



Submitted By:

NOOR UL HUDA 01-134192-071
AYESHA FAWAD 01-134192-014

SAFETRACE

Supervisor:

Dr. Professor Faisal Bashir

**Bachelor of Science in Computer Science
Department of Computer Science Bahria University, Islamabad**

Certificate

We accept the work contained in the report titled 'SafeTrace, written by Mrs. Ayesha Fawad AND Mrs. Noor ul Huda as a confirmation to the required standard for the partial fulfillment of the degree of Bachelor of Science in Computer Science.

Approved by . . . :

Supervisor: Prof. Dr. Faisal Bashir Hussain

Internal Examiner: Name of the Internal Examiner (Title)

External Examiner: Name of the External Examiner (Title)

Project Coordinator: Ms.Maryam Khalid Multani (Assistant Professor)

Head of the Department: Dr.Arif Ur Rehman (Sr. Associate Professor)

May 1st, 2023

Abstract

Tracking Systems to assist pedestrians have been in practice for a long time now. They help users to track and trace their live locations as well as others too. The locations can be in form of either texts or in maps. The issue that users face is that most of the major features require cost and subscription or not on one (1) application. The user interface is often too complex and require manuals to understand. To cater this issue, we have developed an android based Mobile Application "SafeTrace" to help users to Track and Trace themselves or their loved ones free of cost and experience all the necessary features on a single platform. Some additional features have also been added to make SafeTrace unique in the market like Panic Mode, Spot Mode, in-app chats etc. Live Locations can be checked, saved and shared in databases. Users profiles are created with their Phone Numbers and they can connect with people through the app. SafeTrace is designed in a simple user friendly and attractive interface where users can experience all the features.

Acknowledgement

"It is He Who created for you (the faculties of) hearing, sight, feeling and understanding; little thanks it is ye give!" -23:78

We thank Allah for providing us with courage to complete the project. We would like to thank our supervisor, Dr. Faisal Bashir for giving us the opportunity to do our fyp with them. It is indeed with a great sense of pleasure that we acknowledge the guidance, cooperation and support of Our supervisor and Co-supervisor which helped us in successful completion of this project. We are extremely grateful to everyone especially our parents and our friends for their kind cooperation and encouragement throughout.

AYESHA FAWAD AND NOOR UL HUDA
June 2023

*"Education is the most powerful weapon
which you can use to change the world."*

Nelson Mandela.

Contents

1	Introduction	10
1.1	Background Information	10
1.2	SafeTrace	11
1.3	Problem Description	11
1.4	Project Objective	12
1.5	Project Scope	12
2	Literature Review	13
3	Requirement Specification	15
3.1	Existing Requirement Specification	15
3.2	Proposed System	15
3.3	Requirement Specification	15
3.3.1	Software Requirements	15
3.3.2	Hardware Requirements	15
3.4	User Requirement Specification	16
3.4.1	Functional Requirements	16
3.5	Use Cases	17
3.5.1	Login Use Case	18
3.5.2	SignUp Use Case	19
3.5.3	Profile creation Use Case	20
3.5.4	Search Use Case	21
3.5.5	Save Use Case	22
4	Design	23
4.1	System Architecture	23
4.2	Design Methodology	23
4.3	High Level Design	24
4.4	Low-Level Design	27
4.4.1	Use Case Diagram	27
4.4.2	Activity Diagrams	28
4.5	Sequence Diagram	29
4.5.1	Login Sequence Diagram	29
4.5.2	Signup Sequence Diagram	29
4.5.3	Search Sequence Diagram	30
4.5.4	Save Sequence Diagram	30
4.6	Database Diagram	31
4.7	GUI	34
4.7.1	Splash Screen GUI	34
4.7.2	Login Screen GUI	34
4.7.3	Signup Screen GUI	35
4.7.4	Main Menu GUI	35
4.7.5	OTP screen GUI	36
4.7.6	Profile Creation Screen GUI	36
4.7.7	Chat Menu Screen GUI	37
4.7.8	Groups Screen GUI	37
4.7.9	Group Chat GUI	38
4.7.10	Private Chat GUI	38
4.7.11	Create Group GUI	39
4.7.12	Join Group GUI	39
4.7.13	Location History Screen GUI	40
4.7.14	Camera Map GUI	41
4.7.15	Route Finder GUI	41
4.7.16	Distance calculator GUI	42

5	System Implementation	43
5.1	System Architecture	43
5.2	Tools and Technology used	44
5.2.1	Java Language	44
5.2.2	Android Studio	44
5.2.3	Google Maps API	44
5.2.4	Andoid Mobile Phones	45
5.2.5	Firebase	45
6	System Testing	46
6.1	System Architecture	46
6.2	Graphical User Interface Testing	46
6.3	Usability Testing	46
6.3.1	Colour Scheme	46
6.3.2	Buttons	46
6.3.3	Feedbacks	47
6.3.4	Functionality	47
6.4	Test Cases	48
6.4.1	Login/Register Test Case	48
6.4.2	OTP Verification Test Case	49
6.4.3	Chat Module Test Case	50
6.4.4	Camera Map Test Case	51
6.4.5	Travel History Test Case	52
6.4.6	Route Finder Test Case	53
6.4.7	Distance Calculator Test Case	54
6.5	Compatibility Testing	55
6.5.1	Exception handling:	55
6.6	Unit Testing	56
6.7	Integration Testing	56
7	Conclusion	57
7.1	Learning Outcomes	57
7.1.1	Awareness	57
7.1.2	Team Work	57
7.1.3	Familiarity and Scope	57
7.2	Future Work	58

List of Figures

2.0.1 SafeTrace application compared with existing applications	14
3.5.1 Login Use case	18
3.5.2 SignUp Use case	19
3.5.3 Profile creation Use case	20
3.5.4 Search Use case	21
3.5.5 Save Use case	22
4.1.1 System Architecture	23
4.2.1 Agile Methodology incorporated for development of Safe Trace.	24
4.3.1 PCAM model of SafeTrace	25
4.3.2 Partitioning Model of different modules	25
4.3.3 Communication Model between different modules	26
4.3.4 Agglomeration Model of different modules	26
4.4.1 SafeTrace Use Case Diagram	27
4.4.2 SafeTrace Activity Diagram	28
4.5.1 Login Sequence Diagram	29
4.5.2 SignUp Sequence Diagram	29
4.5.3 Search Sequence Diagram	30
4.5.4 Save Sequence Diagram	30
4.6.1 Database Diagram	31
4.6.2 Firebase Database Diagram	31
4.6.3 Authentication of users	32
4.6.4 Realtime database of chats	32
4.6.5 Realtime database of users	32
4.6.6 Realtime database of location	33
4.6.7 Realtime database of groups	33
4.6.8 Realtime database of chats list	33
4.7.1 Splash Screen of Safetrace	34
4.7.2 Login Screen of Safetrace	34
4.7.3 Signup Screen of Safetrace	35
4.7.4 Main Menu of Safetrace	35
4.7.5 Otp screen of Safetrace	36
4.7.6 Profile Creation Screen of Safetrace	36
4.7.7 Chat Screen of Safetrace	37
4.7.8 Groups list of Safetrace	37
4.7.9 Group Chat of safetrace	38
4.7.10 Private Chat of Safetrace	38
4.7.11 Create Group	39
4.7.12 Join Group	39
4.7.13 Location History Screen of Safetrace	40
4.7.14 Location History list Screen of Safetrace	40
4.7.15 Camera Map module	41
4.7.16 Route Finder module	41
4.7.17 Distance Calculator module	42
5.2.1 Java language	44
5.2.2 IDE	44
5.2.3 API key in manifest	44
5.2.4 Firebase Development platform	45
6.5.1 SDK Version	55
6.5.2 Android Gradle Plugin Version	55
6.5.3 Compile SDK and defaultConfig	55
6.5.4 Exceptional Handling blocks	55
6.7.1 Integration Testing	56

List of Tables

3.5.1 Login Use Case Table	18
3.5.2 Sign-Up Use Case Table	19
3.5.3 Profile Creation Use Case Table	20
3.5.4 Search Use Case Table	21
3.5.5 Save Use Case Table	22
6.4.1 Login/Register Test Case	48
6.4.2 OTP Verification Test Case	49
6.4.3 Chat Module Test Case	50
6.4.4 Camera Map Test Case	51
6.4.5 Travel History Test Case	52
6.4.6 Route Finder Test Case	53
6.4.7 Distance Calculator Test Case	54

Acronyms and Abbreviations

OTP	One-Time Password
AVD	Android Virtual Device
API	Application programming interface
GPS	Global Positioning System
Ring	Group

Chapter 1

Introduction

1 Introduction

The ubiquity of wireless networking and the trend toward component miniaturization have led to the evolution of cell phones which led to more mobile application usage among people. Today's mobile phones are typically equipped with devices such as GPS sensors, Wi-Fi and 4G/5G services etc. Mobile Apps have become a need of one's life and with the power of Internet it permits users to access functionalities/ services of app. With such advancement people now a days want location based services to enhance personal security, to share/get real-time location information and that's the reason why location based services attract the attention of a great pool of people. Currently, location-based services are growing its popularity in the IT industry, and according to Markets and Markets report, this market is expected to reach 40 billion dollars by 2024. [6]

Location based apps enables user to tie-up(connect), track and trace the location of anyone who has given the permission, resulting in as a socializing app.[6]

To provide ease to the users and contribute our part in this rea-time tracking problem, Idea of developing SafeTrace has come that provides solutions and in future SafeTrace plans to contribute more. SafeTrace enables users to track the location of their friends and family in real-time, with accurate location updates displayed on a map. The chat feature allows users to communicate with their contacts directly within the app, without the need for separate messaging tools. The application also includes a range of features designed to enhance user privacy and security, including customizable location sharing settings and end-to-end encryption for chat messages.

The document outlines the technical and design aspects of the application, as well as the user experience and potential use cases. Overall, this location tracking mobile application with chat feature offers a convenient and secure way for users to stay connected with their loved ones and track their whereabouts in real-time. SafeTrace is not only to track locations of the users but it provides security, safety and helps the user socialize just through the Application.

1.1 Background Information

In this modern era, real-time tracking is a problem in which known platforms like google and sygic are already working. 'SafeTrace' is an app designed to face this problem. It is a tracking Location app with additional features which one cannot find under single app. This app is an all-in-one where user can experience almost every feature related to live location tracking for the ease and comfort of the user, making it user friendly. There are several tracking apps such as 'A-GPS Tracker'[1], 'Family Locator' [2], 'Glympse' [7] and many more that don't have functionalities which are in one platform and are singular. SafeTrace covers it all!

1.2 SafeTrace

One of the main features of SafeTrace is to 'Track' and 'Trace' a person's live location which can be saved, shared and can be viewed for later purposes. Saved locations can be searched with filters for user's convenience.

Immediate respond during an emergency is the need of people now a days and having such apps that allow people to get notified and being tracked and traced easily is the biggest advantage. SafeTrace has a special 'Panic Mode' that notifies other members within the ring that you are in trouble or in need of help so with the live location sharing feature the user can be traced easily. Apps like 'Geozilla', 'FamilyLocator' with millions of downloads have this feature but it's not free. SafeTrace provides this feature free of cost which makes it unique and preferable for the users.

1.3 Problem Description

People prefer applications that have multiple functionalities related to real-time tracking and socializing, and there is increased demand of such applications in the market. Common requirements are not only tracking the live location of the person with whom they are connected but also by providing multiple functionalities such as chatting, sharing images, maintaining history of locations travelled, option of Panic Mode that generates alert message to the remaining group members in need of help/emergency, pinpoint a meeting spot by sharing Location of that spot in the group i.e. Spot Mode.

Navigation is a real-time problem in which famous platforms like "Google" and "Sygic" are already working[9]. Another crucial feature is the safety of people while they travel using different private or shared commute online services. SafeTrace does OTP (One Time Password) verification to access the private chats of the user. For the ease and convenience of a user, it is important to make applications where a user can experience all-in-one features on a single platform relating to socializing, safe travel and tracking.

1.4 Project Objective

SafeTrace would serve several functionalities. Some of the key objectives of the app are:

- a People after being connecting (Tie-up) can track each other by sharing locations, share pictures and chat messages free of cost!
- b Pinpoint a specific destination for a meeting point of a group.
- c Maintaining history of previously saved locations so user can review, share or delete it later.
- d A panic button within the group to generate alert message to all the members if anyone in the group needs help.
- e If any group member gets disconnected (gets offline), he/she will be shown as offline and status updates as soon the person gets connected back.
- f Route from our current location to any of the group members can also be identified through the map.
- g Straight Route Distance can be calculated on the map.
- h User can get live location and can save it along with the live picture of that place.

1.5 Project Scope

- a Live Location Tracking, viewing and sharing.
- b Maintains history of locations in the database.
- c Chat in either ring or private where you can share images and location.
- d Generate Alert Messages in the group.
- e Share a meeting Point Location in the group.

Chapter 2

Literature Review

2 Literature Review

There are several applications that are mainly developed for

- Tracking i.e tracking mobile phones, cars, routes,
- Socializing (for connecting or chatting purpose),
- Maintain history of travelled routes,
- Safety apps generating notification in case of emergency.

Some of the existing products/applications are discussed in the remaining of this section based on their features, downloads, and rating.

a A-GPS tracker[1]

- (a) It is a tracking app with 1M+ downloads and is rated 3+.
- (b) It is meant specifically for hikers providing them with functionality to record, store and load track from GPX.

b Family Locator[2]

- (a) It is a tracking app with 100k+ downloads and is rated 3+.
- (b) It is meant specifically for family members to track and trace each other.
- (c) It provides with a chat option in circle.
- (d) It also provides with history maintaining feature but with paid packages of (1,6,12 months) only of 30 days.
- (e) Speed limit and low battery notifications and step challenges (premium) are its main features other than location sharing and tracking family members.

c Geozilla[5]

- (a) It is a tracking app with 10M+ downloads and is rated 3+.
- (b) To unlock all access there are paid packages (1 month, 1 week).
- (c) It provides with a chat option in circle.
- (d) Emergency alert can only be sent to a specific contact which is selected.
- (e) Trace people by adding a place so whenever members come and go there, you get to know.
- (f) Emergency alert message.
- (g) History record can be searched only of the current day and have to get the package to unlock feature to search history of routes.

d **Family360** [3]

- (a) Family360 is a tracking application with 1M+ downloads and is rated 3+.
- (b) It provides with a chat option in circle.
- (c) It provides 10-15 days subscription.
- (d) 3 Emergency contacts are selected and SMS is sent to them in case of emergency.
- (e) Live location of a person can be viewed.

Apps	Family Locator	A Tracker	GPS	Family360	Geonection	Safe Trace
Features						
Panic Button	✗	✗		✓ (Paid)	✗	✓ (Free)
Tie-up option	✓	✗		✓	✓	✓ (Free)
Chat option	✗	✗		✓	✗	✓ (Free)
History	✗	GPX track		✗	✓ (Paid)	✓ (Free)
Search routes	✗	✗		✗	✓ (Paid)	✓ (Free)
Spot mode	✗	✗		✓ (Paid)	✓ (Paid)	✓ (Free)
Location sharing	✓	✓			✓	✓ (Free)

Figure 2.0.1: SafeTrace application compared with existing applications
 In Table the proposed SafeTrace android application is compared with few existing applications based on socializing and tracking features which highlights the strengths of proposed android application.

Chapter 3

Requirement Specification

3 Requirement Specification

3.1 Existing Requirement Specification

a **Design**

Figma and Wireframe are used for designing SafeTrace's interface.

b **Verification**

Firebase authentication is done through Firebase Console by enabling Phone Verification for OTP verification, Firebase Console by enabling Email for Email login, Google APIs and dependencies in the android Studio.

c **Maintenance**

App's Gradle version is updated and synced to keep SafeTrace up-to-Date.

3.2 Proposed System

Live Location Tracking is a global problem in which many famous platforms are already working like google and sygic etc. Many applications have been developed and available on playstore but none of the application has all the desired features of tracking under one roof. Some of the features are either missing or have cost for subscriptions. This proposed application covers all the necessary features available on one platform and all are free of cost. As this application is a safety application of a person as well as traveler's need, we proposed this application by keeping all the other apps on playstore in sight, designed SafeTrace for user's ease, comfort and use.

3.3 Requirement Specification

3.3.1 Software Requirements

The software requirements for this project are:

- a Android Studio
- b Firebase Console authentication + Google account + Json file
- c Google Maps Platform
- d GMS services
- e Figma

3.3.2 Hardware Requirements

- a Fast Internet Access
- b Computer/ Laptop
- c Mobile phone to run application

3.4 User Requirement Specification

a Limitations:

- (a) If the user gets disconnected i.e. no internet, the app will show status as offline for that person.
- (b) User will only be able to use the tracking services if the location services option is turned on.
- (c) User will only be able to use the tracking services if the app has permissions for locations.

b Assumptions:

- (a) User has created a profile.
- (b) User is connected with the Internet.
- (c) User has turned on location services.

c Resource Requirement:

- (a) Plugins:
 - 1. Maps Platform
- (b) Libraries
- (c) Permissions
 - For Live Location and map**
 - i. External storage.
 - ii. Access fine location.
 - iii. Access background location.
 - iv. Internet.
 - v. Access coarse location.
 - For camera**
 - i. External storage.
 - ii. Camera.

3.4.1 Functional Requirements

- Android Studio
- Figma
- Google Maps Platform(Api Key)[8]
- SQLite/ Firebase[4]

3.5 Use Cases

Here are some graphical representation of user's possible interaction with the system that includes:

- Login
- Signup
- Profile creation or otp verification
- Search
- Save

3.5.1 Login Use Case

This use case allows the user to login into the system to access the relevant functions according to the user's role. To log in to the system, Users has to enter email and password registered through. Upon successful login, the user will be redirected to the next page.

Table 3.5.1: Login Use Case Table

Use Case Name	LOG IN
Description	This use case allows users to login into the system to access the relevant functions according to the user's role. To log in to the system, Users has to enter the email and password registered through.
Actor	User
Pre-Condition	User's name,email and password already saved in database.
Post-Condition	The system displays the relevant homepage.

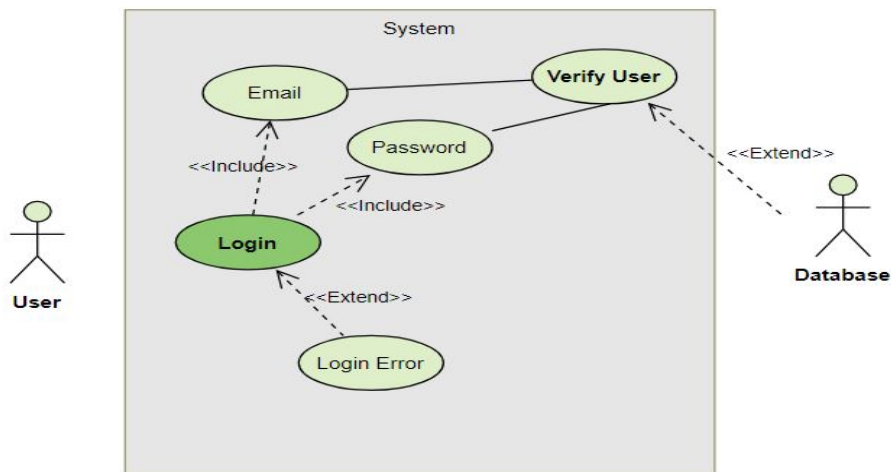


Figure 3.5.1: Login Use case

3.5.2 SignUp Use Case

This use case allows users to SIGN UP where email and password is saved into the Database and by logging in user can get benefits from Safe Trace. As shown in Table and Figure below

Table 3.5.2: Sign-Up Use Case Table

Use Case Name	SIGN-UP
Description	This use case allows users to SIGN UP where email and password is saved into the Database and by logging in user can get benefits from Safe Trace.
Actor	User
Pre-Condition	User must have downloaded Safe Trace.
Post-Condition	User can now use the app..

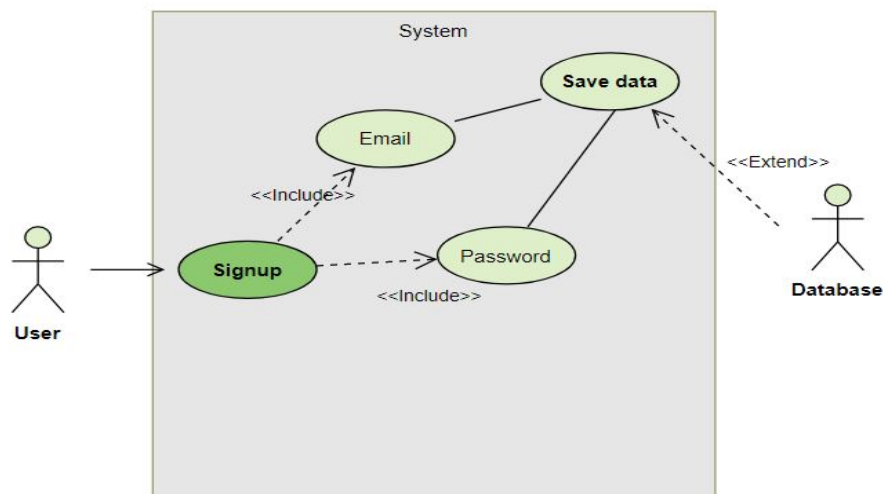


Figure 3.5.2: SignUp Use case

3.5.3 Profile creation Use Case

This use case allows users to login for chat module where user name, profile picture and phone number is required and saved into the Database and by verifying phone number by getting opt user creates the profile and logs in to chat module and can get benefits from Safe Trace. As shown in Table and Figure below

Table 3.5.3: Profile Creation Use Case Table

Use Case Name	PROFILE CREATION
Description	This use case allows users to login for chat module where user name, profile picture and phone number is required and saved into the Database and by verifying phone number by getting opt user creates the profile and logs in to chat module and can get benefits from Safe Trace.
Actor	User
Pre-Condition	User must have logged in Safe Trace.
Post-Condition	User can now use the chat module and other functions.

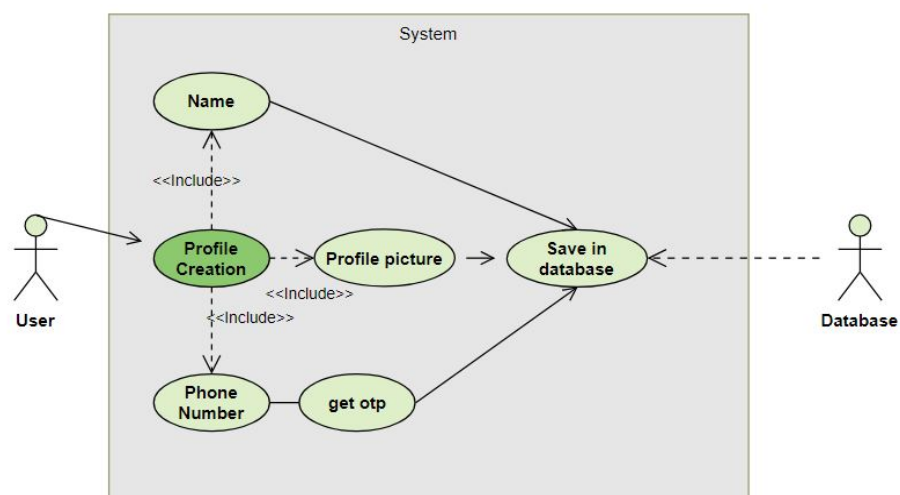


Figure 3.5.3: Profile creation Use case

3.5.4 Search Use Case

The user will enter the required location and using search filter easily data from database can be searched.

Table 3.5.4: Search Use Case Table

Use Case Name	SEARCH
Description	The user will enter the required location and using search filter easily data from database can be searched.
Actor	User
Pre-Condition	User must be logged in first and must have saved something in database searching for.
Post-Condition	User will be displayed with relevant data searched through filter.

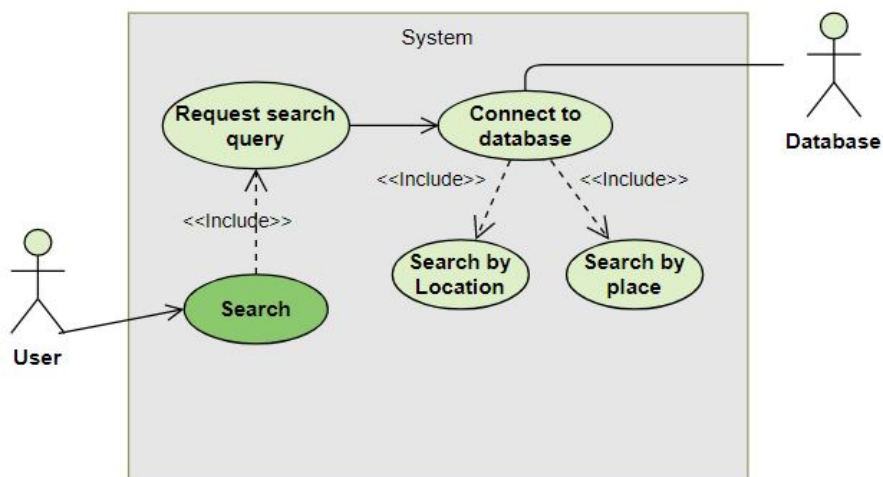


Figure 3.5.4: Search Use case

3.5.5 Save Use Case

This use case will display data that is searched according to the user's search query.

Table 3.5.5: Save Use Case Table

Use Case Name	SAVE
Description	This use case will display data that is searched according to the user's search query.
Actor	User
Pre-Condition	User must be logged in first.
Post-Condition	User will be displayed with the data searched from database.

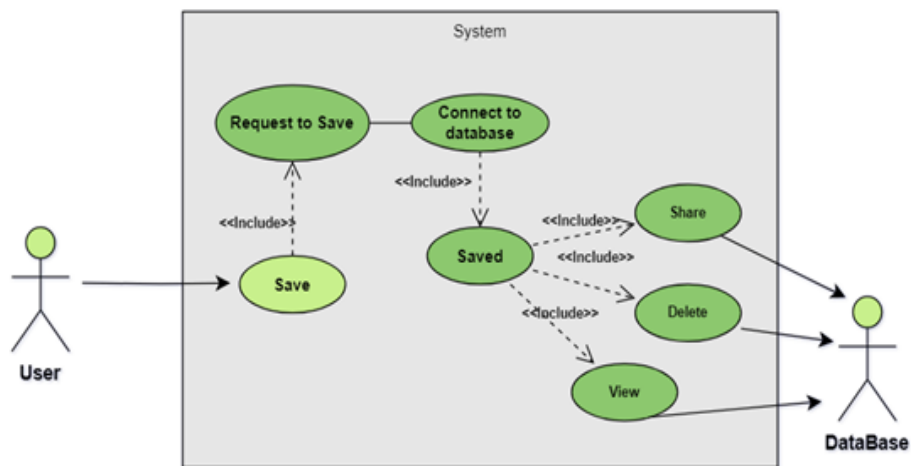


Figure 3.5.5: Save Use case

Chapter 4

Design

4 Design

Systems design is the process of defining the architecture, components, modules, interfaces, and data for a system to satisfy specified requirements. The following sections constitute this chapter:

4.1 System Architecture

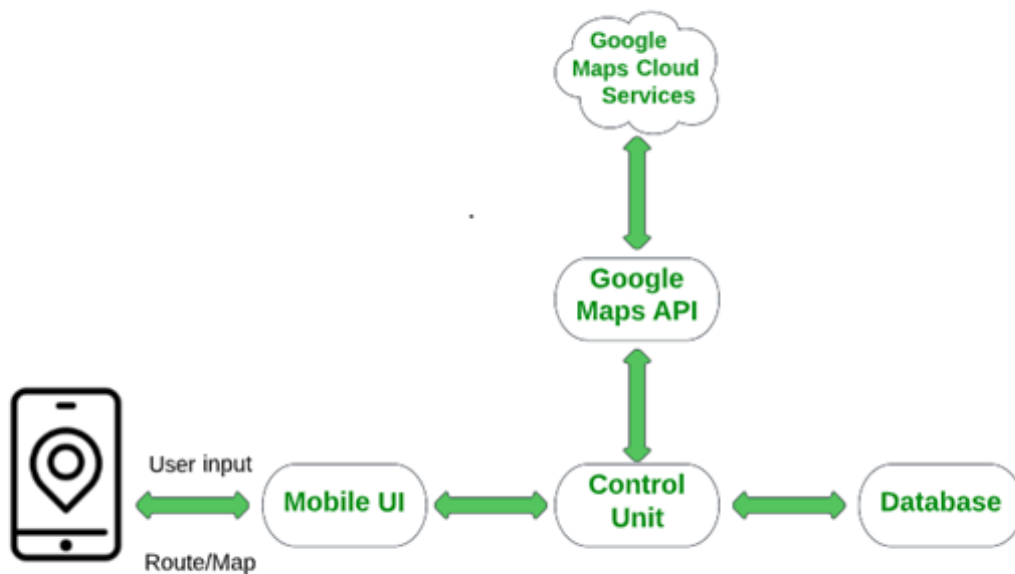


Figure 4.1.1: System Architecture

User interacts with Phone's GUI to fetch/save/search the data(Location) from database and by using Google Maps API and gms services.

4.2 Design Methodology

The proposed project will be implemented using an agile methodology. When it comes to accomplishing this goal, the agile technique will prove very beneficial. According to the agile methodology, the planned project is divided into various phases. Each phase demands continuous communication and feedback from team members, as well as continuous improvement. The agile model incorporates the following steps:

- a Project Proposal
- b Planning
- c Design
- d Development / Implementation
- e Database Connectivity
- f Testing

The following figure shows the design methodology followed for the Safe Trace.

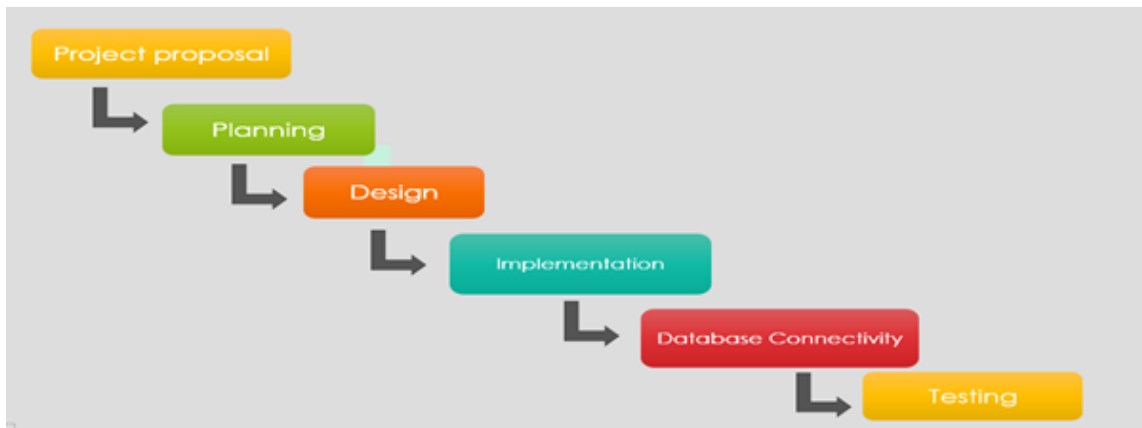


Figure 4.2.1: Agile Methodology incorporated for development of Safe Trace.

4.3 High Level Design

PCAM methodology structures the design process as four distinct stages: partitioning, communication, agglomeration, and mapping. It has four stages:

a **Partitioning**

Partitioning is dividing the computation and the data.

- Decompose the problem
- Identify the concurrent tasks
- Often the most difficult step

b **Communication**

Communication considers the plan for inter-process communication necessary for the parallel program.

c **Agglomeration**

Agglomeration is grouping tasks into larger tasks in order to improve performance or logical abstraction.

d **Mapping**

Mapping is assigning tasks to processors. The goal is to balance computation and communication loads in order to maximize processor utilization and minimize inter-processor communications.

Applying Pcam model to our final year project

- **Problem**

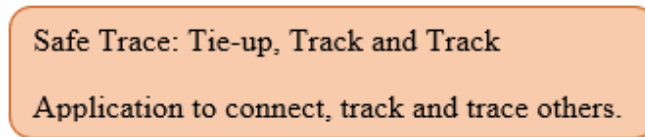


Figure 4.3.1: PCAM model of SafeTrace

- **Partitioning**

We will apply functional partitioning in our FYP i.e., to first decompose the computation to be performed and then deal with the data. Here we have decomposed problem into 4 computations:

- Travel History
- Live Location Tracking
- Panic Button
- Spot Mode

Each of them have a different operation to perform.

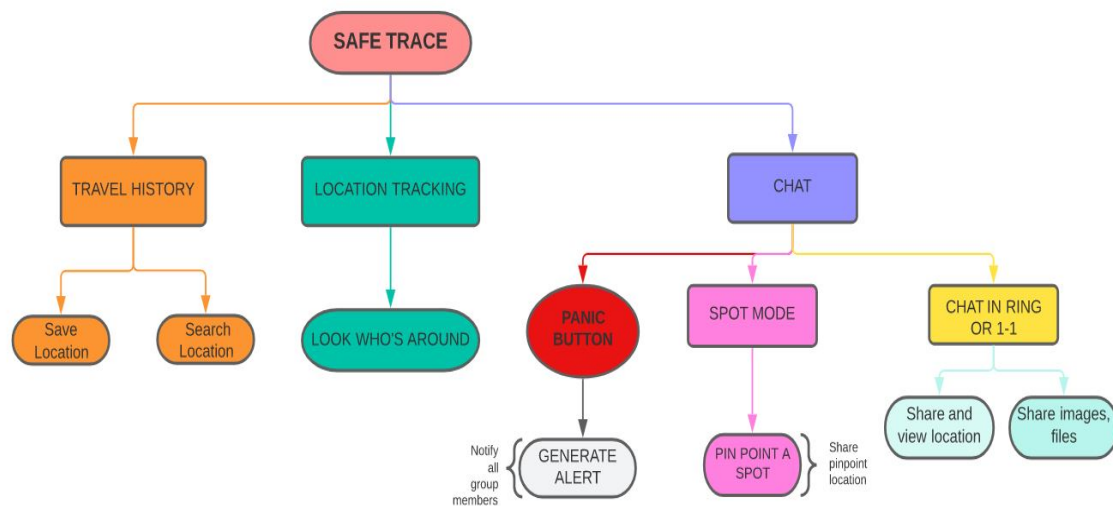


Figure 4.3.2: Partitioning Model of different modules

- **Communication**

Although the partitioning is done based on computations, the data is still not independent and needs some sort of communication for example, the data of live location tracking would be saved in database and will be shown in travel history. To simply illustrate this idea, a diagram is shown below:

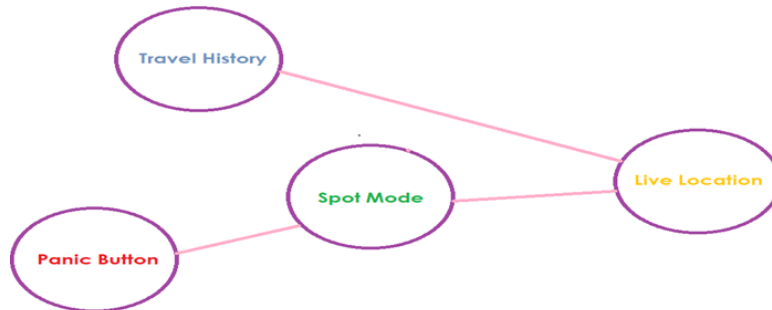


Figure 4.3.3: Communication Model between different modules

- **Agglomeration**

In this part, we will combine smaller tasks into larger tasks for efficiency. We combined front end and back end of our FYP application.

All of the components are linked to each other as live location would be used to spot a person, panic button would be used by a person at any specific location and again using spot mode a person can be located. Similarly, travel history of persons using the application would depend on their locations.

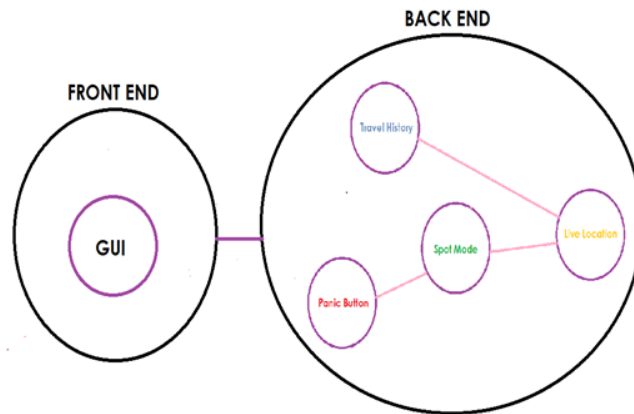


Figure 4.3.4: Agglomeration Model of different modules

- **Mapping**

Each component is assigned a different processor which helps to maximize the utilization of processor and ultimately reduce communication cost.

4.4.2 Activity Diagrams

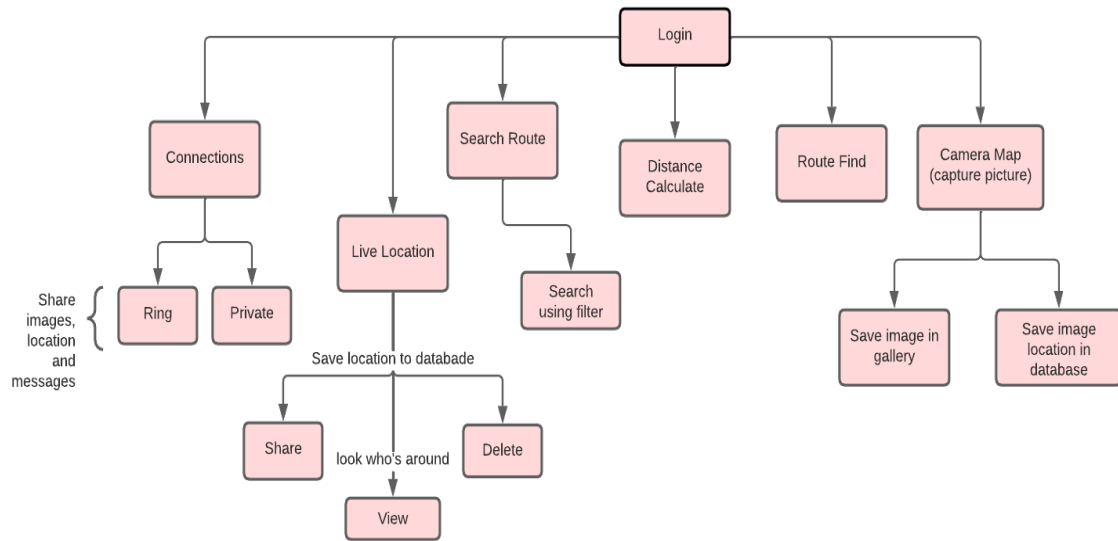


Figure 4.4.2: SafeTrace Activity Diagram

4.5 Sequence Diagram

4.5.1 Login Sequence Diagram

Figure shows how a user will log in to the system while following the sequence as depicted in the diagram.

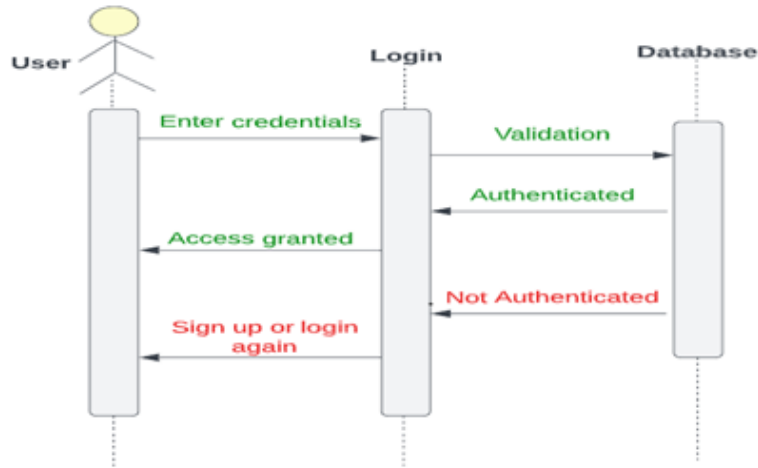


Figure 4.5.1: Login Sequence Diagram

4.5.2 Signup Sequence Diagram

Figure shows how a user will signup in to the system while following the sequence as depicted in the diagram.

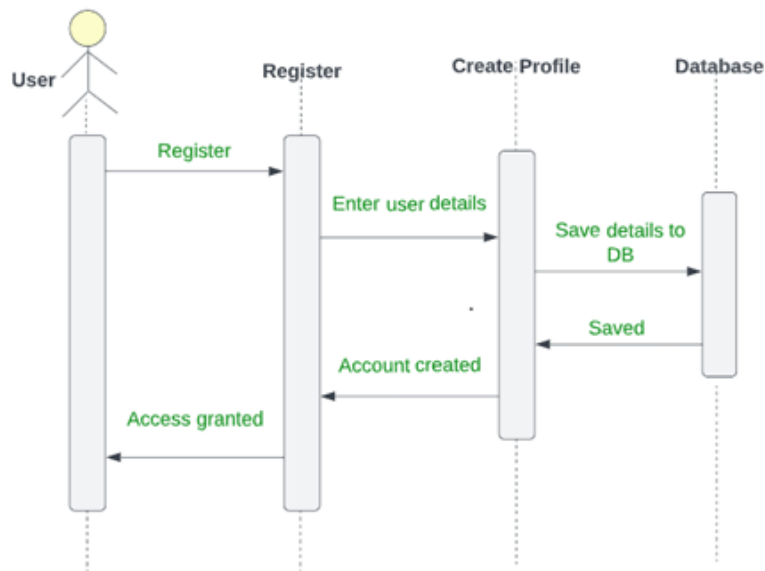


Figure 4.5.2: SignUp Sequence Diagram

4.5.3 Search Sequence Diagram

Figure shows how a user will search data from the database while following the sequence as depicted in the diagram.

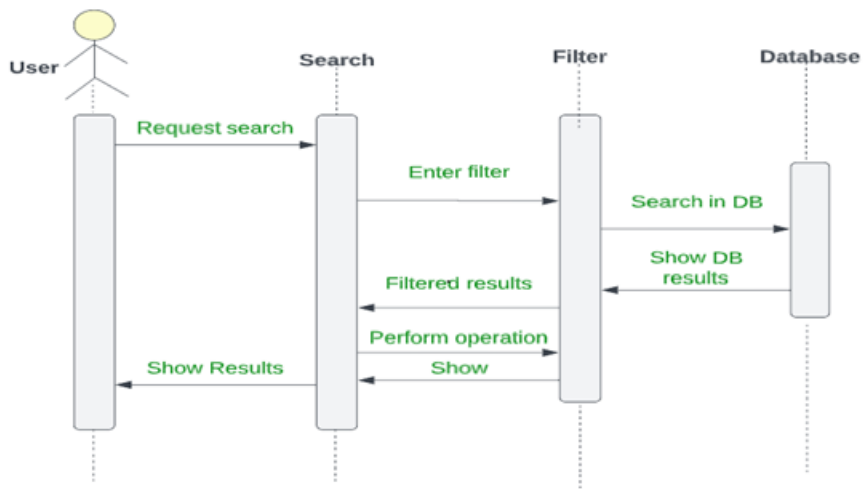


Figure 4.5.3: Search Sequence Diagram

4.5.4 Save Sequence Diagram

Figure shows how a user will save data to the database while following the sequence as depicted in the diagram.

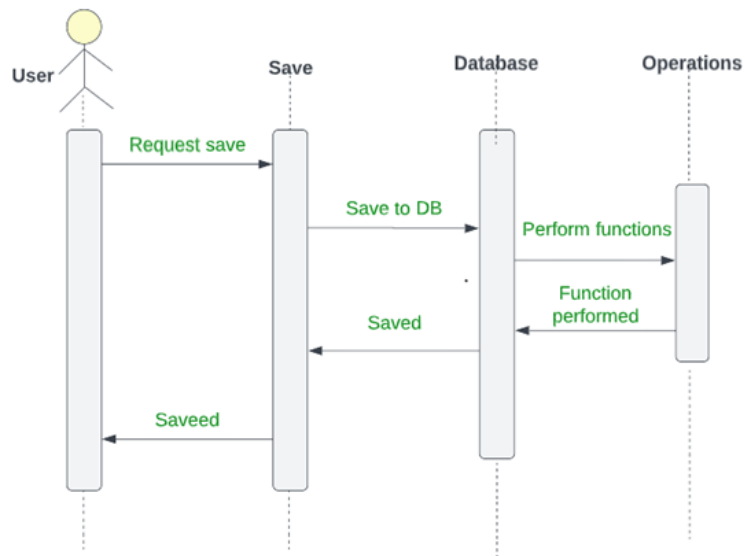


Figure 4.5.4: Save Sequence Diagram

4.6 Database Diagram

Firebase is a Realtime database that lets the user experience cloud-based data storage.

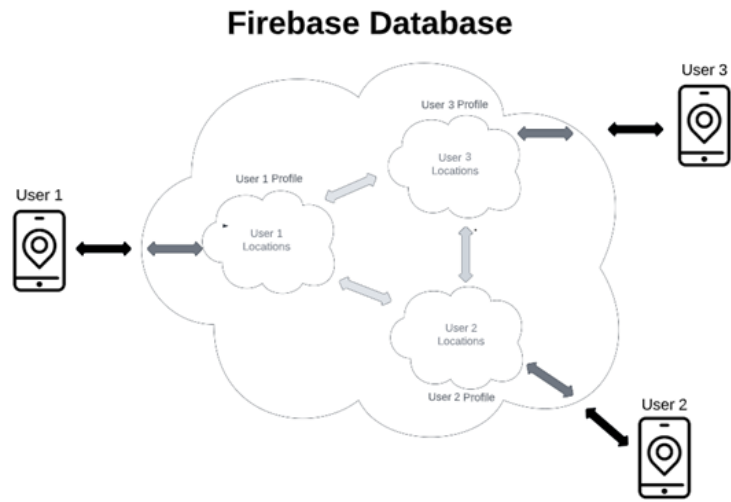


Figure 4.6.1: Database Diagram

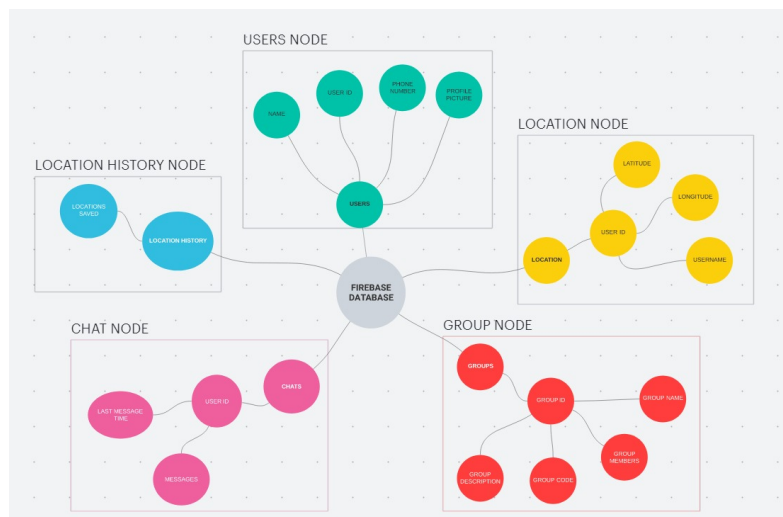


Figure 4.6.2: Firebase Database Diagram

Identifier	Providers	Created ↓	Signed in	User UID
[Redacted]	[Phone]	18 Apr 2023	18 Apr 2023	1ECx1ZL0zkSr0Sq81MDR014WoQ23
[Redacted]	[Phone]	18 Apr 2023	27 Apr 2023	aQeoU8FVA5UxUoxPpyiqUvAKIfQ2
abc@yahoo.com	[Email]	15 Apr 2023	16 Apr 2023	kgzsdv6DZ2ceEF05u7t0GFkGc13
mna@yahoo.com	[Email]	12 Apr 2023	12 Apr 2023	uBcASk4dsgY0HwRBMVvNpB10L...
haleemullah777@gmail.com	[Email]	12 Apr 2023	18 Apr 2023	oISkyKz1VQh08ocd7iuzC4UCRo2
tester@gmail.com	[Email]	9 Apr 2023	18 Apr 2023	zANP6SFobJYRZUghHFBHAFG_JF...
[Redacted]	[Phone]	7 Apr 2023	19 Apr 2023	lvm78MpDr1cgKxu0t0765tvVa3X2
noori@yahoo.com	[Email]	7 Apr 2023	7 Apr 2023	NmmeGr6dBJMPkj6M6uZ02xv4...
ayasha@yahoo.com	[Email]	5 Apr 2023	17 Apr 2023	y8XW8hm377ZWmhzVR8he5hNG...
[Redacted]	[Phone]	27 Mar 2023	18 Apr 2023	JqntKr12nRZ4KYCL4eeM4r4hn3v2
[Redacted]	[Phone]	22 Feb 2023	18 Apr 2023	b7cR6kxJYJhxGhtiHsF7dy8eKZj1
[Redacted]	[Phone]	21 Feb 2023	21 Feb 2023	o12BMKonnaApuZRF0h9ANsw...
[Redacted]	[Phone]	7 Oct 2022	7 Oct 2022	Bh5mCA8EnvNku0ggskS2GrZ78F...
[Redacted]	[Phone]	5 Oct 2022	5 Oct 2022	Qiy33R8mwXaxMmMT63wTcRfp...

Figure 4.6.3: Authentication of users

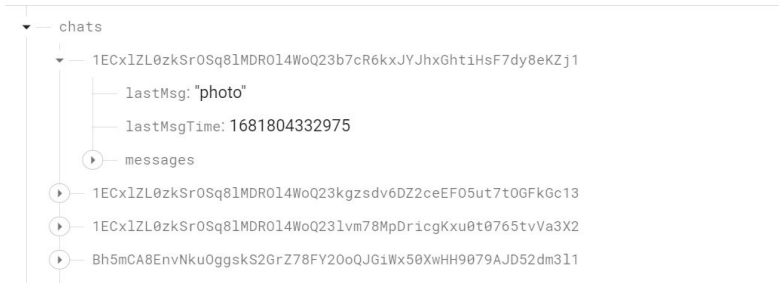


Figure 4.6.4: Realtime database of chats

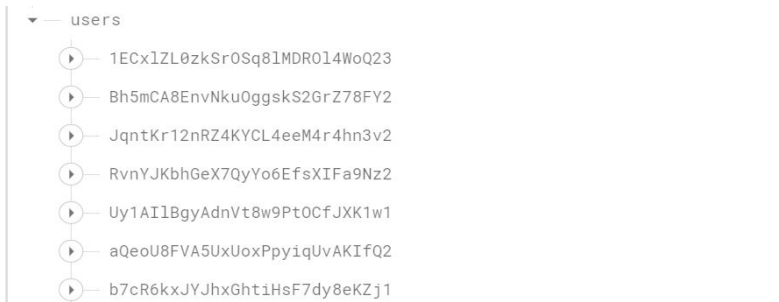


Figure 4.6.5: Realtime database of users

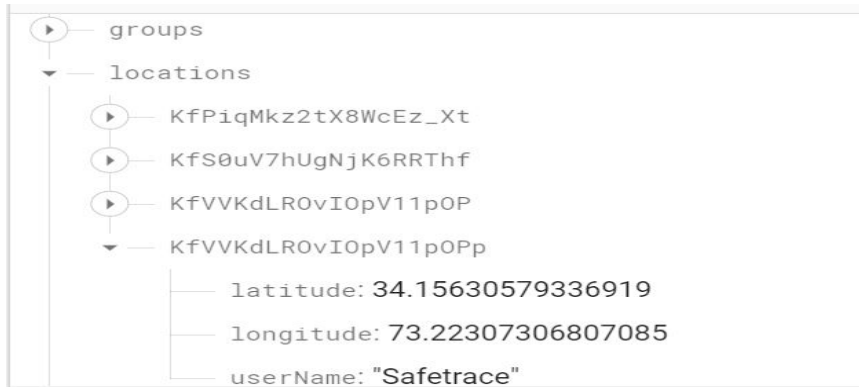


Figure 4.6.6: Realtime database of location

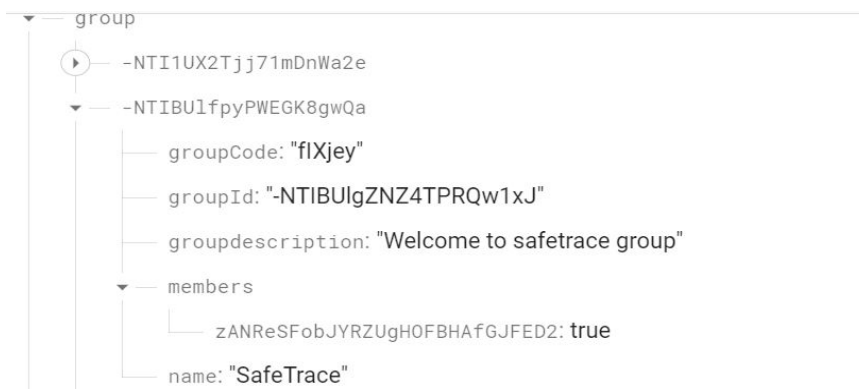


Figure 4.6.7: Realtime database of groups

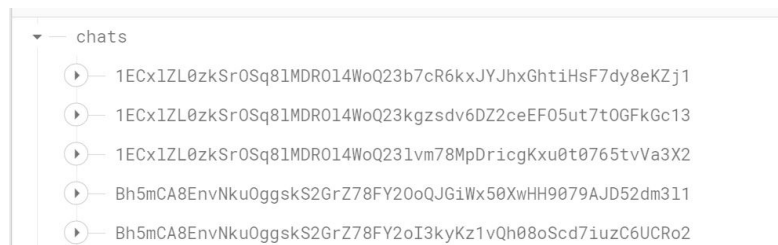


Figure 4.6.8: Realtime database of chats list

4.7 GUI

4.7.1 Splash Screen GUI



Figure 4.7.1: Splash Screen of Safetrace

4.7.2 Login Screen GUI

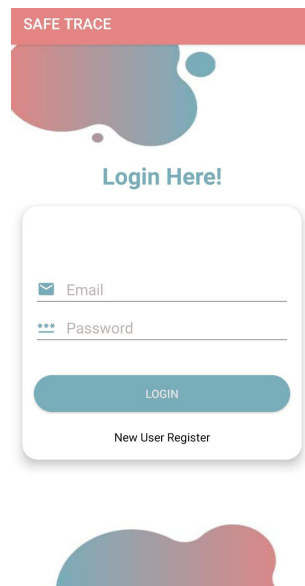


Figure 4.7.2: Login Screen of Safetrace

4.7.3 Signup Screen GUI

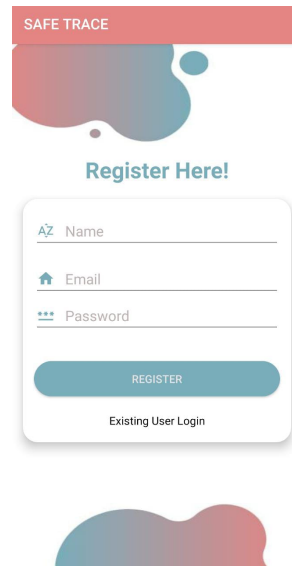


Figure 4.7.3: Signup Screen of Safetrace

4.7.4 Main Menu GUI

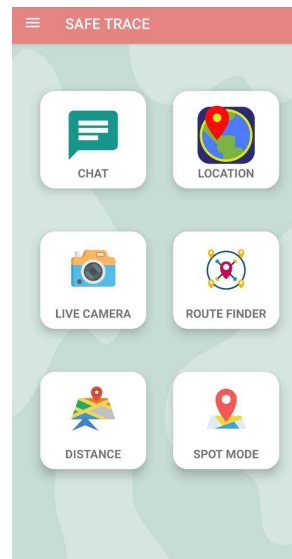


Figure 4.7.4: Main Menu of Safetrace

4.7.5 OTP screen GUI

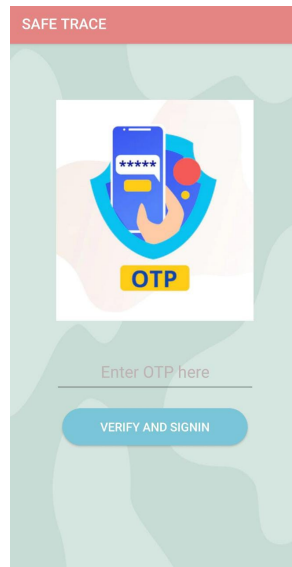


Figure 4.7.5: Otp screen of Safetrace

4.7.6 Profile Creation Screen GUI

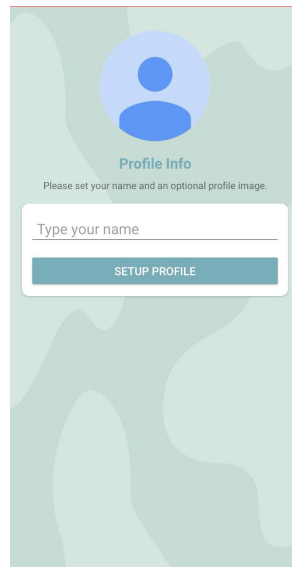


Figure 4.7.6: Profile Creation Screen of Safetrace

4.7.7 Chat Menu Screen GUI

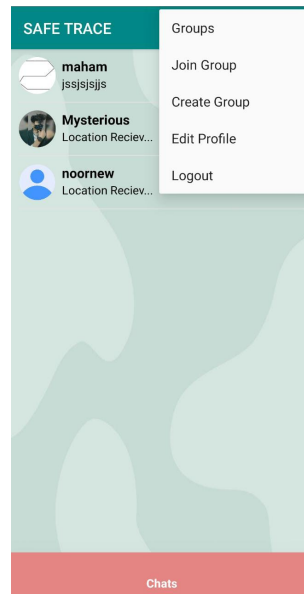


Figure 4.7.7: Chat Screen of Safetrace

4.7.8 Groups Screen GUI

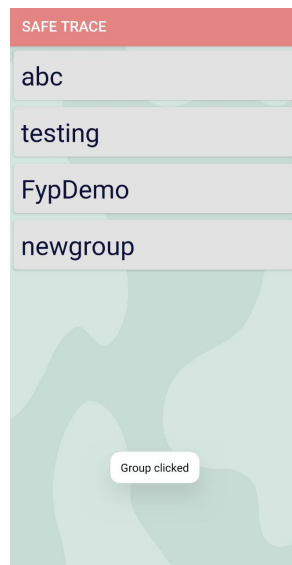


Figure 4.7.8: Groups list of Safetrace

4.7.9 Group Chat GUI

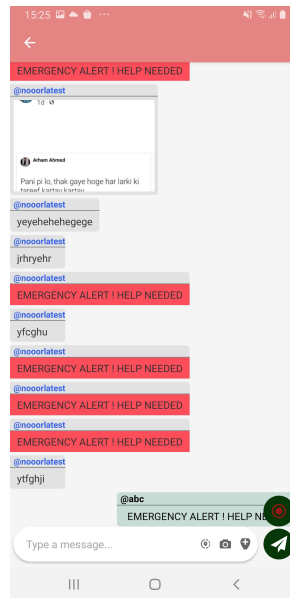


Figure 4.7.9: Group Chat of safetrace

4.7.10 Private Chat GUI

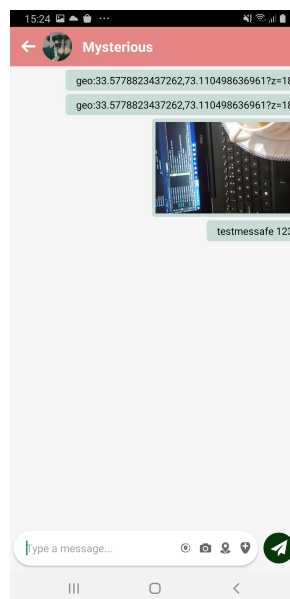


Figure 4.7.10: Private Chat of Safetrace

4.7.11 Create Group GUI

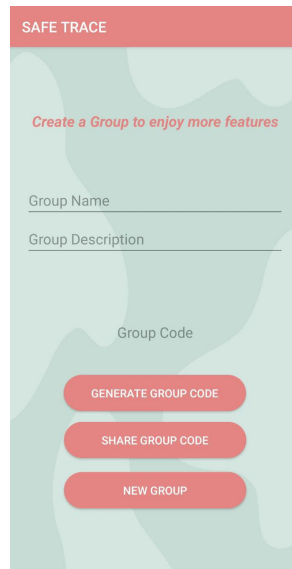


Figure 4.7.11: Create Group

4.7.12 Join Group GUI

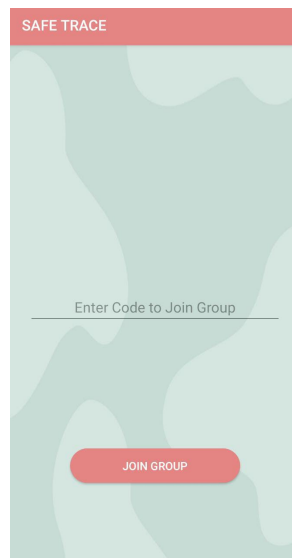


Figure 4.7.12: Join Group

4.7.13 Location History Screen GUI



Figure 4.7.13: Location History Screen of Safetrace

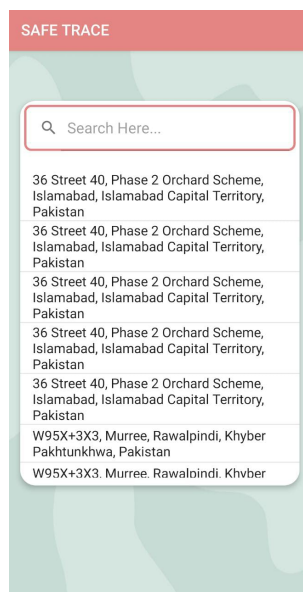


Figure 4.7.14: Location History list Screen of Safetrace

4.7.14 Camera Map GUI



Figure 4.7.15: Camera Map module

4.7.15 Route Finder GUI



Figure 4.7.16: Route Finder module

4.7.16 Distance calculator GUI

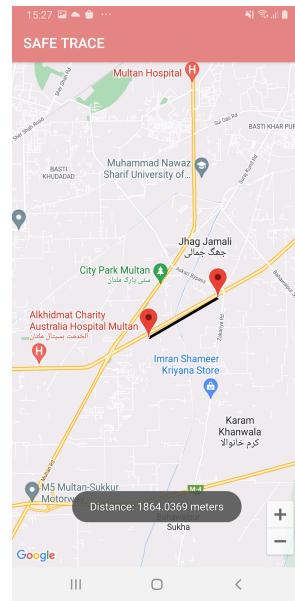


Figure 4.7.17: Distance Calculator module

Chapter 5

System Implementation

5 System Implementation

5.1 System Architecture

Several key components that needs to be considered in designing system architecture is:

a User Interface

The user interface includes the part of the application that the user interacts with. The main purpose of SafeTrace is to provide users a single platform with all-in-one features that are designed to be user-friendly, intuitive, and easy to navigate. SafeTrace includes features like viewing Map, viewing Locations, performing operations on them etc.

b Location Services

The location services component of the application is responsible for obtaining the user's location. This is done by using the device's GPS and by using other location-based technologies such as Wi-Fi. Permissions are asked first to access location and wifi.

c Data Storage

The data storage component of the application is responsible for storing location data as well as user's profile data. SafeTrace uses Firebase to handle and save the data in a secure fashion.

d Server-Side components

The server-side components of the application are responsible for handling data synchronization between devices, managing user accounts, and providing location-based services such as geofencing.

e Security

Security is an important consideration for any application that tracks location data. The application should implement measures to protect user data and secure communication between application and the server-side components. SafeTrace has 2 security features that lets the user access their account:

- Firstly, the user has to log-in/register himself in order to use the application and check its saved data.
- Secondly, to access chat module, SafeTrace do the OTP verification of the user (even though the user is logged in) to access the chats.

f Messaging Services

The mobile application uses messaging services to send and receive messages between users. SafeTrace uses custom-built messaging service that lets the user chat either 1-to-1 or in ring. The user can create groups, invite people, accept requests etc.

5.2 Tools and Technology used

5.2.1 Java Language

Java is used in Android development because of its robustness, reliability, and versatility. The language is well-suited to creating large, complex applications, and it has a large and active community of developers who have created many libraries and frameworks that can be used in Android development. All the coding part of SafeTrace is done in Java language.



Figure 5.2.1: Java language

5.2.2 Android Studio

Android Studio is an Integrated Development Environment (IDE) for developing Android applications. It is developed by Google and is the official IDE for Android app development. Android Studio provides a comprehensive development environment that allows developers to create, test, and deploy Android applications. Android Studio provides many features that make it easy to develop Android applications in Java, such as code completion, debugging tools, and built-in templates for common app components. Additionally, Android Studio comes with a built-in Android Virtual Device (AVD) manager, which allows developers to test their applications on virtual Android devices.



Figure 5.2.2: IDE

5.2.3 Google Maps API

The Google Maps Platform API is a collection of web services and tools provided by Google that developers can use to integrate maps, routes, and location-based features into their applications. The platform includes a wide range of APIs, each of which is designed to solve a specific location-related problem. API key was generated for the Google maps implementation. The syntax of adding API key in Manifest is this:

```
<meta-data
  android:name="com.google.android.geo.API_KEY"
  android:value="AIzaSyAHXx80lbaBIEdhf93hXxVL_PkK2fgeQ4o" />
```

Figure 5.2.3: API key in manifest

5.2.4 Andoid Mobile Phones

Android Mobile phones are the main technology that is used to run SafeTrace. As SafeTrace is an android application, several android phones are used to run and test its working. Among them were:

- Samsung Galaxy S8+
- Huawei Honour 6s
- Android HTC
- Redmi

5.2.5 Firebase

Firebase is a mobile and web application development platform, acquired by Google in 2014. It provides developers with a suite of tools and services to help them build high-quality apps and grow their user base. Firebase offers features such as real-time database, authentication, cloud messaging, hosting, storage, and analytics. With Firebase, developers can quickly build apps without worrying about managing servers or infrastructure.



Figure 5.2.4: Firebase Development platform

Chapter 6

System Testing and Evaluation

6 System Testing

6.1 System Architecture

System testing is a type of software testing that evaluates the complete system or application. The goal of system testing is to ensure that the system or application meets its requirements and performs as expected in its intended environment.

6.2 Graphical User Interface Testing

Graphical User Interface (GUI) testing is a type of software testing that focuses on evaluating the user interface of an application or system. The goal of GUI testing is to ensure that the application or system's user interface is user-friendly, intuitive, and responsive. SafeTrace is designed to beat the complexity and provide the user with a simple and attractive interface. SafeTrace was tested by several new members, and they easily used the application without any guide or manual. The interface appeared the same in different android Mobile Phones. All the interactions are made clear though different and visible feedbacks like Toast, Dialogue Boxes, Buttons etc.

6.3 Usability Testing

This testing is the interaction between the user and the Application (SafeTrace).

6.3.1 Colour Scheme

Pastel colours are often considered aesthetically pleasing because they are soft, gentle, and calming to the eye. These colours are created by adding white to pure colours, which reduces their saturation and intensity. As a result, pastel colours appear muted and subdued, creating a soothing and calming effect. pastel colours are aesthetically pleasing because they are soft, calming, nostalgic, versatile, and can evoke a range of emotions and moods. SafeTrace used the 60-30-10 rule Colours Chosen are:

- FF79ADBA
- E48583
- E2E2E2

6.3.2 Buttons

Big Cylinder buttons are used to improve usability and increase response time. As Fitts Law states 'That the amount of time required for a person to move a pointer to a target area is a function of the distance to the target divided by the size of the target. Thus, the longer the distance and smaller the target's size, the longer it takes for the user to perform an action' so keeping this in mind we made clear big buttons and input fields to improve user usability.

6.3.3 Feedbacks

A user is satisfied when the interaction between user and system is good and responsive so to make this happen, Toast are generated at several places to inform user of his progress and action.

6.3.4 Functionality

Functional testing is an open-ended process to check if the system is doing what it is supposed to do in an orderly and regulatory manner. Tests are performed to maintain balance and check completeness of achieving users' and system requirements.

Overall Interface

Whitebox testing is performed on the system by checking the complexity and efficiency of the code with respect to time and space the system is taking to perform any task. It also tests the coding capabilities of the developers of project.

6.4 Test Cases

6.4.1 Login/Register Test Case

User has to enter login credentials in order to access the features of SafeTrace.

Table 6.4.1: Login/Register Test Case

Test Case ID	ST-L1
Description	The user has to login or signup to use the application.
Pre-Condition	User should enter a valid Email address and password.
Steps	<ol style="list-style-type: none">1. SafeTrace must be installed.2. User open SafeTrace3. User is directed to login/Signup in order to use the Application.4. User Logs in or select register if he is new.
Expected Results	User Successfully logs in or register in the SafeTrace.
Status	Pass

6.4.2 OTP Verification Test Case

User has to enter the phone number where an opt will be generated. User is then redirected to profile creation activity.

Table 6.4.2: OTP Verification Test Case

Test Case ID	ST-OTP1
Description	The user wants to access Chats.
Pre-Condition	User should have a valid Phone Number to get OTP.
Steps	1.User Clicks on Chat Module and is directed to OTP verification Activity. 2. User enter phone number. 3.User get the OTP and enters the OTP. 4. After entering valid OTP, user will be directed to profile creation phase.
Expected Results	User Successfully verifies himself to access SafeTrace's Chat.
Status	Pass

6.4.3 Chat Module Test Case

User is redirected to the screen where the User's chat list is shown, by clicking on any user user can send messages, share images, location etc.

Table 6.4.3: Chat Module Test Case

Test Case ID	ST-CM1
Description	The user wants to chat either 1-to-1 or in ring.
Pre-Condition	User selects Either 1-to-1 or ring chat.
Steps	<ol style="list-style-type: none">1. After creating an account, user is directed to chat phase where he gets option to either chat in person or in ring formation.2. User selects 1-to-1 or ring.3. User sends chat, add people to ring, send images, sends location etc.4. Uses panic Button and spot mode.
Expected Results	User Successfully logs in to SafeTrace's Chat.
Status	Pass

6.4.4 Camera Map Test Case

User is shown with the screen where there is option to capture picture and save in gallery ,and save current location where picture is captured. User can view a list of locations, can share and delete it from record.

Table 6.4.4: Camera Map Test Case

Test Case ID	ST-CM2
Description	The user gets/saves live location and picture.
Pre-Condition	User gives permission to Camera and GPS.
Steps	<ol style="list-style-type: none">1. User clicks on Camera Map.2. User clicks to get live location and take a picture of his live place.3. User views the live location where he can delete, share.4. User's taken picture is saved in his phone's internal storage.5. User views his live Location in Map and can further explore map and get location of that place on long press.
Expected Results	User successfully operates Camera Map.
Status	Pass

6.4.5 Travel History Test Case

User can save the location in database and this location is later searched using filter.

Table 6.4.5: Travel History Test Case

Test Case ID	ST-TH1
Description	The user views his saved locations.
Pre-Condition	User has saved previous locations.
Steps	<ol style="list-style-type: none">1. User clicks on Travel History.2. User can save any location on map by longpress on a map.3. Now the user clicks on view saved locations button and is redirected to a list of locations saved previously and can search any location.
Expected Results	User successfully operates Travel History.
Status	Pass

6.4.6 Route Finder Test Case

Route from your current location to any other location can easily be find and shown on map.

Table 6.4.6: Route Finder Test Case

Test Case ID	ST-RF1
Description	The user finds route in the Map.
Pre-Condition	User has given access to GPS.
Steps	<ol style="list-style-type: none">1. User clicks on Route Finder.2. User enters source and destination.3. User get the desired route shown on the Map.
Expected Results	User successfully operates Route Finder.
Status	Pass

6.4.7 Distance Calculator Test Case

User can add markers (by longpress) anywhere on map and then distance between those points is shown on map.

Table 6.4.7: Distance Calculator Test Case

Test Case ID	ST-D1
Description	The user calculates the distance between any points marked on the map.
Pre-Condition	User has given access to GPS.
Steps	<ol style="list-style-type: none">1. User clicks on Distance Module.2. User pinpoint source by long press on map.3. User then pinpoint destination by long press on map.4. User gets the distance calculated.
Expected Results	User successfully operates Distance Module.
Status	Pass

6.5 Compatibility Testing

The system is developed to be used on Smart android Mobile Phones with 94.4

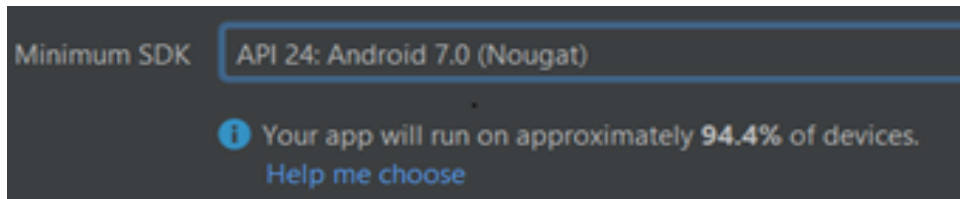


Figure 6.5.1: SDK Version

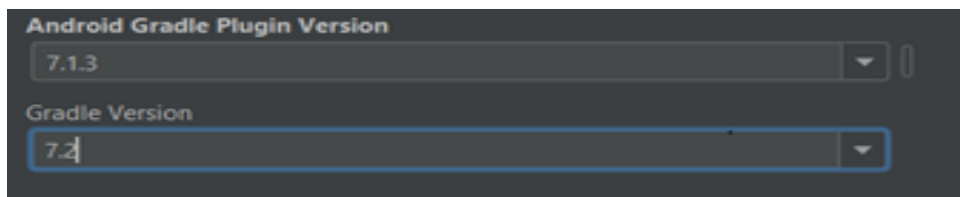


Figure 6.5.2: Android Gradle Plugin Version

```
android {  
    compileSdk 33  
  
    defaultConfig {  
        applicationId "com.example.buddychat"  
        minSdk 21  
        targetSdk 33  
        versionCode 1  
        versionName "1.0"  
    }  
}
```

Figure 6.5.3: Compile SDK and defaultConfig

6.5.1 Exception handling:

Several try-catch blocks are implemented to deal with exceptions.

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

Figure 6.5.4: Exceptional Handling blocks

6.6 Unit Testing

Here we use this technique to test individual module to determine if there are any issues. It helped us to isolate each unit of the system to identify, analyse and fix the defects. The Modules that were tested separately are:

- Login/Register Module
- OTP verification Module
- Chat Module
- Camera Map
- Panic Button
- Map implementation
- Location Tracking
- Location History/Travel History
- Profile Creation and Updating Profile

6.7 Integration Testing

Upon completion of unit testing, the units or modules had integrated which gives raise to integration testing. The purpose of integration testing was to verify the functional, performance, and reliability between the modules that are integrated.

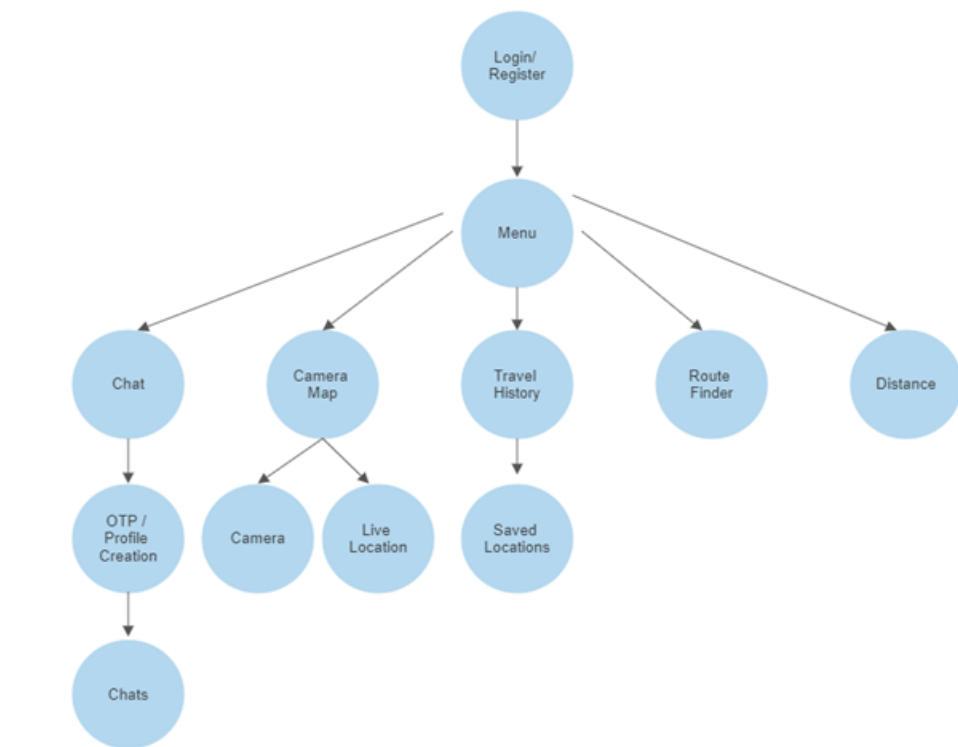


Figure 6.7.1: Integration Testing

Chapter 7

Conclusion

7 Conclusion

Real-time tracking is a problem in which platforms like Google and sygic are already working. Almost everyone in this world has at least used location tracking app once in his life. SafeTrace helps and contributes its solution for this problem. SafeTrace helped us in getting aware of this problem and finding solutions for this problem.

7.1 Learning Outcomes

7.1.1 Awareness

Awareness is the biggest leaning outcome that we got while researching for this Final Year Project. We came to know how important location tracking applications are and how continuously they are being worked on by Google and Sygic for Human ease and comfort. Not only these two, but several other platforms are working on real time tracking. Our small contribution in this issue might not be very useful at the moment but was worth a try.

7.1.2 Team Work

Learning teamwork is extremely important for any final year project. In fact, teamwork is often essential for the success of any project, especially complex and challenging ones. Working in a team allowed us to leverage the strengths of different team members. Everyone brings different skills, knowledge, and experiences to the table. By working together, you can combine your strengths and compensate for each other's weaknesses. In a team, you can divide the workload and responsibilities among team members. This can help to streamline the project and ensure that all tasks are completed in a timely and efficient manner. When you work in a team, you are accountable to your teammates. This can help to ensure that everyone is doing their part and meeting their deadlines. Working in a team requires collaboration, communication, and coordination. These skills are essential in the workplace and learning them during final year project can help prepare us for our future career. Working in a team gave us opportunity to learn from CRC Expert Team. We learned new skills, techniques, and approaches that one may not have been exposed to otherwise. Overall, learning teamwork is crucial for final year projects, as well as for your future career. By working in a team, you can develop essential skills and experiences that will serve you well in the long run.

7.1.3 Familiarity and Scope

Getting familiar with the problem as well as solutions was the best part while doing Final Year Project. We got to know about several softwares, Researches and websites including Android Studio, Google Maps Platform, Firebase, Figma, Wireframe etc. The scope of a location tracking application can be broad or narrow depending on the specific needs and goals of the application. It's important to consider the ethical implications of location tracking, such as privacy concerns and data security, when designing and implementing such an application.

7.2 Future Work

These are the functionalities that safeTrace did but if we ever get the chance to continue this work, we would polish it more and add more functionality and usability to SafeTrace. As we are doing this Final Year Project with Cyber Reconnaissance and combat (CRC Team) Lab of Bahria University Islamabad, we are planning to continue its work and launch it on Play Store to let Actual audience experience it. This would require more time and proper channels to launch it on Play Store. Since improvement is necessary for everything and there is always something in a system that needs to be improved or needs improvement. This version of the application is first version with simple GUI and minimalistic functionalities because of time constraint. In future, we can improve this system through following:

- GUI can be improved to make it more appealing visually to the users. Sounds and Animations can be added.
- The application is currently only developed for android Mobile Phones, it will be released for cross platforms like IOS and Huawei as well as for the windows users in the future.
- Constant updates to meet maximum compatibility.
- Additional features or making the existing feature more useful and helpful.

Appendix A

User Manual

SafeTrace is an Android MOBILE Application designed to be used by everyone easily. Here is a user manual of using SafeTrace

- Download and click on SafeTrace
- SafeTrace splashscreen will be seen that will direct you to the login/register where user will create his/her account.
- Login/Register with valid credentials.
- To use Chat feature, User has to do OTP verification then can proceed to either chat in private or in ring.
- Chat in ring can be either by joining a group or creating a group.
- User can click on Panic Button to generate alert messages that is sent to all members of the group.
- Spot Mode lets the user select a meeting point which can be accessed by other members of the group by directing them to map and exact pinpoint location.
- To use Camera Map feature, User has to give access to location, wifi and Camera.
- Camera feature lets the User take image with camera icon, view saved location by clicking on location icon and can also view his current location on map by clicking on map icon.
- Saved locations can be deleted or shared.
- To use other Tracking features, User has to give access to GPS and switch to map view to find the routes from his current location to any destination
- User can also calculate Straight Line distance between any 2 points on the map on long press.

Appendix B

Guide

- lat — Latitude
- lng — Longitude
- OTP — One Time Password
- app — Application
- del — Delete
- GPS — Global Positioning System
- db — Database
- M — Million
- 3+ — Age above 3 Years
- SMS — short message service
- API — Application programming interface
- PCAM — Partition, Communication, Agglomeration, Mapping
- ring — used for group chat
- IDE — Integrated development environment
- GUI — Graphic User Interface
- Java — Language Used in Android Studio
- gradle — app gradle of android studio
- Firebase — Database name
- SQLite — Database name

References

- [1] *A-GPS Tracker - Apps on Google Play* — *play.google.com*. https://play.google.com/store/apps/details?id=com.giobat.troviamoci&hl=en_US. [Accessed 4-Apr-2023].
- [2] *Family Locator - GPS Tracker - Apps on Google Play* — *play.google.com*. https://play.google.com/store/apps/details?id=com.family.locator.familylocator.app.new.navigacijos.mb&hl=en_US. [Accessed 29-Apr-2023].
- [3] *Family360 - GPS Live Location - Apps on Google Play* — *play.google.com*. https://play.google.com/store/apps/details?id=com.testlab.family360&hl=en_US. [Accessed 29-Apr-2023].
- [4] *Firebase* — *firebase.google.com*. <https://firebase.google.com/>. [Accessed 29-Apr-2023].
- [5] *GeoZilla - Find My Family - Apps on Google Play* — *play.google.com*. https://play.google.com/store/apps/details?id=com.geozilla.family&hl=en_US. [Accessed 29-Apr-2023].
- [6] AIST Global. *Why Do You Need a Location-Based Application?* — *aist.global*. <https://aist.global/en/location-based-application>. [Accessed 29-Apr-2023].
- [7] *Glympse - Share GPS location - Apps on Google Play* — *play.google.com*. <https://play.google.com/store/apps/details?id=com.glympse.android.glympse&hl=en&gl=US>. [Accessed 29-Apr-2023].
- [8] *Google Maps Platform - Location and Mapping Solutions* — *mapsplatform.google.com*. <https://mapsplatform.google.com/>. [Accessed 29-Apr-2023].
- [9] *Sygyic Sygyic GPS Navigation* — *sygyic.com*. <https://www.sygyic.com/>. [Accessed 29-Apr-2023].