Fig. 1 SETU logo

Software Development Year 4 Project

Design Document

Project Title: SiteSafeAI

By: Tomas Smitas
Student No: C00276177
Date: 2024
Supervisor: Dr Oisin Cawley

Fig. 2 SiteSafe Logo

Table of Contents

Introduction

This design document provides a detailed plan for how the AI construction monitoring application will meet its outlined requirements. Building on the functional specification, this document shifts the focus toward the technical approaches and high-level design choices that will shape the implementation process.

The application's primary objective is to enhance construction site safety by detecting personal protective equipment (PPE), specifically helmets and high-visibility jackets, using machine learning.

Following the implementation of the core detection functionality, the application will include additional features such as real-time alerts, data logging, and reporting. These secondary functions are essential to providing actionable insights and ensuring comprehensive safety monitoring on construction sites. It is a way to set this project from others who just detect the safety equipment and then do nothing with it. The design of these supplementary features will be discussed in detail.

Subsequent sections will outline the technological stack selected for this project, including the reasoning behind these choices and their role in the development process. Key elements such as the machine learning model architecture, front-end and back-end frameworks, and supporting tools will be described to provide a clear vision of how the application will be constructed.

Finally, this document will present the interaction flow between the system's components through sequence diagrams and provide a user interface overview. This includes a component diagram and prototype screenshots to demonstrate the application's functionality and user experience.

Technologies

Extensive research guided the selection of technologies for this project to ensure they meet the requirements of real-time video monitoring, detection accuracy, and system integration. Below is an overview of the chosen technologies and their roles in the project:

- **Python** serves as the primary programming language due to its robust libraries for AI and machine learning. It will be used to manage object detection workflows, integrate the chosen frameworks, and handle real-time data processing for the live monitoring system through flask.
- **YOLOv11** (You Only Look Once) is essential for detecting helmets and high-visibility jackets in real-time. It processes live video feeds, identifying and classifying objects in a single pass, making it highly suitable for the speed and accuracy requirements of this project.
- **Ultralytics** enhances YOLO's capabilities. It simplifies the deployment and testing of the detection models, allowing seamless integration into the project's monitoring pipeline.
- **OpenCV** will manage video capture and annotation tasks. It will overlay bounding boxes, labels, and confidence scores on the detected objects in video frames, providing a clear visual representation of safety compliance on the construction site.
- **Flask** will function as the backend framework, enabling communication between the detection system and the web interface. It will handle API requests, process detection results, and serve data to the React-based frontend for display.
- **React** will power the web-based user interface, ensuring interactivity and responsiveness. It will allow users to view live monitoring feeds, configure alerts, and interact with the system seamlessly.
- **SMTP** will be used to send email alerts when a safety violation, such as the absence of a helmet or high-visibility jacket, is detected. This ensures real-time notifications for site supervisor/ safety officer.
- **MySQL** will store detection logs, timestamps, and alert records in a structured format. This database will support efficient data retrieval for generating reports and monitoring site compliance trends.

This combination of technologies ensures a robust, efficient, and scalable solution for real-time safety monitoring on construction sites.

Demo Site & Company Site

This project will feature two distinct sites to cater to different user needs: a Demo Site for showcasing the product and a Company Site for registered companies to use the full system.

Demo Site

The demo site is designed to provide potential users with a hands-on experience of the product's features. Users can explore most functionalities without the need for registration. However, the demo site comes with several limitations:

- **Single Camera Connection**: Users can only connect one camera to test the detection system.
- **Default Settings**: Settings are not saved. Upon refreshing the site, all configurations revert to default values.
- **Limited Dashboard Functionality**: The dashboard button is available, but instead of displaying real-time data, it opens an example dashboard to illustrate the system's capabilities.
- **Limited usage**: Users have a limited time to use the detection model for free to reduce load on the server.

This site is ideal for showcasing the product to prospective customers without requiring a full commitment.

Company Site

The company site is tailored for registered users, providing full access to all system features. Companies must register their details to gain access. Key benefits of the company site include:

- **Multi-Camera Support**: Registered users can connect and monitor multiple cameras simultaneously, enabling comprehensive site surveillance.
- **Persistent Settings**: Configurations, such as alert preferences and notification thresholds, are saved for each user.
- **Fully Functional Dashboard**: The dashboard displays real-time safety statistics, including incidents, trends, and performance metrics. This data helps companies monitor and improve site safety.

By offering both a demo site and a fully functional company site, the project ensures accessibility for new users while providing robust features for registered companies.

Personal Protective Equipment Detection

Helmet Detection

Ensuring workers on construction sites wear helmets is critical for reducing the risk of head injuries caused by accidents or falling objects. The effectiveness of helmet detection using AI is one of the key features of this project, aligning with its objective of improving construction site safety. As such, selecting the most effective methodology for implementing helmet detection is a crucial design decision.

Recent advancements in machine learning provide a strong foundation for this system's design. Helmet detection primarily involves analysing visual input to identify key features of helmets on individuals in a live video feed. Techniques such as object detection using convolutional neural networks (CNNs) have proven effective in similar applications.

Focusing on helmets allows for a more streamlined and accurate first iteration. For this reason, the initial implementation will prioritise helmet detection based on trained object detection models, such as YOLO (You Only Look Once), due to its speed and precision in real-time environments.

Subsequent iterations of the system will expand to include high-visibility jackets alongside helmets. Research has shown that combining multiple safety indicators can significantly enhance the overall reliability of site safety systems (Fig 4. Work Health Safety Research, 2015). By starting with helmets as the primary focus, this project aims to deliver high accuracy and responsiveness, laying the groundwork for broader functionality in future updates.

High-Visibility Jacket Detection

The detection of high-visibility jackets is a vital component of ensuring worker compliance with safety regulations on construction sites. The chosen methodology for identifying high-visibility jackets relies on leveraging object detection models trained to recognise the unique colour, shape and reflective properties of such jackets.

Object detection libraries and frameworks provide the foundation for this functionality. Pretrained models like YOLO (You Only Look Once) have been evaluated for their suitability in identifying helmets, transitioning this to high-visibility jackets should be achievable. YOLO is particularly advantageous due to its ability to process frames in real-time, a crucial feature for live video monitoring.

For initial implementation, YOLO will be employed for its speed and compatibility with a range of camera sources. The system will be trained using datasets containing diverse examples of high-visibility jackets across varying lighting and weather conditions to ensure robustness. One limitation of YOLO, however, is that its accuracy may decrease when distinguishing between jackets and similarly coloured or reflective objects, which will be addressed through iterative model refinement.

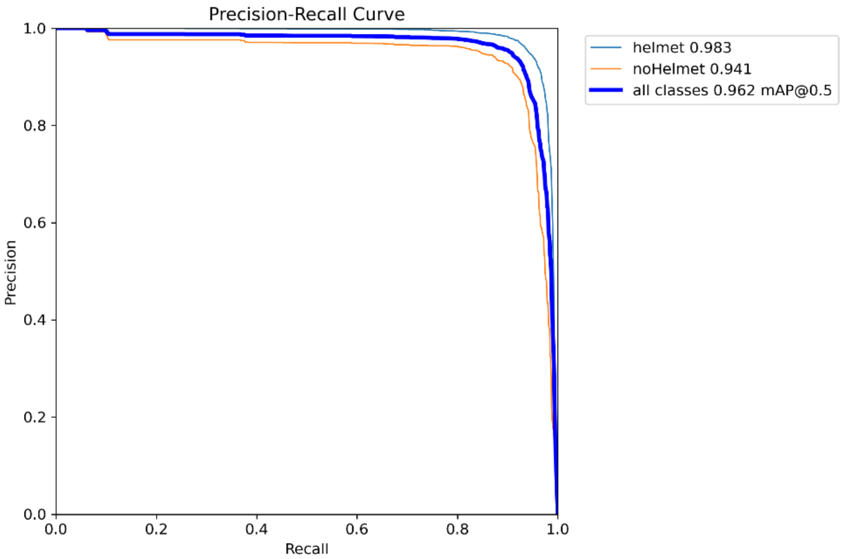
Evaluation of detection model

The evaluation of detection models for helmets and high-visibility jackets will focus on key performance metrics to ensure the system meets safety monitoring standards. Metrics such as confusion matrix, precision, recall, and the F1-score will be used to assess the balance between the model's accuracy and its ability to minimise false positives and false negatives. A confusion matrix will provide a detailed breakdown of true positives, false positives, true negatives, and false negatives, offering insights into areas of improvement for the model.

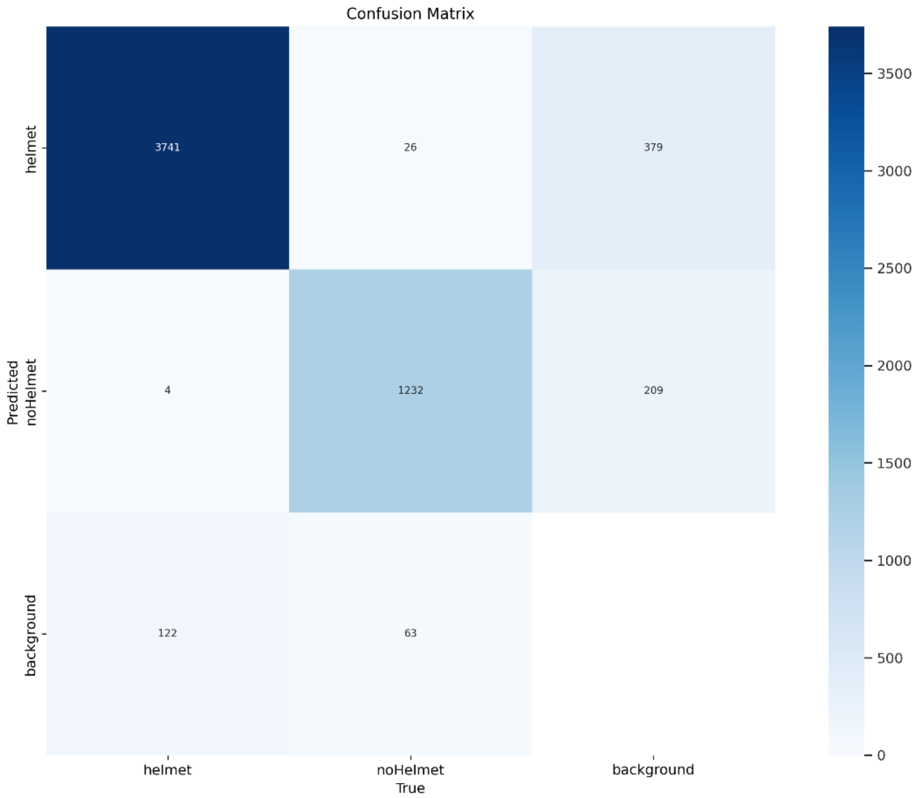
In addition to these metrics, the Precision-Recall (PR) curve will be utilised to analyse the trade-off between precision and recall across different confidence thresholds. This evaluation will help optimise the model's detection threshold for specific scenarios, such as ensuring that all workers wearing PPE are identified while minimising false alarms. High recall will be prioritised for detecting helmets and jackets, as missed detections pose significant safety risks.

These evaluations will be conducted on a diverse test dataset to account for variations in lighting, camera angles, and occlusions, ensuring the robustness of the detection system in real-world construction site environments. By iteratively refining the models based on these evaluation metrics, the system will achieve high reliability and efficiency in monitoring safety compliance. Examples below.

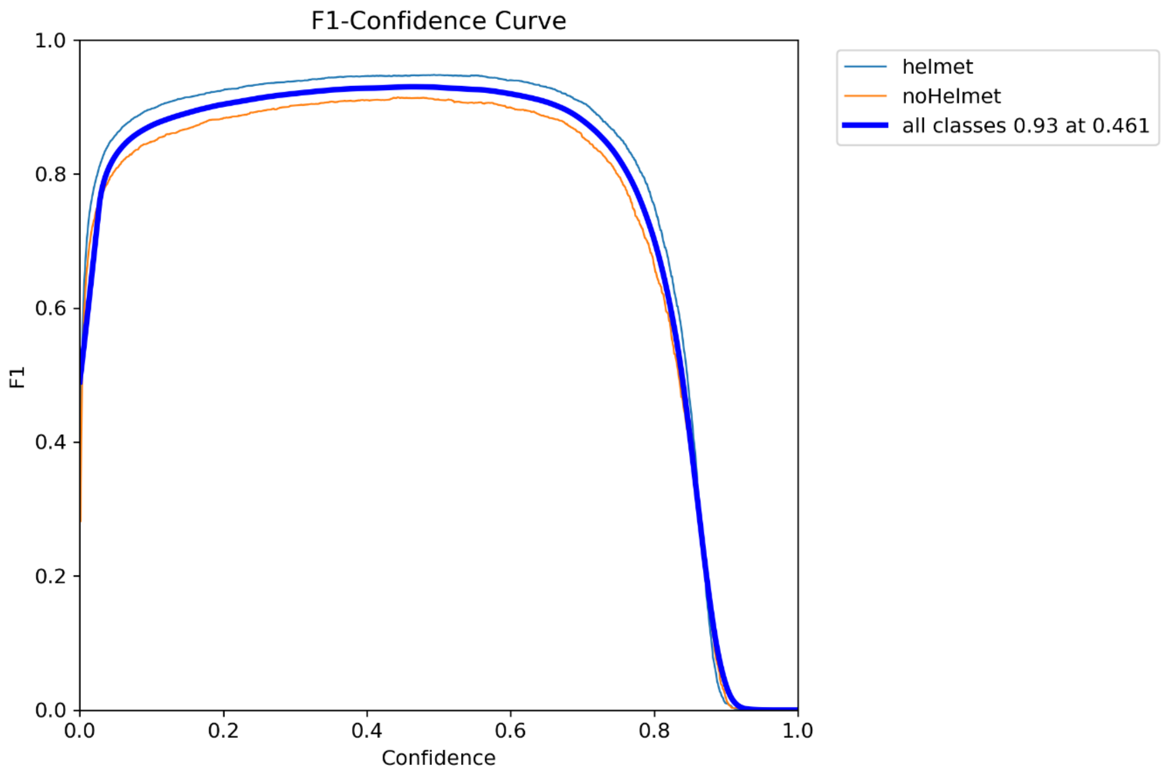
Precision-Recall (PR) curve



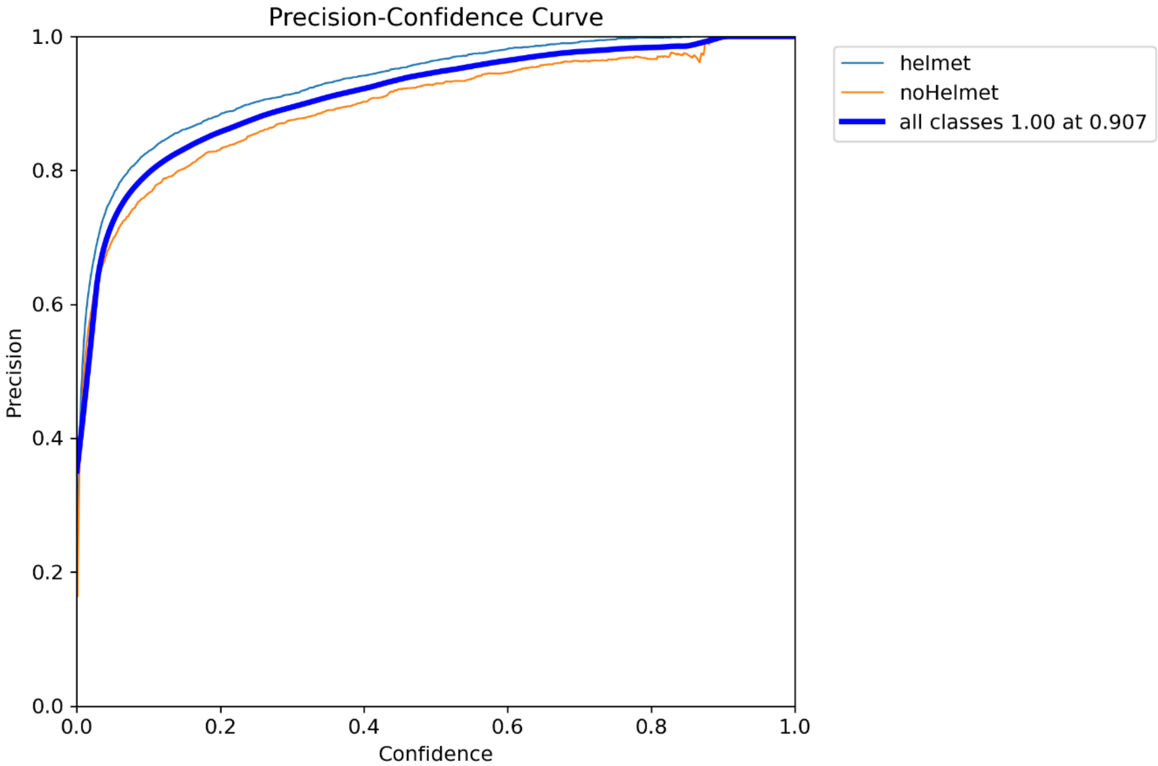
Confusion Matrix



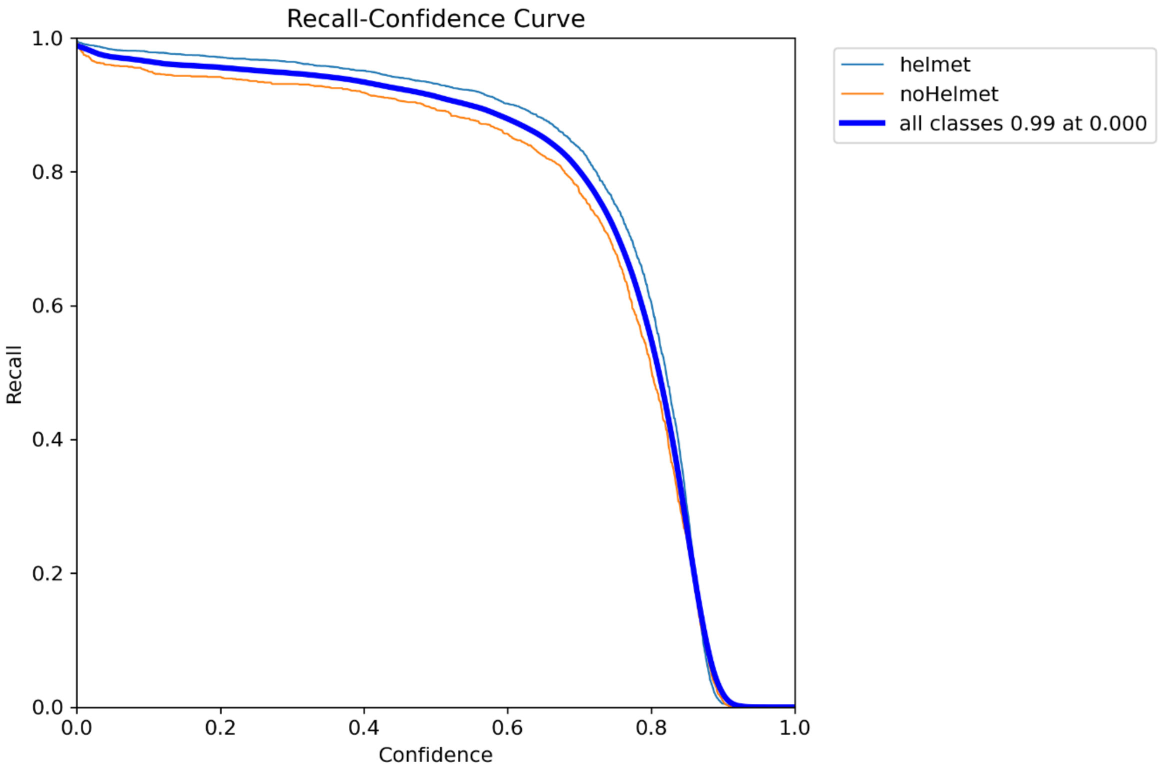
F1 – Confidence Curve



Precision – Confidence Curve



Recall – Confidence Curve



Notification Process

Incident Report

After successful detection of no helmet or/and no high visibility jacket on a construction site an incident report is automatically generated by the SiteSafeAI system. In this report details of the incident will be held, details like date/time, image, detection confidence, etc. This incident report will be accessed by the manager/ safety officer through email. The manager/ safety officer will then decide whether to act on the incident being able to use the incident report as a reference.

Camera Monitoring Specialist

This is the person that will be the main user of the website. After a successful detection of no helmet or/and no high visibility jacket the camera monitoring specialist is notified. This depends on the companies needs. Some companies have an alert system in place while others have specialist watching the cameras constantly. The case that will be implemented is where a monitoring person is watching the screen constantly. This process will be easily modifiable to suit an alert-based system as well.

Email Service

Once an incident is detected, such as a worker not wearing a helmet or high-visibility jacket, the system will send an automatic email notification to the relevant personnel, such as the manager or safety officer. The email will contain a detailed report of the incident, including relevant information like the time of detection and a snapshot of the image with bounding boxes highlighting the violation. This ensures that the safety officer or manager can take immediate action and maintain proper records for safety compliance.

The email service will use SMTP (Simple Mail Transfer Protocol) to send these alerts. The email body will be dynamically generated, pulling data from the database to generate reports with details about the incident. The system will be configured to send these notifications based on the detection of specific safety violations.

SMS Service

The SMS service will follow a similar logic to the email service. In the event of a safety violation, such as a worker not wearing the required protective gear, an SMS alert will be sent to the relevant personnel, such as the safety officer. The SMS will include basic details about the violation, such as the time and nature of the incident, with a link to the full report in the system.

While the structure for this service is planned and designed, the actual implementation will not be done in the initial version of the project. This is because SMS services typically incur ongoing costs, which may not be feasible for the first deployment of the system. However, this feature can be considered for future development, allowing for a more immediate alert system for on-the-ground personnel, especially in cases where email notifications may not be as effective in real-time.

Spam Protection

To prevent the system from sending excessive or redundant alerts, a spam protection mechanism will be implemented. This will ensure that users do not receive an overwhelming number of notifications for the same incident or similar incidents within a short period.

The approach will involve introducing a cooldown period for alerts. After an alert is triggered for a specific violation (e.g., a worker not wearing a helmet or high-visibility jacket), the system will impose a cooldown period (e.g., 15 minutes or more). During this time, even if the same violation is detected again, no new alerts will be sent until the cooldown period has expired. This will prevent the system from spamming alerts for repeated detections of the same issue.

In addition to the cooldown period, the system can be configured to aggregate multiple violations occurring in a short time into a single report. For example, if several workers are detected without helmets or high-visibility jackets within a few minutes, a single alert summarising all the violations can be sent, rather than sending multiple individual alerts.

This spam protection mechanism ensures that alerts remain meaningful and actionable without overwhelming the personnel responsible for responding to the notifications. It also improves the overall user experience, allowing for a more controlled and efficient alerting system.

Notification Email Example with Incident Report

Safety Alert: Incident Detection on Example Site

1NoHelmet.jpg

(Student C00276177) Tomas Smitas
To: (Student C00276177) Tomas Smitas

Reply Reply all Forward

Wed 11/27/2024 4:07 PM

Dear Safety Officer/Manager,

We hope this message finds you well. Our AI monitoring system has detected a potential safety concern at the **Example Site** construction site. Below are the details of the detected incident:

Incident Summary:

- Date and Time: **November 27, 2024, 10:30 AM**
- Location: **Camera 3 - North Sector**
- Issue Detected: **Worker without a helmet**
- Number of Individuals Affected: **1**

(Attached is an example image captured by the AI system to highlight the detected safety issue.)

Ensuring worker safety is our utmost priority. Please take the necessary steps to address this issue at the earliest convenience. Should you require further clarification or support, do not hesitate to reach out.

Thank you for your prompt attention to this matter.

Best regards,
SiteSafeAI System

Attached Incident Image

Fig. 3 Email Incident Detection Example

Setting Functionality

The side menu for the monitoring specialist is located on the right-hand side of the screen, next to the live video monitoring feed. This menu provides essential controls for managing the system's detection and notification features, allowing the monitoring specialist to customise their experience based on the company's needs.

Menu Components and Functionalities

1. **Camera On/Off Button**:
   - This button allows the monitoring specialist to turn the camera on or off. When the camera is turned on, live video monitoring begins, and automatic detection starts. If the camera is turned off, the system will stop processing video feeds and detecting violations.
2. **Set Automatic Email/SMS Button**:
   - When clicked, this button opens a pop-up window where the monitoring specialist can set or update the contact details for receiving notifications. The user can input an email address or a phone number for SMS alerts. This step ensures that the correct contact information is set for sending alerts when a violation is detected.
3. **Email Toggle**:
   - This toggle allows the monitoring specialist to activate or deactivate email notifications. If an email address has not been set through the Set Automatic Email/SMS option previously, the toggle will be disabled and pressing it will not allow the user to enable it until an Email or Phone Number has been set. When enabled, the system will automatically send an email alert to the specified address whenever a detection violation (such as a missing helmet or high-visibility jacket) occurs.
4. **SMS Toggle**:
   - Like the Email Toggle, this switch controls SMS notifications. The monitoring specialist can turn it on or off, but it will only be active if a phone number has been provided for SMS alerts. When activated, the system will send SMS notifications to the specified number when a violation is detected.
5. **Sensitivity Slider**:
   - The sensitivity slider allows the monitoring specialist to adjust the detection sensitivity. The slider ranges from more sensitive to less

sensitive. Adjusting the sensitivity helps fine-tune the system's responsiveness based on the environment and company preferences.

6. **Open Dashboard Button**:
    - This button opens the Dashboard page. The dashboard provides an overview of incidents detected, allowing the monitoring specialist to track safety violations and company performance over time. The dashboard presents key statistics such as the number of violations, trends, and other performance metrics relevant to the company's safety standards.

How the Functionality Works Together:

The side menu gives the monitoring specialist full control over the system's operations. By turning the camera on, they initiate the live feed and the detection process. The ability to set and manage email and SMS notifications ensures that the right people are alerted when a safety violation occurs, while the sensitivity slider allows them to fine-tune detection accuracy. The open dashboard button provides easy access to system data, which can be used to assess performance and make necessary adjustments.

This configuration is designed to ensure the monitoring specialist can quickly and easily respond to incidents while being empowered with tools to manage and optimise the detection and notification system.

Additional Functionalities

Dashboards

A Dashboard button will be made available to all registered users, providing them with an intuitive and easy-to-access overview of key safety metrics. Once clicked, this button will open a dashboard interface displaying important data such as the number of incidents detected for each user, along with other relevant statistics. These insights will help safety officers, managers, and company representatives monitor the overall safety performance of the site in real time.

Key features of the dashboard include:

- **Incident Tracking**: A visual representation of the number of incidents. This will allow managers to quickly assess safety compliance and identify areas needing improvement.
- **Safety Statistics**: The dashboard will also display aggregated statistics through interactive graphs, such as bar charts or line graphs, showing the total number of violations over a specific period. These graphs will highlight trends in safety performance, allowing users to visually track improvements or issues in safety compliance. Key metrics such as the frequency of violations by type (e.g., no helmet, no high-visibility jacket) will be represented, helping users quickly assess how well safety protocols are being followed across the site.
- **Safety Reporting**: Companies can use this dashboard to generate reports that highlight the safety status of their construction sites. These reports will be valuable for audits, safety meetings, and to showcase their commitment to worker safety.

The user interface will be designed to be easy to navigate and visually engaging, making it an essential tool for monitoring and improving workplace safety.

Incident Dashboard

Total Incidents (All Time)	Total Incidents (This Year)	Total Incidents (This Month)	Total No Helmet Incidents	Total No High-Vis Incidents
12	3	0	3	9

Project Plan

Documentation

Specifications 25/October/24

↓

Poster 25/October/24

↓

Presentation 25/October/24

↓

Design 6/November/24

↓

Report 6/March/25

Model

Technology Research 15/September/24

↓

Model training on test data 25/September/24

↓

Dataset collection 28/September/24

↓

Dataset annotation 03/October/24

↓

Model training on 100 images 10/October/24

↓

Model testing 20/October/24

↓

More Data collection and annotation 05/November/24

↓

Model Training on 500 images 25/November/24

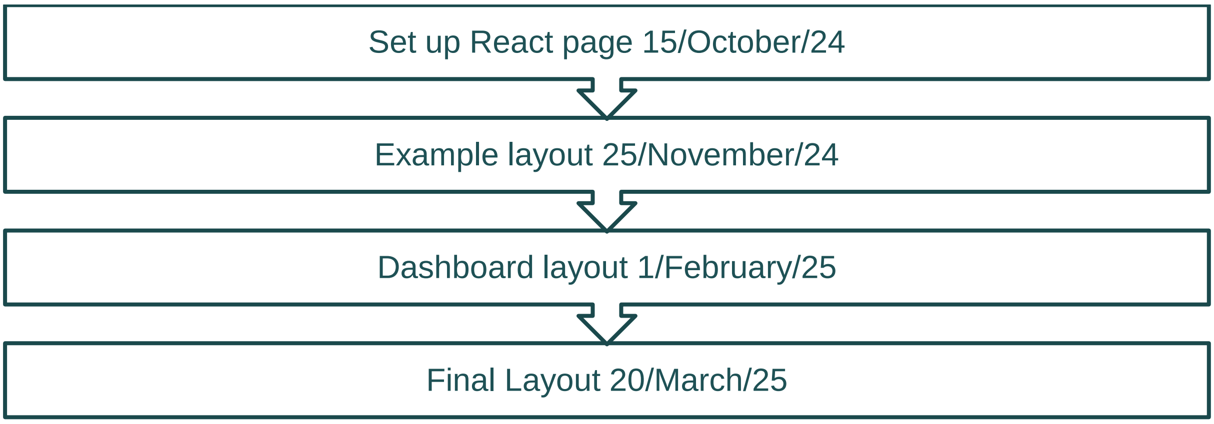
↓

Model testing 30/November/24

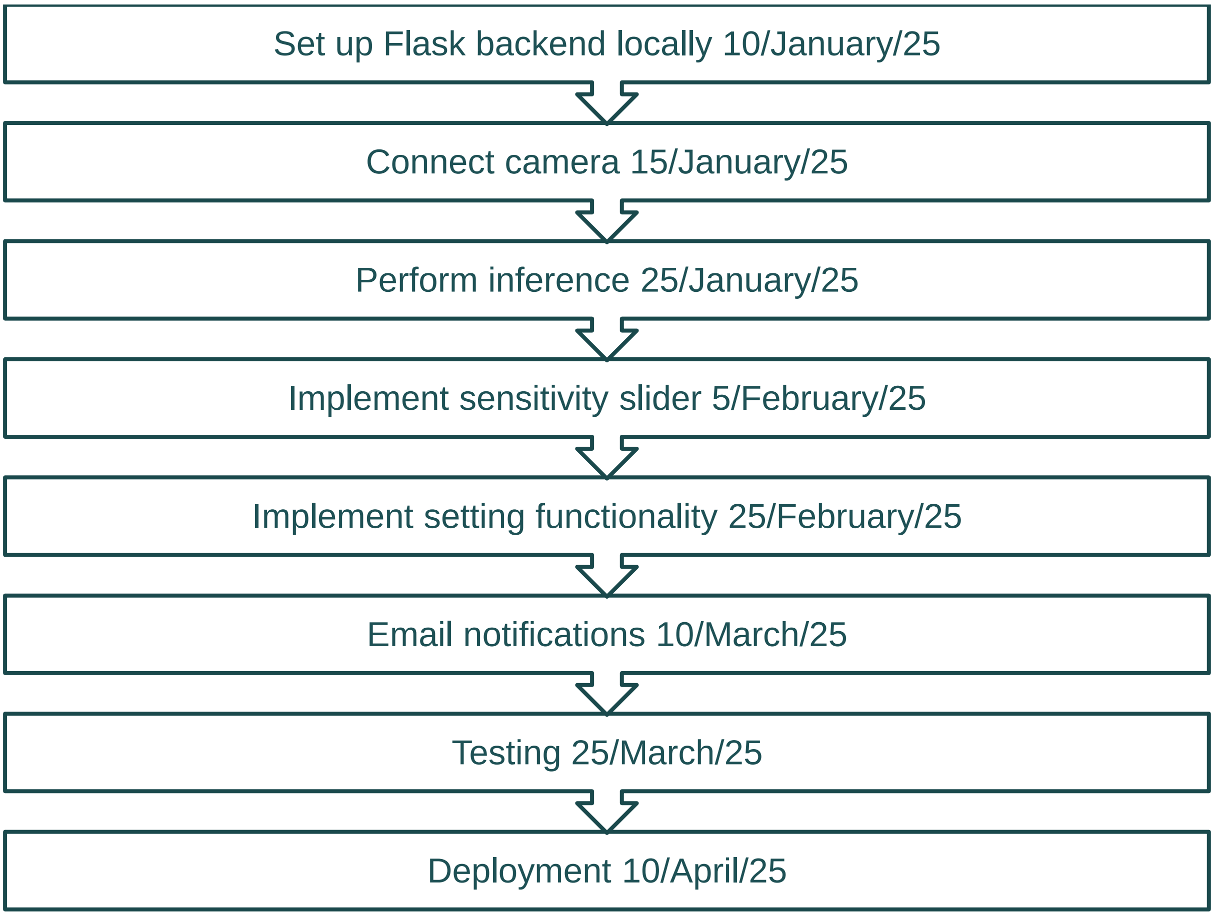
↓

Further model improvements 1/February/25

Frontend



Backend



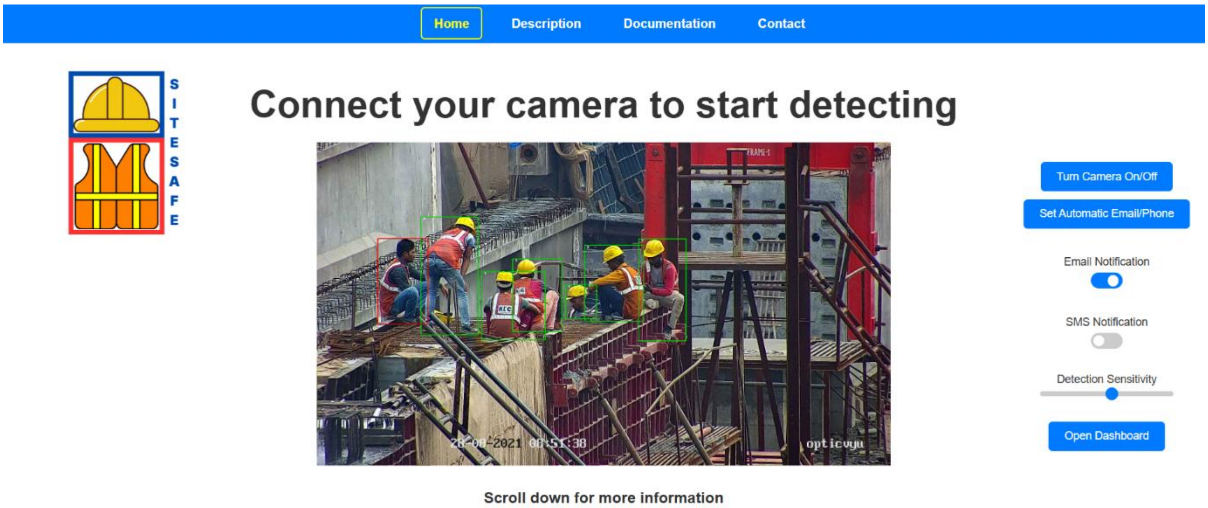
UI/UX Design

The UI/UX design for this project will keep thing as simple as possible for the user. A one-page design will be implemented for the video monitoring with the SiteSafe logo on the left-hand side and the settings on the right-hand side. At the top of the page there will be a menu for navigation, but this should see minimal use since the page is designed to be fully operational from the main page. This is to make the user experience as intuitive as possible.

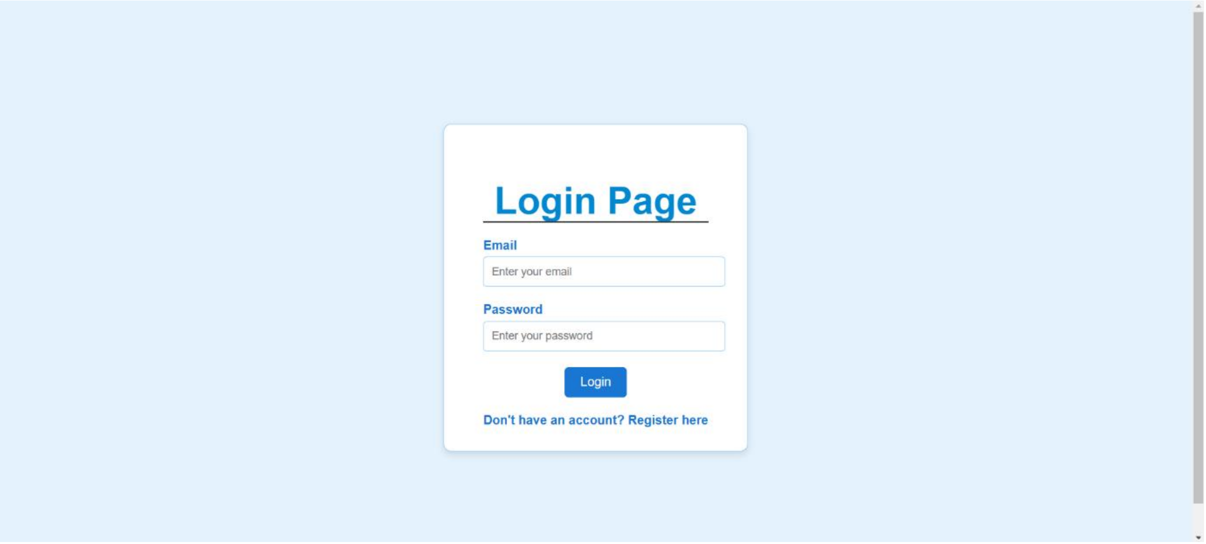
For the demo page the top navigation options will include Home, Description, Documentation and Contact. For the company page the admin will also have an option in the menu called users, where they can CRUD users.

The options on the right-hand side of the screen will include Turn Camera On/Off, Set Automatic Email/Phone, Email notification toggle, SMS notification toggle, Sensitivity slider and Open Dashboard button.

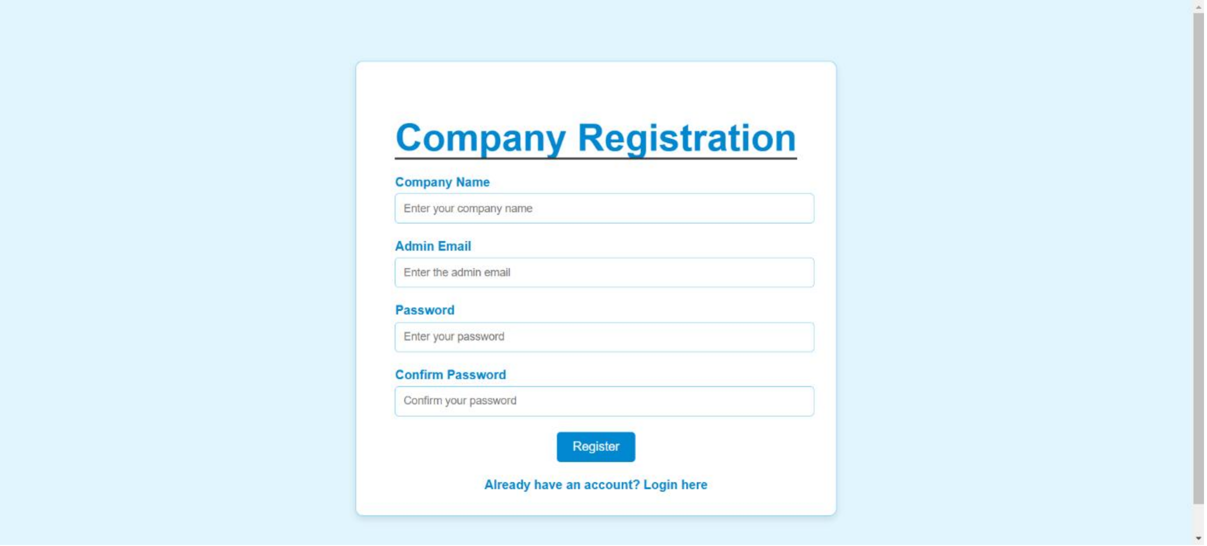
Home Screen Demo Site



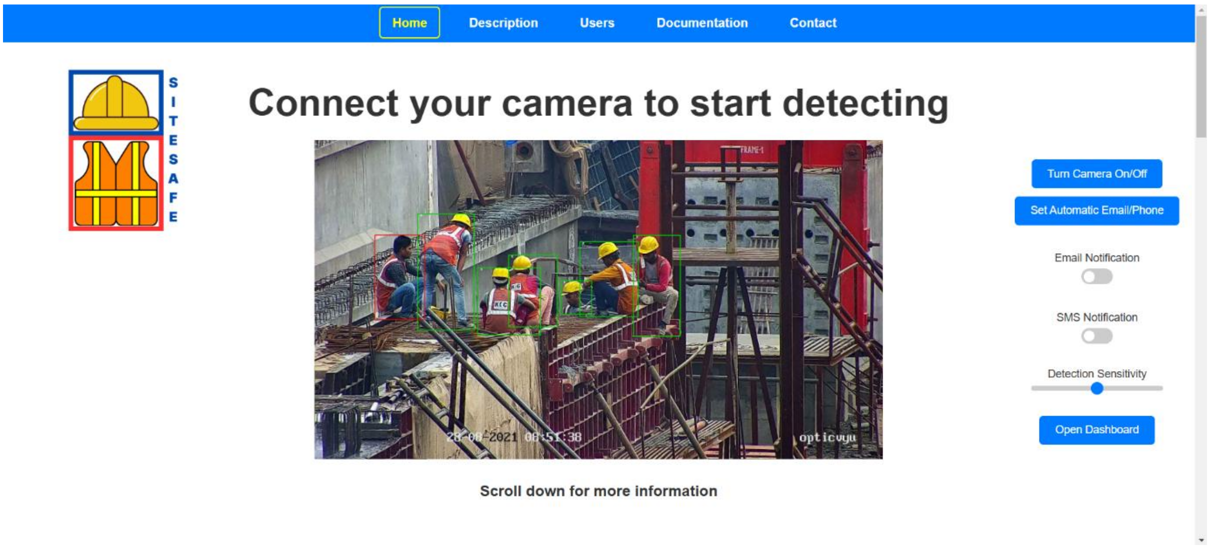
Company Login Page



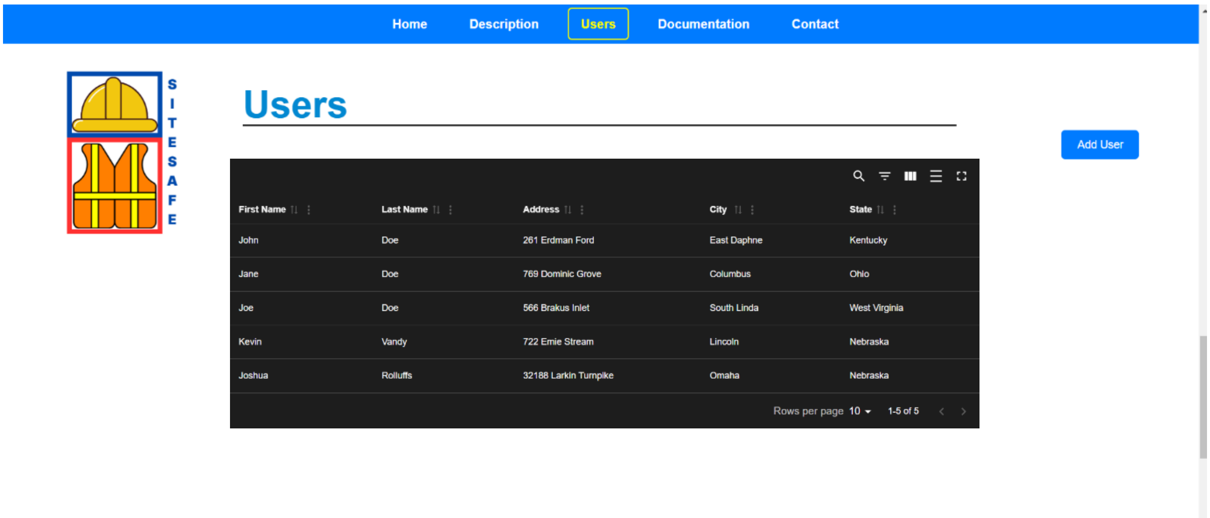
Company Registration Page



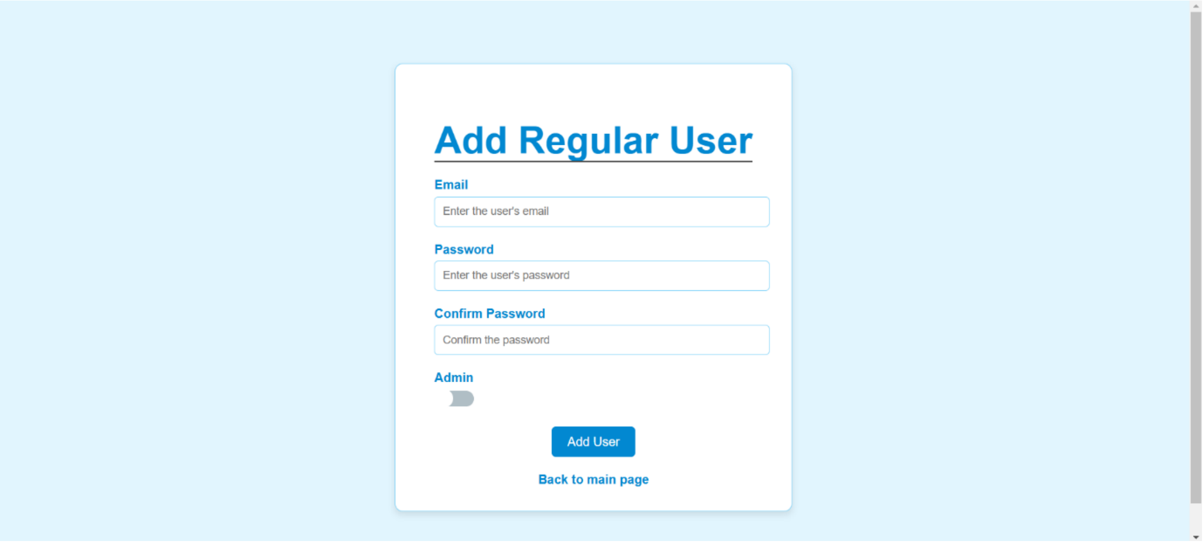
Home Screen Company Site Admin



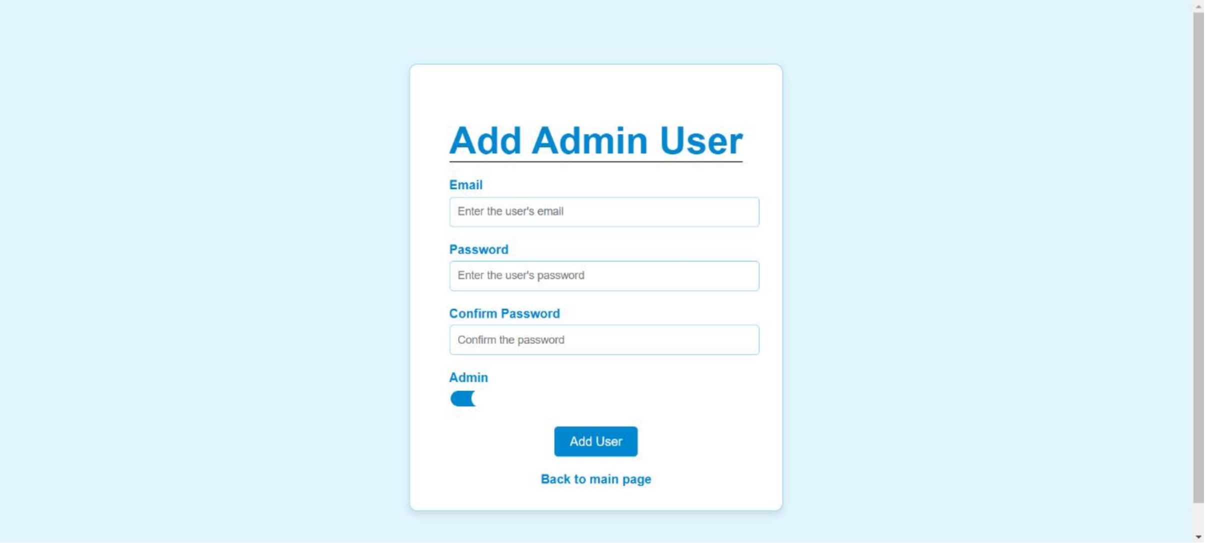
Users Screen Company Site Admin



Add Regular User



Add Admin User



Possible Dashboard Design

Database

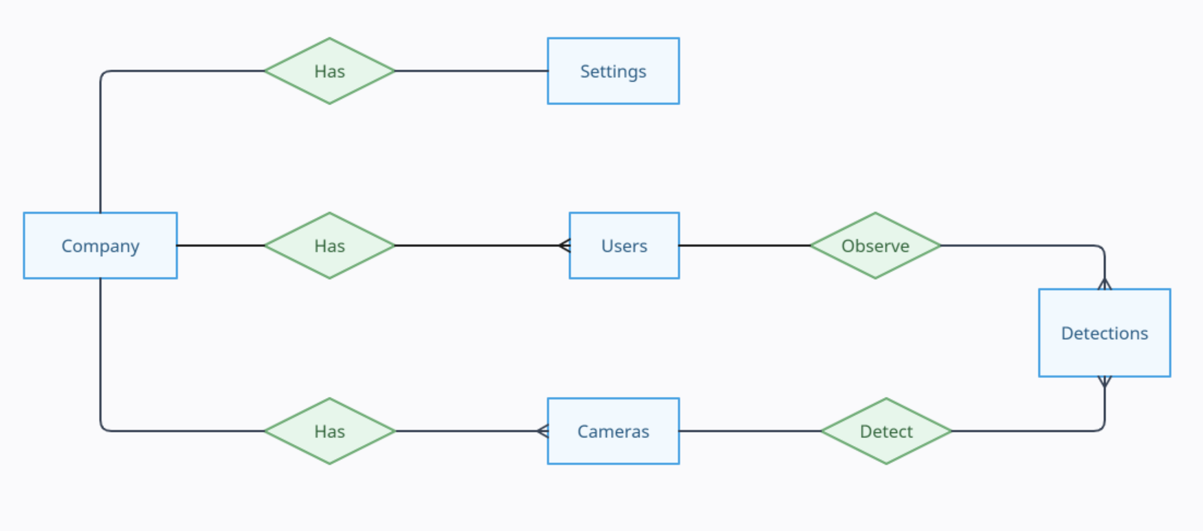
Overview

The database is a critical component of the Construction Safety Monitoring System. It will store key information related to video stream monitoring, object detection results, and user interactions with the system.

MySQL was selected as the database solution for the Construction Safety Monitoring System due to its robust performance, reliability, and widespread use in the industry. MySQL offers high scalability, ease of integration with Python and other web technologies, and strong support for complex queries, which are essential for storing and analysing real-time detection data. Additionally, MySQL's support for Atomicity, Consistency, Isolation, and Durability (ACID) compliance ensures data consistency and integrity, making it an ideal choice for managing critical safety-related information in the system.

The database will facilitate the tracking of safety violations, generate detailed dashboards, and support real-time alerts. Below is the Entity Relationship Diagram (ERD) that reveals the database layout and key directions. Followed by a section that outlines the proposed table structure.

Entity Relationship Diagram (ERD)



Database Tables

The system will be divided into several core tables, each responsible for storing a specific type of data:

Incident Detection Logs Table

This table will store details of each object incident event, including the detection confidence, type and the binary image. It is essential for tracking the results of each incident detection and providing context for alerts.

Column Name	Data Type	Description
detection_id	INT (Primary Key)	A unique identifier for each detection indicent event.
timestamp	DATETIME	The date and time when the detection event was recorded.
image_data	BLOB	Binary data of the image
confidence	FLOAT	The confidence score of the detection (likelihood of detection being accurate).
type	VARCHAR	The type of detection. No helmet or no high-vis jacket or both.
video_camera	INT	Camera number for registered users
user_id	INT	A reference to the user who triggered the detection event (if registered).

Users Table

This table will store information about system users, including their roles (admin or regular user), and password for log in. This information will be useful for managing system permissions and tracking user-specific actions.

Column Name	Data Type	Description
user_id	INT (Primary Key)	A unique identifier for each user.
username	VARCHAR	The username for the user.
email	VARCHAR	The email address for the user, used for notifications.
role	ENUM('admin', 'user')	The role assigned to the user (admin or user).
password_hash	VARCHAR	The hashed password for user authentication.

Settings Table

This table will store user-specific settings and preferences, including their notification preferences (email and SMS), confidence threshold for detection alerts, and contact details for notifications. It will allow for customised user configurations while maintaining a centralised location for managing user-specific settings such as notification methods and thresholds. This information will be crucial for personalising user experience and managing alert delivery preferences.

	A	B	C
1	**Column Name**	**Data Type**	**Description**
2	user_id	INT (Primary Key, Foreign Key)	The unique identifier for the user (linked to the **Users** table).
3	confidence_threshold	FLOAT	The confidence threshold for triggering notifications (e.g., 0.80 for 80%).
4	email_notifications	BOOLEAN	Whether the user has opted in for email notifications.
5	sms_notifications	BOOLEAN	Whether the user has opted in for SMS notifications.
6	notification_email	VARCHAR	The email address for receiving notifications
7	notification_phone	VARCHAR	The phone number for receiving SMS notifications

Dashboard Table

The Dashboard Table will store key performance data related to safety incidents, providing a comprehensive overview of site safety metrics. It will track the total number of incidents, including those related to safety gear violations, over different time frames last year, last month, and last week. The table will also capture specific incident counts, such as the number of occurrences where workers were detected without helmets or high-visibility jackets. This data will be used to generate insights and visualisations on the dashboard, enabling users to assess trends and identify areas that require attention in maintaining safety standards.

Column Name	Data Type	Description
dashboard_id	INT (Primary Key)	A unique identifier for each dashboard record.
user_id	INT (Foreign Key)	A reference to the user_id from the Users Table, linking the dashboard data to a specific user.
year_incidents	INT	Total number of incidents recorded in the last year.
month_incidents	INT	Total number of incidents recorded in the last month.
week_incidents	INT	Total number of incidents recorded in the last week.
no_helmet_count	INT	Total number of incidents where no helmet was detected.
no_high_vis_count	INT	Total number of incidents where no high-visibility jacket was detected.

Pseudo Code for Critical Algorithms

Training YOLO11 model

Single-GPU and CPU Training Example

Device is determined automatically. If a GPU is available then it will be used, otherwise training will start on CPU.

Python CLI

```python
from ultralytics import YOLO

# Load a model
model = YOLO("yolo11n.yaml")  # build a new model from YAML
model = YOLO("yolo11n.pt")  # load a pretrained model (recommended for training)
model = YOLO("yolo11n.yaml").load("yolo11n.pt")  # build from YAML and transfer weights

# Train the model
results = model.train(data="coco8.yaml", epochs=100, imgsz=640)
```

Fig. 5 YOLOv11 Training

Running inference on images with the model

1. Import necessary libraries (YOLO, OpenCV).

2. Load the YOLOv11 model using the specified weights file.

3. Define a dictionary with specific colors for each class (person, helmet, noHelmet, vest).

4. Open the video source (webcam or video file).

5. Check if the video source is opened successfully.

   - If not, print an error message and exit.

6. Start a loop to process video frames:

   a. Read a frame from the video source.

   b. If the frame cannot be read, print a message and exit the loop.

   c. Run the YOLOv11 model on the frame to get detections.

   d. Extract bounding box data from the detection results.

   e. Loop through each detected object:

      i. Extract bounding box coordinates, confidence score, and class ID.

      ii. Convert confidence score to float.

      iii. If confidence is greater than 0.6:

- Convert coordinates to integer values.

- Get the class name using the class ID.

- Assign a specific color to the class (default to white if not mapped).

- Draw a bounding box around the detected object.

- Display the class name and confidence score as text on the frame.

f. Resize the frame to a specified width and height.

g. Display the processed frame in a window.

h. Check if the 'q' key is pressed.

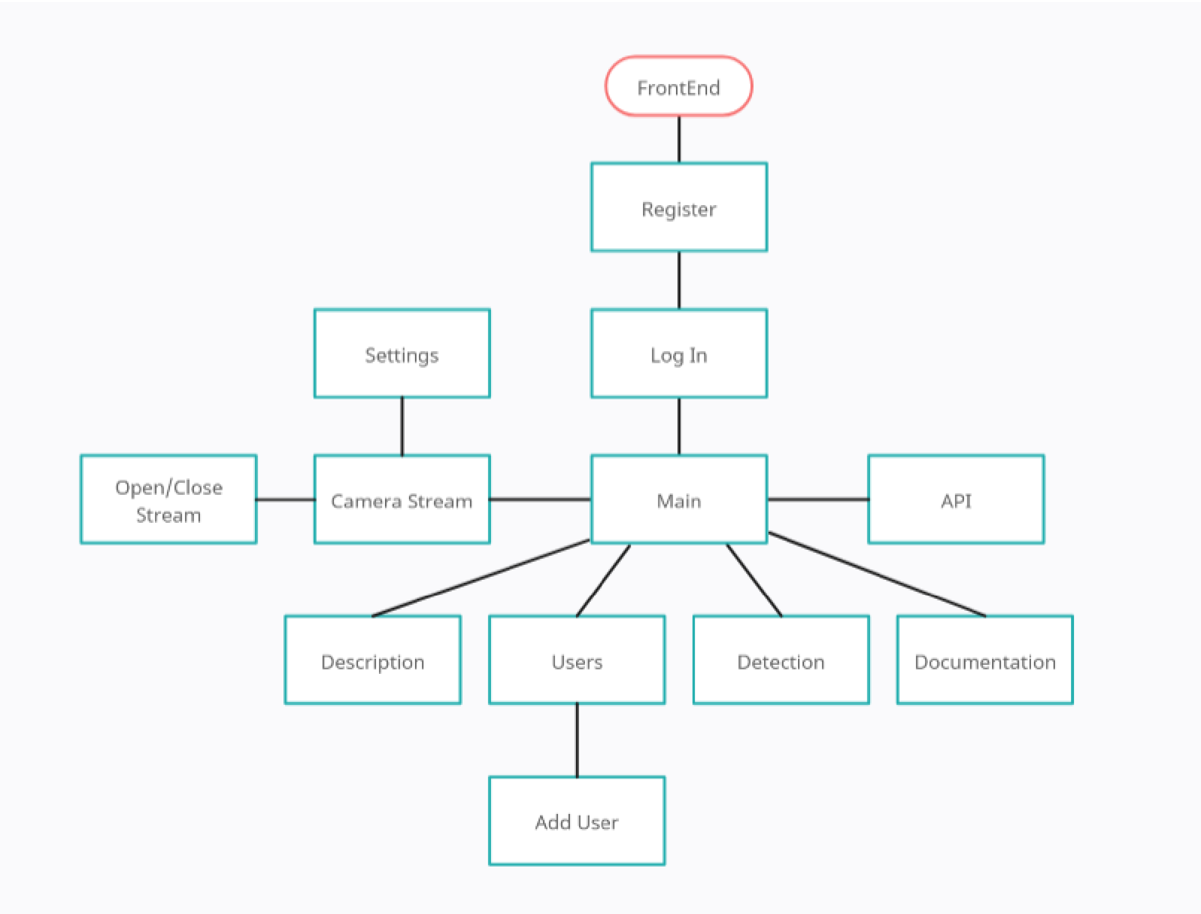
- If pressed, break the loop.

7. Release the video capture object and close all display windows.

Adding a detection row to the database if the last 8/10 detections contain a noHelmet object with a dequeue

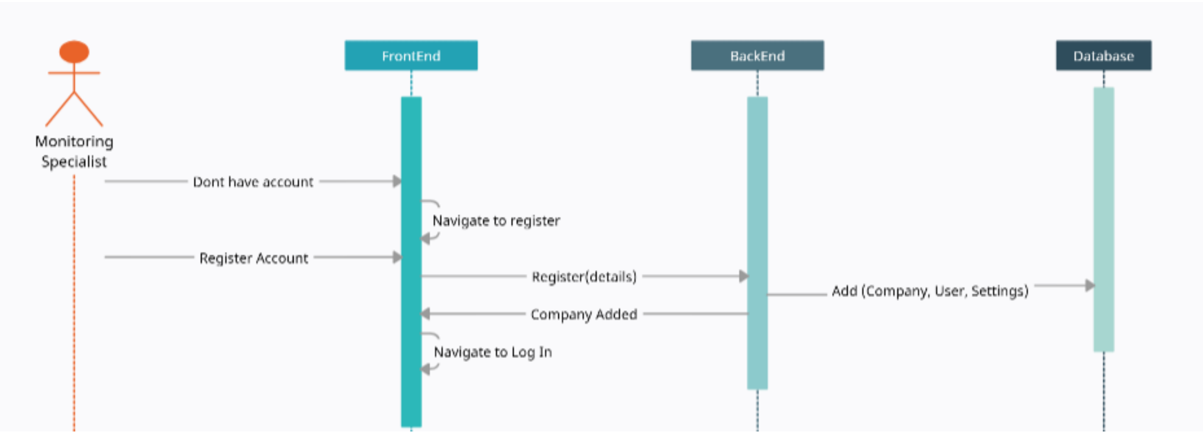
1. Listen for incoming frame data from WebSocket.
2. If valid image data is received:
   - Decode the image from base64.
   - Convert byte data into an OpenCV image.
3. Run YOLO detection on the image.
   - Check if any detection is "NoHelmet."
   - If "NoHelmet" is found, set a flag to True.
4. Store the detection result in a frame history list.
   - Count occurrences of "NoHelmet" in the last 10 frames.
5. If at least 8 out of 10 frames detect "NoHelmet":
   - Check if 30 seconds have passed since the last incident.
   - If cooldown time has passed:
     - Log the detection in the database.
     - Reset the cooldown timer.
   - Otherwise, print remaining cooldown time.

Component Diagram

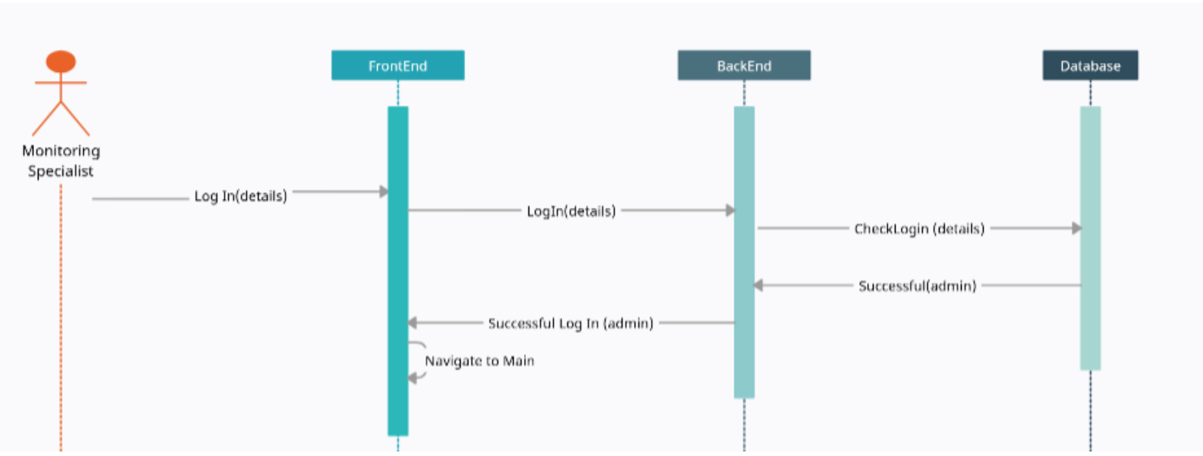


Sequence Diagrams

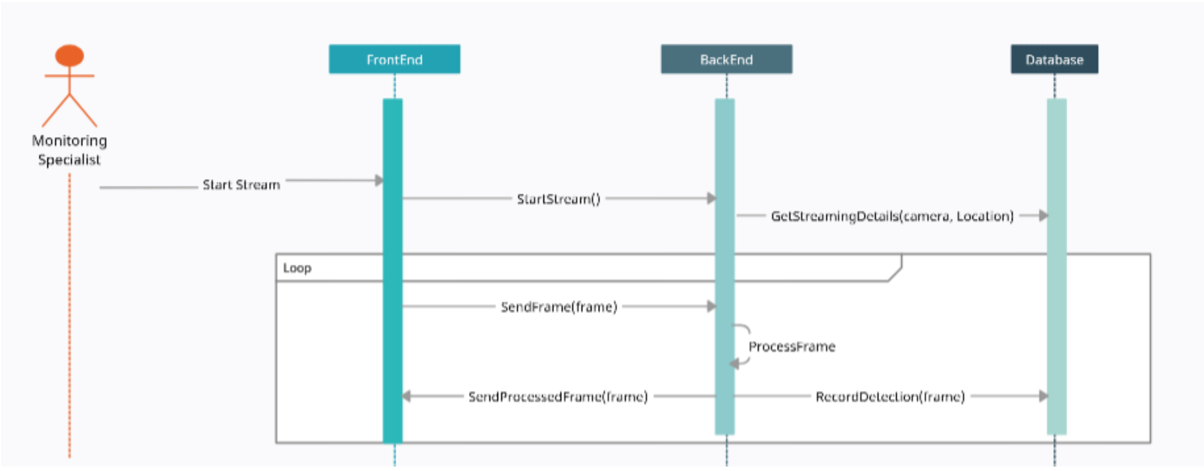
Register



Log In



Start Camera and Detect



References

Figures

Fig 1. SETU Logo: SETU Logo

Fig 2. SiteSafe Logo Creation Tool: https://www.canva.com/design/play?type=TAB7AVEOUWQ&category=tACZCvjI6mE&locale=en

Fig 3. No Helmet Worker: https://markhamglobal.com/products/conqor-ig200/

Fig 4. Safety Document: https://workhealthsafetyresearch.org/wp-content/uploads/2015/02/0612-045-r1-measuring-the-leading-indicators-of-ohs-a-snapshot-review-for-public-release-2.pdf?utm_source=chatgpt.com

Fig 5. YOLO Training Code: https://docs.ultralytics.com/modes/train/

General Information

1. Python: https://www.python.org
2. YOLO: https://yolo-docs.readthedocs.io/en/latest/
3. Ultralytics: https://ultralytics.com
4. OpenCV: https://opencv.org
5. Flask: https://flask.palletsprojects.com
6. React: https://reactjs.org
7. SMTP: https://tools.ietf.org/html/rfc5321
8. MySQL: https://www.mysql.com