|  |
| --- |
| Difference Between NAAC & NBA Accreditation - Haq Se EngineerPREC LONIJai Shriram Engineering College (@JSREC09) / Twitter**JAI SHRIRAM ENGINEERING COLLEGE**  **TIRUPPUR – 638 660**  Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai  Recognized by UGC & Accredited by NAACandNBA (CSE and ECE) |

**DEPARTMENT OF**

**ELECTRONICS AND COMMUNICATION ENGINEERING**

**IBM - Naan Mudhalvan**

**Internet of Things**

**Group 3**

**Phase 3 – Development part-1**

**NAME : J.ARUN**

**NM ID :AU711221106004**

**YEAR : III**

**DEVELOPMENT OF SMART WATER SYSTEM:**

* **Raspberry Pi:** Any model of the Raspberry Pi should work, but a Raspberry Pi 4 or later is recommended for better performance.
* **Power Supply:** Ensure you have a stable power supply for your Raspberry Pi.
* **MicroSD Card:** To run the Raspberry Pi’s operating system.
* **Sensors**:
* **Water Flow Sensor:** Measures the flow of water in pipes.
* **Water Level Sensor**: Monitors the water level in tanks or reservoirs.
* **pH Sensor**: Measures the acidity or alkalinity of water.
* **Turbidity Sensor:** Determines the cloudiness of water caused by particles.
* **Temperature and Humidity Sensor:** Monitors ambient conditions.
* **Relays or Solenoid Valves**: Control water pumps, valves, or other devices based on sensor readings.
* **Pump or Motor**: To control the flow of water.
* **Valves**: To control the direction of water flow.
* **GSM or Wi-Fi Module**: For data transmission and remote monitoring.
* **Power Management:** You may need batteries or a UPS for uninterrupted power supply.
* **Waterproof Enclosures:** Protect the Raspberry Pi and other components from the elements if your system is outdoors.
* **Display (optional)**: An LCD or LED display to show real-time information.
* **Case and Mounting Hardware:** To secure and protect the components.
* **Software:** You’ll need to write software to interface with the sensors, collect data, and make decisions. Python is a popular choice for programming Raspberry Pi.
* **Database**: A database to store historical data.
* **Web Interface:** Create a web-based dashboard for real-time monitoring and control.
* **Alert System:** Implement alerts through email, SMS, or notifications for abnormal water conditions.
* **Powerful Power Strip:** To control and manage power to various components.
* **Wires, Cables, and Connectors:** Ensure you have the necessary wiring to connect everything.
* **Backup System**: Implement data backup and recovery methods to prevent data loss.
* **Weatherproofing materials:** If your system is outdoors, make sure it’s well-sealed against the elements.

RASPBERRY PI

**Block diagram:**

**LDR**

**PH SENSOR**

**Thingspeak**

**FLOW SENSOR**

**TEMPARATURE SENSOR**

**CONDUCTIVITY SENSOR**

**Code for smart water system:**

Import random

Class WaterQualitySensor:

    Def \_\_init\_\_(self, location):

        Self. Location = location

    Def measure\_water\_quality (self):

        Return round (random.uniform (6.0, 9.0), 2)

Class AlertSystem:

    Def send alert (self, location, parameter, value):

        Print (f"Alert: {location} - {parameter} value is {value}")

Def main ():

    Sensors = [

        WaterQualitySensor ("River A"),

        WaterQualitySensor ("River B"),

    ]

    AlertSystem = AlertSystem()

    While True:

        For sensor in sensors:

            Water quality = sensor. Measure\_water\_quality()

            print (f"{sensor. Location} - Water Quality: {water quality} (pH)")

            if water quality < 7.0:

                alert\_system.send\_alert(sensor. Location, "Low pH", water quality)

            Elif water quality > 8.5:

                alert\_system.send\_alert (sensor. Location, "High pH", water quality)