INTRODUCTION

1.INTRODUCTION

In today's world, it is not safe for a person to travel alone at night especially for women; it will be high time to travel alone because a woman is not highly strong as men to protect herself from them. The good way to reduce chances in becoming a victim of violent crime to identify and call on resources to help you out of unsafe situations. Whether you are in instant trouble or got separated from friends during night and do not know to reach home, having these app on your phone can diminish our risk and bring assistance when we require it. In this Project, we present Security Alert an application for smart phones working over android platform. National Crime Records Bureau of India, reported incidents of crime against women increased 6.4% during 2012, and a crime against a woman is committed every three minutes. 65% of Indian men believe women should tolerate violence in order to keep the family together, and women sometimes deserve to be beaten. In January 2011, the International Men and Gender Equality Survey (IMAGES) Questionnaire reported that 24% of Indian men had committed sexual violence at some point during their lives. Our motto in developing this app is to provide a safe environment to women through smart phone as today most of the people are carrying smart phones to wherever they go. Although there are many security measures are implemented and many more security apps are come up into the spotlight but all the existing applications having some drawbacks to provide the ultimate security. So, it is better to take our own safety measures rather than becoming a victim of those crimes. This Application represents women security, an Android Application for the Safety of Women and this app can be activated this app by shaking the mobile, whenever need arises. This app identifies the location of place through GPS and sends a message comprising this location URL to the registered contacts (Emergency contacts) also send messages to nearby police stations , sub control police stations and mobiles which are having this app. This application aims to ensure women safety. This is achieved by addressing the circumstances that compromise the safety of women in today's day and present scenario. This app ensures women are not put into such situations through various features offered by our Android Application system.

1.1 FEASIBILITY STUDY:

- Preliminary investigation examine project feasibility, the likelihood the system will be useful to the organization. The main objective of the feasibility study is to test the Technical, Operational and Economical feasibility for adding new modules and debugging old running system. All system is

feasible if they are unlimited resources and infinite time. There are aspects in the feasibility study portion of the preliminary investigation:

- A feasibility study assesses the economic, technical, and operational merits of the proposed project. A Project is economically feasible if costs do not overshadow benefits. A project is technically feasible if the technology is available and capable of meeting users requests. A project is operationally feasible if the proposed system will operate and be used once it is installed.

1.Technical Feasibility

2.Operational Feasibility

3.Economic Feasibility

1.1.1 TECHNICAL FEASIBILITY

- Evaluating the technical feasibility is the trickiest part of a feasibility study. This is because, at this point in time, not too many detailed design of the system, making it difficult to access issues like performance, costs on (on account of the kind of technology to be deployed) etc. A number of issues have to be considered while doing a technical analysis.
- A study of resource availability that may affect the ability to achieve an acceptable system. Technical feasibility is the most difficult area to ensure at initial stage. Since the objectives, functions, performance cannot be predicted to its fullest, everything seems possible, provided the right assumptions are made.
- It is essential that the process of analysis and definition can be conducted in parallel with an assessment of technical feasibility. The consideration that is normally associated with technical feasibility includes resource availability at the organization where the project is to be developed and implemented.

1.1.2 OPERATIONAL FEASIBILITY

- It deals with the consideration about working of the system after installation. The proposed system would be beneficial to its users as their needs are fully satisfied. As this project satisfies all the requirements of the users it is operationally feasible. All the operational aspects are considered carefully here. Only by spending tie to evaluate feasibility we will be able to reduce the chances for extreme embracement at later stages of a project.
- Early involvement reduces the chances of resistance to the system and in General and increases the likelihood of successful project. Since the proposed system was to help reduce the hardships encountered. In the existing manual system, the new system was considered to be operational feasible

1.1.3 ECONOMIC FEASIBILITY

- Economic feasibility attempts to weigh the costs of developing and implementing a new system, against the benefits that would accrue from having the new system in place. This feasibility study gives the top management the economic justification for the new system. The purpose of an Economic Feasibility

Study (EFS) is to demonstrate the net benefit of a proposed project for accepting or disbursing electronic

funds/benefits, taking into consideration the benefits and costs to the agency, other state agencies, and the general public as a whole.

1.2 EXISTING SYSTEM

- As a part of our project, we go through some applications of women safety that already exist in the market. We observe how these applications work and to see how they can be improved and how are they different. In this present era safety is one of the mostly raising issue faced by the people especially women. It will be high time to travel alone because a women is not highly strong as men to protect herself from others. Although there are many security measures are implemented and many more security apps are come up into the spotlight but all the existing applications having some drawbacks to provide the ultimate security.

Drawbacks of Existing System

The disadvantages of using these applications are:

- They only send the alert messages to the saved contacts but there is no option of calling.
- Because of previous systems there is less possibilities of overcome the threatening situations of women.
- Previous applications also have GPS tracking system for to track the live location of women but it has not specific range.
- Emergency contacts are only up to 3 to 4 persons.

2. PROPOSED SYSTEM:

- Our app ensures women are not put into such situation through various features offered by our Android Application System .This app identifies the location of place through GPS and sends a message comprising this location URL to the registered contacts (Emergency Contacts) also send messages to nearby police stations, mobiles which are having this app.
- Initially it gets the name and the number of the user then it asks for personal details for the purpose of registration, in which the data entered is assured with absolute security. Then the registered user can enter five to seven contacts to get in contact with at the time of emergency, after that the final step of registration is done with an OTP verification (to check if the number given by the user is in use or not). Here the details given by the user at the time of registration is stored in the cloud database.
- Once the registration process gets completed a screen is displayed . On clicking the panic a text message is sent to the registered contacts with the GPS location of the user, stating that the person is in trouble and in need of help. When the other button is clicked it comes out and gets to the registration page. Where the user might meet to reregister in order to use the application.

- If the user is in a very dangerous situation where she might not have enough time to open the application and click on the button . At that time a single shake of the mobile phone is enough, a call is sent to the police. This is the whole working procedure of the proposed system.

Advantages of using Proposed System

Authentication

- Authentication feature in firebase lets only the authorized users to access the application. Firebase provides login through Phone Number, and also it allows the developer to customize their own authentication.

Real-time Database

- Database in firebase is a cloud-based database and does not need SQL queries to store and retrieve data. Database is highly reliable thus even if connection is lost the data is maintained.

Storage

- Firebase also provides storage facility. It can store and retrieve various content like images, videos and audio directly from client SDK. Uploading and downloading is done in the background. Data stores are safe and secure that only the authorized user can access it.

Messaging

- It is a cross-platform messaging solution that lets us dependably convey messages at zero expense. We can inform a customer that new email or other information is accessible to sync. We can send notification messages to drive user reengagement and maintenance.

Notifications

- It enables destination user notifications for mobile app developers and the services are freely available.

SOFTWARE REQUIREMENT

SPECIFICATION

2. SOFTWARE REQUIREMENT SPECIFICATION

2.1 INTRODUCTION

SRS is a document that contains information about the functions and goals of the future digital solution and its principles of operation. Based on this document, the entire process of developing a software product is built and followed by all participants in the development process; based on it, the design is done, and development is carried out. Checklists are written on its basis (which determines the conditions for accepting the finished product).

The SRS is necessary for all structural units of the development team: Project Management, Software Development, Quality Assurance, Client, and Business Development.

Purpose

The purpose of this document is to give a detailed description of the requirements for this application. It will illustrate the purpose and complete declaration for the development of system. It will also explain system constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team. Constraints, interface and interactions with other external applications. This document is primarily intended to be proposed to a customer for its approval and a reference for developing the first version of the system for the development team.

2.2HARDWARE & SOFTWARE REQUIREMENT

Hardware Specification:

- Hard Disk – 5 GB
- A computer with at least 2GB of RAM(Random Access Memory)
- Processor – i3

Software Requirements:

- Android Studio
- Languages: Java and Xmls

2.3 Functional Requirements

- In Software engineering and systems engineering, a functional requirement defines a function of a system or its component. A function is described as a set of inputs, the behaviour and outputs.
- Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish.
- Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases.
- Functional requirements are supported by non-functional requirements which impose constraints on the design or implementation.
- As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system.

2.4 Non functional Requirements

In systems engineering and requirements engineering, a non-functional requirement is a requirement that specifies criteria that can be used to judge the operation of a system, rather than specific behaviours. They are contrasted with functional requirements that define specific behaviour or functions. The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture, because they are usually Architecturally Significant Requirements.

Broadly, functional requirements define what a system is supposed to do and non functional requirements define how a system is supposed to be. Functional requirements are usually in the form of, an individual action or part of the system, perhaps explicitly in the sense of a mathematical function, a black box description input, output, process and control functional model or IPO Model. In contrast, non-functional requirements are in the form of, an overall property of the system as a whole or of a particular aspect and not a specific function.

The system's overall properties commonly mark the difference between whether the development project has succeeded or failed.

Non-functional requirements are often called "quality attributes" of a system. Other terms for non-functional requirements are "qualities", "quality goals", "quality of service requirements", "constraints" and "non-behavioural requirements". Informally these are sometimes called the

"ileitis",from attributes like stability and portability.

- Qualities that is non-functional requirements can be divided into two main categories:

  1. Execution qualities, such as safety, security and usability, which are observable during operation.

  2. Evolution qualities, such as testability, maintainability, extensibility and scalability, which are embodied in the static structure of the system.

ANALYSIS

3.OBJECT ORIENTED ANALYSIS

Object oriented analysis can be defined as investigation and to be more specific it is the investigation of objects. The most important purpose of OO analysis is to identify objects of a system to be designed. This analysis is also done for an existing system. Now an efficient analysis is only possible when we are able to start thinking in a way where objects can be identified. After identifying the objects their relationships are identified and finally the design is produced. The objects should be identified with responsibilities. Responsibilities are the functions performed by the object. Each and every object has some type of responsibilities to be performed. When these responsibilities are collaborated the purpose of the system is fulfilled.

Object oriented analysis is done by using some UML concepts.

- UML is a standard language for specifying, visualizing, constructing, and documenting the artifacts of software systems.
- U.M.L stands for Unified Modeling Language.
- UML is a pictorial language used to make software blue prints.
- OOA can be described by using UML diagrams like:

  1. USECASE DIAGRAM
  2. SEQUENCE DIAGRAM
  3. COLLABORATION DIAGRAM
  4. ACTIVITY DIAGRAM
  5. STATE-CHART DIAGRAM

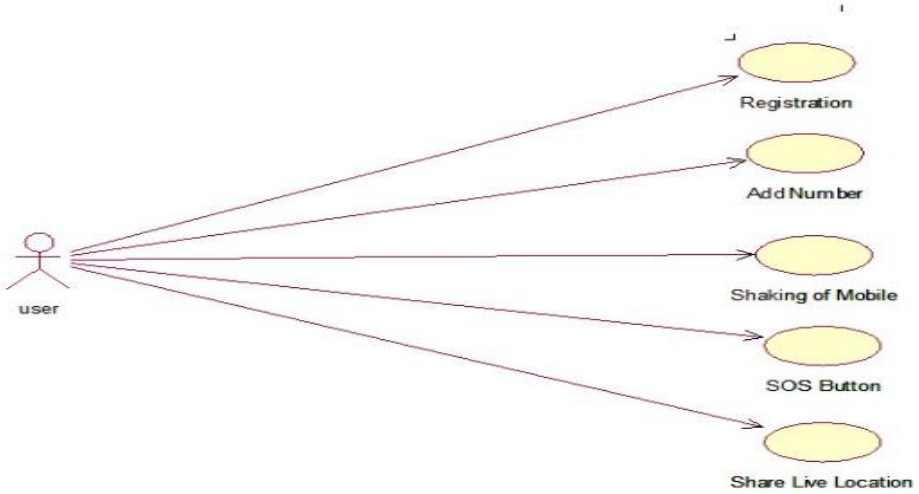
3.1 USECASE DIAGRAM

A usecase diagram at its simplest is a representation of users interaction with the system and depicting the specifications of usecase . A usecase diagram can portray the different types of users of a system and the various ways that they interact with the system.

- The two main components of a use case diagram are use cases and actors which is represented below:

An actor is represents a user or another system that will interact with the system you are modeling. A usecase is an external view of the system that represents some action the user might perform in order to complete a task. They are helpful in exposing requirements and the planning the project.

Usecase diagram:

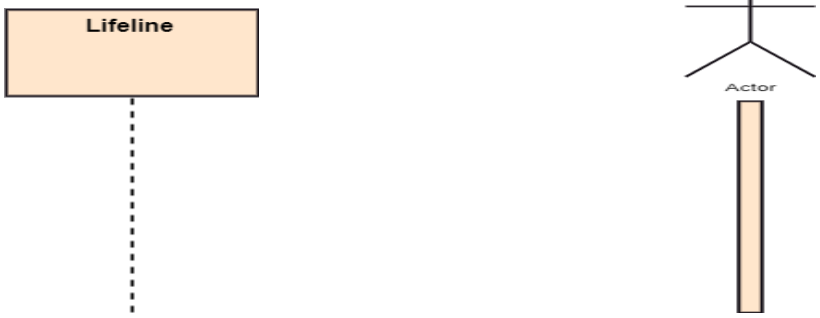


The user will first register into the application and then Add numbers of their beloved ones . whenever the user feels like she is in danger by shaking the mobile it will send the alert message to her guardians along with live location.

3.2 SEQUENCE DIAGRAM

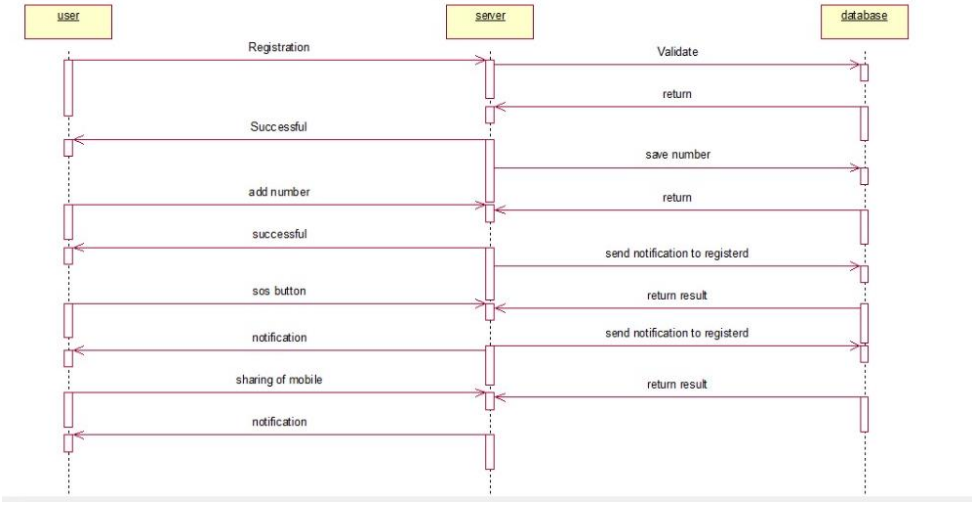
A sequence diagram is an interaction diagram that shows how process operate with one another and in what order. It is a construct of Message Sequence Chart. A sequence diagram shows object interactions arranged in time sequence.

The notations that we are using in our diagram are



- An individual participant in the sequence diagram is represented by a lifeline. It is positioned at the top of the diagram.
- A role played by an entity that interacts with the subject is called as an actor. It is out of the scope of the system. It represents the role, which involves human users and external hardware or subjects. An actor may or may not represent a physical entity, but it purely depicts the role of an entity. Several distinct roles can be played by an actor or vice versa.
- A role played by an entity that interacts with the subject is called as an actor. It is out of the scope of the system. It represents the role, which involves human users and external hardware or subjects. An actor may or may not represent a physical entity, but it purely depicts the role of an entity. Several distinct roles can be played by an actor or vice versa.

Sequence diagram



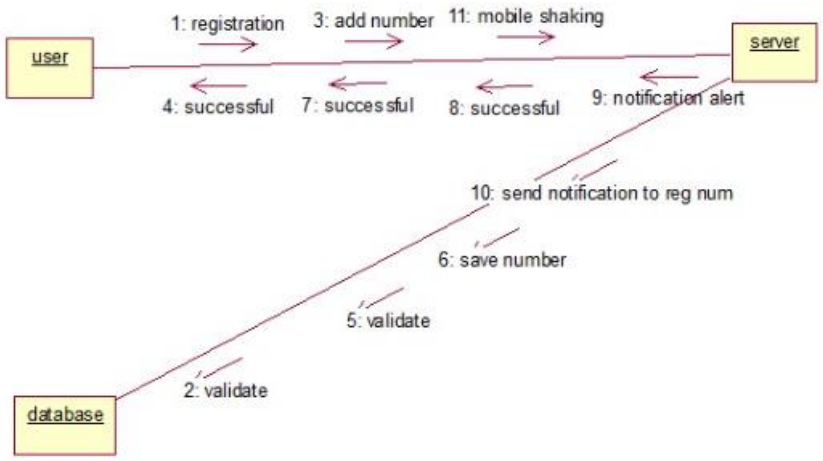
Here the user is to interact with server for registration and the server communicate with database to validate and it sends to the user as successful .These all are interact with each of them for sending messages as shown in the figure.

3.3 COLLABORATION DIAGRAM

Collaboration diagram is also known as communication diagram. A communication diagram models the interaction between objects or parts in terms of sequenced messages.

- The elements of a system work together to accomplish the systems objective and a modeling language must have a way of representing this.
- The UML Collaboration diagram is designed for this purpose, it is an extension of the object diagram. In addition to the association among objects the collaboration diagram shows the messages the objects and each other.
- diagram has three elements,

1. Object: The interaction between objects takes put in a system.
2. Relation/Association: Association among objects is connected by connecting them.
3. Messages: An arrow that commencing from one object to the destination object.



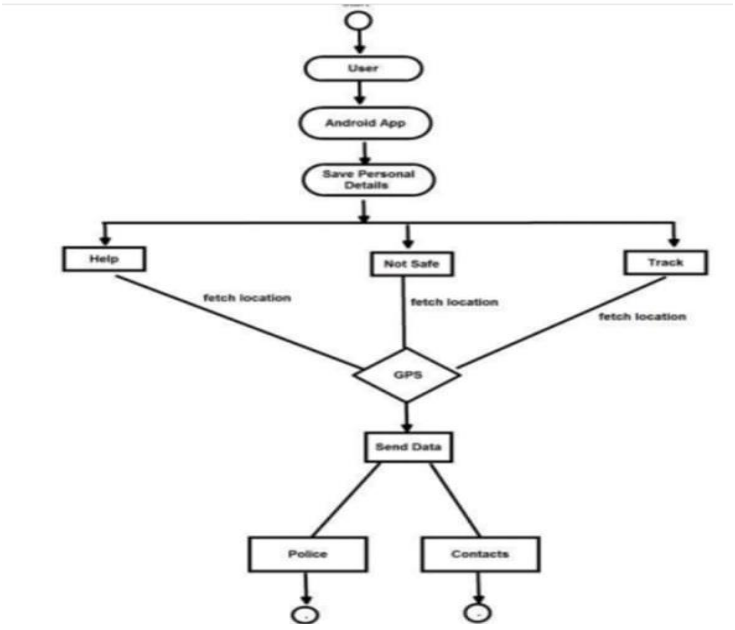
3.4 ACTIVITY DIAGRAM

Activity diagram are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the unified, activity diagrams are intended to model both computational and organizational and organizational process. Activity diagram show the overall flow of control.

Basic components of an activity diagram

Before you begin making an activity diagram, you should first understand its makeup. Some of the most common components of an activity diagram include:

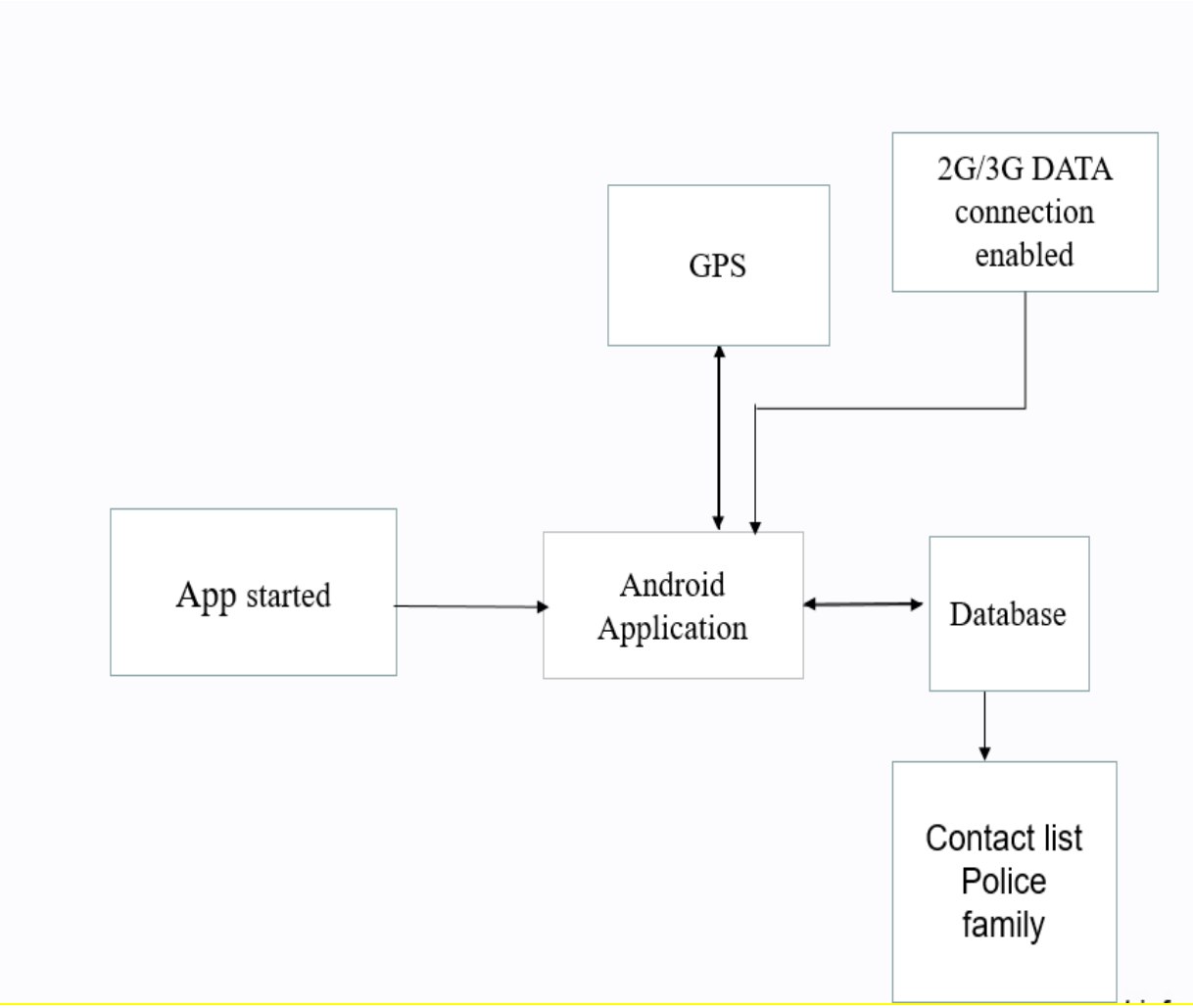
- Action: A step in the activity wherein the users or software perform a given task. In Lucid chart, actions are symbolized with round-edged rectangles.
- Decision node: A conditional branch in the flow that is represented by a diamond. It includes a single input and two or more outputs.
- Control flows: Another name for the connectors that show the flow between steps in the diagram.
- Start node: Symbolizes the beginning of the activity. The start node is represented by a black circle.
- End node: Represents the final step in the activity. The end node is represented by an outlined black circle.



DESIGN

4.1 ARCHITECTURE

- The proposed architecture depicted below shows the exact flow of control of the android application.
- Here the database acts as a storing media between the two mobile devices
- The database information i.e., to which database the information has to be sent, the URL of the database is coded itself in the application.
- From the database, the location coordinates are sent continuously to the registered contacts of the user.



Android Application

This is user-friendly application that can be accessed by anyone who has installed it in their smart phones. Our intention is to provide you with fastest and simplest way to contact your nearest help. In this system user needs to feed five to seven contact numbers, in case of emergency on moving the phone up and down thrice, the system sends SMS and calls on one of the numbers feeded into the system, it sends the decision for the primary added guardian number and sends the message that she is in peril and sends the placement message to the all saved guardian contacts.

This application provides facilities like

- To send an emergency message to a few different contacts (registered contacts).
- To create an outgoing call to the nearby police stations.
- To grasp the user's location.

Global Positioning System

There are a few enabling technologies spurring this progress in location capabilities. You're probably already familiar with global positioning system (GPS) satellite navigation system technology. A GPS receiver calculates its position by precisely timing messages sent by GPS satellites and computing the distance to each satellite.

There's also related capability called Assisted Global Positioning System (A-GPS). A-GPS speeds up the process of getting a "first fix" on the location of a cellular-connected device by using data shared between satellites and cellular networks. In other words, mobile applications don't always have to directly ping the faraway system of satellites to get GPS information, which can take as long as two minutes. Instead, they can extract the GPS's location information that's already stored in cellular network backbone equipment. That cuts the access time to about 30 seconds within approximately a 90-foot accuracy when outdoors.

2G/3G Data Enabled

- Android lets your application connect to the internet or any other local network and allows you to perform network operations.
- A device can have various types of network connections.

Checking Network Connection

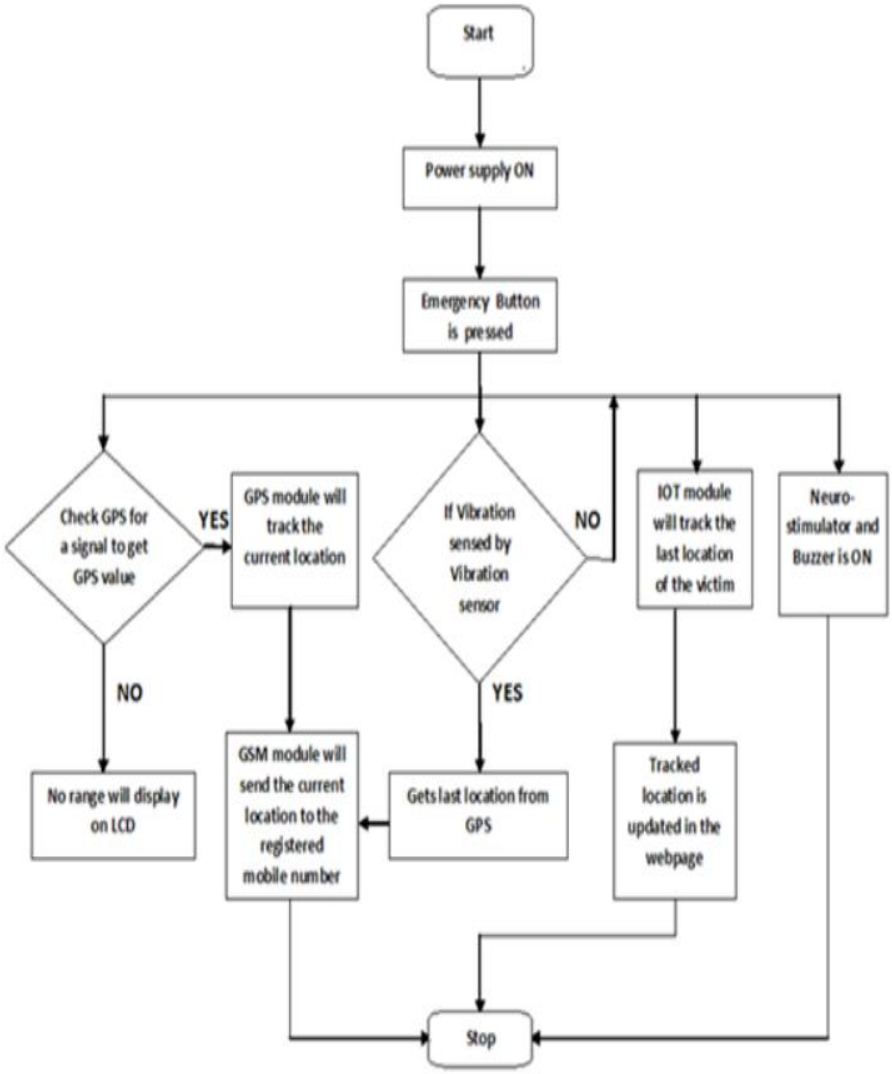
- Before you perform any network operations, you must first check that are you connected to that network or internet e.t.c. For this android provides **ConnectivityManager** class. You need to instantiate an object of this class by calling **getSystemService()** method.
- Once you instantiate the object of **ConnectivityManager** class, you can use **getAllNetworkInfo** method to get the information of all the networks.
- After checking that you are connected to the internet, you can perform any network operation

Database

- **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 bydefault. So, there is no need to perform any database setup or administration task.
- Saving data to a database is ideal for repeating or structured data, such as contact information. The APIs you'll need to use a database on Android are available in the **android.database.sqlite** package.
- Once you have defined how your database looks, you should implement methods that create and maintain the database and tables.

4.2 PROJECT FLOW:

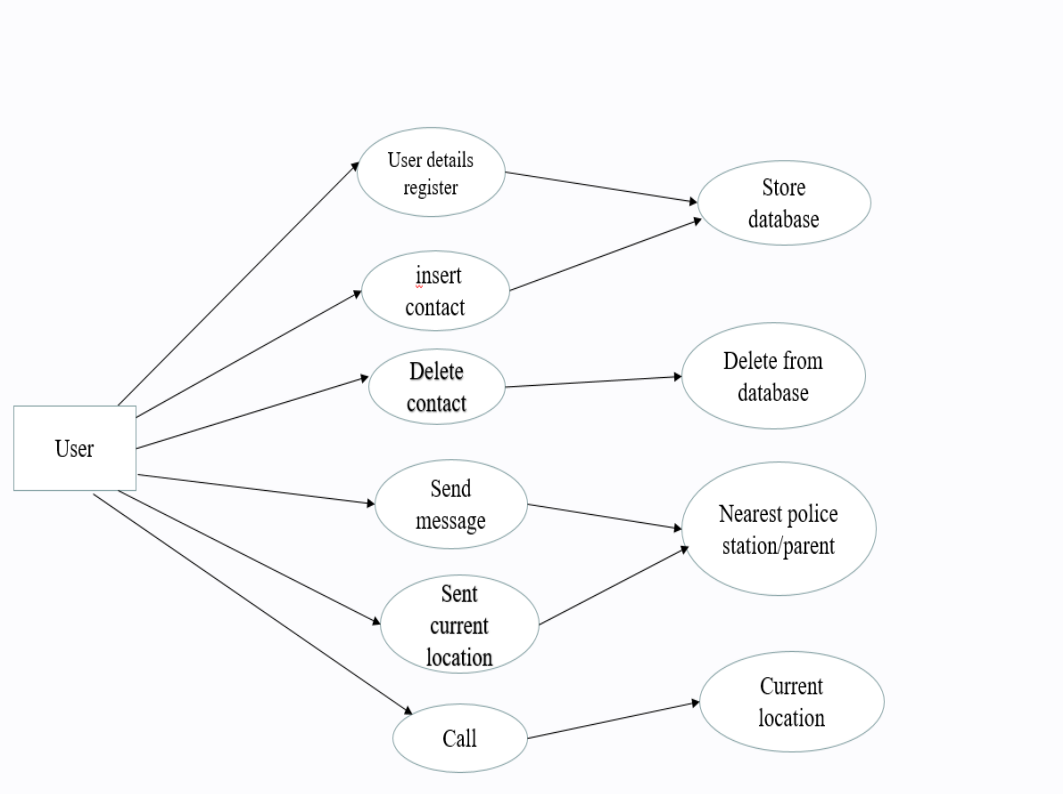
- Project flow defines the project activities and end products that will be performed and describes how the activities will be accomplished. The purpose of the project planning is to define each major task, estimate the time and resources required, and provide a framework for management review and control.
- Project flow can be used in designing and documenting both simple and complex processes or programs and, similar to the other types of diagrams, they can help visualize what happens and so help understand some definite process, and, as a result, find flaws, bottlenecks and other features within it.



- The user will first register into the application and then add the numbers of their guardians,initially it gets the name and the number of the user then it asks for personal details for the purpose of registration, in which the data entered is assured with absolute security.
- Then the registered user can enter five to seven contacts to get in contact with at the time of emergency, Here the details given by the user at the time of registration is stored in the database.
- Once the registration process gets completed a screen is displayed. On clicking the button a text message is sent to the registered contacts with the GPS location of the user, stating that the person is in trouble and in need of help. When the other button is clicked it comes out and gets to the registration page. Where the user might meet to reregister in order to use the application.
- If the user is in a very dangerous situation where she might not have enough time to open the application and click on the button. At that time a single shake of the mobile phone is enough, a call is sent to the police. This is the whole working procedure of the proposed system.

4.3 DATA FLOW DIAGRAM

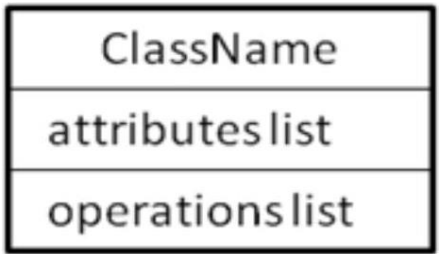
- A data flow diagram (DFD) is a graphical representation of the "flow" of data through an information system, modelling its process aspects.
- A DFD is often used as a preliminary step to create an overview of the system without going into great detail, which can later be elaborated. DFDs can also be used for the visualization of data processing (structured design).
- A DFD shows what kind of information will be input to and output from the system, how the data will advance through the system, and where the data will be stored.
- It does not show information about process timing or whether processes will operate in sequence or in parallel, unlike a traditional structured flowchart which focuses on control flow, or a UML activity workflow diagram, which presents bothcontrol and data flows as a unified model.
- The dataflow diagram depicted as follows



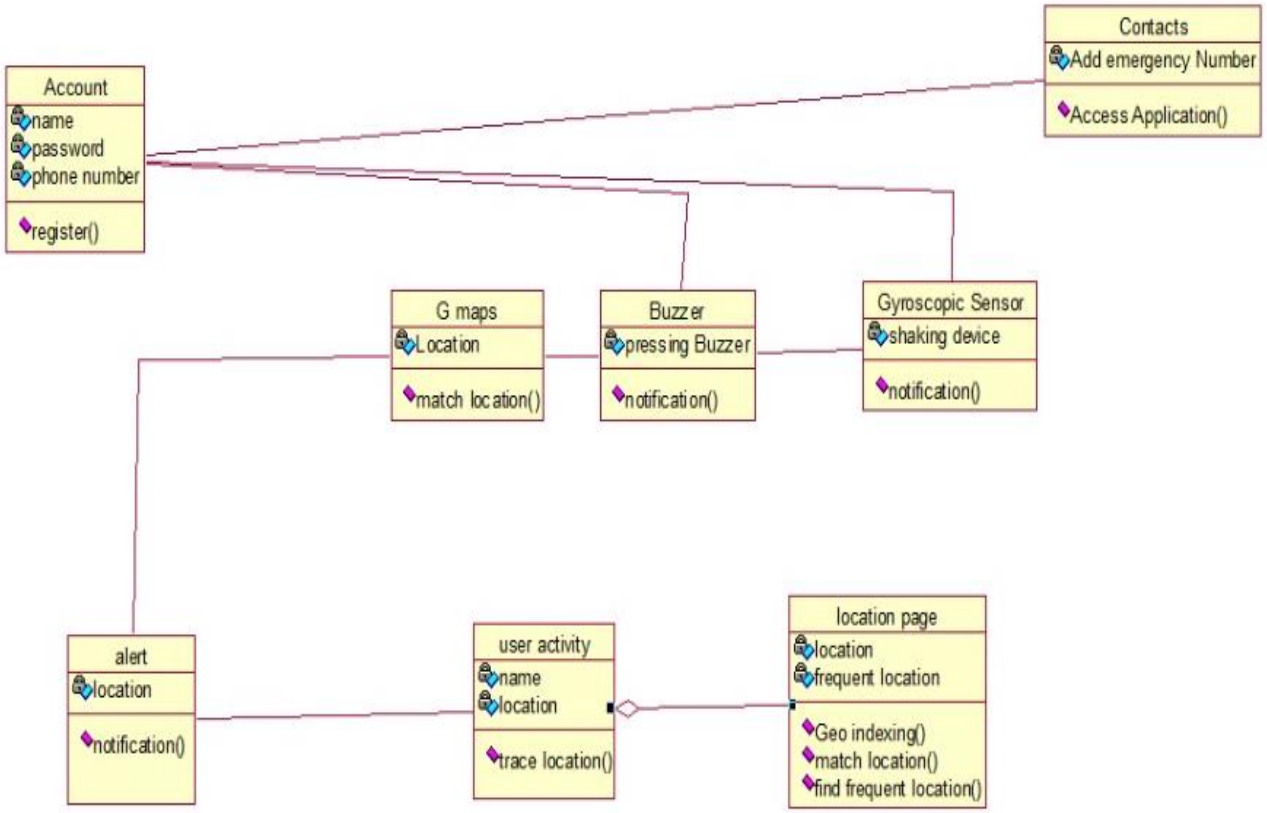
4.4 CLASS DIAGRAM:

- A class diagram in the Unified Modeling Language (UML) is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects. The class diagram is the main building block of objectoriented modeling. It is used both for general conceptual modeling of the systematic of the application, and for detailed modeling translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main objects, interactions in the application and the classes to be programmed.
- Class are composed of three things: a name, attributes, operations.

Representation:



- Class diagram for women safety Android Application consists of 8 classes
  They are: Account, contacts, G maps, Buzzer, Gyroscopic sensor ,alert, user activity, location page.


Account
name
password
phone number
register()
Contacts
Add emergency Number
Access Application()
G maps
Location
match location()
Buzzer
pressing Buzzer
notification()
Gyroscopic Sensor
shaking device
notification()
alert
location
notification()
user activity
name
location
trace location()
location page
location
frequent location
Geo indexing()
match location()
find frequent location()

IMPLEMENTATION

5.1 INRODUCTION TO TECHNOLOGY

This section describes the technologies required to build this project. It is important to understand the following technologies and programming languages in order to achieve the objectives of this project, Android, Java , JSON and MySQL Database.

5.1.1 ANDROID

The evolution of Android has an important impact on the area of technology in today. Android is an operating system that is based on Linux kernel and it is developed by Google. There are different types of Androids which are basically named depending on different devices on which the operating system is used, such as Android Mobile which is used on Mobile devices basically mobile phones and tablets, Android TV which is used on televisions, Android Auto which is used for cars and Android Wear which is used for wearable devices.

The first commercial version of Android popularly known as Android 1.0 was released in 2008 and ever since that year, each subsequent versions of Android had been released with major focus on improving performance, the user interface design and providing many features such as voice searching.

Each version of Android released can be identified with code names which are organized in alphabetical order from the first commercial version with code name Alpha to the latest Android version with code name Nougat which was released in August 22, 2016 Open 21 .

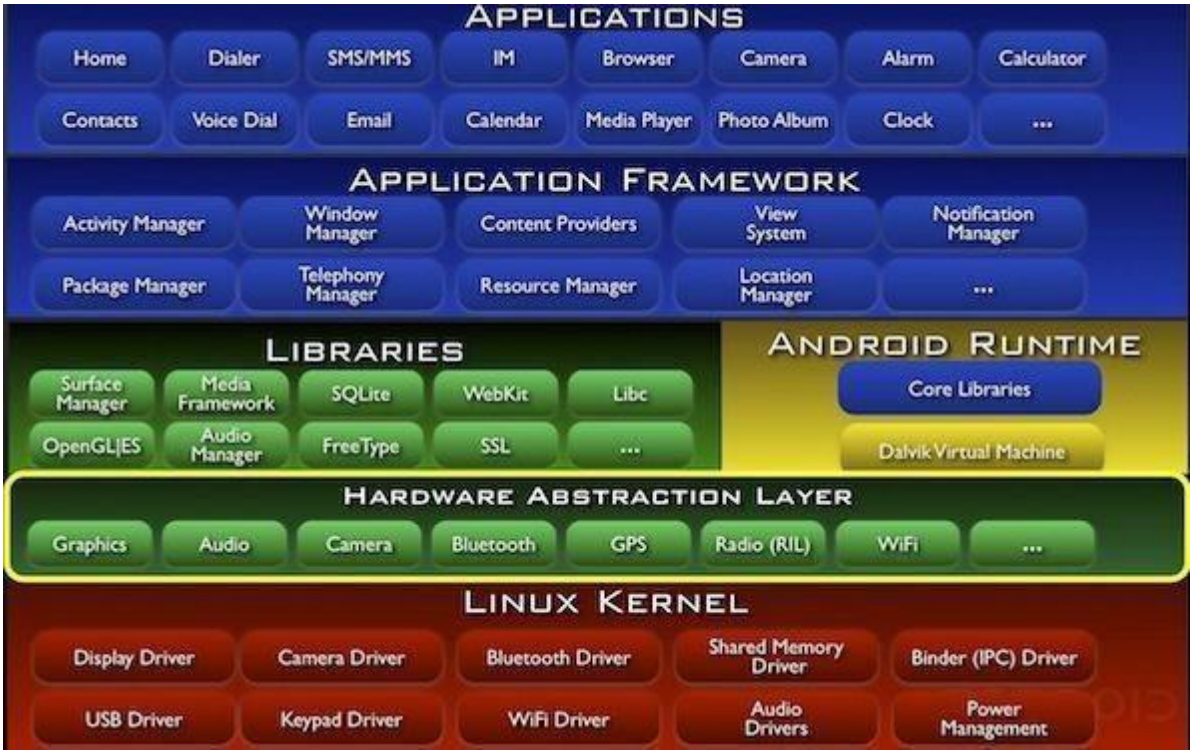


Fig: Android's Architecture Diagram

HOW TO IMPLEMENT

Android is developed by Google until the latest changes and updates are ready to be released, at which point the source code is made available to the Android Open Source Project (AOSP), an open source initiative led by Google. The AOSP code can be found without modification on select devices, mainly the Nexus and Pixel series of devices. The source code is, in turn, customized and adapted by original equipment manufacturers (OEMs) to run on their hardware.

1.UPDATE SCHEDULE : Gooogle announces major incremental upgrades to Android on a yearly basis. The updates can be installed on devices over-theair. The latest major release is 8.0 "Oreo", announced in March 2017, and released the following August. The extensive variation of hardware in Android devices causes significant delays for software upgrades, with new versions of the operating system and security patches typically taking months before reaching consumers, or sometimes not at all. In 2012, Google began decoupling certain aspects of the operating system (particularly its core applications) so they could be updated through the Google Play store independently of the OS. One of those components, Google Play Services, is a closed-source system-level process providing APIs for Google services, installed automatically on nearly all devices running Android 2.2 "Froyo" and

higher. With these changes, Google can add new system functionality through Play Services and update apps without having to distribute an upgrade to the operating system itself. In May 2017, with the announcement of Android 8.0,

1.Google introduced Project Treble, a major re-architect of the Android OS framework designed to make it easier, faster, and less costly for manufacturers to update devices to newer versions of Android. Project Treble separates the vendor implementation (device-specific, lower-level software written by silicon manufacturers) from the Android OS framework via a new "vendor interface". In September 2017, Google's Project Treble team revealed that, as part of their efforts to improve the security lifecycle of Android devices, Google had managed to get the Linux Foundation to agree to extend the support lifecycle of the Linux Long-Term Support (LTS) kernel branch from the 2 years that it has historically lasted to 6 years for future versions of the LTS kernel, starting with Linux kernel 4.4.

LINUX KERNEL:

Android's kernel is based on one of the Linux kernel's longterm support (LTS) branches. As of 2017, Android devices mainly use versions 3.18 or 4.4 of the Linux kernel. The actual kernel depends on the individual device. Android's variant of the Linux kernel has further architectural changes that are implemented by Google outside the typical Linux kernel development cycle, such as the inclusion of components like device trees, ashmem , ION, and different out of memory (OOM) handling. In December 2011, Greg Kroah-Hartman announced the start of Android Mainlining Project, which aims to put some Android drivers, patches and features back into the Linux kernel, starting in Linux 3.3. Linux included the auto sleep and wake locks capabilities in the 3.5 kernel, after many previous attempts at merger. The interfaces are the same but the upstream Linux implementation allows for two different suspend modes: to memory (the traditional suspend that Android uses), and to disk (hibernate, as it is known on the desktop). Google maintains a public code repository that contains their experimental work to re-base Android off the latest stable Linux versions. The flash storage on Android devices is split into several partitions, such as system for the operating system itself, and data for user data and application installations. In contrast to desktop Linux distributions, Android device owners are not given root access to the operating system and sensitive partitions such as /system are read-only. However, root access can be obtained by exploiting security flaws in Android, which is used frequently by the open-source community to enhance the capabilities of their devices, but also by malicious parties to install viruses and malware.

SOFTWARE STACK:

On top of the Linux kernel, there are the middleware, libraries and APIs written in C, and application software running on an application framework which includes Java-compatible libraries. Development of the Linux kernel continues independently of other Android's source code bases.

Until version 5.0, Android used Dalvik as a process virtual machine with tracebased just-in-time (JIT) compilation to run Dalvik "dex-code" (Dalvik Executable), which is usually translated from the Java bytecode. Following the trace-based JIT principle, in addition to interpreting the majority of application code, Dalvik performs the compilation and native execution of select frequently executed code segments ("traces") each time an application is launched. Android 4.4 introduced Android Runtime (ART) as a new runtime environment, which uses ahead-of-time (AOT) compilation to entirely compile the application bytecode into machine code upon the installation of an application. In Android 4.4, ART was an experimental feature and not enabled by default; it became the only runtime option in the next major version of Android, 5.0. For its Java library, the Android platform uses a subset of the now discontinued Apache Harmony project. In December 2015, Google announced that the next version of Android would switch to a Java implementation based on OpenJDK.

ADVANTAGES OF ANDROID

1. Supports 2D, 3D graphics

It supports various platforms like 2D and 3D. Earlier we used to watch movies and play games in almost in 2D, but nowadays various applications are using 3D format. To provide different graphics in videos, games OS should support 3D format. Android supports 2D And 3D format to provide a better advantage in videos and in games.

2. Supports Multiple Languages

Android supports different languages. We can say all famous languages about more than 100. By using this feature it is easy to adopt to different languages. Earlier in the feature phones English is to be the only language in the mobile devices.

3. Java Support

The Java supporting feature enables developers to enhance more features. As it supports Java, operating can be developed.

4. Faster Web Browser

As it enabled with web browser we surf web easily without complexity just like in a computer. It easily loads multimedia so that it makes web browsing faster.

5. It Supports MP4, 3GP, MPEG4, MIDI

It supports different types of formats. There is no need to convert from one format to another, as it enabled with different formats of audio and video styles.

6. Additional Hardware Support

Any hardware can be easily connected with the Android based devices easily. We can make a device to connect internally to get more features.

7. Video Calling

Faster data connection enables to do video call. We can take advantage of bandwidth and new generation networks using Android.

8. Open Source Framework

It makes users to make their own applications and to make changes required for themselves. Enthusiasts can make Andriod more powerful and useful by developing themselves. As it is an open source operating system, we can use it easily and without cost in the equipments.

9. Uses of Tools are Very Simple

It makes use of a single button to do more than assigned work. For example volume control button can be made to click a photo by changing simple algorithm in the android.

10. Availability of Apps

Anyone can make use lot of free apps in the app store and from other android stores. It gives freedom to install from third party users.

ANDROID SDK

Android SDK is a collection of API libraries, tools, scripts, and documentation. This component is included in Android Studio IDE and can also be downloaded, installed as stand-alone SDK tool and it is important to set up the path for the location. The new version of SDK and tools are added to Android Studio as they become available.

This provides developers with a packaged set of developer tools and API libraries that enables building of complete application, testing of the applications on virtual devices, and performing debugging and optimization.

ANDROID STUDIO IDE

Android Studio is the official Integrated Development Environment (IDE) that is based on JetBrains' IntelliJ and developed by Google specifically for building Android application. In accordance to the OS of the computer, Android Studio is available and can be downloaded from the official website of

Android's developer (https://developer.android.com/studio/index.html). Installation guide can be followed during the installation process after download of Android Studio.

ANDROID NDK

Libraries written in C/C++ can be compiled to ARM, MIPS or x86 native code (or their 64bit variants) and installed using the Android Native Development Kit (NDK). These native libraries can be called from Java code running under the Dalvik VM using the System, load Library call, which is part of the standard Android Java classes. Complete applications can be compiled and installed using traditional development tools. However, according to the Android documentation, NDK should not be used solely because the developer prefers to program in C/C++, as using NDK increases complexity while most applications would not benefit from using it. The ADB Debugger gives a root shell under the Android Emulator which allows ARM, MIPS or x86 native codes to be uploaded and executed. Native code can be compiled using Clang or GCC on a standard PC. Running native code is complicated by Android's use of a non-standard C library (libc, known as Bionic). The graphics library that Android uses to arbitrate and control access to this device is called the Skia Graphics Library (SGL), and it has been released under an open source licence. Skia has backends for both Win32 and Unix, allowing the development of cross-platform applications, and it is the graphics engine underlying the Google Chrome web browser. Skia is not an NDK API, though, and NDK developers use OpenGL. It is possible to use the Android Studio with Gradle to develop NDK projects.

DEBUG BRIDGE ANDROID

The Android Debug Bridge (ADB) is a toolkit included in the Android SDK package. It consists of both client and server-side programs that communicate with one another. The ADB is typically accessed through the command-line interface, although numerous graphical user interfaces exist to control ADB. The format for issuing commands through the ADB is typically:

adb [-d|-e|-s <serial Number>] <command>

where –d is the option for specifying the USB-attached device,

-e is for indication a running Android emulator on the computer,

-s is for specifying either one by its adb-assigned serial number

If there is only one attached device or running emulator, these options are not necessary.

5.1.2 JAVA JDK AND JRE

Android application development environments include essential software components such as Java Development Kit (Java JDK) and Java Runtime Environment (JRE). The minimum versions required

are Java JDK 5 and JRE 6. Java JDK is required for building application and one of the most important tools in JDK is the Java compiler which converts Java files into Java bytecode.

5.1.3 JSON

JSON which means JavaScript Object Notation is a lightweight text-based open standard that uses human-readable text to transmit data in name-value pairs. JSON is a language-independent format which was originally specified by Douglas Crockford and it is used with various modern programming languages such as PHP, Python, PERL and Java. JSON filename extension is represented by ,json .

JSON syntax which is a subset of JavaScript syntax specifies that JSON data is in name-values pairs, JSON data should be separated by commas, JSON objects are held inside curly braces {} and that JSON arrays are held inside square brackets [ ]. JSON values can be represented in Number, String, Boolean, Array, Object or Null data types.

The example below describes JSON syntax rules;

```
{
“teachers”: [
{“id”: “T100”, “firstName”: “Jerome”, “lastName”: “Bernard”},
{“id”: “T200”, “firstName”: “James”, “lastName”: “Colman”}
]
}
```

5.1.4 MySQL Database

A database can be defined as collections of data that are well organized in order to easily access, manage and update the data and database management system (DBMS) is a software application that is designed for managing database activities. The DBMS functions can be mainly classified into four types, which are data definition, update, retrieval, and administration. MySQL is one of the most popular DBMS.

MySQL is an open source relation SQL database management system that can be used with modern programming languages such as PHP, PERL, and JAVA, and can be used on various operating systems such as Windows, Linux, Solaris, OS X and FreeBSD. It is originally created by Swedish company MySQL AB, but now owned by Oracle Corporation and can be freely downloaded from its official website http://www.mysql.com/.

5.2 SAMPLE CODE

This section describes the implementation of all the graphical user interfaces involved in this application. Implementation is the realization of application specification, idea and design. It involves programming codes written in order to achieve ideas and requirements of each user interface of the application. Implementation of each user interface involved XML codes for user interface design, ANDROID JAVA codes for back end.

5.2.1Android Manifest

The below section of code snippet shows the use of internet permission for the application which means that the availability of internet on the user's mobile phone is important for the application to work. It shows all the activities involved in the application with intent filter embedded in each activity. The activity with main and launcher inside the embedded intent filter indicates the first page activity of the application that is presented to the users when the application is launched.

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.womensafety">
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/Theme.WomenSafety">
        <activity android:name=".LiveLocationScreen"></activity>
        <activity android:name=".Feedback" />
        <activity android:name=".Report" />
        <activity android:name=".Contact" />
        <activity android:name=".EmergencyCall" />
        <meta-data
            android:name="com.google.android.actions"
```

```
        android:resource="@xml/radius" />
    <activity android:name=".MainActivity" />
    <activity android:name=".Welcome">
        <intent-filter>
            <action android:name="android.intent.action.MAIN" />
            <category android:name="android.intent.category.LAUNCHER" />
        </intent-filter>
    </activity>
  </application>
</manifest>
```

Main Activity

The "main" activity is the activity that loads first and the rest of your application. Every application can have multiple activities, therefore you can list other activities to load and use later on but you can only have one "main" activity.

```
package com.example.womensafety;
import androidx.appcompat.app.AppCompatActivity;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.view.View;
import android.view.Window;
import android.view.WindowManager;
import android.widget.ImageButton;
import android.widget.ImageView;
import android.widget.Toast;
import java.util.Objects;
public class MainActivity extends AppCompatActivity {
   ImageButton call,location,video,contact,report,feedback;
   @Override
   protected void onCreate(Bundle savedInstanceState) {
```

```
        super.onCreate(savedInstanceState);
        requestWindowFeature(Window.FEATURE_NO_TITLE);
this.getWindow().setFlags(WindowManager.LayoutParams.FLAG_FULLSCREEN,WindowManag
er.LayoutParams.FLAG_FULLSCREEN);
        Objects.requireNonNull(getSupportActionBar()).hide();
        setContentView(R.layout.activity_main);
        call = findViewById(R.id.callBtn);
        location = findViewById(R.id.locationBtn);
        video = findViewById(R.id.videoBtn);
        contact = findViewById(R.id.contactBtn);
        report = findViewById(R.id.reportBtn);
        feedback = findViewById(R.id.feedbackBtn);

        call = findViewById(R.id.callBtn);
        call.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(MainActivity.this, "Calling and Sending an Alert Message to your Top
Priority Contacts...", Toast.LENGTH_SHORT).show();
            }
        });
        contact.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this,Contact.class));
            }
        });
        video.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                gotoUrl("https://youtube.com/playlist?list=PLA86B58B7DA1FF904");
            }
        });
```

```
        report.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this,Report.class));
            }
        });
        feedback.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this, Feedback.class));
            }
        });
        location.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                startActivity(new Intent(MainActivity.this, LiveLocationScreen.class));
            }
        });
    }
    private void gotoUrl(String s) {
        Uri uri = Uri.parse(s);
        startActivity(new Intent(Intent.ACTION_VIEW,uri));
    }
}
```

Location

A data class representing a geographic location. A location consists of a latitude, longitude, timestamp, accuracy, and other information such as bearing, altitude and velocity. All locations generated through LocationManager are guaranteed to have a valid latitude, longitude, timestamp (both Unix epoch time and elapsed realtime since boot), and accuracy. All other parameters are optional. Location-based services in android are those services that deal with the device's geographical location.

```
package com.example.womensafety;

    import androidx.annotation.NonNull;

    import androidx.appcompat.app.AppCompatActivity;

    import androidx.core.app.ActivityCompat;

    import android.Manifest;

    import android.content.pm.PackageManager;

    import android.location.Address;

    import android.location.Geocoder;

    import android.location.Location;

    import android.os.Bundle;

    import android.text.Html;

    import android.view.View;

    import android.widget.Button;

    import android.widget.TextView;

    import android.widget.Toast;

    import com.google.android.gms.location.FusedLocationProviderClient;

    import com.google.android.gms.location.LocationServices;

    import com.google.android.gms.tasks.OnCompleteListener;

    import com.google.android.gms.tasks.Task;

    import java.io.IOException;
```

```
import java.util.List;

import java.util.Locale;

//import static androidx.constraintlayout.motion.widget.Debug.getLocation;

public class LiveLocationScreen extends AppCompatActivity {

    Button bt,bt2;

    TextView tv1,tv2,tv3,tv4,tv5;

    FusedLocationProviderClient fusedLocationProviderClient;

    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_live_location_screen);

        bt=findViewById(R.id.bt_location);

        bt2 =findViewById(R.id.sendLocation);

        tv1=findViewById(R.id.text_view1);

        tv2=findViewById(R.id.text_view2);

        tv3=findViewById(R.id.text_view3);

        tv4=findViewById(R.id.text_view4);

        tv5=findViewById(R.id.text_view5);

        fusedLocationProviderClient = LocationServices.getFusedLocationProviderClient(this);

        bt2.setOnClickListener(new View.OnClickListener() {
```

```
        @Override
        public void onClick(View v) {
            Toast.makeText(LiveLocationScreen.this, "Sending Location Update...",
Toast.LENGTH_SHORT).show();
        }
    });
    bt.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            //check permission
            if (ActivityCompat.checkSelfPermission(LiveLocationScreen.this,
Manifest.permission.ACCESS_FINE_LOCATION) ==
PackageManager.PERMISSION_GRANTED){
                // when pemission grantied
                getLocation();
            }else {
                // when denied
                ActivityCompat.requestPermissions(LiveLocationScreen.this
                        ,new String[]{Manifest.permission.ACCESS_FINE_LOCATION} ,44);
            }
        }
```

```
        });
    }
    private void getLocation() {
        fusedLocationProviderClient.getLastLocation().addOnCompleteListener(new
OnCompleteListener<Location>() {
  @Override
  public void onComplete(@NonNull Task<Location> task) {
    // initiatlize location
    Location location = task.getResult();
    if (location !=null){
      try {
        // initalize geocoder
        Geocoder geocoder = new Geocoder( LiveLocationScreen.this, Locale.getDefault());
        //initilize address list
        List<Address> addresses = geocoder.getFromLocation(
            location.getLatitude(),location.getLongitude(),1
        );
        // set latitude on text view
        tv1.setText(Html.fromHtml(
            "<font color='#6200EE'><b>Latitude :</b><br></font>"
```

```
            + addresses.get(0).getLatitude()

));

//set lonitude

tv2.setText(Html.fromHtml(

      "<font color='#6200EE'><b>Longitude :</b><br></font>"

            + addresses.get(0).getLongitude()

));

// sset  country name

tv3.setText(Html.fromHtml(

      "<font color='#6200EE'><b>Country Name :</b><br></font>"

            + addresses.get(0).getCountryName()

));

//set Locality

tv4.setText(Html.fromHtml(

      "<font color='#6200EE'><b>Locality :</b><br></font>"

            + addresses.get(0).getLocality()

));

// set address

tv5.setText(Html.fromHtml(

      "<font color='#6200EE'><b>Address :</b><br></font>"
```

```
                        + addresses.get(0).getAddressLine(0)

                ));

            } catch (IOException e) {

                e.printStackTrace();

            }

        }

    }

});

}

}
```

Contacts Provider

The Contacts Provider is a powerful and flexible Android component that manages the device's central repository of data about people. The Contacts Provider is the source of data you see in the device's contacts application, and you can also access its data in your own application and transfer data between the device and online services.

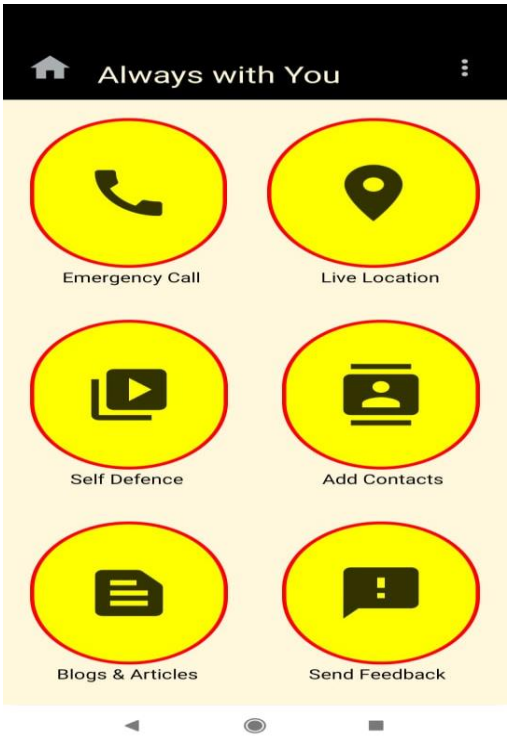
```
package com.example.womensafety;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.Toast;

public class Contact extends AppCompatActivity {

    Button saveContact;
```

```
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_contact);
        saveContact = findViewById(R.id.saveContactBtn);
        saveContact.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                Toast.makeText(Contact.this, "Contact saved successfully !", Toast.LENGTH_SHORT).show();
            }
        });
    }
}
```

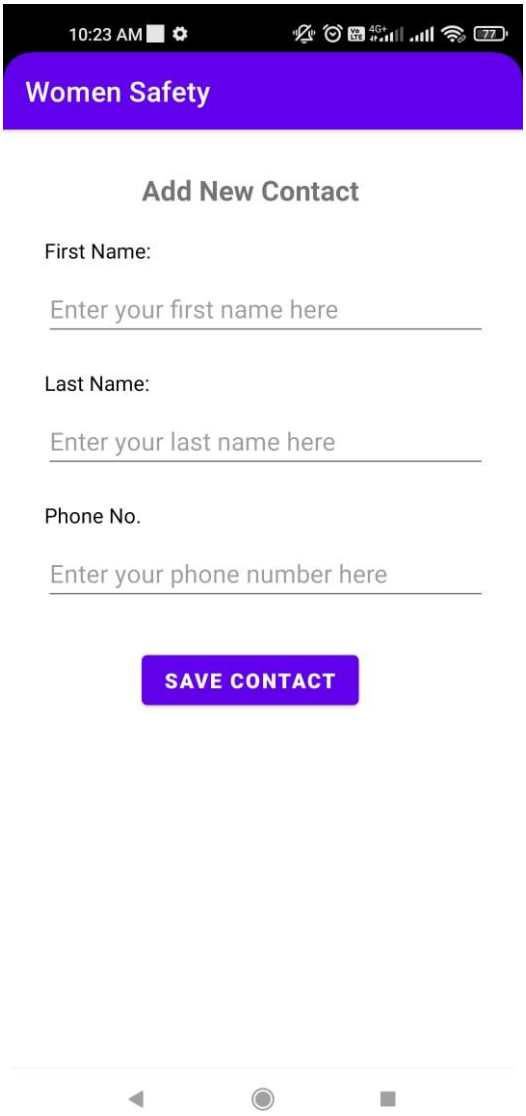
5.3 SCREENSHOTS

5.3.1 START SCREEN

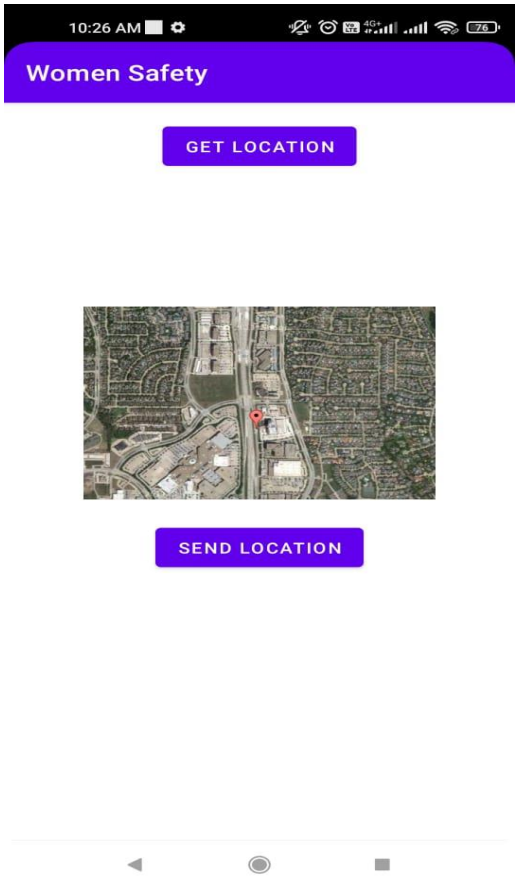
5.3.2 HOME PAGE



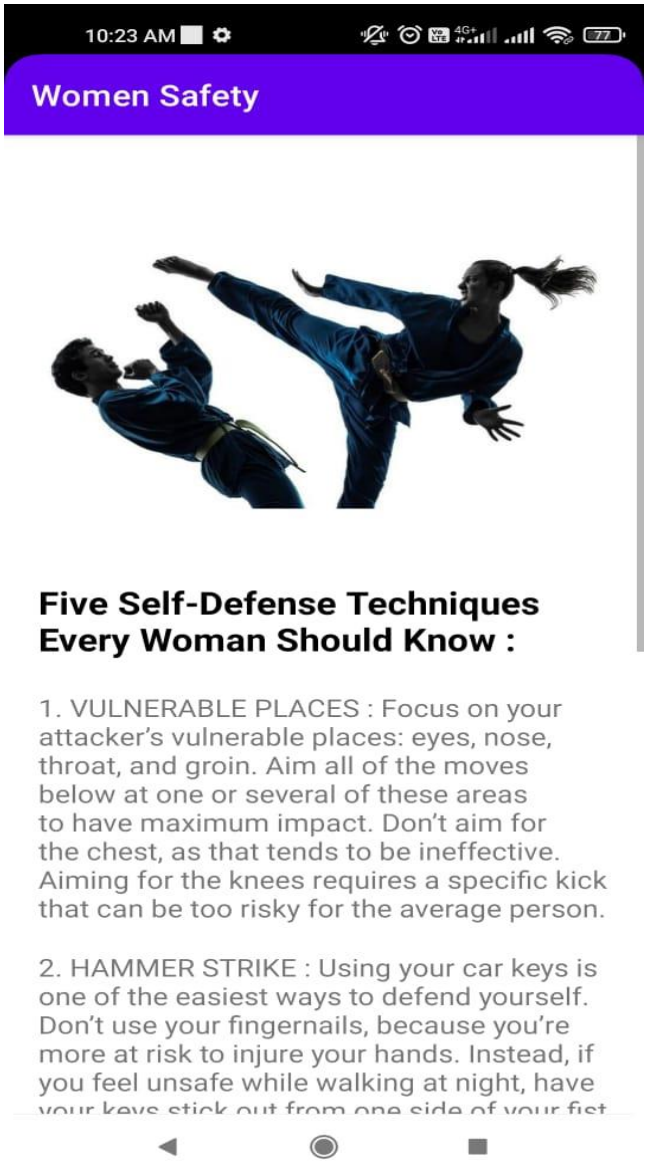
5.3.3 REGISTERING NEW CONTACT



5.3.4 GPS LOCATION

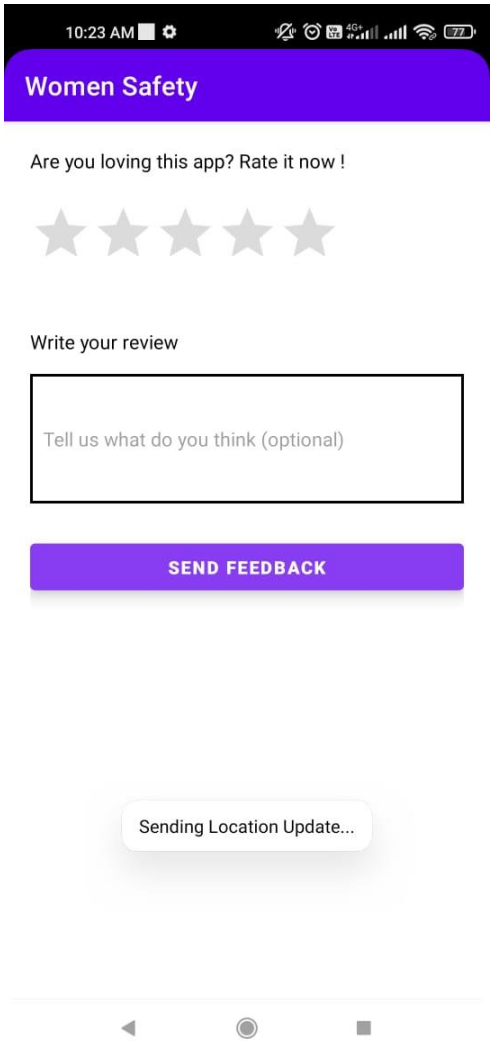


5.3.5 SELF DEFENCE BLOG



5.3.6 message received by one of the contacts

5.3.7 Feedback



TESTING

6.1 INTRODUCTION TO TESTING

Testing describes the process of running and executing software application or software product in order to find bugs in the software. Software testing can be defined as the process of validating and verifying that software program meets business needs and technical requirements that guided the design and development of the software product.

During testing the programs to be tested are executed with set of test cases and the output of program for the test cases is evaluated to determine if the program is performing as expected. Testing forms is the first in determining errors in the program. Once programs were tested individually then the system as a whole was tested. During testing the system is used experimentally to ensure that the software does not fail i.e. it will run according to its specification. The program executed to check for any syntax and logical errors. The errors are corrected and test is made to determine whether the program is doing what it is supposed to do.

There are generally four recognized levels of tests :

- Unit Testing
- Integration Testing
- System Testing
- Acceptance Testing

Unit testing : Testing of individual software components or modules. Typically done by the programmer or not by testers, as it requires detailed knowledge of the internal program design and code.

Integration testing : Testing of integrated modules to verify combined functionality after integration. Modules are typically code modules, individual applications, client and server applications on a network etc. This type of testing is especially relevant to client/server and distributed systems.

System Testing : System testing, or end-to-end testing, tests a completely integrated system to verify that it meets its requirements. Software testing should ensure that the program, as well as working as expected, does not also destroy or partially corrupt its operating environment or cause other processes within that environment to become inoperative (this includes not corrupting shared memory, not

consuming or locking up excessive resources and leaving any parallel processes unharmed by its presence).

Acceptance Testing : Normally this type of testing is done to verify if system meets the users specified requirements. User or customer do this testing to determine whether to accept application.

FUNCTIONAL TESTING

- The identification of functions that the software is expected to perform.
- The creation of input data based on the function's specifications.
- The determination of output based on the function's specifications.
- The comparison of actual and expected outputs.

STRUCTURAL TESTING

Structural test design techniques includes :

- Control flow Testing: - Whether the low of control of the code is in order .
- Data flow Testing: - When data flow between two blocks or within a block occurs. Is it running as needed of if any bugs or present?
- Branch Testing: - The test of branches and loops of the code is done.
- Path Testing: - It can test paths within a unit, paths between units during integration and between subsystem.

This type of testing is also called white box testing. Here, we check the code internally for flaws and bugs.

LEVELS OF TESTING

There are different levels of testing as follows :

Alpha Testing : There are three types of alpha testing namely -

- Unit Testing.
- Integration Testing (Top Down & Bottom Up)
- System Testing.

Acceptance Testing : Acceptance Testing is a formal testing conducted to determine whether a system satisfies its acceptance criteria.

There are two categories of acceptance testing :

- User Acceptance Testing
- Business Acceptance Testing

Beta Testing : It is also known as field testing. It is the second phase of software testing in which a sampling of the intended audience tries the product out. Goal of the beta testing is to place the application in the hands of real users in order to discover any flaws or issues from the user's perspective.

6.2 SAMPLE TEST CASES

Test case ID 1

Test case Description

Enter phone number

Input value

Accepted the phone number only once

Expected output

Redirected to next page

Result

Success

Test case ID 2

Test case Description

Enter phone number

Input value

Accepted the phone number if already exists

Expected output

Redirected to next page

Result

Failure

Test case ID 3

Test case Description

Buzzer pressing

Input value

Notifications only sent to registered contacts

Expected output

Redirected to next page

Result

Success

Test case ID 4

Test case Description

Buzzer pressing

Input value

Notifications are not sent ,if contacts are not registered

Expected output

Redirected to next page

Result

Failure

Test case ID 5

Test case Description

Enter user name

Input value

Enter user name as a character

Expected output

Redirected to next page

Result

Success

Test case ID 6

Test case Description

Enter user name

Input value

Enter user name as a digit

Expected output

Redirected to next page

Result

Failure

Test case ID 7

Test case Description

Check whether Accelerometer sensor is working or not

Input value

Send current location of the user

Expected output

Current location of the user

Result

Success

Test case ID 8

Test case Description

Check whether Accelerometer sensor is working or not

Input value

Less shaking intensity

Expected output

Current location

Result

failure

7. CONCLUSION

The “Andriod App for Women Security System” has been developed to satisfy all the proposed requirements. The process is maintained more simple and easy in ensuring the women safety. The system is highly scalable and user friendly. Almost all the system objectives have been met. The system has been tested under all criteria. The system minimizes the problem arising in the existing manual system and it ensures the immediate action to be taken when an unfavourable situation is encountered. The design of the database is flexible ensuring that the system can be implemented. It is implemented and gone through all validation. All phases of development were conceived using methodologies. User with little training can get the required report. The software executes successfully by fulfilling the objectives of the project. Further extensions to this system can be made required with minor modifications.

8. FUTURE ENHANCEMENT

There is scope for future development of this project. The Computer technology keeps finding new methods and technologies on a day to day basis. It is dynamic and not static. The skills which is prominent today will become obsolete in a few days. To keep in pace with the technical developments, the system may be additionally improved. So, it is not concluded. Yet it will improve with further augmentations. Augmentations can be done in an effectual manner. We can even apprise the same with further changes and can be integrated with minimal alteration. Thus the project is flexible and can be improved at anytime with more progressive features.

This mobile application is helpful in future when any problem arises in travelling or any kind of situations.

- As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment.
- Because it is based on object-oriented design, any further changes can be adaptable to desired environment
- .Based on the future security issues, security can be improved using emerging easily adaptable technologies.

REFERENCES

9. REFERENCES

[1]. Neil Smyth, “Android Studio Development Essentials”, Android 6 edition.

[2]. Ryan Hodson , “Android Programming Succinctly”, edition 3.

[3]. Wei-Meng Lee , “Beginning Android 4 Application Development”, Willey India Pvt., Ltd., 2012.

[4]. J.F. DiMarzio, “Android :A Programmers Guide”, First Edition, Tata Mcgraw-hill,2010.

[5]. Jason Morris, “Android User Interface Development”, Packet publication, 2011., “SQLite” Sams.

[6]. Jay A Kreibich, “Using SQLite”, O,,Reily Media, 2010.

[7]. http://www.android.com/

[8]. http://developer.android.com/guide/topics/ui/

[9]. Portio Research. Portio research mobile factbook 2011. Chippenham, UK. Retrieved from http://www.portioresearch.com/Portio%20Research%20Ltd%20Mobile%20Factbook%202011.pdf (201

[10]. Ofcom. Children and Parents: Media Use and Attitudes Report (2012).

[11]. Dennison, L., Morrison, L., Conway, G., & Yardley, L. (2013).Opportunities and Challenges for Smartphone Applications in Supporting Health Behavior Change: Qualitative Study, Journal of medical Internet research, 15(4): e86.

[12]. Qualcomm Wireless ReachTM, Vital Wave, and GSMA. Transforming Women‟s Livelihoods Through Mobile Broadband. Report, February 2014. Available at: http://www.qualcomm.com/mwomenreport