

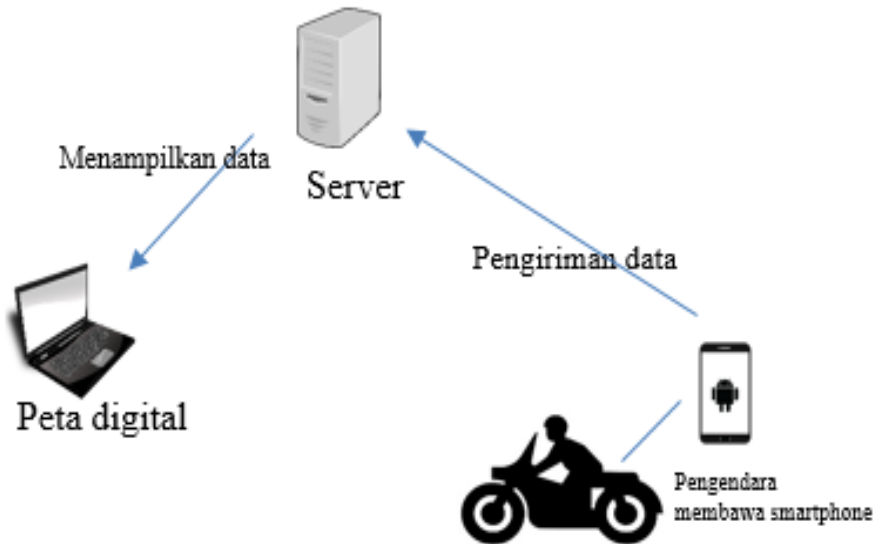
Progress TA

Rancang Bangun Sistem Pemetaan Alat Pembatas Kecepatan Menggunakan Android Smartphone dengan Akselerometer

5112100212

Otniel Yehezkiel

Latar Belakang







Gambar 1. Alur Sistem

Tujuan TA

1. Membuat sistem monitoring yang memetakan lokasi alat pembatas kecepatan (polisi tidur)

Progress

1. Membuat webservice (server) untuk menerima data yang dikirimkan dari accelerometer.
2. Membuat aplikasi pada android untuk mendeteksi perubahan akselerasi.
3. Data akselerometer pada sensor android mengacu pada sumbu x,y,z pada android. Sehingga data akselerasi (x,y,z) sangat berpengaruh pada posisi android. Solusi: Melakukan reorientasi koordinat.

Position	X	Y	Z
UP: 	0	9.81m/s^2	0
LEFT: 	9.81m/s^2	0	0
DOWN: 	0	-9.81m/s^2	0
RIGHT: 	-9.81m/s^2	0	0
FRONT UP: 	0	0	9.81m/s^2
BACK UP: 	0	0	-9.81m/s^2



Progress

4. Server

Database menggunakan postgresql



The screenshot shows the phpPgAdmin web interface in a browser. The address bar displays the URL `128.199.235.115/phpPgAdmin/`. The interface is titled "phpPgAdmin" and shows a connection to PostgreSQL 9.3.11 running on localhost:5432, logged in as user "postgres".

The left sidebar shows a tree view of the database structure:

- Servers
 - PostgreSQL
 - postgres
 - project
 - Schemas
 - public
 - Tables
 - Views
 - Sequences
 - Functions
 - Full Text Search
 - Domains
 - test

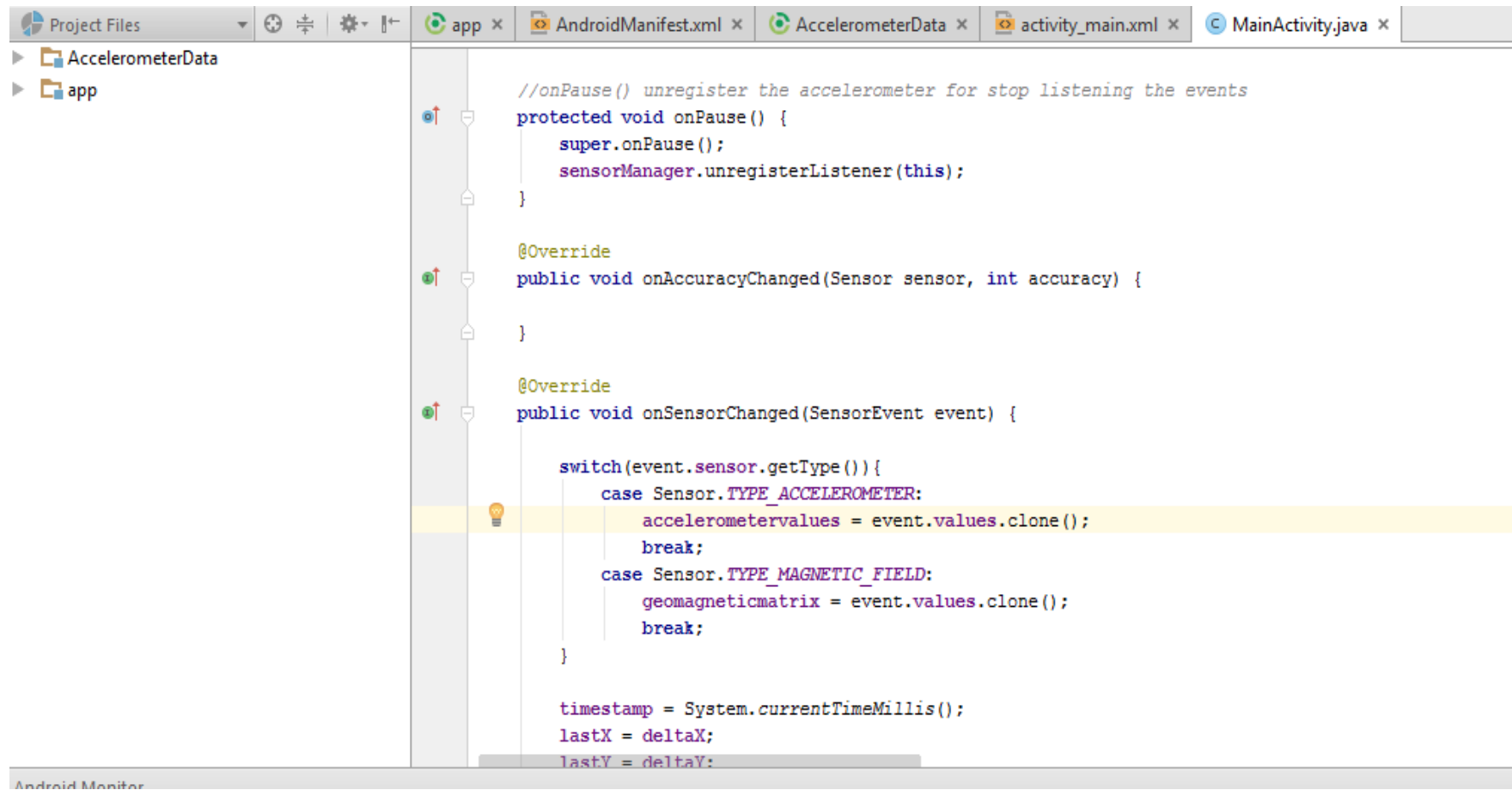
The main content area displays a table of database objects:

Table	Owner	Tablespace	Estimated row count	
<input type="checkbox"/> acc_data	postgres		0	Browse Select Insert
<input type="checkbox"/> jenis	postgres		0	Browse Select Insert
<input type="checkbox"/> location	postgres		0	Browse Select Insert
<input type="checkbox"/> users	postgres		0	Browse Select Insert

Below the table, there are buttons for "Tables?", "Views?", and "Sequences?". At the bottom, there are links for "Create table" and "Create table like".

5. Aplikasi Android

Menggunakan Sensor.TYPE_ACCELEROMETER dan Sensor.TYPE_MAGNETIC_FIELD:



```
//onPause() unregister the accelerometer for stop listening the events
protected void onPause() {
    super.onPause();
    sensorManager.unregisterListener(this);
}

@Override
public void onAccuracyChanged(Sensor sensor, int accuracy) {

}

@Override
public void onSensorChanged(SensorEvent event) {

    switch(event.sensor.getType()){
        case Sensor.TYPE_ACCELEROMETER:
            acceleromtervalues = event.values.clone();
            break;
        case Sensor.TYPE_MAGNETIC_FIELD:
            geomagneticmatrix = event.values.clone();
            break;
    }

    timestamp = System.currentTimeMillis();
    lastX = deltaX;
    lastY = deltaY;
}
```



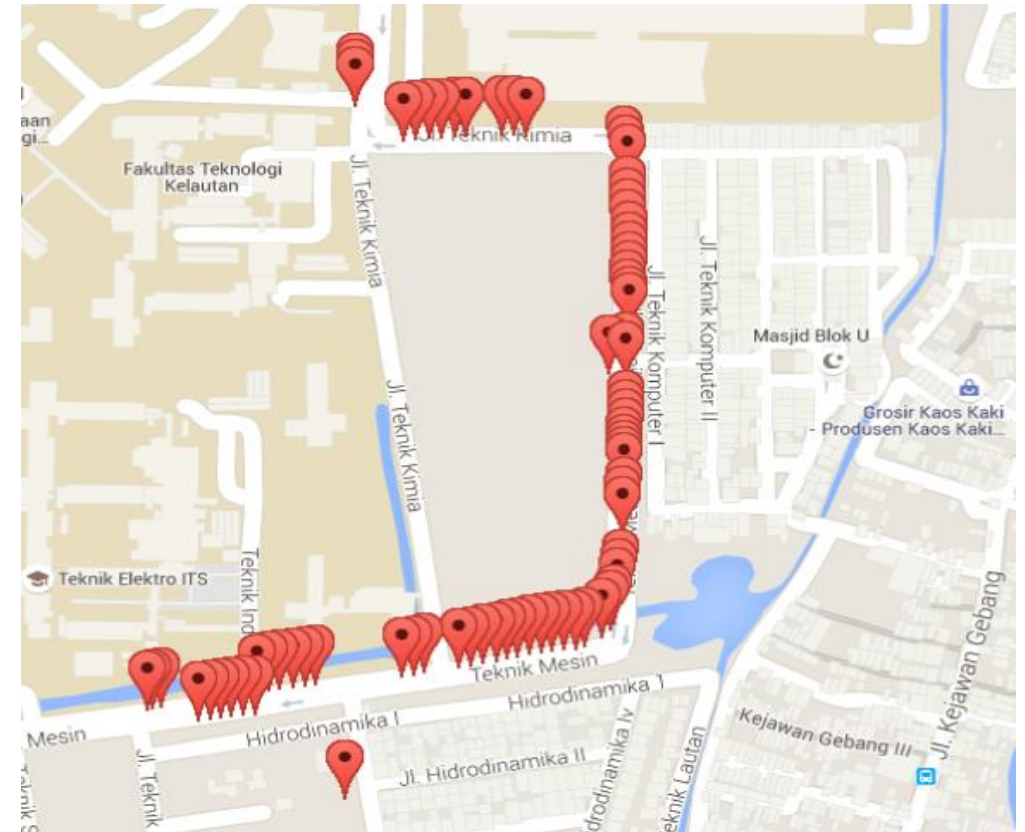
Uji Coba

Percobaan 1

Metode: Treshold z-axis melewati diatas 14 m/s² dan dibawah 6 m/s². Ketika melewati treshold, maka device akan merecord data selama 2 detik setelah terjadi lalu trigger dan dikirim ke server. False positif masih sangat banyak

Evaluasi:

1. Meningkatkan treshold maximum dan mininum.
2. Dalam setiap blok/segment event detection (bump/hole), atribut koordinatnya disamakan dengan koordinat pertama kali trigger event.
3. Membuat marker yang berbeda antara bump dan hole.



Percobaan 2

Metode: Mengendarai motor dengan kecepatan 20-40 kmph. Apabila Threshold z-axis melewati diatas 15 m/s^2 dan dibawah 5 m/s^2 . Ketika melewati treshold, maka device akan merecord data selama 2 detik setelah terjadi lalu trigger dan dikirim ke server. Terdapat false positif bump dianggap sebagai hole.

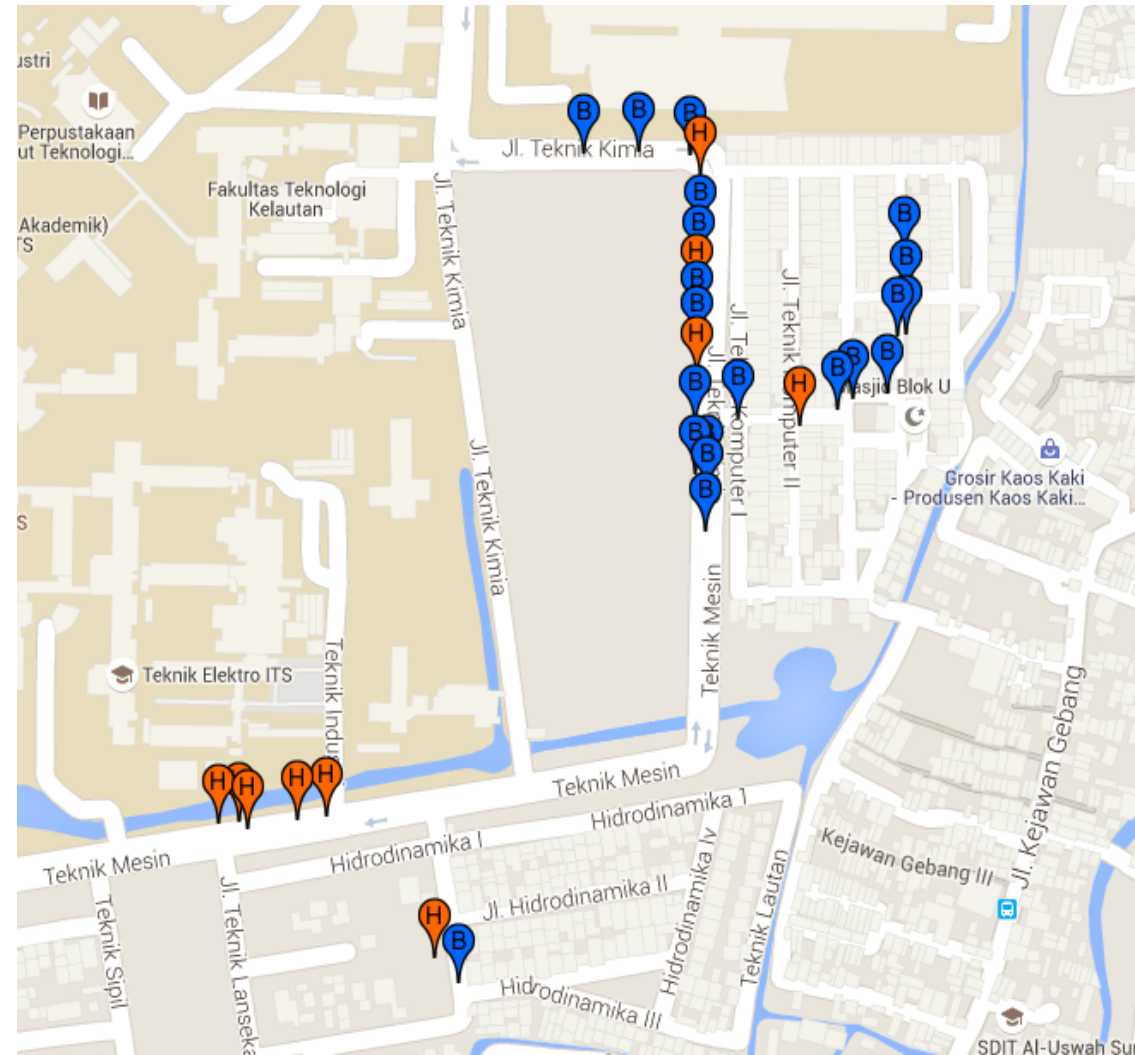
Evaluasi:

1. Melakukan percobaan pada lokasi jalan yang baik (tidak bergelombang).
2. Mencoba percobaan pada lubang.
3. Memperbanyak percobaan.

Keterangan:

Marker B – warna biru merupakan lokasi bump yang terdeteksi

Marker H – warna orange merupakan lokasi lubang yang terdeteksi



Percobaan 3

Metode: Treshold z-axis melewati diatas 15 m/s^2 dan dibawah 5 m/s^2 . Ketika melewati treshold, maka device akan merecord data selama 2 detik setelah terjadi lalu trigger dan dikirim ke server. Terdapat false positif bump dianggap sebagai hole.

Evaluasi:

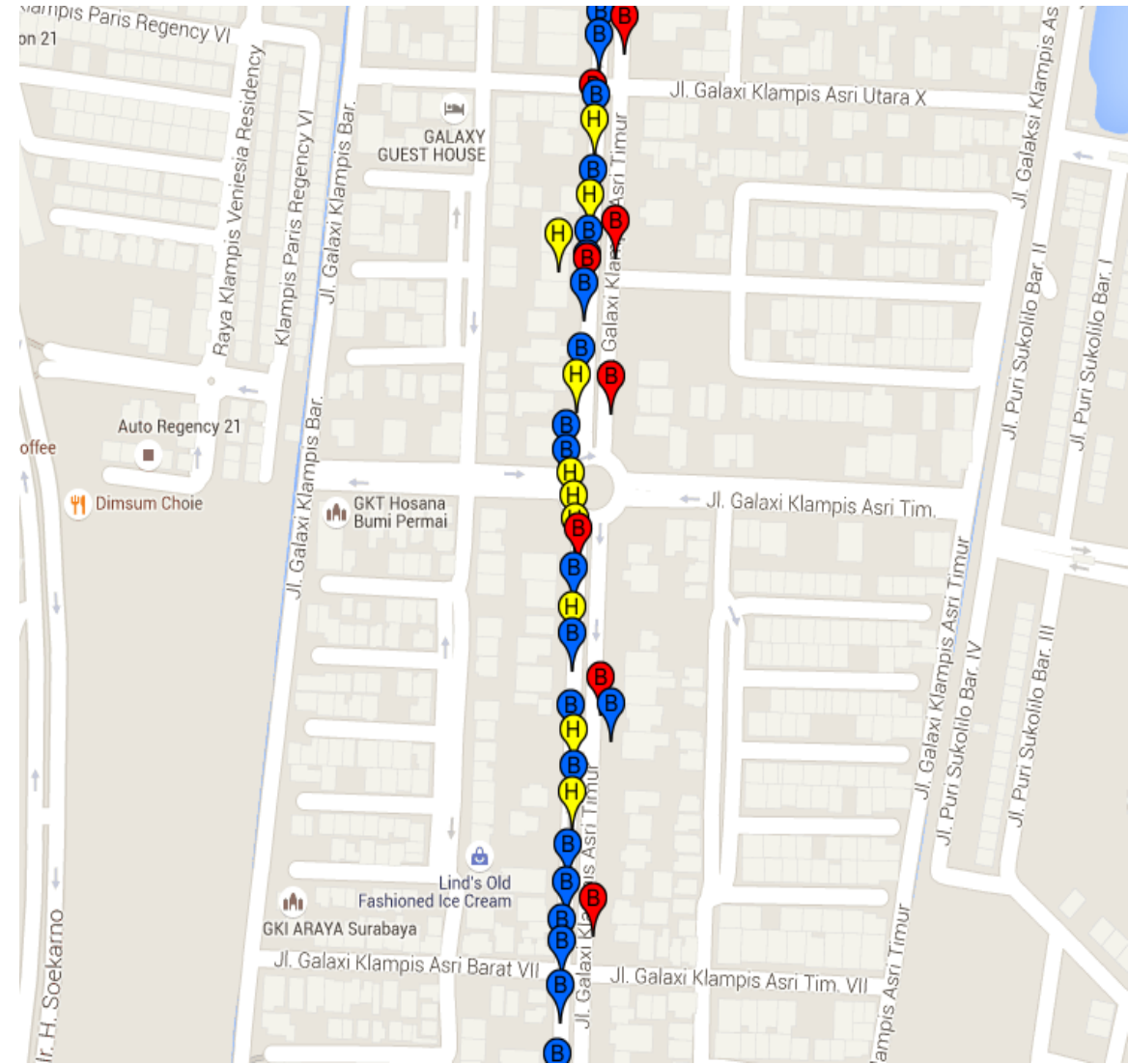
1. Pada kondisi high speed (lebih dari 25kmph) noise sangat tinggi
2. Menaikkan treshold pada high speed.

Keterangan:

Marker B – warna biru merupakan lokasi bump yang terdeteksi

Marker H – warna orange merupakan lokasi lubang yang terdeteksi

Marker B – warna lokasi alat pembatas kecepatan yang benar

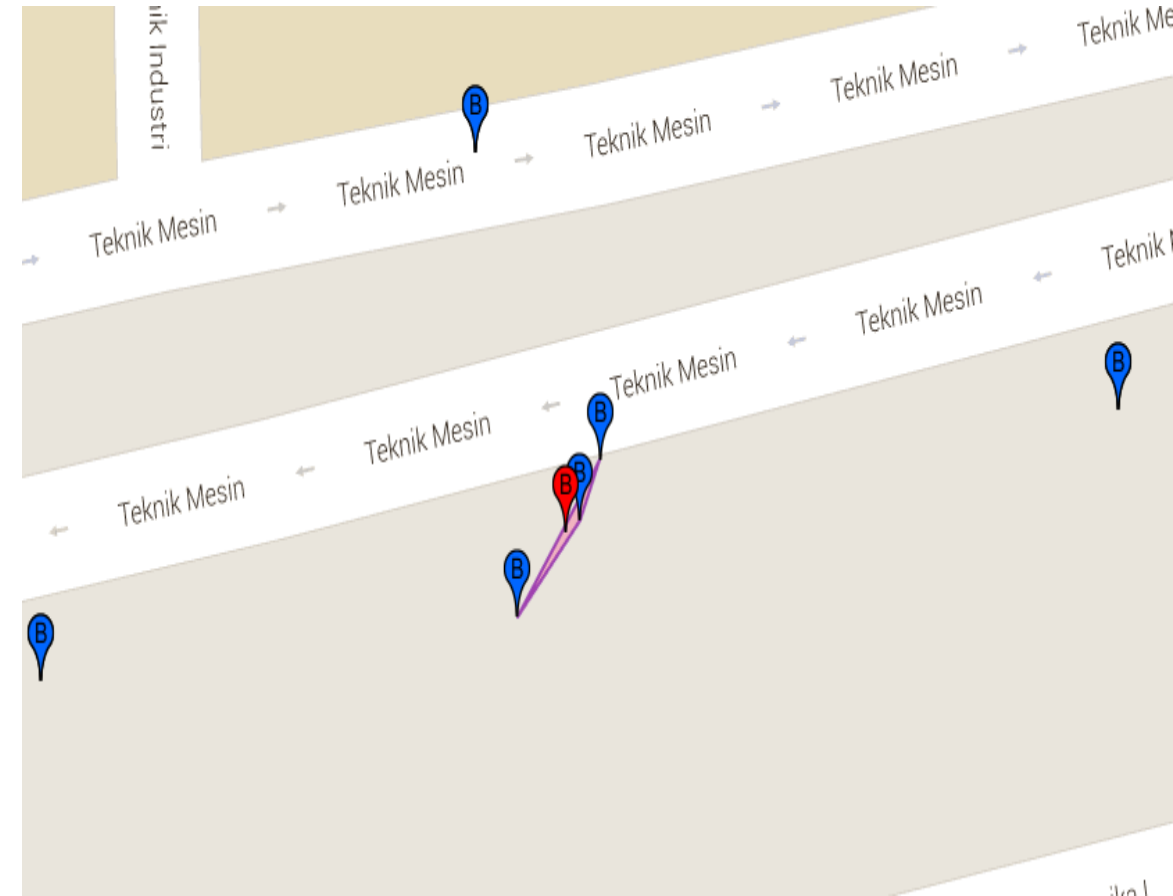


Permasalahan selanjutnya adalah apabila pengendara melewati tempat yang sama dan menemukan suatu bump (lonjakan) namun karena error pada GPS sehingga membuat beberapa titik dalam suatu lokasi. Untuk mewakili titik-titik yang berdekatan tersebut maka diambil titik centroidnya.

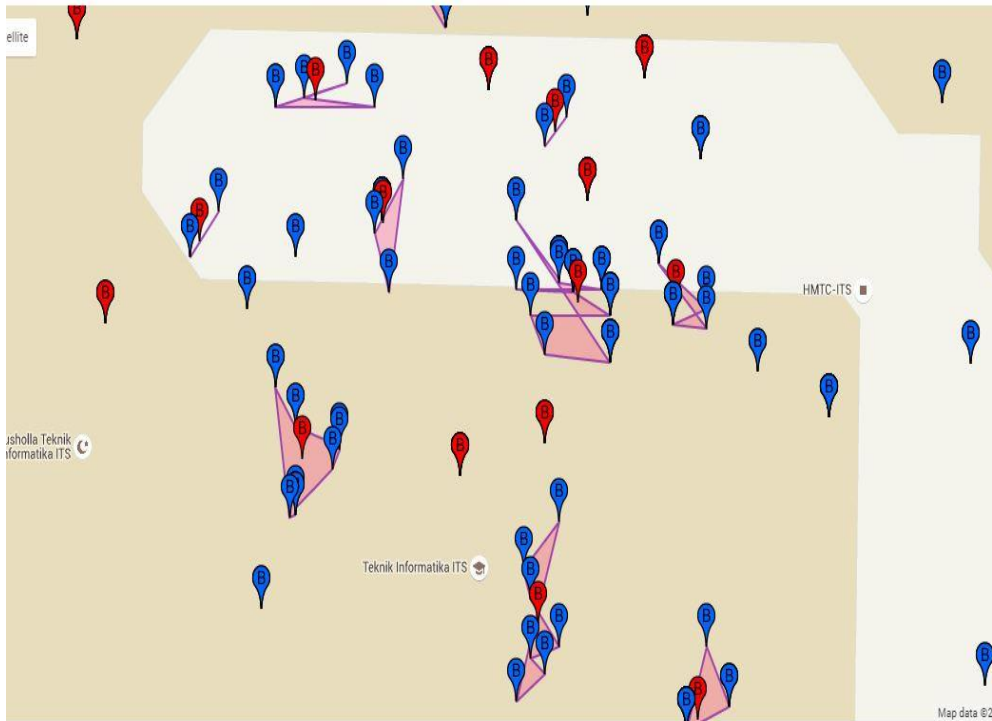
Maka solusi yang diajukan adalah menggunakan metode clustering pada lokasi koordinat hasil deteksi yang berdekatan.

Algoritma clustering yang digunakan adalah DBSCAN menggunakan library di python.

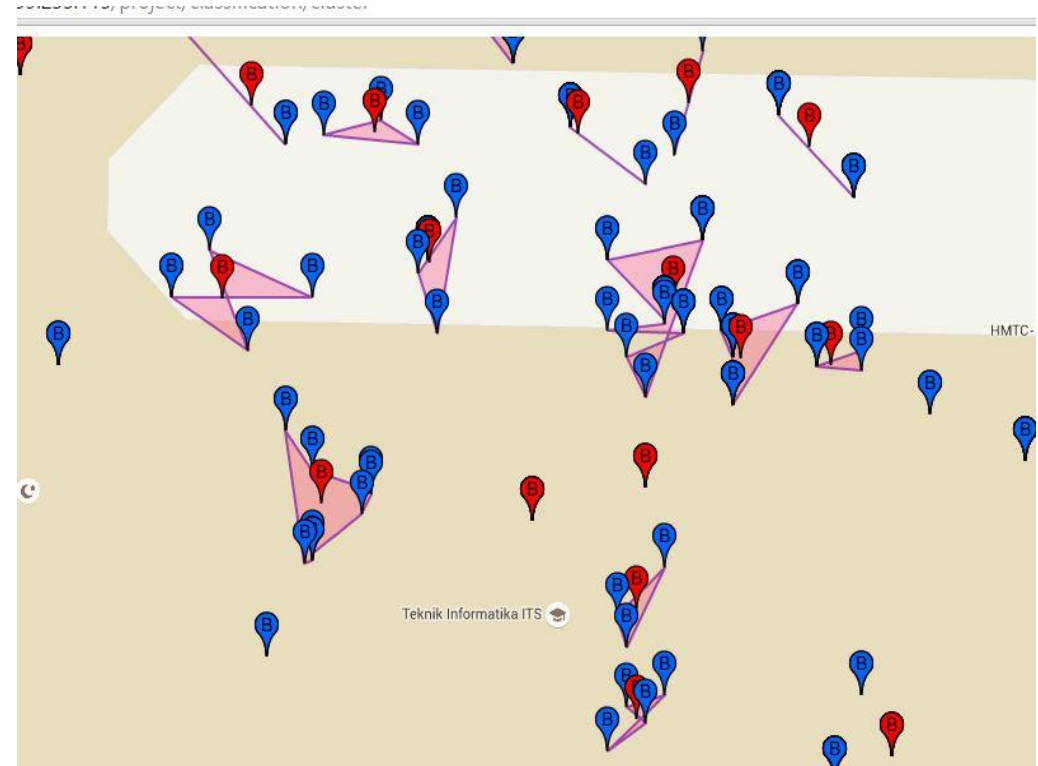
Keterangan gambar: Pada 3 titik Bump (marker biru) yang berdekatan, dikluster lalu direpresentasikan menjadi satu titik, yaitu centroid dari ketiga titik tersebut (marker merah).



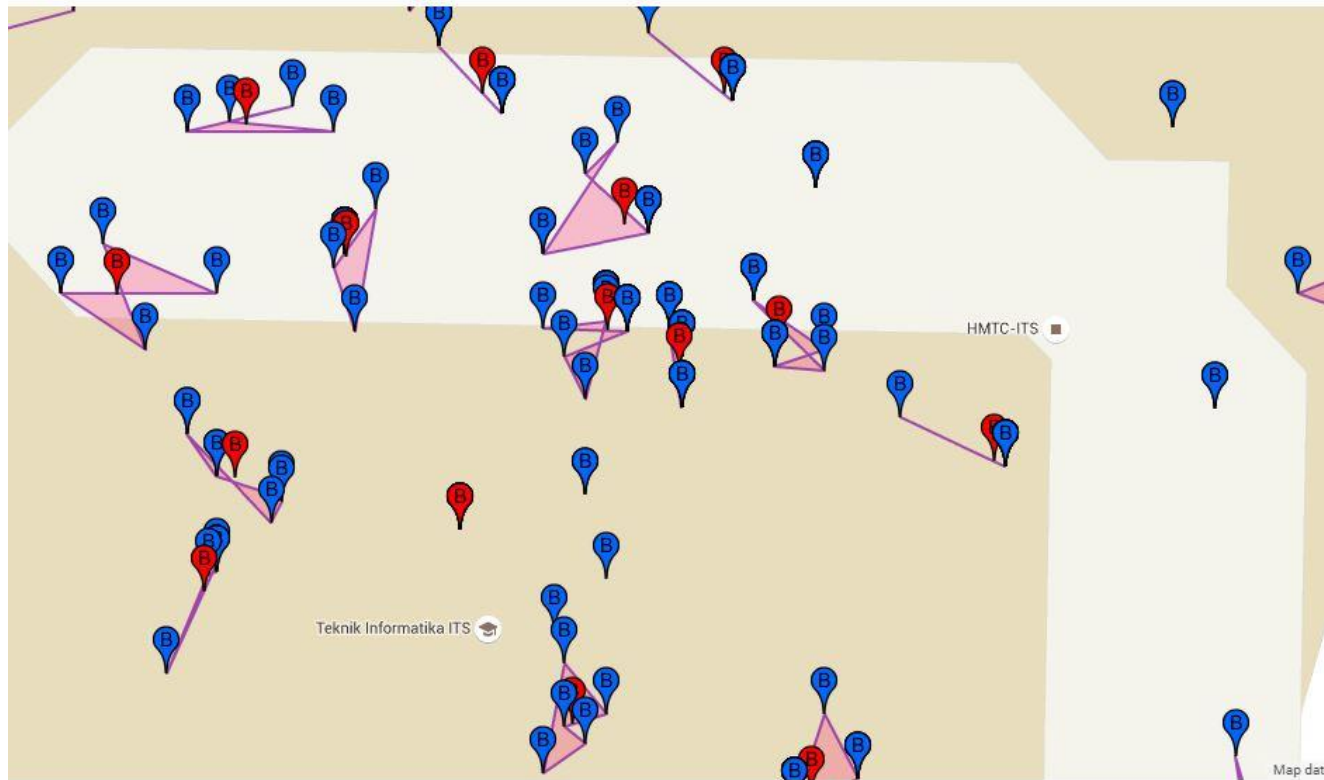
Hasil Perbandingan Clustering



DBSCAN



BIRCH



K-MEANS

Permasalahan

1. Proses Pengiriman data otomatis: Menggunakan threshold pada axis Z akselerometer. Setelah terjadi event (lonjakan) maka data akselerasi akan direcord selama 2 detik setelahnya. Belum menemukan threshold Z yang paling optimal
2. Belum menemukan metode untuk mendeteksi Bump/Pothole.