Project Report on

NARI SURAKSHA : A WOMEN SAFETY APPLICATION USING DETECTION TECHNOLOGY

Submitted in the partial fulfilment of the requirements

for the award of degree of

Bachelor of Science in Computer Science Honours

University of Calcutta

Submitted by

Arpan Mitra

Roll Number : 213513-21-0012

Registration Number : 513-1111-0308-21

B.Sc. Semester – VI (Honours) Examination – 2024 (under CBCS)

Under the Guidance of

Mrs. Sunandana Mukherjee Banerjee

Department of Computer Science

Sammilani Mahavidyalaya

E.M. Bypass, Baghajatin

Kolkata – 700 094

Sammilani Mahavidyalaya

GOVT. AIDED COLLEGE AFFILIATED
TO UNIVERSITY OF CALCUTTA
NACC ACCREDITED B++ IN 2016 (2ND CYCLE)
& ISO 9001 : 2015 CERTIFIED
E. M. Bypass, Baghajatin, Kolkata - 700 094

Phone : (033) 2462-6869
E-mail : principal.sammilani@gmail.com
info@sammilanimahavidyalaya.ac.in
Website : https://www.sammilanimahavidyalaya.ac.in

Certificate of Approval

This is to certify that the dissertation is the record of Final Year (2024) Project, entitled **Women Safety Application using Detection Technology** carried out by **Arpan Mitra** bearing **Roll No: 213513-21-0012**, student of the Department of Computer Science, Sammilani Mahavidyalaya for the partial fulfilment for the award of the degree of Bachelor of Science and registered under University of Calcutta (**Registration No: 513-1111-0308-21**) under my supervision and guidance.

To the best of my knowledge, this report is based on the original work except for citations and quotations. It has not been previously and concurrently submitted for any other degree or award at any university or institutions.

Mrs. Swagata Saha Sau
Head of the Department
Department of Computer Science
Sammilani Mahavidyalaya

Mrs. Sunandana Mukherjee Banerjee
Guide / Supervisor
Department of Computer Science
Sammilani Mahavidyalaya

External Examiner

ACKNOWLEDGEMENT

First of all, I, **Arpan Mitra**, student of B.Sc. Computer Science Honours, would like to take the opportunity to express my sincere gratitude towards my Project Guide **Mrs. Sunandana Mukherjee Banerjee**, Faculty, Department of Computer Science, Sammilani Mahavidyalaya, for her valuable guidance, support and encouragement during the planning and development of my Semester – VI project work : "**Women Safety Application using Detection Technology**".

I also extend my sincere thanks to all the faculty members of the Computer Science Department and non-teaching staff in our college for creating a positive and creative environment for us to work in. Additionally, I would like to thank my parents – Shri Amiya Kumar Mitra and Smt. Sima Mitra and also all of my friends for their constant support and cooperation. Without their help and encouragement, completing this project work would have been very difficult.

To the best of my knowledge, this report is based on the original work except for citations and quotations. It has not been previously and concurrently submitted for any other degree or award at any university or institutions.

Date :

Arpan Mitra

University Roll No: **213513-21-0012**

University Registration No: **513-1111-0308-21**

ABSTRACT

The “Nari Suraksha” is an Android as well as iOS application developed specifically for safety of women. It enables a woman to send alert notifications to the contact numbers she has designated in case of an emergency. By using the detection (shaking) technology, her current location is determined, and several SOS messages, including her location (latitude and longitude), are sent. In an emergency, the woman can also communicate with her guardian through this application by sending regular messages and images. This application also aims to minimize the battery power consumption for users, allowing them to automatically turn on the battery saver on their devices.

TABLE OF CONTENTS

INTRODUCTION

1.1 Introduction :

In today's rapidly advancing digital age, the safety and security of individuals, particularly women, remain critical concerns worldwide. Despite significant strides in gender equality and social awareness, women continue to face various threats and challenges that jeopardize their personal safety. These threats manifest in different forms including harassment, assault, and domestic violence, making it imperative to leverage modern technology to enhance women's safety. Though this application is used by women, but this can also be applicable for all persons.

The "Women's Safety Application" project addresses this pressing issue by developing a comprehensive mobile application designed to provide women with reliable tools and resources to enhance their safety in real-time. This application aims to empower women by offering features such as emergency alerts, location tracking, and easy access to support networks, ensuring that help is always within reach when needed.

The development of this application is not just a technological endeavour but a social commitment to creating a safer environment for women everywhere. This report details the conceptualization, design, implementation, and potential impact of the "Women's Safety Application," highlighting its significance in the ongoing efforts to enhance women's safety through innovative solutions.

1.2 Domain Description :

The domain of women's safety applications leverages modern mobile technologies to provide essential safety tools for women. Mobile applications, compared to traditional websites, offer several distinct advantages in the context of women's safety. They provide an intuitive and user-friendly interface, optimized for mobile devices, enabling users to quickly access essential safety features with minimal effort. The convenience of having a dedicated application ensures that users can promptly seek help and access resources, regardless of their location. The "Women's Safety Application" is developed using Dart programming language and Flutter framework. Flutter, an open-source UI software development kit created by Google, allows for the development of natively compiled applications for mobile, web, and desktop from a single codebase. This cross-platform capability ensures a consistent and efficient user experience across different devices.

1.2.1 Hardware Description

Device Model – Hp Laptop 14s-dq2xxx

Edition – Microsoft Windows 11 Home Single Language (x64)

Processor – 11th Gen Intel® Core™ i3-1125G4

Installed Ram – 16.0 GB

Installed Solid State Drive (SSD) – 512.0 GB

1.2.2 Software Description

Software Name – Android Studio Iguana | 2023.2.1 (Feb 2024) (IDE for App Development)

Technologies Used – Flutter Framework, Dart Programming Language, Firebase Database

1.3 Motivation :

The motivation behind the development of the "Women's Safety Application" stems from the pressing need to address the pervasive issue of women's safety in our society. Despite advancements in technology and societal progress, women continue to face significant risks to their personal safety in various environments, including public spaces, workplaces, and even their own homes. These risks include harassment, assault, stalking, and other forms of violence, which not only affect the physical well-being of women but also their mental health and overall quality of life. [1]

Statistics and reports from around the world highlight the alarming frequency and severity of incidents involving violence against women. These incidents often go unreported due to fear, stigma, or lack of access to timely assistance. The "Women's Safety Application" aims to fill this gap by leveraging modern mobile technology to offer women a sense of security and support, empowering them to take control of their safety. [2]

The widespread availability and usage of smartphones present an opportunity to create impactful safety solutions. By developing an application that integrates real-time location tracking, emergency alerts, and access to support resources, we can provide women with a powerful tool to enhance their personal safety. The motivation to use the Dart programming language and Flutter framework for this project lies in their ability to create a high-performance, cross-platform application, ensuring accessibility for a wide range of users. [3]

In addition to providing immediate safety features, the application also aims to raise awareness about the importance of women's safety and foster a sense of community. By allowing users to report incidents and mark safe and unsafe areas, the application contributes to a collective effort to identify and address safety concerns in different locations. This not only aids law enforcement and community organizations but also encourages societal change by highlighting areas that require attention and improvement. [4]

So, this project represents a commitment to using our skills and resources to make a positive impact on society. We believe that technology can be a powerful force for good, and through this application, we aspire to contribute to a safer, more supportive environment for women everywhere. [5]

1.4 Scope of the Work :

In today’s world, the scope of a women’s safety app is significant. The project aims to create a comprehensive and user-friendly mobile application that provides essential safety features for women, leveraging modern technology to enhance personal security and well-being. But there can be several gaps exist in the application. Let us explore some of the gaps and how they can be overcome.

One significant gap in the application is that – when a women unintentionally shakes her device, then the SOS alert message including her current location is sent to her emergency contact members. This will make panic to all of her emergency contact members. This is absolutely wrong. Another major gap is that – when a women will be in an emergency and if she doesn’t get a chance to shake her device, then she is really in danger but still any of her contact members can’t know the situation. This is also bad.

By performing task, we can overcome these gaps in the “Women’s Safety Application”. The application offers a pop-up to the user which want to confirm the women’s safety after shaking her device. If she confirms safety within some time duration, then the SOS alert message will not be shared with her emergency contact members, else she is really in trouble and in that case, the alert message will be shared to the emergency members with her current location. Also, when the women don’t get a chance to shake her device in danger, then the voice activation will be enabled within the application and instead of not shaking her device, the SOS alert message including her current location will be sent to her emergency contact members by detecting her voice command.

So, with these solutions, the “Women’s Safety Application” gives the way to perform very inclusively and very efficiently.

REVIEWS OF RELATED WORK

In recent years, there has been a notable shift in the approach to Women's Safety Applications, driven by advancements in technology and an increasing reliance on digital solutions. During our research about our related work, we came across various types of research papers. Following is the list of some related works mentioning approaches and gaps with respect to our works :

[1] – Prof. Shubham Bhadre, Divyen Patil, Sanika Bhasme, Vaibhavi Shilimkar, 2024 "Raksha – The Women's Safety Application", *International Research Journal of Engineering and Technology (IRJET)* e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 11 Issue: 05 , May 2024
doi: https://www.irjet.net/archives/V11/i5/IRJET-v11I564.pdf

[2] – Ms. S.Jayapratha, C.Subashini, R.Vaishnavi, 2024, CREATING APPLICATION ON ANDROID FOR WOMEN'S SAFETY*, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCAAIET – 2024* (VOLUME 12 – ISSUE 01)
doi:https://www.ijert.org/research/creating-application-on-android-for-womens-safety-IJERTCONV12IS01036.pdf

Here are some key findings and research gaps of the application, made in 2024, given below ––

Key Findings	Research Gaps
✓ LiveSafe Explore ✓ Emergency Navigation ✓ SOS Custom Button ✓ Track Me option ✓ Fake Call Creation ✓ Send Voice Alert ✓ 24×7 Support	✚ Shake to send SOS alert ✚ Sending location in background ✚ Sending images in chat app ✚ Low Battery Consumption ✚ Offline Availability ✚ Automatic Recording

[3] – Prof. Roshan Kolte, Prachi Tadse, Priti Nikhare, Vanshika Randive, Snehal Raut, Gayatri Narakhede 2023 "An Android App for empowering Women's Safety and Security"*, International Research Journal of Modernization in Engineering Technology and Science (IRJMETS),* e-ISSN: 2582-5208, Volume:05, Issue:04, April-2023
doi: https://www.doi.org/10.56726/IRJMETS36188

Here are some key findings and research gaps of the application, made in 2023, given below ––

Key Findings	Research Gaps
✓ Emergency Contacts option ✓ Send SOS alert by tapping power button thrice ✓ Siren Alarm by tapping power button thrice to both start and stop ✓ Hidden Spy Camera Detector	✚ Shake to send SOS Alert ✚ Sending images in chat app ✚ Low Battery Consumption ✚ Voice Activation for alert ✚ Create Fake Call ✚ Automatic Recording

[4] – Manisha Sharma , Akhil Bansal , Akansha Sharma , Anisha Verma , Prof. Vinay Singh. “An Android Based Women Safety App” , Volume 10, Issue V, *International Journal for Research in Applied Science and Engineering Technology (IJRASET)* Page No: 4758-4764, ISSN:2321-9653, www.ijraset.com

doi: https://doi.org/10.22214/ijraset.2022.43499

Here are some key findings and research gaps of the application, made in 2022, given below ––

Key Findings	Research Gaps
✓ Emergency Contacts option ✓ Custom Alarm Button (makes siren) ✓ Emergency Call Button ✓ Custom Panic Button (send SOS alerts) ✓ Both Online and Offline Availability ✓ Make sound in contact’s phone instead of silent mode	✚ Shake to send SOS alert ✚ Sending images in chat app ✚ Low Battery Consumption ✚ Voice Activation for alert ✚ Create Fake call ✚ Automatic Recording

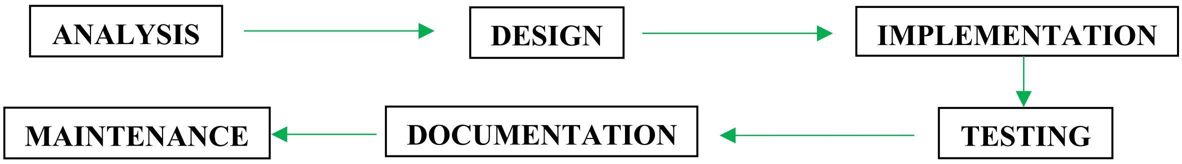
[5] – Mr. Ashutosh More, Ms. Kiran Gawade, Ms. Pradnya Guled, Ms. Shrutika Chippa, Ms. Vijayalaxmi Galgurgi, Prof. Anil Chinchawade, “Sakhi-The Saviour: An Android Application to Help Women in Times of Social Insecurity” , Volume: 08, Issue: 01 Jan 2021, *International Research Journal of Engineering and Technology (IRJET),* e-ISSN: 2395-0056, p-ISSN: 2395-0072, www.irjet.net
doi: https://www.irjet.net/archives/V8/i1/IRJET-V8I1104.pdf

Here are some key findings and research gaps of the application, made in 2021, given below ––

Key Findings	Research Gaps
✓ GPS Module ✓ Identifying location ✓ Shake to send SOS alert ✓ LiveSafe Explore ✓ Emergency Navigation ✓ 24×7 Support ✓ Track Me option	✚ Sending images in chat app ✚ Low Battery Consumption ✚ Voice Activation for alert ✚ Offline Availability ✚ Create Fake Call ✚ Automatic Recording

METHODOLOGY

This part will describe the methodology of the project.



- **Analysis Phase :** In analysis phase, literature review will be conducted by reviewing some similar research papers about "Women's Safety Application". The data and information were gathered and collected during this phase. Thus, the strength and weakness between them are able to compare to each other. After reviewing for the related works, the project objectives and project scope of this android application project can be defined clearly.

- **Design Phase :** The user interface and functionalities for this "Women's Safety Application" will be developed during the design phase. This phase involved designing the overall structure and flow of the app, as well as creating detailed Data Flow Diagrams (DFDs) such as Level-0 DFD, Level-1 DFD and Level-2 DFD. These diagrams provide a visual representation of how data moves through the system and help guide the development process. In this phase, an intuitive user interface is developed. In additions, the hardware and software specification are identified in this phase. The application will be designed by using Flutter framework (which supports Dart programming language) as the graphical user interface is designed completely well.

- **Implementation Phase :** During the implementation phase, the main coding part is started to build the application. This phase involves translating the design specifications into actual working code. The language used for this application development project is Dart. In this phase, the coding will be implemented and the correctness property of the coding will be done. We pay close attention to detail, ensuring that the code is accurate and functional. The implementation phase is a crucial step in bringing the envisioned features and functionality to life, laying the foundation for a robust and user-friendly application.

- **Testing Phase :** Once the complete application is developed with all its functionalities, a testing phase will be conducted. During this phase, the "Women's Safety Application" will be tested by the developer to ensure its proper functioning. The main focus will be on testing the ability to add the emergency contacts, all the emergency functionalities and the overall user interface. The main goal is to verify that the application is performing as intended and meets the desired requirements.

- **Documentation Phase :** After the testing phase is done, the output produced in the previous phase is documented during the last phase which is known as documentation phase. The results of the testing and data collection for this application development project will be also included in the documentation.

- **Maintenance Phase :** The maintenance phase in this application development project involves ongoing support and enhancements to ensure the smooth functioning and continued improvement of the "Women's Safety Application". During the maintenance phase, we work on fixing bugs, improving performance, updating security measures and ensuring compatibility with new technologies. Regular updates keep the app reliable, secure and up-to-date, providing the best user experience. Our dedication to maintenance ensures the app's longevity and success.

3.1 Problem Formulation :

Despite the availability of various safety applications, women continue to face significant risks due to incomplete features and high battery consumption of these apps. The "Women's Safety Application" seeks to address the issues – when the women's device's battery is very low, when they are in danger but doesn't get a chance to shake her device, when they don't get a chance to send some images or videos of that moment with her guardian in chat.

These issues can be resolved by providing optimized battery usage facility, integrating a fake call feature and enabling the automatic transmission of recent images and videos to enhance real-time situational awareness and support.

3.2 Algorithm Description (Step-by-Step Procedure) :

Step 1 : Start

Step 2 : Design an intuitive, user-friendly interface to communicate with user with additional details

Step 3 : Connect the application with Firebase Database

Step 4 : Add Emergency Contacts and store in the database

Step 5 : Create custom SOS Alert button for immediate sending alert message to trusted members

Step 6 : Also enable the option in which the device is Shaked and the SOS Alert message is gone

Step 7 : Make a chat option by which a women can communicate with her guardian in case of emergency

Step 8 : Integrate all the components and thoroughly test the application for functionality, accuracy and user experience

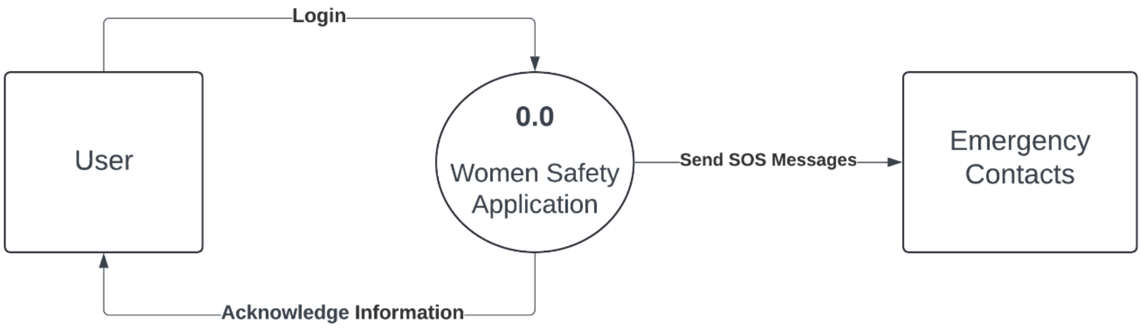
Step 9 : Release the application on appropriate platforms for users to download and use

Step 10 : End

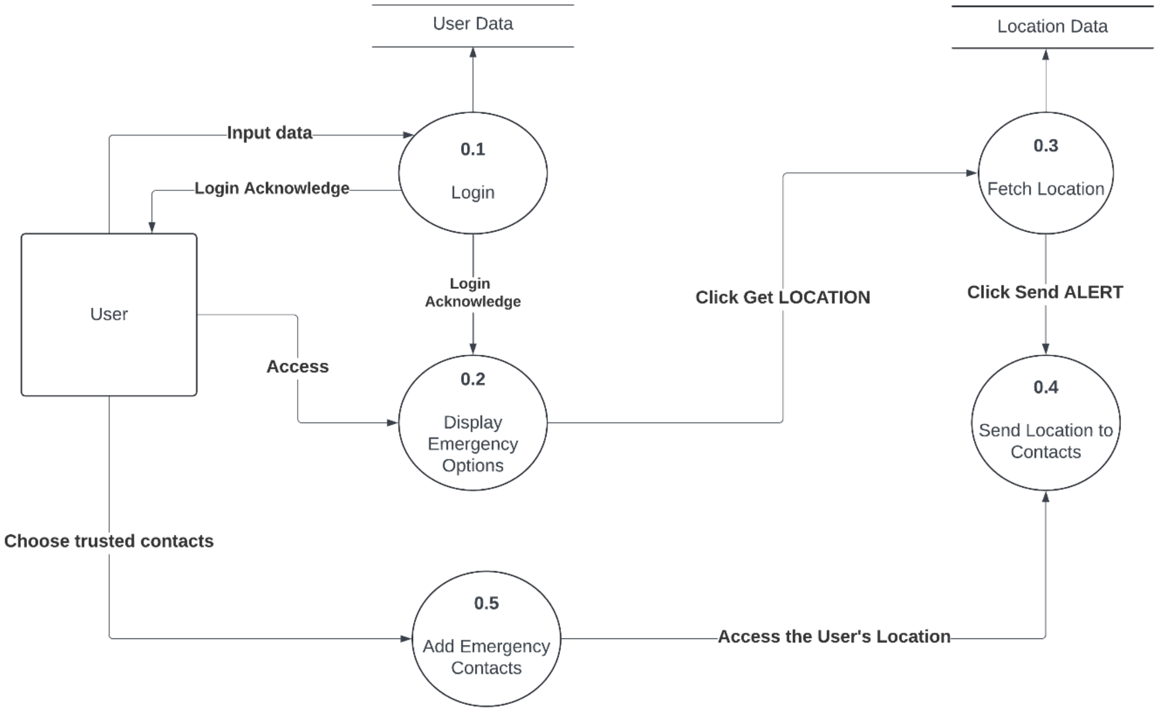
3.3 Design Description :

To describe the working of an application, we can use Data Flow Diagram (DFD). Data Flow Diagram (DFD) is graphical representation of flow of data in an information system. It is capable of depicting incoming data flow, outgoing data flow and stored data. The DFD does not mention anything about how data flows through the system. We can divide DFDs into many levels according to our needs. The first level of DFD is Level-0 DFD (Context Diagram) which is the most abstract view of a software. Then we have Level-1 DFD, then Level-2 DFD and so on.

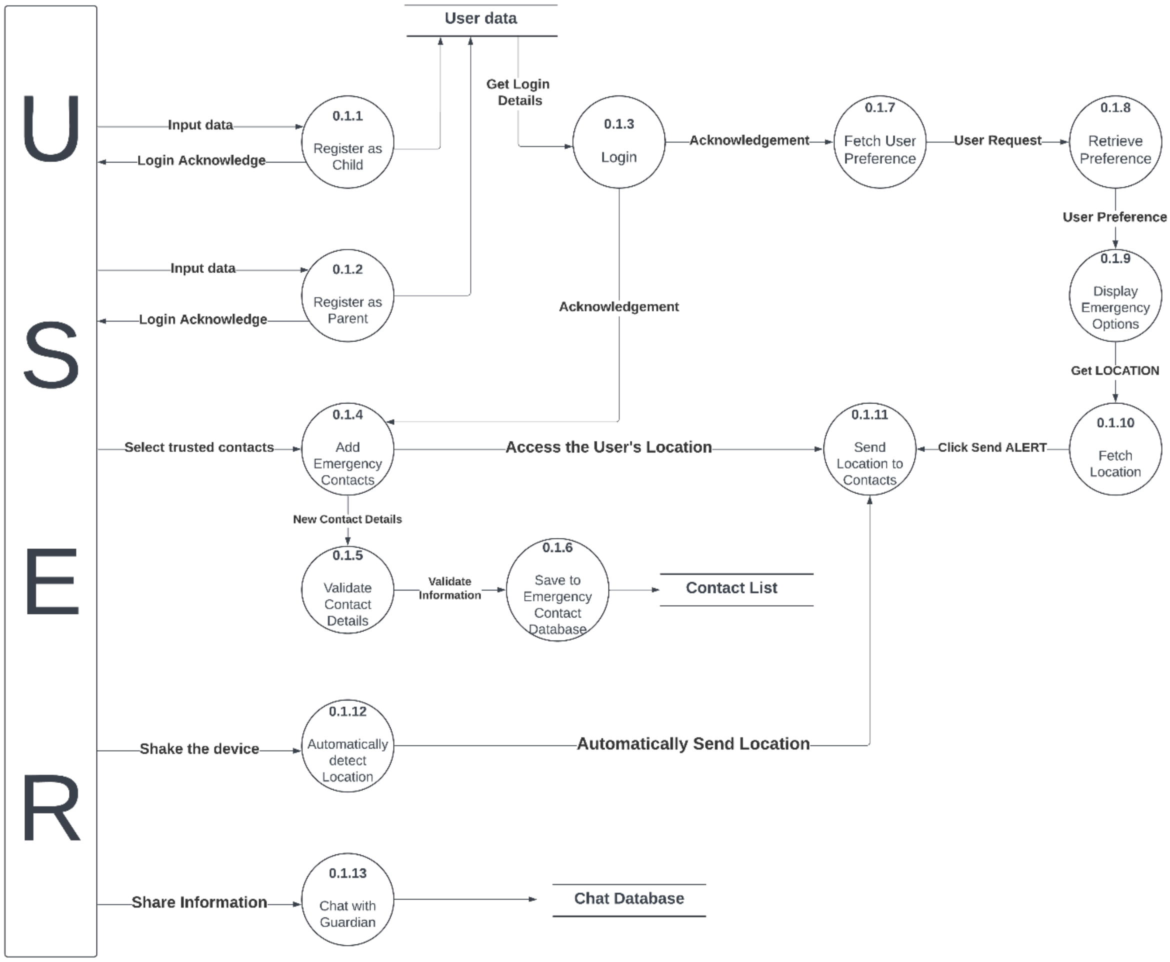
Level-0 Data Flow Diagram (Context Diagram) :



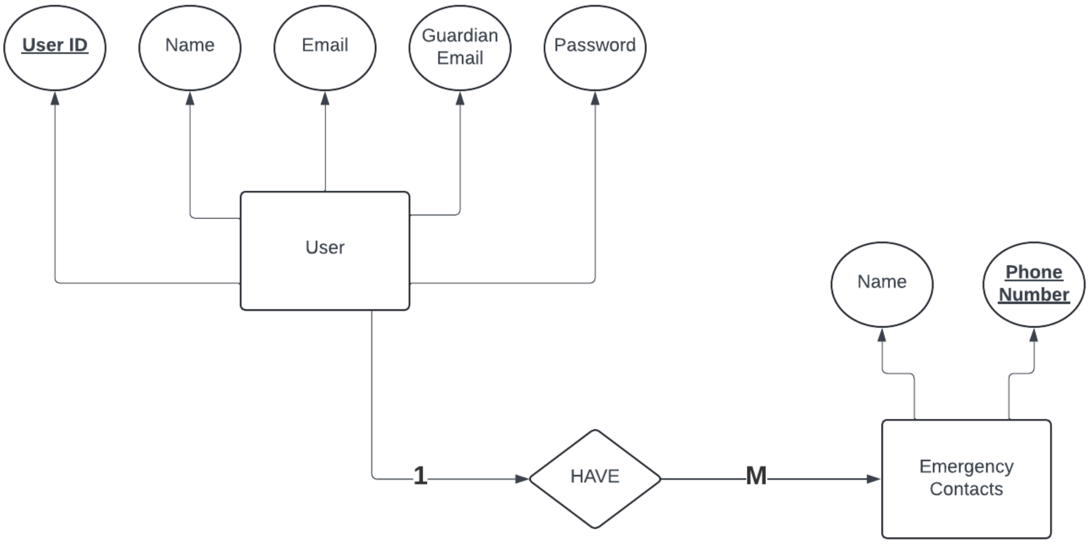
Level-1 Data Flow Diagram :



Level-2 Data Flow Diagram :



Entity-Relationship Diagram (E-R Diagram) :



IMPLEMENTATION

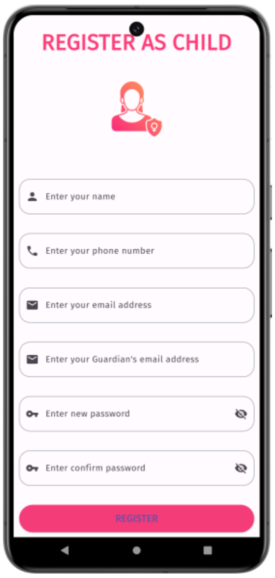
Overview :

“Women’s Safety Application” is a user-friendly Android application that allows users to securely manage and reach at home, providing a seamless and intuitive interface for using the application. With robust security measures, the app empowers users to navigate the world of digital assets with ease and confidence. This Design Of the Application has been done using the following technologies —

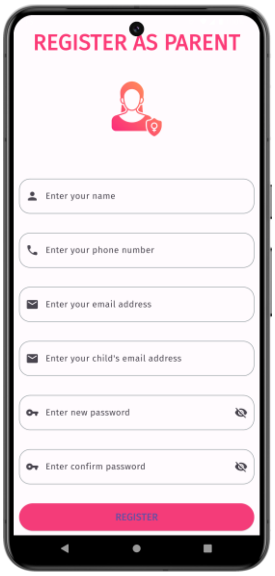
Flutter Framework — Flutter is an open-source UI software development kit created by Google. Flutter is primarily used for developing natively compiled applications for mobile, web, and desktop from a single codebase. It's known for its high-performance, expressive UIs. Flutter uses Dart, a modern programming language also developed by Google. Dart is used both for writing Flutter apps and for handling platform-specific logic. One of the standout features of Flutter is its "Hot Reload" capability, which allows developers to instantly see changes made to the code reflected in the app without restarting the app. This speeds up the development process significantly. Flutter supports multiple platforms: Android and iOS, Web using standards-based web technologies and also Windows, Linux and macOS.

Dart Programming Language — Dart is an open-source, general-purpose programming language developed by Google. Dart is designed for building mobile, desktop, server and web applications. It is optimized for UI development, making it particularly well-suited for frameworks like Flutter. Dart's syntax is similar to other C-style languages such as Java, JavaScript, and C#, making it easier to learn for developers familiar with those languages. Dart code can be compiled ahead-of-time (AOT) into native code for fast performance, and just-in-time (JIT) compiled for development to enable hot reloads. Dart can be used to build web applications, either through direct Dart-to-JavaScript compilation or by using the Dart Web framework. Dart can be used for server-side development, providing libraries and frameworks like Aqueduct and Angel.

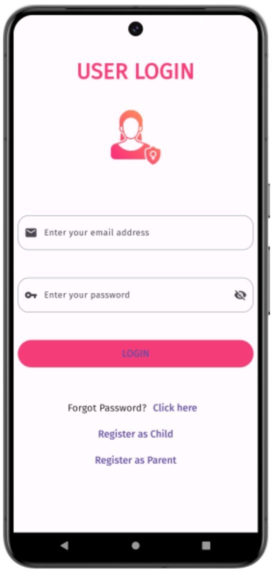
Firebase Database — Firebase Database is a cloud-hosted database provided by Firebase, a platform developed by Google for building mobile and web applications. Firebase offers two types of databases: Firebase Realtime Database and Cloud Firestore. The Firebase Realtime Database is a NoSQL cloud database that stores data in JSON format. Data is stored as a large JSON tree, which can make it complex to structure for some types of applications. Firebase Realtime Database uses Firebase Authentication for user authentication. Security rules are written in JSON and allow for fine-grained control over who can read or write data. Cloud Firestore is a more flexible and scalable NoSQL database compared to Firebase Realtime Database. It supports more advanced querying and indexing capabilities. Data is organized into collections and documents, which allows for a more structured and hierarchical data organization. Similar to Realtime Database, Firestore supports real-time data synchronization across clients. Firestore provides robust offline support, caching data locally to allow read and write operations even when the device is offline. Cloud Firestore is designed to handle large-scale applications and supports automatic scaling.



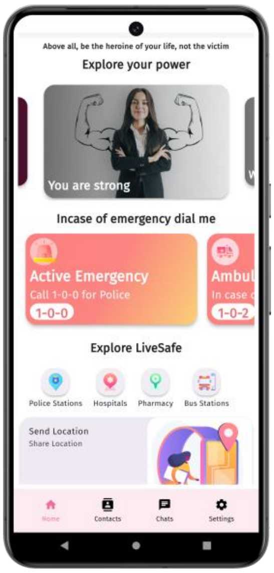
Register Page (1)



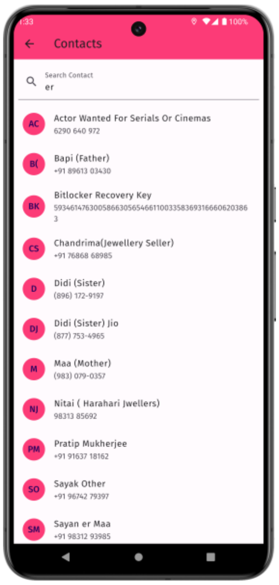
Register Page (2)



Login Page

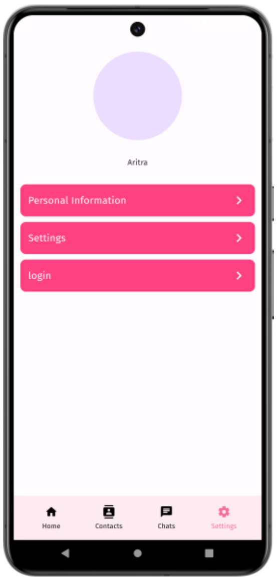


Home Page

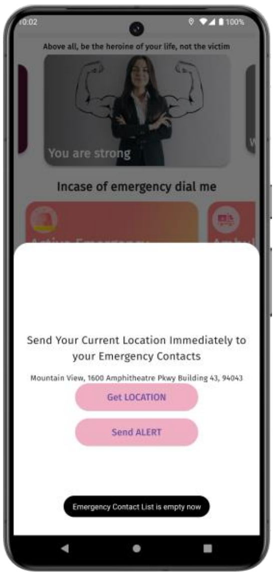


AC	Actor Wanted For Serials Or Cinemas	6290 640 972
B(	Bapi (Father)	+91 89613 03430
BK	Bitlocker Recovery Key	593461476300586630565466110033583693166606203863
CS	Chandrima(Jewellery Seller)	+91 76868 68985
D	Didi (Sister)	(896) 172-9197
DJ	Didi (Sister) Jio	(877) 753-4965
M	Maa (Mother)	(983) 079-0357
NJ	Nitai (Harahari Jwellers)	98313 85692
PM	Pratip Mukherjee	+91 91637 18162
SO	Sayak Other	+91 96742 79397
SM	Sayan er Maa	+91 98312 93985

Contacts Page



Settings Page

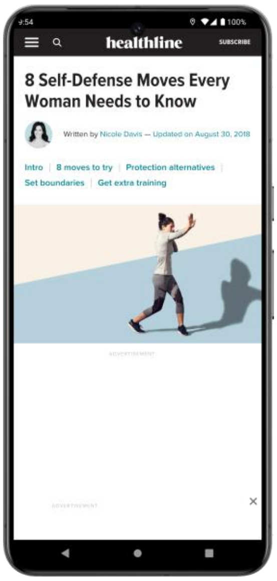

10:02
100%
Above all, be the heroine of your life, not the victim
You are strong
Incase of emergency dial me

Send Your Current Location Immediately to your Emergency Contacts

Mountain View, 1600 Amphitheatre Pkwy Building 43, 94043

Get LOCATION
Send ALERT
Emergency Contact List is empty now

Bottom Sheet


9:54
100%
healthline
SUBSCRIBE

8 Self-Defense Moves Every Woman Needs to Know

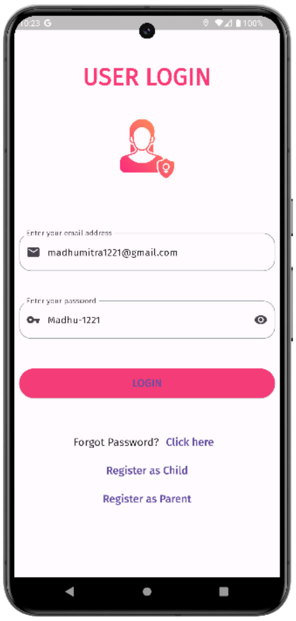
Written by Nicole Davis — Updated on August 30, 2018

Intro | 8 moves to try | Protection alternatives | Set boundaries | Get extra training
ADVERTISEMENT
ADVERTISEMENT

Motivational Page

RESULTS AND DISCUSSIONS

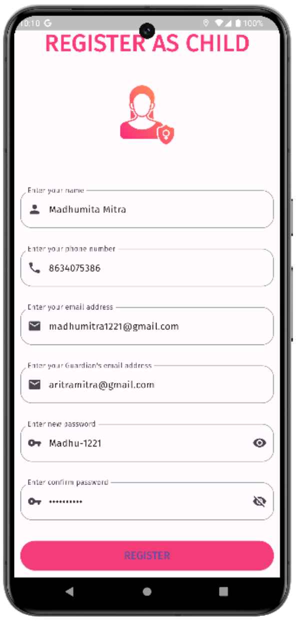
It showcases multiple screenshots of the Women's Safety Application, providing a visual demonstration of its functionality and features.



Enter your email address
madhumitra1221@gmail.com

Enter your password
Madhu-1221

Figure – 1: User Login Page



Enter your name
Madhumita Mitra

Enter your phone number
8634075386

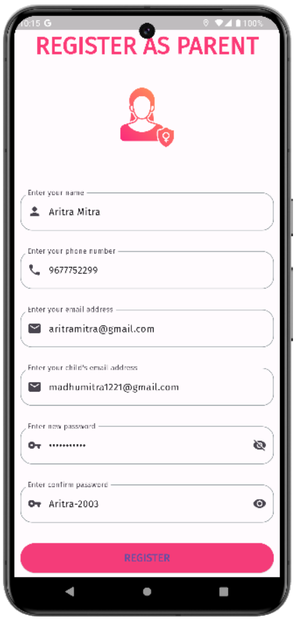
Enter your email address
madhumitra1221@gmail.com

Enter your Guardian's email address
aritramitra@gmail.com

Enter new password
Madhu-1221

Enter confirm password
••••••••••

Figure – 2: Register Page (child)



Enter your name
Aritra Mitra

Enter your phone number
9677752299

Enter your email address
aritramitra@gmail.com

Enter your child's email address
madhumitra1221@gmail.com

Enter new password
•••••••••••

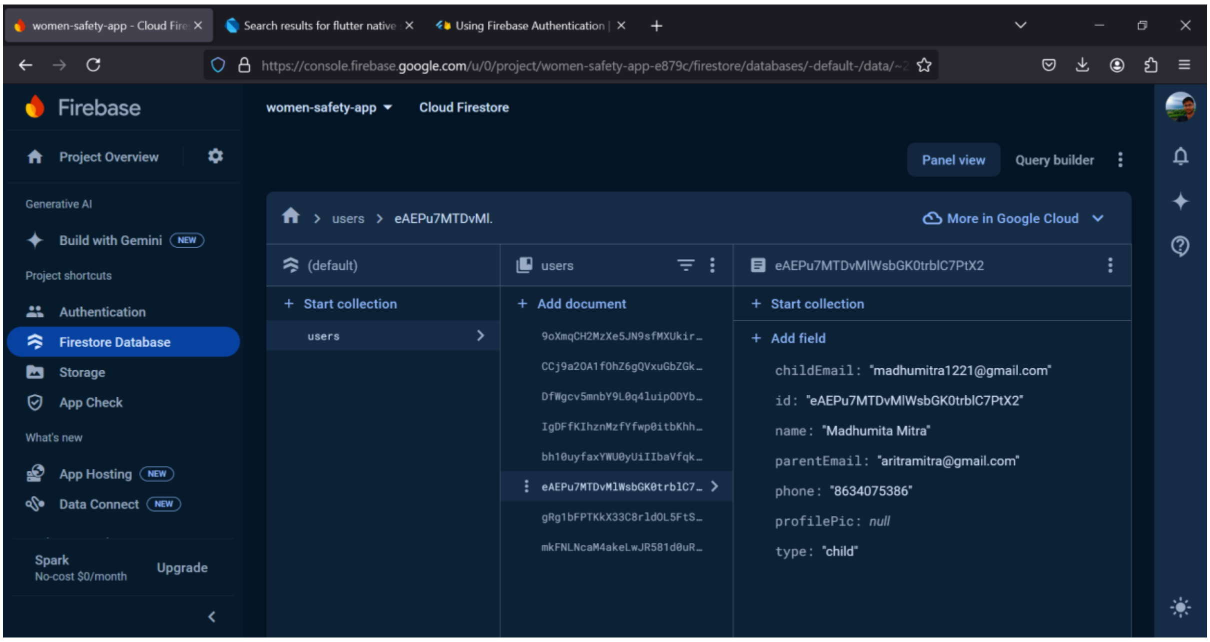
Enter confirm password
Aritra-2003

Figure – 3: Register Page (parent)

Figure 1 is the User Login Page. It is the initial screen, that appears when the user opens the women's safety application for the first time or if the user is not logged in. If the user does not have an account, they can create one account by clicking on the "Register as child" button, which will redirect them to the "Register as Child" page.

Figure 2 is the Register Page for child. Here, the interface is designed with a child-friendly aesthetic, featuring a simple layout and vibrant colors. The page includes input fields for the child's name, phone number and email address along with a separate field for the guardian's email address. It also includes fields for setting and confirming a password with an option to toggle password visibility. The registration process likely requires a parent's involvement as indicated by the guardian's email field, ensuring parental awareness and consent.

Figure 3 is the Register Page for parent. Here, it shows a "Register as Parent" page from a women safety application. This registration form is designed for parents to create an account and link it with their child's account. It includes fields for entering the parent's name, phone number, and email address as well as the child's email address, indicating a connection between the two accounts. The form also has fields for setting and confirming a password, with options to show or hide the password for security.



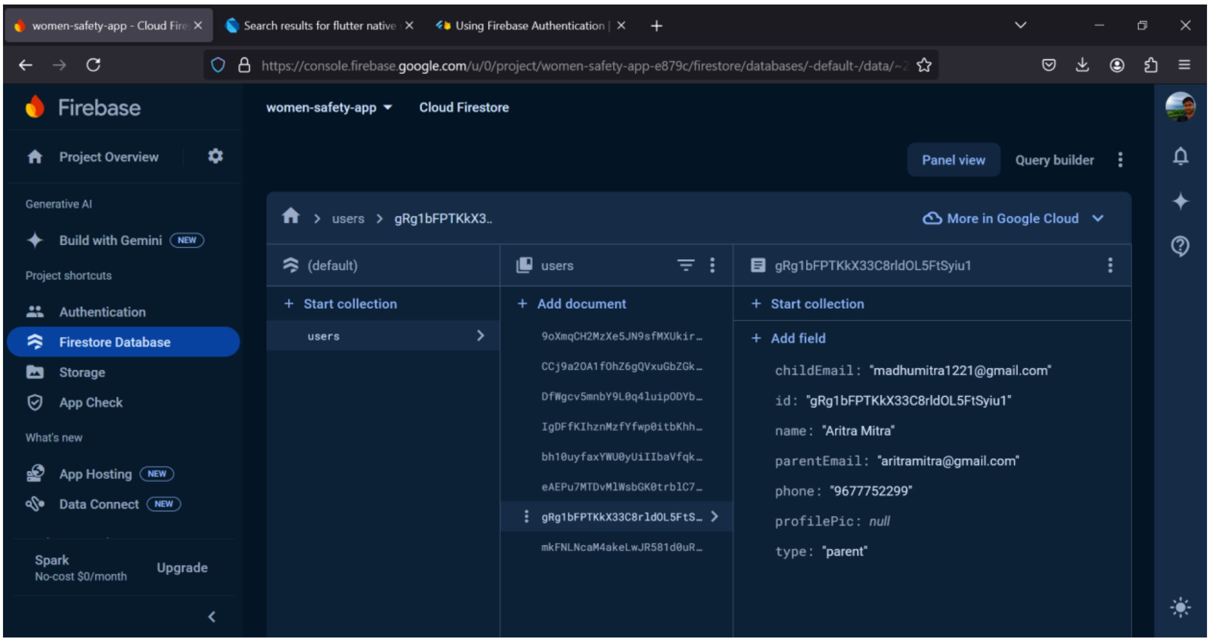
users

- 9oXmqCH2MzXe5JN9sfMXUkir…
- CCj9a2OA1fOhZ6gQVxuGbZGk…
- DfWgcv5mnbY9L0q4luipODYb…
- IgDFfKIhznMzfYfwp0itbKhh…
- bh10uyfaxYWU0yUiIIbaVfqk…
- eAEPu7MTDvMlWsbGK0trblC7…
- gRg1bFPTKkX33C8rldOL5FtS…
- mkFNLNcaM4akeLwJR581d0uR…

eAEPu7MTDvMlWsbGK0trblC7PtX2

```
childEmail: "madhumitra1221@gmail.com"
id: "eAEPu7MTDvMlWsbGK0trblC7PtX2"
name: "Madhumita Mitra"
parentEmail: "aritramitra@gmail.com"
phone: "8634075386"
profilePic: null
type: "child"
```

Figure – 4(a) : Firebase Database (Cloud Firestore) (Child Details)



users

- 9oXmqCH2MzXe5JN9sfMXUkir…
- CCj9a2OA1fOhZ6gQVxuGbZGk…
- DfWgcv5mnbY9L0q4luipODYb…
- IgDFfKIhznMzfYfwp0itbKhh…
- bh10uyfaxYWU0yUiIIbaVfqk…
- eAEPu7MTDvMlWsbGK0trblC7…
- gRg1bFPTKkX33C8rldOL5FtS…
- mkFNLNcaM4akeLwJR581d0uR…

gRg1bFPTKkX33C8rldOL5FtSyiu1

```
childEmail: "madhumitra1221@gmail.com"
id: "gRg1bFPTKkX33C8rldOL5FtSyiu1"
name: "Aritra Mitra"
parentEmail: "aritramitra@gmail.com"
phone: "9677752299"
profilePic: null
type: "parent"
```

Figure – 4(b) : Firebase Database (Cloud Firestore) (Parent Details)

Figure 4(a) and 4(b) are the proofs of Firebase Database. The images display screenshots of a Firebase Firestore database used for a women safety application. The database contains collections named "users" with each document representing individual user records. The database entries show details for a child and parent user. Fields include child email id, name, parent email id, phone and type. The type field is labelled as "child" for child and “parent” for parent indicating the user role within the application. These database entries are part of the system that links parents and children, allowing parents to monitor and manage their child’s activity within the app. The use of Firebase Firestore ensures real-time data updates and secure storage of user information.

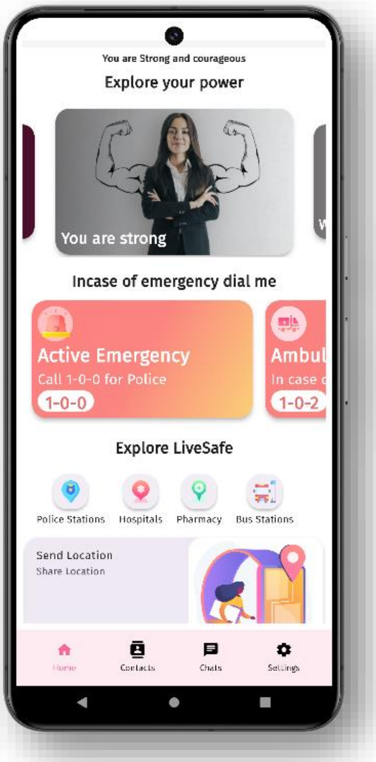


Figure – 5 : Home Page

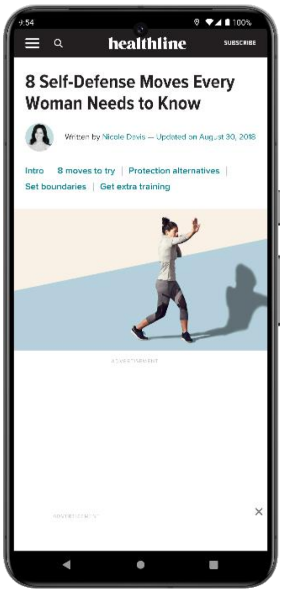
Figure 5 is the User Home Page of the application. The interface is designed to provide quick access to emergency services and essential resources. Key features include :

Motivational Section: There is a motivational message at the top, "You are strong and courageous. Explore your power," accompanied by an image of a woman, likely aimed at empowering users and providing emotional support.

Emergency Contacts and Numbers: The page prominently displays options for contacting emergency services, such as the police ("1-0-0"), ambulance services ("1-0-2") etc, ensuring that users can quickly reach help in critical situations.

LiveSafe Section: The section titled "Explore LiveSafe" suggests additional features or content related to personal safety. It includes buttons for various resources like "Police Stations", "Hospitals", "Pharmacy" and "Bus Stations" which are likely integrated with location services to help users find nearby facilities quickly.

Navigation Bar: A navigation bar at the bottom provides access to different sections of the app, such as "Contacts", "Chats" and "Settings" allowing users to manage their connections, communicate, and customize their app experience.

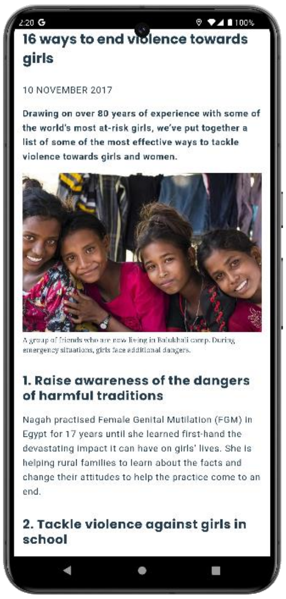


8 Self-Defense Moves Every Woman Needs to Know

Written by Nicole Davis — Updated on August 30, 2018

Intro 8 moves to try | Protection alternatives | Set boundaries | Get extra training

Figure – 6(a):
Motivational Page (1)



16 ways to end violence towards girls

10 NOVEMBER 2017

Drawing on over 80 years of experience with some of the world's most at-risk girls, we've put together a list of some of the most effective ways to tackle violence towards girls and women.

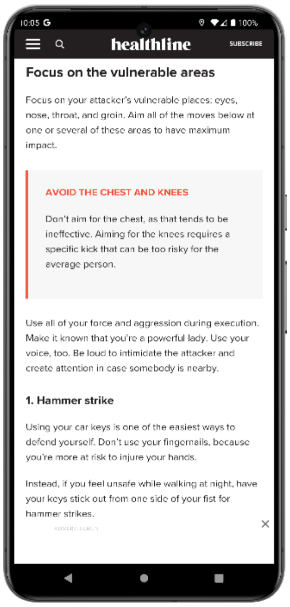
A group of friends who are now living in Balukhali camp. During emergency situations, girls face additional dangers.

1. Raise awareness of the dangers of harmful traditions

Nagah practised Female Genital Mutilation (FGM) in Egypt for 17 years until she learned first-hand the devastating impact it can have on girls' lives. She is helping rural families to learn about the facts and change their attitudes to help the practice come to an end.

2. Tackle violence against girls in school

Figure – 6(b):
Motivational Page (2)



Focus on the vulnerable areas

Focus on your attacker's vulnerable places: eyes, nose, throat, and groin. Aim all of the moves below at one or several of these areas to have maximum impact.

AVOID THE CHEST AND KNEES

Don't aim for the chest, as that tends to be ineffective. Aiming for the knees requires a specific kick that can be too risky for the average person.

Use all of your force and aggression during execution. Make it known that you're a powerful lady. Use your voice, too. Be loud to intimidate the attacker and create attention in case somebody is nearby.

1. Hammer strike

Using your car keys is one of the easiest ways to defend yourself. Don't use your fingernails, because you're more at risk to injure your hands.

Instead, if you feel unsafe while walking at night, have your keys stick out from one side of your fist for hammer strikes.

Figure – 6(c):
Motivational Page (3)

Figure 6(a), 6(b) and 6(c) are some motivational page. The images depict a "Motivational Section" within the application, featuring articles and tips related to self-defence and empowerment. These provide practical advices, focusing on raising awareness and promoting protective measures and specify self-defence techniques, such as targeting vulnerable areas of an attacker to educate and empower women by providing valuable information and skills for their safety and well-being.

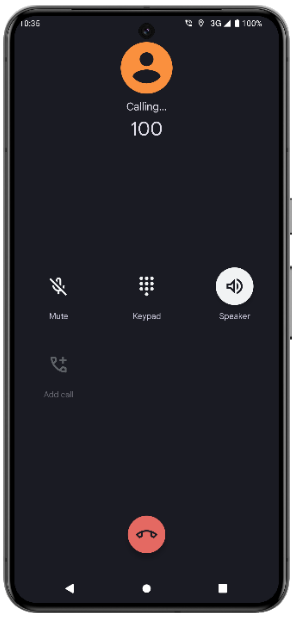


Figure – 7(a) :
Police Caller Page

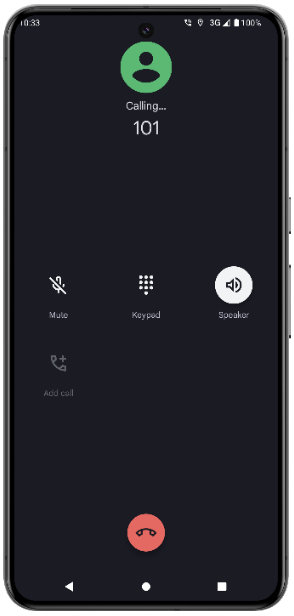


Figure – 7(b) : Fire
Brigade Caller Page

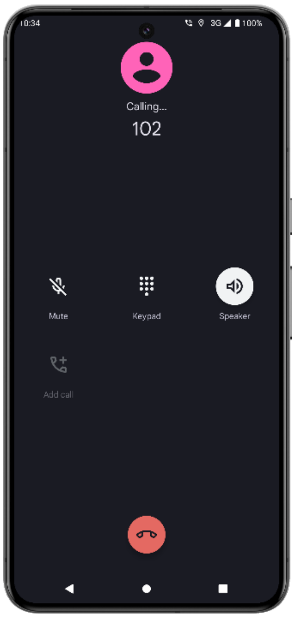


Figure – 7(c) :
Ambulance Caller Page

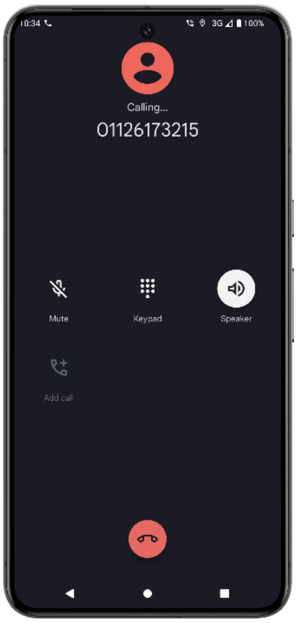
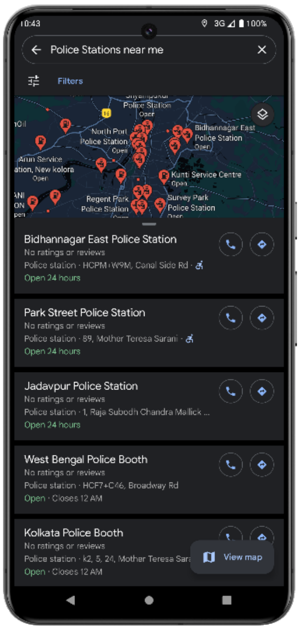


Figure – 7(d) : Indian
Army Caller Page

The images depict the Emergency Caller Page of the application. In these screenshots, different emergency contact numbers (100 for Police, 101 for Fire Brigade, 102 for Ambulance, 01126173215 for Indian Army) are shown as being dialled. The interface features options like mute, keypad and speaker, indicating a standard call screen layout. The color-coded avatars (orange, green) might represent different types of emergency contacts or categories.


10:43
3G 100%
Police Stations near me
Filters
Police Station
North Port Police Station Open
Bidhannagar East Police Station Open
Arun Service ation, New kolora Open
Kunti Service Centre Open
Regent Park Police Station
Survey Park Police Station
View map

Bidhannagar East Police Station
No ratings or reviews
Police station · HCPM+W9M, Canal Side Rd
Open 24 hours

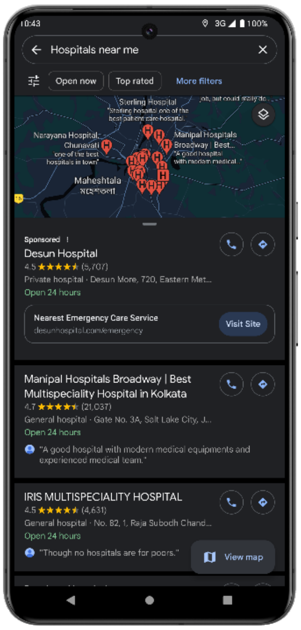
Park Street Police Station
No ratings or reviews
Police station · 89, Mother Teresa Sarani
Open 24 hours

Jadavpur Police Station
No ratings or reviews
Police station · 1, Raja Subodh Chandra Mallick ...
Open 24 hours

West Bengal Police Booth
No ratings or reviews
Police station · HCF7+C46, Broadway Rd
Open · Closes 12 AM

Kolkata Police Booth
No ratings or reviews
Police station · k2, 5, 24, Mother Teresa Sar
Open · Closes 12 AM

Figure – 8(a) : Google Map (Police Stations)

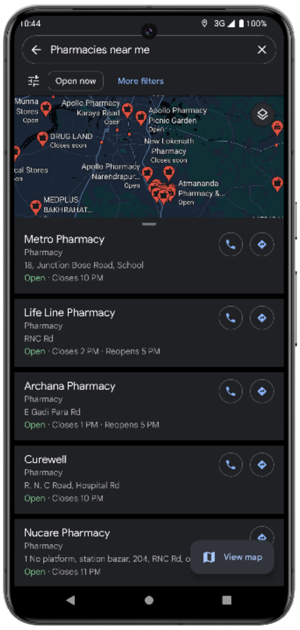

10:43
3G 100%
Hospitals near me
Open now
Top rated
More filters
Sterling Hospital "Sterling hospital one of the best patient care hospital.
Narayana Hospital, Chunavati "one of the best hospitals in town"
Manipal Hospitals Broadway | Best... "A good hospital with modern medical..."
Maheshtala
মহেশতলা
View map

Sponsored
Desun Hospital
4.5 ★★★★★ (5,707)
Private hospital · Desun More, 720, Eastern Met...
Open 24 hours
Nearest Emergency Care Service
desunhospital.com/emergency
Visit Site

Manipal Hospitals Broadway | Best Multispeciality Hospital in Kolkata
4.7 ★★★★★ (21,037)
General hospital · Gate No. 3A, Salt Lake City, J...
Open 24 hours
"A good hospital with modern medical equipments and experienced medical team."

IRIS MULTISPECIALITY HOSPITAL
4.5 ★★★★★ (4,631)
General hospital · No. 82, 1, Raja Subodh Chand...
Open 24 hours
"Though no hospitals are for poors."

Figure – 8(b) : Google Map (Hospitals)


10:44
3G 100%
Pharmacies near me
Open now
More filters
Munna Stores Open
Apollo Pharmacy Karaya Road Open
Apollo Pharmacy Picnic Garden Open
DRUG LAND Closes soon
New Lokenath Pharmacy Closes soon
Apollo Pharmacy Narendrapur... Open
Atmananda Pharmacy &... Open
MEDPLUS BAKHRAHAT...
View map

Metro Pharmacy
Pharmacy
18, Junction Bose Road, School
Open · Closes 10 PM

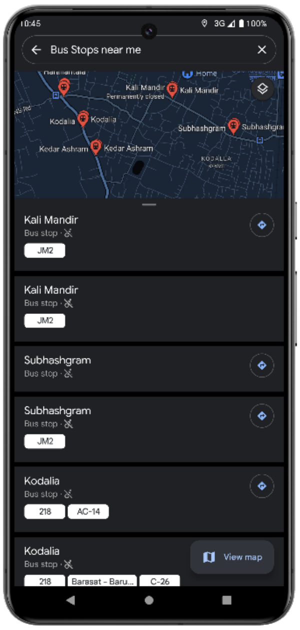
Life Line Pharmacy
Pharmacy
RNC Rd
Open · Closes 2 PM · Reopens 5 PM

Archana Pharmacy
Pharmacy
E Gadi Para Rd
Open · Closes 1 PM · Reopens 5 PM

Curewell
Pharmacy
R. N. C Road, Hospital Rd
Open · Closes 10 PM

Nucare Pharmacy
Pharmacy
1 No platform, station bazar, 204, RNC Rd, o
Open · Closes 11 PM

Figure – 8(c) : Google Map (Pharmacies)



Kali Mandir
Bus stop ·
JM2

Kali Mandir
Bus stop ·
JM2

Subhashgram
Bus stop ·

Subhashgram
Bus stop ·
JM2

Kodalia
Bus stop ·
218 AC-14

Kodalia
Bus stop ·
218 Barasat - Baru... C-26

Figure – 8(d) : Google Map (Bus Stops)

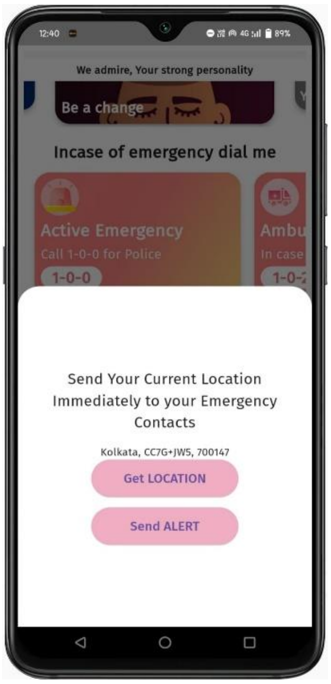
Figure 8(a), 8(b), 8(c) and 8(d) are some emergency locations nearby the user.

“Police Stations near me” page displays a map with red markers indicating the locations of nearby police stations. It includes a list of police stations with their names and contact numbers, allowing users to quickly find and call the nearest station.

“Hospitals near me” page shows a map with hospital locations marked. It provides details like hospital names, contact numbers, ratings, and sometimes additional information like special services or hours of operation.

“Pharmacies near me” page lists nearby pharmacies, marked on a map and listed below with names and phone numbers. Users can quickly access information about the nearest pharmacies, making it easy to find medical supplies in emergencies.

“Bus Stops near me” page shows bus stop locations on a map, along with a list of nearby stops and routes. It's useful for users needing public transportation information in an emergency situation.



Send Your Current Location Immediately to your Emergency Contacts

Kolkata, CC7G+JW5, 700147

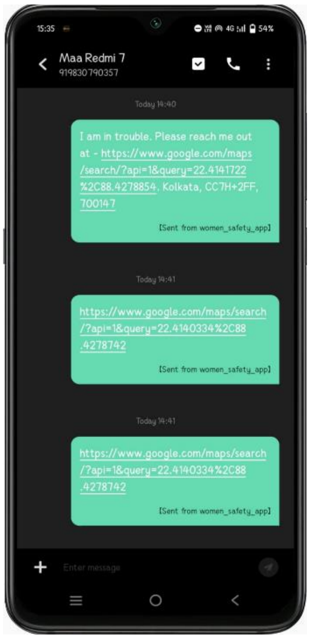
Figure – 9: Bottom
Sheet (Custom Button)

Figure 9 is the Bottom Sheet in which the custom “Get LOCATION” and “Send ALERT” button are present. It displays a feature where users can quickly send their current location to their emergency contacts. The interface shows two main options :

Get Location: This button allows users to retrieve their current location details, which are then displayed on the screen. The location information includes the city name and postal code.

Send Alert: This button enables users to send an emergency alert along with their current location to pre-selected emergency contacts. This feature is crucial for providing immediate assistance by notifying trusted contacts of the user's whereabouts.

The feature is designed to be easily accessible and user-friendly, providing a quick way to share location information in case of an emergency.



15:35 54%

Maa Redmi 7
919830790357

Today 14:40

I am in trouble. Please reach me out at - https://www.google.com/maps/search/?api=1&query=22.4141722%2C88.4278854, Kolkata, CC7H+2FF, 700147

[Sent from women_safety_app]

Today 14:41

https://www.google.com/maps/search/?api=1&query=22.4140334%2C88.4278742

[Sent from women_safety_app]

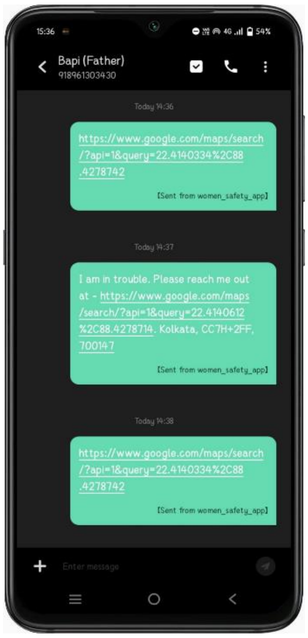
Today 14:41

https://www.google.com/maps/search/?api=1&query=22.4140334%2C88.4278742

[Sent from women_safety_app]

Enter message

Figure – 10(a) : Messages Page (from Child)



15:36 54%

Bapi (Father)
918961303430

Today 14:36

https://www.google.com/maps/search/?api=1&query=22.4140334%2C88.4278742

[Sent from women_safety_app]

Today 14:37

I am in trouble. Please reach me out at - https://www.google.com/maps/search/?api=1&query=22.4140612%2C88.4278714, Kolkata, CC7H+2FF, 700147

[Sent from women_safety_app]

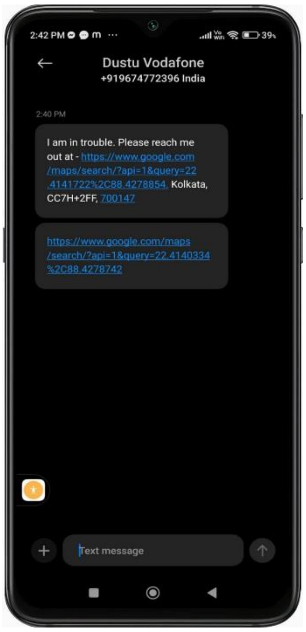
Today 14:38

https://www.google.com/maps/search/?api=1&query=22.4140334%2C88.4278742

[Sent from women_safety_app]

Enter message

Figure – 10(b) : Messages Page (from Child)



2:42 PM 39%

Dustu Vodafone
+919674772396 India

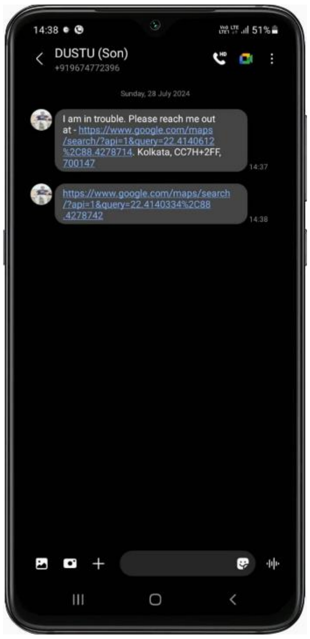
2:40 PM

I am in trouble. Please reach me out at - https://www.google.com/maps/search/?api=1&query=22.4141722%2C88.4278854, Kolkata, CC7H+2FF, 700147

https://www.google.com/maps/search/?api=1&query=22.4140334%2C88.4278742

Text message

Figure – 10(c) : Messages Page (to Contact)



14:38 51%

DUSTU (Son)
+919674772396

Sunday, 28 July 2024

I am in trouble. Please reach me out at - https://www.google.com/maps/search/?api=1&query=22.4140612%2C88.4278714, Kolkata, CC7H+2FF, 700147 14:37

https://www.google.com/maps/search/?api=1&query=22.4140334%2C88.4278742 14:38

Figure – 10(d) : Messages Page (to Contact)

Figure 10(a), 10(b), 10(c) and 10(d) are Messages' Pages. It depicts messages sent from a child to their parents using a custom ALERT button and by shaking the device twice. The messages indicate that the child is in trouble and include a Google Maps link to their location, along with some messages.

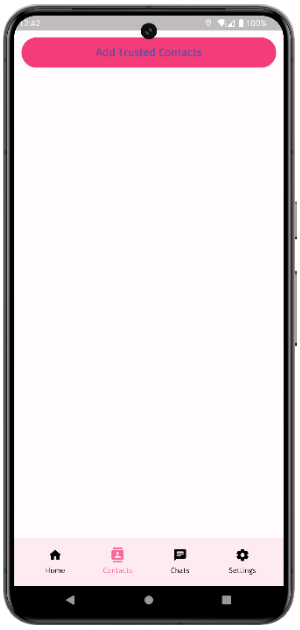
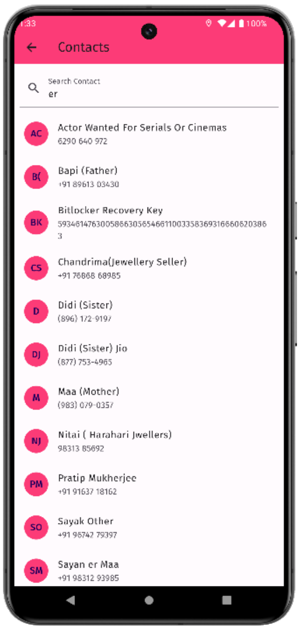


Figure – 11(a) : Contacts Page (Add Option)



er

AC Actor Wanted For Serials Or Cinemas
6290 640 972

B(Bapi (Father)
+91 89613 03430

BK Bitlocker Recovery Key
593461476300586630565466110033583693166606203863

CS Chandrima(Jewellery Seller)
+91 76868 68985

D Didi (Sister)
(896) 172-9197

DJ Didi (Sister) Jio
(877) 753-4965

M Maa (Mother)
(983) 079-0357

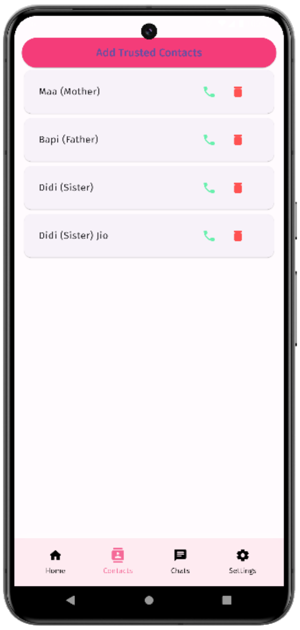
NJ Nitai (Harahari Jwellers)
98313 85692

PM Pratip Mukherjee
+91 91637 18162

SO Sayak Other
+91 96742 79397

SM Sayan er Maa
+91 98312 93985

Figure – 11(b) : Contacts Page (Choose Contact)



Maa (Mother)

Bapi (Father)

Didi (Sister)

Didi (Sister) Jio

Figure – 11(c) : Contact Page (Selected Contacts)

Figure 11(a), 11(b) and 11(c) are Contacts Page. It is designed to send emergency messages to selected contacts. Here's a breakdown of the features :

Contacts Selection Page : This screen allows the user to add trusted contacts who will receive SOS messages in case of an emergency. The interface includes a prominent button labelled "Add Trusted Contacts".

Contact List View : A list of contacts is displayed from which the user can select. This list shows the names and numbers of various contacts, allowing the user to choose which contacts to add to their trusted list.

Selected Contacts View : This screen displays the contacts that have been added as trusted contacts. Each contact has options for making a call or removing them from the list. This setup enables the user to quickly send emergency alerts to these selected contacts by shaking the device or using custom ALERT button (which is present in the Bottom sheet in Home page) within the application.

These features are designed to enhance personal safety by providing an easy way to notify trusted individuals during an emergency.

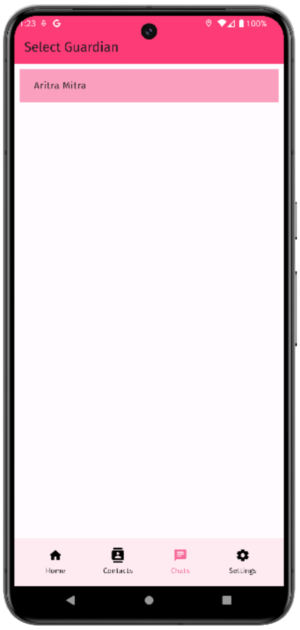


Figure – 12(a) : Chat Page (with Parent)

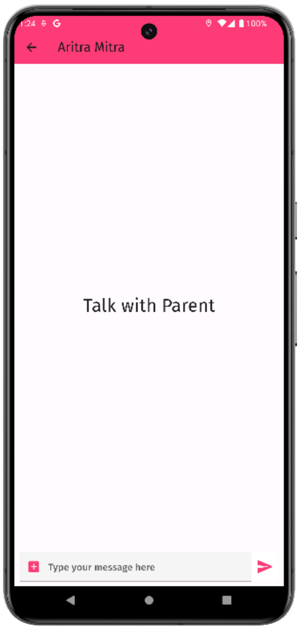
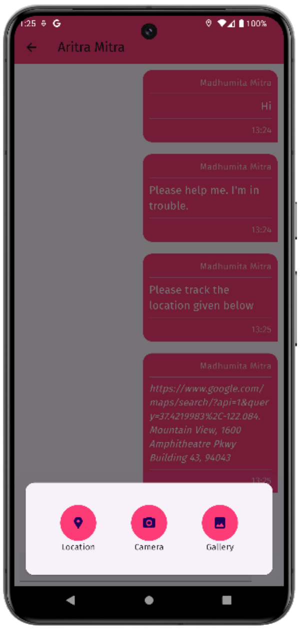


Figure – 12(b) : Chat Page (Main Chatting Page)



Madhumita Mitra
Hi
13:24

Madhumita Mitra
Please help me. I'm in trouble.
13:24

Madhumita Mitra
Please track the location given below
13:25

Madhumita Mitra
https://www.google.com/maps/search/?api=1&query=37.4219983%2C-122.084. Mountain View, 1600 Amphitheatre Pkwy Building 43, 94043
13:25

Figure – 12(c) : Chat Page (Selected Options)

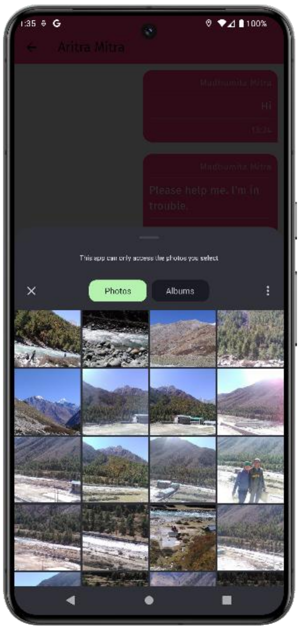


Figure – 12(d) : Chat Page (Photos Selection)

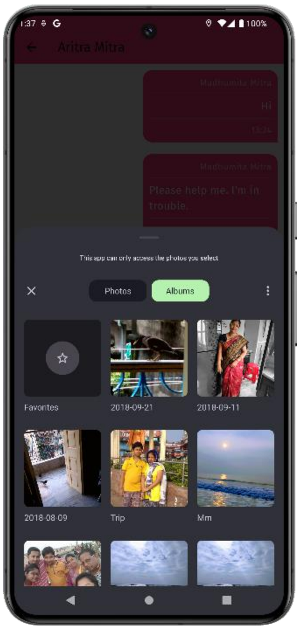


Figure – 12(e) : Chat Page (Photos Selection from Albums)

Figure 12(a), 12(b), 12(c), 12(d) and 12(e) are some portions of the Chat Page. It is a pivotal component in enhancing communication and safety within the application. Designed specifically for parents and children, this feature offers a secure platform for sharing real-time information and staying connected. It provides various functionalities, including the sharing of custom messages, current location updates, and photos of the moment. This section explores the importance and potential impact of the Chat Page feature on user experience and safety.

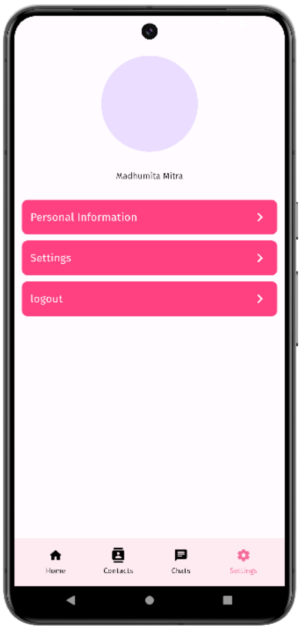
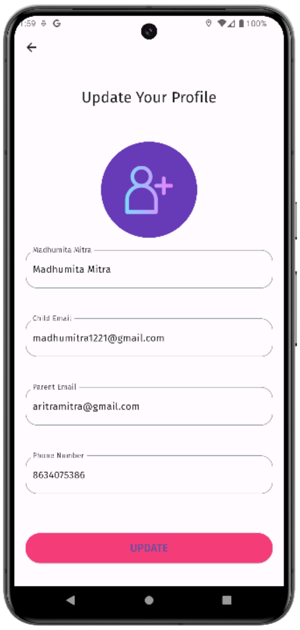


Figure – 13(a) : Settings Page


Update Your Profile

Madhumita Mitra: Madhumita Mitra

Child Email: madhumitra1221@gmail.com

Parent Email: aritramitra@gmail.com

Phone Number: 8634075386

Figure – 13(b) : Update Profile Page

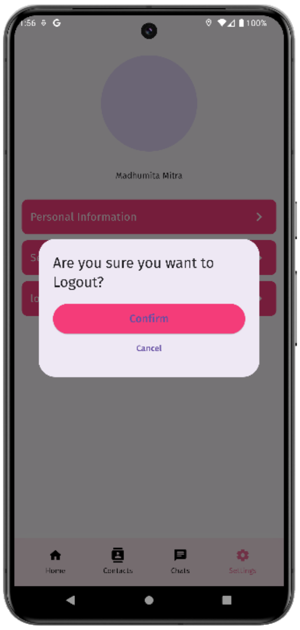

Are you sure you want to Logout?

Figure – 13(c) : Settings Page (Logout Popup)

Figure – 13(d) : Progress Indicator

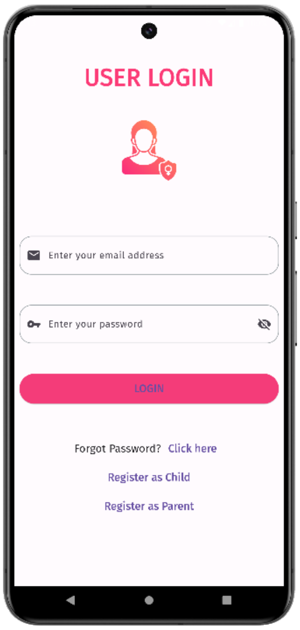


Figure – 13(e) : User Login Page

figure 13(a), 13(b), 13(c), 13(d) and 13(e) are Settings Page in the application. It is a fundamental component of this safety application, providing users with essential tools to manage their personal information and preferences. This feature is designed to offer a user-friendly interface for updating profile details, uploading profile photos and securely logging out of the application. This discussion delves into the significance, functionality, and security considerations of the Settings Page, emphasizing its role in enhancing user experience and ensuring data integrity.

CONCLUSIONS

In conclusion, the "Women's Safety Application" successfully leverages modern technology to provide a robust platform for enhancing the personal safety and security of women. Through features like real-time location tracking, custom SOS alerts, and emergency contact integration, the application offers immediate assistance in critical situations, thereby reducing the risk and impact of potential threats. Though this application is used by women, but this can also be applicable for all persons.

This project not only highlights the importance of comprehensive safety measures but also addresses significant gaps found in existing applications. By incorporating features such as voice-activated SOS alerts and a confirmation pop-up for unintentional alerts, the application ensures more accurate and reliable emergency responses, minimizing false alarms and enhancing user trust. [1, 2, 3, 4, 5]

Emphasizing user experience, the application is designed to be intuitive and accessible, catering to women of all ages and technical proficiency levels. User feedback and rigorous testing have played a pivotal role in refining the app's functionalities, ensuring it meets the real-world needs of its users effectively. Troubleshoot of this application also provides by us with respect to maintenance of the software.

The project sets the foundation for future enhancements and innovations. Ongoing efforts to optimize battery consumption, integrate additional safety features, and expand the app's reach through community and organizational partnerships are essential for its continued success and relevance.

In conclusion, the "Women's Safety Application" represents a significant step forward in utilizing technology for personal safety. It addresses critical gaps in existing solutions [1 – 5], prioritizes user experience, and offers a scalable platform for ongoing improvement and societal benefit. The project underscores the potential of mobile technology in making tangible contributions to women's safety and well-being.

REFERENCES

Research Papers :-

[1] – Prof. Shubham Bhadre, Divyen Patil, Sanika Bhasme, Vaibhavi Shilimkar, 2024 "Raksha – The Women's Safety Application", *International Research Journal of Engineering and Technology (IRJET)* e-ISSN: 2395-0056, p-ISSN: 2395-0072, Volume: 11 Issue: 05 , May 2024
doi: https://www.irjet.net/archives/V11/i5/IRJET-v11I564.pdf

[2] – Ms. S.Jayapratha, C.Subashini, R.Vaishnavi, 2024, CREATING APPLICATION ON ANDROID FOR WOMEN'S SAFETY, *INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCAAIET – 2024* (VOLUME 12 – ISSUE 01)
doi: https://www.ijert.org/research/creating-application-on-android-for-womens-safety-IJERTCONV12IS01036.pdf

[3] – Prof. Roshan Kolte, Prachi Tadse, Priti Nikhare, Vanshika Randive, Snehal Raut, Gayatri Narakhede 2023 "An Android App for empowering Women's Safety and Security"*, International Research Journal of Modernization in Engineering Technology and Science (IRJMETS),* e-ISSN: 2582-5208, Volume:05, Issue:04, April-2023 doi: https://www.doi.org/10.56726/IRJMETS36188

[4] – Manisha Sharma , Akhil Bansal , Akansha Sharma , Anisha Verma , Prof. Vinay Singh. "An Android Based Women Safety App" , Volume 10, Issue V*, International Journal for Research in Applied Science and Engineering Technology (IJRASET)* Page No: 4758-4764, ISSN:2321-9653,
doi: https://doi.org/10.22214/ijraset.2022.43499

[5] – Mr. Ashutosh More, Ms. Kiran Gawade, Ms. Pradnya Guled, Ms. Shrutika Chippa, Ms. Vijayalaxmi Galgurgi, Prof. Anil Chinchawade, "Sakhi-The Saviour: An Android Application to Help Women in Times of Social Insecurity" , Volume: 08, Issue: 01 Jan 2021, *International Research Journal of Engineering and Technology (IRJET),* e-ISSN: 2395-0056, p-ISSN: 2395-0072, www.irjet.net
doi: https://www.irjet.net/archives/V8/i1/IRJET-V8I1104.pdf

[6] – Ms. Priyanka Y. Gonde, Mr. P.B. Ghewari, "REVIEW PAPER ON WOMEN SAFETY SYSTEM", Volume: 08, Issue: 01 Jan 2021, e-ISSN: 2395-0056 , p-ISSN: 2395-0072
doi: https://www.irjet.net/archives/V8/i1/IRJET-V8I1315.pdf

[7] – Salina Akter Shila, "Safety Apps for Women in Android"
doi: https://www.slideshare.net/slideshow/safety-app-for-woman/250579298

[8] – Hari Krishnan, Mrs. Usha, "Women Safety Application" , Volume: 7 , Issue: 10 , ISSN: 2349-6002 doi: https://www.academia.edu/64696498/Women_Safety_Application

Book References :-

[1] – Fundamentals of Software Engineering *by Prof. Rajib Mal* (2014), *Department of Computer Science and Engineering, IIT Kharagpur*

Web References :-

[1] – https://docs.flutter.dev/get-started/install (Flutter and it's SDK)

[2] – https://dart.dev/get-dart (Dart and it's SDK)

[3] – https://pub.dev/ (All the Dart Packages used for Dependencies)

[4] – https://console.firebase.google.com (Firebase Console)

[5] – https://firebase.flutter.dev/docs (FlutterFire Documentation Overview)

ANNEXURE

women_safety_app\lib\main.dart :-

```
import 'package:firebase_core/firebase_core.dart';
import 'package:flutter/material.dart';
import 'package:google_fonts/google_fonts.dart';
import 'package:women_safety_app/child/bottom_page.dart';
import 'package:women_safety_app/db/share_pref.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/parent/parent_home_screen.dart';
import 'package:women_safety_app/utils/constants.dart';
import 'package:women_safety_app/utils/flutter_background_services.dart';
import 'package:audioplayers/audioplayers.dart';
import 'package:sensors_plus/sensors_plus.dart';

final navigatorkey = GlobalKey<ScaffoldMessengerState>();

void main() async {
  WidgetsFlutterBinding.ensureInitialized();
  // print("Initializing Firebase...");
  await Firebase.initializeApp();
  // print("Firebase initialized");
  await MySharedPreference.init();
  await initializeService();
  // await Sensors.platformSensors.requestPermission('accelerometer');
  runApp(const MyApp());
}

class MyApp extends StatelessWidget {
  const MyApp({super.key});

  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: "Flutter Demo",
      debugShowCheckedModeBanner: false,
      theme: ThemeData(
        textTheme: GoogleFonts.firaSansTextTheme(
          Theme.of(context).textTheme,
        ),
        primarySwatch: Colors.blue,
      ),
      // home: LoginScreen()
      home: FutureBuilder(future: MySharedPreference.getUserType(), builder:
(BuildContext context, AsyncSnapshot snapshot) {
        if (snapshot.data == "") {
          return LoginScreen();
        }
        if (snapshot.data == "child"){
          return BottomPage();
        }
        if (snapshot.data == "parent"){
          return ParentHomeScreen();
        }
        return progressIndicator(context);
      }
    ));
  }
}
```

women_safety_app\lib\child\register_child.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:women_safety_app/components/PrimaryButton.dart';
import 'package:women_safety_app/components/SecondaryButton.dart';
import 'package:women_safety_app/components/custom_textfield.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/model/user_model.dart';
import 'package:women_safety_app/utils/constants.dart';

class RegisterChildScreen extends StatefulWidget {
  @override
  State<RegisterChildScreen> createState() => _RegisterChildScreenState();
}

class _RegisterChildScreenState extends State<RegisterChildScreen> {
  // const RegisterChildScreen({super.key});
  bool isPasswordShown = true;
  bool isConfirmPasswordShown = true;
  final _formkey = GlobalKey<FormState>();
  final _formdata = Map<String, Object>();
  bool isLoading = false;

  _onSubmit() async {
    _formkey.currentState!.save();
    if(_formdata['password'] != _formdata['confirm_password']) {
      dialogueBox(context, 'Your entered password and confirm password should
be equal');
    }
    else {
      progressIndicator(context);
      try {
        setState(() {
          isLoading = true;
        });
        UserCredential userCredential = await
FirebaseAuth.instance.createUserWithEmailAndPassword(
            email: _formdata['email'].toString(),
            password: _formdata['password'].toString()
        );
        if(userCredential.user != null) {
          setState(() {
            isLoading = true;
          });
          final v = userCredential.user!.uid;
          DocumentReference<Map<String, dynamic>> db =
FirebaseFirestore.instance.collection('users').doc(v);
          final user = UserModel(
              id: v,
              name: _formdata['name'].toString(),
              phone: _formdata['phone'].toString(),
              childEmail: _formdata['email'].toString(),
              parentEmail: _formdata['guardian_email'].toString(),
              type: 'child'
          );
          final jsonData = user.toJson();
          await db.set(jsonData).whenComplete(() {
            goTo(context, LoginScreen());
            setState(() {
              isLoading = false;
            });
          });
          // goTo(context, LoginScreen());
        }
      } on FirebaseAuthException catch (e) {
        if (e.code == 'weak-password') {
          dialogueBox(context, 'The password provided is too weak.');
          print('The password provided is too weak.');
        } else if (e.code == 'email-already-in-use') {
          dialogueBox(context, 'The account already exists for that email.');
          print('The account already exists for that email.');
        }
        setState(() {
          isLoading = false;
        });
      }
      catch (e) {
        print(e);
        setState(() {
          isLoading = false;
        });
        dialogueBox(context, e.toString());
      }
    }
    print(_formdata['email']);
    print(_formdata['password']);
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Padding(
          padding: const EdgeInsets.symmetric(horizontal: 10),
          child: Stack(
            children: [
              isLoading ? progressIndicator(context)
              : SingleChildScrollView(
                child: Column(
                  children: [
                    Container(
                      height: MediaQuery.of(context).size.height * 0.3,
                      child: Column(
                        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                        children: [
                          Text(
                            'REGISTER AS CHILD',
                            textAlign: TextAlign.center,
                            style: TextStyle(fontSize: 40, color: primaryColor,
fontWeight: FontWeight.bold),
                          ),
                          Image.asset('assets/logo.png', height: 100, width: 100),
                        ],
                      ),
                    ),
                    Container(
                      height: MediaQuery.of(context).size.height * 0.75,
                      child: Form(
                        key: _formkey,
                        child: Column(
                          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                          children: [
                            CustomTextField(
                              hintText: 'Enter your name',
                              textInputAction: TextInputAction.next,
                              keyboardtype: TextInputType.name,
                              prefix: Icon(Icons.person_rounded),
                              onsave: (name) {
                                _formdata['name'] = name ?? "";
                              },
                              validate: (name) {
                                if (name!.isEmpty || name.length<3) {
                                  return 'Please enter valid name';
                                }
                                return null;
                              },
                            ),
                            CustomTextField(
                              hintText: 'Enter your phone number',
                              textInputAction: TextInputAction.next,
                              keyboardtype: TextInputType.phone,
                              prefix: Icon(Icons.phone_rounded),
                              onsave: (phone) {
                                _formdata['phone'] = phone ?? "";
                              },
                              validate: (phone) {
                                if (phone!.isEmpty || phone.length<10) {
                                  return 'Please enter valid phone number';
                                }
                                return null;
                              },
                            ),
                            CustomTextField(
                              hintText: 'Enter your email address',
                              textInputAction: TextInputAction.next,
                              keyboardtype: TextInputType.emailAddress,
                              prefix: Icon(Icons.email_rounded),
                              onsave: (email) {
                                _formdata['email'] = email ?? "";
                              },
                              validate: (email) {
                                if (email!.isEmpty || email.length<12 ||
!email.contains("@") || !email.contains(".com")) {
                                  return 'Please enter valid email address';
                                }
                                // return null;
                              },
                            ),
                            CustomTextField(
                              hintText: 'Enter your Guardian\'s email address',
                              textInputAction: TextInputAction.next,
                              keyboardtype: TextInputType.emailAddress,
                              prefix: Icon(Icons.email_rounded),
                              onsave: (guardian_email) {
                                _formdata['guardian_email'] = guardian_email ?? "";
                              },
                              validate: (guardian_email) {
                                if (guardian_email!.isEmpty || guardian_email.length<12
|| !guardian_email.contains("@") || !guardian_email.contains(".com")) {
                                  return 'Please enter valid email address';
                                }
                                // return null;
                              },
                            ),
                            CustomTextField(
                              hintText: 'Enter new password',
                              isPassword: isPasswordShown,
                              prefix: Icon(Icons.vpn_key_rounded),
                              validate: (password) {
                                if (password!.isEmpty || password.length<7) {
                                  return 'Please enter valid password';
                                }
                                return null;
                              },
                              onsave: (password) {
                                _formdata['password'] = password ?? "";
                              },
                              suffix: IconButton(
                                  onPressed: () {
                                    setState(() {
                                      isPasswordShown = !isPasswordShown;
                                    });
                                  },
                                  icon: isPasswordShown ? Icon(Icons.visibility_off) :
Icon(Icons.visibility)
                              ),
                            ),
                            CustomTextField(
                              hintText: 'Enter confirm password',
                              isPassword: isConfirmPasswordShown,
                              prefix: Icon(Icons.vpn_key_rounded),
                              validate: (password) {
                                if (password!.isEmpty || password.length<7) {
                                  return 'Please enter valid password'; // 'Confirm
password should match with your original password';
                                }
                                return null;
                              },
                              onsave: (password) {
                                _formdata['confirm_password'] = password ?? "";
                              },
                              suffix: IconButton(
                                  onPressed: () {
                                    setState(() {
                                      isConfirmPasswordShown =
!isConfirmPasswordShown;
```

```
                                    });
                                  },
                                  icon: isConfirmPasswordShown ?
Icon(Icons.visibility_off) : Icon(Icons.visibility)
                              ),
                            ),
                            PrimaryButton(title: 'REGISTER', onPressed: () {
                              if (_formkey.currentState!.validate()) {
                                _onSubmit();
                              }
                            }),
                          ],
                        ),
                      ),
                    ),
                    SecondaryButton(title: 'Login into your account', onPressed: () {
                      goTo(context, LoginScreen());
                    })
                  ],
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\child\child_login_screen.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:women_safety_app/child/bottom_page.dart';
import 'package:women_safety_app/components/PrimaryButton.dart';
import 'package:women_safety_app/components/SecondaryButton.dart';
import 'package:women_safety_app/components/custom_textfield.dart';
import 'package:women_safety_app/child/register_child.dart';
import 'package:women_safety_app/db/share_pref.dart';
import 'package:women_safety_app/parent/parent_home_screen.dart';
import 'package:women_safety_app/parent/parent_register_screen.dart';
import 'package:women_safety_app/utils/constants.dart';

class LoginScreen extends StatefulWidget {
  @override
  State<LoginScreen> createState() => _LoginScreenState();
}

class _LoginScreenState extends State<LoginScreen> {
  // const LoginScreen({super.key});
  bool isPasswordShown = true;
  final _formkey = GlobalKey<FormState>();
  final _formdata = Map<String, Object>();
  bool isLoading = false;

  _onSubmit() async {
    _formkey.currentState!.save();
    try {
      if (mounted) {
        setState(() {
          isLoading = true;
        });
      }

      UserCredential userCredential = await
FirebaseAuth.instance.signInWithEmailAndPassword(
          email: _formdata['email'].toString(),
          password: _formdata['password'].toString()
      );

      if (mounted) {
        setState(() {
          isLoading = false;
        });
      }

      if(userCredential.user != null) {

FirebaseFirestore.instance.collection('users').doc(userCredential.user!.uid).get().
then((value) {
          if (mounted) {
            print("====> ${value['type']}");
            if (value['type'] == 'parent'){
              MySharedPreference.saveUserType('parent');
              goTo(context, ParentHomeScreen());
            }
            else {
              MySharedPreference.saveUserType('child');
              goTo(context, BottomPage());
            }
          }
        });
      }
    } on FirebaseAuthException catch (e) {
      // print("123212321\n");
      // print(e);
      // print("\n123212321");
      if (mounted) {
        setState(() {
          isLoading = false;
        });
      }

      // if (e.code == 'user-not-found') {
      //   dialogueBox(context, 'No user found for that email.');
      //   print('No user found for that email.');
      // } else if (e.code == 'wrong-password') {
      //   dialogueBox(context, 'Wrong password provided for that user.');
      //   print('Wrong password provided for that user.');
      // }

      if (e.code == 'invalid-credential') {
        dialogueBox(context, "Please enter valid credential");
        print('Invalid Credential');
      }
    }
    print(_formdata['email']);
    print(_formdata['password']);
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Padding(
          padding: const EdgeInsets.all(8.0),
          child: Stack(
            children: [
              isLoading ? progressIndicator(context)
              : SingleChildScrollView(
                child: Column(
                    children: [
                      Container(
                        height: MediaQuery.of(context).size.height * 0.3,
                        child: Column(
                          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                          children: [
                            Text(
                              'USER LOGIN',
                              style: TextStyle(fontSize: 40, color: primaryColor,
  fontWeight: FontWeight.bold),
                            ),
                            Image.asset('assets/logo.png', height: 100, width: 100),
                          ],
                        ),
                      ),
                      Container(
                        height: MediaQuery.of(context).size.height * 0.4,
                        child: Form(
                          key: _formkey,
                          child: Column(
                            mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                            children: [
                              CustomTextField(
                                  hintText: 'Enter your email address',
                                  textInputAction: TextInputAction.next,
                                  keyboardtype: TextInputType.emailAddress,
                                  prefix: Icon(Icons.email_rounded),
                                  onsave: (email) {
                                    _formdata['email'] = email ?? "";
                                  },
                                  validate: (email) {
                                    if (email!.isEmpty || email.length<3 ||
  !email.contains("@") || !email.contains(".com")) {
                                      return 'Please enter correct email';
                                    }
                                    // return null;
                                  },
                              ),
                              CustomTextField(
                                  hintText: 'Enter your password',
                                  isPassword: isPasswordShown,
                                  prefix: Icon(Icons.vpn_key_rounded),
                                  validate: (password) {
                                    if (password!.isEmpty || password.length<7) {
                                      return 'Please Enter Correct Password';
                                    }
                                    return null;
                                  },
                                  onsave: (password) {
                                    _formdata['password'] = password ?? "";
                                  },
                                  suffix: IconButton(
                                      onPressed: () {
                                        setState(() {
                                          isPasswordShown = !isPasswordShown;
                                        });
                                      },
                                      icon: isPasswordShown ? Icon(Icons.visibility_off) :
  Icon(Icons.visibility)
                                  ),
                              ),
                              PrimaryButton(
                                  title: 'LOGIN',
                                  onPressed: () {
                                    // progressIndicator(context);
                                    if (_formkey.currentState!.validate()) {
                                      _onSubmit();
                                    }
                                  }
                              ),
                            ],
                          ),
                        ),
                      ),
                      Container(
                        child: Row(
                          mainAxisAlignment: MainAxisAlignment.center,
                          children: [
                            Text("Forgot Password?", style: TextStyle(fontSize: 18)),
                            SecondaryButton(title: 'Click here', onPressed: () {}),
                          ],
                        ),
                      ),
                      SecondaryButton(title: 'Register as Child', onPressed: () {
                        goTo(context, RegisterChildScreen());
                      }),
                      SecondaryButton(title: 'Register as Parent', onPressed: () {
                        goTo(context, RegisterParentScreen());
                      })
                    ],
                  ),
                ),
              ],
            ),
          ),
        ),
      );
    }
  }
```

women_safety_app\lib\child\bottom_page.dart :-

```
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:women_safety_app/child/bottom_screens/add_contacts.dart';
import 'package:women_safety_app/child/bottom_screens/chat_page.dart';
import
'package:women_safety_app/child/bottom_screens/child_home_page.dart';
import 'package:women_safety_app/child/bottom_screens/review_page.dart';
import 'package:women_safety_app/components/fab_bar_bottom.dart';
import 'package:women_safety_app/profile_mode/settings.dart';

class BottomPage extends StatefulWidget {
  const BottomPage({super.key});

  @override
  State<BottomPage> createState() => _BottomPageState();
}

class _BottomPageState extends State<BottomPage> {
  int currentIndex = 0;
  List<Widget> pages = [
    HomeScreen(),
    AddContactsPage(),
    // ChatPage(),
    CheckUserStatusBeforeChat(),
    // ProfilePage(),
    // ReviewPage(),
    SettingsPage()
  ];

  onTapped(int index) {
    setState(() {
      currentIndex = index;
    });
  }

  @override
  Widget build(BuildContext context) {
    return PopScope(
        onPopInvoked: (didPop) async {
          if (Navigator.canPop(context)) {
            Navigator.of(context).pop();
          }
          else {
            SystemNavigator.pop();
          }
          // return true;
        },
        child: DefaultTabController(
          initialIndex: currentIndex,
          length: pages.length,
          child: Scaffold(
              body: pages[currentIndex],
              bottomNavigationBar: FABBottomAppBar(
                onTabSelected: onTapped,
                items: [
                  FABBottomAppBarItem(iconData: Icons.home_rounded, text:
"Home"),
                  FABBottomAppBarItem(iconData: Icons.contacts_rounded, text:
"Contacts"),
                  FABBottomAppBarItem(iconData: Icons.chat_rounded, text:
"Chats"),
                  // FABBottomAppBarItem(iconData: Icons.rate_review_rounded,
text: "Ratings"),
                  FABBottomAppBarItem(iconData: Icons.settings_rounded, text:
"Settings"),
                ],
              )
          ),
        )
    );
  }
}
```

women_safety_app\lib\child\bottom_screens\add_contacts.dart :-

```
import 'package:flutter/material.dart';
import 'package:flutter_phone_direct_caller/flutter_phone_direct_caller.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:sqflite/sqflite.dart';
import 'package:women_safety_app/child/bottom_screens/contacts_page.dart';
import 'package:women_safety_app/components/PrimaryButton.dart';
import 'package:women_safety_app/db/db_services.dart';
import 'package:women_safety_app/model/contactsm.dart';

class AddContactsPage extends StatefulWidget {
  const AddContactsPage({super.key});

  @override
  State<AddContactsPage> createState() => _AddContactsPageState();
}

class _AddContactsPageState extends State<AddContactsPage> {
  DatabaseHelper databaseHelper = DatabaseHelper();
  List<TContact>? contactList;
  int count = 0;

  void showList() {
    Future<Database> dbFuture = databaseHelper.initializeDatabase();
    dbFuture.then((database) {
      Future<List<TContact>> contactListFuture =
databaseHelper.getContactList();
```

```
      contactListFuture.then((value) {
        setState(() {
          this.contactList = value;
          this.count = value.length;
        });
      });
    });
  }

  void deleteContact(TContact contact) async {
    int result = await databaseHelper.deleteContact(contact.id);
    if (result != 0) {
      Fluttertoast.showToast(msg: "Contacts removed Successfully");
      showList();
    }
  }

  @override
  void initState() {
    WidgetsBinding.instance.addPostFrameCallback((timeStamp) {
      showList();
    });
    super.initState();
  }

  @override
  Widget build(BuildContext context) {
    if (contactList == null) {
      contactList = [];
    }
    return SafeArea(
      child: Container(
        padding: EdgeInsets.all(12),
        child: Column(
          children: [
            PrimaryButton(
                title: "Add Trusted Contacts",
                onPressed: () async {
                  bool result = await Navigator.push(context,
MaterialPageRoute(builder: (context) => ContactsPage()));
                  if (result == true) {
                    showList();
                  }
                }),
            Expanded(
              child: ListView.builder(itemCount: count, itemBuilder: (BuildContext
context, int index) {
                return Card(
                  child: Padding(
                    padding: const EdgeInsets.all(8.0),
                    child: ListTile(
                    title: Text(contactList![index].name),
                    trailing: Container(
                      width: 100,
                      child: Row(
                        children: [
                          IconButton(
                              onPressed: () async {
                                await
FlutterPhoneDirectCaller.callNumber(contactList![index].number);
                              },
                              icon: Icon(Icons.call_rounded, color: Colors.greenAccent)
                          ),
                          IconButton(
                              onPressed: () {
                                deleteContact(contactList![index]);
                              },
                              icon: Icon(Icons.delete_rounded, color: Colors.redAccent)
                          ),
                        ],
                      ),
                    ),
                  ),
                ),
              );
            }),
          )
        ],
      ),
    ),
  );
  }
}
```

women_safety_app\lib\child\bottom_screens\chat_page.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:women_safety_app/chat_module/chat_screen.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/parent/parent_home_screen.dart';
import 'package:women_safety_app/utils/constants.dart';
class CheckUserStatusBeforeChat extends StatelessWidget {
  const CheckUserStatusBeforeChat({super.key});
  @override
  Widget build(BuildContext context) {
    return StreamBuilder<User?>(
        stream: FirebaseAuth.instance.authStateChanges(),
        builder: (context, snapshot) {
          if (snapshot.connectionState == ConnectionState.waiting) {
            return CircularProgressIndicator();
          }
          else {
            if (snapshot.hasData) {
              print("===>${snapshot.data}");
              return StreamBuilder(
                  stream: FirebaseFirestore.instance.collection("users").where("id",
isEqualTo: FirebaseAuth.instance.currentUser!.uid).snapshots(),
                  builder: (context, snap) {
                    if (snap.connectionState == ConnectionState.waiting) {
                      return CircularProgressIndicator();
                    }
                    else if (snap.hasData) {
                      if (snap.data!.docs.first.data()['type'] == "parent") {
                        return ParentHomeScreen();
                      }
                      else {
                        return ChatPage();
                      }
                    }
                    return SizedBox();
                  }
              );
            }
            else {
              Fluttertoast.showToast(msg: "Please Login First");
              return LoginScreen();
            }
          }
        }
    );
  }
}
```

```
class ChatPage extends StatefulWidget {
  const ChatPage({super.key});
  @override
  State<ChatPage> createState() => _ChatPageState();
}
class _ChatPageState extends State<ChatPage> {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        backgroundColor: primaryColor,
        title: Text("Select Guardian"),
      ),
      body: StreamBuilder(
          stream: FirebaseFirestore.instance.collection('users').where('type',
isEqualTo: 'parent').where('childEmail', isEqualTo:
FirebaseAuth.instance.currentUser!.email).snapshots(),
          builder: (BuildContext context, AsyncSnapshot<QuerySnapshot>
snapshot) {
            if (!snapshot.hasData) {
              return Center(
                  child: progressIndicator(context)
              );
            }
            return ListView.builder(
              itemCount: snapshot.data!.docs.length,
              itemBuilder: (BuildContext context, int index) {
                final d = snapshot.data!.docs[index];
                return Padding(
                  padding: const EdgeInsets.all(8.0),
                  child: Container(
                    color: Color.fromARGB(255, 250, 160, 190),
                    child: ListTile(
                      onTap: () {
                        goTo(context, ChatScreen(
                          currentUserId: FirebaseAuth.instance.currentUser!.uid,
                          friendId: d.id,
                          friendName: d['name'],
                        ));
                      },
                      title: Padding(
                        padding: const EdgeInsets.all(8.0),
                        child: Text(d['name']),
                      ),),
                  ),);
              },
            );
          }),
    );
  }}
```

child/bottom_screens/child_home_page.dart :-

```
import 'dart:math';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:geocoding/geocoding.dart';
import 'package:geolocator/geolocator.dart';
import 'package:permission_handler/permission_handler.dart';
import 'package:women_safety_app/db/db_services.dart';
import 'package:women_safety_app/model/contactsm.dart';
import
'package:women_safety_app/widgets/home_widgets/CustomCarouel.dart';
import
'package:women_safety_app/widgets/home_widgets/custom_appBar.dart';
import 'package:women_safety_app/widgets/home_widgets/emergency.dart';
import
'package:women_safety_app/widgets/home_widgets/safehome/SafeHome.dart';
import 'package:women_safety_app/widgets/live_safe.dart';

class HomeScreen extends StatefulWidget {
  const HomeScreen({super.key});

  @override
  State<HomeScreen> createState() => _HomeScreenState();
}

class _HomeScreenState extends State<HomeScreen> {
  // const HomeScreen({super.key});
  int qIndex = 0;
  Position? _currentPosition;
  String? _currentAddress;
  LocationPermission? permission;

  _getPermission() async => await [Permission.sms].request();
  _isPermissionGranted() async => await Permission.sms.status.isGranted;

  // _sendSMS(String phoneNumber, String message, {int? simSlot}) async {
  //   SmsStatus result = await BackgroundSms.sendMessage(phoneNumber:
phoneNumber, message: message, simSlot: 1);
  //   if (result == SmsStatus.sent) {
  //     Fluttertoast.showToast(msg: "Sent");
  //   }
  //   else {
  //     Fluttertoast.showToast(msg: "Failed");
  //   }
  // }

  String _currentCity = "";
  checkLocationPermission() async {
    bool permissionGranted = await _requestLocationPermission();
    setState(() {
      _locationPermissionGranted = permissionGranted;
    });

    if (_locationPermissionGranted) {
      _getCurrentCity();
    }
  }

  void _getCurrentCity() async {
    try {
      Position position = await Geolocator.getCurrentPosition(desiredAccuracy:
LocationAccuracy.high);
      List<Placemark> placemarks = await
placemarkFromCoordinates(position.latitude, position.longitude);

      if (placemarks.isNotEmpty) {
        Placemark placemark = placemarks[0];
        setState(() {
          _currentCity = placemark.locality ?? "Unknown";
        });
        print(_currentCity);
      }
    } catch (e) {
      print("Error getting current city: $e");
    }
  }

  bool _locationPermissionGranted = false;
  Future<bool> _requestLocationPermission() async {
    var status = await Permission.location.request();
    return status == PermissionStatus.granted;
  }

  void _getCurrentLocation() async {
    try {
      Position position = await Geolocator.getCurrentPosition(desiredAccuracy:
LocationAccuracy.high);
      print("Current Location: $position");
      _getCurrentAddress();
      // Handle the obtained location as needed
    } catch (e) {
      print("Error getting current location: $e");
    }
  }

  String currentCity = "";

  _getCurrentAddress() async {
    try {
      List<Placemark> placemarks = await
placemarkFromCoordinates(_currentPosition!.latitude,
_currentPosition!.longitude);
```

```
      Placemark place = placemarks[0];
      setState(() {
        _currentAddress = "${place.locality}, ${place.street},
${place.postalCode}";
        print(_currentAddress);
      });
    } catch (e) {
      Fluttertoast.showToast(msg: e.toString());
    }
  }

  // _getAddressFromLatLong() async {
  //   try{
  //     List<Placemark> placeMarks = await
placemarkFromCoordinates(_currentPosition!.latitude,
_currentPosition!.longitude);
  //     Placemark place = placeMarks[0];
  //     setState(() {
  //       _currentAddress = "${place.locality}, ${place.street},
${place.postalCode}";
  //     });
  //   }
  //   catch (e) {
  //     Fluttertoast.showToast(msg: e.toString());
  //   }
  // }

  getAndSendSms() async {
    List<TContact> contactList = await DatabaseHelper().getContactList();
    String messageBody =
"https://maps.google.com/?daddr=${_currentPosition!.latitude},${_currentPosit
ion!.longitude}";
    if (await _isPermissionGranted()) {
      contactList.forEach((element) {
        // _sendSMS("${element.number}", "I am in trouble. Please reach me out
at $messageBody");
      });
    }
    else {
      Fluttertoast.showToast(msg: "Something went wrong");
    }
  }

  getRandomQuote() {
    Random random = Random();
    setState(() {
      qIndex = random.nextInt(6);
    });
  }
  @override
  void initState() {
    getRandomQuote();
    super.initState();
    _getPermission();
    // _getCurrentLocation();

    /// Shake Feature ///
    // To close: detector.stopListening();
    // ShakeDetector.waitForStart() waits for user to call
detector.startListening();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      backgroundColor: Colors.white,
      body: SafeArea(
        child: Padding(
          padding: const EdgeInsets.all(8.0),
          child: Column(
            children: [
              SizedBox(
                height: 10,
                child: Container(
                  color: Colors.grey.shade100,
                ),
              ),
              SizedBox(height: 5),
              CustomAppBar(
                  quoteIndex: qIndex,
                  onTap: () {
                    getRandomQuote();
                  }),
              SizedBox(height: 5),
              Expanded(
                child: ListView(
                  shrinkWrap: true,
                  children: [
                    SizedBox(height: 10),
                    Container(
                      decoration: BoxDecoration(
                          color: Colors.grey.shade100,
                          borderRadius: BorderRadius.circular(10)),
                      child: Padding(
                        padding: const EdgeInsets.all(12.0),
                        child: Column(
                          children: [
                            Row(
                              crossAxisAlignment: CrossAxisAlignment.start,
                              children: [
                                CircleAvatar(
                                  child: Icon(Icons.flight_takeoff_outlined),
                                  backgroundColor: Colors.grey.shade300,
                                ),
                                SizedBox(width: 5),
                                Column(
                                  mainAxisAlignment: MainAxisAlignment.start,
                                  crossAxisAlignment: CrossAxisAlignment.start,
                                  children: [
                                    _locationPermissionGranted == false ? Text("Turn on 
location services.", style: TextStyle(fontWeight: FontWeight.bold))
                                    : Text("Location enabled"),
                                    SizedBox(height: 5),
                                    _currentCity.isEmpty ? Text("Please enable locations for 
a better experiences", maxLines: 2, style: TextStyle())
                                    : Text("Current City $_currentCity"),
                                    SizedBox(height: 5),
                                    Align(
                                      alignment: Alignment.centerLeft,
                                      child: _locationPermissionGranted == true ? 
SizedBox()
                                      : MaterialButton(
                                        onPressed: () async {
                                          checkLocationPermission();
                                        },
                                        color: Colors.grey.shade100,
                                        shape: StadiumBorder(),
                                        child: Text("Enable location", style: TextStyle(color: 
Colors.black)),
                                      ),
                                    ),
                                  ],
                                ),
                              ],
                            ),
                          ],
                        ),
                      ),
                    ),
                    Align(
                      alignment: Alignment.center,
                      child: Padding(
                        padding: const EdgeInsets.all(8.0),
                        child: Text(
                          "Explore your power",
                          style: TextStyle(
                              fontSize: 22, fontWeight: FontWeight.bold),
                        ),
                      ),
                    ),
                    SizedBox(height: 10),
                    CustomCarouel(),
                    SizedBox(height: 10),
                    Align(
                      alignment: Alignment.center,
                      child: Padding(
                        padding: const EdgeInsets.all(8.0),
                        child: Text(
                          "Incase of emergency dial me",
                          style: TextStyle(
                              fontSize: 22, fontWeight: FontWeight.bold),
                        ),
                      ),
                    ),
                    Emergency(),
                    SizedBox(height: 10),
                    Align(
                      alignment: Alignment.center,
                      child: Padding(
                        padding: const EdgeInsets.all(8.0),
                        child: Text(
                          "Explore LiveSafe",
                          style: TextStyle(
                              fontSize: 22, fontWeight: FontWeight.bold),
                        ),
                      ),
                    ),
                    SizedBox(height: 10),
                    LiveSafe(),
                    SafeHome(),
                  ],
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\child\bottom_screens\contacts_page.dart :-

```
import 'package:contacts_service/contacts_service.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:permission_handler/permission_handler.dart';
import 'package:women_safety_app/db/db_services.dart';
import 'package:women_safety_app/model/contactsm.dart';
import 'package:women_safety_app/utils/constants.dart';

class ContactsPage extends StatefulWidget {
  const ContactsPage({super.key});

  @override
  State<ContactsPage> createState() => _ContactsPageState();
}

class _ContactsPageState extends State<ContactsPage> {
  List<Contact> contacts = [];
  List<Contact> contactsFiltered = [];
  DatabaseHelper _databaseHelper = DatabaseHelper();

  TextEditingController searchController = TextEditingController();

  @override
  void initState() {
    super.initState();
    askPermissions();
  }

  String flattenPhoneNumber(String phoneStr) {
    return phoneStr.replaceAllMapped(RegExp(r'^(\+)|\D'), (Match m) {
      return m[0] == "+" ? "+" : "";
    });
  }

  filterContacts(){
    List<Contact> _contacts = [];
    _contacts.addAll(contacts);
    if (searchController.text.isNotEmpty) {
      _contacts.retainWhere((element) {
        String searchTerm = searchController.text.toLowerCase();
        String searchTermFlatten = flattenPhoneNumber(searchTerm);
        String contactName = element.displayName ?? "";
        bool nameMatch = contactName.contains(searchTerm);
        if (nameMatch == true) {
          return true;
        }
        if (searchTermFlatten.isEmpty) {
          return false;
        }
        var phone = element.phones!.firstWhere((p) {
          String phoneFlattered = flattenPhoneNumber(p.value!);
          return phoneFlattered.contains(searchTermFlatten);
        });
        return phone.value != null;
      });
    }

    setState(() {
      contactsFiltered = _contacts;
    });
  }

  Future<void> askPermissions() async {
    PermissionStatus permissionStatus = await getContactsPermissions();
    if(permissionStatus == PermissionStatus.granted) {
      getAllContacts();
      searchController.addListener(() {
        filterContacts();
      });
    }
    else {
      handleInvalidPermissions(permissionStatus);
    }
  }

  handleInvalidPermissions(PermissionStatus permissionStatus) {
    if (permissionStatus == PermissionStatus.denied) {
      dialogueBox(context, "Access to the contacts denied by the user");
    }
    else if (permissionStatus == PermissionStatus.permanentlyDenied) {
      dialogueBox(context, "May contact does exist on this device");
    }
  }

  Future<PermissionStatus> getContactsPermissions() async {
    PermissionStatus permission = await Permission.contacts.status;
    if (permission != PermissionStatus.granted && permission !=
PermissionStatus.permanentlyDenied) {
      PermissionStatus permissionStatus = await Permission.contacts.request();
      return permissionStatus;
    }
    else {
      return permission;
    }
  }

  getAllContacts() async {
    List<Contact> _contacts = await
ContactsService.getContacts(withThumbnails: false);
    setState(() {
      contacts = _contacts;
  });
 }

 @override
 Widget build(BuildContext context) {
  bool isSearching = searchController.text.isNotEmpty;
  bool listItemExit = (contactsFiltered.length > 0 || contacts.length > 0);
  return Scaffold(
   appBar: AppBar(title: Text('Contacts'), backgroundColor: primaryColor),
   body: contacts.length == 0 ? Center(child: CircularProgressIndicator())
   : SafeArea(
    child: Column(
     children: [
      Padding(
       padding: const EdgeInsets.all(8.0),
       child: TextField(
        autofocus: true,
        controller: searchController,
        decoration: InputDecoration(
         labelText: "Search Contact",
         prefixIcon: Icon(Icons.search_rounded)
        ),
       ),
      ),
      listItemExit == true
      ? Expanded(
       child: ListView.builder(
         itemCount: isSearching == true ? contactsFiltered.length :
contacts.length,
         itemBuilder: (BuildContext context, int index) {
         Contact contact = isSearching == true ? contactsFiltered[index] :
contacts[index];

         String displayName = contact.displayName!;
         String phoneNumber = contact.phones != null &&
contact.phones!.length > 0 ? contact.phones!.first.value! ?? "No Number" : "No
Number";

         return ListTile(
           title: Text(contact.displayName ?? ""),
           subtitle: Text(phoneNumber),    //
Text(contact.phones!.elementAt(0).value!),
           leading: contact.avatar != null && contact.avatar!.length > 0
            ? CircleAvatar(backgroundColor: primaryColor,
backgroundImage: MemoryImage(contact.avatar!))
            : CircleAvatar(backgroundColor: primaryColor, child:
Text(displayName.length > 0 ? contact.initials() : "NN")),
           onTap: () {
            if (contact.phones!.length > 0) {
             final String phoneNum = contact.phones!.elementAt(0).value!;
             final String name = contact.displayName!;
```

```
                    _addContact(TContact(phoneNum, name));
                  }
                  else {
                    Fluttertoast.showToast(msg: "OOPS! Phone Number of this
contact does not exist");
                  }
              },
            );
            }),
          )
          : Container(
            child: Text("Searching"),
          ),
        ],
      ),
    )
  );
}
void _addContact(TContact newContact) async {
  int result = await _databaseHelper.insertContact(newContact);
  if (result != 0) {
    Fluttertoast.showToast(msg: "Contact added successfully");
  }
  else {
    Fluttertoast.showToast(msg: "Failed to add contacts");
  }
  Navigator.of(context).pop(true);
}
}
```

women_safety_app\lib\child\bottom_screens\profile_page.dart :-

```
import 'dart:io';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:firebase_storage/firebase_storage.dart';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:image_picker/image_picker.dart';
import 'package:uuid/uuid.dart';
import 'package:women_safety_app/child/bottom_page.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/components/PrimaryButton.dart';
import 'package:women_safety_app/components/custom_textfield.dart';
import 'package:women_safety_app/utils/constants.dart';

class CheckUserStatusBeforeChatOnProfile extends StatelessWidget {
  const CheckUserStatusBeforeChatOnProfile({super.key});

  @override
  Widget build(BuildContext context) {
    return StreamBuilder<User?>(
      stream: FirebaseAuth.instance.authStateChanges(),
      builder: (context, snapshot) {
        if (snapshot.connectionState == ConnectionState.waiting) {
          return CircularProgressIndicator();
        }
        else {
          if (snapshot.hasData) {
            return ProfilePage();
          }
          else {
            Fluttertoast.showToast(msg: 'please login first');
            return LoginScreen();
          }
        }
      },
    );
  }
}

class ProfilePage extends StatefulWidget {
  const ProfilePage({super.key});

  @override
  State<ProfilePage> createState() => _ProfilePageState();
}

class _ProfilePageState extends State<ProfilePage> {
  TextEditingController nameC = TextEditingController();
  TextEditingController parentEmailC = TextEditingController();
  TextEditingController childEmailC = TextEditingController();
  TextEditingController phoneC = TextEditingController();

  final key = GlobalKey<FormState>();
  String? id;
  String? profilePic;
  String? downloadURL;
  bool isSaving = false;

  getData() async {
    await FirebaseFirestore.instance.collection('users').where('id', isEqualTo:
FirebaseAuth.instance.currentUser!.uid).get().then((value) {
      setState(() {
        nameC.text = value.docs.first['name'];
        childEmailC.text = value.docs.first['childEmail'];
        parentEmailC.text = value.docs.first['parentEmail'];
        phoneC.text = value.docs.first['phone'];
        id = value.docs.first.id;
        profilePic = value.docs.first['profilePic'];
      });
    });
  }

  @override
  void initState() {
    // TODO: implement initState
    super.initState();
    getData();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(),
      body: isSaving == true ? Center(child:
CircularProgressIndicator(backgroundColor: primaryColor))
      : SafeArea(
        child: Padding(
          padding: const EdgeInsets.all(18.0),
          child: Center(
              child: Form(
                key: key,
                  child: Column(
                    mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                    children: [
                      Text("Update Your Profile", style: TextStyle(fontSize: 25)),
                      SizedBox(height: 15),
                      GestureDetector(
                        onTap: () async {
                          final XFile? pickImage = await
ImagePicker().pickImage(source: ImageSource.gallery, imageQuality: 50);
```

```
                          if (pickImage != null) {
                            setState(() {
                              profilePic = pickImage.path;
                            });
                          }
                        },
                        child: Container(
                          child: profilePic == null ? CircleAvatar(
                              radius: 80, backgroundColor: Colors.deepPurple,
                              child: Center(child: Image.asset('assets/add_pic.png',
height: 80, width: 80))
                          )
                          : profilePic!.contains('http') ? CircleAvatar(
                              radius: 80, backgroundColor: Colors.deepPurple,
                              backgroundImage: NetworkImage(profilePic!),
                          )
                          : CircleAvatar(
                              radius: 80, backgroundColor: Colors.deepPurple,
                              backgroundImage: FileImage(File(profilePic!))
                          ),
                        ),
                      ),
                      CustomTextField(
                          controller: nameC,
                          hintText: nameC.text,
                          validate: (v) {
                            if (v!.isEmpty) {
                              return "Please enter your updated name";
                            }
                            return null;
                          }
                      ),
                      SizedBox(height: 10),
                      CustomTextField(
                          controller: childEmailC,
                          hintText: "Child Email",
                          readOnly: true,
                          validate: (v) {
                            if (v!.isEmpty) {
                              return "Please enter your updated email";
                            }
                            return null;
                          }
                      ),
                      SizedBox(height: 10),
                      CustomTextField(
                          controller: parentEmailC,
                          hintText: "Parent Email",
                          readOnly: true,
                          validate: (v) {
                            if (v!.isEmpty) {
```

```
               return "Please enter your updated parent email";
              }
              return null;
             }
           ),
           SizedBox(height: 10),
           CustomTextField(
             controller: phoneC,
             hintText: "Phone Number",
             readOnly: true,
             validate: (v) {
              if (v!.isEmpty) {
               return "Please enter your updated phone number";
              }
              return null;
             }
           ),
           SizedBox(height: 25),
           PrimaryButton(title: "UPDATE", onPressed: () async {
            if (key.currentState!.validate()) {
             SystemChannels.textInput.invokeMethod('TextInput.hide');
             profilePic == null ? Fluttertoast.showToast(msg: "Please
Select Profile Picture")
             : update();
            }
            // await
FirebaseFirestore.instance.collection('users').doc(id).update({'name':
nameC.text}).then((value) => Fluttertoast.showToast(msg: 'Name updated
successfully'));
           })
        ],
       ))
     ),
    ),
   ),
  );
 }
 Future<String?> uploadImage(String filePath) async {
  try {
   final fileName = Uuid().v4();
   final Reference fbStorage =
FirebaseStorage.instance.ref('profile').child(fileName);
   final UploadTask uploadTask = fbStorage.putFile(File(filePath));
   await uploadTask.then((p0) async {
    downloadURL = await fbStorage.getDownloadURL();
   });
   return downloadURL;
  }
  catch (e) {
   Fluttertoast.showToast(msg: e.toString());
  }
  }

  update() async {
    setState(() {
      isSaving = true;
    });
    uploadImage(profilePic!).then((value) {
      Map<String, dynamic> data = {
        'name': nameC.text,
        'profilePic': downloadURL
      };

FirebaseFirestore.instance.collection('users').doc(FirebaseAuth.instance.currentU
ser!.uid).update(data);
      setState(() {
        isSaving = false;
        goTo(context, BottomPage());
      });
    });
  }
}
```

women_safety_app\lib\parent\parent_register_screen.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:women_safety_app/components/PrimaryButton.dart';
import 'package:women_safety_app/components/SecondaryButton.dart';
import 'package:women_safety_app/components/custom_textfield.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/model/user_model.dart';
import 'package:women_safety_app/utils/constants.dart';

class RegisterParentScreen extends StatefulWidget {
  @override
  State<RegisterParentScreen> createState() => _RegisterParentScreenState();
}

class _RegisterParentScreenState extends State<RegisterParentScreen> {
  // const RegisterParentScreen({super.key});
  bool isPasswordShown = true;
  bool isConfirmPasswordShown = true;

  final _formkey = GlobalKey<FormState>();
  final _formdata = Map<String, Object>();
  bool isLoading = false;

  _onSubmit() async {
    _formkey.currentState!.save();
    if(_formdata['password'] != _formdata['confirm_password']) {
      dialogueBox(context, 'Your entered password and confirm password should
be equal');
    }
    else {
      progressIndicator(context);
      try {
        setState(() {
          isLoading = true;
        });
        UserCredential userCredential = await
FirebaseAuth.instance.createUserWithEmailAndPassword(
            email: _formdata['email'].toString(),
            password: _formdata['password'].toString()
        );
        if(userCredential.user != null) {
          final v = userCredential.user!.uid;
          DocumentReference<Map<String, dynamic>> db =
FirebaseFirestore.instance.collection('users').doc(v);
          final user = UserModel(
              id: v,
              name: _formdata['name'].toString(),
              phone: _formdata['phone'].toString(),
              childEmail: _formdata['child_email'].toString(),
              parentEmail: _formdata['email'].toString(),
              type: 'parent'
          );
          final jsonData = user.toJson();
          await db.set(jsonData).whenComplete(() {
            goTo(context, LoginScreen());
            setState(() {
              isLoading = false;
            });
          });
          // goTo(context, LoginScreen());
        }
      } on FirebaseAuthException catch (e) {
        setState(() {
          isLoading = false;
        });
        if (e.code == 'weak-password') {
          dialogueBox(context, 'The password provided is too weak.');
          print('The password provided is too weak.');
        } else if (e.code == 'email-already-in-use') {
          dialogueBox(context, 'The account already exists for that email.');
          print('The account already exists for that email.');
        }
      } catch (e) {
        setState(() {
          isLoading = false;
        });
        dialogueBox(context, e.toString());
        // print(e);
      }
    }
    print(_formdata['email']);
    print(_formdata['password']);
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: Padding(
          padding: const EdgeInsets.symmetric(horizontal: 10),
          child: Stack(
            children: [
              isLoading ? progressIndicator(context)
              : SingleChildScrollView(
                child: Column(
                  children: [
                    Container(
                      height: MediaQuery.of(context).size.height * 0.3,
                      child: Column(
                        mainAxisAlignment: MainAxisAlignment.spaceEvenly,
              children: [
               Text(
                'REGISTER AS PARENT',
                textAlign: TextAlign.center,
                style: TextStyle(fontSize: 40, color: primaryColor,
  fontWeight: FontWeight.bold),
               ),
               Image.asset('assets/logo.png', height: 100, width: 100),
              ],
             ),
            ),
            Container(
             height: MediaQuery.of(context).size.height * 0.75,
             child: Form(
              key: _formkey,
              child: Column(
               mainAxisAlignment: MainAxisAlignment.spaceEvenly,
               children: [
                CustomTextField(
                 hintText: 'Enter your name',
                 textInputAction: TextInputAction.next,
                 keyboardtype: TextInputType.name,
                 prefix: Icon(Icons.person_rounded),
                 onsave: (name) {
                  _formdata['name'] = name ?? "";
                 },
                 validate: (name) {
                  if (name!.isEmpty || name.length<3) {
                   return 'Please enter correct name';
                  }
                  return null;
                 },
                ),
                CustomTextField(
                 hintText: 'Enter your phone number',
                 textInputAction: TextInputAction.next,
                 keyboardtype: TextInputType.phone,
                 prefix: Icon(Icons.phone_rounded),
                 onsave: (phone) {
                  _formdata['phone'] = phone ?? "";
                 },
                 validate: (phone) {
                  if (phone!.isEmpty || phone.length<10) {
                   return 'Please Enter Correct Phone Number';
                  }
                  return null;
                 },
                ),
                CustomTextField(
                 hintText: 'Enter your email address',
                 textInputAction: TextInputAction.next,
                              keyboardtype: TextInputType.emailAddress,
                              prefix: Icon(Icons.email_rounded),
                              validate: (email) {
                                if (email!.isEmpty || email.length<3 ||
!email.contains("@") || !email.contains(".com")) {
                                  return 'Please Enter Correct Email';
                                }
                                // return null;
                              },
                              onsave: (email) {
                                _formdata['email'] = email ?? "";
                              },
                            ),
                            CustomTextField(
                              hintText: 'Enter your child\'s email address',
                              textInputAction: TextInputAction.next,
                              keyboardtype: TextInputType.emailAddress,
                              prefix: Icon(Icons.email_rounded),
                              validate: (email) {
                                if (email!.isEmpty || email.length<3 ||
!email.contains("@") || !email.contains(".com")) {
                                  return 'Please Enter Correct Email';
                                }
                                // return null;
                              },
                              onsave: (child_email) {
                                _formdata['child_email'] = child_email ?? "";
                              },
                            ),
                            CustomTextField(
                              hintText: 'Enter new password',
                              isPassword: isPasswordShown,
                              prefix: Icon(Icons.vpn_key_rounded),
                              validate: (password) {
                                if (password!.isEmpty || password.length<7) {
                                  return 'Please Enter Correct Password';
                                }
                                return null;
                              },
                              onsave: (password) {
                                _formdata['password'] = password ?? "";
                              },
                              suffix: IconButton(
                                  onPressed: () {
                                    setState(() {
                                      isPasswordShown = !isPasswordShown;
                                    });
                                  },
                                  icon: isPasswordShown ? Icon(Icons.visibility_off) :
Icon(Icons.visibility)
                              ),
                            ),
                            CustomTextField(
                              hintText: 'Enter confirm password',
                              isPassword: isConfirmPasswordShown,
                              prefix: Icon(Icons.vpn_key_rounded),
                              validate: (password) {
                                if (password!.isEmpty || password.length<7) {
                                  return 'Please Enter Correct Password';
                                }
                                return null;
                              },
                              onsave: (password) {
                                _formdata['confirm_password'] = password ?? "";
                              },
                              suffix: IconButton(
                                  onPressed: () {
                                    setState(() {
                                      isConfirmPasswordShown =
!isConfirmPasswordShown;
                                    });
                                  },
                                  icon: isConfirmPasswordShown ?
Icon(Icons.visibility_off) : Icon(Icons.visibility)
                              ),
                            ),
                            PrimaryButton(title: 'REGISTER', onPressed: () {
                              if (_formkey.currentState!.validate()) {
                                _onSubmit();
                              }
                            }),
                          ],
                        ),
                      ),
                    ),
                    SecondaryButton(title: 'Login into your account', onPressed: () {
                      goTo(context, LoginScreen());
                    })
                  ],
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\parent\parent_home_screen.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:women_safety_app/chat_module/chat_screen.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/utils/constants.dart';

class ParentHomeScreen extends StatelessWidget {
 const ParentHomeScreen({super.key});

 @override
 Widget build(BuildContext context) {
  return Scaffold(
   drawer: Drawer(
    child: Column(
     children: [
      DrawerHeader(child: Container()),
      ListTile(
        title: TextButton(
          onPressed: () async {
           try {
            await FirebaseAuth.instance.signOut();
            goTo(context, LoginScreen());
           } on FirebaseAuthException catch (e) {
            dialogueBox(context, e.toString());
           }
          },
          child: Text("SIGN OUT")
        )
      )
     ],
    ),
   ),
   appBar: AppBar(
    backgroundColor: primaryColor,
    title: Text("Select Child"),
   ),
   body: StreamBuilder(
     stream: FirebaseFirestore.instance.collection('users').where('type',
isEqualTo: 'child').where('parentEmail', isEqualTo:
FirebaseAuth.instance.currentUser!.email).snapshots(),
     builder: (BuildContext context, AsyncSnapshot<QuerySnapshot>
snapshot) {
      if (!snapshot.hasData) {
       return Center(
         child: progressIndicator(context)
       );
      }
      return ListView.builder(
       itemCount: snapshot.data!.docs.length,
              itemBuilder: (BuildContext context, int index) {
                final d = snapshot.data!.docs[index];
                return Padding(
                  padding: const EdgeInsets.all(8.0),
                  child: Container(
                    color: Color.fromARGB(255, 250, 160, 190),
                    child: ListTile(
                      onTap: () {
                        goTo(context, ChatScreen(
                          currentUserId: FirebaseAuth.instance.currentUser!.uid,
                          friendId: d.id,
                          friendName: d['name'],
                        ));
                      },
                      title: Padding(
                        padding: const EdgeInsets.all(8.0),
                        child: Text(d['name']),
                      ),
                    ),
                  ),
                );
              },
            );
          }
      ),
    );
  }
}
```

components/custom_textfield.dart :-

```
import 'package:flutter/material.dart';

class CustomTextField extends StatelessWidget {
  // const CustomTextField({super.key});

  final String? hintText;
  final TextEditingController? controller;
  final String? Function(String?)? validate;
  final Function(String?)? onsave;
  final int? maxLines;
  final bool isPassword;
  final bool enable;
  final bool? check;
  final bool readOnly;
  final TextInputType? keyboardtype;
  final TextInputAction? textInputAction;
  final FocusNode? focusNode;
  final Widget? prefix;
  final Widget? suffix;

  CustomTextField({
    this.hintText,
    this.controller,
    this.validate,
    this.onsave,
    this.maxLines,
    this.isPassword = false,
    this.enable = true,
    this.check,
    this.readOnly = false,
    this.keyboardtype,
    this.textInputAction,
    this.focusNode,
    this.prefix,
    this.suffix,
  });

  @override
  Widget build(BuildContext context) {
    return TextFormField(
      enabled: enable == true ? true : enable,
      maxLines: maxLines == null ? 1 : maxLines,
      onSaved: onsave,
      readOnly: readOnly,
      focusNode: focusNode,
      textInputAction: textInputAction,
      keyboardType: keyboardtype == null ? TextInputType.name :
keyboardtype,
      controller: controller,
      validator: validate,
      obscureText: isPassword == false ? false : isPassword,
      decoration: InputDecoration(
        prefixIcon: prefix,
        suffixIcon: suffix,
        labelText: hintText ?? "hint text...",
        focusedBorder: OutlineInputBorder(
          borderRadius: BorderRadius.circular(20),
          borderSide: BorderSide(
            style: BorderStyle.solid,
            color: Theme.of(context).primaryColor,
          ),
        ),
        enabledBorder: OutlineInputBorder(
          borderRadius: BorderRadius.circular(20),
          borderSide: BorderSide(
            style: BorderStyle.solid,
            color: Color(0xFF909A9E),
          ),
        ),
        focusedErrorBorder: OutlineInputBorder(
          borderRadius: BorderRadius.circular(20),
          borderSide: BorderSide(
            style: BorderStyle.solid,
            color: Theme.of(context).primaryColor,
          ),
        ),
        errorBorder: OutlineInputBorder(
          borderRadius: BorderRadius.circular(20),
          borderSide: BorderSide(
            style: BorderStyle.solid,
            color: Colors.red,
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\components\fab_bar_bottom.dart :-

```
import 'package:flutter/material.dart';
import 'package:women_safety_app/utils/constants.dart';

class FABBottomAppBarItem {
  FABBottomAppBarItem({required this.iconData, required this.text});
  final IconData iconData;
  final String text;
}

class FABBottomAppBar extends StatefulWidget {
  FABBottomAppBar({
    required this.items,
    this.centerItemText,
    this.height = 60.0,
    this.iconSize = 24.0,
    this.backgroundColor,
    this.color,
    this.selectedColor,
    this.notchedShape,
    required this.onTabSelected,
  }) {
//    assert(this.items.length == 2 || this.items.length == 4);
  }
  final List<FABBottomAppBarItem> items;
  final String? centerItemText;
  final double height;
  final double iconSize;
  final Color? backgroundColor;
  final Color? color;
  final Color? selectedColor;
  final NotchedShape? notchedShape;
  final ValueChanged<int> onTabSelected;

  @override
  _FABBottomAppBarState createState() => _FABBottomAppBarState();
}

class _FABBottomAppBarState extends State<FABBottomAppBar> {
  int _selectedIndex = 0;

  void _updateIndex(int index) {
    widget.onTabSelected(index);
    setState(() {
      _selectedIndex = index;
    });
  }

  @override
  Widget build(BuildContext context) {
    Widget _buildTabItem({
```

```
  required FABBottomAppBarItem item,
  required int index,
  required ValueChanged<int> onPressed,
}) {
  Color color = _selectedIndex == index ? Color(0xFFF56A98) : Colors.black;
  return Expanded(
    child: SizedBox(
      height: widget.height,
      child: Material(
        type: MaterialType.transparency,
        child: InkWell(
          onTap: () => onPressed(index),
          child: Column(
            mainAxisSize: MainAxisSize.min,
            mainAxisAlignment: MainAxisAlignment.center,
            children: <Widget>[
              Icon(item.iconData, color: color, size: widget.iconSize),
              SizedBox(height: 5.0),
              Text(item.text, style: TextStyle(color: color, fontSize: 12))
            ],
          ),
        ),
      ),
    ),
  );
}

List<Widget> items = List.generate(widget.items.length, (int index) {
  return _buildTabItem(
    item: widget.items[index],
    index: index,
    onPressed: _updateIndex,
  );
});

return Container(
  decoration: BoxDecoration(
    // borderRadius: BorderRadius.circular(10),
    // color: Colors.white,
    boxShadow: [
      BoxShadow(
          color: Colors.white.withOpacity(.5),
          spreadRadius: 2,
          blurRadius: 0
      ),
      BoxShadow(
          color: Colors.white.withOpacity(.5),
          spreadRadius: 0,
          blurRadius: 20
      ),
    ],
      ),
      child: BottomAppBar(
        elevation: 5,
        color: Colors.white54,
        surfaceTintColor: primaryColor,
        child: Row(
          mainAxisSize: MainAxisSize.max,
          mainAxisAlignment: MainAxisAlignment.spaceAround,
          children: items,
        ),
      ),
    );
  }
}
```

women_safety_app\lib\components\PrimaryButton.dart :-

```
import 'package:flutter/material.dart';
import 'package:women_safety_app/utils/constants.dart';

class PrimaryButton extends StatelessWidget {
  // const PrimaryButton({super.key});

  final String title;
  final Function onPressed;
  bool loading;
  PrimaryButton({required this.title, required this.onPressed, this.loading =
false});

  @override
  Widget build(BuildContext context) {
    return Container(
      height: 50,
      width: double.infinity,
      child: ElevatedButton(
        onPressed: () {
          onPressed();
        },
        child: Text(title, style: TextStyle(fontSize: 18)),
        style: ElevatedButton.styleFrom(
          backgroundColor: primaryColor,
          shape: RoundedRectangleBorder(
            borderRadius: BorderRadius.circular(30),
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\components\SecondaryButton.dart :-

```
import 'package:flutter/material.dart';

class SecondaryButton extends StatelessWidget {
  final String title;
  final Function onPressed;
  const SecondaryButton({super.key, required this.title, required
this.onPressed});

  @override
  Widget build(BuildContext context) {
    return Container(
      child: TextButton(
        onPressed: () {
          onPressed();
        },
        child: Text(title, style: TextStyle(fontSize: 18)),
      ),
    );
  }
}
```

women_safety_app\lib\model\contactsm.dart :-

```
class TContact {
  int? _id;
  String? _number;
  String? _name;

  TContact(this._number, this._name);
  TContact.withId(this._id, this._number, this._name);

  //getters
  int get id => _id!;
  String get number => _number!;
  String get name => _name!;

  @override
  String toString(){
    return "Contact: {id: $_id, name: $_name, number: $_number}";
  }

  //setters
  set number(String newNumber) => this._number = newNumber;
  set name(String newName) => this._name = newName;

  //Convert a Contact object to a Map object...
  Map<String, dynamic> toMap() {
    var map = new Map<String, dynamic>();

    map['id'] = this._id;
    map['number'] = this._number;
    map['name'] = this._name;

    return map;
  }

  //Extract Contact Object from a Map object
  TContact.fromMapObject(Map<String, dynamic> map) {
    this._id = map['id'];
    this._number = map['number'];
    this._name = map['name'];
  }
}
```

model/user_model.dart :-

```
class UserModel {
  String? id;
  String? name;
  String? phone;
  String? childEmail;
  String? parentEmail;
  String? type;
  String? profilePic;

  UserModel({this.id, this.name, this.phone, this.childEmail, this.parentEmail,
this.type, this.profilePic});

  Map<String, dynamic> toJson() => {
    'id': id,
    'name': name,
    'phone': phone,
    'childEmail': childEmail,
    'parentEmail': parentEmail,
    'type': type,
    'profilePic': profilePic
  };
}
```

db/db_services.dart :-

```
import 'package:sqflite/sqflite.dart';
import 'package:women_safety_app/model/contactsm.dart';

class DatabaseHelper {
  String contactTable = 'contact_table';
  String colId = 'id';
  String colContactName = 'name';
  String colContactNumber = 'number';

  // Private Constructor (used to create an instance of a singleton class)
  // It will be used to create an instance of the DatabaseHelper class
  DatabaseHelper._createInstance();

  // This will be referenced using 'this' keyword... It helps us to access getters
and setters of the class
  static DatabaseHelper? _databaseHelper;

  // 'factory' keyword allows the constructor to return some value
  factory DatabaseHelper() {
    if (_databaseHelper == null) {
      _databaseHelper = DatabaseHelper._createInstance();
    }
    return _databaseHelper!;
  }

  // Initializing the database...
  static Database? _database;
  Future<Database> get database async {
    if (_database == null) {
      _database = await initializeDatabase();
    }
    return _database!;
  }

  Future<Database> initializeDatabase() async {
    String directoryPath = await getDatabasesPath();
    String dbLocation = directoryPath + 'contact.db';

    var contactDatabase = await openDatabase(dbLocation, version: 1, onCreate:
_createDbTable);
    return contactDatabase;
  }

  void _createDbTable(Database db, int newVersion) async {
    await db.execute('CREATE TABLE $contactTable($colId INTEGER PRIMARY
KEY AUTOINCREMENT, $colContactName TEXT, $colContactNumber TEXT)');
  }

  // Fetching operation: get contact object from db
  Future<List<Map<String, dynamic>>> getContactMapList() async {
    Database db = await this.database;
    List<Map<String, dynamic>> result = await db.rawQuery('SELECT * FROM
$contactTable order by $colId ASC');
    // OR
    // var result = await db.query(contactTable, orderBy: '$colId ASC');
    return result;
  }

  // Inserting a contact object
  Future<int> insertContact(TContact contact) async {
    Database db = await this.database;
    var result = await db.insert(contactTable, contact.toMap());
    // print(await db.query(contactTable));
    return result;
  }

  // Updating a contact object
  Future<int> updateContact(TContact contact) async {
    Database db = await this.database;
    var result = await db.update(contactTable, contact.toMap(), where: '$colId =
?', whereArgs: [contact.id]);
    // print(await db.query(contactTable));
    return result;
  }

  // Deleting a contact object
  Future<int> deleteContact(int id) async {
    Database db = await this.database;
    int result = await db.rawDelete('DELETE FROM $contactTable WHERE $colId =
$id');
    // print(await db.query(contactTable));
    return result;
  }

  // Counting the total contact objects
  Future<int> getCount() async {
    Database db = await this.database;
    List<Map<String, dynamic>> x = await db.rawQuery('SELECT COUNT (*)
FROM $contactTable');
    int result = Sqflite.firstIntValue(x)!;
    return result;
  }

  Future<List<TContact>> getContactList() async {
    var contactMapList = await getContactMapList();
    int count = contactMapList.length;

    List<TContact> contactList = <TContact>[];

    for(int i=0; i< count; i++){
      contactList.add(TContact.fromMapObject(contactMapList[i]));
    }
    return contactList;
  }
}
```

women_safety_app\lib\db\share_pref.dart :-

```
import 'package:shared_preferences/shared_preferences.dart';

class MySharedPreference {
  static SharedPreferences? _preferences;
  static const String key = 'usertype';

  static init() async {
    _preferences = await SharedPreferences.getInstance();
    return _preferences;
  }

  static Future saveUserType(String type) async {
    return await _preferences!.setString(key, type);
  }

  static Future<String>? getUserType() async => await
_preferences!.getString(key) ?? "";
}
```

women_safety_app\lib\chat_module\chat_screen.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:women_safety_app/chat_module/message_text_field.dart';
import 'package:women_safety_app/chat_module/singleMessage.dart';
import 'package:women_safety_app/utils/constants.dart';

class ChatScreen extends StatefulWidget {
  final String currentUserId;
  final String friendId;
  final String friendName;
  const ChatScreen({super.key, required this.currentUserId, required
this.friendId, required this.friendName});

  @override
  State<ChatScreen> createState() => _ChatScreenState();
}

class _ChatScreenState extends State<ChatScreen> {
  String? type;
  String? myName;
  getStatus() async {
    await
FirebaseFirestore.instance.collection('users').doc(widget.currentUserId).get().the
n((value) {
      setState(() {
        type = value.data()!['type'];
        myName = value.data()!['name'];
      });
    });
  }

  @override
  void initState() {
    getStatus();
    super.initState();
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text(widget.friendName),
        backgroundColor: primaryColor,
      ),
      body: Column(
        children: [
          Expanded(
            child: StreamBuilder(
                stream:
FirebaseFirestore.instance.collection('users').doc(widget.currentUserId).collectio
n('messages').doc(widget.friendId).collection('chats').orderBy('date',
descending: false).snapshots(),
                builder: (BuildContext context, AsyncSnapshot<QuerySnapshot>
snapshot) {
                    if (snapshot.hasData) {
                      if (snapshot.data!.docs.length < 1) {
                        return Center(
                          child: Text(type == 'parent' ? "Talk with Users" : "Talk with
Parent", style: TextStyle(fontSize: 30)),
                        );
                      }
                      return Container(
                        child: ListView.builder(
                          itemCount: snapshot.data!.docs.length,
                          itemBuilder: (BuildContext context, int index) {
                            bool isMe = snapshot.data!.docs[index]['senderId'] ==
widget.currentUserId;
                            final data = snapshot.data!.docs[index];
                            // Something has to be newly added...
                            return Dismissible(
                              key: UniqueKey(),
                              onDismissed: (direction) async {
                                await
FirebaseFirestore.instance.collection('users').doc(widget.currentUserId).collectio
n('messages').doc(widget.friendId).collection('chats').doc(data.id).delete();
                                await
FirebaseFirestore.instance.collection('users').doc(widget.friendId).collection('mes
sages').doc(widget.currentUserId).collection('chats').doc(data.id).delete().then((
value) => Fluttertoast.showToast(msg: 'Messages deleted successfully'));
```

```
                              },
                              child: SingleMessage(
                                message: data['message'],
                                isMe: isMe,
                                type: data['type'],
                                myName: myName,
                                friendName: widget.friendName,
                                date: data['date'],
                              ),
                            );
                          },
                        ),
                      );
                    }
                    return progressIndicator(context);
                }
            ),
          ),
          MessageTextField(
            currentId: widget.currentUserId,
            friendId: widget.friendId,
          ),
        ],
      ),
    );
  }
}
```

women_safety_app\lib\chat_module\message_text_field.dart :-

```
import 'dart:io';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_storage/firebase_storage.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:geocoding/geocoding.dart';
import 'package:geolocator/geolocator.dart';
import 'package:image_picker/image_picker.dart';
import 'package:uuid/uuid.dart';
import 'package:women_safety_app/utils/constants.dart';
class MessageTextField extends StatefulWidget {
  final String currentId;
  final String friendId;
  const MessageTextField({super.key, required this.currentId, required
this.friendId});
  @override
  State<MessageTextField> createState() => _MessageTextFieldState();
}
class _MessageTextFieldState extends State<MessageTextField> {
  TextEditingController _controller = TextEditingController();
  Position? _currentPosition;
  String? _currentAddress;
  String? message;
  File? imageFile;
  LocationPermission? permission;
  Future getImage() async {
    ImagePicker _picker = ImagePicker();
    await _picker.pickImage(source: ImageSource.gallery).then((XFile? xFile) {
      if (xFile != null) {
        imageFile = File(xFile.path);
        uploadImage();
      }
    });
  }
  Future getImageFromCamera() async {
    ImagePicker _picker = ImagePicker();
    await _picker.pickImage(source: ImageSource.camera).then((XFile? xFile) {
      if (xFile != null) {
        imageFile = File(xFile.path);
        uploadImage();
      }
    });
  }
  Future uploadImage() async {
    String fileName = Uuid().v1();
    int status = 1;
    var ref =
FirebaseStorage.instance.ref().child('images').child("$fileName.jpg");
```

```
    var uploadTask = await ref.putFile(imageFile!);
    if (status == 1) {
      String imageUrl = await uploadTask.ref.getDownloadURL();
      await sendMessage(imageUrl, 'img');
    }
  }
  Future _getCurrentLocation() async {
    permission = await Geolocator.checkPermission();
    if (permission == LocationPermission.denied) {
      permission = await Geolocator.requestPermission();
      Fluttertoast.showToast(msg: "Location Permissions are denied");
      if(permission == LocationPermission.deniedForever){
        Fluttertoast.showToast(msg: "Location Permissions are permanently
denied");
      }
    }
    Geolocator.getCurrentPosition(
        desiredAccuracy: LocationAccuracy.high,
        forceAndroidLocationManager: true
    ).then((Position position) {
      setState(() {
        _currentPosition = position;
        print(_currentPosition!.latitude);
        _getAddressFromLatLong();
      });
    }).catchError((e) {
      Fluttertoast.showToast(msg: e.toString());
    });
  }
  _getAddressFromLatLong() async {
    try{
      List<Placemark> placeMarks = await
placemarkFromCoordinates(_currentPosition!.latitude,
_currentPosition!.longitude);
      Placemark place = placeMarks[0];
      setState(() {
        _currentAddress = "${place.locality}, ${place.street},
${place.postalCode}";
      });
    }
    catch (e) {
      Fluttertoast.showToast(msg: e.toString());
    }
  }
  sendMessage(String message, String type) async {
    await
FirebaseFirestore.instance.collection('users').doc(widget.currentId).collection('m
essages').doc(widget.friendId).collection('chats').add({
      'senderId': widget.currentId,
      'receiverId': widget.friendId,
      'message': message,
    'type': type,
    'date': DateTime.now()
   });
   await
FirebaseFirestore.instance.collection('users').doc(widget.friendId).collection('mes
sages').doc(widget.currentId).collection('chats').add({
    'senderId': widget.currentId,
    'receiverId': widget.friendId,
    'message': message,
    'type': type,
    'date': DateTime.now()
   });
  }
  @override
  Widget build(BuildContext context) {
   return Padding(
    padding: const EdgeInsets.all(8.0),
    child: Container(
     padding: EdgeInsets.all(0),
     decoration: BoxDecoration(color: Colors.white, borderRadius:
BorderRadius.circular(10)),
     child: Row(
      children: [
       Expanded(
        child: TextField(
         cursorColor: primaryColor,
         controller: _controller,
         decoration: InputDecoration(
          hintText: 'Type your message here',
          fillColor: Colors.grey[100],
          filled: true,
          prefixIcon: IconButton(
            onPressed: () {
             showModalBottomSheet(
              backgroundColor: Colors.transparent,
               context: context,
               builder: (context) => bottomsheet());
            },
            icon: Icon(Icons.add_box_rounded, color: primaryColor))
         ),
        ),
       ),
       Padding(
        padding: const EdgeInsets.all(8.0),
        child: InkWell(
          onTap: () async {
           message = _controller.text;
           sendMessage(message!, 'text');
           _controller.clear();
          },
          child: Icon(Icons.send_rounded, color: primaryColor, size: 30)),
            )
          ],
        ),
      ),
    );
  }
  bottomsheet() {
    return Container(
      height: MediaQuery.of(context).size.height * 0.2,
      width: double.infinity,
      child: Card(
        margin: EdgeInsets.all(18),
        shape: RoundedRectangleBorder(borderRadius: BorderRadius.circular(15)),
        child: Row(
          mainAxisAlignment: MainAxisAlignment.spaceEvenly,
          children: [
            chatIcons(Icons.location_pin, "Location", () async {
              await _getCurrentLocation();
              Future.delayed(Duration(seconds: 2), (){
                message =
"https://www.google.com/maps/search/?api=1&query=${_currentPosition!.latit
ude}%2C${_currentPosition!.longitude}. $_currentAddress";
                sendMessage(message!, 'link');
              });
            }),
            chatIcons(Icons.camera_alt_rounded, "Camera", () async {
              await getImageFromCamera();
            }),
            chatIcons(Icons.insert_photo_rounded, "Gallery", () async {
              await getImage();
            })
          ],
        ),
      ),
    );
  }
  chatIcons(IconData icons, String title, VoidCallback onTap) {
    return InkWell(
      onTap: onTap,
      child: Column(
        mainAxisAlignment: MainAxisAlignment.center,
        children: [
          CircleAvatar(
            radius: 30,
            backgroundColor: primaryColor,
            child: Icon(icons),
          ),
          Text("$title")
        ],),);
  }
}
```

women_safety_app\lib\chat_module\singleMessage.dart :-

```
import 'package:cached_network_image/cached_network_image.dart';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import 'package:url_launcher/url_launcher.dart';
import 'package:women_safety_app/utils/constants.dart';

class SingleMessage extends StatelessWidget {
  final String? message;
  final bool? isMe;
  final String? image;
  final String? type;
  final String? myName;
  final String? friendName;
  final Timestamp? date;
  const SingleMessage({super.key, this.message, this.isMe, this.image,
this.type, this.myName, this.friendName, this.date});

  @override
  Widget build(BuildContext context) {
    final size = MediaQuery.of(context).size;
    DateTime d = DateTime.parse(date!.toDate().toString());
    String chat_date = "${d.hour}" + ":" + "${d.minute}";
    return type == "text"
    ? Container(
      constraints: BoxConstraints(
        maxWidth: size.width / 2
      ),
      alignment: isMe! ? Alignment.centerRight : Alignment.centerLeft,
      padding: EdgeInsets.all(10),
      child: Container(
        decoration: BoxDecoration(
            color: isMe! ? primaryColor : Colors.black,
            borderRadius: isMe!
                ? BorderRadius.only(topLeft: Radius.circular(15), topRight:
Radius.circular(15), bottomLeft: Radius.circular(15))
                : BorderRadius.only(topLeft: Radius.circular(15), topRight:
Radius.circular(15), bottomRight: Radius.circular(15))
        ),
        padding: EdgeInsets.all(10),
        constraints: BoxConstraints(maxWidth: size.width / 2),
        alignment: isMe! ? Alignment.centerRight : Alignment.centerLeft,
        child: Column(
          children: [
            Align(
                alignment: Alignment.centerRight,
                child: Text(isMe! ? myName! : friendName!, style:
TextStyle(fontSize: 15, color: Colors.white70))),
            Divider(),
            Align(
              alignment: Alignment.centerRight,
        child: Text(message!, style: TextStyle(fontSize: 18, color:
Colors.white))),
      Divider(),
      Align(
        alignment: Alignment.centerRight,
        child: Text("$chat_date", style: TextStyle(fontSize: 15, color:
Colors.white70))),
     ],
    ),
   ),
  )
  : type == 'img'
  ? Container(
   height: size.height / 2.5,
   width: size.width,
   // constraints: BoxConstraints(
   //     maxWidth: size.width / 2
   // ),
   alignment: isMe! ? Alignment.centerRight : Alignment.centerLeft,
   padding: EdgeInsets.all(10),
   child: Container(
    height: size.height / 2.5,
    width: size.width,
    decoration: BoxDecoration(
      color: isMe! ? primaryColor : Colors.black,
      borderRadius: isMe!
        ? BorderRadius.only(topLeft: Radius.circular(15), topRight:
Radius.circular(15), bottomLeft: Radius.circular(15))
        : BorderRadius.only(topLeft: Radius.circular(15), topRight:
Radius.circular(15), bottomRight: Radius.circular(15))
    ),
    // padding: EdgeInsets.all(10),
    constraints: BoxConstraints(maxWidth: size.width / 2),
    alignment: isMe! ? Alignment.centerRight : Alignment.centerLeft,
    child: Column(
     children: [
      Align(
        alignment: Alignment.centerRight,
        child: Text(isMe! ? myName! : friendName!, style:
TextStyle(fontSize: 15, color: Colors.white70))),
      Divider(),
      CachedNetworkImage(
        imageUrl: message!, fit: BoxFit.cover, height: size.height / 3.62,
width: size.width,
        placeholder: (context, url) => CircularProgressIndicator(),
        errorWidget: (context, url, error) => Icon(Icons.error_rounded)),
      Divider(),
      Align(
        alignment: Alignment.centerRight,
        child: Text("$chat_date", style: TextStyle(fontSize: 15, color:
Colors.white70))),
     ],
    ),
   ),
  )
  : Container(
   constraints: BoxConstraints(
     maxWidth: size.width / 2
   ),
   alignment: isMe! ? Alignment.centerRight : Alignment.centerLeft,
   padding: EdgeInsets.all(10),
   child: Container(
    decoration: BoxDecoration(
      color: isMe! ? primaryColor : Colors.black,
      borderRadius: isMe!
        ? BorderRadius.only(topLeft: Radius.circular(15), topRight:
Radius.circular(15), bottomLeft: Radius.circular(15))
        : BorderRadius.only(topLeft: Radius.circular(15), topRight:
Radius.circular(15), bottomRight: Radius.circular(15))
    ),
    padding: EdgeInsets.all(10),
    constraints: BoxConstraints(maxWidth: size.width / 2),
    alignment: isMe! ? Alignment.centerRight : Alignment.centerLeft,
    child: Column(
     children: [
      Align(
        alignment: Alignment.centerRight,
        child: Text(isMe! ? myName! : friendName!, style:
TextStyle(fontSize: 15, color: Colors.white70))),
      Divider(),
      Align(
        alignment: Alignment.centerRight,
        child: GestureDetector(
         onTap: () async {
          await launchUrl(Uri.parse("$message"));
         },
          child: Text(message!, style: TextStyle(fontStyle: FontStyle.italic,
fontSize: 16, color: Colors.white))
        )
      ),
      Divider(),
      Align(
        alignment: Alignment.centerRight,
        child: Text("$chat_date", style: TextStyle(fontSize: 15, color:
Colors.white70))),
     ],
    ),
   ),
  );
 }
}
```

women_safety_app\lib\profile_mode\personal_info_page.dart :-

```
import 'dart:io';
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:flutter/material.dart';
import 'package:image_picker/image_picker.dart';

class UserProfile {
  final String uid;
  final String name;
  final String phone;
  final String childEmail;
  final String parentEmail;
  final String profilePicture;

  UserProfile({required this.uid, required this.name, required this.phone,
required this.childEmail, required this.parentEmail, required this.profilePicture});
}

class PersonalInfoPage extends StatefulWidget {
  const PersonalInfoPage({super.key});

  @override
  State<PersonalInfoPage> createState() => _PersonalInfoPageState();
}

class _PersonalInfoPageState extends State<PersonalInfoPage> {
  String? profilePic;
  String? downloadUrl;
  bool isSaving = false;

  Future<UserProfile> getUserProfile(String uid) async {
    DocumentSnapshot<Map<String, dynamic>> snapshot = await
FirebaseFirestore.instance.collection('users').doc(uid).get();

    profilePic = snapshot['profilePic'];

    return UserProfile(
      uid: uid,
      name: snapshot['name'],
      childEmail: snapshot['childEmail'],
      parentEmail: snapshot['parentEmail'],
      phone: snapshot['phone'],
      profilePicture: snapshot['profilePic'] ?? "",
    );
  }

  @override
  Widget build(BuildContext context) {
    return Scaffold(
      body: SafeArea(
        child: FutureBuilder<UserProfile>(
          future: getUserProfile('bh10uyfaxYWU0yUiIbaVfqk9Iq1'), // Replace
'user_id' with the actual user ID
          builder: (context, snapshot) {
            if (snapshot.connectionState == ConnectionState.waiting) {
              return Center(child: CircularProgressIndicator());
            }
            else if (snapshot.hasError) {
              return Center(child: Text('Error: ${snapshot.error}'));
            }
            else {
              return PersonalInfoPageUi(userProfile: snapshot.data!);
            }
          },
        ),
      ),
    );
  }

  Widget PersonalInfoPageUi({required UserProfile userProfile}) {
    return Padding(
        padding: const EdgeInsets.all(16.0),
        child: Column(
          crossAxisAlignment: CrossAxisAlignment.start,
          children: [
            GestureDetector(
              onTap: () async {
                final XFile? pickImage = await ImagePicker().pickImage(source:
ImageSource.gallery, imageQuality: 50);
                if (pickImage != null) {
                  setState(() {
                    profilePic = pickImage.path;
                  });
                }
              },
              child: Container(
                child: profilePic == null
                ? Center(
                  child: CircleAvatar(
                    backgroundColor: Colors.deepPurple,
                    radius: 80,
                    child: Center(
                        child: Image.asset('assets/add_pic.png', height: 80, width: 80,
fit: BoxFit.cover)
                    ),
                  ),
                )
                : profilePic!.contains('http')
                ? Center(
                  child: ClipOval(
                    child: Container(
                      width: 160.0, // 2 times the radius
```

```
           height: 160.0, // 2 times the radius
           color: Colors.deepPurple,
           child: CircleAvatar(
            radius: 80,
            backgroundImage: NetworkImage(profilePic!),
           ),
          ),
         ),
        )
        : Center(
         child: CircleAvatar(
           backgroundColor: Colors.deepPurple,
           radius: 80,
           backgroundImage: FileImage(File(profilePic!))),
        ),
       ),
      ),
      SizedBox(height: 16),
      Center(
       child: Text(
        '${userProfile.name}',
        style: TextStyle(fontSize: 20),
       ),
      ),
      SizedBox(height: 8),
      Divider(),
      SizedBox(height: 8),
      Text(
       'Child Email: ${userProfile.childEmail}',
       style: TextStyle(fontSize: 16),
      ),
      SizedBox(height: 8),
      Text(
       'Parent Email: ${userProfile.parentEmail}',
       style: TextStyle(fontSize: 16),
      ),
      SizedBox(height: 8),
      Text(
       'Phone: ${userProfile.phone}',
       style: TextStyle(fontSize: 16),
      ),
     ],
    )
  );
 }
}
```

women_safety_app\lib\profile_mode\settings.dart :-

```
import 'package:cloud_firestore/cloud_firestore.dart';
import 'package:firebase_auth/firebase_auth.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:women_safety_app/child/bottom_page.dart';
import 'package:women_safety_app/child/child_login_screen.dart';
import 'package:women_safety_app/components/PrimaryButton.dart';
import 'package:women_safety_app/child/bottom_screens/profile_page.dart';
import 'package:women_safety_app/utils/constants.dart';

class ProfileItem {
  final String? item;
  final String? title;

  ProfileItem({this.item, this.title});
}

class SettingsPage extends StatefulWidget {
  const SettingsPage({super.key});

  @override
  State<SettingsPage> createState() => _SettingsPageState();
}

class _SettingsPageState extends State<SettingsPage> {
  List<ProfileItem> _infoItems = [
    ProfileItem(item: 'personal', title: 'Personal Information'),
    ProfileItem(item: 'settings', title: 'Settings'),
    // ProfileItem(item: 'admin', title: 'Admin Login'),
    ProfileItem(
        item: displayName == null ? 'login' : "logout",
        title: displayName == null ? 'login' : "logout"),
    // ProfileItem(item: 'logout', title: 'logout'),
  ];
  static String? displayName;
  Future<String?> getName() async {
    try {
      FirebaseAuth auth = await FirebaseAuth.instance;
      var snapshot = await
FirebaseFirestore.instance.collection("users").doc(auth.currentUser!.uid).get();
      displayName = snapshot['name'];
      return displayName;
    }
    catch (e) {
      print('Error getting display name: $e');
      return null;
    }
  }

  @override
```

```
void initState() {
  // getName();
  super.initState();
}

void infoButtonOnTap(ProfileItem item) {
  switch (item.item) {
    case 'personal':
      if (displayName == null) {
        Fluttertoast.showToast(msg: "Please login first to see information");
      } else {
        goTo(context, ProfilePage());
      }
      // Navigator.pushNamed(context, PersonalInfoPage.id);
      break;
    case 'settings':
      // Navigator.pushNamed(context, VehicleInfoPage.id);
      break;
    // transfer amount
    // case 'admin':
    // // Navigator.pushNamed(context, VehicleInfoPage.id);
    //   break;
    case 'login':
      goTo(context, CheckUserStatusBeforeChatOnProfile());
      break;
    case 'about':
      // _launchUrl();
      break;
    case 'logout':
      logout();
      break;

    default: break;
  }
}

logout() async {
  showDialog(
    context: context,
    builder: (context) {
      return AlertDialog(
        title: Text("Are you sure you want to Logout?"),
        actions: [
          PrimaryButton(
              title: "Confirm",
              onPressed: () async {
                try {
                  await FirebaseAuth.instance.signOut();
                  setState(() {
                    displayName == null;
                  });
                  goTo(context, LoginScreen());
                } on FirebaseAuthException catch (e) {
                  dialogueBox(context, e.toString());
                }
              }),
          Center(
            child: TextButton(
                child: Text("Cancel"),
                onPressed: () {
                  Navigator.pop(context);
                }),
          ),
        ],
      );
    },
  );
}

@override
Widget build(BuildContext context) {
  return SafeArea(
    child: Scaffold(
      body: Padding(
        padding: const EdgeInsets.all(12.0),
        child: Column(
          children: [
            SizedBox(height: 30),
            CircleAvatar(radius: 80),
            SizedBox(height: 30),
            FutureBuilder(
              future: getName(),
              builder: (BuildContext context, AsyncSnapshot snapshot) {
                _infoItems;
                if (snapshot.connectionState == ConnectionState.waiting) {
                  return Center(child: CircularProgressIndicator());
                }
                else if (snapshot.hasError) {
                  return Center(child: Text('Error: ${snapshot.error}'));
                }
                else {
                  return displayName == null ? Text("Data") : Text(displayName!);
                }
              },
            ),
            // displayName == null ? Text("Data") : Text(displayName!),
            SizedBox(height: 30),
            Flexible(
              child: ListView.separated(
                itemBuilder: (BuildContext context, int index) {
                  return GestureDetector(
                    onTap: () {
                        infoButtonOnTap(_infoItems[index]);
                      },
                      child: Container(
                        decoration: BoxDecoration(color: Colors.pinkAccent,
borderRadius: BorderRadius.circular(10)), padding: EdgeInsets.all(14.0),
                        child: Row(
                          mainAxisAlignment: MainAxisAlignment.spaceBetween,
                          children: <Widget>[
                            Text(_infoItems[index].title!, style: TextStyle(fontSize: 18,
color: Colors.white)),
                            Icon(Icons.navigate_next, size: 30, color: Colors.white)
                          ],
                        ),
                      ),
                    );
                  },
                  separatorBuilder: (context, index) {
                    return SizedBox(height: 10.0);
                  },
                  itemCount: _infoItems.length,
                  // shrinkWrap: true,
                ),
              ),
            ],
          ),
        ),
      ),
    );
  }

  Widget title() {
    return Padding(
      padding: const EdgeInsets.all(8.0),
      child: ListTile(
        shape: StadiumBorder(),
        tileColor: Colors.pinkAccent,
        title: Text("data"),
        trailing: Icon(Icons.navigate_next_outlined),
      ),
    );
  }
}
```

women_safety_app\lib\utils\constants.dart :-

```
import 'package:flutter/material.dart';

Color primaryColor = Color(0xFFFc3B77);

void goTo(BuildContext context, Widget nextScreen) {
  Navigator.push(context, MaterialPageRoute(builder: (context) =>
nextScreen));
}

dialogueBox(BuildContext context, String text) {
  showDialog(context: context, builder: (context) => AlertDialog(title:
Text(text)));
}

Widget progressIndicator(BuildContext context) {
  return Center(
    child: CircularProgressIndicator(
      color: Color(0xFFCC2029),
      backgroundColor: Color(0xFF531BF3),
      strokeWidth: 7,
    ),
  );
}
```

women_safety_app\lib\utils\flutter_background_services.dart :-

```
import 'dart:async';
import 'dart:ui';
import 'package:background_location/background_location.dart';
import 'package:flutter_background_service/flutter_background_service.dart';
import 'package:flutter_local_notifications/flutter_local_notifications.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:shake/shake.dart';
import 'package:telephony/telephony.dart';
import 'package:vibration/vibration.dart';
import 'package:women_safety_app/db/db_services.dart';
import 'package:women_safety_app/model/contactsm.dart';

sendMessage(String messageBody) async {
  List<TContact> contactList = await DatabaseHelper().getContactList();
  if (contactList.isEmpty) {
    Fluttertoast.showToast(msg: "No numbers exist. Please add a Number first");
  }
  else {
    for (var i = 0 ; i < contactList.length ; i++) {
      Telephony.backgroundInstance.sendSms(to: contactList[i].number,
message: messageBody).then((value) {
        Fluttertoast.showToast(msg: "Messages sent");
      });
    }
  }
}

Future<void> initializeService() async {
  final service = FlutterBackgroundService();
  AndroidNotificationChannel channel = AndroidNotificationChannel("Women
Safety App ID", "Foreground Service", description: "It is used for important
notification", importance: Importance.low);
  final FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin =
FlutterLocalNotificationsPlugin();
  await
flutterLocalNotificationsPlugin.resolvePlatformSpecificImplementation<AndroidFl
utterLocalNotificationsPlugin>()?.createNotificationChannel(channel);
  await service.configure(iosConfiguration: IosConfiguration(),
androidConfiguration: AndroidConfiguration(onStart: onStart,
isForegroundMode: true, autoStart: true, notificationChannelId: "Women Safety
App ID", initialNotificationTitle: "Foreground Service", initialNotificationContent:
"Initializing", foregroundServiceNotificationId: 888));
  service.startService();
}

@pragma('vm-entry-point')
void onStart(ServiceInstance service) async {
  Location? clocation;
  DartPluginRegistrant.ensureInitialized();
  final FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin =
FlutterLocalNotificationsPlugin();

  if (service is AndroidServiceInstance) {
    service.on('setAsForeground').listen((event) {
      service.setAsForegroundService();
    });
    service.on('setAsBackground').listen((event) {
      service.setAsBackgroundService();
    });
  }
  service.on('stopService').listen((event) {
    service.stopSelf();
  });

  await BackgroundLocation.setAndroidNotification(
    title: "Location tracking is running in the background!",
    message: "You can turn it off from settings menu inside the app",
    icon: '@mipmap/ic_logo'
  );

  BackgroundLocation.startLocationService(distanceFilter: 20);

  BackgroundLocation.getLocationUpdates((location) {
    clocation = location;
  });
  if (service is AndroidServiceInstance) {
    if (await service.isForegroundService()) {
      // await Geolocator.getCurrentPosition(
      //     desiredAccuracy: LocationAccuracy.high,
      //     forceAndroidLocationManager: true
      // ).then((Position position) {
      //   print("bg location: ${position.latitude}");
      // }).catchError((e) {
      //   Fluttertoast.showToast(msg: e.toString());
      // });
      ShakeDetector.autoStart(
          shakeThresholdGravity: 7,
          shakeSlopTimeMS: 500,
          shakeCountResetTime: 3000,
          minimumShakeCount: 1,
          onPhoneShake: () async {
            if (await Vibration.hasVibrator() ?? false) {
              print("Test 2");
              if (await Vibration.hasCustomVibrationsSupport() ?? false) {
                print("Test 3");
                Vibration.vibrate(duration: 1000);
              }
              else {
                print("Test 4");
                Vibration.vibrate();
                await Future.delayed(Duration(milliseconds: 500));
                Vibration.vibrate();
              }
              print("Test 5");
            }
            String messageBody =
"https://www.google.com/maps/search/?api=1&query=${clocation!.latitude}%2
C${clocation!.longitude}";
            sendMessage(messageBody);
          });
      flutterLocalNotificationsPlugin.show(
          888, "Women Safety App",
          clocation == null ? "Please enable the location to use the app" : "Shake
feature enabled ${clocation!.latitude}",
          NotificationDetails(android: AndroidNotificationDetails("Women Safety
App ID", "Foreground Service", channelDescription: "It is used for important
notification", icon: 'ic_bg_service_small', ongoing: true)));
    }
  }
}
```

women_safety_app\lib\utils\pop_with_result_service.dart :-

```
/// PopResult
class PopWithResults<T> {
  /// popped from this page...
  final String fromPage;

  /// pop until this page...
  final String toPage;

  /// results...
  final Map<String,T>? results;

  /// Constructor...
  PopWithResults({required this.fromPage, required this.toPage, this.results});
}
```

women_safety_app\lib\utils\quotes.dart :-

```
List sweetSayings = [
  "Your presence lights up the whole room",
  "We admire, Your strong personality",
  "We’ll help you in any way we can",
  "You are Strong and courageous",
  'I say if I’m beautiful. I say if I’m strong',
  'Above all, be the heroine of your life, not the victim',
];

List<String> articleTitle = [
  "We have to end Violence",
  "Be a change",
  "You are strong"
];
List<String> imageSliders = [
  "https://media.istockphoto.com/vectors/young-woman-looks-at-the-mirror-
and-sees-her-happy-reflection-vector-
id1278815846?k=20&m=1278815846&s=612x612&w=0&h=JUTmV9Of-
_ILOfXBfV9Cmp_41yuTliSdFIcZy5LKuss=",
  "https://media.istockphoto.com/vectors/mental-health-or-psychology-concept-
with-flowering-human-head-vector-
id1268669581?k=20&m=1268669581&s=612x612&w=0&h=YVLTKCZXKugEn40
aqOkir4vcoFeTUAQToa1i3AFYRNU=",
  "https://media.istockphoto.com/photos/confidence-and-strength-concept-
picture-
id1086700012?k=20&m=1086700012&s=612x612&w=0&h=1wWVN3AB7BH7o3
y2A2b-NG3HB9H6Dwkc9OLz2lxgwAY=",
];
```

women_safety_app\lib\widgets\home_widgets\emergencies\AmbulanceEmergency.dart :-

```
import 'package:flutter/material.dart';
import 'package:flutter_phone_direct_caller/flutter_phone_direct_caller.dart';

class AmbulanceEmergency extends StatelessWidget {
  // const AmbulanceEmergency({super.key});

  _callNumber(String number) async{
    await FlutterPhoneDirectCaller.callNumber(number);
  }

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 10.0, bottom: 5),
      child: Card(
        elevation: 5,
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        child: InkWell(
          onTap: () => _callNumber('102'),
          child: Container(
            height: 180,
            width: MediaQuery.of(context).size.width * 0.7,
            decoration: BoxDecoration(
                borderRadius: BorderRadius.circular(20),
                gradient: LinearGradient(
                    begin: Alignment.topLeft,
                    end: Alignment.bottomRight,
                    colors: [
                      Color(0xFFFD8080),
                      Color(0xFFFB8580),
                      Color(0xFFFBD079),
                    ]
                )
            ),
            child: Padding(
              padding: const EdgeInsets.all(8.0),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  CircleAvatar(
                    radius: 25,
                    backgroundColor: Colors.white.withOpacity(0.5),
                    child: Image.asset('assets/ambulance.png'),
                  ),
                  Expanded(
                    child: Column(
                      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                      crossAxisAlignment: CrossAxisAlignment.start,
                      children: [
                        Text(
                          "Ambulance",
                          style: TextStyle(
                            color: Colors.white,
                            fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.06,
                          ),
                        ),
                        Text(
                          "In case of Emergency Call",
                          style: TextStyle(
                            color: Colors.white,
                            // overflow: TextOverflow.ellipsis,
                            // fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.045,
                          ),
                        ),
                        Container(
                          height: 30,
                          width: 80,
                          decoration: BoxDecoration(
                            color: Colors.white,
                            borderRadius: BorderRadius.circular(20),
                          ),
                          child: Center(
                            child: Text("1-0-2",
                              style:
                              TextStyle(
                                color: Colors.red[300],
                                fontSize: MediaQuery.of(context).size.width * 0.055,
                                fontWeight: FontWeight.bold,
                              ),
                            ),
                          ),
                        ),
                      ],
                    ),
                  ),
                ],
              ),
            ),
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\emergencies\ArmyEmergency.dart :-

```
import 'package:flutter/material.dart';
import 'package:flutter_phone_direct_caller/flutter_phone_direct_caller.dart';

class ArmyEmergency extends StatelessWidget {
  // const ArmyEmergency({super.key});

  _callNumber(String number) async{
    await FlutterPhoneDirectCaller.callNumber(number);
  }

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 10.0, bottom: 5),
      child: Card(
        elevation: 5,
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        child: InkWell(
          onTap: () => _callNumber('01126173215'),
          child: Container(
            height: 180,
            width: MediaQuery.of(context).size.width * 0.7,
            decoration: BoxDecoration(
                borderRadius: BorderRadius.circular(20),
                gradient: LinearGradient(
                    begin: Alignment.topLeft,
                    end: Alignment.bottomRight,
                    colors: [
                      Color(0xFFFD8080),
                      Color(0xFFFB8580),
                      Color(0xFFFBD079),
                    ]
                )
            ),
            child: Padding(
              padding: const EdgeInsets.all(8.0),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  CircleAvatar(
                    radius: 25,
                    backgroundColor: Colors.white.withOpacity(0.5),
                    child: Image.asset('assets/army.png'),
                  ),
                  Expanded(
                    child: Column(
                      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                      crossAxisAlignment: CrossAxisAlignment.start,
                      children: [
                        Text(
                          "Indian Army",
                          style: TextStyle(
                            color: Colors.white,
                            fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.06,
                          ),
                        ),
                        Text(
                          "Call Indian Army for Help",
                          style: TextStyle(
                            color: Colors.white,
                            // overflow: TextOverflow.ellipsis,
                            // fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.045,
                          ),
                        ),
                        Container(
                          height: 30,
                          width: 150,
                          decoration: BoxDecoration(
                            color: Colors.white,
                            borderRadius: BorderRadius.circular(20),
                          ),
                          child: Center(
                            child: Text("011-2617-3215",
                              style:
                              TextStyle(
                                color: Colors.red[300],
                                fontSize: MediaQuery.of(context).size.width * 0.05,
                                fontWeight: FontWeight.bold,
                              ),
                            ),
                          ),
                        ),
                      ],
                    ),
                  ),
                ],
              ),
            ),
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\emergencies\FirebrigadeEmergency.dart :-

```
import 'package:flutter/material.dart';
import 'package:flutter_phone_direct_caller/flutter_phone_direct_caller.dart';

class FirebrigadeEmergency extends StatelessWidget {
  // const FirebrigadeEmergency({super.key});

  _callNumber(String number) async{
    await FlutterPhoneDirectCaller.callNumber(number);
  }

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 10.0, bottom: 5),
      child: Card(
        elevation: 5,
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        child: InkWell(
          onTap: () => _callNumber('101'),
          child: Container(
            height: 180,
            width: MediaQuery.of(context).size.width * 0.7,
            decoration: BoxDecoration(
                borderRadius: BorderRadius.circular(20),
                gradient: LinearGradient(
                    begin: Alignment.topLeft,
                    end: Alignment.bottomRight,
                    colors: [
                      Color(0xFFFD8080),
                      Color(0xFFFB8580),
                      Color(0xFFFBD079),
                    ]
                )
            ),
            child: Padding(
              padding: const EdgeInsets.all(8.0),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  CircleAvatar(
                    radius: 25,
                    backgroundColor: Colors.white.withOpacity(0.5),
                    child: Image.asset('assets/flame.png'),
                  ),
                  Expanded(
                    child: Column(
                      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                      crossAxisAlignment: CrossAxisAlignment.start,
                      children: [
                        Text(
                          "Fire Brigade",
                          style: TextStyle(
                            color: Colors.white,
                            fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.06,
                          ),
                        ),
                        Text(
                          "In case of Fire Emergency Call",
                          style: TextStyle(
                            color: Colors.white,
                            // overflow: TextOverflow.ellipsis,
                            // fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.045,
                          ),
                        ),
                        Container(
                          height: 30,
                          width: 80,
                          decoration: BoxDecoration(
                            color: Colors.white,
                            borderRadius: BorderRadius.circular(20),
                          ),
                          child: Center(
                            child: Text("1-0-1",
                              style:
                              TextStyle(
                                color: Colors.red[300],
                                fontSize: MediaQuery.of(context).size.width * 0.055,
                                fontWeight: FontWeight.bold,
                              ),
                            ),
                          ),
                        ),
                      ],
                    ),
                  ),
                ],
              ),
            ),
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\emergencies\policeemergency.dart :-

```
import 'package:flutter/material.dart';
import 'package:flutter_phone_direct_caller/flutter_phone_direct_caller.dart';

class PoliceEmergency extends StatelessWidget {
  // const PoliceEmergency({super.key});

  _callNumber(String number) async{
    await FlutterPhoneDirectCaller.callNumber(number);
  }

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 10.0, bottom: 5),
      child: Card(
        elevation: 5,
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        child: InkWell(
          onTap: () => _callNumber('100'),
          child: Container(
            height: 180,
            width: MediaQuery.of(context).size.width * 0.7,
            decoration: BoxDecoration(
                borderRadius: BorderRadius.circular(20),
              gradient: LinearGradient(
                begin: Alignment.topLeft,
                end: Alignment.bottomRight,
                colors: [
                  Color(0xFFFD8080),
                  Color(0xFFFB8580),
                  Color(0xFFFBD079),
                ]
              )
            ),
            child: Padding(
              padding: const EdgeInsets.all(8.0),
              child: Column(
                crossAxisAlignment: CrossAxisAlignment.start,
                children: [
                  CircleAvatar(
                    radius: 25,
                    backgroundColor: Colors.white.withOpacity(0.5),
                    child: Image.asset('assets/alert.png'),
                  ),
                  Expanded(
                    child: Column(
                      mainAxisAlignment: MainAxisAlignment.spaceEvenly,
                      crossAxisAlignment: CrossAxisAlignment.start,
                      children: [
                        Text(
                          "Active Emergency",
                          style: TextStyle(
                            color: Colors.white,
                            fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.06,
                          ),
                        ),
                        Text(
                          "Call 1-0-0 for Police",
                          style: TextStyle(
                            color: Colors.white,
                            // overflow: TextOverflow.ellipsis,
                            // fontWeight: FontWeight.bold,
                            fontSize: MediaQuery.of(context).size.width * 0.045,
                          ),
                        ),
                        Container(
                          height: 30,
                          width: 80,
                          decoration: BoxDecoration(
                            color: Colors.white,
                            borderRadius: BorderRadius.circular(20),
                          ),
                          child: Center(
                            child: Text("1-0-0",
                              style:
                              TextStyle(
                                color: Colors.red[300],
                                fontSize: MediaQuery.of(context).size.width * 0.055,
                                fontWeight: FontWeight.bold,
                              ),
                            ),
                          ),
                        ),
                      ],
                    ),
                  ),
                ],
              ),
            ),
          ),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\live_safe\BusStationCard.dart :-

```
import 'package:flutter/material.dart';

class BusStationCard extends StatelessWidget {
  final Function? onMapFunction;
  const BusStationCard({super.key, this.onMapFunction});

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 20),
      child: Column(
        children: [
          InkWell(
            onTap: (){
              onMapFunction!('Bus Stops near me');
            },
            child: Card(
              elevation: 3,
              shape: RoundedRectangleBorder(borderRadius:
BorderRadius.circular(20)),
              child: Container(
                height: 50,
                width: 50,
                child: Center(
                  child: Image.asset('assets/bus-stop.png', height: 32),
                ),
              ),
            ),
          ),
          Text('Bus Stations'),
        ],
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\live_safe\HospitalCard.dart :-

```
import 'package:flutter/material.dart';

class HospitalCard extends StatelessWidget {
  final Function? onMapFunction;
  const HospitalCard({super.key, this.onMapFunction});

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 20),
      child: Column(
        children: [
          InkWell(
            onTap: (){
              onMapFunction!('Hospitals near me');
            },
            child: Card(
              elevation: 3,
              shape: RoundedRectangleBorder(borderRadius:
BorderRadius.circular(20)),
              child: Container(
                height: 50,
                width: 50,
                child: Center(
                  child: Image.asset('assets/hospital.png', height: 32,),
                ),
              ),
            ),
          ),
          Text('Hospitals'),
        ],
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\live_safe\PharmacyCard.dart :-

```
import 'package:flutter/material.dart';

class PharmacyCard extends StatelessWidget {
  final Function? onMapFunction;
  const PharmacyCard({super.key, this.onMapFunction});

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 20),
      child: Column(
        children: [
          InkWell(
            onTap: (){
              onMapFunction!('Pharmacies near me');
            },
            child: Card(
              elevation: 3,
              shape: RoundedRectangleBorder(borderRadius:
BorderRadius.circular(20)),
              child: Container(
                height: 50,
                width: 50,
                child: Center(
                  child: Image.asset('assets/pharmacy.png', height: 32),
                ),
              ),
            ),
          ),
          Text('Pharmacy'),
        ],
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\live_safe\PoliceStationCard.dart :-

```
import 'package:flutter/material.dart';

class PoliceStationCard extends StatelessWidget {
  final Function? onMapFunction;
  const PoliceStationCard({super.key, this.onMapFunction});

  @override
  Widget build(BuildContext context) {
    return Padding(
      padding: const EdgeInsets.only(left: 20),
      child: Column(
        children: [
          InkWell(
            onTap: (){
              onMapFunction!('Police Stations near me');
            },
            child: Card(
              elevation: 3,
              shape: RoundedRectangleBorder(borderRadius:
BorderRadius.circular(20)),
              child: Container(
                height: 50,
                width: 50,
                child: Center(
                  child: Image.asset('assets/police-badge.png', height: 32),
                ),
              ),
            ),
          ),
          Text('Police Stations'),
        ],
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\safehome\SafeHome.dart :-

```
import 'package:background_sms/background_sms.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:geocoding/geocoding.dart';
import 'package:geolocator/geolocator.dart';
import 'package:permission_handler/permission_handler.dart';
import 'package:women_safety_app/db/db_services.dart';
import 'package:women_safety_app/model/contactsm.dart';

class SafeHome extends StatefulWidget {
  const SafeHome({super.key});

  @override
  State<SafeHome> createState() => _SafeHomeState();
}

class _SafeHomeState extends State<SafeHome> {
  Position? _currentPosition;
  String? _currentAddress;
  LocationPermission? permission;

  // _getPermission() async => await [Permission.sms].request();
  _isPermissionGranted() async => await Permission.sms.status.isGranted;
  _sendSMS(String phoneNumber, String message, {int? simSlot}) async {
    SmsStatus result = await BackgroundSms.sendMessage(phoneNumber:
phoneNumber, message: message, simSlot: 1);
    if (result == SmsStatus.sent) {
      Fluttertoast.showToast(msg: "Sent");
    }
    else {
      Fluttertoast.showToast(msg: "Failed");
    }
  }

  Future<bool> _handleLocationPermission() async {
    bool serviceEnabled;
    LocationPermission permission;
    serviceEnabled = await Geolocator.isLocationServiceEnabled();
    if(!serviceEnabled) {
      ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content:
Text("Location services are disabled. Please enable the services")));
      return false;
    }
    permission = await Geolocator.checkPermission();
    if(permission == LocationPermission.denied) {
      permission = await Geolocator.requestPermission();
      if(permission == LocationPermission.denied) {
        ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content:
Text("Location permissions are denied")));
        return false;
      }
    }
    if(permission == LocationPermission.deniedForever) {
      ScaffoldMessenger.of(context).showSnackBar(const SnackBar(content:
Text("Location permissions are permanently denied, We cannot request
permissions")));
      return false;
    }
    return true;
  }

  _getCurrentLocation() async {
    final hasPermission = await _handleLocationPermission();
    if(!hasPermission) return;

    // permission = await Geolocator.checkPermission();
    // if (permission == LocationPermission.denied) {
    //   Fluttertoast.showToast(msg: "Location Permissions are denied");
    //   permission = await Geolocator.requestPermission();
    //   if(permission == LocationPermission.deniedForever){
    //     Fluttertoast.showToast(msg: "Location Permissions are permanently
denied");
    //   }
    // }
    await Geolocator.getCurrentPosition(
        desiredAccuracy: LocationAccuracy.high,
        forceAndroidLocationManager: true
    ).then((Position position) {
      setState(() {
        _currentPosition = position;
        print(_currentPosition!.latitude);
        _getAddressFromLatLong();
      });
    }).catchError((e) {
      Fluttertoast.showToast(msg: e.toString());
    });
  }

  _getAddressFromLatLong() async {
    try{
      List<Placemark> placeMarks = await
placemarkFromCoordinates(_currentPosition!.latitude,
_currentPosition!.longitude);
      Placemark place = placeMarks[0];
      setState(() {
        _currentAddress = "${place.locality}, ${place.street},
${place.postalCode}";
      });
    }
    catch (e) {
      Fluttertoast.showToast(msg: e.toString());
    }
  }

  @override
  void initState() {
    super.initState();
    // _getPermission();
    _getCurrentLocation();
  }

  showModelSafeHome(BuildContext context) {
    showModalBottomSheet(
      context: context,
      builder: (context) {
        return Container(
          height: MediaQuery.of(context).size.height / 1.4,
          child: Padding(
            padding: const EdgeInsets.all(14.0),
            child: Column(
              mainAxisAlignment: MainAxisAlignment.center,
              children: [
                Text("Send Your Current Location Immediately to your Emergency
Contacts", style: TextStyle(fontSize: 20), textAlign: TextAlign.center),
                SizedBox(height: 15),
                if(_currentPosition != null) Text(_currentAddress!),
                PrimaryButton(title: "Get LOCATION", onPressed: ()
{_getCurrentLocation();}),
                SizedBox(height: 14),
                PrimaryButton(title: "Send ALERT", onPressed: () async {
                  String recipients = "";
                  List<TContact> contactList = await
DatabaseHelper().getContactList();
                  print(contactList.length);
                  if (contactList.isEmpty) {
                    Fluttertoast.showToast(msg: "Emergency Contact List is empty
now");
                  }
                  else {
                    String messageBody =
"https://www.google.com/maps/search/?api=1&query=${_currentPosition!.latit
ude}%2C${_currentPosition!.longitude}. $_currentAddress";
                    if (await _isPermissionGranted()) {
                      contactList.forEach((element) {
                        _sendSMS("${element.number}",
                            "I am in trouble. Please reach me out at - $messageBody");
                      });
                    }
                    else {
                      Fluttertoast.showToast(msg: "Something went wrong");
                    }
                  }
                })
              ],
            ),
          ),
          decoration: BoxDecoration(
            color: Colors.white,
            borderRadius: BorderRadius.only(
                topLeft: Radius.circular(30),
                topRight: Radius.circular(30),
            ),
          ),
        );
      },
    );
  }

  @override
  Widget build(BuildContext context) {
    return InkWell(
      onTap: () => {
        showModelSafeHome(context),
      },
      child: Card(
        elevation: 5,
        shape: RoundedRectangleBorder(
          borderRadius: BorderRadius.circular(20),
        ),
        child: Container(
          height: 180,
          width: MediaQuery.of(context).size.width * 0.7,
          decoration: BoxDecoration(),
          child: Row(
            children: [
              Expanded(
                  child: Column(
                    children: [
                      ListTile(
                        title: Text("Send Location"),
                        subtitle: Text("Share Location"),
                  ),
                ],
              )),
              ClipRRect(
                  borderRadius: BorderRadius.circular(20),
                  child: Image.asset('assets/route.jpg')
              ),
            ],
          ),
        ),
      ),
    );
```

```
  }
}

class PrimaryButton extends StatelessWidget {
  // const PrimaryButton({super.key});
  final String title;
  final Function onPressed;
  bool loading;
  PrimaryButton({required this.title, required this.onPressed, this.loading =
false});

  @override
  Widget build(BuildContext context) {
    return Container(
      height: 50,
      width: MediaQuery.of(context).size.width * 0.5,
      child: ElevatedButton(
        onPressed: () {
          onPressed();
        },
        child: Text(title, style: TextStyle(fontSize: 17)),
        style: ElevatedButton.styleFrom(
            backgroundColor: Color(0xFFF8B4C5),
            shape: RoundedRectangleBorder(borderRadius:
BorderRadius.circular(30)))
      )
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\custom_appBar.dart :-

```
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:women_safety_app/utils/quotes.dart';

// ignore: must_be_immutable
class CustomAppBar extends StatelessWidget {
  // const CustomAppBar({super.key});
  Function? onTap;
  int? quoteIndex;
  CustomAppBar({this.onTap, this.quoteIndex});

  @override
  Widget build(BuildContext context) {
    return GestureDetector(
      onTap: () {
        onTap!();
      },
      child: Container(
        child: Text(
          sweetSayings[quoteIndex!],
          textAlign: TextAlign.center,
          style: const TextStyle(fontSize: 14, fontWeight: FontWeight.bold),
        ),
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\CustomCarouel.dart :-

```
import 'package:carousel_slider/carousel_slider.dart';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:women_safety_app/utils/quotes.dart';
import 'package:women_safety_app/widgets/home_widgets/safewebview.dart';

class CustomCarouel extends StatelessWidget {
  const CustomCarouel({super.key});

  void navigateToRoute(BuildContext context, Widget route){
    Navigator.push(context, CupertinoPageRoute(builder: (context) => route));
  }

  @override
  Widget build(BuildContext context) {
    return Container(
      child: CarouselSlider(
        options: CarouselOptions(
          aspectRatio: 2.0,
          autoPlay: true,
          enlargeCenterPage: true,
        ),
        items: List.generate(
            imageSliders.length,
                (index) => Card(
              elevation: 5.0,
              shape: RoundedRectangleBorder(
                  borderRadius: BorderRadius.circular(20)),
              child: InkWell(
                onTap: () {
                  if (index == 0) {
                    navigateToRoute(
                        context,
                        SafeWebView(
                            url:
                            "https://plan-international.org/ending-violence/16-ways-
end-violence-girls"));
                  } else if (index == 1) {
                    navigateToRoute(
                        context,
                        SafeWebView(
                            url:
                            "https://www.healthline.com/health/womens-health/self-
defense-tips-escape"));
                  } else {
                    navigateToRoute(
                        context,
                        SafeWebView(
                            url:
                            "https://www.healthline.com/health/womens-health/self-
defense-tips-escape"));
         }
        },
        child: Container(
         decoration: BoxDecoration(
           borderRadius: BorderRadius.circular(20),
           image: DecorationImage(
             fit: BoxFit.cover,
             image: NetworkImage(imageSliders[index]))),
         child: Container(
          decoration: BoxDecoration(
           borderRadius: BorderRadius.circular(20),
            gradient: LinearGradient(colors: [
             Colors.black.withOpacity(0.5),
             Colors.transparent,
            ])),
          child: Align(
           alignment: Alignment.bottomLeft,
           child: Padding(
            padding: const EdgeInsets.only(bottom: 8, left: 8),
            child: Text(
             articleTitle[index],
             style: TextStyle(
              fontWeight: FontWeight.bold,
              color: Colors.white,
              fontSize: MediaQuery.of(context).size.width * 0.05,
             ),
            ),
           ),
          ),
         ),
        ),
       ),
      )
    ),
   ),
  );
 }
}
```

women_safety_app\lib\widgets\home_widgets\emergency.dart :-

```
import 'package:flutter/cupertino.dart';
import
'package:women_safety_app/widgets/home_widgets/emergencies/AmbulanceE
mergency.dart';
import
'package:women_safety_app/widgets/home_widgets/emergencies/ArmyEmerge
ncy.dart';
import
'package:women_safety_app/widgets/home_widgets/emergencies/FirebrigadeE
mergency.dart';
import
'package:women_safety_app/widgets/home_widgets/emergencies/policeemerge
ncy.dart';

class Emergency extends StatelessWidget {
  const Emergency({super.key});

  @override
  Widget build(BuildContext context) {
    return Container(
      width: MediaQuery.of(context).size.width,
      height: 180,
      child: ListView(
        physics: BouncingScrollPhysics(),
        scrollDirection: Axis.horizontal,
        children: [
          PoliceEmergency(),
          AmbulanceEmergency(),
          FirebrigadeEmergency(),
          ArmyEmergency(),
        ],
      ),
    );
  }
}
```

women_safety_app\lib\widgets\home_widgets\safewebview.dart :-

```
import 'package:flutter/material.dart';
import 'package:webview_flutter/webview_flutter.dart';

// ignore: must_be_immutable
class SafeWebView extends StatelessWidget {
  final String? url;
  const SafeWebView({Key? key, this.url}) : super(key: key);

  @override
  Widget build(BuildContext context) {
    return SafeArea(
      child: WebViewWidget(
        controller: WebViewController()..loadRequest(Uri.parse(url!)),
      ),
    );
  }
}
```

women_safety_app\lib\widgets\live_safe.dart :-

```
import 'dart:io';
import 'package:flutter/cupertino.dart';
import 'package:flutter/material.dart';
import 'package:fluttertoast/fluttertoast.dart';
import 'package:url_launcher/url_launcher.dart';
import
'package:women_safety_app/widgets/home_widgets/live_safe/BusStationCard.d
art';
import
'package:women_safety_app/widgets/home_widgets/live_safe/HospitalCard.dart
';
import
'package:women_safety_app/widgets/home_widgets/live_safe/PharmacyCard.da
rt';
import
'package:women_safety_app/widgets/home_widgets/live_safe/PoliceStationCard
.dart';

class LiveSafe extends StatelessWidget {
  const LiveSafe({super.key});

  static Future<void> openMap(String location) async{
    String googleUrl = 'https://www.google.co.in/maps/search/$location';

    if (Platform.isAndroid) {
      if (await canLaunchUrl(Uri.parse(googleUrl))) {
        await launchUrl(Uri.parse(googleUrl));
      }
      else {
        throw 'Could not launch $googleUrl';
      }
    }
    // final Uri _url = Uri.parse(googleUrl);
    // try{
    //   await launchUrl(_url);
    // }
    // catch(e){
    //   Fluttertoast.showToast(msg: 'Something Went Wrong! Call Emergency Number');
    // }
  }

  @override
  Widget build(BuildContext context) {
    return Container(
      height: 90,
      width: MediaQuery.of(context).size.width,
      child: ListView(
        physics: BouncingScrollPhysics(),
        scrollDirection: Axis.horizontal,
        children: [
          PoliceStationCard(onMapFunction: openMap),
          HospitalCard(onMapFunction: openMap),
          PharmacyCard(onMapFunction: openMap),
          BusStationCard(onMapFunction: openMap),
        ],
      ),
    );
  }
}
```

women_safety_app\pubspec.yaml :-

```yaml
name: women_safety_app
description: "A new Flutter project."
publish_to: 'none'
version: 1.0.0+1
environment:
  sdk: '>=3.3.1 <4.0.0'
dependencies:
  flutter:
    sdk: flutter
  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.8
  google_fonts: ^6.2.1
  flutter_native_splash: ^2.4.0
  carousel_slider: ^4.2.1
  webview_flutter: ^4.8.0
  webview_flutter_android: ^3.16.3
  flutter_phone_direct_caller: ^2.1.1
  url_launcher: ^6.3.0
  fluttertoast: ^8.2.6
  firebase_core: ^3.1.0
  firebase_auth: ^5.1.0
  cloud_firestore: ^5.0.1
  shared_preferences: ^2.2.3
  permission_handler: ^11.3.1
  contacts_service: ^0.6.3
  sqflite: ^2.3.3+1
  background_sms: ^0.0.4
  geocoding: ^3.0.0
  geolocator: ^12.0.0
  image_picker: ^1.1.2
  uuid: ^4.4.0
  firebase_storage: ^12.0.1
  cached_network_image: ^3.3.1
  shake: ^2.2.0
  flutter_background_service: ^5.0.6
  flutter_background_service_android: ^6.2.3
  flutter_background_service_ios: ^5.0.1
  flutter_local_notifications: ^17.1.2
  vibration: ^1.9.0
  telephony: ^0.2.0
  background_location: ^0.13.0
  flutter_rating_bar: ^4.0.1
  audioplayers: ^6.0.0
  sensors_plus: ^1.4.1

dev_dependencies:
  flutter_test:
    sdk: flutter
flutter_lints: ^4.0.0
```

```
flutter:
uses-material-design: true

assets:
   - assets/
```

women_safety_app\android\build.gradle :-

```
//buildscript {
//    ext.kotlin_version = '1.7.1' // Add this line to specify the Kotlin version
//    repositories {
//        google()
//        mavenCentral()
//    }
//
//    dependencies {
//        classpath 'com.android.tools.build:gradle:7.0.4'
//        classpath "org.jetbrains.kotlin:kotlin-gradle-plugin:$kotlin_version" //
Add this line to include the Kotlin Gradle plugin
//        // Other dependencies...
//    }
//}

plugins {
    // ...

    // Add the dependency for the Google services Gradle plugin
    id 'com.google.gms.google-services' version '4.4.2' apply false
}

allprojects {
    repositories {
        google()
        mavenCentral()
    }
}

rootProject.buildDir = '../build'
subprojects {
    project.buildDir = "${rootProject.buildDir}/${project.name}"
}
subprojects {
    project.evaluationDependsOn(':app')
}

tasks.register("clean", Delete) {
    delete rootProject.buildDir
}
```

women_safety_app\android\app\build.gradle :-

```
plugins {
    id "com.android.application"
    id "kotlin-android"
    id "dev.flutter.flutter-gradle-plugin"
    id 'com.google.gms.google-services'
}

def localProperties = new Properties()
def localPropertiesFile = rootProject.file('local.properties')
if (localPropertiesFile.exists()) {
    localPropertiesFile.withReader('UTF-8') { reader ->
        localProperties.load(reader)
    }
}

def flutterVersionCode = localProperties.getProperty('flutter.versionCode')
if (flutterVersionCode == null) {
    flutterVersionCode = '1'
}

def flutterVersionName = localProperties.getProperty('flutter.versionName')
if (flutterVersionName == null) {
    flutterVersionName = '1.0'
}

android {
    namespace "com.example.women_safety_app"
//    compileSdk flutter.compileSdkVersion
    compileSdkVersion 34
    ndkVersion flutter.ndkVersion

    compileOptions {
        sourceCompatibility JavaVersion.VERSION_1_8
        targetCompatibility JavaVersion.VERSION_1_8
    }

    kotlinOptions {
        jvmTarget = '1.8'
    }

    sourceSets {
        main.java.srcDirs += 'src/main/kotlin'
    }

    defaultConfig {
        // TODO: Specify your own unique Application ID
(https://developer.android.com/studio/build/application-id.html).
        applicationId "com.example.women_safety_app"
        // You can update the following values to match your application needs.
        // For more information, see:
```

```
https://docs.flutter.dev/deployment/android#reviewing-the-gradle-build-
configuration.
        minSdkVersion 23 //flutter.minSdkVersion
        targetSdkVersion flutter.targetSdkVersion // 30 recommended by youtube
        versionCode flutterVersionCode.toInteger()
        versionName flutterVersionName
    }

    buildTypes {
        release {
            // TODO: Add your own signing config for the release build.
            // Signing with the debug keys for now, so `flutter run --release`
works.
            signingConfig signingConfigs.debug
        }
    }
}

flutter {
    source '../..'
}

dependencies {
    // Import the Firebase BoM
    implementation platform('com.google.firebase:firebase-bom:33.1.1')

    // TODO: Add the dependencies for Firebase products you want to use
    // When using the BoM, don't specify versions in Firebase dependencies
    // https://firebase.google.com/docs/android/setup#available-libraries

}
```

women_safety_app\android\settings.gradle :-

```
pluginManagement {
   def flutterSdkPath = {
      def properties = new Properties()
      file("local.properties").withInputStream { properties.load(it) }
      def flutterSdkPath = properties.getProperty("flutter.sdk")
      assert flutterSdkPath != null, "flutter.sdk not set in local.properties"
      return flutterSdkPath
   }
   settings.ext.flutterSdkPath = flutterSdkPath()

   includeBuild("${settings.ext.flutterSdkPath}/packages/flutter_tools/gradle")

   repositories {
      google()
      mavenCentral()
      gradlePluginPortal()
   }
}

plugins {
   id "dev.flutter.flutter-plugin-loader" version "1.0.0"
   id "com.android.application" version "7.3.0" apply false
   id "org.jetbrains.kotlin.android" version "1.9.0" apply false
}

include ":app"
```

women_safety_app\android\app\src\main\AndroidManifest.xml :-

```
<manifest xmlns:tools="http://schemas.android.com/tools"
   xmlns:android="http://schemas.android.com/apk/res/android">
   <!-- Add the Internet permission here -->
   <uses-feature
      android:name="android.hardware.telephony"
      android:required="false"
      tools:targetApi="eclair" />
   <uses-permission android:name="android.permission.INTERNET"/>
   <uses-permission android:name="android.permission.READ_CONTACTS"/>
   <uses-permission android:name="android.permission.WRITE_CONTACTS"/>
   <uses-permission
android:name="android.permission.ACCESS_FINE_LOCATION"/>
   <uses-permission
android:name="android.permission.ACCESS_COARSE_LOCATION"/>
   <uses-permission android:name="android.permission.SEND_SMS"/>
   <uses-permission android:name="android.permission.VIBRATE"/>
   <application
      android:label="women_safety_app"
      android:name="${applicationName}"
      android:icon="@mipmap/ic_launcher">
      <activity
      android:name=".MainActivity"
      android:exported="true"
      android:launchMode="singleTop"
      android:theme="@style/LaunchTheme"
android:configChanges="orientation|keyboardHidden|keyboard|screenSize|small
estScreenSize|locale|layoutDirection|fontScale|screenLayout|density|uiMode"
      android:hardwareAccelerated="true"
      android:windowSoftInputMode="adjustResize">
      <!-- Specifies an Android theme to apply to this Activity as soon as
         the Android process has started. This theme is visible to the user
         while the Flutter UI initializes. After that, this theme continues
         to determine the Window background behind the Flutter UI. -->
      <meta-data
       android:name="io.flutter.embedding.android.NormalTheme"
       android:resource="@style/NormalTheme"
       />
      <intent-filter>
        <action android:name="android.intent.action.MAIN"/>
        <category android:name="android.intent.category.LAUNCHER"/>
      </intent-filter>
    </activity>
    <!-- Don't delete the meta-data below.
       This is used by the Flutter tool to generate
GeneratedPluginRegistrant.java -->
    <meta-data
      android:name="flutterEmbedding"
      android:value="2" />
  </application>
  <!-- Required to query activities that can process text, see:
     https://developer.android.com/training/package-visibility?hl=en and
https://developer.android.com/reference/android/content/Intent#ACTION_PROC
ESS_TEXT.

     In particular, this is used by the Flutter engine in
io.flutter.plugin.text.ProcessTextPlugin. -->
  <queries>
    <intent>
      <action android:name="android.intent.action.PROCESS_TEXT"/>
      <data android:mimeType="text/plain"/>
    </intent>
  </queries>
</manifest>
```

women_safety_app\android\app\src\debug\AndroidManifest.xml :-

```
<manifest xmlns:android="http://schemas.android.com/apk/res/android">
  <!-- The INTERNET permission is required for development. Specifically,
     the Flutter tool needs it to communicate with the running application
     to allow setting breakpoints, to provide hot reload, etc.
  -->
  <uses-permission android:name="android.permission.INTERNET"/>
</manifest>
```