

SKRIPSI

*PORTING PHP MENJADI JAVA/PLAY FRAMEWORK (STUDI  
KASUS KIRI DASHBOARD SERVER SIDE)*



TOMMY ADHITYA THE

NPM: 2012730031

PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS TEKNOLOGI INFORMASI DAN SAINS  
UNIVERSITAS KATOLIK PARAHYANGAN  
2015



**UNDERGRADUATE THESIS**

**PORTING PHP TO JAVA/PLAY FRAMEWORK (CASE STUDY  
KIRI DASHBOARD SERVER SIDE)**



**TOMMY ADHITYA THE**

**NPM: 2012730031**

**DEPARTMENT OF INFORMATICS  
FACULTY OF INFORMATION TECHNOLOGY AND SCIENCES  
PARAHYANGAN CATHOLIC UNIVERSITY  
2015**



# LEMBAR PENGESAHAN

## *PORTING* PHP MENJADI JAVA/PLAY FRAMEWORK (STUDI KASUS KIRI *DASHBOARD SERVER SIDE*)

TOMMY ADHITYA THE

NPM: 2012730031

Bandung, 17 September 2015

Menyetujui,

Pembimbing Utama

Pembimbing Pendamping

Pascal Alfadian, M.Com.

«pembimbing pendamping/2»

Ketua Tim Penguji

Anggota Tim Penguji

«penguji 1»

«penguji 2»

Mengetahui,

Ketua Program Studi

Mariskha Tri Aditia, PDEng



## PERNYATAAN

Dengan ini saya yang bertandatangan di bawah ini menyatakan bahwa skripsi dengan judul:

### ***PORTING PHP MENJADI JAVA/PLAY FRAMEWORK (STUDI KASUS KIRI DASHBOARD SERVER SIDE)***

adalah benar-benar karya saya sendiri, dan saya tidak melakukan penjiplakan atau pengutipan dengan cara-cara yang tidak sesuai dengan etika keilmuan yang berlaku dalam masyarakat keilmuan.

Atas pernyataan ini, saya siap menanggung segala risiko dan sanksi yang dijatuhkan kepada saya, apabila di kemudian hari ditemukan adanya pelanggaran terhadap etika keilmuan dalam karya saya, atau jika ada tuntutan formal atau non-formal dari pihak lain berkaitan dengan keaslian karya saya ini.

Dinyatakan di Bandung,  
Tanggal 17 September 2015

Meterai

Tommy Adhitya The  
NPM: 2012730031





## ABSTRAK

«Tuliskan abstrak anda di sini, dalam bahasa Indonesia» Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

**Kata-kata kunci:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Indonesia»



## ABSTRACT

«Tuliskan abstrak anda di sini, dalam bahasa Inggris» Fusce mauris. Vestibulum luctus nibh at lectus. Sed bibendum, nulla a faucibus semper, leo velit ultricies tellus, ac venenatis arcu wisi vel nisl. Vestibulum diam. Aliquam pellentesque, augue quis sagittis posuere, turpis lacus congue quam, in hendrerit risus eros eget felis. Maecenas eget erat in sapien mattis porttitor. Vestibulum porttitor. Nulla facilisi. Sed a turpis eu lacus commodo facilisis. Morbi fringilla, wisi in dignissim interdum, justo lectus sagittis dui, et vehicula libero dui cursus dui. Mauris tempor ligula sed lacus. Duis cursus enim ut augue. Cras ac magna. Cras nulla. Nulla egestas. Curabitur a leo. Quisque egestas wisi eget nunc. Nam feugiat lacus vel est. Curabitur consectetur.

**Keywords:** «Tuliskan di sini kata-kata kunci yang anda gunakan, dalam bahasa Inggris»



*«kepada siapa anda mempersembahkan skripsi ini...?»*



## KATA PENGANTAR

Nulla malesuada porttitor diam. Donec felis erat, congue non, volutpat at, tincidunt tristique, libero. Vivamus viverra fermentum felis. Donec nonummy pellentesque ante. Phasellus adipiscing semper elit. Proin fermentum massa ac quam. Sed diam turpis, molestie vitae, placerat a, molestie nec, leo. Maecenas lacinia. Nam ipsum ligula, eleifend at, accumsan nec, suscipit a, ipsum. Morbi blandit ligula feugiat magna. Nunc eleifend consequat lorem. Sed lacinia nulla vitae enim. Pellentesque tincidunt purus vel magna. Integer non enim. Praesent euismod nunc eu purus. Donec bibendum quam in tellus. Nullam cursus pulvinar lectus. Donec et mi. Nam vulputate metus eu enim. Vestibulum pellentesque felis eu massa.

Bandung, September 2015

Penulis





# DAFTAR ISI

KATA PENGANTAR	xv
DAFTAR ISI	xvii
DAFTAR GAMBAR	xviii
DAFTAR TABEL	xix
<b>1 PENDAHULUAN</b>	<b>1</b>
1.1 Latar Belakang . . . . .	1
1.2 Rumusan Masalah . . . . .	1
1.3 Tujuan . . . . .	2
1.4 Batasan Masalah . . . . .	2
1.5 Metode Penelitian . . . . .	2
1.6 Sistematika Penulisan . . . . .	2
<b>2 DASAR TEORI</b>	<b>5</b>
2.1 Kode KIRI <i>Dashboard Server Side</i> . . . . .	5
2.2 MySQL Spatial Extensions . . . . .	5
2.2.1 Point . . . . .	5
2.2.2 LineString . . . . .	6
2.2.3 Polygon . . . . .	6
2.3 Play Framework . . . . .	6
<b>DAFTAR REFERENSI</b>	<b>7</b>
<b>A THE PROGRAM</b>	<b>9</b>
<b>B THE SOURCE CODE</b>	<b>11</b>

## DAFTAR GAMBAR

2.1	Contoh penggunaan tipe data <code>Point[1]</code> . . . . .	5
2.2	Contoh penggunaan tipe data <code>LineString[1]</code> . . . . .	6
2.3	Contoh penggunaan tipe data <code>Polygon[1]</code> . . . . .	6
A.1	Interface of the program . . . . .	9

## DAFTAR TABEL



# BAB 1

## PENDAHULUAN

### 1.1 Latar Belakang

KIRI[2] merupakan situs web untuk membantu pengguna menemukan rute transportasi umum ke tempat tujuannya. Dengan memasukkan lokasi awal serta lokasi tujuan pengguna tersebut, situs web KIRI akan memberikan langkah-langkah (contoh: berjalan sejauh berapa meter, menggunakan angkot, dll) tercepat untuk sampai ke lokasi tujuan. Situs web KIRI telah hadir di berbagai kota seperti Bandung, Depok, Jakarta, Surabaya dan Malang.

KIRI *Dashboard*[3] adalah bagian dari situs web KIRI. KIRI *Dashboard* berfungsi sebagai pengatur proses CRUD (*Create, Read, Update* dan *Delete*) daftar rute yang terdapat dalam *database* situs web KIRI. KIRI *Dashboard Server Side* menggunakan bahasa PHP dalam pembuatannya[4]. Bahasa PHP kurang cocok untuk proyek besar seperti *dashboard*. Salah satu penyebab bahasa PHP kurang cocok adalah karena tidak ada deklarasi dan tipe variabel dalam penggunaan bahasa PHP.

Java merupakan bahasa pemrograman yang umum digunakan oleh banyak orang. Selain umum digunakan, Java juga merupakan bahasa pemrograman yang lebih terstruktur dibandingkan PHP. Adanya deklarasi dan tipe variabel pada Java membuat setiap variabel memiliki kegunaan yang lebih jelas dan mudah dimengerti. Play Framework merupakan *framework* yang membantu implementasi Java dalam pembuatan suatu situs web. Play Framework juga cocok untuk proyek skala besar karena arsitekturnya sudah menggunakan konsep MVC (*Model View Controller*)[5].

Berdasarkan ditemukannya kekurangan-kekurangan pada KIRI *Dashboard Server Side* seperti yang telah dijelaskan, maka solusi untuk mengatasi kekurangan tersebut adalah dibuatlah penelitian ini untuk mengubah KIRI *Dashboard Server Side* yang semula dalam bahasa PHP menjadi bahasa Java dengan menggunakan Play Framework.

### 1.2 Rumusan Masalah

Berikut adalah susunan permasalahan yang akan dibahas pada penelitian ini:

1. Bagaimana isi kode KIRI *Dashboard Server Side* dan apa saja kekurangan yang ada di dalamnya?
2. Bagaimana cara kerja Play Framework berbasis MVC?
3. Bagaimana melakukan *porting* KIRI *Dashboard Server Side* yang semula dalam bahasa PHP menjadi bahasa Java dengan menggunakan Play Framework?

### 1.3 Tujuan

Berdasarkan rumusan masalah yang telah dibuat, maka tujuan skripsi ini dijelaskan ke dalam poin-poin sebagai berikut:

1. Mengetahui isi kode KIRI *Dashboard Server Side* dan kekurangan-kekurangan yang ada di dalamnya.
2. Mengetahui cara kerja Play Framework berbasis MVC.
3. Melakukan *porting* KIRI *Dashboard Server Side* yang semula dalam bahasa PHP menjadi bahasa Java dengan menggunakan Play Framework

### 1.4 Batasan Masalah

Skripsi ini dibuat berdasarkan batasan-batasan sebagai berikut:

1. Play Framework yang digunakan selama penulisan skripsi ini adalah versi 2.4.3.
2. *Porting* Kode KIRI *Dashboard Server Side* yang dilakukan adalah berdasarkan versi terbaru dari Github dengan *username*: “pascalalfadian”[4].

### 1.5 Metode Penelitian

Berikut adalah metode penelitian yang digunakan dalam pembuatan skripsi ini:

1. Melakukan studi literatur mengenai kode KIRI *Dashboard Server Side*, MySQL Spatial Extensions dan Play Framework.
2. Menganalisis teori-teori untuk membangun KIRI *Dashboard Server Side* dalam bahasa Java dengan menggunakan Play Framework.
3. Merancang KIRI *Dashboard Server Side* dalam bahasa Java dengan menggunakan Play Framework.
4. Melakukan *porting* kode situs web KIRI *Dashboard Server Side* menjadi Java dengan menggunakan Play Framework.
5. Melakukan pengujian terhadap fitur-fitur yang sudah dibuat.

### 1.6 Sistematika Penulisan

Setiap bab dalam penulisan ini memiliki sistematika yang dijelaskan ke dalam poin-poin sebagai berikut:

1. Bab 1: Pendahuluan, yaitu membahas mengenai gambaran umum skripsi ini yang berisi tentang latar belakang, rumusan masalah, tujuan, batasan masalah, metode penelitian dan sistematika penulisan.

- 
2. Bab 2: Dasar Teori, yaitu membahas mengenai teori-teori yang mendukung berjalannya skripsi ini yang berisi tentang kode KIRI *Dashboard Server Side*, penggunaan MySQL Spatial Extensions, dan Play Framework.
  3. Bab 3: Analisis, yaitu membahas mengenai analisa masalah yang berisi tentang kode KIRI *Dashboard Server Side* beserta kekurangan-kekurangannya dan cara melakukan *porting* bahasa PHP menjadi Java/Play Framework kode KIRI *Dashboard Server Side*.





## BAB 2

### DASAR TEORI

#### 2.1 Kode KIRI *Dashboard Server Side*

#### 2.2 MySQL Spatial Extensions

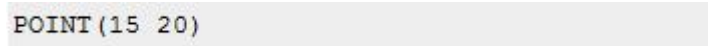
Sarana geografi[6] adalah segala bentuk yang ada di bumi yang memiliki lokasi sebagai penunjuk letak keberadaannya. Geometri adalah teknik yang digunakan untuk mengukur sebuah sarana geografi. Dengan geometri, setiap sarana geografi dapat dinyatakan dengan sebuah titik atau gabungan dari banyak titik (garis). Sarana geografi dapat berupa:

1. Sesuatu yang nyata, contohnya adalah gunung, kolam, kota, dll.
2. Suatu cakupan ruang yang tidak dapat dilihat, contohnya adalah pembagian batas daerah, cuaca, dll.
3. Suatu cakupan ruang yang dapat dilihat, contohnya adalah jalan, persimpangan, dll.

MySQL adalah salah satu perangkat lunak yang digunakan untuk mengatur data-data (*database*) suatu situs web. Bentuk MySQL adalah sekumpulan tabel-tabel yang umumnya memiliki hubungan antar satu dengan yang lainnya. Setiap tabel pada MySQL memiliki kolom dan baris. Kolom pada MySQL menyatakan daftar jenis baris yang ingin dibuat dan baris menyatakan banyaknya data yang ada dalam tabel.

Penamaan suatu kolom dalam MySQL membutuhkan penentuan jenis tipe data yang akan digunakan dalam kolom tersebut. Dalam MySQL terdapat tipe-tipe data yang umum digunakan seperti varchar untuk menyimpan huruf atau kata, int untuk menyimpan angka, boolean untuk menyimpan nilai “true” atau “false”, dan tipe data lainnya. MySQL Spatial Extensions adalah perluasan dari tipe-tipe data yang disediakan MySQL untuk menyatakan nilai geometri dari suatu sarana geografi. Terdapat 3 tipe data umum dalam MySQL Spatial Extensions, yaitu Point, LineString, dan Polygon.

##### 2.2.1 Point



```
POINT(15 20)
```

Gambar 2.1: Contoh penggunaan tipe data Point[1]

Point merupakan salah satu tipe data MySQL Spatial Extensions untuk menyatakan suatu sarana geografi dalam suatu titik yang dinyatakan dalam 2 buah nilai koordinat a dan b dimana a menyatakan letak vertikal dan b menyatakan letak horizontal.

### 2.2.2 LineString

```
LINESTRING(0 0, 10 10, 20 25, 50 60)
```

Gambar 2.2: Contoh penggunaan tipe data LineString[1]

LineString merupakan salah satu tipe data MySQL Spatial Extensions untuk menyatakan suatu sarana geografi dalam suatu garis yang dinyatakan dalam sekumpulan titik (Point). Garis ditarik berdasarkan urutan dari titik 1 ke titik 2, dari titik 2 ke titik 3, dan seterusnya.

### 2.2.3 Polygon

```
POLYGON((0 0,10 0,10 10,0 10,0 0),(5 5,7 5,7 7,5 7, 5 5))
```

Gambar 2.3: Contoh penggunaan tipe data Polygon[1]

Polygon merupakan salah satu tipe data MySQL Spatial Extensions untuk menyatakan suatu sarana geografi dalam bentuk ruang 2 dimensi (memiliki luas dan keliling) yang dinyatakan dalam sekumpulan titik (Point). Mirip seperti LineString, garis ditarik berdasarkan urutan dari titik 1 ke titik 2, dari titik 2 ke titik 3, dan seterusnya, lalu dari titik terakhir juga ditarik garis menuju titik 1 sehingga nantinya akan terbentuk ruang 2 dimensi.

## 2.3 Play Framework

Play Framework adalah sekumpulan kerangka kode yang dapat digunakan untuk membangun suatu situs web. Play Framework tidak hanya menggunakan bahasa Java dalam pembuatannya. Bahasa Scala juga digunakan Play Framework dalam beberapa bagian seperti bagian *view* dan *route*[5]. Play Framework menggunakan konsep MVC (*Model View Controller*) dalam pembuatannya. Konsep MVC pada kode membuat kode mudah dikembangkan baik secara tampilan maupun pengembangan fitur-fiturnya.

## DAFTAR REFERENSI

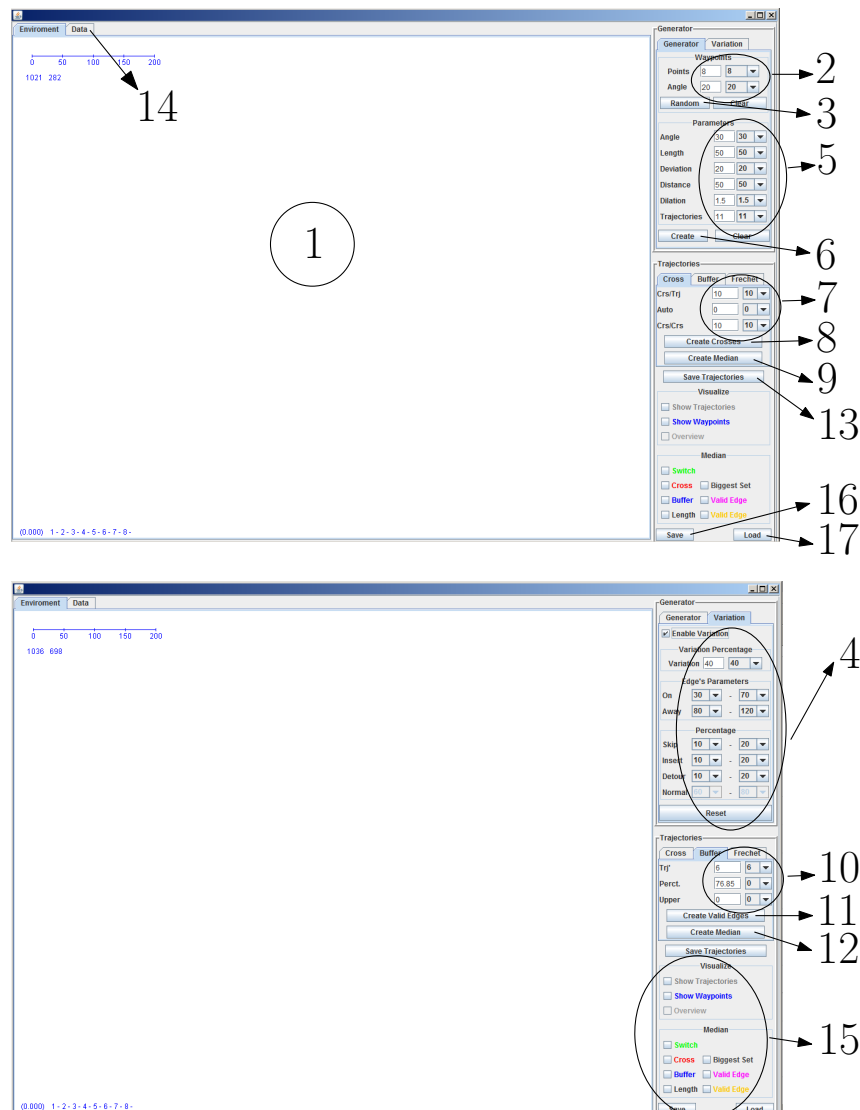
- [1] Oracle, “11.5.3.1 Supported Spatial Data Formats.” <https://dev.mysql.com/doc/refman/5.0/en/gis-data-formats.html>, 2015. [Online; diakses 1-Oktober-2015].
- [2] Pascal Alfadian, “KIRI.” <http://kiri.travel/>, 2014. [Online; diakses 1-Oktober-2015].
- [3] Pascal Alfadian, “KIRI.” <https://dev.kiri.travel/bukitjarian/>, 2014. [Online; diakses 1-Oktober-2015].
- [4] Pascal Alfadian, “TirtayasaGH.” <https://github.com/pascalalfadian/TirtayasaGH>, 2014. [Online; diakses 1-Oktober-2015].
- [5] N. Leroux and S. D. Kaper, *Play for Java*. Manning Publications Co., 2014.
- [6] Oracle, “11.5 Extensions for Spatial Data.” <https://dev.mysql.com/doc/refman/5.0/en/spatial-extensions.html>, 2015. [Online; diakses 1-Oktober-2015].



# LAMPIRAN A

## THE PROGRAM

The interface of the program is shown in Figure A.1:



Gambar A.1: Interface of the program

Step by step to compute the median trajectory using the program:

1. Create several waypoints. Click anywhere in the “Environment” area(1) or create them automatically by setting the parameters for waypoint(2) or clicking the button “Random”(3).

2. The “Variation” tab could be used to create variations by providing values needed to make them(4).
3. Create a set of trajectories by setting all parameters(5) and clicking the button “Create”(6).
4. Compute the median using the homotopic algorithm:
  - Define all parameters needed for the homotopic algorithm(7).
  - Create crosses by clicking the “Create Crosses” button(8).
  - Compute the median by clicking the “Compute Median” button(9).
5. Compute the median using the switching method and the buffer algorithm:
  - Define all parameters needed for the buffer algorithm(10).
  - Create valid edges by clicking the “Create Valid Edges”button(11).
  - Compute the median by clicking the “Compute Median”button(12).
6. Save the resulting median by clicking the “Save Trajectories” button(13). The result is saved in the computer memory and can be seen in “Data” tab(14)
7. The set of trajectories and its median trajectories will appear in the “Environment” area(1) and the user can change what to display by selecting various choices in “Visualize” and “Median” area(15).
8. To save all data to the disk, click the “Save”(16) button. A file dialog menu will appear.
9. To load data from the disk, click the “Load”(17) button.

# LAMPIRAN B

## THE SOURCE CODE

Listing B.1: MyFurSet.java

```

1  |
2  | import java.util.ArrayList;
3  | import java.util.Collections;
4  | import java.util.HashSet;
5  |
6  | /**
7  |  *
8  |  * @author Lionov
9  |  */
10 |
11 | //class for set of vertices close to furthest edge
12 | public class MyFurSet {
13 |     protected int id; //id of the set
14 |     protected MyEdge FurthestEdge; //the furthest edge
15 |     protected HashSet<MyVertex> set; //set of vertices close to furthest edge
16 |     protected ArrayList<ArrayList<Integer>> ordered; //list of all vertices in the set for each
17 |         trajectory
18 |     protected ArrayList<Integer> closeID; //store the ID of all vertices
19 |     protected ArrayList<Double> closeDist; //store the distance of all vertices
20 |     protected int totaltrj; //total trajectories in the set
21 |
22 |     /**
23 |      * Constructor
24 |      * @param id : id of the set
25 |      * @param totaltrj : total number of trajectories in the set
26 |      * @param FurthestEdge : the furthest edge
27 |      */
28 |     public MyFurSet(int id,int totaltrj,MyEdge FurthestEdge) {
29 |         this.id = id;
30 |         this.totaltrj = totaltrj;
31 |         this.FurthestEdge = FurthestEdge;
32 |         set = new HashSet<MyVertex>();
33 |         ordered = new ArrayList<ArrayList<Integer>>();
34 |         for (int i=0;i<totaltrj;i++) ordered.add(new ArrayList<Integer>());
35 |         closeID = new ArrayList<Integer>(totaltrj);
36 |         closeDist = new ArrayList<Double>(totaltrj);
37 |         for (int i = 0; i < totaltrj; i++) {
38 |             closeID.add(-1);
39 |             closeDist.add(Double.MAX_VALUE);
40 |         }
41 |     }
42 |
43 |     /**
44 |      * set a vertex into the set
45 |      * @param v : vertex to be added to the set
46 |      */
47 |     public void add(MyVertex v) {
48 |         set.add(v);
49 |     }
50 |
51 |     /**
52 |      * check whether vertex v is a member of the set
53 |      * @param v : vertex to be checked
54 |      * @return true if v is a member of the set , false otherwise
55 |      */
56 |     public boolean contains(MyVertex v) {
57 |         return this.set.contains(v);
58 |     }

```