ASHESI UNIVERSITY

Using IoT to assist monitoring of the methane gas extraction at lake Kivu.

CAPSTONE REPORT

B.Sc. Computer Engineering

Oscar Uwayo

2021

ASHESI UNIVERSITY

Using IoT to assist monitoring of the methane gas extraction at lake Kivu.

CAPSTONE PROJECT

Capstone Project submitted to the Department of Engineering, Ashesi University, in partial fulfilment of the requirements for the award of Bachelor of Science degree in Computer Engineering.

Oscar Uwayo

2021

DECLARATION

I hereby declare that this capstone is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere. Candidate's Signature:

Candidate's Name: Oscar Uwayo

Date: April 27, 2021

I hereby declare that preparation and presentation of this capstone were supervised in accordance with the guidelines on supervision of capstone laid down by Ashesi University.

Supervisor's Signature:

……………………………………………………………………………………………

Supervisor's Name:

……………………………………………………………………………………………

Date:

……………………………………………………………………………………………

Acknowledgements

I would like to thank my supervisor Mr. Kofi Adu-Labi for his unparalleled support throughout the project in helping me in designing this paper and sharing his knowledge in this field with me. His continuous support, invaluable insights and encouragement have helped me to write this report.

Abstract

Methane gas is a powerful greenhouse gas with global warming potential. The current techniques being used to monitor the leaks are expensive and likely onerous and demands for trained operators. There are available solutions tried by the space agencies such as National Aeronautics and Space Administration (NASA) and European Space Agency (ESA) using satellites to better understand the distribution of greenhouse gases on regional and global scales. Those are ENVISAT, GOSAT, OCO-2, and the recently launched TROPOMI instrument on the Sentinel 5P satellite, but all these, regardless of the advanced technology associated cannot pinpoint the source of emissions.

In this study, the performance of low-cost Internet of Things (IoT) sensors and isolation forest anomaly detection machine learning technique was implemented. Isolation Forest is one of the outstanding outlier detectors in the real-time DataStream for faulty detection, and money laundering in banking industry. It was tested in this system to improve the accuracy in detecting the methane gas leak. According to the experimental results, the anomaly detection based on isolation forest achieved an excellent performance in terms of accuracy of outlier detection while minimizing the false positives. Decarbonization is an essential component in the climate system, and this plays a key role in reducing methane emissions. Finally, the study presents future research directions to carry out research on the machine learning with Internet of Things (IoT).

Table of Contents

List of figures

List of Tables

List of Abbreviations

REMA- Rwanda Environment Management Authority

CH_4-Methane

IoT-Internet of Things

NASA-National Aeronautics and Space Administration

ESA- European Space Agency

LKMP-Lake Kivu Monitoring Program

GWP- Global Warming Potential

IPCC- Intergovernmental Panel on Climate Change

FPGAs- Field Programmable Gate Array

MWe- Megawatt Electric

Chapter 1: Background and Motivation

In 1984, lake Monoun in Cameroon erupted, and the same incident again happened in 1986 to lake Nyos in the same country due to methane gas, and approximately 1738 people died [1]. A few decades ago, another methane gas shore was found in the Eastern African Rift Zone. It has tons of methane dissolved in the water and is far bigger than the two lakes mentioned. Approximately, lake Kivu contains a thousand times more gas than lake Nyos. Its peculiar characteristics have led to the creation of commercial extraction to produce electricity for the Rwanda national grid. That is the approach the government has taken to mitigate the same incidence as Cameroon. This extraction is being carried out in lake Kivu located on the border between the Republic of Rwanda and the Democratic Republic of Congo.

There are over 60 km^2 of the methane dissolved in the water. However, the extraction also has hazardous risks if it is not properly monitored. Almost two million people in the riparian zone of lake Kivu with more than 400 inhabitants/km^2 [2] might be at risk due to gas flaring, and the methane leak. It also contributes to climate change by releasing millions of tons of CO_2 into the atmosphere. Through various technologies that are being used to extract the gas from its shore, the Rwandan government has also put into place LKMP to ensure its environmentally friendly exploitation. This team mainly focuses on monitoring the risks associated with the extraction process and lake stability or any potential deterioration of the lake's general ecosystem.

These specialists have undergone thorough trainings with regard to methane extraction. However, the outcome remains fragile because the local team is not reinforced at the regulatory level, framework of the institution remains weak, and laws in place for extraction still need to be enforced. The current situation calls for proper and adequate monitoring with strong local capacity

[3]. With the present situation of about two million people living around lake Kivu, an engineering solution is required to monitor the extraction activities while making sure that flaring is under control. Gas flaring is a prohibited industrial practice; it causes the increase of greenhouse gases in the atmosphere and poses a large environmental danger and is also economic waste. On a worldwide scale, over 140 billion cubic meters of natural gas is flared annually, resulting in more than 350 million tons of CO_2 emissions [4].

The aim is to provide an onsite system to qualify and quantify the air contaminants, by providing insights related to the real-time methane gas concentration, humidity and temperature to assist in the monitoring of the methane extraction activities on the border of Rwanda and the Democratic Republic of Congo. It will effectively help to enforce law and eliminate flaring and provide economically profitable use for the natural gas that might be illegally flared. In addition, the objective is to assist these economic activities without jeopardizing the life and health of the residents and employees due to the possible emission of the poisonous and explosive gases in the environment.

Methane gas is also a powerful greenhouse gas with a GWP 84 times that of CO_2 over a 20-year period according to the United Nations IPCC [5]. The current techniques to monitor for methane leaks are expensive and likely onerous. The well-known technology being used is Optical Gas Imaging (OGI) which requires an expensive camera, and trained operator to move through various sites.

On the average, a crew of five operators can visits 5 sites depending on the distance between them, and again the performance depends on operator training and skills.

Recently, another solution was tried by the space agencies such as NASA and the ESA using satellites to better understand the distribution of greenhouse gases on regional and global scales. Satellites such as ENVISAT, GOSAT, OCO-2 and the recently launched TROPOMI instrument on the Sentinel 5P satellite have provided invaluable information on the concentration of greenhouse gases across the globe [6]. But as a drawback, none of these novel techniques has been able to pinpoint the source of emissions.

This project involves using an IoT device for the purpose of monitoring leaks of methane (CH_4), which has explosive properties and is the main constituent of the natural gas. It is to avoid the situation where methane is released into the air which could result in limnic eruptions. The device will be mounted near by the pipes, or at a distance to where people live, in order to effectively monitor the methane concentration, humidity, and temperature in real-time. Also, it will involve development of a dashboard for infographics or data analytics. In case of an abnormal concentration in the air, the alert will be triggered via SMS, email, and alarm to the people in charge of monitoring. The machine learning algorithm will be designed to learn and survey the morphology of microclimate parameters or weather conditions such as air temperature, and humidity. This will also help determine the quality of local atmospheric conditions for the two million people living in the vicinity of the site.

1.1 Background

In Rwanda, the cumulative connectivity to electricity rate is 59.7% of Rwandan households, including 43.8% connected to the national grid and 15.9% accessing through off-grid systems (mainly solar) [7]. The target is that by 2024, 100% of households will have access to electricity.

In attaining the plan, Rwanda has started harvesting the methane and it used to produce electricity. It was however advised that this process would reduce the enormous gas dissolved in the water. The first pilot of the project was between 2008 and 2016, and a plant at industrial scale with a capacity of 26.2 MWe [7]. Later, after the first pilot became successful, the government contracted two more operators to obtain a concession to achieve a production of 150 MWe. One of the contractors is KivuWatt Ltd that owns a concession for 25 years and the plant of 100 MWe, and Symbion owns a concession for 25 years with a plant capacity of 50 MWe [7]. Lake Kivu has much economic value, but there are risks associated with gas extraction if not properly monitored. Hence, advanced technology is needed to assist the LKMP to achieve maximum supervision beyond human error.

1.2 Objectives and motivation

1. To apply engineering concepts to improve the existing work in detecting the methane leakage in real-time.
2. To research and come up with the most promising engineering product.
3. To use engineering and technology to improve the lives of others.
4. To improve on current work in IoT with the application of artificial intelligence.

5. To reduce the human interference in the monitoring system by providing information through infographic dashboard and data analytics.
6. Use machine learning model in surveying the morphology on microclimate parameters.

The motivation of working on this project came when I heard that the contractors of the methane extraction project at Lake Kivu were not local companies or government, my fear was basically that they might focus too much on making profit and expose dangers to the inhabitants close to the lake and as well as 2 million people living nearby. Indeed, I was a little paranoid. I thought about the event in the night of 21 August 1986 when Lake Nyos in Cameroon's North West Province exploded. This eruption sent down to the valley beneath a deadly cloud of carbon dioxide that killed most of the living things and 1,746 men, women, and children, and more than 3,000 cattle. Plus, the aftermath the country has experienced [8], including famine and global warming effects. It made me think of what could be done to mitigate such deadly dangers and protect the environment. After all these horrific events, I found it imperative to research more into the early detection of methane leaks, machine learning progress, and the Internet of Things (IoT). This project will be an enabler to applying engineering skills in solving real-life problems while making an impact in society, specifically the Rwandan community and people on the border of the Democratic Republic of Congo where the project is being carried out.

1.3 Expected outcome.

This project is expected to incorporate IoT and machine learning to assist in monitoring the methane gas extraction activities at lake Kivu. There are sensors for methane, temperature, and humidity, and all are controlled by the microcomputer Raspberry Pi 4 model B. The data is locally analyzed and published and sent to cloud for storage and future use. Hence, data is presented on an infographic dashboard that shows the graph and real-time trends of the methane concentration and other mentioned parameters. Also, a liquid crystal display and buzzer are attached to the system for people operating at the extraction plant. In case of abnormal methane concentration, the buzzer will alert the people in the vicinity to quickly check whatever is causing the changes and evacuate. Figure 1.1 shows the general structure of the system.

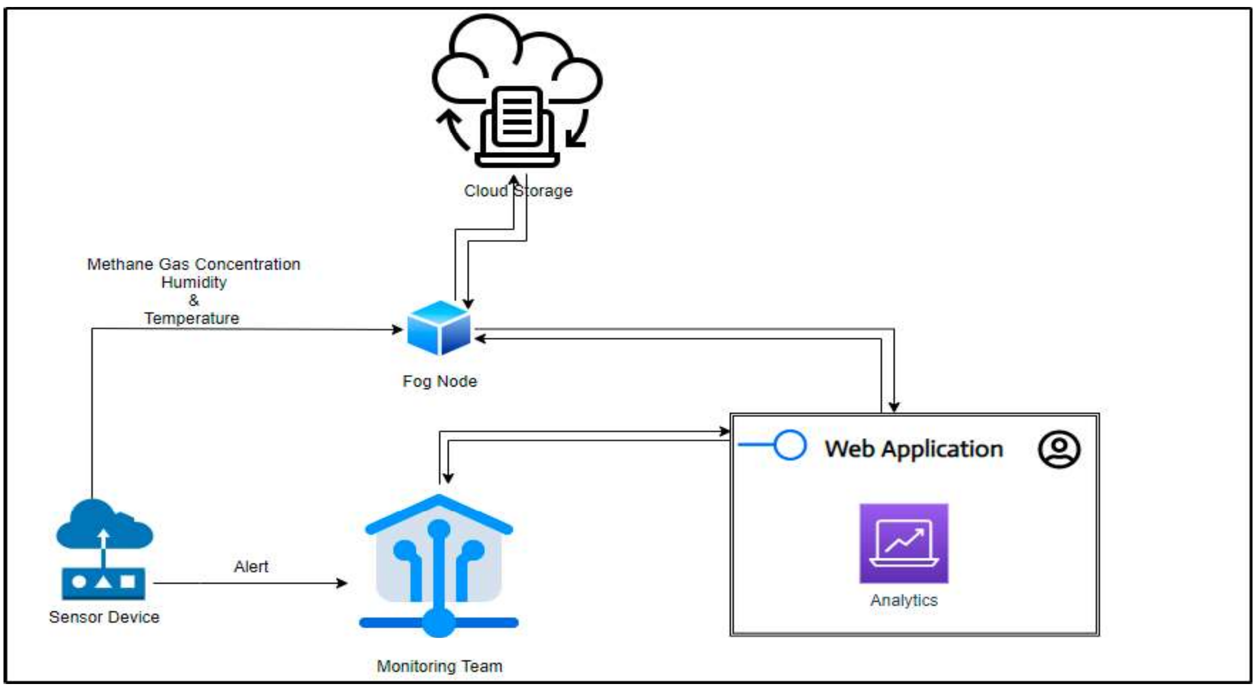


Figure 1.1: High level description of the system

Chapter 2: Literature Review

Gas flaring and methane leakage have been significant problems globally, and many researchers from both academia and industries have been working to address these challenges. The findings addressed in this capstone are not far from what other researchers have found but focuses more on making it accessible at low cost and integrating recent technologies, as well as much emphasis on local context. This chapter combines the reviews on efforts many researchers brought to life with an in-depth analysis of technical aspects, solutions, challenges, and further research recommendations.

2.1 Relevant publications

Many authors, such as Abdullah Erdal Tumer and Mesut Gunduz have worked on modernizing the gas methane leak detection by avoiding the old approach of using cables. After using the wireless sensors, it was stated that the old system of the cabled network has a poor performance of expansion. The cables are poor in aging, wiring resistance and high incidence of failures [9]. In terms of cost, wireless installation minimizes cost and debugging becomes more comfortable in case of incidence, and it is user friendly. The researchers focused solely on the early detection of methane leakage by designing the sensor system for monitoring gas concentrations with gas sensing network technology. This design incorporates two components. The first one is the Micaz as the wireless module, and another one is a gas sensor that uses a pre-calibrated NMG 2611-E13. It has user-friendly software for real-time monitoring from anywhere via an internet web page.

The experiments indicated the system's high accuracy, especially, data communication between the monitor terminal and the methane gas sensor node. Also, it is more flexible and suggests a low-cost installation. Furthermore, these researchers highlighted that the topological change in the wireless movement would be more comfortable, handle disturbances and present data in real-time.

The University of Wollongong published an article on the design, development, and validation of an autonomous gas sensing platform prototype to monitor greenhouse gases such as methane and carbon dioxide [8]. Infrared light sensors were used in a package, and these were an IR gas sensor for CO_2 (Dynament Ltd. IRCEL-CO2), an IR gas sensor for CH_4 (Dynament Ltd. IRCEL-CH4), a humidity sensor (Honeywell HIH-4000-001), and a temperature sensor (Thermometrics DKF103N5). All these sensors were housed in a sample chamber. The researchers focused much on making the communication method efficient by finding the means of transferring the data in different locations irrespective of the range. For example, the communication between the system and a laptop computer, which was for laboratory testing and in-field system management was performed via Bluetooth serial communication [10]. The communication of the data to a remote station was performed via a GSM, and data was statistically represented and sent as SMS to the central station.

Although the four-month validation trial has shown that a monitoring frequency of once per month is inadequate to give an accurate representation of the dynamics of gas production and extraction on a landfill site, it was evident that many events were not recorded which left room for other researchers to explore further the monitoring of the real time trends.

Rebecca Del Papa Moreira Scafutto and Carlos Roberto de Souza Filho from the University of Campinas published an article on detecting methane and heavy hydrocarbon gases in the infrared range using hyperspectral airborne remote sensing. The research aimed at exploring the monitoring tool that could help in the fossil fuel industry and possible ways to improve the estimate of global methane (CH4) budget emissions. The article also looked into ways of reducing production loss and prevention of accidents. Airborne hyperspectral sensors were used to map CH_4 point sources on a local and regional scale. The researchers validated various ranges for detection and quantification of the methane gas. These ranges were short- (SWIR), mid- (MWIR), and longwave (LWIR) infrared.

In the SWIR (1-3 μm), it was shown that the absorption features in this range are weaker in intensity when compared to the features located in longer wavelengths [11]. In this range, interference of atmospheric gases is higher. In the MWIR (3-5 μm), CH_4 has a strong spectral feature located around 3.3 μm [11]. But the detection in this range is complicated because of the mutual contribution of reflectance and emissivity, which makes it tricky to the atmospheric compensation of airborne and limited data availability [11]. Furthermore, the LWIR (7-14 μm) was mentioned as the best due to the ability that allows more robust and sensitive detection of the gas, unlike the other mentioned ranges. In this range, the solar irradiance reflected from the surface can be discarded in the radiative transfer model, and the interference of the reflectance is negligible [11].

The authors concluded that, apart from reflectance, the background temperature, humidity, wind speed, and data acquisition time also influence the detection of CH_4 plumes when processing MWIR data and must be considered prior to data acquisition.

In accordance with the present results about gas monitoring, previous studies by the researchers from the University of Waterloo have demonstrated that the use of natural gas in the future is inevitable, solutions to reduce methane emissions should be developed. In a situation where equipment malfunctions such as cracking or inadequate sealing, release practice like gas flaring, or sometimes by accident should be detected as early as possible. Under this study, the sensors discussed were mainly to allow polluters and policy-enforcers to detect and resolve leaks in a timely manner. Different types of sensors such optical sensor, capacitance-based sensor, calorimetric sensor, resonant sensor, acoustic sensor, semiconducting metal oxide sensor and electrical sensor were studied. Although the most used are optical sensor, calorimetric sensors, pyroelectric sensor, semiconducting metal oxide sensor and electrochemical sensors.

The article proposes optical sensors based on infrared absorption spectroscopy, which disregards the chemical reaction and focuses on physical analysis. Optical sensors have also been chosen because of being immune to electromagnetic interference, which is a challenge to other methane sensors. It has high sensitivity, which is crucial because sensors need to respond as quickly as possible. For future work, the methane sensor should have the ability to operate without oxygen, and being a non-destructive method, and the cost should be taken into consideration [12].

Also, due to the current issue of these sensors being immobile, the authors suggest that for future work, there should be a design of the material that will not consume much power, and mobility should be an aspect of the future evolution. Many sensors are weather unresistant, and as such, those drawbacks were given as recommendations for future research.

This review aimed at analyzing the possible ways IoT systems could achieve the maximum accuracy in monitoring enormous amount of data, it was proposed that machine to machine interactions could help that needed to be smart and have levels of independence [13].

To achieve the goal of making smart devices, the application of machine learning and deep learning were explored as the most recent technologies that have been developed so far. The researchers proposed a better way of maximizing communication latency, whereby designing machine learning model into multiple layers such that the bottom layers are implemented in the edge-device and the top layers in the Cloud [13]. Specifically, fog computing in the end-devices stands out in terms of system security since it is computed locally without transferring the data through other networks. With the Industry 4.0 which is an application area whereby machine learning models, data analytics, and IoT networks are beneficial and helpful tools [13]. Fog-computing has several recommendations due to its privacy, security, time efficiency, and intelligence, which cloud computing could not fulfill in past decades.

[13] has also presented various tools for optimization techniques that could be used to implement machine learning in microcontroller units, field-programmable gate arrays and end devices. It is crucial to decentralize intelligence and make it to the end device.

Chapter 3: Design

To select the right materials, it was imperative to use Pugh matrices, literature review recommendations and drawbacks of each component to assist in the decision making. The series of questions related to environment, performance requirements, and material properties were thoroughly examined. It may be the case therefore that these parameters must be met before any material can be used. Hence, every candidate material was evaluated in order to confirm if the material's properties and their modification by subsequent manufacturing processes have been fully considered. In addition, with prominences of whether shapes and configurations required and at an acceptable price was also a backbone of the analysis [14]. All these questions were answered throughout the material selection process. Table 1 shows key design between the baseline material or datum and the other material based on review and specifications given on the datasheet.

Table 3.1: Design Key for Pugh Matrices

Indicator	Meaning
+	Better than
S	Same
-	Less than

Table 3.2: Pugh Chart for Methane sensors

Selection Criteria	IR sensor	Optical sensor	Calori metric sensor	MQ-4	NGM2611-E13	Electromechan ical sensor
Power Consumpti on	Datum	+	-	+	+	-
Humidity range		-	+	+	+	+
Sampling rate		-	-	-	-	+
Temperatur e range		-	+	+	s	S
Cost		+	+	+	-	-
Lifespan		+	-	+	+	-
Number of pluses		3	3	5	3	2
Number of minuses		3	4	1	2	4

Beyond the mentioned selection criteria, and design analysis between six possible methane sensors, such as IR sensor, optical sensor, calorimetric sensor, MQ-4 sensor, NGM2611-E13 by Figaro and electromechanical sensor. MQ-4 happened to fulfil the requirements, including sensitivity to wide range, ability to only sense methane CH_4 and ignore alcohol and smoke. It is robust and has long lifetime.

Table 3.3: Pugh Chart selection for Humidity and temperature sensor

Selection criteria	DHT11 Sensor	DHT22 sensor	DS18B20	S52100
Accuracy	Datum	+	-	+
Cost		S	+	-
Temperature range		s	-	s
Humidity range		s	-	s
Power Consumption		S	S	-
Response		+	-	-
Installation		+	-	-
Sampling rate		-	-	+
Number of pluses		3	1	2
Number of minuses		1	6	5

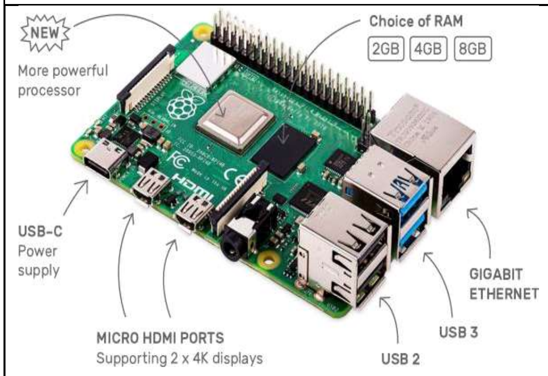
The DHT11 and DHT22 are almost similar, but due to the case where this senor will be used, the DHT22 fits well because of its ability to measure the temperature range of -40°C to +125°C with an accuracy of ± 0.5 degrees. The measuring range for humidity is from 0 to 100%, with an accuracy of 2-5%. These are the two critical specifications that made DTH22 stand out from the DTH11, which can only work with a temperature range of 0-50°C with an accuracy of ± 2 degrees and 20-80% humidity range with an accuracy of ± 5 %. The other sensors did not meet the necessary conditions such as cost, sampling rate, and power consumption.

Table 3.4: Pugh Design for microcontroller

Selection criteria	Atmega8p	Weight	Atmega16 A	PIC 16F628	Atmega32 8-PU
Accuracy	Datum	+	s	s	+
Cost		+	s	-	s
Power consumption		+	+	-	+
Performance		+	-	+	+
Memory capability		+	-	+	+
Peripheral features		+	+	+	S
Number of pluses		6	2	3	4
Number of minuses		0	2	2	0

In selecting the right microcontroller, it turned out that the datum was the best among others. That is due to high performance, its non-volatility and data memory. It can also operate in five different sleep modes: Idle, ADC Noise Reduction, Power-save, Power-down, and Standby and it is mostly used in industrial automation projects. However, this project is time sensitive, and data collected needs to be analyzed in real time, hence Atmega8-p does not have high processing power in terms of handling computational machine intelligence. Table 3.5 shows the specifications and advantages of a microcomputer Raspberry Pi 4 Model B that was used in place of Atmega8-p. It has 64-bit quad-core Cortex-A72 processor, 2GB LPDDR4, Bluetooth 5.2, and 802.11/g/n/wireless to mention a few.

Table 3.5: Reasons for Raspberry Pi

Raspberry Pi microcomputer	Advantage
	• High processing power and storage for edge computing • Friendly firmware • Many interfaces (HDMI, multiple USB, Ethernet, onboard Wi-Fi and Bluetooth etc. • 2.4 GHz and 5.0 GHz IEEE 802.11ac wireless, Bluetooth 5.0, BLE • 5V DC via USB-C connector • Operating temperature 0–50ºC

3.1 System requirement

3.1.1 Functional Requirements

The system must affirm to the following requirements, which clearly describe what the system entails and what it will do.

i. The system should be able to report the trends of atmospheric condition such as temperature, concentration of water vapor present in the air and CH_4 in its surrounding.

ii. The system should be able to send email and message.

iii. The system should be able to differentiate methane and other gases present in its surroundings.

iv. The system should be able record and compute data in the fog, which is a distributed network environment closely associated with cloud computing and the internet of things (IoT)).

v. The system should have ability to develop a pattern of the local atmospheric conditions in real time using a machine learning algorithm.

vi. The system's web application should allow the user to view predictions.

3.1.2 Non-Functional Requirements

These are parameters that define the performance attribute of a software system. Also, these ensure good user experience and ease of operating the software.

i. Fast response to incidents.

ii. Reliability

iii. Instant co mmunication

iv. Cost effective.

v. The system must ensure security.

vi. The system must be responsive and free from errors.

3.2 Use case diagram

The use of case diagram involves a requirements analysis concept, a case of a use of the system/product and describes the system's actions from the point of view of a user. It also tells a story on the sequence of events involving the interaction of a user with the system.

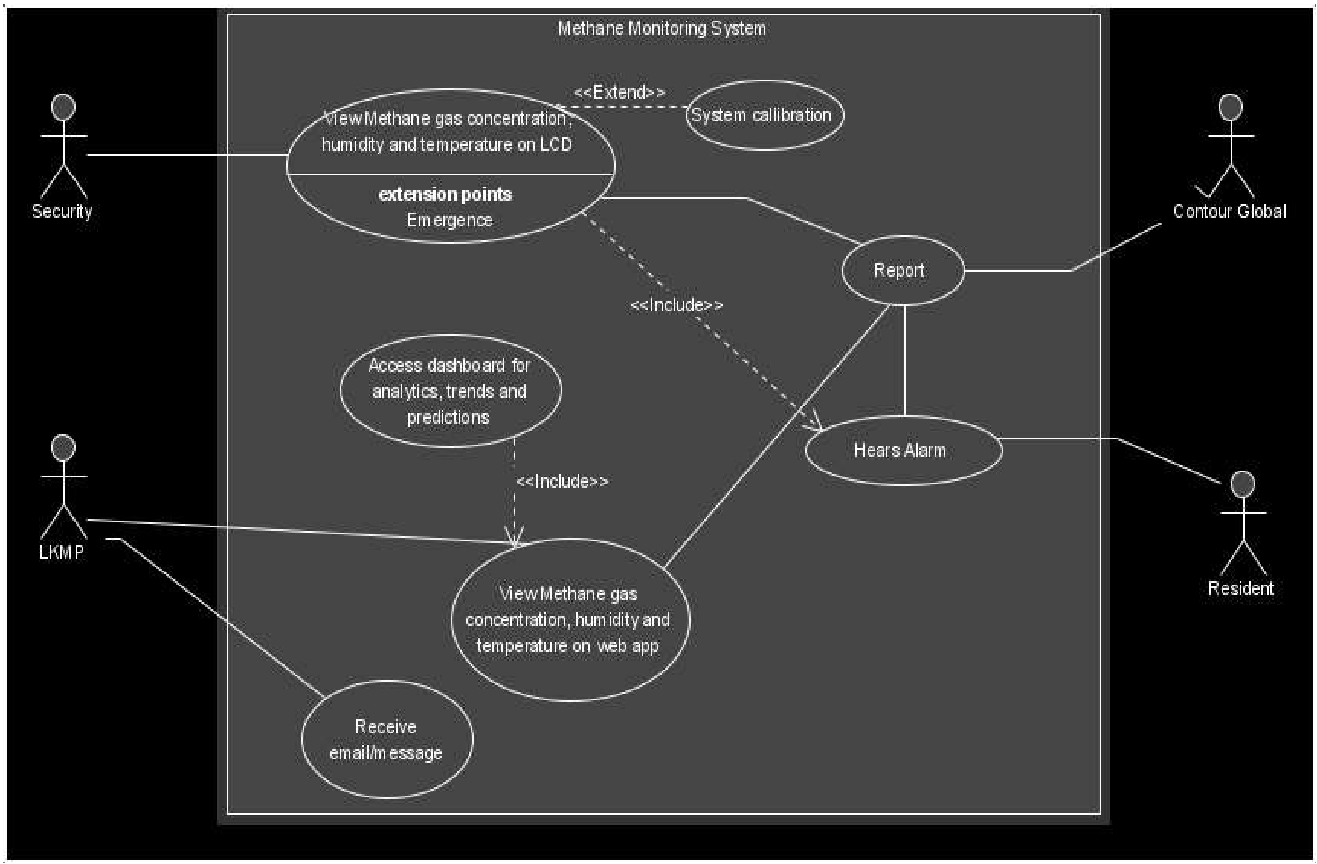


Figure 3.1: Use case diagram

3.3 class diagram

Figure 3.2 shows a class diagram in the form of a unified model language that describes a system's structure by highlighting the attributes, classes and operations, and the relationships among various entries and objects.

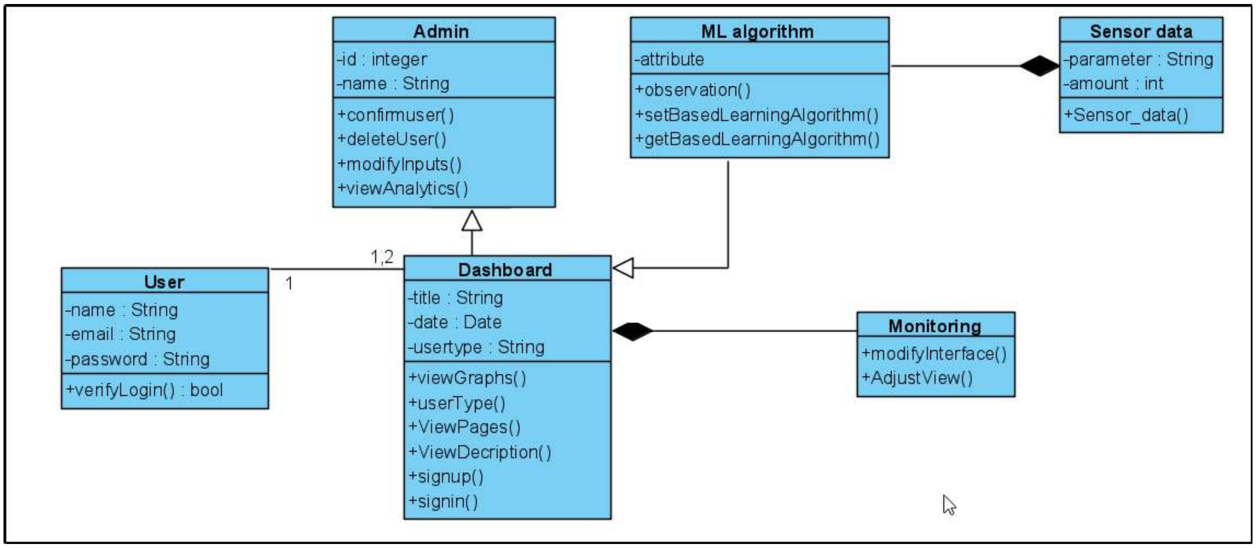


Figure 3.2: Class diagram

3.4 General architecture design

The following figure illustrates the overall functionality of the entire system.

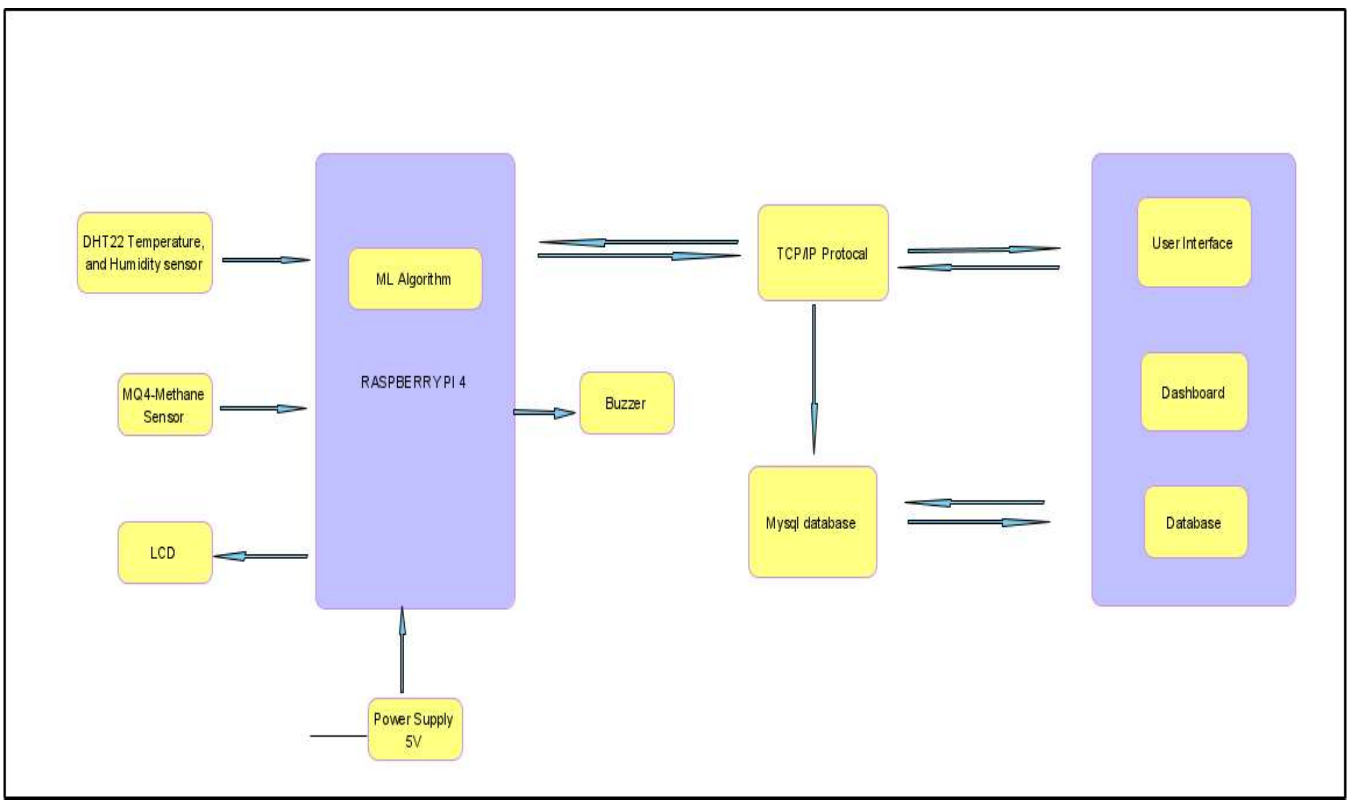


Figure 3.4: Architecture Design

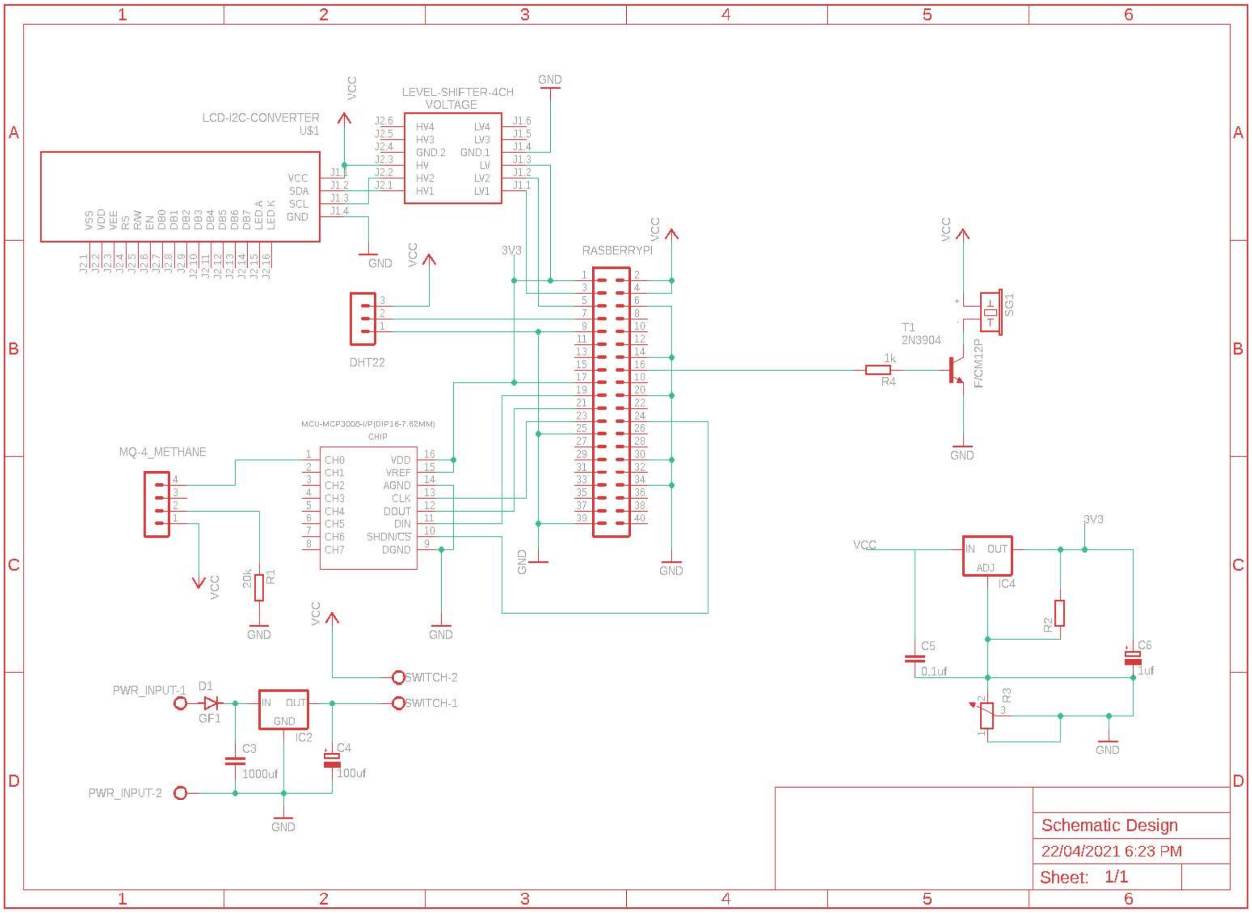


Figure 3.5: Schematic of circuit

3.5 Proposed casing

Figure 3.6 shows a CAD model for a proposed casing of both MQ-4 methane sensor, buzzer, DHT22 humidity, temperature sensor and liquid crystal display that will be attached on the front view of the case.

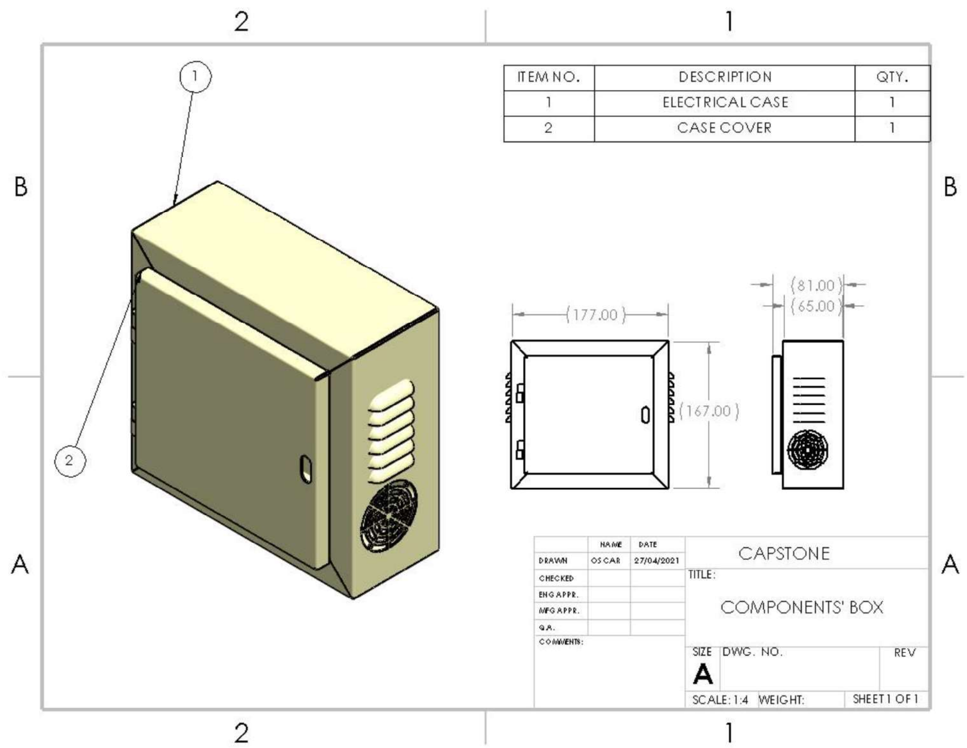


Figure 3.6: Proposed case drawing

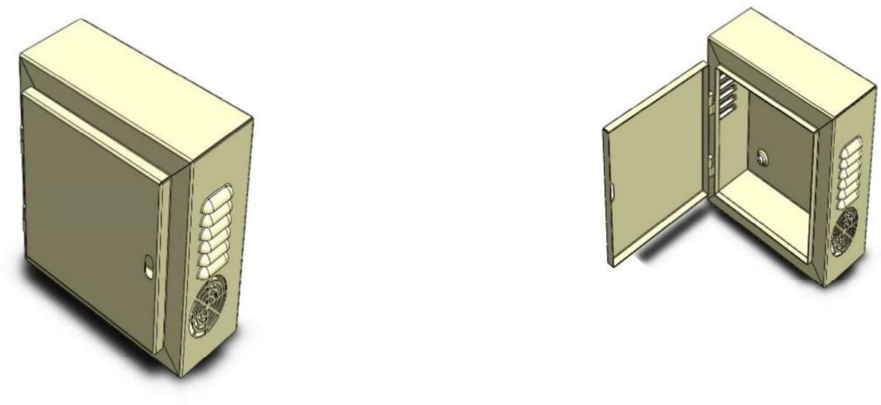


Figure 3.7: Sensors and microcomputer case

Chapter 4: Methodology

4.1 Experimental/Computational Setup

Throughout this project of developing an IoT system to assist in monitoring of the methane gas extraction at lake Kivu, various techniques were carried out. The methodological approach taken in this study is a mixed methodology based on different approaches such as sensors calibration and setup for valid threshold, circuit and printable circuit board implementation, server configuration and web development, machine learning algorithm, system deployment, and system testing.

4.2 Methane gas sensor calibration and setup for valid threshold

First, it is crucial to fully perform the calibration of the methane gas sensor, which is MQ-4, before the utilization to achieve the best accuracy. In a situation where the target gas exists, the sensor's conductivity becomes heavier with the gas concentration rising, and through an electrical circuit, the respective output signal is converted according to the concentration level. The concentration of gas content that the MQ4 sensor can reach is 300-10000ppm [15].

A threshold limit value (TLV) of 1000 ppm is recommended to increase protection use in combination with an auxiliary self-contained breathing apparatus or an emergency escape air cylinder. Exposure to 5000 ppm is immediately dangerous to life and health, and the possibility of explosion if above 5000 ppm [16]. MQ-4 is calibrated to detect 5000 ppm of methane concentration in air and a load resistance of about 20 KΩ or somewhere between 10KΩ to 47KΩ. For accuracy purposes, both temperature and humidity influence should be put into consideration [17].

4.3 Hardware components

The hardware design systematically shows how components are interconnected using a circuit board designed with Eagle CAD software. The system uses Raspberry Pi to gather input from the MQ-4 methane sensor, temperature, and humidity sensor. The buzzer and LCD were interfaced for output purposes. The system is powered using a switching adapter of 12V, which is again converted to 5V to be supplied to the system with the aid of a linear voltage regulator (LM7805). Also, another voltage regulator was used to convert 5V to 3.3V which supplied to the second end of the microcomputer. The I2C LCD display was used to avoid the actual LCD consuming many pins. GPIO pins can be switched (multiplexed) into various other modes backed by dedicated peripheral blocks, including I2C. Raspberry PI GPIO pins can send the output or a signal of 3.3V. However, in using I2C for the LCD, 5V was needed, and to achieve the required voltage, a bidirectional level shifter, as shown in Figure 4.1, was used. The 3.3V is passed through LV pins, and after conversion, it goes through HV pins.

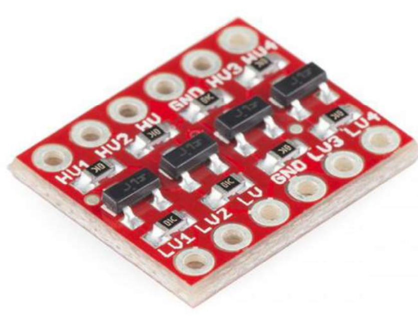


Figure 4.1: Bidirectional Level Shifter

In addition, the case was designed in SolidWorks. Furthermore, The Raspberry Pi computer does not have a way to read analog inputs. There has been a tradeoff between incorporating sensors with only digital output capabilities and those with high sensitivity but analog output. A device such as the MQ-4 methane gas sensor has a high sensitivity to methane, but the manufacturer recommends using its analog output, although it has both analog and digital outputs. It was worth including a microchip MCP 3008, as shown in Figure 4.2, to make the Raspberry Pi analog friendly.

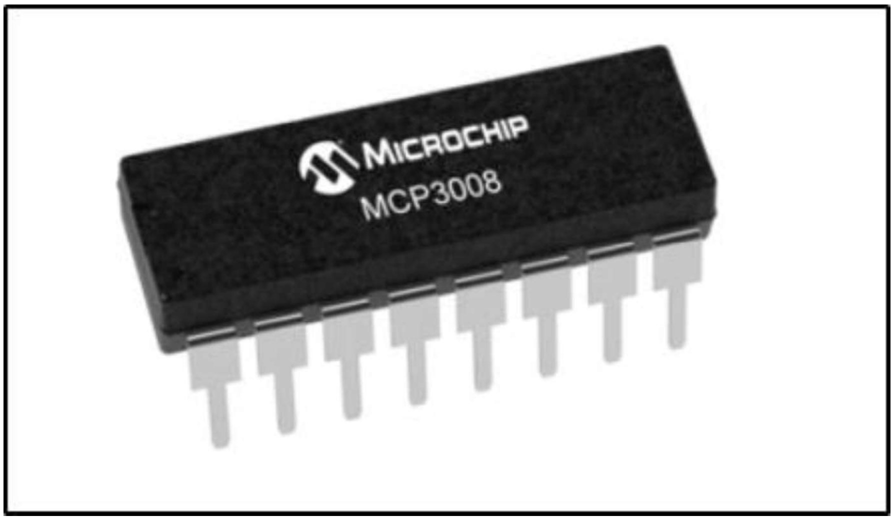


Figure 4.2: MCP 3008

4.4 Software development

This section comprises three principal parts such as Raspberry Pi configuration, Laravel framework for dashboard, and MQTT broker, which mediates communication between the IoT components and the analytics platform.

4.4.1 Microcontroller part

The computational device used in this project is a microcomputer Raspberry Pi 4 Model B to collect the data from MQ-4 and DHT22, and other output devices are programmed using Python programming language which enabled use of various libraries such Adafruit in this project.

In the development process, a local area network was created to assist the communication of microcomputer (Raspberry Pi) and personal computer which had a Xampp server that hosted MySQL database where the data was transferred for further analysis and infographic purposes. Figure 4.3 shows MariaDB granting communication as well as access to the database using the IP address of the Raspberry Pi.

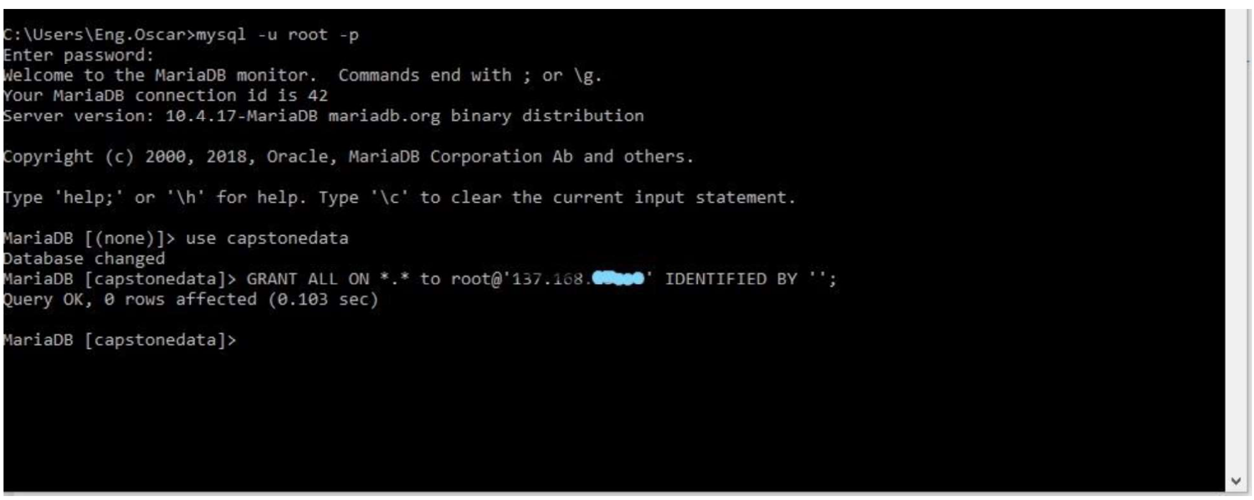


Figure 4.3: Microcomputer and MySQL connection

4.4.2 Laravel framework

Laravel is a web application framework which allows developers to achieve most of the web projects such as routing, authentication, views, sessions, and caching. It was used in the backend of this project. HTML, CSS, and JavaScript specifically Js charts were also used to achieve a friendly and responsive user interface in the frontend. Furthermore, Visual Studio Code was used as a development environment.

4.4.3 MQTT Broker

MQ Telemetry Transport (MQTT) is a lightweight and flexible network protocol. It was originally invented and developed by IBM in the late 1990's, and it was mainly used to link sensors on oil pipelines with satellites, using a messaging protocol that supports asynchronous communication between parties [18]. MQTT was used in this project through publishing and subscription. Figure 4.4 shows the flow of data from fog node to the user interface.

Figure 4.4: MQ Telemetry Transport: Subscriber and publisher

4.5 Machine learning algorithm

The machine learning algorithm in this study aims to contribute to the growing area of research by exploring the use of machine learning at edge/ fog node. A decentralized machine learning algorithm can learn from the experiences, improve its performance, and adapt to changes in the environment. The model helps to improve accuracy of analytics as well as prediction. In addition, methane gas sensors, temperature, and humidity sensor networks leverage the existing machine learning algorithms to provide robust, reliable, and autonomous monitoring. This was achieved using Isolation Forest Anomaly detection with the help of Sklearn.

Given the high number of data coming in real-time to recognize the methane gas leak, the anomaly detection seemed reliable for this specific situation. Anomalies are data patterns that have different data characteristics from normal instances. As shown in Table 4.1, [19] has compared the various algorithm for anomy detection.

Table 4.1: Comparison of anomaly detection techniques

Anomaly detection techniques	Advantage	Disadvantage
One Class Support Vector Machine	It works perfect in finding a separation hyperplane	- Not good for numerical situations - Not good fit for large datasets
K- Means	- Low complexity - Easy to implement	- Necessity of specifying number of centroids. - Sensitive to previous assignments - Sensitive to noise and outlier data points - It can mislead since many clusters have equal number of observations
Local Outlier Factor	- It considers a point as an outlier at a small distance, which make	- Not scalable

	it good for local anomaly detection.	- Bad performance on indirect neighborhood - Highly dependent on local anomalies

While these techniques were explored, in this project, Isolation Forest anomaly detection was used. It has distinguished itself in the banking industry, where it is used to detect fraudulent transactions or money laundering. This is a fast executable model with low memory requirements. It has a direct result of building partial models and requiring only a significantly small sample size compared to the given training set. This capability is unparalleled in the domain of anomaly detection. Besides, the iForest algorithm meets the requirements of real-time, continuous detection without any supervision. Isolation Forest is an accurate and efficient anomaly detector, especially for large databases, making it desirable for real-life applications.

The real-time monitoring of methane gas or any other hazardous situation demands data analytics or a live corresponding tool for evaluation [20]. A proposed method [21] takes advantage of two of anomalies' quantitative properties:

i. They are the minority consisting of fewer instances and

ii. They have attribute-values that are very different from those of normal instances.

The following were reasons to why isolation Forest was chosen from other anomaly detection techniques.

- Automation
- Makes no assumptions about what an anomaly looks like.
- Does not require labeled data

- Has the ability to provide a continuous anomaly score, which eventually ensures safety.

When new data arrives, the isolation forest algorithm automatically assign a score to the observation that will classify whether it is safe or not safe, and as well as minimizing the number of false positives.

4.6 System deployment

The web application is (will be) hosted on 000webhost. Prior to hosting, most of the computations are done in fog nodes since the Internet of Things requires latency-aware computation for real-time application situations. The novel paradigm of fog computing through which the cloud platform model is extended by using network edges to back up computing resources which also provides data storage and application services [22]. Over time, data degenerated from IoT devices are processed in cloud infrastructure, but it has not been efficient enough especially for time-sensitive applications. Cloud infrastructure does not fit well with latency and time-sensitive service requests because of being some distance away from the end device. It increases network congestion, delay, and degrades the quality of service [23]. Also, processing enormous data in the cloud requires more time and effort, which consumes more energy. Based on the investigation, fog computing distinguishes itself as a more promising and well-generalized computing paradigm in the context of Internet of Things that provide services closer to the network.

Chapter 5: Results

5.1 Overview

This chapter highlights the outcome of the entire system. Here, subsystems were tested before they were all merged to function as a single system.

5.2 Detection of methane and microclimate parameters

The sensors mentioned in the previous chapters were connected, as shown in Figure 5.1. The 9-12 V power supply could not be obtained; therefore, an alternative breadboard power supply was used to power MQ-4 to avoid overloading the microcomputer, and a USB-c of 5V was also used to power the Raspberry Pi. Since the output of MQ-4 can be more than the 3.3V the raspberry pi can withstand, the level shifters were used to avoid overloading to the circuit.

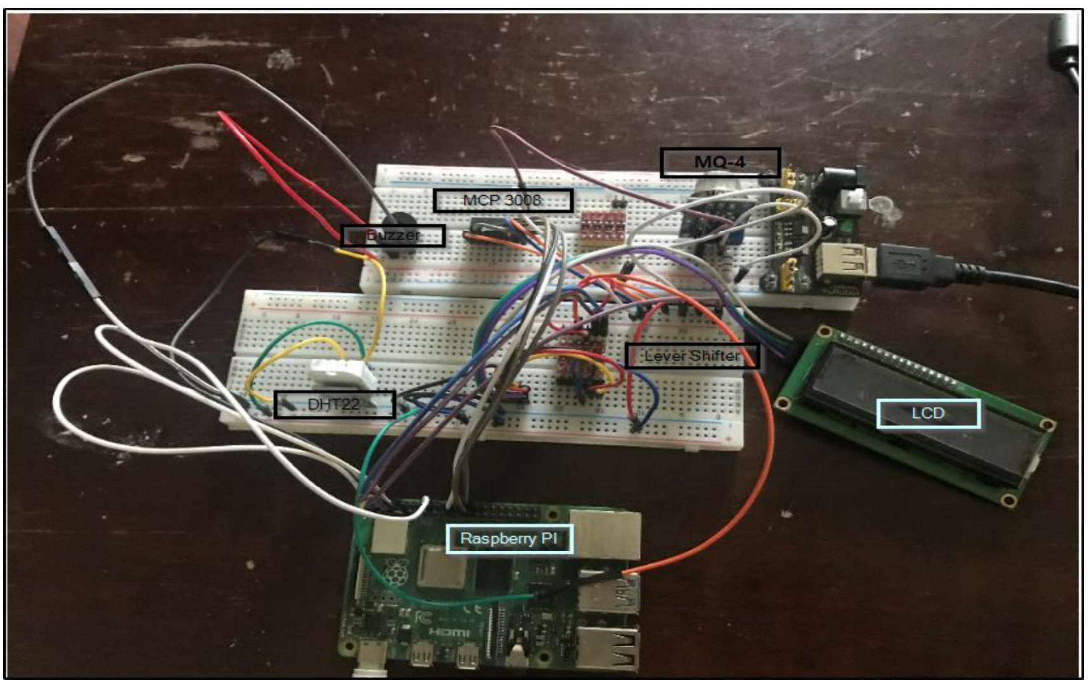


Figure 5.1: Hardware experiment setup

Figure 5.2 shows the sensor reading, and time, specifically MQ-4 methane gas sensor.

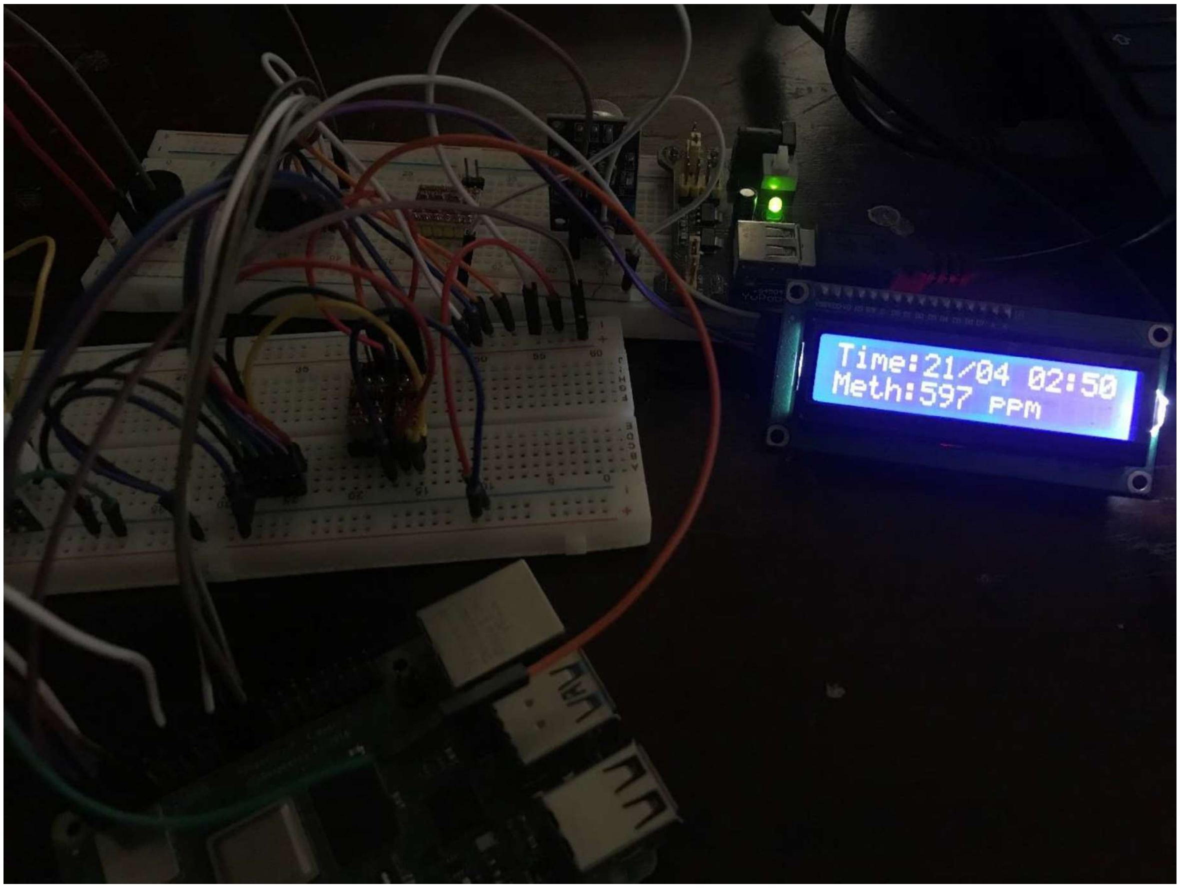


Figure 5.2 Hardware Testing

After having done the setup, the data, including temperature, humidity and methane gas as shown in Figure 5.3 were collected every 5 seconds. Unfortunately, the methane gas reading is not 100% methane rather a few contaminants that are available in air, and that is due to the unavailability of methane on campus. The left part of Figure 5.3 shows the values being displayed in real time. Figure 5.4 shows how Realtime data is saved into the MySQL database. Here the only temperature table is shown but also other tables such temperature and humidity are present.

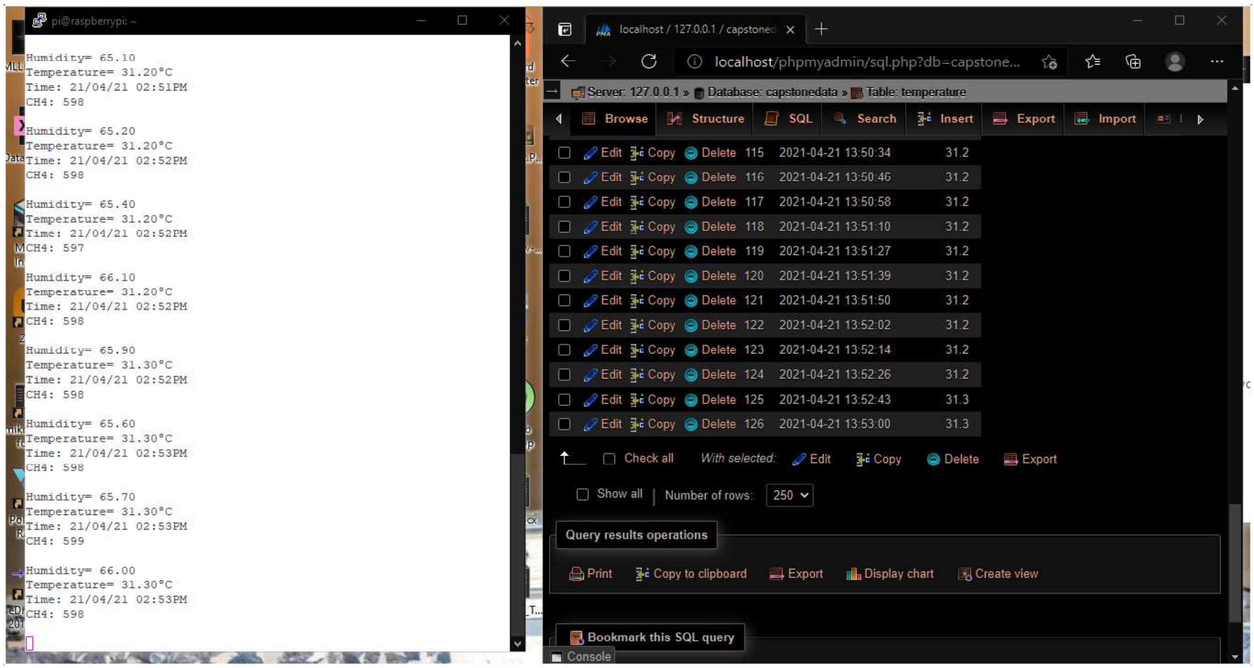


Figure 5.3: Testing sensors

5.3 Dashboard and web application

As part of the design, the web application was first tested to ensure that users access unrestricted information. This was achieved by assigning restrictions to different types of users. Figure 5.3 shows the unit test that was performed on the visibility of pages of other users using assertions.

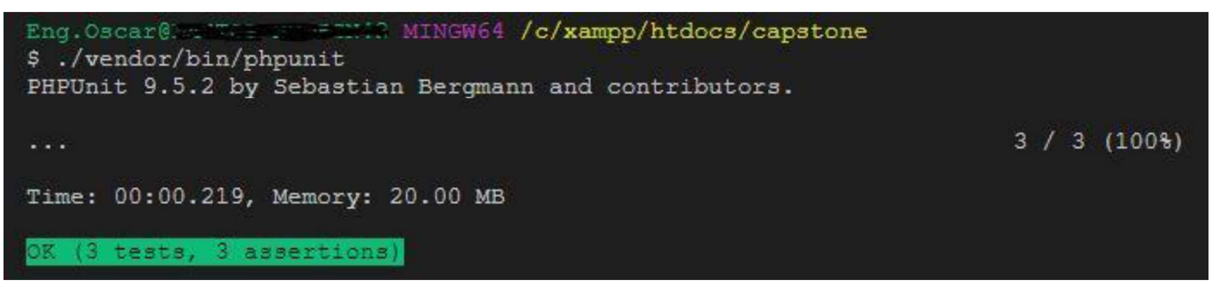


Figure 5.4: Unit tests

A database that that is accessible via a local area network is connected is such a way that gives access to only authorized users. Restrictions were fully set to avoid any attack or fraudulent release of faulty data. Tables are arranged as shown in Figure 5.4.

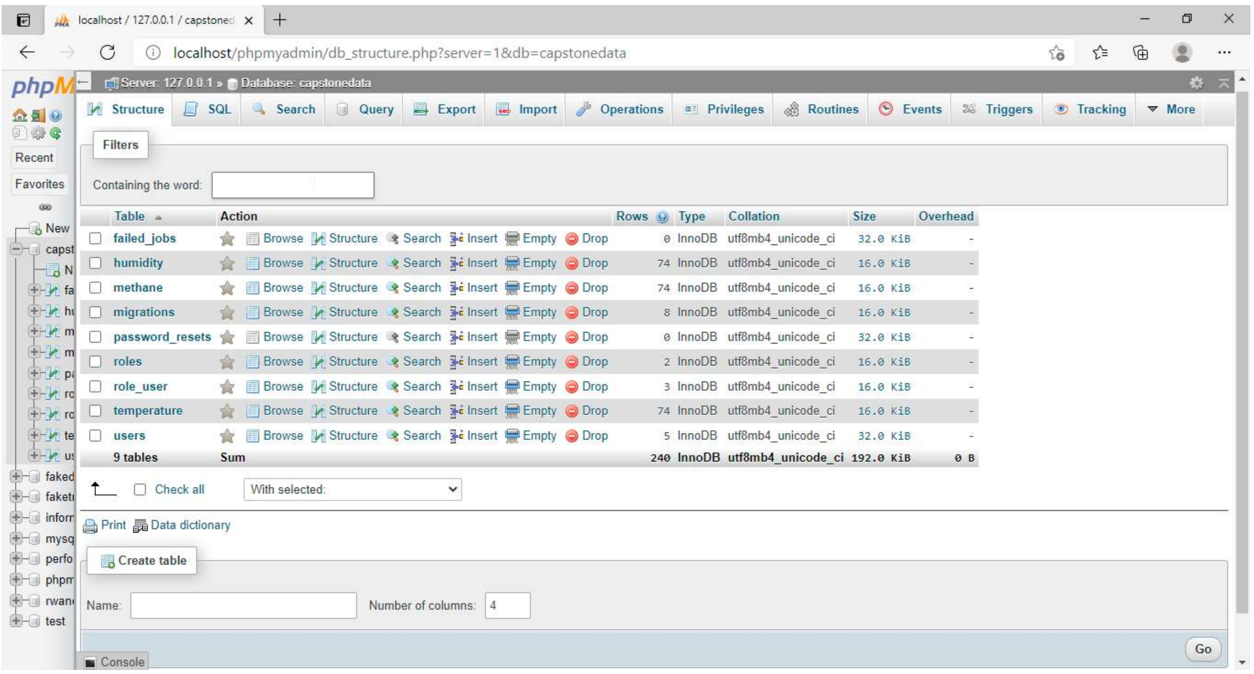


Figure 5.4: Database

Figure 5.5 and Figure 5.6 show the interface every user sees when trying to register and login on the web application respectively.

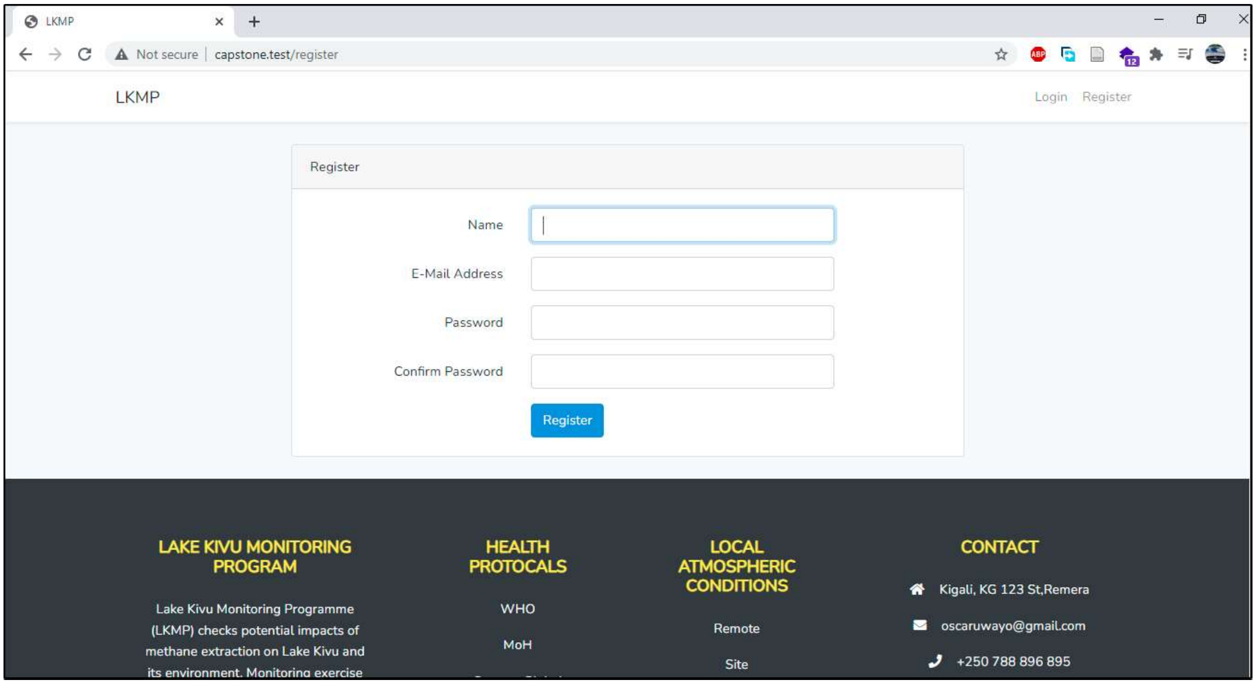


Figure 5.5: User registration

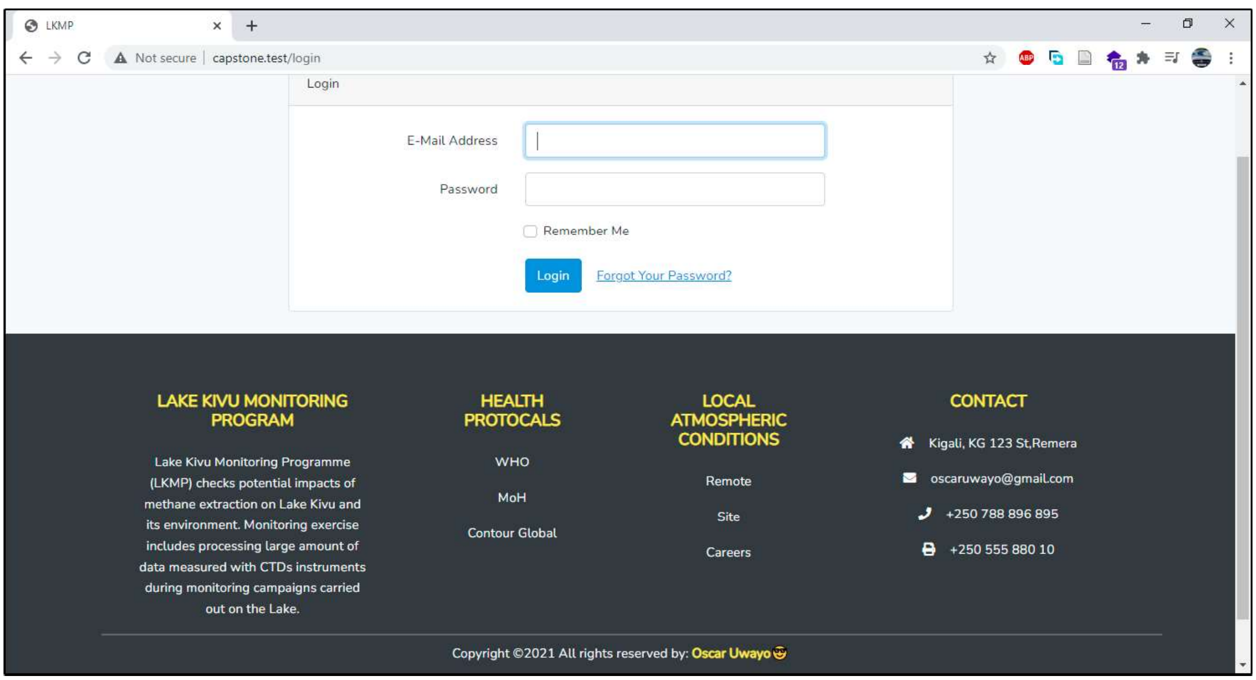


Figure 5.6: User login

Figure 5.7 shows a generic user view which will soon show infographics. On another hand, Figure 5.8 shows the admin views which has permission of assigning privileges through the edit button.

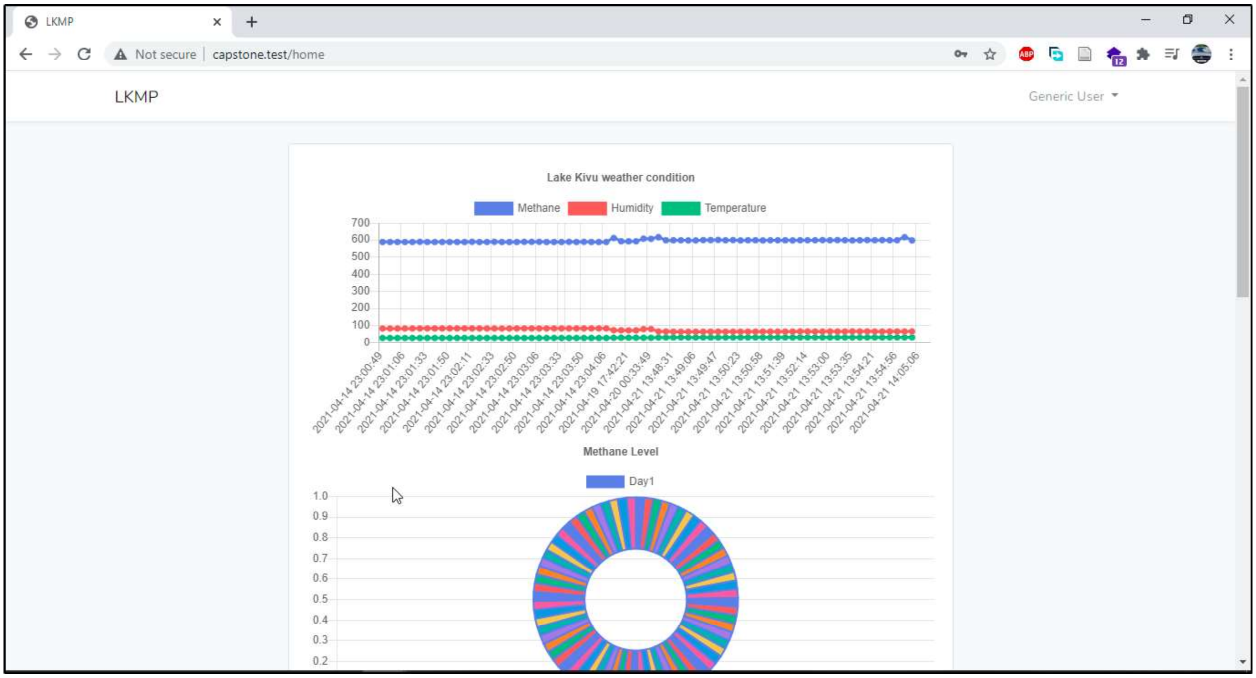


Figure 5.7: Generic user interface

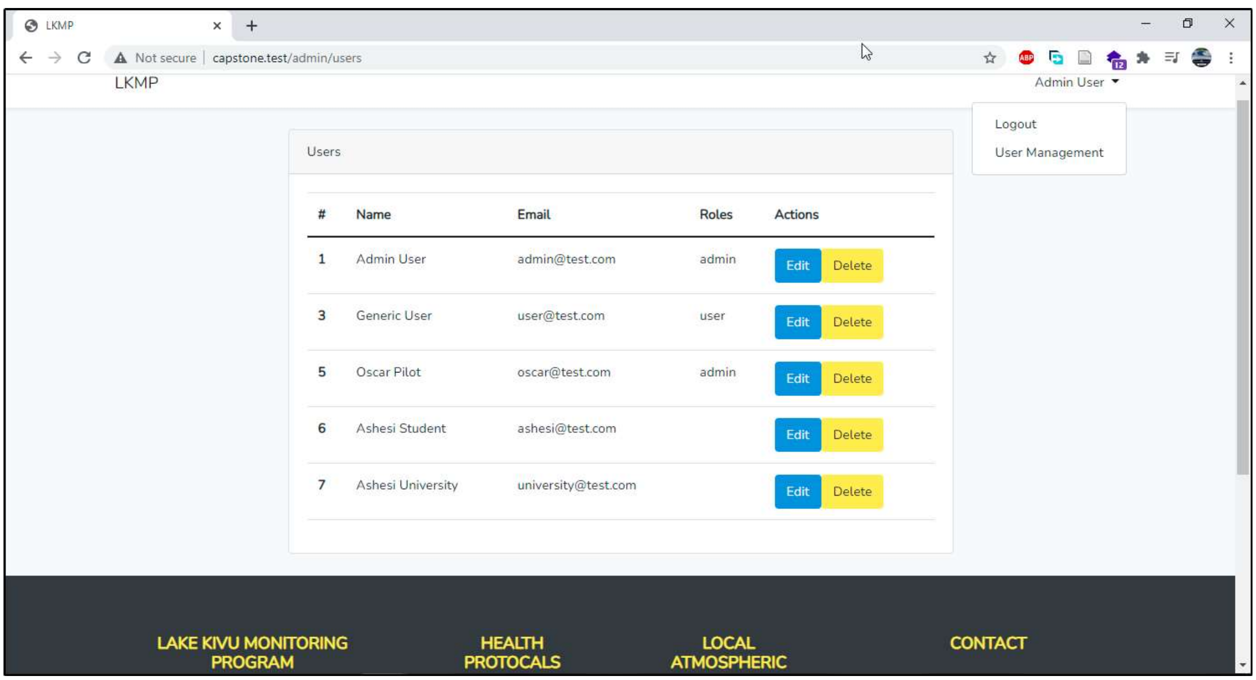


Figure 5.8: Admin user management page

5.4 Machine learning model

Machine learning at the edge device indicates progress in technology and a need to understand the various perceptions of machine intelligence that exist among others. Specifically, in this project, anomaly detection with isolation forest was tried using sklearn [24]. The data used to train the model was generated locally by MQ-4 and DHT22. With that regard, the actual data will be tested when received. Figure 5.9 shows the dataset size.

```
[24] df = pd.read_csv('Data.csv')
     df.head()
```

	Time	Temperature	Humidity	Methane
0	09/04/21 03:42AM	28.8	82.8	611
1	09/04/21 03:42AM	28.8	82.8	593
2	09/04/21 03:42AM	28.8	82.9	593
3	09/04/21 03:42AM	28.8	82.9	592
4	09/04/21 03:42AM	28.8	82.9	593

```
df.shape
(1408, 4)
```

Figure 5.9: Dataset shape

Implementation was made by using sklearn library [24]. By setting a c of 5% in Figure 5.10, which is a score that is used to classify what point is anomalous. Apart from that, this is also a parameter that the algorithm is quite sensitive to; it refers to the expected proportion of outliers in the data set. Hence, 95% of the dataset is identified as normal data points as shown in Figure 5.11.

```
model=IsolationForest(contamination=0.050)
model.fit(mdata)
prediction=model.predict(mdata) #predict offerd by numpy.
#prediction
```

Figure 5.10: Contamination Level

```
[53] (prediction>0).mean() #Ones show normal points, while negavives show abnomal points.

     0.9502840909090909
```

Figure 5.11: Accuracy of the model

Anomaly detection is one of the most popular machine learning techniques. In this situation, it is entirely unsupervised, meaning it does not depend on any manual labeling. Mainly, it is based on the fact that most of the incoming data are normal, and some small percentage would be anomalous.

These anomalous are categorized from those instances of data that do not satisfy the defined normal condition. With no personal involvement, isolation forest identifies anomalies by separating outliers in data. After learning the dynamics of the data generated by MQ-4, it was essential to use isolation forest because other anomaly detection techniques are not optimized to detect anomalies. They are excellent in determining the normal instances. Hence, this would result in a myriad of false positives and detect just a few anomalies. This was the critical reason for using isolation forest.

Chapter 6: Conclusion & Recommendations

6.1 Discussion

This final chapter draws upon the entire project, and it sums up the various theoretical and empirical strands to affirm the objectives mentioned, such as improving the existing work in detecting the methane leakage in real-time, reducing human interference in monitoring process, and using machine learning in surveying the morphology on microclimate parameters, to mention a few. It includes a discussion of the implication of the findings to future research into this area, and areas for further research are identified. In this report, the IoT data analytics using a machine learning algorithm was successfully designed. In addition, fog computing was done to reduce the traffic and latency issue identified when using the cloud as a computational node.

6.2 Limitations

Throughout the execution of this project, various limitations were encountered, and some hindered the completion as well as meeting the stated objectives. The methane gas sensor was tested in an environment which had no methane resent. It was basically tested on the air. The data related to temperature and humidity used was collected in another location which was not at Lake Kivu. Unfortunately, the Rwanda Environment Management Authority does not yet have air quality data around gas extraction plants. This could have helped to design and train a machine learning algorithm and set an online outburst early warning system on the actual data from the site. Every subsystem function as expected, but these systems were not tested on the actual industrial settings. But later, it will be updated with the real data.

6.3 Future work

The system should also be implemented on a PCB, and this was not done because of the cost and time it could take to arrive due to the pandemic. Also, in a situation where there is a need for more than one device at the methane extraction plant, the Raspberry Pi should be used as a gateway for multiple devices. It means other devices can use microprocessors such as AtMega8-p and Wi-Fi module that sends data to a fog node where the Raspberry Pi will be handling them before reaching the cloud. In addition, there should be further research on machine learning at edge device without having to necessarily use a microcomputer. In the future, the researchers also should focus intensely on proper ML models to deal with application specifics and for data analytics.

Although this capstone focused on ensuring the safety of the methane gas extraction processes, by involving stakeholders such as Lake Kivu Monitoring Program and Rwanda Environment Management Authority, the following recommendations were given.

The Rwanda Environment Management Authority recommended through the Lake Kivu Monitoring Program that in order to fully monitor the lake conditions, the system should focus on water quality parameters. The methane gas is dissolved in deep water levels, and a reliable warning system would be based on real-time monitoring of the vertical profiles to about 0- 500 m depth. These parameters are water pressure and concentration of gases such as methane gas and CO_2 and proxy.

References

[1] M. Doevenspeck, “Lake Kivu’s Methane Gas: Natural Risk, or Source of Energy and Political Security?” *Africa Spectrum*, vol. 42, no. 1, pp. 95–110, 2007.

[2] R. Hecky, “Why exploring for oil under Lake Kivu carries unique and huge risks,” *The Conversation*. http://theconversation.com/why-exploring-for-oil-under-lake-kivu-carries-unique-and-huge-risks-78107 (accessed Mar. 18, 2021).

[3] N. Pasche *et al.*, “Building Local Capacities to Monitor Methane Extraction in Lake Kivu,” in *Sustainable Access to Energy in the Global South*, Cham, 2015, pp. 235–244, doi: 10.1007/978-3-319-20209-9_21.

[4] “Zero Routine Flaring by 2030,” *World Bank*. https://www.worldbank.org/en/programs/zero-routine-flaring-by-2030 (accessed Mar. 18, 2021).

[5] J.-F. Gauthier, “MONITORING METHANE EMISSIONS AT INDIVIDUAL OIL AND GAS SITES WITH SATELLITES: A NEW TOOL AT THE DAWN OF GLOBAL TRANSPARENCY,” in *IGARSS 2020 - 2020 IEEE International Geoscience and Remote Sensing Symposium*, Sep. 2020, pp. 3813–3816, doi: 10.1109/IGARSS39084.2020.9323989.

[6] G. Myhre and al, "Anthropogenic and Natural Radiative Forcing. In: Climate Change 2013: The Physical Science Basis. Contribution of Working Group I to the Fifth Assessment," Cambridge University Press, Cambridge, United Kingdom and New York, USA, 2013.

[7] “Access.” https://www.reg.rw/what-we-do/access/ (accessed Mar. 18, 2021).

[8] E. Shanklin, “Beautiful Deadly Lake Nyos: The Explosion and its Aftermath,” *Anthropology Today*, vol. 4, no. 1, pp. 12–14, 1988, doi: 10.2307/3032873.

[9] A. Tümer and M. Gündüz, “Design of a methane monitoring system based on wireless sensor networks,” *Scientific Research and Essays*, vol. 5, Apr. 2010.

[10] S. Beirne *et al.*, “Autonomous greenhouse gas measurement system for analysis of gas migration on landfill sites,” in *2010 IEEE Sensors Applications Symposium (SAS)*, Feb. 2010, pp. 143–148, doi: 10.1109/SAS.2010.5439422.

[11] R. D. P. M. Scafutto and C. R. de S. Filho, “Detection of Methane and Heavy Hydrocarbon Gases in the Infrared Range Using Hyperspectral Airborne Remote Sensing: an Overview,” in *IGARSS 2019 - 2019 IEEE International Geoscience and Remote Sensing Symposium*, Jul. 2019, pp. 5780–5783, doi: 10.1109/IGARSS.2019.8900556.

[12] T. Aldhafeeri, M.-K. Tran, R. Vrolyk, M. Pope, and M. Fowler, “A Review of Methane Gas Detection Sensors: Recent Developments and Future Perspectives,” *Inventions*, vol. 5, no. 3, Art. no. 3, Sep. 2020, doi: 10.3390/inventions5030028.

[13] S. Branco, A.G. Freeing and J Cabral, “Machine Learning in Resource-Scarce Embedded Systems, FPGAs, and End-Devices: A Survey,” *MDPI Electronics*, 8,1289, pp. 20-36, 2019.

[14] G. Dieter and L. Schmidt, “Decision Making and Concept Selection,” in Engineering Design, 5th ed., New York: McGraw-Hill, 2009, pp. 279-298.

[15] R. N. Naik, P. S. N. Reddy, S. N. Kishore and K. T. K. Reddy, Arduino based LPG gas monitoring & automatic cylinder booking with alert system, IOSR Journal of Electronics and Communication Engineering., vol.11, no.4, pp.6-12, 2016.

[16] “New Jersey Health Department”, Hazardous substance fact sheet, Methane,2016. [online]. Available: https://nj.gov/health/eoh/rtkweb/documents/fs/1202.pdf. [Accessed March 3, 2021]

[17] Hanwei Electronics," MQ-4 "Technical data, MQ-4 Gas sensor.

[18] "What is MQTT? Why use MQTT?" *IBM Developer*, Jan. 07, 2020. https://developer.ibm.com/technologies/messaging/articles/iot-mqtt-why-good-for-iot/ (accessed Mar. 18, 2021).

[19] S. Ounacer, H. A. E. Bour, Y. Oubrahim, M. Y. Ghoumari, and M. Azzouazi, "Using isolation forest in anomaly detection: The case of credit card transactions," *Periodicals of Engineering and Natural Sciences (PEN)*, vol. 6, no. 2, Art. no. 2, Nov. 2018, doi: 10.21533/pen. v6i2.533.

[20] S. Ahmad, A. Lavin, S. Purdy, and Z. Agha, "Unsupervised real-time anomaly detection for streaming data," _Neurocomputing_, vol. 262, pp. 134–147, Nov. 2017, doi: [10.1016/j.neucom.2017.04.070] (https://doi.org/10.1016/j.neucom.2017.04.070).

[21] F. T. Liu, K. Ting, and Z.-H. Zhou, "Isolation Forest," Jan. 2009, pp. 413–422, doi: [10.1109/ICDM.2008.17] (https://doi.org/10.1109/ICDM.2008.17).

[22] S. Khan, S. Parkinson, and Y. Qin, ''Fog computing security: A review of current applications and security solutions,'' J. Cloud Comput., Adv., Syst. Appl., vol. 6, no. 1, p. 19, 2017

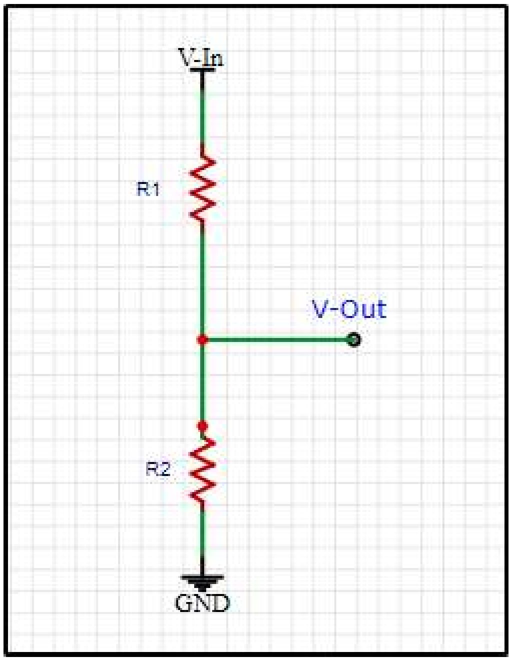
[23] R. K. Naha *et al.*, "Fog Computing: Survey of Trends, Architectures, Requirements, and Research Directions," *IEEE Access*, vol. 6, pp. 47980–48009, 2018, doi: 10.1109/ACCESS.2018.2866491.

[24] "sklearn.ensemble.IsolationForest — scikit-learn 0.24.1 documentation." https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.IsolationForest.html (accessed Apr. 25, 2021).

Appendix

A. This part shows the calculations made to design a voltage regulator.

Voltage regulator: LM317



$V_{R2}=\frac{VIN(R2)}{R1+R2}$

$R_2=\frac{VOUT(R1)}{VIN-VOUT}$

Since, V-In=5v, desired V-out=3.3v.

Let R1=10kΩ.

Then, $R_2=\frac{3.3(10k)}{5-3.3}=12.2k\Omega$

B. A snippet of the Python code implemented on raspberry pi to correct data from MQ-4, DHT22 using Adafruit library.

A: > FinalYear > Capstone > Code > Capstone_scripts.py

```python
import csv
import sys
# Import SPI library (for hardware SPI) and MCP3008 library.
sys.path.append("/usr/local/lib/python2.7/dist-packages")

#Configurations for LCD Display: Using library
sys.path.append("/home/pi/lcd")
import drivers
display = drivers.Lcd()

#For the DH22 : Using Aafruit Libary:
from time import strftime
import time
import adafruit_dht
from board import *
import Adafruit_GPIO.SPI as SPI
import Adafruit_MCP3008
# Hardware SPI configuration:
SPI_PORT   = 0
SPI_DEVICE = 0

#Connecting the database:

import mysql.connector
mydb = mysql.connector.connect(
  host="192.168[redacted],          #localhost=ip address of PC
  user="root",
  passwd="",
```

```
    database="capstonedata",
    port="3306"
)
mycursor = mydb.cursor()


#For the Temp and Hum sensor
SENSOR_PIN =4
mcp = Adafruit_MCP3008.MCP3008(spi=SPI.SpiDev(SPI_PORT, SPI_DEVICE))
dht22 = adafruit_dht.DHT22(SENSOR_PIN, use_pulseio=False)

with open("/home/pi/Data.csv", "a") as log:
        while True:
                try:
                        value = mcp.read_adc_difference(0)
                        date_t=strftime("%d/%m/%y %I:%M%p")
                        val=strftime("%d/%m %I:%M")
                        temperature = dht22.temperature
                        humidity = dht22.humidity

                        print(f"Humidity= {humidity:.2f}")
                        print(f"Temperature= {temperature:.2f}°C")
                        print("Time: "+date_t)
                        print("CH4:",value)
                        print('   ')
                        writer = csv.writer(log)
                        writer.writerow([date_t,temperature,humidity,value])
```

```
                        #tem=temperature
                        sqlt =("INSERT INTO capstonedata.temperature (temperature) VALUES ({})".format(tempe
                        #valt = (temperature)
                        mycursor.execute(sqlt)
 #Humidity:
                        sqlh =("INSERT INTO capstonedata.humidity (humidity) VALUES ({})".format(humidity))
                        #valh = (humidity)
                        mycursor.execute(sqlh)
#Methane:
                        sqlm = ("INSERT INTO capstonedata.methane (methane) VALUES ({})".format(value))
                        #valm = (methane)
                        mycursor.execute(sqlm)
#COMMIT
                        mydb.commit()
#Display on the Liquid Crystal Display.
                        display.lcd_display_string("Temp:"+str(temperature)+"C", 1)
                        display.lcd_display_string("Hum:"+str(humidity)+"%", 2)
                        time.sleep(2)
                        display.lcd_display_string("Time:"+str(val), 1)
                        time.sleep(2)
                        display.lcd_display_string("Meth:"+str(value)+" ppm",2)
                        time.sleep(2)

                except RuntimeError as error:
                        pass
                time.sleep(5)

```

C. A snippet of the PHP code used to connect database and Laravel application as well as chart rendering.

app > Charts > UserChart.php

```php
<?php

declare(strict_types = 1);

namespace App\Charts;
use DB;

use Chartisan\PHP\Chartisan;
use ConsoleTVs\Charts\BaseChart;
use Illuminate\Http\Request;
class UserChart extends BaseChart
{
    /**
     * Handles the HTTP request for the given chart.
     * It must always return an instance of Chartisan
     * and never a string or an array.
     */

    public function handler(Request $request): Chartisan
    {
        $servername = "localhost";
        $username = "root";
        $password = "";
        $dbname = "capstonedata";

// Create connection
$conn = mysqli_connect($servername, $username, $password, $dbname);
// Check connection
```

```php
if (!$conn) {
  die("Connection failed: " . mysqli_connect_error());
}
// For temperature
$sql1 = "SELECT * FROM temperature";
$result1 = mysqli_query($conn, $sql1);
// $temp=array();
$tme=array();
if (mysqli_num_rows($result1) > 0) {
  // output data of each row
  while($row = mysqli_fetch_assoc($result1)) {
    // $temp[]= ($row["temperature"])." ";
    $tme[]=($row["created_at"])." ";
     }
    } else {
    echo "0 results";
  }

  // all tables:
  $sql = "SELECT temperature,humidity,methane FROM temperature t1
  INNER JOIN humidity t2 ON t1.created_at = t2.created_at
  INNER JOIN methane t3 on t2.created_at=t3.created_at";
  // $sql_hum = "SELECT humidity FROM humidity";
  $result = mysqli_query($conn, $sql);
  // $try=array();
  $temp=array();
  $hum=array();
  $meth=array();
  if (mysqli_num_rows($result) > 0) {
    // output data of each row
    while($row= mysqli_fetch_assoc($result)) {
      $humi[]= ($row["humidity"]).""; 
      $temp[]= ($row["temperature"]).""; 
      $meth[]= ($row["methane"]).""; 
    }
  } else {
    echo "0 results";
  }

mysqli_close($conn);

        return Chartisan::build()
            ->labels($tme)

            ->dataset('Methane', $meth)
            ->dataset('Humidity', $humi)
            ->dataset('Temperature', $temp);
    }
}
```