BAB II

LANDASAN TEORI

2.1. Konsep Dasar Sistem

Untuk mendefinisikan suatu sistem pakar harus mempunyai konsep dasar untuk memperkuat pendefinisiannya. Sistem mempunyai suatu konsep yang mendasari sebuah pengertian-pengertian yang di kemukakan oleh berbagai pakar untuk mendefinisikan dari suatu sistem itu sendiri.

2.1.1 Pengertian Sistem

Sistem berasal dari bahasa Latin (*systema*) dan bahasa Yunani (*sustema*) adalah suatu kesatuan yang terdiri dari komponen atau elemen yang dihubungkan bersama untuk memudahkan aliran informasi, materi atau energi. Dalam pengertian yang paling umum, sebuah sistem adalah sekumpulan benda yang memiliki hubungan di antara mereka.

(sumber : http://id.wikipedia.org/wiki/Sistem/13 Juni 2012).

Sedangkan menurut beberapa ahli definisi sistem yaitu, menurut Abdul Kadir (2003:54) Sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan. Selanjutnya, menurut Jogiyanto HM (2001:12) Analisis Dan Desain Sistem Informasi. Andi Offset. Yogyakarta. Sistem adalah kumpulan dari komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu. Sedangkan menurut Raymond McLeod (2004) dikutip oleh Yakub dalam

buku Pengantar Sistem Informasi (2012:1) mendefiniskan sistem adalah sekelompok elemen-elemen yang terintegrasi dengan tujuan yang sama untuk mencapai tujuan. Sistem juga merupakan suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, terkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk tujuan tertentu.

Sehingga berdasarkan dari definisi sistem diatas dapat disimpulkan, sistem merupakan seperangkat elemen-elemen yang saling terhubung dan saling berinteraksi melakukan kegiatan untuk mencapai tujuan yang sama.

Prinsip umum sistem :

1. Sistem merupakan bagian dari sistem yang lebih besar, sekaligus sistem tersebut dapat dipartisi menjadi sub sistem-sub sistem yang lebih kecil.
2. Sistem yang lebih terspesialisasi akan kurang dapat beradaptasi untuk menghadapi keadaan-keadaan yang berbeda.
3. Lebih besar ukuran sistem, maka akan memerlukan sumber daya yang lebih banyak untuk operasi dan pemeliharaannya.
4. Sistem senantiasa mengalami perubahan, tumbuh dan berkembang.

2.1.2 Elemen Sistem

Menurut McLeod yang dikutip oleh Yakub (2012:3) tidak semua sistem memiliki kombinasi elemen-elemen yang sama, tetapi susunan dasarnya sama. Elemen-elemen yang terdapat dalam sistem ditandai dengan adanya :

a. Tujuan

   Tujuan ini menjadi motivasi yang mengarahkan pada sistem, karena tanpa tujuan yang jelas sistem menjadi tidak terarah dan tidak terkendali.

Misalnya dapat berupa tujuan organisasi, kebutuhan organisasi, permasalahan yang ada dalam suatu organisasi maupun urutan prosedur untuk mencapai tujuan organisasi.

b. Batasan

Batasan (*boundary*) sistem adalah pemisah antara sistem dan daerah diluar sistem. Selain itu juga sebagai batasan-batasan dari tujuan yang akan dicapai oleh sistem. Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem dan juga aturan-aturan.

c. Kontrol Sistem

Kontrol atau pengawasan sistem merupan pengawasan terhadap pelaksanaan pencapaian tujuan dari sistem tersebut. Kontrol sistem dapat berupa kontrol terhadap pemasukan data (*input*), kontrol terhadap keluaran data (*output*), kontrol terhadap pengolahan data, kontrol terhadap umpan balik dan sebagainya.

d. Masukan

Masukan (*input*) sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan dapat berupa hal-hal berwujud maupun yang tidak berwujud. Masukan berwujud adalah bahan mentah, sedangkan yang tidak berwujud adalah informasi. Masukan juga dapat berupa jenis data.

e. Proses

Proses merupakan elemen yang bertugas melakukan perubahan atau transformasi dari masukan/data menjadi keluaran/informasi yang berguna dan lebih bernilai.

f. Keluaran

Keluaran (*output*) merupakan hasil dari input yang sudah dilakukan pemerosesan sistem dan keluaran dapat menjadi masukan untuk subsistem lain. *Output* ini bisa berupa laporan grafik, diagram batang dan sebagainya

g. Umpan balik

Umpan balik ini digunakan untuk mengendalikan masukan maupun proses. Umpan balik juga bertugas mengevaluasi bagian dari *output* yang dikeluarkan. Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan, elemen ini sangat penting untuk kemajuan suatu sistem. Karena umpan balik bertugas sebagai evaluasi dari sistem, sehingga digunakan untuk melakukan perbaikan, pemeliharaan sistem dan sebagainya.

2.1.3 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu, Al-bahran bin ladjamudin (2005:3) menyebutkan bahwa karakteristik sistem yaitu :

1. Komponen-komponen

Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponen-

komponen sistem atau elemen-elemen sistem dapat berupa suatu subsistem atau bagian-bagian dari subsistem.

2. Batasan Sistem *(Boundary)*

   Batas sistem merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luarnya. Batas sistem ini memungkinkan suatu sistem dipandang sebagai suatu kesatuan. Batas suatu sistem menunjukkan ruang lingkup dari sistem tersebut.

3. Lingkungan luar sistem

   Lingkungan dari sistem adalah apapun di luar batas dari sistem yang mempengaruhi operasi sistem. Lingkungan luar sistem dapat bersifat menguntungkan dan dapat juga bersifat merugikan sistem tersebut. lingkungan luar yang mengutungkan merupakan energi dari sistem dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak akan mengganggu kelangsungan hidup dari sistem .

4. Penghubung

   Penghubung merupakan media perantara antar subsistem. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke subsistem lainnya. *Output* dari satu subsistem akan menjadi input untuk subsistem yang lainnya dengan melalui penghubung. Dengan penghubung satu subsistem dapat berinteraksi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan

   Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa *maintenance input* dan sinyal *input*. *Maintenance input* adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Sinyal *input* adalah energi yang diproses untuk didapatkan keluaran.

6. Keluaran

   Keluaran adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supra sistem.

7. Pengolah

   Suatu sistem dapat mempunyai suatu bagian pengolah atau sistem itu sendiri sebagai pengolahnya. 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.

8. Sasaran atau tujuan

   Suatu sistem pasti mempunyai tujuan atau sasaran. Kalau suatu sistem tidak mempunyai sasaran, maka operasi sistem tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukan yang dibutuhkan sistem dan keluaran yang akan dihasilkan sistem. Suatu sistem dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.1.4 Klasifikasi Sistem

Menurut Yakub (2012:4) pada buku Pengantar Sistem Informasi, Sistem dapat diklasifikasikan dari beberapa sudut pandang diantaranya :

a Sistem abstrak *(abstract system)*

Sistem abstrak adalah sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik. Sistem teologia yang berisi gagasan tentang hubungan manusia dengan Tuhan merupakan contoh *abstract system*.

b Sistem fisik *(physical system)*

Sistem fisik adalah sistem yang ada secara fisik, sistem komputer, sistem akuntansi, sistem produksi, sistem sekolah, dan sistem transportasi merupakan contoh *physical system.*

c Sistem tertentu *(deterministic system)*

Sistem tertentu adalah sistem yang beroperasi dengan tingkah laku yang dapat diprediksi, interaksi antara bagian dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan. Sistem komputer sudah diprogramkan, merupakan contoh *deterministic system* karena program komputer dapat diprediksi dengan pasti.

d Sistem tak tentu *(probabilistic system)*

Sistem tak tentu adalah suatu sistem yang kondisi masa depannya tidak dapat diprediksikan karena mengandung unsur probabilitas. Sistem arisan merupakan contoh *probabilistic system* karena sistem arisan tidak dapat diprediksikan dengan pasti.

e Sistem tertutup *(close system)*

Sistem tertutup merupakan sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Sistem ini tidak berinteraksi dan tidak dipengaruhi pleh lingkungan, misalnya reaksi kimia dalam tabung terisolasi.

f Sistem terbuka *(open system)*

Sistem ini adalah sistem yang berhubungan dengan lingkungan dan dipengaruhi oleh lingkungan. Sistem perdagangan merupakan contoh *open system*, karena dapat dipengaruhi oleh lingkungan.

2.2 Konsep Dasar Informasi

Memahami konsep dasar informasi merupakan hal yang sangat penting (vital) dalam mendesain sebuah sistem informasi yang efektif *(effective business system)*. Menyiapkan langkah atau metode dalam menyediakan informasi yang berkualitas adalah tujuan dalam mendesain sistem baru.

2.2.1 Pengertian Informasi

Menurut Raymond McLeod dikutip oleh Yakub (2012:8) pada buku Pengertian Sistem Informasi, Informasi adalah data yang telah diolah menjadi bentuk yang memiliki arti bagi si penerima dan bermanfaat bagi pengambilan keputusan saat ini atau mendatang. Dalam buku Konsep Dasar Informasi (2005:8), Jogiyanto menyatakan Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

Disebutkan diatas bahwa data merupakan sumber awal dari informasi. Data adalah representasi fakta nyata yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian adalah sesuatu yang terjadi pada saat tertentu. Kesatuan nyata *(fact and entity)* adalah berupa suatu objek nyata seperti tempat, benda dan orang yang betul-betul ada dan terjadi. Data merupakan bentuk yang dapat memberikan manfaat yang besar bagi penerimanya, namun data saja tidak cukup, karena data jika diibaratkan bahan mentah, yang bisa saja salah dalam penggunaanya jika ada di tangan orang yang tidak tepat. Sehingga perlu suatu proses yang nantinya dilakukan untuk menghasilkan informasi.

Pada kenyataannya, informasi merupakan hal yang dibutuhkan baik dalam organisasi maupun kehidupan sosial. Perusahaan maupun masyarakat mengambil tindakan disesuaikan atas informasi yang mereka dapatkan. Sehingga dengan adanya informasi perusahaan dapat terus berkembang, itulah alasan informasi begitu dibutuhkan.

2.2.2 Kualitas Informasi

1. Akurat *(Accurate)*

   Informasi harus bebas dari kesalahan-kesalahan dan tidak menyesatkan, dalam hal ini informasi harus jelas mencerminkan maksudnya.

2. Tepat waktu *(Time Lines)*

   Informasi yang datang pada penerima tidak boleh terlambat. Informasi yang sudah usang tidak mempunyai nilai lagi karena informasi merupakan suatu landasan dalam mengambil sebuah keputusan dimana bila

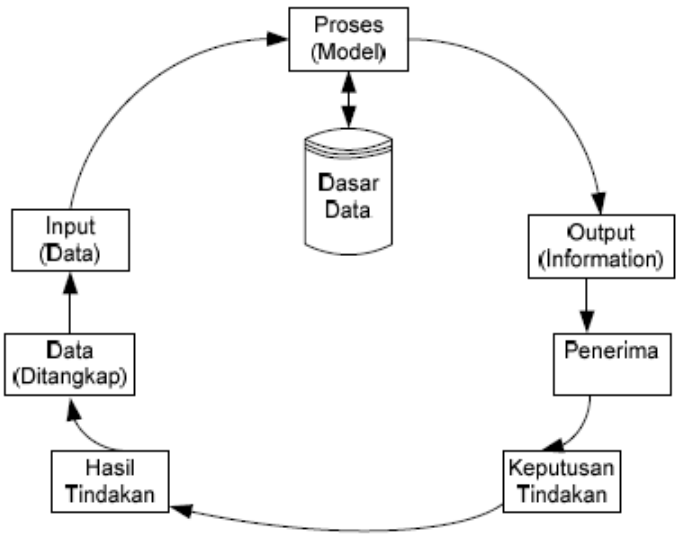
pengambilan keputusan terlambat maka akan berakibat fatal untuk organisasi.

3. Relevan *(Relevant)*

   Informasi tersebut mempunyai manfaat untuk pemakainya karena informasi tersebut akan dipakai untuk pengambilan keputusan. Relevansi informasi untuk tiap-tiap individu berbeda tergantung pada yang menerima dan yang membutuhkan.

2.2.3 Siklus Informasi

Data merupakan bentuk yang masih mentah yang belum dapat bercerita banyak, sehingga perlu diolah lebih lanjut. Data diolah melalui suatu model untuk dihasilkan informasi. Data tersebut akan ditangkap sebagai *input*, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus.



Gambar 2.1 Siklus Informasi

Sumber : *Analisis dan Design*. Jogiyanto, H.M (2008:9)

2.2.4 Nilai Informasi

Menurut Jogiyanto (2005:11) yang dikutip oleh Yakub (2012:9) nilai dari informasi (*value of information*) ditentukan oleh dua hal yaitu, manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya. Pengukuran nilai informasi biasanya dihubungkan dengan analisis *cost effectiveness* atau *cost benefit*.

Kedepannya informasi akan dinikmati oleh banyak pihak, sehingga penulis menyimpulkan bahwa untuk memperoleh informasi yang bernilai yaitu yang dapat bermanfaat dan lebih efektif bila dibandungkan dengan biaya untuk memperolehnya.

2.3 Konsep Dasar Sistem Informasi

Pada hakikatnya sistem informasi adalah seperangkat manusia, data dan prosedur yang bekerja sama secara koordinatif untuk mencapai tujuan tertentu. Tekananya terletak pada konsep sistem yang memperlihatkan bahwa berbagai komponen yang terlihat di dalamnya secara fungsional dan kooperatif mencapai tujuan yang sama. Kegiatan fungsional, strategi dan kooperatif itu meliputi pelaksanaan bisnis setiap hari, komunikasi informasi, manajemen aktifitas, pembuatan keputusan dan menghasilkan keluaran bersifat laporan yang dibutuhkan oleh pihak luar.

2.3.1 Pengertian Sistem Informasi

Andri Kristianto (2008), suatu sistem didalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi, mendukung operasi, bersifat

manajerial kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan. Pendapat lain, Raymon McLeod dalam Yakub (2012) menyatakan Sistem Informasi adalah sistem yang mempunyai kemampuan untuk mengumpulkan informasi dari semua sumber dan menggunakan berbagai media untuk menampilkan informasi. Informasi dalam suatu lingkungan sistem informasi harus mempunyai persyaratan umum sebagai berikut :

1. Harus diketahui oleh penerima sebagai referensi yang tepat.
2. Harus sesuai dengan kebutuhan yang ada dalam proses pembuatan/pengambilan keputusan.
3. Harus mempunyai nilai *surprise*, yaitu hal yang sudah diketahui hendaknya jangan diberikan.
4. Harus dapat menuntun pemakai untuk membuat keputusan. Suatu keputusan tidak selalu menuntut adanya tindakan.

2.3.2 Komponen Sistem Informasi

Sistem informasi mempunyai enam buah komponen atau disebut juga dengan blok bangunan *(building block)*, yaitu :

1. Komponen *input* atau komponen masukan
2. Komponen model
3. Komponen *output* atau komponen keluaran
4. Komponen teknologi
5. Komponen basis data
6. Komponen kendali

Keenam komponen ini harus ada bersama-sama dan membentuk satu kesatuan. Jika satu atau lebih komponen tersebut tidak ada, maka sistem informasi tidak akan dapat melakukan fungsinya, yaitu pengolahan data dan tidak dapat mencapai tujuannya, yaitu menghasilkan informasi yang relevan, tepat waktu dan akurat. Komponen-komponen dari sistem informasi ini dapat digambarkan sebagai berikut ini :

1. Blok Masukan (*Input Block*)

   Input merupakan data yang masuk ke dalam sistem informasi.

2. Blok Model (*Model Block*)

   Kombinasi prosedur, logika, dan model matemetik yang akan memanipulasi data *input* dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diingiinkan.

3. Blok Keluaran (*Output Block*)

   Keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

4. Blok Teknologi (*Technology Block*)

   Teknologi merupakan kotak alat *(tool box)* dalam sistem informasi. Teknologi digunakan untuk menerima *input*, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran dan membantu pengendalian dari sistem secara menyeluruh.

5. Blok Basis Data (*Database Block*)

   Merupakan kumpulan dari data yang saling berhubungan satu sama lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok Kendali (*Control Block*)

   Beberapa pengendalian yang dirancang secara khusus untuk menanggulangi gangguan-gangguan terhadap sistem.

Adapun beberapa elemen atau komponen dalam sistem informasi dalam buku Al-Bahra bin Ladjamudin (2005:14) dapat diklasifikasikan sebagai berikut:

a. *Hardware* dan *software* yang berfungsi sebagai mesin.
b. *People* dan *procedures* yang merupakan manusia dan tata cara menggunakan mesin.
c. Data merupakan jembatan penghubung antara manuia dan mesin agar terjadi suatu proses pengolahan data.

2.4 Pengertian Akademik

Kata akademik berasal dari bahasa Yunani yakni *academos* yang berarti sebuah taman umum (plasa) di sebelah barat laut kota Athena. Sesudah itu, kata *acadomos* berubah menjadi akademik, yaitu semacam tempat perguruan. Para pengikut perguruan tersebut disebut *academist*, sedangkan perguruan semacam itu disebut *academia*. Berdasarkan hal ini, inti dari pengertian akademik adalah keadaan orang-orang bisa menyampaikan dan menerima gagasan, pemikiran, ilmu pengetahuan, dan sekaligus dapat mengujinya secara jujur, terbuka, dan leluasa.

Sedangankan, kegiatan akademik adalah kegiatan pembelajaran di dalam atau di luar ruang kuliah, studio atau laboratorium, pengerjaan tugas-tugas, evaluasi pembelajaran, dan kegiatan administrasi yang menyertainya.
(sumber : http://www.pasca.its.ac.id/dataq/file_content/File/BAB_VI_-_Kegiatan_Akademik.pdf_/ 10 juni 2012)

2.5 Metode Pendekatan Sistem

Pendekatan Sistem adalah upaya untuk melakukan pemecahan masalah yang dilakukan dengan melihat masalah yang ada secara menyeluruh dan melakukan analisis secara sistem. Pendekatan sistem diperlukan apabila kita menghadapi suatu masalah yang kompleks sehingga diperlukan analisa terhadap permasalahan tadi, untuk memahami hubungan bagian dengan bagian lain dalam masalah tersebut, serta kaitan antara masalah tersebut dengan masalah lainnya.

Pendekatan sistem dapat diartikan sebagai titik tolak atau sudut pandang kita terhadap proses sebuah sistem yang merujuk pada pandangan tentang terjadinya suatu proses yang bersifat masih sangat umum dalam mewadahi, mengispirasi, dan melatari metode pendekatan dalam cakupan teoritis tertentu.

2.5.1 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek merupakan suatu teknik atau cara pendekatan dalam melihat permasalahan dan sistem (sistem perangkat lunak, sistem informasi, atau sistem lainnya). Pendekatan berorientasi objek akan memandang sistem yang akan dikembangkan sebagai suatu kumpulan objek yang

berkorespondensi dengan objek-objek dunia nyata. Rosa A.S-M.Shalahuddin (2011:86),

Keuntungan menggunakan metodologi berorientasi objek adalah sebagai berikut :

1. Meningkatkan Produktivitas

   Karena kelas dan objek yang ditemukan dalam suatu masalah masih dapat dipakai ulang untuk masalah lainnya yang melibatkan objek tersebut *(reusable).*

2. Kecepatan Pengembangan

   Karena sistem yang dibangun dengan baik dan benar pada saat analisis dan perancangan akan menyebabkan berkurangnya kesalahan pada saat pengkodean.

3. Kemudahan Pemeliharaan

   Karena dengan model objek, pola-pola yang cenderung tetap dan stabil dapat dipisahkan dan pola-pola yang mungkin sering berubah-ubah.

4. Adanya Konsistensi

   Karena sifat pewarisan dan penggunaan notasi yang sama pada saat analisis, perancangan maupun pengkodean.

5. Meningkatkan Kualitas Perangkat lunak

   Karena pendekatan pengembangan lebih dekat dengan dunia nyata dan adanya konsistensi pada saat pengembangannya, perangkat lunak yang dihasilkan akan mampun memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan.

2.5.1.1 Object Oriented Programming

Menurut Th. Ari Prabawati (2010:88) pada buku Pengembangan Aplikasi Database berbasis JavaDB dengan Netbeans mendefinisikan Object Oriented Programming adalah suatu cara mengorganisasi program dengan memodelkan objek-objek dunia nyata (seperti benda, sifat, sistem, dan lainnya) ke dalam sebuah bahasa pemograman. Dalam bahasa pemograman pemodelan OOP ini diaplikasikan dengan sebuah kelas, method, dan properti atau variabel. Java merupakan salah satu bahasa pemograman berorientasi objek.

2.5.1.2 Unified Modeling Language (UML)

Martin Fowler dalam bukunya yang berjudul UML Distitlled Edisi 3 - Panduan Singkat Bahasa Pemodelan Objek Standar (2005:1) Mengenalkan Unified Modelling Language atau UML merupakan notasi grafis yang didukung oleh meta-model tunggal, yang membantu pendeskripsian dan desain sistem perangkat lunak, khususnya sistem yang dibangun menggunakan pemrograman berorientasi objek (OO). UML merupakan standar yang relatif terbuka yang dikontrol oleh Object Management Group (OMG). UML lahir dari banyak bahasa pemodelan grafis berorientasi objek yang berkembang pesat pada akhir 1980-an sampai awal 1990-an.

UML memiliki 9 diagram, yaitu Diagram *Use Case, Class Diagram, Activity Diagram, Sequence Diagram, Component Diagram, Deployment Diagram, Package Diagram, Diagram Collaboration,* dan *Diagram Statechart.*

1. *Usecase Diagram*

   Diagram *usecase* bersifat statis, diagram ini memperlihatkan himpunan *usecase* dan aktor-aktor. Diagram ini mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat, terutama sangat penting untuk mengorganisasi dan memodelkan perilaku dari suatu sistem yang dibutuhkan dan diharapkan pengguna.

2. *Class Diagram*

   Class diagram mendeskripsikan jenis-jenis objek dalam sistem dan berbagai macam hubungan statis yang memperlihatkan himpunan kelas-kelas, antarmuka, kolaborasi, serta relasi-relasi. Diagram ini umum dijumpai pada pemodelan sistem berorientasi objek. Meskipun bersifat statis, diagram kelas juga sering memuat kelas-kelas aktif. *Class diagram* juga menunjukkan properti dan operasi sebuah *class* dan batasan-batasan yang terdapat dalam hubungan-hubungan objek tersebut.

3. *Activity Diagram*

   Bersifat dinamis, *diagram activity* adalah *type* khusus dari *diagram state* yang memperlihatkan aliran dari suatu aktifitas ke aktifitas yang lainya dalam suatu sistem. Diagram ini terutama penting dalam pemodelan fungsi-fungsi dalam suatu sistem dan memberi tekanan pada aliran kendali antarobjek.

4. *Sequence Diagram*

   Sequence diagram atau diagram urutan adalah diagram interaksi yang menekankan pada pengiriman pesan *(message)* dalam suatu waktu

tertentu. menggambarkan kelakuan objek pada *usecase* dengan mendeskripsikan waktu hidup objek dan *message* yang dikirimkan dan diterima antarobjek. Oleh karena itu untuk menggambar diagram sekuen maka harus diketahui objek-objek yang terlibat dalam sebuah *usecase* beserta metode-metode yang dimiliki kelas yang diinstansiasi menjadi objek itu. Banyaknya *diagram sequence* yang harus digambar adalah sebanyak pendefinisian *usecase* yang memiliki proses sendiri atau yang penting semua *usecase* yang telah didefinisikan interaksi jalannya pesan sudah dicakup pada *diagram sequence* sehingga semakin banyak *usecase* yang didefinisikan maka *diagram sequence* yang harus dibuat juga semakin banyak.

5. *Component Diagram*

   Diagram komponen ini memperlihatkan organisasi serta kebergantungan sistem/perangkat lunak pada komponen-komponen yang telah ada sebelumnya. Diagram ini berhubungan dengan diagram kelas dimana komponen secara tipikal dipetakan ke dalam satu atau lebih kelas-kelas, antarmuka, serta kolaborasi-kolaborasi.

6. *Deployment Diagram*

   Diagram ini memperlihatkan konfigurasi saat aplikasi dijalankan (saat *run-time*). Diagram ini memuat simpul-simpul beserta komponen yang ada di dalamnya. *Deployment diagram* berhubungan erat dengan diagram komponen dimana *deployment diagram* memuat satu atau lebih komponen-komponen. Diagram ini sangat berguna saat aplikasi kita

berlaku sebagai aplikasi yang dijalankan pada banyak mesin *(distributed computing*).

7. *Package Diagram*

   Package Diagram menyediakan cara mengumpulkan elemen-elemen yang saling terkait dalam diagram UML. Hamper semua diagram dalam UML dapat dikelompokkan menggunakan *package diagram.*

2.6 Metode Pengembangan Sistem

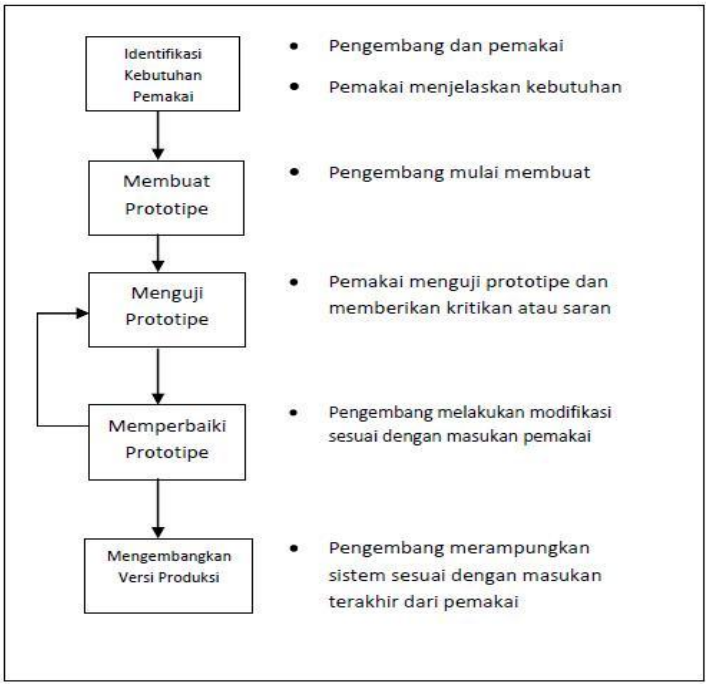
Menurut Roger S. Pressman, Ph.D (2002:39) Dalam menyelesaikan masalah suatu rekayasa perangkat lunak harus menggabungkan strategi pengembangan yang melingkupi lapisan proses, metode, dan alat-alat bantu. Strategi ini sering diacukan sebagai model proses atau paradigma rekayasa perangkat lunak. Model proses untuk rekayasa perangkat lunak dipilih berdasarkan sifat aplikasi dan proyeknya, metode dan alat-alat bantu yang dipakai, dan kontrol serta penyampaian yang dibutuhkan. Setiap model memiliki karakterstik masing-masing.

2.6.1 Model Prototipe

Menurut Abdul Kadir (2002 : 416) Prototipe merupakan suatu metode dalam pengembangan sistem yang menggunakan pendekatan untuk membuat suatu program dengan cepat dan bertahap sehingga segera dapat dievaluasi oleh pemakai. Prototipe membuat proses pengembangan sistem informasi menjadi lebih cepat dan lebih mudah, terutama pada keadaan kebutuhan pemakai sulit untuk diidentifikasi.

Secara garis besar, sasaran prototipe adalah sebagai berikut (Lucas,2000) :

1. Mengurangi waktu sebelum pemakai melihat sesuatu yang kongkrit dari usaha pengembangan sistem.
2. Menyediakan unpan balik yang cepat dari pemakai kepada pengembang.
3. Membantu menggambarkan kebutuhan pemakai dengan kesalahan yang lebih sedikit.
4. Meningkatkan pemahaman pengembangan dan pemakai terhadap sasaran yang seharusnya dicapai oleh sistem.
5. Menjadikan keterlibatan pemakai sangat berarti dalam analisi dan desain sistem.



Gambar 2.2 Mekanisme Pengembangan Sistem dengan Prototype

Sumber : Abdul Kadir, *Pengenalan Sistem Informasi* (2003 : 417)

Adapun tahapan-tahapan dari metode prototipe adalah sebagai berikut :

1. Mengidentifikasi Kebutuhan Pemakai. Pemakai dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasikan semua kebutuhan, dan garis besar sistem yang akan dibuat.
2. Membuat Prototipe. Setelah kebutuhan sistem didapat maka pengembang mulai membuat prototipe.
3. Menguji Prototipe. Pengujian dilakukan oleh pemakai apakah prototipe yang dibuat sudah sesuai dengan keinginan pelanggan.
4. Memperbaiki Prototipe. Pengembang melakukan modifikasi sesuai dengan masukan pemakai.
5. Mengembangkan Versi Produksi. Pengembang menyelesaikan sistem sesuai dengan masukan terakhir dari pemakai.

2.6.1.1 Kelebihan Dan Kelemahan Prototipe

Kelebihan prototipe :

1. Pendefinisian kebutuhan pemakai menjadi lebih baik karena keterlibatan pemakai yang intensif.
2. Meningkatkan kepuasan pemakai dan keterlibatan mereka yang sangat tinggi sehingga sistem memenuhi kebutuhan mereka dengan lebih baik.
3. Mempersingkat waktu pengembangan.
4. Memperkecil kesalahan disebabkan setiap versi prototipe, kesalahan segera terdeteksi oleh pemakai.

5. Pemakai memiliki kesempatan yang lebih banyak dalam meminta perubahan-perubahan.

Kelemahan prototipe :

1. Prototipe hanya bisa berhasil jika pemakai bersungguh-sungguh dalam menyediakan waktu dan pikiran untuk menggarap prototipe.
2. Kemungkinan dokumentasi terabaikan karena pengembang lebih berkosentrasi pada pengujian dan pembuatan prototipe.
3. Jika terlalu banyak proses pengulangan dalam membuat prototipe, ada kemungkinan pemakai menjadi jenuh dan memberikan reaksi yang negatif.

2.7 Arsitektur Jaringan

Untuk dapat mendukung terlaksananya program aplikasi yang akan dibangun ini maka diperlukan beberapa unit komputer, dimana semua komputer tersebut saling berhubungan dan melakukan komunikasi data agar proses komunikasi data dapat berjalan lancar dengan baik melalui media atau perantara yang disebut dengan jaringan komputer. Jaringan komputer adalah interkoneksi antara 2 atau lebih komputer yang saling terhubung dengan media transmisi kabel (*wireless*). Setiap komputer yang terhubung dengan jaringan disebut node. Sistem koneksi antar node yaitu,

1. Peer to peer

   Peer to peer adalah suatu model dimana setiap PC dapat memakai perangkat pada PC lain atau memberikan perangkatnya untuk digunakan oleh PC lain. Dengan kata lain dapat berfungsi sebagai *client* maupun

server pada periode yang sama. Metode peer to peer pada sistem *windows* lebih dikenal sebagai *workgroup*. Dimana setiap komputer dalam satu jaringan dikelompokan dalam suatu kelompok kerja.

2. Client - Server

   Client Server dapat diterapkan di jaringan lokal maupun dengan teknologi internet. Dimana ada satu unit yang berfungsi sebagai Server dan yang lainnya berfungsi sebagai *client*. Pada sistem koneksi Client Server ini, *Client* hanya dapat mengakses sesuai otorisasi yang disediakan oleh administrator.

2.7.1 Jenis-jenis Jaringan

Secara umum jaringan komputer dapat dibedekan lima jenis yang terdiri dari :

1. LAN (Local Area Network)

   Local Area Network (LAN) merupakan jaringan milik pribadi di dalam sebuah gedung atau kampus yang berukuran sampai beberapa kilometer. LAN seringkali digunakan untuk menghubungkan komputer-komputer pribadi dan *workstation* dalam kantor perusahaan atau pabrik-pabrik untuk memakai bersama *resource* (misalnya, printer, scanner) dan saling bertukar informasi. LAN dapat dibedakan dari jenis jaringan lainnya berdasarkan tiga karakteristik: ukuran, teknologi transmisi dan topologinya.

   LAN mempunyai ukuran yang terbatas, yang berarti bahwa waktu transmisi pada keadaan terburuknya terbatas dan dapat diketahui

sebelumnya. Dengan mengetahui keterbatasnnya, menyebabkan adanya kemungkinan untuk menggunakan jenis desain tertentu. Hal ini juga memudahkan manajemen jaringan.

LAN seringkali menggunakan teknologi transmisi kabel tunggal. LAN tradisional beroperasi pada kecepatan mulai 10 sampai 100 Mbps (mega bit/detik) dengan delay rendah (puluhan *mikro second*) dan mempunyai faktor kesalahan yang kecil. LAN-LAN modern dapat beroperasi pada kecepatan yang lebih tinggi, sampai ratusan megabit/detik.

2. MAN (Metropolitan Area Network)

   Metropolitan Area Network (MAN) pada dasarnya merupakan versi LAN yang berukuran lebih besar dan biasanya memakai teknologi yang sama dengan LAN. MAN dapat mencakup kantor-kantor perusahaan yang berdekatan dan dapat dimanfaatkan untuk keperluan pribadi (swasta) atau umum. MAN biasanya mampu menunjang data dan suara, dan bahkan dapat berhubungan dengan jaringan televisi kabel. MAN hanya memiliki sebuah atau dua buah kabel dan tidak mempunyai *elemen switching*, yang berfungsi untuk mengatur paket melalui beberapa *output* kabel. Adanya *elemen switching* membuat rancangan menjadi lebih sederhana.

   Alasan utama memisahkan MAN sebagai kategori khusus adalah telah ditentukannya standart untuk MAN, dan standart ini sekarang sedang diimplementasikan. Standart tersebut disebut DQDB (Distributed Queue Dual Bus) atau 802.6 menurut standart IEEE. DQDB terdiri dari dua buah kabel *unidirectional* dimana semua komputer dihubungkan. Setiap bus

mempunyai sebuah *head-end*, perangkat untuk memulai aktivitas transmisi. Lalulintas yang menuju komputer yang berada di sebelah kanan pengirim menggunakan bus bagian atas. Lalulintas ke arah kiri menggunakan bus yang berada di bawah.

3. WAN (Wide Area Network)

   Wide Area Network (WAN) mencakup daerah geografis yang luas, sering kali mencakup sebuah negara atau benua. WAN terdiri dari kumpulan mesin yang bertujuan untuk mejalankan program-program aplikasi. Kita akan mengikuti penggunaan tradisional dan menyebut mesin-mesin ini sebagai *host*. Istilah *End System* kadang-kadang juga digunakan dalam literatur. *Host* dihubungkan dengan sebuah subnet komunikasi, atau cukup disebut subnet. Tugas subnet adalah membawa pesan dari *host* ke *host* lainnya, seperti halnya sistem telepon yang membawa isi pembicaraan dari pembicara ke pendengar. Dengan memisahkan aspek komunikasi murni sebuah jaringan (subnet) dari aspek-aspek aplikasi (*host*), rancangan jaringan lengkap menjadi jauh lebih sederhana.

   Pada sebagian besar WAN, subnet terdiri dari dua komponen, yaitu kabel transmisi dan elemen *switching*. Kabel transmisi (disebut juga sirkuit, *channel*, atau *trunk*) memindahkan bit-bit dari satu mesin ke mesin lainnya.

   Element switching adalah komputer khusus yang dipakai untuk menghubungkan dua kabel transmisi atau lebih. Saat data sampai ke kabel penerima, *element switching* harus memilih kabel pengirim untuk

meneruskan pesan-pesan tersebut. Sayangnya tidak ada terminologi standart dalam menamakan komputer seperti ini. Namanya sangat bervariasi disebut paket *switching node*, *intermidiate system*, data *switching exchange* dan sebagainya.

4. Jaringan Tanpa Kabel

   Jaringan tanpa kabel mempunyai berbagai manfaat, yang telah umum dikenal adalah kantor *portable*. Orang yang sedang dalam perjalanan seringkali ingin menggunakan peralatan elektronik *portable*-nya untuk mengirim atau menerima telepon, fax, e-mail, membaca fail jarak jauh login ke mesin jarak jauh, dan sebagainya dan juga ingin melakukan hal-hal tersebut dimana saja, darat, laut, udara.

5. Internet

   Orang yang terhubung ke jaringan sering berharap untuk bisa berkomunikasi dengan orang lain yang terhubung ke jaringan lainnya. Keinginan seperti ini memerlukan hubungan antar jaringan yang sering kali tidak kompatibel dan berbeda. Biasanya untuk melakukan hal ini diperlukan sebuah mesin yang disebut *gateway* guna melakukan hubungan dan melaksanakan terjemahan yang diperlukan, baik perangkat keras maupun perangkat lunaknya. Kumpulan jaringan yang terinterkoneksi inilah yang disebut dengan internet.

   (Sumber: http://itartikel.blogspot.com/2012/04/jenis-jenis-jaringan-komputer_13.html/ 22 juni 2012)

2.7.2 Topologi Jaringan

1. Topologi Star / Star Network

   Beberapa node dihubungkan dengan suatu *node* pusat (*central node* atau *host node*) yang membentuk jaringan seperti bentuk bintang (*star*). Semua komunikasi ditangani dan diatur langsung oleh *central node*. *Central node* melakukan semua tanggung jawab untuk mengatur arus informasi diantara *node* yang lainnya.

2. Topologi Pohon / Hierarchical Tree Network

   Network ini berbentuk seperti pohon yang bercabang, yang terdiri dari *central node* dihubungkan dengan *node* yang lain secara berjanjang. *Cenral node* biasanya berupa *large computer* atau *main frame computer* sebagai *host computer* yang merupakan jenjang tertinggi (*top hierarchical*) yang bertugas mengkoordinasi atau mengendalikan *node* jenjang dibawahnya yang dapat berupa *mini computer* atau *micro computer*.

3. Topologi Mesh / Mesh Network

   Topologi mesh memiliki hubungan yang berlebihan antara peralatan-peralatan yang ada. Susunannya pada setiap peralatan yang ada didalam jaringan saling terhubung satu sama lain.

4. Topologi Bus / Bus Network

   Bentuk ini mengubungkan beberapa *node* dalam jalur data (*bus*). Masing-masing *node* dapat melakukan tugas-tugas operasi yang berbeda-beda.Seperti halnya pada *loop network*, tidak ada *central node* dan semua *node* mempunyai

status yang sama. Karena jalur data (*bus*) terbuka maka awal dan akhir bus harus diberi terminator, sebagai penutup jalur data.

5. Topologi Cincin / Ring Network

   Bentuk ini merupakan gabungan bentuk *loop network* dan *bus network*. Jika salah satu *node* yang lain tidak berfungsi atau rusak, maka tidak akan mempengaruhi komunkasi *node* yang lain karena tepisah dari jalur data. Hal ini berbeda dengan *loop network,* bila salah satu *node* rusak, maka akan mempengaruhi *node* yang lainnya.

2.8 Perangkat Lunak Pendukung

Alat Bantu Pendukung yang digunakan adalah NetBeans IDE 6.9, PhpMyAdmin dan MySQL sebagai database, XAMPP, Star UML, IReport, Install Creator dan menggunakan Bahasa Pemrograman Java to Enterprise Edition (J2EE).

Dibawah ini merupakan penjelasan dari masing-masing perangkat lunak :

a. Java

   Java dikembangkan oleh perusahaan *Sun Microsystem*. Java menurut definisi dari *Sun Microsystem* adalah nama untuk sekumpulan teknologi untuk membuat dan menjalankan perangkat lunak pada komputer *standalone* ataupun pada lingkungan jaringan. Java merupakan bahasa pemograman objek murni karena semua kode programnya dihubungkan dalam kelas.

   (Sumber : Rosa a.s dan Shalahuddin 2011:85)

   Java memiliki karakteristik sebagai berikut :

1. Sederhana

   Bahasa pemograman Java menggunakan sintaks mirip C++, namun sintak pada java telah banyak diperbaiki terutama menghilangkan penggunaan *pointer* yang rumit dan *multiple*. *Inheritance* Java juga menggunakan *automatic memory allocation* dan *memory garbage collection.*

2. Berorientasi objek (Object Oriented)

   Java menggunakan pemograman berorientasi objek yang membuat program dapat dibuat secara modular dan dapat dipergunakan kembali. Pemograman berorientasi objek memodelkan dunia nyata ke dalam objek dan melakukan interaksi antara objek-objek tersebut.

3. Dapat didistribusi dengan mudah

   Java dibuat untuk membuat aplikasi terdistribusi secara mudah dengan adanya *libraries networking* yang terintegrasi dengan Java.

4. Interpreter

   Program Java dijalankan dengan interpreter, yaitu Java Virtual Machine (JVM). Hal ini menyebabkan *source code* Java yang telah dikompilasi menjadi Java *bytecodes* dapat dijalankan pada *platform* yang berbeda-beda.

5. Robust

   Java mempunyai realibilitas yang tinggi. *Compiler* pada java mempunyai kemampuan mendeteksi *error* secara lebih teliti dibandingkan bahasa pemograman yang lain. Java mempunyai *runtime-Exception handling* untuk membantu mengalami *error* pada pemograman.

6. Aman

   Sebagai bahasa pemograman untuk aplikasi internet dan terdistribusi, Java memiliki beberapa mekanisme keamanan untuk menjaga aplikasi tidak digunakan untuk merusak sistem komputer yang menjalankan aplikasi tersebut.

7. Architecture Neutral

   Program java merupakan *platform Independent*. Program cukup mempunyai satu buah versi yang dapat dijalankan pada *platform* yang berbeda dengan Java Virtual Machine.

8. Portabel

   Source code maupun program Java dapat dengan mudah dibawa ke *platform* yang berbeda-beda tanpa harus dikomplikasi ulang.

9. Performance

   Performace pada Java sering dikatakan kurang tinggi. Namun *performance* java dapat ditingkatkan menggunakan kompilasi Java lain seperti buatan Inprise, Microsoft, ataupun Symantec yang menggunakan Just In Time (JIT).

10. Multithreaded

    Java mempunyai kemampuan untuk membuat suatu program yang dapat melakukan beberapa pekerjaan secara sekaligus dan simultan.

11. Dinamis

    Java di desain untuk dapat dijalankan pada lingkungan yang dinamis. Perubahan pada suatu class dengan menambahkan properties ataupun

method dapat dilakukan tanpa menganggu program yang menggunakan class tersebut.

(Sumber : Th. Ari Prabawati 2010:76)

b. Netbeans

Netbeans merupakan salah satu IDE yang digunakan untuk melakukan pemograman baik menulis kode, mengompilasi, mencari kesalahan, dan mendistribusikan program. Netbeans juga menyediakan paket yang lengkap dalam pemograman dari pemograman standar (aplikasi desktop), pemograman *enterprise*, dan pemograman perangkat *mobile*.

(Sumber: Th. Ari Prabawati 2010:4).

c. XAMPP

XAMPP adalah paket instalasi mudah Apache server yang sudah termasuk di dalamnya adalah Apache Server, MySQL dan PHP. XAMPP merupakan sebuah *tool* yang menyediakan beberapa paket perangkat lunak ke dalam satu buah paket. Dengan menginstall XAMPP kita tidak perlu lagi melakukan instalasi dan konfigurasi webserver Apache, PHP dan MySQL secara manual. XAMPP akan menginstalasi dan mengkonfigurasikannya secara otomatis.

d. Star UML

StarUML (SU) adalah alat untuk membuat diagram UML kelas dan secara otomatis menghasilkan Java "kode rintisan". SU juga dapat balik Java *source*/kode *byte* untuk menghasilkan UML diagram yang sesuai.

SU akan menghasilkan kode yang mencerminkan struktur kelas, tetapi bukan tindakan spesifik pada setiap objek. Untuk itu, setelah membuat diagram

menggunakan SU, *user* mengedit kode rintisan yang dihasilkan untuk menambahkan sisa fungsi untuk kode, mengisi setiap metode apa yang harus dilakukan.

(sumber : http://cnx.org/content/m15092/latest/starUML-tutorial/20 juni 2012)

e. PhpMyAdmin

PhpMyAdmin merupakan aplikasi berbasiskan web yang dikembangkan menggunakan bahasa pemograman PHP. Melalui PhpMyAdmin, *user* dapat melakukan *query* tanpa harus mengetikkan seperti pada MS DOS.

(Sumber : Sugiri, Haris Saputro, Pengelolaan Database MySQL dengan PhpMyAdmin 2008:4)

f. MySQL

Menurut Sugiri, Haris Saputro (2008:1) mendefinisikan MySQL merupakan database yang bersift client server, dimana data diletakan di server yang bisa diakses melalui client. Pengaksesan dapat dilakukan apabila komputer telah terhubung dengan server.

MySQL merupakan salah satu perangkat lunak sistem pengelola basis data DBMS (Data Base Management System). MySQL merupakan sebuah hubungan Data Base Management System (DBMS) yang membantu sebuah model data yang terdiri atas kumpulan hubungan nama (*named relation*). Database MySQL adalah salah satu database yang open source. MySQL juga merupakan server *multithreaded*, sehingga memungkinkan *daemon* untuk meng-*handle* permintaan layanan secara simultan. Penyebab utama MySQL begitu populer dan digunakan oleh banyak orang diantaranya ialah :

1. MySQL tersedia di berbagai platform dan kompatibel dalam berbagai sistem operasi.
2. MySQL dapat melakukan koneksi dengan *client* menggunakan *protocol* TCP/IP, Unix socket (Unix), atau Named Pipes (NT).
3. MySQL memiliki *overhead* koneksi yang rendah, sehingga kecepatan koneksi relatif tinggi.
4. MySQL dapat menangani database dengan skala besar, dengan jumlah *record* lebih dari 50 juta, 60 ribu tabel, dan bisa menampung 5 miliyar baris data. Selain itu, pada MySQL versi 4.1.2 , batas indeks pada tiap tabel dapat menampung sampai 64 index.

Security database MySQL terkenal baik, karena memiliki lapisan sekuritas seperti level *subnetmask*, nama *host* dan izin akses *user* dengan sistem perizinan khusus serta *password* yang dimiliki setiap *user* dalam bentuk data terenkripsi.

g. IReport

Ireport merupakan perangkat lunak bantu untuk perancangan laporan secara visual yang nantinya dapat di kompilasi dengan menggunakan JasperReport sehingga menjadi file *.jasper atau *.jrxml yang dapat langsung dipanggil oleh program Java.IReport merupakan salah satu *tool* yang sangat *powerfull* dalam pembuatan sebuah laporan pada bahasa pemrograman java, selain itu ireport juga merupakan *software* yang dikeluarkan dengan lisensi *open source*. IReport sendiri merupakan sebuah aplikasi bisa juga diinstall sebagai plug-in di netbean. IReport ini menggunakan JasperReport sebagai library-nya.

(sumber : http://inferno-consulting.blogspot.com /22 juni 2012)

h. Install Creator

Install Creator merupakan sebuah program yang bisa digunakan untuk membuat Installer. Dengan menggunakan *software* ini installer yang dihasilkanpun lebih kecil dari file yang didalamnya. Install Creator *interfacenya* simpel, dan juga mudah digunakan. Saat menjalankan program ini, langsung ada Wizard yang langsung menuntun untuk tahapan membuat Installer. Installernya pun dapat diberi *password*. Install Creator sangat berguna untuk membuat Installer program dengan cepat mudah dan singkat.

Install Creator didalamnya dilengkapi dengan fitur-fitur yang dapat dibutuhkan, termasuk instalasi otomatisasi dari beberapa *screensaver*, *font* atau *activeX* yang mengontrol aplikasi yang dibuat dapat membutuhkannya. Install Creator Pro sangat berguna bagi para *Developer* yang ingin membuat Installer programnya dengan cepat mudah dan singkat

(sumber : http://kekeongeun.blogspot.com/ - 22 juni 2012)

2.9 Pengujian Software

Pengujian *Software* adalah cara atau teknik untuk menguji perangkat lunak, mempunyai mekanisme untuk menentukan data uji yang dapat menguji perangkat lunak secara lengkap dan mempunyai kemungkinan tinggi untuk menemukan kesalahan.

Beberapa teknik pengujian yang dapat digunakan untuk menguji perangkat lunak, diantaranya: Pengujian Black Box dan Pengujian White Box.

1. Pengujian Black Box
    a. Kesesuaian pengujian perangkat lunak hanya dilihat berdasarkan keluaran yang dihasilkan dari data atau kondisi masukan yang diberikan untuk fungsi yang ada tanpa melihat bagaimana proses untuk mendapatkan keluaran tersebut.
    b. Digunakan untuk menguji fungsi-fungsi khusus dari perangkat lunak yang dirancang.
    c. Dari keluaran yang dihasilkan, kemampuan program dalam memenuhi kebutuhan pemakai dapat diukur sekaligus dapat diiketahui kesalahan-kesalahannya.

Beberapa jenis kesalahan yang dapat diidentifikasi :

1. Fungsi tidak benar atau hilang
2. Kesalahan antar muka
3. Kesalahan pada struktur data (pengaksesan basis data)
4. Kesalahan inisialisasi dan akhir program
5. Kesalahan performasi.

2. Pengujian White Box
    a. Digunakan untuk mengetahui cara kerja suatu perangkat lunak secara internal yaitu dengan mengikuti struktur logika dari koding.
    b. Pengujian dilakukan untuk menjamin operasi-operasi internal sesuai dengan spesifikasi yang telah ditetapkan dengan menggunakan struktur kendali dari prosedur yang dirancang.

Pelaksanaan pengujian white box

a. Menjalani *logical decision* pada sisi dan *false*
b. Mengeksekusi pengulangan (*looping*) dalam batas-batas yang ditentukan
c. Menguji struktur data internal