**A SYNOPSIS ON**

**Picture 3**

**Real-Time Traffic Event Reporting and Verification System For Intelligent Traffic Management System**

**Picture 2**

**Submitted in partial fulfilment of the requirement for the award of the degree of**

**BACHELOR OF TECHNOLOGY**

**IN**

**COMPUTER SCIENCE & ENGINEERING**

**Submitted by:**

**Student Name 1: Lakshita Tak** **University Roll No:2018905**

**Student Name 2: Isha Sharma** **University Roll No:2018839**

**Student Name 2: Aditya Bhatt** **University Roll No:2018636**

**Student Name 2: Shrinkhal** **University Roll No:2019123**

***Under the Guidance of***

**Mr. Piyush Agarwal**

**Assistant Professor**

**Project Team ID: MP24CSE044**



**Department of Computer Science and Engineering**

**Graphic Era (Deemed to be University)**

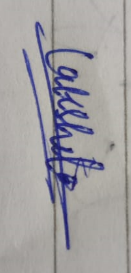
**Dehradun, Uttarakhand**

**October-2024**



**CANDIDATE’S DECLARATION**

We hereby certify that the work which is being presented in the Synopsis entitled **“Real-Time Traffic Event Reporting and Verification System For Intelligent Traffic Management System”** in partial fulfillment of the requirements for the award of the Degree of Bachelor of Technology in Computer Science and Engineering in the Department of Computer Science and Engineering of the Graphic Era (Deemed to be University), Dehradun shall be carried out by the undersigned under the supervision of **Mr. Piyush Agarwal , Assistant Professor**, Department of Computer Science and Engineering, Graphic Era (Deemed to be University), Dehradun.

Lakshita Tak 2018905 

Shrinkhal 2019123 

Isha Sharma 2018839 Picture 6

Aditya Bhatt 2018636 Picture 12

The above mentioned students shall be working under the supervision of the undersigned on the **“Real-Time Traffic Event Reporting and Verification System For Intelligent Traffic Management System”**

Signature Signature

**Supervisor** **Head of the Department**

**Internal Evaluation (By DPRC Committee)**

**Status of the Synopsis:** Accepted / Rejected

**Any Comments:**

**Name of the Committee Members: Signature with Date**

**Table of Contents**

|  |  |  |
| --- | --- | --- |
| **Chapter No.** | **Description** | **Page No.** |
| Chapter 1 | Introduction and Problem Statement | 4-7 |
| Chapter 2 | Background/ Literature Survey | 8-9 |
| Chapter 3 | Objectives | 10 |
| Chapter 4 | Software Requirements | 11-13 |
| Chapter 5 | Possible Approach/ Algorithms  Future Scope | 14-15 |
|  | References |  |

**Chapter 1**

**Introduction and Problem Statement**

**Introduction**

1. In today's scenario, the rapid exchange of information during emergencies holds a strategic role in saving lives and reducing damage. With rising accidents, crimes, and natural disasters, the conventional methods of reporting have often been found incompetent, causing procrastination leading to tragic results. As mobiles touch every life across the globe, there's a great scope for using this platform for real-time reporting and verification of the critical events.
2. This project develops a real-time event reporting and verification system-the mobile application that empowers users to report various events, such as traffic, accidents, rallies, criminal activities, environmental hazards, etc., directly to the concerned local authorities. Report submitted through the application is verified by nearby users while being transmitted for it to arrive at the emergency services.
3. This double verification layer is Vital to the information being communicated to become more believable, and thus the false alarms that might be brought to emergency responding entities as well as cause confusion in cases.
4. The proposed system would connect the most advanced technologies; Machine Learning, and Geolocation Services. The system shall classify different events depending on their nature and priority with the use of ML algorithms so that the application can study the patterns learned by the reported events thus putting them into broad categories.
5. Considering event type and event seriousness, among others. For instance, serious events such as violent crimes or road accidents involving fatalities on public roads should be dealt with before less critical ones. This will, therefore allow for timely response by emergency workers.
6. This application also uses real-time communication that enables a reporter within 100 meters of an incident to clarify reports, a mechanism that increases community engagement and validity of information received by autonomies. The system aims to significantly increase the speed of responses as confirmed reports are sent to the relevant emergency services, and guarantees that aid reaches people as soon as possible.

Furthermore, this application will enable analysis and visualizations of data about events, thus enabling identifications on event frequency, types, and other trends taking place in a given location. With the use of tools such as Google Maps API, users and authorities can easily describe locations of incidents.

enabling better insights into local safety conditions. Generally, this project aims at manipulating technology to generate a more responsive and responsible system for the reporting of events,into public safety and community resilience.

**Problem Statement**

1. Ineffective and unreliable systems for reporting and verifying events pose significant challenges in emergency response management. Current methods rely considerably on traditional modes of communication, such as phone calls or direct notifications, which are plagued with delays and inaccuracies. Reports may filter into the authorities without proper context or verification, and this can lead to misinformed responses and wasted resources. For example, false alarms can divert much-needed emergency services away from actual incidents, putting public safety at risk and eroding community confidence in these services.
2. In built-up areas where incidents frequently occur, even a short delay in verifying reports can leave emergency situations much worse than they need be. Emergency services, even today, cannot respond in timely and accurate manners without information, first responders may not get a proper view of the situation well, hence rushing to wrong conclusions and even disaster consequences. Moreover, without a standardized system for user participation in verification implies that most reports are unverified, and this does increase the complexity of the response process.
3. There is also less involvement of the community in the reporting process. Many people may see incidents, but be powerless to report them or take any action whatsoever due to a lack of accessible reporting mechanisms. As such, valuable first-hand accounts and photographic evidence would be wasted and the effectiveness of the emergency response efforts would decrease.
4. In the event that there is no incentives scheme put in place for verification, then respondents will lack a motivation to report, thereby lowering further the reports' credibility.
5. when the analytics tools used for classification and ranking of reported cases are absent, as it may be the case with the authority, then the authority may not respond appropriately. As shall be pointed out without insight Emergency services therefore cannot allocate resources efficiently nor strategize on reducing future risks based on the frequency and kinds of events.

To overcome such challenges, the proposed Real-Time Event Reporting and Verification System will establish an integrated system where users who report events will also be empowered to verify that event collaboratively through the real-time communication of machine learning for event classification and a structured verification.

In the process, the aim of this project is the integration of community with the emergency services to offer a more responding, safer environment to everyone.

One of major challenges of emergency response management is the lack of an efficient and reliable system for reporting and verifying events. Current systems rely heavily on the conventional communication modes - like a direct call or notification resulting in an enormous delay and an inaccuracy in reporting. These reports may reach authorities with little context or verification thus eliciting poor responses and waste of resources. For instance, false alarms can divert critical emergency services away from real incidents, hence creating a danger to public safety and eroding the confidence of citizens in such services.

Being unable to verify reports quickly in areas where incidents are frequent can cause delays that worsen the emergency situation. First responders will, therefore, be unable to effectively assess the situation if not given current, accurate information, and may, therefore, end up in disastrous outcomes. Moreover, the unavailability of a smooth mechanism to guarantee user involvement in verification implies that many reports go unverified, complicating the response efforts.

Another is the engagement of community members; very few are engaged in reporting activities whereby an individual may see an occurrence but cannot do anything because there are no accessible reporting mechanisms. Valuable accounts and photo evidence may be lost, hence reducing the effectiveness of emergency response efforts. Moreover, this lack of incentives for validating the correctness of reports would deter people from participating and thus further reduce the credibility of the reports submitted.

At the same time, without developed analytical tools for classifying and ranking reported incidents, the ability of authorities to respond rapidly is seriously impeded. Without any statistical analysis of the rates or nature of incidents, emergency services cannot strategically deploy their resources or even make strategic moves to avoid future risks.

Addressing such matters, the real-time event reporting and verification system developed herein will bring together an integrated platform that allows for easy and relatively rapid incident reporting, yet empowers users to verify such reports in a collaborative way. Based on the integration of real-time communication, automatic event classification through machine learning, and structured verification, this project closes the gap for community participants with emergency services, making this environment safer and more responsive to all.

**Chapter 2**

**Background/ Literature Survey**

* To improve traffic flow and reduce congestion in large cities, accurate congestion measurement is critical. [1] Despite advancements in traffic management systems (TMS), challenges remain in measuring congestion levels without overloading control networks with data. As a solution, TRAFFIC, a system that employs an ensemble of classifiers for inter-vehicle communication, is proposed to assess congestion and optimize traffic flow.
* To address urban traffic congestion caused by rising vehicle numbers and fixed traffic signals, local de-congestion protocols have been shown to increase road capacity.[2] utilizing RFID technology and the Internet of Things (IoT), provide effective methods for traffic management. Additionally, intelligent transport systems (ITS) leverage vehicular adhoc networks (VANET) to control congestion and manage traffic signals dynamically, offering a more adaptive approach to improving traffic flow.
* To enhance traffic safety, [3] the article proposes a real-time traffic conflict prediction system using Advanced Driver Assistance Systems (ADAS) and machine learning to analyze in-car sensor data. The system employs a Regional-Convolution Neural Network (R-CNN) for vehicle tracking, challenging traditional methods that rely solely on time-to-collision (TTC). The top Deep Neural Network (DNN) model achieved 94% accuracy on data from the UK's M1 motorway, demonstrating the effectiveness of DNNs in improving safety for automated and connected vehicles.
* To improve traffic efficiency and safety, [4] Advanced Driver Assistance Systems (ADAS) and Machine Learning for real-time traffic conflict prediction within Intelligent Transportation Systems (ITS) are proposed. Many current methods rely solely on time-to-collision (TTC) and ignore other critical traffic dynamics. The proposed solution leverages data from in-car sensors and security cameras, using Regional-Convolution Neural Networks (R-CNN) for better conflict identification and prediction, highlighting the need for advanced algorithms in enhancing safety for automated vehicles.
* To enhance intelligent transportation systems,[5] the proposed framework features a mobile app functioning as a Smart Traffic Assistant (STA) that collects real-time traffic data and shares it with the Traffic Information Center (TIC) via the Traffic Information Exchange Protocol (TIEP) and image capture from cameras. The TIC aggregates data from various sources and optimally weighs it using Shannon entropy to predict short-term traffic status based on key event factors. This user-driven system improves the accuracy of traffic information and spatiotemporal coverage while offering a cost-effective solution for traffic management.
* To evaluate dynamic traffic management systems, [6] the proposed paper introduces a simulation laboratory that integrates traffic control models, driver behavior, and network performance. It features a Traffic Management Simulator based on MIT's Traffic Flow Simulator, a Surveillance System Module, and estimation tools. The lab generates scenarios based on traffic demand, enabling performance measurement and design refinement. Applications include analyzing lane blockage and the impacts of lane use signage and electronic tolling, offering advanced traffic management solutions through offline evaluations.
* To enhance traffic management, [7] a distributed autonomous application model using Service-Oriented Architecture (SOA) and multi-agent systems. It features Intelligent Junctions (IJ) for traffic management and Semantic Web Services (SWS) for communication. The paper demonstrates efficient tracking of missing vehicles by sharing information among junctions, showcasing TMIS's effectiveness. Additionally, the framework applies to emergency response systems, with potential for further research on reconfigurability and dynamic traffic loads.

**Chapter 3**

**Objectives**

**1.To develop an interface for Reporting Road Events:**

The proposed mobile application aims to improve public safety by enabling users to swiftly report emergencies such as accidents, crimes, and natural disasters. Users can easily submit critical details like location, time, and event type, significantly reducing the delay between incident occurrence and reporting. This rapid communication allows for quicker mobilization of emergency services. Real-time communication features will also provide users with instant updates from local authorities, enhancing coordination and building trust in the system.

**2.To create peer Verification Process:**

To ensure reliable reporting, the app will implement a peer verification process. Users within a 100-meter radius of an event can confirm reports before they are sent to authorities, enhancing information accuracy and reducing false alarms. Additionally, machine learning algorithms will classify events by type and severity, assisting authorities in prioritizing responses and identifying patterns for future preparedness.

**3.To manage data and visualize it to the users:**

The application will also focus on data management and visualization. Users will be limited to uploading one image per report to prevent system overload and improve data quality. Integrating tools like the Google Maps API will allow real-time analysis and visualization of incidents, aiding decision-making for both users and authorities. A user rating mechanism will incentivize responsible reporting, while features for updating or rolling back reports will promote accuracy. Overall, the system empowers citizens to engage proactively with their communities, enhancing public safety and awareness.

**Chapter 4**

**Software/Hardware Requirements**

The Real-Time Traffic Event Reporting and Verification System is a mobile application that enables users to report traffic events, such as accidents, crimes, and natural disasters, in real-time. The system aims to provide a reliable and efficient way of reporting and verifying events, ensuring timely response from emergency service

**Software Requirements:**

**1.Development Environment:**

* + Windows 10/11 (64-bit) or higher, macOS or Linux (Ubuntu 18.04 or higher).

**2.IDE/Code Editors**

* + Visual Studio Code or Sublime Text for general coding and frontend or backend development.

**3.Programming Languages**

* JavaScript/TypeScript for mobile development with React Native or Flutter.
* Backend Services: Python

**4.Frameworks and Libraries:**

* React Native or Flutter for cross-platform mobile application development
* Backend API Development: Node.js/Express.js or Django
* Data Storage: Firebase
* For integrating machine learning algorithms: TensorFlow/PyTorch

**5.Version Control:**

* + Git and GitHub/GitLab for version control and collaboration.

**6.Emulators and Simulators:**

* + Android Emulator is included in Android Studio.

**Hardware Requirements:**

* **Server:**
  + Minimum 2 GB RAM and 2 CPU cores.
  + 64-bit operating system.
  + Database
  + Minimum 1 GB storage.
  + 64-bit operating system.
* **Client:**
  + Minimum 1 GB RAM and 1 CPU core.
  + 64-bit operating system.
  + Network Requirements
  + Internet Connection
  + Minimum 3G or 4G or 5G network connection.
  + Wi-Fi connection for optimal performance.
  + Bandwidth
  + Minimum 1 Mbps upload and download speed
* Security Requirements
  + Data Encryption
  + Use of encryption algorithms such as AES-256 or RSA.
  + Encryption of data in transit and at rest.
  + Authenticati
  + Multi-factor authentication for users.
  + Access Control
  + Use of role-based access control to restrict user access.
  + Secure APIs and token-based authentication.

**Chapter 5**

**Possible Approach/ Algorithms**

**1.Generic Overview:**

The Real-Time Event Reporting and Verification System is a system that allows prompt communication, incident verification, dissemination of information in real time to relevant authorities. The project involves the integration of multiple domains - real-time communication, machine learning and cybersecurity with the aim of creating a community reporting system that verifies incidents before sending them to emergency responders.

*(Refer to fig.1 for overview.)*

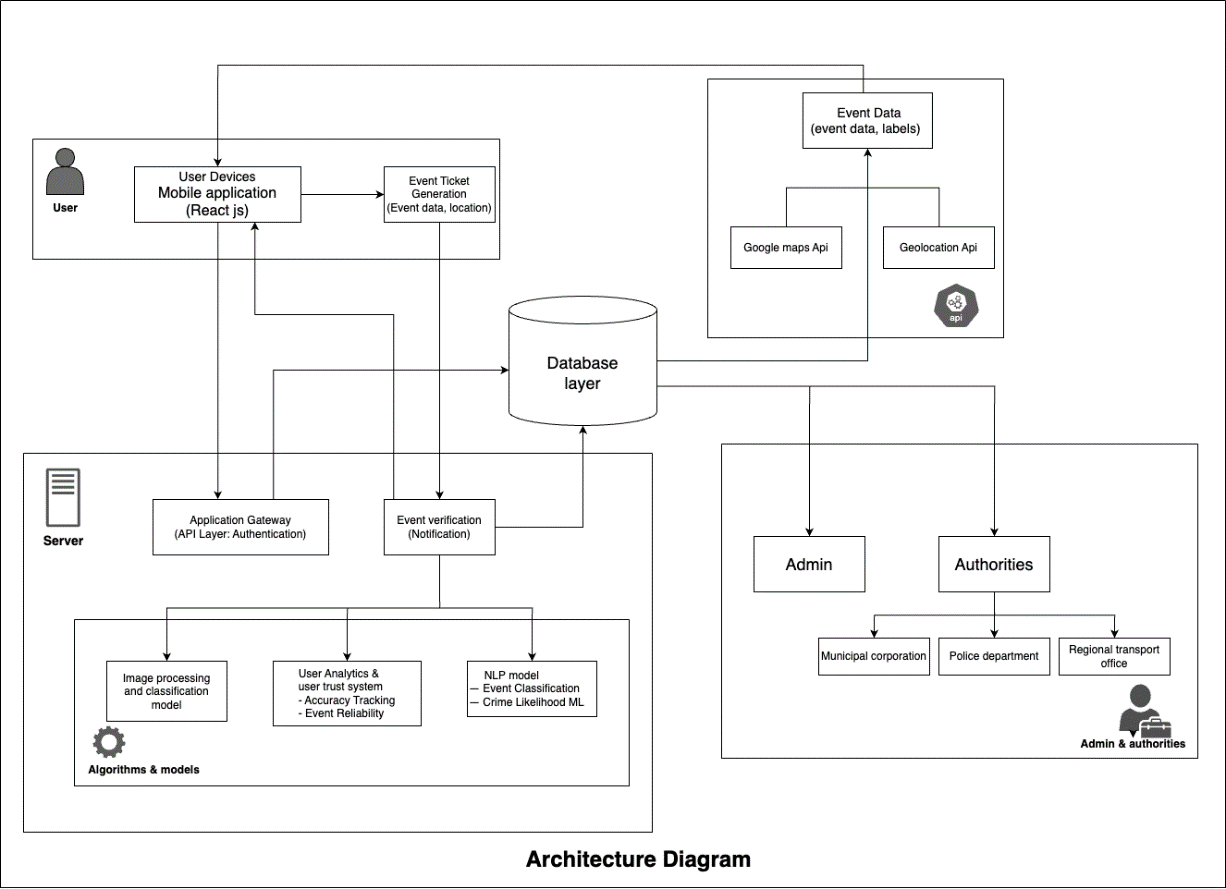
The real-world application centers around the workflow illustrated by the following stages:

Stage 1: A user, through the mobile application, reports an event that has taken place.

Stage 2: Users who are within 100 meters of the said location are informed to eyewitness and confirm the event.

Stage 3: These events that are witnessed are further classified using machine learning algorithms and are forwarded to the appropriate authorities.

Stage 4: Once these verified details reach the authorities, they will decide what could be prioritized in terms of seriousness and credibility.



*Fig.1*

**2. Event Reporting and Data Validations:**

All data in this case will include information such as images, timestamps, and geolocation details about the event reported by the user. The app also handles the problem of one image uploaded per event thread on an IP address to prevent spam and data overload. Here, therefore, lies a challenge in managing the flow of data and ensuring that it has been validated satisfactorily before passing it for further processing. This can be handled through the following:

IP Filtering Algorithm: A simple algorithm associates each user's IP address with a unique event thread ID. No multiple uploads are allowed for the same event. Hash map or dictionary data structure is used to store the IP-event mappings. Once an event thread is created before allowing an upload, the system checks the IP address against the hash map.

Data Validation: The uploaded files, including images and texts, will be validated as to the type of file and size in addition to extracting metadata for images. This would be possible from image extractions to validate the legitimacy of the information uploaded, including the verification of geotags.

**3. Data Communication over Real-Time between devices:**

To enable real-time intercommunication between devices, a Firebase Realtime Database will be used. It is appropriate for the interoperation of users because their interaction can be synchronised and data can be synchronized across all the connected devices in real time. Firebase can handle real-time updates on a mobile application and can synchronize it instantly with all the connected devices.

Algorithm: When the information is forwarded through other persons in a scenario it informs all the devices within a specified 100-meter radius. This can be done by a geographical query system whereby the device captures the location of the user on latitude and longitude. The Haversine formula can be applied to ascertain how far the reported event is from other users in ascertaining the distance ; only alerting those with a stipulated distance.

**4. Validation of the Event by Users :**

The system would have community-driven verification as one of the critical features. In strict terms, after an event is reported, it invites nearby users to verify the report, but then fraudulent verification would have to be avoided with a trust-based system.

Trust Factor Algorithm: Users who verify events correctly most of the time will have a better trust score. However, initially, users begin with default scores. For each event verified correctly with the use of consensus or authority feedback, the trust score increases. If a user verifies false reports all the time, then that score decreases. This is almost like a reputation management algorithm, which is the heart and soul of crowdsourcing platforms.

Event-based Consensus Verification Algorithm: A fixed number of users must verify the incident within a defined time window so that it's sent to the concerned authorities. This can be achieved by using a threshold-based consensus algorithm. Once the number of positive verifications exceeds a pre-defined threshold (say, five users), the event is marked as verified and forwarded to authorities.

**5. ML algorithms used for Event Classification:**

This would thus make data gathered through machine learning significantly dependent for categorizing and sorting events. Actually, there are quite a number of algorithms that may be applicable depending on what sort of data is available: image classification, text analysis, and so on. A few of the best approaches, accordingly, include:

Convolutional Neural Networks (CNNs): Images can be utilized for event classification using CNN. For example, an accident image, a crime image, or a natural disaster image dataset can be used to train a CNN that is able to classify the type of event based solely on an uploaded image. An existing pre-trained CNN model such as ResNet and MobileNet could be fine-tuned for this particular application. TensorFlow or PyTorch would be great frameworks for implementation.

Support Vector Machines (SVMs): The nature of the event, involving textual data or sensor data, such as time and location, can be classified by training an SVM to distinguish among different types of events along particular features. SVM is especially useful for classification with two or more classes in the case.

Naive Bayes Algorithm: It can be used for classifying text-data-based events, such as reports, based on the description of an event by a user. Naive Bayes is very efficient while handling text data; this is the reason it will do pretty well with the reports of the user.

Time-Series Analysis: All events that involved a time component could be categorized by time-series analysis. For instance, crimes could appear at specific times or dates, and a system can be developed to notice the trends in the event occurrence whether based on the day, day of the week, or seasonal trend. For time-based predictions, the best use of LSTM models is herein.

**6. Visualization and Event Data Representation :**

Reported and confirmed events will then be represented on a map for visualization by users and the authorities concerned. The Google Maps API will be used in the facilitating of geographic visualizations of the incidents.

Heat map algorithm: The events may be visualized through a heat map approach whereby all the places with high events of frequency (high crime or accident rates, for example) can be highlighted. That way, the authorities can delegate their resources to critical areas.

**7. Rollback Feature for Users:**

The users who have reported the event can roll back their report in case of certain non-criminal cases where they realize that the report is incorrect. This is managed by

Event Ticket Management Algorithm: There is an event ticket issuing to each of the events. The reporting user may cancel or update an event ticket, which is ruled by a state machine algorithm, whereby an event must be at one of the following stages: pending verification, verified, or canceled.

**8. Ranking Events for Officials:**

Verified events are forwarded to the interested authorities for actions to be taken. Machine learning classifiers and user trust scores will allow an official to rank prioritized high-risk events.

Priority Queue Algorithm: Events enter a priority queue when they have been classified and ranked according to the level of importance. Serious events like crimes and extreme accidents would be given higher priority to ensure that the response is quicker from the authorities.

The Real-Time Traffic Event Reporting and Verification System is a mobile application that enables users to report traffic events, such as accidents, crimes, and natural disasters, in real-time. The system aims to provide a reliable and efficient way of reporting and verifying events, ensuring timely response from emergency services.

**Application Functionalities:**

* User Registration and Login
* Users can register and log in to the application using their username and password.
* Users can reset their password if forgotten.
* Event Reporting
* Users can report traffic events, including accidents, crimes, and natural disasters.
* Users can provide details about the event, including location, time, and description.
* Users can upload images or videos as evidence.
* Peer Verification Process
* Users within a 100-meter radius of the reported event can verify the report.
* Verified reports are sent to authorities for action.
* Data Management and Visualization.
* The system stores and manages event data, including location, time, and description.
* The system provides a map view of reported events, allowing users and authorities to visualize event locations.
* Machine Learning-based Event Classification
* The system uses machine learning algorithms to classify events based on type and severity.
* Classified events are prioritized for response by authorities.
* Real-time Communication
* The system enables real-time communication between users and authorities.
* Users receive updates on the status of their reported events.
* Trust-based System
* Users who verify events correctly earn trust scores.
* Users with high trust scores are given more weight in the verification process.
* Non-Functional Requirements
* Performance
* The system responds to user input within 2 seconds.
* The system can handle a minimum of 100 concurrent users.
* Security
* The system ensures the confidentiality, integrity, and availability of user data.
* The system uses encryption to protect data in transit and at rest.

**Usability:**

* The system is user-friendly and easy to navigate.
* The system provides clear instructions and feedback to users.
* Scalability
* The system can scale to accommodate increasing user traffic.
* The system can handle a minimum of 1000 concurrent users.
* Technical Requirements
* Programming Languages
* Java or Kotlin for Android app development.
* Swift or Objective-C for iOS app development.
* Database
* Firebase Realtime Database or MySQL for storing event data.
* Machine Learning Framework
* TensorFlow or PyTorch for event classification.
* Map API
* Google Maps API for map visualization.
* Operating System
* Android 8.0 or later for Android app.
* iOS 12 or later for iOS app.

**FUTURE SCOPE**

**1.Security Enhancements**

The new versions of the Real-Time Event Reporting and Verification System therefore must have the necessary robust security measures to not only ensure integrity but also the trustworthiness of the system. It will handle sensitive data such as coordinates and images alongside user verification ratings. The focus should be security improvements so that the system continues mishandling the user's data and correct reporting of events, yet do not make it easy for malicious players to exploit it.

**2. Encryption of Data in Transit and at Rest**

The other area of improvement for the future would be data encryption, where data is encrypted to both in-transit and at-rest. Currently, the system uses real-time protocols of communication with other users and authorities. The application of E2EE would ensure that all data transferred between devices is completely encrypted and inaccessible to people who are not permitted to access it, even if intercepted during transmission.

The Data stored in Firebase must be encrypted at rest: images, reports, and even user trust scores. This prevents data breaches. Encryption standards like AES-256 or RSA protect the user information against cyberattacks and compliance with data protection regulations like GDPR.

**3. Multi-Factor Authentication (MFA) for Users**

An added layer of authentication will be provided to the system due to the implementation of MFA, in order to further elevate security. At the moment, it is still using a typical scheme of the username-password type and vulnerable to phishing attacks, password leakage, or brute-force attempts. However, with the implementation of MFA in a system, the access to it could greatly be minimized by unauthorized persons because users will have to present not one but two forms of evidence of their identity-be it an OTP sent to the user's mobile or a biometric scan.

MFA will be very helpful in securing administrative accounts and eliminating fraudulent activity since only the authorized users shall be enabled to report or verify events.

**4.Secure APIs and Token-Based Authentication**

The devices and the servers rely heavily on APIs for information exchange. It is very important to ensure that such APIs are secured as they make up a big part of the overall security of the platform. Future developments should be based on OAuth 2.0, or JWT, or JSON Web Tokens, for token-based authentication, ensuring users need to authenticate themselves by using valid tokens to access platform features, thus making it difficult to have any unauthorized access or session hijacking.

Furthermore, rate limiting and IP filtering should be implemented to prevent Denial-of-Service attacks or spamming from badly straining the system and jeopardizing availability.

**5. Preventing False Reporting and Spam**

A major security challenge for any reporting system is ensuring that it is not flooded with false reports or spam, which could overload the system or cause delays in verifying and escalating real incidents. While the current version implements a single upload per IP per event thread, more sophisticated solutions can be developed to identify and filter out malicious or fraudulent users.

Future generations of the systems will be able to leverage AI-based anomaly detection algorithms to capture users who constantly file false claims or malicious activity and prevent their ip addresses from spamming images or spreading any kind of misinformation. Further, a blockchain-based reputation management system can introduce an immutable record of user behavior that will further secure the system against manipulation.

**6. Geolocation and User Privacy**

Although the system employs geolocation data to establish close proximity to an event, functionality from this regard needs to be in tandem with the features concerned with user's privacy. The future upgrades should be toward making sure that implementation location obfuscation techniques wherein location of a user is anonymized unless it has to be verified for authenticity purposes. Techniques such as differential privacy may also enable the system to provide precise services based on a location without letting an attacker or third party know the exact location.

**References**

[1] Ravish, Roopa, and Shanta Ranga Swamy. "Intelligent traffic management: A review of challenges, solutions, and future perspectives." *Transport and Telecommunication Journal* 22, no. 2 (2021): 163-182.

[2] Atta, Ayesha, Sagheer Abbas, M. Adnan Khan, Gulzar Ahmed, and Umer Farooq. "An adaptive approach: Smart traffic congestion control system." Journal of King Saud University-Computer and Information Sciences 32, no. 9 (2020): 1012-1019.

[3] Formosa, Nicolette, Mohammed Quddus, Stephen Ison, Mohamed Abdel-Aty, and Jinghui Yuan. "Predicting real-time traffic conflicts using deep learning." Accident Analysis & Prevention 136 (2020): 105429.

[4] Wan, Shaohua, Xiaolong Xu, Tian Wang, and Zonghua Gu. "An intelligent video analysis method for abnormal event detection in intelligent transportation systems." IEEE Transactions on Intelligent Transportation Systems 22, no. 7 (2020): 4487-4495.

[5] Lee, Wei-Hsun, Shian-Shyong Tseng, and Wern-Yarng Shieh. "Collaborative real-time traffic information generation and sharing framework for the intelligent transportation system." Information Sciences 180, no. 1 (2010): 62-70.

[6] Ben-Akiva, Moshe E., Haris N. Koutsopoulos, Rabi G. Mishalani, and Qi Yang. "Simulation laboratory for evaluating dynamic traffic management systems." Journal of Transportation Engineering 123, no. 4 (1997): 283-289.

[7] Elci, Atilla, Behnam Rahnama, and Amirhasan Amintabar. "Tracking Reported Vehicles in Traffic Management and Information System using Intelligent Junctions." J. Inf. Sci. Eng. 26, no. 3 (2010): 883-896.