

# ***WSafe* – A WOMEN SAFETY MOBILE APPLICATION FOR ENHANCED PROTECTION AND SECURITY**

## **A PROJECT REPORT**

*Submitted by*

**SURIYAA V (2020503550)**

**SURYA CHARAN P (2020503033)**

**COURSE CODE: CS6303**

**COURSE TITLE: DISTRIBUTED SYSTEMS**



**DEPARTMENT OF COMPUTER TECHNOLOGY**

**ANNA UNIVERSITY, MIT CAMPUS**

**CHENNAI -600044**

**MAY 2023**

**DEPARTMENT OF COMPUTER TECHNOLOGY**

**ANNA UNIVERSITY, MIT CAMPUS  
CHROMEPET, CHENNAI – 600044**

**BONAFIDE CERTIFICATE**

**Certified that the bonafide report of the project work done by Mr. Suriyaa V,  
Mr. Surya Charan P of Register Number – 2020503550, 2020503033, in the  
Distributed Systems, Subject code CS6303 during the period of Jan 2023 to  
May 2023**

**DATE**

**COURSE – IN - CHARGE**

## **ABSTRACT**

Women's safety is a paramount and critical concern in today's society, demanding the development of effective technological solutions. Ensuring the safety and security of women is essential for promoting gender equality, personal well-being, and social progress. Women face various challenges and risks, including harassment, violence, discrimination, and unequal access to resources and opportunities. To address these issues, concerted efforts are being made to raise awareness, advocate for women's rights, and implement effective measures for women's safety. This includes education and awareness campaigns that aim to change societal attitudes and behaviors, promote respectful relationships, and challenge gender stereotypes and biases. Introducing W-Safe, an innovative woman safety application built using Android Studio written in Java, and XML. W-Safe is designed to empower women and bolster their security through a comprehensive set of features. Application is directly supported on most android devices and provides a simple easy to go place to ensure security and safety. Location alerts, emergency calls, safety tips, numbers of different helplines and government laws which help woman are available thus making a significant leap towards a safer society. W-Safe presents a user-friendly interface and a dependable solution for women's safety. Its development marks a significant stride towards addressing the pressing issue of women's safety in contemporary society, empowering women and fostering a safer environment

## INDEX

<b>S NO.</b>	<b>TITLE</b>	<b>PAGE NO.</b>
	<b>ABSTRACT</b>	3
<b>1.</b>	<b>INTRODUCTION</b>	5
	1.1 ANDROID APPLICATION DEVELOPMENT	5
	1.2 WOMEN SAFETY	5
<b>2.</b>	<b>PROBLEM STATEMENT</b>	7
<b>3.</b>	<b>OBJECTIVES</b>	8
<b>4.</b>	<b>APPLICATION SURVEY</b>	10
<b>5.</b>	<b>SYSTEM ARCHITECTURE</b>	13
<b>6.</b>	<b>DETAILED DESIGN</b>	16
	6.1 SYSTEM FLOW DESIGN	16
	6.2 TOOLS REQUIRED	19
	6.3 ANDROID APP INTERFACE	22
<b>7.</b>	<b>IMPLEMENTATION AND RESULTS</b>	23
<b>8.</b>	<b>COMPARATIVE ANALYSIS</b>	39
<b>9.</b>	<b>CONCLUSION</b>	41
	<b>REFERENCES</b>	42

# **1 INTRODUCTION**

## **1.1 Android Application Development**

Android application development has revolutionized the way we interact with technology, bringing a multitude of functionalities and services to our fingertips. Android Studio, the official IDE for Android development, offers a comprehensive set of tools and features that streamline the app creation process. Its intuitive user interface, coupled with a vast array of built-in templates and code snippets, enables developers to design visually appealing and responsive applications. Android Studio also provides robust debugging and testing capabilities, ensuring the reliability and stability of the developed applications. Moreover, its seamless integration with the Android Software Development Kit (SDK) facilitates the utilization of cutting-edge APIs and libraries, enabling developers to harness the full potential of the Android platform.

## **1.2 Women Safety**

The safety and security of women remain crucial societal concerns. Women often face various forms of harassment, violence, and discrimination, both in public spaces and private domains. Promoting women's safety is not only a matter of personal well-being but also a fundamental aspect of gender

equality. Addressing women's safety requires a multifaceted approach encompassing awareness campaigns, policy reforms, educational initiatives, and the development of technological solutions. By developing a woman safety application, this project aims to contribute to this ongoing effort. The application will provide women with tools to enhance their personal security, access emergency assistance, and foster a sense of empowerment in navigating their daily lives.

## **2 PROBLEM STATEMENT**

The safety and security of women continue to be a pressing concern in today's society. Women face various risks, including harassment, violence, and discrimination, which hinder their ability to live freely and confidently. Despite ongoing efforts to address these issues, there is still a need for comprehensive and accessible solutions that empower women to protect themselves and seek help when needed. Existing safety measures and resources are often fragmented and lack integration, making it challenging for women to access timely assistance in emergencies. Additionally, the lack of awareness about women's safety laws and self-defense techniques further exacerbates the problem. Therefore, there is a critical need for a comprehensive women safety application that leverages the power of technology to provide women with a reliable and user-friendly tool for immediate help, information, and empowerment. This project aims to develop such an application, utilizing Android Studio, Java, and XML, to address the multifaceted challenges faced by women in terms of safety and security.

### 3 OBJECTIVE

Developing a comprehensive women safety application, leveraging Android Studio, Java, and XML, to address the pressing concerns of women's safety and security.

- **Enhance Personal Security:** Develop features such as a shake detector and panic button that allow users to trigger emergency alerts and send SOS messages to registered contacts for immediate assistance. Implement a siren sound to attract attention in critical situations.
- **Provide Location-Based Assistance:** Integrate location services to transmit the user's last known location to registered contacts, enabling them to locate and assist the user quickly. Enable the addition of multiple emergency contacts to maximize the chances of receiving immediate help.
- **Access to Emergency Services:** Incorporate features to find nearby police stations and hospitals, providing users with quick access to essential assistance. Include direct calling facilities for national helplines to ensure immediate contact with emergency services.
- **Empower with Knowledge:** Offer information on women safety laws to educate and empower users about their rights and legal protections. Provide short self-defense videos to equip users with practical techniques and guidance in times of emergency.



- **User-Friendly Interface:** Develop an intuitive and user-friendly interface that promotes ease of use and accessibility for women of all backgrounds and age groups. Prioritize simplicity, responsiveness, and visual appeal in the application design.
- **Reliability and Performance:** Ensure the application's reliability, stability, and performance through rigorous testing and debugging processes. Optimize code and utilize best practices to deliver a seamless user experience.
- **Promote Awareness:** Raise awareness about women's safety concerns through the application, encouraging users to share their experiences, access educational resources, and contribute to a safer environment for all.

By achieving these objectives, the project aims to empower women, enhance their personal security, provide quick access to emergency services, and promote a culture of safety and awareness.

## **4 APPLICATION SURVEY**

### Application Name: "Women Safety"

Women Safety has features like SOS, tips for women safety and tips to escape from threat. But it does not have safety tip videos and current laws to protect women. Whatsapp alerts and several contacts for guardians and relations can be stored. Direct call to police feature is available. Addition of whatsapp based alerts is a significant step towards providing a more easy approach of safety.

### Application Name: "Safetipin"

Safetipin offers features like emergency alerts, location tracking, safety scores for areas, and a community safety network. It allows users to share their location with trusted contacts and provides a safety audit of locations based on factors like lighting, visibility, and more. The app also includes resources such as helpline numbers and safety tips.

### Application Name: "bSafe"

bSafe allows users to set guardians who can be notified in emergency situations. It offers features like live location sharing, SOS alarms, fake call activation, and automatic video and audio recording during emergencies. The app also includes a Follow Me feature for tracking user movements and a Timer Mode for ensuring user safety during specified activities.

### Application Name: "My Safetipin"

My Safetipin is a personal safety app that offers features like location tracking, emergency alerts, and a safety score for different areas. It allows users to share their location with trusted contacts and provides safety recommendations for specific locations. The app also includes resources such as helpline numbers and safety tips.

Application Name: "VithU"

VithU is an emergency alert app that sends SOS messages to pre-selected contacts with just two taps. It allows users to add up to two emergency contacts and sends them an SMS alert with the user's location. The app also includes a loud alarm feature to attract attention in emergency situations.

Application Name: "Raksha Women Safety"

Raksha Women Safety offers features like emergency alerts, live location sharing, and an SOS button for immediate assistance. It includes features such as self-defense tips, helpline numbers, and nearby police station locations. The app also provides the option to save emergency contacts.

Application Name: "Smart 24x7"

Smart 24x7 is a safety app that offers features like emergency alerts, live tracking, and panic buttons. It includes features such as audio and video recording during emergencies, location-based safety services, and access to emergency helpline numbers.

Application Name: "Life360"

Life360 is a family safety app that offers features like location sharing, group messaging, and emergency alerts. It allows users to create private groups for family members and provides real-time location updates. The app includes features such as crash detection and roadside assistance.

## **5 SYSTEM ARCHITECTURE**

The system architecture of the W-Safe women safety application revolves around the user, who installs and interacts with the application on an Android operating system device, such as a touch screen mobile phone. The architecture follows a client-server model, with the application residing on the client-side and utilizing various permissions and services provided by the Android platform.

When the user opens the W-Safe application, they are presented with a user interface (UI) consisting of four button cards: Contact, SMS alerts, Women Laws, and Self defense. Each card encapsulates specific functionalities and actions related to women's safety. The system requires the user to grant necessary permissions such as Location, Notifications, Phone, and SMS permissions to enable the application to function effectively.

The Contact card allows the user to add emergency contacts to the application's database, which is used to send SOS SMS alerts and make emergency phone calls. The application includes a high-speed phone contacts cache for quick retrieval and accessibility. The SMS alerts functionality utilizes the user's standard phone service, and charges may apply according to the user's mobile plan.

The Location permission is utilized to retrieve the user's current location using GPS. This information is crucial for sending accurate location data to

emergency contacts in case of emergencies. Additionally, map API calls are made to search for nearby police stations and hospitals, providing the user with vital information and assistance options.

The Accelerometer sensor in the Android devices plays a significant role in the application. It detects the shaking of the phone, which can indicate an emergency or urgency. When the phone is shaken five times consecutively, the application automatically performs phone calls to the saved emergency contacts and sends SMS alerts to all registered contacts, notifying them of the user's distress.

The application also integrates national helpline numbers for Women in Distress, Domestic Abuse, Police, Student/Child Help, and Ambulance. By clicking the respective buttons, the user can make direct calls to these helpline numbers, ensuring immediate access to emergency services.

Fig 5.1 defines the architectural diagram of the proposed system, where the above-mentioned flow is evident. The app interface is used to get the input actions from the user and these actions call the corresponding events.

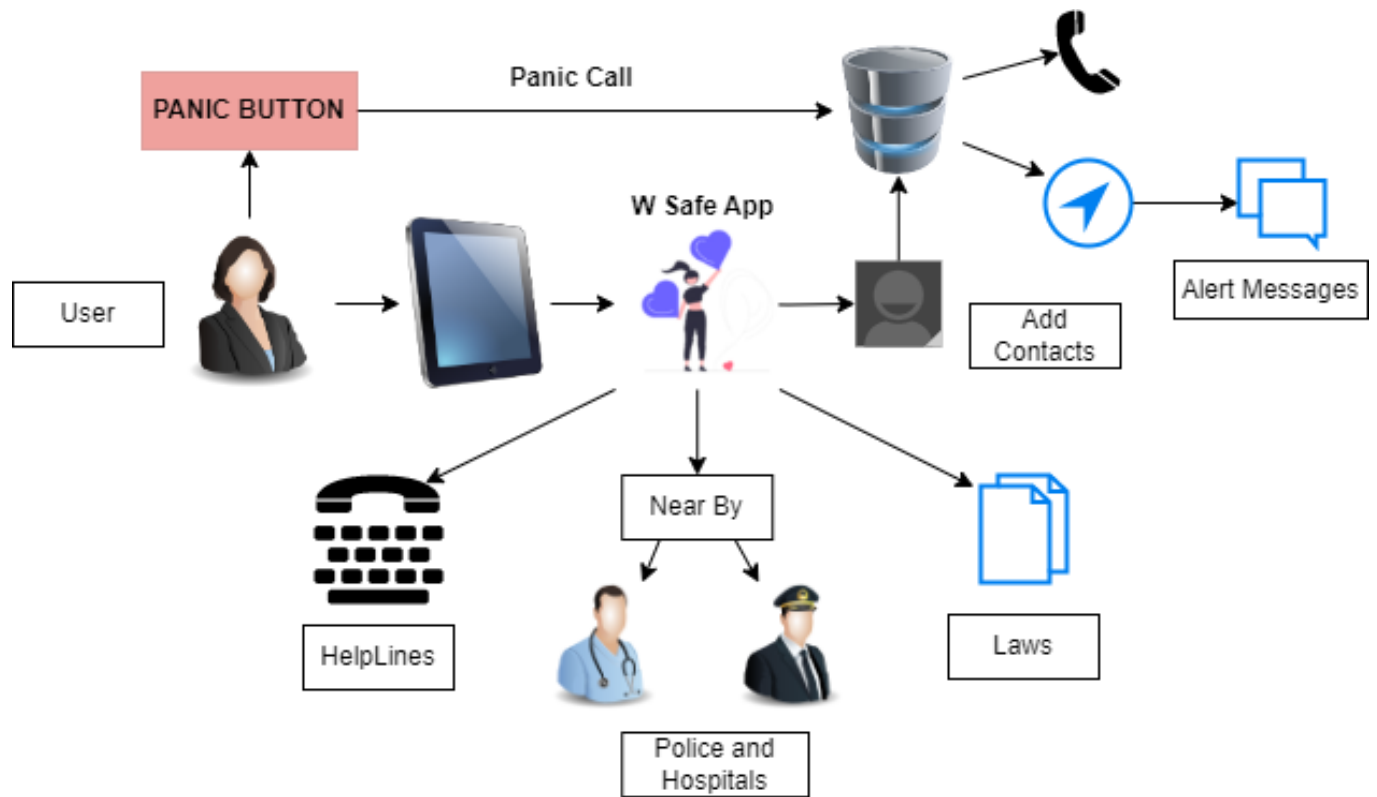


Fig 5.1 Architecture of W-Safe Application

## **6 DETAILED DESIGN**

### **6.1 SYSTEM FLOW DESIGN**

The detailed design flow of the W-Safe women safety application follows a user-centric approach, providing a seamless and intuitive experience for the user. This provides users with a range of features to enhance personal safety and emergency response. The system design incorporates various functionalities. Upon launching the application, the user is presented with the main UI, such as laws information, emergency contacts, self-defense techniques, SMS alerts, and a panic button for immediate assistance. The overall architecture of the application involves multiple activities, data storage, and integration with system services.

The application's main entry point is the MainActivity, which serves as the home page. It displays buttons for different features and implements the View.OnClickListener interface to handle button clicks. When a user clicks on a specific button, the corresponding activity is launched.

One of the features is the LawsActivity, which provides information about relevant laws and regulations related to personal safety. It fetches the data from a reliable source and presents it in a user-friendly manner. Users can navigate through the laws and stay informed about their rights and safety precautions.



The `ContactActivity` allows users to manage their emergency contacts. It utilizes `SharedPreferences` to store the contact information securely. Users can add, delete, and modify contact details through a user interface. The `ContactsAdapter` class handles the `RecyclerView` implementation, displaying the contacts in a list format.

The `SelfDefenseActivity` provides users with self-defense techniques through embedded videos. It uses a `WebView` to load and play instructional videos from a specific URL. Users can watch the videos and learn essential self-defense moves to protect themselves in dangerous situations.

The `SmsActivity` facilitates SMS alerts and service management. Users can start or stop the SMS service for automated alert messages to emergency contacts. It requests necessary permissions such as `SEND_SMS` and location access. The activity also includes a helpline button that launches the `HelplineCall` activity, allowing users to make emergency phone calls to specific helpline numbers.

The `PanicButtonActivity` is a critical feature that enables users to seek immediate help in emergency situations. When the panic button is pressed, the application checks for location permissions and retrieves the user's last known location using the `FusedLocationProviderClient`. It then sends an SOS message to the emergency contacts stored in the application, including the user's current location. Additionally, it initiates a phone call to the first emergency contact number for immediate assistance.

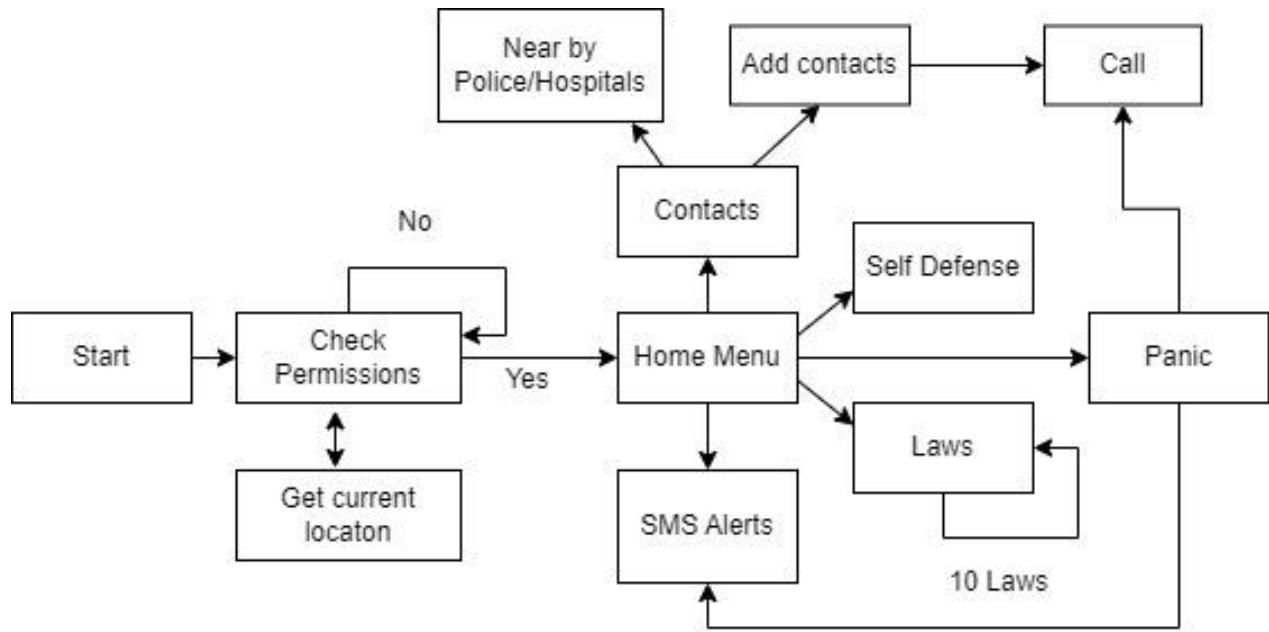


Fig 6.1 System Design Flow

Overall, Fig 6.1 depicts the system design which ensures a seamless flow between different activities and features. The use of SharedPreferences ensures persistent storage of contact information and user preferences. The application utilizes various Android components, including activities, layouts, adapters, and system services such as SMS and location services, to provide a comprehensive safety solution.

The design also prioritizes user experience by implementing intuitive interfaces and responsive actions. It incorporates best practices for data security and privacy, ensuring that contact information and user data are securely stored and transmitted. The application is designed to be scalable, allowing for future enhancements and additions of new safety features.

In conclusion, the safety application's system design encompasses a range of features and functionalities aimed at improving personal safety and emergency response. By providing access to laws information, emergency contacts management, self-defense techniques, SMS alerts, and a panic button, the application empowers users to proactively ensure their safety and quickly seek help in critical situations.

## **6.2 TOOLS REQUIRED**

### **6.2.1 Android Studio**

Android Studio, the integrated development environment (IDE) for Android app development, relies on several essential tools. Firstly, the Java Development Kit (JDK) is necessary as Android Studio is built on Java. It is recommended to have JDK version 8 or newer. Secondly, the Android Software Development Kit (SDK) is crucial as it includes libraries, resources, and tools for Android app creation. Keeping the SDK up to date ensures access to the latest features and APIs. Gradle, the build system integrated into Android Studio, automates tasks such as compilation, packaging, and deployment, managing project dependencies seamlessly. Additionally, the Android Virtual Device (AVD) Manager allows the creation and management of virtual devices (emulators) for testing apps across different configurations. The Android Debug Bridge (ADB), a command-line tool, facilitates communication between the development machine and Android devices or emulators, providing debugging and diagnostic functionalities. Testing on a physical Android device or emulator is essential, with emulators created using AVD

Manager offering versatile testing environments. Furthermore, depending on app requirements, additional libraries and APIs may be incorporated into the project using Android Studio's dependency management system. It is vital to consult the official Android Developer website or Android Studio documentation for the most recent information on the required tools and their versions, as Android Studio is regularly updated with new features and tools.

### **6.2.2 Java for Mobile Application Development**

Java has long been a dominant programming language for mobile application development, offering a host of advantages for developers. Its platform independence allows Java applications to run seamlessly on various operating systems, making it an ideal choice for building cross-platform applications. With its robust object-oriented programming model, Java enables developers to create modular, reusable, and maintainable code. Java's extensive library ecosystem provides access to a wide range of pre-built functions and classes, expediting the development process. Additionally, Java's strong memory management capabilities and automatic garbage collection contribute to the overall performance and efficiency of mobile applications. By leveraging the power of Java, this project ensures a solid foundation for creating a reliable, scalable, and user-friendly woman safety application.

### **6.2.3 Shake Detector**

The Shake Detector package is a library designed to utilize the accelerometer sensor in Android devices for detecting device shaking or movement. This package offers a convenient solution for developers seeking to integrate shake detection functionality into their Android applications. By leveraging the accelerometer sensor, which measures acceleration forces in three dimensions, including X, Y, and Z, the Shake Detector package can analyze the sensor data and identify various types of movements, such as shaking. Key features of this package include the ability to define a shake detection threshold, implement a Shake Listener interface for customizing application behavior based on shake events, and configure parameters like the detection interval to prevent false positives. By including the necessary dependencies, importing the required classes, and following the provided documentation and examples, developers can seamlessly incorporate the Shake Detector package into their Android applications to enable shake detection capabilities. As with any open-source project, it is important to review the documentation, check for licensing requirements, and ensure compatibility with your specific Android development environment before integrating the Shake Detector package into your project.

### **6.2.4 Global Positioning System**

GPS, short for Global Positioning System, is a satellite-based navigation system that provides precise location, velocity, and time information globally.

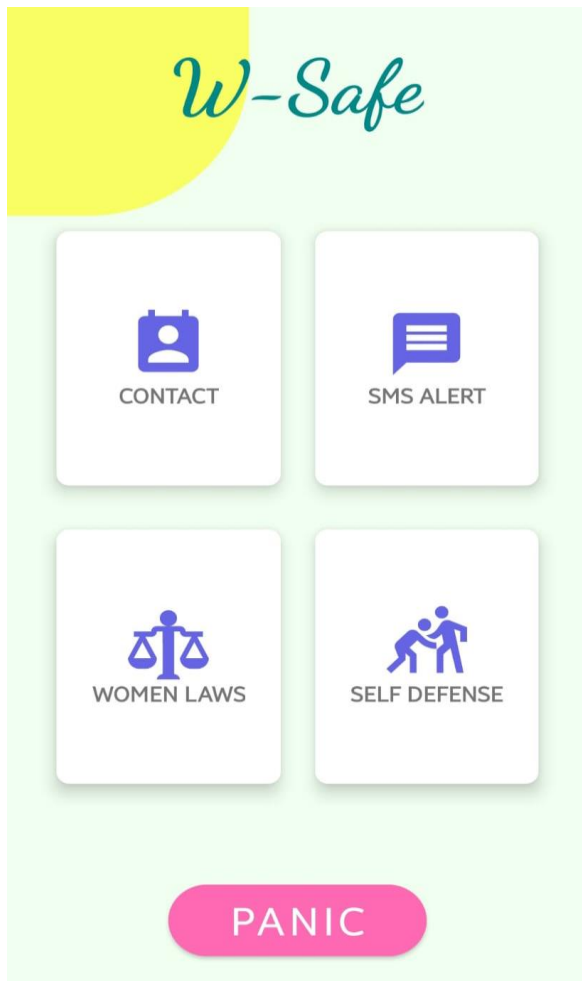
It uses satellites, control stations, and receivers to calculate positions through trilateration. GPS is widely used in navigation, mapping, tracking, and outdoor activities, offering real-time positioning for accurate navigation and route planning. It has become an essential tool in various industries and devices, revolutionizing the way we navigate and interact with the world.

### **6.3 ANDROID APP INTERFACE**

The application interface of W-Safe is thoughtfully designed to provide a user-friendly experience and easy access to essential features for women's safety. The interface is intuitively structured, featuring distinct button cards that represent different functionalities: Contact, SMS alerts, Women Laws, and Self defense. The design focuses on simplicity and visual appeal, with clear and concise icons and text labels. The use of vibrant colors and intuitive navigation enhances the overall user experience. The interface is designed to be responsive and optimized for touch screen mobile devices, ensuring seamless interaction and effortless navigation. By employing a visually pleasing and intuitive interface, W-Safe strives to create a positive and empowering user experience for women seeking to enhance their personal security.

## 7 IMPLEMENTATION AND RESULTS

### Home Page



### Algorithm:

1. Initialize the MainActivity class and implement the View.OnClickListener interface.
2. Initialize the FusedLocationProviderClient for location retrieval.
3. Set the content view to the main activity layout.
4. Set click listeners for the buttons on the home page.

5. When a button is clicked:

- If the "Laws" button is clicked:
  - Start the LawsActivity.
- If the "Contact" button is clicked:
  - Start the ContactActivity.
- If the "Self Defense" button is clicked:
  - Start the SelfDefenseActivity.
- If the "SMS Alerts" button is clicked:
  - Start the SmsActivity.
- If the "Panic Button" is clicked:
  - Check for location permissions.
  - Retrieve the user's last known location using the

FusedLocationProviderClient.

- Send an SOS message to the emergency contacts stored in the application, including the user's current location.
- Retrieve the first emergency contact number.
- Initiate a phone call to the first emergency contact number.

6. The sendMsg function:

- Retrieve the set of emergency contact numbers from shared preferences.
- Send an SOS message to each contact number, including the user's current location.

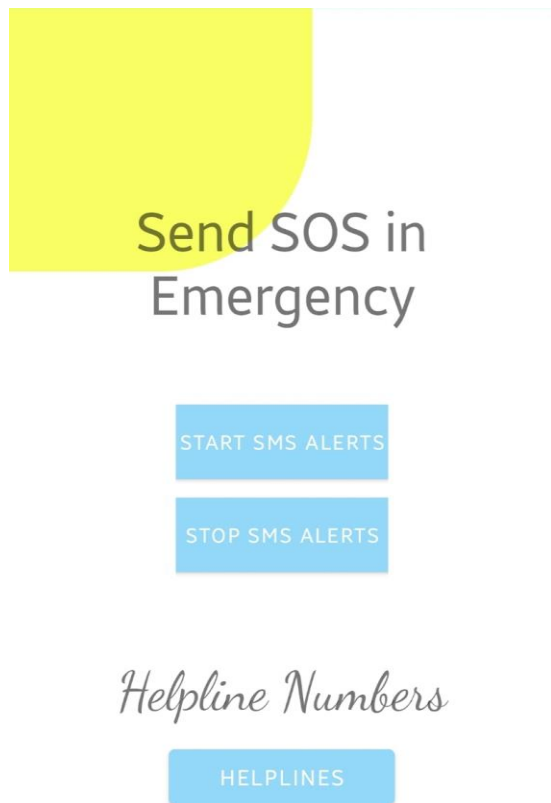
MainActivity implements the View.OnClickListener interface. Then, it initializes the FusedLocationProviderClient to retrieve the user's location. The content view is set to the main activity layout. Click listeners are set for the



buttons on the home page. When a button is clicked, the corresponding action is performed. If the "Laws" button is clicked, the LawsActivity is started. If the "Contact" button is clicked, the ContactActivity is started. If the "Self Defense" button is clicked, the SelfDefenseActivity is started. If the "SMS Alerts" button is clicked, the SmsActivity is started. If the "Panic Button" is clicked, the algorithm checks for location permissions and retrieves the user's last known location using the FusedLocationProviderClient. Then, an SOS message is sent to the emergency contacts stored in the application, including the user's current location. The first emergency contact number is retrieved, and a phone call is initiated to that number.

A sendMsg function that retrieves the set of emergency contact numbers from shared preferences and sends an SOS message to each contact number, including the user's current location.

### **SMS Service:**



### Algorithm:

1. Create SmsActivity class
2. Declare start, stop, and helpline buttons
3. Implement onBackPressed() method:
  - Call super.onBackPressed()
  - Start MainActivity
4. Implement onCreate() method:
  - Call super.onCreate() and setContentView() with activity\_sms layout
  - Find and assign the stop, start, and helpline buttons using findViewById()
  - Set onClick listeners for start, stop, and helpline buttons
5. Implement helplines() method:

- Start HelplineCall activity

#### 6. Implement stopService() method:

- Create an Intent for ServiceMine class with "stop" action
- If Build.VERSION.SDK\_INT is greater than or equal to
  - If ServiceMine.isRunning is true:
    - Start the foreground service using startForegroundService() method
    - Show a Snackbar message "Service Stopped!"
- Else:
  - If ServiceMine.isRunning is true:
    - Start the service using startService() method
    - Show a Snackbar message "Service Stopped!"

#### 7. Implement startServiceV() method:

- If Build.VERSION.SDK\_INT is greater than or equal to

#### Build.VERSION\_CODES.O:

- If the app doesn't have permission to draw overlays:
  - Create an Intent to open the app settings for overlay permission
  - Start the intent
- If the app has SMS, and ACCESS\_FINE\_LOCATION permissions:
  - Create an Intent for ServiceMine class with "Start" action
  - If Build.VERSION.SDK\_INT is greater than or equal to

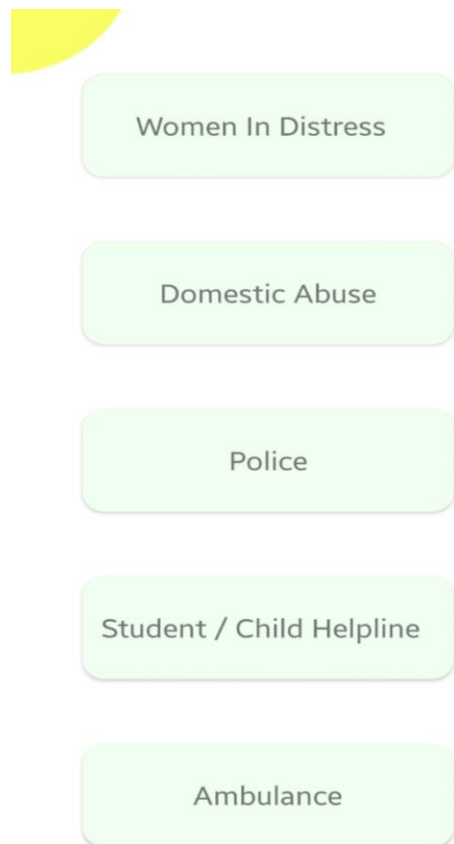
#### Build.VERSION\_CODES.O:

- Start the foreground service using startForegroundService() method
- Show a Snackbar message "Service Started!"
- Else:
  - Start the service using startService() method

- Show a Snackbar message "Service Started!"

The SmsActivity is responsible for managing the SMS service and interacting with the user. It has buttons to start and stop the service, as well as a button to access the helpline numbers. When the user clicks the start button, the activity checks for necessary permissions and starts a service called ServiceMine. The ServiceMine class is not provided here, but it presumably handles the SMS functionality. Clicking the stop button sends a broadcast to stop the ServiceMine service if it is currently running. The helpline button opens the HelplineCall activity, allowing the user to access emergency helpline numbers.

### Helplines Feature:

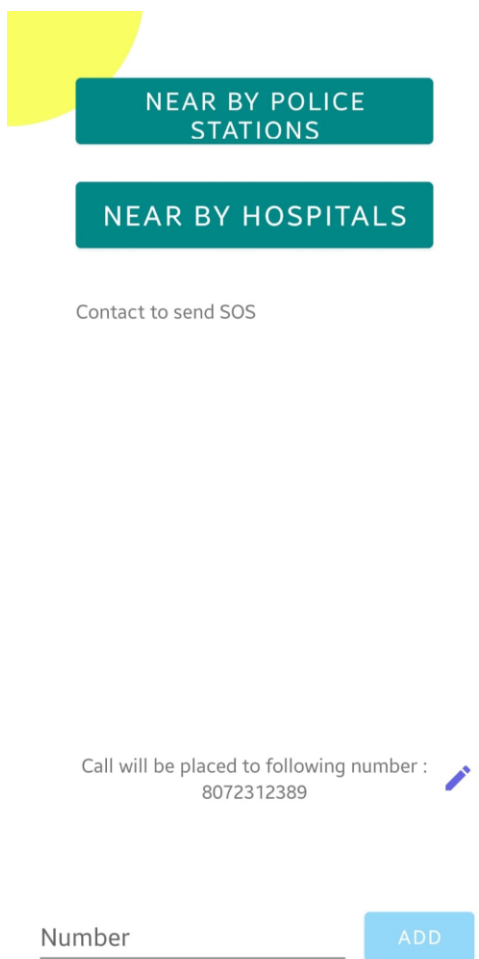


## **Algorithm:**

1. Create HelplineCall class
2. Implement onCreate() method:
  - Call super.onCreate() and setContentView() with activity\_helpline layout
  - Find and assign the distress, abuse, police, helpline, and ambulance buttons using findViewById()
  - Set onClick listeners for all buttons
3. Implement callDistress() method:
  - Create an Intent for dialing the distress helpline number (1091)
  - Start the intent
4. Implement callAbuse() method:
  - Create an Intent for dialing the abuse helpline number (181)
  - Start the intent
5. Implement callPolice() method:
  - Create an Intent for dialing the police helpline number (100)
  - Start the intent
6. Implement callHelpline() method:
  - Create an Intent for dialing the helpline number (1098)
  - Start the intent
7. Implement callAmbulance() method:
  - Create an Intent for dialing the ambulance helpline number (108)
  - Start the intent

The HelplineCall activity displays various buttons for different helpline services, such as distress, abuse, police, helpline, and ambulance. When the user clicks any of these buttons, an intent is created with the corresponding helpline number, and the device's dialer is opened to make the call.

## Contacts and Nearby Feature



### Algorithm:

1. Start the ContactActivity.
2. Initialize the UI components.

3. Load the primary contact number from SharedPreferences and display it on the screen.
4. Set up the RecyclerView and the ContactsAdapter.
5. Retrieve the list of contact numbers from SharedPreferences and populate the RecyclerView.
6. Handle the click event for the add contact button:
  - 6.1. Get the contact number entered by the user.
  - 6.2. Validate the contact number.
  - 6.3. If the contact number is valid:
    - 6.3.1. Add the contact number to the list in SharedPreferences.
    - 6.3.2. Update the RecyclerView to reflect the changes.
7. Handle the click event for the edit primary contact button:
  - 7.1. Show a dialog box with an input field for the contact number.
  - 7.2. When the user clicks on the save button:
    - 7.2.1. Validate the contact number.
    - 7.2.2. If the contact number is valid:
      - 7.2.2.1. Save the contact number as the primary contact in SharedPreferences.
      - 7.2.2.2. Update the displayed primary contact number.
8. Handle the click event for the delete icon in each contact item:
  - 8.1. Get the position of the clicked item.
  - 8.2. Remove the contact number at that position from the list in SharedPreferences.
  - 8.3. Update the RecyclerView to reflect the changes.
9. Handle the click event for the nearby police stations button:

9.1. Launch an intent to open the Google Maps app with a search query for police stations.

10. Handle the click event for the nearby hospitals button:

10.1. Launch an intent to open the Google Maps app with a search query for hospitals.

11. Handle the back button press:

11.1. Return to the MainActivity.

12. End the ContactActivity.

The ContactActivity is an Android activity that allows users to manage their emergency contact numbers. It provides a user interface where users can add, delete, and view their contact numbers. The activity starts by initializing the UI components such as buttons and views. It also handles the click events for adding a contact number and editing the primary contact number. The primary contact number is stored in SharedPreferences. When the user clicks on the edit button, a dialog is shown where the user can enter a 10-digit contact number. If a valid number is entered, it is saved in SharedPreferences as the primary contact number. The activity also includes functionality to display the list of contact numbers using a RecyclerView and a custom ContactsAdapter. The ContactsAdapter is responsible for inflating the layout for each item in the RecyclerView and binding the data. It receives the list of contact numbers as input and provides a way to delete a contact number by clicking on a delete icon associated with each item. The activity includes additional buttons for finding nearby police stations and hospitals. When these buttons are clicked,



the activity launches an intent to open the Google Maps app with a specific search query for police stations or hospitals.

## Laws Feature

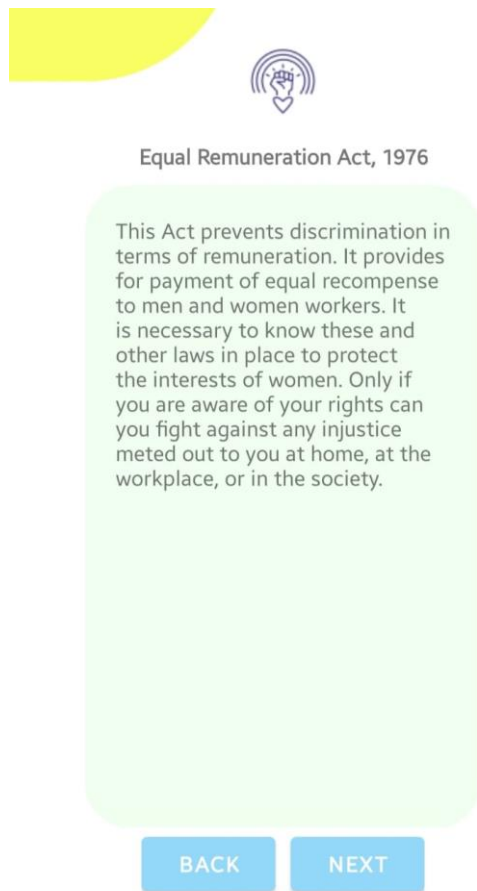


### Algorithm:

1. Initialize the LawsActivity class, extending AppCompatActivity.
2. Override the onBackPressed() method to navigate back to the MainActivity.
3. In the onCreate() method:

- a. Set the content view to the activity\_laws layout.
  - b. Initialize a RecyclerView and find it by its ID.
  - c. Create an array of law names.
  - d. Create an instance of MyAdapter with the laws array and a click listener.
  - e. Set the adapter for the RecyclerView.
  - f. Set a LinearLayoutManager for the RecyclerView.
  - g. Set a click listener for the back button to navigate back to the MainActivity.
4. End of LawsActivity.

## Law Displayer



## Algorithm:

1. Initialize the LawDisplayerActivity class, extending AppCompatActivity and implementing View.OnClickListener.
2. Declare variables for TextViews, buttons, and other views.
3. In the onCreate() method:
  - a. Set the content view to the activity\_law\_displayer layout.
  - b. Find the TextViews, buttons, and other views by their IDs.
  - c. Get the position value from the intent extras.
  - d. Initialize arrays for law names and law content.
  - e. Set a click listener for the close button to navigate back and finish the activity.
  - f. Set click listeners for the back and next buttons.
  - g. Call the setData() method to set the initial data.
4. Create the setData() method:
  - a. Set the oneLine TextView text to the law name at the current counter position.
  - b. Set the big TextView text to the law content at the current counter position.
5. Implement the onClick() method:
  - a. If the next button is clicked, increment the counter if it is within the array bounds, otherwise reset it.
  - b. If the back button is clicked, decrement the counter if it is not zero, otherwise set it to the last index.
  - c. Call the setData() method.

6. End of LawDisplayerActivity.

## Self Defense Feature



This video shows some techniques of self defense

### Algorithm:

1. Start of the activity
2. Set the layout to activity\_self\_defense
3. Find the WebView element by its ID and assign it to the webView variable
4. Configure the webView:
  - 4.1 Set the WebChromeClient for handling web chrome-related interactions
  - 4.2 Enable automatic loading of images
  - 4.3 Enable JavaScript execution
5. Load the YouTube video URL into the webView:

5.1 Call the `loadUrl()` method on the `webView`, passing the URL of the YouTube video to be loaded

6. End of the activity

## Panic Button



### Algorithm:

1. Initialize the `PanicButtonActivity` class.
2. Set the content view to the panic button layout.
3. Initialize the `FusedLocationProviderClient` for location retrieval.
4. Set a click listener for the panic button.
5. When the panic button is clicked:
6. Check for location permissions.
7. Retrieve the user's last known location
8. Send an SOS message to the emergency contacts stored in the application, including the user's current location.
9. Retrieve the first emergency contact number.
10. Initiate a phone call to the first emergency contact number.
11. The `sendSOSMessage` function:
  1. Retrieve the set of emergency contact numbers from shared preferences.
  2. Compose an SOS message with the user's current location.
  3. Send the SOS message to each contact number.
12. The `initiatePhoneCall` function:
  1. Retrieve the first emergency contact number from shared preferences.

2. Initiate a phone call to the first emergency contact number.

## **Shake Detection**

1. Initialize the ShakeDetector class and Override the onSensorChanged method to detect sensor changes.
2. Calculate the acceleration values in the x, y, and z directions using the sensor data.
3. Apply a low-pass filter to remove noise from the acceleration values.
4. Calculate the total acceleration by taking the square root of the sum of squared values from each direction.
5. Update the current acceleration and last acceleration values.
6. Calculate the delta acceleration by subtracting the current acceleration from the last acceleration.
7. Update the last acceleration with the current acceleration.
8. Check if the delta acceleration exceeds a predefined threshold value, indicating a significant change.
9. If the threshold is exceeded, it indicates a shake gesture.
10. Perform the desired action or trigger an event when a shake gesture is detected.

## 8 COMPARATIVE ANALYSIS

Women Safety - has features like SOS, tips for women safety and tips to escape from threat. But it does not have safety tip videos and current laws to protect women. Various Helpline numbers are present in the W-Safe application which is not present in this current application. W-Safe offers a more understandable user interface compared to this.

bSafe - provides features like emergency alerts and live location sharing, while W-Safe offers additional benefits such as integration with local authorities for immediate response and a comprehensive safety plan management system. W-Safe also focuses on proactive measures like self-defense training resources and safety reminders, providing a more holistic approach to women's safety.

Safetipin - offers similar features such as emergency alerts and location tracking. It also provides features like shake detection. But it does not have Nearby hospitals and police stations features which W-Safe has.

“My Safetipin” - offers features like emergency alerts and location tracking. However, W-Safe distinguishes itself by incorporating advanced features like shake detection for panic alerts and personalized safety plans. W-Safe empowers users with customized safety measures tailored to their individual needs, enhancing the overall safety experience.

VithU - focuses primarily on emergency alerts and location sharing, which are also offered by W-Safe. Also, VithU offers functionalities like safety tips. These additional features enhance the effectiveness and versatility of the W-Safe application.

Raksha Women Safety - shares some common features with W-Safe, such as emergency alerts and live location sharing. However, W-Safe distinguishes itself by offering shake detection for panic alerts, and a comprehensive safety plan management system along with self defense techniques. These additional features provide users with a more robust and proactive safety solution.

Smart 24x7 - provides emergency alerts and live tracking, Smart 24x7 stands out by offering advanced features like shake detection for panic alerts, integration with local authorities, and personalized safety plans. W-Safe empowers users with comprehensive safety measures and resources, enhancing their overall security and peace of mind.

Life360 - focuses on family safety and location sharing, W-Safe is specifically designed for women's safety and offers features tailored to their unique needs. W-Safe provides advanced functionalities like shake detection for panic alerts, integration with local authorities, and personalized safety plans. These features make W-Safe a more comprehensive and specialized solution for women's safety.



## **9 CONCLUSION**

The development of the W-Safe application has addressed the critical need for women's safety in today's society. By leveraging advanced technologies such as Android Studio, Java, and XML, W-Safe offers a comprehensive solution to empower women and enhance their personal security and safety. The application incorporates features like a shake detector, panic button, SOS alerts, and location tracking to ensure prompt assistance during emergencies. Additionally, the integration of nearby police stations and hospitals, women safety laws, self-defense videos, and national helpline numbers provides users with valuable resources and information. The user-friendly interface and seamless functionalities make W-Safe accessible and easy to navigate, further enhancing its effectiveness. The successful implementation of the project showcases the potential of technology in promoting women's safety and creating a safer environment. W-Safe serves as a significant step forward in safeguarding women and raising awareness about their rights and well-being.

## REFERENCES

- <https://play.google.com/store/apps/details?id=com.manvithaapps.womensafety>
- <https://play.google.com/store/apps/details?id=com.bipper.app.bsafety>
- <https://safetipin.com/>
- [https://play.google.com/store/apps/details?id=com.safetipin.mysafetipin&hl=en\\_IN&gl=US](https://play.google.com/store/apps/details?id=com.safetipin.mysafetipin&hl=en_IN&gl=US)
- <https://androidjungle.com/vithu-app-woman-safety-app-gumrah.html>
- [https://play.google.com/store/apps/details?id=com.app.raksha&hl=en\\_IN&gl=US](https://play.google.com/store/apps/details?id=com.app.raksha&hl=en_IN&gl=US)
- [https://play.google.com/store/apps/details?id=smart.emergencyservice&hl=en\\_IN&gl=US](https://play.google.com/store/apps/details?id=smart.emergencyservice&hl=en_IN&gl=US)
- <https://play.google.com/store/apps/details?id=com.life360.android.safetymap>