sustainability

MDPI

Article

A Framework and IoT-Based Accident Detection System to Securely Report an Accident and the Driver's Private Information

Amal Hussain Alkhaiwani [1,*] and Badr Soliman Alsamani [2,*]

[1] Computer Science Department, College of Computer and Information Sciences, Imam Mohammed Ibn Saud Islamic University (IMSIU), Riyadh 11432, Saudi Arabia
[2] Information Systems Department, College of Computer and Information Sciences, Imam Mohammed Ibn Saud Islamic University (IMSIU), Riyadh 11432, Saudi Arabia
* Correspondence: aalkhaiwani@sm.imamu.edu.sa (A.H.A.); bsalsamani@imamu.edu.sa (B.S.A.)

Abstract: Road traffic accidents in Saudi Arabia have become a serious issue because many of these accidents lead to deaths, injuries, and financial losses. Human lives are often lost in road accidents due to the delay in accident detection by medical assistance. In fact, the accident's location and the driver's personal information are considered critical information that plays a vital role in preserving human life. Additionally, previous studies have found a limitation in the encryption of sensitive data; in fact, a leak of private information is thought to be one of the challenges that restrict the use of IoT devices. To resolve this problem, this research presents an intelligent security framework, and an Internet-of-Things-based system is proposed for immediate accident detection. Thus, this system requires the highest level of security and privacy to maintain the driver's privacy. Moreover, the design science research methodology was followed to design and evaluate the artifacts. Thus, the study's research resulted in the ability to design a secure and effective IoT-based system to detect and report a car accident instantly. In addition, the message is encrypted using Elliptic Curve Integrated Encryption and sent through Message Queuing Telemetry Transport over GSM. The study's overall results show the flexibility with which the proposed artifact can be used for other purposes related to the IoT security framework to send and encrypt critical information.

Keywords: Internet of Things; security; design science research; GSM; GPS; elliptic curve integrated encryption scheme

check for updates

Citation: Alkhaiwani, A.H.; Alsamani, B.S. A Framework and IoT-Based Accident Detection System to Securely Report an Accident and the Driver's Private Information. *Sustainability* **2023**, *15*, 8314. https://doi.org/10.3390/su15108314

Academic Editors: Mourade Azrour, Azidine Guezzaz, Imad Zeroual, Azeem Irshad, Jamal Mabrouki, Said Benkirane and Shehzad Ashraf Chaudhry

Received: 13 April 2023
Revised: 13 May 2023
Accepted: 16 May 2023
Published: 19 May 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

Nowadays, the number of road accidents has been increasing around the world, especially in Saudi Arabia. The number of cars on the road is increasing with the growing population, which contributes to the serious number of accidents that occur daily. Saudi Arabia has been listed as a country with a high number of road accidents [1]. According to statistics, 330,454 automobile accidents occurred in Saudi Arabia in 2022. The World Health Organization predicts that by 2030, traffic accidents may cause 500 million injuries and 13 million fatalities worldwide. If quick action is not taken, this prediction will come true [2]. In fact, the lack of immediate accident detection, which might save a person's life by a few seconds, is the most common reason for a driver's death when they are involved in an accident. Once an accident occurs, the passengers' lives are put at risk. An accident detection system would detect the accident in a faster way and with a shorter response time, which can provide the variance between life and death in a matter of minutes or seconds. According to statistics, even one minute of rapid response to accidents can save 6% of lives [3]. Therefore, every car should have an intelligent device that not only detects road accidents but also immediately alerts the first responders with the required information. In today's computing world, the IoT has been reaching unexpected bounds. It is a concept

that has the potential to influence not only human lives but also how they function [3], where the selected IoT devices are the optimal solutions, which have the potential to play a critical role in promoting sustainability by optimizing resource usage and promoting sustainable practices in various industries. As such, it is essential to continue to develop and integrate IoT technology into sustainability initiatives to help build a more sustainable future. In addition, the IoT is an emerging technology that intersects with many fields, including science, industry, engineering, and policy. The IoT refers to a variety of products, including sensors [4]. Smart sensors are at the heart of the IoT, without which it would not exist. For communication, these sensors form a huge network. They record minute details of their surroundings and communicate this vital information to one another. Relevant actions are then taken in response to the information obtained [3].

Today, the IoT has improved systems to pave the way for intelligent technology to detect or monitor if human lives are in critical situations [5]. An example of these improved systems is an intelligent transportation system (ITS) that uses IoT sensors to detect accidents and traffic jams. Furthermore, there are a variety of IoT applications, such as home automation systems, that can control a home's electronic devices from a mobile phone. The advent of IoT technology has simplified the prediction of natural disasters and the reporting of temperature fluctuations by monitoring the environment using sensors. Additionally, the IoT is used in healthcare facilities for monitoring patients' health parameters and activities [6].

In Ref. [7], it is indicated that the number of IoT devices will reach 75 billion by 2025, up from approximately 7 billion in 2021. As a result, the internet network will soon become even more complex. In addition, the IoT has three common layers: perception (i.e., sensing interface domain), network (i.e., networking domain), and application layers (i.e., cloud domain) from bottom to top [8]. Each IoT layer is designed to perform a specific function. The perception layer works to gather information using IoT objects. This layer includes RFID tags, sensors, and cameras, which are responsible for collecting information. The network layer transmits the data gathered by the physical layer, also known as the IoT's heart. The application layer is the third layer. This layer's goal is to work as a link between industrial technology and the IoT's social demands [6].

The IoT has become an essential component of potential solutions that span from industrial to daily human life. This new technology is appealing for the facilitation of human life, as it adds a new dimension of knowledge to artifacts and automates decisions [7]. However, the authors in Ref. [9] present some of the IoT's vulnerabilities, which are unprotected network services, not enough authentication/authorization attempts, privacy-violating concerns, unsatisfactory security configurability, and insufficient transport encryption/integrity verification. The latter, due to insufficient encryption/integrity operations, may transport unencrypted data and credentials. A leak of confidential data is considered one of the challenges that limit the utilization of IoT devices. Thus, the contribution of the proposed research is as follows:

- To develop an intelligent IoT framework for instantly and securely detecting and reporting car accidents;
- To develop an IoT product that considers security and privacy requirements for protecting critical information;
- To respond as soon as possible to injured people before the situation becomes critical;
- To evaluate and apply lightweight cryptography (LWC) for preserving and encrypting sensitive information.

Hence, our proposed framework is also designed to contribute to sustainability by improving road safety, reducing traffic congestion, protecting privacy, and promoting more efficient use of resources. This paper is organized into seven main sections. First, Section 1 is the introduction, which includes the research motivation, study issues, and study objectives. Section 2 provides the related work, and this section reviews earlier research pertinent to IoT-based accident detection systems, IoT security and privacy, and LWC to determine which problems have already been addressed and which still need to be

studied further. In Section 3, the methodology is provided by outlining the study's phases, case studies, and their analysis, as well as the software and the design of the components employed; this section illustrates the research approach. Section 4 provides an analysis of the requirements and the design; this section discusses the requirements, explains how to assess the environment and difficulties associated with IoT security artifacts, and provides the architecture and aims of the system's design. Then, Section 5 covers the phases of the study, detailing how to run the stages of the IoT artifact, implement the elliptic curve integrated encryption scheme (ECIES) of the LWC algorithms, and provide the overall design of the proposed artifact. Next, the evaluation and results are provided in Section 6, covering the study's findings and detailing how to evaluate the IoT security artifact, providing an analysis of the test results and the knowledge base that will be contributed. Finally, Section 7 is the conclusion and provides future work. The main conclusions of these case studies are outlined in this chapter, along with recommendations and ideas for further study.

2. Related Works

2.1. IoT Accident Detection System

Vehicle tracking systems are utilized in a variety of industries around the world, including vehicle location tracking, stolen vehicle tracking, and fleet management [10]. The author of [11] proposed a simple vehicle tracking system that can be used in different situations and cases, such as if the car is stolen (i.e., theft detection) or when parents need to track their children's school bus. The vehicle tracking system used GPS and GSM technology to track and send SMS messages. In addition, the author concluded the research with future work by monitoring some parameters of the vehicle, such as an LPG gas sensor and gas leak alarm.

The IoT enables the tracking and monitoring of a variety of operations (for example, an IoT-based framework for vehicle overspeed detection). The concept of the proposed system is to estimate the time required for a certain vehicle to travel from its starting point to its destination. The smart vehicle overspeed detector assesses a vehicle's speed using radar, according to the data. This information is gathered and wirelessly transmitted to the appropriate authorities at a remote location using IoT technology. The device includes a GPS-sensing module with a transmitter and receiver that works in tandem with an electronic tracking device to determine a vehicle's speed. If a speeding car is detected, the proposed device emits a buzzer signal to alert the authorities. The accuracy of the speed tracking is predicted by the Connection App, which uses radar, to be between 40 and 80 percent, depending on the internet speed and connectivity [12].

Moreover, the authors of [13] proposed a fully functional system that works to detect accidents using a shock sensor and sends SMS messages through a GSM module. The reliability test of the proposed system showed that the system is robust, available, valuable, operative, and workable, especially when the IoT device continues to send continual crash notifications until it confirms receipt by the authorities.

The authors of [14] proposed a system that can identify accidents through an accelerometer and GPS in conjunction with a smartphone. On the server, there is data pertaining to the accident. Further, the system is more dependable than others, but failure may still occur in the event of a server failure. Additionally, Khot et al. [15] developed a smartphone-based system that employs an accelerometer to recognize accidents and report the location of such accidents to emergency personnel. Once more, the single point of failure in this system increases the likelihood of an erroneous accident alert. Furthermore, in another paper, Chaturvedi et al. [16] proposed an accident detection and reporting system that recognizes incidents with the aid of one sensor. A location is delivered to the police station when an accident occurs. Moreover, in Ref. [17], a prototype for automatic accident detection using the Vehicular Adhoc Network (VANET) and the Internet of Things (IoT) is shown. It uses mechanical and medical sensors installed in the car to detect accidents and the severity of emergency situations.

2.2. IoT Privacy and Security

The IoT facilitates smart devices becoming more traceable, and, in turn, privacy and security issues have multiplied. In Ref. [18], a single IoT environment that was extensively researched is presented, and the findings relating to privacy, security, and defect detection are provided. Thus, the study of the numerous security challenges related to the IoT is extremely important. One target objective of IoT security is to ensure the privacy and confidentiality of all users, as well as enhanced protection, infrastructure, and a guarantee of the availability of various services provided by the IoT ecosystem. Regardless, issues concerning security and data privacy must be addressed. Appropriate measures must be taken to make the user feel at ease about being a part of the IoT system and sharing private information. It must be clearly defined who owns the data and where it will not be utilized without their permission, especially if the data are shared via the Internet [6].

Thus, IoT privacy and security are the main aspects that must be emphasized. For achieving the security and privacy requirements, there should be data confidentiality, access control, an IoT network, privacy, and trust among users and smart devices, including the enforcement of security and privacy policies [8]. Moreover, the Open Web Application Security Project's (OWASP) IoT Top 10 list named the most popular issues related to the IoT, including privacy concerns surrounding what would happen if critical data were exposed or viewed by unauthorized users. Another IoT issue concerns a lack of data transport encryption, which may cause data leakage or lead to a device or user account being completely compromised [19].

In addition, the authors of [20] presented the most critical threats to IoT devices, which are identification, such as the name and address of the individual entity, and sensitive information, including localization and tracking. The latter, localization and tracking, are considered to be the threat of identifying an individual's position using various methods, such as GPS, internet traffic, or smartphone location [20]. The authors reviewed a total of 122 original research papers on IoT privacy that summarized the top concerns about IoT privacy and security. In fact, location tracking and sharing private information are considered the IoT's top vulnerabilities and concerns that must be solved. Furthermore, they presented an optimal solution for preserving privacy by performing cryptographic techniques, privacy awareness, access control, and data minimization. First, for cryptographic techniques, the researchers may have spent many years suggesting new privacy-preserving strategies. The encryption procedure remains the lead technology in most currently proposed solutions [19].

2.3. Lightweight IoT Cryptography

IoT devices are prone to malicious attacks and data theft due to the fact that critical information is transmitted across public networks. Advanced technology is required to safeguard the system. Thus, cryptographic algorithms are a useful way to ensure data security in the IoT. On the other hand, many IoT devices are still insufficiently able to provide such strong solutions. As a result, algorithms must consume less energy while preserving their efficiency to be used in the IoT [6].

In Ref. [7], the author found that most IoT devices do not currently use robust encryption or authentication techniques in their connections, which can corrupt the transmission of data and lead to eavesdropping attacks. For example, many implantable medical devices, such as ECG pacemakers, are affected in terms of memory, processing capabilities, and power consumption because they rely on embedded microprocessors and integrated circuits (ICs). The failure to use effective encryption on these devices can have major consequences, such as unauthorized access to sensitive information and device failure [6].

In fact, cryptographic algorithms are used to ensure information is kept secret and secure through transmission without exposing it to alteration. These algorithms are divided into two categories: (i) symmetric-key algorithms and (ii) asymmetric-key algorithms. Cryptographic algorithms that use the same cryptographic keys are known as symmetric-

key algorithms. Pairs of keys are used in asymmetric cryptography, including public keys, which are available publicly, and private keys, which are only known by the owner [21].

However, IoT devices are usually small in size with limited computing capacity, memory, and power resources. Applying conventional cryptographic algorithms to IoT devices that require intensive resources is difficult. Therefore, designing lightweight protection schemes for the IoT becomes the optimal and recommended solution [22].

This research attempts to provide an in-depth and up-to-date analysis of lightweight cryptographic primitives and which ones are best to use (asymmetric, symmetric, or a combination of the two) to provide high-level data protection.

In Ref. [22], the authors provided a comprehensive evaluation and comparison among the types of lightweight cryptographic primitives and their performance. Four different types of lightweight cryptographic primitives can be used: (i) lightweight block ciphers (LWBCs), (ii) lightweight stream ciphers (LWSCs), (iii) lightweight hash functions (LWHFs), and (iv) elliptic curve encryption (ECC). A block cipher is a type of symmetrical cipher that processes an entire block at once. Substitution–permutation networks (SPNs) and Feistel block ciphers are two types of LWBCs. A stream cipher encrypts and decrypts "r" bits at a time. Another technique to provide security is to use LWHFs. They take an arbitrary-length message and turn it into a fixed-length "message digest". IoT security using ECC also utilizes an asymmetric cipher's advanced encryption standard (AES). These ciphers offer both authentication and confidentiality. AES requires a larger key size and higher memory consumption. Rivest, Shamir, and Adleman (RSA) and ECC are two primitives that can be utilized in a public-key cryptosystem for IoT encryption transmission messages compared to higher memory consumption.

In comparison to RSA, ECC provides the same level of security with a reduced key size. On 8-bit microcontrollers, AES is 100–1000 times faster than ECC. The computational complexity of the algorithm can be reduced to enhance execution time.

Table 1 presents a comparison of lightweight asymmetric algorithms, which are RSA and ECC, for the IoT environment based on their key size, code length, possible attacks, key generation(s), signature generation(s), and signature verification(s).

Table 1. Comparison of lightweight asymmetric algorithms.

Asymmetric Algorithm	Key Size	Code Length		Possible Attacks	Key Generation(s)	Signature Generation(s)	Signature Verification(s)	
RSA	1024 15,360	900	[20]	Module attack, man-in-the-middle, timing	0.16 679.06	0.01 9.20	0.01 0.03	[5]
Elliptic curve cryptography (ECC)	160 571	8838		Timing attacks, side channel attacks	0.08 1.44	0.15 3.07	0.23 4.53	

In addition, the authors conclude that AES is the preferred method for a symmetrical cryptography network, which provides provisioning and protection. ECC is the optimal solution in asymmetrical cryptography because it can offer authentication and nonrepudiation [22].

The authors of [23] presented secured text encryption cryptography using ECC. Moreover, the authors of [4] described that privacy and security in the IoT have proven to be one of the most complex issues in recent years. In addition, they demonstrated the current cryptographic models and security techniques used in encryption algorithms and privacy standards. The AES ensures confidentiality in most circumstances as symmetric encryption. Asymmetric encryption, digital signatures, and key management are all provided with the asymmetric algorithm RSA. For secure hash functions, the standards are utilized. In asymmetric cryptography, Diffie–Hellman (DH) and ECC are used to provide privacy techniques [3].

Furthermore, the authors of [21] presented a framework design for secure communication among IoT devices and gateway (i.e., intra-network). The authors recommended that, for the enhancement of changing asymmetric cryptographic techniques, RSA should be substituted with ECC for protecting data transmission [19].

2.4. Comparison of Related Works

To summarize, numerous systems place a high priority on accident detection accuracy, as shown in Table 2, which provides a summary of previous studies. In addition, several systems are designed to respond quickly in order to reach the location of the accident. The use of smartphone sensors can also lower a system's overall cost and increase user accessibility. Researchers have offered a variety of smartphone-based alternatives. The accident detection and response systems' sensory inputs are described in detail in Table 3. As can be observed, there are currently only two different types of sensors that can be employed in systems, and no previous studies have used lightweight cryptography with IoT.

Table 2. Summary of related works' findings.

Reference	Methodology/Techniques	Measure	Evaluation
[11]	Uses speed and GPS sensors	Accuracy	Prototype
[12]	Uses GPS and GSM to detect and send messages	Accuracy	Testing and simulation
[13]	Detects the accident using one sensor	Reliability	Testing and simulation
[14]	Uses two sensors: accelerometer and GPS	Accuracy	Actual implementation
[15]	Uses an accelerometer and GPS for detecting and reporting accidents	Response time	Actual implementation
[16]	Based on one sensor: accelerometer for detecting and reporting accidents	Accuracy	Microcontroller Arduino
[17]	Uses an accelerometer for detecting and reporting accidents	Response time	Testing and simulation

Table 3. Summary of sensor types and LWC used in the related works and the proposed work.

Reference	Vibration	Airbag	GPS	Other	Total	LWC Encryption
[11]	☒	☒	☑	☒	1	☒
[12]	☒	☒	☑	☑	2	☒
[13]	☒	☒	☒	☑	1	☒
[14]	☒	☒	☑	☑	2	☒
[15]	☒	☒	☑	☑	2	☒
[16]	☒	☒	☒	☑	1	☒
[17]	☒	☒	☒	☑	2	☒
Proposed work	☑	☑	☑	☒	3	☑
	☑	Include	☒	Is not include		

The following table summarizes several researchers who have proposed works or platforms for automatically locating accidents and informing authorities about them. Despite a few developed proprietary solutions, most of the systems rely on smartphones. Typically, these latter systems usually require manual activation and restrict calls to call centers. The proposed system in this research is composed of the following elements: navigation, communication, accident detection encryption, and rescue. It makes use of three sensors, including accuracy, response time, and encryption of tracking information, for accident detection, which are key components of our technology.

3. Research Design

The appropriate research methodology for this study is to use design science research (DSR). DSR is a type of research paradigm that allows researchers to provide answers addressing human issues and to create valuable artifacts. Furthermore, DSR is used to grow the existing knowledge base [21]. In fact, the developed artifacts are both helpful and challenging to understand, but they solve real-life problems [24].

3.1. Research Methodology

The DSR framework comprises three research cycles: relevance, rigor, and design. First, the relevance cycle works to take the requirement inputs from the contextual environment into the research and initiate the research artifacts to be presented in the testing field. Second, the rigor cycle works to provide the background theories and methods according to the domain experience and expertise in the foundation's knowledge base. Thus, the rigor cycle acquires the new knowledge generated by research and uses it to grow the knowledge base. Third, the central design cycle works as a tighter loop of research activity for constructing and evaluating design artifacts and processes. In addition, the work with these three cycles in a research project clearly defines the positions and differentiates design science from other research paradigms [25]. Figure 1 defines the general combined DSR framework published in Refs. [25,26].

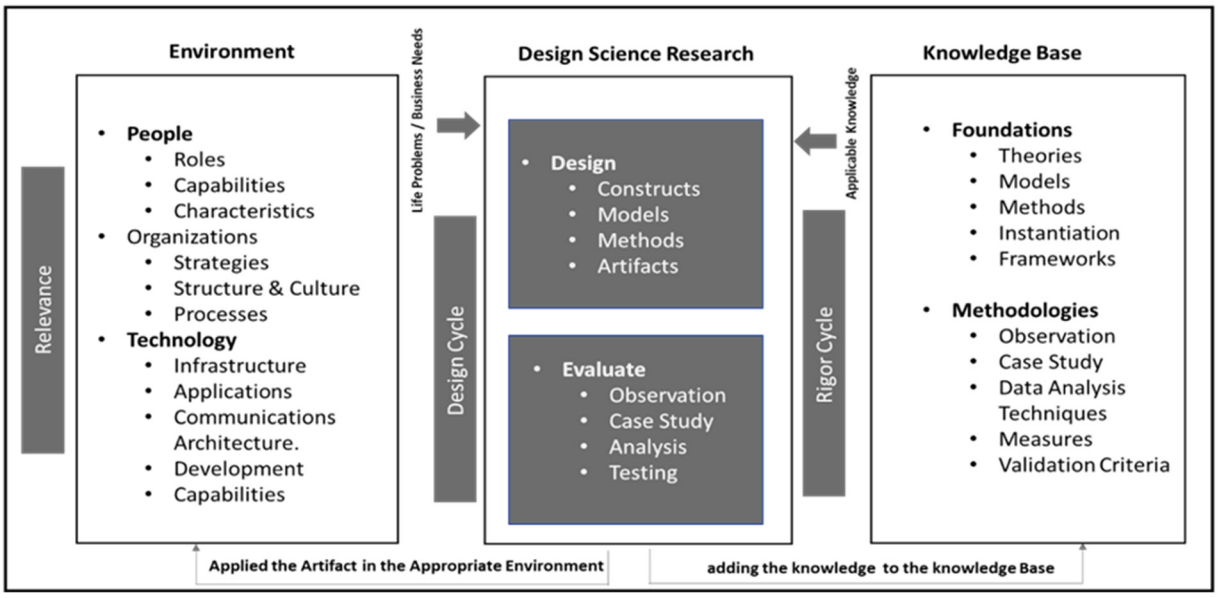

Figure 1. Design science research framework (based on on Hevner, 2007; Hevner., 2004 [25,26]).

3.2. Cycle Design

To explain how the cycles work in this study, Figure 2 presents the roadmap stages and cycles of the research methodology to initiate the process and understand the proposed research, which can be observed in the following:

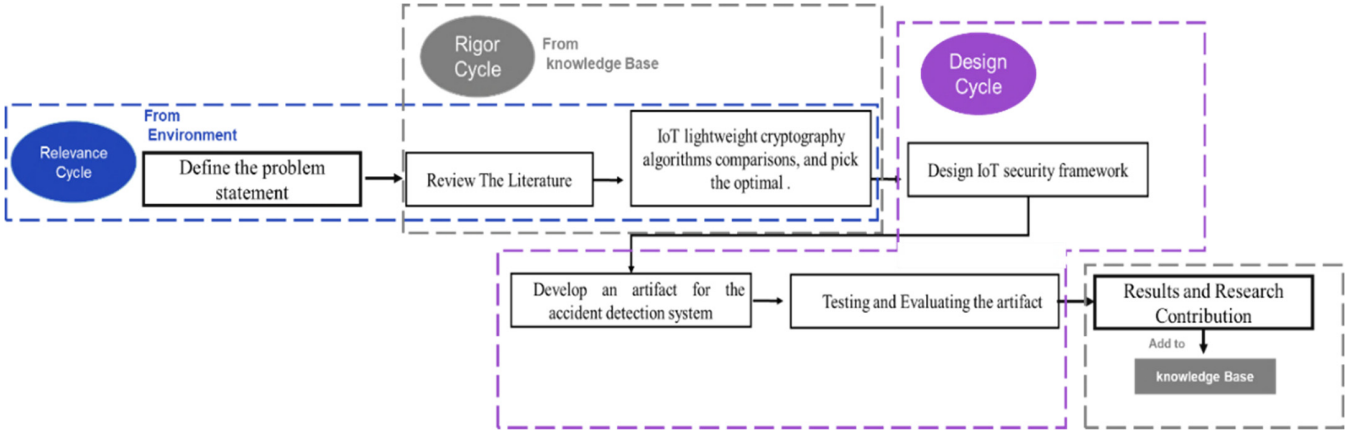

Figure 2. The proposed research roadmap reflects the research methodology.

1. Relevance cycle: The first stage is to define the problem in context and the requirements: a secure and effective IoT-based system to detect and report car accidents instantly while ensuring privacy. In this cycle, as explained in the figure, the IoT security framework literature is searched, and the requirements needed to implement the proposed artifact are determined. Thus, this cycle specifies the scope of the research and the literature review needed for this work.
2. Rigor cycle: This cycle uses a literature review of previous studies that have focused on similar issues. Moreover, the information gathered from the literature will then be used to provide a comprehensive comparison of IoT LWC algorithms. Equally important, this cycle also provides a summary of all studies on IoT security frameworks related to detecting and reporting car accidents instantly.
3. Design cycle: A comparison between the IoT and LWC algorithms will assist in delivering an IoT security framework considering the delivery of sensitive information by message and the authentication processes between the sender and recipient to prevent spoofing by an attacker. Thus, the proposed research follows the DSR process for developing and evaluating a prototype system as an IoT-based security framework artifact for an accident detection system. Additionally, the following section covers how the research methodology works, aligns with the research objectives, and develops the proposed artifact.

3.3. Development of the Study Using the Research Approach

The development of the study using the research methodology is covered in this section. The section above provides a brief explanation of the research methodology's cycles. In addition, this section describes the cycle as it appears in the research study. In addition, Figure 3 demonstrates how the three cycles of the DRS technique map to the proposed study's components.

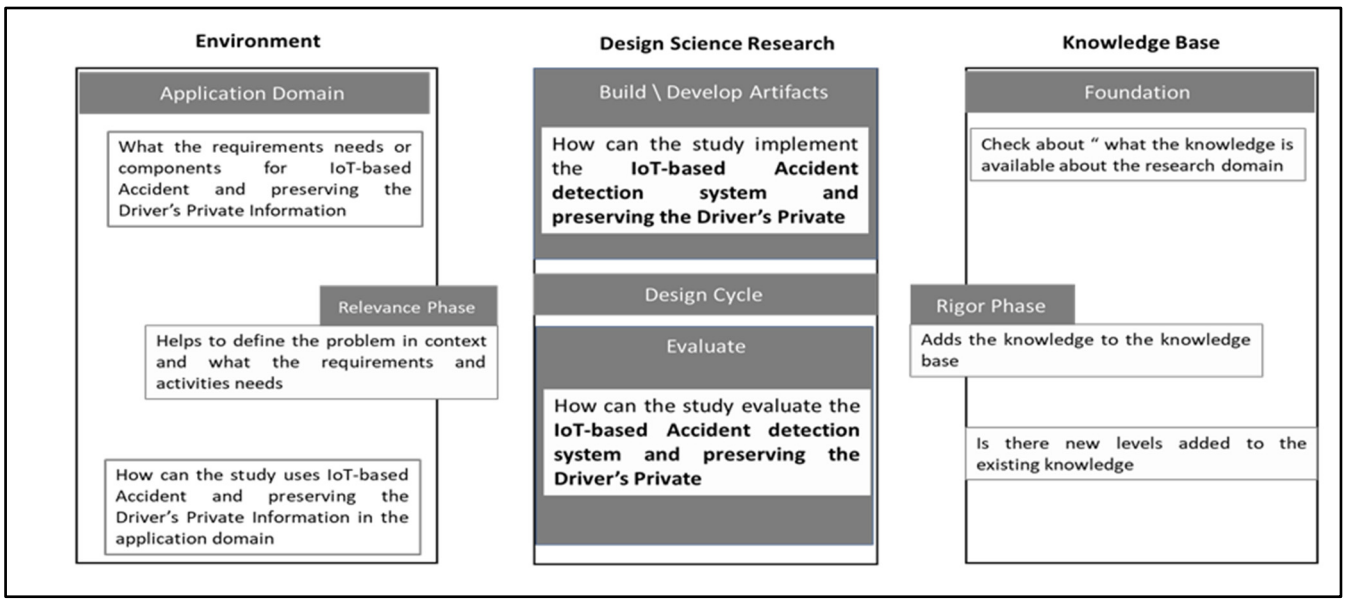

Figure 3. Mapping of the elements of the proposed research to the DSR.

The main elements of the design science research involved in the cycles.

1. Environment (relevance cycle): With this element, the research problem and motivation are clearly defined. In addition, the value of the solution is justified, as mentioned in Section 1, as well as in all related publications' research outputs sections that identify the problem and clearly motivate the research direction toward a solution. The related work is divided into three major sections: IoT accident detection, IoT security and privacy, and the IoT–LWC method. The sections were divided by the research's objectives to

define the problem and present the previous studies mentioned [3,5–8,10–24]. Equally important, the researcher conducted face-to-face interviews with experts in the fields of cryptography and the IoT as well as observational studies to gather data for this study.

2. Designs Science Research: This element is divided into two sections.
    - Build/develop the artifact: This section shows the design and development of the research artifact. Here, the theoretical knowledge obtained from the environment and knowledge base is applied to the creation of an artifact. To address the identified problems, this includes both functionality and architecture. In addition, the artifact is designed and tested, utilized in simulations, or used in other ways as a solution to the perceived problems [26,27]. The proposed solution of this research study was to provide a secure and effective IoT-based system to detect and report car accidents instantly. This used RPM, which relates to other technologies such as IoT sensors (airbag and vibration) and GSM and GPS modules, to detect accidents faster.
    - Evaluate: In this section, an analysis is performed as to how well the research artifact meets the objectives highlighted once the problem has been identified [27]. The evaluating and testing section includes the sections that design and implement the overall proposed artifact. Additionally, improvements can be made to the artifact to ensure the IoT-based security framework for the accident detection system meets the objectives of detecting and reporting a car accident instantly, preserving the driver's private information securely.
3. Knowledge Base:

The last activity is disseminating the obtained results and their effectiveness in solving the problems identified in relevant forums and publishing outlets. The research results are added to the knowledge base [27]. Thus, the proposed solution applies a secure and efficient IoT–LWC artifact for detecting accidents and encrypting the driver's private information in an effort to present and achieve a valuable framework that can prevent location tracking and preserve the privacy of critical information during an accident.

4. Requirement Analysis

For the requirement analysis, information was acquired from a variety of publications and proposed prototypes. To gather the required information on measurements, technologies in use, and challenges in the current approach, as well as to determine more concerning the level of awareness and the challenges that will be encountered when designing an IoT security framework for detecting accidents and encrypting driver's data.

4.1. The Overall Conditions and Challenges for IoT–LWC (Research Study Challenges)

This study's first objective was to determine the optimal solution for preserving privacy by performing cryptographic techniques. The data were used to provide insight into this research using document analysis and a literature review for IoT–LWC. According to the research published in Ref. [27], an in-depth and up-to-date, comprehensive comparison of simple cryptography techniques was presented. The publication presented 54 LWC algorithms in their respective classes, including five different ECC cryptography algorithms, 21 lightweight block ciphers, 19 stream ciphers, and nine lightweight hash functions. The efficiency of the hardware and software, chip size, energy and power consumption, throughput, latency, and figure of merit of the ciphers were compared (FoM). The research concluded with a comparison of AES and ECC as the best lightweight cryptographic primitives to use based on published research in the area of portable cryptography [28]. To protect the privacy of the IoT through messaging transmission, the study used ECC combined with AES in the ECIES algorithm. The length and transmission duration of the message could be a challenge.

The challenge and objective addressed in this research were also how to apply the ECIES–LWC algorithm for preserving and encrypting sensitive information. The second objective of this research was to determine how to apply the IoT-based LWC in the artifact

to securely design an IoT framework for detecting accidents and encrypting the driver's private information.

Challenges Faced in an IoT-Based ECC LWC

The following challenges are divided into two sections.

- Application of the ECIES algorithm:
  1. Time consumption when sending a message: This research explored the use of ECC to improve data privacy and security in the IoT. Due to its effectiveness and performance in terms of time and energy, it is intended to demonstrate that ECC is relevant for the IoT. To do so, this study also emphasizes earlier studies on lightweight ECC to demonstrate the significance of ECC and to assess the effectiveness of techniques, as described in Section 2.3 (Lightweight IoT Cryptography), to quickly report an accident to first responders and encrypt messages. Thus, this study focused on the ECIES algorithm key size and compared it with the RSA algorithm in terms of time consumption in the generation of the RSA key size. Furthermore, a comparison between the ECIES algorithm and the RSA method is covered in the evaluation.
  2. SMS message length: The message content must be kept to a minimum via a maximum SMS message length for mobile devices to achieve the objective of sending a message instantly while also being encrypted by ECC. In fact, the message can only be 160 characters long at most. For instance, if an accident occurs, a location is sent through an SMS message to first responders. For this reason, only the longitude and latitude are encrypted to keep the message short, and only one SMS is sent. However, the encrypted message is still more than 160 characters [28], and to solve this challenge, the message is sent from the device controlling the signals, encrypted, and then sent to the CServer as MQTT. The message is then sent from the CServer to the first responders.
- IoT-based Accident Detection System

The challenges of accident detection techniques and promptly alerting first responders to incidents are provided in this section. Thus, a few sensors are fitted to the vehicles to collect information on the location and direction of the car in the event of an accident. To illustrate, the airbag sensor (AS) and vibration sensor (VS) can be used as examples. The first responders receive information about the accident's location using GPS/GPRS modems. Furthermore, the sensors are examined and evaluated in this study to ensure their functionality. If an incident occurs, the vibration in the VS increases above its maximum level, or the airbag explodes out of its site (i.e., bursts). The GSM module then receives this information. Later, the GSM module transmits a message to the first responders. In brief, the integration of all of the proposed artifact's functionalities with the encryption process was the greatest challenge of this research.

4.2. Requirement Specification

In this section, different hardware, software, and platforms are required for deploying the proposed system.

4.2.1. Software Requirements

- Raspberry Pi OS Raspbian: The Raspbian operating system supports Raspberry Pi. Based on Debian, Raspbian is a free operating system designed specifically for the Raspberry Pi device [29].
- PyCharm IDE Software2022.3.2: PyCharm is a Python IDE developed by JetBrains, the organization that also produces the well-known IntelliJ IDEA IDE for Java. Any developer who wants to construct Python applications, including web applications, data science applications, or even simply basic Python scripts, is encouraged to use

PyCharm [30]. It has a code editor with tools such as automatic indentation, brace matching, and syntax highlighting. A "Python file" is written code or a program.

- The Central Server (CServer)—"Website-Database Server": All information on an accident is kept in a Microsoft SQL database, and MySQL was adopted as the database management system (DBMS). In addition, the website develops client-side interfaces using HTML, CSS, and Bootstrap. For server-side programming, it uses PHP [31].
- MQTT (message queuing telemetry transport): An IoT ecosystem uses the MQTT protocol for communication, which runs on top of the transport control protocol. Moreover, MQTT is considered a lightweight machine-to-machine communication protocol, and it was developed by IBM. It is used to send data from sensors to the server [32].
- Fritzing Software 0.9.10: The Fritzing software package can be helpful during development phases, such as the assembly of the prototype based on the scheme in the mock-up sketch and boards, as well as the automatic generation of schematic diagrams and the PCB [33].
- Twilio SMS Massage Service: The use of virtual phone numbers to deliver SMS messages is made possible by Twilio's cloud-based messaging technology [34].

4.2.2. Hardware Requirements

- Raspberry Pi Model B: The newest and fastest Raspberry Pi model, the Raspberry Pi 4 Model B, was used to build an IoT ecosystem. It features different RAM capacities (2, 4, or 8 GB), a USB-C port for power, a MicroSD card slot for the operating system and file storage, two micro-HDMI ports, and the option to connect to the Internet wirelessly or with an ethernet cable. The features of the Raspberry Pi 4 Model B used in this experiment are listed in Table 4 [35].

Table 4. Raspberry Pi 4 Model B specifications.

Raspberry Pi 4 Model B Specifications	
Operating system	Raspbian
Internet connectivity	Wi-Fi
Card size	4 GB

- Airbag sensor: An airbag is a type of occupant restraint system and a vehicle safety component. It is composed of an inflatable fabric bag, an impact sensor, an inflation module, and an airbag cushion. If an accident causes the airbag to blow, the sensor detects it. If the airbag bursts, the airbag sensor detects that an accident has occurred [36]. The digital airbag sensor in this research requires three wires for connection: a ground pin wire (GND), a voltage pin (+5 Vcc) to supply the circuit with the required electricity, and an input pin to communicate with the Raspberry Pi.
- Vibration sensor (VS): This is a transducer that transforms vibrations into an electrical equivalent, such as a voltage, such as one that uses a laser or piezoelectric crystal. It is also known as a vibration transducer. It detects the vibration of a car, which is a specific and substantial vibration. However, vibration sensors are used to measure and analyze linear velocity, displacement, proximity, and various shock triggers [37].
- GPS and GSM modules: A radio navigation system called GPS, or "global positioning system", enables land, sea, and aerial users to pinpoint their precise location, speed, and time at any time, day or night, in any weather, anywhere in the world. An accident's location will be known. In addition, it is a digital mobile telephony system that is widely utilized in Europe and other parts of the world. It stands for global system for mobile communication [38]. Furthermore, the GSM and GPS modules are utilized as a security system to improve the project and make it more secure by calling or texting the user's phone number if there is a car theft attempt [39].
- Power supply: This is an electronic device that supplies electric energy to an electrical load [38].

4.2.3. Proposed System

This research addresses the challenges by developing an intelligent security framework and an IoT-based system for immediate accident detection. It offers the following primary actions: instant detection of an accident and sharing of the driver's personal health record and other crucial information with the first responders (for example, Najm and ambulance) while ensuring privacy. Figure 4 explores the IoT security framework accident detection system artifact proposed in this research.

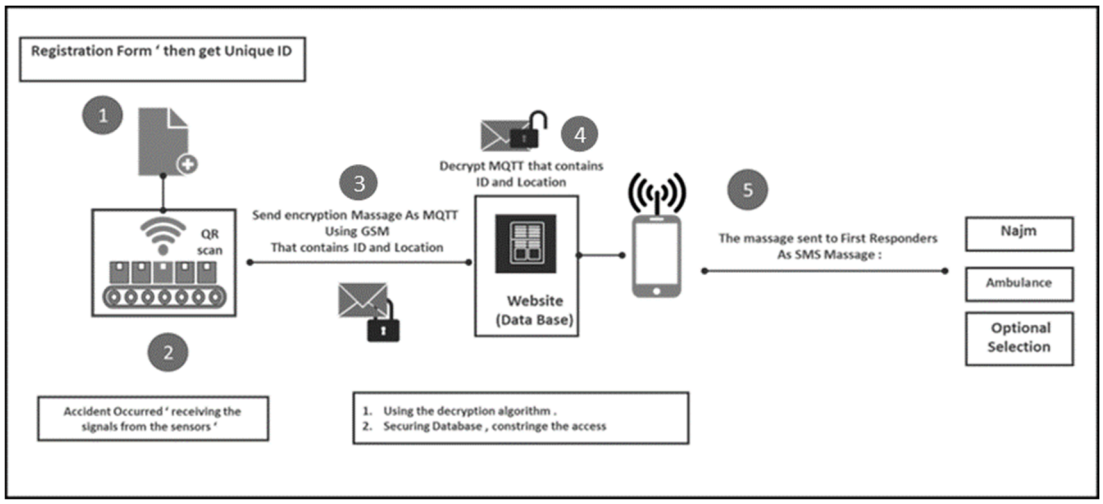

Figure 4. Accident detection system conceptual block diagram design.

First, the device's unique ID is a code generated when the driver fills out their vehicle owner information record and personal health record. Once an accident occurs, the AS and VS that work together to report the accident instantly send the signal to the RPM to retrieve the location from the GPS. Next, the GSM module enables the system to send sensitive data about the accident using the message queue telemetry transport (MQTT) protocol to the CServer. The message also includes coordinates from the GPS module and a unique ID that connects to the system. The message must apply the proposed ECIES algorithm to protect and preserve its security and privacy. Then, to prevent a potential attack, there is an authentication phase between the sender (i.e., the IoT device) and the recipient to check that the alerted accident is real. After that, the Cserver sends a message as an SMS to the first responders, who are then able to view the data.

Eventually, the accident detection system's artifact prototype is tested and evaluated under real-life conditions, and its performance is evaluated using the system's experimental setup.

4.3. System Design

4.3.1. Design Goal

An intelligent security framework and an IoT-based system are proposed as solutions to this issue for instant accident detection. Its main features are instant accident detection, privacy protection, and the ability to share a driver's personal health information with first responders, such as an ambulance and Najm.

4.3.2. Security Issue

This study focuses on two aspects: MQTT privacy transmitted by the artifact and critical information sent as an SMS provided by the website. In any scenario, the system and method of sharing messages should be secured. In fact, it is intended that only authenticated and authorized first responders on the website side will have the right to receive critical information. On the other hand, by creating an artifact that offers a secure and efficient IoT-based system to instantaneously detect and report accidents, this research protects the privacy of drivers' information during SMS transmission.

4.4. System Architecture

The proposed architecture securely designs an IoT framework for detecting accidents and encrypting drivers' private information. Accordingly, the process flow is shown in the following diagram (Figure 5).

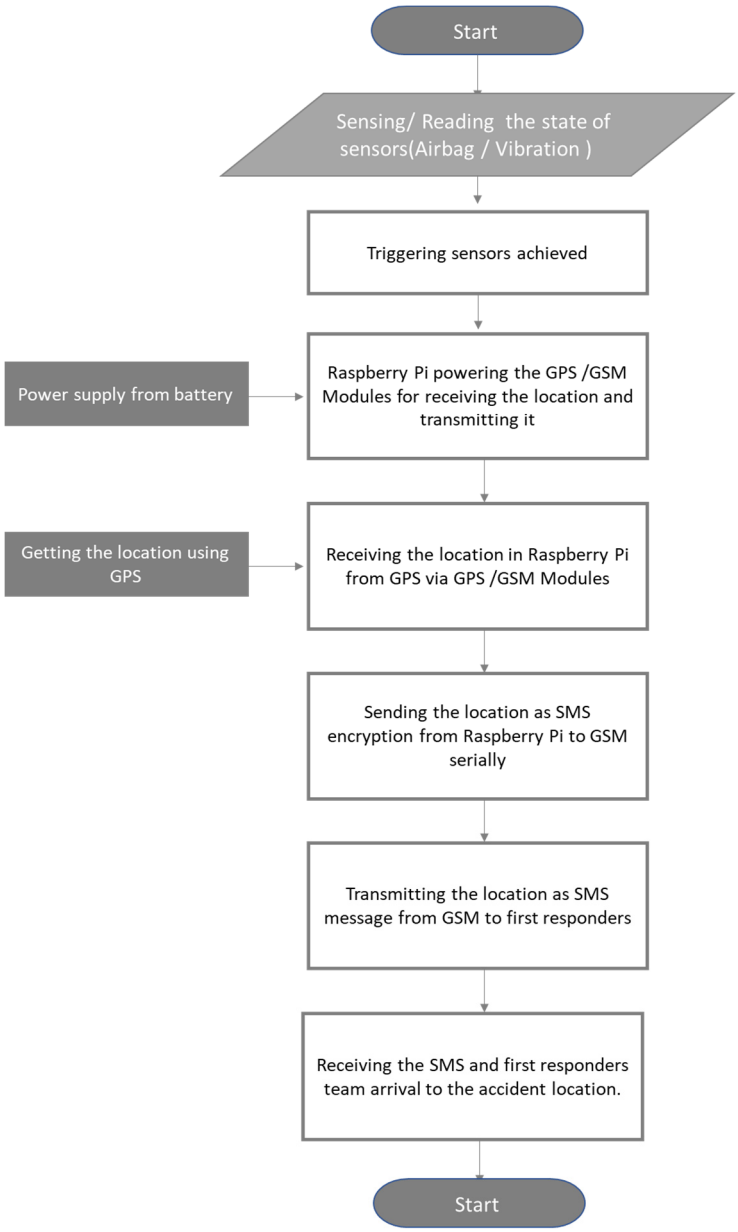

Figure 5. Flowchart for intelligent IoT-based accident detection.

System flowcharts are used to display the proposed project's processes, from the start of accident detection to the end, when the CServer sends an accident report to the first responders.

4.4.1. Block Diagram

A block diagram displays a complex system or process, such as an electronic circuit, in schematic form, along with a general layout of its pieces or components. In addition, adopting this type of diagram will simplify the IoT-based security framework to detect and report a car accident instantly as a block diagram.

The proposed architecture, as shown in Figure 6, is composed of the following elements: first responders (e.g., Najm and ambulance), relationships among them, A9G

GSM/GPS modules, tower, AS, VS, power supply, Raspberry Pi 4 Model B, and the CServer.

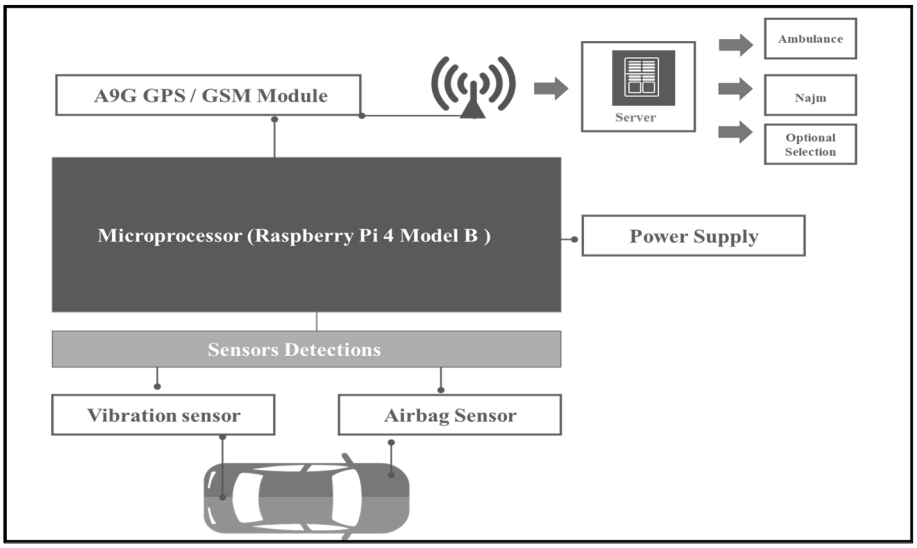

Figure 6. Block diagram of intelligent IoT-based accident detection.

4.4.2. Sequence Diagram

A graphic that illustrates interactions among items and captures how activities are carried out is called a sequence diagram (Figure 7). Therefore, it demonstrates how and in what order various items interact. Additionally, a time-sequenced display of item interactions is provided.

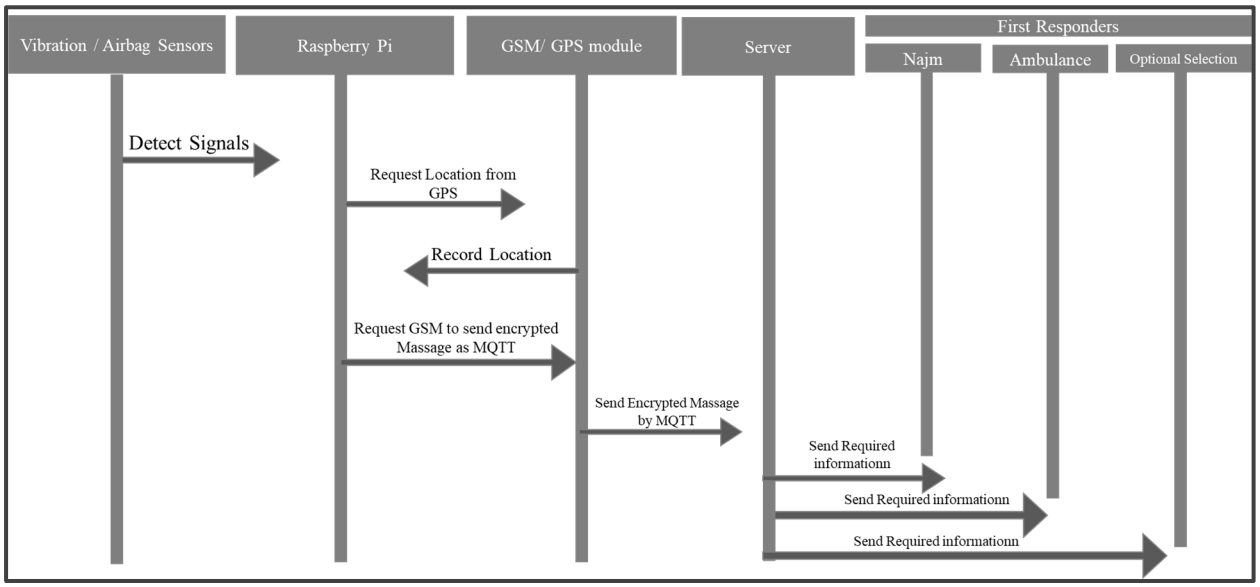

Figure 7. Sequence diagram of automated requests.

5. Implementation

This section discusses the implementation of the prototype using the RPM, sensors, GSM/GPRS, and other modules to communicate with the CServer (website and database server), and the results are presented. The following sections show the implementation of an IoT-based intelligent security framework for immediate accident detection artifacts.

Development Tools

A discussion of the development tools used for the prototypes is provided in this section in phases. Each phase contains the tools used to implement the prototype of the proposed system.

1. Phase 1: QR Code Scanning (Website—Database Server):

This section covers how the required information from the driver will be entered and saved by scanning a QR code generated by the system. After that, the data is stored in the database. Thus, a new record and unique ID are established for the device that was used throughout the accident. Moreover, the system provides an option if the driver needs to share the accident data with any of their family members or their family doctor during the accident.

Additionally, MySQL is the database management system (DBMS) that manages the entire system. Detailed information on the different tables used in the proposed system is as follows:

- Driver details table: This table includes all driver-related data, including Driver_Name, Driver_Address, Driver_Email, and Driver_Phone number;
- Vehicle table: Vehicle information, including Vehicle_ID, Vehicle_Name, and Vehicle_Insuracne, is available and provided in this table;
- Diseases (medical records) table: This table contains data on all diseases and the health insurance of the driver;
- Device table: This table contains all data related to the device, such as Device_ID, Driver_ID, Vehicle_ID, and the private keys utilized in the decryption process. The system generates the QR code for each vehicle to allow the driver to input the required information;
- Accident table: This table is used to keep track of all details of accidents, such as Driver_ID, Device_ID, Location_ latitude, and Location_ longitude, including those that were used during the notification phase.

2. Phase 2: IoT Accident Detection Sensors
    - Airbag sensor (AS): While developing the circuit, an AS is linked to the RPM. Thus, the AS works to detect the signals when an airbag is bursting. Furthermore, the AS can be added to the breadboard and connected to the microprocessor to complete the circuit;
    - Vibration sensor (VS): The second sensor used in this research is the VS, which works to detect signals when it is activated. It also employs a digital VS. The sensor's vibration frequency ranges from 40 Hz to 65,535 Hz when it is attached to the RPM. In addition, the shock sensitivity of the VS was modified by selecting a time alert to detect the shock. Periodically shaking the sensor causes it to automatically produce "1" when working as a digital signal (as an indicator of shaking or movement).
3. Phase 3: Accident Transmission Encryption Message
    - Raspberry Pi Microprocessor (RPM): In this research, the system uses a Raspberry Pi 4 Model B connected to a CServer, an AS, a vibration sensor, GSM/GPRS, and other devices to detect and report accidents instantly. The Raspberry Pi receives the signals, requests the location of the accident, encrypts the message, and sends the encrypted message to the CServer; thus, the IoT devices connect to the Raspberry Pi through input pins (Figure 8), and the CServer retrieves the data through MQTT.
    - GPS and GSM module: The A9 GSM/GPRS module is a tiny GSM modem that can be used in numerous IoT projects. In this research, the GSM was used to send the encrypted message to the CServer via MQTT. Additionally, the exact location of the vehicle is also determined through the GSM/GPRS and GPS tracking systems, which also use GPS technology. In this research, we proposed a GPS

technology system to capture the location. Additionally, the connection pins of the A9G GSM/GPRS+GPS were as follows (Table 5).

Table 5. A9G-TTL pin connection.

TTL	A9G
5V	USB
RX	TX1
TX	RX1

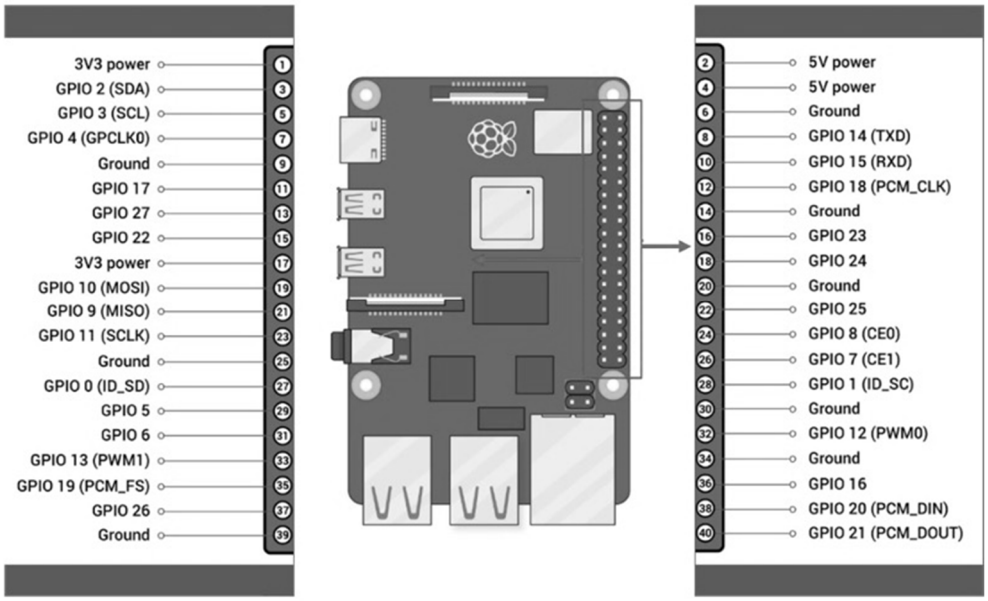

Figure 8. RPM pins.

4. Phase 4: Notifications

The Central Server (CServer) "Website-Database Server":

The Raspberry Pi CPU sends an encrypted message to the main server via MQTT when an accident is detected. Next, the CServer instantly decrypts the message and sends an SMS message containing the website URL to the first responders, informing them of the accident as a report so they may respond as soon as possible. In addition, the CServer is responsible for notifying any family members or the driver's family doctor that have been stored in the database. Additionally, the accident table in this phase will also be updated at the same time based on the accident information.

6. Evaluating the Phases (As Experimental Implementation)

1. Phase 1: QR Code Scanning (Data Entry) Implementation

The registration of vehicles is the goal of this phase. The owner of the vehicle must install the IoT device to get their vehicle ready for this system. The first step after setting up the device is for the owner to scan the QR code to complete the registration process and create the Vehicle ID, which will be kept in the database as shown in the following (Figure 9).

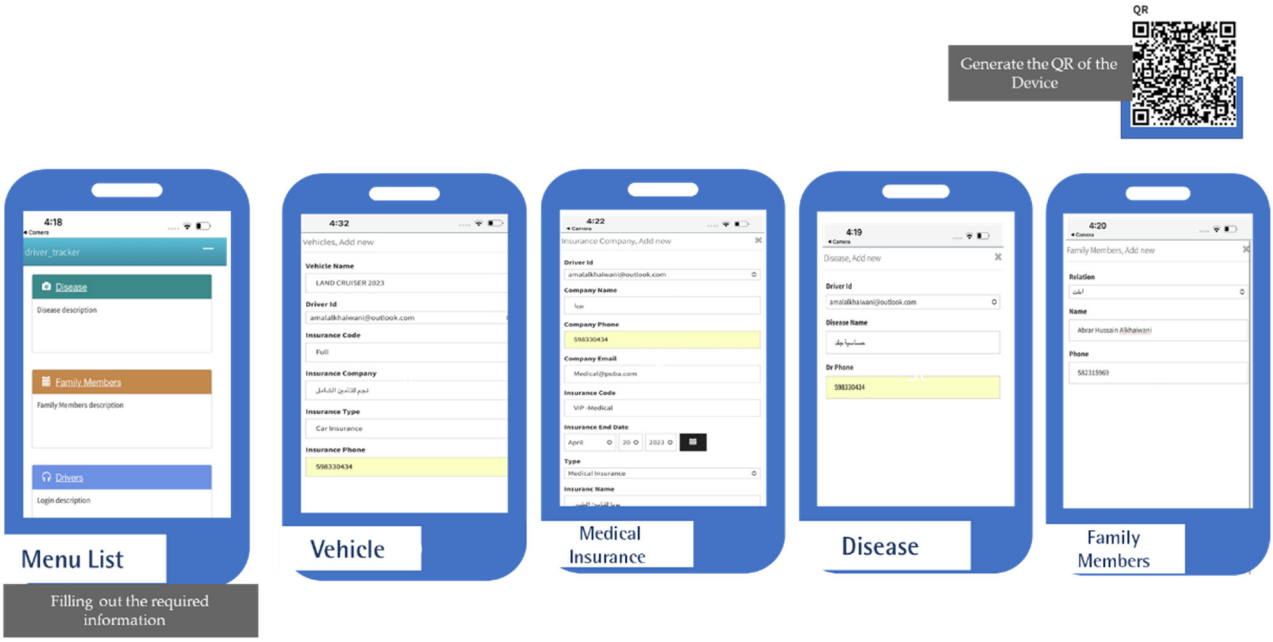

Figure 9. QR scanning and filling in the required information through the website registration.

2. Phase 2: IoT Accident Detection Sensor Implementation

Once an accident occurs, the AS and VS work together to promptly and instantly report the accident and send the signal to the RPM. The circuit is built by the RPM using a breadboard and wiring system. For optimal operation, each component needs to be connected to both GND (ground) and VCC (5 volts). The VS will send analogy signals after identifying the accident to the Raspberry Pi through a wire connected to pin #23, and the AS through a cable linked to pin #24 (Figure 10).

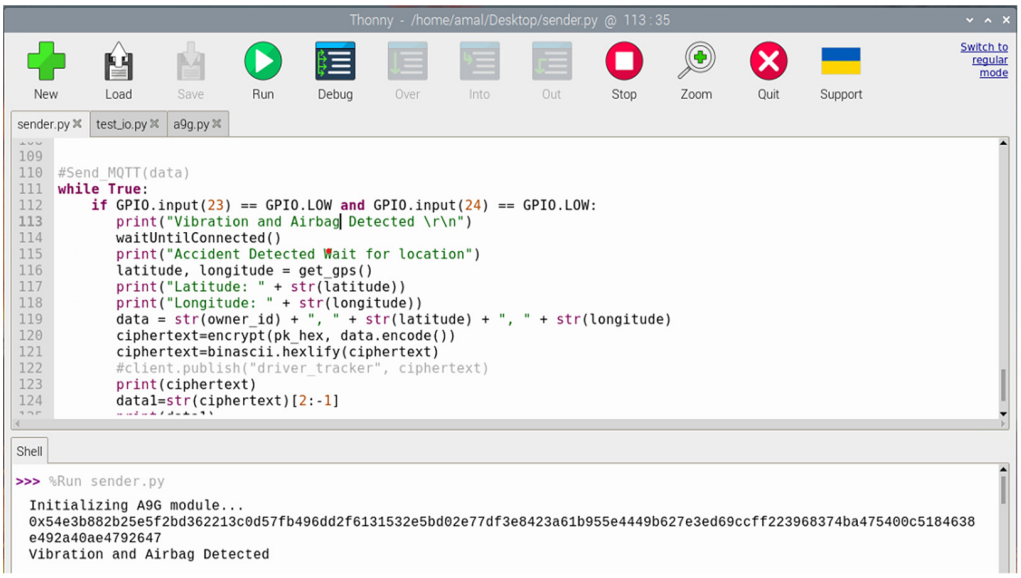

Figure 10. The AS and VS outputs are activated.

3. Phase 3: Accident Transmission Encryption Message

Next, the RPM sends a wire (pin #4) request to GPS to acquire the accident location. The GSM/GPRS USB model is connected to the GPS chip's RX (receiver) pin. The GPS will then save the location, providing its longitude and latitude, and send it back to the Raspberry Pi via the GSM/GPRS USB model that is linked to the TX (transmit) pin. In

addition, upon GSM module activation through the RPM, the encryption feature will be performed. The ECC model generates random values, creates private and public keys, and encrypts the message using the ecies.utils key-generating and encryption libraries. In brief, the ECIES algorithm is being used to implement the data encryption module (Figure 11). Thus, it will use the Message Queue Telemetry Transport (MQTT) Protocol to send the encrypted message to the CServer. Moreover, the encrypted message contains a unique Driver_ID and GPS coordinates.

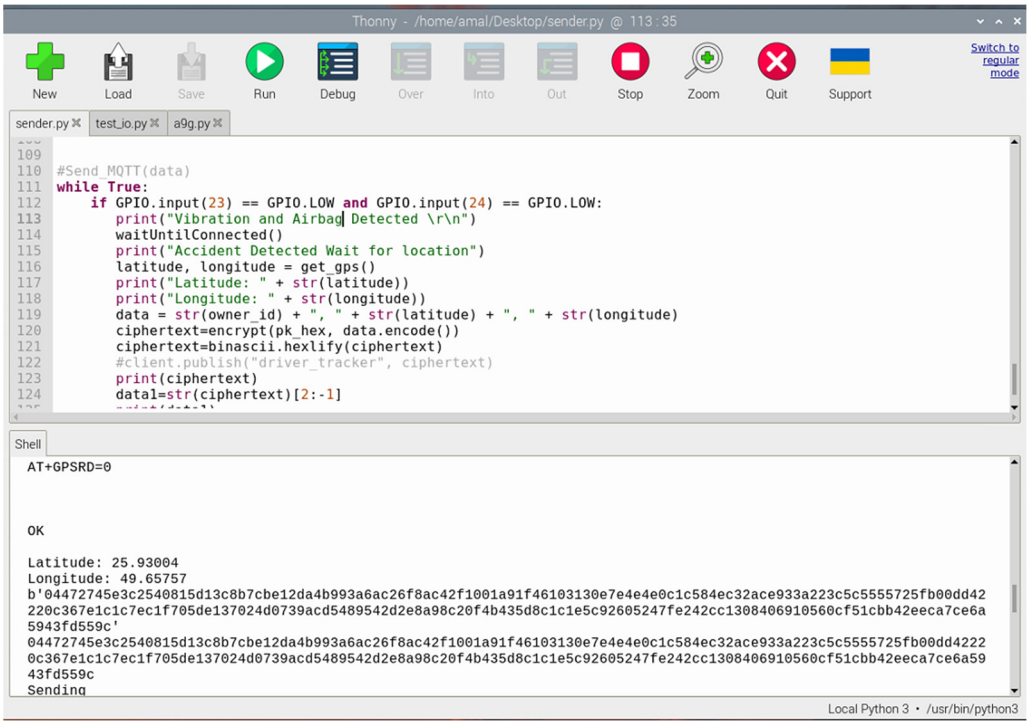

Figure 11. The GPS records the location (longitude and latitude) and encrypts the message.

4. Phase 4: Notifications to First Responders and Accident Details

The CServer receives an encrypted notification about the accident each time it occurs. Following this, CServer utilizes the private key to decrypt the message and displays the message's contents, including the accident's location on a map and the required information on the driver (Figure 12). Thus, the required information is then sent to the first responders via SMS messages, as shown in the figure below, via the CServer (Figure 13).

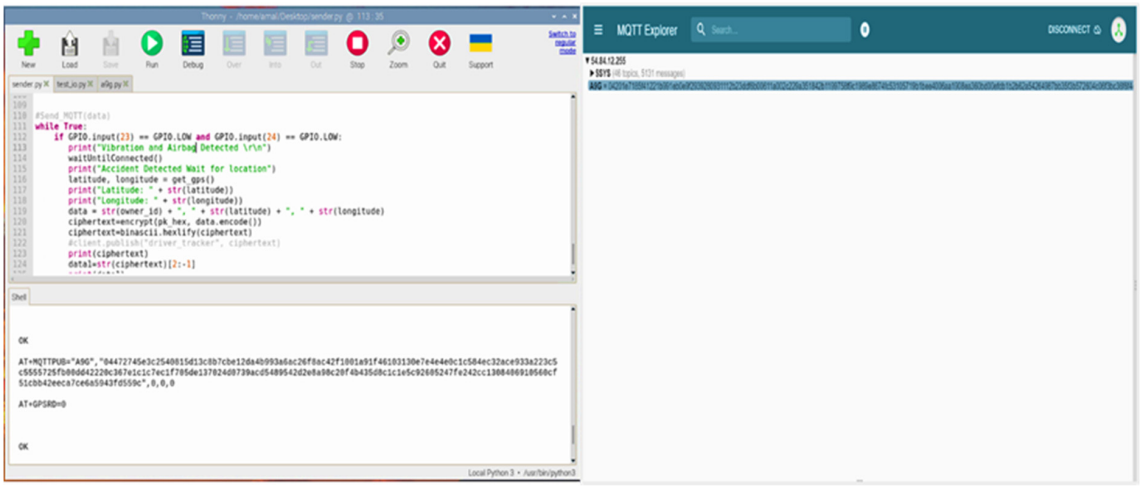

Figure 12. The encrypted message is sent through MQTT.

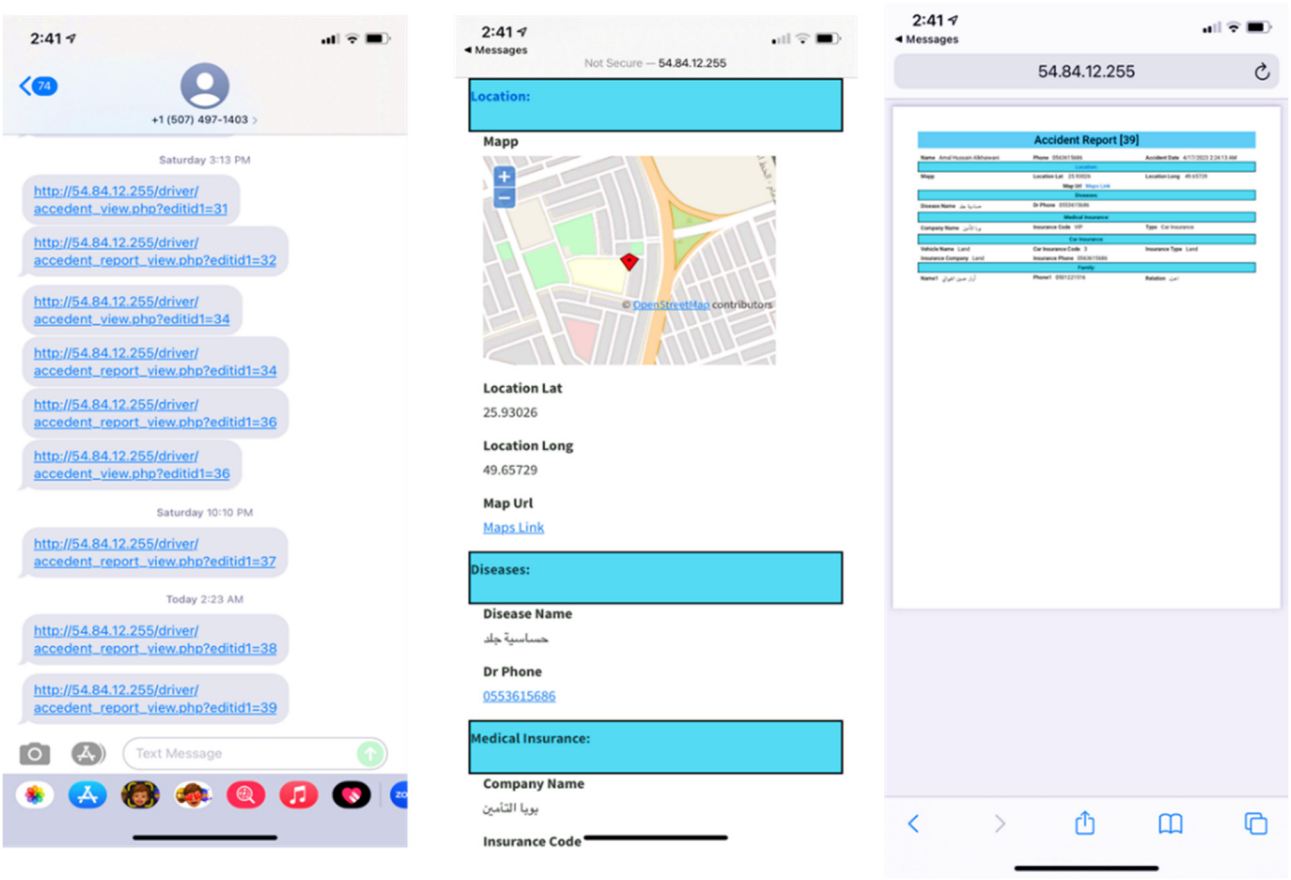

Figure 13. The CServer decrypts the message and sends the SMS message to the first responders.

The following (Figures 14 and 15) presents the complete artifact components connected to the IoT-based system for instant accident detection.

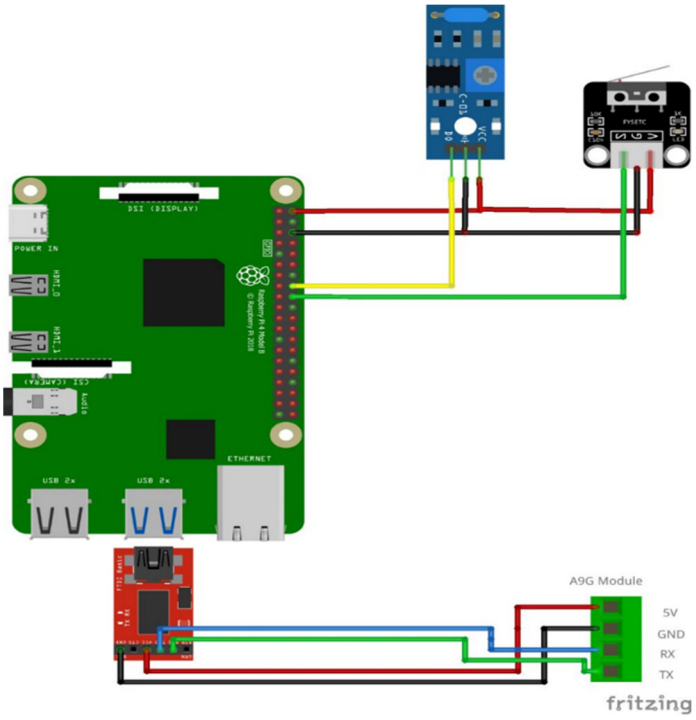

Figure 14. The complete circuit implementation with the CServer.

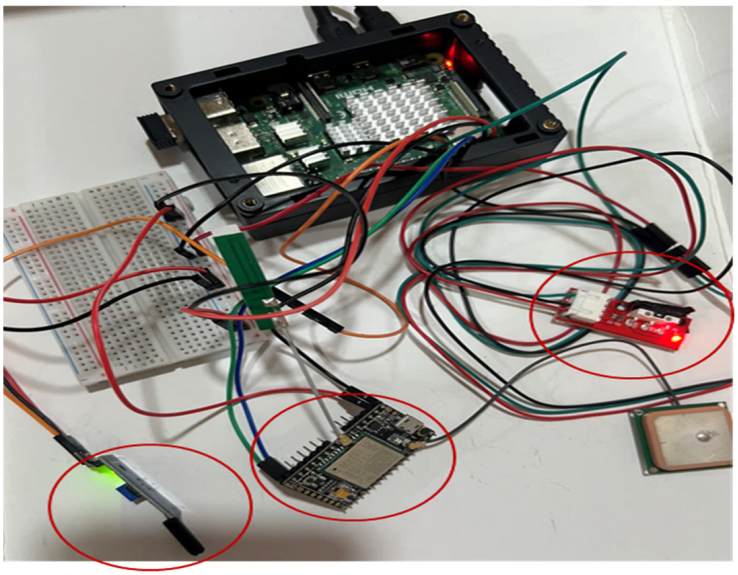

Figure 15. The complete artifact.

Finally, from the time the driver installs the IoT device on the vehicle until an accident occurs, all information related to the accident and the driver is stored in the database.

6.1. Overall Outcomes/Results

The result was instant car accident detection and reporting technology based on the IoT. In addition, first responders (including Najm and ambulances) were able to instantly locate the vehicle involved in the accident and provide medical help, saving their lives. Thus, the IoT-based accident detection system assists in locating the precise location of the accident and relaying this information to the appropriate first responders so that the injured person may obtain assistance as soon as possible.

The system's functionality was confirmed in the preliminary findings, as described in this section. Table 6 shows the expected and actual values of the functions performed by the proposed artifact.

Table 6. Expected and actual functions of the artifact.

Functional Requirement	Test Conducted	Expected	Actual
QR scanning	The QR barcode is read, and the driver's information is entered.	The driver can complete all fields with the required information.	The QR is generated by the system. Then, the driver can complete the required information. When the recorded information is complete, the device's unique ID is created, which can be used during an accident.
AS (detection)	The closing of the airbag's switch is detected.	The airbag communicates with the microprocessor once the airbag switch has been closed or depressed (i.e., blown).	When the airbag button is pressed, the airbag light turns on and sends a signal to the microprocessor to let it know that it has done so.
VS (detection)	Once any pressure or shaking is detected.	The VS should send the signal to the microprocessor by delivering a signal as soon as it moves.	The microprocessor is quickly informed that a vibration just occurred when shaking the VS.

Table 6. *Cont.*

Functional Requirement	Test Conducted	Expected	Actual
Locating the position of the vehicle or accident (location recording)	The GPS responds to the signals it receives from the microprocessor.	Once the GPS receives the signals from the microprocessor, it should locate the current position of the accident.	The GPS locates the current position when it receives a request from the microprocessor.
Encrypting the message using the ECIES algorithm	Sending the location through the GPS to the microprocessor for application of the ECIES algorithm.	The ECIES algorithm should be utilized by the microprocessor to apply the encryption message through the ECIES utilities library, and the microprocessor should be able to identify the device's unique ID and GPS location through the encryption message.	GPS provides the device's unique ID, which the microprocessor recognizes, together with the current location coordinates (longitude and latitude), which are used to encrypt the message.
GSM is operational and transmitting an encrypted message	Transmitting the encryption message using GSM and MQTT.	With MQTT, GSM transmits an encrypted message to the CServer.	The GSM sends the MQTT to the CServer after receiving the encrypted message.
Receiving the message and decrypting it into the Cserver	Decrypting the message and examining the accident details.	Using the database's private keys, which are created using the encryption procedure, the message is decrypted, and the accident details are displayed in full on the website's interface.	The accident's details are displayed on the website's interface, and the messages are decrypted.
CServer and website display the accident details on the web page	To see the driver's detailed information.	After decrypting, the required information about the accident is loaded on the site.	The required information about the accident is displayed on the website's interface.

As briefly stated in the table, the major features were verified, and it was ensured that the flow of all microprocessor system units (such as sensors, GSM, GPS, database server, and ECIES IoT encryption algorithm library) is successful and efficient.

6.2. Case Study and System Limitations

The following (Table 7) provides case studies that were tested during the proposed system's implementation. In general, human-designed systems can provide accurate readings up to a particular threshold, but beyond that point, they will have certain limitations. The table of work in this study describes the sensors' limitations and their intended use in combination to obtain an accurate value. In addition, the following two situations contrast the size of the keys and the time consumed with the RSA and ECC during the encryption process.

Table 7. Case studies tested during the proposed system's implementation.

Case Number	Case	Sensor Test	Output/Result
Case 1	In this case, consider the extracted real vibration value with a system that only has one sensor.	The VS rate is 2200 Hz	Accidents that happen at vibrations lower than this one are not detectable by this system because the detectable vibration is up to 4000 Hz.
Case 2	In this case, consider the extracted real vibration value with a system that only has one sensor.	The VS rate is 4000 Hz	The vibration is activated. The mechanism is activated. However, the accuracy of one sensor's operation is not enough, and a notification is sent even if it turns out to not have been an accident.

Table 7. *Cont.*

Case Number	Case	Sensor Test	Output/Result
Case 3	In this case, consider the possibility of the AS being activated either as a result of an accident or as a result of manufacturing defects.	The AS bursts	The airbag bursts. The system is triggered in any case, and a message is sent.
Case 4	In this case, consider the possibility of two sensors being triggered simultaneously.	The VS rate is 4000 Hz, and the airbag bursts	When the first and second sensors (VS and AS) are activated (hertz value), an accident is recognized (triggered).
Case 5	In this case, the accident is detected by the sensors, and a message is encrypted using the RSA public/private key size (30) via the ECIES algorithm.	The size of the RSA key is 3072	The private and public keys are generated with RSA, and the key size is determined using a time-consuming process.
Case 6	In this case, the accident is detected by the sensors, and a message is encrypted using the ECC public/private key size (256) via the ECIES algorithm.	ECC key size is 256	The private and public keys are generated faster than the RSA key size.

Cases 5 and 6 used an experimental test that compared ECIES and RSA, utilizing the Python, ECIES, and RSA libraries. The National Institute of Standards and Technology (NIST) provided the security strengths of the symmetrical block cipher and asymmetric-key algorithms, shown in (Table 8).

Table 8. Comparable key lengths for ECIES and RSA in bits [40].

Security Level	ECIES Key Length	RSA Key Length
80	160	1024
112	224	2048
128	256	3072
192	384	7680
256	512	15,360

As a result, the test functions were analyzed using the ECIES Python Library, which utilizes the secp256k1 curve, which implies that the ECC key size is 256 at the security level of 128. In light of this, 3072 is the RSA key size presented in the ECC domain parameters with recommended properties [41]. In addition, Table 9 explains the computational time required to generate the ECIES and RSA's public and private keys and to perform the encryption and decryption operations. Furthermore, it was presumed that messages in plaintext, including the location of the Device ID, including longitude and latitude, were sent. The researcher's location was utilized in this case in the following format: b'ID:7865409, LAT:25.933333, LOG:49.666667.

Table 9. Comparison between ECIES and RSA.

Key Size Length		Key Generation Running Time (Sec)		Encryption Process Running Time (Sec)		Decryption Process Running Time (Sec)	
ECC	RSA	ECC	RSA	ECC	RSA	ECC	RSA
256	3072	0.0021	1.887	0.006	0.002	0.003	0.009

According to the comparison between ECIES and RSA, Table 9 shows that ECIES generates keys faster than the RSA algorithm. As the private key is created at random and

the public key is a point on a curve, ECIES has the benefit of being able to generate both keys in a short time. Moreover, the RSA algorithm encrypts data faster than an algorithm based on ECIES. However, it is noted that the difference between the two computational times is comparatively shorter than one second, and this is not a large difference. In addition, the ECC algorithm's decryption process is faster than the RSA process. As a result, the outcomes show that for systems based on ECC, thanks to the mathematical strategies and benefits provided by the curves, ECIES represents an effective substitute for RSA-based cryptosystems. Particularly since embedded systems lack the memory and processing power necessary to complete the calculations required by RSA-based cryptosystems with large numbers, ECIES cryptosystems are ideal for IoT-embedded systems.

Thus, to compare this suggested work to the prior studies, it must be noted that this system was created with privacy in mind and that the proper security mechanisms are in place to protect the data transmitted by the system, which has been overlooked in previous studies. Additionally, there are benefits to having two sensors to provide an accurate alert and a faster response time.

6.3. Comprehensive Remarks on the Overall Proposed System (Adding Knowledge Base)

Utilizing an IoT security framework to encrypt, identify accidents, and detect accidents is our proposed artifact development in this research. Furthermore, it is advantageous for developing a framework for IoT security that can be applied to other purposes. In addition, we have successfully finished the coding of the sensors, the CServer (database server), GPS, and GSM to complete the process of sensing the accident and informing about it. In addition, the procedure of collecting signals from the sensor, using GPS to determine the current location by providing the location's longitude and latitude, and encrypting them using the ECIES algorithm on the CPU. Following this, the MQTT-encrypted message is sent through GSM to the CServer. In addition, the CServer will decrypt the message after receiving it to deliver the accident information, which will then be shown on the website. The accident information will then be sent to the first responders via SMS messages as URL links through the CServer.

Last but not least, the proposed system device may be easily modified and will benefit society by enabling improved outcomes, such as accident detection, if it is operating as planned to deliver an intelligent IoT security framework. Preventing location monitoring and the numerous deaths that can occur as a result of a delayed notification process, preventing the people's injuries from growing worse as soon as possible when the first responders are informed about the accident in detail instantly.

7. Conclusions and Future Work

The number of vehicles has dramatically increased recently in countries such as Saudi Arabia. In addition, accident rates have increased as a result of the increased traffic. Many accident detection devices exist, but a significant number of fatalities still occur. This problem is caused, at least in part, by insufficient automatic accident detection, insufficient warning, and inefficient emergency response routing, which obstruct the proper response to catastrophic accidents. The lack of applicable technologies because of financial and capacity restrictions on retrofitting just makes the situation worse. This research provides an intelligent security framework to handle these problems, and an IoT-based security framework solution is suggested for accident detection instantly.

This research demonstrated that making use of a range of various sensors can improve the accuracy with which a traffic accident is detected while preserving the privacy of the driver's critical information. Thus, the proposed system detects the location of an accident instantly, encrypts the critical information about the accident, and then transmits that information to first responders so that they can save lives.

Indeed, the results showed that the automatic IoT-based security framework accident detection system, which also protects the privacy of the driver's personal information, performed as expected. The main advantage of this research is that it reduces the number of

false alarms about accident reports. Moreover, RSA-based cryptosystems can be effectively replaced by ECIES. ECIES cryptosystems are particularly well suited for IoT-embedded systems since such systems lack the memory and processing power needed to finish the calculations needed by RSA-based cryptosystems with different key sizes. Another advantage is that the artifact might be used as a model for the implementation of similar artifacts in other contexts related to applying an intelligent security framework and the Internet of Things (IoT). A similar artifact, such as one created for automatically predicting plantation location and watering, where data protection is required, might be produced using the prototype as a guide to meet various public or commercial objectives.

Furthermore, there were limitations to the research due to conducting the initial evaluation of the system in a simulated environment. One of the study's limitations is that the system currently cannot calculate the closest responders (e.g., Najm and ambulances) to the accident through the nearest first responders with the shortest response time, computed using the distance matrix API, allocated to the user. In addition, the driver of the first responders receives an SMS specifying the user's location. Further, the system currently does not allow the user to receive information about the probable arrival time.

In the future, there are a number of important recommendations, citing the data that were gathered, including the need for the system to calculate the distance to identify the first responders (for example, Najm and ambulances) who are nearest to the accident and to send an urgent SMS that contains the location to provide assistance instantly. Future advancements in this technology may involve attaching a camera module to take pictures of the accident and sending the pictures to a server. In the future, big data processing can be used to examine some road accident results using the data gathered on the server. Additionally, in the not-too-distant future, system security and privacy will be improved, and these challenges will be fully addressed in upcoming efforts, especially on the website and the General Data Protection Regulation (GDPR).

Author Contributions: Conceptualization, A.H.A.; Methodology, A.H.A. and B.S.A.; Validation, A.H.A. and B.S.A.; Formal analysis, A.H.A.; Investigation, A.H.A. and B.S.A.; Resources, A.H.A.; Writing—original draft, A.H.A.; Writing—review & editing, A.H.A. and B.S.A.; Supervision, A.H.A. and B.S.A.; Project administration, A.H.A. and B.S.A. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Almoshaogeh, M.; Abdulrehman, R.; Haider, H.; Alharbi, F.; Jamal, A.; Alarifi, S. Shafiquzzaman Traffic Accident Risk Assessment Framework for Qassim, Saudi Arabia: Evaluating the Impact of Speed Cameras. *Appl. Sci.* **2021**, *11*, 6682. [CrossRef]
2. Raghad, A.; Areej, S. IoT Based Accident Prevention System Using Machine Learning Techniques. Available online: https://www.researchgate.net/profile/Hala-Qudaih/publication/369595854_IoT_Based_Accident_Prevention_System_using_Machine_Learning_techniques/links/6424132192cfd54f8439c7bc/IoT-Based-Accident-Prevention-System-using-Machine-Learning-techniques.pdf (accessed on 1 April 2023).
3. Sharma, S.; Sebastian, S. IoT based car accident detection and notification algorithm for general road accidents. *Int. J. Electr. Comput. Eng.* **2019**, *9*, 4020. [CrossRef]
4. Sklavos, N.; Zaharakis, I.D. Cryptography and Security in Internet of Things (IoTs): Models, Schemes, and Implementations. In Proceedings of the 2016 8th IFIP International Conference on New Technologies, Mobility and Security (NTMS), Larnaca, Cyprus, 21–23 November 2016; pp. 1–2. [CrossRef]
5. Borgia, E. The Internet of Things vision: Key features, applications and open issues. *Comput. Commun.* **2014**, *54*, 1–31. [CrossRef]

6. Bhardwaj, I.; Kumar, A.; Bansal, M. A review on lightweight cryptography algorithms for data security and authentication in IoTs. In Proceedings of the 2017 4th International Conference on Signal Processing, Computing and Control (ISPCC), Solan, India, 21–23 September 2017; pp. 504–509. [CrossRef]
7. Fotovvat, A.; Rahman, G.M.E.; Vedaei, S.S.; Wahid, K.A. Comparative Performance Analysis of Lightweight Cryptography Algorithms for IoT Sensor Nodes. *IEEE Internet Things J.* **2020**, *8*, 8279–8290. [CrossRef]
8. Sridhar, S.; Smys, S. Intelligent security framework for iot devices cryptography based end-to-end security architecture. In Proceedings of the 2017 International Conference on Inventive Systems and Control (ICISC), Coimbatore, India, 19–20 January 2017; pp. 1–5. [CrossRef]
9. Bertino, E.; Islam, N. Botnets and Internet of Things Security. *Computer* **2017**, *50*, 76–79. [CrossRef]
10. Lee, S.; Tewolde, G.; Kwon, J. Design and implementation of vehicle tracking system using GPS/GSM/GPRS technology and smartphone application. In Proceedings of the 2014 IEEE World Forum on Internet of Things (WF-IoT), Seoul, Republic of Korea, 6–8 March 2014; pp. 353–358. [CrossRef]
11. Jethwa, A. Vehicle Tracking System Using Gps and Gsm Modem—A Review. *Int. J. Recent Sci. Res.* **2015**, *6*, 4805–4808.
12. Khan, M.A.; Khan, S.F. IoT based framework for Vehicle Over-speed detection. In Proceedings of the 2018 1st International Conference on Computer Applications & Information Security (ICCAIS), Riyadh, Saudi Arabia, 4–6 April 2018; pp. 1–4. [CrossRef]
13. Yee, T.H.; Lau, P.Y. Mobile vehicle crash detection system. In Proceedings of the 2018 International Workshop on Advanced Image Technology (IWAIT), Chiang Mai, Thailand, 7–9 January 2018; pp. 1–4. [CrossRef]
14. Nasr, E.; Kfoury, E.; Khoury, D. An IoT approach to vehicle accident detection, reporting, and navigation. In Proceedings of the 2016 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET), Beirut, Lebanon, 14–16 November 2016; pp. 231–236. [CrossRef]
15. Khot, I.; Jadhav, M.; Desai, A.; Bangar, V. Go Safe: Android application for accident detection and notification. *Int. Res. J. Eng. Technol.* **2018**, *5*, 4118–4122.
16. Chaturvedi, N.; Srivastava, P. Automatic Vehicle Accident Detection and Messaging System Using GSM and GPS Modem. *Int. Res. J. Eng. Technol.* **2018**, *5*, 252–254.
17. Khaliq, K.A.; Raza, S.M.; Chughtai, O.; Qayyum, A.; Pannek, J. Experimental validation of an accident detection and management application in vehicular environment. *Comput. Electr. Eng.* **2018**, *71*, 137–150. [CrossRef]
18. Cauteruccio, F.; Cinelli, L.; Corradini, E.; Terracina, G.; Ursino, D.; Virgili, L.; Savaglio, C.; Liotta, A.; Fortino, G. A framework for anomaly detection and classification in Multiple IoT scenarios. *Futur. Gener. Comput. Syst.* **2021**, *114*, 322–335. [CrossRef]
19. OWASP Internet of Things Top Ten Project. The Open Web Application Security Project®. 2014. Available online: https://owasp.org/www-project-internet-of-things-top-10/#tab=OWASP_Internet_of_Things_Top_10_for_2014 (accessed on 5 December 2022).
20. Aleisa, N.; Renaud, K. Privacy of the Internet of Things: A Systematic Literature Review. In Proceedings of the 50th Hawaii International Conference on System Sciences, Waikoloa Village, HI, USA, 4–7 January 2017. [CrossRef]
21. Henriques, T.A.; O'Neill, H. Design science research with focus groups—A pragmatic meta-model. *Int. J. Manag. Proj. Bus.* **2023**, *16*, 119–140. [CrossRef]
22. Dhanda, S.S.; Singh, B.; Jindal, P. Lightweight Cryptography: A Solution to Secure IoT. *Wirel. Pers. Commun.* **2020**, *112*, 1947–1980. [CrossRef]
23. Keerthi, K.; Surendiran, B. Elliptic curve cryptography for secured text encryption. In Proceedings of the 2017 International Conference on Circuit, Power and Computing Technologies (ICCPCT), Kollam, India, 20–21 April 2017; pp. 1–5. [CrossRef]
24. Hevner, A.; Chatterjee, S. Design Science Research in Information Systems. In *Design Research in Information Systems*; Integrated Series in Information Systems; Springer: Boston, MA, USA, 2010; Volume 22, pp. 9–22. [CrossRef]
25. Hevner, A.R. A Three Cycle View of Design Science Research. *Scand. J. Inf. Syst.* **2007**, *19*, 4.
26. Hevner, A.R.; March, S.T.; Park, J.; Ram, S. Design Science in Information Systems Research. *Manag. Inf. Syst. Q.* **2004**, *28*, 75–105. [CrossRef]
27. Imam, R.; Areeb, Q.M.; Alturki, A.; Anwer, F. Systematic and Critical Review of RSA Based Public Key Cryptographic Schemes: Past and Present Status. *IEEE Access* **2021**, *9*, 155949–155976. [CrossRef]
28. Alphonse, R.M.; Malalatiana, R.H.; Choube, M.P. Segment optimization of short message service in telecommunication. *Innov. Technol. Methodical Res. J.* **2021**, *2*, 81–88. [CrossRef]
29. Nayak, M.; Dash, P. Smart surveillance monitoring system using raspberry PI and PIR sensor. *Indian J. Text. Res.* **2018**, *7*, 493–495.
30. Saabith, S.; Thangarajah, V.; Fareez, M. A Review on Python Libraries and IDEs for Data Science. *Int. J. Res. Eng. Sci.* **2021**, *9*, 36–53.
31. Kurniawan, D.E.; Iqbal, M.; Friadi, J.; Borman, R.I.; Rinaldi, R. Smart Monitoring Temperature and Humidity of the Room Server Using Raspberry Pi and Whatsapp Notifications. *J. Phys. Conf. Ser.* **2019**, *1351*, 012006. [CrossRef]
32. Dinculeană, D.; Cheng, X. Vulnerabilities and Limitations of MQTT Protocol Used between IoT Devices. *Appl. Sci.* **2019**, *9*, 848. [CrossRef]
33. Kryvonos, O.; Strutynska, O.; Kryvonos, M. The use of visual electronic circuits modelling and designing software fritzing in the educational process. *Zhytomyr Ivan Franko State Univ. Jo. Pedagogical Sci.* **2022**, *1*, 198–208. [CrossRef]
34. Jacobsen, R.H.; Aliu, D.; Ebeid, E. A Low-cost Vehicle Tracking Platform using Secure SMS. In Proceedings of the 2nd International Conference on Internet of Things, Big Data and Security, Porto, Portugal, 24–26 April 2017; SCITEPRESS—Science and Technology Publications: Porto, Portugal, 2017; pp. 157–166. [CrossRef]

35. Alkhudhayr, F.; Moulahi, T.; Alabdulatif, A. Evaluation Study of Elliptic Curve Cryptography Scalar Multiplication on Raspberry Pi4. *Int. J. Adv. Comput. Sci. Appl.* **2021**, *12*, 472–479. [CrossRef]
36. Narayanan, K.L.; Ram, C.R.S.; Subramanian, M.; Krishnan, R.S.; Robinson, Y.H. IoT based Smart Accident Detection & Insurance Claiming System. In Proceedings of the 2021 Third International Conference on Intelligent Communication Technologies and Virtual Mobile Networks (ICICV), Tirunelveli, India, 4–6 February 2021; pp. 306–311. [CrossRef]
37. Gautam, R.; Choudhary, S.; Surbhi Kaur, I.; Bhusry, M. Cloud based automatic accident detection and vehicle management. In Proceedings of the 2nd International Conference on Science Technology and Management, New Delhi, India, 27 September 2015; pp. 341–352.
38. Jebril, N.A.; Al-Haija, Q.A.; AlBarrak, N.; Almutlaq, G.; Alkhaiwani, A.H. Complete Microcontroller Based Vehicle Accident Detection System with Case Study for Saudi Arabia. *Wseas Trans. Commun. Arch.* **2017**, *16*, 118–130.
39. Ghanem, S.; Ghanim, S. Design and Implementation of an Integrated Vehicle Security System (IVSS). *Int. J. Ind. Sustain. Dev.* **2022**, *3*, 87–97. [CrossRef]
40. Barker, E. *Recommendation for Key Management: Part 1—General*; National Institute of Standards and Technology: Gaithersburg, MD, USA, 2020; NIST SP 800-57pt1r5. [CrossRef]
41. Brown, D. Sec 2: Recommended Elliptic Curve Domain Parameters. secg.org. 2010. Available online: https://www.secg.org/sec2-v2.pdf (accessed on 2 November 2022).

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.