BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

Sistem adalah rangkaian dari dua atau lebih komponen yang saling berhubungan, yang berinteraksi untuk mencapai suatu tujuan. Sistem hampir selalu terdiri dari beberapa subsistem kecil, yang masing-masing melakukan fungsi khusus yang penting dan mendukung bagi sistem yang lebih besar Marshal dan Paul John (2005:2-3).

2.1.1 Elemen Sistem

Ada beberapa elemen yang membentuk sebuah sistem, yaitu: tujuan, masukan, proses, keluaran, batas, mekanisme pengendalian dan umpan balik serta lingkungan. Elemen-elemen yang membentuk sebuah sistem adalah tujuan, masukan, proses, keluaran, batas, mekanisme pengendalian dan umpan balik, dan lingkungan.

Sebuah sistem tidak memiliki kombinasi elemen yang sama tetapi memiliki susunan dasar yang sama yang terdiri dari (*Input, Output,* Transformasi, Mekanisme Pengendalian dan Tujuan).Raymond McLeod (2004: 9).

Setiap sistem memiliki tujuan, tujuan inilah yang menjadi pemotivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tak terarah dan tak terkendali. Tentu saja, tujuan antara satu sistem dengan sistem yang lain berbeda.

Masukan sistem adalah segala sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan yang diproses. Masukan dapat berupa hal-hal yang berwujud maupun yang tidak tampak.

Proses merupakan bagian yang melakukan perubahan atau transformasi dari masukan menjadi keluaran yang berguna dan lebih bernilai, misalnya berupa informasi dan produk, tetapi juga bisa berupa hal-hal yang tidak berguna.

Keluaran (*output*) merupakan hasil dari pemrosesan. Pada sistem informasi, keluaran bisa berupa suatu informasi, saran, cetakan laporan, dan sebagainya.

Batas sistem adalah pemisah antara sistem dan daerah di luar sistem (lingkungan). Batas sistem menentukan konfigurasi, ruang lingkup, atau kemampuan sistem.

Mekanisme pengendalian diwujudkan dengan menggunakan umpan balik, yang mencuplik keluaran. Umpan balik ini digunakan untuk mengendalikan baik masukan maupun proses. Tujuannya adalah untuk mengatur agar sistem berjalan sesuai dengan tujuan.

Lingkungan adalah segala sesuatu yang berada diluar sistem. Lingkungan bisa berpengaruh terhadap operasi sistem dalam arti bisa merugikan atau menguntungkan sistem itu sendiri.

2.1.2 Karakteristik Sistem

Suatu sistem mempunyai karakteristik atau sifat-sifat tertentu yaitu :

1. Bagian (*Component*), Suatu sistem terdiri dari sejumlah komponen yang saling berinteraksi yang artinya bekerja sama membentuk satu kesatuan komponen-komponen sistem atau elemen-elemen sistem dapat berupa suatu sub sistem atau bagian dari sistem.

2. Batas Sistem (*Boundry*), Merupakan daerah yang membatasi antara suatu sistem dengan sistem yang lainnya atau dengan lingkungan luar.

3. Lingkungan Luar Sistem (*Environment*), Lingkungan luar dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi operasi sistem.

4. Penghubung Sistem (*Interface*), Penghubung (*interface*) merupakan media penghubung antara satu subsistem dengan subsistem yang lainnya. Melalui penghubung ini memungkinkan sumber daya mengalir dari suatu subsistem akan menjadi masukan untuk subsistem yang lainnya melalui penghubung ini, dengan begitu suatu sistem dapat berintegrasi dengan subsistem yang lainnya dengan membentuk suatu kesatuan.

5. Masukkan Sistem (*Input*), Energi yang dimasukkan ke dalam sistem. Masukkan dapat berupa maintenance input dan signal input. Maintenance

input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. Signal input adalah energi yang diproses untuk diperoleh keluarannya.

6. Keluaran Sistem (*Output*), Hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan keluaran dapat merupakan masukkan untuk sistem yang lain atau supra sistem.

7. Pengolah Sistem (*Process*), Suatu sistem dapat mempunyai suatu bagian pengolah yang akan merubah masukkan menjadi keluaran. Suatu sistem pengolahan akan mengolah berupa bahan baku dalam hal ini adalah data serta bahan-bahan lainnya untuk menghasilkan keluaran berupa laporan dan informasi yang berguna.

8. Sasaran Sistem (*Objective*), Suatu sistem pasti mempunyai tujuan (goal) atau sasaran (*objective*). Kalau suatu sistem tidak mempunyai sasaran maka operasi tidak akan ada gunanya. Sasaran dari sistem sangat menentukan sekali masukkan yang dibutuhkan dan keluaran sistem yang akan dihasilkan oleh sistem itu sendiri.

2.1.3 Klasifikasi Sistem

Sistem mempunyai beberapa klasifikasi yang perlu kita ketahui agar dapat mendifinisikan sebuah sistem tersebut.

Sistem abstrak dan fisik.

1. Sistem Abstrak : Sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik.

2. Sistem Fisik : Sistem yang ada secara fisik.

Sistem alamiah dan buatan manusia.

A. Sistem Alamiah : Sistem yang terjadi melalui proses alam dan tidak dibuat oleh manusia.

B. Sistem Buatan Manusia : Sistem yang dirancang oleh manusia dan melibatkan interaksi antara manusia dan mesin.

Sistem tertentu dan tak tentu.

1. Sistem Tertentu : Beroperasi degan tingkah laku yang sudah dapat diprediksi,interaksi bagian-bagiannya dapat dideteksi dengan pasti sehingga keluarannya dapat diramalkan.

2. Sistem Tak Tentu : Sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

Sistem tertutup dan terbuka.

1. Sistem Tertutup : Sistem yang berhubungan dan tidak terpengaruh dengan lingkungan luarnya.

2. Sistem Terbuka : Sistem yang berhubungan dan terpengaruh dengan lingkungan luarnya.

2.2 Pengertian Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan bermanfaat bagi yang menerimanya. Siklus Informasi adalah data yang masih merupakan bahan mentah yang harus di olah untuk menghasilkan informasi melalui suatu model.

Menurut Jogiyanto HM., MBA., Akt., Ph.D.**(**2005 : 8**)** bahwa informasi adalah Data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang meneriman.

Kualitas Informasi tergantung pada 3 hal, yaitu informasi harus :

1. Akurat, berarti informasi harus bebas dari kesalahan-kesalahan dan tidak bias atau menyesatkan. Akurat juga berarti informasi harus jelas mencerminkan masudnya. Informasi harus akurat karena dari sumber informasi sampai ke penerima informasi kemungkinan banyak terjadi gangguan *(noise)*
2. Tetap pada waktunya, berarti informasi yang datang pada penerima tidak boleh terlambat.

3. Relevan, berarti informasi tersebut menpunyai manfaat untuk pemakainya. Relevansi informasi untuk tiap-tiap orang satu dengan yang lainnya berbeda.

Nilai Informasi ditentukan dari dua hal, yaitu :

1. Manfaat dan biaya mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
2. Pengukuran nilai informasi biasanya dihubungkan dengan analisis *cost effectiveness* atau *cost benefit.*

Informasi bisa juga diartikan sebagai pengetahuan yang didapatkan dari pembelajaran, pengalaman, atau instruksi. Namun demikian, istilah ini memiliki banyak arti bergantung pada konteksnya, dan secara umum berhubungan erat dengan konsep seperti arti, pengetahuan, *negentropy*, komunikasi, kebenaran, representasi, dan rangsangan mental.

Informasi yang berupa koleksi data dan fakta seringkali dinamakan informasi statistik. Dalam bidang ilmu komputer, informasi adalah data yang disimpan, diproses, atau ditransmisikan. Penelitian ini memfokuskan pada definisi informasi sebagai pengetahuan yang didapatkan dari pembelajaran, pengalaman, atau instruksi dan alirannya.

2.3 Pengertian Sistem Informasi

Sistem informasi adalah sekumpulan komponen pembentuk sistem yang mempunyai keterkaitan antara satu komponen dengan komponen lainnya yang bertujuan menghasilkan suatu informasi dalam suatu bidang tertentu. Dalam sistem informasi diperlukannya klasifikasi alur informasi, hal ini disebabkan keanekaragaman kebutuhan akan suatu informasi oleh pengguna informasi. Kriteria dari sistem informasi antara lain, fleksibel, efektif dan efisien..

Menurut Jogiyanto HM., MBA., Akt., Ph.D. (2005 : 11) Suatu sistem terintegrasi yang mampu menyediakan informasi yang bermanfaat bagi penggunanya, atau sebuah sistem terintegrasi atau sistem manusia-mesin, untuk menyediakan informasi untuk mendukungoperasi, manajemen dalam suatu organisasi. Sistem ini memanfaatkan perangkat keras dan perangkat lunak komputer, prosedur manual, model manajemen dan basis data.

2.3.1 Komponen Sistem Informasi

Sistem informasi terdiri dari komponen-komponen yang disebut blok bangunan, yang terdiri dari komponen *input*, komponen model, komponen *output*, komponen teknologi, komponen *hardware*, komponen *software*, komponen basis data, dan komponen kontrol. Semua komponen tersebut saling berinteraksi satu dengan yang lain membentuk suatu kesatuan untuk mencapai sasaran.

1. Komponen *input*

Input mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode dan media untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dokumen dasar.

2. Komponen model

Komponen ini terdiri dari kombinasi prosedur, logika, dan model matematika yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah ditentukan untuk menghasilkan keluaran yang diinginkan.

3. Komponen *output*

Hasil dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua pemakai sistem.

4. Komponen teknologi

Teknologi merupakan *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 keseluruhan.

5. Komponen *hardware*

Hardware berperan penting sebagai suatu media penyimpanan vital bagi sistem informasi. Yang berfungsi sebagai tempat untuk menampung database atau lebih mudah dikatakan sebagai sumber data dan informasi untuk memperlancar dan mempermudah kerja dari sistem informasi.

6. Komponen *software*

Software berfungsi sebagai tempat untuk mengolah, menghitung dan memanipulasi data yang diambil dari *hardware* untuk menciptakan suatu informasi.

7. Komponen basis data

Basis data (*database*) merupakan kumpulan data yang saling berkaitan dan berhubungan satu dengan yang lain, tersimpan di pernagkat keras komputer dan menggunakan perangkat lunak untuk memanipulasinya. Data perlu disimpan dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa supaya informasi yang dihasilkan berkualitas. Organisasi basis data yang baik juga berguna untuk efisiensi kapasitas penyimpanannya. Basis data diakses atau dimanipulasi menggunakan perangkat lunak paket yang disebut DBMS (*Database Management System*).

8. Komponen kontrol

Banyak hal yang dapat merusak sistem informasi, seperti bencana alam, api, temperatur, air, debu, kecurangan-kecurangan, kegagalan-kegagalan sistem itu sendiri, ketidak efisienan, sabotase dan lain sebagainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

2.3.2 Perancangan Sistem Informasi

Menurut Jogiyanto HM., MBA., Akt., Ph.D. (2005 : 23) perancangan sistem adalah tahap yang dilakukan setelah melakukan analisis sistem, pendefinisian kebutuhan–kebutuhan sistem yang akan dibangun, dan persiapan untuk merancang bangun implementasi sistem dengan menggambarkan sistem yang akan dibangun.

Perancangan sistem dimulai dengan memahami sistem yang sedang berjalan dan kriteria–kriteria sistem yang akan dibangun biasanya menggunakan pemodelan secara terstruktur yang digambarkan oleh grafik atau diagram.

Hal–hal yang harus diperhatikan dalam merancang dan membangun sebuah sistem menurut Jogiyanto Jogiyanto HM., MBA., Akt., Ph.D. (2005 : 25) adalah :

1. Kebutuhan perusahaan, organisasi, atau lembaga, yaitu dengan cara memahami bidang yang akan dikembangkan, sasaran yang dibidik serta media yang akan digunakan.
2. Kebutuhan operator, yaitu kebutuhan operator untuk memperoleh sistem yang mudah dipahami dan dioperasikan serta tampilan yang interaktif. Kebutuhan operator dapat diperoleh melalui proses wawancara atau kuisioner.
3. Kebutuhan pemakai, yaitu keinginan – keinginan dari si pemakai sistem informasi, seperti jaminan keamanan, standarisasi tampilan, kecepatan akses, dan kemudahan dalam pengoperasian.
4. Kebutuhan teknis, yaitu meliputi arsitektur dan konfigurasi sistem. Secara teknis peralatan dan teknologi yang digunakan, termasuk pertimbangan penggunaan peralatan yang tidak standar, seperti peralatan multimedia, kebutuhan *interface, database*, dan perangkat lunaknya.

2.4 Metodologi Berbasis Objek

2.4.1 Sejarah Metode Berorientasi Objek

Metode beorientasi objek mulai berkembang ketika Grady Booch pada tahun 80an mempublikasikan suatu jurnal bagaimana melakukan perancangan untuk bahasa *ADA* namun memberi judul jurnal tersebut dengan : *Object-Oriented Design*. Selanjutnya ide tersebut terus ia kembangkan sampai tahun 90an.

Pada tahun 1991 Peter Coad dan Yourdon memperkenalkan metode berorientasi objek yang lebihsederhana dibandingkan Booch. Metode ini menjadi cepat populer karena mendukung layanan-layanan yang terdapat pada C++. Pada waktu itu C++ merupakan bahasa pemrograman berorientasi objek yang paling populer .

Pada tahun 1991 juga Rumbaugh memperkenalkan *Object Modelling Technique* (OMT). Pendekatan yang digunakan tidak jauh berbeda dengan pendekatan yang digunakan Coad Yourdon namun dengan notasi yang berbeda. OMT tidak hanya sepenuhnya berbasis pada datadriven tapi juga memisahkan proses dari data dengan penggunan *data flow diagram* yang terpisah dengan diagram kelas. OMT juga menggunakan notasi *state transition diagram* untuk memodelkan aspek dinamis sistem.

Pada tahun bulan Oktober 1994 Booch, Rumbaugh dan Jacobson, mempelopori usaha untuk penyatuan notasi pendekatan berorientasi objek. Pada tahun 1995 dihasilkan draftpertama dari UML (versi 0.8). Sejak tahun 1996 pengembangan tersebut dikoordinasikan oleh Object Management.

Tahun 1997 UML versi 1.1 muncul, dan saat ini versi terbaru adalah versi 1.5 yang dirilis bulan Maret2003. Booch, Rumbaugh dan Jacobson menyusun tiga buku serial tentang UML pada tahun 1999.Sejak saat itulah UML telah menjelma menjadi standar bahasa pemodelan untuk aplikasi berorientasi objek .

2.4.2 Pengertian *Object Oriented Analysis and Design*

Object-Oriented Analysis and Design adalah metode analisis yang memerikasa kebutuhan dari sudut pandang kelas kelas dan objek yang ditemui dalam ruang lingkup permasalahan yang mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek system atau subsistem. OOAD merupakan cara baru dalam memikirkan suatu masalah dengan menggunakan model yang dibuat menurut konsep sekitar dunia nyata. Dasar pembuatan adalah objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas.

Menurut Mathiassen *Object-Oriented Analysis and Design* (OOAD) adalah metode untuk menganalisa dan merancang sistem dengan pendekatan berorientasi objek. *Object* diartikan sebagai suatu entitas yang memiliki identitas, *state*, dan *behavior*.

OOAD mencakup analisis dan desain sebuah sistem dengan pendekatan objek, yaitu analisis berorientasi objek (OOA) dan desain berorientasi objek (OOD). OOA adalah metode analisis yang memerika syarat / keperluan yang harus dipenuhi sebuah sistem dari sudut pandang kelas-kelas dan objek-objek yang ditemui dalam ruang lingkup perusahaan. Sedangkan OOD adalah metode untuk mengarahkan arsitektur *software* yang didasarkan pada manipulasi objek-objek sistem atau subsistem.

2.4.3 Kelebihan *Object Oriented Analysisand Design*

Keuntungan dari OOAD menurut Mathiassen (2000:5) adalah :

1. Menurut konsep umum yang dapat digunakan untuk memodelkan hampir semua fenomena dan dapat dinyatakan dalam bahasa umum (*natural language*)
2. Memberikan informasi yang jelas tentang konteks dari sistem
3. Mengurangi biaya *maintainance*
4. Memudahkan untuk mencari hal yang akan diubah
5. Membuat perubahan menjadi lokal tidak berpengaruh pada modul yang lain.

2.4.4 Karakteristik *Object Oriented Analysis and Design*

1. Objek

   Objek adalah suatu kombinasi antara data dan aspek logik yang merepresentasikan suatu entitas dunia nyata. Contoh : mobil toyota kijang adalah suatu objek, maka yang menjadi data adalah warna, pembuat, harga, konsumsi bahan bakar, mekanisme pengereman, dll. Sedangkan yangmenjadi aspek lojiknya adalah menampilkan jarak tempuh, berhenti, bejalan, belok, dll.

2. Kelas

   Pada sistem berorientasi objek, kelas didefinisikan sebagai himpunan objek yang memilikistruktur umum dan perilaku umum yang sama. Sedangkan objek dapat dikatakansebagai instansiasi suatu

kelas. Sebagai contoh mobil adalah suatu kelas, maka mobil Toyota kijang, mobil Proton, dan lain lain adalah objek.

3. Atribut

Atribut merepresentasikan karakteristik atau keadaan objek. Pada contoh kasus di atas, sebuah mobil dapat memiliki atribut warna, harga, dan pembuat. Pada tataran implementasi, warna dapat direpresentasikan sebagai suatu *string* (domain nilainya misalnya : merah, biru, kuning, dan lain lain). Harga dapat berupa bilangan *floating point* atau bilangan *integer*. Sedangkan pembuat dapat bertipe struktur yang terdiri dari nama, identitas korporat, dan lain-lain.

4. Metode

Metode adalah suatu fungsi atau prosedur yang didefinisikan untuk dapat mengakses keadaan internal suatu objek dari suatu kelas. Tiap fungsi atau prosedur mendefinisikan dan mendeskripsikan perilaku khusus suatu objek. Sebagai contoh: kelas Pegawai memiliki metode Hitung Gaji. Metode sebenarnya merupakan antarmuka yang disediakan untuk dapat memanfaatkan perilaku objek tersebut. Sebagai contoh : jika diinginkan dilakukannya perhitungan gaji, maka *message* ‘Hitung Gaji” harus dikirimkan ke objek Pegawai.

5. Message

Message pada dasarnya adalah pemanggilan fungsi. Namun *message* berbeda dari pemanggilans ubrutin. Dengan *message* yang

sama dua objek berbeda dapat melakukan operasi yang berbeda pula. Konsep ini dikenal sebagai *Polymorphism*.

6. Enkapsulasi

Enkapsulasi memadukan karakteristik unit di dalam suatu objek (data dan metode). Konsep inibertujuan untuk menyembunyikan informasi dan karakteristik objek. Objek dapat dimanfaatkan hanya dengan cara memanggil metode yang dimiliki objek tersebut.

7. Hirarki Kelas

Sistem berorientasi objek mengorganisasi kelas ke dalam hirarki *subclass-superclass*. Perbedaan karakteristik dan perilaku digunakan sebagai dasar penilaian untuk membedakan antara kelas dan subkelas.

8. Pewarisan (*Inheritance*)

Konsep pewarisan memungkinkan suatu objek dibangun dari objek lain. Pada konsep ini akan adakelas yang ertindak sebagai parent class atau dikenal sebagai super kelas. Sedangkan kelas turunannya menjadi subkelas. Kelas yang diturunkan dari kelas lain akan memiliki karakteristik dan perilaku yang dimiliki superkelasnya.

9. Polymorphism

Konsep ini memungkinkan suatu metode yang berada pada beberapa kelas yang berbeda dapatmemiliki perilaku berbeda. Dengan konsep ini memudahkan untuk menulis suatu kode yang memiliki

reusability tinggi. Contoh suatu metode Hitung_Luas akan memiliki perilaku / formula berbeda di tiap-tiap kelas. Pada kelas Bujur Sangkar, maka formula hitung luasnya adalah sisi xsisi. Pada kelas Segi Tiga formula Hitung Luas adalah ½ x alas x tinggi.

2.4.5 Ciri Ciri *Object Oriented Programming*

Object Oriented Programming (OOP) adalah suatu pemprograman yang berorientasi objek. Ciri-ciri atau karakteristik pemrograman berorientasi objek, antara lain :

1. Objek

   Objek adalah elemen dasar dari konsep pemrograman, merupakan sesuatu yang memiliki identitas (nama), pada umumnya juga memiliki data tentang dirinya maupun object lain dan mempunyai kemampuan untuk melakukan sesuatu dan bisa bekerja sama dengan objek lainnya. Pada dasarnya semua benda yang ada di dunia dapat dianggap sebagai sebuah object. Rumah, mobil, sepeda motor, meja, dan komputer merupakan contoh-contoh object yang ada di dunia nyata. Setiap object memiliki dua karakteristik yang utama yaitu, atribut dan behavior. Atribut merupakan status object dan behavior merupakan tingkah laku dari object tersebut.

2. Kelas

   Kelas merupakan suatu hasil pemodelan fakta-fakta dari suatu objek yang berguna bagi suatu aplikasi yang diprogram. Proses pemodelan fakta-fakta tersebut disebut dengan abstraksi.

3. *Method*

   Method merupakan suatu fungsi (sub-routine) yang berhubungan dengan objek.

4. *Field*

   Adalah informasi atau atribut yang terdapat dalam suatu object. Bentuk field mirip dengan variabel yaitu dapat dibaca dan diset langsung.

5. Pewarisan

   Pewarisan atau *inheritance*adalah sebuah kelas mewarisi state dan behaviour dari kelas lain.

6. Variabel

   Variabel bisa diartikan sebagai tempat untuk menampung data sementara.

7. Enkapsulasi

   Enkapsulasi secara definisi adalah mekanisme pembungkusan untuk menyembunyikan atau melindungi suatu proses dari kemungkinan interferensi atau penyalahgunaan dari luar sistem juga sekaligus menyederhanakan penggunaan system itu sendiri.

8. *Polymorphism*

   Konsep inidimungkinkan untuk menggunakan suatu interface yang sama agar suatu objek melakukan aksi atau tindakan yang mungkin secara prinsip sama tapi secara proses bisa berbeda-beda.

2.5 Bahasa UML (*Unified Model Language)*

Unified Modeling Language (UML) merupakan satu kumpulan konvensi pemodelan yang digunakan untuk menentukan atau menggambarkan sebuah sistem software yang terkait dengan objek (Whitten L. Jeffery et al, 2004). Sementara menurut Henderi (2007: 4) Unified Modeling Language (UML) adalah sebuah bahasa pemodelan yang telah menjadi standar dalam industri software untuk visualisasi, merancang, dan mendokumentasikan sistem perangkat lunak. Bahasa Pemodelan UML lebih cocok untuk pembuatan perangkat lunak dalam bahasa pemrograman berorientasi objek, namun demikiantetap dapat digunakan pada bahasa pemrograman prosedural (Ziga Turck, 2007).

Unified Modeling Language (UML) biasa digunakan untuk (Henderi, 2007 :11)

1. Menggambarkan batasan sistem dan fungsi-fungsi sistem secara umum, dibuat dengan *use case* dan *actor*
2. Menggambarkan kegiatan atau proses bisnis yang dilaksanakan secara umum, dibuat dengan *interaction diagrams*

3. Menggambarkan representasi struktur statik sebuah sistem dalam bentuk class diagrams
4. Membuat model *behavior* yang menggambarkan kebiasaan atau sifat sebuah sistem dengan state transition diagrams
5. Menyatakan arsitektur implementasi fisik menggunakan *component and development diagrams*
6. Menyampaikan atau memperluas fungsionality dengan *stereotypes* (Ziga Turck, 2007),

Berikut ini adalah penjelasan mengenai berbagai diagram UML serta tujuannya:

1. *Use Case Diagram*

Use Case Diagram secara grafis menggambarkan interaksi antara sistem, sistem eksternal, dan pengguna. Dengan kata lain *Use Case diagram* secara grafis mendeskripsikan siapa yang akan menggunakan sistem dan dalam cara apa pengguna mengharapkan interaksi dengan sistem itu. *Use Case* secara naratif digunakan untuk secara tekstual menggambarkan sekuensi langkah-langkah dari setiap interaksi.

2. *Class Diagram*

Class Diagram menggambarkan struktur object sistem. Diagram ini menunjukan class object yang menyusun sistem dan juga hubungan antara class object tersebut

3. Sequence Diagram

Sequence Diagram: secara grafis menggambarkan bagaimanaobjek berinteraksi dengan satu sama lain melalui pesan pada sekuensi sebuah use case atau operasi. Diagram ini mengilustrasikan bagaimana pesan terkirim dan diterima di antara objek dan dalam sekuensi atau *timing* apa.

4. *Collaboration Diagram*

Diagram kolaborasi / *Collaboration Diagram*: serupa dengan diagram rangkaian / sekuensi, tetapi tidak fokus pada *timing* atau sekuensi pesan. Diagram ini justru menggambarkan interaksi (atau kolaborasi) antara objek dalam sebuah format jaringan.

Diagram rangkaian maupun diagram kolaborasi merupakan *isomorphic* artinya kita dapat mengubah dari satu diagram ke diagram lain.

5. Activity Diagram

Activity Diagram secara grafis digunakan untuk menggambarkan rangkaian aliran aktivitas baik proses bisnis maupun *use case. Activity diagram* dapat juga digunakan untuk memodelkan action yang akan dilakukan saat sebuah operasi dieksekusi, dan memodelkan hasil dari action tersebut.

6. Component Diagram

Component Diagram digunakan untuk menggambarkan organisasi dan ketergantungan komponen-komponen *software* sistem. Komponen diagram dapat digunakan untuk menunjukan bagaimana kode pemrograman dibagi menjadi modul-modul (atau komponen).

7. Deployment Diagram

Deployment diagram digunakan untuk mendeskripsikan arsitekturs fisik dalam istilah *node* untuk *hardware* dan *software* dalam sistem. Diagram ini menggambarkan konfigurasi komponen-komponen *software real-time*, *processor*, dan peralatan yang membentuk arsitektur sistem.

2.6 Jaringan Komputer

Jaringan Komputer adalah sekelompok komputer otonom yang saling berhubungan antara satu dengan lainnya menggunakan protokol komunikasi melalui media komunikasi sehingga dapat saling berbagi informasi, program – program, penggunaan bersama perangkat keras seperti *printer*, *harddisk*, dan sebagainya. Selain itu jaringan komputer bisa diartikan sebagai kumpulan sejumlah terminal komunikasi yang berada diberbagai lokasi yang terdiri dari lebih satu komputer yang saling berhubungan.

Tujuan dari jaringan komputer adalah agar dapat mencapai tujuannya, setiap bagian dari jaringan komputer dapat meminta dan memberikan layanan. Pihak yang meminta atau menerima layanan disebut klien dan yang memberikan atau mengirim layanan disebut peladen. Disain ini disebut juga dengan sistem *client-server*, dan digunakan pada hampir seluruh aplikasi jaringan komputer.

2.6.1 Klasifikasi Jaringan Komputer

Berdasarkan skala :

1. *Local Area Network* (LAN): suatu jaringan komputer yang menghubungkan suatu komputer dengan komputer lain dengan jarak yang terbatas.
2. *Metropolitant Area Network* (MAN): prinsip sama dengan LAN, hanya saja jaraknya lebih luas, yaitu 10-50 km.
3. *Wide Area Network* (WAN): jaraknya antar kota, negara, dan benua. ini sama dengan internet.

Berdasarkan fungsi : Pada dasarnya setiap jaringan komputer ada yang berfungsi sebagai *client* dan juga *server*. Tetapi ada jaringan yang memiliki komputer yang khusus didedikasikan sebagai *server* sedangkan yang lain sebagai *client*. Ada juga yang tidak memiliki komputer yang khusus berfungsi sebagai *server* saja. Karena itu berdasarkan fungsinya maka ada dua jenis jaringan komputer:

1. Client-server

Yaitu jaringan komputer dengan komputer yang didedikasikan khusus sebagai *server*. Sebuah *service*/layanan bisa diberikan oleh sebuah komputer atau lebih. Contohnya adalah sebuah domain seperti www.detik.com yang dilayani oleh banyak komputer *web server*. Atau bisa juga banyak *service*/layanan yang diberikan oleh satu komputer. Contohnya adalah *server*

yang merupakan satu komputer dengan *multi service* yaitu *mail server, web server, file server, database server* dan lainnya.

2. Topologi Jaringan

Dalam jaringan ada beberapa jenis topologi yang sering digunakan untuk ketikan menghubungkan semua komputer yang ada dalam area jaringan tersebut. Berikut adalah jaringan komputer berdasarkan dari topologinya:

1. Topologi bus

   Topologi bus merupakan topologi yang banyak dipergunakan pada masa penggunaan kabel sepaksi menjamur. Dengan menggunakan *T-Connector*, maka komputer atau perangkat jaringan lainnya bisa dengan mudah dihubungkan satu sama lain.

2. Topologi bintang

   Topologi bintang merupakan bentuk topologi jaringan yang berupa konvergensi dari node tengah ke setiap node atau pengguna. Topologi jaringan bintang termasuk topologi jaringan dengan biaya menengah.

3. Topologi cincin

   Topologi cincin adalah topologi jaringan berbentuk rangkaian titik yang masing-masing terhubung ke dua titik lainnya, sedemikian sehingga membentuk jalur melingkar membentuk cincin. Pada topologi cincin, komunikasi data dapat terganggu jika satu titik mengalami gangguan. Jaringan FDDI mengantisipasi kelemahan ini

dengan mengirim data searah jarum jam dan berlawanan dengan arah jarum jam secara bersamaan.

4. Topologi mesh

   Topologi jala atau Topologi mesh adalah suatu bentuk hubungan antar perangkat dimana setiap perangkat terhubung secara langsung ke perangkat lainnya yang ada di dalam jaringan. Akibatnya, dalam topologi mesh setiap perangkat dapat berkomunikasi langsung dengan perangkat yang dituju.

5. Topologi pohon

   Topologi Pohon adalah kombinasi karakteristik antara topologi bintang dan topologi bus. Topologi ini terdiri atas kumpulan topologi bintang yang dihubungkan dalam satu topologi bus sebagai jalur tulang punggung atau backbone. Komputer-komputer dihubungkan ke hub, sedangkan hub lain di hubungkan sebagai jalur tulang punggung.

6. Topologi linier

   Jaringan komputer dengan topologi runtut (*linear topology*) biasa disebut dengan topologi bus beruntut, tata letak ini termasuk tata letak umum. Satu kabel utama menghubungkan tiap titik sambungan (komputer) yang dihubungkan dengan penyambung yang disebut dengan Penyambung-T dan pada ujungnya harus diakhiri dengan sebuah penamat (terminator). Penyambung yang digunakan berjenis

BNC (*British Naval Connector*: Penyambung Bahari Britania), sebenarnya BNC adalah nama penyambung bukan nama kabelnya, kabel yang digunakan adalah RG 58 (Kabel Sepaksi Thinnet). Pemasangan dari topologi bus beruntut ini sangat sederhana dan murah tetapi sebanyaknya hanya dapat terdiri dari 5-7 komputer.

2.7 Perangkat Lunak Pendukung

Berikut adalah beberapa perangkat lunak pendukung dalam melakukan perancangan dan pembangunan sistem informasi ini.

2.7.1 Java

Java adalah bahasa pemrograman yang dapat dijalankan di berbagai komputer termasuk telepon genggam.Bahasa ini awalnya dibuat oleh James Gosling saat masih bergabung di Sun Microsystems saat ini merupakan bagian dari Oracle dan dirilis tahun 1995. Bahasa ini banyak mengadopsi sintaksis yang terdapat pada C dan C++ namun dengan sintaksis model objek yang lebih sederhana serta dukungan rutin-rutin aras bawah yang minimal. Aplikasi-aplikasi berbasis *java* umumnya dikompilasi ke dalam *p-code* (*bytecode*) dan dapat dijalankan pada berbagai Mesin Virtual Java (JVM).Java merupakan bahasa pemrograman yang bersifat umum/non-spesifik (*general purpose*), dan secara khusus didisain untuk memanfaatkan dependensi implementasi seminimal mungkin.Karena fungsionalitasnya yang memungkinkan aplikasi java mampu berjalan di beberapa platform sistem

operasi yang berbeda.Saat ini *java* merupakan bahasa pemrograman yang paling populer digunakan, dan secara luas dimanfaatkan dalam pengembangan berbagai jenis perangkat lunak aplikasi ataupun aplikasi berbasis web.

2.7.2 MySQL

MySQL adalah sebuah implementasi dari sistem manajemen basisdata relasionalyang didistribusikan secara gratis dibawah lisensi *GPL (General Public License).*Setiap pengguna dapat secara bebas menggunakan *MySQL*, namun dengan batasan perangkat lunak tersebut tidak boleh dijadikan produk turunan yang bersifat komersial.*MySQL* sebenarnya merupakan turunan salah satu konsep utama dalam basisdata yang telah ada sebelumnya; *SQL (Structured Query Language).SQL* adalah sebuah konsep pengoperasian basisdata, terutama untuk pemilihan atau seleksi dan pemasukan data, yang memungkinkan pengoperasian data dikerjakan dengan mudah secara otomatis.

Kehandalan suatu sistem basisdata dapat diketahui dari cara kerja pengoptimasi-nya dalam melakukan proses perintah-perintah *SQL* yang dibuat oleh pengguna maupun program-program aplikasi yang memanfaatkannya. Sebagai peladen basis data, *MySQL* mendukung operasi basisdata transaksional maupun operasi basisdata non-transaksional.Pada modus operasi non-transaksional, *MySQL* dapat dikatakan unggul dalam hal unjuk kerja dibandingkan perangkat lunak peladen basisdata kompetitor lainnya.Namun

demikian pada modus non-transaksional tidak ada jaminan atas reliabilitas terhadap data yang tersimpan, karenanya modus non-transaksional hanya cocok untuk jenis aplikasi yang tidak membutuhkan reliabilitas data seperti aplikasi *blogging* berbasis web, *CMS*, dan sejenisnya. Untuk kebutuhan sistem yang ditujukan untuk bisnis sangat disarankan untuk menggunakan modus basisdata transaksional, hanya saja sebagai konsekuensinya unjuk kerja *MySQL* pada modus transaksional tidak secepat unjuk kerja pada modus non-transaksional.

2.8 Konsep Dasar *Maintenance*

2.8.1 Pengertian *Maintenance*

Pengertian *maintenance* atau yang sering disebut juga dengan perawatan Menurut Sisjono dan Iwan Koswara, Perawatan (*maintenance*) ialah suatu kegiatan yang dilakukan secara sengaja terhadap suatu fasilitas dengan menganut suatu sistematika tertentu untuk mencapai hasil telah ditetapkan (Sisjono dan Iwan Koswara, 2004).

2.8.2 Jenis - Jenis *Maintenance*

Didalam konsep dari *maintenance* terbagi menjadi dua jenis yang sering digunakan yaitu *preventive maintenance* dan *corrective maintenance.* Berikut adalah pengertian dari jenis jenis maintenance :

1. Preventive Maintenance

Adalah suatu pengamatan secara sistematik disertai analisis teknis-ekonomis untuk menjamin berfungsinya suatu peralatan produksi dan memperpanjang umur peralatan yang bersangkutan.Tujuan *preventive maintenance* adalah untuk dapat mencapai suatu tingkat pemeliharaan terhadap semua peralatan produksi agar diperoleh suatu kualitas produk yang optimum.

Salah satu dari tujuan *Preventive Maintenance* adalah untuk menemukan suatu tingkat keadaan yang menunjukan gejala kerusakan sebelum alat-alat tersebut mengalami kerusakan fatal. Hal ini dapat dilakukan dengan jalan membuat perencanaan dan penjadwalan kegiatan maintenance dengan interupsi sekecil mungkin terhadap proses produksi.

2. Corrective Maintenance

Corrective Maintenance merupakan kegiatan perawatan yang dilakukan untuk mengatasi kegagalan atau kerusakan yang ditemukan selama masa waktu preventive maintenance.Pada umumnya, *corrective maintenance* bukanlah aktivitas perawatan yang terjadwal, karena dilakukan setelah sebuah komponen mengalami kerusakan dan bertujuan untuk mengembalikan kehandalan sebuah komponen atau sistem ke kondisi semula.

Corrective maintenance, dikenal sebagai breakdown atau *run to failure maintenance*. Pemeliharaan hanya dilakukan setelah peralatan atau mesin rusak. Bila strategi pemeliharaan ini digunakan sebagai strategi utama akan menimbulkan dampak tingginya kegiatan pemeliharaan yang tidak direncanakan dan inventori part pengganti.

2.8.3 Tujuan *Maintenance*

Maintenance memiliki tujuan tujuan untuk unit peralatan produksi itu sendiri, berikut adalah poin poin dari tujuan *maintenance:*

1. Memperpanjang usia pakai dari alat produksi terutama bagi peralatan yang sulit diganti suku cadangnya.
2. Memaksimalkan tingkat kegunaan dari alat produksi tersebut agar bisa digunakan sesuai fungsinya dan meminimalisir kesalahan produksi.
3. Memelihara peralatan dan fasilitas dari kerusakan dan keausan.
4. Menjamin kesiapan operasional dari semua peralatan ketikan kegiatan produksi dilaksanakan.
5. Menjamin keselamatan personil ketikan menggunakan alat tersebut.

2.9 Penjadwalan

Penjadwalan adalah suatu proses pengalokasian sumber daya yang terbatas untuk melakukan beraneka ragam pekerjaan pada bidang manufaktur dengan mesin-mesin dan tenaga kerja sebagai sumber dayanya dan *job* sebagai tugas yang harus diselesaikan.

Menurut Jay heizer dan Barry Render (2001:466) menjelaskan penjadwalan sebagai berikut "Penjadwalan yang efesien dalah bagaimana perusahaan dapat memenuhi batas tanggal seperti yang dijanjikan kepada pelanggan dan menghadapi kompetisi berdasarkan waktu".

2.9.1 Tujuan Penjadwalan

Tujuan penjadwalan produksi menurut Roberto S. Russel dan Bernard W. Tayor III (2006:719) adalah

1. *Meeting customer due date* (membuat tanggal jatuh tempo konsumen)
2. *Minimize job lateness* (meminimalkan keterlambatan kerja)
3. *Minimize response time* (meminimalkan waktu respon)
4. *Minimize completion time* (Meminimalkan waktu penyelesaian)
5. *Minimize time in the system* (meminimalkan waktu dalam sistem)

   Minimize overtime (meminimalkan kelebihan waktu
6. *Maximizing machine or laborutilization* (memaksimalkan penggunaan mesin atau tenaga kerja)

7. *Minimize late time* (meminimalkan waktu keterlambatan)
8. *Minimize work inthe processinventory*(meminimalkanpersediaanbarang dalam proses)

2.9.2 Manfaat Penjadwalan

Menurut Jay Heizer and Barry Render (2001:466) akan pentingnya penjadawalan adalah dengan implikasi strategis, adalah sangat jelas bagi perusahaan akan pentingnya penjadwalan dengan penjadwalan yang efektif, perusahaan menggunakan asetnya dengan efektif dan menghasilkan kapasitas dolar yang diinvestasikan menjadi lebih besar, yang sebaliknya akan mengurangi biaya Penjadwalan menambah kapasitas dan fleksibilitas yang terkait memberikan waktu pengiriman yang lebih cepat dan dengan demikian pelayanan kepada pelanggan menjadi baik keuntungan yang ketiga dari bagusnya penjadwalan adalah keunggulan kompetitif dengan pengiriman yang bisa diandalkan.

Fungsi pokok dari penjadawalan adalah untuk membuat arus barang dapat berjalan lancar sesuai dengan waktu yang direncanakan melalui produksi. Hal ini membuat penyelesaian dan pengiriman produk akan dilakukan tepat waktu, dan secara tidak langsung menghindari keterlambatan penerimaan kepada konsumen.

2.9.3 Kriteria Proses Penjadwalan

Teknik penjadwalan yang benar tergantung pada volume pesanan, ciri operasi,dan keseluruhan kompleksitas pekerjaan, sekaligus pentingnya tempat pada masingmasing dari empat kriteria. Menurut Jay Heizer and Barry Render (2001:468)

menyebutkan kriteria dari penjadwalan adalah sebagai berikut:

1. Meminimalkan waktu penyelesaian. Ini dinilai dengan menentukan rata-rata penyelesaian
2. Memaksimalkan utilisasi. Ini dinilai dengan menentukan persentase waktu fasilitas itu digunakan.
3. Meminimalkan persediaan barang dalam proses. Ini dinilai dengan menentukan rata-rata jumlah pekerjaan dalam sistem. hubungan antara jumlah pekerjaan dalam sistem dan persediaan barang dalam proses adalah tinggi. dengan demikian semakin kecil jumlah pekerjaan yang ada di dalam sistem, maka akan semakin kecil persediaannya.
4. Meminimalkan waktu tungggu pelanggan. Ini dinilai dengan menentukan rata-rata jumlah keterlambatan.

2.9.4 Proses Penjadwalan

Menurut Jay Heizer and Barry Render (2001: 468) untuk mengolah fasilitas dengan cara yang seimbang dan efesien manajer membutuhkan perencanaan produksi dan sistem pengendalian. Proses penjadwalan harus melalui tahapan sebagai berikut:

1. Penjadwalan pesanan yang akan datang tanpa mengganggu kendala kapasitas pusat kerja individual.
2. Mengecek ketersediaan alat-alat dan bahan baku sebelum memberikan pesanan ke suatu departemen.

3. Membuat tanggal jatuh tempo untuk masing-masing pekerjaan dan mengecek kemajuan terhadapa tanggal keperluan dan waktu tempuh pesanan.
4. Mengecek barang dalam proses pada saat pekerjaan bergerak menuju perusahaan
5. Memberikan *feedback* pada pabrik efesiensi pekerjaan dan memonitor waktu operator untuk analisis distribusi tenaga kerja dan gaji dan upah.