DEVELOPMENT OF INDOOR INDUSTRIAL ENVIRONMENT MONITORING SYSTEM BASED ON WIRELESS SENSOR NETWORKS FOR A DIGITAL ASSEMBLY PLANT

Philemon Kiprop Rotich

A Dissertation Submitted In Partial Fulfillment of the Requirements for the Degree of Master`s of Science in Embedded and Mobile System of the Nelson Mandela African Institution of Science and Technology

Arusha, Tanzania

June 2021

ABSTRACT

The existing systems normally used to monitor the industrial environmental condition in factories in Kenya are mostly wired systems and more expensive. Despite the benefits of tools and emerging technology, most companies do not use automated systems due to the high installation and maintenance costs and a lack of formal early-warning mechanism.

This research project intends to provide information using wireless sensor technology, which comprises of sensors, Zigbee, raspberry pi, Arduino Uno, and wireless sensor network (WSN). The system is developed using open-source hardware raspberry pi and Zigbee which proves to be cost-effective and has low power consumption. The sensors will collect the data of various environmental parameters and transmit it to the raspberry pi, which acts as a base station. The raspberry pi will then transmit the data using Zigbee, and the processed data displayed on GUI through the Zigbee on the receiver end.

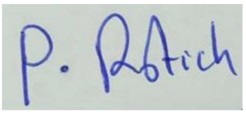
The database was developed using MYSQL DB and the web application written in the PHP programming language. In addidtion, the system is integrated with Zigbee technology to send SMS and e-mail notifications to the respective registered users.

The developed system was tested at different plant areas to detect the concentration of Carbon monoxide, temperature, humidity, and dust in the factory. The result shows that the system responds positively when these parameters were detected and sends notification alerts via e-mail and SMS. Finally, the data collected can be used for further data analysis and research in the future.

DECLARATION

I, Philemon Kiprop Rotich, do hereby declare to the Senate of the Nelson Mandela African Institution of Science and Technology that this dissertation is my own original work and that it has neither been submitted nor being concurrently submitted for degree award in any other institution.

Philemon Kiprop Rotich

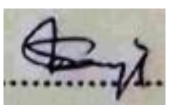


05-08-2021

Name and Signature of Candidate DATE

The above declaration is confirmed

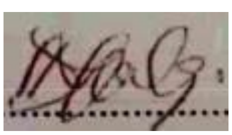
Dr. Shubi F. Kaijage



05-08-2021

Name and Signature of Main Supervisor DATE

Dr. Devota Nyambo



05-08-2021

Name and Signature of Co-Supervisor DATE

Dr. Kiyeng Chumo

5th August, 2021

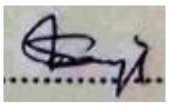
Name and Signature of Co supervisor DATE

COPYRIGHT

CERTIFICATION

The undersigned certify that they have read and found it to be acceptable for examination; hereby recommends for examination of the dissertation entitled: "*Development of Indoor Industrial Environment Monitoring System Based on Wireless Sensor Networks for a Digital Assembly Plant"* as a fulfillment of the Award for the Degree of Master`s of Science in Embedded and Mobile System at The Nelson Mandela African Institution of Science and Technology.

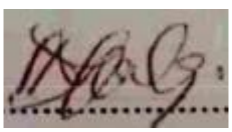
Dr. Shubi F. Kaijage



05-08-2021

Name and Signature of Main Supervisor DATE

Dr. Devota Nyambo



05-08-2021

Name and Signature of Co-Supervisor DATE

Dr. Kiyeng Chumo

05-08-2021

Name and Signature of Co supervisor DATE

DEDICATION

This work is dedicated to my parents, Mr. and Mrs. Francis Kiplagat, for the firm education foundation and encouragement during my entire course of study.

ACKNOWLEDGEMENTS

Firstly, I would like to thank the Almighty God for giving me the strength, understanding, and determination to accomplish my studies at the Nelson Mandela African Institution of Science and Technology (NM-AIST).

I would like to immensely thank my supervisors, Dr. Shubi Kaijage, Dr. Devotha Nyambo, and Dr. Kiyeng Chumo for valuable advice, constant support, and guidance during my research project work

I would also like to thank all faculty members and staff at the school of Computational and Communication Science and Engineering of NM-AIST for helping me during my study period.

My sincere gratitude also goes to the Centre of Excellence for ICT in East Africa (CENIT@EA) for offering me the scholarship that sponsored my coursework and research project towards my Masters‟ degree.

Special thanks to Dr. Kiyeng Chumo, the Project Manager Digital Assembly Plant (DLP) a facility of Moi University, for his assistance and guidance during the research project.

Lastly, many thanks to my classmates and the NM-AIST community at large for their encouragement and moral support during my stay at NM-AIST.

TABLE OF CONTENTS

ABSTRACT i

DECLARATION ii

COPYRIGHT iii

CERTIFICATION iv

DEDICATION v

ACKNOWLEDGEMENTS vi

TABLE OF CONTENTS vii

LIST OF TABLES x

LIST OF FIGURES xi

LIST OF APPENDICES xii

LIST OF ABBREVIATIONS AND SYMBOLS xiii

CHAPTER ONE 1

INTRODUCTION 1

1.1 Background of the Problem 1

1.2 Statement of the Problem 2

1.3 Rationale of the Study 3

1.4 Research Objectives 4

1.4.1 General Objective 4

1.4.2 Specific Objectives 4

1.5 Research Questions 4

1.6 Significance of the Study 4

1.7 Delineation of the Study 5

CHAPTER TWO 6

LITERATURE REVIEW 6

CHAPTER THREE 10

MATERIALS AND METHODS 10
3.1 Materials and Methods Framework 10
3.2 Raspberry Pi 11
3.3 Arduino UNO 11
3.4 Dust Sensor 11
3.5 DHT 22 Sensors 12
3.6 MQ-2 Gas sensor 12
3.7 vi. Xbee Module 13
3.8 ZigBee Standard 14
3.8.1 Advantages of ZigBee Technology 15
3.9 Architectural Design 15
3.10 Development and Implementation of the System 22
3.10.1 Sensing Subsystem 22
3.10.2 Data Processing Subsystem 23
3.10.3 Web Application System Development TOOLS 23
3.10.4 Other Requirements 28
3.11 System Implementation 28
3.11.1 Website Development using XAMPP/PHP 28
3.11.2 Developed Web-Based Application 29
3.11.3 Advantages of the Developed System 32
3.12 System Testing, Verification, and Validation 33
3.12.1 Test Requirements 33
3.12.2 Unit Testing 34
3.12.3 Integration Testing 34
3.12.4 System Testing 35
3.12.5 Acceptance Testing 35

CHAPTER FOUR 37
RESULTS AND DISCUSSION 37
4.1 Results 37
4.1.1 System Implementation Overview 37
4.1.2 Sensing Subsystem 38
4.1.3 The Web-Based Monitoring System 39
4.1.4 Real-Time Environment Monitoring 39
4.1.5 Warehouse Live Data Node 1 42
4.2 Discussion 45
CHAPTER FIVE 47
CONCLUSION AND RECOMMENDATIONS 47
5.1 Conclusion 47
5.3 Recommendations 48
REFERENCE 49
APPENDICES 54

LIST OF TABLES

Table 1: Raspberry Pi comparison with another Single Board Computers (SBC's)..................10

Table 2: User authentication unit testing..34

Table 3: System acceptance testing survey questionnaire..36

LIST OF FIGURES

Figure 1: XBee module Antenna Pro S2B (Wang & Zhang, 2006) 14

Figure 2: ZigBee communication layers (Wang & Zhang, 2006) 15

Figure 3: Database schema 17

Figure 4: Use Case Diagram Showing system Administrator and end-user 18

Figure 5: Overall System Architecture 19

Figure 6: Full Block Diagram of the Base Station and Sensor node 20

Figure 7: Block Diagram of the Sensor Node 21

Figure 8: Mobile Terminated (MT) SMS with delivery receipt using SMPP design 22

Figure 9: Local Server running Apache and MySQL Services on XAMMP Control Panel 27

Figure 10: System administrator and system users‟ login page” 30

Figure 11: Environment monitoring dashboard 31

Figure 12: Summaries of the Sensor Nodes 32

Figure 13: System acceptance testing survey questionnaire 38

Figure 14: Developed Sensing Nodes 39

Figure 15: Summary of environment monitoring dashboard nodes available to the system administrator 40

Figure 16: Summary of environment monitoring dashboard nodes available to the technicians 40

Figure 17: Real-time environment monitoring data 41

Figure 18: Sensor data stored in the database and displayed in the webpage 42

Figure 19: Environment live data level at warehouse Area 1 from sensor node 1 43

Figure 20: Environment live data level at warehouse Area 2 from sensor node 2 44

Figure 21: Environment live data level at Assembly line Area 3 from sensor node 3 45

LIST OF APPENDICES

Appendix 1: Questionnaire for System Validation 54

Appendix 2: Arduino UNO Codes for environment monitoring panel..55

Appendix 3: PHP Codes for the Web- based Application 60

Appendix 4: Configuration of XBee antenna Module as ZED and ZCM using X-CTU software 66

Appendix 5: DHT22 sensor connection diagram Sensor and sensor Pinout...............................67

Appendix 6: MQ-2 Gas sensor and connection diagram of the sensor 67

LIST OF ABBREVIATIONS AND SYMBOLS

ACRONYM	DEFINITION
API	Application Programming Interface
CCTV	Closed-Circuit Television
CNC	Computer Numerical Control
CO	Carbon Monoxide
CSS	Cascading Style Sheets
CSV	Comma Separated Values
DBMS	Database Management System
DLP	Digital Literacy Programme
EEPROM	Electrically Erasable Programmable Read-Only Memory)
GPIO	General-Purpose Input-Output
GSM	Global System for Mobile Communications
GUI	Graphical User Interface
HTML	Hypertext Mark-up Language
HTTP	Hyper-Text Transfer Protocol
HVAC	Heating, ventilation, and air conditioning
ICT	Information and Communication Technologies
IDE	Integrated Development Environment
IEEE	Institute of Electrical and Electronics Engineers
IOT	Internet of Things
IT	Information Technology
JSON	JavaScript Object Notation
LCD	Liquid Crystal Display
MCU	Microcontroller
ORDBMS	Object-oriented Relational Database Management System
OS	Operating System
PAN	Personal Area Networks
PHP	Hypertext Pre-Processor
PPM	Particle Per Million
PWM	Pulse Width Modulation
RAM	Random Access Memory
REST	Representational State Transfer

SMPP	Short Message Peer-to-Peer
SMS	Short Message Service
SQL	Structured Query Language
SSA	Sub-Saharan African
SMTP	Simple Mail Transfer Protocol
UART	Universal Asynchronous Receiver-Transmitter
WAN	Wide Area Network
WAPMS	Wireless Sensor Network Air Pollution Monitoring System
WSN	Wireless Sensor Network
ZCM	ZigBee Coordinator Modem
ZED	Zigbee End Devices

CHAPTER ONE

INTRODUCTION

1.1 Background of the Problem

The MU Technologies, Digital Assembly Plant is based at Rivatex East Africa Limited in Eldoret town, Kenya. Moi University, through its Digital Assembly Plant, has undertaken several critical Government initiatives including Digital Literacy Programme, the 2019 Kenya Population and Housing Census (Census), and the Constituency Innovation Hub project which is tasked with assembling Laptops, Tablets, Desktops, Phones, Servers and digital meters such as power and water. The University is looking forward to making Eldoret an ICT and Innovation Hub. The plants assemble thousands of devices in its assembly line and its warehousing has the capability of storing millions of devices under one roof hence the need to monitor the plant indoor environment to ensure the safety of the devices and staff working inside the plant.

Nowadays the technology has evolved to a remarkable extent, the use of the Internet of Things (IoT) and Wireless Sensor Networks (WSN) technology has proven to revolutionize and hastened the industrial 4.0 revolution (Dutta *et al.,* 2018). The IoT is a technology that connects sensors to embedded systems and allows data from these sensors to be transmitted over the Internet (Sumithra *et al.,* 2018).

Typically, the environmental monitoring in industries is carried out manually and occasionally using wired systems that are normally costly, huge in size and not flexible. The wired systems is not the best in terms of security and cost, as it is necessary to hire extra staff for monitoring measurements in dangerous areas of the factory. In this way, if a catastrophic discharge or alert occurs, the factory staff will not notice the danger in real-time, which can be several days later. That is one of the reasons why WSNs can offer a more reliable and real-time monitoring capabilities (Othman *et al.*, 2012).

Usually, wired communications are appreciated widely by research community due to its stability in services. They are not influenced by external environmental effects as compared to wireless networks. Due to this distinctiveness, wired connections linger trendy, yet wireless system sustained to proceed. Industrial environment monitoring with wired network have some limitations such that wired sensors could not be implemented in remote areas. Also it is very complex and costly to mount and sustain the wired networks (Valverde *et al.*, 2012).

Additionally, if a wire between the two devices gets breaks, the communication between these two stops; hence, the entire network will also fail.

Wireless communications is a rapidly increasing technology provide with the flexibility and mobility in the environment. The benefit of wireless transmission is a key diminution and simplification in wiring. The cabling cost in industrial installations is 130–650 US$ per meter and using wireless technology, it would be eradicated around 20–80% (Othman *et al.*, 2012).

This study, therefore, present a technological solution to issues pertaining environment industrial monitoring and industrial automation using IoT and WSN technology hence the safety of devices and people working in the plant and to regulate the factory environment. The system uses various sensors to monitors the carbon monoxide, temperature, humidity, dust, and smoke, and makes notifications to the phone via SMS and E-mail. It also notifies of any danger that might arise within the plant.

The developed system incorporates the following component, Arduino UNO microcontroller, Raspberry Pi, DHT22 sensors, dust sensor, MQ-2 Smoke sensor Zigbee, and Xbee module. These components been integrated together with web based application to develop a web-based monitoring system. With the WSN technology, the data will be sent to the database server hosted within the PC and is accessible via a web page over the internet. A web-based application has a capability for storage and analysis of data collected and allows the system users to visualize the data in form of real-time graphs or download it as an excel sheet or CSV file on their PC or mobile phones.

1.2 Statement of the Problem

Most African industries face challenges in the monitoring and tracking of facilities and their environment. Existing systems are non-reliable, expensive to acquire and maintain, and are standalone hence lack remote features and cannot be dependent on their own. In today's industries, tracking systems that are used for monitoring are mainly are costly that most industries cannot afford. The absence of a user-friendly control system leads to an unsafe working environment.

Several studies, most of which were conducted outside Africa ,have tried to resolve the problem but were all limited (Goel *et al.*, 2012). The proposed solutions are either too expensive or too technologically advanced to be specifically adapted to our local industries, or lack certain features that appeal to the target market. Bad air quality kills about 6.5 million people anually, with 4.3 million of them dying as a result of industrial pollution. Each year, 32 million years of life were lost or lived with disabilities due to industrial air pollution and worker exposure. By 2030, ground-level ozone emission is expected to reduce staple crop yields by up to 26% (Andrady *et al.*, 2017). Hence, there is a paucity of information on improving the existing systems to enhance remote monitoring and self-regulated automated system of the current environmental monitoring systems. As a result, this research aims to create an environment monitoring system that will improve remote monitoring, ambient working environment, protection, and industrial automation using wireless sensor technology.

Also, there has been a challenge on the assembled devices in the field due to defects, from the call center data. It is reported that there have been inconsistencies for the last year, the number of repairs on devices has increased tremendously. This is attributed to the storage condition at the warehouse since most of the defect cases include LCD screen damage, dust in the projector lens and PC motherboard, battery overheating and discharging. Some of these cases are caused by environmental conditions and can be solved by maintaining the factory environment clean and monitoring its indoor environment to ensure sustainability and reduction in the cost of repairs and maintenance.

1.3 Rationale of the Study

However, much industrialization brings a higher economic development, there is a need to monitor the safety of both the people working within the plant and devices stored in the warehouse. Most industries in East Africa have no automated systems .Some are, however, semi-automated. The existing monitoring systems are wired, manual and more expensive. They are also stand-alone systems e.g. Smoke detector and CCTV cameras which do not serve the overall purpose of monitoring all industrial parameters in real-time and cannot provide an alert mechanism. As a result, in the present study, a real-time environment monitoring system is designed using sensors and a wireless network. The existing monitoring systems which includes smoke detectors and CCTV cameras alone cannot fully monitor the plant, hence the need to have an indoor industrial environment monitoring system to monitor the following parameters; carbon monoxide, temperature, humidity, and dust.

1.4 Research Objectives

1.4.1 General Objective

To design and develop an industrial indoor environment monitoring system based on a wireless sensor network in a Digital Assembly Plant (DLP) factory.

1.4.2 Specific Objectives

(i) To review and analyze the requirement for the development of an indoor industrial environment monitoring system using WSN technology.

(ii) To design and develop an indoor industrial monitoring system using wireless sensors technology.

(iii) To test and validate the performance and deploy the developed system.

1.5 Research Questions

(i) How t o choose various microcontrollers, sensors and analyze their specifications and strength to suit industrial plant use?

(ii) How to monitor the industrial indoor environment using WSN?

(iii) Which web application and communication protocol can be used in real-time to keep, retrieve and communicate information in a specific layout to the respective authority and clients?

1.6 Significance of the Study

This research project is expected to make major contributions to the scientific and industrial communities in the following ways:

(i) Nikhade (2015) PLC and industrial automation using Zigbee are often defined in terms of their benefits and significance. The Zigbee system is a wireless sensor network used to regulate and supervise various processes in industries without interfering with other processes. It is used to manage and track industrial applications. The study will enable the DLP management to adopt the cost-effective and reliable monitoring system in the plant.

(ii) Shinde and Bhagat (2017) presented on the use of wireless circuitry and android pulpit like CNC machines to remotely control and supervise industrial applications or processes.

(iii) This research project would also make an important role in contributing to the scientific community through the introduction of a novel methodology for developing industrial remote monitoring systems. The study demonstrates how IoT and WSN can be combined to install an inexpensive wireless monitoring network for industries.

1.7 Delineation of the Study

The developed system incorporates the following component, Arduino UNO microcontroller, Raspberry Pi, DHT22 sensors, dust sensor, MQ-2 Smoke sensor Zigbee, and Xbee module. These components been integrated together with web based application to develop a web-based monitoring system. With the WSN technology, the data will be sent to the database server hosted within the PC and is accessible via a web page over the internet. A web-based application has a capability for storage and analysis of data collected and allows the system users to visualize the data in form of real-time graphs or download it as an excel sheet or CSV file on their PC or mobile phones.

CHAPTER TWO

LITERATURE REVIEW

With the world's industrialization and urbanization processes moving at breakneck speed, remote monitoring, warehousing management, and environmental monitoring is now an upcoming technology in most of the industries in developing countries. Remote monitoring on the other hand encompasses several applications and a wide environment where wired systems can be complemented with wireless ones to reduce the cost of wiring and enable innovative measurement and monitoring applications. The environmental monitoring of water, air, and soil; monitoring of industrial systems, structural monitoring of bridges or buildings; and asset tracking and monitoring are all examples of remote monitoring applications.

Deshmukh (2016) proposed monitoring and control systems using various control technologies. While interaction between a user and a system is largely accomplished online through wireless communication systems like Bluetooth, RF, and ZigBee, SCADA programs are equally employed to develop user interfaces. Nevertheless, SCADA programs are not adaptable for users owing to their costly libraries. Since the distance between the sender and receiver is short and a small amount of data in transmission, RF, Bluetooth, and ZigBee technologies, commonly used in simple-to-use applications, are employed. However, RF, Bluetooth, and ZigBee wireless communication methods are largely restricted to simple applications owing to data protection, and their slow communication rates and distances.

Goel *et al.* (2012) proposed the monitoring of levels of different contaminants in the air due to environmental changes using a wireless sensor network with numerous sensors nodes. This system advances a strategy that concentrates on extending the sensor network while maintaining time by efficiently processing the collected data, managing energy, and reducing the overhead in transmitting data between sensor nodes, but it does not consider the overall efficiency of the proposed system.

He *et al.* (2016) proposed the use of the Zigbee wireless sensing method in an air quality monitoring device in the oil and gas industry. The system sends results to the monitoring center through the ZigBee wireless network, such that if any unusual circumstances arise, a quick alert is provided to the workers to take appropriate steps to protect human lives and avoid major accidents. According to the National Fire Protection Association's (NFPA) most recent fire statistics, about 37 000 fires happen every year in industrial and manufacturing industries

(Dunlap, 2020). To control this issue, a remote monitoring system with an alarm is needed. These incidents cause 279 and 18 civilian deaths and injuries respectively and about $1 billion in direct property destruction.

The Wireless-Sensor Network Air-Pollution Monitoring-System (WAPMS) is a revolutionary system that uses wireless sensors to track air pollution on Mauritius Island (Khedo *et al.,* 2010). They developed and executed a novel data-aggregation algorithm referred to as Recursive Converging Quartiles to increase the performance of WAPMS (RCQ). This algorithm is employed to combine data to remove duplicates and wrong readings and summarize them in a simplified form thus conserving energy by reducing the quantity of data that must be conveyed to the sink. For more efficient power management, a hierarchical routing protocol is used in WAPMS to cause the notes to sleep in idle time.

Due to the rapid development of communication, remote-sensing, and network technologies, the trend is to design environment monitoring systems in a wireless mode (Zhang *et al.,* 2018). Wireless Sensor Networks have advanced rapidly in recent years. Its benefits that encompass liability, cost-effectiveness, and simplicity range from industrial to military controls. The WSNs are self-configuring networks made up of several low-power and low-cost multi-functional wireless sensor nodes that can detect and respond to a variety of physical and environmental conditions, including temperature, pressure, humidity, sound, and speed. The sensor nodes can communicate wirelessly over short distances and send their data to their intended destinations through the network. The location of wireless sensor nodes in a WSN does not need to be predetermined since they can enter or exit the network at any time (Cao *et al.,* 2017).

The WSN serves as a connection between the virtual and real worlds. It can observe locations where scales. These properties give the WSN several applications, like industrial monitoring, industrial automation, security systems, health care, smart irrigation, etc. Established at the University of Illinois at Urbana-Champaign, the ISHMP system is a well-known structural monitoring system (Ullo & Sinha, 2020). This framework is built on the open-source WSN platform. It uses two common WSN data-acquisition systems; TinyDB and SwissQM, which support large-scale monitoring but lack integration analytics. A different system is given by Sumithra *et al.* (2018), designed for environmental monitoring with no provision for analog interface sensors. The described framework, however, includes sensors support with an analog interface.

Morphology is captured using an indoor wireless monitoring system for fast creation and management flexibility (Zhou & Lee, 2019). The system concentrates on building management using event logic but does not take care of data synchronization. Snow Fort is a free and open-source WSN framework for monitoring the climate and infrastructure. This framework was created for the detection of real-time damage and health monitoring application. The framework developed and presented in this paper, on the other hand, presents application-specific sensing focused on environmental monitoring. The various environmental problems like humidity, harmful gases, and temperature are monitored and transmitted to the end-user (Systems, 2020), is an indoor HVAC system with DSS at the back-end that is equipped for heating, air-conditioning, and ventilation. The Wi-Fi logs, not sensing components, are used in this system. Nevertheless, the system designed in the present study is a combination of sensor nodes that gathers data by sensing.

Environmental monitoring using an outdoor wireless monitoring system is considered (Yang & Yang, 2009). The system is dependable and employs MAC to synchronize data. This device has been tested and configured to last a long time from solar panel use. We have placed environment sensors as part of our system. Another system based which perceives damage of structure by WSN using time-domain algorithm is given (Kritica, 1980). The system uses hypothesis tests to make decisions on whether or not losses occur. It also helps users to remove features from strain gauges and accelerometers that are susceptible to injury. The damage detection system is founded on a frequency domain distributive algorithm and authenticated by ISHMP (Ghiasi & Ghasemi, 2020). This outdoor wireless monitoring system is intended for predicting and monitoring earthquakes (Hackmann *et al.*, 2014). This system uses both WSN and IOT for monitoring.

Gupta *et al.* (2010) proposed a system, where small wireless sensors connected in several applications such as internet-capable devices-smart phones, cars, cameras, toys, home appliances, medical instruments, and energy meters which generate a significant amount of small data bits. The key role of this information is in its interpretation, which leads to important observations and activities that can improve the health of people and our planet. The author has developed a web-based infrastructure monitoring system made of a Sensor Network for searching, analyzing, storing, visualizing, and sharing data from heterogeneous devices. This system also facilitates integration among end-users and devices and using an open REST-based Application Programming Interface (API).

In a separate study, Ashna and George (2013) advanced a GSM-based automated environment communication system with a web portal for data management only. Following that, the data is transmitted to a third party for processing. Nevertheless, the system did not have user-friendly real-time alerts and self-healing in case one node fails, thus could not provide the end-users complete access to system information. As already described, many of these systems do not provide cost-effective real-time systems. Besides, relying on public cellular networks like Bluetooth or GSM to monitor systems heightens operating costs especially in big industries with several sensor nodes because each node must be fixed with a GSM card with adequate data packages hence signal loss.

The present study, therefore, addressed the aforementioned limitations by designing and developing a monitoring system using a wireless sensor network that can enable industrial automation to ensure ambient working and storage environment in the warehouse. Likened to other studies, the advanced system has placed more emphasis on a cost-effective design, stakeholder-driven system needs, and system sustainability by utilizing the existing technologies and infrastructure for development.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Materials and Methods Framework

The methodology and materials used, as well as how the framework was implemented are detailed below:

(i) Hardware used include; Arduino Uno Microcontroller, Raspberry Pi, AM2302 (Wired DHT22) temperature-humidity sensor, Optical Dust Sensor GP2Y1010AU0F, MQ-2 Gas Sensor, XBee antenna, and XBee Module.

(ii) Software used include; Embedded C, XCTU XBee configuration software, Source code editors e.g. Arduino IDE and Sublime text IDE, MYSQL database, and Apache 2.4.

Table 1: Raspberry Pi comparison with another Single Board Computers (SBC's)

Parameter	Raspberry Pi	Arduino Uno	Intel Galileo	UDOO NEO	Beagle bone
Processor	Broadcom BCM2837 64bit ARMv7 Quad Core 1.3Ghz	ATM EGA 8, ATM EGA 1280	Intel Quark X1000-400MH z single core	Free scale I MX 6soloX 1ghZ ARM Cortex A9 with Cortex M4	AM335 9 1GHz Cortex A8
RAM	IGB	16-32KB	512Kb on chip SRAM 256Mb DRAM	512MB or 1GB (full)	512MB DDR3 RAM
POWER	10W	5W	15W	10W	15W
OS	Raspbian, Debian, Fedora, ARCH Linux ARM and FreeBSD	N.A	Arduino Linux distribution for galileo Windriver Rocket	Android Lollipop and Linux UDOObuntu2 (14.04 LTS)	Android, Debian, Angstrom, yacto, Fedora, Ubuntu
COST	$40	$30	$70	$65	$55

No and Ali (2017

The comparison of raspberry pi model with Single Board Computers (SBC's) as shown in Table 1 is noted that Arduino Uno microcontroller and raspberry pi are considered being the most effective in terms of cost and powering consumption. In conclusion of the above table, the raspberry pi used is the latest raspberry pi i.e. raspberry pi model 3 B+ which is cost-efficient SBC's from others and also compatible with complex task management

3.2 Raspberry Pi

In this system, the Raspberry Pi serves as a Base Station. This is a credit-card-sized device that runs the raspbian Linux operating system and provides simpler and more affordable wireless monitoring solutions (Vujovic & Maksimovic, 2014). It runs on an ARM11 microcontroller with a 700 MHz clock frequency. In this implementation Model 4 B, which employs system on chip (SoC) is used (Raspberrypi, 2019). The board is a perfect choice for implementing the system because of low power consumption and cheap price compared to other minicomputers. It also provides ETHERNET to the user which is beneficial because it allows users to control remote access by SSH login using the IP address. For storage, Raspberry Pi has 512 MB of RAM which is provided by an external SD card (Raspberrypi, 2019).

3.3 Arduino UNO

This is a microcontroller board based on ATmega328 with a USB connection, an ICSP header, a 16 MHz ceramic resonator, 14 digital input/output pins, a power jack a reset button, and 6 analog inputs. The board has a flash memory of 32 KB (0.5 KB of which is used by the boot-loader), 1 KB of EEPROM, 16 MHz clock speed, and 2 KB of SRAM (Farnell, 2013). It offers a set of digital and analog pins that can be combined with several other circuits and boards with completely different roles in the design. The board offers a USB serial-communication interface for loading the codes from the computer. Arduino has its software known as integrated development environment (IDE) which completely supports C and C++ programming languages to program the board (Deshmukh, 2016).

3.4 Dust Sensor

Sharp's GP2Y1010AU0F is an optical air quality sensor, designed to sense dust particles. An infrared emitting diode and a phototransistor are diagonally arranged into this device, to allow it to detect the reflected light of dust in the air. The sensor has a very low current consumption of about (20 mA max, 11 mA typical), and can be powered with up to 7VDC. The output of the sensor is an analog voltage proportional to the measured dust density, with a sensitivity of 0.5 V/0.1 mg/m^3. The sensor quantifies the level of Particulate Matter (PM) in the air by enumerating the Lo Pulse Occupancy (LPO) time in a given time unit. The LPO time is proportional to the concentration of PM (Goel *et al.,* 2012). The sensor can deliver very accurate data for air-purifying systems because it is highly sensitive to particulates of 1µm diameter. It can also be interfaced with an Arduino Uno microcontroller via a wired pitch connector with 6-pin, 1.5 mm in size. The sensor is also relatively cheap (it costs roughly 12$) which makes it suitable for low-cost applications. The measurement unit for dust density is micrograms per cubic meter ($\mu g/m^3$) (Qiu *et al.*, 2021).

3.5 DHT 22 Sensors

The DHT22 is a commonly used temperature and humidity sensor. The sensor comes with a dedicated NTC to measure temperature, a resistive-type humidity measurement component and an 8-bit microcontroller to output the values of temperature and humidity as serial data. The sensor is also factory calibrated hence easy to interface with other microcontrollers which offer fast response, excellent quality, cheap, and anti-interference ability. The sensor can measure temperature from -40 °C to 80 °C and humidity from 0% to 100% with an accuracy of ±1°C and ±1%. By using the exclusive digital-signal-acquisition method and temperature and humidity sensing technology, it is highly reliable and stable (Adhav & Vibhute, 2020).

Data is read to the microcontroller from the Signal (S) data pin of the sensor. The DHT22 must be operating in asynchronous serial (Single-bus digital signal output, bidirectional serial data) communication protocol which enables the S pin of the module to be connected to any digital pin of the microcontroller. The required pull-up resistor enables the sensor to operate with both 3.3 V and 5V sources. It connects a 5k resistor between the 1st and 2nd pins of the sensor.

The 1st pin is connected to a 3.3V while the 4th one is connected to GND. The 2nd pin is the output pin that gives input to the Arduino Uno. Pin 3 is left empty. The connection diagram of the DHT22 sensor and pinout diagram is shown in Appendix 5.

3.6 MQ-2 Gas sensor

The MQ-2 gas sensor is an electronic sensor used for detecting the concentration of gases in air as such as LPG, smoke, carbon monoxide, propane, and methane. It can detect CO-gas concentrations anywhere from 20 to 2000 ppm. The sensitivity of the sensor can be adjusted by a potentiometer. The sensor's output is an analog resistance. The circuit consists of a heater coil with 5V, load resistance, and an output connected to an ADC. It contains a sensing material whose resistance changes when it comes in contact with the gas. This change in the value of resistance is used for the detection of gas. This sensor contains a sensing element, mainly aluminum-oxide-based ceramic, coated with Tin dioxide, enclosed in a stainless steel mesh. The sensing element has six connecting legs attached to it.

Two leads are responsible for heating the sensing element while the other four are used for output signals. Some of the features that make this sensor suitable for this application are wide detecting scope, Fast response and High sensitivity, stable and long life, low cost, and simple drive circuit. Appendix 6 shows the MQ-2 gas sensor and connection diagram of the sensor with the arduino uno.

3.7 Xbee Module

This is an embedded solution that offers wireless end-point connectivity to devices. The module uses the IEEE 802.15. 4 networking protocol for rapid peer-to-peer or point-to-multipoint networking. This module is ZigBee based protocol and its interaction with the PC and base station occurs through UART. It offers both API and AT and mode serial interfaces for communication. When operating in AT mode, it functions as a serial replacement but in the API mode, incoming and exiting data is separated into frames that describe operation (Alliance, 2007). The ZigBee model is shown in Fig. 1

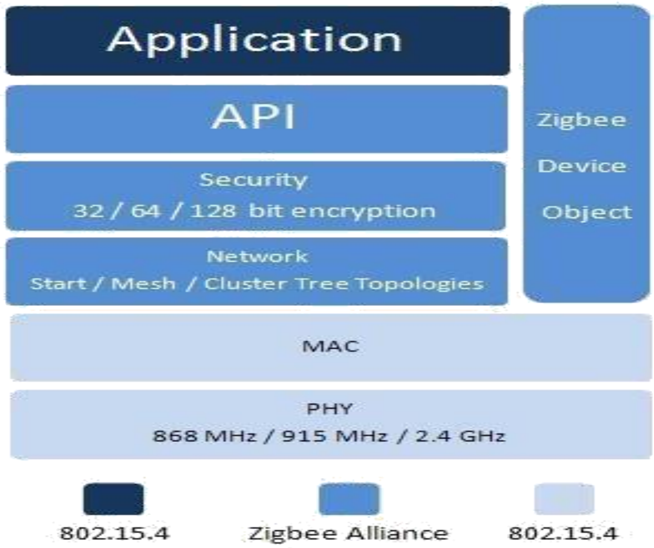


Figure 1: XBee module Antenna Pro S2B (Wang & Zhang, 2006)

3.8 ZigBee Standard

This is a low power, low data rate, and short-range wireless networking method for several real-time applications. It specifies the three bottom layers: Network, Physical, and Data Link, and an API-based on the 7-layer Open System Interconnection (OSI) model for layered communication systems (Alliance, 2007). The layered architecture implemented by the alliance industries is shown in Fig. 2. It should, however, be known that Zigbee Alliance uses a pre-existing data link and physical layer specifications which are IEEE 802.15.4 standards for low-rate personal area networks (PAN).

The IEEE 802.15.4 standard supports three frequency bands of operation, namely 868 MHz, 916 MHz, and 2.4 GHz in Europe, the United States, and worldwide respectively. 2 .4 GHz bands are common worldwide due to Industrial, Scientific, and Medical (ISM) bands. Besides, this band provides 16 communication channels between 2.4 GHz and 2.4835 GHz at the physical layer and the highest achievable data rate of 250 Kbps (Jung *et al.*, 2008).

Figure 2: ZigBee communication layers (Wang & Zhang, 2006)

Common communication distances occur in the range of 30 meters in an indoor environment and 10 to 100 meters in a line-of-sight environment dependent on module specifications. The modules use a dipole-type antenna to raise the gain of the antenna (Adams, 2005).

3.8.1 Advantages of ZigBee Technology

The ZigBee technology is a recently-established low-complexity, short-range, low-power, low data rate, low-cost, and two-way wireless communication technology (Buratti *et al.*, 2009). A 32kB protocol stack that can be implanted in several devices and supports geo-targeting is accessible (Alliance, 2007). The technology is appropriate for wireless sensor networks and is commonly used for short-range wireless communications.

3.9 Architectural Design

3.9.1 Database Design

This is the process of organizing information according to a database model and the method of distinguishing individuals, their relationships, and all of their attributes. The database designerdecides on the information that should be processed and how to relate it to other information. This is to say that database design is the process of classifying and determining the interrelationships of data.

3.9.2 Database Schema

Database Schema is the database management system's conceptual logical view of the entire database (Solihin *et al.*, 2017). It offers a database architecture framework as well as a graphical view and specifies data structures and relationships/connections. Entities, attributes, relationships between entities, and tables make up the conceptual diagram in a database schema. The database schema of the developed system is shown in Fig. 3.

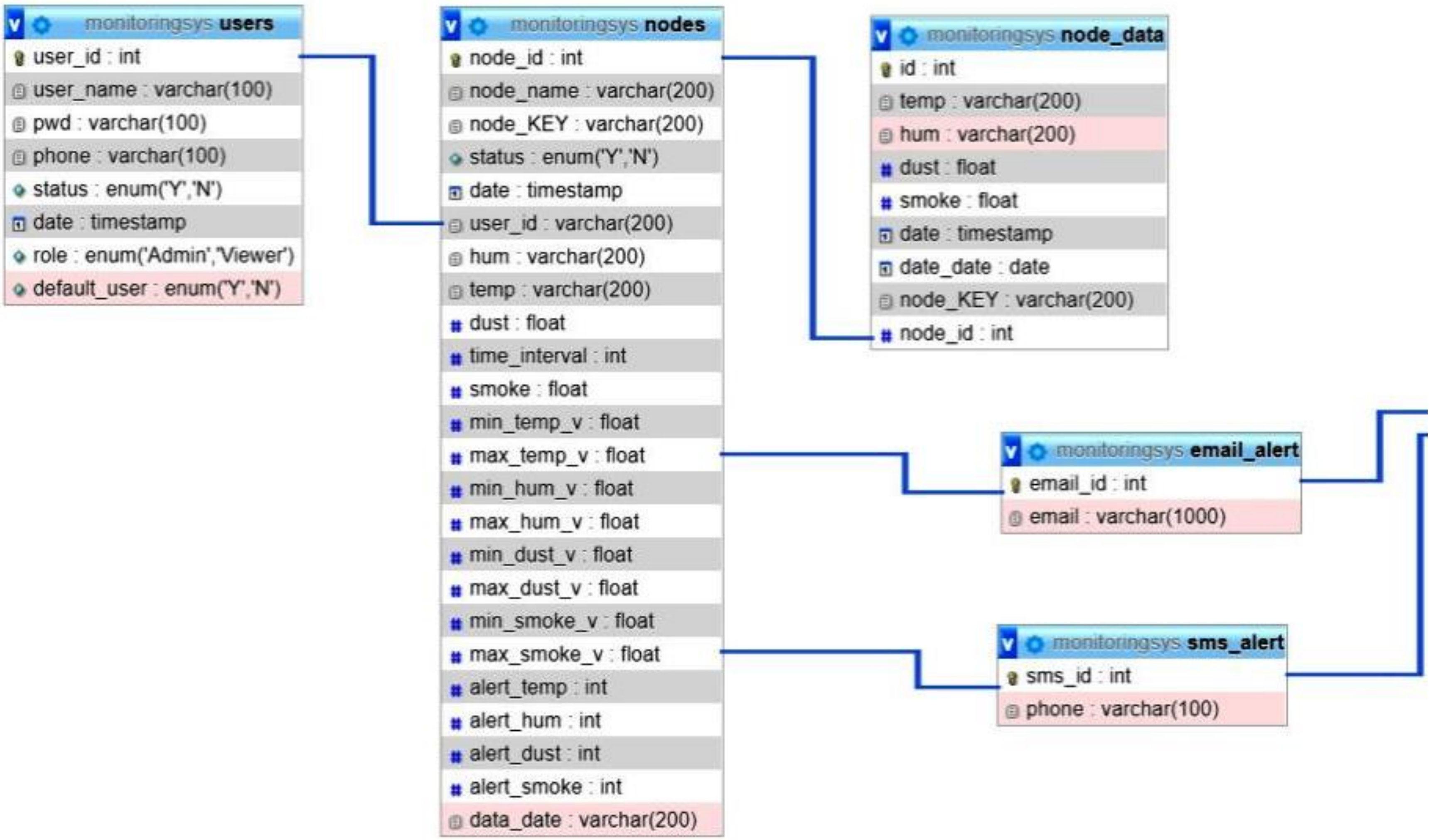


Figure 3: Database schema

3.9.3 Use Case Diagram

This depicts the relationship between the system and its users. It defines the types of activities or procedures that a user may accomplish in the system. Actions are referred to as use cases as is shown in (Fig. 4) below. Cases are referred to as actors which are the users of the system for example; the system administrator, factory manager, technicians, and warehouse supervisors.

The use cases include; login, logout, add new user, delete user, change the name, change the user password, add a node, edit node data, change node setting, monitor node data, disable/enable node, and print/export data.

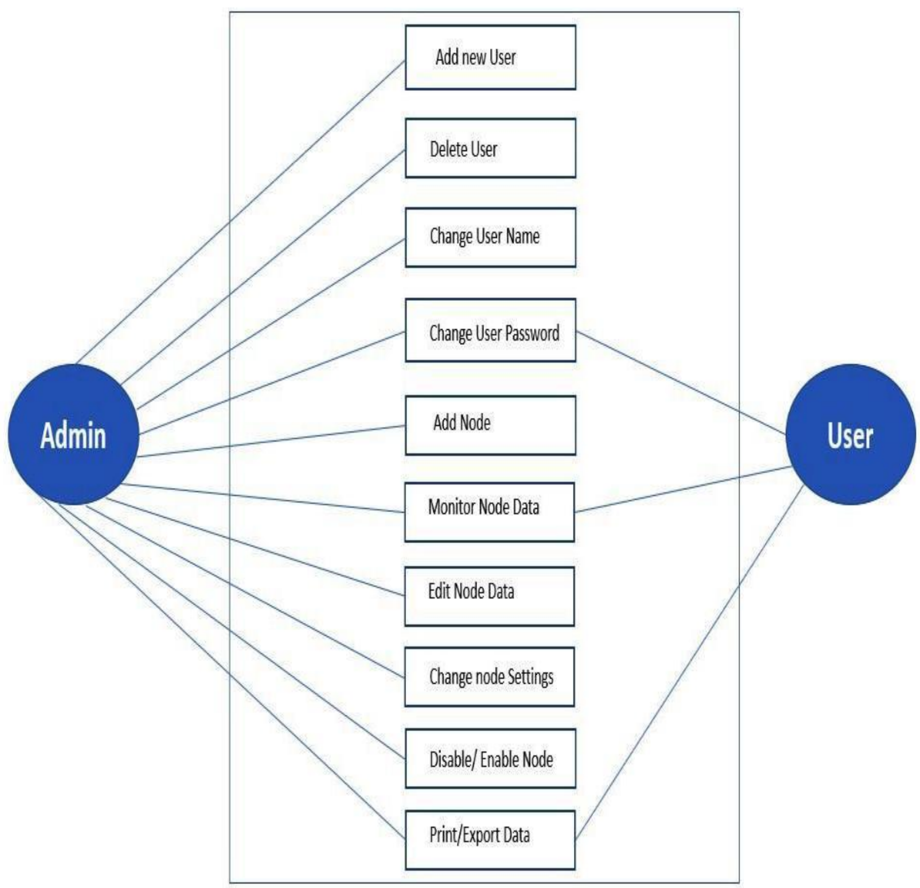


Figure 4: Use Case Diagram Showing system Administrator and end-user

System Design

The project's system architecture is comprised of subsystems that interact together to provide data instantaneously. The subsystems include a sensing system (sensor nodes), information processing system and web-based application for visualization as shown in Fig 5.

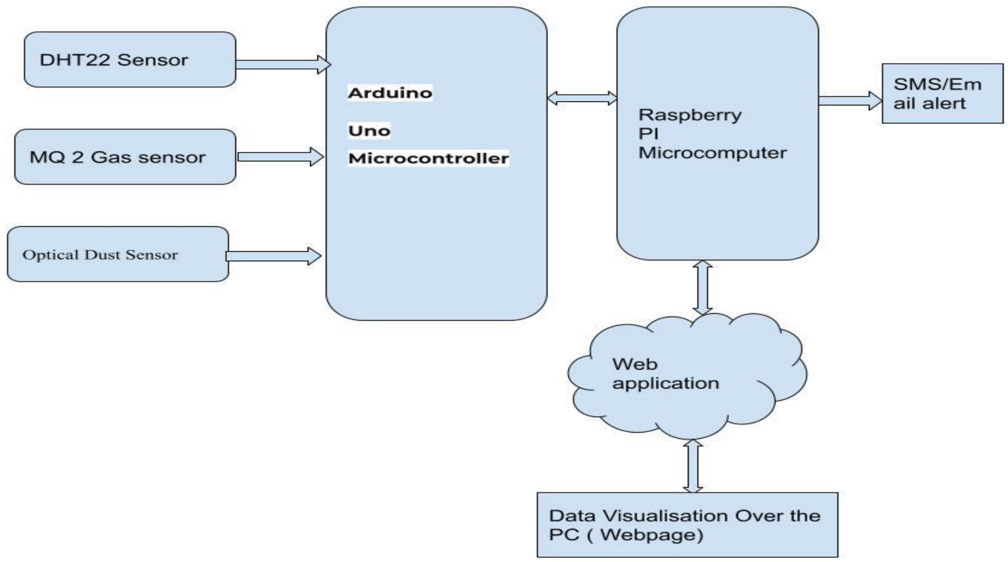


Figure 5: Overall System Architecture

3.9.4 Sink node design

The role of Raspberry Pi is to link the attached XBee coordinator to the sensor nodes and use the XBee module to transmit the message gathered by the sensors to the user.

The Digi international XBee module S2 is adapted to interact with the receiver module. It is powered by the USB cable and the UART serial communication interfaces (Adams, 2005). The module is organized into three modes: router, end-device, and coordinator nodes. Data gathered by the sensor nodes is managed and conveyed via Arduino Uno serially through Tx and Rx pins to the sink node station and stored in the MySQL database. The working principle of Raspberry Pi is presented in Fig. 6.

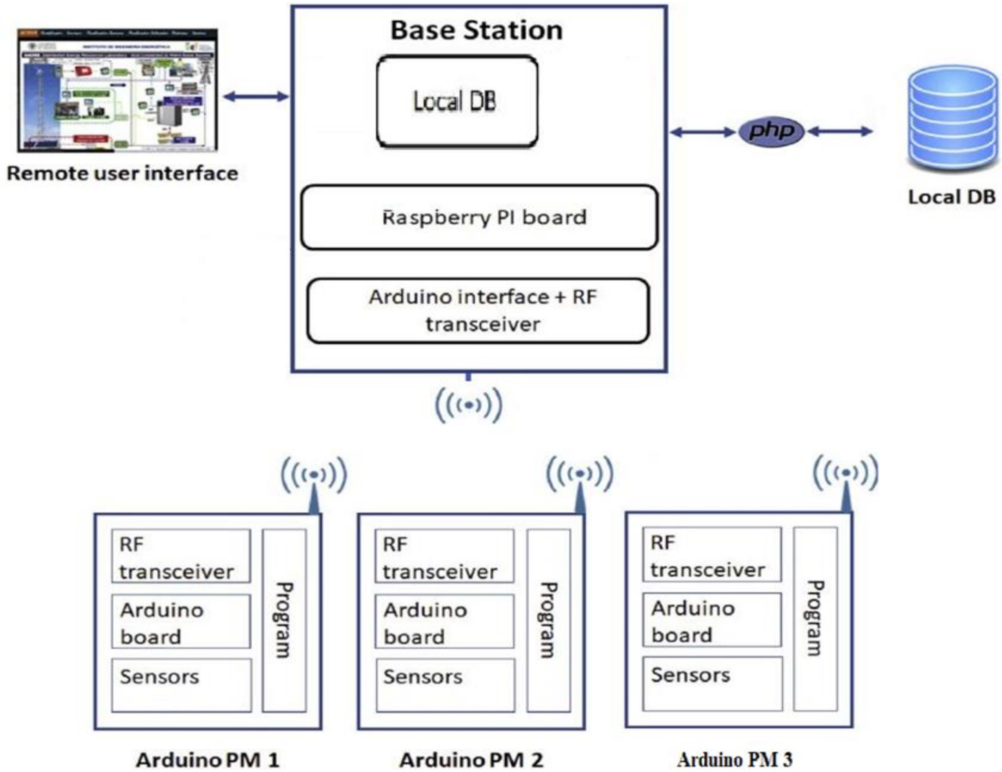


Figure 6: Full Block Diagram of the Base Station and Sensor node

3.9.5 Design of sensor node

Design of the sensor nodes connected with Arduino Uno coupled together with Digi XBee module forms a sensing subsystem as presented. The block diagram of the sensing components of the system connected is in Fig. 7.

The Arduino Uno is connected to the sensor. The digital output of the sensor nodes, on the other hand, is transmitted straight to the Base Station over the Wi-Fi. The ATMega328 will be used to power sensors linked to an Arduino Uno. The Raspberry Pi's ARM11 minicomputer manages the entire device network. The MQ 2 sensor will track the proportion of the emitted CO in the smoke. The output would be a collection of numbers indicating the percentage of pollutants measured in parts per million (PPM) (Fonollosa *et al.*, 2018). The DHT 22 sensor is employed to monitor the humidity and temperature as percentage and degree Celsius respectively. The data from the sensor nodes is coordinated by the base station, which when ready, transmits the data via the XBee S2 module.

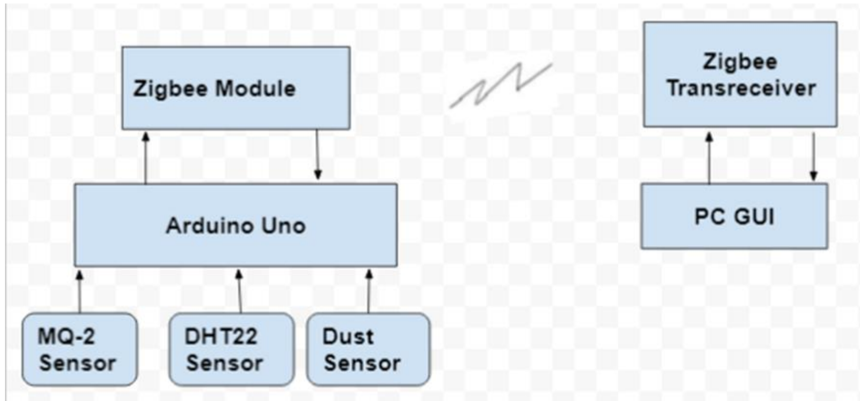


Figure 7: Block Diagram of the Sensor Node

Data transmission and reception between the two devices is via serial communication. On the receiving side, the data is picked by the XBee transceiver which acts as a router and is interfaced by UART serial communication with the PC. The only fixed coordinator is used but the number of end-node devices and routers may be unlimited. Data from the sensor nodes is presented on the PC or smartphone over a browser. The sensor data are transmitted to the MYSQL database in the PC and displayed over on a website installed on a local Apache webserver. The data can be represented in graphs for advanced monitoring of environmental parameters. The user can also access the web interface via a computer. Multiple approved users can also log in remotely to the device using the gateway program and data management.

3.9.6 SMS / Email alert module design

The SMPP (Short Message Peer-to-Peer) protocol is an open, industry standard protocol designed to provide a flexible data communications interface for the transfer of short message data between External Short Message Entities (ESME), Routing Entities (RE) and Message Centres (MC). It is a means by which applications can send SMS messages to mobile devices and receive SMS from mobile devices. This is done using an SMPP connection with a Short Message Service Center (SMSC), It can also be used as an API for use with USSD, CBC and other mobile services as shown in the fig. 8 below. The SMPP Protocol is an API which enable SMS messaging between applications and mobiles.

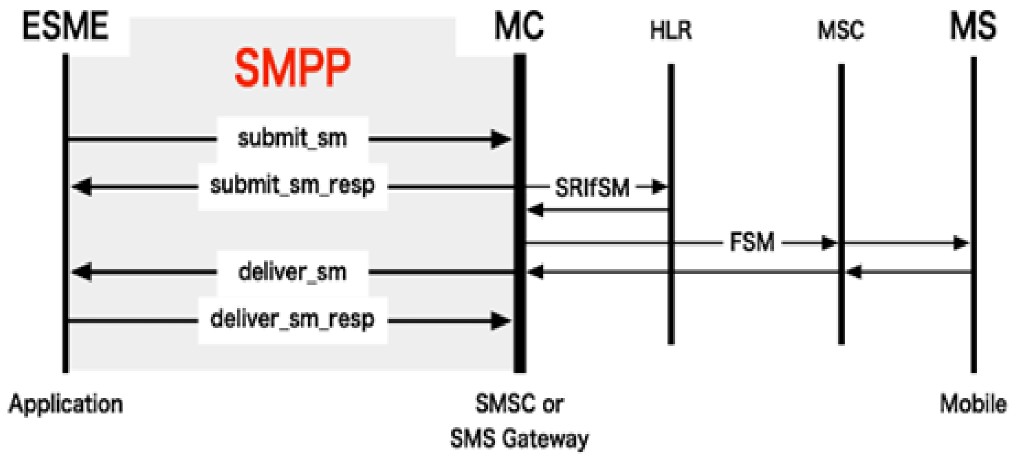


Figure 8: Mobile Terminated (MT) SMS with delivery receipt using SMPP design

The developed system was implemented on a raspberry pi module running on a Linux environment, which supports SMTP (simple mail transfer protocol). The SMTP is an application layer protocol that allows an email to be sent to a registered address from SMTP server. Email is sent via a SMTP server, where an application will log in into an SMTP server as e-mail user and send e-mail to a given registered user address, this can be done via a special API, google mail (Gmail) and PHP Mailer library are used in the implementation of the system.

3.9.7 Wireless Communication Module

To create a small wireless network for an environmental monitoring system, two types of XBee modules were used. One module serves as a transmitter, while the other serves as a receiver. The XBee modules were configured as coordinator and end devices using the X-CTU software program as presented in appendix 4. The network is controlled by a Raspberry Pi-connected minicomputer known as the ZigBee coordinator modem (ZCM). The ZCM gathers data and handles other devices of the network and all ZigBee end devices (ZED) will send data directly to the ZCM.

3.10 Development and Implementation of the System

3.10.1 Sensing Subsystem

This subsystem is comprised of an Arduino UNO microcontroller and sensors that sense environment monitoring parameters. Monitoring values parameters include humidity, temperature, dust, and smoke. In this case, each end node will contain a smoke sensor (MQ-2), a dust sensor, and a humidity-temperature sensor (DHT22) to collect the monitoring parameters.

The sensors input the gathered data is sent to the Arduino board for transmission to the Data processing subsystem via a wireless network (Rotich *et al*., 2020).

3.10.2 Data Processing Subsystem

It obtains data from the sensor nodes marked by unique IDs. This subsystem's controller is a Raspberry Pi. The main role of this device is to process all sample data from the sensing layer in real-time. The processing subsystem's key functions are as follows:

(i) Analyzing the received sensor data.

(ii) Send data to the web server for communication and visualization with web application.

(iii) Initiate local alarm notification if any parameter exceeds the maximum value.

(iv) Sent an SMS and e-mail alerts to the mobile phone of the registered users.

3.10.3 Web Application System Development TOOLS

Client-side languages are used to write source code that is executed by browsers. The execution of source code occurs on the end-users‟ machines. Even though the source code is effected inside the browser, it is transmitted from the web server to the computers of end-users over the internet.

(i) Hypertext Markup Language (HTML)

This is a markup language used to format and view data. It is the most widely used language for website creation. Header tags, paragraph tags, image tags, anchor tags, and so on are all examples of HTML tags. It was used in executing the application`s front end. The HTML provides a basic site layout that is enhanced and developed by further technologies such as JavaScript (JS) and Cascading Style Sheet **(**CSS) and is primarily employed for interface design. The HTML was chosen because of its simplicity, ease of integration with other languages, search-engine friendliness, and compatibility with all browsers.

(ii) Cascading Style Sheet

This is a programming language that specifies the appearance of the elements of a website’s HTML on a user interface page. While the HTML provides the raw resources required to create a website's content structure, the CSS aids in the styling of this material, making it more attractive and well-formatted for the user. This was used in the general design of backgrounds and styling of HTML interfaces to appeal to clients because it is one of the properties of a successful web-based framework. It involved styling the interface design's background pictures, headings, colors, lines, footers, and headers among other things. The “connection” was used to connect a CSS file to an HTML document in the “head” tag of the page.

“Bootstrap.min.css” is a JavaScript library that was used in the design for responsiveness of the webpage to confirm that the portal availability in all devices irrespective of the size and to improve the appearance of the web page. Font-awesome CSS was also employed to enhance the icons, which was accomplished using an open-source Code Distribution Network (CDN) connection. This library makes it simple to keep the user updated using appropriate icons.

(iii) Bootstrap

Modal bootstrap elements were employed to capture data without hiding the web-page text using popup windows that appears whenever a connection or a button is clicked by a user, ensuring that the web pages are consistent. Bootstrap is an open-source and freely available platform for creating web applications. It consists of CSS and HTML form templates, JavaScript extensions, navigation, buttons, and other components of interface design.

(iv) JavaScript

This was employed to change the website's content and style it to react to a user's actions in various ways which include dismissible notifications, confirmation boxes, applying fresh identities to prevailing information, notification alerts, and monitoring the actions of the entire web page elements are all common JavaScript uses. This programming language was often used to validate the data inputs of the system forms' to see whether they would be rejected. For instance, if a wrong username or password is keyed in by a user, JavaScript can warn users. This helps with validation and authentication, performance management, and system status.

Angular JavaScript library (angular.min.js) was employed to confirm the form’s interfaces when the user is typing. It uses a regular expression to synchronize the value of the form input field, highlights the error in red if the input is incorrect, and finally deactivates the “submit button” until the right input is in place. This guarantees that the submitted data by the form is correct, omitting the need for back-end validation and minimizing the amount of time required to process the form.

Asynchronous JavaScript and XML (AJAX) were employed to dynamically load web page contents like chat rooms and update them in the background. This was utilized in the live-table edit of data in tables that view data, where the information in the table field (td>) converts to editable on the "blur" event and is automatically saved to the database when the cursor is removed.

Canvas JavaScript (canvasjs.min.js) was employed to visualize data. To view content dynamically from the database, a JavaScript library was used for plotting graphs along with Hypertext Preprocessor (PHP) and HTML. The data is first obtained from the database, then the array is generated by the PHP, and finally, Canvas JavaScript translates the PHP data array into JavaScript Object Notation (JSON) for simple plotting. To view the details, the produced graphical image can easily be hovered on with the provision of "Print", "Save as JPEG", and "Save as PNG".

(v) Source Code Editors

These are programs that are specifically made to write and edit and computer program source codes. They can be stand-alone programs or frameworks, like Notepad++, or incorporated into an IDE or web browser. They are very vital programming tools because they allows the creation of an executable source code file for input to a compiler/linker. For example, when configuring and programming the Arduino UNO microcontroller and sensors, the program written in C language by an Arduino IDE as shown in Appendix 2.

(vi) Server-side Languages

These are languages to create interactive websites that deliver dynamic web content to users from databases and the other way around. Ruby on Rails, Nodes.js, Django, PHP, ASP.NET, Java, and R are some of the existing server-side scripting languages (Crawford & Hussain, 2017).

PHP was used to create the proposed framework in this research. The dynamic web system's content is created using PHP. Due to its broad developer community and support, PHP is the most widely employed server-side scripting language for web-based systems implementation. The PHP sample code used for developing this web application is shown in Appendix 3.

(vii) Hypertext Preprocessor

This is a widely used, open source, and resourceful server scripting language, is a common tool for creating collaborative and dynamic webpages. The PHP 7.4 was used to create the system's backend, which involved the use of database queries, functions, and data structures like arrays.

When a user registers, for example, the form method "POST" is used to collect and process data for storage with the database functions. This entails using SQL scripts to store and retrieve data from the database. The local server "Apache" must be running along with "MySQL," which unlocks database connections, for the system to operate via port “3306”. This is first accomplished by launching “Xampp Control Panel” and making sure that all the services are running as shown in Fig. 9.

For database queries, the PHP Data Object (PDO) was employed since it ensures continuous connections and generates instances of the database link, allowing the server to execute queries quickly. This improves the efficiency of web pages by allowing them to respond quickly to user requests. The file “config.inc” installed in “C:\xampp\htdocs\monitoringsys\root\” is contained on top of each page with a simple script; <?php require once („root/config.inc') ;?> to guarantee that all pages are linked to the database.

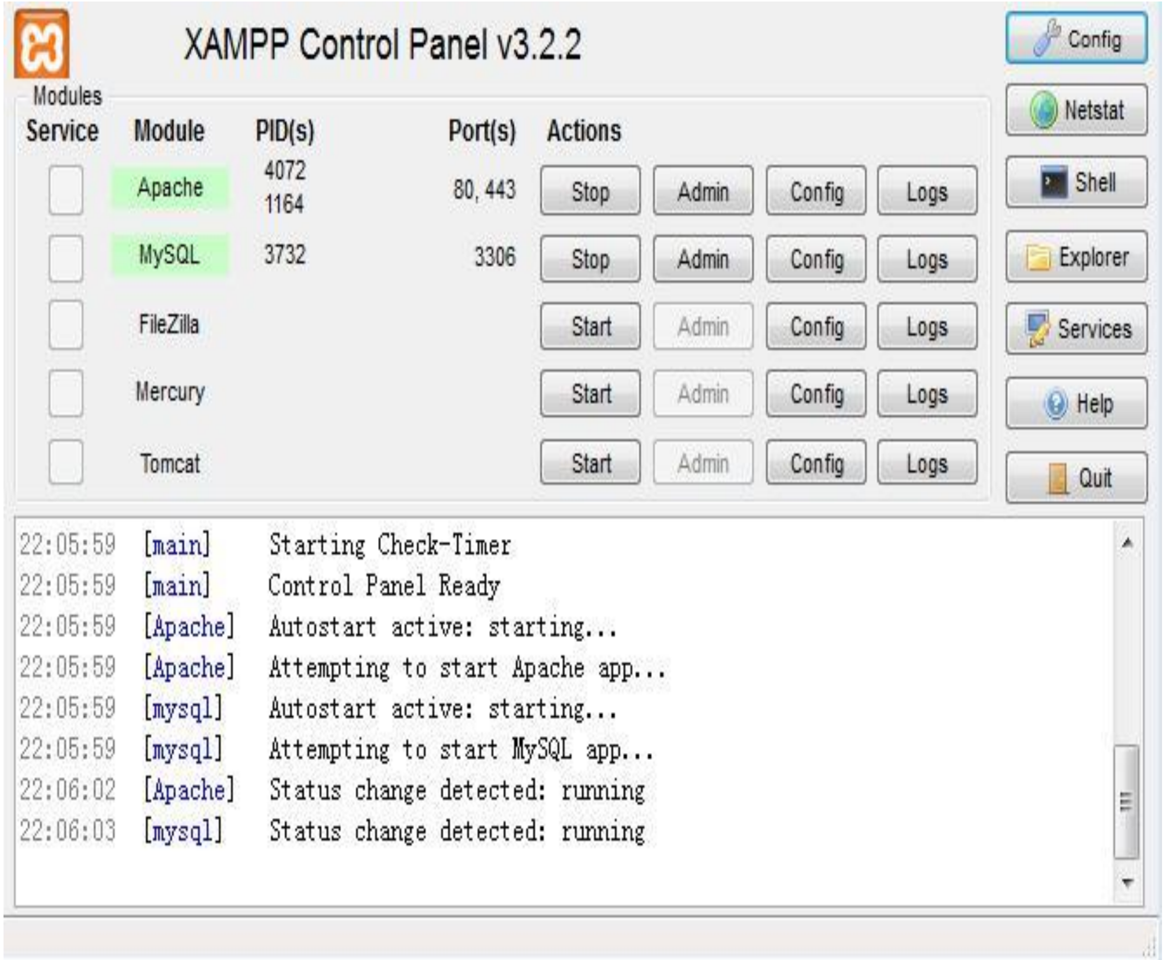


Figure 9: Local Server running Apache and MySQL Services on XAMMP Control Panel

The PHP session functions were employed to manage the views of users and the type of data they could access. Although the login interface is unchanged, the system checks the user type and re-addresses to the appropriate web page based on the session variables set when the user logs in. When a user signs out, the session expires immediately. The upload of the dataset Comma Separated Values (CSV) file was made simpler with PHP arrays and the “array map” feature. This role in the file "update.php" reads the file "visual_data.csv" from the framework folder "C:\xampp\htdocs\mornitoringsys\," extracts data from the excel sheet by reiterating over all rows, and incorporating it into the database for viewing.

(viii) MySQL Database

This is an open-source relational database management system that defines and manipulates data using a standard structured query language (Letkowski, 2015). The MySQL was chosen for many reasons: it is free and runs on a variety of platforms, including Linux and Windows; it has extensive technical support; easy to customize; it is simple to manage; and it supports large databases (Letkowski, 2015).

3.10.4 Other Requirements

(i) Internet connection is needed for the web-based system to function.

(ii) Browsers including; Mozilla Firefox, Google Chrome, Internet Explorer, and Torch.

(iii) Operating system: The web-based system is compatible with Mac OS, Windows 7 and later versions, and unrestricted open-source operating systems like Ubuntu.

3.11 System Implementation

3.11.1 Website Development using XAMPP/PHP

The XAMPP is a cross-platform web server solution stack package that includes the Apache HTTP server, MySQL database, and interpreters for scripts written in PHP. Steps involved in the website creation process using HTML-CSS include:

(i) Analysis

(ii) Design & Development

(iii) Content writing

(iv) Coding

(v) Testing and security

(vi) Maintenance and update

Apache is a web server application that communicates with web browsers and connects them to the information they're looking for. The MySQL is a relational database management system (RDBMS) that makes it easy to store and retrieve structured data. The PHP is a scripting language that works with Apache to create dynamic content, which is often based on user input or from information that is stored in a database.

3.11.2 Developed Web-Based Application

The web-based application system for the industrial indoor environment monitoring system is designed and developed to offer only the essential functionalities suggested by DLP management. Environment tracking, node management data collection, data visualization, user registration, and authentication are among the web application's key features. The key functionalities and interfaces of the established web application are described in the next subsections.

(i) User Authentication

To authenticate the security of the existing system, only registered users can log in to the system to retrieve any information. Operators will be given a default password and username. The passwords were hashed using the BCRYPT algorithm (a PHP built-in one-way hashing algorithm) to ensure access level security. Verification ensures that only authorized users can access a system thus preventing unauthorized users from gaining access to steal information or cause other problems.

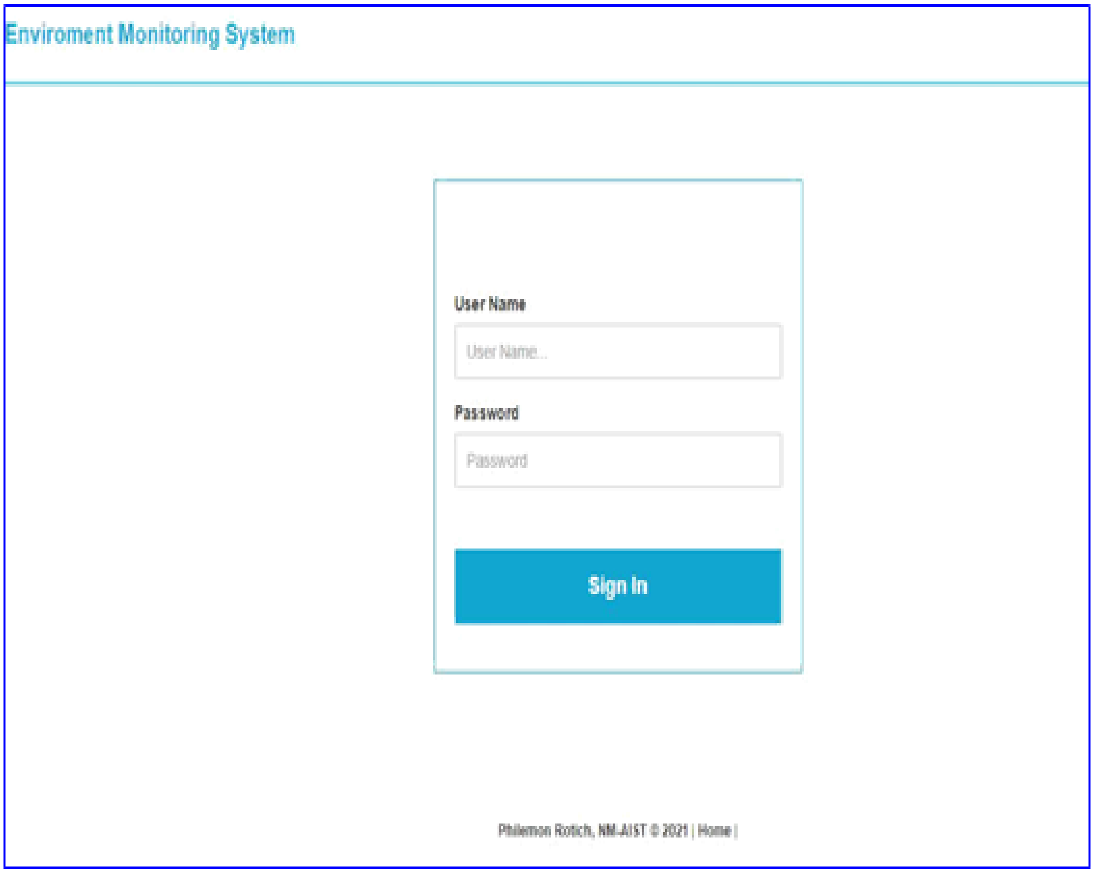


Figure 10: System administrator and system users‟ login page”

Figure 10 presents the user-authentication interfaces for the environment monitoring system administrator, factory manager, and technicians.

(ii) Environment monitoring panel Dashboard

The DLP Company's system administrator and factory manager have various levels of access to this login interface. The Interface shows real-time data from the sensor nodes on the graph in a time interval of 5 seconds, as shown in Fig. 11 and Fig. 12.

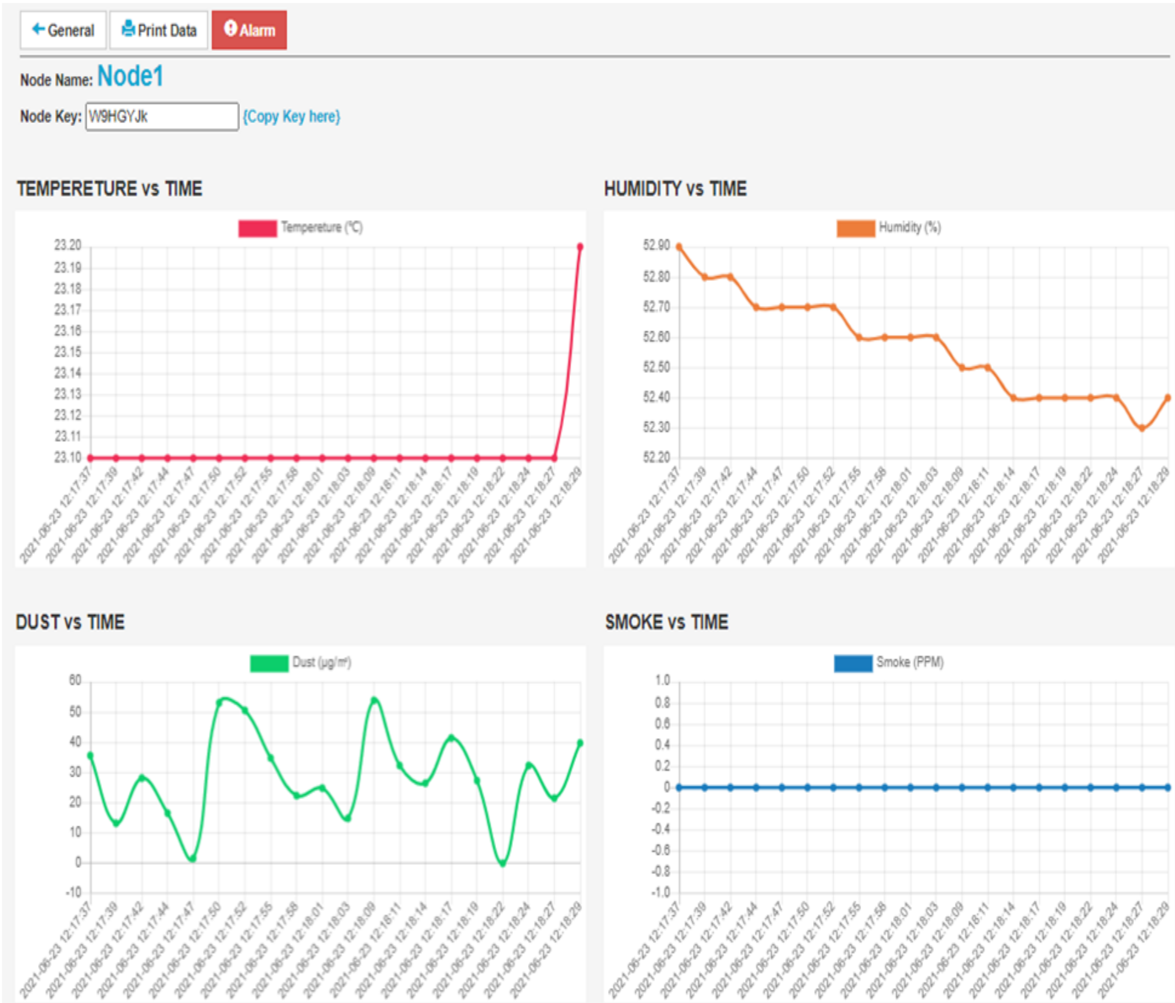


Figure 11: Environment monitoring dashboard

This information is useful for monitoring the environment from a remote place and incase the sensor values exceed the maximum parameter an alarm is delivered by e-mail and SMS to regulate the HVAC system and ventilation within the plant. The system end-user can only monitor nodes data, change user passwords and print or export data into an excel sheet for future analysis. On the other hand, the system administrator can access all the functionalities within the application.

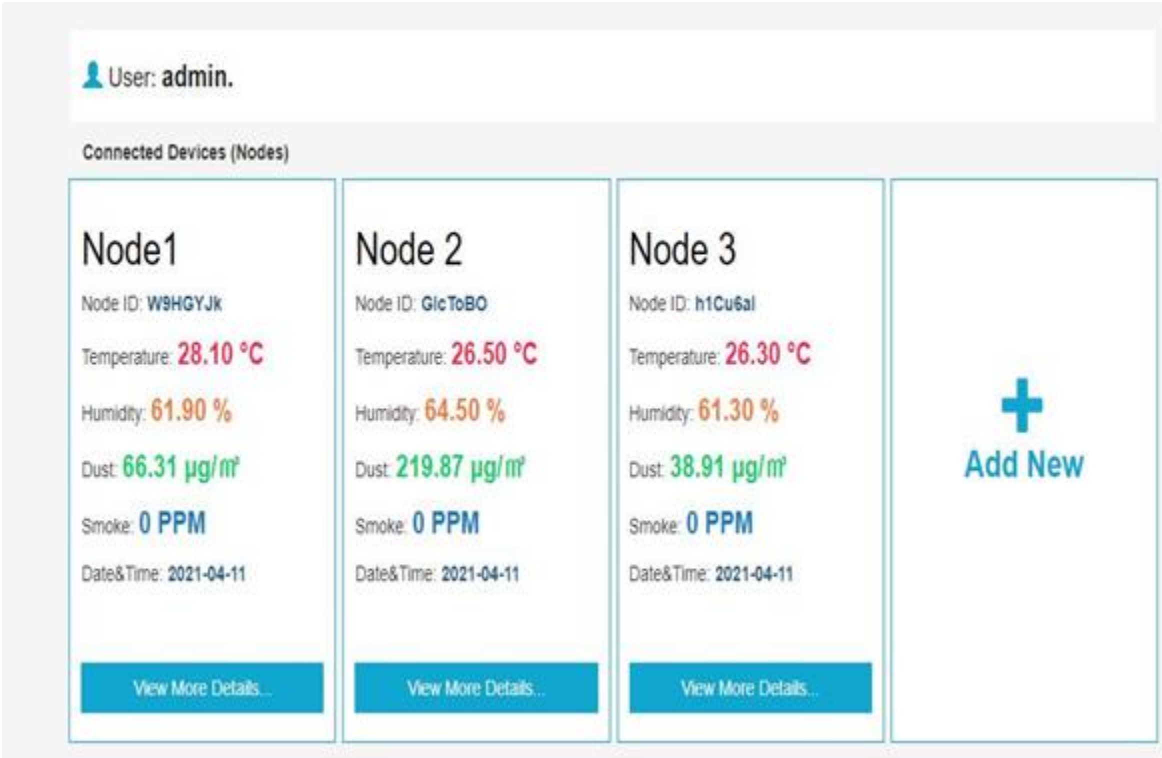


Figure 12: Summaries of the Sensor Nodes

(iii) User Management and Registration

The system administrator has the privilege of registering new users and removing users, while end-users of the system have some limitations when it comes to accessing some functionality.

A User request goes as follows:

(i) Apache receives a request for URL and forwards this request onto PHP.

(ii) PHP sends SQL-based "queries" to the database, which responds by producing the necessary information.

(iii) PHP formats the data into an HTML-based webpage, which is then returned to Apache.

(iv) Apache sends the website to the user's browser, which renders it.

3.11.3 Advantages of the Developed System

(i) Technical Feasibility**:** This system is specially designed for authorized users who can use the network and start secure communication online through the internet in the periphery of the existing network. The system is technically feasible. The system consists of clientserver architecture and is coded in JavaScript and PHP which are easily available. All the resources that are required for the system can be made available easily.

(ii) Availability of real-time data from sensors: Even when abnormal conditions are detected as they occur and remedies are available quickly, real-time data can help minimize downtime. An embedded controller linked to process sensors can also foresee potential failures and take action or can cause self-healing. Technicians can have remote access to sensor data at the comfort at their homes.

(iii) Use of Arduino UNO microcontroller: It is a low powered architecture, simple to get started with excellent online support, and rapid prototyping that is capable of sending data wirelessly to the server through a device with a large number of GPIOs and PWM capabilities, as well as being developer-friendly.

iv) Future development: This research project based on WSN can be further expanded by providing additional modules and functionalities to the industry person with the help of the Android app for achieving better control and monitoring of industries. Further, it allows room for integration of other systems which can be combined to ensure efficiency and industrial automation in the industry.

3.12 System Testing, Verification, and Validation

Unit testing was performed after developing the system to guarantee that every module is functioning properly and correctly. The evaluation of system performance was then performed to assess the efficiency and reliability of data monitoring. Finally, testing for user acceptance was done by including a representative group of DLP technicians.

3.12.1 Test Requirements

The test requirement for sensors, IoT devices and developed web application broadly revolves around security, analytics, device, networks, processors, operating systems, platforms and standards. Test requirements and goals include: involves large amount of data and its application by performing the user authentication testing.

Security testing: In the WSN environment, there are many users are accessing a massive amount of data. Thus, it is important to validate user via authentification by providing username and password.

(i) It`s important to check on data integrity and security of the developed system as it thentication, have data privacy controls as part of security testing.

(ii) Performance Testing: Performance testing is important to create strategic approach for developing and implementing a WSN testing plan.

(iii) Reliability and Scalability Testing: Reliability and scalability is important for building an IOT test environment which involves simulation of sensors by utilizing virtualization tools and technologies.

3.12.2 Unit Testing

This was done in the factory plant environment; each programmed sensor node and sink note prototype was tested to ensure that it generates a valid output that can be displayed on a graph and to verify the efficiency of every device unit before integrating with other modules. It is primarily used to thoroughly assess each functional unit. System unit testing was performed in all separate system functional units like login, account settings, node monitoring dashboard, reports, and user registration. It is majorly performed to validate and test each functional unit separately.

(i) User Authentication Unit

This test aimed to see whether a registered user could log in and out of the web portal successfully using their provided login credentials. The results of this test are shown in Table 1.

Table 2: User authentication unit testing

System requirements	Results
1. Only registered users should be able to log in and out of the system using their password and username	PASS
2. All registered should be able to recover their password via the system	PASS

3.12.3 Integration Testing

This involved the testing and authentication of two or more interconnected system functional units to ensure that they operate properly together (Nidhra, 2012). Various operating groups or units were combined and checked to see whether they functioned accurately. To ensure that a successful authentication brings a user to the home page, integration testing was performed. Another component of this system's integration testing is to guarantee that data sharing and communication between the web-based and mobile systems are working correctly.

3.12.4 System Testing

System tests were carried out on a fully-system to see whether the established system meets the functional criteria and is consistent in its usability. The outcome of all integrated components passing the integration testing phase is system testing. It is not concerned with the structural aspects of source codes, but rather with the practical aspects that are apparent to the end-user (Nidhra, 2012). All tests done on integration were passed by the system while running on a local server installed in the plant. The data entered in the database by the web-based framework could be successfully recovered by the mobile application.

3.12.5 Acceptance Testing

After the internal staff of the DLP company tested the system prototype, it was transferred to the company`s technicians to collect their observations on the system usability. This activity drew a total of twenty-five technicians. The responses of the survey were gathered on a five-point Likert scale as 5: Strongly Agree, 4: Agree, 3: Neutral, 2: Disagree, and 1: Strongly Disagree as the scale responses (Lozano *et al.*, 2008). The Instructions given to the users on how to fill the survey questionnaire is found in the attached Appendix 1.

Table 3: System acceptance testing survey questionnaire

NO	Acceptance Variable	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree	Mean Score
1.	The interfaces of the application are easy to use and interact with.	0	0	0	15	10	3.94
2.	There are no difficulties with compatibility between the framework and various web devices/browsers.	0	0	13	6	4	4.22
3.	I don't need any additional instruction or assistance to completely navigate other application interfaces.	0	5	0	9	11	3.76
4.	The system would make it easier for users to access their environment monitoring data and can use the collected data for data analysis	0	0	0	8	17	4.72
5.	I am satisfied with the general performance of the application	0	0	0	17	8	3.92
6.	I will recommend the use this system to others	0	0	2	14	9	4.138

The results of the system acceptance test are displayed in Table 2. The average score for each response was greater than 3.80, indicating that most respondents were pleased with the general system usability and efficiency.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Results

4.1.1 System Implementation Overview

This chapter discusses the results of the developed system relative to the research/study objectives. The general objective was to develop an indoor industrial environment monitoring system. The first objective was to collect and analyze the requirements for developing an indoor industrial environment system based on wireless sensor network technology. The second objective was to develop a wireless network-based industrial environment monitoring system. The third objective entailed the validation of the developed system. The system was implemented with three transmitter modules; Node1, Node 2, and Node 3, and one receiver module which acted as the base station. The system has incorporated Arduino Uno microcontroller and Raspberry Pi minicomputer and various sensors. With the use of WSN, this system will send the data remotely to a web server via Wi-Fi which can be accessed via web page over a browser, and information about the current indoor environment and concentration of gases in the plant can be displayed on the screen. The developed system was implemented on a raspberry pi module running on a Linux environment, which supports SMTP (simple mail transfer protocol). The SMTP is an application layer protocol that allows an e-mail to be sent to a registered address from the SMTP server. E-mail is sent via an SMTP server, where an application will log in into an SMTP server as an e-mail user and send e-mail to a given registered user address, this can be done via a special API. Google mail (Gmail), and PHP Mailer library are used in the implementation of the system.

For SMS, SMPP protocol is used to communicate with an SMS server that is capable of receiving a request and sendin an SMS over a telecommunication network through a special registered sender ID. Access to a third party or application developer who needs to send SMS can be provided via an API, an API is provided by the company that owns/render the service. For this case, Safaricom is used as the telecommunication service provider.

A notification can be sent to mobile phones through E-mail and SMS if an alarm is generated and immediate action will be taken to prevent any danger from occurring. System specifications include hardware requirements and software requirements. The system is successful when implemented as proposed and the following are the snapshots of the results.

4.1.2 Sensing Subsystem

This subsystem entails correctly interfacing Arduino Uno with various sensors, and it functions well and produces the desired results as outlined in the project. Temperature, humidity, dust, and smoke, which are all monitored in the environment, were accurately detected and sent to the processing subsystem. Figure 13 and Fig. 14 shows the prototype of the transmitter module and developed sensor nodes.

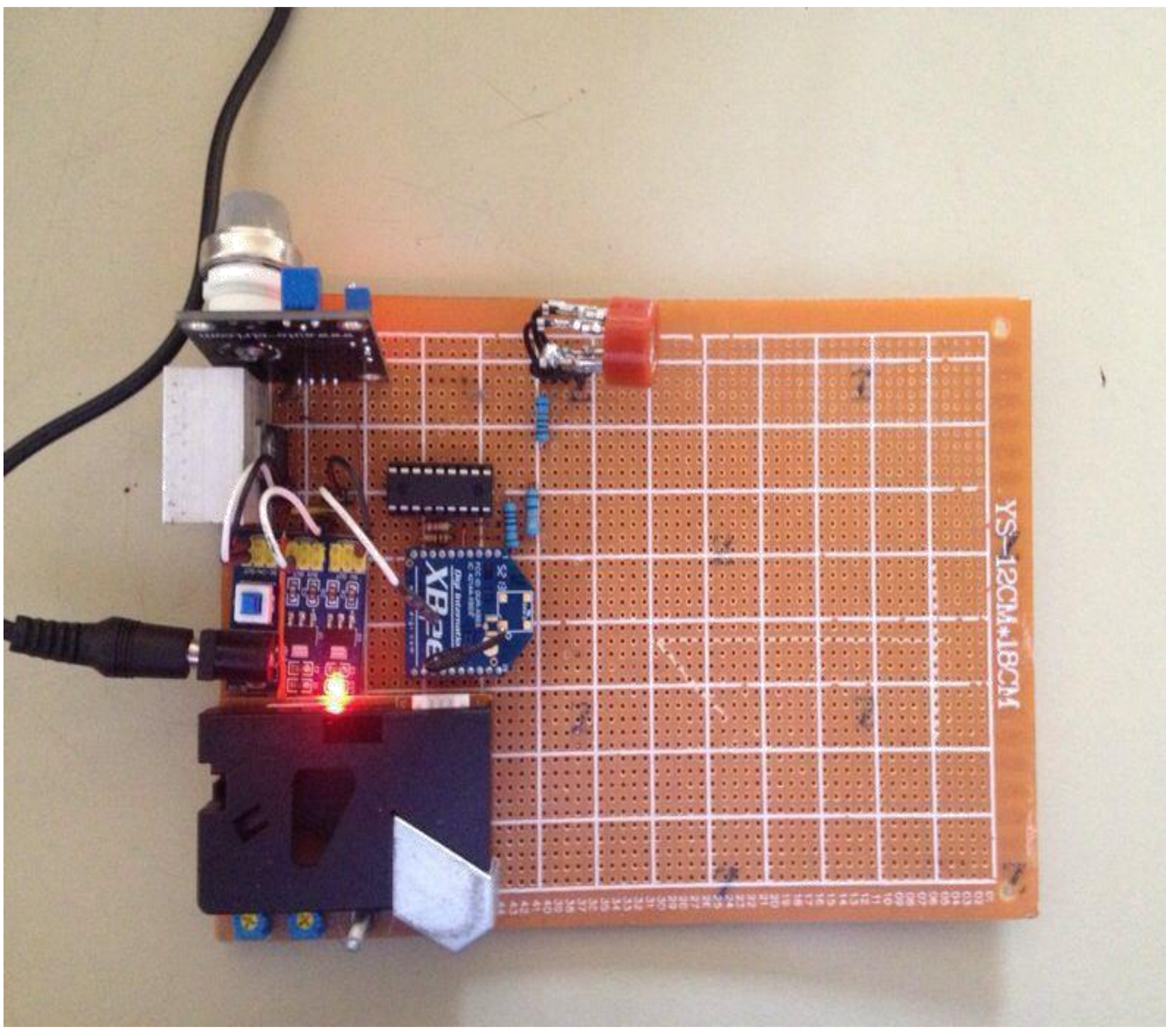


Figure 13: System acceptance testing survey questionnaire

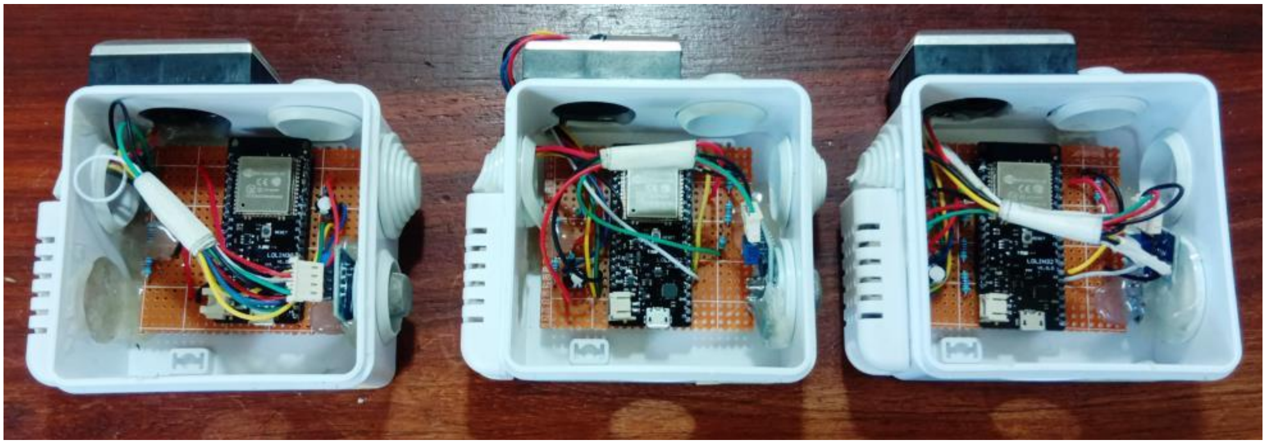


Figure 14: Developed Sensing Nodes

4.1.3 The Web-Based Monitoring System

The web-based monitoring system developed was primarily accessible through the browsers. The web-based monitoring system has functionalities that include user management, sensor node management, account settings, reports dashboard, and monitoring dashboard.

Factory managers and database administrators are the system's privileged users. The database administrator may increase or decrease the number of functionalities that the manager has access to. On the end-user's side, there are limited functionalities which include live data environment monitoring dashboard, print report and password recovery as shown in figure 19 and figure 20 below.

4.1.4 Real-Time Environment Monitoring

The data received in a database can be directly displayed on the screen to enable the device user to visualize the current state of the factory's indoor environment remotely. Figure 15, Fig. 16 and Fig. 17 below show the live monitoring of the environment condition.

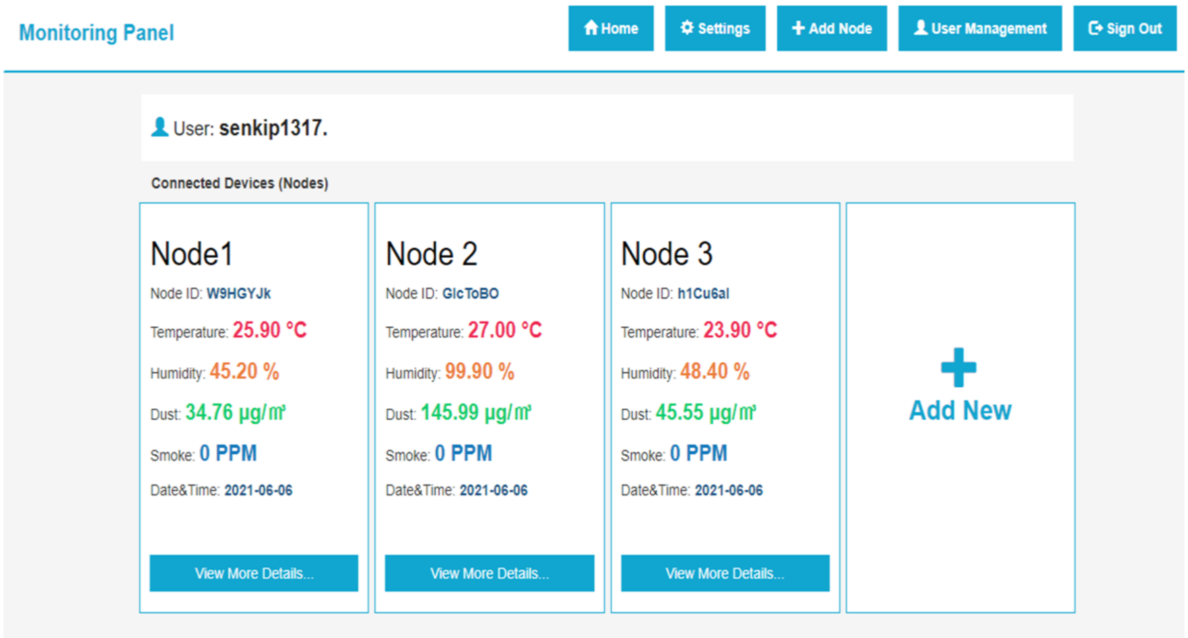


Figure 15: Summary of environment monitoring dashboard nodes available to the system administrator

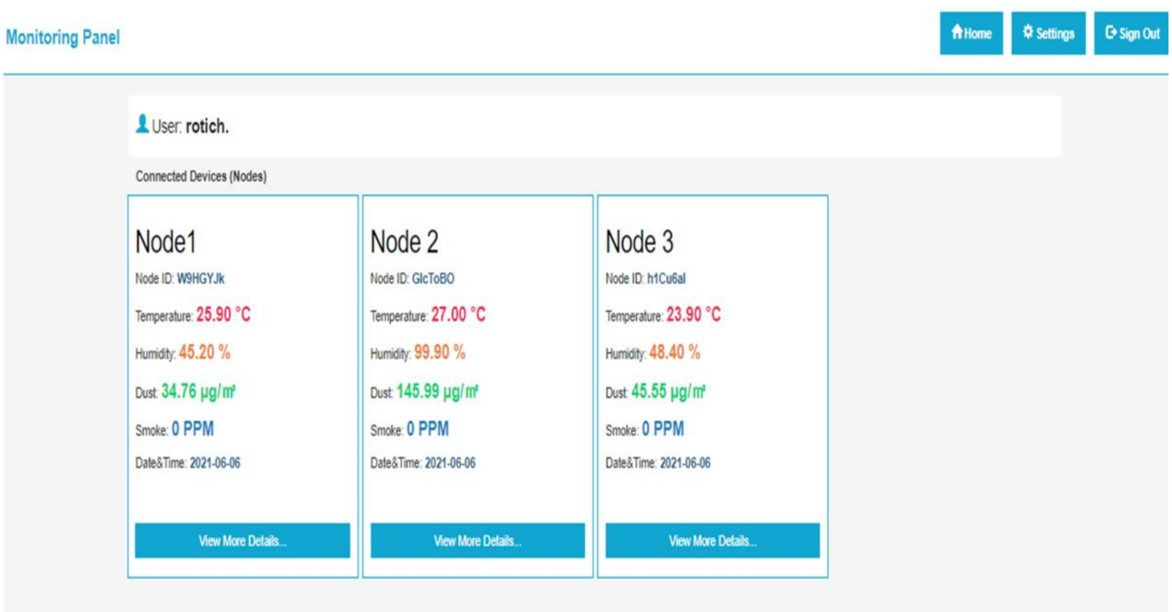


Figure 16: Summary of environment monitoring dashboard nodes available to the technicians

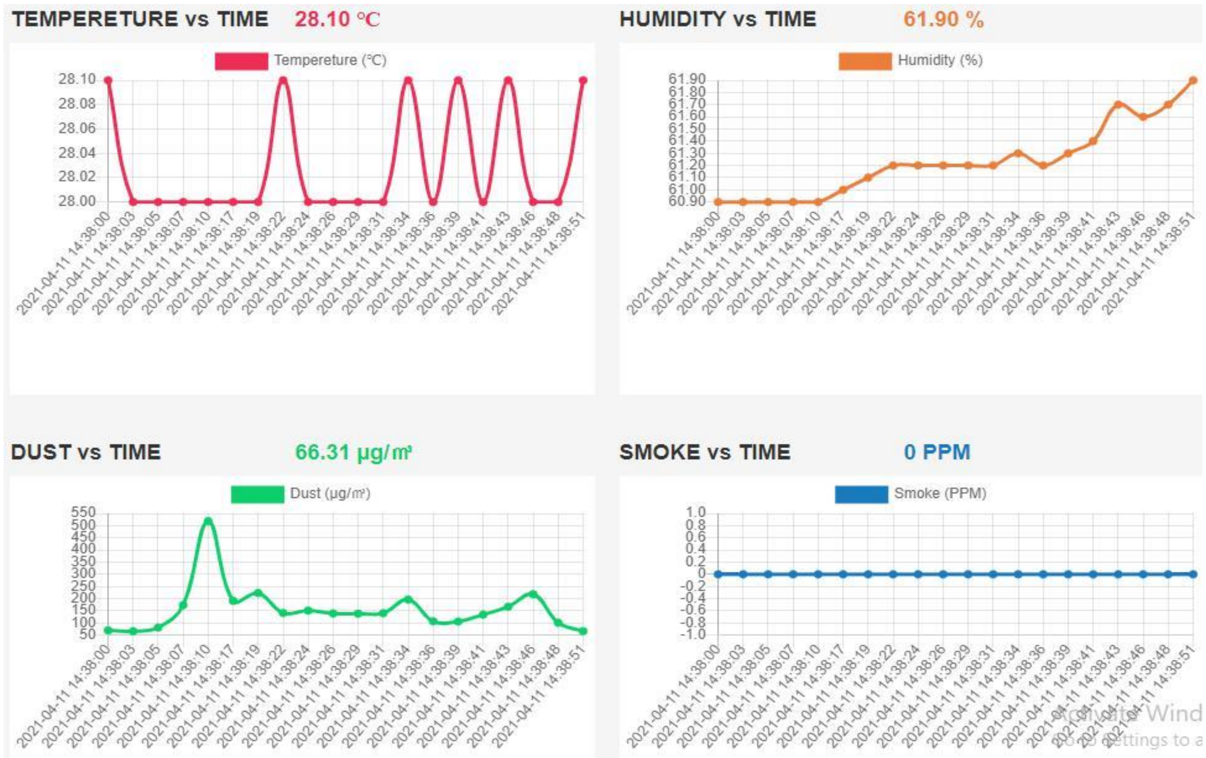
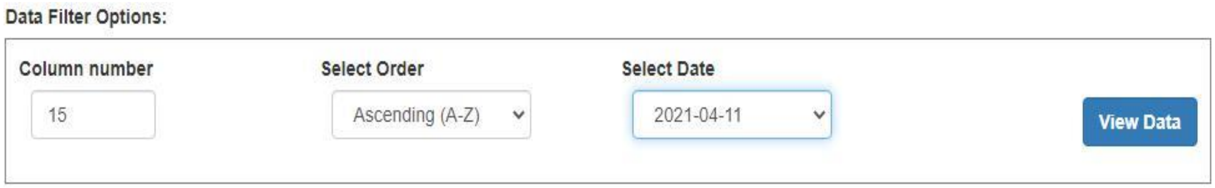


Figure 17: Real-time environment monitoring data



Node name: **Node1**

No	Date & Time	Tempereture (°C)	Humidity (%)	Dust (µg/m³)	Smoke (PPM)
1	2021-04-10 13:30:45	92.00	63.00	139	31
2	2021-04-10 13:30:50	17.00	34.00	129	73
3	2021-04-10 13:30:55	67.00	63.00	26	64
4	2021-04-10 13:31:00	26.00	75.00	145	198
5	2021-04-10 13:31:05	70.00	19.00	107	71
6	2021-04-10 13:31:11	19.00	59.00	158	144
7	2021-04-10 13:31:16	50.00	19.00	141	31
8	2021-04-10 13:31:21	80.00	68.00	185	106
9	2021-04-10 13:31:26	89.00	82.00	87	71
10	2021-04-10 13:31:53	75.00	34.00	81	165
11	2021-04-10 13:31:58	56.00	72.00	97	75
12	2021-04-10 13:32:03	23.00	87.00	73	112
13	2021-04-10 13:32:08	35.00	36.00	79	102
14	2021-04-10 13:32:14	11.00	56.00	193	161
15	2021-04-10 13:32:19	90.00	86.00	169	16

Figure 18: Sensor data stored in the database and displayed in the webpage

The Plant system administrator can view the status of the factory environment by logging in as administrator in the developed system and recall the history from any day up to date according to the preference requested as shown in Fig. 18. The report can also be generated by downloading it in pdf. format or CSV file format, which can finally be printed and presented for management purposes or further research analysis.

4.1.5 Warehouse Live Data Node 1

The level of CO, temperature, humidity, and dust concentration was measured at three different areas within the warehouse, and average values taken at a time interval of thirty seconds as shown in the Fig. 19, Fig. 20, and Fig. 21. During the measurement, it is seen that the dust emission into the air is higher as compared to other pollutants. The time interval can be adjusted to any value preferred by the factory manager. The maximum and minimum values of the environmental parameters can also be adjusted according to the industrial standards to ensure safety and a conducive working environment.

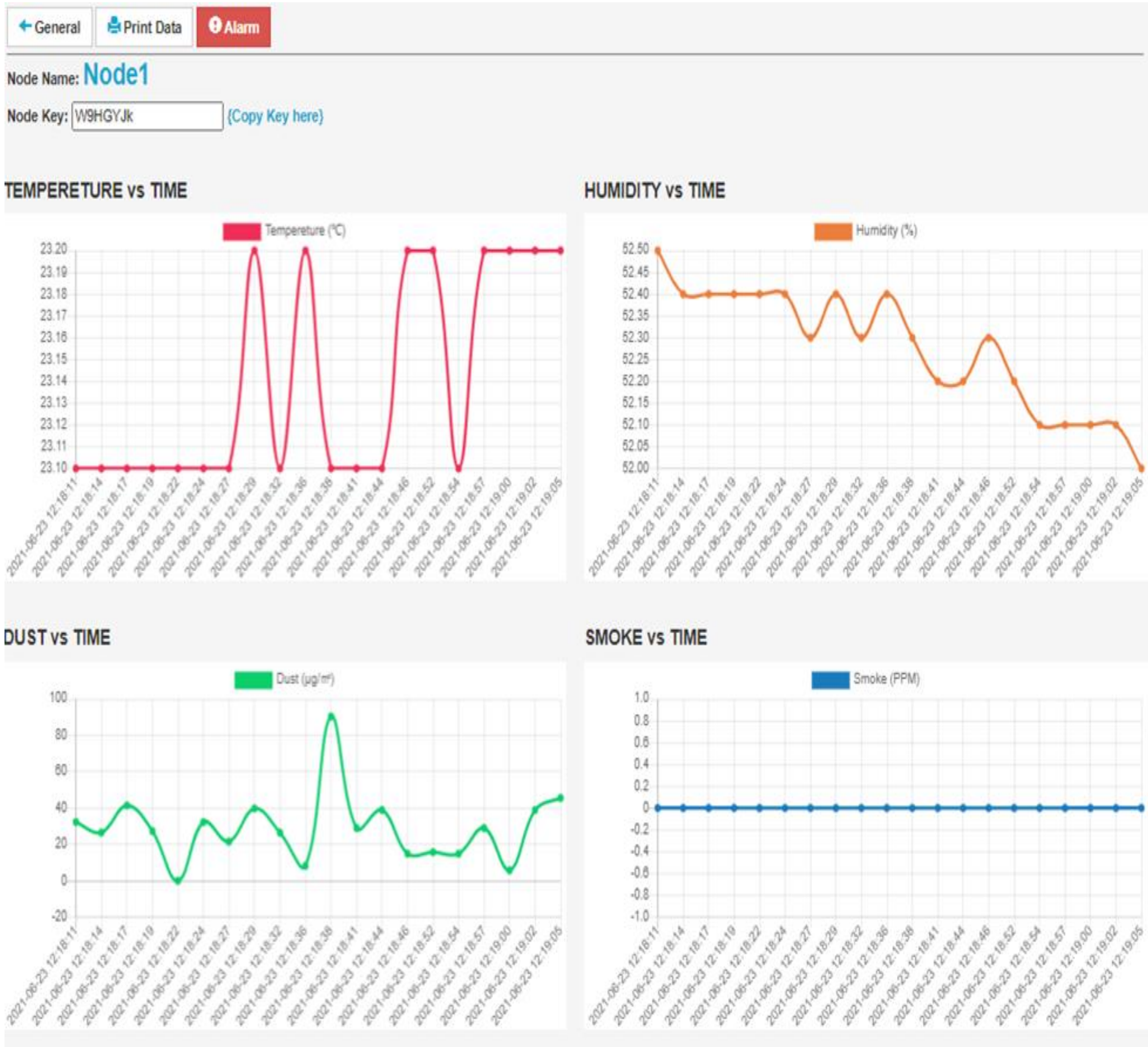


Figure 19: Environment live data level at warehouse Area 1 from sensor node 1

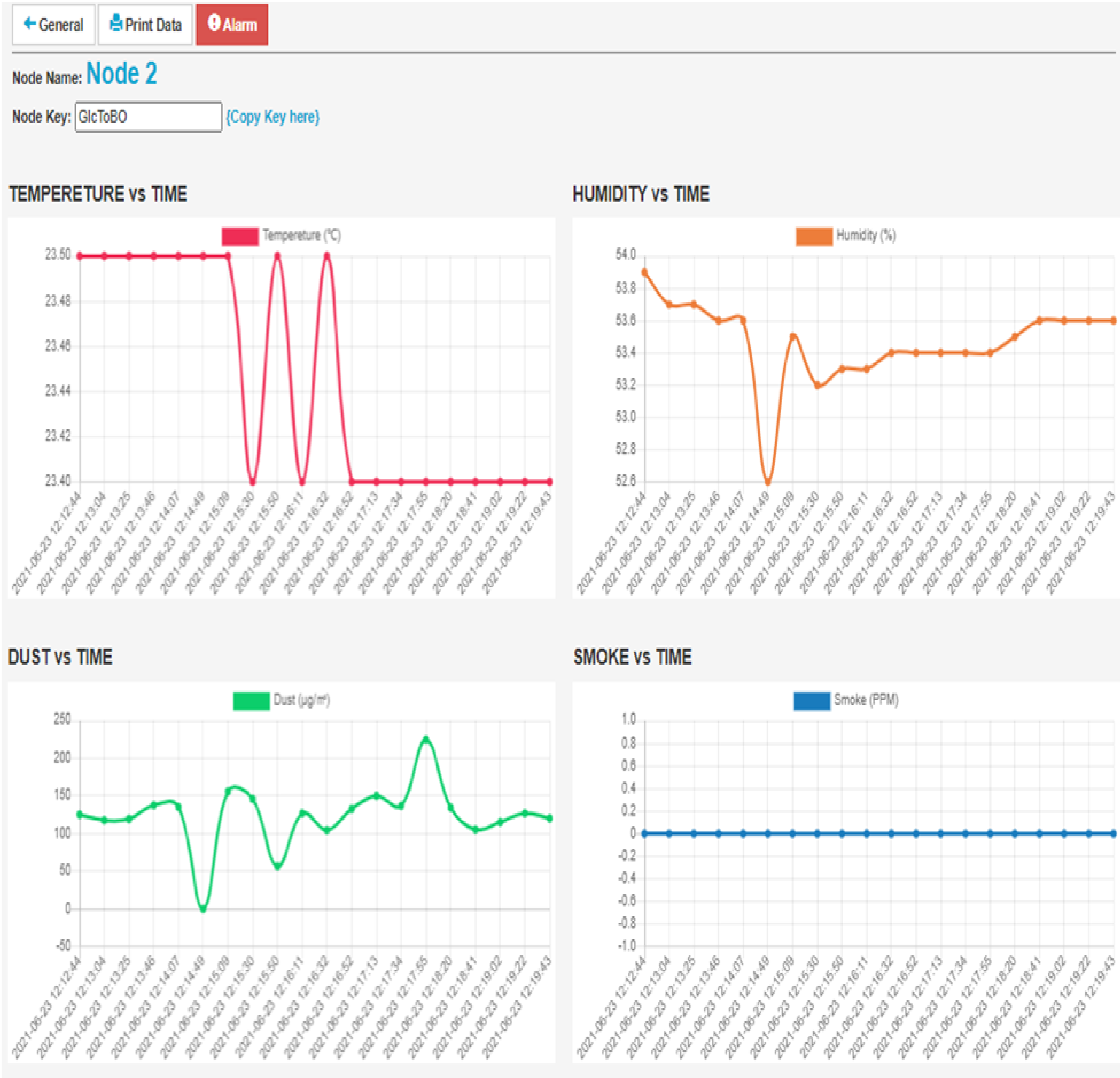


Figure 20: Environment live data level at warehouse Area 2 from sensor node 2

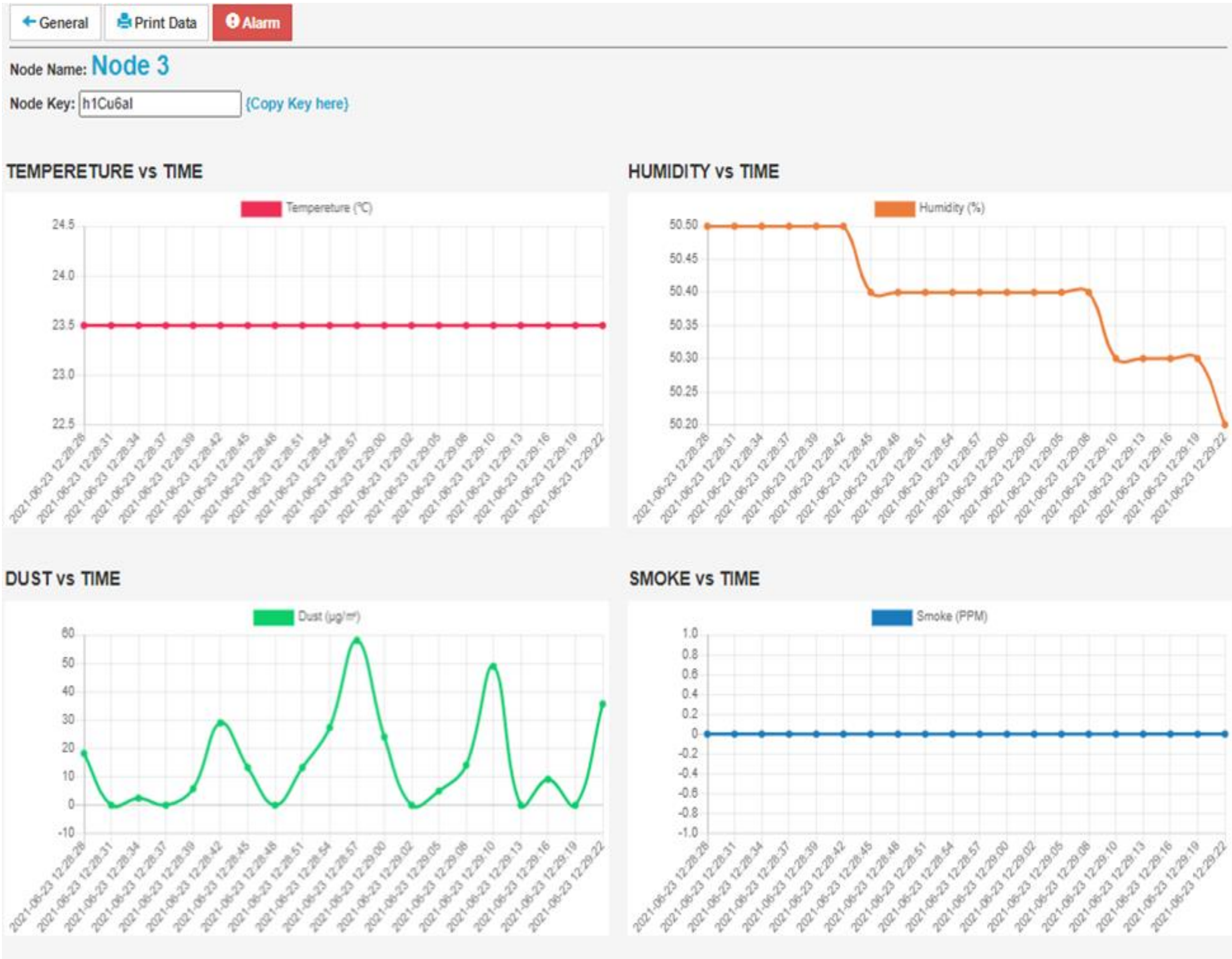


Figure 21: Environment live data level at Assembly line Area 3 from sensor node 3

4.2 Discussion

The designed web-based environment monitoring system could incorporate all of the individual sensors, wireless communication, and programming components into a single module. Furthermore, within a digital assembly environment, the device could detect substantial spatial variations in environmental conditions, which is advantageous in a multi-sensor system since a system mounted at a plant can have several centrally-positioned sensors gathering data in real-time for factory monitoring. This also demonstrates that the system can meet the project aim of gathering data and giving out notifications through SMS and e-mail in case of an alarm.

The system can also detect and activate the HVAC system in case the industrial parameters exceed the maximum value. While the monitoring system met the design aims for a prototype, functioning system, and proof of concept system, there are still several things that could be changed in future iterations, such as the user interface to make it more accessible to a customer base and the addition of more sensors e.g., electrostatic charge sensor to monitor the amount of electrostatic discharge within the assembly line. Generally, the monitoring system has demonstrated the practicability of producing a low-power plant monitoring system with several wireless devices. As radio, ZigBee standards, and MCU technology improve, users will be able to build smaller and less expensive wireless networks. Before these systems gain widespread adoption in the industry, the emerging technologies must be made simpler and more stable.

Wireless monitoring systems can be powerful tools in monitoring the industrial environment if their implementation can be streamlined for any engineer to understand and use the technology.

CHAPTER FIVE

CONCLUSION AND RECOMMENDATIONS

5.1 Conclusion

With the advent of technology, it is anticipated that internet access will be available everywhere. The development of an industrial environment monitoring system based on a network of wireless sensors has proven to automate industrial operations.

The system was designed to provide the live monitoring of the surrounding environment and appliances within the plant and to inform the responsible person to take appropriate measures by regulating the parameters. The system uses the IEEE 802.15.4 protocol to transmit data. The database is designed for storing the history information of the measurement taken so as to provide the accessibility of the records and report at any time and for the future use and analysis. The test was conducted by putting the sensor nodes at different areas and the measurement obtained from the sensors are transmitted directly to the central server (data base) via ZigBee module. The obtained results indicate that the level of dust at all sampled areas is higher than the level of CO and temperature and humidity. This means that dust is in more concentration than other parameters. This will help the management a ways of reducing this dust to avoid device parts damage when stored.

This project offers a wide range of benefits to the Assembly plant management and university researchers, the management at the plant can safely monitored factory environment remotely hence safety of the devices and staff which is very important, to the researchers‟ available data collected from monitoring system can further be analyzed using use of machine learning and artificial intelligence technology to find more useful information. This system has many advantages compared to the wired systems. Among the advantages are: (i) the system is operated remotely so the data reaches the user easily and instantly, (ii) The cost of developing and deploying the system is cheap compared to the wired systems (iii) it can also be installed even in terrestrial areas.

5.2 Recommendations

Industrial environment monitoring involves the measurement of different parameters including humidity, temperature, carbon dioxide (CO_2), smoke, pressure, dust, earthquakes, Nitrogen Oxides (NO_2) concentration, etc. Due to the limitation of scope and time, the developed system

was designed to monitor only the concentration of CO, temperature, humidity and dust emitted to the environment. For better control of the environmental safety and pollution, it highly needs the above mentioned parameters to be considered. This system is limited by the coverage distance so it does not go beyond 100 meters in line of sight. To increase the coverage distance the transmitters and receivers with high power should be used.

On the other hand, due to time limitation, Inventory management system was not achieved as it was indicated in the proposal, this is problem that needs to be considered with urgency to ensure industrial automation and efficiency of operation. The inventory system can be integrated with environment monitoring system to help in tracking of device in the assembly line and the warehouse to enable a fully automated factory.

REFERENCE

Adams, J. (2005). *ZigBee Wireless Technology and the IEEE 802.15. 4 Radio–Enabling Simple Wireless. Texas Wireless Symposium*. https://www.google.com

Adhav, K., & Vibhute, M. (2020). Environment monitoring system using low cost sensors. *International Engineering Research Journal*, 2020, 141–145.

Alliance, Z. (2007). *ZigBee Specification*. https://www.google.com

Andrady, A., Aucamp, P. J., Austin, A. T., Bais, A. F., Ballare, C. L., Barnes, P. W., & Zepp, R. G. (2017). Environmental effects of ozone depletion and its interactions with climate change: progress report, 2015. *Photochemical & Photobiological Sciences*, *16*(2), 107-145. https://doi.org/10.1039/c6pp90004f

Ashna, K., & George, S. (2013). *GSM based automatic energy meter reading system with instant billing*. International Mutli-Conference on Automation, Computing, Communication, Control and Compressed Sensing (IMac4s), 65–72. https:// doi. org/10. 1109/ iMac4s. 2013. 6526385

Buratti, C., Conti, A., Dardari, D., & Verdone, R. (2009). An Overview on Wireless Sensor Networks Technology and Evolution. *Sensors, 9*(9), 6869–6896. https:// doi. org/10. 3390/ s90906869

Cao, X., Liu, L., Cheng, Y., & Shen, X. (2017). Towards Energy-Efficient Wireless Networking in the Big Data Era: A Survey. *IEEE Communications Surveys & Tutorials*, *20*(1), 303–332. https://doi.org/10.1109/COMST.2017.2771534

Crawford, T., & Hussain, T. (2017). *A Comparison of Server Side Scripting Technologies. Proceedings of the 2017 International Conference on Software Engineering Research and Practice.* https://www.google.com

Deshmukh, A. D. (2016). *A Low Cost Environment Monitoring System Using Raspberry Pi and Arduino with Zigbee*. *International Conference on Inventive Computation Technologies* (764–769) https://www.researchgate.net

Dunlap, E. (2020). *National Fire Protection Association* (pp. 243–244). https:// doi. org/ 10.

1201/ 9781003075714-36

Dutta, A., Borah, R., Barman, S., & Dubey, A. (2018). *IoT Based Industrial Equipment Controlling and Parameter Monitoring System*, (1)1, 50–53.

Farnell. (2013). *Arduino Uno Datasheet*. Datasheets. https:// www. farnell. com/ datasheets/ 1682209.pdf

Fonollosa, J., Solórzano, A., & Marco, S. (2018). Chemical sensor systems and associated algorithms for fire detection: A review. *Sensors*, *18*(2), 1–38. https://doi.org/10.3390/s18020553

Ghiasi, R., & Ghasemi, M. R. (2020). Feature Selection in Structural Health Monitoring Big Data Using a Meta-Heuristic Optimization Algorithm. *Journal of Computational Methods In Engineering*, *39*, 1–27.

Goel, A., Ray, S., Agrawal, P., & Chandra, N. (2012). *Air Pollution Detection Based On Head Selection Clustering And Average Method From Wireless Sensor Network*. 2012 Second International Conference on Advanced Computing & Communication Technologies, 434–438. https://doi.org/10.1109/acct.2012.18

Goel, A., Ray, S., & Chandra, N. (2012). *Air Pollution Detection Based On Head Selection Clustering And Average Method From Wireless Sensor Network*. https:// doi. org/10. 1109/ ACCT.2012.18

Gupta, V., Poursohi, A., & Udupi, P. (2010). S*ensor. Network: An open data exchange for the web of things. 2010 8th IEEE International Conference on Pervasive Computing and Communications Workshops*. https://doi.org/10.1016/j.pmcj.2017.02.002

Hackmann, G., Guo, W., Yan, G., Lu, C., & Dyke, S. (2014). Cyber-Physical Codesign of Distributed Structural Health Monitoring With Wireless Sensor Networks. *Transactions on Parallel and Distributed Systems*, *25*(1), 63–72. https://doi.org/10.1145/1795194.1795211

He, Y. L., Geng, S., Peng, X., Hou, L. G., Gao, X. K., & Wang, J. H. (2016). *Design of outdoor air quality monitoring system based on ZigBee wireless sensor network*. https://doi.org/10.1109/ICSICT.2016.7998923

Jung, Y., Lee, Y. K., Lee, D., Ryu, K., & Nittel, S. (2008). Air Pollution Monitoring System based on Geosensor Network. *IEEE International Geoscience and Remote Sensing Symposium*. https://doi.org/10.1109/IGARSS.2008.4779615

Khedo, K., Rajiv, P., & Avinash, M. (2010). A Wireless Sensor Network Air Pollution Monitoring System. *International Journal of Wireless & Mobile Networks*, *2*(2), 31–45. https://doi.org/10.5121/ijwmn.2010.2203

Kritica, B. J. (1980). *The Smart Innovation in Industrial Data Acquisition System Using IOT*. https://www.google.com

Letkowski, J. (2015). Doing database design with MySQL. *Journal of Technology Research, 6*(1), 1–15.

Lozano, L., García-Cueto, E., & Muñiz, J. (2008). Effect of the Number of Response Categories on the Reliability and Validity of Rating Scales. *Methodology*, *4*(2), 73–79. https://doi.org/10.1027/1614-2241.4.2.73

Nidhra, S. (2012). Black Box and White Box Testing Techniques - A Literature Review. *International Journal of Embedded Systems and Applications*, *2*(2), 29–50. https://doi.org/10.5121/ijesa.2012.2204

Nikhade, S. (2015). *Wireless sensor network system using Raspberry Pi and zigbee for environmental monitoring applications*. https:// www. infona. pl/ resource/ bwmeta1. element. ieee-art-000007225445

No, I., & Ali, K. (2017). *Available Online at www.ijarcs.info International Journal of Advanced Research in Computer Science A Comparative Study of Well Known Sorting Algorithms*. *8*(1), 2015–2018.

Othman, M. F., Shazali, K., Murugam, K., Tilak, S., Abu-Ghazaleh, N. B., Heinzelman, W., Valverde, J., Rosello, V., Mujica, G., Portilla, J., Uriarte, A., Riesgo, T., Šećerov, I., Dolinaj, D., Pavić, D., Milošević, D., Savić, S., Popov, S., & Živanov, Ž. (2012). Wireless sensor network applications: A study in environment monitoring system. *Procedia Engineering*, *41*(2), 273–285. https://doi.org/10.4236/wet.2019.101001

Qiu, Z., Ali, M. A., Nichol, J. E., Bilal, M., Tiwari, P., Habtemicheal, B. A., Almazroui, M., Mondal, S. K., Mazhar, U., Wang, Y., Sarker, S., Mustafa, F., & Rahman, M. A. (2021).

Spatiotemporal Investigations of Multi-Sensor Air Pollution Data over Bangladesh during COVID-19 Lockdown. *Remote Sensing*, *13*(5), 1-27. https://doi.org/10.3390/rs13050877

Raspberrypi. (2019). *Datasheet Raspberry Pi Model B*. Raspberrypi.Org. https://www.google.com

Rotich, P., Hillary, R., Sam, A., & Geofrey, A. (2020). Early Fire Detection System in Tanzania Markets. *Engineering Proceedings, 2*(1), 51–55. https://doi.org/10.3390/ecsa-7-08215

Shinde, K. S., & Bhagat, P. H. (2017). *Industrial Process Monitoring Using IoT*. https://www.google.com

Solihin, W., Eastman, C., Lee, Y., & Yang, D.-H. (2017). A simplified relational database schema for transformation of BIM data into a query-efficient and spatially enabled database. *Automation in Construction*, *84*, 367–383. https:// doi. org/ 0. 1016/j. utcon.2017.10.002

Sumithra, P., Nagarajan, R., Padmavathi, M., & Malarvizhi, M. (2018). *IoT Based Industrial Production Monitoring System Using Wireless Sensor Networks*, 5(11), 255-262. https://doi.org/10.22161/ijaers.5.11.35

Systems, M. T. E. (2020). *Low Cost Pollution Control and Air Quality Monitoring System using Raspberry Pi for Internet of Things. 2017 International Conference on Energy, Communication, Data Analytics and Soft Computing (2319–2326).* https://www.google.com

Ullo, S. L., & Sinha, G. R. (2020). *Advances in Smart Environment Monitoring Systems Using IoT and Sensors*, 20(11), 1–18. https://doi.org/10.3390/s20113113

Valverde, J., Rosello, V., Mujica, G., Portilla, J., Uriarte, A., & Riesgo, T. (2012). Wireless sensor network for environmental monitoring: Application in a coffee factory. *International Journal of Distributed Sensor Networks*, 2012, 1-18. https://doi.org/10.1155/2012/638067

Vujovic, V., & Maksimovic, M. (2014). *Raspberry Pi as a Wireless Sensor node: Performances and constraints. 37th International Convention on Information and Communication Technology, Electronics and Microelectronics.* https://scholar.google.com/

Wang, D., & Zhang, J. (2006). Building Wireless Sensor Networks by Zigbee Technology. *Journal of Congqing University*, *29*, 95–97.

Yang, H., & Yang, S. (2009). *Connectionless Indoor Inventory Tracking in Zigbee RFID Sensor Network*. https://scholar.google.com/

Zhang, X., Zhang, M., Meng, F., Qiao, Y., Xu, S., & Hour, S. (2018). A Low-Power Wide-Area Network Information Monitoring System by Combining NB-IoT and LoRa. *IEEE Internet of Things Journal*, *6*(1), 590–598. https://doi.org/10.1109/JIOT.2018.2847702

Zhou, Y., & Lee, W. (2019). D*esign of an Industrial IoT-Based Monitoring System for Power Substations Design of an Industrial IoT-Based Monitoring System for Power Substations Long Zhao Igor Matsuo. IEEE Transactions on Industry Applications*.https://scholar.google.com/

APPENDICES

Appendix 1: Questionnaire for System Validation

Introduction

The primary goal of this survey is to obtain feedback on the usability and efficiency of the developed web-based environment monitoring system.

Respondent post:

a) Administrator of the system

b) Technicians and factory supervisor

Please tick the box corresponding to your perception of the developed system?

Table 3 System acceptance testing survey questionnaire

NO	Acceptance Variable	Strongly Disagree	Disagree	Neutral	Agree	Strongly agree
1.	The interfaces of the application are simple to use and interact with.					
2.	There are no difficulties with compatibility between the framework and various web devices/browsers.					
3.	I don't need any additional instruction or assistance in order to completely navigate other application interfaces.					
4.	The system would make it easier for users to access their environment monitoring data and can use the collected data for data analysis					
5.	I am satisfied with the overall performance of the system					
6.	I will recommend the use this application to others					

Appendix 2: Arduino UNO Codes for environment monitoring panel

```
/* Sensor Node firmware for Data Collection
 *  Made by Philemon Rotich, Researcher, NM-AIST
 *  Arusha, Tanzania
 */
//Libraries used
#include<ArduinoJson.h>
#include<WiFi.h>
#include<HTTPClient.h>
//sensors
#include<Adafruit_Sensor.h>
#include<DHT.h>
#include<DHT_U.h>
#define DHTPIN 17
//sensor type
#define DHTTYPE DHT22 // DHT 22 (AM2302)
DHT_Unified dht(DHTPIN,  DHTTYPE);
//smoke
#include <MQ2.h>
#define MQ2_PIN A6
//smoke
MQ2 mq2 (MQ2_PIN);
//End of libraries here
//Wi-FI credentials
/******IMPORTANT*********
const char* ssid = "vivo 1802";            //Wi-Fi name here
const char* password = "Telkom2020";    //Wi-Fi Password here
const String serverIpAdress = "http://192.168.43.171/"; //Change it here depending on you server adress
```

```
String Node_KEY = "h1Cu6aI"; //**************Changing key is needed during registration
WiFiClient wifi;
int status = WL_IDLE_STATUS;
//dust sensor
int dustSensor = A7;
int ledPin = 16;
int samplingTime = 280;
int deltaTime = 40;
//int sleepTime = 9680;
float voMeasured = 0;
float calcVoltage = 0;
float dustDensity = 0;
//system data
float temp;
float hum ;
float dust;
float smoke;
float dataInterval;
char nodeStatus = 'Y';
long double requestInterval = 5000;
void setup() {
mq2.begin();
ht.begin();
pinMode(ledPin,OUTPUT);
sensor_t sensor;
// initialize serial communications and wait for port to open:
Serial.begin(115200);
while (!Serial) {
```

```
 ; // wait for serial port to connect. Needed for native USB port only }
WiFi.begin(ssid, password);
Serial.print("Attempting to connect to SSID: ");
Serial.print(ssid);
WiFi_connect(); // connecting to a WiFi
Serial.print("IP address: ");
Serial.println(WiFi.localIP());
Serial.println("Connected");}
void loop() {
 //Device Data
// Get temperature event and print its value.
sensors_event_t event;
dht.temperature().getEvent(&event);
if (isnan(event.temperature)) {
//Serial.println(F("Error reading temperature!"));}
else {
 temp = event.temperature;}
// Get humidity event and print its value.
 dht.humidity().getEvent(&event);
if (isnan(event.relative_humidity)) {
//Serial.println(F("Error reading humidity!")); }
else {
hum = event.relative_humidity;}
smoke = mq2.readSmoke(); //read smoke
//read Dust sensor
digitalWrite(ledPin,LOW); //power on the LED

delayMicroseconds(samplingTime);
voMeasured = analogRead(dustSensor); //read the dust value
delayMicroseconds(deltaTime); digitalWrite(ledPin,HIGH);
// turn the LED off
```

```
// 0 - 5V mapped to 0 - 1023 integer values

// recover voltage

calcVoltage = voMeasured * (5.0 / 1024.0);

//linear eqaution taken from http://www.howmuchsnow.com/arduino/airquality/

dust = 170 * calcVoltage - 0.1; //dust density //check for wifi connection

if (WiFi.status() == WL_CONNECTED){

// Serial.println("Connected");

//Preparing HTTP Request to a server

HTTPClient http;

http.begin (serverIpAdress+"API.php?
key="+Node_KEY+"&temp="+temp+"&hum="+hum+"&dust="+dust+"&smoke="+smoke);
//HTTP GET

//start connection and send HTTP header

int httpCode = http.GET();

//httpCode will be negative on error

if (httpCode > 0) {

//HTTP header has been send and Server response header has been handled

//Serial.println( httpCode); //Print out HTTP response code, for a successfully request it is

// file found at server

If (httpCode == HTTP_CODE_OK) {

String payload = http.getString();

Serial.println(payload);

// Deserialize the JSON document

const size_t capacity = JSON_OBJECT_SIZE(3) + JSON_ARRAY_SIZE(2) + 400;

DynamicJsonDocument doc(capacity);

DeserializationError error = deserializeJson(doc, payload);

//Test if parsing succeeds.

 if (error) {

Serial.print(F("deserializeJson() failed: "));

Serial.println(error.c_str());
```

```
return;
}else{
//Deserialization succeded
dataInterval = doc["time"].as<float>();
//update time interval
requestInterval = dataInterval*1000;   }}
} else {
Serial.println( http.errorToString(httpCode).c_str());}
http.end(); //Kill HTTP request to free up resources
}else{
Serial.println("Disconnected");
//Try to reconnect
WiFi_connect();}
//Delay between requests
delay(requestInterval);}
//******************Custom functions*************************************
void WiFi_connect(){
//attempt to connect to WiFi network:
 while (WiFi.status() != WL_CONNECTED) {
 Serial.print(".");
   // Connect to WPA/WPA2 network.
   //wait 5 seconds for connection:
   delay(5000) }}
```

Appendix 3: PHP Codes for the Web- based Application

```
<?php

session_start();

require_once 'class.user.php';

$user_login = new USER();

//redirecting to home page if in session

if($user_login->is_logged_in()!="")

{$user_login->redirect('home.php');}

if(isset($_POST['submitBtn'])){

//Getting Post Data

$userName = trim($_POST['userName']);

//removing spaces

$userName = str_replace(' ', '', $userName);

//password

$pwd = trim($_POST['userPwd']);

//removing spces on password

$pwd = str_replace(' ', '', $pwd);

 //if is seccess, redirect the user to home page

  if($user_login->login($userName,$pwd)) {

 $user_login->redirect('home.php'); }}

?><!doctype html>


< html lang="en">

< head>
```

```
< meta charset="utf-8">

< meta name="viewport" content="width=device-width, initial-scale=1, shrink-to-fit=no">

< meta name="description" content="">

< meta name="author" content="">

<l ink rel="icon" href="">

< title>Home</title>

< !-- Bootstrap core CSS -->

<l ink href="css/bootstrap.min.css" rel="stylesheet">

<style type="text/css">

/*custom style here*/

#head{

font-weight: bold;

font-size: 45px;

color:#11A6D0;}

</style>

</head>

<body style="">

<!-- Nav bar starts here -->

<nav class="navbar " style="background-color: white; border-radius: 0px; border-bottom-style:
solid; border-bottom-color:#11A6D0; border-bottom-width: 2px; ">

 < div class="container-fluid" style="margin-top: 20px; margin-bottom: 10px;" >

<!-- Brand and toggle get grouped for better mobile display –

< div class="navbar-header">
```

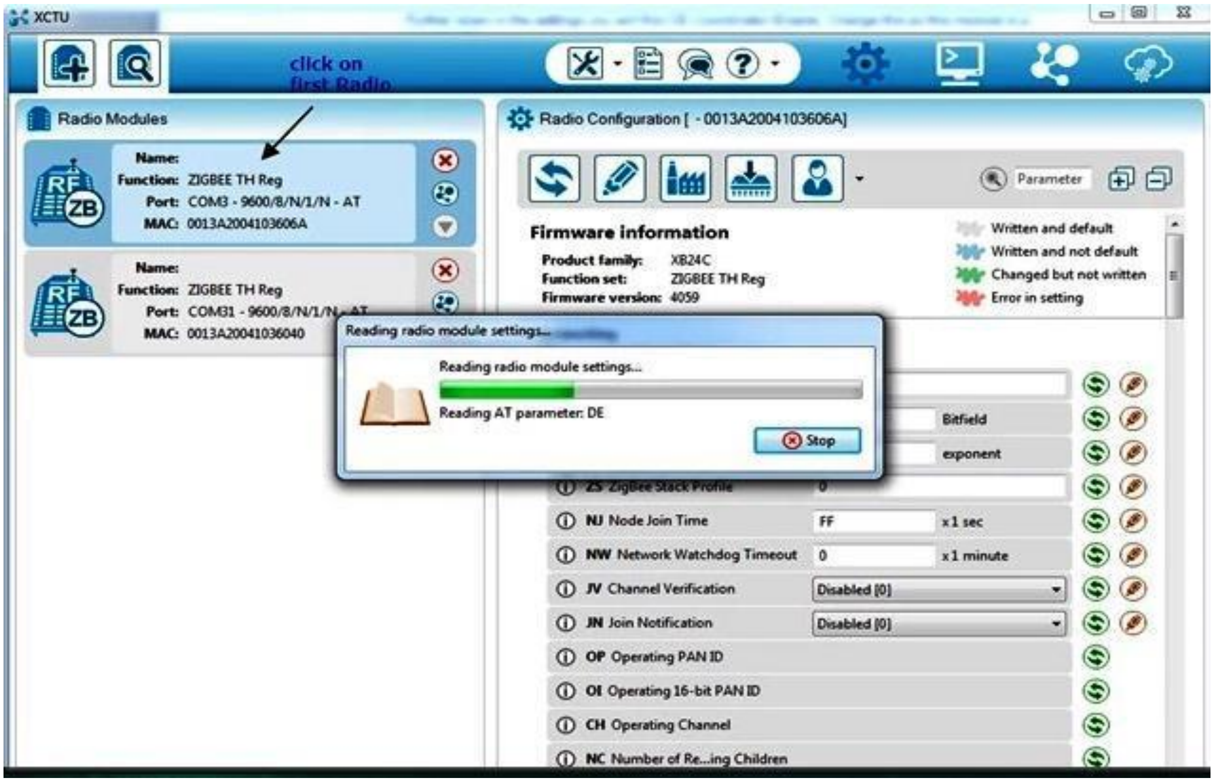
```
<a class="navbar-brand" href="index.php" style="font-size: 20px; font-weight: bold;
color:#11A6D0;">Enviroment Monitoring System</a>

</div>

< !-- Collect the nav links, forms, and other content for toggling -->

< div class="collapse navbar-collapse" id="bs-example-navbar-collapse-1">

<!-- links her -->

<!-- <ul class="nav navbar-nav">

< li ><a href="index.php" style="color: white; font-weight: bold; text-decoration: none;
"><span class="glyphicon glyphicon-home"></span> Home</a></li>

 < li><a href="AboutUs.php" style="color: white; font-weight: bold; margin-left: 20px; text-
decoration: none;"><span class="glyphicon glyphicon-user"></span> About Us</a></li>

< li><a href="contacts.php" style="color: white; font-weight: bold; margin-left: 20px; text-
decoration: none;"><span class="glyphicon glyphicon-pencil"></span> Contact Us</a></

  < /ul >  -->

</div><!-- /.navbar-collapse -->

</nav>

<!-- Nav bar ends here -->

!-- div for contents here -->

<div class="container-fluid" style="min-height: 500px;"

<div style="margin:0 auto; width: 35%; margin-top: 50px; text-align: center;">

</div>

<!-- error here -->

<div style="margin:0 auto; width: 30%; text-align: center;">

<?php
```

```
if(isset($_GET['19d3894f53ce79c3f836f26cf8a3be3b'])) {

echo '<div class="alert alert-warning alert-dismissible" role="alert1">

<button type="button" class="close" data-dismiss="alert1" aria-label="Close"><span aria-hidden="true">&times;</span></button>

<strong>This account has been disabled, please contact the authority.</strong>

</div>';  }

if(isset($_GET['cb5e100e5a9a3e7f6d1fd97512215282'])){

echo '<div class="alert alert-danger alert-dismissible" role="alert2">

<button type="button" class="close" data-dismiss="alert2" aria-label="Close"><span aria-hidden="true">&times;</span></button>

<strong>Incorrect login details, please try again.</strong>

</div>';}

if(isset($_GET['13f173821619ac5abd7d1fedc69f3154'])) {

echo '<div class="alert alert-danger alert-dismissible" role="alert3">

<button type="button" class="close" data-dismiss="alert3" aria-label="Close"><span aria-hidden="true">&times;</span></button>

<strong>Incorrect login details, please try again.</strong>

</div>'; } ?> </div>

<div class="" style="margin:0 auto; border-style: solid; border-width: 1px; border-color: #11A6D0; min-height:350px; width: 30%; margin-top: 50px; margin-bottom: 40px; padding-top: 60px;">

<div style="padding: 20px;">

<form method="post" action="index.php" id="signin_form" autocomplete="off">

<! -- username -->
```

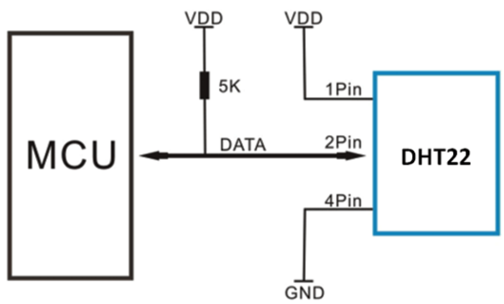
```
<div class="form-group">

<label>User Name</label><br>

<input type="text" class="form-control" style=" border-radius: 0px; height: 40px;"
name="userName" placeholder="User Name..." maxlength="30" required="required">

</div>

<!-- password -->

<div class="form-group">

<label>Password</label><br>

<input type="password" style=" border-radius: 0px; border-radius: 0px; height: 40px;"
class="form-control" name="userPwd" id="userPwd" placeholder="Password" maxlength="30"
required="required">

</div>

<! -- sign in Button -->

<div class="form-group">

<button type="submit" name="submitBtn" class="btn " style="font-weight: bold; border-radius:
0px; border-radius: 0px; height: 55px; width: 100%; font-size: 18px; background-color:
#11A6D0; color: white; margin-top: 30px;"> Sign In</button>

</div

</form>

</div>

</div>

</div>

<! -- Footer -->

<? php include_once'footer.php'; ?>
```

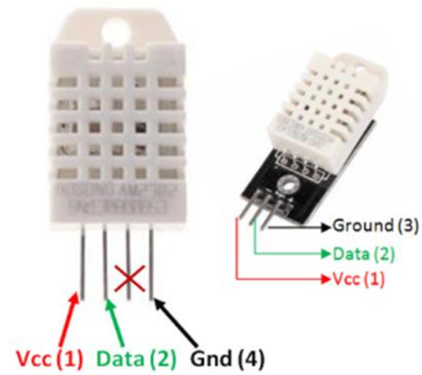
```
<! -- JS scripts here -->

<! -- including jquery library -->

<script type="text/javascript" src="js/jquery-3.1.1.min.js"></script>

<! -- Including the bootstrap library -->

<script type="text/javascript" src="js/bootstrap.min.js"></script>

</body>

</html>
```

Appendix 4: Configuration of XBee antenna Module as ZED and ZCM using X-CTU software



Appendix 5: DHT22 sensor connection diagram Sensor and sensor Pinout

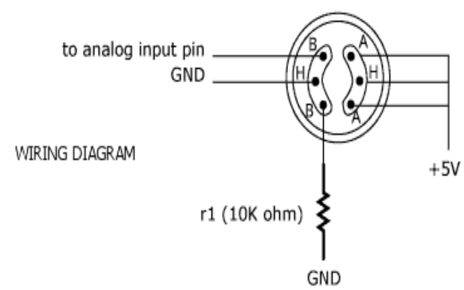




Connection diagram of the sensor

DHT22 Sensor Pinout

Appendix 6: MQ-2 Gas sensor and connection diagram of the sensor



MQ-2 Gas Sensor

Connection diagram of the sensor

RESEARCH OUTPUT