BAB II

LANDASAN TEORI

2.1. Pengertian Sistem

Sistem adalah suatu kumpulan dari unsur atau elemen–elemen yang saling berkaitan atau berinteraksi dan saling mempengaruhi dalam melakukan kegiatan bersama untuk mencapai suatu tujuan tertentu.

Menurut Jogianto (2005:2) mengemukakan bahwa Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata seperti tempat, benda dan orang-orang yang betul-betul ada dan terjadi.

Definisi sistem menurut Hanif Al-fatta (2007:18) Sistem adalah kumpulan atau group dari bagian atau komponen apapun baik fisik maupun non fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. Sistem memiliki syarat-syarat sebagai berikut:

1. Sistem harus dibentuk untuk menyelesaikan masalah.
2. Elemen sistem harus mempunyai rencana yang ditetapkan.
3. Adanya hubungan diantara elemen sistem.

4. Unsur dasar dari proses (arus informasi,energi dan material) lebih penting dari pada elemen sistem.

5. Tujuan organisasi lebih penting dari pada tujuan elemen.

2.1.1. Karakteristik Sistem

Jogianto (2005:3) mengemukakan sistem mempunyai karekteristik atau sifat-sifat tertentu,yakni :

1. Komponen

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari sistem. Setiap subsistem mempunyai sifat-sifat dari sistem untuk menjalankan suatu fungsi tertentu mempengaruhi proses sistem secara keseluruhan.

2. Batasan Sistem

Batasan sistem (*boundary)* merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batasan suatu sistem menunjukan ruang lingkup dari sistem tersebut.

3. Lingkungan Luar Sistem

Lingkungan luar (*evinronment*) dari suatu sistem adalah apapun diluar batas sistem yang mempengaruhi operasi. Lingkungan luar sistem dapat bersifat

menguntungkan dana dapat juga bersifat menguntungkan sistem tersebut. Lingkungan luar yang menguntungkan berupa energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. sedangkan lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan menggangu kalangsungan hidup dari sistem.

4. Penghubung Sistem

Penghubung (*interfance)* merupakan media penghubung antara satu subsistem dengan subsistem yang lainya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem yang lainya. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainya membentuk satu kesatuan.

Sedangkan menurut Burch dan Grundnitski (dalam Jogiyanto 2005:196) desain sistem dapat didefinisikan sebagai penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi. Desain sistem menetukan bagaimana suatu sistem akan menyelesaikan tahap ini menyangkut mengkonfigurasi dari komponen-komponen perangkat lunak dan perangkat keras dari suatu sistem sehingga setelah instalasi dari sistem akan benar-benar memuaskan rancang bangun yang telah ditetapkan pada akhir tahap analisis sistem (Jogiyanto 2005:196). Berdasarkan beberapa defenisi diatas maka desain sistem dapat diartikan sebagai berikut:

Tahap setelah analisis dari siklus pengembangan sistem :

1. Pendefenisisn dari kebutuhan-kebutuhan funsional.
2. Persiapan untuk rancang bangun untuk implementasi.
3. Menggambarkan bagaimana suatu sistem dibentuk.
4. Yang dapat berupa penggambaran, perencanaan dan pembuatan sketsa atau pengaturan dari beberapa elemen yang terpisah kedalam suatu kesatuan yang utuh dan berfungsi.
5. Termasuk menyangkut mengkonfigurasi dari komponenen-komponen perangkat lunak dan perangkat keras dari suatu sistem. Tahap desain sistem mempunyai dua tujuan yaitu :
6. Masukan Sistem

Masukan *(input)* sistem adalah energi yang masuk kedalam sistem. Masukan dapat berupa masukan perawatan *(maintenance input)* dan masukan sinyal *(signal input*). *Maintenance input* adalah energi yang dimasukan supaya tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam komputernya dan data adalah signal input untuk diolah menjadi informasi.

7. Keluaran Sistem

Keluaran *(output)* sistem adalah hasil dari energi yang diolah dan diklafikasikan menjadi keluaran yang berguna dan sisa pembuangan. Misalnya untuk sistem komputer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

8. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu sistem produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi.

9. Sasaran Sistem

Sebuah sistem sudah tentu mempunyai sasaran ataupun tujuan. Dengan adanya sasaran sistem, maka kita dapat menentukan masukan yang dibutuhkan sistem dan keluaran apa yang akan dihasilkan sistem tersebut dapat dikatakan berhasil apabila mencapai atau mengenai sasaran atau pun tujuan.

2.1.2. Klasifikasi Sistem

Menurut(Jogiyanto:2005) sistem dapat diklasifikasikan kedalam beberapa sudut padangan diantaranya sebagai berikut :

1. Sistem diklasifikasikan sebagai sistem abstrak *(abstract sistem)* dan sistem fisik *(physical sistem).* Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide tidak nampak secara fisik. Sistem fisik merupakan sistem yang ada secara fisik.

2. Sistem diklasifikasikan sebagai sistem alamiah dan buatan manusia. Sistem alamiah adalah sistem yang terjadi melalui proses alami, tidak dibuat manusia. Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin yang disebut *Human Machine Sistem*.

3. Sistem diklarifikasikan sebagai sistem tertentu dan sistem tidak tentu. Sistem tertentu beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi antara bagian-bagiannya dan dideteksi dengan pasti sehingga keluar dari sistem dapat diramalkan.

4. Sistem diklarifikasikan sebagai sistem terbuka dan tertutup. Sistem tertutup merupakan sistem yang tidak berhubungan dan tidak dipengaruhi lingkungan luarnya. Sistem terbuka adalah sistem yang berhubungan dan terpengaruhi oleh lingkungan luarnya.

2.2. Pengertian Informasi

Informasi adalah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta.

Abdul Kadir (2002:31) McFadden dkk (1999) mendefinisikan informasi sebagai data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut.

Pengertian Informasi Menurut George H. Bodnar (2000:1) “Informasi adalah data yang diolah sehingga dapat dijadikan dasar untuk mengambil keputusan yang tepat”.

2.3. Pengertian Sistem Informasi

Sistem informasi adalah suatu sistem di dalam suatu organisasi, dimana penggunaan sistem informasi dalam sebuah organisasi tersebut mendukung dalam mengumpulkan, mengolah dan menyediakan informasi yang berguna di dalam perencanaan, pemanfaatan dan pengendalian. Suatu organisasi yang tumbuh dan

menjadi lebih kompleks membuat manajemen, melakukan permintaan yang semakin besar terhadap fungsi sistem informasi.

Sistem informasi adalah suatu sistem di dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian yang mendukung fungsi operasi organisasi yang bersifat manajerial dengan kegiatan strategi dari suatu organisasi untuk dapat menyediakan kepada pihak luar tertentu dengan laporan-laporan yang diperlukan (Tata Sutabri:2004).

2.4. Pengertian Penjualan

Penjualan merupakan pembelian sesuatu (barang atau jasa) dari suatu pihak kepada pihak lainnya dengan mendapatkan ganti uang dari pihak tersebut. Penjualan juga merupakan suatu sumber pendapatan perusahaan, semakin besar penjualan maka semakin besar pula pendapatan yang diperoleh perusahaan.

Pengertian penjualan menurut Henry Simamora (2000:24) dalam buku "Akuntansi Basis Pengambilan Keputusan Bisnis" menyatakan bahwa: "Penjualan adalah pendapatan lazim dalam perusahaan dan merupakan jumlah kotor yang dibebankan kepada pelanggan atas barang dan jasa".

Berdasarkan pengertian di atas, maka dapat disimpulkan bahwa penjualan adalah persetujuan kedua belah pihak antara penjual dan pembeli, dimana penjual menawarkan suatu produk dengan harapan pembeli dapat menyerahkan sejumlah uang sebagai alat ukur produk tersebut sebesar harga jual yang telah disepakati.

2.5. Pengertian Pembelian

Pembelian adalah proses penemuan sumber dan pemesanan bahan, jasa, dan perlengkapan. Tujuan utamanya adalah memperoleh bahan dengan biaya serendah mungkin yang konsisten dengan kualitas dan jasa yang di persyaratkan. (http://indonesia.smetoolkit.org//indonesia/id/content/id/435/pengelolaan-bahananda).

2.6. Desain Penelitian

2.6.1. Metode Deskriptif

Metode deskripsi adalah suatu metode dalam penelitian status kelompok manusia, suatu objek, suatu set kondisi, suatu sistem pemikiran, ataupun suatu kelas peristiwa pada masa sekarang.

2.7. Metode Pendekatan Sistem

Metode pendekatan sistem yang digunakan oleh penulis adalah pendekatan dengan Object Oriented yang menggunakan *OOA(Object Oriented Analysis)* dan *OOD(Object Oriented Design).*

2.7.1. Konsep Dasar Object Oriented Analysis (OOA)

1. Objek adalah suatu benda, entitas, biasanya berupa kata benda (*noun*), sesuatu yang dapat kita angkat maupun kita tendang atau pukul. Beberapa objek berupa benda hidup dan beberapa tidak.

2. Kelas merupakan sebuah model, pola atau *blueprint* (cetak biru) yang digunakan untuk menciptakan sebuah objek (Benyamin L.Sinaga:2005).

3. Enkapsulasi merujuk pada sebuah objek yang menyembunyikan atau mengenkapsulasi atribut-atributnya terhadap operasi yang dikenakan (analoginya objek tersimpan dan terkunci dalam sebuah kapsul dan operasi berada di tepian luar kapsul tersebut).

4. Asosiasi dan Agregasi adalah hubungan antar objek yang saling membutuhkan. Agregasi adalah menggambarkan seluruh bagian dari objek.

2.7.2. Karakteristik Dari Object

1. Identitas berarti bahwa data yang diukur mempunyai nilai tertentu yang berbeda. Entitas disebut objek.

2. Objek dapat kongkrit, seperti halnya arsip dalam sistem atau konseptual seperti kebijakan penjadwalan dalam *multiprocessing* dalam suatu sistem operasi.

3. Setiap objek mempunyai sifat yang melekat pada identitasnya.

4. Dua objek dapat berbeda walaupun semua nilai atributnya identik.

2.7.3. Istilah-istilah Object

1. Atribut : Data item yang menegaskan objek**.**

2. Operasi : Fungsi di dalam kelas yang dikombinasikan ke bentuk tingkah laku kelas.

3. Metode : Pelaksanaan prosedur (bagian dari kode yang mengeksekusi respon terhadap permintaan objek lain didalam sistem).

2.7.4. Kelas Object

1. Kelas merupakan gambaran sekumpulan objek yang terbagi dalam atribut,operasi,metode,hubungan dan makna yang sama.

2. Suatu kegiatan mengumpulkan data (atribut) dan perilaku (operasi) yang mempunyai struktur data sama ke dalam satu grup.

3. Kelas objek merupakan wadah bagi objek. Dapat digunakan untuk menciptakan objek.

4. Objek mewakili fakta keterangan dari sebuah kelas.

2.8. Alat Bantu Analysis dan Perancangan

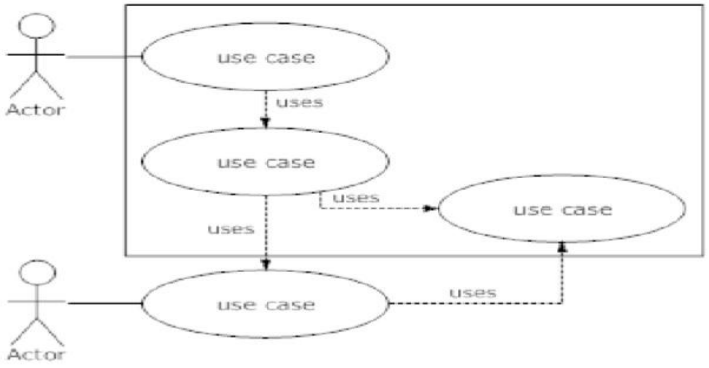
2.8.1. UML (Unified Modelling Languages)

Menurut Miftakul Huda (2010:137) UML adalah sebuah “bahasa” yang telah menjadi standar dalam industri untuk visualisasi, merancang dan mendokumentasikan sistem peranti lunak.

Menurut Miftakul Huda (2010:138) Secara konsep dasar, UML mendefinisikan delapan diagram, yakni sebagai berikut :

2.8.1.1. Use Case Diagram

Mengggambarkan fungsionalitas dari sebuah sistem (apa fungsinya), yang mempersentasikan sebuah interaksi antara aktor dengan sistem (sebuah pekerjaan). Sebuah use case dapat *meng-include* fungsionalitas use case lain. Sebuah use case dapat *di-include* oleh lebih dari satu use case lain, sehingga dapat duplikasi fungsionalitas dapat dihindari dengan menarik keluar fungsionalitas yang *common.* Sebuah use case juga dapat *meng-extend* use case lain dengan behaviour-nya sendiri. Berikut adalah gambar contoh dari use case diagram :



Gambar 2.1. *Use Case* Diagram

2.8.1.2. Class Diagram

Class adalah sebuah spesifikasi objek, yang memiliki atribut atau properti dan layanan atau fungsional (metode atau fungsi). Class diagram menggambarkan struktur dan deskripsi kelas, package dan objek beserta hubungan satu sama lain, seperti containment, pewarisan, asosiasi, dan lain-lain. Kelas memiliki tiga hal pokok: Nama dan stereotype, Atribut dan Metode Hubungan antar kelas sebagai berikut :

1. Asosiasi, yaitu hubungan statis antarkelas, biasanya menggambarkan kelas yang memiliki atribut berupa kelas lain. Terdapat banyak jenis asosiasi, misalnya :

a) Asosiasi sederhana : bentuk asosiasi sederhana.

b) Agregasi, yaitu hubungan yang menyatakan bagian, biasanya hubungan data master dan detailnya. Misal satu pembelian terdiri atas sejumlah item barang.

c) Navigability : menunjukan arah query atau komunikasi antar objek, bisa satu atau dua arah, terlihat pada tanda panahnya .

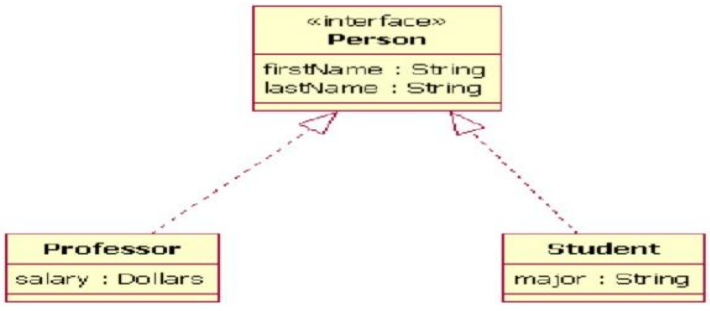
d) Campuran atau Composit : campuran asosiasi.

2. Generalisasi, yaitu hubungan hierarkis antar anak dan bapak. Karena kelas dapat diturunkan dari kelas lain dan mewarisi semua atribut dan metode kelas asalnya serta menambahkan fungsionalitas baru.

3. Implementasi (*Realization*), yaitu hubungan antar objek yang menjamin adanya pola khusus dalam perilaku anggota objek lainnya. Ini dapat diwujudkan dengan adanya kelas yang mengimplementasikan interface tertentu.

4. Ketergantungan (*Dependency)* yaitu sebuah kelas membutuhkan objek lain untuk bisa memfungsikan dirinya sendiri dengan baik.

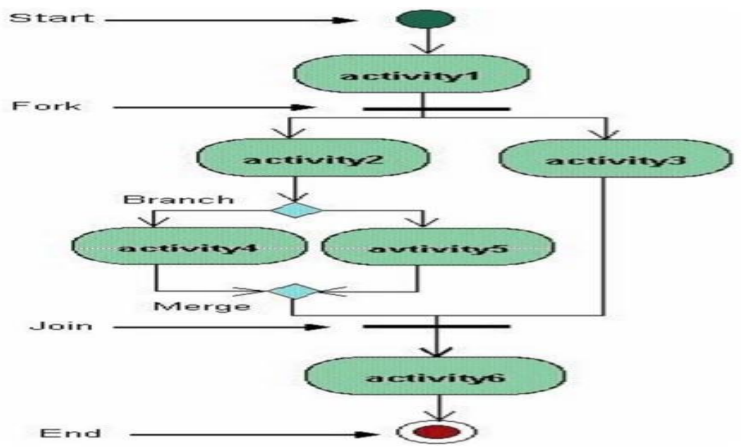
5. Hubungan dinamis, yaitu rangkaian pesan *(message)* yang di-passing dari satu kelas ke kelas lain. Dalam hubungan antarkelas yang bertipe asosiasi, akan muncul multiplicity yaitu batasan jumlah hubungan *(constraint)* yang dilambangkan dengan angka 0, 1, dan * yang artinya 0 adalah belum ada atau tidak ada, 1 adalah ada satu, dan * adalah lebih dari satu atau banyak. Berikut adalah gambar contoh dari class diagram :



Gambar 2.2. *Class* Diagram

2.8.1.3. Activity Diagram

Diagram ini menggambarkan berbagai aktivitas dalam sistem yang sedang dirancang, mulai dari titik awal, melalui kondisi (*decision*) yang mungkin terjadi kemudian sampai pada titik akhir. Diagram ini juga mampu proses paralel yang mungkin terjadi beberapa eksekusi. Diagram ini tidak menggambarkan perilaku atau proses internal sebuah sistem maupun interaksi antar-subsistem, tetapi lebih menggambarkan proses-proses dan jalur-jalur aktivitas secara umum *(global*). Berikut adalah gambar contoh dari Activity diagram :

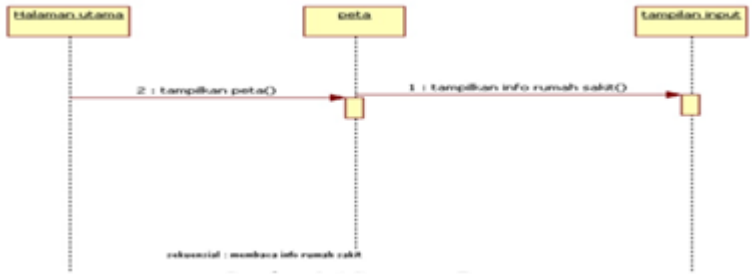


Gambar 2.3. *Activity* Diagram

2.8.1.4. Sequence Diagram

Diagram ini menggambarkan interaksi antar objek di dalam dan di sekitar sistem (termasuk pengguna, display dan sebagainya) berupa message

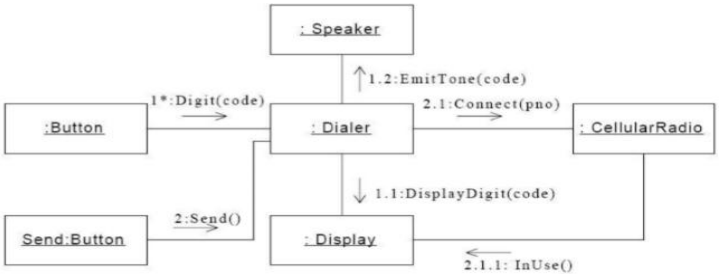
yang digambarkan terhadap waktu, Sequence diagram terdiri atas dimensi vertikal (waktu) dan dimensi horizontal (objek-objek yang terkait). Biasanya digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respon dari sebuah event untuk menghasilkan output tertentu. Diawali dari sebuah aktivitas tertentu, kemudian berproses mengikuti urutan tertentu yang bisa terlihat melalui message antarobjeknya. Berikut adalah gambar contoh dari sequence diagram :



Gambar 2.4. *Sequence* Diagram

2.8.1.5. Collaboration Diagram

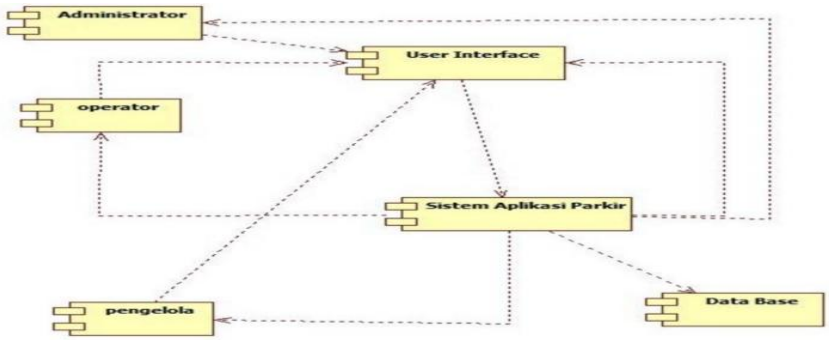
Collaboration diagram juga menggambarkan interaksi antarobjek seperti sequence diagram, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian message. Setiap message memiliki urutan angka, level tertinggi dimulai dari nomor 1, sedangkan untuk message dari level yang sama memiliki prefiks yang sama. Berikut adalah gambar contoh dari collaboration diagram :



Gambar 2.5. *Collaboration* Diagram

2.8.1.6. Component Diagram

Diagram ini menggambarkan struktur dan hubungan antar komponen peranti lunak, termasuk ketergantungan *(defendency*) Di antaranya modul berisi kode, baik berisi ource kode, binary, library, executable. Berikut adalah gambar contoh dari component diagram:

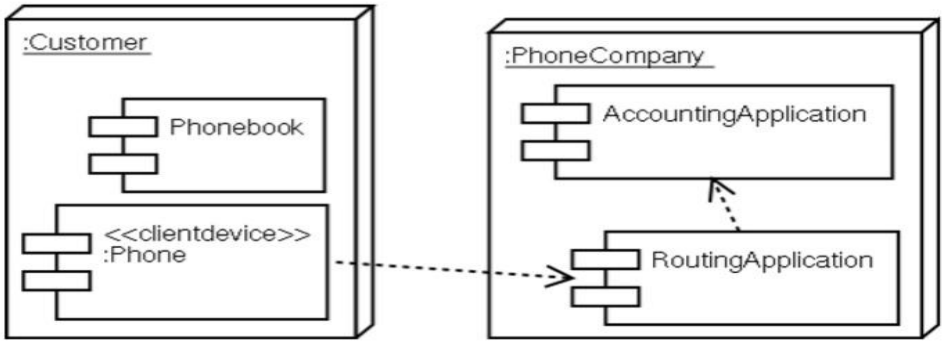


Gambar 2.6. *Component* Diagram

2.8.1.7. Deployment Diagram

Diagram ini menggambarkan detail bagaimana komponen dibentuk dan didistribusikan (*deploy*) dala infrastuktur sistem. Dimana komponen akan

terletak pada mesin, server atau peranti keras apa. Bagaimana jaringan pada lokasi tersebut. Berikut adalah gambar contoh dari deployement diagram :



Gambar 2.7. *Deployment* Diagram

2.8.2. Bahasa Pemrograman Yang Digunakan

2.8.2.1. Java

Java adalah bahasa pemograman tingkat tinggi yang berorientasi objek dan pemograman java tersusun dari bagian yang disebut kelas. Kelas terdiri atas metode-metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para pemogram java banyak mengambil keuntungan dari kumpulan kelas di pustaka kelas java, yang disebut dengan Java Application Programming Interface (API). Kelas-kelas ini diorganisasikan menjadi sekelompok yang disebut paket (package). Java API telah menyediakan fungsionalitas yang memadai untuk menciptakan applet dan aplikasi canggih, Jadi ada dua hal yang harus dipelajari dalam Java, yaitu mempelajari bahasa Java dan bagaimana mempergunakan kelas pada

Java API. Kelas merupakan satu-satunya cara menyatakan bagian eksekusi program. Beberapa keunggulan Java yaitu Java merupakan bahasa yang sederhana. Java dirancang agar mudah dipelajari dan digunakan secara efektif. Java tidak menyediakan fitur-fitur rumit bahasa pemograman tingkat tinggi, serta banyak pekerjaan pemograman yang mulanya harus dilakukan manual, sekarang digantikan dikerjakan Java secara otomatis seperti alokasi memori.

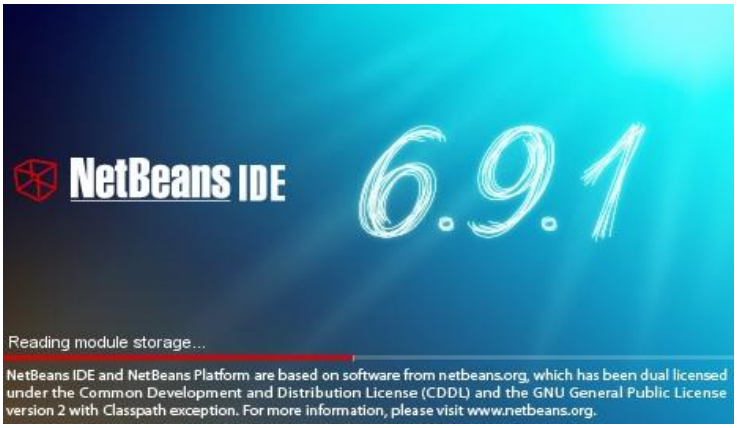
[Sumber : http://id.wikipedia.org/wiki/Java]

2.8.3. Perangkat Lunak Yang Digunakan

Perangkat lunak digunakan untuk pendukung program. Perangkat lunak yang digunakan akan dijelaskan sebagai berikut :

2.8.3.1. NeatBeans 6.9.1

Netbeans merupakan salah satu IDE yang dikembangkan dengan bahasa pemograman java. Netbeans mempunyai lingkup pemograman yang terintregrasi dalam suatu perangkat lunak yang didalamnya menyediakan pembangunan pemograman GUI, text editor, compiler dan interpreter. Netbeans adalah sebuah perangkat lunak open source sehingga dapat digunakan secara gratis untuk keperluan komersial maupun nonkomersial yang di dukung oleh Sun Microsystem.



Gambar 2.8. NetBeans 6.9.1

2.8.3.2. XAMPP

XAMPP adalah perangkat lunak bebas, yang mendukung banyak sistem operasi, merupakan kompilasi dari beberapa program. Fungsinya adalah sebagai server yang berdiri sendiri (localhost), yang terdiri atas program Apache HTTP Server, MySQL database dan penerjemaah bahasa yang ditulis dengan bahasa pemograman PHP dan Perl. Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perl. Program ini tersedia dalam GNU (General Piblic License), merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis.Mengenal bagian XAMPP yang biasa digunakan pada umumnya :

1. phpMyAdmin merupakan bagian untuk mengelola basis data MySQL yang ada dikomputer. Untuk membukanya buka browser lalu ketikkan alamat http://localhost/phpMyAdmin, maka akan muncul halaman phpMyAdmin.

2. Kontrol Panel yang berfungsi untuk mengelola layanan (service) XAMPP. Seperti menghentikan (stop) layanan, ataupun memulai (start).

[Sumber : http://id.wikipedia.org/wiki/XAMPP]



Gambar 2.9. XAMPP

2.8.3.3. iReport 5.0.1

Menurut Hendra Kurniawan, Eri dan Nur (2011:38), iReport adalah report designer visual yang dibangun pada JasperReport. iReport bersifat intuitif dan mudah digunakan pembangun laporan visual atau desainer untuk JasperReport dan tertulis dalam kitab Java. Sebagai alternatif, terdapat tools iReport (dengan library JasperReport) yang dapat membantu dalam pembuatan laporan. Library JasperReport sendiri merupakan Java Library (JAR) yang bersifat open dan dirancang untuk menambahkan kemampuan pelaporan (reporting capabilities) pada aplikasi java. JasperReport memiliki sejumlah fitur, diantaranya adalah sebagai berikut:

1. Layout dan desain laporan yang fleksibel.
2. Dapat menampilkan laporan dalam bentuk teks maupun gambar (chart).
3. Dapat menghasilkan report dalam berbagai format, seperti html, pdf, rtf, xls, dan csv.
4. Dapat menerima data dari berbagai sumber data, seperti: JDBC, Bean Collection, ResultSet, CSV, XML dan Hibernate.