CHAPTER V

IMPLEMENTATION AND TESTING

5. 1 Implementation

First before open the application, make sure that lampp is already installed. After that, to start the lampp with typing:

/opt/lamp/lamp start

When the user start the application by open the browser and type the address:

localhost/tsp/index.html

Then user needs to drop markers by select the cities they want to calculate the route. When user already drop the markers, user should press the calculate button and get the result.

5.1.1 Step 1 - Maps and Drop Markers

This application used Google maps API and to declare the maps is as follows.

```
function initMap(center, zoom, div) {
   var myOptions = {
      zoom: zoom,
      center: center,
      mapTypeId: google.maps.MapTypeId.ROADMAP
   };
   gebMap = new google.maps.Map(div, myOptions);
   google.maps.event.addListener(gebMap, "click", function(event)
{
      tsp.addWaypoint(event.latLng, addWaypointSuccessCallback);
   });
}
```

When the maps were already declare, to put the markers function drawMarkers is needed. The function is as follows.

```
function drawMarker(latlng, addr, label, num) {
   var icon;
   icon = new google.maps.MarkerImage("iconsnew/red" + (num +
1) + ".png");
   var marker = new google.maps.Marker({
      position: latlng,
      icon: icon,
      map: gebMap });
   google.maps.event.addListener(marker, 'click', function(event) {
      var addrStr = (addr == null) ? "" : addr + "<br>";
      var labelStr = (label == null) ? "" : "<b>" + label +
"</b><br>";
      var markerInd = -1;
      for (var i = 0; i < markers.length; ++i) {
         if (markers[i] != null &&
marker.getPosition().equals(markers[i].getPosition())) {
            markerInd = i;
            break;
         }
      }
```

5.1.2 Step 2 - Distance Matrix

Once drop the markers is already, the latitude and longitude to calculate

The distance between the city and save the distance in matrix. Distance matrix

Function is as follows.

```
var distancesMatrixStr = "";
    for (var i = 0; i < dist.length; ++i) {
    for (var j = 0; j < dist[i].length; ++j) {
      distancesMatrixStr += parseInt(dist[i][j]/1000);
      if (j == dist[i].length - 1) {
        distancesMatrixStr += "\n";
      } else {
        distancesMatrixStr += ", ";
      }
    }
  }

document.getElementById("distancesData").innerHTML =
    "<textarea name='csvDistancesMatrix' rows='10'
cols='20'>"
    + distancesMatrixStr + "</textarea><br>";
```

5.1.3 Step 3 - Calculate Ant Colony Optimization

Third step is to calculate the distance matrix with ant colony optimization. Calculate the probabilities with declare the parameters. Ant colony is as function.

```
function tspAntColonyK2(mode) {
  var alfa = 0.1; // The importance of the previous trails
      document.getElementById("alfa").innerHTML = alfa;
  var beta = 2.0; // The importance of the durations
      document.getElementById("beta").innerHTML = beta;
  var rho = 0.1;  // The decay rate of the pheromone trails
  var asymptoteFactor = 0.9; // The sharpness of the reward as the
solutions approach the best solution
  var pher = new Array();
  var nextPher = new Array();
  var prob = new Array();
  var numAnts = 10;
  var numWaves = 10;
  for (var i = 0; i < numActive; ++i) {
    pher[i] = new Array();
    nextPher[i] = new Array();
  }
  for (var i = 0; i < numActive; ++i) {
    for (var j = 0; j < numActive; ++j) {
      pher[i][j] = 1;
      nextPher[i][j] = 0.0;
    }
  }

  var lastNode = 0;
  var startNode = 0;
  var numSteps = numActive - 1;
  var numValidDests = numActive;
  if (mode == 1) {
    lastNode = numActive - 1;
    numSteps = numActive - 2;
    numValidDests = numActive - 1;
  }
  for (var wave = 0; wave < numWaves; ++wave) {
    for (var ant = 0; ant < numAnts; ++ant) {
      var curr = startNode;
      var currDist = 0;
      for (var i = 0; i < numActive; ++i) {
        visited[i] = false;
      }
```

5.1.4 For the direction, Google maps API is needed. And the direction is as follows.

```
function directions(m) {
   jQuery('#dialogProgress').dialog('open');
   mode = m;

tsp.setTravelMode(google.maps.DirectionsTravelMode.DRIVIN
G);
   tsp.setOnProgressCallback(onProgressCallback);
   if (m == 0)
      tsp.solveRoundTrip(onSolveCallback);

}
```

5.2 Testing

This test consists of 5 cities which is Semarang- Pekalongan- Salatiga-Magelang-Yogyakarta-Semarang.

	semarang	pekalongan	salatiga	magelang	yogya
semarang	0	96	49	75	123
pekalongan	95	0	138	165	213
salatiga	48	138	0	55	97
magelang	74	132	55	0	49
yogyakarta	123	181	97	50	0

Figure 5.2 Distance Matrix

	semarang	pekalongan	salatiga	magelang	yogya
semarang	0	0.010416667	0.020408163	0.013333333	0.00813
pekalongan	0.010526316	0	0.007246377	0.006060606	0.004695
salatiga	0.020833333	0.007246377	0	0.018181818	0.010309
magelang	0.013513514	0.007575758	0.018181818	0	0.020408
yogyakarta	0.008130081	0.005524862	0.010309278	0.02	0

Figure 5.2.1 1/distance (nij)

Tij	**0.9**
alfa	**0.1**
beta	**2**
rho	**0.1**
semut	**10**

Figure 5.2.2 initial parameters

semut 1	
	probabilitas komulatif
Probabilitas	0.04705942
semarang	0
pekalongan	0.002281577
salatiga	0.008757608
magelang	0.003738136
yogyakarta	0.001389848
bilangan random	0.38
semarang-salatiga	

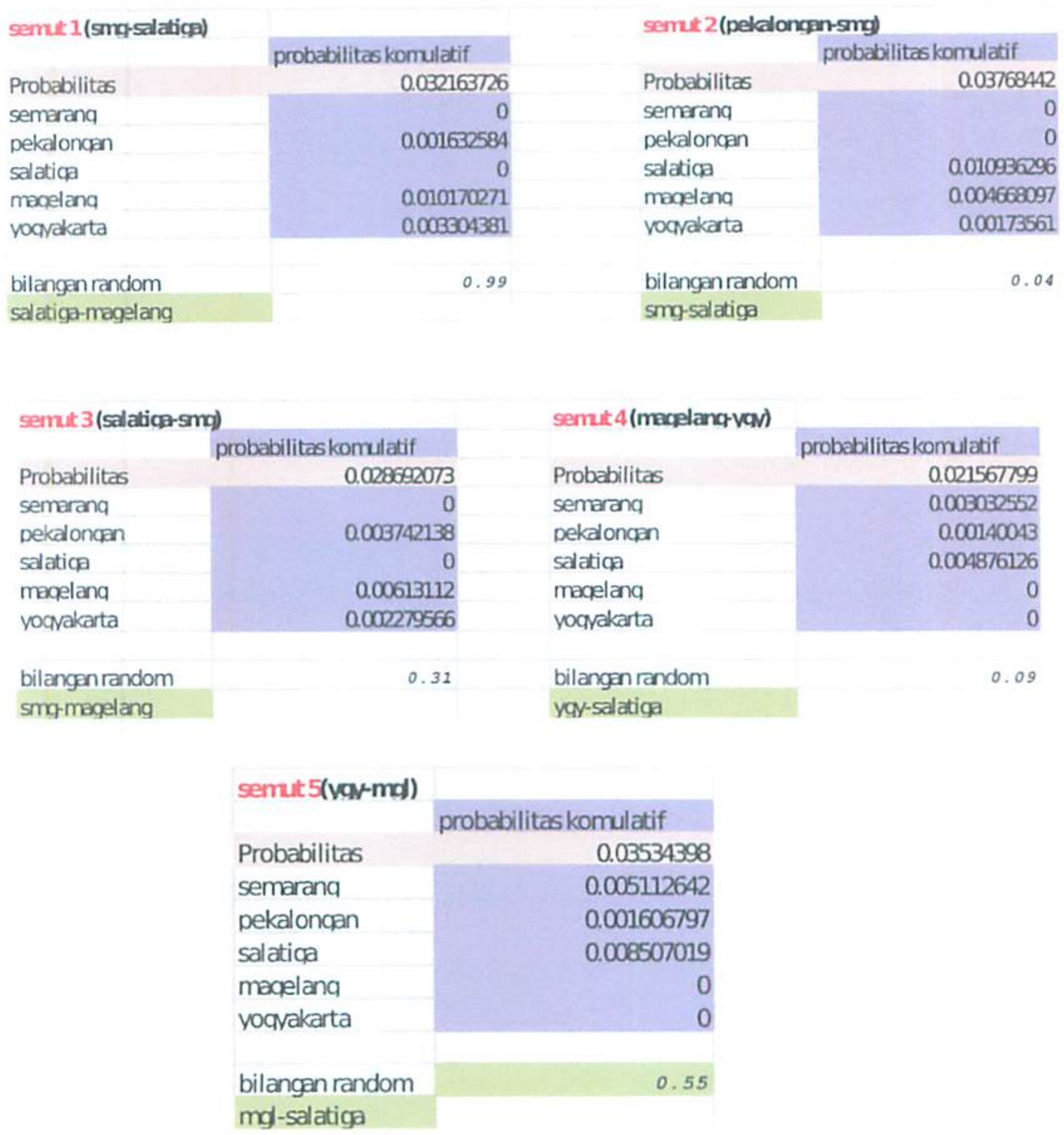
semut 2	
	probabilitas komulatif
Probabilitas	0.025675321
semarang	0.004270327
pekalongan	0
salatiga	0.002023719
magelang	0.0014156
yogyakarta	0.000849472
bilangan random	0.87
pekalongan-smg	

semut 3	
	probabilitas komulatif
Probabilitas	0.050913726
semarang	0.008435424
pekalongan	0.001020543
salatiga	0
magelang	0.006424865
yogyakarta	0.002065598
bilangan random	0.56
salatiga-semarang	

semut 4	
	probabilitas komulatif
Probabilitas	0.053711327
semarang	0.003364302
pekalongan	0.00105733
salatiga	0.00609022
magelang	0
yogyakarta	0.007673018
bilangan random	0.74
magelang-yogyakarta	

semut 5	
	probabilitas komulatif
Probabilitas	0.039567799
semarang	0.001652997
pekalongan	0.000763353
salatiga	0.002657902
magelang	0.010003278
yogyakarta	0
bilangan random	0.48
ygy-magelang	

Figure 5.2.3 Iteration 1



semut 1 (smg-salatiga)

	probabilitas komulatif
Probabilitas	0.032163726
semarang	0
pekalongan	0.001632584
salatiga	0
magelang	0.010170271
yogyakarta	0.003304381
bilangan random	0.99
salatiga-magelang	

semut 2 (pekalongan-smg)

	probabilitas komulatif
Probabilitas	0.03768442
semarang	0
pekalongan	0
salatiga	0.010936296
magelang	0.004668097
yogyakarta	0.00173561
bilangan random	0.04
smg-salatiga	

semut 3 (salatiga-smg)

	probabilitas komulatif
Probabilitas	0.028692073
semarang	0
pekalongan	0.003742138
salatiga	0
magelang	0.00613112
yogyakarta	0.002279566
bilangan random	0.31
smg-magelang	

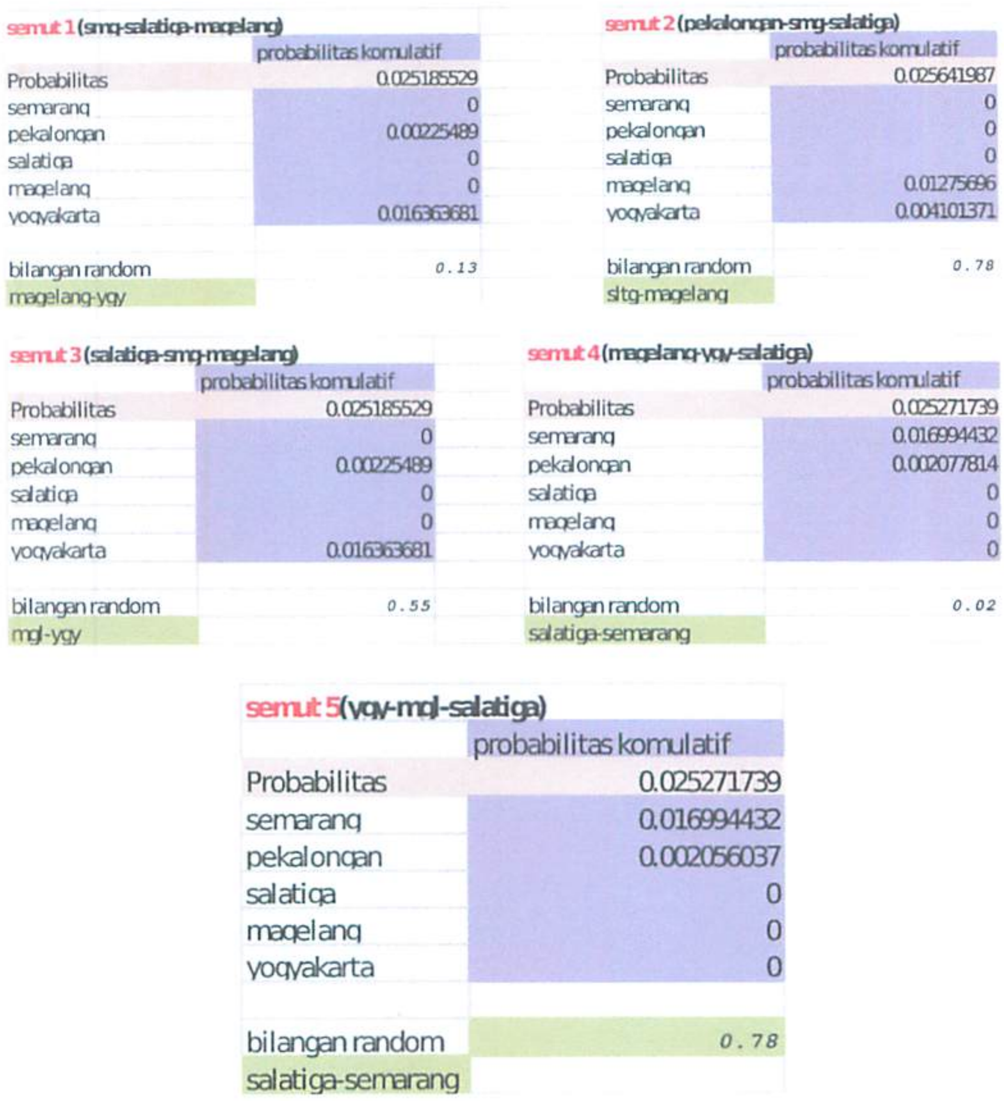
semut 4 (magelang-ygy)

	probabilitas komulatif
Probabilitas	0.021567799
semarang	0.003032552
pekalongan	0.00140043
salatiga	0.004876126
magelang	0
yogyakarta	0
bilangan random	0.09
ygy-salatiga	

semut 5(ygy-mgl)

	probabilitas komulatif
Probabilitas	0.03534398
semarang	0.005112642
pekalongan	0.001606797
salatiga	0.008507019
magelang	0
yogyakarta	0
bilangan random	0.55
mgl-salatiga	

Figure 5.2.4 Iteration 2



semut 1 (smg-salatiga-magelang)

	probabilitas komulatif
Probabilitas	0.025185529
semarang	0
pekalongan	0.00225489
salatiga	0
magelang	0
yogyakarta	0.016363681
bilangan random	0.13
magelang-ygy	

semut 2 (pekalongan-smg-salatiga)

	probabilitas komulatif
Probabilitas	0.025641987
semarang	0
pekalongan	0
salatiga	0
magelang	0.01275696
yogyakarta	0.004101371
bilangan random	0.78
sltg-magelang	

semut 3 (salatiga-smg-magelang)

	probabilitas komulatif
Probabilitas	0.025185529
semarang	0
pekalongan	0.00225489
salatiga	0
magelang	0
yogyakarta	0.016363681
bilangan random	0.55
mgl-ygy	

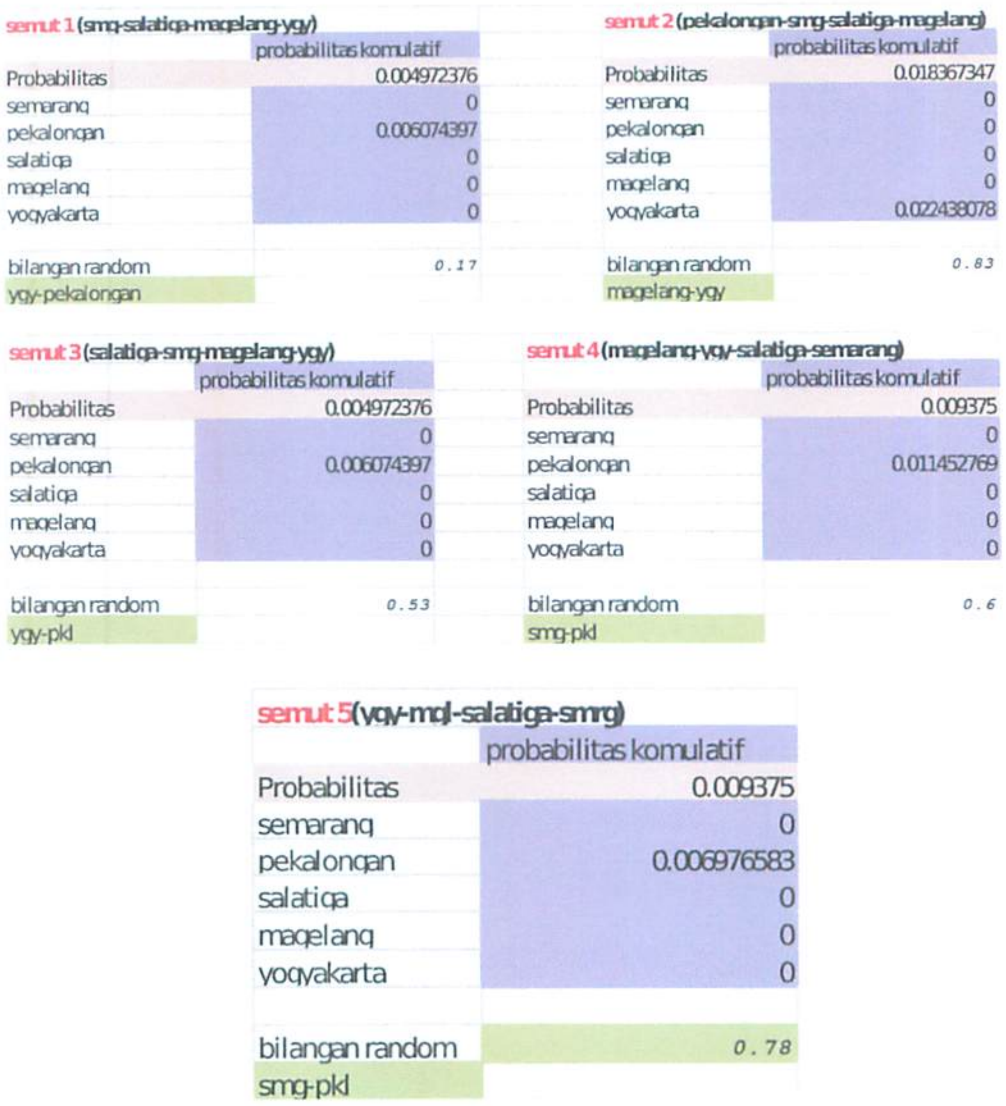
semut 4 (magelang-ygy-salatiga)

	probabilitas komulatif
Probabilitas	0.025271739
semarang	0.016994432
pekalongan	0.002077814
salatiga	0
magelang	0
yogyakarta	0
bilangan random	0.02
salatiga-semarang	

semut 5(ygy-mgl-salatiga)

	probabilitas komulatif
Probabilitas	0.025271739
semarang	0.016994432
pekalongan	0.002056037
salatiga	0
magelang	0
yogyakarta	0
bilangan random	0.78
salatiga-semarang	

Figure 5.2.5 Iteration 3



semut 1 (smg-salatiga-magelang-ygy)	
	probabilitas komulatif
Probabilitas	0.004972376
semarang	0
pekalongan	0.006074397
salatiga	0
magelang	0
yogyakarta	0
bilangan random	0.17
ygy-pekalongan	

semut 2 (pekalongan-smg-salatiga-magelang)	
	probabilitas komulatif
Probabilitas	0.018367347
semarang	0
pekalongan	0
salatiga	0
magelang	0
yogyakarta	0.022438078
bilangan random	0.83
magelang-ygy	

semut 3 (salatiga-smg-magelang-ygy)	
	probabilitas komulatif
Probabilitas	0.004972376
semarang	0
pekalongan	0.006074397
salatiga	0
magelang	0
yogyakarta	0
bilangan random	0.53
ygy-pkl	

semut 4 (magelang-ygy-salatiga-semarang)	
	probabilitas komulatif
Probabilitas	0.009375
semarang	0
pekalongan	0.011452769
salatiga	0
magelang	0
yogyakarta	0
bilangan random	0.6
smg-pkl	

semut 5(ygy-mgl-salatiga-smrg)	
	probabilitas komulatif
Probabilitas	0.009375
semarang	0
pekalongan	0.006976583
salatiga	0
magelang	0
yogyakarta	0
bilangan random	0.78
smg-pkl	

Figure 5.2.6 Iteration 4

semut 1(semarang-salatiga-magelang-yogya-pekalongan)	429
semut2(pekalongan-semarang-salatiga-magelang-jogja)	429
semut3(salatiga-semarang-magelang-yogya-pekalongan)	491
semut4(magelang-yogya-salatiga-semarang-pekalongan)	455
semut5(yogyakarta-magelang-salatiga-semarang-pekalongan)	462

Figure 5.2.7 Result

Semarang-salatiga-magelang-yogya-pekalongan-semarang	429
Semarang-magelang-salatiga-yogya-pekalongan-semarang	503
Semarang-yogya-pekalongan-magelang-salatiga-semarang	572
Semarang-pekalongan-magelang-salatiga-yogya-semarang	463
Semarang-salatiga-yogya-pekalongan-magelang-semarang	566
Semarang-magelang-yogya-salatiga-pekalongan-semarang	454
Semarang-yogya-magelang-salatiga-pekalongan-semarang	461
Semarang-pekalongan-salatiga-yogya-magelang-semarang	455
Semarang-salatiga-pekalongan-yogya-magelang-semarang	524
Semarang-magelang-pekalongan-salatiga-yogya-semarang	565
Semarang-yogya-salatiga-pekalongan-magelang-semarang	597
Semarang-pekalongan-yogya-salatiga-magelang-semarang	535
Semarang-magelang-pekalongan-salatiga-yogya-semarang	565
Semarang-salatiga-magelang-pekalongan-yogya-semarang	572
Semarang-magelang-salatiga-pekalongan-yogya-semarang	514
Semarang-yogya-pekalongan-salatiga-magelang-semarang	571
Semarang-pekalongan-magelang-yogya-salatiga-semarang	455
Semarang-salatiga-yogya-magelang-pekalongan-semarang	539
Semarang-magelang-pekalongan-yogya-salatiga-semarang	565
Semarang-yogya-salatiga-magelang-pekalongan-semarang	502
Semarang-pekalongan-salatiga-magelang-yogya-semarang	461
Semarang-salatiga-pekalongan-magelang-yogya-semarang	572
Semarang-magelang-yogya-pekalongan-salatiga-semarang	491
Semarang-yogya-pekalongan-salatiga-magelang-semarang	571

Figure 5.2.8 24 Possibilities

5.3 Main Interface Window

This is the first and main interface of Traveling Sales Problem using Ant Colony Algorithm. It has four steps:

Step 1. Select the cities by drop the marker

Step 2. Press Calculate Ant Colony button to get the result

Step 3. To look up the distance matrix, open the export tab

Step 4. Reset and start again with select the cities.

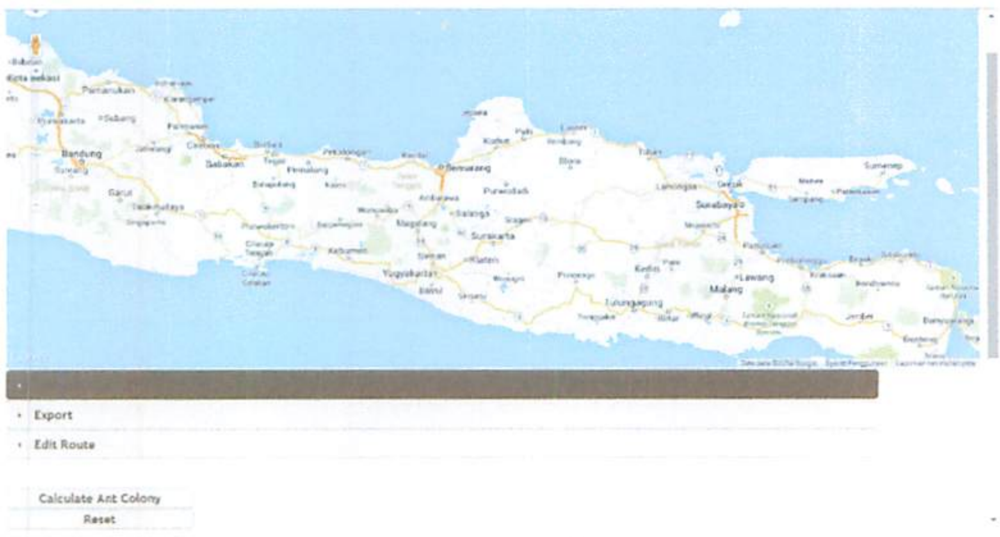


Figure 5.3.1 Main Interface

The main page of the application. Its use Google maps API for show the maps and button for run the function.

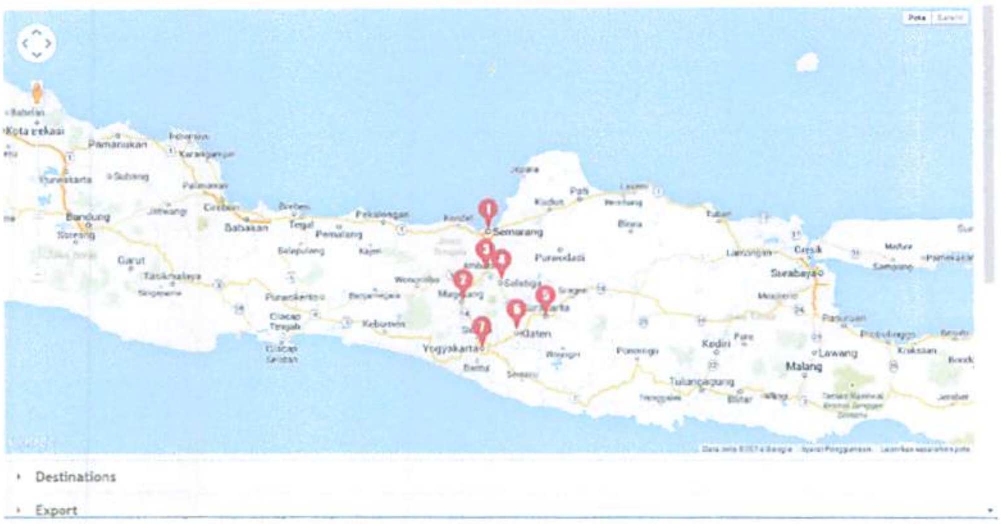


Figure 5.3.2 Drop Markers

Select the cities by drop the markers. Each city will had a marker with each number.

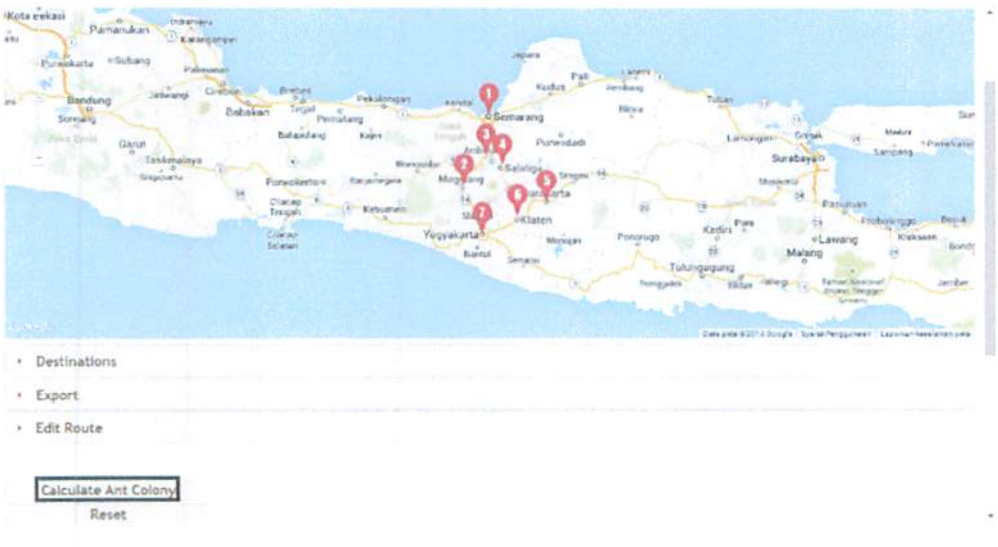


Figure 5.3.3 Calculate Ant Colony

In the *Figure 5.3.2 Drop Markers*, the numbers means the order you select the cities. Press Calculate Ant colony and get the result.

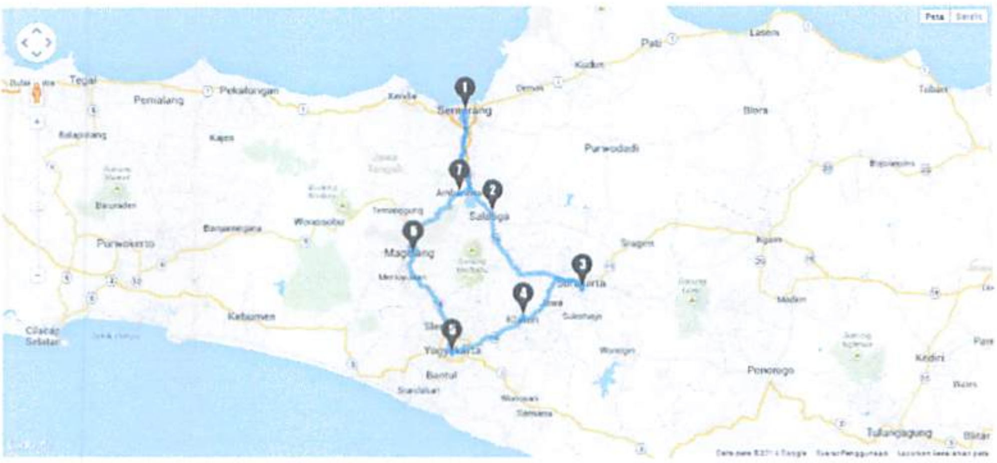
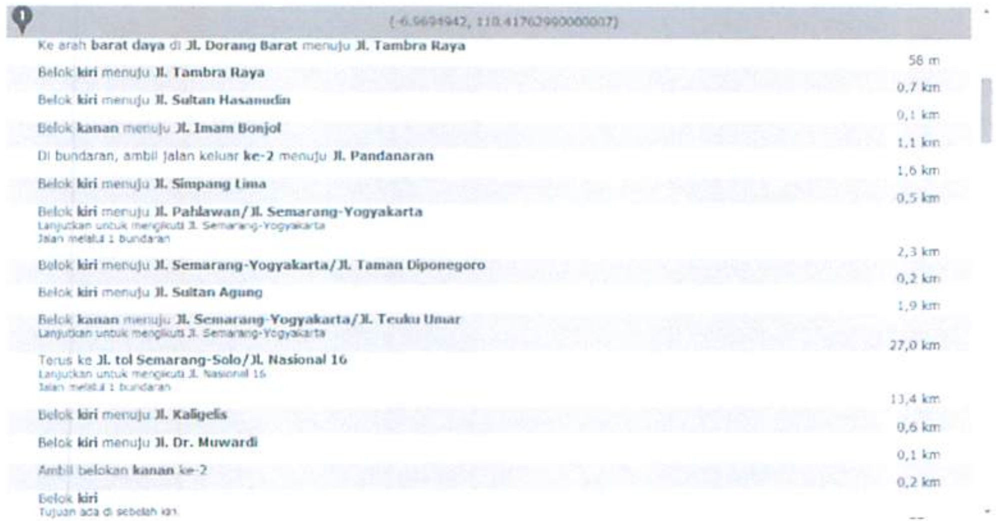


Figure 5.3.4 Result Optimization Route

Third step we will get the result for optimization route and the direction from Google maps direction.

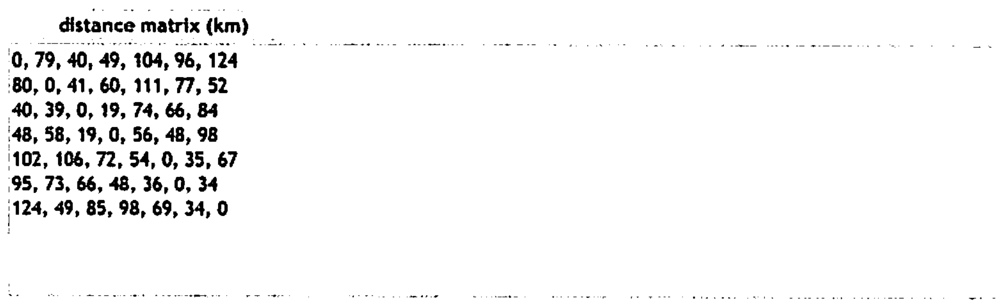


(-6.9694942, 110.41762990000007)

Ke arah **barat daya** di **Jl. Dorang Barat** menuju **Jl. Tambra Raya**	56 m
Belok **kiri** menuju **Jl. Tambra Raya**	0,7 km
Belok **kiri** menuju **Jl. Sultan Hasanudin**	0,1 km
Belok **kanan** menuju **Jl. Imam Bonjol**	1,1 km
Di bundaran, ambil jalan keluar **ke-2** menuju **Jl. Pandanaran**	1,6 km
Belok **kiri** menuju **Jl. Simpang Lima**	0,5 km
Belok **kiri** menuju **Jl. Pahlawan/Jl. Semarang-Yogyakarta** Lanjutkan untuk mengikuti Jl. Semarang-Yogyakarta Jalan melalui 1 bundaran	2,3 km
Belok **kiri** menuju **Jl. Semarang-Yogyakarta/Jl. Taman Diponegoro**	0,2 km
Belok **kiri** menuju **Jl. Sultan Agung**	1,9 km
Belok **kanan** menuju **Jl. Semarang-Yogyakarta/Jl. Teuku Umar** Lanjutkan untuk mengikuti Jl. Semarang-Yogyakarta	27,0 km
Terus ke **Jl. tol Semarang-Solo/Jl. Nasional 16** Lanjutkan untuk mengikuti Jl. Nasional 16 Jalan melalui 1 bundaran	13,4 km
Belok **kiri** menuju **Jl. Kaligelis**	0,6 km
Belok **kiri** menuju **Jl. Dr. Muwardi**	0,1 km
Ambil belokan **kanan** ke-2	0,2 km
Belok **kiri** Tujuan ada di sebelah kiri.	--

Figure 5.3.5 Direction

The result page will give an optimization route and the direction by Google direction.



```
distance matrix (km)
0, 79, 40, 49, 104, 96, 124
80, 0, 41, 60, 111, 77, 52
40, 39, 0, 19, 74, 66, 84
48, 58, 19, 0, 56, 48, 98
102, 106, 72, 54, 0, 35, 67
95, 73, 66, 48, 36, 0, 34
124, 49, 85, 98, 69, 34, 0
```

Figure 5.3.6 Distance Matrix

To look up the distance matrix of the cities, we can press export tab and the list is as follows.