IoT BASED SURVEILLANCE MODEL FOR MONITORING SCHOOL CHILDREN

IRVIN KIPLAGAT KILOT

A Thesis Submitted to the Institute of Postgraduate Studies of Kabarak University in Partial Fulfillment of the Requirements for the Award of Master of Science in Information Technology Security and Audit Degree

KABARAK UNIVERSITY

NOVEMBER, 2024

DECLARATION

1. I do hereby declare that:

   i. This proposal/project/thesis is my own work and to the best of my knowledge it has not been presented for the award of a degree in any university or college.

   ii. The work has not been incorporated material from other works or a paraphrase of such material without due and appropriate acknowledgement

   iii. The work has been subjected to processes of anti-plagiarism and has met Kabarak University 15% similarity index threshold

2. I do understand that issues of academic integrity are paramount and therefore I may be suspended or expelled from the University or my degree may be recalled for academic dishonesty or any other related academic malpractices.

Signed:_________________________________ Date:_____________________

Irvin Kiplagat Kilot

GMIA/NE/1957/09/17

RECOMMENDATION

To the Institute of Postgraduate Studies:

The thesis entitled **"IoT Based Surveillance Model for Monitoring School Children"** written by **Irvin Kiplagat Kilot** is presented to the Institute of Postgraduate Studies of Kabarak University. We have reviewed the thesis and recommend it be accepted in partial fulfilment of the requirement for award of the degree of Master of Science in Information Technology and Security Audit.

Signed:____________________________ Date:____________________

Dr. Nelson Masese

School of Science Engineering and Technology

Kabarak University

Signed:____________________________ Date:____________________

Prof. Simon Karume

School of Science Engineering and Technology

Kabarak University

COPYRIGHT

ACKNOWLEDGMENTS

I want to begin by expressing my deepest gratitude to God for the gift of life and provision, without which I would not have reached this point in my academic journey. All praise, honor, and glory are Yours. I also wish to sincerely thank my supervisors, Prof. Simon Karume and Dr. Nelson Masese, for their unwavering support and guidance throughout my studies. Their patience, inspiration, vast knowledge, wise counsel, and invaluable contributions have been instrumental in the completion of this thesis. I am equally grateful to everyone I have had the privilege of working with during this and other academic endeavors. Special thanks to my classmate, Elvin Saikwa, for his insightful feedback, encouragement, and thought-provoking questions that pushed me to expand my research from various angles. My heartfelt appreciation goes to my parents, whose love, support, and guidance have been a constant source of strength. They are my greatest role models. Lastly, I extend my sincere thanks to the entire Kabarak University for providing me with the opportunity to pursue my studies and for the support they offered throughout my academic journey.

DEDICATION

I dedicate this research thesis to my parents, whose unconditional love and exemplary guidance have instilled in me the value of hard work and perseverance. They have shown me that even the most challenging tasks can be completed by taking one step at a time. I also extend this dedication to everyone who may find this work beneficial.

ABSTRACT

Children are naturally prone to accidents, exploitation, and various forms of abuse due to their inherent vulnerability, reliance on caregivers, and limited understanding of potential hazards and self-defense. Recent media coverage has highlighted an alarming rise in crimes targeting children, including abductions and murders, intensifying concerns about child safety. As children spend most of their time in school, ensuring their continuous monitoring and care becomes a shared responsibility of both the school and parents. The current method of accounting for children in Kenyan schools, involving physical head counts and class list checks, is long, tiresome, and inefficient, making it difficult to track learners' whereabouts throughout the day. To address these challenges, this research aims to develop an IoT-based surveillance system to monitor school children, leveraging technologies like Global Positioning System (GPS), Radio-Frequency Identification (RFID), Ultra-Wide Band (UWB), Bluetooth Low Energy (BLE), Infrared, Wi-Fi, and Zigbee. Collectively known as the Internet of Things (IoT), these technologies offer opportunities to enhance learner safety by enabling real-time monitoring of student movements on school premises. The study focuses on designing, developing, and evaluating an IoT-based surveillance model tailored to the needs of Kenyan schools. The model includes key components such as student registration, device registration, real-time tracking, and alert notifications. The research findings demonstrate that the developed model reliably monitors student movements in real-time, significantly enhancing school safety and addressing parental concerns about their children's whereabouts. Despite its effectiveness, challenges related to network reliability and data privacy were identified. The study concludes that IoT technology, when implemented with robust security protocols, can substantially improve child monitoring in educational settings. The research contributes to the field by offering a practical and scalable solution for real-time surveillance, emphasizing the importance of privacy safeguards. Recommendations include schools adopting IoT surveillance models to enhance student safety and governments creating supportive policies to facilitate the integration of IoT technology in schools while prioritizing data security.

Keywords: *Child Safety, School Children, Surveillance, Internet of Things (IoT), Real-Time Tracking*

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

OPERATIONAL DEFINITION OF TERMS

Internet of Things (IoT: In this project, IoT refers to a network of interconnected devices, such as GPS, RFID, UWB, Infrared, Wi-Fi, and Zigbee sensors, used for monitoring and surveillance. These devices exchange data in real-time to track and report the movements and status of students.

Surveillance Model: A technological framework that integrates IoT devices and sensors, such as GPS and RFID, to monitor and track student location, attendance, and activities in real-time, enhancing school safety and operational efficiency.

Global Positioning System (GPS: A satellite-based system used in this project to determine the exact real-time location of students equipped with GPS-enabled devices.

Radio Frequency Identification (RFID): A wireless communication technology used for automatically identifying and tracking tags attached to student ID cards or wearables to monitor attendance and location.

Sensors: Devices embedded in the surveillance model to detect and measure parameters like motion and proximity, providing critical data for monitoring students within school premises.

Data Privacy: Measures and protocols implemented to ensure the security of student information collected by the surveillance system, preventing unauthorized access or breaches.

Encryption: The process of converting student data into a secure format that can only be accessed or read by authorized users, safeguarding the confidentiality of sensitive information.

Real-time Monitoring: The continuous, immediate tracking and reporting of student movements and attendance, enabling school administrators and parents to receive updates instantly.

Alert Notification: Automated messages or alarms triggered by the surveillance system when predefined conditions, such as unauthorized movement or absence, are detected, prompting timely action.

Role-based Access Control: A security mechanism that restricts system access to authorized users based on their roles, ensuring that only designated personnel, like school administrators, have the appropriate level of data access.

LIST OF ABBREVIATIONS

ACID	Atomicity, Consistency, Isolation, Durability
AES	Advanced Encryption Standard
AI	Artificial Intelligence
AIEKF	Adaptive Iterative Extended Kalman Filter
AKF	Adaptive Kalman Filter
API	Application Programming Interface
BMS	Battery Management System
COVID	Coronavirus Disease
CPU	Central Processing Unit
CRUD	Create, Read, Update, Delete
DB	Database
DSRM	Design Science Research Methodology
ECC	Elliptic Curve Cryptography
EKF	Extended Kalman Filter
GLONASS	Global Navigation Satellite System (Russia)
GNSS	Global Navigation Satellite System
GPRS	General Packet Radio Service
GPS	Global Positioning System
GPSV	Global Positioning System Velocity
GSM	Global System for Mobile Communications
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
ICT	Information and Communications Technology
IDE	Integrated Development Environment
IP	Internet Protocol
JSON	JavaScript Object Notation
JWT	JSON Web Token
KB	Kilobyte
KF	Kalman Filter
KMPS	Kilometers Per Second
MB	Megabyte
MCU	Microcontroller Unit

MFA	Multi-Factor Authentication
MITM	Man-in-the-Middle Attack
MQTT	Message Queuing Telemetry Transport
NFR	Non-Functional Requirement
OLED	Organic Light-Emitting Diode
PF	Particle Filter
PII	Personally Identifiable Information
RAM	Random Access Memory
RBAC	Role-Based Access Control
REST	Representational State Transfer
RMSE	Root Mean Square Error
RX	Receiver
SIM	Subscriber Identity Module
SMS	Short Message Service
SQL	Structured Query Language
SSD	Solid-State Drive
SSL	Secure Sockets Layer
TDMA	Time Division Multiple Access
TLS	Transport Layer Security
TX	Transmitter
UART	Universal Asynchronous Receiver-Transmitter
UHF	Ultra High Frequency
UHIF	Ultra High-Frequency Integrated Filter
UI	User Interface
VPS	Virtual Private Server
VTL	Virtual Tape Library
WAN	Wide Area Network

CHAPTER ONE

INTRODUCTION

1.1 Introduction

In this chapter, the research study was introduced, encompassing the presentation of the problem statement, research objectives, the significance of the study, and the study's scope. In the background discussion, the problem that the study intended to resolve was examined from a global and regional perspective. It then concluded with an exploration of the problem within a local context.

1.2 Background to the Study

The Internet of Things (IoT) refers to a network of interconnected devices that collect, share, and act on data through the internet without human intervention. These devices, embedded with sensors and processors, communicate seamlessly across various platforms, enabling automated systems. The concept, initially confined to industrial and logistical applications, has rapidly evolved, expanding into sectors such as healthcare, agriculture, and urban planning, as more technologies integrate with IoT networks (Bhatti, Khan & Kim, 2022).

The growth of IoT has been fueled by advancements in wireless communication, cloud computing, and the miniaturization of hardware. Predictions estimate that the number of connected devices could surpass 80 billion by 2025. This explosion is driven by the increasing demand for real-time data, automation, and smart solutions across industries (Khan & Bhatti 2023). Today, IoT plays a pivotal role in systems like smart homes, transportation networks, and industrial automation, contributing to the emergence of smart cities and Industry 4.0 initiatives (Bhatti *et al.*, 2022; Khan *et al.*, 2023).

The wide-ranging applications of IoT reflect its versatility. For example, smart agriculture leverages IoT for monitoring environmental conditions, while healthcare uses wearable devices to track patients' vitals in real-time. Urban areas benefit from IoT in the form of connected infrastructure that manages traffic, energy use, and public safety efficiently. Despite its rapid adoption, IoT systems face challenges, such as security vulnerabilities and interoperability issues, that need to be addressed for sustainable growth (Bhatti *et al.*, 2022; Khan *et al.*, 2023).

Schools face significant challenges in ensuring student safety due to various risks, including accidents, health emergencies, and environmental hazards. Children are especially vulnerable to injuries on school premises due to playground accidents, sports activities, and traffic-related incidents, such as when crossing streets near schools. Research highlights that external risks like vehicular accidents and crime remain concerns in urban school environments, posing threats to students' well-being both inside and outside school premises (Costello & Naimy, 2019).

Monitoring children's attendance and preventing chronic absenteeism is another challenge. Students who frequently miss school face higher risks of behavioral issues, academic underperformance, and mental health problems over time. Schools must implement targeted interventions such as enhanced transportation access, structured morning routines, and parental engagement strategies to mitigate these risks (Gottfried et al, 2023). However, efforts to reduce absenteeism are often inconsistent, varying greatly in effectiveness across different regions and schools.

The use of IoT in child monitoring models has gained traction, driven by the need for enhanced student safety during school activities and transportation. These models rely on a combination of GPS, RFID, GSM modules, and cloud-based platforms to monitor children's whereabouts in real time and inform parents or school authorities when necessary. Several studies indicate that tracking technologies are effective in reducing parental anxiety and preventing incidents like missed pickups or unauthorized absences (Al-Mazloum, Omer & Abdullah, 2023). Such systems typically involve wearable devices or identification cards equipped with sensors, enabling real-time location tracking and emergency alerts when a child is detected outside designated safe zones (Priya *et al.*, 2023).

Another significant application is in transportation safety, where IoT systems monitor children's transit on school buses. The integration of RFID and GPS in such setups ensures that children board the correct bus, with notifications sent to parents or guardians as they enter or exit (Ranga Rao & Anjaiah, 2023). These models not only facilitate attendance tracking but also prevent instances of children getting lost or left unattended. Some models also include biometric sensors for additional safety checks, such as verifying a child’s identity or health status while traveling (Gowri, Abirami & Monisha, 2023).

Despite their benefits, IoT-based child monitoring models face certain challenges. Privacy concerns remain paramount, as continuous tracking raises ethical questions about surveillance. IoT devices often handle sensitive information, such as the real-time location of children, which makes them targets for cyberattacks. Unauthorized access can compromise personal data, raising concerns about the compliance of these models with privacy regulations such as GDPR (General Data Protection Regulation) (Bentotahewa, Yousif, Hewage, Nawaf & Williams, 2022). Security measures, including encryption and user authentication protocols, are essential but can be complex and costly to implement effectively.

Another challenge involves the reliability and accuracy of IoT models. Issues such as network latency, device malfunctions, and GPS signal loss can affect model performance. In school environments, obstacles like building structures or environmental conditions may interfere with communication between devices, leading to gaps in real-time monitoring. Addressing these challenges requires continuous maintenance and redundancy measures to ensure uninterrupted service (Shafique, Khawaja, Sabir, Qazi & Mustaqim, 2020).

Furthermore, the cost of deploying and maintaining IoT-based models is a major consideration. Schools often operate under tight budgets, making it difficult to invest in sophisticated IoT infrastructures. Costs are not limited to initial installation; ongoing expenses include data subscriptions, hardware replacements, and software updates. These financial challenges must be balanced with the benefits of the model to encourage adoption in educational settings (Bentotahewa *et al.*, 2022; Shafique *et al.*, 2020).

Recent studies have explored the use of IoT in child monitoring, focusing on wearable devices and location tracking solutions. IoT-based models provide advantages such as automated attendance, route tracking for school buses, and emergency alerts. However, research highlights issues with the scalability of these solutions, as many models struggle to manage increasing data loads and users effectively, making it challenging to adapt them to larger school environments (Rahman Alam, & Rahman, 2021).

Another critical finding from existing research is the limited integration of IoT models with school management platforms. Many current solutions operate as standalone models, making it difficult to synchronize their data with existing student management databases. This lack of integration can reduce the effectiveness of IoT-based monitoring, as educators may not have seamless access to the information they need (Ghasempour, 2019).

In addition to these technological gaps, existing literature also emphasizes the need for user-friendly interfaces. Many models are complex to configure and maintain, discouraging adoption by school administrators and parents. Research recommends focusing on the development of simpler interfaces and better user training to improve model usability (Rahman *et al.*, 2021; Ghasempour, 2019).

Schools face significant challenges in ensuring student safety, especially in densely populated environments where traditional supervision methods may fall short. Concerns such as unauthorized entry, bullying, and emergencies require more advanced solutions. IoT-based surveillance offers a viable alternative, allowing schools to leverage real-time data collection from sensors and cameras to monitor student activities continuously. The integration of such technologies can enhance situational awareness, enabling staff to respond quickly to potential threats or emergencies (Damaševičius *et al.*, 2023; Zhang, Zhang & Liu, 2023).

Automated alerts and real-time tracking are particularly critical in improving safety measures in schools. These models allow for instant notification of incidents, reducing response times and minimizing the risks to students. For example, geofencing and GPS-enabled tracking can notify staff if a student leaves a designated area, prompting immediate action. By providing precise location data and real-time status updates, IoT-based models ensure that safety protocols are more efficiently managed, ultimately creating a more secure learning environment (Damaševičius *et al.*, 2023; Cheung, Kwok & Phusavat, 2022).

The identified challenges in student safety underscore the need for developing a comprehensive IoT-based surveillance model. The background illustrates how real-time monitoring and automated alerts can address safety concerns, leading directly to the research objectives of the proposed model. The model aims to enhance supervision and safety through modules such as user registration, real-time tracking, attendance management, and alert notifications, addressing the gaps left by traditional methods (Damaševičius *et al.*, 2023).

The study's justification for developing a specialized child monitoring model was based on findings from a pilot study (included in this thesis) conducted by the researcher in Kenyan schools in Nakuru. The study revealed that many schools lacked automated models for monitoring students and instead relied on manual processes. Although a few schools had implemented RFID technology for attendance and tracking, its adoption was not widespread and often lacked customization to address specific needs, such as comprehensive child surveillance (Sivarathinabala, Chitra, Sivanesan, Sundar, & Udhaya Kumar, 2019).

Additionally, there was a notable scarcity of empirical research that examined the strengths and limitations of various surveillance technologies within the school context, particularly in the Kenyan education system. This gap highlighted the need for more targeted studies that analyzed how these technologies performed in real-world scenarios, with a focus on cost-effectiveness. To address these gaps, this study aimed to develop a tailored IoT-based surveillance application that catered specifically to the needs of both public and private schools in Kenya. It also intended to provide empirical insights into the performance of existing technologies, offering a valuable foundation for future research and further development of surveillance models in educational settings.

1.3 Problem Statement

The high rate of child kidnappings and crimes has caused growing concern among parents, particularly regarding their children's safety during school hours. In several instances, schools have faced legal action from parents over cases of missing children. A recurring problem is that someday scholar students report to school but then sneak out unnoticed, while others fail to reach the school premises altogether. Ensuring student safety and accountability is a core responsibility of schools, which necessitates an efficient system to monitor and verify student presence. Currently, traditional methods such as physical headcounts and roll calls are lengthy, labor-intensive, and prone to errors, while existing models, like those using RFID tags, lack the capability to efficiently track student movements or detect unauthorized exits.

To address these challenges, there is a clear need for a solution that not only monitors students within school boundaries but also prevents unauthorized exits, ensuring students remain present and accounted for throughout the school day. This study proposes an IoT-based surveillance model that enhances student safety by automatically tracking their real-time location and generating alerts if a student attempts to leave the school premises without authorization. The implementation of this model aims to improve in-school attendance, deter unauthorized exits, and alleviate parents' concerns by providing a more secure and reliable means of monitoring students.

1.4 General Objective of the Study

The main objective of this study was to develop an IoT-based surveillance model specifically designed to monitor school children in Kenyan schools, addressing the unique challenges of student safety, attendance tracking, and unauthorized exits. The study aims to provide a solution that enhances in-school safety practices and informs policy recommendations for integrating advanced surveillance technology into educational institutions.

1.4.1 Specific Objectives of the Study

i. To analyze the weaknesses of existing IoT-based surveillance models for monitoring children through a systematic literature review.

ii. To design a proof of concept of the IoT-based surveillance model for monitoring children addressing weaknesses of existing models.

iii. To implement the proposed IoT-based surveillance model, incorporating features such as real-time attendance logging, geofence-based exit alerts, and secure data management.

iv. To evaluate the IoT-based surveillance model using goal-based evaluation and expert survey.

1.5 Research Questions

i. What are the weaknesses of IoT based surveillance models for monitoring children which are in existence?

ii. How can IoT based surveillance models for monitoring children be designed?

iii. How can an IoT based surveillance model for monitoring children be implemented?

iv. How can the developed IoT-based surveillance model be evaluated to ensure its effectiveness in ensuring the safety and security of children in schools?

1.6 Scope of the Study

This study aimed to develop an IoT-based surveillance model tailored for monitoring the safety of school children, with a focus on real-time tracking and information management. The research specifically addressed attendance tracking and the detection of unauthorized exits from school premises, ensuring that any attempt by a child to leave school grounds during school hours would be promptly detected and addressed.

The study was confined to primary day schools in Kenya, recognizing the unique safety and monitoring challenges faced by these institutions. Boarding schools and secondary schools were not included in the research to maintain a focused scope and address the pressing concerns related to day scholars who may be at risk of sneaking out or facing external threats during the school day.

Furthermore, the research considered the active involvement and perspectives of key stakeholders, including teachers, school administrators, parents, and the learners themselves. These groups were integral to both the design and evaluation of the surveillance model, ensuring that the system met the needs of all parties involved in safeguarding the children.

1.7 Justification of the Study

This research was justified by the urgent need for advanced surveillance models in educational settings. Existing models often lacked comprehensive coverage and real-time capabilities, hindering effective learner monitoring. The identified gaps in current literature and technology underscored the necessity for a more robust and efficient IoT-based surveillance model tailored to the unique context of school environments. This proposed model would assist school management in being accountable by using technology to monitor the movements of school children, a task that had been tedious and challenging. The implementation of the IoT-based surveillance model would improve the safety of children, discourage crimes against them, deter children from engaging in illegal activities, and reduce worries among parents. The model could also be used by law enforcement agencies in cases of missing children.

1.8 Significance of the Study

The significance of this research lay in its potential to revolutionize learner monitoring practices. By developing an optimized IoT-based surveillance model, the study aimed to contribute a novel solution that ensured real-time monitoring, attendance accuracy, and enhanced security within schools. The outcomes had broader implications, positively impacting educational institutions, parents, and policymakers. The advancements made through this research could set a precedent for innovative technological applications in the education sector, fostering a safer and more efficient learning environment while aligning with the evolving landscape of contemporary education.

1.9 Limitations and Assumptions of the Study

The study faced several limitations, first, hardware malfunctions or software bugs may hinder the model’s ability to collect and communicate real-time data accurately. Privacy concerns present another significant challenge, as learners, teachers, and parents may be reluctant to accept constant monitoring, which could affect the model’s overall acceptance and usage. Additionally, resource limitations, such as the need for substantial financial investment, appropriate infrastructure, and skilled personnel, create obstacles to the scalability and sustainability of the model.

Key assumptions were also made in the study. It was assumed that all stakeholders, including learners, parents, and school authorities, would willingly provide informed consent and actively collaborate to ensure the model’s success. Furthermore, the study presumed that end-users, including teachers, learners, and parents, would adopt and adapt to the IoT-based surveillance model without significant resistance.

To address technological constraints, such as hardware malfunctions or software bugs, regular maintenance checks and reliable troubleshooting mechanisms are recommended.

Privacy concerns, raised by stakeholders regarding constant monitoring, can be mitigated through clear communication and the implementation of strong data protection protocols, including encryption and access controls. Resource limitations, particularly financial investments, infrastructure, and skilled personnel, can be managed by introducing the model in phases and seeking external funding from government or private sector sources. Providing training for staff will further support model sustainability. Lastly, to ensure stakeholder cooperation and adoption of the IoT-based surveillance model, ongoing engagement and education, along with transparent policies, are necessary.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter provides a comprehensive review of the literature relevant to the research on IoT-based child monitoring models. It begins by exploring the fundamental concepts of the Internet of Things (IoT) and the devices that enable IoT applications, with a particular focus on their use in child monitoring models. The chapter then delves into an analysis of existing IoT-based surveillance models, highlighting their strengths and limitations. Following this, key security and privacy concerns related to IoT models are discussed, along with the challenges of implementing IoT in child monitoring environments. The chapter also outlines the evaluation metrics used for analyzing the performance and effectiveness of IoT models. After identifying a research gap in the current literature, the chapter concludes by presenting the conceptual framework that will guide the development of the proposed IoT-based child monitoring model.

2.2 Internet of Things (IoT)

Technological advancements and the internet revolution have introduced various new concepts, one of which is the Internet of Things (IoT). The term IoT was initially introduced by Kevin Ashton in 1999, referring to a global network infrastructure that connects numerous devices to the internet, facilitating the exchange of information and enabling smart recognition (El Mrabet & Ait Moussa, 2020). IoT represents a paradigm shift in which all objects can be interconnected, allowing for the identification of physical entities and the seamless transfer, storage, and processing of information between the physical and virtual realms, irrespective of time or location.

The Internet of Things (IoT) has become a critical element in the technological, social, and economic landscapes, influencing how we interact with everyday objects like consumer goods, vehicles, and industrial components, which are now equipped with internet connectivity and advanced data analysis capabilities (Mouha, 2021). As IoT devices increasingly become an unnoticed yet integral part of daily life, they are driving innovations such as "Smart homes," enhancing safety and energy efficiency, and transforming healthcare through wearable health monitors and network-connected medical devices (Mouha, 2021).

Central to IoT is the IoT platform, a key component of the broader ecosystem that connects all devices and supports their seamless interaction. This multi-layer technology simplifies provisioning, managing, and automating connected devices, while ensuring secure and scalable cloud connectivity. By offering ready-to-use features, IoT platforms accelerate the development of applications for these devices, ensuring cross-device compatibility and enterprise-grade security (Perwej Haq, Parwej, & Hassan, 2019). These platforms, often referred to as middleware, manage the interaction between hardware and application layers, performing essential tasks such as data collection, remote configuration, device management, and over-the-air firmware updates (Perwej *et al.*, 2019).

The essence of IoT lies in its ability to transform physical devices equipped with sensors into autonomous tools capable of collecting and transmitting data across networks. These smart devices interact with their surroundings and internal systems, facilitating decision-making processes without the need for human intervention. As IoT continues to evolve, its influence on daily life from smart cities to personalized healthcare—will only grow, creating an increasingly interconnected and intelligent world (Mouha, 2021; Perwej *et al.*, 2019).

Internet of Things Devices

Sensors are fundamental elements in the Internet of Things (IoT). In many applications, sensors are essential for gathering data and information about the system. Once the data is processed, commands are sent to actuators, which then influence the model's behavior. For instance, humidity sensors gather data to control irrigation models, while traffic sensors manage traffic lights, and occupancy sensors regulate building environments. These components are vital across various IoT applications, from smart cities to agriculture and personal health to transportation (Mouha, 2021).

Surveillance IoT devices that leverage GPS, RFID, and GSM technologies play a vital role in real-time tracking and monitoring across various safety applications, particularly for child safety. Each of these technologies serves specific functions, enhancing both the accuracy and reliability of tracking models.

2.2.1 Global Positioning System

GPS, a satellite-based navigation technology, is a cornerstone of location tracking in IoT models. It allows accurate identification of an individual's position and time by calculating coordinates using triangulation. For school children tracking, GPS is used to determine learners' boarding and deboarding locations, and the data is relayed to parents and school authorities through GSM communication. This triangulation involves signals from at least four satellites, ensuring precise real-time location updates (Anusha & Naidu, 2016). GPS-based tracking enhances security and accountability by providing instant updates on children's whereabouts, fostering a safer environment.

Figure 1

GPS Device

GPS Tracking Algorithms

GPS locating algorithms are computational techniques used to estimate the position of an object or individual by processing data from satellite signals. These algorithms are essential for filtering noise, predicting movement, and improving the accuracy and reliability of position estimates, especially in real-time applications. Common GPS algorithms include the Kalman Filter (KF), which optimizes the estimated position by minimizing errors in noisy data, and its non-linear variant, the Extended Kalman Filter (EKF), suitable for tracking models where dynamics are complex. Other advanced algorithms like the Particle Filter (PF) and Vector Tracking Loop (VTL) provide enhanced performance by handling multi-modal distributions and aggregating satellite signals, respectively (Kaplan & Hegarty, 2006; Luo, Yu, Li & El-Sheimy, 2019). These techniques ensure precise and stable location tracking even in challenging environments, such as urban areas with signal interference. The following list contains description of common GPS locating algorithms;

i. Kalman Filter (KF) and Extended Kalman Filter (EKF)

Kalman filters are widely used for estimating positions by minimizing the noise present in GPS data. EKF is an extension that handles non-linear systems, which makes it particularly suitable for tracking in environments where standard assumptions (like linearity) don't hold. This algorithm updates state estimates with each new GPS measurement, refining predictions and correcting deviations in real-time. Due to its adaptability, EKF is often used in tracking loops, especially in dynamic conditions where satellite signals are weak or noisy (Chen, Wang, & Xu, 2023; Petovello & Lachapelle, 2006).

ii. Adaptive Kalman Filter (AKF)

The adaptive variant of Kalman filters introduces self-adjusting features to improve accuracy under varying signal conditions. AKF modifies the covariance matrices dynamically based on environmental changes, such as satellite signal degradation or user dynamics. This flexibility makes it more effective for real-time tracking in urban settings, where GPS signals frequently suffer from multipath interference or obstruction (Chen, Wang, & Xu, 2023). It has been shown to enhance tracking reliability by adjusting parameters based on real-time observations.

iii. Vector Tracking Loop (VTL)

The VTL algorithm integrates the signals from multiple satellites into a single vector-based solution, enabling robust tracking under low-signal conditions. This is particularly beneficial when dealing with weak GPS signals, such as in densely built environments. Compared to scalar tracking methods, VTL improves performance by leveraging the correlation among signals from multiple satellites, resulting in more stable and accurate tracking (Luo, Yu, Li & El-Sheimy, 2019). Additionally, VTL-based approaches are resilient to spoofing attacks and have been used in conjunction with Kalman filters for enhanced accuracy.

iv. Particle Filter (PF)

Particle filters are more computationally intensive than Kalman filters but offer higher accuracy by tracking multiple potential states of the system simultaneously. Each particle represents a possible state, and the algorithm updates the weight of each particle based on observed data. PF is well-suited for scenarios with non-linearities and non-Gaussian noise, making it ideal for high-precision applications and environments with complex dynamics (Luo *et al.*, 2019). However, it is less common in consumer-grade GPS devices due to its computational demands.

v. Adaptive Iterated Extended Kalman Filter (AIEKF)

This algorithm is an improvement over the EKF, combining iterative correction steps with adaptive techniques. AIEKF refines estimates through repeated iterations, significantly reducing root-mean-square error (RMSE) values for tracking coordinates like latitude, longitude, and altitude. Studies indicate that AIEKF provides superior performance over standard EKF, especially under fluctuating signal conditions, making it highly effective for real-time GPS applications (Chen *et al.*, 2023).

2.2.2 Radio-Frequency Identification

Radio frequency Identification technology, another key component in surveillance models, uses tags that store data readable by RFID readers. Tags can be either active, with a power source, or passive, relying on energy from the reader. Passive tags are compact and energy-efficient, making them ideal for applications like school bus tracking. RFID models consist of tags and readers, where the reader retrieves data by emitting a signal toward the tag. In school environments, RFID can monitor attendance, track movement, and enhance security by maintaining an updated record of individuals' locations without requiring a direct line of sight (Jeyakkannan, Karthik, & Lukose, 2021).

Figure 2

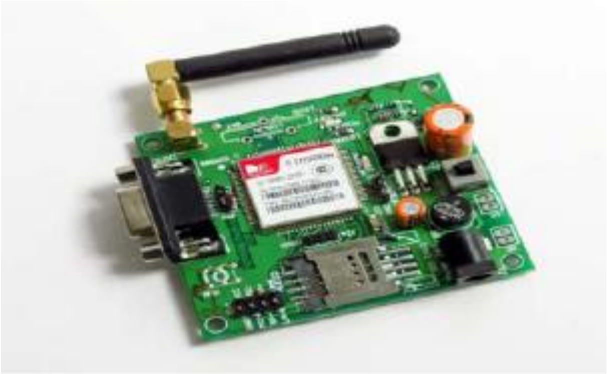
RFID Tag and Reader

2.2.3 GSM (Global System for Mobile Communications)

GSM is essential in IoT-based surveillance models as it facilitates communication between tracking devices and monitoring entities. GSM networks handle data transmission from sensors, such as GPS and RFID, to the relevant authorities. The system utilizes narrowband time-division multiple access (TDMA) for efficient data management, allowing reliable, wide-range data sharing capabilities. In child monitoring applications, GSM ensures that information about the child's location is available to parents or school administrators virtually anywhere, making it a critical element in real-time surveillance models (Anusha & Naidu, 2016).

Figure 3

GSM Sensor



2.3 Internet of Things in Surveillance Models

IoT surveillance models have transformed monitoring and security through features like real-time monitoring, remote access, and enhanced data processing. One such model, proposed by Lulla, Kumar, Deshmukh & Pole (2021), is an IoT-based smart security and surveillance model that uses ultrasonic sensors to detect intrusions. This model not only alerts property owners of unauthorized access but also provides a warning to intruders, allowing them to retreat without triggering an alarm. With additional features such as remote camera surveillance, face recognition for access control, and power failure alerts, this model offers a versatile and reliable security solution for homes, offices, and museums.

In another approach, Hu and Ni (2018) developed an IoT-driven automated object detection model for urban surveillance, focusing on vehicle detection and license plate recognition. This model enhances real-time traffic monitoring, security in restricted areas, and automatic parking management. Their method reduces the data volume required for storage, making it highly effective for smart city applications (Hu & Ni, 2018).

Ke *et al.* (2021) addressed the challenge of managing large volumes of surveillance data by developing a smart parking model that utilizes edge computing. This model processes data locally, reducing the reliance on cloud resources. With real-time video feed analysis and artificial intelligence algorithms, the model achieves over 95% accuracy in detecting parking occupancy, making it a key component in intelligent transportation models and smart cities (Ke *et al.*, 2021).

Elbasi (2020) proposed an IoT-based surveillance model focused on abnormal event detection in high-security environments, such as airports, borders, and streets. This model uses machine learning algorithms to analyze video feeds from multiple cameras, achieving more than 96% accuracy in detecting unusual human activities. It provides a cost-effective solution for areas requiring continuous monitoring to ensure security (Elbasi, 2020). In the healthcare domain, Hsu *et al.* (2017) developed FallCare+, an IoT-based fall detection model aimed at homecare services. This model integrates live video streaming with machine learning techniques to detect falls in real-time, making it highly valuable for elderly care. The model's continuous improvements through incremental development provide enhanced efficiency for homecare and medical services (Hsu *et al.*, 2017).

In conclusion, IoT-based surveillance models offer a range of functionalities, from intrusion detection and vehicle monitoring to healthcare applications like fall detection. These models leverage advanced technologies such as machine learning, edge computing, and AI to provide accurate and real-time monitoring, making them indispensable in modern smart cities, healthcare, and security applications.

2.4 Existing IoT-based Child Monitoring Models

This section reviews relevant research addressing the problem of child monitoring through IoT technologies. Numerous scholars have proposed various models utilizing technologies such as GPS, RFID, Ultra-Wide Band (UWB), Infrared, Wi-Fi, Zigbee, Global System for Mobile Communication (GSM), and Android smartphones to monitor children’s movements and ensure their safety. The following highlights IoT-based child monitoring models that have been developed using these technologies.

2.4.1 IoT-School Attendance Model Using RFID Technology

The proposed architecture for the IoT-based attendance model consisted of three distinct phases: detection, real-time notification, and the dissemination of missed lessons. In the detection phase, the model collected and processed information from learners, cross-referencing it with attendance records. This process utilized Ultra High Frequency (UHF) readers that operated within a range of six meters to identify learners through passive RFID tags attached to their identification cards. During detection, the UHF reader captured signals emitted by the RFID tags and transmitted this data via an Arduino board to a web server, which verified the learner's presence in the database. This automated mechanism eliminated the need for teacher intervention, thereby significantly enhancing the efficiency of classroom attendance management (El Mrabet & Ait Moussa, 2020).

The second phase involved real-time notification, allowing the model to inform school administrators and parents of a learner's attendance status. If a learner was marked absent, an immediate notification was dispatched via email or SMS. This real-time alert model ensured that parents and school authorities were kept informed of the learner's presence, alleviating the necessity for teachers to manually update attendance records.

This feature enhanced communication and monitoring between schools and parents, promoting transparency in attendance tracking (El Mrabet & Ait Moussa, 2020).

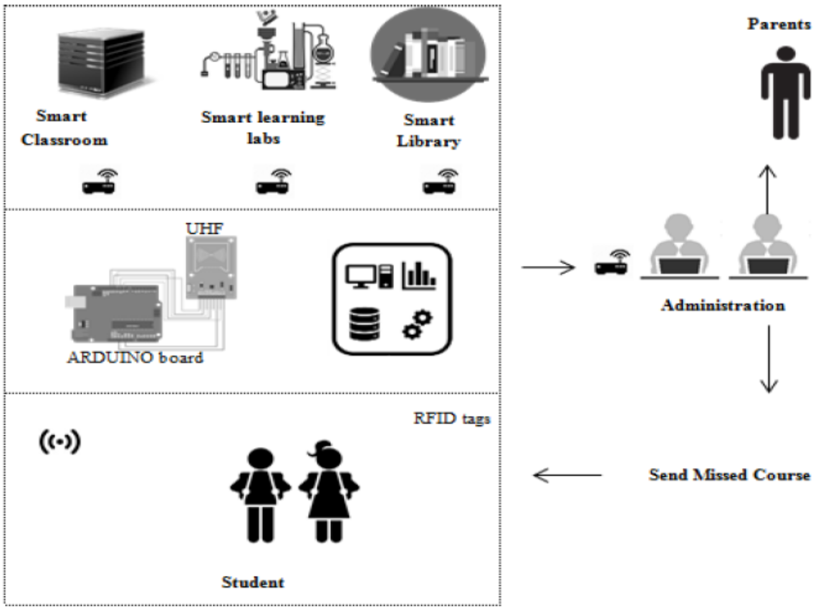
In the final phase, the model addressed the challenges associated with absenteeism by automatically sending missed lessons to learners. When a learner was recorded as absent, the model promptly emailed the relevant lesson materials. This functionality was critical for maintaining academic continuity, ensuring that learners who missed classes still had access to the necessary resources to stay engaged with their studies. As a result, the model fostered equitable learning opportunities for all learners, regardless of their physical presence in class (El Mrabet & Ait Moussa, 2020).

The implementation of this smart attendance model highlighted the increasing significance of Information and Communication Technology (ICT) within the education sector. Traditional attendance methods often consumed valuable instructional time, detracting from teaching activities. In contrast, the proposed IoT-based model not only automated attendance tracking but also enabled educators to allocate more time to instruction. Evaluations of this IoT attendance model demonstrated its effectiveness in enhancing learner performance by fostering collaboration between parents and schools in monitoring attendance. Furthermore, the feature that sent missed lessons contributed to learners' academic progress, ultimately improving performance on assessments (El Mrabet & Ait Moussa, 2020).

Looking ahead, the researchers intended to expand the application of IoT in education by developing a smart guidance model for high school learners. This model aimed to assist learners in navigating suitable academic and professional pathways, facilitating informed decision-making regarding their future educational choices (El Mrabet & Ait Moussa, 2020).

Figure 4

Architecture of IoT-School attendance (El Mrabet & Ait Moussa, 2020).



2.4.2 An Intelligent and Secured Tracking Model for Monitoring School Bus

The model proposed by Ahmed *et al.* (2019) introduced an intelligent and secured tracking model for monitoring school buses, featuring a dual-authentication mechanism designed to ensure the safety and security of learners. The model architecture began with the admin uploading learner and bus information into a web database. This information was critical for real-time tracking and security verification during the school bus journey. The dual-authentication process utilized both Fingerprint and RFID (Radio Frequency Identification) technologies to verify the identity of learners, significantly enhancing the security and reliability of the model. This initial step set up the authentication layers that protected the model from unauthorized access (Ahmed *et al.*, 2019).

In the first phase, the learner's fingerprint was used for initial verification. This represented the first level of authentication, where the fingerprint was scanned and compared against the database. If the fingerprint did not match any records, the attempt was marked as invalid, preventing unauthorized access (Ahmed *et al.*, 2019). The next phase involved RFID verification. Once the fingerprint was authenticated, the model proceeded to the second level of verification, where the RFID tag associated with the learner’s identification card was checked. The RFID record linked to the learner's fingerprint had to match; otherwise, the model considered the attempt invalid, and the parent or guardian was not notified of the learner’s arrival (Ahmed *et al.*, 2019).

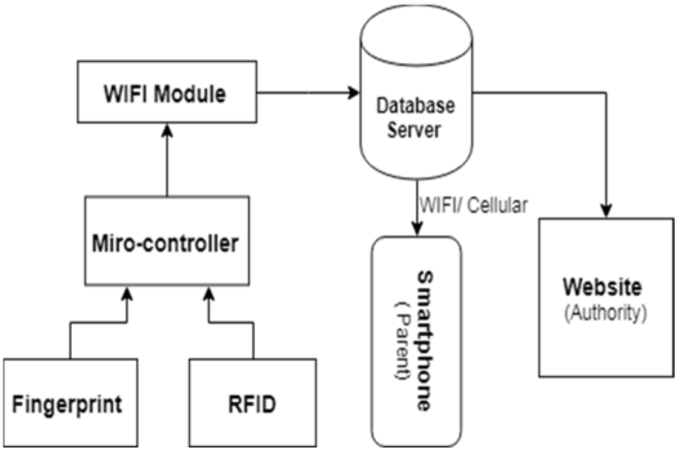
If both fingerprint and RFID checks were successfully matched, the model sent data to the database and immediately notified the parents via the associated Android application. This notification informed them of the learner’s safe arrival either at the school or the drop-off location. The model ensured that both authentication steps were repeated at the time of drop-off to prevent learners from disembarking at unauthorized locations. If a discrepancy was found between the fingerprint and RFID at this stage, the model flagged the attempt as invalid, adding another layer of protection (Ahmed *et al.*, 2019).

In addition to real-time verification, the model offered a live tracking feature for parents. Through an Android application, parents could view the current location of the bus on a map, allowing them to track its movement and arrival times. Notifications were sent at appropriate intervals, providing guardians with up-to-date information on the school bus’s status and ensuring they were prepared for pick-up or drop-off events. This aspect of the model significantly reduced the anxiety and uncertainty often associated with school bus travel, particularly in urban settings like Dhaka (Ahmed *et al.*, 2019).

The secure school bus model proposed by Ahmed *et al.* (2019) offered significant benefits, particularly in terms of enhancing learner safety and easing parental anxiety. By integrating fingerprint and RFID technology, the model ensured a robust and reliable dual-authentication process, reducing the risk of unauthorized access or potential misidentification. The real-time notification model, combined with the ability for parents to track the bus via an Android app, ensured that guardians were always informed of their child's whereabouts. This feature not only improved road safety but also added convenience by streamlining the pick-up and drop-off process. Overall, the proposed model offered a secure, intelligent, and highly reliable solution for school bus monitoring in urban areas (Ahmed *et al.*, 2019).

Figure 5

Proposed Architecture for Bus Monitoring Model



Source: Ahmed et al. (2019)

2.4.3 IoT-Based Touch-Free Attendance Model

The IoT-based touch-free attendance model (ITAS) proposed by Tamilselvan, Ramesh, Niveda, Poonguzhali & Dharani (2021) integrated various components like the NodeMCU, EM18 RFID module, MLX90614 infrared thermometer, and ultrasonic sensor to create a safe, efficient, and automated model for attendance tracking during the COVID-19 pandemic and beyond. The model was built on the NodeMCU platform, an open-source platform ideal for IoT applications, and utilized the ESP8266 Wi-Fi module for wireless connectivity. The NodeMCU's efficient power management, including a deep sleep mode, made it a low-cost, low-power solution, with the ESP-12 Wi-Fi module integrated to eliminate the need for additional hardware for wireless communication (Tamilselvan *et al.*, 2021).

The model began with the user presenting their ID card, embedded with an RFID tag, to the EM18 RFID reader, which used radio waves to activate the tag. The RFID reader communicated with the NodeMCU through UART communication to transfer the data. Simultaneously, the ultrasonic sensor measured the distance between the person and the MLX90614 infrared thermometer to ensure the person was within the 5-10 cm range required for accurate temperature measurement. If the person was not within range, an OLED display prompted the user to "come closer," ensuring the model worked only when optimal conditions were met (Tamilselvan *et al.*, 2021).

Once the user was in range and the RFID card was validated, the model checked the Tag ID against pre-defined data. If a match was found, the infrared thermometer measured the user’s temperature without any physical contact. If the temperature exceeded 98.7°F, the model automatically notified the management by sending an alert, and the learner's details were stored in the database, marked with a red indicator to highlight the fever condition. If the temperature was within the normal range, the data was marked green, and the learner’s attendance details were stored in the database (Tamilselvan *et al.*, 2021).

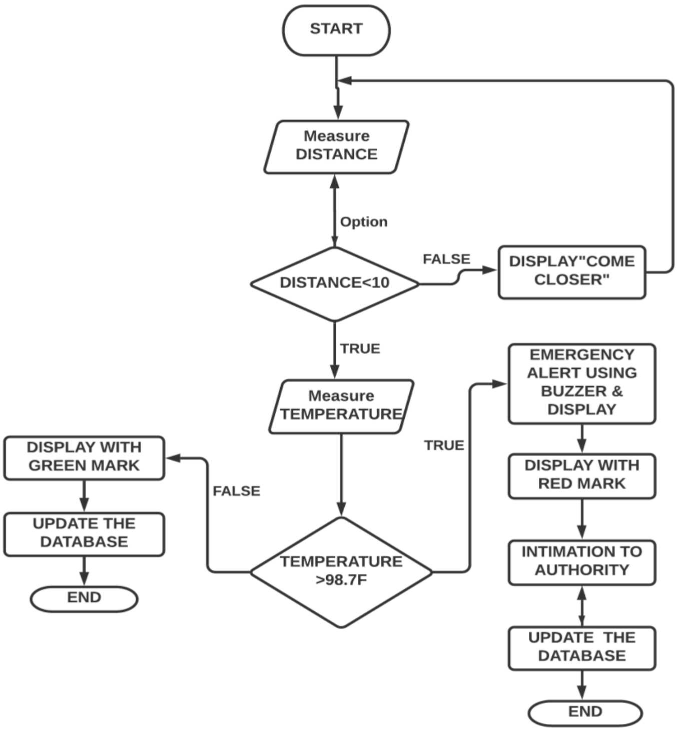
The data, including the ID number, roll number, temperature data, and other personal details, were sent to a web server. The model allowed for real-time monitoring by enabling administrators to access the data by entering the IP address provided by the model into a web browser. All data was stored in the cloud, allowing learners and faculty to check their information and generate weekly reports on attendance and health status, which added to the model's functionality and utility for long-term tracking (Tamilselvan *et al.*, 2021).

The proposed touch-free attendance model offered significant advantages over traditional models, particularly in environments requiring minimal physical interaction during health crises like the COVID-19 pandemic. It provided a cost-effective and immediate solution for tracking attendance while adhering to government-mandated safety protocols. The model's automation reduced the need for human intervention, further enhancing safety in workplaces and institutions. Additionally, the model's infrared temperature sensor allowed for efficient temperature measurement, which was critical for health monitoring. Tamilselvan *et al.* (2021) also suggested that the model could be enhanced by integrating additional technologies, such as using barcodes for competitive examination centers to streamline attendance processes further.

This touch-free attendance model offered an innovative and practical solution for educational institutions and workplaces. Its integration of RFID technology, infrared temperature sensing, and real-time data tracking via the cloud ensured that attendance was automated, safe, and efficient. By reducing human interaction and allowing for contactless temperature measurement, this model provided a robust solution during and after pandemic situations. The flexibility of the model allowed for further customization, making it adaptable to various organizational needs (Tamilselvan *et al.*, 2021).

Figure 6

Flow chart of the ITAS



2.4.4 Design of an Intelligent School Bus Monitoring and Reporting Model via IoT

Ajayalakshmi, Srujana, Ramesh, & Layasri (2021) proposed a school bus monitoring model designed to improve security for commuters. The model incorporated multiple features to enhance safety, including real-time bus tracking, learner detection, and monitoring of various potential issues such as excessive speed, driver intoxication, smoke detection, unscheduled stops, delays, and incidents. This multifaceted approach aimed to create a safer transportation environment for learners.

To facilitate learner identification, each learner was provided with an RFID passport stored in their backpacks. When a learner boarded or alighted from the bus, the RFID reader captured the corresponding information and transmitted it to the database for record-keeping purposes. This method not only streamlined the process of tracking learner attendance on the bus but also ensured that the information was accurately recorded and easily accessible for monitoring (Ajayalakshmi *et al.*, 2021). The model functioned by continuously tracking the bus's location in real-time, allowing stakeholders to monitor its progress and ensure adherence to the planned route. Additionally, learner detection technology facilitated the verification of learner presence during boarding and alighting, thereby ensuring the safety and accountability of each individual (Ajayalakshmi *et al.*, 2021). This proactive measure helped to mitigate the risks associated with learner transportation.

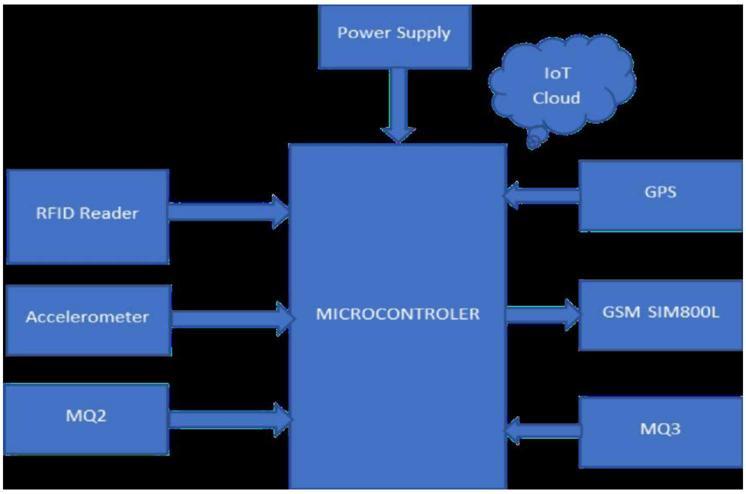
Furthermore, the model employed various sensors and detectors to monitor potential issues that could compromise passenger safety. These included detecting excessive bus speed, driver intoxication, smoke detection within the bus, and unscheduled stops. By continuously monitoring these factors, the model aimed to promptly identify and address any potential risks or incidents (Ajayalakshmi *et al.*, 2021). The integration of these technologies contributed to a comprehensive safety framework.

The collected data from the RFID reader, GPS tracking, and sensor inputs were stored in a database for further analysis and record-keeping. This data could be accessed and utilized by relevant authorities to evaluate bus performance, identify areas for improvement, and address any safety concerns that may arise (Ajayalakshmi *et al.*, 2021). The emphasis on data-driven decision-making underscores the model's potential for enhancing operational efficiency.

In summary, the school bus monitoring model introduced by Ajayalakshmi *et al.* (2021) integrated various features such as real-time tracking, learner detection, and monitoring of potential issues to enhance overall security. The utilization of RFID passports for learner identification, along with continuous data collection and analysis, aimed to provide a comprehensive solution for ensuring the safety and well-being of school bus commuters. This innovative approach represents a significant advancement in the field of learner transportation safety.

Figure 7

School Bus Monitoring and Reporting Model



Source: AJayalakshmi et al., (2021)

2.4.5 Student Monitoring Model for Boarding and Leaving Bus

The Student Monitoring Model for Boarding and Leaving the Bus proposed by Maturkar ,Nandanwar, Rahangdale, Lute & Zile (2020) was designed to enhance the safety of children during their transit to and from school. The model primarily used RFID technology to track learners' movements as they boarded and left the school bus. Each learner was provided with an RFID card containing a unique number, which was read by an RFID reader installed on the bus. When a learner scanned their card, the model sent an SMS notification to the learner's parents, informing them of the learner's boarding time and location. This feature allowed parents to continuously monitor their child's whereabouts during the bus journey (Maturkar *et al.*, 2020).

In addition to tracking learner entry and exit, the model also helped monitor if any learner was mistakenly left on the bus. The model kept a count of learners who boarded and left the bus, making it easier to identify if a learner remained on board. This was a crucial feature that could prevent incidents where a child might be forgotten on the bus. Furthermore, the model included a temperature sensor that monitored the internal temperature of the bus. If the temperature exceeded safe limits, the model alerted the driver, ensuring the bus environment remained safe for learners (Maturkar *et al.*, 2020).

The model also incorporated a speed limit control feature for the bus. If the driver exceeded the prescribed speed limit, a warning message was sent to the driver until the bus speed was reduced to within the allowed limit. The number of times the speed limit was breached was recorded and sent to the school authorities, enabling them to monitor the driver's performance. Additionally, in case the bus was delayed or changed its route, a notification was automatically sent to the parents of the students on that bus, keeping them informed about any unexpected changes (Maturkar *et al.*, 2020).

For enhanced safety, the model included an alcohol sensor to detect the alcohol levels of the driver. If the alcohol concentration exceeded the prescribed limit, the model triggered an alarm, and the bus would not start until the driver was replaced. This feature ensured that learners were not exposed to the risks associated with an intoxicated driver. The overall objective of this model was to provide real-time information to parents, keeping them informed about their child’s status during transit (Maturkar *et al.*, 2020).

The model utilized GSM technology for communication between the modules. By integrating RFID and GSM data, the model enabled cost-effective data transfer via SMS. GSM modems were connected to the microcontroller, which processed and transmitted the data. The coding for this model was written in Embedded C and compiled using Assembly Language, ensuring smooth operation and real-time data transmission. The

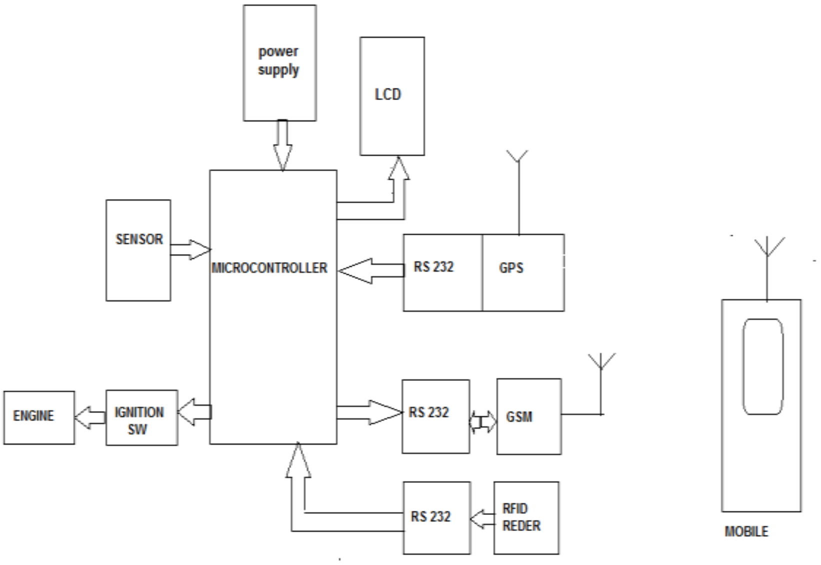
SMS service provided a reliable and low-cost method for parents to receive updates about their child's journey (Maturkar *et al.*, 2020).

The Learner Monitoring Model offered an economical and efficient solution for ensuring the safety of learners during bus rides. By combining RFID and GSM technologies, the model provided real-time updates to parents about their child's boarding and alighting times, helping them track their child's safety. The use of various sensors—such as temperature, speed, and alcohol detection sensors—further enhanced the safety of the bus journey. This technology minimized waiting times at bus stops and ensured emergency situations were handled effectively. Overall, the model provided a reliable solution that could improve the safety and efficiency of learner transportation (Maturkar *et al.*, 2020).

Model Design

Figure 8

Block Diagram of Student Monitoring Model For Boarding and Leaving Bus



2.4.6 IoT based children monitoring model in school based in Ethiopia

The IoT-based Children Monitoring Model developed by Tesfaye (2020) addressed the safety concerns of parents by proposing a smart tag device that monitored the status and location of children in real-time. This model used various sensors integrated into a smart tag worn by the children, which communicated critical information to parents and school authorities via text messages. The device aimed to provide parents with continuous updates on their child’s status and safety, ensuring they could monitor their child's movements and well-being through their mobile phones (Tesfaye, 2020).

The key sensor in the model was the GPS module (NEO-6M-O-001), which was responsible for tracking the child’s location and movement. The smart tag was designed to detect the child’s activity and categorize it into four statuses: “Sleeping,” “Studying,” “Exercising,” and “Dangerous.” These statuses were displayed in the text messages sent to the parents or guardians. The GPS module continuously monitored the speed and movement of the child. If the sensor detected a change in the child’s activity lasting more than three seconds, the status changed to “Exercising.” In contrast, when the child was not in motion, the model categorized the status as “Studying” (Tesfaye, 2020).

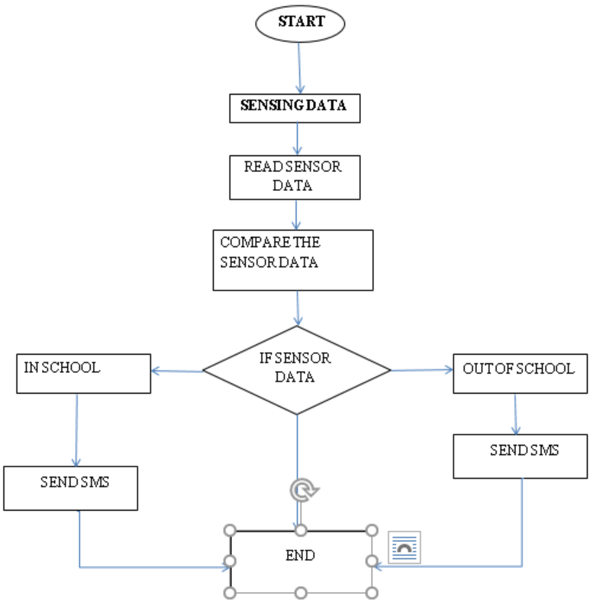
One of the most critical features of this model was its ability to detect when a child was in danger. If the child was absent from school or left the school compound during class time, the status changed to “Dangerous,” and an alert message was sent to the parents or guardians. The model used GPS tracking to provide real-time location data, allowing the school principal or class monitor teacher to be notified if the child was not on school premises. This feature ensured that parents and school authorities were promptly informed of any potential threats to the child's safety, allowing for quick intervention (Tesfaye, 2020).

The GPS module also tracked the child’s route from home to school. If the current location of the child did not match the school’s location during school hours, the model sent an alert with the child’s current location. This functionality was designed to give parents peace of mind by enabling them to track their child’s whereabouts throughout the day. The model used the speed data collected by the GPS module to determine the child’s activity status. It calculated speed in kilometers per second (KMPS) and compared the measured speed with a threshold value to determine the child’s status and send the appropriate text message to the parents (Tesfaye, 2020).

The IoT-based Children Monitoring Model proposed by Tesfaye (2020) offered an innovative solution to enhance the safety of children while they were at school or on their way to and from school. By integrating GPS technology, the model provided real-time monitoring of the child's location and activity status, ensuring that parents and school officials were immediately informed of any potential risks or changes in the child’s routine. The use of text message notifications and location tracking offered a practical, low-cost solution that could be easily accessed by parents. Overall, this model enhanced the ability of parents and schools to monitor children’s safety and ensured prompt responses in emergency situations (Tesfaye, 2020).

Figure 9

Flowchart Diagram For IoT Based Children Monitoring Model in Ethiopia



2.4.7 Child Safety Monitoring Model Based on IoT

The model proposed by Senthamilarasi, Bharathi, Ezhilarasi & Sangavi (2019) was an IoT-based child safety monitoring model designed to provide real-time tracking and monitoring of children. It integrated several components such as a Raspberry Pi microprocessor, GPS, GSM, and various sensors. The model was designed to be user-friendly, requiring parents or guardians to register with their credentials to use the application, which interfaced with the monitoring device. The child was equipped with a monitoring device, and geofencing was applied by setting predefined boundary values within the code. GPS technology was used to track the child's movements, and the data was stored on a server. If the child moved beyond the specified boundary, an alert was triggered through GSM technology, notifying the registered users via an automated call (Senthamilarasi *et al.*, 2019).

The model employed live GPS tracking, updating the location of the child at regular intervals on the server and providing real-time location updates on the application. This functionality allowed parents or guardians to log in and monitor the child’s current location after receiving an alert. Additionally, geofencing was used to maintain attendance records, tracking the entry and exit of the child from school premises, which was updated in the application (Senthamilarasi *et al.*, 2019). The live data, combined with geofencing, provided an efficient way to track and log child movements in real-time.

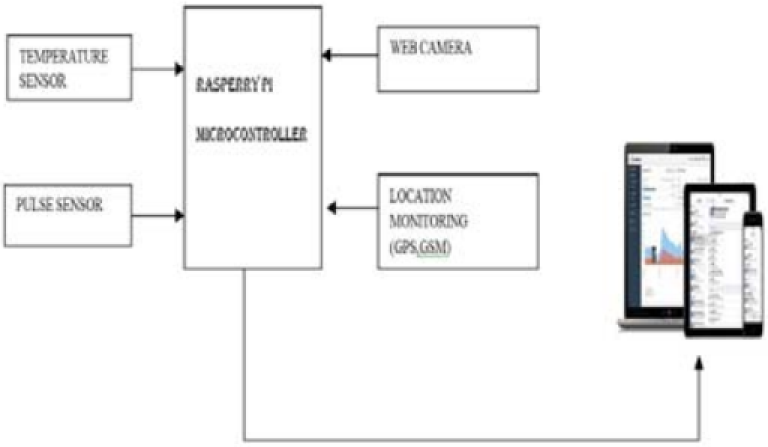
The model also integrated sensors such as temperature and pulse sensors, which monitored the child's health by detecting abnormal variations in body temperature and pulse rate. Predefined threshold values were coded into the model for these sensors, and if the child’s vitals exceeded these thresholds, the model sent alert messages to the registered users. The model compared current sensor data to the preset threshold values, and any abnormal readings resulted in an alert message being sent through GSM. The user could log in to the application to review the alert details and updated health status (Senthamilarasi *et al.*, 2019).

Moreover, the model incorporated live streaming functionality. After receiving an alert, users could visually check the child's status by entering the IP address of a camera synced to the model. Once synced, users could view live-streaming video via the server, providing a visual confirmation of the child’s safety (Senthamilarasi *et al.*, 2019). This feature enhanced the overall functionality of the model by adding an additional layer of surveillance. In terms of results, one of the key modules in this model was the temperature sensor, which monitored both the child’s body temperature and the surrounding environment. If there was an abnormal rise or fall in temperature, the model notified the user with a delay, as per the predefined values in the code. The notification included real-time temperature and humidity values, helping the users to take immediate action in case of any discrepancies in the child’s vitals (Senthamilarasi *et al.*, 2019).

The IoT-based child safety monitoring model by Senthamilarasi *et al.* (2019) provided an efficient, real-time solution for tracking and monitoring children's safety. It incorporated geofencing, GPS tracking, GSM-based alerts, and sensor-based monitoring, allowing users to track their child's movements and health status. The model offered real-time data updates and alert mechanisms, ensuring that any abnormal conditions triggered immediate notifications. Additionally, the integration of live video streaming enhanced the monitoring process by providing visual confirmation of the child's safety. This model showcased the potential of IoT in enhancing child safety with the combination of multiple technologies to ensure continuous and reliable monitoring.

Figure 10

Block diagram for Child Safety Monitoring Model Based on IoT



2.4.8 Weaknesses Based on Technologies Used of Existing Child Monitoring Models Analyzed

The Table 1 was designed to achieve Objective 1: To analyze the weaknesses of existing IoT-based surveillance models for monitoring children includes detailed information about various existing child monitoring models. Each row of the table corresponds to a different model, and the columns provide the following details:

i. **Authors and Name of Model**: The proponents of the model and the title of the model.

ii. **Functionalities**: This column describes the key functionalities provided by each monitoring model.

iii. **Technology Used**: This section outlines the specific technologies each model relies on, such as IoT sensors, RFID tags, Bluetooth Low Energy (BLE), or GSM modules. It emphasizes the technical approach each model adopts for achieving its tracking or monitoring objectives.

iv. **Identified Weaknesses**: This crucial column analyzes the weaknesses of each model, focusing on limitations inherent to the technologies employed. For example, weaknesses may include limited range in RFID models, high power consumption in GPS-based models, or challenges in data security and privacy when using IoT networks. The analysis is based on factors like model scalability, customization capabilities, accuracy of location tracking, and potential gaps in coverage.

The table provides a structured comparison of existing models, enabling a clear understanding of their strengths and, more importantly, their shortcomings. This analysis serves as a foundation for identifying areas where the proposed IoT-based surveillance

Model can improve, offering more tailored and effective solutions for child monitoring in schools.

Table 1

Summary of Models Showing Functionality and Technological Weaknesses

Authors & Model Name	Technology Used	Functionality	Identified Weaknesses
El Mrabet & Ait Moussa (2020) IoT-School Attendance Model Using RFID Technology	UHIF RFID	Sends immediate alerts to parents and school administration regarding absences. Automatically sends missed lessons to absent students. Utilizes UHF RFID readers to accurately detect students within a 6-meter range. Centralizes student attendance data for reporting and analysis. Alerts for Absences: Notifies parents via SMS or email about their child's absence.	No real-time updates on attendance. Tag owners cannot be verified RFID requires close proximity to be scanned by readers, so queues can be formed which is a problem. Privacy concern because the model relies on RFID tags that can be read by any compatible reader, raising potential concerns about unauthorized access to student information. Does not offer additional insights into student behavior or location beyond entry and exit records. It lacks functionality for tracking students' real-time movements within the school premises, which could be important for safety monitoring.
Ahmed et al (2019) An Intelligent and Secured Tracking Model for Monitoring School Bus	RFID, GPS, Fingerprint sensor.	RFID for Identification GPS for monitoring bus location Finger-print for identification Record daily attendance Issue alerts of present and absentee learners	No real-time updates on attendance. The dual-authentication process using fingerprint and RFID technology can complicate the model and cause delays during boarding and drop-off, particularly if either scanner malfunctions, leading to hold-ups, especially during peak times with multiple students. RFID tag owners cannot be verified. RFID requires close proximity to be scanned by readers

		Bus location tracking	Tedious because separate modules are required for GPRS and RFID Biometric models can be sensitive to various factors such as dirty or damaged fingers, which can cause errors in reading fingerprints, especially among younger students who may be less careful.
Senthamilarasi et al (2020) Child Safety Monitoring System Based on IoT	Temperature Sensor, Pulse Sensor, GPS, GSM,	The Temperature sensor to measure temperature of child. Pulse sensor for heart rate. Web camera for video streaming. GPS for tracking location. GSM for sending SMS feedback.	The device was Impractical for children to wear because of the many sensors required. Sensor Reliability: The accuracy of health monitoring sensors (temperature and pulse) is crucial, as malfunctions or miscalibrations can result in false alarms or missed critical health alerts, leading to unnecessary panic or neglect of actual issues. Potential for Technical Failures**:** The model relies on multiple components working seamlessly; a failure in any part (e.g., server, camera, or communication module) can render the entire monitoring model ineffective. Scalability Concerns: As the number of monitored children increases, ensuring the model can manage multiple simultaneous connections and data processing without latency or failure may be challenging. Setup Complexity: Requires integration of multiple sensors and modules, which may complicate initial implementation. Live streaming limitations: The effectiveness of the live streaming feature relies on network conditions; poor connectivity may prevent users from accessing the live feed, hindering their ability to confirm the child's safety.
Tesfaye, (2020) IoT based children monitoring model in school based in Ethiopia	RFID, GPS, GSM	GPS for tracking location. GSM for sending SMS feedback.	Used a smart tag which children could easily have lost. Tedious because separate modules are required for GPRS and RFID Gave no alerts when children moved out of the designated area. False alarms and status changes: The activity change detection mechanism may trigger false alarms, mistakenly labeling a child as “Dangerous” for legitimate reasons, such as being outside for a scheduled school activity, which can cause unnecessary panic for parents. Battery Dependency: The smart tag device requires a reliable power source; if the

			battery runs low or dies, the model cannot monitor the child’s status or location, leaving parents uninformed.
Student Monitoring Model for Boarding and Leaving Bus (Maturkar *et al.*, 2020)	RFID, GPS, Alcohol Sensor, GSM.	RFID for Identification GPS for tracking bus location Detect student presence Issue alerts of present and absentee learners Temperature sensing of bus Monitoring intoxication levels of driver	No real-time updates on attendance. Environmental limitations: The temperature sensor's effectiveness can be limited by environmental factors like extreme heat or cold, which may affect its accuracy and lead to unreliable alerts. RFID tag owners cannot be verified. RFID requires close proximity to be scanned by readers. Tedious because separate modules are required for GPRS and RFID. Alcohol sensor reliability: While the alcohol detection feature enhances safety, its reliability may be questionable, with false positives or malfunctions potentially causing unnecessary bus delays and inconveniences for parents and student Privacy Concerns: The collection of data on students' movements raises privacy concerns, and while secure storage and prevention of misuse are critical, the model does not explicitly address data security measures. Cost Implications: Implementing a comprehensive monitoring model with various sensors and technologies may incur significant initial and ongoing costs for schools, including expenses for hardware, software, maintenance, and staff training
IoT-Based Touch-Free Attendance Model (Tamilselvan *et al.* 2021)	RFID, Temperature Sensor, Ultrasonic sensor	Utilizes RFID technology for contactless student attendance recording. Employs an MLX90614 sensor to measure student temperatures without physical contact. Uses ultrasonic sensors to ensure students are within an appropriate range from the	RFID technology may have range limitations, affecting detection in larger areas. Temperature readings may be influenced by external conditions, potentially leading to inaccurate results. The effectiveness of distance monitoring relies on the precision of ultrasonic sensors. Setup Complexity: Requires integration of multiple sensors and modules, which may complicate initial implementation. If the sensors or the NodeMCU board experience technical issues or malfunctions, it could disrupt the entire attendance tracking process until repairs are completed, impacting the model's overall reliability.

		temperature sensor. Sends attendance and health data to a cloud server for real-time access and reporting. Allows students and faculty to check their attendance and generate reports via a web interface.	Scalability Issues: The model offers a cost-effective solution for small to medium-sized institutions, but scaling it for larger organizations may pose challenges in managing increased RFID data and temperature readings, potentially impacting the speed and efficiency of data processing and storage.
Design of an Intelligent School Bus Monitoring and Reporting Model via IoT	RFID, GPS, Speed sensor, Alcohol sensor, Smoke sensor	RFID for Identification GPS for tracking bus location Detect student presence Issue alerts of present and absentee learners Smoke detection in bus Speed monitoring	No real-time updates on attendance. Complexity of Integration and Maintenance: Integrating multiple sensors (for speed, intoxication, smoke detection, etc.) adds complexity to the model, requiring specific maintenance and calibration for each to ensure accuracy and functionality, which increases the operational burden on school authorities. If one component fails, it could compromise the entire monitoring model. False Positives and Alerts: The model monitors potential issues like excessive speed and driver intoxication, but the algorithms may produce false positives, causing unnecessary panic or confusion among parents and school authorities and undermining trust in the model. The model depends on GPS tracking for real-time bus monitoring, but disruptions in signal availability—common in urban areas with tall buildings or rural areas with limited visibility—could result in inaccurate or delayed tracking information. Cost Implications: Implementing a comprehensive monitoring model with various sensors and technologies may incur significant initial and ongoing costs for schools, including expenses for hardware, software, maintenance, and staff training. RFID tag owners cannot be verified. RFID requires close proximity to be scanned by readers which could create queues leading to another problem.

2.5 Security and Privacy Issues in IoT

With the advent of IoT technology, many traditional applications have evolved into IoT-based smart applications. While considerable progress has been made in developing architectures and protocols for these applications, security and privacy issues continue to be significant concerns. As highlighted by Mohanta, Ramasubbareddy, Daneshmand & Gandomi (2021), IoT technologies face various security and privacy challenges, particularly given the resource constraints of devices and the diverse attack models that apply to IoT applications. The authors detail how the IoT applications developed within their framework encounter multiple security and privacy issues, emphasizing the need for robust solutions.

Security and privacy challenges are pivotal when developing IoT applications, especially concerning authentication and data protection. Mohanta *et al.* (2021) note that technologies like blockchain, fog computing, and machine learning can be instrumental in addressing these challenges. For instance, Jaiswal, Sobhanayak, Mohanta & Jena (2017) proposed a secure framework for data collection in smart healthcare models, which utilize intelligent devices for monitoring critical patients. These smart devices can connect wirelessly or through wired connections, employing technologies such as ZigBee, Bluetooth, or Wi-Fi. However, each connection method presents unique vulnerabilities, highlighting the need for lightweight algorithms or protocols tailored to the resource constraints of IoT devices. Satapathy, Mohanta, Jena & Sobhanayak (2018) proposed an elliptic curve cryptography (ECC)-based algorithm for IoT applications due to its lower computational key size requirements.

The IoT infrastructure comprises three primary layers: physical, network, and application. Security issues permeate each layer, and the following sections elucidate these security and privacy concerns in detail.

2.5.1 Security Challenges in IoT

This section identifies the security challenges associated with IoT applications, which predominantly operate on a three-layer architecture: physical, network, and application layers. In the physical layer, devices are connected through gateways, but their limited capabilities make them susceptible to various attacks. Once a device is compromised, replacing the entire hardware component may not be feasible, underscoring the need for comprehensive security measures across all layers (Mohanta *et al.*, 2021).

i. **Node Capture Attacks**: Smart devices deployed across different locations can be captured or replaced by attackers, allowing unauthorized access to the network. Distinguishing between legitimate and compromised nodes can be particularly challenging in this scenario (Mohanta *et al.*, 2021).

ii. **Replay Attacks**: Attackers may intercept messages from the communication medium and resend them later, impersonating authorized nodes within the IoT network (Mohanta *et al.,* 2021).

iii. **Side-Channel Attacks**: In these attacks, the adversary attempts to derive plaintext from ciphertext, leveraging timing information to extract cryptographic keys used in encryption (Mohanta *et al.*, 2021).

iv. **Eavesdropping**: This passive attack allows adversaries to intercept communications between devices, posing serious risks if the communication channel is not secure (Mohanta *et al.*, 2021).

v. **False Data Injection**: Smart devices gather environmental data, but attackers can inject false information into the network by exploiting insecure communication channels (Mohanta *et al.*, 2021).

vi. **Spoofing**: In the network layer, attackers can masquerade as legitimate nodes, sending false messages and compromising the integrity of the network (Mohanta *et al.*, 2021).

vii. **MITM Attacks**: In man-in-the-middle attacks, attackers intercept and manipulate data packets during transit, leading to both passive and active exploitation of information (Mohanta *et al.*, 2021).

viii. **Sinkhole Attacks**: This routing attack compromises nodes in the network, creating traffic congestion and degrading network performance (Mohanta *et al.*, 2021).

ix. **DoS Attacks**: Denial-of-service attacks overwhelm the server with excessive traffic, rendering it unable to utilize its full bandwidth and resources, although they do not typically lead to data loss (Mohanta *et al.*, 2021).

x. **Unauthorized Access**: Attackers target resource-constrained devices connected to IoT applications, capturing authorization credentials to gain network access (Mohanta *et al.*, 2021).

xi. **Phishing Attacks**: Given the diverse user base of IoT applications, attackers often attempt to gather sensitive information by sending deceptive messages or emails to users (Mohanta *et al.*, 2021).

xii. **Trust Management**: Trust management poses challenges in the application layer, especially when real-time monitoring requires users to share personal information. Maintaining trust among nodes is essential to ensure security (Mohanta *et al.*, 2021).

xiii. **Authentication**: As IoT applications involve various intelligent devices, ensuring that each device is authenticated before network access is crucial. Unauthorized access can lead to the corruption of model data (Mohanta *et al.*, 2021).

xiv. **Malicious Attacks**: Insecure communication channels expose smart devices to malicious code injection, jeopardizing their functionality (Mohanta *et al.*, 2021).

xv. **Policy Enforcement**: Implementing robust security policies is vital for ensuring user privacy while utilizing smart devices effectively (Mohanta *et al.*, 2021).

2.5.2 Privacy Challenges in IoT

The foundational architecture of IoT comprises three layers: physical, network, and application layers. Numerous smart devices within the physical layer collect vast amounts of environmental data. Data collection occurs through three primary methods:

i. **Collection**: Sensors and smart devices gather raw data and forward it for processing (Mohanta *et al.*, 2021).

ii. **Aggregation**: The collected data is combined to derive meaningful information for subsequent processing (Mohanta *et al.*, 2021).

iii. **Analytics**: Through various analytical techniques, meaningful insights are extracted from the aggregated data (Mohanta *et al.*, 2021).

While these data collection and processing steps are crucial for IoT applications, they also raise significant privacy concerns. For example, in a hospital setting, if an attacker gains access to sensitive patient information, it could lead to severe privacy violations. Similarly, in smart city applications, unauthorized access to users' locations and travel details poses similar risks. To address these privacy challenges, effective privacy preservation techniques must be developed within the IoT framework (Mohanta *et al.*, 2021).

2.5.3 Security and Privacy Requirements for IoT Child Monitoring Models

In this section, we address security and privacy techniques for developing secure IoT-based child monitoring models, which are essential in maintaining the confidentiality and integrity of sensitive data like location, health, and activities of children.

i. Privacy by Design

The concept of *privacy by design* is critical for resolving privacy issues in IoT-based child monitoring models. This strategy involves embedding privacy into the model architecture from the outset, ensuring proactive privacy protection rather than reacting after privacy breaches occur. For instance, in the context of child monitoring, privacy considerations should be the default settings, protecting data throughout its entire lifecycle. Additionally, models should ensure transparency and visibility, respecting both parental and child privacy (Al-Turjman, Zahmatkesh, & Shahroze, 2022). These principles have been successfully applied in similar settings, such as remote health monitoring models, where proactivity in privacy protection is crucial (Preuveneers & Joosen, 2016).

ii. Testing and Verification

Testing and verification play a crucial role in ensuring that IoT-based child monitoring models meet their security and privacy requirements. The objective is to identify and prevent any potential information leaks, particularly when dealing with sensitive data such as a child's location or personal activities. Incorporating privacy-related testing into regular testing protocols is necessary to maintain the integrity of these models. Black-box differential testing, for instance, can be used to detect vulnerabilities in the model before they become exploitable by attackers (Jung, Sheth, Greenstein, Wetherall, Maganis & Kohno, 2008; Al-Turjman *et al.*, 2022).

iii. Privacy Architecture

A robust privacy architecture is essential for preventing data leakage within IoT-based child monitoring models. For instance, a trusted data storage model with an intermediary broker could be used to manage access to sensitive child-related data. Such architecture ensures that only authorized individuals, such as parents or teachers, can access data. Cryptographic techniques, including encryption, can further enhance the protection of sensitive information collected by IoT devices in the monitoring model (Choi, Chakraborty, Charbiwala, & Srivastava, 2011; Al-Turjman et al., 2022).

iv. Data Minimization

Data minimization is a key principle in privacy by design, especially for child monitoring models that collect large volumes of data from sensors, cameras, and GPS devices. The goal is to collect only the data necessary for the model's functionality, reducing the risk of privacy breaches. For instance, cameras monitoring children should be designed to avoid recording unnecessary information that does not pertain to the safety or location of the child. This minimizes the chances of unauthorized data usage or breaches (Gürses, Troncoso & Diaz, 2011; Al-Turjman *et al.*, 2022).

v. Secret Sharing

Secret sharing is a method that ensures the confidentiality of sensitive information in IoT-based child monitoring models. By dividing data into multiple shares, this method allows for secure distributed data storage. Only a specific number of shares are required to reconstruct the data, ensuring that it remains protected even if some parts of the network are compromised. This technique can be applied in IoT models that aggregate data from multiple child-monitoring devices, ensuring privacy and reliability (Al-Turjman *et al.*, 2022).

vi. Model Security and Access Control

For child monitoring models, maintaining robust model security and access control is imperative to protect sensitive information. Vulnerabilities in these models can allow unauthorized individuals to access child-related data, putting the privacy and safety of children at risk. By implementing strict access control measures, only authorized users, such as parents or school staff, can retrieve and manage data. Additionally, access control helps prevent the misuse of stored data, such as location information, by unauthorized third parties (Denning, Matuszek, Koscher, Smith & Kohno, 2009; Al-Turjman *et al.*, 2022).

vii. Secure Multiparty Computation

In the context of IoT-based child monitoring, secure multiparty computation can be used to allow multiple stakeholders—such as parents, school administrators, and healthcare professionals—to perform calculations on shared data without revealing their private inputs. For example, if monitoring includes sensitive health data, this cryptographic technique ensures that only the results of computations are shared, while the underlying data, such as the child's health records, remains private (Jha, Kruger & Shmatikov, 2008; Al-Turjman *et al.*, 2022).

2.6 Challenges of IoT Implementation in Child Monitoring Models

The implementation of IoT in child monitoring models faces several challenges which are listed below;

i. **The** data resolution of wearable sensors. These sensors are small to ensure comfort for children, but their compact design often comes with lower resolution compared to non-wearable devices. This can lead to less accurate tracking data, which may affect the overall efficiency of the child monitoring model. As IoT models for child monitoring require precise and reliable data, improving the resolution of wearable sensors remains a critical challenge (Dian, Vahidnia, & Rahmati, 2020).

ii. **Power consumption** is another significant challenge. Wearable devices used in child monitoring must operate for extended periods without requiring frequent battery replacements or recharging. This is particularly important in models where minimizing human interaction is a priority. Energy-efficient designs, such as incorporating solar power or other energy-harvesting technologies, can help address this issue. However, the limited availability of solar energy indoors and during nighttime poses an obstacle to the widespread use of these methods (Wu, Redoute, & Yuce, 2017).

iii. The **wearability** of IoT devices in child monitoring is also crucial. Devices must be lightweight and comfortable enough to be worn by children without causing discomfort or disrupting daily activities. The trade-off between maintaining low weight and incorporating complex computational functionalities creates a design challenge. Recent innovations, like smart clothing integrated with IoT, aim to enhance comfort while maintaining the necessary functionality for real-time child tracking and monitoring (Chen *et al.*, 2017).

iv. **Safety** concerns also arise with IoT-based child monitoring models due to the use of wireless technologies. These technologies emit radiofrequency radiation, which, when in close contact with the body, could pose health risks, especially if the devices are worn on sensitive areas like the head or eyes. As IoT devices transmit data to base stations, ensuring minimal radiation exposure is essential, especially in environments with poor network coverage where devices might increase transmission power (Dian & Vahidnia, 2019).

v. One of the more complex issues with IoT child monitoring models is **security**. Since these devices prioritize low power consumption and reduced complexity, their security features are often less robust. This makes them susceptible to hacking, which could result in unauthorized access to sensitive data, such as a child's location. Implementing stronger encryption protocols while maintaining the device's operational efficiency is a major challenge for developers of IoT-based monitoring models (Lomotey, Sofranko, & Orji, 2018).

vi. Another critical obstacle is the lack of **regulation** surrounding IoT devices, particularly in child monitoring. Without standardized regulations governing the use of such technologies, it is challenging to implement child tracking models across different regions or industries. In some areas, although the technology is available, regulations have yet to catch up, which limits the deployment of IoT monitoring models (Dian, Vahidnia, & Rahmati, 2020).

vii. Finally, **privacy** is a major concern in child monitoring models. Constant data exchanges between wearable IoT devices and monitoring hubs create opportunities for privacy breaches. IoT devices often operate in broadcast mode, making personal data easily accessible to unauthorized parties if proper security measures are not in place. Enhancing privacy protocols, such as employing selective data-sharing models, is necessary to protect sensitive information related to children (Lomotey, Sofranko, & Orji, 2018).

2.7 Evaluation Metrics for Analyzing IoT Models

When analyzing IoT models, several key metrics derived from the ISO/IEC 25010 quality model are often used. The following metrics are among those outlined by the standard:

i. The first metric is **functional suitability**, which measures how well an IoT model meets the needs and requirements of its users. In child monitoring models, this would assess how effectively the model tracks children's locations and provides real-time updates under specific conditions (Ashouri, Lorig, Davidsson, & Spalazzese, 2019).

ii. Another critical metric is **performance efficiency**, which evaluates the model's performance in relation to the resources it uses. For IoT child monitoring models, this could involve assessing the efficiency of data transmission, ensuring that location updates are timely without overloading the network or depleting device batteries quickly (Ashouri *et al.*, 2019).

iii. **Compatibility** is also a vital metric. It refers to the model's ability to function in different hardware or software environments. In child monitoring, IoT models need to seamlessly integrate with various devices, such as smartphones, tablets, or wearables, while exchanging information without errors (Ashouri *et al.*, 2019).

iv. **Usability** measures how easily users can interact with the model. For a child monitoring IoT model, this would focus on how intuitively parents and teachers can use the tracking interface to access information about the child's location and activities (ISO/IEC, 2011).

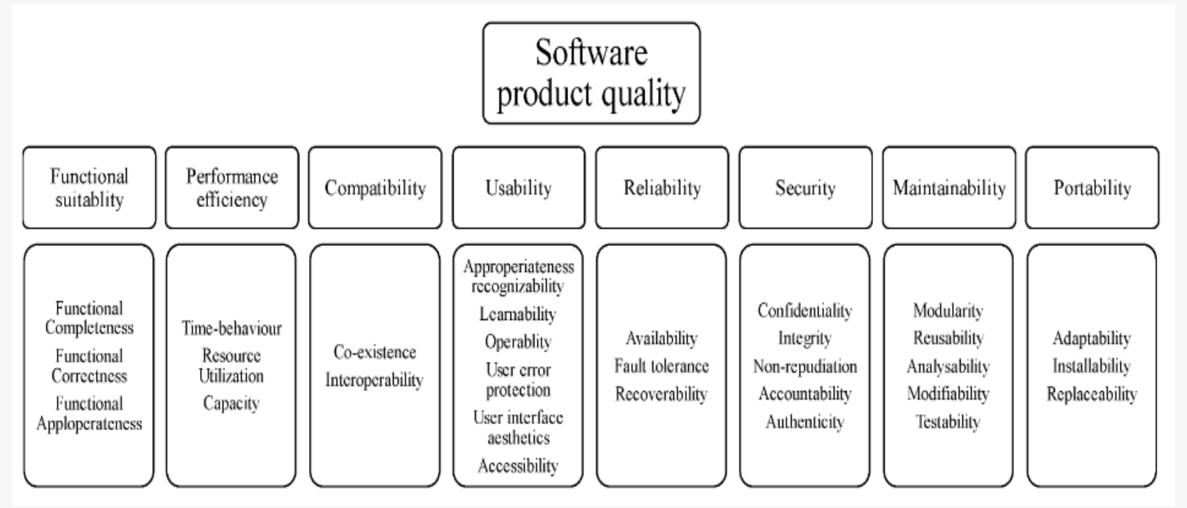
v. **Reliability** is another important consideration, which evaluates whether the model consistently performs its functions under specified conditions. For instance, in child monitoring, the IoT model must provide accurate tracking data over a prolonged period without failures (Ashouri *et al.*, 2019).

vi. **Security** in IoT models is paramount, especially in child monitoring applications where sensitive data like location and personal information are transmitted. The metric assesses how well the model protects data and prevents unauthorized access (Ashouri *et al.*, 2019).

vii. **Maintainability** refers to how easily the model can be modified, corrected, or adapted to changing conditions, a vital factor for IoT models that need to scale or update frequently (ISO/IEC, 2011). Lastly, **portability** measures the ease with which the model can be transferred to different environments, which is crucial for IoT child monitoring models that might be used across various platforms and devices (Ashouri *et al.*, 2019).

Figure 11

ISO/IEC 25010 Quality Model



2.8 Research Gaps

Many IoT-based child monitoring models have been proposed, but most rely on multiple sensors and utilize different technologies, resulting in generalized solutions rather than tailored models. This lack of customization often limits their applicability for specific needs, such as child surveillance in Kenyan schools, where cost-effectiveness is a key consideration. Additionally, while these models offer varied functionalities, such as real-time tracking or health monitoring, their effectiveness is often compromised by the specific weaknesses of the technologies employed—such as the limited range of RFID or the bulkiness of GPS modules. Moreover, many existing models do not adequately address privacy concerns, leaving sensitive information vulnerable to unauthorized access. There is also a shortage of empirical research focused on analyzing the strengths and weaknesses of these different technologies and functionalities. To address these gaps, this study aims to develop a specialized application that caters to both public and private schools in Kenya. It will also provide empirical insights into the strengths and limitations of existing technologies, offering a foundation for future research and development in the field.

Table 2

Research Gaps

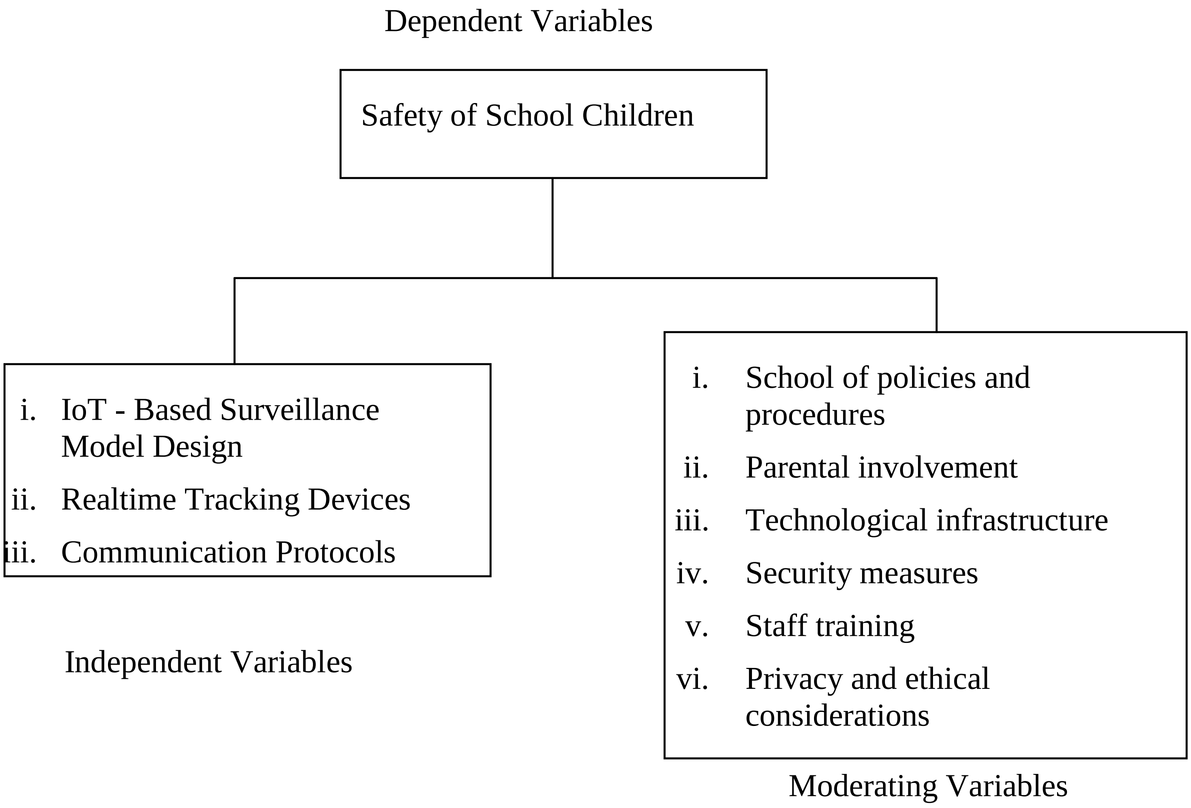
Research Gap Aspect	Details
Generalized Solutions	Existing models rely on multiple sensors and various technologies, leading to generalized rather than tailored solutions. This limits applicability to specific needs, such as school surveillance in Kenya.
Cost Considerations	Many models do not consider cost-effectiveness, which is crucial for implementing surveillance in Kenyan schools.
Technological Weaknesses	Models employ technologies that have specific drawbacks, e.g., the limited range of RFID or the bulkiness of GPS modules, which affect overall efficiency.
Privacy Concerns	Existing models often fail to sufficiently address privacy issues, leaving sensitive data vulnerable to unauthorized access.
Lack of Empirical Research	There is insufficient research focused on analyzing the strengths and weaknesses of different technologies and functionalities used in these models.
Objective of the Study	This research aims to develop a cost-effective, tailored IoT-based surveillance application for Kenyan schools and provide empirical insights into the performance of various technologies.

2.9 Conceptual Framework

The conceptual framework that directed the study was done in two stages, namely; Tier I conceptual framework and Tier II conceptual framework. Tier I conceptual framework presents the independent and moderating variables that influences IoT based surveillance model for monitoring school children while Tier II conceptual framework presents the implementation of the IoT based surveillance model for monitoring school children. Tier I and Tier II conceptual frameworks are presented in figures 13 and 14 respectively.

Figure 12

Conceptual Framework Tier I


Dependent Variables
Safety of School Children
i. IoT - Based Surveillance Model Design
ii. Realtime Tracking Devices
iii. Communication Protocols
Independent Variables
i. School of policies and procedures
ii. Parental involvement
iii. Technological infrastructure
iv. Security measures
v. Staff training
vi. Privacy and ethical considerations
Moderating Variables

Source: Author (2024)

In this conceptual framework, the dependent variable was "Safety of School Children," which represented the overall safety and security of the children in the school environment. The independent variables included:

i. **IoT-based Surveillance Model Design:** This variable referred to the design and implementation of the IoT-based surveillance model, including the selection of devices, model architecture, and data processing methods.

ii. **Real-time Tracking Devices**: This variable referred to the wearable tracking devices worn by children, which provided real-time location and movement data.

iii. **Communication Protocols**: This variable represented the protocols used for communication between the tracking devices, IoT gateway, and server, ensuring reliable and secure data transmission.

The moderating variables were external factors that could influence the relationship between the independent variables and the dependent variable. They included:

i. **School Policies and Procedures**: These policies and procedures governed the implementation and use of the IoT-based surveillance model within the school environment.

ii. **Parental Involvement**: The level of parental engagement and cooperation in utilizing the surveillance model and responding to alerts.

iii. **Technological Infrastructure**: The availability and quality of the technological infrastructure required to support the IoT-based surveillance model, such as internet connectivity and network infrastructure.

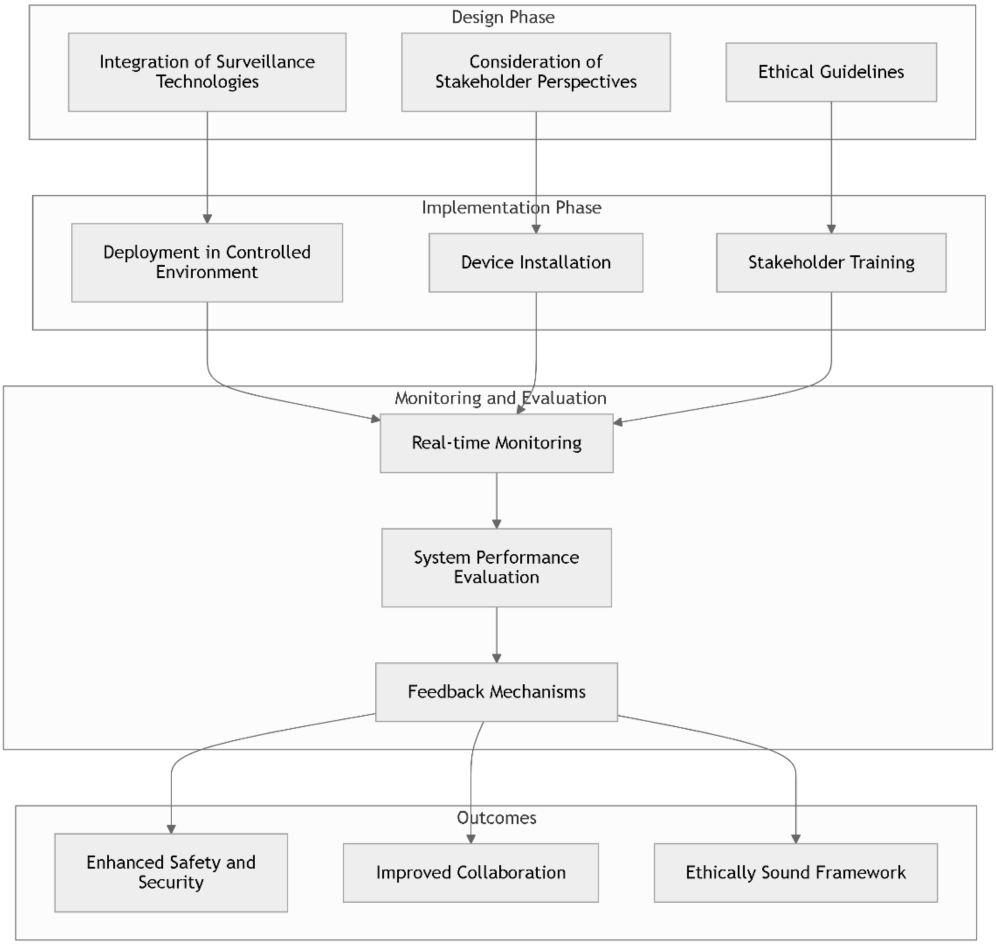
iv. **Security Measures**: The security measures implemented to protect the data and ensure the model's integrity and confidentiality.

v. **Staff Training**: The level of training provided to school staff to effectively utilize and manage the surveillance model.

vi. **Privacy and Ethical Considerations**: The ethical considerations and privacy safeguards implemented to protect the rights and privacy of the children and stakeholders involved.

Figure 13

Conceptual Model Tier II


Design Phase
Integration of Surveillance Technologies
Consideration of Stakeholder Perspectives
Ethical Guidelines
Implementation Phase
Deployment in Controlled Environment
Device Installation
Stakeholder Training
Monitoring and Evaluation
Real-time Monitoring
System Performance Evaluation
Feedback Mechanisms
Outcomes
Enhanced Safety and Security
Improved Collaboration
Ethically Sound Framework

The conceptual model for the "IoT-Based Surveillance Model for Monitoring School Children" project encompassed several key elements. Initially, in the design phase, surveillance technologies such as GPS and sensors were integrated to develop the surveillance model. Stakeholder perspectives, including those of school administrators, teachers, parents, and learners, were considered to ensure the model met user needs and ethical guidelines. During implementation, the surveillance model was deployed in a controlled environment, with devices installed and stakeholders trained on model usage. Real-time monitoring of children's movements and activities was conducted. Continuous evaluation of model performance and reliability occurred throughout the process. Feedback mechanisms, such as expert surveys and interviews, enabled stakeholders to provide input for iterative refinement of the model. Ultimately, the outcome included enhanced safety and security for school children, improved collaboration among stakeholders, and the establishment of an ethically sound surveillance framework for educational environments.

CHAPTER THREE

RESEARCH DESIGN AND METHODOLOGY

3.1 Introduction

This chapter presents the research methodology of the study, covering the following key aspects: The research design, which outlines the methods used to achieve the study's objectives; data collection, describing the procedures for gathering data; a pilot study, conducted to support the study's validity; and the ethical considerations that influenced the research process.

3.2 Research Design

This research employed the Design Science Methodology as proposed by Peffers, Tuunanen, Rothenberger & Chatterjee (2006). The design science approach ensures the systematic development and evaluation of an artifact (the IoT-based surveillance model) by following specific stages, which align with the study's objectives. The following six steps of Design Science guided this research:

i. **Problem Identification and Motivation**:

   This step involved identifying the need for an effective surveillance model to monitor children’s movement within a school setting, ensuring their safety and well-being.

ii. **Objectives for a Solution**:

   Clear objectives were established to address the identified problem and ensure that the surveillance model fulfilled the requirements for effective child monitoring. The objectives for this research were: To analyze the weaknesses of existing IoT-based surveillance models for monitoring children, to design and develop an IoT-based surveillance model for monitoring children, to implement the developed IoT-based surveillance model, to evaluate the IoT-based surveillance model.

iii. **Design and Development**:

This phase involved designing the model, selecting the IoT components, and developing software algorithms.

iv. **Evaluation**:

The model was evaluated to ensure it met predefined objectives and performed effectively in a real-world setting. The evaluation was guided by goal-based evaluation and expert surveys.

v. **Communication**:

The findings from the research were documented and communicated through the thesis to relevant stakeholders

The next section details the specific methodologies employed for each objective.

3.2.1 Systematic Literature Review

To achieve the objective (i), a systematic literature review was employed as a research methodology to analyze the effectiveness of existing models. This involved a systematic investigation relying on existing data throughout the research process. This approach entailed organizing, collating, and analyzing available data sources to draw valid research conclusions. Often referred to as secondary research or desk research, this methodology involved synthesizing data from various sources such as the internet, peer-reviewed journals, textbooks, government archives, and libraries.

By conducting a systematic literature review, it was possible to critically evaluate the strengths and weaknesses of existing IoT-based surveillance models for monitoring children. This methodology allowed for a comprehensive examination of the literature, which was helpful in designing a conceptual model for the research. It served as an initial step to identify areas that required further investigation and informed the subsequent research stages.

In summary, the systematic literature review methodology provided a foundation for objective (i) by utilizing existing data sources to assess the effectiveness of current models, identify gaps, and guide the development of a conceptual model for the research.

3.2.2 Proof of Concept

Proof of Concept (PoC) refers to the realization of an idea or method to demonstrate its feasibility, or a demonstration in principle to verify the practical potential of a concept (Cedarbaum, 2018). In this thesis, the PoC approach was employed to demonstrate and verify the practical potential of the IoT-based surveillance model for monitoring children. The goal was to validate the model's feasibility under controlled conditions, ensuring it could meet the intended objectives before full-scale deployment.

As proposed by Gordon (2014), the model was designed to be simpler and more cost-effective than the full implementation, enabling the researcher to conduct experiments and tests that would not have been feasible in a real school environment due to logistical or financial constraints (Rezaee, 2018). The PoC approach ensured that resources were allocated efficiently, preventing unnecessary investment in solutions that might not be technologically viable or aligned with real-world requirements.

By using this approach, the research provided tangible evidence of the model's value and effectiveness, increasing the confidence of project stakeholders in the concept. It also allowed for early identification and resolution of potential issues, thereby increasing the likelihood of success when the model transitions from PoC to full implementation.

3.2.2.1 Designing an IoT-based Surveillance Model

The design phase of the IoT-based surveillance model followed the Design Science Research Methodology (DSRM). This phase focused on translating the insights gained from the systematic literature review (objective 1) into a viable model for monitoring children. A proof of concept (PoC) approach was employed to demonstrate that the model's fundamental design and architecture were feasible in a school setting. The following was the process of design;

i. **Identifying Requirements**

Requirements were gathered based on the weaknesses identified from the systematic literature review. Key functionalities, model constraints, and user needs were outlined to ensure the design addressed the intended problem effectively.

ii. **Designing the Architecture**

A hybrid architecture was developed to outline the interaction between hardware components, software elements, and communication channels. The architecture defined how different model components would communicate, manage data, and interact with users to achieve the monitoring objectives.

iii. **Developing Key Modules**

Essential model modules, such as registration, tracking, and notification modules, were conceptualized to ensure smooth and effective operation. Data flow and communication between these modules were planned to facilitate seamless interaction and ensure that users can access and manage model functions efficiently.

APIs were incorporated to enable communication between different components, ensuring modularity and scalability within the model.

iv. Validation through Proof of Concept

The PoC focused on testing the critical functionalities of the design to validate their feasibility and effectiveness. This initial testing phase helped to identify potential issues and provided an opportunity to refine the model's design based on early feedback.

The PoC allowed the research to confirm the viability of the design and ensured a smooth transition to the implementation phase, minimizing risks and optimizing the model's overall performance. This design phase ensured that the IoT-based surveillance model was aligned with the requirements identified and prepared for the implementation phase under objective 3.

3.2.2.2 Implementation of the Model

The implementation of the surveillance model built upon the validated design from the PoC phase. It involved assembling the hardware, configuring the software, and integrating all model components into a functioning model to ensure it could operate effectively in a real environment. The following outlines process of implementation:=

i. Integration of Hardware and Software

Hardware components were assembled, and software interfaces were configured to facilitate interaction between the model's modules and users. Communication protocols were established to enable data exchange across the model.

ii. Software Configuration and API Integration

APIs were integrated to ensure smooth data flow and interaction between model components. Backend services were configured to manage data and ensure the consistency of information throughout the model.

iii. Testing in a Controlled Environment

The model was deployed and tested in a controlled environment to monitor its functionality and identify any technical or operational issues. Various scenarios were simulated to ensure the model performed as expected under real-world conditions.

iv. Feedback and Refinement

Feedback was collected from stakeholders involved in the testing phase. Identified issues were resolved, and necessary adjustments were made to optimize the model's performance and ensure it met the intended objectives.

3.2.4 Evaluation of the Model

The goal-based evaluation methodology focuses on assessing whether an initiative or model achieves its predefined objectives and goals. In the context of digital transformation within organizations, this approach emphasizes aligning technological implementations with strategic business outcomes. It evaluates not just the functionality of deployed technologies, but also their effectiveness in meeting operational or strategic goals, such as enhancing productivity, customer engagement, or innovation.

Aldoseri, Al-Khalifa, & Hamouda (2024) applied this methodology to measure how organizations transition to AI-based digital solutions, ensuring that technological adoption aligns with business needs. The framework involves setting clear performance metrics, tracking progress, and identifying gaps between expected and actual outcomes. This evaluation method is crucial for decision-making, as it provides insights into the areas requiring improvement.

The modules to be evaluated in the IoT surveillance model were:

i. **Student Registration**: This module was evaluated to determine its effectiveness in capturing and securely storing learner information. The goal was to ensure that all necessary learner details—such as name, grade, contact information, and associated GPS device—could be registered accurately and without errors. Additionally, the evaluation looked into the model's ability to manage multiple users, integrate seamlessly with the backend database, and verify data integrity during registration processes.

ii. **Device Registration:** The device registration process was tested to ensure that GPS tracking devices could be accurately registered and linked to specific learners. The goal was to verify whether the model could handle multiple devices, track their activity status, and associate the correct devices with the registered learners. The evaluation also checked for potential technical issues related to device connectivity, GPS accuracy, and model responsiveness during device registration.

iii. **Real-Time Tracking/Device List**: The real-time tracking functionality was crucial for evaluating the model's core purpose—monitoring children. This feature's performance was assessed in terms of accuracy, responsiveness, and the ability to display active and inactive devices in real-time. The "Device List" module provided a comprehensive view of all devices connected to the model, ensuring that both active and non-active devices were displayed correctly. The evaluation also focused on ensuring that the tracking model could effectively update learner locations as they moved within or outside the school’s designated boundaries.

iv. **Alert Notification**: The alert notification module was evaluated to confirm its effectiveness in sending timely notifications to parents and administrators when specific events occurred. These events included a learner arriving at school, leaving the school premises. The goal was to assess the reliability and speed of the GSM-based alert system, ensuring that notifications were accurately triggered and delivered to the correct recipients.

Additionally, expert survey was conducted to validate the IoT-based surveillance model for monitoring children. The survey included 15 participants, consisting of IoT experts, school administrators. These participants were selected for their specialized knowledge and relevant experience, enabling them to provide informed feedback on the model's validity. To further assess the performance of the model, four software evaluation metrics were employed: usability, reliability, efficiency, and functionality. Data collection for these metrics was achieved through six structured survey questions, specifically designed to capture the necessary responses for evaluating the model. The survey responses provided valuable insights into the model's user-friendliness, consistency of performance, efficiency in operation, and the extent to which it fulfilled its intended functions. This approach ensured a comprehensive evaluation, combining qualitative insights from experts with quantitative assessments of the model's software performance.

3.4 Data Collection Procedures

In this research, data collection was approached through multiple methodologies to ensure comprehensive insights for developing the IoT-based surveillance model. First, a systematic literature review was conducted to analyze existing models, allowing for an in-depth understanding of the current state of IoT-based child monitoring models, their strengths, weaknesses, and areas requiring improvement.

Additionally, expert surveys and interviews were conducted with IoT experts and school administrators to gather professional and practical perspectives on the proposed model. This helped in validating key components of the model and identifying potential challenges in real-world implementations.

Furthermore, the model was tested in a controlled environment within schools. This phase involved deploying the model and monitoring its functionality in a real-world setting, allowing for the collection of feedback from end-users. This hands-on testing provided valuable insights into the usability, reliability, and efficiency of the model. The combination of these data collection methods enabled a well-rounded analysis and improvement of the proposed model.

3.5 Pilot Study

A pilot study was conducted which aimed to investigate the potential need for an IoT-based surveillance model to monitor school children in Nakuru municipality. The study was done among eight schools, including four private and four public schools. Below is a table showing the results of the survey;

Table 3

Pilot Study Results

Research Question	Yes response	Yes %	No Response	No %	Total Reponses
Does the school have a system to alert parents when their child arrives at school in the morning and when they exit at the end of the school day?	1	12.5%	7	87.5%	8
If yes, is the system able to trace learner movement and report any cases of sneaking?	0	0%	8	100%	8
Is there need for a surveillance system to monitor children at school?	8	100%	0	0%	8

The results of the study showed that there is a potential need for a surveillance model to monitor children at school, as all respondents answered "yes" to the question of whether such a model is necessary. This finding is consistent with previous research that has identified the importance of surveillance models in enhancing the safety and security of school environments (Gallardo-Echenique, Mar-Molinero, Bullock-Rest & Villarreal-Rosas, 2018; Kim *et al.*, 2019).

Regarding the current state of existing models, the study found that one private school was using RFID technology to mark learner movement in and out of school, while the remaining schools did not have a model in place to alert parents when their child arrives at and leaves school. This suggests that there is a potential gap in the current models used by schools in Nakuru municipality to ensure the safety and security of their learners.

Furthermore, the study found that none of the schools surveyed had a model in place to trace learner movement and report cases of sneaking. This finding is consistent with previous research that has identified the potential of IoT-based models to address this issue by enabling real-time tracking and monitoring of learner movement within school premises (Kim, Park, Park, Kim & Jung, 2019).

Overall, these results suggest that there is a potential need for an IoT-based surveillance model to monitor school children in Nakuru municipality, particularly with regard to ensuring the safety and security of learners and addressing cases of sneaking.

3.6 Ethical Considerations

Potential risks for this study included compliance with legal regulations and ethical guidelines related to the surveillance of minors, data protection, and informed consent procedures. By limiting access to surveillance data to authorized personnel only and establishing protocols for data sharing and dissemination, this risk was managed. The management of data generated by the surveillance model was a critical aspect. Access to such sensitive information was handled with utmost care and responsibility to prevent severe consequences if it fell into the wrong hands. This risk was mitigated by implementing strong data encryption and secure data storage practices to protect against unauthorized access or breaches. Data no longer needed for research purposes was disposed of upon the conclusion of the project and any associated follow-up studies.

The researcher conducted informational sessions or meetings with parents and guardians to explain the study's objectives, methodologies, potential risks, and anticipated benefits. These sessions provided ample opportunities for questions and addressed any concerns. Additionally, the researcher developed comprehensive consent forms for parents and guardians that clearly articulated the study's purpose, procedures, data management protocols, and privacy protections, written in clear, non-technical language to ensure understanding.

To conduct the research, official letters were sought from the Institute of Post Graduate Studies at Kabarak University and the National Commission for Science, Technology, and Innovation (NACOSTI). Proper acknowledgment was given to the sources used for data gathering. Whenever handling data required consent, the researcher sought approval from the data sources.

CHAPTER FOUR

DATA PRESENTATION, ANALYSIS AND DISCUSSION

4.1 Introduction

This chapter presents the study's findings in alignment with the research objectives outlined in Section 1.4 and 1.5. It includes an analysis of existing IoT surveillance models for monitoring children, the design and development of a concept IoT surveillance model designed to assist schools, and an evaluation report on the functionality of the concept for monitoring children.

4.2 Existing Models Used in Surveillance of children in Schools

This section outlines the findings related to the first objective of the study, which focused on examining the weaknesses of current models. The IoT surveillance models discussed in Section 2.4 of this study were reviewed.

4.2.1 Weaknesses of the Existing Models

To assess the value of a technological product, its purpose must be clearly defined. Therefore, justifying the development of a new IoT surveillance model requires evidence of inefficiencies in current real-world alternatives. The next two sub-sections will examine these shortcomings from two perspectives: the limitations of specific technologies used and the evaluation of selected key assessment criteria. A generalized list of challenges faced by existing IoT-based child monitoring models was derived from the *table 1* in section 2.4.8 showing specific model weaknesses. This list highlights technological limitations, integration issues, and practical challenges in real-world applications.

i. **Technological Constraints**:

Many of the existing models rely heavily on technologies like RFID, GPS, and GSM, each with distinct limitations. RFID models, while useful for attendance and proximity tracking, are often hindered by the need for close-range scanning, which can result in bottlenecks during the check-in process and may fail to capture real-time data if tags are not scanned properly. Furthermore, RFID models are prone to unauthorized access issues due to the simplicity of reading tags, compromising data security.

GPS-based models provide real-time location tracking but come with drawbacks like high power consumption, which can drain device batteries quickly, especially when used for continuous tracking. The reliance on GPS alone can also lead to inaccuracies in indoor environments where signal strength is low.

GSM-based models facilitate communication for alerts but depend on network coverage. In areas with weak signals, data transmission delays or failures can occur, reducing the model's reliability in emergency situations. This dependency makes GSM models less effective in regions with poor connectivity, which can be a significant barrier in rural or less-developed school areas.

i. **Integration and Customization Issues**:

A common challenge across these models is the complexity involved in integrating multiple sensors and modules, such as temperature sensors, fingerprint readers, and alcohol detection sensors. This complexity can result in higher costs, increased maintenance needs, and difficulties in customizing models to meet the specific requirements of different school environments.

The existing models often lack scalability and flexibility, which is crucial for adapting to the varying sizes and needs of schools. For instance, models designed for small-scale implementations may not effectively manage data and user load when applied to larger institutions. Additionally, the lack of seamless integration between different components can lead to gaps in functionality, such as delays in data processing or failure in triggering alerts. There is also an emphasis on limited data security and privacy measures, particularly in models that track students' locations and activities. With sensitive information like real-time location data being collected, insufficient encryption and data handling protocols can lead to potential breaches, raising concerns among stakeholders like parents and school administrators.

The identified weaknesses suggest that while existing IoT-based child monitoring models offer valuable functionalities, they face significant challenges in ensuring reliability, data security, and adaptability to diverse school settings. Addressing these gaps requires a holistic approach that integrates robust communication technologies, enhanced data protection measures, and cost-effective solutions tailored for different school environments.

4.2.1.2 Weaknesses Analyzed Using Assessment Criteria

The weaknesses of existing IoT child monitoring models was evaluated using key assessment criterion. The criteria set for evaluating these existing models encompass: the accuracy and reliability of detection, model scalability, flexibility in integration, data security, cost-effectiveness in both initial setup and ongoing operations, ease of use and maintenance and finally the model's functionality, including its feature set and overall performance, must be thoroughly evaluated. The table below presents the key of assessment criteria:

Table 4

Key of Assessment Criteria

Assessment Criteria	Definition	Traceability
Accuracy and reliability	Description of how accurately and reliably the model functions.	A
Model Scalability	Explanation of the model's ability to handle increased scale.	B
Flexibility in integration	Details on the model's adaptability to integrate with other systems.	C
Data security	Measures taken to secure sensitive data within the model.	D
Cost-effectiveness	Evaluation of the model’s cost relative to its benefits.	E
Ease of use and maintenance	Assessment of how user-friendly and maintainable the model is.	F
Model's functionality, including feature set	Overview of the model’s features and overall performance.	G

Table 5

Existing Models Assessment

Model	A	B	C	D	E	F	G
(El Mrabet & Ait Moussa, 2020) IoT-School Attendance Model Using RFID Technology	✓	×	✓	×	×	✓	✓
Intelligent and Secured Tracking Model for Monitoring School Bus (Ahmed et al, 2019)	×	×	×	×	×	×	×
IoT-Based Touch-Free Attendance Model by Tamilselvan *et al.* (2021)	✓	×	✓	×	✓	✓	✓
Intelligent School Bus Monitoring and Reporting Model via IoT Ajayalakshmi *et al.* (2021)	×	×	×	×	×	×	✓
Student Monitoring Model for Boarding and Leaving Bus (Maturkar et al, 2020)	×	×	×	×	×	×	✓
Tesfaye’s (2020) model for real-time child tracking using an SMS-based solution	×	×	×	×	✓	✓	✓
Child Safety Monitoring Model Based on IoT Senthamilarasi *et al.* (2019)	×	×	×	×	×	×	✓

Table 6

Weaknesses of IoT Child Monitoring Models Evaluated Using Assessment Criteria

Model	Accuracy and Reliability	Scalability	Flexibility in Integration	Data Security	Cost-Effectiveness	Ease of Use and Maintenance	System Functionality
IoT-Based Attendance Model	Weak RFID tag reliability	Limited hardware scalability	Hardware-dependent	Insufficient data protection	High hardware costs	Requires technical expertise	Dependent on email/SMS
Intelligent Tracking Model for School Bus	Sensitive fingerprint sensors	Struggles with high data volumes	Challenges with system integration	Lacks robust data security	High setup and maintenance costs	Complex user training	Component incompatibility issues
IoT-Based Touch-Free Attendance Model (ITAS)	RFID and sensor dependency	High hardware needs	Limited to specific hardware	Potential data transmission risks	Cost-effective but high maintenance	Regular calibration needed	Internet dependency
Intelligent School Bus Monitoring Model	Potential sensor malfunctions	Data overload risk	Compatibility issues	Lacks robust data encryption	Expensive for large-scale setups	Complex due to multiple technologies	Prone to data overload
Student Monitoring for Bus Boarding /Leaving	Risk of RFID card loss	Limited expansion potential	Limited flexibility with new tech	Centralized data security risks	High setup and maintenance costs	Challenging maintenance	High dependency on each component
IoT Children Monitoring Model (Ethiopia)	GPS/GSM inaccuracy	SMS limitations for larger scale	Limited by specific tech choices	Lacks SMS encryption	Low-cost setup	Simple user operation	Impacted by signal issues
Child Safety Monitoring Model	Sensor calibration challenges	Struggles with large deployments	Difficult to adapt to new tech	SMS data not encrypted	High initial setup cost	Specialized training required	Needs system updates

Generalized Weaknesses of Models According to Assessment Criteria

The table below shows a summary of the analysis of the weaknesses of models according to assessment criterion.

Table 7

Generalized Weaknesses of Models According to Assessment Criteria

Assessment Criterion	Generalized Weaknesses
Accuracy and Reliability	Models often rely on RFID tags or biometric sensors, which can fail due to environmental interference, damaged hardware, or improper calibration. This results in missed detections or false readings.
Model Scalability	Many models struggle to scale effectively, especially when integrating multiple devices and handling increased data. Larger deployments introduce complexity in infrastructure and risk performance degradation.
Flexibility in Integration	Integration challenges arise from the use of proprietary or specific technologies (e.g., RFID, GSM). These technologies often limit interoperability with existing school management models or future upgrades.
Data Security	Weak encryption, lack of robust authentication protocols, and reliance on SMS for data transmission expose sensitive information to risks such as interception, unauthorized access, and data breaches.
Cost-Effectiveness	Initial setup and maintenance costs are high due to hardware requirements (e.g., RFID readers, sensors, microcontrollers). These costs can limit adoption in budget-constrained environments.
Ease of Use and Maintenance	Models with multiple components require ongoing maintenance, calibration, and technical expertise, which can burden users, especially in schools with limited staff training and resources.
Model Functionality and Performance	Functionality depends on the seamless operation of various components. Issues such as poor network connectivity, incompatible modules, or device malfunctions negatively impact performance.

4.2.2.1 Design Recommendations to Counter Weaknesses of Existing Models

To address the identified weaknesses and enhance the design of the proposed IoT-based surveillance model, several key recommendations should be considered.

4.2.2.2 Design Recommendations to Counter Observed Technological Weaknesses of Existing Models

Based on the identified weaknesses of existing IoT child monitoring models, the following design recommendations aim to address the technological constraints and integration challenges to create a more effective, reliable, and secure monitoring model:

i. **Enhanced Technological Integration:**

Hybrid Technology Approach: To overcome the limitations of individual technologies like RFID, GPS, and GSM, a hybrid model should be implemented. For instance, combining GPS with Bluetooth Low Energy (BLE) can provide more accurate indoor positioning while reducing power consumption. BLE is effective for short-range indoor tracking and can supplement GPS, addressing the issue of signal loss in indoor environments. This hybrid approach can ensure continuous location tracking with minimal power usage.

Advanced RFID Security Measures: To mitigate security concerns with RFID, integrating encryption protocols for data communication between tags and readers is essential. Advanced encryption standards (AES) or cryptographic hashing methods can be employed to secure data transfer, reducing the risk of unauthorized access. Additionally, integrating multi-factor authentication (MFA) for RFID models can enhance security by requiring both the RFID scan and an additional verification step before access to student data is granted.

Power Optimization Strategies: For GPS-based models, using low-power GPS modules and implementing power-saving algorithms can extend battery life. Techniques like duty-cycling, where the GPS is activated at set intervals rather than running continuously, can help conserve energy. This is particularly beneficial for real-time tracking in schools, where continuous updates may not always be necessary.

ii. Improved Data Security and Privacy:

End-to-End Encryption: Implement end-to-end encryption for data transmissions between IoT devices and central servers. This will ensure that sensitive data, such as students' location information, is protected throughout the communication process, even in the case of interception. Using secure communication protocols like Transport Layer Security (TLS) can further enhance the confidentiality of data exchanges.

Data Anonymization Techniques: To address privacy concerns, incorporating data anonymization techniques can prevent unauthorized access to personally identifiable information (PII). This involves storing location data without directly linking it to specific student identities until it is needed for authorized access. It can be particularly useful when generating reports or monitoring movement patterns, allowing schools to maintain privacy while still gaining valuable insights.

Role-Based Access Control (RBAC): Implementing RBAC ensures that only authorized personnel (e.g., teachers, school administrators) have access to specific data related to students. This limits access to sensitive information, reducing the risk of data breaches. The model can be designed to provide different access levels based on user roles, improving the overall data management process.

iii. Simplified Model Integration and Customization:

Modular Design Architecture: A modular design can simplify the integration of different sensors and modules, making the model more adaptable to specific needs. By creating a model where each module (e.g., attendance tracking, location monitoring, temperature sensors) operates independently but communicates through a unified interface, customization becomes easier. Schools can add or remove functionalities based on their requirements without disrupting the entire model.

Scalability through Cloud-Based Solutions: Leveraging cloud computing for data storage and processing allows the model to be more scalable. Cloud-based solutions can handle larger volumes of data and provide real-time analytics, making the model suitable for schools of different sizes. Schools can also benefit from reduced infrastructure costs since cloud providers manage server maintenance and upgrades.

User-Friendly Customization Interface: Developing an intuitive user interface that allows schools to configure model settings, such as alert thresholds or tracking intervals, can make the technology more accessible. This can reduce the dependency on technical expertise for model setup and maintenance, making it easier for schools to adjust the model according to their specific needs and resources.

iv. Reliable Communication Mechanisms

Multi-Network Support: To address the limitations of GSM networks, the model should support multiple communication protocols, such as Wi-Fi and LoRaWAN (Long Range Wide Area Network). LoRaWAN is suitable for areas with poor GSM coverage due to its long-range capabilities and low power consumption, making it an effective alternative for rural schools. This redundancy ensures that critical alerts and data transmissions are not lost due to network failure.

Local Data Storage with Cloud Sync: Incorporating local data storage on the monitoring device itself, with periodic syncing to cloud servers, can enhance reliability. This ensures that in the event of network failure, data is not lost and can be synchronized once connectivity is restored. It also allows the model to continue functioning independently during network downtime, maintaining continuous monitoring.

4.2.2.3 Design Recommendations to Counter Weaknesses based on Evaluation of Existing IoT Child Monitoring Models using Key Assessment Criteria

i. Firstly, to improve detection accuracy and reliability, the design should incorporate multi-sensor fusion techniques along with error-checking algorithms. This approach will enhance the model's capability to accurately monitor and track children, minimizing false positives and negatives.

ii. Secondly, data security and privacy are paramount. The model should implement end-to-end encryption and robust access control mechanisms to ensure secure handling and storage of sensitive information. Compliance with privacy regulations must be a core design principle to protect personal data and maintain trust.

iii. Thirdly, cost-effectiveness should be a priority. By selecting affordable hardware and software solutions, exploring bulk procurement discounts, and leveraging open-source software where applicable, the model can balance performance with budget constraints. This approach will ensure that the model remains economically viable while meeting operational needs.

iv. Fourthly, the model must prioritize ease of use and maintenance. Developing a user-friendly interface and incorporating self-diagnosis features will facilitate straightforward operation and minimize downtime. Comprehensive training for users will further support efficient model maintenance and usage.

v. Lastly, the functionality and feature set of the model should be comprehensive and tailored to the school environment. This includes incorporating real-time tracking, emergency alerts. A well-rounded feature set will ensure that the model not only meets basic requirements but also provides valuable insights and supports proactive management of learner safety.

By addressing these recommendations, the proposed model should be better equipped to meet the needs of schools and effectively enhance the safety and monitoring of children.

4.3 Design of IoT Surveillance Model for Monitoring Children

The second objective of this study was to design an IoT-based surveillance model for monitoring children using the design science methodology. This section presents the results of that design process. The model was designed in alignment with the design science process, ensuring it effectively addresses the identified problem and meets the project's goals. The section also details the functional, non-functional, and technical requirements of the model.

4.3.1 Defining Model Functional Requirements

4.3.1.1 User Roles

The implementation detailed in this thesis focuses on three primary user groups: School Administrators, and Parents. To capture the functional requirements for each user type, user stories were developed based on feedback from school administrators and IT experts and are presented in the table below. Each user type is then described in greater detail. The user stories and requirements were intentionally simplified to their essential components, while ensuring the proof of concept remains both usable and secure. *Table 9* describes the roles of various users in the model.

Table 8

User Stories Defining Roles

Archetypical User	Role	Primary Responsibility
School Administrators	- Model Management - Access Control - Monitoring and Reporting	▪ Manage user roles and permissions ▪ Oversee real-time tracking of learners ▪ Generate attendance and security reports ▪ Respond to alerts in emergencies
Parents	-Child Safety Monitoring - Communication	▪ Receive alerts for boundary breaches and emergencies ▪ Inform school management in case of planned absence of learner.
Teachers	- Classroom Management - Learner Safety Monitoring	▪ Track classroom attendance ▪ Monitor and track learner movements within and outside the classroom area

4.3.1.2 Functional Overview of Modules

The Table 10 outlines the key modules of the IoT-based surveillance model. Each module is assigned a name, along with a description of its functionality and the purpose it serves in achieving the overall goal of monitoring children efficiently.

Table 9

Model Functional Overview

Functionality Name	Description	Purpose
User Registration	To facilitate the registration of users (admins and normal users) in the model, ensuring secure access and management of user accounts.	Validates user credentials during registration. Stores user information securely database. Provides feedback on registration success or failure
Student Registration	Securely captures and stores learner details such as name, contact information, grade, and GPS device association within the model's database.	Enables the model to uniquely identify each learner and associate them with a tracking device, laying the foundation for the rest of the model's functions.
Device Registration	To register and link GPS tracking devices with specific learners for real-time monitoring purposes.	Ensures accurate linkage of GPS devices to learners for real-time tracking.
Real-Time Student Tracking (Device List)	The model tracks learners' movements using GPS technology, providing real-time location updates.	To provides device coordinates and status (active/inactive) and so monitoring their whereabouts within and outside the school.
Alert system	Sends automated alerts via SMS notifications when a learner crosses predefined boundaries or during emergencies.	To notify parents and administrators of boundary breaches, security threats.

4.3.1.3 Data Management

The following list describes how data will be managed by the model;

i. **Data Collection**

Data collection involves gathering GPS data, user information, device status, and alerts.

The GPS module captures real-time location data from learners, which is processed by the microcontroller and transmitted to the server over Wi-Fi. In emergency situations, GSM modules enable the model to send SMS-based alerts directly to parents and administrators.

The model collects data from various sources, including the IoT layer devices: a GPS module, a GSM module, and a microcontroller. Additionally, user inputs from the client-side web application, such as registration and configuration details, are considered as data sources.

The collected data includes the following types: GPS coordinates, Timestamps, Device IDs, User registration information, Alert messages.

ii. **Data Processing**:

The microcontroller processes GPS data to determine the learner's location. The server receives and processes incoming location data, analyzing movement patterns and identifying any boundary breaches. Additionally, the server's API handles user requests from the web application, including registration and query submissions. The processed data is used for real-time tracking, determining the current status of learners, and generating necessary alerts.

iii. **Data Storage**:

The model stores data on a MongoDB database, which is located in the server-side block of the architecture. This database houses various types of data, including user information (admins, parents, and students), device information, real-time location data logs, alert history, and. The database is continuously updated as new data is received from the IoT layer through the server. Data is stored in JSON format, which aligns with MongoDB's schema-less structure and ensures consistency.

To manage data retention, GPS tracking logs are retained for a specified period, such as 90 days. This allows for historical tracking and reporting. Meanwhile, alert logs and user data are kept for a longer duration to maintain accountability and support auditing purposes.

iv. **Data Retrieval**

Data is retrieved from the MongoDB database through API endpoints. This allows for programmatic access to the stored information. To ensure data security and privacy, data retrieval is role-based. This means that users can only access data that is relevant to their specific permissions, preventing unauthorized access.

4.3.1.4 Notification System

The following list describes how alerts will be setup in the model;

i. **Alert Generation:** Alerts are triggered based on predefined conditions such as boundary breaches. Alerts are generated on the server-side based on certain conditions, like device inactivity.

ii. **Alert Prioritization:** Alerts are categorized based on their urgency to ensure critical issues receive immediate attention. High-priority alerts include emergencies like unauthorized exits or device malfunctions. Medium-priority alerts encompass scheduled reports or low battery notifications. Low-priority alerts are routine reminders, such as daily attendance notifications.

iii. **Notification Channels:** The model utilizes multiple channels to notify stakeholders: SMS Alerts are sent directly to parents for critical events. Email Notifications are generated by the API server and sent to registered email addresses. Web App Alerts are displayed on the web application for administrators.

iv. **Frequency Control:** Users can customize their notification preferences by enabling or disabling SMS alerts to avoid overwhelming notifications.

v. **Emergency Alerts:** Emergency alerts are automatically triggered in significant safety concerns or manually by administrators. These alerts are always sent via SMS and prominently displayed on the admin dashboard for immediate action.

vi. **Alert Logging and History:** All alerts, regardless of channel, are logged in the MongoDB database. This allows administrators to view historical alert data, track response times, and analyze recurring safety concerns. Alert logs are stored for a defined period to support incident analysis and reporting.

4.3.1.5 User Interaction and Interface Requirements

The model offers a web application accessible to all user roles, including administrators and normal users. The interface is designed to be responsive, ensuring compatibility with both desktop and mobile devices. The layout includes dashboards, forms, data tables, and notification displays for an intuitive user experience. The following are key features of the user interface provided by the model;

i. **Dashboard:** Displays an overview of model modules, including students, devices, and users.

ii. **User Registration and Login**: The model provides a secure registration process for new users, including admins, parents, and normal users. A login page with username/email and password fields ensures secure access. Users can reset their passwords via email verification.

iii. **Device Registration:** The model provides a registration process for devices and a way of linking a device to the learner.

iv. **Real-Time Tracking Interface (Student List):** The tracking page displays a list of registered devices, indicating their online or offline status. GPS coordinates from the devices are processed to show each learner's current position. Users can click on a student's location marker to view more details, such as last updated time, movement history, and active status.

4.3.1.6 Integration with External Systems: Hardware Integration

The following describes how the hardware components will be integrated within the model;

i. **GPS Module Integration:** The GPS module is integrated into the IoT layer to provide real-time location tracking of students. It captures latitude and longitude coordinates and sends them to the microcontroller for further processing. The GPS module continuously monitors the geographic coordinates of a student's location. The captured data is transmitted to the microcontroller through a UART connection.

ii. **GSM Module Integration:** The GSM module is responsible for sending SMS alerts to parents and administrators. It allows for communication through a cellular network. The microcontroller triggers the GSM module when an alert condition is met. The microcontroller sends a command to the GSM module, which then sends an SMS to predefined contact numbers.

iii. **Microcontroller Integration:** The microcontroller serves as the central processing unit for handling communication between IoT hardware components (GPS, GSM) and the server. It is responsible for collecting data, processing it, and ensuring connectivity to the server through Wi-Fi. The microcontroller receives GPS data, processes it, and sends it to the server over Wi-Fi. It also handles incoming commands from the server for actions like starting/stopping tracking or sending specific alerts.

iv. **Wi-Fi Connectivity and Server Integration:** The microcontroller connects to the school's Wi-Fi network, enabling communication with the API server for real-time data transfer. It serves as a bridge between the hardware components (GPS, GSM) and the server-side processing. The microcontroller sends GPS coordinates and other relevant information to the server using REST API calls. The server processes this data, stores it in the MongoDB database, and makes it available to the web app for users to access.

4.3.1.7 Device Management

The following describes how devices will be managed in the model;

i. **Device Registration and Configuration:** Each IoT device must be registered within the model before it can be used for monitoring purposes. Device registration links the physical device to a specific learner profile or user within the model. During registration, device details (e.g., unique device ID, type, student association) are stored in the database through the API server. The web app interface allows administrators to input and update the registration details of each device.

ii. **Device Status Monitoring:** The model monitors the status of each registered device to ensure they are functioning correctly and remain connected to the network. This includes monitoring whether a device is active or inactive. The microcontroller periodically sends status updates of connected devices (e.g., GPS signal strength, GSM signal availability) to the API server. The server processes this information and stores it in the MongoDB database, which is accessible via the web app for real-time status updates.

iii. **Device Association with Users:** Devices need to be associated with specific learners for accurate tracking and alerting. This ensures that data collected by a device is linked to the correct user within the model. When a new device (e.g., a GPS tracker) is registered, the admin uses the web app to link it to a specific student's profile. The MongoDB database stores these associations, allowing the server to reference them when processing location updates or generating alerts.

iv. **Data Encryption and Security:** The model ensures that all data exchanged between devices and the server is encrypted to maintain security and privacy. This includes data transmitted from GPS modules, GSM alerts, and status updates from microcontrollers. The microcontroller encrypts GPS data before sending it over Wi-Fi to the API server. The server decrypts this data and stores it in the database, ensuring that data is protected throughout its journey.

v. **Device Data Synchronization:** The model ensures that all device data is synchronized between the IoT devices, the API server, and the database. This is crucial for maintaining accurate real-time location tracking and attendance records. The microcontroller sends regular updates containing the latest GPS coordinates and device status to the server. The server ensures that this data is synchronized with the MongoDB database, making it available for real-time queries by the web app.

4.3.1.8 Security Functions

The following are the security requirements in the model design;

i. **User Authentication and Authorization:** The model implements robust authentication mechanisms to ensure that only authorized users have access to the model. Role-based access control (RBAC) ensures that users have access only to the functionalities relevant to their role. Users access the web app interface and log in with their credentials, which are verified through the API server against stored records in the MongoDB database. Upon successful login, the server generates a session token that is used for subsequent interactions.

ii. **Data Encryption:** Data encryption is implemented to protect sensitive information during transmission between the IoT devices, API server, and database. This includes GPS coordinates, device status updates, and alert messages sent via the GSM module. The microcontroller encrypts data before sending it to the API server. The API server decrypts incoming data for processing and stores the encrypted data in the MongoDB database.

iii. **Data Integrity and Validation:** The model ensures that data received from IoT devices is accurate and has not been tampered with, maintaining the integrity of student tracking information. This includes validating the consistency of GPS data and timestamps for location tracking. When the microcontroller sends GPS coordinates to the API server, the server validates the data for consistency and accuracy before storing it in the database.

iv. **Secure Data Backup and Recovery:** The model implements data backup and recovery mechanisms to ensure data availability in case of hardware failures, database corruption, or security incidents. Regular backups of student data, device logs, and user records are essential for maintaining service continuity. The database is configured to perform automated backups at scheduled intervals. Backup data is stored securely on an off-site server to prevent loss due to local hardware failure.

4.3.2 Defining Model Non-functional Requirements

Table 10

Model Function Overview

Requirement Category	Specific Requirement	Description
Performance	Response Time	The web app should respond to user actions (e.g., logging in, viewing student locations) within 2 seconds.
	Data Transmission Latency	Data from the microcontroller (e.g., GPS coordinates) to the API server and then to the MongoDB database should have a maximum latency of 1 second.
	Alert Delivery Time	Alerts sent via the GSM module (e.g., SMS notifications for boundary breaches) should reach the parents or admins within 5 seconds.
	Concurrent User Handling	The model should support up to 30 concurrent users accessing the web app without degradation in performance.
Reliability and Availability	Model Uptime	The IoT-based surveillance model should be available 99.5% of the time, allowing minimal downtime for maintenance.
	Data Backup Frequency	Automatic backups of the MongoDB database should occur every 12 hours to ensure data recovery in case of failures.
	Fault Tolerance	The microcontroller and GSM module should continue functioning even if the web app or server experiences downtime.
	Data Recovery	In the event of a failure, the model should be able to restore data to within the last backup period (12 hours).
Scalability	Scalable Architecture	The API server and MongoDB database should support horizontal scaling to accommodate additional devices and users.
	Device Scalability	The model should be capable of integrating up to 500 GPS devices without compromising data transmission efficiency.
	User Scalability	The web app should be capable of supporting an increase in registered users (parents, admins) without requiring significant architectural changes.
Usability	User Interface Design	The web app should follow intuitive UI design principles, allowing users to view real-time location data, manage devices, and respond to alerts with minimal training.
	Mobile Compatibility	The web app should be fully responsive and accessible on mobile devices, as many parents and admins may access it through smartphones.
Security	Data Encryption	All data in transit between the microcontroller, API server, and MongoDB database should be encrypted using TLS.

	Secure Data Storage	Data stored in the database should be encrypted at rest to prevent unauthorized access.
	Regular Security Audits	Conduct security audits of the web app, API server, and IoT devices every six months to identify and mitigate potential vulnerabilities.
	Compliance	The model should comply with data protection regulations such as GDPR (General Data Protection Regulation) for handling user and student data.
Maintainability	Modular Code Structure	The web app, API server, and device firmware should follow a modular code design to facilitate easy updates and bug fixes.
	Logging and Monitoring	The model should maintain logs of user activities, device statuses, and server performance, allowing administrators to monitor and troubleshoot issues efficiently.
Interoperability	Standard Protocol Support	The API server should support RESTful APIs and be compatible with industry-standard protocols (e.g., HTTP, MQTT) to communicate with other IoT platforms.
	Hardware Integration	The microcontroller should be able to integrate with various GPS modules, and GSM modules without requiring significant hardware modifications.
	Data Export	The model should support exporting data in standard formats (e.g., CSV, JSON) for integration with other school management models or data analysis tools.
Data Integrity	Validation Checks	The API server should perform validation checks on all incoming data from IoT devices to ensure that it meets expected formats (e.g., valid GPS coordinates).
	Data Consistency	Updates to the MongoDB database (e.g., location data, user information) should follow ACID (Atomicity, Consistency, Isolation, Durability) principles to prevent data inconsistencies.
	Audit Logs	The model should maintain audit logs for all data changes, providing a history of who made changes, when they were made, and what changes were applied.
	Redundancy	Use data redundancy mechanisms to replicate critical data across multiple servers, reducing the risk of data loss in case of hardware failure.
Responsiveness	Real-Time Updates	The web app should reflect real-time updates of student locations with a refresh rate of no more than 5 seconds.
	Dynamic Alerts	The alert system should adapt to new boundary settings and notify parents or admins immediately when conditions are met.
	Adaptive UI	The web app interface should adjust dynamically to screen size and resolution changes, ensuring a smooth user experience across different devices.

Non-functional requirements (NFRs) are the qualitative aspects of a model that define its characteristics beyond its functional capabilities. They specify how the model should perform, rather than what it should do. In a model, NFRs are essential for ensuring that the model meets user expectations and operates effectively. The table following describes the non- functional requirements of the model;

4.3.3 Defining Model Technical Requirements

Technical requirements are the specific constraints, specifications, and criteria that a software model must meet to function correctly and effectively. They provide a detailed blueprint for the development team, outlining the technical aspects that need to be considered when designing and building the software. The *table 11* specifies the technical requirements designed for the 'IoT based surveillance model for monitoring school children'.

Table 11

Technical Requirements

Requirement Category	Specific Requirement	Description
Hardware Requirements	Microcontroller	Model: NodeMCU Lua Wi-Fi ESP8266
		Purpose: Acts as the main control unit for data communication between the GPS module and the GSM module, and connects to the API server.
		Specifications: Must support 2.4 GHz Wi-Fi, with at least 512 KB of RAM and 4 MB of flash memory.
	GPS Module	Model: NEO-M8N GY-GPSV3
		Purpose: Provides real-time location data of students.
		Specifications: Minimum location accuracy of 2.5 meters, update rate of up to 10 Hz, and support for multiple GNSS systems (e.g., GPS, GLONASS).
	GSM Module	Model: SIM800L or compatible
		Purpose: Sends SMS notifications to parents and admins when predefined conditions are met.
		Specifications: Must support 2G networks and standard AT commands for SMS functionality.
	Power Supply	Specification: 5V/2A power adapter to ensure stable

		power for the NodeMCU and connected modules. Backup Power: A rechargeable battery pack (e.g., 3.7V LiPo battery) to ensure functionality during power outages.
	Server Hardware	Specification: Cloud-based server with at least 2 CPU cores, 4 GB RAM, and 50 GB SSD storage to host the API server and the database. Purpose: Manages API requests, data storage, and user interactions.
Software Requirements	Web Application	Framework: Flask framework for frontend development. Purpose: Provides the interface for users (admins) to view student location data, receive alerts, and manage devices. Server-Side: Python framework for handling backend logic and API endpoints.
	Database	Type: MongoDB Purpose: Stores user data, GPS logs, device information, and alert history. Specification: NoSQL database optimized for read and write operations, capable of handling real-time data from multiple devices.
	API Server	Specification: RESTful API developed using Python. Purpose: Manages data communication between the web app, IoT devices, and the database. Features: Should support JSON data format, secure data exchange using HTTPS, and JWT (JSON Web Tokens) for user authentication.
	Device Firmware	Programming Language: Arduino (C) Purpose: Firmware for the NodeMCU to handle GPS data retrieval, communication with the API server, and triggering SMS alerts through the GSM module.
Network Requirements	Internet Connection	Specification: Minimum upload speed of 1 Mbps and download speed of 5 Mbps to ensure seamless data communication between the NodeMCU and the API server. Purpose: Enables real-time data updates and user interactions via the web app.
	Wi-Fi Coverage	Specification: The NodeMCU should connect to a Wi-Fi network within a radius of 50 meters. Purpose: Supports stable data transmission from the IoT device to the API server.
	SIM Card for GSM Module	Specification: Supports 2G network (GSM/GPRS) for regions with compatible network coverage. Purpose: Facilitates sending SMS alerts when students cross geofenced boundaries or during emergencies.
Data Management	Database Configuration	Type: MongoDB configured for high availability.

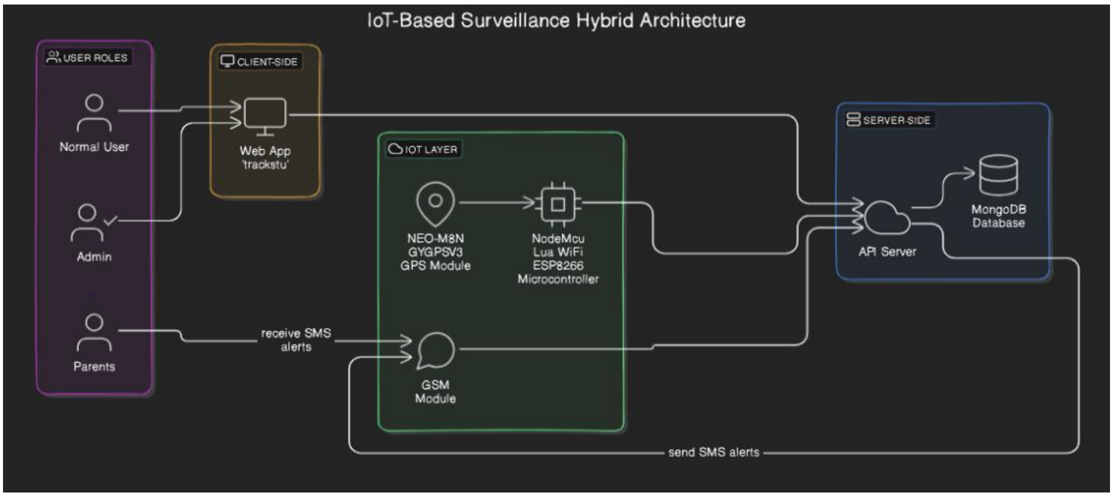
Requirements		
		Purpose: Manages and stores real-time GPS data, user information, and alert logs. Indexing: Implement indexing on GPS data to enhance retrieval times for location history.
	Data Backup Mechanism	Specification: Automatic backups every 12 hours to a cloud storage solution (e.g., AWS S3). Purpose: Prevents data loss in case of server failure.
	Data Retention Policy	Specification: Retain location data for a maximum of 30 days. Purpose: Reduces storage requirements and ensures compliance with data protection regulations.
Interface Requirements	User Interface (UI)	Specification: Responsive design to support desktop and mobile devices with varying screen sizes. Purpose: Ensures usability for admins and parents accessing the model through different devices.
	API Interface	Specification: RESTful endpoints for CRUD (Create, Read, Update, Delete) operations on user data, GPS logs, and device configurations. Purpose: Facilitates seamless communication between the web app and the API server.
	SMS Gateway Interface	Specification: Support for AT commands in the GSM module for sending SMS alerts. Purpose: Ensures compatibility between the GSM module and mobile networks for delivering alerts.
Deployment Requirements	Cloud Server Platform	Specification: AWS EC2 or a similar cloud provider for hosting the API server and MongoDB database. Purpose: Ensures high availability, scalability, and secure access to the server.
	Continuous Integration/Continuous Deployment (CI/CD)	Specification: Use GitHub Actions or Jenkins for automatic deployment of code updates to the server. Purpose: Enables frequent updates and integration of new features without manual intervention.
	IoT Device Setup	Specification: Proper configuration of NodeMCU with firmware and network credentials. Purpose: Ensures that the IoT devices can seamlessly connect to the server and perform their designated functions.

4.3.4 Model Design of the IoT Based Surveillance Model for Monitoring Children

This section presents the model design based on the group discussions and joint development activities in Objective II. From these discussions, the model was designed to include all the factors identified. The *figure 14* shows how the architecture of the model was setup.

Figure 14

Model Architecture for 'IoT based Surveillance Model for Monitoring School Children



The IoT-Based Surveillance Hybrid Architecture model is designed to ensure real-time monitoring and alerting for learner safety. It integrates multiple components, including user roles, client-side interactions, an IoT layer, and server-side elements, to create a seamless tracking model. The user roles include Normal Users (such as teachers), Admins who manage data and model settings, and Parents who receive updates about their child's status. The client-side comprises a web application called ‘trackstu,’ which serves as an interface for users to access real-time tracking data and updates.

At the core of the model is the IoT layer, which includes the NEO-M8N GYGPSV3 GPS module for real-time location tracking and the NodeMCU Lua WiFi ESP8266 microcontroller. The microcontroller processes data from the GPS module and enables communication with the server via Wi-Fi. Additionally, the GSM module is used to handle SMS alerts, ensuring parents receive notifications in critical situations, such as a child moving beyond a designated area.

The server-side architecture features an API server that receives, processes, and manages the data from the IoT layer, storing it in a MongoDB database for easy retrieval and analysis. This database stores user data, GPS coordinates, and records of SMS alerts. Administrators, teachers and other users can access this data through the ‘trackstu’ web app, which communicates with the API server to provide real-time updates. The integration of these components ensures a comprehensive and efficient model for monitoring learner safety, enhancing communication between stakeholders, and offering a reliable solution for educational institutions.

4.4 Development of the IoT Surveillance Model for Monitoring Children

This section showcases the achievement of the study's third research objective, which involved creating an IoT surveillance model for monitoring children. Proof of Concept methodology was applied for this objective and it was used to demonstrate the feasibility and practical functionality of the model. The primary focus of this section is to detail the implementation process of the developed IoT-based surveillance model for monitoring school children. The goal is to address objective III of this research, which aims to deploy a functional model that ensures real-time tracking, automated attendance, and notification of boundary breaches or emergencies.

The section outlines the step-by-step implementation of both hardware and software components that constitute the model. The implemented solution integrates GPS and GSM technologies to provide accurate tracking and data management. Additionally, it involves the development of a web application that serves as the user interface for school administrators, parents, and teachers, allowing them to interact with the model efficiently. The model is designed to provide timely alerts, real-time monitoring, and a user-friendly interface for managing student safety in school environments. This section covers the architectural design of the IoT-based model, the hardware and software integration process, and the deployment strategies used to achieve a working model. Additionally, the implementation challenges encountered during this process are discussed, along with the approaches used to resolve them.

The complete code for the developed model can be found in Appendix I.

4.4.1 Hardware Setup and Configuration

The hardware setup for the IoT-based surveillance model involves the integration of key IoT devices, including the microcontroller, GPS module, GSM module, and power supply, to ensure real-time tracking and communication. The configuration process involves establishing connections between these devices and setting up their functionalities to operate in the monitoring model effectively. Below is a detailed breakdown of the setup and configuration for each component:

Table 12

Hardware Setup and Configuration

Component	Model	Purpose	Setup
Microcontroller	NodeMCU Lua Wi-Fi ESP8266	Acts as the main control unit for data communication between the GPS module and the GSM module, and connects to the API server.	Flash the microcontroller with firmware using the Arduino IDE, enabling it to communicate with the GPS and GSM modules. Configure the Wi-Fi settings to ensure the microcontroller connects to the local Wi-Fi network with a minimum speed of 1 Mbps and a coverage radius of 50 meters. Use the appropriate libraries (e.g., ESP8266WiFi.h for Wi-Fi and Software Serial.h for serial communication) to establish connections with the GPS and GSM modules. Program the microcontroller to parse GPS data, send location updates to the server, and trigger SMS alerts through the GSM module.
GPS Module	NEO-M8N GY-GPSV3	Provides real-time location data of students.	Connect the GPS module to the NodeMCU using the serial communication pins (TX, RX). Power the GPS module using a 3.3V or 5V power source from the NodeMCU. Configure the module to support multiple GNSS systems (e.g., GPS, GLONASS) to ensure accurate location tracking. Calibrate the GPS module to provide a location accuracy of 2.5 meters and configure the update rate to 10 Hz for real-time tracking. Integrate the GPS library (e.g., TinyGPS++ for Arduino) into the NodeMCU firmware to parse and interpret GPS data.
GSM Module	SIM800L or Compatible	Sends SMS notifications to parents and admins when predefined conditions are met, such as boundary breaches.	Connect the GSM module to the NodeMCU using the TX/RX pins and power it using a 5V/2A power adapter for stability. Insert a SIM card that supports 2G networks (GSM/GPRS) with adequate mobile coverage for the school area. Program the NodeMCU to establish communication with the GSM module using the AT command set, which enables sending SMS alerts. Test the GSM module for network connectivity and ensure the module can send and receive SMS messages.
Power Supply	5V/2A power adapter with backup battery	Ensures continuous power supply to the NodeMCU and connected modules.	Use a regulated 5V adapter to power the NodeMCU and connected modules to prevent voltage fluctuations. Connect a 3.7V/1000mAh LiPo battery to the NodeMCU for backup power during power outages, ensuring that real-time tracking and SMS alerts remain functional. Install a battery management system (BMS) to monitor and manage battery charge levels.

Figure 15

Setup of the GSM Module

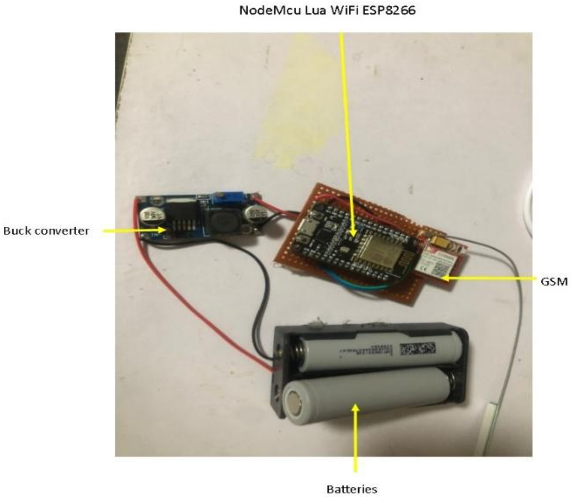


Figure 16

Setup of the GPS Module

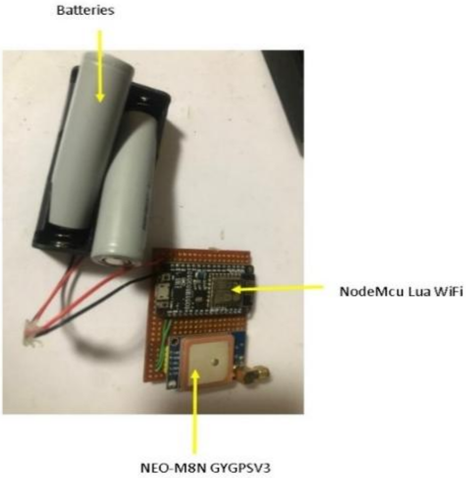


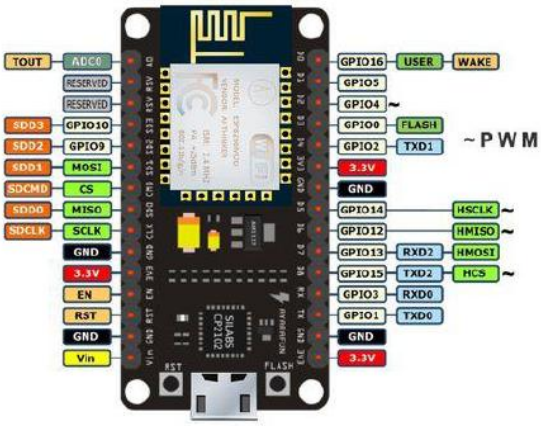
Figure 15 illustrates the complete setup of the GSM module, while *Figure 16* presents the GPS module and its components. A detailed discussion of each component and its role within the microcontroller will follow under this section.

i. **NodeMcu Lua WiFi ESP8266**

This microcontroller will act as the primary processing unit. It will handle communication with other sensors and modules, collect data from them, and transmit that data over WiFi to a designated server. The ESP8266 will also allow for monitoring of the project via a web-based interface.

Figure 17

NodeMcu Lua WiFi ESP8266

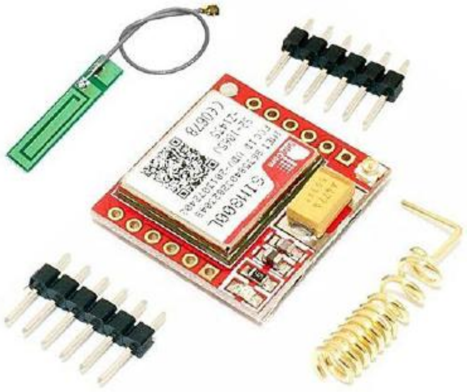


i. GSM Module

The GSM module will enable cellular communication, providing the project with the ability to send message to the parents or relevant individuals of the location of their learner and their status.

Figure 18

GSM Module



ii. NEO-M8N GYGPSV3 Module

This GPS module will provide real-time location data of the student by communicating with satellites. It will be used to track the exact position of the students, offering geographical information such as latitude and longitude.

Figure 19

NEO-M8N GYGPSV3 Module

iii. Buck Converter

The buck converter will regulate the input voltage from a higher voltage power source, such as a battery, and step it down to a lower, stable voltage that is suitable for powering components like the ESP8266, GSM module, or sensors. This will ensure that the components receive the correct voltage without getting damaged.

Figure 20

Buck Converter

iv. Battery and Battery Casing:

The battery will act as the main power source for the entire model, providing the necessary energy for the components to operate. The battery casing will house the battery securely, protecting it from physical damage and ensuring proper electrical connections. Depending on the project, rechargeable batteries may be used for extended operation.

v. Wires and Cabling:

Wires and cables will be used to make the necessary electrical connections between the different components, such as the ESP8266, GSM module, sensors, battery, and power regulation modules. These connections will enable the flow of power and data, ensuring that the model functions as an integrated unit.

4.4.2 Software Implementation

The software implementation of the IoT-based surveillance model was successfully completed, focusing on the development of three key components: the web application, API server, and database. These components were integrated to enable seamless communication between IoT hardware devices, the backend server, and the user-facing interface. Below is a detailed account of each aspect of the software implementation:

Table 13

Software Implementation

Component	Framework/ Language	Purpose	Implementation
Web Application Development	Flask Framework (Python)	To provide a user-friendly interface for administrators and parents, enabling real-time tracking of students, alert management, and report generation.	A Flask-based web application was developed using Python with Flask-RESTful for API integration, deployed on a cloud-based server for high availability. JWT authentication ensured secure access for different user roles. The interface was built using HTML, CSS, JavaScript, incorporating responsive design. Real-time data visualization was enabled using Chart.js and Leaflet.js. Extensive testing ensured usability, responsiveness, and functionality.
API Server Development	Python-based RESTful API using Flask-RESTful	To manage data communication between the IoT devices, the web application, and the database.	The API server was set up with Flask-RESTful, handling CRUD operations on user data, GPS logs, and alerts. Endpoints like /api/location, /api/alerts, and /api/users were implemented. JWT authentication secured API endpoints..
Database Configuration (MongoDB)	MongoDB (NoSQL Database)	To store user data, GPS logs, device information, and alert history efficiently.	A MongoDB instance was deployed on a cloud-based server with collections such as users, gps_logs, alerts, and devices. Indexing enhanced performance of location history queries, and automatic backups to AWS S3 ensured data safety
Integration of IoT Devices with the API Server	Arduino C for Device Firmware	To enable the transmission of GPS data from the IoT devices to the API server.	Custom firmware for the NodeMCU was developed using Arduino IDE, allowing communication between GPS modules and the API server over Wi-Fi. HTTP POST requests transmitted GPS data to the /api/location endpoint. Error handling ensured robust communication and retries during network issues.

4.4.3 Software Modules

The following section provides a detailed overview of the key modules developed within the IoT surveillance model. Each module plays a critical role in ensuring the functionality and effectiveness of the model, contributing to a comprehensive solution for monitoring school children.

4.4.3.1 User Registration and Authentication Module

The registration process is the foundational entry point into the IoT-based surveillance model for monitoring children, supporting both administrative and user roles. Registered users must provide credentials, such as their address and password, to verify their identity during the login process. Based on their role, users are granted specific privileges. While general users can access features according to their permissions, administrators hold enhanced privileges, including the authority to manage the model and perform high-level tasks.

In this model, only administrators have the ability to add new users and devices to the model. During the registration process, administrators are required to submit details of new users and create a password that is at least eight characters long for security purposes. The flowchart below outlines the registration workflow, which includes the different role-based activities. The registration logic is performed by the function *add user* as shown in Listing 1 below, which performs the necessary validation and assigns appropriate roles to each user.

Figure 21

User Registration Flowchart

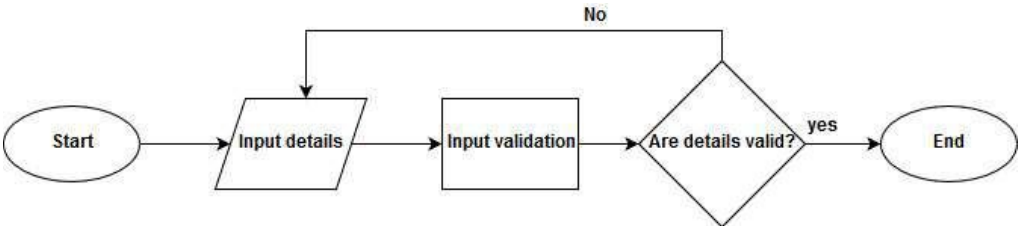
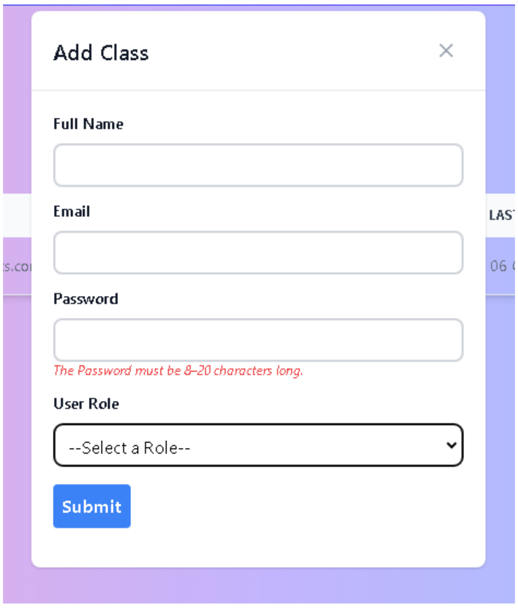


Figure 22

User Registration Form



Add Class ×

Full Name

Email

Password

The Password must be 8–20 characters long.

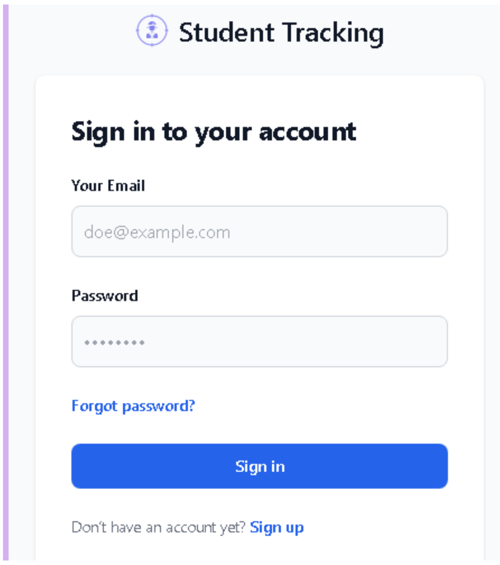
User Role

--Select a Role--

The user authentication process in this IoT-based surveillance model ensures that each user's identity matches the credentials stored in their account, providing a secure mechanism for validating access. This process is crucial for maintaining the integrity of the model by ensuring that only authorized individuals can access the model, preventing any unauthorized entities from infiltrating the model. The authentication process employs cryptographic security measures to safeguard user data and model access. The Python code implemented for user authentication is illustrated in Listing 1, while the login form used to initiate the process is displayed in the figure below. The overall authentication workflow is represented in the flowchart below, detailing the sequence of operations for verifying user identity.

Figure 23

User Login



Student Tracking

Sign in to your account

Your Email

doe@example.com

Password

••••••••

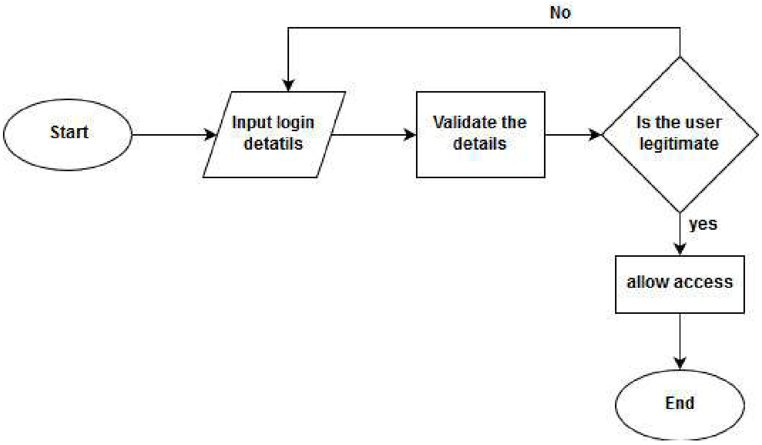
Forgot password?

Sign in

Don't have an account yet? Sign up

Figure 24

User Authentication Flowchart



List 1 User Registration and User Authentication Function:

```
def add_user():
    if request.method == "POST":
        email = request.form.get("email")
        password = request.form.get("password")
        name = request.form.get("name")
        role = request.form.get("role")
        if not email or not password or not name or not role:
            flash("Please fill all required fields", "error")
            return redirect(request.referrer)
        if len(password) < 8:
            flash("Password must be at least 8 characters", "error")
            return redirect(request.referrer)
        hashed_password = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt())
        db.insert(USER_TABLE, {"email": email, "password": hashed_password, "name":
name, "role": role,
                                   "addedOn": datetime.utcnow()})
        flash("User added successfully", "success")
        return redirect(request.referrer)
    return render_template("signup.html")
def login():
    if request.method == "POST":
        username = request.form.get("email")
        password = request.form.get("password")
        next_url = request.args.get("next")
        # print(next_url)
        if not username or not password:
            flash("Please fill all the fields", "error")
            return redirect(request.referrer)
        user = db.find_one(USER_TABLE, {"email": username})
        if not user:
            flash("Invalid credentials", "error")
            return redirect(request.referrer)
        user_class = User()
        user_class.id = str(user["_id"])
        user_class.email = user["email"]
        if user.get("lockedUntil") and user.get("lockedUntil") > datetime.utcnow() and
user.get("failedLoginAttempts", 0) >= 3:
            flash(f"Your account has been locked. Please try again after
{user.get('lockedUntil').strftime('%Y-%m-%d %H:%M:%S')} UTC", "error")
            return redirect(request.referrer)
        if not check_password(user["password"], password):
            # Increment failed login attempts
            last_attempt=user.get("failedLoginAttempts", 0)
            db.update(USER_TABLE, {"_id": ObjectId(user["_id"])},
{"failedLoginAttempts": last_attempt+1})
            if user.get("failedLoginAttempts", 0) >= 3:
                db.update(USER_TABLE,{"_id": ObjectId(user["_id"])},
                                              {"lockedUntil": datetime.utcnow() + timedelta(hours=5)})
                flash("Your account has been locked. Please try again after 5 hours", "error")
            else:
                attempts = 3 - user.get("failedLoginAttempts", 0)
                flash(f"Invalid credentials, you have {attempts} attempts left", "error")
            return redirect(request.referrer)
        if user and check_password(user["password"], password):
            login_user(user_class)
            db.update(USER_TABLE, {"_id": ObjectId(user["_id"]) },
{"failedLoginAttempts": 0})
            flash("Login successful", "success")
            return redirect(next_url or url_for("index"))
    return render_template("login.html")
```

4.4.3.2 Student Registration Module

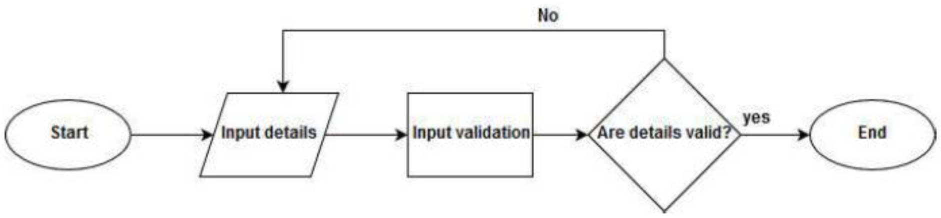
The Student Registration Module is a core component of the IoT-based surveillance model designed to manage learner data for tracking and monitoring purposes. The primary purpose of this module is to capture relevant personal information about the learners and link them with their unique GSM-enabled tracking devices. This ensures that each learner is identifiable in the model and can be monitored in real-time.

The registration process starts when a learner’s personal data, including their name, registration number grade, and parents name and phone number, is entered into the model by an administrator. Once this basic information is captured, the model assigns or GSM device to the learner. This device is crucial for the tracking functionality, allowing the model to monitor the learner's location as they move within and outside the school premises.

After assigning a device, the admin must verify the device’s operational status to ensure it is functioning correctly. If the device is active, it is successfully registered in the model and linked to the learner’s profile. In cases where the device is inactive or faulty, an error message is generated, prompting the admin to replace or troubleshoot the device. Once the learner is successfully registered and their tracking device is linked, the model begins real-time monitoring, and parents can also receive updates about their child’s movements through the model. The learner registration process is outlined in flowchart as shown below. The child registration logic is performed by the function register as shown in listing 2 below.

Figure 25

Registration Flowchart



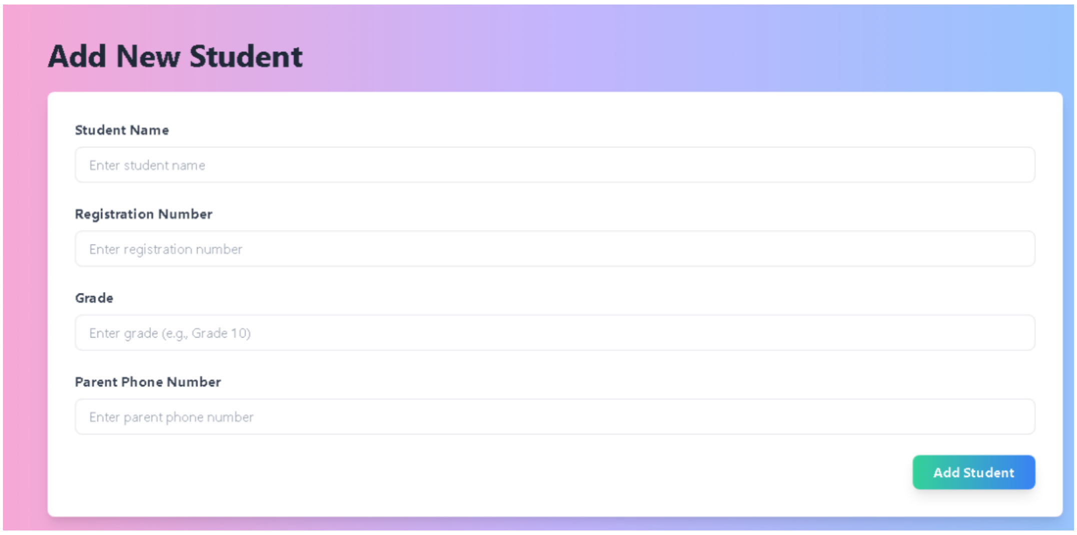
List 2 Registration Function

```
@app.route('/student/add', methods=['POST', 'GET'])
@login_required
def add_student():
    if request.method == 'POST':
        name = request.form.get('student-name')
        registration_number = request.form.get('registration')
        class_name = request.form.get('grade')
        parent_phone = request.form.get('parent-phone')
        parent_name = request.form.get('parent-name')
        student_location = request.form.get('student-location')
        if not name or not registration_number or not class_name or not parent_phone or not parent_name or not student_location:
            flash('Please fill in all the fields', 'error')
            return redirect(url_for('add_student'))
        db.insert(STUDENTS_COLLECTION, {
            'name': name,
            'registration_number': registration_number,
            'class_name': class_name,
            'parent_phone': parent_phone,
            "addedAt": datetime.utcnow(),
            "parent_name": parent_name,
            "home":student_location
        })
        flash('Student added successfully', 'success')
        return redirect(url_for('get_students'))
    return render_template('add_student.html')
```

This Python code outlines the basic process for registering a learner, validating their information.

Figure 26

Student Registration Form



Add New Student

Student Name

Enter student name

Registration Number

Enter registration number

Grade

Enter grade (e.g., Grade 10)

Parent Phone Number

Enter parent phone number

4.4.3.3 Device Registration

The Device Registration Module is a key component of the IoT-based surveillance model designed for tracking and monitoring children. Its purpose is to register and authenticate the devices (GSM-based tracking devices) assigned to child. This module ensures that only active, functional, and authorized devices are linked to the model, facilitating real-time monitoring of learners' locations. The device registration process begins with the model administrator inputting a new device's details, such as the device name, ID and assigned learner. The model then checks whether the device is already registered to avoid duplications.

The model logs every registration, keeping records of active devices for monitoring and reporting purposes. This module is crucial to ensure that every child has a functional device for accurate tracking. The device registration process is outlined in flowchart diagram as shown below. The device registration logic is performed by the function register as shown in listing 3 below.

Figure 27

Device Registration Flowchart

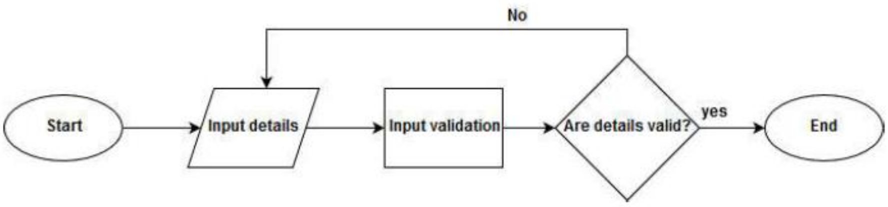
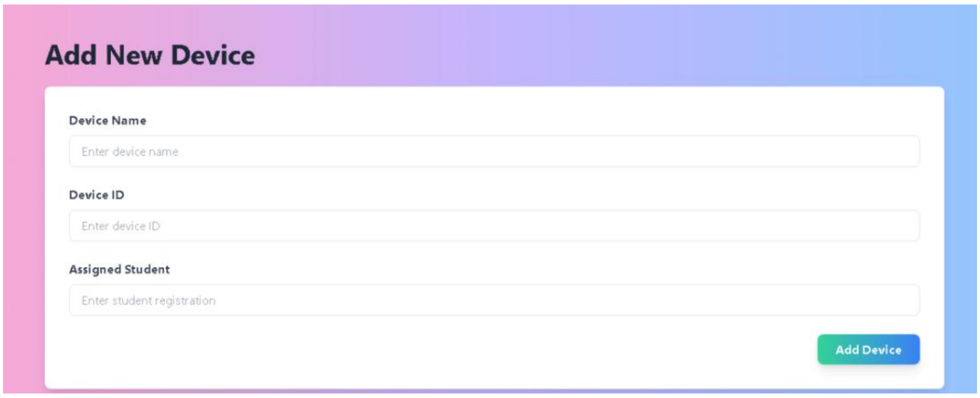


Figure 28

Device Registration Form



Add New Device

Device Name

Enter device name

Device ID

Enter device ID

Assigned Student

Enter student registration

List 3 Device Registration

```
def add_device():
    if request.method == 'POST':
        name = request.form.get('device-name')
        device_id = request.form.get('device-id')
        assigned_student_reg = request.form.get('assigned-student')
        if not name or not device_id or not assigned_student_reg:
            flash('Please fill in all the fields', 'error')
            return redirect(url_for('add_device'))
        student_details = db.find_one(STUDENTS_COLLECTION, {'registration_number':
```

```
assigned_student_reg})
        if not student_details:
            flash('Invalid student registration number', 'error')
            return redirect(url_for('add_device'))
        db.insert(DEVICES_COLLECTION, {
            'name': name,
            'deviceId': device_id,
            'studentId': student_details['_id'],
            "studentName": student_details['name'],
            "addedAt": datetime.utcnow()
        })
        flash('Device added successfully', 'success')
        return redirect(url_for('get_devices'))
    students=db.find(STUDENTS_COLLECTION, {})
    return render_template('add_device.html',
                           students=students)
```

This code outlines the basic steps involved in registering a new device, including checking for duplicates, and successfully registering the device if it passes all checks.

4.4.3.4 Real-Time Tracking

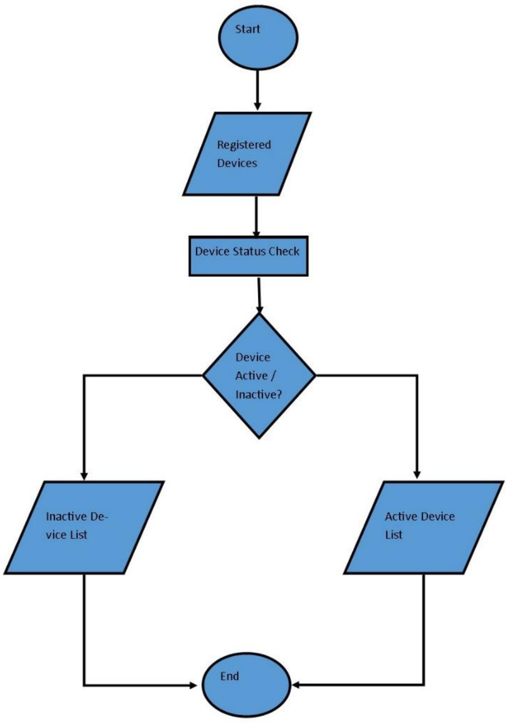
The Device List Module is an integral part of the IoT-based surveillance model for monitoring children. It is designed to display and manage a real-time list of active and inactive devices within the model. This module provides the admin with a centralized dashboard that shows the current status of all registered devices, allowing for quick identification of operational devices that are actively tracking learners, as well as those that are inactive or need troubleshooting. The purpose of this module is to ensure that only functional devices are used for real-time child tracking and that any inactive devices are flagged for immediate action.

The module continuously monitors each registered device, checking their operational status, and connectivity in real-time. It categorizes the devices into two lists: "Active Devices," which are currently functioning and providing real-time tracking data, and "Inactive Devices," which may have lost connectivity, run out of battery, or encountered a technical issue.

This classification allows the admin to quickly address any technical issues with the inactive devices, ensuring that the model maintains optimal functionality. The real-time device status data is refreshed periodically and can be accessed through the dashboard by the model administrators. The real-time tracking process is outlined in flowchart as shown below. The monitoring logic is performed by the function as shown in listing 4 below.

Figure 29

Device List Flowchart



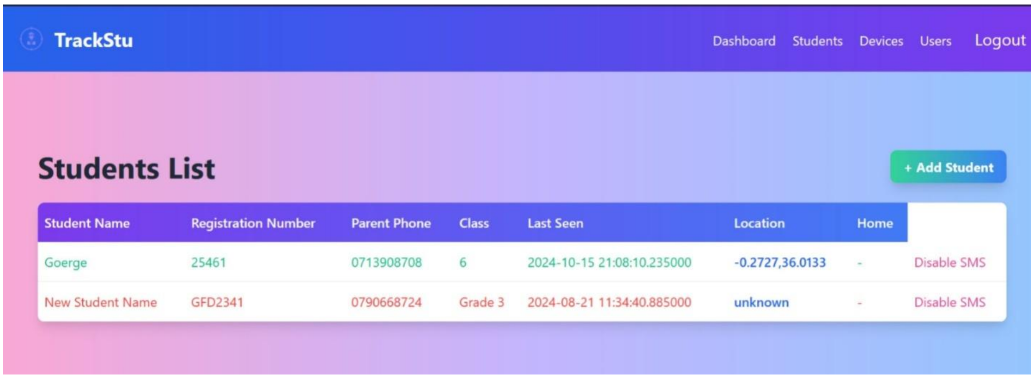
List 4 Device List

```
def get_device(device_id):
    device = db.find_by_id(DEVICES_COLLECTION, device_id)
    device_logs = db.find(LOCATION_LOGS_COLLECTION, {"student_id":
device['deviceId']})
    return render_template('device.html', device=device, device_logs=device_logs)
@app.route('/device/<device_id>/update', methods=['POST', 'GET'])
@login_required
def update_device(device_id):
    if request.method == 'POST':
        device = db.find_by_id(DEVICES_COLLECTION, device_id)
        device['name'] = request.form.get('device-name')
        device['deviceId'] = request.form.get('device-id')
        device['studentId'] = request.form.get('assigned-student')
        db.update(DEVICES_COLLECTION, device_id, device)
        flash('Device updated successfully', 'success')
        return redirect(url_for('get_devices'))
    device = db.find_by_id(DEVICES_COLLECTION, device_id)
    return render_template('update_device.html', device=device)
```

The code simulates the device list module by retrieving a set of registered devices and categorizing them based on their operational status. Devices with an "active" status are displayed in the list of active devices, while those with an "inactive" status are listed in the inactive section. The code then outputs these lists, allowing the admin to track which devices are operational and which need attention.

Figure 30

Device List Form



Students List

Student Name	Registration Number	Parent Phone	Class	Last Seen	Location	Home	
Goerge	25461	0713908708	6	2024-10-15 21:08:10.235000	-0.2727,36.0133	-	Disable SMS
New Student Name	GFD2341	0790668724	Grade 3	2024-08-21 11:34:40.885000	unknown	-	Disable SMS

Figure 30 displays the device list form within the model, where all devices are listed. Each device is associated with information about the student. Active devices are highlighted in green font, while inactive ones are marked in red. Additionally, the form provides the location coordinates and the last known position of each device.

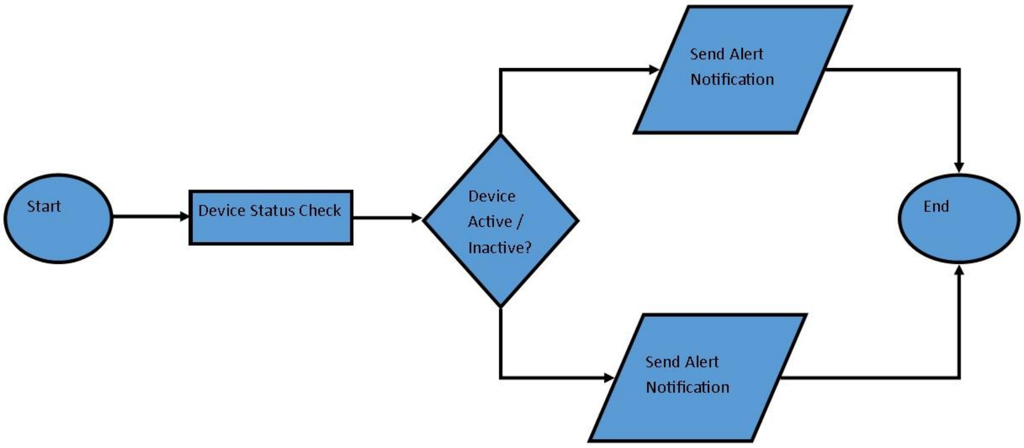
4.4.3.5 Alert Notification Module

The Alert Notification Module is a critical part of the IoT-based surveillance model designed to notify parents and the model administrators about the status of learners in real-time. The module's primary function is to send automatic alerts when learners' devices become active (indicating they have arrived at school) and when they become inactive (indicating they have exited the school). These notifications provide peace of mind to parents and ensure the model administrators are aware of the learners' movements.

The process begins when a learner's device becomes active in the model after entering the school. The module automatically generates an alert, which is sent to the parents' mobile devices via SMS, confirming the learner's arrival. Similarly, when the learner exits the school and the device goes inactive, the module triggers another alert, notifying the parent and updating the model. It ensures timely communication with both parents and the model, providing a safety net for monitoring children’s whereabouts in real-time.

Figure 31

Alert Notification Flowchart



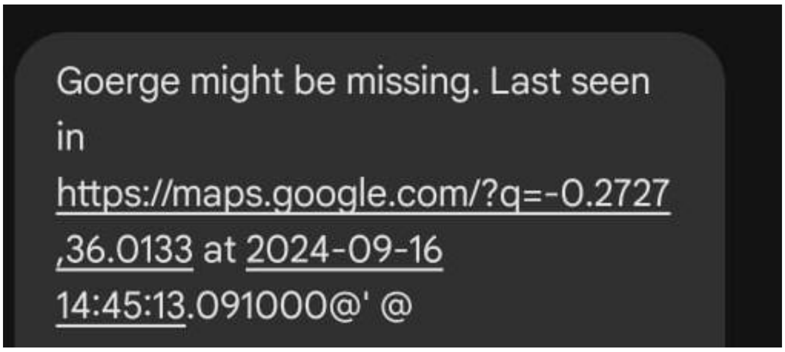
List 4 Alert Notification

```
def check_missing_student():
    students = db.find(STUDENTS_COLLECTION, {
        "lastSeen": {"$lt": datetime.utcnow() - timedelta(minutes=10)}
    })
    for student in students:
        if not student.get("missing_notify"):
            # print("Missing notification disabled")
            continue
        if student.get("last_missing_notification") and
student.get("last_missing_notification") > datetime.utcnow() - timedelta(hours=24):
            # print("Already sent notification")
            continue
        parent_phone = format_phone_number(student.get("parent_phone"))
        if not parent_phone:
            # print("""Parent phone number not found for student:
{}""".format(student.get("name")))
            continue
        last_seen = student.get("lastSeen")
        student_name = student.get("name")
        message=f"{parent_phone}|{student_name} might be missing. Last seen at
{last_seen}"
        mqtt_client.publish(topic=NOTIFY_TOPIC, payload=message )
        db.update_by_id(STUDENTS_COLLECTION, student.get("_id"),
{"last_missing_notification": datetime.utcnow()})
```

In the code, the model continuously checks the device status for each student and sends an alert message when the device becomes active or inactive. The GSM-based alert system provides an efficient way of notifying parents in real-time, ensuring learner safety is always monitored and communicated effectively.

Figure 32

Alert Message



Goerge might be missing. Last seen in https://maps.google.com/?q=-0.2727,36.0133 at 2024-09-16 14:45:13.091000@' @

Figure 32 shows a sample alert message generated by the gsm model when the child goes out of range.

4.4.5 Deployment of the Model

The deployment of the IoT-based surveillance model is critical in ensuring the model becomes operational, enabling stakeholders—such as parents, teachers, and administrators—to access its functionalities through a web interface. This phase involved hosting the web application and database on a secure server environment to provide real-time access to information, accessible at the URL https://student.sh.co.ke. The deployment process included setting up the server, deploying the backend API and frontend web application, and ensuring proper integration with the MongoDB database.

The web application was hosted on a virtual private server (VPS) configured with sufficient computing resources to handle data processing and user requests. The server was set up using a Linux-based operating model to ensure compatibility with the backend and database services. Additionally, the deployment involved registering the domain https://student.sh.co.ke and pointing it to the server's IP address for easy access. The frontend application, developed with the Flask framework, offers an intuitive user interface, allowing parents, teachers, and administrators to interact seamlessly with the model. To manage incoming HTTP requests, a reverse proxy server—either Nginx or Apache—was used to balance the load and ensure scalability. Security was also prioritized through the installation of SSL/TLS certificates, enabling HTTPS to encrypt data transmitted between users and the server.

The MongoDB database was deployed on the same server to store essential information such as user profiles, GPS logs, and alert notifications. To ensure quick data retrieval, high availability and indexing were configured, optimizing database queries. Regular automated backups were scheduled to create snapshots, ensuring that data could be recovered if the model encountered issues. For security and access control, the deployment utilized JSON Web Tokens (JWT) to secure API endpoints and authenticate users, while role-based access control (RBAC) limited data modification and access based on user roles, such as normal users or administrators.

The backend API, written in Python, was deployed alongside the web application to facilitate communication between IoT devices and the server. The API supported Create, Read, Update, and Delete (CRUD) operations, which ensured smooth data interactions between the web application and the MongoDB database. The API endpoints were also configured to process real-time inputs from the IoT devices, including GPS location updates and alerts sent through the GSM module. This deployment strategy ensures that the model remains efficient, secure, and accessible to all intended users.

4.4.6 Challenges Faced in Model Implementation Phase

Table 14

Challenges Faced in Implementation

Challenge Category	Challenges	Solutions
Technical Challenges	Integration of IoT Hardware: Integrating GPS, GSM, and NodeMCU required precise configuration and compatibility checks. Real-Time Data Processing: Achieving low-latency communication for real-time tracking was challenging due to intermittent internet connectivity.	Extensive testing ensured seamless communication between modules. Data buffering and protocol optimization improved transmission rates.
Environmental and Operational Challenges	GPS Signal Accuracy and Interference: Signal accuracy was affected by physical barriers and weather conditions. Power Management for IoT Devices: Managing power consumption for devices, especially in areas with limited access, was a challenge.	High-precision GPS and filtering algorithms improved location accuracy. Low-power modes and rechargeable battery packs ensured continuous operation.
Data Security and Privacy Challenges	Ensuring Data Security: Protecting sensitive student data from unauthorized access was critical. Compliance with Data Privacy Regulations: The model needed to adhere to regulations like GDPR for data protection.	Implemented encryption, JWT for secure API access, and role-based access control. Developed data retention policies and obtained user consent during registration.
Network and Connectivity Challenges	Inconsistent Internet Connectivity: Real-time tracking depended on stable internet, which was not always available. Server Downtime and Maintenance: Downtime during updates impacted user access to the model.	Configured local data storage on GSM module and used 2G fallback. Implemented cloud-based redundancy and scheduled updates during off-peak hours.
Integration with Existing Systems	Compatibility with School Management Systems: Integrating with existing systems posed compatibility issues. Data Synchronization Between Systems: Aligning update cycles for accurate attendance data was a challenge.	To Develop custom APIs and middleware solutions for interoperability. Implement periodic data sync processes to maintain alignment between models.

4.5 Model Evaluation

4.5.1 Goal Based Evaluation

The Goal-based evaluation of IT models was also used to evaluate the IoT based surveillance model for monitoring children. The goal-based evaluation methodology focuses on assessing whether an initiative or model achieves its predefined objectives and goals.

Table 15

Model Evaluation

Objective	Evaluation
User Registration and Authentication: The objective of this module is to register learners in the model, capturing their personal information and the model to be able to authenticate registered users.	The model successfully captured and stored user data, including personal information and the association with a role. The model was able to authenticate existing users in the database who tried to login.
Student Registration: The objective of this module is to register learners in the model, capturing their personal information and linking them to specific GPS devices for tracking purposes.	The model successfully captured and stored learner data, including personal information and the association with a GPS device. The process was efficient and secure, ensuring accurate record-keeping. However, improvements can be made to streamline data input and enhance user experience for administrators.
Device Registration: This module aims to register GPS devices and associate them with individual learners for real-time tracking. It involves linking the GPS device to each learner.	The model accurately registered GPS devices and linked them to individual learners. Device registration was found to be reliable, but model scalability may be a concern when registering large numbers of devices simultaneously. The interface and speed of the device linking process could be further optimized.
Real-time tracking: The objective is to provide continuous monitoring of learners' locations through GPS tracking. It tracks active and inactive devices, showing learners' whereabouts in real time.	The model provided continuous real-time tracking of learners, offering a clear view of active and inactive devices. Tracking was generally accurate but could experience delays in areas with weak GPS signal coverage, impacting the model's reliability in certain locations.
Alert Notification: This module is designed to send alerts to parents and administrators via GSM when learners arrive at or leave the school premises.	The alert notification model using GSM effectively sent timely notifications to parents and administrators when learners arrived or exited the school area. However, occasional delays in sending alerts due to network issues were noted. Further improvements in handling inactive device alerts could improve overall reliability and timeliness of notifications

4.5.2 Expert Survey using Validation Metrics

An expert survey was conducted to establish the validity of the IoT-based surveillance model for monitoring children. The expert survey involved 15 participants, comprising IoT experts, school administrators. The model evaluated based on four software metrics: usability, reliability, efficiency, and functionality. To gather data for these metrics, six (6) expert survey questions were structured to obtain the necessary responses for validating the model. Here is the description of the metrics used;

i. **Usability**: This metric assesses how easy it is for different users (parents, teachers, administrators) to interact with the model. The usability metric would look at the intuitiveness of the user interface, ease of navigation, and the simplicity of key processes like registration, real-time monitoring, and alert notifications.

ii. **Reliability:** This evaluates how consistently the model performs under different conditions. Reliability would involve assessing the model's ability to track children in real-time without significant delays or data loss and ensure that alerts are sent on time.

iii. **Efficiency**: This measures the model's performance in terms of resource usage (e.g., processing power, network bandwidth, battery life for devices) and speed. Efficiency would include testing how quickly the model processes GPS data and sends alerts to parents.

iv. **Functionality:** This metric assesses the overall performance of the model and whether it meets its functional requirements, such as tracking children, sending alerts, and registering devices.

The data for validation of the four metrics were captured as summarized in the table below;

Table 16

Software Validation Metrics

Metrics		Question	Agree	Disagree	Total Responses
Usability	Q1	The interface of the IoT-based surveillance model is user-friendly and easy to navigate.	11	4	15
	Q2	Users (e.g., teachers, administrators) can easily understand and interact with the model's features without extensive training.	8	7	15
		Usability Means	10	6	15
Efficiency	Q3	The model consistently provides accurate and reliable real-time data about children's locations.	9	6	15
	Q4	The model operates without frequent errors, downtimes, or malfunctions during regular use.	14	1	15
		Efficiency Means	12	4	15
Reliability	Q5	The model efficiently processes data and delivers real-time alerts and notifications without significant delays.	9	6	15
Functionality	Q6	The model includes all the necessary features to effectively monitor children and meet the surveillance needs of the school.	10	5	15

The expert survey evaluated the IoT-based surveillance model using four key software metrics: Usability, Efficiency, Reliability, and Functionality. The findings revealed that the model was generally well-received, though a few areas require further refinement to enhance performance and user experience.

Usability was assessed through two questions, with 73.3% of respondents agreeing that the model interface was user-friendly and easy to navigate. However, 26.7% expressed dissatisfaction, indicating that some users found the interface challenging to operate. Additionally, 53.3% of participants agreed that the model's features were understandable without extensive training, but a significant 46.7% felt that onboarding new users required more training or guidance. These results suggest that while the model is accessible for many, additional user support materials such as tutorials or quick-start guides may be necessary.

Under Efficiency, 60% of the experts agreed that the model provided accurate and reliable real-time location data for children, while 40% noted occasional discrepancies. This indicates that the model generally performs well but could benefit from enhancements in GPS calibration and connectivity to address any data inconsistencies. Furthermore, 93.3% of respondents reported that the model operated without frequent errors, downtime, or malfunctions, highlighting the stability of the platform, with only 6.7% expressing concerns about occasional technical issues.

The Reliability of the model was also evaluated, focusing on the timely processing of data and delivery of real-time alerts. The results showed that 60% of the participants agreed that notifications were delivered efficiently without significant delays. However, 40% reported occasional lags, indicating the need for gsm module optimization to ensure consistent and prompt alerts.

In terms of Functionality, 66.7% of experts agreed that the model contained all the necessary features to meet the school's surveillance needs, while 33.3% felt that additional customization options were required.

In summary, the expert survey demonstrated that the IoT-based surveillance model is stable, functional, and effective in most areas. However, the results also highlight a few areas for improvement, particularly in usability and notification reliability. Incorporating feedback from the experts—such as simplifying the user interface, improving GPS accuracy, and enhancing alert delivery—will help optimize the model and ensure it fully meets the needs of all stakeholders.

4.5.3 Expert Recommendations Based on Survey Results

Table 10

Expert Survey Recommendations

Metric	Expert Recommendations
Usability	Simplify the user interface and provide additional support materials such as tutorials or quick-start guides to assist new users in understanding and interacting with the model’s features.
Efficiency	Enhance GPS calibration and improve connectivity to address occasional data discrepancies, ensuring accurate and reliable real-time location data.
Reliability	Optimize the GSM module to reduce occasional notification delays, ensuring consistent and prompt alert delivery.
Functionality	Introduce additional customization options to better align with specific surveillance needs of schools, addressing feedback from users requiring more tailored features.

CHAPTER FIVE

SUMMARY, CONCLUSIONS AND RECOMMENDATIONS

5.1 Introduction

This chapter presents the conclusions and recommendations derived from the research on the design, development, evaluation, and results of the IoT surveillance model for monitoring children. Additionally, it suggests potential areas for further study.

5.2 Summary of the Findings

This study was driven by the limitations of the current child surveillance methods in Kenya. The findings suggest that IoT technology, which has not yet been fully implemented, offers a promising solution to enhance child surveillance by leveraging its numerous advantages.

5.3 Conclusions

This section presents the conclusions made from the research as per the specific objectives. The conclusions in this study with regard to each of the objectives are presented in this section and they are summarized as follows:

In addressing the weaknesses of IoT based surveillance models for monitoring children, the study examined the challenges impacting surveillance models for monitoring children. To achieve this, secondary data was analyzed through a systematic literature review. This method enabled the identification of several issues and weaknesses within the current models.

Existing IoT-based surveillance models for monitoring children present several weaknesses that limit their effectiveness and raise concerns. One of the primary issues is privacy, as these models often collect sensitive personal data, including location information, which can be vulnerable to unauthorized access or data breaches if not properly secured. Additionally, the cost of implementing and maintaining such models can be prohibitively high, particularly for schools with limited resources, as expenses include hardware, software, and regular maintenance.

Data security is another critical concern, as IoT devices are susceptible to hacking and cyberattacks, especially if encryption methods and security protocols are weak or outdated. Furthermore, these models heavily rely on stable network connectivity, and in areas with poor internet infrastructure, the model may experience disruptions, leading to delays or missed alerts. Inaccuracy and false alerts are also common issues, as some models may trigger unnecessary alarms when children are still within the designated safe zones, undermining trust in the model.

Finally, ethical concerns arise from the constant monitoring of children, with some perceiving the surveillance as overly intrusive. These weaknesses highlight the need for a more secure, scalable, and customizable IoT surveillance model that balances safety with privacy and ethical considerations.

In addressing an IoT Based surveillance model for monitoring children, several critical steps were identified based on the findings. The results showed that clearly defining the objectives and requirements of the model is essential. This includes specifying which aspects of children’s activities, such as location, entry and exit points, need to be monitored, and determining the scope of surveillance. Furthermore, the results demonstrated that for effective monitoring, the model must prioritize real-time tracking, accuracy, scalability, and robust data security measures.

The study also highlighted the importance of designing a comprehensive model architecture. This architecture, as indicated by the results, should include essential components like sensors, tracking devices, communication modules, cloud servers, and a user interface. The results further showed that selecting appropriate IoT devices, such as GPS modules for location tracking, is crucial for ensuring reliability and ease of integration into children's belongings.

In terms of communication, the findings suggested that using suitable protocols like Wi-Fi, Bluetooth, or low-power wide-area networks (e.g., LoRa or NB-IoT) enhances real-time data transmission and overall model performance.

Moreover, the results indicated that a user-friendly interface is necessary for enabling administrators, parents, and teachers to access real-time data and receive alerts. Data security was identified as another key element, with the results showing that encryption, authentication, and role-based access control are essential for protecting sensitive information.

In addressing the implementation of an IoT-based surveillance model for monitoring children, the study employed a proof of concept (PoC) approach. The results showed that the PoC was instrumental in developing and refining the surveillance model. Initially, a PoC was designed incorporating IoT devices such as GPS trackers. This approach allowed for the practical assessment of device integration and functionality in a real-world setting.

The results demonstrated that using a PoC enabled the effective evaluation of communication protocols for data transmission, including Wi-Fi and Bluetooth. These protocols facilitated seamless connectivity between the IoT devices and the central cloud-based server, where data was processed and stored.

Furthermore, the PoC phase revealed the importance of developing a robust software platform. The results indicated that the cloud-based model successfully managed data from various devices, performed real-time processing, and triggered alerts based on predefined conditions, such as boundary breaches. The user interface designed during this phase proved to be crucial for administrators, and teachers, allowing them to track children, receive alerts, and manage the model effectively.

Data security measures, including encryption and secure access controls, were integrated into the PoC, addressing privacy concerns. The results showed that these measures were effective in protecting sensitive information.

Testing the PoC in a controlled environment, such as a specific section of a school, provided valuable insights into model performance. Feedback collected during this phase was used to make necessary refinements, ensuring the model’s functionality, accuracy, and reliability before scaling up for broader implementation. The results highlighted that continuous monitoring and iterative improvements based on PoC testing are essential for the successful deployment of the IoT-based surveillance model.

In addressing the evaluation of the suitability of the IoT based surveillance model in monitoring children and determine its success. The model was evaluated to verify whether it met the set objectives, using the Goals-based evaluation methodology. This approach was employed to measure the project’s outcomes against the established goals and objectives. An evaluation test regime was implemented to assess the model following a demonstration to school administrators. This was done to determine if the model was fit-for-purpose and if it achieved its desired goals.

An expert survey was conducted to validate the model’s performance, involving IoT experts, school administrators. The survey gathered insights on various aspects of the model, including functionality, usability, and reliability, based on real-world applications. Following a demonstration to school administrators, an evaluation test regime was implemented to further assess the model's effectiveness and suitability.

In summary, the expert survey demonstrated that the IoT-based surveillance model is stable, functional, and effective in most areas. However, the results also highlighted a few areas for improvement, particularly in usability and notification reliability. Incorporating feedback from the experts—such as simplifying the user interface, improving GPS accuracy, and enhancing alert delivery—will help optimize the model and ensure it fully meets the needs of all stakeholders.

5.4 Recommendations

Schools should adopt IoT-based surveillance models to monitor learners' real-time locations and attendance. This will improve learner safety, especially in environments with high security concerns. Schools should start by implementing these models in high-risk areas such as school gates, classrooms, and playgrounds.

Schools must ensure that learner data collected through IoT surveillance models is securely stored and handled. Schools should implement data privacy protocols to protect against data breaches, in compliance with government regulations. Staff should be trained on handling sensitive data responsibly.

Schools should consider customizing IoT models to fit their specific requirements, such as varying class schedules, transportation models, and the particular security needs of their environment. Flexibility in integration will make the models more effective in a Kenyan context, especially considering regional diversity.

In many rural schools, the lack of stable power, reliable internet connectivity, and technical support presents significant challenges to implementing IoT-based surveillance systems. To address these issues, the proposed model should incorporate low-power, cost-effective solutions that can operate under constrained resources. For instance, using battery-powered GPS devices with energy-efficient tracking modes would allow the system to function consistently, even where power infrastructure is limited or unreliable. Additionally, designing the model to work effectively with intermittent connectivity—by enabling mobile data as a backup option—can enhance reliability in remote areas, ensuring that tracking and monitoring continue even in locations with variable network coverage.

The government should foster partnerships with private technology firms, NGOs, and other stakeholders to facilitate the integration of IoT technologies in schools. This collaboration could lead to improved model scalability, lower costs, and better support for schools that lack the capacity to implement these models independently.

Strategic partnerships with IoT technology providers, telecom companies, and local tech hubs could be particularly beneficial for schools in under-resourced areas. Through these collaborations, schools may access discounted devices or even receive pro bono technical support, significantly reducing the initial setup costs for implementing IoT-based surveillance models. Such partnerships could be structured as formal agreements with schools or supported through government initiatives, making IoT technology more accessible and affordable for institutions with limited budgets.

Furthermore, partnerships with technology providers could offer schools a sustainable and cost-effective way to maintain their IoT systems. Many providers periodically upgrade their devices and may offer schools newer equipment at lower prices as part of the partnership arrangement. Additionally, these providers might include technical support services, encompassing maintenance, troubleshooting, and regular software updates, which would help ensure the longevity and effectiveness of the surveillance model. This ongoing support would be crucial for schools, especially those lacking the in-house technical expertise needed to keep the system running smoothly.

Encouraging public-private collaboration, supported by government incentives for technology providers, could significantly boost IoT adoption in the education sector. By providing financial or tax incentives, the government could make it more appealing for technology companies to invest in educational safety initiatives. This approach would not only facilitate broader IoT implementation in diverse regions but also strengthen the overall impact of IoT technology on student safety across the country.

5.4.1 Policy Recommendations

Firstly, it is essential to establish robust data privacy and security regulations. This includes developing guidelines to ensure that all data collected through surveillance models is protected with strong encryption and secure storage, and limiting data access to authorized personnel. Compliance with national and international data protection laws, such as Kenya's Data Protection Act and the GDPR, is crucial to safeguard the privacy of children and their families.

Standardization and the adoption of best practices should be promoted to ensure consistency and quality across schools. This involves creating national standards for IoT surveillance models, including performance benchmarks and security measures, and encouraging schools to follow these best practices through regular model audits and staff training. Additionally, to address financial constraints, funding support should be provided through government or private sector initiatives, such as grants or subsidies, to help schools with the costs associated with the implementation and maintenance of surveillance models. Financial incentives could also be offered to encourage the adoption of innovative technologies.

Ethical and legal considerations must be addressed by implementing clear ethical guidelines to balance security needs with respect for privacy, ensuring transparency in surveillance practices. Policies should be reviewed and updated regularly to adapt to technological advancements and evolving legal and ethical standards.

Lastly, establishing metrics for evaluating the effectiveness of IoT surveillance models, including their impact on child safety and privacy, and conducting regular audits will help guide ongoing improvements and ensure compliance with privacy standards.

5.4.2 Recommendations for Further Research

First, future research should investigate the integration of emerging technologies such as artificial intelligence, machine learning, and advanced data analytics to improve the accuracy and efficiency of surveillance models. Longitudinal studies are needed to assess the long-term effectiveness of these models in various school environments, evaluating their sustained impact on child safety, privacy, and model reliability.

Additionally, examining stakeholder feedback from parents, teachers, learners, and administrators can provide valuable insights into their experiences and acceptance of IoT surveillance models. This research could reveal how these perspectives influence the technology's effectiveness and adoption. A detailed cost-benefit analysis should be performed to evaluate the economic viability of implementing and maintaining such models in schools, comparing financial impacts with safety and operational benefits.

Studying the impact of IoT surveillance models on educational outcomes and the school environment could provide insights into how monitoring affects learner behavior and learning experiences. Additionally, exploring the interoperability of IoT surveillance models with existing school management systems and other technologies is crucial for seamless integration. Comparative studies of surveillance models used in different countries could offer valuable perspectives on best practices and innovative approaches. Finally, research into the impact of various regulatory and policy frameworks on the deployment and operation of IoT surveillance models is needed to understand how regulations influence model design, implementation, and compliance. These recommendations aim to deepen the understanding of IoT surveillance models and enhance their effectiveness in safeguarding children in educational settings.

REFERENCES

Ahmed, A., Parvez, M., Hasan, H., Nur, F., Nessa, N., Karim, A., Azam, S., Shanmugam, B., & Jonkman, M. (2019). An intelligent and secured tracking model for monitoring school bus. *2019 International Conference on Communication and Electronics Systems (ICCES)*, 1-5. https://doi.org /10.1109/I CCCI.2019.8822187

AJayalakshmi, P. Srujana, P. Ramesh, G. Layasri. (2021). Design of an Intelligent School Bus Monitoring and Reporting System via IoT. Annals of the Romanian Society for Cell Biology, 3937–3944. Retrieved from https://www.a nnalsofrscb .ro/index.php/journal/article/view/5060

Aldoseri, A., Al-Khalifa, K. N., & Hamouda, A. M. (2024). Methodological approach to assessing the current state of organizations for AI-based digital transformation. Applied System Innovation, 7(1), 14. https://doi.org/10.3390/asi7010014The specific objectives evaluated in this

Al-Mazloum, A., Omer, E., & Abdullah, M. F. A. (2023). Child safety wearable device. *Journal of Emerging Technologies and Innovative Research*, 10(5).

Al-Turjman, F., Zahmatkesh, H., & Shahroze, R. (2022). An overview of security and privacy in smart cities' IoT communications. *Trans Emerging Tel Tech*, 33. https://doi.org/10.1002/ett.3677

Anusha, R., & Naidu, R. C. A. (2016). *GPS and RFID Based School Children Tracking System*. International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), 5(6).

Bentotahewa, V., Yousif, M., Hewage, C., Nawaf, L., & Williams, J. (2022). Privacy and security challenges and opportunities for IoT technologies during and beyond COVID-19. In Emerging technologies during and beyond COVID-19 pandemic. Springer. https://doi.org/10.1007/978-3-030-91218-5_3

Bhatti, M. K. L., Khan, I., & Kim, K.-I. (2022). A Comprehensive Review of Internet of Things: Technology Stack, Middlewares, and Fog/Edge Computing Interface. *Sensors, 22*(3), 995. https://doi.org/10.3390/s22030995

Cedarbaum, J. M. (2018). Elephants, Parkinson's Disease, and Proof- of- Concept Clinical Trials. *Movement Disorders*, *33*(5), 697-700.

Chen, M., Ma, Y., Li, Y., Wu, D., Zhang, Y., & Youn, C.-H. (2017). Wearable 2.0: Enabling human-cloud integration in next generation healthcare systems. IEEE Communications Magazine, 55(1), 54-61.

Chen, X., Wang, X., & Xu, Y. (2023). Performance enhancement for a GPS vector-tracking loop utilizing an adaptive iterated extended Kalman filter. *Sensors, 14*(12), 23630-23649. doi:10.3390/s141223630

Cheung, S. K. S., Kwok, L. F., Phusavat, K., & others. (2021). Shaping the future learning environments with smart elements: Challenges and opportunities. International Journal of Educational Technology in Higher Education, 18(16). https://doi.org/10.1186/s41239-021-00254-1

Choi, H., Chakraborty, S., Charbiwala, Z. M., & Srivastava, M. B. (2011). SensorSafe: A framework for privacy-preserving management of personal sensory information. *Secure Data Management*, 85-100. Springer.

Costello, A., & Naimy, Z. (2019). Maternal, newborn, child and adolescent health: Challenges for the next decade. International Health, 11(5), 349-352. https://doi.org/10.1093/inthealth/ihz051

Damaševičius, R., Bacanin, N., & Misra, S. (2023). From Sensors to Safety: Internet of Emergency Services (IoES) for Emergency Response and Disaster Management. *Journal of Sensor and Actuator Networks*, 12(3), 41. https://doi.org /10.3390/jsan12030041

Denning, T., Matuszek, C., Koscher, K., Smith, J. R., & Kohno, T. (2009). A spotlight on security and privacy risks with future household robots: Attacks and lessons. In *Proceedings of the 11th ACM International Conference on Ubiquitous Computing (UbiComp)*, Orlando, FL.

Dian, F. J., & Vahidnia, R. (2019). Radiation safety hazards of cellular IoT devices. IEEE Technology and Society Magazine.

Dian, F. J., Vahidnia, R., & Rahmati, A. (2020). Wearables and the Internet of Things (IoT), Applications, Opportunities, and Challenges: A Survey. IEEE Access.

El Mrabet, H., & Ait Moussa, A. (2020). IoT-school attendance system using RFID technology. *International Journal of Interactive Mobile Technologies (iJIM), 14*(14), 4-16. https://doi.org/10.3991/ijim.v14i14.14625

Elbasi, E. (2020). Reliable abnormal event detection from IoT surveillance systems. Proceedings of the 7th International Conference on Internet of Things: Systems, Management and Security (IOTSMS).

Gallardo-Echenique, E. E., Mar-Molinero, C., Bullock-Rest, N. E., & Villarreal-Rosas, J. (2018). School surveillance and student safety: Examining the relationship between camera presence and student disciplinary infractions. *Journal of School Violence, 17*(4), 461-477.

Ghasempour, A. (2019). Internet of Things in Smart Cities: Challenges and Opportunities. *IEEE Internet of Things Journal*. https://doi.or g/10.1109 /JIOT.2019.2945365

Gordon, N. (2014). Flexible pedagogies: Technology-enhanced learning. The Higher Education academy. http://www.heacad emy.ac.uk/resour ces/detail/fle xiblelearning/flexiblepedagogies/tech_enhanced_learning/main_reportGottfried, M., Page, L., & Edwards, D. (2023). Reducing student absenteeism. *EdResearch.*

Gowri, P. B., Abirami, K., & Monisha, T. (2023). Smart child safety monitoring system. *JETIR*, 10(5).

Gürses, S., Troncoso, C., & Diaz, C. (2011). Engineering privacy by design. *Comput Priv Data Prot*, 14(3), 25.

Hsu, C.-H., Wang, M. Y.-C., Shen, H. C. H., Chiang, R. H.-C., & Wen, C. H. P. (2017). FallCare+: An IoT Surveillance System for Fall Detection. Proceedings of the IEEE International Conference on Applied System Innovation.

Hu, L., & Ni, Q. (2018). IoT-driven automated object detection algorithm for urban surveillance systems in smart cities. IEEE Internet of Things Journal, 5(2), 747–758.

ISO/IEC. (2011). *Systems and software engineering—Systems and software quality requirements and evaluation (SQuaRE)—System and software quality models* (BS ISO/IEC 25010: 2011). BSI Group: Geneva, Switzerland.

Jaiswal, K., Sobhanayak, S., Mohanta, B. K., & Jena, D. (2017). IoT-cloud based framework for patient's data collection in smart healthcare system using Raspberry-Pi. In *Proc. IEEE Int. Conf. Elect. Comput. Technol. Appl. (ICECTA)* (pp. 1–4).

Jeyakkannan, N., Karthik, C., & Lukose, V. (2021). *IoT Based Smart Bus System Using Wireless Sensor Networks*. Journal of Physics: Conference Series, 1937, 012017.

Jha, S., Kruger, L., & Shmatikov, V. (2008). Towards practical privacy for genomic computation. In *Proceedings of the 2008 IEEE Symposium on Security and Privacy (SP),* Oakland, CA.

Jung, J., Sheth, A., Greenstein, B., Wetherall, D., Maganis, G., & Kohno, T. (2008). Privacy oracle: A system for finding application leaks with black-box differential testing. In *Proceedings of the 15th ACM Conference on Computer and Communications Security (CCS),* Alexandria, VA.

Ke, R., Zhuang, Y., Pu, Z., & Wang, Y. (2021). A smart, efficient, and reliable parking surveillance system with edge artificial intelligence on IoT devices. IEEE Transactions on Intelligent Transportation Systems, 22(8), 5054–5065.

Khan, I., & Bhatti, M. K. L. (2023). Unleashing the Power of IoT: Applications and Future Prospects in Healthcare, Agriculture, Smart Homes, Smart Cities, and Industry 4.0. *Sensors, 23*(16), 7194. https://doi.org/10.3390/s23167194

Kim, D., Park, J. H., Park, J., Kim, T. H., & Jung, E. (2019). IoT-based smart school surveillance system for student safety. *Journal of Ambient Intelligence and Humanized Computing, 10*(3), 865-874.

Lomotey, R., Sofranko, K., & Orji, R. (2018). Enhancing privacy in wearable IoT through a provenance architecture. Multimodal Technologies and Interaction.

Lulla, G., Kumar, A., Deshmukh, G., & Pole, G. (2021). IoT-based smart security and surveillance system. Proceedings of the 2021 International Conference on Emerging Smart Computing and Informatics.

Luo, Y., Yu, C., Li, J., & El-Sheimy, N. (2019). Performance of GNSS carrier-tracking loop based on Kalman filter in a challenging environment. ISPRS Archives, XLII-2/W13, 1687–1693. https://doi.org/10.5194/isprs-archives-XLII-2-W13-1687-2019

Maturkar, P., Nandanwar, N., Rahangdale, M., Lute, S., & Zile, S. (2020). Student monitoring system for boarding and leaving bus. *International Journal of Progressive Research in Science and Engineering, 1*(1). Retrieved from http://www.ijprse.com

Mohanta, B. K., Jena, D., Ramasubbareddy, S., Daneshmand, M., & Gandomi, A. H. (2021). Addressing security and privacy issues of IoT using blockchain technology. *IEEE Internet of Things Journal, 8*(2), 881-888. https://doi.org/10.1109/JIOT.2020.3029625

Mouha, R. A. (2021). Internet of Things (IoT). *Journal of Data Analysis and Information Processing, 9*(2), 77–101. https://doi.org/10.4236/jdaip.2021.92006

Peffers, K., Tuunanen, T., Rothenberger, M. A., & Chatterjee, S. (2006). A design science research methodology for information systems research. *Journal of Management Information Systems, 24*(3), 45-77.

Perwej, Y., Haq, K., Parwej, F., & Hassan, M. M. M. (2019). The Internet of Things (IoT) and its application domains. International Journal of Computer Applications, 182(49), 1-7. https://doi.org/10.5120/ijca2019918712

Preuveneers, D., & Joosen, W. (2016). Privacy-enabled remote health monitoring applications for resource-constrained wearable devices. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing (SAC)*, Pisa, Italy.

Rahman, M. A., Alam, M. S., & Rahman, M. M. (2021). IoT-Based Monitoring and Management Systems for Schools: Challenges and Future Directions. *Journal of Systems and Software*.

Ranga Rao, A., & Anjaiah, P. (2023). IoT-based school children transportation safety system. *Journal of Electronics and Communication Engineering*.

Satapathy, U., Mohanta, B. K., Jena, D., & Sobhanayak, S. (2018). An ECC-based lightweight authentication protocol for mobile phone in smart home. In *Proc. IEEE 13th Int. Conf. Ind. Inf. Syst. (ICIIS)* (pp. 303–308).

Senthamilarasi, N., Bharathi, N. D., Ezhilarasi, D., & Sangavi, R. B. (2019). Child Safety Monitoring System based on IoT. *Journal of Physics: Conference Series, 1362*(1), 012012. https://doi.org/10.1088/1742-6596/1362/1/012012

Shafique, K., Khawaja, B. A., Sabir, F., Qazi, S., & Mustaqim, M. (2020). Internet of Things (IoT) for next-generation smart systems: A review of current challenges, future trends and prospects for emerging 5G-IoT scenarios. *Access*, 8(1), 1-29. https://doi.org/10.1109/ACCESS.2020.2970118

Sivarathinabala, M., Chitra, R. J., Sivanesan, J., Sundar, M., & Udhaya Kumar, C. (2024). *IoT based smart students tracking and attendance system using RFID technology*. *AIP Conference Proceedings, 2935*(1), 020002. https://doi.org/10.1063/5 .0198831

Tamilselvan, S., Ramesh, R., Niveda, R., Poonguzhali, P., & Dharani, S. (2021). IoT based touch-free attendance system (ITAS). *Journal of Physics: Conference Series, 1917*(1), 012012. https://doi.org/10.1088/1742-6596/1917/1/012012

Tesfaye, F. (2020). IoT based children monitoring system in school. Retrieved from https://www.researchgate.net/publication/343712876

Wu, T., Redoute, J.-M., & Yuce, M. R. (2017). An autonomous wireless body area network implementation towards IoT connected healthcare applications. IEEE Access.

Zhang, Y., Zhang, X., & Liu, J. (2023). Internet of Things (IoT) for next-generation safety monitoring in educational environments. *IEEE Access*, 11, 22034-22049. https://doi.org/10.1109/ACCESS.2023.3150031

APPENDICES

Appendix I: System Source Code

Back End

```
from datetime import datetime, timedelta
import requests
from bson import ObjectId
from flask import Flask, request, jsonify, render_template, flash, redirect, url_for, session
from flask_login import LoginManager, UserMixin, login_required, logout_user,
login_user
from flask_mqtt import Mqtt
from database_manager import Database as db
from flask_apscheduler import APScheduler
import bcrypt
app = Flask(__name__)
app.secret_key = 'super secret key'
app.config['MQTT_BROKER_URL'] = 'broker.emqx.io'
app.config['MQTT_BROKER_PORT'] = 1883
app.config['MQTT_USERNAME'] = ''  # Set this item when you need to verify
username and password
app.config['MQTT_PASSWORD'] = ''  # Set this item when you need to verify username
and password
app.config['MQTT_KEEPALIVE'] = 30  # Set KeepAlive time in seconds
app.config['MQTT_TLS_ENABLED'] = False  # If your server supports TLS, set it True
mqtt_client = Mqtt(app)
TOPIC = '/students_report/#'
NOTIFY_TOPIC="/notify"
STUDENTS_COLLECTION = "students"
DEVICES_COLLECTION = "devices"
USER_TABLE="user"
LOCATION_LOGS_COLLECTION = "location_logs"
REVERSE_GEO_URL="https://api-bdc.net/data/reverse-geocode-client"
scheduler = APScheduler()
```

```
db.init()
login_manager = LoginManager()
login_manager.login_view = 'login'
login_manager.init_app(app)
scheduler.init_app(app)
scheduler.start()
class User(UserMixin):
    pass
@login_manager.user_loader
def load_user(user_id=None):
    if not user_id:
        user_id = session.get("user_id")
        if not user_id:
            return None
    user_info = db.find_by_id(USER_TABLE, user_id)
    user_class = User()
    user_class.id = str(user_info["_id"])
    user_class.email = user_info["email"]
    user_class.role = user_info["role"]
    user_class.name = user_info["name"]
    db.update_by_id(USER_TABLE, user_id, {"lastActive": datetime.utcnow()} )
    return user_class
ROLES = [
    {
        "name": "Admin",
        "value": "admin"
    },
    {
        "name": "User",
        "value": "user"
    }
]
@app.route('/users')
```

```
@login_required
def users():
    available_users = db.find(USER_TABLE,{})
    return render_template('users.html',
                           users=available_users,
                           roles=ROLES)
@app.route("/logout")
@login_required
def logout():
    logout_user()
    flash("Logout successful", "success")
    return redirect(url_for("login"))
@app.route("/user/add", methods=["GET", "POST"])
@login_required
def add_user():
    if request.method == "POST":
        email = request.form.get("email")
        password = request.form.get("password")
        name = request.form.get("name")
        role = request.form.get("role")
        if not email or not password or not name or not role:
            flash("Please fill all required fields", "error")
            return redirect(request.referrer)
        if len(password) < 8:
            flash("Password must be at least 8 characters", "error")
            return redirect(request.referrer)
        hashed_password = bcrypt.hashpw(password.encode('utf-8'), bcrypt.gensalt())
        db.insert(USER_TABLE, {"email": email, "password": hashed_password, "name":
name, "role": role,
                                   "addedOn": datetime.utcnow()})
        flash("User added successfully", "success")
        return redirect(request.referrer)
    return render_template("signup.html")
```

```
@mqtt_client.on_connect()
def handle_connect(client, userdata, flags, rc):
    if rc == 0:
        # print("Connected successfully")
        mqtt_client.subscribe(TOPIC)  # subscribe topic
    else:
        print('Bad connection. Code:', rc)
def format_phone_number(phone_number):
    if not phone_number:
        return None
    if phone_number.startswith("+254"):
        return phone_number
    if phone_number.startswith("254"):
        return "+254" + phone_number[3:]
    if phone_number.startswith("0"):
        return "+254" + phone_number[1:]
    return phone_number
@scheduler.task('interval', id='missing_students', seconds=30)
def check_missing_student():


    students = db.find(STUDENTS_COLLECTION, {
        "lastSeen": {"$lt": datetime.utcnow() - timedelta(minutes=10)}
    })
    for student in students:
        if not student.get("missing_notify"):
            # print("Missing notification disabled")
            continue
        if student.get("last_missing_notification") and
student.get("last_missing_notification") > datetime.utcnow() - timedelta(hours=24):
            # print("Already sent notification")
            continue
        parent_phone = format_phone_number(student.get("parent_phone"))
        if not parent_phone:
          # print("""Parent phone number not found for student:
{}""".format(student.get("name")))
          continue
       last_seen = student.get("lastSeen")
       student_name = student.get("name")
       message=f"{parent_phone}|{student_name} might be missing. Last seen at
{last_seen}"
       mqtt_client.publish(topic=NOTIFY_TOPIC, payload=message )
       db.update_by_id(STUDENTS_COLLECTION, student.get("_id"),
{"last_missing_notification": datetime.utcnow()})
@mqtt_client.on_message()
def handle_mqtt_message(client, userdata, message):
   """
   Message Structure: Student_id|latitude|longitude|updateStatus|timestamp
   Example message :  100|0.00|0.00|0|1724061579
   :param client:
   :param userdata:
   :param message:
   :return:
   """
   # print("Message received: " + message.topic + " -> " + message.payload.decode())
   data = dict(
       topic=message.topic,
       payload=message.payload.decode()
   )
   student_data= data.get("payload").split("|")
   if not student_data or len(student_data) < 5:
       # print("Invalid message format")
       return
   device_info = db.find_one(DEVICES_COLLECTION, {"deviceId": student_data[0]})
   if not device_info:
       # print ("Device not found")
       return
```

```
    student_id =device_info["studentId"]
    if not student_id:
        # print ("Student not found")
        return
    locality="unknown"
    if student_data[3] == "1":
        student_location=requests.get(REVERSE_GEO_URL,
params={"latitude":student_data[1],"longitude":student_data[2],
                                                    "localityLanguage":"en"}).json()
        locality=student_location.get("locality")
    db.update_by_id(STUDENTS_COLLECTION, student_id, {"latitude":
student_data[1],
                                             "longitude": student_data[2],
                                             "updateStatus": student_data[3],
                                             "timestamp": student_data[4],
                                             "lastSeen": datetime.utcnow(),
                                             "location":locality
                                             })
    relay_id =data.get("topic").split("/")[-1]S
    db.insert(LOCATION_LOGS_COLLECTION, {
        "student_id": student_data[0],
        "latitude": student_data[1],
        "longitude": student_data[2],
        "updateStatus": student_data[3],
        "timestamp": student_data[4],
        "addedAt": datetime.utcnow(),
        "relayId": relay_id
    })
    # print('Received message on topic: {topic} with payload: {payload}'.format(**data))
@app.route('/publish', methods=['POST', 'GET'])
@login_required
def publish_message():
    if request.method == 'POST':
        request_data = request.get_json()
        publish_result = mqtt_client.publish(request_data['topic'], request_data['msg'])
        return jsonify({'code': publish_result[0]})
    else:
        from time import time
        log_topic = "/logs/Error"
        device_id = "123456"
        message = f"{time()}| Command| 127.0.0.1 |{device_id} | Light off"
        publish_result = mqtt_client.publish(topic=log_topic, payload=message,
                                             )
        return "success"
@app.route('/')
def index():  # put application's code here
    flash('Welcome to TrackStu!', 'success')
    return render_template('index.html')
@app.route("/student/<student_id>/missing/toggle")
@login_required
def toggle_missing(student_id):
    student=db.find_by_id(STUDENTS_COLLECTION, student_id)
    if not student:
        flash("Student not found", "error")
        return redirect(request.referrer)
    if not student.get("missing_notify"):
        db.update_by_id(STUDENTS_COLLECTION, student_id, {
            "missing_notify": True
        })
        flash("Student missing notification enabled", "success")
    else:
        db.update_by_id(STUDENTS_COLLECTION, student_id, {
            "missing_notify":False
        })
        flash("Student missing notification disabled", "success")
    return redirect(request.referrer)
```

```
@app.route('/students')
@login_required
def get_students():
    students = db.find(STUDENTS_COLLECTION, {})
    return render_template('students.html', students=students)
@app.route('/student/<student_id>')
@login_required
def get_student(student_id):
    student = db.find_by_id(STUDENTS_COLLECTION, student_id)
    return jsonify(student)
@app.route('/student/add', methods=['POST', 'GET'])
@login_required
def add_student():
    if request.method == 'POST':
        name = request.form.get('student-name')
        registration_number = request.form.get('registration')
        class_name = request.form.get('grade')
        parent_phone = request.form.get('parent-phone')
        parent_name = request.form.get('parent-name')
        student_location = request.form.get('student-location')
        if not name or not registration_number or not class_name or not parent_phone or
not parent_name or not student_location:
            flash('Please fill in all the fields', 'error')
            return redirect(url_for('add_student'))
        db.insert(STUDENTS_COLLECTION, {
            'name': name,
            'registration_number': registration_number,
            'class_name': class_name,
            'parent_phone': parent_phone,
            "addedAt": datetime.utcnow(),
            "parent_name": parent_name,
            "home":student_location
        })
        flash('Student added successfully', 'success')
        return redirect(url_for('get_students'))
    return render_template('add_student.html')
@app.route('/devices')
@login_required
def get_devices():
    devices = db.find(DEVICES_COLLECTION, {})
    return render_template('devices.html', devices=devices)
def check_password(hashed_password, user_password):
    # Check the password
    return bcrypt.checkpw(user_password.encode('utf-8'), hashed_password)
@app.route("/login", methods=["POST","GET"])
def login():
    if request.method == "POST":
        username = request.form.get("email")
        password = request.form.get("password")
        next_url = request.args.get("next")
        # print(next_url)
        if not username or not password:
            flash("Please fill all the fields", "error")
            return redirect(request.referrer)
        user = db.find_one(USER_TABLE, {"email": username})
        if not user:
            flash("Invalid credentials", "error")
            return redirect(request.referrer)
        user_class = User()
        user_class.id = str(user["_id"])
        user_class.email = user["email"]
        if user.get("lockedUntil") and user.get("lockedUntil") > datetime.utcnow() and
user.get("failedLoginAttempts", 0) >= 3:
            flash(f"Your account has been locked. Please try again after
{user.get('lockedUntil').strftime('%Y-%m-%d %H:%M:%S')} UTC", "error")
            return redirect(request.referrer)
        if not check_password(user["password"], password):
            # Increment failed login attempts
            last_attempt=user.get("failedLoginAttempts", 0)
            db.update(USER_TABLE, {"_id": ObjectId(user["_id"])},
{"failedLoginAttempts": last_attempt+1})
            if user.get("failedLoginAttempts", 0) >= 3:
                db.update(USER_TABLE,{"_id": ObjectId(user["_id"])},
                                {"lockedUntil": datetime.utcnow() + timedelta(hours=5)})
                flash("Your account has been locked. Please try again after 5 hours", "error")
            else:
                attempts = 3 - user.get("failedLoginAttempts", 0)
                flash(f"Invalid credentials, you have {attempts} attempts left", "error")
            return redirect(request.referrer)
        if user and check_password(user["password"], password):
            login_user(user_class)
            db.update(USER_TABLE, {"_id": ObjectId(user["_id"]) },
{"failedLoginAttempts": 0})
            flash("Login successful", "success")
            return redirect(next_url or url_for("index"))
    return render_template("login.html")
@app.route('/device/<device_id>')
@login_required
def get_device(device_id):
    device = db.find_by_id(DEVICES_COLLECTION, device_id)
    device_logs = db.find(LOCATION_LOGS_COLLECTION, {"student_id":
device['deviceId']})
    return render_template('device.html', device=device, device_logs=device_logs)
@app.route('/device/<device_id>/update', methods=['POST', 'GET'])
@login_required
def update_device(device_id):
    if request.method == 'POST':
        device = db.find_by_id(DEVICES_COLLECTION, device_id)
        device['name'] = request.form.get('device-name')
        device['deviceId'] = request.form.get('device-id')
        device['studentId'] = request.form.get('assigned-student')
        db.update(DEVICES_COLLECTION, device_id, device)
        flash('Device updated successfully', 'success')
        return redirect(url_for('get_devices'))
    device = db.find_by_id(DEVICES_COLLECTION, device_id)
    return render_template('update_device.html', device=device)
@app.route('/device/<device_id>/delete')
@login_required
def delete_device(device_id):
    db.remove_by_id(DEVICES_COLLECTION, device_id)
    flash('Device deleted successfully', 'success')
    return redirect(url_for('get_devices'))
@app.route('/device/add', methods=['POST', 'GET'])
@login_required
def add_device():
    if request.method == 'POST':
        name = request.form.get('device-name')
        device_id = request.form.get('device-id')
        assigned_student_reg = request.form.get('assigned-student')
        if not name or not device_id or not assigned_student_reg:
            flash('Please fill in all the fields', 'error')
            return redirect(url_for('add_device'))
        student_details = db.find_one(STUDENTS_COLLECTION, {'registration_number':
assigned_student_reg})
        if not student_details:
            flash('Invalid student registration number', 'error')
            return redirect(url_for('add_device'))
        db.insert(DEVICES_COLLECTION, {
            'name': name,
            'deviceId': device_id,
            'studentId': student_details['_id'],
            "studentName": student_details['name'],
            "addedAt": datetime.utcnow()
        })
        flash('Device added successfully', 'success')
        return redirect(url_for('get_devices'))
    students=db.find(STUDENTS_COLLECTION, {})
    return render_template('add_device.html',
                           students=students)
if __name__ == '__main__':
    scheduler.init_app(app)
    scheduler.start()
    app.run()
```

Hardware code

```
#include <ESP8266WiFi.h>
#include <espnow.h>s
#include <ESP8266WiFi.h>
#include <PubSubClient.h>
#include <NTPClient.h>
#include <WiFiUdp.h>
#include <SoftwareSerial.h>
#define simTx D7
#define simRx D6
//Create software serial object to communicate with SIM800L
SoftwareSerial mySerial(simTx, simRx);  //SIM800L Tx & Rx is connected to Arduino #3 & #2
// Update these with values suitable for your network.
const char* ssid = "maize";
const char* password = "Maize@123";
const char* mqtt_server = "broker.emqx.io";
WiFiClient espClient;
PubSubClient client(espClient);
unsigned long lastMsg = 0;
#define MSG_BUFFER_SIZE (100)
#define LOG_CHANNEL "/logs/1"
```

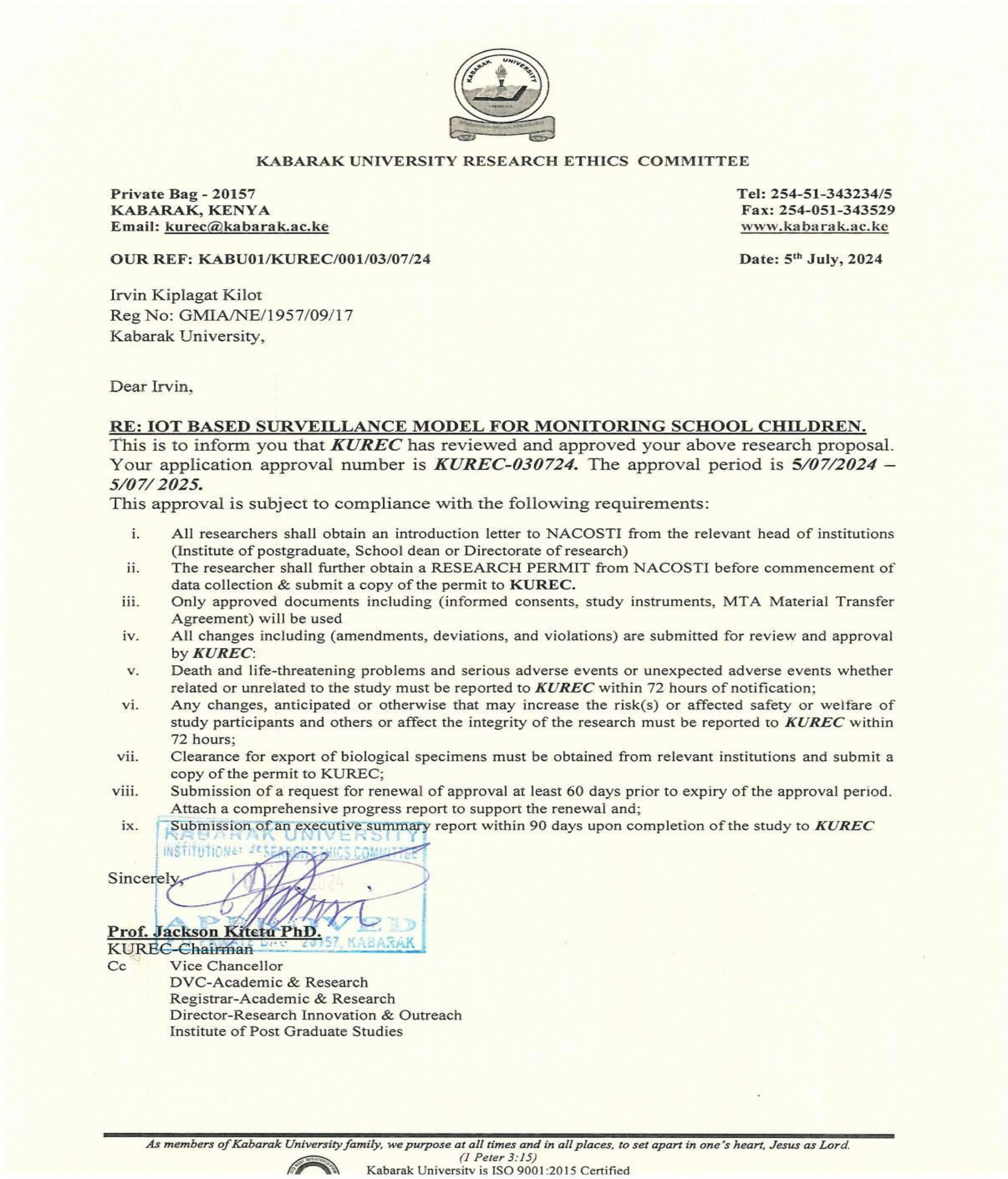
```
#define SUDENT_REPORT "/students_report/1"
#define COMMAND_CHANNEL "/commandStu"
#define NOTIFY "/notify"
void sendSMS(String phone, String message ) {
  delay(100);
  mySerial.println("AT"); //Once the handshake test is successful, it will back to OK
  updateSerial();
  mySerial.println("AT+CMGF=1");  // Configuring TEXT mode
  updateSerial();
  // String phoneCommand="AT+CMGS=\"" + phone + "\"";
  // mySerial.println(phoneCommand);  //change ZZ with country code and xxxxxxxxxxx with phone number to sms
  mySerial.println("AT+CMGS=\""+phone+"\"");//change ZZ with country code and xxxxxxxxxxx with phone number to sms
  updateSerial();
  mySerial.print(message);  //text content
  updateSerial();
  // delay(100);
  mySerial.write(26);
  delay(100);
}
char msg[MSG_BUFFER_SIZE];
// Define NTP Client to get time
int utcOffsetInSeconds = 0;
WiFiUDP ntpUDP;
NTPClient timeClient(ntpUDP, "pool.ntp.org", utcOffsetInSeconds);
void setup_wifi() {
  delay(10);
  // We start by connecting to a WiFi network
  Serial.println();
  Serial.print("Connecting to ");
  Serial.println(ssid);
  WiFi.mode(WIFI_AP_STA);
```

```
 WiFi.begin(ssid, password);
  while (WiFi.status() != WL_CONNECTED) {
    delay(500);
    Serial.print(".");
  }
  Serial.println("");
  Serial.println("WiFi connected");
  Serial.println("IP address: ");
  Serial.println(WiFi.localIP());
}
String getValue(String data, char separator, int index) {
  int found = 0;
  int strIndex[] = { 0, -1 };
  int maxIndex = data.length() - 1;s
  for (int i = 0; i <= maxIndex && found <= index; i++) {
    if (data.charAt(i) == separator || i == maxIndex) {
      found++;
      strIndex[0] = strIndex[1] + 1;
      strIndex[1] = (i == maxIndex) ? i + 1 : i;
    }
  }
  return found > index ? data.substring(strIndex[0], strIndex[1]) : "";
}
void callback(char* topic, byte* payload, unsigned int length) {
  char payloadData[length];
  Serial.print("Message arrived [");
  Serial.print(topic);
  Serial.print("] ");
  for (int i = 0; i < length; i++) {
    payloadData[i] = (char)payload[i];
  }
  Serial.print(topic);
  Serial.print(NOTIFY);
  if (String(topic) == String(NOTIFY)) {
    // Serial.println("Notifying");
    String phone = getValue(String(payloadData), '|', 0);
    String message = getValue(String(payloadData), '|', 1);
    sendSMS(phone, message);
    Serial.println(" ");
    Serial.println(phone);
    Serial.println(message);
  } else {
    for (int i = 0; i < length; i++) {
      Serial.print((char)payload[i]);
    }
    Serial.println();
    // Switch on the LED if an 1 was received as first character
    if ((char)payload[0] == '1') {
      digitalWrite(BUILTIN_LED, LOW);   // Turn the LED on (Note that LOW is the
voltage level
      // but actually the LED is on; this is because
      // it is active low on the ESP-01)
    } else {
      digitalWrite(BUILTIN_LED, HIGH);  // Turn the LED off by making the voltage
HIGH
    }
  }
  client.publish(LOG_CHANNEL, payloadData);
  delay(1000);
}
void reconnect() {
  // Loop until we're reconnected
  while (!client.connected()) {
    Serial.print("Attempting MQTT connection...");
    // Create a random client ID
    String clientId = "ESP8266Client-";
    clientId += String(random(0xffff), HEX);
    // Attempt to connect
    if (client.connect(clientId.c_str())) {
      Serial.println("connected");
      // Once connected, publish an announcement...
      client.publish(LOG_CHANNEL, "Connected");
      // ... and resubscribe
      client.subscribe(COMMAND_CHANNEL);
      client.subscribe(NOTIFY);
    } else {
      Serial.print("failed, rc=");
      Serial.print(client.state());
      Serial.println(" try again in 5 seconds");
      // Wait 5 seconds before retrying
      delay(5000);
    }
  }
}
typedef struct {
  float latitude;
  float longitude;
  float altitude;
  float speed;
  int satellites;
  int hdop;
  int updated;
  int id;
} struct_message;
struct_message gpsData;
// Callback when data is received
void OnDataRecv(uint8_t* mac_addr, uint8_t* incomingData, uint8_t len) {
  memcpy(&gpsData, incomingData, sizeof(gpsData));
  unsigned long time = timeClient.getEpochTime();
  String data = String(gpsData.id) + "|" + String(gpsData.latitude) + "|" + String(gpsData.longitude) + "|" + String(gpsData.updated) + "|" + String(time);
  data.toCharArray(msg, data.length() + 1);
  client.publish(SUDENT_REPORT, msg);
  // Print received GPS data to Serial Monitor
  Serial.print("Latitude: ");
  Serial.println(gpsData.latitude, 6);
  Serial.print("Longitude: ");
  Serial.println(gpsData.longitude, 6);
  Serial.print("Altitude: ");
  Serial.println(gpsData.altitude);
  Serial.print("Speed: ");
  Serial.println(gpsData.speed);
  Serial.print("Satellites: ");
  Serial.println(gpsData.satellites);
  Serial.print("HDOP: ");
  Serial.println(gpsData.hdop);
  Serial.print("Updated: ");
  Serial.println(gpsData.updated);
  Serial.print("Id: ");
  Serial.println(gpsData.id);
  Serial.println();
}
void setup() {
  // Start Serial Monitor
  Serial.begin(9600);
  delay(100);
  mySerial.begin(9600);
  delay(1000);
  Serial.println("Setup started");
  WiFi.mode(WIFI_AP_STA);
  setup_wifi();
  timeClient.begin();
  timeClient.update();
  client.setServer(mqtt_server, 1883);
  client.setCallback(callback);
    if (esp_now_init() != 0) {
    Serial.println("Error initializing ESP-NOW");
    return;
  } else {
    Serial.println("ESP-NOW Initialization successful");
  }
  esp_now_register_recv_cb(esp_now_recv_cb_t(OnDataRecv));
  // esp_now_register_recv_cb(OnDataRecv);
  delay(1000);
  Serial.println("Done with setup");
}
void loop() {
  // Main loop does nothing as all work is done in the callback function
  if (!client.connected()) {
    reconnect();
    snprintf(msg, MSG_BUFFER_SIZE, "%ld | Info |Reporting for duty", 23);
    Serial.print("Publish message: ");
    Serial.println(msg);
    client.publish(LOG_CHANNEL, msg);
  }
  client.loop();
}
void updateSerial()
{
  delay(500);
  while (Serial.available())
  {
    mySerial.write(Serial.read());//Forward what Serial received to Software Serial Port
  }
  while(mySerial.available())
  {
    Serial.write(mySerial.read());//Forward what Software Serial received to Serial Port
  }
}
#include <ESP8266WiFi.h>
#include <espnow.h>
#include <TinyGPS++.h>
#include <SoftwareSerial.h>
// GPS Setup
TinyGPSPlus gps;
SoftwareSerial gpsSerial(D2, D1); // RX, TX (D2, D1)
constexpr char WIFI_SSID[] = "Wegner";
// Receiver MAC Address
uint8_t receiverAddress[] = {0x08, 0xF9, 0xE0, 0x75, 0xE2, 0xC5}; // Replace 'XX' with the correct value
int32_t getWiFiChannel(const char *ssid) {
  if (int32_t n = WiFi.scanNetworks()) {
      for (uint8_t i=0; i<n; i++) {
          if (!strcmp(ssid, WiFi.SSID(i).c_str())) {
              return WiFi.channel(i);
          }
      }
  }
  return 0;
}
// Structure to hold GPS data
typedef struct {
  float latitude;
  float longitude;
  float altitude;
  float speed;
  int satellites;
  int hdop;
  int updated;
  int id=100;

} struct_message;
struct_message gpsData;
// Callback when data is sent
void OnDataSent(uint8_t *mac_addr, uint8_t sendStatus) {
  Serial.print("Last Packet Send Status: ");
  Serial.println(sendStatus == 0 ? "Delivery Success" : "Delivery Fail");
}
void setup() {
  // Start Serial Monitor
  Serial.begin(9600);
  gpsSerial.begin(9600);
  // Set device as a Wi-Fi Station
  WiFi.mode(WIFI_STA);
  int32_t channel = getWiFiChannel(WIFI_SSID);
WiFi.printDiag(Serial); // Uncomment to verify channel number before
  wifi_promiscuous_enable(1);
  wifi_set_channel(channel);
  wifi_promiscuous_enable(0);
  WiFi.printDiag(Serial); // Uncomment to verify channel change after
  // Init ESP-NOW
  if (esp_now_init() != 0) {
    Serial.println("Error initializing ESP-NOW");
    return;
  }
  // Set up send callback
  esp_now_set_self_role(ESP_NOW_ROLE_CONTROLLER);
  esp_now_register_send_cb(OnDataSent);
  // Register peer
  esp_now_add_peer(receiverAddress, ESP_NOW_ROLE_SLAVE, 1, NULL, 0);
}
```

```
void readSendGpsData(){
 while (gpsSerial.available() > 0) {
    gps.encode(gpsSerial.read());
    if (gps.location.isUpdated()) {
      // Fill the structure with GPS data
      gpsData.latitude = gps.location.lat();
      gpsData.longitude = gps.location.lng();
      gpsData.altitude = gps.altitude.meters();
      gpsData.speed = gps.speed.kmph();
      gpsData.satellites = gps.satellites.value();
      gpsData.hdop = gps.hdop.value();
      // Print GPS data to Serial Monitor
      Serial.print("Latitude: ");
      Serial.println(gpsData.latitude, 6);
      Serial.print("Longitude: ");
      Serial.println(gpsData.longitude, 6);
      Serial.print("Altitude: ");
      Serial.println(gpsData.altitude);
      Serial.print("Speed: ");
      Serial.println(gpsData.speed);
      Serial.print("Satellites: ");
      Serial.println(gpsData.satellites);
      Serial.print("HDOP: ");
      Serial.println(gpsData.hdop);
      Serial.println();
      // Send the GPS data via ESP-NOW
    }
else {
gpsData.latitude = 0;
      gpsData.longitude = 0;
      gpsData.altitude = 0;
      gpsData.speed = 0;
      gpsData.satellites =0;
      gpsData.hdop =0;
      gpsData.updated=0;
} esp_now_send(receiverAddress, (uint8_t *)&gpsData, sizeof(gpsData));
  }   esp_now_send(receiverAddress, (uint8_t *)&gpsData, sizeof(gpsData));
}
unsigned long initial=0;
int period =1000;
void loop() {
// Implement better delay
unsigned long now =millis();
if (now-initial>=period){
readSendGpsData();
initial=millis();
}}
```

Appendix II: KUREC Clearance Letter



KABARAK UNIVERSITY RESEARCH ETHICS COMMITTEE

Private Bag - 20157	**Tel: 254-51-343234/5**
KABARAK, KENYA	**Fax: 254-051-343529**
Email: kurec@kabarak.ac.ke	**www.kabarak.ac.ke**

OUR REF: KABU01/KUREC/001/03/07/24 **Date: 5th July, 2024**

Irvin Kiplagat Kilot
Reg No: GMIA/NE/1957/09/17
Kabarak University,

Dear Irvin,

RE: IOT BASED SURVEILLANCE MODEL FOR MONITORING SCHOOL CHILDREN.

This is to inform you that ***KUREC*** has reviewed and approved your above research proposal. Your application approval number is ***KUREC-030724.*** The approval period is ***5/07/2024 – 5/07/ 2025.***

This approval is subject to compliance with the following requirements:

i. All researchers shall obtain an introduction letter to NACOSTI from the relevant head of institutions (Institute of postgraduate, School dean or Directorate of research)

ii. The researcher shall further obtain a RESEARCH PERMIT from NACOSTI before commencement of data collection & submit a copy of the permit to **KUREC.**

iii. Only approved documents including (informed consents, study instruments, MTA Material Transfer Agreement) will be used

iv. All changes including (amendments, deviations, and violations) are submitted for review and approval by ***KUREC***:

v. Death and life-threatening problems and serious adverse events or unexpected adverse events whether related or unrelated to the study must be reported to ***KUREC*** within 72 hours of notification;

vi. Any changes, anticipated or otherwise that may increase the risk(s) or affected safety or welfare of study participants and others or affect the integrity of the research must be reported to ***KUREC*** within 72 hours;

vii. Clearance for export of biological specimens must be obtained from relevant institutions and submit a copy of the permit to KUREC;

viii. Submission of a request for renewal of approval at least 60 days prior to expiry of the approval period. Attach a comprehensive progress report to support the renewal and;

ix. Submission of an executive summary report within 90 days upon completion of the study to ***KUREC***

Sincerely,

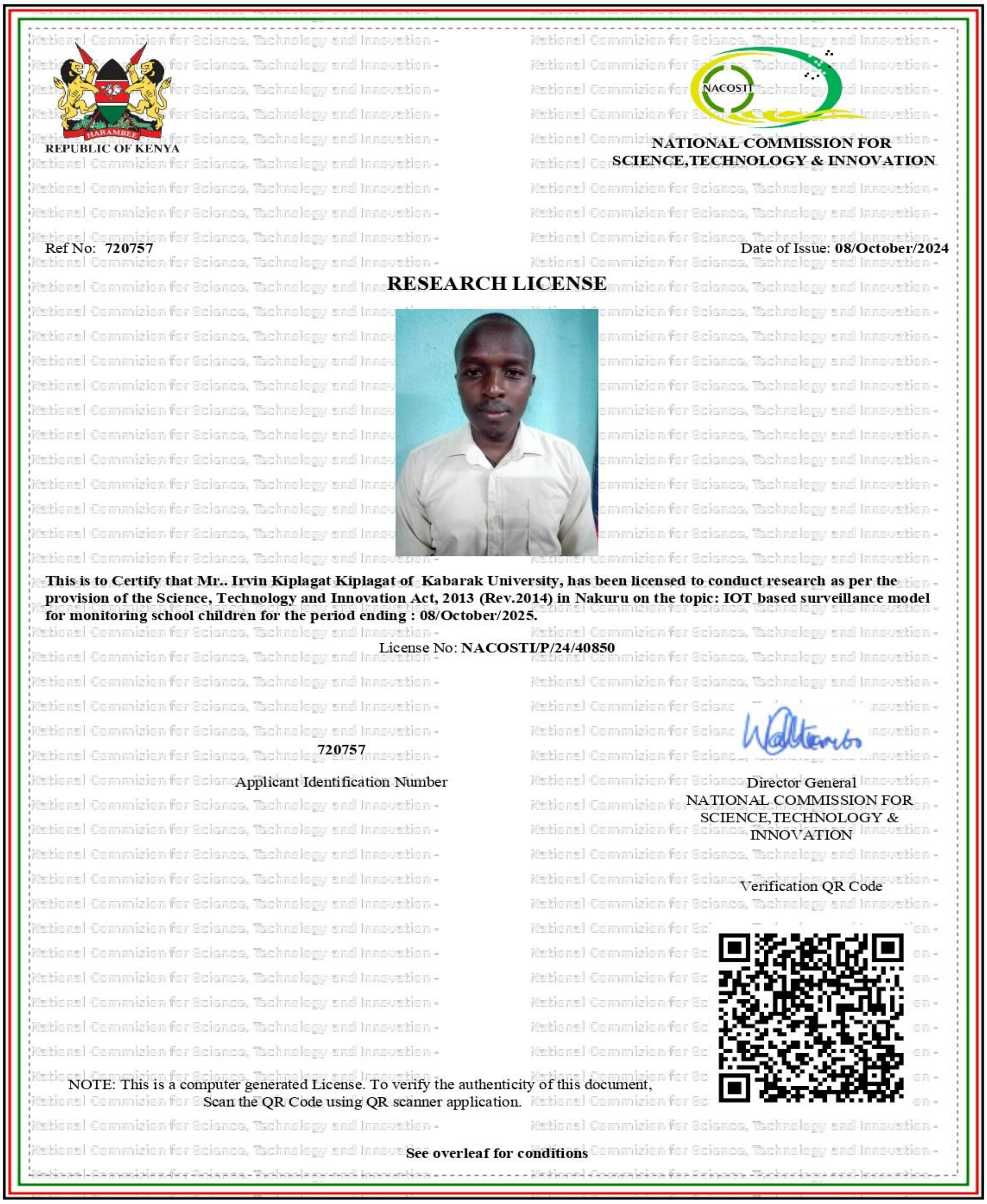
KABARAK UNIVERSITY
INSTITUTIONAL RESEARCH ETHICS COMMITTEE
2024
APPROVED
P.O. PRIVATE BAG 20157, KABARAK

Prof. Jackson Kitetu PhD.
KUREC-Chairman

Cc
Vice Chancellor
DVC-Academic & Research
Registrar-Academic & Research
Director-Research Innovation & Outreach
Institute of Post Graduate Studies

As members of Kabarak University family, we purpose at all times and in all places, to set apart in one's heart, Jesus as Lord.
(1 Peter 3:15)
Kabarak Universitv is ISO 9001:2015 Certified

Appendix III: NACOSTI Research Permit



REPUBLIC OF KENYA

NATIONAL COMMISSION FOR
SCIENCE,TECHNOLOGY & INNOVATION

Ref No: **720757**

Date of Issue: **08/October/2024**

RESEARCH LICENSE

This is to Certify that Mr.. Irvin Kiplagat Kiplagat of Kabarak University, has been licensed to conduct research as per the provision of the Science, Technology and Innovation Act, 2013 (Rev.2014) in Nakuru on the topic: IOT based surveillance model for monitoring school children for the period ending : 08/October/2025.

License No: **NACOSTI/P/24/40850**

720757

Applicant Identification Number

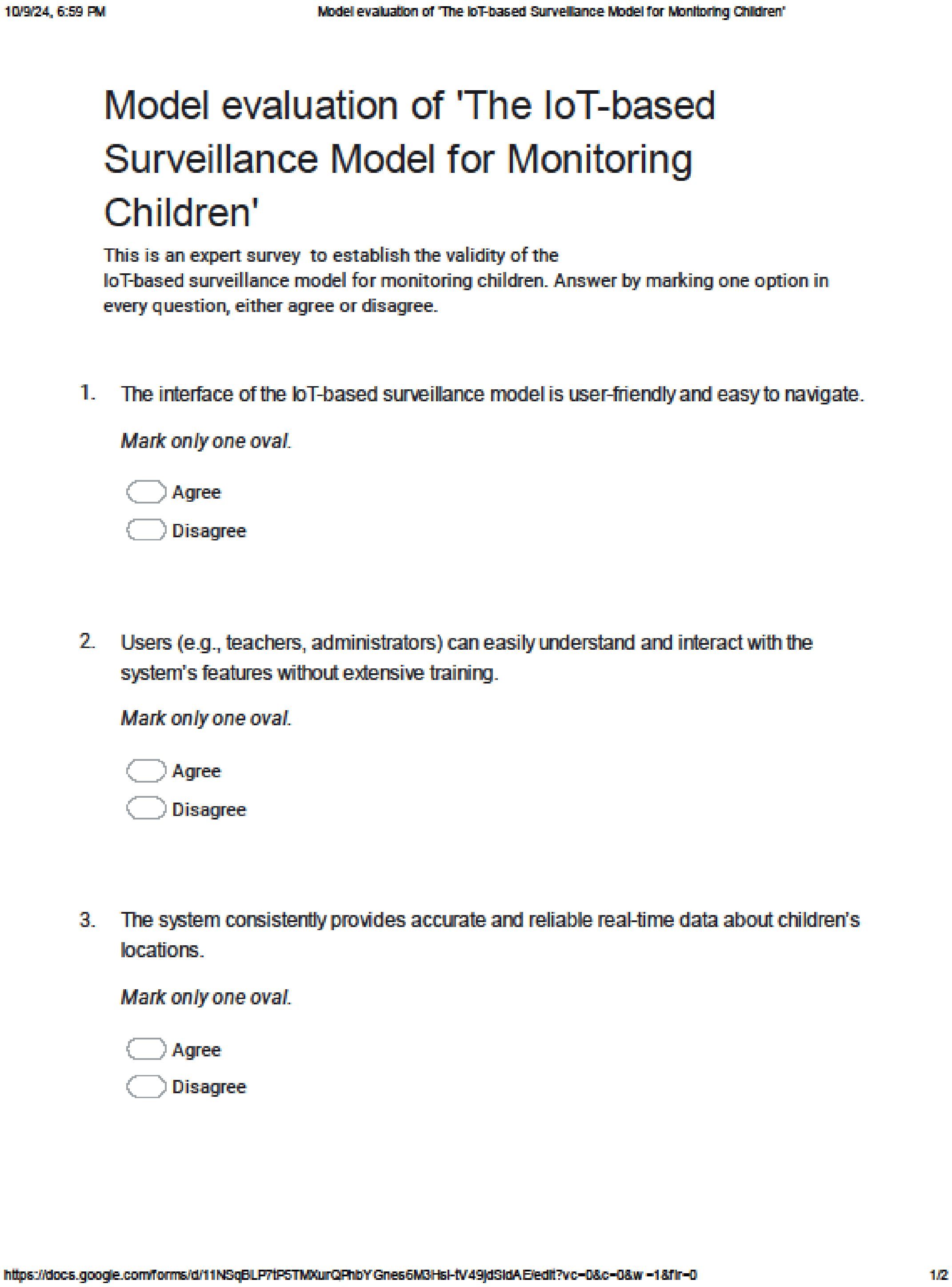
Director General
NATIONAL COMMISSION FOR
SCIENCE,TECHNOLOGY &
INNOVATION

Verification QR Code

NOTE: This is a computer generated License. To verify the authenticity of this document,
Scan the QR Code using QR scanner application.

See overleaf for conditions

Appendix IV: Expert Survey Form



10/9/24, 6:59 PM Model evaluation of 'The IoT-based Surveillance Model for Monitoring Children'

Model evaluation of 'The IoT-based Surveillance Model for Monitoring Children'

This is an expert survey to establish the validity of the IoT-based surveillance model for monitoring children. Answer by marking one option in every question, either agree or disagree.

1. The interface of the IoT-based surveillance model is user-friendly and easy to navigate.

   Mark only one oval.

   ○ Agree
   ○ Disagree

2. Users (e.g., teachers, administrators) can easily understand and interact with the system's features without extensive training.

   Mark only one oval.

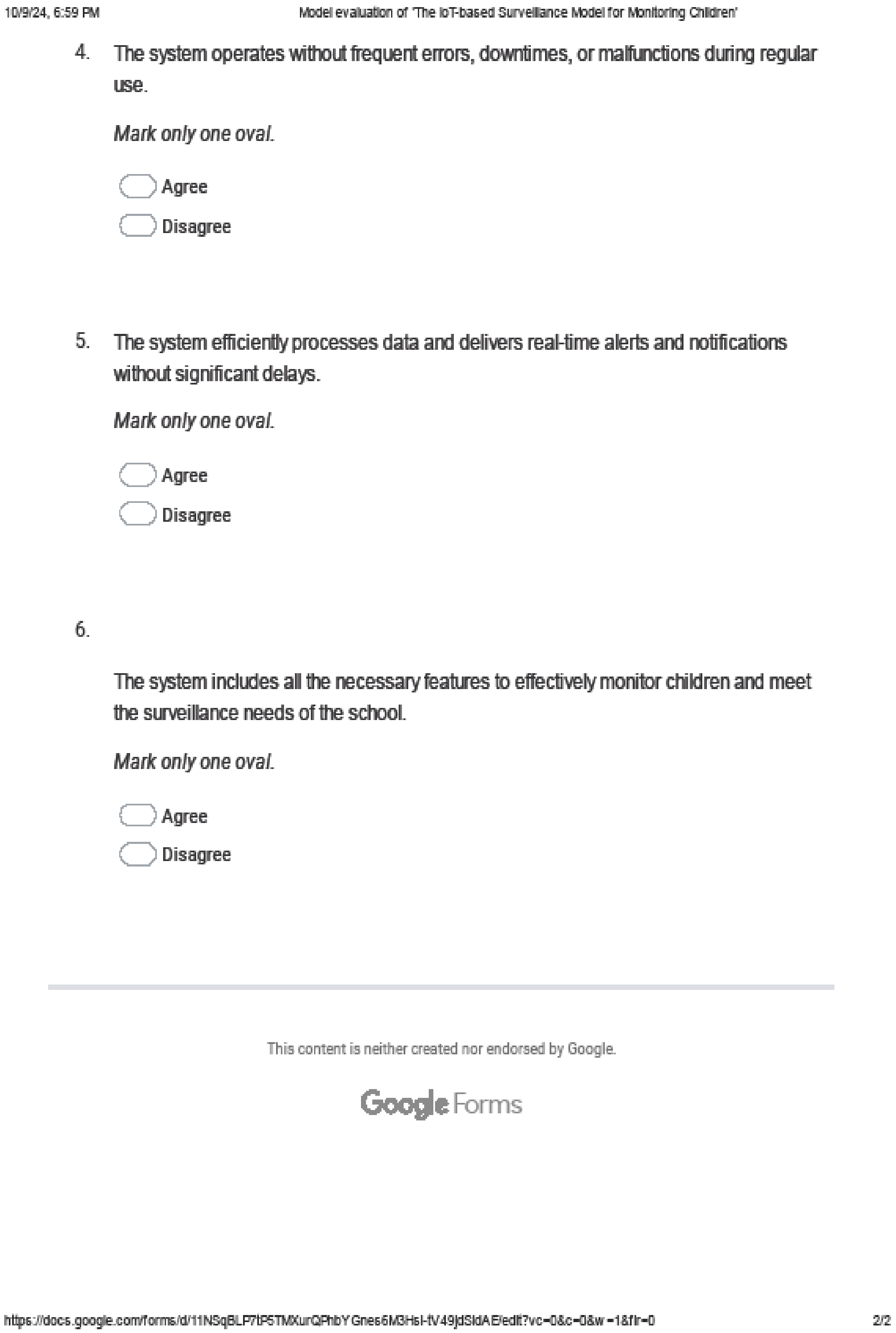
   ○ Agree
   ○ Disagree

3. The system consistently provides accurate and reliable real-time data about children's locations.

   Mark only one oval.

   ○ Agree
   ○ Disagree

https://docs.google.com/forms/d/11NSqBLP7fP5TMXurQPhbYGnes6M3Hsl-tV49jdSldAE/edit?vc=0&c=0&w=1&flr=0 1/2



10/9/24, 6:59 PM Model evaluation of 'The IoT-based Surveillance Model for Monitoring Children'

4. The system operates without frequent errors, downtimes, or malfunctions during regular use.

Mark only one oval.

○ Agree

○ Disagree

5. The system efficiently processes data and delivers real-time alerts and notifications without significant delays.

Mark only one oval.

○ Agree

○ Disagree

6.

The system includes all the necessary features to effectively monitor children and meet the surveillance needs of the school.

Mark only one oval.

○ Agree

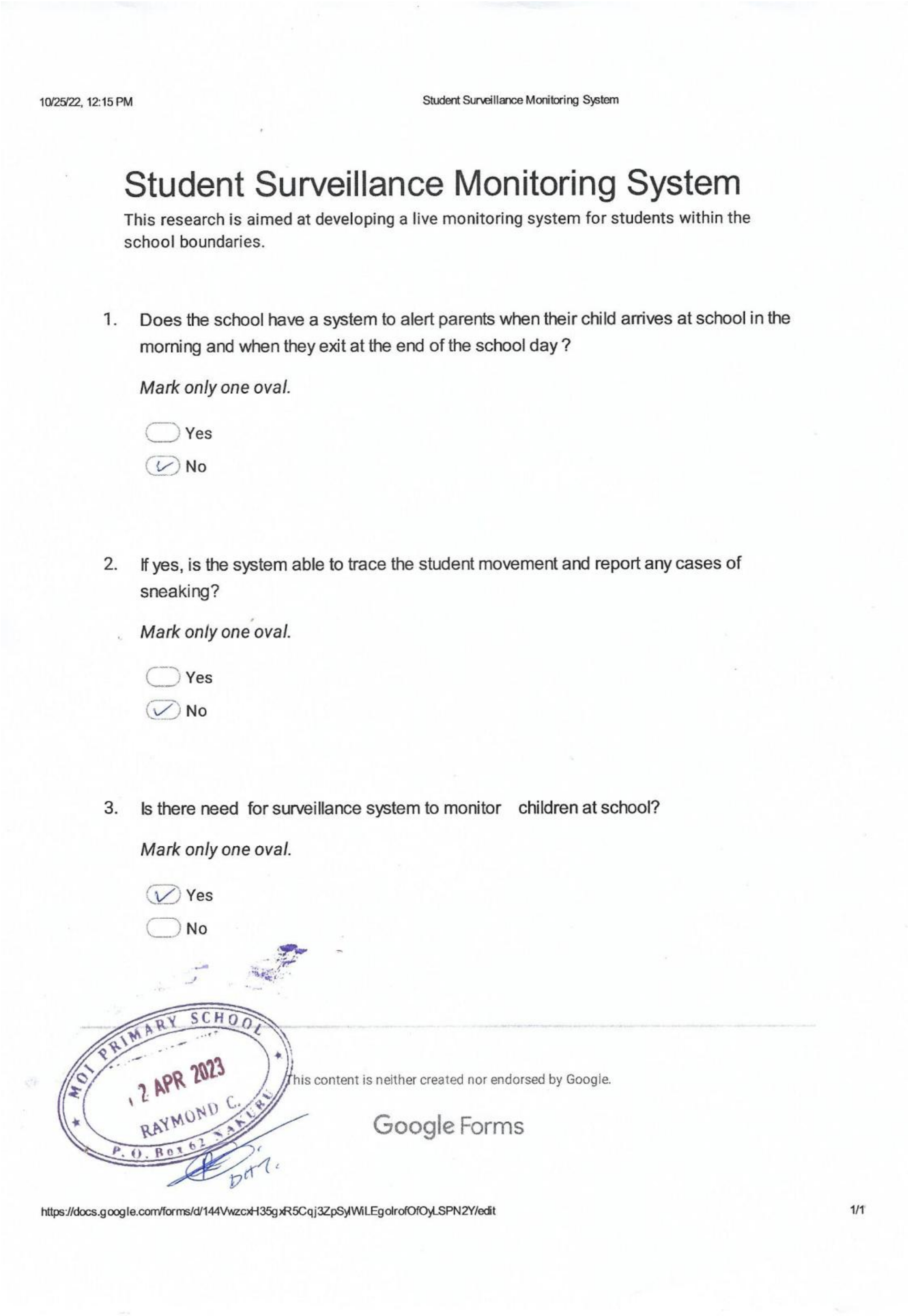
○ Disagree

This content is neither created nor endorsed by Google.

Google Forms

https://docs.google.com/forms/d/11NSqBLP7tP5TMXurQPhbYGnes6M3Hsl-tV49jdSldAE/edit?vc=0&c=0&w=1&flr=0 2/2

Appendix V: Pilot Study Exhibits



10/25/22, 12:15 PM Student Surveillance Monitoring System

Student Surveillance Monitoring System

This research is aimed at developing a live monitoring system for students within the school boundaries.

1. Does the school have a system to alert parents when their child arrives at school in the morning and when they exit at the end of the school day ?

   Mark only one oval.

   ○ Yes
   ✓ No

2. If yes, is the system able to trace the student movement and report any cases of sneaking?

   Mark only one oval.

   ○ Yes
   ✓ No

3. Is there need for surveillance system to monitor children at school?

   Mark only one oval.

   ✓ Yes
   ○ No

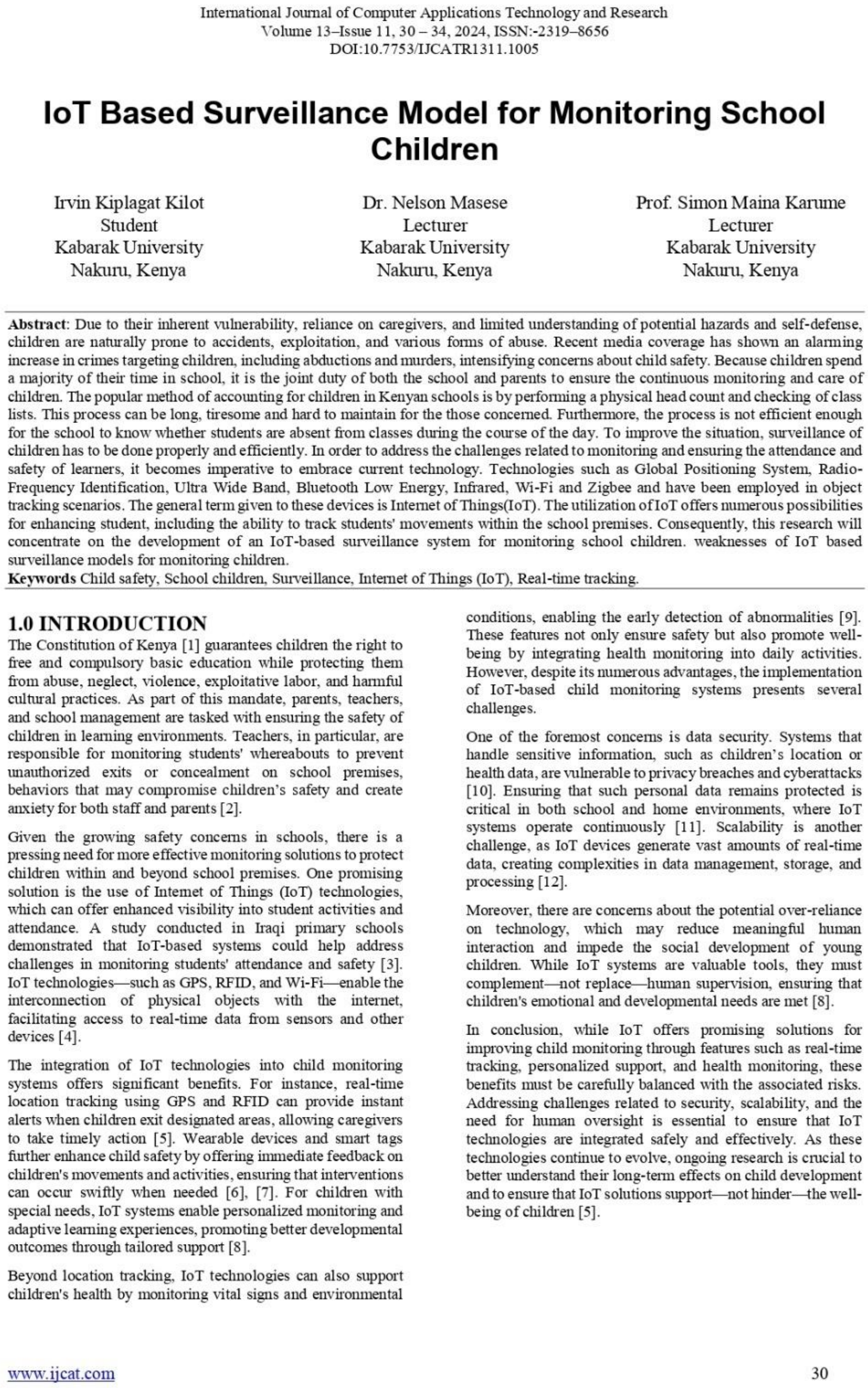
MOI PRIMARY SCHOOL
12 APR 2023
RAYMOND C.
P. O. Box 62 NAKURU

This content is neither created nor endorsed by Google.

Google Forms

https://docs.google.com/forms/d/144VwzcxH35gxR5Cqj3ZpSylWiLEgolrofOfOyLSPN2Y/edit 1/1

Appendix VII: List of Publication



International Journal of Computer Applications Technology and Research
Volume 13–Issue 11, 30 – 34, 2024, ISSN:-2319–8656
DOI:10.7753/IJCATR1311.1005

IoT Based Surveillance Model for Monitoring School Children

Irvin Kiplagat Kilot
Student
Kabarak University
Nakuru, Kenya

Dr. Nelson Masese
Lecturer
Kabarak University
Nakuru, Kenya

Prof. Simon Maina Karume
Lecturer
Kabarak University
Nakuru, Kenya

Abstract: Due to their inherent vulnerability, reliance on caregivers, and limited understanding of potential hazards and self-defense, children are naturally prone to accidents, exploitation, and various forms of abuse. Recent media coverage has shown an alarming increase in crimes targeting children, including abductions and murders, intensifying concerns about child safety. Because children spend a majority of their time in school, it is the joint duty of both the school and parents to ensure the continuous monitoring and care of children. The popular method of accounting for children in Kenyan schools is by performing a physical head count and checking of class lists. This process can be long, tiresome and hard to maintain for the those concerned. Furthermore, the process is not efficient enough for the school to know whether students are absent from classes during the course of the day. To improve the situation, surveillance of children has to be done properly and efficiently. In order to address the challenges related to monitoring and ensuring the attendance and safety of learners, it becomes imperative to embrace current technology. Technologies such as Global Positioning System, Radio-Frequency Identification, Ultra Wide Band, Bluetooth Low Energy, Infrared, Wi-Fi and Zigbee and have been employed in object tracking scenarios. The general term given to these devices is Internet of Things(IoT). The utilization of IoT offers numerous possibilities for enhancing student, including the ability to track students' movements within the school premises. Consequently, this research will concentrate on the development of an IoT-based surveillance system for monitoring school children. weaknesses of IoT based surveillance models for monitoring children.

Keywords Child safety, School children, Surveillance, Internet of Things (IoT), Real-time tracking.

1.0 INTRODUCTION

The Constitution of Kenya [1] guarantees children the right to free and compulsory basic education while protecting them from abuse, neglect, violence, exploitative labor, and harmful cultural practices. As part of this mandate, parents, teachers, and school management are tasked with ensuring the safety of children in learning environments. Teachers, in particular, are responsible for monitoring students' whereabouts to prevent unauthorized exits or concealment on school premises, behaviors that may compromise children's safety and create anxiety for both staff and parents [2].

Given the growing safety concerns in schools, there is a pressing need for more effective monitoring solutions to protect children within and beyond school premises. One promising solution is the use of Internet of Things (IoT) technologies, which can offer enhanced visibility into student activities and attendance. A study conducted in Iraqi primary schools demonstrated that IoT-based systems could help address challenges in monitoring students' attendance and safety [3]. IoT technologies—such as GPS, RFID, and Wi-Fi—enable the interconnection of physical objects with the internet, facilitating access to real-time data from sensors and other devices [4].

The integration of IoT technologies into child monitoring systems offers significant benefits. For instance, real-time location tracking using GPS and RFID can provide instant alerts when children exit designated areas, allowing caregivers to take timely action [5]. Wearable devices and smart tags further enhance child safety by offering immediate feedback on children's movements and activities, ensuring that interventions can occur swiftly when needed [6], [7]. For children with special needs, IoT systems enable personalized monitoring and adaptive learning experiences, promoting better developmental outcomes through tailored support [8].

Beyond location tracking, IoT technologies can also support children's health by monitoring vital signs and environmental conditions, enabling the early detection of abnormalities [9]. These features not only ensure safety but also promote well-being by integrating health monitoring into daily activities. However, despite its numerous advantages, the implementation of IoT-based child monitoring systems presents several challenges.

One of the foremost concerns is data security. Systems that handle sensitive information, such as children's location or health data, are vulnerable to privacy breaches and cyberattacks [10]. Ensuring that such personal data remains protected is critical in both school and home environments, where IoT systems operate continuously [11]. Scalability is another challenge, as IoT devices generate vast amounts of real-time data, creating complexities in data management, storage, and processing [12].

Moreover, there are concerns about the potential over-reliance on technology, which may reduce meaningful human interaction and impede the social development of young children. While IoT systems are valuable tools, they must complement—not replace—human supervision, ensuring that children's emotional and developmental needs are met [8].

In conclusion, while IoT offers promising solutions for improving child monitoring through features such as real-time tracking, personalized support, and health monitoring, these benefits must be carefully balanced with the associated risks. Addressing challenges related to security, scalability, and the need for human oversight is essential to ensure that IoT technologies are integrated safely and effectively. As these technologies continue to evolve, ongoing research is crucial to better understand their long-term effects on child development and to ensure that IoT solutions support—not hinder—the well-being of children [5].