

MICHIGAN STATE

UNIVERSITY

Project Plan Presentation

Small Object Detection Using CCTV Cameras

The Capstone Experience

Team Moii

Angela Majestic

Nathan Srivastava

Ian Valdovinos Granados

Khushi Vora

Hong Zhuang

Department of Computer Science and Engineering
Michigan State University

Fall 2023



*From Students...
...to Professionals*

Project Sponsor Overview

- International Software Company
- AI company focused on tracking & analyzing consumer behavior
- Provide businesses with valuable insights into their operations
- From understanding customer and employee behavior to implementing security systems



Project Functional Specifications

- Automated Firearm Detection
 - Utilizes ML model to automatically detect firearms in surveillance feed
- Real-time Alerts
 - Sends email or text notifications when a firearm is detected
- User-friendly Interface
 - Web interface to view surveillance feeds and check current or past detected threats
- Enhance Security
 - Provides real-time insights so clients can mitigate threats as soon as they occur

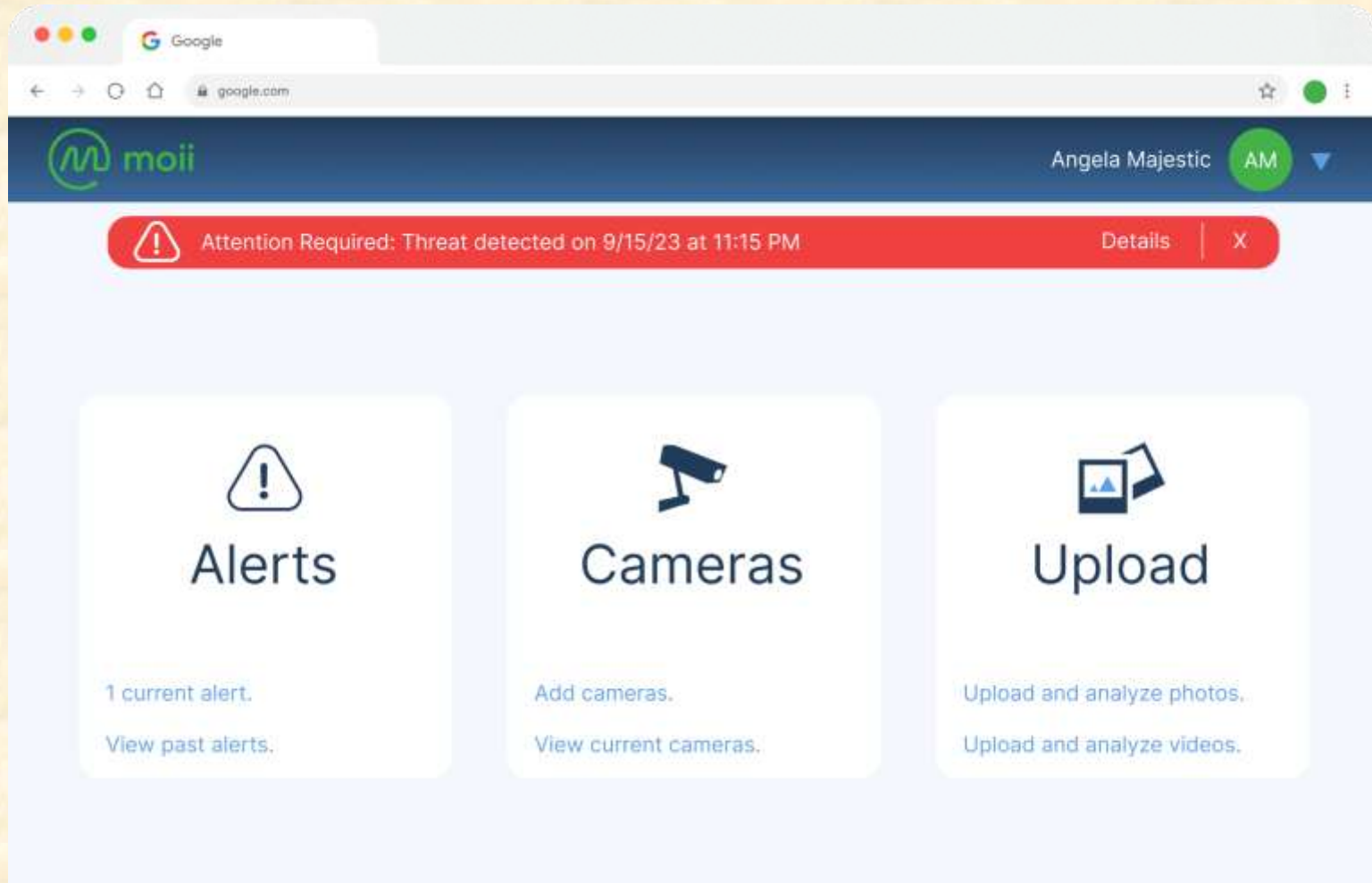


Project Design Specifications

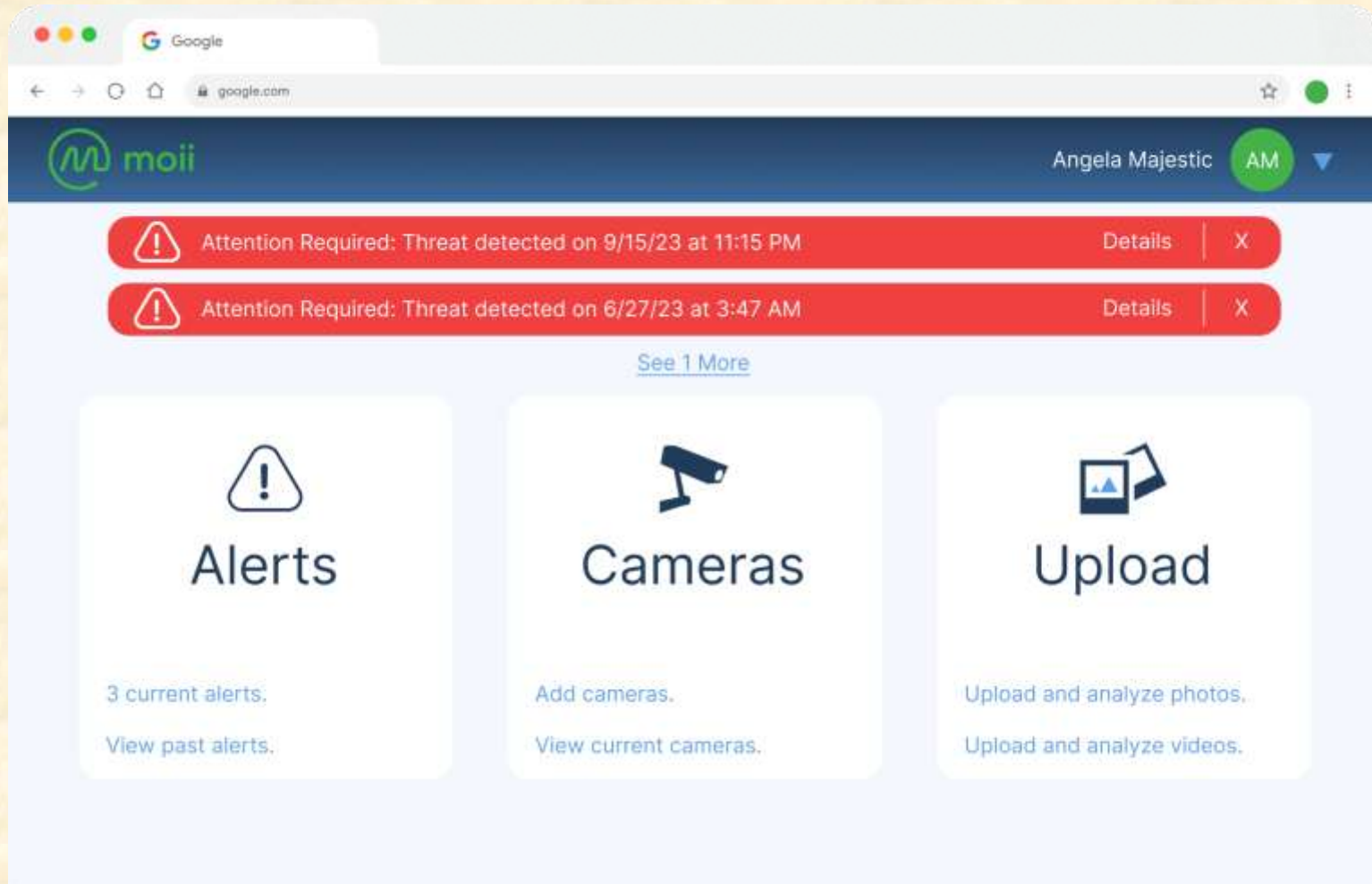
- Utilizes an AI model for small object detection to identify guns in CCTV footage
- Real-time detection on CCTV feeds from uploaded RTSP links
- Send real-time alerts when a gun is detected
- Modular frontend & cloud-based hosting
- Continuous monitoring



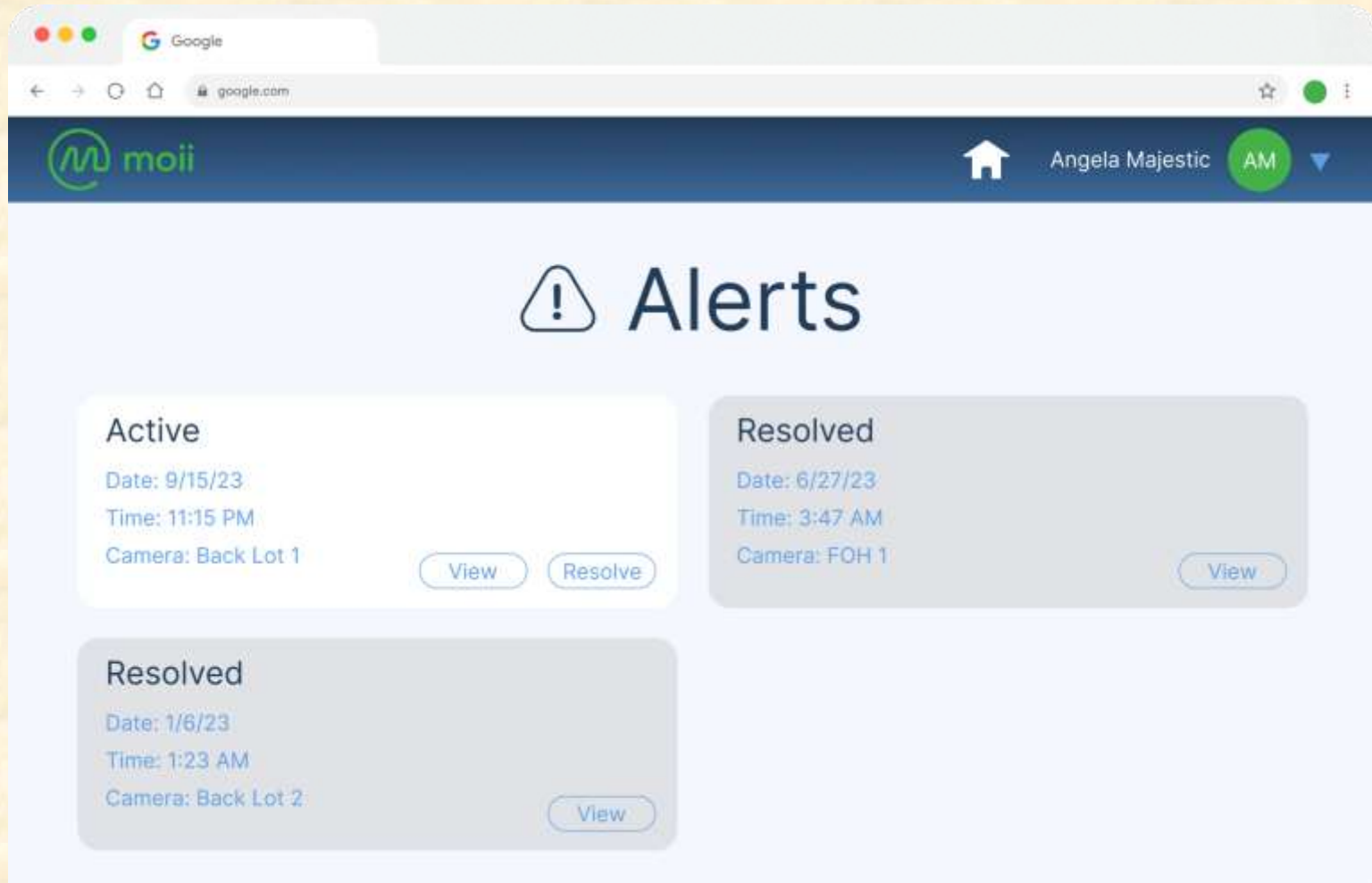
Screen Mockup: Home Dashboard



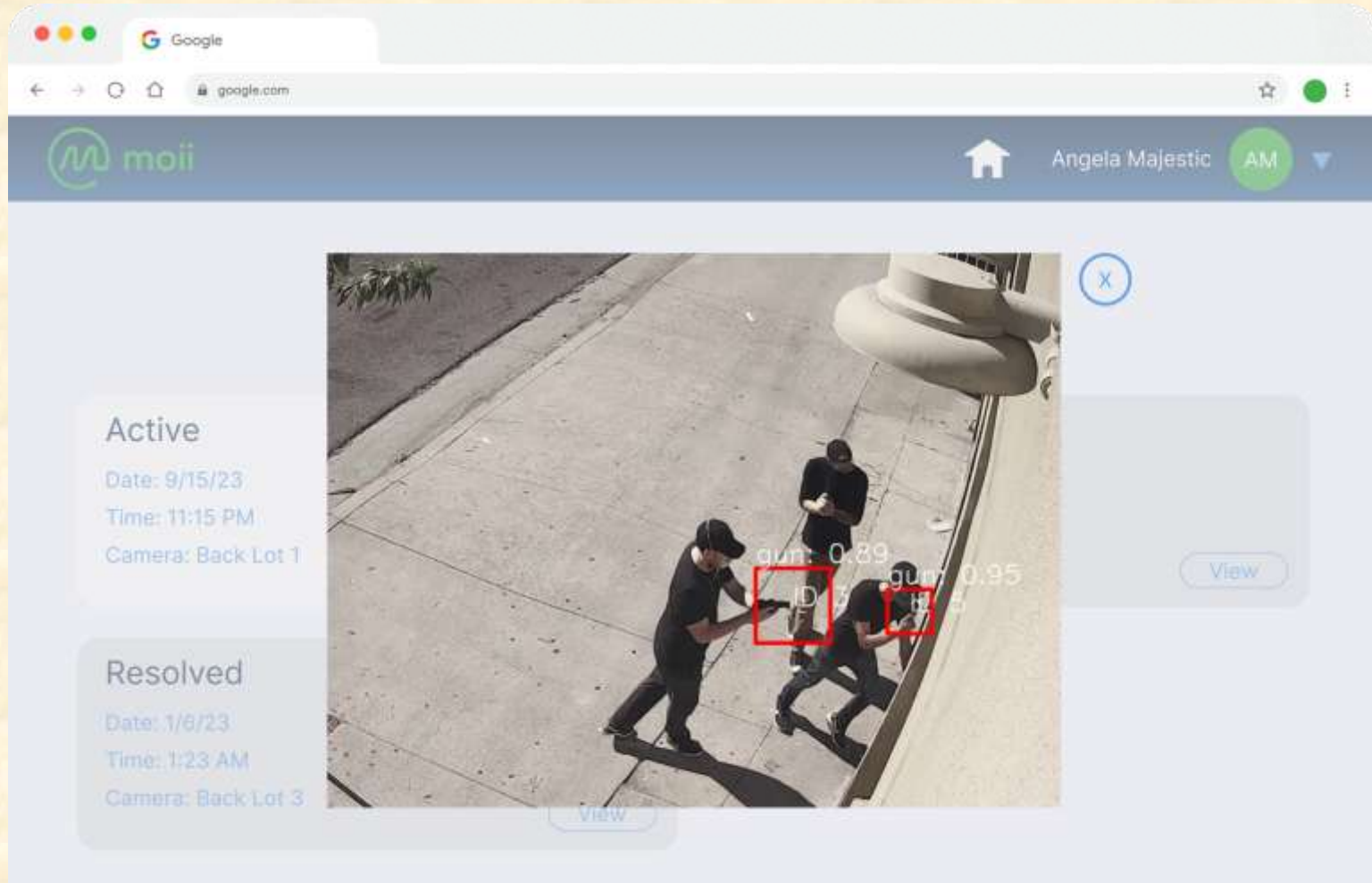
Screen Mockup: Alerts Stack



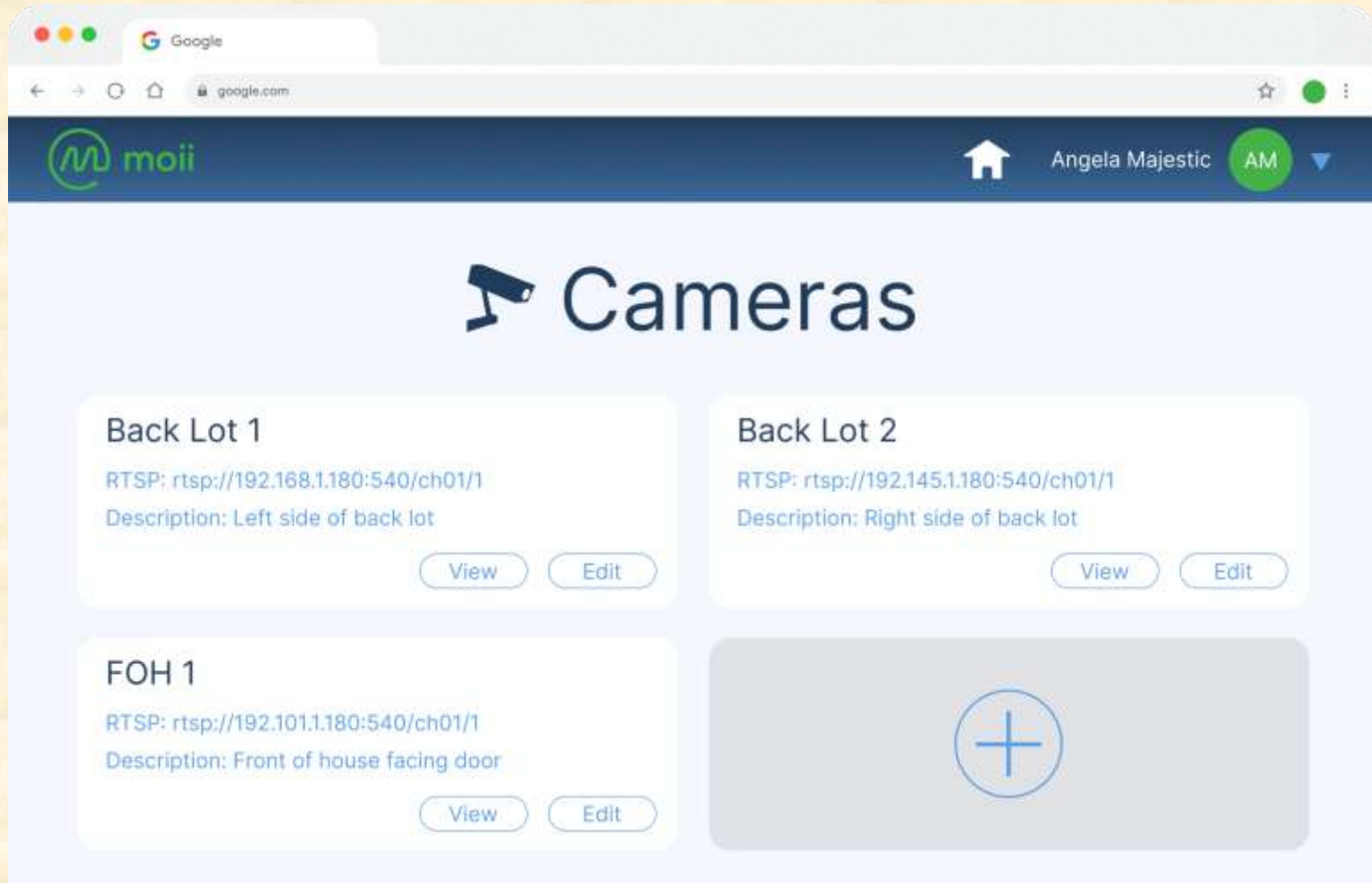
Screen Mockup: Alert Dashboard



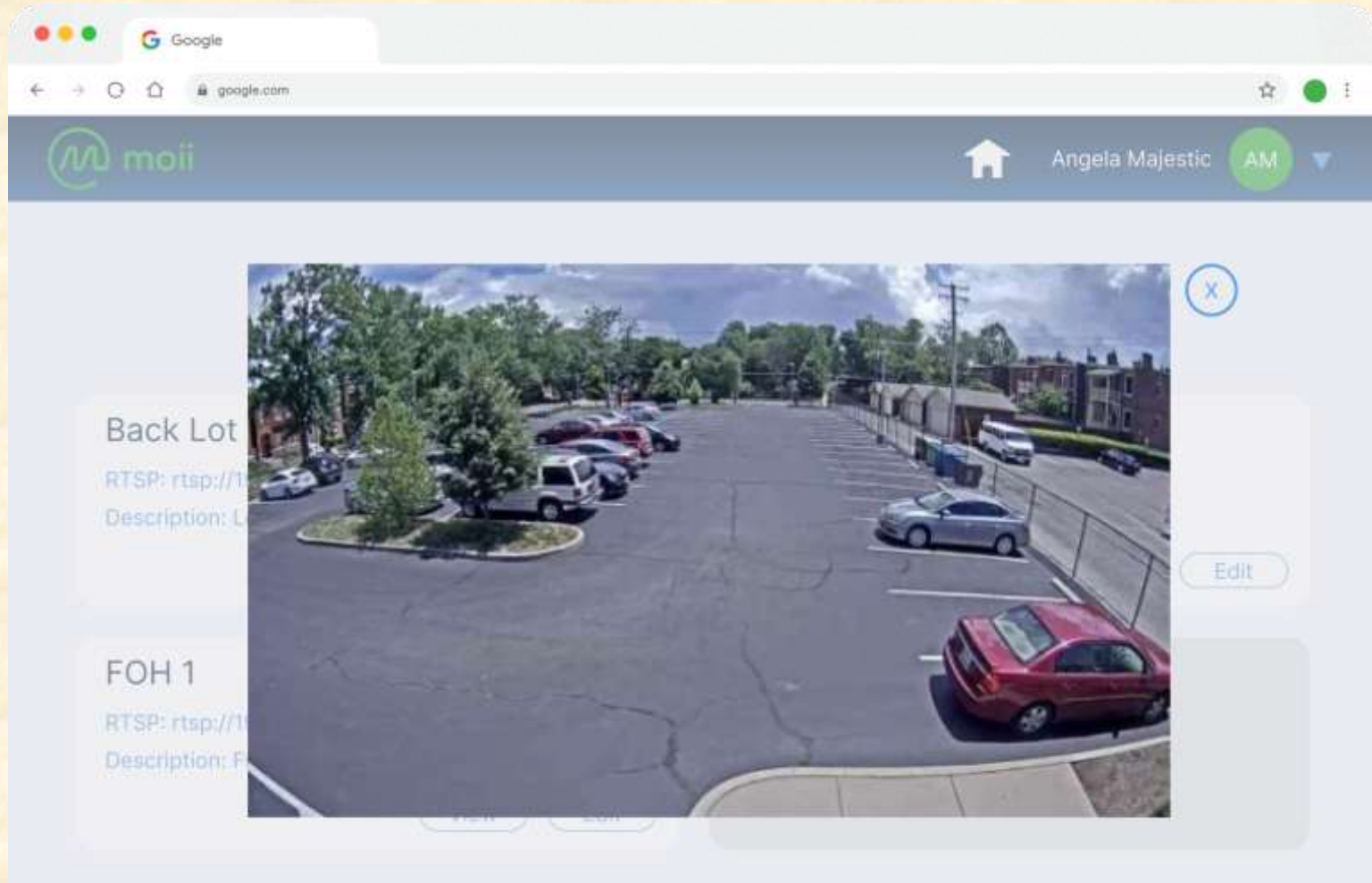
Screen Mockup: View Alert



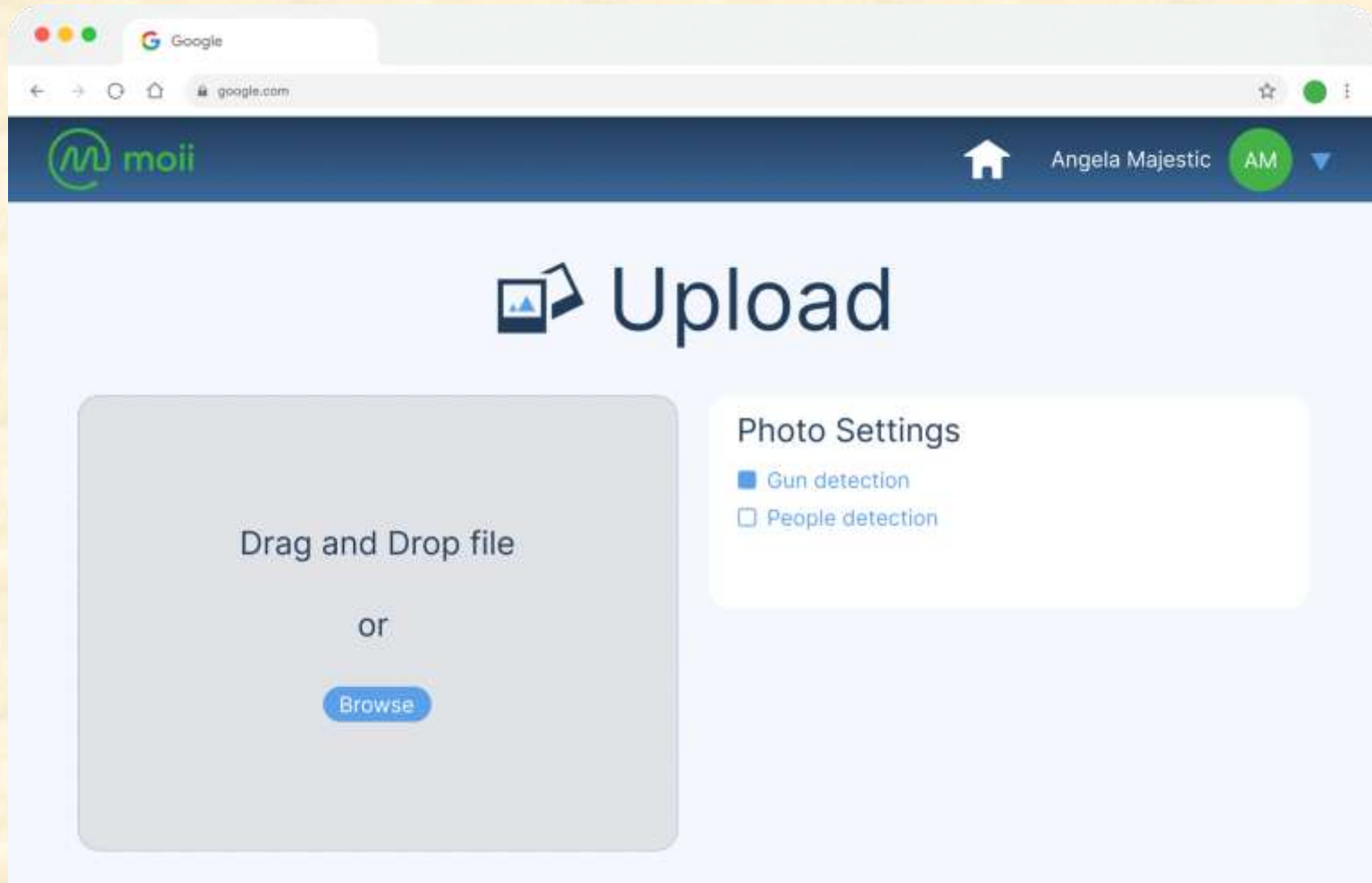
Screen Mockup: Camera Dashboard



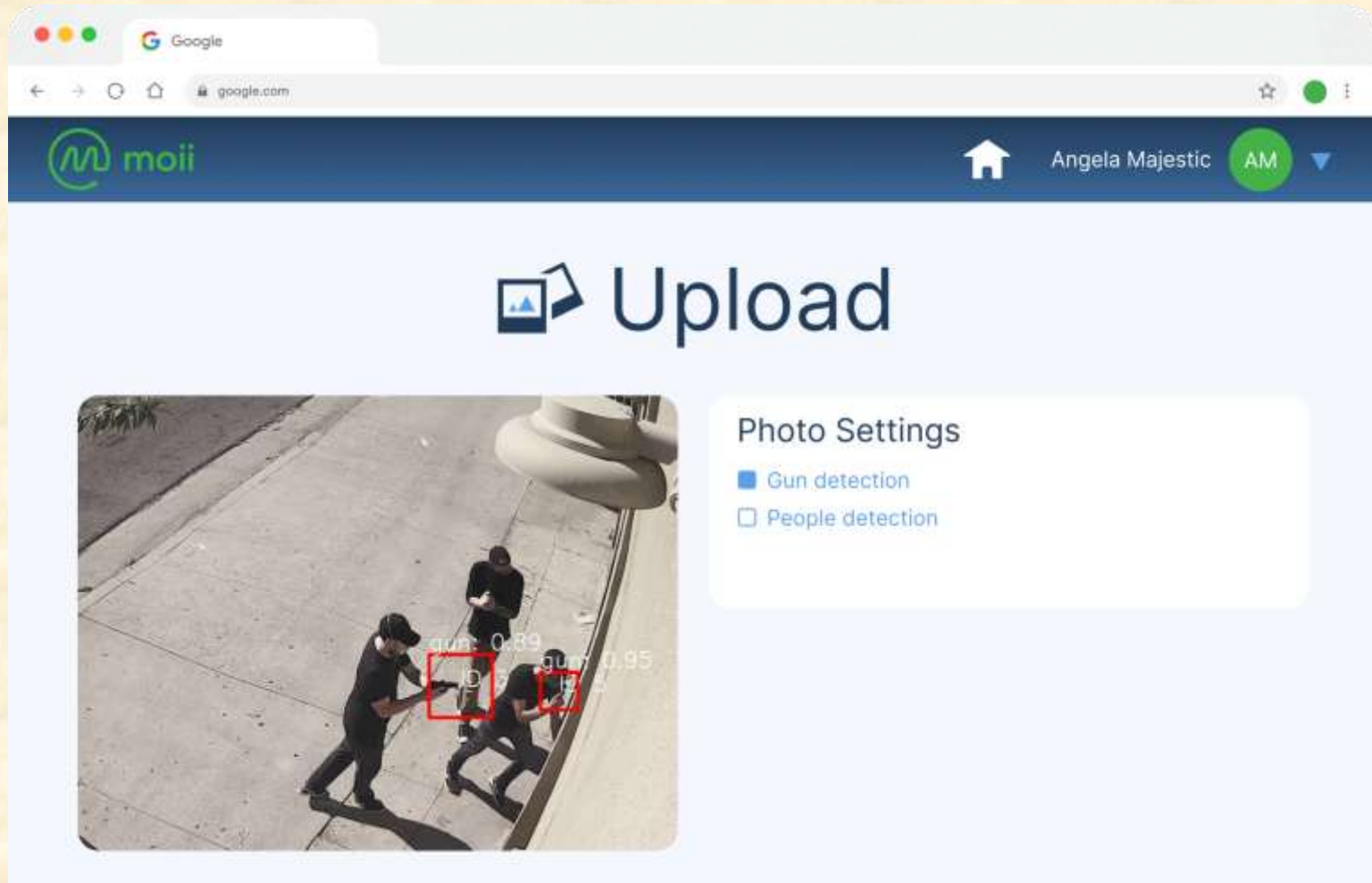
Screen Mockup: View Camera Feed



Screen Mockup: Upload Dashboard



Screen Mockup: Uploaded Photo

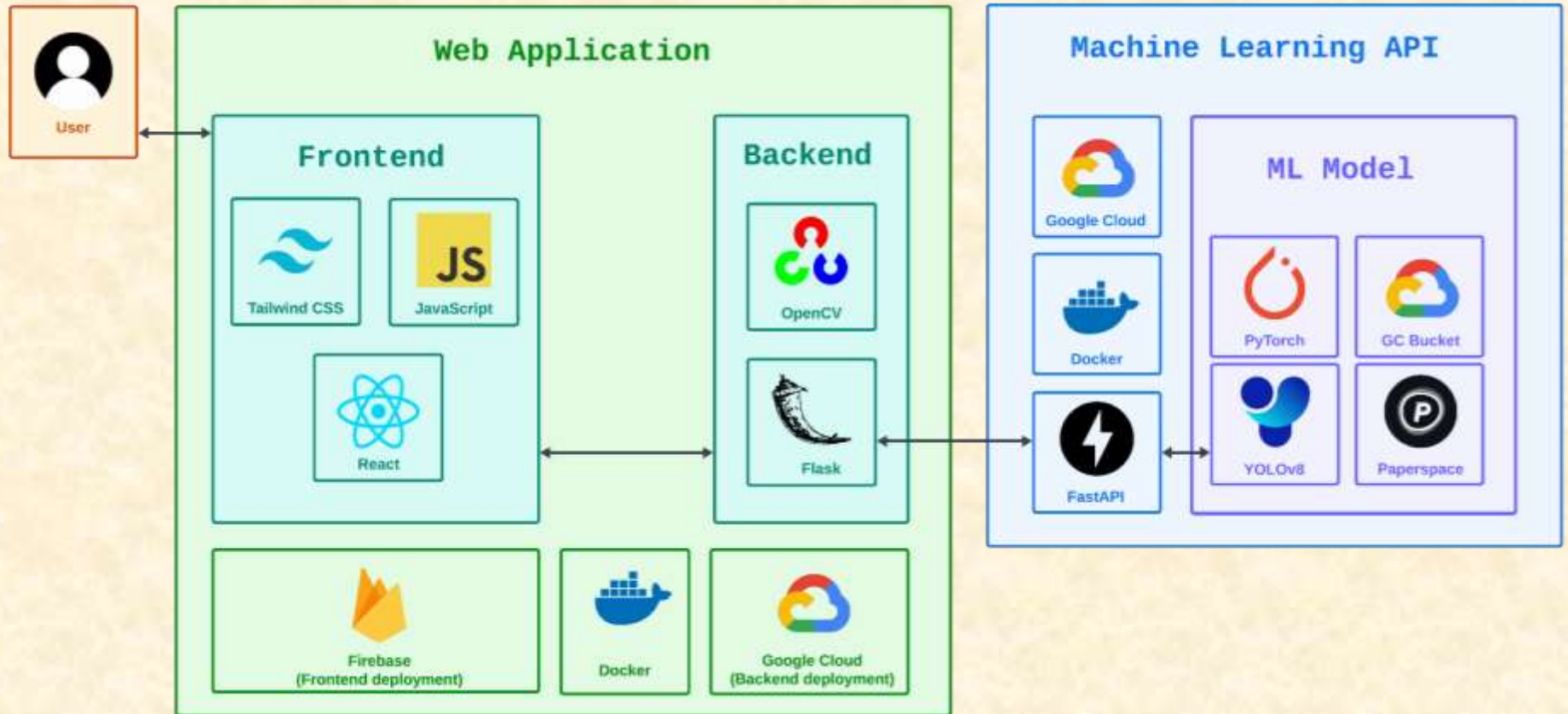


Project Technical Specifications

- **Web Application**
 - **Frontend**
 - React (HTML, JavaScript, & CSS)
 - Deployed on Firebase
 - User can access CCTV camera info and threat alerts
 - **Backend**
 - Flask server framework
 - Deployed on Google Cloud Platform
 - Communicates with the Machine Learning API
 - Sends JSON files for predictions to the ML API (breaks down feed)
- **Machine Learning API**
 - API communicates between the model & web app
 - Deployed on Google Cloud Platform
 - **Machine Learning Model**
 - YOLOv8 model takes JSON file input → returns bounding box
 - Trained using the SAHI method
 - Using PyTorch ML library for training
 - Training data extracted from Google Cloud Bucket



Project System Architecture



Project System Components

- Software Platforms / Technologies
 - Web Application
 - HTML/CSS/JS
 - React
 - Firebase
 - Flask
 - OpenCV
 - Machine Learning API
 - Google Cloud Platform
 - FastAPI
 - Paperspace
 - PyTorch
 - YOLOv8
 - SAHI



Project Risks

- Fetch CCTV camera feed into Flask application
 - Description: Get real-time feed from the CCTV camera.
 - Mitigation: Research OpenCV library, which has a real-time processing feature.
- Break CCTV camera feed in frames
 - Description: Web app should break the real-time CCTV feed up into frames that will be sent to the ML Model API. This should be done efficiently so there is little delay in notifications.
 - Mitigation: OpenCV allows programmers to get individual frames from a video file or video stream.
- Long ML model training time
 - Description: Training a ML model takes a lot of computational resources. Can take several hours if the correct hardware is not used.
 - Mitigation: Will use Paperspace, a cloud computing platform, to rent GPU power.
- Balance ML Model accuracy and speed
 - Description: Important to balance how well the model can predict with how fast it can make the prediction.
 - Mitigation: Use a YOLO model and the SAHI model training method.



Questions?

?

?

?

?

?

?

?

?

?

