**Problem Statement: System for Geo-Tagging of privately owned cameras**

**Problem Statement:**

Develop a streamlined and user-friendly system to geo-tag private cameras with standardized metadata, efficient interfaces, real-time object identification, and automated alerts, addressing the need for enhanced public safety and expediting law enforcement response. The system should solve the problem of lacking clear locations in private cameras, making it difficult for the police to efficiently access relevant video footage during investigations in specific crime-prone areas.

**Solution:**

1. **Geo-tagged Information Management:**

* Develop a centralized database to store geo-tagged information, including camera location data (latitude and longitude).

1. **Standardized Metadata Format:**

* Implementing a consistent metadata format that includes essential details such as camera specifications, owner contact information, and visibility range with a standardized format.

1. **Automated Licensing:**

* Upgrade the system by incorporating an advanced licensing feature that allocates Licenses to individual private cameras, facilitating effortless communication with the central Command and Control Centre.

1. **Object Identification and Alerts:**

* Implement real-time alerts triggered by image processing algorithms for specific object identification, notifying the Command-and-Control Centre of unauthorized persons or suspicious objects.

1. **Communication Facilitation:**

* Facilitate streamlined communication between authorities and camera owners in urgent situations through secure channels of communication to preserve confidentiality.

1. **User-Friendly Interface for Law Enforcement:**

* Improve the efficiency of locating specific video footage by seamlessly integrating advanced search capabilities that are tailored to location, time, and camera specifications.

**Software and Technology:**

1. **Programming Languages:**

* HTML
* CSS
* JavaScript
* Python

1. **Frameworks:**

* React.js (Front-end)
* Node.js/Express.js (Back-end)

1. **Database Management:**

* MongoDB
* Firebase Cloud Messaging for Alert System

1. **Tools:**

* OpenCV for Image Processing
* Firebase Cloud Messaging (FCM) for Alert System

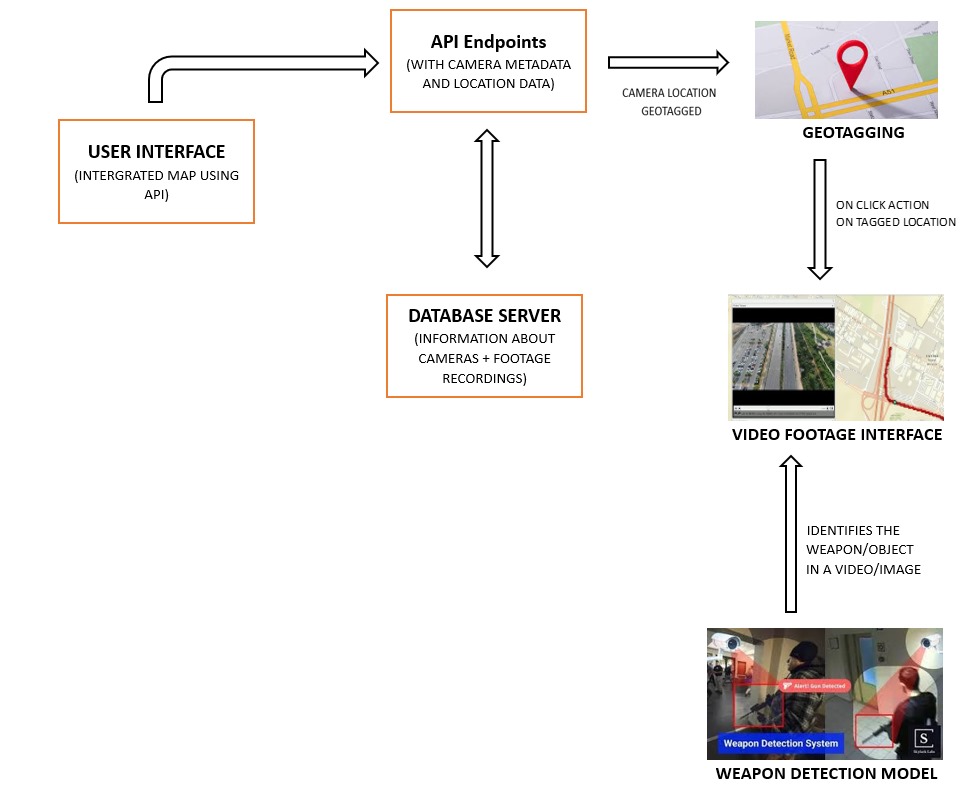
1. **Geospatial Mapping:**

* Maps JavaScript API

**Team Members & Responsibilities:**

* Hetashi Guru Singh Pal – Front-end Development (UI)
* Jashandeep Singh – Back-end Development
* Akash Kapoor – Database and API Integration
* Naman Raj Yadav – Security Operations

**Flow Chart / Graphical Representation:**



**SCHEDULE:**

