BAB III

OBJEK DAN METODE PENELITIAN

3.1 Objek Penelitian

Dalam melakukan penelitian ini, penulis mengambil suatu objek penelitian yang berupa sebuah perusahaan yang bergerak di bidang penyedia jasa transportasi. Perusahaan tersebut bernama Baraya Travel. Dan untuk lebih jelasnya mengenai objek penelitian tersebut, maka penulis akan membahas mengenai sejarah, visi dan misi, struktur organisasi serta deskripsi tugas dari perusahaan tersebut.

3.1.1 Sejarah Singkat Perusahaan

Barya Travel merupakan perusahaan yang bergerak dalam bidang jasa transportasi yang saat ini melayani rute perjalanan Jakarta - Bandung.

Baraya travel berdiri pada bulan oktober 2005, berawal dari kebutuhan masyarakat akan transportasi yang murah, berdasarkan kebutuhan owner Baraya adanya peluang bisnis dalam jasa transportasi.

Baraya Travel didirikan oleh Bambang Utama sebagai owner perusahaan, beliau mengajukan proposal bisnis ke sembilan saudaranya, dengan dasar kebutuhan jasa transportasi yang besar dan didorong rasa sosial untuk membantu masyarakat, maka berdirilah sebuah perusahaan penyedia jasa transportasi yang bernama Baraya

Travel, dengan terbatasnya modal yang tersedia sebesar Rp. 50.000.000,- dan kendaraan sebanyak 2 mobil elf, Baraya Travel mulai beroperasi melayani calon penumpang dengan rute Jakarta-Bandung, Loket penjualan yang tersedia pada saat itu hanya tiga 3 loket yaitu Lombok (Bandung)dan Sarinah, Melawai (Jakarta)

Seiring dengan waktu yang berjalan animo masyarakat untuk menggunakan jasa transportasi Baraya Travel semakin besar, karena harga tiket yang ditawarkan sesuai dengan daya beli masyarakat, bahkan lebih murah dibandingkan perusahaan jasa transportasi lainnya, perlahan namun pasti perusahaan ini pun mulai berkembang dan mulai mendirikan loket-loket baru untuk agar dapat menjangkau masyarakat dengan mudah

Hingga saat ini baraya travel telah memiliki 50 armada dan 14 loket/outlet penjualan, setiap harinya rata-rata melayani 3600 penumpang dengan rute Jakarta-Bandung atau sebaliknya. Pihak manajemen terus berusaha meningkatakan pelayanan kepada calon konsumen untuk dapat memuaskan konsumen.

3.1.2 Visi Dan Misi Perusahaan

VISI yang menjadi petunjuk bagi kami untuk terus berkembang yaitu :

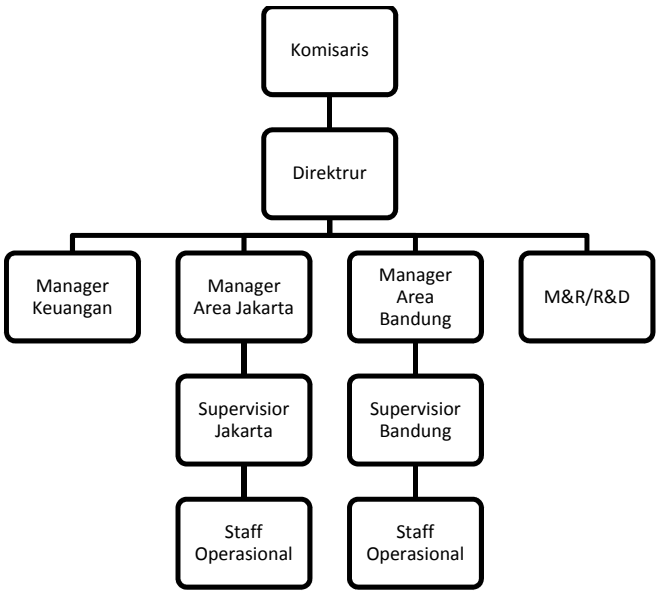
a) Menjadi pioneer atau perintis serta simbol bagi citra terbaik dalam bisnis transportasi .

b) Menjadi sarana penghubung bagi pengguna jasa transportasi.

MISI nya, yaitu :

a) Memberikan pelayanan maksimal untuk profesionalitas kerja yang tinggi.
b) Memberikan rasa aman dalam perjalanan dengan menyediakan kendaraan dalam kondisi prima yang dikendarai oleh pengemudi terbaik, terlatih serta berpengalaman.
c) Selalu memiliki komitmen tinggi terhadap ketepatan waktu dan pelayanan maksimal serta ekslusif.

3.1.3 Struktur Organisasi Perusahaan



Gambar 3.1 Struktur Organisasi

3.1.4 Deskripsi Tugas

Berdasarkan struktur organisasi yang ada di Baraya Travel, dimana setiap level manajemen nya memiliki tanggung jawab dan deskripsi pekerjaan tersebut di antara nya adalah sebagai berikut :

1) **Dewan Komisaris**

Seperti dewan komisaris lain nya, disini mempunyai tugas :

a) Mengambil keputusan tertinggi dalam perusahaan dan juga
b) Sebagai pemodal bagi perusahaan

2) **Direktur Utama**

Bisa dikatakan sebagai pemimpin perusahaan primajasa yang memiliki tanggung jawab:

a) Memiliki tanggung jawaba teratas, untuk menjalankan dan memantau perkembangan perusahaan secara keseluruhan.
b) Dan juga menentukan arah kebijakan keputusan perkembangan perusahaan

3) **Manager Keuangan**

Level ini dalam perusahaan memiliki tanggung jawab sebagai berikut :

a) Ikut serta menetukan kebijakan operasional keuangan perusahaan
b) Menyusun dan memperbaharui laporan keuangan dari semua loket perjalanan.
c) Memantau arus laporan keuangan untuk disampaikan lagi ke level manajemen tertinggi

4) **Manager Area**

Pada level ini manager area memiliki tanggung jawab untuk :

a) Bertanggung Jawab dengan loket area perjalanan

b) Menyusun strategi penjualan untuk target penjualan maksimal

c) Membuat report kepada direktur utama, mengenai perkembangan dan target perusahaan

d) Menetapkan jadwal driver, ruter perjalanan dan tujuan, jadwal keberangkatan disetiap loket.

5) Maintenance & Repair / Research & Development

a) Mengevaluasi semua permasalahan yang berkaitan dengan segi operasional kendaraan seperti penyediaan dan perbaikan

b) Menentukan suplier-suplier untuk kebutuhan bengkel, teknisi mesin, oli, pelumas.

c) Melakukan observasi untuk pengembangan perusahaan dalam segi teknis dan operasional sperti pelayanan tiket dan kebutuhan akan teknologi.

6) Staff Operasional

A. Bagian Tiket

Tanggung jawab dan pekerjaan utamanya adalah :

a) Menangani pemesanan dan penjualan tiket.

b) Membuat transaksi penjualan tiket

B. Bagian Operasi Lapangan

Bagian ini di pimpin oleh satu orang yang memiliki tugas, mengatur serta memantau segala aktivitas perusahaan.

Dibagian ini terdapat juga beberapa karyawan level lower yaitu :

i. **Petugas Keberangkatan**

Memiliki tugas dan tanggung jawab : menangani pemberangkatan penumpang, berupa bagasi dan pengaturan penumpang.

ii. **Sopir**

Tanggung jawabnya : mengendarai kendaraan yang sudah di tetapkan

3.2 Metode Penelitian

Metode yang di gunakan penulis dalam pengembangan sistem adalah metode prototipe. Metode prototipe merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat sebuah program dengan cepat dan bertahap sehingga dapat segera di evaluasi oleh pemakai (user). Berdasarkan pengertian metode prototipe diatas penulis mempunyai beberapa alasan mengapa penulis menggunakan metode pengembangan sistem prototipe yaitu karena penulis akan lebih mudah dalam merancang sistem yang di inginkan perusahaan dan dapat di terima oleh user sebagai pengguna sistem, hal lain nya adalah penulis menginginkan perancangan sistem yang telah dihasilkan kemudian di presentasikan kepada user dan user di berikan kesempatan untuk memberikan masukan ataupun komentar sehingga sistem informasi yang dihasilkan benar-benar sesuai dengan keinginan dan kebutuhan user terutama bagi perusahaan sendiri. Perubahan dan presentasi prototipe ini dapat dilakukan berkali-kali sampai di

capai kesepakatan dari bentuk sistem informasi yang akan di implementasikan.

3.2.1. Desain Penelitian

Desain penelitian yang peneliti pakai disini adalah penelitian deskriptif yaitu suatu metode dalam meneliti status sekelompok manusia, suatu objek, suatu set kondisi, suatu sistem pemikiran ataupun suatu kelas peristiwa masa sekarang.

3.2.2. Jenis Data Metode Pengumpulan Data

Jenis data yang akan digunakan dalam penelitian ini adalah data primer dan data sekunder. Data primer dari penelitian ini di peroleh melalui pengamatan langsung kepada objek penelitian atau respondennya. Data primer ini seperti profile perusahaan, data administratif (meliputi: proses bisnis) dan informasi lainnya yang berkaitan dengan rutinitas bisnis perusahaan. Sedangkan data sekunder ini bisa kita dapatkan melalui hasil dokumentasi, internet, biro informasi dan lain-lain yang tentunya berkaitan erat dengan studi kasus penelitian di Baraya Travel

Mengarah kepada sasaran dan tujuan yang ingin di capai dalam penelitian ini, penulis melakukan pengumpulan data sebagai bahan laporan dengan menggunakan metode yang diberi nama metode deskriptif, yaitu metode yang pada tahap pertama dilakukan dengan cara mengumpulkan data dan bahan yang diperlukan terlebih dahulu, dan pada tahap berikutnya penulis mengolah dan membahas sampai

pada suatu laporan untuk melampirkan semua kegiatan yang dikerjakan selama dilakukan nya penelitian di Baraya Travel

3.2.2.1 Sumber Data Primer

Data primer merupakan pengumpulan data yang dilakukan secara langsung terhadap objek yang sedang di teliti, cara yang digunakan untuk mengumpulkan data primer adalah sebagai berikut :

a. Observasi (Pengamatan)

Observasi yang di maksud dalam penelitian ini adalah melakukan penelitian secara langsung terhadap subbidang kerja / unit / divisi yang ada dengan tujuan untuk lebih memahami dan mengetahui langkah-langkah apa saja yang diambil dalam menyelesaikan permasalahan dan serta hambatan-hambatan yang ditemukan dalam aktivitas bisnis di Baraya Travel

b. Wawancara

Wawancara dilakukan Bapak Hendra sebagai kepala Bagian operasional di Baraya Travel untuk mendapat gambaran tentang proses bisnis yang ada di Baraya Travel, kendala-kendala yang dihadapi dalam proses pemesanan tiket, di harapkan dengan wawancara penulis dapat mengetahui lebih jelas tentang kebutuhan penelitian

3.2.2.2 Sumber Data Sekunder (Dokumentasi)

Aktivitas dokumentasi disini adalah pencarian bahan-bahan atau buku bacaan, karya ilmiah dan sumber-sumber bacaan lainnya seperti internet, biro informasi. Data dokumentasi ini juga meliputi company profile Baraya Travel, data administrative yang di butuhkan sebagai bahan kajian oleh peneliti yang nantinya akan di gunakan sebagai masalah penelitian.

3.2.3. Metode Pengembangan Dan Pendekatan Sistem

3.2.3.1 Metode Pendekatan Sistem

Metode pendekatan sistem yang di gunakan adalah pendekatan dengan Object Oriented yang menggnakan AOO (Analisis Object Oriented) dan DOO yang di visualisasikan dengan UML dan diantara nya adalah sebagai berikut : Use case , Activity diagram, Class Diagram, Sequence diagram, Class Diagram, Collaboration diagram, Component diagram ,dan Deployment diagram.

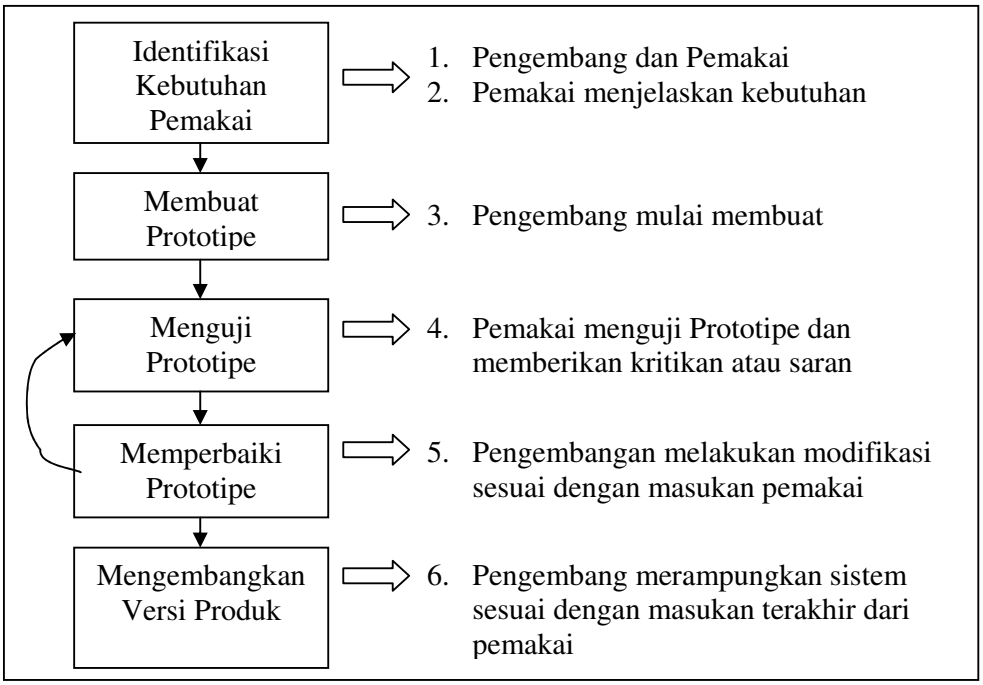
3.2.3.2 Metode Pengembangan Sistem

Metode pengembangan sistem yang di gunakan adalah dengan menggunakan metode pendekatan prototype.

Prototyping merupakan salah satu metode pengembangan perangkat lunak yang banyak di gunakan.

Metode prototype ini di rancang dapat menerima perubahan- perubahan dalam rangka untuk penyempurnaan dari prototype yang sudah ada sehingga akhirnya dapat menghasilkan sistem informasi yang dapat diterima dan perubahan – perubahan yang terjadi di anggap dapat merupakan sebagian dari proses pengembangan itu sendiri.

Pola pendekatan secara prototipe ini di gambarkan sebagai berikut:



Gambar 3.2 Skema Pengambangan sistem secara prototipe

Berikut adalah langkah – langkah yang di lakukan penulis dalam proses perancangan sebuah sistem yang

menggunakan mekanisme pengembangan sistem dengan prototipe yang sesuai dengan gambar 3.2 diatas, berikut penjelasan detail langkah – langkahnya :

1. Pada tahapan ini, dilakuakan pengidentifikasian kebutuhan *user,* supaya penulis bisa merancang sistem yang akan di bangun sesuai dengan yg di harapkan *user.* Sebelum pada tahap perancangan, penulis akan memulai pada tahap awal terlebih dahulu yaiut penulis akan menganalisis sistemdengan cara melakukan pengumpulan data yaitu dengan *field research* (Metode penelitian) atau observasi, dan interview (wawancara) dan literature yaitu dengan dokumentasi terhadap kebutuhan yang di inginkan pemakai, baik dalam model interface, teknik procedural maupun dalam teknologi yang akan digunakan.
2. Tahapan kedua yaitu memperbaiki prototipe, penulis akan membuat prototipe sistem tersebut untuk memperlihatkan kepada pemakai model sistem yang akan dirancang.
3. Tahapan ketiga yaitu pengujian prototipe, penulis akan melakukan uji coba sistem yang telah di rancang untuk memastikan bahwa sistem tersebut dapat digunakan dengan baik dan benar, sesuai kebutuhan pemakai.
4. Tahapan keempat yaitu memperbaiki prototipe, penulis akan menentukan apakah sistem tersebut dapat diterima

oleh pemakai, atau harus dilakukan beberapa perbaikan ataupun bahkan di bangun ulang kembali dari awal. Kemudian setelah perbaikan sistem selesai di kerjakan, penulis akan kembali lagi pada tahap ketiga yaitu melakukan pengujian prototipe kembali.

5. Tahap terakhir adalah tahap kelima yaitu mengembangkan versi produksi , penulis akan merampungkan sesuai masukan terakhir dari pemakai dan memberikan gambaran bagaimana pengguna sistem tersebut kepada pemakai setelah sistem tersebut di setujui.

3.2.3.3 Alat Bantu Analisis Dan Perancangan.

1) Use Case Diagram

Use case diagram menggambarkan fungsionalitas yang di harapkan dari sebuah sistem. Yang ditekankan adalah ”apa” yang diperbuat sistem, dan bukan ”bagaimana”. Sebuah usecase mempresentasikan sebuah interaksi antara aktor dengan sistem. Usecase merupakan sebuah pekerjaan tertentu, misalnya login ke sistem, meng create sebuah daftar belanja, dan sebagainya. Seorang atau 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 requirment sebuah sistem, mengkomunikasikan rancangan dengan klien, merancang test case untuk semua feature yang ada pada sistem. Sebuah usecase dapat meng include fungsionalitas usecase lain sebagai bagian dari proses dalam dirinya. Secara umum di asumsikan bahwa use case yang di include oleh lebih dari satu usecase lain,sehingga duplikasi fungsionalitas dapat dihindari dengan cara menarik keluar fungsionalitas yang common. Sebuah use case juga dapat meng extend use case lain dengan behaviournya sendiri. Sementara hubungan generalisasi antar use case menunjukan bahwa use case yang merupakan spesialisasi dari yang lain.

2) Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan disain berorientasi objek. Class menggambarkan keadaan (atribut/properti) suatu sistem , sekaligus menawarkan layanan untuk manipulasi keadaan tersebut (metoda/fungsi). Class diagram menggambarkan struktur dan deskripsi class, Package dan objek serta hubungan satu sama lain seperti contaiment pewarisan asosiasi dan lain – lain.

Class memiliki 3 area pokok :

1. Nama (stereotype)
2. Atribut
3. Metoda

Atribut dan metoda dapat memiliki sifat sebagai berikut :

a) Private, tidak dapat dipanggil dari luar class yang bersangkutan.
b) Protected,hanya dapat dipanggil oleh class yang bersangkutan dengan anak- anak yang mewarisi nya.
c) Public, dapat di panggil oleh siapa saja.

Class dapat merupakan implementasi dari sebuah interface, yaitu class abstrack yang hanya memiliki metoda. Inteface tidak dapat langsung di instansikan, tetapi harus diimplementasikan dahulu menjadi sebuah class. Dengan demikian interface mendukung resolusi metoda pada saat runtime.

Sesuai perkembangan class model, class dapat dikelompokkan menjadi package. Kita juga dapat membuat diagram yang terdiri atas package.

Hubungan antar class

1. Assosiasi, yaitu hubungan statis antar class. Umumnya menggambarkan class yang memiliki atribut berupa class yang memiliki atribut berupa class lain atau class yang

harus menunjukan arah query class lain. Panah navigability menunjukan arah query antar class.

2. Pewarisan, yaitu hubungan hirarkis antar class. Class dapat diturunkan dari class lain dan mewarisi semua atribut dan metoda class asalnya dan menambahkan fungsionalitas baru, sehingga ia disebut anak dari class yang di warisi nya. Kebalikan dari pewarisan adalah generalisasi.
3. Agregasi, yaitu hubungan yang menyatakan bagian ("terdiri atas..")
4. Hubungan dinamis, yaitu rangkaian pesan (message) yang di passing dari satu class kepada class lain.hubungan dinamis dapat di gambarkan dengan menggunakan sequence diagram yang akan dijelaskan kemudian.

3) Statechart Diagram

Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya) suatu objek sistem sebagai akibat dar stimuli yang diterima. Pada Umumnya state diagram menggambarkan class tertentu (satu class dapat memiliki lebih dari satu state diagram). Dalam UML, state di gambarkan berbentuk segi empat dengan sudut membulat dan meiliki nama sesuai kondisi nya pada saat itu. Transisi antar state pada umumnya memiliki kondisi guard yang merupakan syarat terjadinya

transisi yang bersangkuta, dituliskan dalam kurung siku. Action yang dilakukan sebagai akibat dari event tertentu dituliskan dengan diawali garis miring. Titik awal dan akhir di gambarkan berbentuk lingkaran bewarna penuh dan bewarna setengah.

4) Activity Diagram

Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang di rancang, 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. Dimana 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 usecase atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara usecase menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas. Sama seperti state, standar

UML menggunakan segiempat dengan sudut membulat untuk menggambarkan 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.

5) Sequence Diagram

Sequnce diagram menggambarkan interaksi antara objek di dalam dan di sekitar sistem (termasuk pengguna, display, dan sebagainya.)berpa message yang di gambarkan erhadap waktu. Sequence diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait.). Sequence diagram biasa nya digunaka untuk menggambarkan skenario atau rangkaia langkah-langkah yang di lakukan sebagi respon dari sebuah event untuk menghasilkan output tertentu. Diawali dengan men trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal output apa yang di hasilkan.

Masing-masing objek, termasuk aktor, memiliki lifeline vertikal. Message di gambarkan sebagai garis

berpanah dari satu objek ke objek lainnya. Pada fase desain berikutnya, message akan dipetakan menjadi operasi/metoda class. Activation br menunjukan lama nya eksekusi sebuah proses, biasanya diawali dengan diterima nya sebuah message. Untuk objek-objek yang memiliki sifat khusus, standar UML mendefenisikan icon khusus untuk object boundary, controller dan persistent entity.

6) Collaboration Diagram

Collaboration diagram juga menggambarkan interaksi antar objek seperti sequence diagram, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian message. Setiap message memiliki sequence number, dimana message dari level tertinggi memiliki nomor 1.

Message dari level yang sama memiliki prefiks yang sama.

7) Component Diagram

Component diagram menggambarkan struktur dari hubungan antra komponen piranti lunak, termasuk ketergantungan (dependency) diantaranya. 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 runtime. Umumnya komponen terbentuk beberapa class

dan atau package, tetapi dapat juga dari komponen-komponen yang lebih kecil. Komponen dapat juga berupa interface, yaitu kumpulan yang disediakan sebuah komponen untuk komponen lain.

8) Deployment Diagram

Deployment/physical diagram menggambarkan detail bagaimana komponen di deploy dalam infrastruktur sistem, dimana 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. Sebuah node adalah server, workstation, atau piranti keras lain yang digunakan untuk mendeploy komponen dalam lingkungan sebenarnya. Hubungan antar node (misalnya TCP/IP) dan requirment dapat juga di defenisikan dalam diagram ini.

3.2.4. Pengujian Software

Defenisi pengujian PL (perangkat lunak) adalah :

1. Menurut Myers (1979)

Proses menjalankan program dengan maksud menemukan kesalahan

2. Menurut IEEE (1990)

a. Proses sistem operasi atau komponen menurut kondisi tertentu, pengamatan atau pencatatan hasil dan mengevaluasi beberapa aspek siste, atau komponen.

b. Proses analisis *item* Perangkat Lunak mendeteksi perbedaan antara kondisi yang ada dengan yang diiginkan dan mengevaluasi fitur item Perangkat Lunak.

3.1.3.1 Tujuan Pengujian Sistem

Pengujian sistem bertujuan untuk :

1. Tujuan langsung :
   a) Identifikasi dan menemukan beberapa kesalahan yang mungkin ada dalam Perangkat Lunak yang di uji setelah Perangkat Lunak dibetulkan, diidentifikasi lagi kesalahan dan dites ulang untuk menjamin kualitas level penerimaan.
   b) Membentuk tes yang efisien dam efektif dengan anggaaran dan jadwal yang terbatas.
2. Tujuan tidak langsung

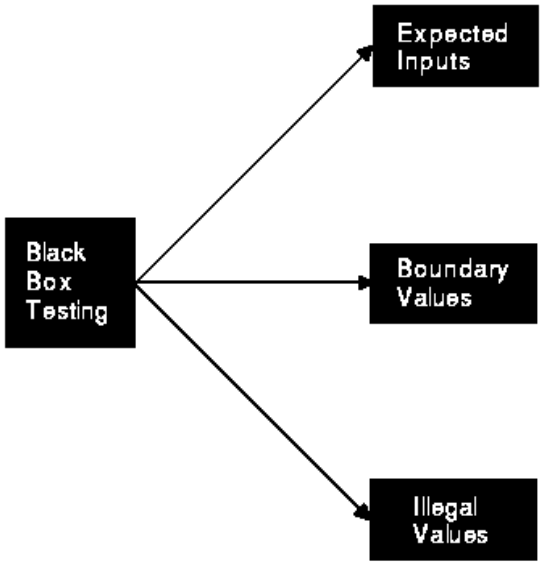
   Tujuan tidak langsung dari pengujian Perangkat Lunak yaitu mengumpulkan daftar kesalahan untuk digunakan dalam daftarn pencegahan kesalahan (tindakan corrective dan preventive).

3.1.3.2 Konsep Pengujian

Pengelompokan pengujian berdasarkan konsep pengujian dibedakan menjadi :

1. *Black box (functionally) testing* : mengidentifikasikan kesalahan yang berhubungan dengan kesalahan fungsionalitas Perangkat Lunak yang tampak dalam kesalahan output.

Black box testing adalah pengujian yang mengabaikan mekanisme internal sistem atau komponen dan fokus semata-mata pada output yang dihasilkan yang merespon *input* yang di pilih dan kondisi eksekusi. *Black box* juga di defenisikan sebagai pengujian yang memegang perhitungan mekanisme internal sistem atau komponen dengan kebutuhan fungsional tertentu (IEEE).



Gambar 3.3 Proses pengujian *black box*

Sumber: http://h71000.www7.hp.com/doc/73final/4518/zk-4071.gif

2. *White box* (*structural*) *testing/glass box testing*: memeriksa kalkulasi internal path untuk mengidentifikasika kesalahan. *White box testing* adalah pengujian yang memegang perhitungam mekanisme internal sistem atau komponen. *White box testing* memilik 4 kategori, yaitu:

1) Data *processing* and *calculations correctness test* : melakukan pengecekan data *processing* untuk setiap kasus tes. Pendekatan yang digunakan :
   a) *Path coverage* : rencana tes yang mencakup semua kemungkina *path*, dimana *coverage* di ukur dengan beberapa % path di cover.
   b) *Line coverage* : Rencana Tes yang mencakup semua baris kode program, dimana *coverage* diukur dengan beberapa *% line dicover*.
2) *Software qualification path*
3) *Maintanability test*
4) *Reusability test.*