**BAB II**

**LANDASAN TEORI**

* 1. **Sistem**
     1. **Pengertian sistem**

Terdapat dua kelompok pendekatan dalam mendefinisikan sistem, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Sistem dapat diartikan sebagai kumpulan dari beberapa komponen yang saling berhubungan satu dengan yang lainnya membentuk satu kesatuan untuk mencapai tujuan tertentu (Jogiyanto, 2009).

Pendapat lain mengatakan bahwa, sistem dapat diartikan sebagai serangkaian komponen-komponen yang saling berinteraksi dan bekerja sama untuk mencapai tujuan tertentu. Dan didalam sistem terkandung tiga elemen penting, yaitu rangkaian komponen, interaksi dan kerja sama dan yang terakhir adalah tujuan (Soeherman, 2008).

* + 1. **Elemen Sistem**

Menurut Sigit (1999) bahwa sistem memiliki komponen-komponen diantaranya : Penghubung sistem, batasan sistem lingkungan luar, masukan, keluaran, dan tujuan. Menurut Budiarti (1999) menyatakan bahwa elemen sistem adalah bagian tang terkecil yang teridentifikasi, ini merupakan penyusunan dari sistem.

* + 1. **Karakteristik Sistem**

Suatu sistem mempunyai karakteristik atau sifat-sifat yang tertentu, yaitu mempunyai komponen, batas sistem, lingkungan luar sistem, penghubung, masukan, keluaran, tujuan (Jogiyanto, 1999:3). Adapun pengertian dari masing-masing karakteristik Sistem tersebut adalah sebagai berikut :

1. Komponen Sistem

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

1. Batasan Sistem

Batasan sistem (boundary) merupakan daerah yang membatasi antara suatu dengan Sistem yang lainnya atau dengan lingkungan luarnya.

1. Lingkunagan Luar Sistem

Lingkungan luar sistem (envronment) dari suatu sistem adalah apapun diluar batas dari sistem yang mempengaruhi oprerasi sistem.

1. Penghubung Sistem

Penghubung (interface) merupakan media penghubung antara satu sub sistem dengan sub sistem yang lainya.

1. Masukan Sistem

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

1. Keluaran Sistem

Keluaran (output) adalah hasil dari energi yang diolah dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan.

1. Pengolahan Sistem

Suatu sistem dapat mempunyai suatu bagian pengolahan yang akan merubah masukan menjadi keluaran.

1. Sasaran atau Tujuan Sistem

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

* 1. **Informasi**
     1. **Pengertian informasi**

Informasi merupakan hasil pengolahan data sehingga menjadi bentuk yang penting bagi penerimanya dan mempunyai kegunaan sebagai dasar dalam pengambilan keputusan yang dapat dirasakan akibatnya secara langsung saat itu juga atau secara tidak langsung pada saat mendatang (Sutanta,2004)

Informasi menjadi penting, karena berdasarkan informasi itu para pengelola dapat mengetahui kondisi obyektif sebuah perusahaan (Sutedjo, 2002),. Tidak semua informasi berkualitas.

* + 1. **Kualitas informasi**

Kualitas informasi itu ditentukan oleh :

1. Keakuratan dan teruji kebenarannya

Informasi harus bebas dari kesalahan-kesalahan, tidak bias, dan tidak menyesatkan. Kesalahan-kesalahan itu dapat berupa kesalahan perhitungan maupun akibat gangguan (noise) yang dapat mengubah dan merusak informasi.

1. Kesempurnaan informasi

Untuk mendukung faktor pertama di atas, maka kesempurnaan informasi menjadi faktor penting, dimana informasi disajikan lengkap tanpa pengurangan, penambahan atau pengubahan.

1. Tepat waktu

Informasi harus disajikan secara tepat waktu, mengingat informasi akan menjadi dasar pengambilan keputusan. Keterlambatan informasi akan mengakibatkan kekeliruan dalam pengambilan keputusan.

1. Relevansi

Informasi akan memiliki nilai manfaat yang tinggi, jika informasi tersebut diterima oleh mereka yang membutuhkan, dan menjadi tidak berguna jika diberikan kepada mereka yang tidak membutuhkan.

1. Mudah dan murah

Saat ini, cara dan biaya untuk memperoleh informasi juga menjadi bahan pertimbangan tersendiri. Bilamana cara dan biaya untuk memperoleh informasi sulit dan mahal, maka orang menjadi tidak berminat untuk memperolehnya atau mencari alternatife subsitusinya. Biaya mahal yang dimaksud disini, jika bobot informasi tidak sebanding dengan biaya yang dikeluarkan. Dan melalui teknologi internet, saat ini ruang atau perusahaan dapat memperoleh informasi dengan mudah dan murah.

* + 1. **Nilai informasi**

Menurut Amsyah (2001) nilai Informasi ,ditentukan Oleh lima Karakteristik yaitu:

* 1. Ketelitian

Perbandingan dari informasi yang benar dengan jumlah seluruh informasi yang dihasilkan pada satu proses pengolah data tertentu.

* 1. Ketepatan Waktu

Informasi yang terlambat tidak akan berguna walaupun informasi itu akurat karena keterlambatan membuat informasi sudah tidak berguna lagi.

* 1. Kelengkapan

Informasi yang kurang lengkap akan mengakibatkan ketertundaan pengambilan keputusan.

* 1. Ringkas

Informasi sangat bernilai jika disajikan dengan ringkas dan langsung ke sasaran yang diperlukan, tidak bertele–tele dan berlebihan.

* 1. Kesesuaian

Informasi bernilai tinggi harus sesuai dengan keperluaan pekerjaan atau keperluan manajemen.

* 1. **Sistem Informasi**

Pengertian sistem informasi yaitu, sebagai berikut:

* + 1. Sistem informasi dalam suatu pemahaman yang sederhana dapat didefinisikan sebagai satu sistem berbasis komputer yang menyediakan informasi bagi beberapa pemakai dengan kebutuhan yang serupa (Sutono, 2007).
    2. Sistem informasi adalah kombinasi dari people, hardware, software, jaringan komunikasi, sumber-sumber data, prosedur dan kebijakan yang terorganisasi dengan baik yang dapat menyimpan, mengadakan lagi, menyimpan, dan menyebarluaskan informasi dalam suatu organisasi (O’Brien dan Marakas, 2009).
    3. Menurut Effendy (1989:11), sistem informasi adalah kombinasi antara prosedur kerja, informasi, orang, dan teknologi informasi yang diorganisasikan untuk mencapai tujuandalam sebuah organisasi.
    4. Menurut Wilkinson (1993), sistem informasi adalah kerangka kerja yang mengkoordinasikan sumber daya (manusia, komputer) untuk mengubah masukan.

Sistem informasi memuat berbagai informasi penting mengenai orang, tempat, dan segala sesuatu yang ada di dalam atau di lingkungan sekitar organisasi. Informasi sendiri mengandung suatu arti yaitu data yang telah diolah ke dalam suatu bentuk yang lebih memiliki arti dan dapat digunakan untuk pengambilan keputusan. Data sendiri merupakan fakta-faktayang mewakili suatu keadaan, kondisi, atau peristiwa yang terjadi atau ada di dalam atau di lingkungan fisik organisasi. Data tidak dapat langsung digunakan untuk pengambilan keputusan, melainkan harus diolah lebih dahulu agar dapat dipahami, lalu dimanfaatkan dalam pengambilan keputusan.

Sistem informasi terdiri dari komponen-komponen, sebagai berikut:

* + 1. *Input*

Input adalah semua data yang dimasukkan didalam sistem informasi, seperti dokumen, formulir dan file.

* + 1. Proses

Proses merupakan kumpulan prosedur yang akan memanipulasi input yang kemudian akan disimpan dalam basis data dan seterusnya akan diolah menjadi output yang akan digantikan oleh penerima.

* + 1. *Output*

Output merupakan semua keluaran atau hasil dari model yang sudah diolah menjadi suatu sistem informasi yang berguna dan dapat dipakai oleh penerima. Komponen ini dapat berupa laporan-laporan yang dapat dibutuhkan oleh pemakai sistem untuk memantau sistem suatu organisasi.

* + 1. Teknologi

Teknologi merupakan komponen yang berfungsi untuk memasukkan input, mengolah input dan menghasilkan output. Teknologi meliputi 3 bagian yaitu perangkat keras, perangkat lunak, perangkat manusia.

* + 1. Basis data

Basis data merupakan kumpulan data-data yang saling berhubungan dengan yang lain, yang disimpan dalam parangkat keras komputer dan akan diolah oleh parangkat lunak.

* + 1. Kendali

Kendali dalam hal ini merupakan semua tindakan yang diambil untuk menjaga semua sistem informasi tersebut agar dapat berjalan dengan lancar dan tidak mengalami gangguan.

* 1. **Gaji**

Gaji adalah pembayaran atas penyerahan jasa yang dilakukan oleh karyawan baik yang mempunyai jabatan maupun karyawan pelaksana (Mulyadi, 2001). Penghasilan yang didapat oleh seorang karyawan terdiri atas (Mulyadi, 2001):

1. Gaji Pokok

Besarnya gaji yang diberikan kepada karyawan sesuai dengan jabatan dan jasa yang diberikan pada perusahaan dan telah ditetapkan gaji pokok minimum pada waktu karyawan tersebut pertama kali bekerja.

1. *Insentive*
2. Uang makan dan transport

Merupakan tambahan yang akan diterima karyawan selain dari gaji pokoknya dan dihitung berdasarkan dari tingkat dan jabatannya sesuai dengan keahliannya dengan cara perhitungannya adalah perhari namun diberikan pada setiap menerima gaji.

1. Uang lembur

Menurut KEPMEN Pasal 1 Ayat 1 (2004:1), waktu kerja lembur adalah waktu kerja yang melebihi 7 (tujuh) jam sehari dan 40 (empat puluh) jam 1 (satu) minggu untuk 6 (enam) hari kerja dalam 1 (satu) minggu atau 8 (delapan) jam sehari dan 40 (empat puluh) jam 1 (satu) minggu untuk 5 (lima) hari kerja dalam 1 (satu) minggu atau waktu kerja pada hari istirahat mingguan dan atau pada hari libur resmi yang ditetapkan.

Menurut KEPMEN Pasal 3 Ayat 1 (2004:2), waktu kerja lembur hanya dapat dilakukan paling banyak 3 (tiga) jam dalam 1 (satu) hari dan 14 (empat belas) jam dalam 1 (satu) minggu.

1. Tunjangan Hari Raya (THR)

Menurut Menteri Tenaga Kerja dan Transmigrasi Republik Indonesia Pasal 2 (1994:2):

* 1. Pengusaha wajib memberikan THR kepada pekerja yang telah mempunyai masa kerja 3 bulan secara terus menerus atau lebih;
  2. THR sebagaimana dimaksud dalam ayat (1) diberikan satu kali dalam setahun.
  3. **Black box testing**

Black Box Testing merupakan pengujian yang berfokus pada spesifikasi fungsional dari perangkat lunak, tester dapat mendefinisikan kumpulan kondisi input dan melakukan pengetesan pada spesifikasi fungsional program (Janner, 2010).

Klasifikasi black-box testing mencakup beberapa pengujian, yaitu :

1. Pengujian fungsional (*Fungsional testing*)

Pada jenis pengujian ini, perangkat lunak diuji untuk persyaratan fungsional.Pengujian dilakukan dalam bentuk tertulis untuk memeriksa apakah aplikasi berjalan seperti yang diharapkan.

1. Pengujian tegangan (*Stress testing*)

Pengujian tegangan berkaitan dengan kualitas aplikasi di dalam lingkungan.Idenya adalah untuk menciptakan sebuah lingkungan yang menuntut aplikasi, tidak seperti saat aplikasi dijalankan pada beban kerja normal. Pengujian ini adalah hal yang paling sulit, cukup kompleks dilakukan, dan memerlukan upaya bersama dari semua tim.

1. Pengujian beban (*Load testing*)

Pada pengujian beban, aplikasi akan diuji dengan beban berat atau masukan, seperti yang terjadi pada pengujian situs web, untuk mengetahui apakah aplikasi/situs gagal atau kinerjanya menurun. Pengujian beban beroperasi pada tingkat beban standar, biasanya beban tertinggi akan diberikan ketika sistem dapat menerima dan tetap berfungsi dengan baik.

1. Pengujian khusus (*Ad-hoc testing*)

Jenis pengujian ini dilakukan tanpa penciptaan rencana pengujian (test plan) atau kasus pengujian (test case).Pengujian khusus membantu dalam menentukan lingkup dan durasi dari berbagai pengujian lainnya dan juga membantu para penguji dalam mempelajari aplikasi sebelum memulai pengujian dengan pengujian lainnya. Pengujian khusus dapat menemukan lubang-lubang dalam pengujian strategi dan dapat mengekspos hubungan di antara subsistem lain yang tidak jelas.

1. Pengujian penyelidikan (*Exploratory testing*)

Pengujian penyelidikan mirip dengan pengujian khusus dan dilakukan untuk mempelajari/mencari aplikasi.Pengujian penyelidikan perangkat lunak ini merupakan pendekatan yang menyenangkan untuk pengujian.

1. Pengujian usabilitas (*Usability testing*)

Pengujian ini disebut juga sebagai pengujian untuk keakraban pengguna (testing for user-friendliness).Pengujian ini dilakukan jika antarmuka pengguna dari aplikasinya penting dan harus spesifik untuk jenis pengguna tertentu. Pengujian usabilitas adalah proses yang bekerja dengan pengguna akhir secara langsung maupun tidak langsung untuk menilai bagaimana pengguna merasakan paket perangkat lunak dan bagaimana mereka berinteraksi dengannya.

1. Pengujian asap (*Smoke testing*)

Jenis pengujian ini disebut juga pengujian kenormalan (sanity testing).Pengujian ini dilakukan untuk memeriksa apakah aplikasi tersebut sudah siap untuk pengujian yang lebih besar dan bekerja dengan baik tanpa cela sampai tingkat yang paling diharapkan.Pada sebuah pengujian baru atau perbaikan peralatan yang terpasang, jika aplikasi “berasap”, aplikasi tersebut tidak bekerja.

1. Pengujian pemulihan (*Recovery testing*)

Pengujian pemulihan (recovery testing) pada dasarnya dilakukan unuk memeriksa seberapa cepat dan baiknya aplikasi bisa pulih terhadap semua jenis crash atau kegagalan hardware, masalah bencana, dan lain-lain. Jenis atau taraf pemulihan ditetapkan dalam persyaratan spesifikasi.

1. Pengujian volume (*Volume testing*)

Pengujian volume, seperti namanya, adalah pengujian sebuah sistem (baik perangkat keras dan perangkat lunak) untuk serangkaian pengujian dengan volume data yang diproses adalah subjek dari pengujian, seperti sistem yang dapat menangkap sistem pengolahan transaksi penjualan real-time atau dapat membarui basis data atau pengembalian data (data retrieval).

1. Pengujian domain (*Domain testing*)

Pengujian domain merupakan penjelasan yang paling sering menjelaskan teknik pengujian.Beberapa penulis hanya menulis tentang pengujian domain ketika mereka menulis disain pengujian.

1. Pengujian skenario (*Scenario testing*)

Pengujian skenario adalah pengujian yang realistis, kredibel dan memotivasi stakeholder, tantangan untuk program dan mempermudah penguji untuk melakukan evaluasi. Pengujian ini menyediakan kombinasi variabel-variabel dan fungsi yang sangat berarti daripada kombinasi buatan yang didapatkan dengan pengujian domain atau disain pengujian kombinasi.

1. Pengujian regresi (*Regression testing*)

Pengujian regresi adalah gaya pengujian yang berfokus pada pengujian ulang (retesting) setelah ada perubahan. Pada pengujian regresi berorientasi risiko (riskoriented regressin testing), daerah yang sama sudah diuji, akan kita uji lagi dengan pengujian yang berbeda (semakin kompleks). Usaha pengujian regresi bertujuan untuk mengurangi risiko berikut ini :

* 1. Perubahan yang dimaksudkan untuk memperbaiki bug yang gagal.
  2. Beberapa perubahan memiliki efek samping, tidak memperbaiki bug lama atau memperkenalkan bug baru.

1. Penerimaan pengguna (*User acceptance*)

Pada jenis pengujian ini, perangkat lunak akan diserahkan kepada pengguna untuk mengetahui apakah perangkat lunak memenuhi harapan pengguna dan bekerja seperti yang diharapkan. Pada pengembangan perangkat lunak, user acceptance testing (UAT), juga disebut pengujian beta (beta testing), pengujian aplikasi (application testing), dan pengujian pengguna akhir (end user testing) adalah tahapan pengembangan perangkat lunak ketika perangkat lunak diuji pada “dunia nyata” yang dimaksudkan oleh pengguna.

1. Pengujian alfa (*Alpha testing*)

Pada jenis pengujian ini, pengguna akan diundang ke pusat pengembangan. Pengguna akan menggunakan aplikasi dan pengembang mencatat setiap masukan atau tindakan yang dilakukan oleh pengguna. Semua jenis perilaku yang tidak normal dari sistem dicatat dan dikoreksi oleh para pengembang.

1. Pengujian beta (*Beta testing*)

Pada jenis pengujian ini, perangkat lunak didistribusikan sebagai sebuah versi beta dengan pengguna yang menguji aplikasi di situs mereka. Pengecualian/cacat yang terjadi akan dilaporkan kepada pengembang. Pengujian beta dilakukan setelah pengujian alfa.Versi perangkat lunak yang dikenal dengan sebutan versi beta dirilis untuk pengguna yang terbatas di luar perusahaan.Perangkat lunak dilepaskan ke kelompok masyarakat agar dapat memastikan perangkat lunak tersebut memiliki beberapa kesalahan atau bug.

* 1. **Perangkat Lunak Pembangun Sistem** 
     1. **Pengertian perangkat lunak**

Pengertian perangkat lunak menurut Al Bahra bin Ladjamudin (2006:3) menjelaskan bahwa perangkat lunak adalah objek tertentu yang dapat dijalankan seperti kode sumber, kode objek atau sebuah program yang lengkap. Produk perangkat lunak memiliki pengertian perangkat lunak yang ditambahkan dengan semua item dan pelayanan pendukung yang secara keseluruhan dapat memenuhi kebutuhan pemakai. Produk perangkat lunak memiliki banyak bagian yang meliputi manual, referensi, tutorial, intruksi instalasi, data sampel, pelayanan pendidikan, pelayanan pendukung teknis dan sebagainya. Semua yang dihasilkan oleh proyek perangkat lunak adalah produk kerja (work product).

Produk kerja meliputi:

1. Dokumen Engineering yang dipakai untuk menentukan, mengontrol, dan memantau usaha kerja.
2. Objek yang dijalankan seperti prototype, kendali test(test harness), dan piranti pengembangan tujuan khusus, dan
3. Data yang digunakan untuk testing, melacak proyek dan sebagainya

Komputer memerlukan program-program penunjang, yang biasanya disebut dengan perangkat lunak system yang akan digunakan untuk mengoperasikan aplikasi perangkat lunak.

Perangkat lunak pada dasarnya merupakan perilaku dinamis dari program suatu program komputer, sedangkan program adalah ekspresi intelektual yang dapat dirancang oleh seorang pemakai pada tingkatan tertentu. Program akan terdiri dari algoritma-algoritma yang terstruktur bahkan akan mengarah atau berorientasi kepada objek tertentu yang diinginkan oleh sipembuat program

* + 1. **Karakteristik Perangkat Lunak**

Penelitian dan pemahaman tentang karakteristik perangkat lunak sangatlah penting, untuk memperoleh pemahaman tentang perangkat lunak yang pada dasarnya berbeda dengan hal-hal lain yang dibangun oleh manusia. Ketika perangkat lunak dibuat proses kreatif manusia (analisis, desain, konstruksi, dan pengujian) diterjemahkan kedalam bentuk fisik.

Perangkat lunak lebih merupakan elemen logika dan bukan merupakan elemen system fisik. Sehingga perangkat lunak memiliki cirri yang berbeda dari perangkat keras:

1. Perangkat lunak dibangun dan dikembangkan, tidak dibuat dalam bentuk klasik.
2. Perangkat lunak tidak pernah usang
3. Sebagian besar perangkat lunak dibuat secara custum-built, serta dapat dirakit dari komponen yang sudah ada.
   * 1. **Aplikasi-aplikasi Perangkat Lunak**

Aplikasi perangkat lunak akan sangat membantu pekerjaan kita sehari-hari, dan sudah banyak dipasaran. Ada beberapa jenis aplikasi perangkat lunak yaitu yang menangani masalah: keuangan, basis data, pengolah kata, grafik, spreadsheet, gambar, multimedia, internet, dan lain-lain. Semua aplikasi perangkat lunak tersebut saat ini sudah bias dioperasikan di bawah Operating System Windows. Aplikasi perangkat lunak tersebut digolongkan menjadi beberapa bagian yaitu:

1. Perangkat Lunak Sistem (System Software) Perangkat lunak system merupakan sekumpulan program yang ditulis untuk melayani program-program yang lain.
2. Perangkat Lunak Real-Time (Real Time Software) Program-program yang memonitor / menganalisis / mengontrol kejadian-kejadian yang sesungguhnya (sedang berlangsung) pada dunia disebut dengan perangkat lunak real-time.
3. Perangkat Lunak Bisnis (Business Software) Pemrosesan informasi bisnis merupakan area aplikasi yang paling luas. Aplikasi dalam area ini akan menyusun kembali struktur data yang ada dengan cara tertentu untuk memperlancar operasi bisnis atau pengambilan keputusan manajemen.
4. Perangkat Lunak Teknik dan Ilmu Pengetahuan (Engineering and Scientific Software) Perangkat lunak teknik dan ilmu pengetahuan ditandai dengan algoritma number crunching. Perangkat lunak ini memiliki jangkauan aplikasi mulai dari astronomi sampai vulkanologi, dari analisis otomotif sampai dinamika orbit pesawat luar angkasa, dan dari biologi molecular sampai pada pabrik yang sudah diotomatisasi.
5. Perangkat Lunak Embedded (Embedded Sofware) Embedded software ada dalam real-only memory dan dipakai untuk mengontrol hasil serta system untuk keperluan konsumen dan pasar industry.
6. Perangkat Lunak Komputer Personal (Personal Computer Software) Pasar perangkat lunak komputer personal telah berkembang selama decade terakhir. Pengolahan kata, spreadsheet, grafik komputer, multimedia, hiburan, manajaemen database, aplikasi keuangan bisnis dan personal, jaringan eksternal atau akses database merupakan beberapa contoh yang dapat disebutkan.
7. Perangkat Lunak kecerdasan Buatan (Artifical Intelligence Sofware{AI}) Perangkat lunak kecerdasan buatan (Artifical Intelligence Sofware{AI}) menggunakan algoritma non-numerik untuk memecahkan masalah kompleks yang tidak sesuai untuk perhitungan atau analisis secara langsung. Area kecerdasan buatan yang aktif adalah sistem pakar (expert system), sering disebut dengan sistem berbasis ilmu pengetahuan
   * 1. **Faktor-faktor Kualitas Perangkat Lunak**

Faktor-faktor yang mempengaruhi kualitas software dibagi menjadi 2 kategori:

1. Faktor-faktor yang dapat diukur secara langsung (misalkan: error)
2. Faktor-faktor yang dapat diukur secara tidak langsung ( misalkan : usability dan maintainability).

Menurut McCall terdapat 3 aspek penting dari suatu produk software, yaitu: karakteristik operasional, kemampuan perubahan ketika software sudah berjalan, dan kemampuan beradaptasi terhadap lingkungan baru.

McCall menyediakan beberapa deskripsi yaitu:

1. Correctness (kebenaran), tingkat pemenuhan program terhadap kebutuhan yang dispesifikasikan dan memenuhi tujuan/misi konsumer.
2. Reliability (keandalan), tingkat kemampuan program yang diharapkan dapat menampilkan fungsi yang dimaksud dengan persisi yang diterapkan.
3. Efficiency (efisien), jumlah sumberdaya yang diproses dan kode yang diperlukan oleh program untuk melaksanakan fungsi tersebut.
4. Integrity (integritas), tingkat kemampuan pengawasan akses terhadap data atau software oleh orang-orang tertentu.
5. Usability, usaha yang diperlukan untuk mempelajari, mengoperasikan, menyiapkan masukan dan mengartikan keluaran oleh program.
6. Maintanability, usaha yang diperlukan untuk menetapkan dan memperbaiki kesalahan dalam program.
7. Flexibility, usaha yang diperlukan untuk memodifikasi program operasional.
8. Testability, usaha yang diperlukan untuk menguji program untuk memastikan bahwa program melaksanakan fungsi yang ditetapkan.
9. Portability, usaha yang diperlukan untuk memindahkan program dari hardware/lingkungan sistem software tertentu ke yang lainnya.
10. Reusability, tingkat kemampuan program/ bagian dari program yang dapat dipakai ulang dalam aplikasi lainnya, berkaitan dengan paket dan lingkup dari fungsi yang dilakukan oleh program.
11. Interoperability, usaha yang diperlukan untuk menggabungkan satu sistem dengan yang lainnya.
    1. ***Java***

*Java* merupakan “bahasa pemrograman yang dikembangkan dari bahasa C++, sehingga bahasa pemrograman ini seperti bahasa C++”. Bahasa *Java* dapat dijalankan pada sebuah komputer dengan menggunakan *software* yang disebut dengan *Compiler* yang berfungsi untuk mengkonversikan kode sumber ke program biner yang berisi *bytecode*, kemudian *interpreter* digunakan untuk interpretasi dengan tujuan kode program yang tidak dapat dieksekusi tetapi tetap berjalan pada komputer yang sudah distandarisasikan, yang disebut *Java Virtual Machine*. Java pertama kali diciptakan oleh James Gosling dan Patrick Naughton pada suatu *project* dengan *Green code* di Sun Micro*system*. Java kemudian diperkenalkan pada awal tahun 1996 dengan sebutan JDK 1.1 (*Java Development Kit* versi 1.1). Dalam pengembangan muncul Java 2 yang dikembangkan dari Java JDK 1.1 yang dilengkapi dengan Swing dengan teknologi *GUI (Graphical User Interface)* yang dapat menciptakan aplikasi berbasis *desktop*. Java terus dikembangkan oleh Sun, sehingga pada tahun 2006 penamaan platform dengan tujuan untuk mencerminkan tingkat kesempurnaan, stabilitas, skalabilitas serta keamanan atau security yang lebih baik lagi. Pada sebelumnya ialah *Java 2 Platform*, dengan *Standar Edition 5.0* maka sekarang disederhanakan menjadi *Java Platform,* dengan *Standar Edition 6 (Java SE6* atau *Java 6)* ( Hamdhani, 2015).

* + 1. **Pengertian *NetBeans***

*Netbeans* merupakan sebuah aplikasi *Integrated Development Environment (IDE)* yang berbasiskan *Java* dari *Sun Microsystems* yang berjalan di atas *swing* dan banyak digunakan sekarang sebagai editor untuk berbagai macam bahasa pemrograman. Sampai sekarang *NetBeans* sudah sampai ke versi 8.0. pada *Netbeans*, kita bisa membuat bahasa pemrograman *Java, Javascript, PHP, Python, Ruby, Groovy, C, C++, Scala, Clojure. Swing* merupakan teknologi *java* untuk pengembangan aplikasi dekstop yang bisa dijalankan di berbagai sistem operasi (Nofriadi, 2015)

* + 1. ***MySQL***

*Mysql* mulai dikembangkan pada tahun 1979 dengan *tool database UNIREG* yang dibuat Michael Monty Widenius untuk perusahaan TcX di Swedia. Pada tahun 1994 TcX mulai mengembangakan *SQl* untuk aplikasi web. Tahun 1995 David Axmark dari Detro H berupaya menekan TcX merelease *Mysql* di dunia maya atau internet. Pada tahun 1996 *Mysql 3.11.1* mulai dipublikasi di dunia dan didistribusikan untuk *Linux*. Sampai saat ini *Mysql* sudah dapat bekerja untuk banyak *Flatform* dengan dilengkapi *source code*.

*Mysql* merupakan *system* manajemen database yang bersifat *open source* atau gratis. Keunggulan yang bisa kita ketahui dari Mysql antara lain (Yanto, 2016):

1. Kecepatan.
2. Kemudahan bagi user dalam penggunaanya.
3. Bersifat open source atau gratis.
4. Support dengan bahasa query.
5. User dapat mengakses lebih dari satu dalam satu waktu.
6. Akses data dapat dilakukan di setiap tempat dengan fasilitas internet.
7. Mysql mudah didapatkan karena source code yang dapat disebarluaskan.
   1. **Database**
      1. **Pengertian *database***

Basis data ( database ) merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras computer dan digunakan perangkat lunak untuk manipulasinya. Satu database menunjukkan satu kumpulan data yang dipakai dalam satu lingkup sistem. Dalam satu file terdapat *Record*-*Record* yang sejenis, sama besar, sama bentuk yang merupakan kesimpulan *entity* yang seragam. Satu *Record* terdiri dari field-field yang saling berhubungan untuk menunjukkan bahwa field tersebut dalam satu pengertian yang lengkap dan direkam dalam satu *Record*. Untuk menyebut isi dari field maka digunakan atribut nama menunjukkan *entity* nama dari pasien. *Entity* merupakan suatu objek yang nyata dan akan direkam.

Beberapa definisi database yang perlu dipahami, antara lain :

* 1. *Entity* adalah sesuatu yang dapat diidentifikasikan dari suatu konsep database, bisa berupa objek, orang, tempat, kejadian atau konsep yang informasinya akan disimpan di database.
  2. *Atribut* adalah suatu *entity* yang mempunyai atribut untuk mewakili suatu *entity*.
  3. Data *Value* adalah Data actual informasi yang disimpan pada setiap data elemen atau atribut.
  4. *Record* adalah kumpulan elemen-elemen yang saling berkaitan dan menginformasikan atau menjelaskan tentang suatu *entity* secara lengkap.
  5. Field adalah suatu atibut dari *Record* yang menunjukkan suatu item data yang sejenis.
  6. File adalah Kumpulan *Record* - *Record* sejenis yang mempunyai panjang elemen yang sama, atribut yang sama, namun berbeda data *value*nya.
  7. Database adalah kumpulan dari tabel – tabel yang saling berelasi, disusun secara logis, sehingga menghasilkan informasi yang bernilai guna dalam proses pengambilan keputusan.
     1. **Kegunaan *Database***

Menurut Iman (1996) kegunaan sistem database adalah agar pemakai atau user mampu menyusun suatu pandangan abstraksi data. Menurut Kristanto (1994) kegunaan suatu database adalah untuk mengatasi masalah pada penyusutan data, antara lain :

1. Multiple User (banyak pemakai)
2. Kesulitan Pengaksesan Data
3. Redudansi dan Inkonsistensi Data
4. Isolasi data untuk standarisasi
5. Masalah Keagamaan (Security)
6. Masalah Integrasi (Kesatuan)
7. Masalah data independence (Kebebasan data)
   1. **Web Cloud**
      1. ***Software as a Service (SaaS)***

*Software as service* merupakan evolusi lanjutan dari konsep *ASP( Application Service Provider*). *Software as service* adalah istilah terhadap *software* atau aplikasi tertentu berbasis internet yang ditawarkan oleh provider kepada pengguna. Dalam hal ini, provider sebagai pemegang licenseatas *software* tersebut hanya memberikan service atau layanan kepada pengguna untuk menggunakannya sesuai kebutuhan pengguna dengan demikian menghilangkan kerumitan dalam hal pemeliharaan *software*, operasional dan support. License, maintenance, support, tingkat kenyamanan dan keamanan atas *software* tersebut sepenuhnya menjadi tanggung jawab dari provider.

Kata – kata ―*Software* merujuk kepada perangkat lunak suatu *system*, dimana perangkat lunak pada umumnya memiliki beragam karakteristik. Tidak semua perangkat lunak yang beredar di pasaran dapat dikategorikan sebagai *SaaS*, ada beberapa karakteristik yang harus terpenuhi :

1. Berbasis internet ; *software* harus dapat diakses dan dikelola oleh pengguna melalui media internet.
2. *Software* bersifat terpusat atau ter-sentral sehingga memungkinkan pengguna untuk mengaksesnya darimana dan kapan saja.
3. Memiliki fasilitas untuk meng-update atau meng-upgrade secara terpusat sehingga pengguna tidak perlu download patch atau upgrade di masing –masing komputer.
4. Aplikasi yang ditawarkan oleh provider bersifat multi tenant

*Software* *as service* menawarkan beberapa keuntungan kepada pengguna dibanding dengan model aplikasi desktop:

1. Model rancangan dan distribusi *software* lebih menarik dan harga terjangkau karena memungkinkan membagi satu aplikasi kepada ratusan perusahaan dan berjalan dalam lingkungan sistem biasa. Secara luas memberikan improvisasi kepada model client /server.
2. Biaya pemakaian bandwidth untuk menjaga tingkat konektivitas relatif terjangkau.
3. Mempermudah pengguna untuk melakukan migrasi aplikasi, dengan menghilangkan sisi pembayaran license *software* dan keharusan membayar upgrade.
4. Meningkatkan produktivitas bagi pengguna

Metodologi pengembangan dari *SaaS* memiliki kesamaan dengan pengembangan *software* desktop baik dari sisi kemampuan aplikasi diakses dalam skala besar, tingkat keamanan dan aplikasi yang nyaman digunakan oleh pengguna. Beberapa faktor keberhasilan dalam implementasi dan pengembangan *SaaS* yaitu :

* Efisiensi sumber daya komputer : *SaaS* memiliki kemampuan memaksimalkan penggunaan sumber daya komputer seperti pemakaian memory dan bandwidth secara bersamaan, penggunaan database berskala besar untuk berbagai pengguna di berbagai lokasi yang berbeda dalam waktu bersamaan.
* Optimasi data dan multi tenant : *SaaS* memiliki kemampuan untuk memilah data – data dan menseleksi data – data berdasarkan kepemilikan pengguna secara bersamaan dalam satu aplikasi ( multi tenant ).
* Fleksibel aplikasi : *SaaS* memiliki tingkat fleksible yang tinggi dan memungkinkan pengguna memodifikasi aplikasi sesuai kebutuhan pengguna.
  + 1. ***Cloud Platform as a Service (Paas)***

Kemampuan yang diberikan kepada konsumen untuk menyebarkan aplikasi yang dibuat konsumen atau diperoleh ke infrastruktur cloud computing menggunakan bahasa pemrograman dan peralatan yang didukung oleh provider. Konsumen tidak mengelola atau mengendalikan infrastruktur cloud yang mendasar termasuk jaringan, server, sistem operasi, atau penyimpanan, namun memiliki kontrol atas aplikasi yang disebarkan dan memungkinkan aplikasi melakukan hosting konfigurasi. Contohnya yang sudah mengimplementasikan ini

adalah Force.com dan Microsoft Azure investment.

* + 1. ***Cloud Infrastructure as a Service (IaaS)***

Kemampuan yang diberikan kepada konsumen untuk memproses, menyimpan, berjaringan, dan sumber komputasi penting yang lain, dimana konsumen dapat menyebarkan dan menjalankan perangkat lunak secara bebas, yang dapat mencakup sistem operasian aplikasi. Konsumen tidak mengelola atau mengendalikan infrastruktur cloud yang mendasar tetapi memiliki kontrol atas sistem operasi, penyimpanan, aplikasi yang disebarkan, dan mungkin kontrol terbatas komponen jaringan yang pilih (misalnya, firewall host). Contohnya seperti Amazon Elastic Compute Cloud dan Simple Storage Service.

Dalam menawarkan layanan IaaS kepada pengguna atau penyewa, provider membagi IaaS dalam beberapa kategori layanan yaitu :

1. Layanan penyimpanan dan komputasi virtual : yaitu VMware rental, penyimpanan online ( Online Storage ).
2. Layanan kustomise : yaitu server template.
3. Layanan automasi dan control : yaitu automation.
4. Layanan penghubung : yaitu remote control, web 2.0.
5. Layanan monitoring : yaitu monitor secara fisik objek yang diinginkan ( posisi koordinat bumi, peta, kamera ).
6. Layanan optimasi objek : yaitu virtualisasi network, virtualisasi penyimpanan, virtualisasi server.
7. Layanan pengukuran objek : yaitu pengukuran fisik suatu objek.
8. Layanan integrateddan kombinasi objek : yaitu load balance.
9. Layanan security : yaitu enkripsi data penyimpanan, VM isolation, VLANdan SSL/SSH
   1. **Konsep Pemodelan Sistem**
      1. **Diagram Konteks**

Diagram konteks bisa disebut dengan model sistem pokok (fundamental *system* model) mewakili keseluruhan elemen *software* dengan masukan (input) dan keluaran (output) yang diidentifikasi dengan anak panah masuk dan keluar memperlihatkan sumber data (Pressman, 1997).

Diagram konteks adalah diagram yang terdiri dari suatu proses yang menggambarkan ruang lingkup suatu sistem (Jogiyanto, 2005). Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem. Ia akan memberi gambaran tentang keseluruhan sistem. Sistem dibatasi oleh boundar (dapat digambarkan dengan garis putus). Dalam diagram konteks hanya ada satu proses. Tidak boleh ada store dalam diagram konteks.

* + 1. **Data Flow Diagram (DFD)**

Data Flow Diagram (DFD) adalah suatu model logika data atau proses yang dibuat untuk menggambarkan dari mana asal data dan kemana tujuan data yang keluar dari sistem, dimana data disimpan, proses apa yang menghasilkan data tersebut dan interaksi antara data yang tersimpan dan proses yang dikenakan pada data tersebut. DFD menggambarkan penyimpanan data dan proses yang mentransformasikan data. DFD menunjukkan hubungan antara data pada sistem dan proses pada *system* (Kristanto, 2003).

Ada dua teknik dasar DFD yang umum dipakai, yaitu Gane/Sarson dan Yourdon/De Marco. Simbol-simbol yang digunakan dalam DFD dapat dilihat pada tabel 2.1.

Tabel 2.1 Komponen DFD (yourdon, 1989).

|  |  |
| --- | --- |
| Simbol | Keterangan |
|  | Proses, menunjukkan transformasi dari masukan dan  keluaran |
|  | Aliran,Komponen ini dipresentasikan dengan  menggunakan panah yang menuju ke atau dari proses.  Digunakan untuk menggambarkan gerakan paket data atau informasi dari satu bagian ke bagian lain dari sistem dimana penyimpanan mewakili lokasi penyimpanan data |
| Nama penyimpanan | Penyimpanan, komponen ini digunakan untuk  memodelkan kumpulan data atau paket data. |
|  | Terminator, komponen berikut ini dipresentasikan menggunakan persegi panjang, yang mewakili *entity* luar dimana sistem berkomunikasi. |

Peraturan-peraturan yang harus diperhatikan dalam penggambaran simbol DFD adalah sebagai berikut (Kristanto, 2003):

1. Antar-entiti luar tidak diijinkan terjadi hubungan atau relasi.
2. Tidak boleh ada aliran data antara entiti luar dengan tempat penyimpanan.
3. Untuk alasan kerapian, entiti luar atau tempat penyimpanan boleh digambar beberapa kali dengan tanda khusus, misalnya diberi nomor.
4. Satu aliran data boleh mengalirkan beberapa struktur data.
5. Bentuk anak panah aliran data boleh bervariasi.
6. Semua obyek harus mempunyai nama.
7. Aliran data selalu diawali dan diakhiri dengan proses.
8. Semua aliran data harus mempunyai tanda arah.

Ada beberapa petunjuk yang dapat dipakai dalam proses pembuatan simbol DFD, yaitu (Kristanto, 2003):

1. Penamaan yang jelas.
2. Memberi nomor pada proses.
3. Penggambaran kembali.
4. Hindarilah proses yang mempunyai masukan tetapi tidak mempunyai keluaran, begitu pula sebaliknya, hindarilah proses yang mempunyai keluaran tetapi tidak mempunyai masukan.
5. Hati-hati dengan aliran data dan proses yang tidak dinamai.

DFD levelled menggambarkan sistem sebagai jaringan kerja antar-fungsi yang berhubungan satu dengan yang lain dengan aliran dan penyimpanan data. Dalam DFD levelled akan terjadi penurunan level dimana dalam penurunan level yang lebih rendah harus mampu merepresentasikan proses tersebut ke dalam spesifikasi proses yang jelas. Jadi dalam DFD levelled bisa dimulai dan DFD level 0 kemudian turun ke DFD level 1 dan seterusnya. Setiap penurunan hanya dilakukan bila perlu. Aliran data yang masuk dan keluar pada suatu proses di level X harus berhubungan dengan aliran data yang masuk dan keluar pada level X+1 yang mendefinisikan proses pada level X tersebut (Kristanto, 2003)

* + 1. ***Entity* Relationship Diagram (ERD)**

*Entity relationship diagram* adalah pemodelan data yang menggunakan beberapa notasi untuk menggambarkan data yang berhubungan dengan *entity* dan *relationship* yang dideskripsikan oleh data tersebut (Whitten dkk, 2004). Simbol-simbol ERD ditunjukkan pada tabel 2.2.

Table 2.2. simbol ERD (Edi, 2009).

|  |  |  |
| --- | --- | --- |
| Notasi | Komponen | Keterangan |
|  | Entitas | Individu yang mewakili suatu objek dan dapat dibedakan dengan objek yang lain |
|  | Atribut | Properti yang dimiliki oleh suatu entitas, dimana dapat mendeskripsikan karakteristik dari entitas tersebut |
|  | Relasi | Menunjukan hubungan diantara sejumlah entitas yang berbeda. |
|  | Relasi 1 : 1 | Relasi yang menunjukan bahwa setiap entitas pada himpunan entitas pertama berhubungan dengan paling banyak satu entitas pada himpunan entitas kedua. |
|  | Relasi 1 : N | Relasi yang menunjukan bahwa hubungan antara entitas pertama dengan entitas kedua adalah satu banding banyak atau sebaliknya. Setiap entitas dapat berelasi dengan banyak entitas pada himpunan entitas yang lain. |

* + 1. ***Flowchart***

Bagan alir (*Flowchart*) adalah bagan (*chart*) yang menunjukan hasil (*flow*) didalam program atau prosedur sistem secara logika. Bagan alir digunakan terutama untuk alat bantu komunikasi dan untuk dokumentasi (Jogiyanto, 2005)

*Flowcart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelsaian suatu masalah. *Flowcart*merupakan cara penyajian dari suatu algoritma (Ladjamuddin, 2006).

Pedoman dalam menggambar suatu bagan alir, analis sistem atau pemrograman sebagai berikut;

* + 1. Bagan alir sebaiknya digambar dari atas ke bawah dan mulai dari bagian kiri dari suatu halaman.
    2. Kegiatan didalam bagan alir harus ditunjukan dengan jelas.
    3. Harus ditunjukan darimana kegiatan akan dimulai dan dimana akan berakhirnya.
    4. Masing-masing kegiatan didalam bagan alir sebaiknya digunakan suatu kata yang mewakili  suatu pekerjaan, misalnya;“persiapkan” dokumen “hitung” gaji.
    5. Masing-masing kegiatan didalam bagan alir harus didalm urutan yang semestinya.
    6. Kegiatan yang terpotong dan akan disambung ketempat lain harus ditunjukan dengan jelas menggunakan symbol penghubung.
    7. Gunakanlah symbol-simbol bagan alir yang standar.

Ada 5 macam menurut jogiyanto bagan alir diantaranya;

* + 1. Bagan alir sistem (*sistems Flowchart*)

       Bagan alir sistem (*system Flowchart*) merupakan bagan yang menunjukkan arus pekerjaan secara keseluruan dari sistem. Bagan menjelaskan urutan-urutan dari prosedure-prosedure yang ada dalam sistem. Bagan alir sistem menunjukan apa yang dikerjakan sistem.

* + 1. Bagan alir dokumen (*document Flowchart*)

      Bagan alir dokumen (*document Flowchart*) atau disebut bagan alir formulir (*form Flowchart*) atau *paperwork Flowchart* merupakan bagan alir yang menunjukan arus dari laporan dan formulir termasuk tembusan-tembusannya. Bagan alir dokumen ini menggunakan simbol-simbol yang sama dengan yang digunakan di dalam bagan alir sistem.

* + 1. Bagan alir skematik (*schematic Flowchart*)

      Bagan alir skematik (*schematic Flowchart*) merupakan bagan alir yang mirip dengan bagan alir sistem, yaitu untuk menggambarkan prosedur di dalam sistem. Perbedaannya adalah bagan alir skematik menggunakan simbol-simbol bagan alir sistem , juga menggunakan gambar - gambar komputer dan peralatan lainnya yang digunakan. Maksud penggunaan gambar-gambar ini adalah untuk memudahkan komunikasi kepada orang yang kurang paham dengan simbol-simbol bagan alir.

* + 1. Bagan alir program (*program Flowchart*)

       Bagan alir program (*program Flowchart*) merupakan bagan yang menjelaskan secara rinci langkah-langkah dari proses program. Bagan alir program dibuat dengan menggunakan simbol-simbol yang ditunjukkan pada gambar 2.3.

* + 1. Bagan alir proses (*process Flowchart*)

Bagan alir proses (*process Flowchart*) merupakan bagan alir yang banyak digunakan teknik industri. Bagan alir juga berguna bagi anilis sistem untuk menggambarkan proses dalam suatu prosedure.

Simbol dari bagan alir ( *Flowchart* ) adalah seperti yang ditunjukkan pada tabel 2.3.

Tabel 2.3. simbol *Flowchart* (Jogiyanto, 2009).

|  |  |  |  |
| --- | --- | --- | --- |
| No | Simbol | Pengertian | Keterangan |
| 1. |  | Mulai / berakhir ( *Terminal* ) | Digunakan untuk memulai, mengakhiri, atau titik henti dalam sebuah proses atau program; juga digunakan untuk menunjukkan pihak eksternal. |
| 2. |  | Dokumen | Sebuah dokumen atau laporan; dokumen dapat dibuat dengan tangan atau dicetak oleh komputer. |
| 4. |  | Input / Output; | Digunakan untuk menggambarkan berbagai media input dan output dalam sebuah bagan alir program. |
| 5. |  | Disk Bermagnit | Data disimpan secara permanen pada disk bermagnit. |
| 6. |  | Penghubung Pada Halaman Berbeda | Menghubungkan bagan alir yang berada dihalaman yang berbeda. |
| 7. |  | Pemasukan Data On Line | Entri data alat oleh on line seperti terminal CRT dan komputer pribadi. |
| 8. |  | Pemrosesan Komputer | Sebuah fungsi pemrosesan yang dilaksanakan oleh komputer biasanya menghasilkan perubahan terhadap data atau informasi |
| 9. |  | Arus Dokumen atau Pemrosesan | Arus dokumen atau pemrosesan; arus normal adalah ke kanan atau ke bawah. |
| 10. |  | Keputusan | Sebuah tahap pembuatan keputusan |
| 11. |  | Penghubung Dalam Sebuah Halaman | Menghubungkan bagan alir yang berada pada halaman yang sama. |
| 12. |  | Dokumen Rangkap | Digambarkan dengan menupuk simbol dokumen dan pencetakan nomor dokumen dibagian depan dokumen pada bagian kiri atas. |