BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

3.1.1 Sejarah Singkat Perusahaan

Klinik Cahaya Qalbu pada awalnya berdiri pada bulan Juni 2006 bernama Klinik Dawaul Hikmah dengan detail sebagai berikut, kepemilikan sarana dan bangunan atas nama Bpk.Iwan Ridwan yang beralamat di Komplek Aria Graha Bandung dan diawal berdirinya hanya dengan unit bisnis apotek yang beroperasi dari jam 08.00 s/d 20.00,dan mulai bulan Agustus 2006 mulai ditambah adanya proyek praktek dokter umum yang buka pagi jam 08.00 s/d 10.00 dan sore jam 17.00 s/d 20.00.Seiring perkembangan, pada bulan Oktober kemudian menjalin suatu kerja sama dengan Bpk.Sariatono dan dr.H.Yani SM dengan kerjasama ini lalu mulai bertambah besar dengan unit bisnis pelayanan Laboratorium Klinik dan ditambah pada bulan Januari 2007 dengan Klinik Rontgen dan EKG. Dengan adanya merger Management antara pemilik (Bpk.Iwan Ridwan) dan pihak ketiga sebagai Pengelola (Bpk.Sariantono dan dr.H.Yani SM) maka terhitung bulan Oktober 2007 terbentuklah CV Dawaul Hikmah sebagai merger tadi sebagai Managerial dan mulai dikembangkannya pelayanan dokter jaga 24 jam termasuk layanan Apotiknya.

Dalam perjalanan dipertengahan tahun 2008 dengan alasan satu dan lain hal maka Unit pelayanan Dokter Jaga 24 jam ditutup dengan waktu yang belum ditentukan kapan akan dibuka lagi dan seiring itu pada waktu yang hamper sama dibubarkan juga merger managerial dengan menutup CV Dawaul Hikmah yang

dilakukan dihadapan Notaris Bpk. Yuniardi, SH. Semua unit bisnis pelayanan baik Apotik, Laboratorium dan Klinik Rontgen masih tetap berjalan seperti biasanya melayani pasien dengan sistem managerial kembali ke semula seperti diawal berdirinya Klinik.

Kemudian karena satu dan lain hal lagi di bulan Januari 2010 terjadi transaksi jual beli kepemilikan sarana bangunan dan semua Unit Bisnis Pelayanan yang ada dalam naungan Klinik Dawaul Hikmah dari Bpk.Iwan Ridwan dan CS kepada dr.H.Kosim Syarief,MKM dan CS. Dalam Kepemilikan dr.H.Kosim Syarief,MKM dan CS Klinik pun berganti nama menjadi Klinik Cahaya Qalbu dalam naungan managerial PT Tri Cahaya Qalbu dengan unit bisnis layanan awal mempertahankan unit Apotek, laboratorium Klinik, EKG dan Klinik Rontgen.

Sesuai perkembangan Klinik Cahaya Qalbu pada pertengahan tahun 2010 kembali mengembangkan dengan mengadakan unit Dokter Jaga 14 jam, dan sampai sekarang awal tahun 2011 sudah dalam tahap pengembangan lahan, bangunan dan perijinan untuk pengembangan ke jenjang Rumah Sakit Ibu dan Anak yang lengkap.

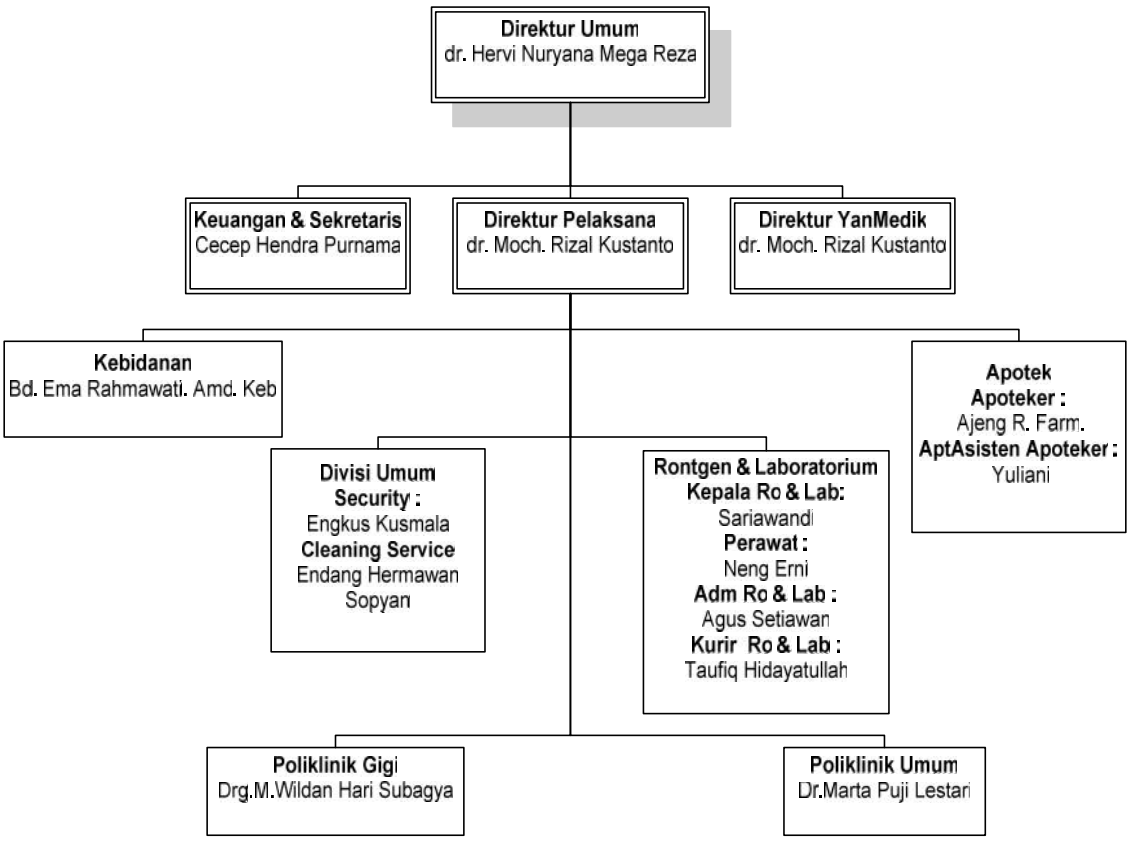
3.1.2 Visi dan Misi Perusahaan

Visi Klinik Utama Cahaya Qalbu : menjadi suatu klinik yang cepat tanggap dan unggul dalam hal pelayanan kepada para pasiennya.

Misi Klinik Utama Cahaya Qalbu :

- Untuk meningkatkan pelayanan di bidang jasa kesehatan khususnya pelayanan Lab, Rontgen, Poliklinik Gigi dan Poliklinik Umum.
- Memberikan kenyamanan pada pasien yang datang ke Klinik Utama Cahaya Qalbu.
- Untuk menjadi sarana solusi yang baik bagi masalah setiap pasien yang datang.

3.1.3 Struktur Organisasi Perusahaan



Gambar 3.1 Struktur Organisasi Cahaya Qalbu

Sumber : Arsip Cahaya Qalbu

3.1.4 Deskripsi Tugas

Tugas pokok setiap Kepala Bagian dalam struktur organisasi BAPELKES adalah :

1. Direktur Utama

   Memimpin Klinik dan mengkoordinasi juga merencanakan masa depan klinik selanjutnya.

2. Direktur Pelaksana

   Mengendalikan dan Mengawasi pelaksanaan kegiatan praktek kesehatan masyarakat di Klinik Utama Cahaya Qalbu.

3. Keuangan & Sekretaris

   Membuat laporan keuangan dan membuat surat – surat atau proposal yang di perlukan perusahaan.

4. Direktur YanMedik

   Menjadi penghubung Dari Direktur pelaksana kepada staf yang lain dan menerima laporan dari staf.

5. Apoteker

   Melayani pasien yang memerlukan obat dari resep dokter, dia dibantu oleh asisten yang sangat diperlukan apabila sedang ada banyak pasien.

6. Petugas lab dan Rontgen

   Memeriksa pasien yang memerlukan pemeriksaan lanjutan yang dianjurkan dokter, petugas juga wajib memberikan laporan kepada Direktur YanMedik.

7. Bidan

   Melayani pasien khususnya ibu – ibu hamil, juga harus memberikan laporan kepada Direktur YanMedik.

8. Poliklinik Umum

   Poliklinik yang menangani atau melayani pasien umum.

9. Poliklinik Gigi

   Poliklinik yang menangani atau melayani pasien spesialis gigi.

10. Divisi Umum

    Bertugas menjaga keamanan dan kenyamanan pasien yang datang.

Tugas utama Klinik Cahaya Qalbu adalah pada bidang kesehatan yaitu membuka suatu usaha yang menyediakan berbagai pelayanan yang optimal kepada pasien yang datang berobat atau sebagainya, dari waktu ke waktu Klinik melakukan beberapa perubahan yang tentunya dilakukan demi terciptanya kenyamanan bagi semua pihak yaitu dengan tambahan beberapa fasilitas yang sekarang melengkapi Klinik ini.

Berikut beberapa fasilitas yang tersedia di Klinik Utama Cahaya Qalbu :

a. Apotek

   Menyediakan berbagai macam obat – obat dan kebutuhan lain yang di butuhkan pasien. Dikelola oleh Apoteker dan asistennya sebagai petugas yang melayani pasien.

b. Laboratorium Klinik dan Lab. Rontgen

   Digunakan untuk pemeriksaan lebih lanjut bagi pasien yang menderita penyakit dalam biasa digunakan untuk pengambilan sampel darah yang dikelola oleh Kepala Lab, Perawat Adm dan Kurir.

c. Poliklinik Bersalin

Merupakan poliklinik yang disediakan untuk melayani pasien ibu hamil untuk memeriksa kandungannya, konsultasi kehamilan maupun untuk melakukan persalinan di poliklinik ini.

d. Poliklinik Umum

Merupakan fasilitas dari Klinik Utama Cahaya Qalbu yang bertujuan melayani melayani pasien umum.

e. Poliklinik Gigi

Merupakan poliklinik baru yang disediakan Klinik Utama Cahaya Kalbu untuk melayani pasien dengan keluhan yang berhubungan dengan kesehatan gigi.

3.2 Metode Penelitian

Dalam suatu penelitian kita diharuskan memiliki suatu metode penelitian yang akan menjadi dasar untuk dijadikan patokan bagi objek yang kita teliti tersebut agar penelitian menjadi lebih terstruktur dan terencana dengan baik metode yang digunakan dalam penelitian dan penyusunan skripsi ini adalah metode penelitian kualitatif. Karena salah satu kelebihan metode ini adalah untuk memahami bagaimana persepsi pengguna mengenai sistem, mengevaluasi sistem serta makna dari sistem bagi pengguna. Memahami perspektif pengguna biasanya sulit dilakukan semata-semata menggunakan pendekatan kuantitatif. Dengan menggunakan pendekatan kualitatif, peneliti dapat menggali penjelasan mengenai perilaku pengguna terhadap sistem, keberhasilan sistem serta kegagalannya dan

sesudah itu dengan penelitian ini terlebih dahulu dilakukan pengambilan beberapa data dan skema yang ada di klinik ini.

3.2.1 Desain Penelitian

Rancangan atau desain penelitian dalam arti sempit dimaknai sebagai suatu proses pengumpulan dan analisis data penelitian. Dalam arti luas rancangan penelitian meliputi proses perencanaan dan pelaksanaan penelitian. Dalam rancangan perencanaan dimulai dengan mengadakan observasi dan evaluasi terhadap penelitian yang biasa dilakukan, sampai pada penetapan kerangka konsep dan hipotesis penelitian yang perlu pembuktian lebih lanjut.

Rancangan pelaksanaan penelitian meliputi proses membuat percobaan ataupun pengamatan serta memilih pengukuran variabel, prosedur dan teknik sampling, instrumen, pengumpulan data, analisis data yang terkumpul, dan pelaporan hasil penelitian.

Berdasarkankan penjelasan tersebut, maka tujuan rancangan penelitian adalah untuk memberikan suatu rencana untuk menjawab pertanyaan-pertanyaan penelitian.

3.2.2 Jenis dan Metode Pengumpulan Data

Metode yang digunakan dalam pengumpulan data adalah mencari dan menggunakan sumber data primer dan sumber data sekunder dalam pengambilan data – data yang dibutuhkan untuk penelitian ini.

3.2.2.1 Sumber Data Primer

1. Wawancara

Merupakan alat *re-cheking* atau pembuktian terhadap informasi atau keterangan yang diperoleh sebelumnya. Tehnik wawancara yang digunakan dalam penelitian ini adalah wawancara mendalam. Wawancara mendalam (in–depth interview) adalah proses memperoleh keterangan untuk tujuan penelitian dengan cara tanya jawab sambil bertatap muka antara pewawancara dengan informan atau orang yang diwawancarai, pada kesempatan ini berwawancara dengan Sekretaris Admin yang bertugas mengolah data – data pasien di Klinik Utama Cahaya Qalbu ini juga dengan perawat klinik.

2. Observasi

Beberapa informasi yang diperoleh dari hasil observasi adalah ruang (tempat), pelaku, kegiatan, objek, perbuatan, kejadian atau peristiwa, waktu, dan perasaan. Alasan peneliti melakukan observasi adalah untuk menyajikan gambaran realistik perilaku atau kejadian, untuk menjawab pertanyaan, untuk membantu mengerti perilaku manusia, dan untuk evaluasi yaitu melakukan pengukuran terhadap aspek tertentu melakukan umpan balik terhadap pengukuran tersebut. Penulis telah melakukan observasi ini pada Klinik Utama Cahaya Qalbu dengan waktu kurang dari 2 minggu.

3.2.2.2 Sumber Data Sekunder (dokumentasi)

Kita tak hanya menggunakan sumber data primer saja sebagai bahan untuk penelitian ini tapi sumber data sekunder juga sangat perlu diambil.

Sejumlah besar fakta dan data tersimpan dalam bahan yang berbentuk dokumentasi. Sebagian besar data yang tersedia adalah berbentuk surat-surat, catatan harian, cenderamata, laporan, artefak, foto, dan sebagainya. Sifat utama data ini tak terbatas pada ruang dan waktu sehingga memberi peluang kepada peneliti untuk mengetahui hal-hal yang pernah terjadi di waktu silam yaitu berupa data – data pasien ataupun arsip – arsip yang bisa dijadikan bahan untuk penelitian.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Metode pendekatan dan pengembangan sistem berisikan beberapa teknik yang digunakan dalam penyusunan skripsi ini, diantaranya diperlukan suatu metode pengembangan sistem, metode pendekatan sistem, dan alat bantu analisis dan perancangan.

3.2.3.1 Metode Pendekatan Sistem

Metode Pendekatan Sistem merupakan perangkat alat atau teknik yang berbentuk kemampuan dalam merumuskan tujuan secara operasional, mengembangkan deskripsi dari sistem yang diterapkan.

Metode pendekatan yang digunakan adalah pendekatan perancangan berbasis objek. Melalui pendekatan objek ini, permasalahan yang komplek

di organisasi dapat dipecahkan dan hasil dari sistem akan mudah untuk dipelihara, fleksibel, lebih memuaskan pemakainya, mempunyai dokumentasi yang baik, tepat waktu, sesuai dengan anggaran biaya pengembangan, dapat meningkatkan produktivitas dan kualitasnya akan lebih baik.

3.2.3.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan untuk merancang sistem informasi ini adalah Metode Pengembangan *Waterfall* dikarenakan metode ini cocok digunakan untuk pendekatan berorientasi objek, adapun uraian metode *waterfall* adalah sebagai berikut :

A. Model Waterfall

Roger S. Pressman (2005 :) Nama model ini sebenarnya adalah "*Linear Sequential Model*". Model ini sering disebut dengan "*classic life cycle*" atau model waterfall. Model ini adalah model yang muncul pertama kali yaitu sekitar tahun 1970 sehingga sering dianggap kuno, tetapi merupakan model yang paling banyak dipakai didalam *Software Engineering* (SE). Model ini melakukan pendekatan secara sistematis dan urut mulai dari level kebutuhan sistem lalu menuju ke tahap *analisis, desain, coding, testing / verification,* dan *maintenance*. Disebut dengan *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan. Sebagai contoh tahap desain harus menunggu selesainya tahap sebelumnya yaitu tahap requirement.

Roger S. Pressman (2005 :) memecah model *Waterfall* menjadi enam tahapan. Berikut ini adalah penjelasan dari tahap-tahap yang dilakukan di dalam model ini menurut Roger S. Pressman :

a. *System / Information Engineering and Modeling*

Permodelan ini diawali dengan mencari kebutuhan dari keseluruhan sistem yang akan diaplikasikan ke dalam bentuk *software*. Hal ini sangat penting, mengingat *software* harus dapat berinteraksi dengan elemen-elemen yang lain seperti *hardware, database*, dsb. Tahap ini sering disebut dengan tahap *Project Definition.*

b. *Software Requirements Analysis*

Proses pencarian kebutuhan diintensifkan dan difokuskan pada *software*. Untuk mengetahui sifat dari program yang akan dibuat, maka para *software engineer* harus mengerti tentang domain informasi dari *software*, misalnya fungsi yang dibutuhkan, *user interface*, dsb. Dari aktivitas pencarian kebutuhan sistem dan *software* tersebut harus didokumentasikan dan ditunjukkan kepada pelanggan.

c. *Design*

Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk rancangan *software* sebelum *coding* dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti dua aktivitas

sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

d. *Coding*

Untuk dapat dimengerti oleh mesin ,maka desain tadi harus diubah bentuknya menjadi bentuk yang dapat dimengerti oleh mesin, yaitu ke dalam bahasa pemrograman melalui proses *coding*. Tahap ini merupakan implementasi dari tahap design yang secara teknis nantinya akan dikerjakan oleh *programmer*.

e. *Testing / Verification*

Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari kesalahan, dan hasilnya harus benar-benar sesuai dengan kebutuhan pemakai sistem yang sudah didefinisikan sebelumnya.

f. *Maintenance*

Pemeliharaan suatu *software* diperlukan, termasuk di dalamnya adalah pengembangan, karena *software* yang dibuat tidak selamanya hanya seperti itu. Ketika dijalankan mungkin saja masih ada kesalahan kecil yang tidak ditemukan sebelumnya, atau ada penambahan fitur-fitur yang belum ada pada *software* tersebut. Pengembangan diperlukan ketika adanya perubahan dari eksternal perusahaan seperti ketika ada pergantian sistem operasi atau perangkat lainnya.

3.2.3.3 Alat Bantu Analisis dan Perancangan

Merupakan representasi grafik yang dapat mempermudah dalam menggambarkan komponen-komponen yang ada di Klinik Utama Cahaya Qalbu ini, proses yang terjadi dan membuat usulan pemecahan masalah secara logika dalam metode berorientasi objek ini penggunaan UML sangat dibutuhkan

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. 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. Berikut beberapa alat bantu yang digunakan :

1) *Use Case* Diagram

Diagram *Use Case* adalah diagram yang menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar dan menjelaskan sistem secara fungsional yang terlihat *user*. Biasanya dibuat pada awal pengembangan. *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, membuat sebuah daftar belanja dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu. *Use case* diagram dapat sangat membantu bila kita sedang menyusun requirement sebuah sistem, mengkomunikasikan rancangan dengan klien, dan merancang test case untuk semua *feature* yang ada pada sistem. Sebuah *use case* dapat memasukan fungsionalitas *use case* lain sebagai bagian dari proses dalam dirinya. Secara umum diasumsikan bahwa *use case* yang disertakan akan dipanggil setiap kali use case yang menyertakan dieksekusi secara normal. Sebuah use case dapat disertakan oleh lebih

dari satu use case lain, sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang common. Sebuah use case juga dapat mengambil use case lain dengan perilakunya sendiri. Sementara hubungan generalisasi antar use case menunjukkan bahwa use case yang satu merupakan spesialisasi dari yang lain.

Use case diagram adalah gambaran graphical dari beberapa atau semua actor, use case, dan interaksi diantara komponen-komponen tersebut yang memperkenalkan suatu sistem yang akan dibangun.Use case diagram menjelaskan manfaat suatu sistem jika dilihat menurut pandangan orang yang berada di luar sistem. Diagram ini menunjukkan fungsionalitas suatu sistem atau kelas dan bagaimana sistem tersebut berinteraksi dengan dunia luar.

Use case diagram dapat digunakan selama proses analisis untuk menangkap *requirements* sistem dan untuk memahami bagaimana sistem seharusnya bekerja. Selama tahap desain, use case diagram berperan untuk menetapkan perilaku (*behavior*) sistem saat diimplementasikan. Dalam sebuah model mungkin terdapat satu atau beberapa use case diagram. Kebutuhan atau *requirements* sistem adalah fungsionalitas apa yang harus disediakan oleh sistem kemudian didokumentasikan pada model use case yang menggambarkan fungsi sistem yang diharapkan (*use case*), dan yang mengelilinginya (*actor*), serta hubungan antara *actor* dengan *use case* (use case diagram) itu sendiri.

2) Skenario *Use Case*

Skenario use case digunakan untuk mempermudah kita dalam menganalisa skenario yang akan digunakan pada fase-fase selanjutnya dengan melakukan penilaian terhadap skenario tersebut, skenario ini mengacu pada uses case diagram yang kita buat sebelumnya.

3) Activity Diagram

Pada bagian ini akan digambarkan dokumentasi alur kerja pada sistem yang sedang berjalan yang bertujuan untuk melihat alur proses sistem yang sedang berjalan.

4) Sequence Diagram

Diagram interaksi yaitu diagram yang menggambarkan suatu interaksi objek-objek dalam usecase.diagram interaksi ini sendiri terdiri dari 2 diagram antara lain diagram sekuen (*sequence diagram*) dan juga diagram kolaborasi (*collaboration diagram*). Diagram sekuen itu menggambarkan perilaku objek yang ada di usecase dengan mendeskripsikan metode-metode yang dimiliki kelas.

Sequence Diagram digunakan untuk menggambarkan interaksi antar objek dalam waktu yang berurutan. Tetapi pada dasarnya sequence Diagram selain digunakan dalam lapisan abstraksi model objek. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara object juga interaksi antara object, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem. Komponen utama sequence diagram terdiri atas objek yang dituliskan dengan kotak segiempat bernama

pesan diwakili oleh garis dengan tanda panah dan waktu yang ditunjukkan dengan proses vertikal.

5) Collaboration Diagram

Collaboration Diagram yaitu diagram yang mengelompokkan pesan pada kumpulan diagram sekuen menjadi sebuah diagram. Dalam diagram tersebut terdapat method yang dijalankan antara objek yang satu dan objek lainnya. Pada diagram kolaborasi ini, objek harus melakukan sinkronisasi pesan dengan serangkaian pesan-pesan lainnya. Diagram ini memberi sebuah cara mengelompokkan potongan-potongan behavior interaksi saat peran-peran dimainkan oleh class yang berbeda. Interaksi penjual dengan sistem pada proses penjualan tanpa penggambaran orientasi waktu.

6) Component Diagram

Component diagram menggambarkan struktur dan hubungan antar komponen peranti lunak, termasuk ketergantungan (*dependency*) diantaranya. Komponen peranti lunak adalah modul berisi code, baik berisi source code maupun binary code, baik library maupun executable, baik yang muncul pada compile time, link time maupun run time. Pada umumnya komponen terbentuk dari bebrapa class dan/atau package, tapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa *interface*, yaitu kumpulan layanan yang disediakan sebuah komponen untuk komponen lain.

7) Deployment Diagram

Deployment diagram yang jika diartikan dalam bahasa Indonesia berarti diagram pendistribusian. Berarti bagaimana caranya kita mempermudah user bila ingin menggunakan sistem yang kita buat, bagian apa dan dimana kita pasang, apakah ada *server* khusus baik *server database* maupun *web server*? Diagram yang satu ini masih masuk dalam kategori statis.

Apa hubungan deployment diagram dini dengan diagram-diagram yang lain? Obyek sejenis dikumpulkan dalam satu class, class-class dalam satu bidang kerja, katakanlah satu transaksi (penjualan, pembelian dan lain-lain) dikelompokkan dalam satu *package* (paket) kemudian *package-package* itu dikelompokkan dalam satu component agar lebih memiliki *dependency* sehingga *component* yang rusak atau harus direvisi tinggal dilepas tanpa mengganggu kerja component lainnya.

3.2.4 Pengujian Software

Perangkat lunak merupakan program komputer yang berfungsi sebagai sarana interaksi antara pengguna dan perangkat keras. Selain itu dapat juga dikatakan sebagai ‘penterjemah’ perintah-perintah yang dijalankan pengguna komputer untuk diteruskan ke atau diproses oleh perangkat keras.

Perangkat Lunak seharusnya memberikan pengguna kebutuhan fungsionalitas dan unjuk kerja yang dapat:

a. Maintanability

Perangkat Lunak yang dihasilkan harus dapat memenuhi perubahan kebutuhan yang sewaktu – waktu diperlukan.

b. Dependability

Perangkat Lunak harus dapat dipercaya dalam artian informasi yang dihasilkan dari *software* yang digunakan ini terbukti kebenarannya

c. Efisiensi

Perangkat Lunak harus efisien dalam penggunaan sumber daya

d. Usability

Perangkat Lunak harus dapat digunakan sesuai dengan yang direncanakan artinya penggunaannya tidak sia – sia dalam artian produk yang dihasilkan berhasil baik dalam menjalankan sistem.

Sebagai tambahan, perangkat lunak yang dihasilkan juga harus mudah digunakan dan menarik dalam hal *interface* sehingga *user* akan lebih optimal dalam pengerjaan sistem informasinya.