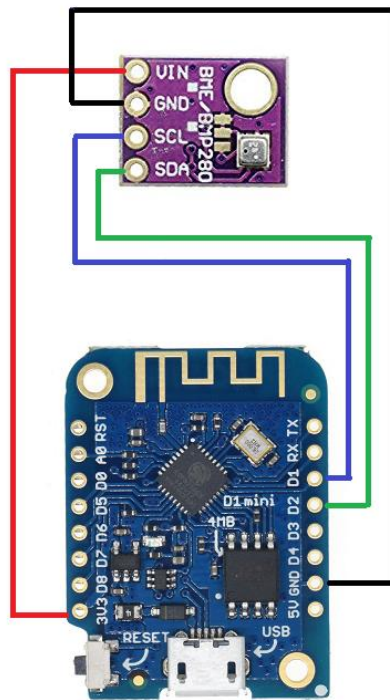
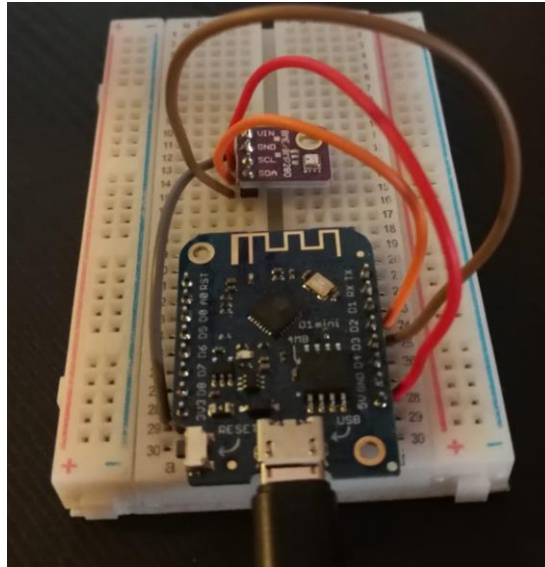


## Sulautetut prosessorijärjestelmät 2018 harjoitustyön dokumentointi

Harjoitustyöksi valitsimme IoT-workshopin sääaseman. Olimme läsnä workshopin ensimmäisellä kerralla ja jatkoimme sitten työskentelyä omatoimisesti. Käytimme työssämme alustana ESP8266-mikrokontrolleria, joka mahdollistaa yksinkertaisen WiFi-yhteyden ja on Arduino-yhteensopiva. Sensori siirtää dataa mikrokontrollerille, joka lähettää sen JSON-muodossa verkkosivulle, joka esittää saadun datan. Mikrokontrolleri on ohjelmoitu Arduino IDE -ohjelmistolla käyttäen C++ -kieltä. Verkkosivun backend on tehty TypeScriptillä ja frontend Reactilla.



Yläpuolella olevasta kytkentäkaaviosta näkee toteutusta varten tarvittavat liitännät. Tässä punainen on virtakytkentä, musta maakytkentä, vihreä kuljettaa dataa ja sininen synkronoi datan siirron. Sensori käyttää tiedonsiirtoon I2C-standardin väylää, jota mikrokontrollerin portit D1 ja D2 tukevat.



Kuva kytkennästä.

```

iot-weatherstation | Arduino 1.8.8
File Edit Sketch Tools Help

iot-weatherstation

WiFi.mode(WIFI_STA);
WiFiMulti.addAP(SSID, PASSWD);
}

void loop() {
  StaticJsonBuffer<capacity> jb;
  char output[128];
  JsonObject& data = jb.createObject();

  data["name"] = NAME;
  data["temperature"] = sensor.readTemperature();
  data["pressure"] = sensor.readPressure();
  data["humidity"] = sensor.readHumidity();

  data.printTo(output);
  data.prettyPrintTo(Serial);
  Serial.println();

  if(WiFiMulti.run() == WL_CONNECTED){

    HTTPClient http;

    Serial.println("[HTTP] begin...");

    http.begin(IP);
    http.addHeader("Content-Type", "application/json");

    int httpcode = http.POST(output);

    if(httpcode == HTTP_CODE_OK){
      Serial.println("Working");
    }else{
      Serial.println("Something went wrong");
      Serial.println(httpcode);
    }

    http.end();
  }

  delay(10000);
}

```

Käytimme mikrokontrollerin koodissa valmiita kirjastoja WiFi- ja HTTP-yhteyden muodostamiseen, sekä datan tallentamiseen JSON-muodossa. Aluksi yhdistetään annettuun WiFi-verkkoon, jonka jälkeen tallennetaan sensorin data (mikrokontrollerin nimi, lämpötila, ilmanpaine ja -kosteus). Sen jälkeen pyritään tekemään HTTP-yhteys annettuun IP-osoitteeseen. Käytimme työssämme localhostia osoitteena, koska halusimme pitää datan sisäverkossa. Datan lähettäminen IP-osoitteeseen toistetaan 10 sekunnin välein.

26.2.2019

```
[
  - {
    sensorname: "Saa-asema",
    temperature: 26.23,
    pressure: 102209,
    humidity: 39.4834,
    timestamp: "2019-02-26T14:25:40.992Z"
  },
  - {
    sensorname: "Saa-asema",
    temperature: 26.17,
    pressure: 102214.3,
    humidity: 39.6748,
    timestamp: "2019-02-26T14:25:30.922Z"
  },
  - {
    sensorname: "Saa-asema",
    temperature: 26.18,
    pressure: 102212.8,
    humidity: 39.59961,
    timestamp: "2019-02-26T14:25:20.845Z"
  },
  - {
    sensorname: "Saa-asema",
    temperature: 26.23,
    pressure: 102217.1,
    humidity: 39.38477,
    timestamp: "2019-02-26T14:25:10.784Z"
  },
  - {
    sensorname: "Saa-asema",
    temperature: 26.27,
    pressure: 102211.9,
    humidity: 39.3457,
    timestamp: "2019-02-26T14:25:00.713Z"
  }
]
```

Tässä localhostilta löytyvä data JSON-muodossa. JSON on parsittu luettavammaksi selainliitännäisellä.

localhost:3000

Sensorit				
Nimi	Ensimmäinen päivitys		Viimeisin päivitys	
Saa-asema	26.2.2019 klo 16.05.21		26.2.2019 klo 16.26.51	
Mittaukset				
Aika	Sensori	Lämpötila	Ilmanpaine	Ilmankosteus
26.2.2019 klo 16.26.51	Saa-asema	26.11 °C	102216.1 hPa	39.67969 %
26.2.2019 klo 16.26.41	Saa-asema	26.14 °C	102216.5 hPa	39.59473 %
26.2.2019 klo 16.26.31	Saa-asema	26.14 °C	102220.2 hPa	39.72656 %
26.2.2019 klo 16.26.21	Saa-asema	26.12 °C	102218.8 hPa	39.77832 %
26.2.2019 klo 16.26.11	Saa-asema	26.1 °C	102213.5 hPa	39.66895 %

Tässä kuva verkkosivusta, joka esittää saatua dataa. Backend käsittelee mikrokontrollerilta saadun datan tallentamalla sen SQLite-tietokantaan. Backend hoitaa myös datan hakemisen tietokannasta frontendin käyttöön. Frontend esittää datan selkeälukuisena lopullisella verkkosivulla. Erittelimme sensorit, jotta voisimme vastaisuudessa yhdistää useamman sensorin samalle verkkosivulle. Rajoitimme mittausdatan viiteen otokseen, jotta käyttöliittymä pysyisi käyttäjäystävällisenä.

Harjoitustyön lähdekoodi löytyy osoitteesta: <https://github.com/PetroSilenius/iot-weatherstation>. Mikrokontrollerin koodi löytyy tiedostosta: iot-weatherstation.ino  
Backendin koodi server-kansiosta ja frontend löytyy client-kansiosta.

26.2.2019

Ohjelman ajamiseksi kytke mikrokontrolleri USB-kaapelilla tietokoneeseen. Täydennä iot-weatherstation.ino tiedostoon WiFi-verkon tiedot ja tarkista, että IP-osoite on oikein. Tämän jälkeen lataa ohjelma mikrokontrolleriin. Backend ja frontend käynnistyvät omista kansioistaan ajamalla komennot "yarn install" ja "yarn start".