BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Dalam penelitian ini, yang menjadi objek penelitian adalah Perusahaan MIKRO CAMACI di Jalan Cibaduyut Block TVRI IV Bandung Telephone : (022) 541104

3.1.1. Sejarah Singkat Perusahaan

Perusahaan MIKRO CAMACI merupakan perusahaan yang memproduksi dan menjual barang berupa sepatu dengan cara grosir dan eceran, baik partai besar maupun partai kecil. Perusahaan MIKRO CAMACI ini didirikan pada tahun 1984 yang berlokasi di jalan Cibaduyut Bandung.

Sebelum berkembangnya dan maju sampai sekarang, pemilik perusahaan ini dulunya adalah seorang pengrajin sepatu, yang membuat prduksi sendiri dan tidak hanya itu penjualannya pun dilakukan berupa membuka lapak dan menawarkan kepada toko-toko di bandung maupun luar bandung.

Pada tahun 1989-1997, Perusahaan MIKRO CAMACI ini memutuskan untuk menambah tenaga kerja karena jumlah pesanan yang lumayan banyak dari setiap toko seapatu. pemilik perusahaan Bu.Hj Eulis dan Bapak H.Utay merasa sudah yakin bahwa produk yang dihasikan layak untuk di jual di toko dan sudah merasa mampu dalam segi biaya apapun.

Akhirnya dari tahun 1997- Sampai sekarang selain memproduksi barang dan menjual barang, perusahaan ini pun menerima pesanan dari berbagai toko-toko yang ingin ikut memasarkan barang, baik itu partai kecil maupun partai besar.

Karena perkembangan itulah perusahan yang bergelut dalam jasa dan produksi ini sekarang memiliki beberapa mitra usaha yang mendukung dalam segi manajemen baik itu pemasaran maupun penjualan.

3.1.2. Visi dan Misi Perusahaan

Visi dari Perusahaan Perusahaan MIKRO CAMACI sendiri adalah :

"Memberikan pelayanan yang mampu memuaskan para pelanggan". Selain itu juga untuk menciptakan lapangan pekerjaan bagi masyarakat setempat, agar turut dalam membangun perekonomian daerah dan negara.

Sedangkan Misi dari Perusahaan MIKRO CAMACI yaitu :

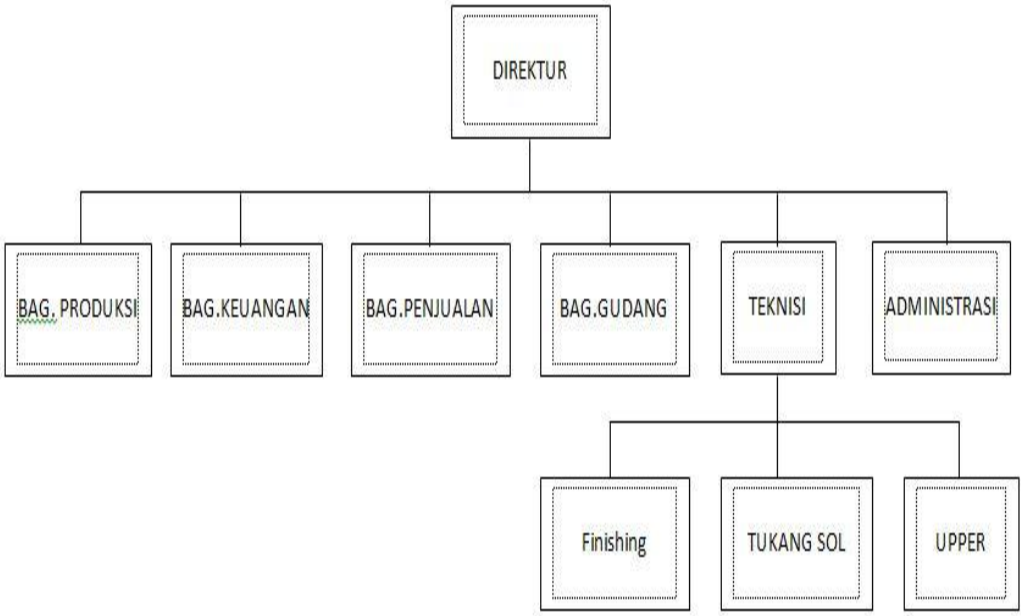
Memasarkan dan memperluas pangsa pasar dan area penjualan di daerah pemasaranya serta menjaga kualitas produk untuk membangun kepercayaan konsumen.

3.1.3. Struktur Organisasi Perusahaan

Struktur organisasi merupakan bagian dari manajemen perusahaan. Dengan adanya struktur organisasi yang baik, maka akan memudahkan pimpinan dan para karyawan untuk mengetahui batas-batas tugas, wewenang dan tanggung jawab, serta hubungan kerja pada tiap personilnya.

Pada pelaksanaan penelitian ini penulis melakukan penelitian pada perusahaan guna untuk mengetahui apa saja proses yang kurang dalam perusahan ini.

Adapun struktur organisasi Perusahaan MIKRO CAMACI yang sedang berjalan saat ini dapat dilihat pada gambar 1.1.



Gambar 3.1 Struktur Organisasi Perusahaan MIKRO CAMACI

[Perusahaan MIKRO CAMACI]

3.1.4. Deskripsi Tugas

Deskripsi kerja adalah pernyataan-pernyataan tertulis yang meliputi tugas-tugas, wewenang, tanggung jawab dan hubungan kerja harus dilaksanakan dengan baik dan benar dalam satu organisasi.

Berikut ini deskripsi kerja dari masing-masing fungsi yang bertugas di Perusahaan MIKRO CAMACI :

Direktur :

1. Mengatur dan mengendalikan jalannya perusahaan.
2. Memperhatikan prestasi perusahaan secara keseluruhan
3. Bertanggung Jawab atas kelancarannya jalan perusahaan.
4. Bertanggung jawa atas kondisi pabrik dan karyawan.

Administrasi :

1. Membantu pimpinan dibidang adminitrasi.
2. Melaksanakan kegiatan bisnis pimpinan.
3. Mengatur kegiatan bisnis produksi.
4. Bertanggung jawab atas dokumen atau arsip perusahaan.
5. Bertanggung jawab atas kelancaran kegiatan bisnis pimpinan.

Kepala Bagian Keuangan :

1. Mengatur pengeluaran biaya, dari setiap proses transaksi penjualan maupun pembelian.

Kepala Bagian Penjualan :

1. Mengatur dan mengorganisir setiap proses transaksi penjualan yang terjadi.

Kepala Bagian Produksi :

1. Menentukan pengawasan dibagian produksi.

2. Memberikan keterangan mengenai cara kerja kepada karyawan jika diperlukan.
3. Bertanggung jawab atas kelancaran produksi.
4. Bertanggugn jawab atas karyawan jika dirasakan kurang tahu dalam melakukan pekerjannya.

Bagian Gudang :

1. Mencatat semua penerimaan dan pengeluaran barang baku dan aksesoris pada gudang.
2. Memeriksa jenis dan kualitas bahan baku yang diterim dari penjual.
3. Bertanggung jawab atas bahan baku yang ada digudang
4. Bertanggung jawab atas kekurangan bahan baku.

Teknisi :

1. Melakukan tugas produksi yang telah ditentukan.
2. Bertanggung jawab atas hasil produk yang dihasilkan.

3.2 Metode Penelitian

Metode ialah suatu kerangka kerja untuk melakukan suatu tindakan, atau suatu kerangka berfikir menyusun gagasan, yang beraturan, berarah dan berkonteks, yang terkait (*relevant*) dengan maksud dan tujuan.

Penelitian adalah cara ilmiah untuk mendapatkan tujuan dan kegunaan tertentu. Seorang peneliti harus menentukan metode yang akan dipergunakan, dan

ditentukannya metode penelitian, maka akan memandu seseorang peneliti mengenai urutan-urutan bagaimana penelitian dilakukan.

Metode penelitian merupakan suatu mekanisme, teknik atau tatacara bagaimana suatu penelitian dilaksanakan. Metode penelitian dianggap sebagai seperangkat pendekatan menyeluruh untuk mengumpulkan data dan menganalisis masalah-masalah tertentu mencakup teknik dan alat.

3.2.1 Desain Penelitian

Desain penelitian yang digunakan dalam penelitian ini bersifat deskriptif. Penelitian deskriptif adalah penelitian yang diarahkan untuk memberikan gejala-gejala, fakta-fakta, atau kejadian-kejadian secara sistematis dan akurat mengenai sifat-sifat populasi atau daerah tertentu. Dalam hal ini berupa sistem informasi yang sedang berjalan serta entitas-entitas yang terkait dalam perusahaan atau instansi yang menjadi objek penelitian. Metode deskriptif dilakukan dengan cara mengumpulkan informasi aktual secara rinci mengenai masalah yang akan diteliti.

Berdasarkan uraian di atas penulis beranggapan bahwa metode deskriptif sesuai digunakan dalam penelitian tentang Sistem Informasi Persediaan Barang. karena metode deskriptif dapat mendeskripsikan data yang aktual berupa gambaran dari masalah yang diteliti dan dapat mengetahui fenomena-fenomena yang terjadi pada sistem informasi yang berjalan di perusahaan atau instansi.

3.2.2 Jenis dan Metode Pengumpulan Data

Agar data yang diperoleh sesuai dengan apa yang ada pada perusahaan maka penulis menggunakan metode penelitian berupa terjun langsung kelapangan untuk dijadikan objek penelitian.

3.2.2.1 Sumber Data Primer

Data primer yaitu data yang diperoleh dari responden secara langsung yang dikumpulkan melalui *survey* lapangan dengan menggunakan teknik pengumpulan data tertentu yang dibuat khusus untuk itu.

Teknik pengumpulan data dalam rangka pembentukan informasi mengenai objek penelitian ini, dilakukan dengan cara:

1. Observasi

Observasi adalah kegiatan mempelajari suatu gejala dan peristiwa melalui upaya mengamati dan mencatat data atau informasi secara sistematis. Observasi dapat dibedakan menjadi dua jenis, yaitu:

1. Observasi langsung, yaitu observasi yang dilakukan dimana observer (dalam hal ini adalah peneliti) berada bersama objek atau pada lingkungan objek yang ditelitinya.
2. Observasi tidak langsung, yaitu observasi yang dilakukan tidak pada saat berlangsungnya suatu peristiwa yang akan diteliti, misalnya dilakukan melalui *audio, video* atau *visual* yang berkaitan dengan objek yang diteliti.

   Obeservasi yang dilaksanakan dalam penelitian ini adalah observasi langsung karena peneliti mengadakan pengamatan langsung kelapangan untuk memperoleh data atau informasi yang akurat mengenai sistem informasi yang dijalankan di perusahaan.

Hasil dari kegiatan obeservasi ini adalah peneliti dapat mengetahui proses-proses yang terjadi pada Sistem Informasi Persediaan Barang pada Perusahaan MIKRO CAMACI.

2. Wawancara

Wawancara *(interview)* merupakan pertemuan dua orang untuk bertukar informasi dan ide melalui tanya jawab sehingga dapat dikonstruksikan makna dalam suatu topik tertentu. Pertanyaan-pertanyaan dalam wawancara ditetapkan terlebih dahulu dengan tujuan untuk menjawab fokus penelitian atau tidak keluar dari maksud dan tujuan penelitian. Dalam penelitian ini wawancara dilakukan terhadap kepala divisi pada Perusahaan MIKRO CAMACI atau yang dianggap mempunyai wewenang untuk memberikan data dan informasi yang diperlukan.

Wawancara ini berisikan pertanyaan-pertanyaan tentang Sistem Informasi Persediaan Barang yang berjalan pada Perusahaan MIKRO CAMACI. Sehingga diidentifikasi beberapa permasalahan mengenai Sistem Informasi Persediaan Barang tersebut.

3. Studi Pustaka

Studi pustaka yaitu teknik pengumpulan data dan informasi melalui buku-buku dan sumber tertulis lainnya yang berhubungan dengan bidang penelitian, sehingga hasilnya dapat dijadikan sebagai dasar untuk penelitian yang dilaksanakan.Studi pustaka yang digunakan pada penelitian ini adalah karya-karya ilmiah dan buku-buku yang membahas tentang sistem informasi perkuliahan atau kegiatan belajar mengajar di instansi pendidikan, salah satunya universitas.

3.2.2.2 Sumber Data Sekunder

Data sekunder adalah data yang digunakan untuk mendukung data primer dan merupakan jenis data yang sudah diolah terlebih dahulu oleh pihak pertama (pihak perusahaan atau instansi yang bersangkutan). Dengan data sekunder yang

ada, peneliti menggunakan teknik dokumentasi untuk mengolah data tersebut. Dokumentasi adalah teknik pengumpulan data dengan cara mengumpulkan dokumen-dokumen yang berhubungan dengan objek penelitian.

Peneliti menganalisa dokumen-dokumen yang terlibat pada sistem informasi Pengajian di Perusahaan MIKRO CAMACI. Hasil dari analisa dokumen-dokumen tersebut akan digunakan dalam menentukan kebutuhan sistem pada perancangan sistem aplikasi Penggajian sebagai sarana penunjang Sistem Informasi Persediaan Barang di Perusahaan MIKRO CAMACI.

3.2.3 Metode Pendekatan dan Pengembangan Sistem

Dalam penelitian ini akan dilaksanakan kegiatan analisis sistem informasi yang sedang berjalan dan perancangan sistem informasi yang diusulkan terhadap perusahaan atau instansi, yang nantinya akan dibuatkan atau menghasilkan sebuah produk sistem informasi berupa sistem aplikasi. Untuk itu diperlukan penggunaan metode pendekatan dan pengembangan sistem, serta alat bantu analisis dan perancangan.

Adapun pendekatan sistem yang dilakukan penulis adalah pendekatan berorientasi objek (*object oriented*), serta alat bantu analisis dan perancangan menggunakan diagram-diagram UML (*Unifield Modeling Language*).

3.2.3.1 Metode Pendekatan Sistem

1) Object Oriented Analysis and Design (OOAD)

Pendekatan baru untuk pengembangan sistem, sering disebut sebagai pendekatan ketiga setelah pendekatan yang berorientasi data dan berorientasi

proses. *OOAD* adalah metode pengembangan sistem yang lebih menekankan pada objek dibandingkan dengan data atau proses. Ada beberapa ciri khas dari pendekatan ini yaitu *object, Inheritance* dan *object class*.

2) Unified Modelling language (UML)

Unified Modelling Language (UML) adalah sebuah "bahasa" yang telah menjadi standar dalam industri untuk menentukan, visualisasi, merancang dan mendokumentasikan artifact dari sistem *software*, untuk memodelkan bisnis dan sistem non *software* lainnya. UML merupakan suatu kumpulan teknik terbaik yang telah terbukti sukses dalam memodelkan sistem yang besar dan kompleks.

Metode pendekatan sistem ini menggunakan beberapa metode dalam perancangannya diantaranya :

1. ***Use case.***
2. ***Activity Diagram.***
3. ***Sequence Diagram.***
4. ***Component Diagram.***
5. ***Deployment Diagram.***
6. ***Class Diagram***

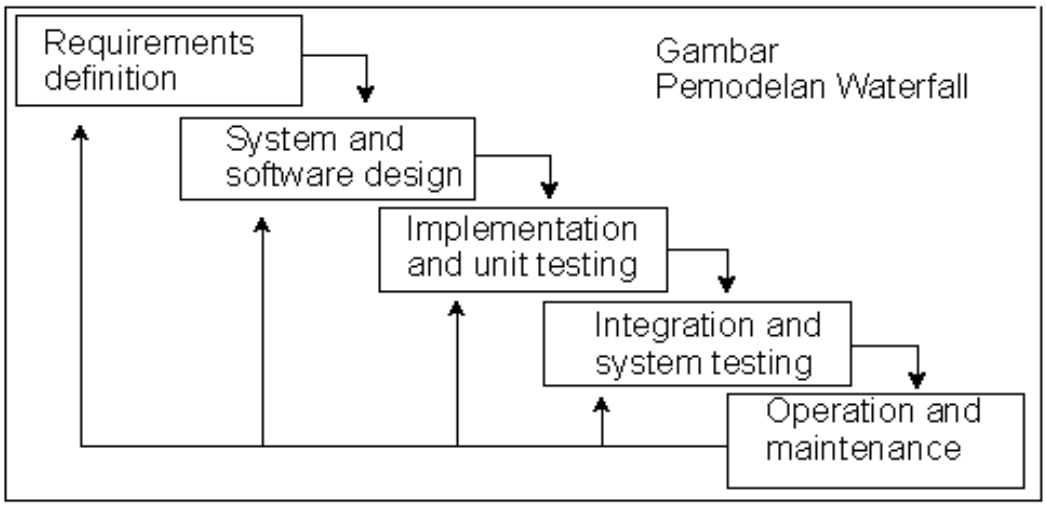
3.2.3.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang digunakan untuk pengembangan sistem adalah *Waterfall* yaitu model pengembangan perangkat lunak secara sekuensial dimana pengembangan sistem dilihat seperti air terjun mengalir ke bawah melalui

beberapa tahapan (Wartika S.Kom, MT, Manajemen Proyek Sistem Informasi. P105).

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.*

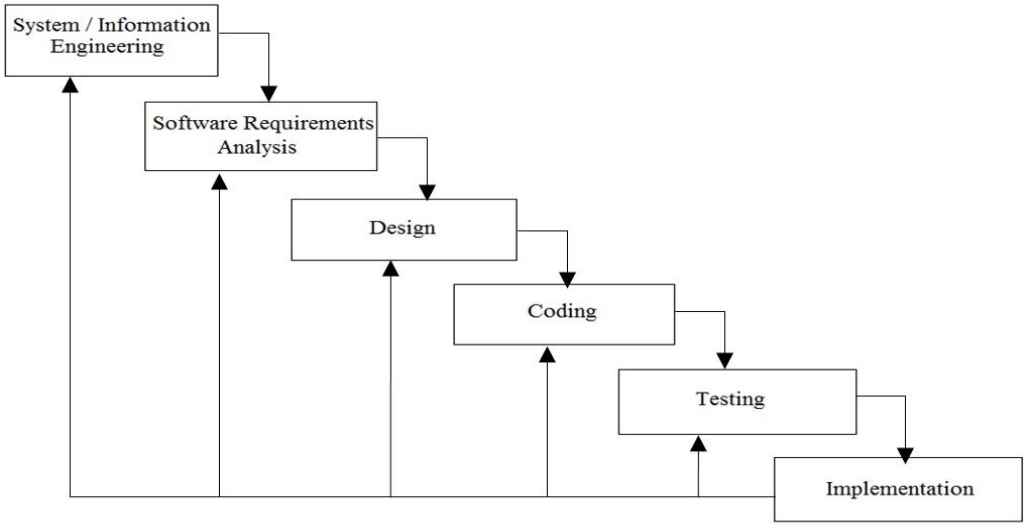
Secara umum tahapan pada model *waterfall* dapat dilihat pada gambar berikut :



Gambar 3.2. *Waterfall (SDLC.*

Sumber : *Roger S. Pressman. Sistem Engineering.2004,P31-32*.

Gambar di atas adalah tahapan umum dari model proses ini. Akan tetapi Roger S. Pressman memecah model ini menjadi 6 tahapan meskipun secara garis besar sama dengan tahapan-tahapan model *waterfall* pada umumnya. Berikut adalah penjelasan dari tahap-tahap yang dilakukan di dalam model ini menurut Pressman:



Gambar 3.3. *Tahapan Waterfall (SDLC) yang digunakan*

1. *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 *Project Definition*.

2. *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 2 aktivitas tersebut (pencarian kebutuhan sistem dan *software*) harus didokumentasikan dan ditunjukkan kepada pelanggan.

3. *Design*. Proses ini digunakan untuk mengubah kebutuhan-kebutuhan diatas menjadi representasi ke dalam bentuk "blueprint" *software* sebelum coding dimulai. Desain harus dapat mengimplementasikan kebutuhan yang telah disebutkan pada tahap sebelumnya. Seperti 2 aktivitas sebelumnya, maka proses ini juga harus didokumentasikan sebagai konfigurasi dari *software*.

4. *Coding*. Untuk dapat dimengerti oleh mesin, dalam hal ini adalah komputer, 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 dikerjakan oleh programmer.

5. *Testing / Verification*. Sesuatu yang dibuat haruslah diujicobakan. Demikian juga dengan *software*. Semua fungsi-fungsi *software* harus diujicobakan, agar *software* bebas dari error, dan hasilnya harus benar-benar sesuai dengan kebutuhan yang sudah didefinisikan sebelumnya.

6. *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 errors 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

1) Actor

Actor menggambarkan segala pengguna *software* aplikasi (user). *Actor* memberikan suatu gambaran jelas tentang apa yang harus dikerjakan *software* aplikasi.

2) Use case

Use case menjelaskan urutan kegiatan yang dilakukan *Actor* dan sistem untuk mencapai suatu tujuan tertentu. Walaupun menjelaskan kegiatan, namun *use case* hanya menjelaskan apa yang dilakukan oleh *Actor* dan sistem bukan bagaimana *Actor* dan sistem melakukan kegiatan tersebut.

a) *Use-case Konkret* adalah *use case* yang dibuat langsung karena keperluan *Actor*. *Actor* dapat melihat dan berinisiatif terhadapnya

b) *Use-case Abstrak* adalah *use case* yang tidak pernah berdiri sendiri. *Use case* abstrak senantiasa termasuk didalam *(include)*, diperluas dari *(extend)* atau memperumum *(generalize) use case* lainnya.

3) *Class*

Class merupakan pembentuk utama dari sistem berorientasi obyek, karena *class* menunjukkan kumpulan obyek yang memiliki atribut dan operasi yang sama. *Class* digunakan untuk mengimplementasikan interface. *Class* digunakan untuk mengabstraksikan elemen-elemen dari sistem yang sedang dibangun. *Class* bisa merepresentasikan baik perangkat lunak maupun perangkat keras, baik konsep maupun benda nyata.

4) Activity Diagram

Menggambarkan aliran fungsionalitas dalam suatu sistem. Dapat digunakan dalam pemodelan bisnis untuk menunjukkan business *workflow*. Atau juga digunakan dalam analisa kebutuhan untuk menggambarkan aliran kejadian melalui suatu *use case*. Mendefinisikan dimana *workflow* dimulai, dimana berhentinya, aktifitas apa yang terjadi selama *workflow*, bagaimana urutan kejadian aktifitas. Suatu aktifitas adalah suatu pekerjaan yang dilaksanakan selama *workflow*.

5) Sequence Diagram

Sequence diagram menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya) berupa *message* yang digambarkan terhadap waktu. *Sequence Diagram* terdiri atar dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait).

6) Collaboration Diagram

Collaboration Diagram juga menggambarkan interaksi antar objek seperti *Sequence Diagram*, tetapi lebih menekankanp ada peran masing-masing objek dan bukan pada waktu penyampaian *message*. Setiap *message* memiliki *sequence number*, di mana *message* dari level tertinggi memiliki nomor 1. *Messages* dari *level* yang sama memiliki prefiks yang sama.

7) Component Diagram

Component Diagram menggambarkan struktur danhubungan antar komponen piranti lunak, termasuk ketergantungan (*dependency*) di antaranya.

Komponen piranti 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.*

8) Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yang bersifat fisikal.

3.2.4. Pengujian *Software*(Perangkat Lunak)

Pengujian adalah suatu proses pengeksekusian program yang bertujuan untuk menemukan kesalahan. Pengujian sebaiknya menemukan kesalahan yang tidak disengaja dan pengujian dinyatakan sukses jika berhasil memperbaiki kesalahan tersebut. Selain itu, pengujian juga bertujuan untuk menunjukkan kesesuaian fungsi-fungsi perangkat lunak dengan spesifikasinya. Sumber : *(Retno.H. 2003.Perancangan Pengujian Perangkat Lunak Berorientasi Obyek: Berbasis Status (state-based testing), Jurnal Universitas Paramadina, Vol. 2 No. 2, Januari: P175-190).*

Black box Testing memfokuskan pada keperluan fungsional dari *software*. Karena itu ujicoba *black box* memungkinkan pengembang *software* untuk membuat himpunan kondisi input yang akan melatih seluruh syarat-syarat fungsional suatu program. Ujicoba *blackbox* bukan merupakan alternatif dari ujicoba *whitebox,* tetapi merupakan pendekatan yang melengkapi untuk

menemukan kesalahan lainnya, selain menggunakan metode *whitebox*. Sumber : (*Ayuliana. 2009. Testing dan Implementasi. Maret).*

Ujicoba *blackbox* berusaha untuk menemukan kesalahan dalam beberapa kategori, diantaranya :

1) Fungsi-fungsi yang salah atau hilang.
2) Kesalahan interface.
3) Kesalahan dalam struktur data atau akses database eksternal.
4) Kesalahan performa, inisialisasi dan terminasi. Sumber : *(Ayuliana. 2009. Testing dan Implementasi. Maret).*