

BAB 2

TINJAUAN PUSTAKA

2.1 Sistem Rekomendasi

Sistem rekomendasi dapat didefinisikan sebagai program yang mencoba untuk merekomendasikan *item* yang cocok (produk atau jasa) untuk pengguna tertentu (individu atau bisnis) dengan memprediksi minat pengguna di *item* berdasarkan informasi terkait tentang *item*, pengguna dan interaksi antar *item* dan pengguna [5]. Sistem rekomendasi bertujuan untuk mengurangi informasi yang berlebihan dengan mengambil informasi dan layanan yang paling relevan dari sejumlah besar data.

Tantangan utama sebuah sistem rekomendasi adalah :

1. *Data Sparsity*

Hal ini sangat jarang bahwa dua pengguna menilai item yang sama berkali-kali. Hal ini membuat lebih sulit untuk menghitung kesamaan.

2. *Cold start for user*

Pengguna yang belum melakukan rating terhadap beberapa barang dapat menyebabkan ia mendapatkan hasil yang tidak akurat. Hal ini mungkin terjadi karena pengguna lain yang memiliki kemiripan dengan pengguna ini tidak ditemukan.

3. *Cold start for item*

Item yang baru diperkenalkan memungkinkan tidak mendapatkan rating yang cukup, yang menyebabkan tidak direkomendasikan kepada pengguna.

4. *Attacks*

Sistem rekomendasi memiliki ancaman terhadap serangan, seperti menyalin keseluruhan profil pengguna dan membuat sistem berpikir bahwa si penyerang dan pengguna tersebut sangat mirip. Hal ini membuat penyerang dapat membodohi sistem dan membuatnya menyarankan item apapun yang di rating oleh penyerang kepada pengguna [5]

2.2 Udara Sehat

Kualitas udara pada umumnya dinilai dari konsentrasi parameter pencemaran udara yang terukur lebih tinggi atau lebih rendah dari nilai Baku Mutu Udara Ambien Nasional. Baku mutu udara adalah ukuran batas atau kadar unsur pencemaran udara yang dapat ditenggang keberadaannya dalam udara ambien. Udara ambien adalah udara bebas di permukaan bumi pada lapisan troposfer (lapisan udara setebal 16 km dari permukaan bumi) yang berada di dalam wilayah yurisdiksi Republik Indonesia yang dibutuhkan dan mempengaruhi kesehatan manusia, makhluk hidup dan unsur lingkungan hidup lainnya. Baku mutu udara ambien nasional ditetapkan sebagai batas maksimum mutu udara ambien untuk mencegah terjadinya pencemaran udara sebagaimana terlampir dalam PP No 41 Tahun 1999. Pemerintah menetapkan Baku Mutu Udara Ambien Nasional untuk melindungi kesehatan dan kenyamanan masyarakat. Baku Mutu Udara Ambien Nasional dilihat pada tabel 2.1 [6].

Tabel 2.1 Baku mutu udara ambien nasional menurut PP No 41 tahun 1999

No	Parameter	Waktu	Baku Mutu
1	Aerosol (PM ₁₀)	24 jam	150 µg/m ³
2	Karbonmonoksida (CO)	1 jam	30000 µg/m ³
		24 jam	10000 µg/m ³
3	Ozon (O ₃)	1 jam	235 µg/m ³
		1 tahun	50 µg/m ³
4	Sulfurdioksida (SO ₂)	24 jam	365 µg/m ³
		1 tahun	80 µg/m ³
5	Nitrogendioksida (NO ₂)	1 jam	0.25 µg/m ³
		1 tahun	100 µg/m ³

2.3 ISPA

Infeksi saluran pernapasan akut atau ISPA adalah infeksi di saluran pernapasan, yang menimbulkan gejala batuk, pilek, disertai dengan demam.

ISPA sangat mudah menular dan dapat dialami oleh siapa saja, terutama anak-anak dan lansia. Sesuai dengan namanya, ISPA akan menimbulkan peradangan pada saluran pernapasan, mulai dari hidung hingga paru-paru. Kebanyakan ISPA disebabkan oleh virus, sehingga dapat sembuh dengan sendirinya tanpa pengobatan khusus dan antibiotik. Walaupun demikian, seseorang perlu waspada dan mengetahui kapan saatnya perlu berkonsultasi dengan dokter, serta cara mencegah penyakit ini.

Penyebab ISPA :

Penyebab ISPA adalah virus atau bakteri, yang mudah sekali menular. Penularan virus atau bakteri penyebab ISPA dapat terjadi melalui kontak dengan percikan air liur orang yang terinfeksi. Virus atau bakteri dalam percikan liur akan menyebar melalui udara, masuk ke hidung atau mulut orang lain.

Selain kontak langsung dengan percikan liur penderita, virus juga dapat menyebar melalui sentuhan dengan benda yang terkontaminasi, atau berjabat tangan dengan penderita. [7].

2.4 Xiaomi Air Quality Monitor PM 2.5

Produk terbaru dari Xiaomi membantu Anda untuk mengetahui kualitas udara yang ada di sekitar menggunakan sensor laser yang akurat. Bentuknya yang kecil membuatnya bisa disimpan dimanapun dengan mudah. Dengan kemampuan untuk terhubung ke Xiaomi Air Purifier, produk ini akan menjaga udara yang Anda hirup.

High Definition OLED Display

Xiaomi melengkapi Air Monitor ini dengan OLED Display yang berfungsi untuk menampilkan berbagai informasi untuk Anda. Selain menunjukkan kualitas udara, layar ini dapat menampilkan informasi lain seperti waktu saat itu dan berbagai status perangkat.

Accurate Laser Sensor

Untuk memberikan status kualitas udara yang tepat, Xiaomi melengkapi Air Monitor ini dengan sensor laser yang dapat mengukur ukuran berbagai partikel di udara dengan tepat. Sensor laser ini mampu mengukur partikel hingga berukuran PM2.5 yang biasa merupakan partikel berbahaya [8].

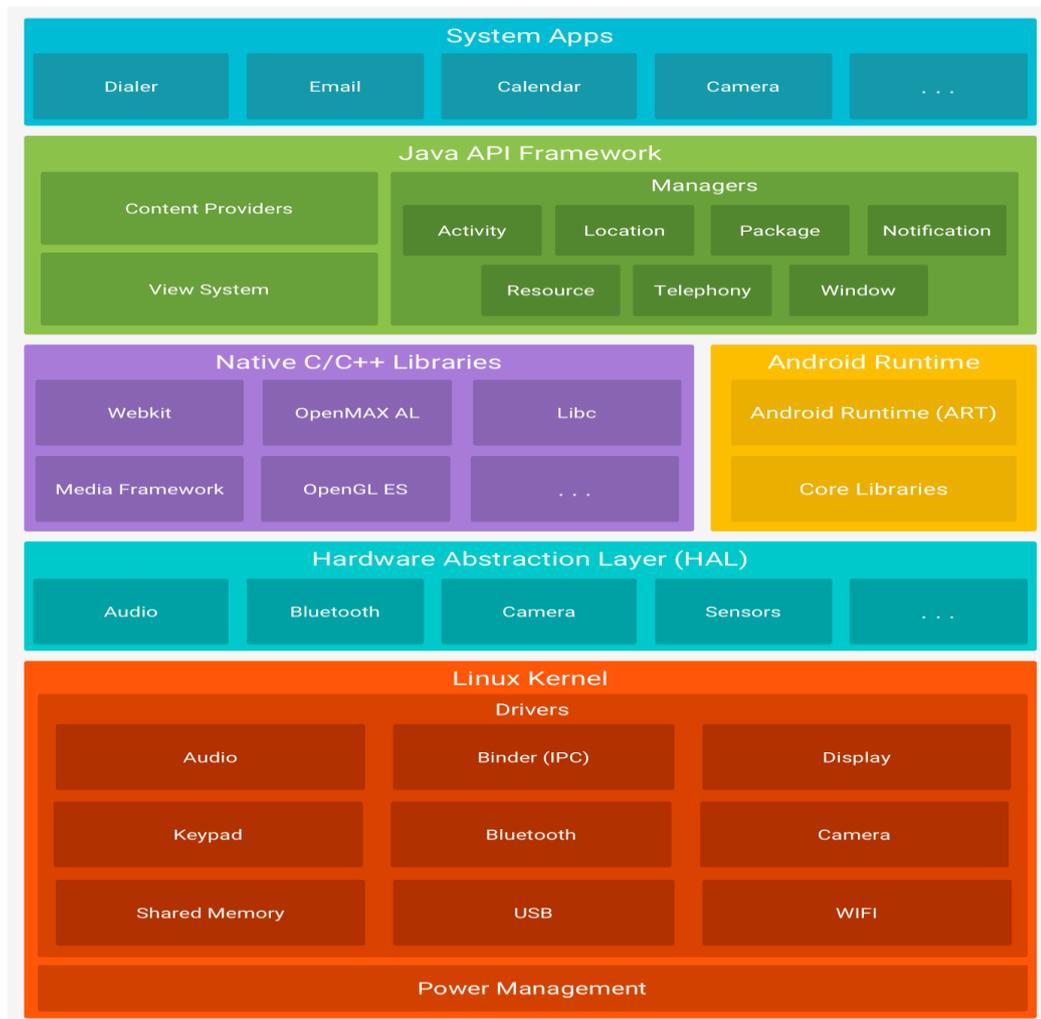
2.5 Android

Dikutip dari buku Android dengan PHP & MySQL karya Akhmad Dharma Kasman [9]. Android merupakan sebuah sistem operasi telepon seluler dan computer tablet layer sentuh (*touch screen*) yang berbasis Linux. Namun seiring perkembangannya, android berubah menjadi platform yang begitu cepat dalam melakukan inovasi. Hal ini tidak lepas dari pengembangan utama dibelakangnya, yaitu Google-lah yang mengakuisisi android, kemudian membuat sebuah platform. Platform android terdiri dari sistem operasi Linux, sebuah GUI (*Graphical User Interface*), sebuah web browser dan aplikasi End-User yang dapat di download dan juga para pengembang bisa dengan leluasa berkarya serta menciptakan aplikasi yang terbaik dan terbuka untuk digunakan oleh berbagai macam perangkat.

Terdapat beberapa penelitian sebelumnya yang berhubungan dengan penelitian ini. Berdasarkan penelitian Eko Budi Setiawan dan Risa Herdianto [10], Android dipilih karena merupakan sistem operasi dengan lisensi open source sehingga dapat dikembangkan secara bebas oleh setiap orang untuk mendukung aktivitas dan pekerjaan sehari-hari.

2.5.1 Arsitektur Aplikasi Berbasis Android

Google mengibaratkan android sebagai sebuah tumpukan software. Setiap lapisan dari tumpukan ini menghimpun beberapa program yang mendukung fungsi-fungsi spesifik dari sistem operasi. Berikut ini susunan dari lapisan-lapisan tersebut jika dilihat dari lapisan dasar hingga lapisan atas [9], yang dapat dilihat pada gambar dibawah ini.



Gambar 2.1 Arsitektur Aplikasi Android

Sumber : <https://developer.android.com/guide/platform>

Berikut merupakan penjelasan dari Gambar di atas:

a. Linux Kernel

Fondasi *platform* Android adalah kernel Linux. Sebagai contoh, Android *Runtime* (ART) bergantung pada kernel Linux untuk fungsionalitas dasar seperti threading dan manajemen memori tingkat rendah.

Menggunakan kernel Linux memungkinkan Android untuk memanfaatkan fitur keamanan inti dan memungkinkan produsen perangkat untuk mengembangkan driver perangkat keras untuk kernel yang cukup dikenal.

b. Hardware Abstraction Layer (HAL)

Hardware Abstraction Layer (HAL) menyediakan antarmuka standar yang mengekspos kemampuan perangkat keras di perangkat ke kerangka kerja Java API yang lebih tinggi. HAL terdiri atas beberapa modul pustaka, masing-masing mengimplementasikan antarmuka untuk komponen perangkat keras tertentu, seperti modul kamera atau *bluetooth*. Bila API kerangka kerja melakukan panggilan untuk mengakses perangkat keras, sistem Android memuat modul pustaka untuk komponen perangkat keras tersebut.

c. Android Runtime

Untuk perangkat yang menjalankan Android versi 5.0 (API level 21) atau yang lebih tinggi, setiap aplikasi menjalankan proses masing-masing dengan tahap Android Runtime (ART). ART ditulis guna menjalankan beberapa mesin virtual pada perangkat bermemori rendah dengan mengeksekusi file DEX, *format* bytecode yang didesain khusus untuk Android yang dioptimalkan untuk footprint memori minimal. Buat rantai aplikasi, misalnya Jack, mengumpulkan sumber Java ke bytecode DEX, yang dapat berjalan pada *platform* Android.

Beberapa fitur utama ART mencakup:

1. Kompilasi mendahului waktu (AOT) dan tepat waktu (JIT).
2. Pengumpulan sampah (GC) yang dioptimalkan.
3. Dukungan debug yang lebih baik, mencakup *profiler* sampling terpisah, pengecualian diagnostik *mendetail* dan laporan kerusakan dan kemampuan untuk mengatur titik pantau guna memantau bidang tertentu.

Sebelum ke Android versi 5.0 (API level 21), Dalvik adalah waktu proses Android. Jika aplikasi Anda berjalan baik pada ART, semestinya berfungsi baik juga pada Dalvik, tetapi mungkin tidak sebaliknya.

Android juga menyertakan serangkaian pustaka waktu proses inti yang menyediakan sebagian besar fungsionalitas bahasa pemrograman Java,

termasuk beberapa fitur bahasa Java 8, yang digunakan kerangka kerja Java API.

d. Pustaka C/C++ Asli

Banyak komponen dan layanan sistem Android inti seperti ART dan HAL dibuat dari kode asli yang memerlukan pustaka asli yang tertulis dalam C dan C++. Platform Android memungkinkan kerangka kerja Java API mengekspos fungsionalitas beberapa pustaka asli pada aplikasi. Misalnya, Anda bisa mengakses OpenGL ES melalui kerangka kerja Java OpenGL API Android guna menambahkan dukungan untuk menggambar dan memanipulasi grafik 2D dan 3D pada aplikasi Anda.

Jika Anda mengembangkan aplikasi yang memerlukan kode C atau C++, Anda bisa menggunakan Android NDK untuk mengakses beberapa pustaka *platform* asli langsung dari kode asli.

e. Kerangka Kerja Java API

Keseluruhan rangkaian fitur pada Android OS tersedia untuk Anda melalui API yang ditulis dalam bahasa Java. API ini membentuk elemen dasar yang Anda perlukan untuk membuat aplikasi Android dengan menyederhanakan penggunaan kembali inti, komponen dan layanan sistem modular, yang menyertakan berikut ini:

1. Tampilan Sistem yang kaya dan luas bisa Anda gunakan untuk membuat UI aplikasi, termasuk daftar, kisi, kotak teks, tombol, dan bahkan browser *web* yang dapat disematkan.
2. Pengelola Sumber Daya, memberikan akses ke sumber daya bukan kode seperti string yang dilokalkan, grafik, dan file layout.
3. Pengelola Notifikasi yang mengaktifkan semua aplikasi guna menampilkan lansiran khusus pada bilah status.
4. Pengelola Aktivitas yang mengelola daur hidup aplikasi dan memberikan back-stack navigasi yang umum.
5. Penyedia Materi yang memungkinkan aplikasi mengakses data dari aplikasi lainnya, seperti aplikasi Kontak, atau untuk berbagi data milik sendiri.

Developer memiliki akses penuh ke API kerangka kerja yang sama dengan yang digunakan oleh aplikasi sistem Android.

f. Aplikasi Sistem

Android dilengkapi dengan serangkaian aplikasi inti untuk email, perpesanan SMS, kalender, menjelajahi internet, kontak, dll. Aplikasi yang disertakan bersama *platform* tidak memiliki status khusus pada aplikasi yang ingin dipasang pengguna. Jadi, aplikasi pihak ketiga dapat menjadi browser *web* utama, pengolah pesan SMS atau bahkan keyboard utama (beberapa pengecualian berlaku, seperti aplikasi Settings sistem).

Aplikasi sistem berfungsi sebagai aplikasi untuk pengguna dan memberikan kemampuan kunci yang dapat diakses oleh developer dari aplikasi mereka sendiri. Misalnya, jika aplikasi Anda ingin mengirimkan pesan SMS, Anda tidak perlu membangun fungsionalitas tersebut sendiri—sebagai gantinya Anda bisa menjalankan aplikasi SMS mana saja yang telah dipasang guna mengirimkan pesan kepada penerima yang anda cantumkan.

2.5.2 Perkembangan Versi OS Android

Keunikan dari nama sistem operasi (OS) android adalah dengan menggunakan nama makanan hidangan penutup (Dessert). Selain itu juga nama-nama OS android memiliki huruf awal berurutan sesuai abjad. Namun juru bicara Google, Randall Sarafa enggan memberi tahu alasannya, Sarafa hanya menyatakan bahwa pemberian nama-nama itu merupakan hasil keputusan internal dan Google memilih tampil sedikit ajaib dalam hal ini.

Tabel 2.2 Perkembangan Versi Android

Code Name	Version	Initial Release Date	API Level
No Codename	1.0	September 23, 2008	1
Petit Four (Only Internal User)	1.1	February 9, 2009	2

Cupcake	1.5	April 27, 2009	3
Donut	1.6	September 15, 2009	4
Éclair	2.0 – 2.1	October 26, 2009	5 – 7
Froyo	2.2 – 2.2.3	May 20, 2010	8
Gingerbeard	2.3 – 2.3.7	December 6, 2010	9 – 10
Honeycomb	3.0 – 3.2.6	February 22, 2011	11 – 13
Ice Cream Sandwich	4.0 – 4.0.4	October 31, 2011	14 – 15
Jelly Bean	4.1 – 4.3.1	July 9, 2012	16 – 18
KitKat	4.4 – 4.4.4	October 31, 2013	19 – 20
Lollipop	5.0 – 5.1.1	November 12, 2014	21 – 22
Marshmallow	6.0 – 6.0.1	October 5, 2015	23
Nougat	7.0	August 22, 2016	24
	7.1	October 4, 2016	25
	7.1.1 - 7.1.2	December 5, 2016	25
Oreo	8.0	August 21, 2017	26
	8.1	December 5, 2017	27
Pie	9.0	August 6, 2018	28
Android 10	10.0	September 3, 2019	29

2.6 Aplikasi

Aplikasi merupakan suatu program yang siap untuk digunakan yang dibuat untuk melaksanakan suatu fungsi bagi pengguna jasa aplikasi serta penggunaan aplikasi lain yang dapat digunakan oleh suatu sasaran yang akan dituju [11]. Menurut kamus computer eksekutif, aplikasi mempunyai arti yaitu pemecahan masalah yang menggunakan salah satu tehnik pemrosesan data aplikasi yang biasanya berpacu pada sebuah komputansi yang diinginkan atau diharapkan maupun pemrosesan data yang di harapkan.

Perkembangan aplikasi saat ini telah tersedia di beberapa *platform*, ada tiga *platform* utama pada pengembangan sebuah aplikasi, yaitu aplikasi berbasis *mobile*, aplikasi berbasis *web*, dan aplikasi berbasis *desktop*.

2.7 Android Studio

Android Studio adalah Lingkungan Pengembangan Terpadu (Integrated Development Environment/IDE) resmi untuk pengembangan aplikasi android, yang didasarkan pada IntelliJ IDEA [12]. Selain sebagai editor kode dan fitur developer IntelliJ yang handal, Android Studio menawarkan banyak fitur yang meningkatkan produktivitas Anda dalam membuat aplikasi android, seperti:

1. Sistem build berbasis *Gradle* yang fleksibel
2. *Emulator* yang cepat dan kaya fitur
3. Lingkungan terpadu tempat Anda bisa mengembangkan aplikasi untuk semua perangkat Android
4. Terapkan Perubahan untuk melakukan *push* pada perubahan kode dan *resource* ke aplikasi yang sedang berjalan tanpa memulai ulang aplikasi
5. *Template* kode dan integrasi GitHub untuk membantu Anda membuat fitur aplikasi umum dan mengimpor kode sampel
6. *Framework* dan fitur pengujian yang lengkap
7. Fitur lint untuk merekam performa, kegunaan, kompatibilitas versi, dan masalah lainnya
8. Dukungan C++ dan NDK

Dukungan bawaan untuk *Google Cloud Platform*, yang memudahkan integrasi *Google Cloud Messaging* dan *App Engine*.

2.8 Java

Java menurut definisi dari Sun adalah mana untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer *stand alone* ataupun pada lingkungan jaringan.

Bahasa java bersifat *case sensitive*, sehingga Anda harus memperhatikan penggunaan huruf besar dan kecil. Selain itu penulisan *source code* program tidak

harus memperhatikan bentuk tertentu, sehingga Anda bisa saja menuliskan semua baris *source code* tersebut dalam satu baris asal Anda tidak lupa membubuhkan tanda titik koma (;), atau menuliskan tiap kata dalam satu baris tersendiri.

Bahasa java memiliki kelemahan dan keunggulan. Kelemahan bahasa java adalah sebagai berikut :

1. Mudah didekompilasi

Dekompilasi adalah proses membalikkan dari kode jadi menjadi kode sumber. Ini dimungkinkan karena kode jadi Java merupakan *bytecode* yang menyimpan banyak atribut bahasa tingkat tinggi, seperti nama-nama kelas, metode, dan tipe data.

2. Penggunaan memori tidak banyak

Penggunaan memori untuk program berbasis java jauh lebih besar daripada bahasa tingkat tinggi generasi sebelumnya seperti C/C++ dan Pascal (lebih spesifik lagi, Delphi dan Object Pascal).

3. Implementasi J2ME tidak global

Misalnya, J2ME untuk Motorola dengan J2ME untuk Sony Ericson tidak sama. Berbeda lagi J2ME untuk Nokia. Setiap produk selalu mempunyai modul tersendiri yang dinilai aneh penerapannya dan harus di-compile dengan modul yang berbeda-beda.

Sedangkan keunggulan bahasa java adalah sebagai berikut :

1. Platform independent

Salah satu keunggulan java adalah sifatnya yang '*platform independence*', artinya Java – baik *source* program maupun hasil kompilasinya – sama sekali tidak bergantung kepada sistem operasi dan *platform* yang digunakan.

2. Sederhana dan berorientasi objek

Java lahir dari suatu pemikiran mendalam akan bahasa pemrograman yang ada pada saat itu, seperti C dan C++. Hal ini akan memudahkan programmer profesional untuk dapat mengerti lebih jelas tentang Java, fungsionalitas, dan lain sebagainya apabila ia memiliki pengetahuan dasar tentang C++ dan konsep pemrograman berorientasi objek.

3. *Automatic garbage collection*

Automatic garbage collection atau pengumpulan sampah otomatis, memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung (seperti halnya dalam bahasa C++) yang dipakai secara luas.

4. Menghilangkan pewarisan berganda yang terdapat pada C++

Walaupun kelihatannya lebih sebagai suatu kekurangan, namun banyak para ahli yang mengakui bahasa konsep pewarisan berganda lebih banyak mengakibatkan kerugian dari pada keuntungan. Java telah didesain sedemikian rupa sehingga Anda tidak akan memerlukan teknik ini dalam pembuatan program apa pun.

5. Mengurangi *pointer* aritmetik

Pengaksesan lokasi memori secara langsung dengan menggunakan *pointer* memungkinkan program untuk melakukan suatu tindakan yang tidak seharusnya atau tidak boleh dilakukan.

6. *Library* yang lengkap

Java terkenal dengan kelengkapan *library*/perpustakaan (kumpulan program program yang disertakan dalam pemrograman Java) yang sangat memudahkan dalam penggunaan oleh para pemrogram untuk membangun aplikasinya

7. Karena OOP jadi sangat cocok bila digunakan untuk membangun program yang besar.

OOP (*Object Oriented Programming* – Pemrogram Berorientasi Objek) yang artinya semua aspek yang terdapat di Java adalah Objek. Java merupakan salah satu bahasa pemrograman berbasis objek secara murni. Semua tipe data diturunkan dari kelas dasar yang disebut *Object*.

8. Bergaya C++

Memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java. Saat ini pengguna Java sangat banyak, sebagian besar adalah pemrogram C++ yang pindah ke Java.

2.9 Java Development Kit (JDK)

Java Development Kit (JDK) adalah sekumpulan perangkat lunak yang dapat digunakan untuk mengembangkan perangkat lunak berbasis Java. Biasanya, setiap JDK berisi satu atau lebih JRE dan berbagai alat pengembangan lain seperti sumber compiler java, buldling, debuggers, development libraries dan lain sebagainya. JRE adalah sebuah implementasi dari *Java Virtual Machine* yang benar-benar digunakan untuk menjalankan program Java [11].

2.10 Web Service

Web Service adalah sekumpulan aplikasi logik beserta objek-objek dan metode-metode yang dimilikinya, yang terletak disuatu server yang terhubung ke internet sehingga dapat diakses menggunakan protokol HTTP (*Hypertext Transfer Protocol*) maupun SOAP (*Simple Object Access Protocol*) [11].

Dalam penggunaannya, *web service* dapat digunakan untuk memeriksa data user yang *login* ke sebuah *website* ataupun untuk digunakan pada transaksi perbankan online yang rumit. Tujuan dari *web service* adalah untuk memudahkan beberapa aplikasi kompenennya untuk saling terhubung dengan aplikasi lain dalam sebuah organisasi maupun diluar organisasi menggunakan standar yang tidak terikat dengan *platform* dan tidak terikat dengan bahasa pemrograman yang digunakan [11].

2.11 MySQL

MySQL adalah suatu perangkat lunak *database* relasi atau *Relational Database Management System* (RDBMS) yang didistribusikan gratis di bawah lisensi GPL (*General Public License*) [13]. MySQL merupakan *database* yang bersifat gratis digunakan oleh siapa saja, akan tetapi tidak diperbolehkan untuk dijadikan produk turunan yang dijadikan closed source atau komersial.

Berikut adalah contoh *source code* pada MySQL :

```
/* Membuat Database */
```

```
CREATE DATABASE peminjaman_buku;
```

```
/* Melihat Database */  
SHOW DATABASES;
```

```
/* Membuat Tabel */  
CREATE TABLE operator(  
    id VARCHAR (20) NOT NULL,  
    nama VARCHAR (50) NOT NULL,  
    password VARCHAR(100) NOT NULL,  
    created_at DATETIME NOT NULL,  
    updated_at TIMESTAMP,  
    PRIMARY KEY (id)  
);
```

```
/* Melihat Tabel */  
SHOW TABLE;
```

```
/* Menghapus Tabel */  
DROP TABLE operator;
```

```
/* Menghapus Database */  
DROP DATABASE peminjaman_buku;
```

2.12 JSON

JSON (*Javascript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generate*) oleh komputer. Format ini dibuat berdasarkan bagian dari Bahasa Pemrograman JavaScript, Standar ECMA-262 Edisi ke-3 - Desember 1999. [14].

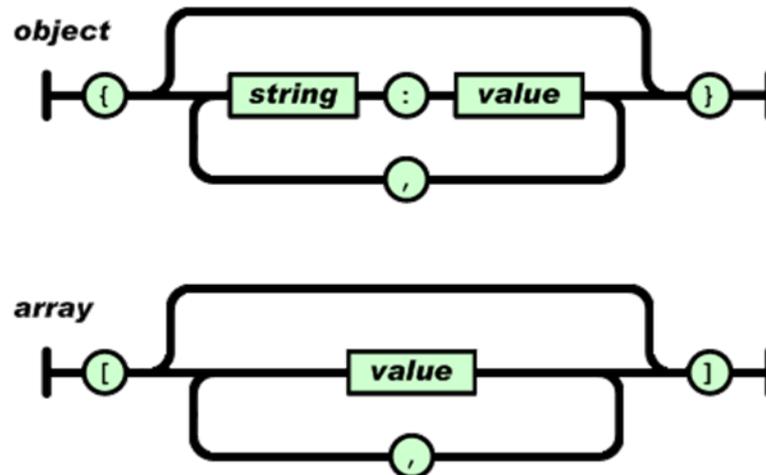
JSON merupakan format teks yang tidak bergantung pada bahasa pemrograman apapun karena menggunakan gaya bahasa yang umum digunakan oleh programmer keluarga C termasuk C, C++, C#, Java, JavaScript, Perl, Python dll. Oleh karena sifat-sifat tersebut, menjadikan JSON ideal sebagai bahasa pertukaran data[10]. JSON terbuat dari dua struktur [14].

4. Kumpulan pasangan nama/nilai. Pada beberapa bahasa, hal ini dinyatakan sebagai objek (*object*), rekaman (*record*), struktur (*struct*), kamus (*dictionary*), tabel hash (*hash table*), daftar berkunci (*keyed list*), atau *associative array*.
5. Daftar nilai terurutkan (*an ordered list of values*). Pada kebanyakan bahasa, hal ini dinyatakan sebagai larik (*array*), vektor (*vector*), daftar (*list*), atau urutan (*sequence*).

JSON menggunakan bentuk sebagai berikut:

1. Objek adalah sepasang nama/nilai yang tidak terurutkan. Objek dimulai dengan { (*kurung kurawal buka*) dan diakhiri dengan } (*kurung kurawal tutup*). Setiap nama diikuti dengan : (*titik dua*) dan setiap pasangan nama/nilai dipisahkan oleh , (*koma*).

Larik adalah kumpulan nilai yang terurutkan. Larik dimulai dengan [(kurung kotak buka) dan diakhiri dengan] (kurung kotak tutup). Setiap nilai dipisahkan oleh , (koma).

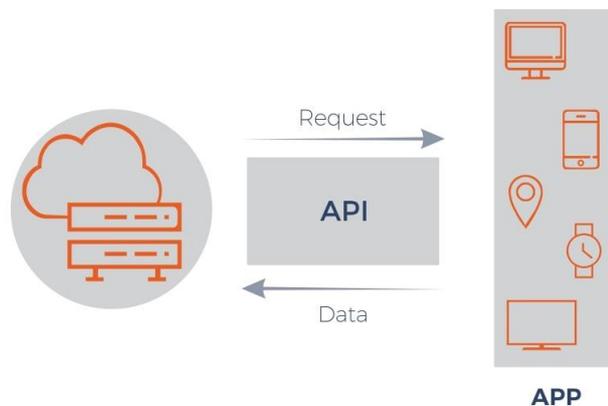


Sumber gambar : Jurnal Ilmiah Komputer dan Informatika (KOMPUTA) [14]

Gambar 2.2 JSON Object dan Array

2.13 Application Programming Interface (API)

Application Programming Interface (API) adalah seperangkat definisi subrutin, protokol, dan alat untuk membangun perangkat lunak aplikasi. API yang bagus memudahkan pengembangan program komputer dengan menyediakan semua blok bangunan, yang kemudian disatukan oleh pemrogram. API mungkin untuk sistem berbasis *web*, sistem operasi, sistem basis data, perangkat keras komputer atau perpustakaan perangkat lunak.



Sumber gambar : <https://qatestlab.com/resources/knowledge-center/application-programming-interface/> [15]

Gambar 2.3 Application Programming Interface (API)

2.14 Google Maps API

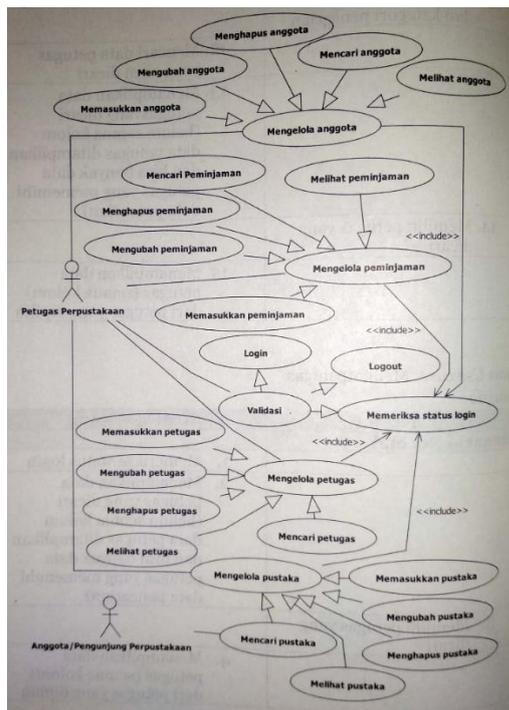
Google Maps adalah layanan pemetaan berbasis *web service* yang disediakan oleh Google dan bersifat gratis, yang memiliki kemampuan terhadap banyak layanan pemetaan berbasis *web*. Google Maps juga memiliki sifat *server side*, yaitu peta yang tersimpan pada *server* Google dapat dimanfaatkan oleh pengguna. Google Maps API adalah suatu library yang berbentuk *Javascript* yang berguna untuk memodifikasi peta yang ada di Google Maps sesuai kebutuhan. Untuk membangun aplikasi yang memanfaatkan Google Maps di desktop dan mobile device maka akan digunakan Google Maps *Javascript* API v3 yang memiliki keunggulan lebih cepat dari versi sebelumnya. [16].

2.15 Unified Modelling Language (UML)

Unified Modelling Language (UML) adalah standar bahasa yang banyak digunakan di dunia industry untuk mendefinisikan *requirement*, membuat analisis & desain, serta menggambarkan arsitektur dalam pemrograman berorientasi objek [4].

2.15.1 Use Case Diagram

Use case diagram merupakan pemodelan untuk kelakuan sistem informasi (*behavior*) sistem informasi yang akan dibuat. *Use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* digunakan untuk mengetahui fungsi apa saja yang ada dalam sebuah sistem dan siapa saja yang berhak menggunakan fungsi-fungsi itu [4].

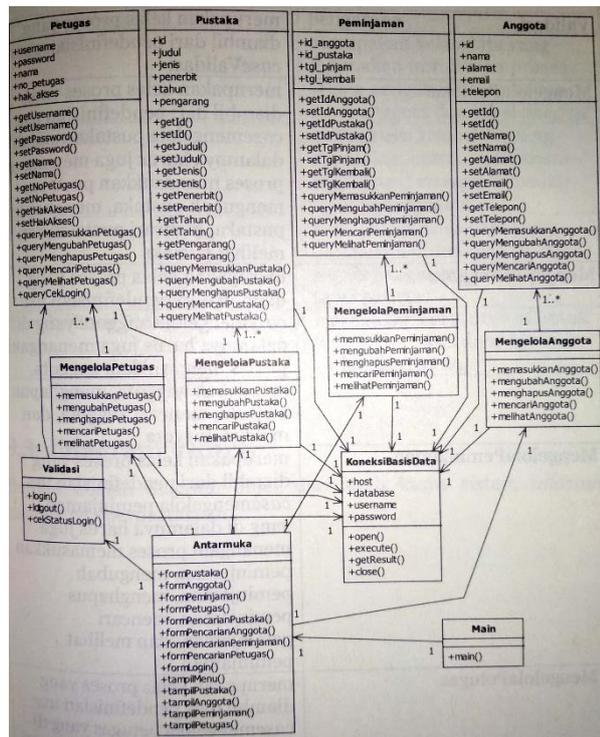


Sumber gambar : *Book Rekayasa Perangkat Lunak (2018)* [4]

Gambar 2.4 Use Case Diagram

2.15.2 Class Diagram

Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem.

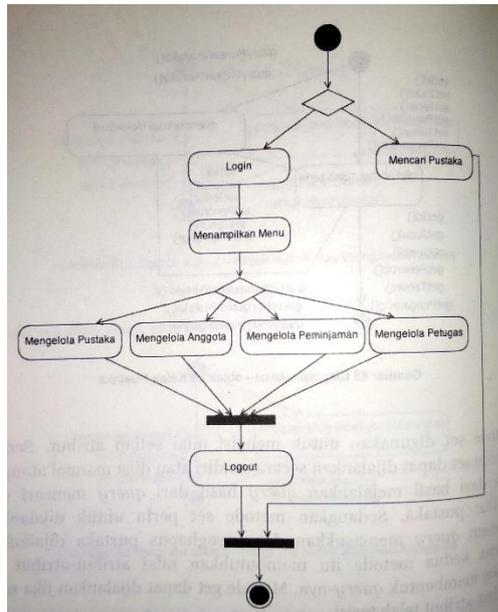


Sumber gambar : Book Rekayasa Perangkat Lunak (2018) [4]

Gambar 2.5 Class Diagram

2.15.3 Activity Diagram

Activity diagram menggambarkan workflow (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan adalah bahwa activity diagram menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem [4].

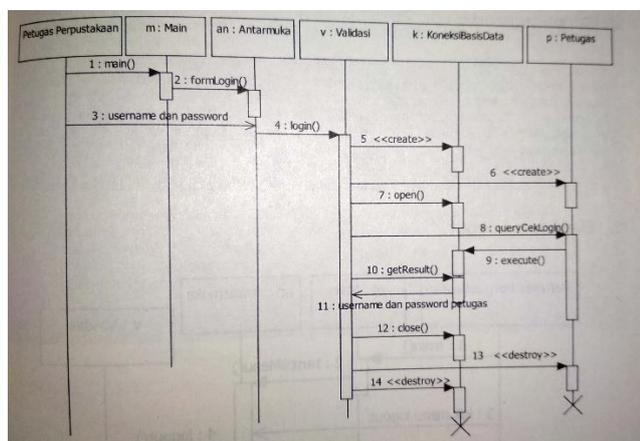


Sumber gambar : *Book Rekayasa Perangkat Lunak (2018)* [4]

Gambar 2.6 Activity Diagram

2.15.4 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada *use case* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antar objek. Untuk menggambarkan *sequence diagram* maka harus diketahui objek – objek yang terlibat dalam sebuah *use case* beserta metode – metode yang dimiliki kelas yang diinstansiasi menjadi objek itu [4].



Sumber gambar : *Book Rekayasa Perangkat Lunak (2018)* [4]

Gambar 2.7 Sequence Diagram

2.16 Pengujian

2.16.1 Blackbox Testing

Blackbox testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak. Tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program. *Blackbox testing* mempunyai ciri-ciri yaitu :

- a. *Blackbox testing* berfokus pada kebutuhan fungsional pada software, berdasarkan pada spesifikasi kebutuhan dari software.

Black box testing melakukan pengujian tanpa pengetahuan detail struktur internal dari sistem atau komponen yang dites. Juga disebut sebagai behavioral testing, specification-based testing, input/output testing atau functional testing [17].

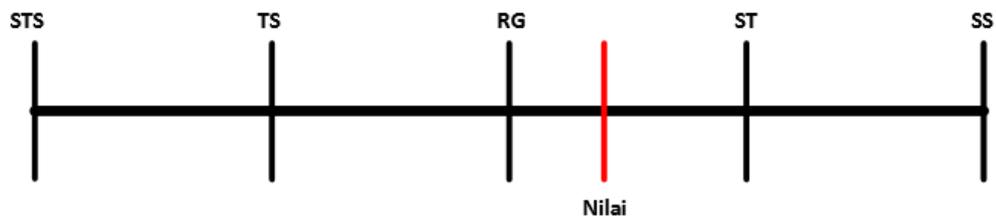
2.16.2 Skala Likert

Skala likert digunakan untuk mengukur sikap, pendapat, dan persepsi seseorang atau sekelompok orang tentang kejadian atau gejala sosial yang telah ditetapkan secara spesifik oleh peneliti (selanjutnya disebut dengan variable penelitian). Dengan menggunakan skala likert, variable yang akan diukur dijabarkan menjadi dimensi. Dimensi lalu dijabarkan menjadi sub variable. Kemudian, subvariable dijabarkan menjadi indikator indikator yang dapat diukur. Akhirnya, indikator-indikator terukur dapat dijadikan titik tolak untuk membuat item instrument yang berupa pertanyaan atau pernyataan yang perlu dijawab oleh responden. Setiap jawaban dihubungkan dengan bentuk pernyataan atau dukungan sikap yang diungkapkan dengan kata-kata sebagai berikut:

- i. Pernyataan sangat positif (5)
- ii. Pernyataan positif (4)
- iii. Pernyataan cukup (3)
- iv. Pernyataan negatif (2)

v. Pernyataan sangat negatif (1)

Dengan data yang direkap dari pengumpulan data angket yang didapatkan data dan di skor perhitungkan dengan cara sebagai berikut. Jumlah penjawab dikalikan dengan pernyataan sangat positif, positif, cukup, negatif dan sangat negatif, lalu hasil dari total nilai yang didapat (diberi contoh a) dari pernyataan dibagikan dengan skor maximal (diberi contoh y) dan dikalikan dengan 100% untuk mendapatkan persentase tingkat persetujuan dari suatu pernyataan. Seperti berikut. $= (a \times y) : 100\% = x\%$, dan hasil dituangkan pada gambar berikut.



Gambar 2.8 Skala Likert

Jadi berdasarkan data yang diperoleh dari x responden maka rata-rata (a) yang berwarna merah terletak pada daerah setuju [18].

