International Journal of Scientific Research in Engineering and Management (IJSREM)
Volume: 09 Issue: 04 | April - 2025 SJIF Rating: 8.586 ISSN: 2582-3930

Corporate Safety System Using SERN Stack

S R SOWMIYA[*1], MADHUMITHA M[*2], PRIYANKA R[*3], RAGAPRIYA J[*4]
**1Assistant professor, Department of CSE, Dhanalakshmi Srinivasan Engineering College (Autonomous), Perambalur, Tamil Nadu, India.*
**2,3,4UG Students, Department of CSE, Dhanalakshmi Srinivasan Engineering College (Autonomous), Perambalur, Tamil Nadu, India.*

Abstract - In recent years, workplace safety has become a critical concern for organizations striving to ensure a secure environment for employees and visitors. Traditional safety systems often lack real-time responsiveness and integrated monitoring, resulting in delayed incident reporting and inefficient emergency responses. This paper presents the Corporate Safety System, an intelligent, web-based solution developed using the SERN stack (SQL, Express.js, React, Node.js) to address the multifaceted aspects of corporate safety. The system features real-time hazard detection, weapon recognition through CCTV integration (Gun), incident reporting and tracking, emergency response automation, and compliance management. Additional components include a chatbot for instant safety assistance, role-based access control, and analytics for incident trends and audits. Through seamless integration of various technologies, the system offers centralized control and monitoring, enhancing workplace safety, response time, and decision-making. The proposed solution outperforms conventional safety systems by providing a proactive, automated, and scalable approach to hazard detection and safety management. Future work aims to incorporate blockchain for secure data handling and expand IoT integration for physical environment sensing.

***Key Words*:** Corporate Safety, Workplace Hazard, SERN Stack, Real-Time Monitoring, Safety Compliance

1.INTRODUCTION

In today's fast-paced industrial and corporate environments, ensuring workplace safety has become more vital than ever. Organizations are responsible not only for maintaining productivity but also for safeguarding the well-being of their employees, visitors, and infrastructure. Despite the implementation of standard safety protocols, many companies still rely on manual methods for hazard detection, incident reporting, and emergency response—leading to delayed actions and inefficient safety management.

Traditional safety systems often lack real-time surveillance, centralized monitoring, and data-driven decision-making capabilities. Moreover, existing solutions may focus on isolated functionalities, such as basic access control or standalone surveillance, without offering a comprehensive, integrated safety management platform.

To address these challenges, we propose the Corporate Safety System, a unified, intelligent, and web-based application designed to streamline and enhance corporate safety procedures. Developed using the SERN stack (SQL, Express.js, React, Node.js), the system is capable of managing multiple aspects of workplace safety, including real-time hazard detection, weapon detection through CCTV analysis using Python, incident reporting and tracking, role-based user access, automated emergency responses, and compliance monitoring.

The system integrates modern technologies to enable instant alerts, data analytics, and proactive safety measures. By centralizing safety functions into a single platform, the Corporate Safety System significantly reduces human error, response delays, and potential threats. It also includes a chatbot for quick guidance and safety support, making it user-friendly and efficient.

This paper explores the architecture, modules, and real-world applications of the Corporate Safety System, highlighting its potential to transform corporate safety management through automation, intelligence, and scalability.

2. IDENTIFY, RESEARCH AND COLLECT IDEA

The foundation of any innovative system lies in identifying a real-world problem and exploring effective technological solutions. In the context of corporate environments, workplace safety has emerged as a critical area requiring technological intervention. With rising concerns about hazards, delayed incident response, and inadequate reporting mechanisms, there is a growing need for a unified and intelligent safety management system.

To address this, extensive research was conducted on existing systems and recent advancements in safety technologies. The investigation included reviewing related literature, understanding safety protocols, and analyzing current solutions involving CCTV surveillance, incident management platforms, and emergency response frameworks.

The idea for the Corporate Safety System emerged from this research process. It focuses on building a centralized, real-time monitoring system powered by modern web technologies. The goal is to automate hazard detection, improve incident tracking, and offer faster emergency responses.

By integrating Python-based CCTV weapon detection, SERN stack web development, and role-based access control, the idea matured into a comprehensive solution capable of transforming corporate safety practices. This stage also included brainstorming innovative modules such as chatbot integration, safety training components, audit systems, and analytics dashboards, all aimed at creating a robust, scalable, and smart workplace safety application.

2.1 Problem Identification

In the modern corporate environment, safety is no longer a secondary concern—it is a fundamental requirement. Despite advancements in digital transformation, many workplaces still rely on outdated or manual safety protocols that are not equipped to handle emergencies in real time. These systems often lack proper hazard detection mechanisms, real-time alert systems, and comprehensive incident tracking tools.

This leads to delayed responses, inefficient reporting, and, in some cases, a failure to prevent serious incidents. Additionally, there is often no centralized platform where safety data can be visualized, managed, and analyzed effectively. Identifying this gap between traditional safety practices and modern workplace needs served as the foundation for developing a more intelligent and automated solution.

2.2 Background Research

To build an effective safety system, a detailed background study was conducted on current workplace safety technologies and strategies. This involved examining industry standards, analyzing research papers, reviewing incident reports, and studying government safety regulations. Existing safety systems were observed to be mostly reactive in nature, focusing on post-incident reporting rather than proactive detection and prevention. Surveillance systems were often isolated from incident management platforms, and many lacked integration with emergency response units. Furthermore, limited access control and lack of user-specific dashboards reduced accountability and hindered safety compliance. This research highlighted the need for a system that combines real-time detection, incident tracking, and smart responses in a single platform.

2.3 Technology Survey

The next phase involved identifying suitable technologies that could bring the safety system concept to life. After a detailed analysis, the SERN stack (SQL, Express.js, React, Node.js) was chosen for its modularity, scalability, and suitability for real-time web applications. React allows dynamic user interfaces, while Node.js and Express.js support fast and efficient backend processing. For intelligent surveillance, Python was selected due to its vast library support and ability to integrate with video processing tools like OpenCV.

Python also enables weapon detection through machine learning and computer vision. In addition, modern practices like role-based access control (RBAC) were selected to ensure secure user-level permissions, and chatbot APIs were considered to provide automated assistance to users during emergencies or safety training. These technologies together form a powerful ecosystem to implement a robust, secure, and responsive safety system.

2.4 Idea Formulation

Based on the problems identified and the technological possibilities explored, the core idea of the Corporate Safety System was formulated. The aim was to develop a centralized web application that integrates real-time CCTV monitoring with intelligent hazard detection, incident reporting, and emergency management. The idea also included supporting features like safety training modules, audit systems, and chatbot assistance to provide a complete safety management solution.

One of the major conceptual breakthroughs was combining weapon detection via CCTV with incident logging and automated emergency alerts, which allows quick action and reduces human dependency. This platform is intended to be used not only by safety officers but also by employees and managers, each with role-specific access and functionalities. The goal was to move from traditional reactive safety systems to a smart, proactive approach powered by automation and intelligence.

2.5 Objective Definition

With a clear problem and a solid idea in place, a set of specific objectives was defined to guide the development of the project. These objectives include:

- Designing a user-friendly web interface using React.js for easy access and navigation.
- Implementing real-time weapon and hazard detection through CCTV feeds using Python and OpenCV.
- Enabling a secure login system with role-based access for Admin, Safety Officer, and Employee.
- Integrating automated incident reporting and alert notifications to minimize response time.
- Creating a chatbot assistant to guide users during emergencies or for safety-related queries.
- Including modules for safety training, compliance checks, and audit management.

• Providing a dashboard with analytics and data visualization to support decision-making. These objectives help in structuring the project into manageable components while ensuring that the final product meets real-world corporate safety requirements.

3.PROPOSED SYSTEM

The Corporate Safety System is proposed as a comprehensive and intelligent web-based platform designed to overcome the limitations of existing workplace safety mechanisms. This system integrates real-time hazard detection, smart surveillance, automated incident reporting, and emergency response features within a centralized and user-friendly interface.

It utilizes modern technologies such as CCTV surveillance with AI-based weapon detection, the SERN stack (SQL, Express.js, React, Node.js) for full-stack web development, and Python for image and video processing. The proposed system aims to enhance safety, improve response time, and ensure regulatory compliance through a smart and scalable architecture.

3.1 System Overview

The proposed Corporate Safety System is a full-fledged web-based safety monitoring platform developed using the SERN stack—SQL, Express.js, React, and Node.js—integrated with Python for real-time video analysis. This system addresses the increasing demand for intelligent workplace safety solutions by offering proactive monitoring and instant incident response capabilities.

Unlike traditional safety models that rely on manual reporting and delayed intervention, this system ensures immediate hazard detection, streamlined reporting, and centralized control. It is designed to safeguard both employees and infrastructure by utilizing artificial intelligence, automated alerts, and structured compliance tools within a secure and interactive web interface. The entire system is accessible via a web application with a responsive and secure interface built using React and Express.js.

3.2 Functional Features

The Corporate Safety System includes a variety of interconnected features that enhance its functionality and reliability. One of the core features is real-time CCTV surveillance, where Python-based modules analyze live video feeds to detect weapons or suspicious activities. When a hazard is detected, the system triggers an automated incident logging mechanism that records critical details such as timestamp, incident type, and location. This is followed by an emergency alert, which may include visual warnings, sound alarms, or SMS/email notifications to the appropriate personnel.

Another vital component is the role-based access control system, ensuring that users only access functionalities relevant to their designated roles, such as Admin, Safety Officer, or Employee. Additionally, the platform incorporates a chatbot that assists users by offering safety guidance, helping with training navigation, and answering emergency questions. The training and audit modules ensure that employees are regularly educated on safety protocols and that the organization remains compliant with regulations through scheduled reviews and documentation.

3.3 System Architecture

The system architecture follows a modular and scalable design. The frontend interface, built using React.js, provides a clean, user-specific dashboard with modules for incident reports, safety training, chatbot communication, and analytics. It interacts with a backend API built with Node.js and Express.js, which processes business logic and handles secure authentication, user management, and routing.

The data is stored in a structured SQL database that maintains records of users, incidents, audit logs, and training status. On the visual detection end, Python scripts are integrated to analyze CCTV footage for potential threats in real time. When a threat is detected, the information is passed through the backend to update the system and trigger appropriate alerts. The architecture ensures low latency communication between components and is built to handle scalability for enterprise-level deployment.

3.4 System Advantages

The proposed Corporate Safety System brings several significant advantages. First, it ensures real-time incident detection and immediate response, drastically reducing the reaction time compared to traditional safety systems. Second, it automates the entire workflow—from hazard identification to logging and alerting—minimizing human error and improving accuracy.

Third, the platform is user-friendly and role-specific, making it easy to use for different users within the organization. It also supports ongoing training and safety audits, which help organizations maintain compliance and improve overall safety culture. Furthermore, the system is designed with scalability in mind, allowing future enhancements such as integration of biometric scanners, fire detection systems, or advanced analytics dashboards. By consolidating various safety components into one centralized platform, this proposed

system offers a complete solution for corporate safety management in modern organizations.

4.SYSTEM DESIGN

4.1 Introduction to System Design

System design serves as the foundational blueprint for developing the Corporate Safety System. It defines the architecture, components, interfaces, and data flow of the system in a way that ensures smooth integration of all features. The objective of system design in this context is to create a scalable, secure, and user-friendly platform that can efficiently monitor, report, and respond to workplace hazards.

System design is a critical phase in software development that defines the overall architecture, components, modules, interfaces, and data flow of the application. It acts as a blueprint for constructing the entire system and ensures that all functional and non-functional requirements are addressed effectively. In the case of the *Corporate Safety System*, the system design phase involves understanding how each module—such as user management, real-time monitoring, incident reporting, and AI-driven hazard detection—will function individually and interact with each other as a unified framework.

The system aims to deliver a robust, scalable, and secure safety monitoring platform using the SERN (SQL, Express.js, React.js, Node.js) stack, integrated with external AI services for real-time hazard detection. The design ensures that user interactions from the frontend are efficiently handled by backend APIs, which communicate with both the database and AI modules. Each component is designed to work independently and collaboratively to achieve system objectives such as immediate alerts, efficient incident tracking, and secure data handling.

System design also involves deciding on key architectural elements like the database schema, user interface layout, backend API endpoints, third-party integrations, and alert mechanisms. The modular structure of the system allows for easy maintenance and future enhancements such as mobile app integration, blockchain security, and advanced analytics. The architecture focuses on fault tolerance, role-based access control, real-time data processing, and adherence to safety regulations, making it ideal for corporate environments.

4.2 Use Case Diagram

The use case diagram illustrates the interactions between users and the system's core functionalities. The primary actors include the Admin, Safety Officer, and Employee. Each actor is assigned specific roles: the Admin manages user accounts, views incidents, oversees audit reports, and controls system settings. The Safety Officer is responsible for responding to alerts, reviewing hazard reports, and updating incident status.

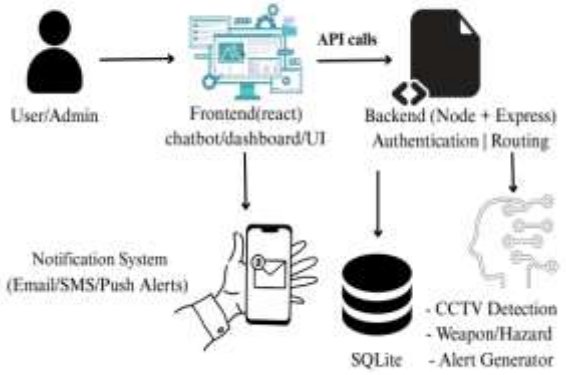

Fig.no: 4.1

The Employee interacts with the system by undergoing safety training, receiving alerts, and communicating through the chatbot in emergency situations. This diagram ensures that all user interactions are defined clearly and can be implemented through proper interface design and role-based access control. clearly and can be implemented through proper interface design and role-based access control.

4.3 Class Diagram

The class diagram presents the structure of the system's backend by showing the classes, their attributes, methods, and relationships. Core classes include User, Incident, Training Module, Audit Report, and Chatbot Response.

The User class contains fields like userID, name, role, and login Status, while the Incident class includes incidentID, type, timestamp, location, and status. Relationships between these classes show how incidents are linked to specific users and how training progress is tracked per employee. This design ensures logical data flow and enables structured querying in the SQL database.

4.4 Data Flow Diagram (DFD)

The Data Flow Diagram (DFD) highlights the movement of data within the system. When an incident is detected by the Python-based CCTV monitoring module, it sends a notification to the backend API. The backend processes this input, updates the SQL database, and

triggers appropriate actions such as sending alerts or logging events.

Simultaneously, the user interface updates the incident dashboard and sends real-time notifications to Safety Officers. Similarly, when users interact with training modules or the chatbot, the frontend sends requests to the backend, which fetches data from the database and delivers the response. This structured data flow ensures real-time responsiveness and minimal delay.

4.5 Sequence Diagram

The sequence diagram explains how the system processes events step-by-step. For example, when a hazard is detected, the sequence starts with the Python script identifying the event and passing the data to the backend. The backend then validates and stores the data in the SQL database. Following this, the system sends an alert to Safety Officers, updates the incident dashboard, and waits for user action. Once an officer acknowledges or resolves the incident, the updated status is reflected across the platform. This sequence of operations ensures synchronization across all modules and keeps records accurate and up-to-date.

4.6 Design Justification

The system design is purposefully modular and layered to ensure maintainability and extensibility. By separating the concerns of user interaction (frontend), logic processing (backend), and data storage (database), the system achieves robustness and scalability. Integration with Python for video surveillance enhances the system's real-time capabilities, while the use of secure APIs ensures safe communication between modules.

The design also considers future enhancements, such as the integration of fire detection systems, facial recognition, and blockchain for immutable incident logging. Thus, the system is not only functionally rich but also architecturally sound.

5.CONCLUSION

The Corporate Safety System developed using the SERN stack is a comprehensive and intelligent platform aimed at ensuring workplace safety through automation, real-time monitoring, and streamlined communication. By integrating advanced technologies such as real-time video surveillance, role-based access control, automated incident logging, and AI-powered chatbot assistance, the system addresses the limitations of traditional safety approaches. The proposed system offers a centralized solution that not only detects and reports hazards instantly but also enhances organizational preparedness through training and compliance modules.

Through its modular architecture and secure design, the system ensures scalability, reliability, and maintainability, making it suitable for large-scale deployment in corporate environments. The inclusion of analytics and audit tools further adds to its capability by enabling safety officers and administrators to make data-driven decisions for continuous improvement. Additionally, the seamless user interface and real-time alerting system foster a proactive safety culture among employees.

In conclusion, the Corporate Safety System stands as a promising technological advancement in the field of workplace safety. Its ability to integrate detection, reporting, training, and compliance into a unified web application reflects a forward-thinking approach to handling modern safety challenges. With further enhancements and integration of technologies like blockchain and advanced machine learning, this system holds great potential to redefine safety standards in corporate infrastructures.

REFERENCES

[1] M. Kiran, P. Ramesh, "Real-Time Hazard Detection Using CCTV and AI," International Journal of Computer Applications, vol. 183, no. 29, pp. 12–16, 2021.
[2] R. S. Mehta, "Modern Safety Management in Corporate Environments," Safety Science, vol. 145, pp. 105–112, 2022.
[3] H. R. Joshi and S. D. Kale, "Web-based Incident Reporting System Using MERN Stack," International Journal of Engineering Research, vol. 13, no. 3, pp. 455–460, 2021.
[4] P. Kumar and A. Gupta, "Node.js and Express.js for Backend Development," International Journal of Web Technologies, vol. 10, no. 1, pp. 22–28, 2020.
[5] T. Martin, "SQL Database Design for Scalable Applications," Journal of Data Science and Applications, vol. 9, pp. 55–61, 2021.
[6] S. Sharma, "React.js: Building Interactive Frontend Interfaces," Journal of Computer and Web Engineering, vol. 11, no. 2, pp. 67–75, 2021.
[7] N. Patel, "AI-Powered Workplace Safety Monitoring," IEEE Access, vol. 9, pp. 122345–122355, 2021.
[8] D. Banerjee and K. Mahadevan, "Incident Management Systems for Enterprises," International Journal of Information Systems, vol. 12, no. 4, pp. 108–116, 2020.
[9] A. Singh, "Design and Development of Chatbot for Workplace Assistance," International Journal of Artificial Intelligence Research, vol. 8, no. 2, pp. 200–209, 2022.
[10] S. Raj, "Express.js and RESTful APIs for Enterprise Apps," International Journal of Programming and Software Engineering, vol. 7, no. 1, pp. 80–88, 2021.

[11] A. Sharma, “Integrating Real-Time Alert Systems in Web Applications,” International Journal of Software and Systems, vol. 13, pp. 102–109, 2022.
[12] B. Nair, "OpenCV-Based Real-Time Video Processing for Surveillance," Computer Vision Journal, vol. 17, no. 3, pp. 92–99, 2021.
[13] R. Thomas and L. Joseph, "Secure Web Application Development Practices," Cybersecurity and Privacy Journal, vol. 14, no. 2, pp. 56–64, 2020.
[14] K. Verma, “Designing Scalable Microservices with Node.js,” Software Architecture Review, vol. 6, pp. 141–148, 2021.
[15] M. Desai, “Real-Time Workplace Risk Analysis Using AI,” Safety Engineering Journal, vol. 10, no. 1, pp. 34–42, 2021.
[16] J. Watson, “User Interface and Experience in Safety Applications,” International Journal of Human-Computer Interaction, vol. 29, no. 3, pp. 157–164, 2022.
[17] R. Bhatt, “Training and Compliance Systems in Modern Enterprises,” Enterprise Systems Management, vol. 8, pp. 88–96, 2020.
[18] S. Das, “WebSockets and Real-Time Communication in Safety Apps,” Journal of Web and Network Technologies, vol. 7, pp. 109–117, 2021.

[19] T. Kumar, “Role-Based Access Control in Corporate Applications,” Journal of Information Security and Policy, vol. 11, pp. 50–58, 2020.

[20] L. Zhang, “Applications of AI in Industrial Safety Systems,” IEEE Transactions on Industrial Informatics, vol. 17, no. 8, pp. 5240–5248, 2021.