**CHAPTER-I**

**INTRODUCTION**

Smart phones provide a great choice of features that facilitate the life for the users. The features and capabilities of mobiles are increasing every day and almost everyone might have experienced misplacing or losing their mobile phones. The existing system provides some features in helping the user to trace the android phones by sending messages and mails. But, in real time every individual own more than one SIM card and hence the users find it annoying when the application sends notification messages for the SIM card changes made by them(the real owners). The proposed intelligent android Anti-theft application provides the safe and alert mode options (to avoid unwanted notifications). It is embedded with a lot of features such as SIM tracking based on location, detecting change in SIM card, application enabling at the time message sending and also delete important data from mobile phone. Thus proves to be different from the existing mobile tracker applications.

The mobile cellular communication has been appreciated since its birth in the early 70’s and the advancement in the field of VLSI has helped in designing less power, smaller size but efficient transceiver for the purpose of communication. But however the technology has not yet answered the loss or misplacement of the lost mobile phone which is significantly increasing. The mobile phone is lost there is possibilities for misuse of data stored in it. As there is no mechanism in place which can help the owner to recover his mobile phone. Mobile tracker which is autonomous and intimates with the owner via SMS and Email when it detects SIM change. Geo code technique is used for detection of the theft mobile and it can be done in efficient manner. The IMEI number is a unique number that is embedded in the mobile phone.

While the Apple iPhone single handedly redefined the term “smart phone” during its first two years of release, Google's Android platform for mobile devices has quickly developed into a serious open source alternative. We explored the Android Operating System (OS) and software development environment and evaluated several of its capabilities by constructing a working application. This application collected speed and location information from the Global Positioning System (GPS) receiver, used the Google Maps Application Programming Interface (API) to determine the location of nearby schools, and sounded an alarm if a person drove over the speed limit in a school zone. The platform proved capable of supporting a melding of different services, and we believe such smart phones have broad applicability to public safety problems.

The technology would weave into the fabric of everyday life until they are indistinguishable from it, “this is the his vision and has come true in this current21stcentury. Following the vision of Mark Weiser, researchers at Carnegie Mellon University in a project Aura demonstrated that, human distractions could be minimized by Ubiquitous systems which will have to be proactive in anticipating the demands, thus self-tunable so as to give a better response to the future hassles. The ability to sense and process a context forms as a fundamental requirement for a system to be proactive. Hence, designing a context -aware application would pave way to satisfy the future growing requirements of user. Most promising type of contextual information is the proximity selection known as Location Based Service (LBS). Tracking location of a mobile device accurately has been a challenging research topic for decades. Global Positioning Systems (GPS) is a top priority technology used for locating adevice position accurately. Methodology for tracking can be done using a GPS receiver which is an additional hardware integrated in most of mobile equipment. We have used GPS as the approach idea for location tracking. The platform used for development is Android Operating System, as described is been proven as the best operating system for a context-aware location based services. Customer friendly user interface letting user to enter the task and store it for future retrieval is done using the exclusive SQLite inbuilt database available in Android mobile. User can align task associated with any location and retrieving details as alert before reaching a desired location of interest. User entering into this application is given an option for connecting to the database so as to verify the location updates. Information is then delivered at the right time in the right place to the right person. The mobile user will also be able to receive retail offers and discount information in the surrounding by this intelligent observer module.

**Visual impairment**, also known as **vision impairment** or **vision loss**, is a decreased ability to [see](https://en.wikipedia.org/wiki/Visual_perception" \o "Visual perception) to a degree that causes problems not fixable by usual means, such as [glasses](https://en.wikipedia.org/wiki/Glasses" \o "Glasses). Some also include those who have a decreased ability to see because they do not have access to glasses or [contact lenses](https://en.wikipedia.org/wiki/Contact_lens" \o "Contact lens). Visual impairment is often defined as a best corrected [visual acuity](https://en.wikipedia.org/wiki/Visual_acuity" \o "Visual acuity) of worse than either 20/40 or 20/60. The term **blindness** is used for complete or nearly complete vision loss. Visual impairment may cause people difficulties with normal daily activities such as driving, reading, socializing, and walking.

The most common causes of visual impairment globally are uncorrected [refractive errors](https://en.wikipedia.org/wiki/Refractive_errors" \o "Refractive errors) (43%), [cataracts](https://en.wikipedia.org/wiki/Cataract" \o "Cataract) (33%), and [glaucoma](https://en.wikipedia.org/wiki/Glaucoma" \o "Glaucoma) (2%). Refractive errors include [near sighted](https://en.wikipedia.org/wiki/Near_sighted" \o "Near sighted), [far sighted](https://en.wikipedia.org/wiki/Far_sighted" \o "Far sighted), [presbyopia](https://en.wikipedia.org/wiki/Presbyopia" \o "Presbyopia), and [astigmatism](https://en.wikipedia.org/wiki/Astigmatism_(eye)" \o "Astigmatism (eye)).Cataracts are the most common cause of blindness.Other disorders that may cause visual problems include [age related macular degeneration](https://en.wikipedia.org/wiki/Age_related_macular_degeneration" \o "Age related macular degeneration), [diabetic retinopathy](https://en.wikipedia.org/wiki/Diabetic_retinopathy" \o "Diabetic retinopathy), [corneal clouding](https://en.wikipedia.org/wiki/Corneal_opacification" \o "Corneal opacification), [childhood blindness](https://en.wikipedia.org/wiki/Childhood_blindness" \o "Childhood blindness), and a number of [infections](https://en.wikipedia.org/wiki/Infection" \o "Infection).Visual impairment can also be caused by problems in the [brain](https://en.wikipedia.org/wiki/Brain" \o "Brain) due to [stroke](https://en.wikipedia.org/wiki/Stroke" \o "Stroke), [premature birth](https://en.wikipedia.org/wiki/Premature_birth" \o "Premature birth), or trauma among others. These cases are known as [cortical visual impairment](https://en.wikipedia.org/wiki/Cortical_visual_impairment" \o "Cortical visual impairment). Screening for vision problems in children may improve future vision and educational achievement. Screening adults without symptoms is of uncertain benefit. Diagnosis is by an [eye exam](https://en.wikipedia.org/wiki/Eye_exam" \o "Eye exam).

As of 2015 there were 940 million people with some degree of vision loss. 246 million had low vision and 39 million were blind. The majority of people with poor vision are in the [developing world](https://en.wikipedia.org/wiki/Developing_world" \o "Developing world) and are over the age of 50 years. Rates of visual impairment have decreased since the 1990s. Visual impairments have considerable economic costs both directly due to the cost of treatment and indirectly due to decreased ability to work.

Smart-phones have become ubiquitous as newer, less expensive models with greater feature sets have been released. While much attention has been given to the danger posed by drivers distracted by talking on the phone or sending text messages, responsible application of smart-phone technology could have a net positive effect on public safety. In our exploration of smart-phone technology, we developed a proof-of-concept system that addressed traffic safety in school zones. Our system addresses the need for drivers to be able to pay full visual attention to the road while still being alerted to the speed of the car. The system integrated several of the features that are becoming more commonplace on smart-phones as well as information retrieved from Internet services. The rest of the paper is structured as follows. We review the relevant technology. In, we discuss a proof-of-concept system to increase public safety and its implementation. The conclusion including a discussion of our future direction.

**A. GPS Technology**

The Global Positioning System (GPS) is a global navigation satellite system deployed by the US Department of Defence and maintained by the US Air Force. GPS is a spacebased radio navigation system that provides accurate location and timing services to anyone with a GPS receiver. This service, made available to civilians in 1996 for navigation purposes, is free of charge, can support an unlimited number of users, and functions anywhere in the world. Starting in 2004, the mobile phone industry began successful tests to incorporate GPS receivers into mobile phone devices to support 911 emergency location. Most of today’s smartphones are equipped with fully functional GPS receivers and supporting applications.

**B. Android Software**

The Apple iPhone has transformed the smartphone’s image from a corporate-level personal organizer to a device that could potentially benefit every consumer. Recently, Google released an alternative Operating System (OS) and Application Programming Interface (API) for mobile phones called Android. Android joins iPhone OS and other smartphone platforms including Symbian OS, Blackberry, and Windows Mobile. Android is backed by the Open Handset Alliance (OHA), whose members include Sony, Samsung, Motorola, and Nvidia. Though these companies are relatively new to the consumer-level smartphone market, they have already shown that by melding multiple technologies together in an open manner some unique applications can result.

On a basic level, Android is a distribution of Linux that includes a Java Virtual Machine (JVM), with Java being the preferred programming language for most Android applications. The Android Software Development Kit (SDK) includes a debugger, libraries, a handset emulator, documentation, sample code, and tutorials. Android’s official integrated development environment is Eclipse using the Android Development Tools (ADT) plug-in. SQLite database support is integrated into the Android platform. The ADT plugin includes an Android emulator that allows for the simulation of GPS and Wi-Fi. The Android emulator is depicted in displaying the Android desktop.

Previously, mobile phone OSs have been proprietary, leading applications to be tied to a specific carrier and phone. The iPhone, for instance, has restrictive licensing terms that allow only applications approved by Apple to be distributed publicly. Android’s open nature is intended to set a new standard for mobile phone OSs. Applications can be written once and then run on a variety of phones and carriers. As such, Android is available as open source software under the Apache License.

As an open source platform, one of the goals of Android is to enable developers to create applications that utilize the features the mobile device has to offer and to tailor its features to the needs of the consumer. Android allows for the combination of information from the web with core features of the phone such as the camera function and text messaging.



**C. Android Capable Hardware**

T-Mobile released the G1 phone, running the Android platform, in October 2008. Subsequently, Google began selling an unlocked version of the G1 as the Android Dev Phone 1 to developers. The Android Dev Phone 1 is not limited to the use of T-Mobile SIM cards and applications may exploit some features that are limited in the consumer version of the phone. The G1 has a dual core ARM11 processor running at 528 MHz and 192 MB RAM. It

features a touch screen, QWERTY keyboard, trackball, GPS receiver, Wi-Fi, accelerometer, compass, 3G, and Bluetooth.



* 1. **PROBLEM DEFINITION**

The [World Health Organization](https://en.wikipedia.org/wiki/World_Health_Organization" \o "World Health Organization) (WHO) estimates that 80% of visual impairment is either preventable or curable with treatment. This includes cataracts, the infections [river blindness](https://en.wikipedia.org/wiki/Onchocerciasis" \o "Onchocerciasis) and [trachoma](https://en.wikipedia.org/wiki/Trachoma" \o "Trachoma), glaucoma, diabetic retinopathy, uncorrected refractive errors, and some cases of childhood blindness. Many people with significant visual impairment benefit from [vision rehabilitation](https://en.wikipedia.org/wiki/Vision_rehabilitation" \o "Vision rehabilitation), changes in their environment, and assistive devices. . The system integrated several of the features that are becoming more commonplace on smart-phones as well as information retrieved from Internet services.

**CHAPTER II**

**2. SYSTEM ANALYSIS**

**2.1 EXISTING SYSTEM:**

The traditional and oldest mobility aids for persons with visual impairments are he walking cane (also called white cane or stick) and guide dogs. The most important drawbacks of these aids are necessary skills and training phase, range of motion and very little information conveyed.

**DISADVANTAGES**

People with visual disabilities are often dependent on external assistance which can be provided by humans, trained dogs. Existing devices are able to detect and recognize objects that emerge on the floor, but a considerable risk is also includes the objects that are at a sudden depth, or obstacles above waist level or stair.

**2.2 PROPOSED SYSTEM**

This blind App used to help the blind navigate independently and safely.Many blind guidance systems use this app because of its immunity to the environmental noise.

Thus we were motivated to develop a Blind App to overcome these limitations

**ADVANTAGES**

No need stick.

Give Self guidance for the blind people

**Feasibility study**

**2.3.1 Functional Feasibility**

The proposed/developed system is a web application and does not require any additional software requirements except hosting. A web hosting company hosts the web application in their web servers. Any client (end-user) with an internet connection and a web browsing software (at client side) can immediately start accessing/using the system. So, the system does not need any installation/setup procedure. Hence, the system is functionally feasible.

**2.3.2 Economic Feasibility**

Author (project stakeholder) does not need to purchase any software and hardware to host the developed system. Web hosting companies purchase and maintain all necessary hardware and software for hosting the websites. Author has to pay only hosting charges to the hosting company. When it comes to the end-user, he/she does not need to purchase any software to use the website. Now-a-days every computer’s operating system is providing built in web browser and so many web browsing software products are available in the market for free.Only the cost that the author should bare is cost of development. Before starting the development, the development team estimates cost of development depends on the features asked by the client. If any new features to be added as per the new requirements of the client, cost of new additions are submitted to the client and after getting the confirmation from the client the new features will be added by the development team. Development team will take care of in-time delivery of the project to avoid excess cost of development due to delay.Hence, the proposed system is economically feasible.

**2.3.3 Technical feasibility**

This is about acceptance of the new system by the existing end-users and employees of the system and author. The website is aiming at employees and now-a-days all of them are aware of web browsing. If any person does not have any idea of web browsing, he/she will be trained to use the system with in one hour time. So, the system can easily be accepted by any kind of end-user. Hence the proposed system is technically feasible.

**CHAPTER-III**

**DEVELOPMENT ENVIRONMENT**

**3.1 HARDWARE REQUIREMENTS**

* Processor : Dual core processor 2.6.0 GHZ
* RAM : 1GB
* Hard disk : 160 GB
* Compact Disk : 650 Mb
* Keyboard : Standard keyboard
* Monitor : 15 inch color monitor

**3.2 SOFTWARE REQUIREMENTS**

* Operating system : Android OS
* Front End : ANDROID SDK (JAVA)
* IDE : android studio

**SOFTWARE SPECIFICATION**

**Android**

Android is a mobile operating system developed by Google, based on a modified version of the Linux kernel and other open source software and designed primarily for touch screen mobile devices such as smart phones and tablets.

Android is a mobile operating system developed by Google, based on a modified version of the Linux kernel and other open source software and designed primarily for touch screen mobile devices such as smart phones and tablets. In addition, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on game consoles, digital cameras, PCs and other electronics.

Initially developed by Android Inc., which Google bought in 2005, Android was unveiled in 2007, with the first commercial Android device launched in September 2008. The operating system has since gone through multiple major releases, with the current version being 8.1 "Oreo", released in December 2017.

Android has been the best-selling OS worldwide on smartphones since 2011 and on tablets since 2013. As of May 2017, it has over two billion monthly active users, the largest installed base of any operating system, and as of 2017, the Google Play store features over 3.5 million apps.

Applications:

Applications ("apps"), which extend the functionality of devices, are written using the Android software development kit (SDK)[77] and, often, the Java programming language.[78] Java may be combined with C/C++,[79] together with a choice of non-default runtimes that allow better C++ support.[80] The Go programming language is also supported, although with a limited set of application programming interfaces (API).[81] In May 2017, Google announced support for Android app development in the Kotlin programming language.[82][83]

The SDK includes a comprehensive set of development tools,[84] including a debugger, software libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Initially, Google's supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) plugin; in December 2014, Google released Android Studio, based on IntelliJ IDEA, as its primary IDE for Android application development. Other development tools are available, including a native development kit (NDK) for applications or extensions in C or C++, Google App Inventor, a visual environment for novice programmers, and various cross platform mobile web applications frameworks. In January 2014, Google unveiled an framework based on Apache Cordova for porting Chrome HTML 5 web applications to Android, wrapped in a native application shell.[85]

Android has a growing selection of third-party applications, which can be acquired by users by downloading and installing the application's APK (Android application package) file, or by downloading them using an application store program that allows users to install, update, and remove applications from their devices. Google Play Store is the primary application store installed on Android devices that comply with Google's compatibility requirements and license the Google Mobile Services software.[86][87] Google Play Store allows users to browse, download and update applications published by Google and third-party developers; as of July 2013, there are more than one million applications available for Android in Play Store.[88] As of July 2013, 50 billion applications have been installed.[89][90] 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.[91] As of May 2017, there are over one billion active users a month for Gmail, Android, Chrome, Google Play and Maps.

What is an APK File?

An Android Package Kit (APK for short) is the package file format used by the Android operating system for distribution and installation of mobile apps. Just like Windows (PC) systems use an .exe file for installing software, Android does the same.

Why would I want to install an APK File?

APK files of new Android builds are often leaked ahead of time, giving you early access to all the exciting new features that otherwise wouldn't be available to you. Also, some apps are restricted to regions outside of yours.

**How to install incompatible apps or region-restricted apps on Android**

App updates can take a a frustrating amount of time to arrive automatically over the air. Getting the latest APK for your favorite apps will let you jump the queue. Or you might even not like a particular app after an update, and want to install an older version instead.And if your Android device lacks access to the Google Play Store, APK files may be your only option for installing apps on your device. If you've accidentally deleted the Google Play Store, here's how to get the APK and restore itBe careful though, you don't want to download stolen apps. Some APK services let you download pirated apps. This is illegal and should be avoided. Do some research before you download so you don't do anything against the law.

**How to install APK Files from your computer**

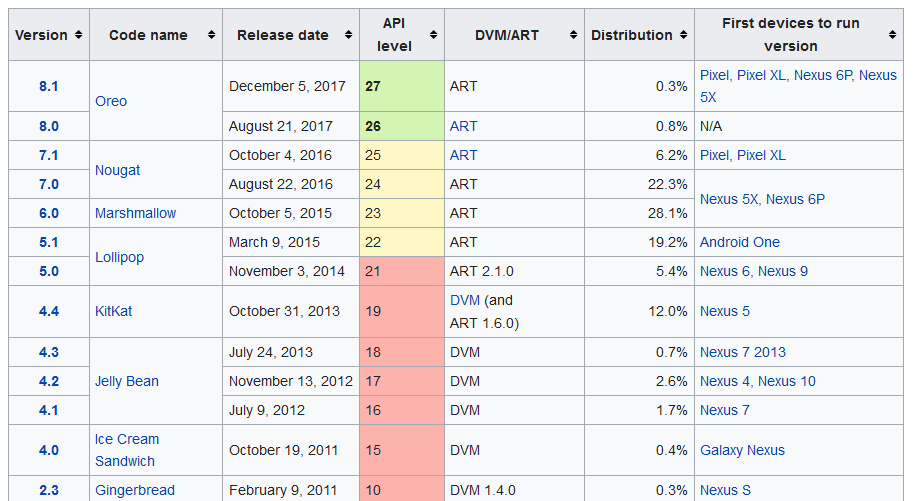
There are many sources of APK files across the web, but be sure to choose a site you trust. Some APK files may contain malicious software (malware) which could compromise your phone's security. It is best to be cautious before downloading or installing any app from outside of the Play Store.

Generally speaking, APK files found on APK Mirror should be safer to download than from elsewhere, but read some reviews and user comments before downloading or installing an APK.

**Best alternative Android app stores**

Firstly, you must find an APK file you want. Search online for an APK file and download it to an easy-to-find location on your computer (like your desktop). Before you can install it on your phone you will need to make sure that third-party apps are allowed on your device. Go to Menu > Settings > Security > and check Unknown Sources to allow your phone to install apps from sources other than the Google Play Store. Android 8.0 Oreo does things a little differently. Rather than check a global setting to allow installation from unknown sources, you'll be you'll be prompted to allow your browser or file manager to install APKs the first time you attempt to do so.

If you like, you can also download a file manager app so you can easily find files on your Android device. Use the link for Solid Explorer below or explore our favorite alternatives in our list of the best file explorers for Android.



**Eclipse for Android Developers**

**Package Description**

An IDE for developers creating Android applications.

**This package includes:**

* Eclipse Git Team Provider
* Eclipse Java Development Tools
* Maven Integration for Eclipse
* Mylyn Task List
* Code Recommenders Tools for Java Developers
* Eclipse XML Editors and Tools

Detailed features list.

**How to setup Android for Eclipse IDE**

In this page, you will learn what softwares are required for running an android application on eclipse IDE. Here, you will be able to learn how to install the android SDK and ADT plugin for Eclipse IDE. Let's see the list of software required to **setup android for eclipse** IDE manually.

1. Install the JDK
2. Download and install the Eclipse for developing android application
3. Download and Install the android SDK
4. Intall the ADT plugin for eclipse
5. Configure the ADT plugin
6. Create the AVD
7. Create the hello android application

**1) Install the Java Development Kit (JDK)**

For creating android application, JDK must be installed if you are developing the android application with Java language. [download the JDK](http://www.oracle.com/technetwork/java/javase/downloads/index.html" \t "_blank)

**2) Download and install the Eclipse IDE**

For developing the android application using eclipse IDE, you need to install the Eclipse. you can download it from this location [download the Eclipse](http://www.eclipse.org/downloads/" \t "_blank). Eclipse classic version is recommended but we are using the Eclipse IDE for JavaEE Developers.

**3) Download and install the android SDK**

First of all, [download the android SDK](http://developer.android.com/sdk/index.html" \t "_blank). In this example we have installed the android SDK for windows (.exe version).

Now double click on the exe file, it will be installed. I am using the android 2.2 version here.

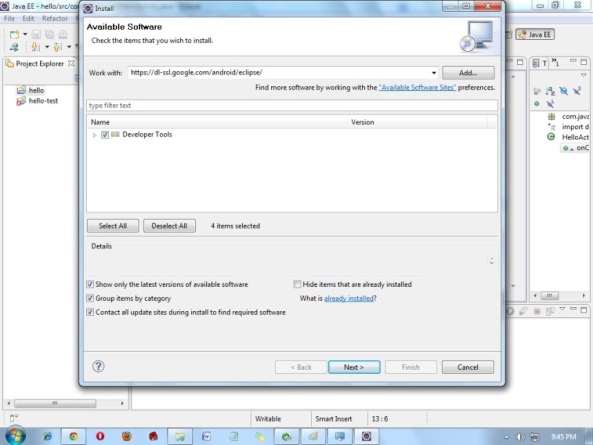
**4) Download the ADT plugin for eclipse**

ADT (Android Development Tools) is required for developing the android application in the eclipse IDE. It is the plugin for Eclipse IDE that is designed to provide the integrated environment.

For downloading the ADT, you need to follow these steps:

1) Start the eclipse IDE, then select **Help** > **Install new software...**

2) In the **work with** combo box, write **https://dl-ssl.google.com/android/eclipse/**



3) **select the checkbox** next to Developer Tools and **click next**

4) You will see, a list of tools to be downloaded here, **click next**

5) **click finish**

6) After completing the installation, restart the eclipse IDE

**5) Configuring the ADT plugin**

After the installing ADT plugin, now tell the eclipse IDE for your android SDK location. To do so:

1. Select the **Window menu > preferences**
2. Now select the android from the left panel. Here you may see a dialog box asking if you want to send the statistics to the google. Click **proceed**.
3. Click on the browse button and locate your SDK directory e.g. my SDK location is C:\Program Files\Android\android-sdk .
4. Click the apply button then OK.

**6) Create an Android Virtual Device (AVD)**

For running the android application in the Android Emulator, you need to create and AVD. For creating the AVD:

1. Select the **Window menu > AVD Manager**
2. Click on the **new** button, to create the AVD
3. Now a dialog appears, write the AVD name e.g. myavd. Now choose the target android version e.g. android2.2.
4. click the **create AVD**

**7) create and run the simple android example**

# How to make android apps

In this page, you will know how to create the simple hello android application. We are creating the simple example of android using the Eclipse IDE. For creating the simple example:

1. Create the new android project
2. Write the message (optional)
3. Run the android application

### Hello Android Example

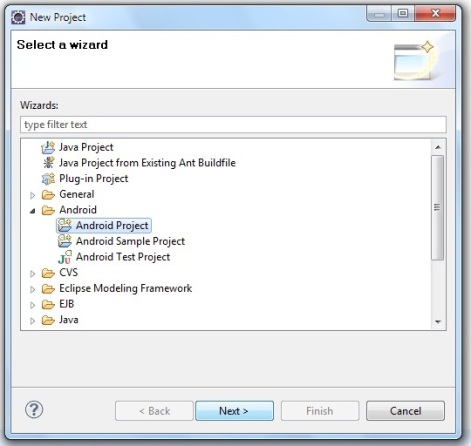
You need to follow the 3 steps mentioned above for creating the Hello android application.

### 1) Create the New Android project

For creating the new android project:

1) Select **File > New > Project...**

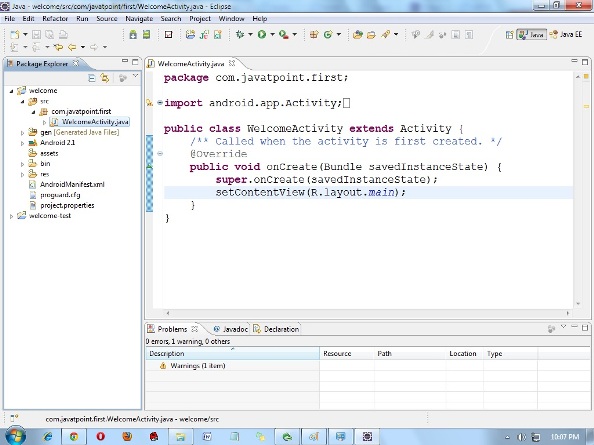
2) Select the android project and click **next**



3) Fill the Details in this dialog box and click **finish**

hello android example

Now an android project have been created. You can explore the android project and see the simple program, it looks like this:



### 2) Write the message

For writing the message we are using the TextView class. Change the onCreate method as:

1. TextView textview=new TextView(this);
2. textview.setText("Hello Android!");
3. setContentView(textview);

Android (stylized as android) is a mobile operating system developed by Google, based on the Linux kernel and designed primarily for touch screen mobile devices such as smart phones and tablets. Android's user interface is mainly based on direct manipulation, using touch gestures that loosely correspond to real-world actions, such as swiping, tapping and pinching, to manipulate on-screen objects, along with a virtual keyboard for text input. In addition to touch screen devices, Google has further developed Android TV for televisions, Android Auto for cars, and Android Wear for wrist watches, each with a specialized user interface. Variants of Android are also used on notebooks, game consoles, digital cameras, and other electronics.

Android has the largest installed base of all operating systems (OS) of any kind. Android has been the best selling OS on tablets since 2013, and on smart phones it is dominant by any metric.

Initially developed by Android, Inc., which Google 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. As of July 2013, the Google Play store has had over one million Android applications ("apps") published – including many "business-class apps" that rival competing mobile platforms – and as of May 2016 over 65 billion applications downloaded. An April–May 2013 survey of mobile application developers found that 71% of developers create applications for Android, and a 2015 survey found that 40% of full-time professional developers see Android as their priority target platform, which is comparable to Apple's iOS on 37% with both platforms far above others. In September 2015, Android had 1.4 billion monthly active devices.

Android's source code is released by Google under an open source license, although most Android devices ultimately ship with a combination of free and open source and proprietary software, including proprietary software required for accessing Google services. Android is popular with technology companies that require a ready-made, low-cost and customizable operating system for high-tech devices. Its open nature has encouraged a large community of developers and enthusiasts to use the open-source code as a foundation for community-driven projects, which deliver updates to older devices, add new features for advanced users or bring Android to devices originally shipped with other operating systems. The success of Android has made it a target for patent (and copyright) litigation as part of the so-called "Smartphone wars" between technology companies.

**Features**

Android's default user interface is mainly 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 objects, along with a virtual keyboard. Game controllers and full-size physical keyboards are supported via Bluetooth or USB. 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 hap tic 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 home screen, the primary navigation and information "hub" on Android devices that is analogous to the desktop found on personal computers. (Android also runs on regular personal computers, as described below). Android home screens 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 home screen. A home screen may be made up of several pages, between which the user can swipe back and forth, though Android's home screen interface is heavily customizable, allowing users to adjust the look and feel of the devices to their tastes. Third-party apps available on Google Play and other app stores can extensively re-theme the home screen, and even mimic the look of other operating systems, such as Windows Phone. Most manufacturers, and some wireless carriers, customize the look and feel of their Android devices to differentiate themselves from their competitors. Applications that handle interactions with the home screen are called "launchers" because they, among other purposes, launch the applications installed on a device.

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. Notifications are persistent until read by tapping it, which opens the relevant app, or dismissed by sliding it off the screen. Beginning on Android 4.1, "expanded notifications" can display expanded details or additional functionality; for instance, a music player can display playback controls, and a "missed call" notification provides buttons for calling back or sending the caller an SMS message.

Android provides the ability to run applications that change the default launcher, and hence the appearance and externally visible behavior of Android. These appearance changes include a multi-page dock or no dock, and many more changes to fundamental features of the user interface.

**Application**

Applications ("apps"), which extend the functionality of devices, are written using the Android software development kit (SDK) and, often, the Java programming language, which has complete access to the Android APIs. Java may be combined with C/C++, together with a choice of non-default runtimes that allow better C++ support; the Go programming language is also supported since its version 1.4, which can also be used exclusively although with a restricted set of Android APIs. The SDK includes a comprehensive set of development tools, including a debugger, software libraries, a handset emulator based on QEMU, documentation, sample code, and tutorials. Initially, Google's supported integrated development environment (IDE) was Eclipse using the Android Development Tools (ADT) plugin; in December 2014, Google released Android Studio, based on IntelliJ IDEA, as its primary IDE for Android application development. Other development tools are available, including a native development kit (NDK) for applications or extensions in C or C++, Google App Inventor, a visual environment for novice programmers, and various cross platform mobile web applications frameworks. In January 2014, Google unveiled an framework based on Apache Cordova for porting Chrome HTML 5 web applications to Android, wrapped in a native application shell.

Android has a growing selection of third-party applications, which can be acquired by users by downloading and installing the application's APK (Android application package) file, or by downloading them using an application store program that allows users to install, update, and remove applications from their devices. Google Play Store is the primary application store installed on Android devices that comply with Google's compatibility requirements and license the Google Mobile Services software. Google Play Store allows users to browse, download and update applications published by Google and third-party developers; as of July 2013, there are more than one million applications available for Android in Play Store. As of July 2013, 50 billion applications have been installed. 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.

Due to the open nature of Android, a number of third-party application marketplaces also exist for Android, either to provide a substitute for devices that are not allowed to ship with Google Play Store, provide applications that cannot be offered on Google Play Store due to policy violations, or for other reasons. Examples of these third-party stores have included the Amazon Appstore, GetJar, and SlideMe. F-Droid, another alternative marketplace, seeks to only provide applications that are distributed under free and open source licenses.

**Memory management**

Since Android devices are usually battery-powered, Android is designed to manage processes to keep power consumption at a minimum. When an application is not in use the system suspends its operation so that, while available for immediate use rather than closed, it does not use battery power or CPU resosurces.

Android manages the applications stored in memory automatically: when memory is low, the system will begin invisibly and automatically closing inactive processes, starting with those that have been inactive for longest. Lifehacker reported in 2011 that third-party task killers were doing more harm than good.

**Virtual reality**

At Google I/O on May 2016, Google announced Daydream, a virtual reality platform that relies on a smartphone and provides VR capabilities through a virtual reality headset and controller designed by Google itself. The platform is built into Android starting waith Android Nougat, differentiating from standalone support for VR capabilities. The software is available for developers, and was released in 2016.

**About Android SDK (JAVA)**

For developing software application popularly termed as apps, developers around the world prefer the Android SDK and Java. SDK for Android is actually a Software Development Kit and the Android SDK provides API libraries as well as developer tools. Using these developer tools and the libraries, building, testing and debugging applications built for Android, can become really easy. Android SDK components are found in ADT or the Android Developer Tools.

Most ADT Bundle also has Eclipse and the ADT plugin, the Android SDK tools, and the Android platform tools. The bundle also has the latest Android platform and the latest Android system image for emulator. All these things are what are needed for developing apps for Android. Customised approach to installing Android SDK can be used for use in an existing version of ECLIPSE or any other IDE. Most Android SDK Development tools can be used in Windows OS. Tools for other OS are also available.

Our expert team of developers is comfortable with ECLIPSE as well as other IDE’s. They also are experts in SDK handling and even develop their own ADT plugins and tools to have more liberty in their work. They are highly imaginative and immensely talented and they design some of the most amazing apps for Android as well as cross platform use.

Android Java development is also extremely popular among developers for developing apps. Java applications can be similar as well as quite different from Java for Android apps. But familiarity with any programming language makes it easy to learn the tricks of Android Java development and churn out amazing and useful applications.

One of the important things needed for developing Android applications or any Java application, is a supporting environment to build and write applications. One of the most popular development environments is Eclipse for Java and Android development. It is free for Windows, Mac and Linux operating systems. There are some other IDEs also for the same purpose. Our developers use all of those IDEs to develop stunning applications for Android as well as for cross platform usage. Some of these apps are free and some are chargeable.

**About Android Eclipse**

Eclipse is an integrated development environment (IDE) used in computer programming, and is the most widely used Java IDE. It contains a base workspace and an extensible plug-in system for customizing the environment. Eclipse is written mostly in Java and its primary use is for developing Java applications, but it may also be used to develop applications in other programming languages through the use of plugins, including: Ada, ABAP, C, C++, COBOL, D, Fortran, Haskell, JavaScript, Julia, Lasso, Lua, NATURAL, Perl, PHP, Prolog, Python, R, Ruby (including Ruby on Rails framework), Rust, Scala, Clojure, Groovy, Scheme, and Erlang. It can also be used to develop documents with LaTeX (through the use of the TeXlipse plugin) and packages for the software Mathematica. Development environments include the Eclipse Java development tools (JDT) for Java and Scala, Eclipse CDT for C/C++ and Eclipse PDT for PHP, among others.

The initial codebase originated from IBM Visual Age. The Eclipse software development kit (SDK), which includes the Java development tools, is meant for Java developers. Users can extend its abilities by installing plug-ins written for the Eclipse Platform, such as development toolkits for other programming languages, and can write and contribute their own plug-in modules. Since Equinox, plug-ins can be plugged/stopped dynamically and are known as (OSGI) bundles.

Released under the terms of the Eclipse Public License, Eclipse SDK is free and open-source software, although it is incompatible with the GNU General Public License. It was one of the first IDEs to run under GNU Class path and it runs without problems under IcedTea.

**About Android SQLITE**

SQLite is a open source SQL database that stores data to a text file on a device. Android comes in with built in SQLite database implementation. SQLite supports all the relational database features. In order to access this database, you don't need to establish any kind of connections for it like JDBC, ODBC e.t.c. SQLite is an open-source relational database i.e. used to perform database operations on android devices such as storing, manipulating or retrieving persistent data from the database. It is embedded in android by default. So, there is no need to perform any database setup or administration task. Here, we are going to see the example of SQLite to store and fetch the data. Data is displayed in the logcat. For displaying data on the spinner or listview, move to the next page. SQLite OpenHelper class provides the functionality to use the SQLite database.

**4.MODULES DESCRIPTION**

**MODULES**

* HOME
* TTS
* IMAGE CAPTURE CAMERA
* DOCUMENT READ
* LOGOUT

**Home:**

All pages that a users see over internet or to be specific on browser is said to be web page. Home page is subset of web pages, the launching page of each site is called as home page of that website.A home page is generally the primary web page which a visitor navigating to a website from a search engine will see, and it may also serve as a landing page to attract visitors.Thus good home page design is usually a high priority for a website.

**Text-to-Speech (TTS)**

Voice synthesis, defined as TTS (acronym for Text-To-Speech), is a computer system that should be able to read aloud any text, regardless of its origin .

The use of TTS aims to produce human voice artificially. Voice synthesis is a complex process and complex

algorithms are needed to produce an intelligible and natural result. TTS synthesis makes use of techniques of Natural

Language Processing. Since the text to be synthesized is the first entry of the system, it must be the first to be processed.

There are several techniques to create a synthesized voice :

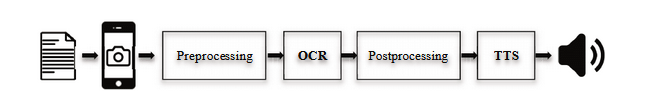
• Articulatory synthesis

• Formant synthesis

• Concatenation synthesis

• Hidden Markov models synthesis

The main synthesis techniques, presented above, are the methods used in the study and development of speech synthesis systems. However, a way to profit from the inherent advantages of each technique is to use a hybrid of the various techniques in the development of future systems speech synthesis.



**Image capture with the camera**

The camera of the mobile device is critical to use the application, since it will be essential for the user to take the picture of the brackets containing text that will be recognized and synthesized. The image capture is done inside the application itself, thus avoiding the use of additional applications such as access to the photo gallery.

**Document ,Currency read:**

Voice Reader is a voice-based mobile app that allows the blind or visually impaired to read anything that contains text, such as websites, local files on your device, Bookshare, and more.The technology of optical character recognition (OCR) enables the recognition of texts from image data. This technology has been widely used in scanned or photographed documents, converting them into electronic copies, which one can edit, search, play its content and easily carry.

**Logout:**

Loging out means to end access to a computer system or a website. Logging out informs the computer or website that the current user wishes to end the login session. Log out is also known as log off, sign off or sign out.

**5.SYSTEM DESIGN**

**UML DIAGRAM**

**USE CASE DIAGRAM**

A **use case diagram** at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different [use cases](https://en.wikipedia.org/wiki/Use_case" \o "Use case) in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well.

Mobile

user

TURN ON CAMERA

Read Document

Read text

**Read object**

Read currency

Logout

**Activity diagram:**

**Activity diagrams** are graphical representations of [workflows](https://en.wikipedia.org/wiki/Workflow" \o "Workflow) of stepwise activities and actions with support for choice, iteration and concurrency. In the [Unified Modeling Language](https://en.wikipedia.org/wiki/Unified_Modeling_Language" \o "Unified Modeling Language), activity diagrams are intended to model both computational and organizational processes (i.e., workflows), as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores.

On camerar

Back camera

Explore data

Read document

Read object

Read currency

logout

**Class diagram:**

The class diagram is the main building block of [object-oriented](https://en.wikipedia.org/wiki/Object-oriented_programming" \o "Object-oriented programming) modelling. It is used for general [conceptual modelling](https://en.wikipedia.org/wiki/Conceptual_model" \o "Conceptual model) of the systematic of the application, and for detailed modelling translating the models into [programming code](https://en.wikipedia.org/wiki/Programming_code" \o "Programming code). Class diagrams can also be used for [data modeling](https://en.wikipedia.org/wiki/Data_modeling" \o "Data modeling). The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed.

**User**

+User details

+Registration()

+Login()

+One time password()

mobile

+image details

+text details)

Blind app

+ voice comment

+read()

**ER diagram:**

Blind app

Register

user

Id

Uname

Rdate

data

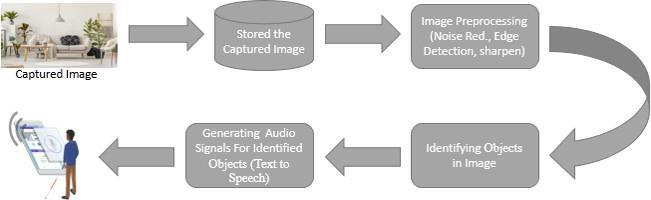
Location

mobile

Camera on

Read document

**ARCHITECTURE DIAGRAM:**



**TABLE DESIGN**

# Database

## Table : user

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** |
| username | varchar(30) | Yes | NULL |
| password | varchar(30) | Yes | NULL |

## Table : Sponsor

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** |
| username | varchar(30) | Yes | NULL |
| password | varchar(30) | Yes | NULL |

## Table : Third Party

|  |  |  |  |
| --- | --- | --- | --- |
| **Field** | **Type** | **Null** | **Default** |
| username | varchar(30) | Yes | NULL |
| password | varchar(30) | Yes | NULL |

**6. CODINGS**

**package** com.example.app;

**import** android.support.v7.app.ActionBarActivity;

**import** android.os.Bundle;

**import** android.view.Menu;

**import** android.view.MenuItem;

**public** **class** MainActivity **extends** ActionBarActivity {

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_main*);

}

@Override

**public** **boolean** onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.*main*, menu);

**return** **true**;

}

@Override

**public** **boolean** onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

**int** id = item.getItemId();

**if** (id == R.id.*action\_settings*) {

**return** **true**;

}

**return** **super**.onOptionsItemSelected(item);

}

}

**package** com.example.app;

**import** android.support.v7.app.ActionBarActivity;

**import** android.os.Bundle;

**import** android.view.Menu;

**import** android.view.MenuItem;

**public** **class** MainActivity **extends** ActionBarActivity {

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_main*);

}

@Override

**public** **boolean** onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.*main*, menu);

**return** **true**;

}

@Override

**public** **boolean** onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

**int** id = item.getItemId();

**if** (id == R.id.*action\_settings*) {

**return** **true**;

}

**return** **super**.onOptionsItemSelected(item);

}

}

**package** com.example.abu;

**import** android.support.v7.app.ActionBarActivity;

**import** android.os.Bundle;

**import** android.view.Menu;

**import** android.view.MenuItem;

**public** **class** Abu **extends** ActionBarActivity {

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_abu*);

}

@Override

**public** **boolean** onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.*abu*, menu);

**return** **true**;

}

@Override

**public** **boolean** onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

**int** id = item.getItemId();

**if** (id == R.id.*action\_settings*) {

**return** **true**;

}

**return** **super**.onOptionsItemSelected(item);

}

}

**package** com.example.abu;

**import** android.support.v7.app.ActionBarActivity;

**import** android.os.Bundle;

**import** android.view.Menu;

**import** android.view.MenuItem;

**public** **class** Abu **extends** ActionBarActivity {

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_abu*);

}

@Override

**public** **boolean** onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.*abu*, menu);

**return** **true**;

}

@Override

**public** **boolean** onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

**int** id = item.getItemId();

**if** (id == R.id.*action\_settings*) {

**return** **true**;

}

**return** **super**.onOptionsItemSelected(item);

}

}

**package** com.example.abu;

**import** android.support.v7.app.ActionBarActivity;

**import** android.os.Bundle;

**import** android.view.Menu;

**import** android.view.MenuItem;

**public** **class** Abu **extends** ActionBarActivity {

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_abu*);

}

@Override

**public** **boolean** onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.*abu*, menu);

**return** **true**;

}

@Override

**public** **boolean** onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

**int** id = item.getItemId();

**if** (id == R.id.*action\_settings*) {

**return** **true**;

}

**return** **super**.onOptionsItemSelected(item);

}

}

**package** com.example.abu;

**import** android.support.v7.app.ActionBarActivity;

**import** android.os.Bundle;

**import** android.view.Menu;

**import** android.view.MenuItem;

**public** **class** Abu **extends** ActionBarActivity {

@Override

**protected** **void** onCreate(Bundle savedInstanceState) {

**super**.onCreate(savedInstanceState);

setContentView(R.layout.*activity\_abu*);

}

@Override

**public** **boolean** onCreateOptionsMenu(Menu menu) {

// Inflate the menu; this adds items to the action bar if it is present.

getMenuInflater().inflate(R.menu.*abu*, menu);

**return** **true**;

}

@Override

**public** **boolean** onOptionsItemSelected(MenuItem item) {

// Handle action bar item clicks here. The action bar will

// automatically handle clicks on the Home/Up button, so long

// as you specify a parent activity in AndroidManifest.xml.

**int** id = item.getItemId();

**if** (id == R.id.*action\_settings*) {

**return** **true**;

}

**return** **super**.onOptionsItemSelected(item);

}

}

<?xml version="1.0" encoding="utf-8" ?>

**[-](file:///C:\\Users\\new%20tech\\Desktop\\spydroid\\Emergency\\AndroidManifest.xml)** <manifest xmlns:android="**http://schemas.android.com/apk/res/android**" package="**com.example.emergency**" android:versionCode="**1**" android:versionName="**1.0**">

  <uses-sdk android:minSdkVersion="**8**" android:targetSdkVersion="**17**" />

  <uses-permission android:name="**android.permission.INTERNET**" />

  <uses-permission android:name="**android.permission.ACCESS\_COARSE\_LOCATION**" />

  <uses-permission android:name="**android.permission.ACCESS\_FINE\_LOCATION**" />

  <uses-permission android:name="**android.permission.ACCESS\_MOCK\_LOCATION**" />

  <uses-permission android:name="**android.permission.ACCESS\_NETWORK\_STATE**" />

**[-](file:///C:\\Users\\new%20tech\\Desktop\\spydroid\\Emergency\\AndroidManifest.xml)** <application android:allowBackup="**true**" android:icon="**@drawable/ic\_launcher**" android:label="**@string/app\_name**" android:theme="**@style/AppTheme**">

**[-](file:///C:\\Users\\new%20tech\\Desktop\\spydroid\\Emergency\\AndroidManifest.xml)** <activity android:name="**.MainActivity**" android:label="**@string/app\_name**">

**[-](file:///C:\\Users\\new%20tech\\Desktop\\spydroid\\Emergency\\AndroidManifest.xml)** <intent-filter>

  <action android:name="**android.intent.action.MAIN**" />

  <category android:name="**android.intent.category.LAUNCHER**" />

  </intent-filter>

  </activity>

  <activity android:name="**.UserLogin**" android:label="**@string/title\_activity\_user\_login**" />

  <activity android:name="**.UserRegisteration**" android:label="**@string/title\_activity\_user\_registeration**" />

  <activity android:name="**.UserHome**" android:label="**@string/title\_activity\_user\_home**" />

  <activity android:name="**.AdminHome**" android:label="**@string/title\_activity\_admin\_home**" />

  <activity android:name="**.CALLWEB**" android:label="**@string/title\_activity\_callweb**" />

  <activity android:name="**.SendAlert**" android:label="**@string/title\_activity\_send\_alert**" />

  <activity android:name="**.AlertMessage**" android:label="**@string/title\_activity\_alert\_message**" />

  <action android:name="**android.intent.action.MAIN**" />

  <category android:name="**android.intent.category.LAUNCHER**" />

  </intent-filter>

  </activity>

  <activity android:name="**.AdminLogin**" android:label="**@string/title\_activity\_admin\_login**" />

  <activity android:name="**.UserLogin**" android:label="**@string/title\_activity\_user\_login**" />

  </application>

  </manifest>

**CHAPTER 7**

**TESTING**

**7.1 Testing Tool**

Tools from a software testing context can be defined as a product that supports one or more test activities right from planning, requirements, creating a build, test execution, defect logging and test analysis. Tools can be classified based on several parameters.

**7.2 Selenium Tool**

SELENIUM is a free (open-source) automated testing framework used to validate web applications across different browsers and platforms. You can use multiple programming languages like Java, C#, Python etc to create Selenium Test Scripts. Testing done using the Selenium tool is usually referred to as Selenium Testing.

**7.3 Testing Methodology**

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.

**7.3.1 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.

**7.3.2 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.

**7.3.3 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.

**CHAPTER – 8**

**CONCLUSION**

The Android platform proved to be capable of supporting a melding of different services. Our sample application showed how GPS data and Google search services could be combined to keep school children safe. Only one type of sensor and one online service was used. Many more novel applications are possible when taking into account Android’s extensive sensor capability and Internet access. The open nature of Android forms the foundation of a hitherto untapped reservoir of mobile applications.

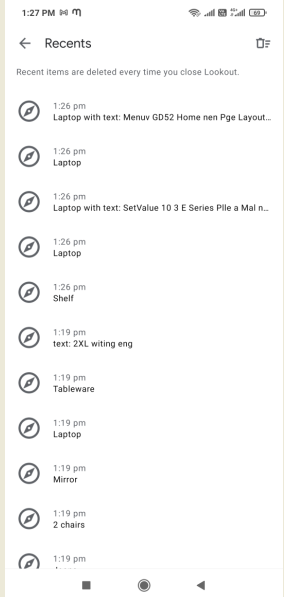
**FUTURE ENHANCEMENT**

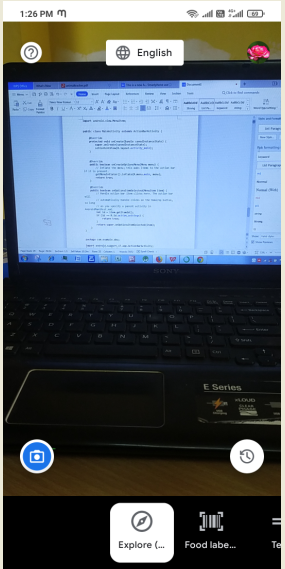
Future development is planned to integrate additional communication capabilities to give the smart-phone the ability to allow it to communicate with an automobile’s on-board diagnostic system to gain more information about driving conditions. This system would use current road conditions and real-time traffic information from the Internet to assist in the determination of the best route given the conditions. Such a system could be used to provide drivers an evacuation route in emergencies.

**APPENDIX 1**

**SCREENSHOTS:**







**APPENDIX 2**

**REFERENCE:**

[1] US Government, Global Positioning System, http://gps.gov/ [last accessed on Mar 1, 2009].

[2] 3G, Assisted-GPS Test Calls for 3G WCDMA Networks, http://www.3g.co.uk/PR/November2004/8641.htm [last accessed on Mar 1, 2009].

[3] Open Handset Alliance, Open Handset Alliance Announces 14 New Members, http://www.openhandsetalliance.com/press\_120908.html [last accessed on Mar 1, 2009].

[4] Google, Inc., What is Android?, http://code.google.com/android/what-isandroid. html [last accessed on Mar 1, 2009].

[5] Computerworld, Inc., iPhone Apps distribution still an Issue for Business,

http://www.computerworld.com/action/article.do?command=viewArticle Basic& article Id=9095398 [last accessed on Mar 1, 2009].

[6] The Apache Foundation, Licenses, http://www.apache.org/licenses/ [last

accessed on Mar 1, 2009].

[7] HTC, T-Mobile G1 - Specification, http://www.htc.com/www/product/g1/specification.html [last accessed on Mar 1, 2009].