

BAB 2

TINJAUAN PUSTAKA

2.1 Tinjauan Umum Instansi

Tinjauan umum instansi dimaksudkan untuk mengetahui keadaan wilayah dan instansi antara lain profil POLRES Lebak, tujuan jangka menengah POLRES Lebak, logo POLRES Lebak, visi dan misi POLRES Lebak, struktur organisasi POLRES Lebak serta tugas pokok satuan organisasi.

2.1.1 Profil Kepolisian Resor (POLRES) Lebak

Kepolisian Resor Lebak atau Polres Lebak berdiri pada tahun 1948. Polres Lebak merupakan pelaksana tugas Kepolisian Republik Indonesia di wilayah Kabupaten Lebak yang berada di jalan Jl. Siliwangi Km. 1 Blok Cileuweung, Rangkasbitung, Lebak, Banten. Polres Lebak merupakan suatu instansi kepolisian yang berada di bawah jajaran Polda Banten yang memiliki tugas utama untuk memelihara keamanan dan ketertiban, menegakkan hukum, memberikan perlindungan, pengayoman dan pelayanan kepada masyarakat di seluruh wilayah hukum yang menjadi tanggung jawabnya di seluruh wilayah Kabupaten Lebak. Dalam menjalankan tugas-tugas utama yang diembannya maka Polres Lebak dibantu oleh keberadaan satuan kerja yang berada di bawahnya.

2.1.2 Tujuan Jangka Menengah Polres Lebak

- a. Terwujudnya keamanan dan ketertiban masyarakat.
- b. Terwujudnya penegakan hukum yang transparan, akuntabel dan anti KKN.
- c. Terwujudnya perlindungan, pengayoman dan pelayanan prima kepolisian.

2.1.3 Logo Polres Lebak

Setiap instansi pemerintahan mempunyai logo masing-masing untuk membedakan dan menjadi salah satu karakter atau ciri dari suatu instansi. Adapun logo dari Kepolisian Resor (POLRES) Lebak dapat dilihat pada gambar Gambar 2.1.



Sumber gambar : <http://jurnalsumatra.com/>

Gambar 2.1 Logo Kepolisian Resor (POLRES) Lebak

2.1.4 Visi dan Misi Polres Lebak

a. Visi Polres Lebak

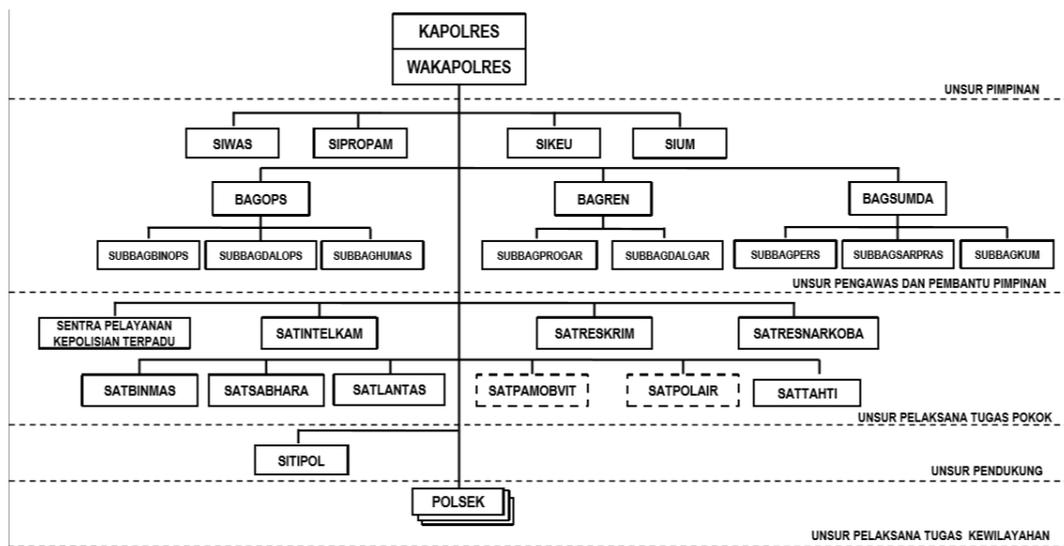
Terwujudnya personel Polres Lebak yang profesional, modern, dan terpercaya.

b. Misi Polres Lebak

1. Menyelenggarakan postur personel Polres Lebak yang ideal, efektif dan efisien.
2. Menyelenggarakan kualitas sumber daya manusia Polres Lebak melalui pendidikan dan pelatihan.
3. Meningkatkan kemampuan pencegahan kejahatan melalui deteksi dini, pemolisian proaktif dan sinergi polisional.
4. Meningkatkan stabilitas Kamtibmas dengan didukung oleh seluruh komponen masyarakat.
5. Menyelenggarakan penegakan hukum yang berkeadilan dan menjamin kepastian hukum dengan menjunjung tinggi hak asasi manusia.
6. Menyelenggarakan pengawasan dalam rangka mewujudkan Polri yang profesional dan akuntabel.

2.1.5 Struktur Organisasi Polres Lebak

Sesuai dengan lampiran peraturan Undang Undang Nomor 43 Tahun 1999 tentang pokok-pokok kepegawaian. Struktur organisasi dapat dilihat pada Gambar 2.2.



Sumber gambar : Peraturan Kapolri nomor 23 tahun 2010 tentang susunan organisasi dan tata kerja pada tingkat polres

Gambar 2.2 Struktur Organisasi Polres (Tipe Polres)

2.1.6 Tugas Pokok Satuan Organisasi

Berikut ini merupakan tugas pokok yang telah ditetapkan untuk setiap bagiannya.

A. Unsur Pimpinan

1. Kapolres

Kapolres merupakan pimpinan Polres yang berada di bawah Polda dan bertanggung jawab kepada Kapolda.

Kapolres bertugas:

- Memimpin, membina, mengawasi, dan mengendalikan satuan organisasi di lingkungan Polres dan unsur pelaksana kewilayahan dalam jajarannya.
- Memberikan saran pertimbangan kepada Kapolda yang terkait dengan pelaksanaan tugasnya.

2. Wakapolres

Wakapolres merupakan unsur pimpinan Polres yang berada di bawah dan bertanggung jawab kepada Kapolres.

Wakapolres bertugas:

- a. Membantu Kapolres dalam melaksanakan tugasnya dengan mengawasi, mengendalikan, mengkoordinir pelaksanaan tugas seluruh satuan organisasi Polres;
- b. Dalam batas kewenangannya memimpin Polres dalam hal Kapolres berhalangan.
- c. Memberikan saran pertimbangan kepada Kapolres dalam hal pengambilan keputusan berkaitan dengan tugas pokok Polres.

B. Unsur Pengawas dan Pembantu Pimpinan

1. Bagops

Bagops merupakan unsur pengawas dan pembantu pimpinan yang berada di bawah Kapolres. Bagops bertugas merencanakan dan mengendalikan administrasi operasi kepolisian, pengamanan kegiatan masyarakat dan/atau instansi pemerintah, menyajikan informasi dan dokumentasi kegiatan Polres serta mengendalikan pengamanan markas. Bagops dipimpin oleh Kabagops yang bertanggung jawab kepada Kapolres, dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres. Bagops dalam melaksanakan tugas dibantu oleh:

1. Subbagian Pembinaan Operasi (Subbagbinops), yang bertugas:
 - a. Menyusun perencanaan operasi dan pelatihan praoperasi serta menyelenggarakan administrasi operasi.
 - b. Melaksanakan koordinasi antar fungsi dan instansi/lembaga terkait dalam rangka pelaksanaan pengamanan kegiatan masyarakat dan/atau pemerintah;
2. Subbagian Pengendalian Operasi (Subbagdalops), yang bertugas:
 - a. Melaksanakan pengendalian operasi dan pengamanan kepolisian.
 - b. Mengumpulkan, mengelola dan menyajikan data dan pelaporan operasi kepolisian serta kegiatan pengamanan.

- c. Mengendalikan pelaksanaan pengamanan markas di lingkungan Polres.
 3. Subbagian Hubungan Masyarakat (Subbaghumas), yang bertugas:
 - a. Mengumpulkan dan mengelola data, serta menyajikan informasi dan dokumentasi kegiatan kepolisian yang berkaitan dengan penyampaian berita di lingkungan Polres.
 - b. Meliput, memantau, memproduksi, dan mendokumentasikan informasi yang berkaitan dengan tugas Polres.
2. Bagren merupakan unsur pengawas dan pembantu pimpinan yang berada di bawah Kapolres. Bagren bertugas menyusun Rencana Kerja (Renja), mengendalikan program dan anggaran, serta menganalisis dan mengevaluasi atas pelaksanaannya, termasuk merencanakan pengembangan satuan kewilayahan. Bagren dipimpin oleh Kabagren yang bertanggung jawab kepada Kapolres, dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres. Dalam melaksanakan tugas dibantu oleh:
1. Subbagian Program dan Anggaran (Subbagprogar), yang bertugas:
 - a. Membantu menyusun rencana jangka sedang dan jangka pendek Polres, antara lain Renstra, Rancangan Renja, dan Renja.
 - b. Membantu menyusun rencana kebutuhan anggaran Polres dalam bentuk RKA-KL, DIPA, penyusunan penetapan kinerja, KAK atau TOR, dan RAB;
 2. Subbagian Pengendalian Anggaran (Subbagdalgar), bertugas:
 - a. Membantu dalam membuat administrasi otorisasi anggaran tingkat Polres.
 - b. Menyusun LRA dan membuat laporan akuntabilitas kinerja Satker dalam bentuk LAKIP meliputi analisis target pencapaian kinerja, program, dan anggaran.
 3. Bagsumda merupakan unsur pengawas dan pembantu pimpinan yang berada di bawah Kapolres. Bagsumda bertugas melaksanakan pembinaan administrasi personel, sarana dan prasarana, pelatihan fungsi, pelayanan kesehatan, bantuan dan penerapan hukum. Bagsumda dipimpin oleh

Kabagsumda, yang bertanggung jawab kepada Kapolres, dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres. Bagsumda dalam melaksanakan tugas dibantu oleh:

1. Subbagian Personel (Subbagpers), yang bertugas melaksanakan pembinaan karier personel, perawatan personel, psikologi personel, pelatihan fungsi, dan pelayanan kesehatan personel Polri di lingkungan Polres.
2. Subbagian Sarana dan Prasarana (Subbagsarpras), yang bertugas melaksanakan inventarisasi, SIMAK BMN, penyaluran perbekalan umum, perawatan alat khusus, senjata api, amunisi dan angkutan, serta memelihara fasilitas jasa dan konstruksi, listrik, air, dan telepon.
3. Subbagian Hukum (Subbagkum), yang bertugas melaksanakan pelayanan bantuan hukum, memberikan pendapat dan saran hukum, penyuluhan hukum, dan pembinaan hukum serta analisis sistem dan metoda terkait dengan ketentuan peraturan perundang-undangan di lingkungan Polres.
4. Siwas merupakan unsur pengawas dan pembantu pimpinan yang berada di bawah Kapolres. Siwas bertugas melaksanakan monitoring dan pengawasan umum baik secara rutin maupun insidentil terhadap pelaksanaan kebijakan pimpinan Polri di bidang pembinaan dan operasional yang dilakukan oleh semua unit kerja, mulai dari proses perencanaan, pelaksanaan, dan pencapaian kinerja serta memberikan saran tindak terhadap penyimpangan yang ditemukan. Siwas dipimpin oleh Kasiwas yang bertanggung jawab kepada Kapolres dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres.
5. Sipropam merupakan unsur pengawas dan pembantu pimpinan yang berada di bawah Kapolres. Sipropam bertugas melaksanakan pembinaan dan pemeliharaan disiplin, pengamanan internal, pelayanan pengaduan masyarakat yang diduga dilakukan oleh anggota Polri dan/atau PNS Polri, melaksanakan sidang disiplin dan/atau kode etik profesi Polri, serta rehabilitasi personel; Sipropam dipimpin oleh Kasipropam yang

bertanggung jawab kepada Kapolres dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres.

6. Sikeu merupakan unsur pembantu pimpinan yang berada di bawah Kapolres. Sikeu bertugas melaksanakan pelayanan fungsi keuangan yang meliputi pembiayaan, pengendalian, pembukuan, akuntansi dan verifikasi, serta pelaporan pertanggungjawaban keuangan. Sikeu dipimpin oleh Kasikeu yang bertanggung jawab kepada Kapolres dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres.
7. Sium merupakan unsur pembantu pimpinan yang berada di bawah Kapolres. Sium bertugas melaksanakan pelayanan administrasi umum dan ketatausahaan serta pelayanan markas di lingkungan Polres

C. Unsur Pelaksana Tugas Pokok

1. SPKT merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. SPKT bertugas memberikan pelayanan kepolisian secara terpadu terhadap laporan/pengaduan masyarakat, memberikan bantuan dan pertolongan, serta memberikan pelayanan informasi.
2. Satintelkam merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satintelkam bertugas menyelenggarakan dan membina fungsi Intelijen bidang keamanan, pelayanan yang berkaitan dengan ijin keramaian umum dan penerbitan SKCK, menerima pemberitahuan kegiatan masyarakat atau kegiatan politik, serta membuat rekomendasi atas permohonan izin pemegang senjata api dan penggunaan bahan peledak.
3. Satreskrim merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satreskrim bertugas melaksanakan penyelidikan, penyidikan, dan pengawasan penyidikan tindak pidana, termasuk fungsi identifikasi dan laboratorium forensik lapangan serta pembinaan, koordinasi dan pengawasan PPNS.
4. Satresnarkoba merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satresnarkoba bertugas melaksanakan pembinaan fungsi penyelidikan, penyidikan, pengawasan penyidikan tindak pidana penyalahgunaan dan peredaran gelap Narkoba berikut prekursorinya, serta

pembinaan dan penyuluhan dalam rangka pencegahan dan rehabilitasi korban penyalahgunaan Narkoba.

5. Satbinmas merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satbinmas bertugas melaksanakan pembinaan masyarakat yang meliputi kegiatan penyuluhan masyarakat, pemberdayaan Perpolisian Masyarakat (Polmas), melaksanakan koordinasi, pengawasan dan pembinaan terhadap bentuk-bentuk pengamanan swakarsa (pam swakarsa), Kepolisian Khusus (Polsus), serta kegiatan kerja sama dengan organisasi, lembaga, instansi, dan/atau tokoh masyarakat guna peningkatan kesadaran dan ketaatan masyarakat terhadap hukum dan ketentuan peraturan perundang-undangan serta terpeliharanya keamanan dan ketertiban masyarakat.
6. Satsabhara merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satsabhara bertugas melaksanakan Turjawali dan pengamanan kegiatan masyarakat dan instansi pemerintah, objek vital, TPTKP, penanganan Tipiring, dan pengendalian massa dalam rangka pemeliharaan keamanan dan ketertiban masyarakat serta pengamanan markas.
7. Satlantas merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satlantas bertugas melaksanakan Turjawali lalu lintas, pendidikan masyarakat lalu lintas (Dikmaslantas), pelayanan registrasi dan identifikasi kendaraan bermotor dan pengemudi, penyidikan kecelakaan lalu lintas dan penegakan hukum di bidang lalu lintas. Satlantas dipimpin oleh Kasatlantas yang bertanggung jawab kepada Kapolres dan dalam pelaksanaan tugas sehari-hari di bawah kendali Wakapolres.
8. Satpamobvit merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satpamobvit bertugas melaksanakan kegiatan pengamanan objek vital (Pamobvit) yang meliputi proyek/instalasi vital, objek wisata, kawasan tertentu, dan VIP yang memerlukan pengamanan kepolisian.
9. Satpolair merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Satpolair bertugas melaksanakan fungsi kepolisian perairan, yang

meliputi patroli perairan, penegakan hukum di perairan, pembinaan masyarakat pantai dan perairan lainnya, serta SAR.

10. Sattahti merupakan unsur pelaksana tugas pokok yang berada di bawah Kapolres. Sattahti bertugas menyelenggarakan perawatan tahanan meliputi pelayanan kesehatan tahanan, pembinaan tahanan serta menerima, menyimpan, dan mengamankan barang bukti beserta administrasinya di lingkungan Polres, melaporkan jumlah dan kondisi tahanan sesuai dengan ketentuan peraturan perundang-undangan.

D. Unsur Pendukung

Sitipol merupakan unsur pendukung yang berada di bawah Kapolres. Sitipol bertugas menyelenggarakan pelayanan teknologi komunikasi dan informasi, meliputi kegiatan komunikasi kepolisian, pengumpulan dan pengolahan serta penyajian data, termasuk informasi kriminal dan pelayanan multimedia.

2.2 Landasan Teori

Landasan teori menjelaskan seperangkat definisi, konsep serta proposisi yang telah disusun rapi dan sistematis yang berkaitan dengan penelitian skripsi sebagai dasar pemahaman dalam sebuah aplikasi yang dibangun serta metode yang digunakan untuk kegiatan pembangunan aplikasi tersebut. Berikut penjelasannya.

2.2.1 Aplikasi Mobile

Aplikasi mobile adalah aplikasi yang telah dirancang khusus untuk platform mobile (misalnya iOS, android, atau windows mobile). Dalam banyak kasus, aplikasi mobile memiliki user interface dengan mekanisme interaksi unik yang disediakan oleh platform mobile, interoperabilitas dengan sumber daya berbasis web yang menyediakan akses ke beragam informasi yang relevan dengan aplikasi, dan kemampuan pemrosesan lokal untuk pengumpulan, analisis, dan format informasi dengan cara yang paling cocok untuk platform mobile. Selain itu aplikasi mobile menyediakan kemampuan penyimpanan persisten dalam platform[10].

Dalam penelitian ini akan membangun aplikasi perangkat mobile berbasis android. Alasan membangun aplikasi mobile yaitu karena aplikasi mobile lebih mudah digunakan serta lebih nyaman digunakan.

2.2.2 Kriminalitas

Kriminalitas merupakan segala macam bentuk tindakan dan perbuatan yang merugikan secara ekonomis dan psikologis yang melanggar hukum yang berlaku dalam negara Indonesia serta norma-norma sosial dan agama. Dapat diartikan bahwa, tindak kriminalitas adalah segala sesuatu perbuatan yang melanggar hukum dan melanggar norma-norma sosial, sehingga masyarakat menentangnya[11].

Sehingga kriminalitas merupakan suatu tindakan dan perbuatan yang melanggar hukum serta dapat merugikan korban dan masyarakat seperti hilangnya ketentraman dan ketertiban. Salah satu bentuk kejahatan yang akhir – akhir ini sering beredar dimedia adalah kejahatan jalanan atau *street crime*.

Street crime atau kejahatan jalanan merupakan kejahatan yang terjadi di jalanan antara lain pencurian dengan kekerasan (begal), pemerasan, dan kejahatan para preman. *Street crime* merupakan hal yang masih sulit diatasi oleh negara-negara berkembang, seperti yang terjadi di Indonesia. Hal ini dilatar belakangi oleh kurangnya lapangan pekerjaan sehingga menyebabkan banyaknya pengangguran yang masih memadati sudut-sudut wilayah tanah air. Hal ini yang menyebabkan maraknya kriminalitas yang masih sulit diatasi karena kebanyakan dilakukan oleh orang yang memiliki tingkat pendidikan dan ekonomi yang rendah.

Kejahatan jalanan merupakan problematika sosial yang berawal dari sikap mental masyarakat yang kurang siap menerima pekerjaan yang dianggap kurang bergengsi. Preman di Indonesia sudah ada sejak jaman penjajahan, kolonial Belanda, selain bertindak sendiri, para pelaku preman juga telah memanfaatkan beberapa jawara lokal untuk melakukan tindakan preman tingkat bawah yang pada umumnya melakukan kejahatan jalanan (*street crime*) seperti pencurian dengan ancaman kekerasan atau biasa disebut dengan begal.

Menurut Kamus Besar Bahasa Indonesia (KBBI), arti kata begal yaitu penyamun, atau perampasan dijalan yang sering terjadi sehingga penduduk di daerah tersebut tidak berani memakai perhiasan[12]. Pembegalan adalah suatu tindakan perampasan di jalanan dengan menghentikan pengendaranya secara paksa. Pembegalan banyak terjadi di jalanan yang sepi dan beroperasi pada waktu tertentu.

Adapun hukum yang mengatur Pencurian dengan kekerasan (begal) diatur dalam (Pasal 365 KUHP), pemerasan (Pasal 368 KUHP), pemerkosaan atau rape (Pasal 285 KUHP), penganiayaan (Pasal 351 KUHP), merusakkan barang (Pasal 406 KUHP) yang tentunya dapat mengganggu ketertiban umum serta menimbulkan keresahan di masyarakat.

2.2.3 Mendeteksi

Deteksi adalah proses untuk melakukan pemeriksaan terhadap suatu objek menggunakan cara tertentu. Deteksi digunakan pada masalah tertentu misalnya dalam sebuah sistem yang mampu melakukan pendeteksian terhadap penyakit. Pada sistem deteksi penyakit sistem akan mengidentifikasi faktor-faktor dari masalah yang ada hubungannya dengan penyakit, biasanya muncul dalam bentuk gejala tertentu. Tujuan melakukan deteksi yaitu melakukan pemecahan masalah menggunakan cara tertentu tergantung dari pendekatan yang digunakan sehingga masalah dapat dipecahkan dan menghasilkan sebuah solusi.

2.2.4 Sistem

A. Definisi Sistem

Menurut Fat pengertian sistem adalah sebagai berikut: "Sistem adalah suatu himpunan suatu "benda" nyata abstrak (*a set of thing*) yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan (*Unity*) untuk mencapai tujuan tertentu secara efisien dan efektif.

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu. Pendekatan sistem yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam sistem[13].

B. Karakteristik Sistem

Sistem yang baik menurut Jeperson Hutahaean yaitu sistem yang memiliki karakteristik seperti berikut.

1. Komponen

Suatu sistem terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling berkerja sama membentuk satu kesatuan keomponen sistem terdiri dari komponen yang berupa subsistem atau bagian-bagian dari sistem.

2. Batasan sistem (boundary)

Batasan sistem merupakan daerah yang membatasi antara uatu sistem dengan sistem yang lain atau dengan lingkungan luarnya. Batasan sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batasan suatu sistem menunjukan ruang lingkup (scope) dari sistem tersebut.

3. Lingkungan luar sistem (environment)

Lingkungan luar sistem adalah diluar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan dapat bersifat menguntungkan yang harus tetap dijaga dan yang merugikan harus dijaga dan dikendalikan, jika tidak akan mengganggu kelangsungan hidup dari sistem.

4. Penghubung sistem (interface)

Penghubung sistem merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya yang mengalir dari subsistem ke subsistem lainnya. Keluaran (output) dari subsistem akan menjad masukan (input) untuk subsistem lain melalui penghubung.

5. Masukan Sistem (input)

Masukan adalah energi yang dimasukan kedalam sistem, yang dapat berupa perawatan (maintenance input) dan masukan sinyal (signal input). Maintenance input adalah energi yang dimasukan agar sistem dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran. Contoh dalam sistem computer program adalah maintenance input sedangkan data adalah signal input untuk diolah menjadi informasi.

6. Keluaran sistem (output)

Keluaran sistem adalah hasil dari energi yang diolah dan diklasiikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh komputer

menghasilkan panas yang merupakan sisa pembuangan, sedangkan informasi adalah keluaran yang dibutuhkan.

7. Pengolah Sistem

Suatu sistem menjadi bagiarrrn pengolah yang akan mengubah masukan menjadi keluaran. SIstem produksi akan mengelola bahan baku menjadi bahan jadi, system akuntansi akan mengelola data menjadi laporan-laporan keuangan.

8. Sasaran Sistem

Suatu sistem pasti mempunyai tujuan (goal) atau sasaran (objective). Sasaran darisistem sangat menentukan input yang dibutuhkan sistem dan keluaran yang aka dihasilkan sistem.

2.2.5 Android

Android merupakan suatu sistem operasi *mobile* yang berbasis pada sistem operasi Linux. Android menawarkan pendekatan yang menyeluruh dalam pengembangan aplikasi. Artinya, satu aplikasi android yang dibangun dapat berjalan diberbagai perangkat yang menggunakan sistem operasi Android baik itu *smatphone, smartwatch, tablet*, dan perangkat lainnya[14].

1. Versi – versi Android

Sistem aplikasi Android terdiri dari beberapa versi. Setiap versi Android terbaru memiliki beberapa jenis kelebihan mulai dari tampilan hingga optimasi keamanan. Berikut daftar nama OS Android menurut versi[15].

1. Android versi 1.0 dirilis tanggal 23 September 2008

Pertama kali dirilis pada 23 Spetember 2008. Sebenarnya Android versi pertama ini akan dinamai dengan nama “Astro” tapi karena alasan hak cipta dan trademark nama”Astro” tidak jadi disematkan pada versi pertama dari OS Android ini. HTC Dream adalah ponsel pertama yang menggunakan OS ini.

2. Android versi 1.1 dirilis tanggal 09 Februari 2009

Pertama kali dirilis pada 9 Februari 2009. Versi Android kedua ini mengalami masalah penamaan yang sama dengan versi pertamanya. Pada awalnya Android ini akan diberi nama “Bender” akan tetapi karena alasan melanggar

trademark, nama “Bender” tidak jadi disematkan pada versi Android ini. Awalnya versi OS Android ini dirilis untuk perangkat T-Mobile G1 saja. Versi ini merupakan update untuk memperbaiki beberapa bugs, mengganti API dan menambahkan beberapa fitur.

3. Android versi 1.5 Cupcake dirilis tanggal 30 April 2009

Merupakan versi pertama yang diluncurkan secara komersil dan juga pertama dinamai berdasarkan nama kue. Android versi ini baru mulai menambahkan fitur – fitur yang sudah lazim ditemukan saat ini seperti *Widget*, *Auto Rotate*, dan dukungan keyboard virtual dari pihak ketiga. Dalam tahap ini bisa dikatakan kalau Google masih mencoba untuk memasukan fitur – fitur dasar.

4. Android versi 1.6 Donut dirilis

Pada Android Donut, smartphone mendapatkan beberapa fitur tambahan seperti dukungan CDMA, mesin teks ke suara, dan indikator penggunaan baterai. Nomor versi Android Donut lebih sebagai update mini.

5. Android versi 2.0 Eclair dirilis tanggal 26 Oktober 2009

Ada perubahan yang cukup banyak dibandingkan versi Android awal. Pada update ini Google memberikan dukungan dan fitur baru yang berfungsi untuk memudahkan para pengguna. Android Eclair mendapatkan beberapa fitur baru yang utamanya pada aplikasi kamera dengan menambahkan dukungan flash, fokus, hingga efek warna. Pada versi ini Google juga menambahkan Live Wallpaper yang cukup keren pada awal kemunculannya.

6. Android versi 2.2 Froyo, dirilis tanggal 10 Mei 2010

Pada versi ini Android mengalami optimalisasi performa yang cukup signifikan dan ditambah dengan dukungan USB Tethering, Wi-Fi Hotspot, dan push notification yang sangat berguna hingga saat ini. Selain itu pada versi ini kita bisa memindahkan aplikasi ke penyimpanan eksternal jika memungkinkan.

7. Android versi 2.3 Gingerbread tergolong paling sukses, dirilis tanggal 06 Desember 2010

8. Android versi 3.0 Honeycomb memiliki tampilan mewah dan kinerja paling baik. Dirilis tanggal 22 Februari 2011

Sistem operasi satu ini dikhususkan untuk perangkat tablet. Pada Honeycomb ini beberapa fitur berguna masih bertahan hingga sekarang seperti System Bar, dukungan prosesor multi-core, hingga layar Home yang dapat dikustomisasi. Android 3.0 Honeycomb pada saat peluncurannya terlihat sangat futuristik dan menarik banyak perhatian.

9. Android versi 4.0 Ice Cream Sandwich sudah mendukung Flash Player, dirilis tanggal 19 Oktober 2011.

Google mempersatukan perangkat smartphone dan tablet serta menambahkan tampilan antarmuka yang lebih minimalis. Salah satu fitur baru yang cukup menarik pada Android versi ini adalah Android Beam yang memungkinkan transfer data secara cepat menggunakan NFC.

10. Android versi 4.1 Jelly Bean memiliki kelebihan di baterai, navigasi gesture, dan kamera. Dirilis tanggal 09 Juli 2012

Versi ini lebih fokus pada peningkatan kinerja dan keamanan. Sebagian besar lebih fokus pada peningkatan yang ada pada versi Android ini terjadi pada sistem Android itu sendiri. Namun ada satu fitur yang benar – benar perlu diketahui, yakni dukungan resolusi UHD 4K.

11. Android versi 4.4 Kitkat dirilis tanggal 31 Oktober 2013

Fokus terhadap peningkatan user experience dan performa khususnya pada perangkat kelas bawah. Android 4.4 Kitkat memberikan batas minimal RAM tersedia untuk Android sebesar 512 MB sehingga perangkat yang memiliki RAM di bawah itu digolongkan sebagai perangkat RAM rendah. Fitur terbarunya adalah *screen recording*, *new translucent system UI*, peningkatan akses notifikasi, *system-wide settings* untuk *closed acptioning*, peningkatan kinerja, mengaktifkan *Sprint Spark band 26* dan *band 41*, perbaikan kerentanan pada *Heartbleed / OpenSSL*.

12. Android versi 5.0 Lollipop dirilis 17 Oktober 2014

Merupakan yang pertama kali mengusung desain material yang masih menjadi tren hingga saat ini. Fiturnya adalah desain baru (material), peningkatan kecepatan, peningkatan daya tahan baterai, perbaikan video *playback* dan kegagalan password, *support multi SIM cards*, *shortcuts* pada *Quick settings*

untuk *Wi-Fi* dan *Bluetooth*, pengamanan jika perangkat hilang atau dicuri, penggunaan suara *High Definition* untuk telepon, peningkatan kinerja dan stabilitas.

13. Android versi 6.0 Marshmallow dirilis tanggal 28 Mei 2015

Semakin memperkuat apa yang telah dilakukan oleh Google selama ini melalui Android Lollipop. Melalui sistem operasi terbaru ini, Google membawa fitur – fitur canggih yang sangat berguna seperti Support USB Type-C, Support autentikasi sidik jari (Fingerprint), daya tahan baterai lebih meningkat dengan manajemen konsumsi baterai oleh Doze, permissions dashboard, akses System UI Turner, support sistem pembayaran dengan Android Pay yang berkolaborasi dengan Fingerprint Authentication sehingga terjamin keamanannya, tambahan fungsi Google Now yang tidak sekedar melayani perintah suara.

14. Android fungsi Nougat diperkenalkan 22 Agustus 2016

Memberikan peningkatan terhadap pengalaman penggunaan agar lebih mudah. Fitur dukungan multi-window menjadi perhatian utama sehingga pengguna dapat menggunakan beberapa aplikasi secara bersamaan. Selain itu juga ada fitur yang membuat pembaruan sistem operasi lebih mulus dan tidak terasa saat perangkat sedang tidak digunakan. Seperti support multi-window, langsung balas pesan dari jendela atau menu notifikasi, tampilan panel notifikasi dan quick settings yang baru, mode doze yang ditingkatkan (doze mode 2.0) dan menu diantara system settings.

15. Android versi 8.0 Oreo dirilis resmi pada tanggal 21 Agustus 2017

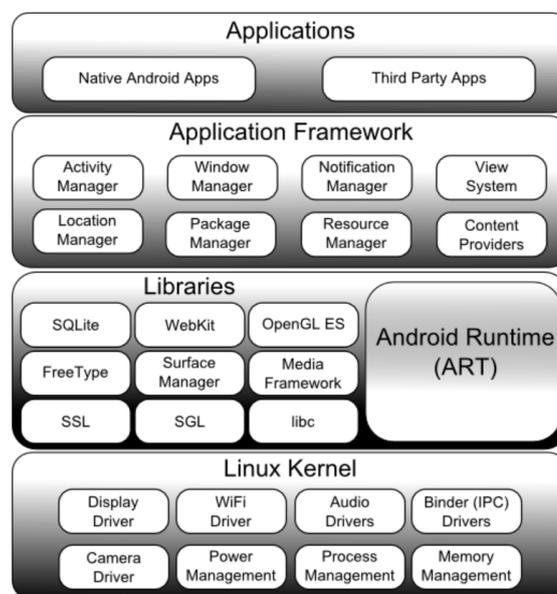
Android O lebih fokus pada kecepatan dan efisiensi, mode picture in picture lebih fleksibel dari android N, aplikasi yang berjalan di latarbelakang lebih diperketat untuk menghemat baterai, baterai lebih tahan lama, dan emoji yang diperbaharui dan lebih banyak.

Alasan menggunakan sistem operasi Android, karena Android merupakan sebuah sistem operasi yang dapat dijalankan diperangkat bergerak (*mobile device*) dan sistem operasi yang paling banyak digunakan berdasarkan data pasar yang diperoleh dari netmarketshare.com situs yang menyediakan statistik pangsa pasar untuk teknologi internet, bahwa Android menguasai pangsa pasar sistem operasi

mobile per Maret 2019, serta keunggulannya sebagai software yang memakai basis kode komputer yang bisa didistribusikan secara terbuka (*open source*) sehingga banyak sekali aplikasi-aplikasi yang bisa diunduh oleh pengguna *smartphone* tanpa harus membeli aplikasi tersebut.

2. Arsitektur Android

Arsitektur sistem operasi android disusun dalam beberapa lapisan perangkat lunak atau *software*. Terdapat empat lapisan dalam arsitektur android yang dimulai dari *Linux Kernel* sebagai lapisan terbawah, *Libraries* dan *Android Runtime*, lalu *Application Framework*, hingga lapisan paling atas yaitu *Applications*[14]. Berikut penjelasan dari setiap lapisan yang digambarkan pada Gambar 2.3.



Sumber gambar : <http://www.techtopia.com/>

Gambar 2.3 Arsitektur Android

1. Linux Kernel

Linux Kernel yang berada pada bagian bawah lapisan perangkat lunak Android menyediakan suatu perantara bagi perangkat keras (*hardware*) dengan lapisan atas perangkat lunak Android. Pada Linux versi 2.6, Linux Kernel menyediakan *preemptive multitasking* layanan level dasar inti sistem seperti *memory*, *process*, dan *power management*. Kernel menyediakan juga suatu

tingkatan jaringan dan *device drivers* untuk perangkat keras seperti monitor, Wi-Fi, dan audio.

2. Android Runtime – ART

Ketika suatu aplikasi Android dibangun dengan Android Studio, aplikasi dikompilasikan menjadi suatu *intermediate bytecode* yang kerap dirujuk sebagai DEX format. Saat aplikasi dijalankan pada suatu perangkat, Android Runtime (ART) menggunakan suatu proses yang disebut sebagai *Ahead-of-Time (AOT) compilation* untuk menerjemahkan *bytecode* tersebut menjadi perintah - perintah dasar yang dapat dieksekusi oleh *processor* perangkat. Format ini dikenal sebagai Executable and Linkable Format (ELF). Setiap kali aplikasi dijalankan dengan versi ELF, aplikasi dapat berjalan dengan cepat dan meningkatkan daya tahan baterai. Hal ini berlawanan dengan pendekatan Just-in-Time (JIT) *compilation* yang digunakan oleh versi Android sebelumnya dimana *bytecode* diterjemahkan dalam suatu *virtual machine* lebih dahulu setiap aplikasi dijalankan.

3. Android Libraries

Android libraries diperuntukan khusus bagi pengembangan Android. *Application framework libraries* dan *libraries* lain yang mendukung pembuatan elemen antarmuka, gambar grafis, dan akses *database* merupakan contoh *libraries* yang masuk dalam kategori ini.

4. Application Framework

Application Framework merupakan sekumpulan layanan yang bersama – sama membentuk suatu lingkungan dimana aplikasi - aplikasi Android berjalan dan diatur. *Framework* ini menerapkan konsep saling berbagi dan saling melengkapi yang mana suatu aplikasi mampu membagikan kemampuan beserta dengan data yang terkait didalamnya untuk dapat digunakan oleh aplikasi lainnya.

5. Applications

Applications terdapat pada lapisan teratas dari susunan arsitektur Android. Pada lapisan ini yaitu seluruh aplikasi bawaan (*native applications*) yang telah dilengkapi dengan fitur khusus seperti aplikasi web browser dan email, serta

aplikasi pihak ketiga (*third party applications*) yang diinstal oleh pengguna sendiri setelah membeli perangkat tersebut.

2.2.6 Android SDK (System Development Kit)

Android SDK adalah *tools API (Application Programming Interface)* yang diperlukan untuk mulai mengembangkan aplikasi data platform Android menggunakan bahasa pemrograman java. Android merupakan subset perangkat lunak untuk ponsel yang meliputi sistem operasi, *middleware* dan aplikasi kunci yang di *release* oleh google. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai pemrograman java[16].

Dengan android SDK dan bahasa pemograman Java, berbagai aplikasi yang canggih dapat dibangun dengan lebih cepat dan menjadi lebih mudah dengan menggunakan semua fitur yang ada di android. Didalamnya terdapat *Add-on* yang dapat ditambahkan untuk memperluas kemampuan android seperti Google API yang akan dibangun berdasarkan Google Maps.

Alasan menggunakan Android SDK pada penelitian ini karena lingkungan pengembangan aplikasi android menggunakan Android Studio, SDK ini sudah disertakan dalam instalasi Android Studio dan SDK ini digunakan untuk membantu proses pembangunan aplikasi Android yang berisikan *libraries – libraries*.

2.2.7 Android Studio Editor

Android Studio merupakan salah satu *modern code* editor yang sangat membantu dalam pembangunan suatu aplikasi Android. Sebagai IDE (*Integrated Development Environment*) resmi yang disarankan oleh Google dalam pembangunan aplikasi Android, Android Studio menawarkan banyak fitur dan *tools* yang dapat digunakan oleh pengembang aplikasi Android. Berikut elemen – elemen yang membentuk jendela kerja Android Studio sebagai berikut[14].

1. Menu Bar

Menu Bar terletak pada bagian paling atas dari jendela kerja Android Studio yaitu menampilkan sekumpulan menu yang dapat dipilih, seperti *File, Edit, View* dan seterusnya. Dibawah menu bar terdapat *toolbar* yang dapat menampilkan *icon – icon shortcut* untuk melakukan aksi tertentu seperti *Open,*

Save, Run dan sebagainya. Kemudian di bawah *toolbar* terdapat *Navigation bar* yang memperlihatkan lokasi *file* yang sedang dibuka dalam bentuk navigasi *folder* dan *file*.

2. Document Tabs

Android Studio mampu membuka sejumlah *file* secara bersamaan untuk diedit oleh pengembang. Saat membuka *file* baru, *document tab* yang baru akan dibentuk yang disertai dengan nama *file* tersebut.

3. Project Window

Project Window berguna untuk memperlihatkan susunan *folder* dan *file* yang digunakan dalam proyek dalam bentuk *tree* yang mudah ditelusuri.

4. Bottom Sidebar

Bottom Sidebar merupakan *sidebar* yang dapat dibuka untuk menampilkan informasi khusus yang dibutuhkan seperti jendela *Terminal, Messages, Event Log* dan sebagainya. Berfungsi untuk membantu dalam proses *debugging* aplikasi.

5. Editor Area

Editor Area merupakan area utama dimana kode program ditampilkan, ditulis dan diubah oleh pengembang. Secara *default*, *editor* akan menampilkan satu *panel* tunggal yang menyajikan isi dari *file* yang sedang dibuka.

6. Editor Gutter Area

Gutter area digunakan oleh editor untuk menampilkan informasi *icons* dan *controls*. Beberapa informasi yang ditampilkan adalah *debugging breakpoint markers, bookmarks*, dan *line numbers*.

7. Validation and Marker Sidebar

Pada bagian atas *validation sidebar* terdapat *icon* kecil yang dapat berubah warna dari hijau yaitu tidak ada peringatan atau kesalahan yang terdeteksi dalam file, kemudian kuning yaitu terdapat peringatan yang terdeteksi, dan merah yaitu terdapat kesalahan yang harus diperbaiki.

8. Status Bar

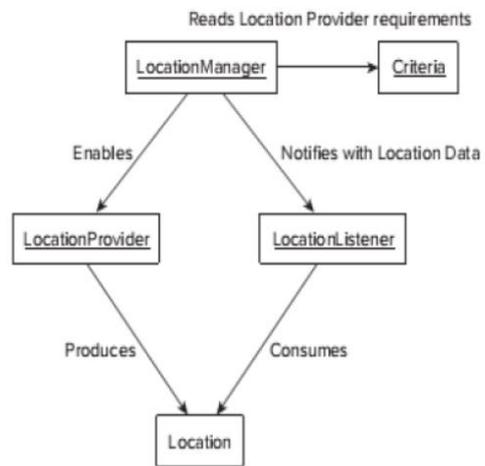
Status Bar yaitu terdapat informasi tentang status sesi *editing* yang sedang aktif. Informasi tersebut dapat berupa posisi *cursor* saat ini dan format *encoding file* yang dibuka.

2.2.8 GPS

Global Positioning System (GPS) adalah sistem satelit navigasi dan penentuan posisi yang dimiliki dan dikelola oleh Amerika Serikat[17]. Sistem ini didesain untuk memberikan posisi dan kecepatan tiga-dimensi serta informasi mengenai waktu, secara terus menerus di seluruh dunia tanpa bergantung waktu dan cuaca, bagi banyak orang secara simultan.

Dalam menentukan lokasi dari perangkat Android, ada beberapa cara yang digunakan dalam memperoleh data tersebut. Cara yang digunakan sebagai berikut.

- a. *GPS Provider*, dalam menentukan lokasi dari user Android, perangkat Android langsung terhubung dengan satelit guna memperoleh koordinat dari pemakai.
- b. *Network Provider*, dalam menentukan lokasi dari user Android, ketika perangkat Android tidak bisa terkoneksi dengan satelit, maka secara otomatis, perangkat tersebut akan mencari posisi dari *Base Transceiver Station (BTS)* dari *network provider* perangkat android tersebut. Sebelum terkoneksi dengan GPS provider ataupun network provider, ada beberapa tahap yang dilakukan seperti yang dijelaskan pada Gambar 2.4.



Sumber gambar : Professional Sensor Android Programming (2012) [18]

Gambar 2.4 Android Location Component

1. Cara Kerja GPS

Dalam sistem navigasi GPS, Bagian yang paling penting adalah beberapa satelit yang berada di orbit bumi atau yang sering kita sebut di ruang angkasa. Satelit GPS saat ini berjumlah 24 unit yang semuanya dapat memancarkan sinyal ke bumi yang lalu dapat ditangkap oleh alat penerima sinyal tersebut atau GPS Tracker. Selain satelit terdapat 2 sistem lain yang saling berhubungan, sehingga jadilah 3 bagian penting dalam sistem GPS. Ketiga bagian tersebut terdiri dari:

a. GPS Control Segment

Control segment GPS terdiri dari lima stasiun yang berada di pangkalan Falcon Air Force, Colorado Springs, Ascension Island, Hawaii, Diego Garcia dan Kwajalein. Kelima stasiun ini adalah mata dan telinga bagi GPS. Sinyal-sinyal dari satelit diterima oleh bagian kontrol, kemudian dikoreksi, dan dikirimkan kembali ke satelit. Data koreksi lokasi yang tepat dari satelit ini disebut data ephemeris, yang kemudian nantinya dikirimkan ke alat navigasi yang dimiliki.

b. GPS Space

Space Segment adalah terdiri dari sebuah jaringan satelit yang terdiri dari beberapa satelit yang berada pada orbit lingkaran yang terdekat dengan tinggi nominal sekitar 20.183 km di atas permukaan bumi. Sinyal yang dipancarkan oleh seluruh satelit tersebut dapat menembus awan, plastik dan kaca, namun tidak bisa menembus benda padat seperti tembok dan rapatnya pepohonan. Terdapat 2 jenis gelombang yang hingga saat ini digunakan sebagai alat navigasi berbasis satelit. Masing-masingnya adalah gelombang L1 dan L2, dimana L1 berjalan pada frekuensi 1575.42 MHz yang bisa digunakan oleh masyarakat umum, dan L2 berjalan pada frekuensi 1227.6 Mhz dimana jenis ini hanya untuk kebutuhan militer saja.

c. GPS User Segment

User segment terdiri dari antenna dan prosesor receiver yang menyediakan positioning, kecepatan dan ketepatan waktu ke pengguna.

Bagian ini menerima data dari satelit-satelit melalui sinyal radio yang dikirimkan setelah mengalami koreksi oleh stasiun pengendali (GPS Control Segment).

2.2.9 Location Based-Service (LBS)

Location Based Service adalah service yang berfungsi untuk mencari dengan teknologi Global Positioning Service (GPS) dan Google's cell-based location. Map dan layanan berbasis lokasi menggunakan lintang dan bujur untuk menentukan lokasi geografis, namun sebagai user kita membutuhkan alamat atau posisi realtime kita bukan nilai lintang dan bujur. Android menyediakan geocoder yang mendukung forward dan reverse geocoding. Menggunakan geocoder, anda dapat mengkonversi nilai lintang bujur menjadi alamat dunia nyata atau sebaliknya. Location based service atau layanan berbasis lokasi adalah istilah umum yang digunakan untuk menggambarkan teknologi yang digunakan untuk menemukan lokasi perangkat yang kita gunakan. Dua unsur utama LBS adalah [18]:

- a. Location Manager (API Maps) Menyediakan tools/resource untuk LBS, Application Programming Interface (API) Maps menyediakan fasilitas untuk menampilkan, memanipulasi maps/peta beserta feature-feature lainnya seperti tampilan satelit, street (jalan), maupun gabungannya. Paket ini berada pada `com.google.android.maps`.
- b. Location Providers (API Location) Menyediakan teknologi pencarian lokasi yang digunakan oleh device/perangkat. API Location berhubungan dengan data GPS dan data lokasi real-time. API Location berada pada paket Android yaitu dalam paket `android.location`. Dengan Location Manager, kita dapat menentukan lokasi kita saat ini, Track gerakan/perpindahan, serta kedekatan dengan lokasi tertentu dengan mendeteksi perpindahan.

2.2.10 Google Maps API

Google Maps merupakan salah satu dari banyak aplikasi yang terintegrasi sebagai *default application* pada *platform* Android. Google Maps digunakan untuk menampilkan, memberi tanda, dan menavigasikan peta. Selain *stand-alone application* Google Maps, fungsi dan tampilan Google Maps dapat dimasukkan ke

dalam aplikasi buatan sendiri. Contoh penggunaan nyata Google Maps API di dalam aplikasi, yaitu Aplikasi Go-Jek, GrabBike, Uber, dan lainnya[14].

Google Maps API adalah sebuah layanan (*service*) yang diberikan oleh Google kepada para pengguna untuk memanfaatkan Google Map dalam mengembangkan aplikasi. Google Maps API menyediakan beberapa fitur untuk memanipulasi peta, dan menambah konten melalui berbagai jenis services yang dimiliki, serta mengizinkan kepada pengguna untuk membangun aplikasi enterprise di dalam websitenya. Pengguna dapat memanfaatkan layanan-layanan yang ditawarkan oleh Google Maps setelah melakukan registrasi dan mendapatkan Google Maps API Key. Google menyediakan layanan ini secara gratis kepada pengguna di seluruh dunia.

2.2.11 Google Maps Directions API

Google Maps Directions API adalah layanan yang disediakan oleh google untuk memudahkan pengembang dalam menghitung arah antara lokasi satu ke lokasi yang lain dengan menggunakan HTTP *request* untuk memanggil Google Directions API.

Google Maps API didesain untuk menghitung arah alamat statis (sudah diketahui sebelumnya) untuk penempatan konten aplikasi pada peta; layanan ini tidak didesain untuk merespons input pengguna secara real-time, misalnya. Untuk perhitungan arah dinamis.

Sehingga, pada penelitian ini digunakan Google Maps Direction API untuk mencari petunjuk arah ketika sedang dalam mengemudi, berjalan atau sedang menggunakan transportasi dan sebagainya karena dengan menggunakan Google Maps Directions API dapat mengembalikan rute yang paling efisien saat menghitung arah dan dapat mengoptimalkan waktu perjalanan ketika memilih rute yang dituju.

2.2.12 Geofencing

Geofencing adalah teknologi yang digunakan untuk memantau objek bergerak (kendaraan, orang, kontainer, dll), dengan menggunakan GPS. Koordinat geografis dari objek secara otomatis dilacak dan secara berkala dikirim ke pusat pengawasan, melalui jaringan perangkat seluler[19].

Salah satu cara untuk melakukan geofencing dengan memasang GPS receiver untuk dilacak ke objek dan menggunakan data GPS dari receiver untuk menentukan dimana objek tersebut berada yang dibandingkan dengan lokasi geofence. Fungsi utama geofencing yaitu untuk melakukan pemantauan jarak jauh (monitoring) suatu perangkat mobile dari peta virtual ketika perangkat mobile keluar atau memasuki daerah yang dibatasi geofence (pagar virtual). Berbagai macam teknik geofencing telah dikembangkan untuk berbagai kebutuhan yang berbeda.

Teknologi geofencing ini digunakan untuk mendeteksi keberadaan titik rawan yang akan dilalui oleh pengguna. Jika pengguna akan melewati daerah atau titik rawan aplikasi ini akan memberikan sebuah peringatan atau notifikasi.

2.2.13 PHP

PHP termasuk bahasa program yang hanya bisa berjalan di sisi server, atau sering disebut *Side Server Language*. Jadi, program yang dibuat dengan kode PHP tidak bisa berjalan kecuali dijalankan pada web server, tanpa adanya web server yang terus berjalan maka tidak bisa dijalankan[20].

PHP memiliki banyak framework yang sangat populer saat ini. Framework PHP adalah suatu kerangka kerja untuk pembuatan program dengan menggunakan PHP, dengan framework ini maka pengembang aplikasi tidak perlu lagi membuat banyak fungsi tambahan sendiri karena sudah disediakan oleh framework. Kelebihan lainnya yaitu teknik pemrograman yang lebih rapi dan terstruktur[21]. Berikut beberapa framework PHP yang populer saat ini adalah:

1. CodeIgniter
2. Laravel
3. Symfony
4. Yii

Salah satu web server yang biasa dipakai dan gratis adalah Apache. Hasil dari PHP adalah berbentuk website maupun aplikasi berbasis web sehingga hasilnya dapat dilihat dengan menjalankannya dari web browser seperti Internet Explorer, Mozilla Firefox, Chrome atau browser lainnya. Peneliti dalam pembuatan

website ini menggunakan framework Laravel karena Laravel adalah framework *open source* yang dapat digunakan secara gratis serta menggunakan pola MVC yaitu dapat membuat struktur kode yang lebih rapi dimana pola tersebut memisahkan antara logika dan view. Website ini digunakan untuk admin yaitu pihak kepolisian sehingga aplikasi yang dibuat terintegrasi dengan aplikasi mobile.

Contoh sintaks PHP :

```
<html>
<head>
<title> Contoh </title>
</head>
<body>
<?php
Echo "Hai, saya dari script PHP!";
?>
</body>
</html>
```

2.2.14 Java

Java merupakan bahasa pemrograman tingkat tinggi (*high level*). Java dibuat oleh perusahaan Sun Microsystems, oleh James Gosling, Patrick Naughton, dan Mike Sheridan pada 1991. Dalam perkembangannya java diakuisisi oleh perusahaan Oracle. Java memiliki banyak keunggulan salah satunya adalah mudah dipelajari oleh semua orang karena *syntax* (tata bahasa) yang mirip dengan bahasa manusia[22].

Java mempunyai konsep *Object Oriented Programming* (OOP). Sehingga dalam mengembangkannya lebih mudah, serta dapat menjaga sistem tetap modular, fleksibel, dan ekstensibel. Java memiliki banyak API (*Application Programming Interface*) yang digunakan untuk pengembangan dan keperluan para *programmer*.

Java memiliki kemampuan untuk menangani kesalahan pada *coding*, khususnya pada *error checking* atau *error runtime*, seperti pada *Exception Handling*.

Hasil dari mengompilasi *source code*, yaitu berupa *byte code* yang berbentuk *file* berekstensi *class*. *Byte code* tersebut dapat langsung dieksekusi pada tiap *platform* dengan menggunakan *Java Virtual Machine (JVM)* sebagai interpreter terhadap *byte code* tersebut. JVM adalah sebuah aplikasi yang berjalan pada sebuah sistem operasi dan menerjemahkan *byte code* program java dan mengeksekusinya sehingga secara konsep bisa dianggap sebagai sebuah *interpreter*.

Kelebihan bahasa pemrograman Java adalah sebagai berikut [23].

1. Multiplatform

Kelebihan utama dari java ialah dapat dijalankan di beberapa platform / sistem operasi komputer. Dengan kelebihan ini pemrogram cukup menulis sebuah program java dan dikompilasi sekali lalu hasilnya dapat dijalankan di beberapa platform tanpa perubahan.

2. OOP (Object Oriented Programming)

Java merupakan bahasa pemrograman berorientasi objek.

3. Pustaka kelas yang lengkap

Java memiliki kelengkapan kelas yang dapat digunakan oleh para pengguna.

4. Begaya C++

Memiliki sintaks seperti bahasa pemrograman C++ sehingga menarik banyak pemrogram C++ untuk pindah ke Java.

5. Pengumpulan sampah otomatis

Memiliki fasilitas pengaturan penggunaan memori sehingga para pemrogram tidak perlu melakukan pengaturan memori secara langsung.

6. 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.

7. Penggunaan memori banyak

Penggunaan memori untuk program berbasis Java jauh lebih besar daripada bahasa tingkat tinggi yang lain.

Dalam Penelitian ini peneliti membuat aplikasi dengan menggunakan bahasa Java karena editor yang digunakan yaitu Android Studio. Berikut contoh sintaks java.

```
public class Latihan1
{
    public static void main(String args[])
    {
        // ini adalah komentar
        System.out.println("Hallooo semua...");
        System.out.println("Selamat belajar Java");
        System.out.println("Asyik kan?..");
    }
}
```

2.2.15 Object Oriented Programming (OOP)

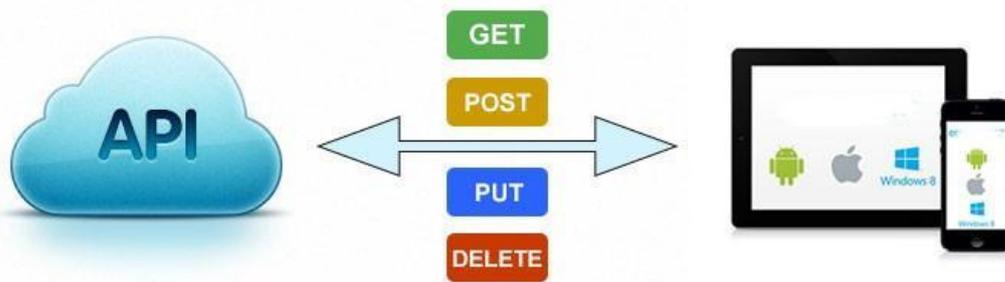
Object Oriented Programming atau metode berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan operasi yang diberlakukan terhadapnya. Metodologi berorientasi objek merupakan suatu cara bagaimana sistem perangkat lunak dibangun melalui pendekatan objek secara sistematis. Metode berorientasi objek meliputi rangkaian aktivitas analisis berorientasi objek, perancangan berorientasi objek, pemrograman berorientasi objek dan pengujian berorientasi objek [24].

Sistem berorientasi objek merupakan sistem yang dibangun dengan berdasarkan metode berorientasi objek yaitu sebuah sistem yang komponennya

dibungkus (dienkapsulasi) menjadi kelompok data dan fungsi. Setiap komponen dalam sistem tersebut dapat mewarisi (inheritance) atribut dan sifat dan komponen lainnya dan dapat berinteraksi satu dengan yang lainnya.

2.2.16 Web API (Application Programming Interface)

Web API adalah antar muka program dari sistem yang dapat diakses lewat *method* dan *header* pada protokol HTTP yang standar. Web API dapat diakses dari berbagai macam HTTP client seperti browser dan perangkat mobile. Web API juga memiliki keuntungan karena menggunakan infrastruktur yang juga digunakan oleh web terutama untuk penggunaan *caching* dan *concurrency*[25]. Fungsi dari API ini pada umumnya adalah sebagai sumber data yang bisa digunakan untuk kebutuhan sistem atau aplikasi tertentu, API memungkinkan untuk dapat bertukar informasi data dari satu aplikasi ke aplikasi lainnya. Untuk lebih jelas gambaran dari API bisa dilihat pada Gambar 2.5.



Sumber gambar : <https://kudo.co.id>

Gambar 2.5 Arsitektur API

Selain sebagai sumber data API juga mempercepat proses *development* dengan menyediakan fungsi secara terpisah sehingga *developer* tidak perlu membuat fungsi yang sama. API sendiri terdiri dari beberapa elemen seperti *functions*, *protocols*, dan *tools* untuk membantu *developer* membuat aplikasi. Apabila membangun aplikasi yang sangat kompleks penggunaan API ini akan sangat terasa sangat membantu sekali contoh dalam pembuatan aplikasi yang membutuhkan integrasi *payment gateway* peran API disini sangat di utamakan.

Untuk melakukan *request* kepada API *client* harus memenuhi beberapa syarat tertentu sebagai berikut :

1. URL

Dalam HTTP (*Hyper Text Transfer Protocol*) URL adalah alamat unik dari suatu web atau sistem tertentu dan akan di arahkan sesuai proses bisnis yang ada pada server. URL dapat dibuat berupa halaman sebuah website, gambar, video, atau data tertentu.

2. Method

Request method yang di lakukan oleh *client* memberitahu server bahwa aksi apa yang perlu di lakukan untuk memenuhi permintaan *client*. Ada 4 method yang biasanya digunakan yaitu :

- a. GET – meminta server untuk memberikan respon sumber.
- b. POST – meminta server untuk membuat sumber baru.
- c. PUT – meminta server untuk melakukan edit/update pada sumber yang sudah ada.
- d. DELETE – meminta server untuk melakukan penghapusan sumber.

3. Headers

Hedaers biasanya berisi meta-informasi seperti waktu permintaan, format data yang diinginkan, atau informasi perangkat yang digunakan oleh *client* sehingga server dapat megirimkan data sesuai dengan perangkat yang digunakan.

4. Body

Body berisi data yang ingin dikirmkan client ke server, sehingga server dapat melakukan respon aksi berdasarkan data yang dikirimkan client ke server.

Format data yang dihasilkan oleh web api biasanya berbentuk JSON. JSON (*JavaScript Object Notation*) adalah salah satu struktur data JavaScript untuk mendefinisikan objek. Model struktur data JSON telah menjadi salah satu standar untuk pertukaran data yang ringan, yang sama populernya dengan XML. Ukuran file JSON lebih kecil daripada file XML, karena pendefinisiannya yang lebih sederhana. Sehingga dalam melakukan *parsing* data dari file JSON menjadi lebih mudah dan cepat. Hal ini membuat JSON menjadi alternatif format dalam pertukaran data[21].

JSON berawal dari bahasa pemrograman JavaScript. Walaupun JSON didasarkan pada subset bahasa pemrograman JavaScript. JSON dianggap sebagai format data yang tak tergantung pada suatu bahasa. JSON didukung banyak bahasa pemrograman, sehingga dapat digunakan pada sestiap bahasa pemrograman.

2.2.17 Firebase Cloud Messaging

Firestore Cloud Messaging (FCM) atau yang dulu dikenal dengan Google Cloud Messaging (GCM) adalah sebuah *platform* yang memungkinkan pengiriman pesan atau notifikasi dilakukan secara *reliable* dan tanpa biaya [26]. Dengan menggunakan FCM, aplikasi klien dapat menerima notifikasi ketika terdapat data baru yang siap disinkronisasi. FCM juga memungkinkan untuk mengirimkan pesan notifikasi kepada masing – masing pengguna atau pengguna pada *segment* tertentu. Berikut adalah arsitektur dari FCM dapat dilihat pada Gambar 2.6.



Sumber: <https://firebase.google.com/docs/cloud-messaging>

Gambar 2.6 Arsitektur Firebase Cloud Messaging

Ada dua komponen utama dalam mengimplementasikan Firebase Cloud Messaging pada sebuah sistem. Aplikasi server yaitu, *trusted environment* atau *Notifications Console GUI* pada firebase yang digunakan untuk mengirimkan dan menargetkan pesan notifikasi ke klien. Aplikasi klien yang berupa iOS, Android atau Web (JavaScript) untuk menerima pesan notifikasi. Dalam Uji coba atau *highly targeted marketing* dan *user engagement* Penggunaan console sangat berguna.

Pengiriman pesan dengan cara ini tidak membutuhkan teknik untuk coding, hanya memasukan data – data yang dibutuhkan. Pada *trusted environment*, pengiriman pesan dapat dilakukan melalui admin SDK dan HTTP/XMPP. Admin SDK menyediakan sebuah API untuk mengirimkan pesan dari FCM kepada *end user device*, tetapi SDK ini hahnya mendukung bahasa pemrograman node.js. Dalam mengirimkan pesan notifikasi dapat dilakukan programmatically kepada masing – masing devices, grup devices, topics, dan conditions. Pengiriman melalui HTTP hanya dapat dilakukan secara *downstream*, yaitu dari *cloud-device*. Sedangkan, XMPP dapat dilakukan secara *upstream* dan *downstream*, yaitu dari *device-cloud* dan *cloud-device*. [27] Adapun Fitur utama pada Firebase Cloud Messaging yang dapat dilihat pada

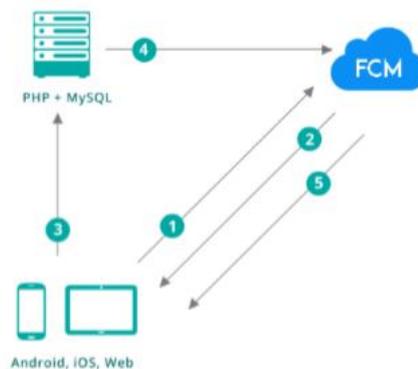
Tabel 2.1 Fitur Utama Firebase Cloud Messaging

Fitur	Keterangan
Mengirim pesan notifikasi atau pesan data	Mengirim pesan notifikasi yang ditampilkan kepada pengguna. Atau mengirim pesan data dan menentukan sepenuhnya apa yang terjadi dalam kode aplikasi
Penargetan pesan serbaguna	Mendistribusikan pesan ke aplikasi klien dengan salah satu dari 3 cara — ke sebuah perangkat, ke grup perangkat, atau ke perangkat yang berlangganan topik.
Mengirim pesan dari aplikasi klien	Mengirim notifikasi, chat, dan pesan lain dari perangkat ke server melalui saluran koneksi FCM yang andal dan hemat baterai.

Untuk bisa menerima pesan notifikasi dari Firebase, aplikasi harus menggunakan *service* yang meng-*extend* Firebase Messaging Service, dimana *service* ini digunakan untuk menangani setiap notifikasi yang didapatkan. Sedangkan, untuk dapat mengirimkan pesan notifikasi ke perangkat yang diinginkan, aplikasi harus meng-*extend* Firebase Instance Service agar mendapatkan *registration token* yang berguna untuk membedakan aplikasi yang ter-*install* pada setiap perangkat.

Adapun cara kerja Firebase Cloud Meesaging sebagai berikut [28].

1. Aplikasi yang ter-instal pada smartphone pengguna akan secara otomatis mendaftarkan smartphone android ke dalam FCM.
2. Selanjutnya FCM akan mengirim token ke smartphone pengguna sebagai penanda, token yang dikirim bersifat unique sehingga berbeda-beda pada tiap smartphone.
3. Smartphone yang telah mendapat token, akan mengirimkannya ke dalam server, lalu server akan menyimpan token tersebut.
4. Jika terjadi perubahan nilai data dari data base, maka FCM akan mendeteksi perubahan nilai data tersebut.
5. FCM akan mengetahui dari mana data dikirim dan akan dikirim melalui token yang telah terdaftar, kemudian FCM akan mengirim notifikasi ke smartphone pengguna, smartphone yang menerima notifikasi akan memanggil broadcast receiver kemudian service notifikasi akan muncul pada layar smartphone pengguna. Alur cara kerja FCM dijelaskan pada Gambar 2.7.



Sumber gambar : Pengembangan Aplikasi Android Bimbingan Skripsi dengan Fitur Notifikasi [28]

Gambar 2.7 Cara Kerja FCM

2.2.18 MySQL

MySQL merupakan software RDBMS (Relational Database Management System atau server database) yang dapat mengelola database dengan sangat cepat, dapat menampung data dalam jumlah sangat besar, dapat diakses oleh banyak user

(multi-user), dan dapat melakukan suatu proses secara sinkron atau berbarengan (multi-threaded)[29].

MySQL merupakan software yang bersifat open source. Database sendiri merupakan sekumpulan tabel-tabel yang saling berhubungan. Sedangkan open source sendiri menyatakan bahwa software ini dilengkapi dengan source code (kode yang dipakai untuk membuat MySQL). Selain tentu saja bentuk executable-nya atau kode yang dapat dijalankan secara langsung dalam sistem operasi.

Beberapa alasan memilih MySQL sebagai server database untuk aplikasi yang dikembangkan.

a. Fleksibel

MySQL dapat digunakan untuk mengembangkan aplikasi *desktop* maupun aplikasi web dengan menggunakan teknologi yang bervariasi.

b. Performa Tinggi

MySQL memiliki mesin *query* dengan performa tinggi, dengan demikian proses transaksional dapat dilakukan dengan sangat cepat. Hal ini terbukti dengan digunakannya MySQL sebagai *database* dari beberapa aplikasi web yang memiliki *traffic* (lalu lintas) sangat tinggi.

c. Lintas Platform

MySQL dapat digunakan pada *platform* atau lingkungan (dalam hal ini sistem operasi) yang beragam, bisa Microsoft Windows, Linux, atau UNIX. Ini menyebabkan proses migrasi data antar sistem operasi dapat dilakukan secara lebih mudah. Misal pada saat ingin mengganti sistem operasi pada mesin server.

d. Gratis

MySQL dapat digunakan secara gratis. Meskipun demikian, ada juga software MySQL yang bersifat komersial. Biasanya yang sudah ditambahi dengan kemampuan spesifik dan mendapat pelayanan dari *technical support*.

e. Proteksi Data yang Andal

Perlindungan terhadap keamanan data merupakan hal nomor satu yang dilakukan oleh para profesional di bidang database. MySQL menyediakan

mekanisme yang *powerful* untuk menangani hal tersebut yaitu dengan menyediakan fasilitas manajemen user, enkripsi data dan lain sebagainya

f. Komunitas Luas

Dikarenakan pengguna MySQL banyak sehingga MySQL memiliki komunitas yang luas. Hal ini sangat berguna apabila sedang menemui suatu permasalahan dalam proses pengolahan data menggunakan MySQL, sehingga dapat mendiskusikan langsung permasalahan tersebut melalui sebuah forum agar mendapatkan solusi.

2.2.18.1 Data Definition Language (DDL)

Data Definition Language (DDL) adalah kumpulan perintah SQL yang berkaitan dengan pembuatan, perubahan, dan penghapusan database maupun objek – objek yang terdapat di dalam database, seperti tabel, indeks, prosedur / fungsi, trigger, dan sebagainya[29].

Perintah SQL yang termasuk kategori DDL adalah :

1. CREATE, berfungsi untuk membuat database dan objek – objek di dalam database.
2. ALTER berfungsi untuk mengubah database dan objek – objek di dalam database. Seperti mengganti nama tabel, menambah kolom, mengubah kolom, menghapus kolom maupun memberikan atribut pada kolom.
3. DROP, berfungsi untuk menghapus database dan objek – objek di dalam database.

Berikut contoh sintaks DDL.

```
CREATE DATABASE BUKUDB;
CREATE TABLE BUKU (
    BUKU_ISBN CHAR(13) NOT NULL,
    BUKU_JUDUL VARCHAR (75),
    PRIMARY KEY (BUKU_ISBN);
ALTER TABLE BUKU
    CHANGE BUKU_ISBN BUKU_ID CHAR(15);
DROP TABLE BUKU;
```

2.2.18.2 Data Management Language (DML)

Data Manipulation Language (DML) adalah kumpulan perintah SQL yang berkaitan dengan data atau isi dari suatu tabel. Dengan perintah – perintah di dalam DML, sehingga dapat memanipulasi (menambah, mengubah, dan menghapus) data yang terdapat pada suatu tabel secara mudah[29].

Perintah – perintah yang termasuk ke dalam DML adalah :

- a. INSERT, berfungsi untuk menambah atau memasukan data baru ke dalam tabel. Penggunaannya setelah database dan tabel selesai dibuat.
- b. UPDATE, berfungsi untuk mengubah data dalam tabel dengan nilai baru. Perintah ini digunakan ketika terdapat data yang salah, maka dapat diubah isi datanya menggunakan perintah UPDATE.
- c. DELETE, perintah ini digunakan untuk menghapus data dari suatu tabel. Biasanya data yang dihapus merupakan data yang sudah tidak diperlukan lagi. Pada saat menghapus data, perintah yang telah dijalankan tidak dapat digagalkan, sehingga data yang telah hilang tidak dapat dikembalikan lagi.

Berikut contoh sintaks DML :

```
INSERT INTO PENERBIT (PENERBIT_ID, PENERBIT_NAMA)
VALUES ('PB06', 'INFORMATIKA');
UPDATE PENERBIT
SET PENERBIT_NAMA = 'INFORMATIKA BANDUNG'
WHERE
PENERBIT_ID = 'PB06';
DELETE FROM BUKU
WHERE BUKU_ISBN = '888-96771-3-1'
```

2.2.18.3 Data Control Language (DCL)

Data Control Language (DCL) adalah salah satu komponen SQL yang berfungsi untuk mengontrol hak ases user. Perintah yang termasuk ke dalam DCL adalah GRANT dan REVOKE.

Selain pengontrolan hak akses user, dalam SQL dapat mengontrol transaksi data yang telah dilakukan. Perintah – perintah yang berkaitan dengan pengontrolan

transaksi data digolongkan ke dalam *Transactional Control Language* (TCL). Perintah yang termasuk ke dalam TCL adalah COMMIT dan ROLLBACK[29].

Perintah DCL ini merupakan perintah yang fungsinya berperan untuk melakukan proses manipulasi user pada sebuah database, dan juga menyangkut hak akses dari sebuah database. Jadi, dengan menggunakan perintah ini dapat menentukan siapa saja yang berhak untuk mendapatkan akses ke dalam database yang dimiliki.

Terdapat beberapa perintah DCL pada MySQL sebagai berikut.

- a. GRANT, perintah ini digunakan untuk memberikan hak akses (privilege) oleh administrator (pemilik utama) server kepada user (pengguna biasa). Melalui pemberian hak akses semacam ini, seorang user memiliki keterbatasan dalam menggunakan database sehingga data akan aman dari pihak – pihak yang tidak berkepentingan. Hak akses tersebut berupa hak membuat (CREATE), mengambil (SELECT), menghapus (DELETE), mengubah (UPDATE), dan hak khusus berkenaan dengan sistem databasenya.
- b. REVOKE, perintah ini memiliki kegunaan terbalik dengan GRANT, yaitu untuk menghilangkan atau mencabut salah satu atau beberapa hak akses yang telah diberikan kepada user oleh administrator.
- c. COMMIT, perintah ini berfungsi untuk menyimpan perubahan – perubahan yang dilakukan terhadap database (melalui perintah INSERT, UPDATE, atau DELETE) secara permanen.
- d. ROLLBACK, perintah ini merupakan kebalikan dari perintah COMMIT, yang berfungsi untuk membatalkan transaksi atau perubahan – perubahan yang telah dilakukan ke dalam database (melalui perintah INSERT, UPDATE, atau DELETE). dengan melakukan pembatalan transaksi, data di dalam database akan kembali ke keadaan awal (keadaan sebelum dilakukan perubahan). Sehingga perubahan yang dilakukan tidak akan pernah disimpan ke dalam database.

Berikut contoh sintaks DCL.

```
GRANT ALL
ON BUKUDB
TO resy@localhost

IDENTIFIED BY 's3c23t';

REVOKE ALL ON BUKUDB.* FROM resy;
```

2.2.19 UML

Unified Modeling Language (UML) adalah sebuah bahasa yang berdasarkan gambar untuk memvisualisasikan, menspesifikasikan, membangun dan pendokumentasian dari sebuah sistem pengembangan perangkat lunak berbasis Objek. Unified Modeling Language (UML) bukanlah merupakan bahasa pemrograman tetapi model - model yang tercipta berhubungan langsung dengan berbagai macam bahasa pemrograman, sehingga memungkinkan melakukan pemetaan (mapping) langsung dari model-model yang dibuat dengan Unified Modeling Language (UML) dengan bahasa-bahasa pemrograman berorientasi obyek, seperti Java[30].

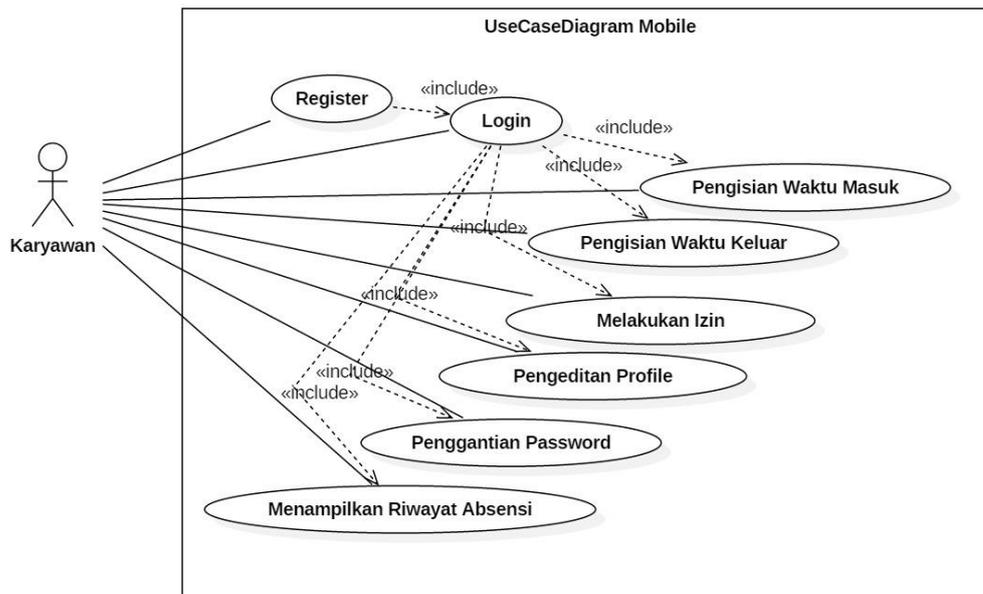
UML merupakan bahasa visual untuk menggambarkan sistem dan aplikasi berbasis objek yang sudah ada sebelumnya atau merancang sistem dan aplikasi di awal pengembangan sebelum memasuki tahap *coding*.

UML tersusun atas sejumlah elemen grafis membentuk 9 diagram-diagram. Dalam penelitian ini melakukan desain hanya 4 diagram yaitu Use Case Diagram, Activity Diagram, Class Diagram dan Sequence Diagram.

1. Use Case Diagram

Use case atau diagram *use case* mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. *Use case* bisa digunakan untuk mengetahui fungsi apa saja yang ada di dalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi – fungsi itu [24].

Sehingga, *use case* diagram dapat digunakan untuk menggambarkan kebutuhan sistem dan memahami bagaimana sistem seharusnya bekerja. Berikut contoh use case diagram yang dapat dilihat pada Gambar 2.8.



Gambar 2.8 Contoh Use Case Diagram

Komponen yang terdapat pada sebuah use case diagram terdiri dari [16]:

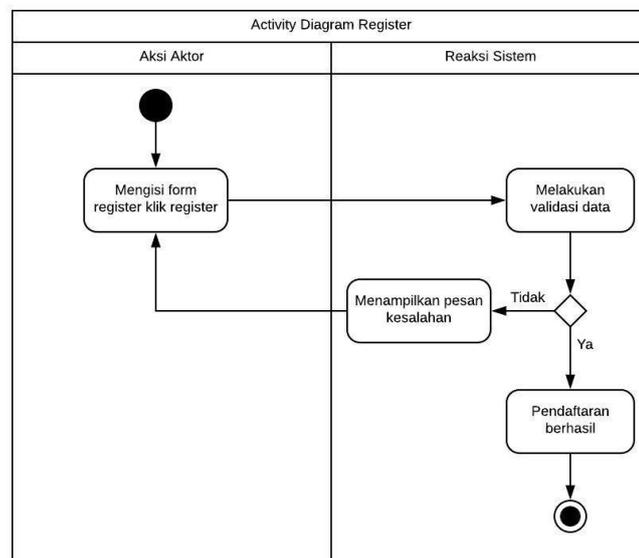
- a. Aktor merupakan orang, proses, atau sisten lain yang berinteraksi dengan sistem informasi yang akan dibuat di luar sistem informasi yang akan dibuat itu sendiri, jadi simbol dari aktor adalah gambar orang, tetapi aktor belum tentu merupakan orang.
 - b. Use case merupakan fungsionalitas yang disediakan sistem sebagai unit – unit yang saling bertukar pesan antar unit atau aktor.
2. Activity diagram

Activity diagram atau diagram aktivitas menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis atau menu yang ada pada perangkat lunak [24].

Diagram aktivitas digunakan untuk mendefinisikan hal – hal berikut.

- a. Rancangan proses bisnis dimana setiap urutan aktivitas yang digambarkan merupakan proses bisnis sistem yang didefinisikan.
- b. Urutan atau pengelompokan tampilan dari sistem atau *user interface* dimana setiap aktivitas dianggap memiliki sebuah rancangan antarmuka tampilan.
- c. Rancangan pengujian dimana setiap aktivitas dianggap memerlukan sebuah pengujian yang perlu didefinisikan kasus ujinya.
- d. Rancangan menu yang ditampilkan pada perangkat lunak.

Berikut contoh activity diagram yang dapat dilihat pada Gambar 2.9.



Gambar 2.9 Contoh Activity Diagram

3. Class Diagram

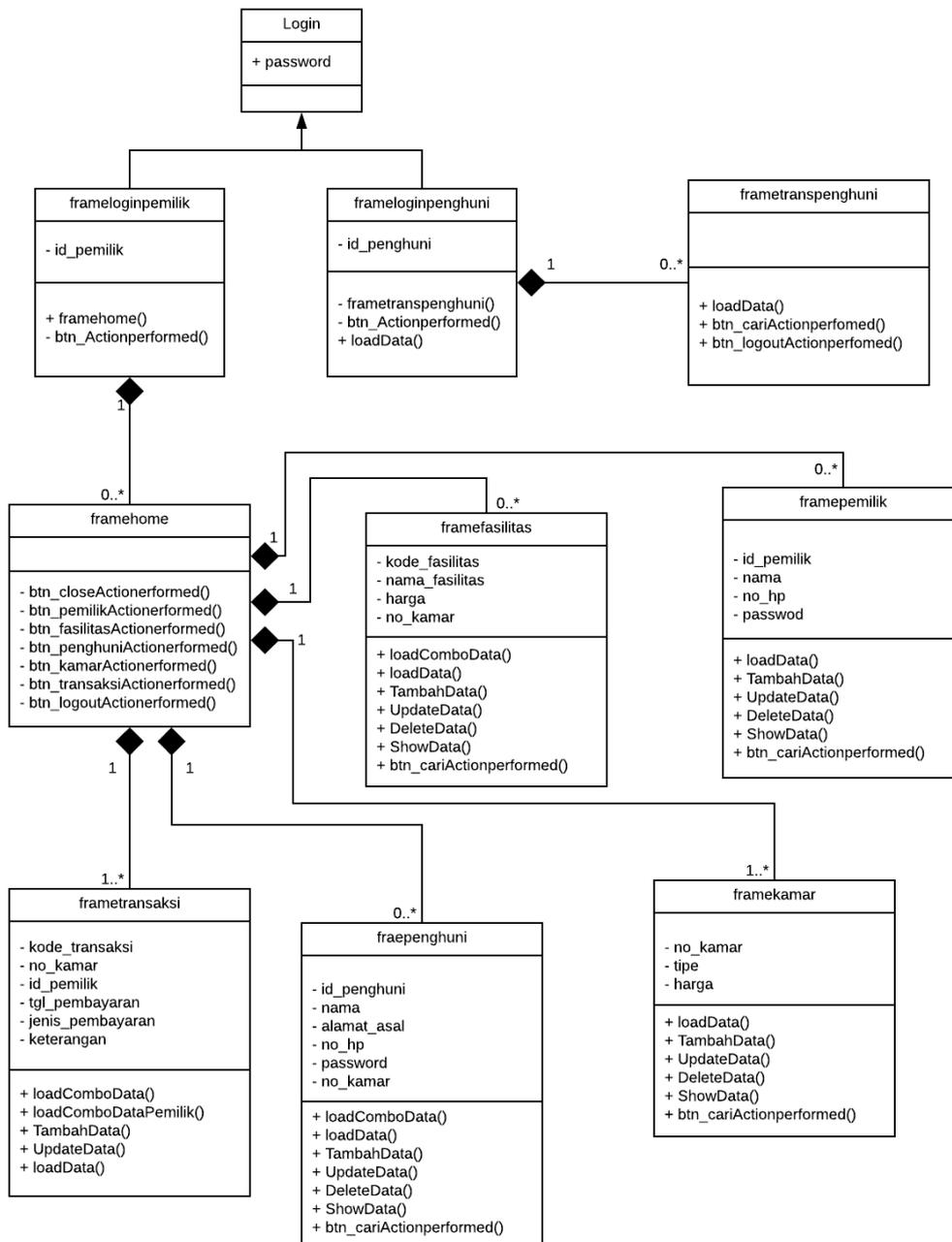
Class diagram atau diagram kelas menggambarkan struktur sistem dari segi pendefinisian kelas – kelas yang akan dibuat untuk membangun sistem. Kelas memiliki atribut dan metode atau operasi [24].

- a. Atribut merupakan variabel – variabel yang dimiliki oleh suatu kelas.
- b. Operasi atau metode adalah fungsi – fungsi yang dimiliki oleh suatu kelas. Diagram kelas dibuat agar programmer membuat kelas – kelas sesuai rancangan didalam diagram kelas agar antara dokumentasi perancangan dan perangkat lunak sesuai. Kelas – kelas yang ada pada struktur sistem harus dapat melakukan fungsi – fungsi sesuai dengan kebutuhan sistem sehingga pembuat

perangkat lunak dapat membuat kelas – kelas di dalam program perangkat lunak sesuai dengan perancangan diagram kelas.

Diagram ini memperlihatkan hubungan antar kelas dan penjelasan detail tiap-tiap kelas didalam model desain (dalam logical view) dari suatu sistem.

Berikut contoh class diagram yang dapat dilihat pada Gambar 2.10.



Gambar 2.10 Contoh Class Diagram

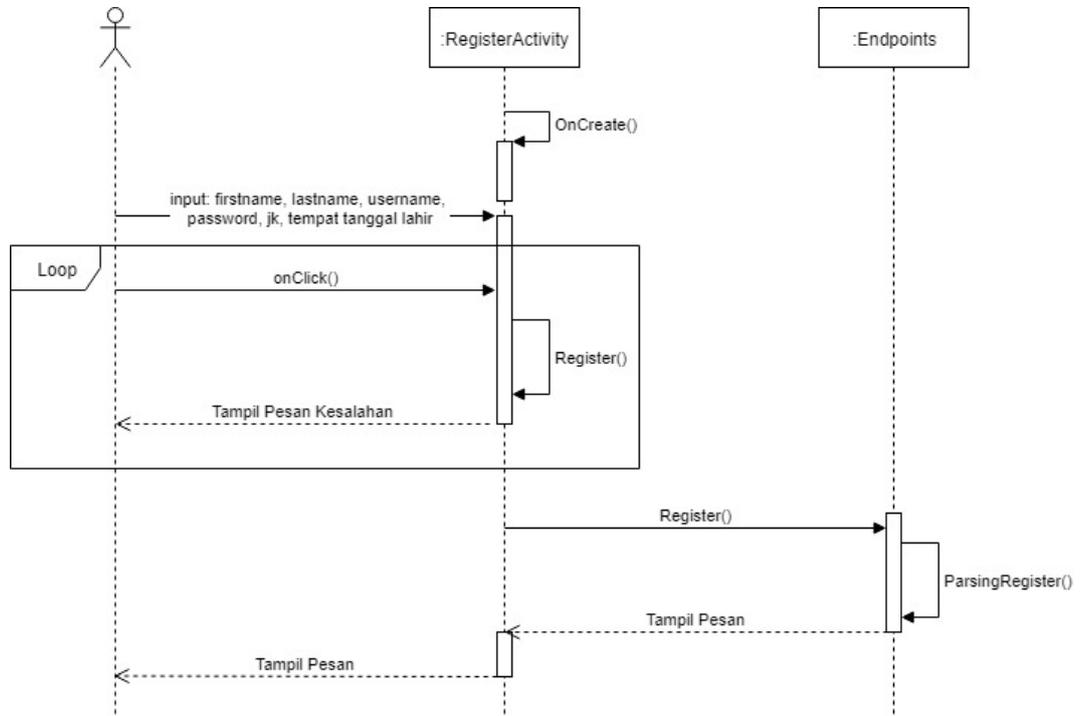
4. Sequence Diagram

Sequence diagram atau diagram sequence menggambarkan karakter objek pada use case dengan mendeskripsikan waktu hidup objek dan message yang dikirimkan dan diterima antar objek [24].

Untuk menggambarkan diagram sequence maka harus diketahui objek – objek yang terlibat dalam sebuah use case beserta metode – metode yang dimiliki kelas yang diinstansiasi menjadi objek tersebut.

Masing-masing objek, termasuk aktor, memiliki lifeline vertikal. Message digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Activation bar menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message. Objek-objek yang memiliki sifat khusus, standar UML mendefinisikan icon khusus untuk objek boundary, controller dan persistent entity[18].

Berikut contoh class diagram yang dapat dilihat pada Gambar 2.11.



Gambar 2.11 Contoh Sequence Diagram

2.2.20 Pengujian Alpha

Pengujian Alpha dilakukan untuk memastikan bahwa aplikasi dapat berjalan dengan benar sesuai dengan kebutuhan dan tujuan yang diharapkan. Pengujian yang digunakan pada aplikasi ini ada 2 (dua), yaitu teknik pengujian *white box testing* dan *black box testing*.

Pengujian *black-box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujian *black-box* memungkinkan perekayasa perangkat lunak mendapatkan serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk program. Pengujian *black-box* berusaha menemukan kesalahan dalam kategori sebagai berikut.

1. Fungsi-fungsi yang tidak benar atau hilang,
2. Kesalahan interface,
3. Kesalahan dalam struktur data atau akses database *eksternal*,
4. Kesalahan kinerja,
5. Inisialisasi dan kesalahan terminasi

Pengujian alpha berfokus pada persyaratan fungsional perangkat lunak. Fungsi-fungsi yang diuji meliputi tombol-tombol menu dengan kasus yang benar dan kasus yang salah.

2.2.21 Pengujian Beta

Pengujian Beta dilakukan secara langsung terhadap pengguna dengan menggunakan kuesioner mengenai kepuasan pengguna atas aplikasi yang telah dibangun. Adapun metode penilaian pengujian yang digunakan adalah metode kuantitatif berdasarkan data dari pengguna. Kuesioner disebarakan menggunakan teknik Simple Random Sampling dimana anggota sample dari populasi dilakukan secara acak tanpa memperhatikan strata yang ada dalam populasi.

2.2.22 Skala Likert

Rensis Likert telah mengembangkan sebuah skala untuk mengukur masyarakat di tahun 1932 yang sekarang terkenal dengan nama skala Likert. Skala Likert hanya menggunakan *item* yang secara pasti baik dan secara pasti buruk, tidak

dimasukan yang agak baik, yang agak kurang, yang netral, dan rangking lain di antara dua sikap yang pasti. *Item* yang pasti disenangi, disukai, yang baik diberi tanda negatif (-). Skor respon responden dijumlahkan dan jumlah ini merupakan total skor, dan total skor inilah ditafsirkan sebagai posisi responden dalam skala Likert [8].

Skala likert biasanya menyediakan lima pilihan skala untuk setiap tanggapan dari pertanyaan yang diajukan dan responden bebas menentukan tingkat persetujuan mereka terhadap pertanyaan tersebut. Untuk keperluan analisis kuantitatif, skala jawaban pada penelitian kali ini diberi skor seperti pada Tabel 2.2.

Tabel 2.2 Tabel Pemberian bobot nilai pada skala likert

Jenis Pertanyaan	Bobot Pendapat				
	SS	S	B	T	STS
Positif	5	4	3	2	1
Negatif	1	2	3	4	5

Instrumen penelitian yang menggunakan skala likert dapat dibuat dalam bentuk checklist ataupun pilihan ganda. Kemudian ketika jawaban sudah dihitung berdasarkan bobot nilainya masing - masing, maka dilanjutkan menghitung rata-rata dari setiap konten pernyataan dan indikator pernyataan itu sendiri dengan menggunakan rumus:

$$P = \frac{f}{n} \times 100\%$$

Keterangan :

P = Nilai Persentase yang dicari

f = Jumlah frekuensi dikalikan dengan nilai yang ditetapkan jawaban

n = Nilai tertinggi dikalikan dengan jumlah sampel

