BAB II
TINJAUAN PUSTAKA

1.1 Profil Perusahaan

Profil tempat penelitian tugas akhir ini akan menjelaskan mengenai sejarah, struktur organisasi serta visi dan misi di PT.Balai Lelang Bandung.

1.1.1 Sejarah Singkat Perusahaan

PT. BALAI LELANG BANDUNG merupakan perusahaan swasta yang bergerak di bidang penyediaan jasa pra lelang, lelang dan pasca lelang. PT. Balai Lelang Bandung didirikan untuk memenuhi kebutuhan masyarakat pada umumnya, dan dunia usaha pada khususnya, terhadap suatu proses penjualan asset yang efisien dan efektif. PT Balai Lelang Bandung berfungsi sebagai koordinator pelaksana seluruh rangkaian kegiatan lelang, mulai dari pemeriksaan dokumen, pengecekan fisik kondisi asset, penyiapan dokumen administrative dan hukum, pemasaran objek lelang, penyiapan saran dan prasrana lelang, hingga membantu kelancaran terpenuhinya seluruh hak dan kewajiban para pihak terkait setelah lelang selesai dilaksanakan.



Gambar 1.1 Logo PT.Balai Lelang Bandung

Dalam melakukan kegiatan-kegiatan dimaksud, PT Balai Lelang Bandung menjalin kerjasama yang erat dengan pemilik asset, kreditur (dalam hal objek lelang masih berstatus jaminan), kalangan investor, Kantor Badan Pertanahan serta Kantor Pelayanan Kekayaan Negara dan Lelang (KPKNL) setempat. Aset yang

dilelang melalui PT.Balai Lelang Bandung terdiri dari berbagai kategori, seperti asset tidak bergerak (tanah kosong, tanah berikut bangunan), asset bergerak (plants & machineries, inventaris atau perlengkapan kantor), saham-saham, piutang dan berbagai asset lainnya yang lazim diperdagangkan atau diperjual belikan. Dengan kata lain, pada prinsipnya seluruh benda yang memiliki nilai ekonomis (dapat di-nominasi atau ditetapkan harganya), sepanjang tidak dilarang oleh peraturan perundang-undangan dan/atau tidak bertentangan dengan kepatutan,dapat dijual melalui pelelangan. Lelang merupakan salah satu media yang sangat efektif untuk membentuk "harga terbaik" atas suatu objek. Hal ini terkait dengan sifat lelang yang terbuka untuk publik/masyarakat umum, terdapat kompetisi harga, serta dilakukan dengan proses dan prosedur yang relatif mudah diikuti. Sebelum menyelenggarakan lelang, PT Balai Lelang Bandung terlebih dahulu memastikan status dan kondisi asset dari sisi hukum serta melakukan pemasaran yang intensif dengan metode yang efektif, terarah dan menarik. Sebagai salah satu alternatif media jual beli, lelang juga memberikan kepastian hukum yang tinggi, karena pemenang lelang akan mendapatkan Risalah

Lelang yang diterbitkan oleh KPKNL, yang kekuatan pembuktiannya otentik dan sama kedudukannya dengan akta jual beli yang dikeluarkan oleh Pejabat Pembuat Akta Tanah (PPAT) untuk tanah. Balai Lelang pada umumnya dan PT.Balai Lelang Bandung pada khususnya juga merupakan instrumen yang sangat positif dan starategis untuk membantu perbankan menyelesaikan kredit bermasalah (non performing loan). Sejauh ini masih ada anggapan dan keluhan bahwa proses likiudasi jaminan melalui pengadilan negeri masih jauh dari prinsip cepat, efisien dan berbiaya murah sebgaimana diamanatkan oleh perundang-undangan. Dalam hal ini, perbankan dengan dukungan PT Balai Lelang Bandung telah memiliki mandat yuridis yang kuat (setingkat undang-undang) untuk melakukan pelelangan paksa atas jaminan-jaminan tanpa harus melalui fiat atau penetapan Pengadilan Negeri, yaitu berdasarkan Undang-Undang Nomor 4 tahun 1996 tentang Hak Tanggungan dan Undang-Undang nomor 42 tahun 1999 tentang Jaminan Fidusia. PT.Balai Lelang Bandung sangat memahami bahwa penyelesaian kredit bermasalah merupakan salah satu factor kunci penyehatan perbankan dan memerlukan

pendekatan-pendekatan yang cermat, efisien dan efektif. PT Balai Lelang Bandung memilik tenaga profesional yang sangat berpengalaman dalam bidang perbankan dan memiliki komitmen yang tinggi untuk menjadi mitra-mitra strategis bank yang dapat dipercaya. Saat ini PT.Balai LeLang Bandung berkantor pusat di JL. Gunung Batu 201 Ruko Maple Kav. G, Bandung, Jawa Barat 40175.

2.1.2 Visi Dan Misi Perusahaan

Berikut ini adalah visi dan misi dari PT.Balai Lelang Bandung:

a. Visi

PT. Balai Lelang Bandung bertekad untuk menjadi salah satu balai lelang terkemuka di Indonesia, baik dari segi volume usaha (business volume), maupun kualitas layanan (service quality).

b. Misi

Berikut ini adalah misi dari PT.Balai Lelang Bandung :

1. Memberikan kontribusi nyata dalam rangka mempercepat dan memperluas pertumbuhan ekonomi masyarakat, khususnya dalam bidang transaksi/jual-beli berbagai bentuk asset.
2. Dengan prinsip accountability, PT. Balai Lelang Bandung akan mengutamakan kepastian hukum bagi pemilik asset/penjual dan pembeli.
3. Dengan prinsip efficiency, PT. Balai Lelang Bandung bertekad untuk mengurangi beban ekonomis (biaya) likiudasi asset sehingga penjual memperoleh margin keuntungan yang maksimum.
4. Dengan prinsip effectiveness, PT. Balai Lelang Bandung menyediakan kecepatan likuidasi asset atau perolehan dana tunai bagi penjual.

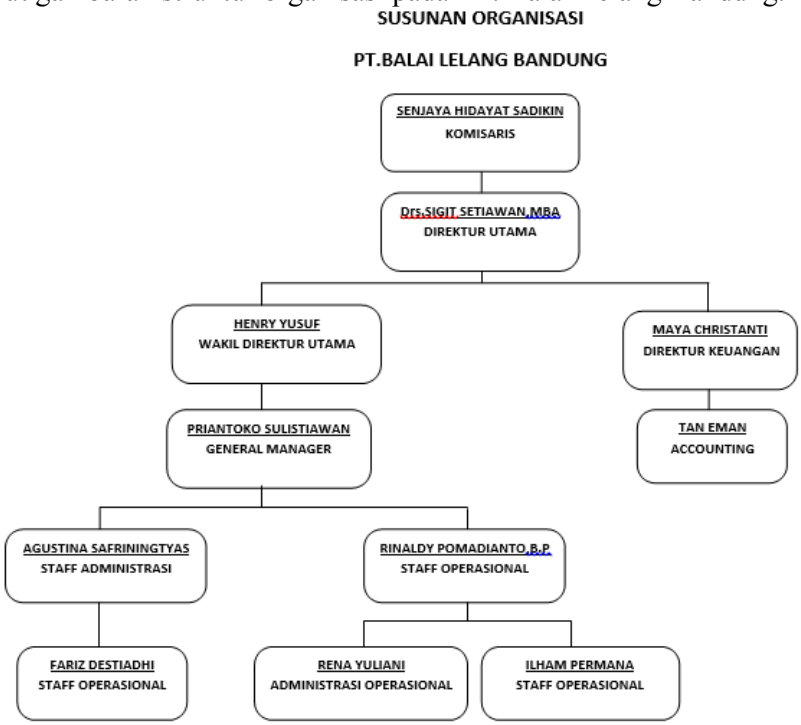
2.1.3 Struktur Organisasi Perusahaan

Struktur adalah cara sesuatu disusun atau dibangun. Sedangkan organisasi sendiri adalah suatu wadah berkumpulnya minimal dua orang untuk mencapai sebuah tujuan. Maka struktur organisasi adalah suatu susunan dan hubungan antara tiap bagian baik secara posisi maupun tugas yang ada pada perusahaan dalam menjalin kegiatan operasional untuk mencapai tujuan. Struktur organisasi merupakan pembagian, pengelompokan dan pengkoordinasian pekerjaan secara

formal. Dalam struktur organisasi terdapat beberapa elemen yang perlu diperhatikan dalam mendesain struktur organisasi yaitu : The Strategic Apex (Top Manager/Manajer Tingkat Puncak),The Midle Line (Midle Manager/Manajer Bagian Penghubung Top Manager dengan Employee), The Operating Core (Karyawan), The Technostructure (BagianAnalisis), The Support Staff (Staff Pendukung). Dan berikut beberapa desain organisasi :

1. Struktur Sederhana: berlaku bagi organisasi kecil.
2. Birokrasi Mesin: untuk organisasi dengan rutinitas berulang dan diatur oleh peraturan yang berlaku.
3. Birokrasi Profesional : untuk organisasi besar, dengan kegiatan rutin dan terdiri dari pada spesialis.
4. Struktur Divisional: untuk organisasi yang membutuhkan banyak pembagian divisi.
5. Adhocracy: untuk organisasi dengan tenaga profesional yang terkumpul pada suatu kegiatan tertentu.

Berikut gambaran struktur organisasi pada PT. Balai Lelang Bandung.



Gambar 2.2 Struktur Organisasi PT.Balai Lelang Bandung

Berikut ini adalah deskripsi jabatan di PT.Balai Lelang Bandung:

Tabel 2.1 Tabel Deskripsi Jabatan PT.Balai Lelang Bandung

No.	Jabatan	Tanggung Jawab	Uraian Tugas
1.	Direktur Utama	. Bertanggung jawab atas kegiatan atau pelaksanaan tugas yang berlangsung. . Bertanggung jawab atas control terhadap seluruh karyawan yang melaksanakan perintah sesuai atasannya.	Menyetujui kontrak kerja sama dengan perusahaan rekanan. Menghadiri sejumlah pertemuan berkaitan dengan kegiatan lelang yang akan dilaksanakan oleh PT. Balai Lelang Bandung. Menyaring transaksi yang akan dilakukan melalui kerja sama dengan perusahaan. Mengecek ulang isi perjanjian/Memorandum of Understanding yang diajukan terhadap perusahaan. Memutuskan setiap pilihan yang akan diambil oleh perusahaan dalam menghadapi setiap masalah atau proyek. Mengontrol laporan kerja dan keuangan perusahaan. Menerima saran dan kritik karyawan dalam rangka memajukan perusahaan dan mengikuti perkembangan yang berjalan. Menjalin hubungan baik dengan banyak pihak lain guna mendukung kelancaran jalannya perusahaan.

Tabel 2.1 Tabel Deskripsi Jabatan PT.Balai Lelang Bandung (Lanjutan)

No.	Jabatan	Tanggung Jawab	Uraian Tugas
			Melakukan kunjungan rutin terhadap partner/rekan perusahaan.).Melakukan briefing terhadap karyawan dan rekan satu kantor dalam rangka memberikan motivasi kerja terhadap karyawan.
2.	Wakil Direktur Utama	Membantu tugas Direktur Utama baik dalam segala hal yang bersangkutan dengan operasional perusahaan.	Menggantikan posisi dengan segala tugas yang seharusnya dikerjakan Direktur Utama saat tidak berada di tempat. Membantu Direktur Utama dalam memenuhi jadwal undangan rapat atau pertemuan lain yang tidak bisa dihadiri. Membantu menjalin hubungan dengan relasi sebanyak mungkin agar dapat menjalin kerja sama dengan banyak pihak. Melakukan pengontrolan secara langsung terhadap proses atau hasil kerja karyawan. Membantu Direktur Utama dalam mengambil setiap keputusan yang akan ditentukan.

Tabel 2.1 Tabel Deskripsi Jabatan PT.Balai Lelang Bandung (Lanjutan)

No.	Jabatan	Tanggung Jawab	Uraian Tugas
3.	Direktur Keuangan	Bertanggung jawab atas segala data dan laporan keuangan. Bertanggung jawab atas hasil pemeriksaan mengenai segala transaksi keuangan perusahaan.	Memeriksa laporan keuangan dari bagian administrasi. Menganalisa setiap dana yang keluar masuk perusahaan. Mengendalikan biaya yang dikeluarkan perusahaan. Mengatur biaya tenaga kerja karyawan. Menyusun anggaran perusahaan. Melaksanakan briefing atasannya
4.	Manajer HRD	.Bertanggung jawab atas suatu perencanaan, pengorganisasian, pelaksanaan dan pengawasan sumber daya manusia. .Bertanggung jawab atas pengadaan, pengembangan, konpensasi dan pemberhentian hubungan kerja.	Mengelola penerimaan karyawan baru yang akan bekerja di perusahaan serta melakukan penyaringan terhadap setiap calon karyawan. Employee of the month & employee of the year Pembuatan dan pendistribusian formulir employe of the month & employee of the year. Rekapitulasi hasil penilaian Penjadwalan acara dan pembagian penghargaan Memperbanyak diktat dan soal-soal yang berkaitan dengan training. Membuat jadwal liburan bersama seluruh karyawan kantor dan pengurus perusahaan.

Tabel 2.1 Tabel Deskripsi Jabatan PT.Balai Lelang Bandung (Lanjutan)

No.	Jabatan	Tanggung Jawab	Uraian Tugas
			). Mengurus dan mengontrol kehadiran seluruh karyawan. 0. Mengatur jadwal kerja dan cuti karyawan. 1. Menerima dan mengatasi keluhan karyawan sehubungan pekerjaan. 2. Membuat peraturan untuk mengontrol disiplin kerja dan kehadiran karyawan.
5.	Staff Administrasi	Bertanggung jawab mengendalikan dan mengatur keuangan kas dan bank yang digunakan setiap hari. Bertanggung jawab terhadap catatan mengenai arus kas yang keluar dan masuk dapat disusun rapih untuk kemudian dilaporkan kepada direktur keuangan.	Memegang sejumlah uang tunai yang digunakan sebagai kas kecil dan sejumlah buku cek yang dapat digunakan untuk pembayaran dengan jumlah besar. Mengelola dana kas kecil dengan bijkasana. Mengelola transaksi dan segala dana yang terhimpun melalui bank. Menerima setoran dana pembayaran atas jasa yang diberikan. Memberikan dana yang dibutuhkan untuk kegiatan operasional perusahaan dengan mengumpulkan segala bukti transaksi. Membagikan gaji karyawan yang telah dihitung sebelumnya. Mengelola penagihkan piutang perusahaan dan pembayar kewajiban/hutang perusahaan

Tabel 2.1 Tabel Deskripsi Jabatan PT.Balai Lelang Bandung (Lanjutan)

No.	Jabatan	Tanggung Jawab	Uraian Tugas
			dengan persetujuan direktur keuangan. Mengikuti briefing rutin. Mengurus absensi daftar hadir karyawan untuk penghitungan gaji karyawan
6.	Staff Pengurus Dokumen	Bertanggung jawab terhadap seluruh kelengkapan, keabsahan dan kejelasan dokumen yang keluar dan masuk perusahaan untuk kelancaran setiap proyek.	Memeriksa dokumen yang akan digunakan untuk kegiatan lelang. Membuat arsip atas setiap dokumen yang telah digunakan dan menyimpannya dengan baik. Mengerjakan dokumen yang digunakan dalam proyek. Melengkapi segala kekurangan pada setiap dokumen agar tidak ada masalah pada saat pelaksanaan. Mengikuti briefing.
7.	Staff Pengurus Dokumen	Bertanggung jawab terhadap seluruh kegiatan yang berhubungan dengan lapangan termasuk pemasaran. Bertanggung jawab terhadap masalah-masalah	Melaksanakan survey dan pengecekan terhadap barang yang akan dijual secara lelang melalui perusahaan ini. Mendatangi setiap client dan rekanan yang bekerjasama agar dapat menjalin komunikasi dan hubungan dengan baik. Melakukan pemasaran kepada tempat-tempat yang potensial bekerjasama menggunakan jasa

Tabel 2.1 Tabel Deskripsi Jabatan PT.Balai Lelang Bandung (Lanjutan)

No.	Jabatan	Tanggung Jawab	Uraian Tugas
		yang terjadi di lapangan.	pengurusan lelang oleh perusahaan ini. 4. Melaksanakan segala kegiatan di luar kantor untuk menyelesaikan tugas kantor.
6.	Staff Operasional	Bertanggung jawab terhadap seluruh kegiatan yang berhubungan dengan operasional pengurusan Pra Lelang, Lelang dan Pasca Lelang.	Menyiapkan berkas atau segala sesuatu yang dibutuhkan untuk melaksanakan kegiatan lelang. Menangani permintaan client dan menyampaikan hal tersebut kepada dirut. Menyiapkan segala kebutuhan teknis pelaksanaan lelang. Melaksanakan kegiatan Pra Lelang. Memeriksa ulang dokumen yang sudah disiapkan oleh staff dokumen lelang. Mengelola data calon pelanggan yang diberikan staff lapangan/marketing.

2.2 Landasan Teori

Landasan teori merupakan penjelasan berbagai konsep dasar dan teori-teori yang berkaitan daam pembangunan aplikasi pelelangan barang online berbasis android.

2.3 Lelang

Lelang pada umumnya adalah suatu sarana untuk mempertemukan penjual dan pembeli dengan tujuan menentukan harga yang wajar bagi suatu barang. Menurut ketentuan Pasal 1 *Vendu Reglement* ordonansi 28 Februari 1908,

menentukan bahwa yang dimaksud dengan "penjualan umum" (*openbare verkopingen)* ialah pelelangan dan penjualan barang yang diadakan di muka umum dengan penawaran harga yang makin meningkat, dengan persetujuan harga yang makin menurun atau dengan pendaftaran harga, atau di mana orang-orang yang diundang atau sebelumnya sudah diberi tahu tentang pelelangan atau penjualan, atau kesempatan yang diberikan kepada orang-orang yang berlelang atau yang membeli untuk menawar harga, menyetujui harga atau mendaftarkan[4].

2.4 Proses Pelaksanaan Lelang

Lingkup pekerjaan mengacu kepada perundang-undangan yang khusus mengatur penyelenggaraan kegiatan usaha balai lelang. Secara garis besar proses pelaksanaan lelang pada umumnya adalah sebagai berikut:

1. OPEN HOUSE

a. Peminat diwajibkan untuk terlebih dahulu melihat atau mengecek barang – barang yang akan dilelang.

b. Peminat dapat memeriksa kondisi fisik dan kelengkapan dokumen objek lelang tersebut tanpa membongkar pasang dan sejenisnya yang dapat merusak objek lelang.

c. Peminat yang ingin mengikuti lelang wajib menyetor uang jaminan lelang.

d. Peminat yang sudah menyetor harus menunjukan slip setoran asli beserta kartu tanda pengenal.

2. PELAKSANAAN LELANG

a. Pelaksanaan lelang akan dilakukan secara lisan oleh pejabat lelang yang akan dibantu oleh pemandu lelang.

b. Harga penawaran tertinggi yang telah mencapai harga limit, ditetapkan sebagai pemenang lelang dengan ditandai ketokan palu oleh pejabat lelang.

3. PEMENANG LELANG

a. Pemenang lelang wajib melunasi seluruh total harga lelang dari lot yang dimenangkan paling lambat 3 hari setelah pengumuman pemenang lelang. Apabila telah lewat dari waktu yang telah ditentukan maka pemenang dianggap

mengundurkan diri dan biaya setoran yang sebelumnya telah dibayar tidak dapat dikembalikan.

b. Peserta yang tidak memenangkan lelang dapat mengambil kembali uang jaminannya pada saat lelang selesai. Uang jaminan dikembalikan utuh tanpa potongan dalam bentuk apapun.

4. PENGAMBILAN OBJEK LELANG

a. Pemenang lelang yang sudah membayar lunas wajib mengambil barang yang telah dimenangkan paling lambat 3 hari setelah proses pembayaran lunas. Apabila dalam waktu yang sudah ditentukan belum juga mengambil barang yang telah dimenangkan maka peserta dikenakan biaya Rp.150.000 untuk prosespenitipan atau keamanan barang.

b. Apabila dalam jangka waktu lebih dari 10 hari barang belum diambil oleh pemenang lelang maka instansi yang melaksanakan lelang tersebut berhak untuk menjual dan berhak untuk mengambil biaya dari hasil penjualan tersebut.

c. Bagi peserta yang tidak dapat megambil secara langsung barang yang telah dimenangkan maka instansi yang bertanggung jawab melaksanakan lelang tersebut berhak untuk membantu malkukan proses pengiriman dengan syarat biaya yang dikeluarkan menjadi tanggung jawab penuh peserta lelang.

2.5 Aplikasi Mobile

Aplikasi *Mobile* adalah sebuah aplikasi yang memungkinkan Anda melakukan mobilitas dengan menggunakan perlengkapan seperti PDA, telepon seluler atau *Handphone*. Dengan menggunakan aplikasi *mobile*, Anda dapat dengan mudah melakukan berbagai macam aktifitas mulai dari hiburan, berjualan, belajar, mengerjakan pekerjaan kantor, *browsing* dan lain sebagainya[5]. Contohnya mobile programing untuk ponsel:

1. Javafx mobile, digunakan di smartphone Android.
2. J2ME (Java 2 Mobile Edition) yang paling umum dan paling banyak yang support, blackberry pun masih menggunakan J2ME dengan tambahan API. J2ME adalah satu set spesifikasi dan teknologi yang fokus kepada perangkat konsumen. Perangkat ini memiliki jumlah memori yang terbatas, menghabiskan sedikit daya dari baterei, layar yang kecil dan bandwith jaringan yang rendah.

3. C++ dalam symbian framework, ini khusus untuk handphone dengan OS symbian.
4. C# .NET, untuk handphone dengan OS Windows mobile
5. Flash Lite, game flash di mobile.

Situs web (bahasa Inggris: *website*) adalah suatu halaman web yang saling berhubungan yang umumnya berada pada peladen yang sama berisikan kumpulan informasi yang disediakan secara perorangan, kelompok, atau organisasi. Sebuah situs web biasanya ditempatkan setidaknya pada sebuah server web yang dapat diakses melalui jaringan seperti Internet, ataupun jaringan wilayah lokal (LAN) melalui alamat Internet yang dikenali sebagai URL. Gabungan atas semua situs yang dapat diakses publik di Internet disebut pula sebagai World Wide Web atau lebih dikenal dengan singkatan WWW.

Meskipun setidaknya halaman beranda situs Internet umumnya dapat diakses publik secara bebas, pada prakteknya tidak semua situs memberikan kebebasan bagi publik untuk mengaksesnya, beberapa situs web mewajibkan pengunjung untuk melakukan pendaftaran sebagai anggota, atau bahkan meminta pembayaran untuk dapat menjadi aggota untuk dapat mengakses isi yang terdapat dalam situs web tersebut, misalnya situs-situs yang menampilkan pornografi, situs-situs berita, layanan surel (*e-mail*), dan lain-lain. Pembatasan-pembatasan ini umumnya dilakukan karena alasan keamanan, menghormati privasi, atau karena tujuan komersil tertentu.

Sebuah halaman web merupakan berkas yang ditulis sebagai berkas teks biasa (plain text) yang diatur dan dikombinasikan sedemikian rupa dengan instruksi-instruksi berbasisHTML, atau XHTML, kadang-kadang pula disisipi dengan sekelumit bahasa skrip. Berkas tersebut kemudian diterjemahkan oleh peramban web dan ditampilkan seperti layaknya sebuah halaman pada monitor komputer.

Halaman-halaman web tersebut diakses oleh pengguna melalui protokol komunikasi jaringan yang disebut sebagai HTTP, sebagai tambahan untuk meningkatkan aspek keamanan dan aspek privasi yang lebih baik, situs web dapat pula mengimplementasikan mekanisme pengaksesan melalui protokol HTTPS.

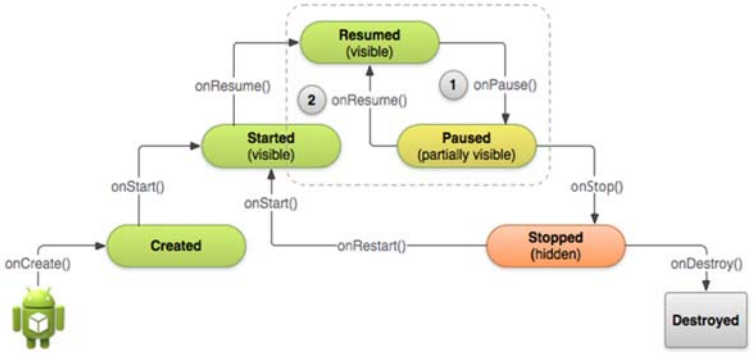
2.6 Android

Android adalah sistem operasi bersifat *Open Source* berbasis Linux dirancang untuk perangkat seluler layar sentuh seperti telepon pintar dan komputer tablet. Android awalnya dikembangkan oleh Android, Inc., dengan dukungan finansial dari Google, yang kemudian membelinya pada tahun 2005. Sistem operasi ini dirilis secara resmi pada tahun 2007, bersamaan dengan didirikannya *Open Handset Alliance*, *konsorsium* dari perusahaan-perusahaan perangkat keras, perangkat lunak, dan telekomunikasi yang bertujuan untuk memajukan standar terbuka perangkat seluler. Ponsel Android pertama mulai dijual pada bulan Oktober 2008 [6]. Android memiliki OS yang sangat baik, cepat dan kuat serta memiliki antarmuka pengguna intuitif yang dikemas dengan pilihan dan fleksibilitas.

1. Android SDK (*Software Development Kit*) menyediakan *tools* dan API yang diperlukan untuk mengembangkan aplikasi pada platform android dengan menggunakan bahasa pemrograman Java.

2. Android memiliki paradigma pemrograman lain tidak seperti paradigma pemrograman biasa di mana aplikasi yang dijalankan pada fungsi main(), sistem android menjalankan kode dalam *method Activity* dengan menerapkan metode *callback* tertentu yang sesuai dengan tahap tertentu dari siklus hidup. Setiap aplikasi yang berjalan dalam sistem operasi android memiliki siklus hidup yang berbeda dengan aplikasi desktop atau *web*, Hal ini dikarenakan aplikasi *mobile* memiliki tingkat interupsi proses yang lumayan tinggi seperti ketika *handling* panggilan masuk aplikasi diharuskan menghentikan proses sementara, Penerapan siklus hidup juga berguna untuk memastikan aplikasi tidak menghabiskan sumber daya baterai pengguna.

3. Di dalam android terdapat *activity* dimana komponen ini memberi interaksi antara *user* dan aplikasi yang dibangun melalui *user interface*. *Activity* ini memiliki siklus hidup yang dinamakan *Android Life Cycle*. *Flowchart* siklus hidup tersebut dapat dilihat pada Gambar.



Gambar 2.3 Android Life Cycle

4. Terdapat beberapa *state* dalam siklus hidup android yang terjadi seperti diilustrasikan pada Gambar 2.3 *Android Life Cycle*, akan tetapi hanya beberapa dari state tersebut yang menjadi statis diantaranya:

1. Resumed

Resumed terjadi ketika aplikasi berjalan setelah state paused. State ini akan menjalankan perintah program yang ditulis pada method onResume().

2. Paused

Dalam keadaan ini aktivitas yang terjadi dihentikan secara sementara tetapi masih terlihat oleh pengguna karena terdapat proses yang memiliki prioritas lebih tinggi seperti panggilan telepon. Aplikasi tidak dapat menjalankan perintah apapun ataupun menampilkan apapun dalam state ini.

3. Stopped

Dalam keadaan ini, aplikasi benar-benar tidak ditampilkan dan tidak terlihat oleh pengguna tetapi masih meninggalkan service dibackground.

5. State lain seperti Created dan Started bersifat sementara dan sistem dengan cepat menjalankan state berikutnya dengan memanggil metode *life cycle callback* berikutnya. Artinya, setelah sistem OnCreate() dipanggil, dengan cepat sistem akan memanggil method OnStart(), krmudian diikuti oleh onResume().

6. Android memiliki beberapa fitur utama yang sering digunakan dalam proses pembangunan aplikasi diantaranya adalah:

1. Multi-proses dan *App Widgets*

Sistem operasi android tidak melarang prosesor menjalankan lebih dari satu aplikasi dalam satu waktu. Sistem operasi android dapat mengatur aplikasi dan thread yang berjalan secara *multitasking.* Keuntungan yang didapat adalah ketika aplikasi berjalan dan berinteraksi dengan pengguna di layer depan sistem operasi, proses dari aplikasi lain dapat berjalan untuk melakukan pembaruan informasi. Sebagai contoh misalnya ketika pengguna memainkan game, proses lain dapat berjalan di belakang aplikasi seperti memeriksa harga saham dan memunculkan peringatan.

App Widgets adalah mini aplikasi yang dapat *embedded* dalam aplikasi seperti home screen. App widgets dapat menjalankan proses *request* seperti musik streaming atau mendeteksi suhu ruangan secara *background*.

Multi-proses dapat memberikan manfaat berupa *user experience* yang lebih banyak, namun penggunaan fitur tersebut dapat menghabiskan banyak energi baterai jika penggunaan tidak benar.

2. *Touch Gestures* dan *Multi-touch*

Touchscreen adalah *user interface* intuitif yang digunakan banyak *smartphone* di dunia. Dengan fitur ini interaksi dapat dibuat lebih mudah karena cukup dengan menggunakan jari tangan. *Multi-touch* adalah kemampuan yang dapat melakukan tracking lebih dari satu tangan dalam satu waktu, Fitur ini sering digunakan untuk interaksi memperbesar atau memutar objek. Selain itu pengembang dapat membuat interaksi baru dengan memanfaatkan fitur tersebut.

3. Hard dan *Soft Keyboard*

Salah satu fitur pada perangkat smartphone adalah tombol fisik dan non fisik, tombol fisik digunakan untuk navigasi pendukung dalam pengoperasian android. Pengembang aplikasi tidak perlu secara manual untuk mengintegrasikan tombol tersebut dalam aplikasi. Tombol non fisik adalah tombol yang dibuat oleh sistem operasi seperti *virtual keyboard*, dan tombol navigasi aplikasi.

2.6.1 Android SDK

Android SDK adalah tools API (*Application Programming Interface*) yang diperlukan untuk mulai mengembangkan aplikasi pada 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[7]. Saat ini disediakan Android SDK (*Software Development Kit*) sebagai alat bantu dan API untuk mulai mengembangkan aplikasi pada platform Android menggunakan bahasa pemrograman Java. sebagai platform aplikasi netral, Android memberi kesempatan untuk membuat aplikasi yang kita butuhkan yang bukan merupakan aplikasi bawaan *Handphone/Smartphone*. Beberapa fitur - fitur Android yang paling penting adalah:

a. Framework Aplikasi yang mendukung penggantian komponen dan *reusable*

b. Mesin *virtual Dalvik* dioptimalkan untuk perangkat *mobile*

c. *Integrated browser* berdasarkan *engine open source WebKit*

d. Grafis yang dioptimalkan dan didukung oleh libraries grafis 2D, grafis 3D berdasrkan spesifikasi opengl ES 1,0 (Opsional akselerasi hardware)

e. SQLite untuk penyimpanan data (*database*)

f. Media Support yang mendukung audio, video, dan gambar.

g. Bluetooth, EDGE, 3G, dan WiFi (tergantung hardware)

h. Kamera, GPS, kompas, dan *accelerometer* (tergantung hardware)

i. Lingkungan *Development* yang lengkap dan kaya termasuk perangkat emulator, tools untuk debugging, profil dan kinerja memori, dan plugin untuk IDE Eclipse.

2.7 JAVA

Java adalah bahasa pemrograman serbaguna. Java dapat digunakan untuk membuat suatu program sebagaimana Anda membuatnya dengan bahasa seperti Pascal atau C++. Yang lebih menarik, Java juga mendukung sumber daya internet yang saat ini populer, yaitu *World Wide Web* atau yang sering disebut web. Java juga mendukung aplikasi klient/server, baik dalam jaringan local (LAN) maupun jaringan berskala luas (WAN)[8].

Java dikembangkan oleh Sun Microsystems pada Agustus 1991, dengan nama semula Oak. Konon Oak adalah pohon semacam Jati yang terlihat dari jendela tempat pembuatnya, James Gosling, bekerja. Ada yang mengatakan bahwa Oak adalah singkatan dari "Object Application Kernel", tetapi ada yang menyatakan hal itu muncul setelah nama Oak diberikan. Pada Januari 1995, karena nama Oak dianggap kurang komersial, maka diganti menjadi Java.

Dalam sejumlah literatur disebutkan bahwa Java merupakan hasil perpaduan sifat dari sejumlah bahasa pemrograman, yaitu C, C++, Object-C, SmallTalk, dan Common LISP. Selain itu Java juga dilengkapi dengan unsur keamanan. Yang tak kalah penting adalah bahwa Java menambahkan paradigma pemrograman yang sederhana. Jika telah mengenal C atau C++, yang mengandalkan pointer, Java justru meninggalkannya sehingga anda akan memperoleh kemudahan saat menggunakannya.

Program java bersifat tidak bergantung pada *platform*, artinya, Java dapat dijalankan pada sembarang komputer dan bahkan pada sembarang sistem operasi. Ketidakbergantungan pada platform sering dinyatakan dengan istilah portabilitas. Yang menarik, tingkat portabilitas java tidak hanya sebatas pada program sumber (*source code*), melainkan juga pada tingkat kode biner yang disebut *bytecode*. Dengan demikian bila telah mengkompilasi program java pada komputer berbasis operasi Windows, dapat juga menjalankan hasil kompilasi pada Macintosh secara langsung, tanpa perlu mengkompilasi ulang.

2.8 ECLIPSE JUNO

Eclipse adalah sebuah IDE (*Integrated Development Environment*) untuk mengembangkan perangkat lunak dan dapat dijalankan di semua platform (platform-independent)[9]. Berikut ini adalah sifat dari Eclipse:

1. Multi-platform: Target sistem operasi *Eclipse* adalah *Microsoft Windows*, *Linux, Solaris, AIX, HP-UX* dan *Mac OS X.*

2. Mulit-language: *Eclipse* dikembangkan dengan bahasa pemrograman Java, akan tetapi *Eclipse* mendukung pengembangan aplikasi berbasis bahasa pemrograman lainnya, seperti *C/C++, Cobol, Python, Perl, PHP*, dan lain sebagainya.

3. Multi-role: Selain sebagai IDE untuk pengembangan aplikasi, *Eclipse* pun bisa digunakan untuk aktivitas dalam siklus pengembangan perangkat lunak, seperti dokumentasi, test perangkat lunak, pengembangan web, dan lain sebagainya.

Eclipse pada saat ini merupakan salah satu IDE favorit dikarenakan gratis dan *open source*, yang berarti setiap orang boleh melihat kode pemrograman perangkat lunak ini. Selain itu, kelebihan dari *Eclipse* yang membuatnya populer adalah

kemampuannya untuk dapat dikembangkan oleh pengguna dengan komponen yang dinamakan *plug-in.*

2.9 MySQL

Menurut Abdul Kadir mendefinisikan bahwa : MySQL merupakan *software* yang tergolong database *server* dan bersifat *Open Source*. *Open Source* menyatakan bahwa software ini di lengkapi oleh *source code* (kode yang di pakai untuk membuat *MySQL*), selain tentu saja bentuk *executable*-nya atau kode dapat di jalankan secara langsung di dalam sistem operasi, dan bias diperoleh dengan cara mengunduh di internet secara gratis. Hal lainnya adalah *MySQL* juga bersifat multiplatform. *MySQL* dapat dijalankan pada berbagai sistem operasi.[10].

2.9.1 PHP

PHP adalah program aplikasi yang bersifat server side, artinya hanya dapat berjalan pada sisi server saja dan tidak dapat berfungsi tanpa adanya sebuah server di dalamnya. PHP juga bukan bahasa pemograman yang lengkap[10].

Maksudnya program ini tidak menyertakan compiler tersendiri yang membuat program hasilnya menjadi program .exe yang dapat di jalankan tersendiri. PHP (Hypertext Preprocessor) merupakan sebuah bahasa pemograman scripting berlisensi Open Source. Script ini dapat bercampur dengan script Tag HTML sehingga karena kemampuannya tersebut, ia disebut bahasa yang embeded pada Tag HTML.

2.9.2 JSON

JSON (*JavaScript Object Notation*) adalah format pertukaran data yang ringan, mudah dibaca dan ditulis oleh manusia, serta mudah diterjemahkan dan dibuat (*generated*) oleh komputer[10]. JSON terbuat dari dua struktur yaitu:

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

JSON menggunakan bentuk sebagai berikut:

1. Objek

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

2. Larik

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

3. Nilai

Nilai (value) dapat berupa sebuah string dalam tanda kutip ganda, atau angka, atau true atau false atau null, atau sebuah objek atau sebuah larik. Struktur-struktur tersebut dapat disusun bertingkat.

4. String

String adalah kumpulan dari nol atau lebih karakter Unicode, yang dibungkus dengan tanda kutip ganda. Di dalam string dapat digunakan backslash escapes "\" untuk membentuk karakter khusus. Sebuah karakter mewakili karakter tunggal pada string. String sangat mirip dengan string C atau Java.Angka, Format oktal dan heksadesimal tidak digunakan.

2.10 Structure Analysis dan Design

Struktur analisis dan desain adalah sebuah metodologi yang di gunakan pada rekayasa perangkat lunak untuk mendeskripsikan sistem kearah fungsional. Pendekatan ini memecahkan masalah-masalah dalam aktifitas bisnis menjadi bagian-bagian kecil yang dapat disatukan kembali menjadi satu kesatuan yang utuh untuk memecahkan masalah

Tools yang dapat digunakan pada pendekatan analisis pengembangan sistem secara terstruktur adalah:

1. ERD (*Entity Relationship Diagram*)

Merupakan satu model jaringan yang menggunakan susunan data yang disimpan dalam sistem. ERD merupakan model jaringan data yang menekankan pada struktur dan hubungan antardata Elemen-elemen ERD:

1. Entitas

Entity (entitas) adalah sesuatu apa saja yang ada di dalam sistem, nyata maupun abstrak di mana data tersimpan atau di mana terdapat data. Entitas diberi nama dengan kata benda dan dapat dikelompokkan dalam 4 (empat) kelas, yaitu : *role* (peran), *events* (kejadian), *locations* (lokasi), *tangible things* / *concepts* (sesuatu yang tidak nyata / konsep). Contoh: pegawai, pembelian, kampus, buku, pembayaran, mahasiswa dan dosen.

2. Relationship

Relationship adalah hubungan alamiah yang terjadi antara entitas. Contohnya: Entitas Mahasiswa dengan NIM = "14534" dan NamaMhs = "Dodo" yang mempunyai relasi dengan Entitas Kuliah dengan KodeKul = "SI-140" dan NamaMK = "Basis Data", sehingga struktur data dari Relasi ini bahwa mahasiswa tersebut mengambil mata kuliah pada suatu perguruan tinggi.

3. Atribut

Deskripsi kelompok data yang mempunyai karakteristik yang sama (data yang mendeksripsikan entity dan relationship), merupakan field yang akan disimpan.

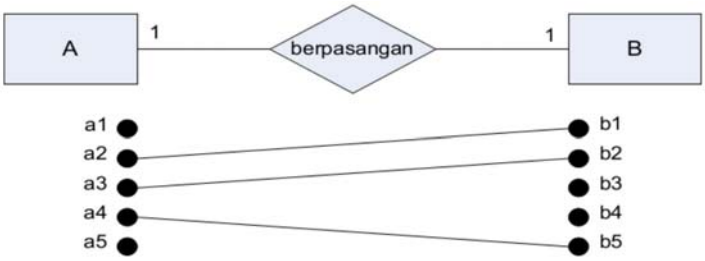
Hubungan antara entitas akan menyangkut dua komponen yang menyatakan jalinan ikatan yang terjadi, yaitu[16]:

1. Derajat (kardinalitas).

Kardinalitas menyatakan jumlah anggota entitas yang terlibat di dalam relasi yang terjadi. Dalam hal ini relasi yang terjadi akan membentuk relasi hubungan (relationship instances). Derajat kardinalitas dibagi menjadi 3, yaitu:

A. Derajat kardinalitas 1 : 1

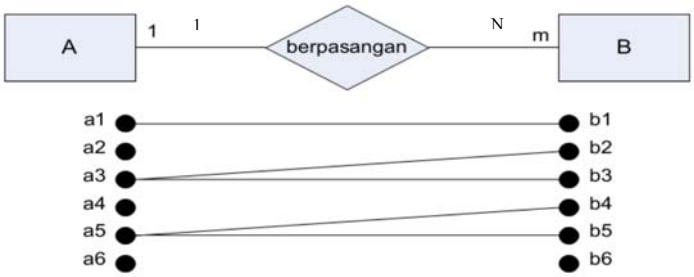
Derajat hubungan 1:1 terjadi bila setiap anggota entitas A hanya boleh berpasangan dengan satu anggota dari entitas B, dan sebaliknya tiap anggota entitas B hanya boleh berpasangan dengan satu anggota dari entitas A. Contoh dari kardinalitas 1:1 dapat dilihat pada Gambar 2.3 Derajat kardinalitas 1:1.



Gambar 2.4 Derajat kardinalitas 1:1

B. Derajat kardinalitas 1:N

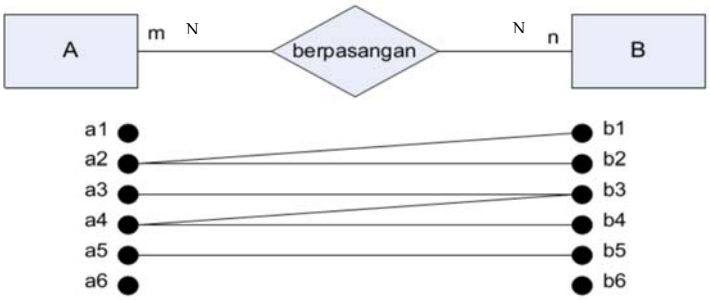
Derajat hubungan ini terjadi bila tiap anggota entitas A boleh berpasangan dengan lebih dari satu anggota entitas B. Sebaliknya tiap anggota entitas B hanya boleh berpasangan dengan satu anggota entitas A. Contoh dari kardinalitas 1:N dapat dilihat pada Gambar 2.4 Derajat kardinalitas 1:N.



Gambar 2.5 Derajat kardinalitas 1:N

C. Derajat kardinalitas N:N

Derajat hubungan antar entitas N:N terjadi bila tiap anggota entitas A dapat berpasangan dengan lebih dari satu anggota entitas B. Sebaliknya setiap anggota entitas B juga dapat berpasangan dengan lebih dari satu anggota entitas A. Contoh dari kardinalitas N:N dapat dilihat pada Gambar 2.5 Derajat kardinalitas N:N.



Gambar 2.6 Derajat kardinalitas N:N

D. Partisipasi hubungan.

Partisipasi hubungan (*participation*) menyatakan sifat keterlibatan tiap anggota entitas dalam relasi terjadinya hubungan. Partisipasi atau keterlibatan tiap anggota entitas dalam membentuk instance hubungan dapat bersifat wajib (mandatory) dan tidak wajib (non mandatory). Dalam pemodelan data, interpretasi jenis partisipasi hubungan dituliskan dalam aturan data. Contoh misal hubungan antara entitas dosen dan mata kuliah. Berikut keterhubungan kardinalitas dan partisipasi pada tabel 2.2.

Tabel 2.2 Kardinalitas dan partisipasi

Kardinalitas	Partisipasi
1-1 (one to one)	0..1 (optional one)
1-N (one to many)	0..N atau 1..N (optinal many)
N-1 (many to one)	1 (mandatory one)
N-N (many to many)	N (mandatory many)

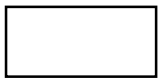
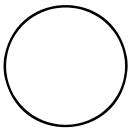
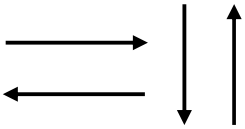
2. Diagram Konteks

Diagram konteks adalah diagram yang menggambarkan sebuah sistem secara menyeluruh yang selanjutnya akan lebih didetailkan oleh DFD. Konteks diagram juga biasa disebut sebagai DFD level .

3. Data Flow Diagram (DFD)

Data Flow Diagram merupakan serangkaian diagram yang menggambarkan kegiatan-kegiatan yang ada dalam satu sistem. Teknik pembangunan DFD dimulai dengan menggambarkan sistem secara global dan dilanjutkan dengan analisis masing-masing bagian. Pada awalnya digambarkan konteks diagram yang menggambarkan sebuah sistem secara menyeluruh yang akan diinvestigasi. Konteks diagram tersebut dapat dikatakan sebagai DFD level 0. Analisis sistem yang lebih detail selanjutnya dapat dilakukan dengan menggambarkan DFD level 1,2 dan seterusnya.Simbol-simbol yang digunakan di diagram konteks dan DFD dapat dilihat pada tabel 2.3 Simbol diagram konteks dan DFD.

Tabel 2.3 Simbol diagram konteks dan DFD

Simbol	Nama Simbol	Fungsi
	Entitas Luar	Menggambarkan entitas eksternal yang berhubungan dengan sistem
	Sistem (konteks)/ Proses (DFD)	Menggambarkan proses yang ada pada suatu sistem
	Aliran data/informasi	Menggambarkan aliran data antar proses, data store, dan entitas luar
	Data Store	Menggambarkan kumpulan dokumen-dokumen sejenis yang disimpan

4. Spesifikasi Proses

Spesifikasi Proses merupakan tabel yang berisi keterangan deskripsi dari semua proses yang terdapat di DFD. Logika proses harus dituliskan secara jelas baik menggunakan bahasa deskriptif atau *pseudo code* (tidak boleh campuran).

5. Kamus Data (*Data Dictionary*)

Kamus Data merupakan fakta tentang data dan kebutuhan-kebutuhan informasi dari sistem informasi. Dengan menggunakan *data dictionary*, analis sistem dapat mendefinisikan data yang mengalir dalam sistem dengan lengkap.

2.11 Object Oriented Analysis and Design

Analisis dan desain berorientasi objek adalah cara baru dalam memikirkan satu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Tujuan dari analisis berorientasi objek adalah untuk mengembangkan model yang menggambarkan perangkat lunak komputer karena bekerja untuk

memenuhi seperangkat persyaratan yang ditentukan user. *Tools* yang dapat digunakan pada pendekatan analisis pengembangan sistem secara objek dapat menggunakan UML.

2.12 Unified Modeling Language (UML)

Unified Modeling Language (UML) adalah bahasa spesifikasi standar untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. Unified Modeling Language (UML) adalah himpunan struktur dan teknik untuk pemodelan desain program berorientasi objek (OOP) serta aplikasinya. UML adalah metodologi untuk mengembangkan sistem OOP dan sekelompok perangkat tool untuk mendukung pengembangan sistem tersebut[11]. UML mulai diperkenalkan oleh Object Management Group, sebuah organisasi yang telah mengembangkan model, teknologi, dan standar OOP sejak tahun 1980-an.Sekarang UML sudah mulai banyak digunakan oleh para praktisi OOP.UML merupakan dasar bagi perangkat (tool) desain berorientasi objek dari IBM.UML adalah suatu bahasa yang digunakan untuk menentukan, memvisualisasikan, membangun, dan mendokumentasikan suatu sistem informasi. UML dikembangkan sebagai suatu alat untuk analisis dan desain berorientasi objek oleh Grady Booch, Jim Rumbaugh, dan Ivar Jacobson. Namun demikian UML dapat digunakan untuk memahami dan mendokumentasikan setiap sistem informasi. Penggunaan UML dalam industri terus meningkat. Ini merupakan standar terbuka yang menjadikannya sebagai bahasa pemodelan yang umum dalam industri peranti lunak dan pengembangan sistem.

2.12.1 Sejarah UML

Sampai era tahun 1990 puluhan metodologi pemodelan berorientasi objek telah bermunculan di dunia. Diantaranya adalah: metodologi booch, metodologi coad, metodologi OOSE, metodologi OMT, metodologi shlaer-mellor, metodologi wirfs-brock, dsb. Masa itu terkenal dengan masa perang metodologi (method war) dalam pendesainan berorientasi objek. Masing-masing metodologi membawa notasi sendiri-sendiri, yang mengakibatkan timbul masalah baru apabila kita bekerjasama dengan kelompok/perusahaan lain yang menggunakan metodologi

yang berlainan. Dimulai pada bulan Oktober 1994 Booch, Rumbaugh dan Jacobson, yang merupakan tiga tokoh yang boleh dikata metodologinya banyak digunakan mempelopori usaha untuk penyatuan metodologi pendesainan berorientasi objek. Pada tahun 1995 direlease draft pertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh Object Management Group[11].

2.12.2 Diagram UML

UML menyediakan 10 macam diagram untuk memodelkan aplikasi berorientasi objek [11], yaitu:

1. Use Case Diagram untuk memodelkan proses bisnis.
2. Conceptual Diagram untuk memodelkan konsep-konsep yang ada di dalam aplikasi.
3. Sequence Diagram untuk memodelkan pengiriman pesan (message) antar objects.
4. Collaboration Diagram untuk memodelkan interaksi antar objects.
5. State Diagram untuk memodelkan perilaku objects di dalam sistem.
6. Activity Diagram untuk memodelkan perilaku Use Cases dan objects di dalam system.
7. Class Diagram untuk memodelkan struktur kelas.
8. Object Diagram untuk memodelkan struktur object.
9. Component Diagram untuk memodelkan komponen object.
10. Deployment Diagram untuk memodelkan distribusi aplikasi.

2.12.2.1 Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang diperbuat sistem, dan bukan "bagaimana". Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem. Use case merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng-create sebuah daftar belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. Sebuah use case dapat meng-include fungsionalitas use case lain sebagai bagian dari proses dalam dirinya. Secara umum

diasumsikan bahwa use case yang di-include akan dipanggil setiap kali use case yang meng-include dieksekusi secara normal. Sebuah use case dapat diinclude oleh lebih dari satu use case lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang common. Sebuah use case juga dapat meng-extend use case lain dengan behaviour-nya sendiri. Sementara hubungan generalisasi antar use case menunjukkan bahwa use case yang satu merupakan spesialisasi dari yang lain.

2.12.2.2 Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir. Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi. Activity diagram merupakan state diagram khusus, di mana sebagian besar state adalah action dan sebagian besar transisi di-trigger oleh selesainya state sebelumnya (internal processing). Oleh karena itu activity diagram tidak menggambarkan behaviour internal sebuah sistem (dan interaksi antar subsistem) secara eksak, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas dari level atas secara umum. Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Decision digunakan untuk menggambarkan behaviour pada kondisi tertentu. Untuk mengilustrasikan proses-proses paralel (fork dan join) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. Activity diagram dapat dibagi menjadi beberapa object swimlane untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

2.12.2.3 Sequence Diagram

Sequence diagram menjelaskan secara detil urutan proses yang dilakukan dalam sistem untuk mencapai tujuan dari use case: interaksi yang terjadi antar class, operasi apa saja yang terlibat, urutan antar operasi, dan informasi yang diperlukan oleh masing-masing operasi.

2.12.2.4 Class Diagram

Class diagram merupakan diagram yang selalu ada di permodelan sistem berorientasi objek. Class diagram menunjukkan hubungan antar class dalam sistem yang sedang dibangun dan bagaimana mereka saling berkolaborasi untuk mencapai suatu tujuan.

2.13 Pengujian Kuisioner

Kuesioner adalah sebuah daftar pernyataan yang harus diisi oleh orang yang akan dievaluasi (responden). Metode yang digunakan dalam kuesioner pada penelitian ini adalah skala Likert. Dalam skala likert, responden diminta untuk membaca dengan seksama setiap pernyataan yang disajikan, kemudian ia diminta untuk menilai pernyataan-pernyataan tersebut. Derajat penilaian responden terhadap suatu pernyataan terbagi dalam 5 kategori yang tersusun secara bertingkat, mulai dari Sangat Tidak Setuju (STS), Tidak Setuju (TS), Ragu - Ragu (RG), Setuju (S), dan Sangat Setuju (SS). Atau dapat pula sebaliknya. Pernyataan tiap kuesioner dibuat berdasarkan aspek-aspek yang diteliti. Bobot pemberian skor yang digunakan dapat dilihat pada Tabel 2.2.

Tabel 2.4 Tabel Pengujian Kuisioner

Bobot Pendapatan				
SS	S	RG	ST	STS
5	4	3	2	1

Skor yang telah dihitung pada setiap pernyataan kemudian dikalikan dengan masing-masing bobot tersebut sesuai dengan skenario kuesioner yang telah dibuat. Setelah itu total kan seluruh bobot jawaban tersebut kemudian bagi dengan total responden yang nantinya menjadi nilai rata-rata. Nilai rata-rata inilah yang diambil sebagai acuan sikap dimana jika nilai rata-rata kurang dari skala netral, maka dapat diartikan responden bersikap negatif, jika nilai rata-rata lebih dari skala netral, maka dapat diartikan responden bersikap positif, dan jika nilai rata-rata sama dengan skala netral maka responden bersikap netral terhadap tujuan yang ingin peneliti capai [12]. Untuk lebih jelasnya dapat dilihat pada rumus seperti berikut.

$$X = \frac{\overline{\sum} Total}{n}$$

Dimana,

$\overline{X}$ > 3 bersikap positif

$\overline{X}$ < 3 bersikap negatif

$\overline{X}$ = 3 bersikap netral

Keterangan :

$\overline{X}$ = rata-rata

∑ Total = jumlah seluruh nilai setelah dikalikan dengan bobot

n = total responden