iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** XI **Month of publication:** November 2025

DOI: https://doi.org/10.22214/ijraset.2025.75053

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com

Citizen Shield: Turning Bystanders into Lifesavers

N. Krishna Venia[1], Potnuru Keerthib[2], Ratti Pallavic[3], Naveen Kumar Patnanad[4], Gollapalli Maheendrae[5], Ungati Dileep Kumarf[6]
Department of Computer Science and Engineering -AI&ML, GMR Institute of Technology, Rajam, 532127, Andhra Pradesh, India

Abstract: ***In today's socio-technological era, safety is a key issue, particularly in times of crisis like harassment, stalking, or kidnapping. The current project suggests an extensive Safety Application that utilizes the capabilities of contemporary smartphones and real-time technologies to provide instant, multi-layered security. The application permits users to instantly send an alert with one click, which automatically conveys their live location, identity, and contextual evidence (photos/videos) to preset emergency contacts, nearby users in a 1 km radius, and police agencies using a dedicated police dashboard. For situational awareness and timely response, the app triggers the device camera and microphone, which captures key evidence automatically without any manual intervention. This system employs GPS tracking, real-time notification, to provide accurate location sharing, with a secure and friendly interface. Bulky wearable devices or outdated software, this mobile-first solution is available, light, and scalable for mass adoption. This application employs real-time communication and intelligent technology to enhance older safety tools. It assists users to rapidly receive help in situations of need by notifying surrounding people and the police. It also creates a high level of community support, such that everyone can be safe more easily. Simple to use and efficient for both urban and rural use, the app helps in making society safer for all.***

Keywords: Emergency Response, Real-Time Location Tracking System, Safety and Surveillance, Alert, Police dashboard

I. INTRODUCTION

Individual safety has come to be a vital issue in contemporary fast-paced and unpredictable society, where dangers, accidents, harassment, and violent attacks can arise unexpectedly. In such cases, swift communication and quick intervention can prove to be the difference between safety and injury. Still, conventional safety measures, like shouting for assistance or reporting to authorities, are frequently constrained by delays, accessibility, or victims' inability to respond under pressure. This emphasizes the demand for a proficient, technology-based system that guarantees prompt notification, effective communication, and effective coordination between the public and officials.

Citizen Shield is a mobile Citizen Safety SOS application that aims to take care of these issues by enabling individuals to solicit assistance instantly in one click. Upon activation, the SOS button initiates emergency notifications through SMS, calls, and emails to pre-registered contacts while at the same time sending the victim's live GPS location. In addition to individual contacts, the system also alerts other Citizen Shield users in the vicinity, facilitating prompt on-ground support and encouraging an atmosphere of community-based security.

What distinguishes Citizen Shield is its inclusion of sophisticated evidence-capturing capabilities. When SOS is triggered, the app automatically activates the device's camera and microphone to capture real-time audio-visual data. Such captures not only upgrade situational awareness but also can be used as key evidence for investigations, thus enhancing accountability and deterrence. For institutional backing, the site offers a police-specific dashboard with police access to real-time emergency alerts, users' locations, and evidence data so that they can make more informed decisions and act more quickly.

This app's focus on security and privacy. With role-based access controls, Citizen Shield guarantees sensitive data such as personal information and evidence recorded remains secure and only released to approved individuals. This protects user trust and openness in emergency management. By closing the gap between authorities, bystanders, and victims, Citizen Shield creates an interconnected environment for speed and safety. Not only is it an SOS device, but it is also an overall digital guardian which gives each citizen a voice, makes communities collaborate more effectively, and increases public safety in emergencies.

II. LITERATURE SURVEY

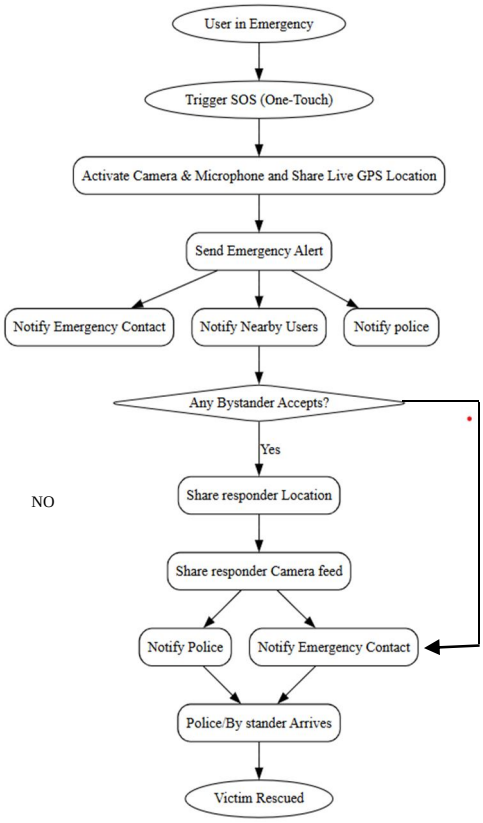
Citizen Shield facilitates touch or shake SOS activation, live GPS tracking, and secure notifications through React Native, Node.js, MongoDB, Firebase, and Google Maps API. It recorded high performance with SOS SMS at 98.6%, live location reporting at 99.2%, and shake detection at 96.7% [1]. A more advanced version of Citizen Shield includes AI-based alert filtering with Flutter/React Native, Firebase, GPS, and secure APIs.

It decreased the response time, increased user confidence in one-tap SOS, and decreased false alarms, making it more reliable [2]. A smartphone application that integrates with smart devices employs GPS, GSM, Android, accelerometer sensors, and blockchain-supported storage. It facilitated optimal evidence capture, reduced battery consumption, and enhanced privacy by means of decentralized verification [3].A women's safety Android application employs GPS, SMS service, and fake call simulation to divert attackers. Performance exhibited real-time location updates at every 5 minutes for guaranteed SOS delivery even in the absence of internet [4]. An IoT-based safety device on Arduino, GPS + GSM modules, and IoT cloud exhibited location tracking, safe-zone monitoring, and auto-audio recording. It enhanced emergency alerting and evidence capture for quick response [5].A smart device that was smartphone-integrated utilized ATmega328, GPS (Ublox NEO-6M), and GSM (SIM900) for real-time SOS triggers. It provided real-time monitoring and reliable network alerts to police, family, and friends for prompt assistance [6].A Flutter, Node.js, Socket.io, WebRTC, GPS, and Google Maps API-based medical SOS app reduced ambulance delays. It gave one-tap live location sharing and real-time doctor-patient video streaming, decreasing response time to 25–35 minutes [7].An Android SOS app based on Firebase, Google Maps API, and Dijkstra's Algorithm provided optimized routing and service within 100 meters. It provided assistance to police, firefighters, and mechanics for multi-purpose emergency management [8].A Java-developed mobile SOS system using Firebase, magnetometer sensors, and media sharing APIs provided spy camera identification, siren notification, and evidence sharing. Performance attained SOS success at 98.3%, tracking at 99.1%, and hidden camera detection at 95.4% [9].An Android application developed using SDK + Java, GPS, geofence, and Google Maps API provided immediate SMS SOS without internet. It had 97.8% SOS success and 98.4% reliability in tracking, with direct helpline calls in emergency situations [10].Elderly patients employed an IoT device that utilized Raspberry Pi, SIM900A GSM, Ublox GPS, and MPU6050 sensors with ML algorithms (KNN, SVM). It had 100% accuracy in crash detection and no false positives in collision alerts [11].A Firebase-based Android harassment monitoring system with SMS + Call modules, Google APIs, and shake sensors gave SOS alarms and unsafe zone identification. It had 93.4% reporting accuracy, 89.7% network tolerance, and 95.1% unsafe zone identification [12].A safety app that was multifunctional utilized Android, voice recognition (Google API), accelerometer, compass, and Google Maps API. It had multiple SOS initiators and concealed camera detection, registering a 0.74 ms response time and rapid initiator activation in 510–781 ms [13].A wearable IoT device with NodeMCU, ESP32-CAM, GPS, GSM, and IoT cloud offered SOS through bands, footwear, and safe-words. Performance involved alerts fired at 99–100°C or ≥105 bpm, accompanied by real-time image sharing and location information [14].A smart band with IoT capabilities constructed with ATmega2560, GPS + ESP-12E, GSM (SIM900), and self-protection devices (buzzer, shock stimulator) provided constant security. It recorded effective SMS alerts, precise GPS positioning, and 96.2% success in initiating emergency calls [15].

A. How citizen shield different from other models?

Feature	IoT Smart Band	SOS Apps	Citizen Shield
Requires Extra Hardware	Yes (GPS, GSM, Node, MCU)	No	No
Emergency Contact Alert	Yes	Yes	Yes
Police Alert	Yes (via SMS/IoT)	Yes	Yes
Nearby Bystanders Alert	No	No	Yes (Unique)
Live Camera Access	No	No	Yes (Unique)
Cost & Scalability	High (needs device)	Medium	Low, scalable easily
Rescue Speed	Slow (wait for police/family)	Medium	Fast (bystanders arrive first)

III. WORK FLOW



IV. METHODOLOGY

Layer	Technology / Tools	Purpose	Usage in Your Model (Citizen Shield)
Frontend	HTML, CSS, JavaScript	Core structure and styling of web pages	Used to design the basic layout and UI elements of your landing, login, register, and dashboard pages.
	React.js	Building dynamic and responsive UI	Manages page navigation (multi-page setup), handles real-time map updates, and integrates components like SOS button, alerts, and live location display.
	Bootstrap / Tailwind CSS	Responsive design and layout styling	Ensures UI looks good on all devices (mobile/desktop), used for buttons, cards, modals, and grid layouts in your dashboard.
Backend	Node.js (Express.js)	Handles API requests, real-time communication, and server logic	Used to build the server that receives SOS requests, triggers emails/SMS, stores data, and communicates with the frontend using REST APIs.
	Socket.io	Enables real-time alerts and nearby user communication	Allows users and police dashboard to see SOS alerts instantly; updates map markers live for nearby users.
Database (optional)	MongoDB / Firebase	Stores user data, location, and alert information	Can be used to store registered users, emergency contacts, previous SOS alerts, and location data. (You may skip this if only using live session data.)

APIs / Services	Nodemailer	Sends emergency alert emails	Automatically sends SOS alert emails to emergency contacts or police with location and user info.
	LocalTunnel / ngrok	Exposes local server for testing and communication across devices	Used during development to make your local Node.js server accessible to others for testing nearby user and email features.
	Geolocation API	Tracks user's live location for SOS and nearby alerts	Captures user's coordinates and displays on dashboard map; enables nearby user detection and route visualization.
Deployment / Tools	npm / npx	Package management and server startup	Used to install dependencies (express, socket.io, nodemailer, etc.) and start frontend/backend servers.
	GitHub	Version control	Stores and tracks project code; used for collaboration and deployment integration.
	Render / Vercel / Railway *(optional)*	Hosting web app	Can host your frontend (React) and backend (Node.js) to make your SOS app publicly accessible.
Security & Communication	HTTPS / SSL	Secure data transmission	Ensures sensitive SOS data (user info, location) is transmitted safely over the network.
	JWT / Session Handling (optional)	User authentication and data security	Can be used for login authentication, protecting routes (dashboard access), and verifying active users.

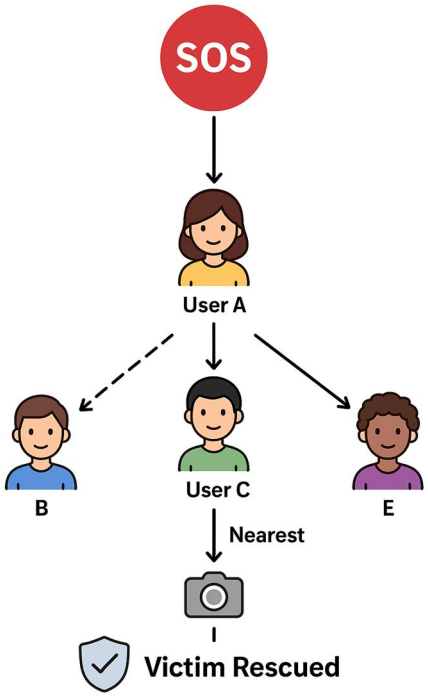
Why This Stack Is Effective?

- Lightweight and scalable – Node.js and React combination.
- Real-time performance – Socket.io enables instant alert communication.
- Security ready – HTTPS and optional JWT keep data safe.
- Developer-friendly – npm and GitHub streamline version control and deployment.

B. How Models Flows?

- User Registration & Login
  A new user registers with username, password, phone number, and emergency contact.After successful registration, the user logs in to access the SOS dashboard.
- SOS Trigger
  On pressing the SOS button, the app immediately fetches the username and live location of the victim.
- Emergency Alert Transmission
  An emergency alert message is sent to the registered emergency contacts (family, friends, or police) containing the victim's details and live location.
- Nearby User Alerts
  The app also sends the SOS alert to all nearby Citizen Shield users within a defined radius.
- Rescuer Acceptance
  If a nearby user accepts the alert, the victim's location and live camera feed are shared with the rescuer, police, and emergency contacts.
- Rescue Coordination
  Nearby users can immediately try to assist the victim while emergency contacts and police are enroute, for quicker intervention.
- Resolution
  When the victim is secured and the incident is resolved, the SOS is reported as resolved.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue XI Nov 2025- Available at www.ijraset.com



Dijkstra's Algorithm is a shortest-path finding algorithm used in graph theory. It was developed by the Dutch computer scientist Edsger W. Dijkstra in 1956.The algorithm is designed to find the shortest possible distance (or minimum cost path) between a starting node and all other nodes in a graph.It is widely used in network routing, GPS navigation, and optimization systems where you need to calculate the most efficient route between two or more points.
Imagine a city map with various roads connecting different places. Each road has a certain distance or travel time (called the *weight*).
Dijkstra's algorithm helps determine the shortest and fastest route from your current location (source node) to your destination (target node).

C. How It Works

There are five users: A, B, C, D, and E.
User A triggers an SOS.
The system identifies nearby users (B, C, D, E) within a specific distance radius using GPS.
All these users and their distances form a graph where:

- Nodes = Users' current GPS locations
- Edges = Distances or travel times between users (obtained via Google Maps API or haversine formula)
- Weight = Distance or estimated time (in meters or minutes)

Step 1: Source Node Initialization

- The system marks User A (the SOS sender) as the source node.
- Distance to A = 0.
- Distance to B, C, D, E = ∞ initially.

Step 2: Distance Calculation

- Using latitude and longitude, the system calculates real-time distance:(Assume)
  - A → B = 1.2 km
  - A → C = 0.8 km
  - A → D = 2.1 km
  - A → E = 3.0 km

Step 3: Algorithm Execution

- Dijkstra's algorithm evaluates each connection and identifies the user with the lowest cost (shortest path) from A.
- It picks C (0.8 km) as the nearest responder.

Step 4: Alert Dispatch

- The system sends the SOS alert first to User C (nearest by route).
- If C accepts, C is assigned as the primary rescuer.
- If C declines or does not respond, the algorithm automatically chooses the next shortest distance user (B) and sends the alert.

Step 5: Police & Emergency Updates

- Once a user accepts (say C), live location, route path, and victim details are instantly shared with police dashboard and emergency contacts.
- The system can also display the shortest navigation route for C to reach A quickly.

V. RESULTS

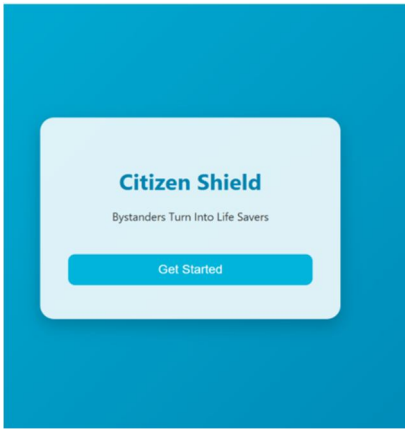


Fig 1: Landing Page

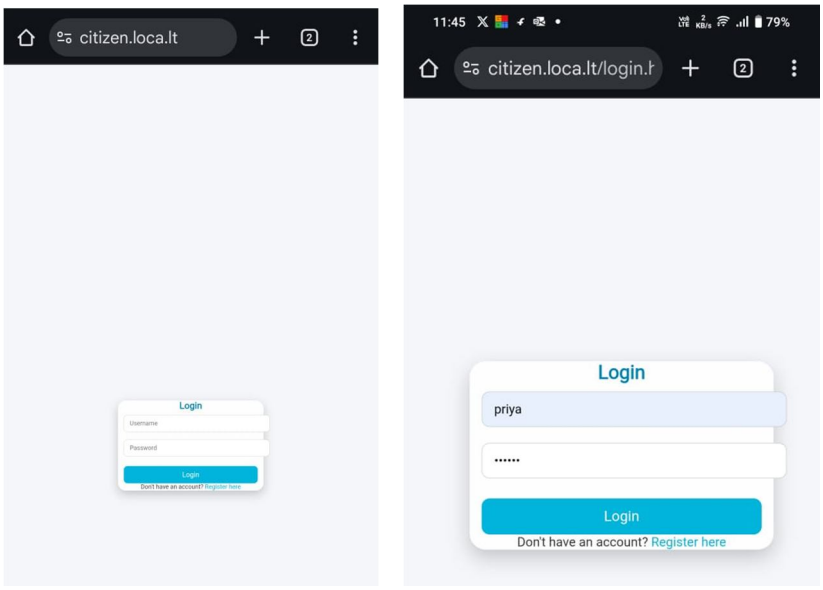


Fig 2:Login Page

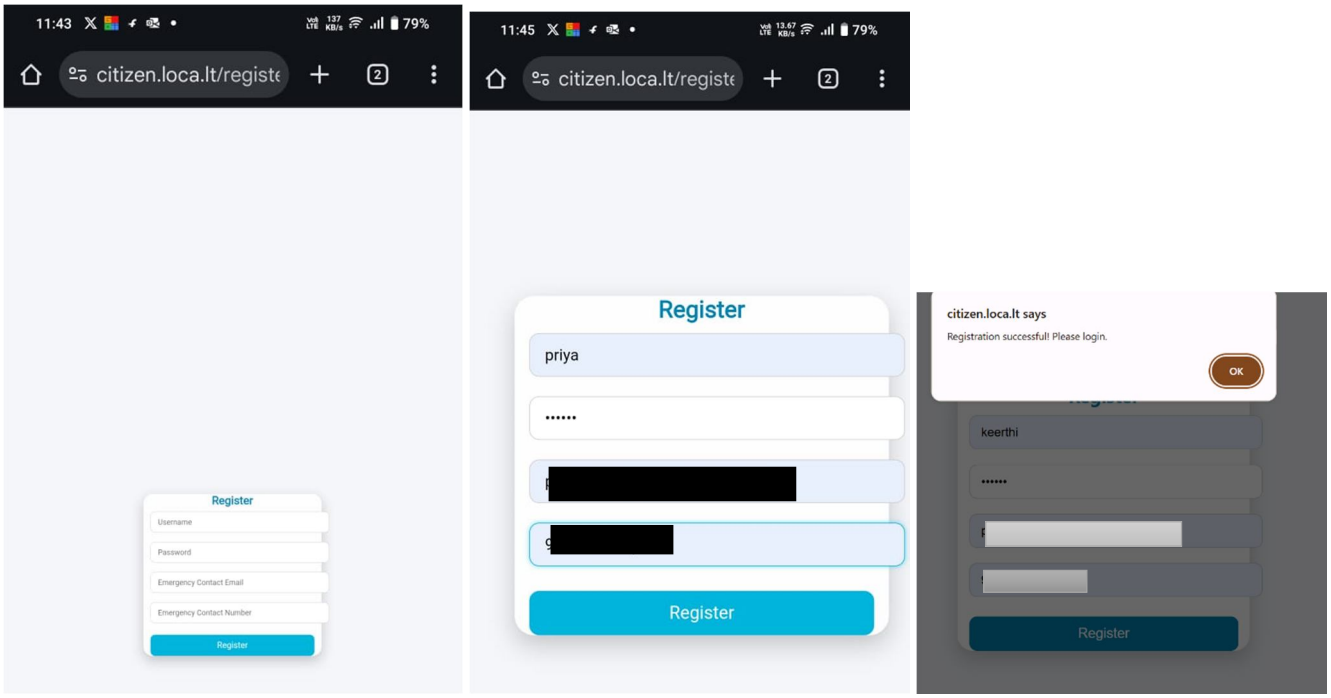


Fig 3:Registration

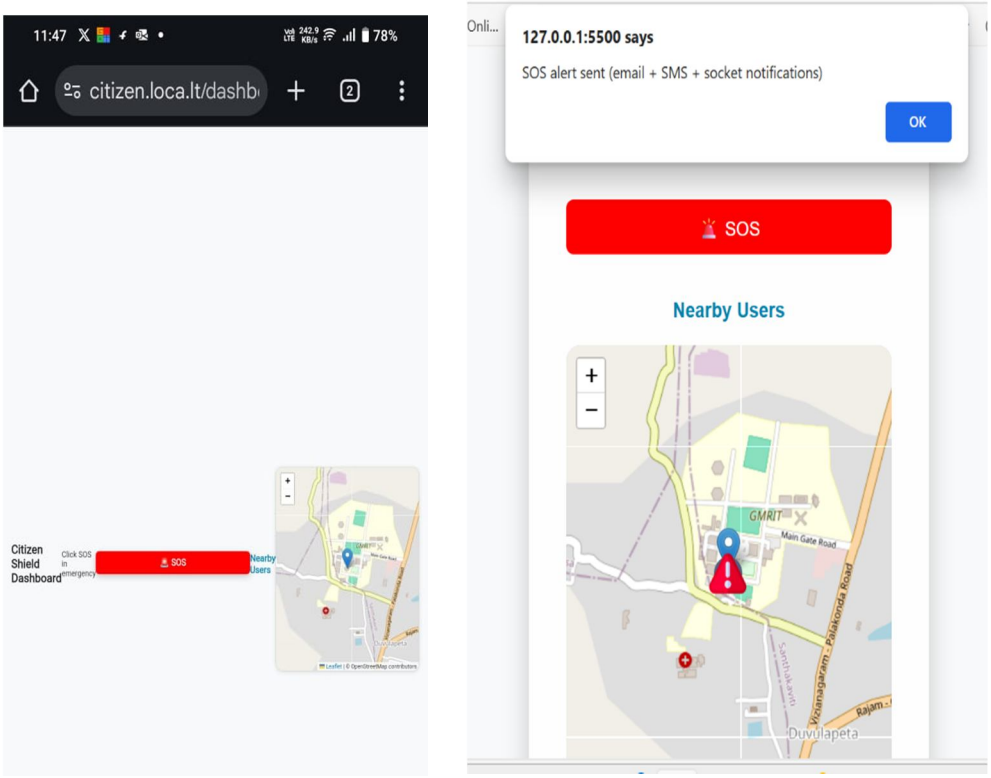


Fig 4:Dashboard if a user clicks sos(Mutiple users logged in)

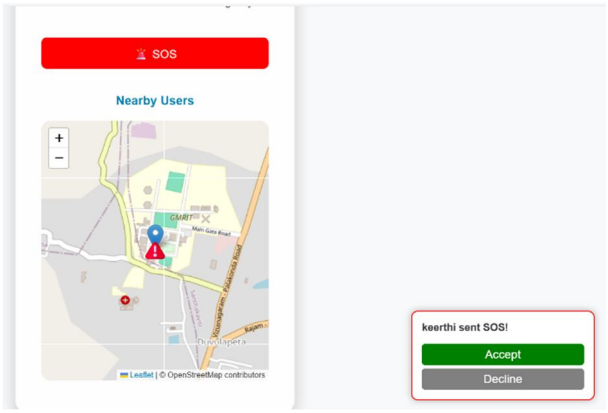


Fig 5: other users tries to save the victim by accepting

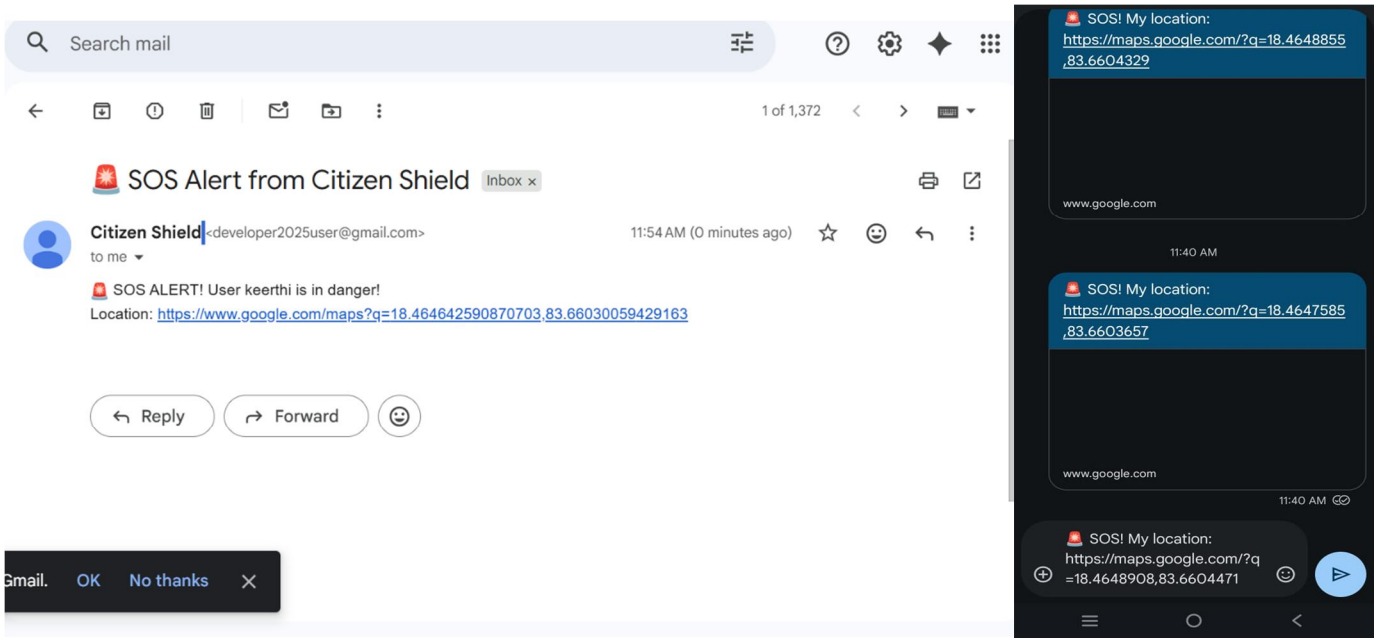


Fig 6:Mail and Sms to their Emergency contacts

VI. FUTURE SCOPE

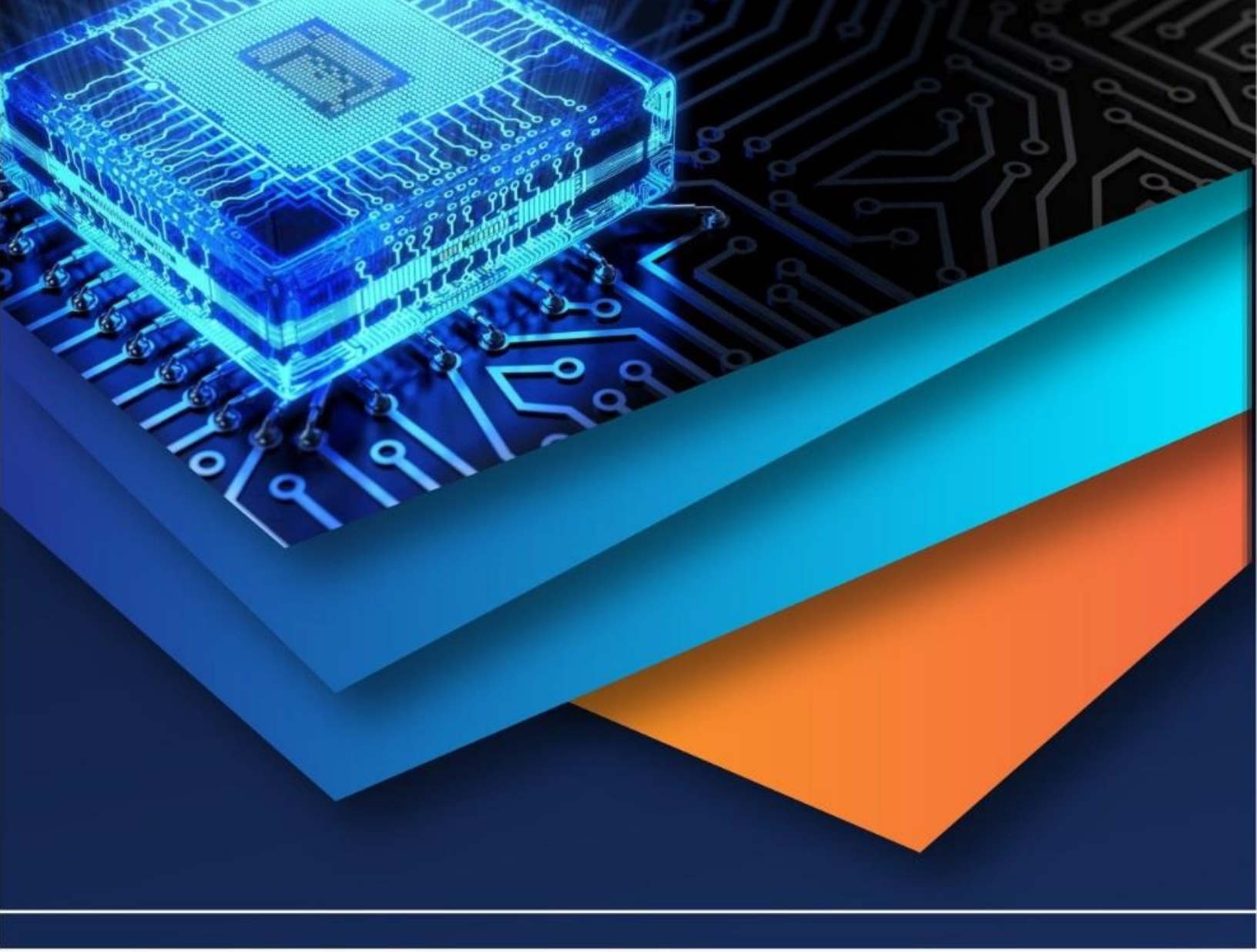
In the future, the system can be enhanced by adding live camera and audio streaming features to provide real-time visual and audio evidence during emergencies. This would enable authorities to monitor ongoing incidents more effectively and respond promptly. Furthermore, the system can be directly integrated with local police stations so that automatic notifications and live location tracking are initiated whenever an SOS is activated. Additionally, victims can be directed to the nearest police stations displayed on an integrated map, allowing them to navigate toward safety while authorities are simultaneously alerted. Such integrations would greatly enhance the speed, accuracy, and effectiveness of emergency responses, ensuring quicker assistance and higher safety for users in need.

VII. CONCLUSION

In conclusion, the Citizen Shield SOS System illustrates the possibility of using new web technologies to create an effective and secure emergency response system. The combination of React.js as an interactive front-end, Node.js with Express.js for a sound back-end, and Socket.io for real-time communication enables instantaneous and seamless dissemination of alerts. The incorporation of Geolocation API for real-time tracking and Node mailer for auto-email notifications by the system boosts responsiveness in case of emergencies. Additionally, security features like HTTPS and optional JWT authentication provide protection for user data and secure communication. This model offers a future-proof basis for other developments like mobile integration, AI-enabled alert prediction, and cloud data analysis, eventually boosting safety and community-based emergency assistance.

REFERENCES

[1] Sinha, A. K., Kumar, A. V., Saha, R., Roy, A., & Kadel, K. J. (2025, April). Women Security Application Using Smart Emergency Response System and Real-Time Location Tracking. International Scientific Journal of Engineering and Management, 04(04), 1–9.
[2] RAGAVENDRAN, G., PRIYAN, K. M., KAMALESHWARAN, S., & SOUNDARIYA, R. S. (2025, March). SOS EMERGENCY ALERT AND ASSISTANCE MOBILE APPLICATION. Peer Reviewed Journal, IRJEdT, 07(03).
[3] Lakpathi, G., Reddy, S. C., Hamsalekha, P., & Goud, D. N. (2025). Disha: Empowering women with safety and security at their fingertips. International Journal Scientific Research in Engineering and Management (IJSREM), 9(5), 1–6.
[4] MohitPasha, S. F. J. (2025). Android app for women safety. International Journal for Multidisciplinary Research (IJFMR), 7(2).
[5] Kamble, M., Naik, S., Jawadekar, V., & Metkar, S. (2025, June). Emergency Safety Device for Women and Children. International Journal for Research in Applied Science & Engineering Technology, 13(VI).
[6] Soni, A., Dalal, A., Singh, M. P., & Chaurasia, K. (2024, January). Emergency Response App for Enhancing Location-based Services using Socket.io and WebRTC.
[7] Choudhary, A. K., Pagare, S., & Bhujade, V. (2023). Design and implementation of women's safety system. International Journal for Research in Applied Science and Engineering Technology (IJRASET), 11(III), 2637–2641.
[8] Khan, A. Y., Luque-Nieto, M. Á., Batool, S., Ahmed, H., Saeed, A. H., & Asif, Z. (2023, January). SOS Application Under Android: Help Pro. Wireless Personal Communications, 128(2), 987–1002.
[9] Aruna, T. N., Kumar, N., Habakuk, N., Nivethitha, S., & Pooja, B. (2023, May). Women Safety App for Improved Personal Security. International Journal for Research in Applied Science & Engineering Technology (IJRASET), 11(V), 1134–1138.
[10] Shashikala, H. K., Priyanka, S., Madhumala, R. B., Meghana, R., Keerthana, C., & Thanmai, Y. (2022, June). Smart Reminder SOS & Emergency Detection Device. 2022 IEEE International Conference on Distributed Computing and Electrical Circuits and Electronics (ICDCECE).
[11] Sahu, B., Chandrakar, A., & Singh, T. G. (2022, February).Raksha: A Safety Alert App.International Journal for Research in Applied Science & Engineering Technology (IJRASET), 10(II), 648–652.
[12] Malik, Z. L. H., Nausheen, K., Tanzila, K., & Samima, A. (2021). Harassment Monitoring System using Android.International Journal of Engineering Research & Technology (IJERT), 9(4), 46–49.
[13] Sahu, B., Chandrakar, A., & Singh, T. G. (2022, February).Raksha: A Safety Alert App.International Journal for Research in Applied Science & Engineering Technology (IJRASET), 10(II), 648–652
[14] Hossain, M. E., Rahman, M., Qaiduzzaman, K. M., Shakir, A. K., & Hassan, M. M. (2019, October). Efficient Anti-Kidnapping and Anti-Harassment (Avoidance-Detection-Notification) Mobile Application for Unwanted Incidents.
[15] Kausikumar, B., Jeyabharathi, P., Saravana Kumar, A., Gopinath, G., & Deepak Kumar, B. (2020, July). Women and children safety SOS system: IoT based witness record system. International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering, 9(7), 2310–2315.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY