

Outfit recommendation system

Introducere

- Sistemul are ca scop recomandarea de haine în funcție atât de datele citite de senzorii de temperatură, umiditate și presiune, cât și de datele oferite de un API de vreme. Sistemul este conectat la o rețea WiFi și îi este dat un oras ca locație iar apoi în funcție de aceste informații este afișată o recomandare pentru haine.
- Ideea acestui proiect a venit în urma dorinței mele de a eficientiza rutina de dimineață.
- Consider că toată lumea are de beneficiat în urma acestui proiect deoarece majoritatea ne putem scurta timpul petrecut pregătindu-ne de dimineață iar parametrii precum hainele, preferințe vestimentare și de temperatură pot fi modificate după preferințe.

Descriere generală

Pentru a obține date despre vreme atât în interiorul încăperii cât și afară, microcontroller-ul primește date de la senzori și face o cerere către un API de vreme ce oferă date despre vreme în decursul zilei. În urma analizării acestor date este realizată o recomandare pe baza unui algoritm și a preferințelor predefinite iar recomandarea este afișată pe ecranul LCD.



Hardware Design

Listă componente

- 1 x Placă de dezvoltare Arduino Uno
- 1 x Senzor de temperatură și umiditate DHT22
- 1 x Senzor de Presiune BMP280 GY
- 1 x Breadboard 830 puncte
- 1 x Display LCD 1601
- 1 x Modul Wifi ESP8266 ESP-01
- 75 x Fire
- 5 x Butoane

Schema Electrică

Modulul de WiFi ESP-01 este alimentat din pinul de 3.3V de pe Arduino, CPD este de asemenea conectat la 3.3V. RX și TX de pe modul sunt conectați la pinii GPIO 2 și 3.

Display-ul LCD cu modul I2C este alimentat la pinul de 5V al plăcuței iar pinii SDA și SCL sunt conectați la pinii A4 și respectiv A5 de pe Arduino.

Pentru senzorul de temperatură și umiditate DHT-22 am conectat pinul de date la pinul GPIO 7 și l-am alimentat cu 3.3V.

Senzorul BMP280GY de presiune este alimentat tot de 3.3V și este conectat folosind I2C la pinii SDA și SCL de pe plăcuță.



Software Design

Biblioteci și mediu de dezvoltare

Pentru mediul de dezvoltare am ales Arduino IDE datorită compatibilității cu placa de dezvoltare Arduino UNO, modulul ESP8266-01 și celelalte componente.

Ca biblioteci software am folosit pentru senzori **Adafruit BMP280 Library** și **DHT Sensor Library** deoarece sunt implementările producătorului pentru componente. Pentru display am folosit **LiquidCrystalIO** din același motiv.

Pentru parsarea răspunsului în format JSON am folosit **ArduinoJSON** deoarece este biblioteca cu cea mai vastă documentație.

În ceea ce privește modulul ESP8266-01, am întâmpinat dificultăți în programarea modulului și comunicarea cu Arduino UNO-ul la care sunt conectate restul componentelelor.

Am ales biblioteca **WiFiEspAT** ce comunică folosind SoftwareSerial cu modulul wifi, nefiind necesară programarea directă a acestuia.

Prezentare generală

- **Garments.h**: Am definit implemetările pentru toate tipurile de articole vestimentare și pentru clasa Outfit.
- **proiect_uno.ino**: Am instanțiat obiecte globale pentru senzori, display și esp8266.
- **setup()**: am apelat metoda `begin()` a obiectelor și am verificat dacă s-au instanțiat cu succes.
- **loop()**: În urma citirii datelor de la senzori și de la modulul esp8266, aleg din lista de haine instanțiate cele mai bune variante în funcție de vreme și de preferințele menționate anterior. Ulterior afisez alegerea pe display într-un while.

Laboratoare folosite

- Am folosit laboratorul de **GPIO** la conectarea cu senzorul DHT-22.
- Laboratorul de **I2C** mi-a fost util în utilizarea atât a display-ului cât și a senzorului de presiune deoarece ulterior le-am legat în paralel. Pentru a afla adresa senzorului de presiune am folosit tool-ul i2cdetect.
- Laboratorul de **UART** a fost esențial pentru comunicarea cu modulul de WiFi și în procesul extensiv de debugging.

Rezultate Obținute

[Prezentare proiect](#)

Concluzii

Pot spune că partea de lucrat cu de senzori și display-ul a fost o experiență plăcută deoarece bibliotecile existente oferă o platformă excelentă pentru realizarea oricărui fel de proiect.

În schimb alegerea de a lucra cu modulul ESP8266-01 s-a dovedit a fi o mare greșeală deoarece opțiunile de a interacționa cu modulul sunt mult mai complicate decât m-aș fi așteptat, prin necesitatea de a programa modulul apoi a schimba cablajul. În final am recurs la comunicarea prin Serial deoarece am considerat că ar fi cea mai accesibilă metodă în absența unui programator dedicat pentru placă.

[Video cu explicații](#)

Download

[codandreibaroiu.zip](#)

Jurnal

Am adăugat rezultate proiectului.

Bibliografie/Resurse

Resurse hardware

- https://arduinogetstarted.com/tutorials/arduino-dht22?utm_content=cmp-true
- <https://circuitdigest.com/microcontroller-projects/interfacing-bmp280-sensor-with-arduino>
- <https://www.instructables.com/TutorialI2C-1602/>
- <https://www.instructables.com/Getting-Started-With-the-ESP8266-ESP-01/>

Resurse software

- <https://github.com/jandrassy/WiFiEspAT?tab=readme-ov-file#limitations>
- https://docs.espressif.com/projects/esp-at/en/release-v2.2.0.0_esp8266/AT_Command_Set/index.html
- <https://arduinojson.org/v7/>

[Export to PDF](#)

From:
<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2024/avaduva/andrei.baroiu>



Last update: **2024/05/25 17:37**