BAB II

LANDASAN TEORI

2.1 Landasan Teori

Pada bab ini dijelaskan beberapa konsep dan dasar teori yang berkaitan dengan permasalah yang akan dibahas penulis sebagai dasar pemahaman dalam mengimplementasikan konsep-konsep tersebut ke dalam semua kegiatan pengembangan sistem. Dalam melakukan analisis terhadap permasalahan dan perancangan perangkat lunak diperlukan pemahaman terhadap sejumlah teori yang mendukung terhadap aktifitas-aktifitas tersebut, penulis mengkaji sejumlah teori yang akan berguna untuk melakukan analisis dan perancangan perangkat lunak. Teori-teori tersebut merupakan konstribusi dari perkuliahan dan hasil studi literatur.

2.2 Konsep Dasar Sistem

Pengertian dan definisi sistem pada berbagai bidang berbeda-beda, tetapi meskipun istilah sistem yang digunakan bervariasi, semua sistem pada bidang-bidang tersebut mempunyai beberapa persyaratan umum, yaitu sistem harus mempunyai elemen, lingkungan, interaksi antar elemen, interaksi antara elemen dengan lingkungannya, dan yang terpenting adalah sistem harus mempunyai tujuan yang akan dicapai.

2.2.1 Pengertian Sistem

Sistem dapat didefinisikan sebagai seperangkat elemen yang digabungkan satu dengan lainnya untuk suatu tujuan bersama. Kumpulan elemen terdiri dari manusia, mesin, prosedur, dokumen, data atau elemen lain yang teroganisir dari elemen-elemen tersebut. Elemen sistem di samping berhubungan satu sama lain, juga berhubungan dengan lingkungannya untuk mencapai tujuan yang telah ditentukan sebelumnya. Terdapat beberapa definisi sistem yaitu :

Dalam buku Pengenalan Sistem Informasi yang dikutip dari Abdul Kadir (2003 : 54) sistem adalah sekumpulan elemen yang saling terkait atau terpadu yang dimaksudkan untuk mencapai suatu tujuan.

Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran tertentu. Jogiyanto HM (2005 : 1).

Menurut James A. O'Brien (2006 : 29) sistem adalah sekelompok komponen yang saling berhubungan, bekerja bersama untuk mencapai tujuan bersama dengan menerima input serta menghasilkan output dalam proses transformasi yang teratur.

Berdasarkan pendapat dari para ahli di atas mengenai sistem, maka dapat diambil kesimpulan bahwa sistem adalah sekelompok dari dua atau lebih komponen atau subsistem yang saling berhubungan dan bersinergi untuk bekerja sama melakukan suatu urutan kegiatan untuk mencapai tujuan yang tertentu.

2.2.2 Karakteristik Sistem

Suatu sistem memiliki beberapa karakteristik atau sifat-sifat yang tertentu. Berikut penjelasan menurut Jogiyanto HM (2005 : 3) :

1. Mempunyai komponen sistem *(components)*

   Komponen sistem adalah segala sesuatu yang menjadi bagian penyusun sistem. Komponen sistem disebut sebagai subsistem, dapat berupa orang, benda, hal atau kejadian yang terlihat di dalam sistem.

2. Mempunyai batas 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 satu kesatuan. Batas suatu sistem menunjukkan ruang lingkup (*scope*) dari sistem tersebut.

3. Mempunyai lingkungan luar sistem *(environment)*

   Lingkungan sistem adalah segala sesuatu yang berada di luar batas sistem. Lingkungan sistem dapat menguntungkan ataupun merugikan. Umumnya, lingkungan yang menguntungkan akan selalu dipertahankan untuk menjaga keberlangsungan sistem. Sedangkan lingkungan sistem yang merugikan akan diupayakan agar mempunyai pengaruh seminimal mungkin, bahkan jika mungkin ditiadakan.

4. Mempunyai penghubung sistem *(interface)*

   Penghubung merupakan komponen sistem, yaitu segala sesuatu yang bertugas menjembatani hubungan antar komponen dalam sistem. Penghubung merupakan sarana yang memungkinkan setiap komponen saling berinteraksi dan berkomunikasi dalam rangka menjalankan fungsi masing-masing komponen.

5. Mempunyai masukan sistem *(input)*

   Masukan adalah energi yang dimasukkan ke dalam sistem. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). Maintenan input adalah energi yang dimasukkan supaya sistem tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran.

6. Mempunyai pengolahan *(processing)*

   Pengolah merupakan komponen sistem yang mempunyai peran utama mengolah masukan agar menghasilkan keluaran yang berguna bagi para pemakainya.

7. Mempunyai keluaran sistem *(output)*

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

8. Mempunyai sasaran *(objective)* dan tujuan *(goal)*

   Setiap komponen dalam sistem perlu dijaga agar saling bekerja sama dengan harapan agar mampu mencapai sasaran dan tujuan sistem. Sasaran berbeda dengan tujuan. Sasaran sistem adalah apa yang ingin dicapai oleh sistem untuk jangkan waktu yang relative pendek.

2.2.3 Klasifikasi Sistem

Menurut Jogiyanto HM (2005 : 6) klasifikasi sistem dapat dibagi dari beberapa sudut pandang, diantaranya adalah sebagai berikut :

a. Sistem abstrak (*abstract*) dan sistem fisik (ph*ysical*)

   Sistem abstrak (*abstract system*) adalah sistem yang berisi gagasan atau konsep yang tidak tampak secara fisik. Misalnya, sistem teologi yang berisi gagasan tentang hubungan manusia dengan Tuhan. Sistem Fisik (*physical system*)adalah sistem yang secara fisik dapat dilihat. Misalnya, sistem komputer, sistem kepegawaian dan sebagainya.

b. Sistem alamiah (*natural*) dan sistem buatan manusia (h*uman made*)

   Sistem alamiah (*natural system*) adalah sistem yang terjadi karena proses alam, tidak dibuat manusia. Misalnya, sistem tata surya. Sistem buatan manusia (*human made system*) adalah sistem yang dibuat manusia dengan melibatkan interaksi antara manusia dengan sistem tersebut.

c. Sistem tertentu (d*eterministic*) dan tak tentu (p*robabilistic*)

   Sistem deterministik (*deterministic system*) adalah suatu sistem yang dapat operasinya dapat diprediksi secara tepat. Interaksi antar bagian-bagian diketahui dengan pasti. Misalnya, sistem komputer yang secara tepat

menjalankan suatu intruksinya. Sistem tak tentu adalah sistem yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilitas.

d. Sistem tertutup (c*losed*) dan terbuka (*open*)

Sistem tertutup (*closed system*) adalah suatu sistem yang tidak bertukar materi, informasi, atau energi dengan lingkungan. Dengan kata lain, sistem ini tidak berinteraksi dan tidak dipengaruhi oleh lingkungan. Sistem ini bekerja otomatis tanpa adanya turut campur tangan dari pihak luarnya. Sistem terbuka (*open system*) adalah sistem yang berhubungan dengan lingkungan dan diperngaruhi oleh lingkungan. Sistem ini menerima masukan dan menghasilkan keluaran untuk lingkungan luar atau subsistem yang lainnya.

2.3 Konsep Dasar Informasi

Informasi merupakan hal yang sangat penting di dalam sebuah sistem. Jika sebuah sistem mengolah informasi yang salah maka penerima informasi akan susah untuk mengambil keputusan masa kini atau masa yang akan datang.

2.3.1 Pengertian Informasi

Menurut McFadden, dkk (1999) dalam buku Pengenalan Sistem Informasi (Abdul Kadir 2003 : 31) mendefinisikan informasi sebagai data yang telah diproses sedemikian rupa sehingga meningkatkan pengetahuan seseorang yang menggunakan data tersebut.

Menurut Davis (1999) dalam buku Pengenalan Sistem Informasi (Abdul Kadir 2003 : 31) informasi adalah data yang telah diolah menjadi sebuah bentuk yang berarti bagi penerimanya dan bermanfaat dalam pengambilan keputusan saat ini atau saat mendatang.

Menurut Jogiyanto HM (2005 : 8) mendefinisikan informasi adalah sebagai data yang telah diolah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya.

Sumber informasi adalah data, data adalah kenyataan yang menggambarkan kejadian–kejadian dan kesatuan nyata. Kejadian (*event*) adalah sesuatu yang terjadi pada saat tertentu. Informasi diperoleh setelah data-data mentah diproses atau diolah. Menurut John Burch dan Gary Grudnitski, agar informasi dihasilkan lebih berharga, maka informasi harus memenuhi kriteria sebagai berikut :

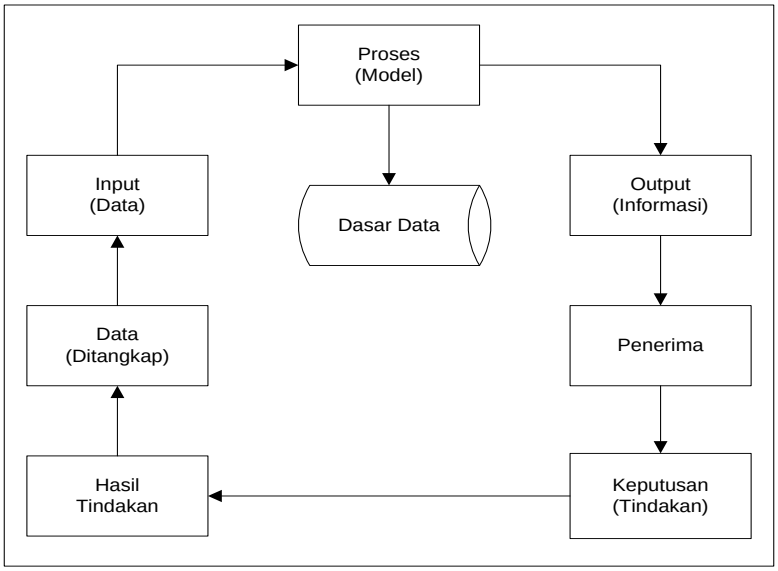
a. Informasi harus akurat, sehingga mendukung pihak manajemen dalam mengambil keputusan.
b. Informasi harus relevan, benar-benar terasa manfaatnya bagi yang membutuhkan.
c. Informasi harus tepat waktu, sehingga tidak ada keterlambatan pada saat dibutuhkan.

Kegunaan informasi adalah untuk mengurangi ketidakpastian di dalam proses pengambilan keputusan tentang suatu keadaan. Informasi yang digunakan di dalam suatu sistem informasi umumnya digunakan untuk beberapa kegunaan informasi digunakan tidak hanya oleh satu orang pihak di dalam organisasi. Nilai

sebuah informasi ditentukan dari dua hal yaitu manfaat dan biaya untuk mendapatkannya. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya untuk mendapatkan informasi tersebut.

2.3.2 Siklus Informasi

Informasi berasal dari data yang belum diolah. Data diolah untuk menghasilkan informasi, penerima kemudian menerima informasi tersebut, membuat suatu keputusan dan melakukan tindakan, yang berarti menghasilkan suatu tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya membentuk suatu siklus.



Gambar 2. 1 Siklus Informasi

Sumber : Jogiyanto HM (2005 : 9)

2.3.3 Kualitas Informasi

Menurut Jogiyanto HM (2005 : 10) kualitas informasi *(quality of information)* sangat dipengaruhi atau ditentukan oleh hal sebagai berikut :

a. Relevan *(relevancy)*

   Seberapa jauh tingkat relevansi informasi tersebut terhadap kenyataan kejadian masa lalu, kejadian hari ini, dan kejadian yang akan datang. Informasi yang berkualitas akan mampu menunjukan benang merah relevansi kejadian masa lalu, hari ini, dan masa depan sebagai bentuk aktivitas yang kongkrit dan mampu dilaksanakan, dan dibuktikan oleh siapa saja.

b. Akurat *(accuracy)*

   Suatu sistem dikatakan berkualitas jika seluruh kebutuhan informasi tersebut telah tersampaikan *(completeness),* seluruh pesan telah benar atau sesuai *(correctness),* serta pesan yang disampaikan sudah lengkap atau hanya sistem yang diinginkan *user*.

c. Tepat waktu *(timeliness)*

   Berbagai proses dapat terselesaikan dengan tepat waktu, laporan yang dibutuhkan dapat disampaikan tepat waktu.

d. Ekonomis *(economy)*

   Informasi yang dihasilkan mempunyai daya jual yang tinggi, serta biaya operasional untuk menghasilkan informasi tersebut minimal, informasi tersebut juga mampu memberikan dampak yang luas terhadap laju pertumbuhan ekonomi dan teknologi informasi.

e. Efisien *(efficiency)*

Informasi yang berkualitas memiliki *sintaks* atau kalimat yang sederhana, namun mampu memberikan makna dan hasil yang mendalam, atau bahkan menggetarkan setiap orang atau benda apapun yang menerimanya.

f. Dapat dipercaya *(reability)*

Informasi tersebut berasal dari sumber yang dapat dipercaya. Sumber tersebut juga telah teruji tingkat kejujurannya.

2.4 Sistem Informasi

Sistem informasi mempunyai peranan yang sangat penting dalam membantu menyediakan, mengumpulkan, memproses, menyimpan dan mendistribusikan informasi untuk mendukung pembuatan keputusan dan pengawasan pada suatu perusahaan. Sistem informasi yang dibutuhkan adalah sistem yang memperlancar proses kegiatan yang sedang berjalan.

2.4.1 Pengertian Sistem Informasi

Definisi menurut Abdul Kadir (2003 : 10) sistem informasi adalah mencakup sejumlah komponen (manusia, komputer, teknologi informasi, dan prosedur kerja), ada sesuatu yang diproses (data menjadi informasi), dan dimaksudkan untuk mencapai suatu sasaran atau tujuan.

Menurut Alter (1992) sistem informasi adalah kombinasi antara prosedur kerja, informasi, oaring dan teknologi informasi yang diorganisasikan untuk mencapai tujuan dalam sebuah organisasi.Abdul Kadir (2003 : 11).

Berdasarkan definisi diatas, sistem informasi adalah suatu sistem yang terdiri dari komponen-komponen dalam organisasi dan teknologi informasi yang saling berinteraksi untuk melakukan suatu proses dan bekerja sama untuk memberikan informasi bagi pengambil keputusan serta untuk mencapai suatu tujuan.

2.4.2 Komponen Sistem Informasi

Dalam suatu sistem informasi terdapat komponen-komponen seperti dalam buku Abdul Kadir (2003 : 70) dapat diklasifikasikan sebagai berikut :

a. Perangkat keras *(hardware)*

   Mencakup peranti-peranti fisik seperti komputer dan printer

b. Perangkat lunak *(software)*

   Sekumpulan instruksi yang memungkinkan perangkat keras untuk dapat memproses data.

c. Prosedur

   Sekumpulan aturan yang dipakai untuk mewujudkan pemrosesan data dan pembangkitan keluaran yang dikehendaki.

d. Orang

   Semua pihak yang bertanggung jawab dalam pengembangan sistem informasi, pemrosesan dan penggunaan keluaran sistem informasi.

e. Basis data *(database)*

   Sekumpulan tabel, hubungan dan lain-lain yang berkaitan dengan penyimpanan data.

f. Jaringan komputer dan komunikasi data

Sistem penghubung yang memungkinkan sumber *(resources)* dipakai secara bersama atau diakses oleh sejumlah pemakai.



Gambar 2. 2 Komponen Sistem Informasi

Sumber : Abdul Kadir (2003:71)

Sedangkan dalam buku Jogiyanto HM (2005 : 13) komponen sistem informasi tersebut disebut dengan istilah blok bangunan (*building block*). Komponen sistem informasi tersebut terdiri dari blok masukan *(input block)*, blok model *(model block)*, blok keluaran *(output block)*, blok teknologi *(technology block)*, blok basis data *(database block)* dan blok kendali (*controls block).*

a. Blok masukan *(input block)*

Input blok memiliki data yang masuk ke dalam sistem informasi, juga metode-metode untuk menangkap data yang dimasukkan yang dapat berupa dokumen-dokumen dasar.

b. Blok model *(model block)*

Blok ini terdiri dari kombinasi prosedur logika dan model matematik yang memanipulasi data *input* dan data yang tersimpan di basis data.

c. Blok keluaran *(output block)*

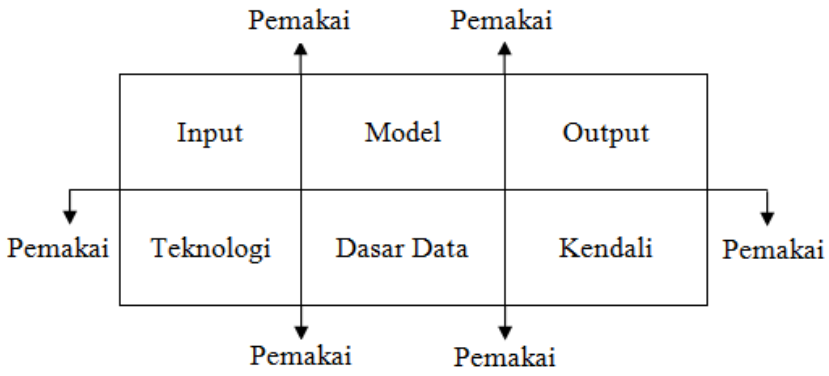
Produk dari sistem informasi adalah keluaran yang merupakan informasi berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai sistem.

d. Blok teknologi *(technology block)*

Blok teknologi digunakan untuk menerima *input*, menyimpan, mengakses data, menghasilkan dan mengirimkan keluaran dari sistem secara keseluruhan. Teknologi terdiri dari tiga bagian utama, yaitu teknisi (*humanware* atau *brainware*), perangkat lunak (*software*), dan perangkat keras (*hardware*).

e. Blok basis data *(database block)*

Basis data merupakan kumpulan data yang saling berhubungan satu dengan yang lainnya, tersimpan di perangkat keras komputer dan digunakan perangkat lunak (*software*) untuk memanipulasinya. Data perlu disimpan di dalam basis data untuk keperluan penyediaan informasi lebih lanjut. Data di dalam basis data perlu diorganisasikan sedemikian rupa, supaya informasi yang dihasilkan berkualitas. Basis data diakses atau dimanipulasi dengan menggunakan perangkat lunak yang disebut dengan DBMS (*Database Management Systems).*



Gambar 2. 3 Blok Sistem Informasi yang Berinteraksi

Sumber : Jogiyanto HM (2005 : 12)

2.4.3 Tujuan Pembangunan Sistem Informasi

Setiap pembangunan sistem informasi memiliki tujuan, berikut tujuan tersebut :

1. Integrasi sistem
    a. Menghubungkan sistem individu/kelompok
    b. Pengkolektifan data dan penyambungan secara otomatis
    c. Peningkatan koordinasi dan pencapaian sinergi
2. Efisiensi pengelolaan sistem
    a. Penggunaan basis data dalam upaya kesamaan pengadministrasian data
    b. Pengelolaan data berkaitan dengan karakteristik informasi
    c. Dukungan keputusan untuk manajemen
    d. Melengkapi informasi guna kebutuhan proses pengambilan keputusan
    e. Akuisisi informasi eksternal melalui jaringan kompter
    f. Ekstraksi dari informasi internal yang terpadu

2.4.4 Manfaat Sistem Informasi

Sistem informasi memiliki banyak manfaat, diantaranya :

1. Menghemat tenaga kerja
2. Peningkatan efisiensi
3. Mempercepat proses
4. Perbaikan dokumentasi
5. Pencapaian standar
6. Perbaikan keputusan

2.5 Pendekatan Berorientasi Objek

Pendekatan berorientasi objek merupakan 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.

2.5.1 Metode Berorientasi Objek

Metodologi berorientasi objek adalah suatu strategi pembangunan perangkat lunak yang mengorganisasikan perangkat lunak sebagai kumpulan objek yang berisi data dan informasi yang diberlakukan terhadapnya. Keuntungan menggunakan metode 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 dengan pola-pola yang mungkin sering berubah-ubah.

4. Adanya konsistensi

   Karena sifat pewarisan 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 mampu memenuhi kebutuhan pemakai serta mempunyai sedikit kesalahan.

2.5.2 Pemrograman Berorientasi Objek

Menurut Adi Nugroho (2005 : 4) pemrograman berorientasi objek atau sering juga disebut dengan OOP (*Object Oriented Programming*) adalah suatu cara baru dalam berpikir serta berlogika dalam menghadapi masalah-masalah yang akan dicoba atau diatasi dengan bantuan komputer. OOP mencoba melihat permasalahan lewat pengamatan dunia nyata dimana setiap objek adalah entitas tunggal yang memiliki kombinasi struktur data dan fungsi tertentu.

Objek adalah konsep atau abstraksi tentang sesuatu yang memiliki arti bagi aplikasi yang akan kita kembangkan. Objek biasanya adalah kata benda, bisa diraba dan dilihat secara kasat mata seperti mobil, pesawat terbang, sapi, kuda dan sebagainya. Pemrograman berorientasi objek dengan paradigma barunya, menjanjikan tingkat penggunaan ulang, kinerja yang lebih optimal dalam mengatasi kendala.

2.6 Pemodelan *Unified Modeling Language* (UML)

Menurut Adi Nugroho (2005 : 17) UML *(Unified Modeling Language)* merupakan bahasa untuk visualisasi, spesifikasi, konstruksi serta dokumentasi. Dalam kerangka visualisasi, para pengembang menggunakan UML sebagai suatu cara untuk mengkomunikasikan idenya kepada para pemrogram serta calon pengguna sistem/perangkat lunak. Dengan adanya bahasa yang bersifat standar, komunikasi perancang atau pemrogram serta calon pengguna diharapkan menjadi lancar. Dalam kerangka spesifikasi, UML menyediakan model-model yang tepat, tidak ambigu (makna ganda) serta lengkap. Secara khusus UML menspesifikasi

langkah-langkah penting dalam pengambilan keputusan analisis, perancangan serta implementasi dalam sistem yang sangat bernuasa perangkat lunak. UML bukanlah merupakan bahasa pemrograman tetapi model-model yang tercipta berhubungan langsung dengan berbagai bahasa pemrograman.

UML merupakan bahasa standar untuk penulisan *Blueprint Software* yang digunakan untuk Visualisasi (*Visualize*), Spesifikasi (*Specify*), Pembentukan (*Construct*) dan Pendokumentasian (*Documentation*) alat-alat dari sistem perangkat lunak.

Tujuan penggunaan UML adalah sebagai berikut:

1. Memodelkan suatu sistem (bukan hanya perangkat lunak) yang menggunakan konsep berorientasi objek.
2. Menciptakan suatu bahasa pemodelan yang dapat digunakan baik oleh manusia maupun mesin.

Keunggulan menggunakan UML dibandingkan menggunakan metodologi terstruktur:

1. *Uniformity*

   Pengembang cukup menggunakan satu metodologi dari tahap analisis hingga perancangan. Memungkinkan merancang komponen antarmuka secara terintegrasi bersama perancangan perangkat lunak dan perancangan struktur data.

2. *Understandability*

   Kode yang dihasilkan dapat diorganisasi ke dalam kelas-kelas yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah untuk dipahami.

3. *Stability*

   Kode program yang dihasilkan relatif stabil sepanjang waktu, karena mendekati permasalahan yang sesungguhnya.

4. *Reusability*

   Dengan metodologi berorientasi objek, dimungkinkan penggunaan ulang kode, sehingga pada akhirnya akan sangat mempercepat waktu pengembangan perangkat lunak atau sistem informasi.

2.7 Teori Alat Bantu Analisis dan Perancangan Berorientasi Objek

Adapun beberapa alat bantu yang digunakan pada tahap analisis dan perancangan digambarkan dalam bentuk sebagai berikut :

1. ***Use case* Diagram**

   Use case mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat. Secara kasar, *use case* digunakan untuk mengetahui fungsi apa saja yang ada didalam sebuah sistem informasi dan siapa saja yang berhak menggunakan fungsi-fungsi itu. Ada dua hal utama pada *use case* yaitu pendefinisian apa yang disebut aktor dan *use case*.

a. Aktor merupakan orang, proses, atau sistem lain yang berinteraksi dengan sistem informasi yang akan dibuat diluar sistem informasi yang akan dibuat itu sendiri, jadi walaupun simbol dari aktor adalah gambar orang, tapi aktor belum tentu merupakan orang.
b. *Use case* merupakan fungsionalitas yang disediakan sistem sebagai unit-unit yang saling bertukar pesan antar unit atau aktor.

2. **Skenario *Use Case***

   Mendeskripsikan aktor-aktor yang melakukan prosedur dalam sistem, serta menjelaskan respon yang ditanggapi oleh sistem tersebut terhadap prosedur yang dilakukan oleh aktor.

3. ***Activity* Diagram**

   Diagram aktivitas atau *activity diagram* menggambarkan *workflow* (aliran kerja) atau aktivitas dari sebuah sistem atau proses bisnis. Yang perlu diperhatikan disini bahwa diagram aktivitas menggambarkan aktivitas sistem bukan apa yang dilakukan aktor, jadi aktivitas yang dapat dilakukan oleh sistem.

4. ***Sequence* Diagram**

   Sequence diagram menggambarkan pola objek pada *use case* dengan mendeskripsikan waktu hidup objek atau *message* yang dikirimkan dan diterima antar objek. Oleh karena itu untuk menggambar diagram *sequence* maka harus diketahui objek-objek yang terlibat dalam sebuah *use case* beserta metode yang dimiliki kelas yang diinisiasi menjadi objek itu.

5. ***Collaboration* diagram**

   Collaboration diagram adalah bentuk lain *sequence* diagram, dimana *sequence* diagram diorganisir oleh waktu, sedangkan *collaboration* diagram diorganisir menurut ruang.

6. ***Class* Diagram**

   Diagram kelas atau *class diagram* menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem. Kelas memiliki apa yang disebut atribut dan metode atau operasi.

   a. Atribut merupakan variable-variabel yang dimiliki oleh suatu kelas.

   b. Operasi atau metode adalah fungsi-fungsi yang dimiliki oleh suatu kelas.

7. ***Component* Diagram**

   Component Diagram atau diagram komponen adalah diagram yang menunjukan model secara fisik komponen perangkat lunak dalam sistem dan hubungan antar mereka.

8. ***Deployment* Diagram**

   Deployment Diagram merupakan suatu tampilan atau pandangan kinerja dari sebuah sistem yang baru dengan perancangan data yang diambil dari beberapa objek.

2.8 Teori Metode Pengembangan Sistem

Metode pendekatan dan pengembangan sistem menggambarkan tahapan-tahapan dalam proses penelitian guna memecahkan masalah penelitian dari awal perencanaan hingga tercapainya tujuan penelitian dan pengembangan sistem.

2.8.1 Metode Pengembangan Sistem

Metodologi adalah kesatuan metode-metode, prosedur-prosedur, konsep-konsep pekerjaan, aturan-aturan, postulat-postulat yang digunakan oleh suatu ilmu pengetahuan, seni atau disiplin lainnya. Metode adalah suatu cara, teknik yang sistematik untuk mengerjakan sesuatu. Metodologi Pengembangan sistem berarti metode-metode, prosedur-prosedur, konsep-konsep pekerjaan, aturan-aturan dan postulat-postulat (kerangka pemikiran) yang akan digunakan untuk mengembangkan suatu sistem informasi.

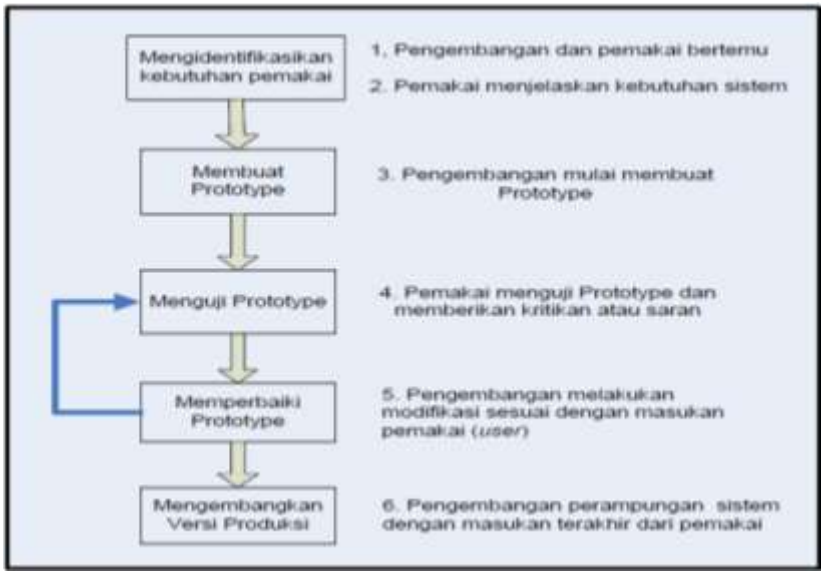
2.8.2 Jenis Model Pengembangan Sistem

Model pengembangan sistem yang akan dijelaskan adalah model pengembangan yang akan digunakan oleh penulis setelah dilakukan analisa dan penelitian jenis model apa yang nantinya akan diterapkan dalam pengembangan sistem informasi pada bengkel Ditech Injection. Berikut penjelasan masing-masing dari model pengembangan sistem tersebut :

1. *Prototype*

   Menurut Roger S. Pressman (2002:40) paradigma *prototyping* dimulai dengan pengumpulan kebutuhan. Pengembang dan pelanggan bertemu dan mendefinisikan obyektif keseluruhan keseluruhan dari perangkat lunak, mengidentifikasi segala kebutuhan yang diketahui dan area garis besar dimana definisi lebih jauh merupakan keharusan kemudian dilakukan “perancangan kilat”. Perancangan kilat berfokus pada penyajian dari aspek-aspek perangkat lunak tersebut yang akan nampak bagi pelanggan/pemakai (contohnya pendekatan *input* dan format *output*). Perancangan kilat membawa kepada

konstruksi sebuah *prototype*. *Prototype* tersebut dievaluasi oleh pengguna dan dipakai untuk menyaring kebutuhan pengembangan perangkat lunak. Iterasi terjadi pada saat *prototype* disetel untuk memenuhi kebutuhan pengguna dan pada saat yang sama memungkinkan pengembang untuk secara lebih baik memahami apa yang harus dilakukan. Secara ideal *prototype* berfungsi sebagai mekanisme untuk mengidentifikasi kebutuhan perangkat lunak. Bila *prototype* yang sedang bekerja dibangun, maka pengembang harus mempergunakan fragmen-fragmen program yang ada atau mengaplikasikan alat-alat bantu. Langkah umum paradigma *prototyping* adalah sebagai berikut :



Gambar 2. 4 Pengembangan model p*rototype*

Sumber : http://www.agungnugroho.net/20 Mei 2012

Adapun penjelasan dari gambar 2.4 adalah sebagai berikut :

a. Mengidentifikasikan kebutuhan pemakai.

   Pada tahap ini merupakan tahap awal dalam membangun sistem informasi. Pengembang akan melakukan studi kelayakan dan studi terhadap kebutuhan pengguna dan pengguna menjelaskan kebutuhan sistem yang akan dibangun oleh pengembang. Baik yang meliputi model *interface,* teknik prosedural maupun dalam teknologi yang akan digunakan.

b. Pembuatan *prototype*

   Setelah menganalisa sistem yang akan dikembangkan serta kebutuhan sistem untuk sistem yang akan dibangun, pengembang sistem mulai membuat *prototype*. Pembuatan ini meliputi perancangan sistem dan kemudian diimplementasikan dengan pembuatan *coding* yaitu menerjemahkan hasil rancangan ke dalam bentuk bahasa pemrograman yang akan menjadi sebuah sistem informasi yang diharapkan oleh pengguna.

c. Pengujian *prototype*

   Setelah tahap pembuatan *prototype* selesai,kemudian pengembang sistem dan pengguna melakukan pengujian perangkat lunak agar perangkt lunak dapat digunakan sesuai dengan kebutuhan, dan pengguna memberikan saran atau masukan bila terdapat kekurangan pada perangkat lunak.

d. Perbaikan *prototype*

   Pada tahap ini pengembang sistem melakukan perbaikan dan modifikasi sesuai dengan masukan dan saran dari pengguna.

e. Mengembangkan versi produksi

Pada tahap ini pengembang sistem menyelesaikan sistem yang telah dibuat sesuai dengan masukan dan saran terakhir dari pengguna sistem dan melakukan implementasi sistem.

Keunggulan *prototype* adalah:

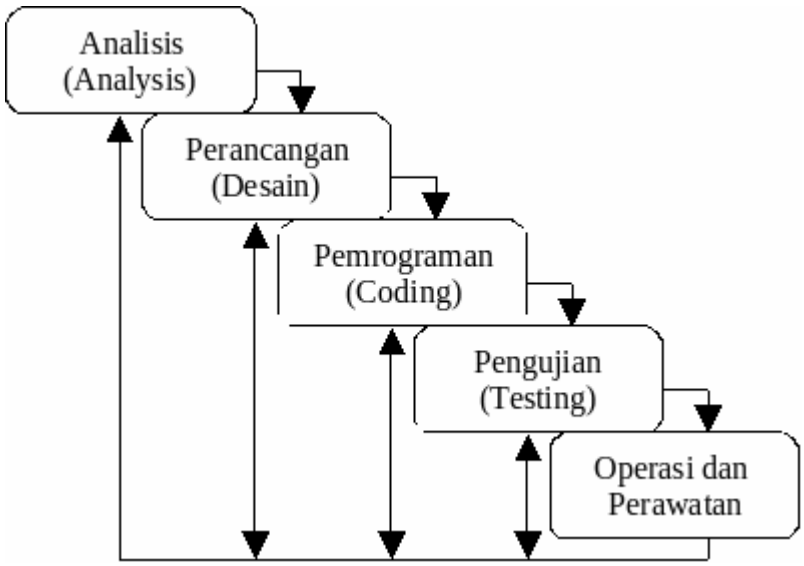
1. Adanya komunikasi yang baik antara pengembang dan pengguna akhir.
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pengguna akhir.
3. Pengguna akhir berperan aktif dalam pengembangan sistem
4. Lebih menghemat waktu dalam pengembangan sistem
5. Penerapan menjadi lebih mudah karena pengguna mengetahui apa yang diharapkannya.

Kelemahan *prototype* adalah :

1. Pengguna kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangka waktu lama.
2. Pengembang biasanya ingin cepat menyelesaikan proyek. Sehingga menggunakan algoritma dan bahasa pemrograman yang sederhana untuk membuat *prototyping* lebih cepat selesai tanpa memikirkan lebih lanjut bahwa program tersebut hanya merupakan cetak biru sistem .
3. Sistem yang dirancang akan baik hasilnya jika pengguna sungguh-sungguh meluangkan waktunya untuk menyelesaikan *prototype*.

2. *Waterfall*

Menurut Roger S Pressman (2011) Metode *Waterfall* adalah suatu proses pengembangan perangkat lunak berurutan, di mana kemajuan dipandang sebagai terus mengalir ke bawah seperti air terjun melewati fase-fase perencanaan, pemodelan, implementasi (konstruksi), dan pengujian. Berikut adalah gambar dari model waterfall :



Gambar 2. 5 Pengembangan model *waterfall*

Sumber : http://www.erisza.blog.ugm.ac.id/3 Juli 2013

Berikut penjelasan dari gambar 2.5 mengenai model *waterfall* :

a. Analisa Kebutuhan

Langkah ini merupakan analisa terhadap kebutuhan sistem. Pengumpulan data dalam tahap ini bisa melakukan sebuah penelitian, wawancara atau studi literatur. Sistem analis akan menggali informasi sebanyak-banyaknya dari *user* sehingga akan tercipta sebuah sistem komputer yang bisa melakukan tugas-tugas yang diinginkan oleh *user* tersebut. Tahapan ini akan

menghasilkan dokumen *user requirement* atau bisa dikatakan sebagai data yang berhubungan dengan keinginan *user* dalam pembuatan sistem. Dokumen ini lah yang akan menjadi acuan sistem analis untuk menerjemahkan ke dalam bahasa pemrogram.

b. Desain Sistem

Tahapan dimana dilakukan penuangan pikiran dan perancangan sistem terhadap solusi dari permasalahan yang ada dengan menggunakan perangkat pemodelan sistem seperti diagram alir data (*data flow diagram*), diagram hubungan entitas (*entity relationship diagram*) serta struktur dan bahasan data.

c. Penulisan Kode Program

Penulisan kode program merupakan penerjemahan desain dalam bahasa yang bisa dikenali oleh komputer. Dilakukan oleh programmer yang akan meterjemahkan transaksi yang diminta oleh user. Tahapan ini lah yang merupakan tahapan secara nyata dalam mengerjakan suatu sistem. Dalam artian penggunaan komputer akan dimaksimalkan dalam tahapan ini. Setelah pengkodean selesai maka akan dilakukan testing terhadap sistem yang telah dibuat tadi. Tujuan testing adalah menemukan kesalahan-kesalahan terhadap sistem tersebut dan kemudian bisa diperbaiki.

d. Pengujian Program

Tahapan akhir dimana sistem yang baru diuji kemampuan dan keefektifannya sehingga didapatkan kekurangan dan kelemahan sistem yang

kemudian dilakukan pengkajian ulang dan perbaikan terhadap aplikasi menjadi lebih baik dan sempurna.

e. Penerapan Program dan Pemeliharaan

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (*peripheral* atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional.

Kelebihan *waterfall* :

1. Pencerminan kepraktisan rekayasa, yang membuat kualitas *software* tetap terjaga karena pengembangannya yang terstruktur dan terawasi.
2. Proses pemeliharaan dapat dilakukan dengan mudah. Akan tetapi dikarenakan dokumentasi yang lengkap dan sangat teknis, membuat pihak klien sulit membaca dokumen yang berujung pada sulitnya komunikasi antar pengembang dan klien.
3. Dokumentasi kode program yang lengkap juga secara tak langsung menghapus ketergantungan pengembang terhadap pemrogram yang keluar dari tim pengembang.

Kelemahan *waterfall* :

1. Lambatnya proses pengembangan perangkat lunak. Dikarenakan prosesnya yang satu persatu dan tidak bisa diloncat-loncat menjadikan model klasik ini sangat memakan waktu dalam pengembangannya. Disisi lain, pihak klien

tidak dapat mencoba sistem sebelum sistem benar-benar selesai pembuatannya.

2. Kinerja personil yang tidak optimal dan tidak efisien karena terdapat proses menunggu suatu tahapan selesai terlebih dahulu.
3. Secara keseluruhan model pendekatan pengembangan software dengan metode *waterfall* cocok untuk pengembangan *software* / perangkat lunak dengan tingkat resiko yang kecil, dan memiliki ukuran yang kecil serta waktu pengembangan yang cukup panjang. Model ini tidak disarankan untuk ukuran perangkat lunak yang besar dan tingkat resiko yang besar.
4. Pengguna akhir sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.
5. Pemakai harus bersabar karena versi dari program tidak akan didapat sampai akhir rentang waktu proyek.
6. Adanya waktu menganggur bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.

2.9 Pengertian Teori Terkait

Dalam hal ini, penulis mendefinisikan arti kata dari judul skripsi yang diambil dari judul "Rancang Bangun Sistem Informasi Penjualan Pembelian Suku Cadang dan Pelayanan Perbaikan Mobil Pada Bengkel Ditech Injection". Pelayanan di sini yaitu perbaikan mobil, penjualan pembelian serta persediaan suku cadang yang ditujukan kepada pelanggan Ditech Injection. Dengan adanya

definisi-definisi ini diharapkan penulis lebih mengerti dan memahami maksud penelitian ini.

2.9.1 Pengertian Penjualan

Penjualan dapat diartikan sebagai usaha atau langkah kongkrit yang dilakukan untuk memindahkan suatu produk, baik itu barang ataupu jasa, dari produsen kepada konsumen sebagai sasarannya. Tujuan utama penjualan yaitu mendatangkan keuntungan atau laba dari produk ataupun barang yang dihasikan produsennya dengan pengelolaan yang baik.

2.9.2 Pengertian Pembelian

Pembelian merupakan salah satu tindakan untuk memenuhi kebutuhan. Maka, pembelian ini adalah suatu proses yang sangat penting yang akan menunjang pada tercapainya keuntungan yang maksimal. Tujuan utamanya adalah memperoleh bahan dengan biaya serendah mungkin yang konsisten dengan kualitas dan jasa yang dipersyaratkan.

2.9.3 Pengertian Pelayanan

Pelayanan dapat diartikan sebagai sebuah kegiatan atau keuntungan yang dapat ditawarkan oleh satu pihak kepada pihak lain. Pelayanan pada dasarnya bersifat *intangible* (tak teraba) dan tidak berujung pada kepemilikan. Produk pelayanan bisa berkaitan dengan produk fisik, bisa juga tidak. Memotong rambut, memperbaki televisi, menyewa kamar hotel, memperbaiki kendaraan adalah contoh kegiatan yang berhubungan dengan pelayanan.

2.9.4 Pengertian Bengkel

Dalam kamus besar bahasa Indonesia bengkel adalah suatu tempat dimana dilakukan perbaikan-perbaikan yang bersifat teknis terhadap suatu produk yang dalam konteks materi ini, produk yang dimaksud adalah kendaraan bermotor.

2.9.5 Pengertian Suku Cadang

Suku cadang atau orang awam menyebut onderdil adalah adalah merupakan komponen atau barang pengganti yang sangat diperlukan pada mesin yang mengalami kerusakan. Jadi suku cadang merupakan faktor pelengkap untuk menjamin kesiapan mesin atau peralatan yang dapat beroperasi lagi setelah mengalami perbaikan.

2.10 Perangkat Lunak Pendukung

Untuk membuat sistem informasi yang terkomputerisasi tentu memerlukan perangkat lunak, yang berfungsi sebagai pendukung pembuatan sistem informasi tersebut. Dibawah ini akan dijelaskan perangkat lunak pendukung yang digunakan penulis.

2.10.1 Bahasa Pemrograman Java

Java merupakan perangkat lunak produksi Sun Microsistem Inc. untuk pemrograman beberapa tujuan (*multi purpose*), dapat berjalan di beberapa sistem operasi (*multiplatform*), mudah dipelajari dan *powerfull*. Aplikasi-aplikasi yang dapat dibuat dengan Java, meliputi pemrograman web (*web programming*), pemrograman desktop (*desktop programming*), pemrograman *mobile*/*handphone* (*mobile programming*).

Sun Microsystem sendiri mendeskripsikan Java sebagai bahasa pemrograman yang sederhana, *garbage collected*, *robust* dan dapat diperluas. Sifat yang dideskripsikan oleh Sun Microsistem dapat dipaparkan sebagai berikut:

1. Sederhana

   Pengembang Java banyak membuang fitur-fitur yang tidak diperlukan seperti yang dimiliki bahasa pemrograman tingkat tinggi lainnya.

2. *Garbage collected* (pembuang sampah)

   Program Java menyingkirkan sendiri sampah-sampah yang tidak berguna, artinya program tidak perlu menghapus objek-objek yang dialokasikannya di memori.

3. *Robust* (tangguh)

   Karena interpretasi Java memeriksa seluruh akses sistem yang dilakukan program, maka program Java tidak akan membuat sistem menjadi *crash*.

4. Dapat diperluas

   Program Java mendukung metode *native* yakni fungsi-fungsi yang ditulis dalam bahasa latin, biasanya C++.

Java membagi versi programnya ke dalam tiga kelompok besar, yaitu :

1. Java 2 *Standar Edition* (J2SE) untuk konsentrasi pada PC
2. Java 2 *Enterprise Edition* (J2EE) untuk konsentrasi pada aplikasi server besar
3. Java 2 *Micro Edition* (J2ME) untuk konsentrasi pada *mobile*

2.10.2 NetBeans

Pengertian NetBeans adalah *Integrated Development Environment* (IDE) berbasiskan Java dari Sun Microsistems yang berjalan di atas *Swing*. *Swing* sebuah teknologi Java untuk pengembangan aplikasi desktop yang dapat bejalan di berbagai macam platforms seperti Windows, Linux, Mac OS X and Solaris.

Suatu IDE adalah lingkup pemrograman yang diintegrasikan kedalam suatu aplikasi perangkat lunak yang menyediakan pembangun *Graphic User Interface* (GUI), suatu text atau kode editor, suatu *compiler* atau *interpreter* dan suatu *debugger*.

Netbeans merupakan *software* development yang *open source*, dengan kata lain *software* ini di bawah pengembangan bersama, bebas biaya. NetBeans merupakan sebuah proyek kode terbuka yang sukses dengan pengguna yang sangat luas, komunitas yang terus tumbuh dan memiliki hampir 100 mitra. Sun Microsistem mendirikan proyek kode terbuka NetBeans pada bulan Juni 2000 dan terus menjadi sponsor utama.

The NetBeans IDE adalah sebuah lingkungan pengembangan - sebuah alat untuk programmer menulis, mengompilasi, mencari kesalahan dan menyebarkan program. Netbeans IDE ditulis dalam Java - namun dapat mendukung bahasa pemrograman lain. Terdapat banyak modul untuk memperluas Netbeans IDE. Netbeans IDE adalah sebuah produk bebas dengan tanpa batasan bagaimana digunakan.

2.10.3 MySQL

MySQL (*My Structurd Query Language*) adalah sebuah program pembuat dan pengelola *database* atau yang sering disebut dengan DBMS (*Database Management Sistem*), sifat dari DBMS ini adalah o*pen source*, yang bisa didapatkan secara gratis.

MySQL sebenarnya produk yang berjalan pada platform linux, dengan adanya perkembangan dan banyaknya pengguna, serta lisensi dari *database* ini adalah o*pen source*, maka para pengembang kemudian merilis versi Windows.

Selain itu MySQL juga merupakan program pengakses *datasase* yang bersifat jaringan, sehingga dapat digunakan untuk aplikasi *multiuser* (banyak pengguna). Kelebihan lain dari MySQL adalah menggunakan bahasa *query* (permintaan) standar SQL (*Standart Query Language*). SQL adalah suatu bahasa permintaan yang terstruktur, SQL telah distandarkan untuk semua program pengakses *database* seperti Oracle, PosgreSql, Sql Server dll.

Sebagai sebuah program penghasil *database*, MySQL tidak mungkin berjalan sendiri tanpa adanya sebuah aplikasi pengguna atau *interface* yang berguna sebagai program aplikasi pengakses *database* yang dihasilkan. MySQL dapat didukung hampir semua program aplikasi baik *open source* seperti php maupun yang tidak *open source* yang ada pada platform Windows seperti Visual Basic dan Delphi.

2.10.4 XAMPP

XAMPP merupakan paket PHP yang berbasis *open source* yang dikembangkan oleh sebuah komunitas *open source*. Dengan menggunakan

XAMPP kita tidak usah lagi bingung melakukan penginstallan program-program yang lain, karena semua kebutuhan sudah disediakan oleh XAMPP. Berikut adalah beberapa paket yang telah disediakan :

1. Apache HTTPD
2. MySQL
3. PHP
4. File Zilla FTP Server
5. PHP Myadmin

2.11 Arsitektur Jaringan

Untuk 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.

2.11.1 Pengertian Jaringan Komputer

Menurut Dede Sopandi (2008 : 2) jaringan komputer (*computer network)* adalah merupakan gabungan antara teknologi komputer dan teknologi telekomunikasi. Gabungan teknologi ini melahirkan pengolahan data yang dapat didistribusikan, mencakup pemakaian *database*, *software* aplikasi dan peralatan *hardware* secara bersamaan, sehingga penggunaan komputer yang sebelumnya hanya berdiri sendiri, kini dapat dilakukan dengan sekumpulan komputer yang terpisah-pisah tetapi saling berhubungan dalam melaksanakan tugasnya.

2.11.2 Jenis Jaringan Komputer

Jaringan komputer dapat dibedakan berdasarkan jenis areanya. Ada 3 jenis yaitu:

1. *Local Area Network* (LAN)

   Local Area Network adalah jaringan lokal yang dibuat pada area tertutup. Misalkan dalam satu gedung atau satu ruangan. Terkadang jaringan lokal disebut juga jaringan privat. LAN biasa digunakan untuk jaringan kecil yang menggunakan *resource* bersama-sama, seperti penggunaan *printer* secara bersama-sama, penggunaan media penyimpanan secara bersama.

2. *Metropolitan Area Network* (MAN)

   Metropolitan Area Network menggunakan metode yang sama dengan LAN namun daerah cakupannya lebih luas. Daerah cakupan MAN bisa satu RW, beberapa kantor yang berada dalam komplek yang sama, satu kota, bahkan satu provinsi. Dapat dikatakan MAN merupakan pengembangan dari LAN.

3. *Wide Area Network* (WAN)

   Wide Area Network cakupannya lebih luas dari MAN. Cakupan WAN meliputi satu kawasan, satu negara, satu pulau bahkan satu benua. Metode yang digunakan WAN hampir sama yang digunakan dengan LAN dan MAN.

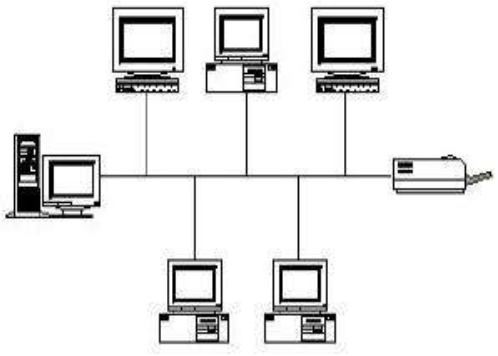
2.11.3 Topologi Jaringan

Topologi adalah suatu aturan bagaimana mengubungkan komputer (*node*) satu sama lain secara fisik dan pola hubungan antara komponen-komponen yang berkomunikasi melalui media dan peralatan jaringan, seperti : server, *workstation.* hub/switch dan pengkabelannya.

Ada dua jenis topologi, yaitu *physical topology* (topologi fisik) dan *logical topologi* (topologi logika). Topologi fisik berkaitan dengan *layout* atau bentuk jaringan, seperti bagaimana memilih perangkat dan melakukan instalasi perangkat jaringan. Sedangkan topologi logika berkaitan dengan bagaimana data mengalir di dalam topologi fisik. Jika topologi fisik bagaikan tubuh maka topologi logika diibaratkan sebagai darah yang mengalir di dalam tubuh. Berikut jenis-jenis topologi :

1. **Topologi Bus**

   Topologi bus sering juga disebut *ethernet bus topologies*. Topologi bus menggunakan perangkat jaringan atau *network interface card* (NIC) bernama *ethernet.* Jaringan yang menggunakan topologi bus dapat dikenali dengan penggunaan sebuah kabel *backbone* (kabel utama) yang menghubungkan semua peralatan jaringan (*device*). Karena kabel *backbone* menjadi satu-satunya jalan bagi lalu lintas data maka apabila kabel *backbone* rusak atau terputus maka akan menyebabkan jaringan mati total.

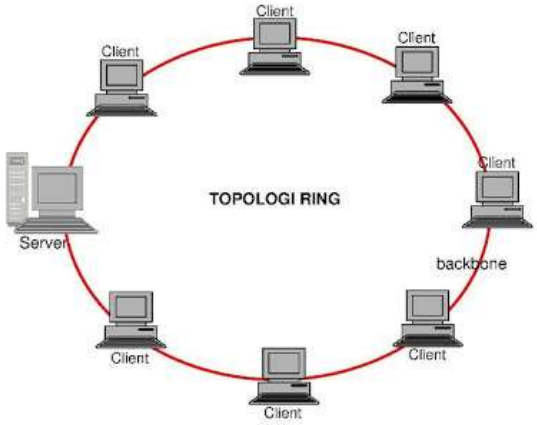


Gambar 2. 6 Topologi Bus

Sumber : Dede Sopandi (2008 : 29)

2. **Topologi Ring**

Topologi ring sangat berbeda dengan topologi bus. Sesuai dengan namanya, jaringan yang menggunakan topologi ini dapat dikenali dari kabel *backbone* yang membentuk cincin. Setiap kali komputer terhubung dengan kabel *backbone*. Setelah sampai pada komputer terakhir maka ujung kabel akan kembali dihubungkan dengan komputer pertama.

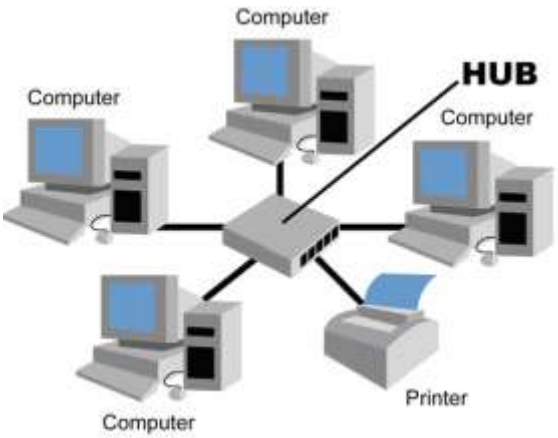


Gambar 2. 7 Topologi Ring

Sumber : Dede Sopandi (2008 : 30)

3. **Topologi Star**

Topologi star dikenali dengan keberadaan sebuah sentral berupa *hub/switch* yang menghubngkan semua *node*. Setiap *node* menggunakan sebuah kabel UTP atau STP yang dihubungkan dari *ethernet card* ke *hub*. Banyak sekali jaringan rumah, sekolah, pertokoan, laboratorium, dan kantor yang menggunakan topologi ini.

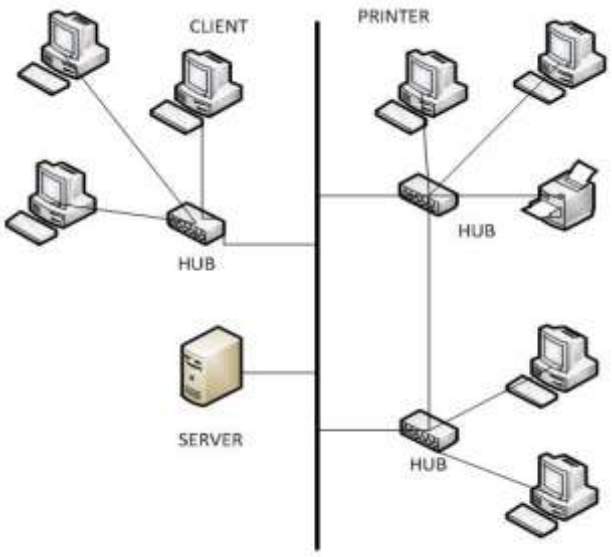


Gambar 2. 8 Topologi Star

Sumber : Dede Sopandi (2008 : 31)

4. **Topologi *Hybrid***

   Topologi *tree* disebut juga topologi *star-bus atau star/bus hybrid*. Topologi *tree* merupakan gabungan beberapa topologi *star* yang dihubungkan dengan topologi *bus*. Topologi *tree* digunakan untuk menghubungkan beberapa LAN dengan LAN lain. Hubungan antar LAN dilakukan via *hub*. Masing-masing *hub* dapat dianggap sebagai akar (*root*) dari masing-masing pohon (*tree*). Topologi *tree* dapat mengatasi kekurangan topologi *bus* yang disebabkan persoalan *broadcast traffic*, dan kekurangan topologi *star* yang disebabkan oleh keterbatasan kapasitas *port hub*.



Gambar 2. 9 Topologi Tree

Sumber : Dede Sopandi (2008 : 32)

2.11.4 Manfaat Jaringan Komputer

Dalam membangun jaringan komputer di perusahaan ada beberapa keuntungan yang dapat diperoleh, yaitu :

1. *Sharing*

   Dapat menggunakan sumber daya yang ada secara bersama-sama. Misal seorang pengguna yang berada 100 km jauhnya dari suatu data, tidak mendapatkan kesulitan dalam menggunakan data tersebut, seolah-olah data tersebut berada didekatnya. Hal ini sering diartikan bahwa jaringan komputer mengatasi masalah jarak.

2. Reliabilitas tinggi

   Dengan jaringan komputer kita akan mendapatkan reliabilitas yang tinggi dengan memiliki sumber-sumber alternatif persediaan. Misalnya, semua file dapat disimpan atau disalin ke semua komputer yang terkoneksi ke jaringan.

Sehingga bila salah satu komputer rusak, maka salinan di komputer yang lain bisa digunakan.

3. Menghemat uang

   Komputer berukuran kecil mempunyai rasio harga dan kinerja yang lebih baik dibandingkan dengan komputer yang besar. Komputer besar seperti mainframe memiliki kecepatan kira-kira sepuluh kali lipat kecepatan komputer kecil/pribadi.

2.11.5 Pengertian Klien Server

Dalam model klien/server, sebuah aplikasi dibagi menjadi dua bagian yang terpisah, tapi masih merupakan sebuah kesatuan yakni komponen klien dan komponen server. Komponen klien juga sering disebut sebagai *front-end*, sementara komponen server disebut sebagai *back-end*.

Komponen klien dari aplikasi tersebut dijalankan dalam sebuah *workstation* dan menerima masukan data dari pengguna. Komponen klien tersebut akan menyiapkan data yang dimasukkan oleh pengguna dengan menggunakan teknologi pemrosesan tertentu dan mengirimkannya kepada komponen server yang dijalankan di atas mesin server, umumnya dalam bentuk *request* terhadap beberapa layanan yang dimiliki oleh server. Komponen server akan menerima request dari klien, dan langsung memprosesnya dan mengembalikan hasil pemrosesan tersebut kepada klien. Klien pun menerima informasi hasil pemrosesan data yang dilakukan server dan menampilkannya kepada pengguna, dengan menggunakan aplikasi yang berinteraksi dengan pengguna.