

# ANALISA SISTEM INFORMASI GEOGRAFIS PEMETAAN RUMAH SAKIT DI KOTA SAMARINDA BERBASIS WEB

**Novianita**

Sekolah Tinggi Manajemen Informatika Dan Komputer  
Samarinda

## ABSTRAK

*There have been many hospitals in Samarinda, but the rapid development of hospitals in Samarinda, not accompanied by the development of media support facilities and hospital information is still lacking especially in terms of mapping hospitals in Samarinda City. Currently, the hospital in Samarinda has 14 hospitals. To speed up and facilitate the search of data of hospitals that currently number 14 hospitals, the proposed problem is to build a web based application of Hospital Geographic Information System (GIS) in Samarinda. The research was conducted at the Samarinda City Health Office. Methods used in data collection include field studies (interviews and observations), and library studies. Data analysis was conducted qualitatively based on technological analysis, GIS data analysis, and analysis of GIS output information. The results showed that this geographic information system could display a map of Samarinda city along with the hospital information. This geographic information system can provide solutions in solving the search for the nearest specialist doctor and show the closest route to the hospital.*

*Keywords: Geographical Information Systems (GIS), hospital.*

## A. Pendahuluan

Saat ini Kota Samarinda berkembang menjadi kota dengan penghasilan utama dari perdagangan dan jasa, hal ini terjadi karena di kota ini mayoritas penduduknya adalah pendatang dari luar pulau Kalimantan seperti Jawa dan Sulawesi, dan kualitas pelayanan kesehatan yang prima tentu sangat didambakan oleh tiap masyarakat. Kualitas tenaga medis yang mumpuni, peralatan medis yang modern dan biaya pelayanan kesehatan yang murah adalah harapan semua lapisan masyarakat. Namun demikian tidak semua harapan masyarakat bisa di wujudkan. Kondisi pelayanan kesehatan masyarakat masih mengalami banyak kendala. Namun demikian semua stakeholder harus bekerja sama untuk meningkatkan kualitas pelayanan kesehatan masyarakat.

Sudah mulai banyak rumah sakit di Kota Samarinda, namun pesatnya perkembangan rumah sakit di Samarinda, tidak dibarengi dengan perkembangan fasilitas pendukung media dan informasi rumah sakit dirasa masih kurang terutama dalam hal pemetaan rumah sakit yang terdapat di Kota Samarinda. Selama ini informasi pemetaan Rumah Sakit di Kota Samarinda hanya sebatas papan penunjuk jalan yang tersebar dipinggiran jalan besar saja.

Saat ini Rumah Sakit yang ada di Samarinda berjumlah 14 Rumah Sakit. Untuk mempercepat dan mempermudah pencarian data Rumah sakit yang saat ini berjumlah 14 Rumah sakit, di usulkan dalam permasalahan tersebut adalah dengan membangun suatu aplikasi Sistem Informasi Geografis (SIG) Rumah Sakit di Kota Samarinda berbasis WEB.

Aplikasi ini berfungsi untuk menyajikan informasi berbasis lokasi melalui visualisasi dalam bentuk data spasial dan data atribut yang dikaitkan dengan kondisi geografis suatu wilayah dalam hal ini kondisi geografis di Kota Samarinda beserta rumah sakit yang tersebar.

Dalam mengimplementasikan Sistem Informasi Geografis ini, digunakan fasilitas dari Google Maps untuk pemetaannya, karena disamping layanan ini bersifat open source, Google Maps juga menyediakan berbagai jenis peta. Selain itu, ditambahkan juga fitur tambahan Augmented Reality yang dapat menampilkan konten-konten pendukung dari rumah sakit yang bisa berinteraksi dengan pengguna, sehingga penyajian informasi yang disampaikan diharapkan menjadi lebih interaktif. Dalam penelitian ini, kerja sama dilakukan dengan Dinas Kesehatan Kota Samarinda dalam hal pengumpulan data mengenai rumah sakit yang ada di wilayah Kota Samarinda.

## **B. Metode Penelitian**

### **1. Tempat dan Waktu Penelitian**

Penelitian dilaksanakan pada Dinas Kesehatan Kota Samarinda yang memiliki informasi seluruh dokter di Samarinda, dan mencari semua informasi serta data yang diperlukan langsung ke rumah sakit di Kota Samarinda. Penelitian ini dilaksanakan pada 10 November – 10 Desember 2016 sampai selesai.

### **2. Metode Pengumpulan Data**

Metode-metode yang digunakan dalam pengumpulan data selama penelitian adalah :

#### **a. Studi Lapangan**

Studi lapangan merupakan metode untuk memperoleh data yang dipergunakan dalam penulisan laporan dengan cara terlibat langsung dengan kegiatan di lapangan. Metode perolehan data dengan studi lapangan ini terbagi menjadi 2 (dua), yaitu :

##### **1) Wawancara (*Interview*)**

Mengadakan wawancara atau tanya jawab langsung kepada beberapa rumah sakit di Kota Samarinda guna mendapatkan informasi dalam penyusunan laporan skripsi mengenai informasi untuk pembuatan Sistem Informasi Geografis pemetaan rumah sakit pada Kota Samarinda.

##### **2) Pengamatan Langsung (*Observasi*)**

Dengan menggunakan metode pengamatan langsung, memperoleh data berupa dokumen dan aliran dokumen yang berkaitan dengan judul yang diteliti.

#### **b. Studi Pustaka**

Mempelajari buku-buku literatur yang berhubungan dengan judul yang diambil sebagai bahan acuan atau dasar pembahasan, serta untuk memperoleh landasan teori dari sistem yang akan dikembangkan, sehingga didalam penulisan laporan tidak menyimpang dari teori-teori yang sebelumnya telah ada dan diakui kebenarannya.

### **3. Metode Pengembangan Sistem**

Adapun metode yang digunakan dalam pembuatan Sistem Informasi Geografis pemetaan rumah sakit ini adalah metode *waterfall*. *Waterfall* merupakan SDLC tertua karena sifatnya yang natural. Urutan *waterfall* ini bersifat serial mulai dari proses perencanaan,

analisis, perancangan, implementasi, dan pemeliharaan. Model ini melakukan pendekatan secara sistematis dan urut. Disebut *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

#### **a. Analisis**

Adapun metode analisis data yang didapatkan yaitu secara kualitatif, dan metode pengumpulan data ini maka dilakukan suatu analisa yaitu sebagai berikut :

##### 1) Analisis Teknologi

###### a) Analisis perangkat keras saat perancangan

Perangkat keras yang diperlukan dalam membuat sebuah Sistem Informasi Geografis ini adalah *Processor Intel Core 2 Duo*, RAM 2GB, dan HDD Sata 320GB.

###### b) Analisis perangkat lunak saat perancangan

Perangkat lunak yang diperlukan dalam membuat sebuah Sistem Informasi Geografis ini adalah Sistem Operasi *Windows 7 Ultimate*, *Adobe Dreamweaver CS4*, *Xampp Win3.2.*, dan *Mozilla Firefox 20.0.1*.

###### c) Analisis perangkat keras yang dianjurkan untuk mengakses SIG yang dibangun.

(1) *Processor minimal Dual Core 2,2Ghz.*

(2) RAM minimal 1GB.

(3) Monitor dengan resolusi 1366 x 768.

(4) Komputer atau laptop yang telah terkoneksi internet.

###### d) Analisis perangkat lunak yang dianjurkan untuk mengakses SIG yang dibangun

(1) Sistem Operasi *Windows 7*

(2) Web browser *Mozilla Firefox* dan *Google Chrome*

##### 2) Analisis Data SIG

Berdasarkan kebutuhan SIG ada dua jenis data yang dibutuhkan :

a) Data spasial (referensi keruangan / geografis), peta yang berasal dari *server Google Maps* berdasarkan titik koordinat dengan parameter *latitude* (garis lintang) dan *longitude* (garis bujur).

b) Data atribut (pendukung informasi pada data spasial), merupakan data berbentuk table dimana table tersebut berisi informasi yang dimiliki oleh rumah sakit dalam bentuk objek.

##### 3) Analisis Informasi *Output* SIG

Berikut ini merupakan penguraian keluaran informasi dari sisi pengguna yang dibutuhkan sehingga dapat memenuhi kebutuhan:

a) Informasi letak Rumah Sakit.

b) Informasi profil Rumah Sakit.

c) Informasi Dokter Spesialis.

#### **b. Desain**

Desain system ini bertujuan untuk memberikan gambaran tentang sistem yang diusulkan. Alat bantu sistem yang digunakan dalam desain sistem ini antara lain :

##### 1) Diagram Alir (*Flowchart*)

Diagram Alir merupakan alat bantu yang banyak dipakai untuk menjelaskan algoritma pemrograman sehingga jelas rangkaian kegiatan yang dilakukan.

##### 2) Desain *Input*

Mendesain *form input* rumah sakit, *form input* dokter, *form input* admin.

### 3) Desain *output*

Mendesain tampilan *output* rumah sakit, tampilan *output* dokter, tampilan *output* admin.

### c. Implementasi

Setelah melakukan tahapan desain, maka rancangan yang dibuat agar bisa digunakan dengan menggunakan bahasa pemrograman PHP, Dalam penelitian ini tahapan implementasi berupa pembuatan antar muka (*user interface*) dalam bentuk tampilan *form*.

### d. Pengujian (*Testing*)

Tahapan terakhir yaitu pengujian / *testing*, adalah elemen kritis dari jaminan kualitas perangkat lunak & merepresentasikan kajian pokok dari spesifikasi, desain, dan pengkodean.

#### 1) Black Box

Black Box testing menyinggung uji coba yang dilakukan pada interface software. Pengujian blackbox berfokus pada persyaratan fungsional perangkat lunak. Pengujian ini memungkinkan analisis system memperoleh kumpulan kondisi input yang akan mengerjakan seluruh keperluan fungsional program. Tujuan metode ini mencari kesalahan pada:

- a) Fungsi yang salah atau hilang.
- b) Kesalahan pada interface.
- c) Kesalahan pada struktur data atau akses database.
- d) Kesalahan inisialisasi dan tujuan akhir.

Tabel 3.1 Pengujian *Black-Box*

No.	Item yang diuji	Hasil yang diharapkan
1.	Pengujian <i>input</i> data dokter.	Data dokter tersimpan ke <i>database</i> pada table dokter.
2.	Pengujian <i>input</i> empat kerja dokter.	Data tempat kerja dokter tersimpan ke <i>database</i> pada table kerja dokter.
3.	Pengujian <i>input</i> data rumah sakit.	Data rumah sakit tersimpan ke <i>database</i> pada table rumah sakit.
4.	Pengujian <i>output</i> data dokter.	menampilkan data dokter dari table dokter.
5.	Pengujian <i>output</i> data tempat kerja dokter.	Menampilkan rumah sakit tempat dokter bekerja dari table kerja dokter.
6.	Pengujian <i>output</i> data rumah sakit	menampilkan data rumah sakit dari table rumah sakit.

#### 2) White Box

Kesimpulan dari pengujian *whitebox* ini diperoleh dengan cara pemeriksaan detail *prosedural* web. Dengan menggunakan metode *whitebox testing* ini akan didapatkan kasus uji :

- a) Memberikan jaminan bahwa semua jalur independen pada suatu model telah digunakan paling tidak satu kali.
- b) Menguji seluruh struktur data internal yang menjamin validitas
- c) Mengeksekusi semua *loop* pada batasan mereka dan pada batas operasional mereka.

Dalam kasus ini prosedur yang di uji pada koneksi *database* dengan web, system pencarian nama dokter, system *output* peta rumah sakit.

Tabel 3.2 Pengujian *White-Box*

No.	Item yang diuji	Kode Program	Hasil
1.	Pengujian <i>input</i> dokter spesialis.	< <i>script</i> >	Daftar nama dokter spesialis
2.	Pengujian <i>input</i> alamat atau jalan	< <i>script</i> >	titikkoordinat lat = lng =
3.	Pengujian proses pencarian	< <i>script</i> >	Nama rumah sakit terdekat
4.	Pengujian menampilkan <i>route</i>	< <i>script</i> >	Tampil <i>route</i> titik asal menuju titik rumah sakit terdekat

#### e. Pemeliharaan

Pada tahapan pemeliharaan dilakukan setelah pengujian dilakukan. Pemeliharaan karena beberapa alasan, yaitu :

- 1) Kemungkinan system masih menyisakan masalah-masalah yang tidak terdeteksi selama masa pengujian sistem.
- 2) Adanya perubahan ataupun permintaan kebutuhan baru.
- 3) Kinerja sistem yang jadi menurun sehingga memungkinkan perubahan-perubahan dalam penulisan program.

### C. Hasil Penelitian Dan Pembahasan

#### 1. Hasil Analisis

##### a. Analisis Data

Dalam penelitian yang telah dilakukan, data-data yang digunakan dalam sistem informasi geografis ini adalah sebagai berikut:

- 1) Tabel *user*, dalam tabel user atributnya adalah *username* dan *password*.
- 2) Tabel dokter, dalam tabel dokter atributnya adalah nama dokter, alamat dokter, no hp, spesialis.
- 3) Tabel rumah sakit, dalam tabel rumah sakit atributnya adalah nama rumah sakit, alamat, titik koordinat, dan informasi rumah sakit.
- 4) Tabel kerja dokter ini adalah relasi dari tabel dokter dengan tabel rumah sakit.

### **b. Analisis Kebutuhan**

Dari hasil penelitian, kebutuhan-kebutuhan yang harus ada dalam sistem informasi geografis ini adalah sebagai berikut ;

- 1) *Form user*, digunakan untuk memasukkan data pengguna.
- 2) *Form dokter*, digunakan untuk memasukkan data dokter.
- 3) *Form rumah sakit*, digunakan untuk memasukkan data rumah sakit.
- 4) *Form kerja dokter*, digunakan untuk memasukkan data tempat kerja dokter.

### **c. Analisis Teknologi**

#### 1) Analisis Perangkat Keras

Dari hasil penelitian, dapat diketahui bahwa mengenai perangkat-perangkat keras yang dibutuhkan dalam pembuatan adalah sebagai berikut:

- a) *Processor Intel Core 2 Duo*.
- b) *Memory (DDR2) 2 GB*.
- c) *Harddisk minimal 320 GB*.

#### 2) Analisis Perangkat Lunak

Dari hasil penelitian, dalam segi perangkat lunak yang dibutuhkan dalam pembuatan aplikasi adalah sebagai berikut:

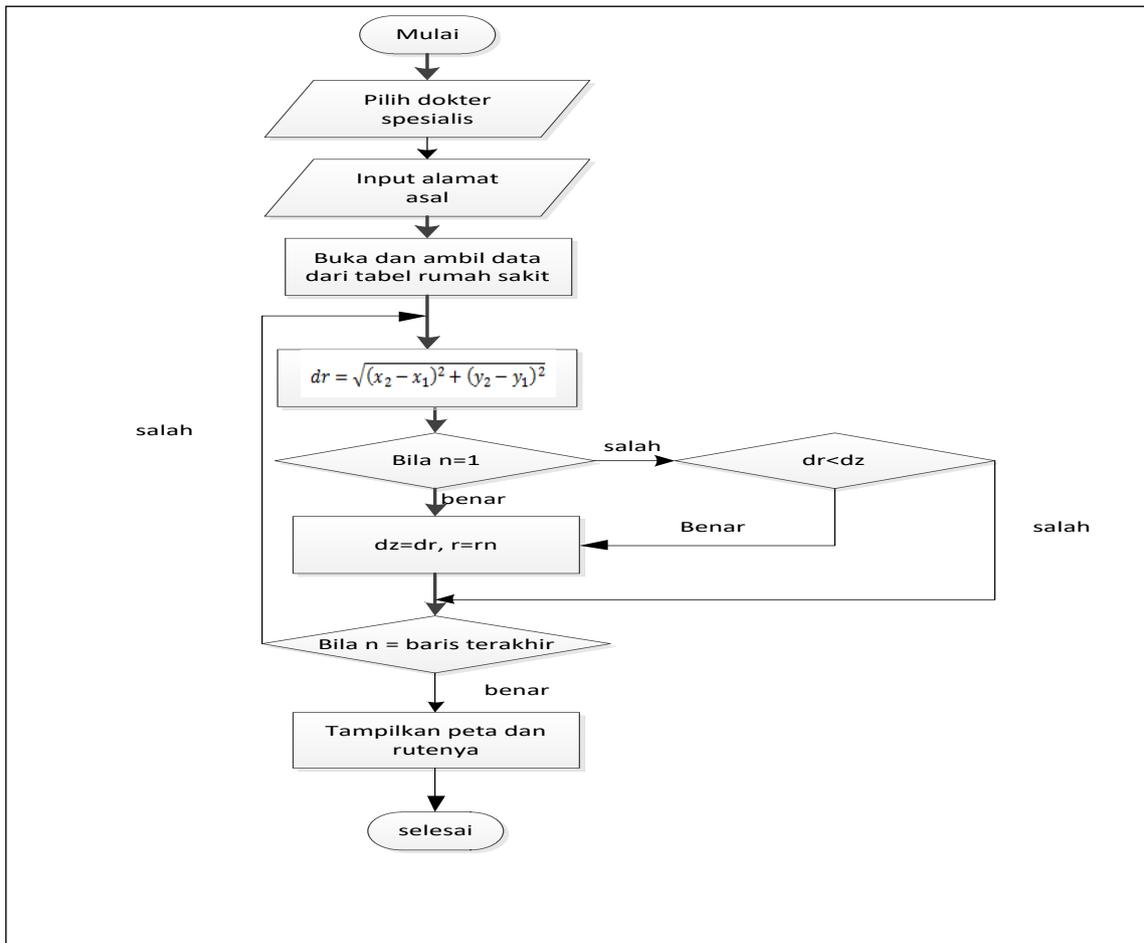
- a) *Sistem Operasi Windows 7*.
- b) *Adobe Dreamweaver CS4*.
- c) *Xampp*.
- d) *Adobe Photoshop CS 3*

## **2. Desain**

Desain sistem ini bertujuan untuk memberikan gambaran umum tentang sistem yang diusulkan. Adapun alat bantu yang digunakan dalam pengembangan sistem yaitu:

### **a. Flowchart Sistem**

*Flowchart* sistem menjelaskan tentang awal alur pencarian informasi rumah sakit yang pertama yaitu dengan menginputkan data spesialis, lalu menginputkan data alamat. kemudian dilanjutkan pada proses pencarian jarak terdekat dengan algoritma kartesius, kemudian akan menghasilkan informasi rumah sakit terdekat.



Gambar 4.2 Flowchart program proses perhitungan

Keterangan :

$n$  = nomor baris rumah sakit yang memiliki dokter spesialis yang dicari

$x_2$  = titik koordinat lintang alamat yang diinput

$y_2$  = titik koordinat bujur alamat yang diinput

$x_1$  = titik koordinat lintang rumah sakit

$y_1$  = titik koordinat bujur rumah sakit

$dr$  = jarak ke rumah sakit

$dz$  = jarak terdekat ke rumah sakit

$r$  = nama rumah sakit terdekat

$rn$  = nama rumah sakit ke titik asal

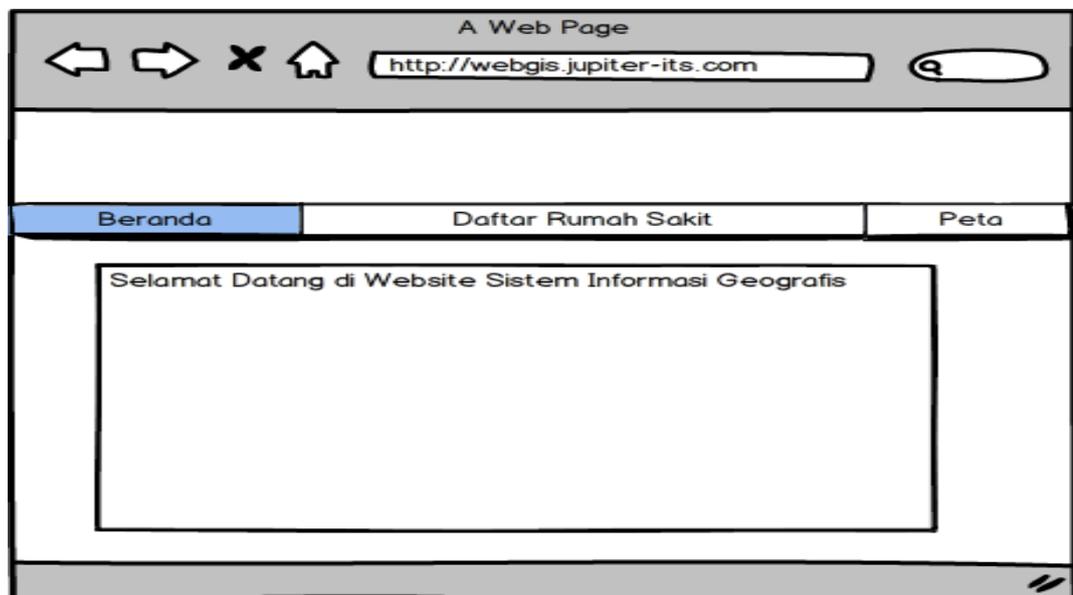
Algoritma pencarian dokter spesialis menggambarkan langkah- langkah untuk menentukan apakah ada informasi dokter spesialis terdekat dari jarak alamat yang telah diinputkan. Berikut tahapan Flowchart pencarian terdekat dari gambar 4.2 :

- 1) Tahap pertama dalam pencarian terdekat adalah input dokter spesialis yang akan dicari.
- 2) Input alamat atau nama jalan tempat asal.

- 3) Buka dan ambil data dari tabel rumah sakit, setelah itu membuat nomor baris antrian rumah sakit yang memiliki dokter spesialis yang dicari.
- 4) Selanjutnya proses perhitungan algoritma.
- 5) Jika yang dihitung pertama adalah nomor antrian pertama maka nilainya ( $dr$ ) menjadi nilai tolak ukur atau jarak terdekat ( $dz$ ).
- 6) Selanjutnya jika nomor baris pertama bukan antrian yang terakhir maka kembali dan akan mengambil data rumah sakit yang kedua untuk masuk dalam proses perhitungan algoritma.
- 7) Jika nilai jarak ( $dr$ ) antrian kedua tidak lebih kecil dari jarak terdekat ( $dz$ ) maka data akan dilewatkan.
- 8) Selanjutnya jika nomor antrian kedua bukan nomor antrian terakhir maka akan kembali ke tahapan nomor 6, dan jika nomor antrian kedua adalah nomor antrian terakhir maka akan menampilkan informasi rumah sakit terdekat dengan menampilkan peta dan route untuk menuju ke rumah sakit dari alamat asal yang telah diinput diawal.

## b. Desain *Form*

- 1) Desain *form* beranda.

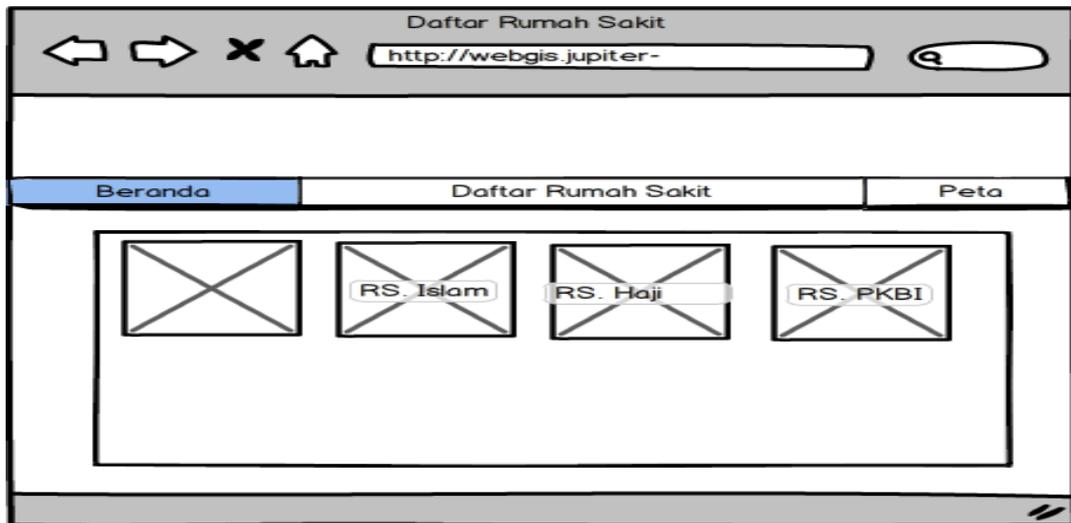


Gambar 4.3 *Desain* Tampilan *Form* beranda

Desain tampilan form beranda ini digunakan untuk menampilkan seputar informasi tentang dinas kesehatan Kota Samarinda dan rumah sakit yang ada di Samarinda.

- 2) Desain *form* daftar rumah sakit.

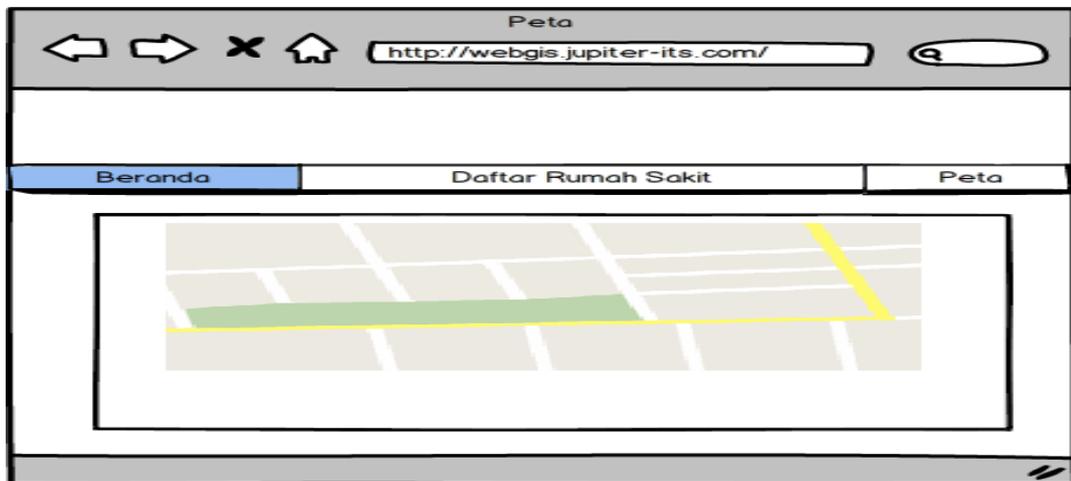
Pada gambar 4.4 menampilkan foto seluruh rumah sakit yang ada di Samarinda dan informasi yang ada tentang rumah sakit.



Gambar 4.4 Desain *Form* daftar rumah sakit

### 3) Desain *form* peta

Desain tampilan form peta ini digunakan untuk melakukan pencarian dokter spesialis. Pada desain form peta ini terdapat pilihan dokter spesialis yang ingin dicari dan input alamat atau nama jalan asal untuk pencarian dokter terdekat. Setelah dapat proses pencarian maka akan menampilkan peta titik alamat yang diinput menuju rumah sakit terdekat dan menampilkan informasi tentang rumah sakit tersebut beserta dengan nama dokter spesialis yang tersedia. Dapat dilihat pada gambar 4.5.

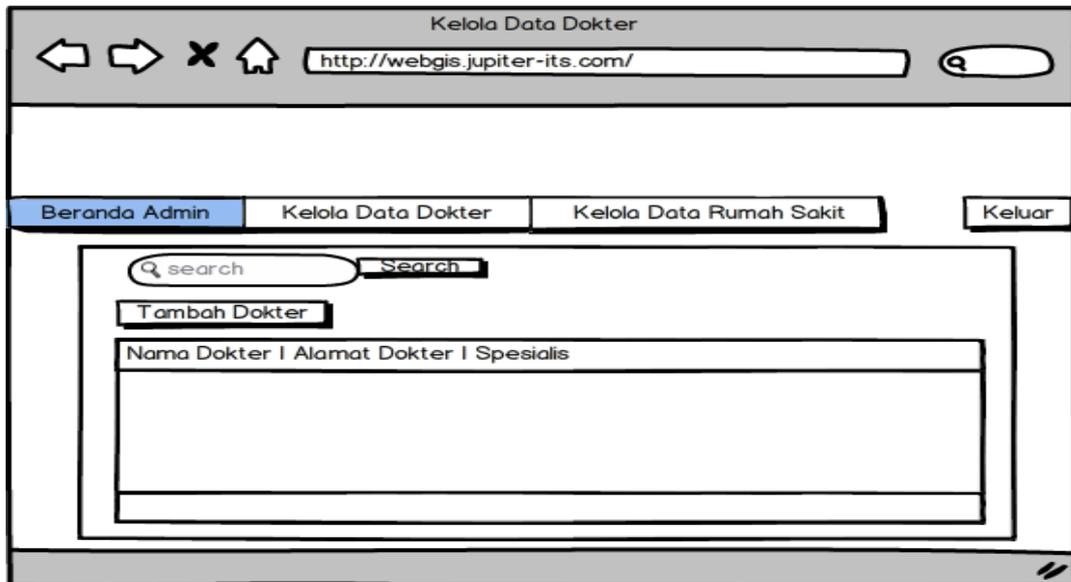


Gambar 4.5 Desain *Form* peta

### 4) Desain *form* menu kelola data dokter

Pada tampilan desain *form* kelola data dokter berisi tentang data dokter spesialis dan pada menu ini terdapat juga pilihan untuk cek di rumah sakit mana dokter bekerja, ada pilihan *edit* untuk merubah data informasi dokter, dan pilihan hapus untuk menghapus data dokter dari *database*. Dan dapat menambah data

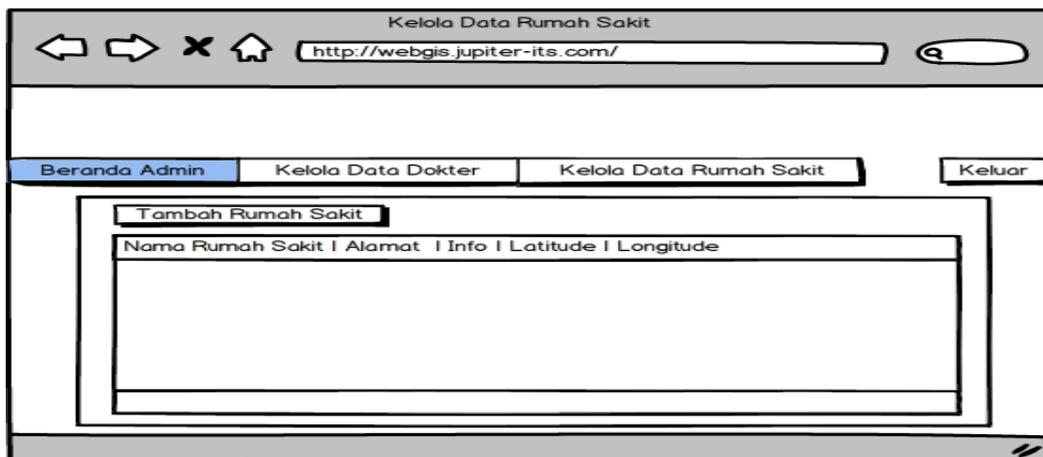
dokter ke *database*, serta dapat melakukan pencarian berdasarkan nama dokter dan spesialis. Tampilan *form* menu kelola data dokter dapat dilihat pada gambar 4.6.



Gambar 4.6 *Desain Form* menu kelola data dokter

5) *Desain form* tampilan menu kelola data rumah sakit

Pada tampilan menu kelola data rumah sakit berisi tentang informasi seluruh rumah sakit yang ada di Samarinda berupa nama rumah sakit, alamat rumah sakit, informasi rumah sakit, foto rumah sakit, titik kordinat *latitude* dan *longitude* . Pada menu ini terdapat pilihan untuk menambah data rumah sakit, edit untuk merubah data rumah sakit, dan hapus untuk menghapus data dari *database*.



Gambar 4.7 *Desain Form* tampilan Menu kelola data rumah sakit

### 3. Implementasi

#### a. Struktur Database

1) Nama Tabel: tb\_user

Primary key: id\_user

Tabel 4.1 Tabel user

NO	Nama	Data Type	Size	Keterangan
1	id_user	Int	11	id user
2	username	Varchar	20	username
3	password	Varchar	20	password

2) Nama Tabel : tb\_dokter

Primary key : id\_dokter

Tabel 4.2 Tabel dokter

NO	Nama	Data Type	Size	Keterangan
1	id_dokter	int	11	id dokter
2	nama_dokter	varchar	50	nama dokter
3	alamat	varchar	50	alamat dokter
4	HP	varchar	15	no hp dokter
5	Spesialis	varchar	20	spesialis

3) Nama Tabel : tb\_rumah\_sakit

Primary key : id\_rumah\_sakit

Tabel 4.3 Tabel rumah sakit

NO	Nama	Data Type	Size	Keterangan
1	id_rumah_sakit	int	11	id rumah sakit
2	Nama	varchar	50	nama rumah sakit
3	Alamat	varchar	50	alamat
4	Info	text		informasi rumah sakit
5	Lat	double		titik lintang
6	Lng	double		titik bujur
7	Foto	varchar	200	nama foto rumah sakit

4) Nama Tabel : tb\_kerja\_dokter

Primary key : id\_kerja\_dokter

Tabel 4.4 Tabel kerja dokter

NO	Nama	Data Type	Size	Keterangan
1	id_kerja_dokter	int	11	id kerja dokter
2	id_dokter	int	11	id dokter
3	id_rumah_sakit	int	11	id rumah sakit

## b. Tampilan Pengguna Menu Utama Pada Program

Pada gambar 4.8 adalah tampilan menu utama dari sistem informasi geografis. Di dalam sistem informasi geografis ini ada beberapa sub menu yang dapat diakses, yaitu daftar rumah sakit, dan peta.



Gambar 4.8 Tampilan menu utama

## c. Tampilan Pengguna Menu Daftar Rumah Sakit



Gambar 4.9 Tampilan daftar rumah sakit

Pada gambar 4.9 pengguna dapat melihat daftar Rumah Sakit berupa foto dan nama rumah sakit, beserta informasi tentang rumah sakit yang ada di Kota Samarinda.

## d. Tampilan Pengguna Menu Peta

Pada gambar 4.10 adalah tampilan peta utama dari sistem informasi geografis. Pada halaman peta ini pengguna dapat melakukan pencarian dokter spesialis yang bekerja di rumah sakit dengan memilih dokter spesialis yang ada pada *combox box* dan menginputkan dari alamat atau jalan dan merubah alamat menjadi titik koordinat

menggunakan geocoding seperti pada gambar 4.11, lalu sistem menghitung pencarian terdekat menggunakan algoritma seperti script pada gambar 4.12.

Gambar 4.10 Tampilan Menu Peta



```

52 function geocode($address){
53     $lat = 0;
54     $lng = 0;
55     $formatted_address = "";
56     $addr = urlencode($address);
57
58     $geocode_url = "http://maps.googleapis.com/maps/api/geocode/json?address=$addr&sensor=false";
59
60     $ch = curl_init($geocode_url);
61     curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
62     $result = curl_exec($ch);
63
64     $httpCode = curl_getinfo($ch, CURLINFO_HTTP_CODE);
65     curl_close($ch);
66
67     if ($httpCode == 200){
68         $geocode = json_decode($result);
69         $geo_status = $geocode->status;
70     }
71     else{
72         $geo_status = "HTTP_FAIL_{$httpCode}";
73     }
74
75     if ($geo_status == "OK"){
76         $lat = $geocode->results[0]->geometry->location->lat;
77         $lng = $geocode->results[0]->geometry->location->lng;
78         $formatted_address = $geocode->results[0]->formatted_address;
79     }
80     return array($lat, $lng, $formatted_address, $geo_status);
81 }

```

Gambar 4.11 Script Geocoding

```

5 function dekat($lng1,$lat1, $dokter){
6     $sql = "SELECT
7         K.*,
8         RS.nama,
9         RS.lat,
10        RS.lng
11        FROM tb_kerja_dokter AS K
12        JOIN tb_rumah_sakit AS RS ON RS.id_rumah_sakit = K.id_rumah_sakit
13        JOIN tb_dokter AS D ON D.id_dokter = K.id_dokter
14        WHERE D.Spesialis = '{$dokter}'
15        GROUP BY RS.id_rumah_sakit
16        ORDER BY K.id_dokter ASC";
17    $query = mysql_query($sql) or die(mysql_error());
18    $x      = 0;
19    $drum   = 0;
20    while($data = mysql_fetch_array($query)){
21        $ind      = $data['id_rumah_sakit'];
22        $id_pembanding[$x] = $data['id_rumah_sakit'];
23        $nt       = $data['nama'];
24        $lat2     = $data['lat'];
25        $lng2    = $data['lng'];
26        $dx      = $lat1-$lat2;
27        $dy      = $lng1-$lng2;
28        $dr      = sqrt(pow($dx,2)+pow($dy,2));
29        if ($id_pembanding[0] == $ind or $dr < $drum){
30            $drum = $dr;
31            $ntd  = $nt;
32            $longi = $lng2;
33            $lati  = $lat2;
34            $inds  = $ind    }
35        $x++;    }
36    return array($ntd, $longi, $lati, $inds);
37 }

```

Gambar 4.12 Script Pencarian

e. Tampilan login admin

Halaman Administrator

Username :

Password :

Gambar 4.13 Tampilan Login Admin

Antarmuka halaman *login* merupakan tampilan yang pertama kali muncul ketika aplikasi diakses oleh admin. Halaman ini berfungsi sebagai otentikasi keamanan dari sistem itu sendiri yang dimana terdapat perintah *input username* dan *input password* agar pengguna dapat mengakses sistem ini. Implementasinya terlihat seperti Gambar 4.13.

## f. Tampilan Admin Kelola Data Dokter

Pada halaman kelola data dokter ini dapat melihat semua data dokter, ada juga fasilitas pencarian berdasarkan nama dokter dan spesialis, fasilitas cek informasi di rumah sakit mana saja dokter bekerja, fasilitas tambah dokter, *edit* data dokter dan hapus data dokter dari *database*.



No	Nama Dokter	Alamat	Spesialis	Tempat Kerja	Aksi
1	dr. Didin Komarudin,Sp.THT-KL,M.Kes	Jl.Wolter Monginsidi 64 Timbau Tenggaraong	THT	Cek	Ubah   Hapus
2	dr. Benida Ernawati,Sp.An	Perum.Puspita Bengkuring Blok AB No.7 SMD	Anak	Cek	Ubah   Hapus
3	dr. Josef Budi Setiono,Sp.A	Jl.Palang Merah No.15.RT.042.Samarinda	Anak	Cek	Ubah   Hapus
4	dr. Fernando Taruly Tua Sitorus,Sp.OG	Jl.H.Adam Malik Samarinda	Kandungan	Cek	Ubah   Hapus
5	dr. A.Dalidjo, Sp.KJ	Jl.Kakap Kompl.	Jiwa	Cek	Ubah   Hapus
6	dr. Moh.Ishaq Hardiansya,Sp U	Jl.A.W.Syahrani Komp.Guru NO.52 RT.023 Smd	Urologi	Cek	Ubah   Hapus

Gambar 4.14 Tampilan Daftar Rumah Sakit

## g. Tampilan Admin Input Data Dokter



Nama Dokter :

Alamat :

No HP :

Spesialis :

Gambar 4.15 Tampilan Input Dokter

Pada gambar 4.15 tampilan admin ini berfungsi untuk menambahkan data dokter ke *database* berupa informasi nama dokter, alamat tempat tinggal dokter, nomor hp dokter, dan informasi tentang spesialisasi dokter.

#### h. Tampilan Admin Ubah Data Dokter

Pada halaman ini admin dapat merubah data informasi dokter seperti nama dokter, alamat dokter, no hp, dan informasi tentang spesialisasi dokter.

Sistem Informasi Geografis Rumah Sakit Kota Samarinda

Sistem Informasi Geografis Kelola Data Dokter Kelola Data Rumah Sakit Keluar

Nama Dokter : dr. Didin Komarudin,Sp.THT-KL,M.Kes

Alamat : Jl.Wolter Monginsidi 64 Timbau Tenggaraong

No HP : 080280281

Spesialis : THT

Update Data

Gambar 4.16 Tampilan Ubah Data Dokter

#### i. Tampilan Admin Data Kerja Dokter

Pada gambar 4.17 tampilan admin data kerja dokter ini admin dapat melihat data kerja dokter pada rumah sakit mana saja, admin juga dapat menambah tempat kerja, merubah data kerja dokter.

Sistem Informasi Geografis Rumah Sakit Kota Samarinda

Sistem Informasi Geografis Kelola Data Dokter Kelola Data Rumah Sakit Keluar

Nama Dokter : dr. Benida Ernawati,Sp.An

Alamat Dokter : Perum.Puspita Bengkuring Blok AB No.7 SMD

Spesialis : Anak

Tambah Kerja Dokter

No	Nama Rumah Sakit	Alamat Rumah Sakit	Aksi
1	RS. Islam	Jl. Gurami No. 18 Samarinda	<a href="#">Edit</a> <a href="#">Delete</a>
2	RS. Dirgahayu	Jl. Gn. Merbabu 62 Samarinda	<a href="#">Edit</a> <a href="#">Delete</a>

Gambar 4.17 Tampilan Data Kerja Dokter

#### j. Tampilan Admin Kelola Data Rumah Sakit

Pada gambar 4.18 tampilan admin data rumah sakit ini admin dapat menambah data rumah sakit, dapat merubah data rumah sakit dan menghapus data rumah sakit.

No	Foto	Nama rumah sakit	Alamat	Lat	Lng	Aksi
1		RS. AW Syahrani	Jl. Dr. Sutomo No 1 Samarinda	-0.481424	117.144058	Ubah   Hapus
2		RS. Islam	Jl. Gurami No. 18 Samarinda	-0.506914	117.159225	Ubah   Hapus
3		RS. Abdul Moeis	Jl. HAM Rivaddin Komplek Bumi Rindang Samarinda 75	-0.551892	117.120063	Ubah   Hapus

Gambar 4.18 Tampilan Kelola Data Rumah Sakit

#### k. Tampilan Admin Input Data Rumah Sakit

Nama Rumah Sakit :   
 Alamat :   
 lat :   
 lng :   
 Informasi Rumah Sakit : 

Path: p

  
 Foto :  No file chosen

Gambar 4.19 Tampilan Input Data Rumah Sakit

Pada gambar 4.19 tampilan admin input kerja dokter ini admin dapat menambah data rumah sakit berupa nama rumah sakit, alamat rumah sakit, titik lintang atau *Latitude*, titik bujur atau *Longitudinal*, mengisi informasi tentang rumah sakit, dan foto rumah sakit.

### 4. Pengujian Sistem

#### a. Pengujian *Blackbox*

Pengujian yang dilakukan untuk mengetahui fungsi-fungsi pada perangkat lunak apakah sudah memenuhi persyaratan yang di buat dalam untuk progam.

Tabel 4.5 Tabel Pengujian ( *Black Box* )

No	Item yang diuji	Hasil yang diharapkan	Jumlah pengujian		
			1	2	3
1	Pengujian <i>input</i> data dokter.	data dokter tersimpan ke <i>database</i> pada tabel dokter.	√	√	√
2	Pengujian <i>input</i> tempat kerja dokter .	data tempat kerja dokter tersimpan ke <i>database</i> pada tabel kerja dokter.	√	√	√
3	Pengujian <i>input</i> data rumah sakit.	data rumah sakit tersimpan ke <i>database</i> pada tabel rumah sakit.	X	√	√
4	Pengujian <i>output</i> data dokter.	menampilkan data dokter dari tabel dokter.	√	√	√
5	Pengujian <i>output</i> data tempat kerja dokter.	menampilkan rumah sakit tempat dokter bekerja dari tabel kerja dokter.	√	√	√
6	Pengujian <i>output</i> data rumah sakit	menampilkan data rumah sakit dari tabel rumah sakit.	√	√	√

#### b. Pengujian *White Box*

Pengujian sistem dilakukan dengan metode *white box* dan hal-hal yang diuji adalah fungsional sistem.

Tabel 4.6 Pengujian *White Box*

No.	Item Pengujian	Kode Program	Hasil Pengujian
1.	Input Dokter Spesialis Anak	<pre>&lt;select name="dokter" id="dokter"&gt; &lt;?php \$sql_rs = mysql_query("SELECT * FROM tb_dokter GROUP BY Spesialis ORDER BY Spesialis ASC"); while(\$data_ds = mysql_fetch_array(\$sql_rs)) { ?&gt; &lt;option value="&lt;?php echo \$data_ds['Spesialis']; ?&gt;"&gt;&lt;?php echo \$data_ds['Spesialis']; ?&gt;&lt;/option&gt; &lt;?php } ?&gt; &lt;/select&gt;</pre>	<ol style="list-style-type: none"> <li>1. Dr. Benida Ernawati, Sp. An</li> <li>2. dr. Josep Budi setiono, Sp.An</li> <li>3. dr. Rudy Albert Frans, Sp.An</li> <li>4. dr. Indra Tamboen, Sp.An</li> </ol>
2.	Input alamat jelawat, samarinda	<pre>function geocode(\$address){ \$lat = 0; \$lng = 0; \$formatted_address = ""; \$addr = urlencode(\$address); \$geocode_url =</pre>	Data koordinat <i>lat</i> = -0.5001596. <i>lng</i> = 117.1599021.

		<pre>"http://maps.googleapis.com/maps /api/geocode/json?address=\$addr&amp;sensor=false"; \$ch = curl_init(\$geocode_url); curl_setopt(\$ch, CURLOPT_RETURNTRANSFER, 1); \$result = curl_exec(\$ch); \$httpCode = curl_getinfo(\$ch, CURLINFO_HTTP_CODE); curl_close(\$ch); if (\$httpCode == 200){ \$geocode = json_decode(\$result); \$geo_status = \$geocode-&gt;status; } else{ \$geo_status = "HTTP_FAIL_\$httpCode"; } if (\$geo_status == "OK"){ \$lat = \$geocode-&gt;results[0]-&gt;geometry- &gt;location-&gt;lat; \$lng = \$geocode-&gt;results[0]-&gt;geometry- &gt;location-&gt;lng; \$formatted_address = \$geocode-&gt;results[0]- &gt;formatted_address; } return array(\$lat, \$lng, \$formatted_address, \$geo_status); }</pre>	
3.	<p>pencairan rumah sakit terdekat yang memiliki dokter spesialis anak dari alamat jelaawat samarinda</p>	<pre>function dekat(\$lng1,\$lat1, \$dokter) { \$sql = "SELECT K.*, RS.nama, RS.lat, RS.lng FROM tb_kerja_dokter AS K JOIN tb_rumah_sakit AS RS ON RS.id_rumah_sakit = K.id_rumah_sakit JOIN tb_dokter AS D ON D.id_dokter = K.id_dokter WHERE D.Spesialis = '{\$dokter}'GROUP BY RS.id_rumah_sakit ORDER BY K.id_dokter ASC"; \$query = mysql_query(\$sql) or die(mysql_error()); \$x = 0; \$drm = 0; while(\$data = mysql_fetch_array(\$query)) { \$ind = \$data['id_rumah_sakit']; \$id_pembanding[\$x] = \$data['id_rumah_sakit']; \$nt = \$data['nama']; \$lat2 = \$data['lat']; \$lng2 = \$data['lng']; \$dx = \$lat1-\$lat2;</pre>	<p>RS. Ibu &amp; Anak H. Thaha Bakrie dengan titik koordinat lat = -0.501821 lng = 117.154893</p>

		<pre> \$dy = \$lng1-\$lng2; \$drr = sqrt(pow(\$dx,2)+pow(\$dy,2));  if (\$id_pembanding[0] == \$ind or \$drr &lt; \$drrm) {     \$drrm = \$drr;     \$ntd = \$nt;     \$longi = \$lng2;     \$lati = \$lat2;     \$inds = \$ind } \$x++; } return array(\$ntd, \$longi, \$lati, \$inds); } </pre>	
4.	menampilkan route dari alamat asal menuju rumah sakit terdekat	<pre> function calcRoute(startt, endd) {     var start = startt;     var end = endd;     var request = {         origin:start,         destination:end,         travelMode:         google.maps.DirectionsTravelMode.DRIVING     };     directionsService.route(request,     function(response, status) {         if (status ==         google.maps.DirectionsStatus.OK) {             directionsDisplay.setDirections(response);         }     }); } </pre>	menampilkan titik asal jelaawat samarinda menuju titik RS. Ibu & Anak H. Thaha Bakrie.

Dari dari beberapa pengujian *Black Box* dan *White Box* di atas penulis dapat mengambil kesimpulan bahwa semua alur program berjalan dengan baik sesuai dengan fungsinya masing – masing.

## 5. Pemeliharaan

Pada tahapan pemeliharaan dilakukan setelah pengujian dilakukan. Bagian dari pemeliharaan sistem informasi geografis ini, yaitu :

- a. Mengupdate *database* jika ada informasi baru.
- b. Memastikan proses pencarian dapat berjalan dengan baik.
- c. Memastikan *domain* tetap aktif agar bisa di akses *online*.

## D. Kesimpulan

Dari hasil penelitian dan pembahasan yang telah diuraikan pada bab sebelumnya, maka penulis dapat menarik kesimpulan sebagai berikut:

1. Dalam pembuatan sistem informasi geografis ini diperlukan data nama dokter, nama dan titik koordinat rumah sakit.

2. Bahwa sistem informasi geografis ini dapat menampilkan peta kota samarinda, menampilkan informasi rumah sakit.

Sistem informasi geografis ini dapat memberikan solusi dalam pemecahan masalah pencarian dokter spesialis terdekat dan menunjukkan rute terdekat menuju rumah sakitnya.

## DAFTAR PUSTAKA

- Agustina, M,S, 2010, Mendesain Website Dinamis dan Menarik dengan Adobe Dreamweaver CS4. Yogyakarta : Andi.
- Al-Bahra, 2005, Analisis dan Desain Sistem Informasi. Yogyakarta : Graha Ilmu.
- Aryono, dkk 2007, Pedoman Penyelenggaraan Pelayanan di Rumah Sakit. Jakarta : Departemen Kesehatan RI.
- Assauri, Sofyan,2004,Manajemen Pemasaran (Dasar, Konsep Dan Strategi). Jakarta :Grafindo Persada.
- Fathansyah, 2004, Basis Data. Bandung : Informatika.
- Hadi, Ardi Al, 2011, <http://ardhyal.wordpress.com/2010/11/03/titik-dan-baris-diagram-kartesius/> diakses tanggal 8 Juli 2014.
- Jogiyanto, 2005, Analisis & Desain Sistem Informasi. Yogyakarta : Andi.
- Prahasta, Eddy, 2009, Konsep-konsep dasar sistem informasi geografis. Bandung :Informatika.
- Pressman, Roger S, 2007. Rekayasa Perangkat Lunak: pendekatan praktisi (buku 1), Yogyakarta : Andi.
- Rafiudin, Rahmat, 2006, Protokol-Protokol Esensial *Internet*. Yogyakarta : Andi.
- Sirenden, Bernadus Herdi, dan Dachi, Ester Laeka, 2012, Buat Sendiri Aplikasi Petamu Menggunakan *CodeIgniter* dan *Google Maps API*. Yogyakarta : Andi.
- Sommerville, Ian, 2003, *Software Engineering*. Jakarta : Erlangga.
- Sutedjo, Budi,2004, Perencanaan dan Pembangunan Sistem Informasi. Yogyakarta : Andi
- Syaiful, M, A, 2010. Membangun Sistem Navigasi di Surabaya Menggunakan Google Maps API. <http://core.kmi.open.ac.uk/download/pdf/12344341.pdf>, diakses tanggal 18 Desember 2013.
- Xampp, <http://www.apachefriends.org/en/xampp-windows.html>, diakses pada tanggal 18 Desember 2013