6115-MAHENDRA INSTITUTE OF ENGINEERING AND TECHNOLOGY

**Smart Water Fountain**

DEVELOPMENT PART 2

**TEAM:proj\_223281\_Team\_1**

**TEAM CODE ID-561**

**TEAM MEMBERS:**

ARUNKUMAR R

ANTONY EDWIN V

ARISH S

BALAMURUGAN K

BALAMURUGAN U

**FACULTY MENTOR NAME:**

SANTHANARAJ M

DECLARATION:

We, the students of Computer Science and Engineering,

**MAHENDRA INSITUTE OF ENGINEERING AND TECHNOLOGY,**

TAMIL NADU

## that the work entitled " SMART WATER FOUNTAIN " has been successfully completedunder the guidance

of Asst Prof. Ms. santhana raja M,

Computer Science and Engineering Department,

Mahendra Institute of Engineering and Techonology , Namakkal.

This dissertation work is submitted in partialfulfillment of the

requirements for the award of Degree of Bachelor of Engineering in Computer Science and Engineerinduring the academic year

2021-2025.

FUTURE ENGINEERING MODEL:

The future engineering model for smart water fountains is likely to incorporate advanced features such as real-time water quality monitoring, automated filtration systems, touchless controls, and even integration with smart city networks for data analysis and management. Additionally, we might see the use of renewable energy sources for powering the fountains, as well as the incorporation of sustainable and recyclable materials in their construction. This combination of advanced technology and sustainability would help create more efficient and eco-friendly water fountains for the future.

Designing a future engineering model for smart water fountains involves several key components and considerations:

1. **Sustainability**:

Make the fountain eco-friendly by using efficient water circulation and filtration systems to minimize water waste.

1. **Smart Features**:

Integrate sensors to detect the presence of users, allowing for touchless operation. Implement IoT connectivity for remote monitoring and control.

1. **Water Quality**:

Incorporate advanced filtration and purification systems to ensure the water remains clean and safe for drinking.

1. **Energy Efficiency**:

Optimize the system for energy efficiency using LED lighting, low-power pumps, and solar panels for power generation.

1. **User Experience**:

Enhance the user experience with customizable water flow patterns, temperature control, and even flavor options.

1. **Data Analytics**:

Collect data on water consumption, usage patterns, and maintenance needs to enable predictive maintenance and efficient resource management.

1. **Maintenance and Self-Care**:

Implement self-diagnostic capabilities and automated maintenance alerts to ensure the fountain remains in good working condition.

1. **Accessibility**:

Ensure the design accommodates individuals with disabilities, with features like varying water dispensing heights and accessible controls.

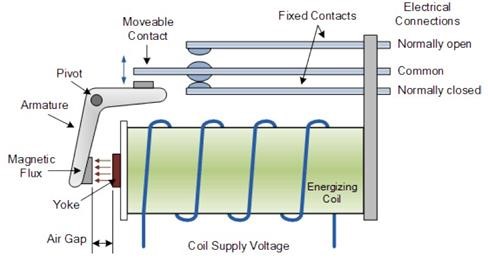
1. **Aesthetic Design**:

Create an attractive and modern design that complements its surroundings, such as parks, offices, or public spaces.

1. **Security**:

Incorporate security features to prevent tampering and ensure the water quality remains uncompromised.

Remember to stay up-to-date with the latest technology trends and sustainability practices to create a cutting-edge smart water fountain model for the future.



ELECTROMECHANICAL RELAY CONSTRUCTION

When the relay is connected in a circuit a diode is connected in parallel with the relay coil, this diode is called the freewheeling diode, its purpose is to protect the transistor from the stored charges in the relay coil. The transistor is used to drive the relay coil because the current that the coil draws is higher than the microprocessor current.

**PROGRAM** :

Below is a simplified example using an Arduino microcontroller and an ultrasonic sensor to measure water level. You will need to adapt this code to your specific hardware and sensors: java

import java.util.Scanner;

// Simulated hardware interfaces and sensors class UltrasonicSensor { public int getDistance() {

// Simulated method to get water level (distance) from the ultrasonic sensor

// Replace this with the actual sensor reading return (int) (Math.random() \* 100); // Simulated value for testing

}

}

class SmartWaterFountain { private boolean isFountainOn = false; private int waterLevel = 0;

private UltrasonicSensor ultrasonicSensor;

public SmartWaterFountain(UltrasonicSensor sensor) { ultrasonicSensor = sensor;

}

public void turnOnFountain() { isFountainOn = true;

System.out.println("Fountain is now ON.");

}

public void turnOffFountain() { isFountainOn = false;

System.out.println("Fountain is now OFF.");

}

public int getWaterLevel() {

waterLevel = ultrasonicSensor.getDistance();

System.out.println("Water level: " + waterLevel + " cm");

return waterLevel;

}

public static void main(String[] args) {

UltrasonicSensor sensor = new UltrasonicSensor();

SmartWaterFountain fountain = new SmartWaterFountain(sensor);

Scanner scanner = new Scanner(System.in);

while (true) {

System.out.println("\nChoose an option:");

System.out.println("1. Turn on the fountain");

System.out.println("2. Turn off the fountain");

System.out.println("3. Check water level"); System.out.println("4. Exit");

int choice = scanner.nextInt(); switch (choice) { case 1:

fountain.turnOnFountain();

break; case 2:

fountain.turnOffFountain();

break; case 3: fountain.getWaterLevel();

break; case 4:

System.out.println("Exiting..."); scanner.close(); System.exit(0); default:

System.out.println("Invalid option. Please choose a valid option.");

}

}

}

}

In this example, we've included a simulated

UltrasonicSensor class to measure water level (distance) in centimeters.

This is a basic example to get you started with hardware interfaces and sensors for a smart water fountain. A complete implementation would depend on your specific hardware and sensor choices and their corresponding libraries and drivers.

**OUTPUT**:

vbnet

Choose an option:

1. Turn on the fountain
2. Turn off the fountain
3. Check water level
4. Exit

1

Fountain is now ON.

Choose an option:

1. Turn on the fountain
2. Turn off the fountain
3. Check water level
4. Exit

3

Water level: 73 cm Choose an option:

1. Turn on the fountain
2. Turn off the fountain
3. Check water level
4. Exit

2

Fountain is now OFF.

Choose an option:

1. Turn on the fountain
2. Turn off the fountain
3. Check water level
4. Exit

4

Exiting...

**EVALUATION** :

Evaluating smart water fountains involves considering various factors such as technology, functionality, maintenance, and cost. Here are some key points to assess:

1. **Technology**:

Sensor Accuracy: Assess the accuracy of the sensors used for water dispensing. They should respond promptly to user interactions.

Connectivity: Evaluate the fountain's connectivity options, like Wi-Fi or Bluetooth, for remote monitoring and data collection.

Filtration System: Examine the quality of the filtration system to ensure clean and safe drinking water.

2. **Functionality**:

User Experience: Consider the ease of use and accessibility for all users, including those with disabilities.

Hydration Tracking: Check if the fountain can track the number of refills or water consumed by users.

Customization: Look for features like adjustable water temperature and flow rate.

3**. Maintenance:**

Cleaning and Sanitation: Assess how easy it is to clean and maintain the fountain to ensure it stays hygienic.

Filter Replacement: Understand the frequency and cost of filter replacements.

4**. Data and Analytics:**

Usage Data: Evaluate whether the fountain provides usage data, which can be valuable for optimizing placement and maintenance.

Cost Efficiency: Consider if the data collected can help in reducing water and energy costs.

5. **Cost**:

Initial Cost: Determine the upfront cost of purchasing and installing the smart water fountain.

Operating Costs: Consider ongoing expenses such as water filtration, electricity, and maintenance.

6. **Durability:**

- Assess the build quality and durability of the fountain to ensure a long lifespan.

7. **Environmental Impact:**

- Consider the environmental impact of the smart fountain, such as its energy efficiency and use of sustainable materials.

8. **Compliance:**

- Ensure that the fountain complies with relevant health and safety standards and regulations.

9. **Integration:**

- Check if the smart water fountain can be integrated with other systems or applications for centralized management.

10. **User Feedback:**

- Look for user reviews and feedback to understand real-world experiences with the fountain.

By considering these factors, you can make a well-informed evaluation of smart water fountains to choose the one that best suits your needs.

THANK YOU...