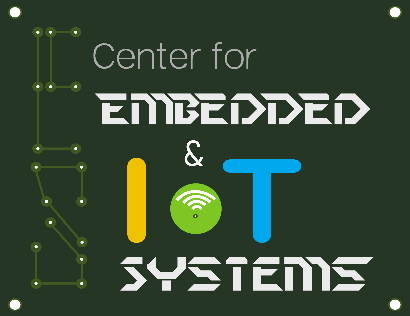
# WATER MONITORING SYSTEM



A project report submitted in partial fulfilment of requirement for the course

On

## Fundamentals of IoT

By

**RAMA KARTHIKEYA (2003A52065)**

**BOLUKONDA VAMSHI (2003A52022)**

**MOHD AFROZ (2003A52102)**

**PANUGANTI NISHWITH (2003A52128)**

Under the guidance of

**Dr. Sumit Gupta**

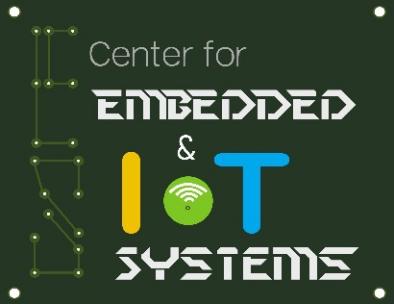
Asst. Prof. Department of ECE

Department of Computer Science Engineering



**Center for Embedded Systems and Internet of things**

**SR UNIVERSITY**



**CERTIFICATE**

This is to certify that the course project entitled **“WATER MONITORING SYSTEM”** is the bonafied work carried out by RAMA KARTHIKEYA (**2003A52065**), BOLUKONDA VAMSHI (**2003A52022**),PANUGANTI NISHWITH (**2003A52128**) and MOHD AFROZ (**2003A52102**) in the partial fulfilment of the requirement for the award of course Internet of Things during the academic year 2022-2023 under our guidance and Supervision.

**Dr. Sumit Gupta**

Asst. Prof.

Department of ECE

# Abstract

As people are busy in doing their daily chores it becomes difficult to track water level in the tanks situated at their home’s rooftop. And as tanks are located on top of most of the houses the individual need to go up and downstairs. During which wastage of water happens. So to solve this issue we have made a Water-Level-Monitoring-System using ESP32 and Ultra-Sonic Sensor (HC-SR04). Water is an essential need. So, not only monitoring it can also be used to stop water wastage.

**CONTENTS**

**Chapter No. Title Page No.**

**1 Introduction 1**

**2 System Description 2**

**2.1 Block Diagram 3**

**3 Hardware & Software Tools 4**

**4 Implementation 6 5 Results 8**

**6 Conclusion 11**

**References 12**

# INTRODUCTION

Over the past years, people are unable to trace the water quantity in the water tank

or container. Even by not tracing water level in the tank there is other issue of water flow after its full, causing the water wastage. Water is one of the most valuable and important resources on planet. Energy conservation in the tank level water monitoring system is defined in this project using an Internet of Things (IOT) based idea. The major goal is to have a sensor that monitors the water level and should inform the user of the amount of water that is currently in the tank.

The ultrasonic sensor is positioned at the top of the tank where the water level will be calculated. If the water's distance from the sensor increases, it indicates that the tank's water level is getting low, and if it hits a certain threshold, the system should send the user a warning message.

* 1. **Existing System**

This system does not include the water level monitor mode, high cost & harder maintenance. Low genuineness of water level monitor. User experience is demonstrable. It gives in accuracy & more time consumption. This system has not advanced outcome of water level monitor.

* 1. **Proposed System**

This water level monitoring system based on new Android application and to preserve more energy. Very easy to maintenance. Low cost and flexible water level monitor. This system used Blynk application to point out the water level. It is used to circuit breaker (on/off) and water level monitor by using Blynk Android application in online mode. The Blynk application is also used to do scheduled process of motor on/off & monitor the water level. It can detect error-free level of water in tank.

# SYSTEM DESCRIPTION

# 2.1 Block Diagram

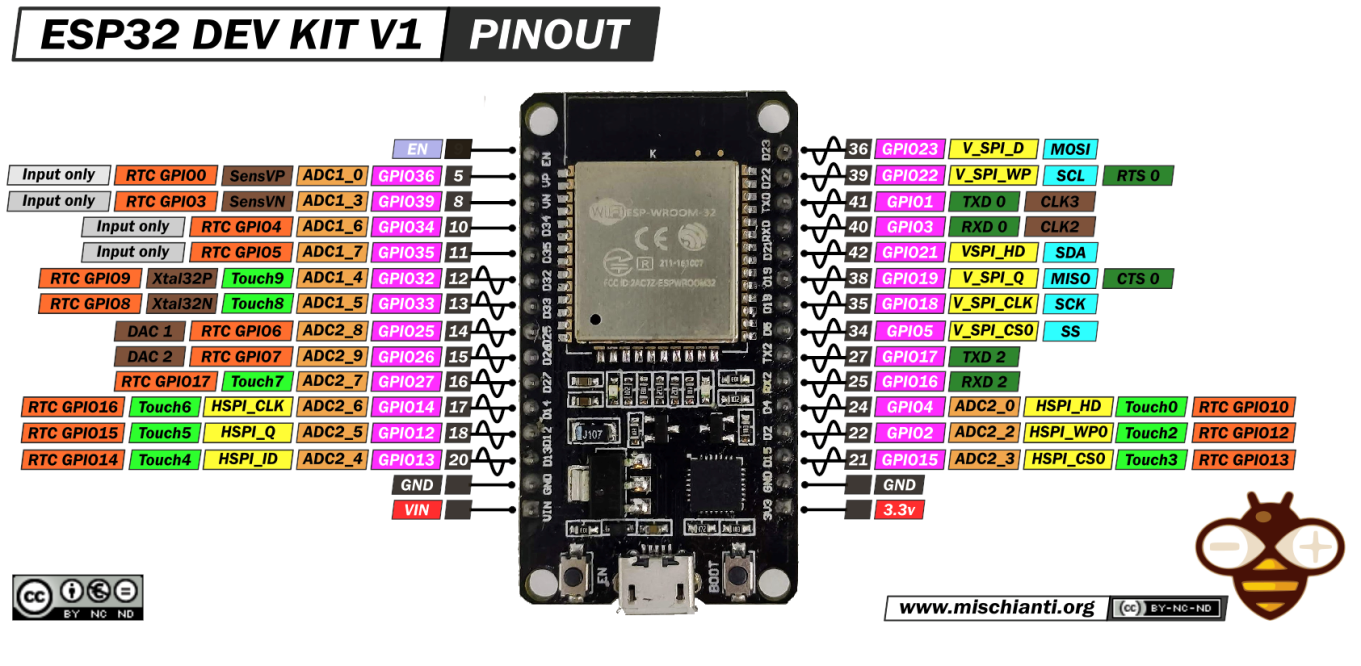
# Hardware & Software Tools

Hardware tools which we have used are mentioned below:

**ESP32 DEVKIT V1:** It is based on the ESP32 microcontroller that boasts Wifi, Bluetooth, Ethernet and Low Power support all in a single chip. The entire solution takes up the least amount of printed circuit board area. This board is used with 2.4 GHz dual-mode Wi-Fi and Bluetooth chips by TSMC 40nm low power technology, power and RF properties best, which is safe, reliable, and scalable to a variety of applications.

**Few Specifications:**

* Operating Voltage: 3.3V
* Input Voltage: 7-12V
* Digital I/O Pins (DIO): 25
* Analog Input Pins (ADC): 6
* Analog Outputs Pins (DAC): 2
* UARTs: 3
* SPIs: 2
* I2Cs: 3
* Flash Memory: 4 MB
* SRAM: 520 KB



**ULTRASONIC SENSOR (HC-SR04):** An ultrasonic sensor is an electronic device that measures the distance of a target object by emitting ultrasonic sound waves, and converts the reflected sound into an electrical signal. Ultrasonic waves travel faster than the speed of audible sound (i.e. the sound that humans can hear). Ultrasonic sensors have two main components: the transmitter (which emits the sound using piezoelectric crystals) and the receiver (which encounters the sound after it has travelled to and from the target). It consists of 4 pins. They are VCC, Trig, Echo and Ground. This sensor provides 2cm to 400cm of non-contact measurement functionality with a ranging accuracy that can reach up to 3mm.



**Software tools which we have used are mentioned below:**

**Blynk:** Blynk is an IoT platformforiOS or Android smartphones that is used to control Arduino, Raspberry Pi and NodeMCU via the Internet. This application is used to create a graphical interface or human machine interface (HMI) by compiling and providing the appropriate address on the available widgets. It can display sensor data, can store the data and visualize it and do many more things.

**Arduino IDE:** The Arduino IDE is an open-source software, which is used to write and upload code to the Arduino boards. The IDE application is suitable for different operating systems such as Windows, Mac OS X, and Linux. It supports the programming languages C and C++. Here, IDE stands for Integrated Development Environment.

It contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them.

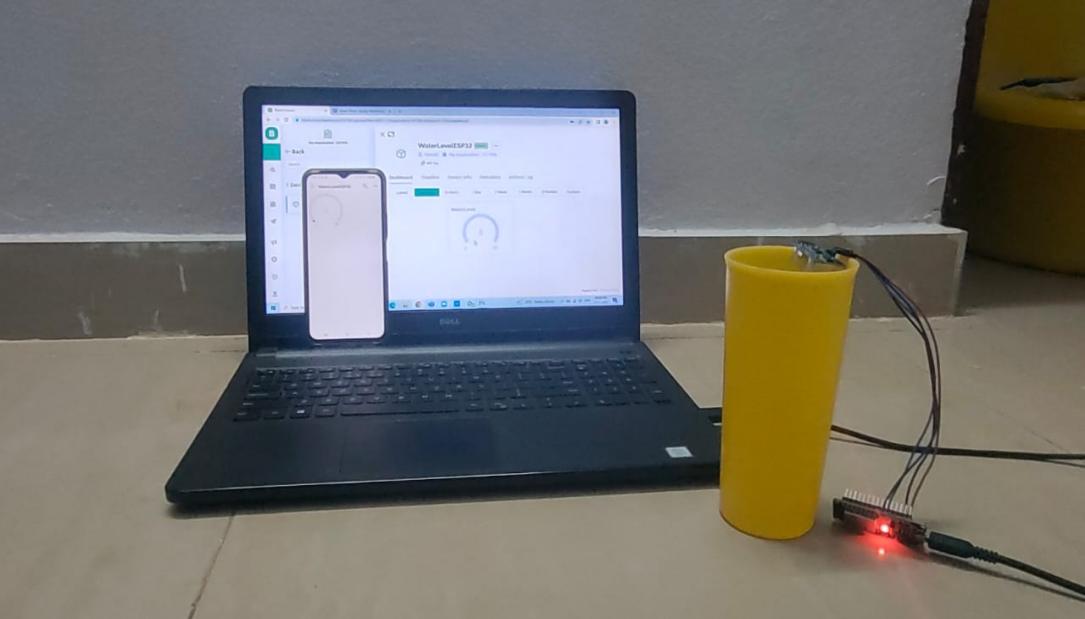
Programs written using Arduino Software (IDE) are called **sketches.** These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom righthand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

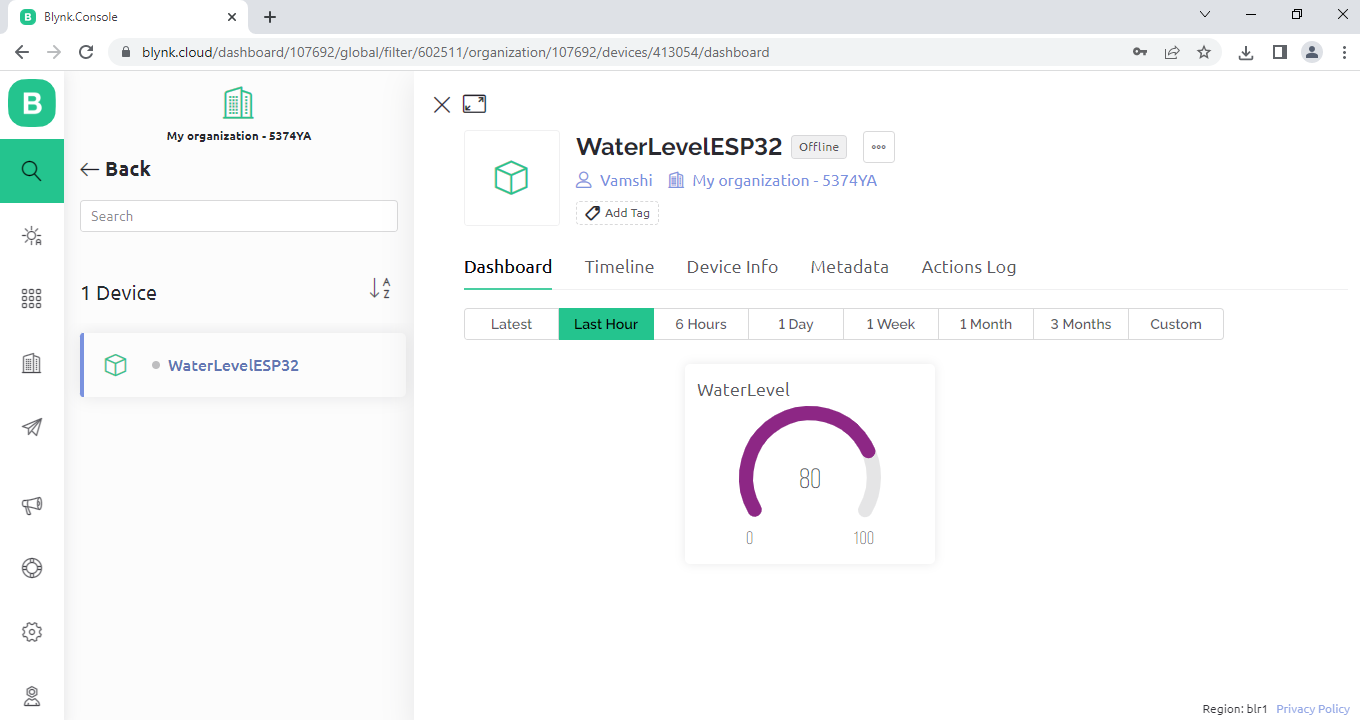
* + 1. **IMPLEMENTATION**

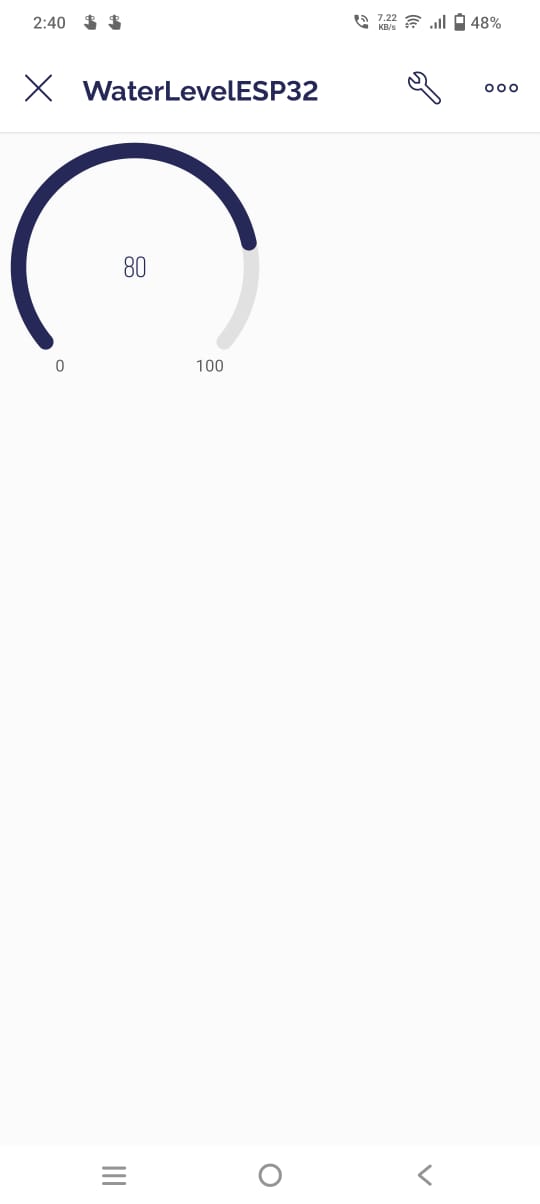
1. Collect the necessary components required for project.
2. Then give the connections like VCC of sensor connected to VIN pin of esp32 and GND pin of sensor connected to GND of esp32, Trig pin of sensor connected to GPIO26 of esp32 and Echo pin of sensor connected to GPIO27 of esp32.
3. Create an account and sign into blynk.io
4. Create a new template in blynk.io after Login.
5. Then add datastreams to it and select Virtual Pin Datastream and fill the necessary details. (like name of datastream, virtual pin of datastream etc.)
6. Then enable the datastream to automation.
7. Then go to Web DashBoard and select Gauge widget and link it with the above virtual pin created.
8. Then save the changes.
9. Now select “New Device” and select “From Template” option.
10. Then it generates Template ID, Device Name and Blynk Authorization Token and these 3 values should be pasted in the sketch/code.
11. Then Setup the Mobile Dashboard in mobile after installing Blynk App from GooglePlayStore or AppleAppStore.
12. Go to Developer’s Mode in Mobile and select the Gauge from WidgetBox.
13. Install the necessary Libraries like WiFi.h, WiFiClient.h, BlynkSimpleEsp32.h
14. Then write the code and Compile and then Upload it in esp32.

# RESULT

This would result in helping to avoid unnecessary wastage of water and helps you track the water level in tank without any physical stress for the individual by displaying the water level percentage with respect to the depth of the tank in the mobile or Blynk web dashboard.



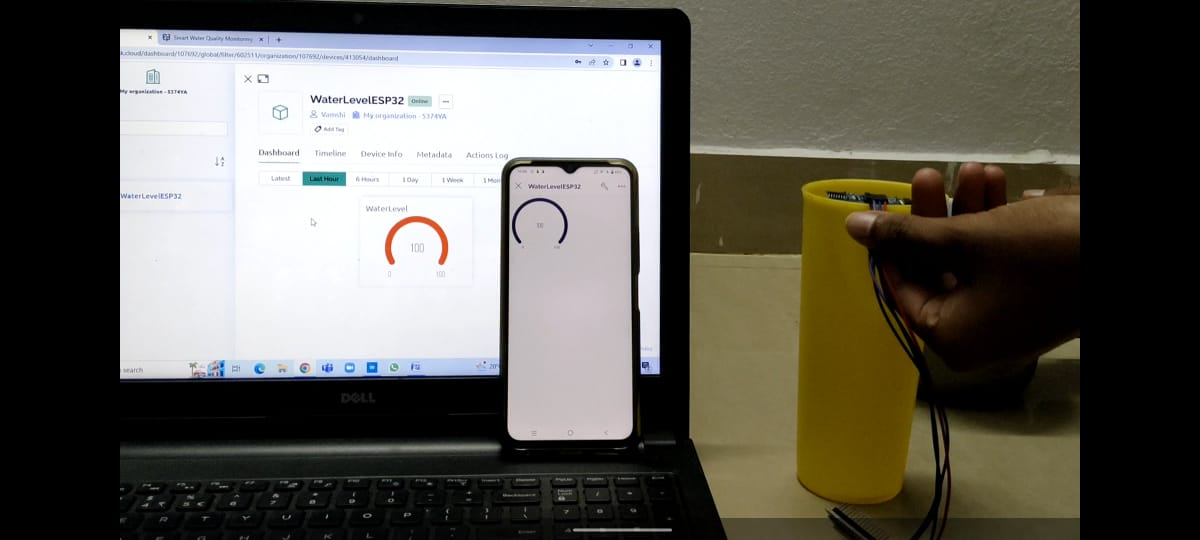












# CONCLUSION

From this project, we learnt how to use ultrasonic sensor for monitoring the water level, which is reducing the burden on the individual and also creating smart widgets in Blynk app and using them for understanding the sensor data. It also helps to save the water.

We can also moniter the water quantity in the tank.

# REFERENCES

1. Bórquez López, R.A.; Martinez Cordova, L.R.; Gil Nuñez, J.C.; Gonzalez Galaviz, J.R.; Ibarra Gamez, J.C.; Casillas Hernandez, R. Implementation and evaluation of open-source hardware to monitor water quality in precision aquaculture. *Sensors* **2020,** 20, 6112.
2. Salim, T.I.; Haiyunnisa, T.; Alam, H.S. Design and implementation of water quality monitoring for eel fish aquaculture. In Proceedings of the International Symposium on Electronics and Smart Devices (ISESD), IEEE, November 2016.
3. Khatri, P.; Gupta, K.K.; Gupta, R.K. Smart Water Quality Monitoring System for Distribution Networks. In proceedings of International Conference on Sustainable Computing in Science, Technology and Management (SUSCOM), Amity University Rajasthan, Jaipur-India, February 2019.
4. <https://ijisrt.com/wp-content/uploads/2018/04/IoT-Based-Real-time-water-Monitoring-System-for-Smart-City.pdf>
5. <https://www.ripublication.com/awmc17/awmcv10n5_24.pdf>
6. https://www.google.com/search?q=ultrasonic+sensor&tbm=isch&ved=2ahUKEwiOgPP46Lb7AhX8K7cAHXG7A9EQ2-cCegQIABAA&oq=ultras&gs\_lcp=CgNpbWcQARgAMgcIABCxAxBDMgQIABBDMgcIABCxAxBDMgQIABBDMggIABCABBCxAzIECAAQQzILCAAQgAQQsQMQgwEyBwgAELEDEEMyCAgAEIAEELEDMgUIABCABDoECCMQJzoHCCMQ6gIQJ1DvA1jsLGCZM2gCcAB4BIAB9AGIAbwRkgEGMC4xMC4zmAEAoAEBqgELZ3dzLXdpei1pbWewAQrAAQE&sclient=img&ei=OAB3Y46EGfzX3LUP8faOiA0&bih=601&biw=1280#imgrc=Mnu1iBuVSy3WSM
7. https://www.google.com/search?q=esp32&tbm=isch&ved=2ahUKEwjKlumF6bb7AhURKbcAHZHOCigQ2-cCegQIABAA&oq=esp32&gs\_lcp=CgNpbWcQAzIECCMQJzIECCMQJzIHCAAQsQMQQzIHCAAQsQMQQzIECAAQQzIECAAQQzIECAAQQzIHCAAQsQMQQzIECAAQQzIFCAAQgAQ6BwgjEOoCECc6CAgAEIAEELEDUJAGWKUZYNYbaAFwAHgEgAHNAYgB0w2SAQYwLjEwLjGYAQCgAQGqAQtnd3Mtd2l6LWltZ7ABCsABAQ&sclient=img&ei=UwB3Y4qhH5HS3LUPkZ2rwAI&bih=601&biw=1280#imgrc=Q3UFV1-higOSgM
8. https://www.google.com/search?q=ardunio+ide&tbm=isch&ved=2ahUKEwip3Kae6bb7AhWEHbcAHTt5Bq8Q2-cCegQIABAA&oq=ardunio+ide&gs\_lcp=CgNpbWcQAzIFCAAQgAQyCQgAEIAEEAoQGDIJCAAQgAQQChAYMgkIABCABBAKEBgyCQgAEIAEEAoQGDIJCAAQgAQQChAYMgkIABCABBAKEBgyCQgAEIAEEAoQGDIJCAAQgAQQChAYOgQIIxAnOgYIABAFEB5QygRYqxFg1RdoAHAAeACAAacBiAGrCJIBAzAuN5gBAKABAaoBC2d3cy13aXotaW1nwAEB&sclient=img&ei=hgB3Y6mBNIS73LUPu\_KZ-Ao&bih=601&biw=1280#imgrc=msGj\_O\_WvdZqPM
9. <https://www.google.com/search?sxsrf=ALiCzsaTHTUEGjzWLp1ikcFm6C091gyRPg:1668743398129&q=blynk+app&tbm=isch&sa=X&ved=2ahUKEwi1xd3L6bb7AhX_TmwGHdyLDYsQ0pQJegQIEBAB&biw=1280&bih=601&dpr=1.5#imgrc=-83N-9WcTMCuaM>
10. https://iotdesignpro.com/projects/iot-controlled-led-using-esp32-with-blynk-app