SMART WATER TANK PROJECT REPORT

Ikram El-hajri

20/12/2023

Table of Contents

Table of Figures:

Introduction

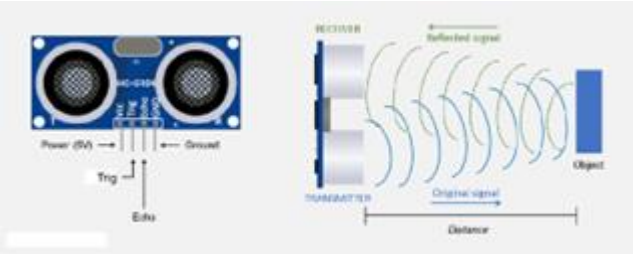
Water, the essence of life, stands at the forefront of global challenges as humanity grapples with environmental issues and the need for sustainable resource management. In the face of growing concerns about water scarcity and the unpredictable dynamics of climate change, innovative solutions have become imperative. This report introduces a cutting-edge approach to water resource monitoring through an Internet of Things (IoT)-based system. With an acute awareness of the urgency surrounding water conservation, our proposed system utilizes IoT technology to offer real-time water level measurement. The user interface, featuring a dynamic dashboard, not only provides immediate insights but also empowers users to actively engage in the conservation effort by customizing minimum water level thresholds. In this way, our system represents a significant stride toward harnessing technology for the greater good of water management and sustainability.

I. COMPONENTS

1. Sensors:

Ultrasonic Sensor:

This critical component serves as the eyes of the system, employing ultrasonic waves to precisely measure the water level within the tank. Its non-intrusive nature makes it ideal for various tank configurations, ensuring accurate readings for effective water management.

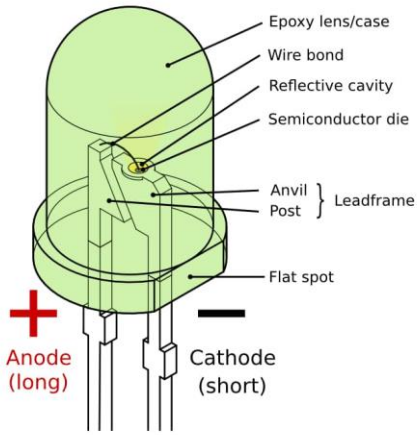


1 Ultrasonic Sensor

2. Actuators:

LEDs (Light Emitting Diodes):

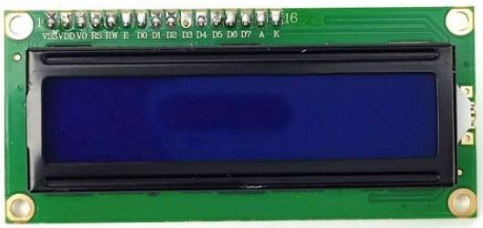
Functioning as visual indicators, LEDs play a pivotal role in communicating the water level status to users. Through a simple and intuitive interface, these lights convey real-time information, enhancing user awareness and engagement.



2 LEDs (Light Emitting Diodes)

LCD (Liquid Crystal Display):

This component further contributes to user interaction by displaying detailed information about the water level. The LCD provides a comprehensive view, ensuring users have access to nuanced data for informed decision-making.

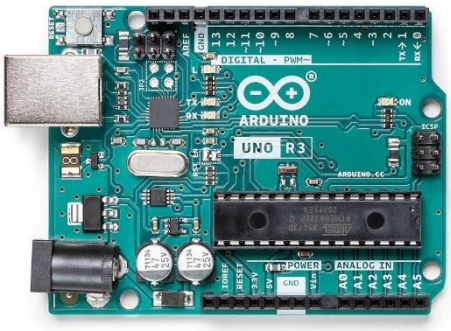


3 LCD (Liquid Crystal Display):

II. III. BASIC CONCEPTS

1. Arduino Uno:

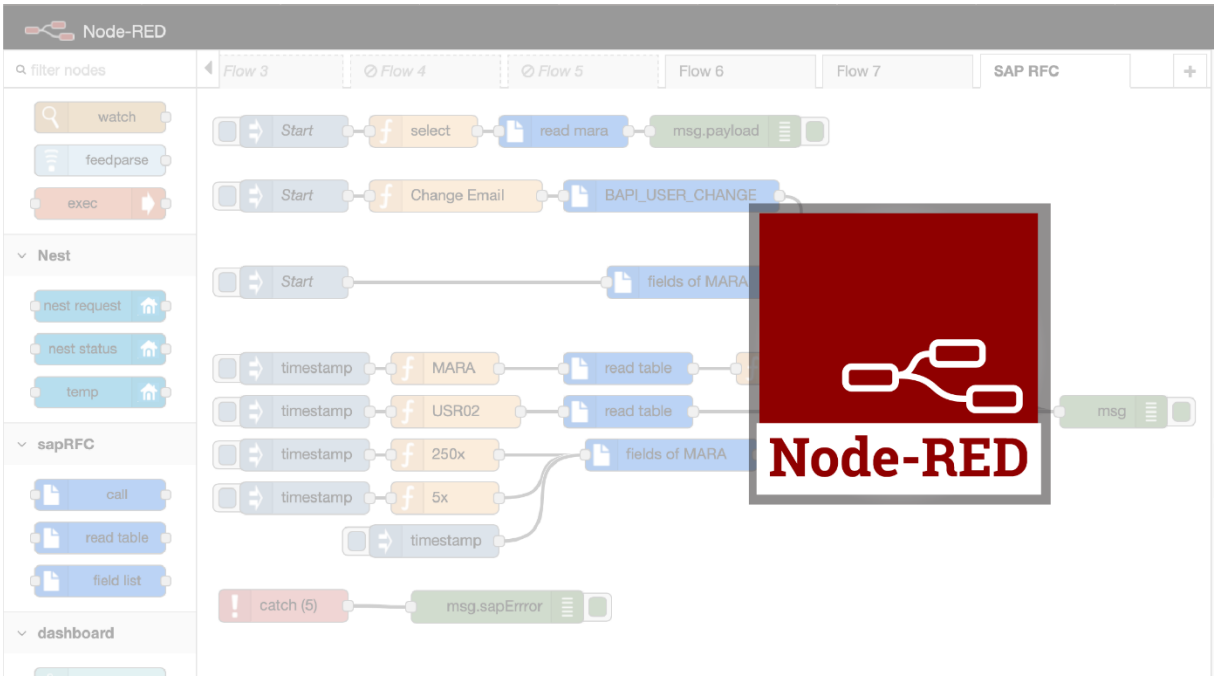
The Arduino Uno serves as the central microcontroller, orchestrating the communication between sensors, actuators, and the user interface. Its versatility and ease of programming make it an ideal choice for managing the data flow within the system.



4 Arduino Uno

2. Node-RED:

Node-RED, a flow-based development tool, facilitates the creation of a seamless connection between hardware devices, APIs, and online services. Its visual programming approach simplifies the integration of different components, enhancing the overall efficiency of the system.



5 Node-Red

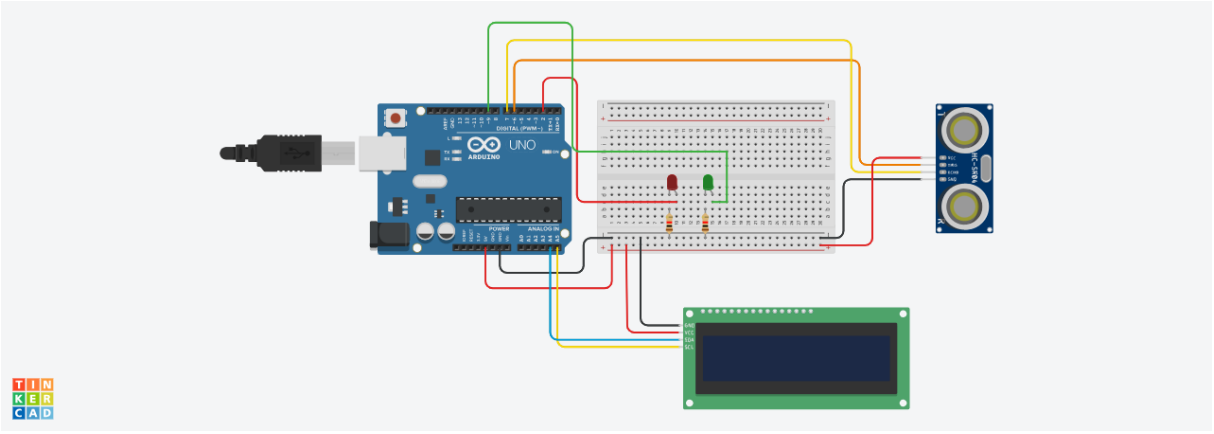
3. Raspberry Pi:

The Raspberry Pi acts as a robust and compact computing unit. It plays a pivotal role in processing data received from the Arduino Uno, hosting the Node-RED server, and supporting the dynamic dashboard. The Raspberry Pi's capabilities contribute to the system's responsiveness and real-time functionality.

6 Raspberry Pi

III. WIRING:

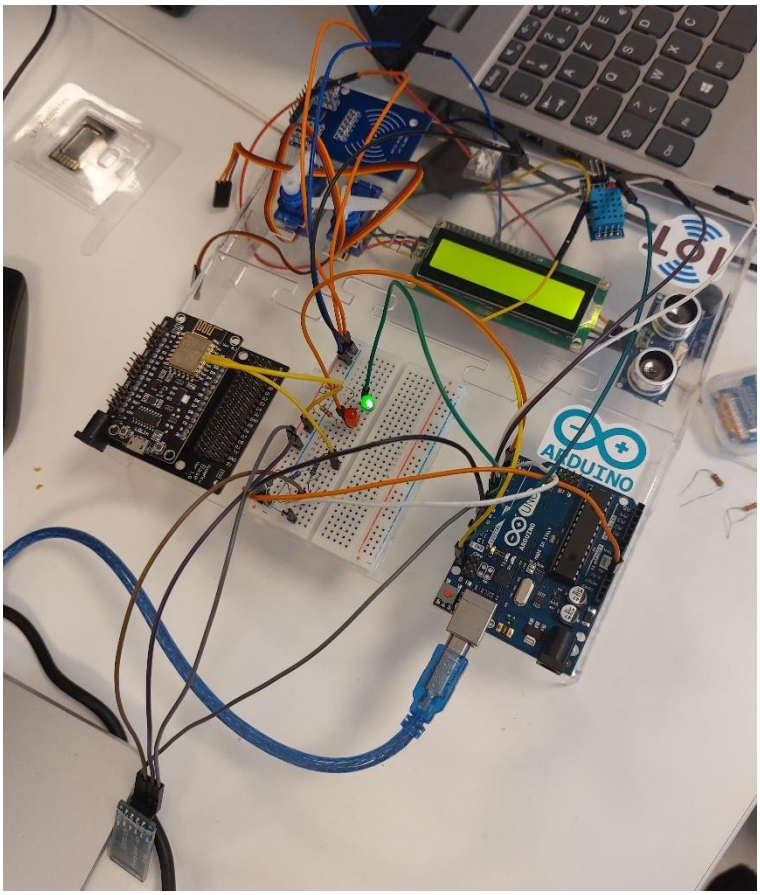
The system's hardware components are intricately connected to ensure seamless functionality. The ultrasonic sensor, responsible for measuring water levels, is connected to the Arduino Uno. This sensor utilizes ultrasonic waves to determine the distance to the water surface accurately. The Arduino Uno processes this data and, in conjunction with the Node-RED flow on the Raspberry Pi, facilitates the communication between components. Actuators, such as LEDs for visual indication and an LCD for additional feedback, are also connected to the Arduino Uno. This wiring configuration allows for efficient data flow and real-time monitoring of water levels.



7 Wrining done in Tinkercad

List of the components used:

Name	Quantity	Component
U1	1	Arduino Uno R3
DIST1	1	Ultrasonic Distance Sensor
D1	1	Red LED
D2	1	Green LED
R1, R2	2	1 kΩ Resistor
U2	1	PCF8574-based, 32 LCD 16 x 2 (I2C)



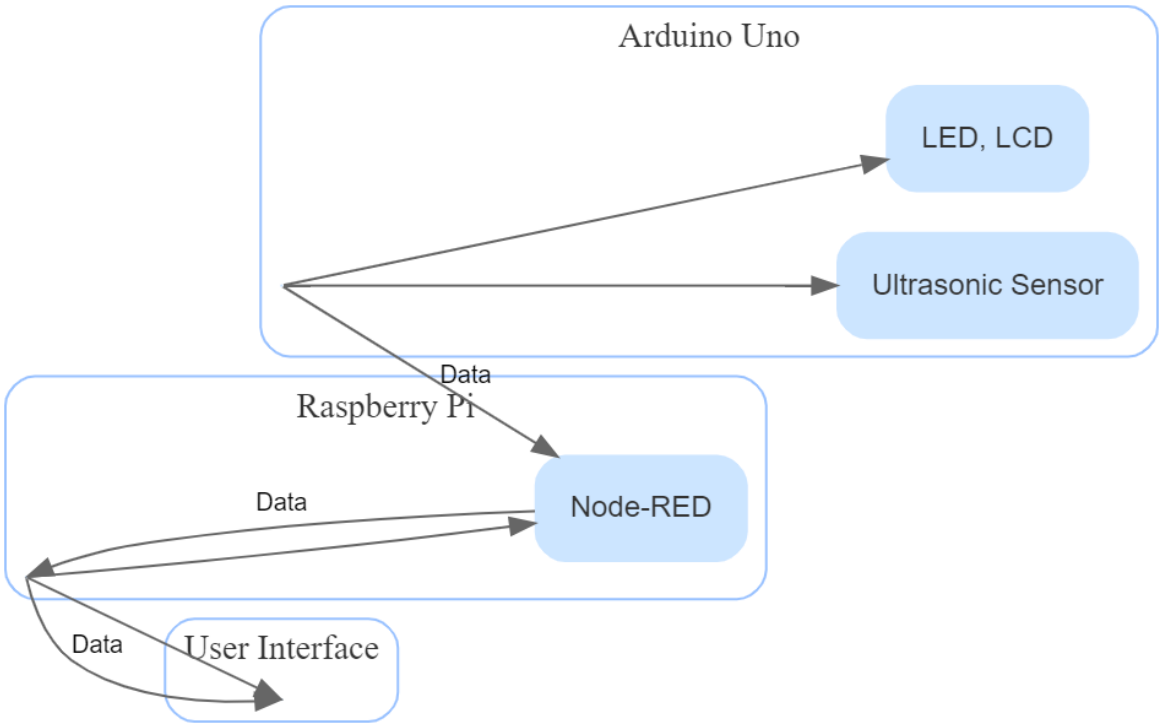
8 Wiring picture.

IV. BASIC BLOCK DIAGRAM AND FLOW CHARTS

1. Block Diagram:

The system's architecture can be represented through a comprehensive block diagram, illustrating the interconnections between key components. The following components are essential for the system's functionality:

1. **Ultrasonic Sensor:** Detects water level.
2. **Arduino Uno:** Processes sensor data and controls actuators.
3. **LEDs and LCD:** Actuators indicating water level.
4. **Raspberry Pi:** Facilitates communication between Arduino Uno and IoT platform.
5. **Node-RED:** Manages data flow and visualization.
6. **User Interface/dashboard:** Displays real-time water level data.



9 Block diagram.

2. Flow Chart:

The flow chart delineates the sequential steps and decision points within the system. It includes the following key processes:

Data Acquisition:

- Ultrasonic sensor measures water level.
- Arduino Uno processes sensor data.

Actuation:

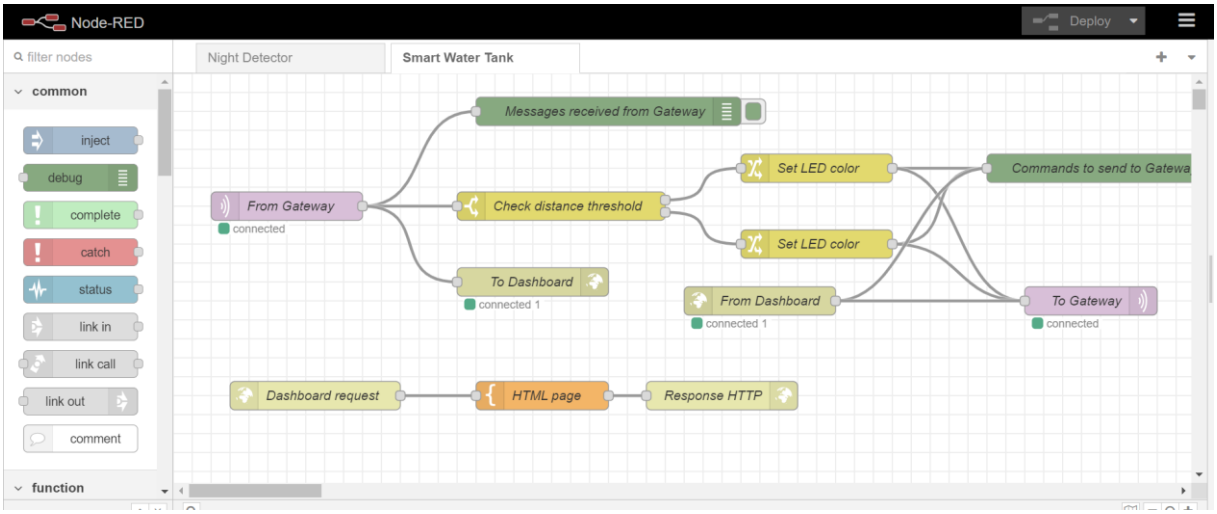
- LEDs and LCD display the water level status.
- User-defined thresholds trigger visual alerts.

Communication:

- Arduino Uno sends data to Raspberry Pi.
- Raspberry Pi facilitates communication with Node-RED.

Data Visualization:

- Node-RED processes data for visualization.

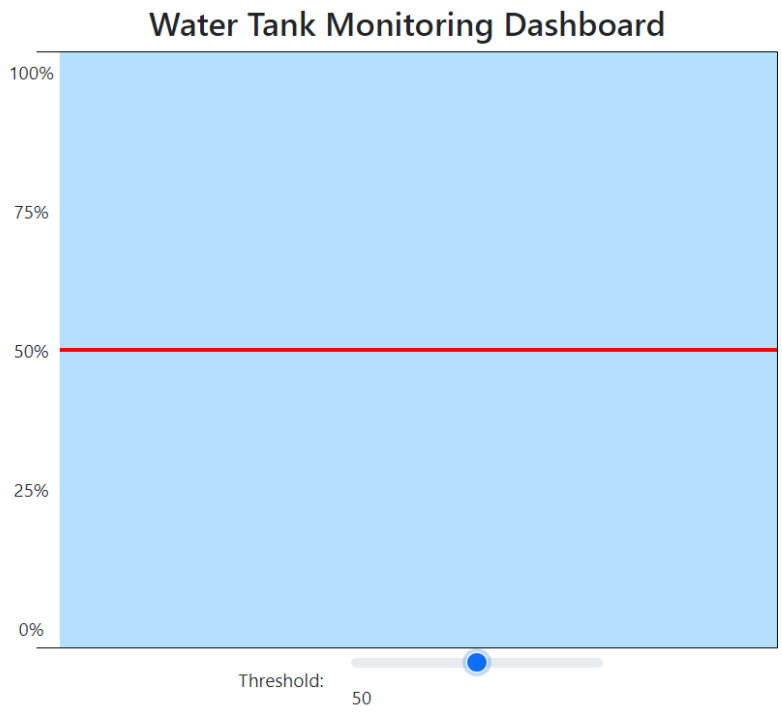


10 Node-Red Flow

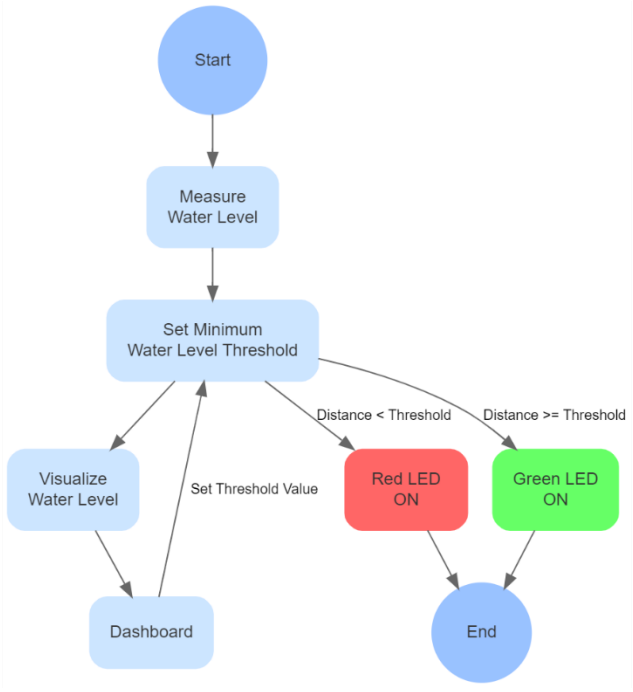
- Real-time water level displayed on the user interface.

User Interaction:

- Users can modify threshold levels through the dashboard.



11 dashboard



12 Flow chart

V. SYSTEM DESIGN AND IMPLEMENTATION

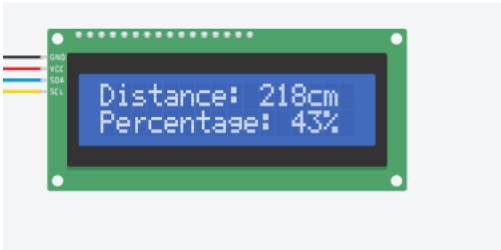
In this section, we delve into the intricacies of the system's design and the practical steps taken for its implementation. The architecture outlined in the previous sections comes to life

as we detail the deployment of hardware components and the development of the software ecosystem.

1. Hardware Implementation:

The physical realization of our system involves the meticulous connection and placement of components. The Ultrasonic Sensor, acting as the sensory input, is strategically positioned within the water tank to ensure accurate measurements. The Arduino Uno, serving as the central processing unit, orchestrates the entire operation. Its connections to the Ultrasonic Sensor, LEDs, and LCD form the backbone of the hardware implementation. Through careful wiring, the system ensures seamless communication, enabling real-time monitoring and actuation.

The LEDs serve as intuitive indicators of water level status, while the LCD provides detailed information for a comprehensive user experience.

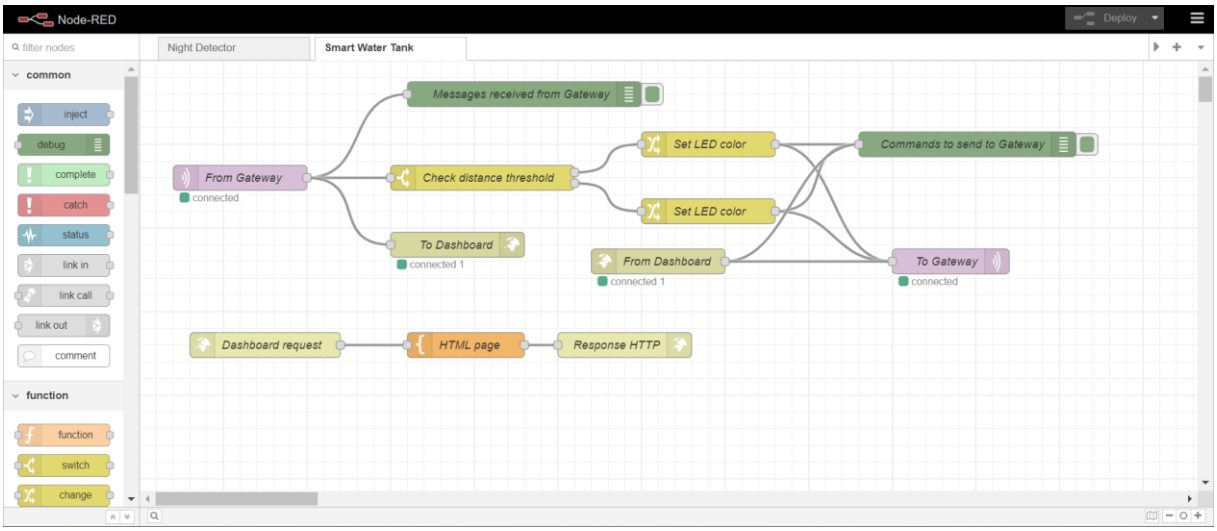


13 LCD displaying distance measured and percentage.

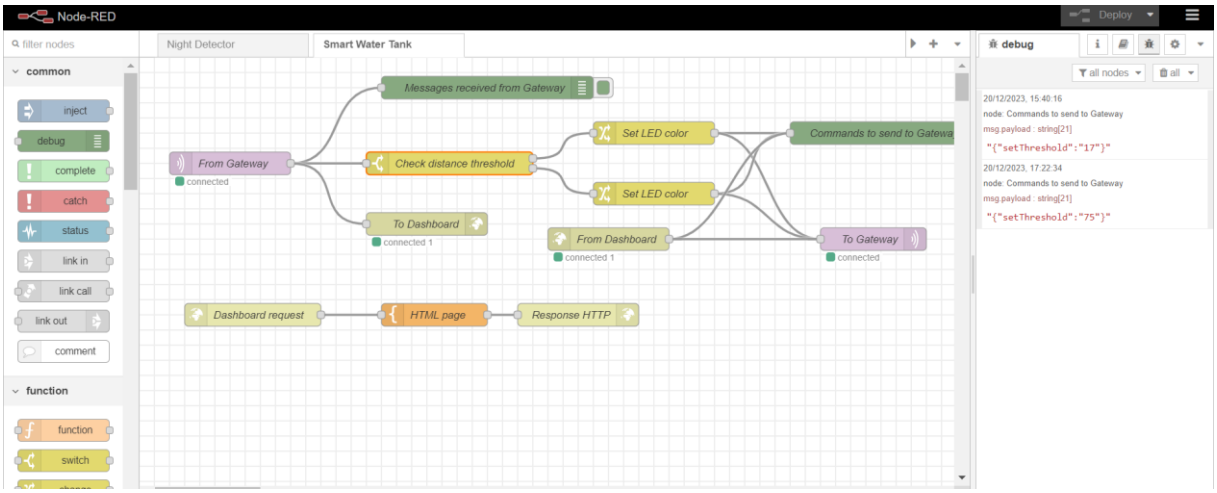
2. Software Architecture:

The software architecture is a symphony of code and logic designed to ensure the seamless interaction of hardware components. The Arduino Uno, with its versatile programming capabilities, executes the code responsible for data processing, actuation, and communication. The Node-RED flow on the Raspberry Pi plays a pivotal role in managing the data flow between the Arduino Uno and the user interface.

Node-RED's acts as a bridge, facilitating communication between the local hardware setup and the cloud-based dashboard. This two-tiered communication ensures that data is not only processed in real-time but also made available for remote monitoring and analysis.



14 Node-RED flow



15 Node-RED Flow 2

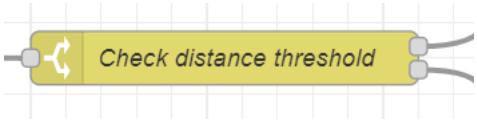
The Node-RED flow is constructed with a series of nodes, each playing a distinct role in the system's functionality. Let's explore the flow and the role of each function:

1. **MQTT In Node:**
    - *Name:* From Gateway

    - *Function:* Listens for incoming telemetry data from the MQTT broker.
2. **Debug Node:**
    - *Name:* Messages received from Gateway

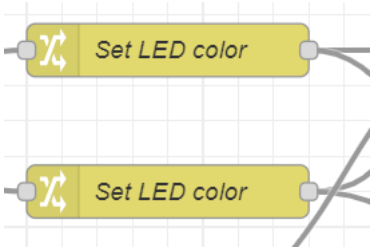
- *Function:* Displays received messages for debugging purposes.

3. **Switch Node:**
   - *Name:* Check distance threshold



   - *Function:* Diverts the flow based on whether the water level is below or above a set threshold (50 in this case).

4. **Change Nodes (Two Instances):**
   - *Name:* Set LED color



   - *Function:* Sets the color (green or red) based on the water level threshold.

5. **Debug Node:**
   - *Name:* Commands to send to Gateway

   - *Function:* Displays the commands to be sent for debugging purposes.

6. **MQTT Out Node:**
   - *Name:* To Gateway

   - *Function:* Sends commands (LED color, etc.) to the Gateway via MQTT.

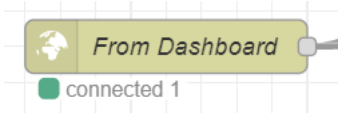
7. **WebSocket Out Node:**
    - *Name:* To Dashboard

    - *Function:* Sends data to the user interface for real-time visualization.
8. **HTTP In Node:**
    - *Name:* Dashboard request

    - *Function:* Listens for requests related to the dashboard.
9. **HTTP Response Node:**
    - *Name:* Response HTTP

    - *Function:* Sends HTTP response for dashboard requests.
10. **Template Node:**
    - *Name:* HTML page

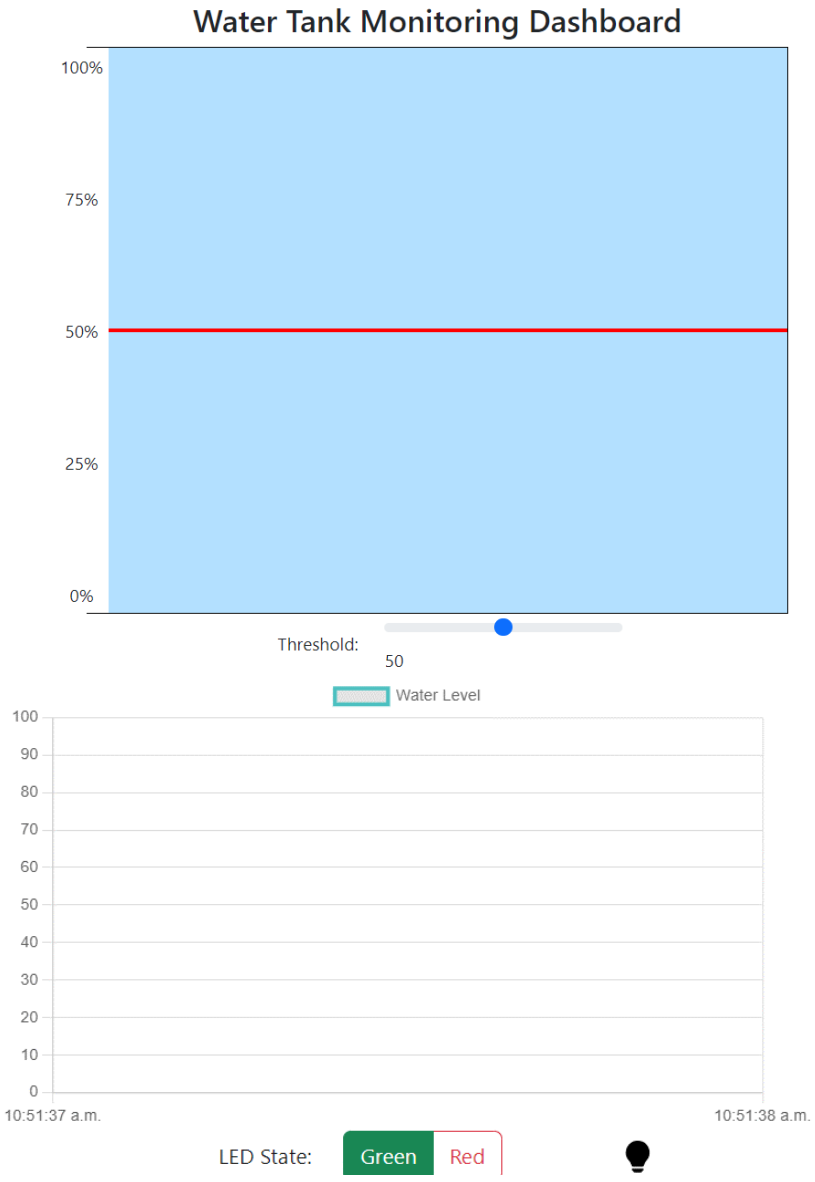
    - *Function:* Generates an HTML page for the dashboard with dynamic elements for water tank visualization, threshold input, historical charts, and LED state indication.
11. **WebSocket In Node:**
    - *Name:* From Dashboard



- *Function:* Listens for incoming data from the dashboard, such as LED state changes.

3. Dashboard:

The heart of user interaction lies in the design and functionality of the dashboard. Users can seamlessly interact with the system by customizing minimum water level thresholds. This customization not only enhances user engagement but also allows individuals to tailor the system to their specific water management needs.



16 Dashboard

4. Testing and Optimization:

A critical phase in the implementation involves rigorous testing to validate the system's accuracy, responsiveness, and reliability. Various scenarios, including fluctuating water levels and threshold modifications, are simulated to ensure the system performs as intended. Iterative optimization is carried out to address any discrepancies and enhance the overall efficiency of the system.

VI. FUTURE SCOPE

The versatility of this project opens doors to a myriad of applications, extending its utility across diverse settings. The installation of this water monitoring system holds promise in the following areas:

1. **Private Residences and Bungalows:**

Providing homeowners with a comprehensive solution for effective water tank management.

2. **Housing Societies:**

Scaling up to cater to the water monitoring needs of entire residential communities.

3. **Apartments:**

Offering a practical and efficient solution for multi-unit dwellings to monitor and optimize water usage.

4. **Institutions (Schools, Colleges, Hostels):**

Enhancing water resource management in educational institutions and hostels, ensuring a sustainable approach.

5. **Hospitals:**

Addressing the critical need for precise water level monitoring in healthcare facilities for various applications.

VII. Conclusion

In conclusion, the Smart Water Tank Monitoring System seamlessly integrates Arduino Uno and Node-RED on Raspberry Pi, ensuring efficient data processing and responsive communication. Node-RED's visual programming enhances system efficiency, enabling real-

time data processing and remote monitoring. The project's versatility extends its applications from private homes to municipal tanks, emphasizing its adaptability. This system represents a transformative approach to water management, offering sustainable solutions for diverse settings. In essence, it addresses immediate concerns and promotes conscientious water usage, showcasing the power of technology in enhancing efficiency and responsibility in resource management.