BAB III

OBJEK DAN METODE PENELITIAN

3.1. Objek Penelitian

Penelitian dilakukan di Direktorat Logistik ITB Bandung yang beralamat di jl. Ganesha no.10 Bandung.

3.1.1. Sejarah Singkat Direktorat Logistik ITB Bandung

Sejarah Direktorat Logistik dimulai dengan pendirian Unit Urusan Logistik (U2L) ITB pada bulan Mei 2003 yang ditugaskan sebagai unit implementasi dalam pengadaan barang dan jasa yang dibutuhkan oleh unit akademik maupun unit pendukung di ITB yang dengan menggunakan Dana Masyarakat (DM). U2L beroperasi dengan berdasarkan pada suatu peraturan modifikasi pengadaan barang pemerintah (Keppres). Pada saat yang bersamaan terdapat unit lain, Proyek Pengembangan (PP) ITB, atau lebih dikenal dengan P2T, yang juga telah beroperasi dalam pengadaan barang dan jasa khusus untuk dana APBN (DIP).

U2L ITB dan Proyek Pengembangan ITB, pada tanggal 3 September 2004, bergabung menjadi Pusat Logistik (Puslog) ITB yang memiliki fungsi sebagai suatu unit pengadaan terpusat di ITB (SK Rektor ITB No. 173/SK/K01/OT/2004). Namun demikian, Puslog baru dapat memulai operasinya, sebagai sebuah unit, pada Januari 2005 untuk melakukan pengadan semua barang dan jasa dengan menggunakan dana baik DM maupun APBN, dan beroperasi berdasarkan pada Keputusan Presiden tentang Pengadaan Barang dan Jasa Pemerintah (Keppres No.

80/2003). Setelah beroperasi selama satu tahun, Puslog pada bulan Januari 2006 berubah nama kembali menjadi UPT Logistik (SK Rektor ITB No. 290/SK/K01/KP/2005).

UPT Logistik melayani lebih dari 65 unit kerja berdasarkan pada Rencana Implementasi Triwulanan dari Rencana Kegiatan dan Anggaran (RKA). Karena kapasitas yang dimiliki dalam mengelola proses pengadaan barang dan jasa, serta mempertimbangkan efisiensi dan efektivitasnya, maka UPT Logistik memberikan wewenang kepada unit kerja untuk melakukan pengadaan sendiri barang dan jasa yang termasuk dalam kategori tertentu berdasarkan jenis dan nilai pengadaan. UPT Logistik ITB mendukung unit kerja dalam beberapa kegiatan pengadaan seperti bantuan teknis dalam pendefinisian kebutuhan, proses pemilihan *supplier*, implementasi pekerjaan, monitoring dan pengawasan pekerjaan, penerimaan, dan distribusi barang dan jasa. Fungsi UPT Logistik dalam pengadaan barang dan jasa memegang peran strategis untuk memastikan semua kegiatan akademik dan operasional dapat berjalan dengan lancar secara efektif dan efesien. Untuk itu, posisi UPT Logistik sangat penting dalam mendukung tercapainya visi ITB, terutama sebagai institusi pendidikan yang baik dalam operasionalnya.

Dengan keluarnya SK Rektor 147/SK/K01/OT/2010, maka UPT Logistik berubah nama menjadi DIrektorat Logsitik yang memiliki fungsi yang tidak berubah, namun memiliki peran tambahan strategis untuk melakukan perencanaan pengadaan barang/jasa, monitoring dan evaluasi proses pengadaan (SK Rektor No. 309/K01.05/SK/2010 tentang Pengelolaan Anggaran Pengadaan Barang yang

Dilaksanakan Melalui Direktorat Logistik) , dan pengelolaan persediaan di lingkungan ITB yang lebih terintegrasi serta komprehensif.

3.1.2. Visi dan Misi Direktorat Logistik ITB Bandung

Visi dan misi sekolah Direktorat Logistik ITB Bandung adalah sebagai berikut:

Visi

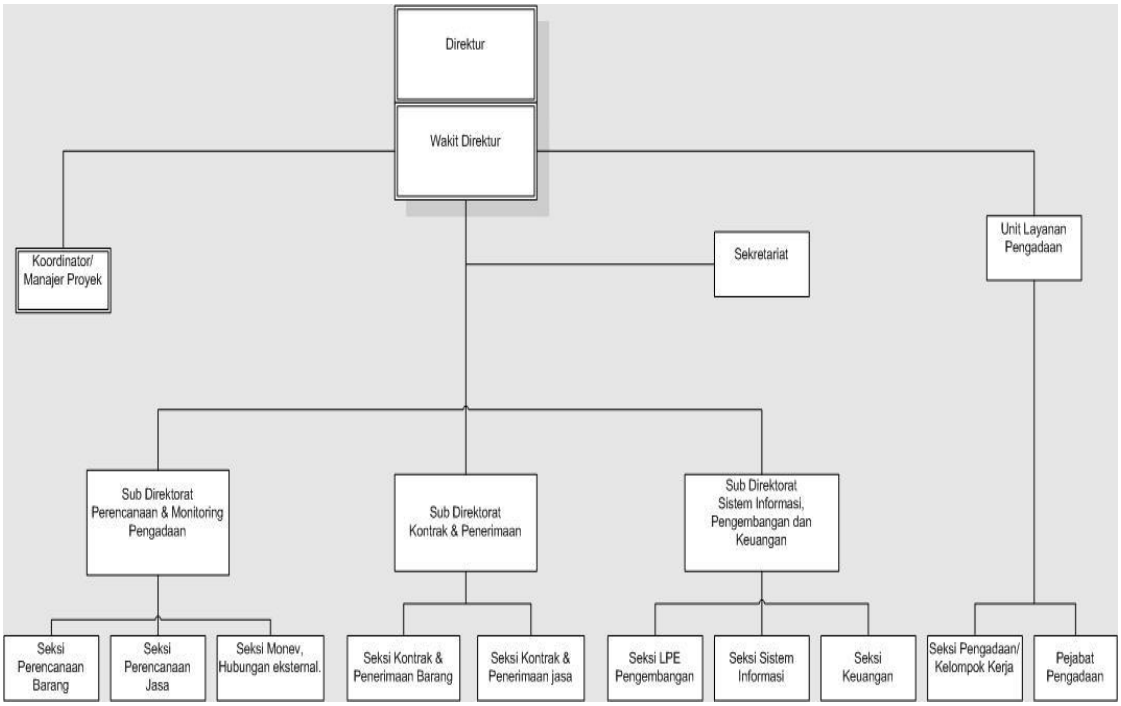
"Direktorat Logistik ITB sebagai lembaga pelayanan proses logistik institusi pendidikan tinggi yang profesional dan terpandang untuk mendukung visi ITB sebagai universitas riset terpandang di Asia dan berperan aktif dalam mengembangkan sistem logistik pendidikan tinggi secara nasional"

Misi

"Melaksanakan proses bisnis dalam sistem logistik ITB secara akuntabel, professional, dengan tingkat pelayanan yang tinggi kepada seluruh unit kerja di ITB

"Mendukung terciptanya atmosfir ITB yang bersih, tertib, harmonis, kondusif dan nyaman dalam lingkup pelaksanaan kegiatan logistik di ITB"

3.1.3. Struktur Organisasi Direktorat Logistik ITB Bandung



Gambar 3.1 Struktur Organisasi Direktorat Logistik ITB Bandung

Sumber : Direktorat Logistik ITB Bandung

3.1.4. Job Descriptions

1. Direktur

Bertanggung jawab atas proses verifikasi dokumen perencanaan yang d ajukan oleh unit kerja, penyusunan dokumen perencanaan pengadaan, pemilihan penyedia barang dan jasa, penerimaan dan pemeriksaan hasil kerja penyedia barang dan jasa, penyimpanan dan distribusi barang, pengawasan dan penerimaan jasa serta pengajuan permohonan pembayaran yang timbul atas kegiatan pengadaan barang dan jasa, dimana seluruh proses tersebut harus dilakukan mengacu pada ketentuan barang dan jasa yang berlaku.

2. **Wakil Direktur**

   Membatu kinerja Direktur Logistik ITB khususnya dalam pengendalian operasional direktorat logistik itb guna tercapainya tujuan pengadaan barang dan jasa di ITB.

3. **Sub Direktorat Perencanaan dan Monitoring Pengadaan**

   mengkoordinasikan kegiatan perencanaan pengadaan barang dan jasa, melakukan proses monitoring dan evaluasi pengadaan barang dan jasa serta menangani jalinan hubungan eksternal yang dapat mendukung tercapainya peningkatan kinerja.

4. **Sub Direktorat Kontrak dan Penerimaan**

   Mengkoordinasikan kegiatan penyusunan kontrak pengadaan barang dan jasa dan mengendalikan pelaksanaan kontrak pengadaan barang dan jasa serta melakukan pemeriksaan serta penerimaan hasil pengadaan barang dan jasa.

5. **Sub Direktorat Sistem Informasi, Pengembangan dan Keuangan**

   Bertanggung jawab dalam pengelolaan sistem informasi logistik(SIRLOG) ITB, pelayanan pengadaan elektronik (LPE) ITB, melakukan fungsi pengembangan proses bisnis dirlog itb serta melaksanakaan verifikasi dokumen permohonan pembayaran dan mengajukannya melalui mekanisme yang berlaku.

6. Unit Pelayanan Pengadaan

Menjalankan fungsi manajerial atas penugasan pejabat pengadaan serta kelompok kerja unit layanan pengadaan serta bertanggung jawab atas program kerja dan oprasional unit layanan pengadaan.

7. Seksi pPerencanaan Pengadaan Barang

Melakukan verifikasi atas pengajuan kebuthan barang dari unit kerja atau *user* serta menyusun dokumen perencanaan pengadaan barang.

8. Seksi Perencanaan Pengadaan Jasa

Melakukan verifikasi atas pengajuan kebuthan jasa dari unit kerja atau *user* serta menyusun dokumen perencanaan pengadaan jasa.

9. Seksi Monitoring dan Evaluasi dan Hubungan Eksternal

Melakukan proses monitoring dan evaluasi atas oprasional yang dijalankan oleh Direktorat Logistik ITB serta menangan hubungan eksternal antara Direktorat Logistik ITB dengan pihak lain.

10. Seksi Kontrak dan Penerimaan Barang

Menyusun kontrak pengadaan barang, mengendalikan pelaksanaan kontrak pengadaan barang serta melakukan pemerikasaan dan penerimaan barang yang disediakan oleh penyedia barang.

11. Seksi Kontrak dan Penerimaan Jasa

Menyusun kontrak pengadaan jasa, mengendalikan pelaksanaan kontrak pengadaan jasa serta melakukan pemerikasaan dan penerimaan jasa yang disediakan oleh penyedia jasa.

12. Seksi ELPE dan pengembangan

Mengelola bidding room, menyediakan layanan pengadaan elektronik bagi kalangan internal ITB maupun institusi lain dan dunia usaha dan melaksanakan fungsi pengembangan Direktorat Logistik ITB guna mendukunng proses bisnis Direktorat Logistik ITB.

13. Seksi Sistem Informasi

Mengelola SIRLOG ITB Direktorak Logistik guna mendukung proses bisnis Direktorat Logistik ITB.

14. Seksi Keuangan

Melakukan verifikasi atas dokumen permohonan pembayaran yang diajukan penyedian barang dan jasa serta menyampaikannya kepada Direktorat Keuangan ITB untuk dilakukan proses pembayaranan.

15. Seksi Pengadaan /Kelompok Kerja

Melaksanakan proses pemilihan peyedia barang dan jasa.

16. Pejabat Pengadaan

Melaksanakan proses pemilihan penyedia barang dan jasa dengan metode pengadaan dan penunjukan langsung.

17. Sekertariat

Mendukung terlaksanakan proses bisnis Direktorat Logistik ITB dengan melaksanakan operasional kantor Direktorat Logistik ITB.

18. Manajer Proyek

Membantu pejabat pembuat komitmen dalam proses pelaksanaan dan pengendalian kontrak pengadaan barang dan jasa.

3.2. Metode Penelitian

3.2.1. Desain Penelitian

Dalam melakukan penelitian perlua adanya metode penelitian, sehingga penelitian berjalan dengan baik. Oleh karena itu penulis memilih metode penelitian tindakan (*Action Research*).

Penelitian tindakan (*Action Research*) ialah penelitian yang bertujuan untuk mengembangkan keterampilan-keterampilan baru, cara pendekatan baru, atau produk pengetahuan yang baru dan untuk memecahkan masalah dengan penerapan langsung di dunia aktual atau lapangan (Narimawati, 2010).

3.2.2. Jenis dan Metode Pengumpulan Data

Data yang diperlukan dalam penelitian ini terdiri dari data primer dan data sekunder. Data primer yang diperlukan dalam penelitian ini dikumpulkan dengan melakukan penelitian lapangan yaitu datang langsung ke perusahaan yang bersangkutan. Data sekunder dikumpulkan melalui penelitian kepustakaan.

3.2.2.1.Sumber Data Primer

Sumber data primer yaitu sumber data yang berasal dari perorangan, kelompok, panel atau sumber terselubung. Dalam memperoleh data primer penulis melakukan wawancara dan observasi di tempat penelitian.

1. Observasi

   Proses untuk mendapatkan data dengan mengamati objek yang akan diteliti secara langsung di tempat penelitian yaitu di Direktorat Logistik ITB Bandung. Yang meliputi bagaimana proses pengumuman pengadaan barang dan jasa yang berjalan di Direktorat Logistik ITB Bandung.

2. Wawancara

   Wawancara proses untuk mendapatkan data dengan melakukan dialog langsung kepada orang yang dianggap bisa memberikan informasi yang dibutuhkan oleh penulis yaitu kepada salah satu staf Direktorat Logistik ITB Bandung.

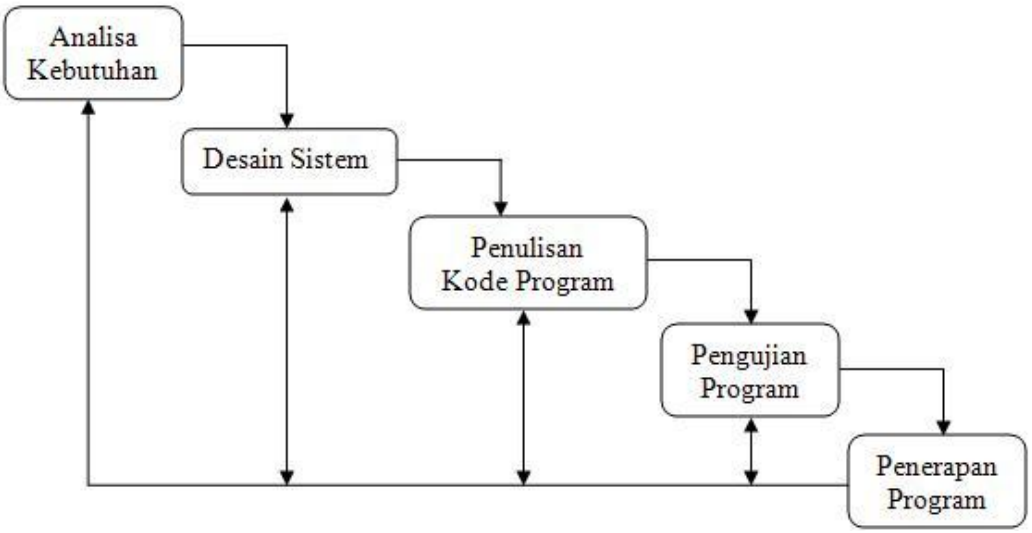
3.2.2.2.Sumber Data Sekunder

Data sekunder yang diambil dari berbagai sumber melalui Studi Kepustakaan, yaitu dengan mempelajari berbagai sumber bacaan / dokumen yang berkaitan erat dengan masalah penelitian, baik berupa artikel dan literatur ilmiah.

3.3. Metode Pendekatan Sistem

Metode pendekatan sistem yang di gunakan adalah pendekatan dengan *Object Oriented* yang menggunakan OOA(*Object Oriented Analysis*) di visualisasikan dengan UML dan di antara nya adalah sebagai berikut : *Use Case Diagram, Activity Diagram, Sequence Diagram, Class Diagram, Component Diagram dan Deployment Diagram*.

3.4. Metode Pengembangan Sistem



Gambar 3.2 Metode *Waterfall*

Sumber: Roger S. Pressman (2002:37)

Pada perkembangan sistem informasi ini penulis menggunakan metode *waterfall* sebagai alternatif pengembangan sistem informasi dengan tujuan mendapatkan kualitas dari sistem yang dihasilkan akan baik.

Metode *waterfall* merupakan metode yang sering digunakan oleh penganalisa sistem pada umumnya. Inti dari metode *waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan atau secara linear. Jadi jika langkah satu belum dikerjakan maka tidak akan bisa melakukan pengerjaan langkah 2, 3 dan seterusnya. Secara otomatis tahapan ke-3 akan bisa dilakukan jika tahap ke-1 dan ke-2 sudah dilakukan.

Secara garis besar metode *waterfall* mempunyai langkah-langkah sebagai berikut : Analisa, *Design*, *Code* dan *Testing*, Penerapan.

1. Analisa

Langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa malakukan sebuah penelitian, wawancara atau *study literatur*. Seorang sistem analis akan menggali informasi sebanyak-banyaknya dari *user* sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan menghasilkan dokumen *user requirment* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan aplikasi penerimaan barang. Dokumen ini lah yang akan menjadi acuan sistem analis untuk menterjemahkan ke dalam bahasa pemprograman.

2. Design

Proses desain akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat *coding*. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail (algoritma) prosedural. Tahapan ini akan menghasilkan dokumen yang disebut *software requirment*. Dokumen inilah yang akan digunakan *programmer* untuk melakukan aktivitas pembuatan sistemnya.

3. Coding & Testing

Coding merupakan penerjemahan *design* dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh *programmer* yang akan menterjemahkan transaksi yang diminta oleh *user*. Tahapan ini lah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan *testing* terhadap

sistem yang telah dibuat tadi. Tujuan *testing* adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

4. Penerapan

Tahapan ini bisa dikatakan *final* dalam pembuatan sebuah sistem atau rancangan. Setelah melakukan analisa, design dan pengkodean maka sistem atau rancangan yang sudah jadi akan digunakan oleh *user*.

3.5. Alat Bantu Analisis dan Perancangan

1. Use case Diagram

Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem. Yang ditekankan adalah "apa" yang di perbuat sistem, dan bukan "bagaimana". Sebuah *use case* merepresentasikan sebuah interaksi antara aktor dengan sistem. *Use case* merupakan sebuah pekerjaan tertentu misalnya *login* 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.

Use case diagram dapat sangat membantu bila kita sedang menyusun requirement sebuah sistem, mengkomunikasikan rancangan klien, dan merancang *test case* untuk semua fitur yang ada pada sistem. 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 di panggil setiap kali *use case* yang meng *included* eksekusi secara normal.

Sebuah *use case* dapat di *include* oleh lebih dari satu *use case* lain, sehingga duplikasi fungsionalitas dapat dihindari dengan menarik keluar fungsionalitas yang *common*. Sebuah *use case* juga dapat meng *extend use case* lain dengan behaviornya sendiri. Sementara hubungan generalisasi antar *use case* menunjukkan bahwa *use case* yang satu merupakan spesialisasi dari yang lain.

2. Activity Diagram

Activity Diagram atau yang di sebut Diagram Aktivitas digunakan untuk mendokumentasikan alur kerja pada sebuah system, yang di mulai dari pandangan *business level* hingga ke operational level, adapaun penjelasanya antara lain :

1. *Activity diagram* merupakan cara lain memodelkan aliran kejadian.
2. Kita gunakan *activity diagram* untuk menggambarkan *workflow*/aliran kerja proses bisnis.
3. *Activity diagram* menggambarkan berbagai alir aktivitas dalam sistem yang sedang di rancang bagaimana masing – masing alir berawal, *decision* yang mungkin terjadi, dan bagaimana mereka berakhir.
4. *Activity diagram* juga dapat menggambarkan proses *parallel* yang mungkin terjadi pada beberapa eksekusi.

3. Class Diagram

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

Class menggambarkan keadaan (*atribut*/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metoda/fungsi). *Class* diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta belanja, dan sebagainya. Seorang/sebuah aktor adalah sebuah entitas manusia atau mesin yang berinteraksi dengan sistem untuk melakukan pekerjaan-pekerjaan tertentu.

Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek. *Class* menggambarkan keadaan (*atribut*/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut (metode/fungsi). *Class* diagram menggambarkan struktur dan deskripsi *class*, *package* dan objek beserta hubungan antar *class*.

4. 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 atas dimensi vertikal (waktu) dan dimensi *horizontal* (objek-objek yang terkait). *Sequence* diagram biasa digunakan untuk menggambarkan skenario atau

rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah *event* untuk menghasilkan output tertentu. Diawali dari apa yang men- *trigger* aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan *output* apa yang dihasilkan. Masing-masing objek, termasuk aktor, memiliki *lifeline vertikal*.

Message digambarkan sebagai garis berpanah dari satu objek ke objek lainnya. Pada *fase* desain berikutnya, *message* akan dipetakan menjadi operasi/metoda dari class. *Activation bar* menunjukkan lamanya eksekusi sebuah proses, biasanya diawali dengan diterimanya sebuah message.Untuk objek-objek yang memiliki sifat khusus, standar UML mendefinisikan *icon* khusus untuk objek *boundary*, *controller* dan *persistent entity*.

5. *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, di mana *message* dari level tertinggi memiliki nomor 1. *Messages* dari level yang sama memiliki prefiks yang sama.

6. Component Diagram

Component diagram menggambarkan struktur dan hubungan 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*. Umumnya komponen terbentuk dari beberapa *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/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 . Sebuah node adalah server, workstation, atau piranti keras lain yang digunakan untuk men-*deploy* komponen dalam lingkungan sebenarnya. Hubungan antar *node* (misalnya TCP/IP) dan *requirement* dapat juga didefinisikan dalam diagram ini.

3.6. Pengujian *Software*

Pengujian *Software* (perangkat Lunak) adalah elemen kritis dari jaminan kualitas perangkat lunak dan mempresentasikan kajian pokok dari spesifikasi, desain dan pengkodean.

Metode pengujian yang digunakan oleh penulis adalah *black-box* testing. *Black Box Testing* digunakan untuk menguji fungsi-fungsi dari perangkat lunak yang dirancang. Pengujian *black box* berfokus pada persyaratan fungsional perangkat lunak. Dengan demikian, pengujuan *black*

box memungkinkan perekayasa perangkat lunak mendapat serangkaian kondisi input yang sepenuhnya menggunakan semua persyaratan fungsional untuk semua program. Pengujian *black box* merupakan pendekatan komlementer yang kemungkinan besar mampu mengungkap kelas kesalahan. Pengujian *black box* berusaha menemukan kesalahan dalam kategori sebagai berikut :

1. Fungsi yang tidak benar atau hilang.
2. Kesalahan antar muka.
3. Kesalahan dalam struktur data atau akses *database* eksternal.
4. Kesalahan kinerja dan Inisialisasi kesalahan terminasi.