Automated Plant Watering System

Hana Ahmed, Mariam Mohamed, Ahmed Ehab, Ahmed Hamouda
Supervised by: Dr.Walaa Hassan Abdelhamid, Eng.Reem Aboushama

May 2024

1 Abstract

This project proposes the development of an automated watering system utilizing Arduino microcontroller technology to ensure optimal growth conditions for plants. The system integrates various sensors, including soil moisture, humidity and temperature, and rainfall, to gather real-time environmental data. Upon detecting soil moisture level above or equal a predefined threshold, the system activates a water pump to irrigate the plants, ensuring they receive adequate hydration. Additionally, a physical rain shield, controlled by a servo motor, is implemented to protect the plants during rainfall and prevent overwatering. An alert mechanism, utilizing a buzzer, notifies users when there is rainfall, blue Led alert user there is rain, green Led notify user plant is watered and the soil is medium or wet, red Led notify user plant soil is dry and plant needs watering. Through the intelligent integration of these components and control logic, the automated watering system offers an efficient and convenient solution for plant care, suitable for both indoor and outdoor environments.

2 Introduction

2.1 Problem Statement

Traditional plant care methods often rely on manual intervention and periodic monitoring, which can be time-consuming and prone to human error. Inefficient watering practices, such as overwatering or underwatering, can lead to poor plant health and growth. Additionally, environmental factors such as soil moisture, humidity and temperature, and rainfall fluctuate unpredictably and require constant monitoring to maintain optimal growth conditions for plants.

The problem this project aims to address is the need for an automated watering system that can effectively monitor environmental conditions and provide timely hydration to plants, both indoors and outdoors. By utilizing sensors and intelligent control logic, the system seeks to optimize plant care by ensuring adequate moisture levels, protecting plants from overwatering during rainfall, and alerting users when attention is required. The goal is to create a convenient and efficient solution for plant care that minimizes manual intervention and maximizes plant health and growth.

We will use Rose plant for the watering system as the Rose plant growing conditions prefer humidity (20 percent to 50 percent) and temperatures (20°C to 35°C). Temperature thresholds should be maintained to prevent stress on the plants.[3]

2.2 Objectives

- Develop an automated watering system using Arduino microcontroller technology.
- Integrate various sensors including soil moisture, humidity and temperature, and rainfall to gather real-time environmental data.
- Implement intelligent control logic to monitor soil moisture levels and respond accordingly.
- Activate a water pump to irrigate plants when soil moisture exceeds a predetermined threshold.
- Control a physical rain shield, powered by a servo motor, to protect plants during rainfall and prevent overwatering.
- Incorporate an alert mechanism utilizing a buzzer and blue led to notify users of the rainfall also using red and green leds for notifying users of whether the plant needs water or not.
- Design the system to be suitable for both indoor and outdoor environments, ensuring optimal plant care under various conditions.
- Ensure the system offers an efficient and convenient solution for plant care, minimizing manual intervention and maximizing plant health and growth.

2.3 Scope Of Project

The scope of this project encompasses the design, development, and implementation of an automated watering system using Arduino uno microcontroller technology. The system will be capable of monitoring soil moisture levels, humidity and temperature, and rainfall in real-time using various sensors.

Key components and features of the project include:

- Integration of soil moisture, humidity and temperature, and rain detection sensors to gather environmental data.
- Development of intelligent control logic to analyze sensor data and make prediction decisions based on predefined thresholds.

- Activation of a water pump to irrigate plants when soil moisture is equal or exceeds a predetermined threshold, ensuring adequate hydration.
- Control of a physical rain shield, powered by a servo motor, to protect plants during rainfall and prevent overwatering.
- Implementation of an alert mechanism, utilizing a buzzer and blue led, to notify users of rainfall , also using red and green leds for notifying users of whether the plant needs water or not. .
- Design of the system to be suitable for both indoor and outdoor environments, accommodating a variety of plant care needs.
- Testing and validation of the system to ensure it offers an efficient and convenient solution for plant care, minimizing manual intervention and maximizing plant health and growth.

  The project scope includes both hardware and software development, as well as testing and validation to ensure the system meets the desired objectives and requirements.

3 Literature Review

3.1 Overview of existing solutions

Existing solutions for automated watering systems often involve the use of microcontrollers like Arduino or Raspberry Pi, along with various sensors and actuators to monitor and control environmental conditions for plants. Here's an overview of some common features found in existing solutions:

- Sensor Integration: Existing systems typically incorporate sensors such as soil moisture sensors, humidity sensors, temperature sensors, and rain sensors to gather real-time environmental data.
- Control Logic: Intelligent control algorithms are used to analyze sensor data and make decisions based on predefined thresholds. This includes determining when to activate the water pump based on soil moisture levels and controlling the rain shield during rainfall events.
- Actuation Mechanisms: Actuators like water pumps and motors are utilized to perform actions such as watering the plants and controlling the rain shield. These actuators are often controlled by the microcontroller based on the analysis of sensor data.
- Alert Mechanisms: Many systems include alert mechanisms to notify users when certain conditions are met, such as low soil moisture levels requiring attention. Buzzer alerts or notifications sent to mobile devices are common methods used for this purpose.

- Indoor and Outdoor Compatibility: Solutions are designed to be suitable for both indoor and outdoor environments, providing flexibility for users to maintain plants in various settings.
- Efficiency and Convenience: The goal of these systems is to offer an efficient and convenient solution for plant care, minimizing manual intervention and maximizing plant health and growth.
- Customization and Scalability: Some solutions offer customization options, allowing users to adjust parameters such as watering schedules and threshold levels. Additionally, scalability features enable users to expand the system to accommodate larger plant setups if needed.

  Overall, existing solutions for automated watering systems aim to provide users with a reliable and effective way to maintain optimal growth conditions for their plants while reducing the time and effort required for plant care.

3.2 Relevance to current project

- The relevance of the current project lies in addressing the need for efficient and convenient plant care solutions in both indoor and outdoor environments. By creating an automated watering system that monitors soil moisture levels, humidity and temperature, and rainfall in real-time, the project aims to ensure optimal growth conditions for plants while minimizing manual intervention.[4]
- The incorporation of various sensors such as soil moisture, humidity and temperature, and rain detection allows the system to gather accurate environmental data, enabling precise control of watering and protection mechanisms. When soil moisture falls below a predetermined threshold, the water pump is activated to provide adequate hydration to the plants, promoting healthy growth. Additionally, the physical rain shield, controlled by a motor, prevents overwatering during rainfall events, further optimizing plant care.[1]
- The alert mechanism, utilizing a buzzer and leds.

  Through the integration of these components and intelligent control logic, the automated watering system offers an efficient and convenient solution for plant care, catering to the needs of both indoor and outdoor plant environments.[5]

4 Proposed System Overview

4.1 Detailed description of the proposed solution

The proposed solution is an automated watering system designed to ensure optimal growth conditions for plants by monitoring key environmental parameters including soil moisture levels, humidity and temperature, and rainfall. The system utilizes a microcontroller such as Arduino to integrate various sensors and actuators, providing real-time data collection and intelligent control capabilities. Below is a detailed description of the components and functionality of the proposed solution:

1. Microcontroller (Arduino):
    - The central component of the system, responsible for controlling the operation of sensors, actuators, and implementing intelligent control logic.
    - Executes programmed instructions to collect sensor data, analyze environmental conditions, and trigger appropriate actions.
2. Sensors:
    - Soil Moisture Sensor: Monitors soil moisture levels to determine plant hydration status.
    - Humidity and Temperature Sensor: Measures ambient humidity levels to assess environmental conditions and kept within optimal temperature ranges.
    - Rain Detection Sensor: Detects rainfall events to prevent overwatering and protect plants during rainy conditions.
3. Actuators:
    - Water Pump: Activated when soil moisture falls below a predetermined threshold, ensuring plants receive adequate hydration.
    - Servo motor: Adjusts to protect plants during rainfall, preventing overwatering and potential damage.
4. Alert Mechanism:
    - Buzzer: Notify user there is rain.
    - Blue Led: Alert user there is rain.
    - Green Led: Notify user plant is watered and the soil is medium or wet.
    - Red Led: Notify user plant soil is dry and plant needs watering.
5. Intelligent Control Logic:
    - Utilizes programmed algorithms to analyze sensor data and make decisions based on predefined thresholds and rules.
    - Determines when to activate the water pump based on soil moisture levels,ensuring plants receive adequate hydration without overwatering.

- The status of water pump is ON or OFF based on OR Logic between threshold and model prediction.
- Controls the servo motor to open or close the shield during rainfall events, preventing excessive watering and protecting plants.

6. User Interface:

   - May include a user interface such as an LCD display or Web application for monitoring Pump status, and scheduling irrigation time.

7. Efficiency and Convenience:

   - Offers an efficient and convenient solution for plant care, minimizing manual intervention and optimizing plant health and growth.
   - Suitable for both indoor and outdoor environments, providing flexibility and versatility in plant care applications.

By integrating these components and intelligent control logic, the proposed automated watering system aims to provide an effective and user-friendly solution for maintaining optimal growth conditions for plants, promoting healthy and thriving vegetation in various settings.[9]

4.2 System Architecture Diagram

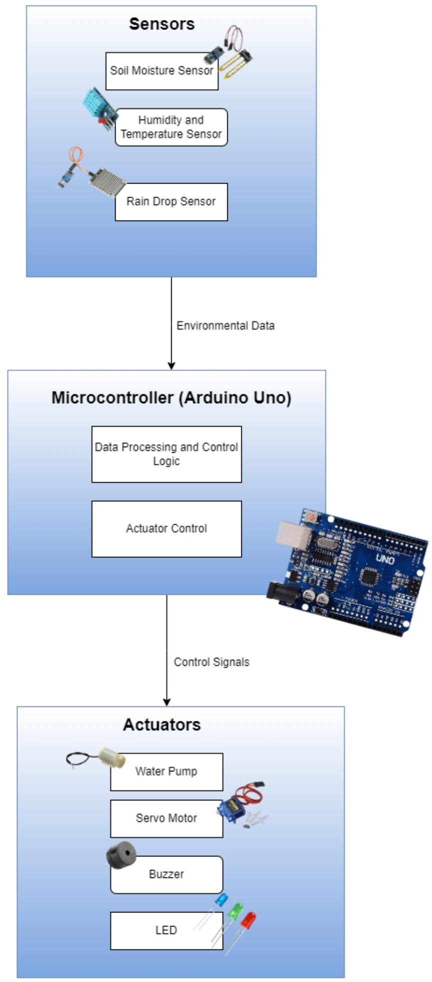

Figure 1: System Architecture Diagram

4.2.1 Software Components

- C programming for arduino
- Python and Machine Learning code for model pump status prediction
- python flask web Application for scheduling each time the plant is watered Being tracked and kept record of.
- ROS (Robot Operating System): For integrating and managing communication between the Arduino and the ML model.

5 Design Details

5.1 System Design

The diagram outlines an automated watering system for plants. It comprises sensors for monitoring soil moisture, humidity and temperature, and rainfall. Data from these sensors is processed by a microcontroller, which applies control logic to activate actuators. Actuators include a water pump for irrigation, a rain shield motor for protection during rainfall, and a buzzer and blue led for alerting users of rainfall. This architecture ensures plants receive optimal care by adjusting watering and protection based on real-time environmental data.[7]

5.2 Agent architecture design and type

1. Sensors Interface Module:
   - Responsible for interfacing with various sensors including soil moisture, humidity and temperature, and rain detection.
   - Translates analog/digital signals from sensors into readable data for the microcontroller.
2. Microcontroller (Arduino):
   - Acts as the central processing unit of the system.
   - Receives data from the sensors interface module.
   - Executes control logic algorithms based on the received data.
   - Controls the operation of actuators (water pump, leds , servo motor for rain shield).
   - Interfaces with the sensors and the water pump, and communicates with the ROS system.[2]
3. Control Logic Module:
   - Implements decision-making algorithms based on real-time environmental data.
   - Determines when to activate the water pump based on soil moisture levels thresholds.
   - Controls the motor for the rain shield to protect plants during rainfall.
   - Manages the alert mechanism to notify users when intervention is required.
4. Actuators:
   - Water Pump:
     - Activated when soil moisture is equal or above the threshold.

  - Controlled by the microcontroller to deliver water to plants.
- servo Motor for Rain Shield:
  - Controls the physical rain shield to protect plants during rainfall.
  - Operated by the microcontroller based on rainfall data.

5. Alert Mechanism:
   - Utilizes a buzzer and blue led to notify users of rainfall and green,red led notify users when plant need watering or not.
   - Controlled by the microcontroller based on predefined conditions.

6. ROS (Robot Operating System):
   - For integrating and managing communication between the Arduino and the ML model.

7. Power Supply:
   - Provides power to all components of the system.
   - Can be battery-powered or connected to mains electricity depending on the application (indoor/outdoor).

5.2.1 Type of Agent Architecture

- The proposed architecture exhibits characteristics of a reactive agent architecture.It reacts to changes in the environment (e.g., soil moisture, rainfall) and takes actions accordingly without maintaining an explicit internal state. Also, Control decisions are made based on real-time sensor data, and actions are executed promptly by the microcontroller.

5.3 Environment Properties

The environment properties of the proposed automated watering system project include:

- Indoor and Outdoor Compatibility: The system is designed to be suitable for both indoor and outdoor environments, catering to a wide range of plant care needs. Whether it's a home garden, greenhouse, or outdoor landscape, the system can be deployed effectively to ensure optimal growth conditions for plants.
- Real-Time Monitoring: The system incorporates various sensors to monitor soil moisture levels, humidity and temperature, and rainfall in real-time. This enables the system to continuously assess environmental conditions and respond promptly to changes, ensuring plants receive the necessary care.

- Efficiency: By automating the watering process and implementing intelligent control logic, the system offers an efficient solution for plant care. It minimizes manual intervention and optimizes resource utilization, leading to efficient water usage and improved plant health.
- Convenience: The automated watering system provides a convenient solution for plant care, reducing the need for frequent manual watering and monitoring. Users can rely on the system to maintain optimal growth conditions for their plants with minimal effort.
- Protection Mechanisms: The system includes a physical rain shield controlled by a motor to protect plants during rainfall and prevent overwatering. This ensures that plants are not subjected to excessive moisture levels, which can lead to root rot and other issues.[8]
- Alert System: An alert mechanism,tilizes a buzzer and blue led to notify users of rainfall and green,red led notify users when plant need watering or not.[6]

5.4 UML Diagrams:

5.4.1 Class Diagram

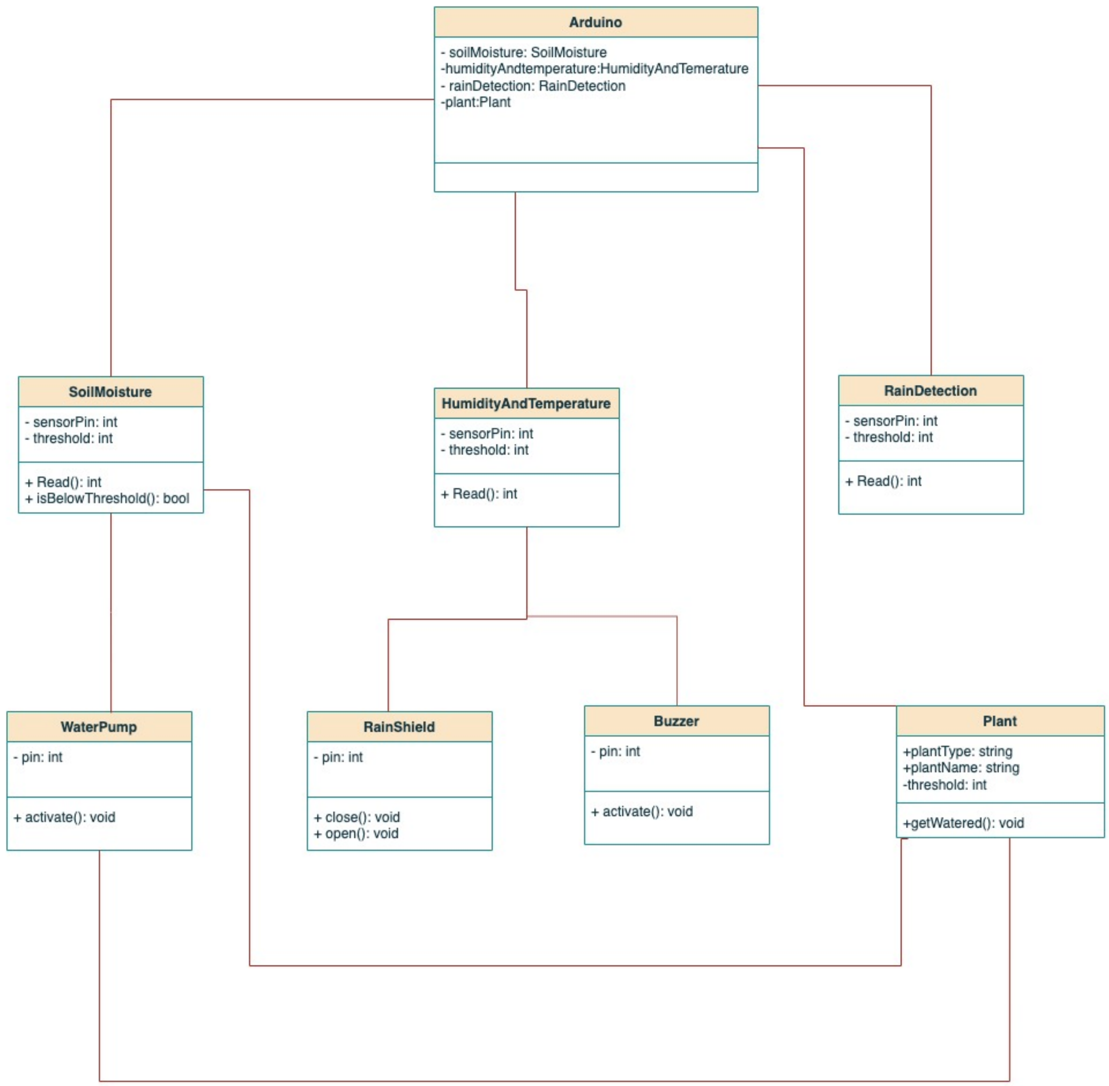

Figure 2: Class Diagram

5.4.2 Activity Diagram

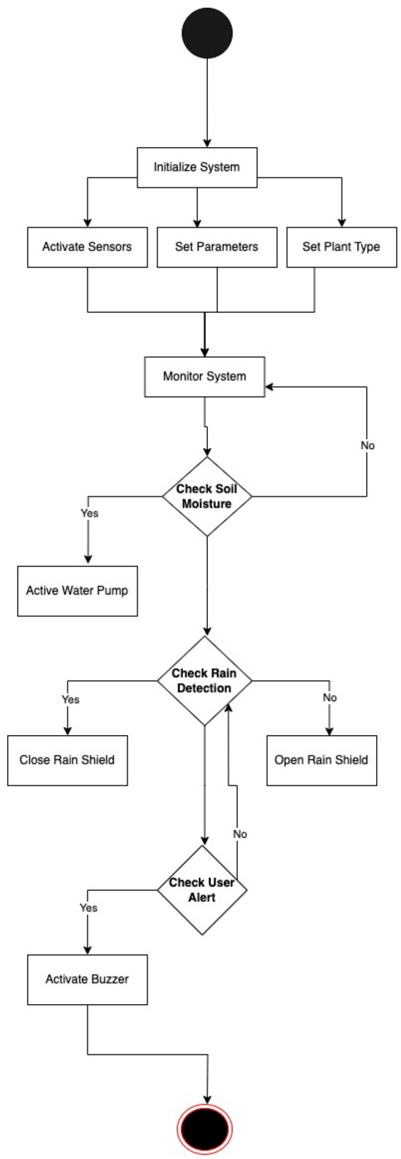

Figure 3: Activity Diagram

5.4.3 Sequence Diagram

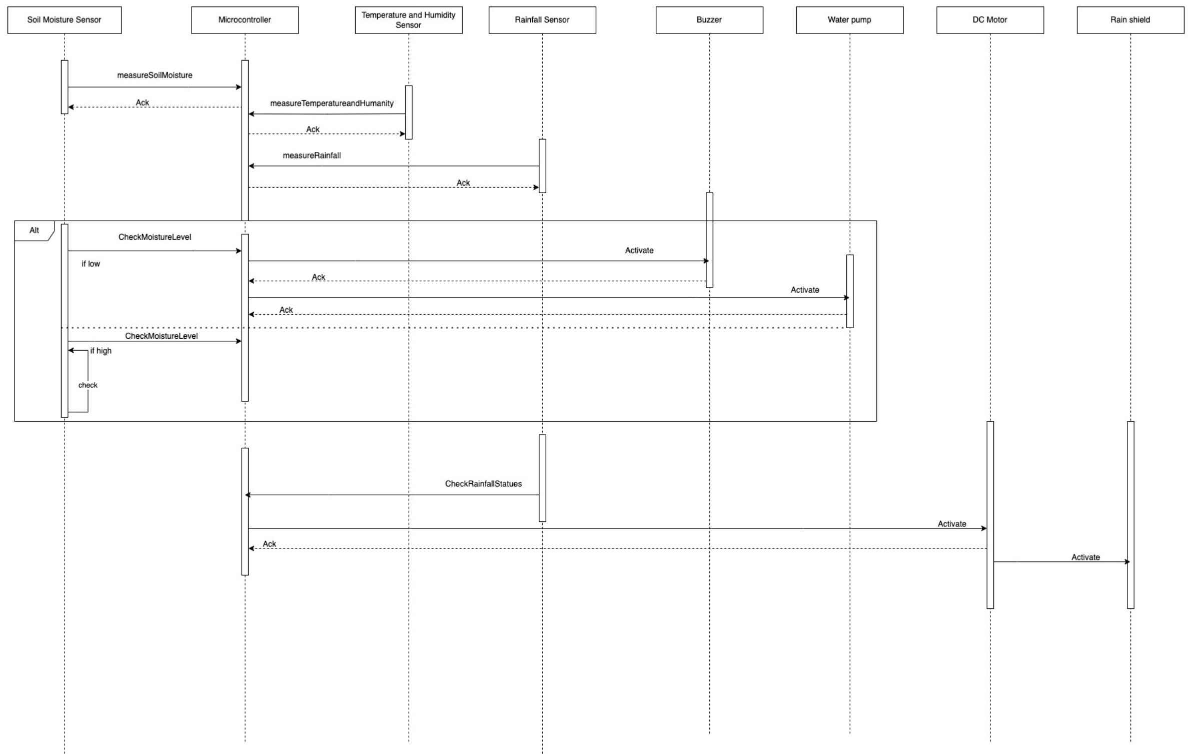

Figure 4: Sequence Diagram

5.4.4 Flow Chart

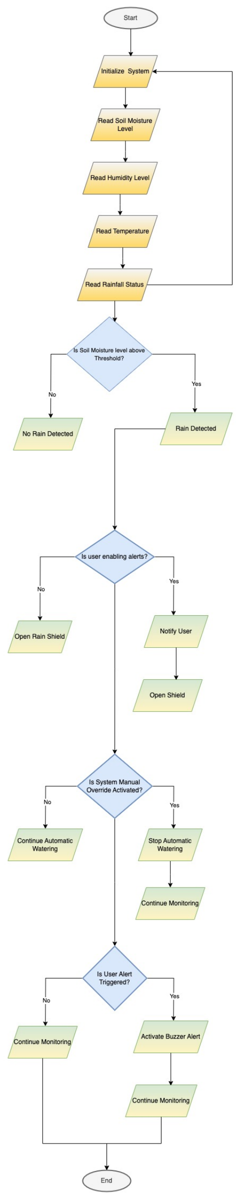

Figure 5: Flow Chart

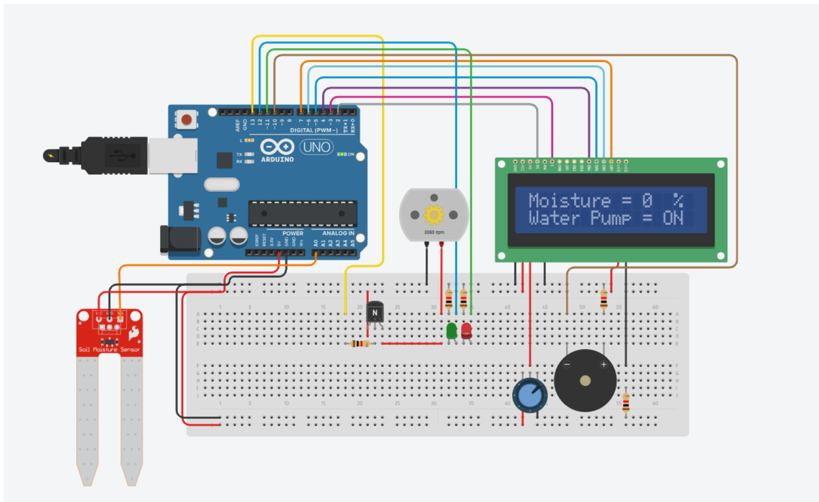

Figure 6:Circuit Diagram Tinkercad

5.4.5 Pseudocode or Algorithm

```
Initialize system
Set SOIL_MOISTURE_THRESHOLD

LOOP FOREVER
  Read soil moisture level
  Read humidity level
  Read temperature
  Check for rainfall

  IF soil moisture level < SOIL_MOISTURE_THRESHOLD THEN
    Activate water pump
    Turn on green LED
    WHILE soil moisture level < SOIL_MOISTURE_THRESHOLD
      Read soil moisture level
    END WHILE
    Deactivate water pump
    Turn off green LED
  ELSE
    Turn on red LED
  END IF

  IF rainfall detected THEN
    Close rain shield
    Turn on buzzer
    Turn on blue LED
    WHILE rainfall detected
      Check for rainfall
    END WHILE
    Open rain shield
    Turn off buzzer
    Turn off blue LED
  END IF

END LOOP
```

Figure 7:Pseudocode

The pseudocode starts by initializing the system and setting the soil moisture threshold. It then enters an infinite loop that reads the soil moisture level, humidity level and temperature, and rainfall status. If the soil moisture level is above or equal the threshold, the water pump is activated until the soil moisture level is below the threshold. If rainfall is detected, the rain shield is closed until the rainfall is no longer detected.the buzzer and blue led to notify users of rainfall , green led notify user when plant is watered and red led notify users when plant need watering.

6

6.1 Hardware Requirements

1. Arduino Uno microcontroller board
2. three sensors:
   - Soil moisture sensor
   - Humidity and Temperature sensor
   - Rainfall sensor
3. Water pump
4. Servo Motor
5. Buzzer and leds for alert mechanism
6. Power supply (batteries)
7. Connecting wires and breadboard

6.2 Software Requirements

1. Arduino Integrated Development Environment (IDE) for writing and uploading code to the Arduino board
2. Arduino libraries for interfacing with sensors and controlling actuators
3. Programming skills in Arduino's C/C++ language to develop the control logic and algorithms for the automated watering system
4. Python and Machine Learning code
5. ROS (Robot Operating System): For integrating and managing communication between the Arduino and the ML model.

6.3 Expected Timeline

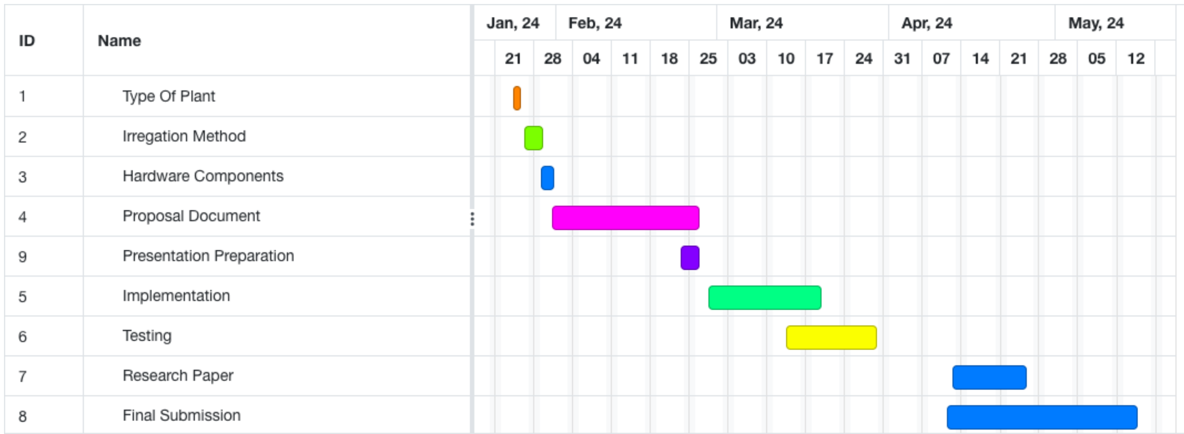

Figure 8: Gantt Chart

7 Budget And Resources

7.1 List of Hardware Components used

- Arduino uno
- Soil moisture sensor
- DHT11 Temperature and Humidity Sensor
- Relay module
- Bread board
- Jumper wires
- rain sensor
- buzzer
- Battery
- Water pump
- Led green, red, and blue
- Servo motor

7.2 Cost Estimation

1. Arduino uno microcontroller board : 939LE
2. Temperature and Humidity sensor Module(DHT11): 80LE
3. Rain Drop sensor: 35LE
4. 5 Relay Module: 275LE
5. Breadboard 840 Pin: 65LE
6. 40 Male-Male Jumper Wire , 40 Female-Female Jumper Wire , 40 Male-Female Jumper wire: 120LE
7. Soil Moisture Sensor: 65LE
8. 5 Buzzer: 50LE
9. 3 Water pump: 210LE
10. 10 Green, 10 red, 10 blue led: 60LE
11. 8 battery: 760LE
12. Total cost:2659LE

7.3 Hardware Constraints

- Microcontroller Platform: The system relies on the Arduino Uno microcontroller, which has limited processing power and memory.
- Sensors: The project utilizes various sensors, including soil moisture sensors, DHT11 temperature and humidity sensors, and rain sensors, which must be compatible with the Arduino Uno.
- Power Supply: The system needs a reliable power source, either through batteries or mains electricity, especially for outdoor use.
- Actuators: Components such as the water pump, servo motor for the rain shield, and alert mechanisms (buzzer and LEDs) need to be controlled accurately and efficiently by the microcontroller.
- Connectivity: The system's components must be properly interconnected using jumper wires and a breadboard, ensuring stable and reliable connections .

7.4 Software Constraints

- Programming Language: The software for the automated watering system must be written in a language supported by the chosen microcontroller platform, such as Arduino's C/C++ language.

- Sensor Data Processing: The software must include algorithms for processing sensor data in real-time and making decisions based on predefined thresholds. This may involve implementing control logic for activating the water pump and controlling the rain shield.

- User Interface: The system may include a user interface for configuring monitoring environmental data. The software must support the development of a user-friendly interface, which may involve graphical user interface (GUI) development or integration with external devices such as smartphones or computers.

- ROS (Robot Operating System): For integrating and managing communication between the Arduino and the ML model..

7.5 Workarounds for Hardware Constraints

- Component Compatibility: Select sensors and actuators that are specifically designed to work with the Arduino platform to ensure compatibility and ease of integration.

- Power Management: Use power-efficient components and consider a dual power supply system (battery and mains) to ensure the system remains operational during power outages.

- Modular Design: Implement a modular design where each component (sensors, actuators, power supply) can be individually replaced or upgraded without affecting the entire system.

- Robust Connections: Use high-quality jumper wires and breadboards to prevent loose connections and signal interference, which are crucial for reliable system performance .

7.6 Workarounds for Software Constraints

- Efficient Coding: Optimize the code for performance to ensure it runs efficiently on the Arduino's limited hardware resources.

- Data Processing: Use lightweight algorithms for real-time sensor data processing to minimize computational load.

- Simplified User Interface: Develop a basic, yet effective, user interface that can be accessed via serial communication or a simple web-based platform.

- Preprocessing Data: Preprocess and filter sensor data before feeding it into the machine learning model to reduce the complexity and size of the data being processed.
- Modular Software Design: Use a modular approach in software development to allow individual components to be updated or replaced without disrupting the entire system .

8 Technicalities

The Automated Plant Watering System leverages sensor data, machine learning algorithms, and the Robot Operating System (ROS) to efficiently manage irrigation. The system's primary goal is to optimize water usage by predicting and deciding when to activate the water pump based on environmental conditions and historical data.

8.1 Implementation Methodolgy

1. Data Collection:
    - Sensor data is collected via the Arduino, which reads from the soil moisture sensor, DHT11 sensor, and rainfall sensor.
    - The data is transmitted to a PC running ROS over a serial connection.
2. Machine Learning Model
    - Data Preparation: Historical sensor data and pump status are used to train the model. The dataset includes soil moisture, temperature, humidity, rainfall, and the corresponding pump status (ON/OFF).
    - Model Training: A decision tree regression model is selected due to its high R2 score , lower MAE, and highest accuracy.The model is trained to predict the pump status based on the sensor inputs.
3. Threshold-Based Logic
    - The system uses predefined thresholds to make decisions: Pump ON: If soil moisture above or equal 940 or temperature above 35°C, and no rainfall. Pump OFF: Otherwise.
    - The threshold-based decision is combined with the model's prediction using an OR logic.
4. Integration with ROS
    - ROS Node for Arduino: Uses rosserial to publish sensor data and subscribe to pump control commands.
    - ROS Node for Decision Making: Receives sensor data, applies the ML model, combines it with threshold logic, and publishes the pump status.

9 Testing and Results

1. Unit Testing
   - Sensor Data Acquisition: Verified the correct functioning of each sensor individually with the Arduino.
   - Data Transmission: Ensured data was correctly transmitted from Arduino to ROS nodes.
2. Integration Testing
   - End-to-End Data Flow: Tested the entire data flow from sensor readings through to ROS, ML model prediction, and pump control.
   - Communication Reliability: Ensured robust communication between the Arduino and ROS nodes without data loss or corruption.
3. Performance Evaluation
   - Model Accuracy: Evaluated the accuracy of models using metrics like Mean Squared Error (MSE), Mean Absolute Error (MAE) and R2 Score.
4. Field Testing
   - Controlled Environment: Conducted tests in a controlled environment to validate the system's responsiveness to different sensor inputs.
   - Real-world Conditions: Deployed the system in a real-world setting to monitor its performance and reliability over time.
5. Results
   - Prediction Accuracy: Achieved high accuracy with the decision tree model, significantly improving over threshold-only control.
   - System Reliability: Demonstrated reliable operation with consistent sensor readings and correct pump activation based on combined ML and threshold logic.
   - Efficiency: Showed improved water usage efficiency by preventing over-watering and under-watering.

10 Conclusion

10.1 Summary

The Automated Plant Watering System successfully integrates sensor data, machine learning, and ROS to create an intelligent irrigation solution. The system's ability to predict and decide on water pump activation based on real-time environmental data and historical trends ensures optimal water usage and plant health.

10.2 Expected Outcomes

- Automated Irrigation: The system will automatically irrigate plants based on soil moisture levels, ensuring optimal hydration without manual intervention.
- Environmental Protection: The physical rain shield will protect plants during rainfall, preventing overwatering and potential damage.
- User Alerts: The alert mechanism (buzzer and LEDs) will notify users of important events, such as the need for watering or the occurrence of rainfall.
- Efficiency and Convenience: The system aims to provide a convenient and efficient solution for plant care, suitable for both indoor and outdoor environments, thus maximizing plant health and growth.
- Real-Time Monitoring: Continuous monitoring and real-time data processing will enable the system to make immediate decisions, ensuring plants receive timely care .

10.3 Future Enhancements

Using Raspberry-pi for the system and doing machine learning model for rain prediction

References

[1] Muhamad Derisa, Edi Mulyana, and Sony Sumaryo. Designing a data logger monitoring system prototype on automatic plantsprinklers. In *2019 IEEE 5th International Conference on Wireless and Telematics (ICWT)*, pages 1–7, 2019.

[2] C. M. Devika, Karthika Bose, and S. Vijayalekshmy. Automatic plant irrigation system using arduino. In *2017 IEEE International Conference on Circuits and Systems (ICCS)*, pages 384–387, 2017.

[3] Drashti Divani, Pallavi Patil, and Sunil K. Punjabi. Automated plant watering system. In *2016 International Conference on Computation of Power, Energy Information and Commuincation (ICCPEIC)*, pages 180–182, 2016.

[4] Manisha Mayuree, Priyanka Aishwarya, and A Bagubali. Automatic plant watering system. In *2019 International Conference on Vision Towards Emerging Trends in Communication and Networking (ViTECoN)*, pages 1–3. IEEE, 2019.

[5] Manisha Mayuree, Priyanka Aishwarya, and A. Bagubali. Automatic plant watering system. In *2019 International Conference on Vision Towards*

Emerging Trends in Communication and Networking (ViTECoN), pages 1–3, 2019.

[6] Yin Yin Nu, SS Lwin, and WW Maw. Automatic plant watering system using arduino uno for university park. *International Journal of Trend in Scientific Research and Development*, 3(4):902–6, 2019.

[7] Mritunjay Ojha, Sheetal Mohite, Shraddha Kathole, and Diksha Tarware. Microcontroller based automatic plant watering system. *International Journal of Computer Science and Engineering*, 5(3):25–36, 2016.

[8] A Rosadi, A Fauzan, and Winarno. Prototype design of automatic plant watering equipment with soil moisture detection system based on arduino uno microcontroller: case study of chili plant. *Journal of Physics: Conference Series*, 1517(1):012077, apr 2020.

[9] P KUSUMA VANI, N AVINASH, M SACHIN KUMAR, M TIRUPATHI RAO, and B SIMHACHALAM. Automated plant watering system. Automated Plant Watering System.