

BAB 3 LANDASAN TEORI

3.1 Web Service

3.1.1 Apa itu Web Service?

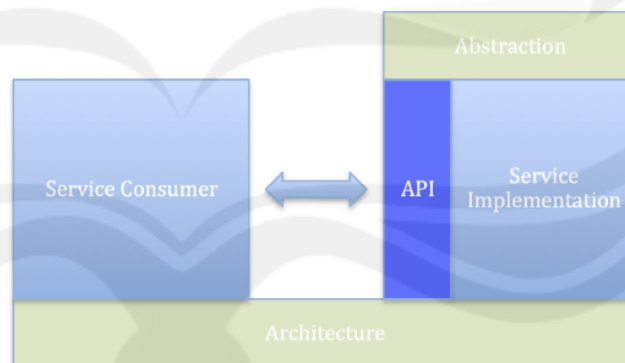
Web Service adalah standard dan metode pemrograman yang digunakan untuk berbagi data antara aplikasi. Khususnya untuk mendistribusikan layanan di internet. Dengan memperhatikan perkembangan perangkat *mobile*, mengembangkan web service pada platform *mobile* menjadi penting. Namun, ada beberapa pertimbangan yang muncul mengenai kemungkinan adanya *overhead*. Hal ini disebabkan oleh 2 faktor. Pertama, dalam proses encoding dan decoding SOAP yang berbasis XML membutuhkan lebih banyak sumber daya. Kedua, adanya batasan-batasan pada lingkungan *mobile* seperti kecepatan prosesor yang terbatas, masa hidup baterai, dan koneksi yang tidak selalu terjamin. REST menawarkan solusi dengan fleksibilitas yang tinggi dan *overhead* yang rendah. (Hamad et al., 2010)

Web service didesain dan dikembangkan dengan menggunakan service oriented architecture (SOA). SOA adalah arsitektur yang mendukung aplikasi baru dikembangkan dengan menggunakan atau mengkombinasikan web service yang sudah ada.

3.1.2 Komponen Web Service

Web service terdiri dari empat komponen utama, yaitu:

- Arsitektur, merupakan bagian yang mengatur logika pengorganisasian data, kode, dan alur komunikasi service.
- Abstraksi, berguna untuk menyembunyikan proses implementasi dan memungkinkan arsitektur service mengontrol fungsionalitas untuk pemakai maupun aplikasi lain (service client).
- Implementasi, berisi kode pemrograman untuk melakukan komputasi dan manipulasi data ketika client mengakses service.
- Application programming interface (API) melakukan pendefinisian dan pengontrolan terhadap operasi yang dilakukan client untuk mengakses implementasi yang telah diberi abstraksi.



Gambar 3.1 Komponen Web Service

API Web Service mengontrol bagian implementasi yang dapat dilihat oleh client dan bagaimana client berinteraksi dengan service dalam program. Dalam implementasi, API memberikan batasan pada client dan service sehingga masing-masing dapat dikembangkan secara terpisah.

API mendefinisikan apa yang dapat diterima service dan jawabannya. Secara khusus, interface service mendeskripsikan:

- Operasi yang didukung oleh service
- Format permintaan dan jawaban yang dapat diproses dan dihasilkan oleh service
- Service end point, misalnya pada protocol HTTP, ini adalah HTTP URLs yang dapat digunakan client untuk mengakses service.
- Detil transport lainnya.
- Batasan Quality of Service (Qos) misalnya jika service aman maka ini mendefinisikan protocol keamanan yang harus digunakan client ketika mengakses service.

API memungkinkan client mengakses service dengan berbagai teknologi, ini merupakan salah satu keunggulan dalam mengurangi biaya implementasi. Selain itu, pengguna dapat menambahkan atau mengurangi service yang serupa tanpa banyak memodifikasi program (Jayathilaka et al., 2014). API dapat mengurangi tuntutan pada pengembangan hardware dan pendukung sehingga membuatnya lebih cepat dan mudah terhubung dengan penyedia data (Chan, 2010).

3.2 REST

3.2.1 Apa itu REST?

REST adalah model arsitektur perangkat lunak yang meliputi cara data ditampilkan, diakses, dan

dimodifikasi pada web. Pada arsitektur REST, data dan fungsionalitas yang tersedia berperan sebagai sumber informasi dan diakses menggunakan *Uniform Resource Identifiers* (URIs) seperti link pada web. Sumber informasi dapat berperan dengan menggunakan metode sederhana yang telah didefinisikan. Arsitektur REST pada dasarnya adalah arsitektur client-server dan dirancang untuk menggunakan *stateless protocol*, seperti HTTP. Pada arsitektur REST, client dan server melakukan pertukaran data menggunakan protokol dan antarmuka yang telah terstandarisasi. Prinsip ini menyebabkan aplikasi REST menjadi sederhana, ringan, dan memiliki kemampuan yang handal. (Hamad et al., 2010)

REST adalah filosofi desain untuk aplikasi terdistribusi, secara khusus sebagai panduan untuk membangun aplikasi yang berjalan pada protocol HTTP.

3.2.2 Penggunaan REST

Dalam penggunaan REST, harus ditetapkan sebuah objek sebagai sumber sumber daya. Informasi yang berkaitan dengan sumber daya tersebut diakses dengan menggunakan URI. Sehingga lebih dari satu URI dapat mengakses sebuah sumber daya. Penyajian data yang diperoleh dapat berdasarkan data itu sendiri atau informasi tambahan pada saat mengakses URI. (Newmarch, 2009)

3.2.3 Prinsip RESTful Web Service menurut Fielding

- **Sumber Daya dan Representasi:** Sumber daya adalah sebuah bagian aplikasi web yang dapat

diakses oleh client. Karena sumber daya ini tidak dapat dikirim melalui jaringan, menyediakan sumber daya berarti menyediakan representasi keadaan sumber daya.

- Dapat diberi alamat dan dihubungkan: Addressability adalah memberi alamat sumber daya dengan memberikan spesifikasi URI. Dalam REST, setiap sumber daya harus memiliki paling sedikit satu URI.
- Memiliki interface yang seragam: Pada arsitektur REST, protocol komunikasi/interface yang digunakan harus seragam, apapun URI yang diakses.
- *Stateless*: *Stateless* berarti sebuah aplikasi web tidak bertanggung jawab terhadap status client. Service hanya bertanggung jawab terhadap sumber daya dan menyediakan antarmuka yang seragam pada client (Saquicela et al., 2012).

3.2.4 Istilah yang didefinisikan menurut Fielding

- HTTP GET digunakan untuk mendapatkan representasi sumber daya dari URI. Service yang diberikan melalui interface ini seharusnya tidak mengakibatkan perubahan keadaan pengguna.
- HTTP DELETE digunakan untuk menghapus representasi sumber daya.
- HTTP POST digunakan untuk mengubah atau membuat representasi sumber daya.

- HTTP PUT digunakan untuk membuat representasi sumber daya (Saquicela et al., 2012).

3.2.5 Karakteristik REST

- Memiliki waktu response dan pemuatan data server yang telah ditingkatkan untuk mendukung *caching*.
- Meningkatkan skalabilitas server dengan mengurangi kebutuhan untuk memperhatikan status komunikasi.
- Membutuhkan perangkat lunak aplikasi pada sisi client yang lebih sedikit dari pada pendekatan lain karena sebuah web browser dapat mengakses aplikasi dan sumber daya apapun.
- Memiliki lebih sedikit ketergantungan dengan vendor perangkat lunak daripada mekanisme yang menggunakan lapisan framework pesan tambahan pada bagian atas HTTP.
- Menyediakan fungsionalitas yang sama jika dibandingkan dengan pendekatan alternative komunikasi lainnya.
- Tidak membutuhkan mekanisme pencarian sumber daya yang terpisah karena menggunakan *hyperlink* sebagai informasi.
- Memiliki karakteristik penyesuaian dan pengembangan jangka panjang yang lebih baik daripada RPC yang disebabkan oleh:
- Tipe dokumen seperti HTML dapat berkembang tanpa merusak kompatibilitas.

- Sumber daya dapat memberikan dukungan pada tipe data yang baru tanpa mengurangi dukungan pada tipe data yang lama (tipe MIME). Berdasarkan penelitian terhadap aplikasi mobile antara SOAP dan REST:
- REST memiliki kelebihan waktu respon yang lebih cepat dan ukuran pesan yang lebih kecil (Hamad et al., 2010).

3.3 JSON (JavaScript Object Notation)

JavaScript Object Notation (JSON) adalah format pertukaran data ringan berbasis teks dan tidak tergantung pada bahasa. JSON pada awalnya berasal dari standard bahasa pemrograman ECMAScript. JSON mendefinisikan sedikit aturan dalam pemrograman untuk penerapan representasi struktur data. JSON mewakili 4 tipe primitive, yaitu string, number, Boolean, dan null serta 2 tipe struktural yaitu objek dan array.

3.3.1 Struktur JSON

Teks JSON berupa serangkaian token. Token ini terdiri dari 6 karakter struktural, tipe data string, number, dan 3 nama literal yaitu false, null, dan true (penulisan dengan huruf kecil). Penulisan string diawali dengan tanda petik. Penulisan number tidak boleh diawali dengan 0.

Teks JSON adalah nilai yang saling berhubungan dan membentuk objek atau array. Struktur array direpresentasikan dengan kurung siku, terdiri dari 0 atau lebih nilai. Struktur objek direpresentasikan dengan kurung kurawal,

terdiri dari 0 atau lebih nama atau anggota. Nama pada sebuah objek harus unik, nama dan nilai dipisahkan dengan tanda titik dua.

Penulisan ruang kosong (ws) pada JSON-text = ws value ws. Ws meliputi spasi, tab horizontal, baris baru, dan *carriage return*. Ws yang tidak didefinisikan hanya diijinkan diletakkan sebelum atau setelah 6 struktural karakter, yang meliputi:

- Awal-array = ws %x5B ws ; [kurung siku kiri
- Awal-objek = ws %x7B ws ; { kurung kurawal kiri
- Akhir-array = ws %x5D ws ;] kurung siku kanan
- Akhir objek = ws %x7D ws ; } kurung kurawal kanan
- Pemisah nama = ws %x3A ws ; : titik dua
- Pemisah nilai = ws %x2C ws ; , koma

3.3.2 Parser

JSON Parser digunakan untuk mengubah teks JSON menjadi bentuk lain. JSON parser dapat menerima semua teks yang sesuai dengan struktur JSON dan mungkin dapat menerima bentuk atau ekstensi yang bukan JSON.

3.3.3 Contoh Penulisan JSON

Contoh JSON Object ditunjukkan pada Gambar 3.2 JSON Object . Image adalah objek yang memiliki anggota Thumbnail (objek) dan IDs berupa array tipe data number. Contoh JSON

Array yang memiliki 2 objek ditunjukkan pada Gambar 3.4 (Bray, 2014). Sedangkan contoh JSON text ditunjukkan pada Gambar 3.3 (Bray, 2014).

```
{
  "Image": {
    "Width": 800,
    "Height": 600,
    "Title": "View from 15th Floor",
    "Thumbnail": {
      "Url": "http://www.example.com/image/481989943",
      "Height": 125,
      "Width": 100
    },
    "Animated" : false,
    "IDs": [116, 943, 234, 38793]
  }
}
```

Gambar 3.2 JSON Object

```
[
  {
    "precision": "zip",
    "Latitude": 37.7668,
    "Longitude": -122.3959,
    "Address": "",
    "City": "SAN FRANCISCO",
    "State": "CA",
    "Zip": "94107",
    "Country": "US"
  },
  {
    "precision": "zip",
    "Latitude": 37.371991,
    "Longitude": -122.026020,
    "Address": "",
    "City": "SUNNYVALE",
    "State": "CA",
    "Zip": "94085",
    "Country": "US"
  }
]
```

```
"Hello world!"
42
true
```

Gambar 3.3 JSON Text

Gambar 3.4 JSON Array

3.4 MySQL

MySQL adalah open source SQL yang disediakan oleh MySQL AB [MSQL+], sebuah perusahaan komersial

yang menyediakan service terhadap basis data MySQL. MySQL menurut Wen Tian:

- MySQL adalah *database management sistem*: basis data adalah struktur kumpulan data. Database management system seperti MySQL diperlukan agar data yang tersimpan pada basis data dapat diakses, ditambahkan, dan diproses.
- MySQL adalah *relational database management system*: sehingga data disimpan dalam tabel yang terpisah untuk meningkatkan fleksibilitas dan kecepatan. Tabel yang ada dihubungkan dengan relasi yang telah didefinisikan sehingga ketika melakukan sebuah permintaan, dapat mengakses data dari berbagai tabel.
- MySQL adalah perangkat lunak open source: dapat digunakan dan dimodifikasi oleh siapapun dan dapat diperoleh secara gratis di internet.
- MySQL adalah sistem client/server: terdiri dari multi-threaded SQL server yang mendukung backbend yang berbeda, program client dan library berbeda, perangkat administrasi, dan beberapa antarmuka pemrograman (Tian, 2001).

3.4.1 Mengapa menggunakan MySQL

Cepat, dapat dipercaya, dan mudah dalam penggunaannya. Konektivitas, kecepatan, dan keamanan yang terjamin membuat MySQL

cocok digunakan untuk mengakses data di internet. Berikut ini (Gambar 3.5) adalah perbandingan antara MySQL dengan manager basis data lain (350,768 baris) (Tian, 2001):

Database	Seconds
mysql	381
mysql_odbc	619
db2_odbc	3460
informix_odbc	2692
ms_sql_odbc	4012
oracle_odbc	11291
solid_odbc	1801
sybase_odbc	4802

Gambar 3.5 Perbandingan MySQL dengan manager basis data lain

3.4.2 Fasilitas Utama MySQL

- Multi-threaded menggunakan kernel thread, sehingga memungkinkan menggunakan lebih dari 1 CPU.
- Mendukung berbagai API: C, C++, Java, Perl, PHP, Python, Tcl.
- Mendukung berbagai platform.
- Memiliki banyak tipe kolom: integer bertanda/tidak bertanda, 1, 2, 3, 4, 8 bytes long, float, double, char, varchar, text, blob, date, time, datetime, timestamp, year, set, dan ENUM type [MSQL+]
- Operasi join yang sangat cepat.
- Mendukung semua fungsi dan operator dalam query select dan where.
- Fungsi SQL diimplementasikan dengan kelas library yang sangat dioptimalkan sehingga

sangat cepat, biasanya tidak ada alokasi memori setelah inisialisasi query.

- Hak akses dan password yang fleksibel dan aman. Semua password dienkripsi ketika terhubung dengan server.
- Mendukung ODBC (Open Database Connectivity).
- Pemrosesan table yang cepat, didukung oleh penekanan index.
- Tersedia hingga 32 indeks per table. Setiap indeks terdiri dari 1-16 kolom atau bagian kolom.
- Setiap kolom memiliki nilai default.
- Tidak ada pembuangan memori. MySQL telah dites dengan Purify (alat pendeteksi pembuangan memori).
- Mengizinkan pemberian alias/nama lain pada table dan kolom (Tian, 2001).

3.5 PHP

3.5.1 Apakah PHP?

PHP adalah bahasa pemrograman HTML yang diperluas pada sisi server. Kode PHP diawali dengan tag awal (<?php) dan tag akhir (?>). Bahasa pemrograman PHP ditujukan pada sisi server. Proses pengkodean dengan PHP pada sisi server tidak diketahui oleh client. Selain itu, PHP mendukung berbagai basis data. Beberapa basis data yang didukung yaitu MySQL, Oracle, dBase, PostgreSQL, IBM, DB2, Informix. PHP juga memungkinkan

komunikasi dengan service lain dengan menggunakan protocol seperti IMAP, SNMP, NNTP, POP3, dan HTTP.

3.5.2 Sejarah PHP

Pada tahun 1994, Rasmus Lerdorf membuat *engine parsing* bernama PHP/FI untuk menterjemahkan beberapa makro. PHP/FI ini digunakan untuk membuat buku tamu, *counter*, dan *homepage*.

PHP/FI bersifat open source, pada tahun 1997, para pengembang berhasil menggunakan PHP/FI pada lebih dari 500.000 website di dunia untuk mengatasi masalah koneksi ke basis data dan menampilkan isi halaman web secara dinamis.

Pada tahun 1998, PHP 3.0 diterbitkan. PHP ini sudah dapat berjalan pada berbagai platform (bukan hanya Linux), sever web, beberapa basis data, protocol SNMP (Simple Network Management Protokol) dan IMAP (Internet Message Access Protocol). Menurut survey yang dilakukan oleh <http://netcraft.org> pada tahun 2011, pengguna php telah mencapai 9,5 juta domain (Id, 2011).

3.6 CodeIgniter

CodeIgniter adalah framework aplikasi web yang bersifat open source dan digunakan untuk pengembangan web dinamis. CodeIgniter telah dilengkapi dengan library yang dapat mempermudah pengembangan (Id, 2011).

3.6.1 Kelebihan CodeIgniter menurut Id

- Menggunakan pola MVC.
- Minimalisasi penggunaan \$_GET dengan url.

- Kemudahan dalam mengakses atau memodifikasi library atau helper.
- Kecepatan, berdasarkan benchmark CodeIgniter merupakan salah satu framework tercepat saat ini.
- Mudah dipelajari.
- Mudah dimodifikasi dan beradaptasi.
- Dokumentasi lengkap dan jelas (Id, 2011).

3.6.2 Kelebihan CodeIgniter menurut Upton

- Menghemat waktu: Baik dalam proses pembelajaran maupun penulisan code. Dengan CodeIgniter, code yang ditulis menjadi lebih pendek.
- Situs menjadi lebih handal.
- Kehandalan basis data lebih terjamin karena adanya library sehingga dapat menghindari symbol-simbol tertentu yang dapat menyebabkan basis data mengalami *crash*.
- Semua code pada CodeIgniter ditulis dengan baik dan handal (Upton, 2007).

3.6.3 Framework CodeIgniter menggunakan pola MVC, apakah MVC?

MVC (Model View Controller) adalah teknik pemrograman yang memisahkan logika pemrograman, data, dan antarmuka.

1 Model (data)

Model berhubungan dengan data dan interaksi basis data atau web service. Model juga mewakili struktur data dari aplikasi, dapat

berupa basis data maupun data lain, seperti file teks, file XML, atau web service. Pada aplikasi web, biasanya berhubungan dengan basis data sehingga model berisi perintah-perintah SQL.

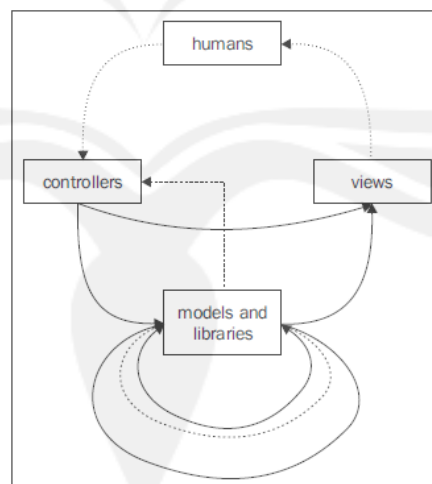
2 View (antarmuka)

View berhubungan dengan segala sesuatu yang akan ditampilkan ke pengguna akhir. Pada view, harus dihindari logika dan pemrosesan data. View hanya dikhususkan menampilkan data hasil dari model dan controller.

3 Controller (proses bisnis)

Controller bertindak sebagai penghubung antara model dan view. Di dalam controller terdapat fungsi kelas-kelas dan fungsi.

3.6.4 Ilustrasi interaksi antar model, view, controller



Gambar 3.6 Ilustrasi model, view, controller

Pada Gambar 3.6, garis yang tidak terputus menunjukkan akses langsung. Garis yang terputus menunjukkan proses pengiriman informasi dengan

mengembalikan nilai. Sedangkan garis titik-titik mewakili pengiriman informasi melalui manusia (Upton, 2007).

3.7 GPS

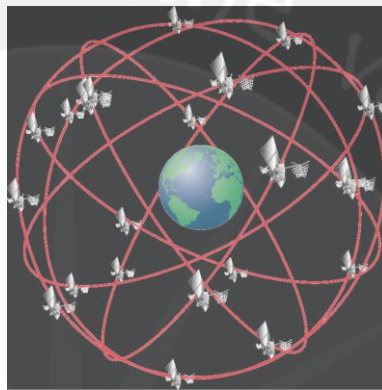
GPS (Global Positioning System) adalah layanan navigasi dan posisi yang paling terkenal. GPS menggunakan satelit di luar angkasa yang mengelilingi bumi untuk mengirimkan sinyal posisi. GPS receiver akan menerima sinyal ini dan menghitung posisinya. GPS menyediakan layanan navigasi dan penentuan posisi untuk semua jenis alat komunikasi dan transportasi di seluruh dunia.

GPS dikembangkan oleh U.S Department of Defense pada awal tahun 1970s. Pada awalnya hanya digunakan untuk keperluan militer. Kemudian pada bulan Juli tahun 2000 terbuka untuk warga negara. (Hsu, 2010).

3.7.1 Prinsip GPS

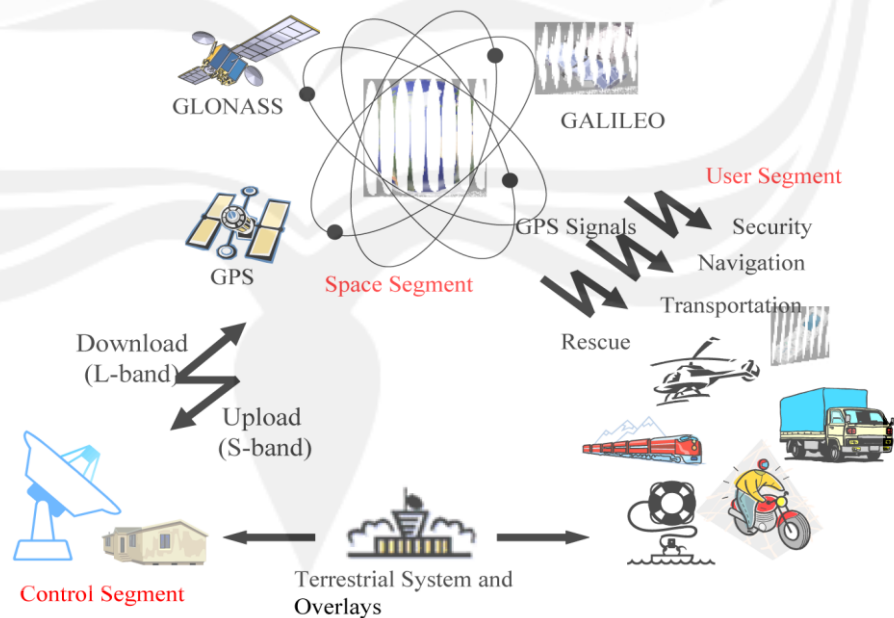
GPS adalah system penentuan posisi dan navigasi dengan 24-32 satelit. GPS secara terus menerus menyediakan informasi posisi dan waktu dimanapun dan kapanpun tanpa memperhatikan cuaca. GPS dapat melayani sebanyak apapun pengguna. Prinsip dasar GPS adalah menghitung jarak antara satelit dan GPS receiver dari transmisi sinyal radio. Sebuah GPS receiver memerlukan 4 atau lebih satelit untuk menyimpulkan lokasi dari jarak antara satelit dan GPS receiver, proses ini disebut triangulation. (Yulianto, 2010).

Satelit GPS mengelilingi bumi dengan orbit berbentuk lonjong, yaitu 55 derajat dari equator. Radius maksimal antara satelit dengan bumi adalah 26,560 km dari pusat bumi. Dalam 1 kali orbit, satelit GPS membutuhkan waktu 11 jam 58 menit. Gambar 3.7 menunjukkan peta perputaran satelit GPS. (Hsu, 2010)



Gambar 3.7 Peta Orbit GPS

3.7.2 Bagian-bagian GPS



Gambar 3.8 Bagian-bagian GPS

Gambar 3.8 adalah bagian-bagian GPS. GPS terdiri dari 3 bagian (Hsu, 2010), yaitu:

1 Bagian luar angkasa

Bagian ini terdiri dari 24 sampai 32 satelit. Selain itu juga terdapat satelit back-up untuk memastikan kemampuan operasional yang penuh. Setiap satelit memancarkan gelombang unik PRN (Pseudo Random Noise) untuk mengukur jarak antara GPS receiver dengan satelit. Selain itu, satelit memiliki antena S-band untuk berkomunikasi dengan bagian control untuk pemeliharaan service GPS.

2 Bagian control

Bagian control terdiri dari Master Control Station (MCS), 5 Monitor Station di seluruh dunia, dan 4 antena bawah tanah. Bagian ini memiliki tanggung jawab untuk melacak, memonitor, memerintahkan, dan mengontrol orbit satelit supaya posisi satelit dapat diketahui, integritas sistem, memantau atomic clock (berfungsi untuk menghasilkan sinyal yang akan dikirimkan ke receiver), serta tanggung jawab lainnya. Pengamatan satelit GPS disimpan pada monitor station dan dikirimkan ke MCS untuk diproses. MSC mengirimkan hasil pemrosesan yang berupa control informasi ke monitor control station dengan antena bawah tanah untuk dikirimkan ke satelit melalui jalur S-band.

3 Bagian pengguna

Ketika pengguna mengaktifkan GPS receiver untuk menerima sinyal L-band dari satelit dan berada pada suatu lokasi, GPS receiver dapat

menentukan lokasi kita dan segala sesuatu yang berhubungan dengannya. GPS Receiver ditunjukkan oleh Gambar 3.9 (Hsu, 2010)



Gambar 3.9 Berbagai GPS Receiver

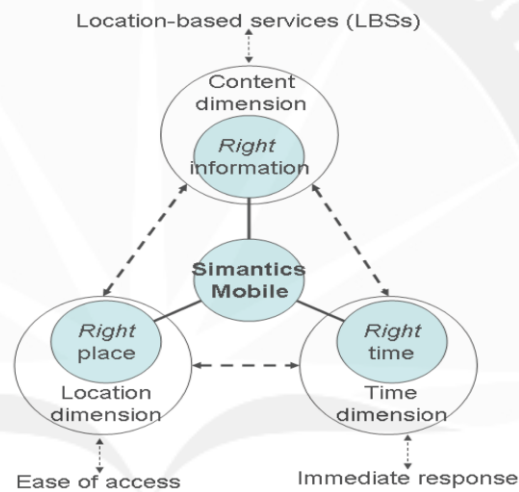
3.8 LBS

LBS (Location Based Service) adalah layanan yang diberikan berdasarkan lokasi perangkat mobile yang digunakan. Berdasarkan penelitian yang dilakukan oleh Fleishmand Hilard (6 Februari 2012) 80% dari pemilik ponsel pintar memiliki layanan berbasis lokasi pada ponsel pintar mereka. Menurut SNL Kagan (Cohen, 20 Januari 2011), pengguna layanan berbasis lokasi di US meningkat dari 12,3 juta pada tahun 2009 menjadi 33,2 juta pada tahun 2010 (Hu et al., 2012).

Location Based Service (LBS) dapat menggunakan 2 pendekatan, misalnya visual tracking dan Nose's positioning technique pada Simantic Mobile. Visual tracking memberikan perkiraan posisi secara tepat waktu. Ini adalah metode navigasi berdasarkan interaksi user dengan aplikasi AR. Sebaliknya, Nose's positioning technique menggunakan standar metode yang sudah ada kemudian

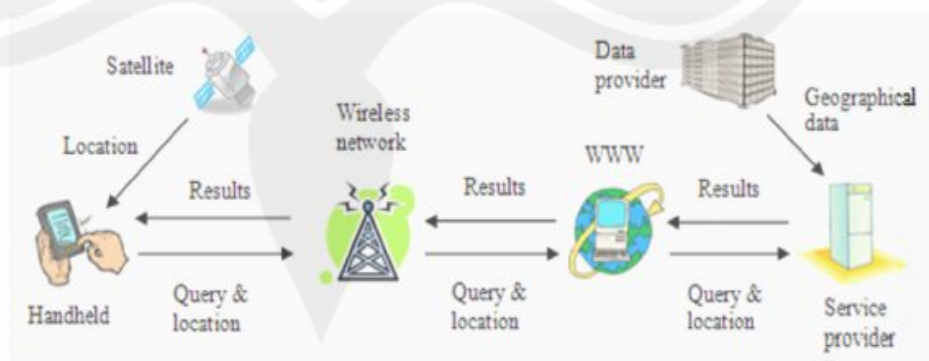
memperkuat metode tersebut untuk meningkatkan fungsionalitas dan kegunaan (Prammanee et al., n.d.).

Menurut Prammanee, Siltanen, Luukkainen, & Karhela, dalam location based services, terdapat 3 dimensi pengetahuan yang berkaitan, yaitu dimensi waktu, dimensi isi, dan dimensi lokasi, dipadukan dalam layanan mobile (Prammanee et al., n.d.), ditunjukkan oleh Gambar 3.10



Gambar 3.10 Location Based Service

3.8.1 Struktur Sistem LBS



Gambar 3.11 Struktur Sistem LBS

Struktur system LBS yang ditunjukkan pada Gambar 3.11 terdiri dari 5 komponen utama (Hu et al., 2012), yaitu:

- 1 Perangkat mobile: computer yang dapat dipegang oleh tangan, pada kebanyakan kasus adalah ponsel pintar.
- 2 Sistem Penentuan Posisi: satelit yang memberikan informasi mengenai waktu dan lokasi kepada siapapun dengan receiver.
- 3 Jaringan mobile dan tanpa kabel: mengirimkan informasi antara perangkat dengan penyedia layanan.
- 4 Service Provider: penyedia layanan berbasis lokasi.
- 5 Penyedia data geografis: basis data yang menyimpan data geografis.

3.9 Android

Android adalah perangkat lunak untuk perangkat mobile, seperti ponsel pintar dan PC tablet. Android dikembangkan oleh Open Handset Alliance, yaitu perkumpulan 80 perusahaan perangkat lunak, perangkat keras, dan telekomunikasi yang dipimpin oleh Google mengembangkan standard terbuka untuk perangkat mobile. Google membayar biaya awal projek pengembangan perangkat lunak Android Inc. pada tahun 2005. Android terdiri dari system operasi, middleware, dan key aplikasi.

Android SDK menyediakan tools dan API yang dibutuhkan untuk mengembangkan aplikasi pada platform android dengan bahasa pemrograman Java (Hu et al., 2012).

Android SDK memiliki API untuk perangkat keras berbasis lokasi seperti GPS, kamera, koneksi jaringan, WiFi, Bluetooth, accelerometers, layar sentuh, dan control power. (De Lucia et al., 2012).

3.10 Kebudayaan

Menurut Kamus Besar Bahasa Indonesia, kebudayaan adalah hasil kegiatan dan penciptaan batin (akal budi) manusia seperti kepercayaan, kesenian, dan adat istiadat (Kebudayaan, 2015). Beberapa tempat wisata Yogyakarta yang wajib dikunjungi di antaranya Candi Prambanan, Pantai Parangtritis, Jalan Malioboro, Goa Jomblang, Arung Jeram Citra Elo, Keraton Yogyakarta, Candi Prambanan, Istana Air Taman Sari (Wisata, 2015). Menurut data dari Statistik Kementerian Pariwisata, devisa Pariwisata pada tahun 2009 memperoleh urutan ke 4 setelah minyak & gas bumi, batu bara, dan minyak kelapa sawit, pada tahun 2010-2012 memperoleh urutan ke 5, dan pada tahun 2013 kembali menduduki peringkat ke 4. (Pariwisata, 2014).