**PLANT MONITORING SYSTEM**

**Group 34**

**Introducing our team and problem :**

**MENTOR: Prof. Prosenjit Kundu**

**GROUP MEMBER DETAILS:**

**Jivani Madhav Bhavesh - 202201285**

**Ghori Zeel Jivrajbhai - 202201287**

**Adit Shah - 202201289**

**Patel Kavit - 202201290**

**Patel Naisargi Nilangbhai - 202201291**

**Soni Devarsh Sushant - 202201446**

**PROBLEM STATEMENT:**

**The main issue noticed is that people are less concerned while taking care of plants. The lack of real-time monitoring and control of key environmental factors, leading to difficulties in detecting early signs of stress or disease, inefficient resource usage, and limited ability to make data-driven decisions for optimal plant care.**

**MOTIVATION:**

**A plant monitoring system provides growers with the ability to closely observe and control environmental factors, detect early signs of stress or disease, optimize resource usage, automate processes, make data-driven decisions, and facilitate scientific research. By continuously monitoring plant health parameters and environmental conditions, such systems enhance plant care, increase productivity, conserve resources, enable remote management, and generate valuable insights for improved plant health and overall efficiency in various contexts, from agriculture and horticulture to scientific experimentation and home gardening.**

**Proposed Solution :**

**DESIGN AND IMPLEMENTATION:**

**The design contains plants enabled with the electronic component Arduino circuit with other electronic components like solar panel, sensors like DHT11, BH1750, JXCT NPK, MPU9250, solar cells, etc.**

**This design helps in automation and remote management, data-driven decision making and resource efficiency.The sensors will detect the problem and send the signal to Arduino which will provide the solution of the problem. There will be water supply when water is required, if there is too much sunshine or wind speed then the shed will come out to protect the plant, etc. To provide voltage to all the electronics sources we also decided to use solar panels and cells so that the whole device is eco-friendly and does not harm the environment. By collecting and analyzing this data, growers can make data-driven decisions to optimize plant care strategies, develop predictive models, and improve overall plant health and productivity.**

**Role of sensors and how system works :**

**DHT11 - for the crop planted in it if temp. rises above a certain level the green shield will drop and also if humidity level of soil drops below a certain level the crops will be watered accordingly to maintain the humidity level of soil in a safe range**

**Also in a case of heavy rain if soil humidity rises above a certain level heavy plastic shield will drop to avoid more water entering the system**

**BH1750 - If temp does not rise much in the area but sun rays are damaging the plant again the green shield will drop**

**JXCT NPK sensor : To keep a track of important nutrients in soil if they are just around boundary the user will be send notifications regarding addition of fertilizer**

**MPU9250 : If strong winds that could uproot plants or may be damaged at a certain level start blowing, a transparent heavy plastic shield will drop in order to avoid them.**

**Timeline :**

**Initial 20 days -> construction of main base and setup**

**Next 10 days including sensors and testing their overall capabilities**

**Next 25 to 30 days including the wind shield and green shield to circuit and make them functional**

**Next 20 to 25 days including self charging i.e. solar panels and battery in setup and connecting to all components**

**Next 5 to 10 days regular testing of whole model in different ways and making it design look good i.e. giving it a good look**

**Next 15 to 20 days(additional work not stated in implementation)-> either connecting beepers Or building a website to notify the user whichever seems more convincing.**

**Tentative Expenditure :**

**DHT11 Temperature And Humidity Sensor - 310**

**BH1750 - Light Intensity Sensor Module -250**

**JXCT SOIL npk sensor - 70**

**MPU9250 9-Axis Gyro Accelerometer Module - 600**

**Arduino - 600**

**Solar Panel - 2000**

**Motor - 500**

**Setup - 2500**

**Total ~ 7000(approx)**

**(subject to change because of battery being used to supply power and store solar energy)**

**CONCLUSION:**

**As this device can reduce the efforts and assistance for taking care of plants, it can be implemented in real life and in manufacturing with many more features added to it, like Bluetooth or Wifi enabled Technology which would connect Google assistant or Alexa to it.**

**ARDUINO CODE:**

#include <dht.h>

#include <BH1750.h>

#include <Wire.h>

#define PUMP\_PIN 10

#define RELAY 5

#define DHT11\_PIN 2

int temp, humid;

dht DHT;

const int soil\_pin = A1; /\* Soil moisture sensor O/P pin \*/

int moisture;

BH1750 lightMeter;

void setup(){

Serial.begin(9600);

pinMode(PUMP\_PIN, OUTPUT);

pinMode(RELAY, OUTPUT);

pinMode(soil\_pin, INPUT); //soil moisture

Wire.begin();

lightMeter.begin(); //bh1750

delay(5000);

}

void loop(){

int chk = DHT.read11(DHT11\_PIN);

Serial.print("Temperature = ");

Serial.println(DHT.temperature);

Serial.print("Humidity = ");

Serial.println(DHT.humidity);

Serial.print("%\n\n");

humid = DHT.humidity;

temp = DHT.temperature;

if(humid <= 100){

digitalWrite(PUMP\_PIN, HIGH); //WATER PUMP

delay(10000);

digitalWrite(PUMP\_PIN, LOW);

}/\*else{

digitalWrite(PUMP\_PIN, LOW);

}\*/

if(temp >= 40){

digitalWrite(RELAY, HIGH); //GREEN SHIELD

delay(5000);

digitalWrite(RELAY, LOW);

}/\*else{

digitalWrite(RELAY, LOW);

}\*/

moisture = analogRead(soil\_pin);

moisture = map(moisture, 0, 1023, 0, 100);

Serial.print("Moisture Percentage = ");

Serial.print(moisture);

Serial.print("%\n\n");

if(moisture <= 10){

digitalWrite(PUMP\_PIN, HIGH); //WATER PUMP

delay(10000);

digitalWrite(PUMP\_PIN, LOW);

}

float lux = lightMeter.readLightLevel();

Serial.print("Light: ");

Serial.print(lux);

Serial.println(" lx");

if(lux >= 12000){

digitalWrite(RELAY, HIGH);

delay(5000);

digitalWrite(RELAY, LOW);

}

delay(15000);

}