

## Project Delivery Sprint –2

Date	30 September 2022
Team ID	PNT2022TMID20174
Project Name	Smart Farmer - IoT Enabled Smart Farming Application

### Code :

```
// Include Libraries
```

```
#include "Arduino.h"
```

```
#include "DHT.h"
```

```
#include "PIR.h"
```

```
#include "SoilMoisture.h"
```

```
#include "Pump.h"
```

```
// Pin Definitions
```

```
#define DHT_PIN_DATA 3
```

```
#define PIR_PIN_SIG 4
```

```
#define SOILMOISTURE_5V_PIN_SIG A10
```

```
#define WATERPUMP_PIN_COIL1 2
```

```
// Global variables and defines
```

```
// object initialization
```

```
DHT dht(DHT_PIN_DATA);
```

```
PIR pir(PIR_PIN_SIG);
```

```
SoilMoisture soilMoisture_5v(SOILMOISTURE_5V_PIN_SIG);
```

```
Pump waterpump(WATERPUMP_PIN_COIL1);
```

```

// define vars for testing menu

const int timeout = 10000; //define timeout of 10 sec char

menuOption = 0;

long time0;


// Setup the essentials for your circuit to work. It runs first every time your circuit is powered with
electricity. void setup()
{
    // Setup Serial which is useful for debugging
    // Use the Serial Monitor to view printed messages Serial.begin(9600);
    while (!Serial) ; // wait for serial port to connect. Needed for native USB
    Serial.println("start"); dht.begin();
    menuOption = menu();
}


// Main logic of your circuit. It defines the interaction between the components you selected. After
setup, it runs over and over again, in an eternal loop.

void loop()
{
    if(menuOption == '1') {
        // DHT22/11 Humidity and Temperature Sensor - Test Code
        // Reading humidity in % float
        dhtHumidity = dht.readHumidity();
        // Read temperature in Celsius, for Fahrenheit use .readTempF() float
        dhtTempC = dht.readTempC();
        Serial.print(F("Humidity: ")); Serial.print(dhtHumidity); Serial.print(F(" [%]\t"));
        Serial.print(F("Temp: ")); Serial.print(dhtTempC); Serial.println(F(" [C]"));
    }
    else if(menuOption == '2') {

```

```

// Infrared PIR Motion Sensor Module - Test Code

bool pirVal = pir.read();

Serial.print(F("Val: ")); Serial.println(pirVal);
}

else if(menuOption == '3') { // Soil Moisture
  Sensor - Test Code int soilMoisture_5vVal =
  soilMoisture_5v.read();

  Serial.print(F("Val: ")); Serial.println(soilMoisture_5vVal);
}

else if(menuOption == '4') {
  // Submersible Pool Water Pump - Test Code // The
  water pump will turn on and off for 2000ms (4 sec)
  waterpump.on(); // 1. turns on delay(2000); // 2. waits
  500 milliseconds (0.5 sec).
  waterpump.off();// 3. turns off
  delay(2000);    // 4. waits 500 milliseconds (0.5 sec).

  }
if (millis() - time0 > timeout)
{
  menuOption = menu();
}

}

// Menu function for selecting the components to be tested
// Follow serial monitor for instructions char
menu()
{

  Serial.println(F("\nWhich component would you like to test?"));
  Serial.println(F("(1) DHT22/11 Humidity and Temperature Sensor"));

```

```

Serial.println(F("(2) Infrared PIR Motion Sensor Module"));
Serial.println(F("(3) Soil Moisture Sensor"));
Serial.println(F("(4) Submersible Pool Water Pump"));
Serial.println(F("(menu) send anything else or press on board reset button\n"));
while (!Serial.available());

// Read data from serial monitor if received while
(Serial.available())
{
    char c = Serial.read(); if
    (isAlphaNumeric(c))
    {

        if(c == '1')

            Serial.println(F("Now Testing DHT22/11 Humidity and Temperature
Sensor"));
        else if(c == '2')

            Serial.println(F("Now Testing Infrared PIR Motion Sensor Module"));
        else if(c == '3')

            Serial.println(F("Now Testing Soil Moisture Sensor"));
        else if(c == '4')

            Serial.println(F("Now Testing Submersible Pool Water Pump"));

    }
    else
    {

        Serial.println(F("illegal input!")); return
        0;
    }
    time0 = millis(); return
    c;

```

```
}  
  
}  
  
}
```

## Circuit Diagram :

