International Institute of Information Technology, Hyderabad

MONITORING WATER-LEVEL IN OVERHEAD TANKS

Anishka Sachdeva, Satyam Viksit Pansari, Subodh Sondkar

Abstract— **Measuring the water levels in tanks can be useful to solve many day to day problems of common people. The problems of leakage of water, tank cleanliness issues etc. require attention. Using the OneM2M layer which offers interoperability, water level data can be posted onto it and Data Analytics of water levels in different ways can help us find remedies to the current problems faced by the world.**

I. INTRODUCTION

A. *Problem Statement*

You are given a water tank. You have to make a project to monitor the water-level by determining what range it belongs to. You also have to upload data on a server using OneM2M framework.

B. *Purpose of the System*

The main purpose of water level monitoring and control is:

1. Aids in saving water.

2. Since the demand of electricity is very high in our country, it will help to save electricity i.e. when the tank has been completely filled, the user can switch off the motor accordingly.

C. *Scope*

* The project gives what range the water level in a tank is in.

* It uploads the boolean values of the water level on the OneM2M server.

D. *Overview*

We have used an ESP32 Node MCU and a photoelectric water / liquid level sensor to determine the water level in an overhead tank. The liquid level sensor is suspended inside the tank at an appropriate height. The ESP32 Node MCU, along with the adapter (for power supply) is stored inside an IP65 enclosure, which protects them from environment (for e.g. rain). The ESP32 Node MCU is connected to WiFi, and we have sent the obtained data to a OneM2M server.

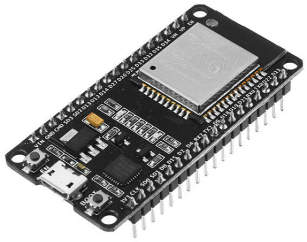
II. DESIGN DOCUMENT

A. *System Requirements*

* AC Power Supply
* WiFi Connectivity

B. *System Specifications*

* ESP32 Node MCU
* DFRobot Gravity Photoelectric Water / Liquid Level Sensor
* IP65 enclosure
* OneM2M Server ”http://onem2m.iiit.ac.in:443/webpage/”



C. *Stakeholders*

* Smart City Project
* Dr. Aftab Hussain
* Our Team
* IIIT Hyderabad
* Assigned TAs

D. *Design Entities*

* IP65 Box
* ESP32 Node MCU
* DFRobot Gravity Photoelectric Water / Liquid Level Sensor

E. *Design Details*

1) Entity Interaction: The DFRobot Gravity Photoelectric Water / Liquid Sensor is suspended from the IP65 Box. Inside the box, it is connected to the ESP32 Node MCU. There is an adapter in the box which converts AC mains to 5V which powers the ESP32.

2) Conceptual Flow: The Liquid Sensor is suspended in the tank. If it is in contact with water (i.e. water level is higher than the height at which the sensor is suspended), then it will read a ”1”, else it will read a ”0”. The sensor will send this data to the ESP32. The ESP32 board will communicate via JioFi and will send the modified (meaningful) data to the OneM2M server.

F. *Operational Requirements*

1) System Needs: A constant DC power suppply of 5V, along with a strong Wifi Connection.

2) UI Design: We were provided the user interface by the owner of the server and our UI is a webpage on ”onem2m.iiit.ac.in/webpage”.

3) Analytical System: Our analytical system is based in a python code used to get Water Level data from Server. The values retrieved will be plotted on a graph using Python.

III. OPERATIONAL DOCUMENT

A. *Introduction*

1) Objective: Objective is to monitor water level in terms of boolean values and upload data to webpage using OneM2M Server.

2) Scope: Main Goal is to Provide User with the water level of the water container at any interval of time. Here, we have set that interval as 10 minutes.————

B. *Product Operational Requirements*

1) Operational Environment: Operating Environment for the Project is any location with Power Supply and WiFi connectivity which doesn’t come in Contact with Water.

2) Power and Other Interfaces: - Micro-controller is connected to AC Power.
- Micro-controller is connected to JioFi.

C. *System Working Model*

1) Base State: In the base state the model returns 0, having the water level below 50%.

2) Working State: In the working state the model returns a 1, having the water level above 50%.