A
Project Report
on

KEEP IN TOUCH ANDROID APPLICATION

BTech-IT, Sem VI

Prepared By:
Chodvadiya Dhruvil (IT-020)
Gajera Rajat (IT-032)

Guided By:
Prof. (Dr.) Harshad B. Prajapati
Dept. of Information Technology

Department of Information Technology
Faculty of technology,
Dharmsinh Desai University
College road, Nadiad- 387001

April, 2021

CANDIDATE’S DECLARATION

I/We declare that pre-final semester report entitled “**Keep In Touch Android Application**” is my /our own work conducted under the supervision of the guide Prof. (Dr.) Harshad B. Prajapati

I/We further declare that to the best of my/our knowledge the report for B.Tech. VI semester does not contain part of the work which has been submitted either in this or any other university without proper citation.

Candidate’s Signature: Dhruvil
Candidate’s Name: Chodvadiya Dhruvilkumar Ashokbhai
Student ID: 18ITUOS054

Candidate’s Signature:
Candidate’s Name: Gajera Rajat Umeshbhai
Student ID: 18ITUOS043

Submitted To:
Prof. (Dr.) Harshad B. Prajapati
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Gujarat.

DHARMSINH DESAI UNIVERSITY

NADIAD-387001, GUJARAT

CERTIFICATE

This is to certify that the project carried out in the subject of Software Design Project, entitled "**Keep In Touch Android Application**" and recorded in this report is a bonafide report of work of

1) Chodvadiya Dhruvil Roll No. IT-020 ID No: 18ITUOS054
2) Gajera Rajat Roll No. IT-032 ID No: 18ITUOS043

of Department of Information Technology, semester VI. ~~He/She~~/They ~~was~~/were involved in Project work during academic year 2021-2022.

Prof. (Dr.) Harshad B. Prajapati
Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad
Date:

Prof. (Dr.) Vipul K. Dabhi,
Head, Department of Information Technology,
Faculty of Technology,
Dharmsinh Desai University, Nadiad.
Date:

ACKNOWLEDGMENT

This project would not have been possible without the guidance and the help of several individuals who in one way or another contributed and extended their valuable assistance in the preparation and completion of this study.

It gives us an immense pleasure submitting this report towards the partial fulfillment of our academics. Success in any mission cannot be achieved single handedly. It is the team effort that sails the ship to shore.

We would like to express our sincere thanks to our Head of Department and our project Guide Prof. (Dr.) Harshad B. Prajapati, who gave us an opportunity to undertake such a great challenging and innovative work, without whose help and encouragement, it would be infeasible for us to carry out the desired work. We are grateful to them for their guidance, encouragement, understanding and insightful support in the development process.

With sincere regards,
1.Dhruvil Chodvadiya
2.Rajat Gajera

TABLE OF CONTENTS

ABSTRACT

Keep In Touch app is an android application, build for providing a seamless experience to customers who are planning to go for group trip anywhere in the world and want to manage this tour very smartly. It will become easier for Group leader to track every group member's location and warn them to stay in their safe zone using this application. This application is built on the sole motive to boost the Trip experience very smartly. This app is built using Android Studio in java language and it uses Firebase as the database.

1. INTRODUCTION

1.1. PROJECT DETAILS

Our Project is on GPS based android application. This project can be used by everyone who knows how to use smartphone and usually going for a trip. In this we will be working on Android Studio. This project will make any simple trip as smart trip. User make a group and others are joining it and then if anyone outside decided area then receive notification and can go to safe zone area. And also they can chat via our application and also share photos.

1.2. PURPOSE

The Purpose of this idea is to make an application and helpful to them who find difficulties to manage people in the trip specially in school or college trip. So we make this app to track everyone’s location and warn them if they move ahead of their group and apart from this trip this application is also useful for family trip and friends tour.

1.3. SCOPE & OBJECTIVES

Our “Keep in Touch” application is mainly useful for the tourist group. In this application Leader can make a group and can track location of every member as well as he can restrict the area of travelling for his group. If anyone move outside the restricted area then every member will get notification or alarm alert. Everyone can also chatting via our application and can discuss about their trip or next destination.

1.4. TECHNOLOGY & LITERATURE REVIEW

We are planning to imply the project on Android Application. For Make group or join group we will use different java classes and android activities. For the database part or the application part we will be taking the help of Firebase database. For connection of data base we will use in built key in android studio. And this whole application will make out of MVVM(Model View View-Model) Structural design pattern. We would be learning about this technologies in depth using various resources and tutorials available on internet.

2. PROJECT MANAGEMENT

2.1. FEASIBILITY STUDY

2.1.1. Technical Feasibility

We are using Java Language to make our android application and it is compatible from 6.1 Nougat android version so that everyone can used with their older system also.

- MVVM model is used to develop separate functionality of our application.
- We also use Firebase database for data handling. There are many types of database in the firebase but we use only 2 types of database.1.Realtime Database 2.Firestore.

2.1.2. Time Scheduling Feasibility

The project has simple working and the basic requirement can be Satisfied within allowed time period so the time development feasibility is satisfied.

2.1.3. Operational Feasibility

Our application is mainly useful for the tourist group. In this application Leader can make a group and can track location of every member as well as he can restrict the area of travelling for his group. If anyone move outside the restricted area then every member will get notification or alarm alert. Everyone can also chat via our application and can discuss about their trip or next destination.

2.2. PROJECT MANAGEMENT

2.2.1. Project Management (Team, Role, Development Model)

Team: - We are two members in Group.

1) Dhruvil A. Chodvadiya (IT020 - 18ITUOS054)
2) Rajat U. Gajera (IT032 - 18ITUOS043)

Role: - we divide the work according to their specialization.

Team Member	**Role**
Dhruvil A. Chodvadiya	Implementation & Documentation
Rajat U. Gajera	Implementation & Testing

2.2.2. Project Development Approach and Justification

For project development the Iterative waterfall model is used.
It is a particular implementation of a software development life cycle that focuses on an initial, simplified implementation, which then progressively gains more complexity and a broader feature set until the final system is complete. In short, iterative development is a way of breaking down the software development of a large application into smaller pieces.

This model divides the cycle into the phases mentioned below:

1. Feasibility Study
2. Requirement analysis and specification
3. Design
4. Coding and unit testing
5. Integration and system testing
6. Maintenance

- Advantages of Iterative Waterfall Model

Feedback Path: In the classical waterfall model, there are no feedback paths, so there is no mechanism for error correction. But in iterative waterfall model feedback path from one phase to its preceding phase allows correcting the errors that are committed and these changes are reflected in the later phases.

Simple: Iterative waterfall model is very simple to understand and use. That's why it is one of the most widely used software development models.

- Drawbacks of Iterative Waterfall Model

Incremental delivery not supported: In the iterative waterfall model, the full software is completely developed and tested before delivery to the customer. There is no scope for any intermediate delivery. So, customers have to wait long for getting the software.

Overlapping of phases not supported: Iterative waterfall model assumes that one phase can start after completion of the previous phase, but in real projects, phases may overlap to reduce the effort & time needed to complete the project.

Risk handling not supported: Projects may suffer from various types of risks. But Iterative waterfall model has no mechanism for risk handling.

2.3. PROJECT PLAN

Date ——————— Phases	26-12-20	02-01-21	16-01-21	23-01-21	27-02-21	13-03-21
Project Definition	▨					
Feasibility Study		▨				
Analysis			▨			
Design & Diagrams				▨		
Implementation & Coding					▨	
Testing						▨

Project Plan and Mile stones

Sr. No.	Documents	Deadline
1	Project Definition, Submit to the department, Final approval:	26-12-2020
2	Feasibility study (Technical, Time Schedule, Learning, etc.)	2-1-2021
3	Analysis: Requirements Gathering, Analysis	2-1-2021
4	Analysis: SRS, Use Case, Wireframe/GUI design, DB design	16-1-2021
5	Design: Sequence Diagram, Class Diagram, MVC DB design	23-1-2021
6	Design: Component Diagram, Deployment Diagram	23-1-2021
7	Implementation-1: Coding, Unit Testing	30-1-2021
8	Implementation-2: Coding, Unit Testing	6-2-2021
9	Implementation-3: Coding, Unit Testing	20-2-2021
10	Implementation-4: Coding, Unit Testing	27-2-2021
11	Testing: Integration Testing, System Testing	6-3-2021
12	Testing: Integration Testing, System Testing	13-3-2021

13	Demonstration	Submission Week
14	Documentation: Report	Submission Week

3. SYSTEM REQUIREMENT STUDY

3.1. TECHNOLOGY

Frontend: Android

Backend: Android

Database: Firebase Database

3.2. HARDWARE AND SOFTWARE REQUIREMENTS

Hardware Requirement:

- Android Device with GPS and Internet.

Software Requirement:

- System must be 6.1 NOUGAT or Higher.

3.3. CONSTRAINT

- User have to make Account to Use application.
- Communicating detail will only be shared between Group.
- Only Group leader or Admin can set Safe Zone radius.
- Only two person (Admin and Red zone member) receives notification.

3.4. ASSUMPTIONS AND DEPENDENCIES

Assumptions:

- The assumption here made is that both users who wants to use this system have the mobile with GPS system and an active internet connection.

Dependencies:

- Group member must be joined in the group.

4. SYSTEM ANALYSIS

4.1. SYSTEM REQUIREMENTS (SRS)

4.1.1. Functional Requirements

Manage User

1. Signup

1.1. User Signup

- **State** : User already click on sign up option.
- **Input**: First name, Last name, Mobile No. Email-id, set Password
- **Processing:** Checks whether all the data is valid(according to rules of database.
- **Output** : After verification the user id is created.
- **Pre condition**: The customer should not be already registered. It has to be a new account.

2. Login

2.1.User Login

- **Discription:** It allows only authorized people to access the application. When the user logs in to the system, the user has to enter login details in the specific fields. The user clicks on login button and if the username and password are validated, then the user is given access to the application and user profile is displayed.
- **Pre-condition:** User must be registered before login.
- **State:** User has Email-Id.
- **Input :** Username , password
- **Processing :** verifying the details entered by the user.
- **Output :** Given that the user has entered correct details , user's profile dashboard is been displayed.
- **Post-Condition :** User is redirected to Home Page.
- **Next Function :** "You have logged in" email sent to the user's email.

2.2.Update Password

- **Pre Condition**: User must logged in.
- **Input** : Username, Old Password New Password.
- **Processing**: Verify username and Password.
- **Output**: Password changed successfully.
- **Post Condition:** User is redirected to profile page again.

2.3. Forgot Password

- **Description:** If the user forgets the password, it can be retrieved by
- Forgot Password? button. A click on this button will let to a security
- question and if the answer is verified. An OTP is send to the user on the
- registered email address and then using that OTP the user can set the

- new password.
- **Pre-Condition:** User must be registered.
- **Input:** Enter email address or username.
- **Processing:** Checks whether user data is valid and already registered
- in the database.
- **Output:** An email for password recovery is sent to user.
- **Post-Condition:** User is given option to reset password.

2.3.1. Reset Password

- **Pre-Condition:** User must get an email to recover password.
- **Input:** Enter new password and confirm the new password.
- **Processing:** Checks whether both the field contains same password.
- **Output:** Message – Password reset successfully.
- **Post-Condition:** New Password is updated in the database.

3. Home Screen

3.1. Join Group

- **Pre condition**: Group must be Created.
- **Description :** Given that Other member have selected the join the group option using code and they can join the group.
- **Input :** User Selection.
- **Output :** Joined the group using 6 digit code and he/she can able to see other member's live location.
- **Post Condition:** User redirected to home page and view as joined in group.

3.2. Create Group

- **Pre condition:** User must be logged in.
- **Description:** Given that Leader has selected the make group option leader will be able to create group.
- **Input:** User Selection.
- **Output:** Group will be created after One Code of 6 digit generation.
- **Post Condition:** User redirected to code generation Window.

3.3. User Profile

- **Description:** Contains all the details of the customer – First Name,
- Last Name, Mobile Number, Email Address.
- **Pre-Condition:** Customer must be logged in and have an account.
- **Input:** Select Profile from menu.
- **Processing:** Based on the selection data is fetched from database.
- **Output:** The selected page is displayed.

3.3.1. Update Profile

- **Description:** Member can change their profile details after the login to application and going to profile page.
- **Pre-Condition**: User must first have a profile.

- **Input:** Click on update Profile to update your profile detail like First name, Last name, email.
- **Processing:** Checks whether edited data satisfy the validation rules.
- **Output:** Profile is updated.
- **Post-Condition:** User Profile is updated in the database.

3.4. Select Group and Chatting

3.4.1. Message

- **Description:** Every member able to send message via send button and receive the message also.
- **Pre-condition:** Member must be in chatting window.
- **Input:** User Select the chat option
- **Output:** Send the message to group and receive the message in the group from any member.
- **Post condition:** Member is able to message again after sending the one message.

3.4.2. Member List

- **Description:** In this leader can see the total group member and outside safe zone group member.
- **Pre-condition:** Member must be in group information window
- **Input:** Select the map option.
- **Output:** The list of total members in group and the list of outside safe zone members.
- **Post condition:** if select chat option then able to chatting else window is steady but map is updated every 5 second.

4. Menu for Other Functionality:

4.1. Notification

- **Pre-condition**: App has a Permission for GPS Enable and Notification Enable and must be in group.
- **Input**: Automatic trace the location and check radius.
- **Outpu**t: Send the notification to Leader and Outside radius User.

4.2. Emergency Menu

- **Pre-condition:** User must be logged in and select the menu option to see this option.
- **Input**: Click on the menu.
- **Output**: Show the emergency contacts.

4.3. Survival Kit Menu

- **Pre condition:** User must be logged in and select the menu option to see this option.
- **Input:** Click on the menu.
- **Output**: Show some Survival trick for general issues.

4.4.FAQs

- **Pre-condition:** User must be logged in and select the menu option to see this option.
- **Input:** Click on the menu.
- **Output**: Show some General questions where most of users get confused or find difficulty.

4.5.Logout

- **Pre-Condition:** User must be logged in before select this.
- **Input:** Click on the menu
- **Output:** Member redirected to login page.
- **Post Condition:** User have to login for using the application.

4.1.2. Non-Functional Requirement

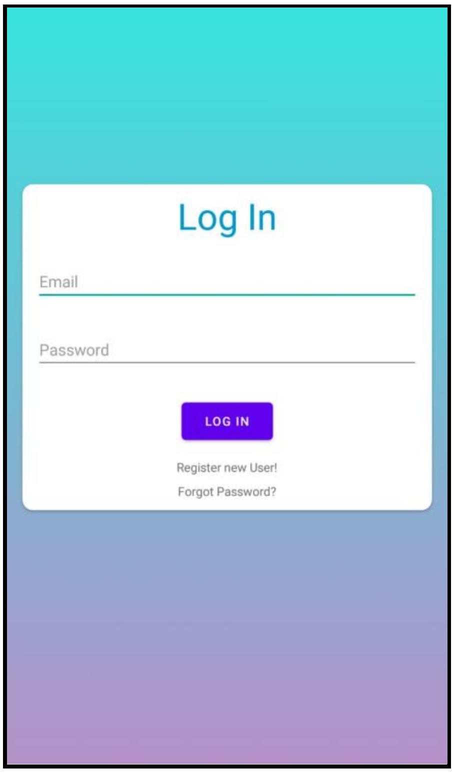
1. Performance Requirements:
   - User should be able to login within two seconds.
   - Fetched location should be displayed within 30 second.
   - The system shall be allowed to take more time when doing large processing jobs.
   - Responses to view information shall take no longer time to appear on the screen.
2. Safety Requirements:
   - System use shall not cause any harm to any users.
   - Data should be made confidential.
3. Security Requirements:
   - There should be well-defined password policy covering password change frequency, invalid attempts allowed, etc.
   - Account username and password should be stored using the good hashing algorithms.
   - All the input fields should be validated for the SQL injection.
4. Reliable:
   - The system shall provide 100% access reliability
5. Scalable:
   - The system should manage many more user at the time.
6. Maintainability:
   - All code artifacts should have proper documentation.
   - All code components should be thoroughly tested.
7. Usability:
   - The system is user friendly which makes the system easy.
8. Efficiency:
   - Even if the system fails, the system will be recovered back up within an hour or less.

5. SYSTEM DESIGN

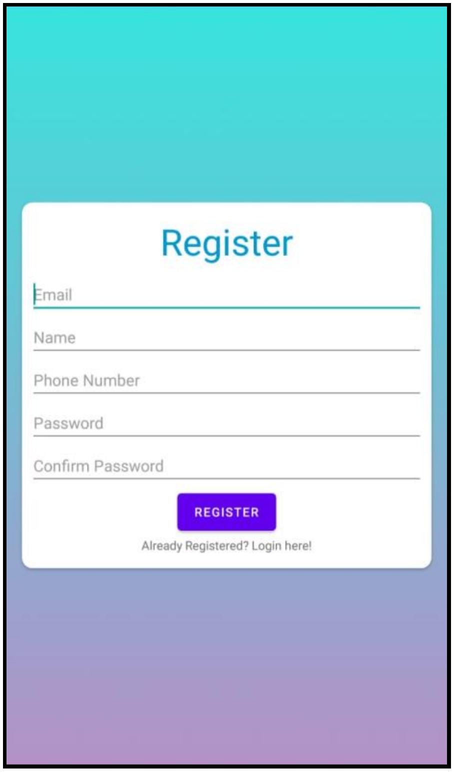
5.1 Wireframe Design

5.1.1. User

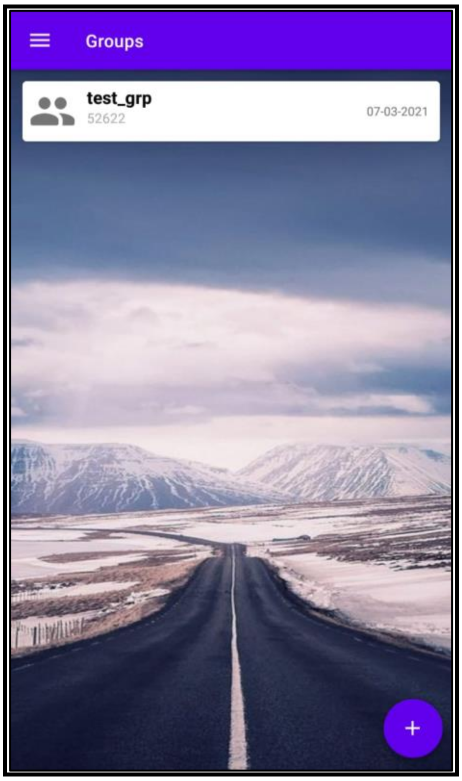
Login



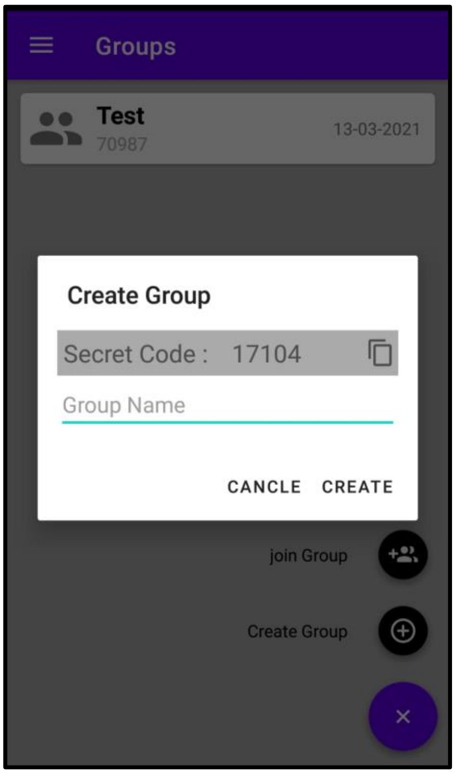
Sign Up



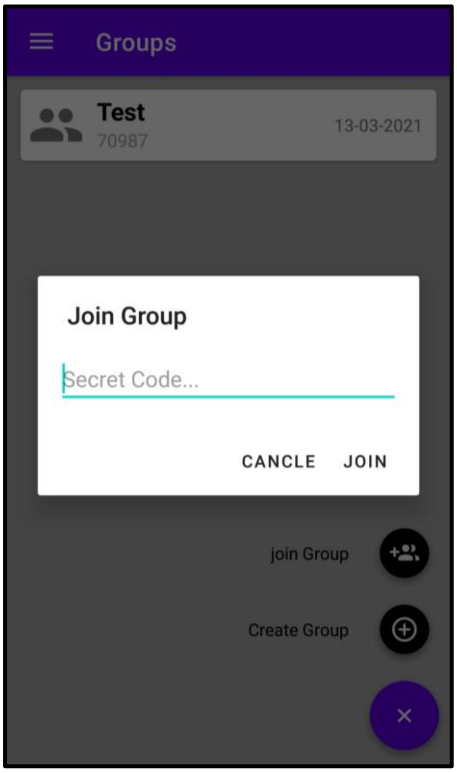
Home



Group Create



Group Join

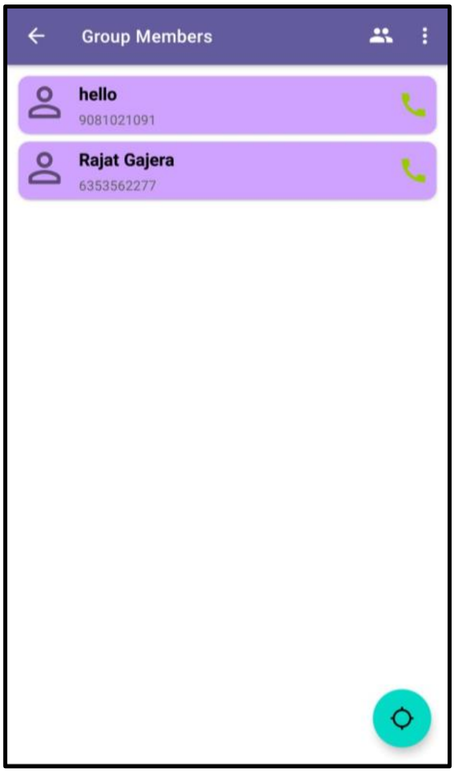


Test
70987
13-03-2021

Join Group

Secret Code...

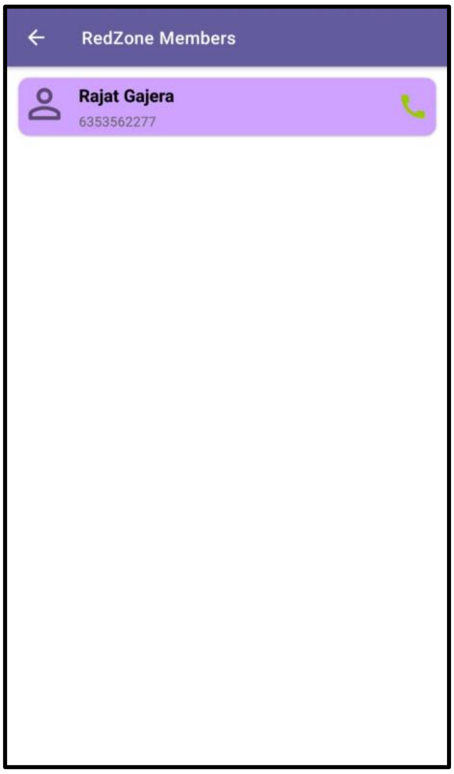
Group Window



hello
9081021091

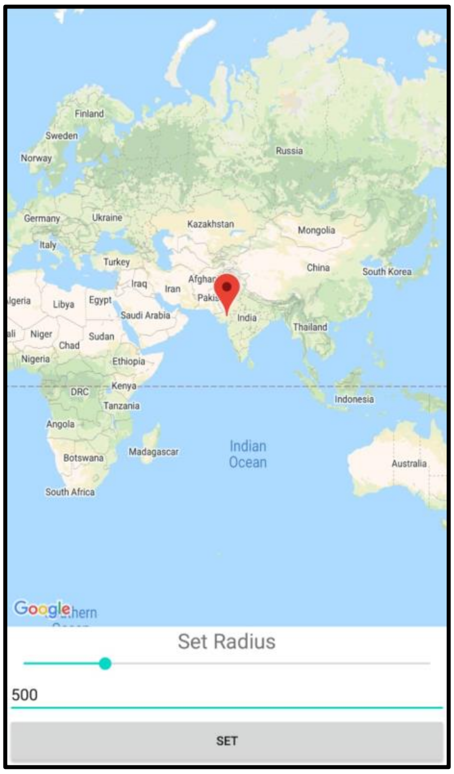
Rajat Gajera
6353562277

RedZone Members



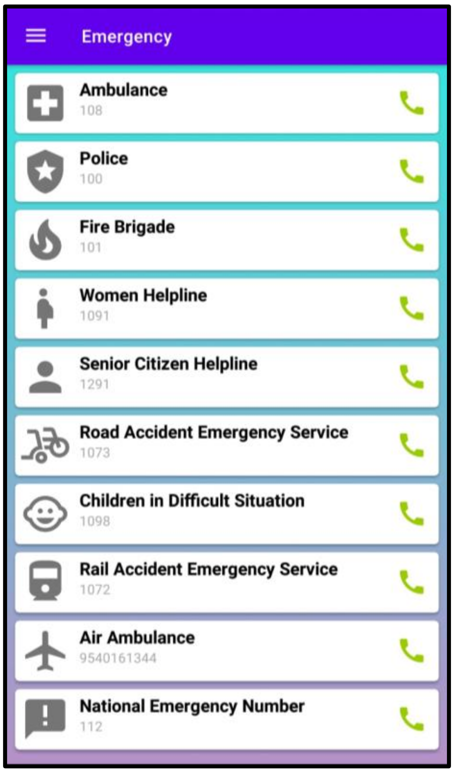
Rajat Gajera
6353562277

Map and Set Radius



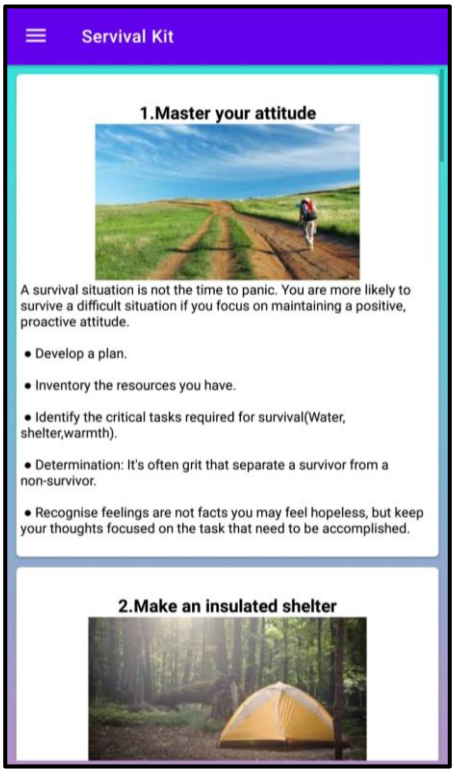
500

Emergency



Ambulance	108
Police	100
Fire Brigade	101
Women Helpline	1091
Senior Citizen Helpline	1291
Road Accident Emergency Service	1073
Children in Difficult Situation	1098
Rail Accident Emergency Service	1072
Air Ambulance	9540161344
National Emergency Number	112

Survival Kit



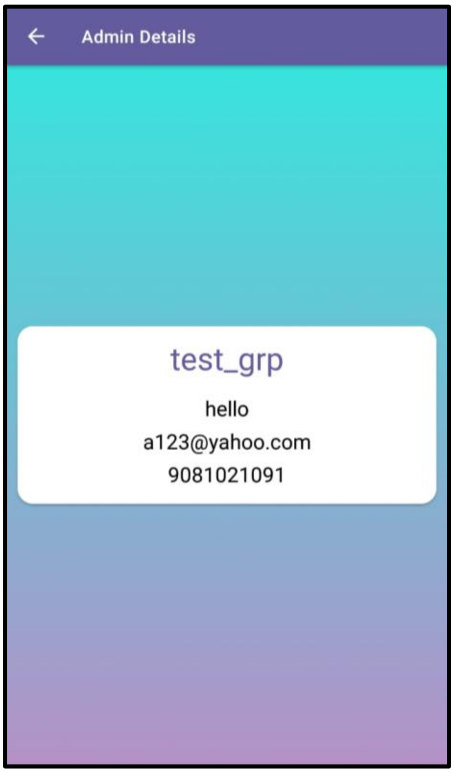
1.Master your attitude

A survival situation is not the time to panic. You are more likely to survive a difficult situation if you focus on maintaining a positive, proactive attitude.

- Develop a plan.
- Inventory the resources you have.
- Identify the critical tasks required for survival(Water, shelter,warmth).
- Determination: It's often grit that separate a survivor from a non-survivor.
- Recognise feelings are not facts you may feel hopeless, but keep your thoughts focused on the task that need to be accomplished.

2.Make an insulated shelter

Admin Details



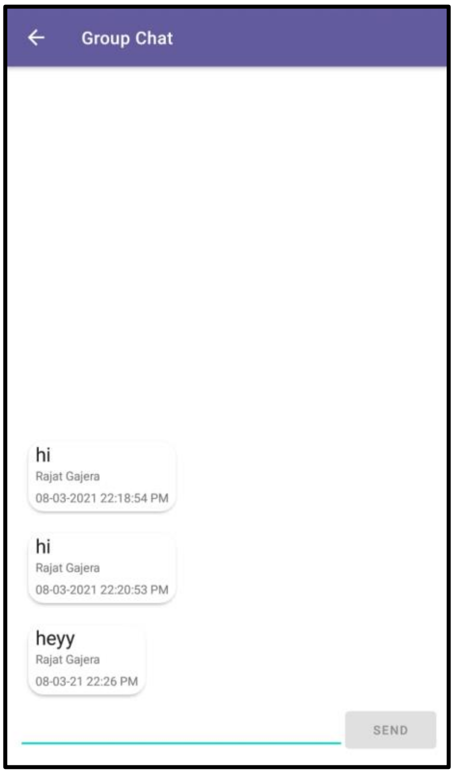
test_grp

hello

a123@yahoo.com

9081021091

Group Chat

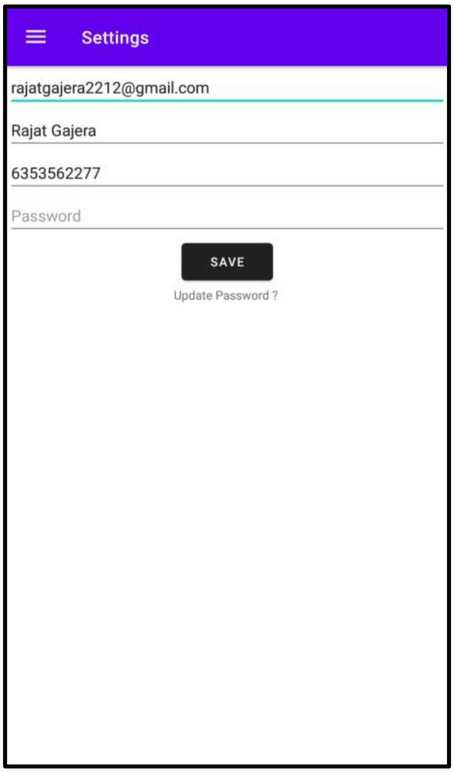


hi
Rajat Gajera
08-03-2021 22:18:54 PM

hi
Rajat Gajera
08-03-2021 22:20:53 PM

heyy
Rajat Gajera
08-03-21 22:26 PM

Settings



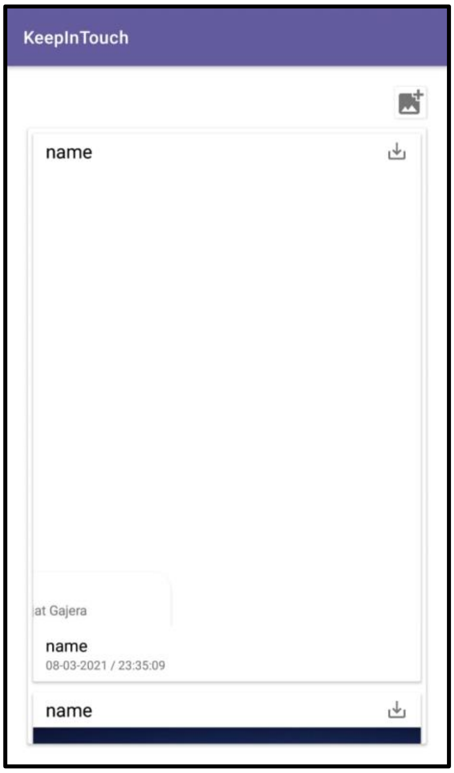
rajatgajera2212@gmail.com

Rajat Gajera

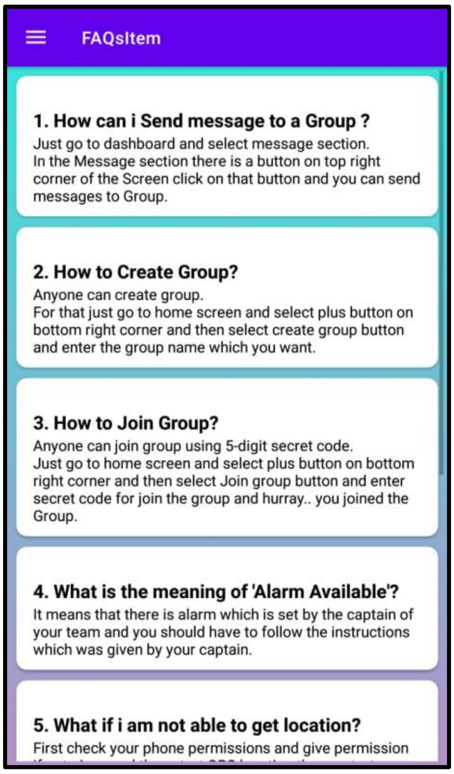
6353562277

Password

Gallery



FAQs



1. How can i Send message to a Group ?
Just go to dashboard and select message section.
In the Message section there is a button on top right corner of the Screen click on that button and you can send messages to Group.

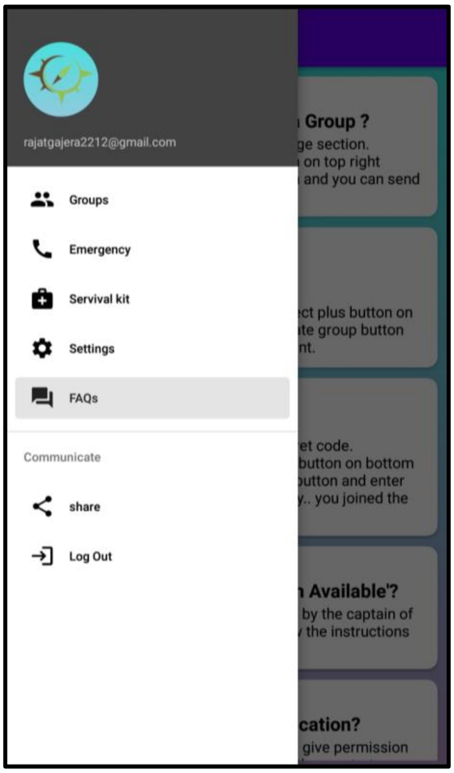
2. How to Create Group?
Anyone can create group.
For that just go to home screen and select plus button on bottom right corner and then select create group button and enter the group name which you want.

3. How to Join Group?
Anyone can join group using 5-digit secret code.
Just go to home screen and select plus button on bottom right corner and then select Join group button and enter secret code for join the group and hurray.. you joined the Group.

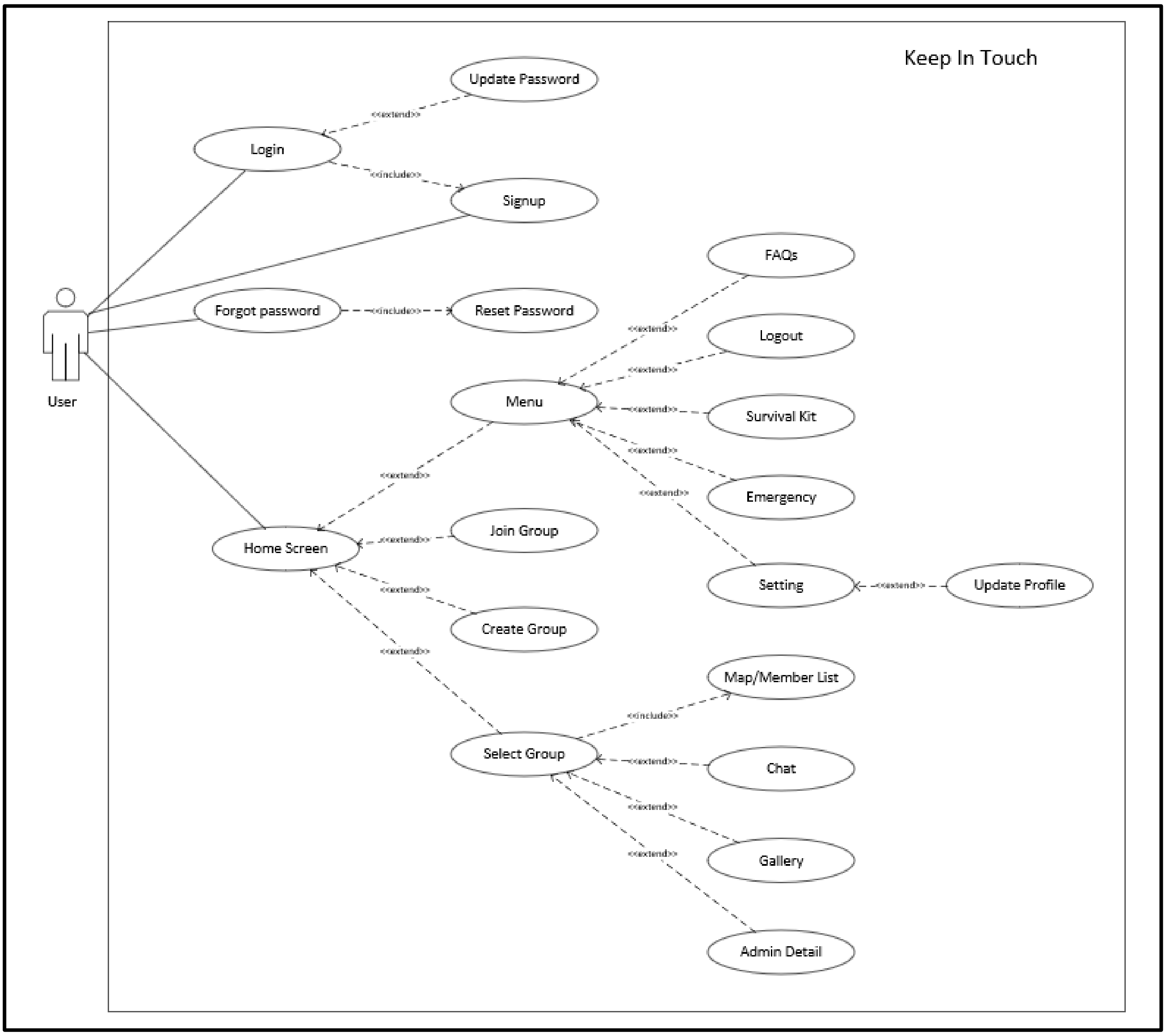
4. What is the meaning of 'Alarm Available'?
It means that there is alarm which is set by the captain of your team and you should have to follow the instructions which was given by your captain.

5. What if i am not able to get location?
First check your phone permissions and give permission

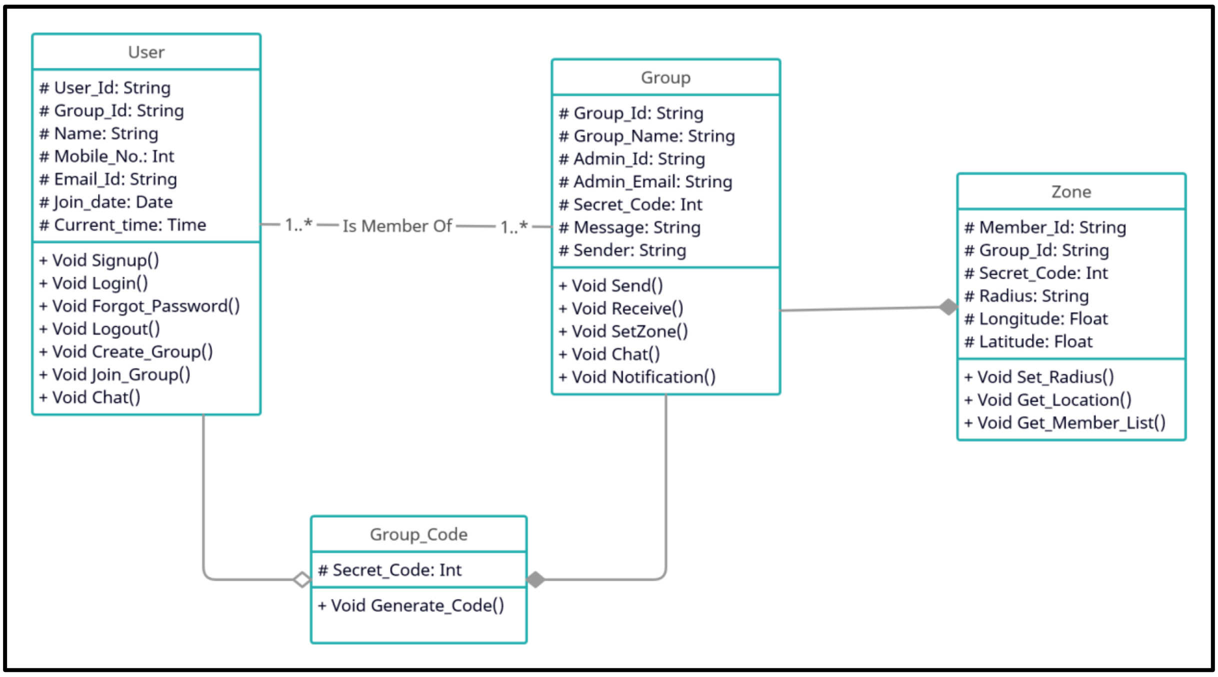
Menu



5.2. Use Case Diagram

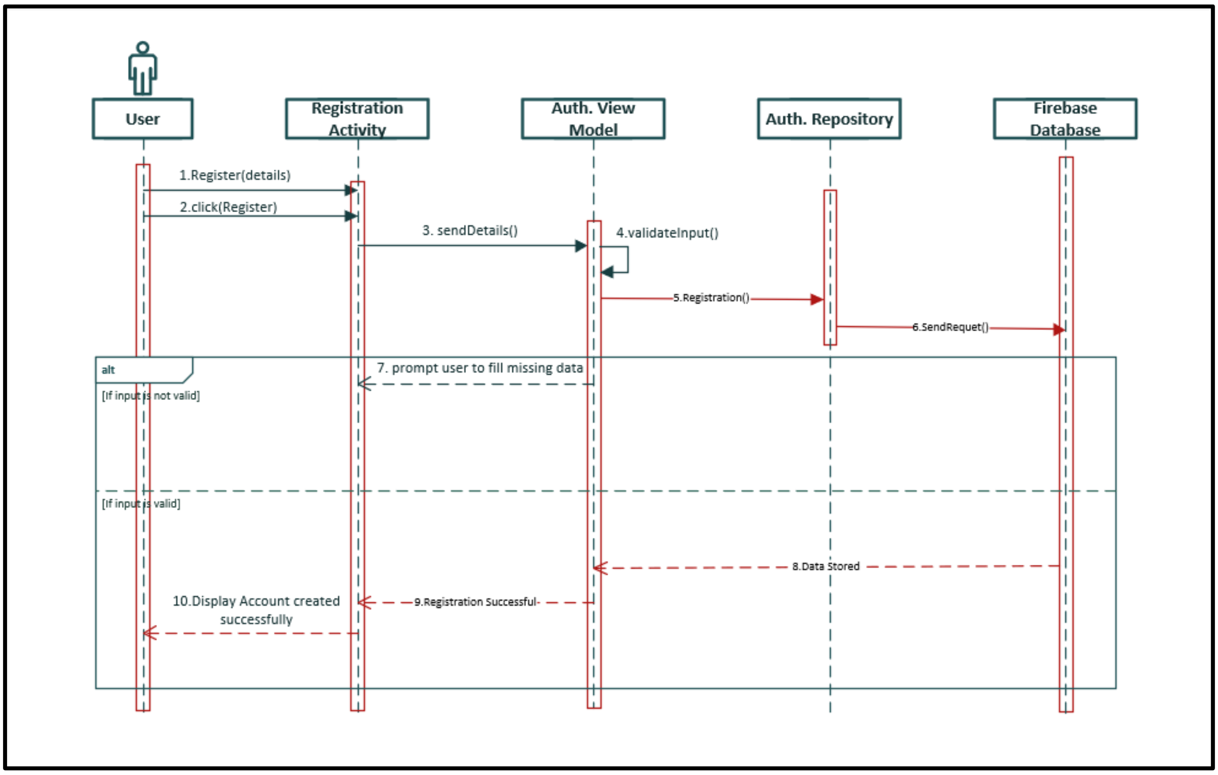


5.3. Class Diagram

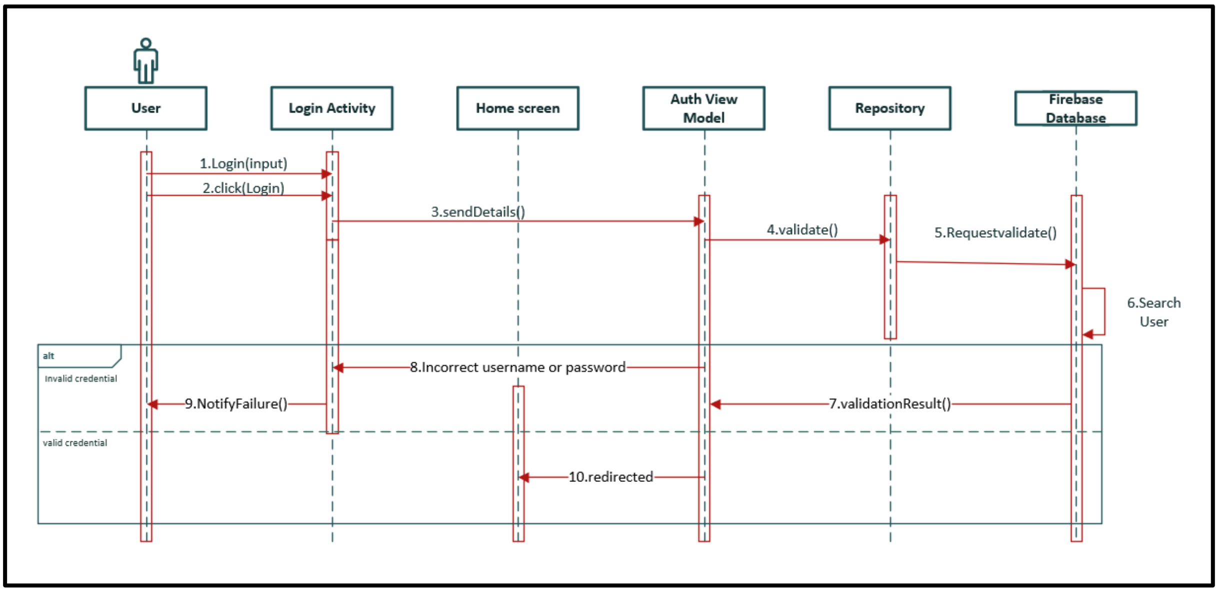


5.4. Sequence Diagram

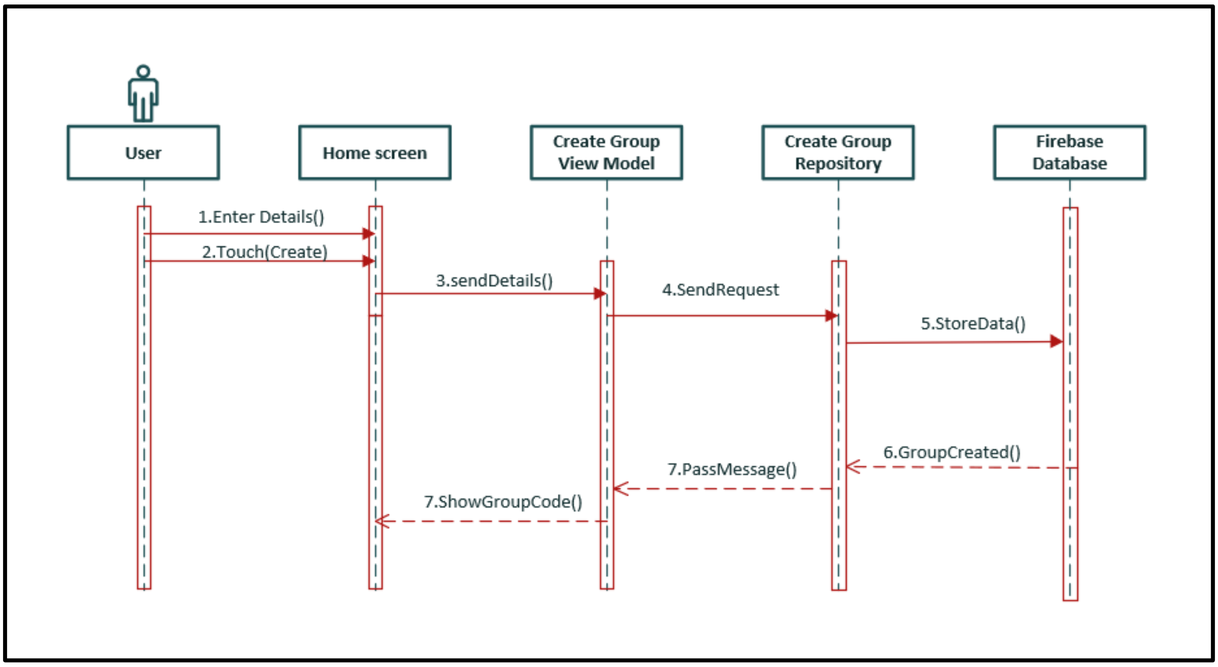
5.4.1. Sign Up Activity



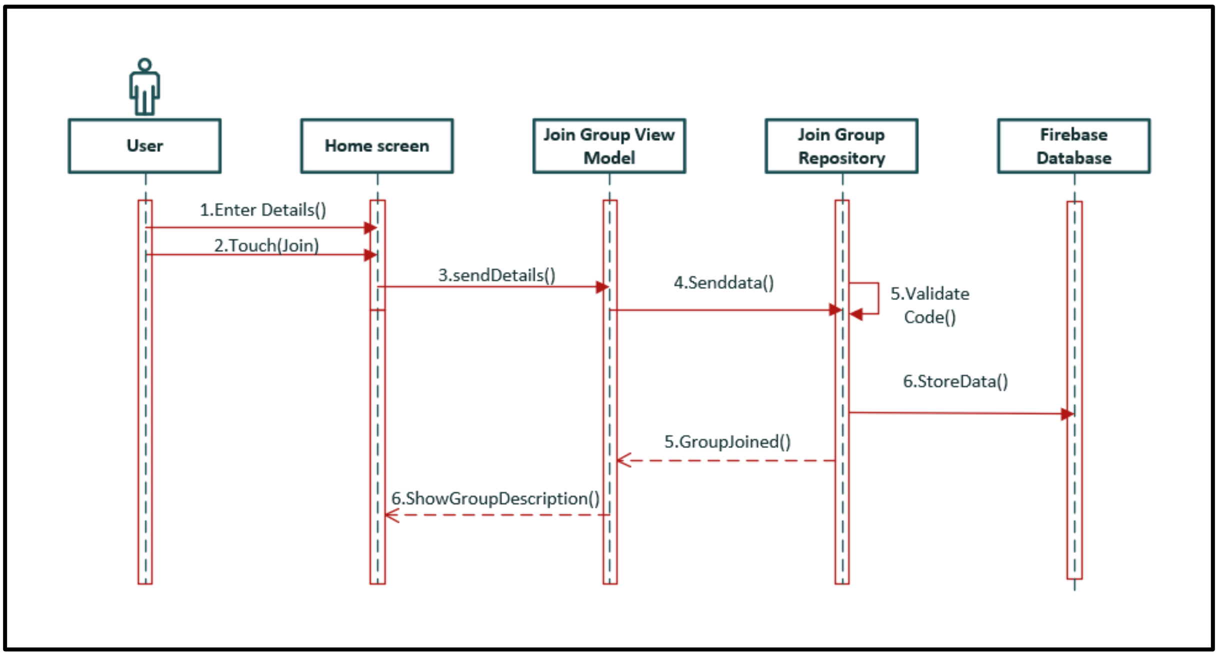
5.4.2. Login Activity



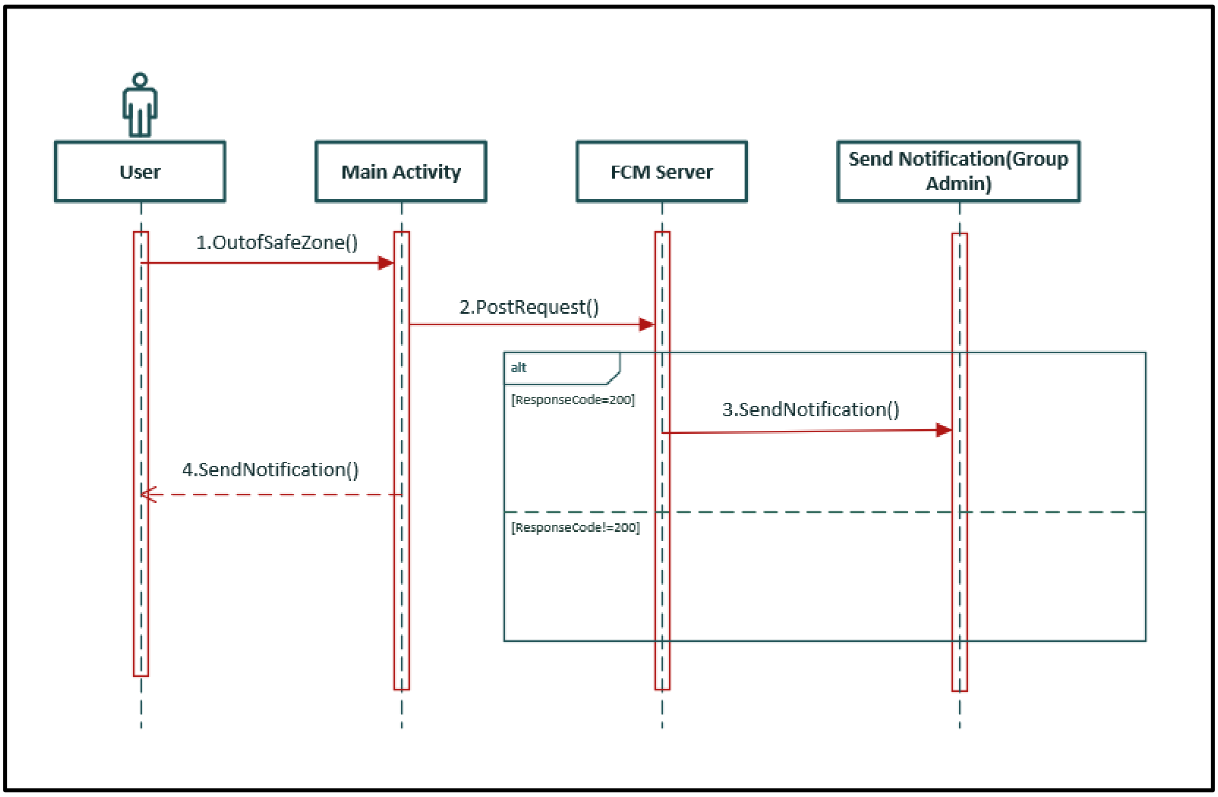
5.4.3. Group Create Activity



5.4.4. Group Join Activity



5.4.5. Send Notification Activity



6. IMPLEMENTATION PLANNING

6.1 Installation

1) Download Android Studio executable file or zip file from https://developer.android.com/studio/#downloads and install it on your system by setting the path where you want to install.

2) Meanwhile Android Studio will be finding available SDK components. After it has found the SDK component download the SDK component. Here we will download SDK 29.

3) Install Google Play Services form SDK manager.

4) Set Up Virtual Device from AVD manager to run out application or connect android device with USB debugging enabled.

6.2 Implementation

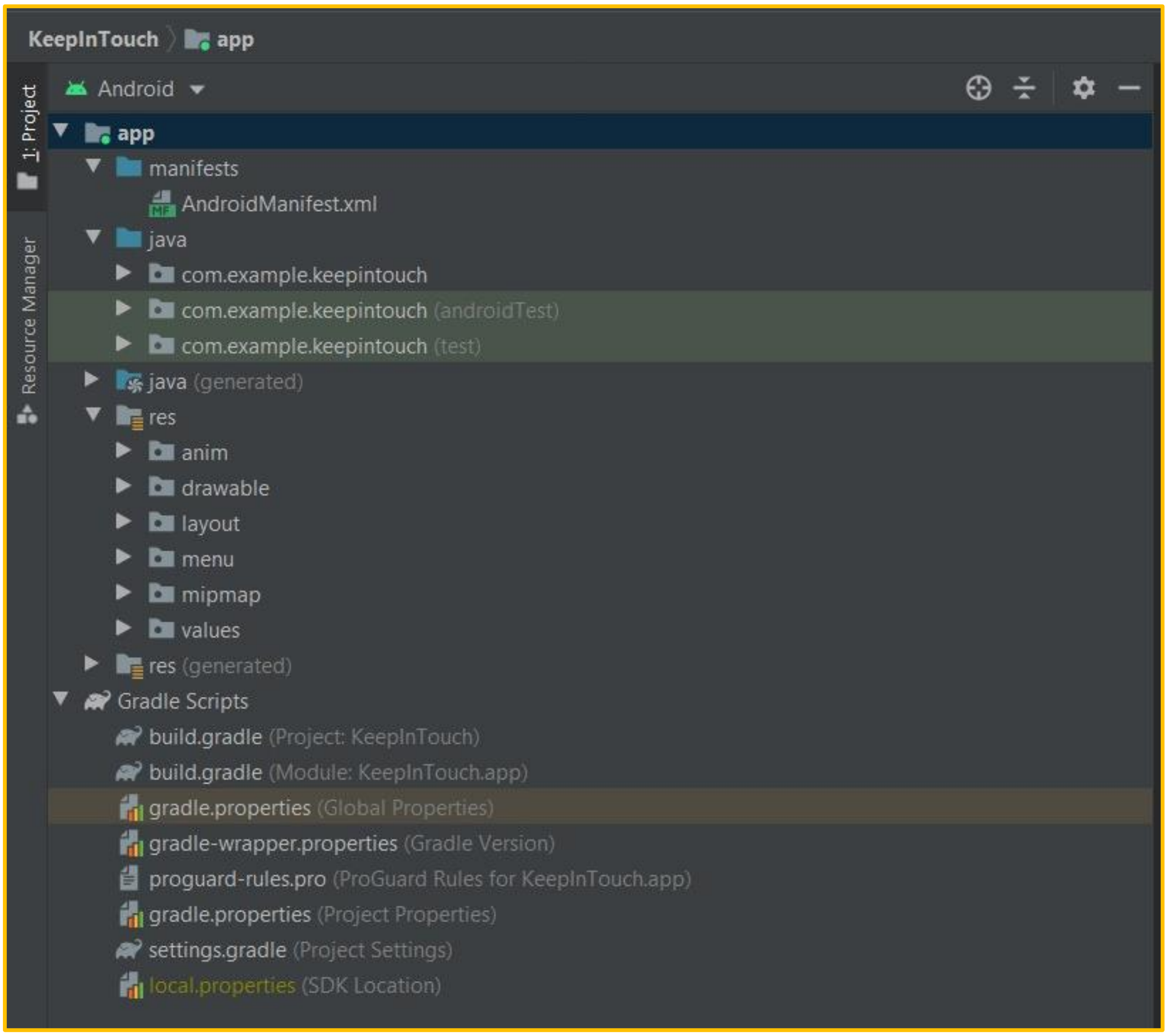
1) Now, Create new project with Empty Activity and we will Select java language.

2) Create Firebase Project on Firebase Console and connect it with our application from Tools Option in android studio.

3) We will be going to user Firebase Authentication, Firebase Firestore Database, Firebase Realtime Database & Firebase Cloud Messaging So Add all dependency in Gradle File.

4) Add Required Permissions in Android Manifest File.

6.2.1 Code Structure -1



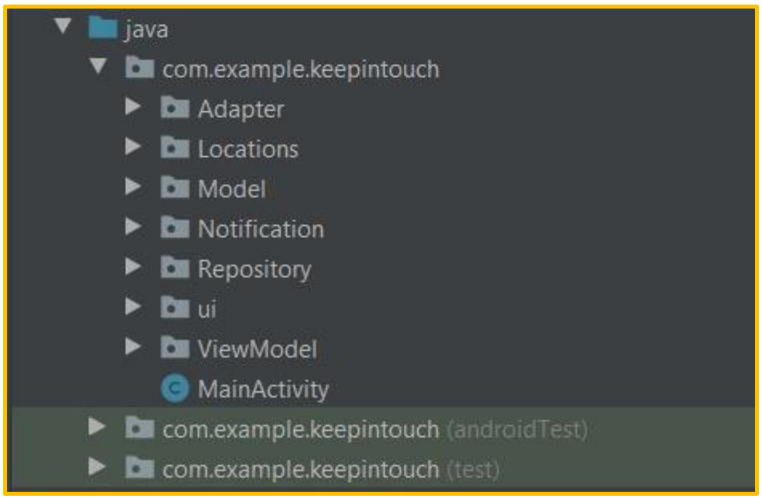
```
▼ app
    ▼ manifests
        AndroidManifest.xml
    ▼ java
        ▶ com.example.keepintouch
        ▶ com.example.keepintouch (androidTest)
        ▶ com.example.keepintouch (test)
    ▶ java (generated)
    ▼ res
        ▶ anim
        ▶ drawable
        ▶ layout
        ▶ menu
        ▶ mipmap
        ▶ values
    ▶ res (generated)
▼ Gradle Scripts
    build.gradle (Project: KeepInTouch)
    build.gradle (Module: KeepInTouch.app)
    gradle.properties (Global Properties)
    gradle-wrapper.properties (Gradle Version)
    proguard-rules.pro (ProGuard Rules for KeepInTouch.app)
    gradle.properties (Project Properties)
    settings.gradle (Project Settings)
    local.properties (SDK Location)
```

Picture shown above is the whole file/Package structure of our application.

The **AndroidManifest.xml file** contains information of your package, including components of the application such as activities, services, broadcast receivers, content providers etc.

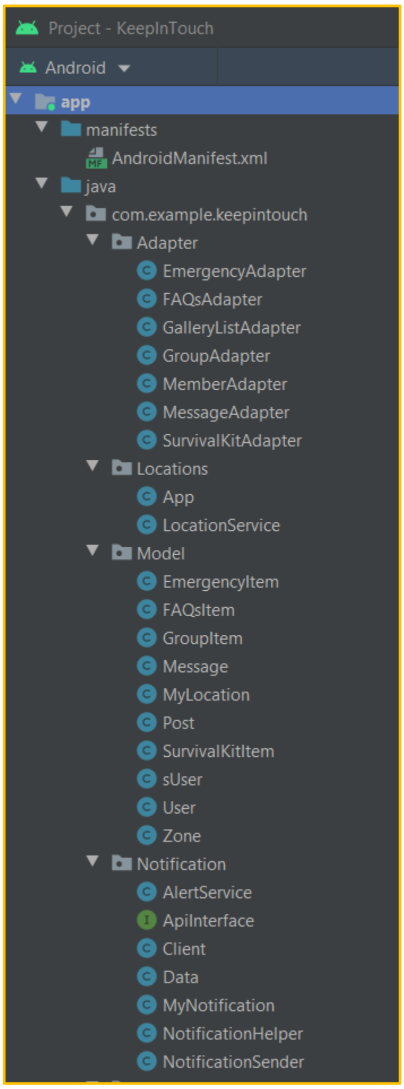
- It is responsible to protect the application to access any protected parts by providing the permissions.
- It also declares the android api that the application is going to use.
- It lists the instrumentation classes. The instrumentation classes provide profiling and other information. These information is removed just before the application is published etc.

6.2.2 Code Structure – 2 (Java Files):

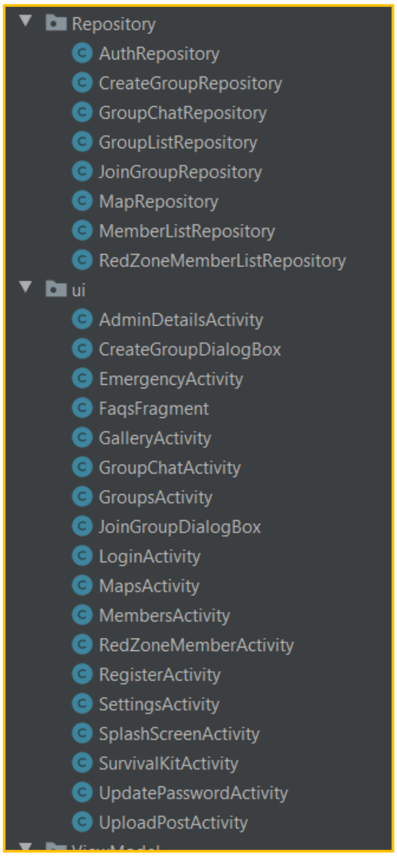


```
▼ java
    ▼ com.example.keepintouch
        ▶ Adapter
        ▶ Locations
        ▶ Model
        ▶ Notification
        ▶ Repository
        ▶ ui
        ▶ ViewModel
          MainActivity
    ▶ com.example.keepintouch (androidTest)
    ▶ com.example.keepintouch (test)
```

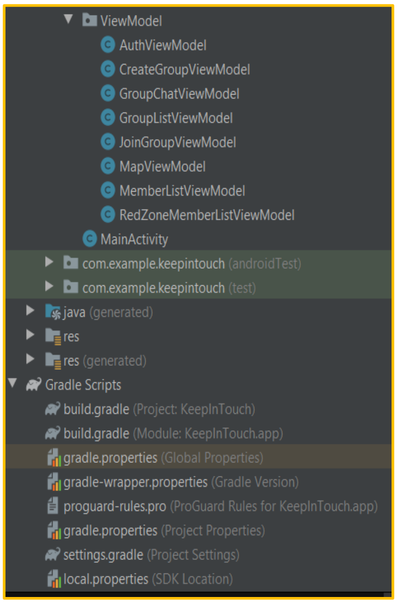
6.2.3 Code Structure – 3 (Adapter, Repository, View Model) :



```
Project - KeepInTouch
Android ▾
▼ app
    ▼ manifests
          AndroidManifest.xml
    ▼ java
        ▼ com.example.keepintouch
            ▼ Adapter
                  EmergencyAdapter
                  FAQsAdapter
                  GalleryListAdapter
                  GroupAdapter
                  MemberAdapter
                  MessageAdapter
                  SurvivalKitAdapter
            ▼ Locations
                  App
                  LocationService
            ▼ Model
                  EmergencyItem
                  FAQsItem
                  GroupItem
                  Message
                  MyLocation
                  Post
                  SurvivalKitItem
                  sUser
                  User
                  Zone
            ▼ Notification
                  AlertService
                  ApiInterface
                  Client
                  Data
                  MyNotification
                  NotificationHelper
                  NotificationSender
```



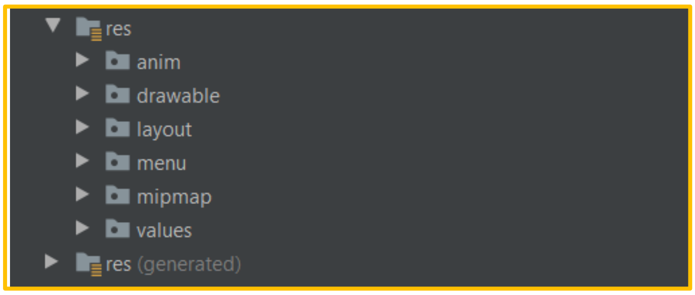
```
▼ Repository
      AuthRepository
      CreateGroupRepository
      GroupChatRepository
      GroupListRepository
      JoinGroupRepository
      MapRepository
      MemberListRepository
      RedZoneMemberListRepository
▼ ui
      AdminDetailsActivity
      CreateGroupDialogBox
      EmergencyActivity
      FaqsFragment
      GalleryActivity
      GroupChatActivity
      GroupsActivity
      JoinGroupDialogBox
      LoginActivity
      MapsActivity
      MembersActivity
      RedZoneMemberActivity
      RegisterActivity
      SettingsActivity
      SplashScreenActivity
      SurvivalKitActivity
      UpdatePasswordActivity
      UploadPostActivity
```



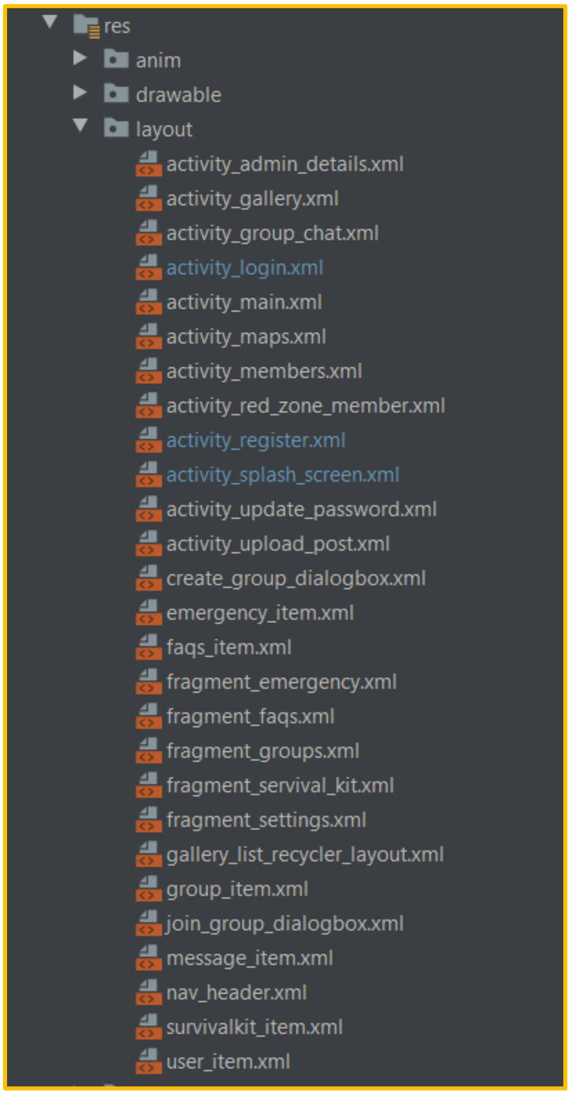
```
            ▼ ViewModel
                  AuthViewModel
                  CreateGroupViewModel
                  GroupChatViewModel
                  GroupListViewModel
                  JoinGroupViewModel
                  MapViewModel
                  MemberListViewModel
                  RedZoneMemberListViewModel
              MainActivity
        ▶ com.example.keepintouch (androidTest)
        ▶ com.example.keepintouch (test)
    ▶ java (generated)
    ▶ res
    ▶ res (generated)
▼ Gradle Scripts
      build.gradle (Project: KeepInTouch)
      build.gradle (Module: KeepInTouch.app)
      gradle.properties (Global Properties)
      gradle-wrapper.properties (Gradle Version)
      proguard-rules.pro (ProGuard Rules for KeepInTouch.app)
      gradle.properties (Project Properties)
      settings.gradle (Project Settings)
      local.properties (SDK Location)
```

Pictures shown above are the whole file stucture of all Activities, Adapter, Model , ViewModel & Repositories of Out Application.

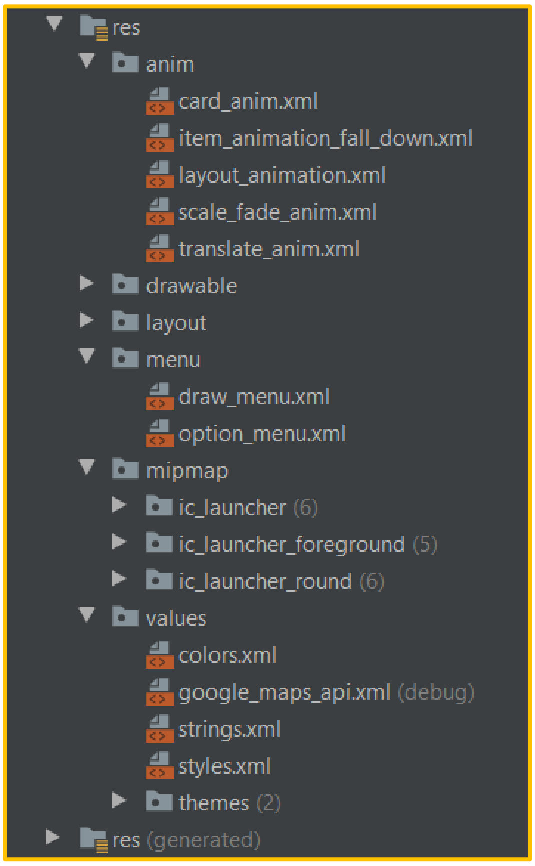
6.2.4 Code Structure – 4 (XML Files) :



```
▼ res
    ▶ anim
    ▶ drawable
    ▶ layout
    ▶ menu
    ▶ mipmap
    ▶ values
▶ res (generated)
```



```
▼ res
    ▶ anim
    ▶ drawable
    ▼ layout
          activity_admin_details.xml
          activity_gallery.xml
          activity_group_chat.xml
          activity_login.xml
          activity_main.xml
          activity_maps.xml
          activity_members.xml
          activity_red_zone_member.xml
          activity_register.xml
          activity_splash_screen.xml
          activity_update_password.xml
          activity_upload_post.xml
          create_group_dialogbox.xml
          emergency_item.xml
          faqs_item.xml
          fragment_emergency.xml
          fragment_faqs.xml
          fragment_groups.xml
          fragment_servival_kit.xml
          fragment_settings.xml
          gallery_list_recycler_layout.xml
          group_item.xml
          join_group_dialogbox.xml
          message_item.xml
          nav_header.xml
          survivalkit_item.xml
          user_item.xml
```



```
▼ res
    ▼ anim
          card_anim.xml
          item_animation_fall_down.xml
          layout_animation.xml
          scale_fade_anim.xml
          translate_anim.xml
    ▶ drawable
    ▶ layout
    ▼ menu
          draw_menu.xml
          option_menu.xml
    ▼ mipmap
        ▶ ic_launcher (6)
        ▶ ic_launcher_foreground (5)
        ▶ ic_launcher_round (6)
    ▼ values
          colors.xml
          google_maps_api.xml (debug)
          strings.xml
          styles.xml
        ▶ themes (2)
▶ res (generated)
```

Pictures shown above are the whole files sturcture of our basic activity’s layout file, Animation file, drawable files and values and photos which are used in our application.

7. SOFTWARE TESTING

7.1 Introduction

Software testing is the process of testing the functionality and correctness of software. Software testing is defined as an activity to check whether the actual results match the expected results and to ensure that the software system is defect free.

7.2 Unit Testing

In this each module is tested individually, Criteria selected for identifying unit test module is to identify module that has core functionality implementation Module could be an individual or procedure. The following is a list of functions for unit testing that will tested:

- Select the dataset.
- Apply Pre-processing.
- Build Individual model.
- Train classification model.
- Test classification model.

7.3 Integration Testing

Integration testing integrates individual modules and tested as a group Integration testing takes as its input modules that have been unit tested, groups them in larger aggregates applies tests defined in an integration test plan to those aggregates and delivers as its output the integrated system for testing.

7.4 Validation Testing

The process of evaluating software during or at the end of the development process in to determine whether it satisfies specified requirements.

7.5 GUI Testing

GUI testing is the process of testing the system's Graphical User Interface of the Application under Test. GUI testing involves checking the screens with the controls like menus, buttons, icons, and all types of bars-menu bar, dialog boxes etc.

7.5.1. Test Cases

Sr. No	Module	Input	State	Expected Output	Actual Output	Test Result
1	Login	Valid Email & Password, Submit	Logout	Success	Success	Pass
2	Login	Valid Email, submit	Logout	All Fields are mandatory	All Fields are mandatory	Pass
3	Login	Valid Email, Invalid Password, Submit	Logout	Login Failed	Login Failed	Pass
4	Login	Valid Email & Password, Submit	Logged In	Previous Session Exists	Previous Session Exists	Pass
5	Register	Enter data except Mobile no.	Logout	All fields are mandatory	Success	Fail
6	Group Create	Press Group Create Button	Logged In	Successfully Creates	Successfully Created	Pass
7	Group Join	Enter 5 digit code and Press Join Button	Logged In	Successfully Joined	Successfully Joined	Pass
8	Group Join	Enter Wrong code or more/less no. of digits	Logged In	Group Not joined	Group Not Joined	Pass
9	Set Radius	Go to Location and try to set radius	Logged In	Radius set successfully	Radius set successfully	Pass

8. USER MANUAL

8.1 Splash Screen

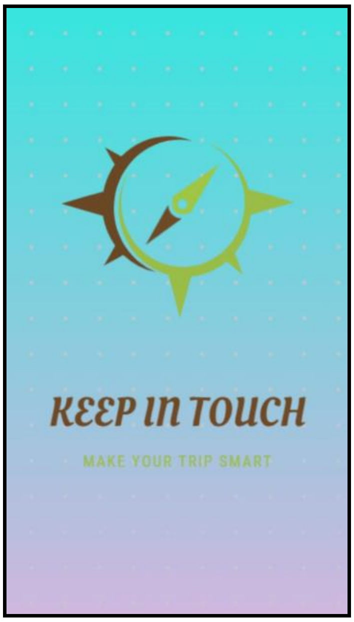


Fig.8.1 Splash Screen

- Picture shown above is the Splash Screen of Application.This is the very first Screen of Our Application.When you start or click on our application you get this screen firstly.

8.2 Registration Screen

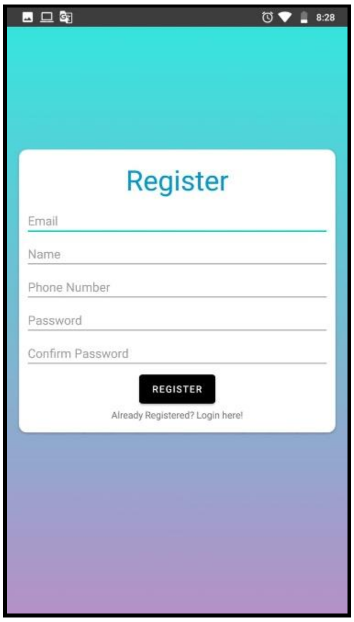
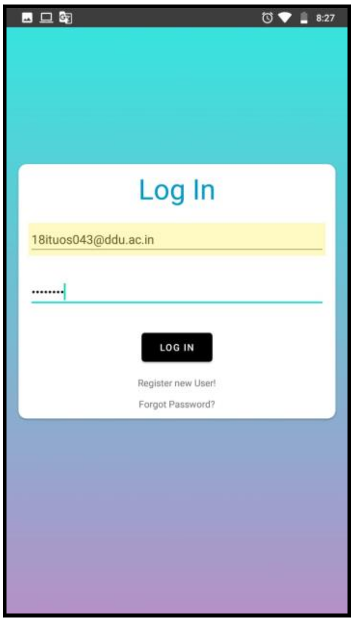


Fig 8.2 Registration Screen

- Picture shown above is Registration Screen. For Successful registration user needs to enter necessary details.
- Input : User Details like name,emailid,phone number and password for security.

- Output : Home Screen.
  (If Registration Successfull)

8.3 Login Screen



18ituos043@ddu.ac.in

••••••••

Fig. 8.3 Login Screen

- Picture shown above is Login Screen. For Successful Login User needs to enter necessary details email and Password.
- Input : User Details
- Output : HomeScreen
  (If LogIn Successfull)

8.4 Home Screen

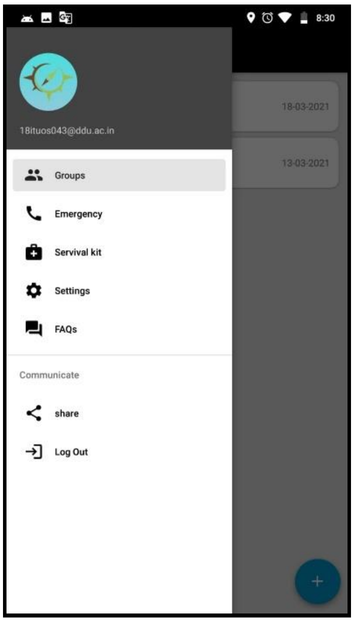


Fig 8.4 Home Screen

- Picture shown above is Home Screen. User can Open this navigation Bar by clicking navigation button from top left corner. Through this Navigation Bar User gets more knowledge about this application like how it’s works and he/she can access many other options placed there.

8.5 Groups

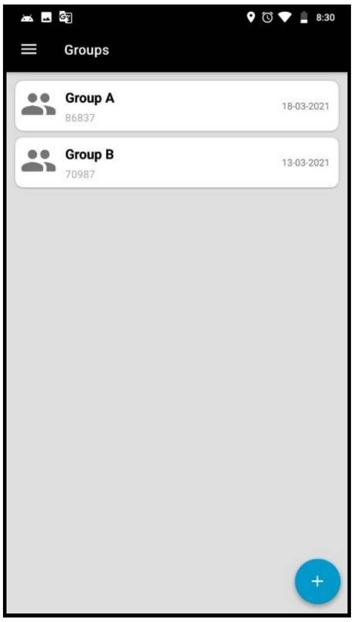
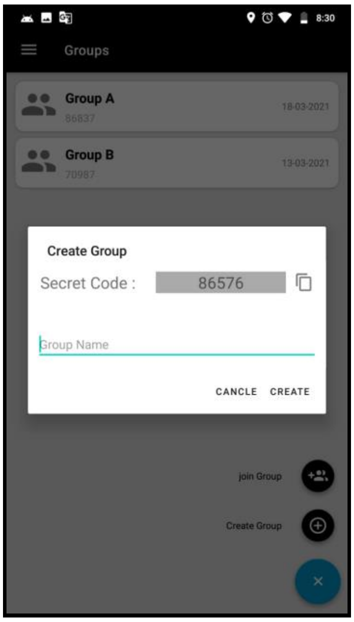


Fig 8.5 Groups

- Picture shown above is the main Screen which shows the list of all groups in which user has joined. And he/she can also delete unneccessary groups by long press on that.

8.6 Create Group



Secret Code : 86576

Fig 8.6 Create Group

- Picture shown above is the create group dialog box, user can open it by clicking floating action button at bottom right corner of Groups Screen.And user will get Secret Unique code of 5 Digit to join the group.
- Input : Group Name
- Output : Group Created

8.7 Join Group

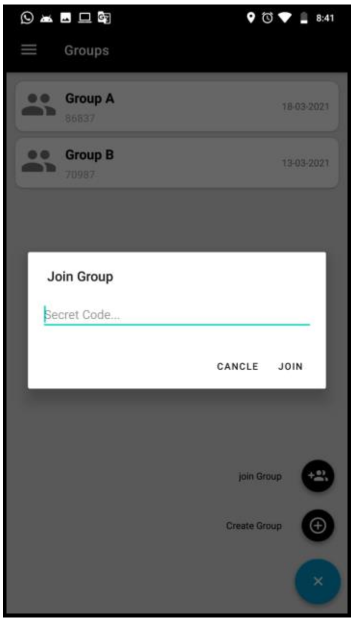
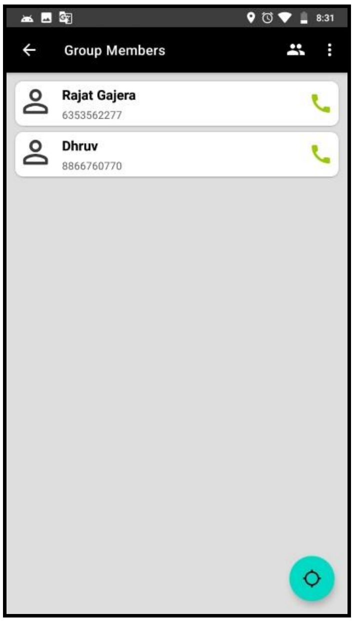


Fig 8.7 Join Group

- Picture shown above is the Join group dialog box, user can open it by clicking flaoting action button at bottom right corner of Groups Screen. And enter 5 digit secret code to join group.
- Input : Secret Unique Code for join the group
- Output : Group Joined

8.8 Group Members



Group Members

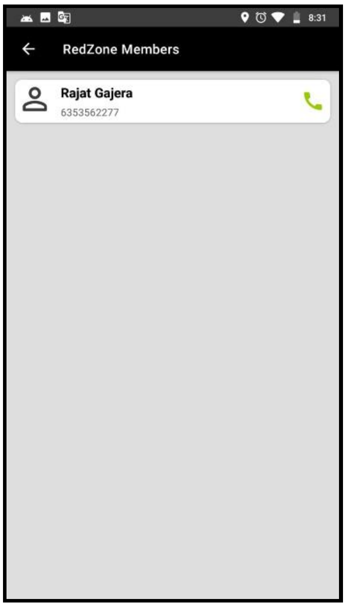
Rajat Gajera
6353562277

Dhruv
8866760770

8.8 Group Members Screen

- Picture shown above is the group members screen which shows the list of all group members.You can also call any group member by clicking the call icon which is placed to right side of every memebr’s name.

8.9 RedZone Members



RedZone Members

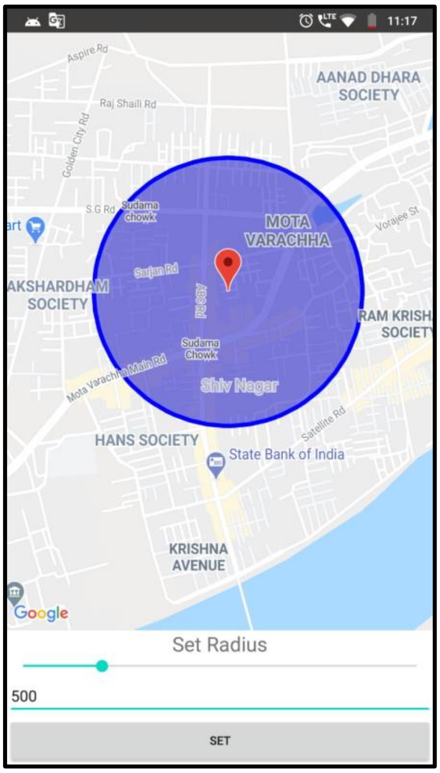
Rajat Gajera
6353562277

Fig 8.9 RedZone Members Screen

- Picture shown above is RedZone Members Screen which shows the list of group members who are out of safe zone, user can open it by clicking the redzone member menu from app bar in group members screen.

8.10 Map Screen

8.10.1 Map (Admin)



8.10.2 Map (Members)

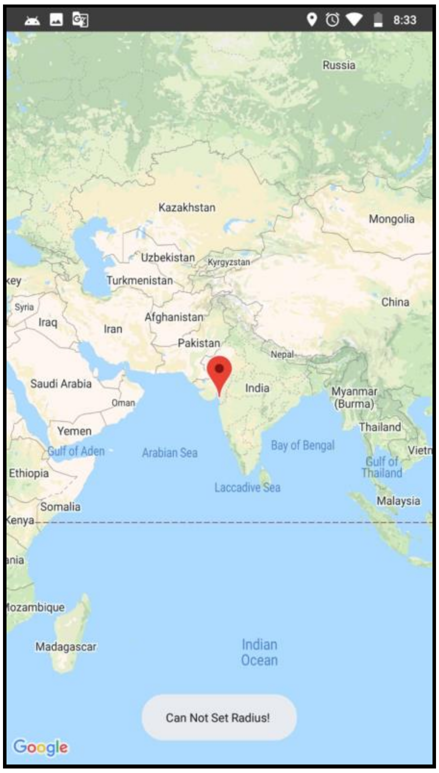
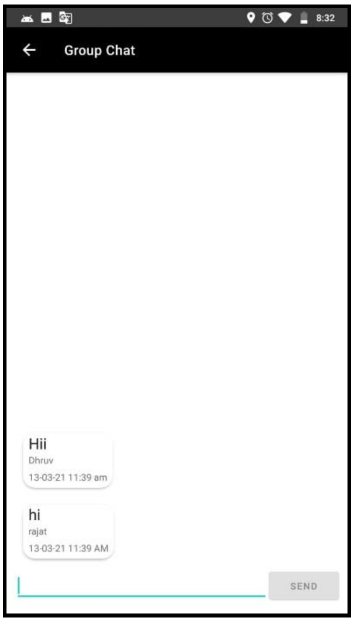


Fig 8.10 Map Screen

- Pictures shown above are the map Screen which shows live location of all members and zone.
- Fig 8.10.1 is for Admin who has the only authority to change the radius of the their group’s Zone
- Fig 8.10.2 is for members who can not set radius. And he/she can only view the location of every memebers and group leader.

8.11 Group Chat



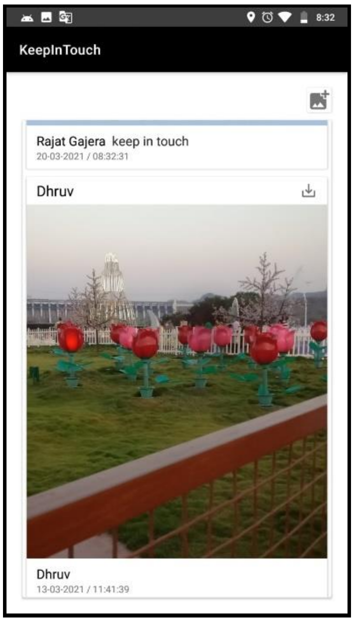
Hii
Dhruv
13-03-21 11:39 am

hi
rajat
13-03-21 11:39 AM

Fig 8.11 Group Chat Screen

- Picture shown above is the Group Chat Screen where all group member can do chatting. User can open it from menu from top right corner.

8.12 Gallery



Rajat Gajera keep in touch
20-03-2021 / 08:32:31

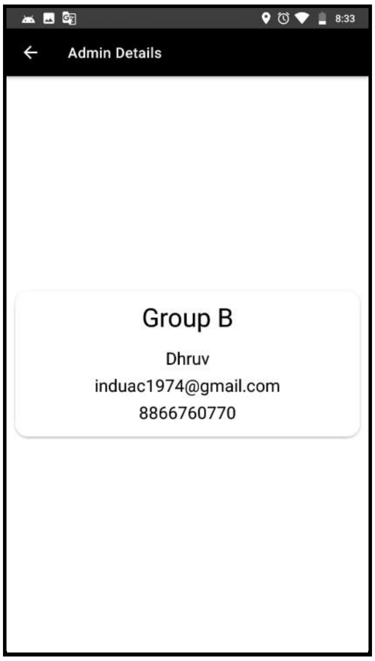
Dhruv

Dhruv
13-03-2021 / 11:41:39

Fig 8.12 Gallery Screen

- Picture shown above is the gallery screen where all members can share photos with original size and clearity.

8.13 Admin Details



Admin Details

Group B

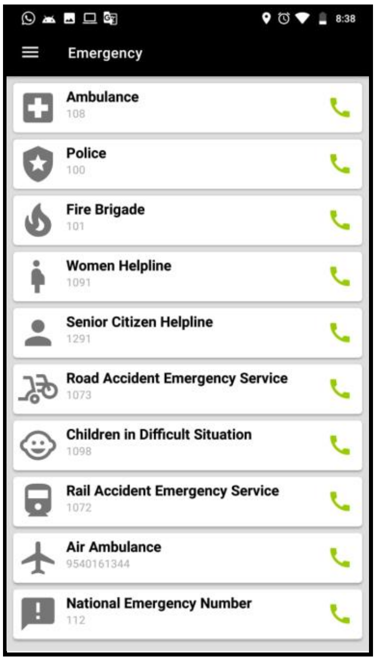
Dhruv

induac1974@gmail.com

8866760770

Fig 8.13 Admin Details Screen

8.14 Emergency

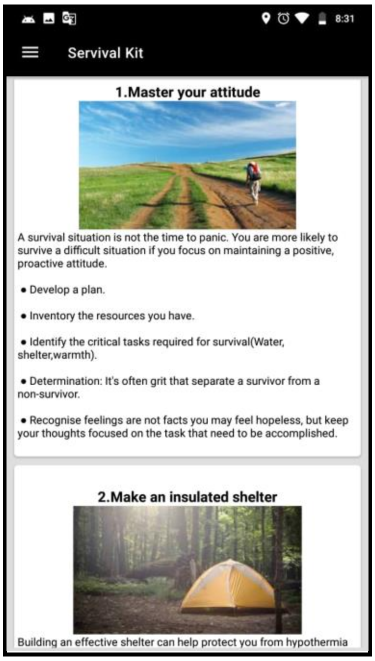


Emergency

Ambulance	108
Police	100
Fire Brigade	101
Women Helpline	1091
Senior Citizen Helpline	1291
Road Accident Emergency Service	1073
Children in Difficult Situation	1098
Rail Accident Emergency Service	1072
Air Ambulance	9540161344
National Emergency Number	112

Fig 8.14 Emergency Screen

8.15 Survival Kit



Servival Kit

1.Master your attitude

A survival situation is not the time to panic. You are more likely to survive a difficult situation if you focus on maintaining a positive, proactive attitude.

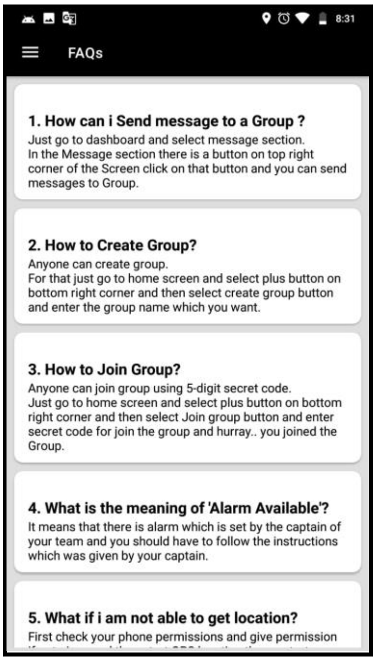
- Develop a plan.
- Inventory the resources you have.
- Identify the critical tasks required for survival(Water, shelter,warmth).
- Determination: It's often grit that separate a survivor from a non-survivor.
- Recognise feelings are not facts you may feel hopeless, but keep your thoughts focused on the task that need to be accomplished.

2.Make an insulated shelter

Building an effective shelter can help protect you from hypothermia

Fig 8.15 Servival kit Screen

8.16 FAQs



FAQs

1. How can i Send message to a Group ?
Just go to dashboard and select message section.
In the Message section there is a button on top right corner of the Screen click on that button and you can send messages to Group.

2. How to Create Group?
Anyone can create group.
For that just go to home screen and select plus button on bottom right corner and then select create group button and enter the group name which you want.

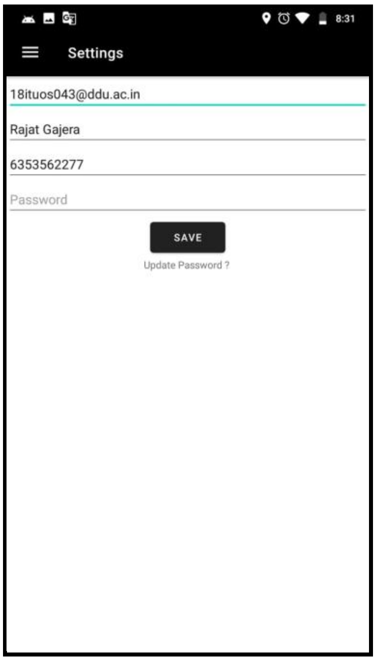
3. How to Join Group?
Anyone can join group using 5-digit secret code.
Just go to home screen and select plus button on bottom right corner and then select Join group button and enter secret code for join the group and hurray.. you joined the Group.

4. What is the meaning of 'Alarm Available'?
It means that there is alarm which is set by the captain of your team and you should have to follow the instructions which was given by your captain.

5. What if i am not able to get location?
First check your phone permissions and give permission

Fig 8.16 FAQs Screen

8.17 Settings



Settings

18ituos043@ddu.ac.in

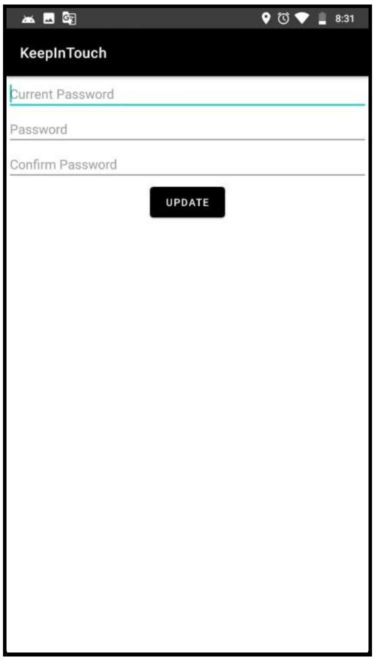
Rajat Gajera

6353562277

Password

Fig 8.17 Settings Screen

8.18 Update Password



KeepInTouch

Current Password

Password

Confirm Password

Fig 8.18 Update Password Screen

9. LIMITATIONS

1. If you clear the app data then you have to re-login to the application.

2. A Strong internet connection is required to load the exact location of any particular member.

3. A moderate connection is needed between the User and Backend server, for the proper data transfer and synchronous communication.

4. Furthermore, In the Gallery some of the Images are loaded directly from the internet, and if those images are removed in case, it can be a minor issue.

5. Every time, A User tries to login or signup, He/she must be remembering user id and password and also have to remember the role like admin or group member so He/she can set radius for others or see the radius which was set by admin.

10. FUTURE ENHANCEMENT

1. Animations can be more optimized for smaller screen devices. Moreover, the Responsiveness of the whole application can be improved further.

2. More secure authentication like OAuth 2.0 can be used instead of simple token-based authentication in the future versions

3. Third-party login and signup functionality can be implemented to enhance the user-friendliness.

4. Reset Password link should be expired after some fixed amount of time. So, in that area also the app can be improved.

5. Admin will be able set alarm to every members of group.

6. If you forgot password then there is only one way to reset the password via Email-id and for this valid email id is necessary and must be registered when you did registration but, in the future, we can add functionality of OTP system and which is linked by mobile number and email id both.

11. CONCLUSION

This project has been implemented from what we have learned in our college curriculum and many rich resources from the web. The scope of this project is subjective to the type of application that needs to be analysed, however, it was developed by keeping in mind the goal to keep the application as generic and minimal as possible.

After doing this project we conclude that we have got more knowledge about emerging technology Android. Also, in India this type of system is not exists so we thank Prof. (Dr.) Harshad B. Prajapati to allow us to work on this new and unique Idea.

12. REFERENCES

- https://developer.android.com/docs
- https://stackoverflow.com/
- https://firebase.google.com/docs
- https://www.javatpoint.com/android-tutorial
- https://www.geeksforgeeks.org/android-app-development-fundamentals-for-beginners/
- https://www.youtube.com/