# GEOFENCING APPLICATION DEVELOPMENT IN MONITORING VEHICLE DELIVERY AT PT. CANNONEX INDONESIA

Joko Priono [1], Eko Budi Setiawan [2]

[1,2] Program Studi Teknik Informatika, Universitas Komputer Indonesia
Jl. Dipatiukur 112-114 Bandung
E-mail : joko.priono@yahoo.com [1], eko@email.unikom.ac.id [2]

## ABSTRACT

This study discusses the application of geofencing method in the case of monitoring a vehicle delivery. Geofencing itself is a technology that allows users or systems to conduct remote monitoring of an area that has been determined. Geofence is a virtual area created based on geographical coordinates. This geofencing will use Google Maps APIs. To activate geofence, a trigger is required. This trigger is in form of a GPS Tracking Unit that provides the coordinate position of a target. By using these two methods, the system will be able to know the existence and what areas have been passed by a vehicle delivery that is in progress.

**Keyword:** Geofencing, GPS Tracking, Google Maps, Monitoring.

## 1. INTRODUCTION

PT. Cannonex Indonesia is a private company that focuses on the delivery of cars, heavy equipment and heavy cargo projects to almost all regions in Indonesia. In times of growing economic development, the company continues to expand its business using the available technology. One of the development is done on the service offered by the company which is door-to-door which is the delivery of vehicles by picking up and delivering vehicles to almost all regions in Indonesia. And this shipment requires an operations management in charge of monitoring the shipment. The system in managing the delivery of vehicles would become a very important aspect to ensure the smoothness of the delivery.

Geofence is an innovative technology that utilizes GPS coordinates (Global Positioning System) by determining geographical or parameter boundaries virtually. Geofencing itself allows one to remotely monitor from a specific geographic area. If the parameters of this geofences are active, the activity status will be automatically saved or recorded [1]. As the driver exits the specified parameters, how long the driver is within the parameters, measures the driving speed in a parameter, or gives information if the driver has entered a particular road or region. This will be very helpful especially for companies engaged in logistics monitoring the delivery of vehicles and riders in various locations.

The current system also does not have information or data about the delivery route of the vehicle making it difficult to plan the route. This is influential in the smoothness of the delivery of vehicles and to find out the routes that are considered dangerous to be passed by vehicles. Information that are now available are only by talking to the driver who has made the delivery.

Reclus and Drouard [1] have presented research on the basic concepts of geofencing and some applications based on this technology in transportation and logistics sector management. Some applications based on geofencing proposed by this research are vehicle location monitoring based on Point of Interest (POI), Fleet Management, and Defense & Security. In the Nait-Sidi-Moh [2] study, it introduces the concept of a system or platform that integrates geofencing techniques for real-time tracking with mobile devices.

This study is expected to build a system that can help provide position information and travel routes, so as to improve work efficiency that will impact on spending savings and provide up-to-date vehicle info.

## 2. CONTENTS OF RESEARCH

### 2.1 Monitoring

Monitoring can be defined as an activity cycle that includes collecting, reviewing, reporting and action on information a process is being implemented [3]. Monitoring activities are more focused on monitoring the activities being carried out.

Monitoring is done by obtaining information regularly based on certain indicators, to find out whether the ongoing activities in accordance with the plans and procedures that have been determined

### 2.2 GPS Tracking

GPS tracking device is a tool used as a determinant of the location of a target that is usually carried by a vehicle or a person. Location data on GPS tracking devices is the geographic coordinates that are sent to the server, as shown in Figure 1, which is part of the tracking system components. The delivery of location data will use the mobile network (GPRS or

SMS), radio or satellite modem embedded in the device in use.

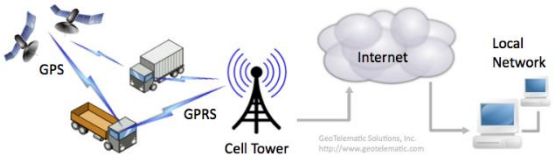



**Figure 1.** GPS Tracking

This device provides the unit asset location that can be displayed with the map background both in real time. GPS Tracker to be used is an android application that will be designed to transmit GPS data in the form of latitude and longitude coordinates to the server in real time with internet connectivity. The android smartphone that are used must has a GPS module device and an internet connection.

### 2.3 Geofencing

Geofencing is a feature in software programs used in conjunction with global positioning system (GPS) in determining the geographic boundaries or virtual parameters of a map. Programs that use geofencing can set up a triggers that can provide information or notifications when a particular target enters or exits a predefined range. The target that triggers the trigger is usually the geographic coordinate of a GPS that is sent regularly to a data server or server.

Geographic coordinates are used to determine the target position and also to create a certain area boundary (mapping) as a virtual fence (geofence) of an area [1]. The system will determine the position of the target tracked outside or inside the geofencing area. This technology can also allow for the detection of proximity between the target position with a specific geofencing area.




**Figure 2.** Geofence

In Figure 2 above can be seen a target or a vehicle that enters an area that has been mapping with geofence. When the target entered the area of the geofence, it will appear a notification of event or promotional information in the trigger by the system. Geofencing can be applied to various fields related to monitoring a target or an information.

### 2.4 Geofence Area

The area size of a geofencing can range from several tens of meters up to several kilometers. The shape of the geofencing area itself may vary, from circular, square or rectangular or more complicated forms such as a polygon. Example of one form of geofencing can be seen in Figure 3.

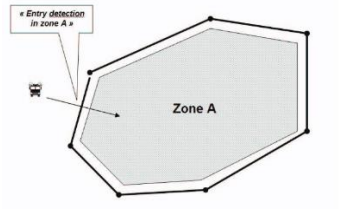



**Figure 3.** Sample Polygon Geofence

Coordinates of point or point characteristics are important in defining geofence parameters. Geofencing mechanism in determining the point is to determine the first latitude and longitude of each characteristic point. Examples such as geofencing shaped circle that has the characteristics of a central coordinate. After getting the latitude and longitude, will be determined the desired radius which is a coordinate to know the radius of the circle. Each geofencing shape has different point characteristics.

### 2.5 Problem Analysis

Problem analysis is an assumption of the problem to be described in data processing procedures. Based on interviews and observations conducted in the field of research, got some problems as follows:

a. The difficulty of the officers in monitoring the position and what path is used by the driver in the delivery of vehicles. This is because the current process has a minimalist monitoring process, example are by phone, and this is only done if necessary. And there is no way to know the position of the driver on a regular basis in the field. By using a GPS tracking and geofencing use, it will help determine the position of the driver on a regular basis as well as a predetermined driver path with pre-defined geofencing area.

b. Difficult to get information about the delivery of vehicles that can help companies in planning better delivery routes. Currently the information obtained only from the driver who has made the delivery and the information itself is not too complete. To help improve data or information that can help route planning, the data from gps tracker and geofencing will be stored and used to help workers.

### 2.5 System Analysis and Design

The system to be built is android-based mobile applications as front-end and web applications as back-end. Architecture system that are going to be built can be seen in Figure 4 below.

.

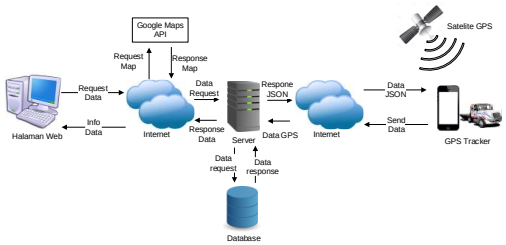



**Figure 4.** System Architecture

The mobile system will be the receiver of the GPS satellite and will send the latitude and longitude data to be stored in the database server. The web system will be used to display data from the coordinates that have been sent and see if the coordinates are inside the geofencing area. The geofencing area itself will be created through the web system and will use the library provided by Google APIs.

### 2.6 Analysis Technology of GPS Sensor

GPS Tracking is a tool used as a determinant of the location of the target. Location data on this device uses the Global Positioning System (GPS) to determine the geographic coordinates that will be sent to the server or data manager. GPS Tracking to be used is an android application that will be designed to send data coordinates latitude and longitude to the server with internet connectivity.

The app will use Google Play service location APIs that provide facilities such as location awareness in the built app. To build an app using Google Play Service APIs, dependencies are required on the Google Play service SDK.

```
1.  apply plugin: 'com.android.application'
2.      ...
3.      dependencies {
4.          compile 'com.google.android.gms:p
    lay-services-location:10.2.4'
5.      }
```

Another thing to consider in determining the location of the application is the request parameter that will be used to determine the accuracy of the location request. To save the parameter we will used LocationRequest.

```
1.  protected void createLocationRequest() {

2.      mLocationRequest = new LocationReques
    t();
3.      mLocationRequest.setInterval(10000);

4.      mLocationRequest.setFastestInterval(5
    000);
5.      mLocationRequest.setPriority(Location
    Request.PRIORITY_HIGH_ACCURACY);
6.      mLocationRequest.setSmallestDisplacem
    ent(10);
7.  }
```

Here are some descriptions of the parameters that are going to be used:

a. Update Interval
   setInterval() – this parameters sets the level of interval in the miliseconds that the application choose to accept.
b. Fastest Update Interval
   setFastestInterval() – This parameter sets the fastest rate in milliseconds when the app handles the latest location updates. This parameter is used as the highest limit of application in handling location updates.
c. Displacement Update
   setSmallestDisplacement() – This parameter sets the minimum distance between locations in handling updates in meters size. By default the distance value is 0.
d. Priority
   setPriority() – This parameter specifies the request priority.

### 2.7 Analysis Technology of Geofencing

Geofencing is used to define the virtual geographic boundaries of a map. The shape of the geofencing area itself may vary, from circular, square or rectangular or more complicated forms such as a polygon. In Google Maps APIs, this geofencing depiction uses the Library provided by the JavaScript API called Drawing Layer. The Drawing Layer shown in Figure 5 below is a class that provides a graphical user interface for drawing circles, polygons, rectangles, polylines and markers on the maps.

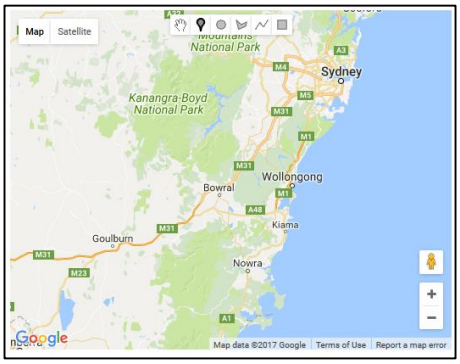



**Figure 5.** Drawing Layer

To use the Library functionality, the app must first load the parameters in the Google Maps API.

```
1.  <script type="text/javascript"
2.    src="https://maps.googleapis.com/maps/ap
    i/js?key=YOUR_API_KEY&libraries=drawing">

3.  </script>
```

After the parameters are added, the Drawing Manager object will be created as follows:

```
1.  var drawingManager = new google.maps.drawi
    ng.DrawingManager();
2.  drawingManager.setMap(map);
```

Drawing Manager has options that will define some of the controls to be displayed, control positions, and state of the image at initialization.

```
1.  var drawingManager = new google.maps.drawi
    ng.DrawingManager({
2.    drawingMode: google.maps.drawing.Overlay
    Type.MARKER,
3.    drawingControl: true,
4.    drawingControlOptions: {
5.      position: google.maps.ControlPosition.
    TOP_CENTER,
6.      drawingModes: ['marker', 'circle', 'po
    lygon', 'polyline', 'rectangle']
7.    },
8.    circleOptions: {
9.         fillColor: '#ffff00',
10.     fillOpacity: 1,
11.     strokeWeight: 5,
12.     clickable: false,
13.   }
14. });
```

### 2.8 Area Geofencing Analysis

Geofencing depictions on application maps have several forms, such as circles, lines, squares and polygons. This form of representation is an object on a map bound to the latitude and longitude coordinates.

Depictions may also be configured so that users can edit or drag the shape of the image. Figure 6 below is an example of a line drawing on google maps.




**Figure 6.** Drawing Line

```
1.  function initMap() {
2.    var map = new google.maps.Map(document.
    getElementById('map'), {
3.      zoom: 3,
4.      center: {lat: 0, lng: -180},
5.      mapTypeId: 'terrain'
6.    });
7.
8.    var flightPlanCoordinates = [
9.      {lat: 37.772, lng: -122.214},
10.     {lat: 21.291, lng: -157.821},
11.     {lat: -18.142, lng: 178.431},
12.     {lat: -27.467, lng: 153.027}
13.   ];
14.   var flightPath = new google.maps.Polyli
    ne({
15.     path: flightPlanCoordinates,
16.     strokeColor: '#FF0000',
17.     strokeOpacity: 1.0,
18.     strokeWeight: 2
19.   });
20.
21.   flightPath.setMap(map);
22. }
```

Figure 7 below is an example of triangular polygon depiction on google maps.

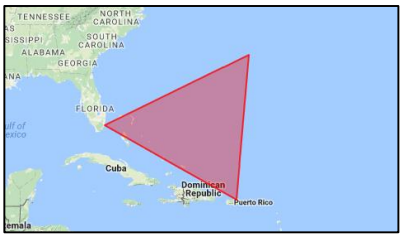



**Figure 7.** Drawing Polygon

```
1.  function initMap() {
2.    var map = new google.maps.Map(document
    .getElementById('map'), {
3.      zoom: 5,
4.      center: {lat: 24.886, lng: -
    70.268},
5.      mapTypeId: 'terrain'
6.    });
7.
8.    // Define the LatLng coordinates for t
    he polygon's path.
9.    var triangleCoords = [
10.     {lat: 25.774, lng: -80.190},
11.     {lat: 18.466, lng: -66.118},
12.     {lat: 32.321, lng: -64.757},
13.     {lat: 25.774, lng: -80.190}
14.   ];
15.
16.   // Construct the polygon.
17.   var bermudaTriangle = new google.maps.
    Polygon({
18.     paths: triangleCoords,
19.     strokeColor: '#FF0000',
20.     strokeOpacity: 0.8,
21.     strokeWeight: 2,
22.     fillColor: '#FF0000',
23.     fillOpacity: 0.35
24.   });
25.   bermudaTriangle.setMap(map);
26. }
```

Figure 8 below is an example of square depiction on google maps.

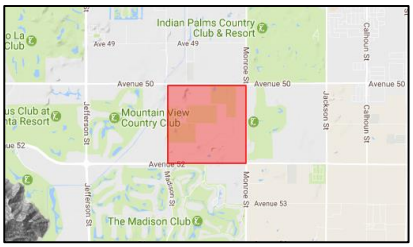



**Figure 8.** Drawing Rectangle

```
1.  function initMap() {
2.    var map = new google.maps.Map(document.ge
    tElementById('map'), {
3.      zoom: 11,
4.      center: {lat: 33.678, lng: -116.243},
5.      mapTypeId: 'terrain'
6.    });
7.
8.    var rectangle = new google.maps.Rectangle
    ({
9.      strokeColor: '#FF0000',
10.     strokeOpacity: 0.8,
11.     strokeWeight: 2,
12.     fillColor: '#FF0000',
13.     fillOpacity: 0.35,
14.     map: map,
15.     bounds: {
16.       north: 33.685,
17.       south: 33.671,
18.       east: -116.234,
19.       west: -116.251
20.     }
21.   });
```

Figure 9 below is an example of square depiction on google maps.




**Figure 9.** Drawing Circle

Each geofence portrait has the same attribute with few different characteristics. For example, for circle depiction using google.maps.Circle with some attribute that can be changed that is color, opacity, weight and others.

```
1.  var citymap = {
2.    chicago: {
3.      center: {lat: 41.878, lng: -
    87.629},population: 2714856}
4.  };
5.
6.  function initMap() {  // Create the map.
7.    var map = new
    google.maps.Map(document.getElementById('
    map'), {
8.      zoom: 4, center: {lat: 37.090, lng: -
    95.712},
9.      mapTypeId: 'terrain'
10.   });
11.
12.   // Construct the circle for each value
    in citymap.
13.   // Note: We scale the area of the
    circle based on the population.
14.   for (var city in citymap) {
15.     // Add the circle for this city to
    the map.
16.     var cityCircle = new
    google.maps.Circle({
17.       strokeColor: '#FF0000',
18.       map: map,
19.       center: citymap[city].center,
20.       radius:
    Math.sqrt(citymap[city].population) * 100
21.     });
```

## 2.9 Geofencing Analysis Detection

Geofencing detection analysis is an analysis of how the system knows the coordinates of the location sent by the GPS Tracker has entered into the geofencing area. Each geofencing detection varies, from the circle area, the square area and the polygon area. For square detection will use the method below:

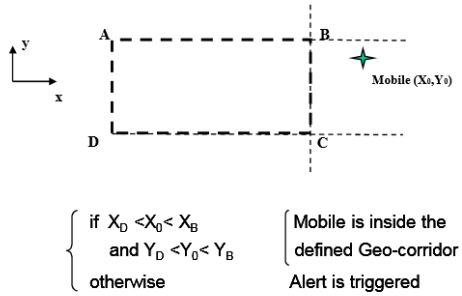



**Figure 10.** Rectangle Calculation

The calculation of square geofencing is done by looking at whether the target coordinates are within the 4 point coordinates that depict the square on the geographical map. If true, then the system will trigger that tells the target is within the geofence area. For the detection of the circle used method as in Figure 23 below.

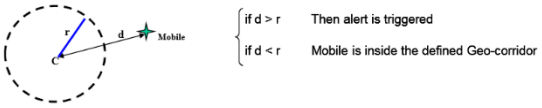



**Figure 12.** Circle Calculation

The geofencing calculations on the circle see the coordinates of the target position and the position of the coordinates of destination. If the distance between the target and destination coordinates is greater than the circle radius, then the target is outside the circle. If the value is smaller, then the target is inside the circle and the system will make a notification about that information.

Because geographically the earth is not a circle, but rather a sphere. Then the distance calculation is not on the flat surface. For that we need a calculation that determines the distance between two points on a ball surface based on latitude and longitude. The calculation to be used is haversine formula.

```
1.  function haversineGreatCircleDistance(
2.    $latitudeFrom, $longitudeFrom, $latitud
    eTo, $longitudeTo, $earthRadius = 6371000
    )
3.  {
4.    // convert from degrees to radians
5.    $latFrom = deg2rad($latitudeFrom);
6.    $lonFrom = deg2rad($longitudeFrom);
7.    $latTo = deg2rad($latitudeTo);
8.    $lonTo = deg2rad($longitudeTo);
9.
10.   $latDelta = $latTo - $latFrom;
11.   $lonDelta = $lonTo - $lonFrom;
12.
13.   $angle = 2 * asin(sqrt(pow(sin($latDelt
    a / 2), 2) +
14.     cos($latFrom) * cos($latTo) * pow(sin
    ($lonDelta / 2), 2)));
15.   return $angle * $earthRadius;
16. }
```

## 2.10 Mobile Interface

The GPS Tracker application interface display on the smartphone used in sending GPS coordinate location data. The menu contains login based on username and password, as well as GPS Tracker activation button.

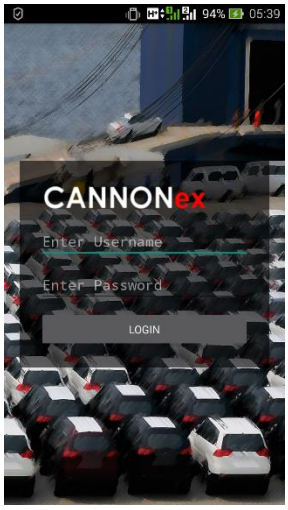



**Figure 13.** Interface Mobile (1)




**Figure 14.** Interface Mobile (2)

Location data sent in the form of coordinates latitude and longitude with time in each delivery. This mobile system can also display the order data received by the driver.

## 2.11 Web Interface

The interface of the website is a back-end program of vehicle delivery monitoring applications. Figure 15 represents all the status of the last position of the shipment.




**Figure 15.** Interface Web (1)

Figure 16 shows the website interface can display the path or route through which the driver.

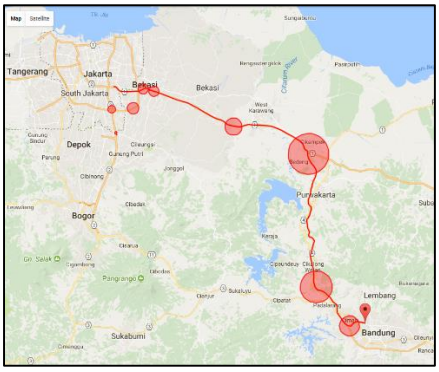



**Figure 16.** Interface Web (2)

.

## 2.12 Fungsional Testing

Testing is a part of software development that aims to ensure that the built software has proven criteria and can run well without any interruption or error. This test is performed especially on testing system functionality and system performance that has been built. Testing is done by black box testing techniques that focus more on finding functional fault program.

Based on the results of blackbox testing performed, it can be concluded that the system runs correctly and release information as expected. If any data does not match, then the system will show an error message.

## 2.13 Beta Testing

Beta testing is done by distributing questionnaires to operational officers. Table 1 is the result of the questionnaire.

**Table 1.** Beta Testing Result

| Pertanyaan 1 : Aplikasi sudah memberikan informasi posisi kendaraan? | | | | |
|---|---|---|---|---|
| SS | ST | RG | TS | STS |
| 1 | 2 | 0 | 0 | 0 |
| Rata-rata = (5+8+0+0+0) / 3 = 4.33 | | | | |
| Pertanyaan 2 : Aplikasi sudah membantu me-monitor rute yang dilewati menggunakan geofencing? | | | | |
| SS | ST | RG | TS | STS |
| 0 | 3 | 0 | 0 | 0 |
| Rata-rata = (0+12+0+0+0) / 3 = 4 | | | | |
| Pertanyaan 3 : Aplikasi mudah untuk digunakan? | | | | |
| SS | ST | RG | TS | STS |
| 0 | 3 | 0 | 0 | 0 |
| Rata-rata = (0+12+0+0+0) / 3 = 4 | | | | |
| Rata-rata Akhir = (4.33+4+4) / 3 = 4.11 | | | | |

From the above results can be concluded that about 82.2% of respondents agree the development of geofencing in monitoring delivery vehicles help and facilitate in managing logistics delivery.

## 3. CLOSING

The conclusions and suggestions obtained from the results of this study are:

### 3.1 Conclusions

Based on the results of research, analysis, design, implementation, and testing, then obtained the following conclusions:

a. Implementation of geofencing can help provide position information of a vehicle.
b. Implementation of geofencing can detect what route is passed when delivery of vehicle.

### 3.2 Suggestion

The implementation of geofencing in monitoring of vehicle delivery is still far from perfect. Therefore it is necessary to do further development and refinement. As for some suggestions that can be applied for further development is:

a. Utilization of geofencing in addition to determining the delivery route of vehicles, such as in terms of security or POI.
b. Merging other Google APIs like Road API or Direction API to improve the quality of the system.
c. The use of geofecing in determining the speed of vehicles in certain areas

## BIBLIOGAPHY


[1] F. Reclus and K. Drouard, "Geofencing for Fleet & Freight Management," Geofencing for Fleet & Freight Management, pp. 353-356, 2009.
[2] A. Nait-Sidi-Moh, "On the Use of Location-Based Services and Geofencing Concepts for Safety and Road Transport Efficiency," Trends in Mobile Web Information Systems, pp. 135-144, 2013.
[3] M. Corps, Design, monitoring, and evaluation guidebook, 2005.
[4] H. Jogiyanto, Analisis & Desain Sistem: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis, Yogyakarta: Andi, 2005.
[5] N. Safaat, Pemrograman Aplikasi Mobile Smarthphone dan Tablet PC Berbasis Android, Bandung: Informatika Bandung, 2015.
[6] A. D. Kasman, Trik Kolaborasi Android dengan PHP dan MySQL, Yogyakarta: CV. Lokomedia, 2015.
[7] R. T. Setiadi and W. Pujiyono, "Spektrum Industri," Sistem Pelacak Kendaraan Berbasis OpenGTS, vol. 11, no. 2, pp. 117-242, 2013.
[8] A. Küpper, U. Bareth and B. Freese, "Geofencing and Background Tracking – The Next Features," 2011.
[9] R. C. Martin, UML for JAVA Programers, New Jersey: Alan Apt, 2002.
[10] D. Namiot and M. Sneps-Sneppe, "Geofence and Network Proximity," Lecture Notes in Computer Science, vol. 8121, 2013.