



T.C.

AFYON KOCATEPE ÜNİVERSİTESİ
MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

212923001 Aslı Şemşimoğlu
212923003 Rabia Durgut

Robotik Dersi Proje Raporu

2024

AFYONKARAHİSAR

Konuřan Akıllı Bitki Bakım Robotu Proje Raporu

Proje Amacı:

Bu projenin amacı, bitkilerin topraktaki nem ve su seviyelerini otomatik olarak takip eden, kullanıcıya hem görsel hem de sesli uyarılarla bilgi veren bir akıllı sistem tasarlamaktır. Sistem; toprağın kuru, nemli veya ıslak durumlarına göre gerçek zamanlı olarak bilgi sunarken, Thingspeak platformu üzerinden verileri anlık olarak izleme olanağı sağlar. Bu çözüm, bitki bakımını kolaylaştırmayı, verimliliği artırmayı ve bitki sağlığını optimize etmeyi hedefler.

Proje Kapsamı:

- Nem ve Su Seviyesi Takibi: Toprak nem seviyesinin ve su seviyesinin sensörlerle tespit edilmesi.
- Görsel Bildirimler: LED ve LCD ekran kullanılarak nem ve su durumu bilgisinin aktarılması.
- Thingspeak Entegrasyonu: Verilerin Thingspeak platformuna gönderimi ve Thingspeak API üzerinden anlık izleme.
- Sesli Uyarı Sistemi: Python kodu kullanılarak toprağın durumuna göre sesli uyarılar verilmesi.

Kullanılan Malzemeler:

- ESP32: Wi-Fi bağlantısı ve Thingspeak entegrasyonu için.
- 3 Renk LED:
 - Kırmızı: Kuru toprağı belirtir.
 - Yeşil: Nemli toprağı belirtir.
 - Mavi: Islak toprağı belirtir.
- Buzzer: Kritik durumlar için sesli uyarı sağlar.
- Potansiyometre: LCD ekran parlaklığını ayarlamak için.
- LCD Ekran: Nem seviyesi ve durum bilgilerini görselleştirir.
- Su Sensörü: Su seviyesini ölçer ve **Thingspeak**'e veri gönderir.
- Nem Sensörü: Toprağın nem seviyesini ölçerek durum bilgisini belirler.

Proje Fonksiyonları ve İşlevleri:

a. Nem ve Su Seviyesi Takibi:

- Nem Sensörü: 0-4095 arasında bir değer üreterek toprağın nem seviyesini ölçer.
- Su Sensörü: Su seviyesini algılar ve Thingspeak platformuna veri gönderir.

b. LCD Ekran ile Görsel Bildirim:

- LCD ekranda anlık nem seviyesi ve durum bilgisi yazılı olarak gösterilir.
 - "Durum: COK KURU!"
 - "Durum: NEMLI"
 - "Durum: COK ISLAK"

c. LED ve Buzzer ile Durum Bilgisi:

- LED'ler:
 - Kırmızı LED: Kuru toprağı belirtir.
 - Yeşil LED: Nemli toprağı belirtir.
 - Mavi LED: Islak toprağı belirtir.
- Buzzer: Kritik durumlarda sesli uyarı sağlar.

d. Thingspeak Entegrasyonu:

- ESP32, sensörlerden aldığı verileri Thingspeak platformuna gönderir.
- Thingspeak üzerinden nem ve su seviyesi gerçek zamanlı olarak takip edilir.

e. Python ile Sesli Uyarı Sistemi:

- Python Uygulaması: Thingspeak API'sinden veri alır ve toprağın durumundaki değişiklikleri algılar.
- Sesli Uyarılar:
 - dry_soil.mp3: "Toprak çok kuru"
 - moist_soil.mp3: "Toprak nemli"
 - wet_soil.mp3: "Toprak çok ıslak"

Sistem İşleyişi:

1. ESP32 İşlevleri:

- Nem ve su sensörlerinden veri alır.
- Bu verileri Thingspeak platformuna gönderir.
- LCD ekranda nem seviyesi ve durum bilgisini gösterir.
- LED'ler ve buzzer ile durum bildirimlerini görsel ve sesli olarak yapar.

2. Thingspeak Entegrasyonu:

- Thingspeak üzerinden nem ve su seviyesi verileri anlık olarak izlenir.

3. Python ile Sesli Uyarı:

- Python kodu Thingspeak'ten verileri çeker.
- Toprak durumundaki değişiklikleri algılar ve uygun ses dosyasını çalar.

Sonuç:

Konuşan Akıllı Bitki Bakım Robotu, kullanıcı dostu bir bitki bakım sistemi sunar. Bu proje, görsel ve sesli uyarı sistemleri sayesinde:

- Seri Monitörde: Su seviyesi detaylı olarak izlenir.
- LCD Ekranda: Toprak nem seviyesi ve durum bilgisi görüntülenir.
- Thingspeak Üzerinden: Anlık nem ve su seviyeleri takip edilir.
- Python Koduyla: Duruma uygun sesli uyarılar verilerek kullanıcı bilgilendirilir.

Arduino Kodları:

```
#include <Arduino.h>
#include <WiFi.h>
#include <HTTPClient.h>
#include <Wire.h>
#include <LiquidCrystal_I2C.h>

// LCD ekran ayarı
LiquidCrystal_I2C lcd(0x27, 16, 2);

// Wi-Fi Bilgileri
#define WIFI_SSID "Your_WIFI_SSID"
#define WIFI_PASSWORD "Your_WIFI_PASSWORD"

// Thingspeak API Bilgileri
#define THINGSPEAK_API_KEY "API_KEY" // Thingspeak Write API Key
#define THINGSPEAK_URL "http://api.thingspeak.com/update"

// Pin Tanımları
#define SOIL_SENSOR_PIN 36 // Toprak nem sensörü
#define WATER_SENSOR_PIN 34 // Su sensörü
#define RED_LED 25 // Kırmızı LED
#define GREEN_LED 27 // Yeşil LED
#define BLUE_LED 26 // Mavi LED
#define BUZZER_PIN 33 // Buzzer

int prevSoilState = -1; // Önceki toprak durumu

// Toprak nem seviyesi okuma
int readSoilMoisture() {
    int total = 0;
    for (int i = 0; i < 10; i++) {
        total += analogRead(SOIL_SENSOR_PIN);
        delay(10);
    }
    return total / 10;
}

// Su seviyesi okuma
int readWaterLevel() {
```

```
int total = 0;
for (int i = 0; i < 10; i++) {
    total += analogRead(WATER_SENSOR_PIN);
    delay(10);
}
return total / 10;
}

void setup() {
    pinMode(SOIL_SENSOR_PIN, INPUT);
    pinMode(WATER_SENSOR_PIN, INPUT);
    pinMode(RED_LED, OUTPUT);
    pinMode(GREEN_LED, OUTPUT);
    pinMode(BLUE_LED, OUTPUT);
    pinMode(BUZZER_PIN, OUTPUT);

    Serial.begin(115200);

    // LCD ekran başlat
    lcd.init();
    lcd.backlight();

    // Wi-Fi bağlantısı
    Serial.print("Wi-Fi'ye bağlanılıyor...");
    WiFi.begin(WIFI_SSID, WIFI_PASSWORD);
    while (WiFi.status() != WL_CONNECTED) {
        delay(500);
        Serial.print(".");
    }
    Serial.println("\nWi-Fi bağlantısı başarılı!");
    lcd.clear();
    lcd.print("Wi-Fi Bağlandı");
}

void loop() {
    int soilMoisture = readSoilMoisture(); // Toprak nemi oku
    int waterLevel = readWaterLevel();    // Su seviyesi oku
    int soilState;

    // Nem seviyesine göre durumu belirle
    if (soilMoisture < 2200) {
        soilState = 2; // Çok sulu
    }
}
```

```
} else if (soilMoisture >= 2200 && soilMoisture < 3900) {
    soilState = 1; // Nemli
} else {
    soilState = 0; // Kuru
}

// LCD ekran güncellemesi
lcd.clear();
lcd.setCursor(0, 0);
lcd.print("Nem Orani: ");
lcd.print(soilMoisture);
lcd.setCursor(0, 1);

if (soilState == 0) {
    lcd.print("Durum: COK KURU!");
} else if (soilState == 1) {
    lcd.print("Durum: NEMLI   ");
} else {
    lcd.print("Durum: COK ISLAK");
}

// LED ve buzzer kontrolü
if (soilState != prevSoilState) {
    if (soilState == 0) {
        digitalWrite(REDA_LED, HIGH);
        digitalWrite(GREEN_LED, LOW);
        digitalWrite(BLUE_LED, LOW);
        digitalWrite(BUZZER_PIN, HIGH);
        Serial.println("DRY");
    } else if (soilState == 1) {
        digitalWrite(REDA_LED, LOW);
        digitalWrite(GREEN_LED, HIGH);
        digitalWrite(BLUE_LED, LOW);
        digitalWrite(BUZZER_PIN, LOW);
        Serial.println("MOIST");
    } else {
        digitalWrite(REDA_LED, LOW);
        digitalWrite(GREEN_LED, LOW);
        digitalWrite(BLUE_LED, HIGH);
        digitalWrite(BUZZER_PIN, LOW);
        Serial.println("WET");
    }
}
```

```
    prevSoilState = soilState;
}

// Su seviyesi kontrolü
Serial.print("Su Seviyesi: ");
Serial.println(waterLevel);

// Thingspeak'e veri gönderimi
HTTPClient http;
String url = THINGSPEAK_URL;
url += "?api_key=" + String(THINGSPEAK_API_KEY);
url += "&field1=" + String(soilMoisture); // Nem seviyesi
url += "&field2=" + String(waterLevel); // Su seviyesi

http.begin(url);
int httpResponseCode = http.GET();

if (httpResponseCode > 0) {
    Serial.print("Thingspeak'e veri gönderildi. HTTP Kodu: ");
    Serial.println(httpResponseCode);
} else {
    Serial.print("Thingspeak'e veri gönderme hatası. HTTP Hatası: ");
    Serial.println(httpResponseCode);
}
http.end();

delay(2000); // 2 saniyede bir güncelle
}
```

Seri Monitör Örnek Çıktı Görseli:

```
Output  Serial Monitor x
Message (Enter to send message to 'ESP32 Dev Module' on 'COM3')

Su Seviyesi: 161
Thingspeak'e veri gönderildi. HTTP Kodu: 200
Su Seviyesi: 168
Thingspeak'e veri gönderildi. HTTP Kodu: 200
Su Seviyesi: 164
Thingspeak'e veri gönderildi. HTTP Kodu: 200
Su Seviyesi: 158
Thingspeak'e veri gönderildi. HTTP Kodu: 200
Su Seviyesi: 151
Thingspeak'e veri gönderildi. HTTP Kodu: 200
Su Seviyesi: 146
Thingspeak'e veri gönderildi. HTTP Kodu: 200
Su Seviyesi: 152
```

Python Kodu:

```
import requests
import time
import pygame # Ses çalma için kütüphane

# ThingSpeak API Bilgileri
READ_API_KEY = "XLTE289H3FKHRGHJ" # ThingSpeak Read API Key
CHANNEL_ID = "2797072" # ThingSpeak Kanal ID
URL =
f"https://api.thingspeak.com/channels/{CHANNEL_ID}/feeds.json?api_key={READ_API_KEY}&results=1"

# Ses dosyaları
SOUNDS = {
    "dry": r"C:\Users\semsi\OneDrive\Masaüstü\Bitki-Bakim-Robotu,\dry_soil.mp3",
    "moist":
r"C:\Users\semsi\OneDrive\Masaüstü\Bitki-Bakim-Robotu,\moist_soil.mp3",
    "wet": r"C:\Users\semsi\OneDrive\Masaüstü\Bitki-Bakim-Robotu,\wet_soil.mp3"
}

# Pygame başlat
pygame.mixer.init()

# Önceki durumun takibi
previous_state = None

while True:
    try:
        # ThingSpeak verilerini al
```



```
response = requests.get(URL)
data = response.json()
print(f"ThingSpeak yanıtı: {data}") # JSON yanıtını kontrol etmek için yazdır

# JSON'dan field1 (Nem Seviyesi) ve field2 (Su Seviyesi) verilerini al
feeds = data.get("feeds", [])
if feeds: # Eğer veri varsa
    soil_moisture = int(feeds[0].get("field1", 0)) # field1 (Nem Seviyesi)
    water_level = int(feeds[0].get("field2", 0)) # field2 (Su Seviyesi)

    print(f"Toprak Nemi: {soil_moisture}, Su Seviyesi: {water_level}")

# Nem seviyesine göre durumu belirle
if soil_moisture < 2200:
    state = "wet"
elif 2200 <= soil_moisture < 3900:
    state = "moist"
else:
    state = "dry"

# Durum değiştiyse ilgili ses dosyasını çal
if state != previous_state:
    print(f"Durum değişti: {state}. Ses çalınıyor...")
    pygame.mixer.music.load(SOUNDS[state]) # Uygun ses dosyasını yükle
    pygame.mixer.music.play() # Ses dosyasını çal
    while pygame.mixer.music.get_busy():
        time.sleep(0.1) # Ses çalınırken bekle
    previous_state = state
else:
    print("Veri alınamadı!")

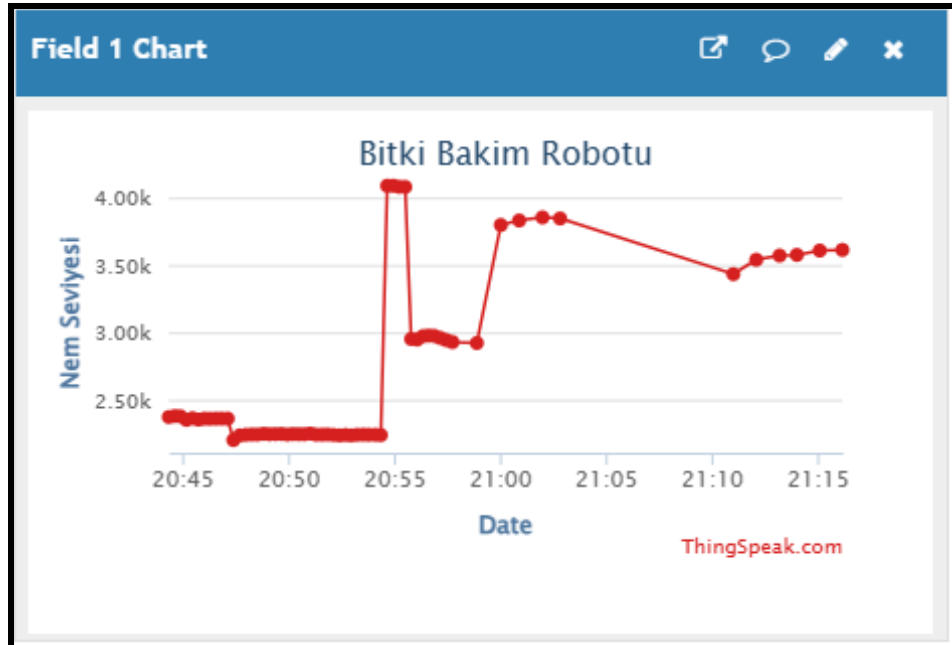
except Exception as e:
    print(f"Hata oluştu: {e}")

time.sleep(2) # 2 saniyede bir kontrol
```

Python Kodu Çıktısı :

```
Run main x
Toprak Nemi: 3615, Su Seviyesi: 159
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3615, Su Seviyesi: 159
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3615, Su Seviyesi: 159
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3615, Su Seviyesi: 159
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3615, Su Seviyesi: 159
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3624, Su Seviyesi: 146
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3624, Su Seviyesi: 146
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3624, Su Seviyesi: 146
ThingSpeak yanıtı: {'channel': {'id': 2797072, 'name': 'Bitki Bakim Robotu'}}
Toprak Nemi: 3624, Su Seviyesi: 146
```

ThingSpeak'e Gönderilen Verilerin Grafikleri:



Field 2 Chart



Bitki Bakim Robotu



ThingSpeak.com