SMART WATER MANAGEMENT

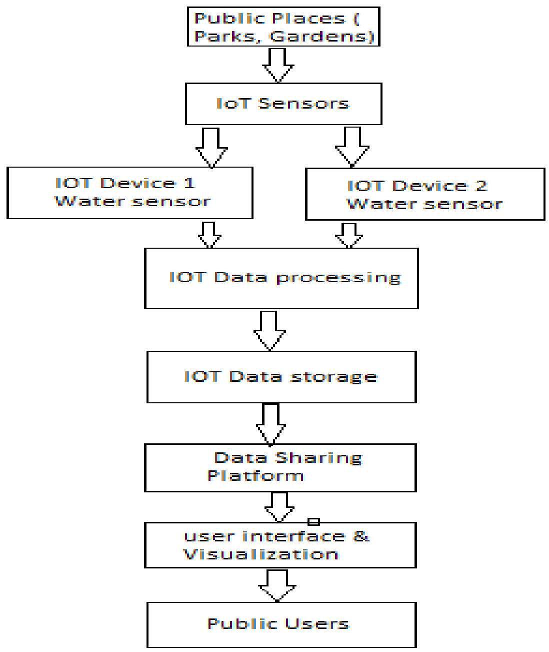
Note:*instead of respberry pi we use the esp32 because we didn't get the free simulator website for respberry pi*

PROBLEM STATEMENT:

Water management in public parks presents several unique challenges. These spaces often have extensive irrigation needs to maintain lush landscapes, which can lead to water overuse. Variations in weather and evapotranspiration rates can make it difficult to determine precise water requirements. Additionally, leakages, system inefficiencies, and irregular maintenance practices can result in water wastage. To tackle these challenges, real-time monitoring is essential. It allows for immediate adjustments in water usage based on factors like weather conditions and soil moisture levels, preventing over-irrigation, conserving water, and ensuring the sustainability of these cherished public areas. Furthermore, real-time monitoring enables quick detection and response to issues like leaks, ultimately promoting efficient water management in parks.

OBJECTIVE :

The IoT-Based Real-Time Water Consumption Monitoring project presents an innovative and impactful solution to address the pressing issue of water conservation in public spaces, particularly parks and gardens. In many regions, the unsustainable use of water resources poses a significant environmental challenge. To combat this issue, our project leverages the power of IoT technology to develop a comprehensive system that monitors water consumption in real-time. By making this vital information accessible to the public and relevant stakeholders, we aim to drive awareness, promote responsible water usage, and ultimately contribute to the preservation of our precious water resources .This initiative reflects our commitment to sustainability and a brighter, water-efficient future.



It represents the design and deployment of IoT sensors to monitor water consumption in public places.

1. Public Places: The locations where you'll install the IoT sensors, such as parks and gardens.

2. IoT Sensors: These sensors (IoT devices) are responsible for monitoring water consumption. They collect data and transmit it to the IoT data processing component.

3. IoT Data Processing: This component processes and validates the data received from the sensors before storing it.

4. IoT Data Storage: Data is stored securely, ensuring its availability for analysis and retrieval.

5. Data Sharing Platform: This platform provides access to real-time water consumption data through a user-friendly interface.

6. User Interface & Visualization: The interface where public users can access and visualize water consumption data. This could be a web application or a mobile app.

7. Public Users: The end-users, which can include the general public, park authorities, or environmental organizations.

We use the ESP32 ,flow sensor, Ultrasonic Distance Sensors, Water Level Sensors, Water Quality Sensors etc…

INTEGRATION APPROACH :

The IoT sensors in our project will send data to the data-sharing platform through a secure and efficient process. These sensors, placed in public areas, will collect water consumption data and transmit it using established communication protocols over the internet or wireless networks. Once received, the data will be processed and made available to the public through a cloud-based platform. Strong security measures will protect the data during transmission, ensuring its accuracy and privacy. This system enables real-time monitoring of water usage and promotes water conservation by providing accessible data to the community

CHALLENGES AND RISK MITIGATION:

While implementing the water monitoring system, there are potential challenges and risks that need to be addressed. These challenges include technical issues such as sensor malfunctions, data transmission errors, and compatibility concerns between sensors and the ESP32 Environmental factors like extreme weather conditions can also affect sensor performance. To mitigate these challenges, robust quality control measures are essential, including routine sensor maintenance, calibration, and redundancy in data transmission. Additionally, cybersecurity measures must be in place to protect sensitive data. Communication protocols must be secure to prevent unauthorized access to the system. The risk of public resistance or misuse of water data can be mitigated through user education and engagement initiatives, emphasizing the importance of responsible water usage. Moreover, having contingency plans for extreme weather conditions and redundancy in sensor deployment ensures system resilience. These strategies collectively help to address potential challenges and risks, ensuring the system's effectiveness and reliability.

Here you can see the result our project gives the output and the It sence the water level and the data Then the how many times the tank is fulled is calculate the how many liter is consumed per day and we Can control the water waste "save water"

This is the link of the website check and simulate:

https://waterconsumptionmonitoring.mydurable.com/#https:/wokwi.com/projects/379653611941141505

Click the “Click here to see the simulation of the project”

This is the python script for this project:

```
BLYNK_TEMPLATE_ID = "TMPLlcLQu4bQ"

BLYNK_TEMPLATE_NAME = "water monitor"

BLYNK_AUTH_TOKEN = "OgvenxCWu9sG7-9deFGLFCLE4rWCGW7N"

ssid = "Wokwi-GUEST"

pass = ""

emptyTankDistance = 150

fullTankDistance = 40

triggerPer = 10

from Adafruit_SSD1306 import Adafruit_SSD1306


from wifi import WiFi

from wifiClient import WiFiClient

from blynkSimpleEsp32 import BlynkSimpleEsp32

from aceButton import AceButton

TRIGPIN = 27

ECHOPIN = 26

wifiLed = 2

BuzzerPin = 13

RelayPin = 14

ButtonPin1 = 12

ButtonPin2 = 33

ButtonPin3 = 32

fullpin = 25
```

```
VPIN_BUTTON_1 = V1
VPIN_BUTTON_2 = V2
VPIN_BUTTON_3 = V3
VPIN_BUTTON_4 = V4
VPIN_BUTTON_5 = V5
SCREEN_WIDTH = 128
SCREEN_HEIGHT = 32
OLED_RESET = -1
display = Adafruit_SSD1306(SCREEN_WIDTH, SCREEN_HEIGHT, Wire, OLED_RESET)
duration = 0.0
distance = 0.0
waterLevelPer = 0
toggleBuzzer = True
toggleRelay = False
modeFlag = True
conection = True
currMode = ""

auth = BLYNK_AUTH_TOKEN
config1 = ButtonConfig()
button1 = AceButton(config1)
config2 = ButtonConfig()
button2 = AceButton(config2)
config3 = ButtonConfig()
button3 = AceButton(config3)
def handleEvent1(button, eventType, buttonState):
pass
```

def handleEvent2(button, eventType, buttonState):

pass

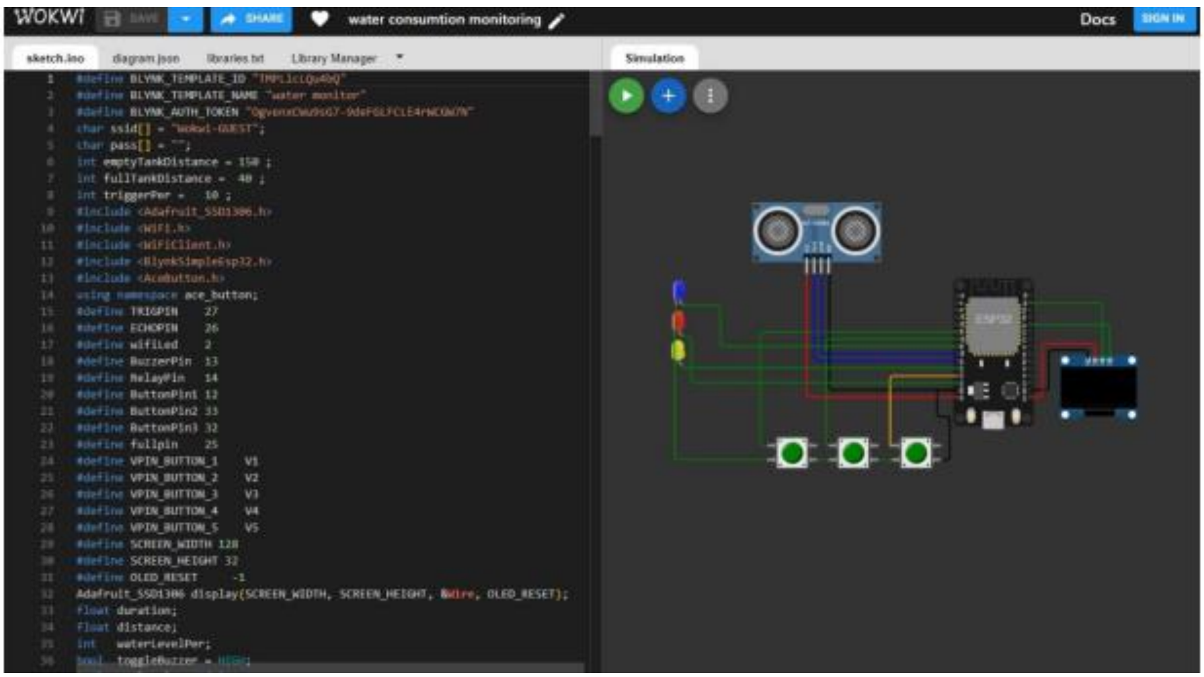
def handleEvent3(button, eventType, buttonState):

pass

timer = BlynkTimer()

def checkBlynkStatus():

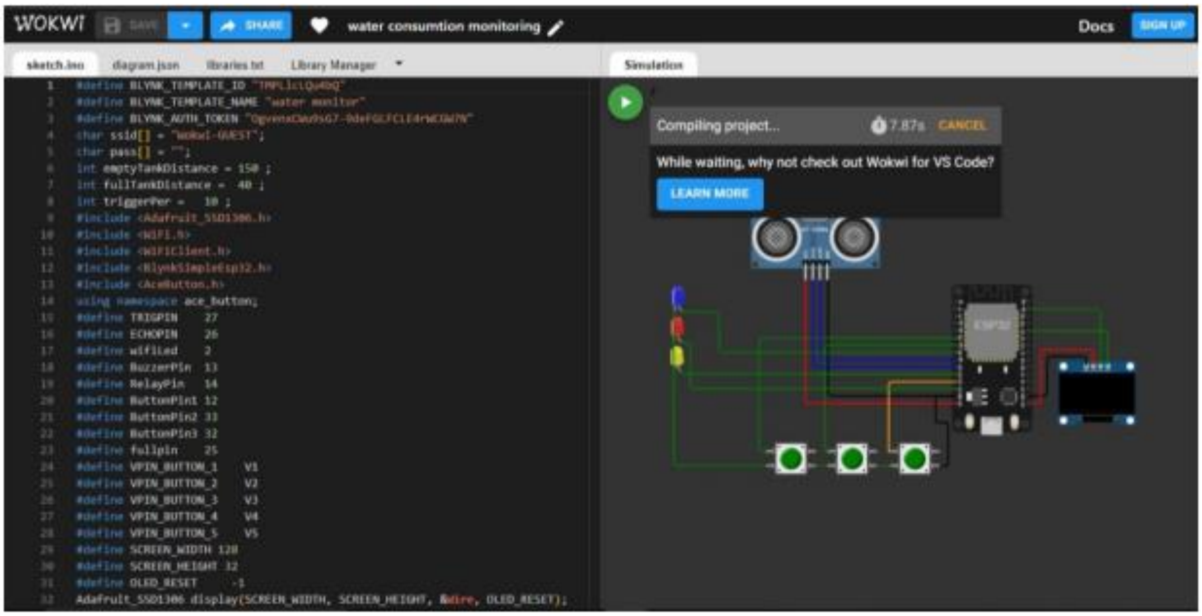
passThen this page will open



This is the simulation of the project

I change the microcontroller board because of there is no simulator for raspberry pi

Then run the program:



Output of the project

```
rst:0x1 (POWERON_RESET),boot:0x13 (SPI_FAST_FLASH_BOOT)
configsip: 0, SPIWP:0xee
clk_drv:0x00,q_drv:0x00,d_drv:0x00,cs0_drv:0x00,hd_drv:0x00,wp_drv:0x00
mode:DIO, clock div:2
load:0x3fff0030,len:1156
load:0x40078000,len:11456
ho 0 tail 12 room 4
load:0x40080400,len:2972
entry 0x400805dc
 water level is in danger
 water level is in danger
 water level is in danger
```

CONCLUSION:

Our water monitoring system is a smart solution that uses cutting-edge technology to keep an eye on the water in public place. It helps us save water, money, and the environment. It does this by using clever sensors to watch how much water there is and if it's safe to use. If there are any issues, it tells us right away on our phones. This smart system is all about making sure our public place stay beautiful and our water stays safe, and it helps us all do our part to protect the environment.