

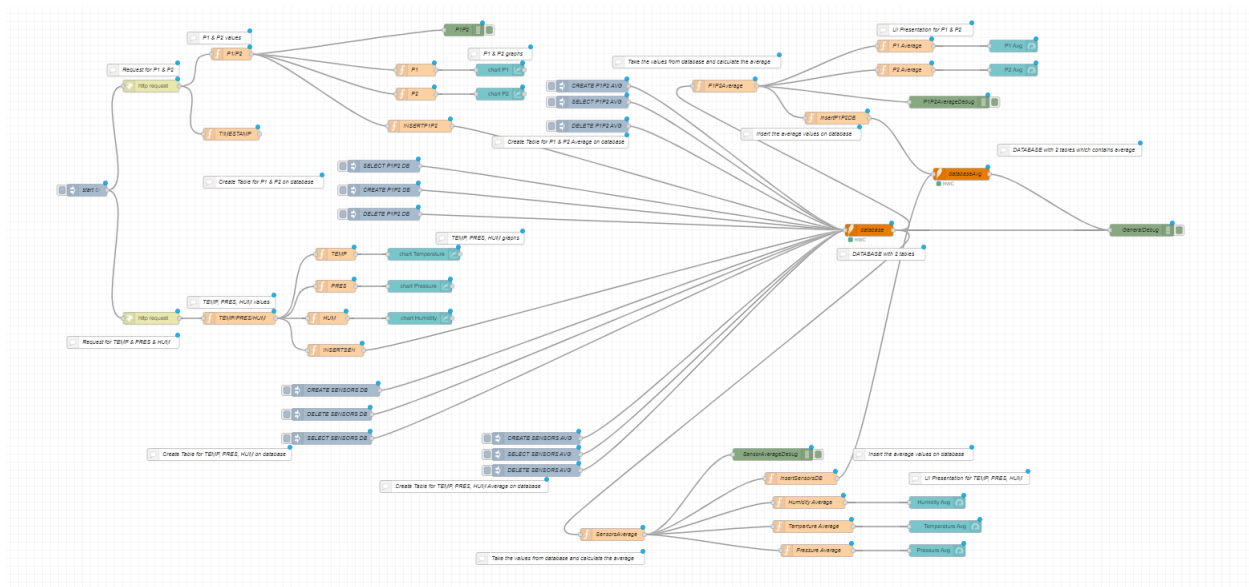
Προγραμματισμός και σχεδιασμός πλατφόρμας
IoT για ανάλυση συνθηκών του περιβάλλοντος πόλης

IOT **NODE** 02/2023 **RED**

ΟΜΑΔΑ:
Κελεπίρη Ζωή - 78
Κονδύλη Ειρήνη - 84

Τα δεδομένα που συγκεντρώσαμε είναι από τον Βόλο από όπου πήραμε τις τιμές αισθητήρων τόσο για το PM10, PM2.5 όσο και την θερμοκρασία, την υγρασία και την πίεση. Το http που χρησιμοποιήσαμε για να κάνουμε τα request μας, είναι το εξής, για το PM10, PM2.5 <https://data.sensor.community/airrohr/v1/sensor/48108/> και για τα στοιχεία temperature, humidity και pressure είναι το <https://data.sensor.community/airrohr/v1/sensor/48109/>. Για να πάρουμε τα αντίστοιχα στοιχεία, κάνουμε κλήση στους αισθητήρες ανά 5 λεπτά.

Η συνολική υλοποίηση στο node red, για το κομμάτι με τους αισθητήρες P1 και P2 αλλά και τους 3 αισθητήρες (θερμοκρασία, υγρασία, πίεση) είναι η παρακάτω:



Για να δημιουργηθεί η απεικόνιση των PM10 και PM2.5 κρατιούνται οι αντίστοιχες τιμές, μέσα από το node P1/P2, αφού έχουμε κάνει πρώτα το http request στον σένσορα με τα κατάλληλα δεδομένα. Έπειτα, μέσα από τα node για το P1 και το P2, απομονώνεται η κάθε τιμή ξεχωριστά και μέσα από τα αντίστοιχα nodes για τα chart γίνεται η γραφική τους παράσταση.

Έπειτα, για να γίνει η αποθήκευση των δεδομένων του P1 και του P2 σε έναν πίνακα της βάσης δεδομένων μας, χρησιμοποιείται το function INSERTP1P2, το οποίο αφού πάρει τα απομονωμένα δεδομένα από το P1/P2, δημιουργεί το query για να προστεθούν στον πίνακα P1P2TABLE. Στη συνέχεια, μέσα από το sql node database εκτελείται το query και προστίθενται τα στοιχεία μας στον πίνακα. Ο πίνακας P1P2TABLE, μπορεί να δημιουργηθεί, να επιλεγθούν όλα τα δεδομένα ή και να διαγραφεί, μέσα από τα κατάλληλα nodes (CREATE P1P2 DB, SELECT P1P2 DB, DELETE P1P2 DB).

Στη συνέχεια, για να γίνει η αποθήκευση των μέσων τιμών P1 και P2, δημιουργήθηκε η P1P2AVERAGE, η οποία υπολογίζει την μέση τιμή για κάθε μία από τις τιμές P1 και P2, ανά 2 ώρες και μέσα από τα nodes P1AVERAGE και P2AVERAGE, περνούν τα δεδομένα στα gauge chart nodes, για να γίνει η απεικόνισή τους, στο dashboard του node red. Για να γίνει η αποθήκευση των μέσων όρων στην βάση χρησιμοποιείται, όπως και στην προηγούμενη περίπτωση, ένα function το οποίο δημιουργεί το query για την προσθήκη των δεδομένων στον κατάλληλο πίνακα της βάσης μας, το INSERTP1P2DB.

Όσον αφορά την θερμοκρασία, την υγρασία και την πίεση, με την χρήση του κόμβου http request διαβάζουμε τα δεδομένα μας και με την χρήση του κόμβου TEMP/PRESS/HUM επιλέγουμε τις τιμές που μας ενδιαφέρουν δηλαδή το temperature, pressure και το humidity. Στην συνέχεια για να αναπαρασταθεί το διάγραμμα με τις τιμές που έχει ο σένσορας ανά 5 λεπτά, απομονώνουμε την κάθε τιμή και με την χρήση του chart κάνουμε την κατάλληλη αναπαράσταση. Η αποθήκευση των δεδομένων μας στην βάση δεδομένων γίνεται με τον κόμβο INSERTSEN όπου προστίθενται στην βάση που έχει ήδη δημιουργηθεί (database). Η βάση που έχουμε χρησιμοποιήσει είναι η database, όπου με την χρήση κατάλληλων εντολών έχουμε τις κατάλληλες ενέργειες για δημιουργία, διαγραφή και επιλογή των τιμών από την βάση δεδομένων (CREATE SENSORS DB, DELETE SENSORS DB, SELECT SENSORS DB).

Παρόμοια διαδικασία ακολουθείται και για την εύρεση του μέσου όρου για τιμές των αισθητήρων και την αποθήκευσή τους σε ξεχωριστό πίνακα στην βάση δεδομένων μας databaseAvg. Για την επίτευξη αυτής της διαδικασίας λαμβάνουμε τις τιμές που έχουν προστεθεί στον πίνακα μετά από δύο ώρες και υπολογίζουμε τον μέσο όρο τους. Η διαδικασία αυτή γίνεται με την χρήση του κόμβου SensorsAverage. Τα δεδομένα από τον κόμβο αυτόν συλλέγονται από την insertSensorsDB και εισάγονται στον πίνακα που έχουμε δημιουργήσει (SENSORS AVG). Για την αναπαράσταση των τιμών στο UI έγινε χρήση του Gauge.

Σχολιασμός Κώδικα

Για την απομόνωση των δεδομένων P1 και P2 από το JSON που έρχεται από το http request, έχουμε δημιουργήσει το function P1/P2, όπου επιλέγονται τα δεδομένα από το sensordatavalue και δημιουργείται ένα νέο αντικείμενο το οποίο περιέχει τα P1, P2 και τις τιμές τους, το οποίο στη συνέχεια επιστρέφεται από την συνάρτηση. Για να γίνει η απεικόνισή τους, δημιουργήθηκαν δύο συναρτήσεις, οι οποίες χρησιμοποιούν το αντικείμενο που επιστρέφει η προηγούμενη συνάρτηση. Η κάθε συνάρτηση από τις P1 και P2, επιλέγουν τα στοιχεία που χρειάζονται από το αντικείμενο και μέσω του msg.topic και msg.payload, περνούν τα κατάλληλα δεδομένα στους κόμβους για τα γραφήματα, όπου και προστίθενται οι νέες τιμές.

Η συνάρτηση INSERTP1P2 παίρνει τις τιμές από το αντικείμενο που είχε δημιουργηθεί από την P1/P2 και κρατάει τις τιμές τους σε κατάλληλες μεταβλητές. Έπειτα δημιουργεί το query για την εισαγωγή τους στον πίνακα P1P2TABLE και το επιστρέφει μέσα από το NewMsg. Έπειτα αυτό εκτελείται από τον κόμβο της βάσης database και έτσι γίνεται η εισαγωγή των δεδομένων στον πίνακα.

Η συνάρτηση P1P2AVERAGE είναι υπεύθυνη για τον υπολογισμό του μέσου όρου των P1 και P2. Για να γίνει αυτό, καθώς έχουμε κάθε 5 λεπτά νέα δεδομένα, χρειάζεται να έχουν γίνει 24 entries στον πίνακα της βάσης μας, για να έχουν περάσει 2 ώρες. Έτσι χρησιμοποιώντας τα δεδομένα της βάσης και ελέγχοντας κάθε φορά αν το mod 24 είναι 0, υπολογίζουμε τον μέσο όρο και τον αποθηκεύουμε σε ένα αντικείμενο το οποίο επιστρέφει η συνάρτηση μας.

Οι συναρτήσεις P1 Average και P2 Average παίρνουν η κάθε μία το αντικείμενο που δημιουργήθηκε από την προηγούμενη συνάρτηση και επιλέγουν αντίστοιχα τις τιμές που τις ενδιαφέρουν. Έπειτα, μέσα από το msg.topic και msg.payload στέλνουν τα κατάλληλα δεδομένα στα gauges, για να γίνει η απεικόνισή τους στο dashboard.

Η συνάρτηση insertP1P2DB χρησιμοποιεί και αυτή το αντικείμενο που επιστρέφεται από την P1P2AVERAGE και δημιουργεί ένα NewMsg με topic το query για την εισαγωγή των δεδομένων στον πίνακα P1P2AVG. Πριν από αυτό όμως, ελέγχει ότι υπάρχει p1AvgValue ή p2AvgValue, δηλαδή ότι έχουν περάσει 2 ώρες από την προηγούμενη φορά που έγινε εισαγωγή στην βάση. Το NewMsg στέλνεται στον κόμβο database για να εκτελεστεί το query και να γίνει η προσθήκη στον κατάλληλο πίνακα.

Προκειμένου να επιλέξουμε τα δεδομένα που θέλουμε και μας παρέχονται από το http request, δημιουργήσαμε την συνάρτηση TEMP/PRESS/HUM, η οποία λαμβάνει το μέγεθος του object που μας επιστρέφεται από το msg. Έτσι λοιπόν, δημιουργούμε ένα καινούριο object το οποίο αποθηκεύει κάθε φορά αποθηκεύει την τριάδα θερμοκρασία, πίεση και υγρασία.

Προκειμένου να απεικονιστούν τα δεδομένα που λαμβάνουμε κάθε 5 λεπτά σε γραφική παράσταση, δημιουργούμε μια συνάρτηση για κάθε δεδομένο. Στην συγκεκριμένη συνάρτηση επιλέγουμε από το obj που μας επιστρέφει η TEMP/PRESS/HUM το δεδομένο που μας ενδιαφέρει. Έτσι οι συναρτήσεις TEMP, PRES, HUM απομονώνουν τις τιμές που λαμβάνουν κάθε 5 λεπτά και απεικονίζονται σε ένα διάγραμμα. Η αποθήκευση των τιμών μας (INSERTSEN) γίνεται σε μία βάση δεδομένων(database κόμβος) όπου εισάγουμε τα δεδομένα μας(τρία) από την συνάρτηση TEMP/PRESS/HUM στο table SENSORTABLE, αυτό επιτυγχάνεται με την χρήση της INSERTSEN.

Χρησιμοποιούμε τα στοιχεία από την βάση δεδομένων μας προκειμένου να υπολογίσουμε τον μέσο όρο κάθε μεταβλητής. Από την εκφώνηση ο μέσος όρος θα πρέπει να υπολογίζεται κάθε 2 ώρες, εμείς έχουμε θέσει από την αρχή να λαμβάνουμε τα δεδομένα μας κάθε 5 λεπτά και επομένως στις 2 ώρες θα έχουμε 24 δεδομένα. Χρησιμοποιώντας έναν μετρητή υπολογίζουμε τον μέσο όρο για κάθε μεταβλητή. Η συνάρτηση υλοποιείται στον κόμβο SensorsAverage επιστρέφει ένα object όπου περιέχει τον μέσο όρο για κάθε μεταβλητή. Στη συνέχεια δημιουργούμε έναν πίνακα στην βάση μας με παρόμοιο τρόπο όπως πριν όπου αποθηκεύει τις τιμές αυτές με την χρήση του κόμβου insertSensorDB. Ο πίνακας με τους μέσους όρους είναι ο SENSORS AVG και προστίθεται στην βάση database. Για την αναπαράσταση των τιμών των μέσων όρων για κάθε μεταβλητή απομονώνουμε τις τιμές για καθεμία από αυτές με την συναρτήσεις που υπάρχουν στα Humidity Average, Temperature Average, Pressure Average.

Παράρτημα Κώδικας

1. Συνάρτηση P1/P2

```
let payloadLength = { payload: msg.payload.length };
let obj = {};

for (var i = 0; i < payloadLength.payload; i++) {
    let sensordatavalue = msg.payload[i]?.sensordatavalues;
    let timestamp = Date.parse(msg.payload[i].timestamp);
    obj = { ...obj, [`${sensordatavalue[i].value_type}`]:
sensordatavalue[i].value, ['Timestamp']: timestamp };
}

return { obj }
```

2. Συνάρτηση P1

```
msg.payload = msg.obj['P1'];
msg.topic = 'P1';

return msg;
```

3. Συνάρτηση P2

```
msg.payload = msg.obj['P2'];
msg.topic = 'P2';

return msg;
```

4. Συνάρτηση INSERTP1P2

```
let p1Value = msg.obj['P1'];
let p2Value = msg.obj['P2'];

let newMsg = {
    "topic": "INSERT INTO P1P2TABLE VALUES ( " + p1Value + ", " + p2Value
+ " ) "
}

return newMsg;
```

5. Συνάρτηση P1P2AVERAGE

```
let payloadLength = { payload: msg.payload.length };
let remainder = payloadLength.payload % 24;
let p1Average;
let p2Average;
let sumP1 = 0;
let sumP2 = 0;
let obj = {};

if(remainder === 0) {
    for (let i = 0; i < payloadLength.payload; i++) {
        sumP1 = sumP1 + msg.payload[i].P1;
        sumP2 = sumP2 + msg.payload[i].P2;
    }
    p1Average = sumP1 / payloadLength.payload;
    p2Average = sumP2 / payloadLength.payload;
    obj.p1Average = p1Average;
    obj.p2Average = p2Average;
}

return { obj };
```

6. Συνάρτηση P1 Average

```
msg.payload = msg.obj['p1Average'];
msg.topic = 'p1Average';

return msg;
```

7. Συνάρτηση P2 Average

```
msg.payload = msg.obj['p2Average'];
msg.topic = 'p2Average';

return msg;
```

8. Συνάρτηση insertP1P2DB

```
let p1AvgValue = msg.obj['p1Average'];
let p2AvgValue = msg.obj['p2Average'];

if (!p1AvgValue || !p2AvgValue) {
  return;
}

let newMsg = {
  "topic": "INSERT INTO P1P2AVG VALUES ( " + p1AvgValue + ", " +
p2AvgValue + ")"
}

return newMsg;
```

9. Συνάρτηση TEMP/PRES/HUM

```
let payloadLength = { payload: msg.payload.length };
let obj1 = {};

if (!msg.payload) {
  return;
}

for(var k = 0; k < payloadLength.payload; k++) {
  let timestamp = Date.parse(msg.payload[k].timestamp);
  for (var i = 0; i < 3; i++) {

    let sensordatavalue = msg.payload[i]?.sensordatavalues;
    for (let j = 0; j < sensordatavalue?.length; j++) {
      obj1 = { ...obj1, [`${sensordatavalue[j].value_type}`]:
sensordatavalue[j].value, ['Timestamp']: timestamp };
    }
  }
}

return { obj1 }
```


10. Συνάρτηση TEMP

```
msg.payload = msg.obj1['temperature'];  
msg.topic = 'temperature';  
  
return msg;
```

11. Συνάρτηση PRES

```
msg.payload = msg.obj1['pressure'];  
msg.topic = 'pressure';  
  
return msg;
```

12. Συνάρτηση HUM

```
msg.payload = msg.obj1['humidity'];  
msg.topic = 'humidity';  
  
return msg;
```

13. Συνάρτηση INSERTSEN

```
let temperature = msg.obj1['temperature'];  
let pressure = msg.obj1['pressure'];  
let humidity = msg.obj1['humidity'];  
  
let newMsg = {  
  "topic": "INSERT INTO SENSORTABLE VALUES ( "+ temperature + "," +  
pressure + "," + humidity + ")"  
}  
  
return newMsg;
```

14. Συνάρτηση SensorsAverage

```
let payloadLength = { payload: msg.payload.length };  
let remainder = payloadLength.payload % 24;  
let averageTemperature;  
let averageHumidity;  
let averagePressure;  
let sumTemperature = 0;  
let sumHumidity = 0;
```

```

let sumPressure = 0;
let obj = {};

if(remainder === 0) {
    for (let i = 0; i < payloadLength.payload; i++) {
        sumHumidity = sumHumidity + msg.payload[i].HUMIDITY;
        sumPressure = sumPressure + msg.payload[i].PRESSURE;
        sumTemperature = sumTemperature +
msg.payload[i].TEMPERATURE;
    }
    averageHumidity = sumHumidity / payloadLength.payload;
    averagePressure = sumPressure / payloadLength.payload;
    averageTemperature = sumTemperature / payloadLength.payload

    obj.averageTemperature = averageTemperature;
    obj.averagePressure = averagePressure;
    obj.averageHumidity = averageHumidity;

    return { obj };
}

```

15. Συνάρτηση Humidity Average

```

msg.payload = msg.obj['averageHumidity'];
msg.topic = 'averageHumidity';

return msg;

```

16. Συνάρτηση Temperature Average

```

msg.payload = msg.obj['averageTemperature'];
msg.topic = 'averageTemperature';

return msg;

```

17. Συνάρτηση Pressure Average

```

msg.payload = msg.obj['averagePressure'];
msg.topic = 'averagePressure';

return msg;

```

18. Συνάρτηση insertSensorsDB

```
let avgTemp = msg.obj['averageTemperature'];
let avgHum = msg.obj['averageHumidity'];
let avgPres = msg.obj['averagePressure'];

if (!avgTemp || !avgHum || !avgPres) {
  return;
}

let newMsg = {
  "topic": "INSERT INTO SENSORAVG VALUES ( " + avgTemp + ", " + avgHum +
  ", " + avgPres + ")"
}

return newMsg;
```