Faculty of Electronics and Computer Technology and Engineering

DEVELOPMENT OF WOMEN SAFETY APPLICATION USING ANDROID STUDIO

SUHANNA BINTI MAHTAR@MOKHTAR

Bachelor of Computer Engineering Technology (Computer Systems) with Honours

2025

DEVELOPMENT OF WOMEN SAFETY APPLICATION USING ANDROID STUDIO

SUHANNA BINTI MAHTAR@MOKHTAR

This report is submitted in partial fulfilment of the requirements for the degree of Bachelor of Electronics Engineering Technology (Industrial Electronics) with Honours

Faculty of Electronics and Computer Technology and Engineering
Universiti Teknikal Malaysia Melaka

2025

UNIVERSITI TEKNIKAL MALAYSIA MELAKA
FAKULTI TEKNOLOGI DAN KEJURUTERAAN ELEKTRONIK DAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : Development Of Woman Safety Application Using Android Studio

Sesi Pengajian : 2024/2025

Saya SUHANNA BINTI MAHTAR@MOKHTAR mengaku membenarkan laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (✓):

☐ **SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐ **TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan.)
☑ **TIDAK TERHAD**	

Disahkan oleh:

(TANDATANGAN PENULIS)

Alamat Tetap: .....
.....
.....

DR. NOR HAFIZAH BINTI HUSSIN
Pensyarah
Fakulti Teknologi Dan Kejuruteraan Elektronik
Dan Komputer (FTKEK)
Universiti Teknikal Malaysia Melaka (UTeM)

Tarikh : 7 JANUARI 2025

Tarikh : 7 JANUARI 2025

*CATATAN: Jika laporan ini SULIT atau TERHAD, sila lampirkan surat daripada pihak berkuasa/organisasi berkenaan dengan menyatakan sekali tempoh laporan ini perlu dikelaskan sebagai SULIT atau TERHAD.

DECLARATION

I declare that this project report entitled “Development Of Woman Safety Application Using Android Studio” is the result of my own research exces cited in the references. The project report has not been accepted for any degree and is not concurrently submitted in candidature of any other degree.

Signature	:	
Student Name	:	SUHANNA BINTI MAHTAR@MOKHTAR
Date	:	7 JANUARY 2025

APPROVAL

I hereby declare that I have checked this Bachelor Degree Project 2 (PSM2) report and in my opinion, this project report is adequate in terms of scope and quality for the award of the degree of Bachelor of Computer Engineering Technology (Computer Systems) with Honours

Signature :

Supervisor Name : DR. NOR HAFIZAH BINTI HUSSIN

Date : 7 JANUARY 2025

:

:

:

DEDICATION

This project is lovingly dedicated to the memory of my late parents, whose unwavering love, sacrifices, and guidance continue to inspire me every day. Though they are no longer with me, their dreams and values have been the foundation of my journey. I also dedicate this work to my family, who have stood by me through thick and thin. Their constant support, encouragement, and belief in my abilities have been my greatest source of strength. To my late parents and my family, this achievement is as much yours as it is mine.

ABSTRACT

The present work studies the development and effectiveness of mobile applications meant to improve women's safety, using Android Studio as the major development tool. Considering the global significance of women's safety, this study addresses the need for dependable, quick solutions in both urban and rural areas where women confront harassment and abuse. The objective is to provide a reliable, quick-response tool that enhances personal safety through features such as emergency notifications, real-time location tracking, and siren activation to deter attackers and attract attention. Women frequently face threats to their safety in urban and rural areas, necessitating the development of an Android application with features like emergency calls, real-time location tracking, and siren activation to provide immediate assistance, alert nearby individuals, and empower users to respond effectively to harassment and abuse. The study discovers that using advanced technologies like GPS, real-time data synchronisation, and user-friendly interfaces improves the app's effectiveness. Application was successfully developed and tested, demonstrating effective emergency response features such as siren activation, GPS tracking, and SMS communication, with high compatibility across devices, though minor optimizations in system responsiveness and user experience are needed for enhanced reliability. The study concludes that the developed application significantly improves user confidence and outlines potential future enhancements to further boost reliability and functionality.

ABSTRAK

Kajian ini mengkaji pembangunan dan keberkesanan aplikasi mudah alih yang bertujuan untuk meningkatkan keselamatan wanita, menggunakan Android Studio sebagai alat pembangunan utama. Memandangkan kepentingan global keselamatan wanita, kajian ini menangani keperluan untuk penyelesaian yang boleh dipercayai dan cepat di kawasan bandar dan luar bandar di mana wanita menghadapi gangguan dan penderaan. Objektifnya adalah untuk menyediakan alat tindak balas cepat yang boleh dipercayai yang meningkatkan keselamatan peribadi melalui ciri-ciri seperti pemberitahuan kecemasan, penjejakan lokasi masa nyata, dan pengaktifan siren untuk menghalang penyerang dan menarik perhatian. Wanita sering menghadapi ancaman terhadap keselamatan mereka di kawasan bandar dan luar bandar, yang memerlukan pembangunan aplikasi Android dengan ciri-ciri seperti panggilan kecemasan, penjejakan lokasi masa nyata, dan pengaktifan siren untuk memberikan bantuan segera, memberi amaran kepada individu yang berhampiran, dan memberdayakan pengguna untuk bertindak balas dengan berkesan terhadap gangguan dan penyalahgunaan. Kajian mendapati bahawa penggunaan teknologi canggih seperti GPS, penyelarasan data masa nyata, dan antara muka mesra pengguna meningkatkan keberkesanan aplikasi dengan ketara. Aplikasi telah berjaya dibangunkan dan diuji, menunjukkan ciri-ciri tindak balas kecemasan yang berkesan seperti pengaktifan siren, penjejakan GPS, dan komunikasi SMS, dengan keserasian tinggi di seluruh peranti, walaupun pengoptimuman kecil dalam responsiviti sistem dan pengalaman pengguna diperlukan untuk meningkatkan kebolehpercayaan. Kajian ini menyimpulkan bahawa aplikasi yang dibangunkan secara signifikan meningkatkan keyakinan pengguna dan menggariskan potensi penambahbaikan masa depan untuk meningkatkan lagi kebolehpercayaan dan fungsi.

ACKNOWLEDGEMENTS

I would like to express my heartfelt gratitude to everyone who contributed to the successful completion of this project, "Development of a Women's Safety Application Using Android Studio."

First and foremost, I would like to thank Dr. Nor Hafizah Binti Hussin, my project supervisor, for her invaluable guidance, encouragement, and constructive feedback throughout this journey. Her expertise and support have been instrumental in shaping this project.

I also extend my sincere appreciation to the Faculty of Electronics and Computer Engineering, Universiti Teknikal Malaysia Melaka (UTeM), for providing the resources and opportunity to undertake this project.

My highest appreciation goes to my late parents and family members for their love and prayers during my studies.

Finally, I would like to thank everyone who contributed, directly or indirectly, to the successful realization of this project.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

UI	-	User Interface
UX	-	User Experience
GPS	-	Global Positioning System
XML	-	Extensible Markup Language
API	-	Application Programming Interface
SDK	-	Software Development Kit
SOS	-	Save Our Souls

LIST OF APPENDICES

CHAPTER 1

INTRODUCTION

1.1 Background

The safety of women has grown in importance on a global scale. The development of smartphones and the spread of mobile applications has opened new possibilities for protecting women's safety. Numerous women safety applications have been developed using Android Studio. These applications are designed with various features aimed at empowering and protecting women in times of distress. They often include functionalities such as sending alerts to selected contacts or emergency services, sharing real-time location, and triggering siren. With the goal of offering reliable safety solutions for women nearly everywhere, the field of developing women's safety applications with Android Studio is expanding. It serves as evidence of how technology may be used to improve society.

1.2 Addressing Global and Society Issues with Personalized Solutions

Women's safety is a pressing global and societal issue that requires immediate attention. To address this issue, the development and utilization of safety applications specifically tailored for women's safety is crucial. These safety applications can provide women with a sense of security and support in various situations. They can include features such as emergency alert systems, real-time location tracking, self-defense tips and resources, and 24/7 helplines. These safety applications aim to empower women and ensure their safety in public spaces, at home, or during travel. The proposed Android application described in the sources aims to address the issue of women's safety by providing a channel for users to

request help in case of emergencies, sending distress messages to pre-entered emergency contacts along with the user's location using GPS. Furthermore, future improvements can be made to these safety applications to enhance their effectiveness. Additionally, the issue of women's safety goes beyond just physical safety. It also encompasses the perception of safety, wherein women may feel less safe overall but are more likely to notice safety enhancements and feel safer as a result.

1.3 Problem Statement

Women's safety is frequently threatened in towns and cities by acts of assault and harassment. Current solutions frequently fail to perform well or do not come through quickly enough. The creation of a thorough Android application that tackles these issues is urgently needed. For women to be able to get help when they need it, this app must include emergency assistance and real-time location tracking capabilities.

Women face serious obstacles to their safety and wellbeing since they frequently experience harassment, abuse, and feelings of insecurity in both urban and rural settings. These problems can include everything from physical assault and threats to verbal abuse and verbal abuse. So, this aimed at improving women's safety by integrating features option to make an emergency call, send a message, and update her whereabouts to her family’s close relative.

There is a critical need for applications that offer immediate assistance and barrier to potential threats. Women can empower themselves and seek the help of surrounding individuals or authorities by discreetly and rapidly signaling for help. The application seeks to integrate a function that sounds a siren to notify individuals in nearby areas when the user

triggers it or when it is triggered by established in advance safety factors. This feature is intended to draw attention to itself and discourage would-be attackers, improving women's safety in a variety of situations.

1.4 Project Objective

The objectives this project aims to achieve are as the following:

a) To study the latest women's personal safety features mobile application and analyses the usefulness of smartphones for women's safety

b) To develop a smartphone app that can improve women's safety by offering a dependable tool that they can use in dangerous circumstances.

c) To evaluate the performance of the developed women safety application.

1.5 Scope of Project

The scope of this project are as follows:

a) Program Development

The scope of program development for the project involves defining the overall structure, objectives, and functionalities of the women's safety application. This includes conducting thorough requirements analysis to identify key features such as real-time location tracking, emergency SOS alerts, safe route recommendations, and crowd-sourced incident reporting. The program development phase entails designing the application architecture, user interface (UI), and user experience (UX) to ensure ease of use and accessibility for women from diverse backgrounds. It also involves

outlining the logic flow, data storage mechanisms, and integration of external services such as GPS and notification systems.

b) Software Development

When it comes to software development, the scope includes implementing Android Studios to construct the women's safety application. To implement the different features and functionalities of the application that were specified during the programme development phase, this involves writing code in the Java or Kotlin programming languages. Establishing the development racking and designing user interfaces with XML layouts, implementing backend logic for data processing and interaction, integrating external APIs or services for communication and location tracking, and managing exceptions and error handling are all part of software development.

c) Testing and Debugging

The testing and debugging aspect of the project involves ensuring the reliability, functionality, and usability of the women's safety application developed using Android Studios. This includes planning and executing various testing activities to identify and address any issues or bugs in the application. The scope of testing encompasses different levels, including:

Unit Testing: Testing individual components, classes, and methods to ensure they perform as expected and meet the specified requirements.

Integration Testing: Testing the integration and interaction between different modules or components of the application to verify that they work together seamlessly.

Functional Testing: Testing the application's features and functionalities from an end-user perspective to ensure they function correctly and meet user requirements.

d) Diagram

This element of the project involves using flowcharts, sequence diagrams, and diagrams to create visual representations of the application's architecture, data flow, and user interface designs. The components, interactions, and functionality of the programme are helped to be clearly and systematically outlined by these diagrams. Diagrams can be used to show the logical sequence of user interactions, the method of setting off SOS alerts, the data flow between various modules or components, and the design and organization of the user interfaces in the context of the women's safety application. For developers, designers, and users to understand and visualize the project scope, requirements, and implementation details, diagrams are useful communication tools. Diagrams also help with documentation since they serve as a reference for upcoming development, maintenance, and updates.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

The demand for technology to improve personal security arises from the increasing incidence of violence based on gender and women's safety concerns. Applications for smartphones that promote women's safety make use of modern advances in technology to provide location tracking, emergency support, and security precautions. This review delves into the development of women's safety applications using Android Studio, focusing on technological frameworks, functionalities, user interface design, and the impact of these applications on user safety.

2.2 Public Safety for Women

Urban areas, while being centers of social activities, often pose safety risks for women who may experience crime in these settings. Ensuring that women feel safe and have effective mobility in cities is vital for their quality of life [1]. Environmental elements and the perception of safety play crucial roles in making public spaces appear secure [1]. Sexual harassment on public transportation is a common issue for women, exacerbated by insufficient punitive measures and a lack of gender-sensitive approaches by authorities. This harassment limits women's freedom and negatively affects their education, career, and social engagements [2]. Public harassment remains inadequately addressed despite the attention given to violence in private settings. Although not all men are viewed as threats, societal narratives often frame all women as potential victims, leading them to avoid areas perceived as dangerous [3]. Women generally report higher levels of fear regarding crime in urban and

green spaces compared to men [1], which influences how they interact with and navigate these environments [4]. For example, many women feel unsafe traveling at night or early morning due to the heightened risk of harassment, assault, and other crimes.

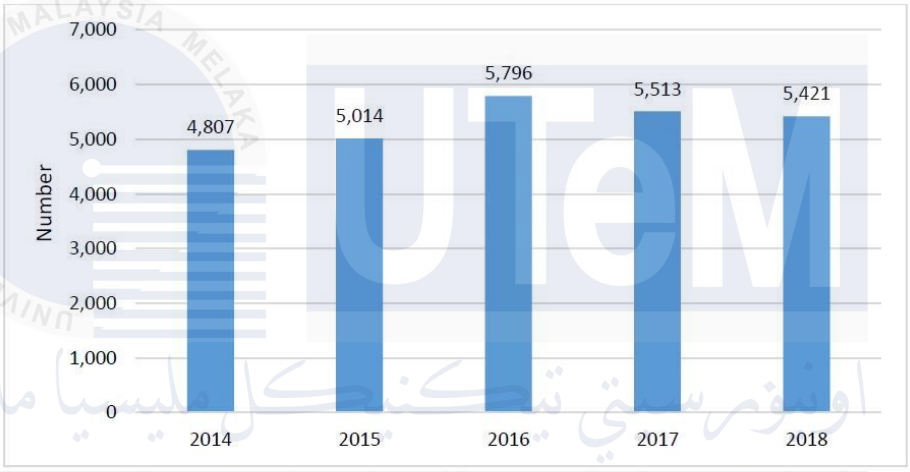
2.3 Global Perspective on Women's Safety

Women around the world continue to face significant persecution, enduring high rates of harassment, molestation, rape, kidnapping, and domestic abuse. Despite the implementation of various preventive measures, these crimes are on the rise [5]. The workplace also emerges as a critical site for sexual harassment, significantly impacting women's comfort, productivity, and overall well-being [6]. According to UN Women, one in three women globally experiences some form of sexual assault. The frequency of severe crimes against women, including kidnapping, rape, and dowry deaths, remains alarmingly high [6]. Law enforcement agencies often struggle to respond promptly to distress signals, hampered by a lack of precise information [7]. Technological advancements offer significant potential for enhancing women's safety. Various platforms and tools have been developed to address these issues; however, the effectiveness of these technologies is often hindered by a lack of action and awareness [7]. Education plays a crucial role in shaping character and is essential in the broader effort to reduce crimes against women.

2.4 Crimes Against Women in Malaysia

In Malaysia, domestic violence persists as a major problem even with legislative attempts to curb it. The groundbreaking Domestic Violence Act of 1994 (DVA) was designed to safeguard victims and bring criminal charges against perpetrators. But cultural norms and enforcement issues frequently make it ineffective. A significant number of domestic abuse instances were reported to the Malaysian police in 2020, while many of these

cases went unreported because of social stigma and fear of reprisals. According to a study, victims of domestic abuse usually face social pressure to preserve family unity, which contributes to the underreporting of domestic violence in Malaysia [8]. For many women, the situation is made worse by restricted access to shelters and support services. The situation of Malaysian women who experience assault has garnered attention once again because of recent events.



Source: Royal Malaysian Police, as obtained from Women's Aid Organisation (WAO)

Figure 2.1 Statistic number of domectic violence cases [10]

Figure 2.1 shows Number of domestic violence cases reported to the Royal Malaysian Police Malaysia, 2014 until 2018. Domestic violence instances significantly increased during the COVID-19 pandemic. The Women's Aid Organisation (WAO) reports that, in 2020, hotline calls concerning domestic abuse during the Movement Control Order (MCO) increased by 44% in comparison to the year before [9]. Increased stress, financial hardship, and being imprisoned with abusers during lockdowns were blamed for this spike [10]. Another well-known example included a 34-year-old lady who was severely beaten by her husband, resulting in being hospitalised. Public outrage over the case and extensive media coverage resulted in demands for improved victim protection and stricter enforcement

of domestic abuse laws [10]. Furthermore, a persistent and alarming trend was highlighted by recent data from the Ministry of Women, Family, and Community Development, which showed that from January to September 2021, over 5,000 incidences of domestic abuse were documented [11].

2.5 Smartphone Efficiency in Enhancing Women's Safety

According to study [12], through the years, smartphones have undergone outstanding developments, including a range of technology that improve user safety. Figure 2.2 below shows statistic number of smartphones users in Malaysia from 2010 to 2020 and a forecast up to 2025 (in millions).

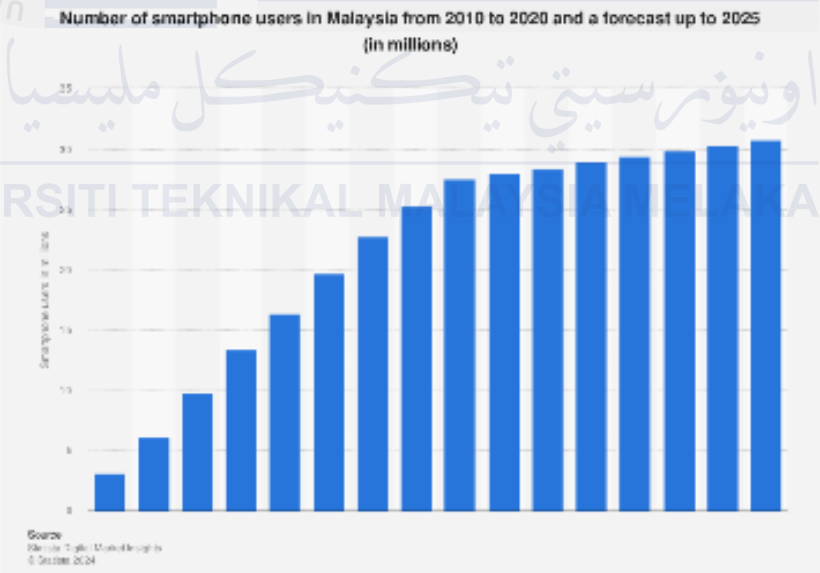


Figure 2.2 Number of smartphone users in Malaysia from 2010 to 2020

With the aid of features like GPS, location sharing, emergency alerts, and real-time tracking, women may now get assistance more quickly and effectively. Thanks to these developments, women can now immediately notify authorities and contacts if they find themselves in risky situations. Predictive safety measures and customized notifications have

been further enhanced using AI and machine learning into safety apps. To improve women's safety, numerous wearable technology and smartphone apps have been created. Applications like "bSafe," "Circle of 6," and "SaveME 999" offer functions including location sharing, audio and video recording during emergencies, and emergency contact notifications [13] [14][20]. Smart jewelry and panic buttons are examples of wearables that can be connected to smartphones to quietly transmit trouble alerts. These are gadgets meant to give women discreet and quick methods to ask for assistance. Numerous safety measures, such as panic buttons, fake call features, and automated recording capabilities, are built into smartphones. To gather evidence and discourage possible attackers, these features might be turned on during vital moments. Users can also establish safe zones with geo-fencing technology, and when these borders are crossed, warnings are sent to pre-specified contacts. These safeguarding technologies establish a virtual safety net that offers comfort and immediate assistance [15].

2.6 Existing Emergency Applications

There are many comparable emergency apps available on the market today [8], and they all have the same objective in mind: protecting the user. Each of them stands out from the others due to its uniqueness, features, usefulness, and competitive advantages. The apps that are going to be compared are mentioned below.

2.6.1 bSafe

Figure 2.3 bSafe

Strength:

bSafe offers a comprehensive suite of safety features, including an SOS button, live audio and video streaming, a fake call option, and a follow-me feature [17]. These capabilities ensure real-time support and evidence gathering during emergencies, making it a robust tool for enhancing user safety and providing immediate assistance.

Weakness:

However, the advanced functionalities of bSafe can lead to significant battery drain, especially due to continuous GPS tracking and live streaming. Additionally, the app raises privacy concerns as it involves extensive data sharing and real-time location tracking, which might be seen as invasive by some users.

2.6.2 Circle of 6

Figure 2.4 Circle of 6

Strength:

Circle of 6 excels in simplicity and speed, providing quick access to six trusted contacts and pre-set messages [17]. This design allows for rapid communication during emergencies, enabling users to summon help quickly without the need for detailed typing, thus enhancing the app's effectiveness in urgent situations [19].

Weakness:

The app, however, lacks more advanced safety features such as live streaming and geofencing, limiting its functionality. Furthermore, it does not offer real-time location tracking, which can be crucial in emergencies, thereby reducing its overall effectiveness compared to more feature-rich applications.

2.6.3 SaveME 999

Figure 2.5 SaveME 999

Strength:

SaveME 999 directly connects users to emergency services (999 in Malaysia) and shares their location with responders, ensuring prompt and accurate assistance. Its support for multiple languages and storage of critical medical and personal information enhances its usability and effectiveness in providing timely help [20][21].

Weakness:

Despite its strengths, SaveME 999 is geographically limited to Malaysia, which reduces its usefulness for international users. Its reliance on the efficiency and response times of local emergency services also means that its effectiveness can vary significantly, potentially delaying critical assistance.

2.7 Summary

Literature review involves examining related projects to enhance the application's effectiveness by comparing various studies. Each study offers unique solutions, operational methodologies, implementation strategies, and system components. By analyzing these aspects, the best practices and innovative approaches can be identified. Table 2.1 shows the comparison of each application.

Table 2.1 Comparison of Applications

Application	Features	Purpose
bSafe	Location sharing, audio and video recording during emergencies, emergency contact notifications	Provides real-time assistance and evidence collection
Circle of 6	Quick communication with trusted contacts, GPS location sharing	Ensures quick help from trusted circle in emergencies
SaveME 999	Location sharing, emergency contact notifications, integration with local emergency services	Facilitates rapid response from emergency services

CHAPTER 3

METHODOLOGY

3.1 Introduction

This project employs the Agile SDLC (Software Development Life Cycle) methodology, focusing on delivering high-quality software efficiently and cost-effectively. The project involves collecting crime rate data from three different cities and utilizing GPS technology to identify unsafe and safe areas for a mobile application aimed at enhancing women's safety. The Agile SDLC is ideal for projects requiring iterative development and rapid adjustments.

3.2 Flow Process of the Methodology

The methodology follows a structured process depicted in the flowchart, outlining the development stages for the personal safety mobile application. The process is divided into three main stages: Research Planning and Analysis, Developing and Designing, and Testing and Evaluation. These stages correspond to the Agile SDLC phases and are described below. Figure 3.1 shows the estimation general process flow project's working system to better comprehend the project constructure flow.

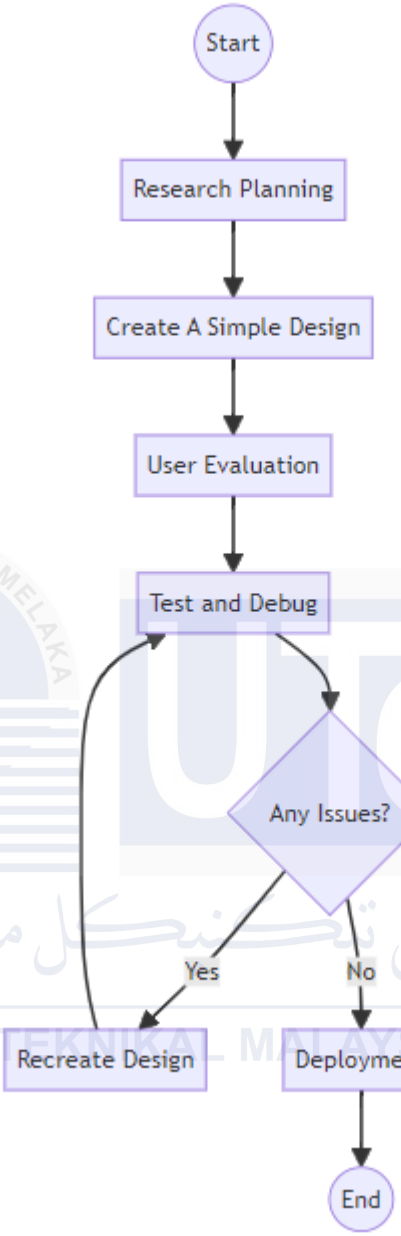


Figure 3.1 Estimation general process flow

Objective I: Research Planning and Analysis

Firstly, the research on violence against women and sexual harassment was carried out thoroughly. Secondly, to understand the requirements and concerns of the present, surveys and studies on women's safety in Malaysia are conducted. Additionally, Android was chosen over iOS because of its larger user base in Malaysia, Linux-based architecture,

and open-source nature. Furthermore, analyse current mobile safety apps and note their advantages and disadvantages to guide the creation of the new app. Moreover, assist with app development, collect relevant information from a variety of sources, such as Malaysian women's aid organisations. The project's viability and possible hazards were evaluated to make sure it is implemented successfully.

Objective II: Developing and Designing

To set up the development environment, download and configure Android Studio, the official IDE for Android app development, which supports coding in XML, C++, and Java. For prototype development, begin with a quick design to create a simple prototype, focusing on essential safety features such as location sharing and SOS mode. Furthermore, for design specification, transform software specifications into a detailed design plan, including user interface (UI) and user experience (UX) designs. Finally, implement the design using Android Studio, developing the app's core features, and ensuring a user-friendly interface.

Objective III: Testing and Evaluation

Firstly, conduct user testing by distributing the app to a group of women to gather their opinions on its reliability and effectiveness in preventing risky situations. Next, perform bug fixing by identifying and addressing errors and shortcomings based on user feedback to ensure the software meets the original specifications. Finally, deploy the application to the production environment, making it accessible to end users.

3.3 Tools for development of Women Safety Applications.

The key tools for designing women's safety applications, with a primary focus on the Android platform. These tools, including Android Studio, Android SDK and Emulator give developers the resources needed to build effective safety apps. Android Studio is the development environment, whereas the SDK contains libraries and tools for app creation. The Emulator allows for testing on virtual devices using sensor-based level in Android Studio. Together, these tools enable developers to create user-friendly interfaces, include safety features such as location tracking and emergency alerts, and facilitate the effective deployment of women's safety applications on Android smartphones, with the goal of improving women's security and well-being.

3.3.1 Android Platform

The Android Platform is an open-source, Linux-based software stack created for a wide array of devices and form factors. Based on Linux and open-source, Android allows developers to freely use and modify it to create custom versions or new features. Figure 3.2 major components of the Android software stack.

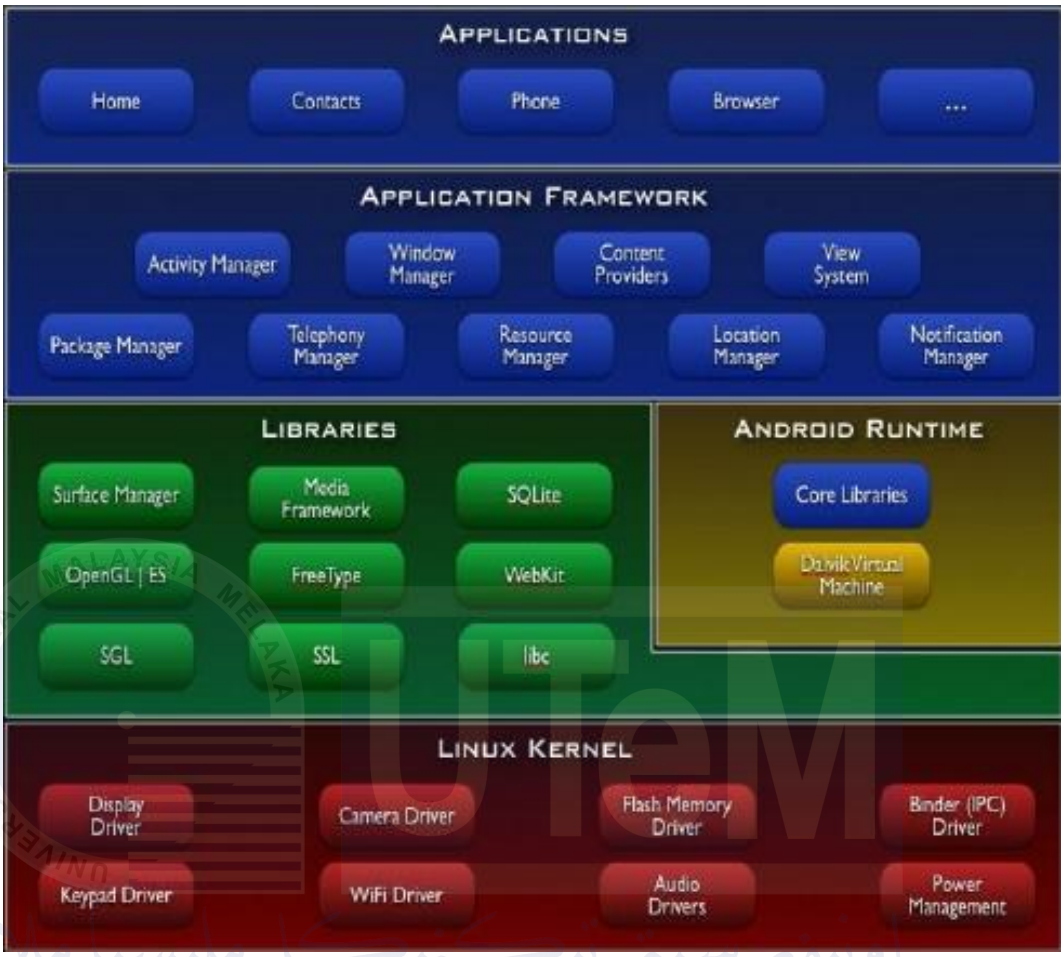


Figure 3.2 Android platform in 5 layers

3.3.2 Android Studio

Android Studio is an integrated development environment (IDE) specifically designed for developing Android applications. It is based on IntelliJ IDEA and incorporates a range of tools for syntax editing and other development tasks. Each project within Android Studio can contain one or more modules, which include resource files and source code necessary for development. Its key features include an advanced code editor with features like code completion, refactoring, and real-time error checking. The emulator allows developers to test applications on a variety of Android devices, while the layout editor

provides a drag-and-drop interface for designing user interfaces. Additionally, build tools such as Gradle automate the build process, streamlining development and deployment.

Figure 3.3 Android Studio

3.3.3 Android SDK

The Android Software Development Kit (SDK) is a crucial set of tools for developing Android applications. It provides a robust framework that includes libraries, a debugger, an emulator, and sample code to simplify the development process. It includes libraries for accessing system functions and services, a debugger for identifying and fixing code issues, an emulator enabling testing applications on virtual Android devices, and sample code providing examples to help developers understand best practices and common use cases.

3.3.4 Emulator

The Android Emulator is a powerful tool included in the Android SDK that allows developers to emulate various Android devices on their computer for testing applications. It

offers a thorough testing environment by allowing developers to simulate various screen sizes, resolutions, and hardware specifications. The emulator supports running multiple instances simultaneously, facilitating thorough testing across different device specifications. The emulator supports running multiple instances simultaneously, facilitating thorough testing across different device specifications.

3.4 Technological Review

In addition to Android tools, several foundational technologies play a crucial role in Android development:

3.4.1 Java

Java is a concurrent, class-based, object-oriented programming language designed with minimal implementation dependencies. The goal is for application developers to "write once, run anywhere" (WORA), allowing compiled Java code to execute on any Java-supported platforms without requiring recompilation. Java applications are compiled to byte code and execute on any Java virtual machine (JVM), regardless of computer architecture.

3.4.2 XML

Extensible Markup Language (XML) is a markup language used for encoding documents in a format that is both human-readable and machine-readable. In Android development, XML is primarily used for designing app layouts, specifying UI components, their arrangement, and styling. Unlike Java, which handles the application logic, XML focuses on defining the visual structure and appearance of the application's user interface.

3.5 System Architecture

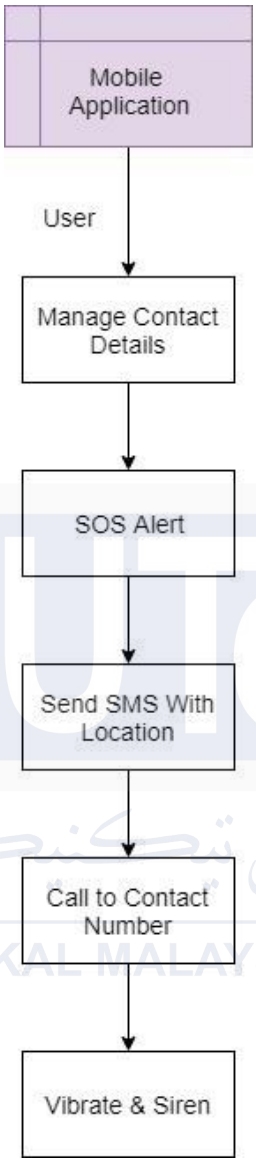


Figure 3.4 System Architecture

Figure 3.4 show system architecture for development of women safety application. The system architecture for a women's safety application developed using Android Studio includes a presentation layer with Activities, Fragments, and ViewModels for managing the user interface and interactions, such as managing contact details and triggering SOS alerts. The application layer handles core functions like sending SMS with location, making phone

calls, and activating device vibration and siren sounds, leveraging services like SMSManager, TelephonyManager, Vibrator, and MediaPlayer.

3.6 Use Case Diagram

In the development of application, use cases play a crucial role by defining user requirements, guiding the design and development process, ensuring comprehensive functionality, facilitating testing and validation, enhancing communication among stakeholders, and helping prioritize features. They provide detailed scenarios of how users will interact with the app, such as managing emergency contacts, sending SOS alerts, sharing location, making emergency calls, and triggering alerts as shown in Figure 3.5. This user-centric approach ensures that the application is intuitive, reliable, and effective in real-world situations, ultimately enhancing women's safety.

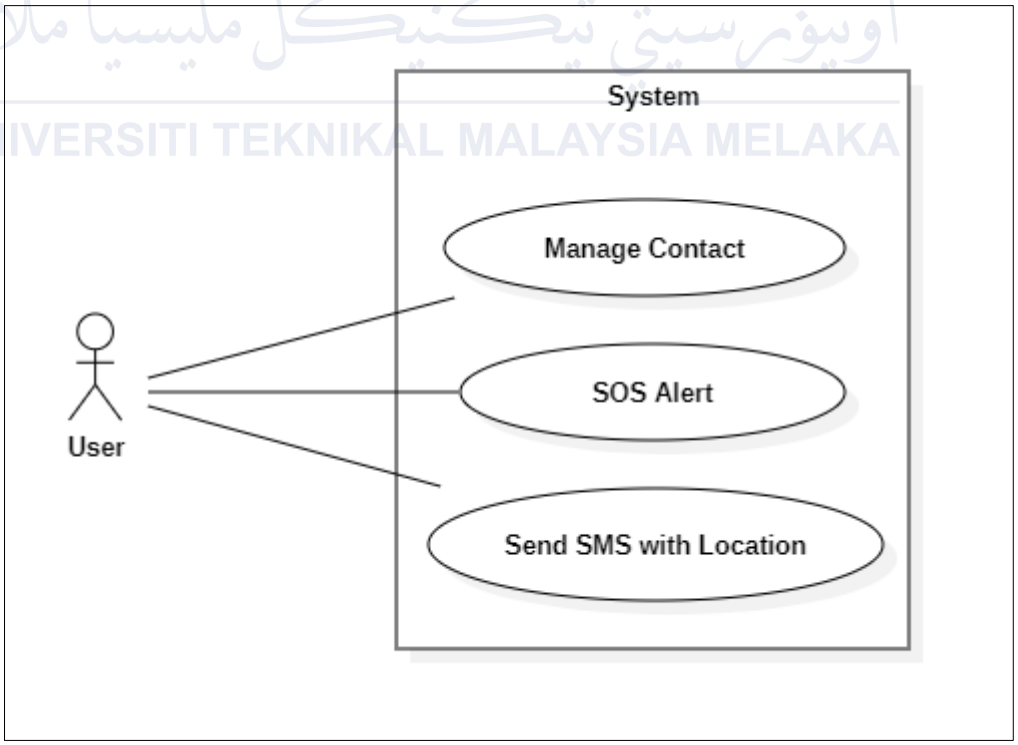


Figure 3.5 Use Case Diagram

3.7 Activity Diagram

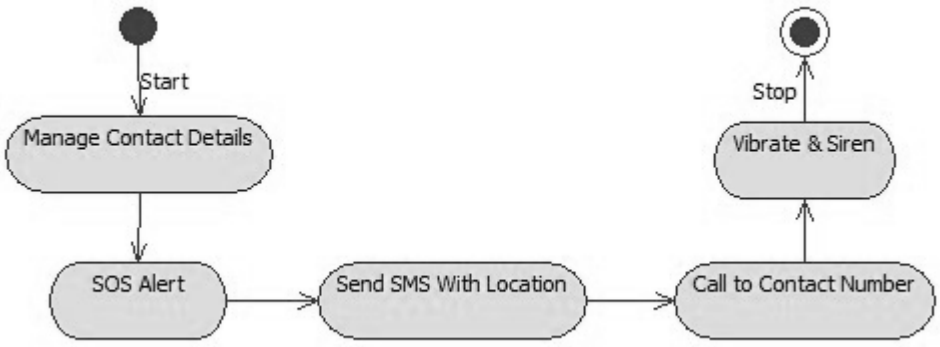


Figure 3.6 Activity Diagram of User

This activity diagram represents the workflow of application as shown in Figure 3.6. It starts with the user managing their contact details by adding or updating emergency contacts. When an SOS alert is triggered by shaking the phone, the app sends an SMS containing the user's location to the saved contacts. Simultaneously, the app initiates a call to the contact number. The workflow concludes with safety actions like activating a siren and vibration, which can be stopped manually when the emergency is resolved.

3.8 Testing Methodology

To uncover the errors, present in different phases we have the concept of levels of testing. The basic levels of testing are:

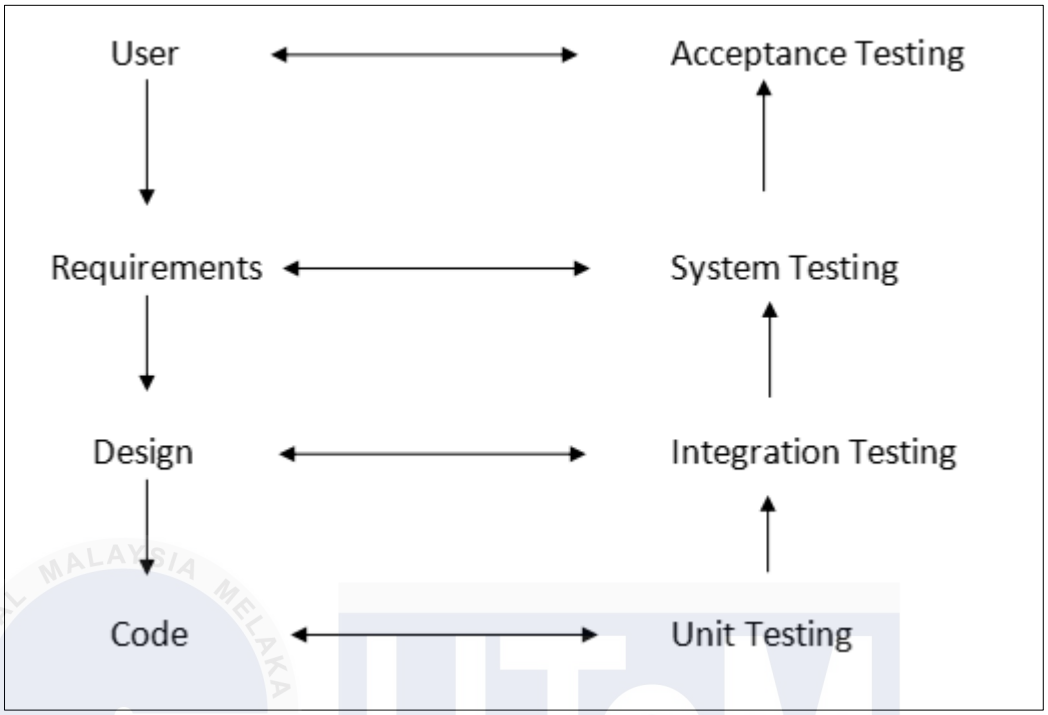


Figure 3.7 Level of Testing

3.8.1 Unit Testing

Unit testing checks the smallest part of the application, called a module, to ensure it works correctly. This testing happens during the coding stage, where each module is tested individually to verify that it produces the expected results.

3.8.2 Integration Testing

Integration testing checks how different modules work together. It ensures that data flows correctly between modules and identifies any errors in their interaction. The goal is to combine all modules into a working program and fix any issues found during testing.

3.8.3 System testing

System testing is the stage of implementation that is aimed at ensuring that the system works accurately and efficiently for live operation commences. Testing is vital to the success of the system. System testing makes a logical assumption that if all the parts of the system are correct, then the goal will be successfully achieved.

3.8.4 Validation Testing

Validation testing checks if the complete software meets the user expectations. It verifies that the application performs as required. If issues are found, they are documented and fixed. The system was validated successfully and met the requirements.

3.8.5 Output Testing

Output testing ensures the application produces the correct outputs in the required format. This includes virtual outputs such as on-screen displays and real-time outputs (notifications). The outputs were confirmed to meet user requirements, so no changes were needed.

3.8.6 User Acceptance Testing

User acceptance of a system is the key factor of the success of any system. The system under study is tested for user acceptance by constantly keeping in touch with the prospective system users at the time of development and making changes wherever required.

3.9 Summary

This project employs the Agile SDLC methodology to develop a mobile app enhancing women's safety by identifying safe and unsafe areas using crime data and GPS technology. It involves three stages: Research Planning and Analysis, Developing and Designing, and Testing and Evaluation. The app will be developed using Android Studio, chosen for its large user base and open-source nature, and will incorporate key safety features like location sharing and emergency mode. Tools such as the Android SDK and Emulator will facilitate development, testing, and real-time updates. The technological foundation includes Java for application logic and XML for UI design for local data storage.

CHAPTER 4

RESULTS AND DISCUSSIONS

4.1 Introduction

This chapter presents the results and discussions related to the development and testing of a women safety application designed using Android Studio. It covers the key features, user interface design, and the functionality of the application. The results reflect the effectiveness of the application across various devices, highlighting compatibility, usability, and performance. Additionally, this section will discuss the initial and testing results of the system, offering insights into its operational capabilities, including the integration of emergency features such as siren alerts, GPS tracking, and communication functions such as SMS sending.

4.2 Results and Analysis

This section provides a detailed examination of the functionalities, performance, and user interfaces of the women's safety application. The results are divided into subsections to cover the application's interface, additional functionalities, and performance analysis, including SMS sending durations on different devices.

4.2.1 Application Interface

Figure 4.1 Starter User Interface

Figure 4.2 User Interface after saved number

Figures 4.1 show the initial user interface of the application, where users can input emergency contact numbers. There are three fields provided: two for emergency message contacts (First Number and Second Number) and one for an emergency call contact. A "Submit" button is displayed at the bottom to save the entered details. Furthermore, Figure 4.2 shows the user interface after emergency contact numbers have been saved. The fields now display the saved numbers, indicating that the information has been successfully stored. Additionally, the interface includes a "Play Siren" toggle switch for activating the siren

feature, and the "Submit" button has been replaced with an "Edit" button to allow users to modify their saved contacts if needed.

4.2.2 Additional Functionalities

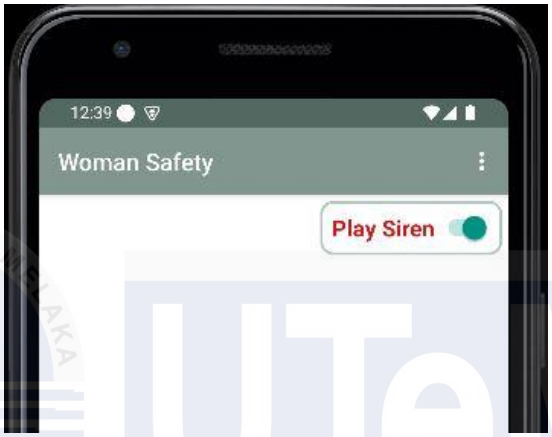


Figure 4.3 Siren Activation Interface

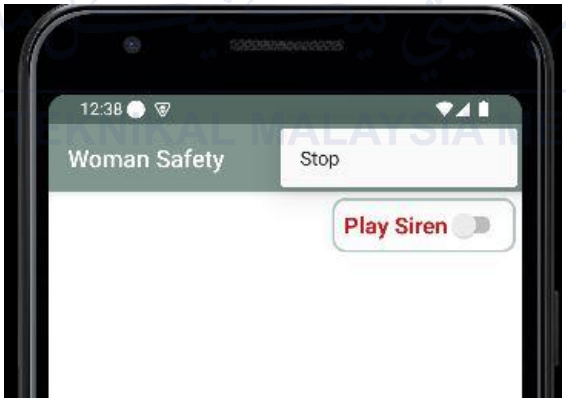


Figure 4.4 Siren Deactivation Interface

Figure 4.3 shows the app with a toggle switch next to "Play Siren". The switch is turned *on* (green), indicating the siren is active or ready to be activated. Moreover, Figure 4.4 shows the same screen, but the "Play Siren" toggle is turned *off* (grey), meaning the siren is off or has been stopped. A "Stop" button is also visible, further suggesting the siren has been deactivated.

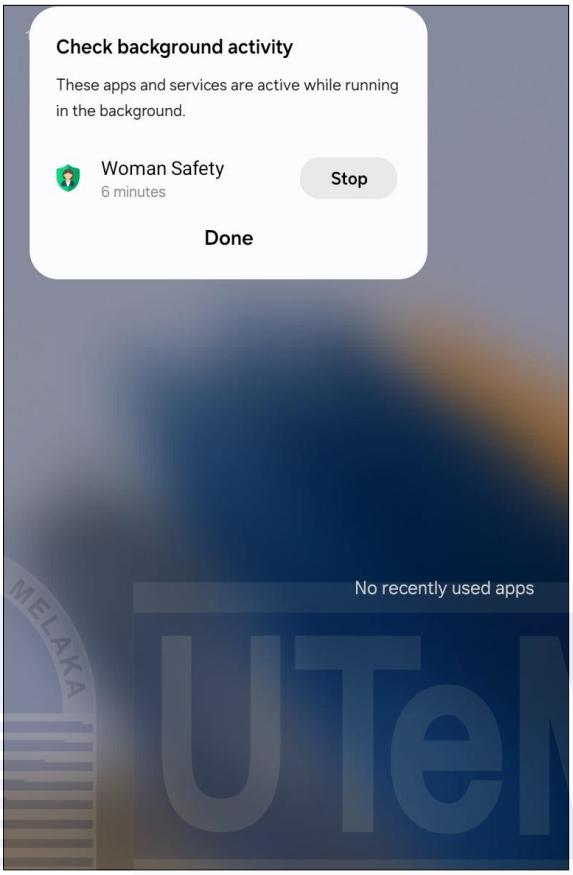


Figure 4.5 "Woman Safety" App Background Usage

This Figure 4.5 illustrates the background activity of application. The notification panel shows the app's active status, along with an option to stop the app's activity if required, providing users control over its background operation.

4.2.3 Application functionality

Figure 4.6 Emergency Call and Message

Figure 4.6 showcases the functionality of the app, including a user interface to input emergency contact numbers for messaging and calls. It also demonstrates an active emergency call screen triggered by shaking the phone.

Figure 4.7 Emergency Location Sharing via SMS

This Figure 4.7 displays the automated SMS messages sent by the "Woman Safety" app during an emergency. The messages contain a link to the user's current location, enabling responders to access their precise coordinates.

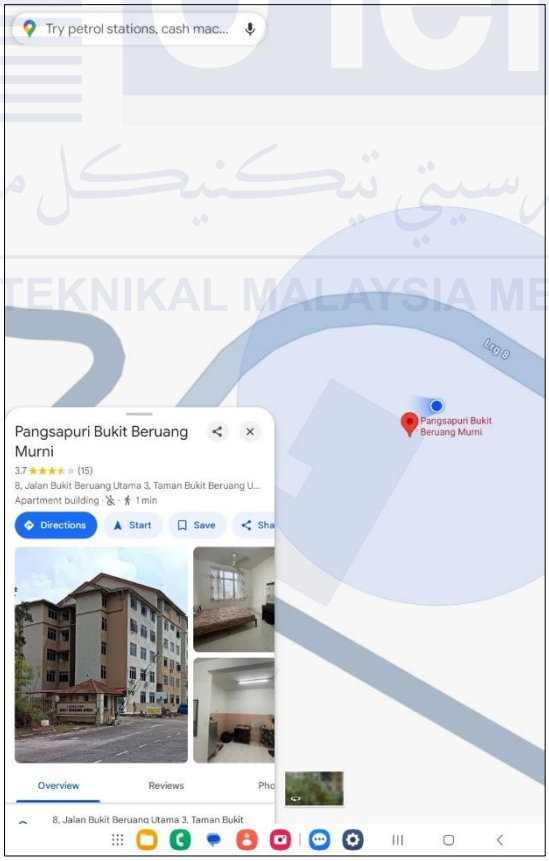


Figure 4.8 Map Display of Emergency Location from SMS Link

Figure 4.8 illustrates the map interface displaying the user's location as accessed from the SMS link sent by app. The map pinpoints the coordinates shared in the emergency message, allowing responders to locate the user quickly and accurately.

4.2.4 Analysis of SMS Sending Durations on Different Devices

Table 4.1 Results of Testing for Various Devices

Aspect	Device	Metrics	Observation
Compatibility	Samsung Galaxy Flip6	Android 13 App installed and launched successfully	Fully compatible, smooth performance with all features functioning as expected.
	Samsung Tab A8	Android 11 App installed and launched successfully. No major issues observed.	Fully compatible, occasional GPS location delays noted during testing.
Functionality Testing	Samsung Galaxy Flip6	Siren Button: Pass GPS Tracking: Pass SMS Sending: Pass	Features executed flawlessly during multiple scenarios.

	Samsung Tab A8	Siren Button: Pass GPS Tracking: Pass SMS Sending: Pass	All features worked as expected, though GPS updates showed slight delays occasionally.

Table 4.1 shows a comparative analysis of the performance of the women safety application on two devices: Samsung Galaxy Flip6 and Samsung Tab A8. The evaluation encompassed three primary aspects: compatibility, functionality, and SMS performance. Both devices demonstrated successful installation and launch of the application, indicating high compatibility. Functionality testing revealed that essential features, such as the SOS button, GPS tracking, and SMS sending, operated as intended on both devices, although minor GPS delays were observed on the Samsung Tab A8.

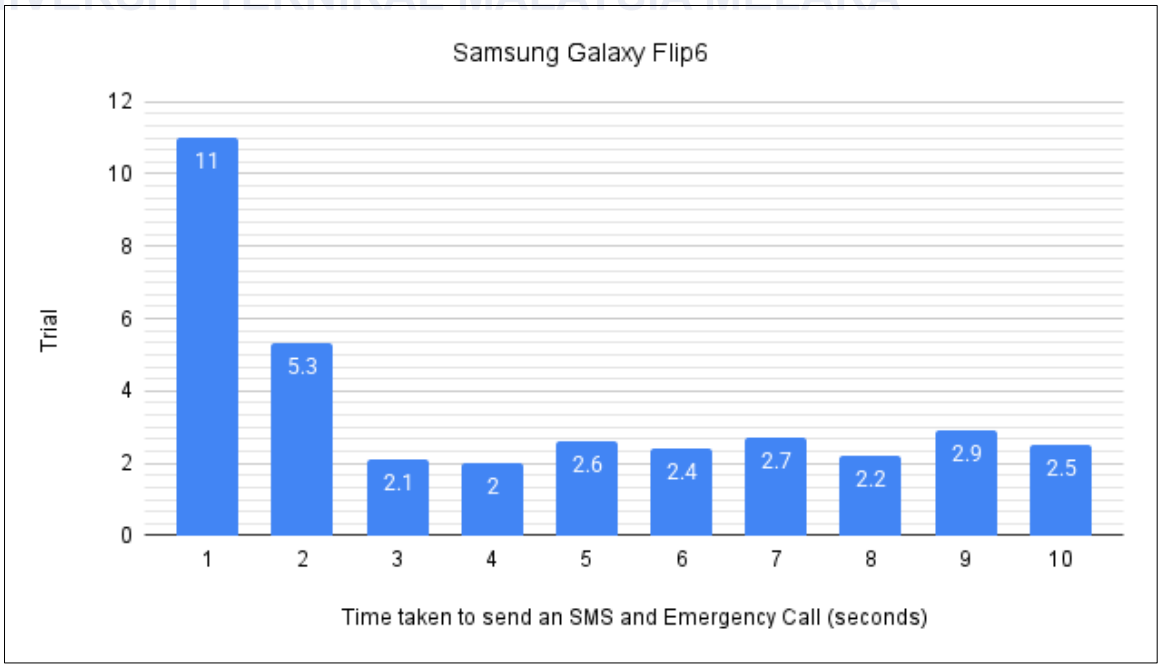


Figure 4.9 Time Taken to Send an SMS and Emergency Call Across Trials

Figure 4.9 shows the time taken to send an SMS using the **Samsung Galaxy Flip6** over 10 different trials. The results indicate that the Galaxy Flip6 performs consistently well, with the time to send an SMS ranging from 2 to 3 seconds. While there is slight variation across the trials, the performance remains generally fast, with the fastest time being 2 seconds and the slowest 11 seconds at the first trial as the app requires specific permissions (e.g., location or SMS), the first run often includes requesting and verifying those permissions, adding to the delay. This reflects the device's reliability and speed when sending SMS messages.

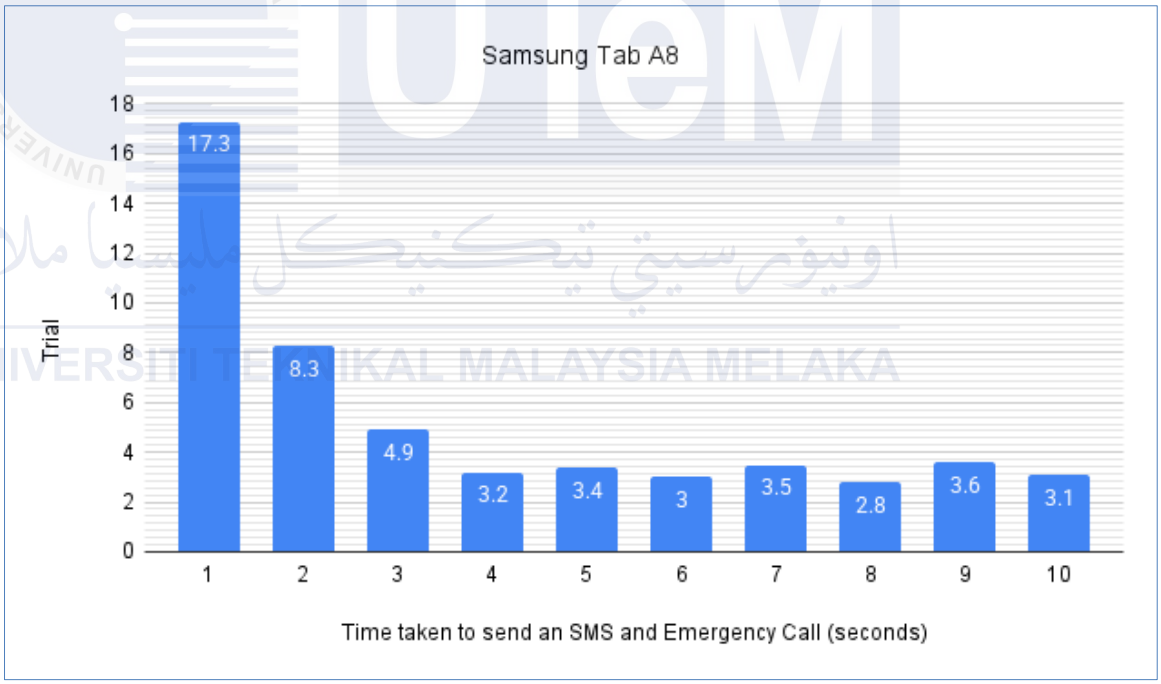


Figure 4.10 Time Taken to Send an SMS and Emergency Call Across Trials

Figure 4.10 shows the time taken to send an SMS using the **Samsung Tab A8** across 10 trials. The Tab A8 generally takes longer than the Samsung Galaxy Flip6, with times ranging from 3 to 4 seconds. While the time remains relatively stable, there is slightly more variation compared to the Samsung Galaxy Flip6. The fastest time recorded is 2.8 seconds,

and the slowest is 17.3 seconds as the same Flip6 device which first trial may take longer due to various initialization processes. This shows that, while functional, the Tab A8 is slower in sending SMS messages compared to the Samsung Galaxy Flip6.

4.2.5 Analysis of User Permission Prompts within the Application

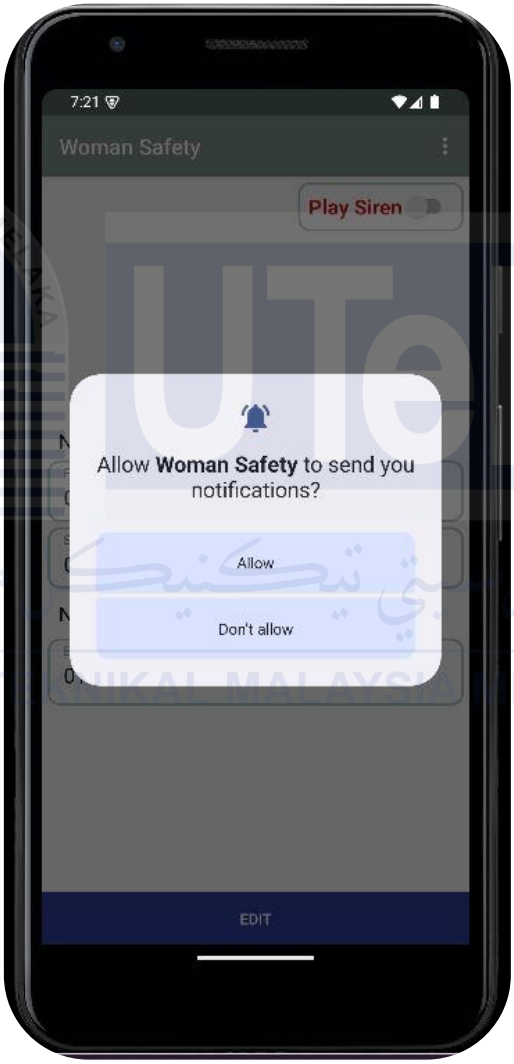


Figure 4.11 Notification Permission Prompt Interface

Figure 4.9 asks the user whether they want to allow the app to send notifications. It presents two options: "Allow" and "Don't Allow." This prompt is a standard Android system feature, ensuring users have control over enabling or disabling notifications from the application.

Figure 4.12 App Permission Request for Phone Calls

Figure 4.12 asks the user for permission to allow the app to make and manage phone calls. This is a common practice in mobile operating systems to protect user privacy.

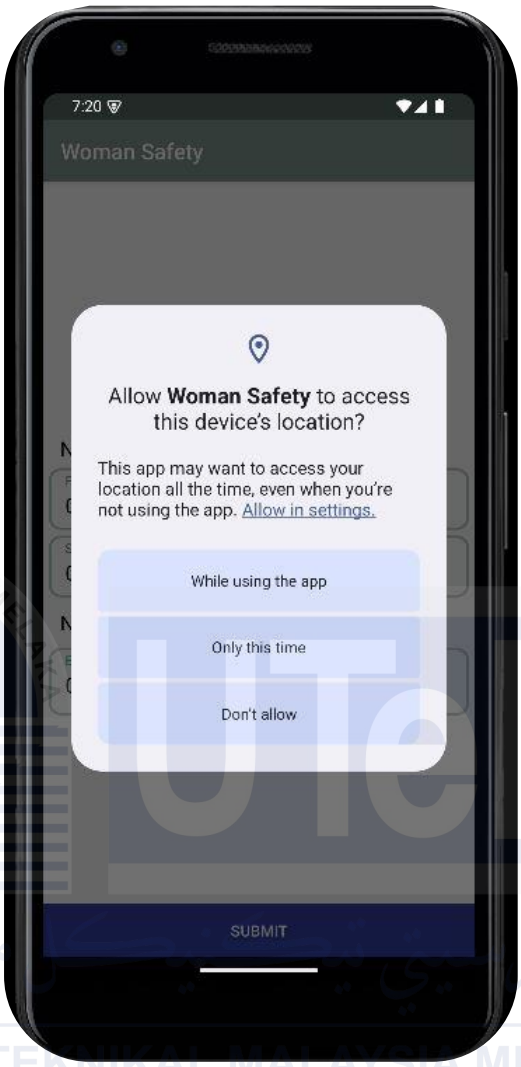


Figure 4.13 App Permission Request for Location Access

Figure 4.13 requesting the user's permission for the app to access the device's location. The dialog presents the user with three button options: "While using the app," which grants location access only when the app is actively open and in use; "Only this time," which grants location access for a single instance of app usage; and "Don't allow," which denies location access entirely.

Figure 4.14 App Permission Request for SMS Access

The Figure 4.14 dialog requests the user permission for the app to send and view SMS messages. Moreover, two buttons are presented: "Allow" to grant permission and "Don't allow" to deny it. It provides a clear way for the user to grant or deny the app ability to send and receive text messages.

Figure 4.15 App Background Execution Permission Request

Figure 4.15 requesting the user's permission for the app to run continuously in the background. The text below the question provides important context: "Allowing Woman Safety to always run in the background may reduce battery life. You can change this later from Settings > Apps." This informs the user of the potential battery impact and how to change the setting later.

4.3 Summary

This chapter highlight the successful development and testing of the "Woman Safety" application, showcasing its effectiveness in providing emergency response features such as siren activation, GPS tracking, and SMS communication. The application demonstrated high compatibility and smooth performance across devices, with essential functionalities like emergency calls and location-based SMS operating as intended. While the Samsung Galaxy Flip6 exhibited faster and more consistent performance compared to the Samsung Tab A8, both devices successfully executed all key features, with only minor delays in GPS updates and SMS delivery noted on the Tab A8. The app's user interface is intuitive, allowing users to easily input and manage emergency contacts, activate safety features, and grant permissions for notifications, location access, and background operation, ensuring user control and privacy. Overall, the application achieves its objective of providing a reliable and user-friendly safety solution for emergencies.

CHAPTER 5

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

In conclusion, the development of women's safety applications with Android Studio represents a huge step forward in using technology to solve globally concerns about women's safety. Several recommendations can enhance the "Women Safety" application. Optimizing the SMS delivery mechanism with asynchronous processes and efficient APIs can reduce delays, while integrating advanced GPS solutions like the Fused Location Provider API and offline tracking will ensure faster and more accurate location updates. Device-specific optimizations, along with power-saving techniques such as WorkManager for background tasks, can improve performance and battery efficiency across various devices. Enhancing permission management with transparent onboarding tutorials can build user trust, and UI/UX refinements, such as intuitive layouts, quick-access buttons, and feedback mechanisms, will create a more seamless experience. Additionally, incorporating features like real-time tracking, community alerts, and pre-recorded voice messages can expand functionality, while broader testing across diverse environments ensures compatibility and reliability. Multilingual support and culturally relevant adaptations can increase adoption, particularly in Malaysia, and futureproofing through regular updates, user feedback integration, and leveraging AI for predictive safety alerts will ensure long-term impact and effectiveness. These enhancements collectively strengthen the app’s potential to empower women and provide reliable safety solutions. Still, research results show that technology has the potential to be a valuable tool in preventing gender-based violence and ensuring women's safety around.

5.2 Recommendations

To further improve the effectiveness and usability of the women safety application, several enhancements are suggested. These recommendations focus on optimizing performance, expanding functionality, and ensuring broader accessibility to make the application more reliable and impactful for users: -

i. Optimized Performance: Implement asynchronous processes and APIs to minimize delays in SMS delivery and emergency call.
ii. Advanced GPS Integration: Use solutions like the Fused Location Provider API for precise and efficient location tracking, even in offline modes.
iii. Cultural Adaptations: Add multilingual support and tailor features to align with local cultural contexts, especially for Malaysian users.

5.3 Project Potential

The women safety application has the potential to become a game-changer in addressing safety concerns. With the ability to scale and evolve through regular updates and user feedback, it can cater to a broad audience, empowering women and providing peace of mind. By fostering collaborations with local authorities and NGOs, the app can expand its outreach and offer more comprehensive safety solutions. Furthermore, as technology progresses, the integration of cutting-edge advancements like wearable devices and IoT (Internet of Things) can further enhance its utility.

REFERENCES

[1] V. Ceccato, "Women's victimisation and safety in transit environments," *Crime Prevention and Community Safety*, vol. 19, no. 3–4, pp. 163–167, Sep. 2017, doi: https://doi.org/10.1057/s41300-017-0024-5.

[2] F. Vera-Gray and L. Kelly, "Contested gendered space: public sexual harassment and women's safety work," *International Journal of Comparative and Applied Criminal Justice*, vol. 44, no. 4, pp. 1–11, Mar. 2020, doi: https://doi.org/10.1080/01924036.2020.1732435.

[3] J. Harrison, "Societal narratives and perceived threats," Journal of Social Psychology, vol. 42, no. 3, pp. 123-135, May 2012.

[4] N. Bajwa et al., "Fear of Crime in Urban and Green Spaces," IEEE Transactions on Social Sciences, vol. 12, no. 4, pp. 567-578, Dec. 2018.

[5] R. Pavitra and S. Karthikeyan, "Survey on womens safety mobile app development," Mar. 2017, doi: https://doi.org/10.1109/iciiecs.2017.8276048.

[6] S. V. Lakshmi, Prathiban, and P. Kumar, "Design and Implementation of Women Safety System," *International Journal for Modern Trends in Science and Technology*, vol. 6, no. 6, pp. 186–190, Jun. 2020, doi: https://doi.org/10.46501/ijmtst060634.

[7] M. S. Farooq, A. Masooma, U. Omer, R. Tehseen, S. A. M. Gilani, and Z. Atal, "The Role of IoT in Woman's Safety: A Systematic Literature Review," *IEEE Access*, pp. 1–1, 2023, doi: https://doi.org/10.1109/ACCESS.2023.3252903.

[8] S. Othman and N. A. Mat Adenan, "Domestic violence management in Malaysia: A survey on the primary health care providers," *Asia Pacific family medicine*, vol. 7, no. 1, p. 2, 2008, doi: https://doi.org/10.1186/1447-056X-7-2.

[9] heanglee, “Sexual Harassment Statistics in Malaysia,” *Women’s Aid Organisation*. https://wao.org.my/sexual-harassment-statistics (accessed Jul. 10, 2022).

[10] Y. Pey, “CRISIS ASSESSMENT Domestic Violence and the Safety of Women during the Covid-19 Pandemic CRISIS ASSESSMENT,” 2020. Available: https://penanginstitute.org/wp-content/uploads/2020/12/Domestic-Violence-Covid-19.pdf Malay Mail, "Domestic abuse statistics in Malaysia," 2021.

[11] A. Yunus, “Over 9,000 domestic violence cases since MCO 1.0 [NSTTV] | New Straits Times,” *NST Online*, Sep. 23, 2021. https://www.nst.com.my/news/nation/2021/09/729951/over-9000-domestic-violence-cases-mco-10-nsttv

[12] Statista, “Smartphone users in Malaysia 2017-2023 | Statistic,” *Statista*, 2017. https://www.statista.com/statistics/494587/smartphone-users-in-malaysia/

[13] “Top 5 Women’s Safety Apps,” *www.linkedin.com*. https://www.linkedin.com/pulse/top-5-womens-safety-apps-tapan-patel/ (accessed Jun. 10, 2024).

[14] F. S. Nokman, “MCMC set to launch SaveME 999 app next month | New Straits Times,” *NST Online*, Feb. 20, 2017. https://www.nst.com.my/news/2017/02/214079/mcmc-set-launch-saveme-999-app-next-month (accessed Jun. 10, 2024).

[15] M. Giordano, “The Best Personal Safety Devices, Apps, and Alarms,” *Wired*, Jul. 24, 2023. https://www.wired.com/story/best-personal-safety-tech/

[16] S. Walunj, A. Gupta, A. Sonone, S. Yadav, and P. Gholap, "WS-APP Women's Safety System Based on Android Application," vol. 2, no. 1, pp. 787, Nov. 2022, doi: 10.48175/IJARSCT-7562.

[17] "5 apps that could save your life," partners.24.com. https://partners.24.com/safety24/the-free-apps-that-could-save-your-life/index.html (accessed Jun. 09, 2024).

[18] By Dr Prem Jagyasi , "bSafe app – Review." Sep. 26, 2014. Accessed: May 23, 2024. [Online]. Available: https://drprem.com/tech/bsafe-app-review/

[19] "Circle of 6," *Apps Against Abuse*. https://appsagainstabuse.devpost.com/submissions/4900-circle-of-6

[20] F. S. Nokman, "MCMC set to launch SaveME 999 app next month | New Straits Times," *NST Online*, Feb. 20, 2017. https://www.nst.com.my/news/2017/02/214079/mcmc-set-launch-saveme-999-app-next-month

[21] "SaveME 999 POLIS," *savemeprov.999.gov.my*. http://savemeprov.999.gov.my/userguide/en/ (accessed May. 10, 2024).

APPENDICES

Appendix A Permissions and Features Coding (AndroidManifest.xml)

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

    <uses-permission android:name="android.permission.FOREGROUND_SERVICE"/>
    <uses-permission android:name="android.permission.CALL_PHONE" />
    <uses-permission android:name="android.permission.SEND_SMS" />
    <uses-permission android:name="android.permission.READ_PHONE_STATE" />
    <uses-permission android:name="android.permission.REQUEST_IGNORE_BATTERY_OPTIMIZATIONS" />
    <uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
    <uses-permission android:name="android.permission.INTERNET" />
    <uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
    <uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />

    <uses-permission android:name="android.permission.VIBRATE"/>

    <uses-feature android:name="android.hardware.sensor.accelerometer" />

    <application
        android:allowBackup="true"
        android:icon="@mipmap/ic_launcher"
        android:label="@string/app_name"
        android:roundIcon="@mipmap/ic_launcher_round"
        android:supportsRtl="true"
        android:theme="@style/AppTheme">
        <activity android:name=".MainActivity">
            <intent-filter>
                <action android:name="android.intent.action.MAIN" />
```

```
29
30                <category android:name="android.
   intent.category.LAUNCHER" />
31            </intent-filter>
32        </activity>
33
34        <service
35            android:name=".background.others.
   BackgroundService"
36            android:enabled="true" />
37
38        <receiver
39            android:name=".background.location.
   LocationUpdatesBroadcastReceiver"
40            android:exported="true">
41            <intent-filter>
42                <action android:name="woman.safety.
   com.background.location.
   LocationUpdatesBroadcastReceiver.
   ACTION_PROCESS_UPDATES" />
43            </intent-filter>
44        </receiver>
45
46        <receiver android:name=".background.
   MyReceiver">
47            <intent-filter>
48                <action android:name="android.intent.
   action.BOOT_COMPLETED"/>
49            </intent-filter>
50        </receiver>
51
52    </application>
53
54 </manifest>
```

Page 2 of 2

Appendix B Design Coding (activity_main.xml)

```
<?xml version="1.0" encoding="utf-8"?>
<android.support.constraint.ConstraintLayout xmlns:
android="http://schemas.android.com/apk/res/android"
    xmlns:app="http://schemas.android.com/apk/res-
auto"
    xmlns:tools="http://schemas.android.com/tools"
    android:layout_width="match_parent"
    android:layout_height="match_parent"
    android:background="@color/ic_launcher_background
"
    tools:context=".MainActivity">

    <Switch
        android:id="@+id/play_switch"
        android:layout_width="wrap_content"
        android:layout_height="45dp"
        android:layout_margin="@dimen/margin_5"
        android:background="@drawable/
switch_background"
        android:text="Play Siren"
        android:textColor="@color/colorRed"
        android:textSize="19sp"
        android:textStyle="bold"
        android:visibility="gone"
        app:layout_constraintRight_toRightOf="parent"
        app:layout_constraintTop_toTopOf="parent" />

    <LinearLayout
        android:layout_width="match_parent"
        android:layout_height="wrap_content"
        android:layout_margin="@dimen/margin_5"
        android:background="@android:color/white"
        android:orientation="vertical"
        app:layout_constraintBottom_toBottomOf="
parent"
        app:layout_constraintTop_toTopOf="parent">

        <TextView
            android:layout_width="match_parent"
            android:layout_height="wrap_content"
            android:layout_marginLeft="@dimen/
```

```
36 margin_5"
37             android:layout_marginRight="@dimen/
   margin_5"
38             android:layout_marginTop="@dimen/margin_5
   "
39             android:padding="@dimen/margin_5"
40             android:text="Number's For Emergency
   Message"
41             android:textColor="@android:color/black"
42             android:textSize="19sp" />
43
44         <android.support.design.widget.
   TextInputLayout
45             android:layout_width="match_parent"
46             android:layout_height="wrap_content"
47             android:background="@drawable/
   edit_background">
48
49             <android.support.design.widget.
   TextInputEditText
50                 android:id="@+id/emer_first_number"
51                 android:layout_width="match_parent"
52                 android:layout_height="wrap_content"
53                 android:layout_marginLeft="@dimen/
   margin_5"
54                 android:layout_marginRight="@dimen/
   margin_5"
55                 android:layout_marginTop="@dimen/
   margin_5"
56                 android:background="@android:color/
   transparent"
57                 android:hint="First Number"
58                 android:inputType="number"
59                 android:maxLength="10"
60                 android:padding="@dimen/margin_5"
61                 android:textColor="@android:color/
   black"
62                 android:textSize="19sp" />
63
64         </android.support.design.widget.
   TextInputLayout>
```

```
65
66          <android.support.design.widget.
    TextInputLayout
67              android:layout_width="match_parent"
68              android:layout_height="wrap_content"
69              android:layout_marginTop="@dimen/
    margin_5"
70              android:background="@drawable/
    edit_background">
71
72              <android.support.design.widget.
    TextInputEditText
73                  android:id="@+id/emer_second_number"
74                  android:layout_width="match_parent"
75                  android:layout_height="wrap_content"
76                  android:layout_marginLeft="@dimen/
    margin_5"
77                  android:layout_marginRight="@dimen/
    margin_5"
78                  android:layout_marginTop="@dimen/
    margin_5"
79                  android:background="@android:color/
    transparent"
80                  android:hint="Second Number"
81                  android:inputType="number"
82                  android:maxLength="10"
83                  android:padding="@dimen/margin_5"
84                  android:textColor="@android:color/
    black"
85                  android:textSize="19sp" />
86
87          </android.support.design.widget.
    TextInputLayout>
88
89
90          <TextView
91              android:layout_width="match_parent"
92              android:layout_height="wrap_content"
93              android:layout_marginLeft="@dimen/
    margin_5"
94              android:layout_marginRight="@dimen/
```

```
94 margin_5"
95             android:layout_marginTop="@dimen/
   margin_5"
96             android:padding="@dimen/margin_5"
97             android:text="Number For Emergency Call"
98             android:textColor="@android:color/black"
99             android:textSize="19sp" />
100
101         <android.support.design.widget.
   TextInputLayout
102             android:layout_width="match_parent"
103             android:layout_height="wrap_content"
104             android:layout_marginTop="@dimen/
   margin_5"
105             android:background="@drawable/
   edit_background">
106
107             <android.support.design.widget.
   TextInputEditText
108                 android:id="@+id/emer_call_number"
109                 android:layout_width="match_parent"
110                 android:layout_height="wrap_content"
111                 android:layout_marginBottom="@dimen/
   margin_5"
112                 android:layout_marginLeft="@dimen/
   margin_5"
113                 android:layout_marginRight="@dimen/
   margin_5"
114                 android:layout_marginTop="@dimen/
   margin_5"
115                 android:background="@android:color/
   transparent"
116                 android:hint="Enter Number for Call"
117                 android:inputType="number"
118                 android:maxLength="10"
119                 android:padding="@dimen/margin_5"
120                 android:textColor="@android:color/
   black"
121                 android:textSize="19sp" />
122
123         </android.support.design.widget.
```

```
123 TextInputLayout>
124
125     </LinearLayout>
126
127     <Button
128         android:id="@+id/buttonSubmit"
129         android:layout_width="match_parent"
130         android:layout_height="wrap_content"
131         android:background="@color/colorPrimaryDark"
132         android:text="Submit"
133         android:textColor="@android:color/white"
134         app:layout_constraintBottom_toBottomOf="
    parent" />
135
136     <Button
137         android:id="@+id/buttonEdit"
138         android:layout_width="match_parent"
139         android:layout_height="wrap_content"
140         android:background="@color/colorPrimaryDark"
141         android:text="Edit"
142         android:textColor="@android:color/white"
143         android:visibility="gone"
144         app:layout_constraintBottom_toBottomOf="
    parent" />
145
146     <Button
147         android:id="@+id/buttonSave"
148         android:layout_width="match_parent"
149         android:layout_height="wrap_content"
150         android:background="#50AC53"
151         android:text="Save"
152         android:textColor="@android:color/white"
153         android:visibility="gone"
154         app:layout_constraintBottom_toBottomOf="
    parent" />
155
156
157 </android.support.constraint.ConstraintLayout>
```

Appendix C SensorManager Coding (MainActivity.java)

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
1  package woman.safety.com;
2
3  import android.Manifest;
4  import android.app.AlarmManager;
5  import android.app.PendingIntent;
6  import android.app.job.JobScheduler;
7  import android.content.Context;
8  import android.content.DialogInterface;
9  import android.content.Intent;
10 import android.content.IntentSender;
11 import android.content.pm.PackageManager;
12 import android.hardware.Sensor;
13 import android.hardware.SensorEvent;
14 import android.hardware.SensorEventListener;
15 import android.hardware.SensorManager;
16 import android.media.MediaPlayer;
17 import android.net.Uri;
18 import android.os.Build;
19 import android.os.PowerManager;
20 import android.provider.Settings;
21 import android.support.annotation.NonNull;
22 import android.support.annotation.Nullable;
23 import android.support.annotation.RequiresApi;
24 import android.support.design.widget.Snackbar;
25 import android.support.v4.app.ActivityCompat;
26 import android.support.v4.content.ContextCompat;
27 import android.support.v7.app.AlertDialog;
28 import android.support.v7.app.AppCompatActivity;
29 import android.os.Bundle;
30 import android.text.TextUtils;
31 import android.util.Log;
32 import android.util.Pair;
33 import android.view.Menu;
34 import android.view.MenuItem;
35 import android.view.View;
36 import android.widget.Button;
37 import android.widget.CompoundButton;
38 import android.widget.EditText;
39 import android.widget.Switch;
40
41 import com.google.android.gms.common.api.
```

Page 1 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
41 PendingResult;
42 import com.google.android.gms.common.api.
   ResultCallback;
43 import com.google.android.gms.common.api.Status;
44 import com.google.android.gms.location.
   FusedLocationProviderClient;
45 import com.google.android.gms.location.
   LocationRequest;
46 import com.google.android.gms.location.
   LocationServices;
47 import com.google.android.gms.location.
   LocationSettingsRequest;
48 import com.google.android.gms.location.
   LocationSettingsResult;
49 import com.google.android.gms.location.
   LocationSettingsStates;
50 import com.google.android.gms.location.
   LocationSettingsStatusCodes;
51
52 import woman.safety.com.background.Helper;
53 import woman.safety.com.background.PreferenceManager;
54 import woman.safety.com.background.others.
   BackgroundService;
55
56
57 public class MainActivity extends AppCompatActivity
   implements SensorEventListener {
58
59     public static final int PERMISSION_REQUEST_CODE
    = 111;
60     public static final int BATTERY_OPTIMIZATION_CODE
    = 112;
61     private static final int GPS_DAILOG = 113;
62     private static final String TAG = "MainActivity";
63     private boolean isBattery = false;
64
65     //Views
66     private Button btnSubmit, btnEdit, btnSave;
67     private EditText firstMsgNumber, secondMsgNumber
   , callNumber;
68
```

Page 2 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
69     //Sensor Listener
70     private static final int SHAKE_THRESHOLD = 3500;
71     private static final int MAIN_COUNT = 3;
72     private SensorManager senSensorManager;
73     private Sensor senAccelerometer;
74     private long lastUpdate = 0;
75     private float last_x, last_y, last_z;
76     private int thresholdCount = 1;
77     private FusedLocationProviderClient
   mGoogleApiClient;
78
79     private Switch aSwitch;
80     private MediaPlayer mediaPlayer;
81
82     @Override
83     protected void onCreate(Bundle
   savedInstanceState) {
84         super.onCreate(savedInstanceState);
85         setContentView(R.layout.activity_main);
86
87         //Initialize View
88         firstMsgNumber = (EditText) findViewById(R.
   id.emer_first_number);
89         secondMsgNumber = (EditText) findViewById(R.
   id.emer_second_number);
90         callNumber = (EditText) findViewById(R.id.
   emer_call_number);
91         aSwitch = (Switch) findViewById(R.id.
   play_switch);
92
93         btnSubmit = findViewById(R.id.buttonSubmit);
94         btnSave = findViewById(R.id.buttonSave);
95         btnEdit = findViewById(R.id.buttonEdit);
96
97         mGoogleApiClient = LocationServices.
   getFusedLocationProviderClient(this);
98
99         /**
100         * Emergency Details is Already Saved, So it
    can be Updated
101         * Else Will Add
```

Page 3 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
102         */
103         if (PreferenceManager.getDataPreference(
    MainActivity.this)) {
104
105             setVisibilityGone(btnSubmit, btnSave);
106
107             setVisibilityVisible(btnEdit, aSwitch);
108
109             setSavedValues();
110
111             StartBackgroundServices();
112
113         } else {
114
115             setVisibilityGone(btnSave, btnEdit);
116
117             setVisibilityVisible(btnSubmit);
118
119             setEnable(firstMsgNumber,
    secondMsgNumber, callNumber);
120         }
121
122         senSensorManager = (SensorManager)
    getSystemService(Context.SENSOR_SERVICE);
123         if (senSensorManager != null) {
124             Log.d(TAG, "onCreate: Sensor Set");
125             senAccelerometer = senSensorManager.
    getDefaultSensor(Sensor.TYPE_ACCELEROMETER);
126         } else {
127             Log.d(TAG, "onCreate: No Sensor Found");
128         }
129     }
130
131     private void setSavedValues() {
132         Pair<String, String> pair =
    PreferenceManager.getMessageNumber(MainActivity.this
    );
133
134         firstMsgNumber.setText(pair.first);
135         secondMsgNumber.setText(pair.second);
136
```

Page 4 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
137         callNumber.setText(PreferenceManager.
    getCallNumber(MainActivity.this));
138
139         setDisable(firstMsgNumber, secondMsgNumber,
    callNumber);
140     }
141
142     private void setVisibilityGone(View... views) {
143         for (View view : views) {
144             view.setVisibility(View.GONE);
145         }
146     }
147
148     private void setEnable(View... views) {
149         for (View view : views) {
150             view.setEnabled(true);
151         }
152     }
153
154     private void setDisable(View... views) {
155         for (View view : views) {
156             view.setEnabled(false);
157         }
158     }
159
160     private void setVisibilityVisible(View... views
    ) {
161         for (View view : views) {
162             view.setVisibility(View.VISIBLE);
163         }
164     }
165
166     @Override
167     protected void onResume() {
168         super.onResume();
169         showGPSEnableDialog();
170     }
171
172     @Override
173     protected void onPause() {
174         super.onPause();
```

Page 5 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
175 //         senSensorManager.unregisterListener(
    MainActivity.this);
176     }
177
178     @Override
179     protected void onStart() {
180         super.onStart();
181
182         if (Build.VERSION.SDK_INT < Build.
    VERSION_CODES.M) {
183             isBattery = true;
184         }
185
186         btnSubmit.setOnClickListener(new View.
    OnClickListener() {
187             @Override
188             public void onClick(View v) {
189                 if (Validate(btnSubmit)) {
190                     if (Build.VERSION.SDK_INT <=
    Build.VERSION_CODES.KITKAT) {
191                         setVisibilityGone(btnSubmit
    );
192                         setVisibilityVisible(btnEdit
    , aSwitch);
193                         setDisable(firstMsgNumber,
    secondMsgNumber, callNumber);
194                         PreferenceManager.
    saveNumbers(MainActivity.this
195                                 , firstMsgNumber.
    getText().toString(), secondMsgNumber.getText().
    toString(), callNumber.getText().toString());
196                         PreferenceManager.
    setDataPreference(MainActivity.this, true);
197
198                         invalidateOptionsMenu();
199                         StartBackgroundServices();
200                     } else {
201                         if (weHavePermission()) {
202                             if (isBattery) {
203                                 setVisibilityGone(
    btnSubmit);
```

Page 6 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
204                                 setVisibilityVisible
    (btnEdit , aSwitch);
205                                 setDisable(
    firstMsgNumber, secondMsgNumber, callNumber);
206                                 PreferenceManager.
    saveNumbers(MainActivity.this
207                                         ,
    firstMsgNumber.getText().toString(), secondMsgNumber
    .getText().toString(), callNumber.getText().toString
    ());
208                                 PreferenceManager.
    setDataPreference(MainActivity.this, true);
209
210
    invalidateOptionsMenu();
211
212
    StartBackgroundServices();
213                             } else {
214
    batteryOptimizePermission();
215                             }
216                         } else {
217                             requestPermission(
    PERMISSION_REQUEST_CODE);
218                         }
219                     }
220                 }
221             }
222         });
223
224         btnEdit.setOnClickListener(new View.
    OnClickListener() {
225             @Override
226             public void onClick(View v) {
227                 setVisibilityGone(btnEdit);
228                 setVisibilityVisible(btnSave);
229                 setEnable(firstMsgNumber,
    secondMsgNumber, callNumber);
230             }
231         });
```

Page 7 of 17

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
232
233         btnSave.setOnClickListener(new View.
    OnClickListener() {
234             @Override
235             public void onClick(View v) {
236                 if (Validate(btnSave)) {
237                     setVisibilityGone(btnSave);
238                     setVisibilityVisible(btnEdit);
239                     setDisable(firstMsgNumber,
    secondMsgNumber, callNumber);
240                     PreferenceManager.saveNumbers(
    MainActivity.this
241                             , firstMsgNumber.getText
    ().toString(), secondMsgNumber.getText().toString
    (), callNumber.getText().toString());
242                     PreferenceManager.
    setDataPreference(MainActivity.this, true);
243                 }
244             }
245         });
246
247         aSwitch.setOnCheckedChangeListener(new
    CompoundButton.OnCheckedChangeListener() {
248             @Override
249             public void onCheckedChanged(
    CompoundButton buttonView, boolean isChecked) {
250                 if(isChecked)
251                     playSiren(true);
252                 else
253                     playSiren(false);
254             }
255         });
256     }
257
258     private void playSiren(boolean playSiren) {
259         Helper.StopAll();
260         if (playSiren) {
261             mediaPlayer = MediaPlayer.create(
    MainActivity.this, R.raw.siren);
262             mediaPlayer.setLooping(true);
263             mediaPlayer.start();
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
264         } else {
265             if(mediaPlayer!=null) {
266                 if (mediaPlayer.isPlaying()) {
267                     mediaPlayer.stop();
268                     mediaPlayer.reset();
269                     mediaPlayer.release();
270                 }
271             }
272         }
273     }
274
275     private boolean Validate(View view) {
276         if (TextUtils.isEmpty(firstMsgNumber.getText
    ().toString())) {
277             Snackbar.make(view, "Please Enter Number
    ", Snackbar.LENGTH_SHORT).show();
278             firstMsgNumber.requestFocus();
279             return false;
280         } else if (firstMsgNumber.getText().length
    () < 10) {
281             Snackbar.make(view, "Please Enter Valid
    Mobile Number", Snackbar.LENGTH_SHORT).show();
282             firstMsgNumber.requestFocus();
283             return false;
284         } else if (TextUtils.isEmpty(secondMsgNumber
    .getText().toString())) {
285             Snackbar.make(view, "Please Enter Second
     Number", Snackbar.LENGTH_SHORT).show();
286             secondMsgNumber.requestFocus();
287             return false;
288         } else if (secondMsgNumber.getText().length
    () < 10) {
289             Snackbar.make(view, "Please Enter Valid
    Mobile Number", Snackbar.LENGTH_SHORT).show();
290             secondMsgNumber.requestFocus();
291             return false;
292         } else if (TextUtils.isEmpty(callNumber.
    getText().toString())) {
293             Snackbar.make(view, "Please Enter Number
    ", Snackbar.LENGTH_SHORT).show();
294             callNumber.requestFocus();
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
295             return false;
296         } else if (callNumber.getText().length() <
    10) {
297             Snackbar.make(view, "Please Enter Valid
    Mobile Number", Snackbar.LENGTH_SHORT).show();
298             callNumber.requestFocus();
299             return false;
300         } else {
301             return true;
302         }
303     }
304
305     private void StartBackgroundServices() {
306         Intent intent1 = new Intent(MainActivity.
    this, BackgroundService.class);
307         stopService(intent1);
308         if (Build.VERSION.SDK_INT >= Build.
    VERSION_CODES.O) {
309             startForegroundService(intent1);
310         }
311         else
312         {
313             startService(intent1);
314         }
315     }
316
317     private boolean weHavePermission() {
318         return (ContextCompat.checkSelfPermission(
    this, Manifest.permission.CALL_PHONE) ==
    PackageManager.PERMISSION_GRANTED
319                 && ContextCompat.checkSelfPermission
    (this, Manifest.permission.SEND_SMS) ==
    PackageManager.PERMISSION_GRANTED
320                 && ContextCompat.checkSelfPermission
    (this, Manifest.permission.ACCESS_FINE_LOCATION) ==
    PackageManager.PERMISSION_GRANTED
321                 && ContextCompat.checkSelfPermission
    (this, Manifest.permission.ACCESS_COARSE_LOCATION
    ) == PackageManager.PERMISSION_GRANTED
322                 && ContextCompat.checkSelfPermission
    (this, Manifest.permission.READ_PHONE_STATE) ==
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
322 PackageManager.PERMISSION_GRANTED);
323     }
324
325     private void requestPermission(int REQUEST_CODE
    ) {
326         ActivityCompat.requestPermissions(this, new
    String[]{Manifest.permission.CALL_PHONE
327                 , Manifest.permission.SEND_SMS,
    Manifest.permission.READ_PHONE_STATE
328                 , Manifest.permission.
    ACCESS_FINE_LOCATION, Manifest.permission.
    ACCESS_COARSE_LOCATION}, REQUEST_CODE);
329     }
330
331     @Override
332     public void onRequestPermissionsResult(int
    requestCode, @NonNull String[] permissions, @NonNull
     int[] grantResults) {
333         super.onRequestPermissionsResult(requestCode
    , permissions, grantResults);
334
335         if (requestCode == PERMISSION_REQUEST_CODE
    ) {
336             if (grantResults.length > 0
337                     && grantResults[0] ==
    PackageManager.PERMISSION_GRANTED) {
338                 batteryOptimizePermission();
339             } else {
340                 requestPermission(
    PERMISSION_REQUEST_CODE);
341             }
342         }
343     }
344
345     @Override
346     protected void onActivityResult(int requestCode
    , int resultCode, @Nullable Intent data) {
347         super.onActivityResult(requestCode,
    resultCode, data);
348         if (requestCode == BATTERY_OPTIMIZATION_CODE
    ) {
```

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
349                 switch (resultCode) {
350                     case RESULT_OK:
351                         isBattery = true;
352                         btnSubmit.performClick();
353                         break;
354                     case RESULT_CANCELED:
355                         isBattery = false;
356                         batteryOptimizePermission();
357                         break;
358                 }
359             } else if (requestCode == GPS_DAILOG) {
360                 switch (resultCode) {
361                     case RESULT_OK:
362                         break;
363                 }
364             }
365         }
366
367         public void batteryOptimizePermission() {
368             if (Build.VERSION.SDK_INT >= Build.
    VERSION_CODES.M) {
369                 Intent intent = new Intent();
370                 String packageName = this.getPackageName
    ();
371                 PowerManager pm = (PowerManager) this.
    getSystemService(Context.POWER_SERVICE);
372                 if (pm.isIgnoringBatteryOptimizations(
    packageName)) {
373                     isBattery = true;
374                     btnSubmit.performClick();
375                 } else {
376                     isBattery = false;
377                     intent.setAction(Settings.
    ACTION_REQUEST_IGNORE_BATTERY_OPTIMIZATIONS);
378                     intent.setData(Uri.parse("package:"
     + packageName));
379                     startActivityForResult(intent,
    BATTERY_OPTIMIZATION_CODE);
380                 }
381             } else {
382                 isBattery = true;
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
383         }
384
385     }
386
387     @Override
388     public void onSensorChanged(SensorEvent event) {
389         Sensor mySensor = event.sensor;
390
391         if (mySensor.getType() == Sensor.
    TYPE_ACCELEROMETER) {
392
393             if (mySensor.getType() == Sensor.
    TYPE_ACCELEROMETER) {
394                 float x = event.values[0];
395                 float y = event.values[1];
396                 float z = event.values[2];
397
398 //                  Log.d(TAG, "onSensorChanged: \n X
     = "+x+"\n Y = "+y+"\n Z = "+z);
399
400                 long curTime = System.
    currentTimeMillis();
401
402                 if ((curTime - lastUpdate) > 100) {
403                     long diffTime = (curTime -
    lastUpdate);
404                     lastUpdate = curTime;
405                     if (((int) ((curTime / 1000) %
    60) - (int) ((lastUpdate / 1000) % 60) < 10)) {
406
407                         float speed = Math.abs(x + y
     + z - last_x - last_y - last_z) / diffTime * 10000;
408
409 //                      Log.d(TAG, "onSensorChanged: \
    n Count = " + thresholdCount +"\n Speed = " + speed
    );
410
411                         if (speed > SHAKE_THRESHOLD
    ) {
412                             if (thresholdCount ==
    MAIN_COUNT) {
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
413                                 thresholdCount = 1;
414 //                          Log.d(TAG, "
    onSensorChanged: \n Speed = " + speed + " \n
    Threshold" + SHAKE_THRESHOLD);
415                             Helper.
    TaskDetailNotification(MainActivity.this, "
    Accelerometer has Changed");
416 //                          Helper.SendSms(
    BackgroundService.this);
417                         } else {
418                             thresholdCount++;
419                         }
420                     }
421                 } else {
422                     thresholdCount = 1;
423                 }
424
425                 Log.d(TAG, "onSensorChanged: \n
    Count = " + thresholdCount);
426
427                 last_x = x;
428                 last_y = y;
429                 last_z = z;
430             }
431         }
432     }
433
434     @Override
435     public void onAccuracyChanged(Sensor sensor, int
     accuracy) {
436
437     }
438
439     private void showGPSEnableDialog() {
440         LocationRequest locationRequest =
    LocationRequest.create();
441         locationRequest.setPriority(LocationRequest.
    PRIORITY_HIGH_ACCURACY);
442 //          locationRequest.setInterval(30 * 1000);
443         locationRequest.setFastestInterval(1000);//5
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
444  sec Time interval for location update
445         LocationSettingsRequest.Builder builder =
    new LocationSettingsRequest.Builder().
    addLocationRequest(locationRequest);
446         builder.setAlwaysShow(true); //this is the
    key ingredient to show dialog always when GPS is off
447
448         PendingResult<LocationSettingsResult> result
     =
449                 LocationServices.SettingsApi.
    checkLocationSettings(mGoogleApiClient.
    asGoogleApiClient(), builder.build());
450         result.setResultCallback(new ResultCallback<
    LocationSettingsResult>() {
451             @Override
452             public void onResult(
    LocationSettingsResult result) {
453                 final Status status = result.
    getStatus();
454                 switch (status.getStatusCode()) {
455                     case LocationSettingsStatusCodes
    .SUCCESS:
456 //                      senSensorManager.
    registerListener(MainActivity.this, senAccelerometer
    , SensorManager.SENSOR_DELAY_NORMAL);
457                         break;
458                     case LocationSettingsStatusCodes
    .RESOLUTION_REQUIRED:
459                         try {
460                             status.
    startResolutionForResult(MainActivity.this,
    GPS_DAILOG);
461                         } catch (IntentSender.
    SendIntentException e) {
462                             e.printStackTrace();
463                         }
464                         break;
465                 }
466             }
467         });
468     }
```

File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
469
470     @Override
471     public boolean onCreateOptionsMenu(Menu menu) {
472         getMenuInflater().inflate(R.menu.main_menu,
    menu);
473         return true;
474     }
475
476     @Override
477     public boolean onPrepareOptionsMenu(Menu menu) {
478         MenuItem menuItem = menu.findItem(R.id.
    stop_services);
479         if (PreferenceManager.getDataPreference(
    MainActivity.this)) {
480             menuItem.setVisible(true);
481             invalidateOptionsMenu();
482         } else {
483             menuItem.setVisible(false);
484             invalidateOptionsMenu();
485         }
486         return super.onPrepareOptionsMenu(menu);
487     }
488
489     @Override
490     public boolean onOptionsItemSelected(MenuItem
    item) {
491         if (item.getItemId() == R.id.stop_services
    ) {
492             showStopAlert();
493         }
494         return super.onOptionsItemSelected(item);
495     }
496
497     public void showStopAlert() {
498         AlertDialog.Builder builder = new
    AlertDialog.Builder(MainActivity.this);
499         builder.setPositiveButton("Stop", new
    DialogInterface.OnClickListener() {
500             @Override
501             public void onClick(DialogInterface
    dialog, int which) {
```


File - C:\Workspace\A135_Android Women Safety App\Women Safety\Project\WomanSafety\app\src\main\java\woman\safety\

```
502                 dialog.dismiss();
503                 stopServices();
504             }
505         });
506         builder.setNegativeButton("Cancel", new
    DialogInterface.OnClickListener() {
507             @Override
508             public void onClick(DialogInterface
    dialog, int which) {
509                 dialog.dismiss();
510             }
511         });
512         AlertDialog alertDialog = builder.create();
513         alertDialog.show();
514     }
515
516     private void stopServices() {
517         StartBackgroundServices();
518         aSwitch.setChecked(false);
519         Helper.StopAll();
520     }
521
522     @Override
523     protected void onDestroy() {
524         super.onDestroy();
525         if(mediaPlayer != null){
526             if (mediaPlayer.isPlaying()) {
527                 mediaPlayer.stop();
528                 mediaPlayer.reset();
529                 mediaPlayer.release();
530             }
531         }
532     }
533 }
534
535
```

Appendix D Gant Chart PSM1

NO	TASK	PSM1													
	WEEK	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
1	Create project title														
2	Project Title Confirmation and Registration														
3	Briefing with Supervisor														
4	Study the Project Background														
5	Drafting Chapter 1 Introduction														
6	Task WEEK 6 LOGBOOK progress evaluation 1														
7	Drafting Chapter 2 Literature Review														
8	Drafting Chapter 3 Methodology														
9	Work on Interface														
11	First Draft submission to Supervisor														
12	Task progress evaluation 2														
13	Drafting Chapter 4														
14	Submission Report to the Panel														
15	Presentation of BDP1														

Appendix E Gant Chart PSM2

No	TASK	PSM2													
	WEEK	W1	W2	W3	W4	W5	W6	W7	W8	W9	W10	W11	W12	W13	W14
1	Define app structure														
2	Frontend Development														
3	Backend Development														
4	Logbook 1 Update														
5	Testing Phase														
6	Logbook 2 Update														
7	Bug fixing and optimization														
8	Project report and documentation														
9	Prepare presentation														
10	Finalize Thesis and Update in ePSM														