ΠΑΝΕΠΙΣΤΗΜΙΟ ΜΑΚΕΔΟΝΙΑΣ
ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΤΜΗΜΑΤΟΣ ΕΦΑΡΜΟΣΜΕΝΗΣ ΠΛΗΡΟΦΟΡΙΚΗΣ

ΧΡΗΣΗ MAP-REDUCE/HADOOP ΓΙΑ ΑΝΑΛΥΣΗ ΚΥΚΛΟΦΟΡΙΑΚΩΝ ΔΕΔΟΜΕΝΩΝ ΜΕΓΑΛΟΥ ΟΓΚΟΥ

ΑΝΤΩΝΙΟΣ ΜΠΟΥΤΟΒΙΝΑΣ

Το πρόβλημα της κυκλοφοριακής συμφόρησης

- Οικονομικές επιπτώσεις
  - 2% ΑΕΠ της Ευρωπαϊκής Ένωσης των 28 μελών
- Περιβαλλοντολογικές επιπτώσεις
  - Αύξηση εκπομπών CO2 κατά 16% έως το 2030 στις Ηνωμένες Πολιτείες της Αμερικής, την Γαλλία, την Γερμανία και το Ηνωμένο Βασίλειο
- Επιπτώσεις στην υγεία
  - 100 εκατομμύρια πολίτες της Ευρώπης είναι εκτεθειμένοι σε με θόρυβο μεγάλης έντασης (άνω των55dB)

Δεδομένα κινούμενων οχημάτων (FCD)

Τα δεδομένα κινούμενων οχημάτων συλλέγονται

- από αυτοκίνητα που έχουν δέκτες GPS
- είτε από άλλες συσκευές με δέκτες GPS, όπως για παράδειγμα κινητά τηλέφωνα

Τα δεδομένα κινούμενων οχημάτων απαρτίζονται από

- Συντεταγμένες
- Ταχύτητα
- Προσανατολισμό σε σχέση με τον βορρά
- Χρονική στιγμή που συλλέχθηκαν

APACHE HADOOP

Το Apache Hadoop είναι μια συλλογή προγραμμάτων ανοιχτού κώδικα η οποία χρησιμοποιείται για την αποθήκευση και επεξεργασία τεράστιου όγκου δεδομένων κλιμακούμενα και κατανεμημένα.

Βασικές ενότητες:

- Hadoop Common
- Hadoop Distributed File System (HDFS)
- Hadoop Yarn
- Hadoop MapReduce
  - Υλοποίηση του προγραμματιστικού μοντέλου MapReduce

Εργαλεία διπλωματικής εργασίας

Χρήση ανοιχτών κυκλοφοριακών δεδομένων του Ινστιτούτου Βιώσιμης Κινητικότητας & Δικτύων Μεταφορών (I.MET) (~140GB)

Χρήση της εικονικής μηχανής cloudera CHD ως περιβάλλον ανάπτυξης και εκτέλεσης

Δεδομένα (1/2)

Δεδομένα ιστορικού

```
2018-02-01 00:00:00.197 22.8963 40.6722416666667    0.69999999999999996 28.0    266.07998657226602
2018-02-01 00:00:00.323 22.952911666666701  40.631435000000003  3.2000000000000002  0.0 280.79000854492199
2018-02-01 00:00:00.467 22.955175000000001  40.636971666666703  4.2999999999999998  18.0    140.82000732421901
2018-02-01 00:00:00.517 22.9666216666667    40.631986666666698  8.4000000000000004  27.0    120.300003051758
2018-02-01 00:00:00.590 22.9587133333333    40.617878333333302  4.5 42.0    343.57000732421898
2018-02-01 00:00:00.607 22.929040000000001  40.662129999999998  3.5 38.0    235.60000610351599
2018-02-01 00:00:00.623 22.94341    40.636215   2.0 0.0 246.63000488281301
2018-02-01 00:00:00.637 22.928128333333301  40.643606666666699  0.40000000000000002 37.0    27.4799995422363
2018-02-01 00:00:00.667 22.959050000000001  40.641539999999999  15.699999999999999  0.0 237.61999511718801
2018-02-01 00:00:00.677 22.959050000000001  40.641539999999999  15.699999999999999  0.0 237.61999511718801
2018-02-01 00:00:00.763 22.950521666666699  40.604356666666703  0.0 42.0    189.80000305175801
2018-02-01 00:00:00.863 22.940288333333299  40.633143333333301  0.90000000000000002 0.0 202.41000366210901
2018-02-01 00:00:00.880 22.951136666666699  40.632211666666699  1.8 0.0 0.0
2018-02-01 00:00:00.897 22.962895   40.6287916666667    5.0 53.0    333.29998779296898
```

Δεδομένα πραγματικού χρόνου

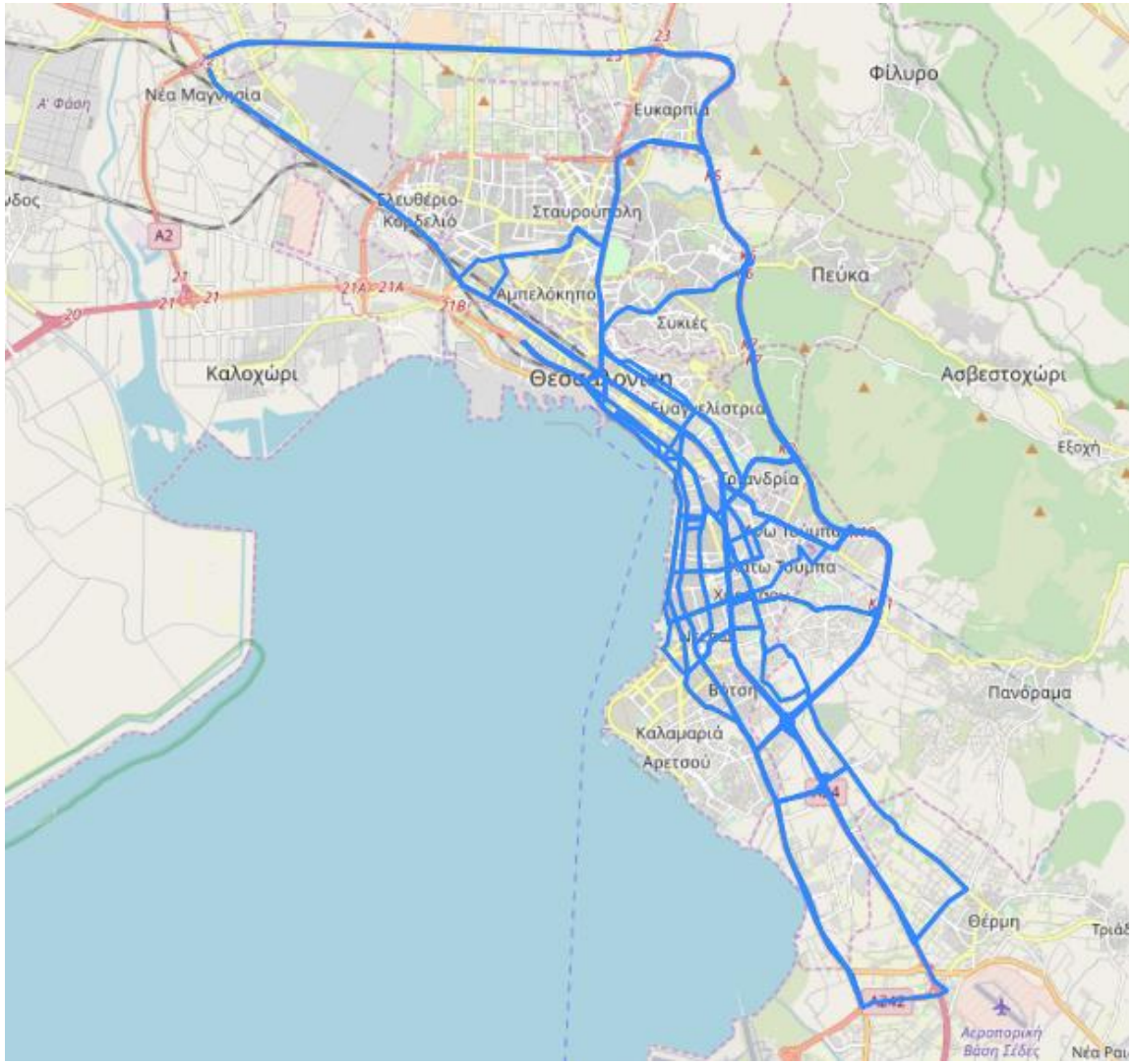
- JSON
- XML
- CSV
- KML
- MAP


```
[
  {
    "recorded_timestamp": "2019-05-04 15:22:02.647",
    "lon": "22.9675216666667",
    "lat": "40.579665",
    "altitude": "9.4",
    "speed": "55",
    "orientation": "146.589996337891"
  },
  {
    "recorded_timestamp": "2019-05-04 15:22:02.790",
    "lon": "22.9350433333333",
    "lat": "40.63537",
    "altitude": "-0.1",
    "speed": "33",
    "orientation": "150.869995117188"
  },
```

Δεδομένα (2/2)

Δεδομένα διαδρομών

- Προκαθορισμένες διαδρομές στο οδικό δίκτυο της Θεσσαλονίκης
- Επαρκείς πληροφορίες για την γεωγραφική αναπαράσταση των διαδρομών
- JSON



Εισαγωγή δεδομένων στο HDFS

Δεδομένα ιστορικού

Αντιγραφή αρχείου από το λειτουργικό σύστημα στο HDFS

hadoop fs – copyFromLocal «διαδρομή του αρχείου στο λειτουργικό σύστημα» «διαδρομή του αρχείου στο HDFS»

Δεδομένα Πραγματικού χρόνου

Κλήση web service που παρέχει τα δεδομένα πραγματικού χρόνου και δημιουργία προσωρινού αρχείου για την αντιγραφή του στο HDFS.

Αντιστοίχιση δεδομένων κινούμενων οχημάτων πάνω στις διαδρομές & δημιουργία (key-value) pairs για την φάση του Map

```
for (MyPath path : myPaths) {
    SpatialRelation relation = path.getPolyline().relate(tmp);
    if (relation.equals(SpatialRelation.CONTAINS)) {
        key.setLocation(new Text(path.getPathId()));
        key.setLatitude(new DoubleWritable(path.getPolyline().getPoints().get(0).getX()));
        key.setLongitude(new DoubleWritable(path.getPolyline().getPoints().get(0).getY()));
        key.setDistance(new DoubleWritable(path.getDistance()));
        value.setTimestamp(new Text(tokens[0]));
        DateFormat dateFormat = new SimpleDateFormat("yyyy-MM-dd hh:mm:ss");
        Date date;
        long unixTime = 0;
        try {
            date = dateFormat.parse(tokens[0]);
            unixTime = (long) date.getTime() / 1000;

        } catch (ParseException ex) {

            java.util.logging.Logger.getLogger(GeoRecordReader.class.getName()).log(Level.S
        }
        value.setUnixTimestamp(new LongWritable(unixTime));
        value.setLatitude(new DoubleWritable(Double.
                parseDouble(tokens[2])));
        value.setLongitude(new DoubleWritable(Double.
                parseDouble(tokens[1])));

        value.setAltitude(new DoubleWritable(Double.
                parseDouble(tokens[3])));
        value.setSpeed(new DoubleWritable(Double.
                parseDouble(tokens[4])));
        value.setOrientation(new DoubleWritable(Double.
                parseDouble(tokens[5])));
    }
    break;
}
```

1. Αντιστοίχιση των δεδομένων με τα γεωγραφικά σχήματα των διαδρομών
2. Δημιουργία **key-value**
    1. **Key** : id διαδρομής & απόσταση διαδρομής
    2. **Value** : δεδομένα των κινούμενων οχημάτων

Η φάση Map

```
@Override
protected void map(GeoKey key, GeoValue value, Mapper.Context
context) throws IOException, InterruptedException {
    long bucket = 0;
    if(key.getLocation() != null){
String location = key.getLocation().toString();
bucket =  value.getUnixTimestamp().get() - (value.getUnixTimestamp().get() % 3600);
GeoKeyMap mapKey = new GeoKeyMap(key,new LongWritable(bucket));

context.write(mapKey,value);

}
}
```

- Μετασχηματισμός του αρχικού κλειδιού GeoKey σε ένα νέο κλειδί το GeoKeyMap
- Δημιουργία χρονικών περιόδων
- (key- list(value))

Η φάση Reduce

- Υπολογισμός στατιστικών για την κάθε διαδρομή και την κάθε χρονική περίοδο.
- Μέγιστη & ελάχιστη ταχύτητα
- Μέσος όρος ταχύτητας
- Πλήθος εντοπισμών
- Χρόνος διαδρομής

```
context.write(keyMap.getKey().getLocation(),
        new Text(String.valueOf(keyMap.getTimeslot())+"\t"+String.valueOf(count)+"\t"+
        String.valueOf(speed)+"\t"+String.valueOf(time)+"\t"+
        String.valueOf(keyMap.getKey().getDistance().get())+"\t"+
        String.valueOf(max)+"\t"+String.valueOf(min)+"\t"+String.valueOf(median)+"\t"+
        String.valueOf(day)+"\t"+String.valueOf(month)+"\t"+String.valueOf(year)));
```

Δημιουργία προφίλ παρόμοιων χρονικών περιόδων για τα δεδομένα πραγματικού χρόνου

- Εύρεση παρομοίων χρονικών περιόδων (όμοια ημέρα και εβδομάδας)
- Ανάκτηση τιμών από την βάση δεδομένων των τιμών για τις χρονικές περιόδους
- Υπολογισμός διαφορών μεταξύ της τρέχουσας και των παρομοίων χρονικών περιόδων
  - Ταχύτητα
  - Πλήθος εντοπισμών
  - Χρόνος Διαδρομής

- Αποθήκευση των προφίλ στην βάση δεδομένων

Αποθήκευση αποτελεσμάτων σε σχεσιακή βάση δεδομένων

- Amazon RDS (MySQL)
- Ανάκτηση του αρχείου με τα αποτελέσματα από το HDFS
- Άνοιγμα σύνδεσης στην βάση
- Ανάγνωση εγγράφων του αρχείου μια προς μια δημιουργία του sql statement
- Εκτέλεση του statement

```
// the mysql insert statement
String query = " insert into FILTER_DATA (PATH_ID,TIMESTAMP ,COUNT, SPEED,"
        + " TIME, MAX_SPEED, MIN_SPEED, MEDIAN_SPEED, DAY, MONTH, YEAR)"
        + " values (?, ?, ?, ?, ?,?, ?, ?, ?,?,?)";

// create the mysql insert preparedstatement
String[] parts = readLine.split("\t");
PreparedStatement preparedStmt = conn.prepareStatement(query);
//PATH_ID
if (!parts[0].equals("") && !parts[0].equals("NaN")) {
    preparedStmt.setInt(1, Integer.parseInt(parts[0]));
} else {
    preparedStmt.setInt(1, 0);
}
```

Δημιουργία εκτελέσιμου αρχείου & εκτέλεση εφαρμογής

1. Μεταγλώττιση project

```
[cloudera@quickstart ~]$ javac -cp /usr/lib/hadoop/*:/usr/lib/hadoop-mapreduce/*:/home/cloudera/.m2/repository/org/json/
json/20180813/*:/home/cloudera/.m2/repository/com/spatial4j/spatial4j/0.5/*:/home/cloudera/Downloads/mysql-connector-jav
a-8.0.15/* /home/cloudera/NetBeansProjects/GeoMapReduceJob/src/main/java/com/anmpout/geomapreducejob/* -d build -Xlint
```

2. Δημιουργία αρχείου jar

```
File  Edit  View  Search  Terminal  Help
[cloudera@quickstart ~]$ jar -cvf geomapfilter.jar -C build/ .
```

3. Εκτέλεση αρχείου jar

```
cloudera@quickstart:~
File  Edit  View  Search  Terminal  Help
[cloudera@quickstart ~]$ hadoop jar geomapfilter.jar com.anmpout.geomapreducejob.GeoFilter
 /user/thesis/samples/fcd_gps_06_2017_1 false true
```

- Διαδρομή με αρχείο εισόδου
- Τύπος δεδομένων εισόδου
- Αποθήκευση δεδομένων σε DB

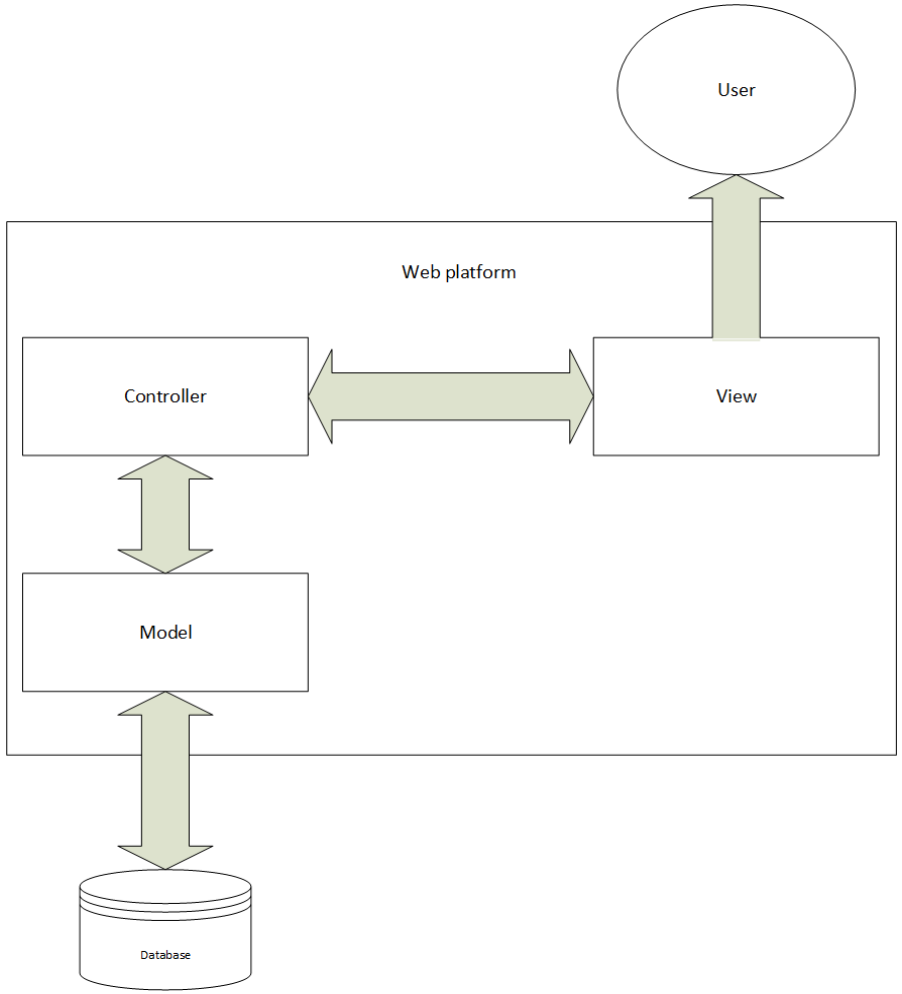
Αυτόματη εκτέλεση διαδικασίας

```
#!/bin/bash
#retrieve date
timestamp=$( date +"%Y-%m-%d" )
#lookup current date file
for entry in "$search_dir"/home/cloudera/Downloads/realTime/$timestamp/*
do
#create file name
file=$(basename "$entry")
#create timestamp
timestamp=$( date +%H )
#convert timestamp
run=$((10#$timestamp-1))
#if we have a match excecute the following commands
if [ "$run" = "$file" ]
then
#copy tmp file from local fs to HDFS
hadoop fs -copyFromLocal $entry /user/thesis/samples/$file
hadoop fs -ls /user/thesis/samples
#execute jar for tmp file
hadoop jar /home/cloudera/geomapfilter.jar
com.anmpout.geomapreducejob.GeoFilter /user/thesis/samples/$file true
true
#remove tmp file
hadoop fs -rm /user/thesis/samples/$file
hadoop fs -ls /user/thesis/samples
fi
done
```

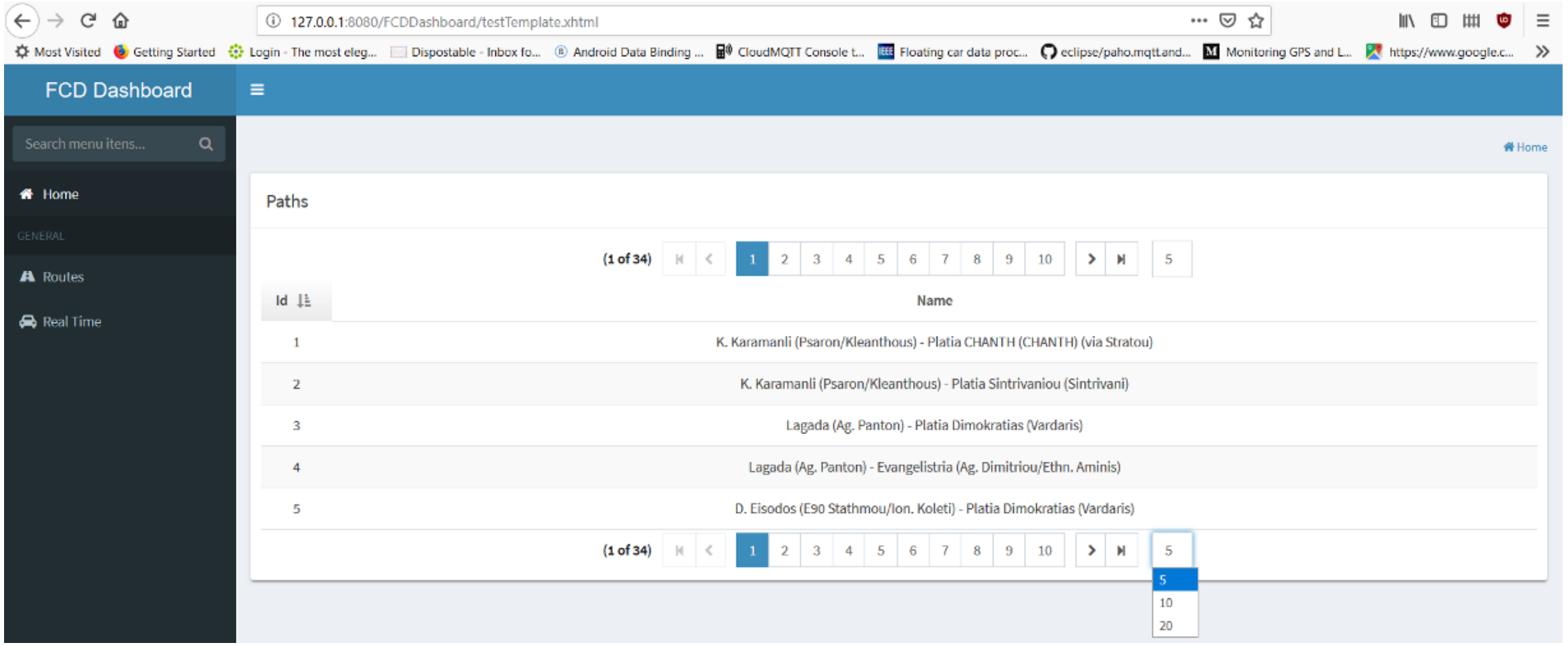
Εκτέλεση cron job κάθε μια ωρολογιακή ώρα

Πλατφόρμα παρουσίασης αποτελεσμάτων

- Java EE
- JSF
- Open source template AdminFaces
- Application server WildFily



Παρουσίαση αποτελεσμάτων δεδομένων ιστορικού (1/3)

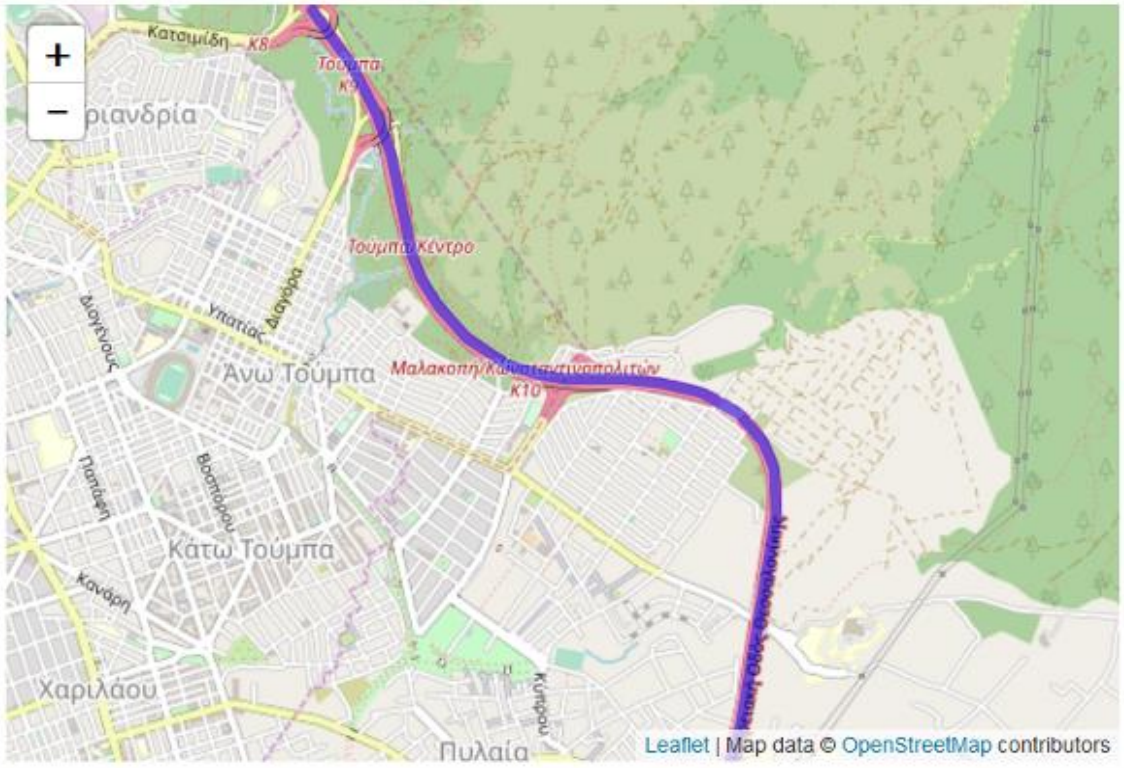


Παρουσίαση αποτελεσμάτων δεδομένων ιστορικού (2/3)

Paths Detail

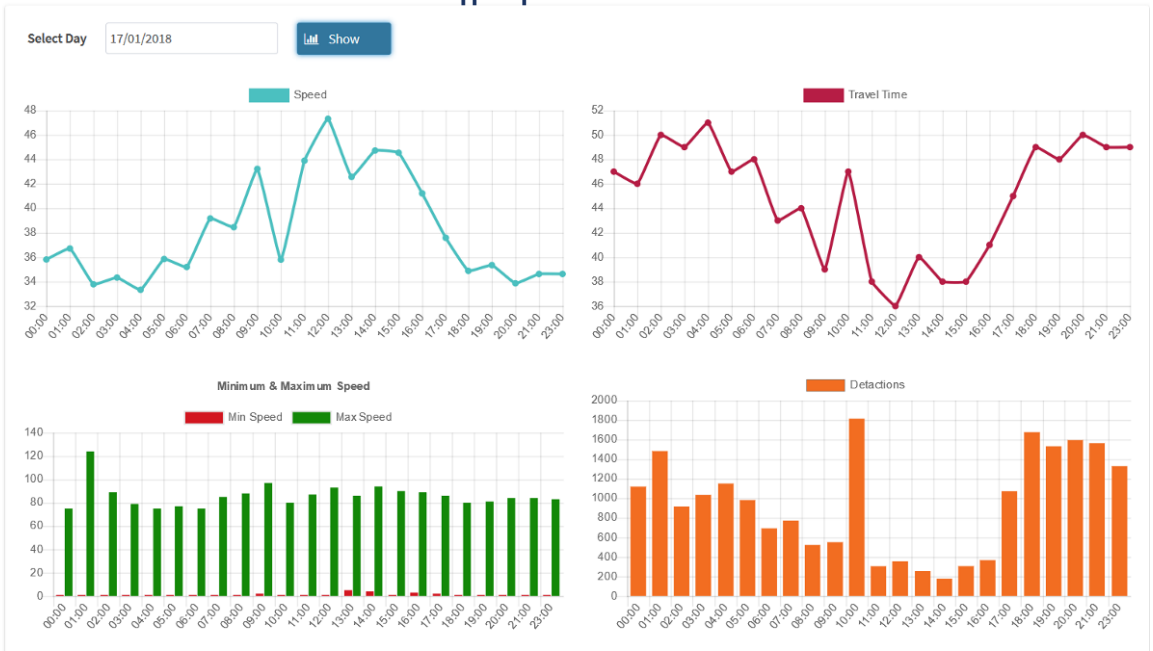
Id:	81
Name:	Panorama Intersection - City Center Intersection
Region :	Tessaloniki
Length :	4.14 Km

Path Position

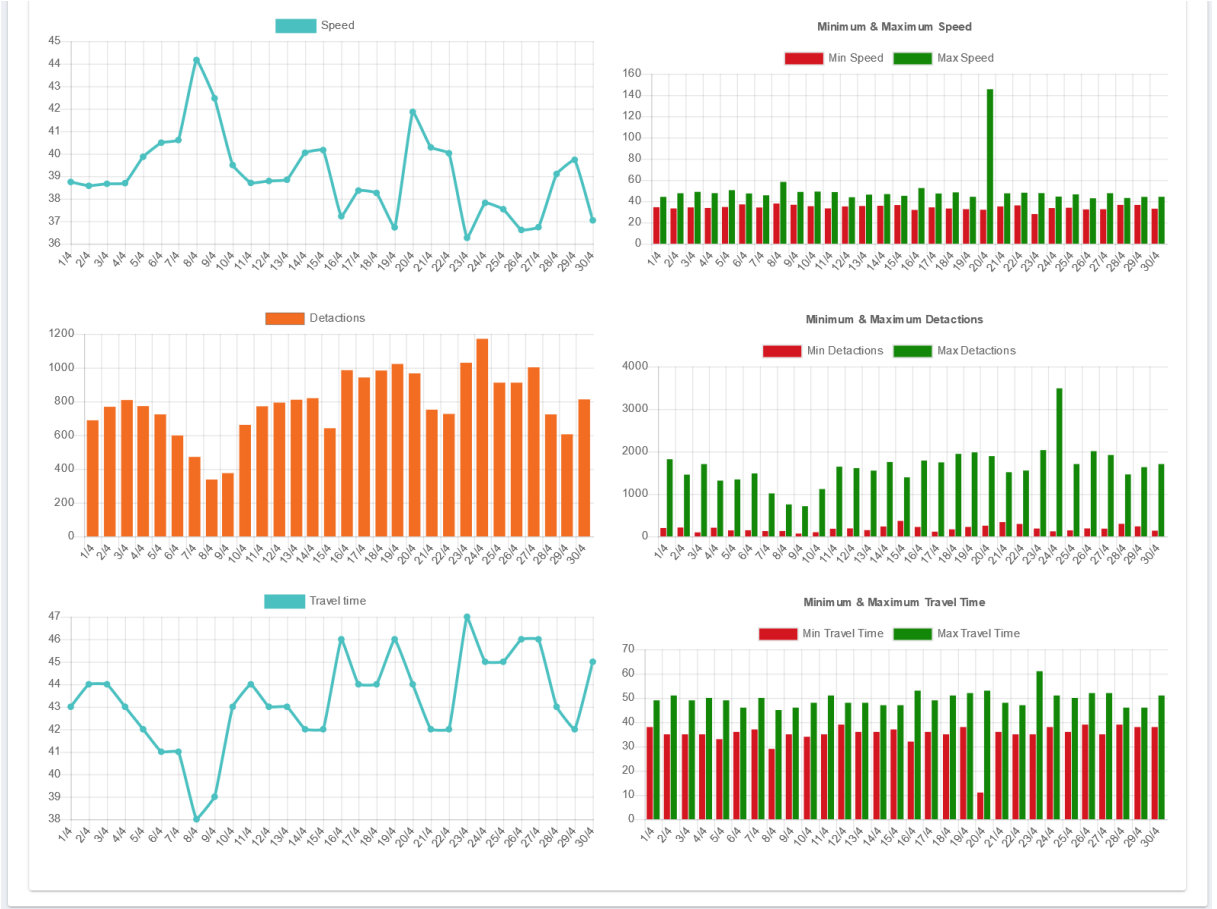


Παρουσίαση αποτελεσμάτων δεδομένων ιστορικού (3/3)

Στατιστικά δεδομένα ανά ημέρα



Στατιστικά δεδομένα ανά μήνα



Παρουσίαση αποτελεσμάτων πραγματικού χρόνου

Statistics

Timestamp	Speed	Time	Count
20/05/2019 20:00:00	16,00	719	52
21/05/2018 20:00:00	4,20%	-4.3%	-7.6%
15/05/2017 20:00:00	8,20%	-8.4%	6.0%

Μελλοντικές επεκτάσεις - βελτιώσεις

- Αύξηση των πηγών λήψης δεδομένων
- Βελτιστοποίηση στο φιλτράρισμα των δεδομένων για ορθότερα αποτελέσματα
  - Χρήση id συσκευής
- Χρήση εφαρμογής σε δεδομένα και άλλων πόλεων
- Εκπόνηση συγκριτικών μελετών ως προς τα αποτελέσματα διαφορετικών πόλεων

Ερωτήσεις - Απορίες

Χρήσιμοι σύνδεσμοι

- https://github.com/ampoutovinas
- http://opendata.imet.gr/
- https://www.cloudera.com/downloads/quickstart_vms/5-13.html

Ευχαριστώ για την προσοχή σας!