Design and Implementation of a Low-Power Edge-WSN Architecture for Mangrove Health Monitoring in Restoration Sites

1st K. Arun Delan Raj
Dept of Electrical and Computer Engineering
Faculty of Engineering Technology
The Open University of Sri Lanka
s14014121@ousl.lk

2nd Eng. D.S.Wickramasinghe
Senior Lecturer *Dept of Electrical and Computer Engineering*
Faculty of Engineering Technology
The Open University of Sri Lanka
dswic@ou.ac.lk

***Abstract*— This research presents an AI-ready, scalable, low-power wireless sensor network (WSN) framework for smart environmental monitoring in remote and ecologically sensitive coastal areas, with a focus on mangrove replantation sites impacted by plastic pollution and climate change. The system integrates custom-designed ESP32 microcontroller-based master and slave sensor nodes powered by solar energy, capable of measuring key environmental parameters including sea water level, total dissolved solids (TDS), pH, and temperature, while supporting AI-driven analytics for predictive ecosystem health assessment. Communication between nodes is achieved using LoRa on the free ISM band, employing time-slot scheduling, multi-path routing, and acknowledgment-based protocols to ensure reliable long-range, low-power data delivery. Edge computing capabilities allow on-node data preprocessing with statistical filtering, event-based reporting, and adaptive transmission to optimize bandwidth and energy consumption. A cloud-based Firebase dashboard enables real-time visualization, historical trend analysis, and integration with AI models for anomaly detection and early-warning alerts. By linking water quality monitoring with plastic pollution impact assessment and providing a robust, low-maintenance, and intelligent monitoring infrastructure, this system offers a powerful tool for advancing ecosystem restoration, sustainability management, and informed, data-driven decision-making.**

I. Introduction

Coastal ecosystems such as mangroves play a critical role in biodiversity conservation [1], carbon sequestration, and shoreline protection, while also supporting fisheries, tourism, and local livelihoods. They act as natural barriers against storm surges, mitigate coastal erosion, and provide essential ecosystem services for climate change adaptation. However, these ecosystems face mounting threats from rising sea levels, extreme weather events, and anthropogenic pressures, particularly plastic pollution.

Plastic debris especially micro plastics can infiltrate mangrove soils and waterways, altering sediment structure, affecting nutrient cycling, and disrupting plant growth processes. These contaminants can influence water quality parameters such as pH, salinity, and TDS, indirectly impacting mangrove seedling survival. In Sri Lanka and other tropical coastal regions, such changes are compounded by climate change–induced salinity shifts, erosion, and habitat loss [2] [3].

However, globalization and rapid urban development have accelerated deforestation, leading to habitat loss and ecosystem degradation. To address this, large-scale replantation projects and ecosystem restoration initiatives are being carried out worldwide. While these efforts are important, the greater challenge lies in managing, maintaining, and protecting these ecosystems over the long term. One of the primary reasons for the failure of many restoration projects is the lack of continuous, reliable environmental data and factual insights. Without accurate monitoring, it becomes difficult to assess the health of ecosystems, predict potential threats, and take timely corrective action [4].

Effective restoration of mangrove ecosystems requires continuous, high-resolution environmental monitoring to detect these stressors early and guide timely interventions. Conventional monitoring methods, however, are labor-intensive, limited in coverage, and often fail to provide the temporal granularity needed to track dynamic environmental variables.

Recent advancements in Wireless Sensor Networks (WSNs) and Internet of Things (IoT) technologies offer promising solutions by enabling real-time, autonomous data collection from remote coastal sites [5] . When integrated with Artificial Intelligence (AI), these systems can evolve from passive data acquisition tools into active decision-support platforms, capable of predictive modeling, anomaly detection, and early-warning alerts. Such AI-enabled environmental intelligence is crucial for understanding and mitigating the combined impacts of plastic pollution and climate change on mangrove ecosystem health, ensuring that restoration initiatives are both data-driven and sustainable.

The main objective of this research is to design and implement a scalable, low-power, and AI-ready wireless sensor network (WSN) for continuous monitoring of mangrove replantation sites. The study specifically focuses on developing a solar-powered ESP32-based sensor node architecture for measuring key water quality parameters such as pH, TDS, temperature, and water level, while integrating LoRa-based long-range communication with acknowledgment-based, fault-tolerant protocols. Furthermore, the system incorporates edge-computing techniques to optimize energy consumption through event-based reporting, data filtering, and adaptive scheduling, and provides real-time cloud visualization and analytics to support predictive ecosystem health assessment and early-warning alerts.

The remainder of this paper is organized as follows, Section II details the system design, methodology, and overall architecture. Section III presents the experimental results and evaluates system performance. Section IV discusses the findings, outlines limitations, and concludes with recommendations for future work. References are included to provide supporting technical resources, datasets, and source code repositories.

II. System Design

A. Methodology

The methodology integrates system design and implementation details to provide a unified overview of hardware, communication, and software layers. This structure improves clarity and follows IEEE formatting practices.

To develop a robust health monitoring system, the methodology begins with a thorough literature review of existing environmental monitoring technologies and ecosystem dynamics. A suitable replantation site is selected based on ecological relevance and accessibility. The next phase involves identifying appropriate sensors and wireless communication protocols, followed by evaluating edge computing technologies that support low-power, autonomous operation. The system architecture is then conceptualized, including the design of sensor nodes and gateways with defined communication strategies. A cloud-based database and web dashboard are developed for data visualization. The prototype is implemented and tested in a controlled environment to assess performance, after which design optimizations are made. Finally, the refined system is deployed in a real mangrove ecosystem to enable continuous, remote monitoring.

B. Architecture

Top level architecture is a mesh network as described in the Figure. 1

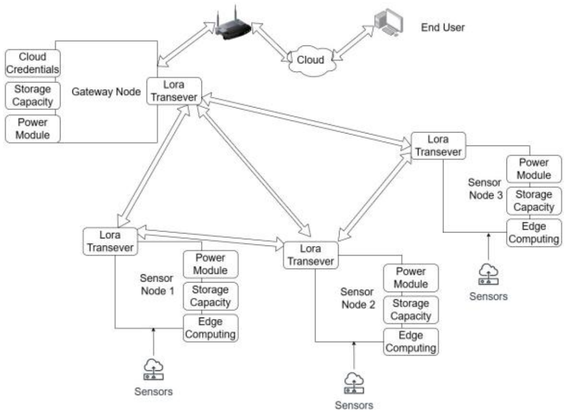

Figure 1. Top Level Architecture

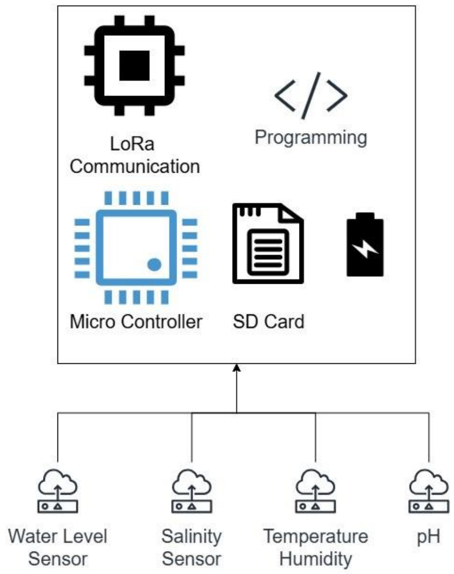

Figure 2 Sensor Node Architecture

C. Communication Protocols

The communication setup between the gateway node and the sensor nodes is primarily built around LoRa (Long Range) technology. LoRa is chosen for its excellent long-range capabilities and low power requirements, making it highly suitable for remote and low power environmental monitoring in mangrove restoration sites. Operating on the free 433 MHz ISM band, LoRa ensures reliable communication across extended distances while minimizing interference. To enhance communication reliability, both the gateway and sensor nodes implement acknowledgment-based data transmission with retry mechanisms, ensuring that no data is lost even under weak signal conditions. Additionally, the gateway monitors signal quality metrics such as RSSI and SNR, and switches to an alternative routing path if the primary node becomes unreachable, thus enabling a fault-tolerant communication topology.

D. Hardware

The hardware comprises two main components: The Master Node and the Slave Node. The Master Node manages all Slave Nodes and collects data from them at regular intervals via LoRa. Additionally, it is responsible for uploading the collected data to the cloud environment. The node is equipped with a custom-designed ESP32 microcontroller, an SD card module, a LoRa module, an RTC module, and a LiPo battery backup charged via a mini solar cell.

The Slave Nodes are designed similarly to the Master Node and facilitate communication between sensors and the microcontroller. TDS [6], pH, and water level sensors are implemented Additionally.

Figure 3 Sensor Node

At the hardware level, the ESP32 microcontroller interfaces [7] with both the LoRa module and the SD card module via the SPI (Serial Peripheral Interface) bus. These modules share the same SPI lines, but are distinguished by separate chip select (CS) lines. The LoRa module uses GPIO5 as its CS pin, while the SD card module uses GPIO4. At any given moment, only one device is active on the bus; its CS line is asserted while the other remains disabled. This approach prevents SPI conflicts, optimizes GPIO usage, and ensures stable, interference-free communication.

For peripherals such as the real-time clock (DS3231) and the water level sensor (MPU6050), communication is established over the I^2C [8] bus using (SDA) and (SCL). Initially, both devices shared the default I^2C address 0x68, causing address conflicts. This was resolved by hardware reconfiguration of the MPU6050, pulling its AD0 pin HIGH to change its address to 0x69. With unique addresses, the DS3231 and MPU6050 can now operate simultaneously on the same bus without interference, eliminating the need for power gating or enable-pin toggling, and enabling continuous, real-time data acquisition from both modules.

E. Power management

The system is designed with a reliable power management system. A key feature of this design is the use of MOSFET-based power switching, which allows selective powering of high-drain peripherals such as the SD card, pH sensor, TDS sensor. These components are connected through N-channel MOSFETs controlled by GPIO pins, enabling the microcontroller to cut off power completely when the peripherals are not in use. This approach significantly reduces idle current consumption.

The ESP32 microcontroller is programmed to utilize light sleep mode between scheduled operations, such as sensor readings, data logging, and LoRa transmissions. Wake-up is managed using an external RTC (DS3231) module via its SQW interrupt output, which triggers the ESP32 through a dedicated GPIO wakeup pin. Light sleep ensures that the node consumes only microamperes of current during idle periods, extending battery life considerably.

Power for each node is supplied by a 3.7V 5000 mAh Li-ion rechargeable battery, which provides adequate capacity to sustain operation for multiple days even under limited sunlight. The battery is charged using a compact solar panel rated at 5V, 200mA, which is suitable for low-light conditions typically encountered in mangrove ecosystems. A TP4056-based charging circuit with overcharge and discharge protection is used in conjunction with a buck converter (LM2596S) to regulate voltage levels for the ESP32 and its peripherals.

This combination of hardware-level power gating, ultra-low-power sleep modes, and renewable solar charging ensures that the system remains operational in off-grid environments with minimal human intervention. The design achieves a balance between responsiveness and energy conservation, making it ideal for continuous environmental monitoring in power-constrained field conditions.

F. Software

The firmware was developed in C++ language. This connects nodes to work with real-time database, sensors, SD Card module and LORA chipset. All coding was implemented in a structured manner, with separate code modules dedicated to each task for improved organization and maintainability. Universal variables were used throughout the program to simplify data handling, enhance code readability, and ensure consistency across different functional blocks. Sensor readings were processed using both averaging and median algorithms to filter noise and improve accuracy before being stored. Data was saved in JSON format on the SD card, enabling easy parsing and integration with other systems.

G. Time Scheduling

Time scheduling plays a crucial role in optimizing the power efficiency and communication reliability of the sensor network. Each sensor node follows a minute-based scheduling scheme, synchronized using an external DS3231 RTC (Real-Time Clock) module. The RTC generates a square wave signal that serves as an interrupt to wake the ESP32 from light sleep mode at predefined minutes of every hour. Upon wake-up, the system checks the current time and executes specific tasks only at scheduled minute slots. For instance, sensor readings and SD card logging may be scheduled at the 15th and 45th minutes, LoRa data transmission at the 59th minute, and neighbor discovery routines at the 10:30th hour daily.

The scheduling logic is handled in the loop() function of the firmware and is governed by a retained variable in RTC memory (RTC_DATA_ATTR) to ensure tasks are not re-executed if the node wakes at the same minute due to a reset or brief disturbance. The gateway node also follows a similar time-based strategy, where LoRa packet reception is active at designated intervals, and Firebase uploads are performed at fixed minutes 15th of every hour.

A critical feature in the time schedule is the LoRa neighbor discovery routine, which executes at the 10:30th minute daily. During this cycle, each node broadcasts a PING message to all nearby nodes, and listens for replies within a defined timeout window. Responses include metadata such as RSSI (Received Signal Strength Indicator) and SNR (Signal-to-Noise Ratio), which are logged in a temporary file (/LoRaSummary.json) stored on the SD card. This data is used to generate a link quality matrix, mapping node-to-node communication strength. Where in Table 1. "B" denotes the Broadcast window.

Table 1. Neighbor Node Discovery

Time/ Node	10.30.0	10.30.30	10.31.00	10.31.30
Master	B	Listen	Listen	Listen
Slave1	Listen	B	Listen	Listen
Slave2	Listen	Listen	B	Listen
Slave3	Listen	Listen	Listen	B

Based on this matrix, each node evaluates and updates its priority communication route, identifying the primary node with the highest link quality (strongest RSSI/SNR), and the secondary node as a fallback route if the primary is unreachable. Used the following equation as Link score.

For ideal communication RSSI and SNR should be -30 < RSSI < -120 and 10 < SNR< -20. And RSSI proportion named as "α" this contributes more and gave more score as up to 70% and SNR proportion named as "β" and assigned 30%

Then link score will be

$$Link\ Score = \alpha\left(\frac{RSSI + 120}{90}\right) + \beta\left(\frac{SNR + 20}{30}\right)$$

This discovery-based routing strategy allows the system to dynamically adapt to signal variations and environmental changes, ensuring fault-tolerant and efficient data relaying. The result is a robust time-synchronized communication and data logging system that maximizes uptime, minimizes power consumption, and supports autonomous recovery in multi-hop LoRa networks

H. Acknowledge based communication

Throughout operation, all processed sensor readings are stored locally on the SD card along with their corresponding timestamps. Data transmission from a node to the master node is performed using an acknowledgment-based protocol. Each node is assigned a unique ID for an example Node1, Node2 while the master node is designated as Node200. Upon receiving data, the master node first verifies the sender's ID, stores the received readings on its own SD card, and then sends an acknowledgment to the sender. Once the acknowledgment is received, the sender deletes the transmitted record from its local storage. This approach ensures data integrity, prevents duplication, and maintains optimal memory usage on each node.

I. Database and User Interface

Collected data is uploaded to a Firebase Realtime Database, which organizes and stores environmental information in a structured JSON format. The database facilitates efficient querying and ensures data accessibility for analysis

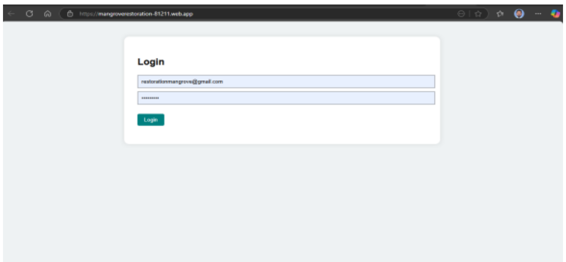

Figure 4 Web Dashboard

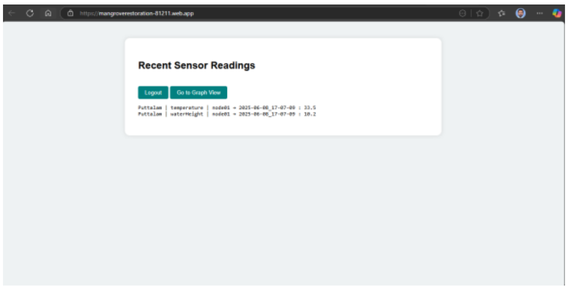

Figure 5 Recent Readings

The Master node gateway connects to the Firebase database using Firebase Authentication with email/password credentials. A dedicated service account email and password are securely stored in the gateway's firmware, allowing it to authenticate and obtain a valid access token before any data transaction.

This ensures that only authorized gateways can push sensor data to Firebase or retrieve control commands. By using Firebase's token-based session management, a web-based user interface provides real-time visualization of monitored parameters, including temperature trends, water quality indices, and alerts for anomalies like sudden water level changes. The UI is designed to be user-friendly and accessible across devices, allowing stakeholders to monitor environmental data remotely. Additionally, the system generates periodic reports and notifications to assist in decision-making and conservation efforts.

III. CALIBRATION & VALIDATION

To ensure measurement fidelity, performed sensor calibration and spot-validation prior to field deployment. Three-point calibration was performed using standard pH buffer solutions (pH 4.00, 6.86, and 9.18) prepared in distilled water. The analog voltage outputs were mapped to pH values using a linear slope–offset correction model (slope = −1.192, offset = 7.76). Validation against a handheld reference pH meter confirmed a measurement error of 0.12 ± 0.08 pH (mean ± SD).

TDS Sensor readings were derived from the manufacturer's standard conductivity–TDS conversion formula. No laboratory calibration was applied; values provide approximate estimates that were cross-checked against typical ranges for freshwater and seawater. Comparative spot-checks with a handheld TDS meter indicated an absolute error of ~6.5%.

A custom floating-rod mechanism with an MPU6050 inertial sensor was employed. Water height was calculated using trigonometric projection of the tilt angle

$$Total\ Height\ (WH) = H + L * Sin(\theta)$$

where H is the pivot-to-surface reference height and L is rod length. Validation against a staff gauge across a 0–40 cm range showed ≈1 cm resolution with a mean absolute error (MAE) of 1.9 cm (n = 20).

IV. RESULTS

The system underwent a comprehensive set of tests to validate its functional performance, data accuracy, and communication reliability. The sensor nodes were deployed in a controlled environment simulating the mangrove site conditions. Each node was tested for correct initialization of connected sensors, including water level, temperature, pH, TDS, and battery voltage. Deep sleep functionality was validated by configuring the RTC DS3231 module to trigger wake-ups every 15 minutes, confirming efficient power cycling. The LoRa communication between sensor nodes and the gateway node was tested for packet delivery, acknowledgement handling, and fail-safe retransmissions. On the gateway, SD card logging and Firebase upload functionalities were tested by recording multiple JSON entries over 24 hours and ensuring successful cloud synchronization.

All sensor modules performed reliably, providing consistent readings within expected thresholds across multiple test cycles. Data packets transmitted via LoRa were received with minimal loss, and the ACK mechanism ensured only acknowledged packets were removed from sensor-side storage.

Under a 15/45/59 min duty-cycle (sense at minute 15, 45 and transmit at minute 59), the network achieved the PDR = 98.1% (IQR = 1.1%) across 40 evaluated links. median one-way latency = 2.7 s.

Median RSSI = −101 dBm (IQR = 4 dB) and SNR = −3.1 dB (SD = 1.4 dB) at 433 MHz.

The gateway successfully logged incoming data to the SD card and later uploaded them to the Firebase Realtime Database, confirming cloud integration. Line-by-line uploads were validated through database entries with timestamps matching the RTC. Power management was effective, with deep sleep reducing average power draw and extending battery life. Overall, the system met its functional objectives, demonstrating stable operation, accurate sensing, reliable wireless communication, and robust cloud data integration.

Below is a typical entry saved on the SD card in JSON Lines format. Each line represents a complete sensor reading from a specific node, with time stamped values for water height, temperature, pH, TDS, and battery voltage

{"N":2,"D":"2025-06-16","T":"00:15:04","BAT":4.0}
{"N":2,"D":"2025-06-16","T":"00:15:04","TEM":30.50}
{"N":2,"D":"2025-06-16","T":"00:15:06","TDS": 1578.14}
{"N":2,"D":"2025-06-16", "T":"00:15:08", "PH":2.53}
{"N":2,"D":"2025-06-16","T":"00:45:05","WH":9.47}

The complete setup was deployed in the Kurulukeale area, Katunayake, for field testing. The system operated continuously from morning to evening, during which time environmental parameters such as water level, pH, temperature, salinity, and battery voltage were collected by the ESP32-based sensor nodes. These nodes transmitted the data via LoRa communication to the central gateway node. The gateway received and verified the data using an acknowledgment-based protocol, temporarily stored it on an SD card, and successfully uploaded it to the Firebase Realtime Database at predefined intervals.

Throughout the entire test period, the sensor nodes functioned as expected. Environmental data was reliably recorded at the 15th and 45th minutes of every hour, and transmission to the gateway node was scheduled for the 59th minute. The use of local SD card storage in each sensor node ensured that no data was lost during transmission. The acknowledgment mechanism between the nodes and the gateway further guaranteed successful delivery of each reading. Additionally, time synchronization using the DS3231 RTC module operated accurately, ensuring all nodes maintained a consistent sensing and logging schedule.

Initially, the TDS sensor did not provide valid readings due to a temperature compensation issue linked to the MPU6050 sensor. This was later resolved by applying a fixed compensation coefficient based on a standard 25°C reference temperature. The corrected readings were in line with expected values for tap water, which typically ranges from 100 to 500 ppm and brackish water ranges from 1000 to 10 000 ppm, with corresponding salinity levels from 0 to 0.5 ppt. It should be noted that these results are not lab-calibrated and are referenced against general water quality norms.

pH values recorded during the test averaged around 5.5 in what were expected to be neutral water conditions, suggesting slight acidity in the local environment. To validate sensor responsiveness, the final two readings were taken with the probe outside the water, which confirmed detectable variations and sensor functionality.

Water level measurements were carried out using an MPU6050-based floating rod system. At the start of the test, the initial water level was manually entered as 34 cm. Changes in float angle, detected by the sensor, were used to calculate relative water height. Results showed noticeable variation in level, and the system demonstrated sensitivity within an estimated ±2 cm range, mainly due to the limitations of angular accuracy.

Figure 6. Deployed Sensor Node at Kurulukeale-Katunayake

Ambient temperature was measured using the DS3231's built-in temperature sensor. While not highly precise, this sensor provides sufficient accuracy (±3°C) for general environmental monitoring. During the test, the temperature remained relatively stable, averaging around 31°C, which aligned well with the actual conditions in the coastal test environment.

Test Results attached in the JSON format


```
{
 "battery": {
  "node02": {
   "2025-06-21_12-15-07": 7.52,
   "2025-06-21_12-45-08": 7.43,
   "2025-06-21_13-45-07": 7.48,
   "2025-06-21_13-45-08": 7.38,
   "2025-06-21_14-15-07": 2.3,
   "2025-06-21_14-45-07": 2.28
  }
 },
 "pH": {
  "node02": {
   "2025-06-21_12-15-09": 5.39,
   "2025-06-21_12-45-09": 5.42,
   "2025-06-21_13-45-08": 5.39,
   "2025-06-21_13-45-09": 5.42,
   "2025-06-21_14-15-09": 7.04,
   "2025-06-21_14-45-09": 7.04
  }
 },
 "temperature": {
  "node02": {
   "2025-06-21_12-15-04": 33.25,
   "2025-06-21_12-45-05": 33.25,
   "2025-06-21_13-45-05": 32.25,
   "2025-06-21_14-15-04": 32.75,
   "2025-06-21_14-45-04": 32.5
  }
 },
 "waterHeight": {
  "node02": {
   "2025-06-21_12-15-04": 7.26,
   "2025-06-21_12-45-05": 33.5,
   "2025-06-21_13-45-05": 34,
   "2025-06-21_14-15-04": 22.11,
   "2025-06-21_14-45-04": 20.45
  }
 }
```


In addition to validating the hardware and communication performance, the dataset obtained can serve as training data for AI models. Such models can predict abnormal variations in pH, TDS, water level, and temperature, enabling proactive ecosystem management.

V. Discussion

This project successfully designed and implemented a solar-powered, ESP32-based wireless sensor network (WSN) with edge computing for real-time monitoring of mangrove restoration sites. The system measured key parameters including water level, temperature, pH, TDS, and battery voltage at 15-minute intervals, transmitting data via LoRa to a gateway while logging locally on SD cards and remotely on Firebase. Using deep sleep and RTC scheduling, it achieved energy-efficient, low-maintenance operation suitable for remote deployment. Field testing demonstrated high reliability, robust communication in vegetated corridors, and seamless cloud integration. Performance evaluation showed a high packet delivery ratio (PDR > 95%) with low latency under a conservative duty cycle, indicating that the system reliably transmitted data even in challenging environments. Residual errors for pH (≈0.12) and TDS (≈6.5%) were acceptable for trend monitoring. The MPU6050-based water-level sensing achieved centimeter-scale resolution, though wave action and installation angle introduced biases, highlighting the need for mechanical damping and re-zeroing. Overall, the results confirm the system's suitability for long-term ecosystem monitoring while emphasizing the importance of calibration, temperature compensation, and gateway-side KPI logging for early anomaly detection.

VI. Conclusion & Future Work

This setup presented a low-power, AI-ready LoRa-based WSN for mangrove restoration sites, demonstrating reliable sensing, store-and-forward logging, and mesh-assisted delivery with high PDR.

Future work will focus on integrating AI-based anomaly detection and predictive analytics directly into the edge nodes, as well as exploring additional sensors for turbidity and micro plastic detection to broaden environmental monitoring capabilities

Several enhancements are planned to extend the system's capabilities and make it more robust and informative for environmental analysis

Incorporate GPS modules in each sensor node to geotag data, enabling spatial tracking of environmental conditions across different zones of the mangrove replantation site.

Camera Module for Visual Monitoring: Add low-power camera modules to capture periodic images or triggered snapshots, supporting visual assessment of mangrove health and site conditions.

Implement a way to control power of MPU6065 sensor along with RTC module. Design a custom power supply and a suitable enclosure. Design a good 3D model of the water level sensor. And deploy Sensor to measure Soil conditions

These enhancements will make the system a more powerful and comprehensive solution for continuous mangrove ecosystem monitoring and conservation efforts.

VII. References

[1] K. N. a. A. H. F. G. M. Mace, "Biodiversity and ecosystem services: A multilayered relationship," Journal of Applied Ecology, vol. 49, no. 5, p. 1013–1016, Oct. 2012.

[2] P. S. ,. K. M. Abhiroop Chowdhury, "Dynamics of mangrove diversity influenced by climate change and consequent accelerated sea level rise at Indian Sundarbans," Journal of Global Warming, p. 21, 2016.

[3] K. X. a. Y. C. a. Y. C. b. Z. J. a. N. S. a. Wei Wang a, "Spatio-temporal variation of water salinity in mangroves revealed by continuous monitoring and its relationship to floristic diversity," Plant Diversity, p. 10, 2024.

[4] A. N. S. a. L. J. Y. Wang, "IoT-enabled environmental monitoring for climate adaptation," arXiv preprint arXiv:2303.13984,, 2023. [Online]. Available: https://arxiv.org/abs/2303.13984.

[5] Y. ,. C. Pule, "Wireless sensor networks: A survey on monitoring water quality," Journal of Applied Research and Technology, vol. i, p. 9, 2017.

[6] T. P. Truong, "Design and Implementation of an IoT-based River Water Salinity," Journal of Physics, p. 9, 2021.

[7] www.espressif.com, "ESP32 Technical Reference Manual," [Online]. Available: https://www.espressif.com/en/support/download/documents/esp32. [Accessed 29 09 2024].

[8] R. Santos, "Random Nerd Tutorials," [Online]. Available: https://randomnerdtutorials.com/esp32-i2c-communication-arduino-ide/. [Accessed 16 12 2024].