

PERANCANGAN SISTEM INFORMASI BADAN EKSEKUTIF MAHASISWA POLPOS BERBASIS WEBSITE



Anisa Rosalia

Yola Vegita

1.18.4.016

1.18.4.001

Program Sarjana Terapan Teknik
Informatika

Politeknik Pos Indonesia

Bandung

2019

KATA PENGANTAR

Assalamualaikum warahmatullahi wabarakatuh. Segala puji bagi Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya, sehingga kami dapat menyelesaikan laporan dengan judul **WEBSITE SISTEM INFORMASI BADAN EKSEKUTIF MAHASISWA POLITEKNIK POS INDONESIA**. Penyusunan laporan ini dimaksudkan untuk memenuhi tugas D4 teknik informatika yaitu Proyek 2. Dalam penyusunan laporan ini, kami mengucapkan kepada pihak yang telah membantu atau membimbing kami dalam penyusunan makalah ini. Laporan ini disusun untuk memenuhi kelulusan matakuliah Proyek 2 pada Program Studi DIV Teknik Informatika. Proses Proyek 2 ini juga tidak terlepas dari bantuan pihak Pembimbing. Oleh karena itu, pada kata pengantar ini penulis menyampaikan terimakasih kepada :

1. Cahyo Prianto, S.Pd., M.T. selaku Pembimbing Internal dan Penguji Utama dalam penyusunan laporan Proyek I ini;
2. Nisa Hanum Harani, S.Kom., M.T. selaku Penguji Pendamping dalam Sidang Proyek I ini;
3. Rolly Maulana Awangga, S.T., M.T. selaku Koordinator Proyek 2 Tahun Akademik 2018/2019;
4. M. Yusril Helmi Setyawan, S.Kom., M.Kom. selaku Ketua Program Studi DIV Teknik Informatika Tahun Akademik 2018/2019;
5. Dr. Ir. Agus Purnomo, M.T. selaku Direktur Politeknik Pos Indonesia Tahun Akademik 2018/2019.

Penulis telah membuat laporan ini dengan sebaik-baiknya, diharapkan memberikan kritik dan saran dari semua pihak yang bersifat membangun, terimakasih.

Bandung, 21 Januari 2020

Penulis

BAB 1

PENDAHULUAN

1.1 Latar Belakang

Di era modern seperti sekarang ini Teknologi Informasi (IT) menjadi kebutuhan bagi masyarakat pada umumnya. Tidak hanya orang dewasa, bahkan anak yang masih di bawah umur pun tidak luput dengan kemajuan di bidang Teknologi ini, Tingginya kebutuhan masyarakat akan informasi dengan adanya website-website yang memberikan apa yang dibutuhkan oleh masyarakat. Perkembangan aplikasi web yang semakin pesat sejak munculnya teknologi internet sangat membantu dalam kemudahan serta kecepatan pengiriman, penyampaian dan penerimaan informasi. Mulai dari perusahaan-perusahaan, sekolah-sekolah, perguruan tinggi, dan lembaga atau organisasi lainnya telah banyak memanfaatkan aplikasi web dalam kegiatan penjualan, promosi, belajar dan kegiatan lainnya dimana dibutuhkan pengiriman, penyebaran dan penerimaan informasi sehingga memberikan kemudahan bagi pengguna (user) yang membutuhkan. Sistem informasi web ini digunakan untuk membuat daftar penerimaan anggota baru, serta informasi mengenai acara-acara yang ada di kampus politeknik pos Indonesia sehingga anggota dapat mengelola secara online. Menyadari akan hal ini, timbul ide untuk membangun suatu aplikasi sistem informasi yang berbasis web pada kampus Politeknik Pos Indonesia. Aplikasi ini diharapkan

membantu Organisasi dan calon anggota baru untuk memanfaatkan aplikasi yang akan dibangun. Oleh karena itu, dibutuhkan suatu perancangan website yang dapat memberikan pelayanan kepada anggota khususnya dalam sistem online, sehingga dapat membantu dan mempermudah anggota dalam mengakses informasi. Untuk menjawab asumsi diatas, maka penulis menganggap perlunya suatu perancangan website yaitu Aplikasi Sistem Informasi Organisasi Kampus Politeknik Pos Indonesia. Rancangan ini diharapkan dapat menyajikan informasi yang lebih cepat dan akurat. Berdasarkan latar belakang di atas, maka penulis mengangkat judul **WEBSITE SISTEM INFORMASI BADAN EKSEKUTIF MAHASISWA POLITEKNIK POS INDONESIA**

1.2 Identifikasi Masalah

Berdasarkan hasil analisis yang telah dilakukan, maka didapat hasil identifikasi permasalahan sebagai berikut :

1. Belum adanya aplikasi yang berfungsi untuk mendukung Sistem Informasi di poltekpos sehingga mahasiswa terkadang tidak tahu jika ada event di poltekpos
2. Membuat kampus agar lebih Up To Date

1.3 Rumusan Masalah

Berdasarkan identifikasi masalah yang sudah diuraikan, maka dirumuskan:

1. Bagaimana sistem informasi unit kegiatan mahasiswa POLTEKPOS yang sedang berjalan
2. Bagaimana perancangan sistem informasi unit kegiatan mahasiswa POLTEKPOS
3. Bagaimana pengujian sistem informasi unit kegiatan mahasiswa POLTEKPOS
4. Bagaimana implementasi sistem informasi unit kegiatan mahasiswa POLTEKPOS

1.4 Maksud Penelitian

Maksud dari penelitian ini adalah mempermudah bagi mahasiswa yang ingin bergabung dengan Sistem Informasi Organisasi yang berada di Kampus Poltekpos, dan juga memudahkan pengurus Organisasi agar

mahasiswa lebih mudah mengetahui informasi apa saja yang terdapat di poltekpos. Tujuan dari perancangan ini adalah sebagai berikut :

1. Untuk mengetahui sistem informasi unit kegiatan mahasiswa yang sedang berjalan di POLTEKPOS
2. Untuk membuat perancangan sistem informasi unit kegiatan mahasiswa yang sedang berjalan di POLTEKPOS
3. Untuk melakukan pengujian sistem informasi unit kegiatan mahasiswa yang sedang berjalan di POLTEKPOS
4. Untuk melakukan implementasi sistem informasi unit kegiatan mahasiswa yang sedang berjalan di POLTEKPOS

1.5 Batasan Masalah

Batasan masalah merupakan hal yang sangat penting agar pembahasan dalam penyusunan laporan ini memiliki tujuan yang jelas dan tidak menyimpang dari judul serta memberikan hasil sesuai dengan yang diinginkan, maka penulis membatasi pembahasan permasalahan yaitu:

1. Sistem ini hanya menangani event yang akan diadakan di POLTEKPOS
2. Sistem ini hanya dikelola oleh pengurus POLTEKPOS yaitu ORMAWA
3. Program pendaftaran ini hanya untuk mahasiswa aktif POLTEKPOS
4. Pada program ini hanya sampai anggota lulus atau keluar dari POLTEKPOS
5. Laporan yang dibuat adalah laporan mengenai Sistem Informasi Organisasi POLTEKPOS

BAB 2

LANDASAN TEORI

2.1 Tinjauan Studi

Analisis ini menjadikan jurnal dari penelitian analisis dan perancangan terdahulu sebagai tolak ukur serta acuan. Adapun hasil penelitian - penelitian yang dijadikan

tolak ukur dan acuan tidak terlepas dari topik penelitian yang dilakukan. Sistem Informasi BEM yang masih kurang efektif, saat ini mahasiswa juga merasa komunikasi antara pengurus BEM dengan mahasiswa juga masih satu arah. Untuk mendapatkan Informasi mengenai BEM, mahasiswa harus mendatangi ruang sekretariat BEM dan tidak jarang pula ruang BEM tersebut lebih sering ditutup. Jika dilihat semua itu akan merugikan kedua belah pihak, yaitu pihak mahasiswa dan juga pihak kampus. Mahasiswa akan merasa keseulitan untuk mendapatkan Informasi, dan pihak BEM dalam melaksanakan kegiatannya akan menemukan kendala kurangnya partisipasi mahasiswa. Saat ini Penulis akan melakukan pengembangan sistem informasi berbasis web. Sistem yang dibuat nantinya akan dapat melakukan proses pooling, dan sarana komunikasi antara pengurus BEM, mahasiswa dan Purek. Desain Sistem Setelah tahap analisis sistem selesai dilakukan, maka analisis sistem telah mendapat gambaran dengan jelas apa yang harus dikerjakan. Selanjutnya adalah memikirkan Bagaimana membentuk sistem tersebut. Tahap ini disebut dengan desain sistem (*Systems Design*). Pada tahap desain sistem yang pertama dilakukan adalah desain database secara umum. [1] Upakan konsep pengembangan yang bisa dimanfaatkan di dalam dunia pendidikan. Selain berguna untuk menjalin hubungan silaturahmi, juga berguna untuk menunjang dalam meningkatkan efektivitas belajar, dengan cara membentuk komunitas berupa kelompok belajar dan diskusi sesuai dengan bidang dan minat bagi orang-orang yang terlibat di dalamnya. Dengan adanya website forum komunitas online ini, diharapkan akan lebih menumbuhkan rasa persatuan dan kekompakkan antara mahasiswa, serta menjadi tempat yang tepat bagi mahasiswa untuk dapat mengembangkan kreatifitas, ide-ide, berbagi pengalaman serta bertukar pikiran dalam memecahkan suatu masalah.[2] System konvensional masih digunakan BEM POLTEKPOS, Bandung seperti misalnya mahasiswa harus datang ke kampus untuk melihat informasi tentang agenda dan berita terbaru, sedangkan mahasiswa tidak semuanya selalu hadir di kampus kecuali ada kegiatan perkuliahan.

Perancangan website BEM POLTEKPOS sangat berguna untuk kegiatan BEM Poltekpos Bandung karena akan memudahkan penyampaian informasi kepada anggotanya dan mahasiswa pada umumnya, Sehingga kegiatan yang diadakan oleh BEM POLTEKPOS Bandung dapat berjalan sesuai dengan jadwal yang ditentukan. Pada penelitian ini dirancang website sistem informasi untuk memberikan informasi kepada mahasiswa dengan menggunakan bahasa pemrograman php dan database sql. Website ini dibuat bertujuan untuk memudahkan BEM dalam menyampaikan informasi kepada mahasiswa. Selain itu dengan adanya website ini penyampaian informasi akan lebih baik.[3] kegiatan kemahasiswaan saat ini banyak yang belum dapat dikontrol dan dimonitoring secara terperinci, sebab pengontrolan hanya dilakukan oleh badan pengawas masing-masing himpunan mahasiswa jurusan melalui laporan pertanggung jawaban dipertengahan dan diakhir masa kepengurusan secara hard-file. Pengawasan tersebut masih mempunyai kelemahan, yang apabila laporan pertanggungjawaban dikumpulkan dari periode kepengurusan satu ke periode kepengurusan berikutnya, maka penumpukkan data hard-file tersebut akan membutuhkan tempat penyimpanan seperti almari yang banyak. Sedangkan ruangan tidak akan efektif apabila terlalu banyak tumpukan data. Selain itu pengawasan anggaran dari setiap kegiatan yang dilakukan oleh masing-masing lembaga kemahasiswaan masih terdapat banyak kelemahan. Sistem ini sangat dibutuhkan oleh Badan Audit Keuangan di tingkat Fakultas. BAK adalah lembaga tinggi independen yang dibentuk untuk melakukan mekanisme audit keuangan terhadap lembaga kemahasiswaan. Sistem ini sangat dibutuhkan oleh Badan Audit Keuangan, karena merupakan sarana yang dapat mendapatkan informasi tentang kegiatan dan anggaran dengan lebih efisien, cepat, dan akurat. Sistem informasi ini merupakan sistem informasi berbasis web dan android client, sistem informasi monitoring kegiatan mahasiswa ini selain dapat diakses dengan computer atau web browser, sistem ini juga memanfaatkan teknologi Android pada mobile untuk memudahkan akses ke sistem

informasi monitoring kegiatan mahasiswa agar lebih mudah dikunjungi dengan fitur yang mengutamakan informasi dan kecepatan akses.[4] Kualitas pelayanan dalam penyebaran informasi yang ada di UMK telah menjadi kebutuhan yang sangat penting, karena hal tersebut berdampak terhadap tingkat kepuasan mahasiswa dan masyarakat luar. Pada akhirnya peningkatan kualitas pelayanan tersebut menjadi indikator keberhasilan UMK dalam meningkatkan Citra di dunia internet dan sekaligus menjadi media publikasi adanya UMK. Penelitian ini menghasilkan perancangan dan website berbasis Content Management Systems. Permasalahan informasi yang kurang mutakhir dan tidak akurat telah diatasi dengan dibentuk sebuah tim pengelolaan manajemen website. Telah dibuat prosedur dan diagram dalam mengelola email. [5]

Tujuan penelitian system informasi kegiatan kemahasiswaan pada bem poltekpos ini di maksudkan dapat memenuhi tujuan (*obyektif*) dari pengolahan data itu sendiri seperti :

memberikan kemudahan bagi pengurus dalam menyampaikan informasi kegiatan yang akan diadakan oleh badan eksekutif mahasiswa. Metodologi penelitian memakai observasi dan wawancara secara langsung. Bahan dan materi yang didapat berupa profil badan eksekutif mahasiswa , informasi anggota bem, informasi kegiatan yang akan diadakan oleh BEM. Metode pengembangan system yang digunakan dalam perancangan dan pembuatan perangkat ini adalah metode Waterfal, desain website ini terdiri dari beberapa unsur yaitu teks, gambar, bahan dan materi penelitian yang dibuat dan diolah menggunakan program aplikasi Macromedia Dreamweaver MX, XAMPP, dan menggunakan database MySQL. [6]

Pengertian Sistem Informasi Sistem adalah himpunan sesuatu benda nyata atau abstrak (*a set of things*) yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, dan saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan (*unity*) untuk mencapai tujuan tertentu secara efisien dan efektif

1. Informasi adalah data yang sudah diolah, dibentuk, atau dimanipulasi sesuai dengan keperluan tertentu.
2. Sistem Informasi adalah sekumpulan komponen yang saling berhubungan, mengumpulkan (atau mendapatkan), memproses, menyimpan, dan mendistribusikan informasi untuk menunjang pengambilan keputusan dan pengawasan dalam suatu organisasi.
3. Komponen-komponen yang menyusun suatu sistem informasi terdiri dari :

1. Perangkat keras, yaitu perangkat keras komponen untuk melengkapi kegiatan memasukkan data, memproses data, dan keluaran data.
2. Perangkat lunak, yaitu program dan instruksi yang diberikan ke komputer.
3. Basis Data, yaitu kumpulan data dan informasi yang diorganisasikan sedemikian rupa sehingga mudah diakses pengguna sistem informasi.
4. Telekomunikasi, yaitu komunikasi yang menghubungkan antara pengguna sistem dengan sistem komputer secara bersama-sama ke dalam suatu jaringan kerja yang efektif.
5. Manusia, yaitu personal dari sistem informasi, meliputi admin, analis, programmer dan operator.
6. Prosedur, yakni tata cara yang meliputi strategi, kebijakan, metode, dan peraturan-peraturan dalam menggunakan sistem informasi berbasis komputer

2.2 Tinjauan Pustaka

2.2.1 Sistem

Sistem adalah kumpulan / group / komponen apapun baik fisik yang saling berhubungan satu sama lain dan bekerja sama secara harmonis untuk mencapai satu tujuan tertentu. Terdapat dua kelompok pendekatan didalam mendefinisikan system, yaitu yang menekankan pada prosedurnya dan yang menekankan pada komponen atau elemennya. Pendekatan system yang lebih menekankan pada prosedur mendefinisikan system sebagai berikut ini :

Suatu Sistem adalah suatu jaringan kerja dari prosedur-prosedur yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu Pendekatan system yang lebih menekankan pada elemen atau komponennya mendefinisikan system sebagai berikut ini : Sistem adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu Kedua kelompok definisi tersebut adalah benar dan tidak bertentangan, yang berbeda adalah cara pendekatannya. Pendekatan system yang merupakan kumpulan elemen-elemen atau komponen-komponen atau subsistem subsistem merupakan definisi yang lebih luas. Definisi ini lebih banyak diterima, karena kenyataannya suatu system dapat terdiri dari beberapa subsistem atau system bagian.. Sebagai misal, system akuntansi dapat terdiri dari beberapa subsistem-subsistem, yaitu subsistem akuntansi penjualan, subsistem akuntansi.

1. Komponen Sistem (*Components*)

Suatu system terdiri dari sejumlah komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen-komponen system atau elemen-elemen system dapat berupa suatu subsistem atau bagian-bagian dari system. Setiap system tidak peduli betapapun kecilnya, selalu mengandung komponen-komponen atau subsistem-subsistem. Setiap subsistem mempunyai sifat-sifat dari system untuk menjalankan suatu fungsi tertentu dan mempengaruhi proses system secara keseluruhan. Jadi, dapat dibayangkan jika dalam suatu system ada subsistem yang tidak berjalan / berfungsi sebagaimana mestinya. Tentunya system tersebut tidak akan berjalan mulus atau mungkin juga system tersebut rusak sehingga dengan sendirinya tujuan system tersebut tidak tercapai.

2. Batas Sistem (*Boundary*)

Merupakan daerah yang membatasi antara suatu system dengan system yang lainnya atau dengan lingkungan luarnya. Atau menurut Azhar Susanto Batas Sistem merupakan garis abstraksi yang memisahkan antara system

dan lingkungannya. Batas system ini bagi setiap orang sangat relative dan tergantung kepada tingkat pengetahuan dan situasi kondisi yang dirasakan oleh orang yang melihat system tersebut. Batas system ini memungkinkan suatu system dipandang sebagai satu kesatuan. Batas suatu system menunjukkan ruang lingkup (*scope*) dari system tersebut.

3. Lingkungan Luar Sistem (*Environments*)

Lingkungan luar dari suatu system adalah apapun diluar batas dari system yang mempengaruhi operasi system. Lingkungan luar system dapat bersifat menguntungkan dan dapat juga bersifat merugikan system tersebut. Lingkungan luar yang menguntungkan merupakan energi dari system dan dengan demikian harus tetap dijaga dan dipelihara. Sedang lingkungan luar yang merugikan harus ditahan dan dikendalikan, kalau tidak maka akan mengganggu kelangsungan hidup dari system.

4. Penghubung (*Interface*)

Sistem Penghubung system merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari satu subsistem ke yang lainnya. Keluaran output dari satu subsistem akan menjadi masukan (*input*) untuk subsistem lainnya dengan melalui penghubung. Dengan penghubung satu subsistem dapat berintegrasi dengan subsistem yang lainnya membentuk satu kesatuan.

5. Masukan (*Input*)

Sistem Masukan system adalah energi yang dimasukkan kedalam system. Masukan dapat berupa masukan perawatan (*maintenance input*) dan masukan sinyal (*signal input*). *Maintanance input* adalah energi yang dimasukkan supaya system tersebut dapat beroperasi. *Signal input* adalah energi yang diproses untuk didapatkan keluaran. Sebagai contoh didalam system computer, program adalah *maintenance input* yang digunakan untuk mengoperasikan komputernya dan data adalah *siganal input* untuk diolah menjadi informasi.

6. Keluaran (*Output*)

Sistem Keluaran system adalah hasil dari energi yang diolah dan diklasifikan menjadi keluaran yang berguna dan sisi pembuangan. Keluaran dapat merupakan masukan untuk subsistem yang lain atau kepada supersistem. Misalnya untuk system computer, panas yang dihasilkan adalah keluaran yang tidak berguna dan merupakan hasil sisa pembuangan, sedang informasi adalah keluaran yang dibutuhkan.

7. Pengolah (*Process*)

Sistem Suatu system dapat mempunyai suatu bagian pengolah yang akan merubah masukan menjadi keluaran. Suatu system produksi akan mengolah masukan berupa bahan baku dan bahan-bahan yang lain menjadi keluaran berupa barang jadi. Sistem akuntansi akan mengolah data-data transaksi menjadi laporan-laporan keuangan dan laporan-laporan lain yang dibutuhkan oleh manajemen.

8. Sasaran (*Objectives*)

Tujuan (*Goal*) Tujuan Sistem merupakan target atau sasaran akhir yang ingin dicapai oleh suatu system. Suatu system pasti mempunyai tujuan atau sasaran. Kalau suatu system tidak mempunyai sasaran, maka operasi system tidak akan ada gunanya. Sasaran dari system sangat menentukan sekali masukan yang dibutuhkan system dan keluaran yang akan dihasilkan system. Suatu system dikatakan berhasil bila mengenai sasaran atau tujuannya.

2.2.2 Pengertian Pengembangan Sistem

Pengembangan system (*system development*) dapat berarti menyusun suatu system yang baru untuk menggantikan system yang lama secara keseluruhan atau memperbaiki system yang telah ada.

1. Alasan Pengembangan Sistem

Adanya permasalahan-permasalahan (*problems*) yang timbul di sistem yang lama yang dapat berupa Ketidakberesan yang tidak dapat beroperasi sesuai dengan yang diharapkan. Ketidakberesan ini dapat berupa :

1. Kecurangan-kecurangan disengaja yang menyebabkan tidak amannya harta kekayaan perusahaan dan kebenaran dari data menjadi kurang terjamin.
2. Kesalahan-kesalahan yang tidak disengaja yang juga dapat menyebabkan kebenaran dari data kurang terjamin.
3. Tidak efisiensinya operasi.
4. Tidak ditaatinya kebijaksanaan manajemen yang telah ditetapkan.

2. Pertumbuhan Organisasi

Pertumbuhan organisasi yang menyebabkan harus disusunnya system yang baru. Pertumbuhan organisasi diantaranya adalah kebutuhan informasi yang semakin luas, volume pengolahan data semakin meningkat, perubahan prinsip akuntansi yang baru. Karena adanya perubahan ini, maka menyebabkan system yang lama tidak efektif lagi, sehingga *system* yang lama sudah tidak dapat memenuhi lagi semua kebutuhan informasi yang dibutuhkan manajemen.

3. Untuk meraih kesempatan-kesempatan (*Opportunities*)

Teknologi informasi telah berkembang dengan cepatnya. Perangkat keras computer, perangkat lunak dan teknologi komunikasi telah begitu cepat berkembang. Organisasi mulai merasakan bahwa teknologi informasi ini perlu digunakan untuk meningkatkan penyediaan informasi sehingga dapat mendukung dalam proses pengambilan keputusan yang akan dilakukan oleh manajemen. Dalam keadaan pasar bersaing, kecepatan informasi atau efisiensi waktu sangat menentukan berhasil atau tidaknya strategi dan rencana-rencana yang telah disusun untuk meraih kesempatan-kesempatan yang ada. Bila pesaing dapat memanfaatkan teknologi ini, maka kesempatan-kesempatan

akan jatuh ke tangan pesaing. Kesempatan-kesempatan ini dapat berupa peluang-peluang pasar, pelayanan yang meningkat kepada pelanggan dan lain sebagainya.

4. Adanya instruksi-instruksi (*derivatives*)

Penyusunan system yang baru dapat juga terjadi karena adanya instruksi-instruksi dari atas pimpinan ataupun dari luar organisasi, seperti misalnya peraturan pemerintah. Berikut ini dapat digunakan sebagai indikator adanya permasalahan-permasalahan dan kesempatan-kesempatan yang dapat diraih, sehingga menyebabkan system yang lama harus diperbaiki, ditingkatkan bahkan diganti keseluruhannya. Indikator-indikator ini diantaranya adalah sebagai berikut :

1. Tanggung jawab yang tidak jelas
2. Tanggapan yang lambat terhadap langganan
3. File-file yang kurang teratur

5. Proses pengembangan system

Dengan telah dikembangkannya system yang baru, maka diharapkan akan terjadi peningkatan-peningkatan di system yang baru. Peningkatan peningkatan ini berhubungan dengan PIECES (merupakan singkatan untuk memudahkan mengingatnya), yaitu sebagai berikut:

1. *Performance* (kinerja)
peningkatan terhadap kinerja (hasil kerja) system yang baru sehingga menjadi lebih efektif. Kinerja dapat diukur dari *throughput* dan *response time*. *Throughput* adalah jumlah dari pekerjaan yang dapat dilakukan suatu saat tertentu. *Response time* adalah rata-rata waktu yang tertunda diantara dua transaksi atau pekerjaan ditambah dengan waktu *response* untuk menanggapi pekerjaan tersebut.
2. *Information* (informasi)
peningkatan terhadap kualitas informasi yang disajikan.
3. *Economy* (ekonomis)

peningkatan terhadap manfaat-manfaat atau keuntungan-keuntungan atau penurunan-penurunan biaya yang terjadi.

4. *Control* (Pengendalian)

peningkatan terhadap pengendalian untuk mendeteksi dan memperbaiki kesalahan - kesalahan serta kecurangan - kecurangan yang dan akan terjadi.

5. *Efficiency* (efisiensi)

peningkatan terhadap efisiensi operasi. Efisiensi berbeda dengan ekonomis. Bila ekonomis berhubungan dengan jumlah sumber daya yang digunakan, efisiensi berhubungan dengan bagaimana sumber daya tersebut digunakan dengan pemborosan yang paling minimum. Efisiensi dapat diukur dari outputnya dibagi dengan inputnya.

6. *Services* (pelayanan)

peningkatan terhadap pelayanan yang diberikan oleh *system*

2.2.3 Siklus Hidup Pengembangan

Sistem Pengembangan system informasi yang berbasis *computer* dapat merupakan tugas kompleks yang membutuhkan banyak sumber daya dan dapat memakan waktu berbulan-bulan bahkan bertahun-tahun untuk menyelesaikannya. Proses pengembangan system melewati beberapa tahapan dari mulai system itu direncanakan sampai dengan system tersebut diterapkan, dioperasikan dan dipelihara. Bila operasi system yang sudah dikembangkan masih timbul kembali permasalahan-permasalahan yang kritis serta tidak dapat diatasi dalam tahap pemeliharaan system, maka perlu dikembangkan kembali suatu system untuk mengatasinya dan proses ini kembali ke tahap yang pertama, yaitu tahap perencanaan system. Siklus ini disebut dengan siklus hidup suatu system (*system life II cycle*). Daur atau siklus hidup dari pengembangan system merupakan suatu bentuk yang digunakan untuk menggambarkan tahapan utama dan langkahlangkah di dalam tahapan tersebut dalam proses

pengembangannya. Dari sekian banyak siklus pengembangan system menurut beberapa penulis sejak tahun 1970 an, diambil salah satu yang akan menjadi acuan kita mengenai pengembangan system ini, yaitu menurut *John Burch, Gary Grudnitski, Information Systems, Theory and Practice (new York: John Wiley dan Sons)* yang menuliskan tahapan pengembangan system sebagai berikut:

1. Kebijakan dan perencanaan system (*System policy and planning*)
2. Pengembangan system (*system development*)
 - Analisis system (*system analysis*)
 - Desain system secara umum (*general system design*)
 - Penilaian system (*system evaluation*)
 - Desain system terinci (*detailed system design*)
 - Implementasi system (*system implementation*)
 - Manajemen system dan operasi (*system management and operation*)
3. Manajemen system dan operasi (*system management and operation*)

2.2.4 Teknik Teknik Pengembangan Sistem

Teknik-teknik dalam pengembangan sistem yang dapat digunakan antara lain sebagai berikut ini :

1. Teknik manajemen proyek, yaitu CPM (*Critical Path Method*) dan PERT (*Program Evaluation and Review Technique*) Teknik ini digunakan untuk penjadwalan waktu pelaksanaan suatu proyek
2. Teknik menemukan fakta (*fact finding techniques*)
Yaitu teknik yang dapat digunakan untuk mengumpulkan data dan menemukan fakta-fakta dalam kegiatan mempelajari sistem yang ada. Teknik-teknik ini diantaranya adalah :
 - 1) Wawancara (*interview*)
Memungkinkan analisis sistem sebagai pewawancara (*interviewer*) untuk mengumpulkan data secara tatap muka langsung dengan orang yang diwawancarai (*interviewee*).
 - 2) Observasi (*observation*)

Adalah pengamatan langsung suatu kegiatan yang sedang dilakukan yang mana pada waktu observasi analis sistem dapat ikut juga berpartisipasi dengan orang-orang yang sedang melakukan suatu kegiatan tersebut.

3) Daftar pertanyaan (*questionnaires*)

Adalah suatu daftar yang berisi dengan pertanyaan-pertanyaan untuk tujuan khusus yang memungkinkan analis sistem untuk mengumpulkan data dan pendapat dari responden-responden yang dipilih.

4) Pengumpulan sampel (*sampling*)

Pengambilan sampel adalah pemilihan sejumlah item tertentu dari seluruh item yang ada dengan tujuan mempelajari sebagian item tersebut untuk mewakili seluruh itemnya dengan pertimbangan biaya dan waktu yang terbatas.

3. Teknik analisis biaya / manfaat (*cost-effectiveness analysis* atau *cost benefit analysis*) Teknik ini menilai dari sisi kelayakan ekonomis suatu pengembangan sistem informasi.
4. Teknik untuk menjalankan rapat Selama proses pengembangan sistem dilakukan, seringkali rapat-rapat diadakan baik oleh tim pengembangan sistem sendiri atau rapat antara tim pengembangan sistem dengan pemakai sistem manajer, sehingga kemampuan analis sistem untuk memimpin atau berpartisipasi di dalam suatu rapat merupakan hal yang penting terhadap kesuksesan proyek pengembangan sistem.
5. Teknik inspeksi / walkthrough Inspeksi merupakan kepentingan dari pemakai sistem dan walkthrough merupakan kepentingan dari analis sistem. Analis sistem melakukan walkthrough untuk maksud supaya dokumentasi yang akan diserahkan kepada pemakai sistem secara teknik tidak mengalami kesalahan dan dapat dilakukan dengan diverifikasi terlebih dahulu oleh analis sistem yang lain. Pemakai sistem melakukan inspeksi untuk maksud

menilai dokumentasi yang diserahkan oleh analis sistem secara teknik tidak mengandung kesalahan.

Penyebab kegagalan pengembangan sistem :

1. Kurangnya penyesuaian pengembangan sistem
2. Kelalaian menetapkan kebutuhan pemakai dan melibatkan pemakai sistem
3. Kurang sempurnanya evaluasi kualitas analisis biaya
4. Adanya kerusakan dan kesalahan rancangan.
5. Penggunaan teknologi computer dan perangkat lunak yang tidak direncanakan dan pemasangan teknologi tidak sesuai
6. Pengembangan sistem yang tidak dapat dipelihara
7. Implementasi yang direncanakan dilaksanakan kurang baik

2.2.5 Fungsi Analisis Sistem

Analisis sistem (*system analyst*) adalah orang yang menganalisis sistem (mempelajari masalah-masalah yang timbul dan menentukan kebutuhan - kebutuhan pemakai sistem) untuk mengidentifikasi pemecahan yang beralasan. Sebutan lain untuk analisis sistem ini adalah analisis informasi (*information analyst*), analisis bisnis (*business analyst*), perancang sistem (*system designer*), konsultan sistem (*system consultant*) dan ahli teknik sistem (*system engineer*).

Analisis sistem berbeda dengan pemrogram. Pemrogram (*programmer*) adalah orang yang menulis kode program untuk suatu aplikasi tertentu berdasarkan rancangan bangun yang telah dibuat oleh analisis sistem. Akan tetapi ada juga analisis sistem yang melakukan tugas-tugas seperti pemrogram dan sebaliknya ada juga pemrogram yang melakukan tugas-tugas yang dilakukan oleh analisis sistem. Orang yang melakukan tugas baik sebagai analisis sistem maupun pemrogram disebut analisis / pemrogram (*analyst /*

programmer) atau pemrogram/ analis (*programmer/analyst*).

1. Tugas dan tanggung jawab analis sistem dan pemrogram adalah berbeda dan dapat dilihat pada table berikut :

| Pemrograman | Analisis Sistem |
|--|--|
| Tanggung jawab pemrogram terbatas pada pembuatan program computer | Tanggung jawab analis sistem tidak hanya pada pembuatan program computer saja, tetapi pada sistem secara keseluruhan. |
| Pengetahuan pemrogram cukup terbatas pada teknologi computer, sistem computer, utilities dan bahasa-bahasa pemrograman yang diperlukan | Pengetahuan analis sistem harus luas, tidak hanya pada teknologi computer, tetapi juga pada bidang aplikasi yang ditanganinya. |
| Pekerjaan pemrogram sifatnya teknis dan harus tepat dalam pembuatan instruksi-instruksi program. | Pekerjaan analis sistem dalam pembuatan program terbatas pada pemecahan masalah secara garis besar. |
| Pekerjaan pemrogram tidak menyangkut hubungan dengan banyak orang, terbatas pada sesama pemrogram dan analis sistem yang mempersiapkan rancang bangun (spesifikasi) programnya | Pekerjaan analis sistem melibatkan hubungan banyak orang, tidak terbatas pada sesama analis sistem, pemrogram, tetapi juga pemakai sistem dan manajer. |

2. Pengetahuan dan keahlian yang diperlukan Analisis Sistem

Analisis sistem harus mempunyai pengetahuan yang luas dan keahlian yang khusus. Beberapa analisis

sistem setuju bahwa pengetahuan-pengetahuan dan keahlian berikut ini sangat diperlukan bagi seorang analis sistem yang baik :

1) Pengetahuan dan keahlian tentang teknik pengolahan data, teknologi computer dan pemrograman computer:

a) Keahlian teknik yang harus dimiliki adalah termasuk keahlian dalam penggunaan alat dan teknik untuk pengembangan perangkat lunak aplikasi serta keahlian dalam menggunakan computer.

b) Pengetahuan teknik yang harus dimiliki meliputi pengetahuan tentang perangkat keras computer, teknologi komunikasi data, bahasa-bahasa computer, sistem operasi, utilites dan paket-paket perangkat lunak lainnya.

3. Pengetahuan tentang metode kuantitatif

Dalam membangun model-model aplikasi, analis sistem banyak menggunakan metode-metode kuantitatif, seperti misalnya pemrograman linier (*linier programming*), pemrograman dinamik (*dynamic programming*), regresi (*regression*), network, pohon keputusan (*decision tree*), trend, simulasi dan lain sebagainya.

4. Keahlian pemecahan masalah

Analisis sistem harus mempunyai kemampuan untuk meletakkan permasalahan-permasalahan kompleks yang dihadapi oleh bisnis, memecah-mecah masalah tersebut ke dalam bagian-bagiannya, menganalisisnya dan kemudian harus dapat merangkainya kembali menjadi suatu sistem yang dapat mengatasi permasalahan-permasalahan tersebut.

5. Keahlian komunikasi antar personil

Analisis sistem harus mempunyai kemampuan untuk mengadakan komunikasi baik secara tertulis. Keahlian ini diperlukan di dalam wawancara, presentasi, rapat dan pembuatan laporan-laporan.

6. Keahlian membina hubungan antar personil
Manusia merupakan faktor yang kritis didalam sistem dan watak manusia satu dengan yang lainnya berbeda. Analis sistem yang kaku dalam membina hubungan kerja dengan personil-personil lainnya yang terlibat, akan membuat pekerjaannya menjadi tidak efektif. Apalagi bila analis sistem tidak dapat membina hubungan yang baik dengan pemakai sistem, maka akan tidak mendapat dukungan dari pemakai sistem atau manajemen dan kecenderungan pemakai sistem akan mempersulitnya.

2.2.6 Tim Pengembangan Sistem

Dalam proyek pengembangan sistem yang kecil dan sederhana, kemungkinan hanya ada seorang analis sistem yang merangkap sebagai pemrogram (*analis/pemrogram*) atau seorang pemrogram yang merangkap sebagai analis sistem (*pemrogram/analis*). Akan tetapi untuk proyek pengembangan sistem yang besar atau kompleks, pekerjaan ini biasanya dilakukan oleh sejumlah orang dalam bentuk tim. Anggota dari tim pengembangan sistem ini tergantung dari besar kecilnya ruang lingkup proyek yang akan ditangani. Tim ini secara umum dapat terdiri dari personil-personil sebagai berikut :

1. Manajer analisis sistem Manajer anaisi sistem (*manager of system analysis*) ini disebut juga sebagai coordinator proyek dan mempunyai tugas dan tanggung jawab sebagai berikut:
 - a) Sebagai ketua/ coordinator tim pengembangan sistem
 - b) Mengarahkan, mengontrol dan mengatur anggota tim pengembangan sistem lainnya
 - c) Membuat jadwal pelaksanaan proyek pengembangan sistem yang akan dilakukan
 - d) Bertanggung jawab dalam mendefinisikan masalah, studi kelayakan, disain sistem dan penerapannya.
 - e) Memberikan rekomendasi-rekomendasi perbaikan sistem

- f) Mewakili tim untuk berhubungan dengan pemakai sistem dalam hal perundingan-perundingan dan pemberian-pemberian nasehat kepada manajemen dan pemakai sistem.
 - g) Membuat laporan-laporan kemajuan proyek (*progress report*)
 - h) Mengkaji ulang dan memeriksa kembali hasil kerja dari tim.
2. Ketua analis sistem Ketua analis sistem (*lead system analyst*) biasanya menjabat sebagai wakil dari manajer analisis sistem. Tugasnya adalah membantu tugas dari manajer analisis sistem dan mewakilinya bila manajer analis sistem berhalangan.
 3. Analis sistem senior Analis sisten senior (*senior system analyst*) merupakan analis sistem yang sudah berpengalaman.
 4. Analis sistem Analys sistem (*system analyst*) merupakan analis sistem yang cukup berpengalaman dan dapat bekerja sendiri tanpa bimbingan dari analis sistem senior.
 5. Analis sistem yunior Analis sistem yunior (*junior system analyst*) merupakan analis sistem yang belum berpengalaman dan masih membutuhkan bimbingan-bimbingan dari analis sistem yang lebih senior. Analis sistem yunior ini sering juga disebut dengan analis sistem yang masih dilatih (*system analyst trainee*).
 6. Pemrogram aplikasi senior Permograman aplikasi senior (*senior application programmer*) merupakan pemrograman computer yang sudah berpengalaman dengan tugas merancang spesifikasi dari program aplikasi dan mengkoordinasi kerja dari pemrogram yang lainnya. Pemrogram aplikasi senior ini kadangkadang juga disebut dengan pemrogram / analis.
 7. Pemrogram aplikasi Pemrogram apliaksi (*application programmer*) merupakan pemrogram computer yang cukup berpengalaman dan dapat melakukan tugasnya tanpa harus dibimbing secara langsung lagi.

8. Pemrogram aplikasi junior Pemrogram aplikasi junior (*junior application programmer*) merupakan pemrogram computer yang belum berpengalaman dan masih dibawah bimbingan langsung dari pemrogram yang lebih senior. Pemrogram aplikasi junior biasanya hanya dilibatkan pada pembuatan modul-modul program yang sederhana, seperti misalnya pembuatan bentuk-bentuk *I/O*. Pemrogram aplikasi junior ini sering juga disebut dengan pemrogram aplikasi yang masih dilatih (*application programmer trainee*).

2.2.7 Deteksi Masalah Sistem (*Detection Of System*)

1. Permasalahan Sistem

Semua sistem informasi akan mempunyai masalah, tanpa memperdulikan seberapa baiknya sistem tersebut didesain. Beberapa hal yang menyebabkan sistem informasi mempunyai masalah, antara lain karena :

- a. Waktu (*overtime*).
- b. Lingkungan sistem yang berubah.
- c. Perubahan prosedur operasional.

Perbaikan masalah sistem informasi disebut *maintenance programming*, yang meliputi tanggapan terhadap masalah sistem dan penambahan fungsi baru ke sistem. *Maintenance programming* mencakup 60 sampai 90 persen dari *programming budget* dan menunjukkan apakah sistem informasi yang memburuk perlu diganti atau dipertahankan dengan melakukan perbaikan kecil (*minor*). Masalah sistem informasi berhubungan dengan karakteristik informasi, yaitu :

2. Relevansi (*relevancy*)
3. Keakuratan (*accuracy*), yang memiliki faktor :
 - a) Kelengkapan (*completeness*)
 - b) kebenaran (*correctness*), dan keamanan (*security*)
 - c) Ketepatan waktu (*timeliness*)
 - d) Ekonomi (*economy*)

- e) biaya (*cost*)
- f) Efisiensi (*eficiency*)
- g) Dapat dipercaya (*reliability*)
- h) Kegunaan (*usability*)
- i) Relevansi (*relevancy*)

Hasil dari sistem informasi (*SI*) harus dapat digunakan untuk kegiatan manajemen ditingkat operasional, taktis dan strategik. Jika tidak dapat digunakan, informasi tersebut layak untuk tidak diperhatikan lagi. Beberapa gejala dari informasi yang tidak lagi relevan, antara lain :

1. Banyak laporan yang isinya terlalu panjang
2. Laporan tidak digunakan oleh pihak yang menerimanya.
3. Permintaan informasi tidak tersedia dalam SI.
4. Sebagai laporan yang tersedia tetapi tidak diminta/dibutuhkan.
5. Bertumpuknya keluhan-keluhan pemakai ketika laporan tidak diproduksi dan disebarluaskan.

4. Kelengkapan (*completeness*)

Data tidak hanya dimasukkan secara benar, tetapi juga harus lengkap. Apabila sebuah sistem informasi memiliki 95% keakuratan data, tetapi hanya 80% dari kebutuhan informasi, maka sistem akan tidak efektif. Berikut beberapa gejala ketidaklengkapan (*incompleteness*).

- 1) Sebagian data dikembalikan ke pemakai karena sumber dokumennya atau isian formulirnya tidak lengkap.
- 2) Pengawas data menunjukkan sebuah atau lebih isian field yang tidak diisi karena kesengajaan atau ketidaksengajaan.
- 3) Bagian pemasukan data menelepon ke pemakai untuk mengklarifikasikan data dari sumber-sumber dokumennya.

5. Kebenaran (*correctness*) Kebenaran biasanya dipikir sebagai keakuratan. Semua data dari field harus dimasukkan secara benar. Berikut gejala dari ketidakbenaran, antara lain :

- 1) Total kesalahan transaksi mengalami kenaikan dibanding kualitasnya.
- 2) Permintaan untuk perubahan program mengalami kenaikan.
- 3) Masalah yang terjadi setelah akhir hari kerja normal mengalami kenaikan.
- 4) Jumlah kesalahan kritis mengalami kenaikan.

Sebagai contoh adalah kesalahan saldo hutang nasabah dapat mengurangi masukan kas, sehingga membuat nasabah mengalami ketidakpuasan.

6. Keamanan (*security*)

Seringkali informasi dikirimkan ke setiap orang yang membutuhkannya. Pengawasan keamanan adalah struktur pengecekan untuk memutuskan jika informasi yang sensitif ditujukan kepada pemakai yang tidak sah.

7. Ketepatan waktu (*timeliness*)

Beberapa gejala yang menunjukkan masalah ketepatan waktu :

- 1) Keluaran (*throughput*) sistem informasi mengalami penurunan.
- 2) Troughput adalah tingkat proses transaksi sampai akhir waktu yang bebas kesalahan.
- 3) Tumpukan pemasukan data mengalami kenaikan.
- 4) Sebuah tumpukan pemasukan data terjadi ketika data transaksi tidak langsung dimasukkan pada saat itu (ditunda/tertunda).
- 5) Keluhan tentang lambatnya sistem membuat laporan mengalami kenaikan.
- 6) Waktu yang dibutuhkan untuk memperbaiki kesalahan program mengalami kenaikan.

- 7) Banyaknya keluhan dari pemakai tentang kesulitan dalam menghubungi staff pemeliharaan program dan staff operasinya.

8. Ekonomi (*economy*)

Biaya sistem informasi akan mengalami kenaikan sesuai dengan berjalannya waktu. Meskipun ada beberapa biaya yang mengalami penurunan, dan sebagian akan naik. Banyak hal yang menunjukkan kenaikan biaya, seperti konsultan pemeliharaan hardware dan program, dan sebagainya. Banyak organisasi merekrut konsultan sebagai programmer atau analis selama proyek. Untuk jangka pendek secara drastis akan menaikkan biaya tenaga kerja, tetapi untuk jangka panjang mengurangi biaya karena mempertimbangkan keuntungan sistem informasi yang didapat.

9. Dapat dipercaya (*reliability*)

Sebuah indikator penting dari sistem informasi yang adalah dengan memperhatikan masalah reliabilitasnya. Beberapa gejala tentang masalah reliabilitas, antara lain :

- 1) Computer downtime, yaitu sistem informasi bekerja dengan baik ketika komputernya bagus, kemudian komputer mengalami penurunan
- 2) Banyaknya karyawan mengalami pergantian (*turnover*), yaitu tingkat rata-rata karyawan bekerja dengan baik keluar, dan karyawan baru ditraining.
- 3) Waktu perbaikan kesalahan program, yaitu pemakai tidak dapat memperkirakan berapa lama waktu yang dibutuhkan untuk memperbaiki sebuah kesalahan informasi, barangkali satu jam atau empat minggu.
- 4) Biaya, yaitu tingginya varian rata-rata biaya setiap bulannya.
- 5) Tumpukan transaksi, yaitu jumlah transaksi yang tertunda atau ditolak.

- 6) Rata-rata kesalahan, yaitu rata-rata kesalahan yang tidak dapat diprediksi, sehingga perlu menguranginya.

10. Kegunaan (*usability*)

Tidak ada hal yang lebih baik dari sebuah sistem yang dirancang sesuai dengan kriteria. Jika sistem sulit digunakan, berarti adalah masalah dalam sistem. Beberapa gejala yang menunjukkan sedikit kegunaan (*poor usability*) sistem, antara lain:

- 1) Lamanya waktu pelatihan bagi pemakai pemula.
- 2) Tingginya rata-rata kesalahan yang terjadi.
- 3) Naiknya keluhan-keluhan pemakai.
- 4) Naiknya kemangkiran dari sebagian pemakai computer

2.2.8 Tahap Perancangan

Rancangan sistem adalah penentuan proses dan data yang diperlukan oleh sistem baru. Jika sistem itu berbasis komputer, rancangan dapat menyertakan spesifikasi jenis peralatann yang akan digunakan. Langkah-langkah tahapan rancangan yaitu :

1. Menyiapkan rancangan sistem yang terinci
Analisis bekerja sama dengan pemakai dan mendokumentasikan rancangan sistem baru dengan alat-alat yang dijelaskan dalam modul teknis. Beberapa alat memudahkan analisis untuk menyiapkan dokumentasi secara top-down, dimulai dengan gambaran besar dan secara bertahap mengarah lebih rinci. Pendekatan top-down ini merupakan ciri rancangan terstruktur (*structured design*), yaitu rancangan bergerak dari tingkat sistem ke tingkat subsistem. Alat-alat dokumentasi yang populer yaitu diagram arus data (*data flow diagram*), diagram hubungan entitas (*entity relationship diagram*), kamus data (*data*

dictionary), flowchart, model hubungan objek, dan spesifikasi kelas.

2. Mengidentifikasi berbagai alternatif konfigurasi sistem

Analisis mengidentifikasi konfigurasi – bukan merek atau model – peralatan komputer yang akan memberikan hasil yang terbaik bagi sistem dalam menyelesaikan pemrosesan. Identifikasi merupakan suatu proses berurutan, dimulai dengan berbagai kombinasi yang dapat menyelesaikan setiap tugas.

3. Mengevaluasi berbagai alternatif konfigurasi sistem

Analisis bekerja sama dengan manajer mengevaluasi berbagai alternatif. Alternatif yang dipilih adalah yang paling memungkinkan subsistem memenuhi kriteria kinerja, dengan kendala yang ada.

4. Memilih konfigurasi terbaik

Analisis mengevaluasi semua konfigurasi subsistem dan menyesuaikan kombinasi peralatan sehingga semua subsistem menjadi satu konfigurasi tunggal. Setelah selesai, analisis membuat rekomendasi kepada manajer untuk disetujui. Bila manajer menyetujui konfigurasi tersebut, persetujuan selanjutnya dilakukan oleh SC MIS.

5. Menyiapkan usulan penerapan

Analisis menyiapkan usulan penerapan (*implementation proposal*) yang mengikhtisarkan tugas-tugas penerapan yang harus dilakukan, keuntungan yang diharapkan, dan biayanya.

6. Menyetujui atau menolak penerapan sistem

Keputusan untuk terus pada tahap penerapan sangatlah penting, karena usaha ini akan sangat meningkatkan jumlah orang yang terlibat. Jika keuntungan yang diharapkan dari sistem melebihi biayanya, maka penerapan akan disetujui. Sedangkan contoh format untuk dokumen usulan penerapan, yaitu sebagai berikut :

- 1) Ikhtisar eksekutif
- 2) Pendahuluan
- 3) Definisi masalah
- 4) Tujuan dan kendala sistem
- 5) Kriteria kinerja
- 6) Rancangan sistem
 - Deskripsi ringkasan
 - Konfigurasi peralatan
7. Proyek penerapan yang disarankan
 - 1) Tugas-tugas yang harus dilaksanakan
 - 2) Kebutuhan sumber daya manusia
 - 3) Jadwal kerja
 - 4) Perkiraan biaya
8. Dampak yang diharapkan dari sistem
 - 1) Dampak pada struktur organisasi
 - 2) Dampak pada operasi organisasi
9. Rencana penerapan umum
10. Ikhtisar

2.2.8 Tahap Penerapan

Penerapan merupakan kegiatan memperoleh dan mengintegrasikan sumber daya fisik dan konseptual yang menghasilkan suatu sistem yang bekerja. Adapun tahapannya yaitu :

1. Merencanakan penerapan
Manajer dan spesialis informasi harus memahami dengan baik pekerjaan yang diperlukan untuk menerapkan rancangan sistem dan untuk mengembangkan rencana penerapan yang sangat rinci.
2. Mengumumkan penerapan
Proyek penerapan diumumkan kepada para pegawai dengan cara yang sama pada penelitian sistem. Tujuannya adalah untuk menginformasikan kepada

para pegawai mengenai keputusan untuk menerapkan sistem baru dan meminta kerjasama mereka.

3. Mendapatkan sumber daya perangkat keras
Rancangan sistem disediakan bagi para pemasok berbagai jenis perangkat keras yang terdapat pada konfigurasi sistem yang disetujui. Setiap pemasok diberikan request for proposal (RFP), yang berisi antara lain :

- 1) Surat yang ditransmisikan
- 2) Tujuan dan kendala sistem
- 3) Rancangan sistem : deskripsi ringkasan, kriteria kerja, konfigurasi peralatan, dokumentasi sistem ringkasan, perkiraan volume transaksi, perkiraan ukuran file.
- 4) Jadwal pemasangan

Selanjutnya mereka membuat usulan tertulis, bagaimana peralatan yang diusulkan akan membuat sistem mencapai kriteria kinerjanya. Ketika semua usulan telah diterima dan dianalisis, SC MIS memilih satu pemasok atau lebih. Spesialis informasi memberi dukungan bagi keputusan tersebut dengan mempelajari usulan dan membuat rekomendasi. Setelah disetujui, perusahaan melakukan pemesanan.

4. Mendapatkan sumber daya perangkat lunak
Ketika perusahaan memutuskan untuk menciptakan sendiri perangkat lunak aplikasinya, programmer menggunakan dokumentasi yang disiapkan oleh analis sistem sebagai titik awal. Programmer dapat menyiapkan dokumentasi yang lebih rinci seperti flowchart atau bahasa semu (*pseudo code*) yang terstruktur, dilakukan pengkodean, dan pengujian program. Hasil akhirnya adalah software library dari program aplikasi. Jika perangkat lunak aplikasi jadi (*prewritten application software*) dibeli, pemilihan

pemasok perangkat lunak dapat mengikuti prosedur yang sama seperti yang digunakan untuk memilih pemasok perangkat keras, yaitu RFP dan Usulan.

5. Menyiapkan database

Pengelola database (*database administrator – DBA*) bertanggung jawab untuk semua kegiatan yang berhubungan dengan data, dan mencakup persiapan database. Hal tersebut memerlukan pengumpulan data baru atau data yang telah ada perlu dibentuk kembali sehingga sesuai dengan rancangan sistem baru dan menggunakan sistem manajemen basis data (*database management sistem – DBMS*).

6. Menyiapkan fasilitas fisik

Jika perangkat keras dan sistem baru tidak sesuai dengan fasilitas yang ada, perlu dilakukan konstruksi baru atau perombakan. Sehingga pembangunan fasilitas tersebut merupakan tugas berat dan harus dijadualkan sehingga sesuai dengan keseluruhan rencana proyek.

7. Mendidik peserta dan pemakai

Sistem baru kemungkinan besar akan mempengaruhi banyak orang. Beberapa orang akan membuat sistem bekerja. Mereka disebut dengan peserta, yang meliputi operator entry data, pegawai coding, dan pegawai administrasi lainnya. Semuanya harus dididik tentang peran mereka dalam sistem. Pendidikan harus dijadualkan jauh setelah siklus hidup dimulai, tepat sebelum bahan-bahan yang dipelajari mulai diterapkan.

8. Menyiapkan usulan *cutover*

Proses menghentikan penggunaan sistem lama dan memulai menggunakan sistem baru disebut *cutover*. Ketika seluruh pekerjaan pengembangan hampir selesai, tim proyek merekomendasikan kepada

manajer agar dilaksanakan *cutover* (dalam memo atau laporan lisan)

9. Menyetujui atau menolak masuk ke sistem baru
Manajer dan SC MIS menelaah status proyek dan menyetujui atau menolak rekomendasi tersebut. Bila manajemen menyetujui maka manajemen menentukan tanggal *cutover*. Namun, bila manajemen menolak maka manajemen menentukan tindakan yang harus diambil dan tugas yang harus diselesaikan sebelum *cutover* akan dipertimbangkan kembali, kemudian manajemen menjadualkan tanggal baru.
10. Masuk ke sistem baru. Ada 4 pendekatan dasar (*cutover*), yaitu :
 - 1) Percontohan (*pilot*) yaitu suatu sistem percobaan yang diterapkan dalam satu subset dari keseluruhan operasi.
 - 2) Serentak (*immediate*) merupakan pendekatan yang paling sederhana yakni beralih dari sistem lama ke sistem baru pada saat yang ditentukan.
 - 3) Bertahap (*phased*), sistem baru digunakan berdasarkan bagian per bagian pada suatu waktu.
 - 4) Paralel (*parallel*), mengharuskan sistem lama dipertahankan sampai sistem baru telah diperiksa secara menyeluruh. Akan memberikan pengamanan yang paling baik terhadap kegagalan tetapi yang paling mahal, karena kedua sumber daya harus dipertahankan. *Cutover* menandakan berakhirnya bagian pengembangan dari siklus hidup sistem. Penggunaan sistem dapat dimulai sekarang.

2.2.9 Tahap Penggunaan

Tahap penggunaan terdiri dari 5 langkah, yaitu :

1. Menggunakan sistem Pemakai menggunakan sistem untuk mencapai tujuan yang diidentifikasi pada tahap perencanaan.
2. Audit sistem Setelah sistem baru mapan, penelitian formal dilakukan untuk menentukan seberapa baik sistem baru itu memenuhi kriteria kinerja. Studi tersebut dikenal dengan istilah penelaahan setelah penerapan (post implementation review). Hasil audit dilaporkan kepada CIO, SC MIS dan pemakai. Proses tersebut diulangi, mungkin setahun sekali, selama penggunaan sistem berlanjut.
3. Memelihara sistem Selama manajer menggunakan sistem, berbagai modifikasi dibuat sehingga sistem terus memberikan dukungan yang diperlukan. Modifikasinya disebut pemeliharaan sistem (*sistem maintenance*). Pemeliharaan sistem dilaksanakan untuk 3 alasan, yakni :
 - 1) Memperbaiki kesalahan
 - 2) Menjaga kemutakhiran sistem
 - 3) Meningkatkan system
4. Menyiapkan usulan rekayasa ulang Ketika sudah jelas bagi para pemakai dan spesialis informasi bahwa sistem tersebut tidak dapat lagi digunakan, diusulkan kepada SC MIS bahwa sistem itu perlu direkayasa ulang (*reengineered*). Usulan itu dapat berbentuk memo atau laporan yang mencakup dukungan untuk beralih pada suatu siklus hidup sistem baru. Dukungan tersebut mencakup penjelasan tentang kelemahan inheren sistem, statistik mengenai biaya perawatan, dan lain-lain.
5. Menyetujui atau menolak rekayasa ulang sistem Manajer dan komite pengarah SIM mengevaluasi usulan rekayasa ulang sistem dan menentukan apakah akan memberikan persetujuan atau tidak.

2.2.10 Ikhtisar

Evolusi sistem berbasis komputer mengikuti suatu pola yang disebut siklus hidup sistem, yang terdiri dari tahap perencanaan, analisis, rancangan, dan penerapan serta penggunaan. Manajer dari area pemakai terlibat dalam perencanaan dengan maksud agar proyek akan memperoleh manfaat. Analisis sistem membantu manajer dalam pendefinisian masalah, menetapkan tujuan dan mengenai kendala-kendala serta studi kelayakan.

Tahap analisis dimulai dengan pengumuman kepada para pegawai dan dibentuknya tim proyek. Kegiatan yang dilakukan yaitu pemakai mendefinisikan kebutuhan informasi, menentukan kriteria kinerja, menyiapkan usulan rancangan untuk merancang sistem baru.

Tahap rancangan mulai saat analisis terlibat dalam rancangan sistem yang terinci, dengan menggunakan teknik-teknik dan peralatan terstruktur yang mendokumentasikan proses dan data. Dilakukan pula identifikasi konfigurasi sistem alternatif dan dievaluasi, dan yang terbaik dipilih. Diajukan usulan penerapan yang akan memberi dasar untuk menciptakan suatu sistem kerja dari dokumentasi rancangan.

Tahap penerapan melibatkan para spesialis informasi lainnya, pemakai tambahan dan mungkin orang luar seperti konsultan dan kontraktor. Setelah perancangan yang terinci diperoleh perangkat keras dan perangkat lunak serta dibuat database. Ketika fasilitas fisik telah siap dan pendidikan yang diperlukan telah dilaksanakan, manajemen menentukan apakah cutover ke sistem baru perlu dilaksanakan. Apabila sistem tersebut dianggap tidak bisa digunakan lagi, pihak manajemen dapat mengotorisasi proyek rekayasa ulang, yang mengulang siklus hidup sistem. Setelah tahap penggunaan dimulai, analisis sistem dan auditor internal melaksanakan penelaahan pasca penerapan, yang diulang secara berkala sepanjang umur hidup sistem. Spesialis informasi juga melakukan

pemeliharaan sistem. Meskipun siklus hidup sistem mewakili bentuk dasar dari kerja sistem, siklus hidup sistem terpengaruh perubahan metodologi lain yang menekankan penggunaan peralatan pengembangan berbasis komputer. Salah satunya yaitu rapid application development – RAD yang menyatukan baik CASE maupun prototyping.

2.2.11 Tahap Penggunaan

Tahap penggunaan terdiri dari 5 langkah, yaitu :

1. Menggunakan sistem Pemakai menggunakan sistem untuk mencapai tujuan yang diidentifikasi pada tahap perencanaan.
2. Audit sistem Setelah sistem baru mapan, penelitian formal dilakukan untuk menentukan seberapa baik sistem baru itu memenuhi kriteria kinerja. Studi tersebut dikenal dengan istilah penelaahan setelah penerapan (*post implementation review*). Hasil audit dilaporkan kepada CIO, SC MIS dan pemakai. Proses tersebut diulangi, mungkin setahun sekali, selama penggunaan sistem berlanjut.
3. Memelihara sistem Selama manajer menggunakan sistem, berbagai modifikasi dibuat sehingga sistem terus memberikan dukungan yang diperlukan. Modifikasinya disebut pemeliharaan sistem (*sistem maintenance*). Pemeliharaan sistem dilaksanakan untuk 3 alasan, yakni :
 - 1) Memperbaiki kesalahan
 - 2) Menjaga kemutakhiran sistem
 - 3) Meningkatkan system
4. Menyiapkan usulan rekayasa ulang Ketika sudah jelas bagi para pemakai dan spesialis informasi bahwa sistem tersebut tidak dapat lagi digunakan, diusulkan kepada SC MIS bahwa sistem itu perlu direkayasa ulang (*reengineered*). Usulan itu dapat berbentuk memo atau laporan yang mencakup dukungan untuk beralih pada suatu siklus hidup sistem baru. Dukungan tersebut mencakup penjelasan tentang kelemahan inheren

sistem, statistik mengenai biaya perawatan, dan lain-lain.

5. Menyetujui atau menolak rekayasa ulang sistem Manajer dan komite pengarah SIM mengevaluasi usulan rekayasa ulang sistem dan menentukan apakah akan memberikan persetujuan atau tidak

2.2.12 Perancangan Proses Sistem

Tujuan dari Perancangan Proses Sistem adalah :

1. Untuk menjaga agar proses data lancar dan teratur sehingga menghasilkan informasi yang benar
2. Untuk mengawasi proses dari sistem

Perancangan Proses Sistem ini bisa digambarkan dengan :

1. Sistem Flowchart
2. DFD
3. Real Time
4. Batch
5. Online
6. Offline

2.2.13 Perancangan Database

Penerapan database dalam sistem informasi disebut dengan database system. Sistem basis data (database system) ini adalah suatu sistem informasi yang mengintegrasikan kumpulan dari data yang saling berhubungan satu dengan lainnya.

1. Tipe dari File

1) File Master Berisi data yang tetap dimana pemrosesan terhadap data hanya pada waktu-waktu tertentu. Terdapat 2 tipe file master :

- a) File Referensi Data yang tetap, dimana pengolahan terhadap data tersebut memerlukan waktu yang lama
- b) File Dinamik Data yang ada dalam file berubah tergantung transaksi

- 2) File Input / Transaksi Berisi data masukan yang berupa data transaksi dimana data-data tersebut akan diolah oleh komputer
- 3) File Laporan Berisi informasi yang akan ditampilkan
- 4) File Sejarah / Arsip Berisi data masa lalu yang sudah tidak aktif lagi, tetapi disimpan untuk keperluan masa datang
- 5) File Backup / Pelindung Berisi salinan data-data yang masih aktif di database pada suatu waktu tertentu
- 6) File Kerja / Temporary File Berisi data-data hasil pemrosesan yang bersifat sementara
- 7) File Library Berisi program-program aplikasi atau utility program

2. Akses File

Metode yang menunjukkan bagaimana suatu program komputer akan membaca record-record dari suatu file. File dapat diakses dengan 2 cara, yaitu :

- 1) Sequential (urut)
- 2) Direct / Random (langsung)

3. Organisasi File

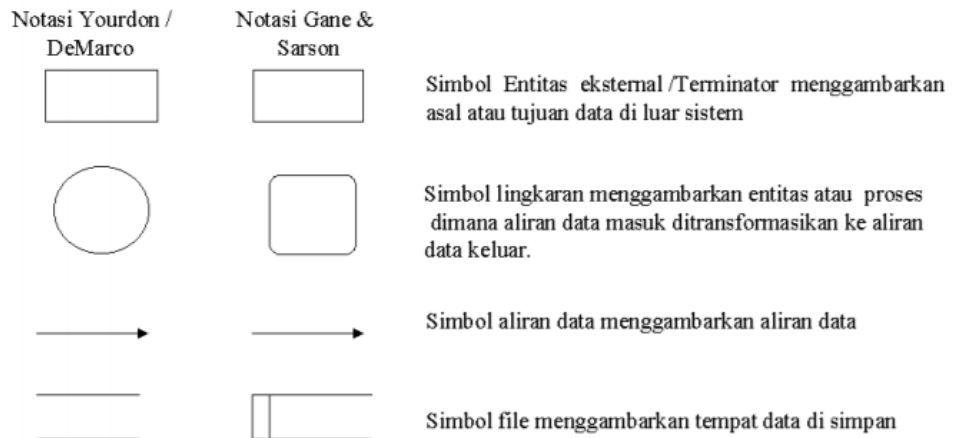
Pengaturan dari record secara logika didalam file dihubungkan satu dengan yang lainnya.

- 1) File Urut (Sequential File)
Merupakan file dengan organisasi urut dengan pengaksesan urut pula
- 2) File Urut Berindex (Indexed Sequential File)
Merupakan file dengan organisasi urut dengan pengaksesan langsung
- 3) File Akses Langsung (Direct Acces File)
Merupakan file dengan organisasi acak dengan pengaksesan langsung
- 4) Alat Perancangan Database
 - a. ERD
 - b. Mapping
- 5) Normalisasi Langkah-langkah Perancangan Database secara umum :

- a. Menentukan kebutuhan file database untuk sistem baru
 - b. Menentukan parameter dari file database
- 6) Perancangan Kontrol
- Tujuannya agar keberadaan sistem setelah diimplementasi dapat memiliki keandalan dalam mencegah kesalahan, kerusakan serta kegagalan proses sistem dan Ancaman Sistem
- a. Kesalahan manusia (lalai, kurang pelatihan)
 - b. Perangkat lunak yang bersifat merusak / menipu (Salami Technique, Trojan Horse, Logic Bomb, Worm, Virus)
 - c. Penyadapan
 - d. Pengaksesan yang tidak sah
 - e. Perubahan / kehilangan database
 - f. Kegagalan landasan teknologi Jenis Kontrol
 - g. Pencegahan
 - h. Pendeteksian
 - i. Pengkoreksian

2.2.14 Perancangan Sistem Secara Umum

Data Flow Diagram (DFD) adalah representasi grafik dari sebuah sistem. DFD menggambarkan komponen-komponen sebuah sistem, aliran-aliran data di mana komponen-komponen tersebut, dan asal, tujuan, dan penyimpanan dari data tersebut. Kita dapat menggunakan DFD untuk dua hal utama, yaitu untuk membuat dokumentasi dari sistem informasi yang ada, atau untuk menyusun dokumentasi untuk sistem informasi yang baru.



Gambar : 2.1 DFD Grafik

Empat simbol yang digunakan ,Ada 3 (tiga) jenis DFD, yaitu ;

- 1) Context Diagram (CD)
- 2) DFD Fisik
- 3) DFD Logis

Diagram Konteks Data Flow Diagram (DFD) yaitu alat bantu yang dapat menggambarkan sistem secara lengkap dan jelas, baik sistem yang sudah ada maupun sistem yang masih dalam rancangan. Dalam DFD dijelaskan mengenai aliran data, informasi proses, basis data dan sumber tujuan data yang dilakukan oleh sistem. Tingkatan atau level DFD dimulai dari diagram konteks yang menjelaskan dan menggambarkan sistem secara umum, terdiri dari beberapa elemen-elemen di luar sistem yang memberikan input ke dalam sistem. Diagram konteks tersebut akan dirinci ke dalam beberapa proses yang ada dalam sistem sehingga menghasilkan uraian sistem dalam level n yang lebih rinci.

Alat Bantu Analisis Analisis data masukan adalah suatu analisis yang dilakukan terhadap data-data dari entitas luar yang dimasukkan kedalam sistem. Dengan tujuan untuk mendapatkan pemahaman sistem secara keseluruhan,

tentang sistem yang berjalan sekarang sehingga permasalahan dapat dipecahkan dan kebutuhan pemakai sistem dapat diidentifikasi dengan benar. Untuk mendapatkan informasi tentang sistem yang berjalan sekarang, penulis banyak melakukan komunikasi dengan pemakai sistem serta pihak lainnya yang berhubungan dengan permasalahan yang sedang dianalisa. Pada tahapan analisis ini, penulis menggunakan beberapa alat bantu untuk dapat menggambarkan sistem secara keseluruhan. Alat bantu yang penulis gunakan adalah :

Flowmap, Diagram Konteks yang dilanjutkan dengan Data Flow Diagram (DFD) beserta diagram rincinya. Informasi yang disajikan dengan penggambaran flowmap ini lebih menekankan pada urutan aktivitas disetiap entitas yang berada dalam sistem. Sedangkan Diagram Konteks menggambarkan aliran data yang mengalir dari setiap entitas ke sistem, dan Data Flow Diagram merupakan penjelasan atau pemecahan dari Diagram Konteks yang menggambarkan aliran data, spesifikasi proses serta penyimpanan data hasil proses. Selain penggambaran dengan alat bantu diatas, penulis juga memberikan pemaparan sebagai penjelasan tambahan agar data yang disajikan dari hasil analisis mudah untuk dipahami. Analisis ini juga meliputi analisis pengkodean, analisis perangkat pengolahan data (Hardware & Software), analisis struktur Organisasi beserta tinjauan proses struktur sistem pengolahan data nilai STMIK DCI, dan analisis kebutuhan informasi. Pada bagian akhir penulis juga menyertakan kesimpulan hasil analisis sebagai tinjauan secara keseluruhan terhadap hasil analisis.

2.2.15 Konversi Sistem dalam Teknologi Informasi dan Fenomenanya

Sering kali organisasi melakukan kesalahan dalam melakukan pengalihan dari suatu sistem lama ke sistem baru (konversi sistem) hal ini tentunya dapat berakibat fatal bagi organisasi. Fenomena kesalahan dalam konversi sistem

informasi dapat terjadi apabila tidak dilakukan langkah-langkah awal dengan tepat sebelum dilakukan konversi. Adapun hal yang perlu dilakukan sebelum proses konversi yaitu

- 1) Proses perencanaan dan permodelan, meliputi analisa kebutuhan dan design,
- 2) konstruksi, meliputi penyusunan kode dan pengujian
- 3) Pemrograman dan pengetesan perangkat lunak (software), meliputi kegiatan :
 - a. Developmental (error testing per modul oleh programmer),
 - b. Alpha testing (error testing ketika sistem digabungkan dengan interface user oleh software tester), dan Beta testing (testing dengan lingkungan dan data sebenarnya).

Dengan memperhatikan hal-hal tersebut sebelum konversi dilakukan akan mampu meminimalisir kesalahan dalam melakukan konversi sistem. Pengalihan Sistem Informasi dari sistem yang lama ke sistem yang baru dapat berakibat fatal, terjadi karena :

1. Belum siapnya sumber daya untuk mengaplikasikan system yang baru.
2. System baru sudah terpasang, namun terdapat kesalahan prosedur dalam pelaksanaannya, sehingga perubahan tidak dapat terjadi. Sehingga keberadaan system baru justru mempersulit kinerja yang sudah ada.
3. Perencanaan dan aplikasi sistem Informasi tidak memiliki arah dan tahapan yang baik.
4. Tidak ada komunikasi yang baik diantara vendor sebagai penyedia IT dengan perusahaan sebagai pengguna, sehingga system baru yang terbentuk menjadi tidak sesuai dengan kebutuhan pengguna.
5. Perusahaan memandang perubahan teknologi merupakan hal yang harus dilakukan agar perusahaan tidak ketinggalan zaman. Namun sebenarnya perusahaan tidak membutuhkan teknologi tersebut.
6. Level kematangan perusahaan terhadap TI masih rendah.

7. Fenomena ini terjadi karena dengan adanya perubahan dari sistem lama ke sistem baru maka akan terjadi keadaan dimana karyawan menghadapi masa transisi yaitu keharusan menjalani adaptasi yang dapat berupa adaptasi teknis (skill, kompetensi, proses kerja), kultural (perilaku, mind set, komitmen) dan politikal (munculnya isu efisiensi karyawan/PHK, sponsorship/dukungan top management). Dengan adanya ketiga hal ini maka terjadi saling tuding di dalam organisasi, dimana manajemen puncak menyalahkan bawahan yang bertanggung jawab, konsultan, vendor bahkan terkadang peranti TI itu sendiri.

Langkah-langkah yang dilakukan agar kesalahan alih sistem informasi dapat dihindari:

1. Lihat kembali dan koreksi visi yang ingin dibangun, pelajari implementasi apa yang belum maksimal dan latih sumber daya manusia agar mampu mengoptimalkan peranti yang sudah dibeli. Hal ini hanya akan mungkin untuk dilaksanakan apabila pimpinan perusahaan mengetahui tentang TI/sedikit tentang TI, sehingga dia paham apa yang ingin dicapai perusahaannya dengan mengaplikasikan TI ini.
2. Harus menciptakan sinergisme diantara subsistem-subsistem yang mendukung pengoperasian sistem sehingga akan terjadi kerjasama secara terintegrasi diantara subsistem-subsistem ini. Asumsi hanya akan tercapai apabila para perancang sistem ini mengetahui masalah-masalah informasi apa yang ada di perusahaan dan yang harus segera di selesaikan. Biasanya para perancang sistem ini akan mulai pada tingkat perusahaan, selanjutnya turun ke tingkat-tingkat sistem.
3. Para perancang Sistem Informasi harus menyadari bagaimana rasa takut di pihak pegawai maupun manajer dapat mempengaruhi keberhasilan atau kegagalan proyek pengembangan dan sistem operasional. Manajemen perusahaan, dibantu oleh spesialis informasi, dapat mengurangi ketakutan ini dan dampaknya yang

merugikan dengan mengambil empat langkah berikut Menggunakan komputer sebagai suatu cara mencapai peningkatan pekerjaan (job enhancement) dengan memberikan pada komputer tugas yang berulang dan membosankan, serta memberikan pada pegawai tugas yang menantang kemampuan mereka. Menggunakan komunikasi awal untuk membuat pegawai terus menyadari maksud perusahaan. Pengumuman oleh pihak manajemen puncak pada awal tahap analisis dan penerapan dari siklus hidup sistem merupakan contoh strategi ini. Membangun hubungan kepercayaan antara pegawai, spesialisasi informasi dan manajemen. Hubungan tersebut tercapai dengan sikap jujur mengenai dampak-dampak dari sistem komputer dan dengan berpegang pada janji. Komunikasi formal dan penyertaan pemakai pada tim proyek mengarah pada tercapainya kepercayaan. Menyelaraskan kebutuhan pegawai dengan tujuan perusahaan. Pertama, identifikasi kebutuhan pegawai, kemudian memotivasi pegawai dengan menunjukkan pada mereka bahwa bekerja menuju tujuan perusahaan juga membantu mereka memenuhi kebutuhan mereka. Terdapat empat metode konversi sistem, yaitu :

1. Konversi Langsung (Direct Conversion)

Konversi ini dilakukan dengan cara menghentikan sistem lama dan menggantikannya dengan sistem baru.. Konversi langsung adalah pengimplementasian sistem baru dan pemutusan jembatan sistem lama, yang kadang-kadang disebut pendekatan cold turkey. Apabila konversi telah dilakukan, maka tak ada cara untuk balik ke sistem lama. Asumsi dari penggunaan sistem ini diantaranya :

1. Data sistem yang lama bisa digantikan sistem yang baru
2. Sistem yang lama sepenuhnya tidak bernilai.

3. Sistem yang baru bersifat kecil atau sederhana atau keduanya.

4. Rancangan sistem baru sangat berbeda dari sistem lama, dan perbandingan antara sistem – sistem tersebut tidak berarti. Kelebihan dari sistem konversi langsung ini yaitu sistem ini relatif murah, namun memiliki resiko kegagalan yang cukup besar.

2. Konversi Paralel (Parallel Conversion)

Konversi Paralel adalah suatu pendekatan dimana baik sistem lama dan baru beroperasi secara serentak untuk beberapa période waktu. Pada konversi ini, sistem baru dan sistem lama sama-sama dijalankan. Setelah melalui masa tertentu, jika sistem baru telah bisa diterima untuk menggantikan sistem lama, maka sistem lama segera dihentikan. Kelebihan dari sistem ini yaitu merupakan pendekatan yang paling aman sedangkan kelemahan dari sistem ini adalah merupakan cara yang paling mahal, karena pemakai harus menjalankan dua sistem sekaligus. Besarnya biaya dikeluarkan untuk penduplikasian fasilitas-fasilitas dan biaya personel yang memelihara sistem rangkap tersebut.

3. Konversi Bertahap (Phase-In Conversion)

Konversi dilakukan dengan menggantikan suatu bagian dari sistem lama dengan sistem baru. Jika terjadi sesuatu, bagian yang baru tersebut akan diganti kembali dengan yang lama. Jika tak terjadi masalah, modul-modul baru akan dipasangkan lagi untuk mengganti modul-modul lama yang lain. Dengan pendekatan seperti ini, akhirnya semua sistem lama akan tergantikan oleh sistem baru. Kelebihan dari sistem konversi ini yaitu kecepatan perubahan dalam organisasi tertentu bisa diminimasi, dan sumber-sumber pemrosesan data dapat diperoleh sedikit demi sedikit selama période waktu yang luas. Sedangkan kelemahannya yaitu keperluan biaya yang harus

diadakan untuk mengembangkan interface temporer dengan sistem lama, daya terapnya terbatas, dan terjadi kemunduran semangat di organisasi, sebab orang-orang tidak pernah merasa menyelesaikan sistem. Sistem konversi ini dianggap lebih aman daripada konversi langsung. Dengan metode Konversi Phase-in, sistem baru diimplementasikan beberapa kali, yang secara sedikit demi sedikit mengganti yang lama. Ia menghindarkan dari risiko yang ditimbulkan oleh konversi langsung dan memberikan waktu yang banyak kepada pemakai untuk mengasimilasi perubahan. Untuk menggunakan metode phase-in, sistem harus disegmentasi.

4. Konversi Pilot (Pilot Conversion)

Pendekatan ini dilakukan dengan cara menerapkan sistem baru hanya pada lokasi tertentu yang diperlakukan sebagai pelopor. Jika konversi ini dianggap berhasil, maka akan diperluas ke tempat-tempat yang lain. Ini merupakan pendekatan dengan biaya dan risiko yang rendah. Dengan metode Konversi Pilot, hanya sebagian dari organisasilah yang mencoba mengembangkan sistem baru. Kalau metode phase-in mensegmentasi sistem, sedangkan metode pilot mensegmentasi organisasi.

Cara untuk mengkonversi file adalah dengan cara :

1. Konversi File Gradual

Konversi file gradual adalah konversi yang dilakukan dengan beberapa tahapan. Biasanya digunakan dengan metode paralel dan phase-in. Cara kerja metode ini adalah :

- a. Suatu transaksi diterima dan dimasukkan ke dalam sistem.
- b. Program mencari file master baru.
- c. Jika record tersebut tidak ditemukan dalam file master baru, file master lama diakses untuk record yang tepat, dan

ditambahkan ke file master baru dan di update.

- d. Jika transaksi tersebut adalah record baru maka record baru disiapkan dan ditambahkan ke file master baru.

2. Konversi File Total

Konversi file total adalah konversi file dengan cara memformat sistem lama ke sistem baru dengan suatu software tertentu. Rancangan file baru hampir selalu mempunyai field-field record tambahan, struktur pengkodean baru, dan cara baru perelasian item-item data (misalnya, filefile relasional). Dengan menggunakan klasifikasi file perlu diperhatikan jenis prosedur kendali yang digunakan selama konversi. Rencana Implementasi Implementasi sistem lama ke sistem baru akan melibatkan :

- a. Tim implementasi
Tim implementasi terdiri dari :
 1. Profesional sistem yang merancang sistem
 2. Para manajer dan beberapa staff
 3. Perwakilan Vendor
 4. Pemakai Primer
 5. Pengcode
 6. Teknisi
- b. Keperluan implementasi sistem
 1. Persiapan tempat
 2. Pelatihan personil
 3. Persiapan/pembuatan dokumentasi
 4. Konversi file dan sistem
 5. Peninjauan Pasca Implementasi

Hal-hal yang dapat dilakukan untuk menghindari kegagalan penerapan sistem yaitu :

1. Adanya keteraturan dan saling terkait agar terjadi kerjasama yang baik.

2. Menerapkan sistem dengan memperhatikan semua aspek yang terlibat di dalamnya
3. Kembali ke tujuan awal dari perusahaan dan sistem harus mampu memenuhi kebutuhan-kebutuhan yang telah ditetapkan untuk dikembangkan
4. Memperhatikan sumber daya manusia dengan cara melakukan sosialisasi dan pelatihan. Perkembangan teknologi yang begitu cepat dan persaingan yang semakin kompetitif menyebabkan perusahaan harus melakukan perubahan pada sistem informasinya. Hal ini dilakukan agar perusahaan tidak ketinggalan dengan pesaing lainnya. Akan tetapi kadang kala terjadi kesalahan besar yang dapat berakibat fatal pada suatu organisasi pada saat pengalihan sistem informasi organisasi tersebut dari suatu sistem lama ke sistem yang baru. Fenomena kesalahan dalam konversi sistem informasi dapat terjadi apabila tidak dilakukan langkah-langkah awal dengan tepat sebelum dilakukan konversi. Adapun hal yang perlu dilakukan sebelum proses konversi yaitu

- 1) Proses perencanaan dan permodelan, meliputi analisa kebutuhan dan design,
- 2) konstruksi, meliputi penyusunan kode dan pengujian
- 3) Pemrograman dan pengetesan perangkat lunak (software), meliputi kegiatan : Developmental (error testing per modul oleh programmer), Alpha testing (error testing ketika sistem digabungkan dengan interface user oleh software tester), dan Beta testing (testing dengan lingkungan dan data sebenarnya)

Langkah-langkah yang dilakukan agar kesalahan alih sistem informasi dapat dihindari :

- a) Lihat kembali dan koreksi visi yang ingin di bangun, pelajari implementasi apa yang belum maksimal dan latih sumber daya manusia agar mampu mengoptimalkan peranti yang sudah dibeli. Hal ini hanya akan mungkin untuk dilaksanakan apabila pimpinan perusahaan mengetahui tentang

TI/sedikit tentang TI, sehingga dia paham apa yang ingin dicapai perusahaannya dengan mengaplikasikan TI ini.

- b) Harus menciptakan sinergisme diantara subsistem-subsistem yang mendukung pengoperasian sistem sehingga akan terjadi kerjasama secara terintegrasi diantara subsistem-subsistem ini. Asumsi hanya akan tercapai apabila para perancang sistem ini mengetahui masalah-masalah informasi apa yang ada di perusahaan dan yang harus segera di selesaikan. Biasanya para perancang sistem ini akan mulai pada tingkat perusahaan, selanjutnya turun ke tingkat-tingkat sistem. Fenomena penyebab kegagalan ini dapat berasal dari 3 (tiga) stakeholder utamanya dari organisasi/perusahaan tersebut, yaitu: management yang mewakili pihak perusahaan, vendors sebagai pihak ketiga yang membantu implementasi sistem baru tersebut, dan user sebagai pihak yang menggunakan sistem tersebut.

Management adalah salah satu penyebab dari kesalahan peralihan sistem lama ke sistem baru, hal tersebut dapat terjadi oleh beberapa faktor, antara lain :

1. Kurangnya dukungan dan komitmen dari pimpinan puncak dan manajemen perusahaan, sehingga inisiatif sistem baru yang digulirkan berjalan dengan tersendat-sendat, Buruknya perencanaan yang disusun oleh pihak manajemen sehingga ketika ingin dieksekusi mengalami banyak hambatan dan kesulitan.
2. Ketidakinginan manajemen dalam “merubah paradigma” berpikir maupun bekerja lebih senang kondisi status quo sehingga berbagai prasyarat utama untuk menjalankan atau mengimplementasikan sistem baru tersebut tidak tercapai

3. Ekspektasi yang terlampau berlebihan dari pihak manajemen terhadap sistem baru yang ingin diterapkan tanpa peduli dengan isu-isu terkait dengan cara atau pendekatan atau strategi menerapkan sistem tersebut secara efektif,
4. Pendefinisian kebutuhan yang kabur, sehingga ruang lingkup sistem baru yang ingin diterapkan menjadi tidak jelas yang tentu saja mempertinggi resiko kegagalan dalam implementasinya.
5. Sosialisasi mengenai sistem baru yang buruk kepada segenap karyawan perusahaan sehingga banyak pihak yang menolak dibandingkan dengan yang mendukung.

Pihak berikutnya yang dapat menyebabkan terjadinya kegagalan peralihan sistem informasi lama ke sistem informasi baru adalah pihak ketiga atau vendor. Faktor faktor kegagalan yang disebabkan oleh vendor :

1. Kurangnya pengalaman dari vendor maupun orang yang ditugaskan untuk mengimplementasikan sistem baru tersebut terutama untuk ruang lingkup penugasan serupa di industri yang sejenis.
2. Tidak mampu memberikan pemahaman dan penjelasan yang baik dan benar mengenai paradigma yang dipergunakan dalam sistem baru kepada mereka yang berkepentingan sehingga seringkali terjadi kekeliruan dalam cara memandangnya.
3. Pemilihan aplikasi yang keliru, atau tidak sesuai dengan situasi dan kondisi perusahaan yang membutuhkannya.
4. Salah dalam usaha membantu manajemen dalam mendefinisikan kebutuhannya sehingga ketika sistem baru tersebut diterapkan, tidak memberikan hasil sebagaimana yang diharapkan oleh para stakeholder terkait.
5. Tidak memberikan pelatihan yang memadai dan efektif kepada segenap stakeholder sehingga mereka tidak dapat menggunakan dan memanfaatkannya secara baik. Pihak terakhir yang memiliki andil besar dalam penyebab kegagalan sistem informasi lama ke sistem informasi baru adalah user sebagai pihak yang menggunakan sistem tersebut. Faktor-faktor yang dapat ditimbulkan oleh user adalah :

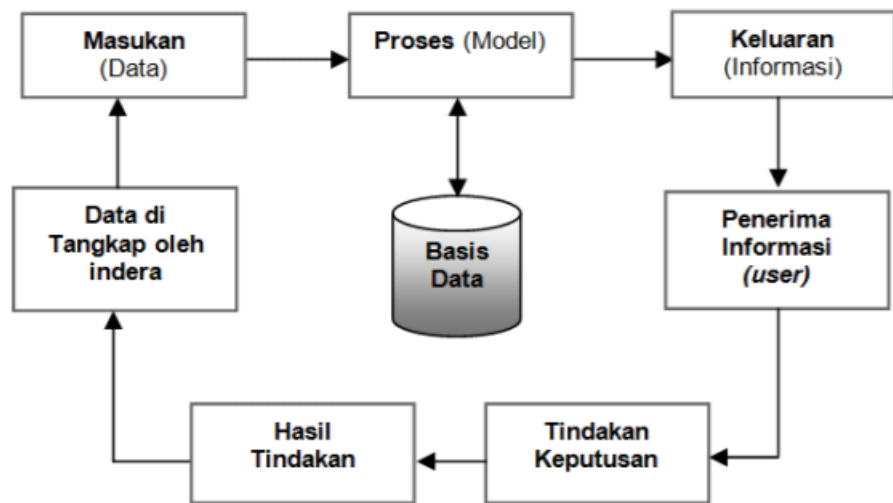
- a. Ketidakinginan para user untuk merubah cara kerja dalam beraktivitas sehari-hari sehingga selalu menentang segala bentuk aplikasi sistem baru tersebut, yang pada dasarnya membutuhkan keinginan dan kemampuan untuk bekerja dengan cara yang lebih efektif dan efisien.
- b. Kurangnya porsi pelatihan bagi para user agar yang bersangkutan memiliki kompetensi dan keahlian yang memadai untuk menjalankan sistem baru tersebut.
- c. Harapan yang berlebihan dan cenderung keliru terhadap sistem yang baru yang biasanya para user menganggap bahwa teknologi informasi dan software dapat menyelesaikan segala masalah dan kesulitan yang ada. Perubahan SI lama ke SI baru dapat mengakibatkan kesalahan yang beresiko dan berakibat fatal bagi jalannya suatu organisasi apabila tidak tepat dalam pelaksanaan SI barunya. Dalam memperkecil resiko yang ada, maka perlu kiranya diperhatikan berbagai cara dalam mengkonversi sistem dan langkah-langkah yang harus dilakukan sebelum pengalihan sistem informasi. Menurut O'Brien (2005) operasi awal dari sistem bisnis yang baru dapat menjadi tugas yang sulit. Hal ini biasanya memerlukan proses konversi dari penggunaan sistem yang ada saat ini (sistem lama) ke operasi aplikasi yang baru atau yang lebih baik.

Data, merupakan sesuatu yang menyangkut barang, kejadian, aktivitas, dan transaksi yang telah tercatat, diklasifikasikan, dan disimpan namun belum memiliki makna. Data dapat berupa nilai numerik, alphanumerik, gambar, dan suara.

Informasi, adalah data yang telah dikelola dalam bentuk tertentu untuk memberikan makna atau arti bagi penerimanya.

2.2.17 Siklus Informasi

1. Data dan informasi akan saling berkesinambungan sehingga membentuk suatu siklus yang disebut information cycle (siklus informasi).
2. Data ditangkap oleh indera kemudian menjadi inputan dalam sebuah model untuk diubah menjadi informasi bagi penerimanya yang nantinya akan membantu pengambilan keputusan dan menjadi sebuah hasil tindakan.



Gambar 2.2 : Siklus informasi

2.2.18 Pentingnya data dan informasi

1. Data dan informasi sebagai sebuah aset penting perusahaan/organisasi.
2. Informasi yang benar dapat menjadikan suatu perusahaan/organisasi memperoleh margin untuk melakukan aksi.
3. Data dan informasi sebagai salah satu parameter kemajuan perusahaan/organisasi (maturity level).

2.2.19 Sistem Basis Data dan Sistem File Pada sebuah institusi

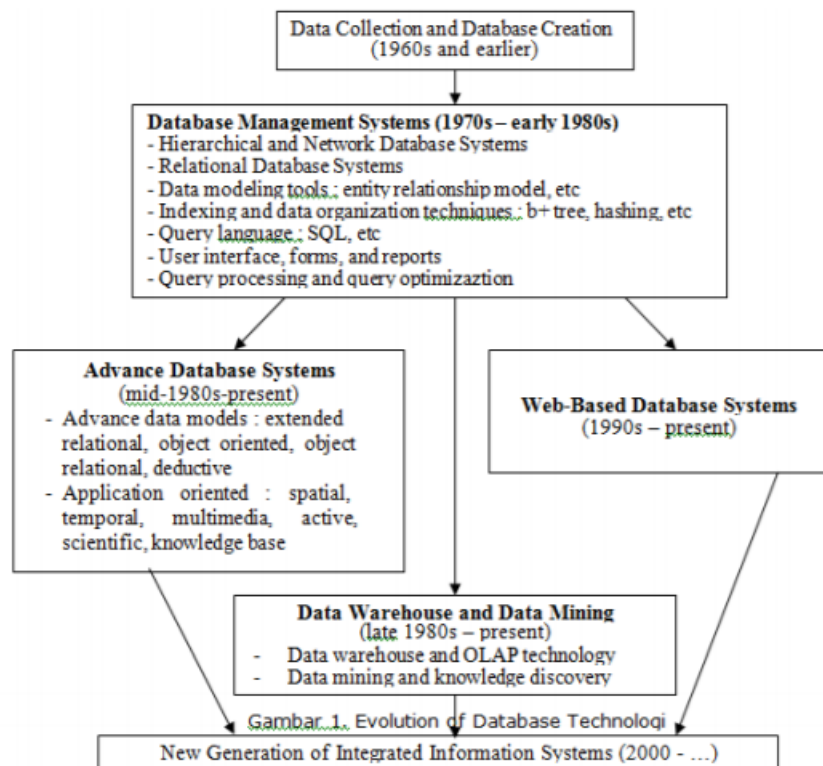
data merupakan salah satu hal yang sangat penting. Setiap bagian/divisi dari institusi memiliki data sendiri-sendiri. Tapi setiap bagian pun membutuhkan sebagian data dari bagian yang lain. Hal ini yang biasa dikenal sebagai “shared data”. Setiap divisi memiliki aplikasi sendiri-sendiri dalam melakukan manipulasi dan pengambilan data tersebut. Setiap aplikasi memiliki file-file dalam sistem operasi yang digunakan untuk menyimpan data-data. Seiring dengan berkembangnya institusi, bertambahnya bagian/divisi, bertambah pula data dan aplikasi yang digunakan. Bertambahnya aplikasi, bertambah pula filefile yang dibuat. Gaya sistem pemrosesan-file tersebut menyebabkan setiap data disimpan dalam bentuk record dalam berbagai macam file, dan diperlukan aplikasi yang berbeda dalam melakukan pengambilan record dari, dan penambahan record ke dalam file. Hal ini berlaku pada masa sebelum adanya Sistem Basis Data (DBMS). Menyimpan data dalam bentuk file yang berbeda-beda, memiliki kekurangan-kekurangan:

1. Data redundancy dan inconsistency. Dikarenakan programer yang berbeda membuat file dan aplikasi masing-masing, menyebabkan beragam format dan aplikasi yang dibuat. Bahkan, aplikasi pun dibuat menggunakan bahasa pemrograman yang berbeda-beda. Lebih jauh lagi, data atau informasi yang sama bisa terdapat dalam beberapa file yang berbeda. Ini yang disebut dengan redundancy. Redundancy data ini lama kelamaan akan menyebabkan inconsistency dari data.
2. Kesulitan dalam pengaksesan data. Dikarenakan setiap aplikasi memiliki file tersendiri untuk penyimpanan dan pengambilan data, maka jika suatu bagian dari institusi membutuhkan data dari bagian lain, akan menemui kesulitan. Hal ini dikarenakan aplikasi yang dimiliki bagian tersebut, tidak dapat membaca file yang terdapat di bagian lain.
3. Isolasi data. Dikarenakan data tersebar dalam berbagai macam file, dan file tersebut dalam beragam format, pembuatan aplikasi baru akan

terasa sulit ketika harus membaca format dari masing-masing file tersebut.

4. Masalah integritas. Data yang disimpan harus memenuhi hal yang dinamakan dengan consistency constraint. Jika sebuah constraint berubah, maka seluruh aplikasi yang digunakan harus mengakomodasinya. Masalah akan muncul, jika constraint melibatkan beberapa data dari file yang berbeda-beda.

5. Masalah keamanan. Tidak semua pengguna dari basis data dapat mengakses semua data. Hal ini akan sulit dilakukan jika menggunakan gaya penyimpanan data dalam file.



Gambar 2.3 : Perkembangan Database

2.3 Definisi Basis Data dan Sistem Basis Data (DBMS)

Basis data adalah penyimpanan kumpulan informasi secara sistematis dalam sebuah komputer sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh informasi dari basis data tersebut. Perangkat Lunak yang digunakan untuk mengelola dan memanggil query basis data disebut Sistem Manajemen Basis Data (Database Management System, DBMS). DBMS memiliki karakteristik sebagai berikut:

1. Software program
2. Supplements operating sistem
3. Manages data
4. Queries data and generates reports
5. Data security

Sedangkan sistem adalah sebuah tatanan yang terdiri atas sejumlah komponen fungsional yang saling berhubungan dan secara bersama-sama bertujuan untuk memenuhi suatu proses pekerjaan. Sehingga bisa dikatakan bahwa sistem basis data adalah sistem yang terdiri atas kumpulan file-file yang saling berhubungan dan dikelola oleh program (DBMS) yang memungkinkan beberapa pemakai dan atau program lain yang memiliki otoritas untuk mengakses dan memanipulasi data tersebut. Kelebihan pemakaian DBMS adalah:

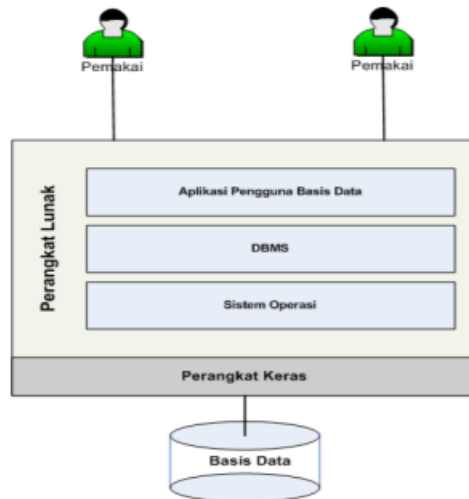
1. Data berdiri sendiri (Data Independence)
2. Pengaksesan data efisien (Efficient data access)
3. Integritas data dan keamanan terjamin (Data integrity and security)
4. Administrasi data (Data administration)
5. Dapat diakses bersamaan (Concurrent access)
6. Recovery saat terjadi kegagalan (Crash recovery)
7. Mengurangi waktu pembangunan aplikasi (Reduced application development time)

2.3.1 Komponen Sistem Basis Data

Komponen-komponen pada sebuah sistem basis data antara lain:

1. Perangkat keras
2. Sistem operasi

3. Basis data
4. DBMS (Database Management System)
5. Pemakai
6. Aplikasi lain



Gambar 2.3 : Komponen DBMS

2.3.2 Abstraksi Data

Tujuan utama dari sistem basis data adalah untuk menyediakan fasilitas untuk view data secara abstrak bagi penggunaannya. Namun bagaimana sistem menyimpan dan mengelola data tersebut, hanya diketahui oleh sistem itu sendiri. Abstraksi data merupakan level dalam bagaimana melihat data dalam sebuah sistem basis data. Berikut ini tiga level abstraksi data:

1. Level fisik

Merupakan level terendah pada abstraksi data yang menunjukkan bagaimana sesungguhnya data disimpan. Pada level ini pemakai melihat data sebagai gabungan dari struktur dan datanya sendiri.

2. Level logik

Merupakan level berikutnya pada abstraksi data, menggambarkan data apa yang disimpan pada basis data dan hubungan apa saja yang ada di antara data tersebut.

3. Level view

Merupakan level tertinggi dari abstraksi data yang hanya menunjukkan sebagian dari basis data. Banyak user dalam sistem basis data tidak akan terlibat dengan semua data atau informasi yang ada atau yang disimpan. Para user umumnya hanya membutuhkan sebagian data atau informasi dalam basis data yang kemunculannya di mata user diatur oleh aplikasi end user.

2.3.3 Model Basis Data

1. Hierarchical Memiliki struktur pohon dimana field hanya memiliki satu buah induk (parent), masing-masing parent memiliki banyak child (anak). Model ini memiliki kecepatan yang baik.

2. Network Relationship dibuat menggunakan linked list (pointer). Berbeda dengan model hierarchial satu anak dapat memiliki beberapa induk. Model ini memiliki fleksibilitas yang tinggi.

3. Relational Model ini direpresentasikan dalam tabel dua dimensi, tabel-tabel tersebut memiliki hubungan yang disebut dengan relasi. Model ini memiliki fleksibilitas dan kecepatan yang tinggi.

4. Object oriented Object Oriented Database adalah sebuah sistem database yang menggabungkan semua konsep object oriented seperti pewarisan, abstraksi, enkapsulasi, dll. Model ini dapat berinteraksi dengan baik dengan bahasa pemrograman berorientasi objek seperti java dan C++. 19. Perancangan Database Di dalam suatu organisasi yang besar, sistem database merupakan bagian penting pada sistem informasi, karena di perlukan untuk mengelola sumber informasi pada organisasi tersebut. Untuk mengelola sumber informasi tersebut yang pertama kali di

lakukan adalah merancang suatu sistem database agar informasi yang ada pada organisasi tersebut dapat digunakan secara maksimal.

Tujuan Perancangan Database

1. Untuk memenuhi kebutuhan akan informasi dari pengguna dan aplikasi
 2. Menyediakan struktur informasi yang natural dan mudah di mengerti oleh pengguna
 3. Mendukung kebutuhan pemrosesan dan beberapa obyek kinerja dari suatu sistem database
- Berikut ini siklus kehidupan sistem informasi di mana terdapat siklus kehidupan sistem database.

2.4 Siklus Kehidupan

2.4.1 Siklus Kehidupan Sistem Informasi

(Macro Life Cycle) Tahapan–tahapan yang ada pada siklus kehidupan sistem informasi yaitu :

- 1) Analisa Kelayakan Tahapan ini memfokuskan pada penganalisaan areal aplikasi yang unggul , mengidentifikasi pengumpulan informasi dan penyebarannya, mempelajari keuntungan dan kerugian , penentuan kompleksitas data dan proses, dan menentukan prioritas aplikasi yang akan digunakan.
- 2) Analisa dan Pengumpulan Kebutuhan Pengguna Kebutuhan–kebutuhan yang detail dikumpulkan dengan berinteraksi pada sekelompok pemakai atau pemakai individu. Mengidentifikasi masalah yang ada dan kebutuhan-butuhan, ketergantungan antar aplikasi, komunikasi dan prosedur laporan.
- 3) Perancangan Perancangan terbagi menjadi dua yaitu : perancangan sistem database dan sistem aplikasi
- 4) Implementasi Mengimplementasikan sistem informasi dengan database yang ada
- 5) Pengujian dan Validasi Pengujian dan validasi sistem database dengan kriteria kinerja yang diinginkan oleh pengguna.

6) Pengoperasian dan Perawatan Pengoperasian sistem setelah di validasi disertai dengan pengawasan dan perawatan sistem

2.4.2 Siklus Kehidupan Aplikasi Database

(Micro Life Cycle) Tahapan yang ada pada siklus kehidupan aplikasi database yaitu :

1. Pendefinisian Sistem Pendefinisian ruang lingkup dari sistem database, pengguna dan aplikasinya.
2. Perancangan Database Perancangan database secara logika dan fisik pada suatu sistem database sesuai dengan sistem manajemen database yang diinginkan.
3. Implementasi Database Pendefinisian database secara konseptual, eksternal dan internal, pembuatan file– file database yang kosong serta implementasi aplikasi software.
4. Pengambilan dan Konversi Data Database ditempatkan dengan baik, sehingga jika ingin memanggil data secara langsung ataupun merubah file–file yang ada dapat di tempatkan kembali sesuai dengan format sistem databasenya.
5. Konversi Aplikasi Software-software aplikasi dari sistem database sebelumnya di konversikan ke dalam sistem database yang baru
6. Pengujian dan Validasi Sistem yang baru telah di test dan di uji kinerja nya
7. Pengoperasian Pengoperasian database sistem dan aplikasinya
8. Pengawasan dan Pemeliharaan Pengawasan dan pemeliharaan sistem database dan aplikasi software
9. Proses Perancangan Database Ada 6 tahap untuk proses perancangan suatu database :
 1. Pengumpulan data dan analisis
 2. Perancangan database secara konseptual
 3. Pemilihan sistem manajemen database
 4. Perancangan database secara logika
 5. Perancangan database secara fisik
 6. Implementasi sistem database Keterangan :

Secara khusus proses perancangan berisikan 2 aktifitas paralel. Aktifitas yang pertama melibatkan perancangan dari isi data dan struktur database, sedangkan aktifitas kedua mengenai perancangan pemrosesan database dan aplikasi–aplikasi perangkat lunak. Dua aktifitas ini saling berkaitan , misalnya mengidentifikasi data item yang akan disimpan dalam database dengan cara menganalisa aplikasi–aplikasi database. Dua aktifitas ini juga saling mempengaruhi satu sama lain. Contohnya tahap perancangan database secara fisik, pada saat memilih struktur penyimpanan dan jalur akses dari file suatu database dimana bergantung dengan aplikasi–aplikasi yang akan menggunakan file tersebut. Penentuan perancangan aplikasi–aplikasi database yang mengarah ke konstruksi skema database telah ditentukan selama aktifitas pertama. Ke-enam tahap yang telah disebutkan sebelumnya dapat di proses secara tidak berurutan . Dalam beberapa hal, dapat dilakukan modifikasi perancangan kembali ke tahap yang pertama (feedback loop) setelah melakukan tahap selanjutnya.

Tahap 1 : Pengumpulan data dan analisis Sebelum merancang suatu database, yang harus dilakukan adalah mengetahui dan menganalisis apa yang diinginkan dari pengguna aplikasi, sehingga proses ini disebut pengumpulan data dan analisis. Untuk menspesifikasikan kebutuhan yang pertama kali dilakukan adalah mengidentifikasi bagian lain di dalam sistem informasi yang berinteraksi dengan sistem database. Termasuk pengguna yang baru atau yang sudah lama juga aplikasinya, kebutuhan–kebutuhan tersebut dikumpulkan dan di analisa.

Kegiatan pengumpulan data dan analisis :

1. Menentukan kelompok pemakai dan areal bidang aplikasinya. Pengguna yang menguasai aplikasi yang lama dari setiap bagian dipilih untuk menyampaikan kebutuhan-kebutuhan dan menspesifikasikannya.
2. Peninjauan dokumentasi yang ada. Dokumen yang berhubungan dengan aplikasi yang akan dibuat dipelajari dan dianalisa, sedangkan dokumen lainnya seperti kebijakan manual, form, laporan–

laporan dan bagan-bagan organisasi diuji dan ditinjau kembali untuk mengetahui apakah dokumen tersebut berpengaruh terhadap pengumpulan data dan proses spesifikasi

3. Analisa lingkungan operasi dan kebutuhan pemrosesan. Lingkungan operasional yang sekarang dan informasi yang direncanakan akan di gunakan dipelajari, termasuk menganalisa jenis-jenis dari transaksi dan frekuensi transaksinya seperti halnya alur informasi dengan sistem. Input dan output data untuk transaksi tersebut harus diperinci.

4. Pengumpulan respon terhadap daftar pertanyaan dan angket yang telah dibuat sebelumnya. Pengumpulan respon dari angket dan daftar pertanyaan berisikan prioritas para pengguna dan penempatan mereka di dalam berbagai aplikasi. Ketua kelompok mungkin akan ditanya untuk membantu para pengguna dalam memberikan informasi yang penting dan menentukan prioritas.

5. Teknik yang digunakan dalam penspesifikasian kebutuhan secara formal :

1. OOA (Object Oriented Analysis)

2. DFD (Data Flow Diagram)

3. HIPO (Hierarchical Input Process Output

)

4. SADT (Structured Analysis & Design)

Tahap 2 : Perancangan database secara konseptual Tujuan dari tahap ini adalah untuk menghasilkan skema konseptual untuk database yang tidak tergantung pada sistem manajemen database yang spesifik. Penggunaan model data tingkat tinggi seperti ER/EER sering digunakan didalam tahap ini. Di dalam skema konseptual dilakukan perincian aplikasi-aplikasi database dan transaksi-transaksi yang diketahui .

Ada dua kegiatan di dalam perancangan database secara konseptual :

1. Perancangan skema konseptual : Pada tahap ini kegiatan yang dilakukan mengecek tentang kebutuhan- kebutuhan

pemakai terhadap data yang dihasilkan dari tahap 1, dimana tujuan dari proses perancangan skema konseptual adalah menyatukan pemahaman dalam struktur database, pengertian semantik, keterhubungan dan batasanbatasannya, dengan membuat sebuah skema database konseptual dengan menggunakan model data ER/EER tanpa tergantung dengan sistem manajemen database. Ada dua pendekatan perancangan skema konseptual :

- a. Terpusat Kebutuhan–kebutuhan dari aplikasi atau kelompok–kelompok pemakai yang berbeda digabungkan menjadi satu set kebutuhan pemakai kemudian dirancang menjadi satu skema konseptual.
- b. Integrasi view–view yang ada Untuk masing–masing aplikasi atau kelompok–kelompok pemakai yang berbeda dirancang sebuah skema eksternal (view) kemudian view – view tersebut disatukan ke dalam sebuah skema konseptual. Ada 4 strategi dalam perancangan skema konseptual :
 1. Top down
 2. Bottom Up
 3. Inside Out
 4. Mixed
 5. Transaksi

Merancang karakteristik dari transaksi–transaksi yang akan di implementasikan tanpa tergantung dengan DBMS yang telah dipilih. Transaksi– transaksi ini digunakan untuk memanipulasi database sewaktu diimplementasikan . Pada tahap ini diidentifikasi input, output dan fungsional . Transaksi ini antara lain :

1. Retrieval
2. Update
3. Delete
4. select dll.

Tahap 3 : Pemilihan Sistem Manajemen Database
Pemilihan sistem manajemen database ditentukan oleh beberapa faktor :

1. Teknik, Ekonomi, dan Politik Organisasi Faktor Teknik :
 - a. Tipe model data (hirarki, jaringan atau relasional)
 - b. Struktur penyimpanan dan jalur pengaksesan yang didukung sistem manajemen database
 - c. Tipe interface dan programmer
 - d. Tipe bahasa queri Faktor Ekonomi :
 - 1) Biaya penyediaan hardware dan software
 - 2) Biaya konversi pembuatan database
 - 3) Biaya personalia
 - 4) Biaya pelatihan
 - 5) Biaya pengoperasian
 - 6) Biaya pemeliharaan Faktor Organisasi :
 - e. Struktur data Jika data yang disimpan dalam database mengikuti struktur hirarki, maka suatu jenis hirarki dari sistem manajemen database harus dipikirkan.
 - f. Personal yang terbiasa dengan sistem yang terdahulu Jika staff programmer dalam suatu organisasi sudah terbiasa dengan satu sistem manajemen database maka hal ini dapat mengurangi biaya latihan dan waktu belajar.
 - g. Ketersediaan dari service vendor Keberadaan fasilitas pelayanan penjual sangat dibutuhkan untuk membantu memecahkan masalah sistem.

Tahap 4 : Perancangan database secara logika (Transformasi model data) Transformasi dari skema konseptual dan eksternal (Tahap 2) ke model data sistem manajemen database yang terpilih, ada dua proses yaitu :

1. Transformasi yang tidak tergantung pada sistem, pada tahap ini transformasi tidak mempertimbangkan karakteristik yang spesifik atau hal– hal khusus yang akan diaplikasikan pada sistem manajemen database
2. Penyesuaian skema ke sistem manajemen database yang spesifik, di lakukan suatu penyesuaian skema yang dihasilkan dari tahap 1 untuk dikonfirmasi pada bentuk implementasi yang spesifik dari suatu model data seperti yang digunakan oleh sistem manajemen database yang

terpilih Hasil dari tahap ini dituliskan dengan perintah DDL ke dalam bahasa sistem manajemen database terpilih. Tapi jika perintah DDL tersebut termasuk dalam parameter-parameter perancangan fisik, maka perintah DDL yang lengkap harus menunggu sampai tahap perancangan database secara fisik telah lengkap.

Tahap 5 : Perancangan Database Secara Fisik Proses pemilihan struktur penyimpanan yang spesifik dan pengaksesan file-file database untuk mencapai kinerja yang terbaik di bermacam-macam aplikasi Kriteria pemilihan perancangan fisik :

1. Waktu respon Waktu transaksi database selama eksekusi untuk menerima respon
2. Penggunaan ruang penyimpanan Jumlah ruang penyimpanan yang digunakan oleh database file dan struktur jalur pengaksesannya
3. Terobosan yang dilakukan file transaksi (Transaction throughput) Merupakan nilai rata-rata transaksi yang dapat di proses permenit oleh sistem database dan merupakan parameter kritis dari sistem transaksi

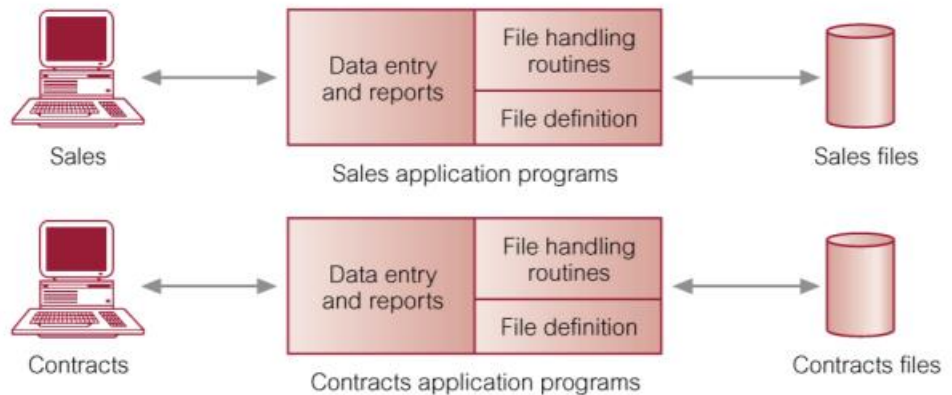
Apabila waktu respon dari database tidak mencapai optimalisasi, maka pada tahap perancangan fisik ini dapat dilakukan denormalisasi. Denormalisasi Denormalisasi merupakan proses yang dilakukan pada database yang sudah dinormalisasi, dengan cara memodifikasi struktur tabel dan mengabaikan kerangkapan data (yang terkontrol) untuk meningkatkan kinerja database. Proses denormalisasi termasuk :

1. Mengkombinasikan tabel-tabel yang terpisah dengan join
2. Mereplikasi/menduplikat data

Tahap 6 : Implementasi Implementasi skema database logik dan fisik ke dalam pernyataan DDL dan SDL dari sistem manajemen database yang telah dipilih, untuk digunakan dalam pembuatan file-file database yang masih kosong

1. Kumpulan dari program-program aplikasi yang menyediakan layanan untuk end users (Co : Laporan).

2. Setiap program mendefinisikan dan mengatur datanya masing-masing. File-Based Processing



Sales Files

PropertyForRent (propertyNo, street, city, postcode, type, rooms, rent, ownerNo)

PrivateOwner (ownerNo, fName, lName, address, telNo)

Client (clientNo, fName, lName, address, telNo, prefType, maxRent)

Contracts Files

Lease (leaseNo, propertyNo, clientNo, rent, paymentMethod, deposit, paid, rentStart, rentFinish, duration)

PropertyForRent (propertyNo, street, city, postcode, rent)

Client (clientNo, fName, lName, address, telNo)

Gambar 2.5 : Skema Database

Keterbatasan dari pendekatan berbasis file (File-Based)

1. Data terpisah-pisah dan terisolasi (Separation and isolation of data) Setiap program mengatur datanya masing-masing. User dari program yang satu tidak dapat menggunakan data yang diperlukan yang berada pada program yang lain.
2. Duplikasi Data (Duplication of data) Data yang sama terdapat pada program yang berbeda. Pemborosan ruang penyimpanan dan sangat potensial untuk terjadinya perbedaan format dan/atau nilai untuk data yang sama.
3. Ketergantungan Data (Data dependence) Struktur file didefinisikan dalam kode program.
4. Ketidaksesuaian Format File (Incompatible file formats) Program dituliskan dalam bahasa pemrograman yang

berbeda, sehingga tidak mudah untuk mengakses data/file yang satu dengan yang lainnya.

5. Program Aplikasi dengan Query/pengembangan yang terbatas (Fixed Queries/Proliferation of application programs) Program dibuat untuk memenuhi fungsi-fungsi khusus. Setiap terdapat kebutuhan baru yang timbul/diperlukan, akan memerlukan program yang baru pula.

2.4.3 Pendekatan Database

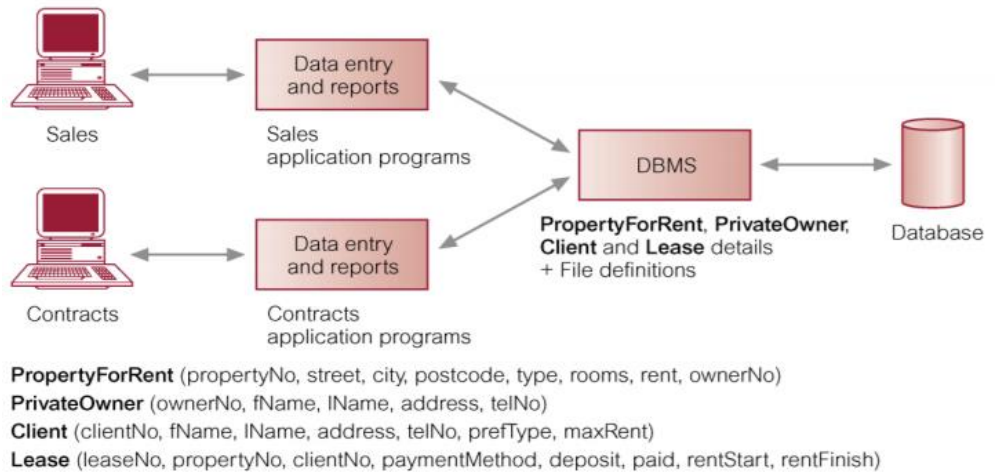
Pendekatan Database Timbul karena :

1. Pendefinisian data disimpan/dilekatkan dalam program aplikasi bukan disimpan pada tempat yang terpisah dan independen.
2. Tidak ada pengaturan/control pada pengaksesan dan manipulasi data yang dilakukan oleh program aplikasi Hasil :

- a. Database dan Database Management System (DBMS).
- b. Kumpulan relasi-relasi logikal dari data (dan deskripsi data) yang dapat digunakan bersama dan dibuat untuk memperoleh informasi yang dibutuhkan oleh perusahaan.
- c. Katalog sistem (metadata) menyediakan deskripsi data untuk memungkinkan kemandirian data-program (program–data independence).
- d. Merupakan relasi data logikal yang terdiri dari entity-entity, atribut-atribut, dan relationship dari informasi organisasi/perusahaan.

2.4.4 Database Management System (DBMS)

1. Sistem software yang memungkinkan user untuk mendefinisikan, membuat, dan memelihara database dan menyediakan akses terkontrol untuk database yang bersangkutan.



Gambar 2.6 : DBMS

Data definition language (DDL).

Memungkinkan spesifikasi tipe data, struktur dan batasan-batasan data. Semua spesifikasi disimpan dalam database. Data manipulation language (DML). Fasilitas pengadaaan umum (dengan query language) mengenai data. Pengontrolan akses terhadap database meliputi :

1. Sistem keamanan.
2. Sistem integritas.
3. Sistem kontrol konkurensi.
4. Sistem kontrol recovery.
5. Katalog yang dapat diakses oleh user.
6. Mekanisme view .
7. Hanya menyediakan data-data yang dibutuhkan atau digunakan oleh user.

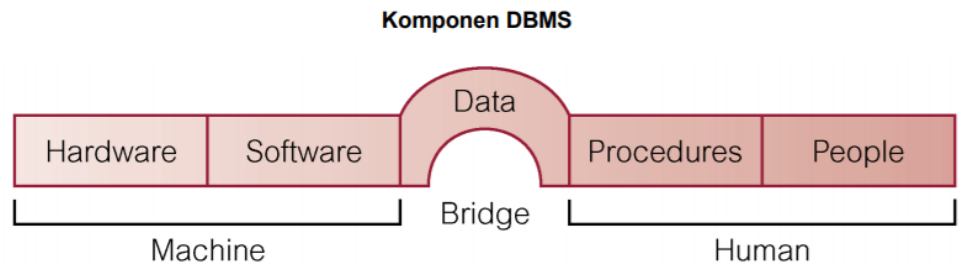
Fungsi DBMS Definisi Data (DDL) Untuk spesifikasi tipe data, struktur dan batasan-batasan lainnya (Skema eksternal, konseptual dan internal, juga mapping-mapping yang ada). Seluruh spesifikasi disimpan dalam database. DBMS harus menyertakan komponen DDL processor atau DDL Compiler. Manipulasi Data (DML) DBMS harus mampu menangani permintaan pemanggilan data, update atau penghapusan data, juga penambahan data baru

kedalam database. DBMS harus menyertakan komponen DML processor atau DML compiler. Optimasi dan Eksekusi DML Request akan diproses oleh komponen Optimizer yang berfungsi untuk menentukan cara yang paling efisien dalam implementasi request.

Requaest yang telah dioptimalkan kemudian dieksekusi dibawah kontrol Run-Time Manager. Keamanan dan Integritas Data DBMS harus mampu memonitor permintaan user, dan menolak segala usaha yang mengganggu batasan keamanan dan integritas data yang telah didefinisikan. Recovery dan Concurrency Data DBMS atau komponen software lain (Transaction Manager) harus menyediakan kontrol system recovery dan concurrency. Data Dictionary Berisikan “data about the data “, yaitu seluruh objek yang terkait dalam system didefinisikan dan disimpan dalam data dictionary. Performance DBMS harus menampilkan semua fungsi yang telah didefinisikan sebelumnya se-efisien mungkin. Mekanisme View. Menyediakan/menampilkan kepada pengguna data-data yang diperlukan dan digunakan saja. Views Memungkinkan setiap pengguna memiliki tampilan database tersendiri. Suatu view pada dasarnya merupakan bagian/subset dari database. Manfaat view meliputi:

- ♣ Mengurangi kerumitan (Reduce complexity)
- ♣ Menyediakan tingkatan keamanan (Provide a level of security)
- ♣ Menyediakan mekanisme untuk mengubah tampilan database
- ♣ Menampilkan struktur database yang konsisten dan tidak berubah walaupun database asal diubah

2.4.5 Komponen DBMS



Gambar 2.7 : Komponen DBMS

1. Hardware

- Meliputi PC sampai dengan jaringan komputer.
- Tempat penyimpanan secondary (manegtic disk),

I/O device

ex : disk drives), device Controller, I/O Channels, dan lainnya.

- Hardware processor dan main memory, digunakan untuk mendukung saat eksekusi system software database.

2. Software - DBMS, operating system, network software (jika diperlukan) dan program aplikasi pendukung lainnya.

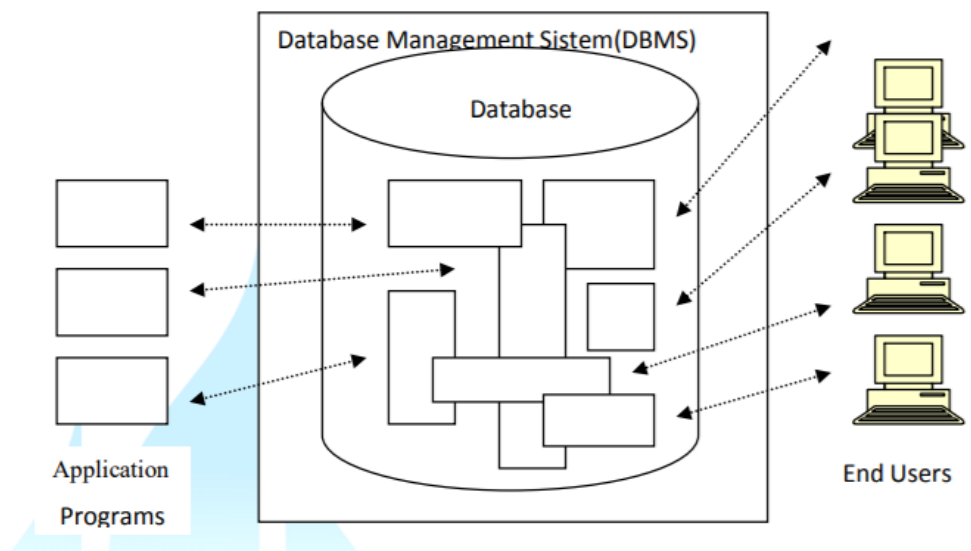
3. Data - Data pada sebuah system database baik itu single-user system maupun multiuser system harus terintegrasi dan dapat ddigunakan bersama (Integrated and Shared). - Digunakan oleh organisasi dan deskripsi dari data disebut schema.

4. Procedures - Instrukti dan aturan yang harus disertakan dalam mendesain dan menggunakan database dan DBMS.

People - DA (Data Administrator), seseorang yang berwenang untuk membuat keputusan stategis dan kebijakan mengenai data yang ada - DBA (DataBase Administrator), menyediakan dukungan teknis untuk implementasi keputusan tersebut, dan bertanggungjawab atas keseluruhan kontrol system pada level teknis - Database Designer (Logical and Physical) - Application

Programmers, bertanggungjawab untuk membuat aplikasi database dengan menggunakan bahasa pemrograman yang ada, seperti :

1. C++, Java, dan lainnya.
2. End Users, Siapapun yang berinteraksi dengan system secara online melalui workstation/terminal.



Gambar 2.8 : Database System

2.4.6 History of Database Systems

1. First generation - Hierarchical and Network
2. Second generation - Relational
3. Third generation - Object Relational - Object-Oriented
- 1.6Keuntungan dan kerugian DBMS Keuntungan DBMS
4. Penggunaan Data Bersama (The Data Can Be Shared)
5. Mengurangi Kerangkapan Data (Redudancy Can Be Reduced)
6. Menghindari Ketidakkonsistenan Data (Inconsistency Can Be Avoided)
7. Integritas Data Terpelihara (Integrity Can Be Maintained)

8. Keamanan Terjamin (Security Can Be Enforced)
9. Kebutuhan User Yang Kompleks Dapat Teratasi (Balanced conflicting requirements)
10. Pelaksanaan Standarisasi (Standards Can Be Enforced)
11. Meningkatkan Produktivitas (Increased productivity)
12. Layanan Back up dan Recovery Semakin Baik (Improved backup and recovery services) Kerugian DBMS
13. Rumit (Complexity)

Karena penetapan fungsi dari DBMS yang baik, menyebabkan DBMS menjadi software yang cukup rumit. Seluruh user harus mengetahui fungsi-fungsi yang ada dengan baik, sehingga dapat memperoleh manfaatnya.


1. Ukuran (Size) Kerumitan dan banyaknya fungsi yang ada menyebabkan DBMS memerlukan banyak software pendukung yang mengakibatkan penambahan tempat penyimpanan dan memory.
2. Biaya DBMS (Cost of DBMS)
3. Biaya Tambahan Hardware (Additional hardware costs)
4. Biaya Konversi (Cost of conversion)
5. Performance Pada dasarnya DBMS dibuat untuk menyediakan banyak aplikasi, akibatnya mungkin beberapa aplikasi akan berjalan tidak seperti biasanya.
6. Higher impact of a failure Karena system yang terpusat, jika seluruh user dan aplikasi terakses dari DBMS maka kerusakan pada bagian manapun dari system, akan menyebabkan operasi terhenti.


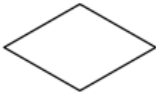
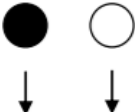
2.5 FLOWCHART

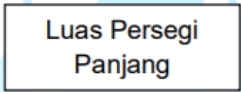
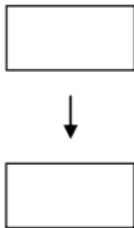
2.5.1 STRUCTUR CHART (BAGAN STRUKTUR)

Fungsi dari Structure Chart digunakan untuk mendefinisikan dan mengilustrasikan dari sistem secara berjenjang dalam bentuk modul dan sub modul. Structure Chart juga menunjukkan hubungan elemen data dan elemen kontrol serta hubungan antar modulnya, sehingga Structure Chart dapat memberikan penjelasan yang lengkap dari sistem dipandang dari elemen data, elemen kontrol, modul dan hubungan antar modulnya. **SIMBOL-SIMBOL DASAR** Dalam menggambarkan struktur organisasi sistem secara berjenjang digunakan beberapa macam simbol.

Simbol-simbol ini merupakan simbol-simbol standar yang paling banyak digunakan. Disamping simbol-simbol standar tersebut, pemrogram juga dapat menambah khasanah simbol yang akan digunakan dalam penggambaran tersebut, tetapi pemrogram juga harus memberi penjelasan tentang maksud dari simbol yang dibuat tersebut dalam bentuk kamus simbol. Namun demikian sebaiknya pemrogram menggunakan simbol-simbol standar untuk menggambarkan struktur sistem tersebut agar mudah dipahami oleh pemrogram lain.

| Gambar | Nama | Keterangan |
|---|--------|------------------------------------|
|  | Module | Simbol ini menunjukkan suatu modul |

| | | |
|--|------------|---|
| | | |
|  | Connection | Simbol ini digunakan untuk menghubungkan suatu modul dengan modul yang Lainnya |
| | Loop | Simbol ini menunjukkan suatu perulangan didalam modul |
|  | Decision | Simbol ini menunjukkan suatu penyeleksian kondisi didalam modul |
|  | Couple | Simbol ini menunjukkan suatu data/elemen yang dikirimkan dari satu modul ke modul lainnya. Anak panah dengan lingkaran yang kosong menunjukkan data yang dikirimkan, sedangkan anak panah dengan lingkaran padat menunjukkan elemen kontrol yang dikirimkan |

| Gambar | Keterangan |
|---|---|
|  | Menunjukkan suatu modul dengan nama "Luas Persegi Panjang" |
|  | Modul A memanggil Modul B. Setelah proses dari modul B selesai, maka proses kembali ke modul yang memanggil |





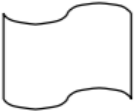
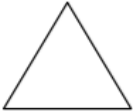
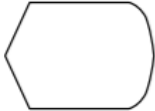
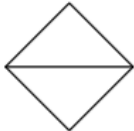


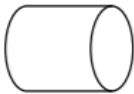
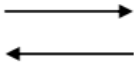
Gambar 2.8 : Structure Chart

Structure Chart memiliki dua model penggambaran sistem, yaitu Transform-Centered dan Transaction-Centered a. Transformed-Centered Structure Chart dengan model Transformed Centered menggambarkan sistem dalam cabang utama, yaitu :

1. Cabang Input yang merupakan cabang yang akan menerima input dan menentukan status input untuk siap di proses
2. Cabang Proses yang merupakan cabang yang akan melakukan fungsi utama dari sistem, yaitu memproses input yang dikirim dari cabang input
3. Cabang Output, merupakan cabang yang akan memformat data menjadi output

FLOWCHART Adalah Bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah. Flowchart merupakan cara penyajian dari suatu algoritma Tujuan Utama dari penggunaan Flowchart adalah


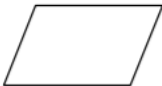
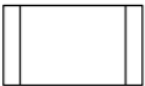
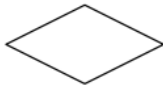


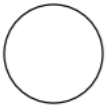

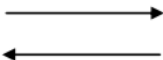
untuk menggambarkan suatu tahapan penyelesaian masalah secara sederhana, terurai, rapi dan jelas dengan menggunakan simbol-simbol yang standar. Tahap penyelesaian masalah yang disajikan harus jelas, sederhana, efektif dan tepat. Dalam penulisan Flowchart dikenal dua model, yaitu Sistem Flowchart dan Program Flowchart. SISTEM FLOWCHART Merupakan diagram alir yang menggambarkan suatu sistem peralatan komputer yang digunakan dalam proses pengolahan data serta hubungan antar peralatan tersebut. Sistem Flowchart ini tidak digunakan untuk menggambarkan urutan langkah untuk memecahkan masalah, tetapi hanya untuk menggambarkan prosedur dalam sistem yang dibentuk. Dalam menggambar Flowchart biasanya digunakan simbol-simbol standar, tetapi pemrograman juga dapat membuat simbol-simbol sendiri apabila simbol-simbol yang telah tersedia dirasa masih kurang. Dalam kasus ini pemrogram harus melengkapi gambar Flowchart tersebut dengan kamus simbol yang digunakannya, agar pemrogram lain dapat mengetahui maksud dari simbol-simbol tersebut. Berikut ini adalah gambar dari simbol-simbol standar yang telah banyak digunakan pada penggambaran Sistem Flowchart serta contoh penggunaannya.

| | | | |
|---|---|---|---|
|  |  |  |  |
| Pita Magnetik | Process | Kartu Plong/ Keyboard | Magnetic Disk |
|  |  |  |  |
| Punched Paper Tape | OffLine Storage | On Line Storage/ VDU | Process Sortir |
|  |  |  |  |
| Input / Output | Proses Merge | Magnetic Drum | Arus |

Gambar 2.9 : Program Flowchart

PROGRAM FLOWCHART Merupakan diagram alir yang menggambarkan urutan logika dari suatu prosedur pemecahan masalah. Dalam menggambarkan program Flowchart, telah tersedia simbol-simbol standar, tetapi seperti pada sistem flowchart, pemrogram dapat menambah khasanah simbol-simbol tersebut, tetapi pemrogram juga

harus melengkapi penggambaran program Flowchart dengan kamus simbol. Berikut ini adalah gambar dari simbol-simbol standar yang digunakan pada program Flowchart

| | | |
|---|---|---|
|  |  |  |
| Proses | Input Output | Keterangan |
|  |  |  |
| Pengujian | Pemberian Nilai Awal | Awal/Akhir Program |
|  |  |  |
| Konektor pada satu halaman | Konektor pada halaman lain | Arah |

Pada penggambaran program Flowchart terdapat dua jenis metode, yaitu Conceptual Flowchart dan Detail Flowchart. Conceptual Flowchart menggambarkan tentang alur dari suatu pemecahan masalah secara global saja, sedangkan Detail Flowchart menggambarkan alur pemecahan masalah secara rinci.

Flow Chart

Flowchart adalah metode untuk menggambarkan tahap-tahap penyelesaian masalah beserta aliran data menggunakan simbol-simbol yang mudah untuk dipahami. Flowchart merupakan hal penting untuk menyederhanakan rangkaian proses guna memudahkan pengguna dalam memahami informasi tersebut. Ada beberapa jenis dalam membuat sebuah flowchart, yaitu :

1. Flowchart dokumen Flowchart dokumen atau yang kerap disebut flowchart formuliran diagram alir yang menggambarkan proses dari laporan dan formulir lengkap dengan tembusan-tembusannya.
2. Flowchart sistem Flowchart sistem atau bagan alir sistem merupakan bagan yang menunjukkan proses pekerjaan di dalam sistem. Bagan ini akan menggambarkan arus pekerjaan secara mendetail dan menyeluruh. Jadi, Anda bisa melihat urutan-urutan proses/prosedur di dalam sistem.
3. Flowchart program Flowchart program merupakan bagan alir yang menggambarkan tahapan dalam proses sebuah program. Flowchart ini merupakan produk turunan dari flowchart sistem. Hal ini guna memudahkan analis sistem dan programmer dalam memahami suatu program.
4. Flowchart skematis Sekilas flowchart skematis tampak menyerupai flowchart sistem, karena memang kedua flowchart ini sama-sama digunakan untuk menggambarkan prosedur/proses dalam sistem. Simbol-simbol yang digunakan keduanya pun sama.
5. Fungsi flowchart ini untuk melihat prosedur dalam suatu proses produksi. Selain itu, flowchart proses juga banyak digunakan untuk melihat jarak dari langkah satu ke langkah yang lain, lengkap dengan time proses yang dibutuhkan. [10]

2.2.8 Flowmap

Flowmap adalah campuran peta dan flow chart, yang menunjukkan pergerakan benda dari satu lokasi ke lokasi lain, seperti jumlah orang dalam migrasi, jumlah barang yang diperdagangkan, atau jumlah paket dalam jaringan.

Flowmap menolong analisis dan programmer untuk memecahkan masalah ke dalam segmen-segmen yang lebih kecil dan menolong dalam menganalisis alternatif-alternatif lain dalam pengoprasian.[11]

2.2.9 MySQL

MySQL (My Structure Query Language) adalah perangkat lunak sistem manajemen basis data SQL (Database Management System) atau DBMS dari sekian banyak DBMS, seperti Oracle, MS SQL, Postgre SQL, dan lain-lain. MySQL merupakan DBMS yang multithread, multi-user yang bersifat gratis dibawah lisensi GNU General Public Licence (GPL) [12]

2.2.10 CSS

CSS adalah singkatan dari Casading Style Sheet yang merupakan kumpulan perintah yang dibentuk dari berbagai sumber yang disusun menurut urutan tertentu sehingga mampu mengatasi konflik style. CSS adalah salah satu bahasa pemrograman web yang mengatur komponen dalam suatu web supaya lebih terstruktur dan lebih seragam.[13]

2.2.11 HTML HTML (Hyper Text Markup Language) adalah suatu format data yang digunakan untuk membuat dokumen hypertext yang dapat dieksekusi dari satu platform computer ke platform computer lainnya tanpa melakukan suatu perubahan apapun dengan suatu alat tertentu. (Fajar Junaedi EP, 2005 : 19) [14] Contoh dari Script HTML sebagai berikut Gambar 2.contoh script HTML

2.2.12 PHP PHP singkatan dari Hypertext Preprocessor yang digunakan sebagai bahasa Script serverside dalam pengembangan Web yang disisipkan pada dokumen HTML.[15]

PENGENLAN TENTANG SISTEM INFORMASI

KONSEP DASAR SISTEM

A. Definisi Sistem

Pengertian system menurut para ahli :

1. Menurut Fat Sistem adalah suatu himpunan “benda” nyata atau abstrak yang terdiri dari bagian-bagian atau komponen-komponen yang saling berkaitan, berhubungan, berketergantungan, saling mendukung, yang secara keseluruhan bersatu dalam satu kesatuan untuk mencapai tujuan tertentu secara efisien dan efektif.
2. Pengertian system menurut Indrajit (2001 : 2) mengemukakan bahwa system mengandung arti kumpulan-kumpulan dari komponen-komponen yang dimiliki unsur keterkaitan antara satu dengan lainnya.
3. Pengertian system menurut Jogianto (2005 : 2) mengemukakan bahwa system adalah kumpulan dari elemen-elemen yang berinteraksi untuk mencapai suatu tujuan tertentu. Sistem ini menggambarkan suatu kejadian-kejadian dan kesatuan yang nyata adalah suatu objek nyata, seperti tempat, benda, dan orang-orang yang betul-betul ada dan terjadi.
4. Pengertian system menurut Murdick, R.G, (1991 : 27) suatu system adalah seperangkat elemen yang membentuk kumpulan atau procedure-

procedure/bagan-bagan pengolah yang mencari suatu tujuan tertentu.

5. Pengertian system menurut Jerry FutGerald, (1982 : 5) system adalah suatu jaringan kerja dari procedure-prosedure yang saling berhubungan, berkumpul bersama-sama untuk melakukan suatu kegiatan atau untuk menyelesaikan suatu sasaran yang tertentu.
6. Pengertian system menurut Davis, G.B, (1991 : 5) system secara fisik adalah kumpulan dari elemen-elemen yang beroperasi bersama-sama untuk menyelesaikan suatu sasaran.
7. Definisi system menurut Dr. Ir. Harijono Djojodihardjo (1984 : 78) suatu system adalah sekumpulan objek yang mencakup hubungan fungsional antara tiap-tiap objek dan hubungan antara ciri tiap objek, dan yang secara keseluruhan merupakan suatu kesatuan secara fungsional.
8. Definisi system menurut Lani Sidharta (1995 : 9), system adalah himpunan dari bagian-bagian yang saling berhubungan yang secara bersama mencapai tujuan-tujuan yang sama.

Dengan demikian, system adalah suatu jaringan kerja dari procedure-prosedure yang saling berhubungan, berkumpul bersama-sama untuk melakukan kegiatan atau untuk melakukan sasaran yang tertentu. Pendekatan system yang merupakan jaringan kerja dari prosedur lebih menekankan urutan-urutan operasi di dalam

system. Menurut Richard F. Neuschel suatu prosedur adalah suatu urutan operasi klerikal (tulis-menulis), yang melibatkan beberapa orang di dalam satu atau lebih departemen, yang diterapkan untuk menjamin penanganan yang seragam dari transaksi-transaksi bisnis yang terjadi.

B. Karakteristik Sistem

Supaya system itu dikatakan system yang baik, maka memiliki karakteristik yaitu :

1. Komponen

Suatu system terdiri dari sejumlah komponen-komponen yang saling berinteraksi, yang artinya saling bekerja sama membentuk satu kesatuan. Komponen system terdiri dari komponen yang berupa subsistem atau bagian-bagian dari system.

2. Batasan Sistem (Boundary)

Batasan system merupakan daerah yang membatasi antara suatu system dengan system yang lain atau dengan lingkungan luarnya. Batasan system ini memungkinkan suatu system dipandang sebagai suatu kesatuan. Batasan system menunjukkan ruang lingkup (scope) dari system tersebut.

3. Lingkungan Luar Sistem (environment)

Lingkungan luar system (environment) adalah diluar batas dari system yang mempengaruhi operasi system. Lingkungan dapat bersifat

menguntungkan yang harus tetap dijaga dan yang merugikan yang harus dijaga dan dikendalikan, kalau tidak akan mengganggu kelangsungan hidup system.

4. Penghubung Sistem (Interface)

Penghubung system merupakan media penghubung antara satu subsistem dengan subsistem lainnya. Melalui penghubung ini memungkinkan sumber-sumber daya mengalir dari subsistem ke subsistem lain. Keluaran (output) dari subsistem lain melalui penghubung.

5. Masukkan Sistem (Input)

Masukkan adalah energy yang dimasukkan kedalam system, yang dapat berupa perawatan (maintenance input), dan masukkan sinyal (dignal input). Maintenance input adalah energy yang dimasukkan agar system dapat beroperasi. Signal input adalah energy yang diproses untuk didapatkan keluaran. Contoh dalam system computer program adalah maintenance input sedangkan data adlah signal input untuk diolah menjadi informasi.

6. Keluaran Sistem (Output)

Keluaran system adalah hasil dari energy yang dioalh dan diklasifikasikan menjadi keluaran yang berguna dan sisa pembuangan. Contoh computer menghasilkan panas yang merupakan

siswa pembuangnya, sedangkan informasi adalah keluaran yang dibutuhkan.

7. Pengolahan System

Suatu system menjadi bagian pengolah yang akan merubah masukanmenjadi keluaran. Sistem produksi akan mengolah bahan baku menjadi bahan jadi, system akuntansi akan mengolah data menjadi laporan-laporan keuangan.

8. Sasaran System

Suatu system pasti mempunyai tujuan atau sasaran (objective). Sasaran dari system sangat menentukan input yang dibutuhkan system dan keluaran yang akan dihasilkan system.

C. Klasifikasi Sistem

Sistem dapat diklasifikasikan dalam beberapa sudut pandang :

1. Klasifikasi system sebagai :

a. Sistem Abstrak (Abstracct System)

System abstrak adalah system yang berupa pemikiran-pemikiran atau ide-ide yang tidak tampak seperti fisik.

b. System Fisik (Physical System)

System fisik adalah system yang ada secara fisik

2. Sistem diklasifikasikan sebagai :

- a. System Alamiyah (Natural System)
System alamiyah adalah system yang terjadi melalui proses alam, tidak dibuat oleh manusia.
- b. System Buatan Manusia (Human Made System)
System buatan manusia adalah system yang dibuat oleh manusia yang melibatkan interaksi antara manusia dengan mesin (human mechine system).

3. System diklasifikasikan sebagai :

- a. System tertentu (Deterministic System)
System tertentu adalah system yang beroperasi dengan tingkah laku yang sudah dapat diprediksi, sebagai keluaran system yang dapat diramalkan.
- b. System tak tentu (Probabilistic System)
System tak tentu adalah system yang kondisi masa depannya tidak dapat diprediksi karena mengandung unsur probabilistic.

4. System diklasifikasikan sebagai :

- a. System tertutup (Close Sistem)
System tertutup adalah system yang tidak terpengaruh dan tidak berhubungan dengan lingkungan luar, system bekerja otomatis

tanpa ada turut campur lingkungan luar. Secara teoritis system tertutup ini ada, kenyataannya tidak ada system yang benar-benar tertutup, yang ada hanya relatively closed system.

b. System terbuka (Open Sistem)

System terbuka adalah system yang berhubungan dan terpengaruh dengan lingkungan luarnya. System ini menerima input dan output dari lingkungan luar atau subsistem lainnya, karena system terbuka terpengaruh lingkungan luar maka harus mempunyai pengendali yang baik.

KONSEP DASAR INFORMASI

A. Definisi Data

1. Menurut Gordon B. Davis, data adalah bahan mentah bagi informasi, dirumuskan sebagai kelompok lambing-lambang tidak acak menunjukkan jumlah-jumlah, tindakan-tindakan, hal-hal dan sebagainya.
2. Metode Pengumpulan Data :
 - a. Melalui pengamatan sendiri secara langsung
 - b. Melalui wawancara
 - c. Melalui perkiraan korespondensi
 - d. Melalui daftar pertanyaan

3. Pengertian Pengelolaan Data :

George R. Terry Ph.D menyatakan, Pengolahan data adalah serangkaian operasi atas informasi yang direncanakan guna mencapai tujuan atau hasil yang diinginkan.

8 unsur pokok pengolahan data, yaitu :

- a. Membaca
- b. Menulis, mengetik
- c. Mencatat atau mencetak
- d. Menyortir
- e. Menyampaikan atau memindahkan
- f. Menghitung
- g. Membandingkan
- h. Menyimpan

B. Definisi Informasi

Informasi adalah data yang diolah menjadi bentuk yang lebih berguna dan lebih berarti bagi penerimanya. Sumber informasi adalah data. Data kenyataan yang menggambarkan suatu kejadian-kejadian dan kesatuan nyata. Kejadian-kejadian adalah kejadian yang terjadi pada saat tertentu.

Menurut Gordon B. Davis : informasi adalah data yang telah diolah menjadi suatu bentuk yang penting bagi si penerima dan mempunyai nilai nyata atau yang dapat dirasakan dalam keputusan-keputusan yang sekarang atau keputusan-keputusan yang akan datang.

1. Fungsi Informasi

Fungsi utamanya yaitu : menambah pengetahuan atau mengurangi ketidakpastian pemakai informasi, karena informasi berguna memberikan gambaran tentang suatu permasalahan sehingga pengambil keputusan dapat menentukan keputusan lebih cepat, informasi juga memberikan standard, aturan maupun indicator bagi pengambil keputusan.

2. Kegunaan Informasi tergantung pada :

- a. Tujuan si penerima :
Bila tujuannya untuk memberi bantuan, maka informasi itu harus membantu si penerima dalam apa yang ia usahakan untuk memperolehnya.
- b. Ketelitian penyampain dan pengolahan data.
Dalam menyampaikan dan mengolah data, inti dan pentingny informasi harus dipertahankan.
- c. Waktu
Yaitu tentang keupdatean informasi
- d. Ruang dan tempat
Ketersediaan informasi dalam ruangan atau tempat yang tepat.
- e. Bentuk
Dapatkah informasi itu digunakan secara efektif. Apakah informasi itu menunjukkan hubungan-hubungan yang diperlukan, bidang-bidang yang memerlukan perhatian mannajemen ? dan apakah informasi itu menentukan situasi-situasi yang ada hubungannya.
- f. Semantic

Apakah hubungan antara kata-kata dan arti yang diinginkan cukup jelas ? apakah ada kemungkinan salah tafsir ?

C. Siklus Informasi

Data yang diolah untuk menghasilkan informasi menggunakan model proses yang tertentu. Misalkan suhu dalam Fahrenheit diubah ke celcius.

Dalam hal ini digunakan model matematik berupa rumus konversi dari derajat fahrenheit menjadi satuan derajat celcius. Data yang diolah melalui suatu model menjadi informasi tersebut, yang berarti menghasilkan keputusan dan melakukan tindakan yang lain yang akan membuat sejumlah data kembali. Data tersebut akan ditangkap sebagai input, diproses kembali lewat suatu model dan seterusnya yang disebut dengan siklus informasi. Siklus ini juga disebut dengan siklus pengolahan data.

D. Nilai Informasi (Value Of Information)

Nilai informasi ditentukan oleh dua hal yaitu manfaat dan biaya mendapatkannya. Suatu informasi dikatakan lebih bernilai jika manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.

Biaya informasi terdiri dari :

1. Biaya perangkat keras :

Merupakan biaya tetap atau biaya tertanam dan akan meningkat untuk tingkat-tingkat mekanisasi yang lebih tinggi.

2. Biaya untuk analisis :

Merupakan biaya tertanam, dan biasanya akan meningkat sesuai dengan tingkat mekanisasi yang lebih tinggi

3. Biaya untuk tempat dan faktor control lingkungan. Biaya ini setengah berubah/semivariable. Biasanya biaya ini meningkat sesuai dengan tingkat mekanisasi yang tinggi.

4. Biaya perubahan :

Biaya ini merupakan biaya tertanam dan meliputi setiap jenis perubahan dari suatu metode ke metode yang lain.

5. Biaya operasi :

Biaya ini pada dasarnya merupakan biaya variable dan meliputi biaya macam-macam pegawai, pemeliharaan fasilitas dan system.

KONSEP DASAR SISTEM INFORMASI

A. Konsep Dasar Sistem Informasi

System informasi adalah suatu system didalam suatu organisasi yang mempertemukan kebutuhan pengelolaan transaksi harian, mendukung operasi, bersifat manajerial, dan kegiatan strategi dari suatu organisasi dan menyediakan pihak luar tertentu dengan laporan-laporan yang dibutuhkan.

B. Konsep Sistem Informasi

System informasi terdiri dari komponen-komponen yang disebut dengan istilah blok bangunan (building block) yaitu :

1. Blok masukan (input block)
Input mewakili data yang masuk ke dalam system informasi. Input disini termasuk metode-metode dan media yang digunakan untuk menangkap data yang akan dimasukkan, yang dapat berupa dokumen dasar.
2. Blok model (model block)
Blok ini terdiri dari kombinasi prosedur, logika dan metode matematik yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara yang sudah tertentu untuk menghasilkan keluaran yang sudah diinginkan.
3. Blok keluaran (output block)
Produk dari system informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkatan manajemen serta semua pemakai system.
4. Blok teknologi (technology block)
Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan dan mengirimkan keluaran

dan membantu pengendalian diri secara keseluruhan.

Teknologi terdiri dari unsur utama :

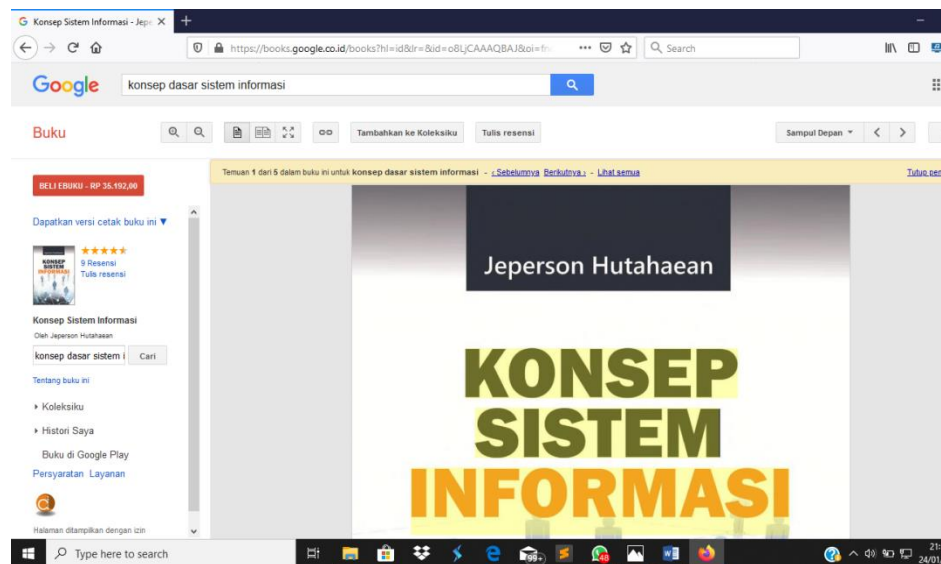
- a. Teknisi (human ware atau brain ware)
- b. Perangkat lunak (software)
- c. Perangkat keras (hardware)

5. Blok basis data (data base block)

Merupakan kumpulan dari data yang saling berhubungan satu dengan yang lainnya, tersimpan diperangkat keras computer dan digunakan perangkat lunak untuk memanipulasinya.

6. Blok kendali (control block)

Banyak faktor yang dapat merusak system informasi, misalnya bencana alam, api, temperature tinggi, air, debu, kecurangan-kecurangan, kegagalan system itu sendiri, kesalahan-kesalahan ketidakefisienan, sabotase dan sebgainya. Beberapa pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak system dapat dicegah atau bila terlanjur terjadi kesalahan dapat langsung diatasi.



PENGENALAN TENTANG WEBSITE

Website adalah sebuah kumpulan halaman pada suatu domain di internet yang dibuat dengan tujuan tertentu dan saling berhubungan serta dapat diakses secara luas melalui halaman depan (home page) menggunakan sebuah browser menggunakan URL website.

A. Sejarah Website

Website pertama kali dibuat oleh Tim Berners-Lee pada akhir 1980an dan baru resmi online tahun 1991. Tujuan awal Tim Berners-Lee membuat sebuah website adalah supaya lebih memudahkan para peneliti di tempatnya bekerja ketika akan bertukar atau melakukan perubahan informasi. Website mulai dapat digunakan secara gratis oleh public baru diumumkan oleh CERN tepatnya tanggal 30 April 1993. Website dapat dimiliki oleh individu, organisasi, atau perusahaan. Pada umumnya sebuah website akan menampilkan informasi atau satu topik tertentu, meskipun saat ini banyak website yang menampilkan berbagai informasi dengan topik yang berbeda.

B. Unsur-Unsur Website

Terdapat 3 unsur yang sangat vital pada website.

1. Domain

Jika website diibaratkan sebagai produk, maka domain adalah merk. Penggunaan domain yang menarik akan membuat orang tertarik untuk memasuki suatu website. Dengan pemilihan nama domain yang unik juga membuat orang mudah mengingatnya untuk nanti dikunjungi kembali.

2. Hosting

Tidak kalah pentingnya dengan domain, hosting memiliki peran untuk menyimpan semua database (script, gambar, video, teks dan lain

sebagainya) yang diperlukan untuk membentuk suatu website. Banyak sekali penyedia jasa hosting di Indonesia.

3. Konten

Tanpa adanya konten pada website, maka website bisa dikatakan tidak memiliki tujuan yang jelas. Konten pada website dapat berupa teks, gambar atau video. Jika dilihat dari konten yang disuguhkan, terdapat beberapa macam website. Misalnya saja, social media, website berita, website jual beli atau website yang berisi konten yang berdasarkan minat, bakat serta hobi.

C. Jenis-Jenis Website

Website merupakan media untuk menampilkan informasi. Awal kemunculannya, informasi yang ditampilkan pada website hanya berupa tulisan. Saat ini berbagai jenis website dapat anda temukan dengan mudah, tidak hanya menampilkan informasi berupa tulisan saja.

Berikut jenis-jenis Website :

1. Website Pribadi

Saat ini tersedia berbagai layanan yang dapat anda gunakan untuk membuat website pribadi. Mulai dari yang gratis sampai layanan berbayar. Untuk yang gratis, anda harus bersedia memiliki alamat website seperti

www.pengertianwebsite.com/usernameanda atau www.namawebsite.blogs . Sangat disarankan dalam pembuatan website melalui layanan berbayar penyedia domain dan hosting agar mengurangi persaingan dalam pencarian di mesin.

2. Website Toko Online

Dengan menggunakan sebuah tool yang tepat anda bisa membuatnya dalam hitungan jam bahkan menit, dan website toko online anda sudah dapat diakses melalui internet.

3. Blog

Blog biasanya berisi artikel yang bertujuan untuk berbagi pengetahuan, ide atau pengalaman penulis. Tujuan dari blog adalah untuk berbagi pengetahuan yang berhubungan dengan web hosting, internet marketing dan hal lainnya.

Untuk artikel yang diterbitkan setiap blog berbeda-beda, mulai dari satu hari satu artikel atau bahkan satu artikel dalam satu minggu. Hal ini tergantung dari kualitas artikel yang dibuat dan jumlah penulisnya.

D. Manfaat Website

Website memiliki banyak manfaat di kehidupan kita. Anda bisa mengakses media sosial menggunakan website. Belanja online juga terjadi di website. Ketika anda

membutuhkan informasi penting, anda juga bisa mengaksesnya melalui website.

Berikut beberapa manfaat website yang penting :

1. Bangun Personal Branding

Dimulai dari manfaat website untuk pribadi, yaitu website untuk personal branding. Dengan website anda bisa membangun personal branding yang kuat dan terpercaya. Anda bisa membagikan hasil karya dan portofolio anda di website. Baik itu tulisan, foto, lukisan, gambar, desain grafis, hingga music. Anda bisa membangun personal branding yang professional menggunakan website.

2. Berbagi Cerita Informasi

Contoh dari manfaat ini yaitu salah satunya seorang traveler yang membagikan kisah perjalanannya di dalam sebuah blog. Biasanya orang seperti itu disebut sebagai blogger. Tidak hanya dalam bidang traveller, anda bisa berbagi cerita dalam bentuk apapun. Misalnya tentang parenting, teknologi, keuangan, hingga kuliner.

3. Menghasilkan Uang

Ada banyak cara untuk menghasilkan uang menggunakan website. Anda bisa menjadi blogger dan mendapatkan banyak endorsement atau tawaran kerja

sama. Selain itu, anda juga bisa mendaftarkan website ke google adsense untuk mendapatkan uang dari iklan google.

Anda juga bisa mengikuti program affiliate marketing untuk menghasilkan uang dari website. Tugas sebagai affiliate adalah mempromosikan produk atau jasa dari perusahaan lain di website anda. Cara lain untuk menghasilkan uang melalui website adalah dengan membuat toko online.

PENGENALAN BAHASA PEMROGRAMAN

[Bahasa Pemrograman](#) merupakan sebuah instruksi untuk memerintah komputer agar bisa menjalankan fungsi tertentu, namun hanya instruksi standar saja. Seperti yang sudah Saya singgung di atas, bahasa pemrograman merupakan sebuah perhimpunan dari aturan sintaks dan semantik yang tugasnya untuk mendefinisikan [program komputer](#). Seseorang yang bisa memahami [bahasa pemrograman](#) dapat menentukan mana data yang akan di simpan / diteruskan, data mana yang akan di olah, dan langkah apa saja yang harus di ambil dalam berbagai situasi.

A. Fungsi Bahasa Pemrograman

Fungsi dari [bahasa pemrograman](#) adalah untuk memerintahkan sebuah komputer agar bisa mengolah data sesuai dengan yang kita inginkan. Jadi, kendali sepenuhnya ada di tangan kita. Keluaran dari bahasa pemrograman tersebut bisa berupa [aplikasi](#) ataupun program khusus, contohnya seperti lampu lalu lintas. Antara lampu merah, kuning, dan hijau memiliki urutan dan jeda waktu yang berbeda, sehingga para pengendara tahu kapan harus berhenti dan kapan harus jalan.

B. Pengelompokkan Bahasa Pemrograman

Bahasa pemrograman di bagi menjadi beberapa kelompok, yaitu :

- **Object Oriented Language** : contohnya adalah Visual C, Delphi, Visual dBase, Visual FoxPro.
- **Low Level Language** : contohnya adalah bahasa Assembly.
- **Middle Level Language** : contohnya adalah Bahasa C.
- **High Level Language** : contohnya Basic dan Pascal.

C. Generasi Bahasa Pemrograman

Generasi pemrograman dapat Kita golongan menjadi beberapa generasi, yaitu :

- **Generasi Pertama** : merupakan bahasa pemrograman yang pertama kali dibuat yaitu Machine Language.
- **Generasi kedua** : merupakan penerus dari generasi pertama yaitu Assembly Language : Asembler.
- **Kenerasi ketiga** : merupakan bahasa pemrograman yang memiliki Level tinggi (High Level Programing language). Contohnya adalah C dan Pascal.

- **Generasi kelima** : bisa di sebut dengan 4 atau kepanjangan dai fourth Generation Language. Contohnya adalah SQL.

- **Generasi kelima** : merupakan generasi kelima dari bahasa pemrograman yaitu Programing Language Based Object Oriented dan [Web Devolopment](#).

D. Tingkatan Bahasa Pemrograman

[Bahasa pemrograman](#) memiliki beberapa tingkatan yang harus anda tahu sebelum mengenal lebih jauh tentang bahasa pemrograman, yaitu :

- **Bahasa Tingkat Rendah** : bahasa tingkat rendah merupakan bahasa yang masih jauh sekali dari [bahasa manusia](#), susah untuk di mengerti. Bahasa yang masuk ke dalam tingkatan ini adalah Bahasa Assembly.

- **Bahasa Tingkat tinggi** : bahasa tingkat tinggi merupakan bahasa yang mendekati [bahasa manusia](#), mudah untuk di mengerti. Bahasa yang masuk ke dalam tingkatan ini adalah bahasa pascal, Basic, PHP< dan Java.

- **Bahasa Tingkat Menengah** : bahasa tingkat menengah merupakan perpaduan antara bahasa tingkat tinggi dan rendah, bahasa nya tidak sulit maupun tidak mudah untuk di mengerti [manusia](#).

E. Contoh Bahasa Pemrograman

Berikut ini ada beberapa [contoh bahasa pemrograman](#) yang paling umum digunakan, yaitu :

- **Java script**

[Java Script](#) merupakan bahasa pemrograman yang paling mudah untuk dikuasai. Fungsi java script dalam [situs web](#) maupun [aplikasi](#) adalah untuk menambahkan fitur interaktif, juga berguna untuk memvalidasi data ataupun formulir untuk diterapkan pada pengembangan sebuah permainan. [Belajar java script](#) merupakan pilihan yang sangat cocok Bagi seorang pemula.

• PHP

PHP merupakan Bahasa Pemrograman yang harus dipelajari jika [Halaman Web](#) anda tampil secara dinamis. Mempelajari PHP akan sangat berguna untuk [pemilik situs web](#) karena PHP bisa mempersingkat kode yang ada di halaman situs sehingga bisa ditampilkan ke dalam seperangkat aturan. Jadi kita tidak akan dibuat ribet jika kita memahami tentang PHP, ketika dibutuhkan bisa memanggil mereka kapan pun Anda mau.

• HTML

Walaupun banyak orang yang menganggap bahwa HTML / CSS bukan bagian dari bahasa pemrograman, namun sangat diperlukan bagi Anda yang masih awam dengan [cara membuat situs web](#). Belajar HTML merupakan pintu utama bagi Kalian yang bingung untuk memulai belajar bahasa pemrograman.

• C ++

Bagi Kalian yang ingin mengenal lebih jauh tentang [bahasa pemrograman](#), mungkin bisa dicoba untuk mempelajari bahasa C ++. Bahasa C ++ bisa dibilang sebagai bahasa tingkat menengah, jadi tidak akan terlalu mudah maupun terlalu susah. Keuntungan belajar C ++ adalah untuk

mempermudah bahasa C yang lainnya seperti
bahasa C dan C#.

Pengenalan Bahasa Pemrograman HTML

A. Pengertian HTML

HTML adalah singkatan dari Hypertext Mark Up Language yang merupakan bahasa standar pemrograman untuk membuat suatu website yang bisa diakses dengan internet. Dengan kata lain halaman website yang kita lihat dan kita baca disusun dengan menggunakan bahasa ini dan kemudian diterjemahkan oleh komputer agar dapat dipahami oleh penggunaannya. HTML merupakan standar pembuatan website secara luas agar laman website dapat ditampilkan pada layar komputer.

HTML adalah bahasa mark up web yang utama dan dijalankan secara alami di setiap browser serta di-maintain oleh World Wide Web Consortium.

HTML disusun dengan kode dan simbol tertentu yang dimasukkan ke dalam sebuah file atau dokumen. Jadi setiap Anda membuka website apapun dengan menggunakan browser maka web tersebut dibuat dengan menggunakan HTML.

Seperti yang sudah saya jelaskan sebelumnya, HTML adalah singkatan dari Hyper Text Markup Language. Tapi mengetahui singkatannya saja tidak akan cukup. Karena itu saya akan menyajikan arti dari kata kata tersebut.

- **HyperText** adalah metode dimana kita “berpindah” disekeliling web, dengan mengeklik sebuah teks yang bernama hyperlink. Hyperlink adalah sebuah teks khusus di internet, dimana saat teks tersebut diklik, akan membawa kita ke halaman web selanjutnya/halaman web lain yang telah ditentukan.
- **Markup** adalah hal yang dilakukan oleh tag HTML kepada teks yang ada didalamnya. HTML menandai teks yang berada didalamnya sebagai tipe teks tertentu. Misalnya saja jika kita menandai sebuah teks dengan tag html `<i>`, maka teks tersebut akan berubah menjadi italic (huruf yang miring). Sedangkan jika kita menandainya dengan ``, maka teks tersebut akan berubah menjadi bold (huruf tebal).
- **Language** yang berarti bahasa. HTML adalah sebuah bahasa, yang memiliki kata kata berupa kode dan syntax seperti bahasa yang lain.

Berdasarkan cara kerja HTML, Anda bisa menggunakannya untuk membuat struktur konten pada website dan aplikasi web. HTML merupakan level terbawah dari teknologi frontend dan berfungsi sebagai dasar *styling* yang bisa Anda tambahkan dengan CSS dan fungsionalitas yang dapat dijalankan menggunakan JavaScript.

B. Definisi HTML Menurut Para Ahli

Berikut ini ada beberapa pengertian HTML menurut para ahli:

- **Lia Kuswayatno**, HTML merupakan halaman yang berada pada suatu situs internet atau web. HTML merupakan metode yang menautkan (link) satu dokumen ke dokumen lain melalui teks
- **Deris Setiaan**, HTML merupakan framework internet, hampir semua situs web yang ada menggunakan HTML untuk menampilkan teks, grafik, suara, dan animasinya
- **Diar Puji Oktavian**, HTML adalah suatu bahasa yang dikenali oleh web browser untuk menampilkan informasi dengan lebih menarik dibandingkan dengan tulisan teks biasa (plain text)
- **Menurut Arief (2011c:23)**, “HTML (HyperText Markup Language) merupakan salah satu format yang digunakan dalam pembuatan dokumen dan aplikasi yang berjalan dihalaman web”.
- **Menurut Sibero (2011c:19)**, “Hyper Text Markup Language atau HTML adalah bahasa yang digunakan pada dokumen web sebagai bahasa untuk pertukaran dokumen web”. Dokumen HTML terdiri dari komponen yaitu tag, elemen dan atribut. Tag adalah tanda awal < dan tanda akhir > yang digunakan sebagai pengapit suatu elemen. Elemen adalah nama penanda yang diapit oleh tag yang memiliki fungsi dan tujuan tertentu pada dokumen HTML. Elemen dapat memiliki elemen anak dan juga nilai. Elemen anak adalah suatu elemen yang berada didalam elemen pembuka dan elemen penutup induknya. Nilai yang dimaksud adalah suatu teks atau karakter yang berada diantara elemen pembuka dan elemen penutup. Atribut adalah properti elemen yang digunakan untuk mengkhususkan suatu elemen. Elemen

dapat memiliki atribut yang berbeda pada tiap masing-masingnya.

- **Menurut Astamal (2006:8)**, HTML (Hyper Text Markup Language) adalah simbol-simbol atau tag-tag yang dituliskan dalam sebuah file yang dimaksudkan untuk menampilkan halaman pada web browser. Tag-tag HTML selalu diawali dengan < dan diakhiri dengan > dimana x tag HTML seperti b, i, u dan sebagainya. Namun ada juga tag yang tidak diakhiri dengan tanda </>.

C. Sejarah HTML

HTML dibuat oleh [Tim Berners-Lee](#), seorang ahli fisika di lembaga penelitian CERN yang berlokasi di Swiss. Dia memiliki ide tentang sistem hypertext yang berbasis internet.

Hypertext merujuk pada teks yang memuat referensi (link) ke teks lain yang bisa diakses langsung oleh viewer. Tim merilis versi pertama HTML pada tahun 1991, dan di dalamnya terdiri atas 18 HTML tag. Sejak saat itu, setiap kali bahasa HTML merilis versi teranyarnya, selalu ada tag dan attribute (tag modifier) terbaru.

Berdasarkan [HTML Element Reference](#) milik Mozilla Developer Network, untuk saat ini, ada 140 HTML tag meskipun sebagiannya sudah usang (tidak lagi didukung oleh versi terbaru browser).

Berkat popularitasnya yang terus meningkat, HTML kini dianggap sebagai web standard yang resmi. Spesifikasi HTML di-*maintain* dan dikembangkan oleh World Wide Web Consortium (W3C). Cek versi terbaru dari bahasa ini di [website W3C](#).

Upgrade HTML besar-besaran terjadi pada tahun 2014, dan hasilnya adalah pengenalan **HTML5**. Pada upgrade tersebut, terdapat semantic baru yang memberitahukan arti dari kontennya sendiri, seperti `<article>`, `<header>`, dan `<footer>`.

HTML pertama kali diciptakan oleh IBM pada tahun 1980. Saat itu tercetus ide untuk menempatkan elemen elemen yang berguna untuk menandai bagian suatu dokumen seperti judul, alamat dan isi dokumen. Pada akhirnya elemen element tersebut dibentuk menjadi suatu program untuk melakukan pemformatan dokumen secara otomatis.

Bahasa pemrograman untuk melakukan tugas ini disebut sebagai bahasa markup. IBM menamai program ini sebagai Generalized Markup Language. Pada tahun 1986, konsep ini disetujui oleh ISO (International Standard Organization) sebagai standar untuk pembuatan dokumen. Bahasa ini dinamai oleh ISO sebagai Standard Generalized Markup Language (SGML)

HTML sendiri, adalah bagian dari SGML. Tim Berners-Lee dari CERN mengemukakan sebuah ide tentang pembuatan skrip bahasa pemrograman dan dokumen yang dapat diakses oleh semua komputer tanpa melihat jenis platformnya. Sejak saat itu HTML menjadi lebih populer dibandingkan SGML.

Didalam HTML telah mengalami berbagai perkembangan dari tiap-tiap versinya. Berikut adalah versi-versi dari perkembangan HTML tersebut:

- **HTML Versi 1.0**

HTML Versi 1.0 merupakan pionir yang di dalamnya masih terdapat banyak sekali kelemahan

hingga wajar jika tampilan yang dihasilkan sangat sederhana. Kemampuan yang dimiliki versi 1.0 ini antara lain heading, paragraf, hypertext, list, serta cetak tebal dan miring pada teks. Versi ini juga mendukung peletakan image pada dokumennya tanpa memperbolehkan teks di sekelilingnya (wrapping)

- **HTML Versi 2.0**

Pada HTML Versi ini, penambahan kualitas HTML terletak pada kemampuannya untuk menampilkan suatu form pada dokumen. Dengan adanya form ini, kita dapat memasukkan nama, alamat, serta saran dan kritik. HTML versi 2.0 ini merupakan pionir dari adanya web interaktif.

- **HTML Versi 3.0**

Versi HTML 3.0 menambahkan beberapa fasilitas baru seperti **FIGURE** yang merupakan perkembangan dari **IMAGE** untuk meletakkan gambar dan tabel. Selain itu, HTML ini juga mendukung adanya rumus-rumus matematika dalam dokumennya. Versi ini yang disebut HTML+- tidak bertahan lama dan segera digantikan dengan versi 3.2.

- **HTML Versi 3.2**

HTML versi ini merupakan HTML yang sering digunakan. Di dalamnya terdapat suatu teknologi untuk meletakkan teks di sekeliling gambar, gambar sebagai latar belakang, tabel, frame, style sheet dan lain-lain. Selain itu pada HTML versi ini Kita bisa menggunakan script di luar HTML untuk mendukung kinerja HTML kita tersebut, seperti Javascript, VBScript dan lain-lain.

- **HTML Versi 4**

HTML ini memuat banyak sekali perubahan dan revisi dari pendahulunya yaitu HTML 3.2. Perubahan ini hampir terjadi di segala perintah HTML seperti tabel, image, link, text, meta, imagemaps, form dan lain-lain.

- **HTML Versi 4.01**

HTML versi 4.01 merupakan revisi dari HTML 4.0. Versi terbaru ini memperbaiki kesalahan-kesalahan kecil (minor errors) pada versi terdahulunya. HTML 4.01 ini juga menjadi standarisasi untuk elemen dan atribut dari script XHTML 1.0.

- **HTML Versi 5.0**

Teknologi ini mulai diluncurkan pada tahun 2009, tetapi pada tanggal 4 Maret 2010 terdapat sebuah informasi bahwa W3C (World Wide Web Consortium) dan IETF (Internet Engineering Task Force) yaitu sebuah organisasi yang menangani HTML sejak versi 2.0 telah mengembangkan versi HTML terbaru, yaitu versi 5.0.

HTML 5 adalah sebuah prosedur pembuatan tampilan web baru yang merupakan penggabungan antara CSS, HTML itu sendiri dengan JavaScript.

Beberapa kelebihan yang dimiliki oleh HTML5 adalah:

- **Cleaner code**

Karena sebagian besar kode telah termasuk di dalam sintaks HTML5, maka kode nampak terlihat lebih sederhana daripada penggabungan antara HTML, CSS dan Java Script.

- **Greater consistency**

HTML5 telah melakukan banyak sekali penambahan sintaks yang dibuat dalam struktur lebih baik dan lebih sederhana daripada sintaks-sintaks sebelumnya. Hal ini membuat developer terbantu dalam meningkatkan konsistensi dalam membangun sebuah web.

- **Improve Semantics**

Berbagai elemen kode di dalam HTML5 yang telah distandarisasi, maka nilai semantik dari sebuah web dapat lebih ditingkatkan. Itu berarti bahwa bagian-bagian dari web seperti header, nav, footer dan beberapa bagian lainnya terdefinisi dengan jelas maksud serta tujuannya. Selain itu juga terbentuk dalam sebuah “machine readable format”.

- **Improved Accessibility**

Teknologi HTML5 yang memudahkan struktur pembangunan sebuah web, maka developer dapat membangun pemahaman yang lebih detil mengenai halaman web.

- **Client-side Database**

HTML5 menyediakan model database SQL yang baru dengan API yang dapat dibangun dalam konsep lokal, dalam hal ini di sisi client.

- **Geolocation**

HTML5 mempunyai API yang terintegrasi terhadap geolocation, fasilitas tersebut dapat diakses melalui GPS atau fasilitas lain seperti Google Latitude pada iphone.

- **Offline Application Cache**

Pengguna dapat terus melakukan interaksi dengan aplikasi meskipun mereka terputus dari jaringan internet.

- **Smarter Forms**

Terdapat semacam reguler expression (regex) yang membuat form mampu mengenali secara lebih baik tentang input, validasi data dan interaksi dengan elemen lain (misal : format email, password dll)

- **Sharper focus on Web Application Requirments**

HTML5 membuat sebuah mekanisme yang lebih mudah dalam hal pembuatan front end, aplikasi chat, tools drag and drop, video player, pengolah grafis dan masih banyak lagi.

D. Fungsi HTML

Html yang dikenal sebagai bahasa pemrograman dalam ilmu komputer memiliki beberapa fungsi, diantaranya adalah sebagai berikut:

1. Fungsi utama html yang diketahui adalah untuk membuat suatu halaman website yang bisa dibaca dan dipahami oleh pengguna dengan lebih mudah. Seluruh laman website yang ada dalam internet dibuat dengan html dan tidak ada pengecualian.
2. Menandai teks pada suatu laman, html ditulis pada suatu halaman dokumen dengan tag atau simbol tertentu dimana simbol dan tag tersebut akan menandai teks menjadi tebal, miring, bergaris tebal dan lain sebagainya. Misal jika kita membuat suatu teks menjadi teks miring

atau italic, dalam laman html dituliskan kode <i>, untuk teks tebal dan <u> untuk teks bergaris bawah.

3. Sebagai dasar website, website yang dibuat tentunya memiliki beberapa fitur yang dibuat dengan menggunakan java script (untuk mengatur perilaku web), implementasi bahasa pemrograman server PHP, dan mendesain web menggunakan CSS. Semua bahasa tersebut dapat diaplikasikan jika web memiliki bahasa html sebagai dasarnya.
4. Menampilkan tabel, gambar, video, dan lainnya. Biasanya dalam website atau blog kita tidak bisa langsung meletakkan tabel, gambar maupun video oleh sebab itu komponen tersebut diletakkan pada web dengan menggunakan bahasa html.
5. Menandai elemen dan membuat online form, html juga berfungsi untuk menandai bagian-bagian dalam website diantaranya header, main, footer, navigation dan lain sebagainya. Selain itu html juga biasanya digunakan sebagai bahasa dalam membuat suatu online form atau formulir digital.

E. Struktur HTML

Berikut ini ada beberapa struktur HTML:

1. Elemen

Elemen terdiri atas tiga bagian, yaitu tag pembuka, isi, dan tag penutup. Contohnya untuk menampilkan judul dokumen HTML pada web browser digunakan element title, dimana: <title> ini adalah tag pembuka judul dokumen HTML.

2. Tag

Tag merupakan teks khusus (markup) berupa dua karakter “<” dan “>”, sebagai contoh<body> adalah tag dengan nama body. Tag ditulis secara berpasangan, yang terdiri atas tagpembuka dan tag penutup (ditambahkan karakter “/” setelah karakter “<”), sebagai contoh<body> ini adalah tag pembuka isi dokumen HTML, dan </body> ini adalah tag penutup isi dokumen HTML. Yang merupakan tag-tag dasar dalam HTML adalah:

- <html> </html>

Tag <html> dan </html> digunakan untuk menandai awal dan akhir dari suatu file HTML.

- <title> </title>

Tulisan yang berada diantara tag <title> dan </title> akan ditampilkan oleh browser pada bagian title dan merupakan title dari jendela browser.

- <head> </head>

Berisi keterangan informasi, seperti title dan jenis dokumen, ditulis diantara <head> </head>.

- <body> </body>

Bagian tag <body> </body> menandai awal dan akhir dari badan dokumen HTML. Tag ini memiliki sejumlah attribut yang dapat ditentukan.

Contoh :

```
<html>
```

```
<head>
```

```

<title>Contoh Document HTML</title>

</head>

<body>SELAMAT MEMBACA</body>

</html>

```

3. Atribut

Atribut terdapat di dalam script sebuah elemen dan memberikan informasi tambahan tentang elemen. Atribut selalu ditentukan dalam tag awal. Atribut ditulis dalam pasangan nama/nilai.

Nilai dari atribut harus selalu tertutup dalam tanda kutip. Tanda kutip ganda adalah yang paling umum, tapi gaya tanda kutip tunggal juga diperbolehkan. Dalam beberapa situasi, ketika nilai atribut itu sendiri berisi tanda kutip, maka perlu menggunakan tanda kutip tunggal. Misalnya: name = 'John "leo" Nelson'.

Di bawah ini adalah daftar dari beberapa atribut yang dapat digunakan pada setiap elemen HTML :

| Atribut | Deskripsi |
|---------|---|
| Class | Menentukan satu atau lebih classnames untuk sebuah elemen (mengacu pada kelas dalam style sheet). |
| Id | Menentukan id unik untuk sebuah elemen. |
| Style | Menentukan inline CSS style untuk elemen. |
| Title | Menentukan informasi tambahan mengenai elemen (ditampilkan sebagai tool tip). |

F. Cara Kerja HTML

Dokumen HTML adalah file yang diakhiri dengan ekstensi **.html** atau **.htm**. Ekstensi file ini bisa dilihat dengan menggunakan web browser apa pun (seperti Google Chrome, Safari, atau Mozilla Firefox). Browser tersebut membaca file HTML dan *me-render* kontennya sehingga user internet bisa melihat dan membacanya.

Biasanya, rata-rata situs web menyertakan sejumlah halaman HTML yang berbeda-beda. Contohnya, beranda utama, halaman ‘tentang kami’, halaman kontak yang semuanya memiliki dokumen HTML terpisah.

Masing-masing halaman HTML terdiri atas seperangkat **tags** (bisa disebut juga **elements**), yang mengacu pada *building block* halaman website. Tag tersebut membuat hirarki yang menyusun konten hingga menjadi bagian, paragraf, heading, dan *block* konten lainnya.

Sebagian besar element HTML memiliki tag pembuka dan penutup yang menggunakan syntax `<tag></tag>`.

Berikut contoh kode dari susunan atau struktur HTML:

```
<div>
  <h1>The Main Heading</h1>
  <h2>A catchy subheading</h2>
  <p>Paragraph one</p>
  
  <p>Paragraph two with a <a href="https://example.com">hyperlink</a>
</p>
</div>
```

- Elemen teratas dan terbawah adalah division sederhana (`<div></div>`) yang bisa Anda gunakan untuk *mark up* bagian konten yang lebih besar.

- Susunan HTML di atas terdiri atas heading (`<h1></h1>`), subheading (`<h2></h2>`), dua paragraf (`<p></p>`), dan satu gambar (``).
- Paragraf kedua meliputi sebuah link (`<a>`) dengan attribute href yang terdiri atas URL tujuan.
- Tag gambar memiliki dua attribute, *src* untuk path gambar dan *alt* untuk deskripsi gambar.

G. Gambaran Umum Tentang Tag HTML yang Sering Digunakan

Tag HTML memiliki dua tipe utama: **block-level** dan **inline tags**.

1. Elemen *block-level* memakai semua space yang tersedia dan selalu membuat line baru di dalam dokumen. Contoh dari tag *block* adalah heading dan paragraf.
2. Elemen *inline* hanya memakai space sesuai dengan kebutuhannya dan tidak membuat line baru di halaman. Biasanya elemen ini akan memformat isi konten dari elemen *block-level*. Contoh dari tag *inline* adalah link dan *emphasized strings*.

H. Tag Block-Level

Tiga tag block-level yang harus dimiliki oleh setiap dokumen HTML adalah `<html>`, `<head>`, dan `<body>`.

1. Tag **<html></html>** adalah elemen level tertinggi yang menyertakan setiap halaman HTML.
2. Tag **<head></head>** menyimpan informasi meta, seperti judul dan *charset* halaman.
3. Tag **<body></body>** melampirkan semua konten yang muncul pada suatu halaman.

```
<html>
  <head>
    <!-- META INFORMATION -->
  </head>
  <body>
    <!-- PAGE CONTENT -->
  </body>
</html>
```

- Heading memiliki 6 level di HTML. Level tersebut bervariasi, mulai dari **<h1></h1>** sampai ke **<h6></h6>**, di mana h1 merupakan level heading tertinggi dan h6 adalah level terendah. Paragraf dibuka dan ditutup dengan tag **<p></p>**, sedangkan blockquote menggunakan tag **<blockquote></blockquote>**.
- Division merupakan bagian konten yang lebih besar dan biasanya terdiri atas beberapa paragraf, gambar, kadang-kadang blockquote, dan elemen lebih kecil lainnya. Kita bisa membuat *mark up* dengan menggunakan tag **<div></div>**. Di dalam elemen div juga terdapat tag div lainnya.
- Anda juga bisa menggunakan tag **** untuk list yang berurutan

dan `` untuk list yang tidak berurutan. Masing-masing list item harus dibuka dan ditutup dengan tag ``. Sebagai contoh, di bawah ini adalah tampilan dasar dari list tidak berurutan dalam HTML:

```
<ul>
  <li>List item 1</li>
  <li>List item 2</li>
  <li>List item 3</li>
</ul>
```

I. Tag Inline

Sebagian besar tag inline digunakan untuk memformat teks. Sebagai contoh, tag `` akan render elemen ke format **bold**, sedangkan tag `` akan ditampilkan dalam format *italic*.

Hyperlink adalah elemen inline yang mewajibkan adanya tag `<a>` dan attribute **href** untuk mengindikasikan tujuan link:

```
<a href="https://example.com/">Click me!</a>
```

Gambar (image) juga merupakan elemen inline. Anda dapat menambahkan satu gambar dengan menggunakan `` tanpa harus membubuhkan tag penutup. Hanya saja, Anda disarankan menggunakan attribute **src** untuk menentukan path gambar, misalnya:

```

```

J. Kelebihan dan Kekurangan HTML

A. Kelebihan

- Bahasa yang digunakan secara luas dan memiliki banyak sumber serta komunitas yang besar.
- Dijalankan secara alami di setiap web browser.
- Memiliki *learning curve* yang mudah.
- Open-source dan sepenuhnya gratis.
- Bahasa markup yang rapi dan konsisten.
- Standard web yang resmi di-maintain oleh World Wide Web Consortium (W3C).
- Mudah diintegrasikan dengan bahasa *backend*, seperti PHP dan Node.js.

B. Kekurangan

- Paling sering digunakan untuk halaman web statis. Untuk fitur dinamis, Anda bisa menggunakan JavaScript atau bahasa backend, seperti PHP.
- HTML tidak memungkinkan user untuk menjalankan *logic*. Alhasil, semua halaman web harus dibuat terpisah meskipun menggunakan elemen yang sama, seperti header dan footer.
- Fitur-fitur baru tidak bisa digunakan secara cepat di sebagian browser.
- Terkadang perilaku browser susah untuk diprediksi (misalnya, browser lama tidak selalu bisa render tag yang lebih baru).

Pengenalan Bahasa Pemrograman PHP

A. Pengenalan PHP

PHP adalah singkatan dari “PHP: Hypertext Preprocessor”, yaitu bahasa pemrograman disisi server yang digunakan secara luas untuk penanganan pembuatan dan pengembangan sebuah situs web dan bisa digunakan bersamaan dengan HTML. Ketika Anda mengakses sebuah URL, maka web browser akan melakukan request ke sebuah web server.

PHP adalah bahasa pemrograman yang sering disisipkan ke dalam HTML. Sejarah PHP pada awalnya merupakan kependekan dari *Personal Home Page* (Situs personal). PHP pertama kali

dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan skrip yang digunakan untuk mengolah data formulir dari web.

Bahasa pemrograman PHP biasanya tidak digunakan pada keseluruhan pengembangan website, melainkan dikombinasikan dengan beberapa bahasa pemrograman lain. Misalnya saja untuk mengatur tampilan, layout, dan berbagai macam menu menggunakan CSS.

Selain itu, terdapat juga beberapa framework PHP; Laravel, Phalcon, Codigniter, Symfoni yang saat ini banyak tersedia di internet untuk memudahkan proses pengembangan website menggunakan bahasa pemrograman tersebut.

Bahasa pemrograman ini menggunakan sistem server-side. *Server-side programming* adalah jenis bahasa pemrograman yang nantinya script/program tersebut akan dijalankan/diproses oleh server. Kelebihannya adalah mudah digunakan, sederhana, dan mudah untuk dimengerti dan dipelajari.

Di bawah ini adalah contoh penggunaan PHP untuk menghubungkan dan membuat database.

```

<?php
$servername = "localhost";
$username = "username";
$password = "password";
// Create connection
$conn = new mysqli($servername, $username, $password);
// Check connection
if ($conn->connect_error) {
    die("Koneksi gagal: " . $conn->connect_error);
}
// Create database
$sql = "CREATE DATABASE myDB";
if ($conn->query($sql) === TRUE) {
    echo "Database berhasil dibuat!";
} else {
    echo "Error membuat database: " . $conn->error;
}
$conn->close();
?>

```

Pada setiap bahasa pemrograman, terdapat yang namanya variable. **variabel** adalah suatu lokasi penyimpanan (di dalam memori komputer) yang berisikan nilai atau informasi yang nilainya tidak diketahui maupun telah diketahui. Nilai – nilai yang

terdapat pada variabel ini memiliki berbagai jenisnya. Pada Bahasa pemrograman pun nilai – nilai ini biasanya disebut sebagai data – data. Pada PHP ini, ada beberapa tipe data yang dapat digunakan antara lain tipe data berbentuk *Integer*, *Float*, *String*, *Boolean*, *Array*, *Object*, *NULL*. Pada bahasa pemrograman lainnya, tipe data – tipe data tersebut haruslah dituliskan berdasarkan format bahasa pemrograman itu sendiri. Pada PHP, penulisan tipe data – tipe data tersebut sangat mudah sekali, hanya perlu membuat sebuah variabel dan menambahkan tanda dolar (\$) di depannya. Berikut pengertian dan contoh penggunaan tipe data tersebut pada bahasa pemrograman PHP :

- Berikut contoh variabel tipe data *Integer* : \$x = 20;
- Berikut contoh variabel tipe data *Float* : \$x = 20.1547521;
- Berikut contoh variabel tipe data *String* : \$x = "Saya";
- Berikut contoh variabel tipe data *Boolean* : \$x = true; || \$x = false;
- Berikut contoh variabel tipe data *Array* : \$x = array("saya","suka","kamu");
- Berikut contoh variabel tipe data *Object* : \$x = new namaClass();
- Berikut contoh variabel tipe data *NULL* : \$x = null;

Selain itu, pada PHP terdapat juga yang namanya *Constans*. *Constans* ini merupakan sebuah tanda pengenal untuk sebuah nilai yang simple. Nilai ini tidak dapat diubah. Untuk membuat *Constans* ini dapat menggunakan syntax "define(nama variabel, nilai variabel, *case-sensitive*)". *Case-sensitive* disini memberitahukan apakah *Constans* itu harus sama persis atau tidak. Kemudian untuk memunculkan variabel yang telah di berikan nilai, dapat menggunakan syntax "echo" dan "print" untuk memunculkannya pada halaman website. Contoh penggunaannya adalah " echo \$x; " dan " print \$x; ". Tetapi banyak para programmer lebih memilih menggunakan echo dalam memunculkan

atau menampilkan sesuatu pada bahasa pemrograman PHP ini.

Pada PHP ini sendiri memiliki beberapa *Operators*. *Operators* ini berguna untuk membantu dalam pengoperasian sebuah variabel atau nilai. Berikut beberapa contoh operator pada PHP :

B. Pengertian PHP Menurut Beberapa Ahli

A. Menurut Arief (2011c:43) [PHP](#) adalah Bahasa server-side –scripting yang menyatu dengan HTML untuk membuat halaman web yang dinamis. Karena [PHP](#) merupakan server-side-scripting maka sintaks dan perintah-perintah PHP akan dieksekusi di server kemudian hasilnya akan dikirimkan ke browser dengan format HTML. Dengan demikian kode program yang ditulis dalam PHP tidak akan terlihat oleh user sehingga keamanan halaman web lebih terjamin. PHP dirancang untuk membuat halaman web yang dinamis, yaitu halaman web yang dapat membentuk suatu tampilan berdasarkan permintaan terkini, seperti menampilkan isi basis data ke halaman web.

B. Sedangkan menurut Nugroho (2006 b:61) “PHP atau singkatan dari Personal Home Page merupakan bahasa skrip yang tertanam dalam HTML untuk dieksekusi bersifat server side”. PHP termasuk dalam open source product, sehingga source code PHP dapat diubah dan didistribusikan secara bebas. Versi terbaru PHP dapat diunduh secara gratis melalui situs resmi PHP : <http://www.php.net>.

PHP juga dapat berjalan pada berbagai web server seperti IIS (Internet Information Server), PWS (Personal Web Server), Apache, Xitami. PHP juga mampu berjalan di banyak sistem operasi yang beredar saat ini, diantaranya : Sistem Operasi Microsoft Windows (semua versi), Linux, Mac Os, Solaris. PHP dapat dibangun sebagai modul web server Apache dan sebagai binary yang dapat berjalan sebagai CGI (Common Gateway Interface).

PHP dapat mengirim HTTP header, dapat mengatur cookies , mengatur authentication dan redirect user. Salah satu keunggulan yang dimiliki PHP adalah kemampuannya untuk melakukan koneksi ke berbagai macam software sistem manajemen basis data atau Database Management Sistem (DBMS), sehingga dapat menciptakan suatu halaman web dinamis.

PHP mempunyai koneksitas yang baik dengan beberapa DBMS seperti Oracle, Sybase, mSQL, MySQL, Microsoft SQL Server, Solid, PostgreSQL, Adabas, FilePro, Velocis, dBase, Unix dbm, dan tidak terkecuali semua database ber-interface ODBC.

Hampir seluruh aplikasi berbasis web dapat dibuat dengan PHP. Namun kekuatan utama adalah konektivitas basis data dengan web. Dengan kemampuan ini kita akan mempunyai suatu sistem basis data yang dapat diakses.

C. Sejarah PHP

Pada awalnya PHP merupakan kependekan dari Personal Home Page (Situs personal). PHP pertama kali dibuat oleh Rasmus Lerdorf pada tahun 1995. Pada waktu itu PHP masih bernama Form Interpreted (FI), yang wujudnya berupa sekumpulan

skrip yang digunakan untuk mengolah data formulir dari web.

Selanjutnya Rasmus merilis kode sumber tersebut untuk umum dan menamakannya PHP/FI. Dengan perilsan kode sumber ini menjadi sumber terbuka, maka banyak pemrogram yang tertarik untuk ikut mengembangkan PHP.

Pada November 1997, dirilis PHP/FI 2.0. Pada rilis ini, interpreter PHP sudah diimplementasikan dalam program C. Dalam rilis ini disertakan juga modul-modul ekstensi yang meningkatkan kemampuan PHP/FI secara signifikan.

Pada tahun 1997, sebuah perusahaan bernama Zend menulis ulang interpreter PHP menjadi lebih bersih, lebih baik, dan lebih cepat. Kemudian pada Juni 1998, perusahaan tersebut merilis interpreter baru untuk PHP dan meresmikan rilis tersebut sebagai PHP 3.0 dan singkatan PHP diubah menjadi akronim berulang PHP: Hypertext Preprocessing.

Pada pertengahan tahun 1999, Zend merilis interpreter PHP baru dan rilis tersebut dikenal dengan PHP 4.0. PHP 4.0 adalah versi PHP yang paling banyak dipakai pada awal abad ke-21. Versi ini banyak dipakai disebabkan kemampuannya untuk membangun aplikasi web kompleks tetapi tetap memiliki kecepatan dan stabilitas yang tinggi.

Pada Juni 2004, Zend merilis PHP 5.0. Dalam versi ini, inti dari interpreter PHP mengalami perubahan besar. Versi ini juga memasukkan model pemrograman berorientasi objek ke dalam PHP untuk menjawab perkembangan bahasa pemrograman ke arah paradigma berorientasi objek. Server web bawaan ditambahkan pada versi 5.4 untuk mempermudah pengembang menjalankan kode PHP tanpa menginstall software server.

Versi terbaru dan stabil dari bahasa pemrograman PHP saat ini adalah versi 7.0.16 dan 7.1.2 yang resmi dirilis pada tanggal 17 Februari 2017.

D. Fungsi PHP

Bahasa pemrograman PHP merupakan bahasa pemrograman yang kebanyakan digunakan untuk mengembangkan website. Anda dapat membuka artikel kami mengenai [bahasa pemrograman untuk membuat website](#).

Jadi sebenarnya untuk membuat sistem berbasis web tidak harus menggunakan PHP. Namun karena fiturnya yang menarik dan dirasa memudahkan, maka kebanyakan pengembangan menggunakan bahasa pemrograman ini.

Sebelum ada PHP, pengembang kebanyakan menggunakan bahasa pemrograman HTML. Bahasa pemrograman ini sudah dapat membuat tampilan website, akan tetapi tampilan yang dihasilkan masih biasa saja. Biasa saja karena website masih bersifat statis atau tetap, tidak bisa menyesuaikan tampilan dengan kondisi pengguna.

Semenjak adanya PHP maka pembuatan website dapat dikembangkan dan diatur agar menjadi website yang dinamis. Dimana website dapat menyesuaikan tampilan sesuai dengan perangkat yang digunakan oleh pengguna. Tidak hanya itu, website juga dapat melakukan input output dengan database.

PHP sendiri biasanya disisipkan ke dalam kode HTML dan dikombinasikan dengan bahasa pemrograman lain, seperti CSS.

- *Arithmetic operators*

| Contoh | Nama Operator | Hasil |
|--------------|---------------------|---|
| $+\$a$ | Positif | Nilai positif dari $\$a$ <small>Duniaillkom.com</small> |
| $-\$a$ | Negatif | Nilai negatif dari $\$a$ |
| $\$a + \b | Penambahan | Total dari $\$a$ dan $\$b$ |
| $\$a - \b | Pengurangan | Selisih dari $\$a$ dan $\$b$ |
| $\$a * \b | Perkalian | Hasil Kali dari $\$a$ dan $\$b$ |
| $\$a / \b | Div/Pembagian | Hasil Bagi dari $\$a$ dan $\$b$ |
| $\$a \% \b | Mod/Sisa hasil bagi | Sisa dari pembagian $\$a / \b |

- *Bitwise operators*

| Contoh | Nama | Hasil |
|------------------|--------------------|--|
| $\$a \& \b | And | 1 Jika kedua bit bernilai 1. |
| $\$a \b | Or (inclusive or) | 1 jika salah satu bit bernilai 1. |
| $\$a \wedge \b | Xor (exclusive or) | 1 jika salah satu bit bernilai 1, tapi bukan keduanya |
| $\sim \$a$ | Not | Bit 0 menjadi 1, dan bit 1 menjadi 0. |
| $\$a \ll \b | Shift left | Menggeser sebanyak $\$b$ bit ke kiri (setiap 1 kali pergeseran = kelipatan 2) |
| $\$a \gg \b | Shift right | Menggeser sebanyak $\$b$ bit ke kanan (setiap 1 kali pergeseran = bagi 2) <small>Duniaillkom.com</small> |

- *Comparison operators*

| Contoh | Nama Operator | Hasil |
|-------------|------------------------------|---|
| \$a == \$b | Sama dengan | TRUE jika \$a sama dengan \$b (tanpa melihat tipe data) |
| \$a === \$b | Identik dengan | TRUE jika \$a sama dengan \$b, dan memiliki tipe data yang sama |
| \$a != \$b | Tidak sama dengan | TRUE jika \$a tidak sama dengan \$b (tanpa melihat tipe data) |
| \$a <> \$b | Tidak sama dengan | TRUE jika \$a tidak sama dengan \$b (tanpa melihat tipe data) |
| \$a !== \$b | Tidak identik dengan | TRUE jika \$a tidak sama dengan \$b, dan memiliki tipe data yang tidak sama |
| \$a < \$b | Kurang dari | TRUE jika \$a kurang dari \$b |
| \$a > \$b | Lebih dari | TRUE jika \$a lebih dari \$b |
| \$a <= \$b | Kurang dari atau sama dengan | TRUE jika \$a kurang dari atau sama dengan \$b |
| \$a >= \$b | Lebih dari atau sama dengan | TRUE jika \$a lebih dari atau sama dengan \$b Duniaillkom.com |

- *Increment/Decrement operators*

| Contoh | Nama | Hasil |
|--------|----------------|--|
| ++\$a | Pre-increment | Tambah nilai \$a sebanyak 1, lalu kirim nilai \$a |
| \$a++ | Post-increment | Kirim nilai \$a, lalu tambah nilai \$a sebanyak 1 |
| --\$a | Pre-decrement | Kurangi nilai \$a sebanyak 1, lalu kirim nilai \$a |
| \$a-- | Post-decrement | Kirim nilai \$a, lalu kurangi nilai \$a sebanyak 1 |

- *Logical operators*

| Contoh | Nama Operator | Hasil |
|-------------|---------------|---|
| \$a and \$b | AND | TRUE jika \$a dan \$b sama-sama bernilai TRUE . |
| \$a && \$b | AND | TRUE jika \$a dan \$b sama-sama bernilai TRUE . |
| \$a or \$b | OR | TRUE jika salah satu dari \$a atau \$b adalah TRUE . |
| \$a \$b | OR | TRUE jika salah satu dari \$a atau \$b adalah TRUE . |
| \$a xor \$b | XOR | TRUE jika salah satu dari \$a atau \$b adalah TRUE , tetapi bukan keduanya. |
| ! \$a | NOT | TRUE jika \$a= FALSE . Duniaillkom.com |

- *String operators*

Pada *Operators*, menggabungkan variabel dengan menggunakan (.). Contohnya adalah sebagai berikut :

```
$x = "Hello ";
$hasil = $x . "World!";
echo $hasil;
```

- *Combined operators*

| Contoh | Nama Operator | Hasil |
|-----------|--------------------|-----------------------|
| \$b += 10 | Plus-equals | \$b = \$b + 10 |
| \$b -= 10 | Minus-equals | \$b = \$b - 10 |
| \$b /= 10 | Divide-equals | \$b = \$b / 10 |
| \$b *= 10 | Multiply-equals | \$b = \$b * 10 |
| \$b %= 10 | Modulus-equals | \$b = \$b % 10 |
| \$b ^= 10 | Bitwise-XOR-equals | \$b = \$b ^ 10 |
| \$b &= 10 | Bitwise-AND-equals | \$b = \$b & 10 |
| \$b = 10 | Bitwise-OR-equals | \$b = \$b 10 |
| \$b .= | Concatenate-equals | \$b = \$b."dunaiikom" |

PENGENALAN MySQL

A. Pengertian MySQL

MySQL adalah sebuah database management system (manajemen basis data) menggunakan perintah dasar SQL (*Structured Query Language*) yang cukup terkenal. Database management system (DBMS) MySQL multi pengguna dan multi alur ini sudah dipakai [lebih dari 6 juta pengguna di seluruh dunia](#).

MySQL adalah DBMS yang open source dengan dua bentuk lisensi, yaitu Free Software (perangkat lunak bebas) dan Shareware (perangkat lunak berpemilik yang penggunaannya terbatas). Jadi MySQL adalah database server yang gratis dengan lisensi GNU General Public License (GPL) sehingga dapat Anda pakai untuk keperluan pribadi atau komersil tanpa harus membayar lisensi yang ada.

Seperti yang sudah disinggung di atas, MySQL masuk ke dalam jenis RDBMS (Relational Database Management System). Maka dari itu, istilah semacam baris, kolom, tabel, dipakai pada MySQL. Contohnya di dalam MySQL sebuah database terdapat satu atau beberapa tabel.

SQL sendiri merupakan suatu bahasa yang dipakai di dalam pengambilan data pada relational database atau database yang terstruktur. Jadi MySQL adalah database management system yang menggunakan bahasa SQL sebagai bahasa penghubung antara perangkat lunak aplikasi dengan database server.

B. Sejarah MySQL

MySQL adalah pengembangan lanjutan dari proyek UNIREG yang dikerjakan oleh Michael Monty Widenius dan TcX (perusahaan perangkat lunak asal Swedia).

Sayangnya, UNIREG belum terlalu kompatibel dengan database dinamis yang dipakai di website. TcX kemudian mencari alternatif lain dan menemukan perangkat lunak yang dikembangkan oleh David Hughes, yaitu miniSQL atau mSQL. Namun, ditemukan masalah lagi karena mSQL tidak mendukung indexing sehingga belum sesuai dengan kebutuhan TcX.

Pada akhirnya muncul kerjasama antara pengembang UNIREG (Michael Monty Widenius), mSQL (David Hughes), dan TcX. Kerjasama ini bertujuan untuk mengembangkan sistem database yang baru, dan pada 1995 dirilislah MySQL seperti yang dikenal saat ini. Saat ini pengembangan MySQL berada di bawah Oracle.

C. Mengenal SQL

Sesuai dengan pengertian MySQL yang sudah dibahas sebelumnya, bahwa sistem manajemen server ini menggunakan SQL sebagai bahasa pemrogramannya.

SQL merupakan bahasa pemrograman yang perlu Anda pahami karena dapat merelasikan antara beberapa tabel dengan database maupun antar database. Ada tiga bentuk SQL yang perlu Anda ketahui, yaitu *Data Definition Language* (DDL), *Data Manipulation Language* (DML), dan *Data Control Language* (DCL).

| | |
|---------------|---|
| CREATE | Dipakai untuk membuat tabel dan database. |
| DROP | Dipakai untuk menghapus database dan tabel. |
| ALTER | Dipakai untuk mengubah struktur tabel yang sudah ada. Alter dapat mengganti field menggunakan perintah "Change", menambahkan field menggunakan perintah "Add", atau menghapus field menggunakan perintah "drop", dan mengubah namanya menggunakan perintah "Rename". |

1. Data Definition Language (DDL)

DDL berguna pada saat Anda ingin mendefinisikan data di dalam database. Terdapat beberapa query yang dikelompokkan ke dalam DDL, yaitu:

Argumen DDL di atas perlu Anda pahami karena merupakan dasar penggunaan SQL di bagian awal pembuatan database. Contohnya saja jika belum menjalankan perintah "CREATE", Anda belum bisa melanjutkan penggunaan argumen yang lainnya.

2. Data Manipulation Language (DML)

DML dapat Anda pakai setelah menjalankan perintah DDL. DML berfungsi untuk memanipulasi, mengubah, atau mengganti isi dari database (tabel) yang sudah ada.

Terdapat beberapa perintah DML yang perlu Anda ketahui, yaitu:

| | |
|---------------|--|
| INSERT | Dipakai untuk memasukkan data ke dalam tabel pada database. |
| UPDATE | Dipakai untuk mengubah data yang ada di dalam tabel pada database. |
| DELETE | Dipakai untuk menghapus data di dalam tabel pada database. |

3. Data Control Language (DCL)

DCL berguna untuk memberikan hak akses database, mendefinisikan space, mengalokasikan space, dan melakukan audit penggunaan database.

Terdapat beberapa perintah DCL yang perlu Anda ketahui, yaitu:

| | |
|-----------------|---|
| GRANT | Dipakai untuk memberikan izin kepada user untuk mengakses database. |
| REVOKE | Dipakai untuk membatalkan izin user untuk mengakses database. |
| COMMIT | Dipakai untuk menetapkan penyimpanan pada database. |
| ROLLBACK | Dipakai untuk membatalkan penyimpanan pada database. |

D. Cara Menginstall MySQL di Windows dan Linux

MySQL server mendukung berbagai macam sistem operasi (*cross-platform*), khususnya Linux dan Windows. Proses instalasinya pun sangat mudah. Jika menggunakan Windows, Anda bisa menginstall XAMPP untuk menjalankan MySQL server yang di dalamnya sudah terdapat juga modul untuk menjalankan Apache, PHP, FileZilla, dan Tomcat.

Sedangkan di Linux, Anda dapat menginstall MySQL secara terpisah atau menginstall LAMP (Linux, Apache, MySQL, PHP) yang sudah ada modul Apache dan PHP juga. Selain itu, XAMPP juga sudah tersedia di Linux. Jadi Anda bisa menyesuaikan proses instalasi sesuai dengan yang Anda nyaman dan suka.

1. Cara Install MySQL di Windows

Proses instalasi MySQL server menggunakan XAMPP sangat mudah.

- **Langkah 1. Unduh file instalasi XAMPP**
- Anda cukup mengunduh file instalasi XAMPP melalui halaman website apachefriends.org.
- **Langkah 2. Install XAMPP**
- Setelah selesai proses unduh, silakan eksekusi file instalasi XAMPP untuk menjalankan proses instalasi. Kemudian ikuti instruksi yang ada.
- **Langkah 3. Aktifkan MySQL server menggunakan GUI XAMPP**
- Saat Anda selesai menjalankan proses instalasi akan muncul panel kontrol XAMPP. Klik “**Start**” pada modul Apache dan MySQL. Sampai di sini MySQL server sudah dapat berjalan dan sudah dapat dipakai untuk menjalankan perintah SQL.

2. Cara Install MySQL Server di Linux

Pada panduan cara install MySQL Server di Linux ini akan menggunakan Terminal sebagai contohnya. Anda bisa menggunakan cara ini atau jika ingin lebih mudah bisa langsung mengunduh file instalasi XAMPP dan menginstallnya seperti contoh Windows di atas.

Langkah 1. Install Server Web Apache

Sebelum Anda menjalankan proses instalasi, ada baiknya untuk memperbarui repositorinya terlebih dahulu. Untuk memperbarui repositori pada Linux caranya cukup mudah, Anda tinggal membuka Terminal kemudian mengetikkan baris komentar berikut:

```
$ sudo apt-get update
```

Kemudian jalankan perintah berikut untuk menginstall Apache.

```
$ sudo apt-get install apache2
```

Langkah 2. Mengecek Proses Instalasi Apache

Jika proses instalasi Apache sudah selesai dan berhasil dijalankan, Anda bisa mengeceknya dengan mengakses web browser kemudian menuliskan alamat IP server atau alamat domain.

```
http://<alamat_ip_address_server>
```

Langkah 3. Instalasi MySQL

Proses instalasi MySQL server di Linux cukup menggunakan satu perintah saja.

```
$ sudo apt-get install mysql-server
```


Proses instalasi ini akan meminta Anda untuk memasukkan username dan password untuk akses ke dalam MySQL server.

Langkah 4. Pengecekan Instalasi MySQL

Anda dapat memastikan apakah proses instalasi MySQL sudah berjalan baik atau belum dengan mengetikkan perintah di bawah ini.

```
$ sudo systemctl status mysql
```

Perintah di atas untuk mengecek apakah MySQL sudah berjalan atau belum. Jika belum, Terminal akan menampilkan status 'Error' atau 'Failed'.

Anda dapat mengikuti artikel [Cara Instal LAMP di Linux](#) untuk melihat panduan lengkapnya.

E. Fungsi-Fungsi di MySQL Server

Jika MySQL sudah dapat berjalan dengan baik di server atau perangkat, ada beberapa fungsi yang bisa Anda jalankan menggunakan teks perintah (*command prompt*).

Untuk masuk ke dalam MySQL server buka 'CMD' di Windows dan 'Terminal' di Linux.

F. Login dan Logout MySQL Server

Pengertian MySQL adalah database manajemen server yang cukup aman sehingga mempunyai aturan hak akses yang ketat. Jadi ketika ingin mengakses MySQL, Anda harus menggunakan password yang sudah diatur sebelumnya.

Jika sebelumnya belum mengatur user untuk masuk ke dalam MySQL, Anda dapat menggunakan user root dengan mengetikkan perintah berikut.

```
$ mysql -u root -p
```

Opsi ‘-u’ merupakan tanda jika ingin login menggunakan user dengan menggunakan password yang dinyatakan dengan opsi ‘-p’.

Jika ingin keluar, Anda hanya perlu mengetikkan perintah “**quit**” atau “\q”.

G. Opsi di MySQL Server

MySQL server menyediakan beberapa bantuan yang bisa Anda akses untuk mengubah dan mendokumentasikan server, yaitu dengan mengetikkan perintah “\h” atau “\?” ke dalam koneksi MySQL yang sedang aktif.

Perhatian! Semua koneksi harus diakhiri tanda titik koma (;). Tanda ini sebagai petunjuk bahwa perintah atau query yang dimasukkan sudah selesai dan siap dieksekusi.

| | | |
|---------|------------|---|
| Bantuan | \h atau \? | Dipakai untuk menampilkan opsi bantuan yang tersedia di dalam MySQL server. |
| Clear | \c | Dipakai untuk menghapus atau membatalkan semua perintah yang berjalan pada satu perintah. |
| Connect | \r | Dipakai untuk me-refresh koneksi ke database yang ada di dalam Server Host. |
| Ego | \G | Dipakai untuk menampilkan data di dalam database server secara horizontal |
| Go | \g | Dipakai untuk mengeksekusi perintah yang sudah dimasukkan. |
| Tee | \T | Dipakai untuk mengatur lokasi file untuk perintah yang ingin didokumentasikan. |
| Note | \t | Dipakai untuk mengakhiri perintah \T yang berguna untuk mendokumentasikan semua perintah yang sudah dijalankan. |
| Print | \p | Dipakai untuk menampilkan seluruh perintah yang sudah dijalankan ke layar. |
| Prompt | \R | Dipakai untuk mengubah 'prompt' sesuai dengan keinginan. |
| Source | \. | Dipakai untuk mengeksekusi perintah dari luar yang berbentuk file .sql |
| Use | \u | Dipakai untuk masuk ke dalam database (mirip dengan perintah "cd" di dalam bash script) |

H. Perintah Administrasi di MySQL Server

Pengertian MySQL server lainnya yaitu sebuah database manajemen sistem yang bersifat client/server. Jadi sangat penting untuk mengatur hak akses setiap user. Pengaturan hak akses ini supaya semua user mempunyai wewenang yang berbeda. Hak akses juga dapat mengatur user supaya hanya dapat mengakses database tertentu saja.

Pengaturan hak akses di MySQL tentu akan sangat berguna apalagi karena perangkat lunak ini berjalan di jaringan publik yang kemungkinan semua user bisa mengaksesnya melalui internet.

Sebelum mengatur hak akses, Anda harus membuat terlebih dahulu user MySQL selain 'root'.

PENGENALAN CSS

A. Pengertian CSS

Oke pertama-tama saat mulai belajar CSS kamu tentu harus paham betul pengertiannya. Sebenarnya apa itu CSS? CSS adalah singkatan dari *Cascading Style Sheets*. Jika diterjemahkan ke bahasa Indonesia secara harfiah, CSS berarti “lembar penataan menurun”. Namun secara konteks, *Cascading Style Sheets* adalah kumpulan perintah yang digunakan untuk menjelaskan tampilan sebuah halaman situs web dalam *mark-up language*. *Mark-up language* atau bahasa markah adalah bahasa pemrograman yang biasanya digunakan membuat *website*.

Kalau kamu sudah mengenal HTML, nah itulah salah satu contoh bahasa mereka. HTML atau *Hypertext Mark Up Language* tergolong sebagai bahasa pemrograman yang paling standar dan lazim digunakan dalam [pembuatan halaman web](#). HTML ini istilahnya ada di “balik layar” tampilan halaman *web* yang kamu akses. Tatanan sebuah situs *web* akan berantakan tanpa adanya bahasa pemrograman seperti HTML.

B. Cara Kerja CSS

Untuk cara kerjanya sendiri kamu bisa melihat pada Gambar 1. CSS beroperasi melalui tag `<style>` dengan atribut `class` warna. Dengan adanya CSS pada HTML tersebut maka pengaturan warna teks akan menjadi lebih mudah. Saat kamu ingin mengganti warna teks cukup mengetikkan tag `` tanpa harus menulis ulang perintah. Jadi bisa disimpulkan bahwa CSS akan menghemat waktumu dengan perintah-perintah yang efisien.

Hal ini bisa terjadi karena CSS sendiri dikembangkan untuk bisa mengubah tampilan laman *website* tanpa harus mengganti isi konten. Jika kembali pada perumpamaan manusia dan pakaian di poin sebelumnya, dengan CSS kamu tidak mengubah bentuk manusianya tapi hanya mengganti pakaiannya.

Dengan begitu untuk mengubah dan memprogram ulang tampilan *website* pun bisa dilakukan dalam waktu cepat.

C. Peran CSS

Dari sini kamu bisa menyimpulkan bahwa peran CSS untuk *website* sangatlah penting. Tanpa adanya CSS, tampilan *website* akan membosankan atau bahkan membutuhkan waktu lama untuk *loading*. Bayangkan saja kamu hanya bisa bergantung pada HTML untuk membuat sebuah situs. Bukan cuma tampilan situs akan “hambar” tapi kamu juga butuh waktu lebih lama karena harus berulang kali mengetikkan perintah.

D. Macam-Macam CSS

Saat belajar CSS kamu perlu mengetahui bahwa CSS sendiri dibagi menjadi beberapa macam dengan penggunaan yang berbeda-beda. Hal ini dimaksudkan agar kamu mencoba mempraktikkan teori yang sudah dipelajari tidak mengalami kekeliruan. Setidaknya ada tiga macam CSS yang sering digunakan oleh *web designer*. Berikut penjelasannya.

- Inline Style Sheet

Pertama ada *inline style sheet*. Sederhananya, CSS model ini adalah CSS dengan perintah pemrograman yang letaknya ada pada objek. Misalnya kamu ingin mengubah sebuah tulisan pada laman tertentu di [website](#) milikmu, *inline style sheet* CSS harus menempel pada elemen tulisan tersebut. Kamu cukup menambahkan tag `<style>` saja untuk menerapkan CSS ini. Contoh penerapannya bisa kamu lihat pada Gambar 1.

- External Style Sheet

Selain *inline style sheet* ada *external style sheet*. CSS ini letaknya berbeda dengan laman yang akan diubah. Cara ini lebih praktis daripada *inline style sheet* karena bisa menghemat ruang dan bisa digunakan berulang-ulang untuk laman *web* yang berbeda. Kamu bisa mengenali CSS tipe ini lewat tag <link rel>. Tag ini akan menghubungkan halaman *coding* pada *external style sheet* CSS yang terpisah.

- Embedded Style Sheet

Terakhir ada *embedded style sheet*. CSS model ini sama seperti *inline style sheet*, sama-sama berada pada satu laman *coding*. Maka tidak mengherankan jika *embedded style sheet* terkadang disebut dengan *internal style sheet*. Biasanya CSS ini diapit oleh tag <head> </head> dan diawali dengan tag <style>. *Embedded style sheet* sering digunakan untuk mengatur laman *web* dengan tampilan yang unik. Misalnya dalam satu paragraf tulisan ada kalimat yang berbeda dan hal tersebut terus berulang.

Sampai di bagian ini kamu bisa menyimpulkan bahwa CSS punya andil yang besar dalam pembuatan *website*. Tanpa adanya CSS tampilan halaman *web* akan terlihat membosankan. Dari sisi *web designer* sendiri, CSS akan membantu mereka untuk menghemat waktu karena tidak harus lagi mengetikkan perintah yang sama.

HUBUNGAN CSS, HTML, PHP DAN MySQL

Lalu, apa hubungannya HTML dengan CSS? Keduanya sangat berhubungan erat. CSS adalah kode-kode yang dipakai untuk mendesain sebuah laman HTML. Jika HTML diibaratkan sebagai seorang manusia, maka CSS adalah pakaian yang membuat penampilan menjadi semakin menarik. CSS akan membantu para *web designer* untuk mengubah tampilan teks (baik dari bentuk dan ukuran *font* maupun warnanya), menambahkan gambar, hingga mengubah latar belakang sebuah halaman HTML.

Keberadaan CSS bisa terlihat dengan adanya atribut warna teks. Di sini CSS memberi perintah berupa teks berwarna biru melalui tag `` dengan atribut `class="warna"`. Jadi

setiap tag muncul, teks yang mengikutinya akan berwarna biru. Pun demikian saat kamu ingin mengganti warnanya. Tinggal mengganti CSS pada tag <style> dari “blue” ke warna lain, maka teks yang awalnya berwarna biru akan berubah warna.

Melihat dari pengertian – pengertian di atas dapat disimpulkan bahwa HTML dan CSS satu paket bahasa pemrograman yang digunakan untuk membuat tampilan sebuah website, sedangkan PHP dan MYSQL digunakan untuk membuat halaman website yang dibuat HTML + CSS menjadi dinamis. PHP berfungsi sebagai penghubung HTML dan MYSQL. Peran PHP MYSQL dalam membuat web dinamis adalah MYSQL sebagai penampung datanya seperti konten – konten yang dapat berubah – ubah dalam web dinamis maka mysql lah yang menampung datanya, sedangkan PHP digunakan untuk menghubungkan query – query yang ada dalam mysql baik untuk menampilkan data dalam web atau menyimpan data dalam mysql.

BAB 4

KESIMPULAN DAN SARAN

4.1 Kesimpulan

Kesimpulan Berdasarkan penelitian yang telah dilaksanakan penulis menarik kesimpulan bahwa Sistem

Informasi Unit Kegiatan Mahasiswa POLTEKPOS merupakan sistem yang dibuat untuk mempermudah proses pendaftaran bagi para calon anggota yang ingin mendaftar agar mengetahui informasi apa saja yang ada di poltekpos, Sistem Informasi Unit Kegiatan Mahasiswa POLTEKPOS ini memudahkan pengurus POLTEKPOS dalam mengelola event yang akan diadakan di kampus politeknik pos Indonesia, dan Sistem Informasi Unit Kegiatan Mahasiswa POLTEKPOS ini memudahkan antara senior dan junior saling berkomunikasi mengenai kendala pada materi perkuliahan ataupun pada tugas tugas yang dimana disitu akan adanya sharring antara kakak kelas dan adik kelas.

4.2 Saran

Penulis menyadari bahwa pada Sistem Informasi Unit Kegiatan Mahasiswa POLTEKPOS ini masih memiliki banyak kekurangan, untuk itu apabila kedepannya penelitian ini akan dilanjutkan, penulis akan memberikan saran mengenai bagian-bagian yang sebaiknya ditingkatkan, yaitu Mengembangkan kembali Sistem Informasi Unit Kegiatan Mahasiswa POLTEKPOS, sehingga Sistem Informasi ini lebih mudah lagi untuk digunakan, sehingga tidak menyulitkan user untuk mengakses berbagai fitur yang ada dalam Sistem Informasi ini, Diharapkan menambahkan fitur-fitur untuk pengurus POLTEKPOS yang lain agar semua sistem yang ada di poltekpos sudah terkomputerisasi.

Chapter 1

Pertemuan 1

1.1 Pengajuan Judul

Ini adalah judul
yang telah kami
ajukan, namun
tidak di acc :
Website Rental
Motor

Kami telah mendapatkan solusi mengenai judul, dan
telah di acc :
Website Sistem Informasi Badan Eksekutif
Mahasiswa Politeknik Pos Indonesia

Chapter 2

Pertemuan 2

2.1 Issues #11

Pada *issues #11* Membuat Tampilan Awal Web
Halaman 1.html

```
<!doctype html>  
<title>Website BEM POLTEKPOS</title>  
Badan Eksekutif Mahasiswa Politeknik Pos Indonesia
```

```
style.css Cadding:  
15px;  
}  
.menu{  
width: 100% 30px;  
}  
.menu a{  
text-decoration: none; color:  
#fff;  
}  
.menu li{  
list-style-type: none; float:  
left;  
width: 100px; height:  
55px; background:  
#333;
```

```

padding: 15px; text-
align: center; cursor:
pointer; line-height:
30px;
transition-duration: 0.6s;
}
.menu li:hover {
background: #fb090a;
}
.menu li ul li { position:
relative; 5px;
height: 85px;
top: 10px;
}
.menu ul li span.arrow { border: 4px
solid transparent; border-top: 9px
solid #fff; position: absolute;
top: 114px; left:
219px;
}
.menu li:hover ul { display:
block; animation:drop 0.3s;
position: relative;
}
@Keyframes drop { 0
%{
opacity: 0; left:
250px;
}
50%{
opacity: 1; left:
100px;
}

```

```
100 % { opacity:
1; left: 5px;
}
}
```

2.2 Issues #12

Pada *issues #12* Membuat Menu Home

```
<div class="menu w3-orange">
<ul>
<li><a href="">Home</a></li>
</ul>
</div>
```

2.3 Issues #13

Pada *issues #13* Membuat Menu Profil

```
<div class="menu w3-orange">
<ul>
<li><a href="">Profil</a></li>
</ul>
</div>
```

2.4 Issues #14

Pada *issues #14* Membuat Menu Informasi

```
<div class="menu w3-orange">
<ul>
<li><a href="">Informasi</a></li>
</ul>
</div>
```

2.5 Issues #15

Pada *issues #15* Membuat Menu dropdown Informasi event

```
<div class="menu w3-orange">
<ul>
<li><a href="">Informasi</a></li>
<ul>
<li><a href="">Informasi Event</a></li>
</ul>
</ul>
</div>
```

2.6 Issues #16

Pada *issues #16* Membuat Menu Dropdown Informasi Kampus

```
<div class="menu w3-orange">
<ul>
<li><a href="">Informasi</a></li>
<ul>
<li><a href="">Informasi Kampus</a></li>
</ul>
</ul>
</div>
```

2.7 Issues #17

Pada *issues #17* Membuat Menu Logout Keluar.php

```
<?php Session_start();
Session_destroy();
Header(\location:from_login.php");
?>
```

Index adminn.php

```
<li><a href="keluar.php">Keluar</a></li>
```

2.8 Issues #18

Pada *issues #18* Membuat Menu dropdown

```
.menu.li{  
List-style_type: none;  
Width: 110px;  
Height: 45px; Padding:  
10px; Text-align: center;  
Cursor: pointer; Line-  
height : 20px;  
Transition-duration: 0.6s;  
Margin-left:0px;  
}
```

2.9 Issues #19

Pada *issues #19* Membuat Halaman Informasi Kampus H
Informasi Kampus.html

```
<title>Website BEM POLTEKPOS</title>  
Badan Eksekutif Mahasiswa Politeknik Pos Indonesia  
INFORMASI KAMPUS  
style.css  
.container { width:  
100 %; height:  
1000px;  
}  
.head{ width:  
100 %; height:  
75px;  
padding: 15px;
```


}

2.10 Issues #20

Pada *issues #20* Membuat Halaman Event.

H Event.html

```
<title>Website BEM POLTEKPOS</title>
Badan Eksekutif Mahasiswa Politeknik Pos Indonesia EVENT
style.css
.container { width:
100 %; height:
1000px;
}
.head{ width:
100 %; height:
75px;
padding: 15px;
}
```

Chapter 3

Pertemuan 3

3.1 Issues #21

Pada *issues #21* Ukuran body dari class left konten kurang besar.
Style.css

```
.left-konten{ width:
20%;
min-height: 250px;
background-color: #fff; float:
left;
margin: 10px;
```

Perbaikan

```
.left-konten{ width:
40%;
min-height: 400px;
background-color: #fff; float:
left;
margin: 10px;
}
```

3.2 Issues #22

Pada *issues #22* Gambar logo bem kabinet arga mahardika tidak muncul. Dikare- nakan gambar berada di dalam folder img.

```
<div class="left-konten">
<div></div>
```

Jika sebuah gambar berada di folder berbeda dari file yang sedang dikerjakan, penulisan script pemanggilan gambar harus mengikutsertakan nama folder tempat gambar berada.
Perbaikan

```
<div class="left-konten">
<div></div>
```

3.3 Issues #23

Pada *issues #23* Ukuran gambar logo bem polpos terlalu besar

```
<div class="logobem"></div>
```

Untuk membuat gambar berukuran sesuai dengan yang dibutuhkan, perlu kita atur terlebih dahulu panjang dan lebar gambar tersebut dengan width dan height.
Perbaikan

```
<div class="logobem"></div>Kegiatan POLPOS FESTIFAL</font>
<font face="rockwell" size="2" color="black">Kegiatan DONOR DARAH</font>
```

Untuk memperbesar ukuran font, ubah angka pada value size
Perbaikan

```
<font face="rockwell" size="4" color="black">Kegiatan POLPOS FESTIFAL</font>
<font face="rockwell" size="4" color="black">Kegiatan DONOR DARAH</font>
```

3.6 Issues #26

Pada *issues #26* Tulisan keterangan foto pada pakaian adat berada di samping foto.

```

<font face="rockwell" size="3" color="black">Pakaian Adat Sulawesi</font>


<font face="rockwell
```

Agar tulisan berada dibawah foto terdapat 2 cara. Bisa menggunakan `¡br¿` (enter) tetapi jaraknya tidak bisa di atur. Atau membuat div class baru, jaraknya dapat diatur di css
Perbaikan

-


```


<br><font face="rockwell" size="3" color="black">Pakaian Adat Sulawesi</font>

<br>
<br><font face="rockwell" size="3" color="black">Pakaian Adat Minang ~ Bazar Ko

```

- <div class>

```


<div class="tulisan"><font face="rockwell" size="3" color="black">Pakaian Adat


<div class="tulisan"><font face="rockwell" size="3" color="black">Pakaian AdatMi

```

Style.css

```

.tulisan{ width:
300px;
margin-bottom: 50px;
margin-left: 25px;
}

```

3.7 Issues #27

Pada *issues #27* Gambar kegiatan donor darah tidak berjajar ke samping. Tetapi ke bawah. Pengaturan letak dan ukuran gambar menggunakan css.

```

.gambar {
margin-top: 20px;
margin-left: 35px;
}

```

Agar gambar dapat berjajar kesamping digunakan float: left;
Perbaikan

```

.gambar { margin-
top: 20px; margin-left:
35px;

```

```
float: left;
}
```

3.8 Issues #28

Pada *issues #28* Ukuran gambar logo twitter tidak sinkron

```

```

Karena lebar dan panjang gambar tidak sama, maka ukuran gambar harus disesuaikan.
Perbaikan

```

```

3.9 Issues #29

Pada *issues #29* Gambar logo medsos bem terlalu dekat dengan gambar kegiatan polpos festival dan kegiatan donor darah

```
<div class="right-konten">
```

```
<center>
```

```
<div class="tulisan">
```

```
</div>
```

```
<div class="gambar">
```

```

```

```

```

```
<div class="tulisan"><font face="rockwell" size="3" color="black">BEM danKSR</fo
```

```

```

```
</div>
```

```

```

```
<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>
```

```


<font face="rockwell" size="2" color="purple">bempoltekpos</font>


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>
</div>

```

Untuk membuat gambar terpisah dan memiliki jarak yang diinginkan, maka dibuat class baru yaitu div class medsos. Dan untuk class kegiatan di tutup dengan `</div>` pada bagian sebelum script gambar medsos .

Perbaikan

```

<div class="right-konten">
<center>
<div class="tulisan">

</div>
<div class="gambar">


<div class="tulisan"><font face="rockwell" size="3" color="black">BEM danKSR</fo

</div>
</center>
</div>
<div class="medsos">

<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>

```

```


<font face="rockwell" size="2" color="purple">bempoltekpos</font>


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>
</div>

```

3.10 Issues #30

Pada *issues #30* Gambar media sosial tidak berada di tengah / center

```

<div class="medsos">

<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>


<font face="rockwell" size="2" color="purple">bempoltekpos</font>


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>
</div>

```

Untuk membuat gambar berada di tengah, menggunakan center.
Perbaikan


```
<div class="medsos">
<center>

<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>


<font face="rockwell" size="2" color="purple">bempoltekpos</font>


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>

</center>
</div>
```

Chapter 4

Pertemuan 4

4.1 Issues #31

Pada *issues #31* Menambahkan menu beasiswa

```
<div class="menu w3-orange">  
<ul>  
<li><a href="">beasiswa</a></li>  
</ul>  
</div>
```

4.2 Issues #32

Pada *issues #32* Menambahkan menu Internship

```
<div class="menu w3-orange">  
<ul>  
<li><a href="">Internship</a></li>  
</ul>  
</div>
```

4.3 Issues #33

Pada *issues #33* Menambahkan menu Event hiburan

```
<div class="menu w3-orange">
```

```
<ul>
<li><a href="">Event Hiburan</a></li>
</ul>
</div>
```

4.4 Issues #34

Pada *issues #34* Menambahkan menu jurusan

```
    <div class="menu w3-orange">
<ul>
<li><a href="">Jurusan</a></li>
</ul>
</div>
```

4.5 Issues #35

Pada *issues #35* Menambahkan menu lomba akademik

```
    <div class="menu w3-orange">
<ul>
<li><a href="">Lomba Akademik</a></li>
</ul>
</div>
```

4.6 Issues #36

Pada *issues #36* Menambahkan menu minat dan bakat.

```
    <div class="menu w3-orange">
<ul>
<li><a href="">Minat dan Bakat</a></li>
</ul>
</div>
```

4.7 Issues #37

Pada *issues #37* Menambahkan menu lomba olahraga.

```
<div class="menu w3-orange">
<ul>
<li><a href="">Lomba Olahraga</a></li>
</ul>
</div>
```

4.8 Issues #38

Pada *issues #38* Menambahkan menu komunitas.

```
<div class="menu w3-orange">
<ul>
<li><a href="">Komunitas</a></li>
</ul>
</div>
```

4.9 Issues #39

Pada *issues #39* Menambahkan menu lomba seni.

```
<div class="menu w3-orange">
<ul>
<li><a href="">Lomba Seni</a></li>
</ul>
</div>
```

4.10 Issues #40

Pada *issues #40* Menambahkan menu pengajuan profosal.

```
<div class="menu w3-orange">
<ul>
<li><a href="">Pengajuan Profosal</a></li>
</ul>
</div>
```

Chapter 5

Pertemuan 5

5.1 Issues #41

Pada *issues #41*

Notice: Undefined index: id_kosan in C:\xampp\htdocs\webbempolpos\tkosan\edi

Edit.php

```
<?php
include 'fungsi.php';
$id_kosan= $_GET['id_kosan'];
$data = mysqli_query($conn,"select * from kosan where id_kosan='$id_kosan'"); while($d =
mysqli_fetch_assoc($data)){
?>
```

Dengan menambahkan script error reporting(E_ALL (E_NOTICE — E_WARN- ING)); yang diletakkan pada baris sebelum baris yang error

```
<?php
include 'fungsi.php';
error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
$id_kosan= $_GET['id_kosan'];
$data = mysqli_query($conn,"select * from kosan where id_kosan='$id_kosan'"); while($d =
mysqli_fetch_assoc($data)){
?>
```

5.2 Issues #42

Pada *issues #42* Login gagal, dikarenakan tidak ada perintah button Form-login.php

```
<input class="btn" type="submit" name="masuk" value="Sign In">
```

Seharusnya menggunakan button

```
<button class="btn" type="submit" name="masuk" value="Sign In">SignIn</button>
```

5.3 Issues #43

Pada *issues #43* Atribut foto belum muncul

```
<tr>
<td><?php echo $no++; ?></td>
<td><?php echo $d['nama'];?></td>
<td><?php echo $d['ukuran'];?></td>
<td><?php echo $d['vasilitas'];?></td>
<td><?php echo $d['harga'];?></td>
<td><?php echo $d['alamat'];?></td>
<td><?php echo $d['contact'];?></td>
<td><?php echo $d['foto'];?></td>
<td>
<button><a href="edit.php?id_kosan=<?php echo $d['id_kosan'];?>">EDIT</a></button>
<button><a href="hapus.php?id_kosan=<?php echo $d['id_kosan'];?>">HAPUS</a></button>
</td>
</tr>
```

menambahkan script img dengan menyertakan folder/lokasi gambar, dan nama atributnya di database.

```
<tr>
<td><?php echo $no++; ?></td>
<td></td>
<td><?php echo $d['nama'];?></td>
<td><?php echo $d['ukuran'];?></td>
```

```

<td><?php echo $d['vasilitas'];?></td>
<td><?php echo $d['harga'];?></td>
<td><?php echo $d['alamat'];?></td>
<td><?php echo $d['contact'];?></td>
<td><?php echo $d['foto'];?></td>
<td>
<button><a href="edit.php?id_kosan=<?php echo $d['id_kosan'];?>">EDIT</a></butto
<button><a href="hapus.php?id_kosan=<?php echo $d['id_kosan'];?>">HAPUS</a></but
</td>
</tr>

```

5.4 Issues #44

Pada *issues #44* menu hapus dan delete tidak berada di tabel tampilan kolom aksi.

```

<tr>
<td><?php echo $no++; ?></td>
<td></td>
<td><?php echo $d['nama'];?></td>
<td><?php echo $d['ukuran'];?></td>
<td><?php echo $d['vasilitas'];?></td>
<td><?php echo $d['harga'];?></td>
<td><?php echo $d['alamat'];?></td>
<td><?php echo $d['contact'];?></td>
<td><?php echo $d['foto'];?></td>
<td>
<button><a href="edit.php?id_kosan=<?php echo $d['id_kosan'];?>">EDIT</a></butto
<button><a href="hapus.php?id_kosan=<?php echo $d['id_kosan'];?>">HAPUS</a></but
</td>
</tr>

```

Menghapus kolom foto, karena sudah ada .

```

<td>
<tr>
<td><?php echo $no++; ?></td>
<td></td>

```



```

<td><?php echo $d['nama'];?></td>
<td><?php echo $d['ukuran'];?></td>
<td><?php echo $d['vasilitas'];?></td>
<td><?php echo $d['harga'];?></td>
<td><?php echo $d['alamat'];?></td>
<td><?php echo $d['contact'];?></td>
<td>
<button><a href="edit.php?id_kosan=<?php echo $d['id_kosan'];?>">EDIT</a></button>
<button><a href="hapus.php?id_kosan=<?php echo $d['id_kosan'];?>">HAPUS</a></button>
</td>
</tr>

```

5.5 Issues #45

Pada *issues #45* Typo pada file Update php.

```
$contacy=$_POST['contacy'];
```

yang benar adalah

```
$contact=$_POST['contact'];
```

5.6 Issues #46

Pada *issues #46* Primary key tabel kosan salah pada file menuuu.php

```

<td>
<button><a href="edit.php?idpro=<?php echo $d['idpro']; ?>">EDIT</a></button>
<button><a href="hapus.php?idpro=<?php echo $d['idpro']; ?>">HAPUS</a></button>
</td>

```

Primary key tabel kosan adalah id kosan

```

<td>
<button><a href="edit.php?id_kosan=<?php echo $d['id_kosan'];?>">EDIT</a></button>
<button><a href="hapus.php?id_kosan=<?php echo $d['id_kosan'];?>">HAPUS</a></button>
</td>

```

5.7 Issues #47

Pada *issues #47* Nama tabel salah pada file update.php pada folder kosan

```
mysqli_query($conn,"update produk set nama='$nama', ukuran='$ukuran',vasilitas='
```

Tabel yang akan diupdate adalah tabel kosan

```
mysqli_query($conn,"update kosan set nama='$nama', ukuran='$ukuran',vasilita
```

5.8 Issues #48

Pada *issues #48* Saat salah satu data di edit/update data yang lain ikut berubah mengikuti data yang diubah

Update.php

```
mysqli_query($conn,"update kosan set nama='$nama', ukuran='$ukuran',vasilita
```

Cara satu masih error edit.php (penggunaan mysql dan mysqli itu berbeda. Mysqli adalah perbaikan dan penambahan dari extension mysql. Seperti penulisan argument resource (varibel yg dibuat), jika mysql diletakkan di akhir. Sedangkan mysqli di awal. Dan selebihnya penggunaan mysqli menambahkan huruf I seperti mysqli-connect(), jika mysql hanya mysql-connect())

```
<?php
include 'fungsi.php';
error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
$id_kosan= $_GET['id_kosan'];
$query_mysql = mysql_query("select * from kosan where id_kosan ='$id_kosan'")or
$nomor = 1;
while($data = mysql_fetch_array($query_mysql)){
?>
```

Cara 2 masih error edit.php

```
<?php
include 'fungsi.php';
error_reporting(E_ALL ^ (E_NOTICE | E_WARNING));
$id_kosan= $_GET['id_kosan'];
$data = mysqli_query($conn,"select * from kosan where id_kosan ='$id_kosan'");
```

```
while($d = mysqli_fetch_array($data)){
?>
```

Update.php

```
mysqli_query($conn,"update kosan set nama='$nama', ukuran='$ukuran',vasilitas='$
```

Cara tiga berhasil. Mengeluarkan Id-kosan/primary key tabel kosan dari database ke html dan menambahkan where beserta primary key tabel kosan di fileupdate.

Edit.php

```
<tr>
<td>Nama</td>
<td> :
<input type="text" name="nama" value="<?php echo $d['nama']; ?>"><br>
<input type="hidden" name="id_kosan" value="<?php echo $d['id_kosan']; ?>"></td>
</tr>
```

Update.php

```
mysqli_query($conn,"update kosan set nama='$nama', ukuran='$ukuran',vasilitas='$
```

5.9 Issues #49

Pada *issues #49* Foto tidak tampil, pada file menuuu.php

```
<td></td>
<td></td>
```

5.10 Issues #50

Pada *issues #50*

Parse error: syntax error, unexpected

```
'$query' (T_VARIABLE) in C:\xampp\htdocs\webbempolpos\t_kosan\fungsi.php on l
```

Kurang tanda ,

```
$query = "INSERT INTO kosan VALUES('','$nama','$ukuran','$vasilitas','$harga
```

yang benar adalah :

- \$query = "INSERT INTO kosan VALUES (' ' , ' \$nama' , ' \$ukuran' , ' \$vasilitas

Masih error

Kurang tanda ; pada variable foto

```
}
```

```
function tambah($data){  
    global $conn;  
    $nama = $data["nama"];  
    $ukuran = $data["ukuran"];  
    $vasilitas = $data["vasilitas"];  
    $harga = $data["harga"];  
    $alamat = $data["alamat"];  
    $contact = $data["contact"];  
    $foto = $data["foto"]} function
```

```
tambah($data){  
    global $conn;  
    $nama = $data["nama"];  
    $ukuran = $data["ukuran"];  
    $vasilitas = $data["vasilitas"];  
    $harga = $data["harga"];  
    $alamat = $data["alamat"];  
    $contact = $data["contact"];  
    $foto = $data["foto"];
```

```
$query = "INSERT INTO kosan VALUES('','$nama','$ukuran','$vasilitas','$harga  
mysqli_query($conn, $query);  
return mysqli_affected_rows($conn);
```

```
}
```

```
$query = "INSERT INTO kosan VALUES('','$nama','$ukuran','$vasilitas','$harga  
mysqli_query($conn, $query);  
return mysqli_affected_rows($conn);
```

}

Chapter 6

Pertemuan 6

6.1 Issues #51

Pada *issues #51* Ukuran body dari class left-konten kurang besar

Style.css

```
.left-konten{ width:
20%;
min-height: 250px;
background-color: #fff; float:
left;
margin: 10px;
}
```

Ini adalah code program yang telah kami ubah untuk membuat left-konten besar

```
.left-konten{ width:
40%;
min-height: 400px;
background-color: #fff; float:
left;
margin: 10px;
}
```

6.2 Issues #52

Pada *issues #52* Gambar logo bem kabinet arga mahardika tidak muncul. Dikarenakan gambar berada di dalam folder img.

```
<div class="left-konten">
<div></div>
```

Ini adalah code program yang telah kami buat, dan gambar logo telah berhasil muncul

```
<div class="left-konten">
<div></div>
```

sebuah gambar berada di folder berbeda dari file yang sedang dikerjakan, penulisan script pemanggilan gambar harus mengikutsertakan nama folder tempat gambar berada. Jika

6.3 Issues #53

Pada *issues #53* Ukuran gambar logo bem polpos terlalu besar.

```
<div class="logobem"></div>
```

Ini adalah Code Program yang telah kita buat, dan telah berhasil dan sesuai

```
<div class="logobem"></div>Kegiatan POLPOS FESTIFAL</font>
<font face="rockwell" size="2" color="black">Kegiatan DONOR DARAH</font>
```

Ini adalah Code program yang telah kami ubah / perbaiki. Dan tulisannya sudah sesuai.

```
<font face="rockwell" size="4" color="black">Kegiatan POLPOS FESTIFAL</font>
<font face="rockwell" size="4" color="black">Kegiatan DONOR DARAH</font>
```

Untuk memperbesar ukuran font, ubah angka pada value size

6.6 Issues #56

Pada *issues #56* Tulisan keterangan foto pada pakaian adat berada di samping foto.

```

<font face="rockwell" size="3" color="black">Pakaian Adat Sulawesi</font>
```

```

<font face="rockwell" size="3" color="black">Pakaian Adat Minang ~ BazarKomunita
```

Agar tulisan berada dibawah foto terdapat 2 cara. Bisa menggunakan `br` (enter) tetapi jaraknya tidak bisa di atur. Atau membuat div class baru, jaraknya dapat diatur di css Dan ini adalah kodingan yang telah kami buat, dan telah sesuai dengan penempatannya


```

<br><font face="rockwell" size="3" color="black">Pakaian Adat Sulawesi</font>
```

```
<br>
<br><font face="rockwell" size="3" color="black">Pakaian Adat Minang ~ Bazar Ko
```

Ini adalah Code program yang telah kami ubah / perbaiki. Dan tulisannya sudah sesuai.

```
• <div class>

<div class="tulisan"><font face="rockwell" size="3" color="black">Pakaian Adat


<div class="tulisan"><font face="rockwell" size="3" color="black">Pakaian AdatMi
```

Style.css

```
.tulisan{ width:
300px;
margin-bottom: 50px;
margin-left: 25px;
}
```

6.7 Issues #57

Pada *issues #57* Gambar kegiatan donor darah tidak berjajar ke samping. Tetapi ke bawah Pengaturan letak dan ukuran gambar menggunakan css.

```
.gambar { margin-
top: 20px; margin-left:
35px;
}
```

Agar gambar dapat berjajar kesamping digunakan float: left; Ini adalah Code Program yang telah kita buat, dan telah sesuai dengan penempatannya.

```
.gambar {
margin-top: 20px;
```

```
margin-left: 35px; float: left;
}
```

6.8 Issues #58

Pada *issues #58* Ukuran gambar logo twitter tidak sinkron.

```

```

Karena lebar dan panjang gambar tidak sama, maka ukuran gambar harus disesuaikan. Ini adalah Code Program yang telah kami perbaiki, dan hasilnya Panjang dan lebar sudah sama.

```

```

6.9 Issues #59

Pada *issues #59* Gambar logo medsos bem terlalu dekat dengan gambar kegiatan polpos festival dan kegiatan donor darah.

```
<div class="right-konten">
```

```
<center>
```

```
<div class="tulisan">
```

```
</div>
```

```
<div class="gambar">
```

```

```

```

```

```
<div class="tulisan"><font face="rockwell" size="3" color="black">BEM danKSR</fo
```

```

```

```
</div>
```

```

```

```
<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>
```

```


<font face="rockwell" size="2" color="purple">bempoltekpos</font>


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>
</div>

```

Ini adalah code Program yang telah kami buat untuk membuat jarak antara kegiatan polpos dan kegiatan donor darah.

```

<div class="right-konten">
<center>
<div class="tulisan">

</div>
<div class="gambar">


<div class="tulisan"><font face="rockwell" size="3" color="black">BEM danKSR</fo

</div>
</center>
</div>
<div class="medsos">

<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>


<font face="rockwell" size="2" color="purple">bempoltekpos</font>

```

```


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>
</div>

```

6.10 Issues #60

Pada *issues #60* Gambar medsos tidak berada di tengah

```

<div class="medsos">

<font face="rockwell" size="2" color="purple">BEM Politeknik PosIndonesia</font>


<font face="rockwell" size="2" color="purple">bempoltekpos</font>


<font face="rockwell" size="2" color="purple">@wlu2012n</font>


<font face="rockwell" size="2" color="purple">@bempoltekpos</font>


<font face="rockwell" size="2" color="purple">BEM poltekpos</font>
</div>

```

Dan ini adalah Code Program yang telah kami ubah untuk membuat gambar media social berada di tengah / center.

```

<div class="medsos">
<center>


```

BEM Politeknik PosIndonesia

bempoltekpos

@wlu2012n

@bempoltekpos

BEM poltekpos

</center>

</div>

Chapter 7

Pertemuan 7

7.1 Issues #61

Pada *issues #61* Menu tidak tampil saat diinputkan secara otomatis. Tetapi tampil saat diinput manual di db.

```
Parse error: syntax error, unexpected '<' in C:\xampp\htdocs\webbempolpos\t_k
<?php $ambil = $koneksi->query("SELECT * FROM kosan");?>
<?php while($perproduk = $ambil->fetch_assoc()) { ?>
```

7.2 Issues #62

Pada *issues #62*

Notice: Undefined index: menu in C:\xampp\htdocs\webbempolpos\t_catering\fungsi.p

```
<tr>
    <td>Menu</td>
    <td> : <input name="vasilitas" id="ukuran" placeholder="TIDAK BOLEH
</tr>
```

```
<tr>
    <td>Menu</td>
    <td> : <input name="vasilitas" id="menu" placeholder="TIDAK BOLEHKOS
</tr>
```

Perbaikan tambah.php Tidak ada type atau type data dari menu, sehingga mem- buat menu tidak terbaca.

```
<tr>
    <td>Menu</td>
    <td> : <input name="menu" id="menu" placeholder="TIDAK BOLEH
    KOSONG!
</tr>
<tr>
    <td>Menu</td>
    <td> : <input type="text" name="menu" id="menu" placeholder="TIDAKBO
</tr>
```

7.3 Issues #63

Pada *issues #63* The requested URL was not found on this server. The link on the referring page seems to be wrong or outdated. Please inform the author of that page about the error. If you think this is a server error, please contact the webmaster. Error 404 localhost Apache/2.4.10 (Win32) OpenSSL/1.0.1i PHP/5.6.3

Tambah.php

```
<button><a href="info_catering.php">Kembali</a></button></td>
```

Salah halaman

```
<button><a href="info_kantin.php">Kembali</a></button></td>
```

7.4 Issues #64

Pada *issues #64*

Notice: Undefined index: contact in C:\xampp\htdocs\webbempolpos\t_laundry\fungsi

Kurang tabel contact pada tambah.php

```
<tr>
    <td>Contact</td>
    <td>: <input type="text" name="contact" id="contact"           placeholder="TI
</tr>
```

\section{Issues \#64}

7.5 Issues #65

Pada *issues #65* Tabel edit pada halaman edit.php tidak muncul Kesalahan pemangilan query tabel

```
$id_la= $_GET['id_la'];  
  
$id_LO= $_GET['id_LO'];
```

7.6 Issues #66

6. Parse error: syntax error, unexpected end of file in C:\xampp\htdocs\webbempol Belum ada penutup untuk php session

```
<?php } ?>
```

7.7 Issues #67

Pada *issues #67* Object not found! The requested URL was not found on this server. The link on the referring page seems to be wrong or outdated. Please inform the author of that page about the error. If you think this is a server error, please contact the webmaster.

Salah pengalamatn folder

```
<?php  
session_start();  
if(!isset($_SESSION['npm'])){  
echo "Anda Belum Masuk! Silahkan <a href='form_login.php'> masuk dulu!</a>";  
}else {  
?>
```

Tambah ../ sebelum alamat link yang dituju

```
<?php  
session_start();  
if(!isset($_SESSION['npm'])){  
echo "Anda Belum Masuk! Silahkan <a href='../form_login.php'> masuk dulu!</a>";  
}else {  
?>
```


7.8 Issues #68

Pada *issues #68*

Fatal error: Call to undefined function query() in C:\xampp\htdocs\webbempolp

Tidak ada pemanggilan fungsi sebagai koneksi ke database

```
<?php
$conn = mysqli_connect("localhost","root","","webbempolpos");
if(isset($_POST["OK"])){
```

Pemanggilan fungsi dari file bernama fungsi.php

```
<?php
require 'fungsi.php';
$conn = mysqli_connect("localhost","root","","webbempolpos");
if(isset($_POST["OK"])){
```

7.9 Issues #69

Pada *issues #69* Tidak dapat menghapus info pada info hiburan. Salah penulisan variable dari fungsi.php

```
mysqli_query($con,"delete from hiburan where id_hiburan='$id_hiburan'");
```

Perbaikan

```
mysqli_query($conn,"delete from hiburan where id_hiburan='$id_hiburan'");
```

7.10 Issues #70

Pada *issues #70* Salah pemanggilan nama atribut pada table hiburan.

```
$hari_tanggal= $data["hari_tanggal"];
```

Harus menyesuaikan nama atribut pada table

```
$hari_tgl= $data["hari_tgl"];
```

Chapter 8

Pertemuan 8

8.1 Issues #71

Pada *issues #71* Tambah info tidak tersimpan dan tidak dapat mereset teks. Tidak ada metode dan enctype

```
form action="" method="post" name="form_input" enctype="multipart/form-data">
```

8.2 Issues #72

Pada *issues #72* Tidak dapat menghapus info pada info kantin. Salah pemanggilan id/query table kantin.

```
$id_kosan= $_GET['id_kosan'];
```

Perbaiki

```
$id_kantin= $_GET['id_kantin'];
```

8.3 Issues #73

Pada *issues #73* Atribut info tidak sesuai dengan table.

```
<td><?php echo $no++; ?></td>  
<td></td>  
<td><?php echo $d['nama'];?></td>  
<td><?php echo $d['jenis_jaitan'];?></td>  
<td><?php echo $d['open_close'];?></td>  
<td><?php echo $d['contact'];?></td>
```

Kurang satu atribut

```
<td><?php echo $no++; ?></td>
<td></td>
<td><?php echo $d['nama'];?></td>
<td><?php echo $d['jenis_jaitan'];?></td>
<td><?php echo $d['open_close'];?></td>
<td><?php echo $d['alamat'];?></td>
<td><?php echo $d['contact'];?></td>
```

8.4 Issues #74

Pada *issues #74* Interface halaman minat dan bakat tidak sesuai. Salah pemanggilan div class dari css.

```
<div class="layar">
```

Perbaiki

```
<div class="layar-1">
```

8.5 Issues #75

Pada *issues #75* 15. Tidak bisa kembali ke halaman awal pada button di view kosan. Salah penamaan link halaman.

```
<button><a href=" ../home.html">Home</a></button>
```

Perbaiki

```
<button><a href=" ../Halaman_1.html">Home</a></button>
```

8.6 Issues #76

Pada *issues #76* Tidak bisa edit pada info laundry. Typo, kurang tanda ; pada file update.

```
$nama = $_POST['nama']
```

Perbaiki

```
$nama = $_POST['nama'];
```

8.7 Issues #77

Pada *issues #77* Menu yang ada di halaman info komunitas, tidak bisa membuka halaman yang dituju. Belum ada link dari halaman masing masing.

```
<li><a href="H_Event.html">Event</a>
<ul>
<li><a href=" ">Hiburan</li></a>
<li><a href="">Lomba</li></a>
<ul>
<li><a href="">Akademik</a>
<li><a href="">Olahraga</a>
<li><a href=". ">Seni</a>
</li>
</ul>
</ul>
</ul>
</li>
```

Memberi link halaman

```
<li><a href="H_Event.html">Event</a>
<ul>
<li><a href=" ../t_hiburan/view_hiburan.php">Hiburan</li></a>
<li><a href="">Lomba</li></a>
<ul>
<li><a href=" ../t_lakademik/view_la.php">Akademik</a>
<li><a href=" ../t_lolahraga/view_lo.php">Olahraga</a>
<li><a href=" ../t_lseni/view_seni.php">Seni</a>
</li>
</ul>
</ul>
</ul>
</li>
```

8.8 Issues #78

Pada *issues #78* Tulisan kalimat pada info komunitas bergabung dengan foto. Tidak ada perintah enter

```

<center>

<font face="forte" size="3" color="black">
- Sunda Polpos -
Perkumpulan akang dan teteh pribumi, tetapi komunitas ini juga
menerima anggotad
</font>
</center>

```

Memberikan perintah `
` sebagai perintah enter.

```

<center>
<br><br>
<font face="forte" size="3" color="black">
- Sunda Polpos - <br>
Perkumpulan akang dan teteh pribumi, tetapi komunitas ini juga
menerima anggotad
</font>
</center>

```

8.9 Issues #79

Pada *issues #79* Mengubah layar div class pada halaman minat dan bakat.

```
<div class="layar-2">
```

Perbaikan Menjadi warna merah

```
<div class="layar-2 w3-red">
```

8.10 Issues #80

Pada *issues #80* Sub judul tidak rata tengah

```
<font face="rockwell" size="6" color="black"> VISI DAN
MISI</font>
```

Untuk membuat rata tengah, menggunakan perintah ;center;

```
<font face="rockwell" size="6" color="black"><center>VISI DAN  
MISI</center></font>
```

DAFTAR PUSTAKA

1. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
2. HM., Yogyianto, Analisis dan Disain Sistem Informasi : Pendekatan Terstruktur, Penerbit Andi Offset, Yogyakarta, 1995.
3. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
4. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
5. HM., Yogyianto, Analisis dan Disain Sistem Informasi : Pendekatan Terstruktur, Penerbit Andi Offset, Yogyakarta, 1995.
6. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
7. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
8. HM., Yogyianto, Analisis dan Disain Sistem Informasi : Pendekatan Terstruktur, Penerbit Andi Offset, Yogyakarta, 1995.
9. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
10. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
11. HM., Yogyianto, Analisis dan Disain Sistem Informasi : Pendekatan Terstruktur, Penerbit Andi Offset, Yogyakarta, 1995.
12. Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.
13. McLeod, Raymond, Management Information System, 7th ed., Prentice Hall, New Jersey, 1998.
14. McNurlin, Barbara C.; Sparague, Ralph H Jr., Information Systems Management in Practice, 4th ed., Prentice Hall, New Jersey, 1998.

15. .Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.

16. HM., Yogyanto, Analisis dan Disain Sistem Informasi : Pendekatan Terstruktur, Penerbit Andi Offset, Yogyakarta, 1995.

17 .Martin, Merle P., Analysis And Design of Business Information Sytems, Mac Millan Publishing Company, New York, 1991.

18. McLeod, Raymond, Management Information System, 7th ed., Prentice Hall, New Jersey, 1998.

19. McNurlin, Barbara C.; Sparague, Ralph H Jr., Information Systems Management in Practice, 4th ed., Prentice Hall, New Jersey, 1998.