

APLIKASI PENCARIAN BARBERSHOP BERBASIS ANDROID

Oleh :

Ganda Yoga Swara¹, Afif Zirwan²

Teknik Informatika

Fakultas Teknologi Industri

Institut Teknologi Padang

gandayogaswara@gmail.com, afifzirwan47@gmail.com**Abstrak**

Permasalahan yang ada di *barbershop* adalah lamanya antrian oleh pelanggan dan rasa bosan menunggu di lokasi barber. Maka muncul solusi yaitu menyediakan jasa kapster secara *online*. Aplikasi ini dibangun berbasis Android dan dilengkapi dengan fitur *tracking* untuk memantau lokasi kapster. Fitur *Tracking* dibangun menggunakan *geolocation* dan *jQuery Latest*. Dari hasil pengujian didapatkan hasil penggunaan fitur *tracking* pada ruangan terbuka dengan kondisi dibawah pepohonan mendapatkan *rank* tertinggi dibandingkan lokasi uji lainnya, dengan persentase data berhasil sebanyak 77,5% , data gagal 12,5% dan rata-rata *delay* antar masuk data berkisar 0,45 detik per 40 data masuk. Penggunaan *jQuery Latest* menimbulkan munculnya *error* data atau data bertipe NaN (*Not A Number*) atau data bukan angka, karena kondisi dimana *geolocation* setelah di *refresh* *jQuery Latest* belum memberikan koordinat dan konversi ke bentuk angka. Data NaN berdampak ke *interface* sistem *tracking* yang menyebabkan *skip* atau *marker* akan hilang beberapa saat sampai menemukan koordinat pada *database* yang diakses. Untuk penelitian selanjutnya dapat dilakukan uji coba pembangunan sistem *tracking* tanpa menggunakan *jQuery Latest* dengan harapan dapat meminimalisir adanya *error* data dan dapat dilakukan pengujian antara sistem *tracking* tanpa menggunakan *jQuery Latest* dengan sistem *tracking* yang menggunakan *jQuery Latest*.

Kata-kata kunci : *Android, Barbershop, Kapster, Tracking, Geolocation, jQuery Latest*

Abstract

Problems that exist in barbershop is the queue of customers and bored waiting in the barber. From this background came solution to providing online capster services. This app is built on Android with tracking system for monitor the capster. Tracking system built using geolocation and jQuery Latest. From the test results obtained the results of the use of tracking in open space under the tress get highest rankings compare other data locations, with the percentage of successful data as much as 77.5%, 12.5% data failed (NaN) and the average delay between entry data 0,45 seconds. NaN data or data does not exist, because the geolocation condition after refresh with jQuery Latest has not provided coordinates and conversions to float data type. The previous data skip or marker will disappear for a while until it finds the coordinates in the database that is accessed. For further research can be tested using a tracking system without using the latest jQuery in hopes to minimize the error data and can be tested between tracking system without using jQuery Latest with tracking system using jQuery Latest.

Keyword : *Android, Barbershop, Kapster, Tracking, Geolocation, jQuery Latest*

1. PENDAHULUAN

Pada saat sekarang tren *startup* yang sedang *booming* adalah layanan jasa yang dilakukan dalam bentuk *online*, mulai dari jasa antar penumpang, barang, makanan, bahkan hingga jasa pembelian obat-obatan. Dilihat dari tren *startup* hari ini, dapat melakukan hipotesa bahwa *user* yang menggunakan aplikasi memiliki sifat yang

konsumtif dan tidak ingin ribet dan ingin simpel dalam melakukan pekerjaan. Dengan hipotesa diatas salah satu aplikasi yang dapat dijadikan salah satu *startup* baru adalah jasa penyedia pemotongan rambut secara *online*.

Menurut survei terhadap beberapa pemilik *barbershop* ternama di Kota Bukittinggi, rata-rata pengunjung *barbershop* berkisar antara 35 orang dalam 1 hari dengan

durasi pelayanan berkisar 40 menit dengan 4 orang kapster yang bekerja di *barbershop*-nya menimbulkan suatu masalah yakni pelanggan harus menunggu untuk dapat dilayani, dan tidak jarang mereka pengunjung mengalami kebosanan dan memilih meninggalkan serta mencari *barbershop* lain.

Melihat dari fakta-fakta di atas, maka munculah sebuah solusi, yakni dengan membangun aplikasi yang menyediakan jasa kapster secara *online*. Aplikasi ini merupakan aplikasi jasa *online* dengan mengumpulkan sumberdaya manusia yang memiliki kemampuan dibidang pemotongan rambut kedalam satu *platform* aplikasi. Dari sisi *business plan*, aplikasi ini memiliki 3 kelompok pengguna yakni, *admin* sebagai monitoring data dan melakukan pengolahan data mulai dari data *user* hingga data transaksi, kapster sebagai tokoh utama dalam aplikasi ini yang akan memberikan jasa potong rambut kepada *user* yang akan memesannya dan yang terakhir adalah *user* yang akan melakukan transaksi didalam aplikasi ini. Pada perancangannya aplikasi ini akan dilengkapi dengan fitur unggulan yakni *tracking system*. Dengan adanya fitur ini *user* dapat melacak dimana posisi kapster yang telah di pesan didalam aplikasi. Fitur *tracking* dibangun dengan menggunakan *geolocation* sebagai pengambilan koordinat dan *jQuery Latest* untuk melakukan *refresh* koordinat.

2. METODOLOGI

Penelitian yang berkaitan dengan *tracking system* sudah banyak dilakukan, seperti I Made Oka Widyantara (2015) pada jurnal "Penerapan Teknologi GPS Tracker Untuk Identifikasi Kondisi Trafik Jalan Raya". Pada jurnal ini dibahas bagaimana membangun sebuah *tracking system* dengan menggunakan perangkat GPS Tracker Tipe TR06 dengan berbantu *software* OpenGTS untuk melakukan *query* permintaan titik koordinat. Penelitian lainnya terkait *tracking system* juga hampir sama yakni, metode yang digunakan untuk membangun sebuah *tracking* adalah dengan menggunakan suatu perangkat *tracker* dan sebuah *software* bantuan. Oleh karena itu penulis ingin melakukan *experimental* terhadap penggunaan *Geolocation* dan *jQuery Latest*

untuk menggantikan perangkat GPS Tracker dan *software* OpenGTS diatas. Penulis memilih *Geolocation* karena memiliki fungsi yang sama dengan perangkat GPS Tracker yakni memberikan koordinat, sedangkan *jQuery Latest* memiliki fungsi yang secara garis besar sama dengan *software* OpenGTS yakni melakukan *query looping* untuk melakukan *update* koordinat *realtime*.

A. Location Based Service(LBS)

Location Based Service atau Layanan Berbasis Lokasi adalah salah satu bidang yang sedang *booming* untuk sekarang, baik untuk aplikasi *handphone*, *desktop*, atau *web*. LBS menentukan lokasi pengguna dengan beberapa teknologi. Posisi yang ditemukan dikombinasikan dengan informasi lain untuk menyediakan layanan aplikasi atau jasa.

LBS mampu menjawab tiga pertanyaan, yaitu:

1. Dimana saya sekarang?
2. Apa yang ada di sekitar saya?
3. Bagaimana saya pergi ke sana?

B. Google Map API

Google Maps adalah sebuah jasa peta globe virtual secara gratis dan *online* disediakan oleh Google, dapat ditemukan di <http://maps.google.com/>. Google Map menawarkan peta yang dapat diseret dan gambar satelit untuk seluruh dunia dan baru-baru ini, dan juga menawarkan perencanaan rute dan pencari letak bisnis di U.S., Kanada, Jepang, Hong Kong, Cina, UK, Irlandia (hanya pusat kota) dan beberapa bagian Eropa. Google Map API merupakan aplikasi *interface* yang dapat diakses lewat JavaScript agar Google Map dapat ditampilkan pada halaman *web* yang sedang dibangun. Untuk dapat mengakses Google Map, harus dilakukan pendaftaran API Key terlebih dahulu pada *website* Google Developer, yang nanti akan disematkan pada *link* pemanggilan API Google Map.

C. jQuery

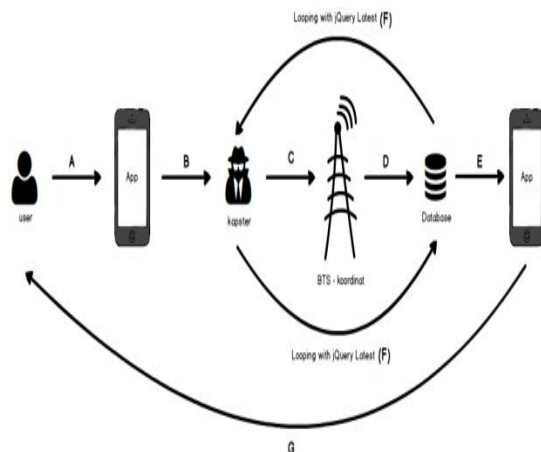
jQuery adalah pustaka JavaScript kecil bersumber terbuka yang menekankan pada interaksi antara JavaScript dan HTML. Pustaka ini dirilis pada Januari 2006 di BarCamp NYC oleh John Resig dan berlisensi ganda di bawah Lisensi MIT dan

GPL (Utama, 2016). jQuery merupakan JavaScript *library*, jQuery mempunyai semboyan “*write less, do more*”. jQuery dirancang untuk memperingkas kode-kode JavaScript. jQuery adalah JavaScript *library* yang cepat dan ringan untuk menangani dokumen HTML, menangani *event*, membuat animasi dan interaksi *ajax*. jQuery dirancang untuk mengubah cara menulis JavaScript (Satria, 2016). *Library* jQuery mempunyai kemampuan:

1. Kemudahan mengakses elemen-elemen HTML
2. Memanipulasi elemen HTML
3. Memanipulasi CSS
4. Penanganan event HTML
5. Efek-efek JavaScript dan animasi
6. Modifikasi HTML DOM
7. AJAX
8. Menyederhanakan kode JavaScript lainnya.

D. jQuery Latest

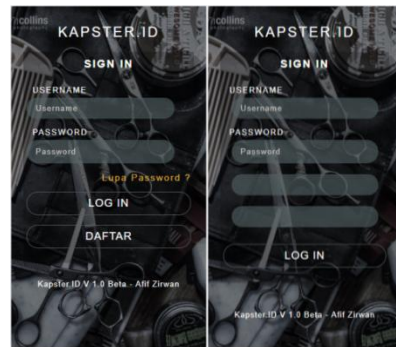
Merupakan salah satu fitur yang diciptakan oleh jQuery. Pada fitur ini jQuery latest digunakan untuk melakukan *refresh* beberapa *div* pada *tag* HTML. Jadi pada konsepnya bisa dikatakan fitur tambahan untuk melakukan *refresh* secara otomatis pada *tag* tertentu tanpa *refresh* halaman HTML. Pada rancangan implementasi jQuery Latest yang akan penulis bangun dapat dilihat pada diagram berikut.



Gambar 1. Diagram Alir Penggunaan jQuery Lates

3. PEMBAHASAN

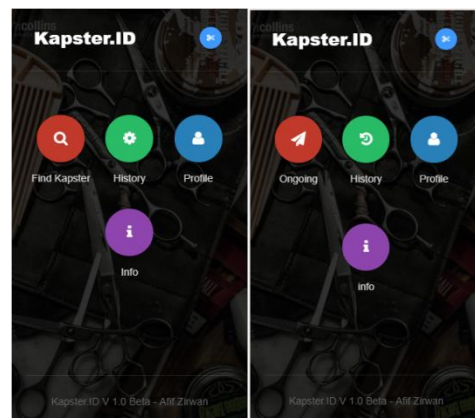
3.1 Halaman Login



Gambar 2. Halaman LoginUser (a) dan Halaman Login Kapster (b)

Pada halaman *login user* (Gambar 2 a), terdapat 3 fungsi utama yakni, *login* atau masuk ke aplikasi, *daftar* untuk melakukan pendaftaran akun bagi *user* yang belum memiliki akun, dan *lupa password* bagi *user* yang mengalami lupa *password*. Sedangkan pada halaman *login kapster* (Gambar 2 b), hanya terdapat satu fitur saja yakni fitur *login*, karena untuk pendaftaran dan lupa *password* dilakukan pada aplikasi *admin*. Pada *login kapster* disediakan dua *textbox* tambahan sebagai tempat untuk menampung nilai dari *latitude* dan *longitude* yang berfungsi untuk mengetahui dimana posisi kapster yang sudah *login* agar dapat tampil pada halaman *find kapster* pada aplikasi *user*

3.2 Halaman Menu Utama

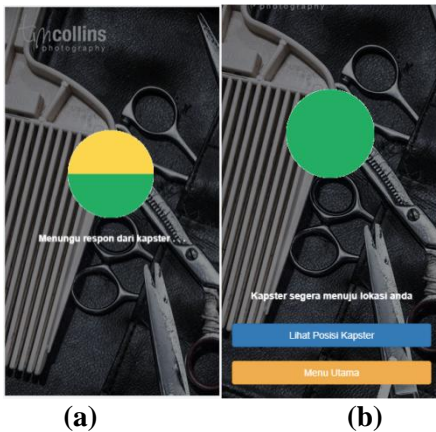


Gambar 3. Halaman Menu Utama User (a) dan Kapster (b)

Pada halaman menu utama *user* (Gambar 3 a), terdapat empat menu utama yakni, *find* kapster, *history*, *profile* dan info. Menu *find* kapster berguna untuk melakukan transaksi, pada menu ini akan ditampilkan sebuah peta dan *marker* yang akan menjadi tanda dari kapster yang sedang *online* dan sedang tidak melakukan pekerjaan atau *standby mode*. Menu *history* terdapat dua sub menu yakni sub menu *ongoing* atau transaksi yang sedang berlangsung dan sub menu *history* transaksi menu ini berfungsi melihat rekam transaksi yang pernah dilakukan ataupun sedang dilakukan oleh *user*. Menu *profile* berfungsi untuk mengatur informasi akun dari *user* mulai dari mengganti nama hingga mengganti *password*. Sedangkan pada menu info digunakan untuk menampilkan nama dan versi aplikasi.

Pada halaman menu utama kapster, juga terdapat empat menu utama yakni, *ongoing* untuk mengecek transaksi yang sedang berlangsung serta menampilkan lokasi pemesan, *history* untuk melihat transaksi yang sudah selesai, *profile* untuk melihat bonus dan pendapatan yang sudah diterima oleh kapster, serta info untuk menampilkan nama dan versi aplikasi.

3.3 Halaman *Waiting* Kapster



Gambar 4. Halaman *Waiting* Kapster

Setelah *user* melakukan pemesanan kapster, maka *user* akan diarahkan ke halaman *waiting* kapster. Halaman ini berfungsi untuk menunggu respon dari kapster yang dipesan. Jika kapster menerima pekerjaan yang ditujukan kepadanya maka halaman akan berubah menjadi seperti Gambar 4. b

diatas, dan *user* dapat melakukan *tracking* lokasi kapster yang di pesan.

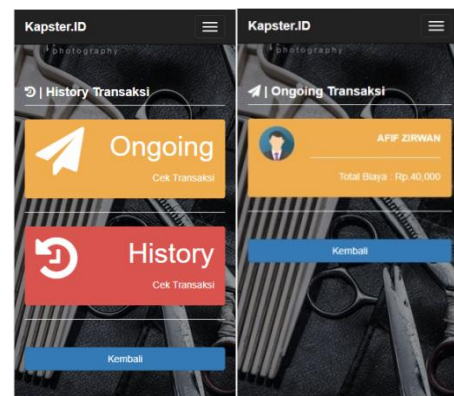
3.4 Halaman *Tracking* Kapster



Gambar 5. Halaman *Tracking* Kapster

Pada halaman ini *user* akan ditampilkan sebuah map untuk mengetahui dimana keberadaan kapster yang akan di pesannya. Pada halaman ini *marker* akan berpindah sesuai dengan koordinat dari kapster yang akan terus di-*update*.

3.5 Halaman *History*

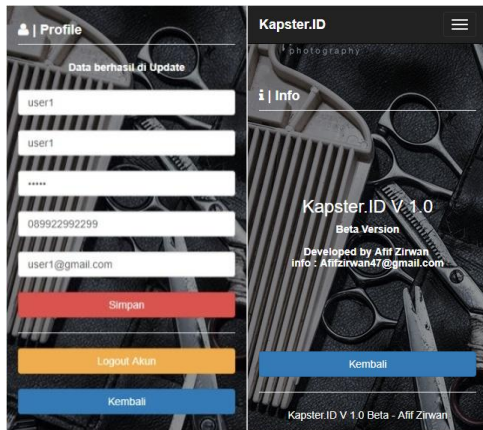


Gambar 6. Halaman *History*

Pada halaman ini ditampilkan dua pilihan menu yakni *ongoing* dan menu *history*. Secara tampilan sama, yang membedakan hanya pemanggilan data yakni *ongoing* adalah transaksi yang berstatus 1 atau 2 sedangkan *history* adalah transaksi berstatus 3 pada *database*. Pengkodean status ini digunakan untuk memudahkan dalam melakukan pengelompokan data didalam *database*. Kode 1 digunakan untuk transaksi sudah diterima oleh kapster yang dipesan, 2

digunakan untuk transaksi yang kapster nya sudah sampai di tempat pemesanan dan 3 digunakan untuk transaksi yang sudah selesai dikerjakan.

3.6 Halaman *Profile* dan *Info*



(a) (b)
Gambar 7. Halaman *Profile* (a) dan Halaman *Info* (b)

Pada halaman *profile*, *user* disediakan fitur untuk mengubah informasi pribadi dari akunnya mulai dari meng-*update* nama hingga *password*. Pada menu ini juga disisipkan tombol untuk keluar (*logout*) dari aplikasi. Sedangkan pada Halaman *info*, digunakan untuk menampilkan nama dan versi aplikasi.

3.7 Perancangan Tracking

3.7.1 Menyisipkan *Geolocation*

```
navigator.geolocation.getCurrentPosition(showPosition);
function showPosition(position) {
    latt = position.coords.latitude;
    lon = position.coords.longitude;
    latlon = new google.maps.LatLng(latt, lon)
}
```

Gambar 8. *Script* Penggunaan *Geolocation*

Pada Gambar 8 dijelaskan bagaimana mengambil koordinat menggunakan *geolocation*, yakni dengan membuat dua *variable* untuk menyimpan nilai *latitude* dan *longitude*. Pada *script* Gambar 3.7 digunakan *variable* *latt* dan *lon* untuk menyimpan nilai koordinat dan digabungkan menjadi sebuah koordinat yang sempurna (*latitude*, *longitude*) dengan fungsi *new google.maps.LatLng* dan disimpan dalam

variable *latlon*. Untuk membuat marker dapat melakukan *refresh* maka dibangun sebuah fungsi seperti Gambar 3.8 berikut.

```
updateMaps();
window.setInterval(updateMaps, 1100); }
Function updateMaps() {
    clearOverlays();
    var timestamp = new Date().getTime();
    var data = 'data_peta2.php?t=' + timestamp;
    $.get(data, {}, function(data) {
        $(data).find("marker").each(
            function() {
                var marker = $(this);
                var status = marker.attr("status")
                var icon = customIcons[status] || {};
                var latlng = new
                google.maps.LatLng(parseFloat(marker.attr("lat")),
                parseFloat(marker.attr("lng")));
                var html = "<b>this location is "+status+"</b><br><a
                href="#">I\'m on it!</a>";
                var marker = new google.maps.Marker({
                position : latlng, map : map,
                icon: icon.icon, shadow: icon.shadow, });
                google.maps.event.addListener(marker, 'click', function() {
                infoWindow.setContent(html);
                infoWindow.open(map, marker); });
                markersArray.push(marker);
                google.maps.event.addListener(marker, "click", function()
                {});
                });
            });
        });
    });
}
```

Gambar 9 *Script* Pengaturan *Update Map*

4. PENGUJIAN *GEOLOCATION* DAN *JQUERY* *LATEST*

Pada pengujian ini penulis membangun sebuah *prototype* terpisah dari aplikasi Kapster.ID, yakni potongan *script* mulai dari mengaktifkan *geolocation*, melakukan *looping* dan proses *insert* data. Pada *prototype* ini terdapat perbedaan pada langkah terakhir yakni melakukan *insert* data. Langkah ini dilakukan bertujuan untuk melihat *delay* dari data koordinat yang di inputkan. Percobaan ini dilakukan sebanyak 40 kali inputan data yang dilakukan pada 3 titik lokasi berbeda dengan tujuan melihat dimana lokasi terbaik dalam penggunaan *geolocation* dan banyaknya *error* data yang terjadi.

Dari pengujian ditemukan 2 jenis data, yang pertama adalah data koordinat dan data *NaN*. Data *NaN* didapat dari proses *AJAX* yang merubah *type* data yang diterima menjadi *type* data *float* atau desimal. Pada kondisi logika program yang dibangun, ada saat dimana kondisi *textbox* yang menerima koordinat bernilai kosong dan dilakukan proses *AJAX*, ketika di lakukan fungsi menukar *type* data maka yang terjadi adalah

data yang kosong (*null*) menjadi **NaN** (*Not A Number*).

Dalam pengolahan data statistik, untuk mendapatkan nilai rata-rata (*mean*) dapat dilakukan dengan menjumlahkan seluruh nilai data suatu kelompok sampel, kemudian dibagi dengan jumlah sampel tersebut (<https://www.rumusstatistik.com>, 2013) seperti persamaan (1) berikut.

$$\bar{x} = \frac{1}{n} (x_1 + x_2 + \dots + x_n)$$

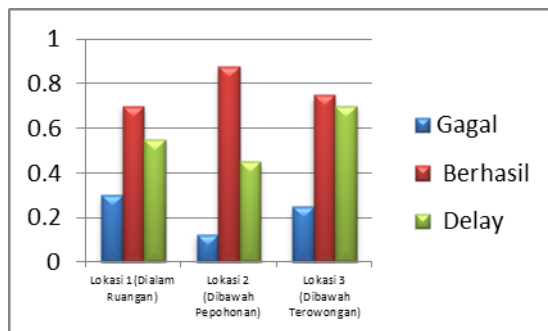
Gambar 10. Persamaan Perhitungan Mean

4.1 ANALISA PENGUJIAN

Pada Tabel 1 merupakan hasil dari pengujian yang telah dilakukan di atas. Dari Tabel 1 dapat dilihat pengaruh lokasi terhadap persentase *delay* data, data yang berhasil dan data gagal. Agar lebih detail, penulis menggunakan grafik dalam memaparkan data yang didapat seperti Gambar 11.

Tabel 1 Tabel Rata-Rata Pengujian Geolocation dengan jQuery Latest

Lokasi	Gagal	Gagal (%)	Berhasil	Berhasil (%)	Delay
1	0,3	30	0,7	70	0,55
2	0,125	12,5	0,875	87,5	0,45
3	0,25	25	0,75	75	0,7



Gambar 11 Grafik Perbandingan Data Antar Lokasi

Dari grafik pada Gambar 11 dapat disimpulkan bahwa lokasi pengujian kedua mendapatkan *rank* tertinggi dibandingkan lokasi uji lainnya. Terlihat tingkat keberhasilan tertinggi mencapai angka 0,875, tingkat gagal terendah di angka 0,125 dan *delay* terkecil pada angka 0,45. Sedangkan

dari sisi penyusunan sistem *tracking*, penggunaan *jQuery Latest* menimbulkan munculnya *error* data atau data bertipe **NaN** (*Not A Number*) karena kondisi dimana inputan koordinat dari *geolocation* di *refresh* oleh *jQuery Latest* masih belum terisi koordinat dan sudah dilakukan fungsi *parsetofloat* atau konversi ke bentuk angka dan diinputkan ke *database*.

Pengaruh dari data *error* atau data **NaN** berdampak pada *interface* sistem *tracking* yang sedang berjalan, yakni hilangnya *marker* beberapa saat karena tidak ditemukannya koordinat pada *database* yang diakses sampai fungsi *auto update marker* menemukan koordinat pada *database* atau terjadi proses *skip*. Hal ini menyebabkan *user* kehilangan jejak *marker* dalam beberapa waktu.

5. KESIMPULAN DAN SARAN

5.1 KESIMPULAN

Dari pengujian sistem *tracking* yang dilakukan terhadap aplikasi yang dibangun, didapatkan hasil bahwa pengujian sistem *tracking* menggunakan *Geolocation* dan *jQuery Latest* pada ruangan terbuka dengan kondisi dibawah pepohonan mendapatkan *rank* tertinggi dibandingkan lokasi uji lainnya, dengan persentase data berhasil sebanyak 77,5 %, data gagal 12,5 % dan rata-rata *delay* antar masuk data berkisar 0,45 detik per 40 data masuk. Disisi penyusunan sistem *tracking*, penggunaan *jQuery Latest* menimbulkan munculnya *error* data atau data bertipe **NaN** (*Not A Number*) karena kondisi dimana inputan koordinat dari *geolocation* di *refresh* oleh *jQuery Latest* masih belum terisi koordinat dan sudah dilakukan fungsi *parse to float* atau konversi ke bentuk angka dan diinputkan ke *database*. Data **NaN** ini juga berdampak ke *interface* sistem *tracking* karena akan terjadi *skip* atau *marker* akan hilang beberapa saat sampai menemukan koordinat pada *database* yang diakses.

5.2 SARAN

Saran dari penelitian ini yaitu:

1. Dari segi *interface* dapat dapat diperbaiki menjadi lebih *responsive* mengikuti layar *smartphone* Android.

2. Untuk penelitian selanjutnya dapat dilakukan uji coba pembangunan sistem *tracking* tanpa menggunakan jQuery *Latest* dengan harapan dapat meminimalisir danya *error* datadan dapat dilakukan pengujian antara sistem *tracking* tanpa menggunakan jQuery *Latest* dengan sistem *tracking* yang menggunakan jQuery *Latest*.
3. Dapat dilakukan penelitian menambah variabel terkait dengan faktor lain (faktor x) yang mempengaruhi nilai *delay* dan tingkat keberhasilan dan kegagalan data

DAFTAR PUSTAKA

- Bintang, Muhammad. "Sistem Pelacak Rute Kendaraan Dengan Teknologi GPS dan GPRS." (2009).
- Fathansyah, "Basis Data". *Informatika, Bandung* (2012).
- Indrajit. "Analisis dan Perancangan Sistem Berorientasi Object". *Informatika, Bandung* (2001).
- Jogiyanto, Hartono Mustakini. "Sistem teknologi informasi." *Andi, Yogyakarta* (2005).
- Raymond Jr, Mcleod. "Sistem Informasi Manajemen, edisi ke-7 terjemahan Teguh, PT." *Prehallindo, Jakarta* (2001).
- Sutedjo , Budi, Dharma Oetomo. "Perencanaan Dan Pembangunan Sistem Informasi." *Andi, Yogyakarta* (2005).
- Pressman, Roger S., "Rekayasa Perangkat Lunak Pendekatan Praktisi." *Buku kesatu, Terjemahan: Harnaningrum LN, edisi kedua, Andi, Yogyakarta* (1997).
- Sisephaputra, Bonda, Antok Supriyanto, dan Teguh Sutanto. "Sistem Pemantauan Keberadaan Jamaah Haji Menggunakan GPS Tracking Pada Smartphone Android (Studi Kasus: KbiH Al Ishlahiyyah Al Ghozaliyyah Tuban)." *Jurnal JSIKA 1.1* (2012).
- Widyantara, I Made Oka. " Penerapan Teknologi GPS Tracker Untuk Identifikasi Kondisi Traffik Jalan Raya". *Jurnal Teknologi Elektro Vol.14* (2015).