Smart Agricultural Water Management System

ECS Final Report

Submitted To:
Dr. Suyog Jhavar

Submitted By:
Aayush Shukla (21BME7011)
Shaik Mohammed Ammar Ahmed (21BME7003)
Benedict Paul (21BCE8836)
Bhabani Shankar Dash (21BCE7196)
Ishaan Deshpande (21BCE8073)

Importance of Water Management in Agriculture

Water management plays a crucial role in agriculture, impacting both crop yield and sustainability.

Crop Yield

- Adequate water supply is essential for the growth and development of crops. Insufficient water can lead to stunted growth, reduced yields, and lower quality produce.
- Proper water management ensures that crops receive the right amount of water at the right time, optimizing their growth and maximizing yield.

Sustainability

- Water scarcity is a global concern, and agriculture is one of the largest consumers of water resources.
- Efficient water management practices, such as drip irrigation and precision farming, help conserve water and reduce wastage.
- By implementing sustainable water management techniques, farmers can minimize their environmental impact and ensure the long-term viability of their operations.

Challenges in Traditional Water Management

Inefficient Water Usage

•Traditional methods often result in excessive water usage due to lack of precision and control.

•Over-irrigation can lead to water wastage and environmental damage.

Manual Monitoring and Maintenance

•Traditional systems require manual monitoring and maintenance, which can be time-consuming and labor-intensive.

•This can lead to delays in identifying and addressing issues, resulting in crop damage and yield loss.

Limited Data Insights

•Traditional methods lack data collection and analysis capabilities, making it difficult to optimize water usage.

•Farmers have limited visibility into crop water requirements and soil moisture levels.

Lack of Automation

•Manual control of irrigation systems can be prone to human error and inconsistency.

•Automation can improve precision and efficiency in water management.

Overview of Smart Agriculture Water Management System

Features

- Real-time monitoring of soil moisture levels
- Automated irrigation system
- Weather-based watering schedule
- Remote control and monitoring via mobile app
- Data analytics and insights
- Water usage optimization
- Alerts and notifications for irrigation issues

Capabilities

The Smart Agriculture Water Management System is designed to optimize water usage in agricultural settings. By continuously monitoring soil moisture levels and considering weather conditions, the system ensures that crops receive the right amount of water at the right time. The automated irrigation system eliminates the need for manual intervention, saving time and effort for farmers. The system also provides data analytics and insights, allowing farmers to make informed decisions about their irrigation practices. With remote control and monitoring via a mobile app, farmers can easily manage and monitor their irrigation systems from anywhere. The system also sends alerts and notifications for any irrigation issues, ensuring prompt action and preventing water wastage.

What’s New?

- We want to add this smart water management system with drip irrigation process for more water saving. In Drip irrigation system water droplets come out of the pipe, thus reducing waste of water.

Benefits of Smart Agriculture Water Management System

🌿 Water Conservation

Smart Agriculture Water Management System helps optimize water usage by providing real-time data on soil moisture levels, weather conditions, and crop water requirements. This enables farmers to irrigate their fields more efficiently, reducing water waste and conserving this valuable resource.

⚡ Increased Crop Yield

By ensuring that crops receive the right amount of water at the right time, the Smart Agriculture Water Management System helps improve crop yield. This system prevents under-watering or over-watering, which can lead to stunted growth or crop damage. As a result, farmers can expect higher productivity and better-quality produce.

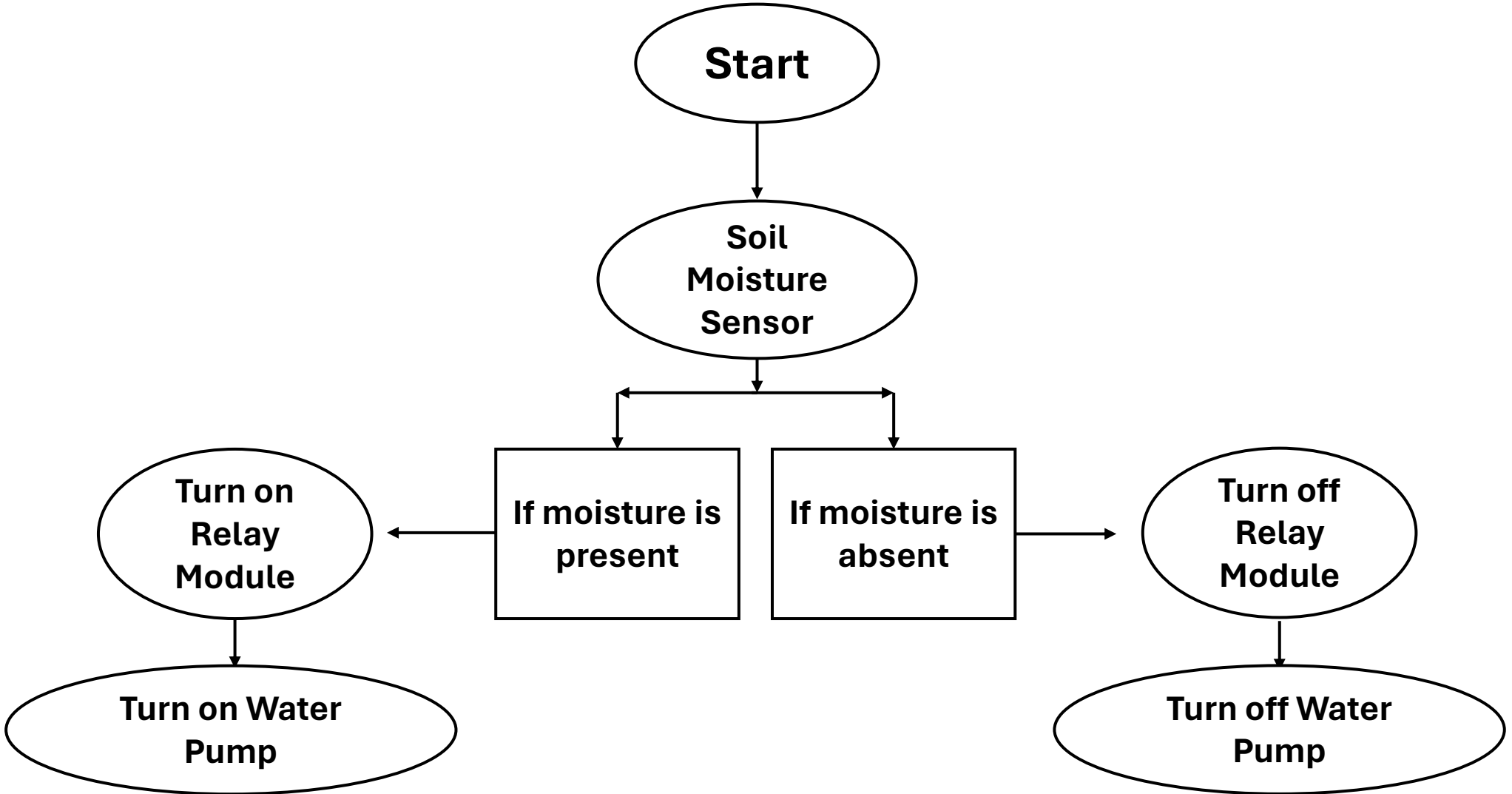
📦 Cost Savings

Implementing the Smart Agriculture Water Management System can lead to significant cost savings for farmers. By optimizing water usage, farmers can reduce their water bills. Additionally, the system provides insights into the irrigation schedule, allowing farmers to save on energy costs associated with pumping water.

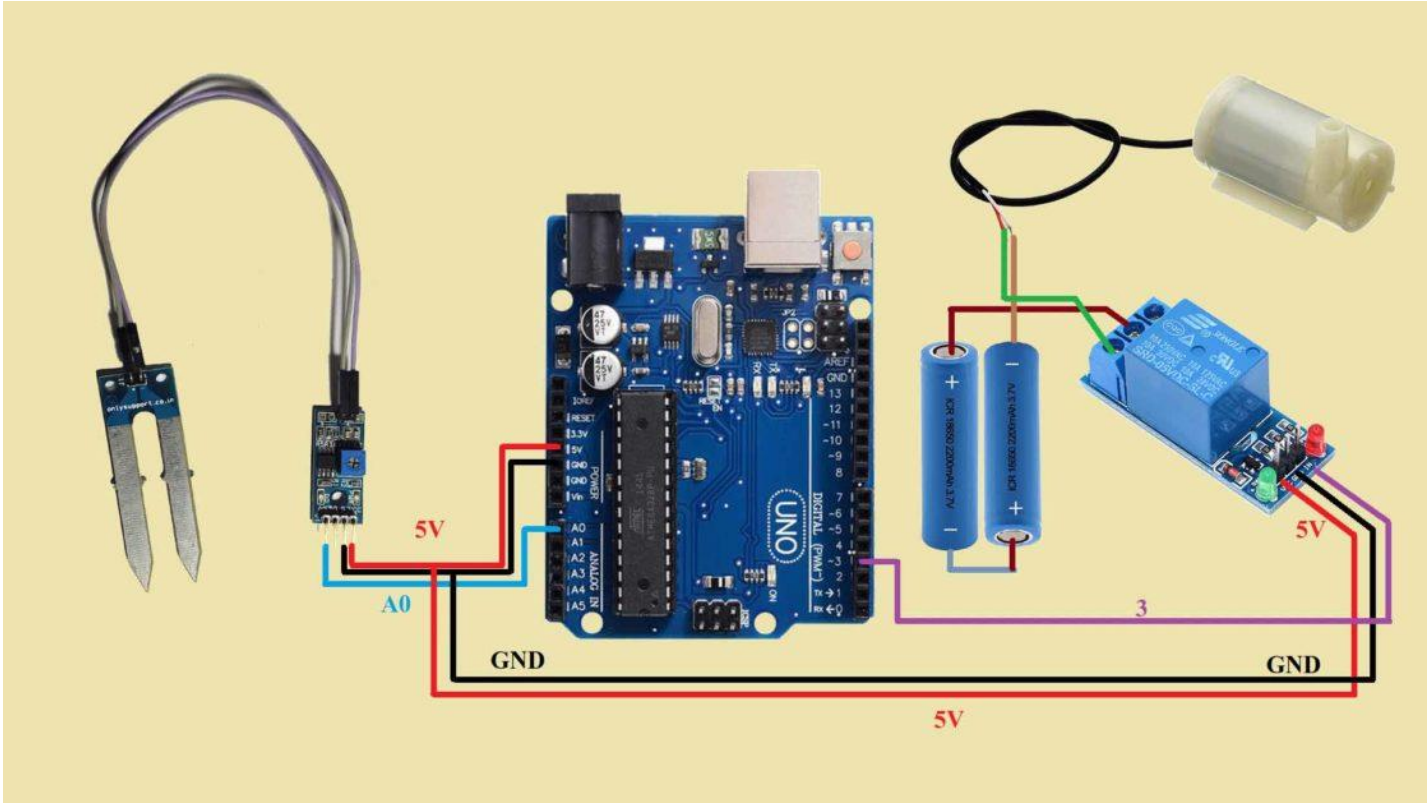
🌎 Environmental Sustainability

By reducing water waste and improving water management practices, the Smart Agriculture Water Management System contributes to environmental sustainability. It helps protect water resources, reduces the impact on local ecosystems, and promotes sustainable farming practices.

Basic Block Diagram



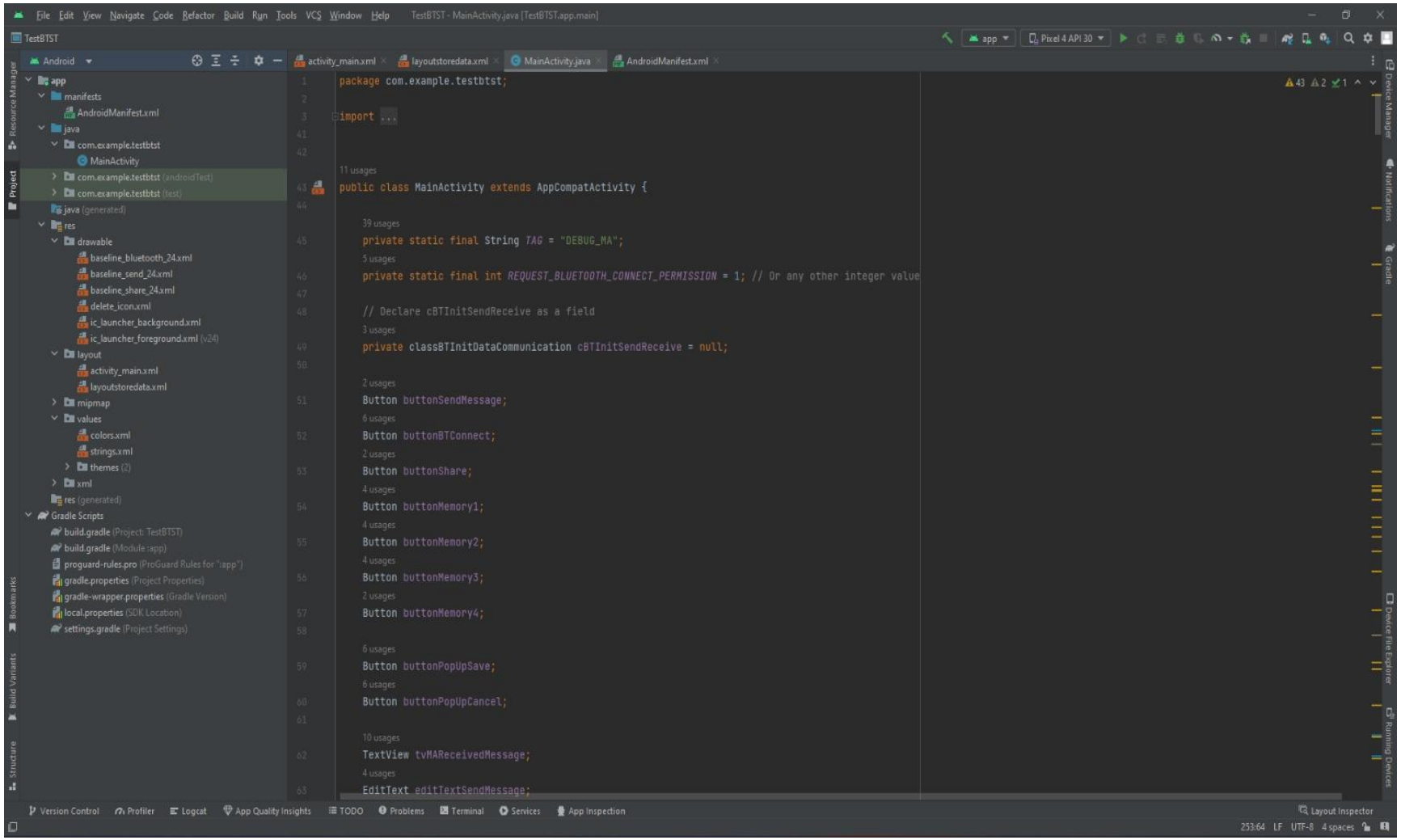
Basic Circuit Diagram



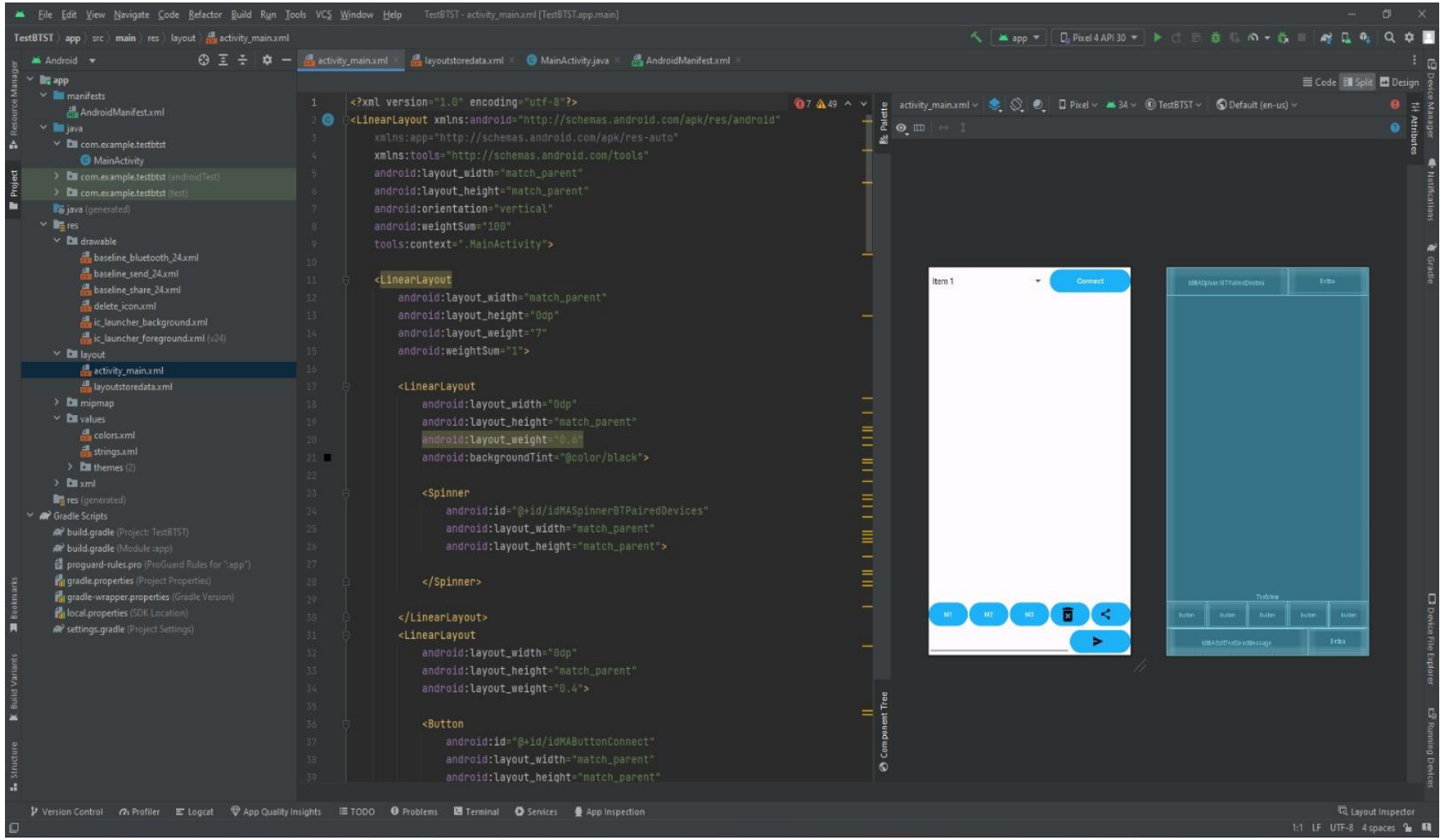
Components Needed For This Project

S. no.	Component	Quantity
1	Arduino UNO R3	1
2	Soil Moisture Sensor	1
3	Jumper Wires	20
4	5V DC Water Pump	1
5	6V Battery	1
6	5V Relay Module	1
7	BreadBoard	1

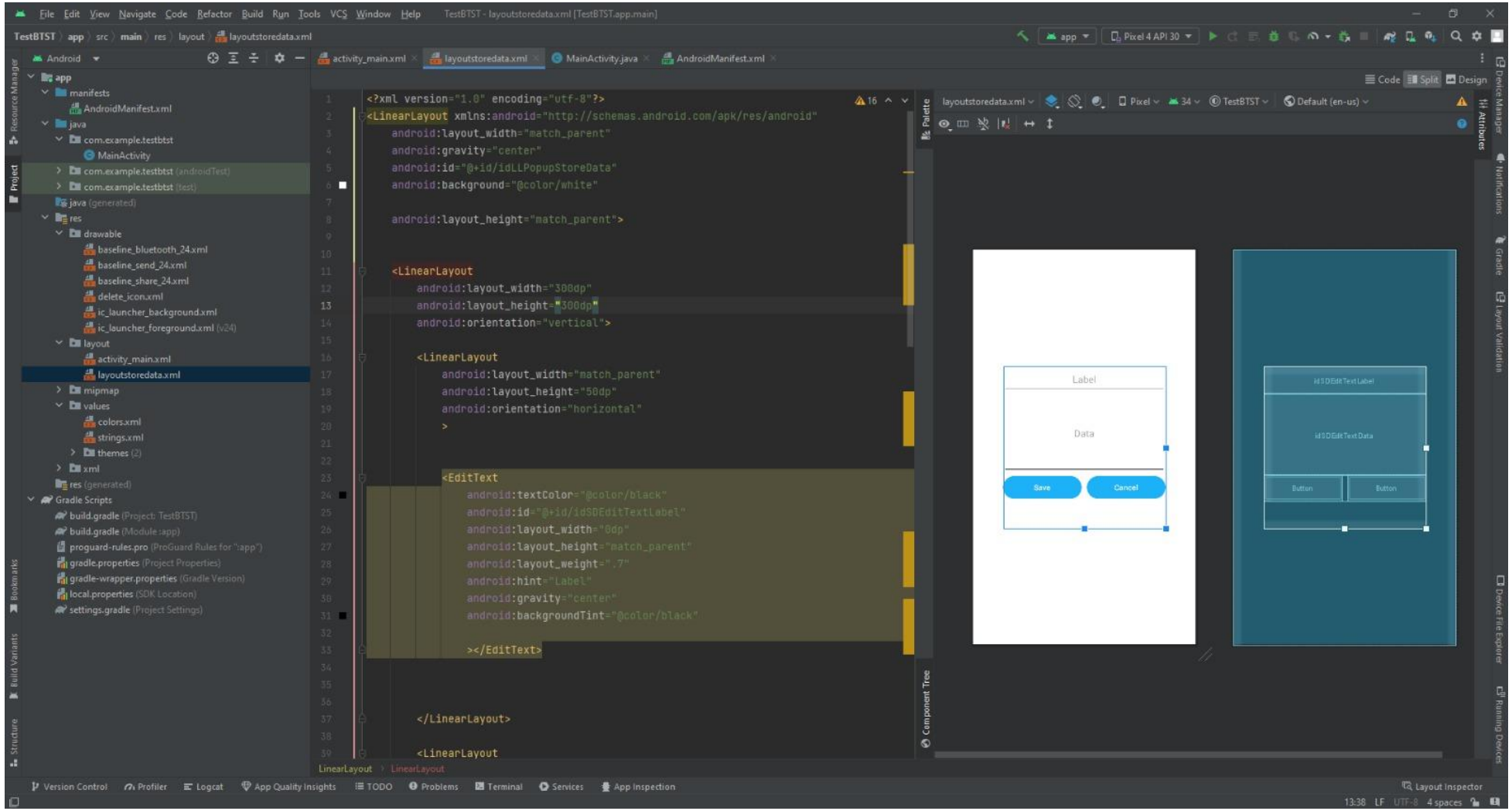
Code


File Edit View Navigate Code Refactor Build Run Tools VCS Window Help
TestBTST - MainActivity.java [TestBTST.app.main]
TestBTST
app
Pixel 4 API 30
Android
activity_main.xml
layoutstoredata.xml
MainActivity.java
AndroidManifest.xml
app
manifests
AndroidManifest.xml
java
com.example.testbtst
MainActivity
com.example.testbtst (androidTest)
com.example.testbtst (test)
java (generated)
res
drawable
baseline_bluetooth_24.xml
baseline_send_24.xml
baseline_share_24.xml
delete_icon.xml
ic_launcher_background.xml
ic_launcher_foreground.xml (v24)
layout
activity_main.xml
layoutstoredata.xml
mipmap
values
colors.xml
strings.xml
themes (2)
xml
res (generated)
Gradle Scripts
build.gradle (Project: TestBTST)
build.gradle (Module :app)
proguard-rules.pro (ProGuard Rules for ":app")
gradle.properties (Project Properties)
gradle-wrapper.properties (Gradle Version)
local.properties (SDK Location)
settings.gradle (Project Settings)
package com.example.testbtst;
import ...
11 usages
public class MainActivity extends AppCompatActivity {
39 usages
private static final String TAG = "DEBUG_MA";
5 usages
private static final int REQUEST_BLUETOOTH_CONNECT_PERMISSION = 1; // Or any other integer value
// Declare cBTInitSendReceive as a field
3 usages
private classBTInitDataCommunication cBTInitSendReceive = null;
2 usages
Button buttonSendMessage;
6 usages
Button buttonBTConnect;
2 usages
Button buttonShare;
4 usages
Button buttonMemory1;
4 usages
Button buttonMemory2;
4 usages
Button buttonMemory3;
2 usages
Button buttonMemory4;
6 usages
Button buttonPopUpSave;
6 usages
Button buttonPopUpCancel;
10 usages
TextView tvMAReceivedMessage;
4 usages
EditText editTextSendMessage;
Version Control
Profiler
Logcat
App Quality Insights
TODO
Problems
Terminal
Services
App Inspection
Layout Inspector
253:64 LF UTF-8 4 spaces

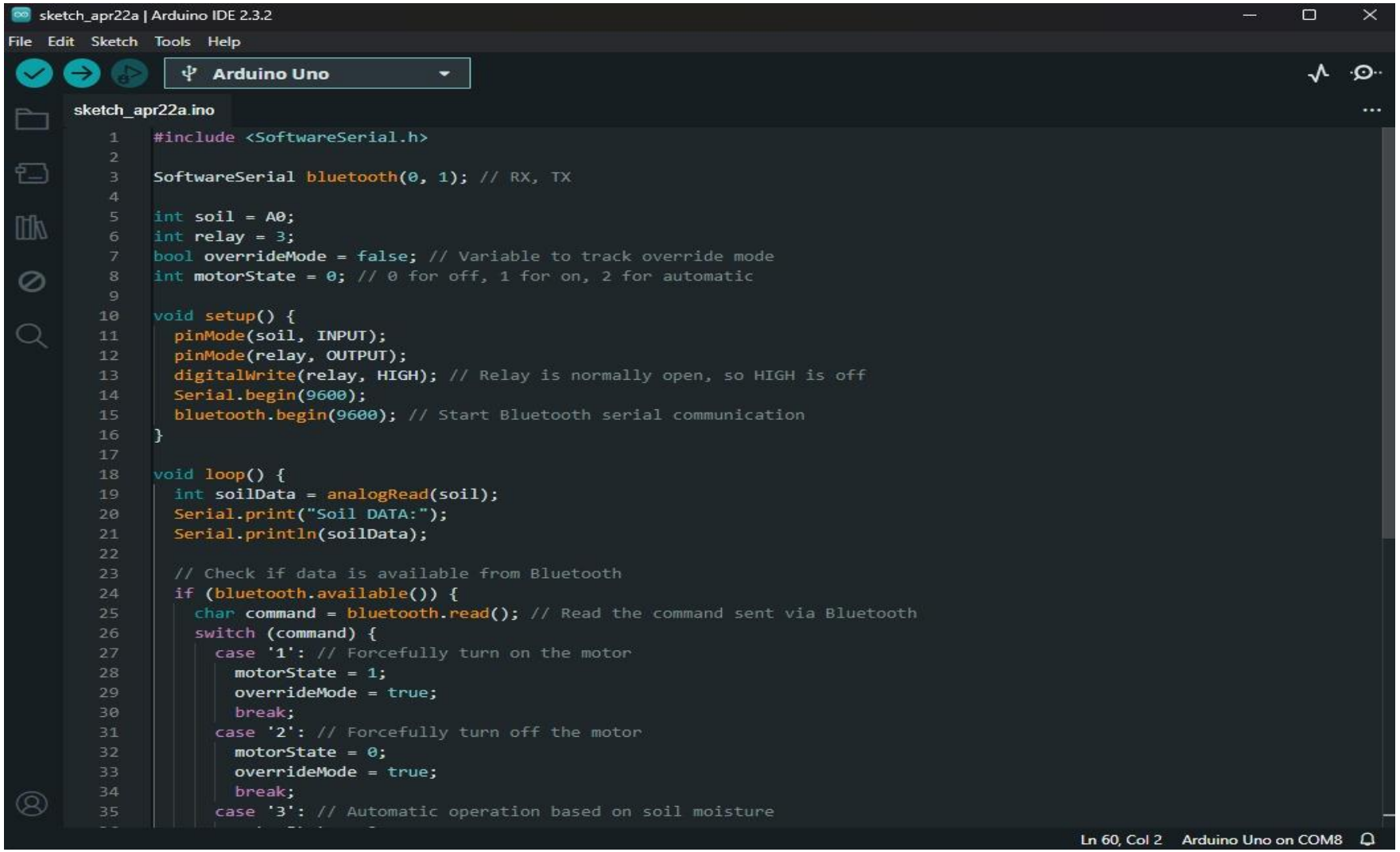
Code and App Layout



App’s Code Layout



Arduino Code



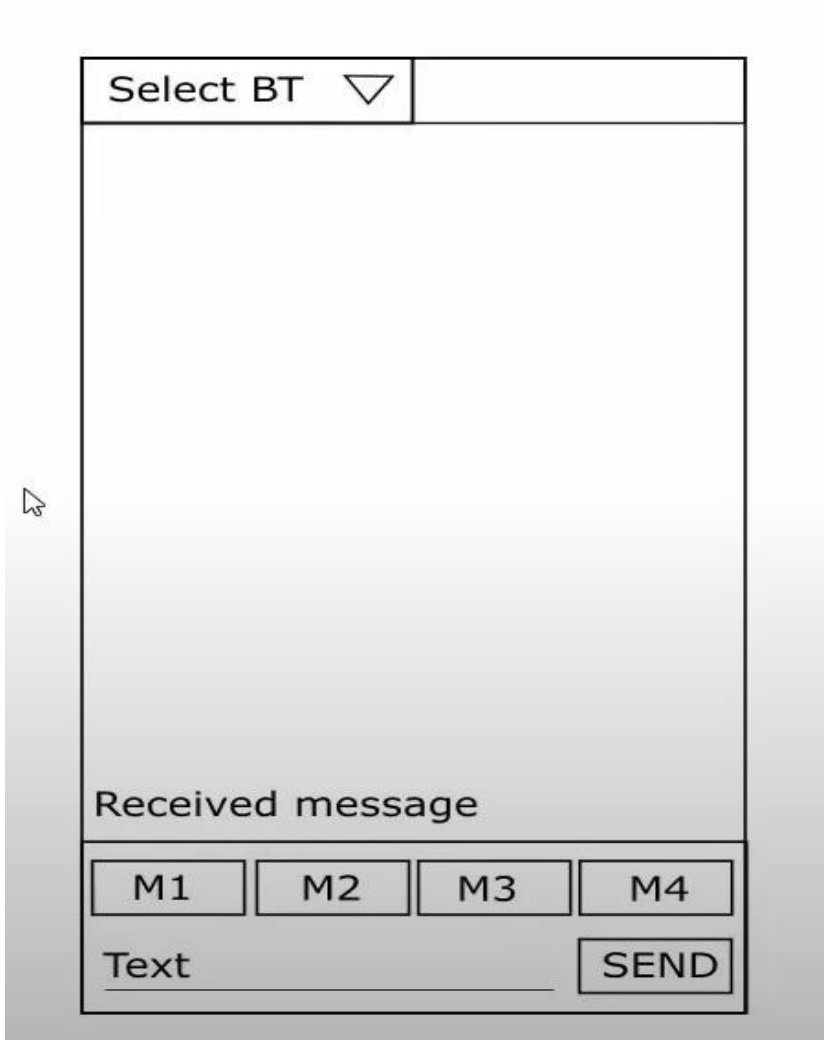
Arduino Code

```
#include <SoftwareSerial.h>

SoftwareSerial bluetooth(0, 1); // RX, TX

int soil = A0;
int relay = 3;
bool overrideMode = false; // Variable to track override mode

void setup() {
  pinMode(soil, INPUT);
  pinMode(relay, OUTPUT);
  digitalWrite(relay, HIGH);
  Serial.begin(9600);
  bluetooth.begin(9600); // Start Bluetooth serial communication
}

void loop() {
  int soilData = analogRead(soil);
  Serial.print("Soil DATA:");
  Serial.println(soilData);

  if (bluetooth.available()) { // Check if data is available from Bluetooth
    char command = bluetooth.read(); // Read the command sent via Bluetooth
    if (command == '1') { // If '1' is received, enter override mode
      overrideMode = true;
      digitalWrite(relay, LOW); // Turn on the motor
    } else if (command == '2') { // If '2' is received, exit override mode
      overrideMode = false;
    }
  }

  if (!overrideMode) { // If not in override mode, follow normal operation
based on soil moisture
    if (soilData > 900) {
      digitalWrite(relay, LOW); // Turn on the motor if soil is dry
    } else {
      digitalWrite(relay, HIGH); // Turn off the motor if soil is wet
    }
  }
}
```

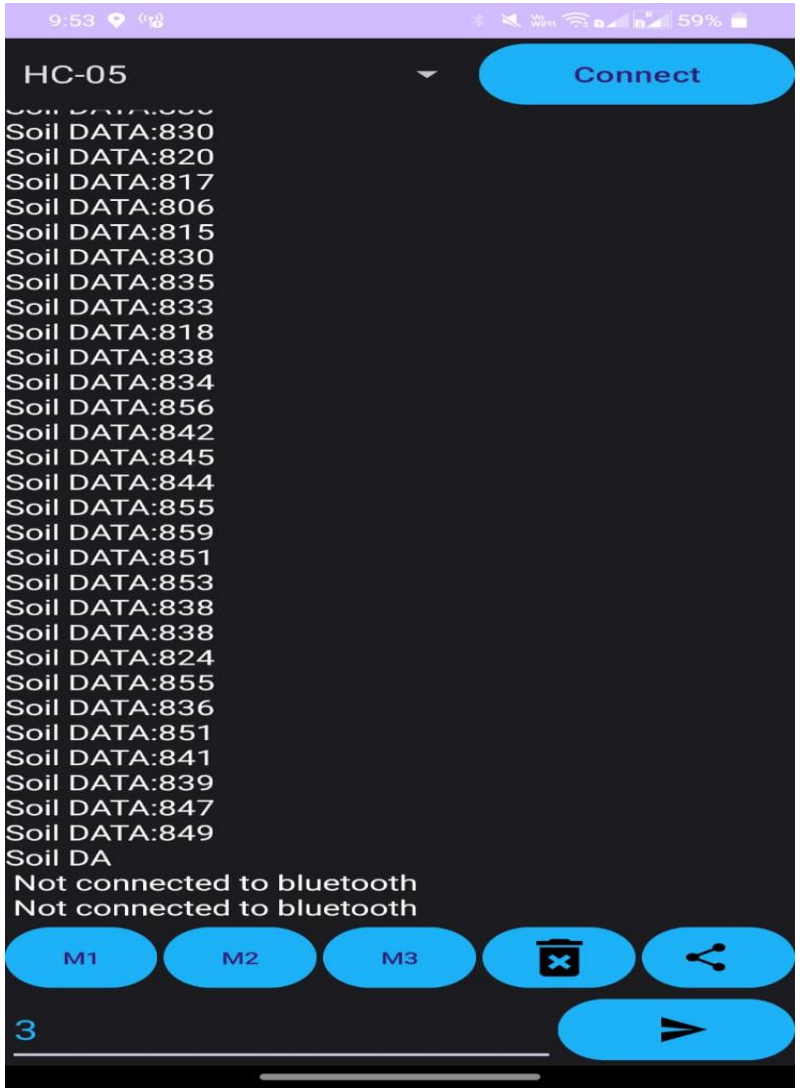
Arduino Code

```
#include <SoftwareSerial.h>

SoftwareSerial bluetooth(0, 1); // RX, TX

int soil = A0;
int relay = 3;
bool overrideMode = false; // Variable to track override mode

void setup() {
  pinMode(soil, INPUT);
  pinMode(relay, OUTPUT);
  digitalWrite(relay, HIGH);
  Serial.begin(9600);
  bluetooth.begin(9600); // Start Bluetooth serial communication
}

void loop() {
  int soilData = analogRead(soil);
  Serial.print("Soil DATA:");
  Serial.println(soilData);

  if (bluetooth.available()) { // Check if data is available from Bluetooth
    char command = bluetooth.read(); // Read the command sent via Bluetooth
    if (command == '1') { // If '1' is received, force motor on
      overrideMode = true;
      digitalWrite(relay, LOW); // Turn on the motor
    } else if (command == '2') { // If '2' is received, force motor off
      overrideMode = false;
      digitalWrite(relay, HIGH); // Turn off the motor
    } else if (command == '3') { // If '3' is received, operate based on
soil moisture
      overrideMode = false;
      if (soilData > 900) {
        digitalWrite(relay, LOW); // Turn on the motor if soil is dry
      } else {
        digitalWrite(relay, HIGH); // Turn off the motor if soil is wet
      }
    }
  }

  if (!overrideMode) { // If not in override mode, follow normal operation
based on soil moisture
    if (soilData > 900) {
      digitalWrite(relay, LOW); // Turn on the motor if soil is dry
    } else {
      digitalWrite(relay, HIGH); // Turn off the motor if soil is wet
    }
  }
}
```

Sketch of a User Interface

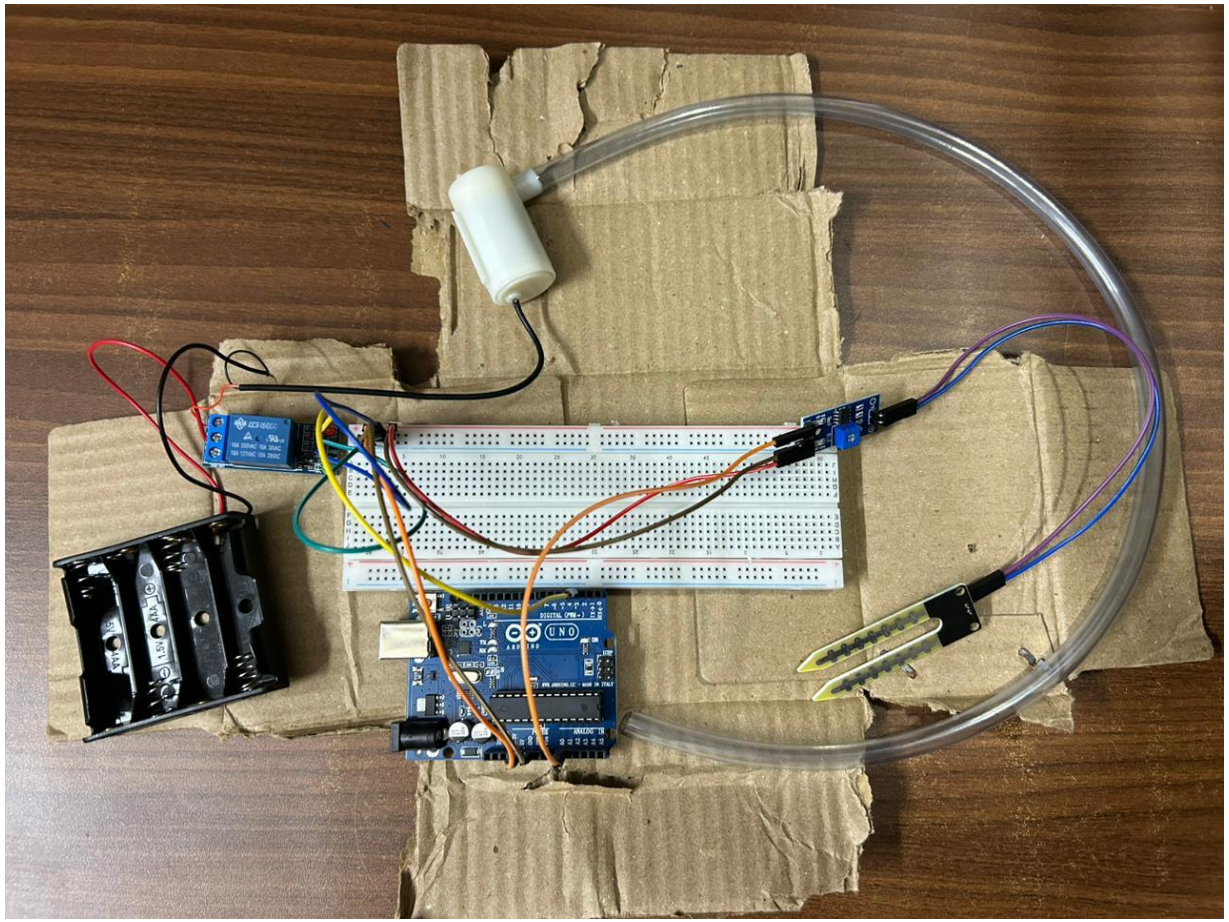


Soil Moisture sensor reading



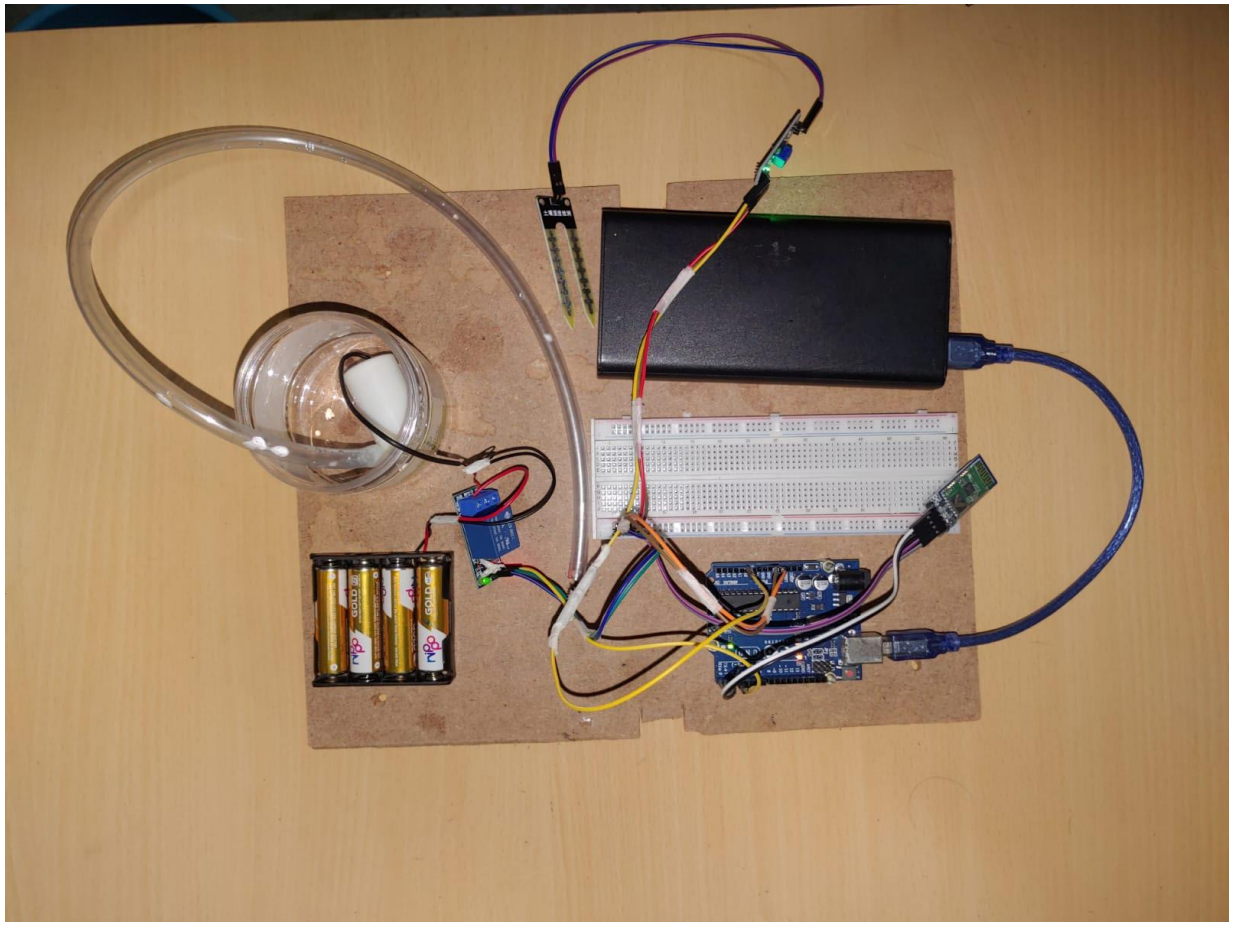
This shows the amount of moisture present in soil after the data is collected through sensor.

Progression



Till now, we have completed the circuit and tested it. We are now planning on How to connect this system with mobile and Bluetooth system. And we are currently writing the codes for Arduino.

Complete product



This is the complete product. After review 2 , we added Bluetooth way to connect it with mobile. It is tested with Arduino codes.

Conclusion

- The Smart Agriculture Water Management System is a crucial tool for optimizing water usage in agriculture.
- It allows farmers to monitor and control irrigation systems remotely, resulting in more efficient water usage.
- By adopting this system, farmers can reduce water waste, increase crop yields, and improve overall sustainability.
- The Smart Agriculture Water Management System is a cost-effective solution that benefits both farmers and the environment.

THANK YOU