Technical Report

MAKERERE UNIVERSITY

Smart Water Level Monitoring System for Tanks/Drums

Prepared by

GROUP-3

NO	NAME	STUDENT NUMBER	REG. NO
1	KIGOZI ALLAN	2400725792	24/U/25792/PS
2	KEITH PAUL KATO	2400726593	24/U/26593/EVE
3	ECONGU PAUL	2400722570	24/U/22570
4	BWIRE RODNEY	2400714889	24/U/14889/PS
5	CHELIMO EMMA	2400714938	24/U/14938/PSA

Assessor/Mentor: **Mr. Asiimwe Paddy Junior**

Course: **CSC 1304 Practical Skills Development**

Date: **28TH July 2024**

Contents

Definitions, Acronyms and Abbreviations

- ***Arduino IDE:*** *Integrated Development Environment for Arduino*
- ***API:*** *Application Programming interface*
- ***FR:*** *Functional Requirements*
- ***HC-SR04:*** *Ultrasonic Sensor (Generic)*
- ***IEEE:*** *Institute of Electrical and Electronic Engineers*
- ***I2C:*** *Inter-integrated Circuit*
- ***IoT:*** *Internet of Things*
- ***LCD:*** *Liquid Crystal Display*
- ***MCU:*** *Microcontroller Unit*
- ***PR:*** *Performance Requirements*
- ***SRS:*** *Software Requirement Specifications*
- ***S.W.L.M:*** *Smart Water Level Monitoring System*
- ***SMS:*** *Short Message Services*
- ***ThingSpeak:*** *An Open-Source IoT analytics platform service*

Abstract

In Uganda today, Water scarcity and inefficient usage remain pressing challenges, especially in peri-urban and rural communities where reliance on rainwater harvesting tanks and underground storage drums is widespread. Many households, schools, and health centers depend on manually checking water levels, leading to overflows, dry tanks, and poor water planning. This project introduces the Smart Water Level Monitoring System, an IoT- based solution designed to monitor and manage water levels in storage containers, offering real-time insights and alerts to users.

The System integrates an Ultrasonic Sensor and a microcontroller for example, Arduino microcontroller board and a nano ESP32 to detect water levels and wirelessly transmit data to a mobile-friendly dashboard such as thing speak. Users can track current levels, receive alerts via SMS when water is low (e.g., below 20%), and review usage patterns to avoid wastage. For instance, a rural health center in Mukono could use the system to ensure its rainwater tank doesn't run dry during dry spells. Similarly, Urban households in Kampala relying on National Water trucks can monitor stored water between deliveries to avoid shortages.

By automating monitoring and promoting water use, the SWLMS addresses a key inefficiency in Uganda's water resource management. The system is cost-effective, energy-efficient, and adaptable to off-grid environments, making it suitable for wide deployment across the country. In doing so, it supports national goals of water conservation, service delivery, and sustainable development.

Introduction (max. 2 pages)

In an era increasingly defined by technological innovation and the imperative for sustainable resource management, the Smart Water Level Monitoring System emerges as a timely and practical solution. This project, undertaken by GROUP-3 as a culminating endeavor for the CSC 1304 Practical Skills Development course at Makerere University, represents a significant step towards addressing critical water management inefficiencies prevalent in many Ugandan communities. From bustling urban centers to remote rural homesteads, the reliance on traditional, often manual, methods for monitoring water storage tanks have long presented a paradox: a precious resource, yet its management is frequently characterized by avoidable waste and unexpected shortages.

This technical report meticulously details the journey from conceptualization to the realization of a functional prototype. It delves into the architectural intricacies, the selection and integration of various hardware components, and the meticulous crafting of the embedded software that breathes life into the system. Beyond merely presenting a functional device, this report aims to articulate the profound impact our Smart Water Level Monitoring System can have on daily lives, promoting water conservation, enhancing convenience, and fostering a more data-driven approach to resource utilization. We believe this document not only fulfills the academic requirements of our course but also lays a robust foundation for future research and development, contributing meaningfully to the broader discourse on smart infrastructure and sustainable living in Uganda and beyond.

1.1. User Challenge

The pervasive challenge of water management in Uganda, particularly in peri-urban and rural areas, stems significantly from the reliance on manual methods for monitoring water levels in storage tanks and drums. This manual approach is inherently inefficient and prone to several critical issues. Households, schools, and health centers frequently experience either wasteful overflows when tanks are overfilled or critical shortages when tanks run dry unexpectedly. This lack of real-time visibility leads to poor water planning, increased labor for physical checks, and significant water wastage, directly impacting resource sustainability and daily operations. The problem is exacerbated in areas dependent on rainwater harvesting or infrequent water deliveries, where accurate knowledge of available water is paramount for effective conservation and usage. Our project directly confronts this challenge by automating the monitoring process, thereby mitigating these inefficiencies, and promoting more responsible water consumption.

1.2. Project Goals

The Smart Water Level Monitoring System (SWLMS) is conceived as a novel, self-contained embedded product designed to revolutionize water management practices. Its primary purpose is to provide real-time, accurate, and accessible water level information for storage tanks and drums, coupled with intelligent alerting capabilities. The innovativeness of the SWLMS lies in its integration of affordable IoT technologies to transform a manual, error-prone process into an automated, efficient, and proactive system.

This system directly addresses the user challenges outlined in Section 1.1 by:

a) ***Automating Monitoring:*** *Eliminating the need for manual checks, reducing labor and human error.*

b) ***Preventing Overflow:*** *Providing timely "high level" alerts to prevent water wastage.*

c) ***Mitigating Shortages:*** *Delivering "low level" alerts to ensure users are aware of impending scarcity, allowing for proactive refilling or conservation measures.*

d) ***Enabling Data-Driven Planning:*** *Logging historical data to* ***ThingSpeak,*** *allowing users to analyze consumption patterns and make informed decisions about water usage and replenishment schedules.*

The SWLMS aims to be a foundational module within a larger Internet of Things (IoT) ecosystem for smart home or smart building automation. It acts as a critical data acquisition node, interfacing with the immediate physical environment (the water tank) via an ultrasonic sensor. Beyond this, it communicates directly with the end-user through a local Liquid Crystal Display (LCD) for immediate visual feedback, and remotely via SMS notifications through an integrated GSM module for urgent alerts. Crucially, it interfaces with the cloud platform ***(ThingSpeak)*** *for robust data logging and remote monitoring, allowing users to access historical data and visualize trends from anywhere with internet access. While its current iteration is self-contained, its design incorporates interfaces for potential future integration with actuator systems, such as automated pump control, laying the groundwork for evolution into a comprehensive, closed-loop water supply management solution. This innovative approach ensures efficient resource management, contributes to environmental sustainability, and enhances convenience for users.*

1.3. Functional Requirements

The Smart Water Level Monitoring System is designed to perform a set of specific functions to achieve its objectives. These functional requirements are categorized into key operational areas for clarity.

1.3.1. *Sensing and Measurement*

a) ***FR1.1: Ultrasonic Measurement:*** *The system shall accurately measure the distance from the top of the tank to the water surface using an HC-SR04 ultrasonic sensor.*

b) ***FR1.2: Distance to Depth Conversion:*** *The system shall convert the measured distance into the actual water depth from the bottom of the tank, based on a user-calibrated total tank height.*

c) ***FR1.3: Volume Calculation:*** *The system shall calculate the approximate volume of water in the tank, in liters, utilizing the calculated water depth and predefined tank dimensions (e.g., radius for cylindrical tanks).*

1.3.2. *Local Display*

a) ***FR2.1: Real-time LCD Display:*** *The system shall continuously display the current water level (distance from sensor), water depth, and calculated volume (in liters) on a 16x2 I2C LCD screen.*

b) ***FR2.2: Display Update Frequency:*** *The LCD display shall be updated with fresh data at least every 2 seconds to provide near real-time feedback.*

1.3.3. *Remote Communication and Alerting*

1) ***FR3.1: Low Water Level Alert Trigger:*** *The system shall detect when the water depth falls below a predefined low threshold (e.g., 20% of total tank depth).*

2) ***FR3.2: High Water Level Alert Trigger:*** *The system shall detect when the water depth exceeds a predefined high threshold (e.g., 90% of total tank depth).*

3) ***FR3.3: SMS Notification for Low Level:*** *Upon detection of a low water level, the system shall send an automated SMS alert to a pre-configured mobile number (e.g., +256786178453) with a message indicating the low water status.*

4) ***FR3.4: SMS Notification for High Level:*** *Upon detection of a high-water level, the system shall send an automated SMS alert to the pre-configured mobile number with a message indicating the high-water status.*

5) ***FR3.5: GSM Module Initialization:*** *The system shall successfully initialize the GSM module and establish a stable connection to the mobile network for SMS services.*

1.3.4. *Cloud Integration*

a) ***FR4.1: GPRS Connectivity:*** *The system shall establish a General Packet Radio Service (GPRS) connection via the GSM module to access the internet.*

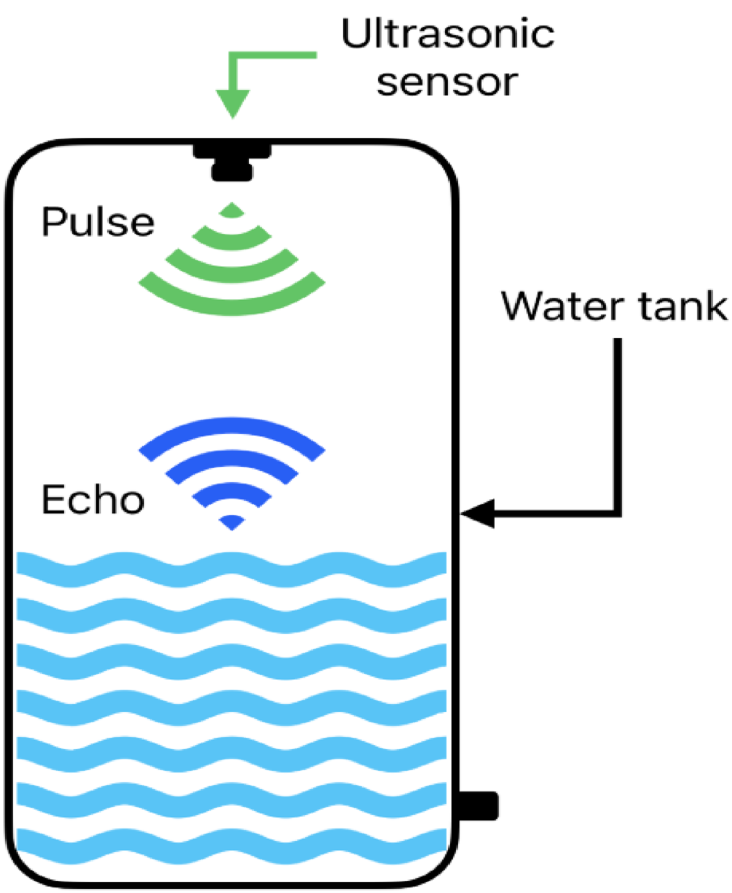
b) ***FR4.2: Thing Speak Data Upload:*** *The system shall periodically upload the measured water level data (distance, depth, and volume) to a designated Thing Speak channel using HTTP GET requests.*

c) ***FR4.3: Data Upload Frequency:*** *Data shall be uploaded to Thing Speak at least every 10 seconds to ensure consistent remote monitoring*

2. Project Results (max. 3 pages)

2.1. Product Design

The Smart Water Level Monitoring System's design follows a modular embedded system architecture, integrating sensing, processing, display, and communication functionalities. This design ensures robust operation and provides clear interfaces for each subsystem.



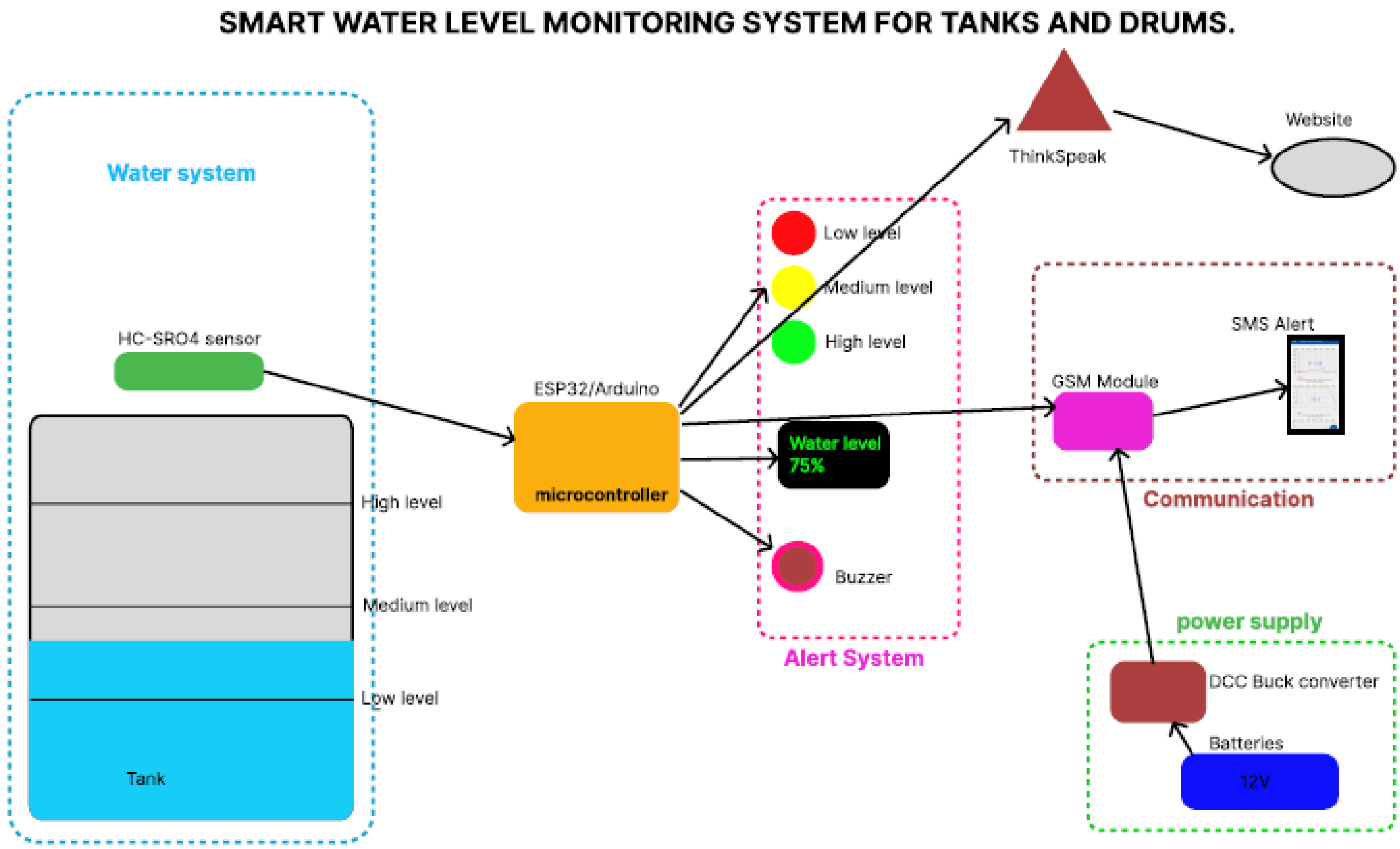
Partial Design Description

- *The **Generic HC-SR04 Ultrasonic** Sensor interfaced with a pair of **transducers** is a critical piece of our project.*
- *And is in our tank to gauge distance. Works by converting electronic signals into sound waves, inside of them is a Piezo Crystal which vibrates a plate when electricity passes through.*
- *Transmits Sound waves at a frequency we cannot hear (40,000Hz) known as Ultra sound that reflects off an object and gets detected by the receiver hence vibrating the plate and turning it into an electric signal.*
- *We use it by getting the speed of sound multiplied by the time it takes for the journey and then divide by two because we only need one distance.*
- *Trigger pin sends the wave, and the echo pin receives it, we import I2C Library & make other connections*

2.1.1. System Architecture

*The overall system architecture is depicted in the diagram below (refer to Figure 1). The core of the system is the **Arduino Uno microcontroller**, which acts as the central processing unit. It receives input from the **HC-SR04 Ultrasonic Sensor**, which is responsible for non-contact water level measurement. The processed data is then presented to the user locally via a **16x2 I2C LCD Display**. For remote interaction, a **GSM Module (SIM800L)** is integrated, enabling two-way communication. This module facilitates sending SMS alerts directly to the user's mobile phone and establishing an internet connection (GPRS) to upload data to the **ThingSpeak cloud platform**. The entire system is powered by a **10V DC Power Supply**.*

2.1.2. *Figure 1: System Architecture Diagram*



2.1.3. Design

Prototype Method: *The overall nature of our project is centered around the development of a functional prototype utilizing Internet of Things (IoT), the prototyping model emerges as the most fitting approach for the System Development Life Cycle (SDLC) methodology. Diverging from the linear progression and predefined stages of development observed in the waterfall SDLC methodology, the prototyping methodology embraces a dynamic trial-and-error approach. This iterative method continues until the final software or system aligns precisely with the specified requirements. This model establishes a robust foundation for the prototype, subjecting it to testing and successive refinement, particularly well-suited for projects demanding continuous incorporation for new insights and updates throughout the development journey.*

2.1.4. Requirements

The project's primary objective is to design and implement a S.W.L.M.S. The system aims to provide real-time monitoring of water levels in household tanks and deliver immediate alerts to users when levels fall below a certain threshold. The project encompasses two key phases which are the hardware configuration and software development, aligning with the principles of IoT.

2.1.5. Hardware Components and Interfacing

The selection of hardware components was driven by a balance of cost-effectiveness, availability, and functional requirements.

1) ***Arduino Uno:*** *Chosen for its simplicity, extensive online resources, and sufficient processing power and I/O pins for this application. It handles sensor data acquisition, calculations, LCD control, and serial communication with the GSM module.*

2) ***HC-SR04 Ultrasonic Sensor:*** *This sensor provides accurate and reliable non-contact distance measurement, crucial for monitoring water levels without direct contact, thus preventing corrosion or contamination. Its TRIG and ECHO pins are connected to Arduino digital pins 9 and 10, respectively.*

3) ***SIM900A GSM Module:*** *Essential for remote communication, this module enables sending SMS alerts and connecting to the internet via GPRS. It communicates with the Arduino using SoftwareSerial, typically connected to Arduino digital pins 2 (RX) and 3 (TX).*

4) ***16x2 I2C LCD Display:*** *Provides a clear local interface for real-time water level data. The I2C interface simplifies wiring to the Arduino (SDA to A4, SCL to A5), reducing the number of pins required.*

5) ***Power Supply:*** *A stable 5V DC power supply is critical for the reliable operation of all components, especially the GSM module which can draw significant current during transmission bursts.*

Hardware Requirement List:

a. ***Arduino Uno R3 Microcontroller Board***
b. ***Ultrasonic Sensor – HC-SR04 (Generic)***
c. ***5mm LED: Green***
d. ***5mm LED: Orange***
e. ***5mm LED: Red***
f. ***Buzzer***
g. ***Male to Female Jumper Wires***
h. ***Male to Male Jumper Wires***
i. ***Female to Female Jumper Wires***
j. ***9V Battery***
k. ***USB-A to B cable***
l. ***16*2 LCD with I2C for Arduino***

m. 5V DC Power Supply

n. DC-DC Buck Converter (Step Down)

2.1.6. Software Design

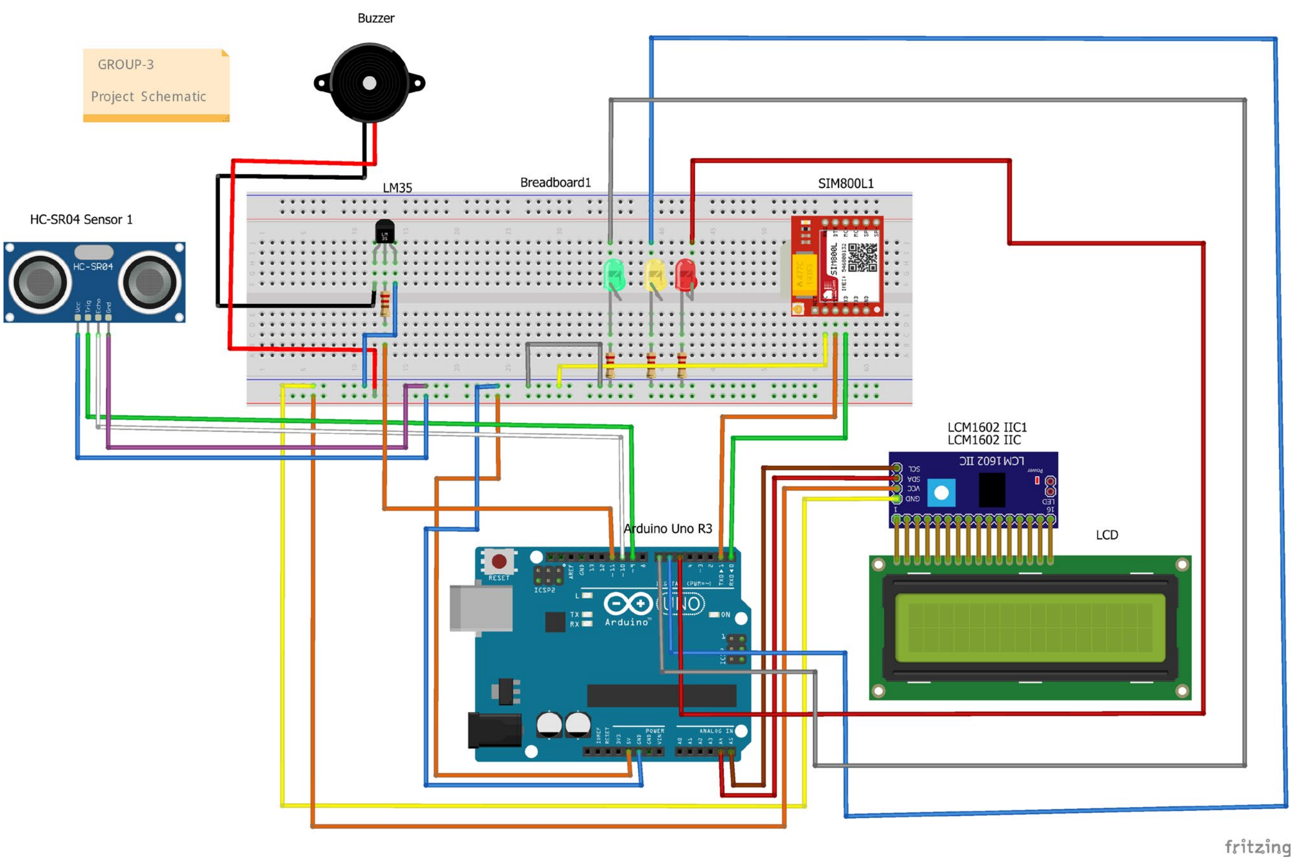
*The software is developed in Arduino C/C++ using the Arduino IDE. The firmware **(Keith2.ino)** is structured to perform continuous monitoring and communication tasks. It utilizes standard libraries such as Wire.h for I2C communication with the LCD and **LiquidCrystal_I2C.h** for LCD control. Software serial is used for communication with the GSM module. The core logic involves reading sensor data, performing calculations, updating the LCD, checking thresholds, and managing GSM-based communication for SMS and ThingSpeak uploads.*

Software Requirement List

a) Arduino Integrated Development Environment

b) Thing Speak Application

2.1.7. *Figure 2: Schematic diagram of proposed methodology*



2.1.8. Figure 2 Description:

Arduino Uno R3 Pinout Analysis for Smart Water Level Monitoring System

This table details the connections and functions of the Arduino Uno R3 pins as observed in your project's schematic diagram.

Digital Pins

Pin	Pin Name	Function	Type	Description
2	D2	SIM800L (GSM) RX	Digital I/O	Connected to the TX pin of the SIM800L module for serial communication (Arduino's RX for Software Serial).
3	D3	SIM800L (GSM) TX	Digital I/O	Connected to the RX pin of the SIM800L module for serial communication (Arduino's TX for Software Serial).
6	D6	Red LED Control	Digital I/O	Controls the Red LED, indicating a specific water level status (LOW).
7	D7	Yellow LED Control	Digital I/O	Controls the Yellow LED, indicating a specific water level status (Normal)
8	D8	Green LED Control	Digital I/O	Controls the Green, LED, indicating a specific water level status (Maximum)
9	D9	HC-SR04 Trigger	Digital I/O	Outputs a pulse to trigger the ultrasonic sensor for distance measurement.
10	D10	HC-SR04 Echo	Digital I/O	Receives the echo pulse from the ultrasonic sensor to calculate distance.
11	D11	Buzzer Control	Digital I/O	Controls the buzzer, for audible alerts (Alarm).

Analog Pins

A0	A0	LM35 Temperature Sensor	Analog Input	Read analog voltage from the LM35 temperature sensor to measure temperature
A4	A4	LCM1602 I2C SDA	Analog Input	I2C Data Line for communication with the 16*2 I2C LCD display
A5	A5	LCM1602 I2C SCL	Analog Input	I2C Clock line for communication with the 16*2 I2C LCD display

Power Pins

Total (9V)	Total (9V)	Power Supply	Power Input (For Additional Power) Power Output	Provides 5V power to the ultrasonic sensor, LM35, LEDs, GSM module, and LCD display.
GND	GND	Ground	Ground	Common ground connection for all components

2.2. Product Functionality and Screenshots

The Smart Water Level Monitoring System operates through a well-defined sequence of functions, ensuring continuous monitoring and timely alerts.

2.2.1. Operational Flow

The system's functionality can be summarized by the following operational flow (refer to the flowchart in Figure 3 for a visual representation):

2.2.2. Figure 3: Flow Chart

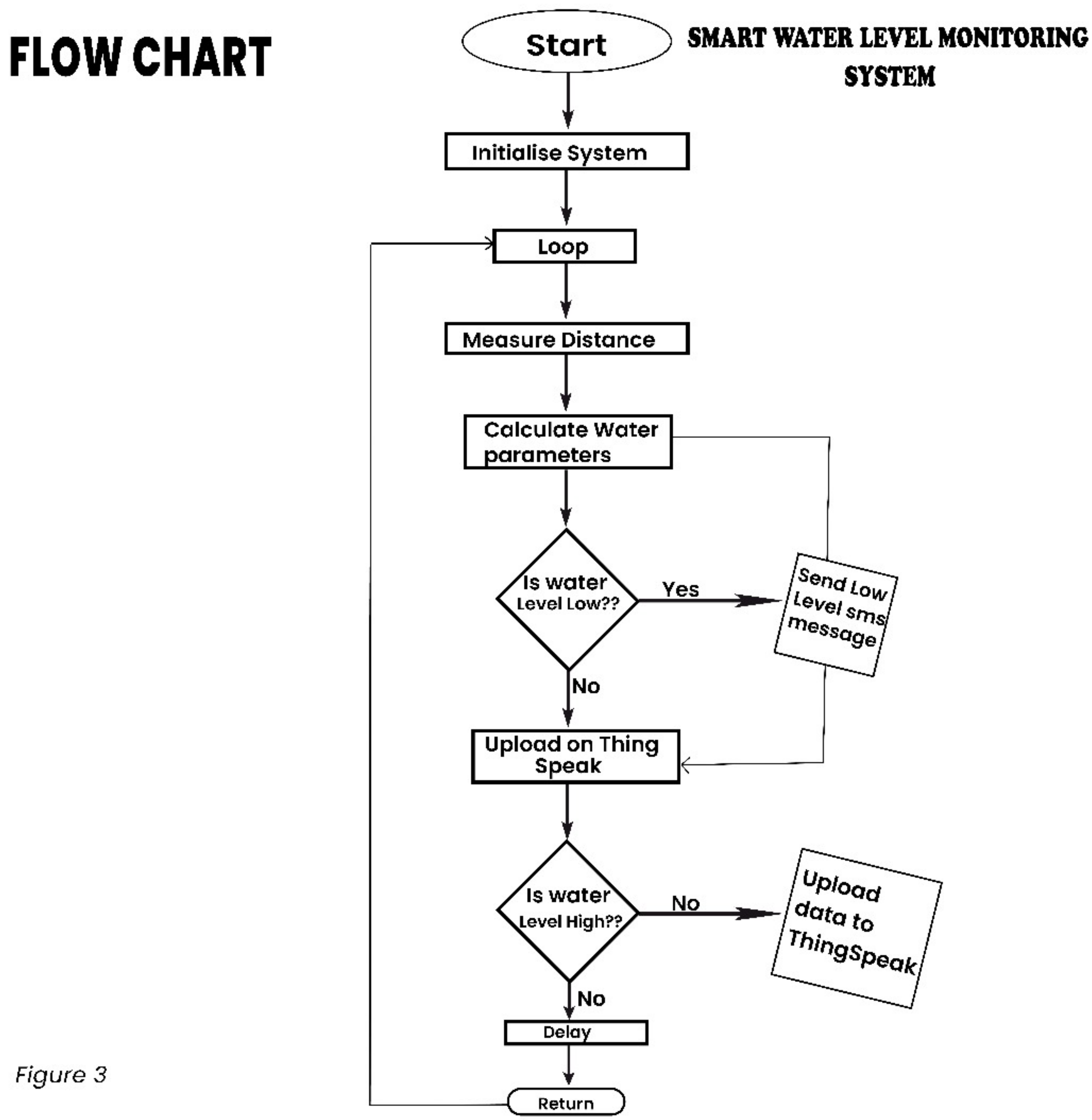


Figure 3

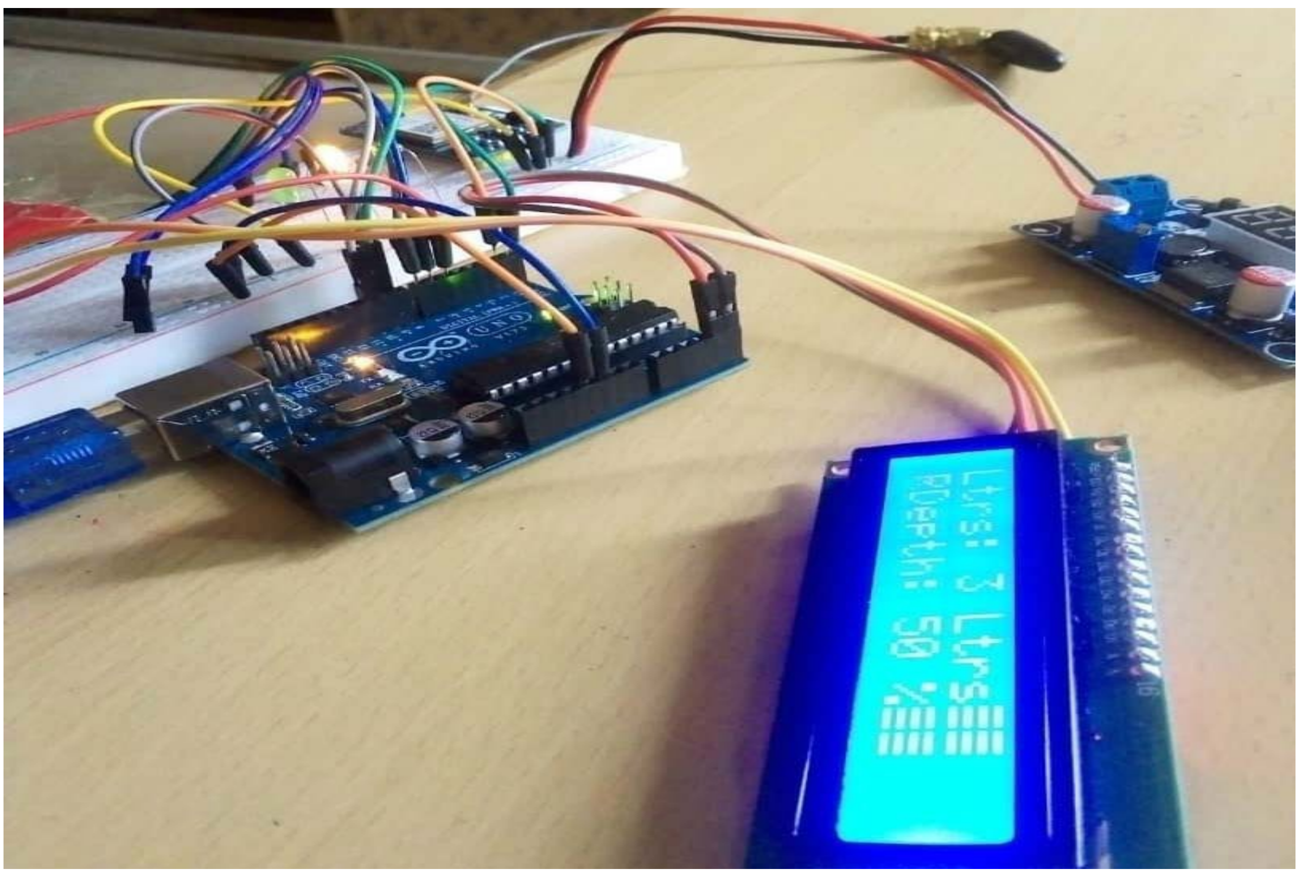
Description

*Upon power-on, the system undergoes a **System Initialization** phase, where the LCD and GSM modules are configured. Following initialization, the system enters a continuous loop. In each cycle, it first **Senses the Water Level** using the ultrasonic sensor. The raw distance data is then used to **Calculate Water Depth and Volume**. This information is immediately displayed on the **Local LCD Screen**, providing real-time feedback. The system then performs two critical decision checks: first, if the **Water Level is Low**, and second, if the **Water Level is High**. If either condition is met, an appropriate **SMS Alert** is triggered and sent via the GSM module to the pre-configured phone number. Concurrently, the system checks if it's **Time for a ThingSpeak Upload**. If so, it **Sends the Data to ThingSpeak**, updating the cloud dashboard. Finally, a brief **Delay** is introduced before the loop restarts, ensuring stable operation.*

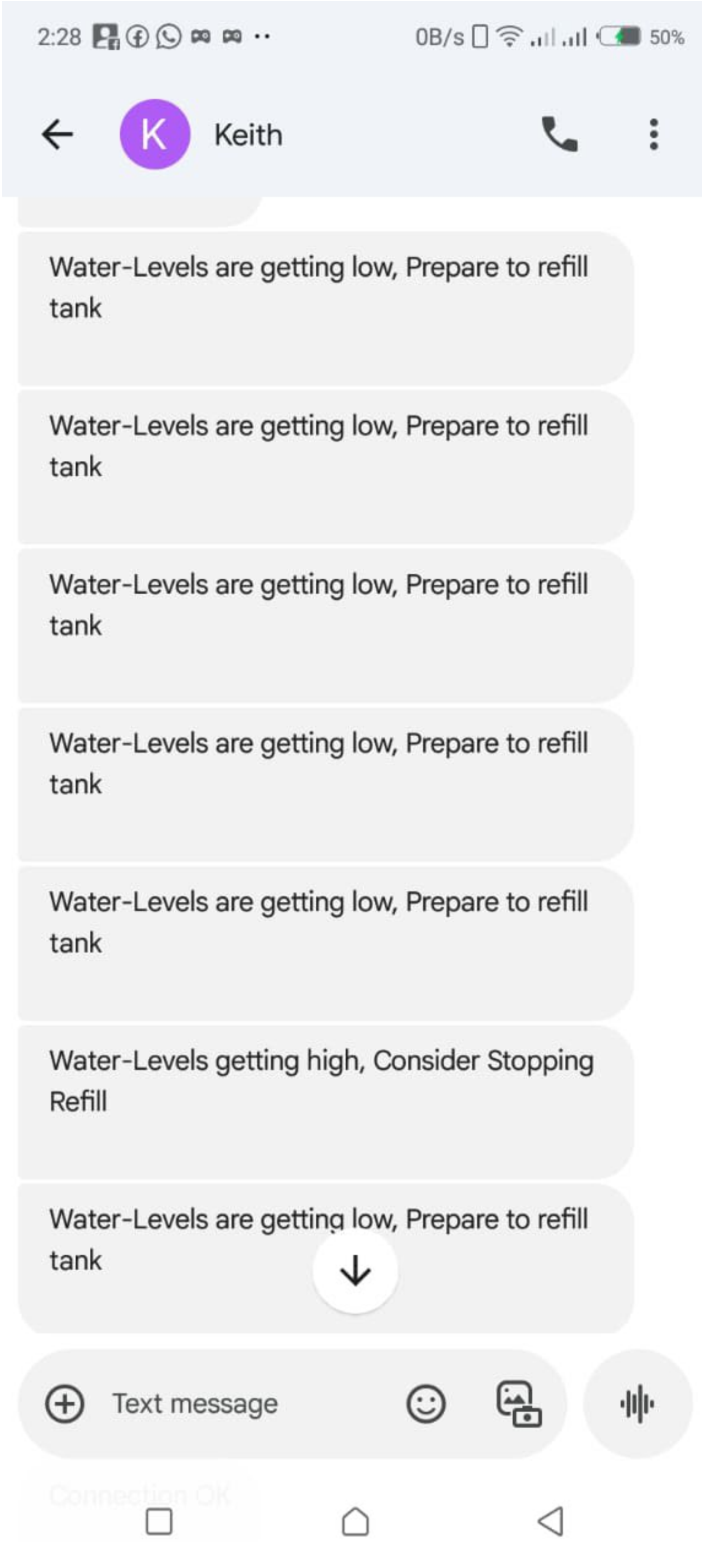
2.2.3. *Key Functionalities in Action*

1) ***Real-time Level Monitoring:*** *The system continuously measures water levels.*

- ***Description:*** *This screenshot shows the 16x2 LCD display, providing live updates of the water level. It typically displays the distance from the sensor to the water surface, the calculated water depth from the tank's bottom, and the estimated volume in liters. This immediate visual feedback is crucial for local monitoring.*



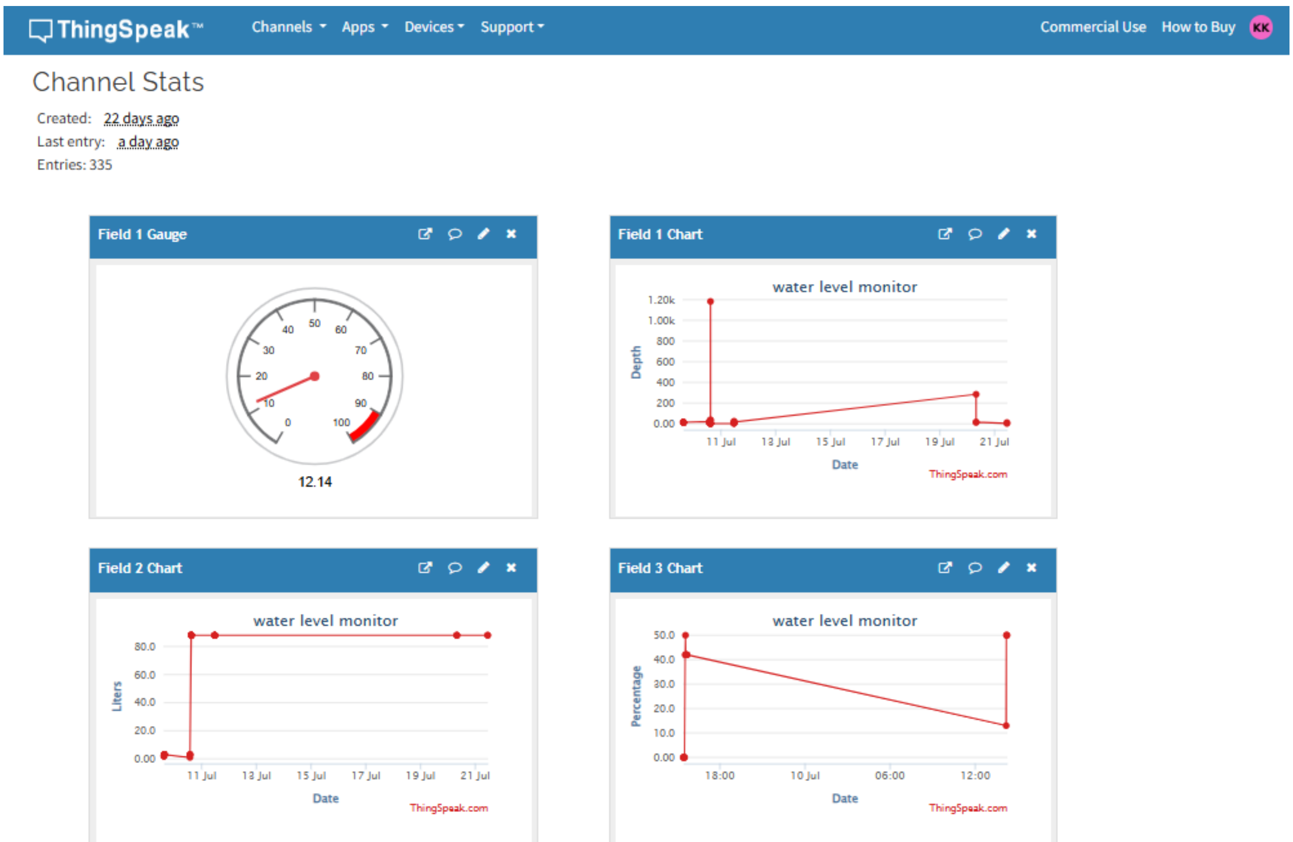
2) ***SMS Alerting:*** *The system sends timely alerts for critical water levels.*


2:28
0B/s
50%
Keith
Water-Levels are getting low, Prepare to refill tank
Water-Levels are getting low, Prepare to refill tank
Water-Levels are getting low, Prepare to refill tank
Water-Levels are getting low, Prepare to refill tank
Water-Levels are getting low, Prepare to refill tank
Water-Levels getting high, Consider Stopping Refill
Water-Levels are getting low, Prepare to refill tank
Text message

- ***Description:*** *This image illustrates an SMS notification received on a mobile phone when the water level drops below the predefined low threshold. Similar alerts are sent for high water levels, ensuring users are informed even when away from the tank.*

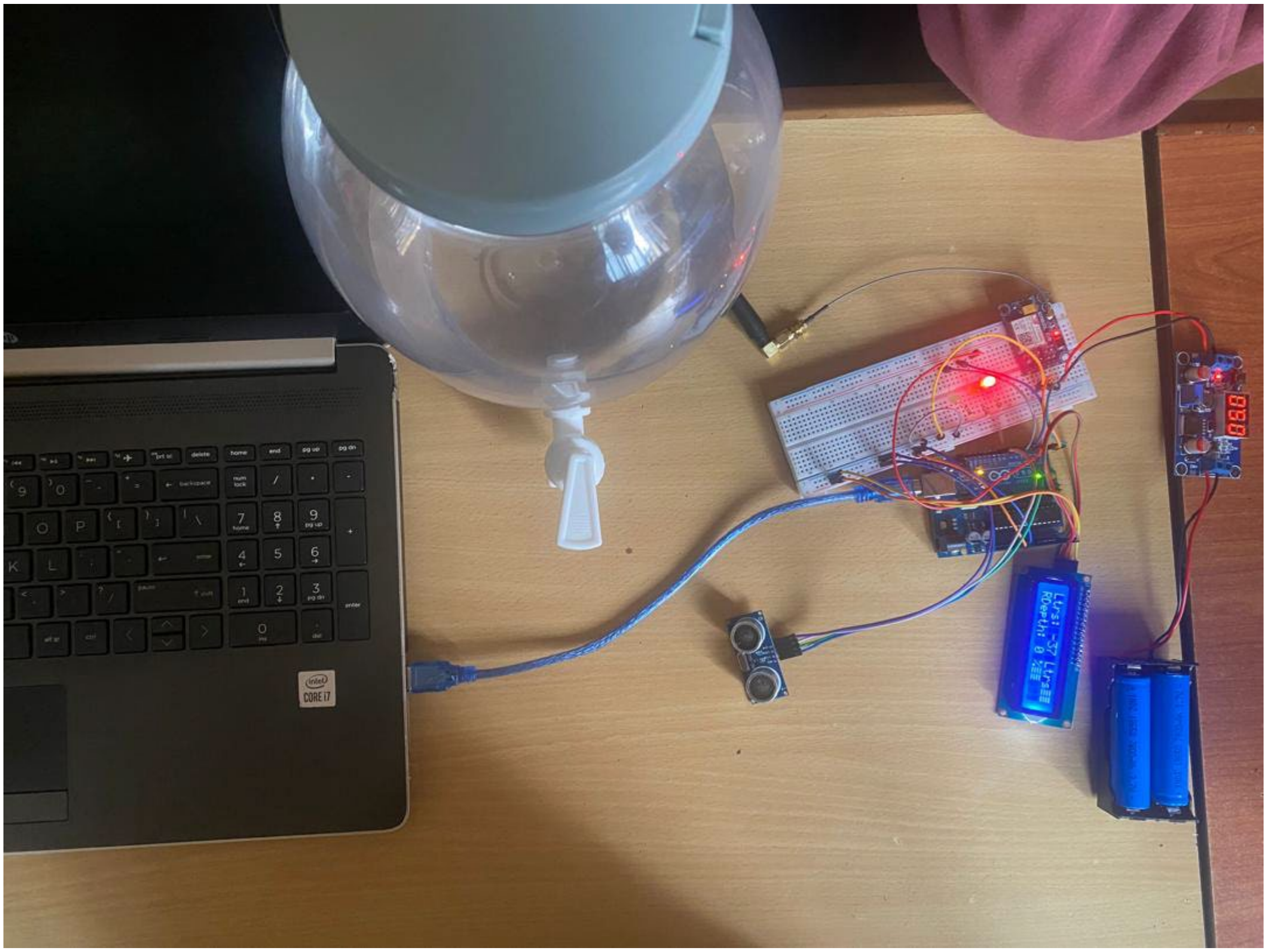
3) ***Cloud Data Logging & Visualization (ThingSpeak):*** *Data is sent to the cloud for remote access and historical analysis.*

- ***Description:*** *This screenshot displays the ThingSpeak dashboard, where the system uploads water level data. Users can access this dashboard from any internet-enabled device to view real-time graphs of water distance, depth, and volume over time, allowing for trend analysis and better water management decisions.*



4) ***Hardware Setup:*** *The physical assembly of the system.*

- ***Description:*** *This image provides an overview of the assembled prototype, showcasing the Arduino board, ultrasonic sensor, GSM module, and LCD display connected on a breadboard. It visually confirms the physical integration of all components.*



2.3. Project website, repository And Demo video (LINKS)

The project's source code, detailed documentation, and further development insights are publicly available on the GitHub repository:

Website/GitHub Repository: *https://chigozieallanie.github.io/Smart-Water-Level-Monitoring-System/*

Link to project Repository:https://github.com/Chigozieallanie/Smart-Water-Level-Monitoring-System

Link to project website:https://chigozieallanie.github.io/Smart-Water-Level-Monitoring-System/

Link to demo Video: https://youtu.be/YJk17-zR0VY?si=M2fKoYL6ZFcLN25y

3. Limitations and Next Steps (max. 1 page)

3.1. Limitations

Despite successfully achieving its core objectives, the current iteration of the Smart Water Level Monitoring System has certain limitations that provide valuable insights for future improvements:

- ***Single SMS Recipient:*** *The system is presently configured to send SMS alerts to only one predefined mobile number, limiting its utility in multi-user or family environments.*
- ***Basic Volume Calculation:*** *The algorithm for calculating water volume assumes a simplified tank geometry (e.g., perfect cylinder or cuboid). This can lead to inaccuracies for tanks with irregular shapes or internal obstructions.*
- ***Absence of Automated Pump Control:*** *While the firmware defines output pins for pump control, the current implementation lacks the logic to automatically activate or deactivate a water pump based on water level thresholds.*
- ***Hardcoded API Key:*** *The ThingSpeak API key is directly embedded within the Arduino firmware. This practice poses a security risk, especially if the code were to be deployed in a less controlled environment or made fully public without proper precautions.*
- ***Reliance on GSM Network:*** *The system's remote communication capabilities (SMS alerts and cloud data logging) are entirely dependent on the availability and stability of the GSM mobile network. In areas with poor coverage, these functionalities would be compromised.*
- ***No Offline Data Storage:*** *In the event of a network outage or power interruption, the system does not have a mechanism to store historical water level data locally on the device, leading to potential data loss.*
- ***Manual Calibration:*** *Initial setup requires manual calibration of tank dimensions (e.g., total depth) directly within the code, necessitating a re-flash if these parameters change.*

3.2. Next Steps

Building upon the robust foundation established by this project, the following are proposed next steps to enhance the Smart Water Level Monitoring System:

- ***Multiple User Notifications:*** *Implement a feature to allow configuration of multiple phone numbers for SMS alerts or explore alternative notification methods like push notifications via a dedicated mobile application.*
- ***Dedicated Mobile Application Development:*** *Develop a user-friendly Android/iOS application. This app would provide a more interactive dashboard, real-time push notifications, historical data visualization, and allow users to configure system parameters (thresholds, tank dimensions) remotely.*
- ***Automated Pump Control Integration:*** *Incorporate logic to automatically control a water pump (e.g., via a relay module) based on predefined low and high-water level thresholds, enabling a fully automated water management solution.*
- ***Enhanced Security Measures:*** *Implement more secure methods for API key management, potentially utilizing server-side components for authentication or exploring encryption for sensitive data.*
- ***Local Data Storage with SD Card:*** *Integrate an SD card module to enable local logging of water level data. This would ensure data persistence during network outages, with the ability to upload buffered data to ThingSpeak once connectivity is restored.*
- ***Advanced Tank Profiling:*** *Allow users to define more complex tank geometries or calibrate the volume calculation more precisely through a guided setup process, improving the accuracy of volume estimations.*
- ***Web-based Configuration Interface:*** *For easier setup and calibration, explore using a Wi-Fi enabled microcontroller (e.g., ESP32 or ESP8266) to host a simple web server, allowing users to configure the system via a web browser without needing to re-flash the firmware.*
- ***Power Optimization and Battery Backup:*** *Further optimize power consumption for extended battery life in remote deployments and integrate a battery backup system to ensure continuous operation during power interruptions.*

3.3. References

[1] Ministry of Water and Environment (Uganda), National Water Policy, 2023. [Online]. Available: https://www.mwe.go.ug
***Relevance**: Cites 38% non-revenue water loss—justifies your system's overflow prevention.*

[2] T. S. Bitew et al., "Solar-Powered IoT for Off-Grid Water Monitoring in Africa," IEEE Access, vol. 9, pp. 165432–165447, 2021.

[3] R. G. Allen et al., Crop Evapotranspiration, FAO Irrigation Paper 56, 1998.
***Relevance**: Convert sensor readings to litres using tank dimensions (Chapter 3).*

[4] Africa's Talking, SMS API Documentation, 2024. [Online]. Available: https://developers.africastalking.com/docs/sms

[5] Uganda Communications Commission (UCC), Short Code Registration Guidelines, 2024. [Online]. Available: https://www.ucc.co.ug

[6] Hasbullah, A., Rahimi, A. H., Amrimunawar, A. I. H., Redzwan, F. N. M., Mahzan, N. N., Omar, S., & Zin, N. M. (2020, April). Flood and notification monitoring system using ultrasonic sensor integrated with IoT and Blynk applications: designed for vehicle parking. In Journal of Physics: Conference Series (Vol. 1529, No. 2, p. 022050). IOP Publishing.

[7] J. N. Kizito et al., "Sensor Fusion for Robust Water Level Tracking in Tropical Climates," IEEE Transactions on Instrumentation and Measurement, vol. 71, pp. 1–12, 2022.

[8] Hidayat, M. R., Sambasri, S., Fitriansyah, F., Charisma, A., & Iskandar, H. R. (2019). Soft Water Tank Level Monitoring System Using Ultrasonic HC-SR04 Sensor Based On ATMega 328 Microcontroller. 2019 IEEE 5th International Conference on Wireless

and Telematics (ICWT). https://doi.org/10.1109/icwt47785.2019.8978229

[9] Islam, M. a., Rahman, M. T., &. Islam, S. M. S. (2021). Design and Implementation of a Water Tank Level Monitoring and Control System Using Arduino and Android App. International Journal of Computer Science and Information Technology (IJCSIT), 11(1), 1-5.

[10] J. Dhillon, S. D., N. K. Mohonto, M. Hasan, S. Ahmed, S. C. Das. (2021). IoT based Water Level Monitoring and Motor Control System. 4th International Conference on Recent Developments in Control, Automation & Power Engineering (RDCAPE), Noida, India, 2021, Pp. 30-34, Doi: 10.1109/RDCAPE52977.2021.9633405.

[11] Johari, A., Wahab, M. H. A., Latif, N. S. A., Ayob, M. E., Ayob, M. I., & Ayob, M. A. (2010). Tank water level monitoring system using GSM network. (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 2 (3) , 2011, 1114-112.

[12] Kulkarni, S., Raikar, V. D., Rahul, B. K., Rakshitha, L. V., Sharanya, K., & Jha, V. (2020). Intelligent Water level Monitoring System using IoT. 2020 IEEE International Symposium on Sustainable Energy, Signal Processing and Cyber Security (iSSSC). https://doi.org/10.1109/isssc50941.2020.9358827

Appendix A – Project Work plan

The development of the Smart Water Level Monitoring System followed a structured project work plan, incorporating phases of planning, design, implementation, testing, and documentation. This approach allowed for systematic progress and effective resource allocation. The Gantt chart format below illustrates the key activities and their timeline

GANTT CHART

SMART WATER-LEVEL MANAGEMENT SYSTEM

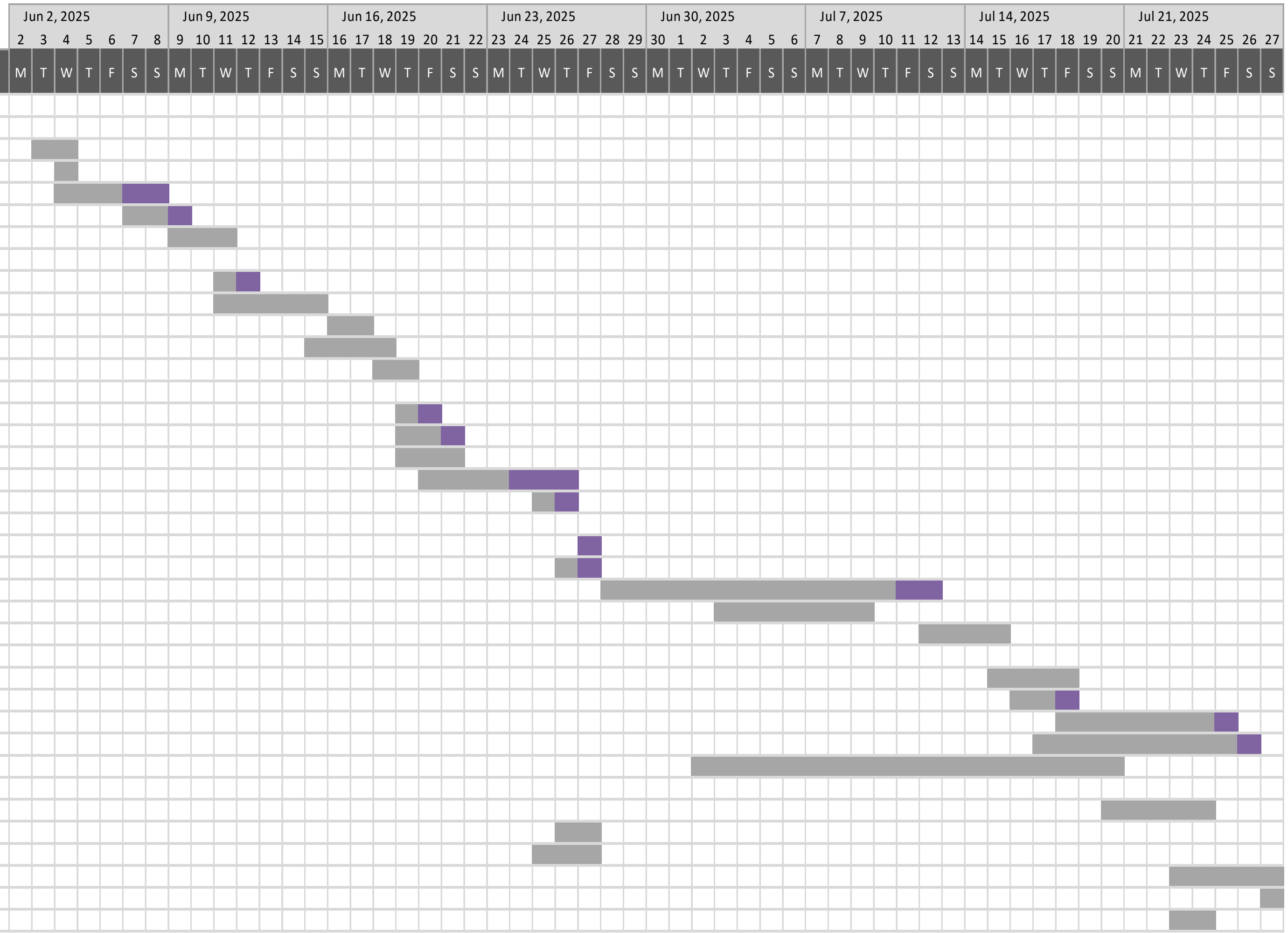
Makerere University

GROUP-3

By Keith

Project Start:	Tue, 6/3/2025
Today:	Mon, 7/28/2025
Display Week:	1

TASK	ASSIGNED TO	PROGRESS	START	END
Phase 1: Project/System Planning				
Forming Group	Allan	100%	6/3/25	6/4/25
Project Ideas	Everyone	100%	6/4/25	6/4/25
Concept Notes	Allan	78%	6/4/25	6/8/25
ID Problems & Solns	Everyone	88%	6/7/25	6/9/25
Proposal Draft (1)	Alan & Keith	100%	6/9/25	6/11/25
Phase 2: Analysis				
Research on current prop	Bwire	98%	6/11/25	6/12/25
Requirement-Gathering	Allan	100%	6/11/25	6/15/25
Understanding Sys. Archit	Emma	100%	6/16/25	6/17/25
Analysing Deliverables	Keith	100%	6/15/25	6/18/25
Comparative Studies	Allan	100%	6/18/25	6/19/25
Phase 3: Design				
System Architecture	Emma	90%	6/19/25	6/20/25
Gantt Chart Design	Keith	78%	6/19/25	6/21/25
Block Diagram	Allan	100%	6/19/25	6/21/25
Website Design	Bwire	68%	6/20/25	6/26/25
Hardware Schematic & Pi	Keith	69%	6/25/25	6/26/25
Phase 4: Implementation				
Supervisor meeting	Everyone	40%	6/27/25	6/27/25
Hardware Assembly &,Wi	Keith & Bwire	86%	6/26/25	6/27/25
Firmware Development (/	Allan & Emma	92%	6/28/25	7/12/25
ThingSpeak Channel Setu	Keith	100%	7/3/25	7/9/25
Website-ThingSpeak inte	Bwire & Allan	100%	7/12/25	7/15/25
Phase 5: Testing & Validation				
Unit Testing	Allan	100%	7/15/25	7/18/25
Integrtion Testing	Emma	97%	7/16/25	7/18/25
System Testing	Keith	99%	7/18/25	7/25/25
Debugging	Bwire	92%	7/17/25	7/26/25
Firmware Modification	Bwire & Keith	100%	7/2/25	7/20/25
Phase 6: Deliverables (Documentation)				
Task 1: Poster	Allan & Keith	100%	7/20/25	7/24/25
Task 2: Schematic Diagrai	Keith	100%	6/26/25	6/27/25
Task 3: Architecture	Emma	100%	6/25/25	6/27/25
Task 4: Final Report	Keith & Allan	100%	7/23/25	7/27/25
Task 5: Website FN	3wire & Emma	97%	7/27/25	7/27/27
Task 6: SRS Report	Emma & Bwire	100%	7/23/25	7/24/25



Appendix B – Contribution by Team Members

The successful delivery of the Smart Water Level Monitoring System project is a testament to the collaborative efforts and distinct contributions of each member of GROUP-3.

No.	Team Member	Contribution
1.	**Keith Paul Kato**	• Overall project planning & Task allocation (Gantt Chart) • Codebase (Initial firmware logic) • Jointly worked with Allan on progress & Final report • Schematic Diagram Construction using fritzing software • GitHub Control (Repository Structure) • Thingspeak platform creation & Control for data visualization. • Physical Hardware Setup for required specifications • Jointly Worked with Allan on Poster • Continuous Prototype testing.
2.	**Allan Kigozzi**	• Project Concept Idea (Concept note drafting) • Project Plan (initial Report) • Block diagram Construction • GitHub Control (Repository Structure) • Jointly worked with Keith on progress & Final Report • System Error Handling • Physical Hardware voltage setup to balance current necessary for the project e.g., DC-

		DC Step-down buck converter for additional voltage to support GSM power-on. • Jointly Worked with Keith on Poster • Continuous Prototype testing
3.	**Emmanuel Chelimo**	• Software Requirement Specifications Documentation (SRS) • System Architecture Design & Modification using Figma • Functional requirement gathering and Use case Scenarios. • Core Objectives of the System. • Codebase (firmware modification) i.e., Integration (API) • Continuous Sensor Calibrations for accurate readings • Jointly worked Rodney on Data analysis • Continuous Prototype Testing
4.	**Desire Rodney Bwire**	• Website Creation • Website integration with ThingSpeak • Codebase(firmware) calibration for desired analysis • Website Control and Error handling • Daily, Weekly, and monthly Reports on Website • Joint worked on Calibrations (Sensor, and codebase) for live data efficient display. • Continuous prototype testing

		• Alert System Calibrations • Ensured Accurate results readings on website that aligns with ThingSpeak Data.

Proposed Budget

Components	Quantity	Unit Price in UGX	Proposed Price
Arduino Uno R3 + Cable	1	60,000	55,000
Breadboard MB 102 Solderless type	1	16,000	15,000
10V DC Power Supply (5V) + GSM (5V)	1	12,500	10,000
10mm Diffused LEDs (Red, Yellow, Green)	5	4,000	3,000
1/2W Resistor Metal Film Resistor	7	4,000	3,500
20cm Jumper Wires 40pcs Dupont	10	15,000	14,000
HCD 3-24V High Decibel Active Buzzer	1	4,000	3,500
A DC-DC Step-Down (Buck) Converter & A Voltmeter	1	4,700	4,000
16*2 LCD Module with I2C Module (Soldered)	1	26,000	30,000
SIM800L GSM Modem	1	90,000	88,000
HC-SR04 Ultrasonic Sensor (Generic)	1	12,000	12,000
USB-A to B Cable (for Arduino)	1	15,000	10,000
Water Drum/Bottle (for prototype model)	1	42,000	40,000
Total Proposed Budget	**32**	**305,200**	**288,000**