# Introduction to project

**1. Background**

Now-a-day’s women security is big question and this project is an answer to this question. This saves time and at the same time creates sense of security where ever she is.

There was need felt that should recognise voice for first time when the App is started.

The SMS shall be automatically sent as soon as the word Help is heard.

This application is generally meant for the attention of the authorities or public in the emergency response capabilities such as terrorist attacks and the natural disaster by facilitating the communication with their respective along the mobile phones.

The difficulties in the existing application are the lack of situational awareness and communication terminology among their respective. Due to this response and recovery is Difficult to the authorities. In respect of the public safety with the support of the network provider the application runs in the android phones in efficient way to identify and recover the problem by the natural disaster or terrorist attacks etc…

Furthermore users are likely to operate the mobile devices for the security purpose to intimate the problem detection to their respective in the emergency cases.

This software is very usable for those who wish to seek help when they in condition of needing help from anyone. The project is related to Women security and makes them independent to go anywhere. They only have to say the word Help to seek help from others.

This software will be used by:

One anyone who is interested in seeking Help from other?

**2. Considered area of optimization /work**

Android is a software stack for mobile devices that includes an operating system, middleware and key applications. Android is a software and operating system for mobile devices based on the Linux Operating System and developed by Google and the Open Handset Alliance. It allow developers to write managed code in JAVA-like language that utilizes Google-developed Java libraries, but does not support programs developed in native code.

The unveiling of the Android platform on 5 November 2007, was announced with the founding of Open Handset Alliance, consortium of 34 hardware, software and telecom companies devoted to advancing open standards for mobile devices. When released in 2008, most of the Android platform will be made available under the Apache free-software and open-source license.

There are a number of technological and organizational skills involved in building a software proHCETt, and the process presented with an opportunity to foster such skills – firstly, to learn a new platform that will likely find its way to the top of mobile segment within the next few years, along with an array of client-server technologies associated with the system; secondly, to get a chance to work with a real proHCETt that concerns a charitable cause; thirdly, to create a foundation for future development of actual Android-based hardware with the inputting and tracing functions built in it.

## 3. Project objectives and related benefits

***“Helping System Android App”*** is a Software Developed to aid any woman in getting help from the predefined number she has saved on her mobile. This is special software developed only for girls and women who are in a danger zone and wish to get help from their near and dear ones or it can be police. The App thus developed needs internet connection for the successful execution. The App works on the principle of voice recognition and shall send an SMS to the person whom she considers can help at that time.

The App is doing the work of guard who is there all the time with her and if she says the word “help” from her mouth that special voice shall be recognized and thus SMS sent to the Contact Number she has stated when starting this App.

We will develop the software for the sake of learning new things in Android and Windows and using some of the hidden features of the language.

**4. Software model comparison and conclusion (this project)**

## b. Analysis and analysis Modeling(conventional approach)

## i. Requirement analysis.

Modern handheld devices such as smart phones and PDA’s have become increasingly powerful in recent years. Dramatic breakthroughs in processing power along with the number of extra features included in these devices have opened the doors to a wide range of commercial possibilities. In particular, most cell phones regularly include cameras, processors comparable to PCs from only a few years ago, and internet access. However even with all these abilities, there are a few applications that allow much passing of the environmental information and location based services.

As mobile devices become more like PCs they will come to replace obHCETts we tend to carry around such as cheque books, credit cards, cameras, planners, mp3 players, etc. In short, we will be using them to accomplish our daily tasks. On application that falls into this category is the HCET College Android App developed for Android phones.

The prime obHCETtive of “HCET College Android App” is to create a full fledged Android application which could locate a list of doctors, hospitals based on location and type of specialty entered by the user. The user not finds a list of Doctors or Hospitals but also he can make a choice of the best doctor based on the rating and specialty he chooses to have. The user has the advantage of placing an emergency call to any of the doctors. The applications fruits doesn’t end over here, further going ahead user can track all the donors wishing to donate a specified blood group. The donors list shall be present as soon as the blood group is specified by the user.

The proHCETt shall be developed using Java Programming Language by using Netbeans IDE. We shall be making use of Android Software Development Kit which includes a variety of custom tools that help us develop mobile applications on the Android platform. The most important of these are the Android Emulator and Android Development Tools plug-in for Netbeans.

Android

Android is a new player in the mobile industry, but it is rapidly gaining support in the software development circles.

Its software is written in JAVA, perhaps the most commonly used language today with support of thousands of modules, extensions and pages of documentation. It’s open and linux based and gives direct access to hardware, meaning that “the only limitation is imagination” in designing Android software.

Google has made steps to cement the platform’s reputation among developers and ensure its future development – steps such as producing regular updates to the SDK and founding the Open Handset Alliance.

Because of the system’s openness and flexibility it’s easy to produce a budget Android-based device to be sold in the targeted areas - the deciding factor of choosing Android over the closed iPhone framework that provides a similar set of features.

Unfortunately, the fact that Android first came to light only a couple of years ago presented a challenge; some of its functions were still poorly documented and required outside research. In addition, the platform has undergone drastic changes and much of code had to be redone.

Mobile Telephony

An Android-based device can receive text messages; in this context it serves as a hub for several mobile phones in vicinity limited only by the network provider’s range of service. The idea for this function originated from a designer who said, “it’s easy to design an Android device. It’s harder to mimic its functionality on a cheap mobile phone.”

The mobile phone users can send coded text messages to the android device in order to input product-related data, or request market information, in which case a response is sent back to the phone. The advantages of this scheme are evident:

1. SMS functionality can be found on virtually any mobile phone that’s less than 2 decades old;
2. Text messages are a universally accepted way of communication and can be sent by someone with minimal technological skills;
3. There’s no need for a 3rd-party receiving service that charges additional fees, the only cost is the cost of sending a message.

There are certain limitations too:

1. The receiving device has to be switched on and the application has to be running;
2. The sender must be within the range of service of the network provider.

Therefore the following arrangement takes place from the producer’s point of view:



Data Model

## Data Flow Diagram

Candidate

Validate

Behavior Model

Person

System

Functional Modeling



4. The software requirement specification

## Hardware Interfaces

Any Mobile with Android OS

## Software Interfaces

* Front End Tool :Netbeans
* Language : JAVA
* Android SDK

Software project planning

i.Project planning benefits

## Pert Chart

The success of any large-scale project is very much dependent upon the quality of planning, scheduling and controlling of the various phase of the project. Unless some type of planning and coordinating tool is used, the number of phases does not to be very large before management starts losing controls. One such OR tool used on large-scale project to aid management in expanding and controlling the utilization of personal, material, facilities, and time is the program evaluation and review technique (PERT). This technique is used to pinpoint critical areas in a project necessary adjustments can be made in order to meet the schedule completion date the project.

Project scheduling by PERT consist of four main steps:-

Planning

Scheduling

Allocating of resources

Controlling

The Pert chart of the overall implementation is shown in following figures:-

|  |  |  |
| --- | --- | --- |
| NODE | DESCRIPTION | UTILIZE THE NO. OF DAYS |
| 1 | Selection of organization | 10 Days |
| 2 | Selection of problem | 7 Days |
| 3 | Collecting the information | 25 Days |
| 4 | Analysis of information | 10 Days |
| 5 | Designing of the Application | 40 Days |
| 6 | Coding for the Application | 45 Days |
| 7 | Review the specification | 7 Days |
| 8 | Validation / Verification | 7 Days |
| 9 | Testing | 18 Days |
| 10 | Dummy activity | 5 Days |

THE PERT Chart is representing the interdependencies among tasks. The PERT provides quantitative tools that allow the software planner to determine the critical path. The PERT Chart use is justified in large projects

.

### THE PERT Chart Diagram

Node 1 - Organization Selection

Node 2 - Proj / Prob Selection

|  |
| --- |
| Notations :  Notations : =>Critical Path  -> Activity  --- > Dummy Activity  Node |

Node 3 - Information collect

Node 4 – Analysis of Information

Node 5 - Design of the application

Node 6 - Review the Specification

Node 7 - Coding

Node 8 - Verification/Validation

Node 9 – Testing

Node 10 – Dummy Activity

**ii Project Scope**

The proposed system i.e. ***“Helping System Android App”*** *is* an App based system with advanced security feature. The system is developed in Android as a front-end tool under the windows platform. The proposed system will take over the existing system very efficiently. The scope of the system includes the following feature:

* A number is required for SMS.
* Internet connection needs to be present.
* The App shall recognise the voice of the user.
* The SMS is feed beforehand.

# iii. Software project estimation.

## GANTT Chart

Which uses a calendar–oriented chart to represent the project schedule. Each activity is represented as a bar in the calendar, starting from the starting date of activity and ending at the ending date for that activity. The start and end of each activity become milestones (or check points) for the project. Coloring each milestone when completed can represent progress. The milestone or checkpoints are usually at the completion of each task.

Let us draw the Gantt chart for each task identified in these project .the horizontal bars indicates the duration of each task and all project tasks are listed in the left handed column. We estimate the no. of days for each task as follows: -Selection of Organization-10 days, Selection of problem-7days,Collection of information –25 days ,Analysis of information –10 days ,Designing the application – 40 days ,Coding for the application -45 days , Review the specification –7days, Verification/Validation – 7 days, Testing -18 days , Dummy activity –5 days , Implementation and follow-up –10 days, user training –7 days

Drawback of GANNT Chart: The main drawback of Gantt chart is that it does not depict the dependency relationships among the different activity.

# 

# Software design

# i.System design

# 1. Modular decomposition and explanation

The complete project is divided in to two modules:

* Page designing module
* Programming and logic module
* Registration Module
* Voice Detection Module
* SMS with Location Send Module

### Page Designing Module:

In this module we create pages, that are desired for giving help to the person who wishes to seek help. Also by these pages we will send SMS on recognition of voice of that user. Also by this module we will register the user. Interface developed shall be Android based pages for running on mobile.

### Programming & Logic Module:

This is the very important module of this project. This module is responsible for all the required under laying programming and logic for the project. In this module we will make possible to put up the information and processing this information of the user. This module will detect all the clicking events and controls the outputs of the system according to the user request. This module will define all the logic and programming aspects of the system. This module will define all the security constraint and validation for the entire system.

In this module we will use android.

**SMS send Module**

Text messaging, or texting, refers to the exchange of written messages between fixed-line phone or mobile phone and fixed or portable devices over a network. SMS Manager contains a powerful rule editor which can be used to automate message processing.

# Product Features

HTTP/HTTPS

TCP/IP

Web Server

Chat Client

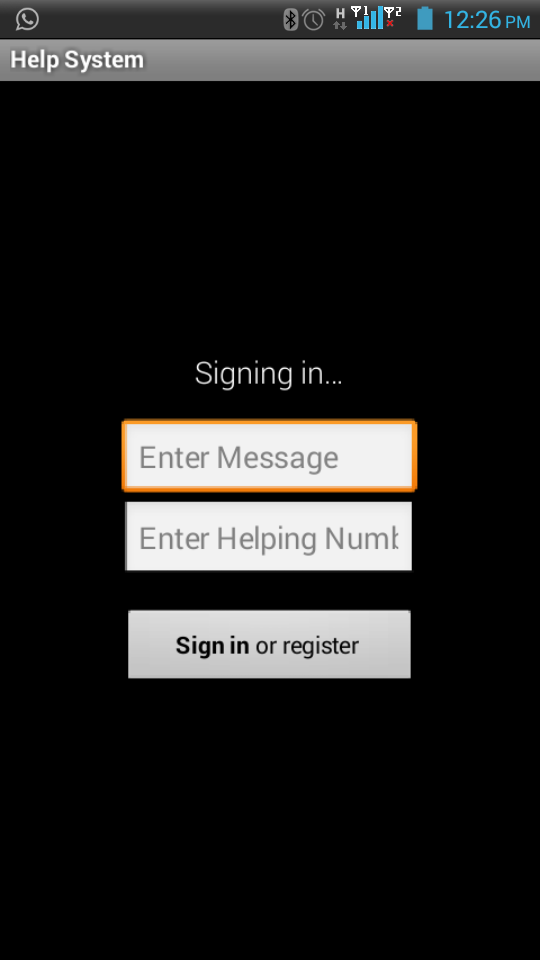
Client Software (System User)

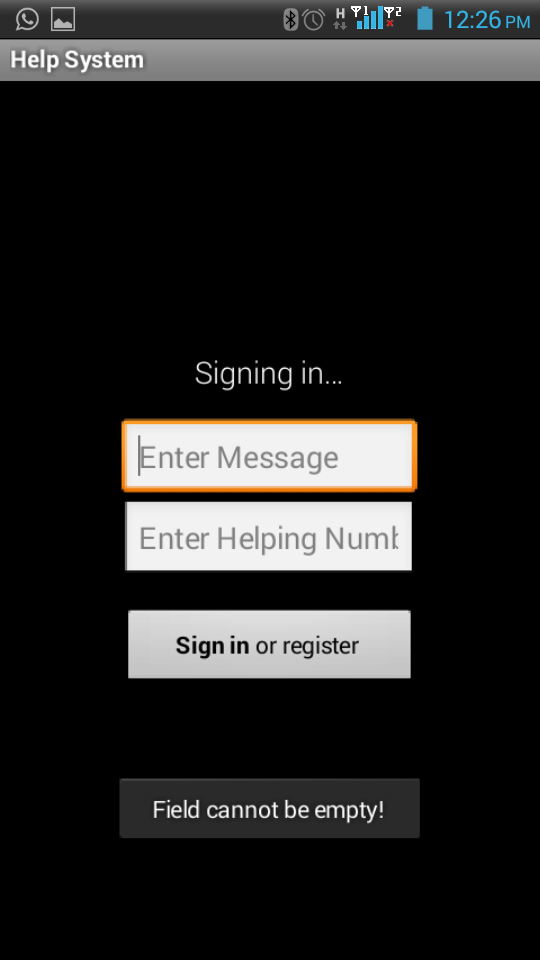
Client Side

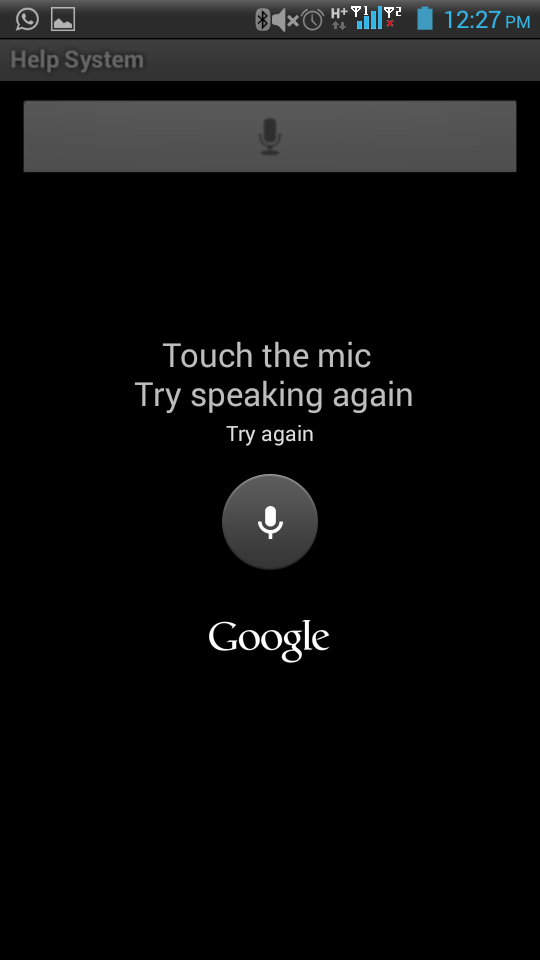
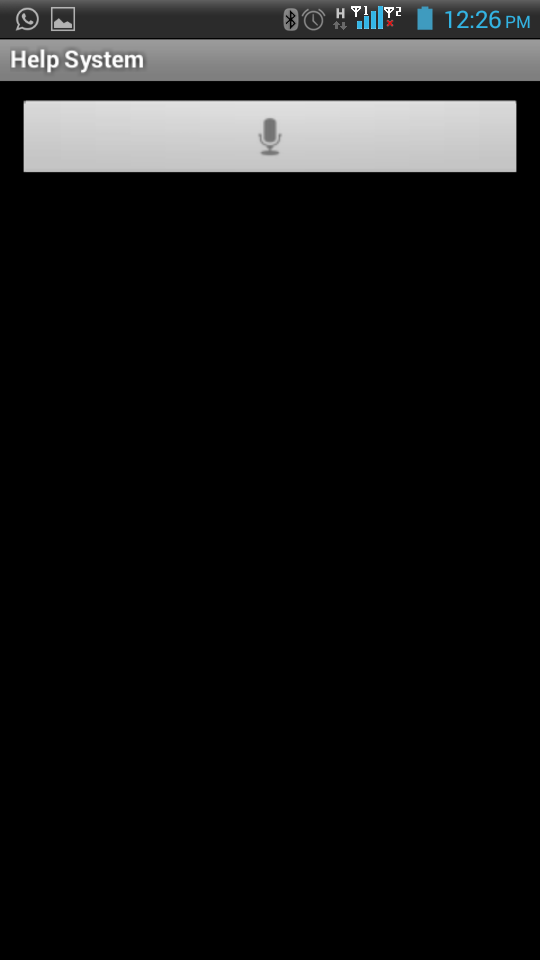
Application Server

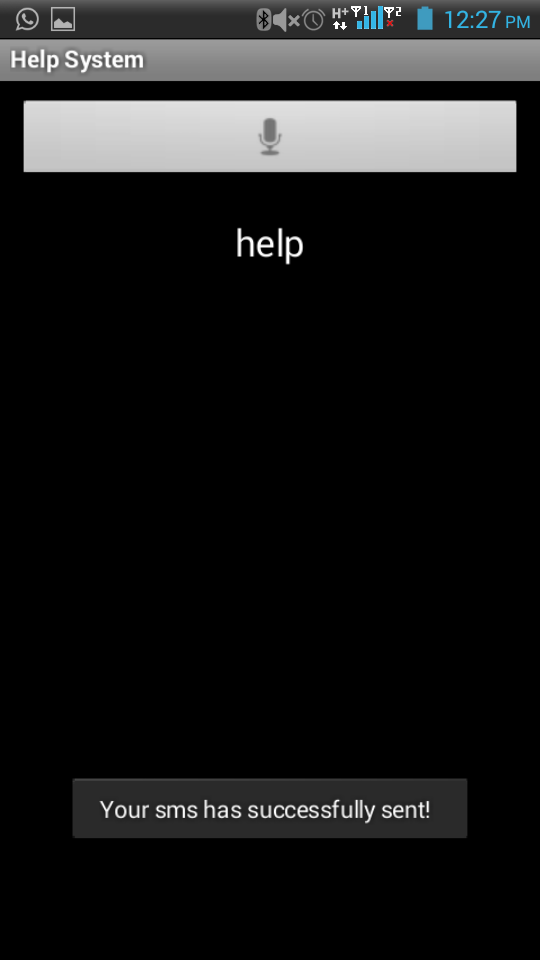
* The Helping System Android App is present to provide the user interface on client side. Communication between client and server is provided through HTTP/HTTPS protocols.
* The client software is to provide the user interface on system user client side and for this TCP/IP protocols are used.
* On the server side web server is for interaction.

Some Screen Shot for Module









## Assumptions and Dependencies

* Internet Connection is must.
* The mobile should have enough amount to send an SMS.

# Tools used

## Android

**Android** is an operating system based on the Linux kernel, and designed primarily for touchscreen mobile devices such as smartphones and tablet computers. Initially developed by Android, Inc., which Google backed financially and later bought in 2005,Android was unveiled in 2007 along with the founding of the Open Handset Alliance: a consortium of hardware, software, and telecommunication companies devoted to advancing open standards for mobile devices. The first Android-powered phone was sold in October 2008.

The user interface of Android is based off direct manipulation, using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching and reverse pinching to manipulate on-screen obHCETts. Internal hardware such as accelerometers, gyroscopes and proximity sensors are used by some applications to respond to additional user actions, for example adjusting the screen from portrait to landscape depending on how the device is oriented. Android allows users to customize their homescreens with shortcuts to applications and widgets, which allow users to display live content, such as emails and weather information, directly on the homescreen. Applications can further send notifications to the user to inform them of relevant information, such as new emails and text messages.

Android is open source and Google releases the code under the Apache License. This open-source code and permissive licensing allows the software to be freely modified and distributed by device manufacturers, wireless carriers and enthusiast developers. However, most Android devices ship with additional proprietary software. Additionally, Android has a large community of developers writing applications ("apps") that extend the functionality of devices, written primarily in the Java programming language. In October 2012, there were approximately 700,000 apps available for Android, and the estimated number of applications downloaded from Google Play, Android's primary app store, was 25 billion. A developer survey conducted in April–May 2013 found that Android is the most popular platform for developers, used by 71% of the mobile developer population.

Android is the world's most widely used smartphone platform, overtaking Symbian in the fourth quarter of 2010. Android is popular with technology companies who require a ready-made, low-cost, customizable and lightweight operating system for high techdevices. Despite being primarily designed for phones and tablets, it has been also used in televisions, games consoles, digital cameras and other electronics. Android's open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven proHCETts, which add new features for advanced users or bring Android to devices which were officially released running other operating systems.

Android's share of the global smartphone market, led by Samsung products, was 64% in March 2013. The operating system's success has made it a target for patent litigation as part of the so-called "smartphone wars" between technology companies. As of May 2013, 48 billion apps have been installed from the Google Play store, and as of September 3, 2013, 1 billion Android devices have been activated.

## Features

### Interface

Android's user interface is based on direct manipulation, using touch inputs that loosely correspond to real-world actions, like swiping, tapping, pinching and reverse pinching to manipulate on-screen obHCETts. The response to user input is designed to be immediate and provides a fluid touch interface, often using the vibration capabilities of the device to provide haptic feedback to the user. Internal hardware such as accelerometers, gyroscopes and proximity sensors are used by some applications to respond to additional user actions, for example adjusting the screen from portrait to landscape depending on how the device is oriented, or allowing the user to steer a vehicle in a racing game by rotating the device, simulating control of a steering wheel.

Android devices boot to the homescreen, the primary navigation and information point on the device, which is similar to the desktrop found on PCs. Android homescreens are typically made up of app icons and widgets; app icons launch the associated app, whereas widgets display live, auto-updating content such as the weather forecast, the user's email inbox, or a news ticker directly on the homescreen. A homescreen may be made up of several pages that the user can swipe back and forth between, though Android's homescreen interface is heavily customisable, allowing the user to adjust the look and feel of the device to their tastes. Third-party apps available on Google Play and other app stores can extensively re-theme the homescreen, and even mimic the look of other operating systems, such as Windows Phone. Most manufacturers, and some wireless carriers, customise the look and feel of their Android devices to differentiate themselves from their competitors.

Present along the top of the screen is a status bar, showing information about the device and its connectivity. This status bar can be "pulled" down to reveal a notification screen where apps display important information or updates, such as a newly received email or SMS text, in a way that does not immediately interrupt or inconvenience the user. In early versions of Android these notifications could be tapped to open the relevant app, but recent updates have provided enhanced functionality, such as the ability to call a number back directly from the missed call notification without having to open the dialer app first. Notifications are persistent until read or dismissed by the user.

### Applications

Android has a growing selection of third party applications, which can be acquired by users either through an app store such as [Google Play](http://en.wikipedia.org/wiki/Google_Play) or the [Amazon Appstore](http://en.wikipedia.org/wiki/Amazon_Appstore), or by downloading and installing the application's [APK](http://en.wikipedia.org/wiki/APK_(file_format)) file from a third-party site. The Play Store application allows users to browse, download and update apps published by Google and third-party developers, and is pre-installed on devices that comply with Google's compatibility requirements. The app filters the list of available applications to those that are compatible with the user's device, and developers may restrict their applications to particular carriers or countries for business reasons. Purchases of unwanted applications can be refunded within 15 minutes of the time of download, and some carriers offer direct carrier billing for Google Play application purchases, where the cost of the application is added to the user's monthly bill. As of September 2012, there were more than 675,000 apps available for Android, and the estimated number of applications downloaded from the Play Store was 25 billion.

Applications are developed in the [Java](http://en.wikipedia.org/wiki/Java_(programming_language)) language using the [Android software development](http://en.wikipedia.org/wiki/Android_software_development) kit (SDK). The SDK includes a comprehensive set of development tools, including a [debugger](http://en.wikipedia.org/wiki/Debugger), [software libraries](http://en.wikipedia.org/wiki/Software_library), a handset [emulator](http://en.wikipedia.org/wiki/Emulator) based on QEMU, documentation, sample code, and tutorials. The officially supported [integrated development environment](http://en.wikipedia.org/wiki/Integrated_development_environment) (IDE) is [Eclipse](http://en.wikipedia.org/wiki/Eclipse_(software)) using the Android Development Tools (ADT) plugin. Other development tools are available, including a [Native Development Kit](http://en.wikipedia.org/wiki/Native_development_kit) for applications or extensions in C or C++, [Google App Inventor](http://en.wikipedia.org/wiki/Google_App_Inventor), a visual environment for novice programmers, and various cross platform mobile web applications frameworks.

### Memory Management

Since Android devices are usually battery-powered, Android is designed to manage memory ([RAM](http://en.wikipedia.org/wiki/Random-access_memory)) to keep power consumption at a minimum, in contrast to desktop operating systems which generally assume they are connected to unlimited [mains electricity](http://en.wikipedia.org/wiki/Mains_electricity). When an Android app is no longer in use, the system will automatically suspend it in memory – while the app is still technically "open," suspended apps consume no resources (e.g. battery power or processing power) and sit idly in the background until needed again. This has the dual benefit of increasing the general responsiveness of Android devices, since apps don't need to be closed and reopened from scratch each time, but also ensuring background apps don't consume power needlessly.

Android manages the apps stored in memory automatically: when memory is low, the system will begin killing apps and processes that have been inactive for a while, in reverse order since they were last used (i.e. oldest first). This process is designed to be invisible to the user, such that users do not need to manage memory or the killing of apps themselves. However, confusion over Android memory management has resulted in third-party task killers becoming popular on the Google Play store; these third-party task killers are generally regarded as doing more harm than good.

### Hardware Requirements

As of November 2013, current versions of Android require at least 512 MB of RAM, and a 32-bit ARMvr7, MrIPS or x86 architecture processor, together with an OpenGL ES 2.0 compatible graphics processing unit (GPU).

The main hardware platform for Android is the 32-bit ARMv7 architecture. There is support for x86 from the Android-x86 proHCETt, and Google TV uses a special x86 version of Android. In 2013, Freescale announced Android on its i.MX processor, i.MX5X and i.MX6X series. In 2012 Intel processors began to appear on more mainstream Android platforms, such as phones

### Updates

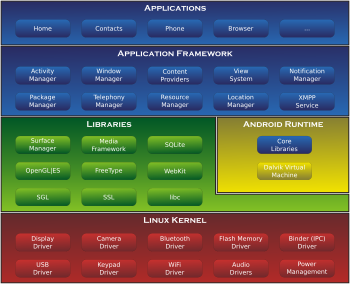
Google provides major updates, incremental in nature, to Android every six to nine months, which most devices are capable of receiving over the air. The latest major update is Android 4.4 KitKat.

Compared to its chief rival mobile operating system, namely iOS, Android updates are typically slow to reach actual devices. For devices not under the Nexus brand, updates often arrive months from the time the given version is officially released. This is caused partly due to the extensive variation in hardware of Android devices, to which each update must be specifically tailored, as the official Google source code only runs on their flagship Nexus devices. Porting Android to specific hardware is a time- and resource-consuming process for device manufacturers, who prioritize their newest devices and often leave older ones behind. Hence, older smartphones are frequently not updated if the manufacturer decides it is not worth their time, regardless of whether the phone is capable of running the update. This problem is compounded when manufacturers customize Android with their own interface and apps, which must be reapplied to each new release. Additional delays can be introduced by wireless carriers who, after receiving updates from manufacturers, further customize and brand Android to their needs and conduct extensive testing on their networks before sending the update out to users.

The lack of after-sale support from manufacturers and carriers has been widely criticized by consumer groups and the technology media. Some commentators have noted that the industry has a financial incentive not to update their devices, as the lack of updates for existing devices fuels the purchase of newer ones, an attitude described as "insulting". The Guardian has complained that the complicated method of distribution for updates is only complicated because manufacturers and carriers have designed it that way.

In 2011, Google partnered with a number of industry players to announce an "Android Update Alliance", pledging to deliver timely updates for every device for 18 months after its release; however, this alliance has never been mentioned since. Google then began updating many of its services, including Google Maps and Google Play Music, as applications independent from Android itself, and also introduced a system-level component providing APIs for Google services — installed automatically and updated directly by Google, and supportive of nearly all devices running Android version 2.2 and higher — through Google Play Services.

### Android Architecture



### Security and Privacy

Android applications run in a sandrbox, an isolated area of the system that does not have access to the rest of the system's resources, unless access permissions are explicitly granted by the user when the application is installed. Before installing an application, the Prlay Store displays all required permissions: a game may need to enable vibration or save data to an SD card, for example, but should not need to read SMS messages or access the phonebook. After reviewing these permissions, the user can choose to accept or refuse them, installing the application only if they accept. The sandboxing and permissions system lessens the impact of vulnerabilities and bugs in applications, but developer confusion and limited documentation has resulted in applications routinely requesting unnecessary permissions, reducing its effectiveness. Several security firms, such as Lookout Mobile Security, AVG Technologies, and McAfee, have released antivirus software for Android devices. This software is ineffective as sandboxing also applies to such applications, limiting their ability to scan the deeper system for threats.

Research from security company Trend Micro lists premium service abuse as the most common type of Android malware, where text messages are sent from infected phones to premium-rate telephone numbers without the consent or even knowledge of the user. Other malware displays unwanted and intrusive adverts on the device, or sends personal information to unauthorised third parties. Security threats on Android are reportedly growing exponentially; however, Google engineers have argued that the malware and virus threat on Android is being exaggerated by security companies for commercial reasons, and have accused the security industry of playing on fears to sell virus protection software to users. Google maintains that dangerous malware is actually extremely rare, and a survey conducted by F-Secure showed that only 0.5% of Android malware reported had come from the Google Play store.

Google currently uses their Google Bouncer malware scanner to watch over and scan the Google Play store apps. It is intended to flag up suspicious apps and warn users of any potential issues with an application before they download it. Android version 4.2 Jelly Bean was released in 2012 with enhanced security features, including a malware scanner built into the system, which works in combination with Google Play but can scan apps installed from third party sources as well, and an alert system which notifies the user when an app tries to send a premium-rate text message, blocking the message unless the user explicitly authorises it.

Android smartphones have the ability to report the location of Wi-Fi access points, encountered as phone users move around, to build databases containing the physical locations of hundreds of millions of such access points. These databases form electronic maps to locate smartphones, allowing them to run apps like Foursquare, Google Latitude, Facebook Places, and to deliver location-based ads. Third party monitoring software such as TaintDroid, an academic research-funded proHCETt, can, in some cases, detect when personal information is being sent from applications to remote servers. In August 2013, Google released the Android Device Manager, a component that allows users to remotely track, locate, and wipe their Android device through an online interface. As it is implemented through Google Play Services instead of within Android itself, it is available to most Android devices with version 2.2 and higher.

The open-source nature of Android allows security contractors to take existing devices and adapt them for highly secure uses. For example Samsung has worked with General Dynamics through their Open Kernel Labs acquisition to rebuild Jelly Bean on top of their hardened microvisor for the "Knox" proHCETt.

As part of the broader 2013 mass surveillance disclosures it was revealed in September 2013 that the American and British intelligence agencies, the NSA and Government Communications Headquarters (GCHQ) respectively, have access to the user data in iPhones, BlackBerrys, and Android phones. They are able to read almost all smartphone information, including SMS, location, emails, and notes.

## Netbeans

All the tools software developers need to create cross-platform Java desktop, enterprise and web applications. Runs on Windows, Linux, MacOS, as well as Solaris. It is easy to install and use, works right out of the box -- and it is open-source and free.



*Features of the NetBeans (Rich Client) Platform*

With the NetBeans Platform, developers get to concentrate on the important parts of an application - the business logic that makes that application unique. The result is a huge savings in time and effort. Some of the features of the platform are:

* ***User interface management***: Windows, menus, toolbars and other presentation components are provided by the Platform. Developers write actions and components that the system will manage - saving time, and producing cleaner, more bug-free code.
* ***Data and presentation management***: The NetBeans Platform contains a rich toolset for presentating data to the user and manipulating that data.
* ***Setting management***: Saving and restoring settings - even complex business obHCETts - is safe, simple, transparent and often automatic
* ***Graphical Editing***: Creating drag and drop, graphical views of data is a snap with the graph library
* ***The Editor***: Available as an extension to the Platform, applications built on NetBeans can use the NetBeans Editor, a powerful and extensible toolset for building custom editors.
* ***The Wizard framework***: A toolset for easily building extensible, user-friendly Wizards to guide users through more complex tasks.
* ***Storage management***: An abstraction of file-based data access. "Files" in the NetBeans paradigm may be local files, or exist remotely, for example, on an FTP server, CVS repository, in an XML file or in a database. Where this data is stored is completely transparent to other modules that work with this data.
* ***A huge selection of additional components***: such as versioning support, specialized editors, specialized UI components, remote data access via FTP and other transports, and convenient ways to work with a variety of Java and internet technologies are available as plugins any application can use.
* ***Internet-based update delivery***: NetBeans-based applications can use Java Web Start technology to deliver custom sets of modules based on a user's role, for complex applications. And a Web Start enabled application is always up-to-date and combines the advantages of centralized management and deployment with the advantages of a rich client user experience. For non-WebStart applications, an optional component is the AutoUpdate module, which downloads updates or new functionality via the web.

*Building on the NetBeans Platform*

Developing applications on top of the NetBeans Platform means you are developing on top of the NetBeans IDE's core. The NetBeans Platform is a generic desktop application, and most desktop applications have common requirements - menus, document management, settings and so forth. Instead of writing the same code over and over again, write modules to implement what you need, bundle them up with the NetBeans Platform, and you have a beautiful, branded, cross-platform application.

*Consistency*

NetBeans-based applications are write-once, run-anywhere. You get prebuilt components for free and you solve common problems by reusing, mixing and matching them. Use the platform and the modules you develop as the basis for multiple applications that share common logic. The NetBeans Platform is a solid foundation and set of standards for clear design.

*Modularity*

Applications based on the NetBeans Platform can install modules dynamically, so users no longer need to download the entire application to get an upgrade or a new release. You can even assemble an application from already existing modules and benefit from the open-source work already done by others. There are lots of useful modules written by the NetBeans community (tasklist, spellchecker, etc.) that are ready to be embedded.

# 

# Conclusion

The project “Helping System Android App” which has been developed using Netbeans and Android as a front-end tool meets the requirement of the organization. The problem of the women safety is increased rapidly in this environment, so I proposed as an effective Android application to prevent such type of the suspicious or natural disaster, by alerting the concern authorities using the android mobile phone which helps to stop such type of illegal activates and to trace the concern.

Developing this software has been a good experience for me. During the development of the project I had got enough to learn and got the chance to increase my knowledge in the field of software. I am satisfied that the system meets all the requirements.

# References

1. Holzuer Steven Beginners for android

DreamTech Publication

1. Grundgeiger Dave Programming Netbeans

O’reilly Publication

1. Mc. Graw hill Software engineering and design

Pressman

1. Galgotia Android Programming

Korth Publication