

SOS (SAVE OUR SOULS)

AN EMERGENCY ANDROID APPLICATION

A Project Submitted in Partial Fulfillment of the Requirements for the Degree of
Bachelor of Science in Computer Science & Engineering of the University of Asia
Pacific

By

Tania Akter Reg: 15201014	Pratik Paul Akash Reg: 15201023
Md. Adib Hossain Reg: 15201027	Tofael Parvez Reg: 15201037

Supervised By

Tanjina Helaly
Assistant Professor



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
UNIVERSITY OF ASIA PACIFIC

September 2019

DECLARATION

We, hereby, declare that the work presented in this Project is the outcome of the investigation performed by us under the supervision of Tanjina Helaly, Assistant Professor, Department of Computer Science & Engineering, University of Asia Pacific. We also declare that no part of this Project and thereof has been or is being submitted elsewhere for the award of any degree or Diploma.

Signature

Countersigned

(Tania Akter)

Tanjina Helaly
Supervisor

(Pratik Paul Akash)

(Md. Adib Hossain)

(Tofael Parvez)

ACKNOWLEDGEMENTS

First, we would like to thank the almighty ALLAH. Today we are successful in completing our work with such ease because He gave us the ability, chance, and cooperating supervisor.

We would like to take the opportunity to express our gratitude to Tanjina Helaly, our respected supervisor. Although she was always loaded with several other activities, she gave us more than enough time in this work. She not only gave us time but also proper guidance and valuable advice whenever we faced with some difficulties. Her comments and guidance helped us in preparing our project report.

We are also thankful to our teachers who helped us in a number of ways by providing various resources and moral support.

Finally yet importantly, We would like to express our thanks to our friends & families as well. They have help me to overcome the difficulties face on this project.

ABSTRACT

This project is about a mobile application for Location Tracking Service and Emergency, which works, in Android platform. It is a combination of existing application and enhancement to this project. This application is uses the GPS-service to track users' travelling status and allow them to use whistle when they are in danger. The focus in this project is to protect and improve personal safety. Hence, the goal of this project is to let user to get help in fastest and convenient way. Agile methodology used in this project to ensure testing conducted in every module. It provides opportunity to reassess throughout the development lifecycle to make sure the entire project meets the expected outcome and operate well. The final system has said to be bug fixed after few times of refining and debugging.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS.....	i
ABSTRACT.....	ii
LIST OF FIGURES.....	v
LIST OF TABLES.....	vi
CHAPTER – 1 INTRODUCTION.....	1
1.1 BACKGROUND	1
1.2 PROBLEM STATEMENT	1
1.2.1 High Failure Rate of Sensitivity.....	2
1.2.2 Request Help with Only One Step, Fast and Immediate	2
1.3 MOTIVATION.....	2
1.4 PROJECT OBJECTIVE.....	2
1.5 IMPACT AND CONTRIBUTION.....	3
1.6 PROJECT SCOPE.....	4
1.7 STRUCTURE OF THE REPORT	4
CHAPTER – 2 LITERATURE REVIEW	5
2.1 EXISTING EMERGENCY APPLICATION.....	5
2.1.1 BD 999	5
2.1.2 Emergency SOS Safety Alert.....	6
2.1.3 bSafe.....	7
2.1.4 Safety – SOS.....	8
CHAPTER – 3 EXPERIMENTAL WORK.....	9
3.1 WORKING PROCEDURE.....	9
3.1.1 Grant Permission.....	9
3.1.2 Sign UP.....	9
3.1.3 Sign IN	10
3.1.4 Main Screen	10
3.2 WORKING MODEL	10
3.3 KEY FEATURES	11
3.4 FLOWCHART	12

CHAPTER – 4 IMPLEMENTATION.....	13
4.1 ENVIRONMENT	13
4.1.1 Tools.....	13
4.1.2 Language	14
4.2 ENVIRONMENT SETUP	15
4.2.1 Steps	15
4.2.2 Installed Android Studio.....	15
4.2.3 Installed JDK	16
4.2.4 Installed SDK Tools	17
4.3 PROGRAMMING.....	18
4.3.1 Java Activity	18
4.3.2 XML Activity.....	18
4.3.3 JSON Generated Code for Firebase	18
4.3.4 SHA-1 Generated Serial for Firebase.....	20
4.4 FIREBASE STRUCTURE.....	20
4.5 FIREBASE DATABASE TABLE	21
CHAPTER – 5 APPLICATION INTERFACE.....	22
5.1 APPLICATION INTERFACE	22
5.1.1 Sign UP.....	22
5.1.2 Sign IN	23
5.1.3 Main Screen	24
5.1.4 Other Buttons.....	25
5.1.5 Cell Phone SMS Screen.....	26
CHAPTER – 6 CONCLUSION	27
6.1 IMPLEMENTATION ISSUES	27
6.2 LIMITATIONS	27
6.3 FUTURE WORK.....	27
APPENDIX.....	28
REFERENCES.....	55

LIST OF FIGURES

Figure No.	Figure Name	Page No.
2.1	Existing Application BD 999	5
2.2	Existing Application Emergency SOS Safety Alert	6
2.3	Existing Application bSafe	7
2.4	Existing Application Safety SOS	8
3.1	Application Working Model	10
4.1	Installed Android Studio Environment	15
4.2	Installed Android Studio Environment	16
4.3	Installed JDK	16
4.4	Installed SDK Tools	17
4.5	Firebase Structure	20
5.1	Sign UP	22
5.2	Sign IN	23
5.3	Main Screen	24
5.4	Other Buttons	25
5.5	Other Buttons	25
5.6	Other Buttons	25
5.7	Cell Phone SMS Screen	26

[N.B. 2.1 means First Figure of Chapter 2. Similarly, 4.4 means fourth figure of chapter 4.]

LIST OF TABLES

Table No.	Table Name	Page No.
4.1	Firebase Database Table 1	21
4.2	Firebase Database Table 2	21

[N.B. 4.1 means First Table of Chapter 4. Similarly, 4.2 means second figure of chapter 4.]

CHAPTER - 1

INTRODUCTION

SOS is an Emergency base Android Application which we can use when we are in danger or any unexpected situation. Now a days anyone can be in an unexpected situation anytime, So SOS is an application, with this we can inform our friends or Emergency Units.

1.1 BACKGROUND

In this era of crime, personal safety problem is getting worse to worst. There are many cases happened such as robbery, burglary, snatching, rapping etc. These cases will seriously affect social stability and could reduce people step out their door. Furthermore, everyone is worrying about their personal safety and their love one, especially their family and beloved partner. For some of the people, they tend to worry other people safety more than they do.

Finally yet importantly, no matter older generation or younger generation, they may face some emergency case sometimes, for example, accidentally fall down, heart attack, kids get kidnapped etc. By using application, they can request for instant help immediately.

1.2 PROBLEM STATEMENT

Public fear of crime is referring to the fear that citizens scaring as they will be one of the next victim of crime. Hence, some family or parent has limit their children or old people steps out the door. Besides that, when people wants to hangout for gathering or to the garden to release stress and relax, they have to worry about their personal safety including their friends.

There are many applications available throughout many countries but only a few in Bangladesh. This application is to reduce the criminal cases and improve personal safety purpose, but there is still some dissatisfaction and imperfection to the user.

1.2.1 High Failure Rate of Sensitivity

Most of the similar application use shake or vibrate to detect emergency alert, but there are some problems for that, once phone accidentally fall down or get knocked, vibration is detected and it will trigger emergency alert and sent SMS to emergency list. It is quite troublesome to notify emergency contact list that is just a mistake. Furthermore, some application may need several shake patterns to trigger that emergency alert. This may lead to failure when user unfortunately meet robber and they point their gun to user's head.

1.2.2 Request Help with Only One Step, Fast and Immediate

In normal practices, when someone is in trouble or emergency, they only able to contact at most one people at one time and need more than one-step to request for help. For instance, when someone is involved in a terrible accident (may has severe injuries) on the way, and need emergency help from friends, what the person can do is contact a friend for help at that time.

1.3 MOTIVATION

The purpose of develop this application is to solve the personal safety against public fear problem in Bangladesh. However, there are several existing applications in market, but all of them have unsatisfied problems. This project would improve and make some innovation to help those users to prevent unhappy case happen and reduce their public fear when they hang out with friends or family.

1.4 PROJECT OBJECTIVE

1. To protect personal safety and get emergency helpline with instant help by using the application through one S.O.S. button when they need help on their way home.

- 1.1 To enable user get help faster. User can get help faster than previous time by using this application, user can broadcast their situation now to their friends in the emergency contact list on the spot.
2. To check and track their gathering member's status, approximately arrival time to get to their destination or home.
3. To remind user to notify friends or family in convenient, easier and fastest way. User can notify friends or family using the pre-set message when they reached home safety with one click of button or pop out dialog.
4. To authenticate user by using secure pin to verify the identity and prevent anonymous or malicious attack.

1.5 IMPACT AND CONTRIBUTION

Although there are numerous of emergency application, but majority of them need purchase to enjoy more features. By using this application, user can get help immediately when they faced problem or in troublesome. As mention in earlier chapter, nowadays people working in the daytime, they only tend to gathering at night. When gathering dismiss, people might be worried about their member's safety when they have on the way home. With the features of this application, users can track their members traveling status, this features also can notify user when their members arrive on estimate time. Not only for tracking purpose, this application also features with SOS emergency alert. Miss out the golden time to get instant help might affect someone loses their life. This emergency alert can help user to escape from worst situation or even can prevent sad or unhappy cases happened. Next is identity verification. User need to key their secure pin to authenticate their identity. As compare to existing application in the market, none of them is using authentication method to prevent anonymous or malicious attack. By having this feature, user can now use this application more securely. This application is suitable for all ages no matter old or young generation. For some family, parent can reduce their worries when their child or parent is outside. With using this application, people can rest assured with their family. Public fear will also decrease at the same time.

1.6 PROJECT SCOPE

This location tracking and emergency system is a mobile application, which is in android platform. User may use to send their emergency alert including current GPS location to their pre-set urgent contact person. Other than that, user also can track their friend list status whether they have reach destination and their approximate arrival time. Furthermore, in order to make sure user identity, user need to key their secure pin to check in to prevent someone with malicious to cheat their member.

1.7 STRUCTURE OF THE REPORT

Chapter 1 provides a brief idea of problem statement, background, impact and contribution, objective and lastly motivation. In problem statement, it explained what the existing problem people faced and its reason. In project scope, it outlines what are the things and area should cover in this propose application. While in background, it provides general introduction for reader to have some idea on it. Impact and contribution mention that how this propose application would impact to the society. Lastly, there are several points' lists in the objective section. It helps proposer to have clear objective in order to complete the project.

Chapter 2 covers the literature review of existing system. There are 4 existing applications are chosen for review. It helps to screen out what are the weakness and strengths of existing application. There are also include some existing product screenshot that show how it looks like. Furthermore, a comparison table is also including in this section.

Chapter 3 covers about the Experimental Work of the application. Chapter 4 covers about the Implementation of the application. Chapter 5 covers about the Application Interface of the application. Chapter 6 covers about the Conclusion.

CHAPTER - 2

LITERATURE REVIEW

2.1 EXISTING EMERGENCY APPLICATION

There are many similar emergency applications available in today's market; each of them only has one common objective, which is personal safety. Each of them have their own uniqueness, functionality, features and the competitive advantages. Several applications that will use for comparison below are:

2.1.1 BD 999

National Emergency Service (BD 999) App for Safety Assistance.

National Emergency Service (BD 999) App for Safety Assistance helps user in emergency by alerting the Bangladesh Police. It alerts Bangladesh Police by sending emergency message along with user's location and a quick video. User's location data will be periodically update so that Police can use this data to respond.

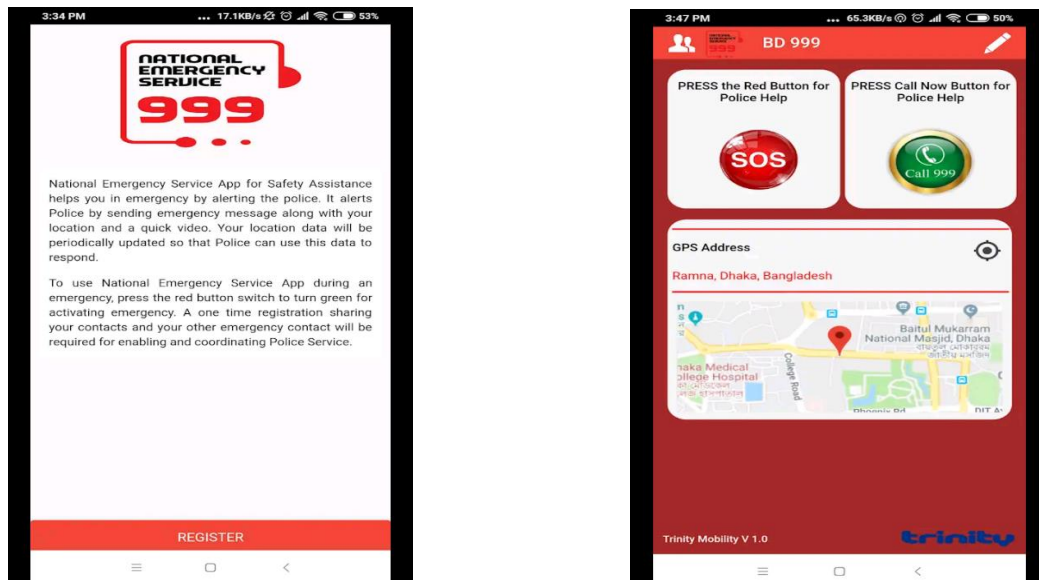


Figure No: 2.1

Weakness: Registration process does not work properly. When the app has been open, it want permission but after allowed all permission the app has hanged on registration screen. [1]

2.1.2 Emergency SOS Safety Alert

The Emergency SOS Safety Alert – Personal Alarm App is been developed to help user quickly and simultaneously advise key people that anyone need urgent help, this can include family, friends, neighbors, colleagues or eve medical staff – whomever user nominate.

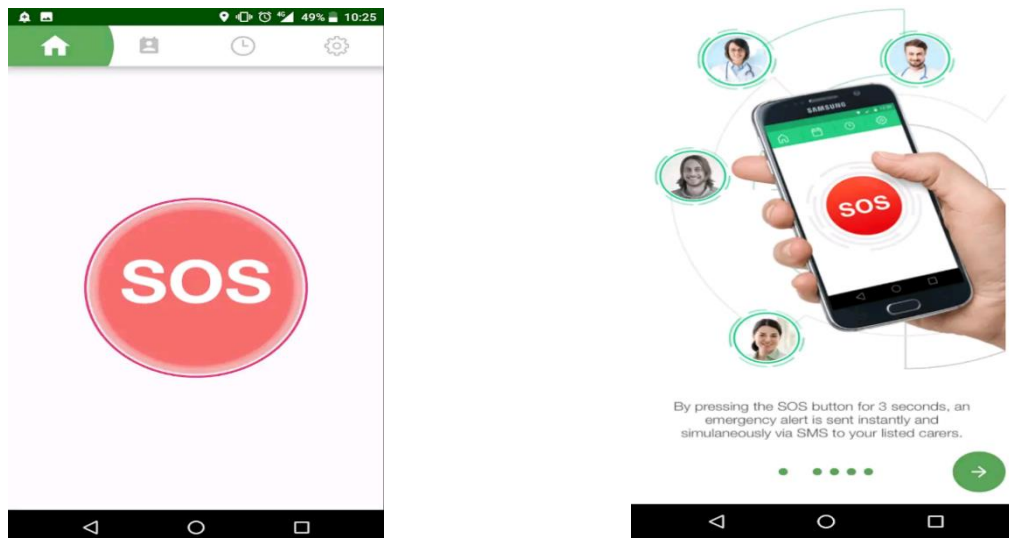


Figure No: 2.2

This app uses a mobile phone's GPS functionality and includes user's location on a map, the time the emergency button has pressed along with a prearranged tailored message to user's message recipients.

Weakness: If once user hit the SOS button, the cancel button does not work and it sends the alert anyway. The problem is user cannot edit profile. [2]

2.1.3 bSafe

Application that can invite friends to walk with user to get home to prevent alone. It allows user to set an automated alarm to notify user so that if user fail to check in after a set amount of time and will notify his/her friends where user have been and where user currently are. A 10- second video will start recording when the SOS alarm is on. Besides that, bSafe come with one interesting features that other application does not have, a Fake Call. This fake call function will trigger a fake call to user itself, in case of such a ruse makes it easier for user to extricate themselves from an uncomfortable or unsafe situation. However, it also can be an interesting feature to play around with kids, or to prank their friends.

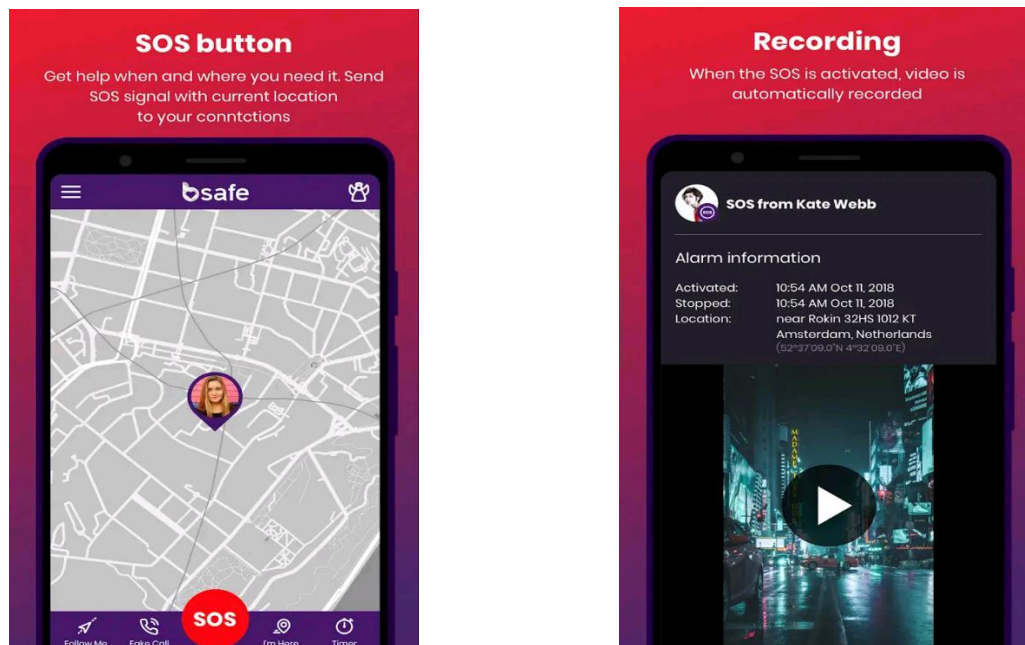


Figure No: 2.3

Strengths: trigger a fake call function that will trigger a fake call to user itself, this fake call can extricate user from an uncomfortable or unsafe situation.

Weakness: Invite friends to walk with user. Although this option is to find nearby friends to accompany or walk with user, but it may bring user to trouble when other people hacks user's friends account. [3]

2.1.4 Safety – SOS

Safety is a free, ad-free and offline personal safety application focused on ensuring the safety of users and users selected contacts with features like Safety emergency activation, offline location tracking and verified disaster alerts. Unique features, all of which are available free makes Safety users perfect personal safety companion.

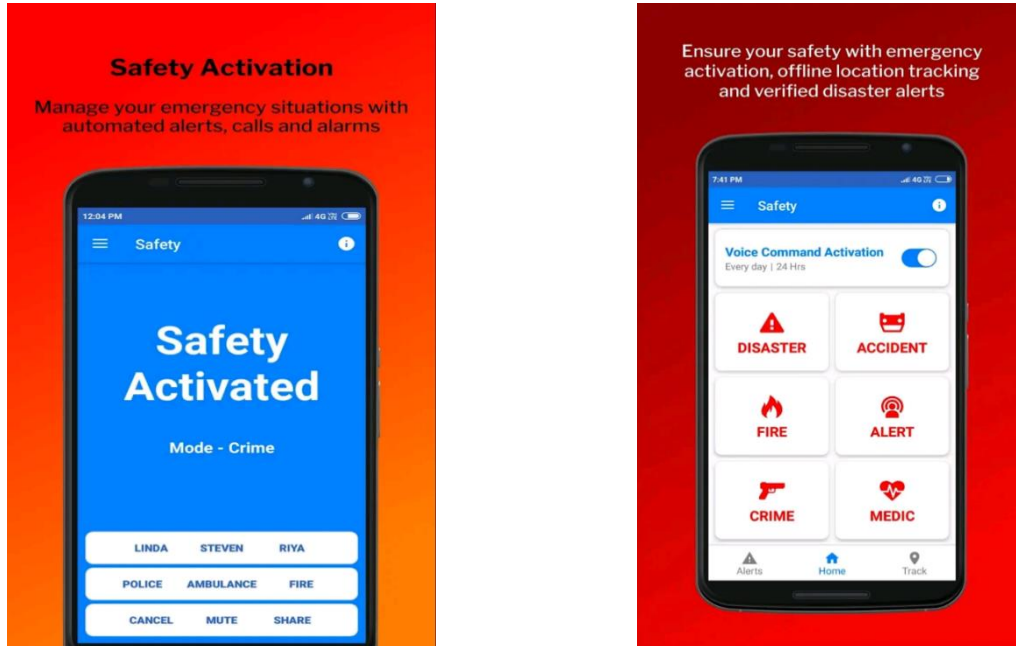


Figure No: 2.4

Weakness: It has self-activated mode, which is not responses much good, sometimes it activates automatically in normal situation. [4]

CHAPTER - 3

EXPERIMENTAL WORK

3.1 WORKING PROCEDURE

We made a very simple and user-friendly emergency app that can be operate by anyone. It has some steps. Those steps has given below:

- Sign UP
- Sign IN
- Grant Permission
- Main Screen, Tap & send Emergency Message to anyone.

3.1.1 Grant Permission

User need to grant some permission to run this app and its necessary.

- Contacts
- Phone
- Messages
- Location

3.1.2 Sign UP

If anyone is new to this app then user have to Sign UP or Register an account, which will save to our Database. User need to fill some form and those attributes given below:

- First Name
- Last Name
- Email
- Password

Fill those attributes to submit users “**Sign Up**” details to create an account.

If user “**Already Have an Account**” then just click to “**Sign In**”.

3.1.3 Sign IN

If user have already registered, user can now just log in and retain the SoS Service .There are some attributes on the Sign In page those has given bellow:

- Email
- Password

Don't Have An Account Then Sign UP & Continue.

3.1.4 Main Screen

The main screen is very easy to use for a user. An SOS button on the middle on the screen. Just tap the screen when user are in danger or an awkward situation, an emergency message will arrive to users wish list number with users current exact location. There are also some attributes on the main screen those given below:

- SOS
- ADD NUMBER
- ABOUT
- SOME EMERGENCY NUMBER

User can Add/Update your Emergency Contacts form Lists.

3.2 WORKING MODEL

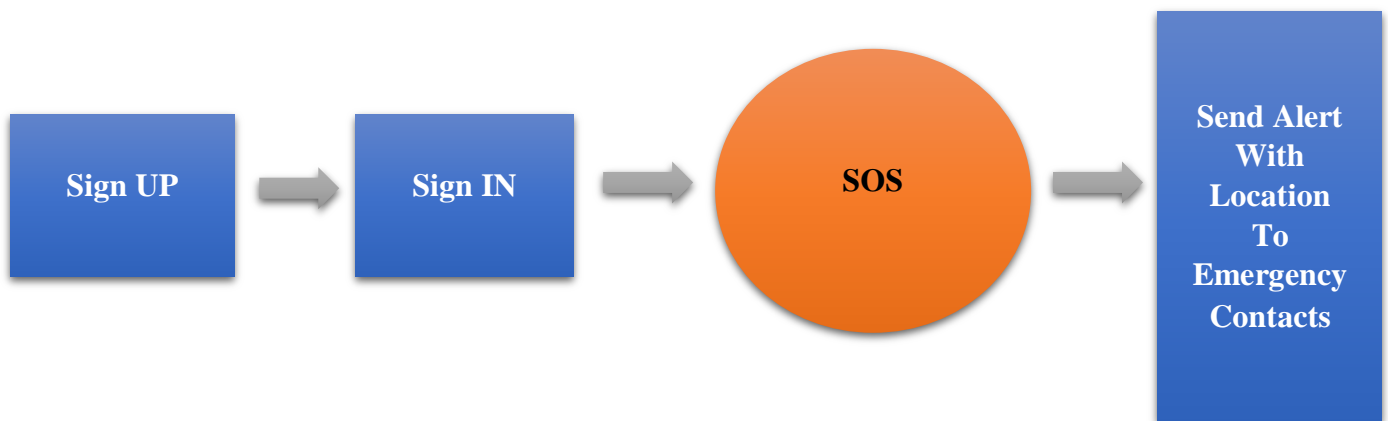
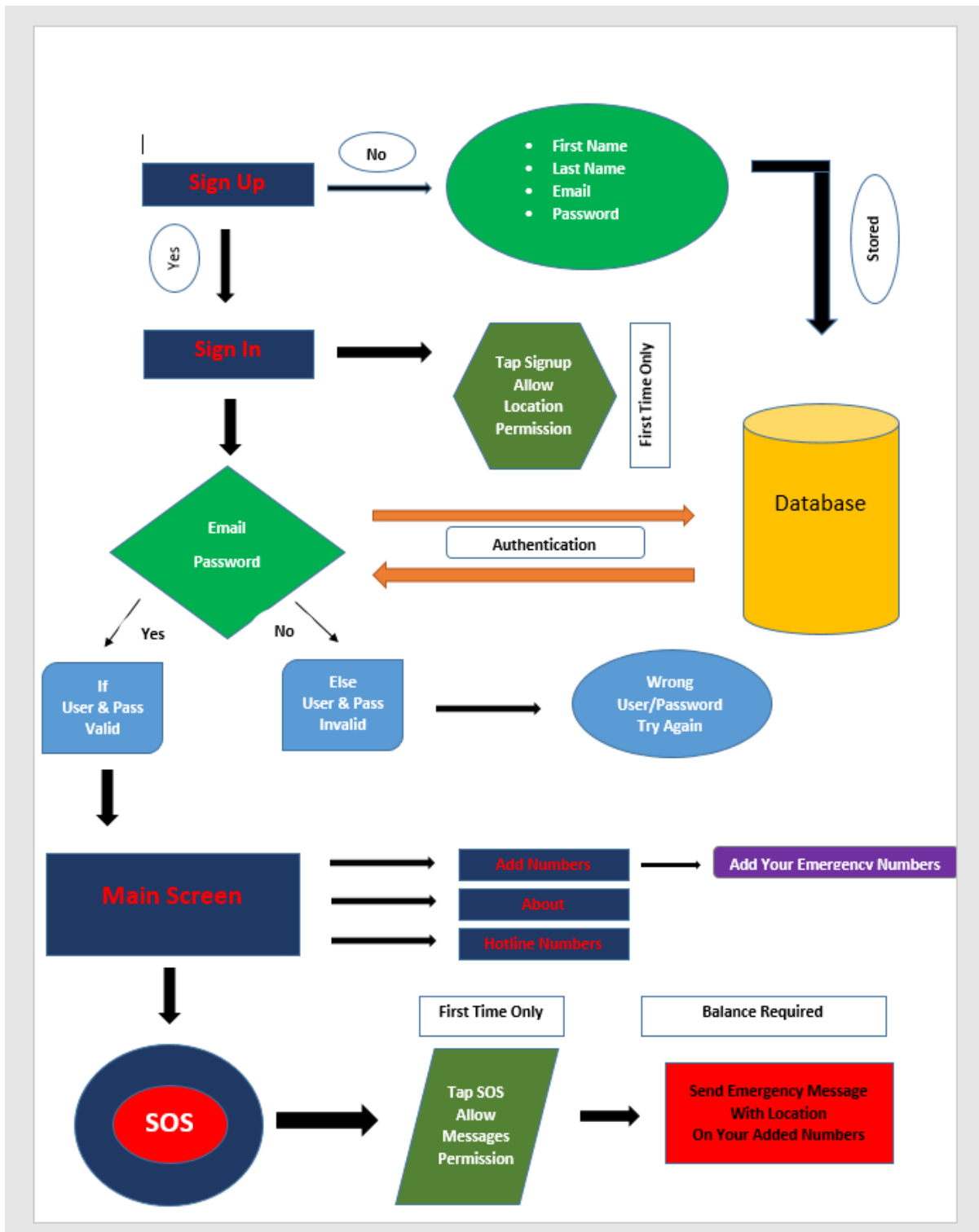


Figure No: 3.1

3.3 KEY FEATURES

- When user in an Emergency situation, just one tap to SOS and send SMS to Users Contacts
- It can be operate in offline
- Just SMS balance required nothing else.
- User can Add & Update his/her contacts.

3.4 FLOWCHART



CHAPTER - 4

IMPLEMENTATION

4.1 ENVIRONMENT

4.1.1 Tools

- JDK
- Android Studio
- Firebase
- SDK Tools

JDK

The Java Development Kit is an implementation of either one of the Java Platform, Standard Edition, Java Platform, Enterprise Edition, or Java Platform, Micro Edition platforms released by Oracle Corporation in the form of a binary product aimed at Java developers on Solaris, Linux, mac OS or Windows.

Android Studio

Android Studio is the official android environment produce by Google.Inc. It is available for Android, Linux OS, mac OS.

Firebase

Firebase is a mobile and web application development platform developed by Firebase, Inc. in 2011, It is used as Database of Android Studio. It is real time database.

SDK Tools

The SDK Tools primarily includes the stock Android emulator, Some SDK tools must needed installed to develop an application. Such as NDK, Android SDK Tools, Google Play Licensing Library, Android Emulator etc.

4.1.2 Language

- Android
- Java
- JSON

Android

The official language for Android development is Java. Large parts of Android has written in Java and its APIs designed to call primarily from Java. It is possible to develop C and C++ app using the Android Native Development Kit (NDK), however it is not something that Google promotes.

Java

Java is a general-purpose programming language that is class-based, object-oriented, and designed to have as few implementation dependencies as possible.

JSON

JSON stands for JavaScript Object Notation. JASON is an independent Parser Platform which can be used alternative of XML. There are four different classes which manipulate JSON Data.

4.2 ENVIRONMENT SETUP

4.2.1 Steps

- Download Android Studio from https://developer.android.com/studio/?gclid=Cj0KCQjwn_LrBRD4ARIsAFEQFKvcOoPXwMcb1BUhWDqGijTzHElnEywUeImXHDAp3wn4L_yuY8lxwRAaAajkaEALw_wcB & install it to computer. [5]
- Install Latest SDK tool on the Android Studio.
- Download JDK from <https://www.oracle.com/technetwork/java/javase/downloads/index.html> & install it to computer. [6]

4.2.2 Installed Android Studio

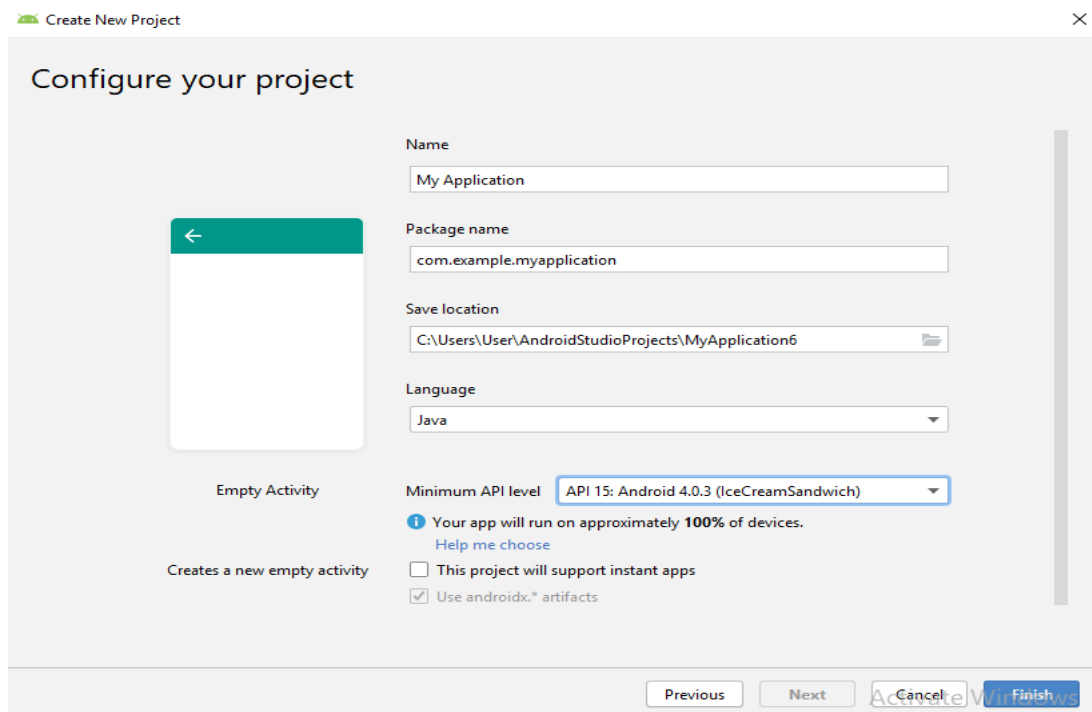


Figure No: 4.1

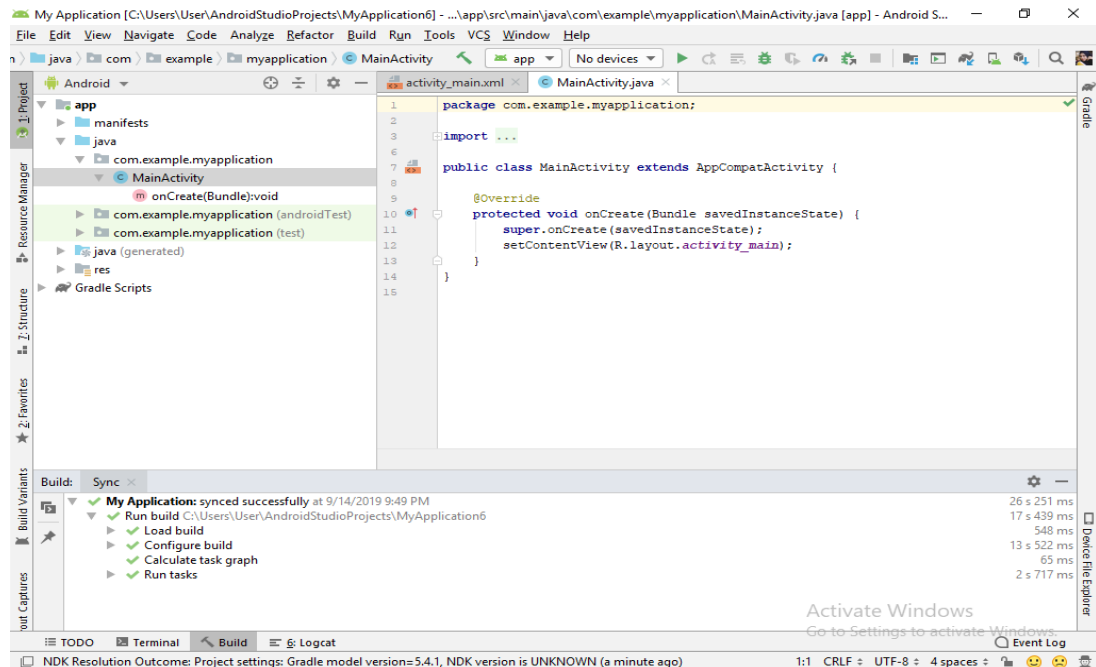


Figure No: 4.2

4.2.3 Installed JDK

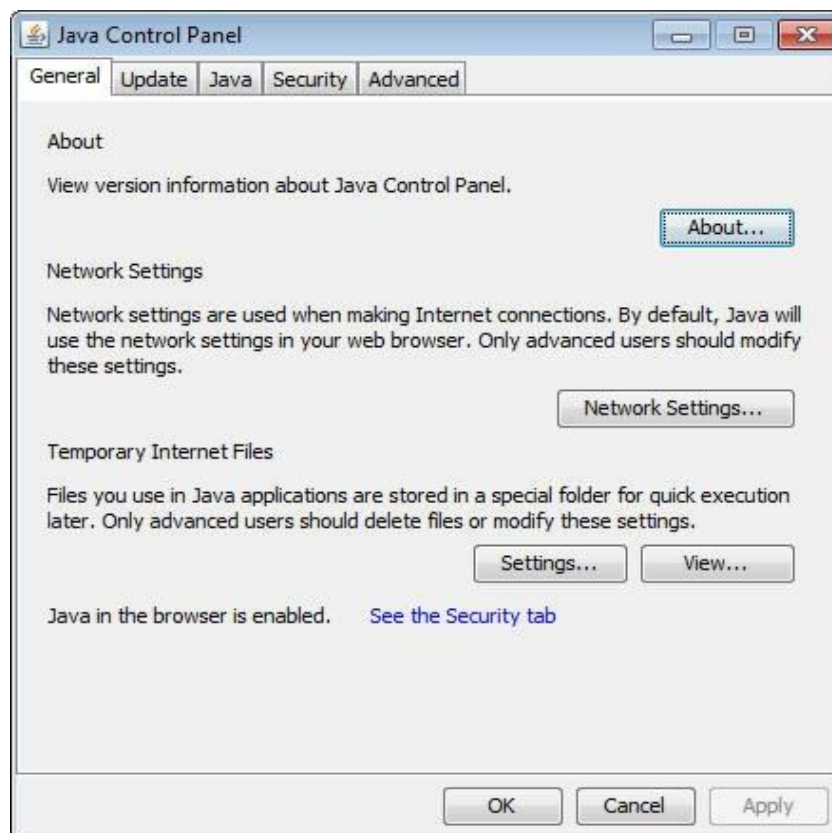


Figure No: 4.3

4.2.4 Installed SDK Tools

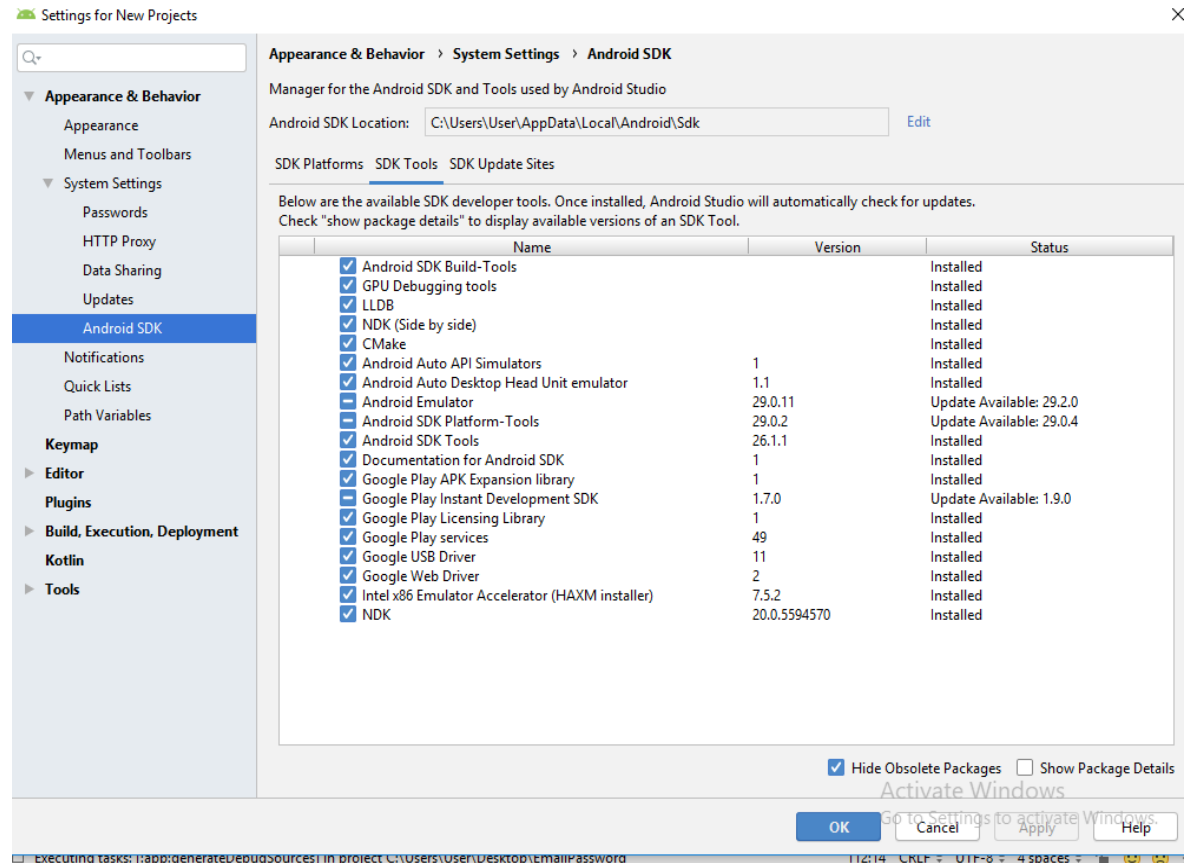


Figure No: 4.4

4.3 PROGRAMMING

4.3.1 Java Activity

- About.java
- AddEmergency.java
- MainActivity.java
- SignInActivity.java
- SignUpActivity.java
- Some.java
- UpdateEmergency.java
- User.java

4.3.2 XML Activity

- activity_about.xml
- activity_add_emergency.xml
- activity_main.xml
- activity_sign_in.xml
- activity_sign_up.xml
- activity_some.xml
- activity_update_emergency.xml

Java Activity & XML Activity resource code has given in the “APPENDIX”. [7]

4.3.3 JSON Generated Code for Firebase

```
{
  "project_info": {
    "project_number": "888500655368",
    "firebase_url": "https://sosapp-66789.firebaseio.com",
    "project_id": "sosapp-66789",
    "storage_bucket": "sosapp-66789.appspot.com"
  },
  "client": [
    {
      "client_info": {
```

```

    "mobilesdk_app_id": "1:888500655368:android:983badd0037028f369c8bd",
    "android_client_info": {
      "package_name": "com.example.emailpassword"
    }
  },
  "oauth_client": [
    {
      "client_id": "888500655368-
opv4g0g58ttpj3mh059vrmap4779hgk.apps.googleusercontent.com",
      "client_type": 1,
      "android_info": {
        "package_name": "com.example.emailpassword",
        "certificate_hash": "78f954650ad5d1d7ff12546bead51a249fae7962"
      }
    },
    {
      "client_id": "888500655368-
8a61tk5suc4dnscihm3296gnqeif9qof.apps.googleusercontent.com",
      "client_type": 3
    }
  ],
  "api_key": [
    {
      "current_key": "AIzaSyDuOvxEDYdi-59F9t4X1Vzm5lXtOYO4VuU"
    }
  ],
  "services": {
    "appinvite_service": {
      "other_platform_oauth_client": [
        {
          "client_id": "888500655368-
8a61tk5suc4dnscihm3296gnqeif9qof.apps.googleusercontent.com",
          "client_type": 3
        }
      ]
    }
  }
],
"configuration_version": "1"
} [8]

```

4.3.4 SHA-1 Generated Serial for Firebase

10:D9:04:13:14:89:BF:AD:26:B0:CA:B1:3D:CC:5F:88:E5:A9:A4:DE

4.4 FIREBASE STRUCTURE

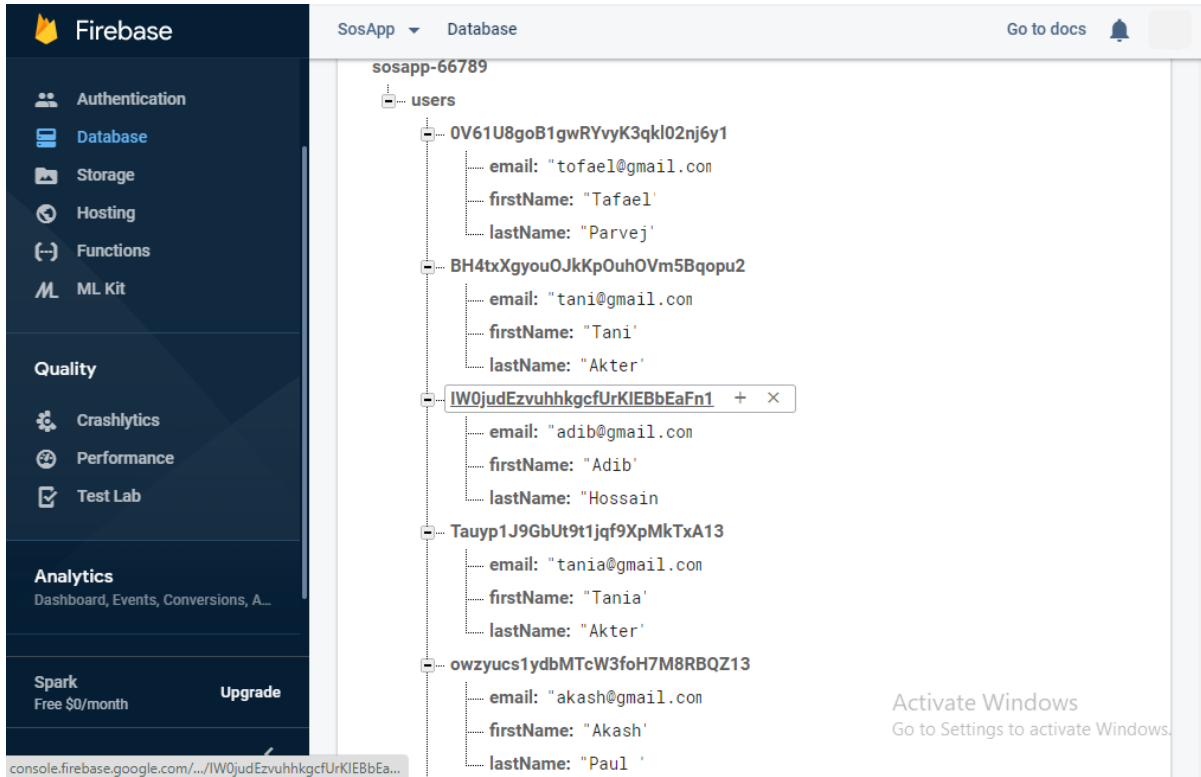


Figure No: 4.5

[9]

4.5 FIREBASE DATABASE TABLE

Table No: 4.1









<div>  Search by email address, phone number, or user UID <div> Add user   </div> </div>				
Identifier	Providers	Created	Signed In	User UID ↑
tofael@gmail.com		Sep 1 ...	Sep 1 ...	0V61U8goB1gwRYv...
tani@gmail.com		Sep 1 ...	Sep 1 ...	BH4txXgyouOJkKpO...
adib@gmail.com		Sep 1 ...	Sep 1 ...	IW0judEzvuhhkgcfU...
tania@gmail.com		Sep 1 ...	Sep 1 ...	Tauyp1J9GbUt9t1jq...
akash@gmail.com		Sep 1 ...	Sep 1 ...	owzyucs1ydbMTcW...

Table No: 4.2

User ID	firstName	lastName	email	Password
01	"Tofael"	"Parvej"	"tofael@gmail.com"	*****
02	"Tani"	"Akter"	"tani@gmail.com"	*****
03	"Adib"	"Hossain"	"adib@gmail.com"	*****
04	"Tania"	"Akter"	"tania@gmail.com"	*****
05	"Akash"	"Paul"	"akash@gmail.com"	*****

[9]

CHAPTER - 5

APPLICATION INTERFACE

5.1 APPLICATION INTERFACE

5.1.1 Sign UP

After installing the application user must be Sign UP an account with some valid details, Sign UP Credentials is given below:

- First Name
- Last Name
- Email
- Password

After filling with this information, press the Sign UP button to continue.

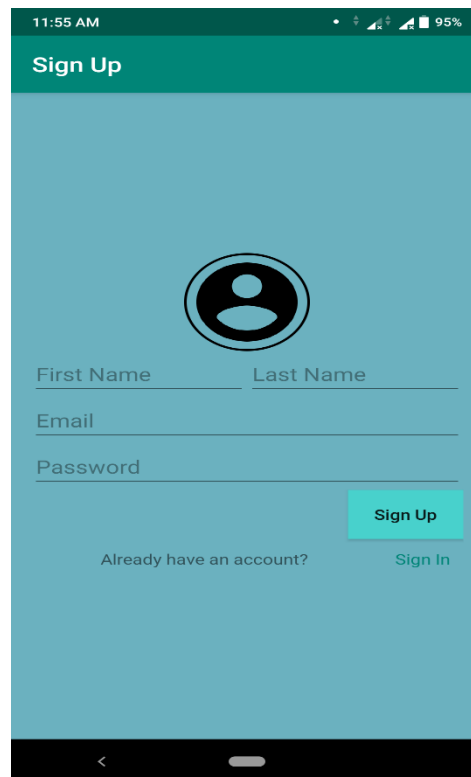


Figure No: 5.1

5.1.2 Sign IN

If you already have an account, then just sign in with your E-mail & Password then press Sign IN button to continue. Sign IN Credentials is given below:

- Email
- Password

After filling with this information, press the Sign IN button to continue.

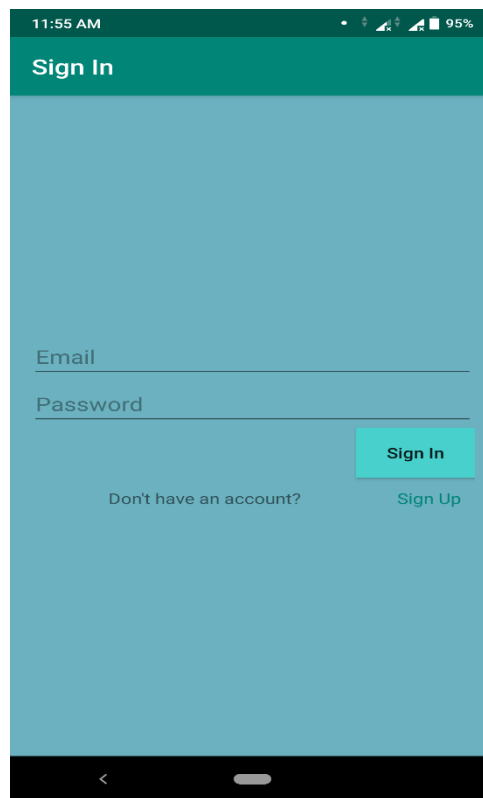


Figure No: 5.2

5.1.3 Main Screen

When anyone in an odd situation or have an accident or any kind of emergency help, User just have to tap the SOS button, an SMS will arrive to his/her added emergency contacts with his/her location.

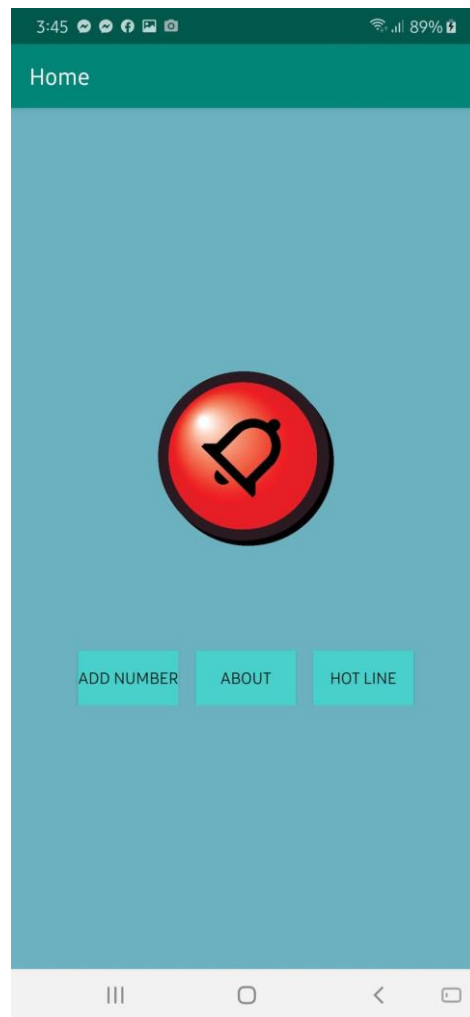
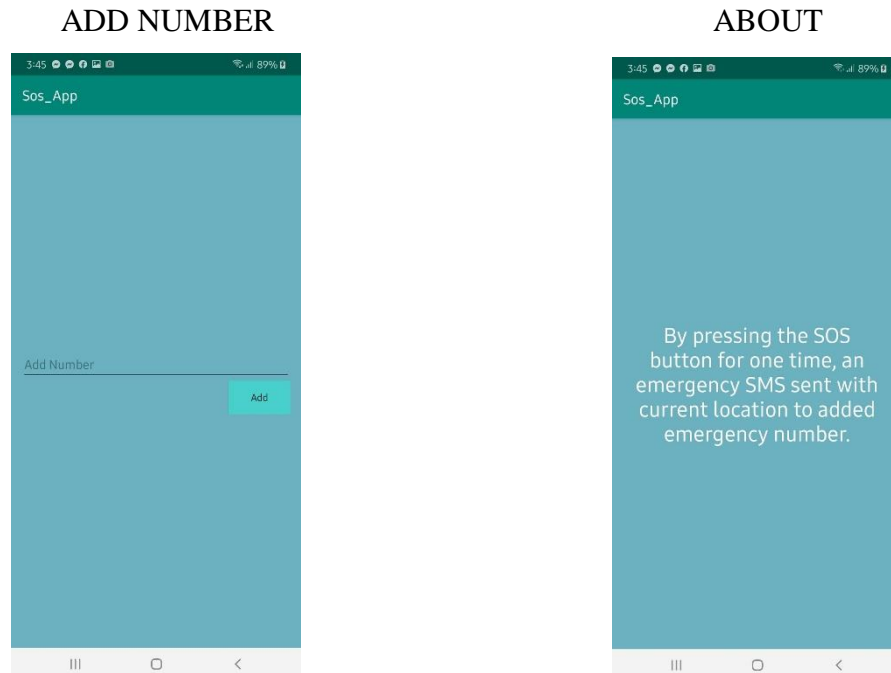


Figure No: 5.3

5.1.4 Other Buttons

Bottom of the SOS button, another 3 buttons there, those has given bellow:



SOME EMERGENCY NUMBER [10,11,12]



Figure No: 5.4. 5.5 & 5.6

5.1.5 Cell Phone SMS Screen

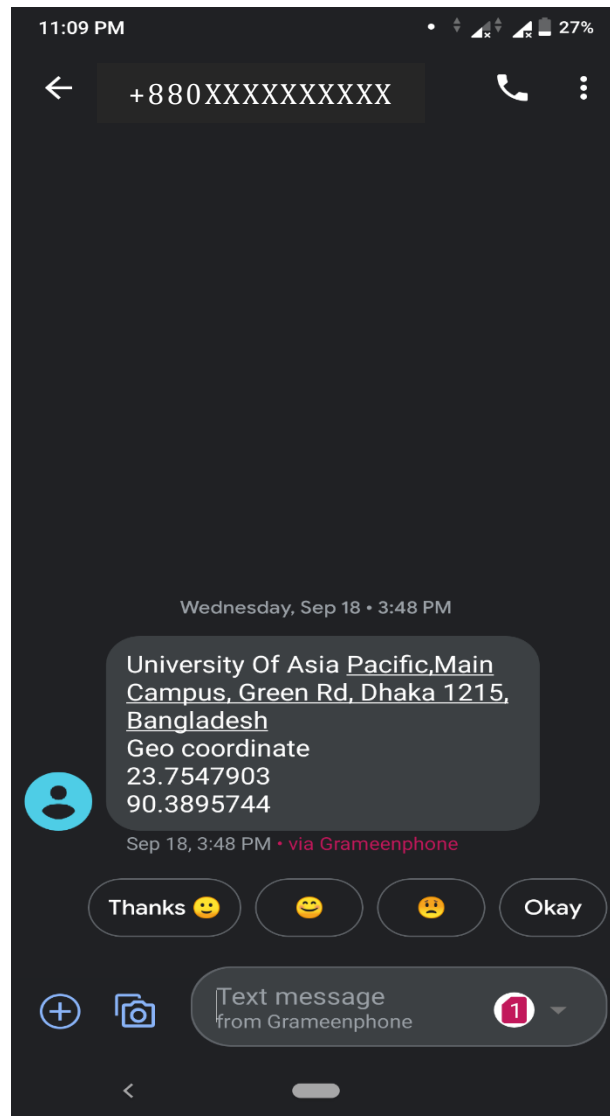


Figure No: 5.7

CHAPTER - 6

CONCLUSION

The Emergency alert app is for used by people of all ages, including children, teenagers, elderly, women and anyone who wants to feel safe and have peace of mind.

The app is can be also used in a variety of situations when user may need to ask for help or seek assistance from others. This could include potential assault, kidnapping, being lost, experiencing a serious health problem, feeling unsafe, being involved in an accident, and many more situations.

6.1 IMPLEMENTATION ISSUES

- We have faced some problem to connect the app with database.
- We used Firebase as Database.
- Faces some major problem during building the app on Android Studio environment.
- Faced issues for Emulator.

6.2 LIMITATIONS

- Need a general message cost while sending the emergency message, So need balance on your phone.
- You cannot update the emergency message, its default. Only Admin can change it.

6.3 FUTURE WORK

- We will implement Volume button press or Voice Command activity system in the app.
- To store some fixed length of audio and video clips, it will recorded automatically while Emergency Button Presses.
- Secure those media data.

APPENDIX

JAVA CODE

SignInActivity.java

```
package com.example.emailpassword;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;
import androidx.databinding.DataBindingUtil;

import android.content.Intent;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.Toast;

import com.example.emailpassword.databinding.ActivitySignInBinding;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnFailureListener;
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;

public class SignInActivity extends AppCompatActivity {

    private ActivitySignInBinding binding;
    private FirebaseAuth firebaseAuth;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = DataBindingUtil.setContentView(this,R.layout.activity_sign_in);

        init();

        if (firebaseAuth.getCurrentUser()!=null){
```

```
        goToMain();
    }

    binding.signInBtn.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            String email = binding.emailET.getText().toString();
            String password = binding.passwordET.getText().toString();

            signInWithEmailPassword(email,password);

            Intent intent = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
            intent.setType("image/*");
        }
    });

    binding.signUpTV.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View view) {
            startActivity(new Intent(SignInActivity.this,SignUpActivity.class));
        }
    });

}

private void init() {
    firebaseAuth = FirebaseAuth.getInstance();
}

private void signInWithEmailPassword(String email, String password) {

firebaseAuth.signInWithEmailAndPassword(email,password).addOnCompleteListener(new
OnCompleteListener<AuthResult>() {
    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        if (task.isSuccessful()){
            Toast.makeText(SignInActivity.this, "Sign In Successful",
Toast.LENGTH_SHORT).show();
```

```

        goToMain();
    }
}
}).addOnFailureListener(new OnFailureListener() {
    @Override
    public void onFailure(@NonNull Exception e) {
        Toast.makeText(SignInActivity.this, e.getMessage(),
Toast.LENGTH_SHORT).show();
    }
});
}

private void goToMain() {
    startActivity(new Intent(SignInActivity.this,MainActivity.class)

.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK|Intent.FLAG_ACTIVITY_NEW_TAS
K));
}
} [9]

```

SignUpActivity.java

```

package com.example.emailpassword;

import androidx.annotation.NonNull;
import androidx.annotation.Nullable;
import androidx.appcompat.app.AppCompatActivity;
import androidx.databinding.DataBindingUtil;

import android.app.ProgressDialog;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.provider.MediaStore;
import android.view.View;
import android.widget.Toast;

import com.example.emailpassword.databinding.ActivitySignUpBinding;
import com.google.android.gms.tasks.OnCompleteListener;
import com.google.android.gms.tasks.OnSuccessListener;

```

```
import com.google.android.gms.tasks.Task;
import com.google.firebase.auth.AuthResult;
import com.google.firebase.auth.FirebaseAuth;
import com.google.firebase.database.DatabaseReference;
import com.google.firebase.database.FirebaseDatabase;
import com.google.firebase.storage.FirebaseStorage;
import com.google.firebase.storage.StorageReference;
import com.google.firebase.storage.UploadTask;

import java.util.Map;

public class SignUpActivity extends AppCompatActivity {

    private ActivitySignUpBinding binding;
    private FirebaseAuth firebaseAuth;
    private DatabaseReference databaseReference;
    private StorageReference storageReference;
    private Uri uri;
    private String profileImageUrl;
    private ProgressDialog progressDialog;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        binding = DataBindingUtil.setContentView(this,R.layout.activity_sign_up);

        init();

        binding.profileImage.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View view) {
                Intent intent = new Intent(Intent.ACTION_PICK,
MediaStore.Images.Media.EXTERNAL_CONTENT_URI);
                intent.setType("image/*");
                startActivityForResult(intent,1);
            }
        });

        binding.signUpBtn.setOnClickListener(new View.OnClickListener() {
            @Override
```

```
public void onClick(View view) {
    String firstName = binding.firstNameET.getText().toString();
    String lastName = binding.lastNameET.getText().toString();
    String email = binding.emailET.getText().toString();
    String password = binding.passwordET.getText().toString();

    progressDialog.setMessage("Please wait...");
    progressDialog.show();
    createUser(firstName,lastName,email,password);

}

});

binding.signInTV.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        onBackPressed();
    }
});
}

private void init() {
    firebaseAuth = FirebaseAuth.getInstance();
    databaseReference = FirebaseDatabase.getInstance().getReference();
    storageReference = FirebaseStorage.getInstance().getReference();
    progressDialog = new ProgressDialog(this);
}

private void createUser(final String firstName, final String lastName, final String email,
final String password) {

firebaseAuth.createUserWithEmailAndPassword(email,password).addOnCompleteListener(n
ew OnCompleteListener<AuthResult>() {
    @Override
    public void onComplete(@NonNull Task<AuthResult> task) {
        String userId = firebaseAuth.getCurrentUser().getUid();
        User user = new User(firstName,lastName,email,profileImageUrl);
```



```
DatabaseReference userRef = databaseReference.child("users").child(userId);
userRef.setValue(user).addOnCompleteListener(new
OnCompleteListener<Void>() {
    @Override
    public void onComplete(@NonNull Task<Void> task) {
        if (task.isSuccessful()){
            progressDialog.dismiss();
            Toast.makeText(SignUpActivity.this, "SignUp Successful",
Toast.LENGTH_SHORT).show();
            goToMain();
        }
    }
});

}
});
}

@Override
protected void onActivityResult(int requestCode, int resultCode, @Nullable Intent data) {
    if (requestCode==1 && resultCode==RESULT_OK){
        uri = data.getData();
        binding.profileImage.setImageURI(uri);
    }
}

private void goToMain() {
    startActivity(new Intent(SignUpActivity.this,MainActivity.class)

.addFlags(Intent.FLAG_ACTIVITY_CLEAR_TASK|Intent.FLAG_ACTIVITY_NEW_TAS
K));
}
} [9]
```

User.java

```
package com.example.emailpassword;

public class User {
```

```
private String firstName;
private String lastName;
private String email;
private String profileImageUrl;

public User() {
}

public User(String firstName, String lastName, String email, String profileImageUrl) {
    this.firstName = firstName;
    this.lastName = lastName;
    this.email = email;
    this.profileImageUrl = profileImageUrl;
}

public String getFirstName() {
    return firstName;
}

public String getLastName() {
    return lastName;
}

public String getEmail() {
    return email;
}

public String getProfileImageUrl() {
    return profileImageUrl;
}
}
```

MainActivity.java

```
package com.example.emailpassword;

import androidx.annotation.Nullable;
import androidx.appcompat.app.AlertDialog;
import androidx.appcompat.app.AppCompatActivity;
import androidx.core.app.ActivityCompat;
import androidx.core.content.ContextCompat;

import android.Manifest;
```

```
import android.content.Context;
import android.content.DialogInterface;
import android.content.Intent;
import android.content.pm.PackageManager;
import android.location.Address;
import android.location.Geocoder;
import android.location.Location;
import android.location.LocationManager;
import android.os.Bundle;
import android.os.PersistableBundle;
import android.provider.Settings;
import android.telephony.SmsManager;
import android.view.View;
import android.widget.Button;
import android.widget.TextView;
import android.widget.Toast;

import java.io.IOException;
import java.util.List;
import java.util.Locale;

public class MainActivity extends AppCompatActivity {

    private static final int REQUEST_LOCATION = 1;
    Button sosbutton;
    Button add;
    Button emergency;
    Button about;
    LocationManager locationManager;
    String latitude,longitude;

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main); [13]
```

```
ActivityCompat.requestPermissions(this,new String[]
    {Manifest.permission.ACCESS_FINE_LOCATION}, REQUEST_LOCATION);

sosbutton = findViewById(R.id.sos);
add = findViewById(R.id.add);
emergency = findViewById(R.id.emergency);
about = findViewById(R.id.about);

sosbutton.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        locationManager=(LocationManager)
getSystemService(Context.LOCATION_SERVICE);

        if (!locationManager.isProviderEnabled(LocationManager.GPS_PROVIDER)) {
            OnGPS();
        }
        else{
            getLocation();
        }
    }
});

add.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( MainActivity.this , AddEmergency.class);
        startActivity(intent);
    }
});

emergency.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( MainActivity.this , Some.class);
        startActivity(intent);
    }
});
```

```
about.setOnClickListener(new View.OnClickListener() {
    @Override
    public void onClick(View view) {
        Intent intent = new Intent( MainActivity.this , About.class);
        startActivity(intent);
    }
});

}

private void getLocation() {
    if
(ActivityCompat.checkSelfPermission(MainActivity.this,Manifest.permission.ACCESS_FINE_LOCATION) != PackageManager.PERMISSION_GRANTED &&
ActivityCompat.checkSelfPermission(MainActivity.this,

        Manifest.permission.ACCESS_COARSE_LOCATION)
!=PackageManager.PERMISSION_GRANTED)
    {
        ActivityCompat.requestPermissions(this,new String[]
            {Manifest.permission.ACCESS_FINE_LOCATION},
REQUEST_LOCATION);
    }
    else
    {
        ActivityCompat.requestPermissions(MainActivity.this , new String[]
{Manifest.permission.SEND_SMS,Manifest.permission.READ_SMS};
PackageManager.PERMISSION_GRANTED);
        Location LocationGps=
locationManager.getLastKnownLocation(LocationManager.GPS_PROVIDER);
        Location
LocationNetwork=locationManager.getLastKnownLocation(LocationManager.NETWORK_
PROVIDER);
        Location
LocationPassive=locationManager.getLastKnownLocation(LocationManager.PASSIVE_PR
OVIDER);
        if (LocationGps !=null) {
            Geocoder geocoder = new Geocoder(MainActivity.this, Locale.getDefault());
            try {
```

```
List<Address> addresses =
geocoder.getFromLocation(LocationGps.getLatitude(), LocationGps.getLongitude(), 1);
String adresse = addresses.get(0).getAddressLine(0);
String sender = adresse + "\n" + "Geo coordinate\n" +
LocationGps.getLatitude() + "\n" + LocationGps.getLongitude();
SmsManager smgr = SmsManager.getDefault();
smgr.sendTextMessage("+8801985590704",null,sender,null,null);

} catch (IOException e) {
    e.printStackTrace();
}

}
else if (LocationNetwork !=null)
{
    Geocoder geocoder = new Geocoder( MainActivity.this , Locale.getDefault());
    try {
        List<Address> addresses =
geocoder.getFromLocation(LocationNetwork.getLatitude(),
LocationNetwork.getLongitude(), 1);
        String adresse = addresses.get(0).getAddressLine(0);
        String sender = adresse + "\n" + "Geo coordinate\n" +
LocationNetwork.getLatitude() + "\n" + LocationNetwork.getLongitude();
        SmsManager smgr = SmsManager.getDefault();
        smgr.sendTextMessage("+8801985590704",null,sender,null,null);
    }
    catch (IOException e) {
        e.printStackTrace();
    }
}
else if (LocationPassive !=null)
{
    Geocoder geocoder = new Geocoder( MainActivity.this , Locale.getDefault());
    try {
        List<Address> addresses =
geocoder.getFromLocation(LocationPassive.getLatitude(), LocationPassive.getLongitude(),
1);
        String adresse = addresses.get(0).getAddressLine(0); [14]
```

```
String sender = adresse + "\n" + "Geo coordinate\n" +
LocationPassive.getLatitude() + "\n" + LocationPassive.getLongitude();
SmsManager smgr = SmsManager.getDefault();
smgr.sendTextMessage("+8801985590704",null,sender,null,null);

    }catch (IOException e) {
        e.printStackTrace();
    }
}
else
{
    Toast.makeText(this, "Can't Get Your Location",
Toast.LENGTH_SHORT).show();
}
}
}

private void OnGPS() {
    final AlertDialog.Builder builder= new AlertDialog.Builder(this);

    builder.setMessage("Enable GPS").setCancelable(false).setPositiveButton("YES", new
DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {
            startActivity(new Intent(Settings.ACTION_LOCATION_SOURCE_SETTINGS));
        }
    }).setNegativeButton("NO", new DialogInterface.OnClickListener() {
        @Override
        public void onClick(DialogInterface dialog, int which) {

            dialog.cancel();
        }
    });
    final AlertDialog alertDialog=builder.create();
    alertDialog.show();
}

public boolean checkPermissions(String permissions) {
    int temp = ContextCompat.checkSelfPermission(this, permissions);
    return (temp == PackageManager.PERMISSION_GRANTED); [15]
```

```
}  
  
}
```

About.java

```
package com.example.emailpassword;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
  
public class About extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_about);  
    }  
}
```

AddEmergency.java

```
package com.example.emailpassword;  
  
import androidx.appcompat.app.AppCompatActivity;  
  
import android.os.Bundle;  
  
public class AddEmergency extends AppCompatActivity {  
  
    @Override  
    protected void onCreate(Bundle savedInstanceState) {  
        super.onCreate(savedInstanceState);  
        setContentView(R.layout.activity_add_emergency);  
    }  
}
```


Some.java

```
package com.example.emailpassword;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class Some extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_some);
    }
}
```

UpdateEmergency.java

```
package com.example.emailpassword;

import androidx.appcompat.app.AppCompatActivity;

import android.os.Bundle;

public class UpdateEmergency extends AppCompatActivity {

    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_update_emergency);
    }
}
```

XML CODE

activity_sign_in.xml

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

    <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".SignInActivity">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:layout_margin="15dp"
            android:orientation="vertical">

            <EditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/emailET"
                android:hint="Email"
                android:inputType="textEmailAddress"
                android:singleLine="true"
                android:imeOptions="actionNext"/>

            <EditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/passwordET"
                android:hint="Password"
                android:singleLine="true"
                android:imeOptions="actionDone"
                android:inputType="textPassword"/>

            <Button
                android:layout_width="wrap_content"
```

```

android:layout_height="wrap_content"
android:id="@+id/signInBtn"
android:text="Sign In"
android:layout_gravity="right"
android:textAllCaps="false"/>

```

```

<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <TextView
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:layout_weight="1"
        android:textAlignment="center"
        android:layout_gravity="center"
        android:text="Don't have an account? "/>
    <TextView
        android:layout_width="wrap_content"
        android:layout_height="wrap_content"
        android:text="Sign Up"
        android:padding="10dp"
        android:id="@+id/signUpTV"
        android:textColor="@color/colorPrimary"/>
</LinearLayout>

```

```
</LinearLayout>
```

```
</RelativeLayout>
```

```
</layout>
```

activity_sign_up.xml

```

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

```

```

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"

```

```
android:layout_width="match_parent"
android:layout_height="match_parent"
tools:context=".SignUpActivity">
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:layout_centerInParent="true"
    android:layout_margin="15dp"
    android:orientation="vertical">
```

```
    <de.hdodenhof.circleimageview.CircleImageView
xmlns:app="http://schemas.android.com/apk/res-auto"
        android:id="@+id/profileImage"
        android:layout_width="96dp"
        android:layout_height="96dp"
        android:layout_gravity="center"
        android:src="@drawable/ic_account_circle_black_24dp"
        app:civ_border_color="#FF000000"
        app:civ_border_width="2dp" />
```

```
<LinearLayout
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:orientation="horizontal">
    <EditText
        android:layout_width="0dp"
        android:layout_height="wrap_content"
        android:id="@+id/firstNameET"
        android:hint="First Name"
        android:inputType="textCapWords"
        android:singleLine="true"
        android:layout_weight=".5"
        android:imeOptions="actionNext"/>
    <EditText
        android:layout_width="0dp"
```

```
android:layout_height="wrap_content"
android:id="@+id/lastNameET"
android:hint="Last Name"
android:inputType="textCapWords"
android:singleLine="true"
android:layout_weight=".5"
android:imeOptions="actionNext"/>
```

```
</LinearLayout>
```

```
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/emailET"
    android:hint="Email"
    android:inputType="textEmailAddress"
    android:singleLine="true"
    android:imeOptions="actionNext"/>
```

```
<EditText
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:id="@+id/passwordET"
    android:hint="Password"
    android:singleLine="true"
    android:imeOptions="actionDone"
    android:inputType="textPassword"/>
```

```
<Button
    android:layout_width="wrap_content"
    android:layout_height="wrap_content"
    android:id="@+id/signUpBtn"
    android:text="Sign Up"
    android:layout_gravity="right"
    android:textAllCaps="false"/>
```

```
<LinearLayout
```

```

        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:orientation="horizontal">
        <TextView
            android:layout_width="0dp"
            android:layout_height="wrap_content"
            android:layout_weight="1"
            android:textAlignment="center"
            android:layout_gravity="center"
            android:text="Already have an account? "/>
        <TextView
            android:layout_width="wrap_content"
            android:layout_height="wrap_content"
            android:text="Sign In"
            android:padding="10dp"
            android:id="@+id/signInTV"
            android:textColor="@color/colorPrimary"/>
    </LinearLayout>

</LinearLayout>

</RelativeLayout>
</layout>

```

activity_main.xml

```

<?xml version="1.0" encoding="utf-8"?>
<LinearLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:orientation="vertical"
    android:gravity="center"
    tools:context=".MainActivity">

    <Button
        android:id="@+id/sos"
        android:layout_width="150dp"
        android:layout_height="150dp"

```

```
android:background="@drawable/sos"  
tools:layout_editor_absoluteX="130dp"  
tools:layout_editor_absoluteY="290dp" />
```

```
<TextView  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:textSize="10dp"  
    android:text=""  
    android:id="@+id/text_show"/>
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="126dp">
```

```
<Button  
    android:id="@+id/add"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="bottom"  
    android:text="Add Number"  
/>
```

```
<Button  
    android:id="@+id/about"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="bottom"  
    android:text="About"  
/>
```

```
<Button  
    android:id="@+id/emergency"  
    android:layout_width="wrap_content"  
    android:layout_height="wrap_content"  
    android:layout_gravity="bottom"  
    android:text="Some Emergency Number"  
/>
```

```
</LinearLayout>
```

```
</LinearLayout>
```

AndroidManifest.xml

```
<?xml version="1.0" encoding="utf-8"?>
<manifest xmlns:android="http://schemas.android.com/apk/res/android"
    package="com.example.emailpassword">

    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION"
/>
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
    <uses-permission android:name="android.permission.SEND_SMS" />
    <uses-permission android:name="android.permission.RECEIVE_SMS" />
    <uses-permission android:name="android.permission.READ_SMS" />
    <uses-permission android:name="android.permission.INTERNET" />

    <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/AppTheme">
        <activity android:name=".Some"></activity>
        <activity android:name=".UpdateEmergency" />
        <activity android:name=".AddEmergency" />
        <activity android:name=".About" />
        <activity
            android:name=".SignUpActivity"
            android:label="Sign Up" />
        <activity
            android:name=".SignInActivity"
            android:label="Sign In">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />

                <category android:name="android.intent.category.LAUNCHER" />
            </intent-filter>
```



```
</activity>
<activity
    android:name=".MainActivity"
    android:label="Home" />
</application>
</manifest>
```

activity_about.xml

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

<RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".SignUpActivity">

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_centerInParent="true"
        android:layout_margin="15dp"
        android:orientation="vertical">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:textSize="30dp"
            android:textAlignment="center"
            android:textColor="@android:color/holo_red_dark"
            android:text="This project is about a mobile application for location Tracking Service
and Emergency which works in Android platform."
        />

    </LinearLayout>
</RelativeLayout>
</layout>
```

activity_add_emergency.xml

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

    <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"
        xmlns:app="http://schemas.android.com/apk/res-auto"
        xmlns:tools="http://schemas.android.com/tools"
        android:layout_width="match_parent"
        android:layout_height="match_parent"
        tools:context=".SignUpActivity">

        <LinearLayout
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_centerInParent="true"
            android:layout_margin="15dp"
            android:orientation="vertical">

            <LinearLayout
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:orientation="horizontal"/>

            <EditText
                android:layout_width="match_parent"
                android:layout_height="wrap_content"
                android:id="@+id/number"
                android:hint="Add Number"
                android:inputType="textEmailAddress"
                android:singleLine="true"
                android:imeOptions="actionNext"/>

            <Button
                android:layout_width="wrap_content"
                android:layout_height="wrap_content"
                android:id="@+id/add"
                android:text="Add"
                android:layout_gravity="right"
```

```
android:textAllCaps="false"/>
```

```
<LinearLayout  
    android:layout_width="match_parent"  
    android:layout_height="wrap_content"  
    android:orientation="horizontal"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>  
</layout>
```

activity_some.xml

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

```
    <RelativeLayout xmlns:android="http://schemas.android.com/apk/res/android"  
        xmlns:app="http://schemas.android.com/apk/res-auto"  
        xmlns:tools="http://schemas.android.com/tools"  
        android:layout_width="match_parent"  
        android:layout_height="match_parent"  
        tools:context=".SignUpActivity">
```

```
        <LinearLayout  
            android:layout_width="match_parent"  
            android:layout_height="wrap_content"  
            android:layout_centerInParent="true"  
            android:layout_margin="15dp"  
            android:orientation="vertical">
```

```
            <TextView  
                android:layout_width="match_parent"  
                android:layout_height="wrap_content"  
                android:textSize="30dp"  
                android:text="Police"  
                android:textColor="@android:color/holo_red_dark"
```

/>

<View

android:layout_width="match_parent"
android:layout_height="2dp"
android:background="#000000"/>

<TextView

android:layout_width="match_parent"
android:layout_height="wrap_content"
android:textSize="25dp"
android:text="Bangladesh Police Head Quarters, Dhaka, 02-9561967"

/>

<View

android:layout_width="match_parent"
android:layout_height="2dp"
android:background="#000000"/>

<TextView

android:layout_width="match_parent"
android:layout_height="wrap_content"
android:textSize="25dp"
android:text="Dhaka Metropolitan Police Headquarters,02-8614300"

/>

<View

android:layout_width="match_parent"
android:layout_height="2dp"
android:background="#000000"/>

<TextView

android:layout_width="match_parent"
android:layout_height="wrap_content"
android:textSize="25dp"
android:text="New Market Police Station, 02-8631942"

/>

```
<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:background="#000000"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:text="Shahbagh Police Station,02-9676699"
/>
```

```
<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:background="#000000"/>

<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:text="Ramna Model Police Station, 01769-691652"
/>
```

```
<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:background="#000000"/>
```

```
<TextView
    android:layout_width="match_parent"
```

```
android:layout_height="wrap_content"
android:textSize="25dp"
android:text="Kalabagan Police Station, 01713-398339"
/>
```

```
<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:background="#000000"/>
```

```
<TextView
    android:layout_width="match_parent"
    android:layout_height="wrap_content"
    android:textSize="25dp"
    android:text="Dhanmondi Model Thana, 02-8631941"
/>
```

```
<View
    android:layout_width="match_parent"
    android:layout_height="2dp"
    android:background="#000000"/>
```

```
</LinearLayout>
```

```
</RelativeLayout>
```

```
</layout>
```

activity_update_emergency.xml

```
<?xml version="1.0" encoding="utf-8"?>
<androidx.constraintlayout.widget.ConstraintLayout
    xmlns:android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    tools:context=".UpdateEmergency">
```

```
</androidx.constraintlayout.widget.ConstraintLayout>
```

REFERENCES

1. **BD 999**
<https://play.google.com/store/apps/details?id=com.trinity.bangladeshpolicesos&hl=en> [5]
 [2.1.1]
2. **Emergency SOS Safety Alert – Personal Alarm App**
<https://play.google.com/store/apps/details?id=com.clarity.eap.alert&hl=en> [6] [2.1.2]
3. **bSafe**
<https://play.google.com/store/apps/details?id=com.bipper.app.bsaf&hl=en> [7] [2.1.3]
4. **Safety**
<https://play.google.com/store/apps/details?id=com.siriussoftwares.safetyapp&hl=en> [8]
 [2.1.4]
5. **Download Android Studio**
https://developer.android.com/studio/?gclid=Cj0KCQjwn_LrBRD4ARIsAFEQFKvcOoPXwMcb1BUhWDqGijTzHElnEywUeImXHDAp3wn4L_yuY8lxwRAaAjkaEALw_wcB
 [15] [4.2.1]
6. **Download JDK**
<https://www.oracle.com/technetwork/java/javase/downloads/index.html> [15] [4.2.1]
7. **BITM Android Course**
<https://bitm.org.bd/course-details/47/professional-android-application-development> [18]
 [4.3]
8. **JSON Parser**
https://www.tutorialspoint.com/android/android_json_parser.html [18] [4.3.3]
9. **Firestore Authentication With User & Password**
<https://www.youtube.com/watch?v=Zo5hxJCASAI&list=PLjkWr26KfaWyTsW8h8o3gBQC7MCbAXADL&index=41> [20,21,27,29,33] [4.4]
10. **Bangladesh Police**
https://www.police.gov.bd/en/contact_us [25] [5.1.4]
11. **Ambulance Service in Dhaka**
<http://directory.shopnobaz.com/ambulance-service-in-dhaka-telephone-number/> [25]
 [5.1.4]
12. **Bangladesh Fire Service & Civil Defense**
<http://www.fireservice.gov.bd/> [25] [5.1.4]
13. **Get Current Location**
<https://www.youtube.com/watch?v=aPv8rqEDX3A&list=PLjkWr26KfaWyTsW8h8o3gBQC7MCbAXADL&index=31> [34-38]
14. **Geocoder**
<https://developer.android.com/reference/android/location/Geocoder> [37]
15. **Android - Sending SMS**
https://www.tutorialspoint.com/android/android_sending_sms.html [38]