

APLIKASI GEO-LOCATION MEDICAL BERBASIS ANDROID

TUGAS AKHIR

Oleh :

ENI KURNIA 3311211024

Disusun untuk memenuhi syarat kelulusan Program Diploma III



PROGRAM STUDI TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

BATAM

2014

HALAMAN PENGESAHAN

APLIKASI *GEO-LOCATION MEDICAL* BERBASIS ANDROID

Oleh :

ENI KURNIA (3311211024)

Tugas Akhir ini telah diterima dan disahkan
sebagai persyaratan untuk memperoleh gelar

Ahli Madya

di

PROGRAM STUDI DIPLOMA 3 TEKNIK INFORMATIKA

POLITEKNIK NEGERI BATAM

Batam, 2015

Disetujui oleh:

Pembimbing I,

Pembimbing II,

Dwi Ely Kurniawan, M. Kom

NIK. 112094

Supardianto, S. ST

NIK. 113105

HALAMAN PERNYATAAN

Dengan ini, saya:

NIM : 3311211024

Nama : ENI KURNIA

adalah mahasiswa Teknik Informatika Politeknik Batam yang menyatakan bahwa tugas akhir dengan judul:

APLIKASI GEO – LOCATION MEDICAL BERBASIS ANDROID

disusun dengan:

1. tidak melakukan plagiat terhadap naskah karya orang lain
2. tidak melakukan pemalsuan data
3. tidak menggunakan karya orang lain tanpa menyebut sumber asli atau tanpa izin pemilik

Jika kemudian terbukti terjadi pelanggaran terhadap pernyataan di atas, maka saya bersedia menerima sanksi apapun termasuk pencabutan gelar akademik.

Lembar pernyataan ini juga memberikan hak kepada Politeknik Negeri Batam untuk mempergunakan, mendistribusikan ataupun memproduksi ulang seluruh hasil Tugas Akhir ini.

Batam, Januari 2015

ENI KURNIA
3311211024

KATA PENGANTAR

Dengan rahmat dan rasa syukur penulis haturkan kehadiran Allah S.WT., atas karunia-Nya Penulis dapat menyelesaikan laporan Tugas Akhir ”Aplikasi *Geo-Location Medical* Berbasis *Android*” dengan tepat waktu.

Penulisan laporan Tugas Akhir ini dimaksudkan untuk persyaratan kelulusan matakuliah Tugas Akhir. Dalam pengerjaannya mendapat banyak arahan dan bimbingan dari berbagai pihak, oleh karena itu dalam kesempatan ini Penulis hendak mengucapkan ungkapan terima kasih yang tak terhingga kepada:

1. Allah S.W.T. yang senantiasa memberikan limpahan anugerah dan nikmatNya kepada Penulis.
2. Ayahanda Muhammad Ali Hamzah dan Ibunda Nurmia yang selalu memberikan semangat, kasih sayang, doa serta dukungan kepada Penulis.
3. Bapak Dwi Ely Kurniawan M. Kom, selaku Dosen Pengampuh dan Dosen Pembimbing I yang telah meluangkan banyak waktu dan tak henti-hentinya memberikan motivasi, bimbingan serta arahan demi kelancaran Tugas Akhir ini.
4. Bapak Supardianto, S.ST., selaku Dosen Pembimbing II dan Dosen Wali Teknik Informatika.
5. Teman-teman IF A Karyawan, terimakasih atas kebersamaan selama 3 tahun ini.
6. Seluruh pihak yang ikut serta memberikan motivasi dan membantu dalam menyelesaikan Laporan Tugas Akhir ini.

Semoga Tugas Akhir ini bermanfaat bagi pembaca, khususnya bagi yang ingin mengembangkan analisis serupa.

Batam, Desember 2014

Penulis

ABSTRAK

APLIKASI GEO-LOCATION MEDICAL BERBASIS ANDROID

Fasilitas kesehatan di Kota Batam yang mencukupi kebutuhan tidak dilengkapi dengan informasi yang tersebar di kalangan penduduk. Sulitnya mencari informasi mengenai kesehatan menjadi salah satu kendala untuk mendapatkan kebutuhan mengenai kesehatan. Salah satu cara yang mudah untuk mendapatkan informasi kesehatan yaitu menyatukan setiap pusat kesehatan kedalam suatu media elektronik yang mudah di akses. Langkah menyatukan pusat kesehatan ini dipermudah dengan adanya teknologi *mobilephoneAndroid* yang bisa digunakan oleh siapapun, kapanpun, dan dimanapun. Pencarian informasi kesehatan menjadi lebih mudah dan lebih praktis.

Kata Kunci : Kesehatan, *Android*

ABSTRACT

APLIKASI GEO-LOCATION MEDICAL BERBASIS ANDROID

Health facility in Batam which provide for not equipped with the information spread among the population. The difficulty of finding information about health to be one of the obstacles to getting health needs. One easy way to get health information that unites each health center into an electronic medium that is easy to access. Step unite health center is made easy with the Android mobilephone technology that can be used by anyone, anytime, and anywhere. Search health information becomes easier and more practical.

Keywords: *Health, Android*

DAFTAR ISI

APLIKASI GEO-LOCATION MEDICAL BERBASIS ANDROID.....	i
HALAMAN PENGESAHAN.....	ii
HALAMAN PERNYATAAN	iii
KATA PENGANTAR	iv
ABSTRAK	v
ABSTRACT	vi
DAFTAR ISI.....	vii
DAFTAR GAMBAR	ix
DAFTAR TABEL.....	x
BAB I PENDAHULUAN	1
1.1 Latar Belakang.....	1
1.2 Rumusan Masalah.....	2
1.3 Batasan Masalah	2
1.4 Tujuan.....	2
1.5 Sistematika Penulisan	2
BAB II LANDASAN TEORI	4
BAB III ANALISIS DAN PERANCANGAN	13
3.1 Analisis Sistem	13
3.1.3 Analisis Kebutuhan Fungsional	15
3.2 Perancangan Sistem	15
3.3 Sequence Diagram	17
3.4 Class Diagram.....	18
3.5 Perancangan Antarmuka.....	19
BAB IV HASIL DAN PEMBAHASAN	23
4.1 Hasil Implementasi	26
4.2 Implementasi Class	27
4.3 Implementasi Antarmuka	28
4.4 Pengujian dan Pembahasan	34

4.4.1	Strategi Pengujian.....	34
4.4.2	Deskripsi Pengujian.....	34
4.4.3	Hasil Pengujian.....	34
BAB V KESIMPULAN DAN SARAN.....		38
4.5	Kesimpulan.....	38
4.6	Saran	38

Daftar Pustaka

DAFTAR GAMBAR

Gambar 1. Deskripsi Umum Sistem.....	13
Gambar 2. Use Case Diagram.....	15
Gambar 3. Sequence Diagram MenampilkanData.....	17
Gambar 4. Sequence Diagram Menampilkan Peta	17
Gambar 5. Class Diagram	18
Gambar 6. ER Diagram.....	19
Gambar 7. Perancangan Antarmuka Home.....	19
Gambar 8. Perancangan Antarmuka Menu Tentang dan Bantuan	20
Gambar 9. Perancangan Antarmuka Menu Pencarian	22
Gambar 10. Halaman utama aplikasi (Home).....	28
Gambar 11. Halaman Menu Pencarian	30
Gambar 12. Halaman Menu Pencarian	30
Gambar 13. Tampilan Peta Lokasi.....	32
Gambar 14. Halaman Menu Tentang	33
Gambar 15. Halaman Menu Bantuan.....	33

DAFTAR TABEL

Tabel 1. Perbandingan Penelitian.....	4
Tabel 2. Deskripsi Halaman Menu Utama.....	20
Tabel 3. Deskripsi Halaman Menu Tentang	20
Tabel 4. Deskripsi Halaman Menu Tentang	21
Tabel 5. Deskripsi Halaman Menu Pencarian.....	23
Tabel 6. Tabel fasilitas_kesehatan	26
Tabel 7. Tabel Dokter	26
Tabel 8. Tabel Fasilitas Medis	27
Tabel 9. Tabel Implementasi Class	27

BAB I

PENDAHULUAN

1.1 Latar Belakang

Kesehatan adalah keadaan sejahtera dari badan, jiwa, dan sosial yang memungkinkan masyarakat hidup produktif secara sosial dan ekonomis. Setiap hari masyarakat membutuhkan penanganan kesehatan yang cepat. Keterbatasan informasi mengenai letak lokasi rumah sakit atau klinik, serta fasilitas yang ada untuk menangani itu kadang mengakibatkan suatu hal yang fatal bagi mereka yang membutuhkan penanganan medis tepat waktu. Seiring dengan laju pertumbuhan penduduk Kota Batam pada tahun 2014 yang berjumlah 1.140.480 jiwa. Menjadi tantangan bagi Pemerintahan Kota Batam khususnya Dinas Kesehatan Kota Batam untuk menyediakan sarana pelayanan kesehatan yang memadai bagi masyarakat kota Batam. Profil kesehatan Kota Batam juga diharapkan dapat digunakan sebagai sarana pemantauan, pembinaan dan pengawasan upaya program dan pelayanan kesehatan kota, karena sebagian besar masyarakat Kota Batam masih sulit mendapatkan pelayanan kesehatan walaupun dalam skala minimal.

Di era modern ini *Android* merupakan suatu teknologi yang sudah umum dan hampir dimiliki oleh setiap orang. Untuk menjalankan aplikasi layanan berbasis *smartphone* ini, telepon selular harus berjalan pada sistem operasi yang mendukung penuh. Salah satu sistem operasi yang mendukung penuh layanan tersebut adalah sistem operasi *Android*.

Berdasarkan latar belakang tersebut maka dibutuhkanlah sebuah sistem yang mampu menjadi media informasi tentang lokasi, fasilitas yang tersedia, ketersediaan tim medis atau dokter, serta pelayanan kesehatan lain yang bersifat *mobile* dan memanfaatkan sistem *geo-location* sebagai salah satu fasilitas pencarian yang baik untuk mendapatkan informasi lokasi dan kebutuhan akan penanganan masalah kesehatan dapat segera diatasi dengan baik.

1.2 Rumusan Masalah

Adapun rumusan masalah yang adadalam pembuatan aplikasi ini antara lain :

1. Bagaimana merancang sebuah aplikasi berbasis *Android* yang menyediakan fitur untuk memberikan informasi berupa lokasi, fasilitas kesehatan, serta pelayanan kesehatanlain ?
2. Bagaimana mengimplementasikan aplikasi yang sudah dibangun menjadi sebuah sistem informasi berbasis *Android* yang dibutuhkan pengguna dalam pencarian lokasi pusat kesehatan terdekat?

1.3 Batasan Masalah

Adapun batasan masalah yang dihadapi dalam pembuatan aplikasi ini antara lain :

1. Aplikasi ini hanya untuk wilayah Kota Batam.
2. Aplikasi ini mempunyai parameter data berupa informasi berdasarkan letak lokasi, fasilitas kesehatan serta pelayanan kesehatanyang tersedia.
3. Aplikasi ini berjalan di Platform *Android 4.0*

1.4 Tujuan

Adapun tujuan dari pembuatan aplikasi ini antara lain :

1. Membangun sebuah aplikasi *geo-location* berbasis *Android* yang menyediakan informasi berupa lokasi, fasilitas kesehatan, serta pelayanan kesehatan lain.
2. Mengimplementasikan aplikasi *geo-location* berbasis *Android* menjadi sebuah sistem informasi yang dibutuhkan pengguna dalam mencari pusat kesehatan terdekat.

1.5 Sistematika Penulisan

BAB I PENDAHULUAN

Bab ini diuraikan tentang latar belakang masalah, rumusan masalah, batasan masalah, tujuan dan sistematika penulisan.

BAB II LANDASAN TEORI

Bab ini diuraikan tentang tinjauan pustaka, serta dasar teori berupa pengertian kesehatan, pengertian *geo -Location*, pengertian

Android4.0, pengertian *PHP(Personal Home Page)*, Pengertian *MySql*, Pengertian *Eclipse*, Pengertian *API Google Mapss*, Pengertian *PhoneGap*.

BAB III : ANALISIS DAN PERANCANGAN

Bab ini diuraikan tentang deskripsi umum sistem berupa analisis sistem dan perancangan sistem yang dibuat berupa penggambaran sistem menggunakan sequence diagram menggunakan *Visual Paradigm for UML CE 10.1*, dan Diagram *ERD*.

BAB IV : IMPLEMENTASI DAN PENGUJIAN

Bab ini diuraikan tentang skema pembuatan aplikasi secara keseluruhan berupa implementasi sistem, implementasi antarmuka, serta pengujian dan pembahasan.

BAB V : KESIMPULAN DAN SARAN

Bab ini diuraikan tentang kesimpulan dan saran.

BAB II

LANDASAN TEORI

2.1 Tinjauan Pustaka

Aplikasi *geo-location medical* berbasis *Android* ini adalah sebuah media yang berkaitan dengan pelayanan kesehatan masyarakat yang bertujuan untuk memberikan informasi tentang lokasi rumah sakit atau klinik, fasilitas kesehatan yang tersedia, serta ketersediaan tim medis.

Tabel 1. Perbandingan Penelitian

NO	Parameter Perbandingan	Aplikasi yang sudah ada (batamkota.go.id)	Aplikasi yang pernah dilakukan (Batam <i>Medical Website</i> oleh Sherly Zilvia 3311101007/ 2014)	Aplikasi yang akan dirancang (Tugas Akhir Aplikasi <i>Geo – Location Medical</i> Berbasis <i>Android</i>)
1.	Informasi	Alamat, nomor telpon, tim medis, fasilitas	Alamat, nomor telpon, tim medis, fasilitas, artikel	Alamat, nomor telpon, tim medis, fasilitas
2.	Bahasa Pemrograman	Java	PHP, JavaScript	JavaScript, html5, css
3.	Platform	Website	Website	Android 4.0
4.	DBMS	MySql	MySql	Mysql
5	Ruang Lingkup Peta	Tidak menampilkan Peta	Tidak menampilkan Peta	Kota Batam

2.2 Dasar Teori

Adapun dasar teori yang digunakan dalam pembuatan aplikasi ini adalah:

2.2.1 Kesehatan

Sehat Menurut Ahli *WHO*(1948) kesehatan adalah sebagai “suatu keadaan fisik, mental, dan sosial kesejahteraan dan bukan hanya ketiadaan penyakit atau kelemahan”. Pada tahun 1986, *WHO*, dalam Piagam Ottawa untuk Promosi Kesehatan, mengatakan bahwa pengertian kesehatan adalah “sumber daya bagi kehidupan sehari-hari, bukan tujuan hidup Kesehatan adalah konsep positif menekankan sumber daya sosial dan pribadi, serta kemampuan fisik.

a) Rumah Sakit

Beberapa pengertian rumah sakit yang dikemukakan oleh para ahli (Azrul Azwar, 1996), diantaranya :

- a. Menurut Assosiation Hospital Care (1947) rumah sakit adalah pusat dimana pelayanan kesehatan masyarakat, pendidikan serta penelitian kedokteran diselenggarakan.
- b. Menurut American Hospital Assosiation (1974) rumah sakit adalah suatu alat organisasi yang terdiri dari tenaga medis professional yang terorganisir serta sarana kedokteran yang permanen menyelenggarakan pelayanan kedokteran, asuhan keperawatan yang berkesinambungan, diagnosis serta pengobatan penyakit yang diderita oleh pasien.
- c. Menurut Wolper dan Pena (1997) rumah sakit adalah tempat dimana orang sakit mencari dan menerima pelayanan kedokteran serta tempat dimana pendidikan klinik untuk mahasiswa kedokteran, perawat, dan tenaga profesi kesehatan lainnya diselenggarakan

b) Klinik

Menurut Dendy(2008) Klinik adalah Organisasi kesehatan yang bergerak dalam penyediaan pelayanan kesehatan kuratif(diagnosis dan pengobatan), biasanya terhadap satu macam gangguan kesehatan. Klinik juga biasa disebut balai pengobatan khusus.

c) Puskesmas

Menurut DepKes 2008 Mendefinisikan PUSKESMAS sebagai Suatu unit organisasi yang bergerak dalam bidang pelayanan kesehatan yang berada di garda terdepan dan mempunyai misi sebagai pusat pengembangan pelayanan kesehatan, yang melaksanakan pembinaan dan pelayanan kesehatan secara menyeluruh dan terpadu untuk masyarakat di suatu wilayah kerja tertentu yang telah ditentukan secara mandiri dalam menentukan kegiatan pelayanan namun tidak mencakup aspek pembiayaan. Puskesmas merupakan unit pelayanan kesehatan yang letaknya berada paling dekat ditengah-tengah masyarakat dan mudah dijangkau

dibandingkan dengan unit pelayanan kesehatan lainnya (Rumah Sakit Swasta maupun Negeri).

d) Apotek

Dalam peraturan pemerintah nomor 25 tahun 1980 yang dimaksud apotek adalah "suatu tempat tertentu, tempat dilakukan pekerjaan kefarmasian dan penyaluran obat kepada masyarakat" (Harianto, Nana Khasanah dan Sudibyo Supardi : 2005). Adapun tugas dan fungsi apotek adalah sebagai tempat pengabdian profesi seorang apoteker yang telah mengucapkan sumpah jabatan, sarana farmasi yang melaksanakan peracikan, pengubahan bentuk, pencampuran dan penyerahan obat atau bahan obat, dan sarana penyalur perbekalan farmasi yang harus menyebarkan obat yang diperlukan masyarakat secara meluas dan merata.

2.2.2 Geo – Location

Menurut Saputra Ichsana (2011) Geo-location adalah prosedur pelacakan dan pemetaan yang memungkinkan pengguna internet lokasi geografis. *Geo location* adalah identifikasi lokasi geografis dunia nyata dari sebuah objek, seperti radar, ponsel atau komputer terminal yang terhubung ke Internet. *Geo location* mungkin merujuk pada praktek menilai lokasi, atau ke lokasi yang dinilai sebenarnya. *Geo location* berkaitan erat dengan penggunaan sistem penentuan posisi tetapi dapat dibedakan dari itu dengan penekanan lebih besar pada penentuan lokasi bermakna (misalnya alamat jalan) bukan hanya satu set koordinat geografis.

2.2.3 Android Versi 4.0

a) Pengertian Android

Android adalah suatu sistem operasi yang berjalan pada smartphone saat ini dan menyesuaikan spesifikasi di kelas *low-end* hingga *high-end*. Hampir semua vendor saat ini mengembangkan produknya dengan sistem operasi *Android*, karena peminatnya yang semakin meningkat tajam. *Android* ini diresmikan pada tahun 2007 seiring dengan berdirinya *Open Handset Alliance-konsorsium hardware, software*, dan perusahaan telekomunikasi yang ditujukan untuk

memajukan standar perangkat selular.perkembangan*Android* dari awal hingga sekarang ini sudah mencapai 19 kali.

1. 2.1 *Android 1.0*
2. 2.2 *Android 1.1*
3. 2.3 *Android 1.5 Cupcake*
4. 2.4 *Android 1.6 Donut*
5. 2.5 *Android 2.0 Eclair*
6. 2.6 *Android 2.0.1 Eclair*
7. 2.7 *Android 2.1 Eclair*
8. 2.8 *Android 2.2–2.2.3 Froyo*
9. 2.9 *Android 2.3–2.3.2 Gingerbread*
10. 2.10 *Android 2.3.3–2.3.7 Gingerbread*
11. 2.11 *Android 3.0 Honeycomb*
12. 2.12 *Android 3.1 Honeycomb*
13. 2.13 *Android 3.2 Honeycomb*
14. 2.14 *Android 4.0–4.0.2 Ice Cream Sandwich*
15. 2.15 *Android 4.0.3–4.0.4 Ice Cream Sandwich*
16. 2.16 *Android 4.1 Jelly Bean*
17. 2.17 *Android 4.2 Jelly Bean*
18. 2.18 *Android 4.3 Jelly Bean*
19. 2.19 *Android 4.4 KitKat*

b) Kelebihan *Android*

- a. *User Friendly*, sangat mudah mengoperasikan komputer hanya dengan belajar beberapa hari bahkan beberapa jam saja, dan ini juga melekat pada *Android* yang berjalan pada *Smartphone*.
- b. *Notifications*, sangat mudah mendapatkan notifikasi dari *smartphone Android* dengan mengatur beberapa akun *Email, SMS, Voice Dial, Update* dan lain sebagainya.

- c. Tampilan, Dari segi tampilan, *Android* tidak kalah bagusnya dari *iOs* miliknya *Apple*, karena memang dari awal *Android* hampir mengusung teknologi *iOs*, hanya saja ini versi murahannya.
- d. *Open Source, Operating system* ini memang dibuat *open source* oleh penciptanya, karena memang berbasis *kernel Linux*. Jadi jangan kaget jika diluar sana ada banyak sekali *Custom Rom* untuk masing-masing perangkat *Android*.
- e. Aplikasi, untuk aplikasi akan disajikan jutaan pilihan aplikasi yang menarik dari yang gratis hingga berbayar, dan bisa mendownloadnya di *Google Play*.

c) Kekurangan Android

- a. *Update System*, untuk melakukan update system tidaklah mudah, anda harus menunggu dari masing-masing Vendor untuk merilis *Update Versi* yang terbaru.
- b. Baterai Cepat Habis, Ini sering terjadi jika menyalakan paket data dan menggunakan *widget* serta aplikasi yang berjalan secara berlebihan, untuk mengatasinya harus mengurangi aktivitas aplikasi pada *smartphone*.
- c. Lambat atau *Lag*, ada kaitannya dengan spesifikasi dari masing-masing perangkat, namun ada kalanya *Android* ini tidak bersahabat dengan beberapa aplikasi dikarenakan *RAM* ataupun Prosesornya yang kurang memadai, jadi sesuaikan aplikasi dengan perangkatnya.

d) Android Versi 4.0

Dalam pembuatan aplikasi ini, *Android* yang digunakan adalah *Android* versi 4.0 *Ice Cream Sandwich* atau disingkat *ICS* adalah *Android* pertama yang mempunyai fitur baru membuka kunci dengan pengenalan wajah. *ICS* juga mempunyai penampilan *interface* yang bersih dan *smooth*. *ICS* mempunyai font yang bernama *Roboto*. Jika digunakan *font* ini terlihat sederhana namun elegan, tampilannya pun unik.

Adapun kelebihan dari *Android versi 4.0 Ice Cream Sandwich* ini dari segi *Browser*. Perubahan yang cukup signifikan terletak pada tampilan *browser*. *Browser* ini mampu membuka hingga 16 *tab*, tampilannya pun tidak

berbentuk horizontal pada umumnya, namun berbentuk vertikal dan bertumpuk. Pada versi *Android 4.0* atau *Ice Cream Sandwich* ini akan mempermudah ke tampilan *web desktop* hanya dengan memilih *option full screen*.

2.2.4 Eclipse

Eclipse adalah *IDE (Integared Development Environtment)* untuk pengembangan *java/Android* yang *free* dan dapat didownload di [*http://www.eclipse.org/downloads/*](http://www.eclipse.org/downloads/). *IDE* merupakan program komputer yang memiliki beberapa fasilitas yang diperlukan dalam pembangunan perangkat lunak. *Eclipse* memiliki *Androidplugin* lengkap yang tersedia untuk mengembangkan aplikasi *Android*. Selain itu *Eclipse* juga mendapat dukungan dari *Google* untuk menjadi *IDE* pengembangan *Android* untuk membuat *coding project Android* dimana *source software* langsung dari situs resminya. *Eclipse* dikembangkan dengan bahasa pemrograman *Java*, akan tetapi mendukung dengan bahasa pemrograman aplikasi berbasis *C/C++*, *Cobol*, *Phyton*, *Perl*, *PHP*, dan lain sebagainya.

2.2.5 Pengertian Database Mysql dan Phpmyadmin

Database adalah sebuah system yang dibuat untuk mengorganisasi, menyimpan dan menarik data dengan mudah. *Database* terdiri dari kumpulan data yang terorganisir untuk 1 atau lebih penggunaan dalam bentuk digital. *Database digital* di atur menggunakan *Database Management System (DBMS)*, yang menyimpan isi *database*, mengizinkan pembuatan data dan pencarian serta akses lain. Beberapa *database* yang ada saat ini adalah: *mysql*, *Sql Server*, *MS. Access*, *Oracle*, dan *PostgreSql*.

a) Pengertian Mysql

Menurut Didik Dwi Prasetyo (2004 :18) *MySQL* merupakan salah satu *database server* yang berkembang di lingkungan *open source* dan didistribusikan secara *free* (gratis) dibawah lisensi *GPL*. *MySQL* merupakan *RDBMS (Relational Database Management System)server*. *RDBMS* adalah program yang memungkinkan pengguna database untuk membuat, mengelola, dan menggunakan data pada suatu model relasional. Dengan demikian, tabel-tabel yang ada pada database memiliki relasi antara satu tabel dengan tabel lainnya.

Beberapa keunggulan dari *MySQL* yaitu :

- a. Cepat, handal dan mudah dalam penggunaannya *MySQL* lebih cepat tiga sampai empat kali dari pada *database server* komersial yang beredar saat ini, mudah diatur dan tidak memerlukan seseorang yang ahli untuk mengatur administrasi pemasangan *MySQL*.
- b. Didukung oleh berbagai bahasa, *database server MySQL* dapat memberikan pesan *error* dalam berbagai bahasa seperti Belanda, Portugis, Spanyol, Inggris, Perancis, Jerman, dan Italia.
- c. Mampu membuat tabel berukuran sangat besar 24, ukuran maksimal dari setiap tabel yang dapat dibuat dengan *MySQL* adalah 4 GB sampai dengan ukuran *file* yang dapat ditangani oleh sistem operasi yang dipakai.
- d. Lebih murah, *MySQL* bersifat *open source* dan didistribusikan dengan gratis tanpa biaya untuk *UNIX platform*, *OS/2* dan *Windows platform*.
- e. Melekatnya integrasi *PHP* dengan *MySQL*, keterikatan antara *PHP* dengan *MySQL* yang sama-sama *software opensource* sangat kuat, sehingga koneksi yang terjadi lebih cepat jika dibandingkan dengan menggunakan *database server* lainnya. Modul *MySQL* di *PHP* telah dibuat *built-in* sehingga tidak memerlukan konfigurasi tambahan pada *file* konfigurasi *php* ini.

b) Pengertian *Phpmyadmin*

Phpmyadmin adalah sebuah aplikasi yang *open source* yang berfungsi untuk memudahkan manajemen *MySQL*. Dengan menggunakan *phpmyadmin*, dapat membuat *database*, *table*, *insert*, menghapus, meng-*update* data dengan *GUI* dan terasa lebih mudah, tanpa perlu mengetikkan perintah *SQL* secara manual.

2.2.6 *Html5*

HTML5 merupakan sebuah bahasa markah untuk menstrukturkan dan menampilkan isi dari *World Wide Web*, sebuah teknologi inti dari internet. *HTML5* merupakan hasil proyek dari *W3C (World Wide Web Consortium* dan *WHATWG (Web Hypertext Application Technology Working Group)*). Dimana *WHATWG* bekerja dengan bentuk *web* dan aplikasi dan *W3C* merupakan pengembang dari

XHTML 2.0 pada tahun 2006, kemudian mereka memutuskan untuk bekerja sama dan membentuk versi baru dari HTML. Tujuan dibuatnya HTML5 yaitu:

- a. Fitur baru harus didasarkan pada *HTML*, *CSS*, *DOM*, dan *JavaScript*.
- b. Mengurangi kebutuhan untuk *plugin* eksternal (Seperti *Flash*).
- c. Penanganan kesalahan yang lebih baik.
- d. Lebih markup untuk menggantikan *scripting*.
- e. *HTML5* merupakan perangkat mandiri.
- f. Proses pembangunan dapat terlihat untuk umum.

2.2.7 PhoneGaps

PhoneGap adalah sebuah kerangka kerja/framework *open source* yang dipakai untuk membuat aplikasi *cross-platform mobile* dengan *HTML5*, *CSS*, dan *JavaScript*. *PhoneGap* menjadi suatu solusi yang ideal untuk seorang *web developer* yang tertarik dalam pembuatan aplikasi di *smartphone*. dengan framework *phonegap* dengan hanya melakukan satu kali *coding* langsung bisa di *compile* kesemua *platform* sekaligus.

2.2.8 JSON

JSON (Javascript Object Notation). *JSON* dipakai untuk dua hal yaitu untuk format *transfer* data antar *device/OS/bahasa* pemrograman yang berbeda dan sebagai media penyimpanan data seperti *MongoDB*. dalam layanan *Web API* atau *webservice*, sebagian layanan menggunakan *JSON*, sebagian yang lain menggunakan format *XML*. Dua format ini bisa saling menggantikan.

a) Kelebihan JSON

Adapun kelebihan dari *JSON* antara lain :

- a. Ukuran lebih kecil dibanding *XML*, efeknya *transfer* data lebih cepat dan lebih hemat *resource*, terutama *bandwidth*.
- b. *JSON* adalah format data bawaan di *Javascript*, artinya jika data dari server di kirim ke *client* , dan *client* menggunakan *Javascript*, maka tidak perlu *library* tambahan untuk memprosesnya.
- c. Dibanding *XML*, format *JSON* lebih sederhana.
- d. *Library JSON* ada di setiap bahasa pemrograman sehingga memudahkan *programer* yang berbeda bahasa pemrograman.

b) Definisi Cara Kerja JSON

Secara garis besar *JSON* merupakan sebuah objek data yang dapat dipanggil dan ditampilkan oleh sebuah program, baik itu oleh *PHP*, *JavaScript*, *C#*, *Java*, *Objective-C*, dan bahasa pemrograman lain yang sudah mendukung dan memiliki *JSON Library*. Selain itu, peran *JSON* bukan hanya memanggil dan menampilkan data saja, *JSON* juga bisa untuk memasukan data ke dalam database dan menampilkannya secara dinamis dalam suatu program. Objek yang ada pada *JSON* secara keseluruhan merupakan array, dimana dari setiap objek tersebut *JSON* merubah setiap tipe data lalu menampilkannya dalam jenis *string* agar mudah terbaca oleh program. Objek pada *JSON* memiliki *identifier* (identitas), dimana *identifier* ini nantinya dipanggil oleh program agar menampilkan data dalam jenis string yang bisa dibaca dan dieksekusi oleh program apabila kondisi data memiliki perubahan atau tereksekusi secara *realtime* (*real time object execution*).

c) Implementasi Cara Kerja JSON

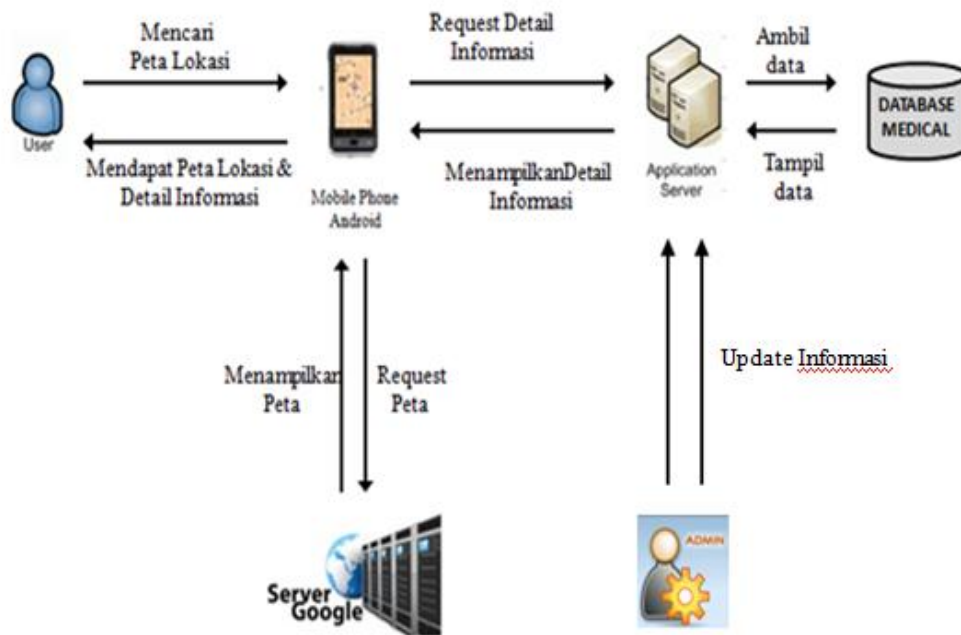
Struktur kode dalam *JSON* memiliki turunan saat menampilkan sebuah data dalam *identifier*, dimana setiap data yang ada dalam *identifier* tersebut sifatnya dinamis dan dapat dieksekusi oleh program berdasarkan *identifier* yang sudah dipilih dan ditentukan untuk ditampilkan dalam program yang dibuat. Setiap *identifier* memiliki sebuah data yang bisa dipanggil oleh program. Setelah data diambil berdasarkan *identifier* yang dipilih, maka data tersebut akan menampilkan output sesuai *identifier* yang sudah ditentukan.

BAB III

ANALISIS DAN PERANCANGAN

3.1 Analisis Sistem

3.1.1 Deskripsi Umum Sistem



Gambar 1. Deskripsi Umum Sistem

Pada gambar 1 diatas menjelaskan bahwa :

- a) Pengguna melakukan pencarian lokasi yang diinginkan pada *mobile phone Android*.
- b) *Mobile Phone Android* akan melakukan *request* peta lokasi ke server *Google Maps*.
- c) *Server Google Maps* akan menampilkan peta lokasi yang dicari oleh Pengguna.
- d) Pengguna kemudian melakukan pencarian informasi berdasarkan parameter yang dipilih.
- e) *Mobile Phone Android* akan melakukan *request* detail informasi ke *server aplikasi*.

- f) Menggunakan mendapat detail informasi yang dibutuhkan dari *mobile phone Android* yang sebelumnya telah melakukan *request* ke *server* aplikasi, dimana data tersebut diambil dari *database* sistem.
- g) Admin bertugas untuk meng-*updated* data informasi.

Aplikasi *Geo-Location Medical* berbasis *Android* ini dirancang khusus untuk pengguna *Mobile Phone* yang memiliki sistem operasi *Android*, dimana aplikasi ini mampu memberikan informasi berdasarkan parameter (lokasi, tempat kesehatan, dan fasilitas kesehatan) yang dimasukkan oleh pengguna *Mobile Phone*. Seluruh informasi yang didapatkan dari aplikasi tersebut berasal dari basis data (*database*) yang terintegrasi dengan aplikasi. Untuk menampilkan peta sendiri berasal dari *GoogleMaps*.

3.1.2 Analisis Kebutuhan Sistem

Analisis kebutuhan sistem menjelaskan tentang kebutuhan yang dibutuhkan dalam merancang aplikasi *geo-location medical* berbasis *Android* ini. Berikut adalah kebutuhan yang digunakan :

a) Spesifikasi Perangkat Keras

Menjelaskan tentang perangkat keras yang digunakan dalam proses perancangan aplikasi ini. Adapun spesifikasinya sebagai berikut :

1. *Laptop Samsung 300PC spesifikasi Intel(R) Celeron(R) CPU 847 @1.10GHz 1.10 GHz, RAM 2,00 GB.*
2. *Mobile Phone Android 4.1 Oppo*

b) Spesifikasi Perangkat Lunak

Menjelaskan tentang perangkat lunak yang digunakan dalam proses perancangan aplikasi ini. Adapun spesifikasinya sebagai berikut :

1. *Eclipse*, sebagai media pembuatan aplikasi.
2. *Android 4.0* sebagai sistem operasi pada *mobile phone*.
3. *Windows 7*, sebagai sistem operasi pada *PC*.

4. *Visual Paradigm for UML CE 10.1*, sebagai perancangan diagram aplikasi.
5. *Notepad ++*, sebagai perangkat lunak pendukung.

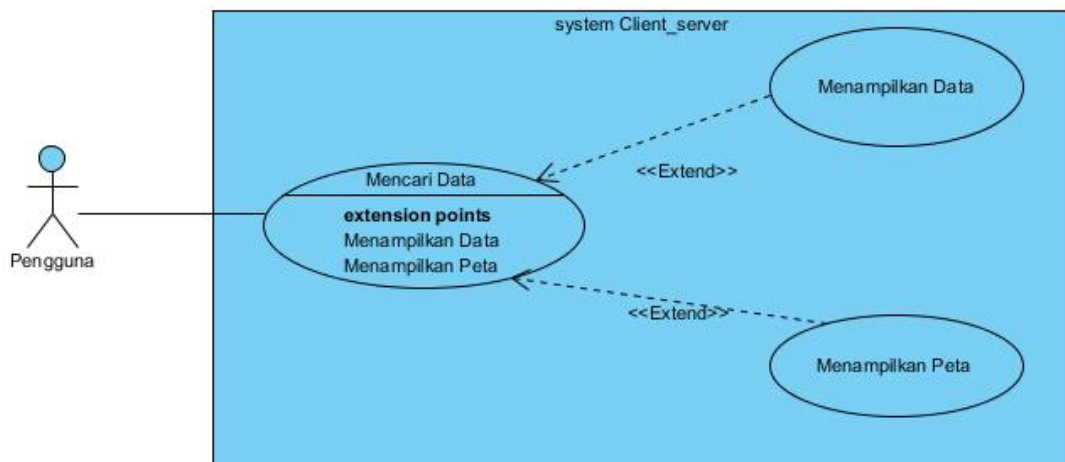
3.1.3 Analisis Kebutuhan Fungsional

1. Aplikasi mampu menampilkan peta lokasi pusat kesehatan di Batam melalui *Mobile Phone*.
2. Aplikasi menampilkan pilihan fasilitas kesehatan di Batam melalui *Mobile Phone*.
3. Aplikasi informasi fasilitas kesehatan di Batam melalui *Mobile Phone*.

3.2 Perancangan Sistem

Perancangan sistem secara umum merupakan tahap perancangan secara rinci yang akan diterapkan. Adapun tujuan perancangan sistem ini adalah untuk memberikan gambaran umum kepada pengguna tentang sistem yang akan dibuat.

3.2.1 Diagram Use Case



Gambar 2. *Use Case Diagram*

Dari gambar 2 menjelaskan bahwa aplikasi ini memiliki satu *actor* yaitu pengguna, dimana pengguna dapat langsung masuk ke *interface* utama aplikasi ini tanpa adanya proses *login* terlebih dahulu, pengguna bisa langsung mencari data berdasarkan parameter yang sudah ditampilkan sistem. Kemudian pengguna dapat mencari lokasi peta sesuai dengan lokasi yang diinginkan.

3.2.2 Skenario Use Case

Skenario *use case* untuk perancangan aplikasi ini adalah sebagai berikut :

a) Skenario Use Case Mencari Data

Actor	: Pengguna
Kondisi awal	: Pengguna telah masuk halaman utama aplikasi
Kondisi akhir	: Pengguna berhasil menemukan data fasilitas kesehatan
Skenario	: 1. Memilih menu Pencarian 2. Memilih parameter yang tersedia 3. Data fasilitas kesehatan hasil pencarian ditampilkan
Skenario alternatif	: Jika parameter yang dimasukkan Pengguna tidak ada pada <i>database</i> maka ditampilkan layar kosong.

b) Skenario Menampilkan Data

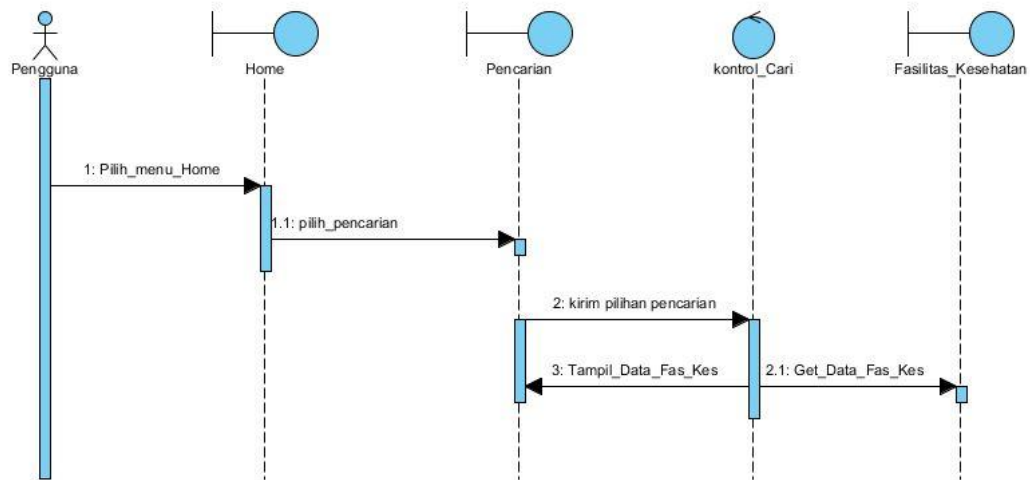
Actor	: Pengguna
Kondisi awal	: Pengguna telah mencari data
Kondisi akhir	: Aplikasi menampilkan data fasilitas kesehatan
Skenario	: 1. Memilih menu Pencarian 2. Memilih parameter yang tersedia 3. Data fasilitas kesehatan hasil pencarian ditampilkan

c) Skenario Menampilkan Peta

Actor	: Pengguna
Kondisi awal	: Pengguna telah mencari peta lokasi
Kondisi akhir	: Aplikasi menampilkan peta lokasi fasilitas kesehatan
Skenario	: 1. Memilih menu Pencarian 2. Memilih parameter yang tersedia 3. Peta Lokasi fasilitas kesehatan hasil pencarian ditampilkan

3.3 Sequence Diagram

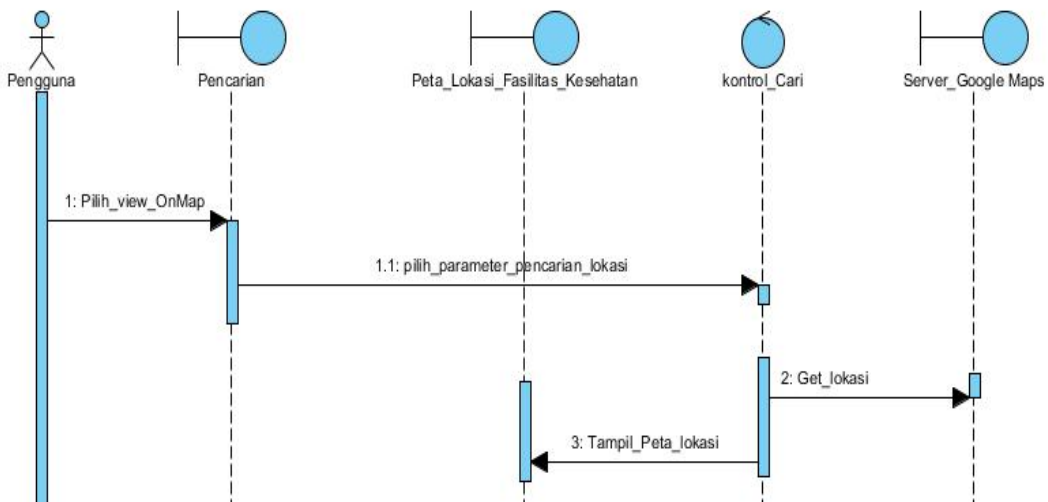
3.3.1 Sequence Diagram MenampilkanData



Gambar 3. Sequence Diagram Menampilkan Data

Pada gambar 3 di atas, dimulai dari pengguna membuka aplikasi, dan mulai mencari data, kemudian aplikasi akan menampilkan data yang dicari sesuai dengan parameter yang dipilih pengguna.

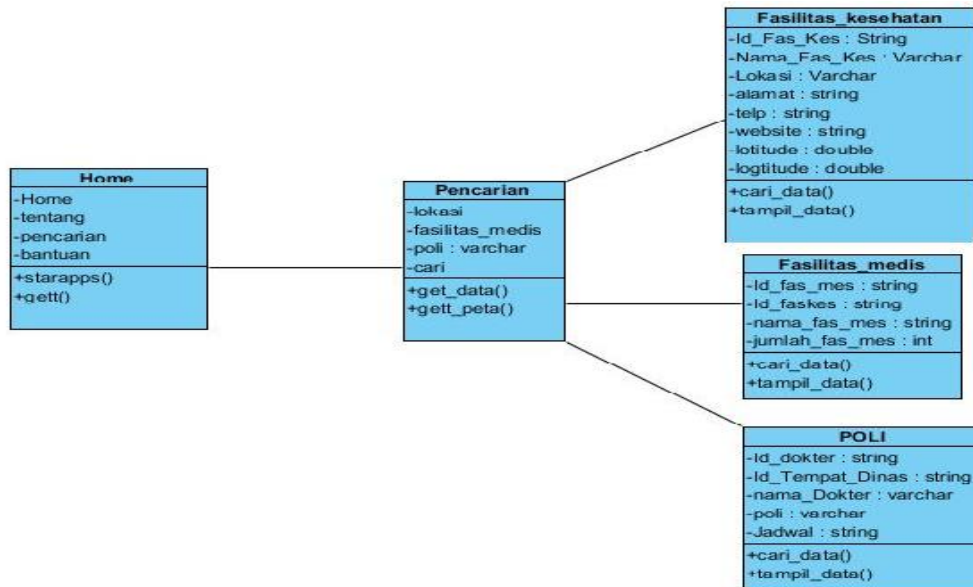
3.3.2 Sequence Diagram Menampilkan Peta



Gambar 4. Sequence Diagram Menampilkan Peta

Pada gambar 4, menjelaskan bahwa setelah aplikasi meminta lokasi peta yang diinginkan dari server aplikasi maka server aplikasi berhasil menampilkan peta lokasi fasilitas kesehatan yang diinginkan pengguna.

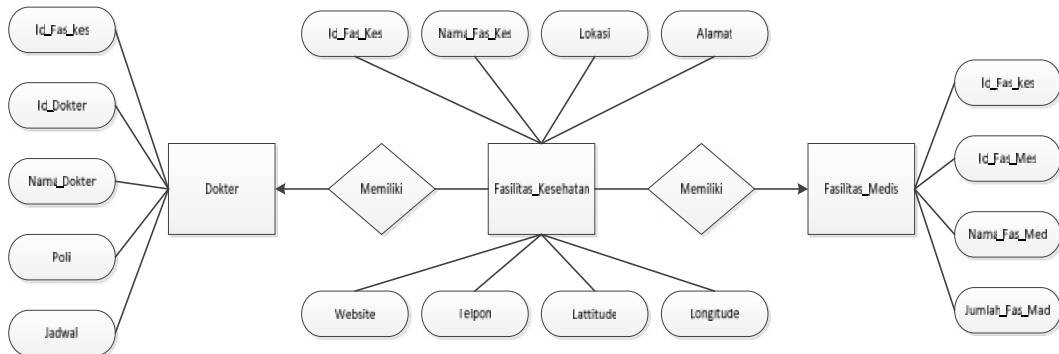
3.4 Class Diagram



Gambar 5. Class Diagram

Keterangan : pada sistem ini terdapat beberapa kelas *interface*. Home merupakan tampilan halaman utama sistem aplikasi ini. *Class* pencarian digunakan untuk melakukan pencarian data fasilitas kesehatan. Pada *class* pencarian terdapat *class* Fasilitas_kesehatan, *class* Fasilitas_Medis, dan *class* Poli.

3.5 ER Diagram

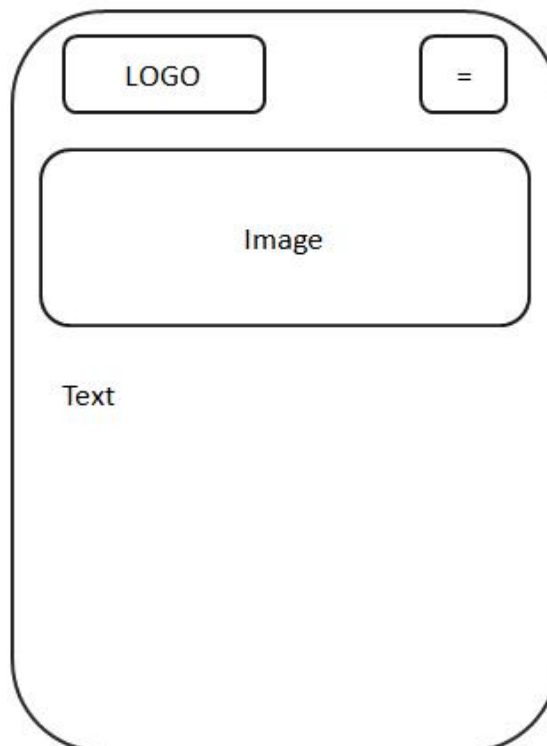


Gambar 6.ER Diagram

Pada Gambar diatas dijelaskan bahwa setiap Fasilitas Kesehatan (terdiri dari Id Fas_kes, Nama_Fas_kes, Website, alamat, lokasi, latitude, dan longitude) memiliki satu Fasilitas Medis (terdiri dari Id_fas_medis, Id_fas_kes, Nama_fas_medis, dan Jumlah fas_medis) dan memilikidokter (terdiri dari Id_dokter, Id_Fas_kes, serta jadwal).

3.6 Perancangan Antarmuka

3.6.1 Perancangan Antarmuka Halaman Utama Home



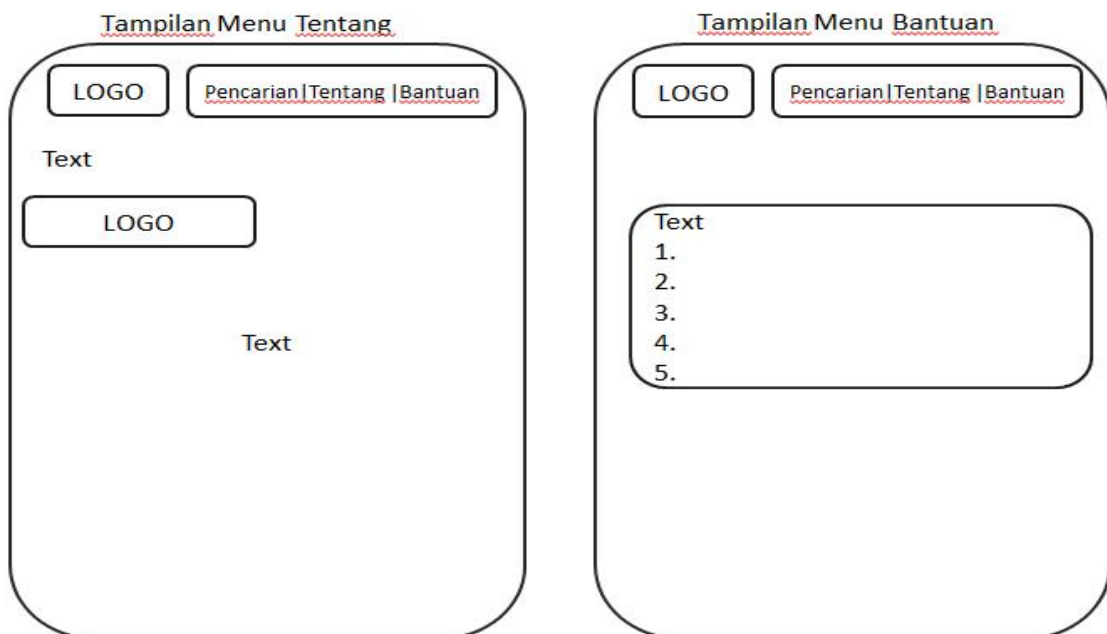
Gambar 7. Perancangan Antarmuka Home

Pada antarmuka halaman awal aplikasi ini terdapat beberapa gambaran mengenai logo aplikasi dan *button* penghubung ke halaman-halaman berikutnya.

Tabel 2.Deskripsi Halaman Menu Utama

Id	Jenis	Nama	Keterangan
Btn Text	Button	Home	Tampilan Tulisan “GEO-LOCATION MEDICAL”
Btn =	Button	Menu	Membuka Halaman Berikutya (Pencarian, Tentang, dan About)
Image view	Image	Image	Image Home
Text Field	Text	Text	Tampilan kata pengantar halaman utama.

3.6.2 Perancangan Antarmuka Halaman Menu Tentang dan Bantuan



Gambar 8. Perancangan Antarmuka Menu Tentang dan Bantuan

Pada gambar 8 diatas menjelaskan bahwa , pada tampilan menu Tentang akan tampil informasi tentang definisi aplikasi, sedangkan pada tampilan menu Bantuan, akan tampil informasi cara penggunaan aplikasi yang akan memudahkan *user* menggunakan aplikasi ini.

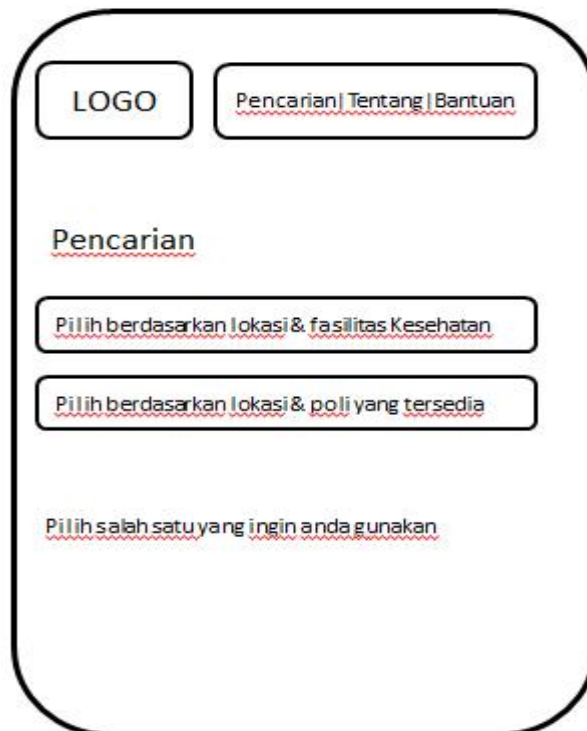
Tabel 3.Deskripsi Halaman Menu Tentang

Id	Jenis	Nama	Keterangan
Btn Logo	Button	Logo	Tampilan Tulisan “GEO LOCATION MEDICAL” untuk kembali ke Home
Btn Pencarian	Button	Pencarian	Membuka Halaman ke menu pencarian
Btn Tentang	Button	Tentang	Membuka Halaman ke menu Tentang
Btn Bantuan	Button	Bantuan	Membuka Halaman ke menu Bantuan
Text Field	Text	Text	Menampilkan penjelasan tentang aplikasi GEO LOCATION MEDICAL”

Tabel 4.Deskripsi Halaman Menu Tentang

Id	Jenis	Nama	Keterangan
Btn Logo	Button	Logo	Tampilan Tulisan “GEO LOCATION MEDICAL” untuk kembali ke Home
Btn Pencarian	Button	Pencarian	Membuka Halaman ke menu pencarian
Btn Tentang	Button	Tentang	Membuka Halaman ke menu Tentang
Btn Bantuan	Button	Bantuan	Membuka Halaman ke menu Bantuan
Text Field	Text	Text	Menampilkan penjelasan penggunaan aplikasi GEO LOCATION MEDICAL”

3.6.3 Perancangan Antarmuka Halaman Menu Pencarian



Gambar 9. Perancangan Antarmuka Menu Pencarian

Pada gambar 9 menjelaskan bahwa pengguna mulai dapat mencari informasi berdasarkan pilihan parameter yang telah disediakan kemudian memilih salah satu pencarian sesuai keinginan pengguna ada dua pilihan bantuan pencarian informasi, yang pertama pilih berdasarkan lokasi dan fasilitas kesehatan dan yang kedua pilih berdasarkan lokasi dan poli yang tersedia.

Tabel 5.Deskripsi Halaman Menu Pencarian

Id	Jenis	Nama	Keterangan
Btn Logo	Button	Logo	Tampilan Tulisan “GEO LOCATION MEDICAL” untuk kembali ke Home.
Btn Pencarian	Button	Pencarian	Membuka Halaman ke menu pencarian.
Btn Tentang	Button	Tentang	Membuka Halaman ke menu Tentang.
Btn Bantuan	Button	Bantuan	Membuka Halaman ke menu Bantuan.
Btn Lokasi & Fasilitas Kesehatan	Button	Lokasi & fasilitas Kesehatan	Pilihan berdasarkan lokasi & fasilitas kesehatan yang dicari
Btn Lokasi & Poli	Button	Lokasi & poli	Pilihan berdasarkan lokasi & poli yang dicari

3.6.3.1 Perancangan Antarmuka Halaman Menu Pencarian 1



Gambar 10. Perancangan Antarmuka Menu Pencarian 1

Tabel 6.Deskripsi Halaman Menu Pencarian 1

Id	Jenis	Nama	Keterangan
Btn Logo	Button	Logo	Tampilan Tulisan “GEO LOCATION MEDICAL” untuk kembali ke Home.
Btn Pencarian	Button	Pencarian	Membuka Halaman ke menu pencarian.
Btn Tentang	Button	Tentang	Membuka Halaman ke menu Tentang.
Btn Bantuan	Button	Bantuan	Membuka Halaman ke menu Bantuan.
Combo Box Pilih Lokasi	ComboBox	Lokasi	Pilihan berdasarkan lokasi yang dicari
Combo Box Pilih Fas Kesehatan	ComboBox	Fas Kesehatan	Pilihan berdasarkan Fasilitas Kesehatan yang tersedia
Btn Cari	Button	Cari	Menampilkan Informasi fasilitas kesehatan serta menampilkan peta lokasi fasilitas kesehatan.

3.6.3.2 Perancangan AntarmukaHalaman Menu Pencarian 2



Gambar 11. Perancangan Antarmuka Menu Pencarian 2

Tabel 7.Deskripsi Halaman Menu Pencarian 2

Id	Jenis	Nama	Keterangan
Btn Logo	Button	Logo	Tampilan Tulisan “GEO LOCATION MEDICAL” untuk kembali ke Home.
Btn Pencarian	Button	Pencarian	Membuka Halaman ke menu pencarian.
Btn Tentang	Button	Tentang	Membuka Halaman ke menu Tentang.
Btn Bantuan	Button	Bantuan	Membuka Halaman ke menu Bantuan.
Combo Box Pilih Lokasi	ComboBox	Lokasi	Pilihan berdasarkan lokasi dicari
Combo Box Pilih Poli	ComboBox	Poli	Pilihan berdasarkan poli yang tersedia
Btn Cari	Button	Cari	Menampilkan Informasi Poli yang tersedia serta menampilkan peta lokasi

BAB IV

HASIL DAN PEMBAHASAN

4.1 Hasil Implementasi

4.1.1 Implementasi Database

4.1.1.1 Tabel fasilitas_kesehatan

Tabel dibawah ini merupakan database untuk fasilitas kesehatan.

Tabel 6. Tabel fasilitas_kesehatan

No.	Nama Field	Jenis	Volume	Keterangan
1.	Id fasilitas kesehatan	integer	10	Primary key
2.	nama fasilitas	varchar	100	
3.	lokasi	varchar	50	
4.	alamat	varchar	100	
5.	telpon	string	15	
6.	website	varchar	100	
7.	Latitude	Double		

4.1.1.2 Tabel Dokter

Tabel dibawah ini merupakan database untuk dokter

Tabel 7. Tabel Dokter

No.	Nama Field	Jenis	Volume	Keterangan
1.	Id dokter	integer	10	Primary key
2.	Id_Fasilitas kesehatan	integer	10	
3.	nama dokter	varchar	255	
4.	Poli	varchar	255	
5.	jadwal	varchar	255	

4.1.1.3 Tabel Fasilitas Medis

Tabel dibawah ini merupakan database untuk fasilitas_medis

Tabel 8. Tabel Fasilitas Medis

No.	Nama Field	Jenis	Volume	Keterangan
1.	Id faseilitas medis	integer	10	Primary key
2.	Id fas_kes	integer	10	
3.	Nama fas_mes	varchar	255	
4.	Jumlah fas_mes	integer	100	

4.2 Implementasi Class

Berdasarkan pada perancangan yang telah dilakukan, maka hasil implementasi kelas dan antarmuka yang dibuat secara detail dari perancangan yang telah dilakukan, selanjutnya pada implementasi yang menghasilkan kelas-kelas yang saling terhubung antara controller, model dan view. Perbedaan antara perancangan dan implementasi dapat dilihat pada table berikut :

Tabel 9. Tabel Implementasi Class

Nama Kelas Perancangan	Nama File Implementasi
Home	Index.html
Pencarian	Pencarian.html
Tentang	About.html
Bantuan	Help.html

4.3 Implementasi Antarmuka

Pada antarmuka halaman awal aplikasi ini terdapat beberapa gambaran mengenai logo aplikasi dan menu penghubung ke halaman – halaman selanjutnya seperti Pencarian, Tentang, Bantuan. Berikut gambar halaman utama aplikasi :



Gambar 10. Halaman Utama Aplikasi

Setelah pengguna dihadapkan pada antarmuka halaman utama aplikasi ini, pengguna dapat memilih *button* pencarian. Pada halaman ini digunakan untuk melakukan pencarian fasilitas kesehatan berdasarkan parameter yang dimasukkan oleh pengguna. Berikut adalah rancangan antarmuka yang akan muncul ketika pengguna memilih menu Pencarian pada aplikasi ini.

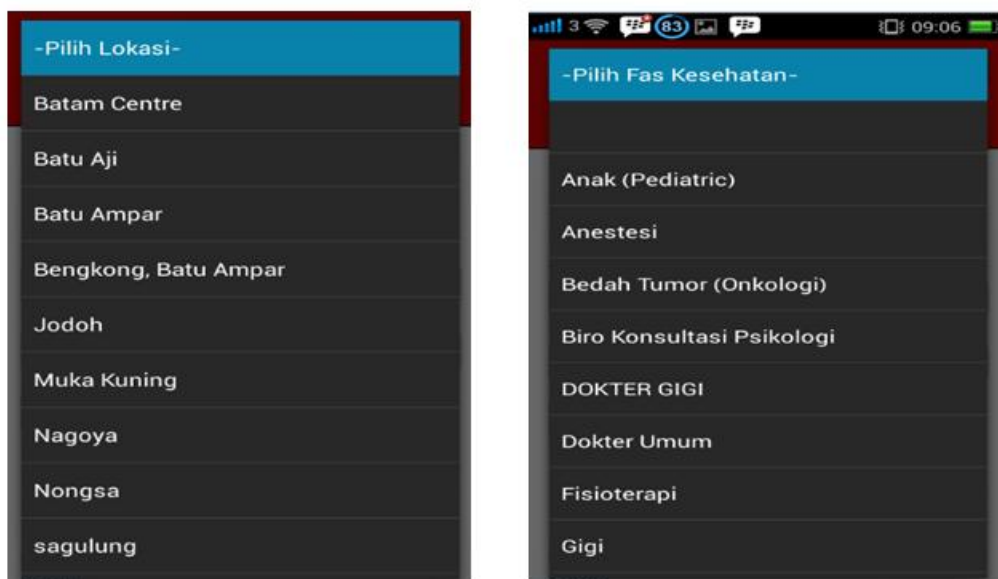


Gambar 11. Halaman Menu Pencarian

Pada halaman menu pencarian terdapat dua pilihan menu untuk mencari informasi berdasarkan pilihan yang tersedia pada aplikasi. Berikut adalah tampilan antarmuka yang akan muncul ketika pengguna memilih menu Pencarian 1 atau pencarian 2 pada aplikasi ini.



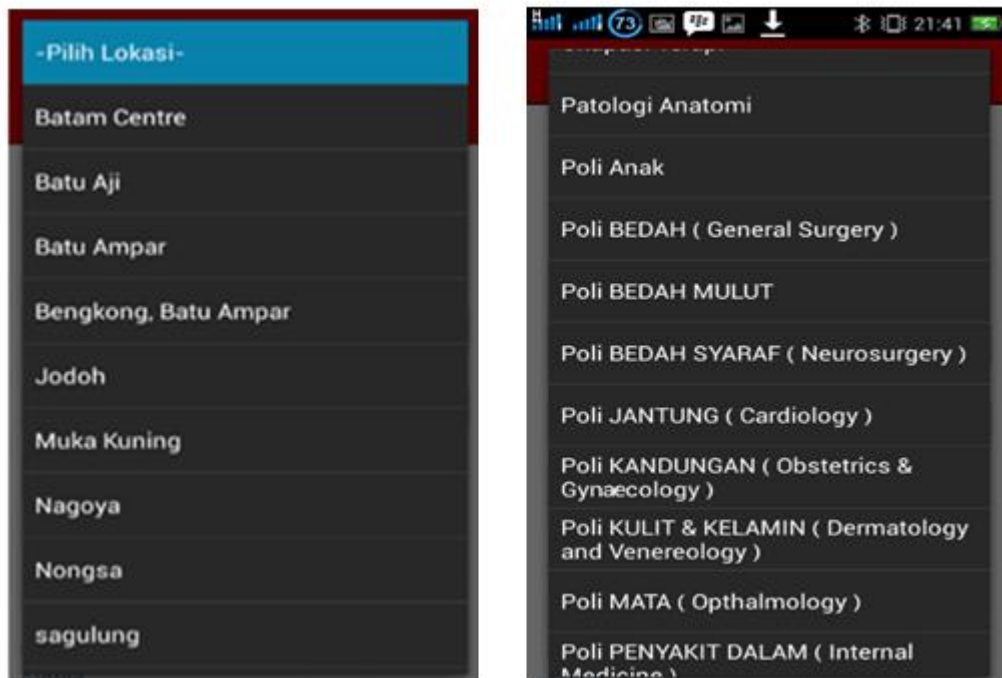
Gambar 12. Halaman Menu Pencarian 1



Gambar 13. Tampilan Pilihan Berdasarkan Lokasi Dan Fas. Kesehatan

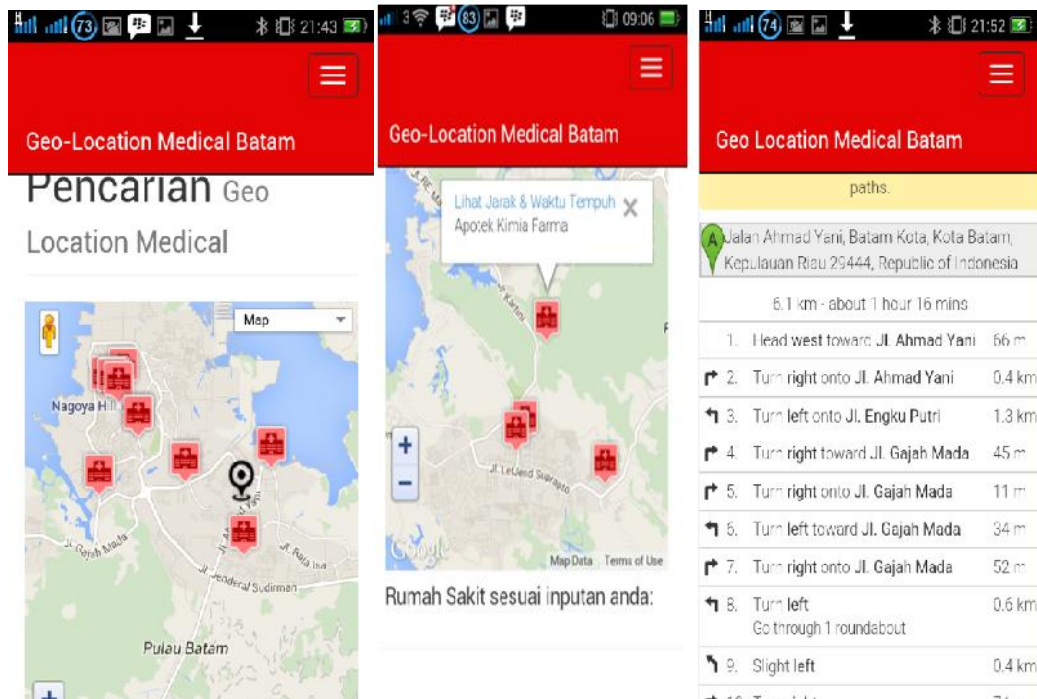


Gambar 14. Halaman Menu Pencarian 2



Gambar 15. Tampilan Pilihan Berdasarkan Lokasi Dan Poli

Selanjutnya adalah halaman antarmuka peta lokasi yang dicari pengguna pada halaman Pencarian. Tampilan yang akan muncul ketika pengguna menekan *button* cari pada halaman menu pencarian adalah sebagai berikut :



Gambar 16. Tampilan Peta Lokasi

Halaman menu Tentang pada aplikasi ini digunakan untuk mengetahui definisi dari aplikasi yang dibuat. Berikut adalah tampilan antarmuka yang akan muncul ketika pengguna memilih menu *button* Tentang pada aplikasi ini.



Gambar 17. Halaman Menu Tentang

Pada halaman Bantuan digunakan untuk mengetahui cara penggunaan dari aplikasi yang dibuat. Berikut adalah rancangan antarmuka yang akan muncul ketika pengguna memilih *button* Bantuan pada aplikasi ini.



Gambar 18. Halaman Menu Bantuan

4.4 Pengujian dan Pembahasan

4.4.1 Strategi Pengujian

Strategi pengujian pada aplikasi “Aplikasi *Geo-Location Medical* Berbasis *Android*” menggunakan metode *Black Box* yaitu berfokus pada kebutuhan deskripsi fungsional guna untuk mengetahui jalannya sistem secara lengkap serta *Apache Jmeter* yang digunakan untuk melakukan simulasi trafik yang tinggi pada server maupun network selain itu *Apache Jmeter* juga dapat membuat analisis grafis dari kinerja atau untuk menguji server /script /perilaku objek di bawah beban berat bersamaan.

4.4.2 Deskripsi Pengujian

Deskripsi pengujian dalam aplikasi ini adalah sebagai berikut:

1. Melakukan pencarian lokasi fasilitas kesehatan, fasilitas medis, dan poli
2. Menampilkan informasi fasilitas kesehatan dari hasil pencarian
3. Menampilkan peta lokasi fasilitas kesehatan dari hasil pencarian

4.4.3 Hasil Pengujian

Pengujian aplikasi telah dilaksanakan dengan baik sesuai dengan metode *Black Box* yang digunakan dan hasilnya dapat dilihat pada tabel berikut ini:

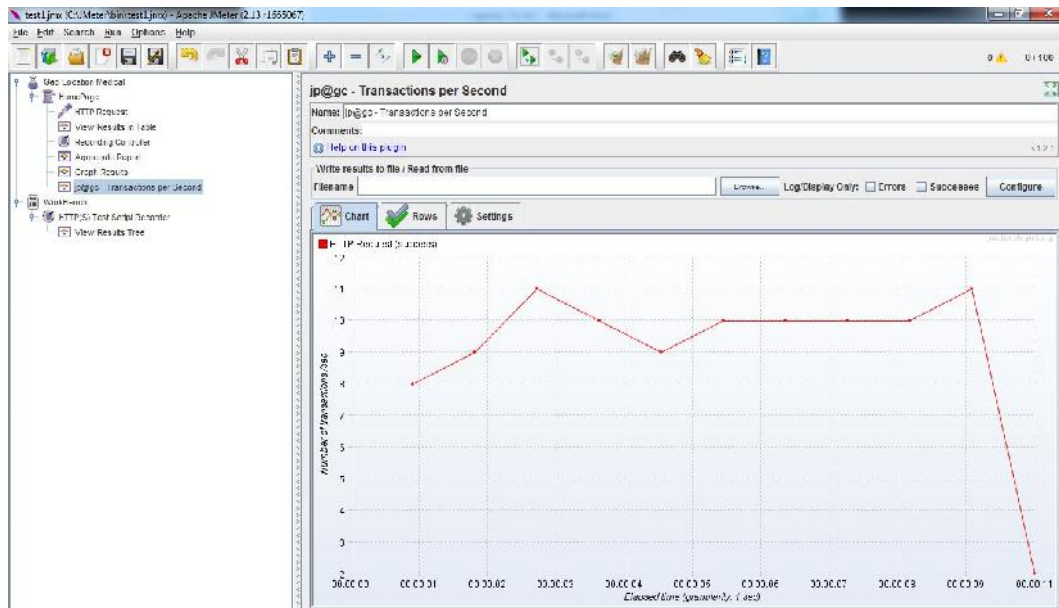
a. Pengujian Apache Jmeter

The screenshot shows the Apache Jmeter interface with the 'View Results in Table' window open. The table displays the following data:

Sample #	Start Time	Thread Name	Label	Sample Time (ms)	Status	Bytes	Latency	Connect Time (ms)
1	15:52:02.443	HomePoco 1-43	HTTP Request	13387	Success	1167	3442	824
2	15:52:02.445	HomePoco 1-24	HTTP Request	13316	Success	1167	3442	735
3	15:52:02.447	HomePoco 1-131	HTTP Request	7369	Success	1167	3427	6
4	15:52:04.269	HomePoco 1-73	HTTP Request	3501	Success	1167	3464	2
5	15:52:05.061	HomePoco 1-63	HTTP Request	7746	Success	1167	3679	3
6	15:52:02.445	HomePoco 1-43	HTTP Request	13349	Success	1167	3425	872
7	15:52:02.445	HomePoco 1-33	HTTP Request	13445	Success	1167	3472	072
8	15:52:02.451	HomePoco 1-23	HTTP Request	13545	Success	1167	3467	738
9	15:52:02.440	HomePoco 1-41	HTTP Request	13575	Success	1167	3553	813
10	15:52:02.609	HomePoco 1-72	HTTP Request	3138	Success	1167	3615	10
11	15:52:02.448	HomePoco 1-13	HTTP Request	13661	Success	1167	3530	874
12	15:52:02.541	HomePoco 1-63	HTTP Request	3517	Success	1167	7200	33
13	15:52:02.447	HomePoco 1-17	HTTP Request	13725	Success	1167	3703	877
14	15:52:02.447	HomePoco 1-27	HTTP Request	13770	Success	1167	3610	753
15	15:52:02.442	HomePoco 1-32	HTTP Request	13959	Success	1167	3349	812
16	15:52:02.147	HomePoco 1-33	HTTP Request	11316	Success	1167	3509	808
17	15:52:02.451	HomePoco 1-42	HTTP Request	11347	Success	1167	3541	739
18	15:52:02.443	HomePoco 1-55	HTTP Request	11204	Success	1167	3351	731
19	15:52:02.453	HomePoco 1-67	HTTP Request	7247	Success	1167	4471	4
20	15:52:04.702	HomePoco 1-83	HTTP Request	9354	Success	1167	3105	2
21	15:52:02.453	HomePoco 1-65	HTTP Request	11560	Success	1167	3270	801
22	15:52:02.163	HomePoco 1-83	HTTP Request	3367	Success	1167	5763	23
23	15:52:05.500	HomePoco 1-63	HTTP Request	3722	Success	1167	5044	3
24	15:52:02.470	HomePoco 1-63	HTTP Request	11360	Success	1167	3373	877
25	15:52:02.447	HomePoco 1-53	HTTP Request	11906	Success	1167	3370	877
26	15:52:02.145	HomePoco 1-34	HTTP Request	12348	Success	1167	3389	811
27	15:52:03.057	HomePoco 1-64	HTTP Request	11444	Success	1167	7714	236
28	15:52:02.450	HomePoco 1-44	HTTP Request	12100	Success	1167	3345	070
29	15:52:02.477	HomePoco 1-62	HTTP Request	41225	Success	1167	7574	432

Gambar 19. Tampilan Table Pengujian Apache Jmeter

Pada gambar diatas menjelaskan bahwa pengujian ini menggunakan 100 user yang mampu mengakses dalam waktu 10 detik dan pengujian ini berhasil terlihat dari status yang tampil bercentang hijau yang menandakan tidak adanya kegagalan user dalam mengakses.



Gambar 19. Tampilan Grafik Pengujian Apache Jmeter

Pada gambar diatas menjelaskan bahwa pada *JMeter* terdapat juga *listener* yang menerima data sampling hasil pengujian lalu menampilkan dalam bentuk grafik atau kita simpan ke dalam file. Dalam grafik dikatakan bahwa pengujian yang dihasilkan adalah pada saat user mengakses dalam keadaan stabil atau normal dengan tidak adanya kegagalan 100 *user* dalam mengakses dan dalam kurun waktu rata – rata 9,9 per detik. Tidak adanya penurunan drastis dalam pengujian. Akses berhenti di angka 11 karena tidak adanya request yang dilakukan lagi.

Tabel 10. Tabel Hasil Pengujian

No	Use Case	Fungsi	Skenario	Data Uji	Target	Pengujian	Penguji
1	Mencari data	Cari	<ol style="list-style-type: none"> 1. Memilih menu Pencarian 2. Memilih berdasarkan lokasi dan Fasilitas kesehatan 3. Memilih berdasarkan lokasi dan poli yang ada 4. Menampilkan data hasil pencarian 	Nama lokasi : Sei Panas	- Data fasilitas kesehatan ditampilkan di Pencarian	OK	Sudirman
2	Menampilkan data	Data Fasilitas Kesehatan	Memilih pilihan parameter pencarian	Nama industri : RS. Awal bross	Informasi di tampilkan	OK	Sudirman
3	Menampilkan peta	GeoLocation	Mengklik <i>button</i> cari	Latitude, longitude	Peta lokasi ditampilkan	OK	Sudirman

BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Adapun kesimpulan yang dapat di tarik dari Tugas Akhir “Aplikasi *Geo-Location Medical* Berbasis *Android*” ini adalah sebagai berikut:

1. Aplikasi ini dapat membantu para pengguna *Android* dalam melakukan pencarian informasi berupa lokasi, fasilitas kesehatan, serta pelayanan kesehatan lain berdasarkan kebutuhan yang diinginkan pengguna.
2. Selain dapat menampilkan informasi, aplikasi ini juga dapat menampilkan posisi pengguna untuk memilih jalur terdekat menuju lokasi pusat kesehatan sesuai hasil pencarian pengguna.

5.2 Saran

Sebagai langkah pengembangan aplikasi ini pada masa yang akan datang, maka saran yang dapat diberikan adalah:

1. Untuk pengembang selanjutnya diharapkan menangani penentuan rute menuju lokasi fasilitas kesehatan yang dituju.
2. Pengembang nantinya dapat merubah peta lokasi dengan mengembangkan peta yang dapat dibuat secara digitalisasi.
3. Pengembang diharapkan dapat menambahkan *fitur input* data baru oleh *user* atau admin.

DAFTAR PUSTAKA

- Riyanto, membuat sendiri aplikasi mobile GIS: platform JAVA ME, Blackberry & Android, Penerbit Andi Yogyakarta, 2010.
- Prasetio Adhi, 2013, Belajar *HTML 5 Geolocation*, Website :<http://prothelon.com/belajar/belajar-html5/belajar-html5-geolocation-bagian-2> , diakses tanggal 25 September 2014.
- Zilvia Sherly, 2014, Batam Medical Website, Politeknik Negeri Batam, Kepri.
- Neriswandi Dean Tofaneo, 2014, Aplikasi Informasi Lokasi WisataBelanja Batam Berbasis Android, Politeknik Negeri Batam, Kepri.
- Siswanto, 2011, Sistem Informasi GeografisObjek WisataMenggunakan Maps Api, Institut Teknologi Sepuluh Nopember, Surabaya.
- Sevanda Sigit Wima, 2010, Perancangan dan Pembuatan Sistem Informasi Lokasi Berbasis Pys60 Untuk Geo – Location Sharing, Institut Teknologi Sepuluh Nopember, Surabaya.
- Adhi, Reza, 2013, Seminar UKM Komputer PNB, Paper Android, Jimbaran
- Ichtiara C., 2008, Aplikasi Sistem Pemetaan, SIG, dan Google Maps, Universitas Indonesia, Jakarta.
- Clark Richard, Murphy Christopher, 2012, Beginning Responsive Web Design with HTML5 and CSS3, Website: <http://www.teknologi-informasi.net/2014/01/pengertianpenjelasankelebihandan-tujuan.html>, diakses tanggal: 22 Oktober 2014.

LAMPIRAN

1. Analisis Pengukuran Waktu Tempuh, Kecepatan, dan jarak Tempuh Peta

Ada berbagai macam pengukuran mulai pengukuran panjang, berat, waktu, volume (isi) dan lain-lain. Kali ini kita akan membahas tentang : Pengukuran waktu tempuh, Pengukuran jarak tempuh dan Pengukuran kecepatan rata-rata.

1. Pengukuran Jarak Tempuh

Pengukuran waktu tempuh sangat erat kaitannya dengan satuan waktu dan satuan panjang. Seperti contoh dibawah ini :

Seorang mengendarai mobil dengan kecepatan rata-rata 60 km/jam. Ia berangkat dari muka Kuning menuju Politeknik Negeri Batam pukul 19.00. Jika Ia tiba di Politeknik Negeri Batam Pukul 19.12

Berapa Km-kah jarak yang telah ditempuh ?

Jawaban : $s = v \times t$

Rumus jarak tempuh

Waktu yang ditempuh = waktu tiba – waktu berangkat

= Pukul 19.12 – Pukul 19.00

= 15 mnt

Jarak yang ditempuh = kecepatan X waktu

= 60 km/jam X 12 mnt

= 7.5 km

Jadi jarak yang ditempuh adalah 7.5 km

2. Pengukuran Waktu Tempuh

Jarak Muka Kuning – Politeknik Negeri Batam 60 km. Kemudian berangkat ke Politeknik Negeri Batam dengan sepeda motor pukul 19.30

Kecepatan rata-rata 7.5 km/jam.

- b. Berapa lama waktu tempuh yang dibutuhkan untuk sampai ke Politeknik Negeri Batam ?
- b. Pukul Berapakai tiba di politeknik negeri Batam?

Jawaban :

a. Rumus menghitung waktu tempuh = $t = \frac{s}{v}$ atau

= Jarak : kecepatan

= 60 km : 7.5 km/jam

= 12 Mnt

Jadi waktu untuk menempuh jarak Muka Kuning – Politeknik Negeri Batam adalah 12 mnt.

- b. Tiba di Politeknik Negeri Batam Waktu keberangkatan + waktu tempuh

= 19.30 + 12

= 19.42

Jadi tiba di Politeknik Negeri Batam jam 19.42

- c. Pengukuran Kecepatan Rata-Rata

Jarak dari Muka Kuning – Politeknik Negeri Batam 7.5 Km. Jika mengendarai motor berangkat dari Muka Kuning pukul 19.00 dan tiba di Politeknik Negeri Batam pukul 19.12. Berapakah kecepatan rata-rata mengendarai motor ?

Jawaban :

Sebelum menghitung kecepatan rata-rata tentukan waktu yang digunakanyaitu = Tiba – berangkat

= pukul 19.00 – pukul 19.12

= 12 mnt

Jadi waktu yang digunakan adalah 12 mnt

Selanjutnya tentukan kecepatan rata-rata motor.

Rumus kecepatan rata-rata =

$$v = \frac{s_{total}}{t_{total}} = \frac{V_1 \times t_1 + V_2 \times t_2 + \dots + V_n \times t_n}{t_1 + t_2 + \dots + t_n}$$

Atau

= jarak : waktu tempuh

= 7.5 km : 12 mnt

= 40 km/jam

Jadi kecepatan rata-rata 40 km/jam.

Aplikasi coding Menggunakan Google Maps

Google maps yang disertakan pada Android, akan menghubungkan kita dengan Google Maps di internet, tentunya ini menjadi salah satu keunggulan pada platform android. dengan emulator kita bisa mencoba activity Mpas yang disertakan. sedangkan untuk kita sebagai pembuat aplikasi, bisa menggunakan MapView dan Map Activity untuk membuat agar Google Maps bisa kita masukkan ke dalam aplikasi berbasis peta yang kita buat.

Location Based Service.

LBS terdiri dari dua pilar yaitu API Map dan API Location, paket (package) setiap dari API ini berbeda dan terpisahkan. Paket Map berada pada com.google.android.maps dan paket location adalah android.location. API Maps menyediakan fasilitas untuk mendisplay dan memanipulasi peta, seperti zoom, mode peta (satelit view), Street View), atau menambahkan kustom data (overlay) dll. sedangkan API Location adalah berhubungan dengan data GPS (Global Positioning System) dan data lokasi real-time.

Google Map adalah layanan aplikasi dan teknologi peta berbasis web yang disediakan oleh Google secara gratis (bukan untuk kepentingan komersial), termasuk di dalamnya website Google Map (<http://maps.google.com>), Google Ride Finder, Google Transit, dan peta yang dapat disisipkan pada website lain melalui Google Maps API. Saat ini Google Map adalah layanan pemetaan berbasis web yang populer.

Kita dapat menambahkan layanan Google Map ke website kita sendiri menggunakan Google Maps API. Google Maps API dapat ditambahkan ke website kita menggunakan JavaScript. API tersebut menyediakan banyak fasilitas dan utilitas untuk memanipulasi peta dan menambahkan konten ke peta melalui berbagai layanan, memungkinkan Anda untuk membuat aplikasi peta yang kuat pada website Anda.

Pengetahuan yang diperlukan untuk mengembangkan Google Maps API adalah tentang HTML dan JavaScript, sedangkan peta sudah disediakan oleh Google. Jadi kita hanya berkonsentrasi tentang data dan biarkan urusan peta ditangani oleh Google, sehingga dapat menghemat waktu.

Saat penulis membuat artikel ini versi terakhir Google Maps API adalah versi 3. JavaScript API mirip dengan versi sebelumnya. Versi 3 disesain lebih cepat, khususnya untuk browser mobile seperti Android dan iPhone.

Google Maps API Key

Mendaftarkan API Key

Agar peta dapat ditampilkan ke dalam website kita, maka kita harus mempunyai account Google, kemudian mendaftarkan diri dahulu

<https://code.google.com/apis/console/?noredirect>

Gambar Membuat Google Maps API Key

Pada form di atas Anda disuruh memasukkan nama domain yang akan digunakan untuk menampilkan peta dari Google map.

Key yang didapat penulis ketika mendaftarkan domain dengan nama <http://localhost> adalah:

ABQIAAAAJK9VAI7f5EbkTYtTgYmfGRT2yXp_ZAY8_ufC3CFXhHIE1NvwxSiZayCg1x-00Tg2PBrceaMzyZjNQ

Dokumen Sederhana Google Maps

Selain mendapatkan API Key, Anda juga diberikan contoh simple yang dapat menampilkan peta sederhana. Contoh berikut halaman peta yang berpusat di Palo Alto California:

```
<!DOCTYPE html "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"&gt;
<html xmlns="http://www.w3.org/1999/xhtml"&gt;
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8 />
    <title>Google Maps JavaScript API Example</title>
    <script
src="http://maps.google.com/maps?file=api&v=2&key=abcdefg&sensor=true\_or\_false&#8221;
      type="text/javascript"></script>
    <script type="text/javascript">

function initialize() {
  if (GBrowserIsCompatible()) {
    var map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(37.4419, -122.1419), 13);
```

```

        map.setUIToDefault();
    }
}

</script>
</head>
<body onload="initialize()" onunload="GUnload()">
    <div id="map_canvas" style="width: 500px; height: 300px"></div>
</body>
</html>

```

Elemen fundamental dalam aplikasi Google Maps API adalah “peta” itu sendiri. Tulisan ini akan membahas penggunaan fundamental obyek GMap2 dan operasi dasar peta.

Loading Google Maps API Key

Key yang telah Anda peroleh harus dimasukkan ke aplikasi. Key tersebut dimasukkan menggantikan variabel key.

```

<script
src="http://maps.google.com/maps?file=api&v=2&key=&sensor=true\_or\_fals
e&#8221;
        type="text/javascript">

</script>

```

URL <http://maps.google.com/maps?file=api&v=2&key=abcdefg> menunjukkan ke lokasi file JavaScript yang didalamnya termasuk semua simbol definisi yang

diperlukan menggunakan Google Maps API. Halaman Anda harus berisi tag skrip yang menunjuk ke URL ini, menggunakan kunci yang Anda terima ketika mendaftarkan API. Dalam kasus diatas kuncinya adalah “abcdefg”. Setelah key yang didapat diperoleh dan dimasukkan, maka hasilnya adalah sebagai berikut:

```
<script src="http://maps.google.com/maps?file=api&v=2&key=
ABQIAAAAJK9VAI7f5EbkTYtTgYmfGRT2yXp_ZAY8_ufC3CFXhHIE1N
vwkxSiZayCg1x-0OTg2PBrceaMzyZjNQ&sensor=true_or_false"
type="text/javascript">

</script>
```

Catatan: Key di atashanya berlaku untuk domain <http://localhost>

Parameter sensor berisi true digunakan untuk menggunakan sensor lokasi , misalnya GPS (jika ada, misalnya jika aplikasi web based kita diakses dengan ponsel tertentu yang memiliki GPS).

Memulainya Google Maps API

Saat ini akan ditampilkan peta kota tepatnya jalan Malioboro, Yogyakarta. Ukuran peta yang digunakan adalah ukuran 500pixel x 300pixel. Sebelum dimulai kita harus survei dan didapat salah satu titik koordinat jalan Malioboro adalah - 7.794182, 110.365537.

Berikut contoh simple, ada lima hal yang diperhatikan:

1. Kita memasukkan Map API JavaScript menggunakan sebuah tag script.
2. Kita membuat elemen div bernama “map_canvas” untuk menampilkan peta.
3. Kita membuat function JavaScript untuk membuat object “map”.
4. Kita menentukan titik tengah geografis dan setting UI sebagai default konfigurasi.
5. Kita inisialisasi obyek peta dari tag body pada event onLoad.

Beginilah kode program dasarnya:

```
<!DOCTYPE html "-//W3C//DTD XHTML 1.0 Strict//EN"
  "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd"&gt;
<html xmlns="http://www.w3.org/1999/xhtml"&gt;
  <head>
    <meta http-equiv="content-type" content="text/html; charset=utf-8 />
    <title>Google Maps JavaScript API Example</title>

    <script src="http://maps.google.com/maps?file=api&v=2&key=
ABQIAAAAJK9VAI7f5EbkTYtTgYmfGRT2yXp\_ZAY8\_ufC3CFXhHIE1Nvwx
xSiZayCg1x-0OTg2PBceaMzyZjNQ &sensor=true_or_false"
      type="text/javascript"></script>

    <script type="text/javascript">

function initialize() {
  if (GBrowserIsCompatible()) {
    var map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(-7.794182, 110.365537), 13);
    map.setUIToDefault();
  }
}

</script>
</head>
<body onload="initialize()" onunload="GUnload()">
  <div id="map_canvas" style="width: 500px; height: 300px"></div>
</body>
</html>
```

Maps API

Element DOM Map

Perhatikan baris berikut:

```
<div id="map_canvas" style="width: 500px; height: 300px"></div>
```

Dari skrip di atas, peta akan di tempatkan pada elemen div bernama "map_canvas" dengan ukuran lebar 500px dan tinggi 300px.

Peta dapat ditampilkan di sebuah web, kita harus mempersiapkan titik untuk meletakkannya. Biasanya, dibuat elemen div mendapatkan referensi elemen ini pada browser menggunakan *Document Object Model (DOM)*.

Contoh di atas, kita mendefinisikan sebuah div bernama "map_canvas" dan diset ukurannya menggunakan atribut elemen tersebut. Kecuali ukuran peta dapat ditentukan menggunakan GMapOptions pada konstruktor, jika tidak maka akan mengikuti lebar yang disediakan pada elemen div.

Obyek GMap2

Perhatikan skrip berikut:

```
var map = new GMap2(document.getElementById("map_canvas"));
```

Kelas JavaScript yang mewakili peta adalah kelas GMap2. Kelas ini mendefinisikan semua peta tunggal pada sebuah halaman website. Kita dapat membuat lebih dari satu bentuk instan dari kelas ini setiap obyek yang akan mendefinisikan sebuah peta terpisah pada halaman. Kita dapat membuat kelas bentuk instan baru menggunakan operator new.

Ketika kita membuat sebuah map instan baru, kita menentukan sebuah node DOM dalam halaman (biasanya sebuah elemen div) sebagai wadah untuk peta. HTML

node adalah node anak dari JavaScript obyek document dan kita mendapatkan referensi ke elemen melalui method `document.getElementById()`.

Kode ini mendefinisikan sebuah variabel (bernama map) dan mengisi variabel tersebut sebuah obyek baru GMap2. Function GMap2() diketahui adalah sebuah konstruktor dan didefinisikan dari Google Maps API Reference seperti dibawah ini:

Konstruktor	Keterangan
GMap2(container, opts?)	Membuat sebuah peta baru di dalam HTML container, yang biasanya elemen div. Atau anda melewati parameter opsional tipe GmapOptions dalam parameter opts.

Inisialisasi Map

Perhatikan kode berikut:

```
map.setCenter(new GLatLng(-7.794182, 110.365537), 13);  
map.setUIToDefault();
```

Setelah membuat sebuah peta melalui konstruktor GMap2, kita perlu menginisialisasinya. Inisialisasi diselesaikan dengan menggunakan method `setCenter()`. Method memerlukan koordinat GLatLng dan zoom level dan harus dikirm sebelum operasi dilakukan pada peta, termasuk pengaturan atribut lain pada peta itu sendiri.

Selain itu juga memanggil method `setUIToDefault()` pada map. Method tersebut menset antar muka pengguna (input handling dan setting kontrol) menjadi konfigurasi menjadi default, termasuk pan dan kontrol zoom, seleksi tipe peta, dan sebagainya. Setting default UI akan dibahas pada bab selanjutnya.

Loading Map

Perhatikan kode berikut:

```
<body onload="" initialize()" onunload="" GUnload()">
```

Sementara merender sebuah halaman HTML, *document object model (DOM)* dibangun, dan semua gambar dan skrip eksternal diterima dan dimasukkan obyek document. Untuk memastikan bahwa peta ditempatkan pada halaman setelah halaman selesai diload, kita hanya menjalankan fungsi yang di konstruktor obyek GMap2 sekali elemen <body> HTML menerima event onload. Melakukan hal yang dapat menghindari perilaku tak terduga dan memberi kita lebih banyak kontrol tentang bagaimana dan kapan peta digambar.

Atribut onload adalah sebuah contoh *event handler*. Google Maps API juga menyediakan beberapa event yang dapat “listen” untuk menentukan status perubahan. Untuk melihat events dan event listener dapat dilihat pada: <http://code.google.com/apis/maps/documentation/events.html>

Fungsi GUnload() adalah fungsi utilitas yang dirancang untuk mencegah kebocoran memory.

Latitudes and Longitudes

Sekarang kita telah mempunyai sebuah peta, kita membutuhkan lokasi yang mengacu ke lokasi pada peta. Obyek GLatLng menyediakan mekanisme seperti pada Google Maps API. Kita dapat membangun obyek GLatLng, melewati parameternya dalam urutan (latitude, longitude) seperti yang ada pada pemetaan.

```
var myGeographicCoordinates = new GLatLng(myLatitude, myLongitude)
```

Catatan: proses merubah sebuah alamat menjadi titik geografis dikenal dengan *geocoding* dan dibahas secara rinci dalam sesi Google Maps API Service (<http://code.google.com/apis/maps/documentation/services.html#Geocoding>).

Selain berguna untuk mereferensikan sebuah titik geografis, juga dapat digunakan untuk menentukan batas-batas geografis sebuah obyek. Sebagai contoh sebuah peta ditampilkan dalam window dan didalamnya dikenal dengan *viewport*. *Viewport* dapat didefinisikan oleh titik pada tiap sudutnya. Obyek `GLatLngBounds` menyediakan fungsionalitas, menentukan sebuah daerah kotak menggunakan dua obyek `GLatLng` mewakili sudut barat daya (*southwest*) dan timur laut (*northeast*).

Obyek `GLatLng` banyak digunakan dalam Google Maps API. Obyek `GMarker` mengambil `GLatLng` dalam konstruktornya, misalnya, menempatkan sebuah marker pada peta pada lokasi geografis tertentu.

Contoh berikut menggunakan `getBounds()` untuk mengembalikan nilai *viewport*, dan secara random membuat 10 marker dalam peta didalam batas-batas tersebut:

```
function initialize() {
    var map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(-7.794182, 110.365537), 13);

    // Add 10 markers to the map at random locations
    var bounds = map.getBounds();
    var southWest = bounds.getSouthWest();
    var northEast = bounds.getNorthEast();
    var lngSpan = northEast.lng() - southWest.lng();
    var latSpan = northEast.lat() - southWest.lat();
    for (var i = 0; i < 10; i++) {
        var point = new GLatLng(southWest.lat() + latSpan * Math.random(),
            southWest.lng() + lngSpan * Math.random());
        map.addOverlay(new GMarker(point));
    }
}
```

Pada gambar di atas terlihat sepuluh marker terbentuk secara random disekitar kota Yogyakarta (sebagai pusat peta).

Attribute Map

Setiap peta berisi sejumlah atribut yang dapat diperiksa atau ditetapkan. Sebagai contoh, untuk mencari dimensi *viewport* saat ini, menggunakan obyek GMap2 method `getBounds()` untuk mengembalikan nilai `GLatLngBounds`.

Setiap peta juga berisi, *zoom level*, yang mendefinisikan resolusi tampilan aktif. *Zoom level* bernilai antara 0 (level terendah, yang akan menampilkan peta dunia) sampai 19 (level tertinggi, sampai bangunan) yang mungkin dalam tampilan peta normal. *Zoom level* sangat tergantung dibagian mana dunia yang dicari berada. *Zoom level* sampai 20 mungkin sampai tampilan satelit.

Kita dapat mengambil level zoom yang sedang aktif menggunakan obyek GMap2 method `getZoom()`.

Tipe Map

Ada banyak tipe peta yang tersedia dalam Google Maps API. Secara default, peta muncul dalam Google Maps API menggunakan standar “painted” peta jalan. Namun, Google Maps API juga mendukung jenis data yang lain.

Tipe Map yang Sering Digunakan

Berikut tipe peta yang biasanya digunakan dalam Google Maps API:

1. `G_NORMAL_MAP` menampilkan default view peta jalan.
2. `G_SATELLITE_MAP` menampilkan gambar Google Earth (citra satelit).
3. `G_HYBRID_MAP` menampilkan campuran normal datelit view.
4. `G_DEFAULT_MAP_TYPES` berisi sebuah array dari tiga tipe, digunakan untuk proses interaktif.
5. `G_PHYSICAL_MAP` menampilkan sebuah bentuk fisik peta berbasis informasi terrain.

Kita dapat mengatur tipe peta menggunakan obyek GMap2 method setMapType(). Seperti contoh berikut mengatur menggunakan tampilan satelit dari Google Earth.

```
var map = new GMap2(document.getElementById("map_canvas"));
map.setMapType(G_SATELLITE_MAP);
```

Gambar peta menggunakan tipe G_SATELLITE_MAP

Tipe Marker Map Google

Google Maps API juga mendukung membuat tanda menggunakan Google Map Marker. Tanda Google Map Marker diciptakan melalui kontribusi dari seluruh dunia, dan area yang belum tercakup oleh Google Maps.

Berikut tipe Google Map Marker yang tersedia:

1. G_MAPMAKER_NORMAL_MAP menampilkan standar tampilan peta jalan
2. G_MAPMAKER_HYBRID_MAP menampilkan campuran tampilan normal dan citra satelit

Informasi tentang Google Map Marker dapat dilihat:

<http://sites.google.com/site/mapmakeruserhelp/home>

Tipe Map Celestial

Google Maps menyediakan beberapa tipe tambahan untuk berhubungan dengan ruang angkasa diluar bumi:

1. G_MOON_ELEVATION_MAP menampilkan displays sebuah bayangan terrain peta dari permukaan bulan, kode warna berdasar ketinggian.
2. G_MOON_VISIBLE_MAP menampilkan photographi image diambil dari orbit disekitar bulan.
3. G_MARS_ELEVATION_MAP menampilkan bayangan terrain peta permukaan Mars, kode warna berdasar ketinggian.
4. G_MARS_VISIBLE_MAP menampilkan photograhi diambil dari orbit disekitar Mars.

5. `G_MARS_INFRARED_MAP` menampilkan sebuah bayangan infrared peta permukaan Mars. displays a shaded infrared map of the surface of Mars, dimana daerah hangat terlihat lebih cerah dan daerah dingin terlihat lebih gelap.
6. `G_SKY_VISIBLE_MAP` menampilkan sebuah mosaic langit, seperti yang terlihat di bumi.

Tipe Map Plugin Google Earth

Jenis peta khusus, `G_SATELLITE_3D_MAP` menampilkan sebuah peta 3D dalam plugin Google Earth. Untuk user yang sudah menginstal plugin, memilih peta tipe ini akan membuat instan Earth untuk peta dan inisialisasi peta. Anda dapat menggunakan `GMap2.getEarthInstance()` untuk mengambil contoh dan Earth memanipulasi menggunakan Google Earth API. User tanpa plugin akan meminta untuk menginstalnya. Dokumentasi tentang Google Earth API dapat dibaca pada: <http://code.google.com/apis/earth/documentation>

Interaksi Map

Sekarang setelah mempunyai sebuah obyek `GMap2`, dan dapat berinteraksi dengannya. Obyek peta dasar terlihat dan berlaku hampir sama dengan ketika berinteraksi dengan Google Maps di website dengan banyak *buit-in* yang dapat dimanfaatkan oleh obyek itu sendiri.

Secara default, obyek peta cenderung berinteraksi sama dengan aktivitas user di <http://maps.google.com>. Anda dapat mengubah perilaku ini dengan sejumlah metode utilitas. Sebagai contoh method `GMap2.disableDragging()` yang menonaktifkan kemampuan klik dan drag peta ke lokasi baru, juga dapat berinteraksi dengan pemrograman peta. Obyek `GMap2` mendukung sejumlah method mengubah status peta secara langsung. Contoh method-method yang dapat dioperasikan melalui pemrograman peta: `setCenter()`, `panTo()`, `zoomIn()`, bukan melalui interaksi user ke peta.

Contoh berikut menampilkan sebuah peta dan menyediakan tombol untuk menjalankan method `panTo`, yang berpusat pada titik tertentu. Jika titik tertentu

tersebut terlihat dipeta, peta akan berpindah titik pusatnya dengan halus, jika tidak maka akan melompat ke titik tersebut.

```
var map;

function initialize() {
  if (GBrowserIsCompatible()) {
    map = new GMap2(document.getElementById("map_canvas"));
    map.setCenter(new GLatLng(-7.794182, 110.365537), 13);
  }
}

function animate() {
  map.panTo(new GLatLng(-7.794182, 110.465537));
}
```

Interaksi lebih kompleks dapat dilihat melalui

<http://code.google.com/apis/maps/documentation/events.html>

Window Info

Masing-masing peta dalam Google Maps API dapat menunjukkan satu informasi "info window" dari tipe GInfoWindow, yang menampilkan window berisi HTML diatas peta. Info window terlihat seperti dalam komik (*word ballon*), memiliki isi dan batang yang meruncing pada titik tertentu pada peta. Dapat dilihat info window beraksi dengan mengklik tanda di Google Maps.

Obyek GInfoWindow tidak mempunyai konstruktor. Info window secara otomatis terbentuk dan melekat ketika membuat peta. Kita tidak dapat menampilkan lebih dari satu window pada waktu yang sama pada sebuah peta, tetapi dapat memindahkan info window dan mengubah isinya sesuai dengan kebutuhan.

Obyek GMap2 menyediakan method `openInfoWindow()`, yang membutuhkan titik dan sebuah elemen HTML DOM sebagai argumen. Elemen HTML DOM ditambahkan ke jendela info container, dan jendela info tip mengacu pada titik tertentu.

Method `GMap2openInfoWindowHtml()` mirip, tapi dibutuhkan string HTML sebagai argumen kedua dari pada sebuah elemen DOM.

Untuk membuat info window, panggil method `openInfoWindow`, lewatkan sebuah lokasi dan sebuah elemen DOM untuk ditampilkan. Kode berikut akan menampilkan jendela info terletak ditengah dengan pesan sederhana "Hello, world".

```
var map = new GMap2(document.getElementById("map_canvas"));
map.setCenter(new GLatLng(-7.794182, 110.365537), 13);
map.openInfoWindow(map.getCenter(),
    document.createTextNode("Hello, world"));
```

Hasil skrip diatas adalah sebagai berikut:

Google Maps API adalah fasilitas dari Google untuk menambahkan peta dalam website. Google Maps API dapat ditambahkan ke website menggunakan JavaScript. API tersebut menyediakan banyak fasilitas dan utilitas untuk memanipulasi peta dan menambahkan konten ke peta melalui berbagai layanan, memungkinkan untuk membuat aplikasi peta yang kuat pada website.