**Weather Station**

**Team name**: SkyBast

**Team members**:

* Cazacu Iulia
* Ghițun Ștefania
* Nechifor Radu

**Hardware components design**:

The hardware components we have used are:

- UDOO NEO Basic[1]

# - Temperature and Humidity Sensor DHT22[2]

# - Temperature and Pressure Sensor BMP180[3]

# - Light Sensor Brick[4]

# - HC-05 Serial Bluetooth[5]

# - Graphic LCD 84x48 - Nokia 5110[6]

# C:\Users\HAL9000\Desktop\b.jpg

# Figure 1. Wiring Diagram of The Weather Station

**Software components design:**

For development, we used Arduino IDE.

Libraries used:

- DHT22 library (incorporated in UDOO NEO Arduino examples)

- BME280 library

- LCD library

Source cod:

#include <stdint.h>

#include "SparkFunBME280.h"

#include "Wire.h"

#include "SPI.h"

#include <dht.h>

#include "lcd.h"

dht DHT;

#define DHT22\_PIN 13

int LED=8;

BME280 mySensor;

struct

{

uint32\_t total;

uint32\_t ok;

uint32\_t crc\_error;

uint32\_t time\_out;

uint32\_t connect;

uint32\_t ack\_l;

uint32\_t ack\_h;

uint32\_t unknown;

} stat = { 0,0,0,0,0,0,0,0};

void setup() {

pinMode(LED, OUTPUT);

mySensor.settings.commInterface = I2C\_MODE;

mySensor.settings.I2CAddress = 0x76;

mySensor.settings.runMode = 3;

mySensor.settings.tStandby = 0;

mySensor.settings.filter = 0;

mySensor.settings.tempOverSample = 1;

mySensor.settings.pressOverSample = 1;

mySensor.settings.humidOverSample = 1;

mySensor.begin();

LcdInitialise();

LcdClear();

Serial0.begin(9600);

Serial.begin(9600);

}

void loop(void) {

int chk = DHT.read22(DHT22\_PIN);

int valoareIluminare = analogRead(0);

Serial0.print("Indice lumina: ");

Serial0.println(valoareIluminare);

float h;

h=-100 ;

float t;

t=-100 ;

float tep;

tep=-100;

float pres;

//citire date

h= DHT.humidity;

t = DHT.temperature;

tep=mySensor.readTempC();

pres=mySensor.readFloatPressure()\*0.00750061683;

//afisare date citite

if(h!=-100&&t!=-100&&tep!=-100)

{

gotoXY(0,0);

LcdString ("Light:");

afisint(valoareIluminare,5);

gotoXY(3,2);

LcdString("out:");

Serial0.print("Temperature out:");

Serial0.print(tep);

Serial0.println("\*C");

afistemp(tep,4);

LcdString("\*C");

gotoXY(3,1);

LcdString("in:");

afistemp(t,4);

LcdString("\*C");

Serial0.print("Temperature in:");

Serial0.print(t);

Serial0.println("\*C");

gotoXY(0,3);

LcdString("Hum.: ");

afisint(h,3);

LcdString("%");

Serial0.print("Humditi :");

Serial0.print(h);

Serial0.println("%");

gotoXY(0,4);

LcdString("Pressure:");

gotoXY(4,5);

afistemp(pres,5);

LcdString("mmHg");

Serial0.print("Pressure:");

Serial0.print(pres);

Serial0.println("mmHg");

Serial0.println();

Serial0.println();

Serial0.println();

}

else

{

gotoXY(20,1);

LcdString("Eroare");

gotoXY(30,2);

LcdString("Citire");

gotoXY(20,3);

LcdString("Date!!");

}

digitalWrite(LED, LOW);

delay(1000);

digitalWrite(LED, HIGH);

}

**User guide:**

The station is compact and easy to use and set up. It comes with its own charger so you just need to plug that in any regular house socket. The device has an outside sensor that must be placed outside the house (or in any other particular room or place that the user wishes to measure the temperature). The user be warned that the outside sensor is wired so they must treat with care and not forget to check the connection when moving the sensor.

The LCD will display data on outside temperature, inside temperature, pressure and light intensities, all with units specified. The LCD usually updates its display every one second.

**Implementation issues:**

Our first issue arose when we realised only but one could use the microcontroller due to missing drivers. We tried to install the rest, but only succeeded partially so in the end we simply decided for the teammate that had Windows 10 and had all the working drivers to keep the microcontroller.

We tried some test examples and everything seemed to be fine, but then we realised the microcontroller wasn't retaining in its memory the code and after a while even a error appeared whenever we tried to upload anything new. After we went over the tutorials and set up the A9 M4 communication everything went smoothly.

Another problem we faced was when we needed to build an app for the project. Initially we couldn't read the Arduino serial, but it worked after we changed that to Serial1. And we managed to get data readings on the app, but they were all mingled and displaying unknown characters. The tried displaying just one, we tried the Udoo and App Inventor open source project, but in the end we couldn't rectify the issue in time.

**Bibliography:**

[1] UDOO NEO Documentation

< https://www.udoo.org/docs-neo/Introduction/Introduction.html>

# [2] DHT22 Datasheet

<https://www.sparkfun.com/datasheets/Sensors/Temperature/DHT22.pdf>

[3] BMP180 Datasheet

<https://cdn-shop.adafruit.com/datasheets/BST-BMP180-DS000-09.pdf>

# [4] Light Sensor Brick Datasheet

< https://www.robofun.ro/docs/ELPT15-21C.pdf>

[5] HC-05 Serial Bluetooth Datasheet

< http://www.electronicaestudio.com/docs/istd016A.pdf>

[6] Graphic LCD 84x48 - Nokia 5110 Datasheet

< https://www.sparkfun.com/datasheets/LCD/Monochrome/Nokia5110.pdf>