

DWIT COLLEGE
DEERWALK INSTITUTE OF TECHNOLOGY
Tribhuvan University
Institute of Science and Technology



REDALERT- A MOBILE BASED SOS APPLICATION

A PROJECT REPORT

Submitted to
Department of Computer Science and Information Technology
DWIT College

*In partial fulfillment of the requirements for the Bachelor's Degree in Computer Science
and Information Technology*

Submitted by
Asmita Bista
August, 2016

DWIT College
DEERWALK INSTITUTE OF TECHNOLOGY
Tribhuvan University

SUPERVISOR'S RECOMENDATION

I hereby recommend that this project prepared under my supervision by ASMITA BISTA entitled **“REDALERT- A MOBILE BASED SOS APPLICATION”** in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Information Technology be processed for the evaluation.

.....

Rituraj Lamsal

Lecturer

Deerwalk Institute of Technology

DWIT College

DWIT College
DEERWALK INSTITUTE OF TECHNOLOGY
Tribhuvan University

LETTER OF APPROVAL

This is to certify that this project prepared by ASMITA BISTA entitled “**REDALERT-A MOBILE BASED SOS APPLICATION**” in partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Information Technology has been well studied. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

<p>.....</p> <p>Rituraj Lamsal [Supervisor] Lecturer DWIT College</p>	<p>.....</p> <p>Hitesh Karki Chief Academic Officer DWIT College</p>
<p>.....</p> <p>Jagdish Bhatta [External Examiner] IOST, Tribhuvan University</p>	<p>.....</p> <p>Sarbin Sayami [Internal Examiner] Assistant Professor IOST, Tribhuvan University</p>

ACKNOWLEDGMENT

This project would not have come into being without the guidance of my supervisor Mr. Rituraj Lamsal. He has been very supportive and his feedback has greatly improved my work. He has given me many valuable suggestions, discussions and clarifications to this project. His encouragement for this whole work and co-operative behavior are sincerely acknowledged.

I would like to thank Bijaya Kumar Shrestha for providing the opportunity to carry out this project.

Hitesh Karki has been of great help during the writing stages of this project. His advices on report writing have been a good source to learn how to write scientific reports in an appropriate manner. He has also given tips on where to find appropriate research material.

I would also like to express sincere thanks to all of my friends and other teachers who helped me directly or indirectly during this project work.

Asmita Bista

TU Exam Roll no: 1795/069

STUDENT'S DECLARATION

I hereby declare that I am the only author of this work and that no sources other than the listed have been used in this work.

... ..

Asmita Bista

Date: August, 2016

ABSTRACT

Safety has become the basic concern for every one of us. Many unfortunate incidents have been taking place. Increasing number of rapes and murder has raised question mark for safety of every woman. It is necessary to keep ourselves safe at our own otherwise inform our loved ones about our condition.

In an emergency situation like attempt to rape, kidnapping, heart attack, accident, etc., we may not get enough time to make a call. And even if we did, they may not receive the call or they may be unavailable at that point of time. Trying to contact second person and more others of our contact list may not be possible in such emergency condition.

RedAlert is an android application which can be used in emergency condition like, accidents, medical emergency, safety risk, etc. It allows us to inform our family and friends about our emergency condition. It helps sending SMS to 5 nearest contacts added manually by the user.

TABLE OF CONTENTS

LETTER OF APPROVAL	i
ACKNOWLEDGMENT.....	ii
STUDENT’S DECLARATION	iii
ABSTRACT.....	iv
LIST OF FIGURES	vii
LIST OF TABLES.....	viii
LIST OF ABBREVIATIONS.....	ix
CHAPTER1: INTRODUCTION	1
1.1 Background	1
1.2 Problem Statement	1
1.3 Objective	2
1.4 Scope and Limitations.....	2
1.5 Report Organization.....	3
CHAPTER 2: REQUIREMENT ANALYSIS AND FEASIBILITY ANALYSIS	4
2.1 Literature Review.....	4
2.1.1 VithU app.....	4
2.1.2 Safetipin	4
2.3 Requirement Analysis.....	6
2.4 Feasibility Analysis.....	8
2.4.1 Technical feasibility	8
2.4.2 Operational feasibility.....	8
2.4.3 Schedule feasibility	8
CHAPTER 3: SYSTEM DESIGN.....	10
3.1 Methodology	10
3.2 System Design	12
3.2.1 Class diagram.....	12
3.2.2 Sequence diagram	13
3.2.3 State diagram	14
CHAPTER 4: IMPLEMENTATION AND TESTING	15

4.1 Implementation	15
4.1.1 Tools used	15
4.1.2 Description of main modules	15
4.2 Testing.....	17
CHAPTER 5: MAINTENANCE AND SUPPORT.....	20
5.1 Corrective Maintenance	20
5.2 Adaptive Maintenance	20
CHAPTER 6: CONCLUSION AND RECOMMENDATION	21
6.1 Conclusion	21
6.2 Recommendations.....	21
APPENDIX I	22
REFERENCES	28

LIST OF FIGURES

Figure 1- Outline of the document.....	3
Figure 2- Use case diagram of RedAlert.....	7
Figure 3- Gantt chart of RedAlert.....	9
Figure 4- Flow chart of KNN algorithm.....	11
Figure 5- Class diagram of RedAlert.....	12
Figure 6- Sequence diagram of RedAlert.....	13
Figure 7- State diagram of RedAlert.....	14

LIST OF TABLES

Table 1-Functional and non-functional requirement	6
Table 2-Test case 1	17
Table 3-Test case 2	18
Table 4-Test case 3	18
Table 5-Test case 4	19
Table 6-Test case 5	19

LIST OF ABBREVIATIONS

KNN – K-Nearest Neighborhood

GSM – Global System for Mobile

GPS – Global Positioning System

API – Application Program Interface

IDE – Integrated Development Environment

SMS – Short Message Service

CHAPTER1: INTRODUCTION

1.1 Background

Safety matters a lot for us whether we are at home or outside the home. We fear whenever we get to know that we are in danger. It is essential to keep ourselves safe. Our personal safety is our personal responsibility. Safety should be taken very seriously in order to avoid any incidents which may even take our life.

The number of rape and kidnapping cases has been increasing day by day. We heard about the Neha Pun case. We still cannot forget the Khyati Shrestha case few years back. These incidents show the safety line of our country. If only they could send any message to their family or friends when they got hint about the danger, they wouldn't have faced such painful death.

The emergency situation also includes an emergency health conditions like heart attack, accidents, etc. The RedAlert focuses on helping us at the time of emergency. It helps us sending 'HelpMe' message along with our current location to 5 of our added contacts on one click.

1.2 Problem Statement

According to Nepal Police, 1326 incidents of rape were reported in the year 2013/014 and there were 677 incidents of rape reported in the previous year. In the last six years, the number of incidents related to rape continued to increase, with 3796 incidents. There were 381, 376, 481, 555, 677 and 1,276 rape cases registered in Nepal police in the year 2009, 2010, 2011, 2012, 2013 and 2014 respectively[1].

The rate of kidnapping and then killing them has also increased day by day. The safety status does not seem that good. If we fall in any emergency health condition like heart attack or road accident and we don't have enough time to make a call, we may even die in such condition. And even if we made a call, our contacts may not receive the call or they may be unavailable at that point of time.

RedAlert is an application easily installed onto android mobile phones and tablets. It is focused on adding ten contacts and sending SMS to nearest five of them with just a click of button. It helps saving our time for typing message in an emergency condition.

1.3 Objective

The main objectives of the application are:

- To alert our family/friends about our emergency condition.
- To provide the rising need of a simple yet efficient safety app for women.

1.4 Scope and Limitations

The scope of the applications is as follows:

- Nearest five locations can be calculated from given 10 locations.
- Contacts can be changed (deleted and added) as per our contact preferences.

The limitation of the application is as follows:

- Does not have button sequence.
- Manual entry of contacts.
- Does not track contacts in real time as their GPS may not be activated.

1.5 Report Organization

The remaining part of the document is organized and represented in project block diagram as in Figure 1 given below:

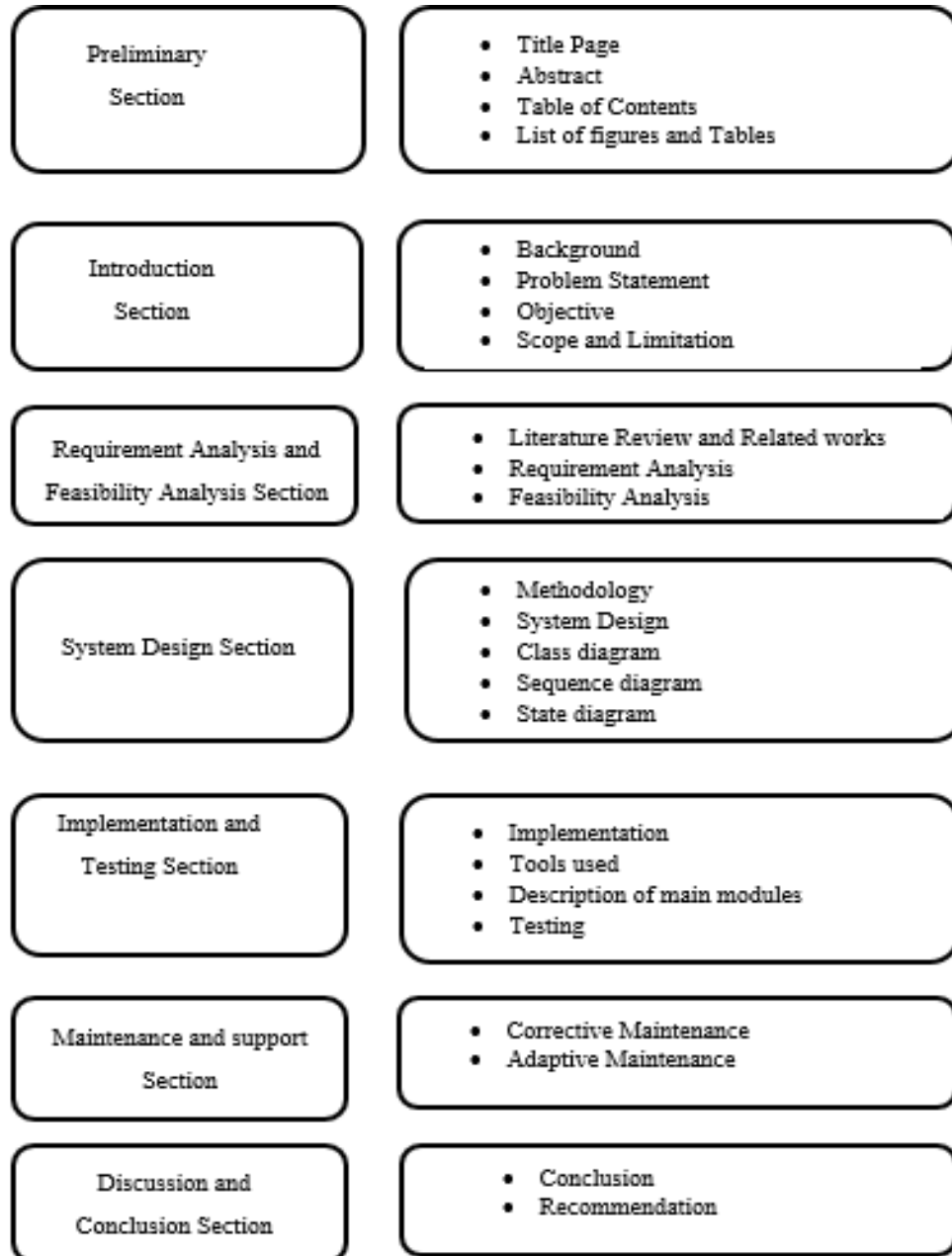


Figure 1- Outline of the document

CHAPTER 2: REQUIREMENT ANALYSIS AND FEASIBILITY ANALYSIS

2.1 Literature Review

For the development of this project, various journals and surveys were studied. Similar projects developed by various developers were found. Some of the apps are: Safetipin, Women safety, VithU App, etc.

2.1.1 VithU app

It allows to send an SOS alert to contacts with the pressing of power button twice. It marks the nearby location using GPS and sends the message to the family members whose numbers are saved in the database. In order to start VithU app, we have to double tap on the app icon and then only the messages will be send to the family members [2].

The drawback of this application is that a person may unknowingly double tap on the app and the false message will get send to the members that we have stored in the database.

2.1.2 Safetipin

Safetipin is the safety application that measures nine parameters including lighting, the state of the walk path, the presence of people and specifically women on the streets, visibility, security and nearness of public transport. Each audit appears as a pin on the map and is used

to compute the safety score of an area [3]. It has special features like GPS tracking, emergency contact numbers, direction to safe location, etc. It also pins the safe areas and also allows users to pin unsafe areas and help others. For the city authority, Safetipin provides large-scale data that is crowdsourced and provides a platform for interaction with citizens on their safety concern [4].

The app uses ‘pins’ to indicate the level of safety — green for safe, amber for less safe and red for unsafe. It allows the user to record instances of harassment or hazards and gives location information like timings of shops, availability of auto rickshaws and general directions. [3]

RedAlert is designed to send alert message to 5 of our contacts. It calculates the nearest location among our contact list and alert message is sent only to 5 of them. It does not facilitate double tap of the application. User must open the application in order to send message. It allows deleting contacts and adding new ones. Location of contacts can be manually set in this application whereas user’s location can be tracked with the help of GPS.

2.3 Requirement Analysis

A requirement analysis was done on the product and the following data was obtained:

The functional and non-functional requirements addressed by the application are listed below in the table:

Table 1-Functional and non-functional requirement

S.N	Functional Requirement	Non-functional Requirement
1.	User Registration	A user can register using a valid phone number. The username should contain at least 3 characters and should be valid name. The phone number provided should be valid.
2.	Add contacts	User can add at most 10 contacts for sending message. The details for adding contacts include id, name, phone number and location of contacts.
3.	View contacts	User can view name and phone number of contacts.
4.	Delete contacts	User can delete previously added contacts and add new contacts.
3.	Send message	User can send alert message along with location to any five of the nearest contacts.

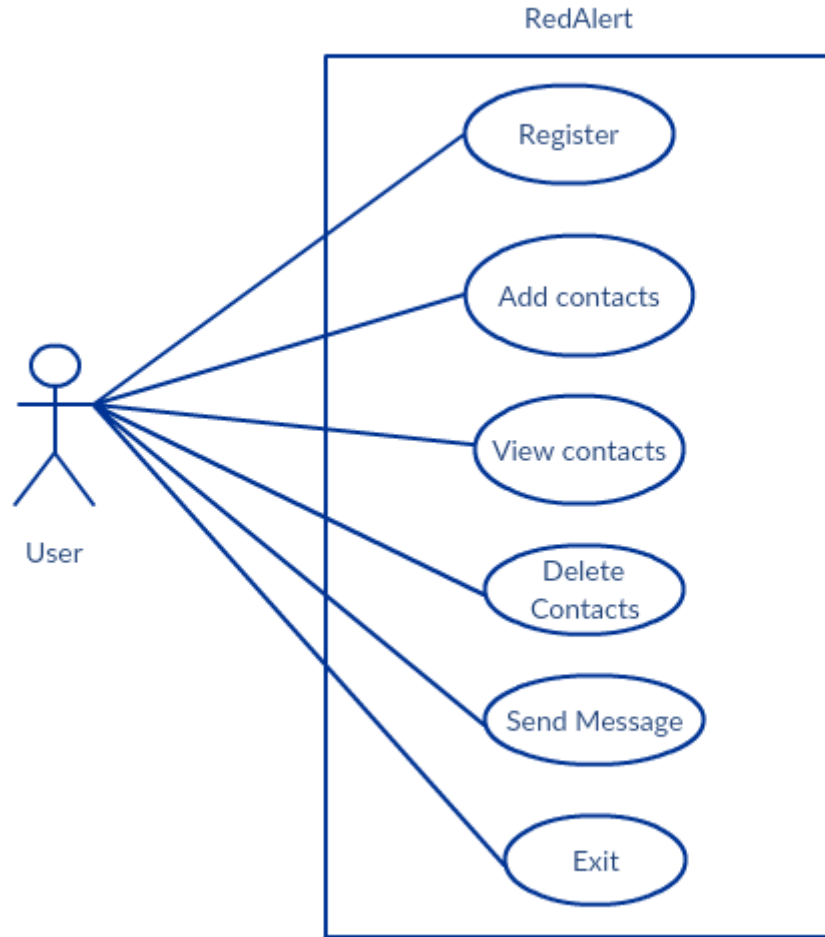


Figure 2- Use case diagram of RedAlert

As shown in the Figure 2, the end user is allowed to create a new account by registering to the application. The phone number must be unique.

The user can add any 10 contacts to whom they want to send alert SMS. The user need to provide name, phone number along with location of each contacts. The user can send message to nearest 5 contacts from those 10 added contacts.

2.4 Feasibility Analysis

The following result was obtained while performing a feasibility analysis:

2.4.1 Technical feasibility

RedAlert is a mobile application based on android studio. It uses SQLite at the back-end. It uses GPS for retrieving user's current location. It uses google places API for setting location of contacts. It requires a server, client, and internet connection to function properly. It supports only Android Operating System. All of the technology required by the application are available and can be accessed freely, hence it was determined technically feasible.

2.4.2 Operational feasibility

The application works under 2-tier architecture where end user sends message to 5 nearest contacts. The server keeps record of all contacts and their phone numbers. The application sends message from anywhere if sim network exists. It requires internet connection only for setting location of contacts at the beginning. It does not necessarily requires internet connection for sending SMS. So, it is operationally feasible.

2.4.3 Schedule feasibility

This project has been completed on time. It took about 14 weeks for the completion of the project. Overall the plans and procedures were carried out as per the schedule. So, it is feasible in terms of schedule.

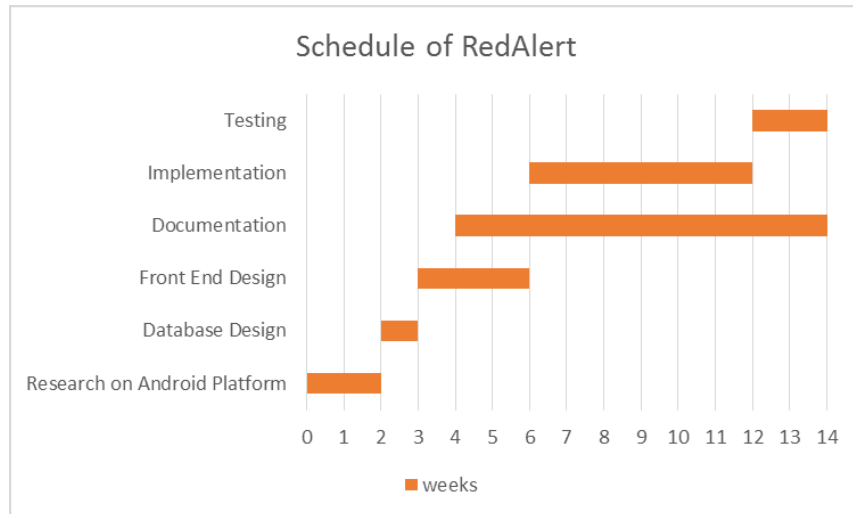


Figure 3- Gantt chart of RedAlert

CHAPTER 3: SYSTEM DESIGN

3.1 Methodology

This application was started with the research on android development. Different documents were studied for the development of android application. Similar applications were studied and used. For nearest neighbor problem, different algorithms were studied. K- Nearest Neighbor Algorithm is used in the application for finding nearest location of contacts.

KNN is a method to classify objects based on the closest matching entries obtained from training data. Training data has several attributes that represents its characteristics, this is modeled as many-dimensional space representation. The space is divided into sections based on the classification of training data. An entry in this space is marked if the class c is the entry with the closest entry with in a K nearest neighbor range [5].

The algorithm stores unique features represented as vectors and the class of each data. In the classification phase, the whole distance of training data are calculated against an object to be classified. Once the result obtained, the distances are then sorted ascendingly to obtain the closest to the farthest similarity. Then, from the sorted data, it is selected in a number of K to obtain K number of stores entries with the closest distances.

The steps of computing K-nearest neighbors (KNN) algorithm are:

- a. Determine the parameter K . (K = number of nearest neighbors)
- b. Calculate the distance between the query instance and all the samples using Euclidean distance formula.

$$d_{xy} = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

- c. Sort the distance and determine nearest neighbors based on the K th minimum distance.

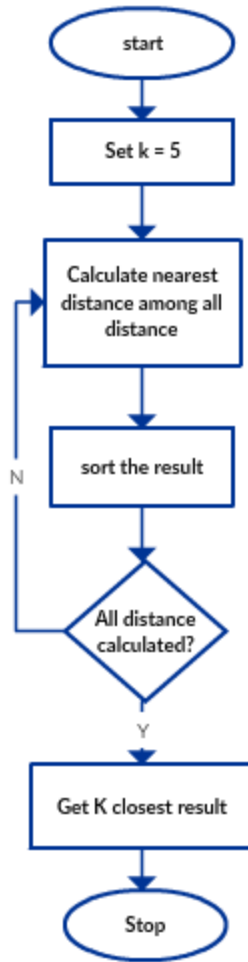


Figure 4- Flow chart of KNN algorithm

In this application, KNN algorithm is used for finding the nearest five location. User sets location of all those 10 contacts manually with Google Places API. With the algorithm, all the distances are calculated. They are then sorted in ascending order. The Kth nearest location i.e., 5 nearest locations are then retrieved from the list.

3.2 System Design

3.2.1 Class diagram

The class diagram of this application includes three activity classes altogether. First activity class is LoginActivity class where user has to enter valid name and phone number for login into the system. Second activity class is ContactActivity class where adding of contacts, deleting of contacts and viewing the details of contacts can be done. After that last activity class is FinalActivity class where KNN algorithm is implemented. This class includes a button which helps in sending SMS to targeted contacts. All these activity classes have instance of Database class required for accessing SQLite database.

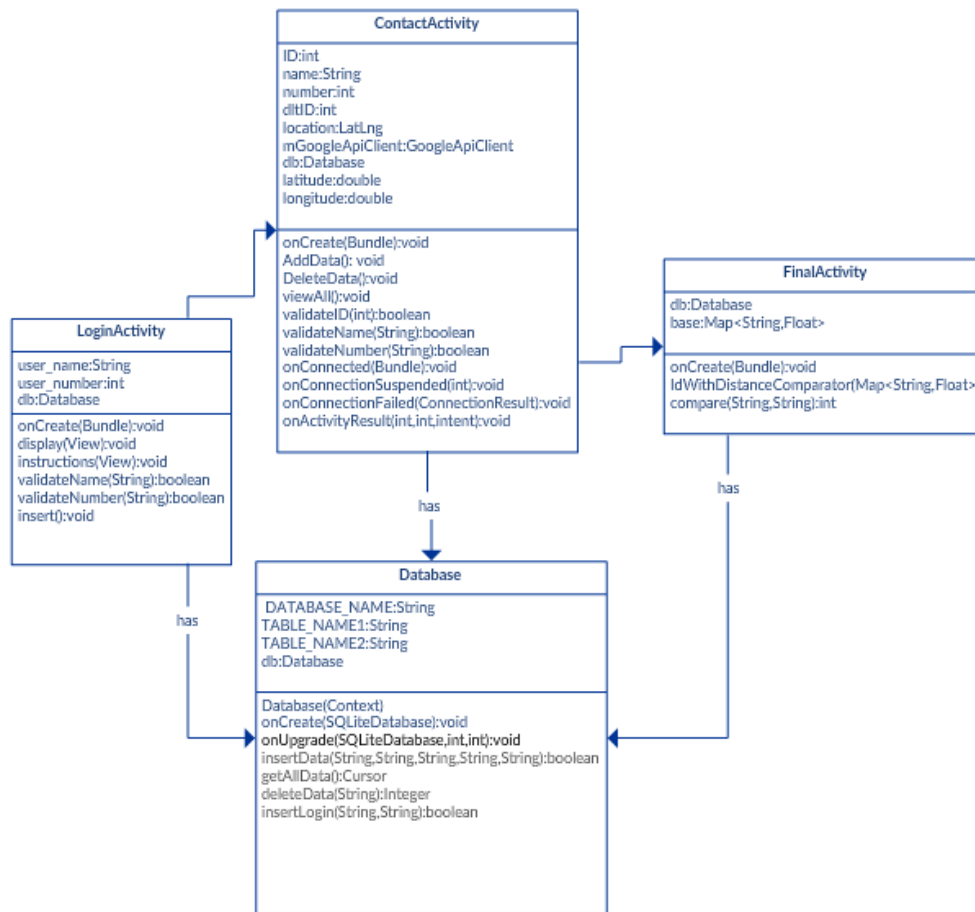


Figure 5- Class diagram of RedAlert

3.2.2 Sequence diagram

Firstly, user needs to register to the application with valid phone number. If the provided phone number is not valid, user is not allowed to log into the system.

After successful login to the system, user is allowed to add any 10 contacts. Those contacts are then stored in the database. These contacts can be deleted and new contacts can be added.

Whenever user requires to send message, the system selects any five of those contacts out of 10 using KNN algorithm and SMS is sent to only those nearest five contacts.

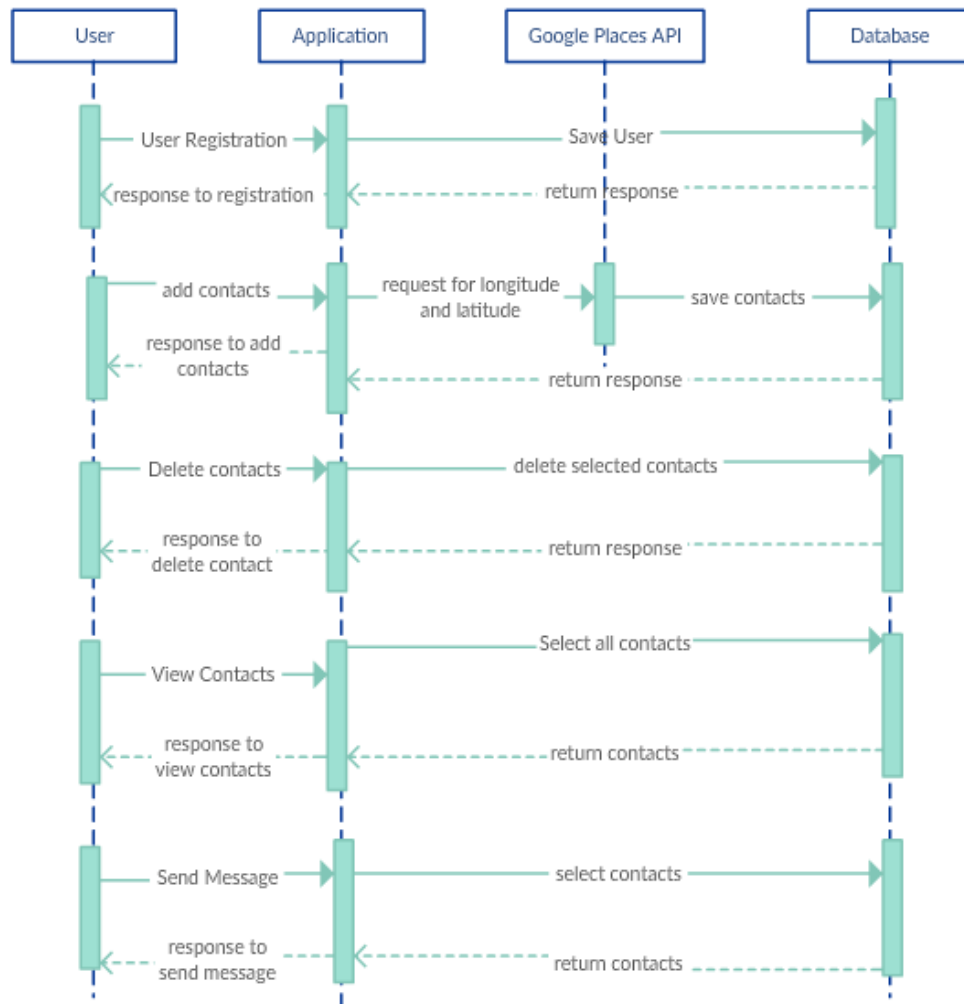


Figure 6- Sequence diagram of RedAlert

3.2.3 State diagram

State diagram of RedAlert is explained below in the figure. User has to register to the system on the first opening of application. After that user is required to add 10 contacts in order to send alert message. After that whenever user tries to send message in emergency condition, message is sent to nearest 5 contacts among those 10 contacts.

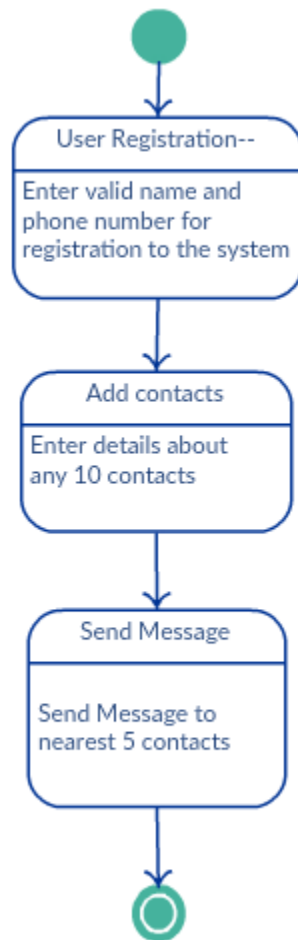


Figure 7- State diagram of RedAlert

CHAPTER 4: IMPLEMENTATION AND TESTING

4.1 Implementation

4.1.1 Tools used

Android studio was used for developing this application. It is as IntelliJ IDEA based IDE and declared by Google as the official IDE for android application development [6]. SQLite offered the facility of database. It uses GPS for tracking user's own location. Reverse geocoding is done for converting longitude and latitude into readable address.

4.1.2 Description of main modules

SMS Manager API

For Sending SMS, it uses SMS Manager API. It manages SMS operations such as sending data, text, and pdu SMS messages. In android development, it is necessary to allow permission for sending SMS. For this, we need SEND_SMS permission:

```
<uses-permission android:name="android.permission.SEND_SMS" />
```

After that following code is used to send sms,

```
SmsManager manager = SmsManager.getDefault();  
manager.sendTextMessage(number, null, message, null, null);
```

The number is retrieved from the database depending on the shortest distance as processed by algorithm.

Since, it uses GSM mobile phone service, it uses main balance available in sim card for sending message.

Google Places API

Google places API is used for adding location of contacts. It searches location from given input and provides longitude and latitude as an output.

```
mGoogleApiClient=new
GoogleApiClient.Builder(this).addApi(Places.GEO_DATA_API).enableAutoManage(this,th
is).build();

mGoogleApiClient.connect();

PlaceAutocompleteFragment autocompleteFragment = (PlaceAutocompleteFragment)
getFragmentManager().findFragmentById(R.id.place_autocomplete_fragment);

AutocompleteFilter typeFilter = new
AutocompleteFilter.Builder().setTypeFilter(AutocompleteFilter.TYPE_FILTER_ADDRES).
build();

autocompleteFragment.setFilter(typeFilter);

autocompleteFragment.setOnPlaceSelectedListener(new PlaceSelectionListener() {

@Override

public void onPlaceSelected(Place place) {

System.out.println("Place" + place.getName());

LatLng location = place.getLatLng();

latitude = location.latitude;
```

```
longitude = location.longitude;  
  
}
```

4.2 Testing

The following testing was done.

Table 2-Test case 1

Test case number	1
Title	Login
Precondition	The landing page of application is opened in the mobile.
Test steps	<ul style="list-style-type: none">i. Enter valid nameii. Enter valid mobile number.iii. Click save button.
Expected outcome	Should allow to login.
Actual Outcome	Login successful.

Table 3-Test case 2

Test case number	2
Title	Adding contacts
Precondition	User has successfully login into the system and the mobile device must have internet connection.
Test steps	<ul style="list-style-type: none">i. Enter ID.ii. Enter valid name.iii. Enter valid mobile number.iv. Enter location of user via autocomplete fragment.v. Click add button.
Expected outcome	Contact detail should be inserted and “data inserted” message should be displayed.
Actual Outcome	Contact detail is inserted and “data inserted” message is displayed.

Table 4-Test case 3

Test case number	3
Title	Viewing contacts
Precondition	User has successfully login into the system.
Test steps	Click view button.
Expected outcome	All inserted contacts should be displayed.
Actual Outcome	All inserted contacts is displayed.

Table 5-Test case 4

Test case number	4
Title	Deleting contacts
Precondition	User has successfully login into the system.
Test steps	i. Insert ID for deleting contact. ii. Click delete button.
Expected outcome	Selected contact should be deleted and “data deleted” message should be displayed.
Actual Outcome	Selected contact is deleted and “data deleted” message is displayed.

Table 6-Test case 5

Test case number	5
Title	Sending SMS
Precondition	i. User has successfully added 10 contacts with valid phone number and their location. ii. The GPS of user’s mobile device is on.
Test steps	Click send message button.
Expected outcome	“HelpMe. I’m at (user’s current location)” message should be sent to nearest five contacts.
Actual Outcome	“HelpMe. I’m at: Jaya Bageshwori Road, Kathmandu 44600, Nepal” message is sent to nearest five contacts.

CHAPTER 5: MAINTENANCE AND SUPPORT

After using and testing of this system I am looking forward to further enhance its features and update this system for better performance. Regular support and maintenance facility will be provided to the users. Some of the strategies are:

5.1 Corrective Maintenance

As application could be placed in google play store for public use. If there occurs unresolved issues and if a user complains about it, the maintenance has to be done.

5.2 Adaptive Maintenance

Adaptive maintenance can be done in order to achieve some modification in certain functionalities if there occurs any changes in business trends and needs. Maintenance can be done for upgrading the system to achieve better performance and better functionalities in this application.

CHAPTER 6: CONCLUSION AND RECOMMENDATION

6.1 Conclusion

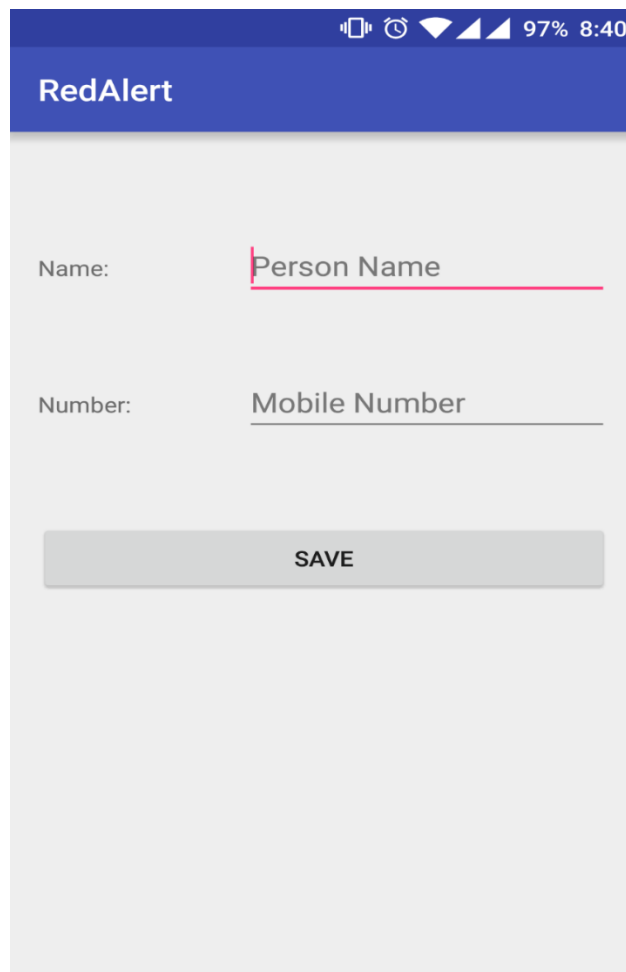
This Final Year Project was a great opportunity for me to discover new fields and ways of working of android application. The application RedAlert was developed successfully with the feature of sending SMS to nearest contacts in emergency condition. With this mobile app, we can alert our contacts that we are not safe.

6.2 Recommendations

As the technology emerges, it is possible to upgrade the system and can be adaptable to desired environment. The application can be further taken with the feature of button sequence. Real time tracking of contact location can be done in future.

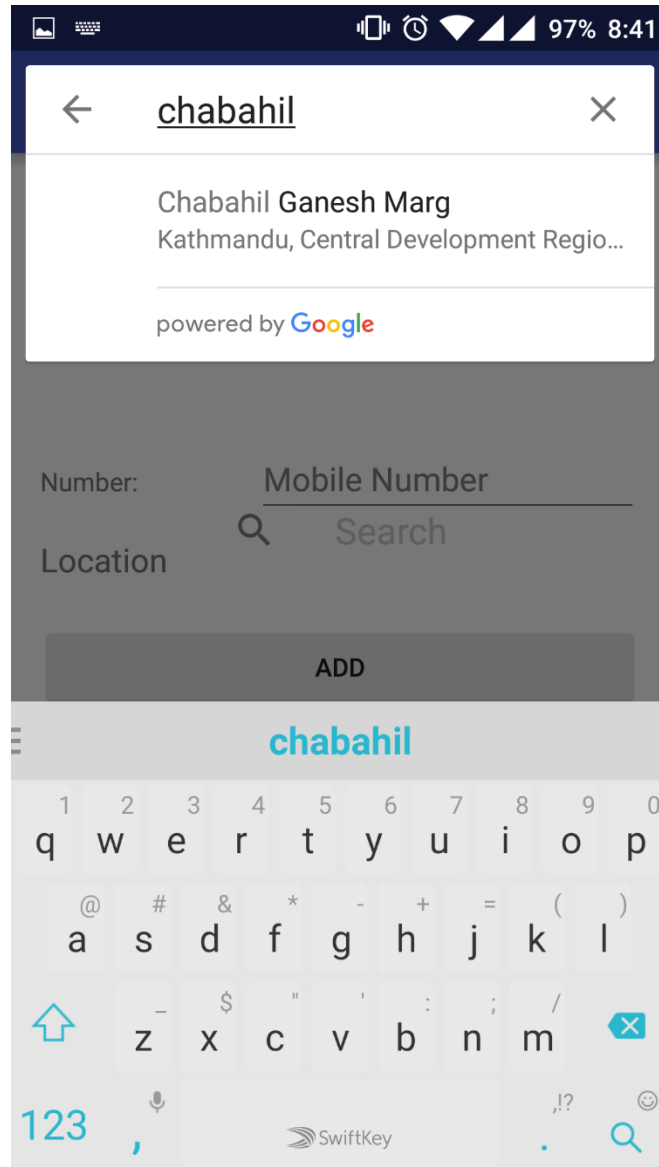
APPENDIX I

Login Page



The screenshot displays the login interface of the RedAlert mobile application. At the top, a dark blue header bar contains the app's name, "RedAlert", in white text. Above this header, the Android status bar is visible, showing icons for vibration, alarm, Wi-Fi, and cellular signal, along with a 97% battery level and the time 8:40. The main content area has a light gray background. It features two input fields: the first is labeled "Name:" and contains the placeholder text "Person Name" with a pink underline; the second is labeled "Number:" and contains the placeholder text "Mobile Number" with a gray underline. Below these fields is a wide, light gray button with the word "SAVE" in black capital letters.

Setting location of contact



Adding contact

The image shows a mobile application interface titled "RedAlert" in a blue header bar. The status bar at the top displays icons for signal, alarm, Wi-Fi, and battery, along with "97% 8:42". The form contains the following fields and controls:

- ID:** A text input field containing the value "1".
- Name:** A text input field containing the value "Anju".
- Number:** A text input field containing the value "9805862224".
- Location:** A text input field containing the value "Chabal". To the left of the text is a magnifying glass icon, and to the right is a close icon (X).

Below the input fields are three stacked buttons: "ADD", "VIEW", and "DELETE". At the bottom of the screen, there is a horizontal line above a "Data Inserted" message in a dark rounded rectangle. Below this message are two buttons: "BACK" on the left and "NEXT" on the right.

Deleting contact

RedAlert

ID

Name:

Number:

Location

ADD

VIEW

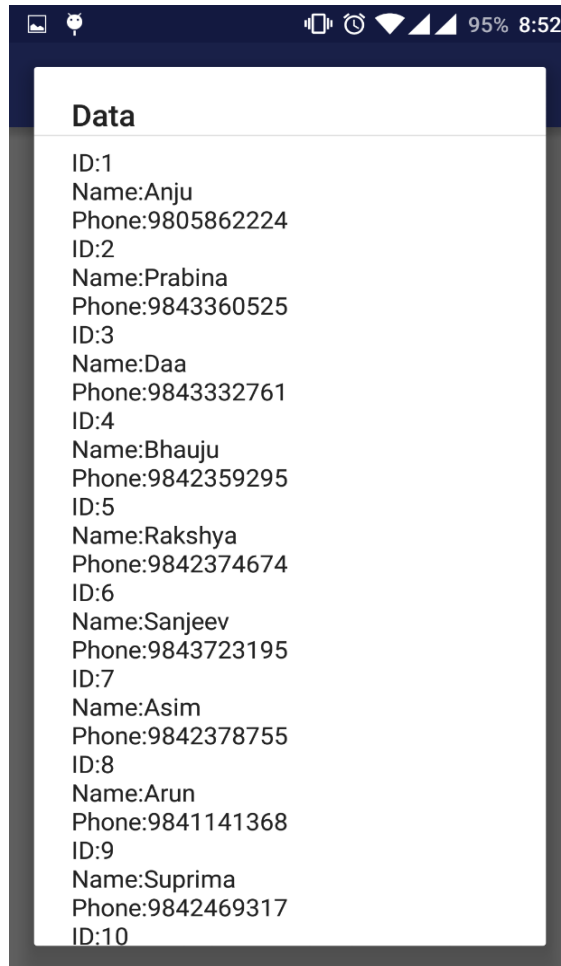
DELETE

1

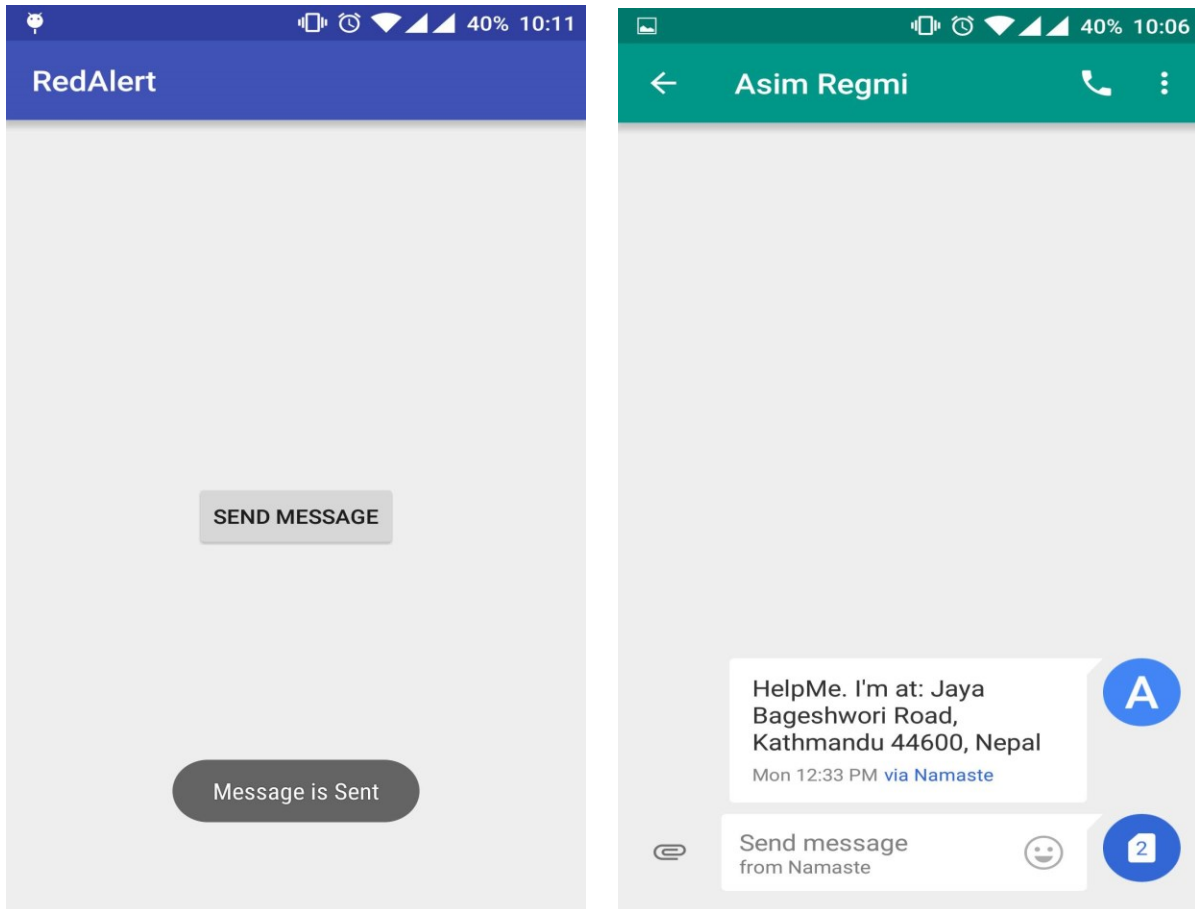
Data Deleted

BACK NEXT

Viewing Contacts



Sending message



REFERENCES

- [1]. "VIOLENCE AGAINST WOMEN: Alarms Ringing | Newspotlight Nepal News Magazine". Spotlightnepal.com. N.p., 2014. Web. Aug. 2016.
- [2]. "Vithu App Specially For GIRLS SAFETY". Yup & Nope. N.p., 2013. Web. 12 Sept. 2016.
- [3]. Menon, Meena. "'Safetipin' Audits Show Poor Security For Women". *The Hindu*. N.p., 2015. Web. Aug. 2016.
- [4]. Panchbhai, Tejas. Emergency App Using Voice Recognition. 1st ed. www.IJCAT.org, 2014. Print.
- [5]. Thakare, R.T. "ANDROID BASED SOS EMERGENCY ALERT BUTTON". International Journal of Research In Science & Engineering I (2014): n. pag. Print.
- [6]. "Android App Tutorial Intellij". Google Docs. N.p., 2016. Web. 11 Sept. 2016.