CSE 564 Project Report Number 4

Team 16

Team Member Names:

1. Aditya Pant
2. Ameya Shahu
3. Lalit Arvind Balaji
4. Pravalika Mukkiri

Table of Contents

Table of Contents

1. Requirements

1.1. Real Time Surveillance

1.1.1. Feature

- Continuous real-time surveillance of the industrial area.

1.1.2. Requirement Source

- Industrial Security Team

1.1.3. Inputs and Stimulus

- Data from drones, motion sensors, and perimeter cameras.

1.1.4. Sequence of Operations and Responses

- The industrial region is continually monitored by drones.
- Drones notify the central control system of any unauthorized motion.
- When an alarm is set off by the control system, security staff must react.

1.2. Real-time Surveillance Streaming

1.2.1. Feature

- Sharing Real-time footage with Central control system

1.2.2. Requirement Source

- Real-time Surveillance, Autonomous operation

1.2.3. Inputs and Stimulus

- Data from drone surveillance cameras

1.2.4. Sequence of Operations and Responses

- Drones should share live footage of the actions of the encroached entity while following it.

1.3. Autonomous Operation

1.3.1. Feature

- Drones should operate autonomously without requiring drone operators when needed.

1.3.2. Requirement Source

- System Architecture Team, Technical Team

1.3.3. Inputs and Stimulus

- Video data from the drones, autonomous drone algorithms for encroachment detection

1.3.4. Sequence of Operations and Responses

- Drones fly on their own, using algorithms to detect violations and according to predetermined flying patterns.
- They are able to monitor intruders by making decisions in real time.

1.4. Operation types

1.4.1. Feature

- Operation types in autonomous operation

1.4.2. Requirement Source

- Autonomous operation

1.4.3. Inputs and Stimulus

- Instruction/ requests from Central control system

1.4.4. Sequence of Operations and Responses

- Scheduled monitoring -The drones are supposed to fly over the whole industrial zone at regular intervals.
- Demand monitoring - For extra security, the drones can be asked to keep an eye on a specific area within the industrial zone.

1.5. Data Processing in Realtime

1.5.1. Feature

- Real-time data processing with reduced latency

1.5.2. Requirement Source

- Development Team and Data Processing Team

1.5.3. Static Numerical Requirement

- Data must be processed in real-time, with a delay of no more than 500 milliseconds

1.5.4. Dynamic Numerical Requirement

- The system must process data from up to ten drones at the same time.

1.6. Load Balancing

1.6.1. Feature

- Load balancing for data processing

1.6.2. Requirement Source

- Data Processing in Realtime

1.6.3. Static Numerical Requirement

- Processing data from all drones simultaneously should not cause delay more than 100ms.

1.6.4. Dynamic Numerical Requirement

- When numerous drones (up to 10) are operating simultaneously, the system should divide up the data processing jobs so that performance is not compromised.

1.7. Real-time fused processing

1.7.1. Feature

- Data Fusion and Analysis
- Real-time integration and analysis of data from multiple sources

1.7.2. Requirement Source

- Data processing in real-time

1.7.3. Static Numerical Requirement

- The total amount of time that all of the sensors' data is processed plus synchronization should not exceed 500 ms.

1.7.4. Dynamic Numerical Requirement

- For real-time analysis, the system should be able to combine data from perimeter

cameras, motion sensors, and drones.

- Finding intrusions and other irregularities in the industrial area should be part of this investigation.

1.8. Surveillance Range

1.8.1. Feature

- Extended surveillance range for drones

1.8.2. Requirement Source

- Managers, Factory Owners, and Drone Manufacture

1.8.3. Static Numerical Requirement

- Drones must have a maximum surveillance range of at least 5 kilometers.

1.8.4. Dynamic Numerical Requirement

- Even in bad weather, the drone monitoring system needs to function and continue to monitor.

1.9. Data Export Interface

1.9.1. Feature

- Easy data export for external storage and analysis

1.9.2. Requirement Source

- Development Team and Security Administrators

1.9.3. Source of Input or Destination of Output

- It should be used to export surveillance data to cloud platforms or external storage devices for long-term analysis and storage.

1.10. User Operating Interface

1.10.1. Feature

- User Friendly and interactive Control Interface

1.10.2. Requirement Source

- Drone operators, application users and technicians

1.10.3. Source of Input or Destination of Output

- User-friendly interfaces on users' devices must enable them to operate drones and manage their operations.

1.11. Interactive Control Interface

1.11.1. Feature

- Interface to view recordings.

1.11.2. Requirement Source

- User Operating Interface

1.11.3. Source of Input or Destination of Output

- Users should be able to see live video from drone cameras on the interface, as well as the location of the drone in case of invasion.

1.12. User access Control

1.12.1. Feature

- User access authorization and authentication

1.12.2. Requirement Source

- User Operating Interface

1.12.3. Source of Input or Destination of Output

- Role-based access control should be an element of the user-friendly control interface, enabling various users to have varied degrees of access and control over the system's capabilities.

1.13. Data Logging

1.13.1. Feature

- Comprehensive data logging for audit and compliance

1.13.2. Requirement Source

- Compliance Regulations, Data research and Improvement team

1.13.3. Types of data and/or operations performed

- For auditing and compliance purposes, the system must log all intrusion occurrences, including timestamps, position, and drone actions.

1.14. Secure Logging

1.14.1. Feature

- Access Management Logging

1.14.2. Requirement Source

- Data Logging

1.14.3. Types of data and/or operations performed

- To prevent unwanted access, surveillance data that is exported to cloud computing platforms or external storage should be sent securely via authentication and encryption protocols.

1.15. Geographical Data Integration

1.15.1. Feature

- Geographical Data Integration for exact feature mapping

1.15.2. Requirement Source

- Geographic Information System (GIS) Providers

1.15.3. Types of data and/or operations performed

- The integration of geographical data into the system is necessary to enhance location-based analysis and precise monitoring within the industrial region.

1.16. Geographical data enhancement

1.16.1. Feature

- Updating of geographic data

1.16.2. Requirement Source

- Geographic data integration

1.16.3. Types of data and/or operations performed

- Enhancing feature mapping and location-based analysis requires that the system not only incorporate geographical data but also continuously update and improve it.

2. Architecturally Significant Elements

2.1. Overview and Architectural Views

In the context of our drone-based encroachment detection system, a comprehensive architectural view is essential to understand how various components interact to achieve our system's goals. The 4+1 model provides a holistic perspective –

- **Logical View** - The Logical View focuses on the high-level structural elements of our system.
- **Process View** - The Process View delves into the dynamic aspects of our system.
- **Development View** - The Development View provides insights into software development.
- **Physical View** - The Physical View considers the deployment and hardware aspects of the system.

2.2. Logical/Quality Elements and why each is architecturally significant

Several logical and qualitative components contribute to the architectural importance of the system in the Logical View –

- **Scalability** - Scalability is essential for adjusting to changing environmental conditions, allowing for smooth growth for monitoring small and large regions without sacrificing performance.
- **Modularity** - By grouping logical pieces into reusable modules, modularity increases flexibility and facilitates maintenance, upgrades, and the integration of new features or sensor types.
- **Security** - Security is critical for protecting sensitive data and system integrity, as well as preventing unwanted access to drone hardware and transmitted data via built-in logical security.
- **Efficiency** - Efficiency is critical for real-time threat detection, which necessitates improved logical architecture of algorithms and data flows to enable fast notifications.

2.3. Database Elements and why each is architecturally significant

- **Data Storage and Retrieval** - Databases are essential for archiving reference material and historical data, which makes trend analysis and threat tracking easier.
- **Data Consistency** - To guarantee the accuracy of threat detection and reporting, it is essential to maintain consistent, current data throughout the system.
- **Redundancy and Failover** - In the event of a hardware or network failure, database components must have redundancy and failover procedures in place to safeguard system continuity and avoid data loss.
- **Data Privacy and Compliance** - With features like encryption and access controls, the database should support both data privacy and regulatory compliance.

2.4. Reuse Elements and why each is architecturally significant

The creation of a drone-based encroachment detection system must prioritize reusability:

- **Sensor Reusability** - Designing drones with interchangeable sensors allows you to adapt to various situations and applications without having to completely redesign the hardware.
- **Algorithm Reusability** - Reusable computer vision and machine learning algorithms allow for effective adaptation to novel challenges in encroachment detection.
- **Integration Reusability**- By reusing existing integration components, a well-defined integration approach guarantees that the system can be quickly adapted to different environments.

- **Reusable User Interface** - User interface components facilitate the creation of customized dashboards and interfaces for a range of users and stakeholders.

3. Draft Architecture

3.1. Overview and Architectural Views

- Within the architectural framework of our drone-based encroachment detection system, each view assumes a critical role in establishing a resilient and adaptable solution.
- The Logical View elucidates the core components responsible for user interfaces, camera video processing, intrusion alert generation, and data storage.
- The Process View delineates the dynamic interactions between system components, encompassing drone operations, flight planning, and camera video processing.
- The Physical View shifts the focus to the system's hardware, including drones, databases, control, ensuring that the physical architecture aligns with environmental requirements.
- The Deployment View spotlights the technological underpinnings, such as the software programming languages employed detection in camera video, and database management.
- The system's versatility comes to the fore in Scenarios View, where it adeptly manages diverse scenarios.
- Also including user interactions with operators, efficient flight scheduling between charging intervals, and rapid alert systems for encroachment detection.
- These architectural perspectives collaborate harmoniously, crafting a well-rounded system designed to cater.
- As well as to a broad spectrum of applications while upholding efficiency, security, and responsiveness.

3.2. Logical / Quality Elements

- **Alarm System** - The alarm system plays an important function in informing authorities and property owners of identified encroachments.
- This allows for fast reactions and serves a range of functions, including emergency management, environmental preservation, and perimeter protection for a variety of locales.
- **Drones** - Drones are essential for conducting effective border surveys, frequent patrols, high-resolution data collecting, and quick reactions to incursion detection.
- Their contributions improve system accuracy and enable rapid responses in crucial situations.
- **Drone Operations** -The drone management system manages fleets, simplifies mission planning, and guarantees regulatory compliance.
- It provides real-time monitoring, streamlines maintenance scheduling, and effectively reduces operating expenses, increasing drone efficiency in a variety of industries.
- **Camera System** - The camera systems built into encroachment detection monitor borders, identify incursions, and provide continuous surveillance.
- These features allow for quick reactions to encroachments while also assisting with dispute resolution and keeping historical data for trend research and continuing monitoring.

3.3. Database Elements

- **Centralized Information Management Database** - The centralized RDBMS serves as the core data repository for storing critical information in our drone-based encroachment detection system.

- **Surveillance Record Storage** - Camera video and accompanying data are archived efficiently in our drone-based encroachment detection system for analysis and retrieval.

3.4. Reuse Elements

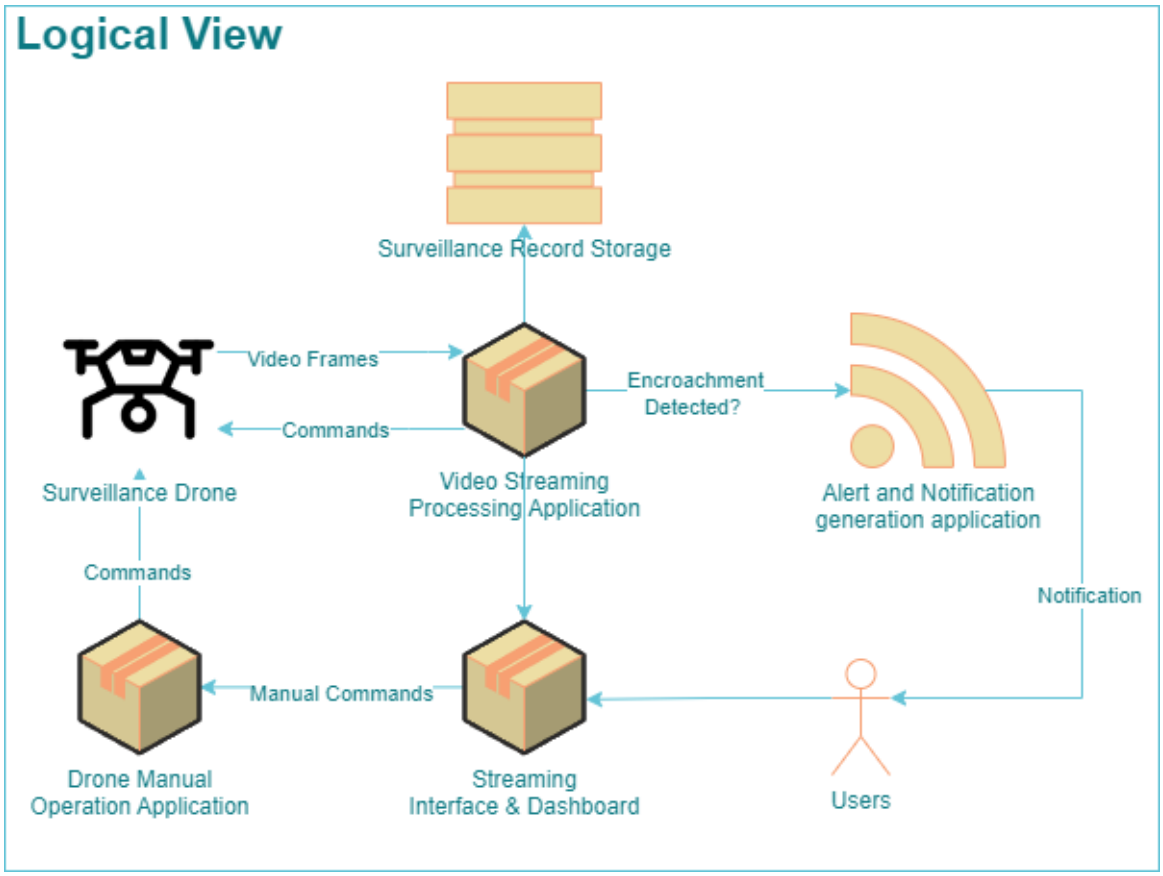
- State of the Art open-source drone simulator which can efficiently work in limited computing.
- Open-source drone footage for data model training for encroachment detection engine.
- Stable and community developed data cleaning and machine learning libraries.
- State of the art pre-trained deep learning data models which are trained on millions of images.
- Open-source programming language, python for model training and video stream processing.
- Cloud based RDBMS database and storage service such as AWS S3.

3.5. Logical View

3.5.1. Introduction and rationale for the view

- For incursion detection, our surveillance system combines drones, cameras, and deep learning.
- The logical approach emphasizes the software components, user interfaces, and alerting systems that structure the operation of the system.
- It also allows for live streaming and archiving of surveillance footage, allowing for real-time monitoring and historical study of encroachment.

3.5.2. Model / Graphical Representation



3.5.3. Key Details

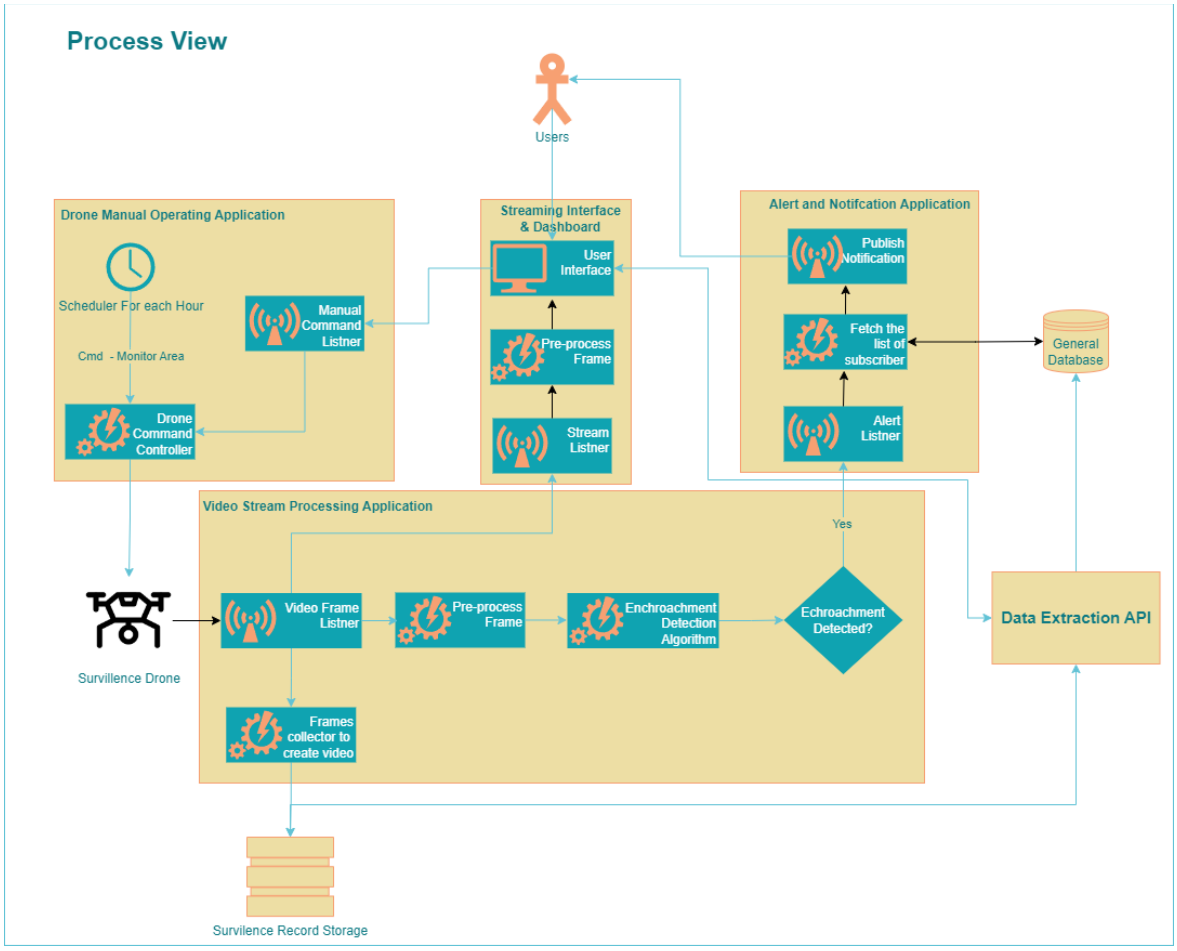
- Streaming Interface and Dashboard
    - Real-Time Video Display: Provides live video feeds for real-time monitoring.
    - User-Driven Navigation: Allows users to customize their view.
- Drone Operation Application
    - Alert Response: Enables users to respond to intrusion alerts.
    - Safety Safeguards: Ensures safe drone operation.
- Surveillance Drone
    - GPS Tracking: Provides precise drone location information.
    - Data Transmission: Transfers real-time data for analysis.
- Alert Application
    - Alert Dashboard: Presents real-time alerts for situational analysis.
    - Details of Alerts: Offers detailed information about intrusion alerts.
- Surveillance Records
    - Archive of Camera Footage: Preserves high-resolution camera data.
    - Historical Encroachment Records: Maintains records of past intrusion incidents.

3.6. Process View

3.6.1. Introduction and rationale for the view

The system's three primary components are the Drone Operating Application for real-time control, scheduled patrols, and intrusion warnings. The Live Streaming Interface manages camera access, ensures data security, and collects real-time drone data. The Alert Application records intrusion details for later analysis, sends out alerts promptly, analyzes camera data, and detects intrusions.

3.6.2. Model / Graphical Representation



3.6.3. Key DetailsProcess View

- Drone Operating Application
    - Drone Data Monitoring: Collects real-time data for operator information and control.
    - Pre-planned Paths: Facilitates routine patrols with predefined drone routes.
    - Intrusion Alerts: Identifies anomalies and notifies operators and response teams for quick security reactions.
- Live Streaming Interface
    - Controlled Access: Restricts access to live camera feeds based on user roles.
    - Camera Data: Receives real-time video and data from multiple drones
    - Data Privacy: Uses encryption and strong security measures to protect camera feed data.
- Alert Application
    - Alert Trigger: Identifies intrusions and triggers alerts using logic and algorithms.

- Data Collection and Analysis: Gathers and analyzes camera feeds for intrusion verification.
- Instant Alerts: Generates and sends rapid alerts through push, SMS, or email notifications.

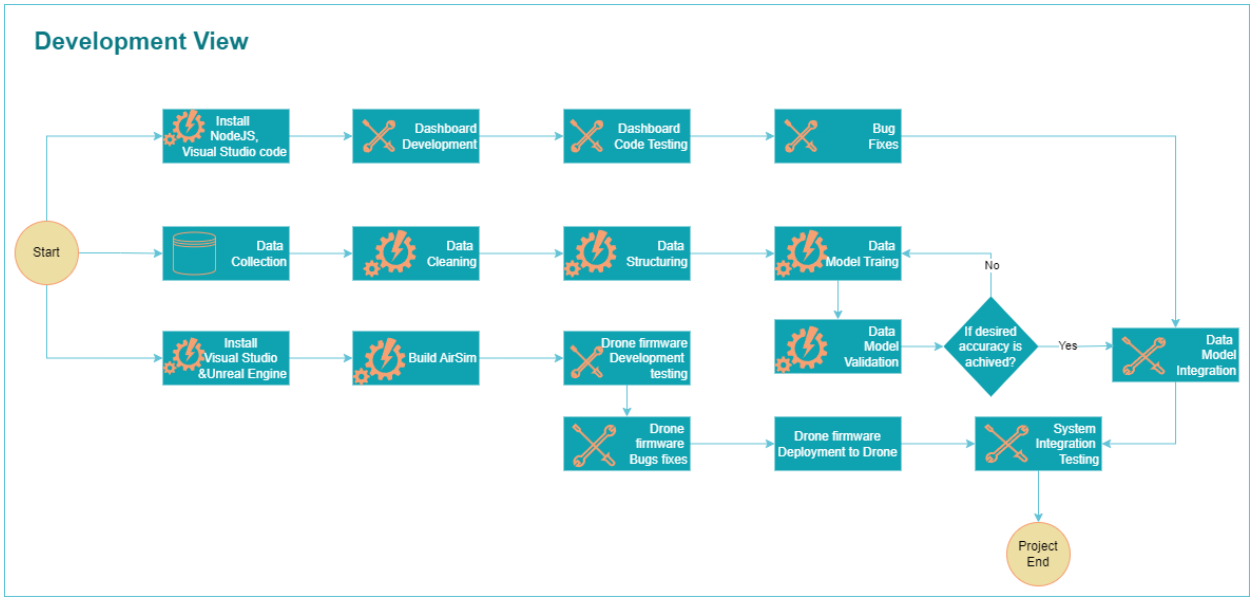
- Logging: Records detailed information about intrusions for analysis and future use.

3.7. Development View

3.7.1. Introduction and Rationale

- The development perspective can be divided into three components.
- Developing a machine learning model to detect invasion
- Drone flight pattern simulation and configuration
- Creating a dashboard and establishing communication channels with drones

3.7.2. Model / Graphical Representation



3.7.3. Key Details

- Dashboard Creation
- NodeJS Web App: Used for creating the dashboard interface, enabling user interactions with drones.
- Communication Channels: Establishes communication between the application server and drones for data exchange and command transmission.
  - Model Training
  - Python Language: Preferred for training the YOLO8 model to detect and identify objects in drone video footage.
  - Object Position Extraction: Extracts the position of detected objects in the video.

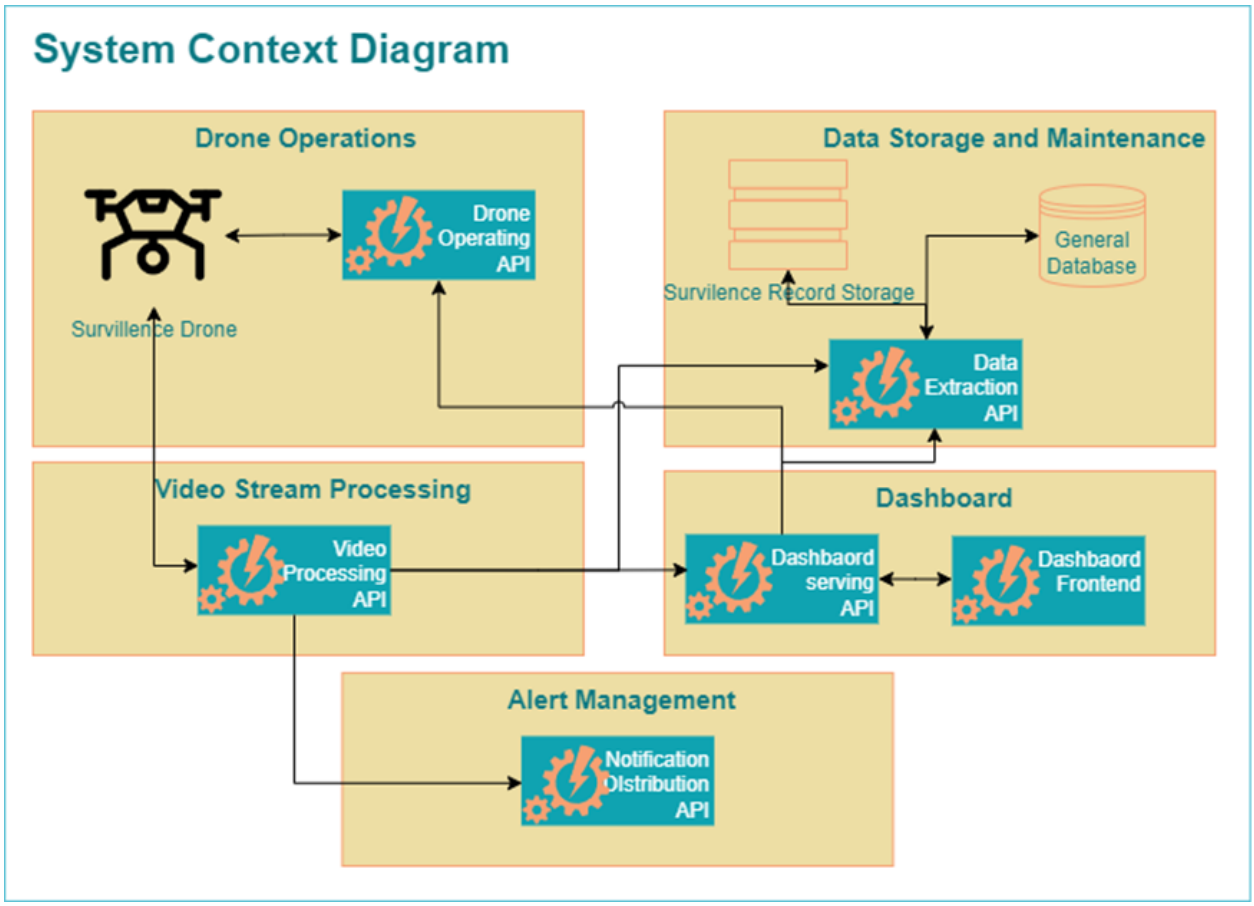
- Simulation: Utilized for simulating various system components and testing in a controlled environment.

4. Initial Detailed Design

4.1. Highest Priority Detailed Design Element

- **Overview Description**
- The drone-based encroachment detection system is intended to improve security and surveillance in restricted or sensitive areas with unmanned aerial vehicles (UAVs), often known as drones.
- Drones are fitted with specific sensors and a dedicated operating program monitor and identify unwanted entries into secured zones in real time.
- By notifying security professionals of possible threats and incursions, this technology aids in the protection of essential infrastructure, private assets, and public areas.

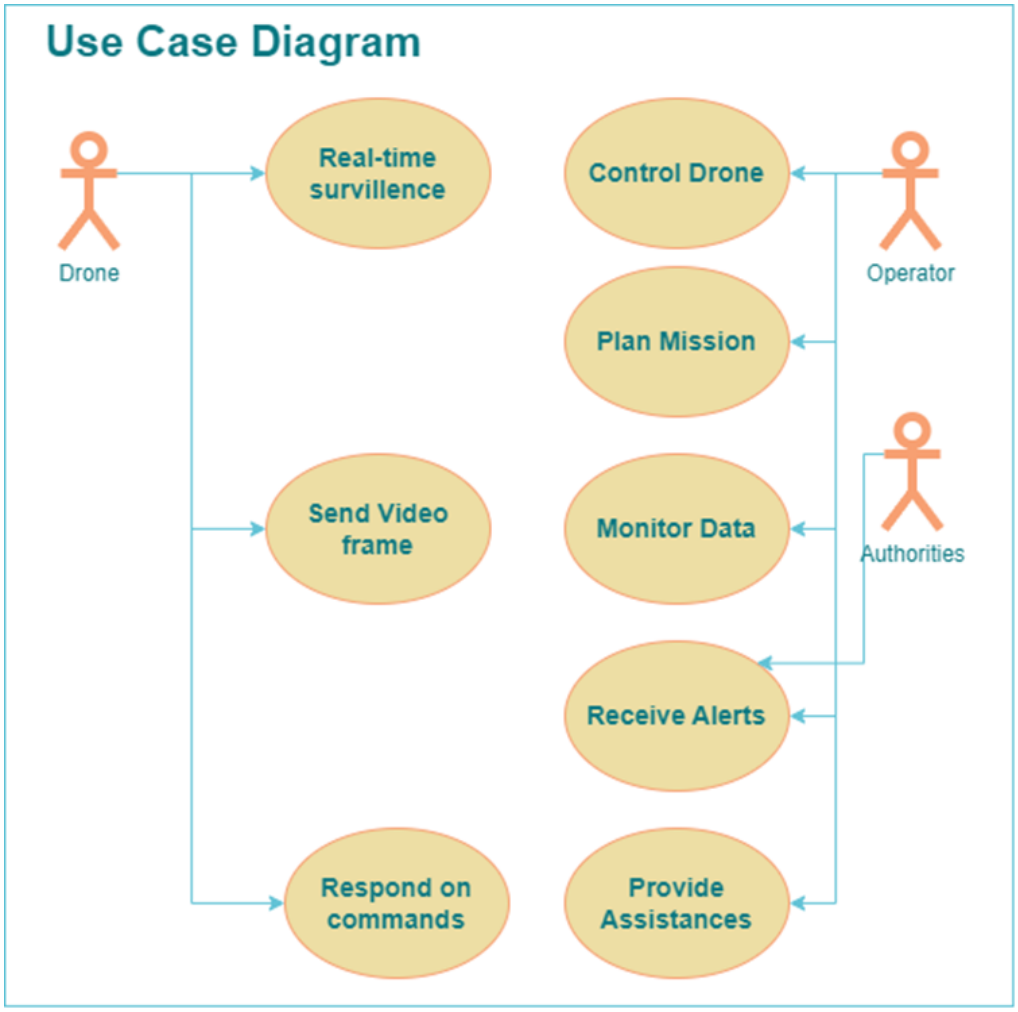
- **System Context Diagram**



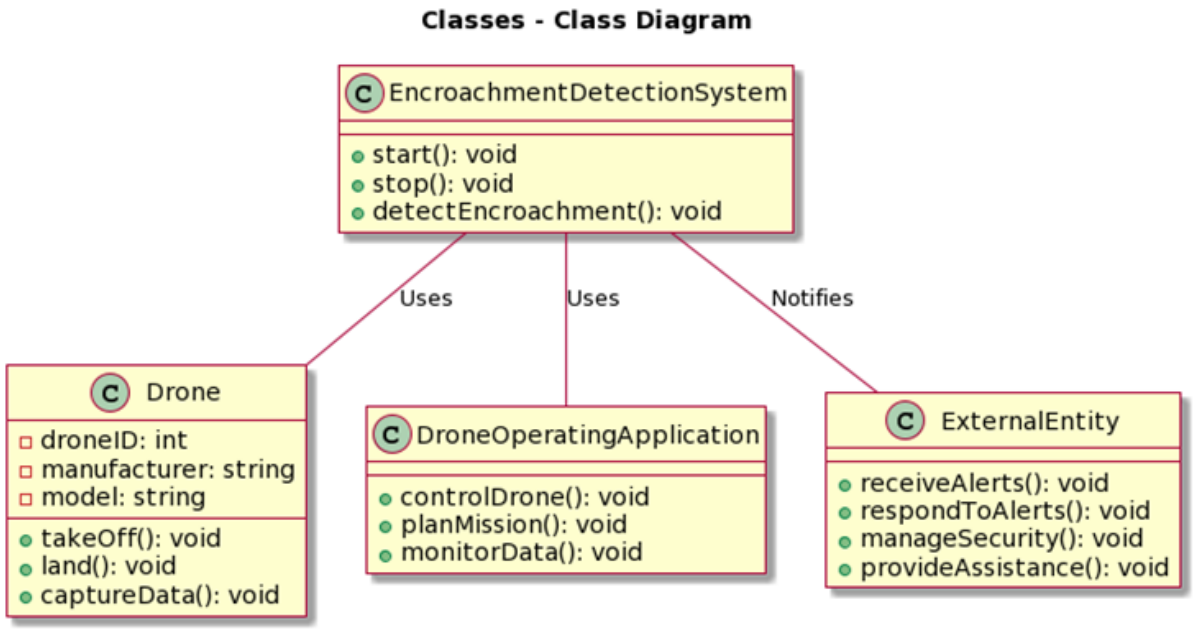
- **Brief Description of Each Major Context Element**
- A drone is a highly maneuverable, remote-controlled, or autonomous aircraft outfitted with specialized sensors for visual and environmental data collection for surveillance purposes.

- The software that acts as the primary interface for operators to manage and monitor drone operations is known as the drone operating application.
- It allows operators to control the drone's motions, modify sensor settings, and get real-time data, allowing them to respond to possible encroachments more quickly.
- **Detailed Design of Drone Operations**
- **Description of the design element and why it is significant and a priority**
- The drone and the drone operating application represent the core technological components of the encroachment detection system. They are significant and a top priority for several reasons:
- **Real-Time Surveillance and Data Collection** - Drones fitted with modern sensors, such as high-resolution cameras, and other environmental sensors, may collect real-time data.
- This information can be useful in detecting and analyzing potential encroachments, security breaches, or safety problems.
- The capacity of the drone to visit difficult-to-reach or hazardous regions makes it crucial for getting timely information.
- **Integration with Alert Systems** - When odd behavior or encroachments are noticed, the drone operating application is integrated with alert systems, allowing it to send alarms, notifications.
- This integration guarantees that new issues are addressed in a timely manner.
- **Scalability and Coverage** - Drones can cover enormous regions fast and effectively, making them very scalable for a variety of deployment situations.
- They can fly over large areas of land, infrastructure, offering a complete picture that would be difficult to acquire with fixed surveillance systems or manned patrols.
- This scalability is vital for monitoring broad perimeters, critical infrastructure, and disaster zones.
- **Adaptability and Flexibility** - The drone and the operating application can be tailored to suit specific use cases.
- They can adapt to different environmental conditions, security requirements, and operational needs, making them highly flexible for a wide range of applications.
- **The team member working on this design element**
- Ameya Shahu is working on this system context.
- **System Context with a focus on this design element**
- **Drone** - The physical unmanned aerial vehicle (UAV) outfitted with sensors for data collection and surveillance is known as a drone.
- It is critical in obtaining real-time data and photographs from the surveillance area.
- **Drone Operating API** - The software interface that lets other programs, including the user interface, control the drone's motions, and modify sensor settings.
- It serves as a link between the operator and the drone, allowing for smooth communication and control.
- **External Entities** - authorized individuals who can access the system's data through the API, control the drone, and act when suspicious activities are detected.
- First responders who may be informed in an emergency using the API and give support and help as needed.
- External systems that can receive alerts and messages from the incursion detection system via the API, triggering alarms or automating actions.
- Existing surveillance systems or infrastructure that may connect with the incursion detection system via the API to provide full security and surveillance capabilities.

- **Use Case Diagram with a focus on this design element**



- **Design Element Class Diagram and Rationale**

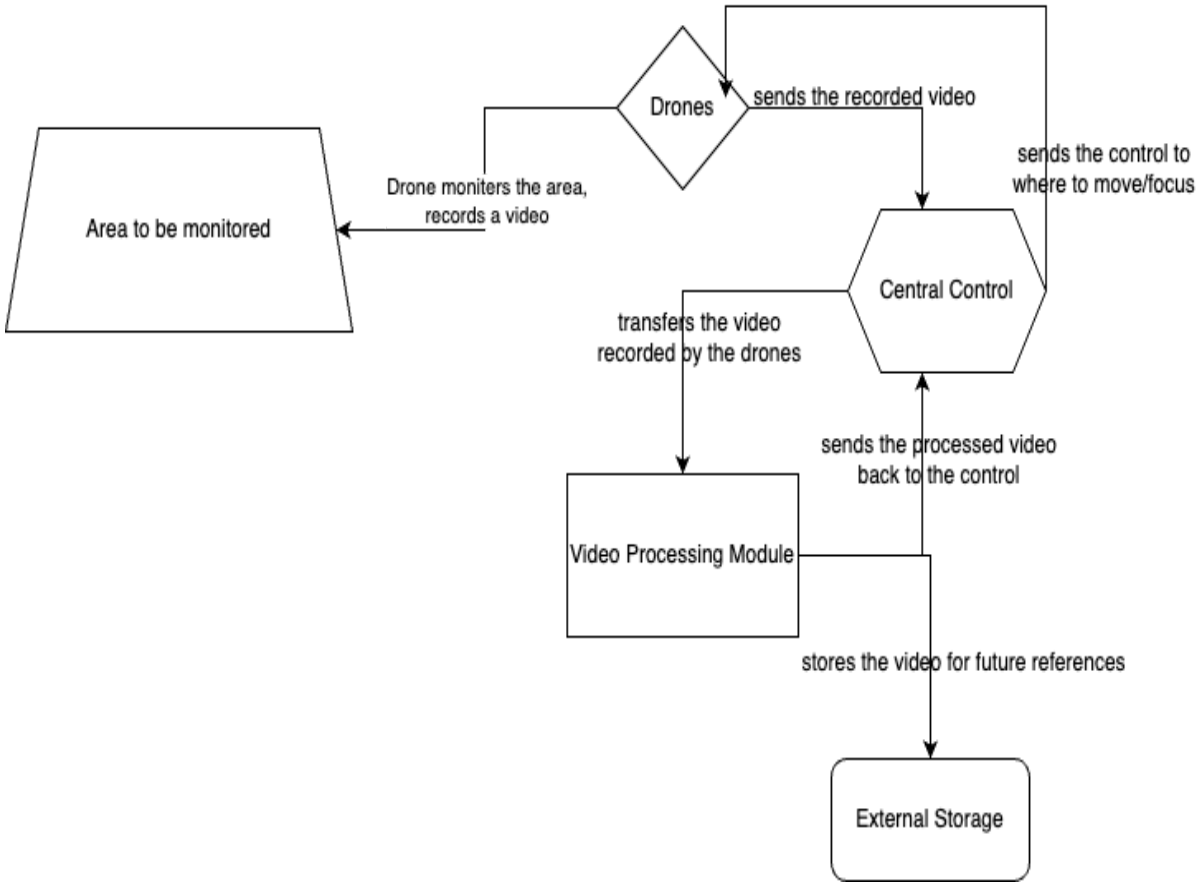


4.2. Second Highest Priority Detailed Design Element

- **Description of the design element and why it is significant and a priority**
- The "Video Processing Module" is a vital design feature since it is in charge of real-time video data analysis.
- It is critical to the primary goal of identifying encroachments in the industrial area.
- This module will use image processing and computer vision techniques to detect any illegal activity.
- Using this we can ensure that true security threats are discovered and notified as soon as possible.
- This element's fast and accurate operation is critical because it is the heart of our intrusion-detection system.
- **Pravalika Mukkiri** is working on this system context.
- **System Context with a focus on this design element**

  System Context Diagram Overview

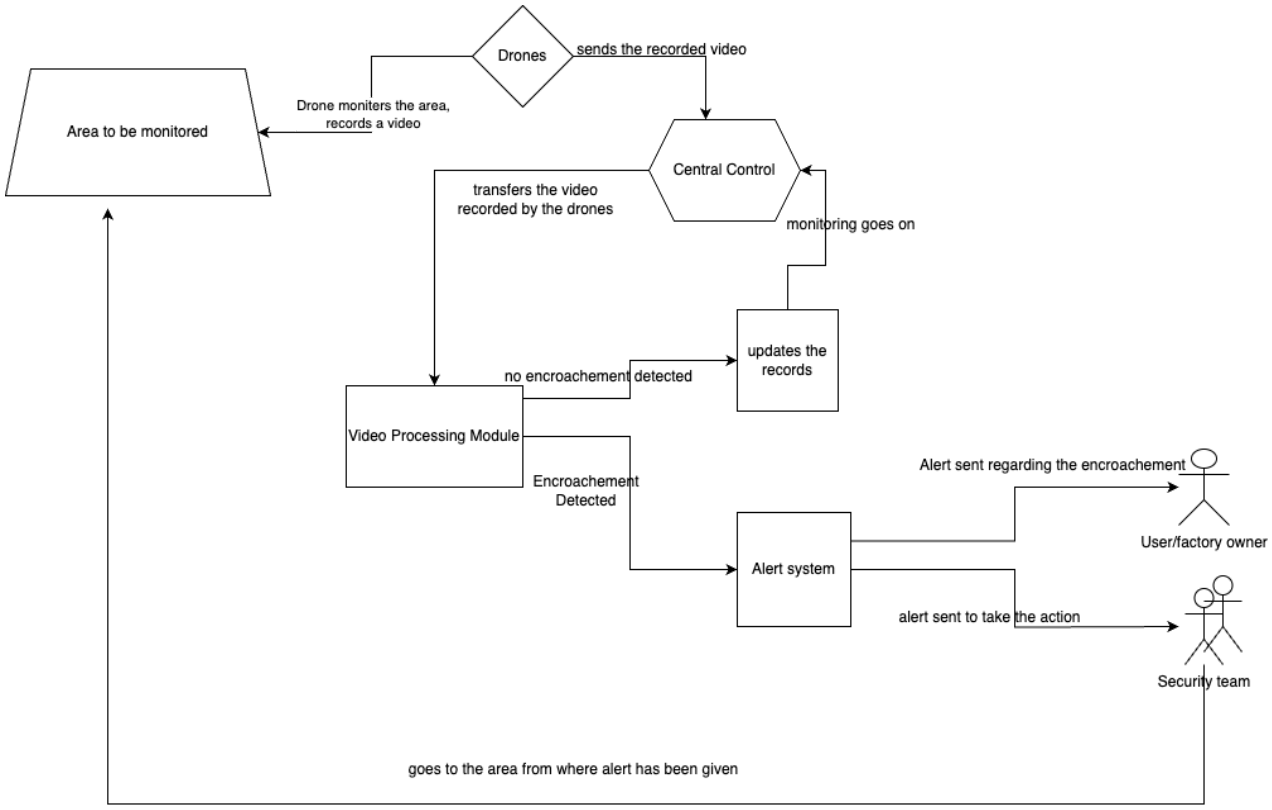
  - Visual depiction at a high level
  - Describes the connections between the Video Processing Module and external entities or important system components.



- Purpose and Significance
  - A high-level depiction of the Video Processing Module's place in the system architecture.
  - Interactions with external entities are depicted.
  - Emphasizes the impact of external components on the module and vice versa.
- External Entities of Importance
  - Collect video data from the industrial region using drones.
  - Central Control System: Coordinates module activities and manages drone operations.
  - External Storage: Saves processed video data for future use.
  -
- Data Flow and Interaction
  - Arrows, lines, and notations depict data, control, and communication flow
  - Illustrates fundamental interactions between the Video Processing Module and external elements
- Understanding System Architecture
  - Offers insight into the roles and relationships of system components
  - Foundation for comprehending video processing for encroachment detection
  - Stakeholder and Team Value
    - The System Context Diagram is a useful tool for stakeholders and team

members.
- It will help to understand the critical linkages that support the Video Processing Module's operation and integration with other system elements.

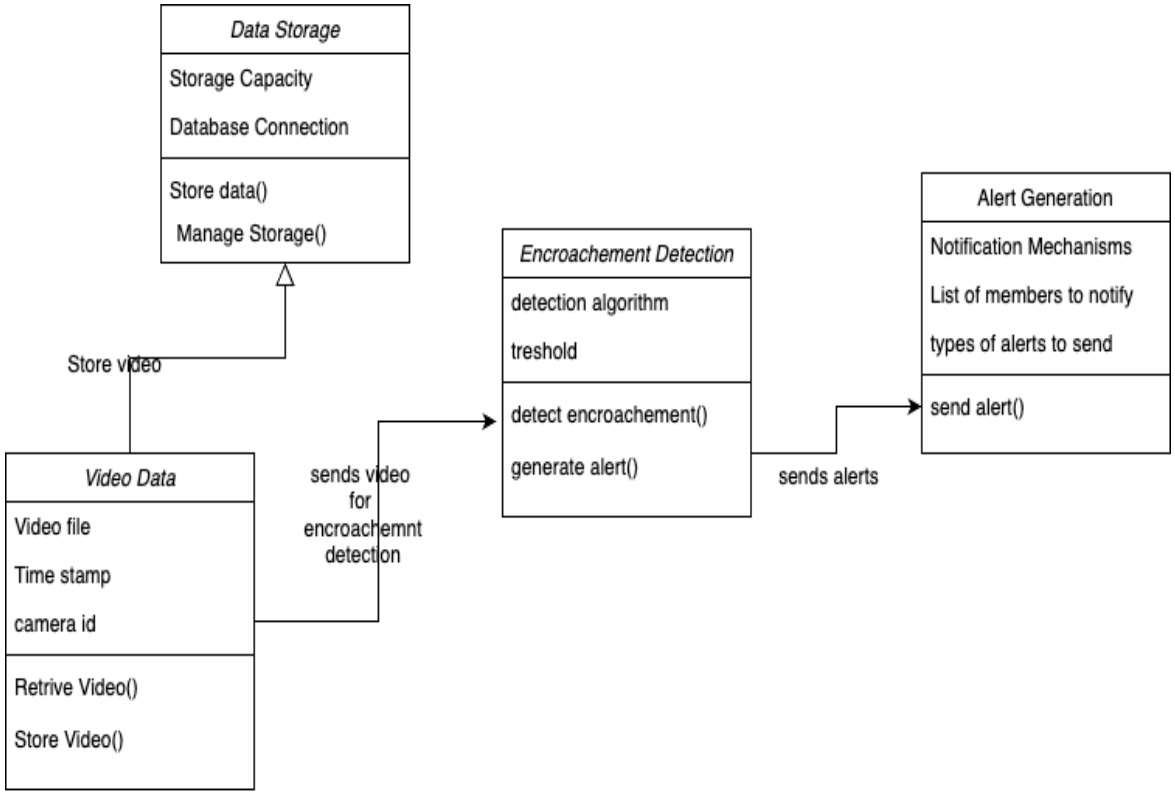
- **Use Case Diagram with a focus on this design element**
- **Use Case Diagram Purpose**
  - A graphical representation of interactions between the Video Processing Module and system actors or use cases
  - Emphasizes the specific functionalities and responsibilities of the Video Processing Module within the system
  - Use Case: Video Surveillance Processing
    - The surveillance video is monitored.
- Use Case Diagram



  - Actors
    - Drone
    - Security Team
    - User/factory owner
  - Preconditions:
    - Drones are actively surveilling the industrial area.
    - Video data from drones is available for processing.
  - Postconditions:

        - Encroachment detection results are generated.
        - Encroachment alerts are sent to the central control system.
        - Processed data is archived in external storage.
    - Primary Pathway:
        - Drone captures the video
        - The video information is sent to the central control system.
        - The video data is processed by the central control system to detect any encroachments.
        - An incursion alarm is triggered if unlawful motion is detected.
        - The alarm is delivered to the central control system and the appropriate individuals.
        - For future investigation, encroachment data is preserved in external storage.

- **Design Element Class Diagram and Rationale**
    - This diagram illustrates how the classes interact with each other and provides a blueprint for designing the system.



-

    - The "Video Data" class
        - This class handles raw video data from drones, containing attributes like

Timestamp, Video File, and Camera ID.
    - The methods "RetrieveVideo()" and "StoreVideo()" make video data retrieval and storage easier.
    - Through the "StoreVideo" method, the Video Data class is closely related to the Data Storage class.
- "Encroachment Detection" class
    - This class is in charge of detecting encroachments and includes attributes such as Detection Algorithm and Threshold.
    - The "DetectEncroachment()" method detects encroachments by applying certain algorithms to video data.
    - whereas the "GenerateAlert()" method generates alerts when an encroachment is identified.
    - It is linked to the Video Data class for video data reception and the Alert generating class for alert generating.

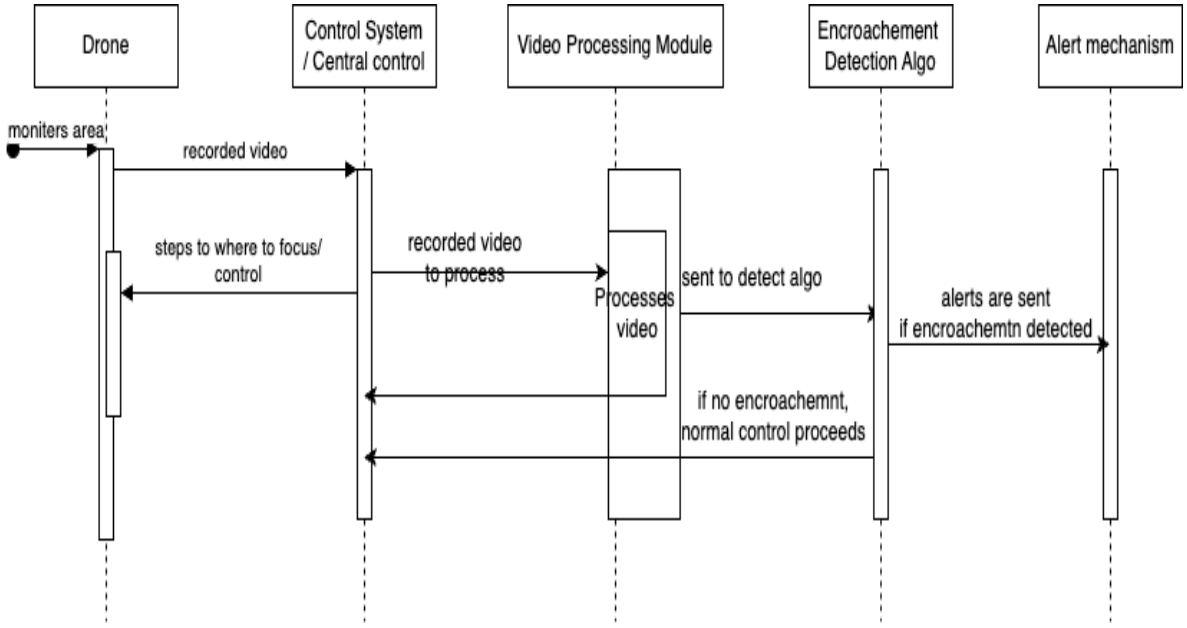
- "Data Storage" class
    - This class is responsible for storing processed data and includes attributes such as Database Connection and Storage Capacity.
    - The "StoreData()" method is used to store processed data, such as video files
    - whereas "ManageStorage()" provides effective storage capacity and data retention management.
    - It is linked to the Video Data class via the "StoreData" function and is essential for data storage and retrieval.

- "Alert Generation" class.
    - This class is focused on alert creation and has attributes related to Notification Mechanisms.
    - When encroachments are detected, the "SendAlert()" method is responsible for delivering alerts to the central control system or security staff.
    - For alert distribution, the Alert Generation class can be linked to external notification systems.
- These classes work together to ensure efficient video processing and encroachment detection while maintaining system integrity.
- The class diagram for this system should visually depict these relationships and the roles each class plays in the system architecture.

- **Relevant UML Models**
- **Sequence Diagram**
    - Sequence diagrams depict the interactions of several components or classes

throughout time.

- Sequence diagrams in the context of the video processing module can show how data flows between the Video Data, Encroachment Detection, Data Storage, and Alert Generation classes during the video processing and threat detection process.
- These diagrams might aid in visualizing the sequence of events that occur when an invasion is identified, from video retrieval to alarm creation.



- This sequence diagram depicts the dynamic interactions between the core components of the Video Processing Module.
- It explains how the system processes video data, recognizes dangers, and regulates data flow to stakeholders.
- The figure provides insights into the internal workings of the module during threat detection scenarios by following the chain of messages and exchanges.

4.3. Next Highest Priority Detailed Design Element

- **Description of the design element and why it is significant and a priority**

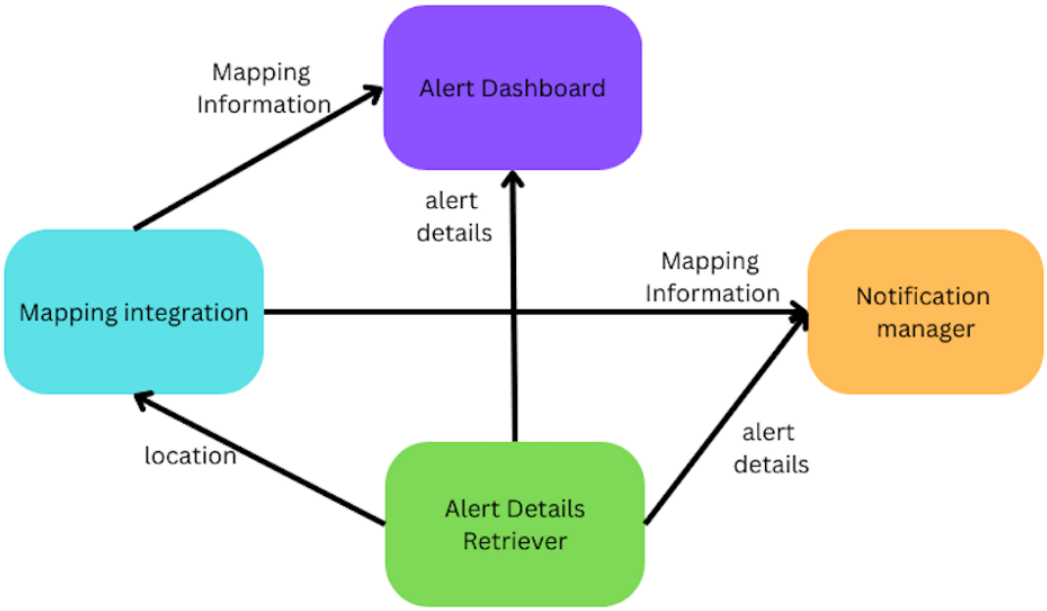
  Responsibilities:

  - Alert Aggregation and Categorization: The API is responsible for aggregating alerts generated by drones and categorizing them based on severity, location, and type of encroachment detected.
  - Notification and Alert Dissemination: It manages the distribution of alerts to relevant stakeholders or response teams via various communication channels like email, SMS, push notifications, etc.
  - Analysis: The API facilitates trend analysis and generating reports for insight into patterns and trends related to encroachments.
  - Real-time Dashboard Provision: It powers a real-time alert dashboard, displaying detailed information about each alert, allowing for quick situational analysis and informed decision-making.
  - Integration with Mapping Services: Integrating with mapping services, the API overlays the alerts on a map, providing visual representation and precise location details for better situational awareness and quicker response.

  Importance:

  - Rapid Response: Efficient alert management is critical in enabling a swift response to unauthorized encroachments. The API ensures that alerts are handled promptly and appropriately distributed, reducing response times to potential threats or intrusions.
  - Situational Awareness: By organizing and displaying alerts in real-time on a dashboard and map overlay, the API enhances situational awareness. This allows response teams to understand the exact locations and types of encroachments, aiding in effective decision-making.
  - Data Analysis and Trend Identification: The historical data stored by the database allows for trend analysis, identifying patterns of encroachments. This insight can be invaluable for improving security measures and planning preventive actions.
  - Integration and Coordination: It facilitates coordination between different stakeholders or response teams by ensuring that the right information reaches the right people through various communication channels.

- **The team member working on this design element: Lalit Arvind Balaji**
- **System Context with a focus on this design element**



- The alert details retriever gets the location of encroachment from the drone and the videos from which it identifies the severity and type of encroachment.
- The location details are sent to Mapping integration where the location is overlayed on map to help security team navigate to the location
- The notification manager receives the mapping info and other alert details and notifies the registered authorities through their preferred communication channel with full details.
- The alert dashboard get receives the alert and displays them prioritized by severity

- **Use Case Diagram with a focus on this design element**

Actors:

- Stakeholders
- Security team
- Management authority
- Alert Management service

Preconditions:

- The drones are monitoring the designated area
- Encroachment is detected
- Service is enabled
- Notification channels are set for users

Postconditions:

- Notifications are sent in appropriate channels to all the registered authorities
- Alerts are organized in dashboard based on severity

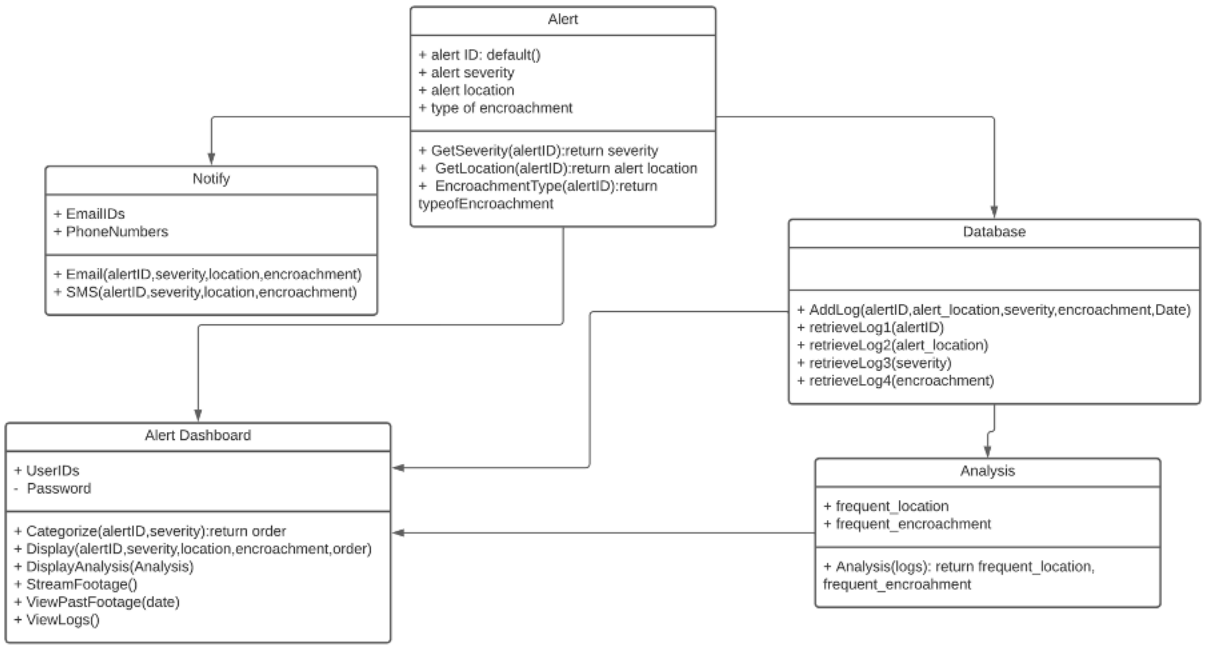
Primary Pathway:

- Users enable alert management service and provide details of notification method(email id, phone number)
- Drones that are monitoring designated area detect encroachment
- Alert notifications are sent to registered users who have enabled notification through

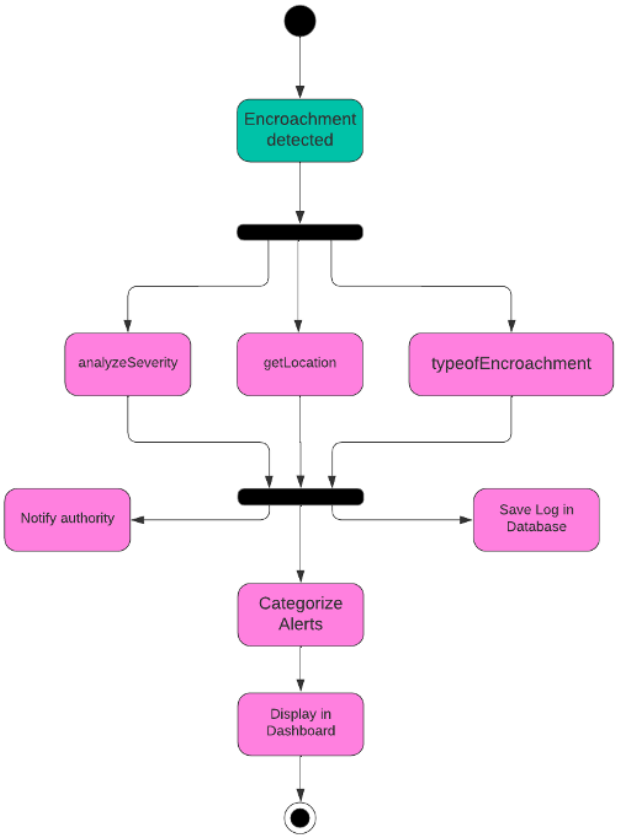
their preferred notification channel.

- The alert is updated and organized in alert dashboard

- **Design Element Class Diagram and Rationale**



- **Relevant UML Models**



4.4. Next Highest Priority Detailed Design Element

- **Description of the design element and why it is significant and a priority**
- **The team member working on this design element**
- **System Context with a focus on this design element**
- **Use Case Diagram with a focus on this design element**
- **Design Element Class Diagram and Rationale**
- **Relevant UML Models**

4.1.1 Brief Description of Major Context Element assigned to this team member

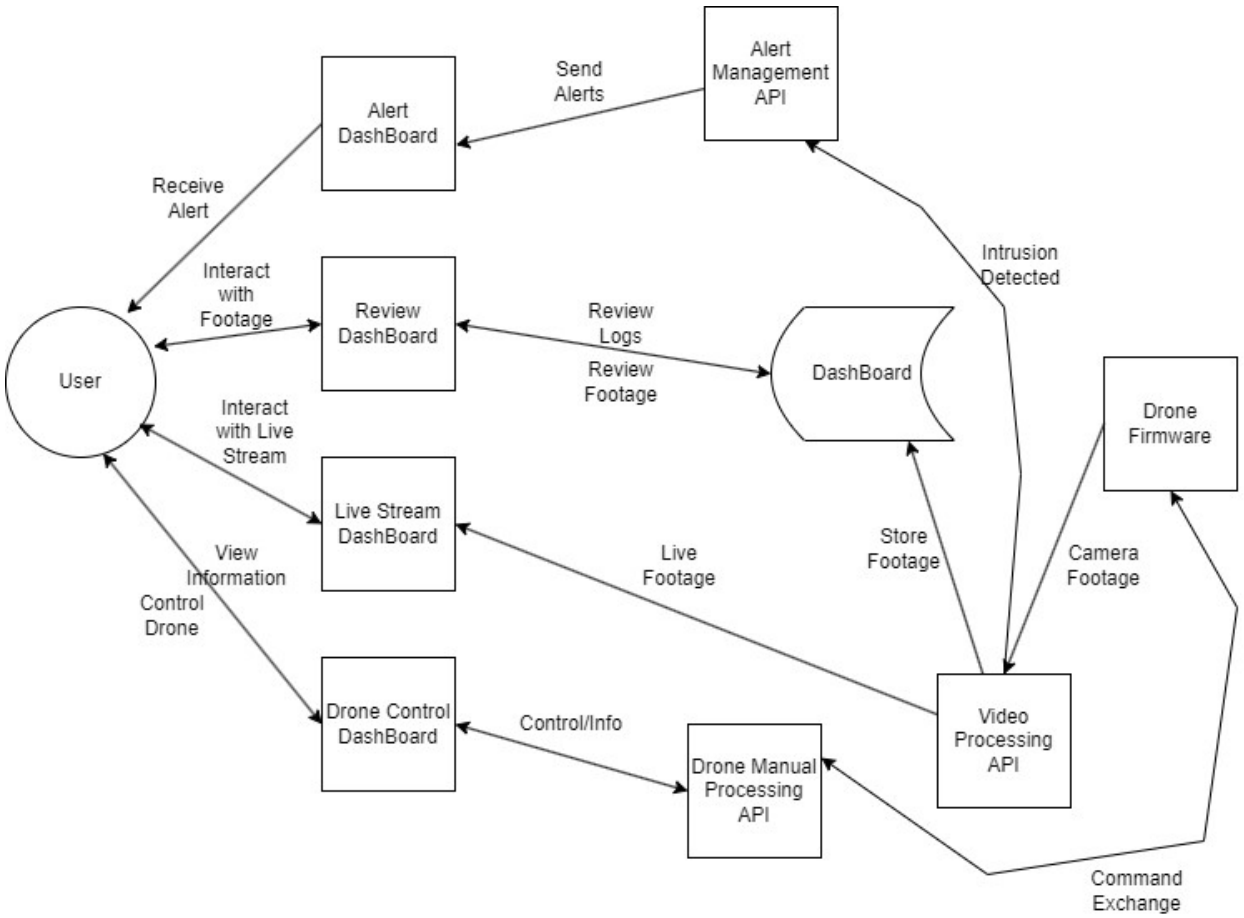
The dashboard organizes all of the functionality of the system and presents it to the viewer in an accessible form which is easy to navigate and use.

2.1. Detailed Design Element for this team member - Dashboard

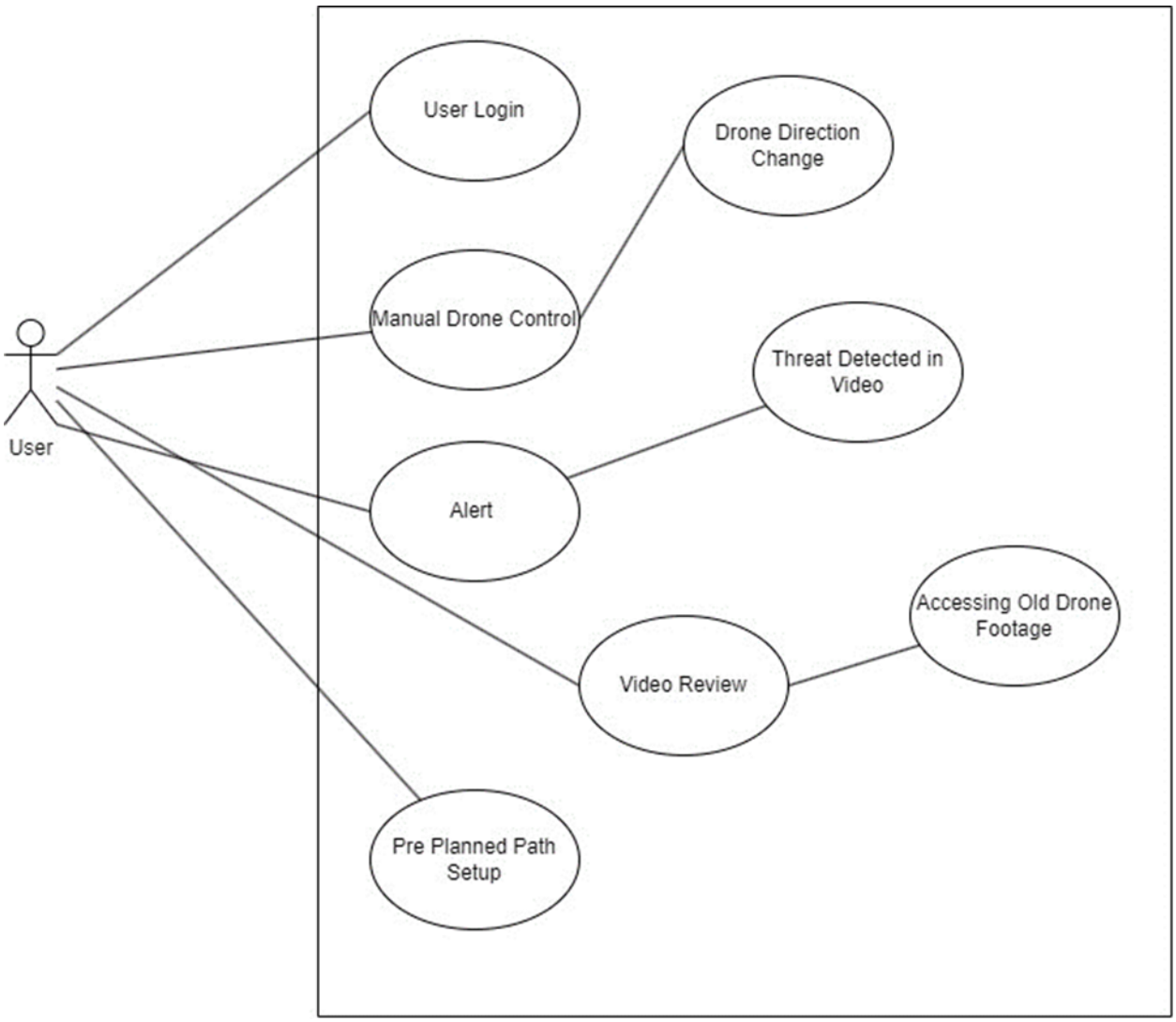
4.2.1 Description of the design element and why it is significant and a priority

- Live Stream DashBoard: This sub-component is in charge of providing the viewer with live drone camera feed for manual monitoring.
- Drone Manual Control: This subcomponent is in charge of informing the spectator about the drone's location, battery life, and manual flight controls.
- Alert Display: Should the Video Processing API identify a threat, this subcomponent will display an alarm.
- Reviewing Earlier Footage: This component gives the user access to videos from the database so they can revisit earlier footage and blogs.

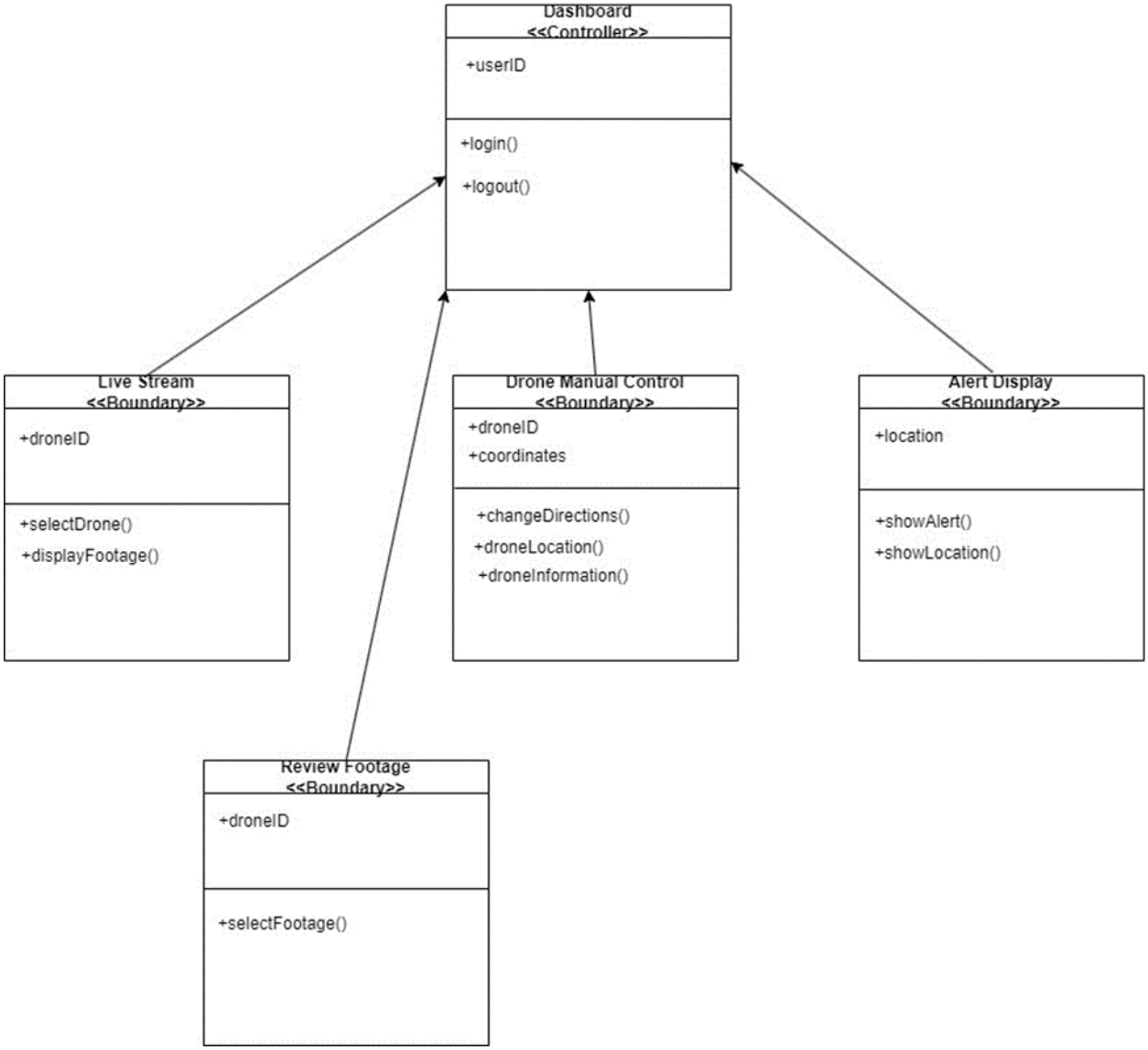
4.2.2 System Context with a focus on this design element



4.2.3 Use Case Diagram with a focus on this design element


User Login
Drone Direction Change
Manual Drone Control
Threat Detected in Video
User
Alert
Accessing Old Drone Footage
Video Review
Pre Planned Path Setup

4.2.4 Design Element Class Diagram and Rationale



5. Risk Analysis

5.1. Overview

Our project poses several major hazards that must be carefully considered. Drone performance in bad weather is the major priority, with a prototype assuring dependability. Another problem is correctly identifying recognized items in an industrial setting without false alarms, which is solved by dynamic threshold modifications. Data caching and a retention strategy are used to manage storage capacity risk. Encryption and secure protocols ensure system integrity and data safety in cyberspace. Team members are working hard to address each risk to the system's efficacy and dependability in various operational circumstances.

5.2. **Drone Performance Degradation or getting lost**

- The primary project risk is the drone's performance decline in adverse weather situations such as hurricanes, snowstorms, and thunderstorms.
- These harsh circumstances may even cause the drone to become uncontrollable and fly off-course, endangering the efficiency of invasion detection.
- This risk is critical since it directly affects the system's essential functionality and is thus a top priority concern.
- Pravalika Mukkiri is working on this risk
- To address this danger, a specialized prototype will be rigorously created and tested under terrible weather conditions to minimize the possible impact of severe weather on drone operation.
- The prototype will be tested for flight stability, sensor precision, and communication reliability.
- Adjustments will be made to improve the drone's performance under difficult situations, taking into account the insights gained from data analysis.
- The prototype will be improved through an iterative testing and validation process to satisfy set performance targets.
- The knowledge gained and enhancements made will be seamlessly implemented into production drones, considerably reducing the danger of degraded surveillance in the operational system during severe weather conditions.

5.3. Drone identifying known entity as encroachment

- In a busy industrial context, it is vital to guarantee that drones correctly recognize known things, especially while many industry-related tasks are taking place.
- It's critical that drones don't wrongly label these well-known individuals, who are part of the industry's faculty, as encroachers, which could lead to unjustified false alerts.
- This circumstance is quite concerning because it has an immediate impact on the system's credibility. False alarms have the potential to muddy the line between real

alerts and erroneous communications, increasing confusion and limiting the security team's ability to respond quickly.

- Lalit Arvind Balaji has been tasked with reducing this risk.
- To overcome this issue, the system will dynamically alter its encroachment detection thresholds based on environmental conditions and the severity of the encroachment.

5.4. Exceeded Storage Capacity

- Data is collected from various sensors and drones, including video cameras and motion sensors.
- This data is stored in a scalable storage/database.
- The risk is a priority due to limited and expensive storage, making efficient data management
- crucial for incident reference.
- Storage of captured data is essential for the system's functionality, emphasizing the need for
- storage management to adapt to data volume.
- Aditya Pant is responsible for addressing this risk.
- A prototype solution involves caching data for a short interval after recording.
- After the caching period, data is moved to storage.
- A fixed retention period of three months is established, with data erasure unless needed for
- specific purposes

6. Conclusion

6.1. Requirements

6.1.1. Essential System Requirements

- Real-time surveillance streaming, operation types, load balancing, processing, control interface, user access control, secure logging, and geographic data augmentation are all discussed.
- This course investigates database elements such as data storage, retrieval, consistency, redundancy, failover, and data privacy and compliance.

6.2. Initial Design

- The initial design addresses architecturally significant elements and their importance.
- It provides a system context diagram with a detailed focus on the described design element.
- The use case diagram highlights the major responsibilities and activities of the design element.
- A design element class diagram is presented, emphasizing entity, controller, and boundary classes.
- Relevant UML models express critical aspects of the design element, ensuring clarity and understanding

6.3. **Items for future consideration**

1. **Scalability:**

- As the industrial base expands or new surveillance methods or modes are required
- The system should be scalable to accommodate all future changes.

2. **Advanced Sensor Integration:**

- Investigating the use of sophisticated sensors and technologies.
- It will improve surveillance capabilities, such as thermal imaging, LiDAR, or AI-based anomaly detection.

3. **Energy Efficiency:**

- Investigating strategies to improve drone energy efficiency
- Methods such as longer flight durations and sustainable power sources are used
- It will reduce operational costs and environmental impact

4. **Collaboration with Local Governments**
    - Processes for collaboration and information sharing with local law enforcement agencies must be developed in the event of a security breach or invasion

Appendix A: Credit Sheet

Team Member Name	Contributions
Aditya Pant	Draft Architecture, Initial Design-Dashboard, Risk Analysis and Conclusion
Ameya Shahu	Draft Architecture Initial Design
Lalit Arvind Balaji	Requirements Initial Detailed Design - Alert Management API
Pravalika Mukkiri	Initial Design Conclusion Risk Analysis