ASHESI UNIVERSITY COLLEGE

A LOW COST IoT SYSTEM FOR MONITORING FUEL TANK CONDITIONS

CAPSTONE PROJECT

B.Sc. Electrical and Electronic Engineering

Jacob Nii Sackey Vanderpuye

2019

ASHESI UNIVERSITY

A LOW COST IoT SYSTEM FOR MONITORING FUEL TANK CONDITIONS

CAPSTONE PROJECT

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

Jacob, Vanderpuye

2019

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:

…………………………………………………………………………………………

Date:

…………………………………………………………………………………………

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 College.

Supervisor's Signature:

……………………………………………………………………………………………

Supervisor's Name:

……………………………………………………………………………………………

Date:

…………………………………………………………………………………………

Table of Contents

Table of Figures

List of tables

List of Abbreviations

AI – Artificial Intelligence

API – Application Programming Interface

DB – Database

IoT – Internet of Things

IP – Internet protocol

MQTT – Message Queuing Telemetry Transport

NSP – Network Service Provider

NGL – Natural gas Liquids

TCP – Transmission control protocol

UART - Universal asynchronous receiver-transmitter

Dedication

This paper is dedicated to my mother Abigail Pardie, whose faith in me can only be described as blind and limitless.

Acknowledgments

I would like to acknowledge my supervisor, Mr. Francis Gatsi, who was supported and guided me throughout my capstone project. Thank you very much. I would also like to say a big thank you to my friends, Benedict Quartey, Barnabas Nomo, Emmanuel Nimo, Christopher Anamalia, and Asantewaa Bremang who provided me with the much needed technical and moral support. I reserve the last of my “thank you”s for Mr. Nicholas Tali, who helped me to carry out this project and gain confidence in my abilities. God bless you all.

Abstract

Following the discovery of Oil in Ghana the upstream industry of the Oil and Gas has received a jumpstart and is augmenting the older and more established midstream and downstream industry. The midstream and downstream industries involve the processes that refine, transport, and distribute oil and other crude products and as economic activities increase there is an urgent need to vamp up production, and this can be done by increasing the efficiency of storage processes.

This paper explores in detail the design and creation of an IoT system tailored to the needs of the Ghanaian oil and gas industry with a focus on the midstream and downstream storage section. This paper delves deep into the underlying technology, mathematical models, communication protocols, storage architecture, and hardware components needed to realize an IoT system capable of significantly impacting the landscape of the storage of crude oil and other crude products.

The paper concludes with an account of limitations and how they can be used to develop future works and improve the current concept.

Chapter 1

1.0 Introduction

Since the Ghanaian government broke ground on the Oil and Gas industry has taken flight in Ghana, boasting a viral upstream, midstream and downstream industries that contribute to the nation's GDP [1]. In the upstream industry, crude oil and natural gas products are extracted from natural reserves. Midstream takes over to process, store, market and transport commodities such as crude oil, natural gas, natural gas liquids (NGLs, mainly ethane, propane, and butane) and Sulphur. The midstream provides the vital link between the far-flung petroleum producing areas and the population centers where most consumers are located [2]. Then the petroleum industry looks to the downstream industry to refine the petroleum products and make, brand them, make by-products and retail them to the end consumers [2].

Internet of things (IoT), is also a new and thriving technology that can be incorporated into several industries to achieve certain desired effects in the selected industry. The Internet of Things (IoT) refers to the use of intelligently connected devices and systems to leverage data gathered by embedded sensors and actuators in machines and other physical objects. IoT is expected to spread rapidly over the coming years, and this convergence will unleash a new dimension of services that improve the quality of life of consumers and productivity of enterprises [3]. IoT is already beginning to have far-reaching effects on many industries and have opened new prospects in industries such as healthcare, manufacturing, education, real estate, energy, Oil and Gas, and even construction. IoT has increased access to data and critical information needed for smooth running of institutions and systems. We live in the information age and volumes of data collected every day are helpful in the studying of trends and training AI algorithms to solve and avert problems. Concordantly, the Oil and Gas industry has been steering toward adopting IoT systems in monitoring and controlling their day to day operations [4]. However, most oil-based companies employ analog systems in their monitoring efforts due to the reduced risk of fire and explosions caused by to sparks from the electronic/digital components, and due to the minimal maintenance sed analog devices require [5]. Is this a valid concern? Can this concern be allayed? What is the way forward? Is the future of Oil-gas industry going to remain analog?

As previously mentioned, IoT involves connecting various devices, applications, and technologies in one coherent network. IoT integrates the following major components to attain the desired effects;

- Device (sensor)
- Local network
- Internet
- Back-end devices
- Applications

These components form nodes in the general IoT networks that generate and enable information to flow. In summary, this network merges, analytic engines, high-end processing, neural networks, LAN, WAN, machine learning, sensors, servers, and various algorithms to create a system that functions as a single yet diverse unit.

The benefits of IoT systems are very critical to the development of the oil and gas industry in the years to come [5]. Free access to data in the oil and gas sector could streamline many processes and make necessary improvements to the industry procedures, and this could give way to new and more convenient ways of doing things. In addition to making data available, IoT systems would enable remote control as well as automatic emergency responses.

In a paper published on IoT systems for liquid tanks in 2015, the author's designed and implemented an IoT system for fuel tanks. Viswanath, Belcastro, and Barton, designed and created a wireless low power liquid monitoring system using ultrasonic sensors [6]. In their paper, they remarked “Monitoring Systems are necessary to understand the changes that take place in environments. Remote monitoring and data collection systems are useful and effective tools to collect information from bulk storage tanks and to monitor the same. The measurement of liquid inside the tank is very important, and such systems are useful in industries which are as safety-critical systems” [6]. Based on such a premise they developed a working product, which could be remade with the African context in mind and could be improved to serve a wider range of purposes. This African context considers the limited internet access, power cuts, poor road networks, weak billing policies, inadequate data handling infrastructure.

1.1 Problem statement

Many fuel stations in Ghana use archaic instruments in their operations; consequently, the operators of these fuel stations suffer from thefts, preventable shortages, lack of inventory, inaccurate measurements and losses in both fuel and funds. These consequences can seep through the cracks on the interim; however, they have dire long-term effects on some major oil and gas companies as well as small scale owners. This deficit in technology and ethics may have far-reaching consequences on a company’s baseline and its capability to meet its goals [1]. Thus the need to develop a system capable of monitoring levels in stationary tanks to shore up efficiency, make mechanisms for taking inventory available, and for keeping the administrative body (business owner) updated in real time.

1.2 Proposed solution

To remedy the aforementioned problem, I must build an IoT system capable of tackling the pain points expressed by the user. Therefore, the solution would be capable of sensing conditions, smart use of data, interacting with operators, streamline inventory and accountability procedures, and being responsive – operating in real time.

1.4 Objectives

1. To develop a mechanism that would streamline the process of bulk purchasing of oil products – from the purchasing itself to the delivery of the purchased products to gas stations all around the country.
2. To make stock sales and inventory data easily accessible and easy to create.
3. To ensure a stable and efficient inventory system.
4. To bring more transparency and accountability to the oil and gas retailing processes
5. To reduce human interference in the monitoring system.

1.5 Expected outcome

1. A working prototype of a fuel head detection module
2. An effective and efficient web app to serve as a graphical user interface and a platform for other administrative functions that come with the product.
3. Large volumes of data that can be exported into multiple formats and adapted to a variety of purposes.

1.6 Motivation

This project unites the following disciplines; programming, database programming, sensor design, embedded systems, modeling, software design, design, and machine learning. This interdisciplinary project would be challenging and would require hard work in order to meet the project objectives. Also, this project would yield a product that would be very beneficial to mainstream oil and gas industry and would improve upon transparency within the fuel delivery system. These strides could catapult the oil and gas industry into newer heights with top-notch and affordable technology. Also, there is a personal motivation that urges me to challenge myself to take on such a daunting task, to climax my four-year bachelor’s degree. This project would draw out the strength, wit, determination, will power and calm of anybody who undertakes it; thus it becomes a learning process for myself.

1.7 Research Methods

- Interviews
- Prototype testing
- Questionnaires
- Observation
- Experimentation

The main research method would be to review other literature within this given field, and this would provide insights into tried and tested ways of monitoring fuel levels remotely. So far, the novelty in this project lies in incorporating various aspects of software design, databases, sensing console, and a tracking algorithm.

1.8 Facilities, Logistics, and materials needed

Facilities

- Fabrication lab
- Test Fuel Station

Logistics

- Computer
- 3D printer
- Screw set
- Power supply
- Server space

Materials needed

- PCB board
- 3D printer filament
- Ultra-Sonic sensor

1.9 Scope

This project would have to be limited in scope albeit its far-reaching effect within the industry. The limited scope would ensure that the researcher meets the designated deadlines and requirements, stays within the financial constraints and most importantly to ensure the problems are solved and the user is satisfied. The scope would limit the design to a device built with an integrated sensor, a GSM model and a microcontroller. The device would work with some databases controlled by some sophisticated queries and hosted on a cloud. Then there would be a simple web app or an API to serve as a graphical user interface.

1.91 System overview:

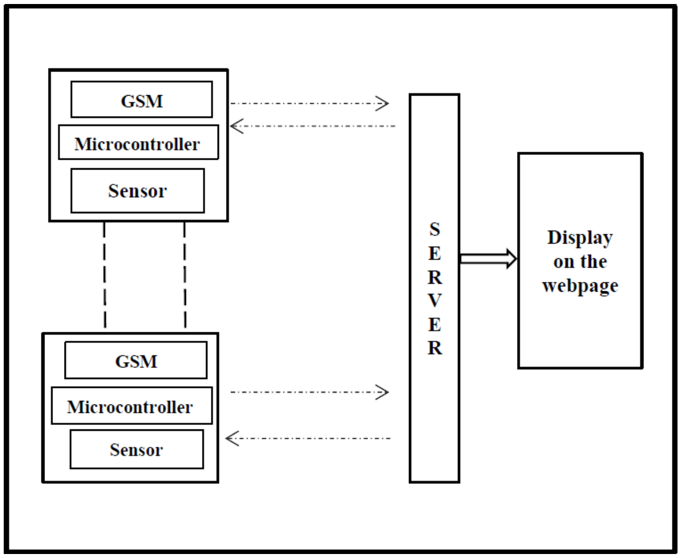


Figure 1Block diagram showing the overview of the system plan [6]

Chapter 2

2.0 Literature Review

To successfully execute this project similar projects were reviewed using two methods – a literature review and a series of interviews with industry experts. To lend more credence to this project a literature review was carried out on works by other engineers and scientist in the field of specialized liquid monitoring systems [6]. Reviewing these papers and identifying gaps gave more context to the work and provided insights on what to do. The papers under review [7], [6] were written in 2015 by Samarth Viswanath, Marco Belcastro, John Barton, Brendan O'Flynn, Nicholas Holmes, and Paul Dixon, and Kindawi respectively. This body of work served as excellent reference literature on the topic of remote monitoring of liquid holding tanks.

2.1 Introduction

In summary, this paper presents the architecture and initial testing results of a low power wireless system for tank level monitoring using ultrasonic sensors. The researchers designed, prototyped, implemented and tested a device that was capable of reading sensor values, transducing the values read, processing them and sending them to a server, before displaying the readings on a webpage [6].

2.2 Problem statement

These projects under review were motivated by the need for monitoring systems to understand changes in the environment (tank) especially in case of fluids which's spillage may be hazardous and close inspection may be perilous to human life. These fluids include flammable fuels which are stored in tanks to be dispensed as and when they are needed [7]. Remote monitoring and data collection systems are useful and effective tools to collect information from bulk storage tanks and to monitor the same. The measurement of liquid inside the tank is most important and such systems are useful in industries which are categorized as critical safety systems. Thus, the need for a remote monitoring system, also since these tanks are located far away from locations inhabited by human beings; it is important that they are wireless and can be remotely controlled [6].

2.3 Approach to solving the problem

The problem in question was remedied by implementing the system illustrated in the block diagram below. The system was accompanied by a high-level software and a web-based GUI to enable operators monitor and control the system remotely and in real time.

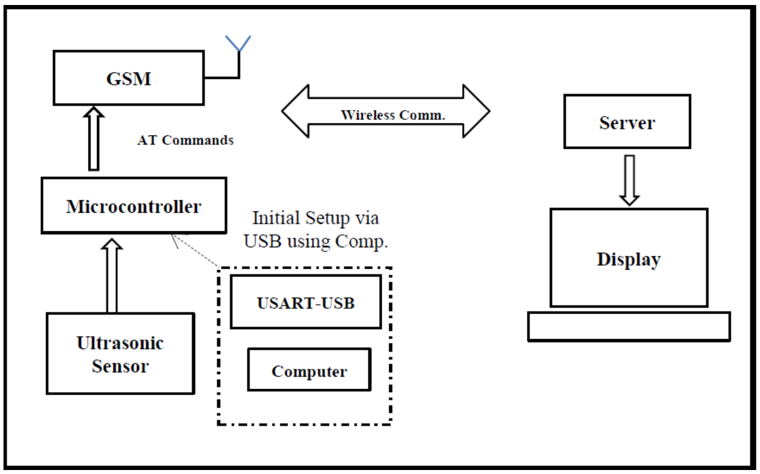


Figure 2 Overview of the operation of the entire system [6]

As shown in figure 2.1 above, the ultrasonic sensor reads the tank levels and these discrete values are sent to the microcontroller unit for processing. The values are then sent to a server using GSM wireless communication. Once the information reaches the server, it could either be stored or displayed on the web interface.

2.3.1 Sensing

An ultrasonic sensor is used to sense the amount of liquid inside the tank. The general principle of the ultrasonic sensor is to send out high-frequency waves which are reflected when it strikes an obstacle, in this case, the liquid surface. The period between the transmitting and reflecting waves (Known as the time of flight) is measured by the microcontroller. This time of flight is used to determine the distance traveled by the waves and extrapolate the depth of the liquid in the tank from the point where the sensor is placed [6].

2.3.2 Microcontroller Unit

The MCU used was an ARM-based 32-bit controller. It was used for analog to digital conversions and storing the information gathered onto a flash ROM before sending the information onto the server using a GSM module. The MCU also had two USART links and some features that enabled it to operate in low power mode [6].

2.3.3 GSM module

The researchers used a Huawei MG323-B which uses the serial communication link to and from the microcontroller [6].

2.3.4 Server

The GSM module directly connects to the server which collects these values and represents them on the webpage graphically.

2.3.5 General Hardware Implementation

The main hardware component consisted of a PCB board encased in a plastic casing. The PCB board consisted of the GSM module, microcontroller unit, and the sensor. It also had a power system which consists of a voltage regulator to ensure the two levels of voltage required are met (i.e., 3.3 V and 5 V).

2.3.6 Software Implementation

The software that accompanied the system was written in embedded C. The STM32 microcontroller was programmed using a GCC Compiler based integrated development environment and the ST-Link debugger was used to program the controller. The system would initially set up with the ultrasonic sensor, the ADC module, the USART channel all connected to USB via the USB to UART converter and then another USART to the GSM module. The AT commands are used to communicate to the GSM module. The system enters the GPRS mode and tries to connect to the host whose IP address and the port of access is provided during the initial setup. Then, as and when it connects to the host server, the server would send the initial setup configuration to the microcontroller in the tank and is received via GSM module. This generally would contain the current time, the wake-up time, the higher and lower threshold concerning temperature and other concerned parameters [6].

2.4 Gaps

Upon thoroughly reviewing the paper, I noticed that the system received volumes of data that were either displayed, trigger an alarm or remained stored on a server. However, the researchers failed to use data acquired to train the system. The potential of an AI system would be great in improving efficiency and making the system smarter. If an AI system could be included, triggering of the system could be made automatic, the system could send messages to operators, the system could also put itself into standby mode and power saving mode without any input from the operator. Also, an intelligent system could warn operators of battery power available and if the system has been tampered with [6]. To incorporate AI into the system there would have to be a machine learning algorithm that would train the system based on the data collected. This would result in a system that is autonomous and dynamic.

2.5 Conclusion

In conclusion, the researchers designed and developed a wireless liquid monitoring system using ultrasonic sensors. The system was tested, and it was found that it runs as expected with the required functionalities such as power saving mode and battery save mode and rest mode [6]. Thus, per their initial requirement to develop a system that could last over two years without a change in battery, their project was a success.

Based on the findings from the literature reviewed, a combination of GSM technology and ultrasonic sensors would be a good fit for the problem under consideration. Although the paper under review was addressing a problem associated with water tanks, the technology could be easily appropriated to suit fuel tanks. GSM technology ties into project objectives seamlessly because it is cheap and easy to use compared to the other modes of communication that were available (LORA and Wi-Fi modules) [7]. Also, Ultrasound technology is cheaper and easier to use compared to the fuel probes, which were expensive and functioned on patented technology [6]. By combining these technologies, a good solution can be produced within the time, financial and user complexity constraints.

Chapter 3

This chapter details out the entire process involved in planning out the final design to solve the problem at under focus. This chapter introduces, justifies, and presents alternatives to the technology and devices used to complete this capstone project.

3.**0 Requirement analysis and Design**

This chapter offers insights as to the desired features imposed on the product either by the user or other constraints. These features are recorded as either user requirements, design requirements, or standards depending on their source. These were further sorted and filtered using a series of Pugh matrices.

3.1 Requirements

To give the design process more focus and direction, a series of analysis was carried out on all the requirements that were imposed during both ideation and design execution stages. These requirements could be divided into two broad groups. User requirements, Design requirements, and standards.

3.1.1 User requirements

These are the demands made by the end users of the product being designed. This ensures that the solution developed is the best fit for the given problem being addressed.

Here are a list of user requirements and their priority to the estimated priority rating

Table 15 Summarized Pugh matrix ranking the requirements' priority

Requirement	Weight	Baseline	Cost	Complexity	Need	score	Rank
A GUI that would enable operators to see changes	5	0	0	-	+	5	2
A low power system	3	0	0	-	-	-6	5
A wireless system	4	0	+	-	+	4	4
No fire hazards	5	0	+	+	+	15	1
The quick response of the system to changes	4	0	0	0	+	4	2

From the table above the user's expectations are observed, and the design could be guided to meet such expectations.

3.1.2 Design Requirements

The design requirement features the essential features that need to be incorporated to ensure the design is realized as planned. Design requirements may impose constraints or include some other factors that may make the project more successful [8].

Table 16 Concise Pugh matrix ranking the design constraints

Constraint	Weight	Baseline	Cost	Complexity	Score	Rank
Physically light system	4	0	+	-	4	3
Minimal bandwidth consumption during data transfer	5	0	0	-	3	3
Data transfer process must be triggered	3	0	-	0	2	3
Minimal heating	4	0	0	0	4	4
Low power consumption	4	0	-	-	1	3
Capable of reset	4	0	-	-	1	
Data collected must be capable of exportation to other forms	5	0	0	+	6	3
The casing must be liquid proof	5	0	0	0	5	4

These requirements would give more context to design decisions and would justify certain design processes.

3.1.3 Standards

These are the markers and constraints imposed by the engineering community, in general, to ensure that the product delivered meets professional, ethical and engineering standards. These standards when imposed would ensure the product qualifies as an engineering solution and that it does not put life or property at risk. The Pugh matrix ensures that the design is executed within the confines set by proper and sustainable engineering practice. It also ensures that the standards are prioritized so that none of the environmental standards are breached during the design or by the finished product.

Table 17 Concise Pugh matrix ranking the regulatory standards

Standards	Weight	Baseline	Cost	Complexity	Score	Rank
Environmental (CO_2) emissions	5	0	0	0	5	1
Emission of radio waves	3	0	-	-	1	3
Power consumption	4	0	-	0	2	2

3.2 Design

Once the requirements had been filtered and prioritized, the design progressed to the stage where modules and components were selected and arranged. Alternatives were also sought at this stage of design.

The system comprises of various modules and concepts which were integrated. These modules and subsystems are explained in the paragraphs that follow:

Sensors: The main sensors used were the ultrasonic sensor and the temperature sensor combined in a sensor suite and adapted to read the temperature and fluid level respectively [9]. The ultrasonic sensor was positioned facing downward so that the waves could have a direct path to be incident on the fluid surface and could have a direct line of sight to travel back. The temperature sensor used was also adapted to read ambient temperature [3].

Microcontroller unit: The MCU used is an Atmega8 and this device was responsible for setting the frequency of measurements and transduction, analog to digital conversion and interfacing with other components [8].

GSM module: The GSM module used is a SIM800L and it is connected to an MCU and uses serial communication to transfer data. To transfer data packets using GSM, a UART link is established and uses a series of AT commands to communicate with the GSM module and configures the same to GPRS mode and sends and receives data packets wirelessly [8].

Server: The server receives data packets directly from the GSM module wirelessly. The data received and kept in the server is then made available to the web app and this could be displayed on web pages or analyzed using the Artificial intelligence (AI) module.

AI Bot: This is an automation system with no physical manifestations whatsoever. It reviews data received from the console and then makes decisions on the next course of action. It also publishes automated messages in the form of alerts, salutations, and reminders [10].

The system design would be discussed in two phases, the software, and the hardware. This design would incorporate software and hardware and work seamlessly by interfacing the two to produce the needed outcome.

3.3 Hardware Design.

As mentioned in earlier chapters, the main hardware components are the printed circuit board and a durable plastic encasement.

The PCB enables the integration and interconnection of the major electronic components and power system. The components soldered onto the PCB board are described in the paragraphs below:

The MCU: The ATmega8 is a low-power CMOS 8-bit microcontroller based on the AVR RISC architecture. By executing powerful instructions in a single clock cycle, the ATmega8 achieves throughputs approaching 1MIPS per MHz; It has 32 working general purpose registers which are directly connected to the ALU thus allowing two independent registers to be accessed in one single instruction executed in one clock cycle. The resulting architecture is more code efficient while achieving throughputs up to ten times faster than conventional CISC microcontrollers [8].

Sensor: The sensor used was the HC-SR04 from Adafruit. The ultrasonic ranging module HC - SR04 provides 2cm - 400cm non-contact measurement function, the ranging accuracy can reach up to 3mm. The module includes ultrasonic transmitters, receiver, and control circuit [9].

GSM module. The GSM module used was SIM800L is a miniature cellular module which allows for GPRS transmission, sending and receiving SMS and making and receiving voice calls. Low cost and small footprint and quad-band frequency support make this perfect module solution for any project that requires long-range connectivity.

Casing. The physical casing for the console was made of polylactic acid (PLA), as a biodegradable thermoplastic, PLA is more environmentally friendly and thus was chosen. The casing was fabricated using 3D printing.

3.4 Software Design

The software design used for all the three types of software involved are illustrated using flow charts which serve as a roadmap for programming the software.

3.4.1 MCU control software

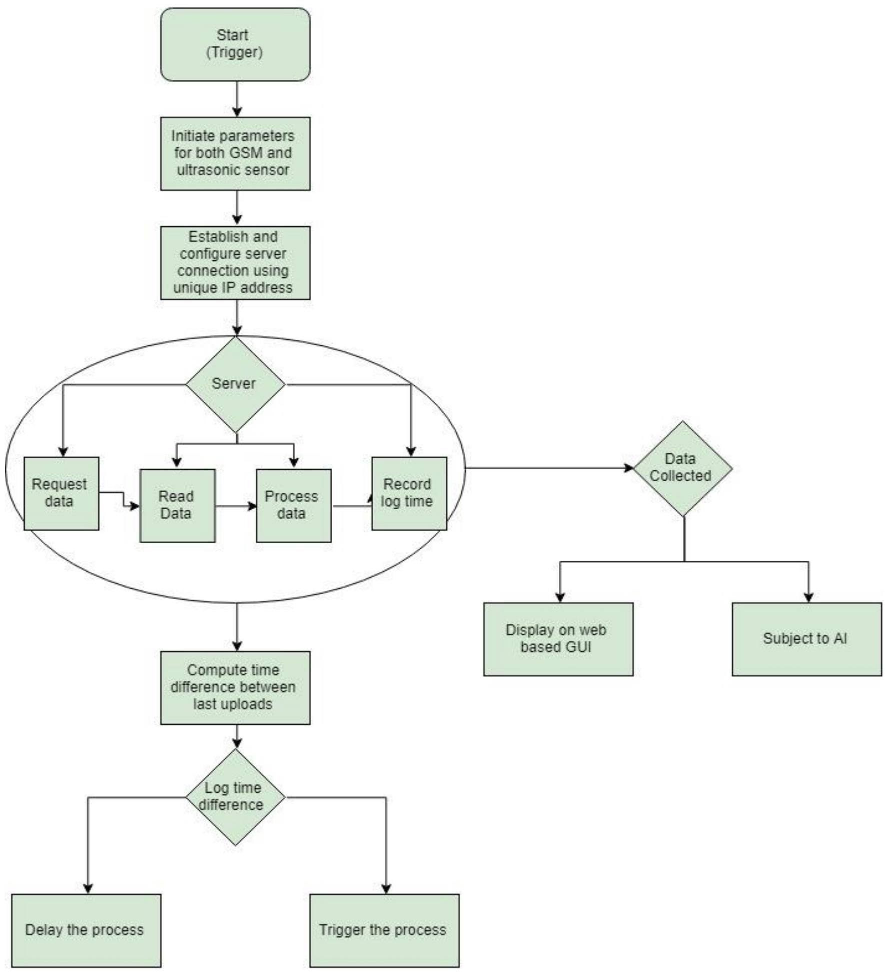


Figure 3 Flow chart showing the step by step procedure of operation of the console software

This flowchart shows the stages of operation that make up the operation of the MCU control software.

3.4.2 Web app implementation

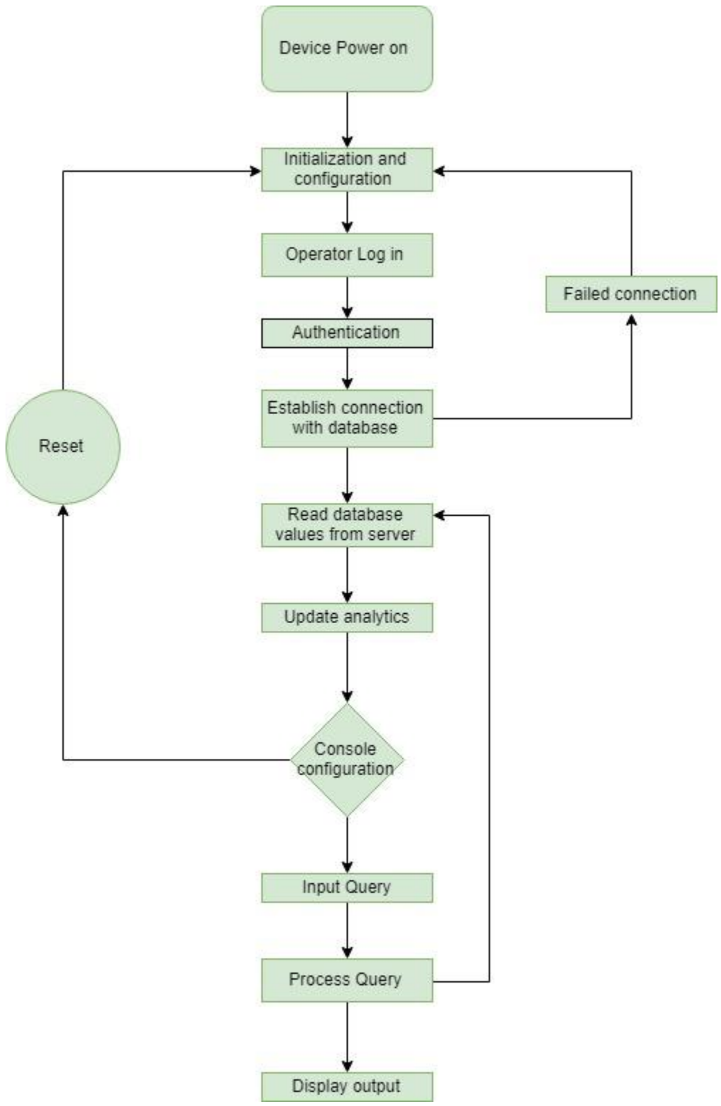


Figure 4 Flow chart showing the operational procedures of the web app

This flowchart shows the background processes that ensure the operation of the web app and rapid response to queries and rendering of web app front end.

The web app ensures the usability of the product and offers the operator the chance to interact and monitor the system, by providing a UI that displays and interprets data in both numerical and graphical form. It also receives search and sort queries and processes them to satisfy the user's needs. The web app was coded in node.js a runtime of JavaScript due to the developer's preferences. Also, the developers used the Adonis framework since it offers scores of libraries and templates for creating functional web apps. The web services can be accessed using the given URL and operate on minimal bandwidth thus addressing an aspect of the user's pain point associated with cost. This design touched on the user requirement

3.4.3 AI bot implementation

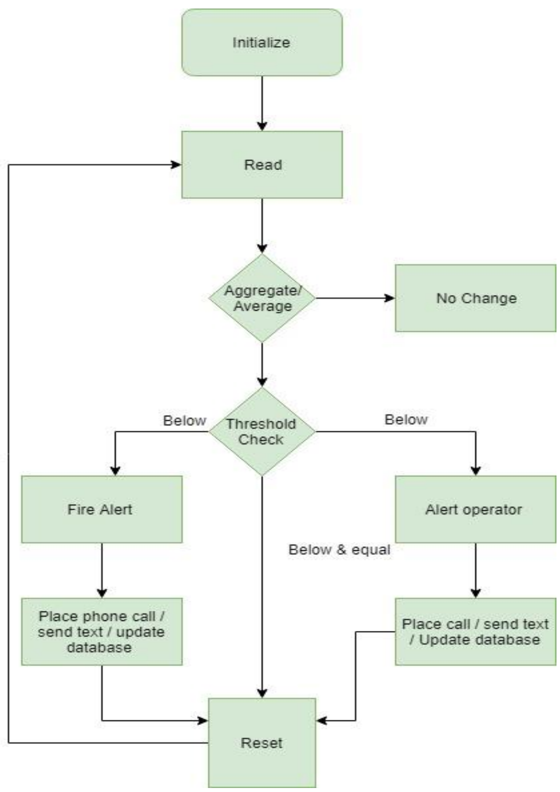


Figure 5 Flow chart illustrating the operational procedure of the AI bot

The AI bot was created to assist the operator in their management of the system. The AI bot is trained using information that was collected over a certain time window; thus it would be able to place calls, send SMS, process data, and interact with the operator. This chatbot, although primitive in function, would be essential when it comes to giving the operator an easy and enjoyable experience when using the IOT system. Also, certain basic AI algorithms would be deployed to make the predictive functions of the system more accurate.

As mentioned already, this project would incorporate all of the modules mentioned above and components to ensure the project objectives, as well as the requirements, are met.

3.5 Implementation processes

This subchapter describes how the respective technologies were put together to ensure the aims of the project design were met.

Sensing: This is mainly carried out by the ultrasonic sensor. The sensor used was the HC-SR04 which can provide 2cm - 400cm non-contact measurement function, with a ranging accuracy, can reach to 3mm. The principle of operation is described as follows:

- An IO is used to trigger for at least 10 microseconds
- The module then sends out at least 40KHz signal and simultaneously detects if any signals are being sent back
- The time the feedback is received is recorded and using the time of flight the distance is calculated using the following formula:

$$\text{Test distance} = (\text{time of flight} \times \text{velocity of sound } (340\text{M/S}) \,/\, 2 \qquad [9]$$

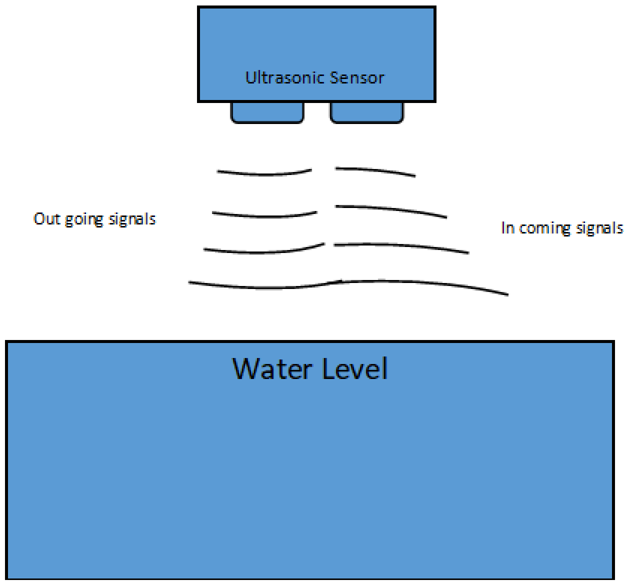


Figure 6 Illustration of the principle of operation of the ultrasonic sensor

Level Detection

To create a system that is more responsive to level changes and can smartly use data sensed, the level detection process was modeled using a tank with an assumed height and cross-sectional area.

Given a tank of Height H, the height of fuel is 'h,' and the distance between the sensor and the fuel surface is l. To develop a comprehensive mathematical model the tank was modeled with a unit cross-sectional area and fixed capacity. The importance of this is that it forms the premise for developing the primitive version of the AI, which would serve as the basis for increased functionality and more complex processes.

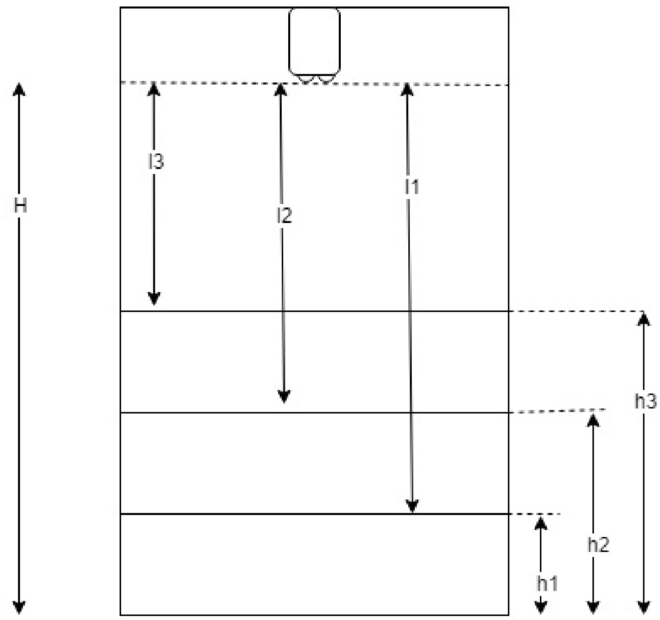


Figure 7 Illustration of ideal tank model

$$Total\ height\ of\ tank = li + h\mathrm{i}$$

$$\mathrm{H} = l3 + h3$$

$$H = l2 + h2$$

$$H = l1 + h1$$

$$\therefore hi = H - li$$

$$Rate\ of\ change\ in\ head = \frac{\triangle\, hi}{\triangle\, t} = \frac{H - l_i}{\triangle\, t}$$

$$R = \frac{dhi}{dt}$$

Given the rate of change in head, and a fixed unit cross-sectional area, and a steady rate of inflow and outflow, we can model the quantity of fuel in the tank at every point in time.

Also, assuming the inflow and outflow rates can be estimated, then the time at which a shortage would occur can be determined using the mathematical model.

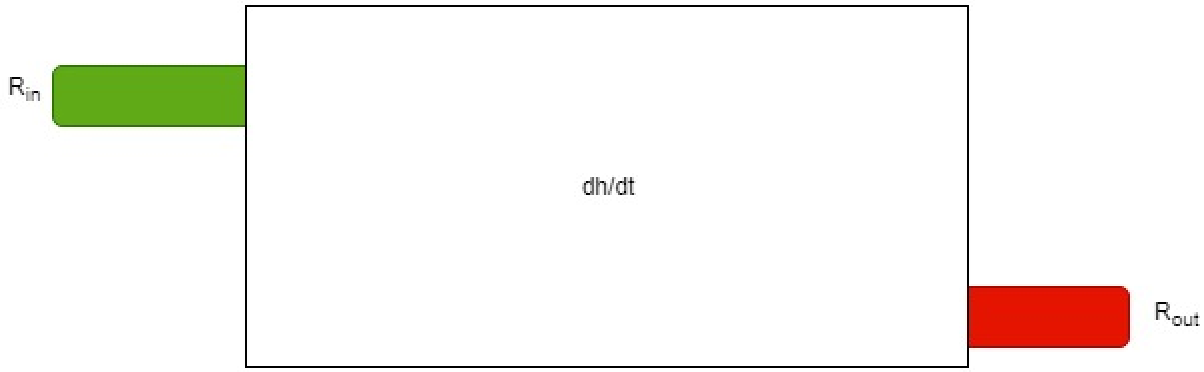


Figure 8 Model of the tank as an unsteady flow system

To effectively model the system the following assumptions were made:

The system was characterized by unsteady flow

The analysis only begins after a major supply run, i.e., R_{in} is taken over a short period.

$$\frac{dh}{dt} = R_{in} - R_{out}$$

$$In\ terms\ of\ volume\ flow\ rate\ (V^{\cdot})$$

$$\frac{dV}{dt} = v_{in}^{\cdot} - v_{out}^{\cdot}$$

$$S(t) = \int_{t_o}^{t} (R_{in} - R_{out}) + S(t_0)$$

The equation above makes it possible to estimate the quantity of fuel in stock (within the tank) at any given time. [11] This model features prominently in the development of the structure for the predictive feature of the system software, i.e. the model can be used to calculate the daily consumption rate within active hours in order to give a prediction of when shortages may occur. The above scenario holds if the threshold for shortage is set manually. However, if the threshold is not set, then this model can be used in conjunction with several assumptions to make the shortage predictions and set the necessary thresholds.

3.6 Processes within the Integrated circuit.

Receiving data: This could either come from the sensors or from the console itself, i.e., due to proprioceptive sensing that may occur (e.g., Battery levels, internet connection, etc.) [8]

Analog to digital conversion: Since the data received from the sensor is analog, there is a need for sed data to be converted into digital form to enable the MCU to work with the given data values. The Analog to digital conversion is done by MCU, specifically port PC0. The MCU takes an analog signal from either the sensors or the proprioceptive sensor ports and then samples and quantizes the continuous signal to yield a discrete signal [12].

Configuration: The IP address shall be checked and updated upon request because it is not static.In fact, most of the layers of communication fall within the communication procedure.

Transfer: This is the sending of information from the console to the server and the web app. This mode of data transfer makes use of MQTT to transfer data packets with minimal bandwidth.

3.7 Processes within the front-end interface

Data reception: The web app retrieves data from the server, from a SQL database. However, before populating the database, a python script listens to a particular MQTT channel and collects discreetly sent datapoints which are processed and used to populate the database tables. The data is then further processed and put to the needed use. Since the data received is in XML format it is extracted and sectioned into tables for display. Also, it could be categorized to produce graphs and smart shapes to represent the data. In transferring data I used MQTT protocol which has very minimal bandwidth consumption, thus, ensuring that the low-cost design objective will be realized.

Queries: Queries utilized certain keywords to issue a command that returns some category of data values. The queries could be used for time-based searches, location-based searches, quantity (trend) based searches, etc. Also, queries could be built into the app or input by the operators.

Chapter 4

This chapter covers the process of executing the design plan and creating the product. It details out the manner and timeline involved in executing design plans and pivots and changes made to the plan.

4.0 Implementation.

In implementing this project, I followed the floorplan outlined in the product design (Chapter 3) and pivoted slightly whenever the need arose. During the implementation phase the bulk of resources were expended – the implementation spanned five months with an average daily input of 5-man hours and a total cost of 77.4 dollars in purchases. As it was with the design, the implementation was also divided into two sections – hardware implementation and software implementation. These two sections were carried out simultaneously due to the time constraint however they were independent of each other due to their varying levels of complexity, until' they were finally integrated at the end of the implementation process.

4.1 Hardware implementation

After assessing the problem statement and the user requirements, I followed the design suggested in chapter 3 to develop the prototype for the IoT device which consisted of the sensor suite, GSM module, microcontroller, power supply, and outer casing. Except for the casing, the components were integrated into a single circuited printed onto a Bakelite circuit board.

The hardware implementation began with designing a circuit based on the given user and design requirements. The circuit was first planned out on paper and then designed on Eagle CAD software. It was simulated in Proteus, and then simulated again on a breadboard before the circuit designed using Eagle AutoCAD was printed out, and the respective components were soldered onto it.

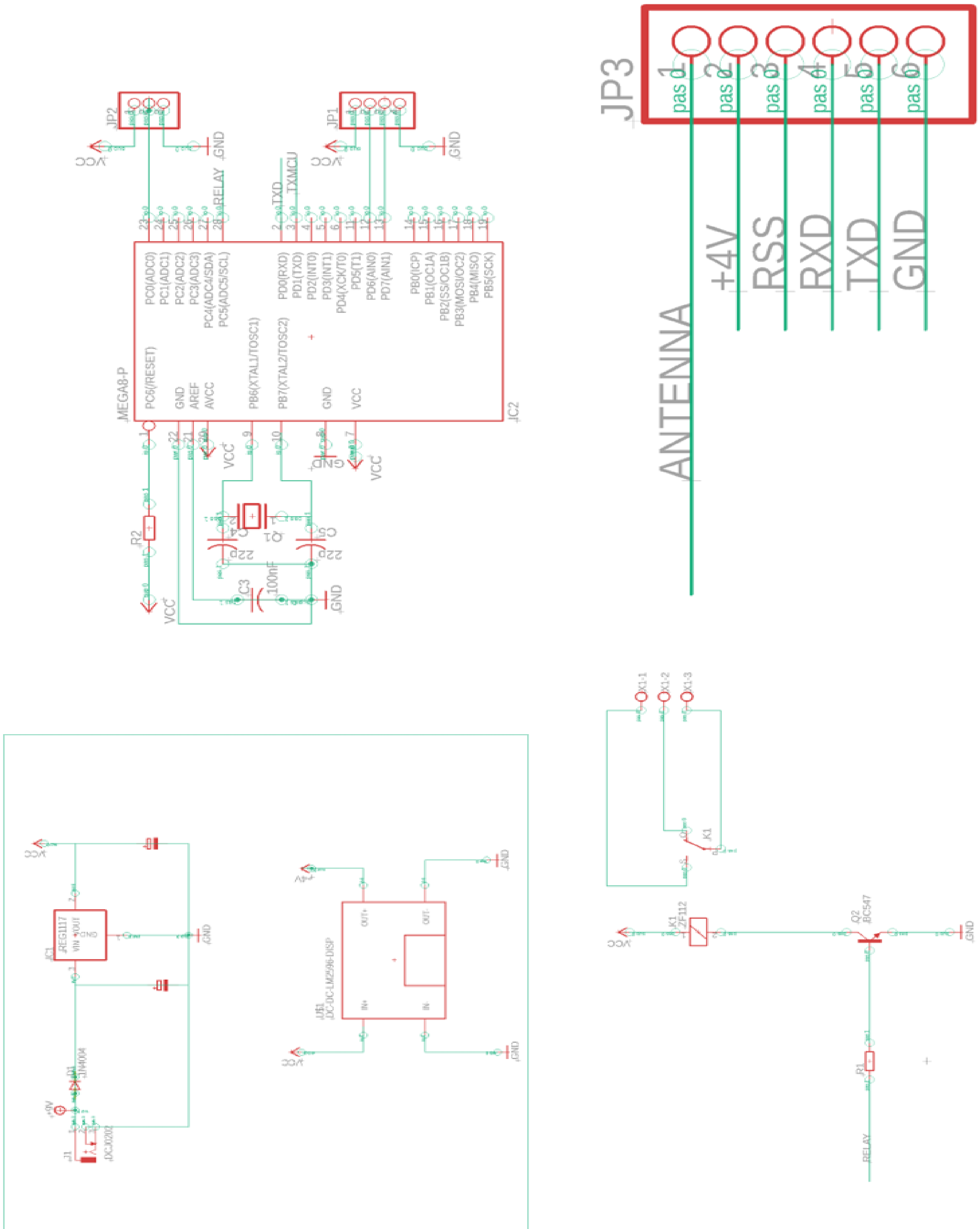


Figure 9Circuit schematic designed in Eagle CAD software

In printing the board, a drill bit of diameter 3.2mm was used and a copper trace of width 0.15 mm was used for the routing. The board was then finished off with a green solder mask – the board has an area of 5756.35mm^2 and a height of 71.11mm, with two copper layers.

As seen in figure 7 above, the respective circuit components were designed individually as a precaution to ensure maximum efficiency before it was eventually arranged to form a single unit. Before the PCB was printed out, the circuit was simulated in Proteus to ensure continuity and compatibility of the ICs for the GSM module, the MCU, and the buck converter.

To carry out the simulation the circuit design was exported into Proteus, and a test signal was sent through the necessary channels, and the results were observed. Once continuity was confirmed, and compatibility was assessed, the circuit schematic and the factory schematic were sent to be printed in a factory in China.

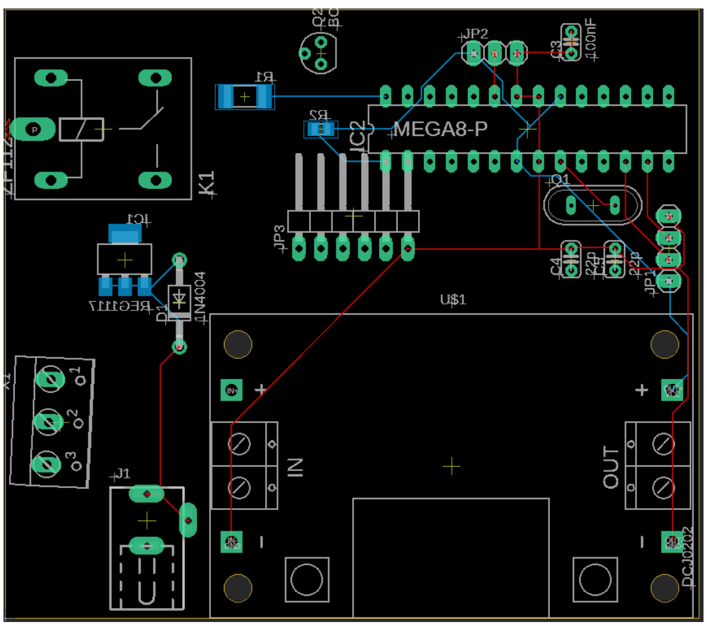


Figure 10 Board Design

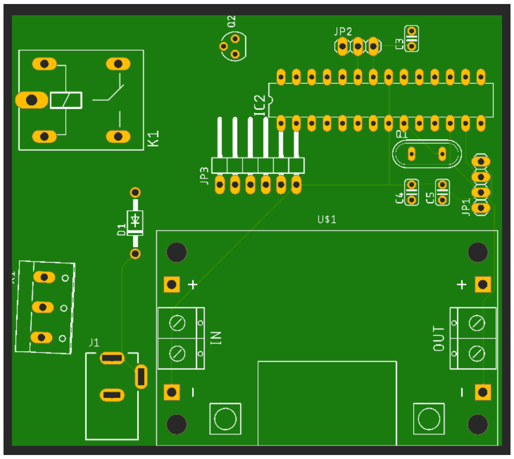


Figure 11 manufacturing layout front

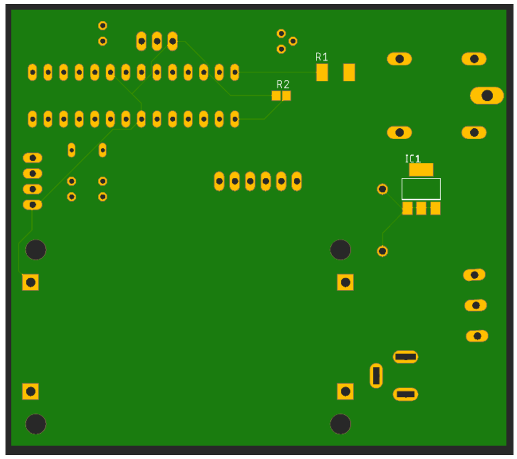


Figure 12 manufacturing layout back

Soldering

Soldering of the was carried out with the utmost caution and the surface mount components were placed and glued with pincers. The soldering was done with lead and a hot soldering iron.

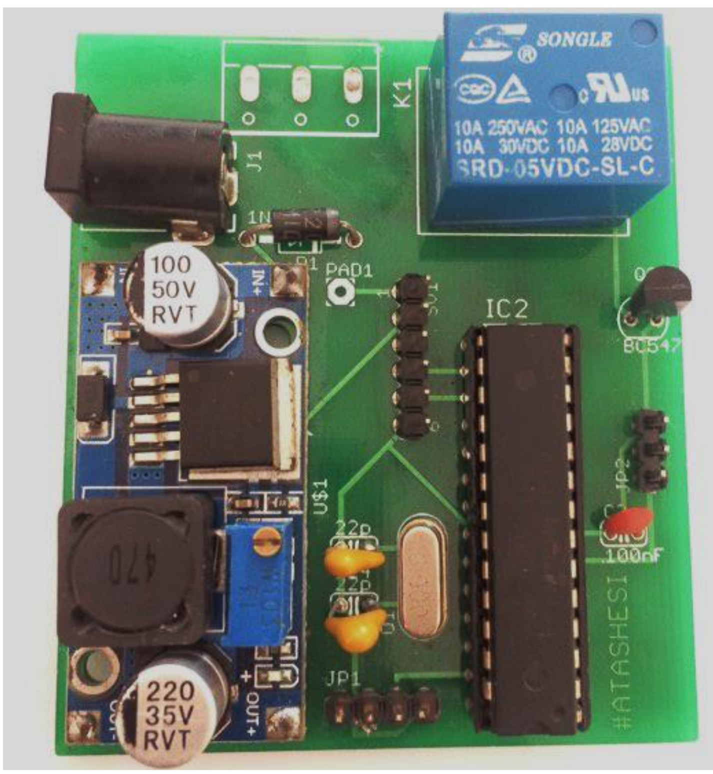


Figure 13 image of the circuit board after soldering

After soldering, some further tests were carried out to ensure continuity once again and that there were no short circuits. To carry out this test, I applied a series of test waveforms and measured their output using the Analog discovery waveform.

4.1.1 Assembling

After collecting and configuring all the discrete physical components, the next step was to assemble them into a single unit.

4.2 Software implementation

In general, the approach taken to developing the product prototype was the agile method, with few adaptations. Two types of software programming were carried out for this project, first was software written to interface with the MCU and the other major circuit components, (GSM, Sensor, Buck converter, etc.). The second software was written to develop a web interface, databases, and artificial intelligence. Each type of software would be further discussed in the paragraphs that follow.

4.2.1 MCU software:

The MCU software was written in a servlet of C, called Arduino. The program fulfilled the following objectives:

I. Trigger to exit from battery saver mode
II. Initialize the GSM module
III. Initialize the sensors and start taking values.
IV. Collect the data point and store it in the flash ROM
V. Establish a connection with the database and send data points.

4.2.2 Web application software

The following are brief descriptions of the tasks performed by the web app software

I. Receives data from the sensor
II. Populates database with data points
III. Receives queries and fetches data from the databases.
IV. Provides an interface for the operators and users to observe the conditions in the tanks

As mentioned in previous chapters, the web application software was programmed using node.js, in conjunction with MySQL. The raven interface is hosted locally on port 3333 and enables for certain extraneous functionalities such as inventory and employee information, and any other functionalities that the user may desire. To populate the database the sensing device uses MQTT protocol which is a lightweight data transfer protocol to send data to a broker. A python script was then written to automatically subscribe to the Topic with which the MCU communicated with the broker and then insert data points into the database.

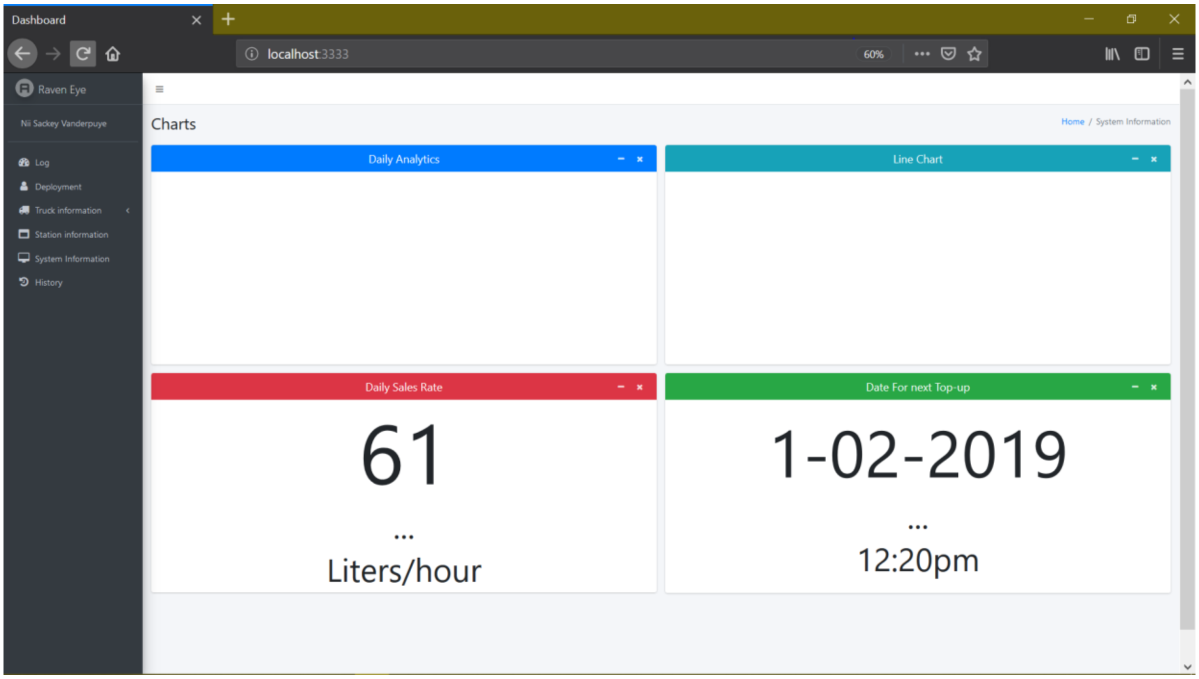


Figure 14 Image of Raven Eye home interface

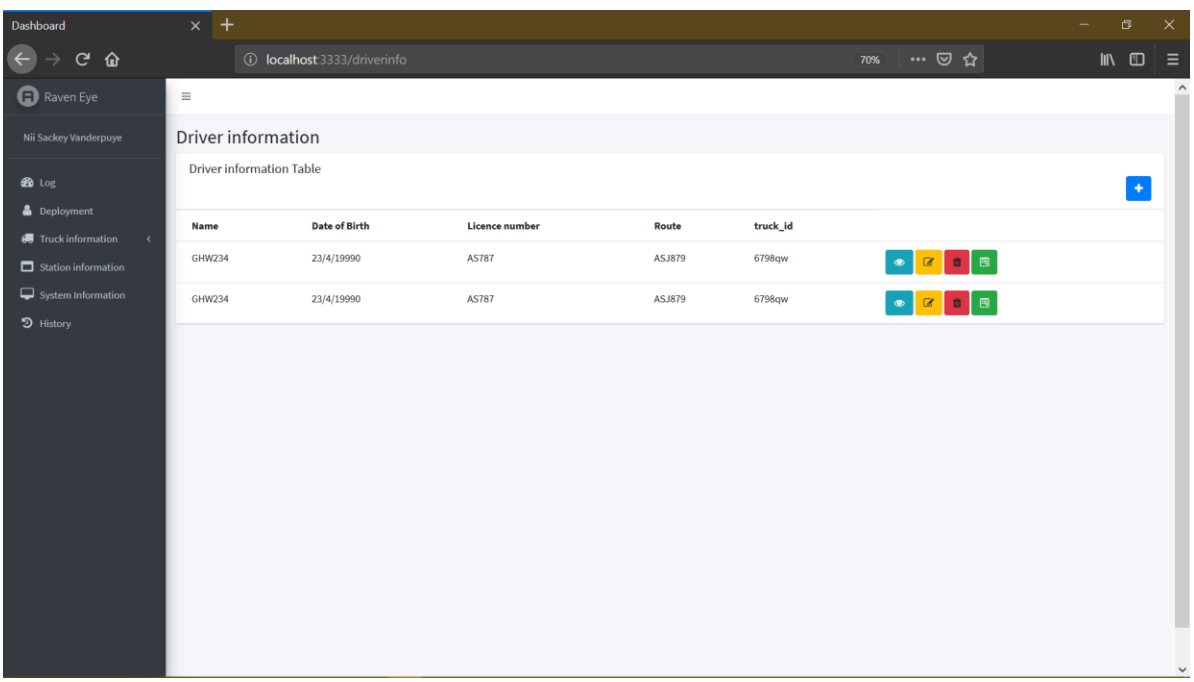


Figure 15 Image of Raven Eye's Custom features

4.3 AI algorithm

The AI algorithm forms the last part of the project, and it features a basic predictive function which is based on the flow characteristics observed within the tank. However, the AI algorithm can take multiple datasets and use them to finetune its predictive output. It is intended to take the date, the time, and flow characteristics of available datasets into account when revising its predictive output.

Chapter 5: Results and Analysis

5.0 Execution

Following the implementation and building phase, the product was piloted under certain conditions to test its functionality, durability, efficiency, accuracy, and speed. To test all the aforementioned variables the test environment was manipulated to simulate some conditions that the product would encounter in the field if deployed. The findings from this stage would be used to inform and justify any needed changes to the original design.

Consequently, a series of experiments were designed and carried out to test the respective variables.

5.1 Functionality

5.1.1 App functionality

The test for App functionality was carried out by simply clicking buttons within the app. The test yielded a maximum response time 1.44 seconds. Also, a positive correlation was observed between internet speed and response time, as expected. Another variation of the functionality test was carried out by opening the application on four different devices.

Table 18 App functionality test results

Device	Grade (out of 10)	Notes
Desktop computer	9.5	All features of the app were displayed as expected
Mobile Phone	7	• Some buttons were unresponsive • Animations and graphs did not show
Tablet (iPad)	7.5	• Some buttons were unresponsive
iPad Pro	9.5	All features of the app were displayed as expected

The average score across board for the respective devices was 8, which gives a rough estimation of the chances of failure of the web app is 20% in general and 5% on a desktop computer.

5.12 AI functionality

The experiment to determine the system's intelligence involved simulating certain conditions and recording the AI's response to the given conditions (varying flow rates) and then comparing them with the expected results which were manually calculated using the math model upon which the AI was built. These operations were performed under the assumption that the $20m^3$ tank was full and the net inflow rate was zero.

Table 19 Precision checks

Conditions	AI result for time (s)	Math model for time (s)	Error
Outflow rate = 0.001 m^3/s	16321.193	20000seconds	0.184
Out flow rate = 0.01m^3/s	1778.4332	2000seconds	0.110
Outflow rate = 0.1 m^3/s	192.33469	200seconds	0.0383
Outflow rate = 1 m^3/s	19.00422	20seconds	0.05
Outflow rate = 10 m^3/s	2.38579	2 seconds	-1.2495

The average error for this set of tests was 0.047 (4.78%) which is an acceptable tolerance rate.

To further test the predictive abilities of the AI algorithm, it was made to predict the time to that would elapse before fuel in a tank is exhausted. This test was carried out using water since its viscosity is close to that of petrol (Gasoline), in a tank with a fixed volume of $5500cm^3$ (5.5 liters). The performance of the algorithm was then measured three times under similar conditions, with varying levels of training, and the error was observed. The levels of training were:

- No data set, only flow rate calculations
- Training with one data set
- Training with two data sets.

This AI algorithm is basic in both function and construction; it uses a fluid flow model to calculate the time to empty the tank using the inflow and outflow rates, it then uses a given data set to refine the output of systems dynamic model and supplies that output as a response if queried. In refining the prediction, the algorithm considers the time, date, and flow rates.

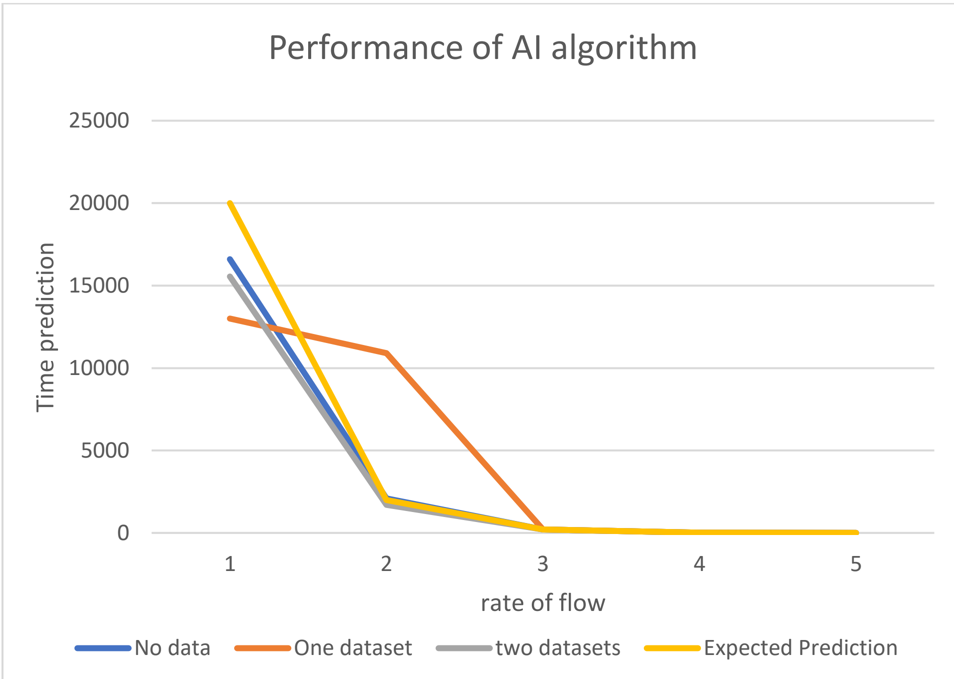


Figure 16 Illustration of performance of the AI module

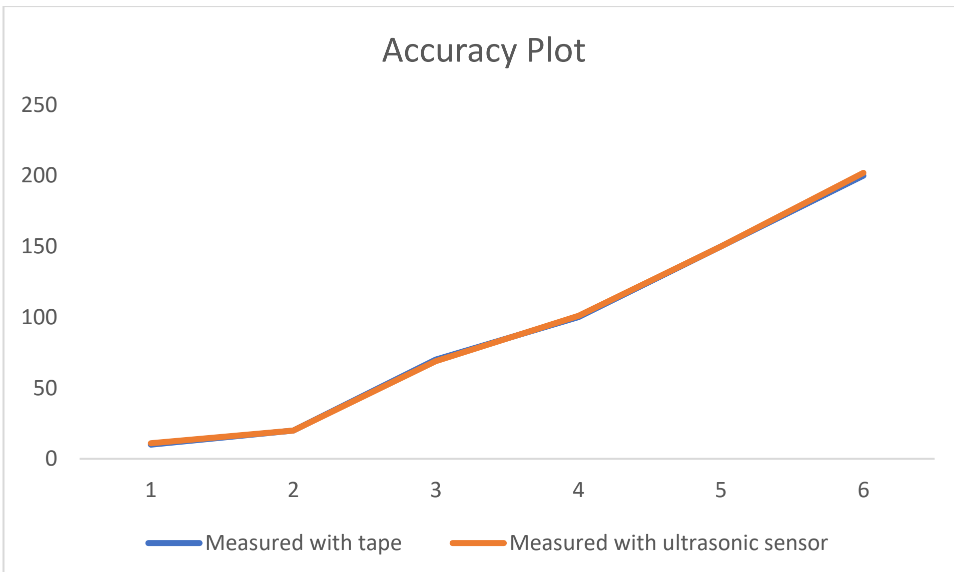
As shown in the graph above, the AI algorithm does not do much to improve the prediction by the AI module. In fact, it worsens the prediction and increases the error; however, this does not affect the project outcomes, because the system would rely on the math model for the daily automated functions.

5.2 Accuracy

This functionality test was carried out by cross-referencing sensor readings with premeasured values after calibration.

Table 20 Accuracy test results

Premeasured length	Device measured distance
10cm	11.2cm
20cm	20.7cm
70cm	68.88cm
100cm	101cm
150cm	150cm
200m	203cm



5.3 Durability

5.3.1 Drop test

This test was designed to measure how sturdy the product and the case was. The methodology involved was to drop the assembled product from some heights and examine damages to the case and the circuit.

The next test was the liquid-proof test; this test was carried out to ensure the casing was watertight, hence prevent damages from occurring when it falls into the fuel.

5.4 Efficiency

The efficiency test involved a series of experiments designed and carried out to determine the optimum conditions under which the system would be at maximum efficiency. This probe was qualitative by nature. However, numerical ratings could be ascribed to qualify how good or bad the performance is. The principal test, however, was to vary data size and frequency and monitor the network charges. For this test, each variation of data size and transmission frequency was carried out using a fixed amount of prepaid airtime – the time taken to exhaust the airtime was then measured and recorded. The desired length was obtained by either truncating the given data points or padding them with zeros. For the following test, a fixed amount of 10 Peswas (0.02 cents) was used for each iteration

Table 21 Transmission efficiency experiment results (transmission frequency)

Transmission frequency (points per second)	**Data length (Number of Digits)**	**Time to Exhaust 0.1 cedis Airtime (days)**
0.5	4	3.3
1	4	3.1
10	4	2.55
30	4	1.8

Table 22 Transmission efficiency experiment results (data point length)

Transmission frequency (points per second)	**Data length (Number of Digits)**	**Time to Exhaust 0.1 cedis Airtime (days)**
1	2	5.3
1	5	5.3
1	10	5.2
1	15	5.1

Upon reviewing the data in the tables above, it could be deduced that as the size of the data point increases the higher the cost of transmission. Also, the higher the transmission frequency, the larger the transmission cost given the length of the data point is held constant.

Speed:

This measure response of the entire system and the capacity to handle large volumes of data efficiently and effectively. The areas of concern when it comes to speed are;

- data transmission from the sensor
- data insertion into the database
- speed at which web app updates to reflect new data

Data transmission from the sensor: The net volume of data sent per hour is 7200 data points per hour. The Broker can handle such large volumes of data; however, it can be adjusted to meet the user's needs.

Data Insertion into the database: The net volume of data that is inserted into the database tables per hour is 7200 data points at an average rate of 2 data points per second. All tests indicate that the database is capable, and the insertion script is capable of handling such a large influx of data.

The speed at which web app updates to reflect the new data: This takes into account the speed at which the web app responds to real-time changes in the tank. This criterion includes the graph's response and how rapidly the rendered database table updates.

Chapter 6: Conclusion

The impact of IoT and smart systems are gradually being recognized and developed all around the world, and it would be unfortunate if Ghanaian industries are left out of the trend. As mentioned throughout the length of this paper, the implications of the Internet of Things are far-reaching and are capable to shaping the future of local industries, especially the Oil and Gas industry and as shown in this paper, IoT can be used to bolster the efficiency of local industries thereby increasing economic output. A low cost, energy efficient, and accurate IoT system would significantly impact the landscape of the middle stream and downstream aspects of the Oil and Gas industry in Ghana.

6.0 Summary

The system designed in this project is an efficient means to monitor conditions within a holding tank. It fuses affordable and accessible technology to yield a system capable of local oil and Gas industry. The systems chiefly consist of a sensor, some MCUs, power supply and a GSM module that makes up the main hardware composition. These discrete components are assembled in a manner which demands low power consumption, thus prolonging battery life. The other aspect of the IoT system is the software aspect; this involves several programming scripts, and data structures integrated to give device commands, a web interface, and a predictive algorithm. This software, although unsophisticated would offer a blueprint for further development of IoT within the Ghanaian context.

In this paper, a series of tests were carried out to ensure the proposed IoT system meets all requirements standards and is durable. Although the system has not entirely ready to be rolled out for public use, it presents a framework within IoT could be applied in a local context. Another point worth noting is that in constructing the predictive algorithm, a systems dynamic approach was taken to obtain the necessary parameters needing for time and rate calculations. The outcomes of the above tests are indicative of the fact that the concept of an artificially intelligent IoT system is feasible. Concordantly, I can be concluded that this project was generally successful, with 82% of milestones were achieved; 60% of tests were carried out, 70% of expected results were confirmed and 90% of requirements were met.

6.1 Limitations

The AI algorithm did not perform as expected; in most test cases the AI algorithm increased the error of the prediction by an average of 31.2%. This affected the final deliverable of the project; however, this did not detract from the impact and scope of the project.

Also, the experimental logistics fell short and failed to perform as expected. Especially, the tank's valve; this prevented controlled outflow and consequently affected the experimental readings.

6.2 Future works

As mentioned in the preceding chapters, the AI algorithm is not very sophisticated. Therefore, it may not make full use of datasets provided, or it may wrongly influence the prediction. This issue may be tackled in future works that explore the use of AI and IoT to solve problems within local industries. Also, the functions of the system could be extended by adding more sensors to the sensor suite, thereby meeting a larger demand for a more diverse user pool.

Bibliography

[1] F. Obeng-Odoom, "Global political economy and frontier economies in Africa: Implications from the oil and gas industry in Ghana.," *Energy Research & Social Science*, vol. 10, no. 6, pp. 41-56, 2015.

[2] C. A. Willink, "U.S. Patent No. 6,962,199," Patent and Trademark Office, Washington, DC, 2005.

[3] S. D. T. Kelly, N. K. Suryadevara and S. C. Mukhopadhyay, "Towards the implementation of IoT for environmental condition monitoring in homes," *IEEE sensors journal*, vol. 10, no. 13, pp. 3846-3853, 2013.

[4] J. Gubbi, R. Buyya, and M. Palaniswami, "Internet of Things (IoT): A vision, architectural elements, and future directions.," *Future generation computer systems*, vol. 7, no. 29, pp. 1645-1660, 2013.

[5] I. Lee and K. Lee, "The Internet of Things (IoT): Applications, investments, and challenges for enterprises.," *Business Horizons*, vol. 4, no. 58, pp. 431-440, 2015.

[6] S. Viswanath, M. Belcastro, J. Barton, B. O'Flynn, N. Holmes, and P. Dixon, "LOW-POWER WIRELESS LIQUID MONITORING SYSTEM USING ULTRASONIC SENSORS," *International Journal on Smart Sensing & Intelligent Systems*, vol. 8, no. 1, pp. 1-18, 2015.

[7] S. Kindawi, A. Mahdi and M. A. J. R. Al-Aameri, "Remote Monitoring and Automatic Controlling of Water Tank Level Based on GSM".

[8] M. Predko and M. Predko, Handbook of microcontrollers, Colorado: McGraw-Hill, Inc, 1998.

[9] P. Marian, Hc-sr04 datasheet, Electro Schematics, 2018.

[10] L. Steels and R. Brooks, The artificial life route to artificial intelligence: Building embodied, situated agents., Routledge, 2018.

[11] S. S.P, "A review of system dynamics models applied in transportation," *Transportmetrica B*, vol. 2, no. 2, pp. 1-23, 2014.

[12] T. M. L. Mounika and K. P. A. Priya, New Sensor Based Communication for Environmental Monitoring., 2008.

[13] G. D. Obikoya, "Design, construction, and implementation of a remote fuel-level monitoring system.," *EURASIP Journal on Wireless Communications and Networking*, vol. 1, no. 76, pp. 1-33, 2014.

[14] O. B. A. T. E. S. M. Vermesan, "Internet of robotic things: converging sensing/actuating, hypoconnectivity, artificial intelligence and IoT Platforms," in *Cognitive Hyperconnected Digital Transformation*, River Publishers, 2017, pp. 97-155.

[15] D. E. O'Leary, "Artificial Intelligence and Big Data," *IEEE intelligent systems*, vol. 28, no. 2, pp. 96-99, 2013.

[16] R. R. S. T. Somayya Madakam, "Internet of Things (IoT): A Literature," *Journal of Computer and Communications,* vol. 3, pp. 164-173, 2015.

[17] J. Blum, Exploring Arduino: tools and techniques for engineering wizardry, John Wiley & Sons, 2013.

[18] M. P. Alessio Carullo, "An Ultrasonic Sensor for Distance Measurement in," *IEEE Sensors Journal*, vol. 1, no. 02, pp. 143-148, 2001.

Appendix

Appendix 1: The assembled device

Figure 17 Image of the assembled device

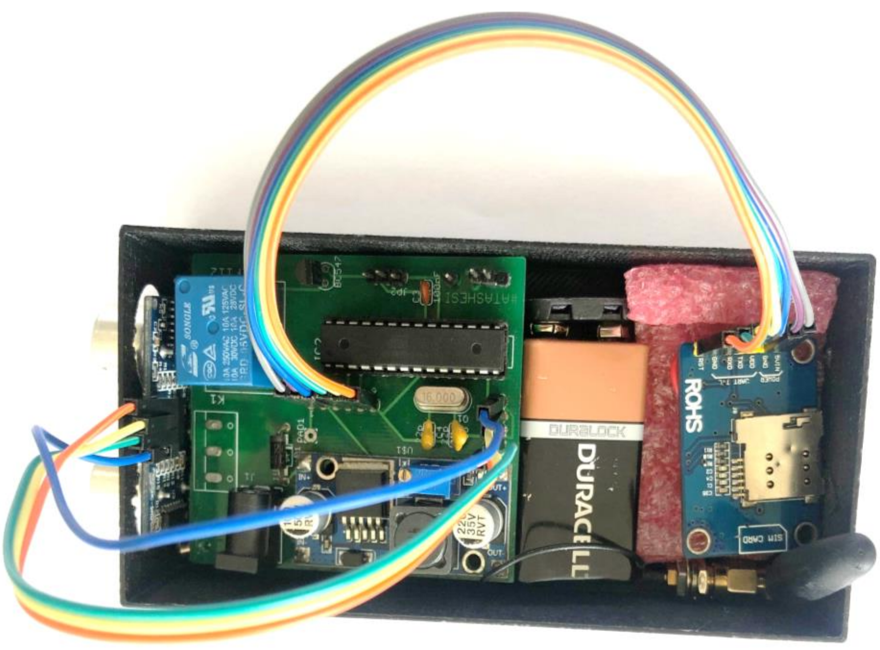


Figure 18 Image of the connected circuit

Appendix 2: Rendering the Database tables

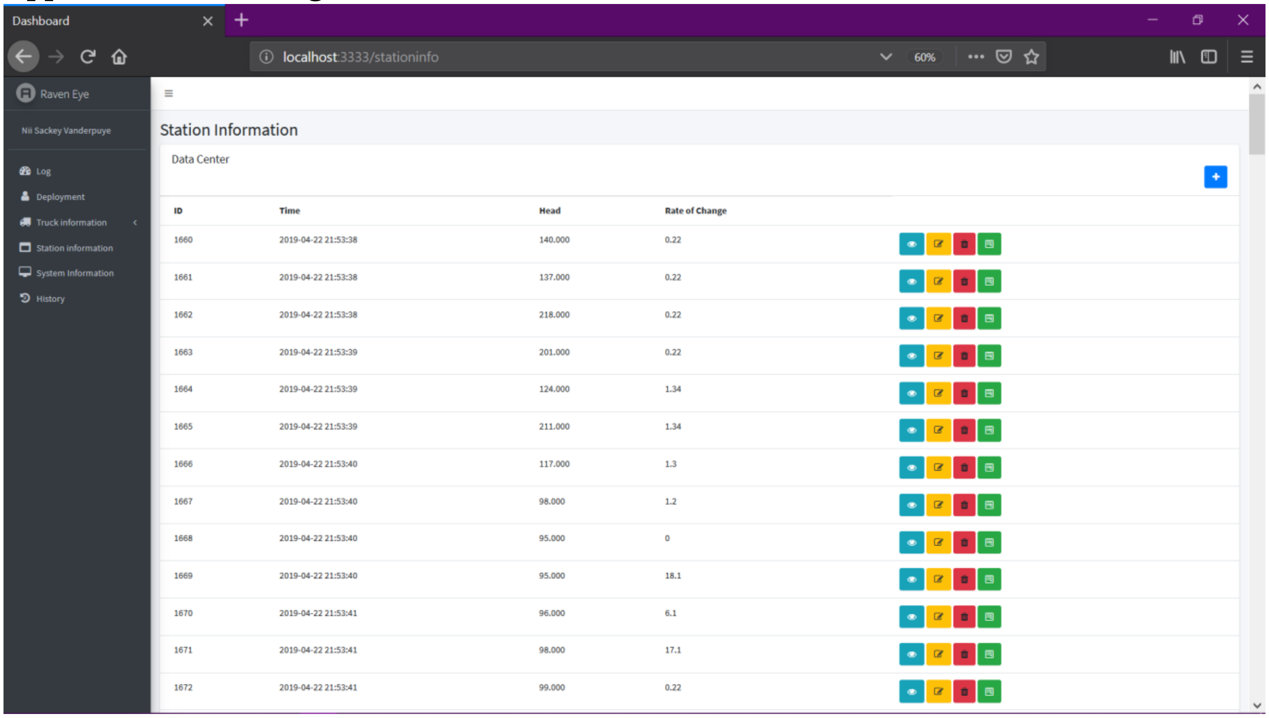


Figure 19 Front end rendition of SQL database

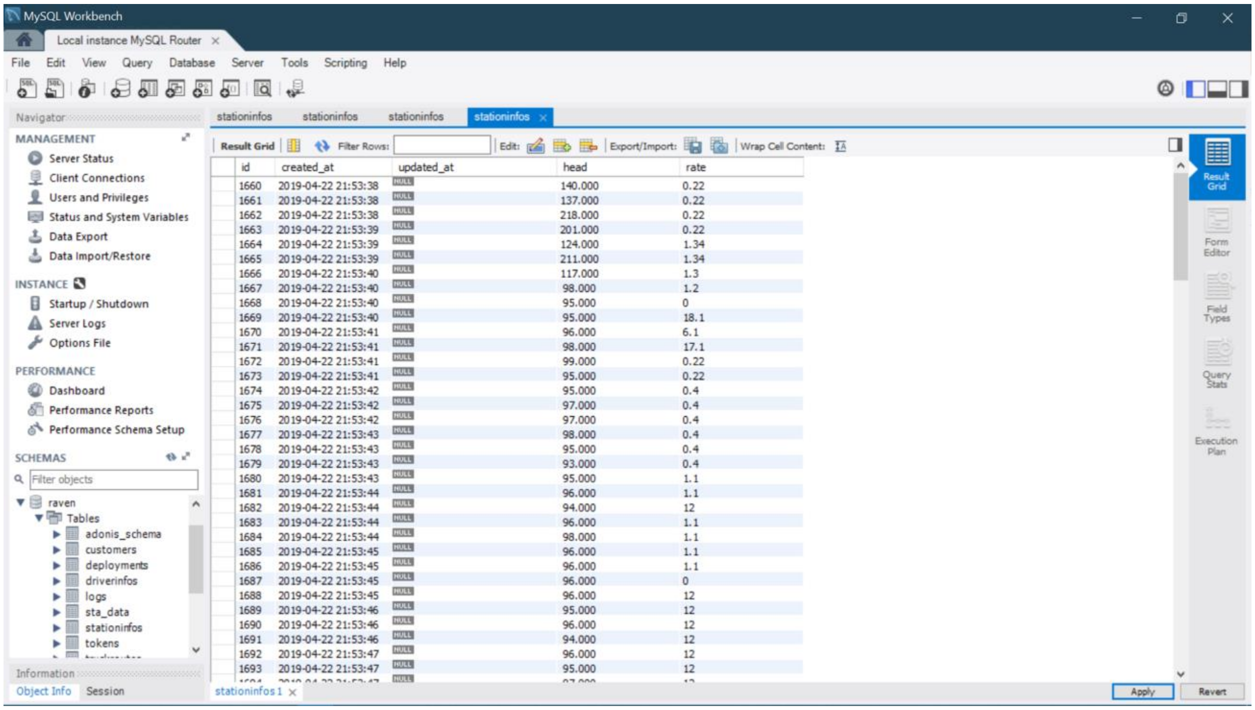


Figure 20 Backend rendition of part of the database

Appendix 3: Database insertion script

```python
import paho.mqtt.client as mqtt
import mysql.connector
import datetime
global belt
belt = ''
global prev
prev = 0
mydb = mysql.connector.connect(
    host="localhost",
    user="root",
    passwd="EvaPeron!18553",
    database="raven"
    )

mycursor = mydb.cursor()
    #print(mydb)


#mqtt networking methods
def connected(client, userdata, flags, rc):
    print("Connected with result code "+str(rc))
    client.subscribe('jay/init')
    #client.subscribe('RollE_MKII/throttle')

def message_in(client,userdata,msg):
    print("Time: " +  "    Topic: "+msg.topic+"        Message: "+str(msg.payload))
    belt = int(msg.payload)
    x = datetime.datetime.now()
    rate = (30-int(belt))/(x-x0)
    sql = "INSERT INTO stationinfos (head,created_at, updated_at, rate) VALUES (%s,%s, %s, %s)"
    val = (belt,x, x, rate)
    mycursor.execute(sql, val)
    mydb.commit()
    prev = belt
    x0 = x
    return prev


def networking_init():
    #instantiates mqtt client connects to broker and awaits messages in endless loop

    client = mqtt.Client('modem')
    client.on_connect = connected
    client.on_message = message_in
    client.connect('broker.hivemq.com')
    print('setting up done, starting ...')
    client.loop_forever()

networking_init()
```

Figure 21 Snippet of code used to populate the Database

Appendix 4: Experimental setup

Figure 22 Image of the experimental setup

Appendix 5: Relevant Pugh Matrices

Table 23 Pugh chart for sensor selection

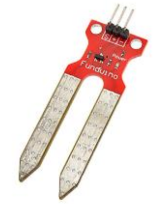
Sensor	Weight	LLS Fuel level sensor	Ultrasonic	Pressure sensor	Moisture sensor (Extended version)	Inflow modeling
Parameters						
Baseline	-	0	0	0	0	0
Cost	4	-	0	0	+	-
Size	4	+	+	+	0	0
Suitability case	5	-	+	-	0	0
Cable	3	0	+	+	-	-
Material	2	+	+	+	+	0
Range	5	0	+	0	0	0
Resolution	3	+	+	0	+	0
Voltage supply	4	0	+	+	+	-
Data transmission rate (b)	5	0	+	+	+	0
Power	3	0	+	+	+	0
Grade		8	45	21	31	2
Rank		4	1	3	2	5

Table 24 Pugh chart for communication device selection

	Base	Weight of criteria	GSM(Arduino)	WIFI (module)	ATM	Lease lines
Cost	0	4	+	-	+	0
Efficiency	0	3	0	+	+	0
Complexity	0	5	+	-	0	0
Speed	0	3	+	+	+	+
Grade			12	-3	10	3
Rank			1	4	2	3