Manajemen Proyek Sistem Informasi

**Oleh :** Imelda, S.T.,M.T

(Digunakan di lingkungan sendiri, sebagai buku ajar

mata kuliah ……..)



**Fakultas Teknik dan Ilmu Komputer**

**Program Studi Manajemen Informatika**

**Universitas Komputer Indonesia**

**1. Pendahuluan**

1.1 Mata kuliah ini memberikan pemahaman konsep mengenai manajemen proyek system informasi, perancanaan dan strategi sumber daya informasi, pemanfaatan proyek dengan pengelolaan teknologi informasi dalam organisasi dan memahami model serta metodologi manajemen proyek system informasi.

1.2 Kontrak Perkuliahan

1. Perkuliahan diselenggarakan 14 kali pertemuan (3 SKS)
2. Wajib kehadiran Mahasiswa 80% (-3 kali tidak masuk)
3. Materi perkuliahan akan diberikan salinannya kepada Mahasiswa
4. Batas keterlambatan 15 menit setelah perkuliahan dimulai
5. Mahasiswa diperbolehkan berkonsultasi dengan dosen; mengenai materi perkuliahan secara personal atau kelompok di luar jam perkuliahan (tatap muka; via email; kuliah online)
6. Mengikuti tata tertib perkuliahan
7. Tidak diperbolehkan menggunakan perangkat komunikasi selama perkuliahan (setting silent/vibrate)
8. Bersikap sopan dan tidak mengganggu keberlangsungan perkuliahan
9. Tersedia waktu Shalat bagi yang beragama Islam.

**Silabus :**

**Pertemuan 1 : Pendahuluan, Kontrak Perkuliahan dan Silabus**

**Pertemuan 2 : Manajer, Tim IT dan Kerjasama Tim**

a. Kualifikasi dan Syarat Manajer Proyek

b. Kualifikasi dan Syarat Tim IT dalam suatu Proyek

**Pertemuan 3 dan 4: Fase Definisi Proyek, Pengembangan dan Pemeliharaan**

a. Objektif

b. Model Proses Perangkat Lunak

c. Prototype

d. Sequence dll

**Pertemuan 5 dan 6 : Scoping, Planning, Estimating, Scheduling, Organizing, Directing, Controlling, and Closing**

a. System

b. Karakteristik Sistem

c. Sistem Informasi berbasis Computer (CBIS)

d. Database

**Pertemuan 7 : Pert Chart dan Gannt Chart**

a. Teknik Penjadwalan hanya berbasis pada pembagian tugas

b. Teknik Penjadwalan dengan bantuan hari kerja dalam Kalender berdasarkan tahapan pekerjaan.

**Pertemuan 8 : UJIAN TENGAH SEMESTER**

**Pertemuan 9 : Work Breakdown Structure**

a. Fase-fase, aktifitas-aktifitas dan tugas-tugas dari proyek yang akan dikerjakan

**Pertemuan 10: Penugasan Proyek**

a. Tahapan-tahapan dalam penugasan proyek

b. Surat tugas

**Pertemuan 11 dan 12 : Faktor-faktor dalam menginisialisasi proyek**

a. Permasalahan

b. Deskripsi Produk

c. Faktor penentu keberhasilan

d. Keuntungan yang diharapkan

e. Teknologi yang akan digunakan

f. Deskripsi proyek

g. Perencanaan aktifitas secara global

h. Asumsi

**Pertemuan 13 : Perencanaan Proyek**

a. Dokumen Control

b. Ruang lingkup proyek

c. Tujuan

d. Jadwal Proyek

e. Struktur Organisasi Proyek

f. Manajemen Komunikasi

g. Manajemen Perubahan

**Pertemuan 14 dan 15 : Pelaksanaan, Kontroling dan Closing Proyek**

a. Form pelaksanaan proyek

b. Form agenda perubahan

c. Form penyelesaian proyek

d. Form penutupan proyek

**Pertemuan 16 : UJIAN AKHIR SEMESTER**

**3. Fase Definisi Proyek dan Pengembangan & Pemeliharaan**

3.1 Objektif

3.2 Model Proses Perangkat Lunak

3.2.1 Model Sekuensial Linier atau Waterfall Development Model



* Rekayasa dan pemodelan sistem/informasi
* Analisis kebutuhan perangkat lunak
* Desain
* Pengkodean
* Pengujian
* Pemeliharaan

Kelebihan Model Sekuensial Linear / Waterfall Development Model :

* Tahapan proses pengembangannya tetap (pasti), mudah diaplikasikan, dan prosesnya teratur.
* Cocok digunakan untuk produk software/program yang sudah jelas kebutuhannya di awal, sehingga minim kesalahannya.
* Software yang dikembangkan dengan metode ini biasanya menghasilkan kualitas yang baik.
* Documen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya.

Kekurangan Model Sekuensial Linear / Waterfall Development Model :

* Proyek yang sebenarnya jarang mengikuti alur sekuensial seperti diusulkan, sehingga perubahan yang terjadi dapat menyebabkan hasil yang sudah didapatkan tim pengembang harus diubah kembali/iterasi sering menyebabkan masalah baru.
* Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakukan pada tahap awal proses.
* Sulit untuk mengalami perubahan kebutuhan yang diinginkan oleh customer/pelanggan.
* Pelanggan harus sabar untuk menanti produk selesai, karena dikerjakan tahap per tahap, dan proses pengerjaanya akan berlanjut ke setiap tahapan bila tahap sebelumnya sudah benar-benar selesai.
* Perubahan ditengah-tengah pengerjaan produk akan membuat bingung tim pengembang yang sedang membuat produk.
* Adanya waktu kosong (menganggur) bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.

3.2.2 Model Prototype



Teknik – teknik Prototyping Meliputi :

* Perancangan Model
* Perancangan Dialog
* Simulasi

Berikut adalah 4 langkah yang menjadi karakteristik dalam proses pengembangan pada metode prototype, yaitu :

* Pemilihan fungsi
* Penyusunan Sistem Informasi
* Evaluasi
* Penggunaan Selanjutnya

Berikut adalah Tahapan – tahapan Proses Pengembangan dalam Model Prototype, yaitu :

* Pengumpulan kebutuhan
* Membangun prototyping
* Evaluasi protoptyping
* Mengkodekan sistem
* Menguji sistem
* Evaluasi Sistem
* Menggunakan sistem

Kelebihan Model Prototype :

* Pelanggan berpartisipasi aktif dalam pengembangan sistem, sehingga hasil produk pengembangan akan semakin mudah disesuaikan dengan keinginan dan kebutuhan pelanggan.
* Penentuan kebutuhan lebih mudah diwujudkan.
* Mempersingkat waktu pengembangan produk perangkat lunak.
* Adanya komunikasi yang baik antara pengembang dan pelanggan.
* Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
* Lebih menghemat waktu dalam pengembangan sistem.
* Penerapan menjadi lebih mudah karena pelanggan mengetahui apa yang diharapkannya.

Kekurangan Model Prototype :

* Proses analisis dan perancangan terlalu singkat.
* Biasanya kurang fleksibel dalam mengahadapi perubahan.
* Walaupun pemakai melihat berbagai perbaikan dari setiap versi prototype, tetapi pemakai mungkin tidak menyadari bahwa versi tersebut dibuat tanpa memperhatikan kualitas dan pemeliharaan jangka panjang.
* Pengembang kadang-kadang membuat kompromi implementasi dengan menggunakan sistem operasi yang tidak relevan dan algoritma yang tidak efisien.

3.2.3 Model Rapid Application Development (RAD)

Rapid Aplication Development (RAD) adalah sebuah model proses perkembanganperangkat lunak sekuensial linier yang menekankan siklus perkembangan yang sangat pendek (kira-kira 60 sampai 90 hari).

Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sekuensial linier dimana perkembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen.



Berikut adalah Tahapan – tahapan Proses Pengembangan dalam Model Rapid Application Development (RAD), yaitu :

* Bussiness Modeling
* Data Modeling.
* Proses Modeling
* Aplication Generation
* Testing dan Turnover

Kelebihan Model RAD :

* Lebih efektif dari Pengembangan Model waterfall/sequential linear dalam menghasilkan sistem yang memenuhi kebutuhan langsung dari pelanggan.
* Cocok untuk proyek yang memerlukan waktu yang singkat.
* Model RAD mengikuti tahap pengembangan sistem seperti pada umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada sehingga pengembang tidak perlu membuatnya dari awal lagi sehingga waktu pengembangan menjadi lebih singkat dan efisien.

Kekurangan Model RAD :

* Model RAD menuntut pengembangan dan pelanggan memiliki komitmen di dalam aktivitas rapid-fire yang diperlukan untuk melengkapi sebuah sistem, di dalam kerangka waktu yang sangat diperpendek. Jika komitmen tersebut tidak ada, proyek RAD akan gagal.
* Tidak semua aplikasi sesuai untuk RAD, bila system tidak dapat dimodulkan dengan teratur, pembangunan komponen penting pada RAD akan menjadi sangat bermasalah.
* RAD tidak cocok digunakan untuk sistem yang mempunyai resiko teknik yang tinggi.
* Membutuhkan Tenaga kerja yang banyak untuk menyelesaikan sebuah proyek dalam skala besar.
* Jika ada perubahan di tengah-tengah pengerjaan maka harus membuat kontrak baru antara pengembang dan pelanggan.

3.2.4 Model Evolutionary Development / Evolutionary Software Process Models

Model Evolutionary Development bersifat iteratif (mengandung perulangan). Hasil prosesnya berupa produk yang makin lama makin lengkap sampai versi terlengkap dihasilkan sebagai produk akhir dari proses. Model Evolutionary Development / Evolutionary Software Process terbagi menjadi 2, yaitu :

Model Incremental

Model Incremental merupakan hasil kombinasi elemen-elemen dari model waterfall yang diaplikasikan secara berulang, atau bisa disebut gabungan dari Model linear sekuensial (waterfall) dengan Model Prototype. Elemen-elemen tersebut dikerjakan hingga menghasilkan produk dengan spesifikasi tertentu kemudian proses dimulai dari awal kembali hingga muncul hasil yang spesifikasinya lebih lengkap dari sebelumnya dan tentunya memenuhi kebutuhan pemakai.



Model ini berfokus pada penyampaian produk operasional dalam Setiap pertambahanya. Pertambahan awal ada di versi stripped down dari produk akhir, tetapi memberikan kemampuan untuk melayani pemakai dan juga menyediakan platform untuk evaluasi oleh pemakai.

Model ini cocok dipakai untuk proyek kecil dengan anggota tim yang sedikit dan ketersediaan waktu yang terbatas.

Kelebihan Model Incremental :

* Personil bekerja optimal.
* mampu mengakomodasi perubahan secara fleksibel, dengan waktu yang relatif singkat dan tidak dibutuhkan anggota/tim kerja yang banyak untuk menjalankannya.
* Pihak konsumen dapat langsung menggunakan dahulu bagian-bagian yang telah selesai dibangun. Contohnya pemasukan data karyawan.
* Mengurangi trauma karena perubahan sistem. Klien dibiasakan perlahan-lahan menggunakan produknya setiap bagian demi bagian.
* Memaksimalkan pengembalian modal investasi konsumen.

Kekurangan Model Incremental :

* Tidak cocok untuk proyek berukuran besar (lebih dari 200.000 baris coding).
* Sulit untuk memetakan kebutuhan pemakai ke dalam rencana spesifikasi tiap-tiap hasil dari increament.

3.2.5 Model Spiral / Model Boehm

Model ini mengadaptasi dua model perangkat lunak yang ada yaitu model prototyping dengan pengulangannya dan model waterfall dengan pengendalian dan sistematikanya.  Model ini dikenal dengan sebutan Spiral Boehm. Pengembang dalam model ini memadupadankan beberapa model umum tersebut untuk menghasilkan produk khusus atau untuk menjawab persoalan-persoalan tertentu selama proses pengerjaan proyek.



Tahap-tahap model ini dapat dijelaskan secara ringkas sebagai berikut :

* Tahap Liason:pada tahap ini dibangun komunikasi yang baik dengan calon pengguna/pemakai.
* Tahap Planning (perencanaan):pada tahap ini ditentukan sumber-sumber informasi, batas waktu dan informasi-informasi yang dapat menjelaskan proyek.
* Tahap Analisis Resiko:mendefinisikan resiko, menentukan apa saja yang menjadi resiko baik teknis maupun manajemen.
* Tahap Rekayasa (engineering):pembuatan prototipe.
* Tahap Konstruksi dan Pelepasan (release):pada tahap ini dilakukan pembangunan perangkat lunak yang dimaksud, diuji, diinstal dan diberikan sokongan-sokongan tambahan untuk keberhasilan proyek.
* Tahap Evaluasi:Pelanggan/pemakai/pengguna biasanya memberikan masukan berdasarkan hasil yang didapat dari tahap engineering dan instalasi.

Kelebihan model ini adalah sangat mempertimbangkan resiko kemungkinan munculnya kesalahan sehingga sangat dapat diandalkan untuk pengembangan perangkat lunak skala besar. Pendekatan model ini dilakukan melalui tahapan-tahapan yang sangat baik dengan menggabungkan model waterfall ditambah dengan pengulangan-pengulangan sehingga lebih realistis untuk mencerminkan keadaan sebenarnya. Baik pengembang maupun pemakai dapat cepat mengetahui letak kekurangan dan kesalahan dari sistem karena proses-prosesnya dapat diamati dengan baik.

Sedangkan Kekurangan model ini adalah waktu yang dibutuhkan untuk mengembangkan perangkat lunak cukup panjang demikian juga biaya yang besar. Selain itu, sangat tergantung kepada tenaga ahli yang dapat memperkirakan resiko. Terdapat pula kesulitan untuk mengontrol proses. Sampai saat ini, karena masih relatif baru, belum ada bukti apakah metode ini cukup handal untuk diterapkan.

3.2.6       WINWIN Spiral Model



Dalam hal ini win win merupakan situasi kemenangan antara tim pengembang dan pelanggan. Yang membedakan antara win win spiral model dan spiral model adalah setelah selesai mendapatkan feed back dari pelanggan, tim pengembang perangkat lunak dan pelanggan akan kembali melakukan negosiasi untuk perkembangan perangkat lunak tersebut.

Tahapan-tahapan WINWIN Spiral model:

* Identifikasi kunci sistem atau subsistem dari yang berkepentinagn
* Penentuan kondisi kemenangan dari yang berkepentingan
* Negosiasi kondisi menang yang berkepentingan agar terjadi kedamaian.

Berikut ada beberapa kelebihan dan kekurangan WINWIN Spiral Model

Kelebihan :

* Sama sama adanya kesepakatan developer dengan customer
* Terdapat kepuasan dan keuntungan antara developer dengan customer karena aplikasi yang dijalankan dengan negoisasi sesuai kesepakatan
* Sangat efektif untuk digunakan karena kesepakatan antara developer dengan customer sama-sama disepakati sehingga tidak akan menimbulkan ketidak puasan customer

Kekurangan :

* Membutuhkan waktu yang cukup lama2.Seringkali pada awalnya customer dengan developer mengalami kecekcokkan pada saat negoisasi

3.2.7       Component Based Development

Component-based development sangat berkaitan dengan teknologi berorientasi objek. Pada pemrograman berorientasi objek, banyak class yang dibangun dan menjadi komponen dalam suatu software. Class-class tersebut bersifat reusable artinya bisa digunakan kembali. Model ini bersifat iteratif atau berulang-ulang prosesnya.

Secara umum proses yang terjadi dalam model ini adalah:

* Identifikasi class-class yang akan digunakan kembali dengan menguji class tersebut dengan data yang akan dimanipulasi dengan aplikasi/software dan algoritma yang baru
* Class yang dibuat pada proyek sebelumnya disimpan dalam class library, sehingga bisa langsung diambil dari library yang sudah ada. Jika ternyata ada kebutuhan class baru, maka class baru dibuat dengan metode berorientasi objek.
* Bangun software dengan class-class yang sudah ditentukan atau class baru yang dibuat, integrasikan.

Penggunaan kembali komponen software yang sudah ada menguntungkan dari segi:

* Siklus waktu pengembangan   software, karena   mampu  mengurangi  waktu 70%
* Biaya produksi berkurang sampai 84% arena pembangunan komponen berkurang

Pembangunan software dengan menggunakan komponen yang sudah tersedia dapat menggunakan komponen COTS (Commercial off-the-shelf) – yang bisa didapatkan dengan membeli atau komponen yang sudah dibangun sebelumnya secara internal. Component-Based Software Engineering (CBSE) adalah proses yang menekankan perancangan dan pembangunan software dengan menggunakan komponen software yang sudah ada. CBSE terdiri dari dua bagian yang terjadi secara paralel yaitu software engineering (component-based development) dan domain engineering.

Berikut adalah beberapa kekurangan dan kelebihan dari Component Based Development           :

A.     Kelebihan model ini adalah tinggal mencaplok atau menggunakan program atau komponen yang sudah ada dan menyusunnya menjadi sebuah program yang lebih kompleks dan berkembang sesuai dengan kebutuhan user/pengguna sehingga dapat mengefisienkan penggunaan waktu dan tenaga.Selain itu,model ini juga menyediakan kemampuan untuk memvisualisasikan hasil rakitan dengan kesanggupan untuk mengukur, menganalisa, merancang dan merancang ulang program.

B.     Kekurangan model ini adalah seringnya program atau komponen-komponen terdahulu tidak kompatibel atau sejalan dengan model perakitan komponen ini sehingga untuk perusahaan berskala kecil akan kesulitan menemukan komponen yang sesuai untuk dirakit.

6.      Formal Method Model

Teknik formal method adalah teknik yang mengandalkan perhitungan matematika dalam setiap prosesnya. Hanya digunakan pada sistem yang sangat memperhatikan keamanan atau keselamatan dari pengguna. Contoh penggunaan teknik ini adalah aerospace engineering.

Keuntungan menggunakan teknik formal method adalah meminimalkan resiko dengan adanya perhitungan komputasi.

Sedangkan kerugiannya adalah:

1.      Biaya Tinggi.

2.      Kompleks

3.      Tidak Umum untuk Proyek Software pada umumnya

3.2.8      4th Generation

Metode pengembangan perangkat lunak 4GT menggunakan perangkat bantu (tools) yang akan membuat kode sumber secara otomatis berdasarkan spesifikasi dari pengembang perangkat lunak. Hanya digunakan untuk menggunakan perangkat lunak yang menggunakan bahasa khusus atau notasi grafik yang diselesaikan dengan syarat yang dimengerti pemakai. Cakupan aktivitas 4GT meliputi:

* Pengumpulan kebutuhan, idealnya pelanggan akan menjelaskan kebutuhan yang akan ditranslasikan ke prototype operasional.
* Translasi kebutuhan menjadi prototype operasional, atau langsung melakukan implementasi secara langsung dengan menggunakan bahasa generasi keempat (4GL) jika aplikasi relatif kecil.
* Untuk aplikasi yang cukup besar, dibutuhkan strategi perancangan sistem walaupun 4GL akan digunakan.

Pengujian.

* Membuat dokumentasi.
* Melaksanakan seluruh aktivitas untuk mengintegrasikan solusi-solusi yang membutuhkan paradigma rekayasa perangkat lunak lainnya.

Kelebihan dari metode pengembangan perangkat lunak ini diantaranya :

* Pengurangan waktu dan peningkatan produktivitas secara besar
* Karena 4GT menggunakan 4GL yang merupakan bahasa pemrograman yang  khusus dirancang dengan tujuan tertentu (spesifik), maka untuk permasalahan yang tertentu dengan 4GL tertentu pula sangat tepat menggunakan 4GT.
* Tool yang menggunakan metode pengembangan perangkat lunak 4GL bisa meng-generate sistem dari output yang dihasilkan oleh CASE tools.
* Kekurangan metode pengembangan perangkat lunak ini
* Penggunaan perangkat bantu (tools) dibandingkan dengan bahasa pemrograman, dan juga kode sumber yang dihasilkannya tidak efisien.
* Untuk usaha yang besar, dibutuhkan pengembangan strategi desain untuk sistem, walau digunakan bahasa 4GL.
* Penggunaan 4GT tanpa perencanaan matang (untuk proyek besar) akan menyebabkan kesulitan yang sama (kualitas dan pemeliharaan yang jelek, ketidakpuasan pelanggan) seperti dengan metode konvensional.
* 4GL tidak selalu berhasil menghasilkan sistem yang diinginkan.

3.3 Prototype

Pengertian dari prototyping : proses pengembangan sistem seringkali menggunakan pendekatan prototipe (prototyping). Metode ini sangat baik digunakan untuk menyelesesaikan masalah kesalahpahaman antara user dan analis yang timbul akibat user tidak mampu mendefinisikan secara jelas kebutuhannya (Mulyanto, 2009).

 Prototyping adalah pengembangan yang cepat dan pengujian terhadap model kerja (prototipe) dari aplikasi baru melalui proses interaksi dan berulang-ulang yang biasa digunakan ahli sistem informasi dan ahli bisnis.

 Prototyping disebut juga desain aplikasi cepat (rapid application design/RAD) karena menyederhanakan dan mempercepat desain sistem (O'Brien, 2005).
Sebagian user kesulitan mengungkapkan keinginannya untuk mendapatkan aplikasi yang sesuai dengan kebutuhannya.

 Kesulitan ini yang perlu diselesaikan oleh analis dengan memahami kebutuhan user dan menerjemahkannya ke dalam bentuk model (prototipe). Model ini selanjutnya diperbaiki secara terus menerus sampai sesuai dengan kebutuhan user.

 Keunggulan prototyping adalah :

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

Sedangkan kelemahan prototyping adalah :

1)     Pelanggan tidak melihat bahwa perangkat lunak belum mencerminkan kualitas perangkat lunak secara keseluruhan dan belum memikirkan peneliharaan dalam jangka waktu yang lama.
2)     Pengembang biasanya ingin cepat menyelesaikan proyek sehingga menggunakan algoritma dan bahasa pemrograman sederhana.
3)     Hubungan pelanggan dengan komputer mungkin tidak menggambarkan teknik perancangan yang baik.

Berdasarkan karakteristiknya prototipe sebuah sistem dapat berupa low fidelity dan high fidelity. Fidelity mengacu kepada tingkat kerincian sebuah sistem (Walker et al, 2003).

 Low fidelity prototype tidak terlalu rinci menggambarkan sistem. Karakteristik dari low fidelity prototype adalah mempunyai fungsi atau interaksi yang terbatas, lebih menggambarkan kosep perancangan dan layout dibandingkan dengan model interaksi, tidak memperlihatkan secara rinci operasional sistem, mendemostrasikan secara umum feel and look dari antarmuka pengguna dan hanya menggambarkan konsep pendekatan secara umum (Walker et al, 2003).

 High fidelity protoype lebih rinci menggambarkan sistem.
Prototipe ini mempunyai interaksi penuh dengan pengguna dimana pengguna dapat memasukkan data dan berinteraksi dengan dengan sistem, mewakili fungsi-fungsi inti sehingga dapat mensimulasikan sebagian besar fungsi dari sistem akhir dan mempunyai penampilan yang sangat mirip dengan produk sebenarnya (Walker et al, 2003).

 Fitur yang akan diimplementasikan pada prototipe sistem dapat dibatasi dengan teknik vertikal atau horizontal. Vertical prototype mengandung fungsi yang detail tetapi hanya untuk beberapa fitur terpilih, tidak pada keseluruhan fitur sistem. Horizontal prototype mencakup seluruh fitur antarmuka pengguna namun tanpa fungsi pokok hanya berupa simulasi dan belum dapat digunakan untuk melakukan pekerjaan yang sebenarnya (Walker et al, 2003).

Langkah-Langkah Prototyping

a.  Analisis Kebutuhan Sistem

 Pembangunan sistem informasi memerlukan penyelidikan dan analisis mengenai alasan timbulnya ide atau gagasan untuk membangun dan mengembangkan sistem informasi. Analisis dilakukan untuk melihat berbagai komponen yang dipakai sistem yang sedang berjalan meliputi hardware, software, jaringan dan sumber daya manusia.

 Analisis juga mendokumentasikan aktivitas sistem informasi meliputi input, pemrosesan, output, penyimpanan dan pengendalian (O'Brien, 2005).
Selanjutnya melakukan studi kelayakan (feasibility study) untuk merumuskan informasi yang dibutuhkan pemakai akhir, kebutuhan sumber daya, biaya, manfaat dan kelayakan proyek yang diusulkan (Mulyanto, 2009).

 Analisis kebutuhan sistem sebagai bagian dari studi awal bertujuan mengidentifikasi masalah dan kebutuhan spesifik sistem. Kebutuhan spesifik sistem adalah spesifikasi mengenai hal-hal yang akan dilakukan sistem ketika diimplementasikan (Mulyanto, 2009).

 Analisis kebutuhan sistem harus mendefinisikan kebutuhan sistem yang spesifik antara lain :

1)     Masukan yang diperlukan sistem (input)
2)     Keluaran yang dihasilkan (output)
3)     Operasi-operasi yang dilakukan (proses)
4)     Sumber data yang ditangani
5)     Pengendalian (kontrol)

Spesifikasi Kebutuhan Sistem

 Tahap analisis kebutuhan sistem memerlukan evaluasi untuk mengetahui kemampuan sistem dengan mendefinisikan apa yang seharusnya dapat dilakukan oleh sistem tersebut kemudian menentukan kriteria yang harus dipenuhi sistem.

 Beberapa kriteria yang harus dipenuhi adalah pencapaian tujuan, kecepatan, biaya, kualitas informasi yang dihasilkan, efisiensi dan produktivitas, ketelitian dan validitas dan kehandalan atau reliabilitas (Mulyanto, 2009).

b.  Desain Sistem

 Analisis sistem (system analysis) mendeskripsikan apa yang harus dilakukan sistem untuk memenuhi kebutuhan informasi pemakai.

 Desain sistem  (system design) menentukan bagaimana sistem akan memenuhi tujuan tersebut. Desain sistem terdiri dari aktivitas desain yang menghasilkan spesifikasi fungsional.

 Desain sistem dapat dipandang sebagai desain interface, data dan proses dengan tujuan menghasilkan spesifikasi yang sesuai dengan produk dan metode interface pemakai, struktur database serta pemrosesan dan prosedur pengendalian (Ioanna et al., 2007).

 Desain sistem akan menghasilkan paket software prototipe, produk yang baik sebaiknya mencakup tujuh bagian :

1)     Fitur menu yang cepat dan mudah.
2)     Tampilan input dan output.
3)     Laporan yang mudah dicetak.
4)     Data dictionary yang menyimpan  informasi pada setiap field termasuk panjang field, pengeditan dalam setiap laporan dan format field yang digunakan.
5)     Database dengan format dan kunci record yang optimal.
6)     Menampilkan query online secara tepat ke data yang tersimpan pada database.
7)     Struktur yang sederhana dengan bahasa pemrograman yang mengizinkan pemakai melakukan pemrosesan khusus, waktu kejadian, prosedur otomatis dan lain-lain.

C.  Pengujian Sistem

 Paket software prototipe diuji, diimplementasikan, dievaluasi dan dimodifikasi berulang-ulang hingga dapat diterima pemakainya (O'Brien, 2005). Pengujian sistem bertujuan menemukan kesalahan-kesalahan yang terjadi pada sistem dan melakukan revisi sistem.

 Tahap ini penting untuk memastikan bahwa sistem bebas dari kesalahan (Mulyanto, 2009).

Menurut Sommerville (2001) pengujian sistem terdiri dari :

1)     Pengujian unit untuk menguji komponen individual secara independen tanpa komponen sistem yang lain untuk menjamin sistem operasi yang benar.
2)     Pengujian modul yang terdiri dari komponen yang saling berhubungan.
3)     Pengujian sub sistem yang terdiri dari beberapa modul yang telah diintegrasikan.
4)     Pengujian sistem untuk menemukan kesalahan yang diakibatkan dari interaksi antara subsistem dengan interfacenya serta memvalidasi persyaratan fungsional dan non fungsional.
5)     Pengujian penerimaan dengan data yang dientry oleh pemakai dan bukan uji data simulasi.
6)     Dokumentasi berupa pencatatan terhadap setiap langkah pekerjaan dari awal sampai akhir pembuatan program.

 Pengujian sistem informasi berbasis web dapat menggunakan teknik dan metode pengujian perangkat lunak tradisional. Pengujian aplikasi web meliputi pengujian tautan, pengujian browser, pengujian usabilitas, pengujian muatan, tegangan dan pengujian malar  (Simarmata, 2009).

 Penerimaan pengguna (user) terhadap sistem dapat dievaluasi dengan mengukur kepuasan user terhadap sistem yang diujikan. Pengukuran kepuasan meliputi tampilan sistem, kesesuaian dengan kebutuhan user, kecepatan dan ketepatan sistem untuk menghasilkan informasi yang diinginkan user. Ada beberapa model pengukuran kepuasan user terhadap sistem, diantaranya adalah Technology Acceptance Model (TAM), End User Computing (EUC) Satisfaction, Task Technology Fit (TTF) Analysis dan  Human Organizational Technology (HOT) Fit Model.

 Salah satu model pengukuran yang telah diterjemahkan ke dalam beberapa bahasa berbeda dan tidak menunjukkan perbedaan hasil pengukuran yang signifikan adalah End User Computing (EUC) Satisfaction. Model ini menekankan kepuasan user terhadap aspek teknologi meliputi aspek isi, keakuratan, format, waktu dan kemudahan penggunaan sistem (Chin & Mathew, 2000).

D.  Implementasi
Setelah prototipe diterima maka pada tahap ini merupakan implementasi sistem yang siap dioperasikan dan selanjutnya terjadi proses pembelajaran terhadap sistem baru dan membandingkannya dengan sistem lama, evaluasi secara teknis dan operasional serta  interaksi pengguna, sistem dan teknologi informasi.

 Perancangan sistem membutuhkan peralatan berupa alat alat perancangan proses dan  alat perancangan data. Alat perancangan proses terdiri dari diagram aliran data dan diagram arus sistem. Sedangkan alat perancangan data terdiri dari diagram relasi entitas (entity relationship) dan kamus data (data dictionary).

A.      Diagram Aliran Data
 Diagram aliran data (data flow diagram/DFD) adalah sebuah alat dokumentasi grafik yang menggunakan simbol-simbol untuk menjelaskan sebuah proses. Diagram ini menunjukkan aliran proses seluruh sistem kepada pemakai dan dapat diatur detailnya sesuai dengan kemampuan pemahaman pemakai.

 DFD terdiri dari tiga elemen yaitu lingkungan, pemrosesan, aliran data dan penyimpanan data. Salah satu keuntungan menggunakan DFD adalah memudahkan pemakai yang kurang menguasai bidang komputer untuk mengerti sistem yang sedang akan dikerjakan (Ladjamudin, 2005).

B.     Diagram Arus Sistem
 Diagram arus sistem (Sistem Flow chart) adalah peralatan yang digunakan untuk menggambarkan proses sistem secara rinci untuk menggambarkan aliran sistem informasi dan diagram arus sistem untuk menggambarkan aliran program (Ladjamudin, 2005).

C.      Diagram Relasi Entitas
 Diagram relasi entitas menunjukkan antar entitas satu dengan yang lain dan bentuk hubungannya sehingga data tergabung dalam satu kesatuan yang terintegrasi (Ladjamudin, 2005).

D.     Kamus Data
 Kamus data adalah penjelasan tertulis lengkap dari data yang diisikan ke dalam database (Ladjamudin, 2005).

3.4 Sequence

4.1 System dan Karakteristik System

  Sistem adalah sekelompok komponen dan elemen yang digabungkan menjadi satu untuk mencapai tujuan tertentu. Ada banyak pendapat tentang pengertian dan definisi sistem yang dijelaskan oleh beberapa ahli. Berikut pengertian dan definisi sistem menurut beberapa ahli:

* Jogianto (2005:2), Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata, seperti tempat, benda dan orang-orang yang betul-betul ada dan terjadi.
* Indrajit (2001:2), Sistem adalah kumpulan-kumpulan dari komponen-komponen yang memiliki unsur keterkaitan antara satu dengan lainnya.
* Lani Sidharta (1995:9), Sistem adalah himpunan dari bagian-bagian yang saling berhubungan, yang secara bersama mencapai tujuan-tujuan yang sama.
* Murdick, R. G (1991:27), Sistem adalah seperangkat elemen yang membentuk kumpulan atau prosedur-prosedur atau bagan-bagan pengolahan yang mencari suatu tujuan bagian atau tujuan bersama dengan mengoperasikan data dan/atau barang pada waktu rujukan tertentu untuk menghasilkan informasi dan/atau energi dan/atau barang.
* Davis, G. B (1991:45), Sistem adalah kumpulan dari elemen-elemen yang beroperai bersama-sama untuk menyelesaikan suatu sasaran.

Komponen atau Karakteristik sistem adalah bagian yang membentuk sebuah sistem, diantaranya:

* Objek, merupakan bagian, elemen atau variabel. Ia dapat berupa benda fisik, abstrak atau keduanya.
* Atribut, merupakan penentu kualitas atau sifat kepemilikian sistem dan objeknya.
* Hubungan internal, merupakan penghubungan diantara objek-objej yang terdapat dalam sebuah sistem.
* Lingkungan, merupakan tempat dimana sistem berada.
* Tujuan, Setiap sistem memiliki tujuan dan tujuan inilah yang menjadi motivasi yang mengarahkan sistem. Tanpa tujuan, sistem menjadi tidak terkendali. Tentu tujuan antara satu sistem dengan sistem yang lain berbeda.
* Masukan, adalah sesuatu yang masuk ke dalam sistem dan selanjutnya menjadi bahan untuk diproses. Masukan tersebut dapat berupa hal-hal yang tampak fisik (bahan mentah) atau yang tidak tampak (jasa).
* Proses, adalah bagian yang melakukan perubahan dari masukan menjadi keluaran yang berguna dan lebih bernilai (informasi) atau yang tidak berguna (limbah)
* Keluaran, adalah hasil dari proses. Pada sistem informasi berupa informasi atau laporan, dsb
* Batas, adalah pemisah antara sistem dan daerah luar sistem. Batas disini menentukan konfigurasi, ruang lingkup atau kemampuan sistem. Batas juga dapat diubah atau dimodifikai sehingga dapat merubah perilaku sistem.
* Mekanisme pengendalian dan umpan balik, digunakan untuk mengendalikan masukan atau proses. Tujuannya untuk mengatur agar sistem berjalan sesuai dengan tujuan.

4.2 Sistem Informasi Berbasis Komputer (CBIS)



**Definisi Sistem**adalah suatu kesatuan usaha yang terdiri dari bagian-bagian yang berkaitan satu sama lain yang berusaha mencapai suatu tujuan dalam suatu lingkungan yang kompleks atau merupakankumpulan dari beberapa elemen yang saling berintegrasi untuk mencapai tujuan tertentu. Elemen-elemen yang mewakili suatu sistem secara umum adalah masukan *(input),*pengolahan *(processing)*dan keluaran *(output).*Elemen-elemen sistem secara garis besar dapat digambarkan sebagai berikut :

**OUTPUT**    —–>    **PROCESSING —–>**  **INPUT**

Sistem mempunyai karakteristik atau sifat – sifat tertentu, yaitu :

1. Komponen Sistem

2. Batasan Sistem

3. Lingkungan Luar Sistem

4. Penghubung Sistem

5. Masukan Sistem

6. Keluaran Sistem

7. Pengolahan Sistem

8. Sasaran Sistem

**Informasi** adalah data yang diolah menjadi bentuk yang berguna dan menjadi  berarti bagi       penerimanya. Kegunaan informasi adalah untuk mengurangi ketidakpastian di dalam         proses pengambilan keputusan tentang suatu      keadaan. Suatu informasi dikatakan bernilai bila manfaatnya lebih efektif           dibandingkan dengan biaya untuk mendapatkan informasi   tersebut.

Kualitas informasi sangat dipengaruhi atau ditentukan oleh beberapa hal yaitu :

*a.*Relevan *(Relevancy)*

*b.*Akurat *(Accurancy)*

*c.*Tepat waktu *(Time liness)*

d. Ekonomis *(Economy)*

*e.*Efisien *(Efficiency)*

f. Ketersediaan *(Availability)*

g. Dapat dipercaya *(Reliability)*

h. Konsisten

***Computer Based Information System*(CBIS)** atau yang dalam Bahasa Indonesia disebut juga Sistem Informasi Berbasis Komputer merupakan sistem pengolah data menjadi sebuah informasi yang berkualitas, berguna bagi penerimanya, dan dipergunakan untuk suatu alat bantu pengambilan keputusan. Sistem Informasi “berbasis komputer” mengandung arti bahwa komputer memainkan peranan penting dalam sebuah sistem informasi.

**Contoh aplikasi Sistem Informasi Berbasis Komputer :**

**(1). SIA /SISTEM INFORMASI AKUNTASI**

Sistem informasi akuntansi melaksanakan akuntansi perusahaan, aplikasi ini ditandai dengan penngolahan data yang tinggi.

Tujuan Pengolahan Data –> mengumpulkan data yang menjelaskan kegiatan perusahaan, mengubah data tersebut menjadi informasi serta menyediakan informasi bagi pemakai didalam maupun di luar perusahaan.

SIA melaksanakan 4 tugas dasar :

* pengumpulan data
* manipulasi data
* pengklasifikasian, penyortiran, perhitungan, pengikhtisaran, penyiapan dokumen.
* penyimpanan data
* penyiapan data

Contoh Sistem Informasi Akuntansi :
Sistem terdistribusi yang digunakan perusahaan distribusi yaitu perusahaan yang mendistribusikan produk dan jasanya ke pelanggan (mis : perusahaan yang berorientasi produk seperti : manufaktur, pedagang besar, pengecer dll).

**(2). SIM /SISTEM INFORMASI MANAJEMEN**

Sistem berbasis komputer yang menyediakan informasi bagi pemakai dengan kebutuhan yang serupa dan integrasi manusia/mesin guna menyediakan informasi untuk mendukung fungsi operasional manajemen & pengambilan keputusan pada suatu organisasi.

Tujuan SIM —> Memenuhi kebutuhan informasi umum semua manajer dalam perusahaan atau dalam sub unit organisasional perusahaan (subunit dapat disasarkan pada area fungsional atau tingkatan manajemen).

**(3). SPK /SISTEM PENUNJANG KEPUTUSAN**

SPK –> Sistem komputer yang interaktif yang membantu pembuat keputusan dalam menggunakan & memanfaatkan data & model untuk memecahkan masalah yang tidak terstruktur.

Tujuan :

* Memberikan dukungan untuk pembuatan keputusan pada masalah yang semi/tidak terstruktur.
* Memberikan dukungan pembuatan keputusan kepada manajer pada semua tingkat untuk membantu integrasi antar tingkat.
* Meningkatkan efektifitas manajer dalam pembuatan keputusan & bukan peningkatan efisiennya.

**(4). OTOMATISASI KANTOR / OFFICE AUTOMATION / OA**

Semua sistem elektronik formal & informal terutama yang berkaitan dengan komunikasi informasi ke dan dari orang-orang di dalam maupun di luar perusahaan.

Fungsi OA adalah —> Untuk memudahkan segala jenis komunikasi baik lisan maupun tulisan & menyediakan informasi yang lebih baik untuk pengambilan keputusan.

**(5). SISTEM INFORMASI MANUFAKTUR /MANUFACTURING INFORMATION SYSTEM**

Merupakan subsistem SIM yang menyediakan informasi untuk digunakan dalam pemecahan masalah manufaktur.

contoh SI manufaktur

* subsistem produk menelusuri arus suatu pekerjaan, perusahaan merakit lampu sepeda, senter ini dirakit dari beberapa bagian
* jadual produksi, menentukan langkah-langkah proses produksi yang akan dilakukan menentukan status sehingga pekerjaannya dapat ditanyakan.

**Sistem Pakar**



**Definisi Sistem pakar menurut para ahli :**

* Menurut Durkin Sistem pakar adalah suatu program komputer yang dirancang untuk memodelkan kemampuan penyelesaian masalah yang dilakukan seorang pakar.
* Menurut Ignizio Sistem pakar adalah suatu model dan prosedur yang berkaitan, dalam suatu  domain  tertentu,  yang  mana  tingkat  keahliannya  dapat dibandingkan dengan keahlian seorang pakar.
* Menurut Giarratano dan Riley  Sistem pakar adalah suatu sistem komputer yang bisa menyamai atau meniru kemampuan seorang pakar
* Menurut Turban (dalam Kusrini, 2006) Sistem pakar adalah program computer yang menirukan penalaran seorang pakar dengan keahlian suatu wilayah pengetahuan tertentu.
* Menurut Martin dan Oxman (dalam Kusrini, 2006) Sistem pakar adalah sistem berbasis computer, yang menggunakan pengetahuan, fakta, dan teknik penalaran dalam memecahkan masalah yang biasanya hanya dapat dipecahkan oleh seorang pakar dalam bidang tersebut.

**Beberapa aktivitas pemecahan yang dimaksud dalama sistem pakar antara lain:** Pembuatan keputusan (decicion making), pemaduan pengetahuan (knowledge fusing), pembuatan desain (designing), perencanaan (planning), prakiraan (forecasting), pengaturan (regulating), pengendalian (cotrolling), perumusan (prescribing), penjelasan (explaining), pemberian nasihat (advising) dan pelatihan (tutoring). Selain itu sistem pakar juga dapat berfungsi sebagai asisten yang pandai dari seorang pakar (Martin dan Oxman, dalam Kusrini 2006).

**Konsep Dasar Sistem Pakar**

Sistem pakar mencakup beberapa persoalan mendasar, antara lain siapa yang disebut pakar, apa yang dimaksud dengan keahlian, bagaimana keahlian dapat ditransfer, dan bagaimana sistem bekerja. Menurut Turban dan Frenzel, konsep dasar sistem pakar terdiri atas:

1)      Kepakaran (Expertise)

Kepakaran merupakan penguasaan pengetahuan di bidang tertentu yang diperoleh dari serangkaian pelatihan, membaca atau pengalaman.

2)      Pakar (Expert)

Seorang pakar adalah orang yang memiliki pengetahuan, penilaian, pengalaman, metode khusus, serta kemampuan untuk menerapkan bakat ini dalam memberi nasihat dan memecahkan masalah.

3)      Pengalihan Kepakaran

Tujuan utama sistem pakar adalah mengalihkan kepakaran seorang pakar dalam computer yang akan digunakan oleh pihak lain yang bukan pakar, untuk menemukan solusi atas permasalahan. Pengetahuan yang disimpan dalam mesin disebut dengan nama basis pengetahuan.

4)      Penalaran (Inference)

Salah satu fitur yang harus dimiliki oleh sistem pakar adalah kemampuan untuk menalar. Jika kepakaran sudah tersimpan sebagai basis pengetahuan dan tersedia program yang mampu mengakses data, maka computer harus dapat deprogram untuk membuat inferensi. Proses kesimpulan ini dikemas dalam bentuk motor inferensi.

5)      Aturan – aturan (Rules)

Sebagian besar sistem pakar adalah sistem berbasis aturan. Aturan tersebut biasanya berbentuk IF – THEN. Aturan digunakan sebagai prosedur untuk memecahkan permasalahan.

6)      Kemampuan Penjelasan (Explanation Capability)

Kemampuan menjelaskan merupakan komponen tambahan dari sistem pakar yang berfungsi untuk memberikan penjelasan kepada user mengapa suatu pertanyaan ditanyakan oleh sistem pakar, bagaimana kesimpulan dapat diperoleh, kenapa solusi tertentu ditolak, dan apa rencana untuk mencapai solusi.

**Struktur Sistem Pakar**

            Menurut Turban dan Frenzel sistem pakar disusun oleh dua bagian utama, yaitu lingkungan pengembangan (development environment) dan linkungan konsultasi (consultation environment). Lingkungan pengembangan digunakan sebagai pembangunan sistem pakar baik dari segi pembangunan komponen maupun basis pengetahuan. Lingkungan konsultasi digunakan oleh seseorang yang bukan pakar untuk memperoleh pengetahuan pakar.

**Sistem Pengambilan Keputusan**



Dalam sebuah pengambilan keputusan seseorang sering dihadapkan pada berbagai  situasi         dan kondisi yang kompleks. Kondisi yang dimaksud yaitu bingungnya  seseorang dalam pengambilan keputusan karena suatu keputusan yang hanya satu kali dan tidak dapat diulang dimasa yang akan datang. Seseorang diharuskan untuk memikirkan dampak jangka panjang dan jangka pendek dari keputusan yang dibuat.

Menurut Mangkusubroto dan Trisnadi (dalam Marimin, ) pada prinsipnya terdapat dua basis dalam pengambilan keputusan, yaitu pengambilan keputusan berdasarkan intuisi dan pengambilan keputusan rasional, berdasarkan hasil analisis keputusan.

Pengambilan keputusan dapat melalui 2 kerangka kerja meliputi (1) pengambilan keputusan tanpa percobaan, hal ini dilakukan dengan cara menyususn secara sistematis cara kerja umum sebelum mencari solusi bagi masalah yang diharapkan (2) pengambilan keputusan berdasarkan suatu pecobaan.

Mengambil atau membuat keputusan adalah suatu proses yang dilaksanakan orang berdasarkan pengetahuan dan informasi yang ada padanya pada saat tersebut dengan harapan bahwa sesuatu akan terjadi. Keputusan dapat diambil dari alternatif-alternatif keputusan yang ada, kemudian informasi tersebut diolah dan disajikan dengan sistem penunjang keputusan. Adapun informasi terbentuk dari adanya data yang terdiri dari bilangan dan terms yang disusun, diolah dan disajiakan dengan dukungan sistem informasi manajemen. Kemudian keputusan yang diambil perlu ditindak lanjuti dengan aksi yang dalam pelaksanaannya perlu mengacu pada standar prosedur operasi dan akan membentuk kembali data, begitu seterusnya yang terjadi dalam siklus data, informasi, keputusan dan aksi.

***Artificial Intelligence***



**Kecerdasan Buatan** ([bahasa Inggris](http://id.wikipedia.org/wiki/Bahasa_Inggris): *Artificial Intelligence* atau *AI*) didefinisikan sebagai kecerdasan *entitas* ilmiah. Sistem seperti ini umumnya dianggap komputer. Kecerdasan diciptakan dan dimasukkan ke dalam suatu mesin ([komputer](http://id.wikipedia.org/wiki/Komputer)) agar dapat melakukan pekerjaan seperti yang dapat dilakukan [manusia](http://id.wikipedia.org/wiki/Manusia).

Beberapa macam bidang yang menggunakan kecerdasan buatan antara lain sistem pakar, [permainan komputer](http://id.wikipedia.org/wiki/Permainan_komputer) (*games*), [*logika fuzzy*](http://id.wikipedia.org/wiki/Logika_Fuzzy), [jaringan syaraf tiruan](http://id.wikipedia.org/wiki/Jaringan_Syaraf_Tiruan_%28Artificial_Neural_Networks%29) dan [robotika](http://id.wikipedia.org/wiki/Robot).

Walaupun AI memiliki konotasi fiksi ilmiah yang kuat, AI membentuk cabang yang sangat penting pada ilmu komputer, berhubungan dengan perilaku, pembelajaran dan adaptasi yang cerdas dalam sebuah mesin. Penelitian dalam AI menyangkut pembuatan mesin untuk mengotomatisasikan tugas-tugas yang membutuhkan perilaku cerdas. Termasuk contohnya adalah pengendalian, perencanaan dan penjadwalan, kemampuan untuk menjawab diagnosa dan pertanyaan pelanggan, serta pengenalan tulisan tangan, suara dan wajah. Hal-hal seperti itu telah menjadi disiplin ilmu tersendiri, yang memusatkan perhatian pada penyediaan solusi masalah kehidupan yang nyata. Sistem AI sekarang ini sering digunakan dalam bidang ekonomi, obat-obatan, teknik dan militer, seperti yang telah dibangun dalam beberapa aplikasi perangkat lunak komputer rumah dan video game.

### ****Contoh aplikasi AI :****

**1. Visualisasi komputer —>**Kecerdasan buatan pada bidang visualisasi komputer ini memungkinkan sebuah sistem komputer mengenali gambar sebagai *input*.

Contoh: mengenali pola pada gambar.

**2. Pengenalan Suara —->**Kecerdasan buatan pada pengenalan suara ini dapat mengenali suara manusia. Cara mengenali suara ini dengan mencocokannya pada acuan yang telah diprogramkan terlebih dahulu.

Contoh : perintah komputer dengan menggunakan suara *user.*

**3. Sistem Pakar —>**Kecerdasan buatan pada sistem pakar ini memungkinkan sebuah sistem komputer memiliki cara berpikir dan penalaran seorang ahli dalam mengambil keputusan, untuk memecahkan masalah yang ada pada saat itu.

Contoh : program komputer yang dapat mendiagnosa penyakit dengan memasukan gejala-gejala apa saja yang dialami pasien.

**4. Permainan —>**Kecerdasan buatan pada permainan ini memungkinkan sebuah sistem komputer untuk memiliki cara berpikir manusia dalam bermain.

Contoh : permainan yang memiliki fasilitas orang melawan komputer. Komputer sudah di program sedemikian rupa agar memiliki cara bermain seperti seorang manusia bahkan bisa melebihi seorang manusia.

4.3 Database

Basis data (database) adalah kumpulan data yang disimpan secara sistematis di dalam komputer yang dapat diolah atau dimanipulasi menggunakan perangkat lunak (program aplikasi) untuk menghasilkan informasi. Pendefinisian basis data meliputi spesifikasi berupa tipe data, struktur data dan juga batasan-batasan pada data yang akan disimpan. Basis data merupakan aspek yang sangat penting dalam sistem informasi karena berfungsi sebagai gudang penyimpanan data yang akan diolah lebih lanjut. Basis data menjadi penting karena dapat mengorganisasi data, menghidari duplikasi data, menghindari hubungan antar data yang tidak jelas dan juga update yang rumit.

Proses memasukkan dan mengambil data ke dan dari media penyimpanan data memerlukan perangkat lunak yang disebut dengan sistem manajemen basis data (database management system | DBMS). DBMS merupakan sistem perangkat lunak yang memungkinkan pengguna basis data (database user) untuk memelihara, mengontrol dan mengakses data secara praktis dan efisien. Dengan kata lain, semua akses ke basis data akan ditangani oleh DBMS. DBMS ini menjadi lapisan yang menghubungkan basis data dengan program aplikasi untuk memastikan bahwa basis data tetap terorganisasi secara konsisten dan dapat diakses dengan mudah.

Ada beberapa fungsi yang harus ditangani DBMS seperti mengolah pendefinisian data, menangani permintaan pengguna untuk mengakses data, memeriksa sekuriti dan integriti data yang didefinisikan oleh DBA (Database Administrator), menangani kegagalan dalam pengaksesan data yang disebabkan oleh kerusakan sistem maupun media penyimpanan (disk) dan juga menangani unjuk kerja semua fungsi secara efisien. Tujuan utama DBMS adalah untuk memberikan tinjauan abstrak data kepada pengguna. Jadi sistem menyembunyikan informasi tentang bagaimana data disimpan, dipelihara dan juga bisa diakses secara efisien. Pertimbangan efisien di sini adalah rancangan struktur data yang kompleks tetapi masih bisa digunakan oleh pengguna awam tanpa mengetahui kompleksitas strukturnya.

Untuk lebih memahami tetang perbedaan basis data dan DBMS, Anda dapat membaca artikel [Memahami Basis Data (Database), DBMS dan Program Aplikasi Di MySQL](http://www.termasmedia.com/database/mysql/473-memahami-basis-data-database-dbms-dan-program-aplikasi-di-mysql.html)

### 2. Pembagian basis data.

Menurut jenisnya, basis data dapat dibagi menjadi:

#### a. Basis data flat-file.

Basis data ini ideal untuk data berukuran kecil dan dapat dirubah dengan mudah. Pada dasarnya, basis data flat-file tersusun dari sekumpulan string dalam satu atau lebih file yang dapat diurai untuk mendapatkan informasi yang disimpan. Basis data flat-file cocok untuk menyimpan daftar atau data yang sederhana dan dalam jumlah kecil. Basis data flat-file akan menjadi sangat rumit apabila digunakan untuk menyimpan data dengan struktur kompleks walaupun dimungkinkan pula untuk itu. Beberapa kendala dalam menggunakan basis data jenis ini adalah rentan pada korupsi data karena tidak adanya penguncian yang melekat ketika data digunakan atau dimodifikasi dan juga adanya duplikasi data yang mungkin sulit dihindari. Salah satu tipe basis data flat-file adalah file CSV yang menggunakan pemisah koma untuk setiap nilainya.

#### b. Basis data relasional.

Basis data ini mempunyai struktur yang lebih logis terkait cara penyimpanan. Kata "relasional" berasal dari kenyataan bahwa tabel-tabel yang ada di basis data relasional dihubungkan satu dengan lainnya. Basis data relasional menggunakan sekumpulan tabel dua dimensi yang masing-masing tabel tersusun atas baris (tupel) dan kolom (atribut). Untuk membuat hubungan antara dua atau lebih tabel, digunakan key (atribut kunci) yaitu primary key di salah satu tabel dan foreign key di tabel yang lain. Saat ini, basis data relasional menjadi pilihan utama karena keunggulannya. Program aplikasi untuk mengakses basis data relasional menjadi lebih mudah dibuat dan dikembangkan dibandingkan dengan penggunaan basis data flat-file.

Beberapa kekurangan yang mungkin dirasakan di basis data jenis ini adalah implementasi yang lebih sulit untuk data dalam jumlah besar dengan tingkat kompleksitasnya yang tinggi. Selain itu, proses pencarian informasi juga menjadi lebih lambat karena perlu menghubungkan tabel-tabel terlebih dahulu apabila datanya tersebar di beberapa tabel. Namun, terlepas dari beberapa kekurangannya, basis data relasional telah digunakan secara luas. Saat ini, basis data relasional telah banyak dimanfaatkan oleh perusahaan-perusahaan dari skala kecil, menengah hingga besar. Beberapa basis data ternama yang ada saat ini, baik yang berasal dari sumber terbuka (open source) atau yang komersil, adalah juga basis data relasional.

### 3. Contoh beberapa basis data.

Berikut ini adalah contoh beberapa basis data relasional yang paling banyak digunakan dan sejarahnya:

#### a. Basis data MySQL.



MySQL merupakan basis data sumber terbuka yang paling popular dan banyak digunakan untuk aplikasi berbasis web seperti website dinamis dan e-commerce. Tahun 2013, MySQL merupakan basis data kedua yang paling banyak digunakan di dunia dan yang pertama untuk basis data sumber terbuka. Dilihat dari sejarahnya, MySQL dibuat tahun 1995 dan disponsori oleh perusahaan Swedia, MySQL AB. Pengembang platform MySQL adalah Michael Widenius, David Axmark dan Allan Larsson. MySQL dibuat untuk menyediakan opsi pengelolaan data yang efisien, terpercaya dan handal. Pada tahun 2000, platform MySQL berubah menjadi sumber terbuka dan mengikuti ketentuan GPL.

Penggunaan MySQL sebagai basis data utama untuk aplikasi web sering dipadukan dengan PHP sebagai bahasa skrip berorientasi obyek. MySQL adalah salah satu komponen penting dari web service solution stack LAMP (Linux, Apache, MySQL and PHP) yaitu platform pengembangan web sumber terbuka dimana Linux sebagai sistem operasi, Apache sebagai Web Server, MySQL sebagai basis data dan PHP sebagai bahasa skrip. Apabila Anda membangun blog atau website menggunakan CMS seperti Joomla, Wordpress, Drupal atau Magento, Anda sedang menggunakan MySQL sebagai solusi basis datanya. MySQL juga banyak digunakan oleh perusahaan-perusahaan besar dunia seperti Facebook, Google, Adobe, Alcatel Lucent dan juga Zappos.

Pada Januari 2008, MySQL diakuisisi oleh Sun Microsystems. Pada April 2009, terjadi pencapaian kesepakatan antara Sun Microsystems dan Oracle Corporation terkait pembelian Sun Microsystems beserta hak cipta (copyright) dan merek dagang (trademark) MySQL oleh Oracle. Namun baru pada Januari 2010, MySQL secara resmi diakuisisi oleh Oracle. Di bawah naungan Oracle Corporation, MySQL tersedia melalui skema lisensi ganda. Anda dapat menggunakan opsi lisensi sumber terbuka (GPL) selama masih mematuhi aturan lisensi tersebut. Jika Anda ingin mendistribusikan aplikasi non-GPL dimana terdapat MySQL di dalamnya, Anda dapat membeli lisensi komersial sebagai gantinya.

#### b. Basis data Oracle.



Oracle merupakan basis data relasional terkemuka yang dimiliki oleh Oracle Corporation. Oracle telah dianggap sebagai basis data terbaik untuk versi basis data komersial. Oracle sendiri tersedia dalam berbagai konfigurasi dengan cakupan tool yang dapat disesuaikan untuk perusahaan skala kecil, menengah hingga besar yang membutuhkan solusi yang terbaik dan tepat dari sebuah basis data untuk keperluan bisnisnya. Beberapa kalangan mengatakan bahwa Oracle dianggap lebih baik untuk masalah kinerja dan skalabilitas dibandingkan dengan basis data komersil lainnya.

Sejarah Oracle dimulai pertama kali pada tahun 1977 ketika versi pertamanya dikembangkan oleh SDL (Software Development Laboratories) yang di dalamnya terdapat Larry Ellison dan dua temannya, Bob Miner dan Ed Oates. Nama Oracle sendiri berasal dari nama kode (code-name) sebuah proyek yang didanai oleh CIA ketika Larry Ellison bekerja di pekerjaan sebelumnya di Ampex. Pada tahun 1979, versi komersial Oracle tersedia untuk pertama kali, sedangkan versi terkini Oracle adalah Oracle 12c, dimana "c" mengacu pada cloud computing (komputasi awan). Cloud computing merupakan refleksi kerja Oracle untuk memperluas basis datanya yang memungkinkan perusahaan mengkonsolidasi dan mengelola basis data sebagai cloud service.

#### c. Basis data Microsoft SQL Server.



Dari namanya, Anda bisa menebak dengan mudah siapa pemilik basis data ini. Benar, Microsoft tidak hanya terkenal dengan perangkat lunak sistem operasi Microsot Windows dan Microsoft Windows Server, aplikasi perkantoran Microsoft Office dan programming language suite Microsoft Visual Studio, tetapi juga dengan basis datanya, Microsoft SQL Server. Microsoft SQL Server adalah basis data relasional yang bersifat komersial. Tidak seperti Microsoft Office Access yang peruntukannya untuk komputer dekstop, Microsoft SQL Server menyediakan layanan pengelolaan basis data untuk kelas perusahaan dan juga alat intelijen bisnis terpadu (integrated business intelligence (BI) tools).

Banyak perusahaan skala kecil, menengah dan besar dalam menjalankan server-nya menggunakan sistem operasi Microsoft Windows Server. Microsoft SQL Server tentu menjadi pilihan bagi pengguna Microsoft Windows Server karena selain mudah digunakan dan faktor ketersediaan, juga telah terintegrasi dengan Microsoft Windows Server. Versi terbaru yaitu Microsoft SQL Server 2016 tersedia untuk platform on-premises (yang dikelola di server lokal) dan juga sebagai cloud database. Versi terbaru ini juga hadir dengan fitur baru dan sejumlah peningkatan yang akan memberikan terobosan kinerja, keamanan lanjutan serta pelaporan dan analisis terpadu.

Awal sejarah Microsoft SQL Server dimulai ketika Microsoft dan Sybase mengembangkan sistem manajemen basis data yang diberi nama Sybase SQL Server untuk bisa berkompetisi dengan basis data yang sudah ada saat itu seperti IBM dan Oracle. Sybase akan mendapatkan hak penjualan dan keuntungan dari versi produk basis data yang tidak didesain untuk platform Microsoft. Pada tahun 1989, basis data pertama dirilis. Kemudian Sybase menjual hak ciptanya tersebut ke Microsoft dan akhirnya Microsoft memberi nama ulang basis data tersebut menjadi Microsoft SQL Server.

#### d. Basis data MariaDB.



Mungkin sebagian dari Anda masih asing atau penah mendengar nama basis data ini. MariaDB merupakan sistem basis data relational yang sepenuhnya sumber terbuka. Semua kode sumber basis data MariaDB dirilis di bawah lisensi GPL, LGPL atau BSD. MariaDB sebenarnya adalah fork dari basis data MySQL. Fork di sini dapat diartikan sebagai proyek terkait yang dapat dianggap sebagai versi lain (mini) dari MySQL standar. MariaDB merupakan pengembangan versi lain MySQL oleh Michael Widenius yang telah meninggalkan Sun Microsystems setelah diakuisisi secara resmi oleh Oracle Corporation. Baca sejarah dan fork MySQL di artikel [Mengenal Database MySQL Dan Keunggulannya](http://www.termasmedia.com/database/mysql/555-mengenal-database-mysql-dan-keunggulannya.html).

Kepopuleran basis data ini terus merangkak naik. Ini terbukti pada tahun 2013, Red Hat Enterprise Linux (RHEL) telah menyingkirkan MySQL dan kemudian menggantinya dengan MariaDB. Demikian juga dengan Fedora yang lebih memilih MariaDB dibandingkan MySQL untuk rilis Fedora 19. Rilis stabil terbaru MariaDB adalah versi 10.1 yang juga dikenal dengan MariaDB Enterprise 2016. Di rilis stabil terbarunya ini, MariaDB memiliki proteksi yang jauh lebih baik dengan pengamanan basis data pada setiap lapisan. MariaDB dapat mendeteksi dan mencegah serangan dengan menggunakan access control, Key Management Services dan authentication.

**DAFTAR PUSTAKA**

1. Abdul Kodir, Pengantar Sistem Informasi
2. Yogiyanto H.M, Pengenalan Komputer
3. Raymond Mc.Cleod,Jr. Management Information System 10 Editions Mc.Graw Hill