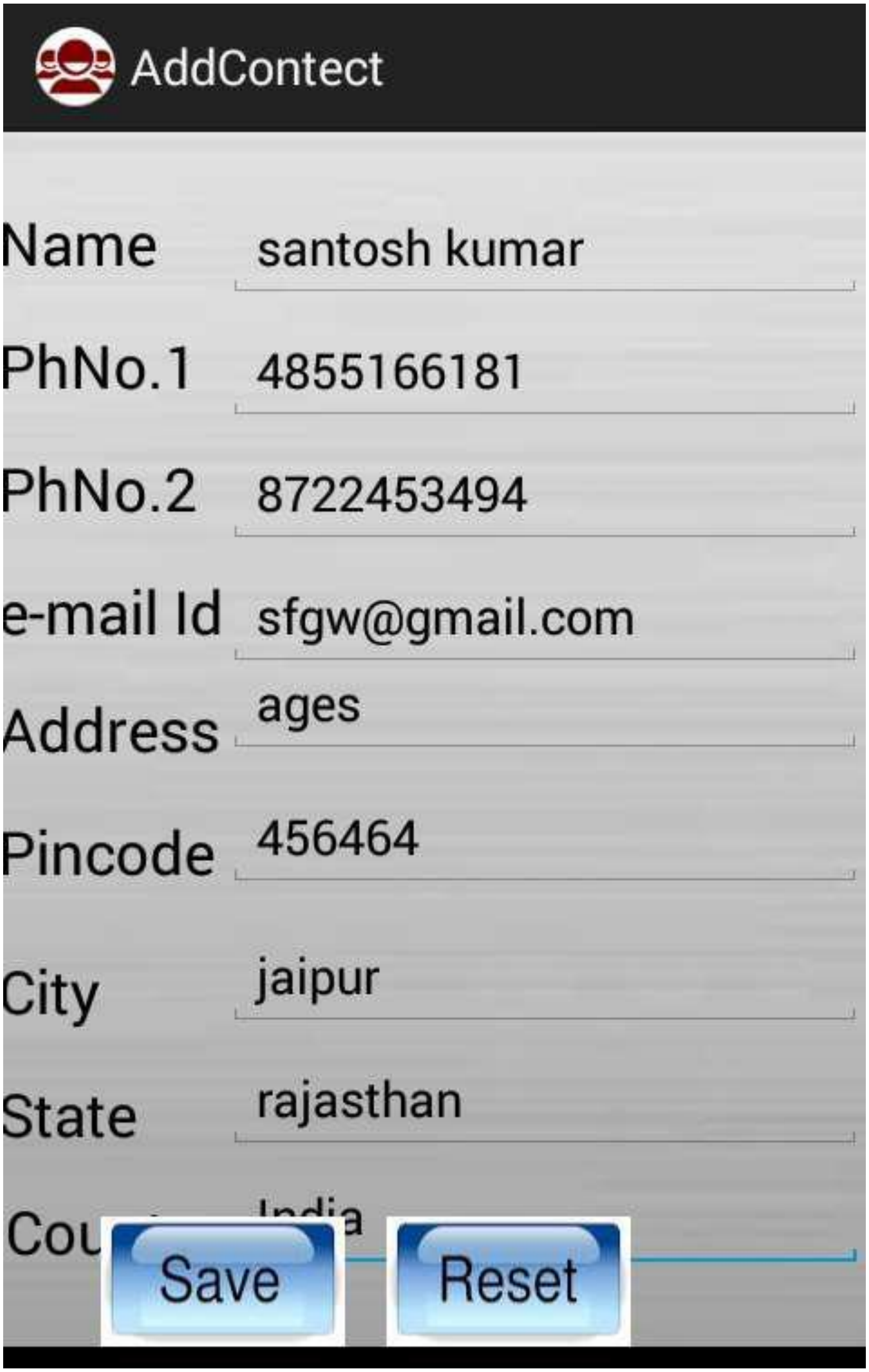
CHAPTER 5

Snapshots

5.1 Search View



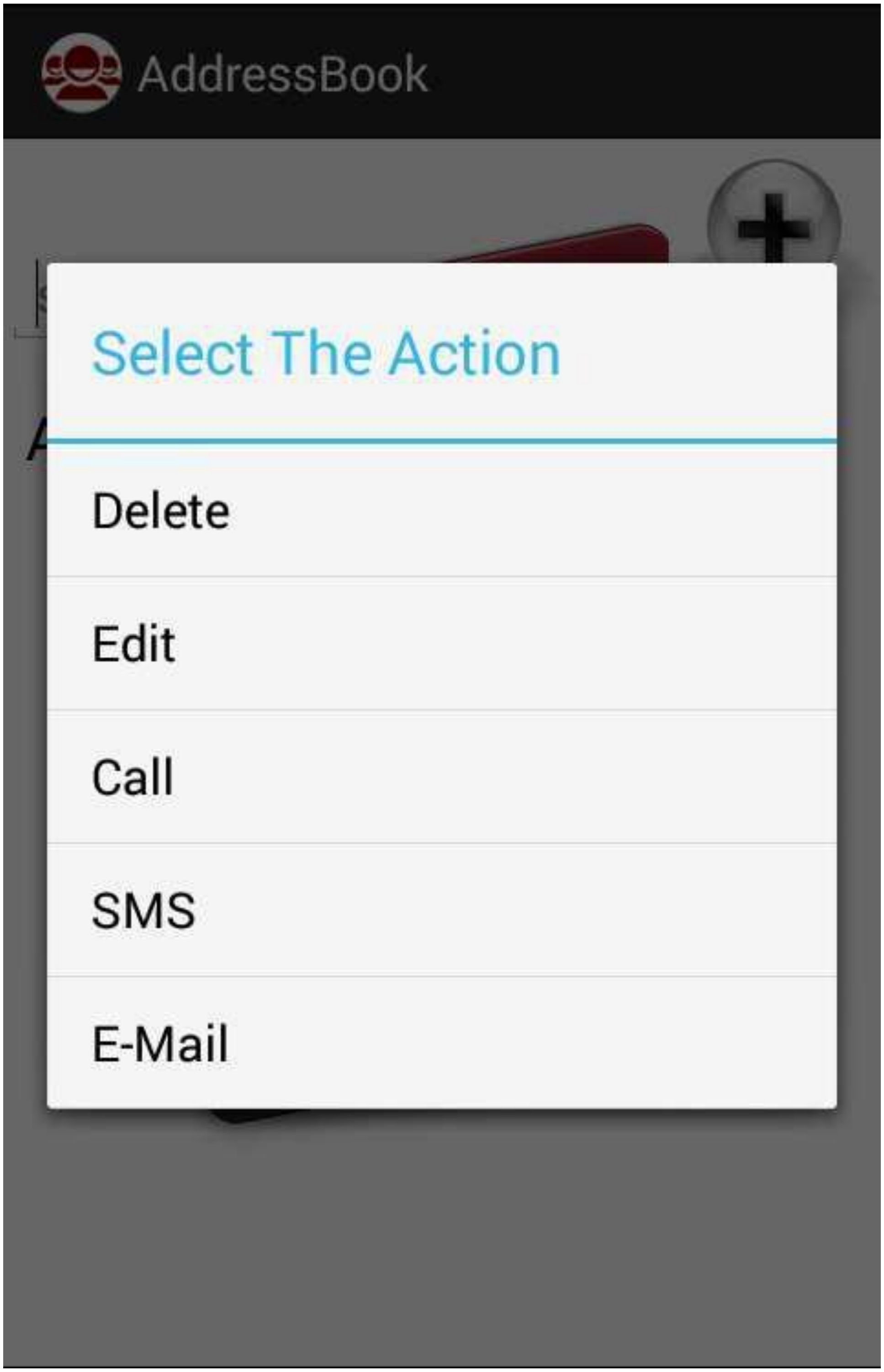
5.2 Add Contact View



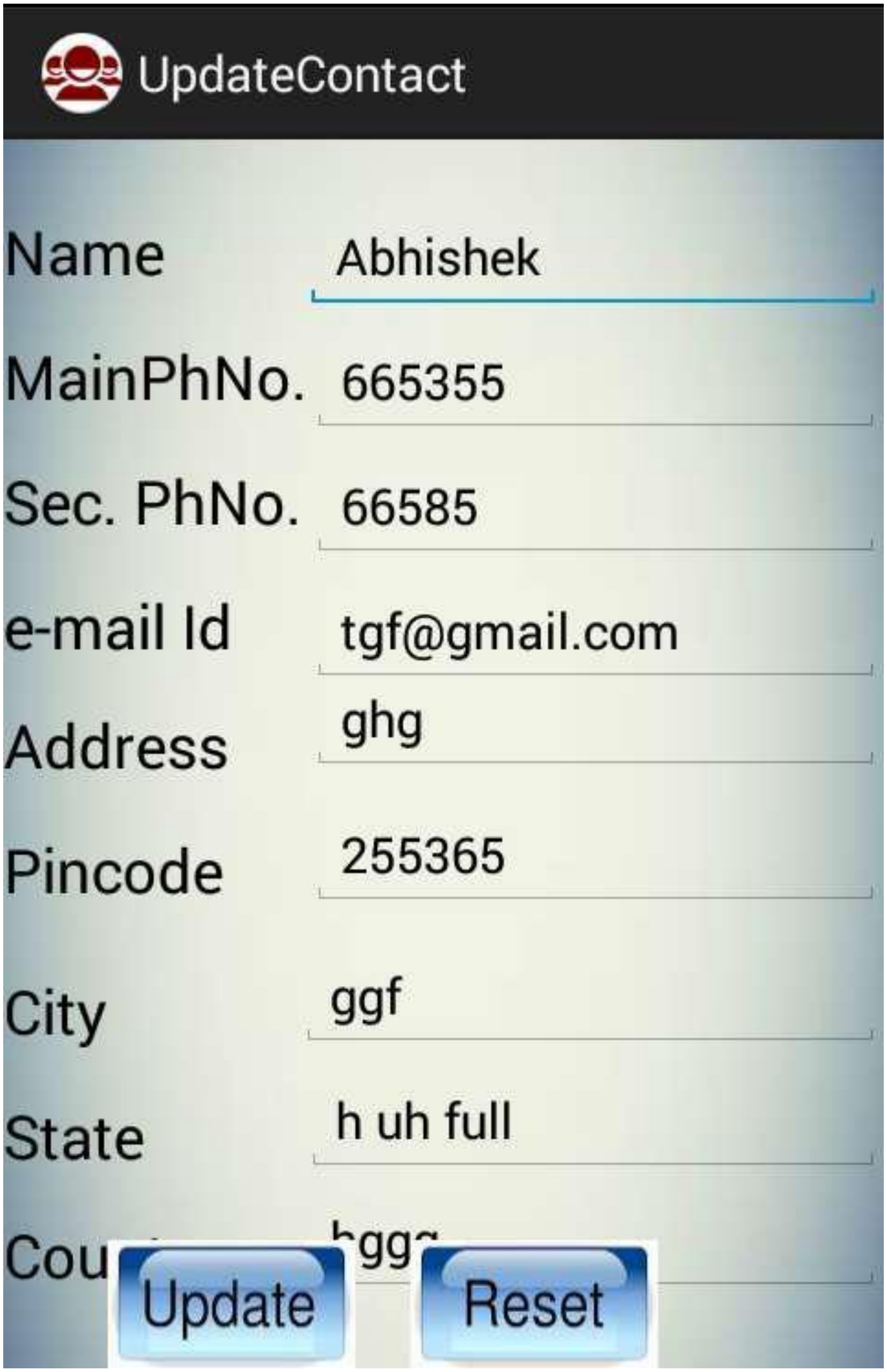
AddContect

Name	santosh kumar
PhNo.1	4855166181
PhNo.2	8722453494
e-mail Id	sfgw@gmail.com
Address	ages
Pincode	456464
City	jaipur
State	rajasthan
Cou	India

5.3 Action View

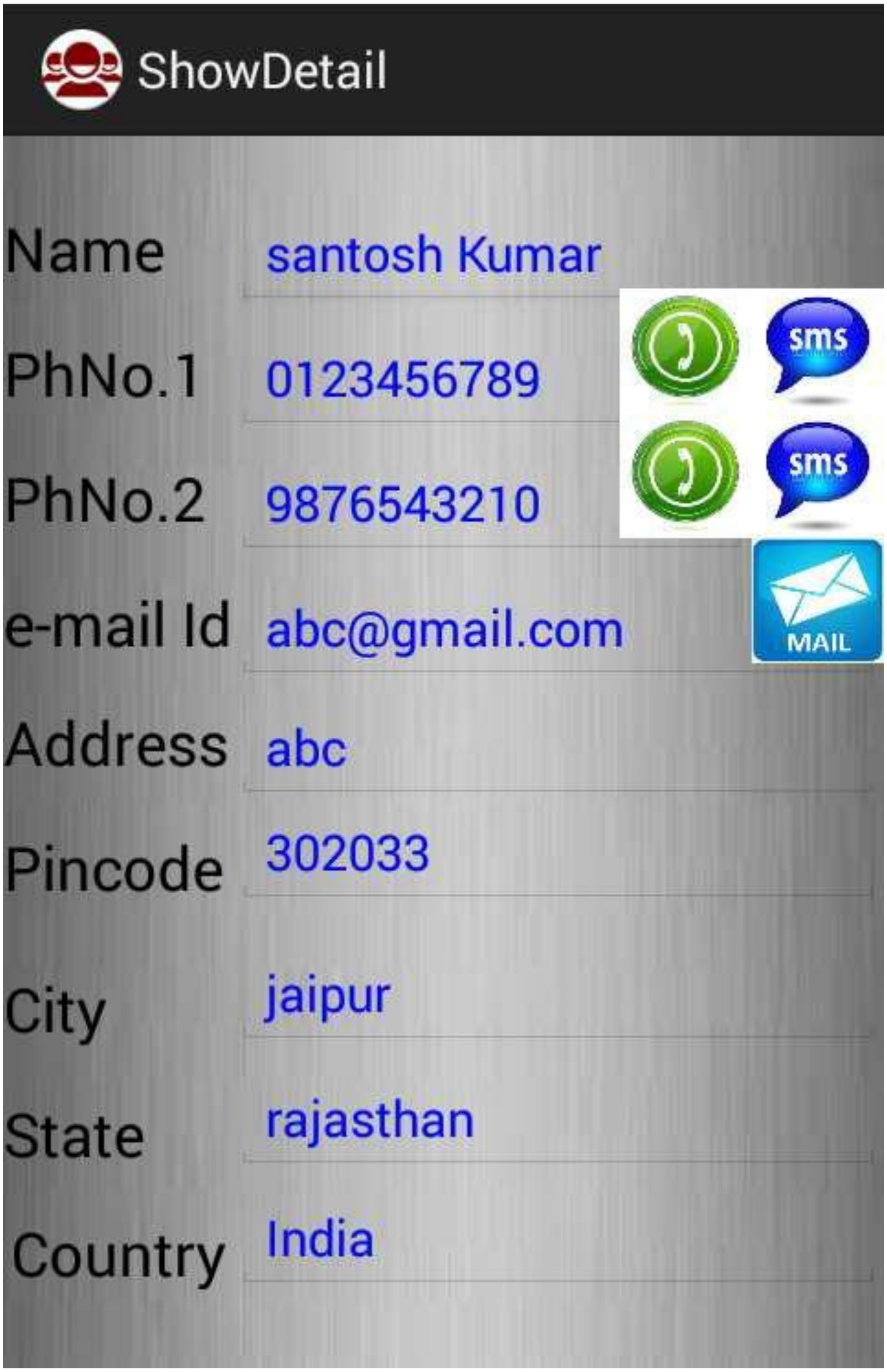


5.4 Update Contact



Name	Abhishek
MainPhNo.	665355
Sec. PhNo.	66585
e-mail Id	tgf@gmail.com
Address	ghg
Pincode	255365
City	ggf
State	h uh full
Cou	hgg

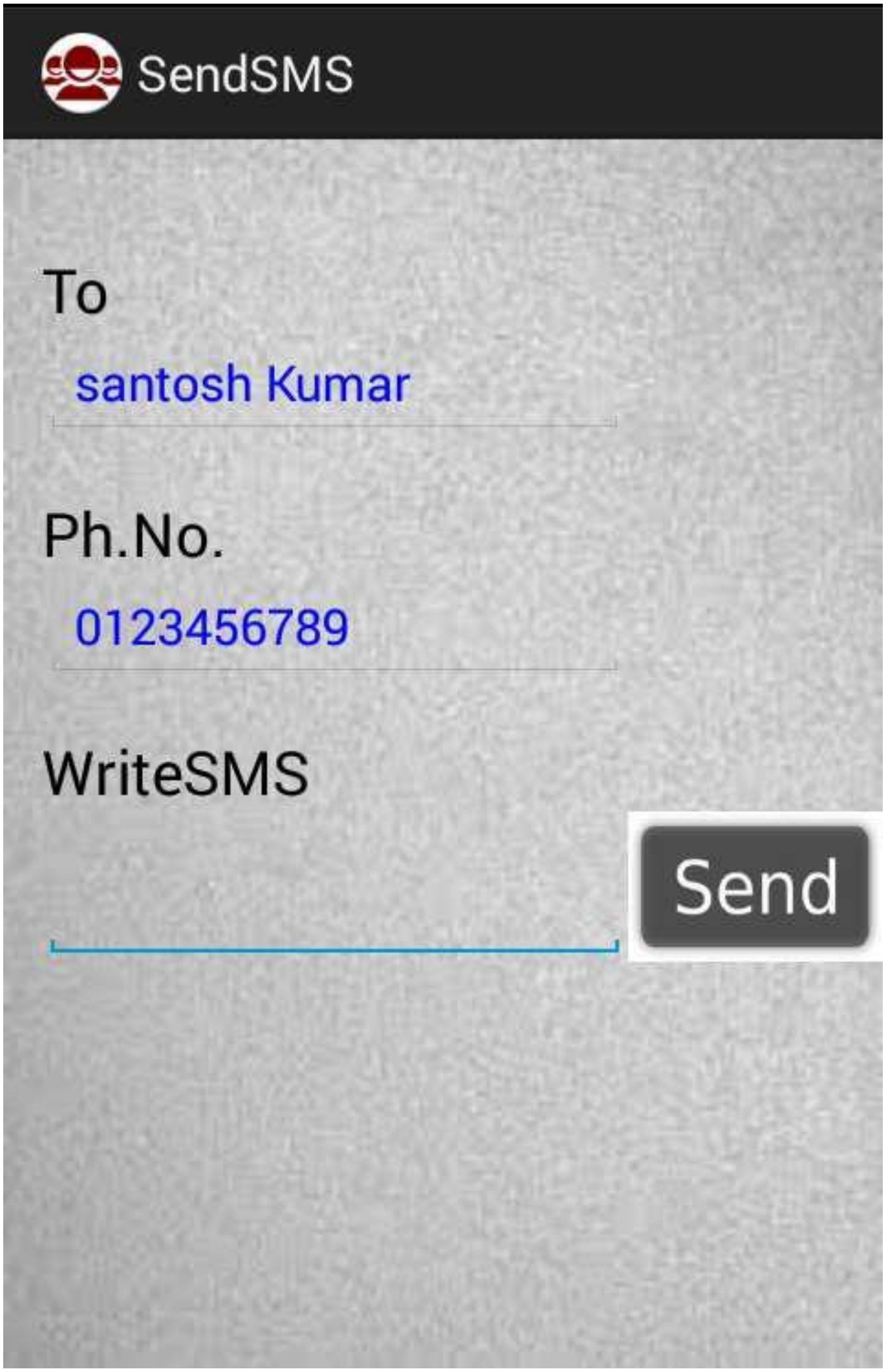
5.5 Show Details View



ShowDetail

Name	santosh Kumar
PhNo.1	0123456789
PhNo.2	9876543210
e-mail Id	abc@gmail.com
Address	abc
Pincode	302033
City	jaipur
State	rajasthan
Country	India

5.6 Sending SMS View



To

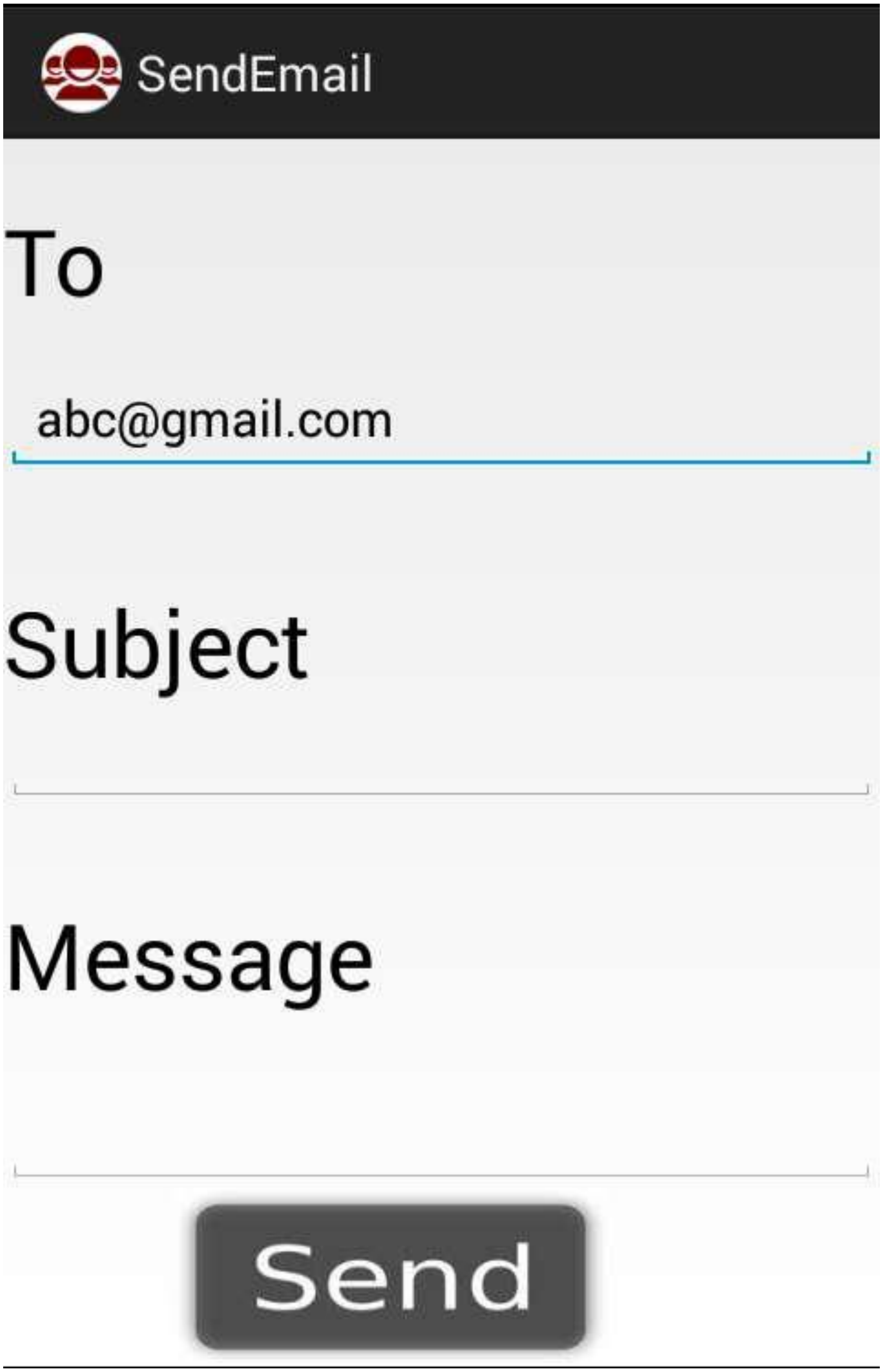
santosh Kumar

Ph.No.

0123456789

WriteSMS

5.7 Sending Email View



To

abc@gmail.com

Subject

Message

Conclusion

Android is a genuinely open, free improvement platform in light of Linux and open source. Handset producers can utilize and alter the stage without paying a sovereignty.

A part based engineering motivated by Internet concoction. Parts of one application can be utilized as a part of another in ways not initially imagined by the designer can even supplant worked in segments with claim enhanced renditions. This will unleash another round of innovativeness in the portable space.

- Android is interested in all: industry, designers and clients

- Participating in a large number of the fruitful open source ventures

- Aims to be as simple to work for as the web.

- Google Android is venturing into the following level of Mobile Internet

dx

The dx instrument changes .class bytecode into Android bytecode (put away in .dex documents.)

The computerization of this project won't just enhance the productivity however will likewise decrease human stress accordingly in a roundabout way enhancing HR.

Future Scope

- Android Developers have built outstanding careers through the development activity. Demand for new apps and enhancing the existing apps is increasing.

- It has opened a new stream of technological advancements. And interestingly, there seems no end to experimenting and evolving new applications and mobile phones using this innovative and dynamic technology.

- The android Learning app gets updated with new modules every fixed period of time. The future of the interactive learning.

References

- Kamaruddin Mamat, Farok Azmat (InCULT 2012) “Mobile Learning Application for Basic Router and SwitchConfiguration on Android Platform” Procedia - Social and Behavioral Sciences 90 (2013) 235 – 244.
- Sharples, M. (2005). Disruptive devices: mobile technology for conversational learning. International Journal of Continuing Engineering Education and Life Long Learning, 12(5/6), 504-520.
- Jones V., Jo J. H., Cranitch G.: HyWeb: A Holistic Approach to Technology -based Tertiary Education, Proc. of 14th World Conference on Educational Multimedia, Hypermedia and Telecommunications , Denver, CO, USA , June 2012
- Chang C., Sheu J.(2007). Design and Implementation of Ad Hoc Classroom and eSchoolbag Systems for Ubiquitous Learning, Proc. of IEEE International Workshop on Wireless and Mobile Technologies in Education (WMTE 2002) , (p. 8-14).
- Botzer, G and Yershalmy, M. (2007) Mobile application for mobile learning. Paperpresented at IADIS International Conference on Cognition and Exploratory Learning in Digital Age (CELDA 2007).
- P. Pocatilu, F. Alecu and M. Vetrici (2010). Measuring the Efficiency of Cloud Computing for E-learning Systems, WSEAS TRANSACTIONS on COMPUTERS, Issue1, Volume 9, January 2010, pp. 42-51.
- Mahmoud, Q.H and Popowicz, P. (2010). A mobile application development approachto teaching introductory programming. Paper presented at Frontiers in Education Conference (FIE), 2010.
- Potts, J, Moore, N and Sukittanon, S. (2011). Developing Mobile Learning Applications for Electrical Engineering Courses. Paper presented at Southeaston, 2011 Proceedings of IEEE.
- Klopfer E., Squire K., Jenkins H.: Environmental Detectives: PDAs as a Window into a Virtual Simulated World, Proc. of IEEE International Workshop on Wireless and Mobile Technologies in Education , (p. 95-98), Växjö, Sweden, August 2011
- http://code.google.com/android/
- http://developer.android.com/