PEOPLES'S DEMOCRATIC REPUBLIC OF ALGERIA

MINISTRY OF HIGHER EDUCATION AND SCIENTIFIC RESEARCH
University of Saad Dahleb -Blida 01
Faculty of Sciences
Department of Informatics

Proposal of an emergency call system taking advantage of 4G LTE networks

Domain :MI
Affiliation : Informatics
Specialsation : Computer systems and networks

Thesis Submitted by :
TALBI Baya
ZAOUAK Farah Imene

Under the supervision of :
DR.ALLA Tiberkak
Professeur Mohamed Ould-Khaoua

Members of jury :
OUL DAISSA Ahmed
BENYAHYIA Mohammed

Blida 2022-2023

Acknowledgment

Prior to anything else, we wish to express our gratitude to ALLAH for giving us the perseverance and strength to complete this thesis.

we would like to thank our supervisor Mr. Allal TIBERKAK, and our promoter Mr. Mohamed OULD-KHAOUA, for their precious advice and help throughout the work period.

Our sincere thanks also go to the members of the jury for the interest they have shown in our research by agreeing to act as judges.

Finally, we would like to thank to our families and all the people who took part, directly or indirectly, in the in the realization of this work.

ملخص

تهدف هذه الرسالة إلى معالجة التحديات التي تواجهها أنظمة مكالمات الطوارئ في الجزائر واقتراح حل لتعزيز فعاليتها واستجابتها من خلال اقتراح نظام مكالمات طوارئ متقدم يشتمل على التقنيات الحديثة مثل تقنيات الجيل الرابع (4G LTE) و WebRTC (اتصال الويب في الوقت الفعلي). يتيح استخدام شبكات الجيل الرابع نقل البيانات بشكل سريع وموثوق ، بينما تسهل WebRTC اتصالات الصوت والفيديو والبيانات في الوقت الفعلي. سيركز النظام المقترح على الاستخدام الواسع النطاق للأجهزة المحمولة في الجزائر. سيتم تصميم النظام كتطبيق محمول سهل الاستخدام يمكّن الأفراد من الإبلاغ عن حالات المستعجلة ومشاركة المعلومات الأساسية والانخراط في اتصال في الوقت الفعلي مع مصلحة الاستعجالات. يهدف النظام المقترح إلى تسهيل الإبلاغ عن حالات الاستعجالات ، مما يؤدي إلى أوقات استجابة أسرع وتحسين إدارة مصلحة الاستعجالات. الهدف هو تسخير فوائد شبكات الجيل الرابع واستخدام الهاتف المحمول لتعزيز الاستجابة الشاملة وفعالية الاتصالات في حالات المستعجلة في الجزائر. من خلال الجمع بين التطورات في تقنية الجيل الرابع ، وإمكانيات WebRTC والاستخدام الواسع للهواتف المحمولة، يهدف النظام المقترح إلى إحداث ثورة في أنظمة مكالمات الاستعجالات في الجزائر. ويسعى إلى التغلب على قيود الأنظمة الحالية ، وتقليل أوقات الاستجابة ، وتحسين تخصيص موارد مصلحة الاستعجالات ، وإسعاف المصابين في اسرع وقت ممكن. ستساهم نتائج وتوصيات هذه الأطروحة في تطوير أنظمة اتصالات الاستعجالات في الجزائر وتوفر أساسًا للتحسينات المستقبلية في السلامة العامة.

الكلمات المفتاحية: نظام مكالمات الاستعجالات ،(اتصال الويب في الوقت الفعلي) WebRTC ، (شبكات الجيل الرابع) 4G LTE ، الهواتف المحمولة ، اتصالات الوسائط المتعددة ، الإبلاغ عن الحوادث.

Abstract

This thesis aims to address the challenges faced by emergency call systems in Algeria and propose a solution to enhance their effectiveness and responsiveness by proposing an advanced emergency call system that incorporates modern technologies such as 4G LTE and WebRTC (Web Real-Time Communication). The utilization of 4G LTE networks enables fast and reliable data transmission, while WebRTC facilitates real-time audio, video, and data communication. The integration of 4G LTE and WebRTC technologies allows for seamless and efficient communication during emergency situations. The proposed system will focus on mobile phone accessibility, taking into consideration the widespread use of mobile devices in Algeria. The system will be designed as a user-friendly mobile application that enables individuals to report emergencies, share essential information, and engage in real-time communication with emergency operators. By leveraging the widespread use of mobile phones, the proposed system aims to increase the accessibility and ease of reporting emergencies, leading to faster response times and improved emergency management. The objective is to harness the benefits of 4G LTE networks and mobile phone usage to enhance the overall responsiveness and effectiveness of emergency communication in Algeria. By combining the advancements in 4G LTE technology, the capabilities of WebRTC, and the widespread use of mobile phones, the proposed system aims to revolutionize the emergency call systems in Algeria. It seeks to overcome the limitations of existing systems, reduce response times, improve emergency resource allocation, and ultimately save lives. The findings and recommendations of this thesis will contribute to the advancement of emergency communication systems in Algeria and provide a foundation for future enhancements in public safety and emergency management.

Keywords: emergency call system, WebRTC, 4G LTE networks, mobile phones, multimedia communication, incident reporting.

Résume

Cette thèse vise à relever les défis auxquels sont confrontés les systèmes d'appel d'urgence en Algérie et à proposer une solution pour renforcer leur efficacité et leur réactivité en proposant un système d'appel d'urgence avancé intégrant des technologies modernes telles que la 4G LTE et le WebRTC (Web Real-Time Communication). L'utilisation des réseaux 4G LTE permet une transmission de données rapide et fiable, tandis que WebRTC facilite la communication audio, vidéo et de données en temps réel. Cette intégration des technologies 4G LTE et WebRTC permet une communication transparente et efficace lors de situations d'urgence. Le système proposé se concentrera sur l'accessibilité des téléphones mobiles, en tenant compte de l'utilisation généralisée des appareils mobiles en Algérie. Le système sera conçu comme une application mobile conviviale qui permettra aux individus de signaler les urgences, de partager des informations essentielles et de communiquer en temps réel avec les opérateurs d'urgence. En tirant parti de l'utilisation généralisée des téléphones portables, le système proposé vise à accroître l'accessibilité et la facilité de signalement des urgences, ce qui permet d'accélérer les temps de réponse et d'améliorer la gestion des urgences. L'objectif est d'exploiter les avantages des réseaux 4G LTE et de l'utilisation du téléphone mobile pour améliorer la réactivité et l'efficacité globales des communications d'urgence en Algérie. En combinant les avancées de la technologie 4G LTE, les capacités du WebRTC et l'utilisation généralisée des téléphones mobiles, le système proposé vise à révolutionner les systèmes d'appel d'urgence en Algérie. Il vise à surmonter les limites des systèmes existants, à réduire les temps de réponse, à améliorer l'allocation des ressources d'urgence et, en fin de compte, à sauver des vies. Les conclusions et les recommandations de cette thèse contribueront à l'avancement des systèmes de communication d'urgence en Algérie et fourniront une base pour les améliorations futures de la sécurité publique et de la gestion des urgences.

Mots-clés : système d'appel d'urgence, WebRTC, réseaux 4G LTE, téléphonie mobile, communication multimédia, rapport d'incident.

Contents

Chapter 4 implementation

Chapter 5: Result

List of figures

List of abbreviations

WebRTC: Web Real-Time Communication

P2P: Peer to peer

HTTP: Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

ICE: Interactive Connectivity Establishment

NAT: Network Address Translation

STUN: Simple Traversal of User Datagram Protocol through NAT

TURN: Traversal Using Relay NAT

LTE: Long Term Evolution

4G: fourth generation of wireless cellular network

GPS: Global Positioning System

SCSS: Sassy Cascading Style Sheets

DBMS: A database management system

SCTP: Stream Control Transmission Protocol

SRTP: Secure Real-time Transport Protocol

General introduction

With the rapid urbanization and technological advancements witnessed globally, the concept of smart cities has emerged as a solution to address the complex challenges posed by emergencies. Smart city initiatives leverage innovative technologies and data-driven approaches to enhance the safety, security, and resilience of urban environments. These initiatives aim to transform cities into interconnected ecosystems that can proactively respond to emergencies and protect their inhabitants.

Emergency call systems are critical infrastructures that allow people to connect with emergency services and seek quick assistance during times of crisis. These systems are intended to process emergency calls efficiently, collect critical information, and dispatch appropriate resources to address reported situations, the situation of emergency call systems in Algeria presents distinct obstacles and constraints.

In Algeria, emergency call system administration and operation encounter severe challenges, resulting in unsatisfactory performance and limited effectiveness. The current situation of Algeria's emergency call services can be linked to a variety of causes, including infrastructure limits, organizational inefficiencies, and regulatory restraints. These obstacles impair the seamless operation of emergency call systems and their ability to offer timely and efficient emergency services to the public.

Problematic:

There are multiple challenges with Algeria's current emergency call services that affect their effectiveness and response. Inadequate infrastructure, limited coverage in rural regions, antiquated communication technologies, ineffective call routing and processing processes, and organizational weaknesses in emergency response coordination are among the obstacles. These challenges lead to prolonged response times, miscommunication, and inefficient use of emergency resources. As a result, the entire effectiveness of Algeria's emergency call services is jeopardized, potentially risking the safety and well-being of people in emergency situations.

Infrastructure constraints play a significant influence in Algeria's poor emergency call services. Inadequate telecommunications infrastructure, such as obsolete or unreliable networks, can lead to poor call quality, dropped calls, and even system breakdowns. The lack of robust network coverage in isolated or rural locations exacerbates the problem by limiting emergency services' accessibility and reach.

Organizational inefficiencies and insufficient resource allocation also add to the difficulties that Algerian emergency call services encounter. Inefficient call routing mechanisms, inadequate call center operator training, and a lack of coordination among emergency service providers can all lead to delays and miscommunication during an emergency. Furthermore, the lack of an emergency response structure and established processes might stymie seamless coordination and collaboration among various institutions participating in emergency management.

The limitations and problems of Algeria's emergency call services underline the need for considerable improvements and strategic interventions. To enable successful emergency call systems, it is critical to address infrastructure inadequacies, strengthen organizational capability, and develop strong regulatory frameworks. By overcoming these obstacles, Algeria can create a cutting-edge emergency call system that provides prompt and effective emergency response services, thereby protecting the well-being and safety of its people.

Proposed solution

This thesis proposes the development and implementation of an upgraded emergency call system leveraging modern technologies and improved operational frameworks to address the problems and improve Algeria's emergency call system. The proposed system will use cutting-edge communication technologies like WebRTC (Web Real-Time Communication) to enable real-time voice, video, and data transmission between callers and emergency operators.

The system will have a user interface that is simple to use that will be accessible via web or mobile applications, allowing individuals to easily report incidents and send critical information to emergency operators. The proposed system will also include sophisticated call routing algorithms to enable efficient and timely dispatch of emergency responders depending on geographical location, kind of emergency, and available resources.

Furthermore, the system will incorporate capabilities such as event reporting, where individuals can offer thorough incident descriptions and supplemental information to assist emergency responders in properly assessing and responding to the issue. The system will also include location tracking, allowing callers to share specific coordinates with operators, allowing for speedier response times and better resource allocation accuracy.

The suggested solution will be tailored to the specific needs and challenges of Algeria's emergency call systems. It will take into account the country's specific infrastructure limits, operational constraints, and regulatory frameworks. The proposed system aims to improve the overall effectiveness and responsiveness of Algeria's emergency call services by leveraging current technologies, streamlining operational processes, and encouraging collaboration across emergency response bodies.

Organization of the report

The remainder of the report is organized into four chapters:

- Chapter 1: Provides an overview of what an emergency call system should contain to be affective;
- Chapter 2: Provides a review of existing emergency call systems worldwide, comparison of their functionality, and identify the gaps and shortcomings in the existing systems;
- Chapter 3: Presents the design of the proposed system and its functionalities;

- Chapter 4: Presents the implementation process and explain the tools and programming languages used;
- Chapter 5: Presenting the proposed system;
- Conclusion: summary of the research findings and discuss the potential for future enhancement and scalability of the system.

Chapter 1

Emergency call system

1.1 Introduction

Good response systems for both small and large crises are typically divided into first responders (medical personnel, fire-fighters and police) and different command structures. A Command Centre often holds the highest authority, making strategic decisions. Close to the incident site, incident commanders and other experts adapt response to the situation at hand, handling local resources and constraints, making decisions, evaluating risks and following up on operation progress. For an incident to become an accident site someone has to perceive and define the situation as an accident. This someone – the first responder – is often a lay first responder [5].

Emergency management is commonly divided into different phases. Preparation means getting ready for imaginable future incidents, developing systems, managing resources, developing scenarios and plans and engaging in realistic training. Mitigation concerns actions taken to reduce the chances of an accident occurring (e.g. through risk assessment) or early attempts to make the consequences of an accident as small as possible. The Response phase covers actions carried out during the emergency, where joint efforts are made to save lives and minimize structural damage. Response may in turn be divided into sub- phases. How much time is needed for each of the phases Detection, Preparation, Response Travel and Clearance may affect outcomes. Finally, the Recovery phase concerns activities to restore infrastructure, people and their property to normal [5].

1.2 Design of The Emergency Call Systems:

A well-designed emergency call system based on a web app should possess several key components to ensure efficient and effective emergency response. The architecture of such a system typically includes a client-side web app, a help center server, an emergency responder dispatch system, and integration with a 4G LTE network, database, and WebRTC technology. The client-side web app enables users to initiate emergency calls and report incidents, utilizing WebRTC for secure real-time audio and video communication with the help center. The help center server acts as a central hub, facilitating the exchange of data between the client and the operator, while also utilizing the 4G LTE network for seamless connectivity. The integration with a database enables the storage and retrieval of incident information and related data, ensuring quick access during emergency responses. Additionally, the emergency responder dispatch system connects with the help center server to receive incident reports and dispatch available responders. By combining these elements, a comprehensive emergency call system empowers users to swiftly report emergencies, enables operators to effectively communicate with callers, and facilitates the dispatch of appropriate emergency responders in a timely manner.

1.3 WebRTC technology:

WebRTC (Web Real-Time Communication) is a free and open-source and supported by Apple, Google, Microsoft and Mozilla, amongst others. This page is maintained by the Google WebRTC team. With WebRTC, you can add real-time communication capabilities to your application that works on top of an open standard. It supports video, voice, and generic data to be sent between peers, allowing developers to build powerful voice- and video-communication solutions. The technology is available on all modern browsers as well as on native clients for all major platforms. The technologies behind WebRTC are implemented as an open web standard and available as regular JavaScript APIs in all major browsers. For native clients, like Android and iOS applications, a library is available that provides the same functionality [18].

1.3.1 Architecture of WebRTC:

WebRTC plays a crucial role in an emergency call system by enabling real-time audio and video communication between the client and the help center operators. In Figure 1 a simple architecture explaining functionality of WebRTC:

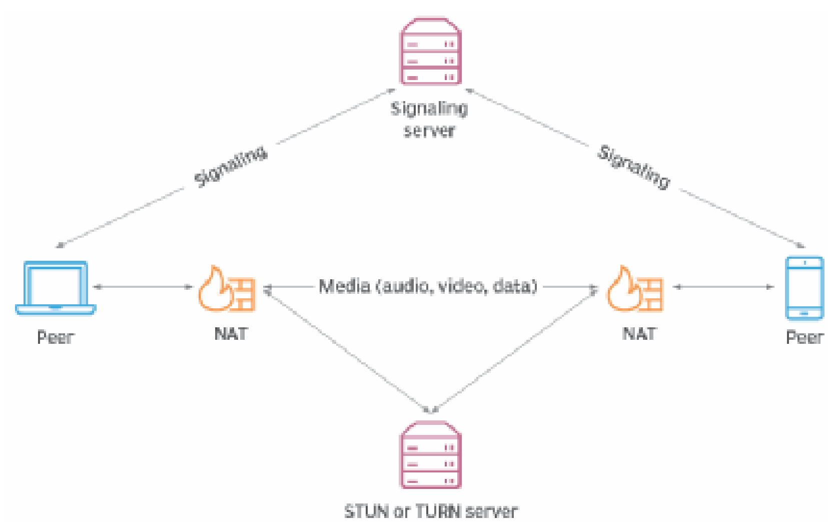

Figure 1: Functionality of WebRTC [6]

WebRTC uses JavaScript, APIs and Hypertext Markup Language (HTTP) to embed communications technologies within web browsers. It is designed to make audio, video and data communication between browsers user-friendly and easy to implement. WebRTC works with most major web browsers [6].

WebRTC APIs perform several key functions, including accessing and recording video-, audio- and text-based data from devices to initiating, monitoring and ending P2P connections between devices via browsers and facilitating bidirectional data transfer over multiple data channels.

In most cases, WebRTC connects users by transferring real-time audio, video and data from device to device using P2P communications. In situations where users are on different Internet Protocol (IP) networks that have Network Address Translation (NAT) firewalls that prevent RTC, WebRTC can be used in conjunction with Session Traversal Utilities for NAT (STUN) servers. This enables a given IP address to be translated into a public internet address so peer connections can be established. But there are also networks that are so restrictive that even a STUN server cannot be used to translate IP addresses. In these cases, WebRTC is used with a Traversal Using Relays around NAT (TURN) server, which relays traffic between users, enabling them to connect. The Interactive Connectivity Establishment protocol is used to find the best connection [6].
Before audio and video files are sent, they must be compressed due to their large size. Also, media that is received over a peer connection must be decompressed. WebRTC uses a codec process to do this [6].

1.3.2 WebRTC's functions in an emergency call system:

In an emergency call system, WebRTC plays the following functions:

- Real-Time Communication: WebRTC provides the necessary protocols and APIs to establish direct peer-to-peer communication between the client's web browser and the help center server. It enables seamless and instant transmission of audio and video streams, allowing clients to communicate their emergency details to help center operators in real-time.
- Secure and Encrypted Communication: WebRTC incorporates built-in security mechanisms to ensure the confidentiality and integrity of communication. It uses SRTP (Secure Real-time Transport Protocol) to secure WebRTC connections, protecting sensitive information exchanged during emergency calls from unauthorized access or interception [7].
- Peer-to-Peer Connection Establishment: WebRTC facilitates the establishment of a direct connection between the client's web browser and the help center server without the need for intermediate servers. This peer-to-peer connection minimizes latency and improves the quality of real-time communication by eliminating the reliance on additional network infrastructure.
- NAT Traversal: WebRTC includes techniques for traversing Network Address Translation (NAT) devices. It allows the client's device to establish a direct connection with the help center server, even if they are located behind firewalls or routers.
- Media Streaming and Codecs: WebRTC supports audio and video media streaming, providing the necessary APIs to capture, encode, and transmit media data in real-time. It incorporates various audio and video codecs to optimize bandwidth usage and ensure efficient transmission of audio and video streams during emergency calls [8].

By leveraging the capabilities of WebRTC, an emergency call system can provide reliable, secure, and real-time communication between clients and help center operators. It facilitates quick incident reporting, accurate assessment of emergency situations, and efficient coordination of emergency response efforts, ultimately improving the effectiveness and timeliness of emergency services.

1.4 4G LTE technology

4G (4th generation mobile communication technology), it is a wireless communication technology evolution following the third generation. Based on the wireless communication network, it provides higher transmission speed, better anti-jamming performance, greater compatibility and safer technical support, However LTE (Long Term Evolution) is a particular technology that is included in the 4G category. One of the most widely deployed 4G technologies, it provides minimal latency and high-speed data delivery. LTE transmits data using a packet-switching network architecture and all-IP (Internet Protocol) network infrastructure. Compared to older 3G technologies, it offers quicker download and upload speeds [9].

The integration of 4G LTE networks with the emergency call system enhances its capabilities by providing reliable and high-speed mobile connectivity, the following are some significant ways that 4G LTE helps in emergency communication:

- Mobile Device Connectivity: Clients can use their 4G-enabled mobile devices to access the client-side web app and make emergency calls. The 4G LTE network ensures a stable and fast connection [9], enabling clients to report incidents quickly and efficiently from any location with network coverage.
- Real-Time Communication: 4G LTE networks offer low-latency and high-bandwidth connections [9], which are crucial for real-time communication during emergency calls. WebRTC leverages the capabilities of 4G LTE networks to transmit audio and video streams seamlessly between clients and help center operators, ensuring clear and uninterrupted communication.
- Mobile Data Transmission: 4G LTE networks provide high-speed data transmission [9], allowing the client-side web app to transmit incident details, multimedia content, and other relevant information to the help center server efficiently. This ensures that all crucial data related to the incident is conveyed promptly and accurately.
- Compatibility and Coverage: 4G LTE networks are widely available [9], offering extensive coverage in urban, suburban, and rural areas. This enables the emergency call system to reach a larger user base and provide emergency services to a broader population. The system can be designed to be compatible with various 4G LTE network providers, ensuring compatibility and accessibility for users across different network carriers.

Chapter 2

State of the art

2.1 Introduction

To provide a comprehensive understanding of the field, a thorough review of the existing literature on emergency call systems has been conducted. This literature review focuses on the features, functionalities, architectures, and performance metrics of various emergency call systems. By synthesizing and analyzing this body of work, valuable insights and best practices have been identified.

2.2 Review of existing emergency call systems worldwide:

A comprehensive review of existing emergency call systems worldwide reveals a diverse range of approaches and technologies employed to facilitate emergency communication. Examples include systems that leverage internet-based voice and video calls, smartphone applications for emergency reporting, and integrated platforms that allow for multimedia data exchange. These systems often integrate advanced technologies, such as artificial intelligence and location-based services, to enhance incident detection, resource allocation, and response coordination.

2.2.1 Quick Accident Detection and Response System

the Quick Accident Response System (QARS) was proposed to provide a better and quicker emergency response to the nearby Emergency Response units ERU in case of an accident, it can analyze the accident details and send prompt alerts through a mobile application and offline messaging. It also sends the live location of the accident site using a GPS module. The two separate mobile applications within this system consists of a Driver mode for synchronizing his account to the QARS system, a Pedestrian mode for sending photos/videos of the accident site and an Emergency Services portal which receives all alerts, notifications and information of all accidents occurring. As a result, the Quick Accident Response System (QARS) will contribute to a decrease in the number of fatalities associated with traffic accidents, the workflow of the system is demonstrated in Figure 2 [1].

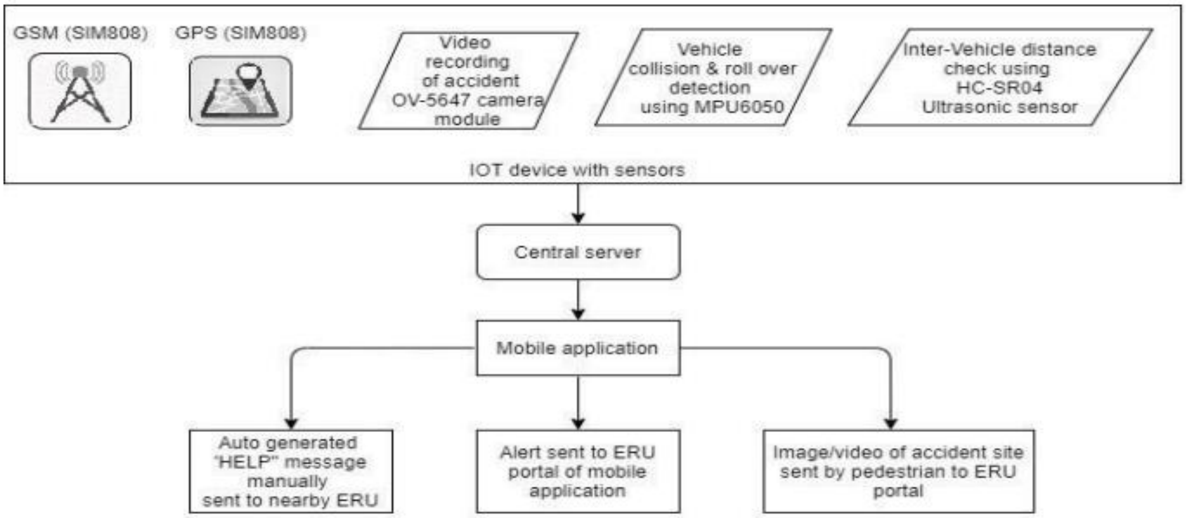

Figure 2: Workflow of the system [1]

2.2.2 Smart hospital emergency system

The proposed Smart Hospital Emergency System (SHES), aims to make use of the technology embedded within smart-phones (e.g., sensors and location services) to increase emergency service efficiency by providing faster response time. The main purpose of SHES is to automatically send patients information in lieu of using the conventional method in which the callers relays their information verbally on the phone. The overall aim of SHES will be to reduce the response time by which emergency calls are made and received by streamlining the number of tasks that are involved in the current process, the architecture of the system is shown in Figure 3.

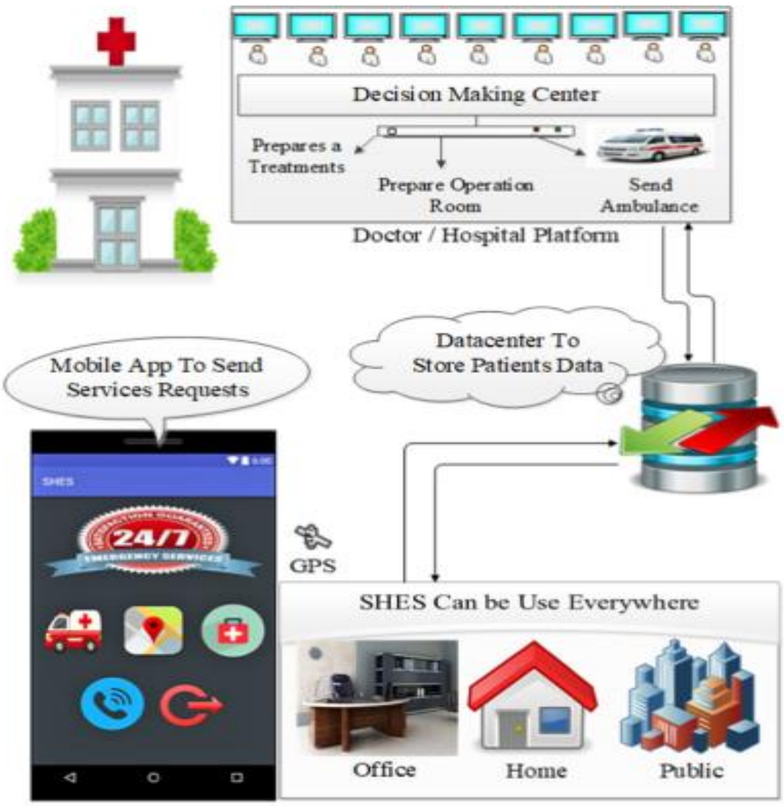

Figure 3: Architecture of the system [3]

2.2.3 eEmergency System to Support Emergency call Evaluation and Ambulance dispatch Procedures

This system was created to enhance communication between call centers, ambulances, and hospitals and emergency response. It improves the way emergency calls are handled and dispatch protocols. The system uses technology like tablet computers and telemedicine devices to record and send bio signals and incident data. A central Control Server and a Database Server for data storage are also included. Security safeguards guarantee confidentiality and adherence to healthcare legislation. The system has been implemented effectively at Cyprus' Ministry of Health's Ambulance Department, processing over 62,000 emergency calls and demonstrating its efficacy in enhancing emergency response and communication among stakeholders in emergency medical care the architecture of the system is shown in Figure 4 [2].

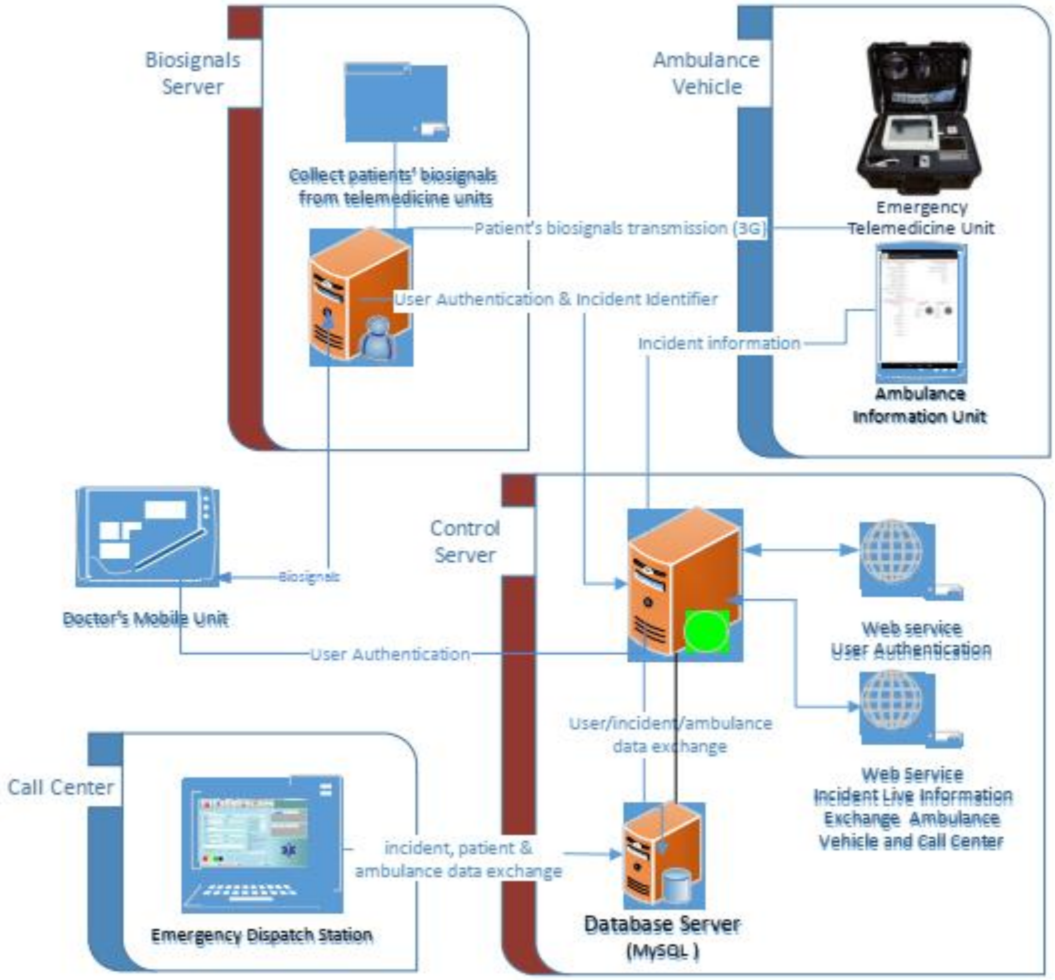

Figure 4: The system's architecture [2]

2.2.4 Analysis of emergency call tracking in a 4G/LTE mobile network [4]

Analyzing emergency call tracking in a 4G/LTE mobile network is a process of measuring and evaluating the performance and quality of the emergency communication system using the 4G/LTE network architecture. It involves the following steps:

- Triggering emergency calls using smart devices that send sensor data via Bluetooth to a smartphone.
- Routing emergency calls through an NG112 platform that supports additional data and video transmission using the IP-Multimedia Subsystem (IMS) network nodes, Figure 5 displays an IMS Emergency Call.
- Measuring the latency between sending the data and receiving the call using TTCN-3 tools and adapters.

- Defining and executing test cases using TTCN-3 language and syntax to test different scenarios and technologies involved in NG1121.
- Identifying and resolving issues or bottlenecks that may occur in the emergency communication system, such as network connectivity, data quality, interoperability, security.

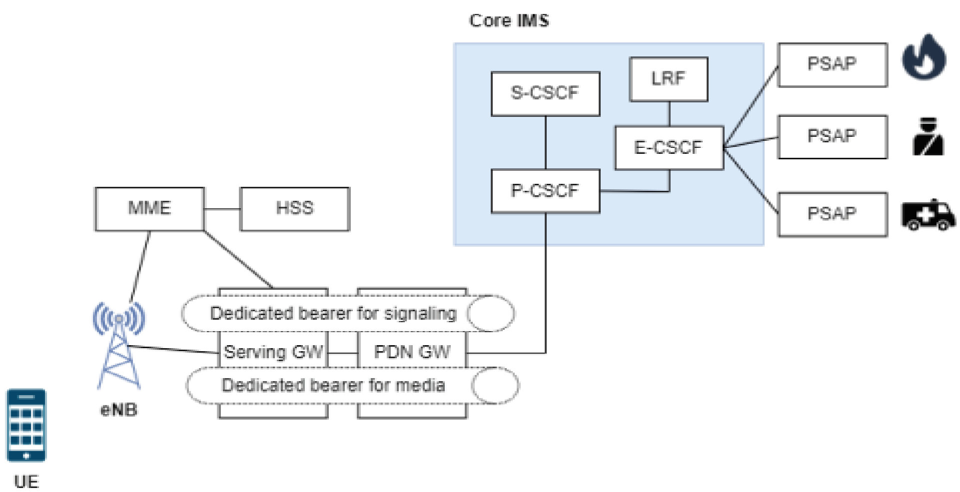

Figure 5: Emergency Call over IMS [4]

2.3 Analysis of Their Features, Strengths, and Limitations

By analyzing the features and functionalities of existing emergency call systems, we can identify their strengths and limitations. Some systems like smart hospital emergency system excel in providing real-time multimedia communication, enabling citizens to share crucial incident details through images, videos, and location data. Others like the eEmergency System to Support Emergency call Evaluation and Ambulance dispatch Procedures, focus on efficient resource allocation and dispatch, optimizing response times by considering factors like incident severity, proximity, and available resources. However, certain limitations persist, such as potential network congestion during emergencies, and challenges in ensuring data privacy and security.

2.3.1 Features and advantages

The systems previously discussed have a variety of possibilities to improve emergency communication. will highlight the essential benefits and features of each of them

2.3.1.a. Quick Accident Response System [1]:

- *Dedicated mobile applications for drivers*: pedestrians, and emergency services are included in the system. This application provides a convenient and effective way to report incidents and organize emergency response activities by acting as a centralized platform for communication.
- *Rapid accident detection*: is made possible by the system's use of IoT technologies and multipurpose sensors like accelerometers and ultrasonic sensors. This enables quick action and prompt medical attention, potentially lowering mortality.
- *Automatic alarm system*: As soon as an accident is discovered, the system instantly transmits alerts via a specialized mobile application to the closest Emergency Response Units (ERUs). This guarantees prompt notification of emergency agencies, speeding up response times.
- *Accurate accident data*: The system records video of the accident scene and records the precise location using a GPS-GSM module. The ERUs get this information, allowing them to assess the gravity of the crisis and offer the proper support.
- *Participation of pedestrians:* The system has a pedestrian gateway in the mobile application that enables users to give emergency services real-time image and video feeds. This improves situational awareness and aids emergency personnel in more efficient scene assessment.
- *Offline accident alerts*: The system offers an offline feature in cases where the victim is unable to call for help. It enables users to send text messages to the closest ERUs or pre-saved emergency contact numbers with accident alerts and the particular accident location.

2.3.1.b. Smart hospital emergency system [3]:

- This system can use the technology embedded within smart-phones (e.g., sensors and location services) to increase emergency service efficiency by providing faster response time.
- It can process health data held within a personal smartphone, and internal tracked data (GPU, Accelerometer, Gyroscope etc.), efficiently, and securely, through automatic communications with emergency services, ultimately reducing communication bottlenecks.
- It can provide live video-streaming through real-time video communication protocols to improve initial communications between emergency services and patients.
- It can reduce the strain on emergency call systems, which estimated by a 9 million calls (including both landline and mobile) made in 2014 alone in the UK.
- It can improve diagnosis and treatment by providing accurate and timely information about the patient’s condition and medical history.
- It can enable seamless functioning of the hospital facility by integrating with other systems and devices, such as smart ambulances, smart beds, smart monitors.

2.3.1.c. eEmergency System to Support Emergency call Evaluation and Ambulance dispatch Procedures [2]*:*

- *Improved emergency call processing*: The technology supports and enhances the emergency call handling process, ensuring faster and more efficient response times.
- *Improved fleet management for ambulances*: The system supports fleet management for ambulances, enabling effective resource allocation and coordination.
- *Support for emergency call evaluation and triage*: The system aids dispatchers in determining the urgency of incidents and allocating the proper resources by reviewing and triaging emergency calls.
- *Coordination:* Better coordination and decision-making are made possible because to the system, which enables real-time data interchange between the contact center, ambulance trucks, and hospital emergency rooms.
- *Managing a huge number of calls effectively*: Over 62,000 calls were successfully handled and recorded over the system's more than a year of use, demonstrating its reliability and effectiveness.

2.3.1.d. Analysis of emergency call tracking in a 4G/LTE mobile network [4]:

- It can offer high-speed data transfer services that can improve the efficiency and quality of the emergency communication system.
- It can provide additional data and information to the emergency services that can improve the diagnosis and treatment of the patient.
- It can use a standardized and flexible language (TTCN-3) to define and execute test cases for different scenarios and technologies involved in NG112.
- It can identify and resolve issues or bottlenecks that may occur in the emergency communication system.
- It can verify and validate the functionality and performance of the emergency communication system according to the requirements and specifications.

2.3.2 Limitations and disadvantages

The systems that were previously discussed have several disadvantages; we w point out some of them.

2.3.2.a. Quick Accident Response System:

- *Cost:* There may be a large financial commitment necessary to implement and maintain the suggested solution. The price of setting up and maintaining the required infrastructure, which includes sensors, IoT devices, and mobile application development, can be high. This might make it less feasible, especially in places with poor infrastructure or resources.
- *Privacy concerns:* The system collects and transmits private information, including location data and video recordings. Regarding the retention, accessibility, and potential abuse of this

data, privacy issues could surface. To solve these issues and safeguard user information, there has to be clear privacy rules and security procedures in place.

- *False alarms*: could occur since the system depends on so many different sensors. Potholes, other road irregularities, or abrupt braking could set off the accident detection system, causing pointless notifications and possibly disrupting emergency services. To reduce false alarms and ensure accurate accident detection, steps must be taken.
- *User acceptance*: users' acceptance and adoption are essential for the suggested system to be successful. It is necessary to inform and train drivers, pedestrians, and emergency response workers about the advantages of the system. Widespread system acceptance may be hampered by people's resistance to change, a lack of awareness, or reluctance to utilize the mobile application.
- *Technical difficulties*: The suggested system's implementation may encounter technical difficulties. It might be challenging to integrate numerous sensors, guarantee their accuracy and dependability, and maintain the system's overall operation. The system may require frequent calibration, upgrades, and maintenance to remain functional.

2.3.2.b. Smart hospital emergency system:

- *Network connectivity*: SHES relies on the availability and reliability of network connections to transmit data and video between the patient and the emergency services. If the network is slow, congested, or disrupted, it may affect the quality and timeliness of the communication.
- *Data quality*: SHES depends on the accuracy and completeness of the data collected from the patient's smartphone and other devices. If the data is corrupted, missing, or outdated, it may lead to incorrect or incomplete diagnosis and treatment.
- *Interoperability*: SHES needs to be able to exchange data and information with other systems and devices in the emergency care ecosystem, such as smart ambulances, smart hospitals, smart beds, etc. If there are incompatible standards, formats, or protocols, it may hinder the integration and coordination of care.
- *Infrastructure*: SHES requires a robust and scalable infrastructure to support the data processing, storage, and transmission needs of the system. If there are insufficient or outdated hardware, software, or network components, it may affect the performance and functionality of the system.
- *Equipment*: SHES involves the use of various devices and sensors to collect and monitor the patient's health data and video. If there are faulty or incompatible devices or sensors, it may affect the quality and reliability of the data and video.
- *Training:* SHES entails a new way of delivering emergency care that involves the use of technology and automation. If there are inadequate or ineffective training programs for the emergency staff and the patients, it may affect their ability and willingness to use the system.
- *Data ownership*: SHES involves the collection and sharing of the patient's data and video with various parties in the emergency care ecosystem. If there are unclear or conflicting rules or

agreements on who owns, controls, or accesses the data and video, it may create disputes or confusion.

- *Consent*: SHES requires the patient's consent to collect and share their data and video with the emergency services. If there are ambiguous or inconsistent methods or policies for obtaining and managing the patient's consent, it may violate their rights or preferences.
- *Privacy*: SHES involves the exposure of the patient's personal and sensitive information to various parties in the emergency care ecosystem. If there are insufficient or inappropriate measures or safeguards to protect the patient's privacy, it may cause harm or distress to them.
- *Liability*: SHES entails a complex and dynamic process of emergency care that involves multiple parties and technologies. If there are errors, failures, or harms that occur during or as a result of using the system, it may raise questions or issues about who is responsible or accountable for them.
- *Dependency*: SHES increases the reliance on technology and automation in emergency care. If there are situations where the technology or automation fails or malfunctions, it may affect the availability or quality of care.
- *Human* interaction: SHES reduces the human interaction between the patient and the emergency staff. If there are cases where the patient needs emotional support or reassurance from a human voice or face, it may affect their satisfaction or comfort.

2.3.2.c. eEmergency system to support emergency call evaluation and ambulance dispatch procedures:

- *Logistical issues*: Despite technical developments, logistical issues might still occur, especially when planning the allocation of supplies during emergencies.
- *Dependence on certain protocols:* The system is dependent on particular emergency dispatch procedures, which may limit its ability to modify to various emergency situations or procedures used in other regions or nations.
- *Security considerations*: Although the system has security safeguards like authentication and encryption, there may still be security risks or vulnerabilities that need to be fixed to protect the privacy and confidentiality of sensitive data.
- *Limitations on user access*: The system establishes various access levels for various user roles, which may prevent some users from accessing specific features or data, thereby affecting workflow or decision-making.
- *Device constraints*: The capabilities or restrictions of the used devices, such as the Ambulance Information Unit or Doctor Mobile Unit, may have an impact on the system's functionality and effectiveness.

2.3.2.d. Analysis of emergency call tracking in a 4G/LTE mobile network [4]:

- *Infrastructure Investment*: It may require significant investment in infrastructure, equipment, and training.
- *Legal and Ethical Issues*: It may encounter legal and ethical issues such as data ownership, consent, privacy, and liability.
- *Compatibility Issues*: It may not be compatible with some existing systems or devices that are not 4G/LTE-enabled.
- *Increased Dependency*: It may increase the dependency on technology and reduce the human interaction in emergency care.

2.4 Identification of Common Challenges and Areas for Improvement

Despite improvements in emergency call systems, several challenges persist in their implementation

- *Offline functionality*: One of the main issues is that emergency call systems often neglect offline functionality, however it is critical to take into account situations when internet connectivity may be limited or degraded, including isolated regions or natural disasters. The system's availability and accessibility can be greatly improved by implementing offline capabilities that enable users to save and sync data locally, ensuring that crucial information is accessible even under difficult circumstances.
- *Privacy concerns*: The need to manage privacy and confidentiality concerns related to sensitive data, such as video and location data. It is essential to protect this data in order to preserve public confidence and safeguard individual privacy.
- *Open access*: Many current emergency call systems do not make their functions and data available to the public, it is critical to encourage transparency in these systems, enabling contributions and innovation from outside developers and researchers. Open access can promote cooperation, encourage the creation of independent software and services, and result in ongoing system capability enhancements.

2.5 Conclusion

In this chapter we have given an overview of the current emergency call systems. These systems are essential for guaranteeing public safety and quick emergency response. We have covered some key features and benefits of these systems. This chapter attempts to improve the understanding of the facilities and tools that are available to manage emergency situations effectively and efficiently.

Chapter 3
Design of the proposed system

3.1 Introduction:

Based on the findings from the state-of-the-art review, there is a clear need for a new emergency call system adapted to the Algerian context. The current reliance on green numbers and voice calls alone poses limitations in terms of communication efficiency, data exchange, and scalability. By leveraging the benefits of 4G LTE networks, WebRTC technologies and mobile phones, a new emergency call system can overcome these limitations and provide a more robust and effective solution.

4G LTE networks have revolutionized communication and made mobile phones an integral part of our daily lives. With its considerable throughput, low latency, and widespread adoption, 4G LTE networks enables fast and reliable data transmission, while WebRTC facilitates real-time audio, video, and data communication. This integration of 4G LTE and WebRTC technologies allows for efficient communication during emergency situations.

The web-based Emergency Call System proposed in this thesis empowers citizens by offering a user-friendly interface to report incidents in real-time. With the online feature, users can easily access the system and report incidents using the web application from anywhere with an internet connection. This ensures immediate incident reporting and enables emergency responders to initiate a swift response.

However, we understand that network connectivity may not always be available in emergency situations. To address this limitation, our system includes an offline feature that allows users to report incidents even without an internet connection. The offline functionality utilizes local storage on the user's device to save incident reports temporarily. Once an internet connection becomes available, the system automatically synchronizes the locally stored reports with the central database. This ensures that no incident goes unreported, even in areas with limited or no network coverage.

3.2 System architecture

This Emergency Call System is a web-based application designed to provide citizens with a reliable and efficient means of reporting incidents in real time.
The system also incorporates offline capabilities through local storage and synchronization mechanisms, allowing users to report incidents and make emergency calls even in the absence of connectivity.
Figure 6 shows the architecture of our system, including three layers: caller side, help center side, and the emergency responder’s side.

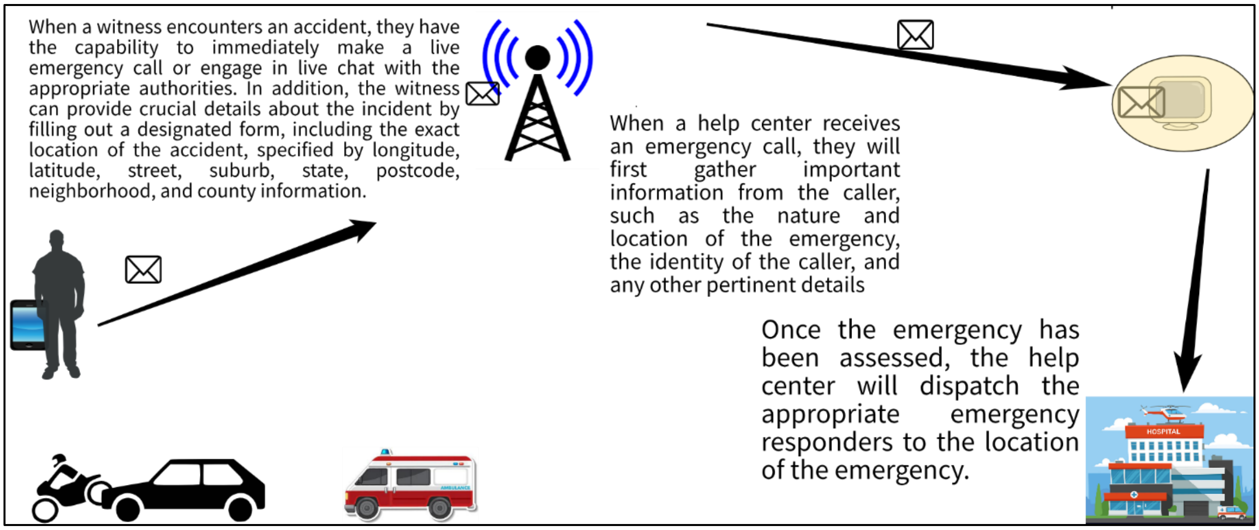

Figure 6: Architecture of the proposed system

The caller side, the help center side, and the emergency responders side make up the three primary parts of the planned emergency call system. To ensure effective incident reporting, review, and response, each component is essential.

1. Caller Side:
    - *User Interface (UI)*: A web application accessible to the caller for reporting incidents and making emergency calls. It provides a user-friendly interface for capturing incident details and initiating calls, the interface is designed to be accessible and easy to navigate.
    - *Incident Reporting*: The web application allows the caller to input relevant information about the incident, such as the type of emergency, location, and additional details. This data is sent to the help center side for evaluation and coordination.
    - *Emergency Call*: The web application includes a dedicated button to initiate emergency calls. When pressed, it establishes a connection to the help center side, allowing the caller to communicate directly with the operators.

2. Help Center Side:
    - *Operator Interface*: The help center staff utilizes a web application interface to receive and manage emergency calls from callers. The app displays relevant caller information, incident reports, and real-time communication options (audio, video, chat) with the caller.
    - *Incident Evaluation*: The operators assess the severity and urgency of each reported incident based on the information provided by the caller. They analyze the incident details, location, and caller's situation to determine the appropriate response.

- *Emergency Responder Coordination*: Upon evaluating the incident, the help center operator uses the system to coordinate and dispatch the available emergency responders to the scene of the incident.
- *Caller Assistance*: The operator maintains communication with the caller, providing guidance, reassurance, and instructions for performing first aid or safety measures until the emergency responders arrive.

3. Emergency Responders Side:
    - *Responder Interface*: The emergency responders, equipped with mobile devices connected to the application, receive real-time updates and information from the help center side.
    - *Information Relay*: The help center operator relays the gathered information about the nature of the emergency, location, and any other relevant details to the emergency responders through the web application.
    - *Real-time Updates*: The responders receive the information in real-time, enabling them to assess the situation, prepare necessary resources, and plan their response accordingly. They may communicate with the operator or each other for further coordination or clarification.
    - *on Route and Scene Management*: The responders utilize the application to navigate to the incident scene, using GPS directions and real-time updates. Once at the scene, they take charge of the situation, providing appropriate assistance, performing first aid, and ensuring the safety of the caller and others involved.

The three layers are connected through APIs, ensuring the flow of data and interactions between the caller, help center, and emergency responders.

Figure 7 shows the case where a caller is not in a 4G LTE coverage area, the emergency call system is designed to handle such situations and provide alternative means of communication.

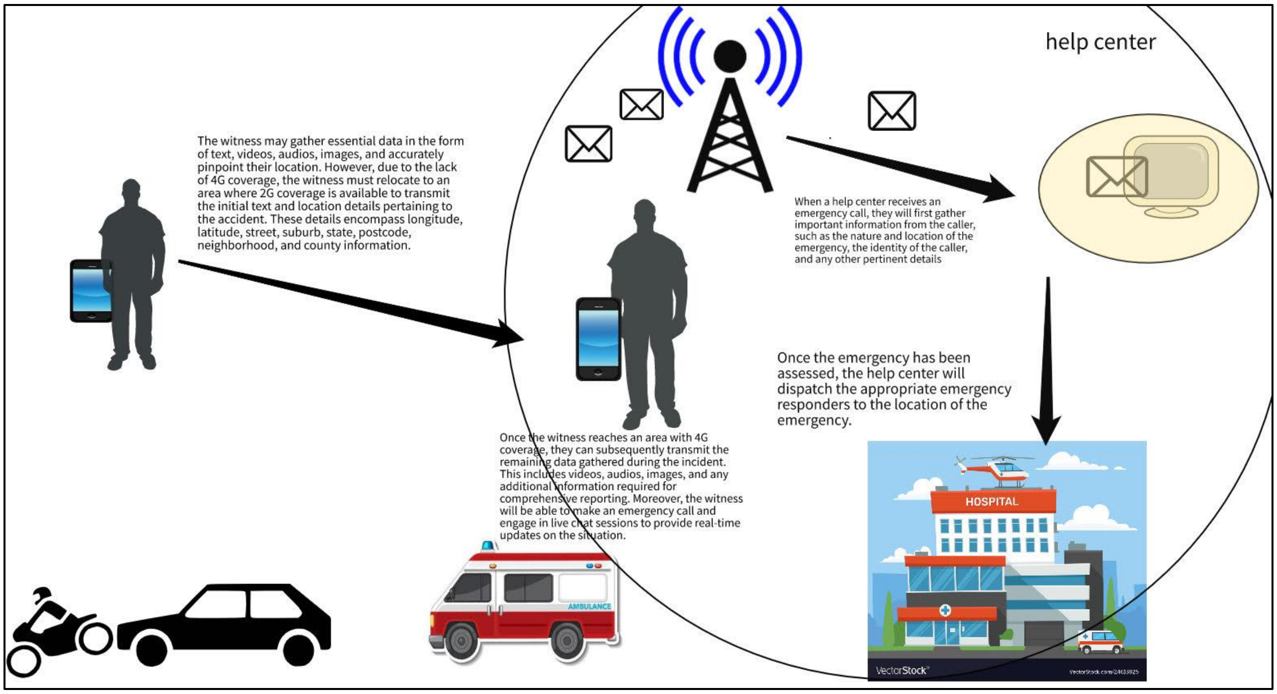

Figure 7: Architecture of the system in an area with no 4G LTE coverage

In offline scenarios, the emergency call system is designed to handle situations where internet connectivity is limited or temporarily unavailable. The system employs a strategy that prioritizes local storage and synchronization to ensure that essential data, such as incident reports, videos or audio recording about the incident, and location of the incident, which are saved locally until a connection becomes available. Once a connection is established, the system automatically synchronizes the stored data with the help center. This local storage allows the system to continue functioning even when there is no immediate connection to the help center.

3.3 Emergency call Workflow description

The following is a description of the sequence of events that occurs during the emergency call scenario represented in Figure 8:

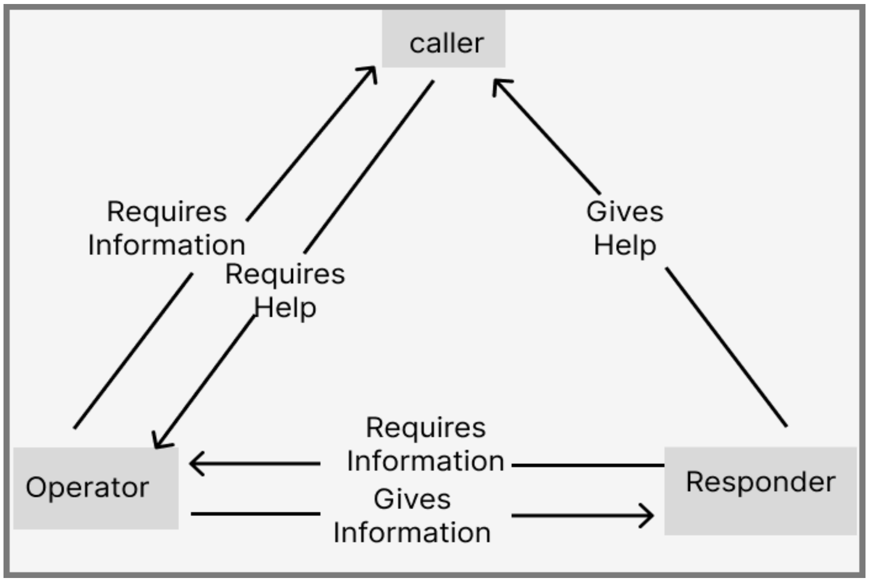

Figure 8: Workflow of the system

1. Caller initiates the emergency call:
   - The caller accesses the app, through a mobile device.
   - They open the app and locate the "Emergency Call" button.
   - The caller clicks on the "Emergency Call" button to initiate the call
2. Call routing to the operator via the web app:
   - The call is automatically routed to the emergency call center through the 4G LTE network.
   - The operator receives the incoming call.
   - The operator's interface displays relevant details about the call, such as the caller's location (obtained through GPS), caller ID, and any additional data available like video, audio recordings or images.
3. Caller's location verification and data retrieval:
   - The web application retrieves the caller's location from their mobile phone, leveraging the GPS functionality.
   - The web application may also gather additional data associated with the caller's mobile phone, such as medical history or emergency contacts, if the caller has granted permission to access that information
4. Caller provides emergency details:
   - The operator establishes a live audio or video connection with the caller through the web app using WebRTC technologies.
   - The caller explains the nature of the emergency, providing crucial information about the situation.
   - the caller can share images, videos, or other relevant media to provide a better understanding of the emergency.
5. Operator assesses the situation and coordinates with the responder:

- The operator actively listens to the caller and uses the information provided to assess the severity and urgency of the situation.
- Simultaneously, the operator can relay the gathered information to the appropriate emergency responder through the web app.
- The responder receives real-time updates, including the nature of the emergency, the caller's location, and any other relevant details to aid in their response.

6. Responder on route and real-time updates:
   - The responder, equipped with a mobile device connected to the web application, receives the emergency details and acknowledges the assignment.
   - The application provides turn-by-turn directions using GPS navigation to guide the responder to the location of the emergency.
   - The web app continues to provide real-time updates to the responder as additional information becomes available or if the situation changes.
7. Operator maintains communication with the caller:
   - While the responder is on route, the operator remains in contact with the caller and guides them on how to provide first aid to the injured until the paramedics arrive at the scene. They provide step-by-step instructions, including checking vital signs, ensuring a clear airway, and stabilizing any external injuries.
8. If the caller were in an area with no network coverage, the offline mode of the Emergency Call System would come into play. In such a situation, the caller would still be able to report the incident using the web application. The system would save the incident report locally on their device, utilizing the offline storage capability. Once the caller regains network coverage or moves to an area with connectivity, the system automatically synchronizes the locally stored report with the central database, ensuring that the incident is reported and addressed without delay.
9. Responder arrives and handles the emergency:
   - The responder, utilizing their training and expertise, reaches the location of the emergency and takes charge of the situation.
   - They use their mobile device and the web app to update their status, communicate with the operator, or request additional resources if needed.

By integrating features such as incident reporting, real-time communication, data exchange, and offline accessibility, the Emergency Call System enhances the effectiveness and efficiency of emergency communication, ultimately contributing to the safety and well-being of the community.

3.4 Use cases:

Figure 9 shows the use cases in the emergency call system that describe the specific interactions and actions that various actors, including the caller, help center operators, and emergency responders, perform within the system. These use cases represent the core functionalities that the system supports, ensuring efficient incident reporting, coordination, and assistance during emergency situations.

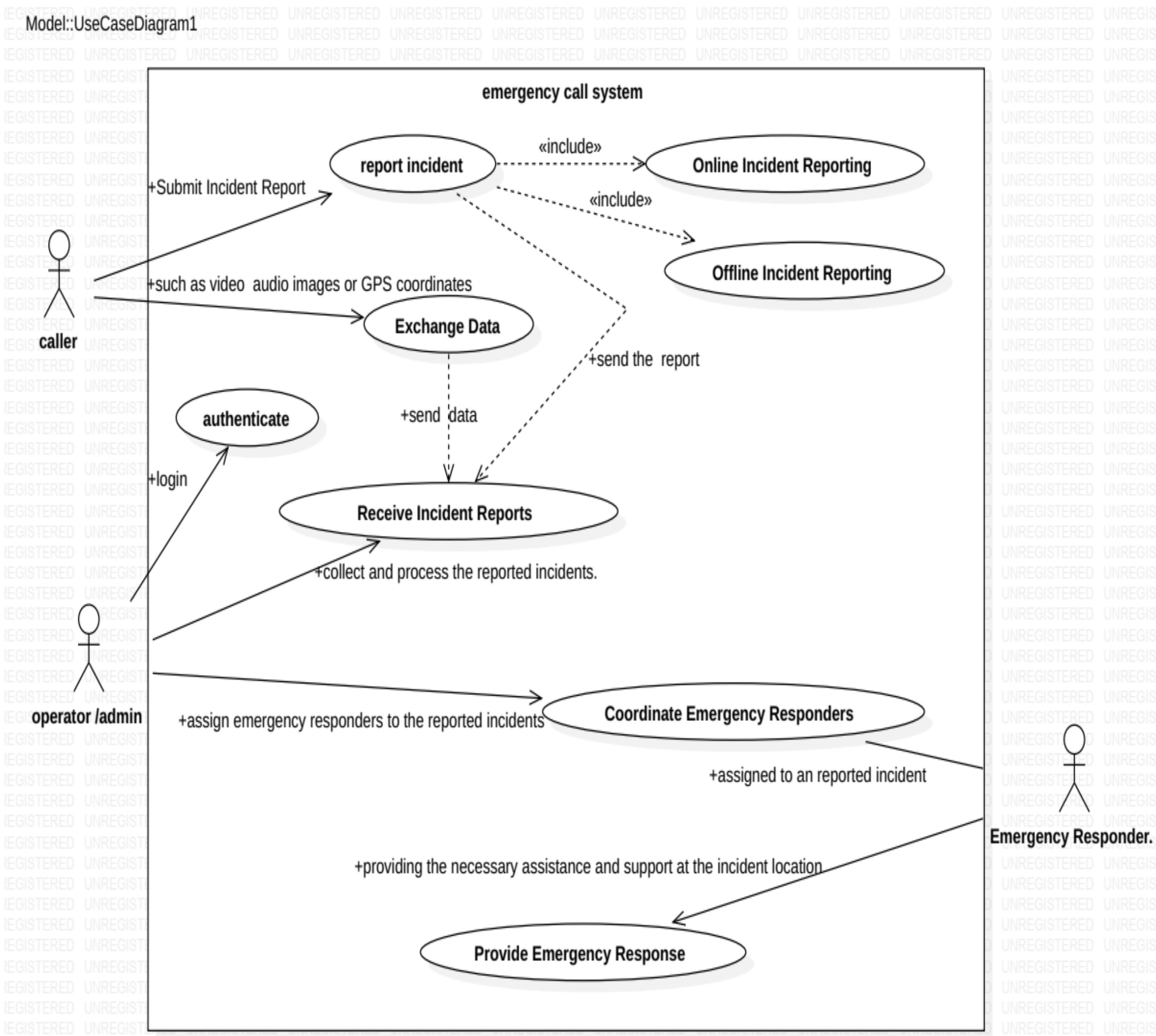

Figure 9: Use case diagram

The actors involved in this case are the caller, the operator (admin) and emergency responders and the main use cases are report incident, authenticate, receive incident report, coordinate emergency responders and provide emergency response explained as followed:

1. Caller (User):
   - *Report Incident*: The caller reports an incident, providing all the necessary details.
   - *Exchange Data*: The caller exchanges data, such as video, audio, images, and GPS coordinates related to the incident.

2. Operator (admin):
    - *Authenticate*: The operator authenticates themselves to gain access to the system.
    - *Receive Incident Reports*: The operator receives incident reports from the callers.
    - *Coordinate Emergency Responders*: The operator coordinates with emergency responders based on the received incident reports.

3. Emergency Responders:
    - *Coordinate Emergency Responders*: The emergency responders receive an alert and instructions from the operator regarding the reported incidents.
    - *Provide Assistance*: The emergency responders provide the necessary assistance at the incident location.

The "Incident Reporting" use case includes the "Online Incident Reporting" and "Offline Incident Reporting" use cases, as indicated by the <<include>> relationship. This accounts for both online and offline scenarios where the system allows incident reporting regardless of internet connectivity. The specific details and interactions within each use case can be further elaborated in separate sequence diagrams.

3.5 Sequence diagrams

Present sequence diagrams that depict the step-by-step interactions between system components and actors in specific scenarios.
These diagrams focus on illustrating the dynamic behavior of the system during incident reporting and real-time communication.
Show the sequence of messages exchanged between system components, highlighting the order of events and the collaboration between actors to accomplish specific tasks.

3.5.1 Sequence diagram of “Authenticate “:

The “Authenticate **“**sequence diagram illustrates the interaction between the operator and emergency call system during authentication process. The operator initiates a request to login, which is received by the emergency call system. The emergency call system then verifying the operator’s credentials in the system’s database. After verifying, the operator can get access to the system confirming the successful logging as shown in Figure 10.

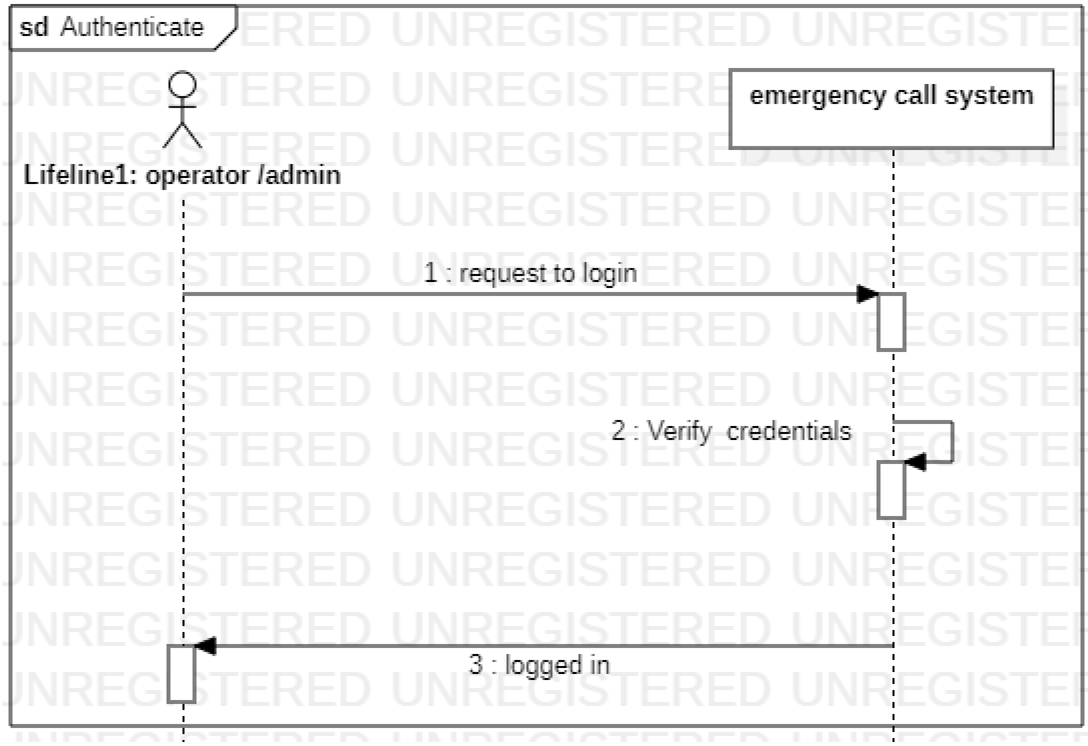

Figure 10: Sequence diagram of authenticate

3.5.2. Sequence diagram of "report incident":

The "report incident" sequence diagram illustrates the interaction between the caller, emergency call system, and "receive incident report "use case during incident reporting in online mode. The user initiates a request to report an incident, which is received by the emergency call system. The emergency call system then saves the incident report in the system's database, then the system sends the data to the operator, then the system sends a notification to the user confirming the successful submission.

In case the incident reporting was in offline mode, when the caller saves a report without an internet connection, the report is stored locally on the device using local storage. Once an internet connection is available, the report is synchronized with the system's database. The user then receives a confirmation indicating the successful synchronization.

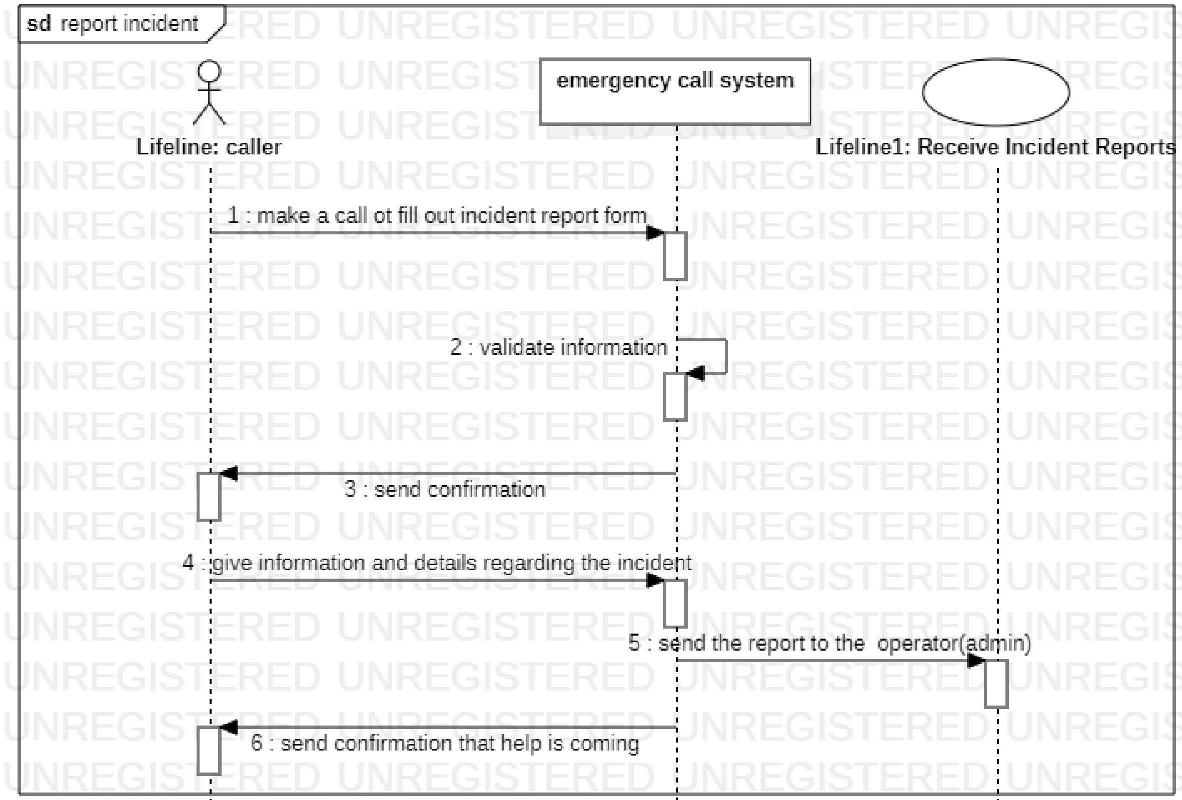

Figure 11: Report incident

3.5.3 Sequence diagram of “exchange data “:

This sequence diagram illustrates the interaction between the caller, emergency call system, and “receive incident report “use case during data exchange. The caller can send data such as video, audio, images, or GPS coordinates. The emergency call system than validate the information and saves the data in the system’s database. After saving the data, the system sends the data to the operator.

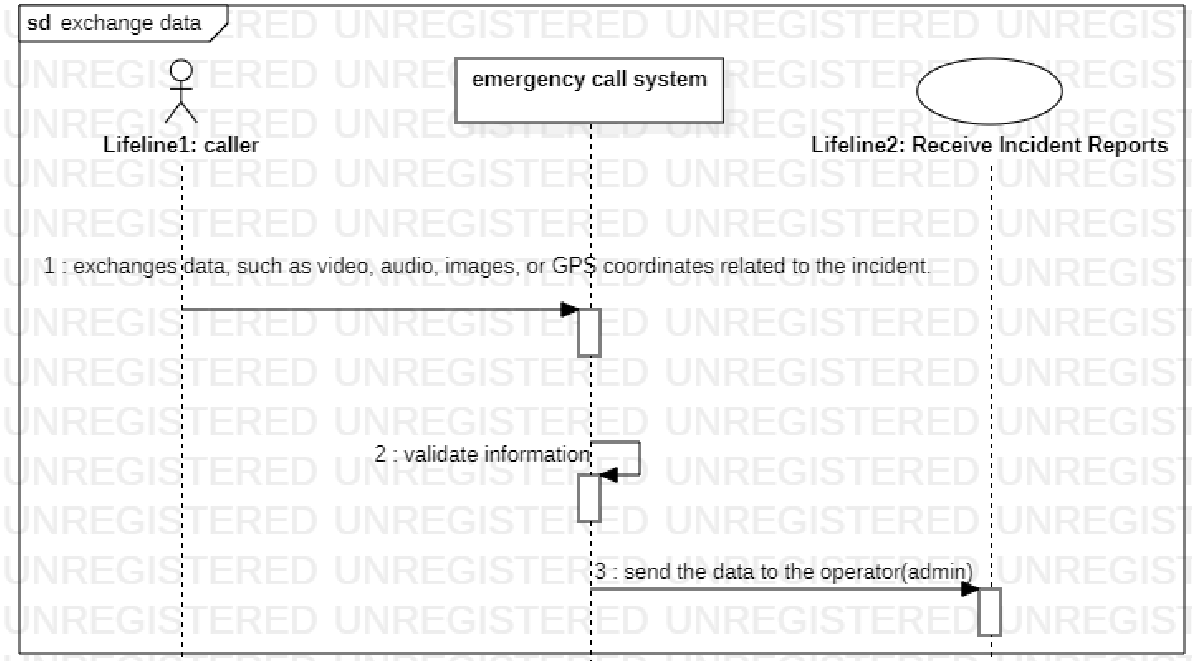

Figure 12: Sequence diagram of “Exchange data”

3.5.4 Sequence diagram of “receive incident reports “:

The “receive incident reports “sequence diagram illustrates the interaction between the operator/admin, and emergency call system during receive incident reports. The operator requests the incident reports, the emergency call system than process the request. After processing the request, the system then display the incident reports to the operator.

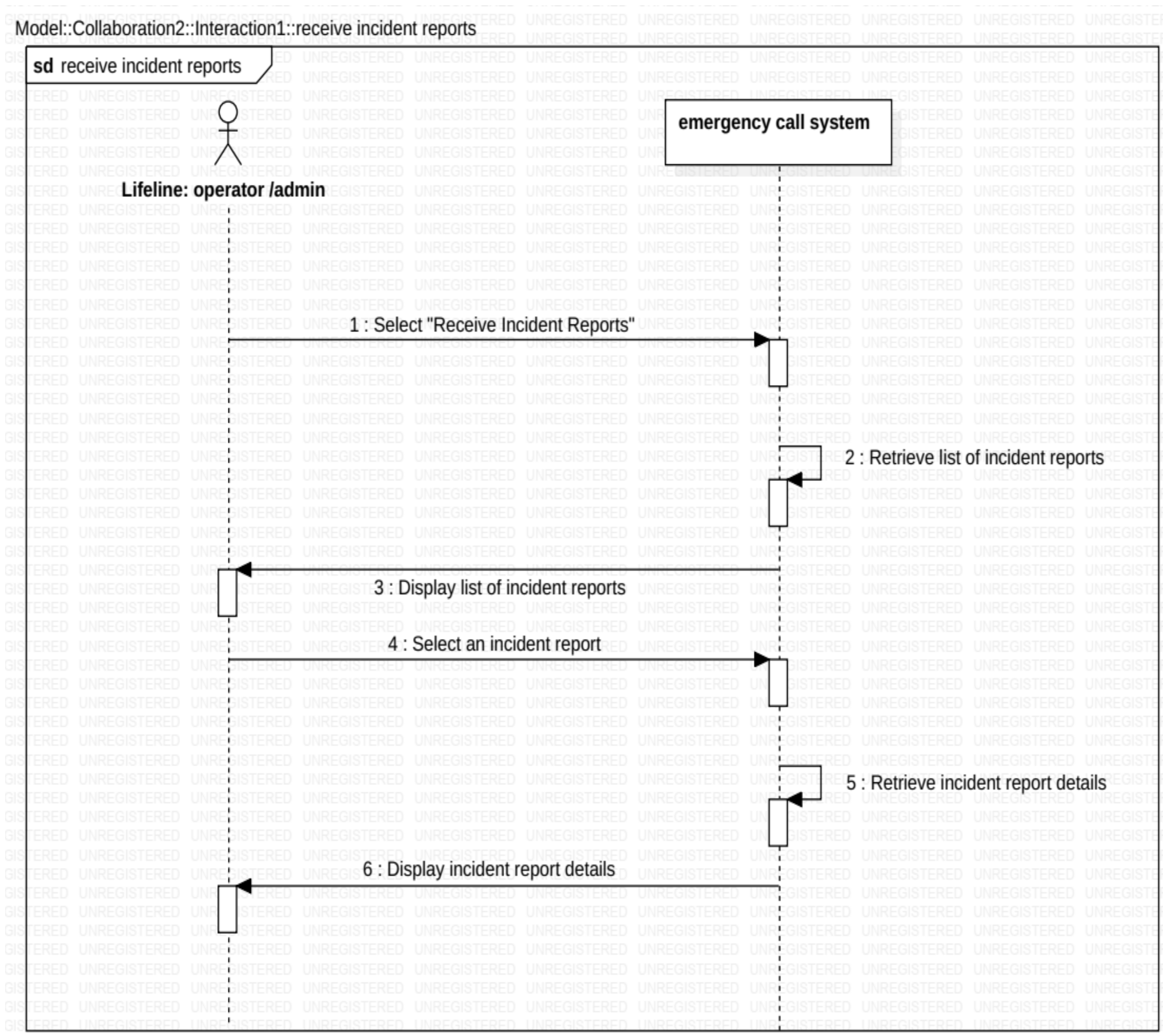

Figure 13: Sequence diagram of “receive incident reports “

3.5.5 Sequence diagram of “Coordinate emergency responders “:

This sequence diagram illustrates the interaction between the operator/admin, and emergency call system during “coordinate emergency responders”. The operator requests the list of available response teams, the emergency call system than process the request. After processing the request, the system then display the list of available response teams to the operator then, the operator select a response team and assign them to an incident report.

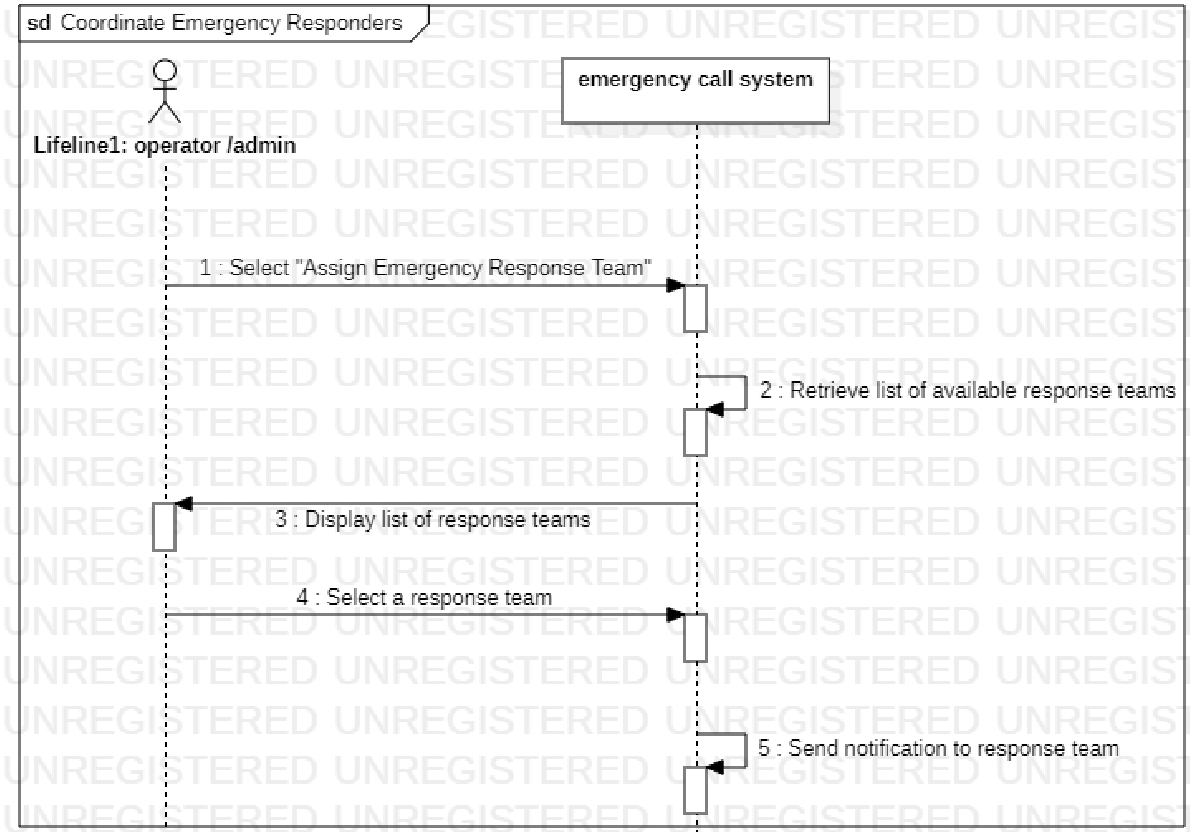

Figure 14: Sequence diagram of “Coordinate emergency responders “

3.5.6 Sequence diagram of “provide emergency response “:

The “provide emergency response “sequence diagram illustrates the interaction between the emergency responders, and emergency call system during provide emergency responders. The emergency responder gets assigned to an incident report, the system then retrieve incident report details and displays them to the emergency responder, then the emergency responders get to the scene of the incident and provide the necessary assistance and support at the incident location.

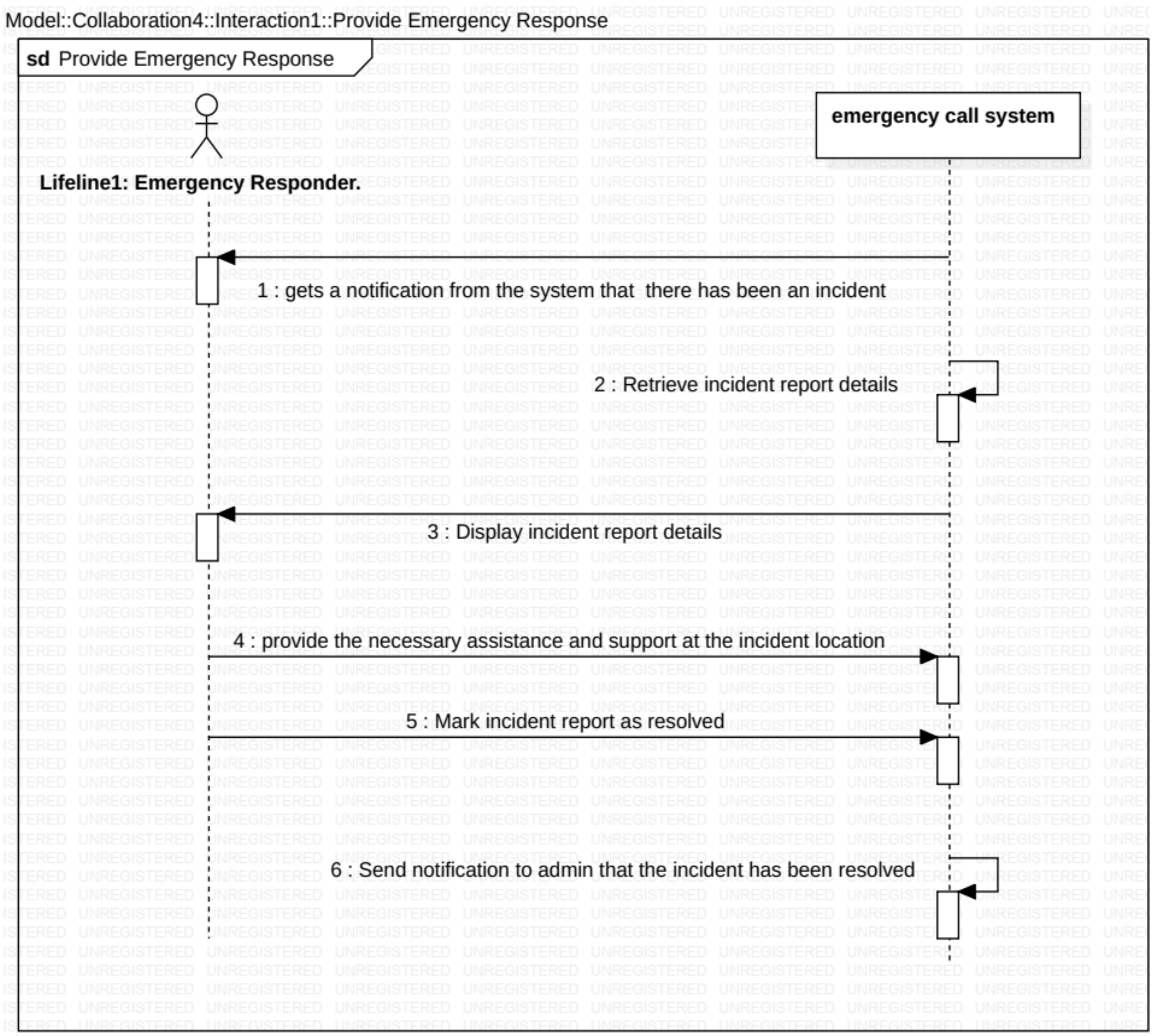

Figure 15: Sequence diagram of “Provide emergency response “

3.5.7 Sequence diagram of “Data transfer channel “:

Figure 16 represents the WebRTC data transfer channel establishment using a WebSocket server

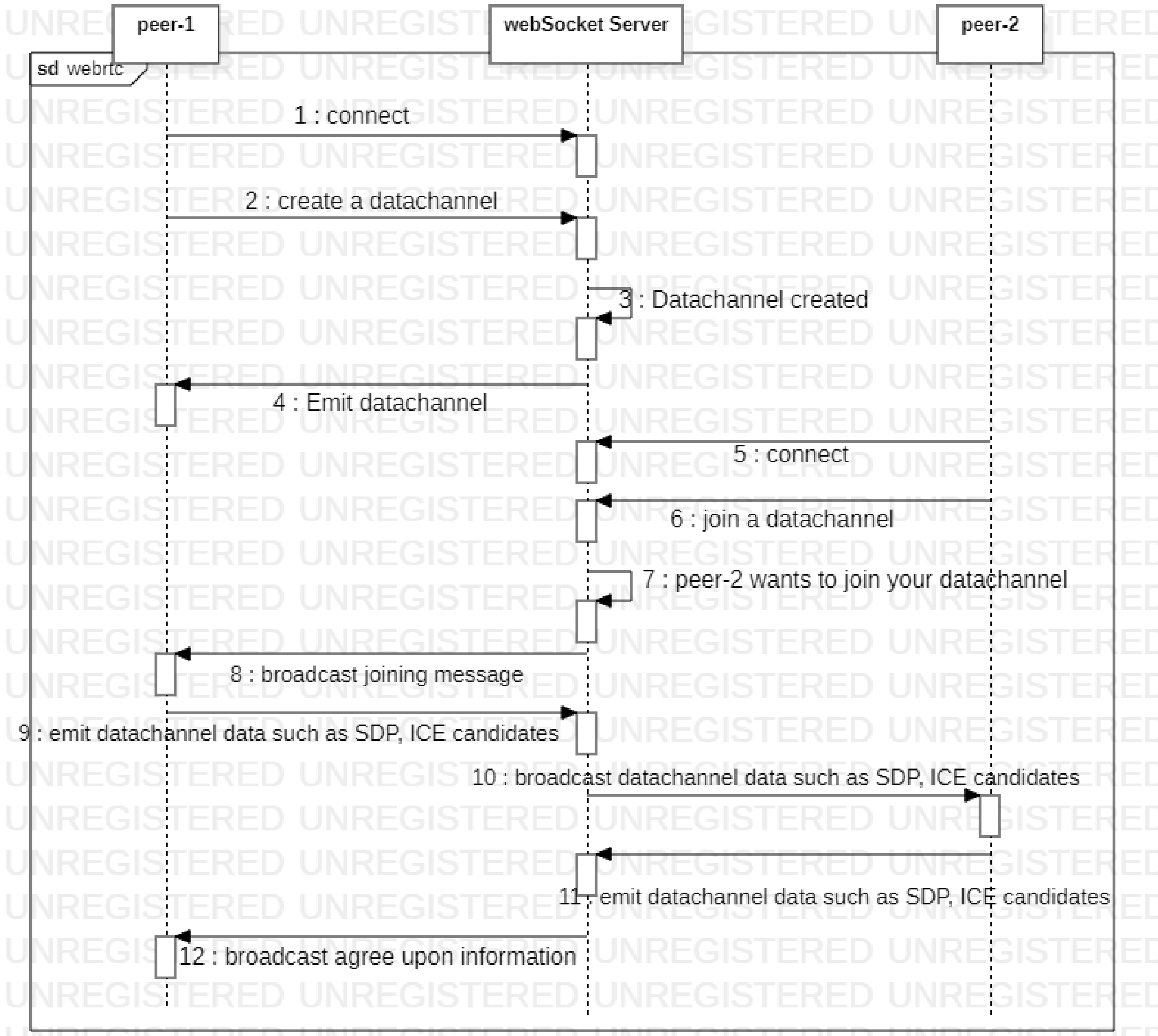

Figure 16: Bidirectional data transfer channel establishment

3.6 Class diagram

The class diagram represents the key entities and their relationships in the emergency call system.

The system is designed to facilitate efficient incident reporting, emergency call handling, incident evaluation, responder coordination, caller assistance, and effective communication between all parties involved.

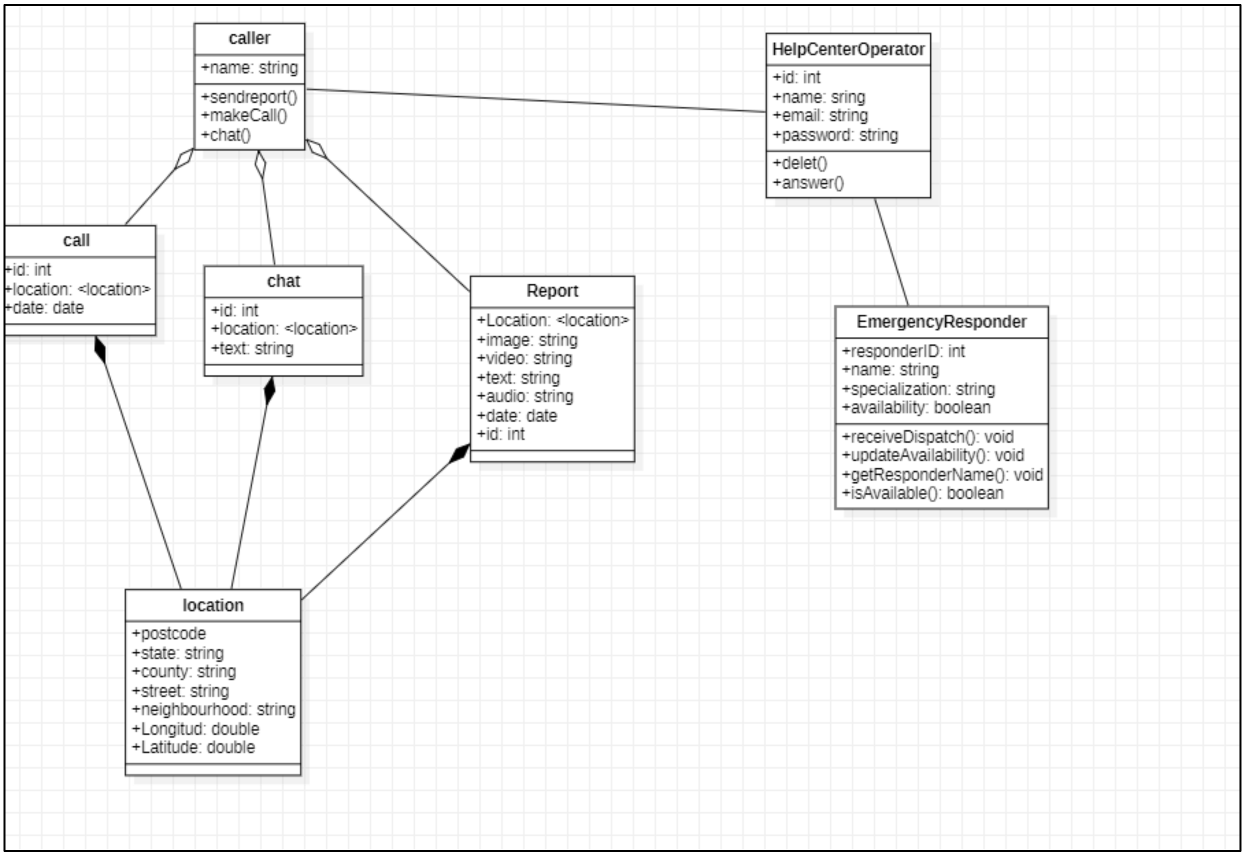

Figure 17: Class diagram

In this class diagram, we have classes representing different entities in the system: Caller, HelpCenterOperator, report, chat, call, location and EmergencyResponder.

- *The Caller class*: represents the caller who reports the incident and makes emergency calls. It has attributes for the caller's name. It provides methods for making emergency calls, reporting incidents, and the ability to chat.
- *The report class:* represents an incident reported by the caller. It stores information such as the incident ID, location, text, video, audio, date, and images.
- *The HelpCenterOperator class:* represents the operator working at the help center. It has attributes for the operator's ID, name, and password. The class includes methods for receiving calls and delete.
- *The EmergencyResponder class:* represents the emergency responder who is dispatched to the incident scene. It has attributes for the responder's ID, name, specialization, and availability. The class provides methods for receiving dispatch notifications, updating availability status, and retrieving responder information.
- *The call class*: represent the call being made by caller to the operator. It has attributes for the call's id, location, and date.

- *The chat class*: represent the chat that happens between caller and the operator. It has attributes for the call's id, location, and text.
- *The location class*: represent the location of the incident. It has attributes like postcode, state, county, street, neighborhood, longitude, latitude.

3.7 Modular architecture

The architecture of the system follows a modular approach, ensuring flexibility, scalability, and maintainability. Each module is responsible for specific functionalities, allowing for easy integration and future enhancements as shown in Figure 18.

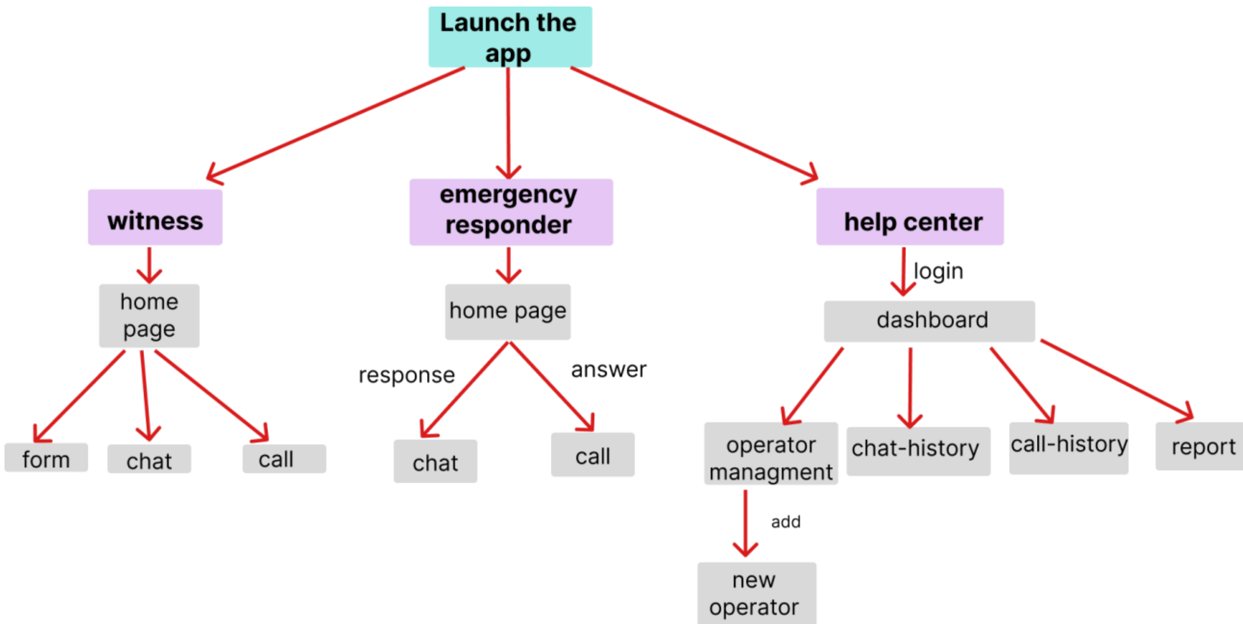

Figure 18: Modular architecture

The proposed modular architecture of the emergency call system consists of three main modules: Witness, Emergency Responder, and Help Center. Each module is responsible for specific functionalities and interacts with other modules to facilitate efficient incident reporting and response.

3.7.1 Witness Module:

- *Home Page Module*: This module comprises three key components: The Form Module, the Chat Module, and the Call Module
    - *Form Module*: Allows witnesses to fill out an incident report form, providing essential details.
    - *Chat Module*: Enables witnesses to engage in real-time text-based communication with the Help Center.

- *Call Module*: Facilitates real-time audio and video communication between witnesses and the operator in the Help Center.

3.7.2. Emergency Responder Module:

- *Home Page Module*: This module comprises three key components: The Chat Module, and the Call Module
    - *Chat Module*: Enables real-time text-based communication between emergency responders and the Help Center.
    - *Call Module*: Facilitates real-time audio and video communication between emergency responders and the operator in the Help Center.

3.7.3 Help Center Module:

- *Operator Dashboard Module*: after login it Provides operators with a centralized interface for managing incoming incidents and coordinating response efforts. This module comprises three key components: Operator Management Module, Chat History Module, and Call History Module
    - *Operator Management Module*: Allows operators to add new operators to the system and manage their access privileges.
    - *Chat History Module*: Stores and displays the history of text-based communication between witnesses, and operators.
    - *Call History Module*: Records and presents the history of audio and video calls made within the system.
    - *Report Module*: Enables operators to generate incident reports based on the information received from witnesses and emergency responders.

In the Witness module, witnesses have access to the Form, Chat, and Call sub-modules on the Home Page, allowing them to report incidents, communicate with the Help Center via text or audio/video, and share their location information. The Emergency Responder module provides emergency responders with the Chat and Call sub-modules on their Home Page, enabling them to communicate directly with the Help Center during incident response.

The Help Center module consists of the Operator Dashboard, which serves as the central hub for operators to manage incoming incidents. The Operator Management module allows operators to add new operators. The Chat History and Call History modules store the communication history for reference and analysis. The Report module allows operators to generate comprehensive incident reports based on the collected information.

By employing this modular architecture, the emergency call system ensures an efficient workflow, enabling effective incident reporting and response. Location sharing across all Witness modules feature provides geographical information to help emergency responders in their actions.

3.8 Conclusion

In overall, the proposed Emergency Call System, designed to meet the needs of the Algerian civil protection, offers a comprehensive solution for reporting incidents and initiating emergency response. Leveraging the benefits of 4G LTE technology and WebRTC technologies, the system provides online and offline functionalities, allowing citizens to report emergencies in real-time and receive timely guidance from emergency responders, by using features such as incident reporting, real-time communication, data exchange, and offline accessibility.

Chapter 4

Implementation

4.1 Introduction

During the implementation phase of the proposed emergency call system, several tools and programming languages were utilized to bring the system to life. The following sections provide an overview of the key aspects of the implementation, including front-end and back-end development, integration of the Mapbox API for location tracking, and offline functionality implementation.

4.2 High-level overview of configuring the web app

The development of the emergency call system employed a range of tools and programming languages to ensure a robust and efficient implementation. Here are the steps we followed:

- Step 1: we Installed Node.js and npm (Node Package Manager)
- Step 2: we Opened a terminal or command prompt and created a new directory for our project using the following command:
  ```
  mkdir emergency-call
  cd emergency-call
  ```
- Step 3: Initialize a new npm project using the following command:
  ```
  yarn init –y
  ```
- Step 4: Install Next.js as a dependency in our project:
  ```
  yarn add next react react-dom sqlite3 webrtc mapbox socket.io
  ```
- Step 5: Create the basic structure of our Next.js app by adding a few essential files. Create a new folder called pages and inside it, create an index.js file:
  ```
  mkdir pages
  cd pages
  mkdir components
  touch index.js
  ```
- *Step 6:* we Configured the *Next.js* framework by Creating a next*.config.js* file in the project root directory and add the necessary configuration settings for *Next.js*.
- Step 7: Develop React components by Creating React components in the component's directory using *JavaScript and JSX* syntax to define the UI elements and their functionality.
- Step 8: Implementing server-side logic with Node.js by Creating a server file *server.js*, we Used *Node.js* to implement server-side logic for handling user requests, managing data, and facilitating real-time communication.
- Step 9: Setting up the SQLite3 database by Create a database file*, database.db*, and we defined the necessary tables and schemas using SQLite3.

- Step 10: Integrating WebRTC for real-time communication Implement WebRTC technology to enable real-time video and audio communication between the caller and operators. We used getUserMedia, RTCPeerConnection WebRTC APIs to establish peer-to-peer connections, handle media streaming, and enable real-time data transmission.
- Step 11: Configure WebSocket for data transmission Implement WebSocket protocol to establish a reliable and efficient data transmission channel between the caller and the help center. We used socket.io a WebSocket libraries to handle bidirectional communication and enable real-time updates.
- Step 12: Integrate Mapbox API for location tracking Register for a Mapbox account and obtain API credentials. We used Mapbox API and libraries to integrate location tracking and mapping functionality into our web app. Implement features such as displaying incident locations.
- Step 13: Implementing HTTPS and SSL for secure communication by Generating SSL certificates for our domain and Configuring our server to use HTTPS protocol for secure communication.
- Step 14: Implement offline functionality to allow users to use the web app even when an internet connection is not available. Using caching strategies to store and synchronize data locally on the user's device.
- Step 16: run the web app using this command:

```
yarn dev
```

These steps provide a high-level overview of configuring our application using the mentioned technologies

4.3 Front-end development (HTML, SCSS, React.js, Next.js, JavaScript):

Front-end development focused on designing and developing the user interface (UI) of the emergency call system, it involved creating various features such as incident reporting, emergency call initiation, user interaction and creating a user-friendly interface using HTML, SCSS, React.js, Next.js, and JavaScript. Here's an overview of the steps:

- *HTML (Hypertext Markup Language*): is the standard markup language for documents designed to be displayed in a web browser, html forms were utilized to capture incident details, including the type of emergency, location, and additional information.
- *SCSS (Sassy Cascading Style Sheets*): is one of two syntaxes available for the popular CSS preprocessor Sass (Syntactically Awesome Stylesheets). It can be used to style the visual elements of a webpage, including buttons, sliders, images, color schemes, fonts, themes, and layouts. As a true superset of CSS, all valid CSS is also valid SCSS [10], it was used to style the forms and create a good interface.
- *React.js*: is a free and open-source front-end JavaScript library for building user interfaces based on components. It is maintained by Meta and a community of individual developers and companies. React.js is built using JSX – A combination of JavaScript and XML.

Elements are created using JSX, then use JavaScript to render them on the web site. While React has a steep learning curve for a junior developer, it's quickly shaping into one of the most popular and in-demand JavaScript libraries [12], it was utilized for building reusable UI components and managing component state.

- *Next.js:* is a free and open-source project maintained by Vercel and a community of individual developers and companies, it is a React framework that gives a building blocks to create web applications. It is used by some of the world's largest companies, such as Netflix, Hulu, Twitch, GitHub, and more. The fact React lets putting components together, but it doesn't prescribe how to do routing and data fetching, so we used Next.js to provided server-side rendering, data fetching, integrations, routing, and other helpful features for building web applications [13].
- *JavaScript*: is a programming language that is one of the core technologies of the World Wide Web, while it is most well-known as the scripting language for Web pages, many non-browser environments also use it, such as Node.js, Apache CouchDB and Adobe Acrobat [14]. It was used to add interactivity and handle client-side functionalities, and provides the necessary interactivity and functionality, such as user interactions.

A lot of features and functionalities were implemented using these front-end technologies. For instance, the incident reporting form allows users to input relevant information about the emergency, such as images, location, and video or audio recordings. The emergency call feature provides a dedicated button to initiate emergency calls, establishing a connection to the help center side and enabling direct communication with the operators.

4.4 Back-end Development:

The back-end development involved the implementation of the server-side components responsible for processing and managing data.

- *Node.js*: is an open-source, cross-platform JavaScript runtime environment that executes JavaScript code outside of a web browser. As an asynchronous event-driven JavaScript runtime, Node.js is designed to build scalable network applications [15]. Node.js was used to build the back-end logic and handle tasks such as user authentication, incident reporting, enabling real-time communication between the callers, help center operators, and emergency responders.
- *A database management system (DBMS):* is a software tool that enables users to manage a database easily. It allows users to access and interact with the underlying data in the database. These actions can range from simply querying data to defining database schemas that fundamentally affect the database structure, DBMS allow users to interact with a database securely and concurrently *without* interfering with each user *and* while maintaining data integrity. Overall, employing a DBMS helps in maintaining data

consistency, reliability, and security, making it an essential component in managing and storing system data effectively [16].

We used Axios library to send data between the server and client.

```
let res = await axios.post('api/reports', dataToSave)
if (res.data.success) {
  router.push("/")
  toast.success(res.data.message)
}
```

Figure 19: Code for sending data

We used socket.emit method to transmit personalized events between the server and client. Emit events on one side of the socket.io API and register listeners on the other side because it was influenced by the node.js EventEmitter.

```
const joinRoom = () => {
    let room_name = roomName || Math.random().toString(36).slice(2)
    socket.emit("new witness joined the chat", { roomName: room_name, selectedAdmin })
    if (selectedAdmin)
        return router.push(`/chat/${room_name}`)
    setError("Select agent first")
}
```

Figure 20: Transmit personalized events between the server and client

4.5 Integration of Mapbox API for Location Tracking

In order to enhance the location tracking functionality of our emergency call system, we have integrated the Mapbox API. Mapbox provides powerful routing engines, accurate, traffic-powered travel times, and intuitive turn-by-turn directions to help building engaging navigation experiences [11]. The integration of Mapbox API allows us to precisely pinpoint the caller's location and provide accurate information to emergency responders.

The user interface is also enhanced to display the incident location on a map, providing a visual representation for better situational awareness.

```
const getInfo = (e) => {
  axios.get(`token`)
    .then(res => {
      try {
        commonProps.setValue("location", {
          longitude: e.lngLat[1],
          latitude: e.lngLat[0],
          street: res.data.results[0].components.road,
          suburb: res.data.results[0].components.suburb,
          state: res.data.results[0].components.state,
          postcode: res.data.results[0].components.postcode,
          neighbourhood: res.data.results[0].components.neighbourhood,
          county: res.data.results[0].components.county,
        })
        setInfo({
          street: res.data.results[0].components.road,
          suburb: res.data.results[0].components.suburb,
          state: res.data.results[0].components.state,
          postcode: res.data.results[0].components.postcode,
          neighbourhood: res.data.results[0].components.neighbourhood,
          county: res.data.results[0].components.county,
        })
      } catch (error) {
        console.log(error);
      }
```

Figure 21: Example code for getting incident locations

4.6 Database Integration using SQLite3:

To support data persistence and efficient data management, the emergency call system utilizes a relational database management system (DBMS).
We chose SQLite3 as the database solution due to its lightweight nature and ease of integration with Node.js. SQLite3 is a self-contained, serverless, and zero-configuration database engine that operates on a single file. SQLite3 provides full ACID (Atomicity, Consistency, Isolation, Durability) compliance and supports SQL queries for data manipulation and retrieval. SQLite3 is a C-language library that can be embedded into other applications or run as a standalone program [17]. The integration of SQLite3 in the emergency call system offers the following advantages:

- *Data Storage*: SQLite3 allows for the storage of structured data, such as incident details, chat history, emergency response records. The data is organized into tables with defined schemas, providing a structured format for efficient storage and retrieval.
- *Data Persistence*: By persisting data in a database, the emergency call system ensures that important information is retained even if the server or application restarts. This enables the system to resume normal operation without data loss or inconsistency.

- *Data Querying*: With the support of SQL, SQLite3 enables powerful and flexible querying capabilities. Complex queries can be executed to retrieve specific data based on various conditions.

The integration of SQLite3 with the emergency call system involves the following steps:

- *Database Creation*: The system creates a new SQLite3 database file or connects to an existing one. This can be done during the system initialization process or on-demand when needed.

```
const sqlite3 = require('sqlite3').verbose();
const fs = require("fs");

export default function start(req, res) {

  // Check if folder exists
  if (!fs.existsSync('./emergency.sqlite')) {
    // Create folder if it does not exist
    fs.writeFile('./emergency.sqlite', "", (err) => {
      if (err) {
        // Handle error
        console.error(err);
      } else {
        // File created successfully
        console.log("File created successfully");
      }
    });
  }

  const db = new sqlite3.Database('./emergency.sqlite', (error) => {
    if (error) {
      return console.error(error.message);
    }
  });
```

Figure 22: Code showcasing the integration of SQLite3 in Node.js

- *Table Schema Design*: The necessary tables and their corresponding schemas are defined based on the data requirements of the system. This includes specifying column names, data types, constraints, and relationships between tables.
- *Data Manipulation*: The system utilizes SQL statements, such as INSERT, DELETE, and SELECT, to manipulate data in the SQLite3 database.

```
form.on('end', () => {
  // prepare the INSERT statement
  const stmt = db.prepare("INSERT INTO Reports (text, location, image, video, audio,created_at) VALUES (?, ?, ?, ?, ?,?)");
  // run the INSERT statement
  stmt.run(data.text, JSON.stringify(data.location), JSON.stringify(data.images), data.video, data.audio, moment().format("DD/MM/YYYY"));
  // finalize the statement and close the database connection
  stmt.finalize();
  db.close();
  // send a response to the client
  res.status(200).json({
    message: 'Your Information sent successfully',
    success: true
  });
})
```

Figure 23: Example code for inserting incident report

- *Error Handling*: The system incorporates appropriate error handling mechanisms when interacting with the SQLite3 database. This ensures that any errors or exceptions during database operations are properly captured and handled, preventing system crashes.

By integrating SQLite3 into the emergency call system, data can be efficiently stored, retrieved, and manipulated, providing a reliable and structured approach to managing critical information.

4.7 Offline Functionality Implementation:

The emergency call system was designed to function even in scenarios where internet connectivity is limited or temporarily unavailable. Offline functionality was implemented to ensure uninterrupted operation and enable users to report incidents in such situations.

- *Local Data Storage*: The system allows for the storage of incident reports and other critical information (video or audio recordings, images, location) locally on the user's device. This allows callers to capture incident details even without an internet connection.
- *Synchronization*: Once an internet connection became available, the system can synchronize_the locally stored data with the server. This ensures that incident reports information would be transmitted to the help center for evaluation and coordination.

```
    default:

      localStorage.setItem("Data_to_be_sent", JSON.stringify(dataToSave));
      //api.post("/url/", data)

      break;
  }
```

Figure 24: Code for local data storage

By implementing offline functionality, the emergency call system enhances its reliability and availability, allowing users to access critical features even in situations with limited or no internet connectivity.

4.8 WebRTC Data Channels:

In the implementation phase of our emergency call system, we focus on incorporating WebRTC's Channels (video, audio and data), As shown in Figure 25, they enable the transmission of video, audio, and other data in real-time, ensuring seamless collaboration and information exchange.

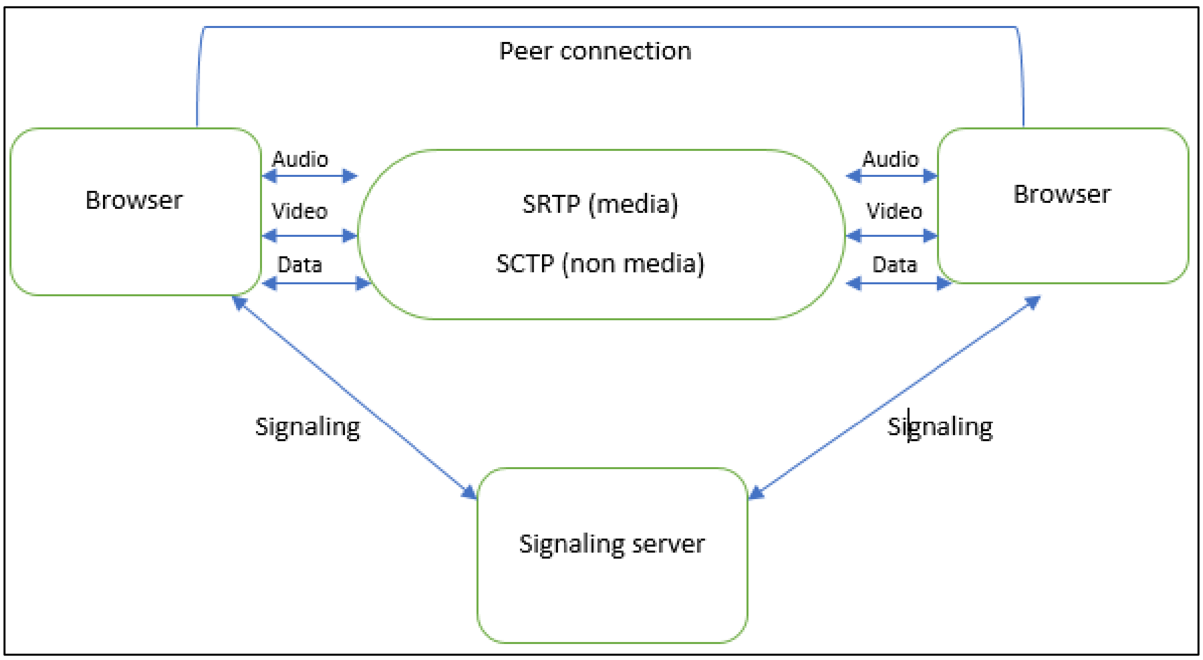

Figure 25: WebRTC data channels

- *Video and audio Online Channel:* Are responsible for real-time video and audio communication. It utilizes WebRTC's Secure Real-time Transport Protocol (SRTP) to encrypt and transmit video and audio streams securely between the participants. Through the implementation of SRTP, the system ensures the confidentiality and integrity of video data during transmission between the caller and operator [19].
- *Data Transmission Channel:* The Data Transmission Channel facilitates the exchange of additional data, such as text messages, file transfers. It employs the Stream Control Transmission Protocol (SCTP) over DTLS for secure and reliable data transmission. SCTP ensures the ordered and error-free delivery of data packets [19].

By implementing WebRTC's Data Channels and leveraging, our system ensures secure and reliable transmission of video, audio, and data. This enhances the overall communication experience and contributes to the effectiveness of our emergency call system.

4.9 Security and Authentication

Ensuring the security and privacy of sensitive information is important in our emergency call system. To achieve this, we have implemented HTTPS (Hypertext Transfer Protocol Secure) and SSL (Secure Sockets Layer) protocols.

HTTPS is a secure version of the HTTP protocol that encrypts the data transmitted between the user's browser and the server. This encryption provides an additional layer of protection against unauthorized access and eavesdropping. By implementing HTTPS, we guarantee that all communication between the caller's device and our system's server is encrypted, safeguarding sensitive information such as personal details and incident reports.

SSL certificate is utilized to establish a secure connection between the user's browser and the server. These certificates ensure the authenticity and integrity of the transmitted data. This prevents potential attacks from malicious entities attempting to impersonate our system.

These security measures reduce the risk of data breaches and unauthorized access, instilling trust in users and ensuring the confidentiality of sensitive information.

4.10 Testing and Quality Assurance:

Thorough testing and quality assurance practices were implemented throughout the development process of the emergency call system. End-to-end testing, testing was conducted to verify the functionality, performance, and reliability of the system. Quality assurance techniques, including code reviews and bug tracking, were employed to identify and resolve any issues or inconsistencies in the system's implementation.

4.11 Conclusion

In summary, the implementation phase of the emergency call system involved the use of various tools and programming languages. Front-end development encompassed HTML, SCSS, ReactJS, NextJS, and JavaScript to create an intuitive user interface. Back-end development relied on Node.js to handle server-side logic and facilitate real-time communication. Integration of the Mapbox API provided location tracking and mapping functionality. Offline functionality was implemented to enable data storage and synchronization when internet connectivity is limited.

Chapter 5

Result

5.1Video communication:

Video communication is a key feature of our proposed system. Callers can initiate real-time video communication requests with operators using the web application, as shown in Figure 26.

The system is designed to promptly display incoming video communication requests on the operator's dashboard.

The requests are listed along with the caller's name, and include an invitation acceptance button for the operator to attend the call. Upon accepting the invitation, a new session is created, allowing both the caller and the operator to engage in a conversation.

The WebRTC APIs facilitate the real-time transmission of voice and video, enabling seamless media sharing between the participants. The system adapts to 4G LTE networks, by automatically adjusting the resolution and bandwidth constraints.

The utilization of WebRTC in our system enhances the caller-operator communication by providing reliable and real-time multimedia capabilities, this eliminates the need for additional plug-ins and ensures efficient multimedia sharing.

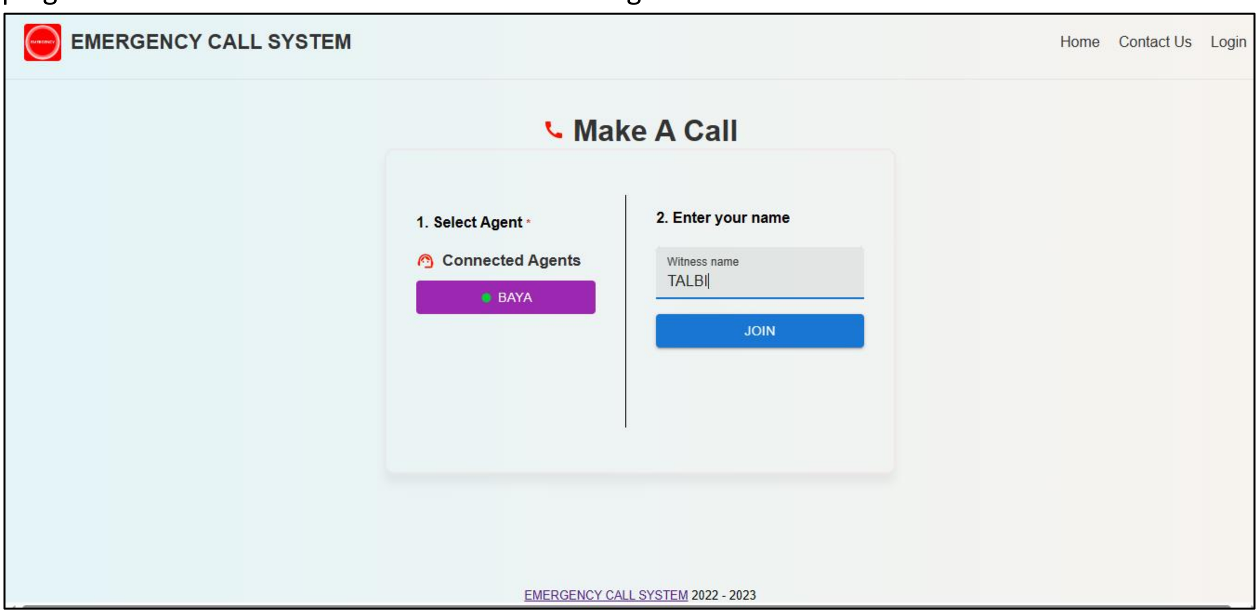

Figure 26: Make a call

5.2 Data exchange

Text communication, as well as the ability to send incident reports, photos and audio, are important features incorporated into our proposed system as shown in Figure 27 and Figure 28, users can engage in real-time text conversations with operators, allowing for quick and efficient communication of relevant information.

Once the caller establishes a connection with the help center, they are presented with a text input field on the user interface. They can type their message, providing essential details about the emergency or any additional information they wish to communicate. Upon hitting the send button, the text message is instantly transmitted to the help center's operators.

If the caller needs to share visual information, such images of injuries, they can easily do so. Within the user interface, there is an option to attach a photo. By clicking on this option, the caller can browse their device's gallery or take a photo using the device's camera. Once the desired photo is selected, the caller can send it along with their text message. The photo is securely transmitted to the help center, where it can be accessed and reviewed by the operators.

To improve communication and provide additional context, the system allows callers to send audio recordings. The caller can initiate the audio recording feature by clicking on the designated button within the user interface. They are then able to record their voice, describing the situation of the emergency or providing any necessary information.

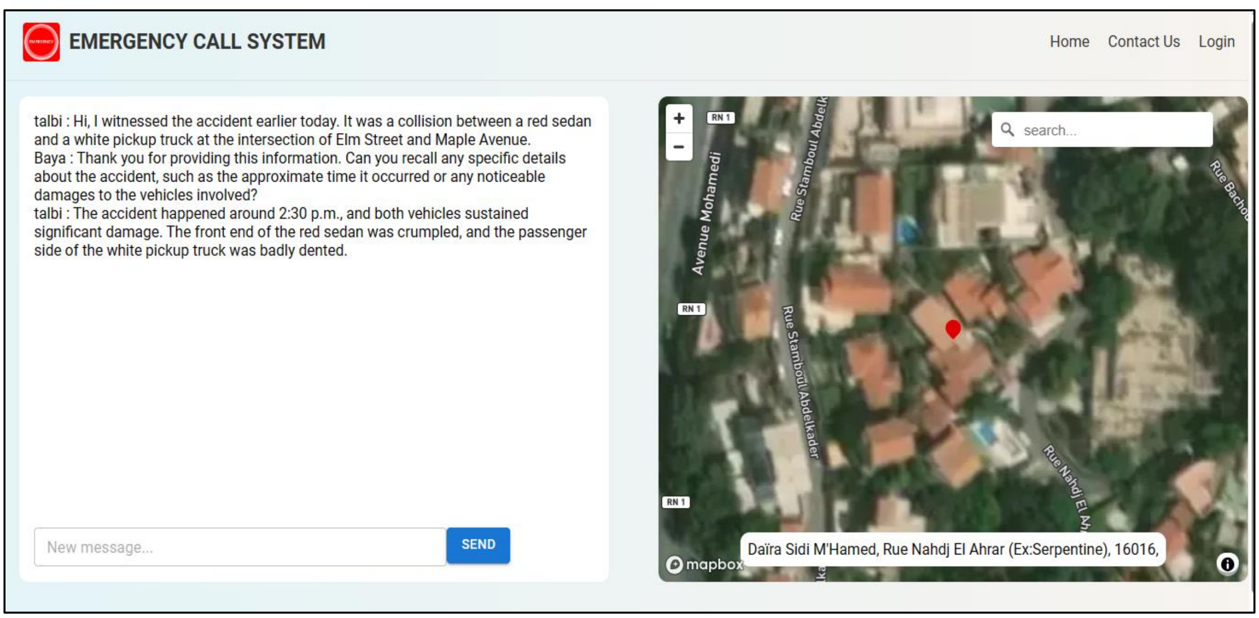

Figure 27: Chat interface

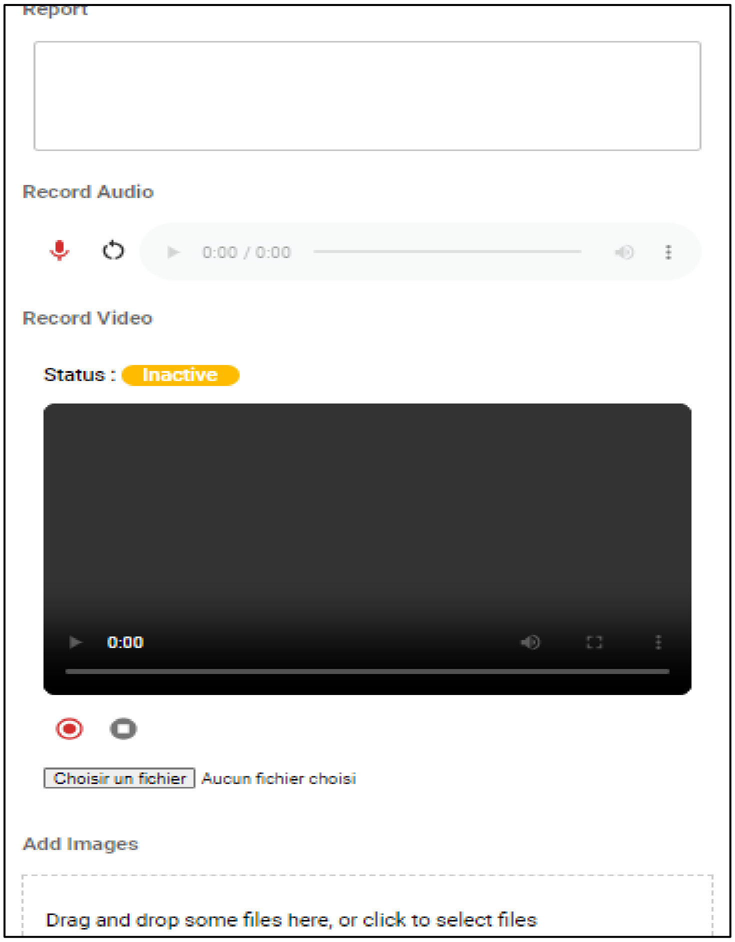

Figure 28: The user interface to submit information

Figure 29: Application Interface

5.3 Sending coordinates

In our system, we have implemented the capability for callers to send their coordinates to the operator, providing precise location information for the reported incidents. This feature enhances the emergency call process by enabling operators to accurately identify the incident location and dispatch appropriate resources.

The system incorporates a map interface that allows callers to manually pinpoint their location on the map. By interacting with the map, callers can navigate to their exact location, and mark the spot where the incident occurred.

The system offers the option to automatically retrieve the caller's coordinates through GPS, the system utilizes the device's GPS capabilities to determine the caller's current location. By using GPS, callers can provide their coordinates without the need for manual input, saving time and reducing the risk of errors in location reporting.

By incorporating these methods of sending coordinates, our system enables callers to provide precise location to the operator, whether through manual selection on the map or using GPS, the system ensures that operators receive the accurate location. This feature enhances the emergency call process.

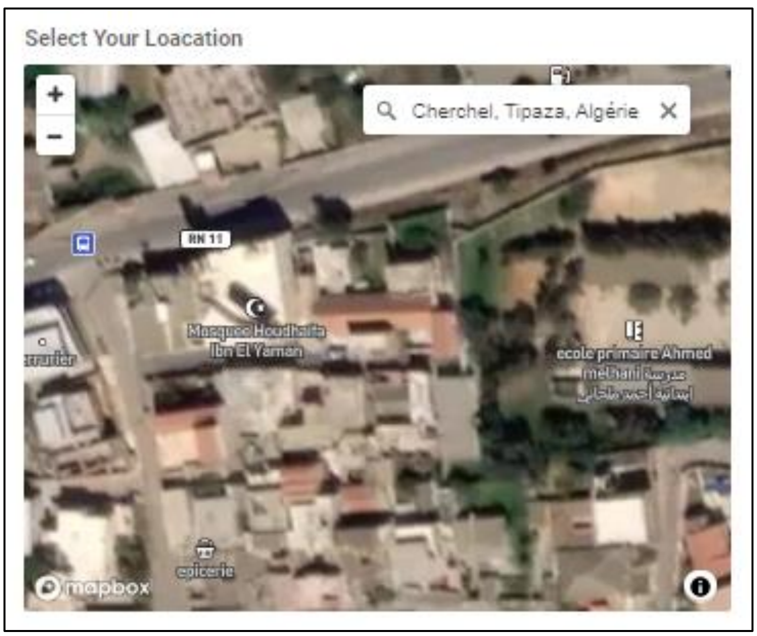

Figure 30: Map

5.4 Conclusion

in this chapter we have presented the functionality of our web application and it's features. Video communication, data exchange, sending coordinates are all key features of our web application

General Conclusion

In conclusion, this thesis has successfully designed and implemented an advanced emergency call system that takes advantage of 4G LTE technology, WebRTC and the spread of mobile phones, the system enables real-time incident reporting, facilitating faster response times and more efficient resource allocation. Multimedia communication capabilities allows callers to provide visual information, audio recordings, and text-based reports, enhancing the accuracy of communication between callers and operators. The system's accurate location tracking feature ensures that emergency responders can precisely locate incidents, enabling them to reach the scene quickly and effectively.

By improving real-time incident reporting and response and offering offline features, the system enhances public safety and emergency management. It enables operators to receive timely and accurate information, leading to better decision-making and more effective coordination of resources. The system's capacity for handling multimedia communication improves situational awareness, allowing first responders to perform more thorough assessments of incidents and offer the right kind of aid.

Future Directions

Moving forward, several avenues for system enhancement and expansion can be explored. One potential direction is the integration of emerging technologies like 5G, which offers even faster and more reliable communication capabilities. The adoption of 5G can further enhance the system's performance, enabling seamless multimedia communication, enhanced data transmission speeds, and improved location accuracy.

Another potential future direction involves the development of a mobile application to Expand the accessibility and reach of the emergency call system by developing dedicated mobile applications for popular platforms like iOS and Android. Mobile applications offer convenience and ease of use for users, allowing them to access the system directly from their smartphones.

Additionally, to ensure the system's functionality in high-pressure situations such as earthquakes, it is crucial to enhance its resilience and adaptability. Implement mechanisms that allow the system to withstand increased user traffic and provide uninterrupted service during such critical events. This may involve scaling up the infrastructure, optimizing resource allocation, and implementing load balancing techniques to distribute the workload efficiently.

Furthermore, consider augmenting the system by replacing the traditional operator in the help center with an intelligent program or chatbot. Develop advanced natural language processing algorithms and machine learning models to enable the program to understand and respond to emergency calls effectively. This automation can significantly reduce response times, eliminate the possibility of human error, and handle a higher volume of incoming calls simultaneously. The program should be designed to interact with callers, gather relevant information about the emergency, and provide appropriate instructions or assistance based on predefined protocols and real-time data analysis. This approach ensures consistent and accurate responses while alleviating the strain on human operators during peak periods.

References

[1] S. Rana, S. Sengupta, S. Jana, R. Dan, M. Sultana and D. Sengupta, "Prototype Proposal for Quick Accident Detection and Response System," 2020 Fifth International Conference on Research in Computational Intelligence and Communication Networks (ICRCICN), Bangalore, India, 2020, pp. 191-195, doi: 10.1109/ICRCICN50933.2020.9296153.

[2] E. Kyriacou, R. Constantinou, C. Kronis, G. Hadjichristofi and C. Pattichis, "eEmergency System to Support Emergency call Evaluation and Ambulance dispatch Procedures," 2020 IEEE 20th Mediterranean Electrotechnical Conference (MELECON), Palermo, Italy, 2020, pp. 354-357, doi: 10.1109/MELECON48756.2020.9140544.

[3] Al-khafajiy, M., Kolivand, H., Baker, T. et al. Smart hospital emergency system. Multimed Tools Appl 78, 20087–20111 (2019). https://doi.org/10.1007/s11042-019-7274-4

[4] J. J. A. Arnez, M. G. L. Damasceno, R. K. G. Do Reis, L. A. Da Silva, L. B. C. Tribuzy and M. C. Lucena, "Analysis of emergency call tracking in a 4G/LTE mobile network," 2022 IEEE-APS Topical Conference on Antennas and Propagation in Wireless Communications (APWC), Cape Town, South Africa, 2022, pp. 077-082, doi: 10.1109/APWC49427.2022.9900040.

[5] Bram, S., & Vestergren, S. (2012). Emergency Response Systems: Concepts, features, evaluation and design. Retrieved from http://urn.kb.se/resolve?urn=urn:nbn:se:liu

[6] https://www.techtarget.com/searchunifiedcommunications/definition/WebRTC-Web-Real-Time-Communications
[7] https://techrrival.com/webrtc-encryption-and-security/
[8] https://developer.mozilla.org/en-US/docs/Web/Media/Formats/WebRTC_codecs#supported_video_codecs
[9] Y. Zhou, X. -k. Li, M. -s. Liu, J. -l. Sui and F. -x. Gao, "Application prospect of 4G/LTE technology in PROFIBUS," 2016 IEEE International Conference on Mechatronics and Automation, Harbin, China, 2016, pp. 1268-1272, doi: 10.1109/ICMA.2016.7558744.

[10] https://www.upwork.com/resources/what-is-scss
[11] https://www.mapbox.com/
[12] https://reactjs.org/
[13] https://nextjs.org/learn/foundations/about-nextjs/what-is-nextjs
[14] https://developer.mozilla.org/en-US/docs/Web/JavaScript
[15] https://nodejs.dev/en/about/
[16] https://www.bmc.com/blogs/dbms-database-management-systems/
[17] https://www.sqlite.org/about.html
[18] https://webrtc.org/
[19] F. Weinrank, M. Becke, J. Flohr, E. Rathgeb, I. Rungeler and M. Tuxen, "WebRTC Data Channels," in IEEE Communications Standards Magazine, vol. 1, no. 2, pp. 28-35, 2017, doi: 10.1109/MCOMSTD.2017.1700007.