

# RANCANG BANGUN APLIKASI INFORMASI NAMA STASIUN KERETA JURUSAN JAKARTA – CIREBON BERBASIS *SMARTPHONE* ANDROID

Rohadi, Husni Teja Sukmana, Rayi Pradono Iswara

<sup>1,2,3</sup>Program Studi Teknik Informatika, Fakultas Sains dan Teknologi  
UIN Syarif Hidayatullah Jakarta

<sup>1</sup>rohaditarsum@gmail.com, <sup>2</sup>husniteja@uinjkt.ac.id, <sup>3</sup>rayi.iswara@uinjkt.ac.id

## ABSTRAK

Diantara banyaknya jasa pelayanan masyarakat, adalah kendaraan umum seperti kereta api (KA) dan kereta rel listrik (KRL) yang banyak sekali digunakan sebagai kendaraan alternatif dalam melakukan perjalanan, karena harga perjalanan yang terjangkau juga waktu perjalanan yang relatif stabil karena memiliki jalur khusus, meski keterlambatan waktu kadang terjadi tapi tingkatnya lebih rendah dibanding dengan angkutan umum seperti bus, mikrolet, atau angkot. Sarana yang ditawarkan oleh kereta api (KA) dan kereta rel listrik (KRL) adalah waktu yang stabil dan harga yang terjangkau karena terdapat kelas-kelas untuk jenis KA dan KRL yaitu kelas Ekonomi, Bisnis, Eksekutif. Dikarenakan jalur khusus KA dan KRL ini lebih banyak mengambil jalur yang jauh dari kawasan penduduk (pertimbangan akibat resiko kecelakaan) atau jalan umum yang besar ditambah dengan kecepatan KA atau KRL yang konstan mengakibatkan penumpang sulit mencari informasi tentang daerah atau lokasi yang tengah dilewati dan yang akan dituju. Sehingga penumpang kebingungan dimana harus turun, jika adapun informasi di stasiun tidak memadai dan kurang layak untuk sebuah kenyamanan informasi. Ada beberapa aplikasi yang sudah ada seperti *Google Map*. Aplikasi tersebut memang dapat melihat lokasi peta secara real time, akan tetapi aplikasi ini tidak dapat memberikan notifikasi menggunakan suara. Disini kami akan menggabungkan notifikasi agar penumpang dapat mengetahui dimana posisi kereta walaupun *device* dalam keadaan *stand by*, dengan memanfaatkan teknologi GPS sebagai penentu koordinat dari setiap stasiun singgah yang telah di tentukan.

**Kata kunci:** *Transportasi, Kereta Api (KA), Kereta Rel Listrik (KRL), Android, Google Map..*

## I. PENDAHULUAN

### 1.1. Latar Belakang

Diantara banyaknya jasa pelayanan masyarakat, adalah kendaraan umum seperti kereta api (KA) dan kereta rel listrik (KRL) yang banyak sekali digunakan sebagai kendaraan alternatif dalam melakukan perjalanan, karena harga perjalanan yang terjangkau juga waktu perjalanan yang relatif stabil karena memiliki jalur khusus, meski keterlambatan waktu kadang terjadi tapi tingkatnya lebih rendah dibanding dengan angkutan umum seperti bus, mikrolet, atau angkot.

Sarana yang ditawarkan oleh kereta api (KA) dan kereta rel listrik (KRL) adalah waktu yang stabil dan harga yang terjangkau karena terdapat kelas-kelas untuk jenis KA dan KRL yaitu kelas Ekonomi, Bisnis, Eksekutif.

Berdasarkan hasil survey yang dilakukan penulis dengan mewawancarai lima belas mahasiswa dan warga wialyah tiga Cirebon yang tinggal di Jakarta yang melakukan perjalanan pulang atau pergi menggunakan transportasi kereta api, menyatakan 80% mengalami kesulitan mengetahui nama stasiun yang tengah disinggahi dan 86,66% menyatakan perlu adanya aplikasi bantu yang dapat memberikan informasi tentang nama-nama stasiun singgah.

Dari data hasil survey di atas, maka dapat di simpulkan terdapat sebuah permasalahan sebagai berikut. Dikarenakan jalur khusus KA dan KRL ini lebih banyak mengambil jalur yang jauh dari kawasan penduduk (pertimbangan akibat resiko kecelakaan) atau jalan umum yang besar ditambah dengan kecepatan KA atau KRL yang konstan mengakibatkan penumpang sulit mencari informasi tentang daerah atau lokasi yang tengah dilewatikan dan yang akan dituju. Sehingga penumpang kebingungan dimana harus turun, jika adapun informasi di stasiun tidak memadai dan kurang layak untuk sebuah kenyamanan informasi.

Ada beberapa aplikasi yang sudah ada seperti *Google Map*. Aplikasi tersebut memang dapat melihat lokasi peta secara real time, akan tetapi aplikasi ini tidak dapat memberikan notifikasi menggunakan suara. Disini kami akan menggabungkan notifikasi agar penumpang dapat mengetahui dimana posisi kereta walaupun *device* dalam keadaan *stand by*.

### 1.2. Rumusan Masalah

Berdasarkan latar belakang yang telah diuraikan di atas, maka diambil suatu rumusan masalah yaitu:

1. Apakah teknologi Android dan GPS dapat membantu pengguna kereta menentukan lokasi stasiun yang akan disinggahi?

2. Bagaimana hasil uji coba teknologi *Android* pada *smartphone* untuk menentukan lokasi stasiun singgah?

### 1.3. Rumusan Masalah

Karena terbatasnya waktu dan terbentur oleh biaya penelitian, demi memanfaatkan waktu yang sangat singkat dan hasil yang optimal. Maka penulis membatasi masalah yang akan dibahas adalah:

1. Aplikasi ini hanya berbasis *Android*.
2. Menggunakan koordinat dari *Google map*.
3. Aplikasi ini hanya untuk angkutan kereta api.
4. Trayek kereta api jurusan Jakarta – Cirebon.
5. Radius pemberitahuan dapat berjalan dalam jarak 1KM sebelumsampai pada lokasi yang dituju.
6. Minimum spesifikasi perangkat yang digunakan: *Jelly Bean 4.1*, *Dual Core 1 Ghz*, *Ram 512*, *Memory Internal 4 Gb*

### 1.4. Tujuan Penelitian

Tujuan dari penelitian skripsi yang dilakukan tersebut adalah untuk:

1. Merancang suatu aplikasi *Android* pemberitahuan nama-nama stasiun yang secara otomatis dapat berjalan dalam bentuk suara.
2. Menghasilkan aplikasi pemberitahuan yang dapat bermanfaat bagi masyarakat yang melakukan perjalanan menggunakan transportasi umum Kereta Api.
3. Memberikan pelayanan bagi para penumpang baik yang pertama kali atau yang sudah lamamenggunakan sarana transportasi kereta api

## II. LANDASAN TEORI

### 2.1. Pengertian Transportasi

Sistem transportasi berkembang sesuai dengan kebutuhan manusia pada masa lalu, ketika mobilitas penduduk masih sangat rendah dan luas mobilitasnya masih terbatas didalam lingkungan hidupnya manusia belum begitu membutuhkan sarana transportasi. Akan tetapi ketika tingkat mobilitas semakin tinggi dan jangkauannya semakin luas maka dibutuhkan sarana transportasi yang memadai dan menunjang mobilitasnya.

Perkembangan transportasi dalam sejarah bergerak dengan sangat perlahan, berevolusi dengan terjadi perubahan sedikit-demi sedikit, yang sebenarnya diawali dengan perjalan jarak jauh berjalan kaki pada jaman *paleolithic*. Sejarah manusia menunjukkan bahwa selain berjalan kaki juga dibantu dengan pemanfaatan hewan yang menyeret suatu muatan yang tidak bisa diangkat oleh manusia dan penggunaan rakit di sungai. Beberapa rekaman mengenai transportasi terekam dalam relief yang dipahat dibatu pada daerah Mesir Kuno dan daerah sekitarnya yang terekam dalam Relief yang ditemukan di Ibukota Assyrian *Dur Sharrukin*, 8 abad SM.

### 2.2. Smartphone

*Smartphone* atau telepon pintar dalam terjemahan bahasa Indonesia merupakan komputasi bergerak (*mobile*) dalam konvergensi antara komunikasi, computer, dan pengguna perangkat elektronik yang menjadi terminal bergerak terpadu dengan membawa fungsi mobilitas dan akses jaringan di semua tempat.

Menurut *Pei Zheng* dan *Lionel Ni* mendefinisikan *smartphone* sebagai sebuah kelas baru pada teknologi telepon seluler yang mampu memfasilitasi akses data dan pemrosesan informasi dengan kemampuan komputasi secara signifikan. Selain memiliki fungsi tradisional yang terdapat pada telepon seluler seperti menelpon dan berkirim pesan (*short messages Services*), *smartphone* dilengkapi dengan manajemen informasi personal (*PIM*) dan komunikasi ke beberapa media akses nirkabel.

### 2.3. Android

*Android* adalah sebuah *platform* pertama yang betul-betul terbuka dan komprehensif untuk perangkat *mobile*, semua perangkat lunak yang ada difungsikan untuk menjalankan sebuah *mobile device* tanpa memikirkan kendala kepemilikan yang menghambat inovasi pada teknologi *mobile* (*Meier*, 2008). Dalam definisi lain, *android* merupakan subset perangkat lunak untuk perangkat *mobile* berbasis *linux* yang meliputi sistem operasi, *middleware*, dan aplikasi inti yang dirilis oleh *Google* (*Nazaruddin*, 2010). Sedangkan *Android SDK (Software Development Kit)* menyediakan *Tools* dan *API* yang diperlukan untuk mengembangkan aplikasi pada *platform* *Android* dengan menggunakan bahasa pemrograman *Java*.

### 2.4. Global Positioning System (GPS)

*Global Positioning System* adalah sistem untuk menentukan letak di permukaan bumi dengan bantuan penyalarsan (*synchronization*) sinyal satelit. Sistem ini menggunakan 24 satelit yang mengirimkan sinyal gelombang mikro ke Bumi. Sinyal ini diterima oleh alat penerima di permukaan, dan digunakan untuk menentukan letak, kecepatan, arah, dan waktu.



Gambar 1 Global Positioning System (GPS)

### III. METODOLOGI PENELITIAN

Metode Pengumpulan Data yang peneliti gunakan adalah wawancara, kuesioner, studi pustaka, dan studi sejenis. Sedangkan metode pengembangan sistem yang peneliti gunakan dalam penelitian ini adalah Rapid Application Development (RAD), yang memiliki tahapan-tahapan berikut (Kendall, 2010) : Fase Perencanaan Syarat-syarat, Fase *Workshop Design*, Fase Implementasi.

#### 3.1. Fase perencanaan Syarat-syarat

Dalam fase ini, pengguna dan penganalisis bertemu untuk mengidentifikasi tujuan-tujuan aplikasi atau sistem serta untuk mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut. Fase ini memerlukan peran aktif mendalam dari kedua kelompok tersebut. Selain itu juga melibatkan pengguna dari beberapa level yang berbeda dalam organisasi. Orientasi dalam fase ini ialah menyelesaikan masalah-masalah perusahaan. Meskipun teknologi informasi dan sistem bisa mengarahkan sebagian dari sistem yang diajukan, fokusnya akan selalu tetap pada upaya pencapaian tujuan-tujuan perusahaan.

#### 3.2. Fase Workshop Design

Fase ini adalah fase untuk merancang dan memperbaiki yang dapat digambarkan sebagai *workshop*. Selama *workshop design* RAD, pengguna merespons *working prototype* yang ada dan penganalisis memperbaiki modul-modul yang dirancang (menggunakan perangkat lunak) berdasarkan respon pengguna.

#### III.3. Fase Implementasi

Penganalisis bekerja dengan para pengguna secara intens selama *workshop* untuk merancang aspek-aspek bisnis dan non-teknis dari perusahaan. Segera sesudah aspek-aspek ini disetujui dan sistem-sistem dibangun dan disaring, sistem-sistem baru atau bagian dari sistem diuji coba dan diperkenalkan kepada *stakeholder*

### IV. ANALISA DAN PERANCANGAN

#### 4.1. Fase Perencanaan Syarat-syarat

Dalam fase ini langkah-langkah yang dilakukan adalah sebagai berikut:

- Gambaran umum dari trayek kereta api, yang bertujuan mengetahui data-data dari stasiun-stasiun yang disinggahi dan dilewati oleh kereta api teyak Jakarta – Cirebon yang akan menjadi patokan dari aplikasi yang akan dibuat;
- Mengidentifikasi fitur-fitur berdasarkan tujuan-tujuan aplikasi atau sistem serta mengidentifikasi syarat-syarat informasi yang ditimbulkan dari tujuan-tujuan tersebut;

#### 4.2. Fase Workshop Design

Setelah disusun sistem yang ada termasuk penyelesaian kendala-kendala atau permasalahan-permasalahan yang ada, tahap selanjutnya adalah mendesain aplikasi yang diusulkan agar dapat berjalan lebih baik dan diharapkan dapat mengatasi masalah-masalah yang ada.

Dalam fase ini peneliti membuat desain model aplikasi:

- Perancangan spesifikasi proses yang dibutuhkan, dengan menerjemahkan proses-proses yang terjadi di dalam sistem ini kedalam bentuk algoritma sederhana yang akan di implementasikan dalam bentuk program;
- Perancangan *interface*, dengan membuat rancangan layar tampilan yang berupa input-input yang bertujuan untuk memfasilitasi komunikasi antara pengguna dengan sistem. Setelah rancangan layar tampilan terbentuk maka dilakukan tahap implementasi;

#### 4.3 Perancangan User Interface

Pada tahapan ini, peneliti merancang tampilan Antar Muka dari aplikasi.

##### a. Tampilan Menu Utama



Gambar 2 Rancangan Menu Utama

Tampilan menu utama terdiri dari empat tombol menu utama yaitu Profil, You Here, Map, Me dan satu tombol Exit untuk keluar dari aplikasi

##### b. Tampilan Menu Profil



Gambar 3 Rancangan menu Profil

Tampilan pada menu Profil ini menampilkan informasi tentang sejarah dari PT. KAI yang di ambil langsung dari situs resminya. Tujuannya memberikan informasi tentang sejarah dari moda transportasi kereta yang ada di Indonesia.

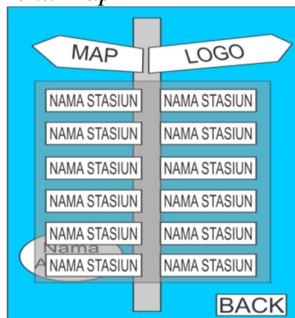
c. Tampilan Menu You Here



Gambar 4 Rancangan menu You Here

Tampilan pada menu You Here adalah koordinat dari stasiun terdekat dari posisi perangkat yang user gunakan dan posisi user itu sendiri serta jarak antara user dan stasiun terdekat.

d. Tampilan Menu Map



Gambar 5 Rancangan Menu Map

Tampilan pada menu Map adalah dua belas list nama stasiun yang ada dalam aplikasi, yang ketika dipilih akan menampilkan posisi stasiun tersebut pada google map.

e. Tampilan Menu Me

Pada tampilan menu Me adalah profil pribadi dari pembuat aplikasi.



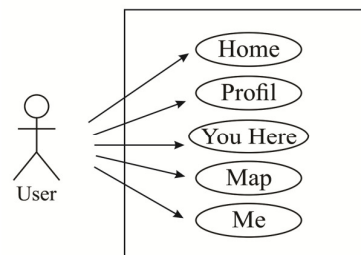
Gambar 6 Rancangan Menu Me

4.4. Perancangan Use Case Diagram

Use Case mendeskripsikan interaksi actor di dalam aplikasi yang di kembangkan. Dalam konteks ini peneliti memilih pengguna Android (*user*) sebagai actor.

Tabel 1 Identifikasi Actor dan Use Case

No	Actor	Description
1	User	Orang yang mengakses data dan mendapat <i>notification</i> nama stasiun kereta yang disinggahi melalui aplikasi <i>native</i> yang terinstal pada <i>smartphone</i> Android

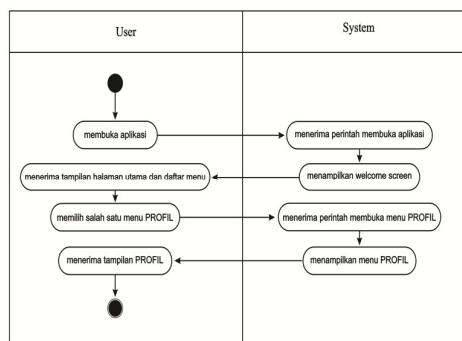


Gambar 7 Use Case Diagram

Dalam *use case diagram* diatas, *user* sebagai actor yang mempunyai *use case* pilih Profil, *You Here*, *Map*, *Me*.

4.5. Perancangan Activity Diagram

*Activity Diagram* menggambarkan katifitas-aktifitas yang terjadi dalam aplikasi ‘Stasiun Manakah?’ dari aktifitas dimulai sampai aktifitas berhenti.



Gambar 8 Activity Diagram Pilih Profil

Di dalam *Activity Diagram Use Case* pilih profil, aktivitas pertama dimulai dari *user* dengan membuka aplikasi ‘Stasiun Manakah?’, lalu sistem merespon dan menampilkan menu Home yang berisi beberapa pilihan menu lain. Kemudian *user* memilih menu Profil, yang kemudian di respon oleh sistem dengan menampilkan profil KAI.

#### 4.6. Fase Implementasi

Implementasi merupakan tahapan presentasi dari hasil perancangan ke dalam program. Peneliti menggunakan Adobe Flash CS6 sebagai bahasa pemrograman yang di dalamnya sudah terdapat Android SDK (*Software Development Kit*). Dilanjutkan dengan pemasangan (*install*) ke *smartphone* Android serta melakukan pengujian atau testing aplikasi secara *blackbox* yaitu dengan mengetahui fungsi yang ditentukan dimana produk dirancang untuk melakukan sesuatu, pengujiannya dapat dilakukan untuk memperlihatkan bahwa masing-masing fungsi beroperasi sepenuhnya, pada waktu yang sama mencari kesalahan pada setiap fungsi (Pressman, 2002).

Setelah pengujian *blackbox* di lanjutkan tahap uji coba aplikasi dilapangan, tujuannya untuk mengukur akurasi dari koordinat yang sudah didapat dari google map dan untuk mengetahui apakah aplikasi yang sudah terinstal dapat berjalan sesuai dengan keinginan atau tidak.

#### 4.7. Pengujian Black Box

Pengujian ini dilakukan dengan menguji validasi terhadap aplikasi ‘Stasiun Manakah?’. Pengujian ini dilakukan dengan cara pengujian mandiri dengan menyebarkan beberapa kuesioner kepada masyarakat luar daerah Jakarta yang melakukan perjalanan pulang atau dari daerah berangkat ke Jakarta yang menggunakan *smartphone* Anroid.

Tabel 2 Pengujian Menu Home

Kasus dan Hasil Uji	
Data Masukan	User membuka aplikasi Stasiun Manakah? Yang telah terinstal pada <i>smartphone</i> Android.
Yang Diharapkan	Dapat menampilkan <i>home screen</i> yang berupa tampilan <i>button</i> menu-menu pilihan.
Pengamatan	Sistem dapat menampilkan empat kategori <i>button</i> menu.
Kesimpulan	Berhasil

#### 4.8. Pengujian Akurasi Koordinat

Pengujian ini dilakukan untuk mengetahui seberapa akurat aplikasi ‘Stasiun Manakah?’ dapat membaca koordinat dari google map yang sudah dimasukkan dalam sistem pengkodean. Tujuannya agar mengetahui apakah koordinat yang telah di ambil dari google map sinkron dengan data yang ada di lapangan atau tidak.

Tabel 3 Hasil Pengujian 1

Jakarta-Cirebon	Latitude	Longitude	Latitude	Longitude	Aplikasi berjalan (meter)
Jakarta (Kota)	-6.137293	106.814668	-6.133887821	106.8220628	906.2654539
Bekasi	-6.23591	106.9985	-6.240912346	107.0053291	942.338126
Cikarang	-6.270584	107.179991	-6.270398469	107.1878168	871.4088949
Karawang	-6.305876	107.300464	-6.30932885	107.3053704	664.871358
Cikampek	-6.406731	107.458768	-6.408540028	107.4665874	893.4437432
Pagaden	-6.453742	107.817061	-6.459652108	107.8198961	729.6889263
Haurgeulis	-6.458572	107.940884	-6.456402167	107.9490933	945.2324248
Trisi	-6.459977	108.161011	-	-	-
Kedokan Gabus	-6.467752	108.106837	-6.468823514	108.1153864	959.1545903
Jatibarang	-6.473021	108.306284	-6.479839423	108.308502	798.1695414
Arjawinangun	-6.644682	108.41451	-6.646899285	108.4217427	842.140331
Cirebon	-6.705338	108.555296	-6.709988995	108.5575752	576.5711302

Cirebon-Jakarta	Latitude	Longitude	Latitude	Longitude	Aplikasi berjalan (meter)
Jakarta (Kota)	-6.137293	106.814668	-6.133887801	106.8220328	898.958182
Bekasi	-6.23591	106.9985	-6.240922356	107.0054291	948.2806596
Cikarang	-6.270584	107.179991	-6.271398469	107.1878388	873.0911201
Karawang	-6.305876	107.300464	-6.30832885	107.3043704	511.2429247
Cikampek	-6.406731	107.458768	-6.408541028	107.4675874	996.2105862
Pagaden	-6.453742	107.817061	-6.453652108	107.8188961	203.2083427
Haurgeulis	-6.458572	107.940884	-6.456302167	107.9470933	731.8176078
Trisi	-6.459977	108.161011	-	-	-
Kedokan Gabus	-6.467752	108.106837	-6.468323514	108.1154864	958.8000035
Jatibarang	-6.473021	108.306284	-6.474639424	108.3075021	224.9629632
Arjawinangun	-6.644682	108.41451	-6.647899285	108.4227417	978.0943693
Cirebon	-6.705338	108.555296	-6.709918995	108.5570752	546.5661307

Tabel 4 Hasil Pengujian 2

Cirebon-Jakarta	Latitude	Longitude	Latitude	Longitude	Aplikasi berjalan (meter)
Jakarta (Kota)	-6.137293	106.814668	-6.132887801	106.8210328	861.6751666
Bekasi	-6.23591	106.9985	-6.241922356	107.004429	939.9811087
Cikarang	-6.270584	107.179991	-6.271998469	107.1878689	890.9819783
Karawang	-6.305876	107.300464	-6.30962885	107.3059705	741.7972464
Cikampek	-6.406731	107.458768	-6.408431023	107.4665679	888.66563
Pagaden	-6.453742	107.817061	-6.453952101	107.8088905	900.8340103
Haurgeulis	-6.458572	107.940884	-6.456702102	107.9480932	829.073403
Trisi	-6.459977	108.161011	-	-	-
Kedokan Gabus	-6.467752	108.106837	-6.466320502	108.1151626	940.4026725
Jatibarang	-6.473021	108.306284	-6.477739467	108.307503	542.4997831
Arjawinangun	-6.644682	108.41451	-6.645899266	108.4217349	815.6061515
Cirebon	-6.705338	108.555296	-6.709938964	108.5589765	655.8849178

Jakarta-Cirebon	Latitude	Longitude	Latitude	Longitude	Aplikasi berjalan (meter)
Jakarta (Kota)	-6.137293	106.814668	-6.133882355	106.8221628	916.6405426
Bekasi	-6.23591	106.9985	-6.240935649	107.0053356	944.4630931
Cikarang	-6.270584	107.179991	-6.270312356	107.1878456	874.8615279
Karawang	-6.305876	107.300464	-6.309356487	107.3053316	666.1209194
Cikampek	-6.406731	107.458768	-6.408511565	107.4665356	887.1138282
Pagaden	-6.453742	107.817061	-6.459123565	107.819636	664.1150053
Haurgeulis	-6.458572	107.940884	-6.456531656	107.9492355	957.018545
Trisi	-6.459977	108.161011	-	-	-
Kedokan Gabus	-6.467752	108.106837	-6.468123565	108.1151236	923.3789814
Jatibarang	-6.473021	108.306284	-6.479235649	108.3085612	736.7909826
Arjawinangun	-6.644682	108.41451	-6.646235649	108.4211236	756.2527277
Cirebon	-6.705338	108.555296	-6.70926549	108.5573564	493.7160122

Tabel 5 Hasil Pengujian 3

Cirebon-Jakarta	Latitude	Longitude	Latitude	Longitude	Aplikasi berjalan (meter)
Jakarta (Kota)	-6.137293	106.814668	-6.133887821	106.8220364	903.5992602
Bekasi	-6.23591	106.9985	-6.240822014	107.0056461	965.3041431
Cikarang	-6.270584	107.179991	-6.271398165	107.1878123	875.3589755
Karawang	-6.305876	107.300464	-6.309328164	107.3053135	662.6497179
Cikampek	-6.406731	107.458768	-6.408541135	107.4675155	994.3895914
Pagaden	-6.453742	107.817061	-6.457659916	107.8198131	532.985655
Haurgeulis	-6.458572	107.940884	-6.456312035	107.9471935	746.0684194
Trisi	-6.459977	108.161011	-	-	-
Kedokan Gabus	-6.467752	108.106837	-6.478246898	108.1153329	1503.108059
Jatibarang	-6.473021	108.306284	-6.47563466	108.3095014	461.4384077
Arjawinangun	-6.644682	108.41451	-6.647599156	108.4217123	865.0255898
Cirebon	-6.705338	108.555296	-6.709998946	108.5551456	519.1218653

Jakarta-Cirebon	Latitude	Longitude	Latitude	Longitude	Aplikasi berjalan (meter)
Jakarta (Kota)	-6.137293	106.814668	-6.133888957	106.8220056	900.4365531
Bekasi	-6.23591	106.9985	-6.240813565	107.0033123	764.8160402
Cikarang	-6.270584	107.179991	-6.270345469	107.1878457	874.7758717
Karawang	-6.305876	107.300464	-6.30985649	107.3063456	790.5818556
Cikampek	-6.406731	107.458768	-6.408546589	107.4665356	887.9930701
Pagaden	-6.453742	107.817061	-6.459665894	107.8198066	731.0665451
Haurgeulis	-6.458572	107.940884	-6.456455649	107.9490236	936.214942
Trisi	-6.459977	108.161011	-	-	-
Kedokan Gabus	-6.467752	108.106837	-6.468865424	108.1143365	843.9890395
Jatibarang	-6.473021	108.306284	-6.479825649	108.3095659	840.9884737
Arjawinangun	-6.644682	108.41451	-6.646823565	108.4207565	735.0838089
Cirebon	-6.705338	108.555296	-6.709856892	108.5592654	669.5519355

## V. KESIMPULAN DAN SARAN

### 5.1. Kesimpulan

Dari pembahasan yang sudah diuraikan, dapat disimpulkan bahwa:

1. Aplikasi Android yang terkoneksi dengan GPS dapat membantu pengguna kereta mengetahui nama stasiun singgah yang dilewati.
2. Akurasi dari GPS *smartphone* Android dapat membaca koordinat yang telah di tentukan secara *realtime* tergantung dari spesifikasi *smartphone* yang digunakan.

### 5.2. Saran

Untuk pengembangan lebih lanjut, ada beberapa hal yang dapat peneliti sarankan untuk melengkapi aplikasi ini menjadi lebih baik lagi, antara lain:

1. Penambahan jadwal pemberangkatan dan kedatangan kereta api secara *real time* sehingga penumpang yang hendak melakukan perjalanan dapat memantau ketera api yang hendak ditumpanginya melalui *smartphone* yang dimiliki tanpa harus bertanya pada penjaga.
2. Penambahan jenis informasi suara yang lebih lengkap sehingga dapat memperjelas dan lebih menarik bagi *user*.
3. Penambahan database informasi nama stasiun singgah yang dapat mengcover sampai seluruh pulau jawa.

## DAFTAR PUSTAKA

- [1] A.A. Waskito (2009). *Kamus Praktis Bahasa Indonesia*. Jakarta : Wahyu Media.
- [2] Arikunto, Suharsimi (2010). *Prosedur Penelitian Suatu Pendekatan Praktek*. Jakarta: Rineka Cipta.
- [3] Anonymous. "Learning animal". From <http://www.learnenglish.de/basics/animals.html>, 30 Oktober 2013.
- [4] Anonymous. "Action Script 3.0". From <http://dasaranimasikita.blogspot.com/>, 5 November 2013. <http://stackoverflow.com>
- [5] Chonoles, M.J, James A. Schardt. (2003). *UML 2 for Dummies*. New York : Willey Publishing, Inc.
- [6] Di. H. Martinis Yamin, M.Pd, Dr. Jamilah Sabri, M.Pd. I (2010). *Panduan Pendidikan Anank Usia Dini*. Jakarta : Gaung Persada
- [7] Kendall, Kenneth E dan Kendall, Julie E. (2008). *Analisis dan Perancangan Sistem, Edisi Kelima (Versi Indonesia)*. Jakarta :Indeks.
- [8] Mulyadi, Adi. (2010). *Membangun Aplikasi Android*. Yogyakarta : Multimedia Center Publishing.
- [9] Nazir (1998). *Metode Penelitian*, Jakarta : Rineka Cipta
- [10] Nugroho, Adi (2005). *Rational Rose untuk Pemodelan Berorientasi Objek*. Bandung : Informatika.
- [11] Pressman, Roger S (2002). *Rekayasa Perangkat Lunak*. Yogyakarta : Penerbit Andi.
- [12] Safaat, Nazruddin (2011). *Pemrograman Aplikasi Mobile Smartphone dan Tablet PC Berbasis Android*. Bandung: Informatika.
- [13] Septiawan ebta."KBBi Online". from <http://kbbi.web.id/>, 15 agustus 2013.
- [14] Sugiyono (2010). *Metode Penelitian Kuantitatif Kualitatif dan R&D*. Bandung : Alfabeta.
- [15] Zaki, Ali. (2008). *E-Life Style Memanfaatkan Beragam Perangkat Teknologi Digital*. Jakarta : Salemba Infotek.