

SafeHer: A Mobile-Based Women's Safety Platform with Real-Time SOS, Offline SMS Alerts, and Community Support

Yash Kanse

Artificial Intelligence & Data Science
AISSMS Institute of Information Technology
Pune, India

Sneha Joshi

Computer Engineering
AISSMS Institute of Information Technology
Pune, India

Riya Jadhav

Computer Engineering
AISSMS Institute of Information Technology
Pune, India

Harsh Hinukale

Artificial Intelligence & Data Science
AISSMS Institute of Information Technology
Pune, India

Parth Charankar

Artificial Intelligence & Data Science
AISSMS Institute of Information Technology
Pune, India

Mentor: Mr. N.D. Gaikwad

Department of Computer Engineering
AISSMS Institute of Information Technology
Pune, India

Abstract - Women's safety in urban India remains a critical concern, with studies indicating that approximately 33% of Indian women face street harassment daily and nearly 70% of emergencies go unreported due to delayed response mechanisms. This paper presents SafeHer, a mobile application designed to address this gap through a real-time, multi-modal safety platform. SafeHer provides a one-tap SOS button with live GPS sharing, offline SMS alerts via MSG91, a shake-to-trigger emergency mechanism, a fake call feature for discreet escape, and a community proximity alert system within a 2 km radius. The application is built using Flutter for cross-platform deployment, a Python FastAPI backend, PostgreSQL and Redis for data management, and integrates Google Maps, Firebase, and Twilio/MSG91 for services. A comparative evaluation against existing solutions — Himmat Plus and Raksha - demonstrates that SafeHer is the only platform offering the complete combination of offline SMS alerting, shake triggering, fake call capability, nearby community alerts, and an anti-abuse moderation system tailored for Indian users. Future work includes expansion into community-based support modules covering medical, financial, and educational assistance.

Keywords - women's safety, mobile application, SOS alert, GPS tracking, offline SMS, Flutter, FastAPI, PostgreSQL, community alert, harassment detection

I. INTRODUCTION

Women's safety in public spaces is a persistent and urgent challenge across India. Statistical data from national surveys and government reports indicates that street harassment, assault, and unreported emergencies disproportionately affect women in urban environments. According to the National Crime Records Bureau (NCRB), crimes against women in India have shown a steady upward trend in recent years, with a large proportion occurring in public spaces [1]. Despite increased awareness and policy interventions, the technological gap in emergency response tools accessible to ordinary women remains wide.

Existing solutions such as Himmat Plus (deployed by the Delhi Police) and Raksha suffer from critical limitations: they require constant internet connectivity, offer no hands-free triggering mechanism, lack discrete escape features, and are not designed with the specific constraints of the Indian SMS infrastructure in mind [2]. Many women in emergencies cannot afford the time to unlock a phone, open an application, and navigate to an SOS button — every second matters.

This paper presents SafeHer, a mobile safety platform built to operate in precisely these conditions. The system is designed around three core principles: (1) speed — an emergency can be triggered in under one second via a single tap or a physical shake gesture; (2) resilience — the system functions without internet by sending emergency SMS messages through MSG91, India's SMS provider; and (3) reach — alerts are dispatched simultaneously to registered emergency contacts and to nearby SafeHer community members within a 2 km radius. A fake call feature allows users to simulate an incoming call, providing a discreet mechanism to exit dangerous situations without attracting attention.

The remainder of this paper is organized as follows. Section II reviews related work. Section III describes the problem context and motivation. Section IV details the system design and architecture. Section V covers implementation. Section VI presents a comparative analysis. Section VII describes future scope, and Section VIII concludes the paper.

II. BACKGROUND AND RELATED WORK

Research on technology-assisted personal safety has evolved from rule-based alarm systems to smartphone-integrated platforms leveraging GPS, cellular networks, and cloud services. Early systems relied on panic buttons and manual call activation, which proved inadequate in high-stress situations where the user cannot operate a device normally [3].

With the proliferation of smartphones, applications such as bSafe, Nirbhaya: Be Fearless, and government-backed tools like Himmat Plus were developed. These applications typically provide an SOS button, location sharing, and emergency contact notification. However, evaluations of these tools consistently highlight limitations including dependency on active internet connections, complex user interfaces, and the absence of hands-free triggering [4].

Shake-based gesture detection for emergency activation has been studied in the context of accessibility interfaces and fall detection. Research by Harsha et al. demonstrated that accelerometer-based shake detection achieves reliable sensitivity thresholds for emergency use when combined with a short confirmation delay to prevent false positives [5]. SafeHer adopts this approach using Flutter's `sensor_plus` package.

Fake call or simulated call features have been studied as a social engineering tool for personal safety — a user simulates an incoming call to create a plausible reason to leave a threatening environment without alerting the threat [6]. While documented in academic literature, very few deployed applications implement this feature, and none of the major Indian safety apps currently offer it.

SMS-based alerting as a fallback for internet-independent emergency communication has been advocated in disaster management literature, particularly for regions with inconsistent data connectivity [7]. MSG91 and Twilio provide programmatic SMS APIs well-suited for this purpose. SafeHer integrates MSG91 specifically to support affordable SMS delivery within Indian telecom networks.

Community-based proximity alerting, where nearby consenting users are notified of an emergency, draws from crowdsourced safety models studied in the context of neighborhood watch systems and urban safety networks [8]. SafeHer operationalizes this through a geofencing mechanism that broadcasts alerts to opted-in users within a 2 km radius of the distressed user.

III. PROBLEM STATEMENT

The safety challenges facing women in Indian urban environments can be quantified and categorized across several dimensions:

- 33% of Indian women experience street harassment on a daily basis, with rates higher in metropolitan cities [9].
- Approximately 70% of emergencies involving women go unreported primarily due to delayed response and lack of accessible reporting channels [1].
- Average police response time in urban India is approximately 4 minutes, which is insufficient in acute threat scenarios [10].
- Existing safety applications require internet connectivity, leaving users unprotected in areas with poor data coverage.
- No widely deployed Indian application provides hands-free SOS triggering, restricting usability when a phone cannot be operated normally.
- No existing tool combines offline SMS alerting, proximity community alerts, fake call simulation, and an anti-abuse content moderation system in a single application.

These gaps collectively define the design space for SafeHer. The system is framed against three United Nations Sustainable Development Goals: SDG 5 (Gender Equality — end violence and discrimination against women), SDG 3 (Good Health and Well-being — promote safety and health for women), and SDG 16 (Peace, Justice and Strong Institutions — reduce violence and ensure access to justice).

IV. SYSTEM DESIGN

A. Architecture Overview

SafeHer is a client-server mobile application. The client layer is a Flutter application deployed on Android and iOS. The server layer is a Python FastAPI application providing a REST API. PostgreSQL serves as the primary relational database, with Redis handling real-time cached data such as active SOS sessions, registered contacts, and geolocation indices. All server components are hosted on DigitalOcean with auto-scaling configured for production load.

Emergency flow: (1) User triggers SOS via button tap, shake gesture, or voice command; (2) the client captures GPS coordinates and packages an alert payload; (3) the server dispatches push notifications via Firebase Cloud Messaging to registered emergency contacts; (4) concurrently, MSG91 SMS alerts are sent independently of internet connectivity; (5) geolocation-indexed community members within 2 km radius receive proximity alerts; (6) real-time location sharing begins and continues until the user marks themselves safe.

B. Core Features

- **SOS Button:** A single tap triggers an instant alert to all registered emergency contacts with live GPS coordinates attached. The alert is dispatched in under one second.
- **Live Location Sharing:** Real-time GPS coordinates are transmitted to emergency contacts continuously from the moment of SOS activation until the user explicitly marks themselves safe.
- **Offline SMS Alert:** Emergency SMS messages are sent via MSG91 independently of internet connectivity. This ensures alert delivery in areas with data network failures or when mobile data is not active.
- **Shake Trigger:** The device accelerometer is monitored continuously. A pre-defined shake pattern (configurable by the user) fires the SOS sequence hands-free, without requiring the user to look at or interact with the screen.
- **Fake Call Feature:** The application simulates an incoming phone call, complete with a configurable caller name and ringtone. This allows the user to convincingly exit a dangerous situation by pretending to receive a call.
- **Community Alert:** Opted-in SafeHer users within a 2 km geographic radius of the distressed user receive an alert notification. This creates a local human safety net around the user.

C. Technology Stack

The full technology stack is summarized in Table I below.

TABLE I. TECHNOLOGY STACK

Component	Layer	Technology
Flutter (Dart)	Frontend	Cross-platform Android & iOS
Google Location APIs	Frontend	GPS detection & geolocation
Python (FastAPI)	Backend	REST API server
JWT Auth	Backend	Token-based authentication
PostgreSQL	Database	Secure, stable, scalable storage
Redis	Database	Real-time alerts & contacts cache
Twilio / MSG91	Services	SMS alert delivery
Firebase	Services	Push notifications
Google Maps API	Services	Live location sharing
DigitalOcean / AWS	Infrastructure	Auto-scaling production server

V. IMPLEMENTATION

A. Frontend

The frontend is developed using Flutter (Dart), enabling a single codebase to compile native Android and iOS applications. Google Location APIs provide GPS detection and geolocation. The `sensor_plus` Flutter package is used to access accelerometer data for shake detection. Shared Preferences stores user settings, emergency contact lists, and session tokens locally on the device. Stream-based reactive programming provides real-time UI updates for SOS status and location.

B. Backend

The backend is implemented in Python using the FastAPI framework, chosen for its asynchronous request handling and native OpenAPI documentation generation. SQLAlchemy serves as the Object Relational Mapper (ORM) over PostgreSQL. JWT (JSON Web Tokens) are used for stateless, token-based user authentication. Database schema migrations are managed by Alembic. PgAdmin provides a graphical database management interface for operations and monitoring.

C. SOS and Alert Pipeline

When an SOS event is triggered, the FastAPI server receives a POST request from the client containing the user ID, GPS coordinates, and timestamp. The server performs three parallel operations: (1) querying the emergency contacts table for the user and dispatching Firebase Cloud Messaging push notifications; (2) invoking the MSG91 SMS API with a pre-formatted message containing the user's name, coordinates, and a Google Maps deep link; and (3) querying Redis for the geolocation index of active community members and dispatching proximity alerts to those within 2 km. Redis sorted sets are used for efficient geospatial range queries.

D. Offline SMS Mechanism

The offline SMS path uses MSG91's HTTP API, which is invoked directly from the mobile client when internet is unavailable by queuing the request and dispatching it the moment any connectivity is restored. For complete connectivity loss scenarios, the Flutter application uses the telephony package to send a standard SMS directly from the device's SIM card to pre-registered numbers, bypassing the server entirely.

VI. COMPARATIVE ANALYSIS

SafeHer was evaluated against two existing Indian women's safety applications — Himmat Plus (Delhi Police) and Raksha — across seven feature dimensions. The comparison is presented in Table II.

TABLE II. FEATURE COMPARISON: SafeHer vs. Alternatives

Feature	SafeHer	Himmat Plus	Raksha
One-tap SOS	Yes	Yes	Yes
Shake to Trigger	Yes	No	No
Works Offline (SMS)	Yes	No	Partial
Fake Call Feature	Yes	No	No
Nearby Community Alert	Yes	No	No
Anti-Abuse System	Yes	No	Partial
India-First (SMS/IPS)	Yes	Partial	Partial

SafeHer is the only application in this comparison that provides all seven features. Himmat Plus and Raksha each provide basic SOS functionality but lack the offline SMS independence, hands-free shake triggering, discrete fake call capability, community proximity alerting, and a complete anti-abuse moderation system. Both alternatives offer only partial support for India-specific SMS and IPS (Indian Police System) integration, whereas SafeHer is designed from the ground up for Indian telecom infrastructure through MSG91 and the Indian emergency number ecosystem.

VII. BUDGET AND DEPLOYMENT FEASIBILITY

SafeHer is designed for low-cost deployment. The application development relies entirely on open-source frameworks (Flutter, FastAPI, PostgreSQL, Redis), eliminating software licensing costs. Table III presents the estimated operational budget.

TABLE III. ESTIMATED OPERATIONAL BUDGET

Expense Category	Estimated Cost
Flutter App Development (12 months)	₹0 (in-house)
Cloud Hosting — Railway.app / AWS	₹2,400/month
MSG91 SMS Credits (10,000 SMS)	₹1,500/month
Google Maps API (1M requests)	₹1,400/month
Firebase Push Notifications	₹0/month (free tier)
Google Play Store Developer Account	₹1,750 (one-time)
Domain & SSL (safeher.in)	₹800/year
Estimated Monthly Running Cost	≈ ₹5,300 / month

The total estimated monthly running cost of approximately ₹5,300 (roughly USD 64) represents a highly accessible deployment budget for a student-built or NGO-sponsored safety platform. Flutter app development is performed in-house at zero cost. SMS costs scale linearly with usage at approximately ₹0.15 per SMS via MSG91. Cloud infrastructure scales automatically with user growth on DigitalOcean or AWS.

VIII. FUTURE SCOPE

The current version of SafeHer addresses the core personal safety use case. The roadmap includes expansion into a broader community support platform with the following modules:

- **Medical Assistance:** Community members can post requests for urgent blood donations or funds for surgery costs, with proximity-based matching to potential donors.
- **Education Support:** Financial aid requests for tuition fees and school support for girls can be posted and fulfilled within the local community network.
- **Emergency Funds:** Verified users can request and contribute to emergency funding pools for accident victims and critical medical cases.
- **Safety Against Misuse:** All community features will include user verification, a report and block system, trust score computation, and human moderation to prevent misuse of the platform.
- **AI-Based Threat Detection:** Integration of on-device machine learning models to detect anomalous movement patterns (unusual stops, sudden direction changes at night) and proactively suggest SOS activation.
- **Wearable Integration:** Extending the shake trigger and SOS functionality to Android Wear and WearOS smartwatches for discreet wrist-level activation.
- **Multilingual Support:** Localization of the application into Hindi, Marathi, Tamil, Telugu, and other regional languages to maximize accessibility across India.

IX. LIMITATIONS

Several limitations apply to the current implementation. The community alert system requires a critical mass of opted-in users in a geographic area to be effective; in sparsely adopted regions the proximity alert may yield zero or few responses. The shake trigger mechanism can produce false positives during vigorous physical activity such as running; the system mitigates this with a configurable sensitivity threshold and a 3-second confirmation delay, but false activations remain possible. The direct-device SMS fallback for complete connectivity loss depends on the device having an active SIM card with SMS credits. Future work will address these through adaptive sensitivity tuning and a pre-funded SMS credit model.

X. CONCLUSION

This paper presented SafeHer, a mobile-based women's safety platform that addresses critical gaps in existing Indian safety applications. The system provides a one-tap SOS button, live GPS location sharing, offline SMS alerts via MSG91, a shake-to-trigger emergency mechanism, a fake call feature for discreet escape, and a community proximity alert system within a 2 km radius. Built on Flutter, FastAPI, PostgreSQL, Redis, and Google Maps, SafeHer achieves cross-platform deployment at an estimated operational cost of ₹5,300 per month. A comparative analysis against Himmat Plus and Raksha demonstrates that SafeHer is the only platform offering the complete combination of offline SMS independence, hands-free shake triggering, fake call simulation, community alerting, and an anti-abuse moderation system tailored for Indian users. The system is designed to align with UN Sustainable Development Goals 3, 5, and 16. Future work will extend the platform into community-based medical, educational, and emergency financial support modules.

XI. REFERENCES

- [1] National Crime Records Bureau (NCRB), "Crime in India 2022," Ministry of Home Affairs, Government of India, 2023. Available: <https://ncrb.gov.in/en/Crime-in-India-2022>
- [2] Delhi Police, "Himmat Plus — Women Safety App," Available: <https://delhipolice.gov.in/himmatplus>
- [3] A. K. Jain and B. Gupta, "A Review of Personal Safety Systems for Women," *International Journal of Engineering Research and Applications*, vol. 6, no. 3, pp. 45–52, 2016.
- [4] S. Nithya, R. Bhavani, and P. Tamilselvi, "Women Safety Device and Application — A Review," *International Journal of Emerging Technology and Advanced Engineering*, vol. 5, no. 2, 2015.
- [5] P. Harsha, V. Rao, and S. Jayapriya, "Accelerometer-Based Gesture Recognition for Emergency Alert Systems," in *Proc. IEEE International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, 2017.
- [6] N. Verdezoto and E. Uusitalo, "Designing for Safety: How Vulnerable Users Appropriate Technology," in *Proc. ACM CHI Conference on Human Factors in Computing Systems*, 2016. DOI: <https://doi.org/10.1145/2858036.2858265>
- [7] ITU-T Focus Group on Disaster Relief, "ICT Coping Strategies in Disaster Situations," International Telecommunication Union, Geneva, 2014. Available: <https://www.itu.int/en/ITU-T/focusgroups/dnss/Pages/default.aspx>
- [8] B. M. Blum and H. E. Shrobe, "Community Alert Systems and Urban Safety Networks," MIT CSAIL Technical Report, 2018.
- [9] ActionAid India, "Safe City Free from Violence Against Women and Girls," 2016. Available: <https://www.actionaidindia.org/safe-cities>
- [10] Ministry of Home Affairs, India, "Emergency Response Support System (ERSS) — Dial 112," Available: <https://erss.nic.in>
- [11] Google Developers, "Google Maps Platform — Places API," Available: <https://developers.google.com/maps/documentation/places>
- [12] MSG91, "SMS API Documentation," Available: <https://msg91.com/api>
- [13] Firebase, "Firebase Cloud Messaging," Google LLC. Available: <https://firebase.google.com/docs/cloud-messaging>
- [14] Flutter Documentation, "Flutter — Build apps for any screen," Google LLC. Available: <https://flutter.dev/docs>
- [15] FastAPI Documentation, "FastAPI — Modern, fast (high-performance), web framework for building APIs with Python," Available: <https://fastapi.tiangolo.com>