

BAB 2

LANDASAN TEORI

2.1 Landasan Teori

Landasan teori menjelaskan teori dasar yang berhubungan dengan aplikasi yang akan di bangun. Landasan teori yang digunakan dalam penyusunan Aplikasi Rekomendasi Makanan untuk Ibu Hamil meliputi pengertian teori dasar gizi, ibu hamil, aplikasi, android, API, bahasa pemrograman Java, PHP, JSON, MySQL, dan UML.

2.2 Kebutuhan Gizi

Kebutuhan gizi merupakan zat gizi minimal yang diperlukan tubuh untuk bisa mendapatkan energi dan bisa hidup sehat. Energi sendiri dibutuhkan manusia untuk melakukan segala aktivitas fisik sehari-hari dan membantu proses tumbuh kembang khususnya yang masih di masa pertumbuhan. Untuk mengetahui kalori yang dibutuhkan tubuh agar mendapatkan energi yang cukup, bisa diperoleh dengan menggunakan algoritma Harris-Benedict. Istilah untuk mengetahui kalori biasa disebut Angka Metabolisme Basal (AMB). Dalam menentukan AMB ada dua jenis sesuai dengan jenis kelamin, berikut rumusnya :

$$\text{AMB (L)} = 66 + (13,7 \times \text{BB(kg)}) + (5,0 \times \text{TB(cm)}) - (6,8 \times \text{umur(th)}) \dots (\text{II.1})$$

$$\text{AMB (P)} = 655 + (9,6 \times \text{BB(kg)}) + (1,8 \times \text{TB(cm)}) - (4,7 \times \text{umur(th)}) \dots (\text{II.2})$$

Terdapat pengecualian dalam perhitungan kalori yang dibutuhkan untuk ibu hamil. Berat badan yang digunakan adalah berat badan ideal (Rumus Braco) dan memiliki pertambahan kalori sebesar 180 kal pada trimester I dan 300 kal pada trimester II dan trimester III. Berikut rumus yang digunakan untuk menentukan kalori pada ibu hamil :

$$\text{BBI} = (\text{TB} - 100) - (15\% \times (\text{TB} - 100)) \dots (\text{II.3})$$

$$\text{AMB} = 655 + (9,6 \times \text{BBI(kg)}) + (1,8 \times \text{TB(cm)}) - (4,7 \times \text{umur(th)}) \dots (\text{II.4})$$

Setelah mendapatkan AMB, diperlukan Angka Kecukupan Energi atau biasa disebut AKE dengan menalikan AMB dengan faktor aktivitas. Faktor Aktivitas (FA) memiliki 5 aktivitas, antara lain :

1. Sangat ringan, memiliki FA = 1,3
2. Ringan, memiliki FA = 1,56
3. Sedang, memiliki FA = 1,76
4. Berat, memiliki FA = 2,0

Untuk ibu hamil faktor aktivitas yang digunakan adalah kategori sangat ringan. Dikarenakan pada kategori sangat ringan aktivitas yg dilakukan tanpa aktivitas berat dan lebih sering melakukan aktivitas sangat ringan hingga sedang. Berikut rumus untuk mendapatkan Angka Kecukupan Energi :

$$AKE = AMB \times FA \dots (II.5)$$

Terdapat 3 zat gizi dasar untuk bisa memenuhi kebutuhan energi atau biasa disebut energi makro[10], antara lain :

2.2.1 Karbohidrat

Karbohidrat memiliki peran penting dalam memberikan sumber energi bagi manusia, juga mudah didapatkan dan memiliki harga yang relative murah. Umumnya manusia memerlukan karbohidrat sebesar 65% dari kebutuhan kalorinya. Angka energi karbohidrat adalah 4 kalori setiap 1 gram. Untuk mendapatkan Angka Kecukupan Karbohidrat (AKK) bisa didapat dengan rumus :

$$AKK = (65\% \times AKE) / 4 \dots (II.6)$$

2.2.2 Protein

Protein berasal dari kata Yunani “Protos”, yang memiliki arti “yang paling utama”. Dikatakan paling utama karena sangat membantu pertumbuhan dan perbaikan jaringan tubuh, juga sebagai bahan pembentuk zat kimia seperti enzim. Manusia memerlukan sebanyak 15% dari kebutuhan kalorinya dan memiliki angka

energi 4 kalori/gram. Untuk mendapatkan Angka Kebutuhan Protein (AKP) bisa menggunakan rumus :

$$AKP = (15\% \times AKE) / 4 \dots (II.7)$$

2.2.3 Lemak

Lemak berfungsi sebagai bahan baku hormon, pelindung organ tubuh bagian dalam, menghemat serta sebagai insulin terhadap terjadinya perubahan suhu. Lemak yang diperlukan manusia sebanyak 20% dari kebutuhan kalori, dimana 1 gram lemak mengandung 9 kalori. Maka dari itu pengkonsumsian lemak tidak boleh melebihi karbohidrat dan protein. Untuk mendapatkan Angka Kecukupan Lemak (AKL) bisa menggunakan rumus :

$$AKL = (20\% \times AKE) / 9 \dots (II.8)$$

2.3 Rekomendasi

Rekomendasi merupakan saran terhadap apa yang berhubungan dengan keinginan atau kebutuhan dan bersifat menganjurkan atau membenarkan seseorang. Rekomendasi sangat penting untuk meyakinkan orang lain terhadap sesuatu yang tepat dan layak. Ada banyak penerapan rekomendasi pada kehidupan kita sehari-hari. Rekomendasi juga bisa diterapkan pada aplikasi atau sistem yang bisa di kelola secara otomatis. Berikut contoh penerapan rekomendasi pada aplikasi:

1. Rekomendasi barang pada toko online (e-commerce) yang diberikan berdasarkan barang yang sering dicari oleh pengguna,
2. Rekomendasi tempat wisata berdasarkan tempat terdekat atau tempat wisata yang sedang populer,
3. Rekomendasi makanan berdasarkan makanan kesukaan atau tempat makan yang sedang digemari banyak orang, dan lain-lain.

Rekomendasi juga pada sebuah sistem atau aplikasi untuk membantu kebutuhan seseorang dalam melakukan kegiatan sehari-hari. Sistem rekomendasi dapat didefinisikan sebagai program yang mencoba untuk merekomendasikan *item* yang cocok (baik produk atau jasa) untuk pengguna tertentu (individu atau usaha) dengan memprediksi minat pengguna di *item* berdasarkan informasi terkait[11].

Tujuan dari pengembangan sistem rekomendasi adalah untuk mengurangi informasi yang berlebihan dengan mengambil informasi dan layanan yang paling relevan dari sejumlah besar data, sehingga memberikan pelayanan pribadi. Fitur yang paling penting dari sebuah sistem rekomendasi adalah kemampuannya untuk menebak kepentingan dari pengguna dengan menganalisis tinjauan pengguna.

2.4 Tabel Komposisi Pangan Indonesia

Tabel Komposisi Pangan Indonesia atau biasa di singkat TKPI merupakan tabel dari nilai gizi suatu makanan, baik bahan mentah atau yang telah di olah menjadi makanan dengan bumbu. Tabel komposisi pangan merupakan salah satu alat ukur yang sangat penting untuk menyusun menu dan menilai kecukupan asupan konsumsi pangan individu dan kelompok dalam satu wilayah maupun negara baik pada kondisi sehat maupun sakit. Dengan adanya data yang terstandar, masyarakat dapat memiliki cukup informasi dalam memilih dan mengkombinasikan pangan sehat menurut kandungan gizinya.

Hingga saat ini data komposisi pangan di Indonesia masih terus dikembangkan. Pada akhir tahun 2017 telah dihasilkan Tabel Komposisi Pangan Indonesia (TKPI) yang merupakan pengembangan dari TKPI tahun 2009, dengan melengkapi nilai gizi dari TKPI 2009 yang belum memiliki nilai atau masih kosong, menggunakan metode imputed values dan borrowed values yang berasal dari table komposisi bahan pangan negara lain yang serupa. Contoh tabel data komposisi bahan makanan bisa dilihat pada gambar 2.4. Untuk data lengkap berada pada lampiran.

Berikut penjelasan dari data yang ada pada tabel TKPI di atas:

- a. Kolom pertama terdapat kode bahan makanan
- b. Kolom kedua menampilkan nama bahan makanan tunggal atau yang sudah diolah
- c. Data komposisi makanan yang disediakan dengan berat 100gram
- d. Terdapat 21 kolom mengandung nilai gizi yang terdapat pada makanan tersebut
- e. Kolom BDD (berat dapat dimakan) merupakan persentase dari 100gram makanan

2.5 Ibu Hamil

Sangat penting bagi ibu hamil untuk menjaga asupan gizi janin agar tidak kelebihan ataupun kekurangan gizi. Sangat banyak resiko saat masa kehamilan, mulai dari *stunting* hingga menyebabkan kematian kepada si ibu. Pada saat hamil harus memakan makanan yang mengandung nilai gizi bermutu tinggi tapi tidak berarti makanan yang mahal.

Gizi pada waktu kehamilan ibu hamil harusnya mengkonsumsi yang mengandung protein, zat besi, dan minum cukup cairan. Kebutuhan akan nutrisi tersebut meningkat hingga 2-3 kali lipat dikarenakan bertambahnya volume darah hingga 35%. Zat besi juga dapat meningkatkan pertumbuhan bayi, plasenta dan lainnya.

2.5.1. Pemenuhan Gizi Ibu Hamil

Seorang ibu hamil memiliki kebutuhan gizi khusus. Beberapa kebutuhan gizi ibu hamil dapat ditutupi oleh makanan sehat yang seimbang. Selain makanan sehat, pada saat kehamilan dibutuhkan vitamin. Idealnya adalah tiga bulan sebelum kehamilan. Hal ini dapat membantu ibu hamil untuk mendapatkan gizi yang dibutuhkan.

Penambahan kebutuhan zat gizi ibu hamil saat masa kehamilan untuk menyeimbangkan antara kebutuhan janin dan ibu hamil. Penambahan gizi

dilakukan berdasarkan usia kehamilan ibu hamil tersebut. Pada trimester I (1 sampai 3 bulan) terjadi penambah 180 kal, dan pada trimester II dan III (bulan ke 3 sampai bulan ke 9) penambahan sebesar 300 kal.

2.5.2. Pola Makan Ibu Hamil

Pola makan sehat pada ibu hamil dengan menu seimbang sangat penting untuk perkembangan janin dalam kandungan. Tetapi kadang kala calon ibu tidak terlalu memerhatikan hal tersebut. Wanita sebagai calon ibu harus di dorong makan makanan yang banyak mengandung gizi. Karena pada saat masa kehamilan kebutuhan akan kalsium, zat besi, dan asam folat meningkat [12].

Frekuensi makan untuk ibu hamil dalam sehari dibagi menjadi 2 bagian, makan utama dan makan selingan. Baiknya saat masa kehamilan, ibu hamil mengatur frekuensi makan menjadi 3 kali makan utama (pagi, siang, dan malam) dan 2 kali makan selingan (selingan pagi dan selingan sore)[13].

Makanan yang di konsumsi ibu hamil harus memiliki gizi seimbang yang bervariasi. Makan makanan dengan protein tinggi baik dari nabati atau hewani sangat penting saat masa kehamilan, karena sebagai zat pembangun untuk ibu hamil tersebut dan janin. Ibu hamil harus menghindari memakan gorengan semasa kehamilan. Untuk buah dan sayur paling banyak dikonsumsi sekitar 25-30gram per satu hari. Agar bisa memenuhi kebutuhan buah dan sayur per harinya, konsumsi buah dan sayur pada saat makan pagi, siang, malam, maupun pada saat makan selingan.

2.6 Aplikasi

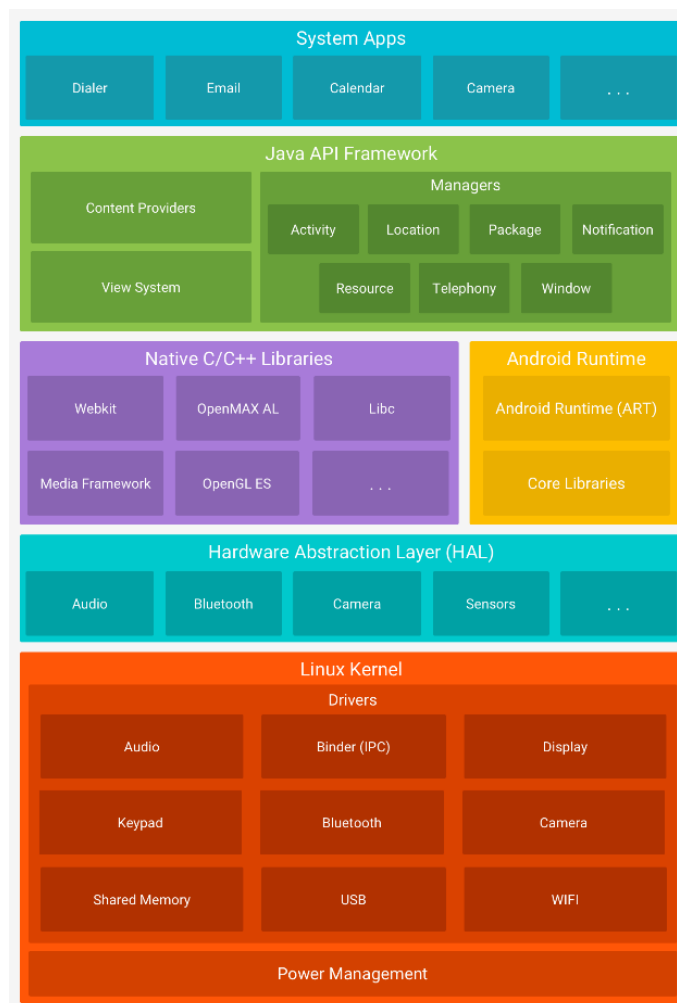
Aplikasi adalah program yang berisikan perintah-perintah untuk mengelolah kumpulan data agar lebih efektif dan efisien[14]. Pada dunia teknologi informasi, aplikasi dibuat untuk mempermudah dan membantu perkerjaan manusia dari cara manual menjadi optimal dengan memanfaatkan komputer. Aplikasi dapat dibuat ataupun dikembangkan dengan maksud untuk melakukan tugas yang bersifat umum atau khusus.

Pada dunia teknologi aplikasi merupakan sebuah perangkat lunak yang dibuat untuk membantu atau mempermudah pekerjaan manusia. Aplikasi juga merupakan suatu program perangkat lunak yang berjalan pada suatu sistem tertentu. Sebuah aplikasi dapat dibuat dan dikembangkan dengan tujuan untuk melakukan tugas yang bersifat umum atau juga dapat dikembangkan untuk melakukan tugas yang bersifat spesifik/khusus.

2.7 Android

Secara istilah aplikasi adalah sistem operasi untuk telepon seluler yang berbasis Linux[15]. Android menyediakan platform terbuka bagi para pengembang untuk menciptakan aplikasi mereka. Dengan begini pengembangan maupun pembuatan Android bisa dipermudah oleh setiap *developer*.

Android memiliki komponen utama yang dapat membuat berjalan dengan baik dan disebut dengan Arsitektur *Platform* Android. Berikut diagram komponen dari Android :



Sumber: <https://developer.android.com/guide/platform/index.html?hl=id>

Gambar 2.2 Arsitektur Platform Android

Adapun penjelasan dari setiap komponen yang ada pada Arsitektur Platform Android antara lain:

2.6.1. *Linux Kernel*

Fondasi platform pada sistem Android adalah Linux kernel. Sebagai contoh, Android Runtime (ART) bergantung pada Linux kernel untuk menjalankan fungsionalitas dasar seperti *threading* dan manajemen memori tingkat rendah. Menggunakan Linux kernel memungkinkan Android untuk memanfaatkan fitur keamanan inti dan memungkinkan produsen perangkat untuk mengembangkan driver perangkat keras untuk kernel yang cukup terkenal.

2.6.2. *Hardware Abstraction Layer*

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.

2.6.3. *Android Runtime*

Perangkat Android 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:

- a Kompilasi mendahului waktu (AOT) dan tepat waktu (JIT)
- b Pengumpulan sampah (GC) yang dioptimalkan
- c 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.

2.6.4. Native C/C++ Libraries

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.

2.6.5. Java API Framework

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:

- a 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
- b Pengelola Sumber Daya, memberikan akses ke sumber daya bukan kode seperti string yang dilokalkan, grafik, dan file layout
- c Pengelola Notifikasi yang mengaktifkan semua aplikasi guna menampilkan lansiran khusus pada bilah status
- d Pengelola Aktivitas yang mengelola daur hidup aplikasi dan memberikan back-stack navigasi yang umum

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.

2.6.6. System Apps

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 tetapkan.

2.8 Versi Android

Versi Android diawali dengan dirilisnya Android beta pada bulan November 2007. Versi komersial pertama, Android 1.0, dirilis pada September 2008. Android dikembangkan secara berkelanjutan oleh Google dan Open Handset Alliance (OHA), yang telah merilis sejumlah pembaruan sistem operasi ini sejak dirilisnya versi awal.

Sejak April 2009, versi Android dikembangkan dengan nama kode yang dinamai berdasarkan makanan pencuci mulut dan panganan manis. Masing-masing versi dirilis sesuai urutan alfabet, yakni Cupcake (1.5), Donut (1.6), Eclair (2.0–2.1), Froyo (2.2–2.2.3), Gingerbread (2.3–2.3.7), Honeycomb (3.0–3.2.6), Ice Cream Sandwich (4.0–4.0.4), Jelly Bean (4.1–4.3), KitKat (4.4+), Lollipop (5.0+), Marshmallow (6.0+), hingga yang terbaru adalah Nougat (7.0+) dan selanjutnya versi android terbaru yang ditunggu-tunggu adalah Android O (8.0+).

Pada tanggal 3 September 2013, Google mengumumkan bahwa sekitar 1 miliar perangkat seluler aktif di seluruh dunia menggunakan OS Android. Berikut adalah versi android dan perubahan fitur berdasarkan tanggal rilis:

1. Android 1.0 (API level 1)

Android 1.0 merupakan versi komersial pertama Android yang dirilis pada 23 September 2008. Awalnya versi 1.0 ini akan dinamai “Astro”, tapi dikarenakan ada permasalahan hak cipta dan trademark nama “Astro” maka penamaan pada versi pertama tersebut dibatalkan oleh Google. Perangkat Android pertama yang tersedia secara komersial adalah HTC Dream.

2. Android 1.2 (API level 2)

Pada 9 Februari 2009, pemutakhiran Android 1.1 dirilis, awalnya hanya untuk HTC Dream. Android 1.1 juga dikenal dengan "Bender", meskipun nama ini tidak digunakan secara resmi dikarenakan masalah yang sama dengan versi pertamanya. Versi ini memperbaiki beberapa bug, mengubah API Android, dan menambahkan beberapa fitur.

3. Android 1.5 Cupcake (API level 3)

Versi ini dirilis pada tanggal 27 April 2009, menggunakan kernel Linux 2.6.27. Mulai dari versi ini, android melakukan penamaan berdasarkan nama pencuci mulut. Karena ini merupakan versi ketiga yang dirilis, sehingga penamaan diawali dari huruf “C”, yaitu “Cupcake”. Pembaruan pada versi ini mulai terlihat dari fitur baru dan perubahan UI. Pada versi ini juga sudah mulai terdapat “widget” yang dapat di rubah ukurannya.

4. Android 1.6 Donut (API level 4)

Pada 15 September 2009, Android 1.6 dirilis. Versi ini dinamai “Donut”, dan menggunakan kernel Linux 2.6.29. Versi ini hadir untuk menutupi bug pada versi sebelumnya sekaligus untuk penambahan beberapa fitur, misalnya dukungan untuk perangkat ukuran layar yang lebih besar.

5. Android 2.0 – 2.1 Eclair (API level 5-7)

Pada 26 Oktober 2009, SDK Android 2.0 dirilis. Versi ini diberi nama Eclair, dan menggunakan kernel Linux 2.6.29. Pada tanggal 12 Januari 2010, versi

2.1 dirilis. Terdapat perubahan UI dan beberapa penambahan fitur ini, seperti *Bluetooth*, *flash* pada kamera, *digital zoom* pada kamera, *multi touch*, *live wallpaper*, dan lainnya.

6. Android 2.2 – 2.2.3 Froyo (API level 8)

Pada 20 Mei 2010, SDK Android 2.2 dirilis dan diberi nama Froyo (singkatan dari *Froze Yoghurt*) menggunakan kernel Linux 2.6.32. Perangkat pertama yang menggunakan versi Android ini adalah Google Nexus One. Pada versi ini sudah mendukung kecepatan dalam membuka dan menutup aplikasi, serta penambahan penyimpanan aplikasi dengan memanfaatkan *SD Card*.

7. Android 2.3 – 2.3.7 Gingerbread (API level 9-10)

Pada tanggal 6 Desember 2010, SDK Android 2.3 (Gingerbread) dirilis, berbasis kernel Linux 2.6.35. Versi 2.3.1 yang rilis desember 2010 dan versi 2.3.2 yang rilis januari 2011 berisi perbaikan bug pada Google Nexus S. Versi 2.3.3 sampai versi 2.3.7 rilis antara bulan februari 2011 sampai bulan september 2011.

Dirilis pada tanggal 6 Desember 2010 bersamaan dengan dihadirkannya Nexus S yang merupakan perangkat smartphone seri Nexus yang diproduksi oleh Samsung. Versi OS ini juga mengawali kesuksesan Android di jagad smartphone meski masih kalah populer dengan BlackBerry OS.

8. Android 3.0 – 3.2 Honeycomb (API level 11-13)

Pada 22 Februari 2011, SDK Android 3.0 (Honeycomb) – pembaruan pertama Android yang ditujukan hanya untuk komputer tablet – dirilis, berdasarkan kernel Linux 2.6.36. Perangkat pertama yang menggunakan versi ini adalah tablet Motorola Xoom, yang dirilis pada 24 Februari 2011. Versi 3.1 rilis tanggal 10 mei 2011. Versi 3.2 rilis tanggal 15 juli 2011. Google TV generasi pertama dan kedua menggunakan Honeycomb.

9. **Android 4.0 – 4.0.4 Ice Cream Sandwich (API level 14 – 15)**

SDK Android 4.0.1 (Ice Cream Sandwich), berdasarkan kernel Linux 3.0.1, dirilis pada 19 Oktober 2011. Petinggi Google, Gabe Cohen, menyatakan bahwa Android 4.0 "secara teoretis kompatibel" dengan perangkat Android 2.3x yang diproduksi pada saat itu. Kode sumber untuk Android 4.0 tersedia pada tanggal 14 November 2011. Pembaruan pada versi ini antara lain:

Ice Cream Sandwich adalah versi terakhir yang mendukung Flash player Adobe System. Versi 4.0.3 rilis tanggal 16 desember 2011. Versi 4.0.4 rilis tanggal 29 maret 2012.

10. **Android 4.1 – 4.3 Jelly Bean (API level 16 – 18)**

Google mengumumkan Android 4.1 (Jelly Bean) dalam konferensi Google I/O pada tanggal 27 Juni 2012. Berdasarkan kernel Linux 3.0.31, Jelly Bean adalah pembaruan penting yang bertujuan untuk meningkatkan fungsi dan kinerja antarmuka pengguna (UI). Penambahan fitur yang diberikan diantaranya meningkatkan input keyboard, desain baru fitur pencarian, UI yang baru dan pencariia melalui *Voice Search* yang lebih cepat. Android 4.1 Jelly Bean dirilis untuk Android Open Source Project pada tanggal 9 Juli 2012. Perangkat pertama yang menggunakan sistem operasi ini adalah tablet Nexus 7, yang dirilis pada 13 Juli 2012.

Google berencana merilis Jelly Bean 4.2 pada sebuah acara di New York City pada 29 Oktober 2012, tapi dibatalkan karena Badai Sandy. Jelly Bean 4.2 didasarkan pada kernel Linux 3.4.0, dan pertama kali digunakan pada Nexus 4 LG dan Nexus 10 Samsung, yang dirilis pada 13 November 2012.

Google merilis Jelly Bean 4.3 pada 24 Juli 2013 di San Francisco. Kebanyakan perangkat Nexus menerima pembaruan dengan segera. Nexus 7 generasi kedua adalah perangkat pertama yang menggunakan sistem operasi ini. Sebuah pembaruan minor dirilis pada tanggal 22 Agustus 2013.

11. Android 4.4 Kitkat (API level 19)

Google mengumumkan Android 4.4 KitKat (dinamai dengan izin dari Nestlé dan Hershey) pada 3 September 2013, dengan tanggal rilis 31 Oktober 2013. Sebelumnya, rilis berikutnya setelah Jelly Bean diperkirakan akan diberi nomor 5.0 dan dinamai 'Key Lime Pie'. Fitur andalan yang diberikan pada Android versi 4.4 ini adalah pengoptimalan perangkat dengan minimum RAM yaitu 512 MB. Dikarenakan untuk menjalankan versi ini dibutuhkan RAM setidaknya 340 MB.

12. Android 5.0 Lollipop (API level 21)

Android 5.0 pertama kali diperkenalkan di bawah codename "Android L" pada 25 Juni 2014 selama presentasi keynote pada konferensi pengembang Google I/O. Di samping Lollipop, presentasi difokuskan pada sejumlah platform Android yang berorientasi dan teknologi baru, termasuk Android TV, pada platform Android Auto, dpt dipakai pada platform komputasi Android Wear, dan platform pelacakan kesehatan Google Fit.

Bagian dari presentasi didedikasikan untuk bahasa desain cross-platform baru yang disebut sebagai "material design". Memperluas pada "kartu" motif pertama kali terlihat di Google Now, adalah desain dengan peningkatan penggunaan layout berbasis grid, animasi dan transisi responsif, padding, dan efek kedalaman seperti pencahayaan dan bayangan.

- Desain antarmuka (tampilan) yang dinamakan "*Material Design*".
- 64-bit ART compiler
- Project volta, yang berguna untuk meningkatkan daya hidup baterai 30% lebih tahan lama.
- '*factory reset protection*'. Fitur ini berguna ketika smartphone hilang, ia tidak bisa direset ulang tanpa memasukkan id google dan kata sandi (password).

13. Android 6.0 Marshmallow (API level 23)

Android 6.0 ini dirilis pada tanggal 25 Juni 2014 dan diberi kode nama Marshmallow pada saat acara Google I/O. Android Marshmallow memberikan dukungan asli untuk pengenalan sidik jari, memungkinkan penggunaan sidik jari untuk membuka perangkat dan otentikasi Play Store dan pembelian Android Pay; API standar juga tersedia untuk melaksanakan otentikasi berbasis sidik jari dalam aplikasi lain. Android Marshmallow mendukung USB Type-C, termasuk kemampuan untuk menginstruksikan perangkat untuk mengisi daya perangkat lain melalui USB. Marshmallow juga memperkenalkan "pranala yang diverifikasi" yang dapat dikonfigurasi untuk membuka langsung dalam aplikasi tertentu mereka tanpa petunjuk pengguna lanjut.

14. Android 7.0 – 7.1 Nougat (API level 24 -25)

Android "Nougat" (kode nama N dalam pengembangan) adalah rilis 7.0 besar dari sistem operasi Android. Ini pertama kali dirilis sebagai pratinjau pengembang pada tanggal 9 Maret 2016, dengan gambar pabrik untuk perangkat Nexus saat ini, serta dengan "Program Beta Beta" baru yang memungkinkan perangkat yang didukung ditingkatkan versinya ke versi Android Nougat melalui over-the-air update. Rilis terakhir adalah pada tanggal 22 Agustus 2016. Pratinjau akhir pembuatannya dirilis pada tanggal 18 Juli 2016, dengan nomor bangunan NPD90G.

Pada tanggal 19 Oktober 2016, Google merilis Android 7.1.1 sebagai pratinjau pengembang untuk Nexus 5X, Nexus 6P dan Pixel C. Pratinjau kedua mulai tersedia pada 22 November 2016, sebelum versi final diluncurkan ke publik pada bulan Desember. 5, 2016.

15. Android 8.0 Oreo (API level 26)

Android Oreo adalah rilis utama ke 8 dari sistem operasi Android. Ini pertama kali dirilis sebagai preview pengembang pada tanggal 21 Maret 2017, dengan gambar pabrik untuk perangkat Nexus dan Pixel saat ini. Pratinjau pengembang terakhir dirilis pada tanggal 24 Juli 2017, dengan rilis stabil yang diharapkan pada bulan Agustus atau September 2017.

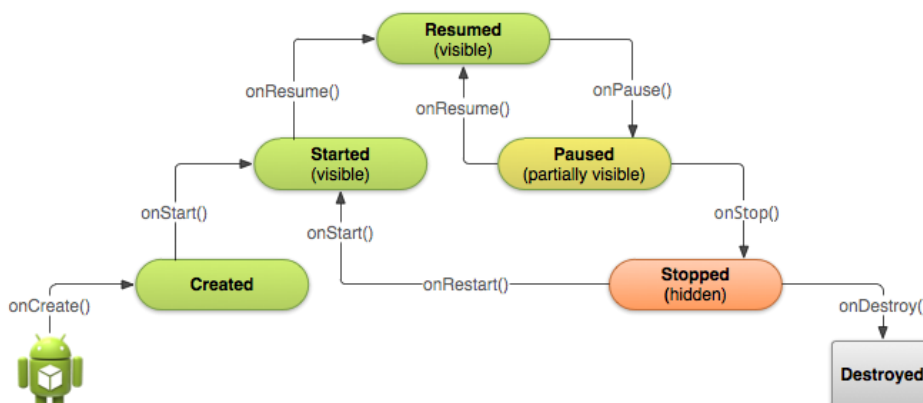
16. Android 9.0 Pie (API level 27)

Android 9.0 resmi dirilis pada Agustus 2018 dengan menggunakan nama kode “Pie”. Pada versi ini Google mengoptimalkan penggunaan baterai, sehingga menjaga ketahanan perangkat untuk bisa digunakan dalam jangka waktu lama. Fitur ini diperkenalkan Google dengan sebutan *adaptive battery*, yang menggunakan pembelajaran mesin untuk meramalkan kapan suatu aplikasi tidak digunakan. Android akan “membekukan” aplikasi pada saat-saat tertentu.

Pada Android Pie, Google secara resmi mengintegrasikan dukungan untuk fitur kamera ganda, yang sebelumnya sudah diperkenalkan pada beberapa model ponsel Android unggulan. Pembuat ponsel akan lebih mudah memberikan dukungan terhadap fitur tersebut dari perangkat lunak.

2.9 Siklus Hidup Android

Di dalam Sistem operasi Android terdapat Siklus hidup dimana pada siklus ini untuk menavigasi antara tahap *Activity Life-Cycle*, Android itu sendiri dengan menyediakan 6 *method* inti *callback* yaitu *onCreate()*, *onStart()*, *onResume()*, *onPause()*, *onStop()*, and *onDestroy()*. Dalam siklus hidup *method callback*, kita dapat mendeklarasikan cara perilaku *activity* saat pengguna meninggalkan dan memasuki kembali ke sebuah *activity* itu. Disitulah peranan *method callback* dalam sebuah aplikasi pada *platform* Android . Berikut pada gambar 2.2 merupakan *Activity Life-Cycle* pada *platform* Android.



Sumber: <https://developer.android.com/training/basics/activity-lifecycle/starting?hl=id>

Gambar 2.3 Siklus Hidup Android

Dari siklus tersebut dapat diketahui bahwa sebuah kelas *Activity* menyediakan enam *method callback*, dan berikut penjelasannya:

1. **onCreate()**

Method ini adalah *method* utama dari setiap *Activity*. *Method* ini akan dipanggil pertama kali ketika menjalankan sebuah sistem. Pengembang harus mengimplementasikan metode *onCreate()* untuk menjalankan logika memulai aplikasi dasar yang hanya boleh terjadi satu kali selama hidup aktivitas. Misalnya, implementasi *onCreate()* Anda harus mendefinisikan antarmuka pengguna dan mungkin membuat instance beberapa variabel dalam cakupan-kelas.

2. **onStart()**

Method ini dipanggil ketika *method onCreate()* telah dipanggil. *onStart()* dipanggil ketika terlihat oleh *user*. *Method* ini selesai dengan cepat dan dilanjutkan dengan *method* setelahnya yaitu *onResume()*.

3. **onResume()**

Method ini dipanggil ketika *method onStart()* selesai dipanggil. *Method* ini adalah keadaan dimana pengguna berinteraksi dengan aplikasi. Aplikasi akan tetap dalam keadaan ini sampai terjadi suatu *statement* dari aplikasi semisal menerima panggilan telepon atau mematikan layar *smartphone*.

4. **onPause()**

Method ini dipanggil ketika pengguna meninggalkan *activity* (meskipun tidak selalu berarti *activity* dihancurkan). *Method* ini berguna untuk menghentikan sementara operasi yang sedang berjalan semisal menjeda pemutaran musik dan lain-lain.

5. **onStop()**

Method ini dipanggil ketika *activity* tidak terlihat lagi oleh pengguna, dengan kata lain *activity* berhenti dijalankan. Hal ini dapat terjadi semisal ada aktivitas baru dijalankan meliputi seluruh layar. Sistem juga dapat menghubungi *method* ini ketika *activity* selesai berjalan, dan akan segera dihentikan.

6. **onResume()**

Method ini adalah *method callback* ketika *activity* telah selesai dijalankan dan kemudian memanggil *method finish()* atau karena sistem untuk sementara menghancurkan proses yang berisi *activity* tersebut untuk menghemat ruang memori.

2.10 API

API adalah *software interface* yang memiliki intruksi dan disimpan dalam bentuk *library* dan dapat terhubung atau berinteraksi dengan *software* lain[16]. Secara struktural, API merupakan spesifikasi dari suatu struktur data, *object*, *function*, beserta parameter-parameter yang diperlukan untuk mengakses *resource* dari aplikasi tersebut. Seluruh spesifikasi tersebut membentuk suatu *interface* yang dimiliki oleh aplikasi untuk berkomunikasi dengan aplikasi lain.

Dengan API, panggilan-panggilan yang bolak-balik antar aplikasi diatur melalui *web service*. *Web service* adalah kumpulan standar teknis dan protokol, termasuk XML (*Extensible Markup Language*), bahasa umum yang digunakan oleh aplikasi-aplikasi tersebut selama berkomunikasi di internet. API dan *web service* sepenuhnya bekerja di belakang layar. Dengan demikian, API menjadi data terbuka milik perusahaan *software* atau perusahaan lainnya yang bisa digunakan untuk pembuatan aplikasi dari layanan yang telah diberikan.

2.11 Web Service

Web service adalah suatu sistem perangkat lunak yang dirancang untuk mendukung interoperabilitas dan interaksi antar sistem pada suatu jaringan. *Web service* digunakan sebagai suatu fasilitas yang disediakan oleh suatu web site untuk menyediakan layanan (dalam bentuk informasi) kepada sistem lain, sehingga sistem lain dapat berinteraksi dengan sistem tersebut melalui layanan-layanan (*service*) yang disediakan oleh suatu sistem yang menyediakan *web service*. *Web service* menyimpan data informasi dalam format XML, sehingga data ini dapat diakses oleh sistem lain walaupun berbeda platform, sistem operasi, maupun bahasa compiler.

Web service bertujuan untuk meningkatkan kolaborasi antar pemrograman dan perusahaan, yang memungkinkan sebuah fungsi di dalam *Web service* dapat di pinjam oleh aplikasi lain tanpa perlu mengetahui detail pemrograman yang terdapat di dalamnya. Beberapa alasan mengapa digunakannya *web service* adalah sebagai berikut :

1. *Web service* dapat digunakan untuk mentransformasikan satu atau beberapa bisnis logic atau class dan objek yang terpisah dalam satu ruang lingkup yang menjadi satu, sehingga tingkat keamanan dapat ditangani dengan baik.
2. *Web service* memiliki kemudahan dalam proses deployment, karena tidak memerlukan registrasi khusus ke dalam suatu sistem operasi. *Web service* cukup di-upload ke web server dan siap diakses oleh pihak-pihak yang telah di berikan otoritas.
3. *Web service* berjalan di port 80 yang merupakan protokol standar HTTP, dengan demikian *web service* tidak memerlukan konfigurasi khusus di sisi *firewall*.

2.12 Firebase

Firebase menyediakan database *realtime* dan *backend* sebagai layanan (*Backend as a Service*). Layanan ini menyediakan pengembang aplikasi API yang memungkinkan aplikasi data yang akan disinkronisasi pada klien dan di simpan di *cloud firebase*. *Firebase* menyediakan *library* untuk berbagai client platform yang

memungkinkan integrasi dengan Android, iOS, JavaScript, Java, Objective-C, dan NodeJs juga disebut sebagai layanan DbaaS (*Database as A Service*) dengan konsep realtime.

Realtime Database menyediakan bahasa aturan berbasis ekspresi yang fleksibel, atau disebut juga Aturan Keamanan Firebase Realtime Database, untuk menentukan metode strukturisasi data dan kapan data dapat dibaca atau ditulis. Ketika diintegrasikan dengan Firebase Authentication, developer dapat menentukan siapa yang memiliki akses ke data tertentu dan bagaimana mereka dapat mengaksesnya.

2.12.1. Firebase Cloud Messaging

FCM adalah sebuah layanan yang digunakan untuk melakukan pemberitahuan (notifications) pada aplikasi berbasis Android, iOS maupun aplikasi web. Dahulunya Firebase Cloud Messaging ini bernama Google Cloud Messaging atau GCM, namun sekarang sudah berubah dan menjadi lebih besar di Firebase. Langkah utama untuk mengimplementasikan FCM di Android adalah membuat project di Firebase dan mengintegrasikannya dengan aplikasi Android[17]. Langkah langkah yang diperlukan adalah :

1. Membuat akun atau project console di Firebase Console, Lalu Create New Project atau buatlah project baru, beri nama sesuai keperluan Anda.
2. Setelah masuk dashboard, lalu carilah tombol Add Firebase to your Android app dan ikuti saja langkahnya (masukan nama namespace dari aplikasi anda, lalu generate dan download file confignya (google-services.json).
3. Letakan file google-services.json tersebut di folder app/ dari project Anda.
4. Jangan lupa tambahkan dependensi pada gradle, lalu sync project anda.

Sejauh ini ada dua metode cara kirim notifikasi. Metode pertama adalah paling simple, mengirim melalui halaman console firebase. Secara sederhana, kita login ke Console Firebase, lalu kita mengirimkan pesan notif melalui fitur yang

sudah tersedia disana. Metode kedua adalah dengan dengan membuat server sebagai pengirim pesan, bahasa pemrogramannya bisa menggunakan php, go lang, python, java ataupun bahasa alien lainnya.

2.13 Google Measure

Google Measure merupakan pengembangan lanjutan dari ARCore. Perangkat mobile apapun yang bisa menjalankan ARCore, maka bisa menjalankan teknologi Google Measure. Google Measure memanfaatkan *Augmented Reality* dan kamera pada perangkat untuk bisa mengukur tinggi ataupun panjang objek.

Google Measure bisa digunakan apabila smartphone tersebut kompatibel untuk menjalankan ARCore. Versi android yang di rekomendasikan untuk menjalankan Google Measure ini adalah versi 7.0 (Nougat), tapi lebih optimal jika memiliki versi 8.0 (Oreo). Namun dikarenakan Google Measure masih dalam pengembangan, jadi masih beberapa perangkat yang bisa menggunakan Google Measure.

Google Measure bisa digunakan sebagai penggaris virtual dengan memanfaatkan *Augmented Reality*. Ada dua tipe pengukuran di Measure: panjang (Length) dan tinggi (Height), sehingga Anda dapat mengukur panjang sebuah karpet, lebar lemari, maupun tinggi meja. Hasil pengukurannya bisa disimpan dalam bentuk foto untuk dijadikan referensi ketika diperlukan, semisal ketika berkunjung ke toko furniture guna berburu perabot baru. Berikut contoh dari gambar 2.5, penggunaan Google Measure untuk mengukur tinggi meja.



Gambar 2.4 Contoh Perhitungan Tinggi Meja

Keakuratan Google Measure ini belum mencapai kata sempurna, dapat dilihat dari ulasan pada Google Play Store. Tapi keakuratannya berbeda sedikit dengan ukuran nyatanya. Google Measure memiliki kekurangan jika memantulkan cahaya, sehingga membuat pengukuran objek menjadi berbeda dengan ukuran nyatanya.

2.14 Java

Java adalah salah satu bahasa pemrograman yang paling populer. Java dapat digunakan untuk berbagai hal, termasuk pengembangan perangkat lunak, aplikasi mobile, dan pengembang sistem yang besar. Inilah yang membuat bahasa pemrograman Java sangat terkenal di lingkungan pengembangan perangkat lunak.

Bahasa pemrograman java banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana. Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (general purpose), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin. Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem operasi yang berbeda, java dikenal pula dengan slogannya, "Tulis sekali, jalankan di mana pun". Saat ini

java merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi. Berikut contoh dari bahasa pemrograman java:

```
Public class HelloWorld () {
    Public static void main (String[] args) {
        System.out.println("Hello World");
    }
}
```

2.15 PHP

PHP (HyperText Preprocessor) adalah sebuah bahasa utama *script server-side* yang disisipkan pada HTML yang dijalankan di server, dan juga bisa digunakan untuk membuat aplikasi dekstop. PHP bersifat *open-source* sehingga dapat dipakai secara cuma-cuma dan mampu lintas platform, dalam artian bisa digunakan pada sistem operasi Windows ataupun Linux[18].

Tentunya bahasa pemrograman PHP berbeda dengan HTML, pada PHP Script/kode yang di buat tidak dapat di tampilkan pada halaman/muka *website* begitu saja, tapi harus diproses terlebih dahulu oleh web server lalu di tampilkan dalam bentuk halaman website di *web browser*, Script PHP juga dapat di sisipkan pada HTML dan script PHP selalu diawali dengan `<?php` dan di akhiri dengan `?>`. Berikut contoh penggunaan PHP :

```
<?php
    echo "Hello World...!!!";
    echo "<br />";
    print "Hello Again World...!!!";
    print "<br /><br />";
?>
```

2.16 JSON

JSON (JavaScript Object Notation) merupakan format pertukaran data yang ringan, mudah dibaca, dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat oleh komputer. Pertukaran data dengan menggunakan format JSON sangat ideal karena format JSON berbasis teks dan bisa terbaca dengan mudah oleh

manusia, serta digunakan untuk merepresentasikan struktur data sederhana dan larik assosiatif (Norwandi dkk,2012).

Sebuah objek JSON adalah format data *key-value* yang biasanya di render di dalam kurung kurawal. Saat kita bekerja dengan JSON, kita akan sering melihat objek JSON disimpan di dalam sebuah file .json, tapi mereka juga dapat disimpan sebagai objek JSON atau string di dalam sebuah program. Penggunaan JSON dapat dilihat pada contoh berikut :

```
{
  "data_diri": {
    "nama_depan" : "Sammy",
    "nama_belakang" : "Ducky",
    "alamat" : "Sekeloa",
    "pekerjaan" : "Mahasiswa"
  }
}
```

Pasangan key-value memiliki tanda titik dua diantara mereka "key" : "value". Setiap key-value dipisahkan oleh sebuah koma, sehingga ditengah isi sebuah JSON terlihat seperti in: "key" : "value", "key" : "value", "key": "value". Pada contoh di atas, nilai pertama pasangan *key-value* kita adalah "data_diri" : "(data berupa nama, alamat, dan pekerjaan)". Lalu dalam value data_diri terdapat *key* baru yaitu "nama_depan" dan memiliki *value* : "Sammy".

2.17 MySQL

MySQL merupakan *software database* yang termasuk paling populer di lingkungan Linux, kepopuleran ini karena ditunjang oleh performansi query dari databasenya yang saat itu bisa dikatakan paling cepat dan jarang bermasalah. Dalam penggunaan *database* MySQL, setiap perintah yang diketikkan disebut *query*. Perintah pada MySQL dapat dikelompokkan menjadi 3 bagian, antara lain:

1. DDL (*Data Definition Language*)

Perintah dalam SQL yang pertama adalah perintah DDL. DDL dapat diartikan sebagai perintah yang berhubungan dengan pendefinisian dari

suatu struktur *database*. Terdapat beberapa perintah DDL pada MySQL sebagai berikut:

- CREATE, berfungsi untuk membuat *database* baru, table baru, *view* baru dan kolom.
- ALTER, berfungsi untuk mengubah struktur tabel. Seperti mengubah kolom, menambah kolom, mengganti nama tabel, memberikan atribut pada kolom maupun menghapus kolom.
- DROP, berfungsi untuk menghapus *database* dan tabel yang telah dibuat.
- TRUNCATE, berfungsi untuk menghapus semua catatan dari tabel.
- COMMENT, berfungsi untuk menambahkan komentar untuk sebagai penjelasan pada data.
- RENAME, berfungsi untuk mengubah nama obyek.

Berikut contoh penggunaan DDL dalam MySQL:

```
CREATE DATABASE mahasiswa;
ALTER TABLE mata_kuliah ADD (thn_ajaran
CHAR(9));
DROP DATABASE ukm;
TRUNCATE TABLE kendaraan;
RENAME nilai TO nilai_akhir;
```

2. DML (*Data Manipulation Language*)

Data Manipulation Language (DML) ialah perintah yang digunakan untuk mengelola/manipulasi data dalam *database*. Terdapat beberapa perintah DML pada MySQL sebagai berikut :

- SELECT berfungsi untuk mengambil/menampilkan data dari *database*.
- INSERT berfungsi untuk memasukkan data ke dalam tabel.
- UPDATE berfungsi untuk memperbarui data dalam tabel.
- DELETE berfungsi untuk menghapus data dari tabel.
- CALL berfungsi untuk memanggil subprogram PL / SQL atau Java.

- EXPLAIN PLAN berfungsi untuk menjelaskan jalur akses ke data.
- LOCK TABLE berfungsi untuk mengunci tabel.

Berikut contoh penggunaan DML pada MySQL :

```
SELECT nim FROM mahasiswa;
INSERT INTO mahasiswa VALUES ("10119001","Sammy
Ducky");
UPDATE mahasiswa SET nama = "Michael Asep"
WHERE nim = "10119001";
DELETE FROM mahasiswa WHERE nim = "10119001";
```

3. DCL (*Data Control Language*)

Data Control Language (DCL) ialah perintah yang digunakan untuk melakukan pengontrolan data dan server databasenya. Terdapat beberapa perintah DCL pada MySQL sebagai berikut :

- GRANT berfungsi untuk memberikan hak akses pengguna ke database.
- REVOKE berfungsi untuk menghilangkan hak akses yang telah diberikan dengan perintah GRANT.

```
GRANT all privileges on * to nm_user@localhost
identified by 'nm_passwd' with grand option;

REVOKE all on nm_db.nm_tbl from nm_user@localhost
identified by 'nm_passwd';
```

2.18 UML

UML (Unified Modelling Language) adalah sebuah bahasa yang telah menjadi standar dalam industri visualisasi, merancang dan mendokumentasikan sistem perangkat lunak. UML dapat dibuat model untuk semua jenis aplikasi perangkat lunak, dimana aplikasi tersebut dapat berjalan pada perangkat keras, sistem operasi dan jaringan apapun, serta ditulis dalam bahasa pemrograman apapun. Tetapi karena UML menggunakan *class* dan *operation* dalam konsep

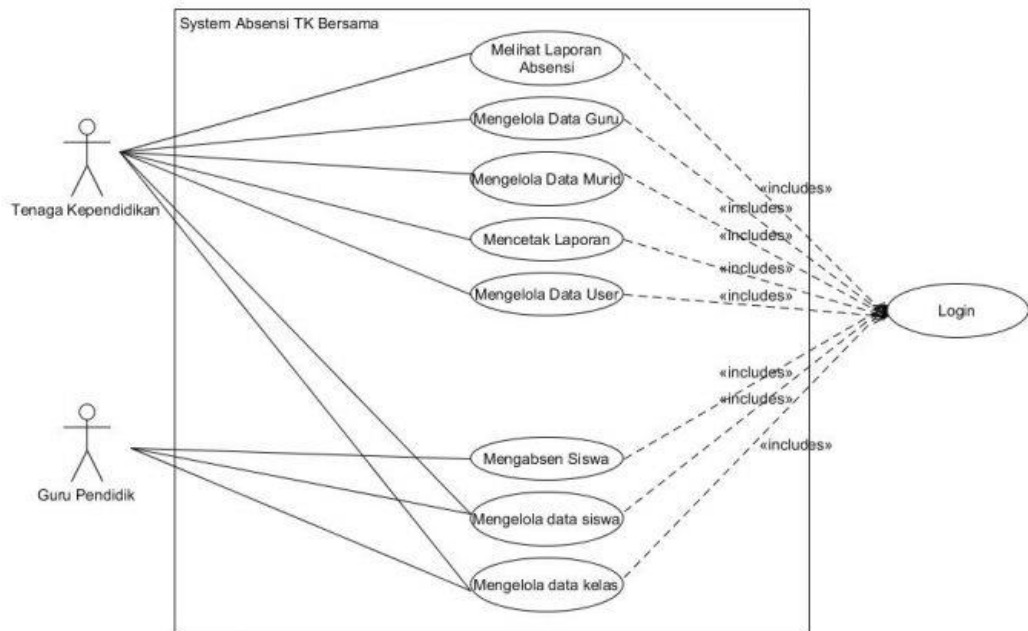
dasarnya, maka lebih cocok untuk penulisan perangkat lunak dalam bahasa berorientasi objek seperti C++, Java, atau VB. NET[19].

Seperti bahasa-bahasa lainnya, UML mendefinisikan notasi dan syntax/semantik. Notasi UML merupakan sekumpulan bentuk khusus untuk menggambarkan berbagai diagram piranti lunak. Setiap bentuk memiliki makna tertentu, dan UML syntax mendefinisikan bagaimana bentuk-bentuk tersebut dapat dikombinasikan. Notasi UML terutama diturunkan dari 3 notasi yang telah ada sebelumnya: Grady Booch OOD (Object-Oriented Design), Jim Rumbaugh OMT (Object Modeling Technique), dan Ivar Jacobson OOSE (Object-Oriented Software Engineering).

2.16.1 Use Case

Use case diagram merupakan bagian tertinggi dari fungsionalitas yang dimiliki sistem yang menggambarkan bagaimana seseorang atau aktor dalam menggunakan dan memanfaatkan sistem. *Use case* terdiri dari tiga bagian yaitu identifikasi aktor, identifikasi *use case*, dan skenario *use case*. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut *use case* dan aktor.

- Use case merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.
- Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu orang.

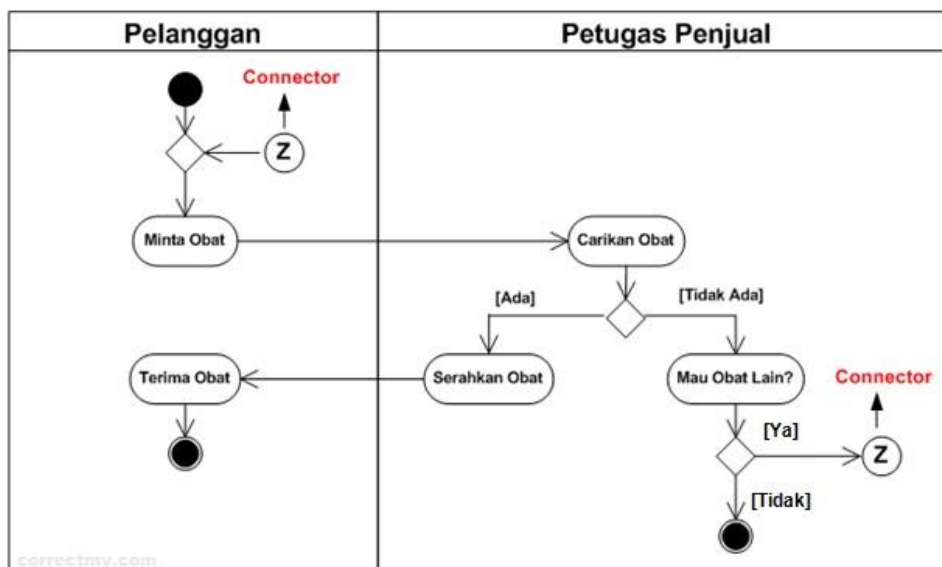


Gambar 2.5 Contoh Use Case Diagram

2.16.2 Activity Diagram

Diagram aktivitas atau activity diagram menggambarkan workflow (aliran kerja) atau aktifitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak. Yang perlu diperhatikan disini adalah bahwa diagram aktifitas menggambarkan aktifitas bukan apa yang dilakukan aktor. Diagram aktifitas juga banyak digunakan untuk mendefinisikan hal-hak berikut:

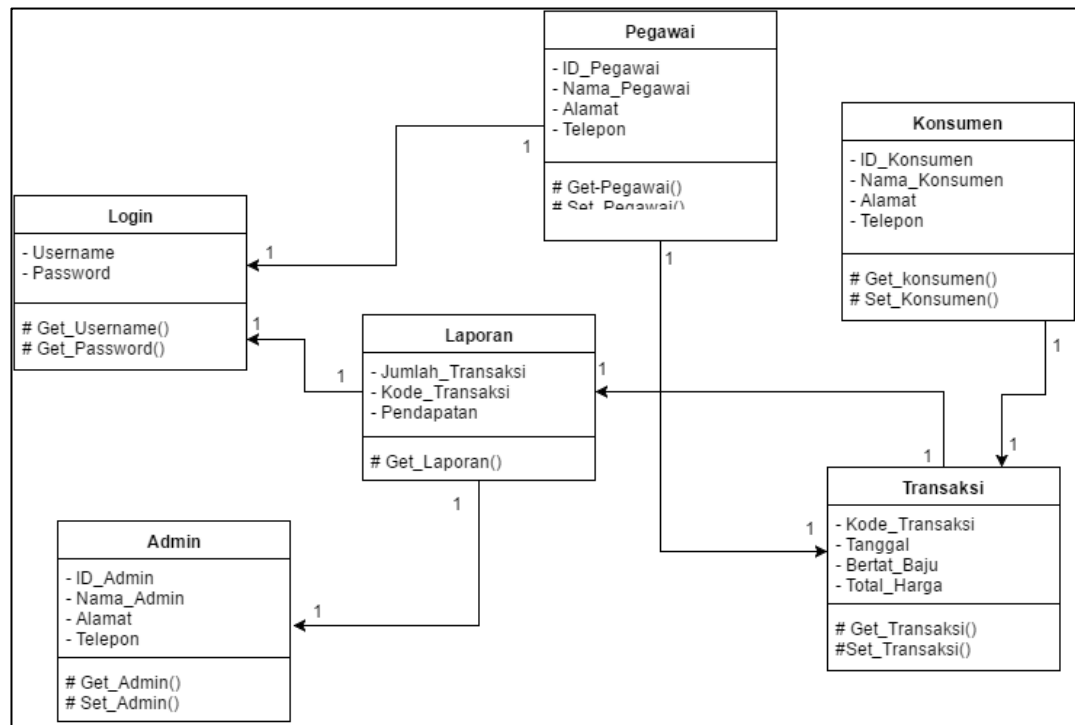
- Rancangan proses bisnis dimana setiap urutan aktifitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- Urutan atau pengelompokan tampilan dari sistem / user interface dimana setiap aktifitas dianggap memiliki sebuah rancangan antarmuka tampilan
- Rancangan pengujian dimana setiap aktifitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya
- Rancangan menu yang ditampilkan pada diagram aktifitas



Gambar 2.6 Contoh Activity Diagram

2.16.3 Class Diagram

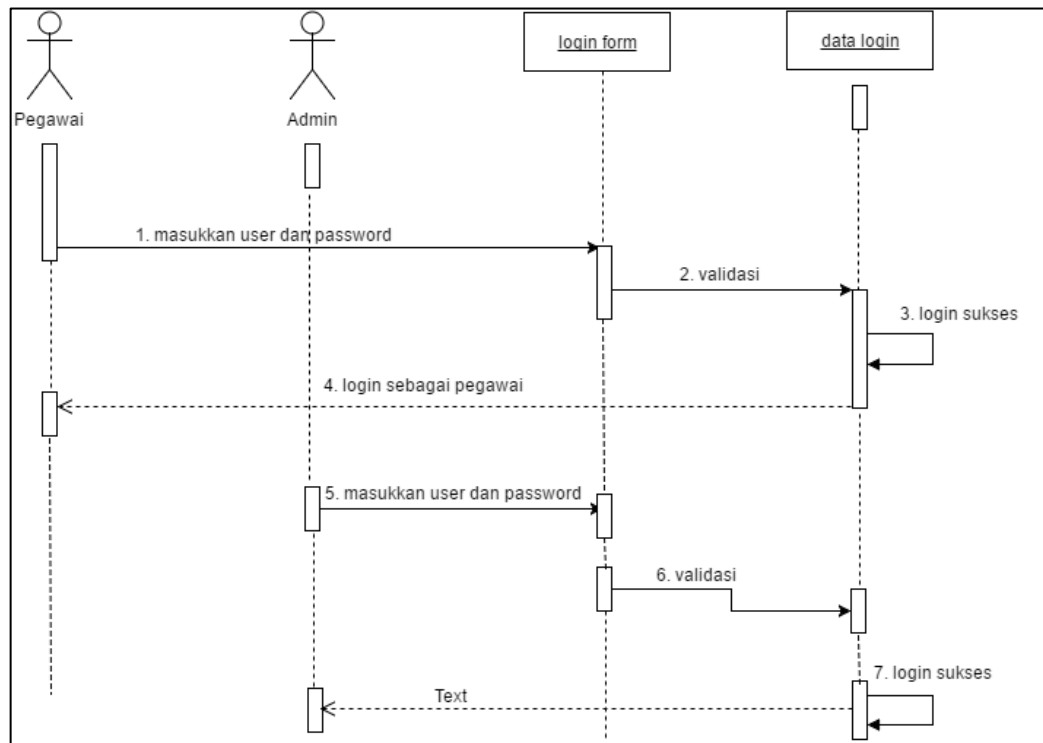
Class diagram membantu dalam visualisasi struktur kelas – kelas dari suatu sistem dan merupakan tipe diagram yang paling banyak. Class diagram memperlihatkan hubungan antar kelas dan penjelasan detail tiap – tiap kelas di dalam model desain (dalam *logical view*) dari suatu sistem. Selama proses analisis, class diagram memperlihatkan aturan – aturan dan tanggung jawab entitas yang menentukan perilaku sistem. Selama proses analisis, class diagram memperlihatkan aturan – aturan dan tanggung jawab entitas yang menentukan perilaku sistem.



Gambar 2.7 Contoh Class Diagram

2.16.4 Sequence Diagram

Sequence diagram menggambarkan kelakuan objek pada use case dengan mendeskripsikan waktu hidup objek dan pesan yang dikirim dan diterima antar objek. Oleh karena itu untuk menggambarkan diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah use case beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu.



Gambar 2.8 Contoh *Sequence Diagram*

