DEERWALK INSTITUTE OF TECHNOLOGY

Tribhuvan University

Faculty of Humanities and Social Sciences

WATER LEVEL TRACKING DASHBOARD

A PROJECT REPORT

Submitted To:

Department of Computer Application

DWIT College

In partial fulfillment of the requirements for Bachelor's in Computer Application

Submitted By:

Bipashree Aryal

[TU Roll Number]

DWIT COLLEGE

DEERWALK INSTITUTE OF TECHNOLOGY

Tribhuvan University

Faculty of Humanities and Social Sciences

DWIT College

SUPERVISORS RECOMMENDATION

I hereby recommend that this project prepared under my supervision by BIPASHREE ARYAL entitled **"WATER LEVEL TRACKING DASHBOARD"** in partial fulfillment of the criteria for the degree of Bachelor's in Computer Applications to be taken further for evaluation.

_____

Shyam Sundar Khatiwada

[Designation]

DWIT College, Sifal, Kathmandu

Tribhuvan University

Faculty of Humanities and Social Sciences

DWIT College

LETTER OF APPROVAL

This is to certify that this project prepared by BIPASHREE ARYAL entitled **"WATER LEVEL TRACKING DASHBOARD"** in partial fulfillment of the requirements of the degree of Bachelor's in Computer Application has been well studied. It is gratifying in scope and quality as a project for the required degree.

Shyam Sundar Khatiwada [Academic Designation] DWIT College	[Examiner] [Academic Designation] FoHSS, Tribhuvan University
[Project Coordinator] [Academic Designation] DWIT College	[Campus Chief] [Academic Designation] FoHSS, Tribhuvan University

ABSTRACT

The Water Level Tracking Dashboard is an Internet of Things (IoT) project that combines the ESP32 Microcontroller and HCSR04 Sonar Module to address water level management challenges in household and commercial tanks. This cost-effective solution provides real-time insights into water consumption and allows for remote control of the water pump. By utilizing an iterative prototyping model and evaluating multiple microcontrollers, including Raspberry Pi, Arduino Uno, and ESP32, it was determined that the ESP32 offers the best value for the money.

The Water Level Tracking Dashboard fills a market void by providing an affordable and accessible smart solution for homeowners and tenants in Nepal. With its ability to accurately track water levels and offer valuable consumption data, this system helps alleviate the concerns associated with water management. By enabling remote control of the water pump, users can effectively regulate water usage based on the tank's level. This project has implications for a wide range of applications, benefiting both residential and commercial sectors that rely on water tanks in Nepal.

Overall, the Water Level Tracking Dashboard contributes to water conservation efforts and promotes efficient water usage in Nepal. It provides an effective tool for monitoring water levels, gaining insights into consumption patterns, and remotely controlling the water pump, ultimately simplifying the water management process for users.

Keywords: Internet of Things (IoT); ESP32 Microcontroller; HCSR04 Sonar Module; Remote On/Off Functionality; Iterative Prototyping; Water Tanks.

ACKNOWLEDGEMENT

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

IoT: Internet of Things

WLTD: Water-Level Tracking Dashboard

ESP32: Espressif Systems-32

HCSR04: High-Conductance Ultrasonic Sensor

NPR: Nepalese Rupee

IDE: Integrated Development Environment

Wi-Fi: Wireless Fidelity

IO: Input/Output

RTDB: Real-Time Database

API: Application Programming Interface

CHAPTER 1: INTRODUCTION

1.1. Overview

The Water Level Tracking Dashboard project was initiated due to personal experiences and concerns regarding the hassle of manually controlling water pumps and the decreasing groundwater levels during winter seasons. The project aims to address the problem of inefficient water level management in household and commercial water tanks.

The primary objective of the project is to develop a smart solution that simplifies water level management and optimizes the utilization of precious groundwater resources. By incorporating remote control functionality for the water pump and providing real-time insights into water consumption patterns, users can make informed decisions regarding their water usage.

The scope of the project focuses on creating a user-friendly dashboard that accurately tracks water levels and offers real-time consumption data. While the project aims to enhance water management practices, it is important to note that it does not inherently decrease water consumption. Rather, it empowers users with valuable information to better manage their water usage and promote conscious water management practices.

1.2. Problem Statement

The current problem lies in the inefficient management of water levels in household and commercial water tanks, compounded by the hassle of manually controlling water pumps. This issue arises due to the lack of convenient and automated solutions that provide real-time monitoring and control of water levels. The existing systems require constant manual intervention and fail to offer insights into water consumption patterns, making it challenging for users to effectively manage their water resources. The evidence of this problem can be seen in the frustrating and time-consuming process faced by individuals in maintaining optimal water levels in tanks and the resulting wastage or shortage of water.

The Water Level Tracking Dashboard project proposes a solution to address this problem by developing a smart system that automates the control of water pumps and offers real-time monitoring of water levels. By incorporating advanced sensors and microcontroller technology, the project enables users to remotely monitor and regulate the water levels in their tanks. The system will provide accurate and timely information on water levels, empowering users to make informed decisions and take proactive measures to manage water efficiently.

1.3. Objectives

i. Develop a Water Level Tracking Dashboard that automates water pump control based on real-time water level data, simplifying water level management in household and commercial water tanks.

    a. Integrate an ESP32 Microcontroller and HCSR04 Sonar Module for accurate water level monitoring.

    b. Implement remote control functionality for seamless on/off operation of the water pump.

ii. Provide insights into water consumption patterns through the Water Level Tracking Dashboard, enabling users to make informed decisions regarding their water usage.

    a. Display real-time water consumption data on the dashboard.

iii. Enhance water management practices and promote efficient utilization of groundwater resources.

    a. Facilitate optimal utilization of water resources by enabling users to maintain appropriate water levels in their tanks.

    b. Foster awareness of water conservation by providing information on water consumption and encouraging responsible water usage.

iv. Offer a cost-effective and user-friendly solution for homeowners and tenants in Nepal, addressing the market void for affordable smart water level management systems.

1.4. Scope and Limitation

1.4.1. Scope

i. The project focuses on the development of a Water Level Tracking Dashboard using the ESP32 Microcontroller and HCSR04 Sonar Module.

ii. The dashboard will provide real-time monitoring of water levels in household and commercial water tanks.

iii. The system will incorporate remote control functionality for the water pump, enabling users to turn it on or off based on the water level.

iv. The dashboard will display insights into water consumption patterns, allowing users to make informed decisions regarding their water usage.

1.4.2. Limitations

   i.   The project's scope is limited to monitoring and controlling water levels in tanks and providing insights on water consumption. It does not include mechanisms to directly reduce water consumption.

   ii.   The accuracy of the water level measurements may be affected by external factors such as temperature changes, tank material and sensor calibration.

   iii.   The project assumes a stable internet connection for remote access and control of the Water Level Tracking Dashboard.

1.5.   Report Organization

This Project Report is Organized as follows:

   i.   Preliminary Section:

        This Section consists of the Title Page, Abstract, Acknowledgement, Table of Contents, List of Figures, List of Tables, and the List of Abbreviations.

   ii.   Introduction Section:

        This Section contains the Overview, Problem Statement, Objectives, Scope, and the Limitation of the project.

   iii.   Background Study and Literature Review Section:

        In this section, the Background Study and the Literature Review are performed before the selection of the project is discussed.

   iv.   System Analysis:

        This section consists of the Requirement Analysis (Functional and Non-Functional), Feasibility Analysis (Technical, Operational, Economic), Data Modeling, and System Design.

   v.   Implementation and Testing:

        The Implementation of the project and the testing conducted on the project along with the Test Cases, Input Data and Output Expected are discussed in this section.

   vi.   Conclusion and Future Recommendation:

        The Outcome of the Project, Conclusion, and the Future Recommendations to further improve the scope of the project are discussed in this section.

CHAPTER 2: BACKGROUND STUDY AND LITERATURE REVIEW

2.1.   Background Study

The HCSR04 Sonar Module [1] is an ultrasonic sensor that calculates distance by measuring the time it takes for ultrasonic waves to travel and return. By emitting ultrasonic waves and detecting their reflections, the module can determine the distance to an object. This functionality is essential for accurately measuring the water level in a tank, as the module can be positioned at the top of the tank to measure the distance to the water surface.

The ESP32 microcontroller [2], known for its Wi-Fi capabilities, can be used to post data to a Firebase Realtime Database [3]. Firebase provides cloud-based services, including a Realtime Database that allows for the storage and retrieval of data in real-time. By establishing a Wi-Fi connection and utilizing the Firebase API [4], the ESP32 can send HTTP POST requests containing relevant data. This data is stored in the Firebase Realtime Database, enabling users to access and monitor the water level information in real-time through the Water Tracking Level Dashboard.

The combination of the HCSR04 Sonar Module's distance calculation capabilities and the ESP32's data posting functionality to Firebase Realtime Database forms the foundation of the Water Level Tracking Dashboard project. These technologies enable accurate water level measurements and seamless data communication, facilitating efficient monitoring and control of water levels in tanks.

2.2.   Literature Review

After conducting both internet and market research, it was discovered that there are no "Smart" solutions accessible in the market. Existing methods employ a Static On/Off Flow Control Valve, which turns on or off the Inlet Pipe dependent on the level of the water, but this does not manage the condition of the electric motor, resulting in long-term damage to the costly component.

Additional investigation in prominent online retailers yielded no results. Ivent Solutions Limited, situated in New Zealand, sells a comparable product named Smart-Water Online[TM]. Which provides identical functionality to the Water Level Tracking Dashboard but costs roughly $435 for the starting pack alone, which is a staggering 29-Times the price of the hardware that the Water Level Tracking Dashboard utilizes [5].

CHAPTER 3: SYSTEM ANALYSIS

3.1.1. Requirement Analysis

i. Hardware Requirements

- ESP32 Microcontroller
- HCSR04 Sonar Module
- Relay Module
- Water Pump

ii. Functional Requirements

- **Water Level Measurement**: The System must accurately measure the water level in the tank using the HCSR04 Sonar Module. It should provide real-time data on the water level, enabling users to monitor and track changes.

- **Data Visualization**: The dashboard should present the water level information in a clear and intuitive manner. It should include visual representation such as graphs or charts that allow users to easily interpret and analyze the data.

- **Remote Control Functionality**: The system should provide the capability to remotely control the water pump and based on the water level. Users should be able to turn the pump on or off through the dashboard interface.

iii. Non-Functional Requirements

- **User-Friendly Interface**: The dashboard should have a user-friendly interface that is easy to navigate and understand. It should be intuitive and require minimal technical expertise to operate.

- **Real-Time Data Updates**: The system should provide real-time updates of water level measurements and pump status. Users should be able to access the latest information without significant delays.

- **Reliability and Accuracy**: The system should be reliable, providing accurate measurements and ensuring the proper functioning of remote-control functionality. It should minimize errors and maintain consistency in data reporting.

- **Security and Privacy**: The system should implement appropriate security measures to protect user data and ensure privacy. Access to the dashboard and control features should be restricted to authorized individuals.

3.1.2. Feasibility Analysis

i. Technical Feasibility

- **Hardware Availability**: The required hardware components, including the ESP32 Microcontroller and HCSR04 Sonar Module, are readily available on the market for a combined amount of less than Rs. 1,500 NPR. These components are widely used and have established compatibility with each other, ensuring the feasibility of integrating them into the system.

- **Software and Programming**: The ESP32 Microcontroller can be programmed using various development environments, such as Arduino IDE or PlatformIO. These environments provide extensive libraries and resources for interfacing with sensors and implementing Wi-Fi connectivity for no additional cost whatsoever. The availability of these software tools and the documentation surrounding them make the development process feasible.

- **Communication and Data Management**: The integration of the ESP32 with the Firebase RTDB presents technical feasibility. The Firebase API provides the necessary functions for data posting and retrieval, allowing for real-time communication between the microcontroller and database. The availability of robust documentation and community support for Firebase facilitates the implementation of data management and synchronization features.

- **Sensor Accuracy and Performance**: The HCSR04 Sonar Module is a widely used and reliable sensor for distance measurement. It offers satisfactory accuracy and performance, making it feasible for accurately measuring the water level in a tank. Furthermore, for improved accuracy, the system can be equipped with a DHT11/DHT 22 Sensor to measure the temperature and humidity of the tank and the distance can be calculated relative to the values obtained. However, this falls out of scope in this condition.

- **Wi-Fi Connectivity**: The ESP32 Microcontroller's built-in Wi-Fi capabilities allow for seamless connectivity to the internet, facilitating real-time data transmission and remote-control functionality. The widespread availability of Wi-Fi networks ensures technical feasibility in establishing the required connections.

ii. Operational Feasibility

- **Install and Forget**: The WLTD is a One-time Install and Use Forever solution. This means that once the system is installed and configured, it can be utilized continuously without the need for recurring installations or significant maintenance efforts. This nature of the system offers operational benefits by reducing ongoing operational costs and minimizing disruptions to daily operations.

- **Ease of Use**: The dashboard should have a user-friendly interface that is intuitive and requires minimal technical expertise. Users should be able to navigate through the dashboard, access water level data, and control the water pump with ease.

- **Integration with Existing Systems**: The WLTD should seamlessly integrate with existing water management processes. It should be compatible with standard water tanks used in household/commercial settings. The system should not require significant modifications to the existing infrastructure, minimizing disruption to operations during installation and implementation.

iii. Economic Feasibility

- **Cost of Development**: Listed below is the total cost related to the hardware components used in the development of the Water Level Tracking Dashboard:

S/N	Particulars	Quantity	Rate	Price
1	ESP32 Microcontroller	1	Rs. 900	Rs. 900
2	HCSR04 Sonar Module	1	Rs. 350	Rs. 300
3	2-Way Relay Module	1	Rs. 250	Rs. 250
4	Printed Circuit Board (PCB)	1	Rs. 50	Rs. 50
Total				Rs. 1,500

Table 1: Cost of Development for WLTD

- **Cost Savings**: WLTD offers potential cost savings by enabling efficient water management. By accurately monitoring water levels and providing insights to consumption patterns, users can optimize their water usage and minimize wastage. Additionally, this also provides a fail-proof measure to mitigate overflows in water tanks resulting in reduced wastage of water and consequently reduced water bills eventually.

- **Energy Efficiency**: The remote-control functionality of WLTD allows users to efficiently manage the water pump's operation resulting in energy cost savings as well.
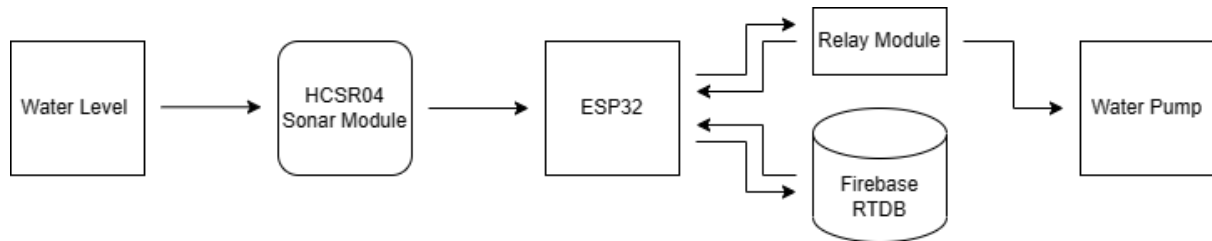
3.1.3. Data Modeling

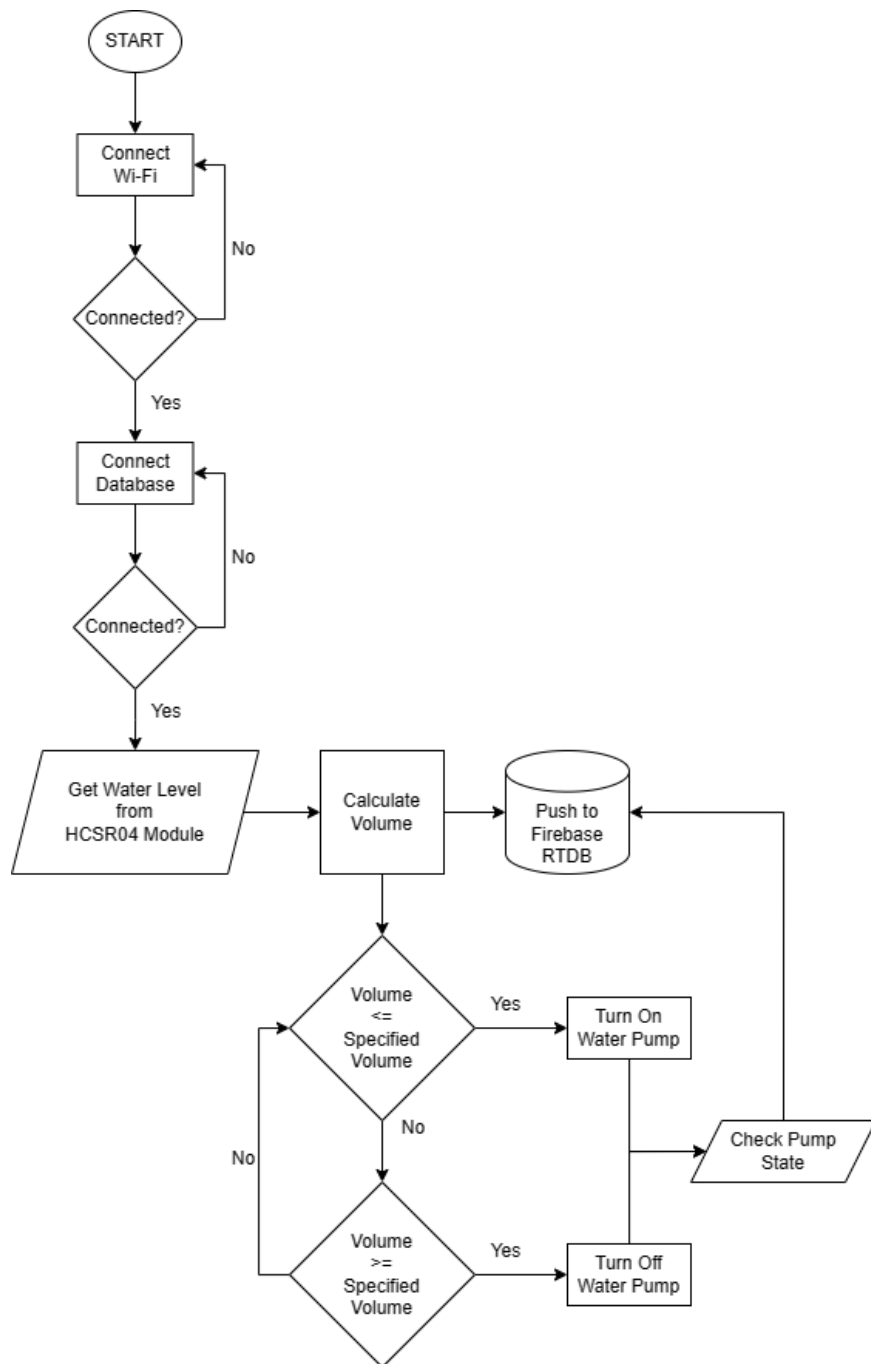**Figure 1: Data-Flow Diagram for WLTD**

3.1.4. Process Modeling

Figure 2: Flowchart for WLTD

3.2. SYSTEM DESIGN

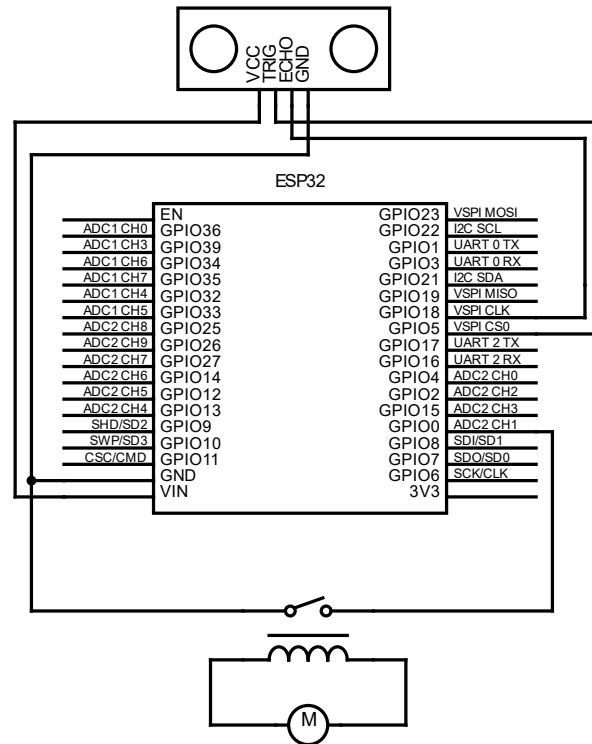### 3.2.1. Circuit Design

Figure 3: Circuit Diagram for WLTD

CHAPTER 4: IMPLEMENTATION AND TESTING

4.1.   IMPLEMENTATION

After evaluating diverse options, it was determined that the Raspberry Pi [6], while capable, is expensive and computationally overkill for the project's scale.

The Arduino Uno [7], on the other hand, is a reliable microcontroller but lacks built-in Wi-Fi or Bluetooth capabilities. To make it suitable for IoT-based projects, the addition of a wireless module is required. However, this not only increases the cost but also introduces technical complexities.

Considering these factors, the ESP32 [2] microcontroller emerged as the ideal choice for this project. It offers a great package for the price, with built-in wireless features that eliminate the need for additional modules. The ESP32 provides the necessary capabilities for seamless data communication and connectivity [8].

However, flashing the code to the ESP32 can be a tedious process. It has two modes: Download mode, used for flashing data to the chip, and Hardware mode, used for utilizing the ESP32 as a HID (Human Interface Device) Class device. One notable advantage of the ESP32 over the Arduino Uno is its ability to function as a HID device without requiring additional hardware.

During the code flashing process, it is necessary to hold the boot button on the ESP32 module until the computer establishes a connection with the chip. However, this inconvenience can be addressed by placing a 10μF capacitor between the Enable (EN) and Ground pins of the ESP32 module.

4.1.1. Tools Used

   i.   Hardware Components: ESP32 Microcontroller, HCSR04 Sonar Module, 2-Channel Relay Module, Breadboard, Jumper Wires, Micro-USB Cable.
  ii.   Time Protocol: Network Time Protocol (NTP).
 iii.   Backend Code: Arduino Programming Language (A Variation of C++).
  iv.   Frontend Code: Flutter (A Framework of Dart Programming Language).
   v.   Database Platform: Google Firebase Real-Time Database.
  vi.   IDEs/Code Editors: Arduino IDE, Visual Studio Code.
 vii.   Serial Monitoring Software: Open Serial Port Monitor.
viii.   Diagrams and Charts: Draw.io, TeamGantt, Circuit-Diagram.org.
  ix.   Documentation: Microsoft Word, Microsoft PowerPoint.

4.1.2. Implementation Details of Modules

The Implementation of Modules in WLTD can be differentiated into multiple layers as discussed below:

a. Hardware Implementation:

- Sensor Integration: Connection between the HCSR04 Sonar Module and the ESP32 Microcontroller.

- Pump Control Interface: Interfacing between the ESP32 with the Relay Module to enable remote control functionality.

b. Communication Implementation:

- Wi-Fi Module: Utilizing the built-in Wi-Fi capabilities of the ESP32 along with the Wi-Fi Library for connection to the internet and subsequently the Firebase Database.

- Firebase Integration: Interfacing the ESP32 with the Firebase Realtime Database (RTDB) to store and retrieve water level data and pump control status using the Firebase-ESP32 Library.

c. User Interface Implementation:

- Mobile Application: Providing a platform for the users using Flutter to provide an interactive platform for monitoring the water levels and the pump control options.

- Real-Time Updates: Using Flutter's Real-Time Listeners to receive live updates whenever the water level changes.

4.2. TESTING

4.2.1. Test Cases for Unit Testing

S/N	Test Case	Test Input	Expected Outcome
1.	Sensor Reading	Simulated Water Level Data	Correct water reading is captured
2.	Pump Control	User turns on the pump	Pump state changes to ON
		User turns off the pump	Pump state changes to OFF
3.	Data Synchronization	New water level recorded	Real-Time update of water level on UI
4.	Database Connectivity	Valid Database Credentials	Successful connection to the Firebase RTDB
		Invalid Database Credentials	Connection error message displayed

Table 2: Test Cases for Unit Testing WLTD

4.2.2. Test Cases for System Testing

S/N	Test Case	Test Input	Expected Outcome
1.	Sensor Reading	Place the sensor in a water tank with known water levels	Water level data is accurately captured and displayed.
		Verify the displayed water level readings with the actual water levels.	Water level readings match the actual levels
2.	Pump Control	Send a command to turn the pump ON	Pump state changes to ON
		Send a command to turn the pump OFF	Pump state changes to OFF
3.	Data Synchronization	Simulate changes in water level data	Real-Time update of water level on UI

Table 3: Test Cases for System Testing WLTD

CHAPTER 5: CONCLUSION AND FUTURE RECOMMENDATION

5.1. LESSON LEARNT/OUTCOME

During the Water Level Tracking Dashboard project, several valuable lessons were learned that contributed to the overall development and implementation process. These lessons provided insights and guidance for future projects of a similar nature.

Firstly, the importance of selecting the right microcontroller for the project was highlighted. Through the evaluation of different options such as Raspberry Pi, Arduino Uno, and ESP32, it was realized that the ESP32 microcontroller proved to be the most suitable choice. It offered a good balance between cost, functionality, and ease of use, with its built-in wireless capabilities and affordability. This experience emphasized the significance of considering specific project requirements and constraints when selecting hardware components.

Additionally, the process of flashing code to the ESP32 highlighted the need for thorough testing and troubleshooting. Understanding the different modes of the ESP32, such as the download mode and hardware mode, was crucial in ensuring successful code deployment. The discovery of using a 10μF capacitor to address the boot button holding issue demonstrated the importance of resourcefulness and creative problem-solving when encountering technical challenges.

Lastly, the project highlighted the significance of thorough testing at different stages. From unit testing the individual modules to system testing the overall functionality, conducting comprehensive tests helped identify and rectify issues early on. The importance of a robust testing strategy became evident in ensuring the reliability, accuracy, and performance of the Water Level Tracking Dashboard.

In conclusion, the lessons learned throughout the project encompassed hardware selection, troubleshooting, and rigorous testing. These insights will undoubtedly guide future endeavors, emphasizing the significance of thoughtful decision-making, adaptability, and continuous improvement in achieving successful project outcomes.

5.2. CONCLUSION

In conclusion, the Water Level Tracking Dashboard project has successfully tackled the challenge of water level management in household and commercial settings. By leveraging the ESP32 microcontroller and the HCSR04 Sonar Module, this project has offered a valuable solution for monitoring water levels and controlling water pumps.

Through iterative prototyping and comprehensive testing, the project demonstrated the feasibility and effectiveness of using the ESP32 microcontroller as a cost-efficient and feature-rich platform. Its built-in wireless capabilities and affordability make it an ideal choice for implementing IoT-based water level tracking systems.

The successful implementation of the Water Level Tracking Dashboard project highlights the potential of IoT technologies in addressing real-world challenges. It offers a practical and affordable solution for water level monitoring and management, benefiting homeowners and industries in Nepal and beyond. The project's findings and insights can serve as a foundation for future endeavors in smart water management, fostering a more sustainable approach to water resource utilization.

REFERENCES

[1] ElecFreaks, "HC-SR04 Ultrasonic Module User Guide," [Online]. Available: https://www.elecfreaks.com/blog/post/hc-sr04-ultrasonic-module-user-guide.html.

[2] E. Systems, "Arduino-ESP32 Libraries," 2020. [Online]. Available: https://github.com/espressif/arduino-esp32/blob/master/libraries/WiFi/src/WiFi.h.

[3] Google Inc., "Firebase Documentation," Google Inc., [Online]. Available: https://firebase.google.com/docs. [Accessed 03 04 2023].

[4] Mobizt, "Firebase-ESP32," 19 04 2019. [Online]. Available: https://github.com/mobizt/Firebase-ESP32.

[5] "Smart Water Online," 16 02 2021. [Online]. Available: https://smartwateronline.com.

[6] The Pi Foundation, "Raspberry Pi Documentation," [Online]. Available: https://www.raspberrypi.com/documentation/. [Accessed 23 03 2023].

[7] Arduino Inc., "Arduino Docs," [Online]. Available: https://docs.arduino.cc/. [Accessed 23 03 2023].

[8] J.-L. Aufranc, "CNX-Software," [Online]. Available: https://www.cnx-software.com/2020/03/24/know-the-differences-between-raspberry-pi-arduino-and-esp8266-esp32/. [Accessed 21 03 2023].

[9] A. N. R. L. Steven Sachio, "CORE," 14 12 2018. [Online]. Available: https://www.core.ac.uk.

[10] F. Dev, "Flutter Documentation," [Online]. Available: https://docs.flutter.dev/.

[11] M. M. A. K. V. M. Shweta Karad, "IoT Based Real Time Water Monitoring System for Smart City," *International Journal of Innovative Science and Research Technology,* p. 251, 2018.

[12] S. C. Priya Jayakrishnan, "Water Level Monitoring System Using IoT," *International Research Journal of Engineering and Technology (IRJET),* p. 1817, 2017.

[13] Espressif Systems, "ESP32 Technical References," [Online]. Available: https://www.espressif.com/sites/default/files/documentation/esp32_technical_reference_manual_en.pdf. [Accessed 22 03 2023].