

BAB II

LANDASAN TEORI

1.1. RFID (Radio Frequency Identification)

Menurut (Saputra, 2008) *Frequency Identification* (RFID) adalah terminologi umum untuk teknologi non kontak yang menggunakan gelombang radio untuk mengidentifikasi orang atau objek secara otomatis. Ada sejumlah metoda identifikasi, namun yang paling umum adalah menyimpan nomor seri yang mengidentifikasi orang atau objek alam sebuah *microchip* yang dihubungkan dengan sebuah antena. Kombinasi antena dan *microchip* disebut RFID *transponder* atau RFID *tag*. Alat tersebut bekerja bersama sebuah RFID *reader*.

RFID adalah proses identifikasi seseorang atau objek dengan menggunakan frekuensi *transmisi* radio. RFID menggunakan frekuensi radio untuk membaca informasi dari sebuah *device* kecil yang disebut *tag* atau *transponder* (*transmitter* dan *responder*). *Tag* RFID akan mengenali diri sendiri ketika mendeteksi sinyal dari *device* yang kompatibel, yaitu pembaca RFID (RFID *reader*). RFID adalah teknologi identifikasi yang fleksibel, mudah digunakan, dan sangat cocok untuk operasi otomatis. RFID mengkombinasikan keunggulan yang tidak tersedia pada teknologi identifikasi yang lain. RFID dapat disediakan dalam *device* yang hanya dapat dibaca saja (*Read Only*) atau dapat dibaca dan ditulis (*Read* atau *Write*), tidak memerlukan kontak langsung maupun jalur cahaya untuk dapat beroperasi, dapat berfungsi pada berbagai variasi kondisi lingkungan, dan menyediakan tingkat integritas data yang tinggi. Sebagai tambahan, karena teknologi ini sulit untuk dipalsukan, maka RFID dapat menyediakan tingkat keamanan yang tinggi.

Sistem RFID umumnya, *tag* atau *transponder* ditempelkan pada suatu objek. Setiap *tag* dapat membawa informasi yang unik, di antaranya: *serial number*, model, warna, tempat perakitan, dan data lain dari objek tersebut. Ketika *tag* ini melalui medan yang dihasilkan oleh pembaca RFID yang kompatibel, *tag* akan mentransmisikan informasi yang ada pada *tag* kepada pembaca RFID, sehingga proses identifikasi objek dapat dilakukan.

Sistem	Barco	OCR	Voice	Biometry	Smart card	RFID
---------------	--------------	------------	--------------	-----------------	-------------------	-------------

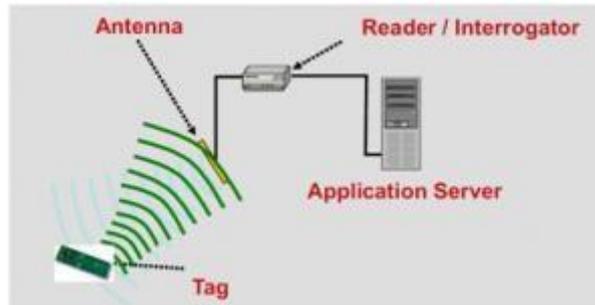
<i>Typical data quantity</i>	<i>1-100</i>	<i>1-100</i>	<i>-</i>	<i>-</i>	<i>16-64 k</i>	<i>16-64 k</i>
<i>Data density</i>	<i>Low</i>	<i>Low</i>	<i>High</i>	<i>High</i>	<i>Very High</i>	<i>high</i>
<i>Machine</i>	<i>Good</i>	<i>Good</i>	<i>Expen</i>	<i>Expensive</i>	<i>Good</i>	<i>good</i>
<i>Degradation</i>	<i>Limite</i>	<i>Limited</i>	<i>-</i>	<i>-</i>	<i>Contacs</i>	<i>No</i>
<i>/wPurchase cost/reading</i>	<i>dVery low</i>	<i>Medium</i>	<i>Very high</i>	<i>Very high</i>	<i>Low</i>	<i>influencemedium</i>
<i>Operating</i>	<i>Low</i>	<i>Low</i>	<i>None</i>	<i>None</i>	<i>Medium</i>	<i>none</i>
<i>Unauthorized</i>	<i>Slight</i>	<i>Slight</i>	<i>Possible</i>	<i>Impossible</i>	<i>Impossible</i>	<i>impossible</i>
<i>Reading speed</i>	<i>low~4s</i>	<i>low~3s</i>	<i>Very low</i>	<i>Very low >5-10s</i>	<i>low~4s</i>	<i>Very fast~0.5s</i>
<i>Maximum Distance</i>	<i>0-50cm</i>	<i><1 cm Scanner</i>	<i>0-50 Cm</i>	<i>Direct Contact</i>	<i>Direct Contact</i>	<i>Direct contact</i>

Tabel 2.1 menunjukkan perbandingan beberapa metode identifikasi yang ada dengan teknologi identifikasi menggunakan RFID.

Tabel 2.1 Perbandingan Antara RFID Dengan Sistem Identifikasi Lainnya

Sumber: (Saputra, 2008)

Teknologi RFID didasarkan pada prinsip kerja elektromagnetik, dimana komponen utama dari RFID *tag* adalah *chip* dan *tag* antena, dimana *chip* berisi informasi dan terhubung dengan *tag* antena. Informasi yang berada atau tersimpan dalam *chip* ini akan dikirim atau terbaca melalui gelombang elektromagnetik setelah *tag* antena menerima pancaran gelombang elektromagnetik dari *reader* antena. RFID *reader* ini yang sekaligus akan meneruskan informasi pada *aplication server*.



Gambar 2.1 Diagram Sederhana Sistem RFID Secara Umum.

Sumber: (Saputra, 2008)

Gambar 2.1 bisa dilihat diatas adalah diagram sederhana sistem RFID secara umum.

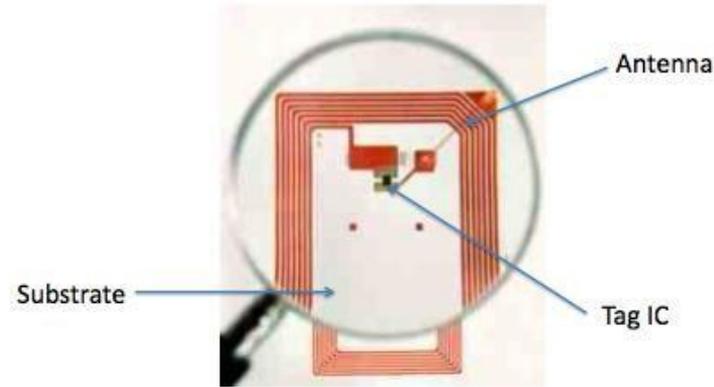
Seperti yang terlihat pada Gambar 2.1. Bisa dilihat bahwa dalam sistem RFID terdapat beberapa komponen untuk menunjang sistem agar dapat berjalan dengan baik. alam mengalokasikan sistem RFID tersebut, terdapat beberapa hal yang harus diperhatikan, yaitu:

1. Jenis *reader* yang dipakai
2. Jenis *tag* yang digunakan
3. Frekuensi operasi dari sistem dan
4. Jarak antara *reader* dan *tag* yang diinginkan.

1.1.1. RFID Tag (Transponder)

Menurut (Saputra, 2008) Sistem RFID umumnya, *tag* atau *transponder* ditempelkan pada suatu objek. Setiap *tag* dapat membawa beberapa jenis informasi. Ketika *tag* ini terdeteksi oleh medan yang dihasilkan oleh pembaca RFID yang kompatibel, *tag* akan mentransmisikan informasi yang ada pada *chip tag* kepada pembaca RFID, sehingga proses identifikasi objek dapat dilakukan.

Tag RFID adalah *device* yang dibuat dari rangkaian elektronika dan antenna yang terintegrasi di dalam rangkaian tersebut. Rangkaian elektronik dari *tag* RFID umumnya memiliki memori sehingga *tag* ini mempunyai kemampuan untuk menyimpan data. Memori pada *tag* terdapat beberapa jenis sel. Beberapa sel menyimpan data *read only*, misalnya *serial number* yang unik yang disimpan pada saat tag tersebut diproduksi. Sel lain pada RFID mungkin juga dapat ditulis dan dibaca secara berulang.



Gambar 2.2 Layout Dasar RFID Tag

Tag RFID diklasifikasikan menjadi lima kelas, yaitu:

1. CLASS 0/1-Read Only, Factory programmed

Jenis ini adalah jenis *tag* paling sederhana, dimana data di tulis sekali ketika dimanufaktur. Lalu memori dinonaktifkan dari segala bentuk pembaruan (*updates*).

2. CLASS II- *Write Once Read Only, Factory or User programmed*. Dalam kasus ini *tag* diproduksi tanpa adanya data yang tertulis di dalam memori. Data dapat ditulis oleh pemanufaktur *tag*, atau oleh pengguna untuk satu kali. Setelah itu *tag* tidak dapat lagi diprogram, tetapi hanya dapat dibaca.

3. CLASS III-*Read Write*

Jenis ini merupakan jenis *tag* yang fleksibel, dimana pengguna mempunyai akses untuk menulis dan membaca data kedalam memori *tag*.

4. CLASS VI-*Read Write with on board sensors*

Tag jenis ini mempunyai sensor *onboard* untuk merekam parameter seperti *temperature*, tekanan udara dan pergerakan, yang dapat direkam dengan menuliskannya kedalam memori *tag*. Pembacaan parameter dilakukan ketika terhubung dengan *reader*, *tag* bisa dari jenis aktif atau semi-pasif.

5. CLASS V-*Read Write with integrated transmitters*.

Jenis *tag* ini seperti *miniature radio*, yang dapat berkomunikasi dengan *tag* dan peralatan lain, tanpa harus adanya *reader*. *Tag* ini berarti aktif dengan *power* dari baterai sendiri.

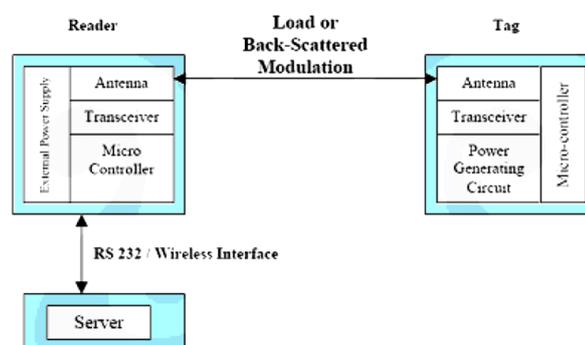
Tabel 2.2 Karakteristik Umum Tag RFID

Jenis	Tag	Tag	Tag
-------	-----	-----	-----

<i>tag</i>	pasif	semipasif	aktif
Catu daya	Eksterna l (dari <i>reader</i>)	Baterai internal	Baterai internal
Rentang baca	Dapat mencapai 20 kaki	Dapat mencapai 100 kaki	Dapat mencapai 750 kaki
Tipe memori	Umumnya <i>read-only</i>	<i>Read-write</i>	<i>Read-write</i>
Usia <i>tag</i>	Dapat mencapai 20 tahun	2 sampai 7 tahun	5 sampai 10 tahun

1.1.2. Komponen Dalam Sistem RFID

Menurut (Fery Suryadi, 2016) Sistem RFID terdapat beberapa komponen penunjang dari sistem tersebut, beberapa komponen tersebut yaitu sistem RFID pasif terdiri dari *tag* tanpa baterai *onboard* dan *reader* yang terhubung dengan *server* melalui konektor data atau *wireless interface* seperti yang ditunjukkan dalam Gambar 2.3. Tetapi, *reader* memiliki *supply power* yang berlimpah, karena disuplai dari sumber eksternal.



Gambar 2.3 Blok diagram pada pasif RFID
(Saputra, 2008)

1.1.3. RFID Reader

Reader merupakan komponen pengidentifikasi pada sistem RFID, dengan teknologi yang digunakan untuk memungkinkan *reader* dalam melacak dan

mengidentifikasi keberadaan *tag*. Sebuah pembaca RFID harus menyelesaikan dua buah tugas, yaitu:

1. Menerima perintah dari software aplikasi
2. Berkomunikasi dengan tag RFID

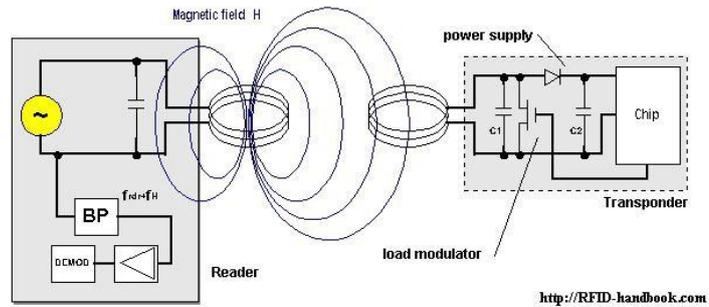
Pembaca RFID juga menjadi penghubung antara *software* aplikasi dengan antenna yang akan meradiasikan gelombang radio ke *tag* RFID. Gelombang radio yang diemisikan oleh antenna berpropagasi pada ruangan di sekitarnya. Akibatnya data dapat berpindah secara *wireless* ke *tag* RFID yang berada berdekatan dengan antenna. Menurut bentuknya, *reader* dapat berupa *reader* bergerak seperti peralatan genggam, atau stasioner seperti peralatan *point-of-sale* di *supermarket*. *Reader* dibedakan berdasarkan kapasitas penyimpanannya, kemampuan pemrosesannya, serta frekuensi yang dapat dibacanya.

1.1.4. Cara Kerja Perpindahan Data Pada RFID *Reader*

Menurut (Saputra, 2008) Perpindahan data terjadi ketika sebuah *tag* didekatkan pada sebuah *reader* dikenal sebagai *coupling*. Perbedaan frekuensi yang digunakan oleh RFID *tag* aktif dengan RFID *tag* pasif menyebabkan perbedaan metode perpindahan data yang digunakan pada kedua *tag* tersebut. Perpindahan data pada RFID *tag* pasif menggunakan metode *magnetic*.

Metode *magnetic* terjadi pada frekuensi rendah. Ketika medan gelombang radio dari *reader* didekati oleh *tag* pasif, koil antenna yang terdapat pada *tag* pasif ini akan membentuk suatu medan magnet. Medan magnet ini akan menginduksi suatu tegangan listrik yang memberi tenaga pada *tag* pasif. Pada saat yang sama terjadi suatu tegangan jatuh pada beban *tag*. Tegangan jatuh ini akan terbaca oleh *reader*. Perubahan tegangan ini berlaku sebagai *amplitude* modulasi untuk bit data.

Ilustrasi untuk *Inductive coupling* digambarkan oleh gambar 2.4.



Gambar 2.4 Cara kerja perpindahan data RFID

(Saputra, 2008)

1.1.5. Frekuensi Radio pada RFID

Menurut (Saputra, 2008) Pemilihan frekuensi radio merupakan kunci karakteristik operasi sistem RFID. Secara umum tingginya frekuensi mengindikasikan jauhnya jarak baca.

Frekuensi yang lebih tinggi mengindikasikan jarak baca yang lebih jauh. Pemilihan tipe frekuensi juga dapat ditentukan oleh tipe aplikasinya. Aplikasi tertentu lebih cocok untuk salah satu tipe frekuensi dibandingkan dengan tipe lainnya karena gelombang radio memiliki perilaku yang berbeda-beda menurut frekuensinya. Contohnya, gelombang LF memiliki kemampuan penetrasi terhadap dinding tembok yang lebih baik dibandingkan dengan gelombang dengan frekuensi yang lebih tinggi, tetapi frekuensi yang lebih tinggi memiliki laju data (*data rate*) yang lebih cepat.

Sistem RFID menggunakan rentang frekuensi yang tak berlisensi dan diklasifikasikan sebagai peralatan *industrial scientific-medical* atau peralatan berjarak pendek (*short-range device*) yang diizinkan oleh *Federal Communications Commission* (FCC). Peralatan yang beroperasi pada bandwidth ini tidak menyebabkan interferensi yang membahayakan dan harus menerima interferensi yang diterima. FCC juga mengatur batas daya spesifik yang berasosiasi dengan masing-masing frekuensi.

Berikut ini adalah empat frekuensi utama yang digunakan oleh sistem RFID:

- a. Band LF berkisar dari 125 KHz hingga 134 KHz. *Band* ini paling sesuai untuk penggunaan jarak pendek (*short-range*) seperti sistem anti pencurian, identifikasi hewan dan sistem kunci mobil.

b. Band HF beroperasi pada 13.56 MHz. Frekuensi ini memungkinkan akurasi yang lebih baik dalam jarak tiga kaki dan karena itu dapat mereduksi risiko kesalahan pembacaan *tag*. Sebagai konsekuensinya *band* ini lebih cocok pada tingkat item pembacaan (*item-level reading*). *Tag* pasif dengan frekuensi 13.56 MHz dapat dibaca dengan laju 10 to 100 *tag* perdetik pada jarak tiga kaki atau kurang. *Tag* RFID HF digunakan untuk pelacakan barang-barang di perpustakaan, toko buku, kontrol akses gedung, pelacakan bagasi pesawat terbang, pelacakan item pakaian.

c. Band UHF beroperasi di sekitar 900 MHz dan dapat dibaca dari jarak yang lebih jauh dari *tag* HF, berkisar dari 3 hingga 15 kaki. *Tag* ini lebih sensitif terhadap faktor-faktor lingkungan daripada *tag-tag* yang beroperasi pada frekuensi lainnya. *Band* 900 MHz muncul sebagai *band* yang lebih disukai untuk aplikasi *supply* disebabkan laju dan rentang bacanya. *Tag* UHF pasif dapat dibaca dengan laju sekitar 100 hingga 1000 *tag* perdetik. *Tag* ini umumnya digunakan pada pelacakan kontainer, truk, trailer, terminal peti kemas.

d. *Tag* yang beroperasi pada frekuensi gelombang mikro, biasanya 2.45 GHz dan 5.8 GHz, mengalami lebih banyak pantulan gelombang radio dari obyek-obyek didekatnya yang dapat mengganggu kemampuan *reader* untuk berkomunikasi dengan *tag*.

lebih jelasnya dapat dilihat pada Tabel 2.3 berikut.

Tabel 2.5 Frekuensi RFID Yang Umum Beroperasi Pada *Tag* Pasif

Gelombang	Frekuensi	Rentang dan laju baca
LF	125 Khz	<ul style="list-style-type: none">• Dapat mencapai jarak ± 30 cm• Kecepatan baca rendah
HF	13,56 Mhz	<ul style="list-style-type: none">• Dapat mencapai jarak ± 90 cm• Kecepatan baca sedang
UHF	860-930 Mhz	<ul style="list-style-type: none">• Dapat mencapai jarak ± 4 Meter• Kecepatan baca tinggi
Gelombang mikro	2,45/5,8 Ghz	<ul style="list-style-type: none">• Dapat mencapai jarak diatas 5 meter• Kecepatan baca tinggi

1.2. MySQL

Kehandalan suatu sistem basis data (DBMS) dapat diketahui dari cara kerja pengoptimasinya dalam melakukan proses perintah-perintah *SQL* yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai penyedia basis data, *MySQL* mendukung operasi basis data transaksional maupun operasi basis data non-transaksional. Pada modus operasi nontransaksional, *MySQL*

dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak penyedia basis data kompetitor lainnya. demikian pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi blogging berbasis Web (*wordpress*), *CMS* dan sejenisnya.

Ada banyak kelebihan Mysql yaitu *source MySQL* dapat diperoleh dengan mudah dan gratis, *MySQL* merupakan program yang multithreading, sehingga dapat dipasang pada server yang memiliki *multiCPU* dan didukung program-program umum seperti *C*, *C++*, *Java*, *PHP*, dsb. Bekerja pada berbagai *platform* (tersedia dalam berbagai versi untuk berbagai sistem operasi), Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem *database*.

1.3. XAMPP

XAMPP dikembangkan dari sebuah tim proyek bernama Apache Friends, yang terdiri dari Tim inti (*Core Team*), Tim Pengembang (*Development Team*) dan Tim Dukungan (*Support Team*).

1.4. PhpMyAdmin

PhpMyAdmin adalah *interface web* yang dibuat untuk mengelola *database MySQL*. phpMyAdmin dibuat menggunakan bahasa PHP dan bersifat *open source*. Dengan phpmyadmin, administrator *web server* bisa mengelola *database* tanpa harus menguasai perintah berbasis baris teks (*command line*) dari SQL (*Structure Query Language*). Phpmyadmin sering digunakan pengembang web untuk menyiapkan database dari aplikasi web seperti CMS, Blog dsb.

Beberapa fitur penting dari phpmyadmin antara lain:

1. Membuat, menghapus dan mengedit baik database, tabel, record, struktur.
2. Membuat pencarian sederhana dan kompleks
3. Inport CSV (bisa digunakan untuk menimport data spreadsheet)
4. Eksport ke CSV, XML, Pdf, spreadsheet.

1.5. Web Server

Server web atau peladen *web* dapat merujuk baik pada perangkat keras ataupun perangkat lunak yang menyediakan layanan akses kepada pengguna melalui protokol komunikasi yang terdapat pada suatu situs *web* dalam layanan ke pengguna dengan menggunakan aplikasi tertentu seperti peramban *web*. Fungsi utama sebuah *server web* adalah untuk mentransfer berkas atas permintaan pengguna melalui protokol komunikasi yang telah ditentukan. Pemanfaatan server web berfungsi pula untuk mentransfer seluruh aspek pemberkasan dalam sebuah halaman *web* yang terkait termasuk di dalamnya teks, gambar, video, atau lainnya.

1.6. Database Server

Database Server adalah sebuah program komputer yang menyediakan layanan pengelolaan basis data dan melayani komputer atau program aplikasi basis data yang menggunakan model klien/server. Istilah ini juga merujuk kepada sebuah komputer (umumnya merupakan server) yang didedikasikan untuk menjalankan program yang bersangkutan.

1.7. Application Programming Interface (API)

API diimplementasikan dengan menulis panggilan fungsi dalam program, yang menyediakan hubungan ke subrutin yang diperlukan untuk eksekusi. Demikian, API menyiratkan bahwa *driver* atau program modul tersedia dalam komputer untuk melakukan operasi atau perangkat lunak yang harus dihubungkan ke dalam program yang ada untuk melakukan tugas-tugas. API dapat menjelaskan cara sebuah tugas (*task*) tertentu dilakukan.

Keuntungan memprogram dengan menggunakan API diantaranya adalah sebagai berikut:

1. Portabilitas. Programmer yang menggunakan API dapat menjalankan programnya dalam sistem operasi mana saja asalkan sudah ter-*install* API tersebut. Sedangkan *system call* berbeda antar sistem operasi, dengan catatan dalam implementasinya mungkin saja berbeda.

2. Lebih Mudah Dimengerti. API menggunakan bahasa yang lebih terstruktur dan mudah dimengerti daripada bahasa *system call*. Hal ini sangat penting dalam hal editing dan pengembangan.

2.8, API TESTING

Testing API dilakukan untuk sistem yang memiliki banyak API untuk di tes. Pengecekan dilakukan dengan mensimulasikan penggunaan API oleh pengguna. Ada beberapa hal yang harus diperhatikan dalam pengecekan API yakni:

1. Menjelajahi kondisi batas dan memastikan bahwa tes memanfaatkan berbagai parameter dari API panggilan dengan cara yang memverifikasi fungsi dan mengekspos kegagalan.
2. Menghasilkan kombinasi nilai parameter yang menarik untuk panggilan dengan dua parameter atau lebih.
3. Memverifikasi perilaku API yang mempertimbangkan kondisi lingkungan eksternal seperti file, perangkat perifer, dan sebagainya
4. Memverifikasi Urutan panggilan API dan memeriksa apakah API menghasilkan hasil yang bermanfaat dari panggilan berturut-turut.

Pada umumnya pengecekan yang dilakukan terhadap API yaitu:

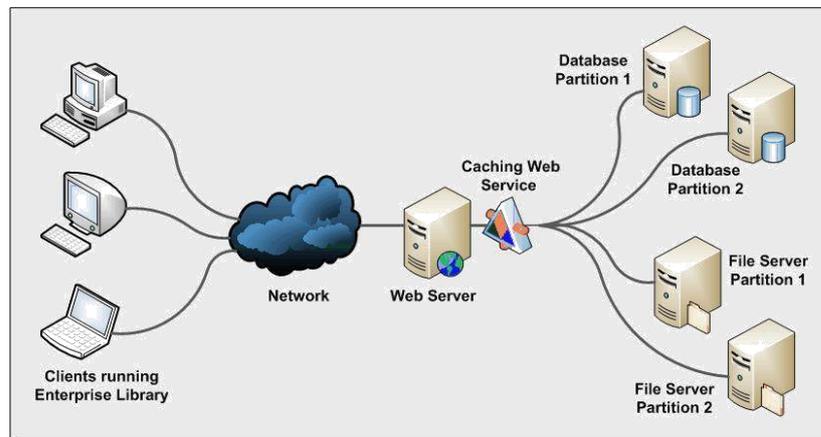
1. Nilai yang diberikan sesuai kondisi yang diberikan.
2. Memastikan bahwa API tidak memberikan nilai apapun selain kondisi
3. Memastikan jika API memicu panggilan terhadap API lain, pengeluaran harus bisa dilacak dan diverifikasi.
4. Memastikan jika API mengubah struktur data

2.9. ANDROID

Saat ini, sistem operasi Android tidak saja berjalan di perangkat *mobile* seperti telepon dan tablet, tetapi juga televisi bahkan jam tangan. Aplikasi Android secara *native* dibangun menggunakan bahasa pemrograman Java, namun pada perkembangannya kita dapat menggunakan bahasa pemrograman lain dalam membuat aplikasi Android.

2.10. WEB SERVICE

Web Service biasanya berupa operasi logika maupun operasi *query* yang dimanfaatkan oleh banyak *client*. *Client* yang dimaksud yaitu oleh *program* atau aplikasi lain, atau bisa juga dimanfaatkan oleh *web service* lainnya.



Gambar 2.5. Skema Sistem Web Service

(sumber: bambangsuhartono.wordpress.com)

Juga mendukung *database* seperti MySQL, PostgreSQL, mSQL, Informix, SQL Server, dan Oracle.

2.11. Flowchart

Flowchart adalah suatu bagan dengan simbol-simbol tertentu yang menggambarkan urutan proses secara mendetail dan hubungan antara suatu proses (instruksi) dengan proses lainnya dalam suatu program (Dharmayanti, 2013). Berikut adalah simbol-simbol flowchart:

Tabel 2.4 simbol-simbol flowchart

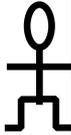
Simbol-Simbol	Keterangan
	Mulai (awal)/akhir (start/end)
	Proses/kegiatan (process/activity)
	Kondisional/keputusan (conditional/decision)

	Arah aliran (flow direction)
	Masukan/keluaran (input/output)

2.12. Use Case Diagram

Use case adalah rangkaian/uraian sekelompok yang saling terkait dan membentuk sistem secara teratur yang dilakukan atau diawasi oleh sebuah aktor. Use case digunakan untuk membentuk tingkah laku benda dalam sebuah mode serta direalisasikan oleh sebuah collaborator, umumnya use case digambarkan dengan sebuah elips dengan garis yang solid, biasanya mengandung nama. Use case menggambarkan proses sistem (kebutuhan sistem dari sudut pandang user). Secara umum use case adalah: pola perilaku sistem dan urutan transaksi yang berhubungan yang dilakukan oleh aktor. Use Case Diagram terdiri dari: use case, aktor, relationship, sistem boundary boxes, packages (Rosa A.S dan M. Shalahudin, 2015). Berikut adalah simbol-simbol use case diagram:

Tabel 2.5 Simbol-Simbol Use Case Diagram

No.	Gambar	Nama	Keterangan
1.		Actor	Menspesifikasikan himpunan pesan yang pengguna mainkan ketika berinteraksi dengan use case.
2.		Include	Menspesifikasikan bahwa use case secara eksplisit.
3.		Association	Apa yang menghubungkan antara objek satu dengan objek lainnya.

.		System	Menspesifikasi paket yang menampilkan sistem sistem secara terbatas.
.		Use Case	Deskripsi dari urutan aksi-aksi yang ditampilkan sitem yang menghasilkan suatu hasil yang terukur bagi suatu aktor.

2.13. PHP

PHP (*Hypertext Preprocessor*) adalah suatu bahasa pemrograman yang berfungsi untuk membangun suatu *website* dinamis (Warman & Saputra, 2012). PHP berjalan pada sisi server sehingga PHP sering juga disebut *server side scripting*. Hal ini berarti bahwa untuk menjalankan *script* PHP harus melalui *web server*.

PHP bersifat *open source* dan mampu berjalan pada sistem operasi apa saja. PHP mendukung beberapa *web server* seperti Apache, IIS, Lighttpd, dan Xitami. Selain itu PHP

juga mendukung *database* seperti MySQL, PostgreSQL, mSQL, Informix, SQL Server, dan Oracle.