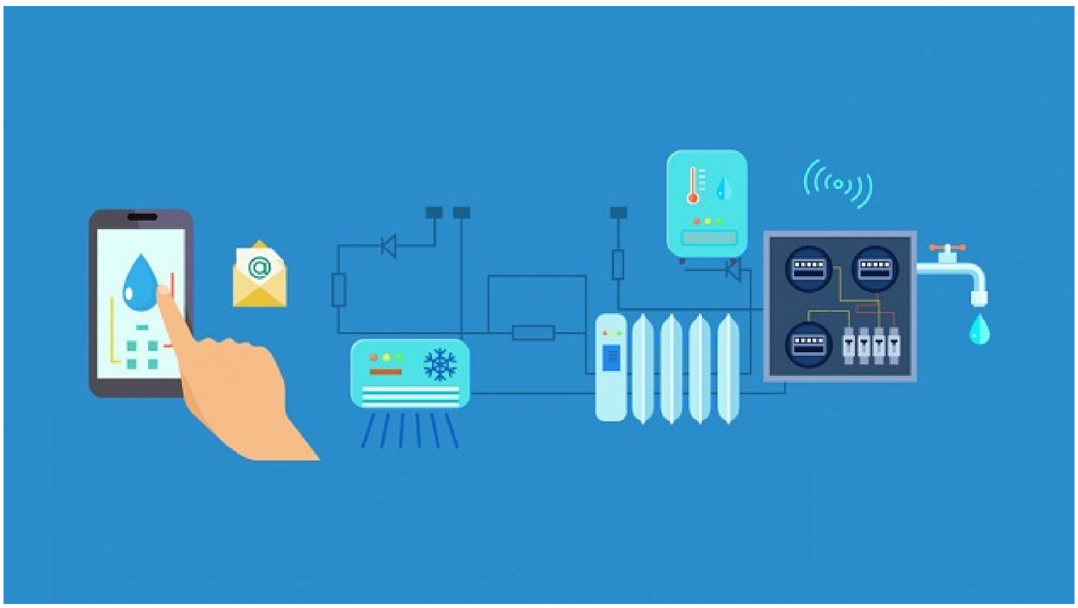
SMART WATER MANAGEMENT

PHASE 5 SUBMISSION

Project Objective:

To implement a comprehensive smart water management system in parks andgardens with the aim of optimizing water consumption, reducing waste, and promoting sustainability. This system will employ sensor technology, data analytics, and efficient irrigation practices to ensure the responsible use of water resources while maintaining the health and beauty of green spaces.



Iot Sensor setup:

A typical IoT sensor setup for smart water management in public places involves a network of sensors and devices that monitor water consumption. These setups usually consist of:

Water Flow Sensors:

These are installed in water pipes to measure the flow and volume of water being used. They provide data on water consumption patterns.

Smart Meters:

These devices monitor water usage, often providing real-time data on consumption. They can be connected to a network to relay information for analysis.

IoT Gateway:

Acts as a bridge between the sensors/devices and the central system, enabling data transmission and reception. It aggregates data from various sensors for processing.

Cloud-based Software:

Data from the sensors is transmitted to cloud-based servers for storage and analysis. This software interprets the data, identifies trends, and offers insights for efficient water management.

Alert Systems:

Automated alerts or notifications can be set up to notify authorities in case of leaks, unusual water usage, or other anomalies.

Integration with Control Systems:

The setup might integrate with water control systems to regulate water flow based on demand or requirements, optimizing water usage.

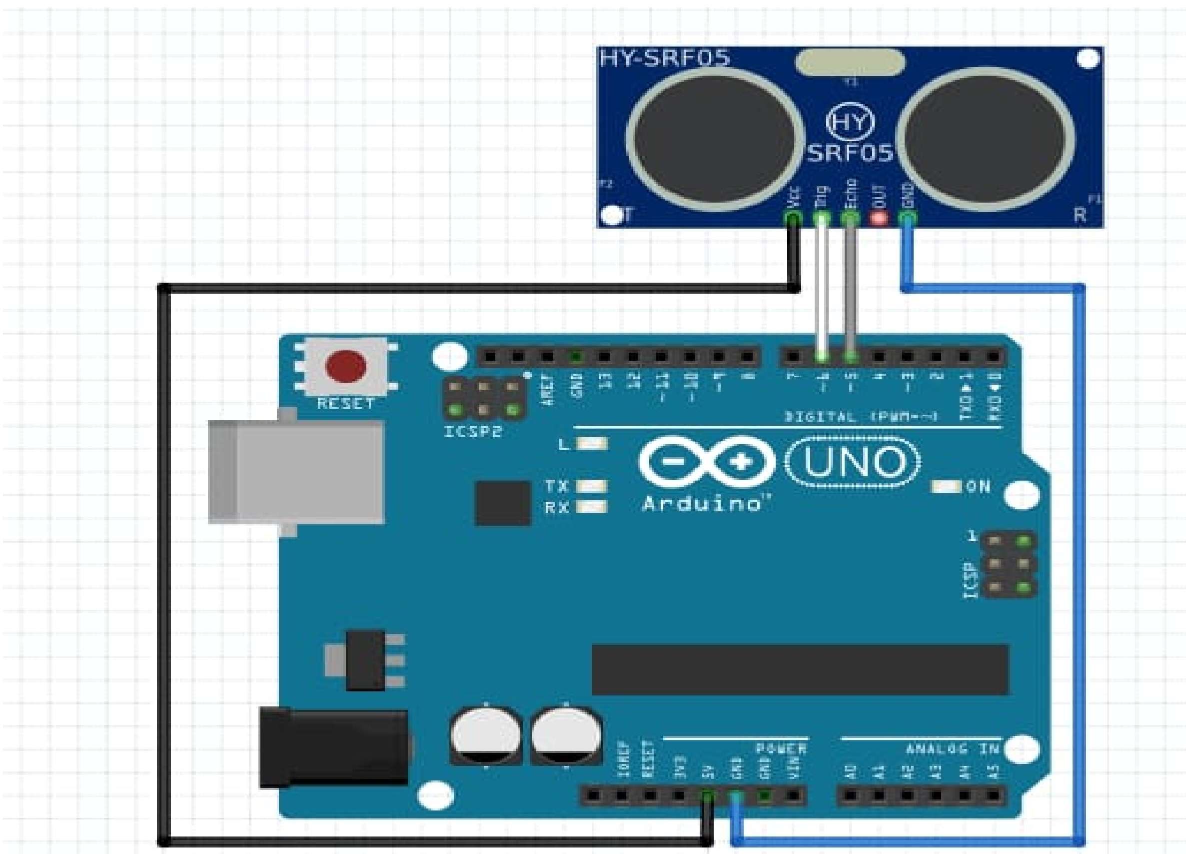
Mobile/Web Interface:

Users or authorities can access the data, reports, and control systems through an interface accessible via web or mobile applications. Overall, these components work together to collect, analyze, and provide actionable insights for effective water management in public places, aiming to reduce waste and ensure efficient usage.

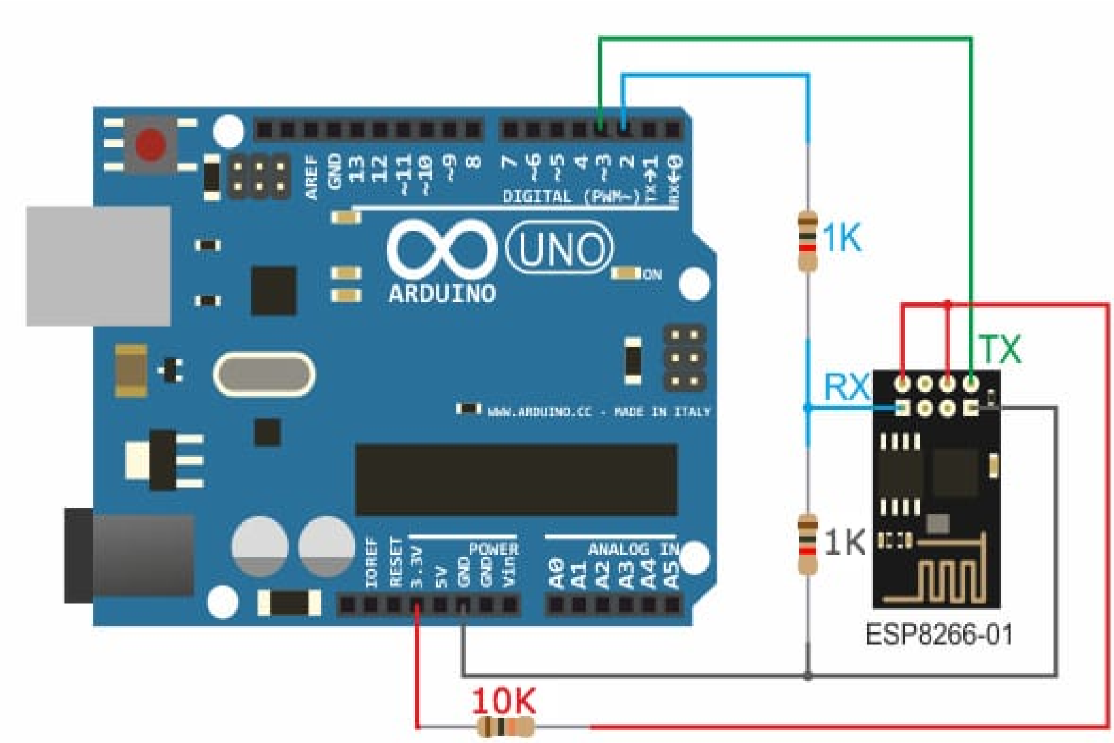
Sensor using smart water management:

This project uses the Arduino UNO is an open-source microcontroller to connect and coordinate the different sensors. Arduino UNO is an open source microcontroller developed by the Arduino group. This board has 14 I/O pins to connect different sensor and transmitter to collect the data and transmit the data to the cloud. It is programmed using the Arduino IDE and a simple USB cable.This board is powered by a 9volt power supply. Different sensors are used to measure the water conditions and levels for data collection. The microcontroller, sensors and transmitters used are:

- Arduino UNO
- SRF-05 sensor (ultrasonic sensor)
- YF-S201 Hall effect sensor (Water flow sensor)
- LM393 chip-based sensor (Rain sensor)
- ESP-8266 Wi-Fi transmitter
- Stepper motor



SRF-05 ultrasonic sensor with Arduino UNO



ESP-8266 Wi-Fi transmitter connected to Arduino

Rashberry pi intergration:

Integrating Raspberry Pi into a smart water management system can involve various components, including sensors, data processing, and potentially controlling water-related devices. Here's an overview of the steps involved

Hardware setup:

Raspberry Pi: Start by setting up your Raspberry Pi board.

Sensors: Connect water flow sensors, water level sensors, or other relevant sensors to the GPIO pins of the Raspberry Pi to monitor water-related data.

Programming the Raspberry Pi:

Use Python or another suitable language to interact with the sensors and collect data.Utilize libraries or GPIO libraries specific to your sensors to read data from them.

Data Processing:

Analyze the data collected from the sensors on the Raspberry Pi. Process the data for specific insights or triggers, such as detecting leaks,monitoring water usage, or generating alerts based on certain thresholds.

Communication and Data Transmission:

Establish communication protocols for transmitting the data to the central system or cloud service, such as MQTT, HTTP, or other APIs. Send the processed data to your backend system or cloud for further analysis or storage.

Mobile App Integration:

Develop a mobile app (as previously outlined) that communicates with your backend system or cloud service to retrieve and display the water-related data.

Implement features in the app to visualize water usage, set alerts, or control waterrelated devices.

Security Measures:

Ensure proper security measures are in place for data transmission and access to the system, especially when controlling water-related devices.

This process involves hardware setup, programming the Raspberry Pi, data processing, communication protocols, and app integration. The code for this system would be quite extensive and would vary depending on the specific sensors used, the backend setup, and the mobile app's functionality.

Mobile app development:

Creating an entire mobile app for smart water management involves several steps and depends on the specific functionalities you want to include. Here's a simplified example of what such an app might look like using Flutter (Dart) for cross-platform development:

dart

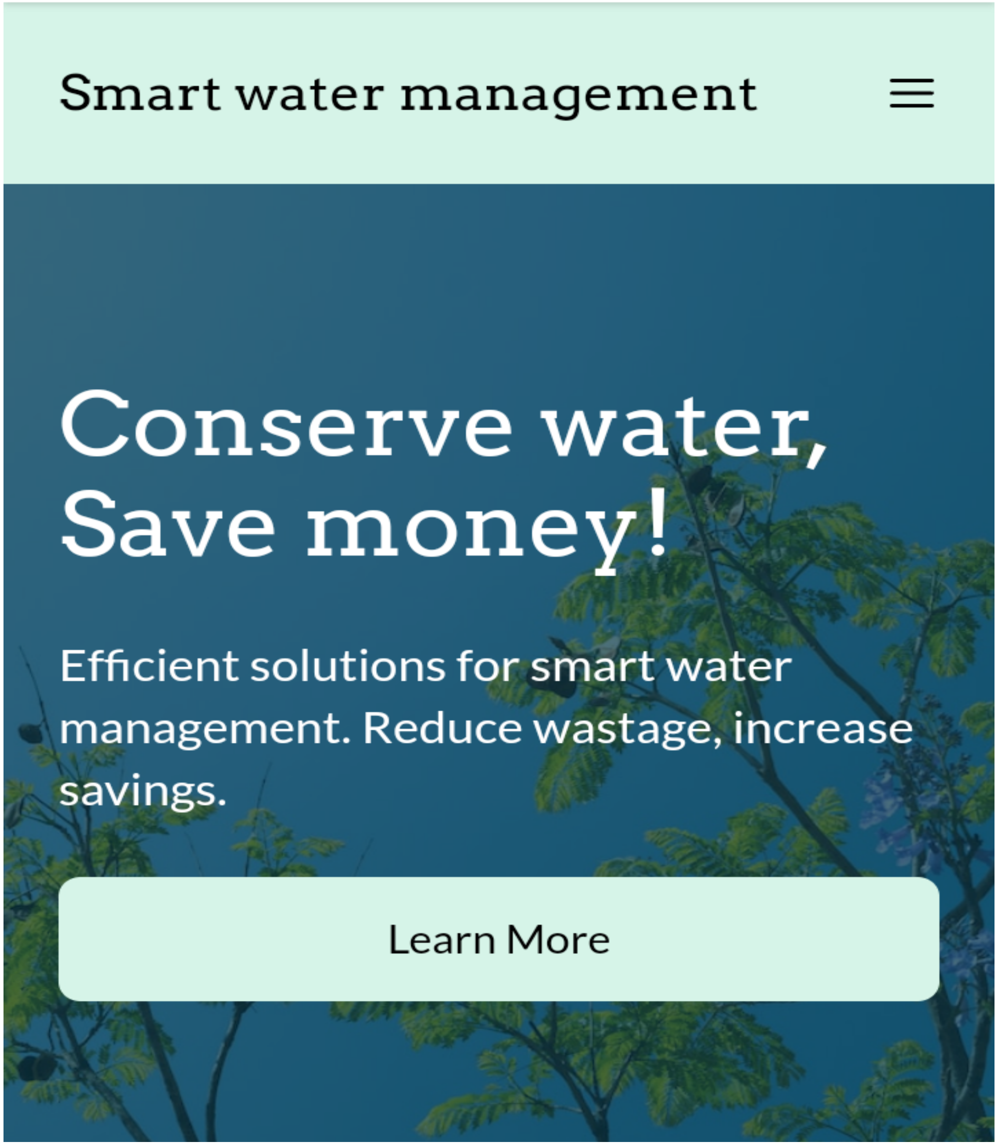
```
import 'package:flutter/material.dart';

void main() {
```

```
  runAppp(SmartWaterApp());
}
class SmartWaterApp extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return MaterialApp(
      title: 'Smart Water Management',
      theme: ThemeData(
        primarySwatch: Colors.blue,
      ),
      home: HomePage(),
    );
  }
}
class HomePage extends StatelessWidget {
  @override
  Widget build(BuildContext context) {
    return Scaffold(
      appBar: AppBar(
        title: Text('Smart Water Management'),
      ),
      body: Center(
        child: Column(
          mainAxisAlignment: MainAxisAlignment.center,
```

```
      children: <Widget>[
        ElevatedButton(
          onPressed: () {
            // Functionality to display water usage data
            // Implement logic to show water consumption statistics
          },
          child: Text('View Water Usage'),
        ),
        ElevatedButton(
          onPressed: () {
            // Functionality to control smart water devices
            // Implement controls for smart water systems (e.g., valves, irrigation systems)
          },
          child: Text('Control Water Systems'),
        ),
        // Other functionalities and components can be added as required
      ],
    ),
  ),
 );
 }
}
```

This is a basic structure and does not include the integration with actual hardware or IoT devices used for water management. For a real application, you'd need to incorporate data collection from sensors, potentially use APIs to communicate with hardware, and ensure security measures, among other factors.

A real smart water management app might involve integrating with IoT devices or sensors, implementing algorithms to analyze water usage, setting up user profiles, incorporating a dashboard with usage statistics, and potentially leveraging machine learning for predictive analysis, among other functionalities. This requires a more detailed, intricate, and device-specific development process.



Recent work

Water Conservation Solution

Real-time Water Monitoring

Developed a smart water management platform that allows users to monitor water consumption, detect leaks, and receive alerts in real-time.

Web platform using HTML, CSS, and JavaScript to display water consumption data for a garden or park:

```
<!DOCTYPE html>

<html>

<head>

<title>Water Consumption Data</title>

<style>

body {

font-family: Arial, sans-serif;

Color:blue;

}

.container {

max-width: 800px;

margin: 0 auto;

}

h1 {

text-align: center;

}

#data {

text-align: center;

}

</style>

</head>

<body>
```

```
<div class="container">

<h1>Water Consumption Data for Garden or Park</h1>

<div id="data">

<p>Current Water Usage: <span id="waterUsage">0 gallons</span></p>

<p>Last Update: <span id="lastUpdate">N/A</span></p>

</div>

</div>

<script>

// Simulated data - Replace this with real-time IoT data

let currentWaterUsage = 250; // in gallons

let lastUpdate = new Date().toLocaleString();

// Update the displayed data

function updateData() {

document.getElementById('waterUsage').textContent = currentWaterUsage + '

gallons';

document.getElementById('lastUpdate').textContent = lastUpdate;

}

// Call updateData initially and set an interval for updates

updateData();

setInterval(updateData, 60000); // Update data every minute (adjust as needed)

</script>

</body>

</html>
```

This code creates a simple web page that displays current water consumption data for a garden or park.

A real-time water consumption monitoring system can significantly promote water conservation and sustainable practices through several key mechanisms:

Awareness and Education:

Real-Time Data Display:Providing users with immediate access to their water consumption allows them to understand their usage patterns and make informed decisions about their water habits.

Visual Representation:

Visualizing water consumption in real time via graphs or analytics in apps or dashboards helps users comprehend their usage trends, facilitating more conscious water usage.

Behavior Modification:

Immediate Feedback: Users can adjust their behavior in real-time based on the data, fostering immediate changes in usage.

Setting Targets:

Establishing consumption goals helps users strive for efficientuse and encourages responsible water practices.

Leak Detection and Alerts:

Early Detection:Real-time monitoring systems can quickly detect leaks or unusual consumption, prompting users to take immediate action to rectify problems.

Alert Mechanisms:

Sending alerts to users when abnormal usage patterns are detected can prevent wastage and potential damage caused by leaks.

Resource Management:

Systematic Planning: For municipalities or water management authorities, realtime monitoring provides insights into peak usage times, allowing for better resource allocation and infrastructure planning.

Demand Forecasting: Understanding usage patterns helps in forecasting future demand, allowing authorities to manage resources more effectively.

Community Engagement:

Collaborative Efforts:Sharing community or neighborhood usage data promotes

healthy competition and encourages collective efforts towards conservation.

Policy and Decision Making: Real-time data can influence policy decisions and initiatives aimed at sustainable water usage and conservation practices at larger scales.

Sustainable Development:

Reduced Water Waste: By actively engaging users and authorities in conscious

usage practices, the system helps reduce water waste and enhances sustainability.

Long-Term Impact: Promoting responsible water usage leads to the conservation

of this vital resource for future generations, contributing to a more sustainable environment.

In summary, a real-time water consumption monitoring system empowers individuals, communities, and authorities to actively manage and conserve water resources. By fostering awareness, facilitating behavioral changes, aiding in early detection, optimizing resource management, and fostering community engagement, such systems play a crucial role in promoting sustainable water practices and ensuring a more responsible use of this invaluable resource.