**(날씨를 부탁해) 개발보고서**

**2학년 YA반 이름:** 엄지희 **날자: 2021. 6. 13.**

|  |
| --- |
| **1. 개발목표와 특징** |

*(각 팀에서 개발하고자 하는 제안시스템의 특징을 반영한 개발목표를 기술)*

|  |
| --- |
| 날씨 정보(온/습도)를 받아 좀 더 편리하게 외출준비를 할 수 있게끔 만들어주는 시스템.  날씨에 따라 알람 소리가 바뀌어 아침에 일어날 때 오늘 비가 오는지, 오지 않을 확률이  큰지 알 수 있고, 온도에 따라 입을 옷을 추천해준다. 예를 들어, 날이 더우면 ‘반팔에 얇은 바지를 입으세요!’ 라는 문장을 출력해준다. 날이 추우면 ‘칼바람이 부니 장갑과 목도리 필수’라는 문장을 출력한다. 비가 오면 ‘우산을 챙기세요!’를 출력한다. |

|  |
| --- |
| **2. 개발필요성** |

*(왜 본 시스템 개발이 필요한지 기술적 환경 및 수요 배경을 중심으로 기술)*

|  |
| --- |
| 일반적으로 사람들은 매일 아침에 날씨 앱에 들어가서 온도와 날씨, 습도를 체크하고, 거기에 맞춰 입을 옷을 찾느라 또 시간을 허비한다. 매번 이런 루틴을 반복해가며 옷을 찾기 힘들어서 아예 기능이 합쳐져 있도록 더 편리하게 만들어보려고 한다. 날씨가 15도이면 얇은 반팔을 입는 것을 추천하고, 아침에 일어나자마자 굳이 날씨 앱에 들어갈 필요 없이 알람만으로 오늘 비가 오는지 오지 않을지를 확인하는 등 날씨 앱에서는 볼 수 없는 특별한 기능을 제공한다. 자주 덜렁대는 사람에게 추천한다. |

|  |
| --- |
| **3. H/W 구현** |

*그림, 사진 등으로 설명*

|  |
| --- |
| 온습도 센서를 이용하여 현재 방 안의 온도와 습도를 측정한다. 이더넷 네트워킹을 이용하여 기상청에서 날씨 정보(최고/최저기온 날씨, 습도 등)를 받아온다. 온습도에 따라 알맞은 알람 소리를 부저 센서를 통해 출력한다. 예를 들어, 오늘 비가 오면 곰 세마리, 오늘 비가 오지 않으면 학교 종이 땡땡땡을 출력한다. 이때, 시리얼 LCD 디스플레이로 온도와 습도 등 날씨 정보를 출력한다. 알람이 끝나면 오늘 무슨 옷을 입으면 좋을지 시리얼 LCD로 출력한다. 일교차와 온도 습도 등을 종합적으로 고려해 옷을 추천한다. 날씨가 흐리고 비가 올 것 같으면 우산을 챙길 것을 추천한다. |

|  |
| --- |
| **4. S/W 구현** |

(스케치 코드 첨부, 가져온 소스 표시)

/\*

Repeating Web client

This sketch connects to a a web server and makes a request

using a Wiznet Ethernet shield. You can use the Arduino Ethernet shield, or

the Adafruit Ethernet shield, either one will work, as long as it's got

a Wiznet Ethernet module on board.

This example uses DNS, by assigning the Ethernet client with a MAC address,

IP address, and DNS address.

Circuit:

\* Ethernet shield attached to pins 10, 11, 12, 13

created 19 Apr 2012

by Tom Igoe

modified 21 Jan 2014

by Federico Vanzati

http://www.arduino.cc/en/Tutorial/WebClientRepeating

This code is in the public domain.

\*/

#include <SPI.h>

#include <Ethernet.h>

#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x3F, 16, 2); // 안되면 주소를 0x27->0x3F로 변경

int speakerPin = 8;

#define C4 262

#define CS4 277

#define D4 294

#define DS4 311

#define E4 330

#define F4 349

#define FS4 370

#define G4 392

#define GS4 415

#define A4 440

#define AS4 466

#define B4 494

#define C5 523

// assign a MAC address for the ethernet controller.

// fill in your address here:

byte mac[] = { 0xDE, 0xAD, 0xBE, 0xEF, 0xFE, 0xED };

// Set the static IP address to use if the DHCP fails to assign

IPAddress ip(192, 168, 219, 102);

IPAddress myDns(61, 41, 153, 2);

// initialize the library instance:

EthernetClient client;

char server[] = "www.kma.go.kr"; // also change the Host line in httpRequest()

//IPAddress server(64,131,82,241);

unsigned long lastConnectionTime = 0; // last time you connected to the server, in milliseconds

const unsigned long postingInterval = 10\*1000; // delay between updates, in milliseconds

void setup() {

// You can use Ethernet.init(pin) to configure the CS pin

//Ethernet.init(10); // Most Arduino shields

//Ethernet.init(5); // MKR ETH shield

//Ethernet.init(0); // Teensy 2.0

//Ethernet.init(20); // Teensy++ 2.0

//Ethernet.init(15); // ESP8266 with Adafruit Featherwing Ethernet

//Ethernet.init(33); // ESP32 with Adafruit Featherwing Ethernet

// start serial port:

Serial.begin(9600);

while (!Serial) {

; // wait for serial port to connect. Needed for native USB port only

}

// start the Ethernet connection:

Serial.println("Initialize Ethernet with DHCP:");

if (Ethernet.begin(mac) == 0) {

Serial.println("Failed to configure Ethernet using DHCP");

// Check for Ethernet hardware present

if (Ethernet.hardwareStatus() == EthernetNoHardware) {

Serial.println("Ethernet shield was not found. Sorry, can't run without hardware. :(");

while (true) {

delay(1); // do nothing, no point running without Ethernet hardware

}

}

if (Ethernet.linkStatus() == LinkOFF) {

Serial.println("Ethernet cable is not connected.");

}

// try to congifure using IP address instead of DHCP:

Ethernet.begin(mac, ip, myDns);

Serial.print("My IP address: ");

Serial.println(Ethernet.localIP());

} else {

Serial.print(" DHCP assigned IP ");

Serial.println(Ethernet.localIP());

}

// give the Ethernet shield a second to initialize:

delay(1000);

// initialize the LCD

lcd.begin();

lcd.clear(); //화면 초기화

lcd.backlight();

lcd.print("Hello, world!");

}

void loop() {

// if there's incoming data from the net connection.

// send it out the serial port. This is for debugging

// purposes only:

String a[3];

int i=0;

String tmp; // 온도

String wfEn; // 날씨

String reh; // 습도

String tmp\_str;

int tmp\_index;

int tmp\_length;

String rec, rec1, rec2; // 옷차림 결정

if (client.connect(server, 80)) { //starts client connection, checks for connection

client.println("GET /wid/queryDFSRSS.jsp?zone=1156063000 HTTP/1.1");

client.println("Host: www.kma.go.kr");

client.println("Connection: close");

client.println();

delay(10);

while(client.available()){

String line = client.readStringUntil('\n');

i= line.indexOf("</temp>");

if(i>0){

tmp\_str="<temp>";

tmp = line.substring(line.indexOf(tmp\_str)+tmp\_str.length(),i);

Serial.println(tmp);

rec = determineRec(tmp);

tmp\_index = rec.indexOf(","); // 첫 번재 콤마 위치

tmp\_length = rec.length();

rec1 = rec.substring(0, tmp\_index);

rec2 = rec.substring(tmp\_index + 1, tmp\_length);

Serial.println(rec1);

Serial.println(rec2);

}

i= line.indexOf("</wfEn>");

if(i>0){

tmp\_str="<wfEn>";

wfEn = line.substring(line.indexOf(tmp\_str)+tmp\_str.length(),i);

if(wfEn.equals("rain")) {

Serial.println("It's a rainy day");

Serial.println("Take an umbrella");

bearSong();

}

else {

Serial.print("Today's weather");

Serial.println(" is " + wfEn);

schoolSong();

}

}

i= line.indexOf("</reh>");

if(i>0){

tmp\_str="<reh>";

reh = line.substring(line.indexOf(tmp\_str)+tmp\_str.length(),i);

Serial.println(reh);

Serial.println();

break;

}

}

//(1) 날씨 출력

lcd.clear();

lcd.setCursor(0, 0); // 맨 위, 첫 번째 줄

lcd.print("T:");

lcd.print(tmp);

lcd.print((char)223);

lcd.print("C ");

lcd.print(" H:");

lcd.print(reh);

lcd.print("%");

lcd.setCursor(0, 1); // 맨 아래, 두 번째 줄

lcd.print("Cloudy: ");

lcd.print(wfEn);

delay(3000);

//(2) 옷 추천

lcd.clear();

lcd.setCursor(0, 0); // 맨 위, 첫 번째 줄

lcd.print(rec1);

lcd.setCursor(0, 1); // 맨 아래, 두 번째 줄

lcd.print(rec2);

delay(3000);

//(3) 우산 챙길지말지

if(wfEn.equals("rain")) {

lcd.clear();

lcd.setCursor(0, 0); // 맨 위, 첫 번째 줄

lcd.print("It's a rainy day");

lcd.setCursor(0, 1); // 맨 아래, 두 번째 줄

lcd.print("Take an umbrella");

}

else {

lcd.clear();

lcd.setCursor(0, 0); // 맨 위, 첫 번째 줄

lcd.print("Today's weather");

lcd.setCursor(0, 1); // 맨 아래, 두 번째 줄

lcd.print("is " + wfEn);

}

delay(3000);

}

// if ten seconds have passed since your last connection,

// then connect again and send data:

if (millis() - lastConnectionTime > postingInterval) {

httpRequest();

}

}

// this method makes a HTTP connection to the server:

void httpRequest() {

// close any connection before send a new request.

// This will free the socket on the WiFi shield

client.stop();

// if there's a successful connection:

if (client.connect(server, 80)) { //starts client connection, checks for connection

client.println("GET /wid/queryDFSRSS.jsp?zone=1156063000 HTTP/1.1");

client.println("Host: www.kma.go.kr");

client.println("Connection: close");

client.println();

// note the time that the connection was made:

lastConnectionTime = millis();

} else {

// if you couldn't make a connection:

Serial.println("connection failed");

}

}

// 어떤 옷을 입을지 결정

String determineRec(String temperature) {

String rec;

int tmpt;

tmpt = temperature.toInt();

if(tmpt >= 28) {

rec = "Rec: sleeveless,shorts & a dress";

// 민소매, 반팔, 반바지, 원피스 추천

}

if(tmpt >= 23 && tmpt < 28) {

rec = "Rec: thin shirts,and cotton pants";

// 반팔, 얇은 셔츠, 반바지, 면바지 추천

}

if(tmpt >= 20 && tmpt < 23) {

rec = "Rec: long sleeve,cardigan & jeans";

// 얇은 가디건, 긴팔, 면바지, 청바지 추천

}

if(tmpt >= 17 && tmpt < 20) {

rec = "Rec: thin knit,mantoman & jeans";

// 얇은 니트, 맨투맨, 가디건, 청바지 추천

}

if(tmpt >= 12 && tmpt < 17) {

rec = "Rec: stockings,jacket & jeans";

// 자켓, 가디건, 야상, 스타킹, 청바지, 면바지 추천

}

if(tmpt >= 9 && tmpt < 12) {

rec = "Rec: trench coat,knit and jeans";

// 자켓, 트렌치코트, 야상, 니트, 청바지, 스타킹 추천

}

if(tmpt >= 5 && tmpt < 9) {

rec = "Rec: coat/hittec,knit & leggings";

// 코트, 가죽자켓, 히트텍, 니트, 레깅스 추천

}

if(tmpt < 5) {

rec = "Rec: padding,nanny products";

// 패딩, 두꺼운 코트, 목도리, 기모제품 추천

}

return rec;

}

// 곰 세마리 알람노래

void bearSong() {

tone(speakerPin, C4); //곰

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //세

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //마

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //리

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //가

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //한

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //집

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //에

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //있

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //어

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //아

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //빠

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //곰

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //엄

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //마

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //곰

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //애

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //기

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //곰

delay(950);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //아

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //빠

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //곰

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //은

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //뚱

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //뚱

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //해

delay(950);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //엄

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //마

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //곰

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //은

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //날

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //씬

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //해

delay(950);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //애

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //기

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //곰

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //은

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //너

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //무

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //귀

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, A4); //여

delay(200);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //워

delay(950);

noTone(speakerPin);

delay(50);

tone(speakerPin, C5); //으

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //쓱

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C5); //으

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //쓱

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //잘

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, D4); //한

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //다

delay(950);

noTone(speakerPin);

delay(50);

noTone(speakerPin);

delay(1800);

}

// 학교 종이 땡땡땡 알람노래

void schoolSong() {

tone(speakerPin, G4); //학

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //교

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, A4); //종

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, A4); //이

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //땡

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //땡

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //땡

delay(950);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //어

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //서

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //모

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //이

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, D4); //자

delay(1450);

noTone(speakerPin);

delay(50);

noTone(speakerPin);

delay(500);

tone(speakerPin, G4); //선

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //생

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, A4); //님

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, A4); //이

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //우

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //리

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //를

delay(950);

noTone(speakerPin);

delay(50);

tone(speakerPin, G4); //기

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //다

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, D4); //리

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, E4); //신

delay(450);

noTone(speakerPin);

delay(50);

tone(speakerPin, C4); //다

delay(1450);

noTone(speakerPin);

delay(50);

noTone(speakerPin);

delay(500);

noTone(speakerPin);

delay(1800);

}

|  |
| --- |
| **5. 구현 소감** |

(개발중 있었던 일과 느낌)

IoT 프로젝트를 내 손으로 스스로 만들어나가는 과정이 무척 재미있었고 어려웠지만 그만큼 완성되었을 때 보람을 느낄 수 있는 시간이었다. 소스코드를 이렇게까지 길게 작성해본 것은 거의 처음이라 제작하는 데 조금의 어려움이 있었지만 그래도 처음 치고는 이정도면 잘했다 잘 헤쳐나갔다 하고 스스로를 칭찬해주고싶다.

|  |
| --- |
| **6. 향후 보완 사항** |

알람을 완벽히 구현하지 못한 것이 아쉽다.

알람 소스 코드를 찾았음에도 불구하고 둘을 합치는 과정이 너무 어려웠다. 알람의 시간 초와 이더넷에서 가져오는 소스코드를 가져오는(리붓하는) 타이머에 충돌이 있는 것 같았는데, 내 실력으로는 해결하기 어려워서 알람시계를 만드는 소스코드와 날씨 추천기능을 따로 제작할 수밖에 없었다. 그 점이 조금 아쉽다. 다음에는 그 점을 더 보완하고 싶다.

|  |
| --- |
| **7. 참고자료 (참고한 자료, 사이트)** |

아두이노 이더넷 웹서버 만들기

<https://raspberrystory.tistory.com/48?category=908276>

아두이노 이더넷 쉴드 설정하기

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=damtaja&logNo=140206137375>

Failed to Configure Ethernet Using DHCP

<https://forum.arduino.cc/t/failed-to-configure-ethernet-using-dhcp-please-help/370494>

아두이노 인터넷 사용 – 이더넷과 웹브라우저

<https://it-g-house.tistory.com/entry/Arduino-%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-%EC%9D%B8%ED%84%B0%EB%84%B7-%EC%82%AC%EC%9A%A9-%EC%9D%B4%EB%8D%94%EB%84%B7Ethernet%EA%B3%BC-%EC%9B%B9%EB%B8%8C%EB%9D%BC%EC%9A%B0%EC%A0%80>

아두이노 시리얼 통신 데이터를 엑셀에 저장/연동하기

<https://it-g-house.tistory.com/entry/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8Arduino%EB%A5%BC-%EC%82%AC%EC%9A%A9%ED%95%9C-%EB%8D%B0%EC%9D%B4%ED%84%B0%EB%A1%9C%EA%B1%B0Datalogger-%EB%A7%8C%EB%93%A4%EA%B8%B0?category=736079>

아두이노 nodemcu를 이용해서 기상청에서 날씨정보 가져오기

<https://ilikesan.com/entry/%EC%95%84%EB%91%90%EC%9D%B4%EB%85%B8-nodemcu-%EC%9D%84-%EC%9D%B4%EC%9A%A9%ED%95%B4%EC%84%9C-%EA%B8%B0%EC%83%81%EC%B2%AD%EC%97%90%EC%84%9C-%EB%82%A0%EC%94%A8-%EA%B0%80%EC%A0%B8%EC%98%A4%EA%B8%B0>

OpenWeatherMap API 키 받기

<https://diy-project.tistory.com/5>

아두이노로 날씨 정보 출력하기

<https://diy-project.tistory.com/73>

OpenWeather

<https://openweathermap.org/city/1835848>

아두이노 이더넷모듈의 사용 – 설정, 웹서버, …

<https://deneb21.tistory.com/284>

아두이노 WI-FI 날씨 정보 가져오기

<https://blog.naver.com/compass1111/221133450681>

아두이노 이더넷쉴드로 기상청 날씨데이터 가져오기

<https://blog.naver.com/PostView.nhn?blogId=mkhee77&logNo=221458277679>

날씨정보를 분류하여 표시하기

<https://blog.naver.com/PostView.nhn?blogId=mkhee77&logNo=221458380530&parentCategoryNo=&categoryNo=&viewDate=&isShowPopularPosts=false&from=postView>

문자열 분리하기

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=okarduino&logNo=220762181563>

문자열 분리하기 - String 객체 이용

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=okarduino&logNo=220768107510>

아두이노 컴파일 에러메시지 해결

<https://kocoafab.cc/tutorial/view/268>

아두이노 String 객체 설명

<https://m.blog.naver.com/PostView.naver?isHttpsRedirect=true&blogId=yuyyulee&logNo=220305968049>

아두이노 String 클래스

<https://blog.daum.net/rockjjy99/2699>

How do I compare string in java?

<https://dzone.com/articles/how-do-i-compare-strings-in-java>

Arduino - String()

<https://www.arduino.cc/reference/en/language/variables/data-types/stringobject/>

아두이노 Serial.read()에서 문자열 비교 방법

<https://pythonq.com/so/serial-port/811617>

시리얼로 문자 입력받아 문자 비교

<https://lovelymanju.tistory.com/65>

C++ 아두이노 버튼을 통한 루프 정지

<https://www.python2.net/questions-763441.htm>

아두이노 LCD와 피에조부저, 스위치를 사용하여 알람시계 만들기

<https://kocoafab.cc/make/view/144>