Campus Incident Management System

Project Phase-2 (BCS786)

Harshavardhan J R, LR Dhayatri, Chaithanya C, Harshitha CK

Associative Professor, Student, Student, Student

Computer Science of Engineering and Management,

K S School of Engineering and Management, Bangalore, India

harshavardhanjr@kssem.edu.in, lekkalapudidhayatri@gmail.com, chaithanya84313@gmail.com, harshick1408@gmail.com

***Abstract* — This electronic document provides a centralized web-based platform designed to streamline the process of reporting, monitoring, and resolving incidents within an educational institution. The system allows students, faculty, and staff to record incidents such as safety issues, technical failures, or disciplinary matters. Reported incidents are stored in a centralized database where administrators can review, assign responders, and monitor the case until closure. Key features include photo and video evidence uploading, email notifications, and analytical report generation to improve overall campus safety and operational efficiency.**

***Index Terms* — Campus Incident Management System, incident reporting, centralized database, administrative dashboard, safety compliance.**

I. INTRODUCTION

The Campus Incident Management System is a centralized web-based platform designed to streamline the process of reporting, managing, and resolving various incidents that occur within an educational institution. In large university environments, handling issues such as safety hazards, technical failures, and medical emergencies can be challenging without a structured system. This project provides a reliable and transparent solution to address these challenges effectively.

A. Purpose of the Project (Heading 2)

The primary purpose is to establish an efficient platform for reporting, tracking, and resolving incidents. Without a proper system, communication gaps and delayed responses can lead to a lack of accountability. This project aims to minimize manual paperwork and reduce communication delays.

B. Scope of the Project

The scope covers the complete process from initial reporting to final resolution. The system serves as a centralized digital platform that improves coordination among students, faculty, staff, and administrative departments. It can be implemented across academic buildings, hostels, laboratories, and administrative offices.

II. Literature Survey

A. Incident Management Technical Guidelines

Dr. Youcef Remil [1] emphasizes that modern IT system management requires scalability and efficiency when dealing with huge streams of data. Conventional manual tasks and rule-based methods are often inefficient for large volumes.

B. Software Ecosystem Frameworks

Luiz Alexandre Costa [2] introduces an IM Framework grounded in a review of 293 studies. This framework includes five categories: organizational goals, practices, success factors, benefits, and barriers.

III. System Requirements and Specification

The following software and hardware were utilized:

- **Operating System**: Windows, macOS, Linux, Android, or iOS.
- **Frontend**: HTML, CSS, JavaScript, and React.js.
- **Backend**: Node.js Express and Python Django.
- **Database**: MongoDB.
- **Minimum Hardware**: Dual-core processor with at least 4 GB RAM.

IV. Methodology and Design

The methodology focuses on four key objectives:

1. **Efficient Reporting**: Enable role-based access for students, faculty, and security.
2. **Timely Response**: Use rule-based logic to assign responders based on incident severity.
3. **Analytical Tracking**: Implement dashboards to monitor trends and recurring issues.
4. **Data Security**: Encrypt all stored and transmitted data using secure protocols.

V. Acknowledgment

We would like to express our gratitude to the management of K.S. School of Engineering and Management for providing the infrastructure necessary for this project. Special thanks to our guide, Mr. Harshavardhan J R, and Dr. K. Venkata Rao for their support and valuable guidance.

VI. References

[1] Y. Remil, A. Bendimerad, R. Mathonat, and M. Kaytoue, "AIOps Solutions for Incident Management," arXiv, April 2024.

[2] L. A. Costa, A. Fontaob, R. P. dos Santos, and A. Serebrenik, "Towards an Incident Management Framework in Proprietary Software Ecosystems," arXiv, October 2024.

[3] G. Antoniol, K. Ayari, M. Di Penta, F. Khomh, and Y. G. Guéhéneuc, "Is it a bug or an enhancement? A text-based approach to classify change requests," CASCON, 2008.

[4] T. Berghout and M. Benbouzid, "A systematic guide for predicting remaining useful life with machine learning," Electronics, vol. 11, no. 7, 2022.