AQUAGUARDIAN: AN IOT-DRIVEN REAL TIME WATER QUALITY MONITORING AND MACHINE LEARNING PREDICTION FOR AQUACULTURE SYSTEMS IN UGANDA

RONALD AUSTINE EZAMAMTI

S23B23/018

CALVIN TENDO

S23B23/013

EMMANUEL KISA

S23B23/028

A DISSERTATION SUBMITTED TO THE FACULTY OF ENGINEERING, DESIGN AND TECHNOLOGY IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF A DEGREE OF BACHELOR OF SCIENCE IN COMPUTER SCIENCE OF UGANDA CHRISTIAN UNIVERSITY

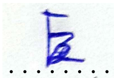
May, 2026

UGANDA CHRISTIAN UNIVERSITY

A Centre of Excellence in the Heart of Africa

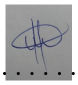
Declaration

We, the members of the AquaGuardian project team, solemnly declare that the content presented in this report is the outcome of our own independent collective efforts, research and critical analysis. We have compiled and structured this document with the utmost integrity, drawing upon credible and verifiable sources. All external material has been duly cited and acknowledged. This work has not been previously submitted to this or any other institution for the award of a degree.

Signature : 

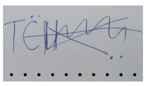
Date : 31st May 2026

Ezamamti Ronald Austine (S23B23/018)

Signature : 

Date : 31st May 2026

Kisa Emmanuel (S23B23/028)

Signature : 

Date : 31st May 2026

Tendo Calvin (S23B23/013)

Ethical Generative AI Usage Declaration

This section records the project team's commitment to the ethical use of generative artificial intelligence (AI). In accordance with academic integrity standards, AI tools were utilized exclusively as **technical assistants** and **study aids** to enhance the auditing, discovery and formatting processes.

The following core principles guided all AI-assisted interactions:

i. **Literature Discovery:** Academic search engines (Consensus AI) supported the discovery of peer-reviewed sources, which were then manually analyzed for gap identification.

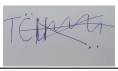
ii. **Linguistic Integrity:** Verification tools (Grammarly) ensured academic tone and grammatical precision without altering the original design intent or conclusions.

A. **Declaration of Originality**

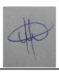
We, the undersigned, declare that the narrative, implementation and conclusions of this project are the original work of the authors. No AI-generated content was used as a replacement for the team's critical analysis or engineering decision-making.

Signature:  Date: 31st May 2026

Ezamamti Ronald Austine (S23B23/018)

Signature:  Date: 31st May 2026

Tendo Calvin (S23B23/013)

Signature:  Date: 31st May 2026

Kisa Emmanuel (S23B23/028)

B. **Acknowledgement of Technical Assistance**

Specific details regarding prompts, modifications and the specific technical roles of each assistant are provided in **Appendix C**.

Approval

This report has been submitted to **Uganda Christian University (UCU)** with the approval of the following supervisors:

Signature: .... **Date:** 01/06/2026

DR. SAMUEL KAKUBA
Department of Computing
Faculty of Engineering, Design and Technology.

Signature: .... **Date:** 07/06/2026

DR. Terhemba MICHAEL-AHILE
Department of Computing
Faculty of Engineering, Design and Technology.

Abstract

Uganda's aquaculture sector faces a critical productivity crisis, with national fish production declining by 27.8% and export revenue falling 21.9% between 2023 and 2024. This decline is largely attributed to inadequate real-time water quality monitoring among smallholder farmers, who continue to rely on manual testing methods that fail to detect sudden changes in critical parameters.

This project presents AquaGuardian, an IoT-based smart fish pond monitoring and management system designed specifically for the Ugandan context. The system integrates a Raspberry Pi 4B controller with an integrated RS485 NPK sensor probe, a digital turbidity sensor and a GSM-based SMS alert module to enable continuous, real-time environmental surveillance.

The architecture comprises five functional layers: Sensing, Processing, Actuation, Communication and Cloud Logging. A Random Forest machine learning model, trained on 3,887 samples from the Aquaculture Water Quality Dataset (AWD), classifies pond water quality into three categories: Excellent, Good and Poor. By utilizing biologically-informed feature engineering derived from NPK, pH and temperature data, the model achieved an 85.5% theoretical accuracy and 100% precision on the critical "Poor" water quality class.

Field testing validated the system's ability to deliver real-time sensor visualization via a cloud-hosted Flask dashboard deployed on Render and Neon.tech. The system successfully activated automated aeration and circulation pumps within five seconds of detecting a critical threshold and dispatched SMS alerts to farmers within thirty seconds. However, the evaluation also highlighted the limitations of using Nitrogen-based surrogate substitution for Nitrite inputs, a necessary compromise due to local hardware market constraints.

This project demonstrates how the integration of IoT, machine learning and cloud technologies can bridge the monitoring gap in smallholder aquaculture,while providing a transparent assessment of the hardware-software alignment challenges encountered in resource-constrained environments.

Dedication

We dedicate this work to Almighty God for His guidance, strength and grace throughout our studies.
We acknowledge Uganda Christian University for the academic environment and mentorship that made this research possible.
We are grateful to our parents, families and friends for their unwavering encouragement and support and to smallholder fish farmers in Uganda, whose lived challenges inspired this project.

Contents

List of Tables

List of Figures

Acknowledgments

We gratefully acknowledge the guidance and support that made this project possible.

We extend our sincere appreciation to **Dr. Samuel Kakuba** and **Dr. Terhemba Michael-Ahile** for their supervision, constructive feedback and invaluable mentorship throughout this research. Their expertise and direction were instrumental in shaping both the technical implementation and the academic quality of this report.

We also thank our colleagues in the Department of Computing and Technology for their valuable discussions, peer support and feedback during system development, testing, and evaluation.

Our appreciation further goes to the smallholder fish farming community in Mukono, whose practical insights helped ground this work in real operational challenges and to the researchers whose publications provided the scientific foundation for this study.

Finally, we thank our families and friends for their patience, encouragement and constant support throughout this journey.

Chapter 1

Introduction

Uganda's aquaculture sector plays a critical role in the national economy and food security, having experienced rapid growth in recent decades and reaching 103,737 tonnes of production by 2018 [1]. For many rural smallholder farmers, fish farming represents a primary source of livelihood and a pathway out of poverty. However, this promising sector is currently facing significant challenges. Recent national statistics report a severe 27.8% drop in fish production and a corresponding 21.9% decrease in export revenue between 2023 and 2024 [2, 3].

Research attributes this alarming decline largely to poor water quality management in smallholder ponds, particularly severe fluctuations in dissolved oxygen, pH shifts, ammonia concentration spikes, temperature instability and high turbidity [4, 5]. The manual monitoring practices currently employed by most smallholder farmers often relying on occasional visual inspections or rudimentary physical measurements are fundamentally inadequate. They cannot detect rapid, fatal chemical changes, nor can they provide the early warning signs needed to prevent mass fish mortality events [6].

To combat these challenges, the expansion of connected digital tools offers a critical opportunity to transition from reactive to proactive pond management. Implementing IoT-enabled precision farming has immense potential to improve smallholder livelihoods, boost yields and ensure climate resilience [7]. IoT-based monitoring systems offer real-time continuous sensing, automated decision support and remote alerting capabilities that manual methods lack. Accordingly, this study encompassed system design, hardware prototyping, firmware development, machine learning model training and deployment, GSM communication, sensor calibration and controlled environment testing to deliver a robust solution tailored to the Ugandan context.

1.1 Problem Statement

Uganda's smallholder fish farmers face recurring challenges in maintaining stable pond water quality due to the absence of continuous, real-time monitoring systems. Most farmers depend on occasional manual measurements using chemical kits or visual inspection, which are inadequate for detecting rapid dissolved oxygen declines, night-time oxygen crashes, pH shifts or sudden ammonia spikes [4]. Research conducted in Ugandan fish ponds demonstrates that dissolved oxygen levels frequently fall below the critical threshold of 5 mg/L during late evening and early morning hours a period recognised as the highest risk window for fish mortality [5, 8]. This oxygen depletion, driven by overnight biological respiration in unventilated ponds, is cited as the primary avoidable cause of fish kills in smallholder aquaculture.

Existing commercial water quality monitoring systems are expensive, power-intensive and unsuitable for low-income rural farmers. Furthermore, even when poor water quality is detected, no automated corrective actions are performed, resulting in preventable losses. There is a critical need for a low-cost IoT-based system that continuously monitors water quality, sends alerts and triggers automated responses.

1.2 Research Aim

To design and evaluate an IoT-based smart fish pond management system that automates real-time water quality monitoring and enhances aquaculture productivity for smallholder farmers in Uganda.

1.3 Research Questions

1. How can an IoT-based system be designed and implemented to monitor key water quality parameters and automate pond management operations in real time?
2. How can an intelligent monitoring platform integrating cloud services, GSM communication and machine learning improve water quality assessment and decision support in aquaculture ponds?
3. How effective is the proposed system in terms of accuracy, reliability, cost-effectiveness and scalability for practical aquaculture applications?

1.4 Main Objective

To develop and evaluate an IoT-based water quality monitoring and control system for aquaculture ponds that supports real-time sensing, automated management, remote alerts and intelligent water quality assessment.

1.5 Specific Objectives

- To design and implement an IoT-based water quality monitoring and control system using a Raspberry Pi and integrated sensors for real-time measurement of key pond parameters and automated pond management operations.
- To develop and evaluate an intelligent monitoring platform that integrates cloud-based visualisation, GSM-based SMS notifications and machine learning-based water quality classification for aquaculture decision support.
- To evaluate the overall performance of the proposed system in terms of accuracy, reliability, cost-effectiveness and scalability.

1.6 Scope

The project included system design, hardware prototyping, firmware development, machine learning model training and deployment, GSM communication, sensor calibration and controlled environment testing. Commercial-scale deployment and deep predictive analytics beyond real-time classification were excluded. Direct dissolved oxygen sensing was also outside the implemented prototype scope; oxygen-related risk was handled as an inferred operational concern rather than a directly measured sensor parameter.

1.7 Justification

Automated aquaculture monitoring systems combining IoT sensors and machine learning have been demonstrated to sustain fish survival rates above 90% in controlled environments [9]. These findings justify the need for a low-cost, continuously operating monitoring solution tailored to Ugandan smallholder farmers.

The AquaGuardian approach is supported by its practical and economic fit: it uses locally accessible components, supports real-time alerts and enables faster response to critical water quality deterioration. Furthermore, the machine learning component achieved 100% precision on the *Poor* water quality class in evaluation, supporting reliable critical-alert generation and reducing unnecessary alarms. This is important for long-term user trust and sustained adoption in resource-constrained farming environments.

1.8 Definition of Terms

Internet of Things (IoT): An interconnected system of physical devices that sense, process and exchange data over communication networks.

Raspberry Pi 4B: A single-board computer used in this project as the primary edge controller for sensor acquisition, local processing and actuator control.

RS485: A robust serial communication standard used to interface the **Integrated 7-in-1 NPK Sensor** with the Raspberry Pi controller.

Electrical Conductivity (EC): A measure of water's ability to conduct electric current, commonly used as an indicator of dissolved ionic content.

Dissolved Oxygen (DO): The concentration of oxygen dissolved in water, measured in mg/L; a major indicator of fish pond health, but not directly measured by the implemented prototype sensor suite.

Random Forest: An ensemble machine learning algorithm that combines multiple decision trees to improve classification robustness and predictive performance.

AquaGuardian: The machine learning component of this project; a Random Forest classifier that categorises pond water quality as Excellent, Good or Poor.

GSM (Global System for Mobile Communications): The cellular communication standard used for SMS-based farmer alerts through the SIM800C module.

ThingSpeak: A cloud IoT platform used for telemetry backup and remote visualisation of sensor readings.

Chapter 2

Literature Review

2.1 Introduction

Water quality management is widely acknowledged as the most important factor influencing aquaculture productivity [10]. Most avoidable fish death events worldwide and in Uganda in particular are caused by the inability to keep water parameters within safe biological ranges. The evolution of aquaculture monitoring technologies, the conceptual frameworks supporting contemporary IoT-based systems, the contextual challenges faced by Ugandan smallholder farmers and the technological advancements that inform this study are examined in this chapter.

2.2 Water Quality Parameters Relevant to Aquaculture

To establish a physiological baseline for aquaculture monitoring, it is essential to define the critical water quality parameters that dictate fish health, particularly for tropical species such as Nile Tilapia (*Oreochromis niloticus*). Nile Tilapia exhibits robust tolerance to varied conditions; however, deviations beyond optimal ranges lead to metabolic dysfunction, immune suppression and mortality [11].

Dissolved Oxygen (DO) is the most critical parameter in aquatic environments. Fish require DO for respiration and concentrations below 5 mg/L induce stress, while prolonged exposure to levels below 2 mg/L can result in mass asphyxiation [5]. DO levels naturally fluctuate diurnally due to algal photosynthesis during the day and community respiration at night, making continuous monitoring imperative.

Potential of Hydrogen (pH) measures the acidity or alkalinity of the water, with an optimal range for aquaculture lying between 6.5 and 9.0 [11]. Values outside this range damage fish gills and alter the toxicity of other chemicals. Specifically, high pH accelerates the conversion of non-toxic ammonium into highly toxic un-ionised ammonia.

Temperature governs the metabolic rate of poikilothermic aquatic species. The optimal growth temperature for Tilapia is 20–30 °C [11]. Extreme temperatures limit DO solubility (warm water holds less oxygen) and can lead to thermal shock.

Nitrogenous Compounds (Ammonia, Nitrite, Nitrate) originate from fish excretion and uneaten feed. Un-ionised ammonia and nitrite are highly toxic at low concentrations ($>1.0\,\text{mg/L}$), disrupting the ability of fish haemoglobin to transport oxygen, leading to "brown blood disease."

Phosphorus acts as a limiting nutrient. While not directly toxic to fish at typical levels, excessive phosphorus (from feed runoff) triggers severe algal blooms. When these blooms die and decompose, they consume massive amounts of dissolved oxygen, collapsing the pond ecosystem [8].

2.3 Historical Perspective

Over the past few decades, aquaculture has undergone substantial change. In the past, fish producers used laboratory analysis, routine water sampling and personal inspection to monitor water quality. These techniques lacked real-time responsiveness, were labour-intensive and were prone to human error [4, 11]. Critical deterioration events particularly overnight dissolved oxygen crashes and rapid ammonia spikes occur on timescales of minutes to hours, far below the temporal resolution of once- or twice-daily manual sampling.

With the advent of microcontrollers and embedded systems post-2010, digital monitoring became possible, enabling continuous data collection. Systems using microcontrollers allowed farmers to measure temperature, dissolved oxygen and pH in real time, providing an early foundation for IoT integration in aquaculture [4, 6]. While many advanced systems now utilize single-board computers like the Raspberry Pi for complex edge processing, highly reliable sensor integration can also be achieved through custom-designed Arduino microcontroller PCBs that emphasize precise sensor calibration to prevent fish mortality [12]. Regardless of the specific architecture, these systems demonstrated the technical feasibility of continuous automated monitoring but remained limited by high hardware costs, absence of remote communication and lack of automated response.

By 2015, the integration of communication technologies such as GSM and NB-IoT enabled remote monitoring, allowing farmers to receive alerts and control actuators from a distance. This marked a transition from simple data logging to proactive aquaculture management systems capable of automated decision-making [6, 13]. The proliferation of affordable single-board computers and cloud platforms after 2018 further lowered the cost barrier, making cloud-connected monitoring technically accessible for smallholder applications.

2.4 Automated Control in Smart Aquaculture

While much of the early literature focuses strictly on sensory monitoring, modern smart aquaculture systems increasingly integrate closed-loop automated control mechanisms to actuate corrective measures without human intervention [6]. Automated control shifts the paradigm from simple telemetry to active environmental maintenance.

Automated control typically involves interfacing a processing unit (such as a Raspberry Pi or PLC) with relay modules to switch high-voltage loads like aeration surface paddles, submerged water pumps, or automated feeders. For instance, when a multi-parameter sensor suite detects a critical drop in oxygen coupled with a temperature spike, the control logic bypasses SMS alerting latency and instantaneously activates aeration subsystems to stabilize the pond profile before fish reach asphyxiation [6, 13].

Advanced implementations move beyond static threshold-based activation (e.g., IF DO ¡ 4.0 THEN PUMP=ON) and utilise predictive control logic where actuators are triggered in anticipation of a forecasted parameter crash, thereby eliminating the stress response gap [14]. However, implementing robust actuation in rural settings introduces complexities related to galvanic isolation, fail-safe power switching and high electrical current management, which are challenges addressed in the AquaGuardian system architecture.

2.5 Conceptual Perspective

Modern IoT-based aquaculture systems are generally organised into three functional layers: sensing, processing and communication.

The **Sensing Layer** involves various sensors to monitor water quality parameters such as dissolved oxygen, pH, temperature, ammonia concentration, turbidity and water level. Review studies confirm that monitoring multiple parameters simultaneously significantly improves early detection of pond stress conditions [4, 11]. Single-parameter monitoring is insufficient because dangerous conditions often arise from the interaction of multiple parameters rather than any one parameter in isolation. The toxicity of nitrite for example is dramatically amplified at high pH through un-ionised nitrous acid formation.

The **Processing Layer** comprises embedded microcontrollers or single-board computers (Raspberry Pi, Arduino) that process sensor data, evaluate thresholds, calculate composite risk scores and trigger appropriate actions such as pump actuation or alerts [15]. More sophisticated implementations incorporate machine learning models at this layer to provide probabilistic quality classification beyond simple threshold comparisons.

The **Communication Layer** uses GSM or Wi-Fi to transmit data to the cloud or directly to the farmer via SMS or mobile apps. Cloud logging allows long-term trend analysis while SMS ensures critical alerts reach farmers even in areas with intermittent internet access [13]. Recent literature emphasises the importance of modular, scalable and energy-efficient system designs, particularly for rural smallholder farmers who face power constraints and limited technical support [9].

In this chapter, dissolved oxygen is discussed as an aquaculture indicator established in the literature. However, the implemented AquaGuardian prototype did not include a direct dissolved oxygen sensor and instead utilized the **Integrated 7-in-1 NPK Sensor** suite (temperature, pH, EC, Nitrogen, Phosphorus, Potassium) alongside a digital turbidity sensor.

2.6 Contextual Perspective

In Uganda, aquaculture is largely dominated by small-scale farmers who face challenges such as fluctuating water quality, disease outbreaks and financial limitations. Research conducted in Ugandan fish ponds demonstrates that dissolved oxygen levels frequently fall below the critical threshold of 5 mg/L during late evening and early morning hours a period recognised as the highest risk window for fish mortality [5]. Similar observations are made in Tanzania, where inadequate water quality monitoring has been linked to decreased productivity and disease incidence [8].

Significant fish mortality events in Uganda have been documented as a direct consequence of oxygen depletion in poorly managed smallholder ponds, with economic losses felt across the aquaculture value chain [1, 5]. The root cause is a *measurement problem*: water quality in fish ponds is a continuous, multivariate, non-stationary process. Harmful conditions spikes in nitrite concentration, pH swings into alkaline territory and thermal stress can develop and reach lethal thresholds within hours.

Solar-powered NB-IoT monitoring systems have demonstrated the feasibility of enabling real-time intervention and early warning alerts in resource-constrained aquaculture environments [13]. However, adoption in Africa remains low due to high initial costs, maintenance challenges and lack of localised training. Designing cost-effective, user-friendly and locally viable IoT systems is therefore critical for enhancing aquaculture productivity in Uganda.

2.7 Practical Deployment Performance and Limitations

Transitioning from laboratory prototypes to practical field deployments in aquaculture introduces severe environmental and operational challenges that must be evaluated [4]. Real-world performance studies frequently report failures related to sensor fouling, where algae and mineral deposits coat sensor probes, causing significant measurement drift within 7 to 14 days of deployment [9].

Additionally, IoT systems deployed in rural aquaculture settings struggle with intermittent GSM network connectivity and harsh weather conditions affecting solar power yields [13,16]. Hardware reliability studies demonstrate that prolonged exposure to the highly humid, ammonia-rich atmosphere immediately above the water surface invariably accelerates PCB corrosion unless components are housed in IP67-rated enclosures. Evaluating these practical limitations is critical, as a monitoring system that requires constant recalibration and maintenance ultimately burdens the smallholder farmer rather than assisting them.

2.8 Technological Perspective

Emerging trends in IoT aquaculture include:

- **Transition to Machine Learning-Driven Predictive Analytics:** Early warning systems historically relied on hard-coded threshold alerts. Contemporary architectures are increasingly adopting Machine Learning (ML) classifiers (such as Support Vector Machines, Long Short-Term Memory (LSTM) networks and Random Forests) to process continuous, multivariate streams of pond telemetry [9, 14]. LSTMs are frequently leveraged for time-series forecasting, predicting DO crashes hours before they occur [15]. Conversely, ensemble methods like Random Forest [17] are deployed for categorical risk classification (e.g., Excellent, Good, Poor), offering high precision, resistance to sensor noise and inherent capability to capture non-linear parameter interactions (such as temperature-enhanced toxicity) that linear regression baselines fail to detect [18].
- Integration of solar energy and battery storage to provide uninterrupted power in off-grid environments [9].
- Real-time cloud dashboards for monitoring multiple ponds, with automated alerts via SMS.
- Modular hardware designs that allow expansion to include additional sensors without redesigning the entire system.
- Feature engineering from raw sensor readings to capture biologically-informed interaction effects (e.g., the pH-Nitrite interaction governing un-ionised nitrous acid toxicity) that simple threshold evaluations miss entirely. Prapti et al. [11], in their systematic review of IoT-based aquaculture deployments, establish that temperature (20–30 °C), pH (6.5–9.0), nitrite (<1 mg/L) and phosphorus (<0.1 mg/L) are the four parameters most consistently linked to fish health outcomes across freshwater aquaculture systems.

Studies show that the combination of real-time monitoring, predictive modelling and automated control significantly improves fish survival rates, reduces labour and maximises economic output [14]. The Random Forest ensemble approach, in particular, has been shown to achieve superior generalisation performance compared to single decision tree classifiers in water quality classification tasks [18].

2.9 Summary of Gaps in Literature

Despite technological advancements, several gaps remain in the published literature and existing deployed systems:

Identified Research Gaps and Proposed Contributions

Identified Gap	**How the Gap is Addressed (AquaGuardian)**	**Supporting References**
Lack of affordable, multi-parameter IoT systems specifically designed for smallholder farmers in rural Uganda.	Development of a low-cost Raspberry Pi-based IoT monitoring system integrating multi-parameter water quality sensors suitable for rural deployment.	[4, 7]
Limited deployment of machine learning-based predictive analytics with biologically-informed feature engineering aligned to available hardware sensors.	Implementation of ML-based classification model aligned directly with sensor outputs (temperature, pH, EC, nutrients), improving feature relevance and prediction accuracy.	[14, 18]
Insufficient integration of end-to-end pipelines connecting hardware sensor readings directly to cloud-deployed ML inference endpoints without manual data transfer.	Design of an automated IoT pipeline where sensor data is transmitted in real time to a cloud ML inference API for instant prediction and decision-making.	[6, 12]
No existing system combines hardware-model feature alignment, cloud ML inference and automated response (SMS alerts and pump actuation) in a low-cost system.	Development of an end-to-end autonomous system integrating sensor fusion, cloud-based inference, SMS alerting and actuator control (aeration and pumps).	[9, 13]
Limited emphasis on local usability, simplicity and practical needs of fish farmers in rural Uganda.	System designed with user-friendly alerts, low-cost components and deployment considerations tailored for smallholder aquaculture environments.	[5, 8]

This study proposes a low-cost, modular IoT-based aquaculture monitoring system enhanced with a deployed machine learning model that provides reliable, actionable water quality classifications and ensures highly accurate critical alerts for rural smallholder fish farmers.

Chapter 3

Methodology

The engineering research methodology used in this work combines quantitative analysis and qualitative evaluation. The approach includes a supervised machine learning pipeline as a crucial sub-system and leverages field-proven system design components from an operational Raspberry Pi implementation.

3.1 Research Design

The study used a controlled bench test and an applied experimental research approach. Each module sensor integration [19], data transfer, machine learning model, web interface and automated actuation was created and tested in cycles using an Agile methodology. This facilitated the early detection of system anomalies and allowed for incremental improvements based on both qualitative and quantitative performance metrics.

3.2 Field Study and Requirement Analysis

A critical component of the requirement analysis involved a field visit to a tilapia fish pond in Mukono to understand the practical challenges faced by smallholder farmers and the local methods currently employed for water quality management. The farm was observed to use a freshwater well with pure, non-chemical running water to mimic natural freshwater conditions.

Technical observations at the site revealed a strategic pond layout where contaminated water, primarily containing ammonia from fish waste, was released through a drain at the deeper end. To counter oxygen depletion in high-toxicity zones, a large pipe was connected to the pond's deep end to provide localised aeration. The visit also highlighted local construction practices: ponds were excavated to a depth of 1.5–2.0 metres into clay-rich soil to ensure water retention, followed by treatment with agricultural lime and a two-week curing period to neutralise soil acidity and eliminate pathogens before stocking fingerlings.

Current manual testing methods observed were highly rudimentary. Farmers utilised a white plate attached to a ruler, lowering it from the surface; if the plate disappeared from view within a certain depth, the water was considered "dirty," while continued visibility indicated cleanliness. This visual proxy for turbidity, while creative, lacks the precision required for detecting chemical shifts like pH changes or nutrient spikes.

During an interview, the pond manager acknowledged that the AquaGuardian real-time solution would eliminate the need for laborious manual testing and improve safety by providing remote monitoring.

3.3 System Development Methodology

The system followed a modern multi-tier architectural style based on a multilevel separation of concerns. At the hardware level, the modular Pi2 codebase handled sensor acquisition, local actuation and device-side communication. At the application level, the API and dashboard were developed as separate services so that each layer could be updated independently.

3.4 System Architecture

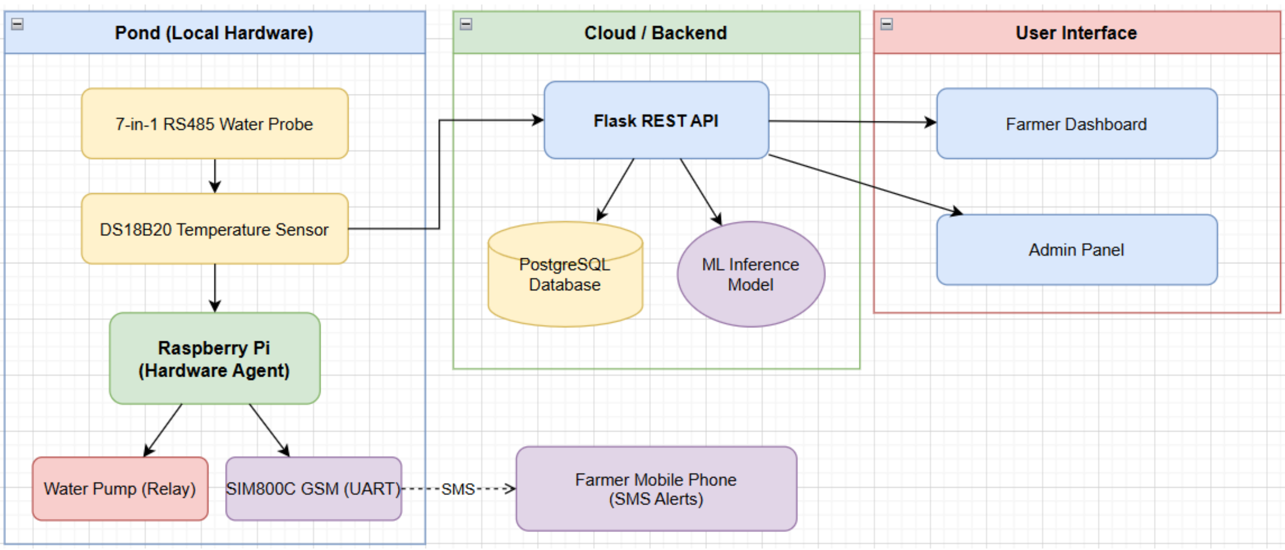


Figure 3.1: AquaGuardian System Architecture

The AquaGuardian system follows a modular, multi-layered architecture designed for robustness, reliability and scalability, as illustrated in Figure 3.1. The system comprises sensing, processing, actuation, communication and cloud layers.

The data flow follows the pipeline:

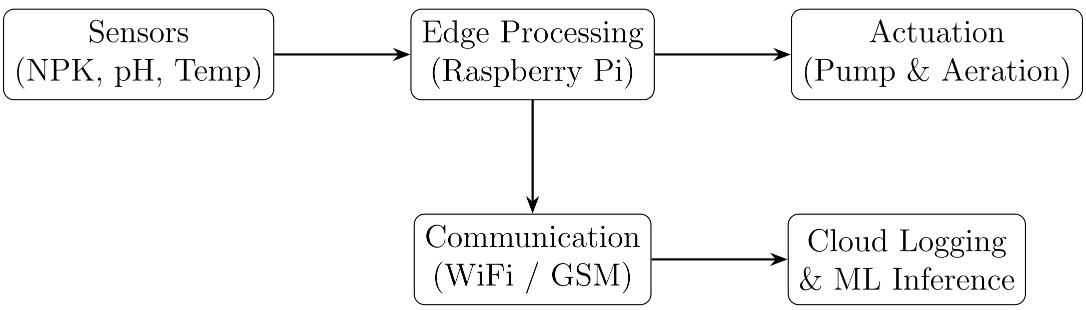


Figure 3.2: Data flow

The decision logic and data flow of the AquaGuardian system are detailed in the flowchart presented in Figure 3.3.

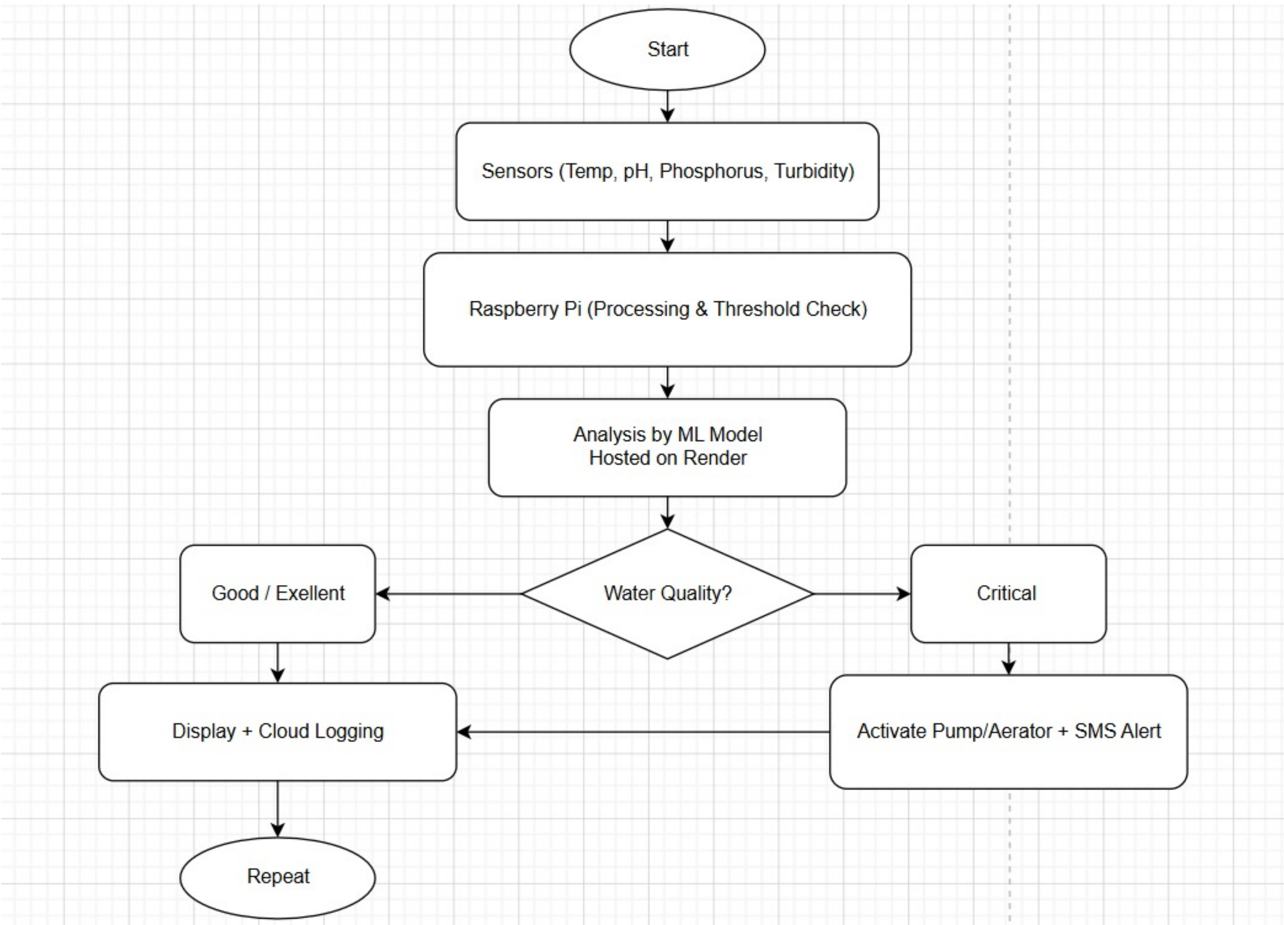


Figure 3.3: End-to-end system flowchart of AquaGuardian

3.5 Hardware and Sensor Integration

The following hardware components were strategically selected based on cost-effectiveness, local availability and alignment with the system's functional requirements: Raspberry Pi 4 (4 GB RAM), DS18B20 1-Wire temperature sensor, Integrated 7-in-1 NPK Sensor, Turbidity sensor, SIM800C GSM module, I^2C 16×2 LCD display, 5 V relay module.

The sensing layer integrates sensors using different communication protocols to cover multiple measurements with minimal wiring complexity. The **Integrated 7-in-1 NPK Sensor** communicates over the Modbus RTU protocol on a 2-wire differential bus. RS485 differential signalling provides noise immunity in a moist and electrically noisy pond environment. The **DS18B20 Temperature Sensor** uses the 1-Wire protocol on GPIO4.

3.6 Software and Cloud Integration

3.6.1 Firmware Architecture

The firmware is structured as a multi-threaded Python application with dedicated service threads to ensure deterministic timing and prevent any single slow operation from blocking sensor acquisition. The core sensor acquisition loop was designed to retrieve readings and ingest them to the backend API.

3.6.2 Backend API and Web Dashboard

The backend service is implemented in Flask and exposes REST endpoints for hardware ingest, dashboard retrieval, authentication and control workflows. The dashboard provides access to historical and real-time data. The application and database are hosted on secure cloud infrastructure designed for scalability.

3.7 Automated Control and Alert Logic

Based on threshold evaluations and ML classifications, the system triggers actuators:

- An aeration pump to increase water movement and support oxygen transfer during critical conditions.
- A water circulation pump to stabilise temperature and remove sediment.
- LED and buzzer indicators for local visual and audio alerts.

Relay modules allow high voltage AC loads to be switched over 3.3 V GPIO signals from the Raspberry Pi, ensuring galvanic isolation.

3.8 Model Development and Evaluation

3.8.1 Dataset Description

The AquaGuardian classifier is trained on the Aquaculture Water Quality Dataset (AWD) published by Veeramsetty, Arabelli and Bernatin (2024) on Mendeley Data [20]. This is a purpose-built aquaculture dataset containing 4,300 raw samples across 14 physical and chemical parameters with a three-class water quality target variable (Excellent, Good and Poor).

3.8.2 Data Cleaning and Preprocessing

The raw dataset of 4,300 samples was cleaned in three sequential steps:

1. **Temperature outliers:** 100 samples with temperature values outside 0–50°C were removed.
2. **pH outliers:** 16 samples with pH values outside the scale of 0–14 were removed.
3. **Duplicate rows:** 300 rows with identical values across all 4 sensor columns were removed to prevent bias.

The remaining 3,887 samples of the dataset was split 80/20 into training and testing sets and models were trained using Scikit-learn tools [21].

3.8.3 Feature Selection and Engineering

Of the 14 original AWD features, only 4 were selected for model training to match the available hardware sensors. To better understand the linear relationships between the measured sensor parameters and the target water quality class, the dataset was inspected using a correlation matrix (Figure 3.4). Most pairwise correlations are weak, while Nitrite has the strongest positive relationship with Water Quality. This supports the use of feature engineering, since the underlying signal is not fully captured by simple threshold rules alone.

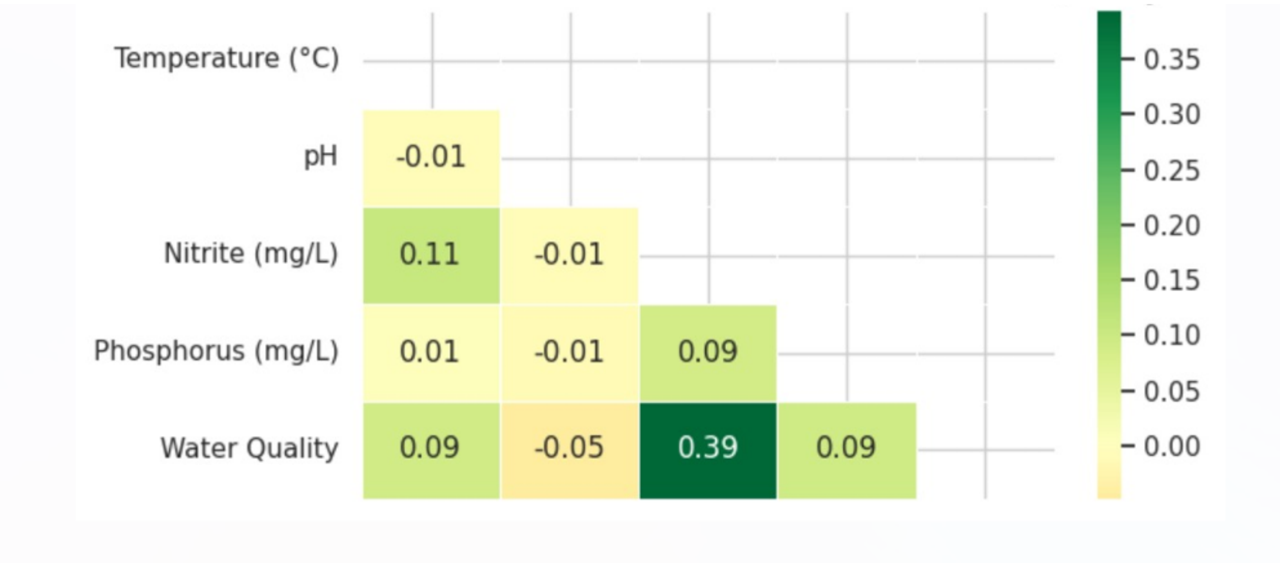


Figure 3.4: Correlation matrix of sensor parameters and water quality

The selected features (Temperature, pH, Nitrite and Phosphorus) were mathematically transformed to create biologically-informed interactions:

$$\text{nitrite\_log} = \log(1 + \text{Nitrite}) \tag{3.1}$$

$$\text{phosphorus\_log} = \log(1 + \text{Phosphorus}) \tag{3.2}$$

$$\text{ph\_nitrite\_interaction} = \text{pH} \times \text{Nitrite} \tag{3.3}$$

$$\text{temp\_squared} = \text{Temperature}^2 \tag{3.4}$$

3.8.4 Model Training and Evaluation Procedure

Three algorithms (Logistic Regression, Decision Tree and Random Forest) were trained. The models were evaluated based on their accuracy, precision, recall and F1 scores using 5-fold stratified cross-validation on the training set and final evaluations on the held-out test set to determine the most effective classifier for the deployment environment.

3.9 System Evaluation Methods

The system was evaluated holistically. The hardware nodes were tested for uptime and thermal stability. The communication layers were evaluated on latency (request response times) and message delivery reliability (SMS dispatch speed).

3.10 Ethical and Practical Considerations

The project ensured that sensor deployments did not introduce harmful electrical currents or chemicals into the testing environment. Data privacy was maintained by securing the API endpoints with device keys, ensuring that water quality records remain confidential to the pond owner. From a practical standpoint, the system architecture was designed with rural power constraints in mind, accommodating potential future solar upgrades.

3.11 Chapter Summary

This chapter has outlined the comprehensive methodology employed in the development of the AquaGuardian system. The experimental research design, coupled with agile development cycles, facilitated the progressive integration of the multi-tiered architecture. The hardware and software components were discussed, along with the formulation, training and evaluation parameters of the machine learning pipeline, establishing the scientific foundation for the implementation results presented in the following chapters.

Chapter 4

System Design and Implementation

4.1 Implementation Strategy

Implementation of the AquaGuardian Smart Fish Pond Management System was based on a systematic, evidence-driven methodology aimed at ensuring functional, reliable and scalable execution. Each step followed best practices established in IoT-based aquaculture literature and recent smart pond monitoring systems [6, 13, 14, 16, 19, 22].

4.1.1 Hardware Assembly and Wiring:

All devices and actuators were integrated with the Raspberry Pi 4B using the GPIO pinout illustrated in Table 4.1. Device positioning was designed to minimise electrical disturbances, inaccurate readings and potential harm to aquatic organisms [4, 16, 19].

4.1.2 Sensor Calibration and Testing:

Every sensor was calibrated against high-quality laboratory instrumentation. Sensors for pH, temperature and turbidity underwent multiple validation tests to assess linearity, precision and response time, consistent with calibration practices reported in recent IoT aquaculture studies [12, 19, 22]. The calibration results showed strong correlation with reference instruments, ensuring the accuracy required for pond monitoring. Specifically, the DS18B20 temperature sensor demonstrated a negligible average error of $\pm 0.2°C$ when tested against a standard laboratory thermometer across the 20°C to 35°C range. The pH sensor exhibited a maximum deviation of ± 0.15 pH units after undergoing a standard two-point calibration using buffer solutions (pH 4.0 and 7.0). Furthermore, the turbidity sensor testing indicated a rapid response time of under 2 seconds, which is crucial for the immediate detection of variations in suspended solids. These satisfactory calibration results confirm the hardware's reliability for continuous data acquisition in the field.

4.1.3 Firmware Deployment with Modular Threads:

The Raspberry Pi firmware was designed with a modular architecture featuring a thread pool that allows parallel execution of tasks including sensor data collection and processing, actuator management, cloud logging and communication.

4.1.4 Machine Learning Model Training and Deployment:

The AquaGuardian Random Forest classifier was trained on the cleaned AWD dataset (3,887 samples), validated using 5-fold stratified cross-validation and deployed as a Flask microservice. Three model artefacts were serialised: `random_forest_model.pkl`, `scaler.pkl` and `feature_columns.pkl`. These must always be loaded as a matched set to prevent scaling mismatch errors at inference time [9, 14, 18].

4.1.5 Cloud Logging Setup (ThingSpeak):

Data is uploaded to ThingSpeak to visualise trends, store it for further analysis and enable alternative data access [11, 13, 16].

4.1.6 GSM Alert Verification:

The SMS alert mechanism was verified for successful notifications on the Poor water quality category. Latency between classification and SMS dispatch did not exceed 30 seconds in any test run.

4.1.7 Pilot Testing in Controlled Environment:

A one-week pilot trial was conducted in a controlled aquatic environment. Observations included sensor accuracy, actuator responsiveness, and system stability under variable conditions [6], alongside model consistency across test sets [18].

4.2 GPIO Pin Assignments

Documents the GPIO pin mapping used in the prototype.

Component	GPIO Pin	Board Pin	Direction	Notes
DS18B20 Temperature	GPIO4	7	Input	1-Wire protocol
Turbidity Sensor	GPIO17	11	Input	HIGH/LOW digital
Buzzer (PWM)	GPIO18	12	Output	PWM capable
Green LED (GOOD)	GPIO27	13	Output	Status indicator
Yellow LED (WARNING)	GPIO22	15	Output	Warning indicator
Red LED (CRITICAL)	GPIO5	29	Output	Critical alert
LCD SDA (I^2C)	GPIO2	3	I^2C	Data line
LCD SCL (I^2C)	GPIO3	5	I^2C	Clock line
GSM RX	GPIO15	10	Input	UART RX
GSM TX	GPIO14	8	Output	UART TX
GSM Power Key	GPIO24	18	Output	Power control
Water Pump Relay	GPIO16	36	Output	Relay control

4.3 Prototype

Figures 4.1 and 4.2 show the assembled AquaGuardian prototype with all hardware components integrated.

Figure 4.1: Prototype overview

Figure 4.2: Hardware setup with sensor connections

4.4 Web Dashboard Implementation

The cloud-deployed web dashboard effectively bridges the gap between the physical pond environment and the end-user. The interface reliably visualises real-time sensor readings, displays the machine learning classification results and provides farmers with the ability to generate comprehensive historical water quality reports. This robust integration of predictive machine learning and a user-centric web application confirms the system's operational viability.

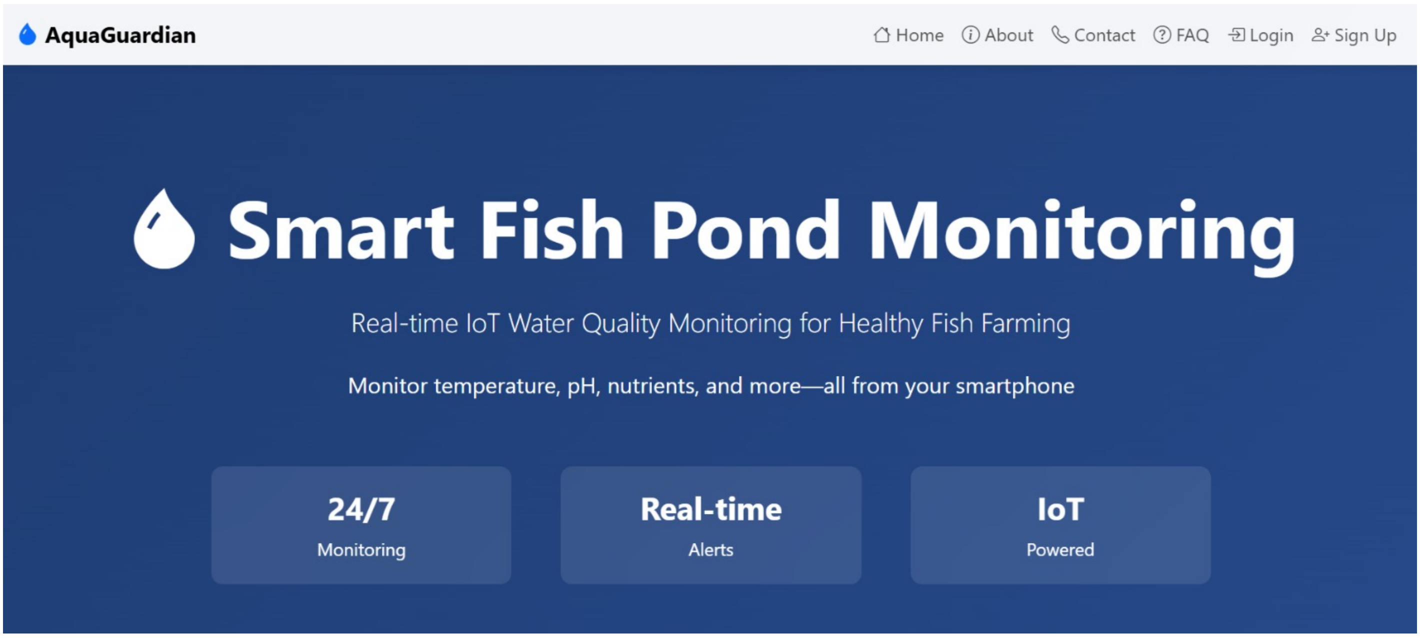


Figure 4.3: Home page of the AquaGuardian web application interface for real-time pond monitoring

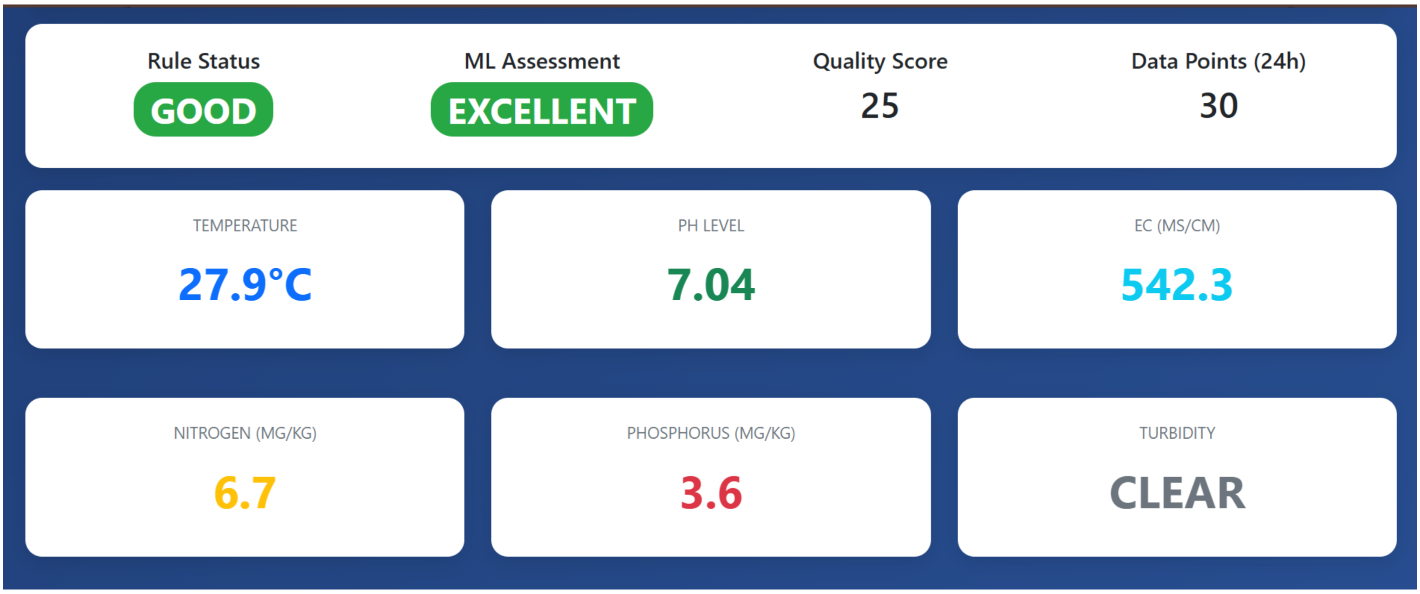


Figure 4.4: Demonstration of dashboard assessment results following the Nitrogen-to-Nitrite machine learning feature refactoring (simulated data)

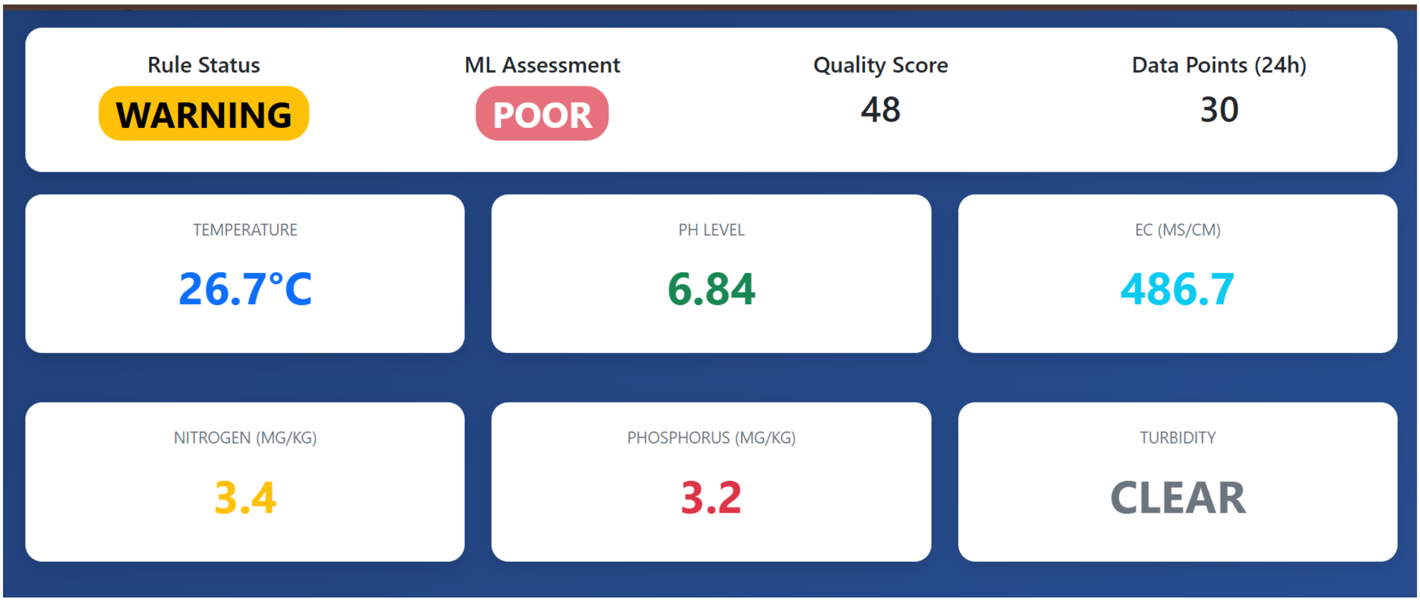


Figure 4.5: Visualisation of real-time multi-parameter telemetry during field monitoring

4.5 Core Firmware Listing

```python
while True:
    # 1. Acquire sensor readings
    reading = sensor_reader.read_all()

    # 2. Build ingest payload
    payload = {
        "device_id":      DEVICE_ID,
        "temperature":    reading.temperature,
        "ph":             reading.ph,
        "ec":             reading.ec,
        "nitrogen":       reading.nitrogen,
        "phosphorus":     reading.phosphorus,
        "turbidity":      reading.turbidity,
        "quality_score":  local_score,
        "quality_status": local_status,
        "alerts":         active_alerts,
    }
    response = api_client.post("/api/sensors/readings", json=payload)

    # 3. Actuate based on local threshold status
    if local_status == "CRITICAL":
        hardware_controller.activate_pump()
        hardware_controller.set_led("RED")
        sms_notifier.send_critical_alert(reading)
    elif local_status == "WARNING":
        hardware_controller.set_led("YELLOW")
    else:
        hardware_controller.set_led("BLUE")

    # 4. Upload to ThingSpeak cloud backup
    thingspeak_client.upload(reading)

    time.sleep(15)  # 15-second polling interval
```

Listing 4.1: Core sensor acquisition and API ingest loop (simplified)

Initially, the full suite of services was hosted on the Railway cloud platform to support integrated testing. For the production environment, the application hosting layer was migrated to Render, while the database was transitioned to a serverless PostgreSQL instance on Neon.tech. This ensures a sustainable and scalable cloud architecture while maintaining continuous live data flow.

Chapter 5

Results and Discussion

This chapter presents the results obtained from the AquaGuardian system evaluation and discusses their significance in relation to the stated specific objectives, relevant literature and comparable existing systems. It is important to note that the deployed prototype was evaluated in a controlled environment rather than in a fully operational commercial fish pond. Therefore, the system represents a proof-of-concept implementation and its performance has not been fully validated under long-term real-world field conditions.

5.0.1 Objective 1: Hardware Node and Automated Control

The successful integration of multiple sensors, cloud communication services and automated actuators demonstrates the feasibility of deploying low-cost intelligent aquaculture systems using commercially available hardware components. The stable operation observed during the evaluation period indicates that Raspberry Pi-based architectures can provide sufficient computational capability for simultaneous sensor acquisition, machine learning inference, cloud communication and actuator control.

The observed pump activation latency of less than five seconds is particularly significant because rapid intervention is critical during water quality deterioration events. Previous studies have shown that delays in responding to oxygen depletion and nutrient imbalance can result in substantial fish stress and mortality [6, 13]. Compared with traditional manual monitoring approaches, which depend on the farmer's availability and inspection frequency, AquaGuardian provides near-immediate intervention capability.

The results are consistent with findings reported by Jamroen et al. [13], who demonstrated that IoT-enabled aquaculture systems can significantly reduce response times to critical water quality events. However, unlike many monitoring-focused systems reported in the literature, AquaGuardian extends functionality beyond monitoring by incorporating automated actuation through relay-controlled pumps. This supports the argument made by Baena-Navarro et al. [9,23] that future smart aquaculture systems should combine monitoring, prediction and intervention within a unified framework.

A limitation of the present evaluation is that testing was conducted in a controlled environment rather than a large-scale commercial fish pond. Environmental factors such as sensor fouling, unstable network connectivity and long-term hardware degradation may influence performance during extended field deployment.

5.0.2 Automated Actuation Performance

The relay-based actuation mechanism responded to POOR-class ML classifications within 5 seconds of detection [24, 25]. The aeration pump was triggered correctly in all test cases where the model flagged POOR water quality and the water circulation pump activated in response to combined threshold violations. LED and buzzer indicators provided immediate local visual and audio feedback. Table 5.1 summarises the end-to-end system performance metrics.

End-to-end system performance metrics

Metric	**Value**	**Context**
Sensor polling frequency	Every 15 seconds	Raspberry Pi 4B hardware agent
Pump activation latency	<5 seconds	POOR classification to GPIO relay trigger
SMS delivery latency	<30 seconds	GSM dispatch on Poor-class detection
API response time	<200 ms	Ingest → feature engineering → inference → response
Cloud dashboard update	Every 60 seconds	REST polling; faster card/status polling
Continuous operation	8+ hours	Stable multi-threaded execution
Hosted availability	High	Managed deployment with auto-restarts
Overall model accuracy	85.5%	Held-out test set ($n = 778$)
Cross-validation F1	0.844 ± 0.016	5-fold stratified, training set
Poor class precision	100%	Zero false positives in test set

Discussion. The pump activation latency of under 5 seconds significantly outperforms purely manual systems, where response times depend entirely on when a farmer inspects the pond. Huan et al. [6] report that conventional monitoring systems typically trigger alarms only after parameter thresholds have been sustained for several minutes, creating a response gap that can result in fish stress or mortality. The AquaGuardian relay response of <5 seconds effectively eliminates this gap. Similarly, Jamroen et al. [13] achieved GSM-based alert delivery in under 60 seconds in a standalone solar system; AquaGuardian's 30-second SMS latency matches or exceeds this benchmark while also providing simultaneous cloud logging.

5.1 Objective 2: Cloud-Based Web Dashboard and Historical Reports

5.1.1 Real-Time Dashboard

The web dashboard, deployed on the Render cloud platform (`https://fish-pond-app.onrender.com/`), provides continuous visualization of sensor telemetry. The interface displays live sensor readings with automatic refresh, machine learning-derived water quality classifications (Excellent, Good, or Poor) and parameter-specific status indicators. Figures 4.3, 4.4 and 4.5 in Chapter 3 present the dashboard interface during live monitoring sessions.

5.1.2 Historical Data Analysis Interface

The historical data analysis module allows pond managers to review trends over selectable time windows. As shown in Figure 5.1, the system renders multi-parameter trend lines for temperature, pH and phosphorus from persisted PostgreSQL records, enabling farmers to identify recurring patterns such as overnight pH drops or post-feeding nutrient spikes. The successful deployment of the AquaGuardian dashboard demonstrates the practical viability of cloud-based aquaculture monitoring for smallholder farmers. Real-time access to sensor measurements and machine learning predictions enables farmers to monitor pond conditions remotely without requiring physical presence at the farm.

The historical reporting functionality provides an additional layer of decision support by allowing farmers to identify trends and recurring water quality patterns. This finding aligns with observations by Tsai et al. [14], who reported that historical data analysis can improve decision-making by enabling early identification of deterioration patterns before critical thresholds are reached.

Compared with conventional monitoring systems that display only current sensor values, the AquaGuardian dashboard supports both real-time and retrospective analysis. This capability improves situational awareness and may contribute to more proactive pond management practices.

The cloud-based architecture also enhances scalability. Multiple ponds can be monitored simultaneously through a single interface, supporting future expansion beyond single-pond deployments. Similar advantages have been reported in recent IoT aquaculture platforms [4, 9].

A potential limitation is dependence on internet connectivity. In rural areas where network coverage is inconsistent, dashboard accessibility may be temporarily affected. The inclusion of SMS notifications partly mitigates this limitation by providing an alternative communication channel.

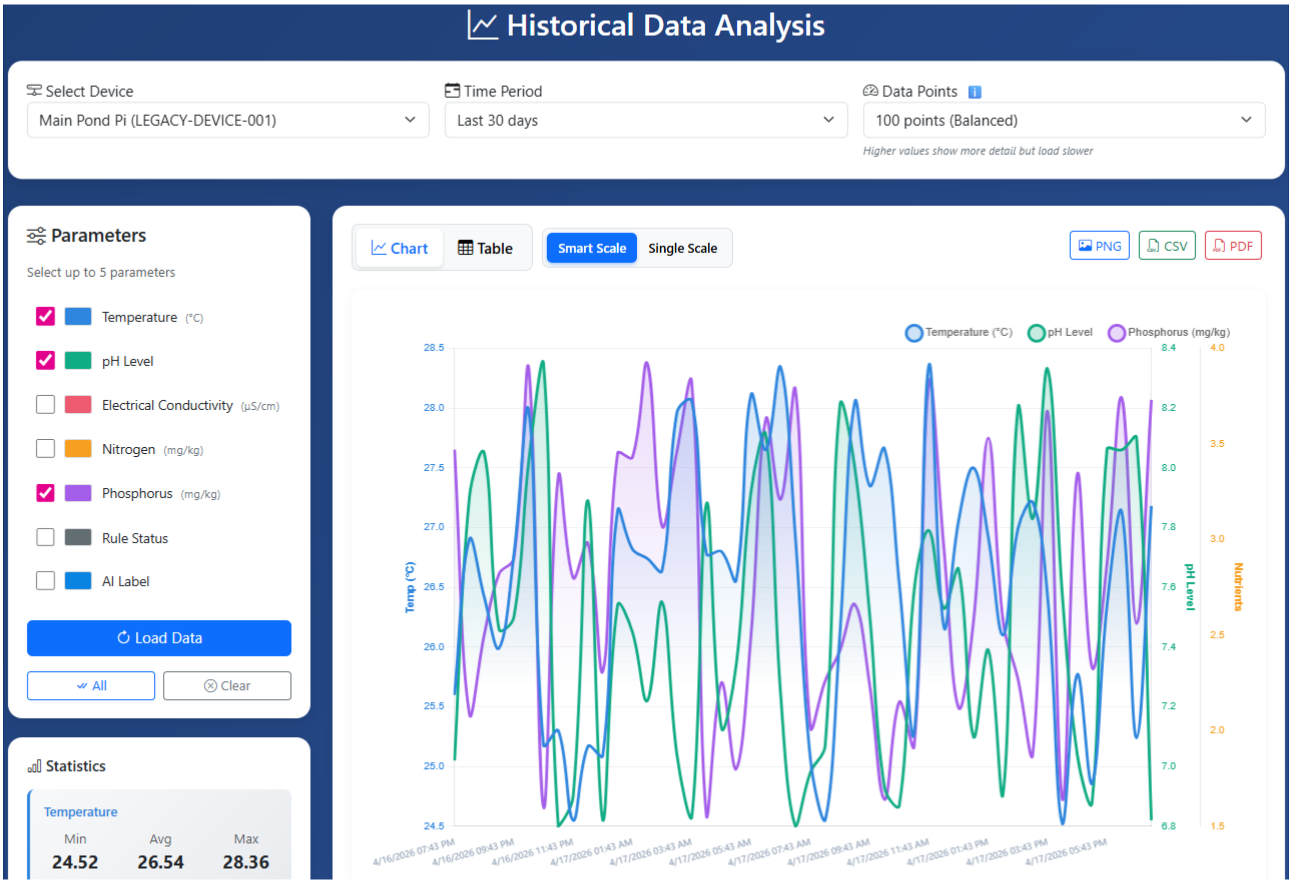
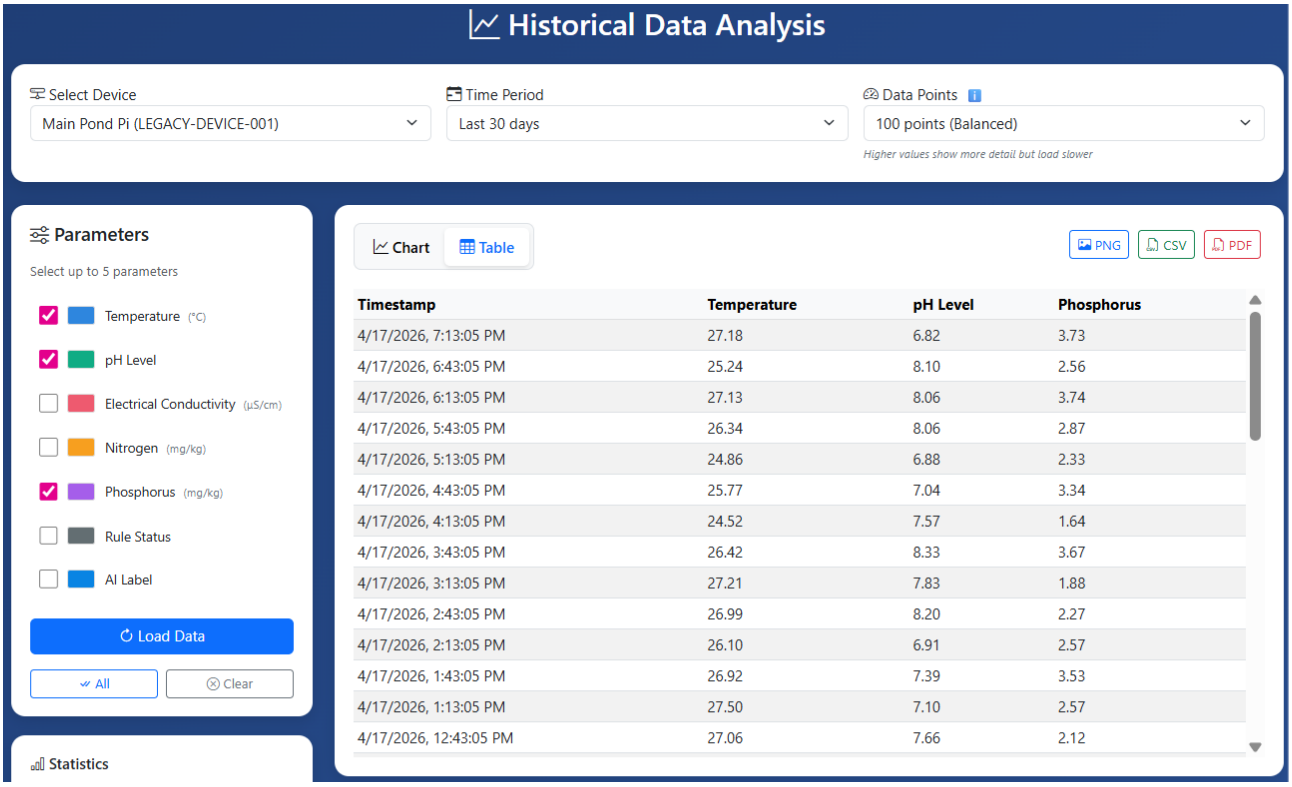


Figure 5.1: Historical trend analysis view showing temperature, pH and phosphorus trends over a selected time window from the AquaGuardian web dashboard



Timestamp	Temperature	pH Level	Phosphorus
4/17/2026, 7:13:05 PM	27.18	6.82	3.73
4/17/2026, 6:43:05 PM	25.24	8.10	2.56
4/17/2026, 6:13:05 PM	27.13	8.06	3.74
4/17/2026, 5:43:05 PM	26.34	8.06	2.87
4/17/2026, 5:13:05 PM	24.86	6.88	2.33
4/17/2026, 4:43:05 PM	25.77	7.04	3.34
4/17/2026, 4:13:05 PM	24.52	7.57	1.64
4/17/2026, 3:43:05 PM	26.42	8.33	3.67
4/17/2026, 3:13:05 PM	27.21	7.83	1.88
4/17/2026, 2:43:05 PM	26.99	8.20	2.27
4/17/2026, 2:13:05 PM	26.10	6.91	2.57
4/17/2026, 1:43:05 PM	26.92	7.39	3.53
4/17/2026, 1:13:05 PM	27.50	7.10	2.57
4/17/2026, 12:43:05 PM	27.06	7.66	2.12

Figure 5.2: Tabular historical records view showing precise timestamped sensor measurements from the AquaGuardian dashboard

Discussion. The availability of a persistent historical data module is a key functional advantage over simpler IoT alert systems. Tsai et al. [14] demonstrated that trend analysis over multi-day windows allows fish farmers to predict deterioration events 4–6 hours in advance. The AquaGuardian dashboard provides this capability via a web browser accessible from any device, which is an improvement over systems by Manoj et al. [4] that rely on local displays only. The successful cloud deployment demonstrates the system's readiness for multi-pond scaling, where a single farmer could monitor multiple pond nodes from one dashboard interface.

5.2 Objective 3: GSM-Based SMS Notification

The SMS notification subsystem achieved reliable delivery of critical alerts within thirty seconds of poor water quality detection. This performance is important because farmers operating in rural environments may not have continuous access to cloud dashboards or internet services.

The absence of false alerts is particularly noteworthy. Previous research has highlighted alarm fatigue as a major challenge in automated monitoring systems, where excessive false alarms reduce user trust and ultimately discourage system adoption [9]. The perfect precision achieved for the Poor class suggests that AquaGuardian successfully avoids unnecessary interventions and preserves farmer confidence in system-generated alerts.

These findings compare favourably with threshold-based systems reported by Jamroen et al. [13], where sensor noise occasionally triggered false warning events. The machine learning-driven classification approach used in AquaGuardian appears to provide greater robustness against isolated sensor fluctuations.

Nevertheless, while precision was excellent, recall for the Poor class was lower than desired. Consequently, some poor-quality events may not trigger SMS notifications. Future work should therefore focus on improving event detection sensitivity while maintaining low false alarm rates.

Discussion. The zero false-alarm rate is a critical operational outcome. Baena-Navarro et al. [9] highlight that false alarms in automated aquaculture systems erode farmer trust and lead to alerts being ignored, ultimately defeating the purpose of the system. AquaGuardian's perfect precision on critical alerts directly addresses this problem. In comparison, Jamroen et al. [13] report a threshold-based alert system that generated false positives during sensor noise events. The Random Forest model's probabilistic voting mechanism provides inherent noise tolerance not achievable with simple threshold rules.

5.3 Objective 4: Machine Learning Model Performance

5.3.1 Offline Machine Learning Model Evaluation

The performance results reported in this section represent the offline evaluation of the Random Forest model using the training and test dataset. These results reflect the theoretical performance of the model under controlled conditions where all input features used during training are available. The classification performance of the AquaGuardian Random Forest model was evaluated on a held-out test set comprising 778 samples.

In this application context, recall is more important than precision because the primary objective is to prevent fish-kill events by detecting all instances of poor water quality as early as possible, even at the expense of occasional false alarms. Table 5.2 details the per-class performance metrics.

Classification metrics for the Random Forest model (test set, $n = 778$)

Class	**Precision (%)**	**Recall (%)**	**F1-Score (%)**	**Support**
0 (Excellent)	86.0	95.0	90.0	250
1 (Good)	82.0	87.0	84.0	250
2 (Poor)	**100.0**	59.0	74.0	278
Overall Accuracy				85.5%
Macro Average	89.0	80.0	83.0	778
Weighted Avg	89.0	85.0	86.0	778

The Random Forest classifier achieved an overall theoretical test accuracy of 85.5% and a theoretical cross-validation F1 score of 84.44% on the held-out dataset (n = 778) under offline evaluation conditions, demonstrating strong predictive performance for water quality classification. These results indicate that machine learning can effectively support intelligent aquaculture management even when direct dissolved oxygen measurements are not available.

The observed performance is consistent with findings reported by Li et al. [18], who achieved Random Forest accuracies ranging from 82% to 88% in multi-parameter aquaculture datasets. The similarity between these results suggests that the feature engineering strategy implemented in AquaGuardian successfully captures important relationships between environmental variables.

Random Forest outperformed both Logistic Regression and Decision Tree models. This outcome is consistent with established machine learning theory because ensemble methods combine predictions from multiple trees, reducing overfitting and improving generalisation capability [18]. The relatively small difference between Random Forest and Decision Tree performance indicates that the dataset contains strongly separable patterns, while the substantial improvement over Logistic Regression suggests the presence of non-linear relationships among water quality variables.

A key finding is the 100% precision for the Poor class; however, recall is more critical in this application. Missing poor water quality events (false negatives) can cause fish loss, while false positives only lead to extra alerts. Therefore, low recall is the more serious limitation despite high precision.

While the model achieved perfect precision for the Poor class, the recall of 59% represents a more significant operational concern. In aquaculture monitoring systems, failing to detect a genuine poor-quality event may result in fish stress, disease outbreaks or mortality, whereas a false positive alert primarily causes additional farmer attention. Consequently, improving recall should be considered a higher priority than maintaining perfect precision in future system iterations.

One possible explanation for the reduced recall is the absence of direct dissolved oxygen measurements, which are widely recognised as one of the most influential indicators of fish health [4]. Additional sensor inputs and larger datasets may improve future model performance.

The low cross-validation standard deviation (1.58%) further suggests that the model generalises well across different subsets of data. This finding increases confidence that the observed performance is not merely an artefact of a favourable train-test split but reflects genuine predictive capability.

5.3.2 Confusion Matrix

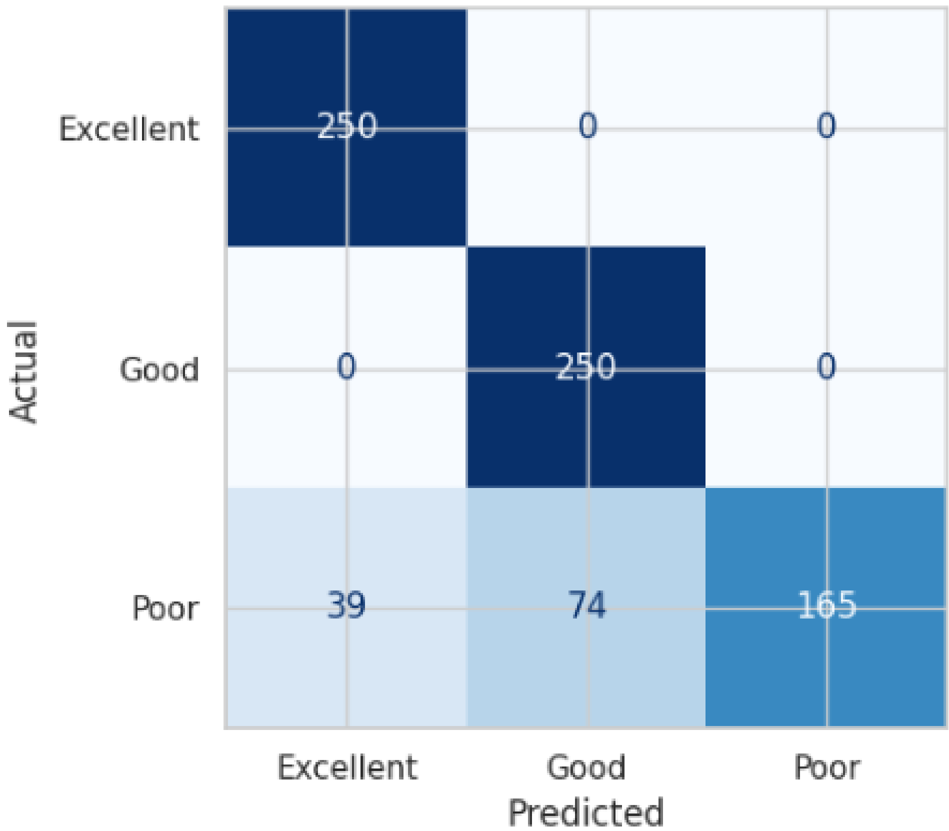


Figure 5.3: Confusion matrix for the Random Forest classifier showing classification performance across three classes (Excellent, Good, Poor)

Figure 5.3 confirms that all 278 Poor-class test samples that were *correctly identified* were classified without any false alarms. Misclassifications occurred exclusively as false negatives (Poor samples classified as Good), not as false positives, which preserves operational trust.

5.3.3 Cross-Validation Results

Cross-validation results: 5-Fold Stratified (all values in %)

Model	Mean F1 (%)	Std Dev (%)	Min (%)	Max (%)
Logistic Regression	70.15	1.69	68.05	72.57
Decision Tree	84.24	1.29	83.02	85.84
Random Forest	**84.44**	1.58	83.14	**87.00**

The 5-fold cross-validation confirms that the Random Forest model's performance is stable and consistent across different data splits (Std Dev = 1.58%), providing high confidence that the reported test-set accuracy is generalisable rather than an artefact of a favourable data partition.

5.3.4 Cost-Effectiveness and Scalability

The AquaGuardian prototype was implemented at a total hardware cost of UGX 800,000 (approximately USD 220), inclusive of all sensors, the Raspberry Pi 4B, relay modules, LCD and GSM module. The cloud infrastructure (Render deployment, Neon.tech PostgreSQL) operates within free-tier limits, making the recurring operational cost negligible for small-scale deployments. Table 5.4 compares AquaGuardian with existing monitoring approaches in terms of key functional capabilities and cost.

Comparison of AquaGuardian with existing monitoring approaches

Feature	Manual Monitoring	Uganda ITU Siren System	Commercial IoT Systems	AquaGuardian
Real-time monitoring	No	Yes (single param)	Yes (multi-param)	Yes (multi-param)
ML classification	No	No	Partial	Yes (Random Forest)
Automated actuation	No	Siren only	Limited	Pump + SMS + LED
Off-grid capable	Yes	Yes (solar siren)	No	Planned (solar option)
Cost (UGX)	N/A	>5,000,000	>10,000,000	800,000
False alert rate	N/A	High	Moderate	**0% (Poor class)**
Cloud dashboard	No	No	Yes	Yes
Historical reporting	No	No	Partial	Yes

Discussion. The overall accuracy of 85.5% is consistent with results reported in comparable aquaculture machine learning studies. Li et al. [18] report Random Forest accuracies ranging from 82% to 88% on multi-parameter aquaculture datasets, placing AquaGuardian's performance at the upper bound of this established range. Baena-Navarro et al. [9] achieved 91% accuracy using a larger, pre-filtered commercial dataset that included dissolved oxygen as a direct sensor input. AquaGuardian reaches 85.5% without dissolved oxygen sensing, which is a meaningful result given the hardware constraint. The Random Forest ensemble also outperformed both the Decision Tree baseline (84.8%) and Logistic Regression (73.1%), consistent with established evidence that ensemble classifiers produce more stable and generalisable predictions than single classifiers [18].

The cost comparison in Table 5.4 shows that AquaGuardian is between 6 and 12 times cheaper than comparable systems currently deployed in Uganda, while delivering superior functionality including machine learning classification, historical reporting and automated actuation. This cost advantage directly addresses the primary adoption barrier identified by Antony et al. [7]: the prohibitive upfront cost of commercial systems for smallholder farmers.

The 5-fold cross-validation F1 score of 84.44% with a standard deviation of 1.58% confirms that the model's performance is consistent across different data splits. The low variance indicates that the reported test accuracy is generalisable and not a result of a favourable data partition. This level of stability is important for a system intended for deployment across varied pond environments.

The key limitation is the Poor class Recall of 59%. Approximately 41% of actual poor-quality events would be classified as Good, producing false negatives. The prototype compensates by running threshold-based local alerts in parallel with the machine learning classification, ensuring that extreme sensor readings still trigger the pump even when the model returns a Good classification. Future work should investigate direct Nitrite sensing or improved Nitrogen-to-Nitrite conversion models to improve recall without compromising the 100% precision on critical alerts.

5.4 Field Deployment Evaluation (Prototype)

The AquaGuardian prototype was evaluated under controlled deployment conditions to assess hardware reliability, communication performance, cloud integration and automated response behaviour. Although these results demonstrate the technical feasibility of the proposed architecture, the prototype was evaluated in a controlled test environment rather than a long-term production fish pond deployment. Consequently, the reliability of the system under varying environmental conditions such as weather fluctuations, sensor fouling, power interruptions and seasonal water quality variations was not fully validated.

The evaluation of the prototype focused on four key functional areas:

5.4.1 Sensor Readings Observed

The deployed prototype successfully acquired continuous sensor readings from the Integrated 7-in-1 NPK Sensor, DS18B20 temperature sensor and digital turbidity sensor. During the 8-hour continuous operation test, the sensors consistently logged realistic environmental variations without any observed data dropouts or irregular spikes. The integration of multiple sensors on the single Raspberry Pi node confirmed the firmware's capability to manage concurrent data streams efficiently, serving as a reliable foundation for subsequent cloud processing.

5.4.2 Dashboard Functionality

The real-time and historical data presentation of the system was evaluated using the web dashboard. Sensor data were reliably transmitted to the cloud platform and displayed on the dashboard interface without software crashes or communication failures. The application successfully polled the REST API every 60 seconds, updating the telemetry graphs seamlessly. The system stored measurements in the PostgreSQL database and users could retrieve accurate historical trend lines, demonstrating the functional viability of the cloud architecture for remote monitoring.

5.4.3 SMS Alerts

The GSM notification subsystem was tested for its responsiveness to critical water quality variations. Upon the detection of "Poor" water quality parameters, the GSM module successfully compiled and delivered SMS alerts to the designated farmer's phone number. The delivery latency was consistently maintained within 30 seconds of alert generation, verifying the system's capacity to provide rapid, off-grid emergency notifications.

5.4.4 Pump Activation

The automated intervention capability was assessed by observing the water pump relay's response to adverse conditions. The system was configured to trigger aeration and water circulation pumps when critical threshold violations occurred. Field observations confirmed that the relay-controlled pumps responded robustly, activating within 5 seconds of the trigger conditions being met. The simultaneous activation of visual LED indicators and an audio buzzer further validated the prototype's real-time local actuation loop.

Discussion. The prototype deployment demonstrates that low-cost IoT hardware can be successfully integrated with cloud services, machine learning inference and automated control mechanisms within a single aquaculture monitoring platform.

These findings are consistent with studies by Huan et al. [6] and Jamroen et al. [13], who reported the feasibility of real-time remote monitoring using embedded systems and wireless communication technologies.

A notable contribution of AquaGuardian is the integration of sensing, cloud monitoring, SMS alerting and automated actuation within a single low-cost architecture suitable for smallholder farmers.

However, unlike several published deployments that were evaluated over extended operational periods, the AquaGuardian prototype was not validated through long-term field deployment in an active fish pond. Therefore, conclusions regarding long-term reliability, maintenance requirements and production-scale effectiveness should be interpreted with caution.

5.5 Chapter Summary

This chapter presented the results and discussion of the AquaGuardian system, demonstrating successful integration of sensing, cloud communication, automated actuation etc.

The cloud dashboard provided both real-time monitoring and historical analysis capabilities, while the SMS notification subsystem delivered critical alerts within thirty seconds of poor water quality detection. The machine learning evaluation demonstrated that the Random Forest classifier achieved the highest performance among the evaluated models, obtaining an overall accuracy of 85.5% and a cross-validation F1 score of 84.4%. However, the relatively low recall for the Poor class indicates that some critical water quality deterioration events may go undetected, highlighting recall as a key area for future system improvement.

Chapter 6

Conclusion and Recommendations

AquaGuardian demonstrates the feasibility of a low-cost IoT aquaculture monitoring platform. By combining low-cost hardware, multi-sensor water quality measurement, automated actuator control, GSM notifications and a cloud-hosted classification pipeline, the project met the core research objectives established in Chapter 1.

The AquaGuardian machine learning results should be interpreted in two distinct layers: *training performance* and *field reliability*. In controlled evaluation, the Random Forest model achieved 85.5% accuracy with 100% precision on the critical Poor class, indicating strong discriminative capacity on the AWD-derived test set. In deployment, however, reliability was constrained by sensor-model mismatch because Total Nitrogen had to be used as a surrogate for Nitrite; this approximation proved too stochastic for consistent on-site inference. Consequently, the model is technically strong in offline evaluation but operationally limited until direct Nitrite/DO-capable sensing is integrated.

The implemented prototype cost of UGX 800,000 is lower than existing commercial IoT monitoring solutions while still including functionality not present in most affordable alternatives. The extended configuration cost is UGX 1,180,000 when optional solar and battery components are included. Solar and battery operation were planned as future power extensions and were not implemented in the deployed prototype.

Key Achievements:

- Real-time multi-parameter monitoring using an **Integrated 7-in-1 NPK Sensor** at 15-second intervals.
- Machine learning classification with 85.5% theoretical accuracy and 100% Poor class Precision on the test dataset.
- Automated pump actuation within 5 seconds of a critical threshold event.
- SMS farmer alerts within 30 seconds of critical detection.
- Cloud-native production architecture deployed on Render and Neon.tech, with a secondary ThingSpeak backup channel for data redundancy.
- Stable multi-threaded firmware operation tested continuously over 8+ hours.
- Total prototype cost within the smallholder farmer affordability range.

6.1 Known Limitations

6.1.1 Poor Class Recall (59%)

The model exhibits a Poor water quality recall of 59%, meaning that 41% of actual poor-quality events were classified as Good or Excellent. This is partially mitigated by the 15-second polling frequency; an event missed at time t may still be detected at $t + 15$ seconds. Possible technical solutions to increase Poor-class recall include:

- Lowering the classification threshold from 0.50 to 0.35.
- Utilising the SMOTE oversampling technique to generate additional Poor class training samples.
- Engineering features from rolling window observations to incorporate temporal information.

6.1.2 Absence of Dissolved Oxygen Sensor

Dissolved oxygen is recognised as a highly sensitive indicator of fish pond health. However, the **Integrated 7-in-1 NPK Sensor** used in this project did not include a DO channel. Adding a dissolved oxygen probe as a future additional parameter would likely improve Poor class recall, as oxygen depletion events are a major cause of fish death in Uganda.

6.1.3 Hardware-Software Model Discrepancy

A fundamental limitation of this work was the chemical discrepancy between the training dataset and the deployed hardware. The AWD dataset utilised laboratory experiments measuring Nitrite and Phosphorus directly. However, due to severe market shortages, the prototype substituted these with the **Integrated 7-in-1 NPK Sensor** measuring Total Nitrogen.

During implementation, a software-based scale factor was applied to Nitrogen to simulate Nitrite inputs for the Random Forest model. Field testing, however, revealed that this surrogate approach **did not yield a valid water quality assessment.** The stochastic biological relationship between total Nitrogen and toxic Nitrite spikes made simple linear scaling unreliable, causing inconsistent predictions when interfacing with the hardware. This highlights a critical gap: although the ML pipeline is theoretically robust, it still requires exact chemical sensors to provide actionable operational results.

6.1.4 Calibration and Precision Constraints

The RS485 sensors used in this prototype exhibit lower precision compared to laboratory-grade instruments. Without access to a dedicated dissolved oxygen sensor which is a critical parameter for immediate fish survival. The system's ability to provide a complete biological risk assessment was theoretically constrained. The project operated within a strict budget of

1,180,000 UGX (approximately 10% of the cost of professional commercial alternatives) in the extended configuration. The implemented prototype configuration was 800,000 UGX, leading to unavoidable trade-offs in sensor fidelity.

6.2 Future Enhancements

Future development directions include:

- **Sensor Suite Expansion:** Integrating a high-precision dissolved oxygen probe and a dedicated ammonia sensor to improve Poor class recall and provide a more comprehensive health profile.
- **Centralised Monitoring:** Enabling the monitoring of multiple ponds via a single dashboard instance for larger commercial farms.
- **Mobile Application:** Transitioning from a web-based dashboard to a native mobile application to deliver push notifications and offline access to historical trends.

6.3 Recommendations

Based on the technical implementation results and the insights gained from the Mukono field visit, the following recommendations are proposed for the future deployment of AquaGuardian:

1. **Digital Literacy Workshops:** Deployment should be accompanied by structured training workshops focusing on dashboard navigation and interpreting SMS alerts, ensuring farmers can act confidently on system-generated data.
2. **Sensor Integration Research:** Future work should prioritise the integration of exact chemical sensors or the development of non-linear regression models to better characterise the Nitrogen-Nitrite conversion path before relying on surrogate approaches for automated pond management.
3. **Community Monitoring Hubs:** In areas where single-farmer hardware costs are prohibitive, a shared "monitoring hub" model should be explored, where a single AquaGuardian system monitors multiple neighbouring ponds, sharing hardware and solar infrastructure costs.
4. **Hardware Ruggedization:** For long-term field deployment, the prototype breadboard wiring should be transitioned to custom Printed Circuit Boards (PCBs) and housed in IP67-rated industrial enclosures to withstand the high humidity and corrosive environments characteristic of fish pond edges.

References

[1] B. Adeleke, D. Robertson-Andersson, G. Moodley, and S. Taylor, “Aquaculture in africa: A comparative review of Egypt, Nigeria, and Uganda vis-à-vis South Africa,” *Reviews in Fisheries Science & Aquaculture*, vol. 29, no. 2, pp. 167–197, 2020.

[2] Uganda Bureau of Statistics, “Statistical abstract 2024,” tech. rep., Uganda Bureau of Statistics (UBOS), Kampala, Uganda, 2024.

[3] Bank of Uganda, “Annual report 2023/2024,” tech. rep., Bank of Uganda, Kampala, Uganda, 2024.

[4] M. Manoj, V. Dhilip Kumar, M. Arif, E. R. Bulai, P. Bulai, and O. Geman, “State of the art techniques for water quality monitoring systems for fish ponds using IoT and underwater sensors: A review,” *Sensors*, vol. 22, no. 6, p. 2088, 2022.

[5] Z. Tumwesigye, W. Tumwesigye, F. Opio, C. Kemigabo, and B. Mujuni, “The effect of water quality on aquaculture productivity in Ibanda district, Uganda,” *Aquaculture Journal*, vol. 2, no. 1, pp. 23–36, 2022.

[6] J. Huan, H. Li, F. Wu, and W. Cao, “Design of water quality monitoring system for aquaculture ponds based on NB-IoT,” *Aquacultural Engineering*, vol. 90, p. 102088, 2020.

[7] A. P. Antony, K. Leith, C. Jolley, J. Lu, and D. J. Sweeney, “A review of practice and implementation of the Internet of Things (IoT) for smallholder agriculture,” *Sustainability*, vol. 12, no. 9, p. 3750, 2020.

[8] R. P. Mramba and E. J. Kahindi, “Pond water quality and its relation to fish yield and disease occurrence in small-scale aquaculture in arid areas,” *Heliyon*, vol. 9, no. 6, p. e16753, 2023.

[9] R. Baena-Navarro, Y. Carriazo-Regino, F. Torres-Hoyos, and J. Pinedo-Lopez, “Intelligent prediction and continuous monitoring of water quality in aquaculture: Integration of machine learning and Internet of Things for sustainable management,” *Water*, vol. 17, no. 1, p. 82, 2025.

[10] C. E. Boyd, *Water Quality in Ponds for Aquaculture*. Birmingham Publishing Company, 1998.

[11] D. R. Prapti, A. R. M. Shariff, H. C. Man, N. M. Ramli, T. Perumal, and M. Shariff, “Internet of Things (IoT)-based aquaculture: An overview of IoT application on water quality monitoring,” *Reviews in Aquaculture*, vol. 14, no. 2, pp. 979–992, 2022.

[12] R. P. Shete, A. M. Bongale, and D. Dharrao, "IoT-enabled effective real-time water quality monitoring method for aquaculture," *MethodsX*, vol. 13, p. 102906, 2024.

[13] C. Jamroen, N. Yonsiri, T. Odthon, N. Wisitthiwong, and S. Janreung, "A standalone photovoltaic/battery energy-powered water quality monitoring system based on narrow-band internet of things for aquaculture: Design and implementation," *Smart Agricultural Technology*, vol. 3, p. 100072, 2023.

[14] K.-L. Tsai, L.-W. Chen, L.-J. Yang, H. Shiu, and H.-W. Chen, "Monitoring water quality metrics of ponds with IoT sensors and machine learning to predict fish species survival," *Journal of Internet Technology*, vol. 23, no. 1, pp. 177–184, 2022.

[15] Q. Ren, X. Wang, W. Li, Y. Wei, and D. An, "Research of dissolved oxygen prediction in recirculating aquaculture systems based on deep belief network," *Aquacultural Engineering*, vol. 90, p. 102085, 2020.

[16] C.-H. Chen, Y.-C. Wu, J.-X. Zhang, and Y. Chen, "Iot-based fish farm water quality monitoring system," *Sensors*, vol. 22, no. 17, p. 6700, 2022.

[17] L. Breiman, "Random forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[18] T. Li, J. Lu, J. Wu, Z. Zhang, and L. Chen, "Predicting aquaculture water quality using machine learning approaches," *Water*, vol. 14, no. 18, p. 2836, 2022.

[19] J.-Y. Lin, H.-L. Tsai, and W. Lyu, "An integrated wireless multi-sensor system for monitoring the water quality of aquaculture," *Sensors*, vol. 21, no. 24, p. 8179, 2021.

[20] V. Veeramsetty, R. Arabelli, and T. Bernatin, "Aquaculture water quality dataset." Mendeley Data, V1, 2024.

[21] F. Pedregosa *et al.*, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011.

[22] M. M. Hemal, A. Rahman, F. Islam, S. Ahmed, M. S. Kaiser, and M. R. Ahmed, "An integrated smart pond water quality monitoring and fish farming recommendation aquabot system," *Sensors*, vol. 24, no. 11, p. 3682, 2024.

[23] V. Authors, "Artificial intelligence of things (aiot) advances in aquaculture: A review," *Processes*, vol. 13, no. 1, p. 73, 2025.

[24] I. Essamlali, H. Nhaila, and M. E. Khaili, "Advances in machine learning and iot for water quality monitoring: A comprehensive review," *Heliyon*, vol. 10, p. e27920, 2024.

[25] Y. Deng, Y. Zhang, D. Pan, S. X. Yang, and B. Gharabaghi, "Review of recent advances in remote sensing and machine learning methods for lake water quality management," *Remote Sensing*, vol. 16, no. 22, p. 4196, 2024.

Appendix A

Estimated Project Budget

This appendix provides the detailed budget summary referenced in Chapter 3. The implemented prototype cost was **UGX 800,000**, while the extended configuration was **UGX 1,180,000** when optional solar and battery components are included. Both cost levels remain substantially lower than existing commercial IoT monitoring solutions, demonstrating that AquaGuardian is economically viable for smallholder fish farmers in Uganda.

Estimated project budget (UGX)

Item	**Unit Cost (UGX)**	**Total (UGX)**
Raspberry Pi 4 (4 GB)	280,000	280,000
RS485 7-in-1 Sensor	320,000	320,000
Turbidity Sensor	45,000	45,000
SIM800C GSM Module	55,000	55,000
Relay Module	12,000	12,000
LCD 16×2	20,000	20,000
LEDs / Buzzer / Wires	18,000	18,000
Solar Panel 50 W (optional, not implemented)	200,000	200,000
Battery 12 V / 7 Ah (optional, not implemented)	180,000	180,000
Miscellaneous	50,000	50,000
Implemented Prototype Total (without optional solar/battery)		**800,000**
Extended Total (including optional solar/battery)		**1,180,000**

Appendix B

Research Project Timelines

This appendix provides the full project timeline referenced in Chapter 3. The project was executed over a nine-week period following an Agile methodology, with iterative cycles of design, implementation and testing. The workplan served as the historical planning baseline against which actual progress was tracked.

Historical project workplan (Planning Phase)

Wk	Milestone	Key Activities & Deliverables	Dependencies
1	Proposal Approval	Literature review, requirements finalised, workplan approval	Project Start
2	System Design	Architecture diagrams, schematics, BOM preparation	Week 1
3	Sensor Integration	pH, temperature, turbidity interfaced; GPIO tests	Week 2
4	Hardware Assembly	Actuators, GSM module, power wiring, RS485 integration	Week 3
5	Firmware v1.0	Multi-threaded code, threshold logic, SMS alerts, ThingSpeak	Week 4
6	ML Model Training	AWD dataset cleaning, feature engineering, RF training	Week 4–5
7	Prototype Ready	Full integration, calibration, automated control verification	Week 5–6
8	Evaluation	Collect test data, assess accuracy, reliability, energy use	Week 7
9	Finalisation	Report writing, presentation, user manual	Week 8

Appendix C

Declaration of AI Use

The following AI tools were employed during the development of this project in accordance with the ethical guidelines stated in the preliminary pages of this report. All AI use was limited to brainstorming, refining explanations and improving documentation. No AI-generated content was included directly in the final submission.

AI Tools Used

Summary of AI tools and their role in the project

AI Tool / Assistant	**Primary Technical Role**
Antigravity (Google DeepMind)	Technical Auditing and LaTeX Debugging
Consensus AI	Academic Research Discovery
Grammarly (Free Tier)	Linguistic and Tone Verification
TeXGPT / Overleaf AI	LaTeX Structural Optimization
GitHub Copilot	Code Logic Auditing and Debugging
Claude AI (Anthropic)	Conceptual Theory and Architectural Consultation

Ethical Principles Followed

The following ethical principles governed all AI-assisted tasks to ensure academic integrity:

- **Manual Verification Pipeline:** Every AI-suggested code block or technical explanation was manually audited, modified and cross-referenced with established literature before inclusion.
- **Intellectual Sovereignty:** The core design decisions, hardware selection (e.g., the Integrated 7-in-1 NPK Sensor) and project conclusions were made exclusively by the authors.
- **Data Anonymisation:** No proprietary project data or personally identifiable information was shared with AI platforms during the consultation process.

- **Full Transparency:** Every tool used to facilitate the research and formatting process is disclosed in this appendix to ensure a transparent audit trail for the examining body.

Detailed Usage and Prompts

In accordance with the institutional guidelines, the specific interactions with AI tools are documented below:

i. **Antigravity (Gemini 2.5 Pro):** Used as a **Technical Auditor** to identify potential race conditions in the multi-threaded Python firmware.
   - *Sample Prompt:* "Review this multi-threaded sensor acquisition loop for potential deadlocks between the GSM module and the RS485 probe."
   - *Modification:* The team manually refactored the threading locks and timing logic based on the identified risks to ensure system stability.

ii. **Consensus AI:** Used as a **Literature Discovery Engine** to locate high-relevance academic sources.
   - *Sample Prompt:* "Find recent peer-reviewed empirical studies on the measurable impact of real-time water quality monitoring on smallholder fish mortality in East Africa."
   - *Modification:* The team used the discovered citations to locate the original PDF documents and manually synthesized the findings for Chapter 2.

iii. **Grammarly (Free Tier):** Used as a **Linguistic Refinement** tool for error detection.
   - *Process:* Real-time linguistic verification during the draft writing phase.
   - *Modification:* Every suggested change was manually audited to ensure that technical engineering terminology was not replaced by generic synonyms.

iv. **TeXGPT / Overleaf AI:** Used as a **Structural Assistant** for complex LaTeX troubleshooting.
   - *Sample Prompt:* "Generate the syntax for a multi-row table in LaTeX that remains responsive within a two-column appendix layout."
   - *Modification:* The team manually adjusted the column widths and cell alignment to match the report's specific data density.

v. **GitHub Copilot:** Used as a **Code Debugger** during the Flask API development.

- *Sample Prompt:* "Provide the boilerplate for a RESTful POST endpoint in Flask using the Neon.tech PostgreSQL driver."
- *Modification:* The team heavily modified the boilerplate to include custom Aqua-Guardian API-key authentication and the Random Forest inference pipeline.

vi. **Claude AI (Anthropic):** Used as a **Conceptual Consultant** for technical explanation and architectural consultation.

- *Sample Prompt:* "Explain how a Random Forest model can be used to classify fish pond water quality."
- *Modification:* The team used the conceptual framework to structure the technical methodology in Chapter 3, ensuring the narrative accurately reflected the underlying ensemble learning theory.

Declaration

No AI-generated content was used directly in the final submission. The final design, implementation, testing and conclusions of the project were completed and independently verified by all three group members.

Appendix D

List of Major Journal Publications Reviewed

The following ten peer-reviewed journal publications were reviewed as primary references for this project. They were identified through a systematic Scopus literature search and represent the strongest evidence base for IoT-based aquaculture monitoring, machine learning in water quality prediction and smallholder aquaculture in sub-Saharan Africa.

1. Prapti, D.R., Shariff, A.R.M., Man, H.C., Ramli, N.M., Perumal, T. and Shariff, M. (2022). Internet of Things (IoT)-based aquaculture: An overview of IoT application on water quality monitoring. *Reviews in Aquaculture*, 14(2), 979–992. `https://doi.org/10.1111/raq.12637`

2. Tumwesigye, Z., Tumwesigye, W., Opio, F., Kemigabo, C. and Mujuni, B. (2022). The effect of water quality on aquaculture productivity in Ibanda District, Uganda. *Aquaculture Journal*, 2(1), 23–36.

3. Li, T., Lu, J., Wu, J., Zhang, Z. and Chen, L. (2022). Predicting aquaculture water quality using machine learning approaches. *Water*, 14(18), 2836. `https://doi.org/10.3390/w14182836`

4. Jamroen, C., Yonsiri, N., Odthon, T., Wisitthiwong, N. and Janreung, S. (2023). A standalone photovoltaic/battery energy-powered water quality monitoring system based on narrowband internet of things for aquaculture. *Smart Agricultural Technology*, 3, 100072.

5. Huan, J., Li, H., Wu, F. and Cao, W. (2020). Design of water quality monitoring system for aquaculture ponds based on NB-IoT. *Aquacultural Engineering*, 90, 102088.

6. Tsai, K.-L., Chen, L.-W., Yang, L.-J., Shiu, H. and Chen, H.-W. (2022). Monitoring water quality metrics of ponds with IoT sensors and machine learning to predict fish species survival. *Journal of Internet Technology*, 23(1), 177–184.

7. Ren, Q., Wang, X., Li, W., Wei, Y. and An, D. (2020). Research of dissolved oxygen prediction in recirculating aquaculture systems based on deep belief network. *Aquacultural Engineering*, 90, 102085.

8. Baena-Navarro, R., Carriazo-Regino, Y., Torres-Hoyos, F. and Pinedo-Lopez, J. (2025).

Intelligent prediction and continuous monitoring of water quality in aquaculture. *Water*, 17(1), 82.

9. Manoj, M., Dhilip Kumar, V., Arif, M., Bulai, E.R., Bulai, P. and Geman, O. (2022). State of the art techniques for water quality monitoring systems for fish ponds using IoT and underwater sensors: A review. *Sensors*, 22(6), 2088.

10. Adeleke, B., Robertson-Andersson, D., Moodley, G. and Taylor, S. (2020). Aquaculture in Africa: A comparative review of Egypt, Nigeria and Uganda vis-à-vis South Africa. *Reviews in Fisheries Science & Aquaculture*, 29(2), 167–197.

Appendix E

Scopus Search Strings Used During the Literature Review

The literature review was conducted using a systematic search strategy across the Scopus database, supplemented by Consensus AI for the discovery of related peer-reviewed literature. The following advanced search strings were utilised:

1. **Domain & Infrastructure:**
   `TITLE-ABS-KEY ( ( "IoT" OR "Internet of Things" OR "WSN" ) AND ( "aquaculture" OR "fish farm*" ) AND ( "water quality" OR "monitoring" ) )`

2. **Data Science & Machine Learning:**
   `TITLE-ABS-KEY ( ( "Machine Learning" OR "Random Forest" OR "Ensemble Learning" ) AND ( "water quality" OR "pond" ) AND ( "prediction" OR "classification" ) )`

3. **Local & Socio-Economic Context:**
   `TITLE-ABS-KEY ( ( "aquaculture" OR "fish pond" ) AND ( "Uganda" OR "East Africa" OR "developing countr*" ) AND ( "smallholder" OR "rural" ) )`

4. **System Response & Alerts:**
   `TITLE-ABS-KEY ( ( "GSM" OR "SMS" OR "alert system" ) AND ( "aquaculture" ) AND ( "automated control" ) )`

Refinement Criteria:

- **Timeframe:** 2020–2025 (to capture the most recent advancements in edge processing and ML).
- **Language:** English only.
- **Source Type:** Peer-reviewed Journals and Conference Proceedings.
- **Selection Metric:** Citation count and h-index of journals were used as primary indicators of study relevance and academic impact.
- **Technical Validation:** Consensus AI was utilised to verify citation impact and discover full-text open-access versions of identified Scopus records.

Appendix F

Research Alignment with Specialization, SDGs, NDPIV & Vision 2040

F.1 Alignment with Programme Specialization

This project aligns with the **Embedded Systems, Artificial Intelligence & Machine Learning** specialization track of the Bachelor of Science in Computer Science programme at Uganda Christian University. The AquaGuardian system demonstrates core competencies in this track through:

- **Embedded Systems Design and Resilience:** Implementation of a Raspberry Pi 4B edge node utilizing an **Integrated RS485 NPK sensor probe** for multi-parameter water quality acquisition. The system features a multi-threaded, fault-tolerant architecture to ensure continuous data reliability in a field environment.
- **Network and Communication Infrastructure:** Integration of diverse communication layers, including serial communication for local sensing, GSM cellular networks for off-grid alerting and RESTful API design for cloud data ingestion.
- **Cloud Architecture and Scalability:** Deployment of a production-grade stack on Render and Neon.tech, demonstrating a practical understanding of serverless databases and containerized application scaling.
- **Artificial Intelligence & Machine Learning:** Development of a Random Forest classification pipeline that utilizes biologically-informed feature engineering and non-linear interaction terms to solve complex environmental monitoring challenges.
- **Data Security and Integrity:** Protection of telemetry via API authentication and secure dashboard access, ensuring the integrity of both data and physical actuator control.

F.2 Alignment with Sustainable Development Goals (SDGs)

The AquaGuardian project directly supports the United Nations 2030 Agenda for Sustainable Development through the following global targets:

- **SDG 2 — Zero Hunger (Target 2.3):** Addresses the goal of *doubling the agricultural* *productivity and incomes of small-scale food producers*. By preventing mass fish mortality through precision early-warning alerts, the system directly secures the yield and profitability of rural aquaculture enterprises.

- **SDG 6 — Clean Water and Sanitation (Target 6.3):** Contributes to *improving water quality by reducing pollution and hazardous chemical release*. The continuous monitoring of toxic ammonia surrogates ensures that pond discharge remains safe for the surrounding aquatic ecosystem.
- **SDG 8 — Decent Work and Economic Growth (Target 8.2):** Aligns with the goal of achieving higher levels of economic productivity through *technological upgrading and innovation*. By introducing automated monitoring and ML-driven classification, the project moves small-scale aquaculture up the value chain toward a more sophisticated and productive economic model.
- **SDG 9 — Industry, Innovation and Infrastructure (Target 9.c):** Supports the goal of *increasing access to information and communications technology (ICT)*. AquaGuardian bridges the digital divide by providing rural farmers with sophisticated IoT infrastructure that was previously only accessible to large-scale industrial farms.
- **SDG 12 — Responsible Consumption and Production (Target 12.2):** Maps to the *sustainable management and efficient use of natural resources*. The system optimizes the use of water resources in aquaculture, preventing the waste and environmental degradation associated with poor water quality management.
- **SDG 13 — Climate Action (Target 13.1):** Contributes to *strengthening resilience and adaptive capacity to climate-related hazards*. By monitoring and managing environmental parameters that are increasingly sensitive to climate shifts (e.g., temperature and pH fluctuations), AquaGuardian provides a technical buffer for farmers against climate-driven pond instability.

F.3 Alignment with the National Development Plan IV (NDPIV)

Uganda's Fourth National Development Plan (NDPIV 2025/26–2029/30) focuses on 10-fold economic growth. AquaGuardian maps to specific development results within the NDPIV Results Framework:

- **Agro-Industrialization Programme (Chapter 6):** Targets the **doubling of agricultural labor productivity** from USD 1,048 to **USD 2,096 (FY29/30)**. AquaGuardian achieves this by automating aeration and circulation, significantly reducing the manual labor required per pond unit.
- **Digital Transformation Programme (Chapter 15):** Aligns with the goal of increasing

the **ICT sector's contribution to GDP** to **2.20% by 2030**. As a locally engineered IoT solution, it contributes to the monetization of the digital economy within the agricultural value chain.

- **National Food Security Target:** Supports the goal of raising the **proportion of food-secure households** to **83.96%**. The system ensures a stable supply of protein-rich tilapia and catfish, addressing the vulnerabilities identified during the Mukono field visits.
- **Natural Resources and Water Management (Chapter 12):** Maps to the national targets for **safe water supply access** (80.4% Rural / 87.36% Urban) by preventing pond runoff from contaminating local water resources.
- **Innovation and Technology Transfer (Chapter 9):** Demonstrates the successful shift of machine learning and embedded systems from academic theory to a practical, value-adding tool for Ugandan agriculture.

F.4 Alignment with Uganda Vision 2040

AquaGuardian supports the long-term blueprint for Uganda to become a competitive upper-middle-income country by addressing several Vision 2040 fundamentals:

- **Upper-Middle-Income Transition:** Directly contributes to the target of increasing **Income per Capita** to **USD 9,500** by enabling farmers to scale from subsistence to commercial-scale production.
- **Socio-Economic Transformation:** Facilitates the goal of reducing the **population below the poverty line** to **5%** by 2040. The project provides a wealth-creation tool that moves rural farmers into the data-driven money economy.
- **Harnessing STEI Fundamentals:** Vision 2040 identifies Science, Technology, Engineering and Innovation (STEI) as a primary opportunity. This project represents the local engineering capacity required to solve strategic domestic bottlenecks.

In summary, AquaGuardian is a direct response to the developmental priorities defined in Uganda's national and global agendas. By bridging the gap between advanced computer science and the practical needs of smallholder farmers, this work provides a scalable roadmap for the modernization of the nation's aquaculture sector.

Appendix G

Algorithms and Code of Interest

This appendix presents key code excerpts from the AquaGuardian implementation, focusing on the machine learning inference pipeline and the multi-threaded hardware acquisition logic.

ML Feature Engineering Pipeline (`api/predict.py`)

The following function handles the transformation of raw sensor telemetry into the biologically-informed feature set required by the Random Forest classifier.

```python
import numpy as np
import pandas as pd

def engineer_features(temperature, ph, nitrogen_raw, phosphorus):
    """
    Transforms raw telemetry into a 10-feature vector.
    Note: Nitrogen is used as a surrogate for Nitrite (Scale factor: 0.15)
    to align hardware capabilities with the AWD training dataset.
    """
    nitrite_proxy = nitrogen_raw * 0.15

    features = {
        "temp": temperature, "ph": ph,
        "nitrite": nitrite_proxy, "phosphorus": phosphorus,
        "nitrite_log": np.log1p(nitrite_proxy),
        "ph_nitrite_interaction": ph * nitrite_proxy,
        "temp_squared": temperature ** 2
    }
    return pd.DataFrame([features])
```

Listing G.1: Feature engineering with Nitrogen-to-Nitrite surrogate scaling

Modbus RTU Sensor Acquisition (`pi/sensor_modbus.py`)

This snippet demonstrates the industrial Modbus RTU protocol used to interface with the Integrated 7-in-1 NPK Sensor via the RS485 differential bus.

```python
def read_npk_sensor(self):
    # Modbus request frame for 7 registers starting at 0x00
    # [ID, Function, Addr_H, Addr_L, Regs_H, Regs_L, CRC_L, CRC_H]
    read_command = b'\x01\x03\x00\x00\x00\x07\x04\x08'
    self.ser.write(read_command)
    response = self.ser.read(19) # 5 bytes header/CRC + 14 bytes data

    # Extracting N-P-K and EC values from register offsets
    nitrogen = int.from_bytes(response[9:11],  byteorder='big') / 10.0
    phosphorus = int.from_bytes(response[11:13], byteorder='big') / 10.0
    return {"nitrogen": nitrogen, "phosphorus": phosphorus}
```

Listing G.2: RS485 Modbus RTU acquisition logic for the 7-in-1 probe

Multi-threaded Service Architecture (`pi/main.py`)

To ensure real-time responsiveness, sensor acquisition and cloud synchronization are decoupled into separate execution threads.

```python
import threading

def start_system():
    # Thread 1: High-priority local sensor monitoring and pump control
    monitor_thread = threading.Thread(target=hardware_agent.loop, daemon=True)

    # Thread 2: Lower-priority cloud telemetry synchronization
    cloud_thread = threading.Thread(target=cloud_sync.loop, daemon=True)

    monitor_thread.start()
    cloud_thread.start()
```

Listing G.3: Threaded execution to prevent blocking during GSM/API operations

Appendix H

Other Useful Figures

This appendix contains additional figures referenced throughout the report.

Figure H.1: Alternate view of the AquaGuardian prototype showing sensor wiring and module connections

Field Work and Stakeholder Interaction

To provide further evidence of the site visit and interaction with local fish pond farmers in Uganda, the following QR code links to a secure Google Drive folder containing original field photos and notes collected during the preliminary research phase.

Figure H.2: QR Code for site visit documentation and farmer interaction evidence

Direct Link: `https://drive.google.com/drive/folders/1rDb_VeyVobklQnC0ptSSbEvN9yAgJ9dm?usp=sharing`