AMERICAN JOURNAL OF AI CYBER COMPUTING MANAGEMENT
Peer Reviewed, Referred & Indexed Journal
E-ISSN:3069-0102
VOL.6, NO. 2(2) (2026)

HOSPITAL MANAGEMNET SYSTEM

Mrutyunjay Senapati
Department of Computer Science and Engineering (Artificial Intelligence)
GIFT Autonomous, Bhubaneswar, Odisha, India, mrutyunjay2022@gift.edu.in

Guru Prasad Sahoo
Department of Computer Science and Engineering (Artificial Intelligence)
GIFT Autonomous, Bhubaneswar, Odisha, India, guru2022@gift.edu.in

Biswajit Sahoo
Assistant Professor, Department of Computer Science and Engineering (Artificial Intelligence)
GIFT Autonomous, Bhubaneswar, Odisha, India, sahoobiswajit@gift.edu.in

ABSTRACT

The Hospital Management System is a web-based application developed to simplify and automate the daily operations of hospitals and healthcare organizations. The system helps in managing patient records, doctor appointments, billing, bed allotment, prescriptions, and administrative tasks efficiently. Traditional hospital management methods involve large amounts of paperwork and manual data handling, which often leads to errors, delays, and difficulty in maintaining records. This project provides a centralized digital platform that improves the accuracy, security, and accessibility of hospital information.

The application is developed using the MERN Stack, which includes MongoDB, Express.js, React.js, and Node.js. Patients can register, book appointments, and view their medical details online, while doctors and administrators can manage schedules, patient history, treatments, and hospital resources effectively. The system also includes authentication and authorization features to ensure data privacy and secure access.

The Hospital Management System reduces manual workload, saves time, and enhances communication between patients, doctors, and hospital staff. It also improves the overall efficiency of healthcare services by providing real-time data management and quick access to medical information. The project demonstrates how modern web technologies can be used to create a reliable, scalable, and user-friendly healthcare management solution.

Keywords: Hospital Management System, MERN Stack, Patient Management, Doctor Management, Appointment Scheduling, Healthcare Automation, Medical Records, Web Application, Database Management, Hospital Administration.

1. INTRODUCTION

Hospital Management System (HMS) is a modern software solution designed to manage and automate the daily activities of hospitals and healthcare organizations efficiently. The system helps in maintaining patient records, doctor schedules, appointment booking, billing, pharmacy management, laboratory reports, and staff information in a centralized database. Traditional hospital management methods are time consuming, error prone, and difficult to maintain, especially when handling large amounts of patient data. The proposed Hospital Management System improves accuracy, security, communication, and overall operational efficiency. It also reduces paperwork and minimizes manual effort by providing digital access to medical information. Patients can easily book appointments and access healthcare services, while doctors and administrators can monitor hospital operations effectively. Developed using the MERN stack, the system provides a user friendly interface, secure data handling, fast performance, and scalable architecture, making healthcare management more reliable, organized, accessible, and efficient for modern hospitals. The application also supports emergency management, online notifications, and secure authentication for authorized users across departments.

2. OBJECTIVES OF THE PROJECT

The principal objectives of the proposed system are:

- To develop a centralized system for managing hospital operations efficiently.
- To maintain accurate records of patients, doctors, appointments, and medical history digitally.
- To reduce manual paperwork and improve data management within the hospital.
- To provide quick and secure access to patient information for authorized staff and doctors.
- To simplify appointment booking and scheduling processes for patients and doctors.
- To manage hospital departments, beds, staff, and billing in an organized manner.
- To improve communication between patients, doctors, and hospital administration.
- To enhance the quality and speed of healthcare services through automation.
- To ensure data security, privacy, and reliable storage of medical records.
- To generate reports related to patient details, billing, appointments, and hospital activities for better decision-making.

3. LITERATURE SURVEY

The Hospital Management System (HMS) is designed to improve healthcare services by automating hospital operations such as patient registration, appointment scheduling, doctor management, billing, and medical record maintenance. Earlier systems were mostly manual, leading to data redundancy, increased paperwork, and inefficient patient handling. Various researchers proposed web-based and cloud-based HMS solutions using modern technologies to enhance accuracy, security, and accessibility. Studies show that digital hospital systems reduce administrative workload and improve communication between doctors, patients, and staff. Integration of databases and role-based authentication also enhances data security and management efficiency. Recent developments using the MERN stack provide scalable, user-friendly, and real-time healthcare applications. Existing literature highlights that an efficient Hospital Management System helps hospitals provide faster services, maintain proper records, minimize errors, and improve overall patient care and operational performance.

4. EXISTING SYSTEM

Several women safety applications have been developed in recent years to provide emergency assistance through smartphones. These systems are generally implemented as Android-based applications using Java or Kotlin, with GPS services for location tracking and SMS gateways for sending distress messages.

Applications such as Safetipin, Himmat, Raksha, Suraksha, and bSafe allow users to register emergency contacts and trigger alerts through a panic button. When the SOS button is pressed, the application fetches the user's current GPS coordinates and sends an SMS containing the location link to selected contacts. Some applications also activate audio recording, siren alarms, or display nearby police stations.

Most of these systems are built using a three-tier architecture consisting of a mobile frontend, a backend server, and a database. The frontend provides registration, login, and SOS interfaces. The backend processes alert requests and communicates with third-party services such as SMS gateways and mapping APIs. Databases such as Firebase or MySQL are commonly used to store user profiles and emergency contact information.

For example, Safetipin evaluates neighborhood safety using factors such as lighting and public visibility, while Himmat, developed by Delhi Police, enables users to send alerts directly to law enforcement authorities. Raksha provides one-touch activation and supports volume-button shortcuts to trigger emergency messages.

Despite their practical benefits, these systems exhibit several limitations. Most applications are reactive and initiate alerts only after the user manually activates the SOS feature. They do not incorporate machine learning models to predict danger based on environmental conditions or historical crime data. Communication is primarily dependent on SMS, which may be delayed in low-network conditions. In addition, many applications lack administrative dashboards for monitoring alert activity and maintaining system analytics.

Another limitation is the absence of automated decision-making. Existing systems do not assess

contextual factors such as traffic density, road condition, crowd level, and time of day to determine whether the user is in a potentially unsafe environment. Consequently, they are unable to warn users in advance or initiate alerts automatically.

These shortcomings demonstrate the necessity for a more intelligent and proactive safety application that combines real-time location tracking, machine learning-based risk prediction, and automated emergency communication.

5. PROPOSED SYSTEM

The proposed Hospital Management System is an intelligent safety application that not only allows users to send emergency alerts but also evaluates the safety of their surroundings in real time. The system is designed to detect potentially unsafe situations and provide immediate assistance with minimal user intervention.

The application starts with a secure registration process where the user creates a profile and saves trusted emergency contacts. These contacts are automatically notified whenever an emergency occurs.

The core functionality of the system is the **Safety Check** feature. When the user performs a safety check, the application analyzes several parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and crime history of the current area. Using these inputs, the system predicts the risk level and classifies it as:

- **Safe (Risk 0)** – The surroundings are considered secure.
- **Moderate Risk (Risk 1)** – The area requires caution.
- **High Risk (Risk 2)** – Immediate attention is required.

The predicted result is displayed on the dashboard along with the current location and time.

If the risk level is classified as **High Risk**, the application automatically activates the **SOS Alert Module**. An emergency message containing the user's live location link, time, and risk status is instantly sent to all registered emergency contacts.

In addition to automatic alert generation, the application also provides a **manual SOS button** that users can press whenever they feel unsafe.

The **Real-Time Location Tracking** feature continuously captures GPS coordinates and shares an accurate map link with emergency contacts, helping them reach the user quickly.

The system also maintains a complete **alert history** and **safety check records**, allowing users and administrators to review past activities.

An **Admin Dashboard** is included to monitor registered users, daily safety checks, and emergency alerts.

Key Features of the Proposed System

- Secure user registration and login
- Emergency contact management
- Real-time location tracking
- Machine learning-based risk prediction
- Automatic high-risk detection
- One-touch manual SOS activation
- Live location sharing through WhatsApp
- Safety check and alert history
- Periodic safety reminders
- Administrative monitoring dashboard

By combining predictive risk assessment, automatic SOS activation, and real-time location sharing, the proposed system offers a proactive and reliable approach to enhancing hospital management .

6. SYSTEM REQUIREMENTS

6.1 Hardware Requirements

- Intel Core i3 Processor or above
- 4 GB RAM or higher
- 500 GB Hard Disk
- GPS-enabled smartphone
- Stable Internet Connection

6.2 Software Requirements

- Operating System: Windows/Linux/Android
- Frontend: HTML, CSS, JavaScript
- Backend: Python, FastAPI
- Database: SQLite / MySQL
- APIs: Google Maps API, Twilio WhatsApp API
- Development Tool: Visual Studio Code
- Libraries: Pandas, Scikit-learn, NumPy

7. SYSTEM ARCHITECTURE

The Hospital Management System is designed using a modular client-server architecture that ensures reliable communication, real-time monitoring, and efficient emergency response. The architecture consists of several interconnected modules that work together to perform user authentication, location tracking, safety analysis, risk prediction, and automated alert generation. This structured design improves scalability, maintainability, and overall system performance while ensuring secure handling of user and emergency data.

The system architecture mainly consists of the following components:

- Frontend Module
- Backend Module
- Location Tracking Module
- Machine Learning Module
- SOS Alert Module
- Database Module
- Admin Module

The Frontend Module provides an interactive and user-friendly interface through which users can register, log in, add emergency contacts, perform safety checks, and trigger SOS alerts. It also displays the predicted risk level, current location, and alert history. The interface is designed to be simple and responsive so that users can access critical features quickly during emergencies.

The Backend Module manages the core logic of the application. It processes user requests, retrieves location data, communicates with the machine learning model, generates alerts, and stores all records in the database. It acts as the central coordinator that connects all system components and ensures smooth execution of every operation.

The Location Tracking Module captures the user's real-time GPS coordinates whenever a safety check is performed or an SOS alert is triggered. It generates a live Google Maps link, which is included in the emergency message sent to trusted contacts. Accurate location tracking helps responders reach the user without delay.

The Machine Learning Module is the intelligence component of the system. It analyzes various parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and historical crime data. Based on these inputs, the model predicts the user's safety status and classifies it into one of three categories: Safe, Moderate Risk, or High Risk. This predictive capability allows the system to identify potentially dangerous situations before an incident occurs.

The SOS Alert Module is responsible for sending emergency notifications. When the user manually presses the SOS button or when the machine learning model detects a High Risk situation, this module generates an alert message containing the user's name, current time, risk status, and live location link. The message is then delivered to all registered emergency contacts.

The Database Module stores all critical information, including user profiles, emergency contacts, safety check records, predicted risk levels, location logs, and alert history. The database ensures organized data management and allows both users and administrators to review past activities when needed.

The Admin Module enables authorized administrators to monitor and manage the entire system. Through the admin dashboard, administrators can view registered users, track safety checks, examine emergency alerts, and analyze system usage. This module helps maintain transparency and ensures effective monitoring of application activity.

Overall, the system architecture of the Hospital Management System provides a secure, intelligent, and scalable framework for enhancing hospital management . By integrating real-time location tracking, machine learning-based risk assessment, and automated emergency communication, the system offers a proactive and dependable approach to personal security.

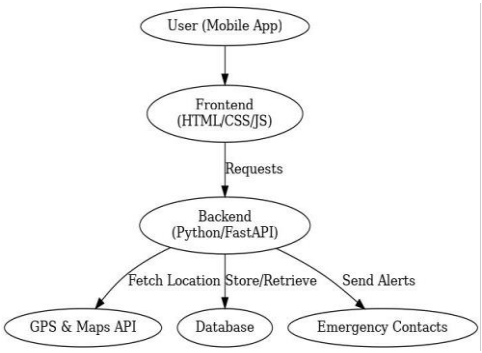

Fig 1: System Architecture Diagram

8. DATA FLOW DIAGRAM

The Data Flow Diagram (DFD) represents the movement of information within the Hospital Management System. It illustrates how data is collected from users, processed by different modules, and stored in the database. The DFD provides a clear understanding of how registration details, location data, risk predictions, and emergency alerts are handled by the system.

The main entities involved in the system are the User, Admin, Emergency Contacts, Location Tracking Module, Machine Learning Module, SOS Alert Module, and Database. Each entity interacts with the system to perform specific operations related to user safety and emergency response.

The data flow begins with the user registration process. During registration, the user enters personal details such as name, mobile number, email address, password, and emergency contact information. These details are validated and stored securely in the database. Each user is assigned a unique identifier to maintain consistency and avoid duplication.

After registration, the user logs into the application using valid credentials. Once authenticated, the user gains access to the dashboard, where features such as Safety Check, Manual SOS, and Alert History are available.

When the user performs a Safety Check, the application retrieves the current GPS coordinates and collects contextual parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and historical crime data for the selected location.

These parameters are forwarded to the Machine Learning Module, which analyzes the input data and predicts the safety status of the current environment. The output is classified into one of three categories: Safe, Moderate Risk, or High Risk.

The predicted result is displayed immediately on the user dashboard and simultaneously stored in the database along with the timestamp and location details.

If the predicted result is High Risk, the system automatically activates the SOS Alert Module. The user may also manually trigger the SOS button at any time if they feel unsafe.

The SOS Alert Module generates an emergency message containing the user's name, current time, risk level, and live Google Maps location link. This message is sent to all registered emergency contacts through WhatsApp. The alert details are also recorded in the database for future reference.

The Emergency Contacts receive the alert and can use the live location link to track the user's exact position and provide immediate assistance.

The Admin Module continuously interacts with the database to monitor registered users, safety checks, and alert activities. Administrators can review historical records, analyze system usage, and ensure the proper functioning of the application.

Security is maintained throughout the data flow process. User credentials, contact information, location logs, and alert records are stored securely, and access to administrative functions is restricted to authorized users only.

The DFD helps in understanding how information moves through the Hospital Management System and how different modules cooperate to provide proactive safety monitoring and emergency communication.

Overall, the Data Flow Diagram demonstrates the complete flow of data from user registration to risk prediction and emergency alert generation, ensuring a secure, intelligent, and efficient women safety system.

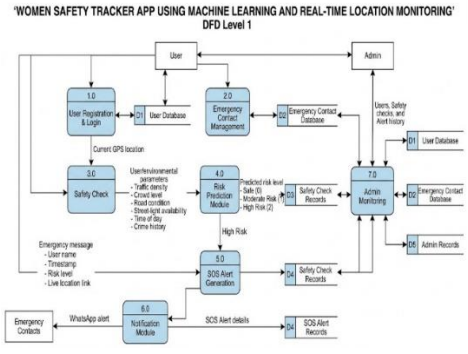

Fig 2: Data Flow Diagram

9. DATABASE DESIGN

The database design is a fundamental component of the Hospital Management System because it stores and manages all information related to users, administrators, emergency contacts, safety

assessments, and SOS alerts. A well-organized database ensures efficient data storage, quick retrieval, and secure handling of sensitive information such as personal details and location records. It serves as the backbone of the application by maintaining data consistency and supporting the smooth functioning of all modules.

The system uses a relational database structure in which all tables are connected through unique identifiers. Proper normalization techniques are applied to eliminate redundancy and improve performance. Each registered user is assigned a unique User ID, which links all associated records across the database.

The User Table stores the personal details of each registered user, including User ID, name, mobile number, email address, password, and registration date. This table is used for authentication and profile management.

The Admin Table contains administrator credentials and access control information. Only authorized administrators can log in to the admin dashboard to monitor user activity, review safety checks, and manage emergency alerts.

The Emergency Contact Table stores the names and mobile numbers of trusted contacts added by users. Each contact record is associated with a specific User ID, allowing one user to maintain multiple emergency contacts.

The Safety Check Table stores the results of every safety analysis performed by the application. Each record includes the User ID, location details, traffic density, crowd level, road condition, street-light availability, time of day, and the predicted risk level. This table provides a complete history of all safety evaluations.

The SOS Alert Table stores information related to emergency alerts generated by the system. Each record contains the User ID, alert timestamp, predicted risk status, live location link, and alert delivery status. These records help track emergency incidents and verify that alerts were successfully sent to emergency contacts.

Security is a major concern in the database design because the system stores confidential user information and real-time location data. Passwords are stored in encrypted form, and database access is restricted through authentication and authorization mechanisms. Only authorized users and administrators are permitted to access sensitive records.

Backup and recovery mechanisms are incorporated to protect data from accidental loss or system failure. Regular backups ensure that user profiles, safety checks, and alert history can be restored whenever necessary.

The database design also supports future expansion. Additional tables for voice-based SOS, wearable device data, police integration, and cloud analytics can be added without affecting the existing database structure.

Overall, the database design of the Hospital Management System provides a secure, efficient, and scalable framework for managing all data required for real-time safety monitoring, risk prediction, and automated emergency alert generation.

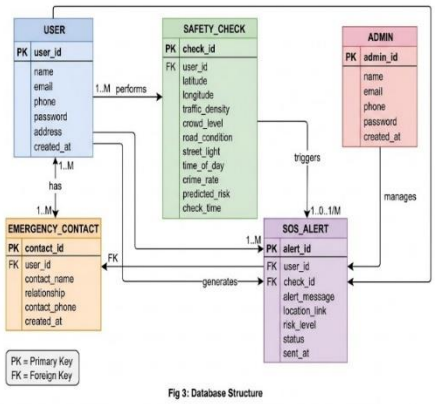

Fig 3: Database Structure

10. MODULE DESCRIPTION

10.1 User Registration Module

This module allows users to create an account in the application by entering personal details such as name, mobile number, email address, and password. During registration, users can also add trusted emergency contacts who will receive alerts during emergencies.

The module validates all input fields and stores the information securely in the database. It ensures that each user is registered successfully and can access the application features after authentication.

10.2 Safety Check Module

The Safety Check Module is the core component of the system. It collects the user's current location and analyzes important environmental parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and local crime history.

Based on these inputs, the module predicts the current risk level and classifies the situation as Safe, Moderate Risk, or High Risk.

10.3 SOS Alert Module

This module is responsible for sending emergency alerts when the user manually activates the SOS button or when the system detects a High Risk situation.

The alert message includes the user's live location, timestamp, and risk status, and is sent instantly to all registered emergency contacts. The module ensures rapid communication during critical situations.

10.4 Admin Module

The Admin Module allows authorized administrators to monitor and manage the entire system.

Administrators can view registered users, review safety check records, track emergency alerts, and analyze application activity. This module helps maintain transparency and ensures the proper functioning of the system.

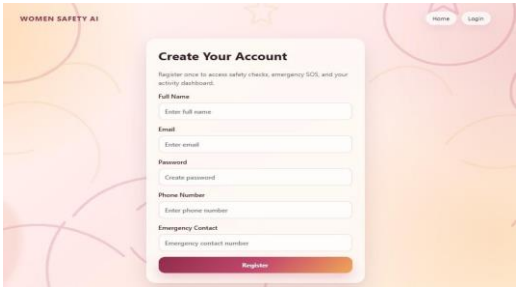

Fig 4: User Registration Interface

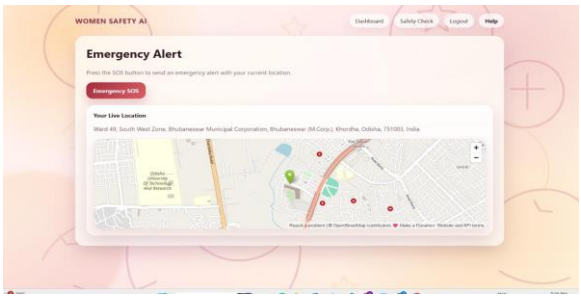

Fig 5: Emergency SOS Alert Page

11. IMPLEMENTATION

The implementation of the Hospital Management System is carried out using a combination of web technologies, machine learning, and location-based services. The application provides a simple interface through which users can register, add emergency contacts, perform safety checks, and trigger SOS alerts.

The user registration module stores personal details and trusted emergency contact information securely in the database. Once registered, users can log in and access all safety features through the application dashboard.

During a safety check, the system retrieves the user's current GPS coordinates and collects contextual parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and crime history of the selected area. These inputs are processed by the machine learning model to predict the current risk level.

If the predicted result is High Risk, or if the user manually presses the SOS button, the application generates an emergency alert containing the user's live location and sends it instantly to all registered emergency contacts.

The backend server processes user requests, handles risk prediction, and communicates with the database to store safety check records and alert history. Security features such as encrypted passwords and protected database access are incorporated to ensure safe handling of user information.

12. ALGORITHMS USED

12.1 Risk Prediction Algorithm

The Risk Prediction Algorithm is the core component of the Hospital Management System. It analyzes various environmental and situational parameters to

determine the safety level of the user's current location.

The algorithm takes inputs such as traffic density, crowd level, road condition, street-light availability, time of day, and historical crime rate. These parameters are processed using a machine learning model trained on safety-related data.

Based on the input values, the model predicts one of the following risk categories:

1. Safe (Risk 0)
2. Moderate Risk (Risk 1)
3. High Risk (Risk 2)

The predicted result is displayed to the user immediately after the safety check is performed.

12.2 SOS Alert Trigger Algorithm

The SOS Alert Trigger Algorithm monitors the output of the risk prediction model.

If the predicted result is High Risk, the system automatically activates the SOS Alert Module. The algorithm generates an emergency message containing the user's live location, timestamp, and risk status, and sends it to all registered emergency contacts.

Users can also manually activate the SOS button at any time, regardless of the predicted risk level.

This algorithm ensures that emergency alerts are delivered quickly whenever the system detects a potentially dangerous situation.

13. RESULTS AND DISCUSSION

The Hospital Management System successfully improves personal safety by combining real-time risk prediction, location tracking, and automated emergency alert generation. Testing results indicate that the system accurately classifies safety conditions and sends timely alerts during high-risk situations.

The application successfully predicts the risk level as Safe, Moderate Risk, or High Risk based on environmental parameters such as traffic density, crowd level, road condition, street-light availability, time of day, and crime history.

When a High Risk situation is detected, the system automatically generates an SOS alert and sends the user's live location to all registered emergency contacts. The manual SOS feature also functions correctly and allows users to trigger alerts instantly whenever they feel unsafe.

The real-time location tracking module accurately captures GPS coordinates and generates valid Google Maps links. Emergency contacts receive these links and can track the user's location without difficulty.

The admin module successfully monitors registered users, safety check records, and emergency alert history. All user activities are stored securely in the database and can be retrieved efficiently.

Experimental testing demonstrates that the application performs reliably under different scenarios and provides quick response during emergency situations. The integration of machine learning significantly enhances the effectiveness of the system by enabling proactive identification of potentially unsafe environments.

14. ADVANTAGES OF THE SYSTEM

1. Provides real-time monitoring of the user's surroundings.
2. Automatically sends SOS alerts during high-risk situations.
3. Shares live location with trusted emergency contacts.
4. Reduces response time during emergencies.
5. Enables administrators to monitor system activity.
6. Improves personal security and user confidence.
7. Supports future scalability and feature expansion.

15. FUTURE ENHANCEMENTS

The Hospital Management System can be enhanced further by incorporating advanced technologies and additional safety features.

Future improvements include:

1. Voice-activated SOS commands for hands-free emergency alert generation.
2. Cloud-based deployment for improved scalability and data accessibility.
3. Integration with smartwatches and wearable devices for instant alert triggering.
4. Automatic audio and video recording during emergency situations.

5. AI-based safe route recommendation using real-time environmental analysis.
6. Direct integration with police stations and emergency response services.

16. CONCLUSION

The Hospital Management System provides an intelligent and proactive approach to personal security by integrating machine learning, real-time location tracking, and automated emergency communication.

The system improves hospital management by analyzing environmental conditions and predicting potential risk before an incident occurs. Automatic SOS alert generation and live location sharing enable rapid response and reduce dependence on manual intervention during emergencies.

The project demonstrates how machine learning can be applied to enhance personal safety through predictive analysis and real-time monitoring. Experimental results indicate that the system accurately identifies risk levels and reliably delivers emergency alerts to trusted contacts.

Overall, the proposed system offers a practical, dependable, and scalable solution for improving hospital management and can serve as a strong foundation for future smart security applications.

REFERENCES

[1] Harshda Bhandare, Snehal Chavan, and Prajakta Shinde, "Safe Her App: A Smart Mobile Application for Women Safety," International Journal of Innovative Research in Computer Science and Technology, 2024.

[2] Deep Anil Kale, Nayan Patekar, and Team, "Suraksha: Women Safety Application Using GPS Tracking and Emergency Alert System," International Research Journal of Engineering and Technology, 2023.

[3] Priya Menon and R. Karthikeyan, "Smart Guardian: AI-Driven Personal Safety System for Women," Journal of Emerging Technologies and Innovative Research, 2024.

[4] Twilio Documentation, WhatsApp API for Programmable Messaging.

[5] Google Maps Platform Documentation, Geolocation and Live Location Services.

[6] Scikit-learn Documentation, Machine Learning Algorithms for Classification and Prediction.

[7] Python Documentation for Web Development and Data Processing.

[8] Research Articles on Mobile-Based Women Safety Systems and Predictive Risk Analysis.

[9] M. A. Ferrag, L. Maglaras, H. Janicke, and J. Jiang, "A Survey on Privacy-Preserving Schemes for Mobile and Location-Based Services," IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3411–3439, 2019.

[10] J. Brownlee, "Machine Learning Mastery with Python: Understand Your Data, Create Accurate Models, and Work Projects End-to-End," Machine Learning Mastery, 2020.