A Project Report on

Android Crime Reporter and Missing Person Finder

Submitted to

JAWAHARLAL NEHRU TECHONOLOGICAL UNIVERSITYANANTAPUR, ANANTAPURAM

In partial fulfilment of the requirement for the award of degree

MASTER OF COMPUTER APPLICATIONS

By

AVETA SHIVASANKAR

18BF1F0003

Under the esteemed guidance of

Prof. K SEKAR

DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS

SRI VENKATESWARA COLLEGE OF ENGINEERING

Opp. LIC Training Center, Karakambadi Road, Tirupati – 517507, A.P.
Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapur
Recognized under sections 2(f) & 12(B) of UGC act 1956, Accredited by NAAC with ‘A’ Grade
Three B.Tech Programmes CSE, ECE & EEE are accredited by NBA, New Delhi

SRI VENKATESWARA COLLEGE OF ENGINEERING

Opp. LIC Training Center, Karakambadi Road, Tirupati – 517507, A.P.
Approved by AICTE, New Delhi & Affiliated to JNTUA, Anantapur
Recognized under sections 2(f) & 12(B) of UGC act 1956, Accredited by NAAC with 'A' Grade
Three B.Tech Programmes CSE, ECE & EEE are accredited by NBA, New Delhi
DEPARTMENT OF MASTER OF COMPUTER APPLICATIONS 2018-2021

CERTIFICATE

This is to certify that the project report entitled **"Android Crime Reporter and Missing Person Finder"** ***a bonafide record of the project work done and submitted by***

AVETA SHIVASANKAR

(18BF1F0003)

for the partial fulfillment of the requirements for the award of **MASTER OF COMPUTER APPLICATIONS** *Degree from JNT University Anantapur,Ananthapuramu*.

GUIDE HEAD OF THEDEPARTMENT

INTERNALEXAMINER **EXTERNAL EXAMINER**

ACKNOWLEDGEMENT

I am thankful to my guide **Prof. K SEKAR** for his valuable guidance and encouragement. His helping attitude and suggestions have helped me in the successful completion of the project**.**

I express my thanks to **Prof. K SEKAR**, Vice Principal & Head of the Department of MASTER OF COMPUTER APPLICATIONS, for his kind help and encouragement during the course of my study and in the successful completion of the project work.

I would like to express my sincere thanks to **DR. N. SUDHAKAR REDDY,**Principal, S.V College of Engineering, Tirupati.

Successful completion of my project cannot be done without proper support and encouragement. My sincere thanks to the Management for providing all the necessary facilities during the Course of my study.

I would like to thank my parents and friends, who have the greatest contributions in all my achievements, for the great care and blessings in making me successful in all my endeavors.
I would like to express my deep gratitude to all those who helped me directly or indirectly to transform an idea into my working project.

AVETA SHIVASANKAR
(18BF1F0003)

DECLARATION

I hereby declare that the project entitled **"Android Crime Reporter and Missing Person Finder"** submitted to the Department of MASTER OF COMPUTER APPLICATIONS,**S.V. COLLEGE OF ENGINEERING, TIRUPATI** in partial fulfillment of requirements for the award of the degree of **MASTER OF COMPUTER APPLICATIONS**.

This project is the result of my own effort and it has not been submitted to any other University or Institution for the award of any degree or diploma other than specifiedabove.

Signature of the student
AVETA SHIVASANKAR
(18BF1F0003)

ABSTRACT

The project titled as “Android Crime Reporter and Missing Person Finder “is an android application. This app provides facility for reporting online crimes, complaints, missing persons, show most wanted person details. In existing system if any user can give complaints on any issue means they can go to police station and give complaint or crime details. So it takes time to go to police station and give complaint. To overcome this problem we move to proposed model called Crime Reporter and Missing Person Finder. In this project the user can give any complaints through online. Here the admin can respond all the user complaints and crime details and he has an ability to add missing person details and hot news. Missing person details and hot news can be viewed by all the users who are in this app. finally user can get the response for all the crime and complaints from admin.

INDEX

LIST OF FIGURES

1. Introduction

Scope:

To have a general awareness it will be more convenient for department regarding information on the courses, materials, syllabus, subjects, and timetables. It also provides the support that student can also check about daily schedules, notifications, events. This application provides a virtual tour of department.

Objective

- ☐ The project "Web Based Application to Preserve MCA[SVCE] Department Data" is a software package developed to manage department data like materials , syllabus, notifications, events, placement, etc. The software is very helpful to maintain the department from anywhere. The Project can manage the department details by storing it in a server database. The student will get information and the admin has to upload the data regarding materials, syllabus and notifications, etc. Admin can perform the following operations: Uploading details of materials.
- ☐ Uploading details of individual syllabus and daily schedules.
- ☐ Uploading details of events and placement information.

Description of the project

It is well known that maintaining the department information where students and faculty access the information regarding entire department like materials and syllabus, events and placement information, etc. This hectic process is done by human in which it takes lots of time, to overcome this burden the whole process is computerized in such a way that a dynamic web page is created where admin should logged in to their account to update the information.

The project admin will generate the databases for the storing information. The students will view the entire information which is uploaded by admin and they can access whenever they need or they can view. The student their see their department information.

Proposed System

In this project the user can give any complaints through online. Here the admin can respond all the user complaints and crime details and he has an ability to add missing person details and hot news.missing person details and hot news can be viewed by all the users who are in this app. finally user can get the response for all the crime and complaints from admin.

Advantages of the Proposed System

- Less time.
- No manual calculation.
- High performance.

2. Literature Survey

A.Manual System vs B. Android application

Finding lost person can be difficult task. The currently available Manual System for finding missing person has very long procedure and takes more time. More time is required for launching an FIR (First Information Report) in police station. Also time required for finding lost person is more. Also during manual process number of manpower for searching lost person is less. And in some missing person related website they required FIR No for upload complaint on their website. Relative of lost person will go to trust rather than going in police station to launch complaint. Trust user will add their complaint in our Android Application. This information will be stored on server which can be accessed by all trust members so that they can find lost children in specific region

3. System Requirements Analysis

System Requirement Specification:

Software environment:

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. Google Inc. purchased the initial developer of the software, Android Inc., in 2005.

Android's mobile operating system is based on the Linux kernel. Google and other members of the Open Handset Alliance collaborated on Android's development and release.

The Android Open Source Project (AOSP) is tasked with the maintenance and further development of Android. The Android operating system is the world's best-selling Smartphone platform.[

The Android SDK provides the tools and APIs necessary to begin developing applications Android platform using the Java programming language. Android has a large community of developers writing applications ("apps") that extend the functionality of the devices. There are currently over 250,000 apps available for Android.

.Features

- **Application framework** enabling reuse and replacement of components
- **Dalvik virtual machine** optimized for mobile devices
- **Integrated browser** based on the open source WebKit engine
- **Optimized graphics** powered by a custom 2D graphics library; 3D graphics based on the OpenGL ES 1.0 specification (hardware acceleration optional)
- **SQLite** for structured data storage
- **Media support** for common audio, video, and still image formats (MPEG4, H.264, MP3, AAC, AMR, JPG, PNG, GIF)
- **GSM Telephony** (hardware dependent)

- **Bluetooth, EDGE, 3G, and WiFi** (hardware dependent)
- **Camera, GPS, compass, and accelerometer** (hardware dependent)
- **Rich development environment** including a device emulator, tools for debugging, memory and performance profiling, and a plugin for the Eclipse IDE

Android Architecture

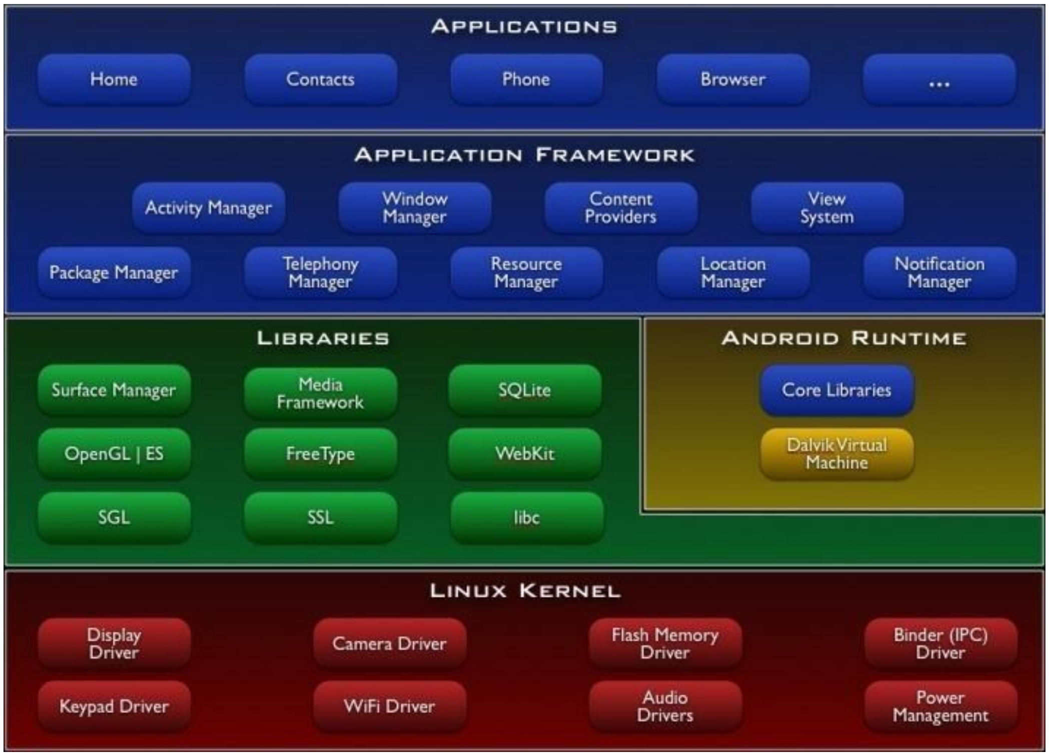


Fig: 3.1.1.1 Android Architecture

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-derived implementation of the standard C system library (libc), tuned for embedded Linux-based devices
- **Media Libraries** - based on PacketVideo's OpenCORE; the libraries support playback and recording of many popular audio and video formats, as well as static image files, including MPEG4, H.264, MP3, AAC, AMR, JPG, and PNG

- **Surface Manager** - manages access to the display subsystem and seamlessly composites 2D and 3D graphic layers from multiple applications
- **LibWebCore** - a modern web browser engine which powers both the Android browser and an embeddable web view
- **SGL** - the underlying 2D graphics engine
- **3D libraries** - an implementation based on OpenGL ES 1.0 APIs; the libraries use either hardware 3D acceleration (where available) or the included, highly optimized 3D software rasterizer
- **FreeType** - bitmap and vector font rendering
- **SQLite** - a powerful and lightweight relational database engine available to all applications

Android Runtime

Android includes a set of core libraries that provides most of the functionality available in the core libraries of the Java programming language.

Every Android application runs in its own process, with its own instance of the Dalvik virtual machine. Dalvik has been written so that a device can run multiple VMs efficiently. The Dalvik VM executes files in the Dalvik Executable (.dex) format which is optimized for minimal memory footprint. The VM is register-based, and runs classes compiled by a Java language compiler that have been transformed into the .dex format by the included "dx" tool.

The Dalvik VM relies on the Linux kernel for underlying functionality such as threading and low-level memory management.

Linux Kernel

Android relies on Linux version 2.6 for core system services such as security, memory management, process management, network stack, and driver model. The kernel also acts as an abstraction layer between the hardware and the rest of the software stack.

The Linux kernel is an operating system kernel used by the Linux family of Unix-like operating systems. It is one of the most prominent examples of free and open source software.

The Linux kernel is released under the GNU General Public License version 2 (GPLv2), (plus some firmware images with various licenses), and is developed by contributors worldwide. Day-to-day development takes place on the Linux kernel mailing list.

The Linux kernel was initially conceived and created by Finnish computer science student Linus Torvalds in 1991. Linux rapidly accumulated developers and users who adapted code from other free software projects for use with the new operating system. The Linux kernel has received contributions from thousands of programmers. Many Linux distributions have been released based upon the Linux kernel.

The Linux kernel has extensive support for and runs on many virtual machine architectures both as the host operating system and as a guest operating system. The virtual machines usually emulate Intel x86 family of processors, though in a few cases PowerPC or ARM processors are also emulated.

At Google, the team led by Rubin developed a mobile device platform powered by the Linux kernel. Google marketed the platform to handset makers and carriers on the premise of providing a flexible, upgradable system. Google had lined up a series of hardware component and software partners and signaled to carriers that it was open to various degrees of cooperation on their part.

Speculation about Google's intention to enter the mobile communications market continued to build through December 2006. Reports from the BBC and The Wall Street Journal noted that Google wanted its search and applications on mobile phones and it was working hard to deliver that. Print and online media outlets soon reported rumors that Google was developing a Google-branded handset.

Some speculated that as Google was defining technical specifications, it was showing prototypes to cell phone manufacturers and network operators.

Hardware running Android

The main supported platform for Android is the ARM architecture.

The Android OS can be used as an operating system for cellphones, netbooks and tablets, including the Dell Streak, Samsung Galaxy Tab, TV and other devices.[68][69] The first commercially available phone to run the Android operating system was the HTC Dream, released on 22 October 2008. In early 2010 Google collaborated with HTC to launch its flagship Android device, the Nexus One. This was followed later in 2010 with the Samsung-made Nexus S.

The early feedback on developing applications for the Android platform was mixed. Issues cited include bugs, lack of documentation, inadequate QA infrastructure, and no public issue-tracking system. (Google announced an issue tracker on 18 January 2008.) In December 2007, MergeLab mobile startup founder Adam MacBeth stated, "Functionality is not there, is poorly documented or just doesn't work... It's clearly not ready for prime time." Despite this, Android-targeted applications began to appear the week after the platform was announced. The first publicly available application was the Snake game The Android Dev Phone is a SIM-unlocked and hardware-unlocked device that is designed for advanced developers. While developers can use regular consumer devices purchased at retail to test and use their applications, some developers may choose not to use a retail device, preferring an unlocked or no-contract device.

The Android software development kit (SDK) includes a comprehensive set of development tools. These include a debugger, libraries, a handset emulator (based on QEMU), documentation, sample code, and tutorials. The SDK is downloadable on the android developer website. Currently supported development platforms include computers running Linux (any modern desktop Linux distribution), Mac OS X 10.4.9 or later, Windows XP or later. The officially supported integrated development environment (IDE) is Eclipse (currently 3.5 or 3.6) using the Android Development Tools (ADT) Plugin, though developers may use any text editor to edit Java and XML files then use command line tools (Java Development Kit and Apache Ant are required) to create, build and debug Android applications as well as control attached Android devices (e.g., triggering a reboot, installing software package(s) remotely).[81]

Android applications are packaged in .apk format and stored under `/data/app` folder on the Android OS (the folder is accessible to root user only for security reasons). APK package contains .dex files (compiled byte code files called Dalvik executables), resource files, etc.

Android Operation System

Android is an operating system based on Linux with a Java programming interface. It provides tools, e.g. a compiler, debugger and a device emulator as well as its own Java Virtual machine (Dalvik Virtual Machine - DVM). Android is created by the Open Handset Alliance which is lead by Google.

Android uses a special virtual machine, e.g. the Dalvik Virtual Machine. Dalvik uses special bytecode. Therefore you cannot run standard Java bytecode on Android. Android provides a tool "dx" which allows to convert Java Class files into "dex" (Dalvik Executable) files. Android applications are packed into an .apk (Android Package) file by the program "aapt" (Android Asset Packaging Tool) To simplify development Google provides the Android Development Tools (ADT) for Eclipse . The ADT performs automatically the conversion from class to dex files and creates the apk during deployment.

Android supports 2-D and 3-D graphics using the OpenGL libraries and supports data storage in a SQLite database.

Every Android applications runs in its own process and under its own userid which is generated automatically by the Android system during deployment. Therefore the application is isolated from other running applications and a misbehaving application cannot easily harm other Android applications.

Important Android components

An Android application consists out of the following parts:

- Activity - Represents the presentation layer of an Android application, e.g. a screen which the user sees. An Android application can have several activities and it can be switched between them during runtime of the application.
- Views - The User interface of an Activities is build with widgets classes which inherent from "android.view.View". The layout of the views is managed by "android.view.ViewGroups".
- Services - perform background tasks without providing an UI. They can notify the user via the notification framework in Android.
- Content Provider - provides data to applications, via a content provider your application can share data with other applications. Android contains a SQLite DB which can serve as data provider
- Intents are asynchronous messages which allow the application to request functionality from other services or activities. An application can call directly a service or activity (explicit intent) or asked the Android system for registered services and applications for an intent (implicit intents). For example the application could ask via an intent for a contact application. Application register themself to an intent via an IntentFilter. Intents are a powerful concept as they allow to create loosely coupled applications.
- Broadcast Receiver - receives system messages and implicit intents, can be used to react to changed conditions in the system. An application can register as a broadcast receiver for certain events and can be started if such an event occurs.
- A Java Virtual Machine (JVM) enables a set of computer software programs and data structures to use a virtual machine model for the execution of other computer programs and scripts. The model used by a JVM accepts a form of computer intermediate language commonly referred to as Java bytecode. This language conceptually represents the instruction set of a stack-oriented, capability architecture. Sun Microsystems states there are over 4.5 billion JVM-enabled devices
- A JVM can also execute bytecode compiled from programming languages other than Java. For example, Ada source code can be compiled to execute on a JVM.

JVMs can also be released by other companies besides Oracle (the developer of Java) — JVMs using the "Java" trademark may be developed by other companies as long as they adhere to the JVM specification published by Oracle and to related contractual obligations.

- Java was conceived with the concept of WORA: "write once, run anywhere". This is done using the Java Virtual Machine. The JVM is the environment in which java programs execute. It is software that is implemented on non-virtual hardware and on standard operating systems.
- JVM is a crucial component of the Java platform, and because JVMs are available for many hardware and software platforms, Java can be both middleware and a platform in its own right, hence the trademark write once, run anywhere. The use of the same bytecode for all platforms allows Java to be described as "compile once, run anywhere", as opposed to "write once, compile anywhere", which describes cross-platform compiled languages. A JVM also enables such features as automated exception handling, which provides "root-cause" debugging information for every software error (exception), independent of the source code.
- A JVM is distributed along with a set of standard class libraries that implement the Java application programming interface (API). Appropriate APIs bundled together form the Java Runtime Environment (JRE).
- Java's execution environment is termed the Java Runtime Environment, or JRE.
- Programs intended to run on a JVM must be compiled into a standardized portable binary format, which typically comes in the form of .class files. A program may consist of many classes in different files. For easier distribution of large programs, multiple class files may be packaged together in a .jar file (short for Java archive).
- The Java application launcher, java, offers a standard way of executing Java code. Compare javaw.[2]
- The JVM runtime executes .class or .jar files, emulating the JVM instruction set by interpreting it, or using a just-in-time compiler (JIT) such as Oracle's HotSpot. JIT compiling, not interpreting, is used in most JVMs today to achieve greater speed. There are also ahead-of-time compilers that enable developers to precompile class files into native code for particular platforms.
- Like most virtual machines, the Java Virtual Machine has a stack-based architecture akin to a microcontroller/microprocessor. However, the JVM also has

low-level support for Java-like classes and methods, which amounts to a highly idiosyncratic[clarification needed] memory model and capability-based architecture.

Download the Android SDK

Welcome Developers! If you are new to the Android SDK, please read the steps below, for an overview of how to set up the SDK.

If you're already using the Android SDK, you should update to the latest tools or platform using the *Android SDK and AVD Manager*, rather than downloading a new SDK starter package. See Adding SDK Components.

Here an overview of the steps you must follow to set up the Android SDK:

1. Prepare your development computer and ensure it meets the system requirements.
2. Install the SDK starter package from the table above. (If you're on Windows, download the installer for help with the initial setup.)
3. Install the ADT Plugin for Eclipse (if you'll be developing in Eclipse).
4. Add Android platforms and other components to your SDK.
5. Explore the contents of the Android SDK (optional).

To get started, download the appropriate package from the table above, then read the guide to Installing the SDK.

Installing the SDK

Step 1. Preparing Your Development Computer

Before getting started with the Android SDK, take a moment to confirm that your development computer meets the System Requirements. In particular, you might need to install the JDK, if you don't have it already.

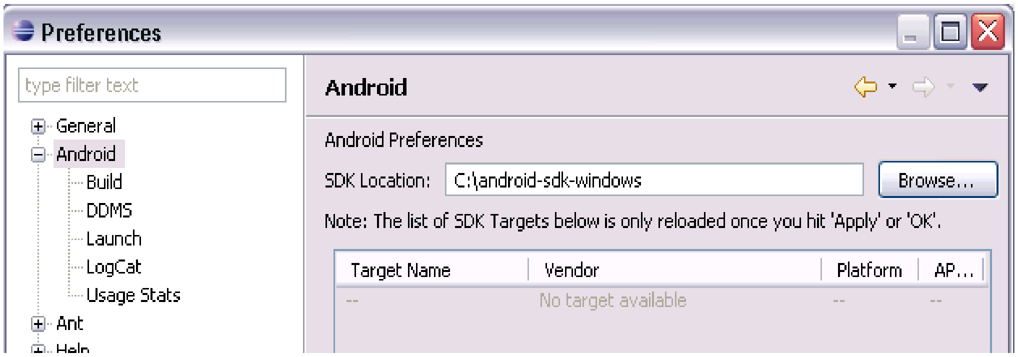
If you will be developing in Eclipse with the Android Development Tools (ADT) Plugin—the recommended path if you are new to Android—make sure that you have a suitable version of Eclipse installed on your computer as described in the System Requirements document. If you need to install Eclipse, you can download it from this location:

The "Eclipse Classic" version is recommended. Otherwise, a Java or RCP version of Eclipse is recommended.

Use the Eclipse update manager to install all available plugins for the Android Development Tools (ADT) from the URL https://dl-ssl.google.com/android/eclipse/ .

Configuration

In Eclipse open the Preferences dialog via Windows -> Preferences. Select Android and maintain the installation path of the Android SDK.



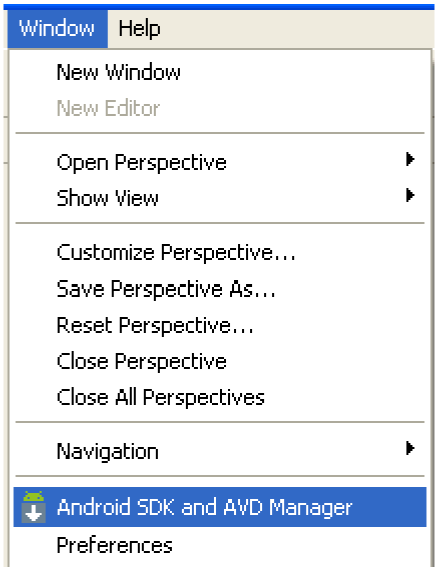
Android Preferences

SDK Location: C:\android-sdk-windows

Note: The list of SDK Targets below is only reloaded once you hit 'Apply' or 'OK'.

Target Name	Vendor	Platform	AP...
--	No target available	--	--

Select Window -> Android SDK and AVD Manager from the menu.



Select available packages and select the latest version of the SDK.

Step 2. Downloading the SDK Starter Package

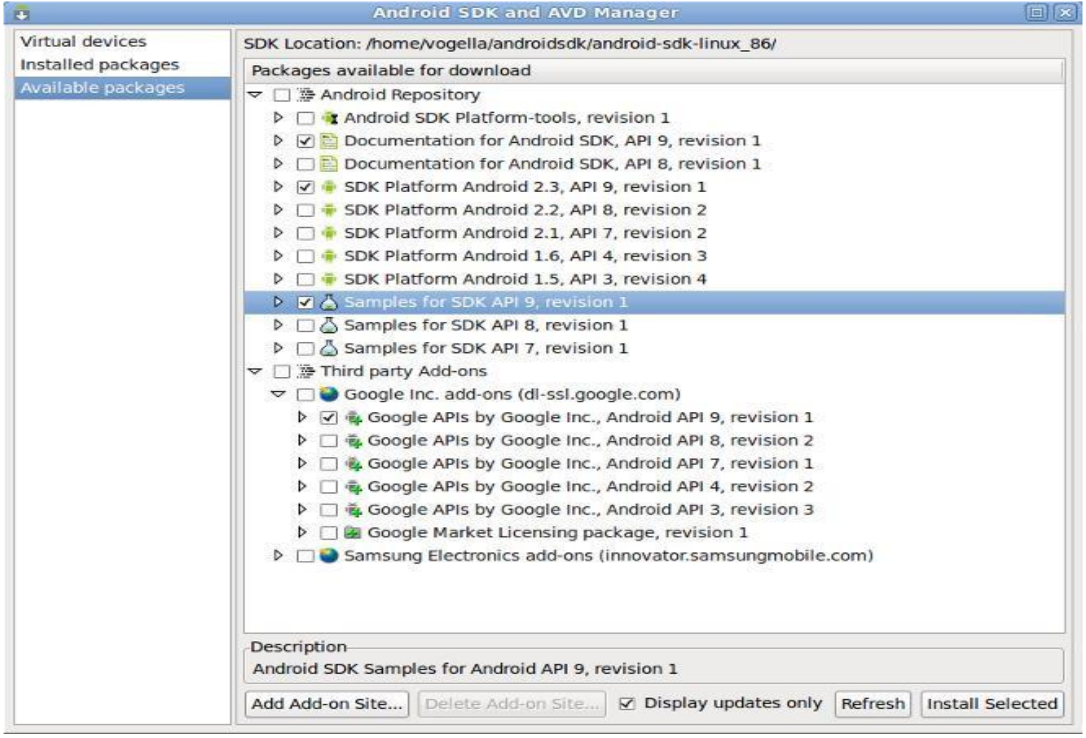
The SDK starter package is not a full development environment—it includes only the core SDK Tools, which you can use to download the rest of the SDK components (such as the latest Android platform).

If you haven't already, get the latest version of the SDK starter package from the SDK download page.

If you downloaded a `.zip` or `.tgz` package (instead of the SDK installer), unpack it to a safe location on your machine. By default, the SDK files are unpacked into a directory named `android-sdk-<machine-platform>`.

If you downloaded the Windows installer (`.exe` file), run it now and it will check whether the proper Java SE Development Kit (JDK) is installed (installing it, if necessary), then install the SDK Tools into a default location (which you can modify).

Make a note of the name and location of the SDK directory on your system—you will need to refer to the SDK directory later, when setting up the ADT plugin and when using the SDK tools from the command line.



SDK Location: /home/vogella/androidsdk/android-sdk-linux_86/

Packages available for download

- □ Android Repository
  - □ Android SDK Platform-tools, revision 1
  - ☑ Documentation for Android SDK, API 9, revision 1
  - □ Documentation for Android SDK, API 8, revision 1
  - ☑ SDK Platform Android 2.3, API 9, revision 1
  - □ SDK Platform Android 2.2, API 8, revision 2
  - □ SDK Platform Android 2.1, API 7, revision 2
  - □ SDK Platform Android 1.6, API 4, revision 3
  - □ SDK Platform Android 1.5, API 3, revision 4
  - ☑ Samples for SDK API 9, revision 1
  - □ Samples for SDK API 8, revision 1
  - □ Samples for SDK API 7, revision 1
- □ Third party Add-ons
  - □ Google Inc. add-ons (dl-ssl.google.com)
    - ☑ Google APIs by Google Inc., Android API 9, revision 1
    - □ Google APIs by Google Inc., Android API 8, revision 2
    - □ Google APIs by Google Inc., Android API 7, revision 1
    - □ Google APIs by Google Inc., Android API 4, revision 2
    - □ Google APIs by Google Inc., Android API 3, revision 3
    - □ Google Market Licensing package, revision 1
  - □ Samsung Electronics add-ons (innovator.samsungmobile.com)

Description
Android SDK Samples for Android API 9, revision 1

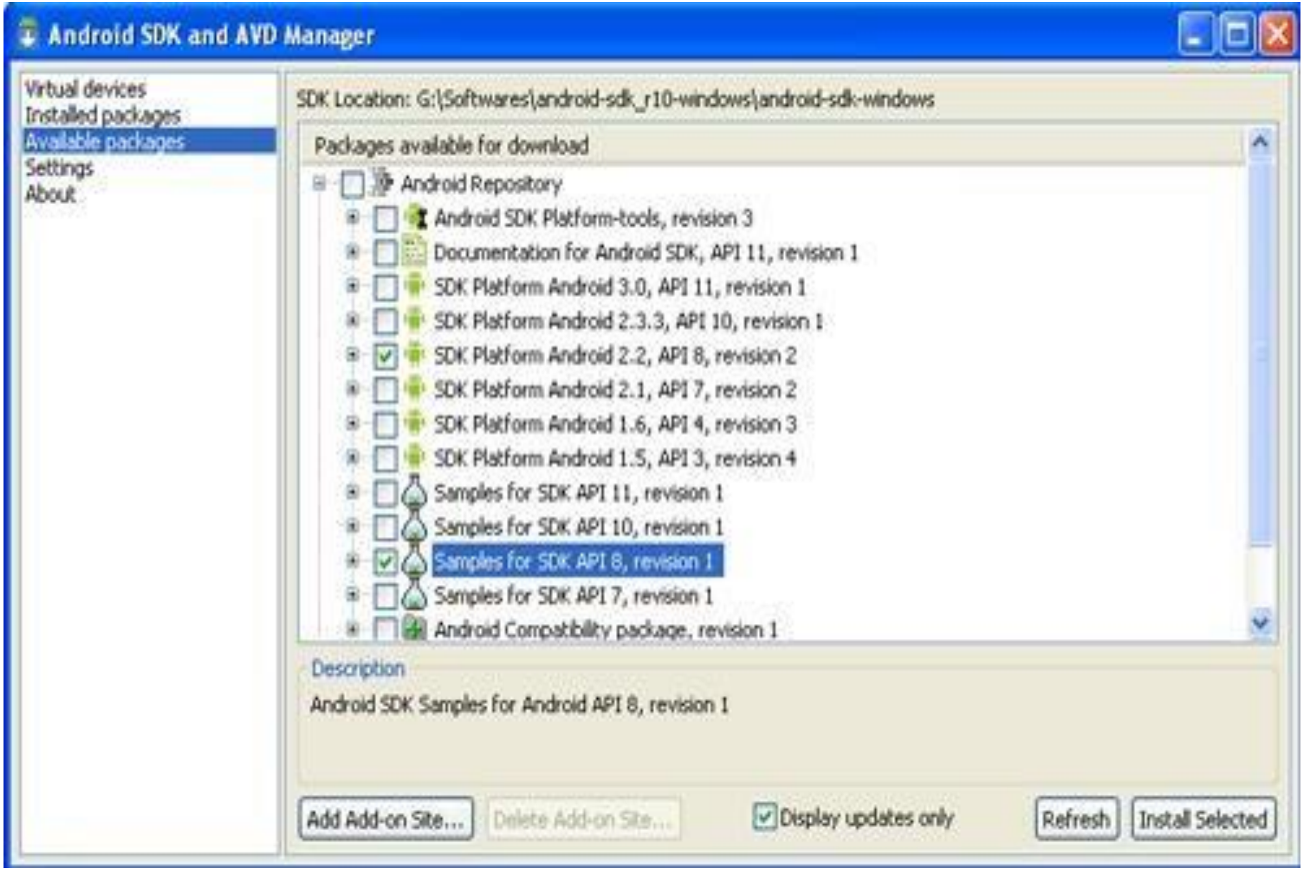
☑ Display updates only

Step 3. Installing the ADT Plugin for Eclipse

Android offers a custom plugin for the Eclipse IDE, called Android Development Tools (ADT), that is designed to give you a powerful, integrated environment in which to build Android applications. It extends the capabilites of Eclipse to let you quickly set up new Android projects, create an application UI, debug your applications using the Android SDK tools, and even export signed (or unsigned) APKs in order to distribute your application. In general, developing in Eclipse with ADT is a highly recommended approach and is the fastest way to get started with Android.

If you'd like to use ADT for developing Android applications, install it now. Read Installing the ADT Plugin for step-by-step installation instructions, then return here to continue the last step in setting up your Android SDK.

If you prefer to work in a different IDE, you do not need to install Eclipse or ADT. Instead, you can directly use the SDK tools to build and debug your application. The Introduction to Android application development outlines the major steps that you need to complete when developing in Eclipse or other IDEs.



SDK Location: G:\Softwares\android-sdk_r10-windows\android-sdk-windows

Packages available for download

- □ Android Repository
  - □ Android SDK Platform-tools, revision 3
  - □ Documentation for Android SDK, API 11, revision 1
  - □ SDK Platform Android 3.0, API 11, revision 1
  - □ SDK Platform Android 2.3.3, API 10, revision 1
  - ☑ SDK Platform Android 2.2, API 8, revision 2
  - □ SDK Platform Android 2.1, API 7, revision 2
  - □ SDK Platform Android 1.6, API 4, revision 3
  - □ SDK Platform Android 1.5, API 3, revision 4
  - □ Samples for SDK API 11, revision 1
  - □ Samples for SDK API 10, revision 1
  - ☑ Samples for SDK API 8, revision 1
  - □ Samples for SDK API 7, revision 1
  - □ Android Compatibility package, revision 1

Description

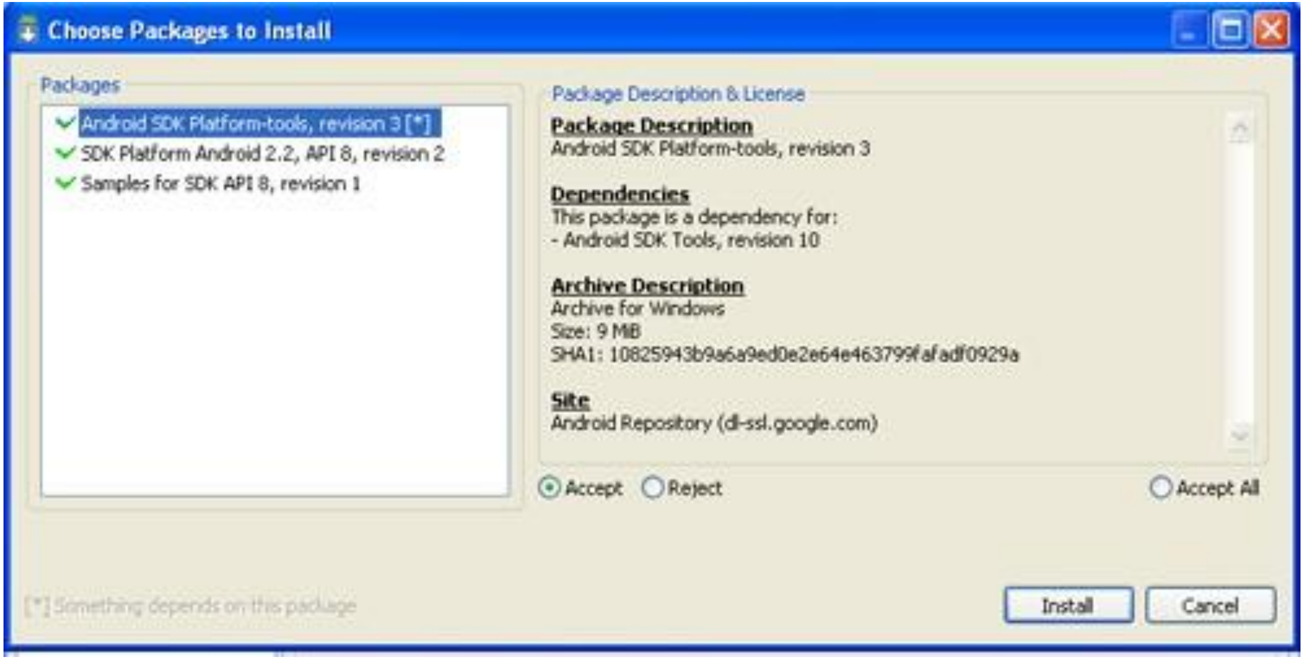
Android SDK Samples for Android API 8, revision 1

☑ Display updates only

Step 4. Adding Platforms and Other Components

The last step in setting up your SDK is using the Android SDK and AVD Manager (a tool included in the SDK starter package) to download essential SDK components into your development environment.

The SDK uses a modular structure that separates the major parts of the SDK—Android platform versions, add-ons, tools, samples, and documentation—into a set of separately installable components. The SDK starter package, which you've already downloaded, includes only a single component: the latest version of the SDK Tools. To develop an Android application, you also need to download at least one Android platform and the associated platform tools. You can add other components and platforms as well, which is highly recommended.


Choose Packages to Install
Packages
Package Description & License
● Accept ○ Reject ○ Accept All
[*] Something depends on this package
Install
Cancel

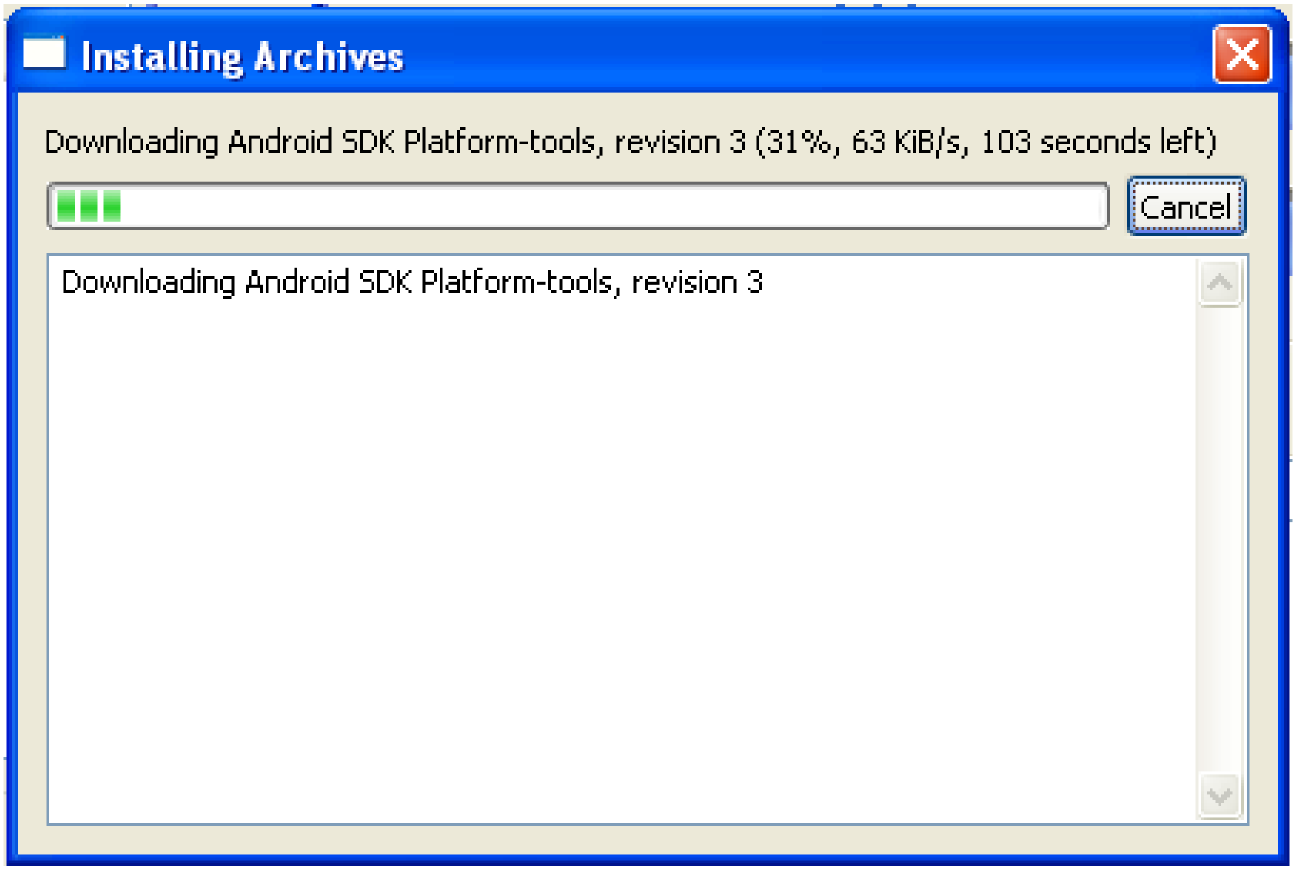
✓ Android SDK Platform-tools, revision 3 [*]
✓ SDK Platform Android 2.2, API 8, revision 2
✓ Samples for SDK API 8, revision 1

Package Description
Android SDK Platform-tools, revision 3

Dependencies
This package is a dependency for:
- Android SDK Tools, revision 10

Archive Description
Archive for Windows
Size: 9 MiB
SHA1: 10825943b9a6a9ed0e2e64e463799fafadf0929a

Site
Android Repository (dl-ssl.google.com)



Downloading Android SDK Platform-tools, revision 3 (31%, 63 KiB/s, 103 seconds left)

Downloading Android SDK Platform-tools, revision 3

If you used the Windows installer, when you complete the installation wizard, it will launch the Android SDK and AVD Manager with a default set of platforms and other components selected for you to install. Simply click **Install** to accept the recommended set of components and install them. You can then skip to Step 5, but we recommend you first read the section about the Available Components to better understand the components available from the Android SDK and AVD Manager.

You can launch the Android SDK and AVD Manager in one of the following ways:

- From within Eclipse, select **Window > Android SDK and AVD Manager**.
- On Windows, double-click the `SDK Manager.exe` file at the root of the Android SDK directory.
- On Mac or Linux, open a terminal and navigate to the `tools/` directory in the Android SDK, then execute:

To download components, use the graphical UI of the Android SDK and AVD Manager to browse the SDK repository and select new or updated components (see figure 1). The Android SDK and AVD Manager installs the selected components in your SDK environment. For information about which components you should download, see Recommended Components.

The *Android Repository* offers these types of components:

- **SDK Tools** — Contains tools for debugging and testing your application and other utility tools. These tools are installed with the Android SDK starter package and receive periodic updates. You can access these tools in the `<sdk>/tools/` directory of your SDK. To learn more about them, see SDK Tools in the developer guide.
- **SDK Platform-tools** — Contains platform-dependent tools for developing and debugging your application. These tools support the latest features of the Android platform and are typically updated only when a new platform becomes available. You can access these tools in the `<sdk>/platform-tools/` directory. To learn more about them, see Platform Tools in the developer guide.

- **Android platforms** — An SDK platform is available for every production Android platform deployable to Android-powered devices. Each SDK platform component includes a fully compliant Android library, system image, sample code, and emulator skins. To learn more about a specific platform, see the list of platforms that appears under the section "Downloadable SDK Components" on the left part of this page.
- **USB Driver for Windows** (Windows only) — Contains driver files that you can install on your Windows computer, so that you can run and debug your applications on an actual device. You *do not* need the USB driver unless you plan to debug your application on an actual Android-powered device. If you develop on Mac OS X or Linux, you do not need a special driver to debug your application on an Android-powered device. See Using Hardware Devices for more information about developing on a real device.
- **Samples** — Contains the sample code and apps available for each Android development platform. If you are just getting started with Android development, make sure to download the samples to your SDK.
- **Documentation** — Contains a local copy of the latest multiversion documentation for the Android framework API.

The *Third party Add-ons* provide components that allow you to create a development environment using a specific Android external library (such as the Google Maps library) or a customized (but fully compliant) Android system image. You can add additional Add-on repositories by clicking **Add Add-on Site**.

ECLIPSE:

Eclipse is an open source community whose projects are focused on building an extensible development platform, runtimes and application frameworks for building, deploying and managing software across the entire software lifecycle. Many people know us, and hopefully love us, as a Java IDE but Eclipse is much more than a Java IDE.

The Eclipse open source community has over 60 open source projects. These projects can be conceptually organized into seven different "pillars" or categories:

1. Enterprise Development
2. Embedded and Device Development
3. Rich Client Platform
4. Rich Internet Applications
5. Application Frameworks
6. Application Lifecycle Management (ALM)
7. Service Oriented Architecture (SOA)

The Eclipse community is also supported by a large and vibrant ecosystem of major IT solution providers, innovative start-ups, universities and research institutions and individuals that extend, support and complement the Eclipse Platform.

The exciting thing about Eclipse is many people are using Eclipse in ways that we have never imagined. The common thread is that they are building innovative, industrial strength software and want to use great tools, frameworks and runtimes to make their job easier.

Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system. It is written mostly in Java and can be used to develop applications in Java and, by means of various plug-ins, other programming languages including Ada, C, C++, COBOL, Perl, PHP, Python, Ruby (including Ruby on Rails framework), Scala, Clojure, and Scheme. The IDE is often called Eclipse ADT for Ada, Eclipse CDT for C/C++, Eclipse JDT for Java, and Eclipse PDT for PHP.

ARCHITECTURE:

Eclipse employs plug-ins in order to provide all of its functionality on top of (and including) the runtime system, in contrast to some other applications where functionality is typically hard coded. The runtime system of Eclipse is based on Equinox, an OSGi standard compliant implementation.

This plug-in mechanism is a lightweight software componentry framework. In addition to allowing Eclipse to be extended using other programming languages such as C and Python, the plug-in framework allows Eclipse to work with typesetting languages like LaTeX,[2] networking applications such as telnet, and database management systems. The plug-in architecture supports writing any desired extension to the environment, such as for configuration management. Java and CVS support is provided in the Eclipse SDK, with Subversion support provided by third-party plug-ins.

With the exception of a small run-time kernel, everything in Eclipse is a plug-in. This means that every plug-in developed integrates with Eclipse in exactly the same way as other plug-ins; in this respect, all features are "created equal". Eclipse provides plug-ins for a wide variety of features, some of which are through third parties using both free and commercial models. Examples of plug-ins include a UML plug-in for Sequence and other UML diagrams, a plug-in for DB Explorer, and many others.

The Eclipse SDK includes the Eclipse Java Development Tools (JDT), offering an IDE with a built-in incremental Java compiler and a full model of the Java source files. This allows for advanced refactoring techniques and code analysis. The IDE also makes use of a workspace, in this case a set of metadata over a flat filespace allowing external file modifications as long as the corresponding workspace "resource" is refreshed afterwards.

Eclipse implements widgets through a widget toolkit for Java called SWT, unlike most Java applications, which use the Java standard Abstract Window Toolkit (AWT) or Swing. Eclipse's user interface also uses an intermediate GUI layer called JFace, which simplifies the construction of applications based on SWT.

Rich Client Platform

- Equinox OSGi – a standard bundling framework
- Core platform – boot Eclipse, run plug-ins
- Standard Widget Toolkit (SWT) – a portable widget toolkit
- JFace – viewer classes to bring model view controller programming to SWT, file buffers, text handling, text editors
- Eclipse Workbench – views, editors, perspectives, wizards

History

Eclipse began as an IBM Canada project. It was developed by Object Technology International (OTI) as a Java-based replacement for the Smalltalk based VisualAge family of IDE products,[4] which itself had been developed by OTI.[1] In November 2001, a consortium was formed to further the development of Eclipse as open source. In January 2004, the Eclipse Foundation was created.[5]

Eclipse 3.0 (released on 21 June 2004) selected the OSGi Service Platform specifications as the runtime architecture.[6]

Eclipse was originally released under the Common Public License, but was later relicensed under the Eclipse Public License. The Free Software Foundation has said that both licenses are free software licenses, but are incompatible with the GNU General Public License (GPL). Mike Milinkovich, of the Eclipse Foundation commented that moving to the GPL would be considered when version 3 of the GPL was released.

According to Lee Nackman, Chief Technology Officer of IBM's Rational division at that time and later head of Rational software development and support, the name "Eclipse" was chosen to target Microsoft's Visual Studio product, and not Sun Microsystems.[9] Ironically, Nackman is now himself a Microsoft employee.

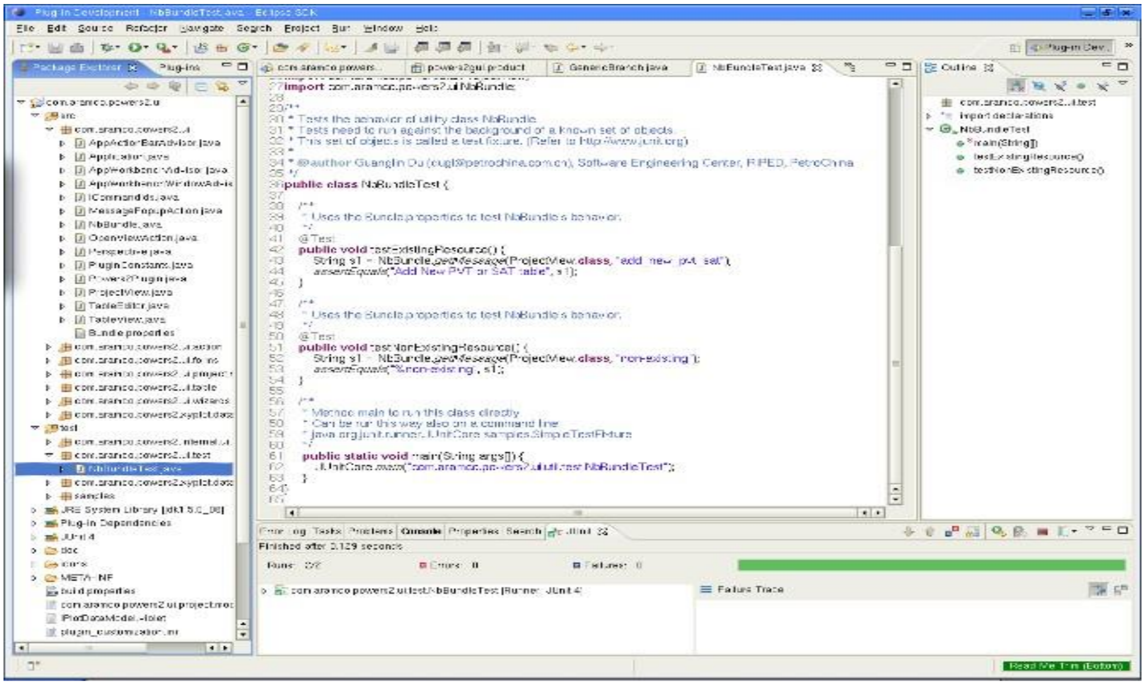
Eclipse (SDK)

Eclipse Software Development Kit (SDK) is a Java based open-source integrated development environment (IDE) which combines a number of different Eclipse projects including Platform, Java Development Tools (JDT) and the Plug-in Development Environment (PDE).

Eclipse can be used to create a large array of software applications using languages ranging from PHP, C++ programs, to Java. It is one of the most popular development tools in both the open-source and commercial worlds.

It provides Java editing with validation, incremental compilation, cross-referencing, code assist; an XML Editor; Mylyn; and much more.

Eclipse is released under the Eclipse Foundation, a commercially friendly license that allows organizations to include Eclipse software in their commercial products, while at the same time asking those who create derivative works of EPL code to contribute back to the community.



```
import com.aramco.powers2.ui.NbBundle;

/**
 * Tests the behavior of utility class NbBundle.
 * Tests need to run against the background of a known set of objects.
 * This set of objects is called a test fixture. (Refer to http://www.junit.org)
 *
 * @author Guanglin Du (cugl@petrochina.com.cn), Software Engineering Center, RIPED, PetroChina
 */
public class NbBundleTest {

    /**
     * Uses the Bundle.properties to test NbBundle's behavior.
     */
    @Test
    public void testExistingResource() {
        String s1 = NbBundle.getMessage(ProjectView.class, "add_new_pvt_sat");
        assertEquals("Add New PVT or SAT table", s1);
    }

    /**
     * Uses the Bundle.properties to test NbBundle's behavior.
     */
    @Test
    public void testNonExistingResource() {
        String s1 = NbBundle.getMessage(ProjectView.class, "non-existing");
        assertEquals("%non-existing", s1);
    }

    /**
     * Method main to run this class directly
     * Can be run this way also on a command line
     * java org.junit.runner.JUnitCore samples.SimpleTestFixture
     */
    public static void main(String args[]) {
        JUnitCore.main("com.aramco.powers2.ui.util.test.NbBundleTest");
    }
```

Finished after 0.129 seconds

Runs: 2/2 Errors: 0 Failures: 0

com.aramco.powers2.ui.test.NbBundleTest [Runner: JUnit 4]

Plug-in Development - NbBundleTest.java - Eclipse SDK
File Edit Source Refactor Navigate Search Project Run Window Help
Package Explorer
Plug-ins
NbBundleTest.java
Outline
Plug-in Dev...
Error Log Tasks Problems Console Properties Search JUnit
Failure Trace

Eclipse Platform

The Eclipse Platform provides the core frameworks and services upon which all plug-in extensions are created. It also provides the runtime in which plug-ins are loaded, integrated, and executed. The primary purpose of the Platform is to enable other tool developers to easily build and deliver integrated tools.

Features include:

- Supports the construction of a variety of tools for application development
- Supports an unrestricted set of tool providers, including independent software vendors (ISVs)
- Supports tools to manipulate arbitrary content types (e.g., HTML, Java, C, JSP, EJB, XML, and GIF)
- Facilitates seamless integration of tools within and across different content types and tool providers

- Supports both GUI and non-GUI-based application development environments

Java Development Tools (JDT)

The JDT project provides the tool plug-ins that implement a Java IDE supporting the development of any Java application, including Eclipse plug-ins. It adds a Java project nature and Java perspective to the Eclipse Workbench as well as a number of views, editors, wizards, builders, and code merging and refactoring tools. The JDT project allows Eclipse to be a development environment for itself.

Features include:

- Java projects with source files arranged in package directories
- Editing with keyword and syntax coloring, outline showing declaration structure
- Code formatter
- Refactoring
- Search
- Compare
- Compile - JCK-compliant Java compiler
- Run Java programs in a separate target Java virtual machine
- Debug programs with JPDA-compliant Java virtual machine

Android Source Code

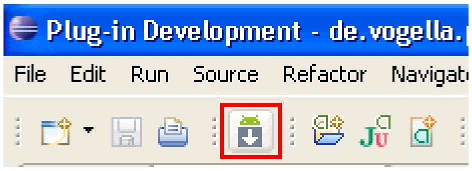
The following step is optional.

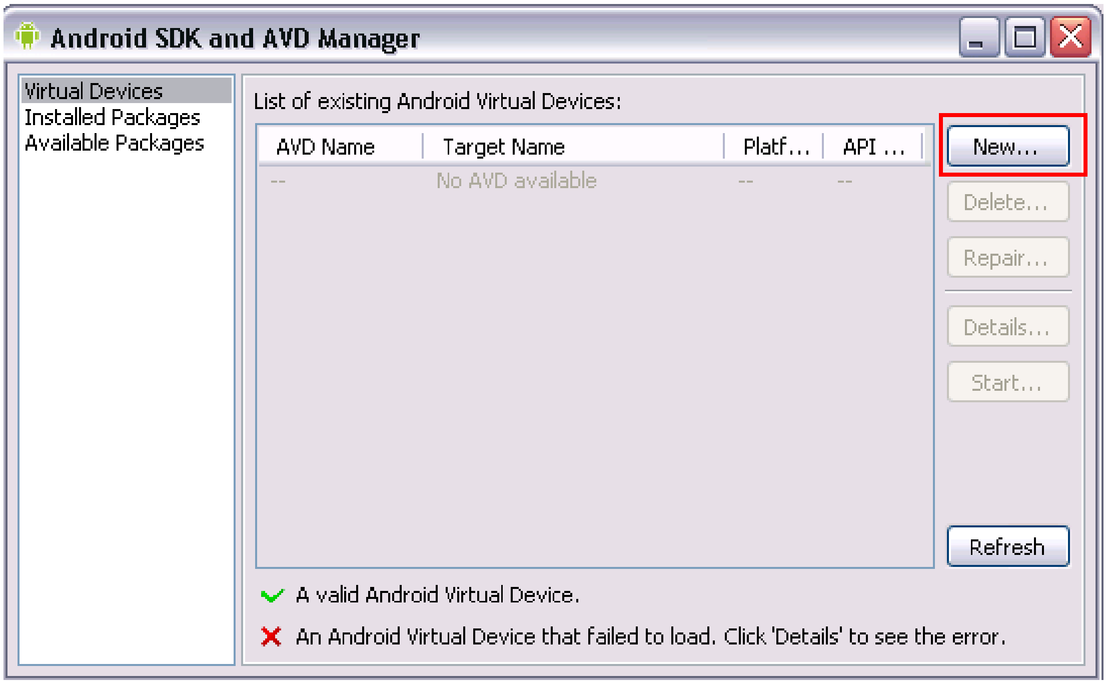
During Android development it is very useful to have the Android source code available as Android uses a lot of defaults.

Haris Peco maintains plugins with provides access to the Android Source code. Use the Eclipse update manager to install two of his plugins.
Create an Android Emulator Device

The Android tools include an emulator. This emulator behaves like a real Android device in most cases and allow you to test your application without having a real device. You can emulate one or several devices with different configurations. Each configuration is defined via an "Android Virtual Device" (AVD).

To define an AVD press the device manager button, press "New" and maintain the following.



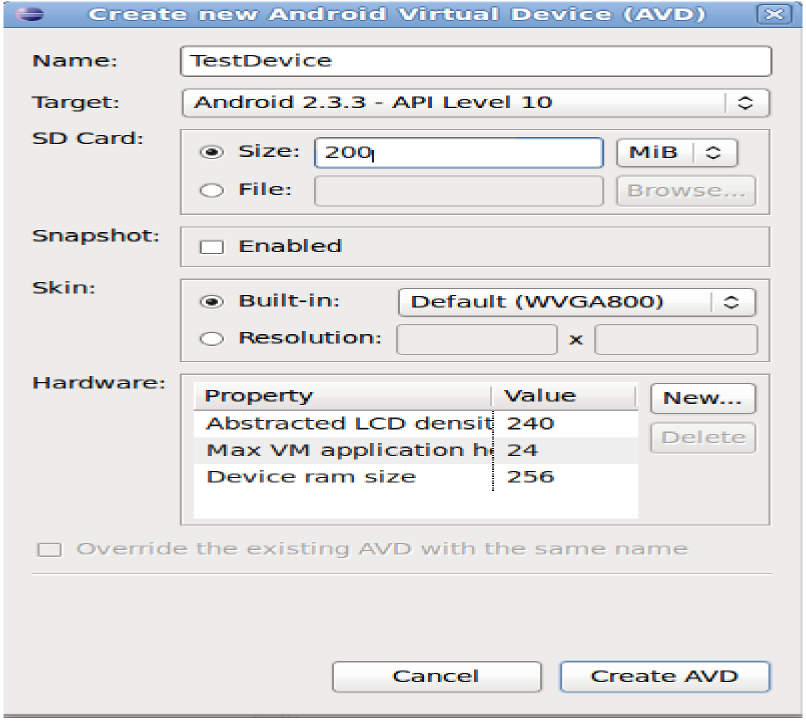


List of existing Android Virtual Devices:

AVD Name	Target Name	Platf...	API ...
--	No AVD available	--	--

✓ A valid Android Virtual Device.

✗ An Android Virtual Device that failed to load. Click 'Details' to see the error.



Create new Android Virtual Device (AVD)

Name: TestDevice

Target: Android 2.3.3 - API Level 10

SD Card: ● Size: 200 MiB ○ File:

Snapshot: □ Enabled

Skin: ● Built-in: Default (WVGA800) ○ Resolution: x

Hardware:

Property	Value
Abstracted LCD densit	240
Max VM application he	24
Device ram size	256

□ Override the existing AVD with the same name

Press "Create AVD".This will create the device and display it under the "Virtual devices". To test if your setup is correct, select your device and press "Start".

4. Error handling

Things are not always working as they should be. Several users report that get the following errors:

1. Project ... is missing required source folder: 'gen'
2. The project could not be built until build path errors are resolved.
3. Unable to open class file R.java.

To solve this error select from the menu Project -> Clean.

If you having problems with your own code you can use the LogCat viewer as described in LogCat Viewer .

Software & Hardware Requirements:

Software Requirements:

Tools	:	Android Studio
Emulator	:	Geny motion
Operating system	:	Windows 10/8/7.
Front End	:	Android Studio, SDK, Java SE 7
Language	:	Java, xml
Database	:	php.

Hardware Requirements:

System	:	Windows 10/8/7/ vista (64 bits) .
Hard Disk	:	400mb hard disk space+atleast 1GB for android sdk.
Ram	:	8GB
Monitor	:	15’’ LED
Input Devices	:	Keyboard, Mouse

Input Design

The input design is the link between the information system and the user. It comprises the developing specification and procedures for data preparation and those steps are necessary to put transaction data in to a usable form for processing can be achieved by inspecting the computer to read data from a written or printed document or it can occur by having people keying the data directly into the system. The design of input focuses on controlling the amount of input required, controlling the errors, avoiding delay, avoiding extra steps and keeping the process simple. The input is designed in such a way so that it provides security and ease of use with retaining the privacy. Input Design considered the following things:

- What data should be given as input?
- How the data should be arranged or coded?
- The dialog to guide the operating personnel in providing input.
- Methods for preparing input validations and steps to follow when error occur.

Objectives

1. Input Design is the process of converting a user-oriented description of the input into a computer-based system. This design is important to avoid errors in the data input process and show the correct direction to the management for getting correct information from the computerized system.

2. It is achieved by creating user-friendly screens for the data entry to handle large volume of data. The goal of designing input is to make data entry easier and to be free from errors. The data entry screen is designed in such a way that all the data manipulates can be performed. It also provides record viewing facilities.

3. When the data is entered it will check for its validity. Data can be entered with the help of screens. Appropriate messages are provided as when needed so that the user will not be in maize of instant. Thus the objective of input design is to create an input layout that is easy to follow

Output Design

A quality output is one, which meets the requirements of the end user and presents the information clearly. In any system results of processing are communicated to the users and to other system through outputs. In output design it is determined how the information is to be displaced for immediate need and also the hard copy output. It is the most important and direct source information to the user. Efficient and intelligent output design improves the system's relationship to help user decision-making.

1. Designing computer output should proceed in an organized, well thought out manner; the right output must be developed while ensuring that each output element is designed so that people will find the system can use easily and effectively. When analysis design computer output, they should Identify the specific output that is needed to meet the requirements.

2. Select methods for presenting information.

3. Create document, report, or other formats that contain information produced by the system.

The output form of an information system should accomplish one or more of the following objectives.

- Convey information about past activities, current status or projections of the
- Future.
- Signal important events, opportunities, problems, or warnings.
- Trigger an action.
- Confirm an action.

4. Feasibility Study

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

Economical Feasibility:

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

Technical Feasibility:

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

Social Feasibility :

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

5. System Design

5.1Architecture of the system

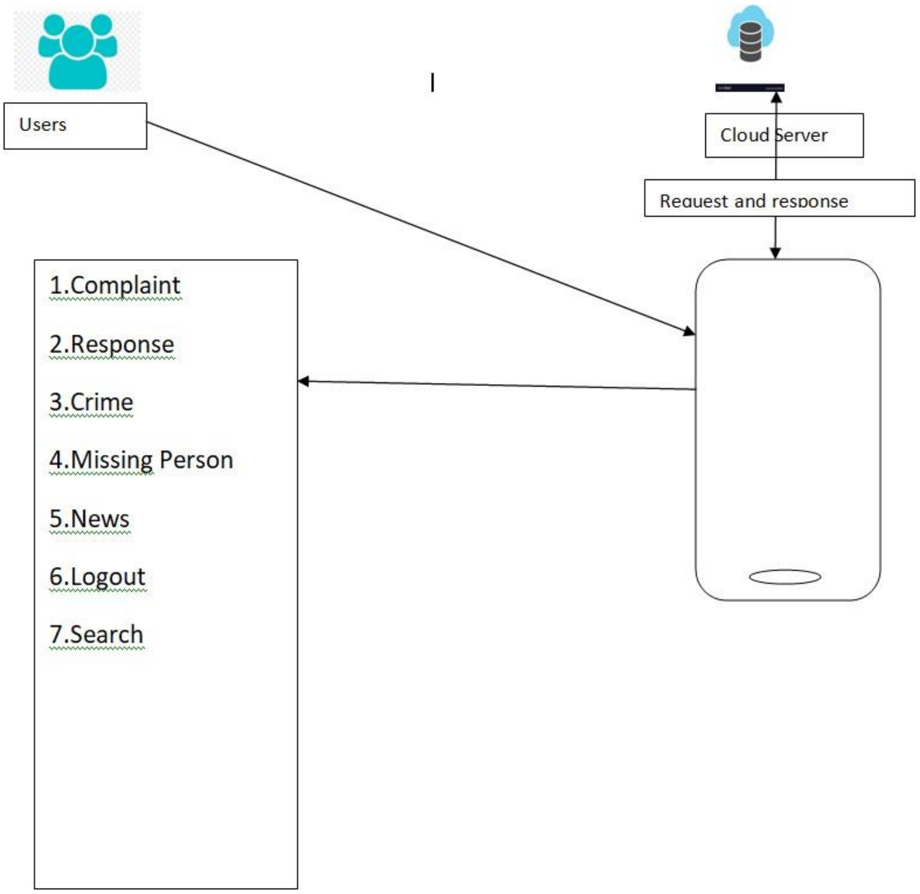


FIG :5.1.1 Arcictecture of the system

Process Design 5.2.1Modules

Description

Two Modules: 1.USER 2.ADMIN

USER:

- User has to register to file complaints, crimes or missing report. User can login to system to file and check the status of his complaints or missing reports. Complaints consist of basic details the system asks and the user has to fill in order to register a complaint and check the status of his complaints.

ADMIN:

- Admin and the user have the same app, no different system. The admin has to check the cases and update the status of that case. The Admin asks the user to enter all the details of the person with a photograph. Admin can add the missing persons and receive complains an crime reports, and give response to user complaints.

UML Diagrams:

UML represents Unified Modeling Language. UML is an institutionalized universally useful showing dialect in the subject of article situated programming designing. The fashionable is overseen, and become made by way of, the Object Management Group.

The goal is for UML to become a regular dialect for making fashions of item arranged PC programming. In its gift frame UML is contained two noteworthy components: a Meta-show and documentation. Later on, a few type of method or system can also likewise be brought to; or related with, UML.

The Unified Modeling Language is a popular dialect for indicating, Visualization, Constructing and archiving the curios of programming framework, and for business demonstrating and different non-programming frameworks.

The UML speaks to an accumulation of first-rate building practices which have verified fruitful in the showing of full-size and complicated frameworks.

The UML is a essential piece of creating gadgets located programming and the product development method. The UML makes use of commonly graphical documentations to specific the plan of programming ventures.

Goals:

The Primary goals inside the plan of the UML are as in step with the subsequent:

1. Provide clients a prepared to-utilize, expressive visual showing Language on the way to create and change massive models.
2. Provide extendibility and specialization units to make bigger the middle ideas.
3. Be free of specific programming dialects and advancement manner.
4. Provide a proper cause for understanding the displaying dialect.
5. Encourage the improvement of OO gadgets exhibit.
6. Support large amount advancement thoughts, for example, joint efforts, systems, examples and components.
7. Integrate widespread procedures.

Use Case Diagram:

A use case diagram in the Unified Modeling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.

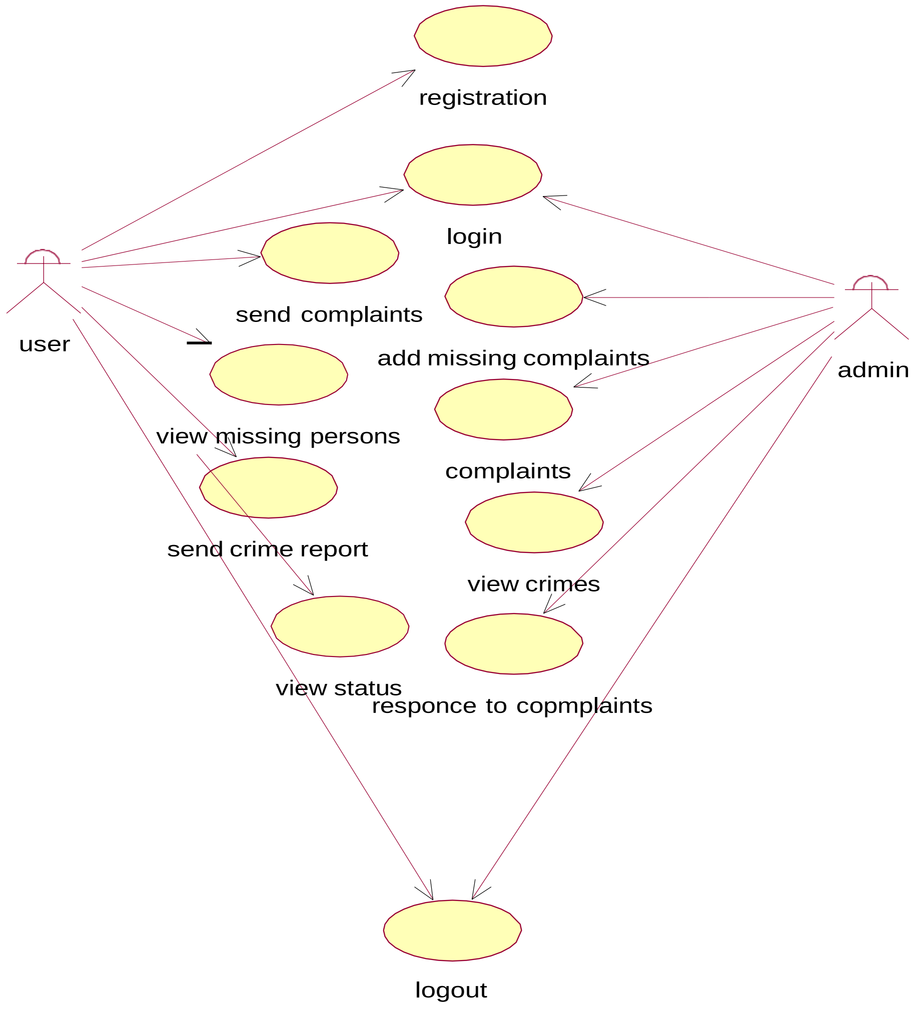


FIG:5.2.2.1.1 Usecase Diagram

Activity Diagram:

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

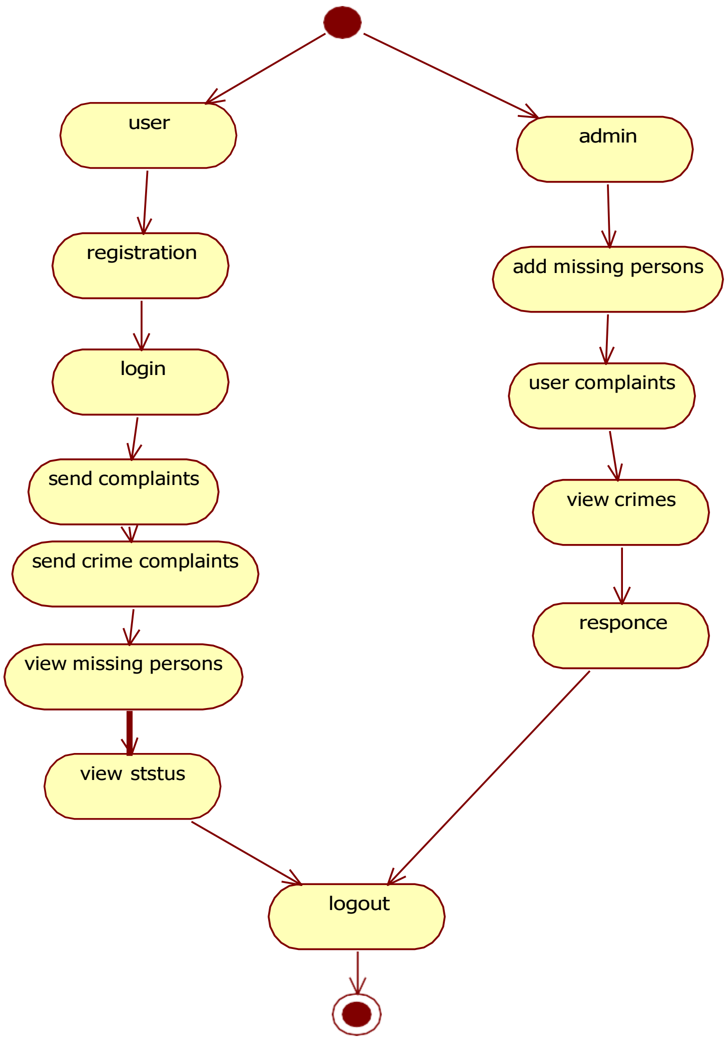


FIG:5.2.2.2.1 Activity diagram

Class Diagram:

In software engineering, a class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among the classes. It explains which class contains information.

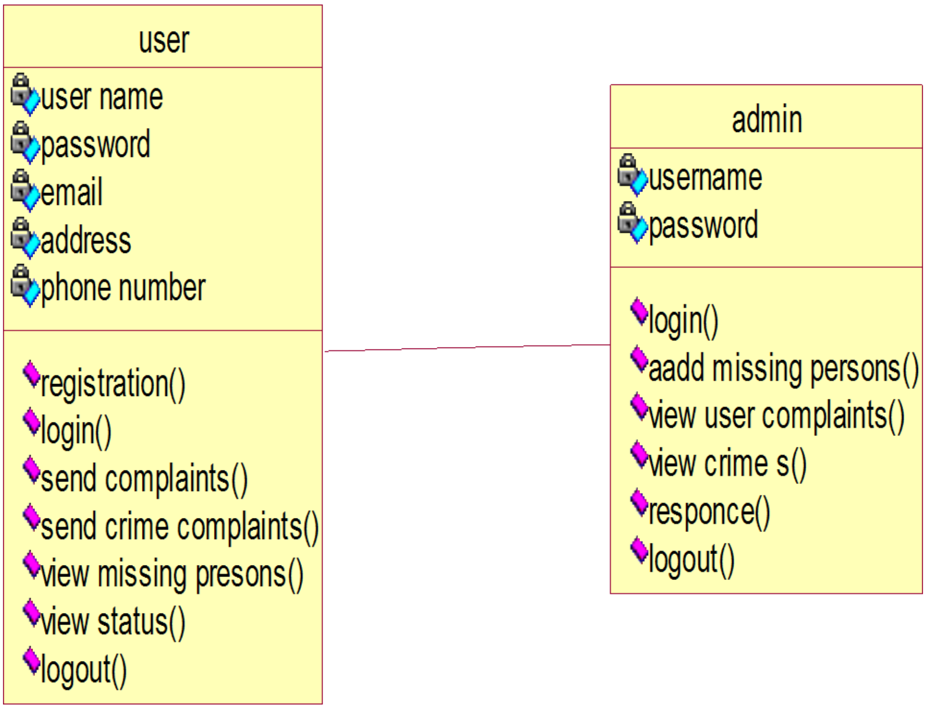


FIG: 5.2.2.3.1 Class diagram

Sequence Diagram:

A sequence diagram in Unified Modeling Language (UML) is a kind of interaction diagram that shows how processes operate with one another and in what order. It is a construct of a Message Sequence Chart. Sequence diagrams are sometimes called event diagrams, event scenarios, and timing diagrams.

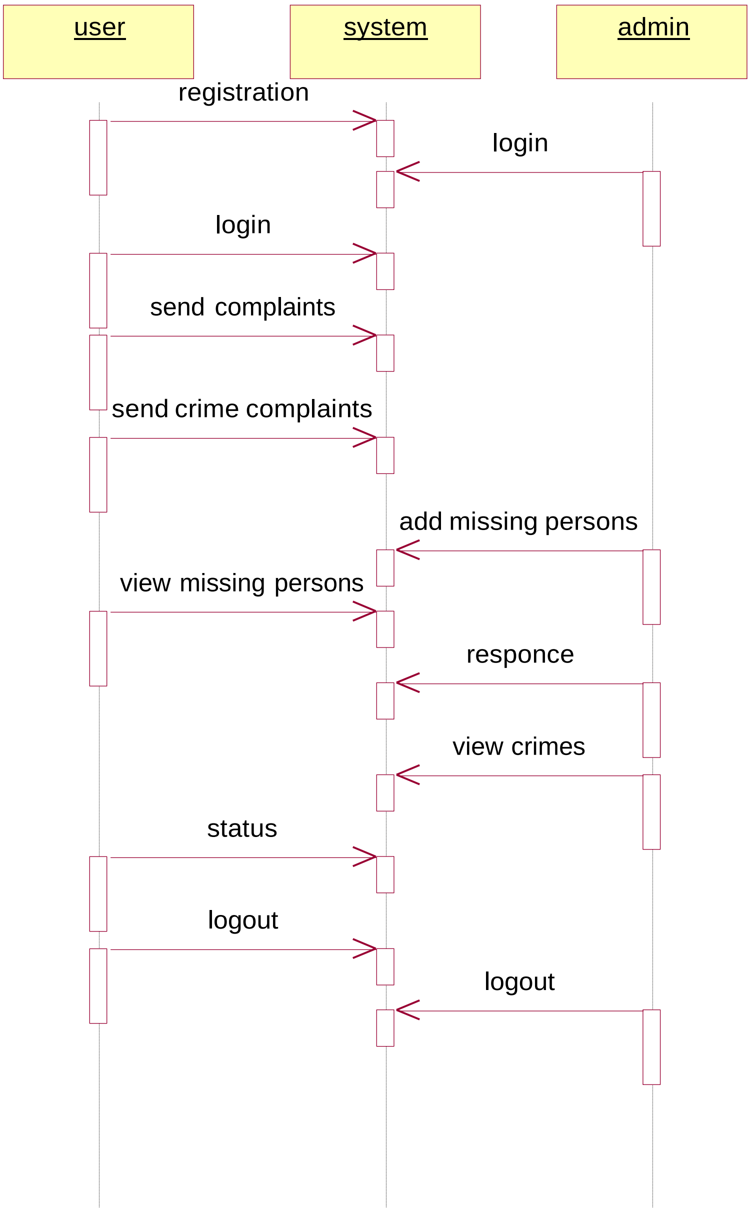


FIG: 5.2.2.4.1 Sequence diagram

Collaboration Diagram:

In collaboration diagram the method call sequence is indicated by some numbering technique as shown below. The number indicates how the methods are called one after another. We have taken the same order management system to describe the collaboration diagram. The method calls are similar to that of a sequence diagram. But the difference is that the sequence diagram does not describe the object organization where as the collaboration diagram shows the object organization.

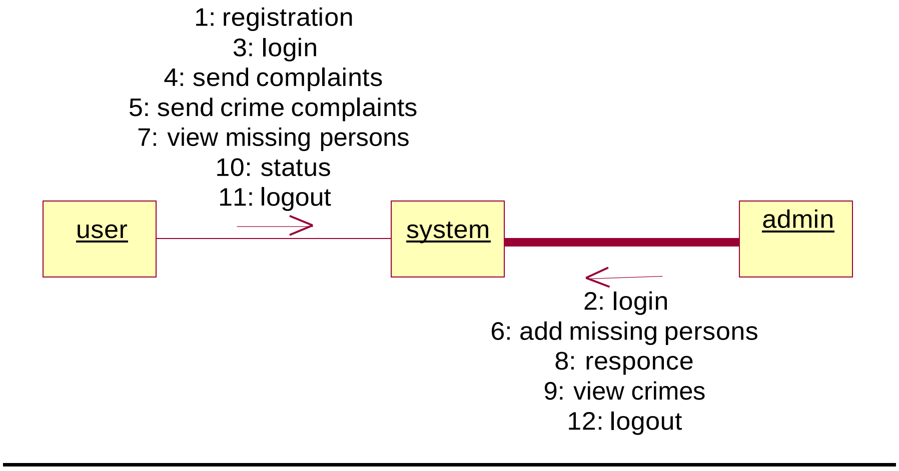


FIG: 5.2.2.5.1 Collaboration diagram

Component Diagram:

Component diagrams are used to describe the physical artifacts of a system. This artifact includes files, executables, libraries etc. So the purpose of this diagram is different, Component diagrams are used during the implementation phase of an application. But it is prepared well in advance to visualize the implementation details. Initially the system is designed using different UML diagrams and then when the artifacts are ready component diagrams are used to get an idea of the implementation.

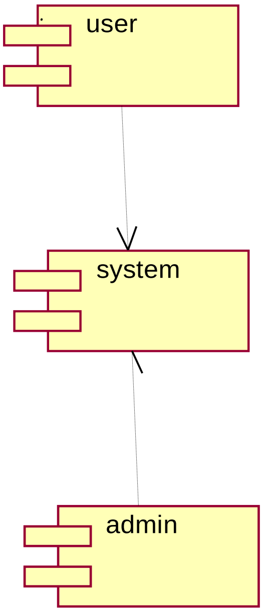


FIG: 5.2.2.6.1 Component diagram

Deployment Diagram:

Deployment diagram represents the deployment view of a system. It is related to the component diagram. Because the components are deployed using the deployment diagrams. A deployment diagram consists of nodes. Nodes are nothing but physical hard ware's used to deploy the application.

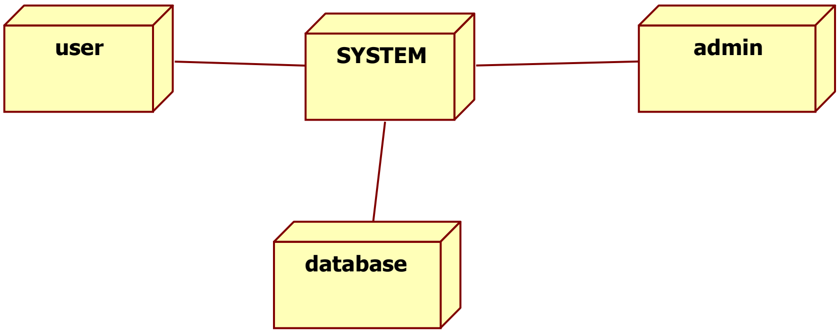


FIG: 5.2.2.7.1 Deployment diagram

User Interface Design

FIG:5.3.1 User Interface Design

6. Coding

Android Java Code

Addcomplaint.java

```
package com.example.srinivasulum.citynews;

import android.app.Activity;
import android.content.Intent;
import android.os.Bundle;
import android.support.v7.app.AppCompatActivity;
import android.support.v7.widget.AppCompatButton;
import android.support.v7.widget.AppCompatEditText;
import android.support.v7.widget.AppCompatTextView;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.TextView;
import android.widget.Toast;

import com.android.volley.AuthFailureError;
import com.android.volley.Request;
import com.android.volley.RequestQueue;
import com.android.volley.Response;
import com.android.volley.VolleyError;
import com.android.volley.toolbox.StringRequest;
import com.android.volley.toolbox.Volley;

import org.json.JSONException;
import org.json.JSONObject;

import java.util.HashMap;
import java.util.Map;
import java.util.UUID;
```

```
/**
 * Created by Srinivasulu M on 2/4/2019.
 */
public class Addcomplaint extends AppCompatActivity {

    EditText  ct,cn,cdn,dt,adrs,mb,des,wtn;
   Button  c;

    UUID gfg2 = UUID.randomUUID();
    String uid=gfg2.toString();
    String id=uid.substring(0,5);

    private static final String
REGISTER_URL="http://dbandroid.online/crimereport/complaints.php";
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.add);

        ct=(EditText)findViewById(R.id.user);
        cn=(EditText)findViewById(R.id.a);
        cdn=(EditText)findViewById(R.id.pas);
        dt=(EditText)findViewById(R.id.name);
        adrs=(EditText)findViewById(R.id.mob);
        mb=(EditText)findViewById(R.id.add);
        des=(EditText)findViewById(R.id.ad);
        wtn=(EditText)findViewById(R.id.ad1);

        c=(Button)findViewById(R.id.su);

        c.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                if(ct.getText().toString().isEmpty()){
                    ct.setError("Complaint Type");
                }
                if(cn.getText().toString().isEmpty()){
                    cn.setError("Complaint Name");
                }
                if(cdn.getText().toString().isEmpty()){
                    cdn.setError("Candidate Name");
                }
                if(dt.getText().toString().isEmpty()){
                    dt.setError(" Date");
                }
                if(adrs.getText().toString().isEmpty()){
                    adrs.setError(" Address");
                }
                if(mb.getText().toString().isEmpty()){
                    mb.setError(" Mobiole");
                }
                if(des.getText().toString().isEmpty()){
                    des.setError(" Description");
                }
                if(wtn.getText().toString().isEmpty()){
                    wtn.setError(" Witness");
                }
                else
                {
                method();
                }
            }
        });
    }
   private void method() {
        StringRequest stringRequest = new
StringRequest(Request.Method.POST, REGISTER_URL,
                new Response.Listener<String>() {
                    @Override
                    public void onResponse(String response) {
                        try {
                            JSONObject jsonObject= null;
                            try {
                                jsonObject = new JSONObject(response);

                            } catch (JSONException e) {
                                e.printStackTrace();
                            }
                            if(jsonObject.getString("result").equals("succes")){

                                Intent i =new Intent(getApplicationContext(),Main.class);
                                startActivity(i);
                                finish();
                            }
                            Toast.makeText(Addcomplaint.this, "res..."+id,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+ct,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+cn,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+cdn,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+dt,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+adrs,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+mb,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+des,
Toast.LENGTH_SHORT).show();
                            Toast.makeText(Addcomplaint.this, "res..."+wtn,
Toast.LENGTH_SHORT).show();
                        }
                        catch (JSONException e) {
                            e.printStackTrace();
                        }
                    }
                },
                new Response.ErrorListener() {
                    @Override
                    public void onErrorResponse(VolleyError error) {
                        Toast.makeText(Addcomplaint.this, error.toString(),
Toast.LENGTH_LONG).show();
                    }
                }){
            @Override
            protected Map<String,String> getParams() throws AuthFailureError {
                Map<String,String> params = new HashMap<String, String>();

                params.put("i",id);
                params.put("c",ct.getText().toString().trim());
                params.put("cn",cn.getText().toString().trim());
                params.put("cd",cdn.getText().toString().trim());
                params.put("d",dt.getText().toString().trim());
                params.put("a",adrs.getText().toString().trim());
                params.put("m",mb.getText().toString().trim());
                params.put("de",des.getText().toString().trim());
                params.put("w",wtn.getText().toString().trim());
                return params;
            }
        };
        RequestQueue requestQueue = Volley.newRequestQueue(this);
        requestQueue.add(stringRequest);
      }
}
```

Addmsngprsn.java

```
package com.example.srinivasulum.citynews;

import android.app.DatePickerDialog;
import android.app.ProgressDialog;
import android.content.Intent;
import android.graphics.Bitmap;
import android.net.Uri;
import android.os.AsyncTask;
import android.os.Bundle;
import android.provider.MediaStore;
import android.support.v7.app.AppCompatActivity;
import android.text.InputFilter;
import android.util.Base64;
import android.view.View;
import android.widget.Button;
import android.widget.DatePicker;
import android.widget.EditText;
import android.widget.ImageView;
import android.widget.TextView;
import android.widget.Toast;

import java.io.BufferedReader;
import java.io.BufferedWriter;
import java.io.ByteArrayOutputStream;
import java.io.IOException;
import java.io.InputStreamReader;
import java.io.OutputStream;
import java.io.OutputStreamWriter;
import java.io.UnsupportedEncodingException;
import java.net.HttpURLConnection;
import java.net.URL;
import java.net.URLEncoder;
import java.text.SimpleDateFormat;
import java.util.Calendar;
import java.util.Date;
import java.util.HashMap;
import java.util.Map;
import java.util.UUID;

import javax.net.ssl.HttpsURLConnection;

import static android.widget.Toast.LENGTH_LONG;

/**
 * Created by Srinivasulu M on 12/1/2018.
 */
public class Addmsngprsn extends AppCompatActivity implements
View.OnClickListener {

    Bitmap bitmap;
    String heading = "heading" ;
    String ImagePath = "image_path" ;
    boolean check = true;

    ImageView image;
    EditText title,description,mob;
    TextView dat;
    Button  submit,pick;
    ProgressDialog progressDialog ;
    private int mYear, mMonth, mDay;

    static  String news,date,tit,mo;

    UUID gfg2 = UUID.randomUUID();
    String uid=gfg2.toString();
    String id=uid.substring(0,5);
    private static final String REGISTER_URL
="http://dbandroid.online/crimereport/missing.php";
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.addmsngprsn);

        Date What_Is_Today= Calendar.getInstance().getTime();
        SimpleDateFormat Dateformat = new SimpleDateFormat("yyyy-MM-
dd");
        String Today=Dateformat.format(What_Is_Today);
        Toast.makeText(this, Today, LENGTH_LONG).show();



        image=(ImageView)findViewById(R.id.ima);
        title=(EditText)findViewById(R.id.tit);
        description=(EditText)findViewById(R.id.des);
        mob=(EditText)findViewById(R.id.m);
        description.setFilters(new InputFilter[]{new
InputFilter.LengthFilter(500)});
        dat=(TextView)findViewById(R.id.in_date);
        submit=(Button)findViewById(R.id.su);
        pick=(Button)findViewById(R.id.pick);
        dat.setOnClickListener(this);
        submit.setOnClickListener(this);
        pick.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Intent intent = new Intent();
                intent.setType("image/*");
                intent.setAction(Intent.ACTION_GET_CONTENT);
                startActivityForResult(Intent.createChooser(intent, "Select Image
From Gallery"), 1);
            }
        });

    }

    @Override
    public void onClick(View v) {

        if(title.getText().toString().isEmpty()){
            Toast.makeText(Addmsngprsn.this, "Enter Name",
Toast.LENGTH_SHORT).show();

        }
        if(description.getText().toString().isEmpty()){
            Toast.makeText(Addmsngprsn.this, "Enter Details",
Toast.LENGTH_SHORT).show();

        }
        if(dat.getText().toString().isEmpty()){
            Toast.makeText(Addmsngprsn.this, "Select Date",
Toast.LENGTH_SHORT).show();
        }
        if(mob.getText().toString().isEmpty()){
            Toast.makeText(Addmsngprsn.this, " Enter Mobile",
Toast.LENGTH_SHORT).show();
        }


            else{
            tit=title.getText().toString();
            news=description.getText().toString();
            date=dat.getText().toString();
            mo=mob.getText().toString().trim();

            upload();
        }
        if(v.equals(dat)){
            final Calendar c = Calendar.getInstance();
            mYear = c.get(Calendar.YEAR);
            mMonth = c.get(Calendar.MONTH);
            mDay = c.get(Calendar.DAY_OF_MONTH);

            DatePickerDialog datePickerDialog = new DatePickerDialog(this,
                new DatePickerDialog.OnDateSetListener() {

                  @Override
                  public void onDateSet(DatePicker view, int year,
                             int monthOfYear, int dayOfMonth) {

                    dat.setText(year + "-" + (monthOfYear + 1) + "-" +
dayOfMonth);

                  }
                }, mYear, mMonth, mDay);
            datePickerDialog.show();
        }

    }

    protected void onActivityResult(int RC, int RQC, Intent I) {

        super.onActivityResult(RC, RQC, I);

        if (RC == 1 && RQC == RESULT_OK && I != null && I.getData() !=
null) {

            Uri uri = I.getData();

            try {
```

```
                bitmap =
MediaStore.Images.Media.getBitmap(getContentResolver(), uri);

                image.setImageBitmap(bitmap);

            } catch (IOException e) {

                e.printStackTrace();
            }
        }
    }

    private void upload() {

        Toast.makeText(Addmsngprsn.this, "iii"+id,
Toast.LENGTH_SHORT).show();
        Toast.makeText(Addmsngprsn.this, "ttt"+title,
Toast.LENGTH_SHORT).show();
        Toast.makeText(Addmsngprsn.this, "ddd"+description,
Toast.LENGTH_SHORT).show();



        ByteArrayOutputStream byteArrayOutputStreamObject ;

        byteArrayOutputStreamObject = new ByteArrayOutputStream();

        bitmap.compress(Bitmap.CompressFormat.JPEG, 100,
byteArrayOutputStreamObject);

        byte[] byteArrayVar = byteArrayOutputStreamObject.toByteArray();

        final String ConvertImage = Base64.encodeToString(byteArrayVar,
Base64.DEFAULT);
```

```
    class AsyncTaskUploadClass extends AsyncTask<Void,Void,String> {

        @Override
        protected void onPreExecute() {

            super.onPreExecute();

            progressDialog = ProgressDialog.show(Addmsngprsn.this, "Image is
Uploading", "Please Wait", false, false);
        }

        @Override
        protected void onPostExecute(String string1) {

            super.onPostExecute(string1);

            // Dismiss the progress dialog after done uploading.
            progressDialog.dismiss();

            // Printing uploading success message coming from server on
android app.
            Toast.makeText(Addmsngprsn.this,string1,
LENGTH_LONG).show();

            // Setting image as transparent after done uploading.
            image.setImageResource(android.R.color.transparent);

        }

        @Override
        protected String doInBackground(Void... params) {

            ImageProcessClass imageProcessClass = new ImageProcessClass();
```

```
                HashMap<String,String> HashMapParams = new
HashMap<String,String>();

                HashMapParams.put(heading,tit);
                HashMapParams.put("ns",news);
                HashMapParams.put("d", date);
                HashMapParams.put("i", id);
                HashMapParams.put("m", mo);

                HashMapParams.put(ImagePath, ConvertImage);

                String FinalData =
imageProcessClass.ImageHttpRequest(REGISTER_URL, HashMapParams);

                Intent i= new Intent(getApplicationContext(),Adminhome.class);
                startActivity(i);
                finish();

                return FinalData;

            }



        }
        AsyncTaskUploadClass AsyncTaskUploadClassOBJ = new
AsyncTaskUploadClass();

        AsyncTaskUploadClassOBJ.execute();
    }



    private class ImageProcessClass {
```

```
    public String ImageHttpRequest(String requestURL,HashMap<String,
String> PData) {

        StringBuilder stringBuilder = new StringBuilder();

        try {

            URL url;
            HttpURLConnection httpURLConnectionObject ;
            OutputStream OutPutStream;
            BufferedWriter bufferedWriterObject ;
            BufferedReader bufferedReaderObject ;
            int RC ;

            url = new URL(requestURL);

            httpURLConnectionObject = (HttpURLConnection)
url.openConnection();

            httpURLConnectionObject.setReadTimeout(19000);

            httpURLConnectionObject.setConnectTimeout(19000);

            httpURLConnectionObject.setRequestMethod("POST");

            httpURLConnectionObject.setDoInput(true);

            httpURLConnectionObject.setDoOutput(true);

            OutPutStream = httpURLConnectionObject.getOutputStream();

            bufferedWriterObject = new BufferedWriter(

                    new OutputStreamWriter(OutPutStream, "UTF-8"));
```

```
                bufferedWriterObject.write(bufferedWriterDataFN(PData));

                bufferedWriterObject.flush();

                bufferedWriterObject.close();

                OutPutStream.close();

                RC = httpURLConnectionObject.getResponseCode();

                if (RC == HttpsURLConnection.HTTP_OK) {

                    bufferedReaderObject = new BufferedReader(new
InputStreamReader(httpURLConnectionObject.getInputStream()));

                    stringBuilder = new StringBuilder();

                    String RC2;

                    while ((RC2 = bufferedReaderObject.readLine()) != null){

                        stringBuilder.append(RC2);
                    }
                }

            } catch (Exception e) {
                e.printStackTrace();
            }
            return stringBuilder.toString();
        }

        private String bufferedWriterDataFN(HashMap<String, String>
HashMapParams) throws UnsupportedEncodingException {
```

```
            StringBuilder stringBuilderObject;

            stringBuilderObject = new StringBuilder();

            for (Map.Entry<String, String> KEY : HashMapParams.entrySet()) {

                if (check)
                    check = false;
                else
                    stringBuilderObject.append("&");
            stringBuilderObject.append(URLEncoder.encode(KEY.getKey(),
"UTF-8"));
                stringBuilderObject.append("=");
      stringBuilderObject.append(URLEncoder.encode(KEY.getValue(), "UTF-
8"));
            }
            return stringBuilderObject.toString();
        }
    }
}
```

Android Xml Code

Activitymain.Xml

```
<?xml version="1.0" encoding="UTF-8"?>
<android.support.v4.widget.DrawerLayout tools:openDrawer="start"
android:fitsSystemWindows="true" android:layout_height="match_parent"
android:layout_width="match_parent" android:id="@+id/drawer_layout"
xmlns:tools="http://schemas.android.com/tools"
xmlns:app="http://schemas.android.com/apk/res-auto"
xmlns:android="http://schemas.android.com/apk/res/android"><include
android:layout_height="match_parent" android:layout_width="match_parent"
layout="@layout/app_bar_main"/><android.support.design.widget.Navigation
View android:fitsSystemWindows="true"
android:layout_height="match_parent" android:layout_width="wrap_content"
android:id="@+id/nav_view" app:menu="@menu/activity_main_drawer"
app:headerLayout="@layout/nav_header_main"
android:layout_gravity="start"/></android.support.v4.widget.DrawerLayout>
```

Add.Xml

```
<?xml version="1.0" encoding="UTF-8"?>
<LinearLayout android:gravity="center"
android:layout_height="match_parent" android:layout_width="match_parent"
android:orientation="vertical"
xmlns:android="http://schemas.android.com/apk/res/android"><EditText
android:layout_height="wrap_content" android:layout_width="291dp"
android:layout_marginTop="10dp" android:hint="Complaint Type"
android:id="@+id/user"/><EditText android:layout_height="wrap_content"
android:layout_width="291dp" android:layout_marginTop="10dp"
android:hint="Complaint Name" android:id="@+id/a"/><EditText
android:layout_height="wrap_content" android:layout_width="291dp"
android:layout_marginTop="10dp" android:hint="Candidate Name"
android:id="@+id/pas"/><EditText android:layout_height="wrap_content"
android:layout_width="291dp" android:layout_marginTop="10dp"
android:hint="Date" android:id="@+id/name"/><EditText
android:layout_height="wrap_content" android:layout_width="291dp"
android:layout_marginTop="10dp" android:hint="Address"
```

```
android:id="@+id/mob"/><EditText android:layout_height="wrap_content"
android:layout_width="291dp" android:layout_marginTop="10dp"
android:hint="Mobile" android:id="@+id/add"/><EditText
android:layout_height="wrap_content" android:layout_width="291dp"
android:layout_marginTop="10dp" android:hint="Description"
android:id="@+id/ad"/><EditText android:layout_height="wrap_content"
android:layout_width="291dp" android:layout_marginTop="10dp"
android:hint="List Any Witness" android:id="@+id/ad1"/><Button
android:layout_height="wrap_content" android:layout_width="291dp"
android:hint="Complaint" android:id="@+id/su"/></LinearLayout>
```

7. System Testing

The purpose of testing is to discover errors. Testing is the process of trying to discover every conceivable fault or weakness in a work product. It provides a way to check the functionality of components, sub assemblies, assemblies and/or a finished product It is the process of exercising software with the intent of ensuring that the
Software system meets its requirements and user expectations and does not fail in an unacceptable manner. There are various types of test. Each test type addresses a specific testing requirement.

Test Cases

- **Unit testing**

Unit testing involves the design of test cases that validate that the internal program logic is functioning properly, and that program inputs produce valid outputs. All decision branches and internal code flow should be validated. It is the testing of individual software units of the application .it is done after the completion of an individual unit before integration. This is a structural testing, that relies on knowledge of its construction and is invasive. Unit tests perform basic tests at component level and test a specific business process, application, and/or system configuration. Unit tests ensure that each unique path of a business process performs accurately to the documented specifications and contains clearly defined inputs and expected results.

- **Integration testing**

Integration tests are designed to test integrated software components to determine if they actually run as one program. Testing is event driven and is more concerned with the basic outcome of screens or fields. Integration tests demonstrate that although the components were individually satisfaction, as shown by successfully unit testing, the combination of components is correct and consistent. Integration testing is specifically aimed at exposing the problems that arise from the combination of components.

- **Functional test**

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

Valid Input : identified classes of valid input must be accepted.
Invalid Input : identified classes of invalid input must be rejected.
Functions : identified functions must be exercised.
Output : identified classes of application outputs must be exercised.
Systems/Procedures : interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

- **System Test**

System testing ensures that the entire integrated software system meets requirements. It tests a configuration to ensure known and predictable results. An example of system testing is the configuration oriented system integration test. System testing is based on process descriptions and flows, emphasizing pre-driven process links and integration points.

- **White Box Testing**

White Box Testing is a testing in which in which the software tester has knowledge of the inner workings, structure and language of the software, or at least its purpose. It is purpose. It is used to test areas that cannot be reached from a black box level.

- **Black Box Testing**

Black Box Testing is testing the software without any knowledge of the inner workings, structure or language of the module being tested. Black box tests, as most other kinds of tests, must be written from a definitive source document, such as specification or requirements document, such as specification or requirements document. It is a testing in which the software under test is treated, as a black box .you cannot “see” into it. The test provides inputs and responds to outputs without considering how the software works.

Test Report

- **Unit Testing:**

Unit testing is usually conducted as part of a combined code and unit test phase of the software lifecycle, although it is not uncommon for coding and unit testing to be conducted as two distinct phases.

Test strategy and approach

Field testing will be performed manually and functional tests will be written in detail.

Test objectives

- All field entries must work properly.
- Pages must be activated from the identified link.
- The entry screen, messages and responses must not be delayed.

Features to be tested

- Verify that the entries are of the correct format
- No duplicate entries should be allowed
- All links should take the user to the correct page.

- **Integration Testing**

Software integration testing is the incremental integration testing of two or more integrated software components on a single platform to produce failures caused by interface defects.

The task of the integration test is to check that components or software applications, e.g. components in a software system or – one step up – software applications at the company level – interact without error.

Test Results: All the test cases mentioned above passed successfully. No defects encountered.

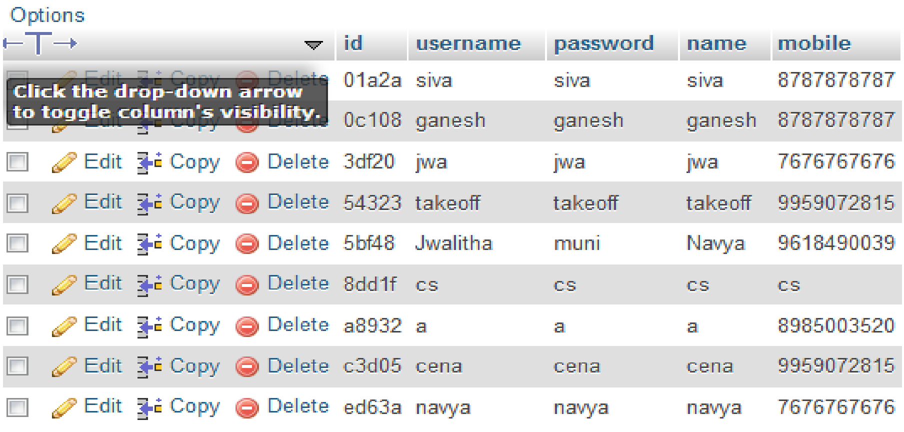
- **Acceptance Testing**

User Acceptance Testing is a critical phase of any project and requires significant participation by the end user. It also ensures that the system meets the functional requirements.

Test Results: All the test cases mentioned above passed successfully.

defects encountered.

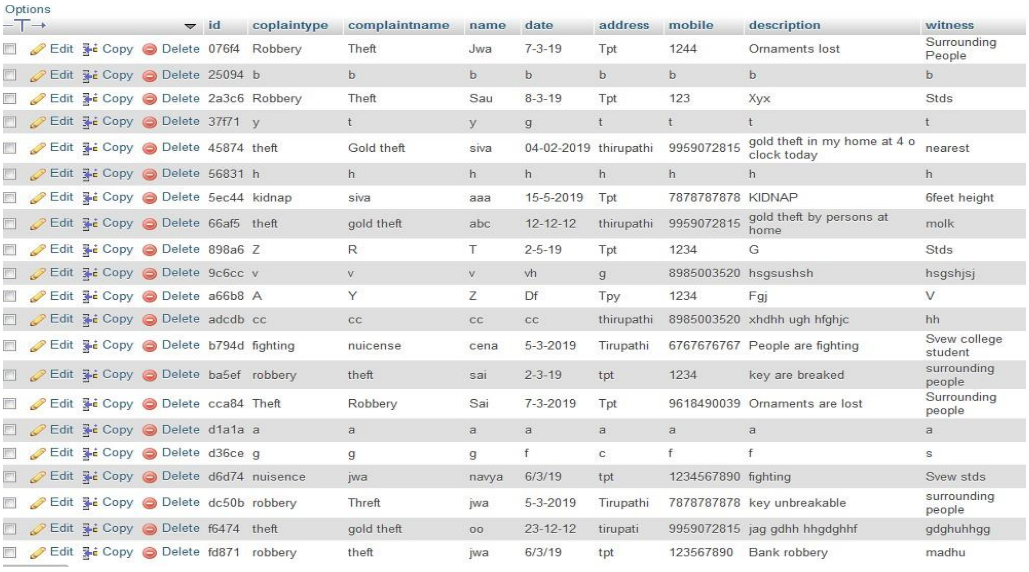
8. Output Screenshots



Click the drop-down arrow to toggle column's visibility.

Options	id	username	password	name	mobile
□ Edit Copy Delete	01a2a	siva	siva	siva	8787878787
□ Edit Copy Delete	0c108	ganesh	ganesh	ganesh	8787878787
□ Edit Copy Delete	3df20	jwa	jwa	jwa	7676767676
□ Edit Copy Delete	54323	takeoff	takeoff	takeoff	9959072815
□ Edit Copy Delete	5bf48	Jwalitha	muni	Navya	9618490039
□ Edit Copy Delete	8dd1f	cs	cs	cs	cs
□ Edit Copy Delete	a8932	a	a	a	8985003520
□ Edit Copy Delete	c3d05	cena	cena	cena	9959072815
□ Edit Copy Delete	ed63a	navya	navya	navya	7676767676

FIG:8.1Status of registeration process in server database.



Options	id	coplaintype	complaintname	name	date	address	mobile	description	witness
□ Edit Copy Delete	076f4	Robbery	Theft	Jwa	7-3-19	Tpt	1244	Ornaments lost	Surrounding People
□ Edit Copy Delete	25094	b	b	b	b	b	b	b	b
□ Edit Copy Delete	2a3c6	Robbery	Theft	Sau	8-3-19	Tpt	123	Xyx	Stds
□ Edit Copy Delete	37f71	y	t	y	g	t	t	t	t
□ Edit Copy Delete	45874	theft	Gold theft	siva	04-02-2019	thirupathi	9959072815	gold theft in my home at 4 o clock today	nearest
□ Edit Copy Delete	56831	h	h	h	h	h	h	h	h
□ Edit Copy Delete	5ec44	kidnap	siva	aaa	15-5-2019	Tpt	7878787878	KIDNAP	6feet height
□ Edit Copy Delete	66af5	theft	gold theft	abc	12-12-12	thirupathi	9959072815	gold theft by persons at home	molk
□ Edit Copy Delete	898a6	Z	R	T	2-5-19	Tpt	1234	G	Stds
□ Edit Copy Delete	9c6cc	v	v	v	vh	g	8985003520	hsgsushsh	hsgshjsj
□ Edit Copy Delete	a66b8	A	Y	Z	Df	Tpy	1234	Fgj	V
□ Edit Copy Delete	adcdb	cc	cc	cc	cc	thirupathi	8985003520	xhdhh ugh hfghjc	hh
□ Edit Copy Delete	b794d	fighting	nuicense	cena	5-3-2019	Tirupathi	6767676767	People are fighting	Svew college student
□ Edit Copy Delete	ba5ef	robbery	theft	sai	2-3-19	tpt	1234	key are breaked	surrounding people
□ Edit Copy Delete	cca84	Theft	Robbery	Sai	7-3-2019	Tpt	9618490039	Ornaments are lost	Surrounding people
□ Edit Copy Delete	d1a1a	a	a	a	a	a	a	a	a
□ Edit Copy Delete	d36ce	g	g	g	f	c	f	f	s
□ Edit Copy Delete	d6d74	nuisence	jwa	navya	6/3/19	tpt	1234567890	fighting	Svew stds
□ Edit Copy Delete	dc50b	robbery	Threft	jwa	5-3-2019	Tirupathi	7878787878	key unbreakable	surrounding people
□ Edit Copy Delete	f6474	theft	gold theft	oo	23-12-12	tirupati	9959072815	jag gdhh hhgdghhf	gdghuhhgg
□ Edit Copy Delete	fd871	robbery	theft	jwa	6/3/19	tpt	123567890	Bank robbery	madhu

FIG:8.2Status of add complaints process in server database.

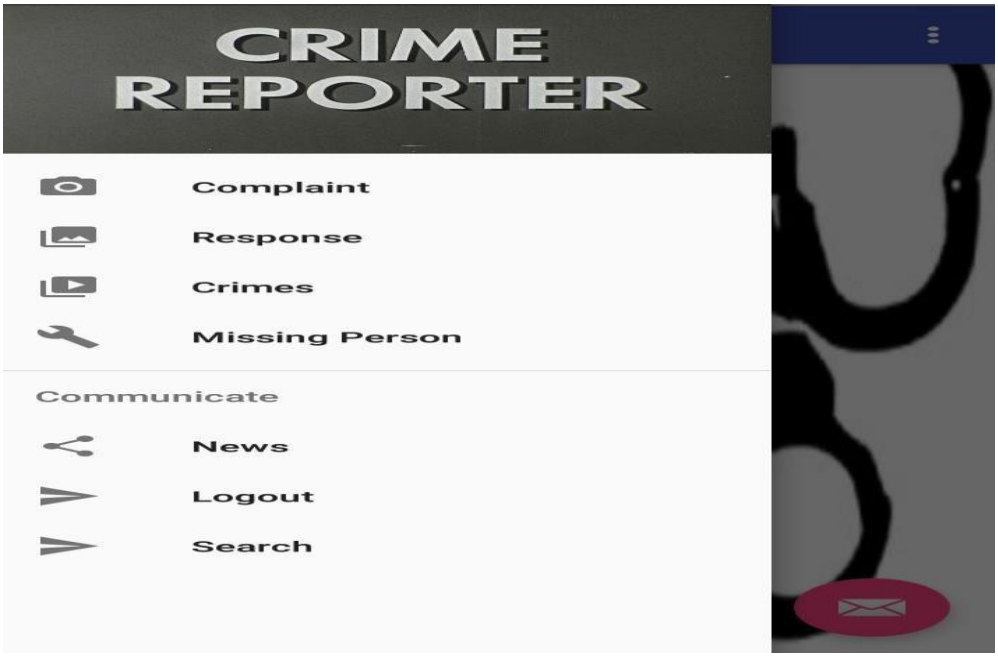


Fig: 8.3 Home page

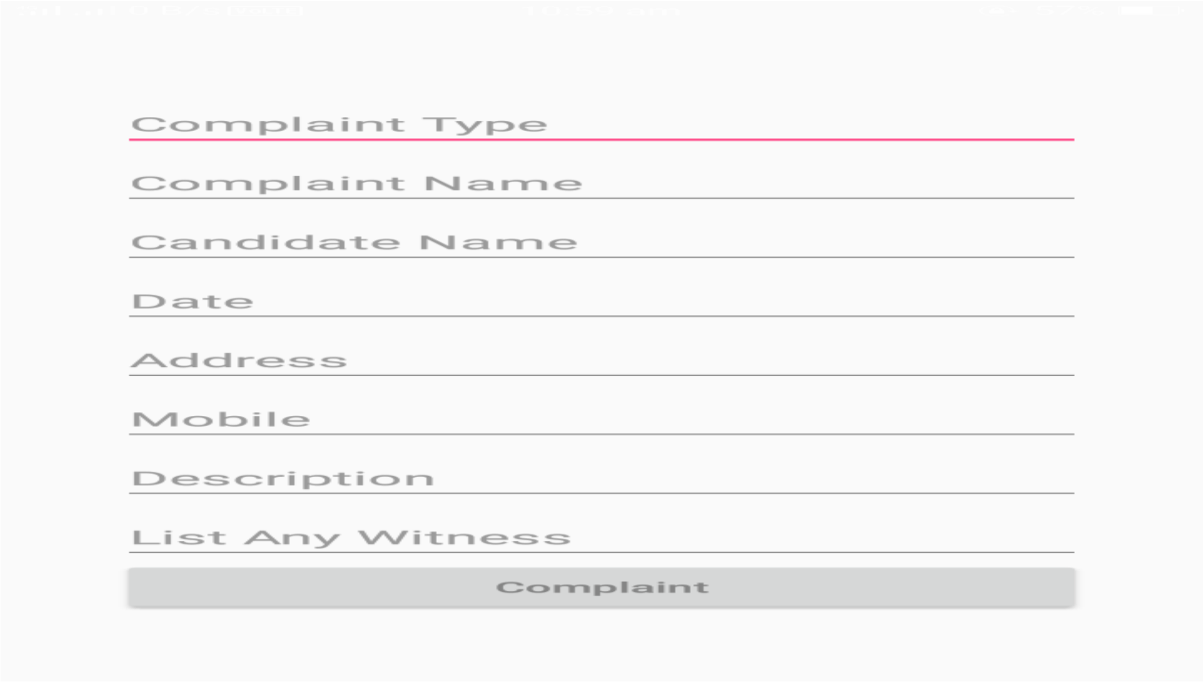
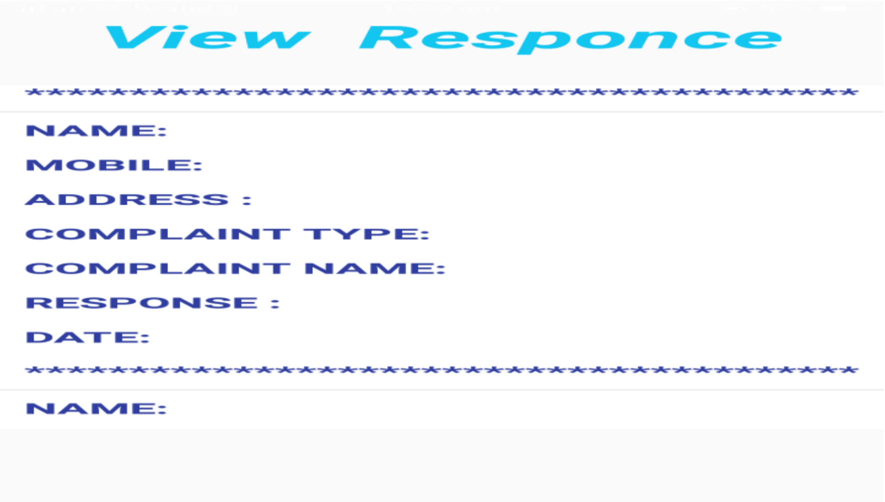


Fig:8.4 Complaint page



View Responce

****

NAME:

MOBILE:

ADDRESS :

COMPLAINT TYPE:

COMPLAINT NAME:

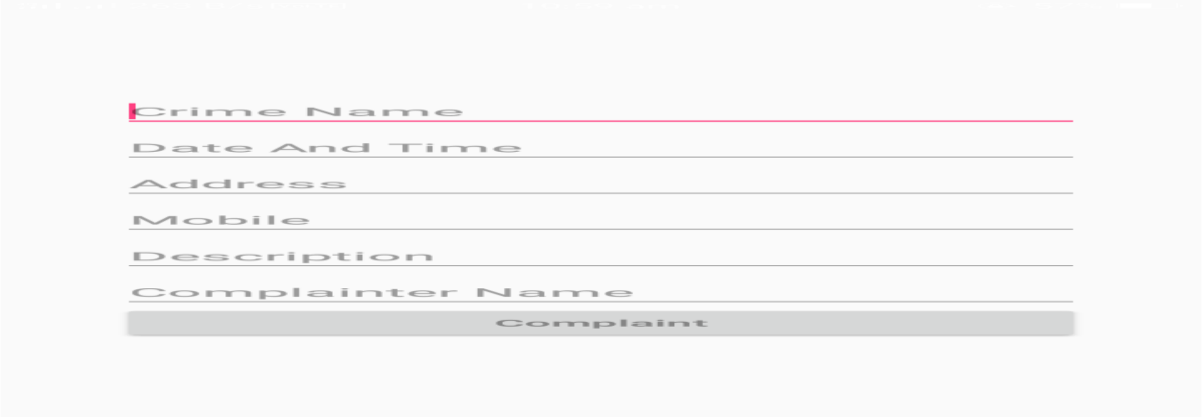
RESPONSE :

DATE:

****

NAME:

Fig:8.5 Response page



Crime Name

Date And Time

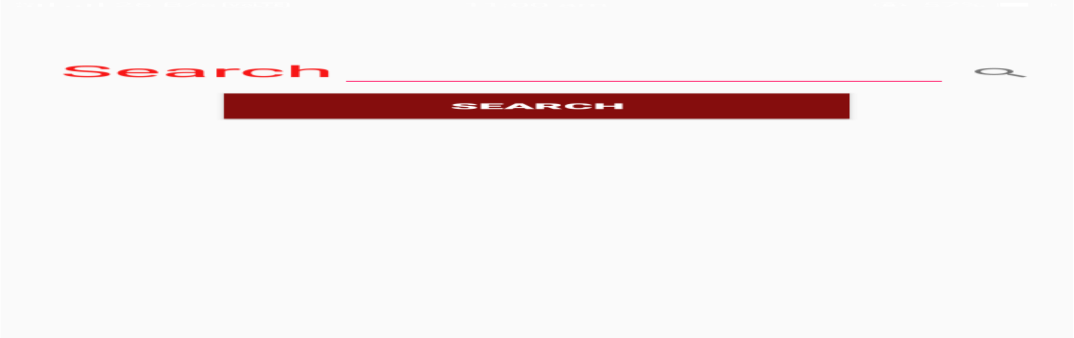
Address

Mobile

Description

Complainter Name

Fig:8.6 Complaint page for crime



Search

Fig: 8.7 Search page

9. Future Scope of the Project

The system can be further developed in the last two years, the number of mobile phone users rapidly increased and the counting is still on. India stands second in the number of active mobile phone users in the world. Out of six billion mobile phones in the world, around one billion is being used in India, according to the 70% of India’s current population. Around 6 million subscribers join the group every month. The top mobile app development company currently works on the latest mobile development platforms.

People using smart-phones are demanding for better applications and want to update existing ones, which in turn created a huge scope for Android application development in India.

Nowadays Android has become very popular as it is an open-source, Linux-based operating system, mainly designed by Google for smart-phones and tablets. It is designed in such a way that allows the developers and device manufacturers to alter the software design according to their needs. Let’s jump to the facts why android development is popular these days.

No doubt, Android is taking over the world’s tech. The Android operating system is holding 85% of the total market share for the mobile operating system. The businesses who are looking forward to investing in the mobile application should think twice.

10. Conclusion

The study also looked at the crime rates and crime of frequencies in the city. From results, we can deduce that Lusaka is actually a very violent city and its residents are exposed to high levels of crime. Residents experience a lot of challenges with regards reporting of the crime because of the long distances to police stations and the fact that very few citizens even know the phone numbers of police stations and can therefore not contact them in case of an emergency. The other reason is that the Zambia police toll free lines hardly work and calls made to the toll free number go unanswered. To safeguard people's lives and property, the integration of its in crime reporting and monitoring process is key as this will effectively bridge the communication gap between the police and the general public in fighting crime. The ultimate result of this study is to therefore develop a mobile crime fighting application that will be used for crime reporting and monitoring.

11. Bibliography

REFERENCES

[1] Gift Lyoko Mwangala Mwiya1, Jackson Phiri. Public crime reporting and monitoring system model using gsm and gis technologies: A case of zambia police service. Mwangala Mwiya et al, International Journal of Computer Science and Mobile Computing, 4, 2015.

[2] Dipali Thavare Rahul Patil Mrs. Sonali Pakhmode Rahul Mote, Pooja Rakate. Android based application on missing person finder. IJSRD - International Journal for Scientific Research & Development, 3, 2015.

[3] D Sai Suma A Chaithanya V. Yadagiri, C Hruthik Teja. Crime reporter and missing person finder. International Journal of Smart Home, 4, 2017.