**SentryVision**

**Software Requirement Specification(SRS)**

# Introduction:

Experience the future of security with Sentry Vision: an AI-driven CCTV app detecting criminal incidents in real-time, sending instant notifications for swift response, and ensuring proactive surveillance at your fingertips. Stay ahead of threats effortlessly.

## Purpose:

The purpose of the Sentry Vision project is to leverage advanced artificial intelligence technology to

enhance security surveillance systems. By deploying sophisticated algorithms, the project aims to

detect and alert authorities or stakeholders about criminal incidents captured by CCTV cameras in

real-time. This proactive approach to monitoring aims to reduce response times to security threats

enhance overall safety, and provide peace of mind to users by effectively preventing and

managing security incidents.

### From User’s and Security Organisation’s point of view:

* + - Sentry Vision offers users enhanced security through real-time surveillance, safeguarding both their safety and property
    - Sentry Vision empowers users with customizable security settings, seamless access to CCTV footage, and personalized alerts, fostering a user-centric approach to security control.
    - Sentry Vision enhances incident response for security organizations by swiftly detecting and notifying them of security threats, enabling effective deployment of resources.

### From Research and Development point of view:

* + - The purpose of Sentry Vision from a research and development perspective is to advance the state-of-the-art in AI and computer vision technologies applied to surveillance systems.
    - The project aims to provide complete software for security enhancement .

So, this project is an attempt to overcome the security issues and provides a trustworthy and easy to use product .

## Scope:

The RAD-CFS shall be capable of processing real-time video data from CCTV cameras, extracting relevant features, training a machine learning model for anomaly detection, integrating the model into a real-time video processing system, implementing a system for recording anomaly metadata, and developing an alert system for notifications.

***In Scope:***

* Real-time video data processing from CCTV cameras.
* Feature extraction for anomaly detection.
* Machine learning model training and adaptation.
* Integration with a real-time video processing system.
* Metadata recording for detected anomalies.
* Alert system for notifications.

***Out of Scope:***

* Face recognition or biometric identification techniques.
* Audio data analysis.
* Storage and management of raw CCTV footage data.
* Integration with other surveillance or security systems.
* On-site installation or maintenance of hardware.
* End-user training or support.
* Customization or modification beyond specified requirements.

## Definitions, Acronyms, and Abbreviations:

* **Anomaly:** Unusual or suspicious event or activity in the CCTV footage, such as a car accident, explosion, or conflict.
* **CCTV:** Closed-circuit television, a video surveillance system used for monitoring a particular area.
* **RAD-CFS**: Real-Time Anomaly Detection in CCTV Footage System.
* **ML:** Machine learning, a type of artificial intelligence that allows systems to learn and improve from experience.

## Overview:

The rest of this SRS is organized as follows: Section 2 gives an overall description of the software. It gives what level of proficiency is expected of the user, some general constraints while making the software and some assumptions and dependencies that are assumed. Section 3 gives specific requirements which the software is expected to deliver. Functional requirements are given by various use cases. Some performance requirements and design constraints are also given.

# Overall Description:

## Product Perspective:

The product perspective for Real-Time Video Monitoring and Anomaly Detection System focuses on providing a user-centric solution that caters to the security and safety needs of organizations and individuals. The users of the system would primarily be security professionals (such as security guards or police officers), business owners, managers, or homeowners who require a reliable system to monitor and respond to incidents in real-time. Key product features include robust authentication and authorization mechanisms to ensure secure access, AI/ML models for real-time video monitoring and anomaly detection, and integration capabilities with current security setups. These features contribute to the system's overall effectiveness and usability, allowing users to maintain a safe and secure environment.

## Product Functions:

"SentryVision" supports the following use cases:

| Use cases | Description of use cases |
| --- | --- |
| System: | |
| Capture Video footage | 1.This primary use case allows this system to capture real-time videos from surveillance cameras  2.Include: Adjust camera settings (adjusting resolution,zoom level,etc) |
| Detect anomalies | 1.This primary use case enables the system to detect accidents,thefts,explosions or any other anomalies captured in the video footage.  2.Extend: Classify anomaly type(identifying specific types of anomalies like theft,accident,etc) |
| Generate incident reports | 1.This will basically generate a report about the classified anomaly  2.Extend: Generate incident reports(creating detailed reports about recorded incidents) |
| Security Officer: |  |
| Generate incident reports | 1.This will basically generate a report about the classified anomaly  2.Extend: Generate incident reports(creating detailed reports about recorded incidents) |
| Notify User | 1.The primary use case notifies the authorized user via email ,SMS or push notifications when anomalies are detected.  2.Extend : Customize notifications(tailoring notification content,frequency,etc.) |
| Secure Login | Allows secure login for the system and also for the users. |
| User: | |
| User Authorisation | 1.This primary use case authorizes the user’s access level to different features and data within the system based on their role and permissions.  2.Extend:Manage User Permissions (modifying user access rights). |
| Access the accident reports | 1.This use case will be accessible for viewing the accident reports. |
| Secure Login | Allows secure login for the system and also for the users. |

## User Characteristics:

The user should be familiar with the operation of Web Applications.

## Principal Actors:

The three principal actors in "SentryVision" are “System”, “End User” and “Security Officer”.

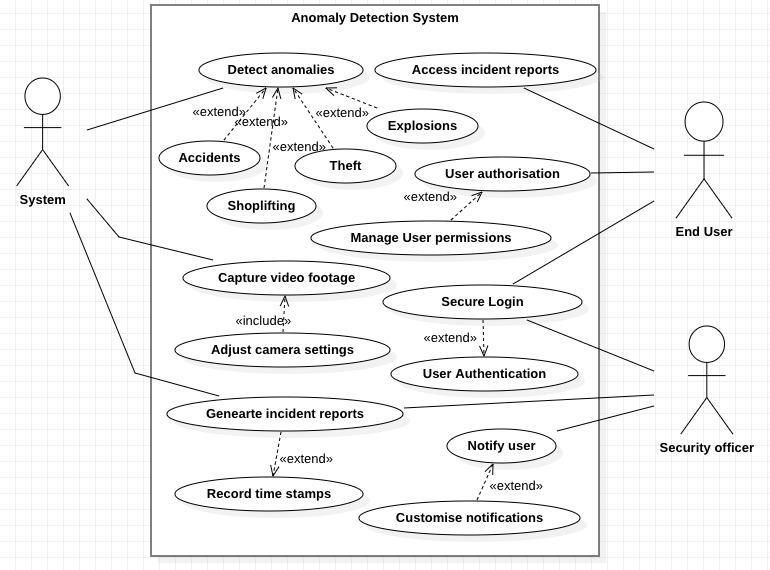
## General Constraints:

* Working of "SentryVision" requires an active Internet connection.
* The system relies on the assumption that the anomaly that has occured is of sufficient time duration for meaningful anomaly detection.

## 2.6 Assumptions and Dependencies:

* Working of "SentryVision" is dependent on the availability of Internet connection.
* The system assumes the availability of sufficient resources (e.g., computing power, storage) to perform anomaly detection tasks.

APPENDIX A: USE CASE DIAGRAM



Use case diagram for anomaly detection system.

## 2.7 Functional Requirements:

We describe the functional requirements by giving various use cases.

## Use Case 1:

**Name:** Capture Video footage

**Summary**: Allows this system to capture real-time videos from surveillance cameras

**Actors**: System

### Pre-conditions:

Internet connectivity.

### Main success scenario:

Footage gets recorded.

### Extension: NIL

### Post-condition: NIL

## Use Case 2:

**Name**: Detect anomalies

**Summary**: Enables the system to detect accidents,thefts,explosions or any other anomalies captured in the video footage.

**Actors:** System

### Pre-conditions:

* Internet Connectivity.
* ML Model for Anomaly Detection.

### Main success scenario:

`Anomaly Detected

**Extension:**  Classify anomaly type(identifying specific types of anomalies like theft,accident,etc)

**Post-condition:** NIL

## Use Case 3:

**Name:** Generate incident reports

**Summary:** generate a report about the classified anomaly

**Actors:** System

### Pre-conditions:

* Internet connectivity.
* Some anomaly has to be detected.

### Main success scenario:

Report generated successfully.

### Extension: Generate incident reports(creating detailed reports about recorded incidents)

### Post-condition: NIL

## Use Case 4:

**Name**: Notify User

**Summary**: Notifies the authorized user via email ,SMS or push notifications when anomalies are detected.

**Actors**: Security Officer

### Pre-conditions:

* Internet connectivity.
* Anomaly has to be detected and report has to be pre generated.

### Main success scenario:

Notification sent successfully.

### Extensions: Customize notifications(tailoring notification content,frequency,etc.)

### Post-conditions: NIL

## Use Case 5:

**Name:** Secure Login

**Summary**: Allows secure login for the system and also for the users.

**Actors:** Security Officer

### Pre - Conditions:

Login ID and Password available for the authorised user.

### Main Success scenario:

User gets successfully logged in.

### Extension: Change password for a particular username.

### Post condition: NIL

## Use Case 6:

**Name:** User Authorisation

**Summary:** Authorizes the user’s access level to different features and data within the system based on their role and permissions.

**Actors:** End user

### Pre - Conditions:

Login ID and Password available for the authorised user.

### Main Success Scenario:

User gets successfully logged in.

### Extension: User Permissions (modifying user access rights).

### Post condition: NIL

## Use Case 7:

**Name:** Access the accident reports

**Summary:** Accessible for viewing the accident reports.

**Actors:** End user

### Pre - Conditions: .

User is already logged in and has access rights.

### Main Success Scenario:

Incident report successfully accessed.

### Post condition:

User has viewed the report and decided the next course of action.

### Extension: NIL

### Post condition: NIL

## 2.8 Non-Functional Requirements:

* Login restrictions should be in place, allowing only authorized users with specified credentials to log in.
* A notification system should enable administrators to send real-time alerts to users via various channels.
* Anomaly detection algorithms should meet defined performance benchmarks, ensuring accuracy and responsiveness.
* It should support the addition of new monitoring devices and users without requiring extensive modifications or downtime.
* Compatibility with common video streaming protocols and formats is crucial for seamless integration.

## 2.9 Hardware Requirements:

* **CPU:** A high-performance CPU with multiple cores can be beneficial for handling the computational load of video processing and machine learning tasks. Consider using CPUs with good single-threaded performance, like those from Intel or AMD.
* **GPU:** A dedicated GPU can significantly speed up machine learning tasks, especially for deep learning models. NVIDIA GPUs are commonly used in machine learning due to their CUDA platform and good support for libraries like TensorFlow and PyTorch.
* **Memory:** The system should have enough RAM to handle the processing of high-resolution video data and to store the trained machine learning models. Consider using systems with at least 16 GB of RAM, with the option to upgrade to 32 GB or more.
* **Storage**: Use fast storage devices like SSDs (Solid State Drives) for storing and retrieving video data and other relevant data. Consider using RAID (Redundant Array of Independent Disks) configurations for redundancy and reliability.
* **Network:** A stable and fast network connection is necessary for real-time video streaming and data transfer. Consider using wired connections or high-speed wireless networks for connectivity.
* **Camera System:** The CCTV system should have cameras with good resolution and low-light performance. Consider using IP cameras that can stream video data over a network.
* **Other Hardware:** Depending on the specific requirements of the system, you might need additional hardware like power backup systems, monitors, or input devices.

## 2.10 Software Requirements:

* **Operating System:** A Linux-based operating system is often preferred for machine learning applications due to its flexibility and support for open-source tools.
* **Programming Language**: Python is a popular choice for machine learning due to its libraries like TensorFlow and OpenCV.
* **Libraries and Frameworks:** Libraries like OpenCV (Open Source Computer Vision Library) can be used for computer vision tasks like feature extraction and object detection. Deep learning frameworks like TensorFlow, PyTorch, can be used for training and deploying machine learning models. Flask can be used for backend development.
* **Video Streaming and Processing:** Tools like FFmpeg can be used to handle the streaming and processing of CCTV video data.
* **Data Storage and Management:** Use databases or data stores like MySQL or MongoDB to store and manage the metadata related to anomalies, such as timestamps, locations, and types of anomalies.
* **Alert System:** Tools like SMTP (Simple Mail Transfer Protocol) or push notification services can be used to send alerts when anomalies are detected.

## 2.11 Design Constraints:

* **Data Security:** Encrypt video data and anomaly detection results to maintain the confidentiality and integrity of sensitive information.
* **Bandwidth limitations**: Consideration for network bandwidth limitations, especially in remote or bandwidth-constrained locations, must be factored into the system design.
* The system design must adhere to standard video compression formats for efficient storage and transmission.

**GROUP MEMBERS**

**(Team: CodeFreaks)**

* Prernendu Bhagat (IIT2022016)
* Mansa Mahendru (IIT2022017)
* Piyush Tiwari (IIT2022020)
* Kushal Bansal (IIT2022021)
* Agatsya Narayanan (IIT2022032)
* Harshini Bhukya (IIT2022084)