COMSATS University, Islamabad Pakistan

SafeSitePlus

By

Shahryar Amjad **CIIT/FA21-BCS-085/ISB**

Zamin Raza **CIIT/FA21-BCS-096/ISB**

Supervisor

Dr Akber Abid Gardezi

Bachelor of Science in Computer Science(2021-2025)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

COMSATS University, Islamabad Pakistan

SafeSitePlus

A project presented to

COMSATS University, Islamabad

In partial fulfillment

of the requirement for the degree of

Bachelors of Science in Computer Science (2021-2025)

By

Shahryar Amjad CIIT/FA21-BCS-085/ISB

Zamin Raza CIIT/FA21-BCS-096/ISB

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other, we will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Shahryar Amjad

Zamin Raza

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (CS) SafeSitePlus was developed by **Shahryar Amjad (CIIT/FA21-BCS-085)** and **Zamin Raza (CIIT/FA21-BCS-096)** under the supervision of Dr Akber Abid Gardezi and that in his opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Sciences

---

Supervisor

External Examiner

Head of Department
(Department of Computer Science)

Executive Summary

SafeSitePlus is an advanced safety management system designed to **automate the detection of personal protective equipment (PPE) compliance and site hazards** on construction sites. By leveraging AI-powered anomaly detection, it reduces the workload on supervisors by identifying safety violations and notifying them only when necessary.

The system ensures **efficient site management** by tracking safety compliance, logging incidents, and prioritizing alerts based on risk levels. AI-driven **prioritization** helps supervisors focus on the most critical issues first, improving response times and overall site safety. Additionally, SafeSitePlus provides tools for **incident management, reporting, and analytics**, enabling construction companies to maintain compliance and enhance workplace safety with minimal manual intervention. Its integrated **chatbot assistant** provides instant support,

By offering a **centralized and automated** approach to safety monitoring, SafeSitePlus helps supervisors manage multiple sites effectively, reducing the chances of oversight and ensuring a safer working environment.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor Dr Akber Abid Gardezi Without their personal supervision, advice and valuable guidance, completion of this project would have been doubtful. We are grateful to them for their encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us with the values of honesty & hard work.

Shahryar Amjad

Zamin Raza

Abbreviations

SRS	Software Require Specification
SDS	Software Design Specification
PC	Personal Computer
YOLO	You look only once
PPE	Personal Protective Equipment

Table of Content

1 Introduction 10

1.1 Vision Statement 10
1.2 Related System Analysis/Literature Review 10
1.3 Project Deliverables 11
1.4 System Limitations/Constraints 11
1.5 Tools and Technologies 11
1.6 Relevance to Course Modules 12

2 Problem Definition 13

2.1 Problem Statement 13
2.2 Problem Solution 13
2.3 Objectives of the Proposed System 14
2.4 Scope 14
2.5 Modules 14
2.5.1 Module 1: User Administration 14
2.5.2 Module 2: Site Management and Footage Assessment 15
2.5.3 Module 3: Supervisor Monitoring Interface 15
2.5.4 Module 4: Video Processing and Frame Display via WebSockets 15
2.5.5 Module 5: Dataset Management and Preprocessing 16
2.5.6 Module 6: Safety Anomaly Detection Framework 16
2.5.7 Module 7: Incident Management 16
2.5.8 Module 8: Data Analytics / Suggestions and Forecasting 16
2.5.9 Module 9: Reporting and Visualization 17
2.5.10 Module 10: AI Assistance/Chatbot 17

3 Requirement Analysis 18

3.1 User classes and characteristics 18
3.2 Requirement Identifying Technique 18
3.2.1 User Case Diagram 19
3.2.2 User Case Tabular 21
3.3 Functional Requirements 43
3.3.1 UFR-01: Login.Input.Username 43
3.3.2 UC-02: Forget Password 47
3.3.3 UC-03 View Profile 50
3.3.4 UC-04 Edit Profile 51
3.3.5 UC-05 Password Change 54
3.3.6 UC-06: Quick Access 59
3.3.7 UC-7 : View Notifications 62
3.3.8 UC-08: Keyword Search 64
3.3.9 UC-09: Logout 66
3.3.10 UC-10 : Register Supervisor 69
3.3.11 UC-11 View Supervisor Detail 73
3.3.12 UC-12 Suspend Supervisor Account 74
3.3.13 UC-13 Delete Supervisor Account 76
3.3.14 UC-14 Audit Logging 79
3.3.15 UC-15: View Cameras Details 81

3.3.16 UC-16: View Camera Status 83
3.3.17 UC-17: Customize Interface 84
3.3.18 UC-18: View Feedback 85
3.3.19 UC-19 : Monitor Camera Surveillance 86
3.3.20 UC-20: Switch between Cameras 87
3.3.21 UC-21:Guide (Manual) for PPE Usage 89
3.3.22 UC-22: Customize Anomaly Parameters 90
3.3.23 UC-23: View Weather Forecast 92
3.3.24 UC-24: View Alerts 92
3.3.25 UC-25: Acknowledge Alerts 93
3.3.26 UC-26: Download Footages 95
3.3.27 UC-27: View Automated Generated Reports 96
3.3.28 UC-28: Handle Reports 97
3.3.29 UC-29: Regenerate Reports 99
3.3.30 UC-30: Download Reports 100
3.3.31 UC-31: Provide Feedback to Admin 102
3.3.32 UC-32: View Safety Trends/Forecasts 103
3.3.33 UC-33: Give Usage/Help Tutorial 104
3.3.34 UC-34: Supervisor's Task Log 105
3.3.35 UC-35: Visualization 107
30.3.11.3 FR-131: Visualization.FilterDataByTimeRange 108
3.3.36 FRs For Events 110
3.3.37 FRs For UC-36: AI Assistance/Chatbot 120
3.4 Non-Functional Requirements 123
3.4.1 Reliability 123
3.4.2 Usability 123
3.4.3 Performance 123
3.4.4 Security 124
3.5 External Interface Requirements 124
3.5.1 Software interfaces 125
3.5.2 Hardware interfaces 125
3.5.3 Communications interfaces 126

4 Design and Architecture 127

4.1 Architectural Design 127
4.1.1 Box And Line Diagram 127
4.1.2 Architecture Diagram 128
4.2 Design Models 129
4.2.1 Activity Diagram 129
30.3.11.3 Login 129
4.2.1.2 Add Supervisor 130
4.2.1.3 Delete Supervisor 131
30.3.11.3 Customize Anomaly Parameters 134
30.3.11.3 Get Video Feed 135
4.2.2 Data Flow Diagram 138
30.3.11 State Transition Diagram 141
4.3 Data Design 142
4.3.1 Data Dictionary 142

5 Implementation 148

5.1 Algorithm 148
6.1 External APIs/SDKs 154
6.2 User Interface. 155

6.2.1 Landing Page 155
6.2.2 Login Page 157
6.2.3 Admin Dashboard 158
30.3.11 Add Supervisor 159
30.3.11 Audit Logging 160
5.3.6 Register a Site 161
5.3.7 Stats Page 162
5.3.8 Update Supervisor Credentials 163
5.3.9 Supervisor Dashboard 164
5.3.10 Supervisor Anomaly Parameters 165
5.3.11 Supervisor Sites 166
..... 166

7 Testing and Evaluation 168

7.1 Unit Testing 168
7.2 Functional Testing 176
7.3 Business Rules Testing 184
7.4 Integration Testing 189

8 Conclusion and Future Work 195

8.1 Conclusion 195
8.2 Future Work 196

8. References 197

9. Plagiarism 198

1 Introduction

The safety of a construction site depends on effective supervision and hazard management. However, manually monitoring multiple sites for safety compliance can be challenging and time-consuming. SafeSitePlus streamlines this process by automating PPE compliance detection, incident tracking, and risk prioritization, reducing the burden on supervisors.
With AI-powered hazard detection and automated alerts, supervisors receive only the most critical notifications, allowing them to focus on urgent safety concerns. The system also provides incident logging, analytics, and reporting tools, helping businesses track safety trends and improve compliance. By centralizing safety management, SafeSitePlus enables construction companies to enhance site security, reduce human effort, and improve overall operational efficiency.

1.1 Vision Statement

For construction site supervisors and stakeholders **Who** require enhanced safety monitoring and incident management, **The** SafeSite Plus **is** a comprehensive web-based platform **that** offers real-time monitoring, anomaly detection, incident reporting, and analytics capabilities. **Unlike** traditional manual supervision methods, **Our product** enables supervisors to remotely monitor construction sites, customize anomaly detection settings, and receive instant alerts for potential safety breaches. With SafeSite Plus, supervisors can address safety concerns, ensure compliance with standard operating procedures, and optimize construction site operations for enhanced productivity and worker well-being.

1.2 Related System Analysis/Literature Review

Existing PPE detection systems lack the capability to detect new anomalies or sudden unusual behaviors effectively, limiting their ability to adapt to evolving safety challenges on construction sites. Additionally, these systems may lack advanced analytics and reporting functionalities, hindering the ability to effectively track and analyze safety statistics for informed decision-making. Moreover, existing PPE detection systems do not provide prediction or forecasting of preventive measures, further limiting their ability to proactively address potential safety risks.

Table 1 Related System Analysis with proposed project solution

Application Name	Weakness	Proposed Project Solution
PPE detection	**1.** Offer less sufficient options to adapt safety protocols at construction sites **2.** Lack of advanced analytics and reporting functionalities to keep track of stats	**1.** Implement customizable safety protocols for construction sites. **2.** Integrate advanced analytics and reporting functionalities. **3.** Incorporate AI-based prediction and forecasting models to suggest preventive actions.

	3.Inability to predict or forecast potential risks or preventive actions.	

1.3 Project Deliverables

List down the project deliverables.

Table 2: Project Deliverables

PD-01	Scope
PD-02	SRS
PD-03	SDS Document and 30%
PD-04	Final report-1 and 60 %

1.4 System Limitations/Constraints

LI-1: Limited Accessibility: The effectiveness of the monitoring system may be restricted by factors such as internet connectivity and power supply availability at remote construction sites.
LI-2: Integration Challenges: Ensuring seamless integration with existing surveillance infrastructure and compatibility with various camera systems could pose technical challenges during implementation.
LI-3: Footage Delays: Due to network bandwidth limitations or camera processing speed, real-time video footage may experience delays, affecting the system's ability to provide immediate insights or alerts
LI-4: Data Storage Limitations: Storing large volumes of video footage, especially from multiple cameras, could strain local and cloud storage capacities, potentially leading to data retention issues or additional costs.

1.5 Tools and Technologies

Following are the tools and technologies that will be used to design and develop POPULA:

Table 3: Tools and Technologies for Proposed Project

	Tools	**Version**	**Rationale**
Tools And Technologies	Visual Studio Code	2015	IDE
	MongoDB	7.0	DBMS
	Figma	Latest	Design Work
	MS Word	360	Documentation
	MS Power Point	360	Presentation

	Postman	Latest	API testing
	Git	Latest	Version Control
	Docker	27.2	Deployment
	Technology	**Version**	**Rationale**
	Python	3.12.0	Programming Language
	JavaScript	2.2.0	Programming Language
	Node.js	21.1.0	Back-end Development
	React	0.72.6	Front-end Development
	Pytorch	1.0.2	Library
	TensorFlow	2.15.0	Library
	FastApi	0.111.0	Back-end Development
	OpenCv	4.9.0	Library
	Matplotlib	3.8.0	Visualization library
	AWS EC2	latest	Cloud computing service
	Yolov8	6.3	Model (CV)

1.6 Relevance to Course Modules

The SafeSitePlus project has strong relevance to several courses studied during a Bachelor of Computer Science (BSCS) program.

Web Development: Both frontend and backend aspects of **SafeSitePlus** leverage skills learned in web development courses. The **MERN stack** (MongoDB, Express.js, React, Node.js) is used to build an interactive user interface and ensure smooth data flow between system modules.

Database Management Systems: The project relies on **MongoDB** to store and manage critical data such as user profiles, site monitoring logs, incident reports, and system configurations. Proper indexing and data structuring techniques ensure efficient querying and retrieval.

Human-Computer Interaction (HCI): A user-friendly and intuitive dashboard for supervisors and administrators has been designed based on HCI principles. Features such as real-time alerts, clear data visualizations, and responsive UI enhance accessibility and usability.

Software Engineering Concepts: The development of SafeSitePlus follows a structured Software Development Life Cycle (SDLC), including requirements analysis, system design, implementation, and testing. Modular development ensures maintainability and scalability.

Artificial Intelligence & Machine Learning: The SafeSitePlus project integrates AI and ML techniques to enhance construction site monitoring. Machine learning algorithms are used for predictive analysis, anomaly detection, and incident classification. The system learns from historical data to identify patterns and potential safety risks, ensuring proactive hazard

management. Concepts such as supervised learning, classification models, and data-driven decision-making, covered in AI and ML courses, are applied.

Computer Vision: The anomaly detection module in SafeSitePlus leverages computer vision to analyze real-time video feeds from construction sites. By using image processing and deep learning models, the system detects hazardous conditions, unauthorized access, and safety violations. Key techniques, such as object detection, image segmentation. These concepts are directly drawn from Computer Vision coursework, enhancing the system's capability to ensure site safety.

2 Problem Definition

2.1 Problem Statement

On construction sites, safety violations are a significant concern, especially in regions like Pakistan, where many sites lack effective monitoring systems. Supervisors are often overwhelmed, as they are limited to manual observation or CCTV footage, which may not always catch critical safety issues like PPE (Personal Protective Equipment) non-compliance. Additionally, there is potential gap in observation and detection of critical incidents like fall detection or crowd analysis. These missed incidents pose serious risks to worker safety and can lead to delayed responses. Delays in incident reporting can lead to serious consequences, and supervisors have little to no support in identifying patterns or forecasting risks. With no comprehensive data consolidation or tools to help predict potential accidents, site safety remains reactive, leaving workers exposed to unnecessary risks. This highlights the pressing need for better safety oversight and proactive risk management.

2.2 Problem Solution

Our Construction Site Monitoring System (SafeSite-Plus) address these critical safety concerns on construction sites, a comprehensive monitoring system is essential. The proposed solution is to implement a real-time safety monitoring platform that integrates advanced camera systems, automated incident detection, and AI-driven anomaly detection. This system will enhance PPE compliance monitoring and detect incidents like fall detection or crowd analysis without relying solely on human observation. By consolidating data from multiple sources, the system will provide real-time alerts and assist supervisors in identifying trends through predictive analytics. With features like incident logging, automated reporting, and risk forecasting, the solution empowers supervisors to proactively manage safety on multiple sites simultaneously, reducing the burden of manual oversight. Furthermore, it will minimize delays in incident detection and reporting, ensuring immediate action to prevent accidents and protect workers. This comprehensive approach not only helps in early detection of lapses but also supports fostering a safer working environment on construction sites.

2.3 Objectives of the Proposed System

Below objectives will facilitate the marketing aspects of SafeSitePlus:

BO-1: Improve safety monitoring efficiency by automating processes, aiming to reduce reliance on manual supervision methods.

BO-2: Enhance safety practices through real-time monitoring and AI-based object detection to accurately identify safety anomalies.

BO-3: Enable proactive risk mitigation by implementing prompt incident reporting and SOP compliance tracking functionalities.

BO-4: Provide valuable insights for safety protocol optimization through statistical analysis of safety data.

BO-5: Ensure effective communication and access control with a robust notification system and user authentication mechanism.

2.4 Scope

Our project aims to provide construction site supervisors with a robust construction site monitoring system tailored to meet their specific needs and workflows. The scope of our project includes several features designed to increase user productivity, security, and efficiency. These features include a user-friendly interface optimized for ease of use and accessibility across devices, tools to keep users informed of ongoing construction activities and safety conditions, an incident reporting feature to facilitate the reporting and resolution of safety incidents, and notification options to alert users of critical events or security breaches. Additionally, the system will offer user management functions to control access and permissions. Furthermore, the system will provide comprehensive reporting and analysis tools to help users gain insights into security performance and identify areas for improvement. By equipping users with intuitive tools and valuable insights, our project aims to empower them to effectively manage construction sites and contribute to a safer work environment.

2.5 Modules

2.5.1 Module 1: User Administration

FE-1: User Management: Admins can manage user accounts, including the creation, updating, suspension, and deletion of accounts.
FE-2: Profile Management: Registered users (including supervisors and admin), can update their profile details through a self-service portal, keeping their personal and contact information up to date.

FE-3: Password Recovery: Users can recover lost passwords through a secure password recovery system using email verification.

2.5.2 Module 2: Site Management and Footage Assessment

FE-1: Site and Supervisor Management: Admins can register construction sites and assign supervisors to specific sites for improved monitoring and accountability.
FE-2: Reporting and Exporting: Allow admins to generate detailed reports on site assignments, supervisor activities, and footage analytics. Reports can be downloaded in Excel or PDF formats.
FE-3: Dashboard Navigation: Provide user-specific dashboards (e.g., Admin Dashboard for site and supervisor management, Supervisor Dashboard for monitoring assigned sites).

2.5.3 Module 3: Supervisor Monitoring Interface

FE-1: Video Monitoring: Supervisors can monitor video feeds from various construction site footages through the dashboard.
FE-2: Site-Specific Footages Filtering: Supervisors can filter camera views by criteria such as location or site status (e.g., active or closed). This enables focused monitoring of each site while allowing them to switch between different camera views for comprehensive oversight.
FE-3: Customizable Anomaly Parameters: Personalize detection settings for safety anomalies like SOP violations, Personal protective equipment breaches and fall detection based on site sensitivity.
FE-4: Task Management System: Supervisors have access to a digital notepad where they can record tasks, notes, and updates related to site management, reducing reliance on manual diaries. Tasks can be assigned deadlines and priorities.
FE-5: Weather Forecast: Integrate weather forecasting for better planning (construction site work might stop due to weather).
FE-6: Guide Manual & Help Tutorial: Provide an in-app usage tutorial or a video guide that explains how to navigate and use the app. App will also provide a manual for understanding Personal Protective Equipment (PPE) requirements and best practices to ensure on-site safety compliance.

2.5.4 Module 4: Video Processing and Frame Display via WebSockets

FE-1: Live Frame Processing with YOLOv8 Integration: Implemented a threaded video processing system where each uploaded video is analyzed frame-by-frame using the YOLOv8 model to detect safety violations like missing PPE or fall incidents.
FE-2: WebSocket-Based Frame Streaming with Error Handling: Used WebSockets to continuously send processed video frames (with bounding boxes and detections) to the frontend UI. Robust error handling ensures graceful disconnection and logs transmission issues.
FE-3: Anomaly Detection with Auto-Screenshot and Incident Logging: Detected anomalies (e.g., missing helmets, falls) trigger automatic alert generation, screenshot capture, and database logging for later incident reporting and visualization.

2.5.5 Module 5: Dataset Management and Preprocessing

This module focuses on handling and preparing the dataset for anomaly detection tasks.

FE-1: Data Set Annotation and Labeling: Gather existing datasets related to construction sites, ensuring diverse scenarios.Label the collected images for PPE, fall scenarios, and other hazards using tools like LabelImg or Roboflow.
FE-2: Data Augmentation: Apply augmentation techniques (e.g., flipping, rotation, brightness adjustment) to expand the dataset and simulate diverse conditions.
FE-3: Image Preprocessing: Preprocess images by resizing to a standard size, denoising, and applying contrast adjustments to ensure consistency and improve model performance.
FE-4: Combined Dataset Integration: Integrate annotated datasets for PPE detection, fall detection, and fire detection into a single, unified dataset. This ensures comprehensive training of the YOLOv8 model to detect multiple types of safety hazards effectively.

2.5.6 Module 6: Safety Anomaly Detection Framework

Leverages computer vision techniques and YOLOv8 for detecting anomalies in construction sites.

FE-1: Object Detection for PPE Compliance: Detect PPE elements like helmets, vests, gloves, and safety boots using YOLOv8, providing real-time alerts for violations.
FE-2: Fall Detection and Hazard Identification: Implement algorithms to detect falls and identify hazardous situations like fire, sending notifications for timely interventions.

2.5.7 Module 7: Incident Management

FE-1: Automated Incident Logging: Automatically log incidents by type, time, and location.
FE-2: Real-Time Notifications: Notify supervisors in real-time with video snippets and relevant data when an incident occurs.
FE-3: Prioritized Alerts: Use AI to prioritize incident alerts based on severity, enabling faster response to critical issues.
FE-4: Incident Categorization: Allow supervisors to manually categorize incidents to refine safety protocols and enhance system learning.
FE-5: Response Time Tracking and Escalation: Track supervisors' response times to incidents, with escalation to higher authorities if no response is received within a specified timeframe, ensuring timely action and accountability.

2.5.8 Module 8: Data Analytics / Suggestions and Forecasting

Consolidate data for detailed analysis, identify safety trends, and use predictive models to forecast potential risks and improve proactive safety measures.

FE-1: Data Consolidation and Storage: Consolidate data from multiple sources (e.g., camera feeds, incident reports, weather) into a centralized database for detailed analysis.

FE-2: Risk Prediction Using Scoring Algorithm: Use historical anomaly data, weather conditions, and site sensitivity to predict the overall safety risk level (e.g., Low, Medium, High) using a risk prediction algorithm based on a set of weighted features.
FE-3: Recommendations for Safety Improvement: Generate actionable safety recommendations for site supervisors, including adjusting site sensitivity, providing additional training on specific safety issues, operational changes, and weather-related actions based on the predicted risk level.
FE-4: Weather Data Integration and Impact Analysis: Integrate enhanced weather data with severity levels and risk factors, assessing how weather conditions (e.g., storms, heat) impact safety risks on construction sites.

2.5.9 Module 9: Reporting and Visualization

FE1: Automated Report Generation: Automatically generate detailed reports containing comprehensive analysis, statistics, and actionable recommendations based on detected anomalies and safety incident data.
FE2: Visual Data Representation: Create visualizations such as charts, graphs, or heat maps to represent safety-related data in an easily understandable format, using the appropriate method depending on what is needed.
FE-3: Report Handling: Users can regenerate reports due to updates or corrections and download reports for offline use, ensuring they have the latest, accurate information available for decision-making and record-keeping

2.5.10 Module 10: AI Assistance/Chatbot

The AI Assistant Chatbot in **SafeSitePlus** is designed to provide **instant support** to supervisors by answering queries related to company policies, work shifts, PPE compliance, emergency protocols, and more.

FE1: Chunking & Preprocessing: Breaks down large PDFs, policy documents, and company manuals into manageable text chunks for efficient retrieval. Cleans and structures data to improve search relevance and response accuracy.
FE2: Embedding Creation: Converts PDFs, URLs, and internal documents into vector embeddings for efficient semantic search.
FE-3: Query Processing & Intelligent Retrieval: Accepts natural language queries from supervisors and converts them into searchable vector representations. Preprocesses user input to remove ambiguity, correct errors, and enhance retrieval precision. Matches queries with the most relevant document chunks based on semantic similarity. Ranks and selects the top results before sending them to the LLM for context-aware response generation.
FE4: Integration with LLM (Gemini, OpenAI, etc.): Sends the selected text chunks to an LLM (e.g., Gemini, OpenAI, or local models) for context-aware response generation. Ensures that responses are accurate, well-structured, and aligned with company policies.

3 Requirement Analysis

The requirement analysis of SafeSitePlus is given below:

3.1 User classes and characteristics

Table 4 Shows user classes and characteristic for SafeSite Plus

User Class	Description
Admin	The Admin is responsible for managing user accounts, configuring system settings, and overseeing notifications within the Safesiteplus system. The Admin sets notification preferences and ensures that supervisors are promptly alerted to any safety incidents. They can also review system logs to monitor recent activity. Admins are typically safety managers or system operators, familiar with basic software interfaces, and require only minimal training to use the system efficiently.
Supervisor	The Supervisor is primarily involved in monitoring real-time safety at construction sites. They use the system to view live camera feeds, configure site-specific anomaly detection parameters, and receive notifications regarding potential safety violations. Supervisors manage task assignments related to safety and inspect safety incidents. They review weather forecasts to anticipate safety issues related to weather conditions. Most Supervisors are construction managers or safety officers who need only minimal guidance on how to use the system's interface due to its intuitive design.

3.2 Requirement Identifying Technique

The following are the requirement identifying techniques used for **SafeSitePlus**:

- **Brainstorming**
  Collaborative brainstorming sessions were conducted with team members, construction site supervisors, and safety experts to identify key system requirements.

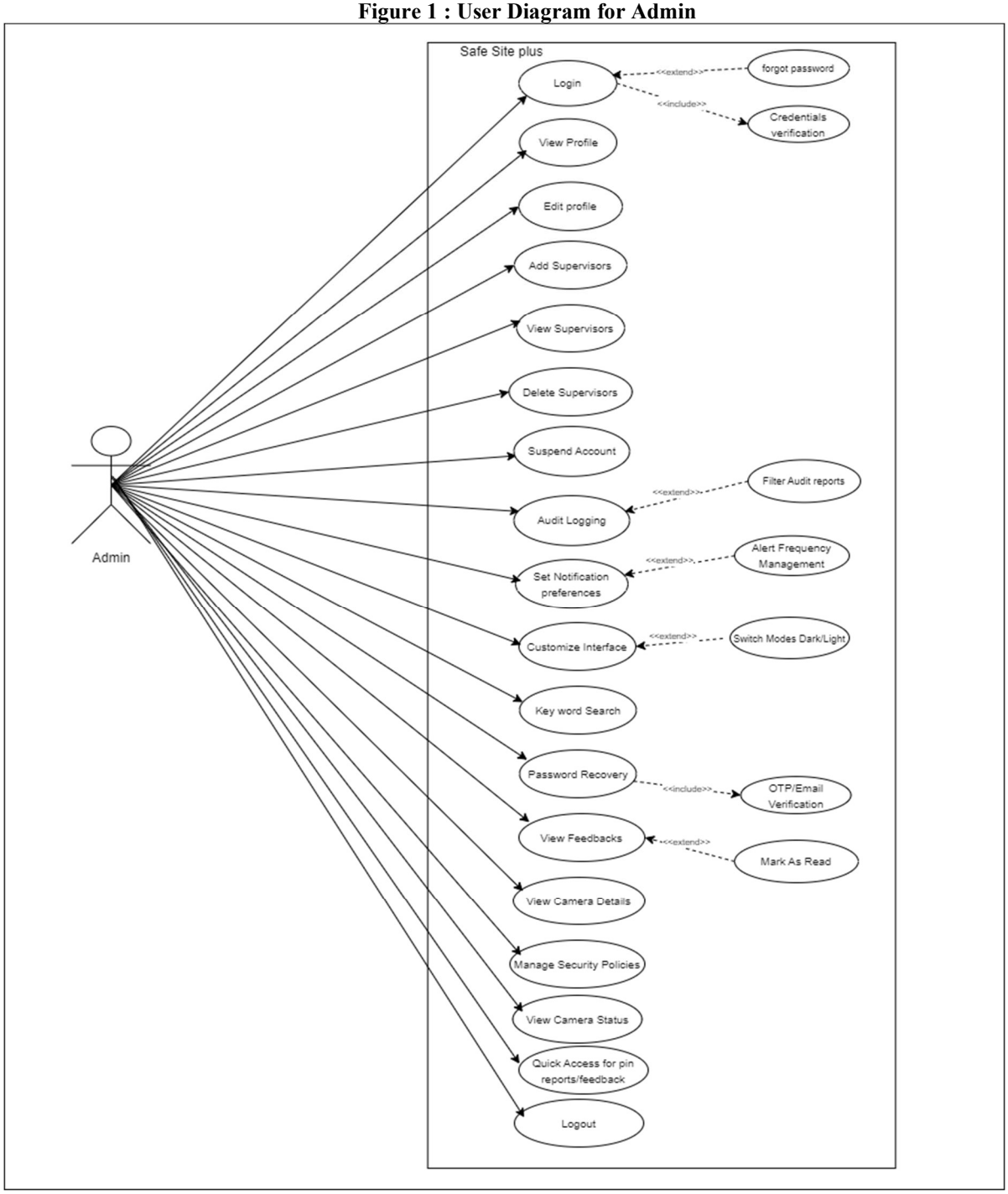
- **Interviews**
  Structured interviews were held with site supervisors and administrators to understand the challenges of construction site monitoring, incident reporting, and safety management.

- **Discussions**
  Discussions took place with industry professionals and safety compliance officers to refine system requirements and ensure alignment with construction safety standards.

3.2.1 User Case Diagram

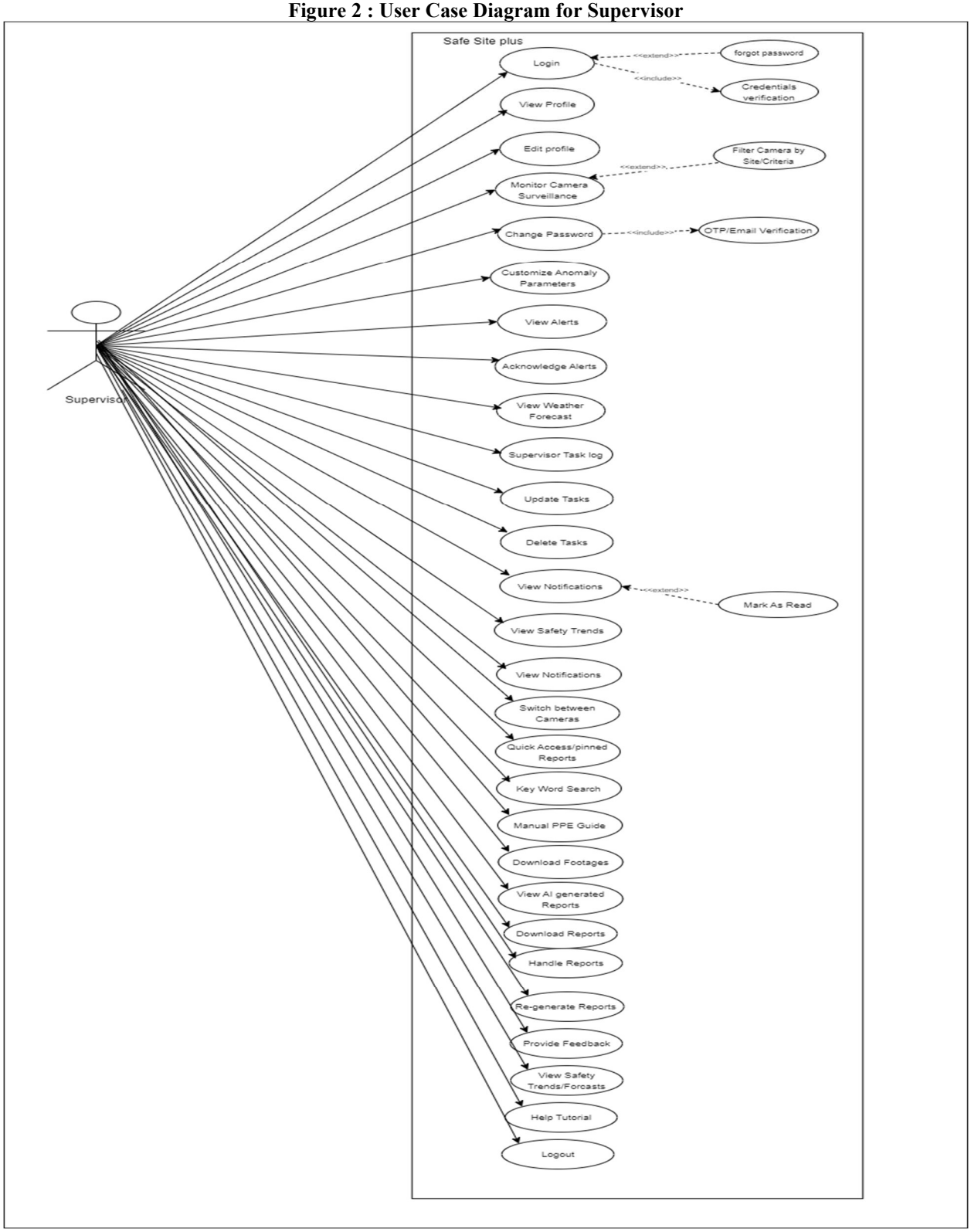
3.2.1.1 Admin

Figure 1 : User Diagram for Admin



3.2.1.2 Supervisor

Figure 2 : User Case Diagram for Supervisor



3.2.2 User Case Tabular

UC-1: Login

In this use case, a user accesses the system and logs in using their username and password. The system validates the credentials and provides access to the dashboard based on the user role (Admin or Supervisor).

Table 5: UC1-Login

Field	Details
Actors	Admin, Supervisor
Trigger	The user initiates login by entering credentials.
Preconditions	1. User must be registered in the system. 2. System is active.
Postconditions	1. User is authenticated and redirected to the appropriate dashboard.
Normal Flow	1. User enters username and password. 2. System validates credentials. 3. User is logged in and session is initiated.
Alternative Flows	None
Exceptions	1. If credentials are invalid, the system shows an error message.
Business Rules	1.Only users with valid accounts can log in.
Assumptions	The user has an active internet connection

UC-2: Forgot Password

In this use case, a user requests to reset their password. The system verifies the user's email and sends a recovery link for resetting the password.

Table 6: UC-2 Forgot Password

Field	Details
Actors	Admin, Supervisor
Trigger	User selects the "Forgot Password" option.
Preconditions	1. User must be registered in the system. 2. System is active.
Postconditions	1. Password reset link is sent to the registered email.

Normal Flow	1. User clicks “Forgot Password”. 2. System prompts for email. 3. User provides email. 4. System sends a password reset link to the email.
Alternative Flows	None
Exceptions	1. If email is not registered, the system displays an error message.
Business Rules	BR-1: The password reset link is valid for 15 minutes.
Assumptions	The email system is functional and secure.

UC-3: View Profile
In this use case, a user views their profile information such as name, email, role, and contact details.

Table 7: UC-3 View Profile

Field	Details
Actors	Admin, Supervisor
Trigger	User selects the option to view their profile.
Preconditions	1. User must be logged in. 2. Profile information must exist.
Postconditions	1. Profile information is displayed.
Normal Flow	1. User navigates to the profile section. 2. System retrieves and displays profile information.
Alternative Flows	If the user wants to edit their profile information, they can click on the “Edit Profile” button and make changes to their information.
Exceptions	None
Business Rules	Profile data should be accessible based on user role and permissions.
Assumptions	Profile data is up-to-date and accurately stored in the database.

UC-4: Edit Profile

Table 8: UC-2 Edit Profile

Use Case ID	UC-4
Use Case Name	Edit Profile
Actors	Primary Actor: Admin, Supervisor

	Secondary Actor: None
Description	The Admin or Supervisor edits their personal profile information, including updating their email, phone number, or password.
Trigger	The Admin or Supervisor clicks the “Edit Profile” option on the profile page.
Preconditions	PRE-1: The user must be logged in to access the “Edit Profile” feature.
Postconditions	POST-1: The system updates the user’s profile information and stores it.
Normal Flow	1. The user navigates to the profile page.
	2. The user clicks on the “Edit Profile” button.
	3. The user updates the desired profile fields such as phone number, email, or password. 4.The user clicks the “Save” button. 5.The system validates the new information. 6. The system saves the updated information and confirms the update.
Alternative Flows	Password Update: The user enters a new password and re-enters it for confirmation. The system checks if the two passwords match. If they match, the system updates the password and confirms.
Exceptions	EX-1: If the chosen email is already registered, the system notifies the user. EX-2: If the passwords do not match, the system asks the user to re-enter the password.
Business Rules	BR-1: Email addresses must be unique across the system. BR-2: The password must meet security criteria (e.g., length, complexity).
Assumptions	ASSUM-1: The user has an active internet connection.

UC-5: Change Password

In this use case, a user updates their password by providing the old password and entering a new one.

Table 9: UC-5 Change Password

Use Case ID	**UC-5**
Use Case Name	Change Password
Actors	Primary Actor: Admin, Supervisor
Description	Users can change their password after logging in to the system. The system verifies the old password before allowing the user to set a new password.
Trigger	User clicks the “Change Password” option in their profile settings.
Preconditions	1. User must be logged in to the system. 2. User must know their current password.

Postconditions	1. User successfully updates their password. 2. User is logged out and required to log in again with the new password.
Normal Flow	1. User navigates to the profile settings and selects the "Change Password" option. 2. User enters their current password and new password. 3. System verifies the current password. 4. If verified, the system updates the password and logs the user out. 5. User logs in again with the new password.
Alternative Flows	1.1 Incorrect current password: system prompts the user to enter the correct password.
Exceptions	1. Password doesn't meet criteria: system shows an error message for password strength (e.g., weak password).

UC-6: Quick Access

In this use case, users can quickly access pinned reports or feedback items, allowing them to view critical information without searching.

Table 10: UC-6 Quick Access

Field	**Details**
Actors	Admin, Supervisor
Trigger	User selects an item from their quick access list.
Preconditions	1. User must have pinned items (reports/feedback).
Postconditions	1. Selected item is displayed in detail.
Normal Flow	1. User clicks on the quick access section. 2. System displays pinned items. 3. User selects an item to view.
Alternative Flows	None
Exceptions	1. If no items are pinned, the system displays an appropriate message.
Business Rules	Items should be updated in real-time to reflect the most recent pins.
Assumptions	Users regularly pin important items for quick access.

UC-7: View Notifications
In this use case, users can view system-generated notifications related to alerts, reports, or system status.

Table 11: UC-7 View Notifications

Field	Details
Actors	Admin, Supervisor
Trigger	User clicks on the notifications icon.
Preconditions	1. User must be logged in. 2. Notifications must exist for the user.
Postconditions	1. Notifications are marked as "read" after viewing.
Normal Flow	1. User clicks the notifications icon. 2. System displays all unread notifications. 3. User views the notification details.
Alternative Flows	None
Exceptions	1. If no notifications exist, the system displays a "No notifications" message.
Business Rules	Notifications should be timely and relevant to the user's role.
Assumptions	Notifications are generated accurately based on system events.

UC-8: Keyword Search
In this use case, users can search through the system for specific reports, feedback, or other relevant data using keywords.

Table 12: UC-8 Keyword Search

Field	Details
Actors	Admin, Supervisor
Trigger	User enters a search term in the search bar.
Preconditions	1. User must be logged in. 2. Searchable data must exist.
Postconditions	1. Search results are displayed.

Normal Flow	1. User enters a keyword in the search bar. 2. System retrieves matching results from reports, feedback, etc. 3. Search results are displayed to the user.
Alternative Flows	None
Exceptions	1. If no matching results are found, the system shows “No results found.”
Business Rules	Search functionality should support various data types and be fast.
Assumptions	Search index is kept up to date for optimal performance.

UC-9: Logout
In this use case, the user logs out of the system, terminating the session and returning to the login screen.

Table 13: UC-9 Logout

Use Case ID	**UC-9**
Use Case Name	Logout
Actors	Primary Actor: Admin, Supervisor
Description	The Admin or Supervisor logs out from the system to terminate their session and remove access to their features.
Trigger	The Admin or Supervisor clicks the “Logout” button.
Preconditions	PRE-1: The user must be logged in to the system.
Postconditions	POST-1: The user is successfully logged out, and the session is terminated.
	POST-2: The system logs the logout event.
Normal Flow	1. The user clicks the “Logout” button on the system interface.

UC-10: Register Supervisor
In this use case, the admin registers a new supervisor by entering the required details such as name, email, and role-specific information.

Table 14: UC-10 Register Supervisor

Field	**Details**
Use Case ID	**UC-10**

Use Case Name	Register Supervisor
Actors	Primary Actor: Admin
Description	Admin registers a new supervisor by entering the supervisor's details and assigning them appropriate roles and permissions.
Trigger	Admin clicks on "Add Supervisor" in the User Management section.
Preconditions	1. Admin is logged in. 2. Supervisor details (name, email, role) are provided.
Postconditions	1. Supervisor is successfully registered and notified via email.
Normal Flow	1. Admin navigates to User Management and selects "Register Supervisor." 2. Admin enters the required details and assigns permissions. 3. System verifies the data and registers the new supervisor. 4. Supervisor receives a confirmation email with login credentials.
Alternative Flows	None.

UC-11: View Supervisor Detail
In this use case, the admin views detailed information of a specific supervisor, including their activity and account status.

Table 15: UC-11 Supervisor Detail

Field	**Details**
Actors	Admin
Trigger	Admin selects a supervisor from the list to view details.
Preconditions	1. Admin must be logged in. 2. Supervisors must be registered in the system.
Postconditions	1. Supervisor details are displayed.
Normal Flow	1. Admin navigates to the supervisor list. 2. Admin selects a supervisor. 3. System displays the supervisor's detailed profile.
Alternative Flows	None
Exceptions	1. If the supervisor profile is unavailable, the system shows an error.
Business Rules	Supervisor details must be accurate and up-to-date.

Assumptions	Admin regularly monitors supervisor activity.

UC-12: Suspend Supervisor Account
In this use case, the admin suspends a supervisor's account, preventing them from accessing the system.

Table 16: UC-12 Suspend Supervisor Account

Field	Details
Actors	Admin
Trigger	Admin selects the option to suspend a supervisor's account.
Preconditions	1. Admin must be logged in. 2. Supervisor account must exist.
Postconditions	1. Supervisor account is suspended and access is restricted.
Normal Flow	1. Admin navigates to the supervisor list. 2. Admin selects the option to suspend the account. 3. System confirms and suspends the supervisor's account.
Alternative Flows	None
Exceptions	None
Business Rules	Suspended accounts must not have access to any part of the system.
Assumptions	Admin may suspend accounts for inactivity or rule violations.

UC-13: Delete Supervisor Account
In this use case, the admin permanently deletes a supervisor's account from the system.

Table 17: UC-13 Delete Supervisor Account

Field	Details
Actors	Admin
Trigger	Admin selects the option to delete a supervisor's account.
Preconditions	1. Admin must be logged in. 2. Supervisor account must exist.
Postconditions	1. Supervisor account is permanently removed from the system.

Normal Flow	1. Admin navigates to the supervisor list. 2. Admin selects the option to delete the account. 3. System confirms and deletes the supervisor's account.
Alternative Flows	None
Exceptions	1. If there are unresolved incidents, the system prompts the admin.
Business Rules	Deleted accounts cannot be recovered.
Assumptions	Admin has the authority to delete supervisor accounts.

UC-14: Audit Logging
In this use case, the admin reviews the audit logs to monitor user activities, track login attempts, and other system actions.

Table 18: UC-14 Audit Logging

Field	Details
Actors	Admin
Trigger	Admin selects the option to view audit logs.
Preconditions	1. Admin must be logged in. 2. Audit logging must be enabled.
Postconditions	1. Audit logs are displayed with timestamps and user actions.
Normal Flow	1. Admin navigates to the audit logs section. 2. System retrieves and displays user activities.
Alternative Flows	None
Exceptions	1. If logs are not available, the system shows an error message.
Business Rules	Logs must be immutable to ensure integrity.
Assumptions	The system captures all relevant activities in audit logs.

UC-15: View Cameras Details
In this use case, the admin views detailed information about the cameras in the system, including location, status, and type.

Table 19: UC-15 View Cameras Details

Field	Details

Actors	Admin
Trigger	Admin selects a camera to view details.
Preconditions	1. Cameras must be registered in the system. 2. Admin must be logged in.
Postconditions	1. Camera details are displayed, including current status.
Normal Flow	1. Admin navigates to the camera list. 2. Admin selects a camera. 3. System displays the camera details.
Alternative Flows	None
Exceptions	1. If the camera is offline, the system shows an error message.
Business Rules	Only registered cameras should appear in the system.
Assumptions	Camera details are kept updated in real-time.

UC-16: View Camera Status

In this use case, the admin checks the operational status of all cameras, including online/offline status and recent activity.

Table 20: UC-16 View Camera Status

Field	Details
Actors	Admin
Trigger	Admin selects the option to view camera statuses.
Preconditions	1. Cameras must be operational. 2. Admin must be logged in.
Postconditions	1. Camera status (online/offline) is displayed.
Normal Flow	1. Admin navigates to the camera status section. 2. System retrieves and displays the status of all cameras.
Alternative Flows	None
Exceptions	1. If the system cannot fetch the status, an error is displayed.
Business Rules	The system should regularly update camera status.
Assumptions	Camera statuses are monitored in real-time.

UC-17: Customize Interface
In this use case, the admin customizes the system interface by changing themes, layouts, or enabling specific widgets.

Table 21: UC-17 Customize Interface

Field	Details
Actors	Admin
Trigger	Admin selects the option to customize the interface.
Preconditions	1. Admin must be logged in. 2. Customization options must be available.
Postconditions	1. Interface customization settings are saved.
Normal Flow	1. Admin navigates to the interface customization section. 2. Admin selects desired theme or layout changes. 3. System saves and applies the changes.
Alternative Flows	None
Exceptions	1. If customization fails, the system shows an error message.
Business Rules	Interface changes should not disrupt system functionality.
Assumptions	Admins have customization privileges.

UC-18: View Feedback
In this use case, the admin reviews feedback from supervisors or other users to address any issues or suggestions.

Table 22: UC-18 View Feedback

Field	Details
Actors	Admin
Trigger	Admin selects the option to view feedback.
Preconditions	1. Feedback must exist in the system. 2. Admin must be logged in.
Postconditions	1. Feedback is displayed for review.
Normal Flow	1. Admin navigates to the feedback section. 2. System retrieves and displays feedback details.

Alternative Flows	None
Exceptions	1. If no feedback exists, the system displays a notification.
Business Rules	Feedback should be reviewed regularly for improvements.
Assumptions	The system captures relevant feedback for the admin to review.

UC-19: Monitor Camera Surveillance
In this use case, the supervisor monitors live surveillance footage from the system's cameras to ensure site safety.

Table 23: UC-19 Monitor Camera Surveillance

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to view live camera feeds.
Preconditions	1. Supervisor must be logged in. 2. Cameras must be connected and operational.
Postconditions	1. Live camera feed is displayed for monitoring.
Normal Flow	1. Supervisor navigates to the surveillance section. 2. Supervisor selects a camera to monitor. 3. System displays the live feed from the selected camera.
Alternative Flows	None
Exceptions	1. If the camera is offline, the system shows an error message.
Business Rules	Live feeds should be real-time with minimal lag.
Assumptions	Supervisor regularly monitors site activity via live feeds.

UC-20: Switch Between Cameras
In this use case, the supervisor switches between different camera feeds to monitor various locations.

Table 24: UC-20 Switch between Cameras

Field	Details
Actor	Supervisor

Trigger	Supervisor selects the option to switch to a different camera feed.
Preconditions	1. Supervisor must be logged in. 2. Multiple cameras must be available and operational.
Postconditions	1. The feed from the selected camera is displayed.
Normal Flow	1. Supervisor navigates to the camera list. 2. Supervisor selects a different camera. 3. System switches and displays the feed from the new camera.
Alternative Flows	None
Exceptions	1. If the selected camera is offline, an error message is shown.
Business Rules	Supervisors should be able to switch cameras with minimal delays.
Assumptions	Multiple cameras are installed across the site.

UC-21: Guide (Manual) for PPE Usage
In this use case, the supervisor accesses a guide or manual on the proper use of Personal Protective Equipment (PPE) for workers.

Table 25: UC-21 Guide(Manual) for PPE Usage

Field	**Details**
Actor	Supervisor
Trigger	Supervisor selects the option to view the PPE usage guide.
Preconditions	1. Supervisor must be logged in. 2. PPE guidelines must be uploaded to the system.
Postconditions	1. PPE usage guide is displayed.
Normal Flow	1. Supervisor navigates to the PPE guide section. 2. System retrieves and displays the PPE usage manual.
Alternative Flows	None
Exceptions	1. If the guide is unavailable, an error message is shown.
Business Rules	The PPE guide must be kept up-to-date based on industry standards.
Assumptions	Supervisors use the guide to enforce proper PPE usage on-site.

UC-22: Customize Anomaly Parameters
In this use case, the supervisor customizes the parameters for detecting safety anomalies (e.g., PPE compliance, unauthorized access) based on site-specific needs.

Table 26: UC-22 Customize Anomaly Parameters

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to customize anomaly detection parameters.
Preconditions	1. Supervisor must be logged in. 2. System must allow configurable anomaly parameters.
Postconditions	1. Anomaly detection parameters are updated and saved.
Normal Flow	1. Supervisor navigates to the anomaly parameters section. 2. Supervisor selects and adjusts the desired parameters. 3. System applies and saves the new parameters.
Alternative Flows	None
Exceptions	1. If parameter changes fail to save, the system shows an error.
Business Rules	Anomaly parameters should be flexible to adapt to changing safety requirements.
Assumptions	Supervisors regularly update anomaly detection based on site conditions.

UC-23: View Weather Forecast
In this use case, the supervisor views the weather forecast to anticipate conditions that may affect site safety.

Table 27: UC-23 View Weather Forecast

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to view the weather forecast.
Preconditions	1. Supervisor must be logged in. 2. Weather data must be integrated with the system.
Postconditions	1. Weather forecast is displayed.
Normal Flow	1. Supervisor navigates to the weather forecast section. 2. System retrieves and displays the latest weather forecast.

Alternative Flows	None
Exceptions	1. If weather data is unavailable, an error message is shown.
Business Rules	Weather data should be accurate and updated in real-time.
Assumptions	Weather conditions are monitored regularly to ensure site safety.

UC-24: View Alerts
In this use case, the supervisor views safety alerts triggered by anomalies or system-generated warnings.

Table 28: UC-24 View Alerts

Field	**Details**
Actor	Supervisor
Trigger	Supervisor receives or selects the option to view alerts.
Preconditions	1. Supervisor must be logged in. 2. Anomalies must be detected in the system.
Postconditions	1. Alerts are displayed for review.
Normal Flow	1. Supervisor navigates to the alerts section. 2. System retrieves and displays current alerts.
Alternative Flows	None
Exceptions	None
Business Rules	Alerts should be prominently displayed and prioritized based on severity.
Assumptions	Supervisors regularly check and respond to alerts to maintain safety.

UC-25: Acknowledge Alerts
In this use case, the supervisor acknowledges safety alerts after reviewing and addressing the issues raised.

Table 29: UC-25 Acknowledge Alerts

Field	**Details**
Actor	Supervisor
Trigger	Supervisor selects the option to acknowledge an alert.

Preconditions	1. Supervisor must be logged in. 2. Alerts must be displayed in the system.
Postconditions	1. The alert is marked as acknowledged and resolved.
Normal Flow	1. Supervisor reviews the alert. 2. Supervisor selects the option to acknowledge the alert. 3. System marks the alert as resolved.
Alternative Flows	None
Exceptions	1. If acknowledgment fails, the system displays an error.
Business Rules	Alerts should only be acknowledged after resolution of the issue.
Assumptions	Supervisors promptly respond to alerts for site safety.

UC-26: Download Footages
In this use case, the supervisor downloads video footage from specific cameras for further review or record-keeping.

Table 30: UC-26 Download Footages

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to download footage.
Preconditions	1. Supervisor must be logged in. 2. The camera must have recorded footage.
Postconditions	1. The selected footage is downloaded.
Normal Flow	1. Supervisor navigates to the footage section. 2. Supervisor selects a time range or specific camera. 3. System prepares and downloads the footage.
Alternative Flows	None
Exceptions	1. If the footage is unavailable, an error message is shown.
Business Rules	Footage must be securely stored and downloaded.
Assumptions	Supervisors may download footage for evidence or review purposes.

UC-28: View Automated Generated Reports
In this use case, the supervisor views AI-generated safety reports based on camera surveillance and system data.

Table 31: UC-28 View Automated Generated Reports

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to view reports.
Preconditions	1. Supervisor must be logged in. 2. Reports must be generated by the system.
Postconditions	1. Automated reports are displayed.
Normal Flow	1. Supervisor navigates to the reports section. 2. System retrieves and displays AI-generated reports.
Alternative Flows	None
Exceptions	1. If no reports are available, the system displays a notification.
Business Rules	Reports must be accurate and reflect current site conditions.
Assumptions	Supervisors rely on reports for insights into safety compliance.

UC-28: Handle Reports (Approve/Disapprove AI-Generated Reports)
In this use case, the supervisor reviews AI-generated safety reports and has the option to approve or disapprove them.

Table 32: UC-28 Handle Reports

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to approve or disapprove a report.
Preconditions	1. Supervisor must be logged in. 2. AI-generated reports must be available.
Postconditions	1. Report is either approved or disapproved based on supervisor's input.
Normal Flow	1. Supervisor navigates to the reports section. 2. Supervisor reviews a report. 3. Supervisor approves or disapproves the report. 4. System logs the supervisor's decision.

Alternative Flows	None
Exceptions	1. If a report is not accessible, the system shows an error message.
Business Rules	Only approved reports should be available for further actions like downloading or regenerating.
Assumptions	Supervisors are responsible for ensuring the accuracy of AI-generated reports.

UC-29: Regenerate Reports
In this use case, the supervisor regenerates reports if the previous AI-generated reports are outdated or contain errors.

Table 33: UC-29 Regenerate Reports

Field	**Details**
Actor	Supervisor
Trigger	Supervisor selects the option to regenerate a report.
Preconditions	1. Supervisor must be logged in. 2. A previously generated report must be available.
Postconditions	1. The report is regenerated and updated based on the latest data.
Normal Flow	1. Supervisor navigates to the reports section. 2. Supervisor selects a report to regenerate. 3. System regenerates the report based on the current data. 4. Updated report is displayed to the supervisor.
Alternative Flows	None
Exceptions	1. If regeneration fails, the system shows an error message.
Business Rules	Regenerated reports must reflect the latest available data.
Assumptions	Supervisors can request report regeneration as new site data becomes available.

UC-30: Download Reports
In this use case, the supervisor downloads AI-generated or approved reports for record-keeping or further analysis.

Table 34: UC-30 Download Reports

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to download a report.
Preconditions	1. Supervisor must be logged in. 2. The report must be generated or approved.
Postconditions	1. The selected report is downloaded to the supervisor's device.
Normal Flow	1. Supervisor navigates to the reports section. 2. Supervisor selects a report. 3. System prepares and downloads the report.
Alternative Flows	None
Exceptions	1. If the report is not available for download, the system shows an error.
Business Rules	Reports should be downloadable in standard formats like PDF or CSV.
Assumptions	Supervisors download reports for official purposes or site analysis.

UC-31: Provide Feedback to Admin
In this use case, the supervisor provides feedback to the admin regarding the system's performance or incidents on the site.

Table 35: UC-31 Provide Feedback to Admin

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to send feedback to the admin.
Preconditions	1. Supervisor must be logged in. 2. Feedback form must be available.
Postconditions	1. Feedback is submitted and sent to the admin for review.

Normal Flow	1. Supervisor navigates to the feedback section. 2. Supervisor fills out the feedback form. 3. System submits the feedback to the admin.
Alternative Flows	None
Exceptions	1. If feedback fails to send, the system shows an error message.
Business Rules	Supervisors should have a dedicated channel to submit feedback easily.
Assumptions	Supervisors provide constructive feedback to improve site safety or system performance.

UC-32: View Safety Trends/Forecasts
In this use case, the supervisor views AI-generated trends and forecasts related to site safety based on historical data.

Table 36: UC-32 View Safety Trends/ Forecasts

Field	Details
Actor	Supervisor
Trigger	Supervisor selects the option to view safety trends or forecasts.
Preconditions	1. Supervisor must be logged in. 2. Trends and forecasts must be generated by the system.
Postconditions	1. Safety trends and forecasts are displayed to the supervisor.
Normal Flow	1. Supervisor navigates to the trends or forecast section. 2. System retrieves and displays relevant trends based on past data. 3. Supervisor analyzes the displayed trends.
Alternative Flows	None
Exceptions	1. If trends or forecasts are unavailable, the system displays a notification.
Business Rules	Forecasts must be based on accurate data and algorithms.
Assumptions	Supervisors use forecasts to predict and prevent safety incidents.

UC-33: Give Usage/Help Tutorial
In this use case, the supervisor accesses a help tutorial or usage guide for understanding system functions.

Table 37: UC-33 Give Usage/Help Tutorial

Field	Details
Actor	Supervisor

Trigger	Supervisor selects the option to view a help tutorial or usage guide.
Preconditions	1. Supervisor must be logged in. 2. Help documentation or tutorial must be available in the system.
Postconditions	1. Tutorial or guide is displayed to the supervisor.
Normal Flow	1. Supervisor navigates to the help section. 2. Supervisor selects the relevant tutorial or guide. 3. System displays the selected guide.
Alternative Flows	None
Exceptions	1. If the guide is unavailable, the system displays an error message.
Business Rules	Help guides must be regularly updated to reflect the system's latest features.
Assumptions	Supervisors may use the guide to understand new or complex system features.

UC-34: Supervisor's Task Log
In this use case, the supervisor reviews a log of tasks they have completed or are pending within the system.

Table 38: UC-29 Supervisor's Task Log

Field	**Details**
Actor	Supervisor
Trigger	Supervisor selects the option to view their task log.
Preconditions	1. Supervisor must be logged in. 2. The system must track and log tasks assigned or completed by the supervisor.
Postconditions	1. Task log is displayed showing both completed and pending tasks.
Normal Flow	1. Supervisor navigates to the task log section. 2. System displays a list of tasks. 3. Supervisor can view or manage tasks from the log.
Alternative Flows	None
Exceptions	1. If the task log is empty or fails to load, the system shows an error.
Business Rules	Tasks must be accurately logged and easily retrievable for review.
Assumptions	Supervisors may regularly review their task history to track progress.

UC-35: View Visualization/Graph

Table 39: UC35-View Visualization/Graph

Field	Details
Actors	Supervisor
Trigger	The supervisor initiates the action to view graphical representations of anomalies and trends.
Preconditions	1. The supervisor is logged into the system. 2. The system has recorded data on anomalies and trends.
Postconditions	1. Supervisor is able to view graphical representations of anomalies and trends. 2. Data is displayed in a user-friendly, visual format.
Normal Flow	1. Supervisor navigates to the visualization/graph section. 2. System retrieves data on anomalies and trends. 3. System generates graphical representations. 4. Supervisor views the generated graphs.
Alternative Flows	1. If no data is available, the system shows a "No Data Available" message.
Exceptions	1. If there is a system error retrieving data, an error message is displayed. 2. If the supervisor's permissions are insufficient, access is denied.
Business Rules	1. Only supervisors with valid permissions can access visualization features. 2. Data shown must be relevant to the specific sites the supervisor monitors.
Assumptions	1. The supervisor has an active internet connection. 2. Data on anomalies and trends is regularly updated in the system.

UC-36: AI Assistance/Chatbot

Field	Details
Actors	Supervisor
Trigger	Supervisor asks a query in the chatbot interface.
Preconditions	1. Supervisor must be logged into the system. 2. Relevant documents (policies, guidelines) must be embedded and stored.
Postconditions	1. AI Assistant provides an accurate response based on the available data. 2. The response is logged for future improvements.
Normal Flow	1. Supervisor opens the AI Assistant module. 2. Supervisor types or speaks a query (e.g., "What are the PPE requirements?"). 3. The system processes the query and converts it into a searchable format. 4. The system retrieves relevant document chunks and ranks them based on relevance. 5. The best-matched information is sent to the LLM (e.g., Gemini, OpenAI) for response generation. 6. AI Assistant displays a structured response to the supervisor.
Alternative Flows	1. If no exact match is found, the system suggests closely related topics. 2. If the supervisor asks a follow-up query, the chatbot maintains context for a relevant response.

Exceptions	1. If the system fails to retrieve relevant data, it informs the supervisor and provides alternative ways to find the information (e.g., manual search). 2. If the system is offline, an error message is displayed.
Business Rules	1. Responses must be accurate and aligned with company policies. 2. Confidential data should not be disclosed unless the supervisor has the correct access level.
Assumptions	1. Supervisors prefer instant, automated responses over manual document searching. 2. The chatbot system improves over time by learning from interactions.

3.3 Functional Requirements

3.3.1 UFR-01: Login.Input.Username

Table 40: Login.Input.Username

Identifier	**FR-1**
Title	Login.Input.Username
Requirement	The user shall be able to input their username for login.
Source	Zamin
Rationale	Users need to provide their username to authenticate.
Business Rule	BR-1: Username is required for login.
Dependencies	None
Priority	High

3.3.1.1 FR-02: Login.Input.Password

Table 41: Login.Input.Password

Identifier	**FR-2**
Title	Login.Input.Password

Requirement	The user shall be able to input their password for login.
Source	Zamin
Rationale	Users need to provide their password to authenticate.
Business Rule	BR-2: Password is required for login.
Dependencies	None
Priority	High

3.3.1.2 FR-03: Login.Validate.Username

Table 42: Login.Validate.Username

Identifier	**FR-3**
Title	Login.Validate.Username
Requirement	The system shall validate the entered username against the user database.
Source	Functional Specification
Rationale	To ensure the username exists in the database.
Business Rule	BR-3: Only registered usernames can be validated.
Dependencies	FR-1
Priority	High

3.3.1.3 FR-04: Login.Validate.Password

Table 43: Login.Validate.Password

Identifier	**FR-4**
Title	Login.Validate.Password

Requirement	The system shall validate the entered password against the stored password for the username.
Source	Functional Specification
Rationale	To ensure the password matches the username for authentication.
Business Rule	BR-4: Password must correspond to the validated username.
Dependencies	FR-2, FR-3
Priority	High

3.3.1.4 FR-05: Login.Authenticate

Table 44: Login.Authenticate

Identifier	**FR-5**
Title	Login.Authenticate
Requirement	The system shall authenticate the user after validating username and password.
Source	Functional Specification
Rationale	To grant access based on valid credentials.
Business Rule	BR-5: Access is granted only upon successful authentication.
Dependencies	FR-3, FR-4
Priority	High

3.3.1.5 FR-06: Login.Failed

Table 45: Login.Failed

Identifier	**FR-6**
Title	Login.Failed
Requirement	The system shall display an error message if authentication fails.
Source	Zamin

Rationale	To inform users about failed login attempts.
Business Rule	BR-6: A failed attempt triggers an error message.
Dependencies	FR-5
Priority	Medium

3.3.1.6 FR-07: Login.Successful

Table 46: Login.Successful

Identifier	**FR-7**
Title	Login.Successful
Requirement	The system shall provide a confirmation message upon successful login.
Source	Zamin
Rationale	To indicate a successful login to the user.
Business Rule	BR-7: Confirmation message displayed on successful login.
Dependencies	FR-5
Priority	High

3.3.1.7 FR-08: Login.SessionCreation

Table 47: Login.SessionCreation

Identifier	**FR-8**
Title	Login.SessionCreation
Requirement	The system shall create a user session upon successful login.
Source	Functional Specification
Rationale	To maintain user state during their session.

Business Rule	BR-8: User sessions must be created for authenticated users.
Dependencies	FR-7
Priority	High

3.3.1.8 FR-09: Login.RedirectToDashboard

Table 48: Login.RedirectToDashboard

Identifier	**FR-9**
Title	Login.RedirectToDashboard
Requirement	The system shall redirect the user to the dashboard after successful login.
Source	Functional Specification
Rationale	To provide immediate access to the user's main interface.
Business Rule	BR-9: Users must be redirected to their dashboard upon login.
Dependencies	FR-8
Priority	High

3.3.2 UC-02: Forget Password

3.3.2.1 FR-10: ForgetPassword.Input.Email

Table 49: ForgetPassword.Input.Email

Identifier	**FR-10**
Title	ForgetPassword.Input.Email
Requirement	The user shall be able to input their registered email address for password recovery.

Source	Zamin
Rationale	Users need to provide their email to receive a reset link.
Business Rule	BR-10: Email input is mandatory for the password reset process.
Dependencies	None
Priority	High

3.3.2.2 FR-11: ForgetPassword.Validate.Email

Table 50: ForgetPassword.Validate.Email

Identifier	**FR-11**
Title	ForgetPassword.Validate.Email
Requirement	The system shall validate the entered email against the user database.
Source	Functional Specification
Rationale	To ensure the email is associated with a registered account.
Business Rule	BR-11: Only registered emails can be validated for password reset.
Dependencies	FR-10
Priority	High

3.3.2.3 FR-12: ForgetPassword.GenerateResetLink

Table 51: ForgetPassword.GenerateResetLink

Identifier	**FR-12**
Title	ForgetPassword.GenerateResetLink
Requirement	The system shall generate a secure reset link for the validated email.
Source	Functional Specification
Rationale	To enable users to reset their password securely.
Business Rule	BR-12: A reset link must be generated for valid email requests.

Dependencies	FR-11
Priority	High

3.3.2.4 FR-13: ForgetPassword.SendResetLink

Table 52: ForgetPassword.SendResetLink

Identifier	**FR-13**
Title	ForgetPassword.SendResetLink
Requirement	The system shall send the generated reset link to the user's email address.
Source	Functional Specification
Rationale	To provide users with the means to reset their password.
Business Rule	BR-13: Email must be sent successfully for valid reset requests.
Dependencies	FR-12
Priority	High

3.3.2.5 FR-14: ForgetPassword.ResetSuccess

Table 53: ForgetPassword.ResetSuccess

Identifier	**FR-14**
Title	ForgetPassword.ResetSuccess
Requirement	The system shall display a confirmation message after successfully sending the reset link.
Source	Zamin
Rationale	To inform users that the reset link has been sent.
Business Rule	BR-14: Confirmation message must be displayed after sending the link.

Dependencies	FR-13
Priority	Medium

3.3.2.6 FR-15: ForgetPassword.ResetFailed

Table 54: ForgetPassword.ResetFailed

Identifier	**FR-15**
Title	ForgetPassword.ResetFailed
Requirement	The system shall display an error message if sending the reset link fails.
Source	Zamin
Rationale	To inform users about any issues encountered during the reset link process.
Business Rule	BR-15: An error message must be displayed for failed reset requests.
Dependencies	FR-13
Priority	Medium

3.3.3 UC-03 View Profile

3.3.3.1 FR-16: Profile.View.Click

Table 55: Profile.View.Click

Identifier	**FR-16**
Title	Profile.View.Click
Requirement	The user shall be able to click on a “View Profile” button to access their profile information.
Source	Zamin
Rationale	Users need an easy way to access their profile details.
Business Rule	BR-16: The “View Profile” button must be available in the user dashboard.

Dependencies	None
Priority	High

3.3.3.2 FR-17: Profile.View.Display

Table 56: Profile.View.Display

Identifier	**FR-17**
Title	Profile.View.Display
Requirement	The system shall display the user's profile information upon clicking the "View Profile" button.
Source	Functional Specification
Rationale	To provide users with access to their personal information and settings.
Business Rule	BR-17: Profile information must be displayed accurately and securely.
Dependencies	FR-16
Priority	High

3.3.4 UC-04 Edit Profile

3.3.4.1 FR-18: Profile.Edit.Info

Table 57: Profile.Edit.Info

Identifier	**FR-18**
Title	Profile.Edit.Info
Requirement	The user shall be able to access an editable form containing their profile information.
Source	Zamin

Rationale	Users need the ability to modify their profile details as necessary.
Business Rule	BR-18: The profile form must include fields for all user information.
Dependencies	FR-19
Priority	High

3.3.4.2 FR-19: Profile.Edit.Validate

Table 58: Profile.Edit.Validate

Identifier	**FR-19**
Title	Profile.Edit.Validate
Requirement	The system shall validate the input data for completeness and correctness before saving changes.
Source	Functional Specification
Rationale	To ensure that all user inputs meet the required format and standards.
Business Rule	BR-19: All fields must be validated according to specific criteria.
Dependencies	FR-18
Priority	High

3.3.4.3 FR-20: Profile.Edit.Save

Table 59: Profile.Edit.Save

Identifier	**FR-20**
Title	Profile.Edit.Save
Requirement	The user shall be able to save changes made to their profile information.
Source	Zamin
Rationale	Users need to confirm changes to their profile.

Business Rule	BR-20: Changes must be saved only if validation is successful.
Dependencies	FR-19
Priority	High

3.3.4.4 FR-21: Profile.Edit.Cancel

Table 60: Profile.Edit.Cancel

Identifier	**FR-21**
Title	Profile.Edit.Cancel
Requirement	The user shall be able to cancel editing their profile, reverting any unsaved changes.
Source	Zamin
Rationale	Users may change their minds and need a way to discard changes.
Business Rule	BR-21: Cancelling must restore the original profile data.
Dependencies	FR-18
Priority	Medium

3.3.4.5 FR-22 : Profile.Edit.UpdateSuccess

Table 61: Profile.Edit.UpdateSuccess

Identifier	**FR-22**
Title	Profile.Edit.UpdateSuccess
Requirement	The system shall provide feedback indicating successful profile updates.
Source	Zamin
Rationale	Users need confirmation that their changes have been saved successfully.

Business Rule	BR-22: A success message must be displayed after a successful save.
Dependencies	FR-20
Priority	High

3.3.4.6 FR-23: Profile.Edit.UpdateFailed

Table 62: Profile.Edit.UpdateFailed

Identifier	**FR-23**
Title	Profile.Edit.UpdateFailed
Requirement	The system shall provide feedback indicating failure to update the profile.
Source	Zamin
Rationale	Users need to be informed if their changes could not be saved.
Business Rule	BR-23: An error message must be displayed in case of a failure.
Dependencies	FR-20
Priority	High

3.3.5 UC-05 Password Change

3.3.5.1 FR-24: PasswordChange.Input.OldPassword

Table 63: PasswordChange.Input.OldPassword

Identifier	**FR-24**
Title	PasswordChange.Input.OldPassword
Requirement	The user shall be able to input their current password to initiate the password change process.
Source	Zamin

Rationale	Verification of the old password is necessary for security.
Business Rule	BR-24: The old password must be validated against the current user credentials.
Dependencies	FR-27
Priority	High

3.3.5.2 FR-25: PasswordChange.Input.NewPassword

Table 64: PasswordChange.Input.NewPassword

Identifier	**FR-25**
Title	PasswordChange.Input.NewPassword
Requirement	The user shall be able to input a new password for their account.
Source	Zamin
Rationale	Users need to set a new password to ensure account security.
Business Rule	BR-25: The new password must meet specific security criteria (e.g., length, complexity).
Dependencies	FR-24
Priority	High

3.3.5.3 FR-26: PasswordChange.Input.ConfirmPassword

Table 65: PasswordChange.Input.ConfirmPassword

Identifier	**FR-26**
Title	PasswordChange.Input.ConfirmPassword
Requirement	The user shall be able to confirm their new password by re-entering it.
Source	Zamin
Rationale	Confirming the new password helps prevent typographical errors.

Business Rule	BR-26: The confirmed password must match the new password input.
Dependencies	FR-25
Priority	High

3.3.5.4 FR-27: PasswordChange.Validate.OldPassword

Table 66: PasswordChange.Validate.OldPassword

Identifier	**FR-27**
Title	PasswordChange.Validate.OldPassword
Requirement	The system shall validate the entered old password against the stored password.
Source	Zamin
Rationale	To ensure that the user is authorized to change the password.
Business Rule	BR-27: If validation fails, notify the user of incorrect input.
Dependencies	FR-24
Priority	High

3.3.5.5 FR-28: PasswordChange.Validate.NewPassword

Table 67: PasswordChange.Validate.NewPassword

Identifier	**FR-28**
Title	PasswordChange.Validate.NewPassword
Requirement	The system shall validate the new password against security policies.
Source	Zamin
Rationale	To ensure the new password meets security requirements.
Business Rule	BR-28: Password must meet defined criteria (e.g., complexity, length).

Dependencies	FR-25
Priority	High

3.3.5.6 FR-29: PasswordChange.Validate.ConfirmPassword

Table 68: PasswordChange.Validate.ConfirmPassword

Identifier	**FR-29**
Title	PasswordChange.Validate.ConfirmPassword
Requirement	The system shall validate that the confirmed password matches the new password.
Source	Functional Specification
Rationale	To prevent users from mistyping their new password.
Business Rule	BR-29: The confirmed password must exactly match the new password input.
Dependencies	FR-26
Priority	High

3.3.5.7 FR-30: PasswordChange.Save

Table 69: PasswordChange.Save

Identifier	**FR-30**
Title	PasswordChange.Save
Requirement	The system shall save the new password after all validations are successful.
Source	Functional Specification
Rationale	To update the user's password securely in the database.
Business Rule	BR-30: The new password must overwrite the old password upon successful save.
Dependencies	FR-27, FR-28, FR-29

Priority	High

3.3.5.8 FR-31: PasswordChange.Confirm

Table 70: PasswordChange.Confirm

Identifier	**FR-31**
Title	PasswordChange.Confirm
Requirement	The system shall provide confirmation to the user that their password has been successfully changed.
Source	Zamin
Rationale	Users need assurance that their password change was successful.
Business Rule	BR-31: A success message must be displayed after the password is changed.
Dependencies	FR-30
Priority	High

3.3.5.9 FR-32: PasswordChange.Success

Table 71: PasswordChange.Success

Identifier	**FR-32**
Title	PasswordChange.Success
Requirement	The system shall log the password change event for security auditing.
Source	Zamin
Rationale	Keeping a record of password changes helps with security audits.
Business Rule	BR-32: All password change events must be logged with timestamps.
Dependencies	FR-30

Priority	Medium

3.3.5.10 FR-33: PasswordChange.Failed

Table 72: PasswordChange.Failed

Identifier	**FR-33**
Title	PasswordChange.Failed
Requirement	The system shall handle and display an error message when a password change attempt fails due to validation issues or other errors.
Source	Zamin
Rationale	Users need feedback to understand why their password change attempt was unsuccessful.
Business Rule	BR-33: The system must not disclose specific details about validation failures for security reasons.
Dependencies	FR-27, FR-28, FR-29
Priority	High

3.3.6 UC-06: Quick Access

3.3.6.1 FR-34: QuickAccess.DisplayResults

Table 73: QuickAccess.DisplayResults

Identifier	**FR-34**
Title	QuickAccess.DisplayResults
Requirement	The system shall display pinned reports and feedbacks in a quick access panel.
Source	Zamin
Rationale	Users need easy access to frequently referenced reports for efficiency.
Business Rule	BR-34: Only reports/feedback marked as pinned should appear in the quick access panel.

Dependencies	FR-35, FR-36
Priority	High

3.3.6.2 FR-35: QuickAccess.DisplayResults.Unpin

Table 74: QuickAccess.DisplayResults.Unpin

Identifier	**FR-35**
Title	QuickAccess.DisplayResults.Unpin
Requirement	The system shall allow users to unpin reports or feedbacks from the quick access panel.
Source	Zamin
Rationale	Users may need to customize their quick access list based on changing priorities.
Business Rule	BR-35: Only items currently pinned can be unpinned.
Dependencies	FR-34
Priority	Medium

3.3.6.3 FR-36: QuickAccess.Filter

Table 75: QuickAccess.Filter

Identifier	**FR-36**
Title	QuickAccess.Filter
Requirement	The system shall provide filtering options to narrow down displayed reports and feedbacks.
Source	Zamin
Rationale	Users need to quickly find specific reports/feedbacks among many.
Business Rule	BR-36: Filters should not affect pinned items' visibility.
Dependencies	FR-34

Priority	High

3.3.6.4 FR-37: QuickAccess.Filter.Display

Table 76: QuickAccess.Filter.Display

Identifier	**FR-37**
Title	QuickAccess.Filter.Display
Requirement	The system shall display filtered results based on user-selected criteria.
Source	Zamin
Rationale	Users require immediate feedback on their filtering choices for efficient navigation.
Business Rule	BR-37: Filters should be applied in real-time as users select criteria.
Dependencies	FR-36
Priority	High

3.3.6.5 FR-38: QuickAccess.Filter.Display.Unpin

Table 77: QuickAccess.Filter.Display.Unpin

Identifier	**FR-38**
Title	QuickAccess.Filter.Display.Unpin
Requirement	The system shall allow users to unpin items directly from the filtered results display.
Source	Zamin
Rationale	Enhancing user convenience by enabling item management from the current view.
Business Rule	BR-38: Unpinning from filtered results should update the main quick access panel accordingly.
Dependencies	FR-36, FR-37

Priority	Medium

3.3.7 UC-7 : View Notifications

3.3.7.1 FR-39: Notifications.View

Table 78: Notifications.View

Identifier	**FR-39**
Title	Notifications.View
Requirement	The system shall allow users to view a list of notifications.
Source	Zamin
Rationale	Users need to stay informed about system updates, alerts, and actions requiring attention.
Business Rule	BR-39: Notifications should be presented in chronological order.
Dependencies	None
Priority	High

3.3.7.2 FR-40: Notifications.Filter

Table 79: Notifications.Filter

Identifier	**FR-40**
Title	Notifications.Filter
Requirement	The system shall provide filtering options for notifications based on categories or urgency.
Source	Zamin

Rationale	Users need to quickly find relevant notifications among numerous entries.
Business Rule	BR-40: Filters should not affect the chronological order of notifications.
Dependencies	FR-39
Priority	High

3.3.7.3 FR-41: Notifications.ReadStatus

Table 80: Notifications.ReadStatus

Identifier	**FR-41**
Title	Notifications.ReadStatus
Requirement	The system shall indicate whether each notification has been read or unread.
Source	Zamin
Rationale	Users need to easily identify which notifications require attention.
Business Rule	BR-41: Unread notifications should be visually distinct from read ones.
Dependencies	FR-39
Priority	Medium

3.3.7.4 FR-42: Notifications.Clear

Table 81: Notifications.Clear

Identifier	**FR-42**
Title	Notifications.Clear
Requirement	The system shall allow users to clear notifications from their list.
Source	Zamin
Rationale	Users may want to remove outdated or irrelevant notifications for a cleaner interface.

Business Rule	BR-42: Clearing notifications should require user confirmation.
Dependencies	FR-39
Priority	Medium

3.3.7.5 FR-43: Notifications.MarkAsRead

Table 82: Notifications.MarkAsRead

Identifier	**FR-43**
Title	Notifications.MarkAsRead
Requirement	The system shall provide an option for users to mark notifications as read.
Source	Zamin
Rationale	Users need to manage their notification list effectively by indicating attention to certain notifications.
Business Rule	BR-43: Marking a notification as read should update its status immediately.
Dependencies	FR-39, FR-41
Priority	Medium

3.3.8 UC-08: Keyword Search

3.3.8.1 FR-44: Search.Input

Table 83: Search.Input

Identifier	**FR-44**
Title	Search.Input
Requirement	The system shall provide an input field for users to enter search keywords.
Source	Zamin

Rationale	Users need a way to specify what they are searching for to retrieve relevant results.
Business Rule	BR-44: The input field should validate for non-empty keywords before executing a search.
Dependencies	None
Priority	High

3.3.8.2 FR-45: Search.Execute

Table 84: Search.Execute

Identifier	**FR-45**
Title	Search.Execute
Requirement	The system shall execute the search based on the input keywords when the user initiates the search.
Source	Zamin
Rationale	Users expect immediate results based on their search criteria.
Business Rule	BR-45: The search should be executed only if the input passes validation.
Dependencies	FR-44
Priority	High

3.3.8.3 FR-46: Search.ResultsDisplay

Table 85: Search.ResultsDisplay

Identifier	**FR-46**
Title	Search.ResultsDisplay
Requirement	The system shall display search results in a user-friendly format after executing a search.
Source	Zamin

Rationale	Users need to see the results clearly to find the information they are looking for.
Business Rule	BR-46: Results should be displayed in a paginated format if there are too many results.
Dependencies	FR-45
Priority	High

3.3.8.4 FR-47: Search.FilterResults

Table 86: Search.FilterResults

Identifier	**FR-47**
Title	Search.FilterResults
Requirement	The system shall provide options to filter search results based on various criteria.
Source	Zamin
Rationale	Users often want to narrow down results to find specific information efficiently.
Business Rule	BR-47: Filtering should be applicable after results are displayed.
Dependencies	FR-46
Priority	Medium

3.3.9 UC-09: Logout

3.3.9.1 FR-48: Logout.Request

Table 87: Logout.Request

Identifier	**FR-48**
Title	Logout.Request

Requirement	The system shall allow the user to initiate a logout request.
Source	Zamin
Rationale	Users need a clear option to log out of the application for security purposes.
Business Rule	BR-48: Logout options must be easily accessible in the user interface.
Dependencies	None
Priority	High

3.3.9.2 FR-49: Logout.Confirm

Table 88: Logout.Confirm

Identifier	**FR-49**
Title	Logout.Confirm
Requirement	The system shall prompt the user to confirm their logout action.
Source	Zamin
Rationale	A confirmation step helps prevent accidental logouts.
Business Rule	BR-49: Confirmation dialog must include options to proceed or cancel.
Dependencies	FR-48

3.3.9.3 FR-50: Logout.Redirect

Table 89: Logout.Redirect

Identifier	**FR-50**
Title	Logout.Redirect
Requirement	The system shall redirect the user to the login page after successful logout.
Source	Zamin

Rationale	Redirecting ensures the user is guided back to the login interface after logging out.
Business Rule	BR-50: The redirect should occur only after a successful logout process.
Dependencies	FR-49
Priority	High

3.3.9.4 FR-51: Logout.SessionDeletion

Table 90: Logout.SessionDeletion

Identifier	**FR-51**
Title	Logout.SessionDeletion
Requirement	The system shall delete the user session upon logout.
Source	Zamin
Rationale	Session deletion is essential for security and to protect user data.
Business Rule	BR-51: User sessions must be completely cleared upon logout.
Dependencies	FR-49
Priority	High

3.3.9.5 FR-52: Logout.CredentialsRemoval

Table 91: Logout.CredentialsRemoval

Identifier	**FR-52**
Title	Logout.CredentialsRemoval
Requirement	The system shall remove user credentials from local storage after logout.
Source	Zamin

Rationale	Clearing credentials enhances security by preventing unauthorized access.
Business Rule	BR-52: User credentials must not persist after logout.
Dependencies	FR-51
Priority	High

3.3.10 UC-10 : Register Supervisor

3.3.10.1 FR-53: Admin.RegisterSupervisor.Input

Table 92: Admin.RegisterSupervisor.Input

Identifier	**FR-53**
Title	Admin.RegisterSupervisor.Input
Requirement	The admin shall be able to input the necessary details to register a supervisor.
Source	Shahryar
Rationale	Admins need to provide relevant supervisor details for registration.
Business Rule	BR-53: All required fields must be filled before submission.
Dependencies	None
Priority	High

3.3.10.2 FR-54: Admin.RegisterSupervisor.Validate

Table 93: Admin.RegisterSupervisor.Validate

Identifier	**FR-54**
Title	Admin.RegisterSupervisor.Validate
Requirement	The system shall validate the input provided by the admin for supervisor registration.

Source	Shahryar
Rationale	Validation ensures that only correct and complete data is submitted.
Business Rule	BR-54: Inputs must meet defined criteria for successful registration.
Dependencies	FR-53
Priority	High

3.3.10.3 FR-55: Admin.RegisterSupervisor.Validate.Success

Table 94: Admin.RegisterSupervisor.Validate.Success

Identifier	**FR-55**
Title	Admin.RegisterSupervisor.Validate.Success
Requirement	The system shall indicate successful validation of supervisor registration input.
Source	Shahryar
Rationale	Admins need confirmation that inputs are valid before proceeding.
Business Rule	BR-55: A success message must be displayed upon valid input.
Dependencies	FR-54
Priority	High

3.3.10.4 FR-56: Admin.RegisterSupervisor.Invalid

Table 95: Admin.RegisterSupervisor.Invalid

Identifier	**FR-56**
Title	Admin.RegisterSupervisor.Invalid
Requirement	The system shall indicate if the input provided by the admin is invalid.

Source	Shahryar
Rationale	Identifying invalid input helps admins correct errors before submission.
Business Rule	BR-56: An error message must specify invalid fields.
Dependencies	FR-54

3.3.10.5 FR-57: Admin.RegisterSupervisor.Validate.Failure

Table 96: Admin.RegisterSupervisor.Validate.Failure

Identifier	**FR-57**
Title	Admin.RegisterSupervisor.Validate.Failure
Requirement	The system shall handle and report validation failures during supervisor registration.
Source	Shahryar
Rationale	Proper error handling ensures users are aware of validation issues.
Business Rule	BR-57: All validation errors must be displayed to the admin.
Dependencies	FR-54
Priority	High

3.3.10.6 FR-58: Admin.RegisterSupervisor.RedirectToInput

Table 97: Admin.RegisterSupervisor.RedirectToInput

Identifier	**FR-58**
Title	Admin.RegisterSupervisor.RedirectToInput
Requirement	The system shall redirect the admin to the input form if validation fails.
Source	Shahryar

Rationale	Redirecting ensures that admins can correct any errors in input.
Business Rule	BR-58: Redirection must occur immediately after validation failure.
Dependencies	FR-56, FR-57
Priority	High

3.3.10.7 FR-59: Admin.RegisterSupervisor.Save

Table 98: Admin.RegisterSupervisor.Save

Identifier	**FR-59**
Title	Admin.RegisterSupervisor.Save
Requirement	The system shall save the supervisor registration details in the database.
Source	Shahryar
Rationale	Saving details is essential for creating a new supervisor account.
Business Rule	BR-59: Registration must be confirmed before saving to the database.
Dependencies	FR-55
Priority	High

3.3.10.8 FR-60: Admin.RegisterSupervisor.Confirm

Table 99: Admin.RegisterSupervisor.Confirm

Identifier	**FR-60**
Title	Admin.RegisterSupervisor.Confirm
Requirement	The system shall confirm successful registration of the supervisor.
Source	Shahryar
Rationale	Confirmation informs the admin that the supervisor has been registered successfully.
Business Rule	BR-60: A success message must be shown after successful registration.

Dependencies	FR-59
Priority	High

3.3.11 UC-11 View Supervisor Detail

3.3.11.1 FR-61: Admin.ViewSupervisor.InputDetails

Table 100: Admin.ViewSupervisor.InputDetails

Identifier	**FR-61**
Title	Admin.ViewSupervisor.InputDetails
Requirement	The admin shall be able to input the necessary details to view a specific supervisor's information.
Source	Shahryar
Rationale	Admins need to specify which supervisor's details they wish to view.
Business Rule	BR-61: The admin must select a supervisor from a list or input an identifier.
Dependencies	None
Priority	High

3.3.11.2 FR-62: Admin.ViewSupervisor.DisplayDetails

Table 101: Admin.ViewSupervisor.DisplayDetails

Identifier	**FR-62**
Title	Admin.ViewSupervisor.DisplayDetails
Requirement	The system shall display the details of the selected supervisor after input is provided.
Source	Shahryar

Rationale	Displaying details allows admins to review supervisor information for management purposes.
Business Rule	BR-62: All relevant supervisor information must be shown clearly.
Dependencies	FR-61
Priority	High

3.3.12 UC-12 Suspend Supervisor Account

3.3.12.1 FR-63: Admin.SuspendSupervisor.Input

Table 102: Admin.SuspendSupervisor.Input

Identifier	**FR-63**
Title	Admin.SuspendSupervisor.Input
Requirement	The admin shall be able to input the necessary details to suspend a supervisor's account.
Source	Shahryar
Rationale	Admins need to specify which supervisor's account is to be suspended.
Business Rule	BR-63: The admin must select a supervisor from a list or input an identifier.
Dependencies	None
Priority	High

3.3.12.2 FR-64: Admin.SuspendSupervisor.Confirm

Table 103: Admin.SuspendSupervisor.Confirm

Identifier	**FR-64**
Title	Admin.SuspendSupervisor.Confirm
Requirement	The system shall prompt the admin to confirm the suspension of the selected supervisor's account.

Source	Shahryar
Rationale	A confirmation step prevents accidental suspensions of supervisor accounts.
Business Rule	BR-64: Confirmation must require explicit admin action (e.g., clicking "Yes").
Dependencies	FR-63
Priority	High

3.3.12.3 FR-65: Admin.SuspendSupervisor.Notify

Table 104: Admin.SuspendSupervisor.Notify

Identifier	**FR-65**
Title	Admin.SuspendSupervisor.Notify
Requirement	The system shall notify the supervisor about the suspension of their account.
Source	Shahryar
Rationale	Notifying the supervisor is essential for transparency and communication.
Business Rule	BR-65: Notification must be sent via email or in-app message.
Dependencies	FR-64
Priority	Medium

3.3.12.4 FR-66: Admin.SuspendSupervisor.UpdateStatus

Table 105: Admin.SuspendSupervisor.UpdateStatus

Identifier	**FR-66**
Title	Admin.SuspendSupervisor.UpdateStatus
Requirement	The system shall update the status of the supervisor's account to "suspended" in the database.

Source	Shahryar
Rationale	Accurate status tracking is necessary for effective account management.
Business Rule	BR-66: The system must reflect the updated status immediately.
Dependencies	FR-64
Priority	High

3.3.13 UC-13 Delete Supervisor Account

3.3.13.1 FR-67: Admin.DeleteSupervisor.Input

Table 106: Admin.DeleteSupervisor.Input

Identifier	**FR-67**
Title	Admin.DeleteSupervisor.Input
Requirement	The admin shall be able to input the necessary details to delete a supervisor's account.
Source	Shahryar
Rationale	Admins need to specify which supervisor's account is to be deleted.
Business Rule	BR-67: The admin must select a supervisor from a list or input an identifier.
Dependencies	None
Priority	High

3.3.13.2 FR-68: Admin.DeleteSupervisor.Confirm

Table 107: Admin.DeleteSupervisor.Confirm

Identifier	**FR-68**
Title	Admin.DeleteSupervisor.Confirm
Requirement	The system shall prompt the admin to confirm the deletion of the selected supervisor's account.
Source	Shahryar

Rationale	A confirmation step prevents accidental deletions of supervisor accounts.
Business Rule	BR-68: Confirmation must require explicit admin action (e.g., clicking “Yes”).
Dependencies	FR-67
Priority	High

3.3.13.3 FR-69: Admin.DeleteSupervisor.Success

Table 108: Admin.DeleteSupervisor.Success

Identifier	**FR-69**
Title	Admin.DeleteSupervisor.Success
Requirement	The system shall display a success message after the supervisor’s account has been deleted.
Source	Zamin
Rationale	Providing feedback to the admin ensures they are aware of the successful action.
Business Rule	BR-69: The success message must be clear and visible.
Dependencies	FR-68
Priority	Medium

3.3.13.4 FR-70: Admin.DeleteSupervisor.Failure

Table 109: Admin.DeleteSupervisor.Failure

Identifier	**FR-70**
Title	Admin.DeleteSupervisor.Failure
Requirement	The system shall display a failure message if the supervisor’s account deletion fails.
Source	Zamin
Rationale	Clear communication about failures allows the admin to address issues promptly.

Business Rule	BR-70: The failure message must include reasons for failure, if applicable.
Dependencies	FR-68
Priority	High

3.3.13.5 FR-71: Admin.DeleteSupervisor.RedirectToInput

Table 110: Admin.DeleteSupervisor.RedirectToInput

Identifier	**FR-71**
Title	Admin.DeleteSupervisor.RedirectToInput
Requirement	After the deletion process, the system shall redirect the admin back to the input screen for further actions.
Source	Shahryar
Rationale	Redirection facilitates efficient workflow for admins needing to perform multiple actions.
Business Rule	BR-71: Redirection must happen after successful or failed deletion.
Dependencies	FR-69, FR-70
Priority	Medium

3.3.13.6 FR-72: Admin.DeleteSupervisor.Notify

Table 111: Admin.DeleteSupervisor.Notify

Identifier	**FR-72**
Title	Admin.DeleteSupervisor.Notify
Requirement	The system shall notify the supervisor about the deletion of their account.
Source	Shahryar
Rationale	Notifying the supervisor is essential for transparency and communication.

Business Rule	BR-72: Notification must be sent via email or in-app message.
Dependencies	FR-68
Priority	Medium

3.3.13.7 FR-73: Admin.DeleteSupervisor.RemoveFromList

Table 112: Admin.DeleteSupervisor.RemoveFromList

Identifier	**FR-73**
Title	Admin.DeleteSupervisor.RemoveFromList
Requirement	The system shall remove the supervisor's account from the list of supervisors after deletion.
Source	Shahryar
Rationale	Accurate list management is crucial for maintaining up-to-date records.
Business Rule	BR-73: The supervisor must no longer appear in the list once deleted.
Dependencies	FR-69
Priority	High

3.3.14 UC-14 Audit Logging

3.3.14.1 Admin.AuditLogging.View

Table 113: Admin.AuditLogging.View

Identifier	**FR-74**
Title	Admin.AuditLogging.View
Requirement	The admin shall be able to view the audit logs of system activities.
Source	Shahryar
Rationale	Viewing logs helps admins monitor system usage and detect anomalies.

Business Rule	BR-74: Only authorized admins can access audit logs.
Dependencies	None
Priority	High

3.3.14.2 FR-75: Admin.AuditLogging.Search

Table 114: Admin.AuditLogging.Search

Identifier	**FR-75**
Title	Admin.AuditLogging.Search
Requirement	The admin shall be able to search the audit logs based on specified criteria.
Source	Shahryar
Rationale	Searching enables admins to quickly find relevant logs.
Business Rule	BR-75: Search must support multiple criteria (e.g., date, user, action).
Dependencies	FR-74
Priority	Medium

3.3.14.3 FR-76: Admin.AuditLogging.Filter

Table 115: Admin.AuditLogging.Filter

Identifier	**FR-76**
Title	Admin.AuditLogging.Filter
Requirement	The admin shall be able to filter the audit logs to display specific entries.
Source	Shahryar
Rationale	Filtering helps focus on specific log entries relevant to the admin's needs.
Business Rule	BR-76: Filter options must be user-friendly and comprehensive.
Dependencies	FR-74

Priority	Medium

3.3.14.4 FR-77: Admin.AuditLogging.Export

Table 116: Admin.AuditLogging.Export

Identifier	**FR-77**
Title	Admin.AuditLogging.Export
Requirement	The admin shall be able to export the audit logs in various formats (e.g., CSV, PDF).
Source	Export Functionality
Rationale	Exporting logs allows for offline analysis and reporting.
Business Rule	BR-77: The export feature must maintain the integrity of log data.
Dependencies	FR-74
Priority	Medium

3.3.15 UC-15: View Cameras Details

3.3.15.1 FR-78: Admin.CamerasDetails.Input

Table 117: Admin.CamerasDetails.Input

Identifier	**FR-78**
Title	Admin.CamerasDetails.Input
Requirement	The admin shall be able to input parameters to query camera details.
Source	Shahryar
Rationale	Input parameters allow for targeted queries for specific camera details.
Business Rule	BR-78: Admins must provide valid input to retrieve camera details.
Dependencies	None

Priority	High

3.3.15.2 FR-79: Admin.CamerasDetails.View

Table 118: Admin.CamerasDetails.View

Identifier	**FR-79**
Title	Admin.CamerasDetails.View
Requirement	The admin shall be able to view the details of all registered cameras.
Source	Shahryar
Rationale	Viewing camera details is essential for monitoring and managing camera operations.
Business Rule	BR-79: Only authorized admins can view camera details.
Dependencies	FR-78
Priority	High

3.3.15.3 FR-79 Admin.CamerasDetails.Filter

Table 119: Admin.CamerasDetails.Filter

Identifier	**FR-80**
Title	Admin.CamerasDetails.Filter
Requirement	The admin shall be able to filter camera details based on specific criteria (e.g., location, status).
Source	Shahryar
Rationale	Filtering enables admins to focus on specific cameras relevant to their needs.
Business Rule	BR-80: Filter options must be intuitive and provide relevant categories.
Dependencies	FR-79
Priority	Medium

3.3.16 UC-16: View Camera Status

3.3.16.1 FR-81: Admin.CameraStatus.View

Table 120: Admin.CameraStatus.View

Identifier	**FR-81**
Title	Admin.CameraStatus.View
Requirement	The admin shall be able to view the status of all cameras in the system.
Source	Shahryar
Rationale	Monitoring camera status is crucial for ensuring operational integrity and security.
Business Rule	BR-81: Only authorized admins can view camera statuses.
Dependencies	None
Priority	High

3.3.16.2 FR-82: Admin.CameraStatus.DisplayList

Table 121: Admin.CameraStatus.DisplayList

Identifier	**FR-82**
Title	Admin.CameraStatus.DisplayList
Requirement	The system shall display a list of all cameras along with their current status (e.g., online, offline).
Source	Shahryar
Rationale	A clear display of camera statuses aids in quick decision-making and troubleshooting.
Business Rule	BR-82: The displayed list must be updated in real-time.
Dependencies	FR-81
Priority	High

3.3.17 UC-17: Customize Interface

3.3.17.1 FR-83: Admin.CustomizeInterface.Input

Table 122: Admin.CustomizeInterface.Input

Identifier	**FR-83**
Title	Admin.CustomizeInterface.Input
Requirement	The admin shall be able to input customization options for the interface, such as themes and layout preferences.
Source	Zamin
Rationale	Customizing the interface improves user experience and accessibility for different user roles.
Business Rule	BR-83: Only admins can customize the interface.
Dependencies	None
Priority	Medium

3.3.17.2 FR-84: Admin.CustomizeInterface.Save

Table 123: Admin.CustomizeInterface.Save

Identifier	**FR-84**
Title	Admin.CustomizeInterface.Save
Requirement	The system shall save the admin's customization choices for future sessions.
Source	Shahryar
Rationale	Saving customization settings ensures a consistent user experience.
Business Rule	BR-84: Customizations must be saved securely to prevent unauthorized changes.
Dependencies	FR-83
Priority	High

3.3.18 UC-18: View Feedback

3.3.18.1 FR-85: Admin.SupervisorFeedback.View

Table 124: Admin.SupervisorFeedback.View

Identifier	**FR-85**
Title	Admin.SupervisorFeedback.View
Requirement	The admin shall be able to view feedback submitted by supervisors.
Source	Shahryar
Rationale	Viewing feedback helps admins address concerns and improve processes.
Business Rule	BR-85: Only authorized admins can view supervisor feedback.
Dependencies	None
Priority	High

3.3.18.2 FR-86: Admin.SupervisorFeedback.MarkAsRead

Table 125: Admin.SupervisorFeedback.MarkAsRead

Identifier	**FR-86**
Title	Admin.SupervisorFeedback.MarkAsRead
Requirement	The admin shall be able to mark supervisor feedback as read.
Source	Shahryar
Rationale	Marking feedback as read helps track which feedback has been acknowledged.
Business Rule	BR-86: Feedback must be marked as read to remove it from the unread notifications.
Dependencies	FR-85
Priority	Medium

3.3.18.3 FR-87: Admin.SupervisorFeedback.Pin

Table 126: Admin.SupervisorFeedback.Pin

Identifier	**FR-87**
Title	Admin.SupervisorFeedback.Pin
Requirement	The admin shall be able to pin important supervisor feedback for easy access.
Source	Shahryar
Rationale	Pinning feedback allows admins to prioritize important messages.
Business Rule	BR-87: Only admins can pin feedback.
Dependencies	FR-85
Priority	Medium

3.3.19 UC-19 : Monitor Camera Surveillance

3.3.19.1 FR-88 : Supervisor.MonitorCameraSurveillance.View

Table 127: Supervisor.MonitorCameraSurveillance.View

Identifier	**FR-88**
Title	Supervisor.MonitorCameraSurveillance.View
Requirement	The supervisor shall be able to view live camera surveillance footage.
Source	Shahryar
Rationale	Supervisors need to monitor live footage to ensure site safety and incident response.
Business Rule	BR-88: Only authorized supervisors can access the live camera feeds.
Dependencies	None
Priority	High

3.3.19.2 FR-89: Supervisor.MonitorCameraSurveillance.Display

Table 128: Supervisor.MonitorCameraSurveillance.Display

Identifier	**FR-89**
Title	Supervisor.MonitorCameraSurveillance.Display
Requirement	The system shall display live camera feeds with options for zoom, pan, and tilt.
Source	Shahryar
Rationale	Supervisors need to adjust views for detailed monitoring.
Business Rule	BR-89: Camera controls are restricted based on camera permissions.
Dependencies	FR-88
Priority	Medium

3.3.20 UC-20: Switch between Cameras

3.3.20.1 FR-90: Supervisor.SwitchCameras.Input

Table 129: Supervisor.SwitchCameras.Input

Identifier	**FR-90**
Title	Supervisor.SwitchCameras.Input
Requirement	The supervisor shall be able to select and switch between different camera feeds.
Source	Shahryar
Rationale	Supervisors need to switch between different camera feeds to monitor various areas efficiently.
Business Rule	BR-90: Only authorized supervisors can switch camera feeds.
Dependencies	None
Priority	High

3.3.20.2 FR-91: Supervisor.SwitchCamera.Display

Table 130: Supervisor.SwitchCamera.Display

Identifier	**FR-91**
Title	Supervisor.SwitchCamera.Display
Requirement	The system shall display the selected camera feed promptly after switching.
Source	Shahryar
Rationale	The selected camera feed must display instantly for effective monitoring.
Business Rule	BR-91: Switching between cameras must not delay the feed display.
Dependencies	FR-90
Priority	High

3.3.20.3 FR-92: Supervisor.SwitchCameras.Notify

Table 131: Supervisor.SwitchCameras.Notify

Identifier	**FR-92**
Title	Supervisor.SwitchCameras.Notify
Requirement	The system shall notify the supervisor when the camera feed has successfully switched.
Source	Shahryar
Rationale	The supervisor should be informed when the camera feed is changed to avoid confusion.
Business Rule	BR-92: Notifications must be displayed in real-time to the supervisor.
Dependencies	FR-90, FR-91
Priority	Medium

3.3.21 UC-21:Guide (Manual) for PPE Usage

3.3.21.1 FR-93 : Supervisor.GuidePPEUsage.View

Table 132: Supervisor.GuidePPEUsage.View

Identifier	**FR-93**
Title	Supervisor.GuidePPEUsage.View
Requirement	The supervisor shall be able to view the guide or manual for proper PPE usage.
Source	Shahryar
Rationale	Supervisors need access to PPE usage guidelines to ensure compliance with safety standards.
Business Rule	BR-93: Only authorized personnel can view the PPE guide.
Dependencies	None
Priority	High

3.3.21.2 FR-94: Supervisor.GuidePPEUsage.Download

Table 133: Supervisor.GuidePPEUsage.Download

Identifier	**FR-94**
Title	Supervisor.GuidePPEUsage.Download
Requirement	The supervisor shall be able to download the PPE usage guide for offline reference.
Source	Shahryar
Rationale	Downloading the guide allows supervisors to have access to it when offline or in remote areas.
Business Rule	BR-94: The downloaded guide must be in a standard, accessible format (e.g., PDF).
Dependencies	FR-93

Priority	Medium

3.3.22 UC-22: Customize Anomaly Parameters

3.3.22.1 FR-95: Supervisor.CustomizeAnomalyParameters.Input

Table 134: Supervisor.CustomizeAnomalyParameters.Input

Identifier	**FR-95**
Title	Supervisor.CustomizeAnomalyParameters.Input
Requirement	The supervisor shall be able to input custom parameters for anomaly detection.
Source	Zamin
Rationale	Customizing anomaly parameters allows supervisors to tailor detection to the specific needs of the construction site.
Business Rule	BR-95: Anomaly parameters must adhere to the system's detection range limits.
Dependencies	None
Priority	High

3.3.22.2 FR-96: Supervisor.CustomizeAnomalyParameters.Save

Table 135: Supervisor.CustomizeAnomalyParameters.Save

Identifier	**FR-96**
Title	Supervisor.CustomizeAnomalyParameters.Save
Requirement	The system shall save the custom anomaly parameters input by the supervisor.
Source	Zamin
Rationale	Saving the parameters ensures they are retained for future anomaly detection.
Business Rule	BR-96: The system must validate input before saving the parameters.
Dependencies	FR-95

Priority	High

3.3.22.3 FR-97: Supervisor.CustomizeAnomalyParameters.Confirm

Table 136: Supervisor.CustomizeAnomalyParameters.Confirm

Identifier	**FR-97**
Title	Supervisor.CustomizeAnomalyParameters.Confirm
Requirement	The system shall confirm successful customization of anomaly parameters.
Source	Zamin
Rationale	Providing confirmation assures the supervisor that their changes have been applied.
Business Rule	BR-97: Confirmation should appear immediately after saving parameters.
Dependencies	FR-96
Priority	Medium

3.3.22.4 FR-98: Supervisor.CustomizeAnomalyParameters.Notify

Table 137: Supervisor.CustomizeAnomalyParameters.Notify

Identifier	**FR-98**
Title	Supervisor.CustomizeAnomalyParameters.Notify
Requirement	The system shall notify relevant personnel if anomaly detection parameters are changed.
Source	Zamin
Rationale	Notifications ensure transparency and accountability for any changes in detection settings.
Business Rule	BR-98: Notifications should be sent to all stakeholders via email or in-app alerts.
Dependencies	FR-97
Priority	Medium

3.3.23 UC-23: View Weather Forecast

3.3.23.1 FR-99: Supervisor.ViewWeatherForecast.View

Table 138: Supervisor.ViewWeatherForecast.View

Identifier	**FR-99**
Title	Supervisor.ViewWeatherForecast.View
Requirement	The supervisor shall be able to view the weather forecast for the construction site area.
Source	Zamin
Rationale	Access to weather information helps the supervisor plan for weather-related disruptions and safety precautions.
Business Rule	BR-99: Weather data must be fetched from a reliable external source and updated in real-time.
Dependencies	None
Priority	Medium

3.3.24 UC-24: View Alerts

3.3.24.1 FR-100: Supervisor.ViewAlerts.View

Table 139: Supervisor.ViewAlerts.View

Identifier	**FR-100**
Title	Supervisor.ViewAlerts.View
Requirement	The supervisor shall be able to view all system-generated alerts related to site surveillance.
Source	Zamin
Rationale	Allows the supervisor to stay informed about critical events and take necessary actions.
Business Rule	BR-100: Alerts should be displayed in real-time and should include relevant details like date, time, and alert type.

Dependencies	FR-101
Priority	High

3.3.24.2 FR-101: Supervisor.ViewAlerts.Sort

Table 140: Supervisor.ViewAlerts.Sort

Identifier	**FR-101**
Title	Supervisor.ViewAlerts.Sort
Requirement	The supervisor shall be able to sort alerts based on parameters like date, type, and severity.
Source	Zamin
Rationale	Sorting helps the supervisor prioritize and manage alerts efficiently.
Business Rule	BR-101: Alerts must be sortable by default and filterable by user input.
Dependencies	FR-100
Priority	Medium

3.3.25 UC-25: Acknowledge Alerts

3.3.25.1 FR-102: Supervisor.AcknowledgeAlerts.Approve

Table 141: Supervisor.AcknowledgeAlerts.Approve

Identifier	**FR-102**
Title	Supervisor.AcknowledgeAlerts.Approve
Requirement	The supervisor shall be able to approve alerts as acknowledged.
Source	Zamin
Rationale	Allows the supervisor to confirm alerts have been reviewed and addressed.
Business Rule	BR-102: Only supervisors with specific roles can approve alerts.

Dependencies	FR-104
Priority	High

3.3.25.2 FR-103: Supervisor.AcknowledgeAlerts.Disapprove

Table 142: Supervisor.AcknowledgeAlerts.Disapprove

Identifier	**FR-103**
Title	Supervisor.AcknowledgeAlerts.Disapprove
Requirement	The supervisor shall be able to disapprove alerts if necessary, marking them as unresolved.
Source	Zamin
Rationale	Enables the supervisor to take appropriate actions if an alert is considered unresolved or incorrect.
Business Rule	BR-103: Disapproved alerts should trigger further investigation or escalation.
Dependencies	FR-104
Priority	Medium

3.3.25.3 FR-104: Supervisor.AcknowledgeAlerts.StatusUpdated

Table 143: Supervisor.AcknowledgeAlerts.StatusUpdated

Identifier	**FR-104**
Title	Supervisor.AcknowledgeAlerts.StatusUpdated
Requirement	The system shall update the status of alerts after they are either approved or disapproved by the supervisor.
Source	Zamin
Rationale	Keeps track of the alert handling process to maintain an accurate log of actions taken.
Business Rule	BR-104: The system must log the time and action taken by the supervisor for each alert.

Dependencies	FR-102, FR-103
Priority	High

3.3.26 UC-26: Download Footages

3.3.26.1 FR-105: Supervisor.DownloadFootages.Select

Table 144: Supervisor.DownloadFootages.Select

Identifier	**FR-105**
Title	Supervisor.DownloadFootages.Select
Requirement	The supervisor shall be able to select specific footage for downloading.
Source	Zamin
Rationale	Allows the supervisor to choose the relevant camera footage for download.
Business Rule	BR-105: Only authorized supervisors can select and download footage.
Dependencies	None
Priority	High

3.3.26.2 FR-106: Supervisor.DownloadFootages.Save

Table 145: Supervisor.DownloadFootages.Save

Identifier	**FR-106**
Title	Supervisor.DownloadFootages.Save
Requirement	The system shall allow the supervisor to save the selected footage to their device.
Source	Zamin
Rationale	Ensures the footage can be securely downloaded for future use.
Business Rule	BR-106: Footage must be stored in an encrypted format if required by security policies.

Dependencies	FR-105
Priority	High

3.3.26.3 FR-107: Supervisor.DownloadFootages.Notify

Table 146: Supervisor.DownloadFootages.Notify

Identifier	**FR-107**
Title	Supervisor.DownloadFootages.Notify
Requirement	The system shall notify the supervisor when the footage has been successfully downloaded.
Source	Zamin
Rationale	Confirms the completion of the download process for the supervisor.
Business Rule	BR-107: Notifications must include the footage details, including date, time, and camera ID.
Dependencies	FR-106
Priority	Medium

3.3.27 UC-27: View Automated Generated Reports

3.3.27.1 FR-108: Supervisor.AutomatedGeneratedReports.View

Table 147: Supervisor.AutomatedGeneratedReports.View

Identifier	**FR-108**
Title	Supervisor.AutomatedGeneratedReports.View
Requirement	The supervisor shall be able to view all automated generated reports.
Source	Shahryar
Rationale	Supervisors need to review system-generated reports for monitoring.

Business Rule	BR-25: Only authorized supervisors can access reports.
Dependencies	None
Priority	High

3.3.27.2 FR-109: Supervisor.AutomatedGeneratedReports.Sort

Table 148: Supervisor.AutomatedGeneratedReports.Sort

Identifier	**FR-109**
Title	Supervisor.AutomatedGeneratedReports.Sort
Requirement	The supervisor shall be able to sort automated reports by time, type, or severity.
Source	Shahryar
Rationale	Sorting helps to prioritize critical reports for immediate action.
Business Rule	BR-26: Sorting should follow company-defined report categories.
Dependencies	FR-108
Priority	Medium

3.3.28 UC-28: Handle Reports

3.3.28.1 FR-110: Supervisor. HandleReports.Approve

Table 149: Supervisor.HandleReports.Approve

Identifier	**FR-110**
Title	Supervisor.HandleReports.Approve
Requirement	The supervisor shall be able to approve reports for further action.
Source	Shahryar

Rationale	Approved reports need to be processed for follow-up tasks.
Business Rule	BR-27: Only authorized supervisors can approve reports.
Dependencies	FR-108
Priority	High

3.3.28.2 FR-111: Supervisor.HandleReports.Save

Table 150: Supervisor.HandleReports.Save

Identifier	**FR-111**
Title	Supervisor.HandleReports.Save
Requirement	The supervisor shall be able to save reports for later review or follow-up.
Source	Shahryar
Rationale	Saving reports allows supervisors to revisit them at a convenient time.
Business Rule	BR-28: Saved reports must be securely stored.
Dependencies	FR-110
Priority	Medium

3.3.28.3 FR: Supervisor.HandleReports.Pin

Table 151: Supervisor.HandleReports.Pin

Identifier	**FR-112**
Title	Supervisor.HandleReports.Pin
Requirement	The supervisor shall be able to pin important reports for quick access.
Source	Shahryar
Rationale	Pinning enables easy retrieval of critical reports.
Business Rule	BR-29: Pinned reports must remain accessible across sessions.

Dependencies	FR-108
Priority	Medium

3.3.28.4 FR-113: Supervisor.HandleReports.Disapprove

Table 152: Supervisor.HandleReports.Disapprove

Identifier	**FR-113**
Title	Supervisor.HandleReports.Disapprove
Requirement	The supervisor shall be able to disapprove reports that do not require further action.
Source	Shahryar
Rationale	Disapproving unnecessary reports reduces workload.
Business Rule	BR-30: Disapproved reports must be logged with a rationale.
Dependencies	FR-108
Priority	Medium

3.3.29 UC-29: Regenerate Reports

3.3.29.1 FR-114: Supervisor.RegenerateReports.Click

Table 153: Supervisor.RegenerateReports.Click

Identifier	**FR-114**
Title	Supervisor.RegenerateReports.Click
Requirement	The supervisor shall be able to click a button to regenerate reports.
Source	Shahryar
Rationale	Regenerating reports allows for the latest data and insights.
Business Rule	BR-31: Only authorized supervisors can regenerate reports.

Dependencies	None
Priority	High

3.3.29.2 FR-115: Supervisor.RegenerateReports.NewReport

Table 154: Supervisor.RegenerateReports.NewReport

Identifier	**FR-115**
Title	Supervisor.RegenerateReports.NewReport
Requirement	The system shall generate a new report based on the latest data when requested.
Source	Shahryar
Rationale	New reports provide up-to-date information for decision-making.
Business Rule	BR-32: New reports must be generated within a specified time frame.
Dependencies	FR-114
Priority	High

3.3.30 UC-30: Download Reports

3.3.30.1 FR-116: Supervisor.DownloadReports.Input

Table 155: Supervisor.DownloadReports.Input

Identifier	**FR-116**
Title	Supervisor.DownloadReports.Input
Requirement	The supervisor shall be able to input criteria for downloading reports.

Source	Shahryar
Rationale	Inputting criteria allows for tailored report downloads.
Business Rule	BR-33: Input fields must validate the supervisor's selections.
Dependencies	None
Priority	Medium

3.3.30.2 FR-117: Supervisor.DownloadReports.Save

Table 156: Supervisor.DownloadReports.Save

Identifier	**FR-117**
Title	Supervisor.DownloadReports.Save
Requirement	The supervisor shall be able to save downloaded reports to a specified location.
Source	Shahryar
Rationale	Saving reports ensures they can be accessed offline or for future reference.
Business Rule	BR-34: Saved reports must comply with data retention policies.
Dependencies	FR-116
Priority	High

3.3.30.3 FR-118: Supervisor.DownloadReports.Notify

Table 157: Supervisor.DownloadReports.Notify

Identifier	**FR-118**

Title	Supervisor.DownloadReports.Notify
Requirement	The system shall notify the supervisor upon successful download of reports.
Source	Shahryar
Rationale	Notifications confirm that the report download process was successful.
Business Rule	BR-35: Notifications must include relevant details about the downloaded report.
Dependencies	FR-117
Priority	Medium

3.3.31 UC-31: Provide Feedback to Admin

3.3.31.1 FR-119: Supervisor.ProvideFeedbackToAdmin.Input

Table 158: Supervisor.ProvideFeedbackToAdmin.Input

Identifier	**FR-119**
Title	Supervisor.ProvideFeedbackToAdmin.Input
Requirement	The supervisor shall be able to input feedback in a designated form.
Source	Shahryar
Rationale	Inputting feedback allows supervisors to communicate suggestions or concerns effectively.
Business Rule	BR-36: Feedback must include a valid subject and description.
Dependencies	None
Priority	High

3.3.31.2 FR-120: Supervisor.ProvideFeedbackToAdmin.Send

Table 159: Supervisor.ProvideFeedbackToAdmin.Send

Identifier	**FR-120**
Title	Supervisor.ProvideFeedbackToAdmin.Send
Requirement	The system shall allow the supervisor to send the feedback to the admin.
Source	Shahryar
Rationale	Sending feedback ensures that the admin receives the supervisor's input promptly.
Business Rule	BR-37: Feedback can only be sent if all required fields are completed.
Dependencies	FR-119
Priority	High

3.3.32 UC-32: View Safety Trends/Forecasts

3.3.32.1 FR-121: Supervisor.SafetyTrendsForecasts.View

Table 160: Supervisor.SafetyTrendsForecasts.View

Identifier	**FR-121**
Title	Supervisor.SafetyTrendsForecasts.View
Requirement	The supervisor shall be able to view safety trends and forecasts related to site conditions.
Source	Shahryar
Rationale	Understanding safety trends helps supervisors make informed decisions to improve workplace safety.
Business Rule	BR-38: Only authorized supervisors can access safety trends and forecasts.
Dependencies	None
Priority	High

3.3.32.2 FR-122: Supervisor.ViewSafetyTrendsForecasts.Sort

Table 161: Supervisor.ViewSafetyTrendsForecasts.Sort

Identifier	**FR-122**
Title	Supervisor.ViewSafetyTrendsForecasts.Sort
Requirement	The supervisor shall be able to sort the displayed safety trends and forecasts by various criteria.
Source	Shahryar
Rationale	Sorting capabilities enhance data usability and help supervisors quickly identify key information.
Business Rule	BR-39: Sorting options must include date, severity, and type of incident.
Dependencies	FR-121
Priority	Medium

3.3.33 UC-33: Give Usage/Help Tutorial

3.3.33.1 FR-123: Supervisor.GiveUsageHelpTutorial.View

Table 162: Supervisor.GiveUsageHelpTutorial.View

Identifier	**FR-123**
Title	Supervisor.GiveUsageHelpTutorial.View
Requirement	The supervisor shall be able to view the usage/help tutorial for the application.
Source	Shahryar
Rationale	Providing a tutorial helps supervisors understand how to use the system effectively.
Business Rule	BR-40: The tutorial must be accessible from the main dashboard.
Dependencies	None
Priority	High

3.3.34 UC-34: Supervisor's Task Log

3.3.34.1 FR-124: Supervisor.TaskLog.View

Table 163: Supervisor.TaskLog.View

Identifier	**FR-124**
Title	Supervisor.TaskLog.View
Requirement	The supervisor shall be able to view the task log of all activities performed.
Source	Shahryar
Rationale	Viewing the task log helps supervisors track completed tasks and maintain accountability.
Business Rule	BR-41: Only supervisors can view the task log.
Dependencies	None
Priority	High

3.3.34.2 FR-125: Supervisor.TaskLog.Create

Table 164: Supervisor.TaskLog.Create

Identifier	**FR-125**
Title	Supervisor.TaskLog.Create
Requirement	The supervisor shall be able to create a new entry in the task log.
Source	Shahryar
Rationale	Creating entries in the task log allows supervisors to document their activities and responsibilities.
Business Rule	BR-42: Task log entries must include a date and description.
Dependencies	FR-124
Priority	High

3.3.34.3 FR-126: Supervisor.TaskLog.Save

Table 165: Supervisor.TaskLog.Save

Identifier	**FR-126**
Title	Supervisor.TaskLog.Save
Requirement	The supervisor shall be able to save the entries created in the task log.
Source	Shahryar
Rationale	Saving task log entries ensures that the records are maintained for future reference.
Business Rule	BR-43: Saved entries must be accessible for future editing.
Dependencies	FR-125
Priority	High

3.3.34.4 FR-127: Supervisor.TaskLog.Update

Table 166: Supervisor.TaskLog.Update

Identifier	**FR-127**
Title	Supervisor.TaskLog.Update
Requirement	The supervisor shall be able to update an existing entry in the task log.
Source	Shahryar
Rationale	Updating task log entries ensures that the information remains accurate and reflects any changes in tasks.
Business Rule	BR-44: Only the creator of the task log entry can update it.
Dependencies	FR-124, FR-125
Priority	High

3.3.34.5 FR-128: Supervisor.TaskLog.Delete

Table 167: Supervisor.TaskLog.Delete

Identifier	**FR-128**
Title	Supervisor.TaskLog.Delete
Requirement	The supervisor shall be able to delete an existing entry from the task log.
Source	Shahryar
Rationale	Deleting unnecessary or incorrect entries keeps the task log relevant and organized.
Business Rule	BR-45: Deleted entries cannot be recovered unless specifically archived.
Dependencies	FR-124
Priority	High

3.3.35 UC-35: Visualization

3.3.35.1 FR-129: Visualization.AccessGraphs

Table 168: Visualization.AccessGraphs

Identifier	**FR-129**
Title	Access Graphical Visualization
Requirement	The supervisor shall be able to access the graphical representation of anomalies and trends from the dashboard.
Source	Zamin
Rationale	To provide an overview of site performance visually.
Business Rule	Only authorized supervisors can view graphs.
Dependencies	None
Priority	High

3.3.35.2 FR-130:Visualization.ViewAnomalyTrends

Table 169: Visualization.ViewAnomalyTrends

Identifier	FR-130
Title	Visualization.ViewAnomalyTrends
Requirement	The supervisor shall be able to view trends related to anomalies over a selected time range.
Source	Zamin
Rationale	Understanding the frequency and type of anomalies for decision-making.
Business Rule	Data must be updated in real-time.
Dependencies	FR-129
Priority	High

30.3.11.3 FR-131: Visualization.FilterDataByTimeRange

Table 170: Visualization.FilterDataByTimeRange

Identifier	FR-131
Title	Visualization.FilterDataByTimeRange
Requirement	The supervisor shall be able to filter graphical data based on specific time ranges (e.g., daily, weekly, monthly).
Source	Zamin
Rationale	Provides flexibility in analyzing data over different periods.
Business Rule	Time filters should be customizable by the user.
Dependencies	FR-129
Priority	Medium

3.3.35.4 FR-132: Visualization.DrillDownAnomalyDetails

Table 171: Visualization.DrillDownAnomalyDetails

Identifier	**FR-132**
Title	Drill Down into Anomaly Details
Requirement	The supervisor shall be able to click on a specific graph data point to view detailed information about the anomaly.
Source	Zamin
Rationale	Helps in understanding specific incidents in detail.
Business Rule	Drill-down data should be accessible only for authorized users.
Dependencies	FR-130
Priority	High

3.3.35.5 FR-133 Visualization.ExportGraphAsImage

Table 172: Visualization.ExportGraphAsImage

Identifier	**FR-133**
Title	Visualization.ExportGraphAsImage
Requirement	The supervisor shall be able to export any graph or trend analysis as an image file (e.g., PNG, JPEG).
Source	Zamin
Rationale	To provide easy sharing or documentation of graphical reports.
Business Rule	Only supervisors can export the data.
Dependencies	FR-129
Priority	Medium

3.3.35.6 FR-134 Visualization.HandleNoDataAvailable

Table 173: Visualization.HandleNoDataAvailable

Identifier	**FR-134**
Title	Visualization.HandleNoDataAvailable
Requirement	The system shall display a message or placeholder when there is no data available for a selected time range or anomaly type.
Source	Zamin
Rationale	Improves user experience by clearly indicating the absence of data.
Business Rule	A message should be displayed when there is no data to avoid confusion.
Dependencies	FR-129
Priority	Low

3.3.36 FRs For Events

3.3.36.1 FR: Alert.Prioritize

Table 174: Alert.Prioritize

Identifier	**FR-135**
Title	Alert.Priortize
Requirement	The system shall prioritize alerts based on their severity, urgency, and type, ensuring the most critical ones are addressed first.
Source	Zamin
Rationale	Ensures that high-severity incidents are handled before lower-priority ones.
Business Rule	Critical alerts must be displayed at the top of the alert dashboard.
Dependencies	None
Priority	High

3.3.36.2 FR:Alert.Severity.Check

Table 175: Alert.Severity.Check

Identifier	**FR-136:**
Title	Alert.Severity.Check
Requirement	The system shall evaluate and assign a severity level to each alert, determining whether it is critical, warning, or low-level.
Source	Zamin
Rationale	Helps in categorizing alerts based on the severity for proper action.
Business Rule	Severity levels must be configurable.
Dependencies	FR-135
Priority	High

3.3.36.3 FR: Alert.Priority.Update

Table 176: Alert.Priority.Update

Identifier	**FR-137**
Title	Update Alert Priority
Requirement	The system shall allow supervisors to manually adjust the priority level of an alert if they deem it necessary.
Source	Zamin
Rationale	Provides flexibility to supervisors in handling alerts.
Business Rule	Only authorized users can update alert priorities.
Dependencies	FR-135, FR-136
Priority	Medium

3.3.36.4 FR-138: Alert.Escalation.Trigger

Table 177: Alert.Escalation.Trigger

Identifier	**FR-138: Alert.Escalation.Trigger**
Title	Trigger Alert Escalation

Requirement	The system shall trigger an escalation process if a critical alert is not addressed within a predefined time frame.
Source	Zamin
Rationale	Ensures that critical alerts receive timely responses.
Business Rule	Escalation timing must be configurable.
Dependencies	FR-135, FR-136
Priority	High

3.3.36.5 FR-139: Alert.Escalation.Alternate.Contact

Table 178: Alert.Escalation.Alternate.Contact

Identifier	**FR-139**
Title	Escalate to Alternate Contact
Requirement	The system shall escalate an alert to an alternate contact if the primary contact does not respond within a set time.
Source	Zamin
Rationale	Ensures redundancy in the alert escalation process.
Business Rule	Alternate contacts must be configurable by admin.
Dependencies	FR-138
Priority	Medium

3.3.36.6 FR-140:Alert.Escalation.Retry

Table 179: Alert.Escalation.Retry

Identifier	**FR-140**
Title	Retry Alert Escalation

Requirement	The system shall automatically retry the escalation process if no response is received after the first attempt.
Source	Zamin
Rationale	Increases the chances of timely response to critical incidents.
Business Rule	The number of retries must be configurable.
Dependencies	FR-138, FR-139
Priority	Medium

3.3.36.7 FR-141: Report.Auto.Generate

Table 180: Report.Auto.Generate

Identifier	**FR-141**
Title	Auto Generate Reports
Requirement	The system shall automatically generate reports based on predefined criteria at regular intervals.
Source	Zamin
Rationale	Saves time by automating the report generation process.
Business Rule	The reporting schedule must be configurable.
Dependencies	None
Priority	High

3.3.36.8 FR-142: Report.Scheduled.Generation

Table 181: Report.Scheduled.Generation

Identifier	**FR-142**
Title	Scheduled Report Generation

Requirement	The system shall allow users to schedule report generation for specific times or intervals (e.g., daily, weekly).
Source	Zamin
Rationale	Provides flexibility in generating reports when required.
Business Rule	Users must be able to configure the report generation schedule.
Dependencies	FR-141
Priority	Medium

3.3.36.9 FR-143: Report.Custom.Data.Range

Table 182: Report.Custom.Data.Range

Identifier	**FR-143**
Title	Generate Reports for Custom Data Range
Requirement	The system shall allow users to generate reports for a custom date range by selecting start and end dates.
Source	Zamin
Rationale	Enables users to generate specific reports based on their needs.
Business Rule	Date range selection must be flexible and accurate.
Dependencies	FR-141
Priority	Medium

3.3.36.10 FR-144:Forecast.Predict.Safety.Risks

Table 183: Forecast.Predict.Safety.Risks

Identifier	**FR-144**
Title	Predict Safety Risks

Requirement	The system shall analyze data to predict potential safety risks at the construction site.
Source	Zamin
Rationale	Helps in taking preventive measures for site safety.
Business Rule	Predictions must be based on historical data and trends.
Dependencies	None
Priority	High

3.3.36.11 FR-145:Forecast.Anomaly.Trend.Analysis

Table 184: Forecast.Anomaly.Trend.Analysis

Identifier	**FR-145**
Title	Analyze Anomaly Trends
Requirement	The system shall analyze anomaly trends to provide insights into recurring safety violations or unusual activities.
Source	Zamin
Rationale	Helps in identifying patterns and improving safety measures.
Business Rule	Trend analysis must be updated in real-time.
Dependencies	None
Priority	High

3.3.36.12 FR-146:Forecast.Suggest.Prevention

Table 185: Forecast.Suggest.Prevention

Identifier	**FR-146**

Title	Suggest Safety Prevention Measures
Requirement	The system shall suggest preventive measures based on the prediction of safety risks and anomaly trends.
Source	Zamin
Rationale	Helps supervisors take proactive steps to ensure site safety.
Business Rule	Suggestions must be based on valid data trends and analysis.
Dependencies	FR-144, FR-145
Priority	High

3.3.36.13 FR-147: CameraSecure.VideoConnection

Table 186: CameraSecure.VideoConnection

Identifier	FR-147
Title	CameraSecure.VideoConnection
Requirement	The system shall ensure a secure, encrypted real-time video connection for surveillance cameras.
Source	Zamin
Rationale	Ensures secure and uninterrupted video streaming for monitoring.
Business Rule	The connection must support encryption and automatic recovery from disconnections.
Dependencies	None
Priority	High

3.3.36.14 FR-148: Camera.HealthMonitoring

Table 187: Camera.HealthMonitoring

Identifier	FR-148
Title	Camera.HealthMonitoring
Requirement	The system shall continuously monitor camera health and generate real-time alerts for connectivity or performance issues.

Source	Zamin
Rationale	Ensures that all cameras are functioning correctly for proper monitoring.
Business Rule	Alerts should be generated instantly in case of malfunctions.
Dependencies	None
Priority	High

3.3.36.15 FR-149: CameraVideo.Streaming.Storage

Table 188: CameraVideo.Streaming.Storage

Identifier	**FR-149**
Title	CameraVideo .Streaming.Storage
Requirement	The system shall provide real-time video streaming and store footage securely in local and cloud storage.
Source	Zamin
Rationale	Ensures seamless access to live video and archived footage.
Business Rule	Footage must be stored redundantly to avoid data loss.
Dependencies	None
Priority	High

3.3.36.16 FR-150: Camera.Cloud.Storage-Retrieval

Table 189: Camera.Cloud.Storage-Retrieval

Identifier	**FR-150**
Title	Camera.Cloud.Storage-Retrieval
Requirement	The system shall support the retrieval of stored video footage from cloud storage for on-demand access.

Source	Zamin
Rationale	Allows users to access archived footage for investigations or reviews.
Business Rule	Cloud storage must have proper access controls and be highly available.
Dependencies	None
Priority	Medium

3.3.36.17 FR-151: Video.Chunk.Creation

Table 190: Video.Chunk.Creation

Identifier	**FR-151**
Title	Anomaly Video Chunk Creation
Requirement	The system shall split video feeds into chunks for processing in the anomaly detection model.
Source	Zamin
Rationale	Prepares video feeds for efficient anomaly detection.
Business Rule	Video chunks should be of uniform length for model input.
Dependencies	None
Priority	High

3.3.36.18 FR-152: Video.Chunk.Processing

Table 191: Video.Chunk.Processing

Identifier	**FR-152**
Title	Video.Chunk.Processing
Requirement	The system shall send video chunks to the anomaly detection model for analysis.

Source	Zamin
Rationale	Ensures that the system can detect anomalies efficiently.
Business Rule	Video chunks should be processed in real-time for prompt anomaly detection.
Dependencies	FR-151
Priority	High

3.3.36.19 FR-153 : Anomaly.PPE Detection.Model

Table 192: Anomaly.PPE.Detection.Model

Identifier	**FR-153**
Title	Anomaly PPE Detection Model
Requirement	The system shall detect PPE compliance in video feeds using machine learning models.
Source	Zamin
Rationale	Enhances safety by ensuring workers wear appropriate PPE.
Business Rule	Detection accuracy must meet a predefined threshold for reliability.
Dependencies	FR-152
Priority	High

3.3.36.20 FR-154 Anomaly.Alert.Generation

Table 193: Anomaly.Alert.Generation

Identifier	**FR-154**
Title	Anomaly Alert Generation
Requirement	The system shall generate alerts when anomalies are detected in real-time.
Source	Zamin

Rationale	Immediate alerts help supervisors take prompt action to address safety issues.
Business Rule	Alerts must be based on predefined thresholds and must be delivered to the relevant personnel.
Dependencies	FR-153
Priority	High

3.3.37 FRs For UC-36: AI Assistance/Chatbot

3.3.37.1 FR-155: Chatbot.QueryProcessing

Field	Details
Identifier	FR-140
Title	Chatbot.QueryProcessing
Requirement	The AI Assistant shall be able to process natural language queries from supervisors and convert them into searchable vectors for accurate retrieval.
Source	SafeSitePlus Development Team
Rationale	Enables supervisors to interact with the chatbot naturally without needing specific commands.
Business Rule	Queries should be processed in real-time with minimal delay.
Dependencies	None
Priority	High

3.3.37.2 FR-156: Chatbot.EmbeddingCreation

Field	Details
Identifier	FR-141
Title	Chatbot.EmbeddingCreation

Requirement	The system shall generate embeddings from company policies, safety guidelines, and other reference documents to facilitate accurate response generation.
Source	SafeSitePlus Development Team
Rationale	Ensures that chatbot responses are based on officially documented information.
Business Rule	Embeddings should be updated regularly as documents are modified.
Dependencies	FR-155
Priority	High

3.3.37.3 FR-157: Chatbot.ChunkSelection

Field	Details
Identifier	FR-142
Title	Chatbot.ChunkSelection
Requirement	The system shall match a user's query with the most relevant document chunks based on semantic similarity and rank them before response generation.
Source	SafeSitePlus Development Team
Rationale	Improves the accuracy of chatbot responses by ensuring only the most relevant data is used.
Business Rule	The system should retrieve and rank at least the top three relevant chunks per query.
Dependencies	FR-157
Priority	High

3.3.37.4 FR-158: Chatbot.IntegrationWithLLM

Field	Details
Identifier	FR-143
Title	Chatbot.IntegrationWithLLM

Requirement	The chatbot shall integrate with a Large Language Model (LLM) such as Gemini to generate contextually relevant and structured responses.
Source	SafeSitePlus Development Team
Rationale	Leverages advanced AI capabilities to enhance response quality and ensure contextual understanding.
Business Rule	The LLM should only generate responses based on retrieved document chunks to maintain accuracy.
Dependencies	FR-158
Priority	High

3.3.37.5 FR-159: Chatbot.QueryLoggingAndAnalytics

Field	Details
Identifier	FR-145
Title	Chatbot.QueryLoggingAndAnalytics
Requirement	The system shall log all chatbot queries and responses for analysis and future improvements.
Source	SafeSitePlus Development Team
Rationale	Helps in improving chatbot accuracy and understanding usage patterns.
Business Rule	Logs should be anonymized and stored securely, with access restricted to authorized personnel.
Dependencies	FR-158
Priority	Medium

3.4 Non-Functional Requirements

This section specifies nonfunctional requirements other than constraints. These quality requirements should be specific, quantifiable and verifiable. The Non-functional requirements related to our system includes Usability, Performance, Interoperability, Security and Integrity.

3.4.1 Reliability

The following features show the reliability of the application.

Availability: The web application are available 24/7 for the users.
Mean Time Between Failures (MTBF): System is efficient and there are no as such big issues, such as real-time anomaly detection, camera streaming
However, it will be down maybe once in a month for maintenance. The software is reliable. It just needs
good internet connection.
Mean Time to Repair (MTTR): The system will take maximum 4 hours to repair.
Efficiency: The system works efficiently and all the user issues, if any, are resolved timely.

3.4.2 Usability

The following features show the usability of the application.
USE-1: The interface of the website is user friendly. One can easily use it if he/she knows Basic English.
USE-2: Notifications for anomalies, camera statuses, and fatigue levels should be clear and easy to understand.
USE-3: The interface for monitoring and managing alerts should be intuitive, allowing users to easily understand and interact with live camera feeds, anomaly alerts, and reports.
USE-4: All destructive actions (such as deleting a user, removing a camera, or deleting incident logs) will require a confirmation step to prevent accidental loss of important data or configurations.
USE-5: The Safesiteplus system will offer visual feedback for user actions (such as successful form submissions, setting changes, or data retrieval), so that users are clearly informed about the status of their requests.

3.4.3 Performance

The following features show the performance of the application.

PER-1: PPE compliance and anomaly detection alerts shall be generated within 5-10 seconds of detecting a safety violation on the construction site.
PER-2: The supervisor's dashboard shall load and display all necessary data (View surveillance Footage, task management updates, etc.) within 10 seconds of login.
PER-3: The system shall generate incident reports and make them available to the supervisor within 10 seconds of the report generation request.

PER-4: The system shall process weather data updates and display relevant information to the supervisor interface within 10 seconds of receiving the weather API data.
PER-5 : Reports available for download within 15 seconds of the request.

3.4.4 Security

SEC-1: All users shall be required to provide valid login credentials to access any part of the system.
SEC-2: Each user shall have access only to the features, data, and reports that align with their role (admin or supervisor) within the system, ensuring role-based access control.
SEC-3 Data related to anomalies, camera statuses, and weather conditions should remain consistent and accurate, with no unauthorized modifications.
SEC-4: The system shall enforce automatic session timeouts after a period of inactivity, requiring users to reauthenticate to continue using the platform.

3.5 External Interface Requirements

This section provides information to ensure that the system will communicate properly with This section provides information to ensure that the system will communicate properly with users and with external hardware or software elements. A complex system with multiple subcomponents should create a separate interface specification or system architecture specification. The interface documentation could incorporate material from other documents by reference. For instance, it could point to a hardware device manual that lists the error codes that the device could send to the software.

User Interfaces Requirements

The Safesiteplus user interface will be designed with accessibility, usability, and responsiveness in mind. The system will adhere to best practices for user experience to ensure that users, including administrators and supervisors, can efficiently navigate the platform and manage site safety operations.

UI-1: The color scheme shall use soft, neutral colors to reduce eye strain for users, especially when working on monitoring systems for extended periods.
UI-2: A collapsible side menu shall be available on all screens, providing access to key modules such as Anomaly Detection, User Administration, Incident Management, and Reporting.
UI-3: A back button shall be displayed on each screen for easy navigation between pages.
UI-4: A search bar shall be prominently placed on the main dashboard, allowing users to search for incidents, reports, or safety trends.
UI-5: Error messages displayed to users shall be clear, concise, and non-technical, providing possible solutions where applicable.

UI-6: Tooltips shall be provided for certain icons and buttons to guide users on how to utilize specific features effectively.
UI-7: The system shall support keyboard navigation and provide shortcuts for frequent actions, such as "Ctrl + S" to save configurations.
UI-8: A customizable dashboard shall allow supervisors to modify the arrangement of key metrics, visualizations, and alerts.
UI-9: The system will support both light and dark mode options, with consistent font standards, iconography, and button styles across both modes.
UI-10: The interface shall be optimized for different screen sizes, ensuring responsiveness on desktops, tablets, and mobile devices.

3.5.1 Software interfaces

Safesiteplus will need to integrate with various software tools, APIs, and databases to enable key functionalities such as anomaly detection, incident logging, and analytics.

SI-1: Safesiteplus shall use the Google Maps API to provide real-time location data and mapping services for monitoring site activities and responding to incidents.

SI-2: The system shall ensure compatibility with modern browsers, including Chrome, Firefox, and Safari, to ensure a consistent experience across platforms.

SI-3: Safesiteplus shall communicate with third-party CCTV software systems (e.g., Milestone, Genetec) via their APIs to retrieve real-time video feeds for anomaly detection.

SI-4: The system shall integrate with cloud storage solutions like AWS S3 for video storage and retrieval.

SI-5: Safesiteplus will integrate with incident management software to ensure seamless tracking and reporting of safety issues.

3.5.2 Hardware interfaces

SafeSitePlus integrates with various hardware systems to process video feeds, support AI-based monitoring, and enhance site safety through automated detection mechanisms.

HI-1: The system shall interface with on-site CCTV cameras to process video feeds for anomaly detection, PPE compliance verification, and hazard identification.

HI-2: SafeSitePlus shall be compatible with servers capable of processing video streams and running machine learning models for real-time anomaly detection.

HI-3: The platform shall support GPU-enabled processing units to optimize AI-based video analysis for improved performance and efficiency.

HI-4: SafeSitePlus will communicate with storage systems to securely store, retrieve, and analyze recorded video data for compliance and incident reporting.

3.5.3 Communications interfaces

SafeSitePlus will use secure and reliable communication methods to exchange information between the system, users, and other platforms.

CI-1: The system shall securely communicate with backend services using HTTPS to protect data in transit.

CI-2: The system shall securely communicate with the database using encrypted connections to ensure data integrity and confidentiality.

CI-3: SafeSitePlus shall send real-time email notifications to supervisors when a critical incident is detected.

CI-4: The system shall provide in-app alerts within the dashboard to immediately notify supervisors about safety violations or anomalies.

4 Design and Architecture

4.1 Architectural Design

4.1.1 Box And Line Diagram

. The following figure shows the box and line diagram of SafeSitePlus.

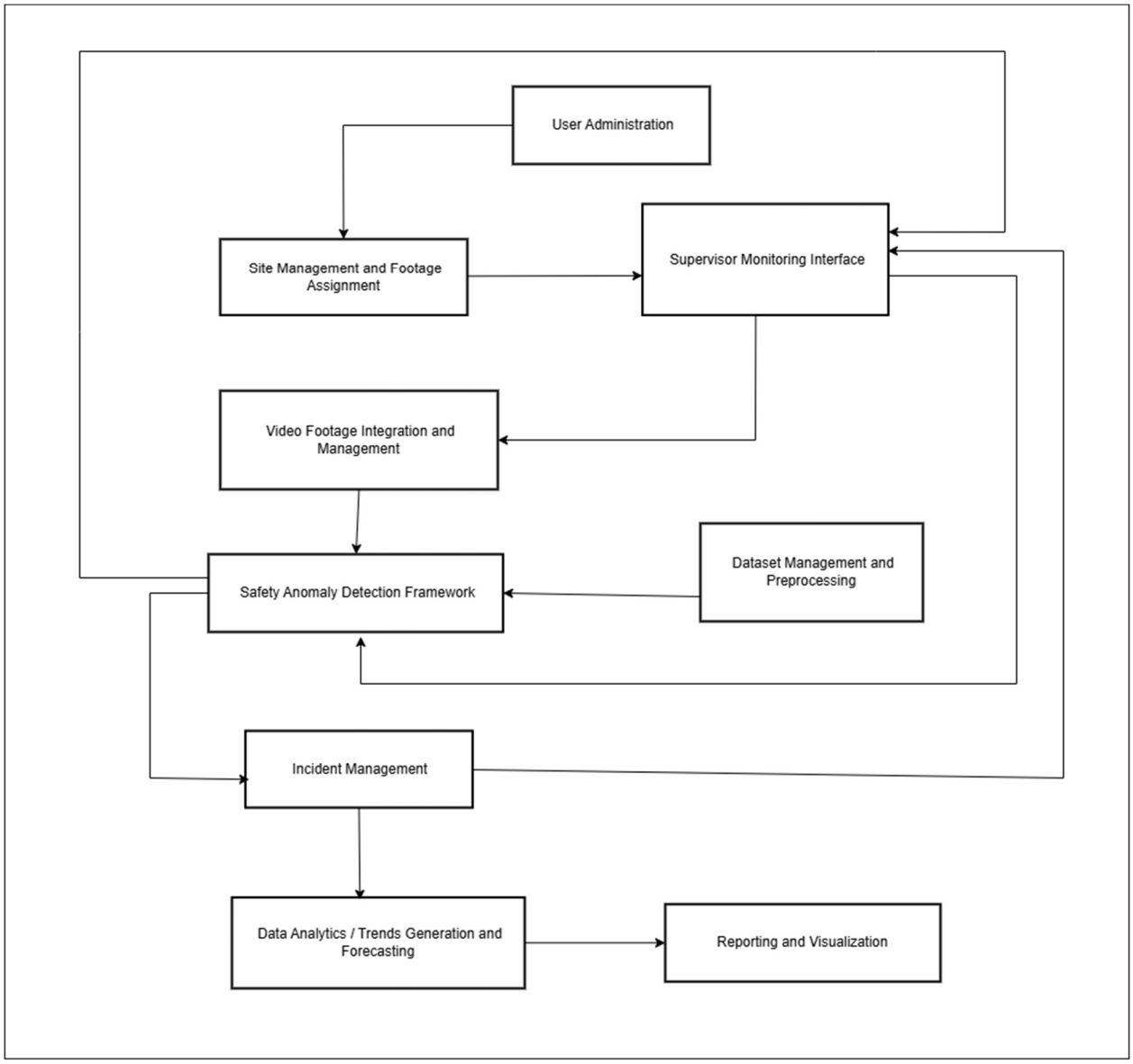


Figure 3: Box and Line Diagram of SafeSite Plus

4.1.2 Architecture Diagram

The system follows a three-tier architecture. The front-end serves as the presentation tier, providing user interface elements that enable users to interact with the system. This includes web-based dashboards and interfaces used by admins, supervisors, and other users to manage site operations, monitor data, and review analytics in SafeSitePlus. The application tier houses the business logic and processing of the system. It handles tasks such as anomaly detection, dataset preprocessing, video footage analysis, incident management, and data analytics. Core functionalities like user administration, role-based access, anomaly detection, and reporting are implemented in this layer. The application layer communicates with the data tier, which manages the storage and retrieval of data. In SafeSitePlus, this includes databases or cloud storage systems used for storing video footage, processed datasets, user profiles, incident logs, and historical analytics data. This tiered architecture ensures seamless interaction, efficient processing, and secure data management.

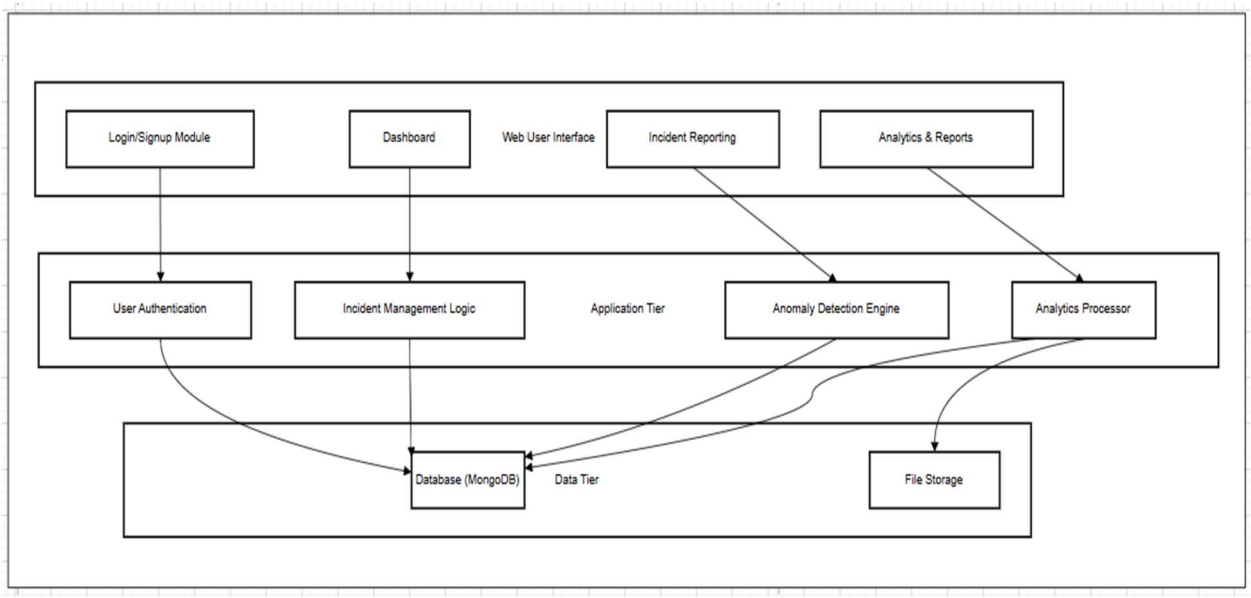


Figure 4: Architectural Diagram for SafeSitePlus

4.2 Design Models

4.2.1 Activity Diagram

30.3.11.3 Login

The following diagram outlines the user login workflow, including entering credentials, handling errors, password recovery, and accessing the dashboard.

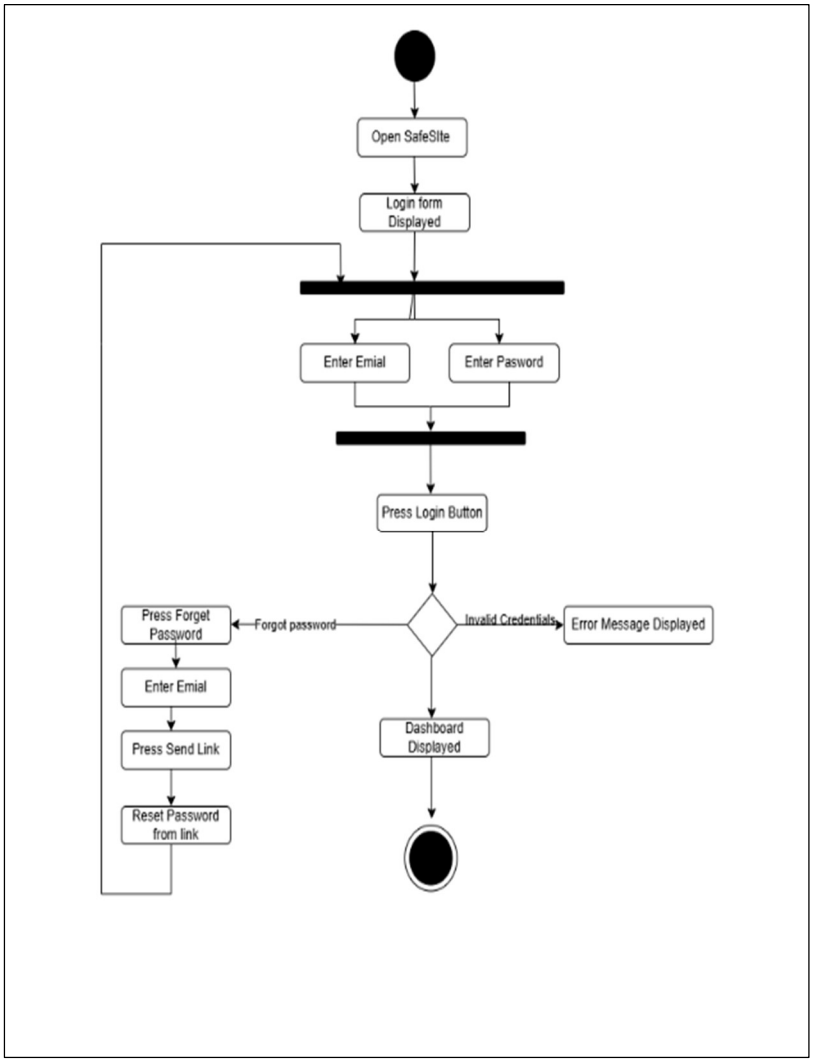


Figure 5 Activity Diagram for Login

4.2.1.2 Add Supervisor

This diagram outlines the user login workflow, including entering credentials, handling errors, password recovery, and accessing the dashboard.

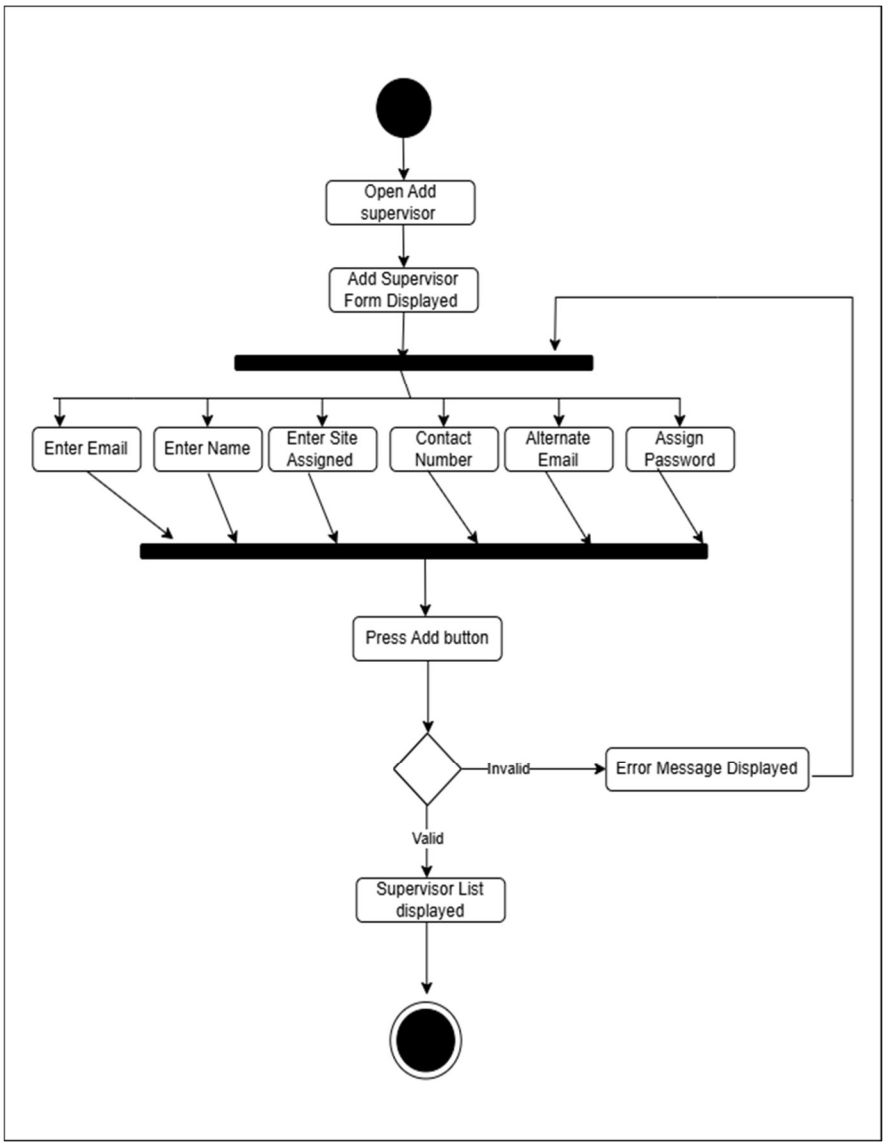


Figure 6 Activity Diagram to Add supervisor

4.2.1.3 Delete Supervisor

This diagram represents the process of deleting a supervisor, including selecting a supervisor, confirming deletion, and handling success or errors.

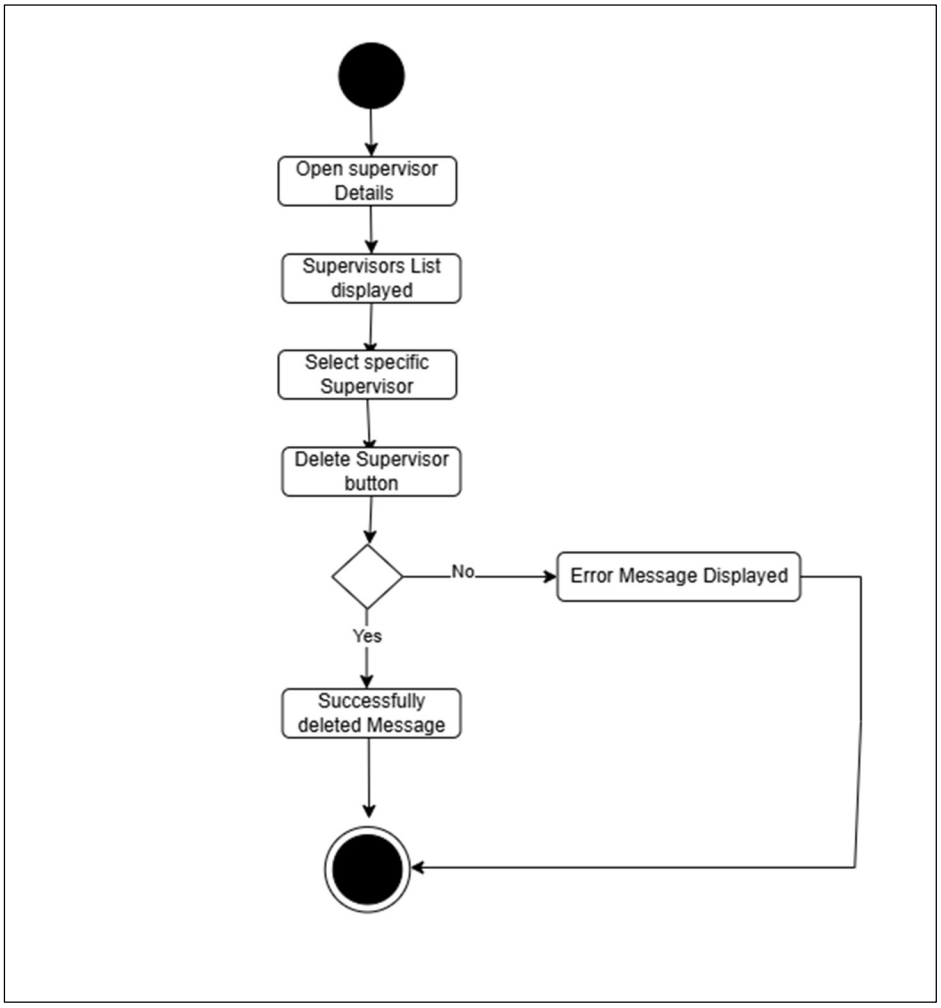


Figure 7 Activity Diagram to Delete Supervisor

4.2.1.4 Register a Site

The following diagram outlines the process of adding a site, including entering site details, handling validation, and displaying the updated site list.

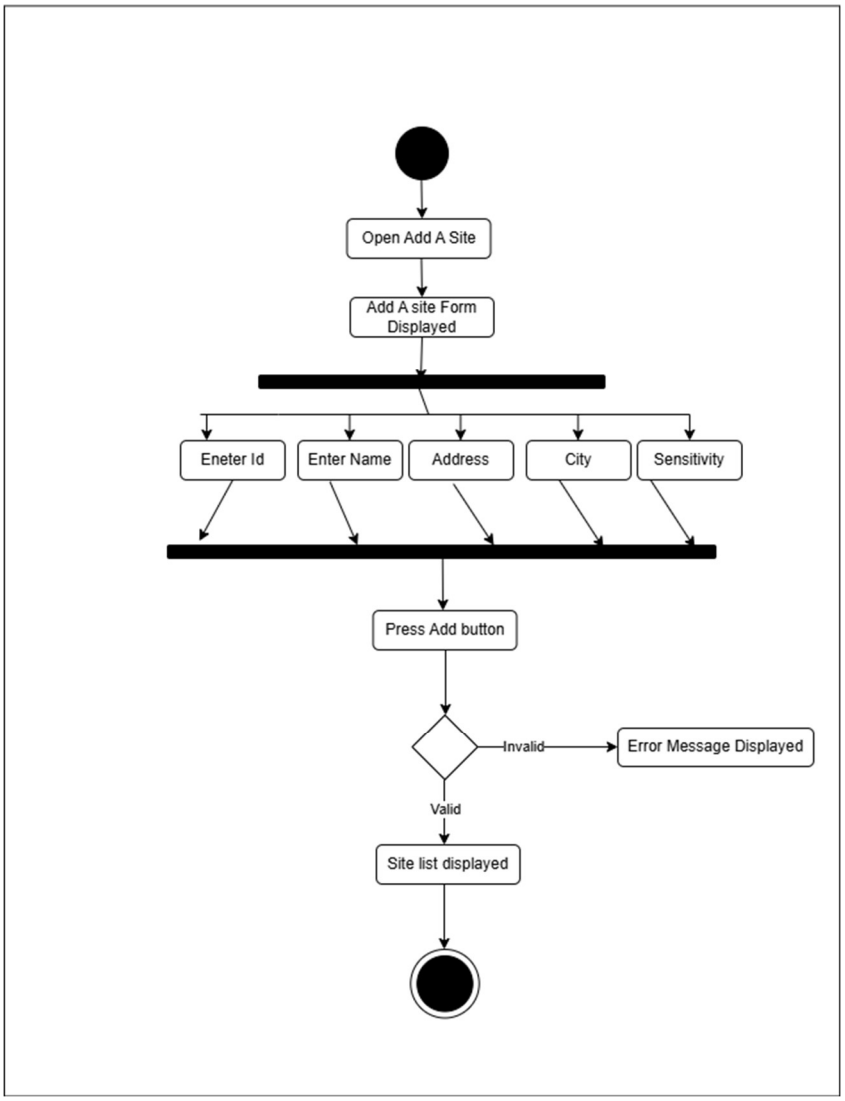


Figure 8 Activity Diagram to Register a Site

4.2.1.5 Update Supervisor Credentials

This diagram depicts the process of updating a supervisor's details, including editing information, handling validation, and displaying the updated supervisor list.

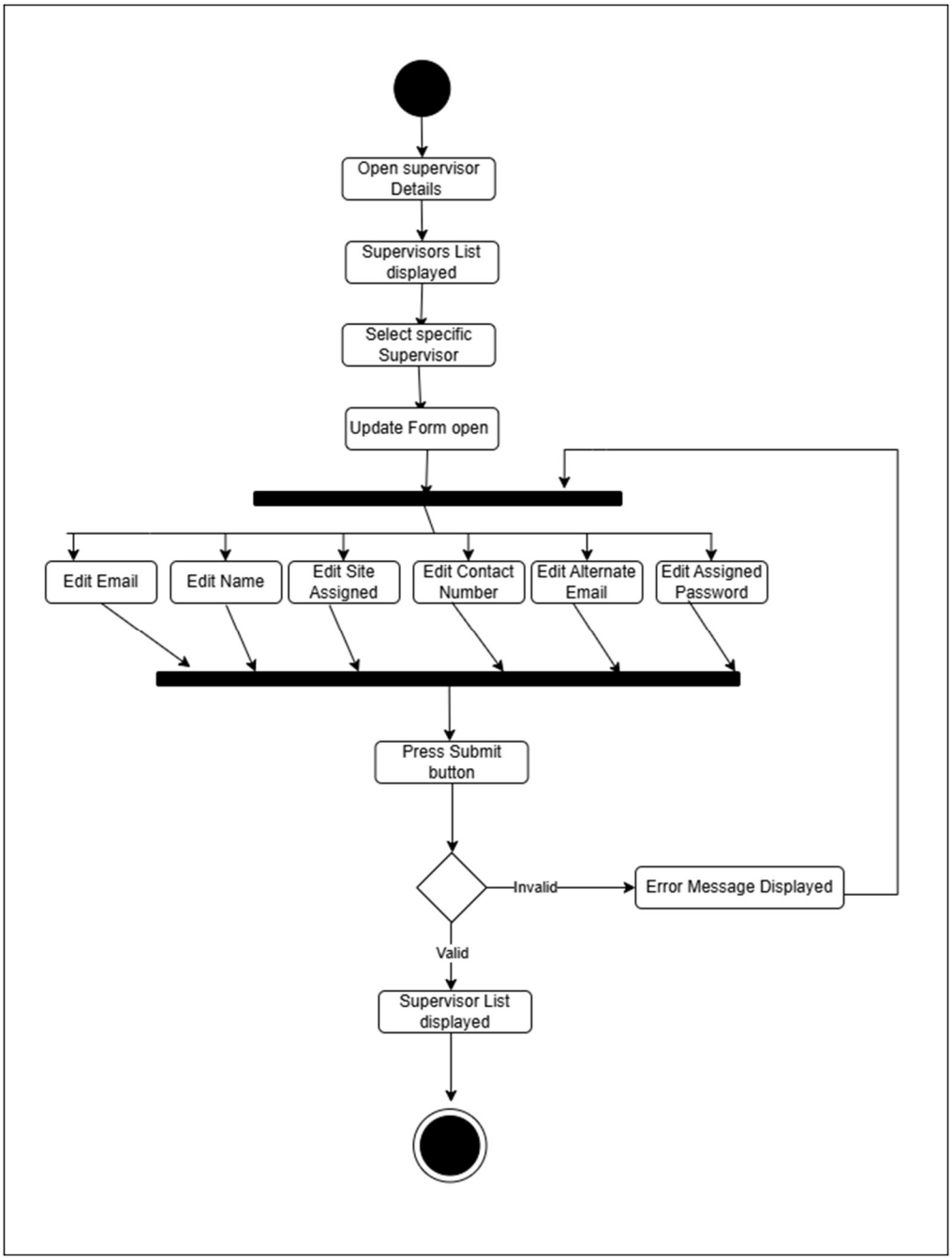


Figure 9 Activity Diagram to Update Supervisor Credentials

30.3.11.3 Customize Anomaly Parameters

This diagram illustrates the process of customizing anomaly parameters for a specific site, including selecting parameters and updating them successfully.

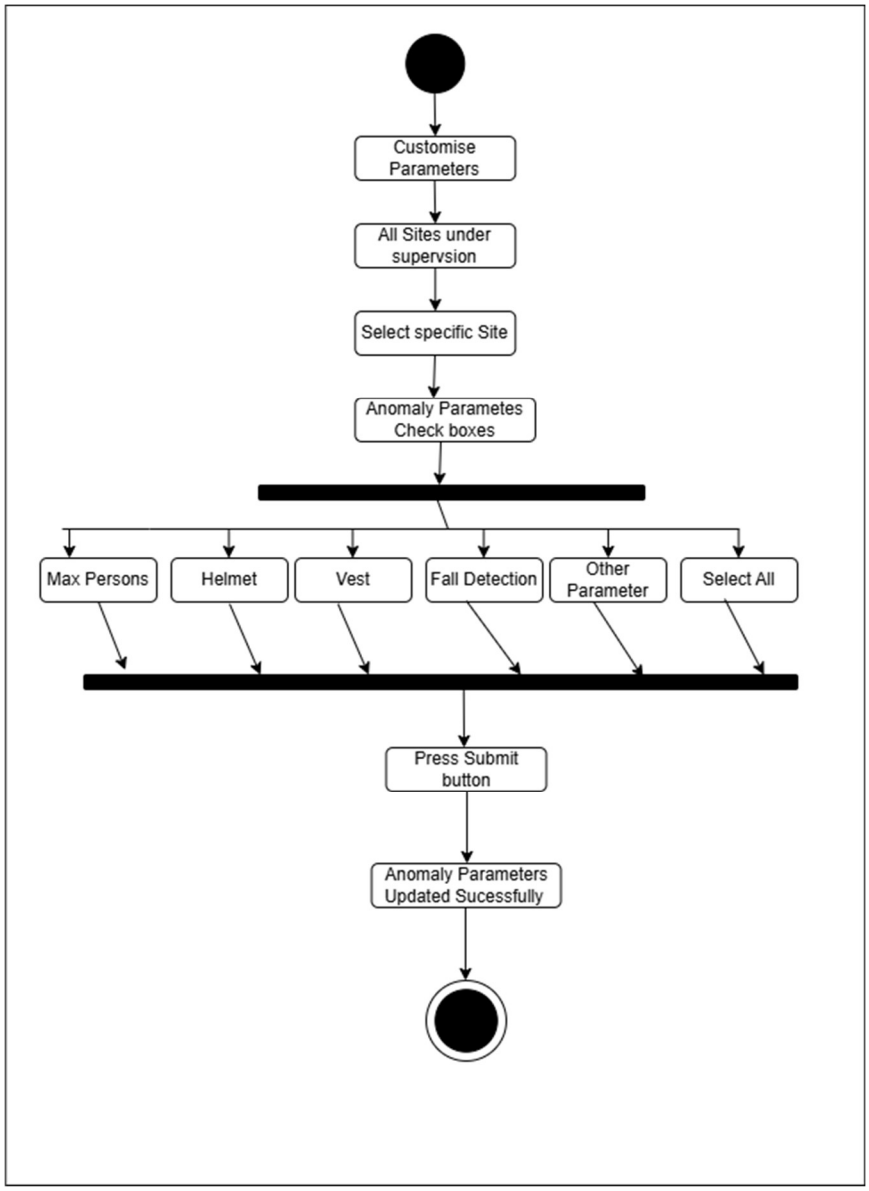


Figure 10 Activity Diagram to customize Anomaly Parameter for Site

30.3.11.3 Get Video Feed

This diagram outlines the process of accessing and filtering surveillance feeds, including viewing filtered sites and detailed feed information.

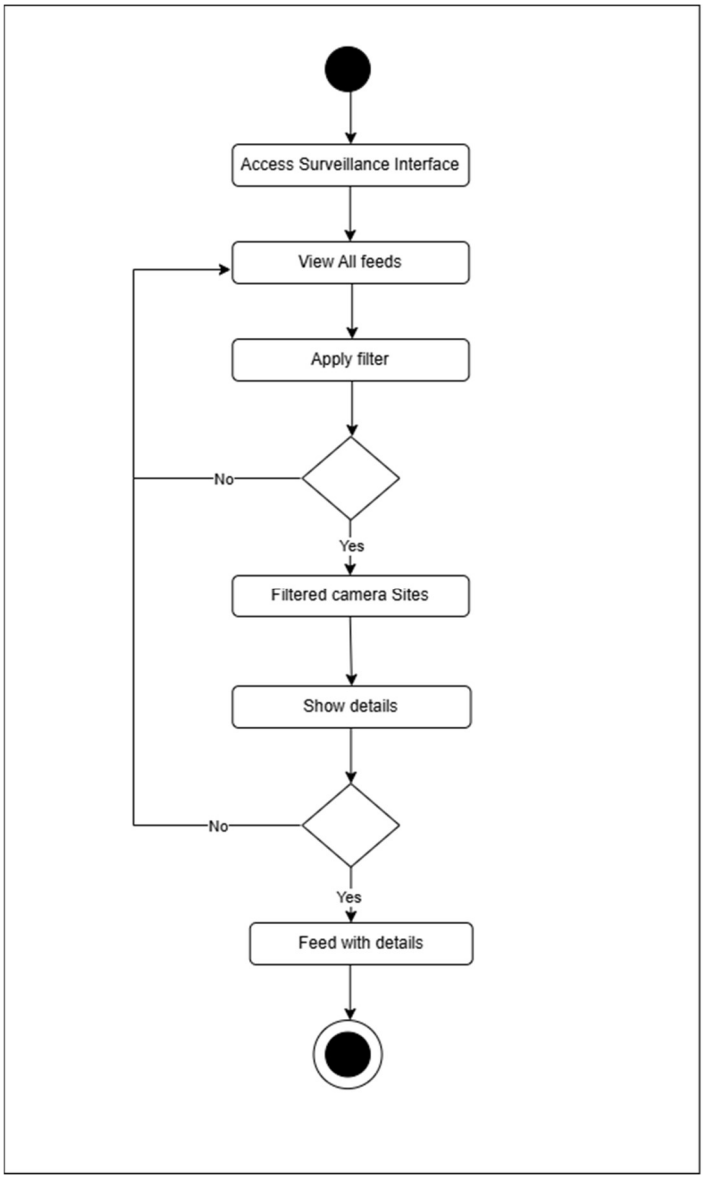


Figure 11 Activity Diagram to Get video Feed

4.2.1.8 View and Filter Alerts

This diagram depicts the process of viewing and filtering alerts, including selecting a specific alert, viewing details, categorizing it, and marking it as seen.

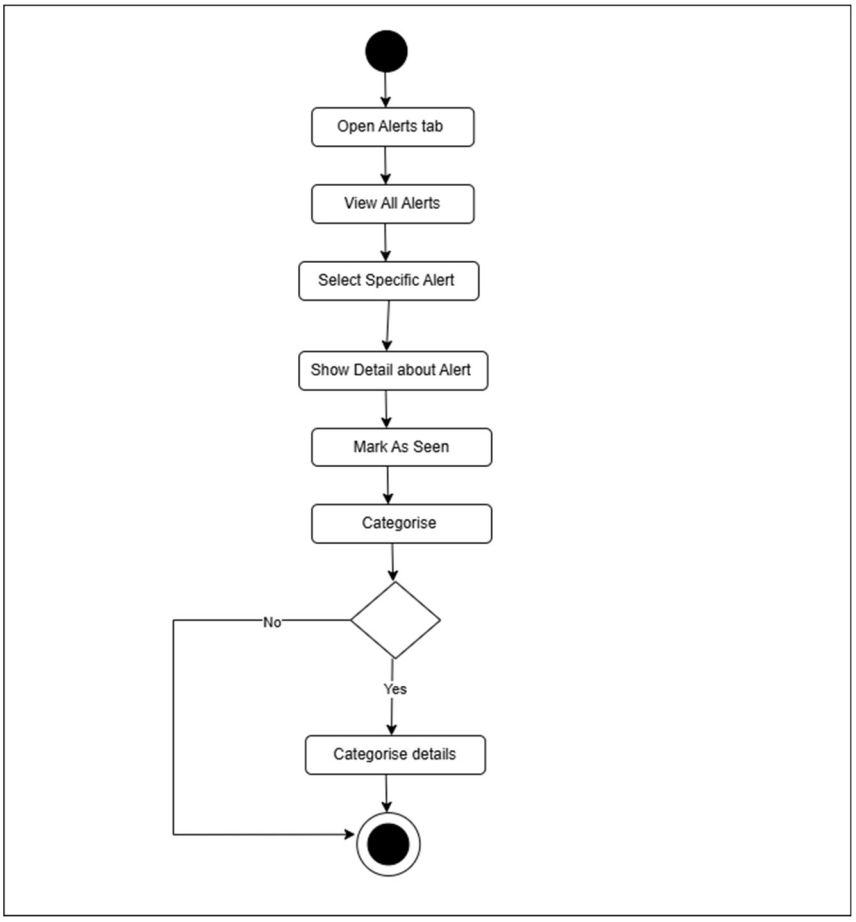


Figure 12 Activity Diagram to View Alerts

4.2.1.9 Reports and Graphs

This diagram depicts the process of Generating and Exporting reports.

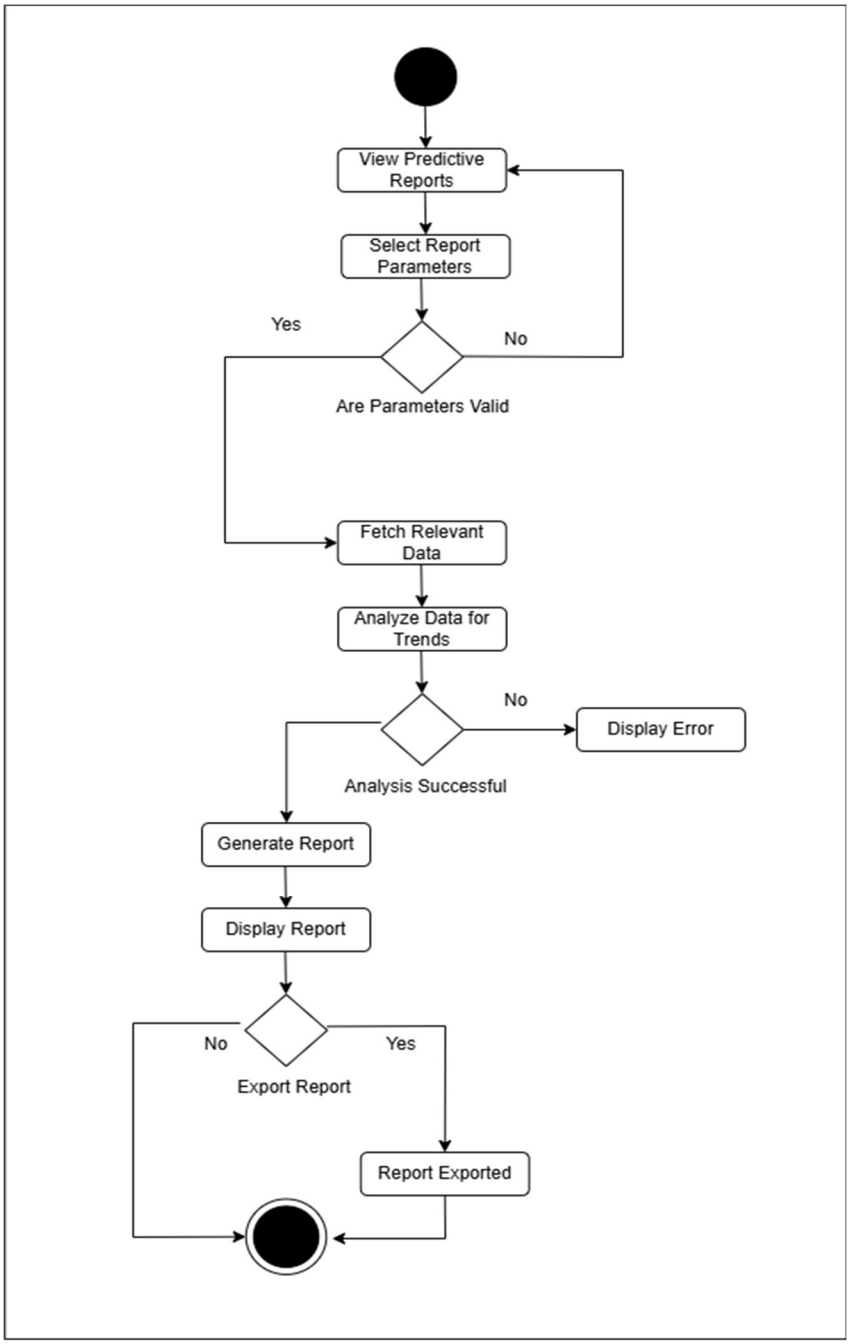


Figure 13 Activity Diagram to Generate and Export Reports

4.2.2 Data Flow Diagram

30.3.11.3 *Level 0 – DFD*

The following figure shows the level 0 data flow diagram of the system. It shows the major entities of the system and their dataflow. It tells what data flows in and out of the system and to which entities. It is at a higher level of abstraction and gives an overview of the system and its dataflow

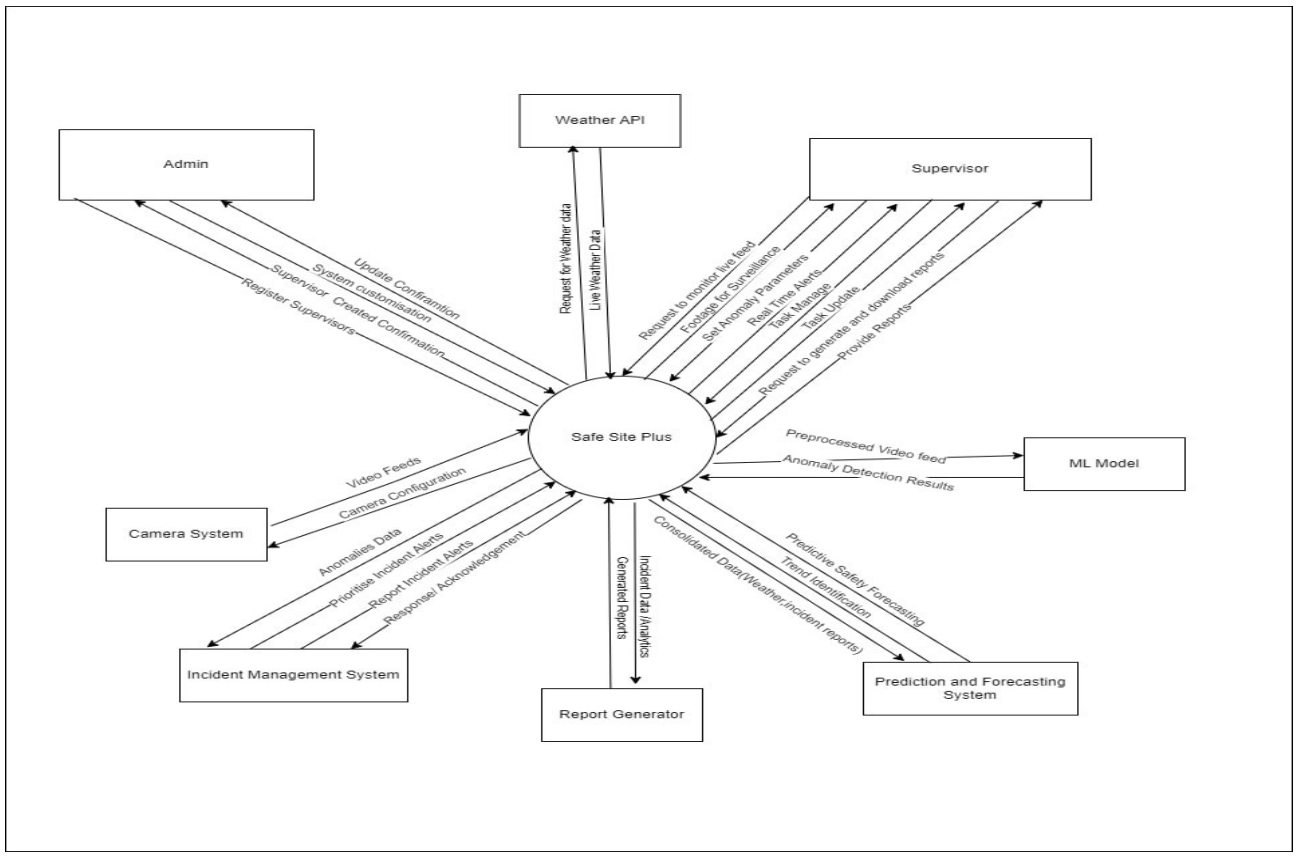


Figure 14 Level 0 Data Flow Diagram

4.2.2.2 DFD level 1

The following figures show the level 1 data flow diagram of the system, They illustrate the different modules of the system as processes, entities, data stores, and data flow. These diagrams detail which entities and processes send and receive data, how different modules communicate

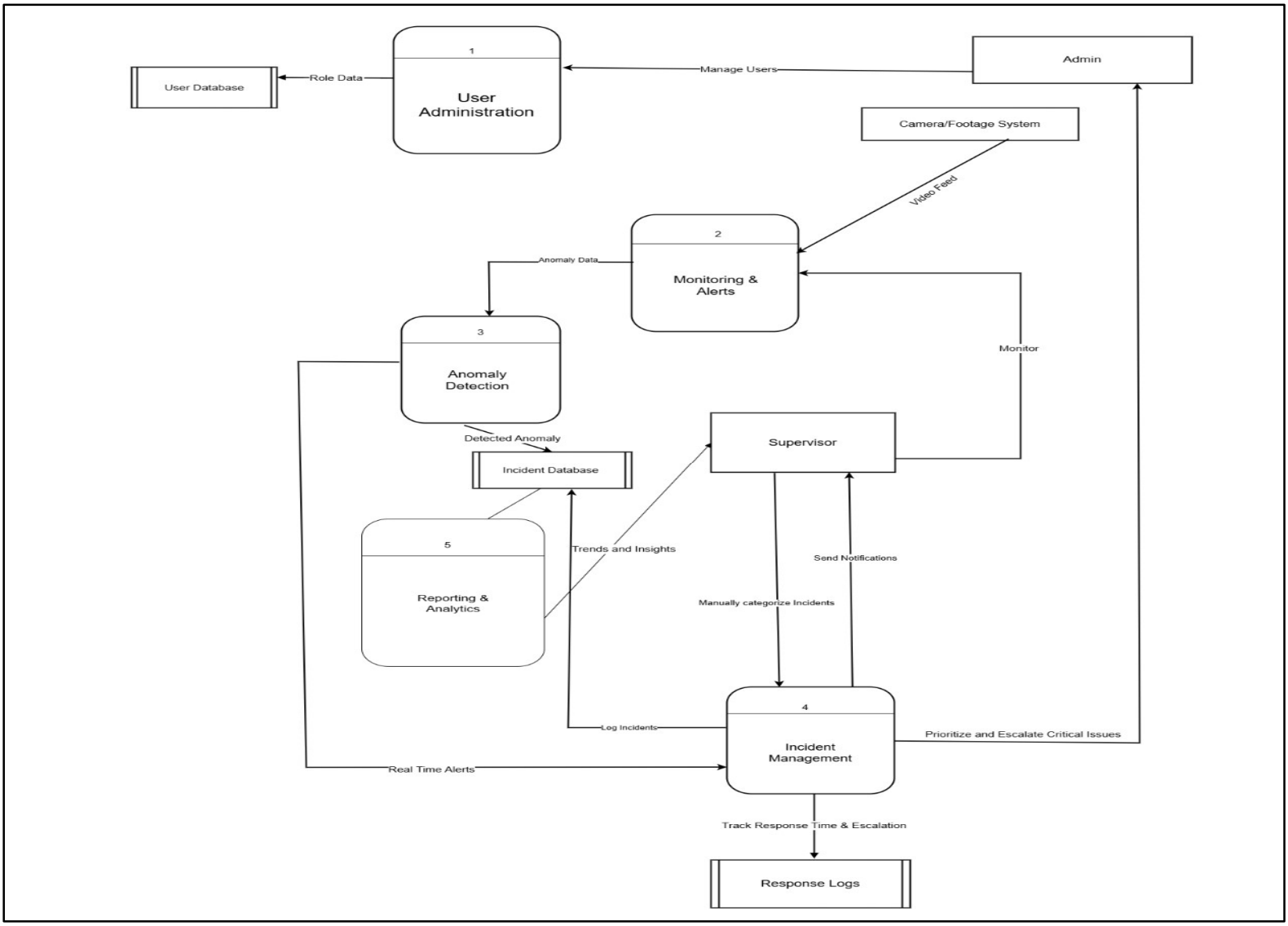


Figure 15 Level 1 DFD

30.3.11.3 *DFD level 2*

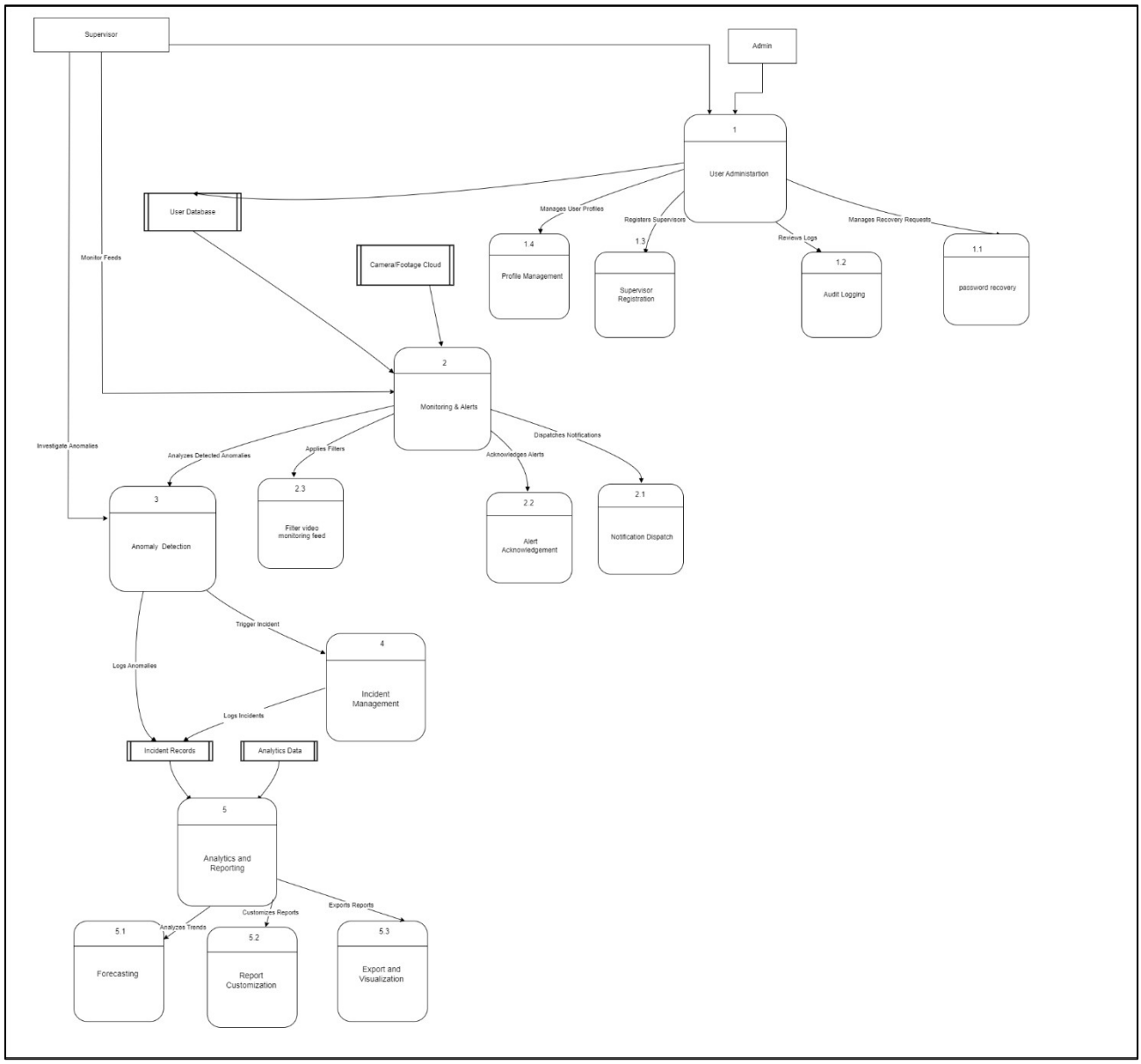


Figure 16 Level 2 DFD

30.3.11 State Transition Diagram

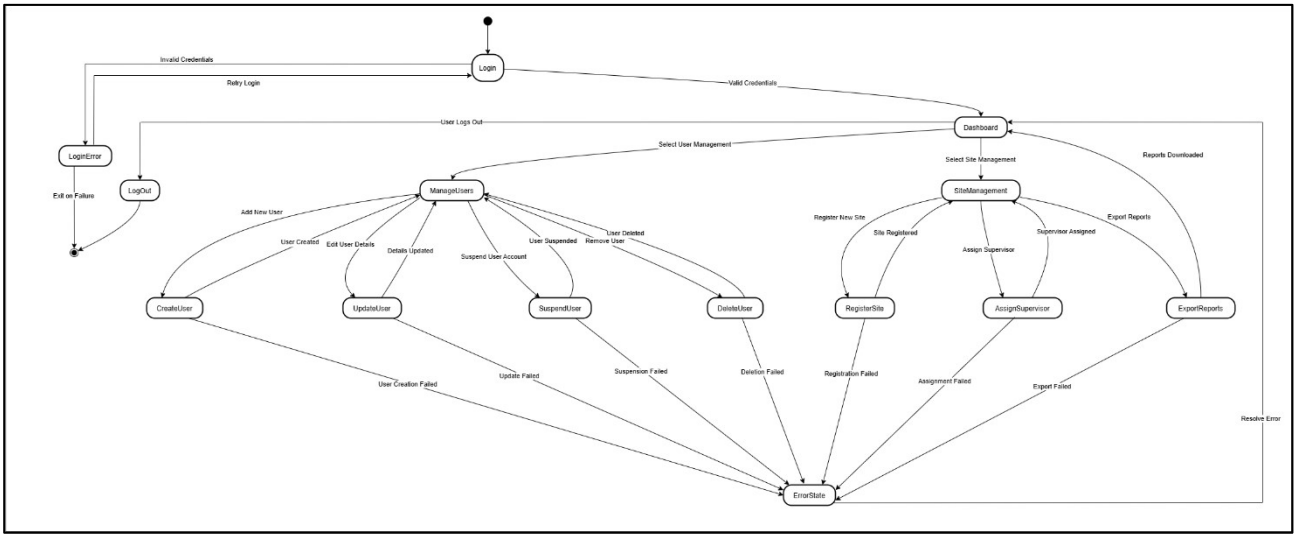


Figure 17: State Transition Diagram for Admin Taks

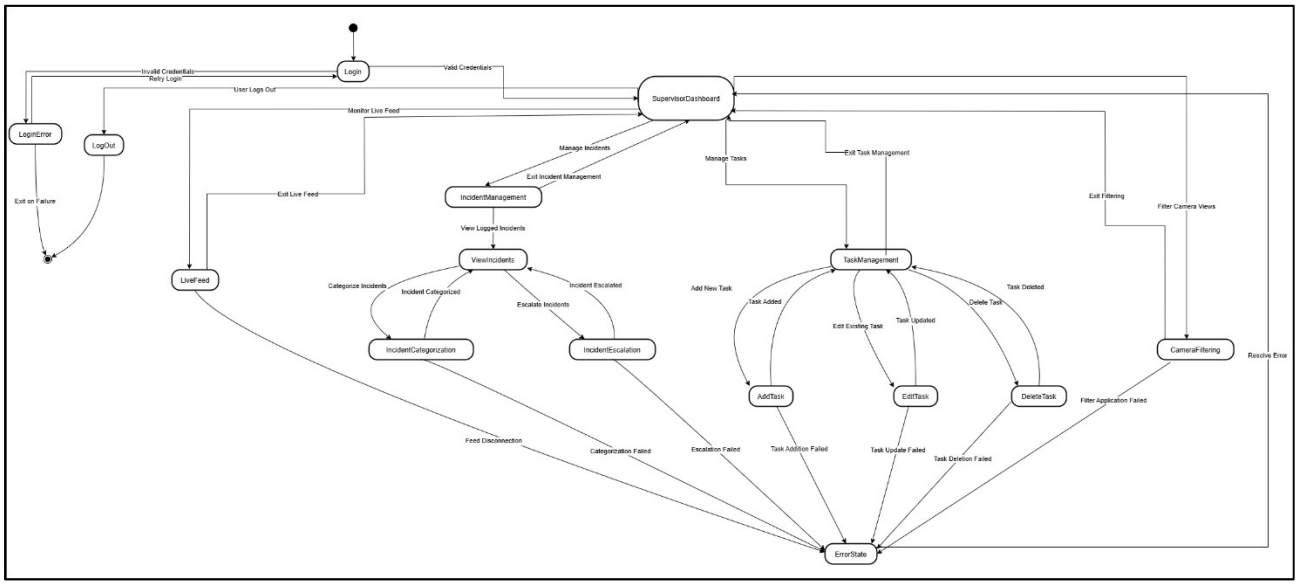


Figure 18: State Transition Diagram for Supervisor Taks

4.3 Data Design

The database used in our application is MongoDB. MongoDB is a cross-platform, document-based NoSQL database, that used BSON (Binary JSON) type documents. The database of this system is being deployed on MongoDB Atlas, a cloud-based service to deploy and configure database. For offline data storage.

4.3.1 Data Dictionary

The following schema defines the fields of the Supervisor. It stores information about all Supervisor, including SiteAssigned to Supervisor.

Supervisor Schema

```
{
  “_id”: “ObjectId”,
  “name”: “string”,
  “email”: “string”,
  “password”: “string”,
  “phone”: “string”,
  “alternateContact”: “string”,
  “alternateEmail”: “string”,
  “siteAssigned”: [“ObjectId”],
  “status”: “string”,
  “createdAt”: “date”,
  “updatedAt”: “date”
}
```

Table 194: Data Dictionary for Supervisor

Field	Data Type	Description
_id	ObjectId	Unique identifier for the Supervisor
name	string	Full name of the Supervisor
email	string	Email address of the Supervisor
password	string	Encrypted password of the Supervisor
phone	string	Contact number of the Supervisor
alternateContact	string	Alternate contact number of the Supervisor
alternateEmail	string	Alternate email address for the Supervisor
siteAssigned	Array of Object	List of site IDs assigned to the Supervisor
status	string	Status of the Supervisor (active, inactive, suspended)

createdAt	date	Date and time when the Supervisor was created
updatedAt	date	Date and time when the Supervisor was last updated

The following schema defines the fields to Register A site . It stores information about Site Registered by Admin

Site Schema

```
{
 “_id”: “ObjectId”,
 “SiteID”: “string”,
 “SiteName”: “string”,
 “SiteAddresss”: “string”,
 “City”: “string”,
 “monitored”: “143refere”,
 “Active”: “143refere”,
 “Sensitivity”: “string”,
 “createdAt”: “date”,
 “updatedAt”: “date”
}
```

Table 195: Data Dictionary for Site

Field	Data Type	Description
_id	ObjectId	Unique identifier for the site
SiteID	string	Unique ID assigned to the site
SiteName	string	Name of the site
SiteAddresss	string	Physical address of the site
City	string	City where the site is located
monitored	boolean	Indicates whether the site is currently monitored
Active	boolean	Indicates whether the site is active
Sensitivity	string	Sensitivity level of the site (Low, Medium, High)
createdAt	date	Date and time when the site was created
updatedAt	date	Date and time when the site was last updated

Anomaly Detail Schema:

The following schema defines the fields of the Anoamly Detail of Each Site registered by the Admin. It stores information about all Anoamly parameters of a Site.

```
{
 “_id”: “ObjectId”,
 “siteId”: “ObjectId”,
 “detectionRequirements”: {
  “helmet”: “144refere”,
  “vest”: “144refere”,
  “max_persons”: “number”,
  “gloves”: “144refere”,
  “safetyBoots”: “144refere”,
  “faceShield”: “144refere”,
  “otherPPE”: “144refere”
 },
 “createdAt”: “date”,
 “updatedAt”: “date”
}
```

Table 196: Data Dictionary for Anomaly Detail

Field	Data Type	Description
_id	ObjectId	Unique identifier for anomaly details
siteId	ObjectId	Reference to the site where anomalies are detected
detectionRequirements	object	Requirements for detecting anomalies
detectionRequirements.helmet	boolean	Whether a helmet is required (default: true)
detectionRequirements.vest	boolean	Whether a vest is required (default: true)
detectionRequirements.max_persons	number	Maximum persons allowed at the site (default: 4)
detectionRequirements.gloves	boolean	Whether gloves are required (default: false)
detectionRequirements.safetyBoots	boolean	Whether safety boots are required (default: false)

detectionRequirements.faceShield	boolean	Whether a face shield is required (default: false)
detectionRequirements.otherPPE	boolean	Whether other PPE is required (default: false)
createdAt	date	Date and time when the anomaly details were created
updatedAt	date	Date and time when the anomaly details were updated

Detected Anomaly Schema :

The following schema defines the Schema of the detected Anoamlies . It stores information at which Site anomaly occurred , time, date , status of the Anmaoly reported

```
{
 “_id”: “ObjectId”,
 “siteId”: “ObjectId”,
 “description”: “string”,
 “detectedAt”: “date”,
 “type”: “string”,
 “severity”: “string”,
 “status”: “string”,
 “images”: [“string”]
}
```

Table 197: Detected Anomaly Schema

Field	Data Type	Description
_id	ObjectId	Unique identifier for the anomaly
siteId	ObjectId	ID of the site where the anomaly was detected
description	string	Description of the anomaly
detectedAt	date	Date and time when the anomaly was detected
type	string	Type of anomaly (e.g., safety violation, hazard)
severity	string	Severity level of the anomaly (e.g., high, medium)
status	string	Current status of the anomaly (e.g., open, resolved)

images	Array of strings	List of image paths associated with the anomaly

Incident Schema:

The following schema handles data to keep track of Incident reported. It stores information about Anomaly after which this incident was created , it keeps response time .

```
{
 “_id”: “ObjectId”,
 “anomalyId”: “ObjectId”,
 “responseTime”: “number”,
 “responseDetails”: “string”,
 “status”: “string”,
 “resolvedAt”: “date”
}
```

Table 198: Data Dictionary for Incident Schema

Field	Data Type	Description
_id	ObjectId	Unique identifier for the incident
anomalyId	ObjectId	ID of the associated anomaly
responseTime	number	Time taken to respond to the incident (in minutes)
responseDetails	string	Details of the response
status	string	Current status of the incident (e.g., open, closed)
resolvedAt	date	Date and time when the incident was resolved

Notepad Schema:

Handles data to keep track of Notes created by Supervisor.

```
{
 “_id”: “ObjectId”,
 “supervisorId”: “ObjectId”,
 “title”: “string”,
 “content”: “string”,
 “priority”: “string”,
 “dueDate”: “date”,
 “isImportant”: “146refere”,
 “createdAt”: “date”,
```

“updatedAt”: “date”
}

Table 199: Data Dictionary for Notepad

Field	Data Type	Description
_id	ObjectId	Unique identifier for the note
supervisorId	ObjectId	Reference to the supervisor who created the note
title	string	Title of the note
content	string	Content or description of the note
priority	string	Priority level of the note (High, Medium, Low)
dueDate	date	Optional due date for the note
isImportant	boolean	Indicates whether the note is marked as important (default: false)
createdAt	date	Date and time when the note was created
updatedAt	date	Date and time when the note was last updated

Report Schema:

{
“_id”: “ObjectId”,
“generatedBy”: “ObjectId”,
“Siteid”: “ObjectId”,
“preport type”: “string”,
“generated At”: “date”,
“file”: “string”
“createdAt”: “date”,
“updatedAt”: “date”
}

Table 200: Data Dictionary for Report Schema

Field	Data Type	Description
_id	ObjectId	Unique identifier for the report
generatedBy	ObjectId	ID of the user who generated the report
siteId	ObjectId	ID of the site the report is about
reportType	string	Type of report (e.g., safety, anomaly)

generatedAt	date	Date and time when the report was generated
filePath	string	File path where the report is stored

5 Implementation

5.1 Algorithm

Algorithm 1: UserAuthentication

Input: username, password

Output: authenticationStatus (True/False)

```
Begin
hashedPassword ← hash(password)  // Hash the input password for security
userRecord ← fetchUser(username) // Fetch user data from the database
If userRecord is NULL then
    Print "User not found"
    authenticationStatus ← False
Else
    If userRecord.password == hashedPassword then
        Print "Authentication Successful"
        authenticationStatus ← True
    Else
        Print "Incorrect Password"
        authenticationStatus ← False
    End If
End If
Log authentication attempt with timestamp and status
Return authenticationStatus
End
```

Algorithm 2: DetectAnomalies

Input: videoFrames[], anomalyDetectionModel, threshold

Output: anomalies[]

```
Begin
anomalies ← []  // Initialize an empty list for anomalies
For each frame in videoFrames do
    processedFrame ← preprocessFrame(frame) // Resize, normalize the image
    anomalyScore ← anomalyDetectionModel.predict(processedFrame) // Get model output
    If anomalyScore > threshold then
        Print "Anomaly Detected in Frame:", frame.id
        anomalies.append({
            "frameID": frame.id,
            "anomalyScore": anomalyScore,
            "timestamp": getCurrentTimestamp()
        })
    Else
        Print "No Anomaly in Frame:", frame.id
    End If
End For
```

```
17:  Return anomalies
18:  End
```

Algorithm 3: Real-Time Monitoring

Input: videoStream, configSettings, anomalyDetectionModel

Output: monitoredStream

```
1:  Begin
2:  monitoredStream ← []  // Initialize empty list for processed video
3:  While (videoStream.isActive()) do
4:      frame ← captureFrame(videoStream) // Get the next frame
5:      If configSettings.enableAnomalyDetection == True then
6:          anomalies ← DetectAnomalies([frame], anomalyDetectionModel, configSettings.threshold)
7:          If anomalies is not empty then
8:              Print “Highlighting Anomalies in Frame:”, frame.id
9:              frame ← highlightAnomalies(frame, anomalies) // Add highlights to frame
10:         Else
11:             Print “No Anomalies Detected in Frame:”, frame.id
12:         End If
13:     End If
14:     monitoredStream.append(frame)
15:     Display frame on monitoringInterface
16:  End While
17:  Log monitoring session with start and end timestamps
18:  Return monitoredStream
19:  End
```

Algorithm 4: GenerateAnalyticsReport

Input: incidentLogs[], timeRange, userPreferences (User 149references refer to customizable settings or criteria provided by the user that influence how the data analytics report is generated, formatted, or filtered)

Output: report

```
1:  Begin
2:  filteredLogs ← []
3:  For each log in incidentLogs do
4:      If log.timestamp is within timeRange then
5:          filteredLogs.append(log)
6:      End If
7:  End For
8:  analyticsData ← {
9:      “totalIncidents”: count(filteredLogs),
10:     “incidentTypes”: analyzeIncidentTypes(filteredLogs),
11:     “incidentFrequency”: calculateFrequency(filteredLogs, userPreferences.interval),
12:     “anomaliesDetected”: summarizeAnomalies(filteredLogs)
13: }
14: report ← formatReport(analyticsData, userPreferences.format) // PDF, Excel, etc.
15: Print “Analytics Report Generated”
16: Log report generation event
17: Return report
18: End
```

Algorithm 5: ManageIncident

Input: anomalyDetails, userInput, notificationSettings

Output: incidentReport, notificationStatus

```
Begin
incidentReport ← {
    "anomalyID": anomalyDetails.id,
    "description": anomalyDetails.description,
    "timestamp": getCurrentTimestamp(),
    "status": "Reported"
}
saveIncidentToDatabase(incidentReport) // Store the report in the database
If userInput.requiresNotification == True then
    recipientList ← getNotificationRecipients(notificationSettings)
    notificationStatus ← sendNotification(recipientList, anomalyDetails)
    Print "Notification Sent to:", recipientList
Else
    notificationStatus ← "No Notification Sent"
    Print "Notification Disabled"
End If
Log incident and notification details
Return incidentReport, notificationStatus
End
```

Algorithm 6: Update Site Anomaly
Input: selectedAnomalies, siteID, anomalyData, maxPersons
Output: filteredAnomalies

```
Begin
Retrieve all anomalyData for siteID
Filter anomalies based on selectedAnomalies checkboxes
If maxPersons is set then
Filter anomalies where detectedPersons ≤ maxPersons
End If
If filteredAnomalies is empty then
Print "No anomalies detected for selected criteria"
Else
Display filteredAnomalies
End If
Log anomaly filter action
Return filteredAnomalies
End
```

Algorithm 7: Alert Prioritization using K-Means Clustering
Input: alertData (list of alerts with severity score, PPE compliance, incident type, etc.)
k = 3 (number of clusters for Low, Medium, and High priority)
Output: classifiedAlerts (alerts categorized into Low, Medium, or High priority)

```
Begin
Normalize alertData to bring all features to a common scale
Initialize k-means clustering with k = 3
Select 3 random data points as initial cluster centroids
Repeat until convergence or max iterations reached:
a. Assign each alert to the nearest centroid using Euclidean distance
b. Update cluster centroids by averaging assigned alerts
End Repeat
Label clusters as Low, Medium, or High priority based on severity distribution
For each new alert:
a. Compute distance to cluster centroids
b. Assign alert to the closest priority category
Generate notifications based on priority level:
```

14: a. If High priority, send immediate alert to supervisor
15: b. If Medium priority, log and send periodic reminder
16: c. If Low priority, store for future analysis
17: Log alert classification action
18: Return classifiedAlerts
19: End

Algorithm 8: Chatbot
Input: userQuery (natural language input)
Output: chatbotResponse (generated response based on retrieved information)

1: Begin
2: Load pre-trained embeddings for company documents and policies
3: Initialize LLM (Gemini) for response generation
4: Receive userQuery as input
5: Preprocess userQuery:
    a. Convert to lowercase
    b. Remove stop words and special characters
    c. Tokenize and vectorize
6: Retrieve relevant document chunks:
    a. Compute similarity between userQuery and stored embeddings
    b. Select top-ranked document chunks
7: Pass the selected chunks and userQuery to LLM for response generation
8: Generate chatbotResponse based on LLM output
9: Display chatbotResponse to user
10: Log the conversation for future improvements
11: End

Algorithm 9: Alert Prioritization using K-Means Clustering
Input: alertData (list of alerts with severity score, PPE compliance, incident type, etc.)
k = 3 (number of clusters for Low, Medium, and High priority)
Output: classifiedAlerts (alerts categorized into Low, Medium, or High priority)

1: Begin
2: Normalize alertData to bring all features to a common scale
3: Initialize k-means clustering with k = 3
4: Select 3 random data points as initial cluster centroids
5: Repeat until convergence or max iterations reached:
6: a. Assign each alert to the nearest centroid using Euclidean distance
7: b. Update cluster centroids by averaging assigned alerts
8: End Repeat
9: Label clusters as Low, Medium, or High priority based on severity distribution
10: For each new alert:
11: a. Compute distance to cluster centroids
12: b. Assign alert to the closest priority category
13: Generate notifications based on priority level:
14: a. If High priority, send immediate alert to supervisor

15: b. If Medium priority, log and send periodic reminder
16: c. If Low priority, store for future analysis
17: Log alert classification action
18: Return classifiedAlerts
19: End

Algorithm 10: Safety Risk Forecasting and Recommendation Generation
Input:

- siteData: List of site records from the database
- alerts: List of alerts related to each site
- weatherConditions: Mock weather forecast per city
- riskWeights: Predefined risk weights for PPE, Fall, Weather, and Sensitivity

Output:

- siteAnalysisResults: Risk breakdown and safety recommendations for each site

1: Begin
2: Define risk factors for:
a. PPE types (e.g., Hardhat, SafetyVest, Gloves, Boots)
b. Fall incidents
c. Weather conditions (e.g., Storm, Rain, Hot)
d. Sensitivity levels (e.g., High, Medium, Low)
3: For each site in siteData:
4: a. Retrieve siteId, siteName, city, and sensitivity level
5: b. Fetch all related alerts from the Alerts collection using siteId
6: c. Get weather forecast for the site's city
i. Randomly select weather condition (Storm, Rain, Hot, Clear)
ii. Assign weather-based risk factor
7: d. Initialize risk scores:
i. ppe_risk = 0
ii. fall_risk = 0
iii. weather_risk = weather's risk factor
iv. sensitivity_risk = lookup based on sensitivity level
8: e. For each alert:
i. Convert alert description to lowercase
ii. For each PPE item:
- If mentioned in description:
· Extract missing count
· Add (count × risk weight) to ppe_risk
iii. If "fall" in description:
- Extract fall count and add (count × fall risk) to fall_risk
9: f. Calculate total_risk as a weighted sum of:
i. 40% ppe_risk
ii. 30% fall_risk
iii. 20% weather_risk
iv. 10% sensitivity_risk

10: g. Set default risk_level = "Low"
11: If total_risk > 0.7, set risk_level = "High"
12: Else if total_risk > 0.4, set risk_level = "Medium"
13: h. Generate recommendations:
i. Immediate actions:
- If fall_risk > 0.5 → "Conduct immediate fall hazard inspection"
- If ppe_risk > 0.6 → "Perform PPE compliance audit today"
ii. Weather-based:
- If weather_risk > 0.7 → "Implement severe weather protocol"
- Else if weather_risk > 0.4 → "Adjust work schedule"
iii. Training needs:
- If ppe_risk > 0.3 → "PPE compliance training refresher"
- If fall_risk > 0.2 → "Fall protection training"
iv. Long-term:
- If alert count > 10 → "Review and update safety protocols"
- If total_risk > 0.5 → "Install additional safety monitoring systems"
14: i. Store analysis for site:
- siteId, siteName, risk_breakdown, weather, recommendations, analysis_date

15: End For

16: Return siteAnalysisResults
17: End

6 Algorithm 11: AI-Powered Site Safety Report Generation via LLM

Input:

- prediction_col: Safety predictions per site from MongoDB
- site_col: Site metadata from MongoDB
- anomaly_col: List of anomalies per site
- incident_col: Incident details linked to anomalies
- gemini_model: Configured Gemini AI model

Output:

- Cleaned safety reports for each site in structured markdown format

1: Begin
2: Connect to MongoDB using connection URI and select test database
a. Initialize collections:
• safety_analytics, sites, Alerts, Incident_info
3: Define function generate_report_data():
a. Initialize empty list final_response
4: Fetch all prediction documents from safety_analytics
5: For each prediction in predictions:
a. Extract SiteID and convert to ObjectId
b. Fetch corresponding site document from sites collection
c. If site not found or invalid, skip to next prediction
6: Retrieve all anomalies from Alerts where siteId matches
a. Collect all anomaly _ids
7: Initialize incident statistics:
i. Resolved = 0, Unresolved = 0, In Progress = 0
8: If anomalies exist:
a. Use aggregation pipeline on Incident_info to count incidents by status
b. For each incident group:
• Update incident_stats with count of each status
9: Construct report_entry:
a. Include:

• SiteID, SiteName, City, SiteAddress
• PredictedRiskLevel, WeatherCondition, WeatherSeverity
• TotalAnomalies30Days, RecentAnomalies7Days
• AnomalyBreakdown, DailyAnomalyPattern, Recommendations
• AnomalyCount, IncidentStats
10: Append report_entry to final_response
11: End For
12: Return final_response
13: Define function reportusingLLM():
a. Call generate_report_data() → get res
b. Construct prompt string using markdown with strict formatting rules:
• Section headers, bullet points, indentation, emojis, and priorities
c. Pass prompt to gemini_model.generate_content(prompt)
d. Print raw AI response
e. Clean the response by calling clean_report(response.text)
14: Define function clean_report(text):
a. Standardize 2-space indentation for nested bullets
b. Remove extra blank lines
c. Ensure exactly one newline between different site reports
15: Return cleaned report
16: End

6.1 External APIs/SDKs

Table 201 Details of APIs used in the project

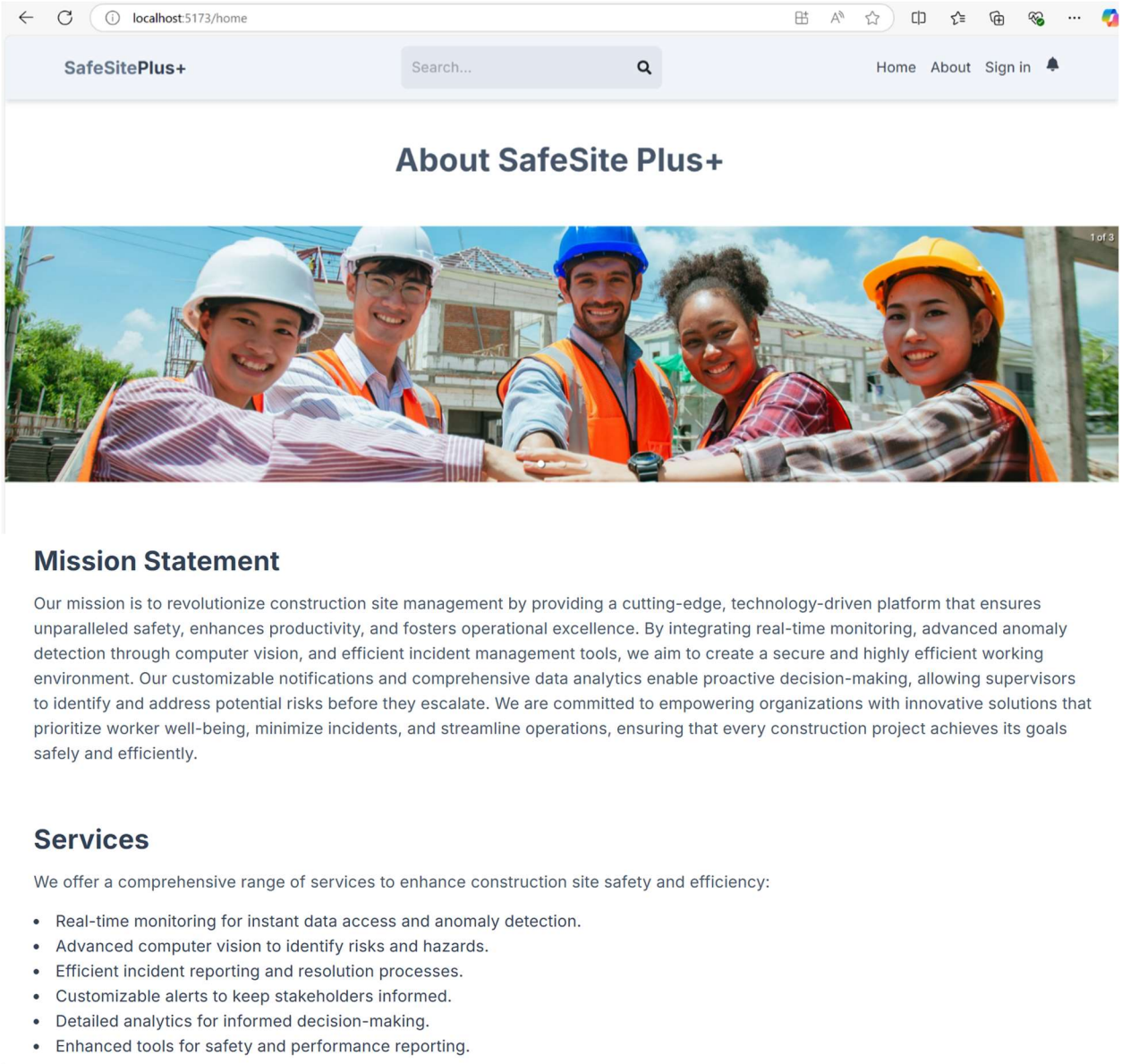
Name of API and version	**Description of API**	**Purpose of usage**	**List down the API endpoint/function/class in which it is used**
OpenWeatherMap API (v2.5)	Weather data (current, forecast, historical)	Track weather conditions at the construction site and store data for analysis	https://api.openweathermap.org/data/2.5/weather, https://api.openweathermap.org/data/2.5/forecast
YOLOv8 SDK (by Ultralytics)	SDK for computer vision tasks, including object detection and segmentation.	Detect safety violations, PPE compliance, and hazardous situations in construction site footage.	YOLO.detect
Gemini (Google Generative AI, gemini-pro)	Google's large language model capable of generating text-based responses, summaries, and insights using prompts.	To generate structured and human-readable safety reports based on prediction and anomaly data using markdown formatting.	- reportusingLLM()

6.2 User Interface.

Following are few examples of User Interfaces:

6.2.1 Landing Page

The homepage introducing SafeSitePlus, offering navigation to other sections.



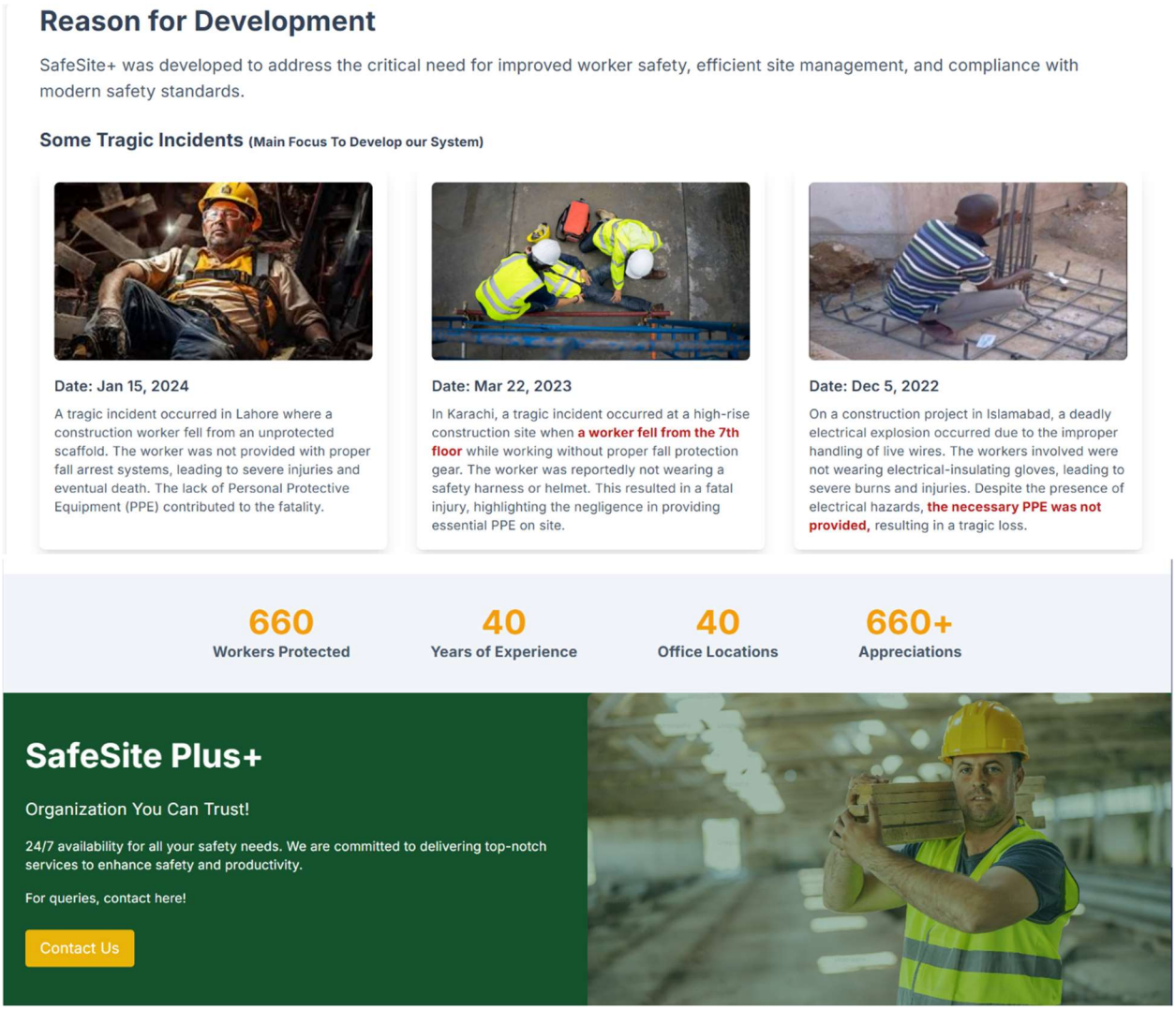


Figure 19 Landing Page

6.2.2 Login Page

A user authentication screen where users log in with their credentials.

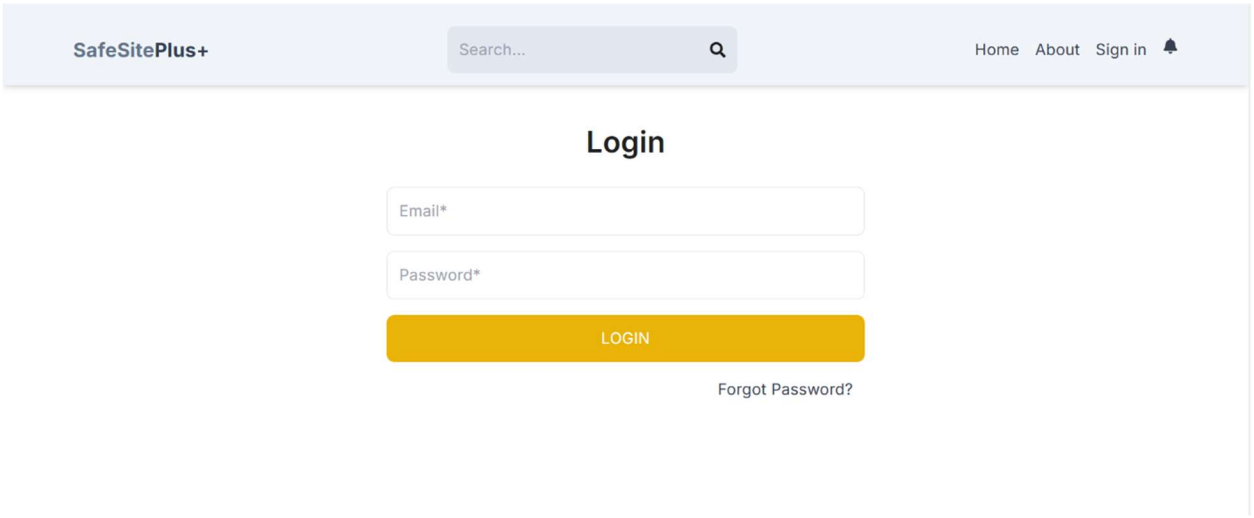


Figure 20: Login Page

6.2.3 Admin Dashboard

Admin Dashboard: The central hub for admins to manage users, view statistics, and monitor system activities.

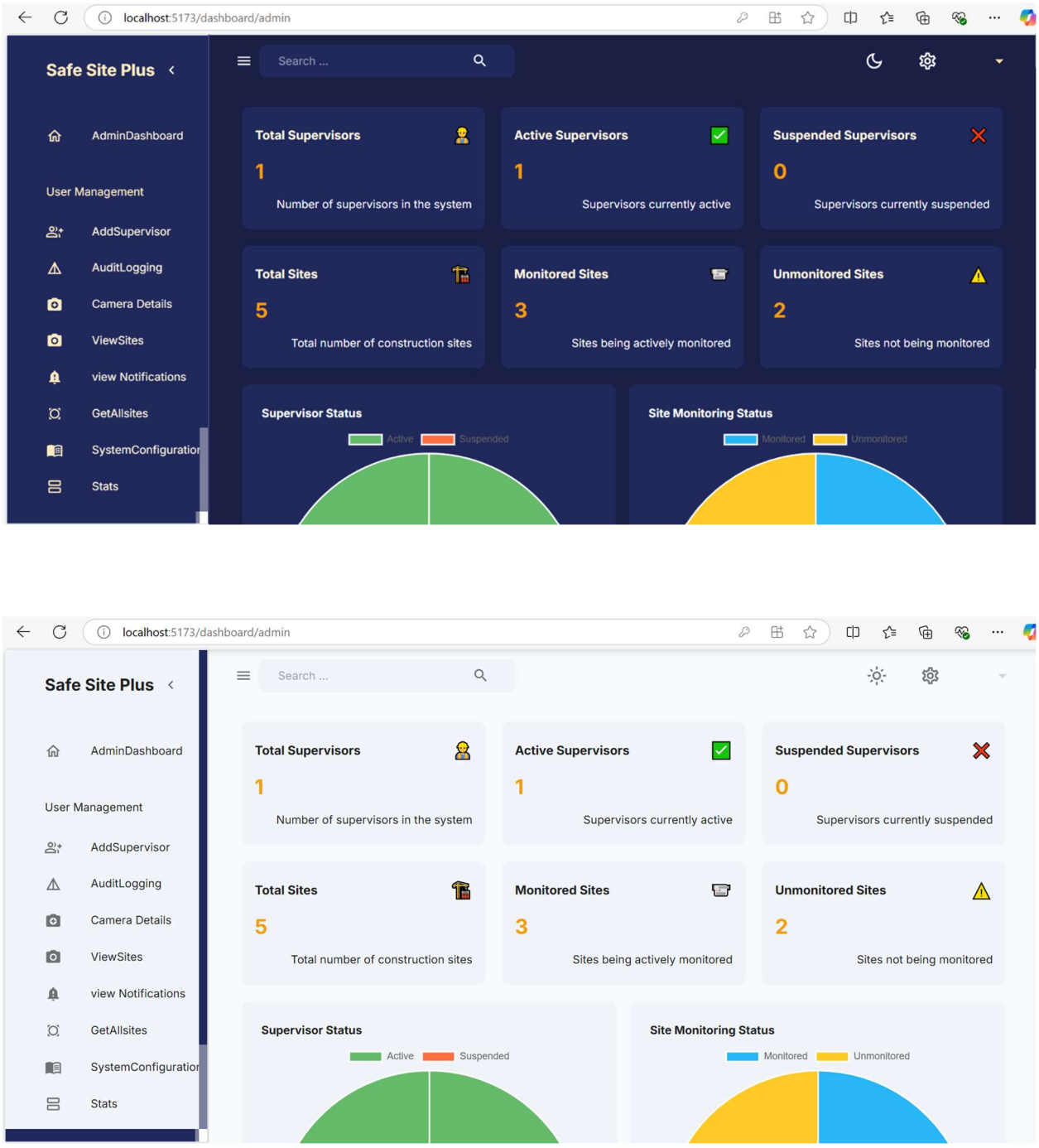
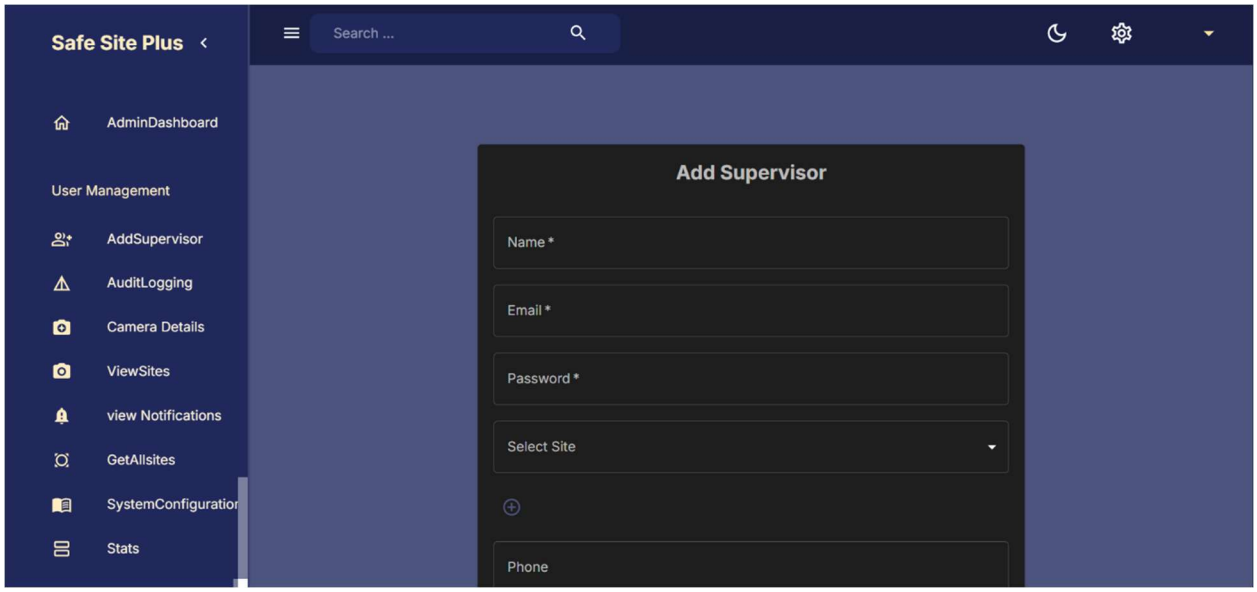


Figure 21: Admin Dashboard

30.3.11 Add Supervisor

A form for adding new supervisor details to the system.



S

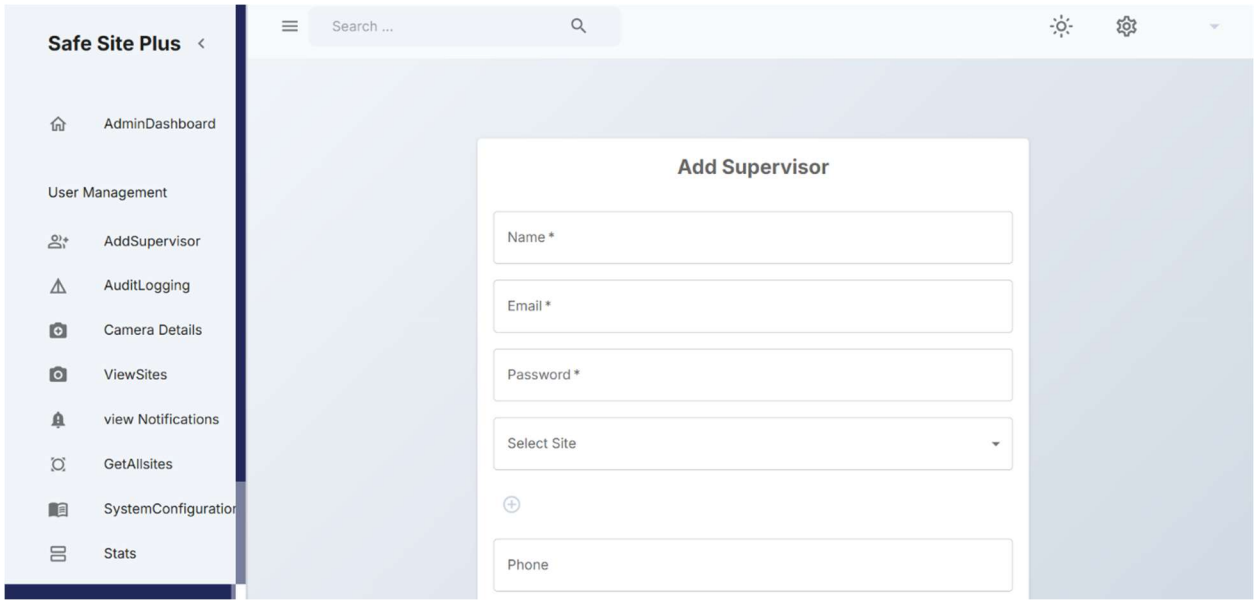


Figure 22: Form to Add Supervisor

30.3.11 Audit Logging

Audit Logging: A screen displaying logs of system activities for security and tracking purposes.

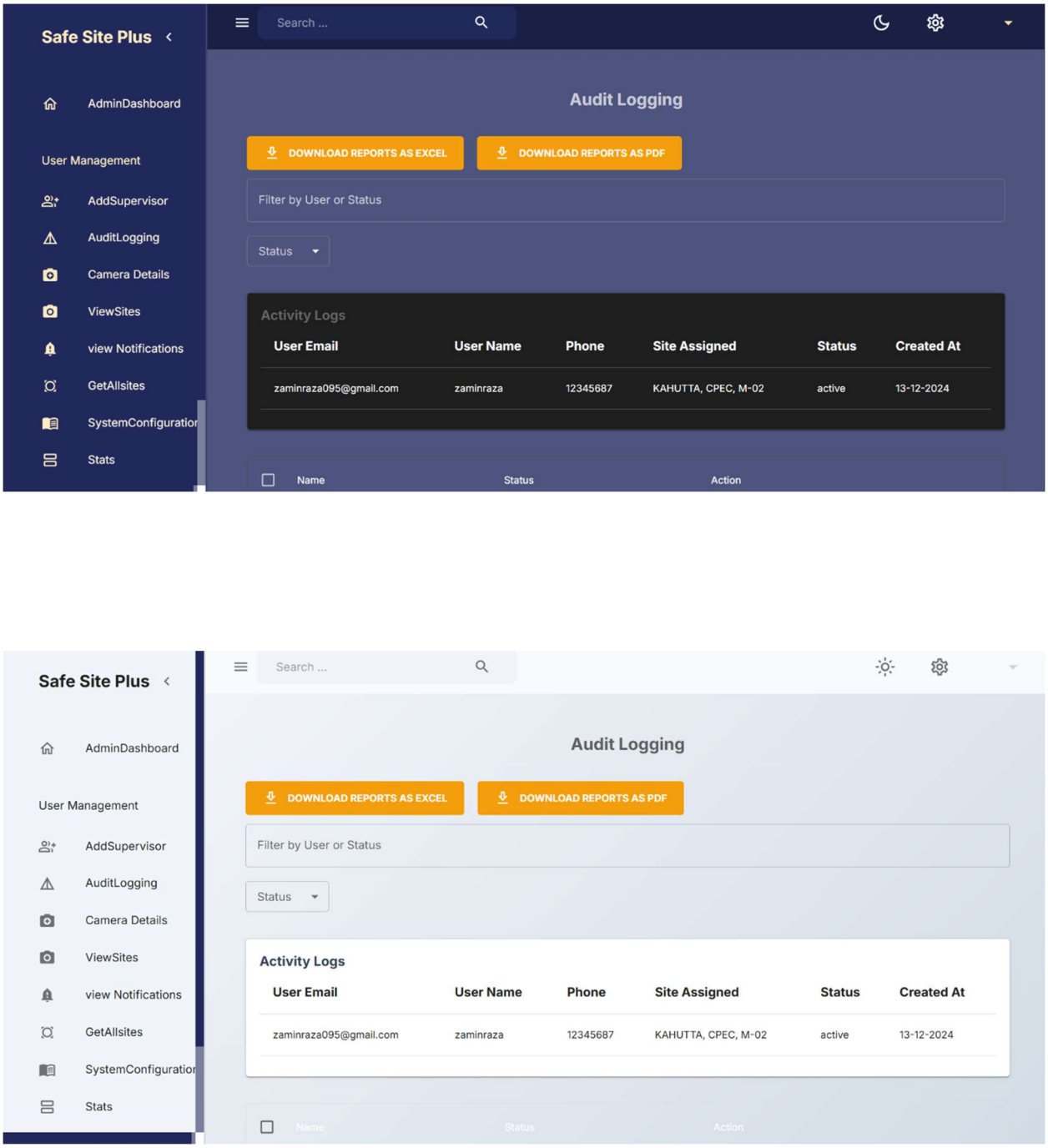


Figure 23 Audit Logging

5.3.6 Register a Site

A form where new sites can be registered with required details.

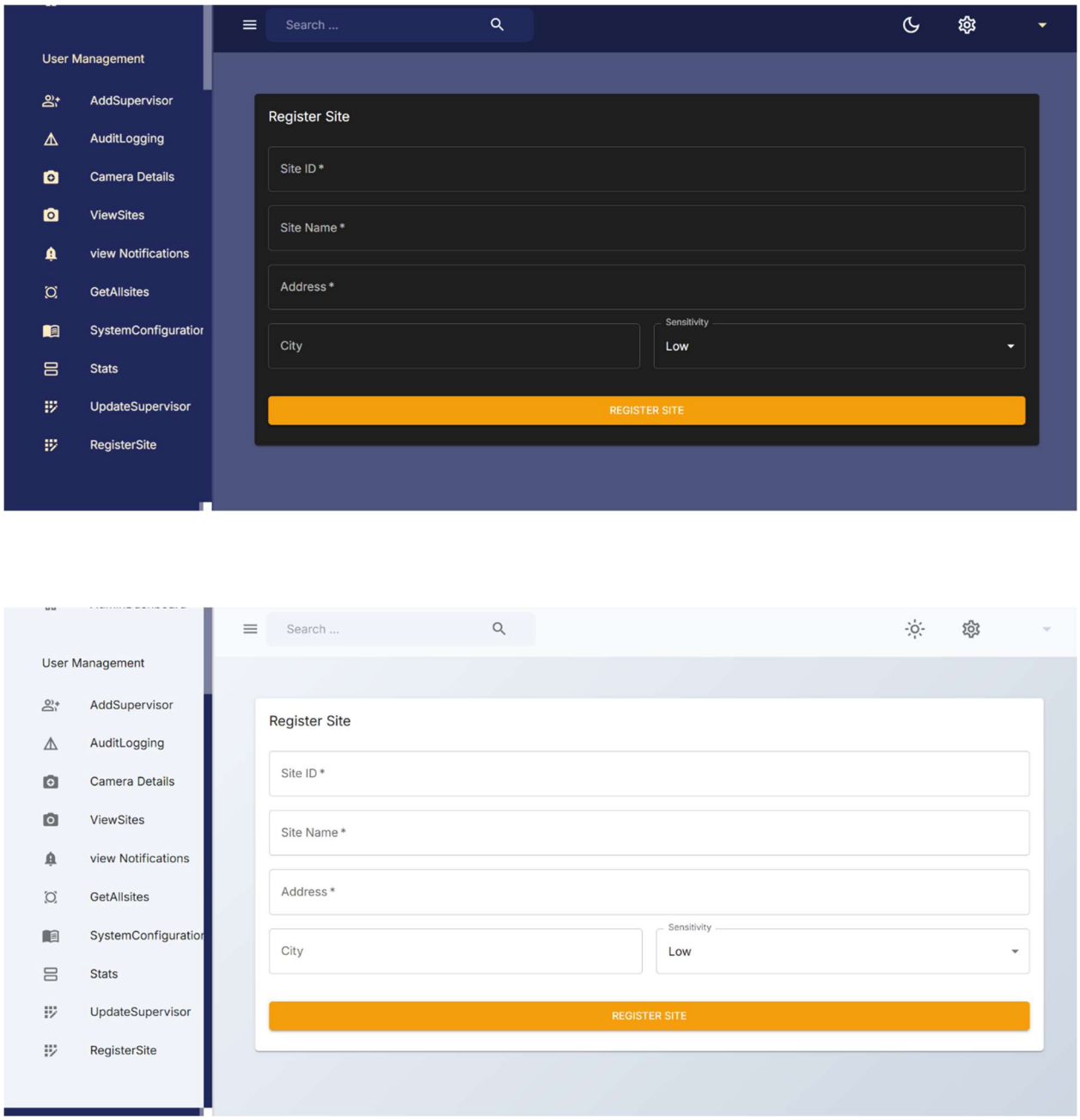


Figure 24 Register A Site

5.3.7 Stats Page

Displays statistical data and analytics of the User's and registered Sites and their Status

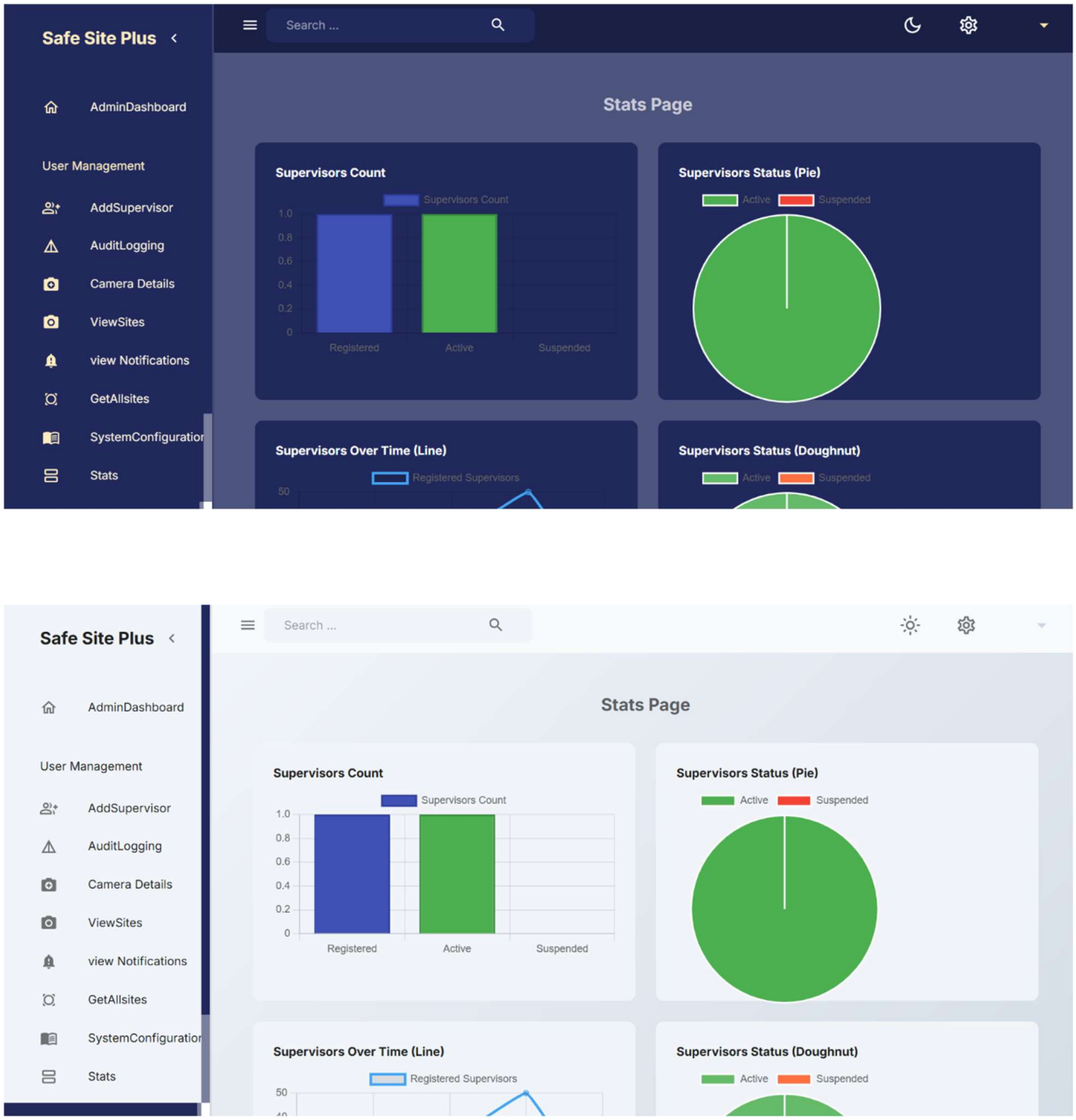


Figure 25 View Stats

5.3.8 Update Supervisor Credentials

Display to edit supervisor information.

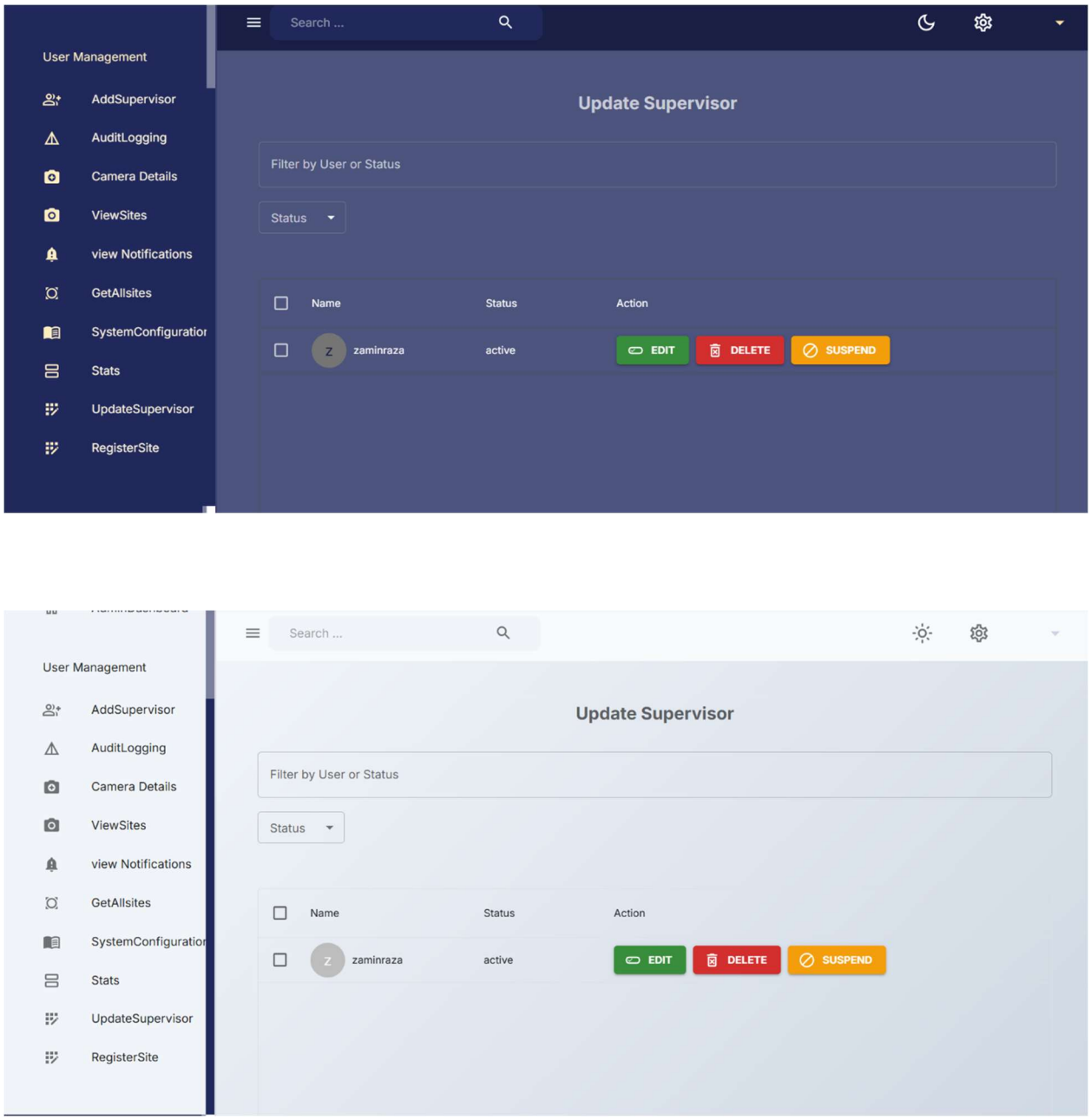


Figure 26 Update Supervisor Credentials

5.3.9 Supervisor Dashboard

The supervisor's main dashboard to view activities and monitor the site.

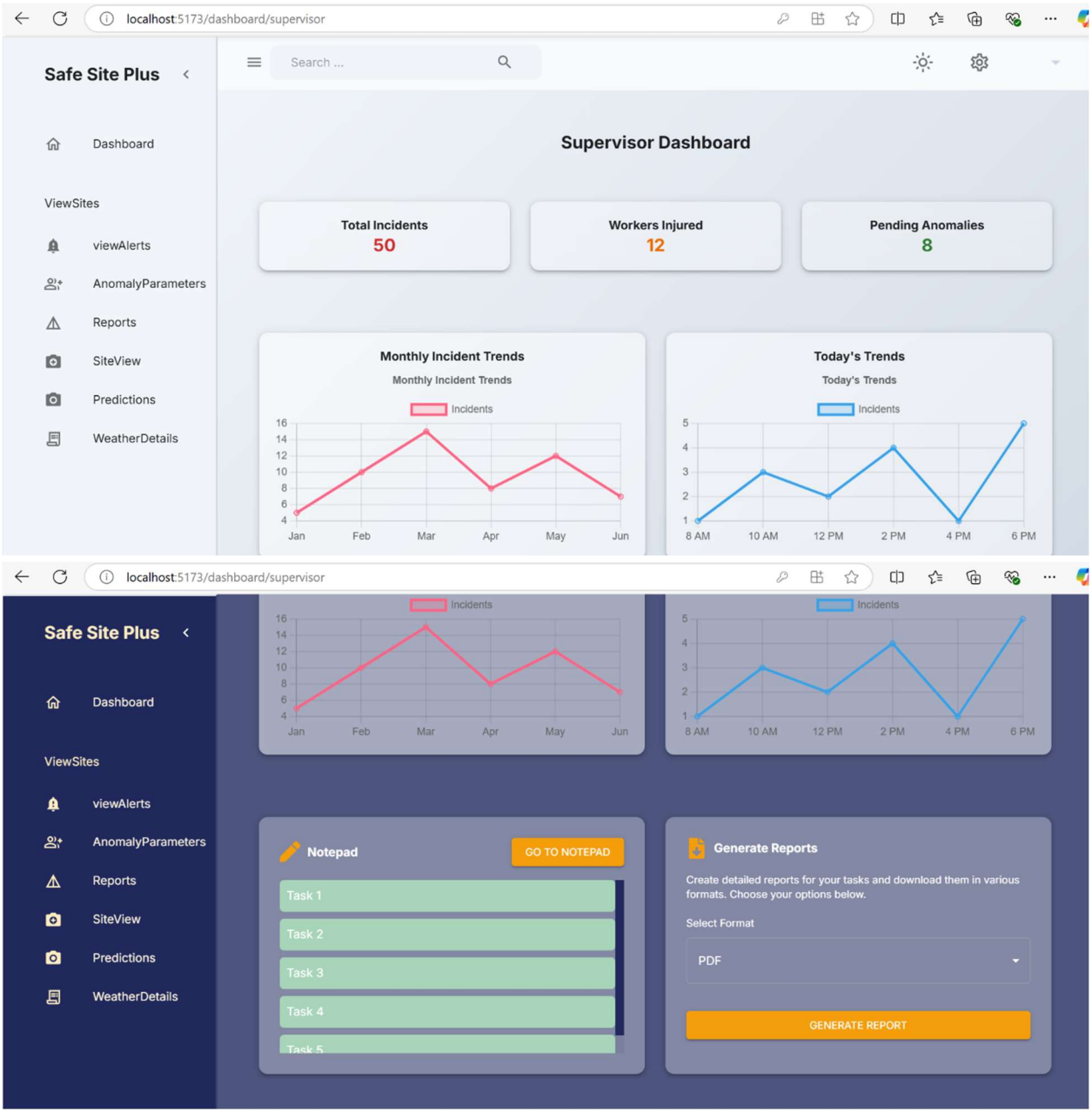


Figure 27 Dashboard Supervisor

5.3.10 Supervisor Anomaly Parameters

Screen where supervisors can customize or filter to detect specific anomalies at the site.

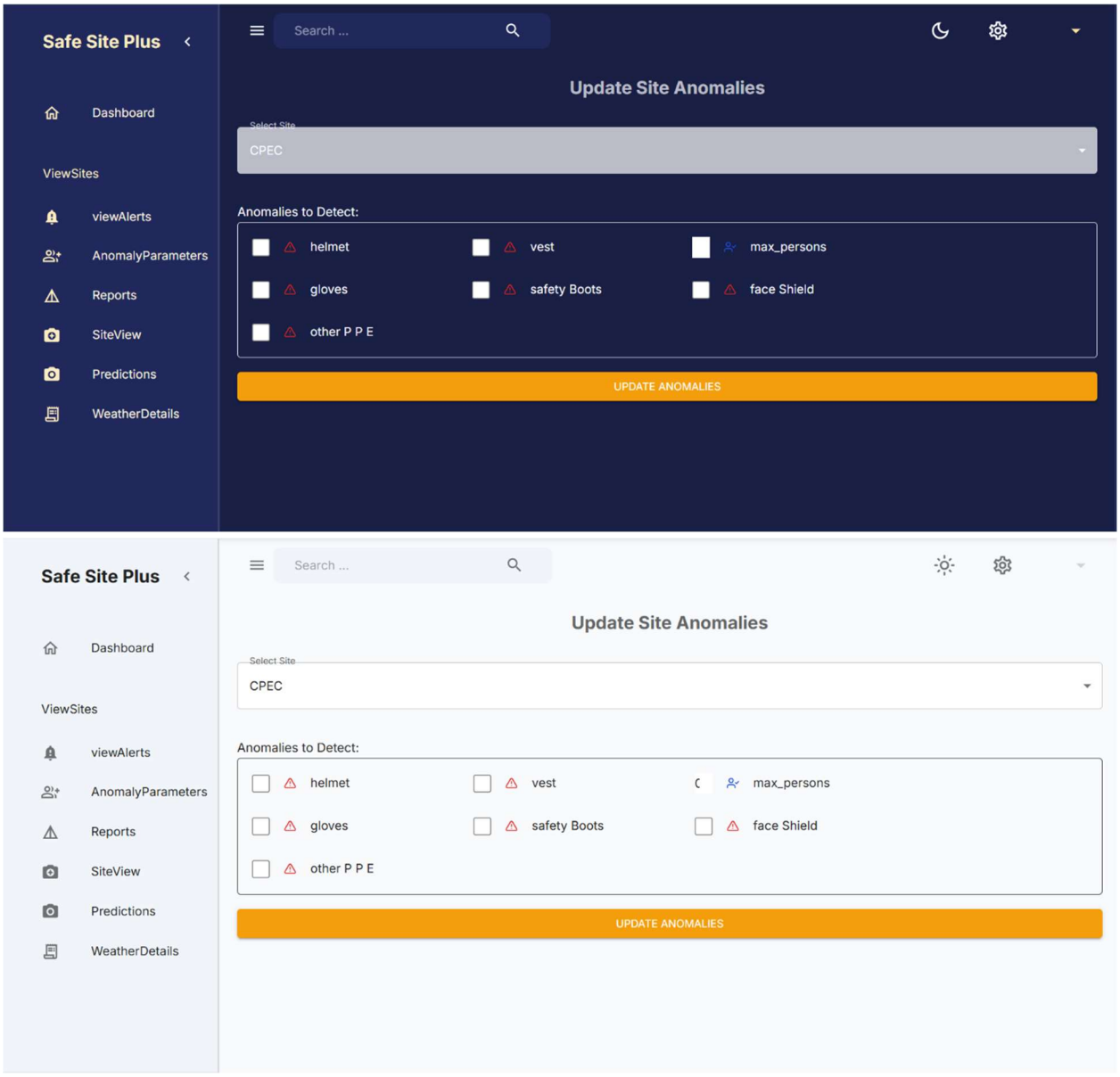


Figure 28 Select Anomaly parameters for Site

5.3.11 Supervisor Sites

A screen where supervisors monitor all the sites and detect anomalies at the site.

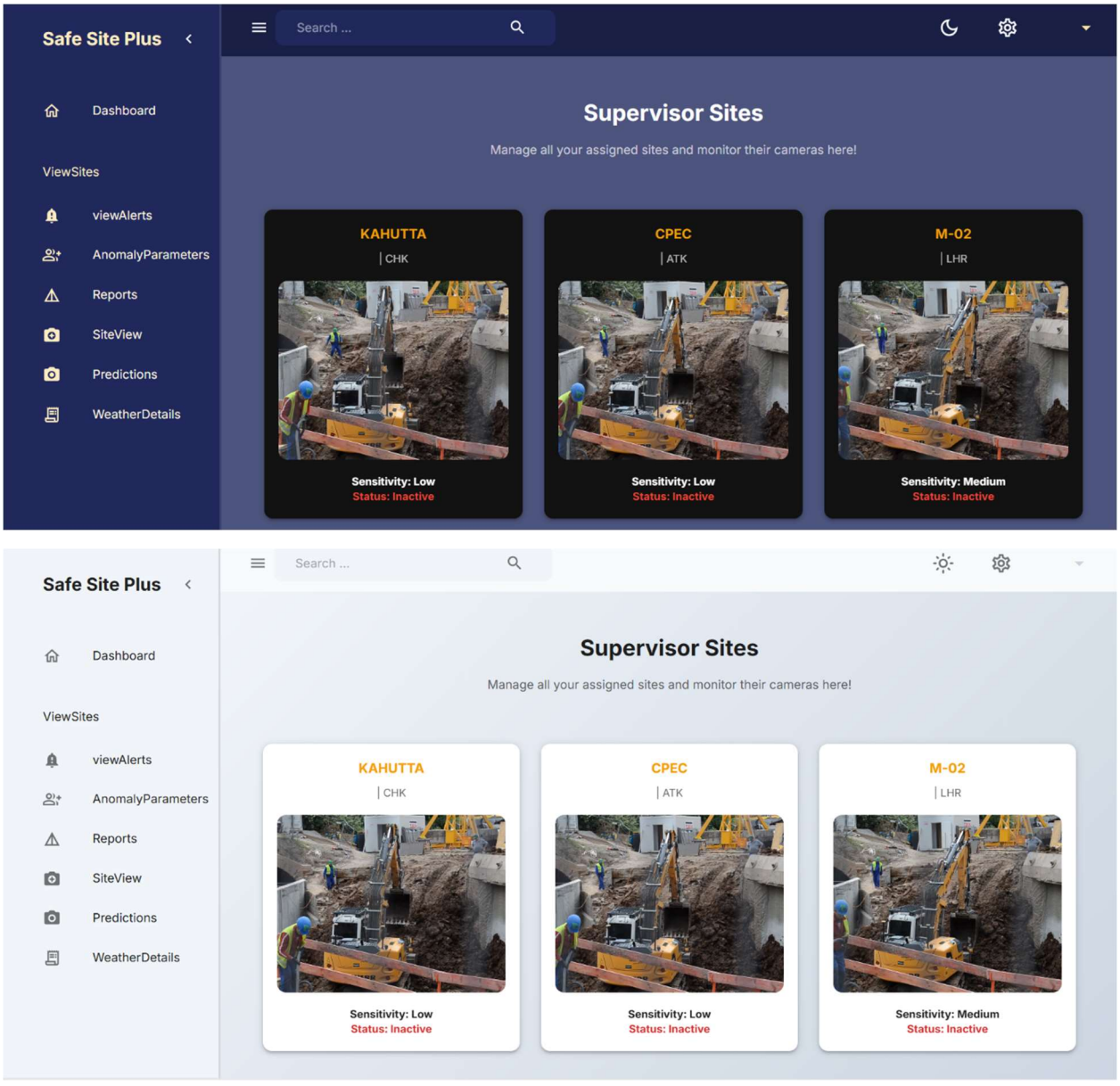


Figure 29 View Video Feeds of Sites

A page where supervisors can view and manage notifications related to site operations.

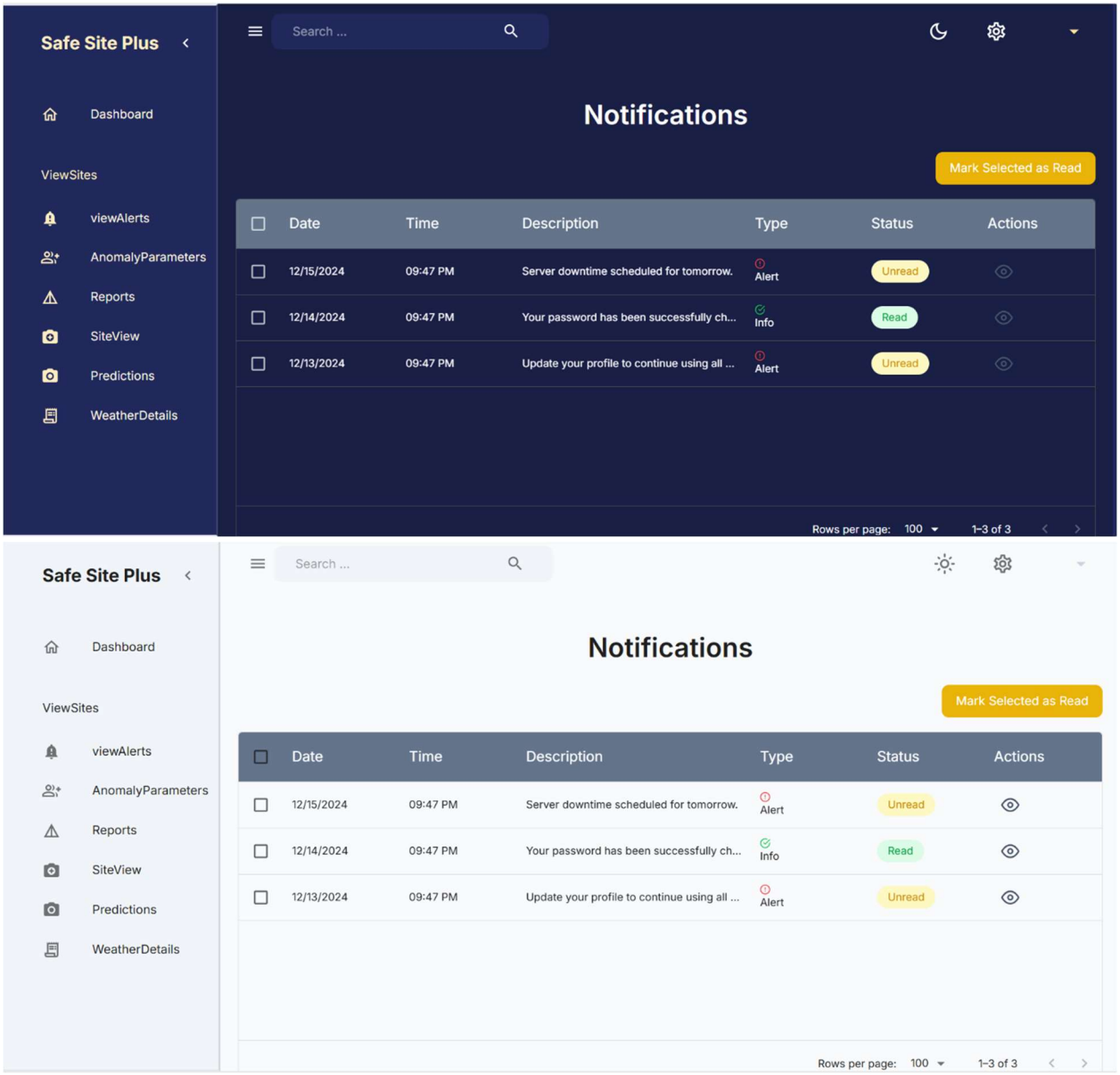


Figure 30 : View Notifications and Alerts

7 Testing and Evaluation

7.1 Unit Testing

It's a level of software testing where individual units of a software/component are tested. The purpose is to validate that each unit of the software performs as designed.

Unit Testing 1: Validate User Authentication
Testing Objective: To ensure the login form is working correctly with valid and invalid credentials/inputs.

No.	Test Case/Test Script	Expected Result	Result
1	Login with valid email and password	Successful login, redirect to dashboard	Pass
2	Login with invalid email format	Display error message: *Invalid email format*	Pass
3	Login with incorrect password	Display error message: *Incorrect password*	Pass
4	Login with unregistered email	Display error message: *User not found*	Pass
5	Attempt login with empty email or password fields	Display error message: *Fields cannot be empty*	Pass
6	Admin login with correct credentials	Redirect to admin dashboard	Pass
7	Supervisor login with correct credentials	Redirect to supervisor interface	Pass
8	Suspended account login attempt	Display error message: *Account is Suspended*	Pass
9	Login with expired session	Prompt for re-login	Pass

Unit Testing 2: Validate Password Recovery

No.	Test Case/Test Script	Expected Result	Result
1	Submit password recovery request with valid email	Send password recovery link to the registered email	Pass

2	Submit password recovery request with invalid email format	Display error message: *Invalid email format*	Pass
3	Submit request with an unregistered email	Display error message: *Email not found*	Pass
4	Submit request with empty email field	Display error message: *Email cannot be empty*	Pass
5	Click on password recovery link	Redirect to password reset page	Pass
7	Reset password with valid and matching passwords	Update password and redirect to login page	Pass
8	Use an expired password recovery link	Display error message: *Link expired, request a new one*	Pass

Unit Testing 3: Update Profile

No.	**Test Case/Test Script**	**Expected Result**	**Result**
1	Update profile with valid data	Successfully update profile and save changes	Pass
2	Attempt to update profile with empty required fields	Display error message: *Fields cannot be empty*	Pass
3	Update profile with invalid email format	Display error message: *Invalid email format*	Pass
4	Attempt to update profile with existing email	Display error message: *Email already in use*	Pass

Unit Testing 4: Supervisor Registration

No.	**Test Case/Test Script**	**Expected Result**	**Result**
1	Register supervisor with valid details	Successfully create supervisor account	Pass
2	Register with missing required fields	Display error message: *Fields cannot be empty*	Pass
3	Register with duplicate email	Display error message: *Email already exists*	Pass

4	Register with invalid email format	Display error message: *Invalid email format*	Pass
5	Register with a weak password	Display error message: *Password is too weak*	Pass
6	Add Site Already added to the list	Display error message: *Already Added to the list*	

Unit Testing 5: Site Registration

No.	Test Case/Test Script	Expected Result	Result
1	Register site with valid details	Successfully create site record	Pass
2	Register with missing required fields	Display error message: *Fields cannot be empty*	Pass
3	Register with duplicate site name	Display error message: *Site name already exists*	Pass

Unit Testing 6: Change Password

No.	Test Case/Test Script	Expected Result	Result
1	Change password with valid current password	Successfully change password	Pass
2	Change password with incorrect current password	Display error message: *Current password is incorrect*	Pass
3	Attempt to change password with mismatched passwords	Display error message: *Passwords do not match*	Pass
4	Attempt to set a weak password	Display error message: *Password is too weak*	Pass

Unit Testing 7: Delete Supervisor

No.	Test Case/Test Script	Expected Result	Result
1	Delete existing supervisor	Successfully delete supervisor account	Pass
2	Attempt to delete non-existing supervisor	Display error message: *Supervisor not found*	Pass

3	Attempt to delete supervisor without permissions	Display error message: *Unauthorized action*	Pass

Unit Testing 8: Delete Site

No.	**Test Case/Test Script**	**Expected Result**	**Result**
1	Delete existing site	Successfully delete site record	Pass
2	Attempt to delete non-existing site	Display error message: *Site not found*	Pass
3	Attempt to delete site without permissions	Display error message: *Unauthorized action*	Pass

Unit Testing 9: Update Site Information

No.	**Test Case/Test Script**	**Expected Result**	**Result**
1	Update site details with valid data	Successfully save updated information	Pass
2	Attempt to update site with missing fields	Display error message: *Fields cannot be empty*	Pass
3	Attempt to update site with invalid input	Display error message: *Invalid input format*	Pass
4	Update site assigned supervisors	Successfully update site supervisor list	Pass

Unit Testing 10: Update Site Information

No.	Test Case/Test Script	Expected Result	Result
1	Change site status from Active to Closed	Successfully update site status	Pass
2	Change site status from Closed to Active	Successfully update site status	Pass
3	Attempt to change status of a non-existing site	Display error message: Site not found	Pass
4	Attempt to change status without permissions	Display error message: Unauthorized action	Pass

Unit Testing 11: Change Site Sensitivity

No.	Test Case/Test Script	Expected Result	Result
1	Change site sensitivity to High	Successfully update site sensitivity	Pass
2	Change site sensitivity to Medium	Successfully update site sensitivity	Pass
3	Change site sensitivity to Low	Successfully update site sensitivity	Pass
4	Attempt to change sensitivity of a non-existing site	Display error message: Site not found	Pass
5	Attempt to change sensitivity without permissions	Display error message: Unauthorized action	Pass

Unit Testing 12: View Alerts

No.	Test Case/Test Script	Expected Result	Result
1	View all alerts for a site	Display list of alerts with details (timestamp, violation type, status)	Pass
2	View alerts for a non-existing site	Display error message: Site not found	Pass
3	View alerts without permissions	Display error message: Unauthorized action	Pass

Unit Testing 13: Mark Alerts

No.	Test Case/Test Script	Expected Result	Result
1	Mark an alert as “In Progress”	Successfully update alert status	Pass
2	Mark an alert as “Done”	Successfully update alert status	Pass
3	Mark a non-existing alert	Display error message: Alert not found	Pass

4	Attempt to mark alert without permissions	Display error message: Unauthorized action	Pass

Unit Testing 14: View Alert Details

No.	Test Case/Test Script	Expected Result	Result
1	View details of an alert	Display alert details (timestamp, violation type, status, location)	Pass
2	View details of a non-existing alert	Display error message: Alert not found	Pass
3	View details without permissions	Display error message: Unauthorized action	Pass

Unit Testing 15: Mark Alerts as Spam

No.	Test Case/Test Script	Expected Result	Result
1	Mark an alert as spam	Successfully mark alert as spam	Pass
2	Mark a non-existing alert as spam	Display error message: Alert not found	Pass
3	Attempt to mark alert as spam without permissions	Display error message: Unauthorized action	Pass

Unit Testing 16: Register Alert Response

No.	Test Case/Test Script	Expected Result	Result
1	Register response as “In Progress”	Successfully update alert response	Pass
2	Register response as “Done”	Successfully update alert response	Pass

3	Register response for a non-existing alert	Display error message: Alert not found	Pass
4	Attempt to register response without permissions	Display error message: Unauthorized action	Pass

Unit Testing 17: Incident Management

No.	Test Case/Test Script	Expected Result	Result
1	Log incident with valid video data	Incident logged with type, time, and location	Pass
2	Log incident with missing video data	Display error message: Missing video input	Pass

Unit Testing 18: Incident Categorization

No.	Test Case/Test Script	Expected Result	Result
1	Categorize incident as “Fall”	Incident updated successfully	Pass
2	Categorize a non-existing incident	Display error message: Incident not found	Pass

Unit Testing 19: Response Time Tracking and Escalation

No.	Test Case/Test Script	Expected Result	Result
1	Response within time	No escalation triggered	Pass
2	No response in time	Escalation triggered to admin	Pass
3	Track response time of supervisor	Time logged correctly	Pass

Unit Testing 20: Generate Safety Report

No.	Test Case/Test Script	Expected Result	Result
1	Generate report after logging incident	Report includes new incident data with correct format	Pass
2	Generate report with no incidents	Display message: No incidents to report	Pass
3	Generate report as user	Successfully generate and view full report	Pass
4	Attempt to generate report without permissions	Display error message: Unauthorized action	Pass

Unit Testing 21: Visualize Data Trends

No.	**Test Case/Test Script**	**Expected Result**	**Result**
1	View trend graph for past days	Display accurate trend chart	Pass
2	Select invalid date range	Display error message: Invalid date range	Pass
3	View heatmap of violations	Heatmap displayed with accurate location data	Pass
4	Attempt to view analytics without permissions	Display error message: Unauthorized action	Pass

Unit Testing 22: Forecast Risk Levels

No.	**Test Case/Test Script**	**Expected Result**	**Result**
1	Predict risk for a new day	Display forecast with risk level (Low/Medium/High)	Pass

2	Forecast with incomplete data	Display message: Insufficient data for forecasting	Pass
3	View forecast as supervisor	Show personalized risk dashboard	Pass

Unit Testing 23: Chatbot Response Handling

No.	Test Case/Test Script	Expected Result	Result
1	Ask policy-related question	Accurate and structured response from LLM	Pass
2	Ask question with typo ("saftey helmet rule")	Chatbot corrects and answers accurately	Pass
3	Ask non-relevant query ("What is your name?")	Display message: Query not relevant to safety protocol	Pass
4	Ask query with no matching documents	Display fallback message: No information available	Pass

7.2 Functional Testing

The functional testing will take place after the unit testing. In this functional testing, the functionality of each of the module is tested. This is to ensure that the system produced meets the specifications and requirements.

Functional Testing 1: Login with different roles (Admin, Supervisor)
Objective: To ensure that the correct page with the correct navigation bar is loaded based on the role.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Login as an Admin user	Username: admin001, Password: correctpassword	Main page for Admin is loaded with Admin navigation bar	Logged in and redirected to Admin main page	Pass

2	Login as a Supervisor user	Username: sup001, Password: correctpassword	Main page for Supervisor is loaded with Supervisor navigation bar	Logged in and redirected to Supervisor page	Pass
3	Login as an Admin with incorrect password	Username: admin001, Password: wrongpassword	Display error message: *Invalid credentials*	Login failed with error message	Pass
4	Login as an unregistered user	Username: testuser, Password: testpassword	Display error message: *User not found*	Login failed with error message	Pass

Functional Testing 2: Register Supervisor
Objective: To ensure that supervisors are successfully registered and validation rules are enforced.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Register a new supervisor with valid data	Name: Zamin, Email: zamin @example.com, Password: validPassword123	Supervisor account is created successfully	Supervisor registered successfully	Pass
2	Register a supervisor with duplicate email	Name: Jane Doe, Email: zamin @example.com, Password: newPassword123	Display error message: *Email already exists*	Registration failed with error message	Pass
3	Register with missing required fields	Name: (empty), Email: supervisor2@example.com, Password: validPassword123	Display error message: *Fields cannot be empty*	Registration failed with error message	Pass
4	Register with weak password	Name: Shahryar, Email: shahryar @example.com, Password: 12345	Display error message:	Registration failed with error message	Pass

			Password is too weak		

Functional Testing 3: Change Site Status
Objective: To ensure that site status can be toggled between Active and Closed by authorized users.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Change site status from Active to Closed	Site ID: 123, Status: Closed	Site status updated to Closed	Status updated successfully	Pass
2	Change site status from Closed to Active	Site ID: 123, Status: Active	Site status updated to Active	Status updated successfully	Pass
3	Attempt to change status of a non-existing site	Site ID: 999, Status: Closed	Display error message: Site not found	Error message displayed	Pass
4	Attempt to change status without permissions	Site ID: 123, Status: Closed (by unauthorized user)	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 4: Change Site Sensitivity
Objective: To ensure that site sensitivity can be updated by authorized users.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Change site sensitivity to High	Site ID: 123, Sensitivity: High	Site sensitivity updated to High	Sensitivity updated successfully	Pass
2	Change site sensitivity to Medium	Site ID: 123, Sensitivity: Medium	Site sensitivity updated to Medium	Sensitivity updated successfully	Pass
3	Change site sensitivity to Low	Site ID: 123, Sensitivity: Low	Site sensitivity updated to Low	Sensitivity updated successfully	Pass
4	Attempt to change sensitivity of a non-existing site	Site ID: 999, Sensitivity: High	Display error message: Site not found	Error message displayed	Pass
5	Attempt to change sensitivity without permissions	Site ID: 123, Sensitivity: High	Display error message:	Error message displayed	Pass

		(by unauthorized user)	Unauthorized action		

Functional Testing 5: View Alerts
Objective: To ensure that users can view alerts for a specific site.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	View all alerts for a site	Site ID: 123	Display list of alerts with details (timestamp, violation type, status)	Alerts displayed successfully	Pass
2	View alerts for a non-existing site	Site ID: 999	Display error message: Site not found	Error message displayed	Pass
3	View alerts without permissions	Site ID: 123 (by unauthorized user)	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 6: Mark Alerts
Objective: To ensure that alerts can be marked as "In Progress" or "Done" by authorized users.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Mark an alert as "In Progress"	Alert ID: 456, Status: In Progress	Alert status updated to In Progress	Status updated successfully	Pass
2	Mark an alert as "Done"	Alert ID: 456, Status: Done	Alert status updated to Done	Status updated successfully	Pass
3	Mark a non-existing alert	Alert ID: 999, Status: In Progress	Display error message: Alert not found	Error message displayed	Pass
4	Attempt to mark alert without permissions	Alert ID: 456, Status: In Progress (by unauthorized user)	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 7: View Alert Details
Objective: To ensure that users can view detailed information for a specific alert.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	View details of an alert	Alert ID: 456	Display alert details (timestamp, violation type, status, location)	Alert details displayed successfully	Pass

2	View details of a non-existing alert	Alert ID: 999	Display error message: Alert not found	Error message displayed	Pass
3	View details without permissions	Alert ID: 456 (by unauthorized user)	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 8: Mark Alerts as Spam
Objective: To ensure that alerts can be marked as spam by authorized users.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Mark an alert as spam	Alert ID: 456, Spam: True	Alert marked as spam	Alert marked successfully	Pass
2	Mark a non-existing alert as spam	Alert ID: 999, Spam: True	Display error message: Alert not found	Error message displayed	Pass
3	Attempt to mark alert as spam without permissions	Alert ID: 456, Spam: True (by unauthorized user)	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 9: Register Alert Response
Objective: To ensure that users can register responses for alerts (In Progress, Done).

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Register response as "In Progress"	Alert ID: 456, Response: In Progress	Alert response updated to In Progress	Response updated successfully	Pass
2	Register response as "Done"	Alert ID: 456, Response: Done	Alert response updated to Done	Response updated successfully	Pass
3	Register response for a non-existing alert	Alert ID: 999, Response: In Progress	Display error message: Alert not found	Error message displayed	Pass
4	Attempt to register response without permissions	Alert ID: 456, Response: In Progress (by unauthorized user)	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 10: Add Site
Objective: To ensure that new construction sites are successfully added to the system.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Add a site with valid details	Name: Site A, Location: City A, Supervisors: [supervisor1]	Site is created successfully	Site added successfully	Pass
2	Add a site with duplicate name	Name: Site A, Location: City B, Supervisors: [supervisor2]	Display error message: Site name already exists	Site creation failed with error message	Pass
3	Add a site with missing location	Name: Site B, Location: (empty), Supervisors: [supervisor3]	Display error message: Location is required	Site creation failed with error message	Pass

Functional Testing 11: Automated Incident Logging
Objective: Ensure incidents are automatically logged with correct metadata.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Log valid incident	Type: Fall, Time: 08:30, Location: Site A	Incident logged	Logged successfully	Pass
2	Log with incomplete data	Type: Fire, Time: Missing	Error: Incomplete data	Error shown	Pass
3	Log incident without login in	User: Not logged in	Error: Unauthorized action	Error shown	Pass

Functional Testing 12: Prioritized Alerts
Objective: Verify that alerts are prioritized based on severity.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	High severity alert	Severity: High	Alert prioritized	Alert marked High	Pass
2	Low severity alert	Severity: Low	Alert marked Low	Alert marked Low	Pass

3	Alert with missing severity	Severity: Null	Error: Cannot prioritize	Error shown	Pass

Functional Testing 13: Incident Categorization
Objective: Test manual categorization of incidents.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Categorize as “Fire”	Category: Fire	Incident updated	Category assigned	Pass
2	Categorize without login in	User: Not logged in	Error: Unauthorized action	Error shown	Pass
3	Select invalid category	Category: "Flying"	Error: Invalid category	Error shown	Pass

Functional Testing 14: Response Time Tracking and Escalation
Objective: Ensure response times are tracked and escalations are triggered appropriately.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Supervisor responds in 2 mins	Time: < 5 mins	No escalation	No escalation triggered	Pass
2	No response in 10 mins	Time: > 5 mins	Escalation triggered	Escalated to admin	Pass
3	Track without login in	User: Not logged in	Error: Unauthorized action	Error shown	Pass

Functional Testing 15: Generate Safety Report
Objective: To ensure users can generate reports containing incident data, analysis, and recommendations.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Generate report after incident	Incident Type: Fall, Time: 10:30 AM	Report includes incident with full details	Report generated successfully	Pass
2	Generate report with no incidents	Incident List: Empty	Display message: No incidents to report	Message displayed correctly	Pass
3	Generate report as Admin	User Role: Admin	Full report with analytics generated	Full report visible	Pass
4	Generate report without login in	User: Not logged in	Display error message: Unauthorized action	Error message displayed	Pass

Functional Testing 16: Visualize Data Trends
Objective: To verify the system displays correct charts and patterns from incident data.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	View trend chart	Date Range: Last 7 days	Chart showing safety violation trends	Chart displayed correctly	Pass
2	Select invalid date range	Start Date: 2026-12-01, End Date: 2025-01-01	Display error: Invalid range	Error shown	Pass
3	View heatmap of incidents	Location Filter: Site A	Heatmap displays violation density	Heatmap loaded	Pass
4	View analytics without login in	User: Not logged in	Display error message: Unauthorized action	Error shown	Pass

Functional Testing 17: Forecast Risk Levels
Objective: To test that the system predicts safety risks using machine learning.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Predict risk for next day	Data: Past 30 days incidents	Display predicted risk level	Forecast displayed	Pass
2	Forecast with incomplete data	Data: Only 1 incident	Show message: Insufficient data	Message displayed	Pass
3	Forecast as supervisor	Role: Supervisor	Show personalized prediction dashboard	Dashboard shown	Pass

Functional Testing 18: Chatbot Query Processing
Objective: To ensure chatbot responds to user queries accurately using document search and LLM.

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Ask policy-related question	Query: PPE requirement	Relevant answer from policy	Correct response shown	Pass
2	Ask query with typo	Query: “emrgency protcol”	Corrected and answered	Response with corrected term	Pass
3	Ask off-topic query	Query: “Tell me a joke”	Show message: Not supported	Message shown	Pass
4	Ask question without login in	User: Not logged in	Show error message: Unauthorized action	Error shown	Pass

7.3 Business Rules Testing

Following are the business rule test cases of SafeSitePlus:

Business Rule Testing 1: Each user must have a unique email and username per role.

Conditions	Rule 1	Rule 2	Rule 3	Rule 4	Rule 5	Rule 6	Rule 7	Rule 8
Unique Username?	Yes	Yes	No	No	Yes	Yes	No	No
Unregistered Email?	Yes	No	Yes	No	Yes	No	Yes	No
Unique Role?	N/A	Yes	N/A	Yes	No	No	No	No
Actions								
Create Account	Yes	Yes	No	No	No	No	No	No
Show "Account Already Exists"	No	No	No	Yes	Yes	Yes	Yes	Yes

Business Rule Testing 2: Password must meet security criteria (at least 8 characters, 1 uppercase, 1 lowercase, and 1 number).

Conditions	**Rule 1**	**Rule 2**	**Rule 3**
At least one uppercase letter?	Yes	No	No
At least one lowercase letter?	Yes	No	No
At least one number?	Yes	No	No
Chars >= 8?	Yes	No	No
Actions			
Set Password	Yes	No	No
Show "Invalid Password"	No	Yes	Yes

Business Rule Testing 3: Email must follow the standard format (user@example.com).

Conditions	**Rule 1**	**Rule 2**
Right Email Format?	Yes	No
Actions		
Accept Email	Yes	No
Show "Invalid Email"	No	Yes

Business Rule Testing 4: The username must be unique.

Conditions	**Rule 1**	**Rule 2**
Unique Username?	Yes	No
Actions		
Create Account	Yes	No
Show "Account Already Exists"	No	Yes

Business Rule Testing 5: Only authorized supervisors or admins can manage incidents.

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
Authorized User?	Yes	No	Yes	No
Valid Incident Info?	Yes	Yes	No	No
Actions				
Report/Resolve Incident	Yes	No	No	No
Show "Not Authorized"	No	Yes	Yes	Yes

Business Rule Testing 6: Only authorized admins can configure site settings.

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
Authorized Admin?	Yes	No	Yes	No
Valid Settings?	Yes	Yes	No	No
Actions				
Update Settings	Yes	No	No	No
Show "Not Authorized"	No	Yes	Yes	Yes

Business Rule Testing 7: Only authorized users can access live camera monitoring.

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
Authorized User?	Yes	No	Yes	No
Camera Available?	Yes	Yes	No	No
Actions				
Access Camera	Yes	No	No	No
Show "Access Denied"	No	Yes	Yes	Yes

Business Rule Testing 8: Anomaly detection must trigger alerts only for real threats.

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
AI Detects Threat?	Yes	Yes	No	No
False Positive Check?	No	Yes	N/A	N/A
Actions				
Trigger Alert	Yes	No	No	No
Show "False Alarm"	No	Yes	N/A	N/A

Business Rule Testing 9: Only supervisors and admins can access the monitoring dashboard.

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
Admin or Supervisor?	Yes	No	Yes	No

Valid Site Data?	Yes	Yes	No	No
Actions				
Access Dashboard	Yes	No	No	No
Show "Access Denied"	No	Yes	Yes	Yes

Business Rule Testing 10: Incident Reports Can Only Be Accessed and Generated by Authorized Users

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**	**Rule 5**	**Rule 6**
Authorized User?	Yes	No	Yes	No	Yes	No
Valid Incident Data?	Yes	Yes	No	No	Yes	Yes
Actions						
Generate Incident Report	Yes	No	No	No	Yes	No
Access Incident Report	Yes	No	No	No	Yes	No
Show "Access Denied"	No	Yes	Yes	Yes	No	Yes

Business Rule Testing 11: Only Admins Can Suspend or Remove Supervisors

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**
User is Admin?	Yes	No	Yes	No
Supervisor Exists?	Yes	Yes	No	No
Actions				
Suspend/Remove Supervisor	Yes	No	No	No
Show "Supervisor Not Found"	No	No	Yes	Yes
Show "Access Denied"	No	Yes	No	Yes

Business Rule Testing 12: Only Logged-in Users Can Interact with Incident Features

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**	**Rule 5**	**Rule 6**
Logged-in User?	Yes	No	Yes	No	Yes	No
Valid Incident Data?	Yes	Yes	No	No	Yes	Yes

Sufficient Role (e.g., Supervisor)?	Yes	Yes	Yes	Yes	No	No
Actions						
Log Incident	Yes	No	No	No	No	No
Categorize Incident	Yes	No	No	No	No	No
Register Response	Yes	No	No	No	No	No
Access Incident Reports	Yes	No	No	No	No	No
Trigger Real-Time Notification	Yes	No	No	No	No	No
Show "Access Denied"	No	Yes	Yes	Yes	Yes	Yes

Business Rule Testing 13: Data Visualization Access Control

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**	**Rule 5**	**Rule 6**
Logged-in User?	Yes	No	Yes	No	Yes	No
Date Range Valid?	Yes	Yes	No	No	Yes	Yes
Has Access to Site A?	Yes	Yes	Yes	Yes	No	No
Actions						
View Trend Charts	Yes	No	No	No	No	No
Load Heatmap	Yes	No	No	No	No	No
Show "Access Denied"	No	Yes	Yes	Yes	Yes	Yes

Business Rule Testing 14: Risk Forecast is Restricted to Logged-in Supervisors

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**	**Rule 5**	**Rule 6**
Logged-in User?	Yes	No	Yes	Yes	Yes	No
Role: Supervisor?	Yes	Yes	No	Yes	No	Yes

Sufficient Data?	Yes	Yes	Yes	No	Yes	Yes
Actions						
Show Risk Prediction	Yes	No	No	No	No	No
Access Dashboard	Yes	No	No	No	No	No
Show "Access Denied"	No	Yes	Yes	Yes	Yes	Yes

Business Rule Testing 15: Chatbot Queries Are Allowed Only for Logged-in Users

Conditions	**Rule 1**	**Rule 2**	**Rule 3**	**Rule 4**	**Rule 5**	**Rule 6**
Logged-in User?	Yes	No	Yes	No	Yes	No
Query Type Valid?	Yes	Yes	No	No	Yes	Yes
Domain-Related Question?	Yes	Yes	No	No	No	No
Actions						
Process Chat Query	Yes	No	No	No	No	No
Show Error Message	No	Yes	Yes	Yes	Yes	Yes

7.4 Integration Testing

Table 1: Integration Test Table for Incident Reporting

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Login as Authorized User	Username: user@example.com, Password: userpassword	User is logged in and redirected to the incident report dashboard	Logged in and redirected successfully	Pass

2	Submit New Incident Report	Incident Type: Fire, Location: Zone A, Description: Fire near machinery	Incident report is created successfully	Incident reported successfully	Pass
3	Attach Evidence (Image/Video)	File: fire_incident.jpg	Evidence is attached to the report successfully	File uploaded and attached	Pass
4	Assign Incident to Supervisor	Supervisor: John Doe	Supervisor receives an assigned incident notification	Supervisor notified successfully	Pass
5	View Incident Reports	Role: Authorized User	Incident reports are displayed based on permissions	Reports displayed correctly	Pass
6	Attempt Unauthorized Access	Role: Unauthorized User	System restricts access	Access denied message displayed	Pass

Table 2: Integration Test Table for Supervisor Dashboard

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	**Login as Supervisor**	Username: zaminraza095@gmail.com, Password: zamin1234	Supervisor dashboard is displayed with relevant options	Logged in and redirected successfully	Pass
2	**View Alerts**	Click on **Alerts Section**	List of recent alerts is displayed	Alerts displayed successfully	Pass
3	**View Notifications**	Click on **Notifications**	Latest notifications appear	Notifications displayed correctly	Pass
4	**Update Site Anomalies**	Select **Site A**, Update **Anomaly Status: Resolved**	Anomaly status is updated successfully	Status updated	Pass
5	**View Sites**	Click on **View Sites**	A list of all assigned sites appears	Sites displayed correctly	Pass

6	**Access Notes Hub**	Click on **Notes Hub**	Supervisor can create, edit, and delete notes	Notes Hub is functional	Pass
7	**View Stats**	Click on **Stats**	Displays site performance and incident trends	Stats displayed successfully	Pass
8	**Use Chatbot**	Enter **"How to report an issue?"**	Chatbot responds with a help message	Chatbot works as expected	Pass
9	**See Weather Details**	Click on **Weather Section**	Displays live weather updates for assigned sites	Weather details displayed	Pass

Table 3: Integration Test Table for Admin Dashboard

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	**Login as Admin**	Username: zaminraza095@gmail.com, Password: zamin1234	Admin dashboard loads with navigation menu	Dashboard loaded successfully	Pass
2	**View All Supervisors**	Click on "Supervisors" tab	List of all registered supervisors appears	Supervisors displayed correctly	Pass
3	**View All Sites**	Click on "Sites" tab	List of all registered sites appears	Sites displayed correctly	Pass
4	**Add a New Supervisor**	Enter supervisor details (Name, Email, Contact) and click "Add"	Supervisor is added successfully and appears in the list	Supervisor added successfully	Pass
5	**Add a New Site**	Enter site details (Name, Location, Type) and click "Add"	New site is added and listed under "Sites"	Site added successfully	Pass

6	**Assign a Site to a Supervisor**	Select a supervisor and choose a site to assign	Site is assigned to the selected supervisor	Site assigned successfully	Pass
7	**Manage Supervisors**	Edit, suspend, or remove a supervisor	Supervisor details updated, suspended, or removed successfully	Supervisor managed successfully	Pass
8	**Review Incident Reports**	Open "Incident Reports" section	List of all incident reports appears with filtering options	Incident reports displayed correctly	Pass
9	**Monitor User Activity**	Open "User Activity" log	Displays login history and actions taken by supervisors	User activity log displayed successfully	Pass

Table 4: Integration Test Table for Alerts Prioritization

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	**View Alerts**	Open alerts panel	List of all alerts appears, sorted by priority	Alerts displayed correctly	Pass
2	**Prioritize Alerts**	Select priority: **Critical, High, Medium, Low**	Alerts are categorized and color-coded based on priority	Alerts prioritized correctly	Pass
3	**Acknowledge an Alert**	Select alert and mark as **Acknowledged**	Alert status changes from "Unresolved" to "Acknowledged"	Alert acknowledged successfully	Pass
4	**Resolve an Alert**	Select alert and mark as **Resolved**	Alert status updates to "Resolved"	Alert resolved successfully	Pass

Table 5: Integration Testing – Chatbot Response Handling

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Query policy information	Query: "PPE guidelines"	Accurate response retrieved from knowledge base	Correct response shown	Pass
2	Handle misspelled input	Query: "emrgency protcol"	Corrected and accurate response provided	Response with correction	Pass
3	Process off-topic query	Query: "Tell me a joke"	Display message: Not supported	Appropriate message shown	Pass
4	Access chatbot without login in	User: Not logged in	Display error message: Unauthorized action	Error message displayed	Pass

Table 6: Integration Testing – Forecast Risk Levels

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Predict risk for upcoming day	Data: Past 30 days incidents	Display predicted risk level	Forecast displayed	Pass
2	Forecast with insufficient data	Data: Only 1 incident	Show message: Insufficient data	Message displayed	Pass
3	Forecast as supervisor	Role: Supervisor	Show personalized prediction dashboard	Dashboard shown	Pass
4	Access forecast without login in	User: Not logged in	Display error message: Unauthorized action	Error message displayed	Pass

Table 7: Integration Testing – Incident Management

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Log valid incident	Type: Fall, Time: 08:30, Location: Site A	Incident logged	Logged successfully	Pass
2	Log with incomplete data	Type: Fire, Time: Missing	Error: Incomplete data	Error shown	Pass
3	Log incident without login in	User: Not logged in	Error: Unauthorized action	Error message displayed	Pass
4	Resolve an incident	Incident ID: 456, Action: Resolve	Incident status updated to Resolved	Status updated successfully	Pass

Table 8: Integration Testing – Incident Categorization

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Categorize incident as "Fall"	Category: Fall	Incident updated successfully	Category assigned	Pass
2	Categorize a non-existing incident	Incident ID: 999	Display error message: Incident not found	Error message displayed	Pass
3	Categorize without selection	Category: None	Display error: Category required	Error shown	Pass
4	Categorize incident without login in	User: Not logged in	Display error message: Unauthorized action	Error message displayed	Pass

Table 9: Integration Testing – Response Time Tracking and Escalation

No.	Test Case/Test Script	Attribute and Value	Expected Result	Actual Result	Result
1	Supervisor responds in 2 mins	Time: < 5 mins	No escalation	No escalation triggered	Pass
2	No response in 10 mins	Time: > 5 mins	Escalation triggered	Escalated to admin	Pass
3	Track response time of supervisor	Supervisor ID: 123	Time logged correctly	Time recorded accurately	Pass
4	Track response without login in	User: Not logged in	Display error message: Unauthorized action	Error message displayed	Pass

8 Conclusion and Future Work

8.1 Conclusion

SafeSitePlus provides a comprehensive solution for construction site safety management, focusing on automated PPE compliance detection, incident tracking, and supervisor support. This document outlines the essential requirements and features that make SafeSitePlus a complete safety management tool.

SafeSitePlus addresses key challenges faced by site supervisors and safety managers by integrating hazard detection, automated alerts, incident logging, and AI-driven risk prioritization into a single platform. The system reduces manual monitoring efforts while ensuring compliance with safety regulations.

With automated safety checks, incident reporting, and analytics, businesses can proactively manage site safety, reduce risks, and improve compliance. SafeSitePlus allows supervisors to efficiently oversee multiple sites, prioritize critical incidents, and make data-driven decisions to enhance workplace safety. Its user-friendly interface and AI-powered tools empower businesses to create a safer, more efficient work environment with minimal effort

8.2 Future Work

In the future, SafeSitePlus will focus on optimizing algorithms for faster processing and improved efficiency. This includes refining predictive models, optimizing database queries, and implementing load balancing techniques to handle larger datasets and ensure quicker responses. These improvements will enhance the system's performance, scalability, and overall user experience.

8. References

[1] William S. Chao, Software Requirements Specification (SRS) 2.0: The Structure-Behavior Coalescence Approach, CreateSpace Independent Publishing Platform, 2013.

[2] Beatty, K. W. (2013). Software Requirements. Redmond, Washington: Microsoft Press.Ali Behforooz& Frederick J.Hudson, (1996), Software Engineering Fundamentals, Oxford University Press. Chapter 8, pp255-235.

[3] Ian Sommerville, (2015), Software Engineering, 10th Edition, Pearson. Chapter 4, ISBN: 9780137586691.

[4] JamaSoftware,"Functional Requirements Examples and Templates," https://www.jamasoftware.com/requirements-management-guide/writing-requirements/functional-requirements-examples-and-templates, Last accessed September 24, 2024.

[5] Al Daghan, A.T.A., Vineeta, Kesh, S., Manek, A.S. "A Deep Learning Model for Detecting PPE to Minimize Risk at Construction Sites." Published in: 2021 IEEE International Conference on Electronics, Computing and Communication Technologies (CONECCT).

[6] Delhi, V. S. K. "Detection of Personal Protective Equipment (PPE) Compliance on Construction Site Using Computer Vision Based Deep Learning Techniques." Internet: https://www.frontiersin.org/articles/10.3389/fbuil.2020.00136/full, 24 September 2020 [Sep,16,2023].

9. Plagiarism