BAB II

LANDASAN TEORI

2.1 Pengertian Sistem

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

Menurut (Jogiyanto:2001) mendefinisikan sebuah sistem sebagai berikut: "Sistem merupakan suatu jaringan kerja prosedur-prosedur yang saling berhubungan berkumpul bersama-sama untuk melakukan suatu kegiatan atau menyelesaikan suatu sasaran tertentu".

2.1.1 Karakteristik Sistem

Menurut Jogiyanto (2005 : 3) Sistem mempunyai karakteristik yang tidak bisa dipisahkan antara satu karakteristik dengan karakteristik yang lain. beberapa karakteristik tersebut antara lain:

1. Komponen sistem

   Suatu sistem terdiri dari sejumlah komponen yang saling berintegrasi, yang artinya saling bekerja sama membentuk suatu kesatuan. Komponenkomponen sistem atau elemen-elemen sistem dapat berupa subsistem atau bagian-bagian dari sistem.

2. Batasan sistem

Merupakan suatu daerah yang membatasi suatu sistem dengan sistem yang lain atau lingkungan luar, dengan batasan ini kita bisa mengetahui ruang lingkup suatu sistem.

3. Lingkungan luar sistem

   Apapun yang ada diluar batas sistem yang mempengaruhi oprasi suatu sistem.

4. Penghubung sistem

   Merupakan media penghubung antara satu sub sistem dengan sistem lainnya. Dengan penghubung ini akan mengalir data-data sub sistem dimana keluaran (output) dari sistem akan menjadi masukan (input) untuk sub sistem yang lain, sehingga antara sistem dan sub sistem akan saling berinteraksi membentuk satu kesatuan.

5. Masukan sistem

   Merupakan energi yang dimasukan kedalam sistem, dimana masukan ini dapat berupa masukan perawatan (maintenance input) dan masukan sinyal (signal input).

6. Keluaran sistem

   Merupakan hasil dari energy yang diolah dan didefinisikan menjadi keluaran yang berguna dan mampu menjadikan masukan baru atau informasi yang dibutuhkan.

7. Pengolahan sistem

   Setiap sistem pasti mempunyai pengolahan data masukan untuk diolah menjadi sebuah informasi.

8. Sasaran sistem

   Merupakan penentu dari tujuan untuk menentukan masukan yang dibutuhkan dan keluar yang akan dihasilkan sebuah sistem.

2.1.2 Klasifikasi Sistem

Sistem diklasifikasikan berdasarkan jenisnya, diantaranya yaitu :

1) Sistem Abstrak : sistem yang berupa pemikiran atau ide-ide yang tidak tampak secara fisik (sistem teologia).
2) Sistem Fisik : merupakan sistem yang ada secara fisik (sistem komputer, sistem akutansi, sistem produksi, dll).
3) Sistem Alamiah ; sistem yang terjadi melalui proses alam. (sistem matahari, sistem luar angkasa, sistem reproduksi dll).
4) Sistem Buatan Manusia ; sistem yang dirancang oleh manusia, Sistem buatan manusia yang melibatkan interaksi manusia dengan mesin disebut human-machine sistem (contoh ; sistem informasi).
5) Sistem Tertentu (Deterministic sistem); beroperasi dengan tingkah laku yang sudah dapat diprediksi. Interaksi bagian-bagiannya dapat dideteksi dengan pasti sehingga keluaran dari sistem dapat diramalkan (contoh : sistem komputer).
6) Sistem Tak Tentu (probabilistic sistem) : sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.
7) Sistem Tertutup (close sistem) : sistem yang tidak berhubungan dan tidak terpengaruh dengan sistem luarnya. Sistem ini bekerja secara

otomatis tanpa adanya turut campur ada sistem yang benar-benar tertutup, yang ada hanyalah relatively closed sistem (secara relatif tertutup, tidak benar-benar tertutup).

8) Sistem Terbuka (open sistem) : sistem yang berhubungan dan terpangaruh dengan lingkungan luarnya.

2.2. Pengertian informasi

Definisi umum dari Informasi adalah data yang telah diproses menjadi bentuk yang memiliki arti bagi penerima dan dapat berupa fakta, suatu nilai yang bermanfaat.

Menurut Jogiyanto (2005 : 8) Informasi diartikan sebagai data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi yang menerimanya.

2.3. Pengertian Sistem Informasi

Menurut Jogiyanto (2005 : 11) yang dimaksud sistem informasi adalah suatu sistem dalam suatu organisasi yang mempertemukan kebutuhan pengolahan transaksi harian, mendukung operasi, bersifat manajerial dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang diperlukan.

Menurut Jogiyanto (2005:36), tujuan dari sistem informasi adalah menghasilkan informasi. Tugas dari sistem informasi adalah untuk melakukan siklus pengolahan data. Untuk melakukan siklus ini, maka sebagai suatu sistem diperlukan komponen-komponen tertentu.

Abdul Kadir (2003 : 70) juga menyebutkan dalam suatu sistem informasi terdapat komponen-komponen seperti :

1. Perangkat keras (*hardware*), yaitu mencakup pidanti-piranti fisik seperti komputer dan printer.
2. Perangkat lunak (*software*), yaitu sekumpulan instruksi yang memungkinkan perangkat keras untuk dapat memproses data.
3. Prosedur, yaitu sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.
4. Orang (*user*), yaitu pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan, dan penggunaan keluaran sistem informasi.
5. Basis data (*database*), yaitu sekumpulan table, hubungan, dan lain-lain yang berkaitan dengan penyimpanan data.

Jaringan komputer dan komunikasi data, yaitu sistem penghubung yang memungkinkan sesumber (*resources*) dipakai secara bersama atau diakses oleh sejumlah pemakai.

2.4. Analisis dan Perancangan Terobject

Proses penyiapan spesifikasi yang terperinci untuk mengembangkan sistem baru, dalam langkah permulaan perancangan sistem adalah rencana pengembangan disiapkan selama sintesa sebagaimana di modifikasi dan disetujui oleh manajemen. Tahap perancangannya harus mengisi semua perincian rencana pengembangan agar sistem baru dapat diimplementasikan secara maksimal dan memuaskan. Tujuan dari perancangan sistem secara global adalah membentuk kerangka sistem pengolahan data dengan bantuan komputer, untuk mewujudkannya dilakukan beberapa tahap yaitu :

1. Menentukan persyaratan dan batasan sistem yang dirancang.

2. Menentukan pola rancangan aliran informasi.
3. Menentukan rancangan sistem pengolahan data dan basis data.

2.5. Metode Pendekatan Sistem

Metode pendekatan sistem yang digunakan oleh penulis adalah pendekatan dengan *Object Oriented* yang menggunakan OOA(*Object Oriented Analisis*).

2.5.1. Konsep Dasar *Object Oriented Analisis* (OOA)

1. Obyek adalah benda secara fisik atau konseptual yang memiliki keadaan (state) dan perilaku (behavior).
2. Kelas (Class) adalah definisi umum (pola, template atau cetak biru) untuk himpunan obyek sejenis.
3. Kotak hitam dan Interface sebuah obyek digambarkan sebagai kotak hitam untuk mengakses obyek melalui interface. Kotak hitam berisi Kode (himpunan instruksi dengan bahasa yang dipahami komputer) dan Data.
4. Association dan Aggregation Association adalah hubungan antar obyek yang saling membutuhkan. Aggregation adalah menggambarkan seluruh bagian dari obyek.

2.5.2. Karakteristik dari Objek

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

2. Objek dapat kongkrit, seperti halnya arsip dalam sistem, atau konseptual seperti kebijakan penjadualan dalam multiprocessing dlm suatu sistem operasi.
3. Setiap objek mempunyai sifat yang melekat pada identitasnya.
4. Dua objek dapat berbeda walaupun semua nilai atributnya identik.

2.5.3. Istilah-istilah Objek

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)

2.5.4. 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.

Objek mewakili fakta/keterangan dari sebuah kelas

2.5.5. Metodologi Berorientasi Objek

1. Encapsulation (pengkapsulan) merupakan dasar untuk pembatasan ruang lingkup program terhadap data yang diproses, data dan

prosedur atau fungsi dikemas bersama-sama dalam suatu objek, sehingga prosedur atau fungsi lain dari luar tidak dapat mengaksesnya, data terlindung dari prosedur atau objek lain, kecuali prosedur yang berada dalam objek itu sendiri

2. Inheritance (pewarisan) adalah teknik yang menyatakan bahwa anak dari objek akan mewarisi data/atribut dan metode dari induknya langsung. Atribut dan metode dari objek induk diturunkan kepada anak objek, demikian seterusnya, Inheritance mempunyai arti bahwa atribut dan operasi akan dimiliki bersama di antara kelas yang mempunyai hubungan secara hirarki, suatu kelas dapat ditentukan secara umum, kemudian ditentukan spesifik menjadi subkelas. Setiap subkelas mempunyai hubungan atau mewarisi semua sifat yang dimiliki oleh kelas induknya, dan ditambah dengan sifat unik yang dimilikinya.
3. Polymorphism (polimorfisme) yaitu konsep yang menyatakan bahwa sesuatu yang sama dapat mempunyai bentuk dan perilaku berbeda, Polimorfisme mempunyai arti bahwa operasi yang sama mungkin mempunyai perbedaan dalam kelas yang berbeda, kemampuan objek-objek yang berbeda untuk melakukan metode yang sesuai dalam merespon pesan yang sama, seleksi dari metode yang sesuai, bergantung pada kelas yang seharusnya menciptakan objek.

2.6. Data dan Basis Data

Menurut [Jogiyanto:2001] menyebutkan bahwa yang dimaksud dengan data adalah “fakta tentang sesuatu di dunia nyata yang dapat direkam dan disimpan pada media komputer”.Sedangkan basis data menurut [Jogiyanto:2001] adalah ”kumpulan terorganisasi dari datadata yang berhubungan sedemikian rupa sehingga mudah disimpan, dimanipulasi serta dipanggil oleh pengguna”.

2.7. Alat Bantu Analisis dan Perancangan

2.7.1. UML (*Unified Modeling Languages*)

Menurut Bambang Hariyanto. Ir.,MT. (2004:17) UML adalah bahasa grafis untuk mendokumentasikan, menspesifikasikan, dan membangun sistem perangkat lunak. UML berorientasi objek menerapkan banyak level abstraksi, tidak bergantung proses pengembangan, tidak bergantung bahasa dan teknologi.

Tujuan UML

1. Menyediakan bahasa pemodelan visual yang ekspresif dan siap pakai untuk mengembangkan pertukaran model-model yang berarti.
2. Menyediakan mekanisme perluasan dan spesialisasi untuk memperluas konsep-konsep inti.
3. Mendukung spesifikasi independent bahasa pemprograman dan proses pengembangan tertentu.
4. Menyediakan basis formal untuk pemahaman bahasa pemodelan.

2.7.1.1. Bagian-Bagian Dari UML

UML sendiri terdiri atas pengelompokkan diagram-diagram sistem menurut aspek atau sudut pandang tertentu. Diagram adalah yang menggambarkan permasalahan maupun solusi dari permasalahan suatu model.

1. *Use Case Diagram*

   Sebuah *use case* menggambarkan suatu urutan interaksi antara satu atau lebih aktor dan sistem. Dalam fase *requirements*, model *use case* mengambarkan sistem sebagai sebuah kotak hitam dan interaksi antara aktor dan sistem dalam suatu bentuk naratif, yang terdiri dari input user dan respon-respon sistem. Setiap use case menggambarkan perilaku sejumlah aspek sistem, tanpa mengurangi struktur internalnya. Selama pembuatan model use case secara pararel juga harus ditetapkan objek-objek yang terlibat dalam setiap use case.

   Use case Menggambarkan sejumlah *external actors* dan hubungannya ke *use case* yang diberikan oleh sistem. *Use case* adalah deskripsi fungsi yang disediakan oleh sistem kedalam bentuk teks sebagai dokumentasi dari *use case symbol* namun dapat juga dilakukan dalam *activity diagrams*.

   Use case digambarkan hanya yang dilihat dari luar oleh *actor* (keadaan lingkungan sistem yang dilihat user) dan bukan

bagaimana fungsi yang ada di dalam sistem. Simbol yang digunakan yaitu:

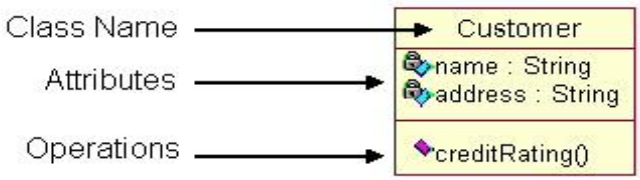


Gambar 2.1. Use Case Diagram

2. *Class Diagram*

Menggambarkan struktur statis *class* di dalam sistem. *Class* merepresentasikan sesuatu yang ditangani oleh sistem. *Class* dapat berhubungan dengan yang lain melalui berbagai cara: *associated* (terhubung satu sama lain), *dependent* (satu *class* tergantung/menggunakan *class* yang lain), *specialed* (satu *class* merupakan spesialisasi dari *class* lainnya) atau *package* (grup bersama sebagai satu unit). Sebuah sistem biasanya mempunyai beberapa *class diagram.*

Suatu *class* biasanya terdiri dari 3 bagian, yaitu nama, atribut, dan operasi. Berikut adalah contoh dari suatu *class*:

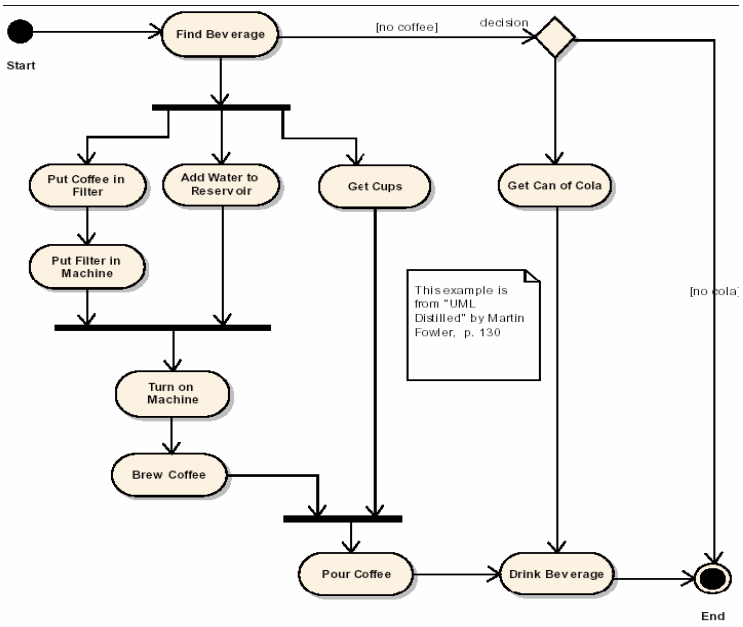


Gambar 2.2. Bagian-bagian dari class

3. Activity Diagram

Menggambarkan rangkaian aliran dari aktivitas, digunakan untuk mendeskripsikan aktifitas yang dibentuk dalam suatu operasi sehingga dapat juga digunakan untuk aktifitas lainnya seperti *use case* atau interaksi.Sebuah aktivitas dapat direalisasikan oleh satu *use case* atau lebih. Aktivitas menggambarkan proses yang berjalan, sementara *use case* menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Sama dengan *state,* standar *UML* menggunakan segi empat dengan sudut membulat untuk menggambarkan aktivitas. *Decision* digunakan untuk menggambarkan *behaviour* pada kondisi tertentu. Untuk mengilustrasikan proses-proses pararel (*fork on join*) digunakan titik sinkronisasi yang dapat berupa titik, garis horizontal atau vertikal. *Activity diagram* dapat dibagi menjadi beberapa *object swimlane* untuk menggambarkan objek mana yang bertanggung jawab untuk aktivitas tertentu.

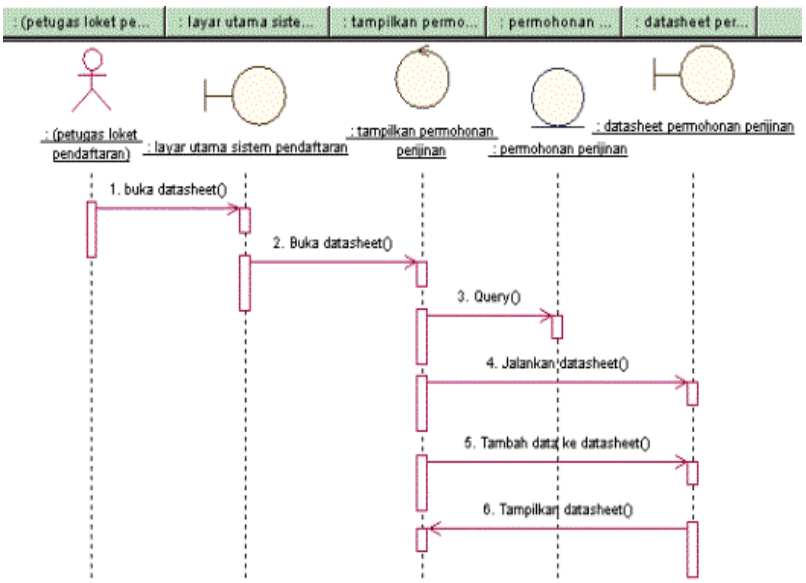


Gambar 2.3. Activity Diagram

4. Sequence Diagram

Menggambarkan kolaborasi dinamis antara sejumlah *object*. Kegunaannya untuk menunjukkan rangkaian pesan yang dikirim antara *object* juga interaksi antara *object*, sesuatu yang terjadi pada titik tertentu dalam eksekusi sistem.

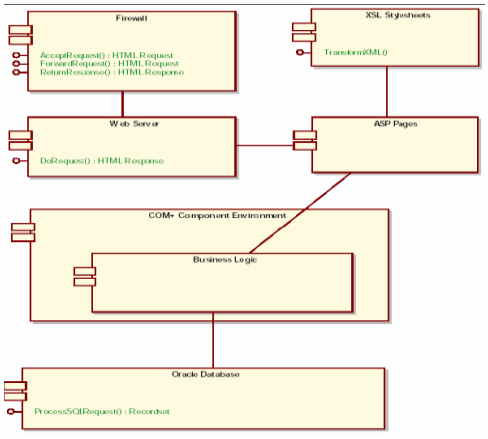
Komponen utama *sequence diagram* terdiri atas objek yang dituliskan dengan kotak segiempat bernama pesan diwakili oleh garis dengan tanda panah dan waktu.



Gambar 2.4. Sequence Diagram

5. Component Diagram

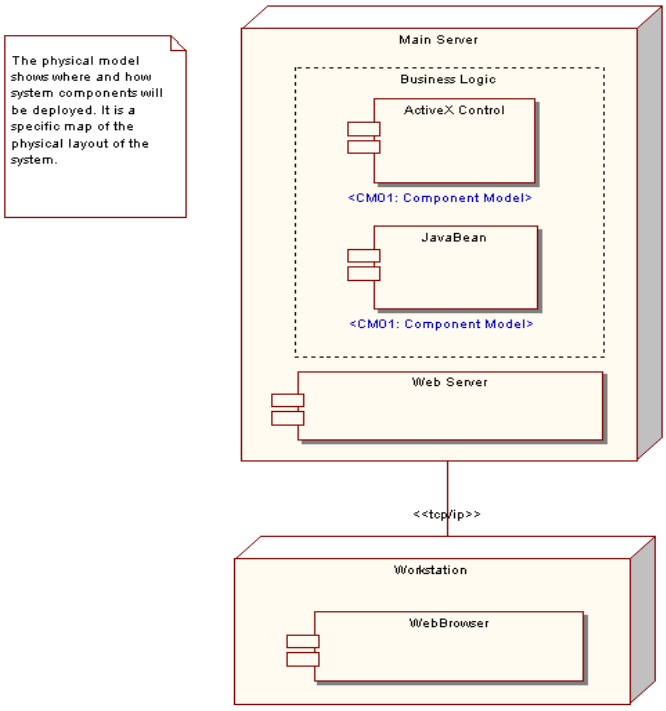
Menggambarkan struktur fisik kode dari komponen. Komponen dapat berupa *source code*, komponen biner atau *executeable component*. Sebuah komponen berisi informasi tentang *logic class* atau *class* yang diimplementasikan sehingga membuat pemetaan dari *logical view* ke *component view*.



Gambar 2.5. Component Diagram

6. Deployment Diagram

Menggambarkan arsitektur fisik dari perangkat keras dan perangkat lunak sistem, menunjukkan hubungan komputer dengan perangkat (*nodes*) satu sama lain dan jenis hubungannya. Di dalam *nodes*, *executeable component* dan *object* yang dialokasikan untuk memperlihatkan unit perangkat lunak yang dieksekusi oleh *node* tertentu dan ketergantungan komponen.



Gambar 2.6. Deployment Diagram

2.7.2. Tools Yang Mendukung UML

2.7.2.1. Rational Rose

Dalam penelitian ini, *Rational Rose* berfungsi sebagai *tool* untuk pemodelan sistem yang menggambarkan proses-proses yang ada pada sistem ini. *Rational Rose* adalah kakas (*tools)* pemodelan visual untuk pengembangan sistem berbasis objek yang sangat handal untuk digunakan sebagai bantuan bagi para pengembang dalam melakukan analisis dan perancangan sistem.

Rational Rose digunakan untuk melakukan pemodelan sistem sebelum pengembang menulis kode-kode dalam bahasa pemrograman tertentu. Ia juga membantu analisis sistem dengan cara pengembang membuat diagram *use case* untuk melihat fungsionalitas sistem secara keseluruhan sesuai dengan harapan dan keinginan pengguna. Kemudian, ia juga menuntut pengembang untuk mengembangkan *Interaction* diagram untuk melihat bagaimana objek-objek saling bekerjasama dalam menyediakan fungsionalitas yang diperlukan (Adi Nugroho, 2005:20).

2.8. Definisi Permasalahan

2.8.1. Penjualan

Penjualan merupakan salah satu fungsi dari pemasaran atau merupakan bagian dari kegiatan pemasaran. Penjualan sangat penting dan menetukan karena suatu perusahaan untuk melakukan suatu penjualan yang baik harus mempunyai pemasaran yang baik pula.

Jadi Sistem Informasi Penjualan adalah suatu sistem informasi yang mengorganisasikan serangkaian prosedur dan metode yang dirancang untuk menghasilkan, menganalisa, menyebarkan dan memperoleh informasi guna mendukung pengambilan keputusan mengenai penjualan.

2.8.2. Pembelian

Pembelian merupakan salah satu kegiatan dari pembelanjaan. Pembelian juga mempunyai peranan yang penting, sama halnya dengan penjualan yaitu untuk memenuhi kebutuhan perusahaan.

Pembelian adalah membeli aktiva produk untuk digunakan dalam kegiatan perusahaan yang digunakan untuk mencatat pembelian barang dagangan selama satu periode, hanya digunakan untuk mencatat pembelian barang dagangannya.

2.8.3. Persediaan Barang

Persediaan adalah suatu aktiva yang meliputi barang-barang milik perusahaan dengan maksud untuk dijual dalam suatu periode usaha yang normal. Jadi persediaan merupakan sejumlah barang yang disediakan untuk memenuhi permintaan dari pelanggan.

Dalam perusahaan perdagangan pada dasarnya hanya ada satu golongan inventory (persediaan), yang mempunyai sifat perputaran yang sama yaitu yang disebut "*Merchandise Inventory*" (persediaan barang dagangan). Persediaan ini merupakan persediaan barang yang selalu dalam perputaran, yang selalu dibeli dan dijual, yang tidak mengalami proses lebih

lanjut didalam perusahaan tersebut yang mengakibatkan perubahan bentuk dari barang yang bersangkutan.

2.8.4. Pemesanan Barang

Dalam dunia usaha pemesanan adalah hal yang penting dan tidak bisa dihilangkan. menurut Dr. Dendry Sugono Kepala Pusat Bahasa, pengertian pesan adalah suruhan (perintah, nasehat, Permintaan, amanat) yang harus dilakukan atau disampaikan kepada orang lain.

Pesanan adalah permintaan hendak membeli supaya dikirimkan, dibuatkan dan sebagainya. Pemesan/konsumen masa kini menemukan banyak sekali produk dalam setiap kategori. Pemesan memiliki beragam kebutuhan dalam kombinasi serta harga barang/jasa. Harapan pemesan akan mutu dan pelayanan akan semakin tinggi serta terus meningkat. Ditengah pilihan yang paling sesuai dengan kebutuhan serta harapan dan membeli berdasarkan padangan nilai mereka.

Pemesanan tidak hanya dipandangi sebagai satu bagian. Bagian penjualan terlibat dalam mengambil keputusan manajemen sebelum produk dirancang sampai terjual. Perusahaan-perusahaan unggulan masa kini beralih dari memandang perushaan sebagai kumpul bagian – bagian menajadi suatu sistem untuk mengelola dan menguasai proses ini seperti penciptaan pemesanan dan pemenuhan pesanan. Setiap proses meliputi beberapa langkah dan memerlukan masukan dari beberapa bagian produksi, keuangan dan manager.

Berdasarkan pengertian diatas maka penulis dapat menyimpulkan bahwa pemesanan barang adalah permintaan dibuatkan barang sesuai dengan keinginan pelanggan, yang telah disepakati spesifikasinya oleh bagian administrasi yang kemudian akan di kerjakan oleh bagian produksi dan ketika barang itu jadi maka akan di kirimkan kepada pelanggan yang bersangkutan. (http://pusatbahasa.diknas.go.id/kbbi/index.php).

2.9. Bahasa Pemrograman Yang Digunakan

2.9.1. Java

Java adalah bahasa pemrograman tingkat tinggi yang berorientasi objek dan program java tersusun dari bagian yang disebut kelas. Kelas terdiri atas metode metode yang melakukan pekerjaan dan mengembalikan informasi setelah melakukan tugasnya. Para pemrogram 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 pemrograman tingkat

tinggi, serta banyak pekerjaan pemrograman yang mulanya harus dilakukan manual, sekarang digantikan dikerjakan Java secara otomatis seperti alokasi memori.

2.9.2. MySQL

MySQL merupakan *software* sistem manajemen database *(Database Management Sistem-DBMS)* yang sangat popular dikalangan pemrogram Web, terutama di lingkungan Linux dengan menggunakan *script* PHP dan *perl software*. Database ini telah tersedia juga pada *platform* sistem operasi Windows (98/ME ataupun NT/2000/XP) sebagai sumber dan pengelolaan datanya.

Komponen – komponen Mysql meliputi:

a. Tabel

Tabel adalah kumpulan data yang tersusun menurut aturan tertentu . Secara fisik, tabel berupa *grid* yang terdiri atas baris dan kolom. Baris menunjukkan *record* data dan kolom menunjukkan *field* data.

b. *Key*

Key berguna sebagai wakil dari tabel yang menunjukkan nilai unik suatu *field.* Dengan adanya *key*, proses pencarian menjadi lebih mudah karena tidak ada nilai yang kembar.

Ada tiga macam *key* yang dapat diterapkan pada suatu tabel yaitu:

1. Superkey

Superkey merupakan satu atau lebih atribut yang dapat membedakan setiap baris data dalam sebuah tabel secara unik, ada lebih dari satu kumpulan atribut yang bersifat seperti itu pada sebuah tabel.

2. *Candidate key*

   Candidate key merupakan kumpulan atribut minimal yang dapat membedakan setiap baris data dalam sebuah tabel secara unik. Jadi, sebuah *candidate key* pastilah *superkey*, tapi belum tentu sebaliknya.

3. *Primary key*

   Salah satu dari *candidate key* (jika memang ada lebih dari satu) dapat dijadikan sebagai *primary key*.

Pemilihan *primary key* dari sejumlah *candidate key* tersebut umumnya didasari oleh :

1. *Key* tersebut lebih sering dijadikan sebagai acuan,
2. *Key* tersebut lebih ringkas,
3. Jaminan keunikan *key* tersebut lebih baik.

c. *Relationship*

Database dengan kapasitas kecil cukup menggunakan sebuah tabel saja. *Database* yang demikian lazim disebut dengan *file-file database*. Sedangkan untuk suatu sistem *database* yang baik dan handal, organisasi data harus melibatkan lebih dari satu tabel dan seluruh tabel itu harus saling dihubungkan, untuk itu mutlak diperlukan kehadiran suatu *relationship* (relasi).

Setelah membuat sebuah hubungan, kita wajib menentukan aturan yang dikenakan padanya. Aturan dasar yang telah baku antara lain:

1. *field* yang dihubungkan dari tabel utama haruslah merupakan *primary key,*
2. *field* yang saling terhubung harus memiliki jenis data yang sama.

2.10. Perangkat Lunak Yang Digunakan

2.10.1. NetBeans

NetBeans mengacu pada kedua platform kerangka untuk aplikasi dektop java, dan sebuah lingkungan pengembangan terpadu (IDE) untuk pengembangan dengan java, java script, php, python, ruby, groovy, c, c++, scala, clojure, dan lain-lain. NetBeans yang ditulis dalam java dan berjalan di mana-mana JVM diinstal, termasuk windows, Mac Os, Linux, solaris. Sebuah JDK diperlukan untuk pengembangan fungsionalitas jawa, tetapi tidak diperlukan untuk pembangunan di bahasa pemograman lain.

Platform NetBeans memungkinkan aplikasi untuk dikembangkan dari satu set modular komponen software yang disebut modul. Aplikasi berbasis platform NeatBeans(termasuk IDE NetBeans) dapat diperpanjang oleh pengembangan pihak ketiga. (Miftakhul Huda & Bunafit Komputer. 2010. *Membuat Aplikasi Database dengan Java, MySQL, dan NetBeans*. PT Elex Media Komputindo. Jakarta.).

2.10.2. 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 penerjemah bahasa yang ditulis dengan bahasa pemrograman PHP dan Perl.

Nama XAMPP merupakan singkatan dari X (empat sistem operasi apapun), Apache, MySQL, PHP dan Perlengkapan Program ini tersedia dalam GNU (General Public License) dan bebas, merupakan web server yang mudah digunakan yang dapat melayani tampilan halaman web yang dinamis. Untuk mendapatkanya dapat mendownload langsung dari web resminya.

Mengenal bagian XAMPP yang biasa digunakan pada umumnya :

1. htdoc adalah folder tempat meletakkan berkas-berkas yang akan dijalankan, seperti berkas PHP, HTML dan skrip lain.
2. 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.
3. Kontrol Panel yang berfungsi untuk mengelola layanan (*service*) XAMPP. Seperti menghentikan (*stop*) layanan, ataupun memulai (*start*).