

ILKA ZUFRIA, M.KOM

ANALISIS DAN PERANCANGAN SISTEM INFORMASI



Editor:
SUENDRI, M.Kom

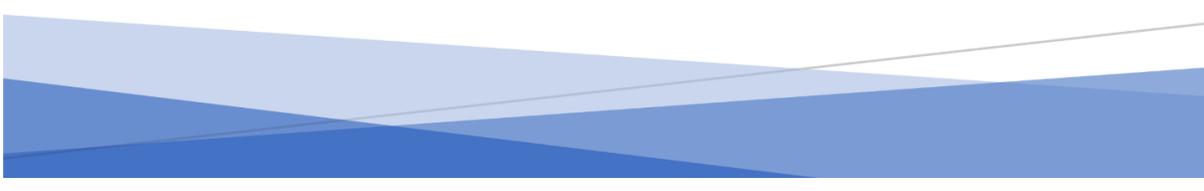


**ANALISIS DAN PERANCANGAN
SISTEM INFORMASI**

ILKA ZUFRIA, M.KOM

EDITOR.

SUENDRI, M.KOM



ANALISIS DAN PERANCANGAN SISTEM INFORMASI

ILKA ZUFRIA, M.KOM

EDITOR.

SUENDRI, M.KOM



CV. Puskira Mitra Jaya
Jln. Willièm Iskandar Muda No – 2K/22 Medan
Tlpon. (061) 8008-8209 (0813-6106-0465)
Email: cv.pusdikramitrajaya@gmail.com

**Sanksi Pelanggaran Pasal 113 Undang - Undang Nomor 28 Tahun 2014 Tentang Hak Cipta
Sebagaimana Yang Diatur Dan Diubah Dari Undang - Undang Nomor 19 Tahun 2002
Bahwa: Kutipan Pasal 113**

1. Setiap orang yang dengan tanpa hak melakukan pelanggaran hak ekonomi sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf i untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 1 (satu) tahun dan/atau pidana denda paling banyak **Rp. 100.000.000 (Seratus Juta Rupiah)**.
2. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf c, huruf d, huruf f, dan/atau huruf h untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 3 (tiga) tahun dan/atau pidana denda paling banyak **Rp. 500.000.000,00 (Lima Ratus Juta Rupiah)**.
3. Setiap Orang yang dengan tanpa hak dan/atau tanpa izin Pencipta atau pemegang Hak Cipta melakukan pelanggaran hak ekonomi Pencipta sebagaimana dimaksud dalam Pasal 9 ayat (1) huruf a, huruf b, huruf e, dan/atau huruf g untuk Penggunaan Secara Komersial dipidana dengan pidana penjara paling lama 4 (empat) tahun dan/atau pidana denda paling banyak **Rp. 1.000.000.000,00 (Satu Miliar Rupiah)**.
4. Setiap Orang yang memenuhi unsur sebagaimana dimaksud pada ayat (3) yang dilakukan dalam bentuk pembajakan, dipidana dengan pidana penjara paling lama 10 (sepuluh) tahun dan/atau pidana denda paling banyak Rp. 4.000.000.000,00 (Empat Miliar Rupiah).

Analisis Dan Perancangan Sistem Informasi

Cet. 1. – Medan CV. Pusedikra Mitra Jaya, 2022

I. 138 hlm, 25 Cm.

Bibliografi. 139

Hak Cipta Pada, Penulis Dan Penerbit

Januari 2022

Karya

Ilka Zufria, M.Kom

Editor

Suendri,, M.Kom

Diterbitkan Oleh:

CV. Pusedikra Mitra Jaya

Jln. Williem Iskandar Muda No – 2K/22 Medan

Tlpn. (061) 8008 - 8209 (0813-6106-0465)

Email: cv.pusedikramitrajaya@gmail.com

Anggota IKAPI (Ikatan Penerbit Buku Indonesia)

IKAPI. No. 049/SUT/2020

Dicetak Oleh CV.Pusedikra Mitra Jaya.

PMJ. NO. 07/ B.1/PMJ/ ISBN// 2022

Copyright © 2022 - CV. Pusedikra Mitra Jaya



Cetakan Pertama Januari 2022

Hak Cipta Dilindungi Undang – Undang. Dilarang Mengutip Sebagian Atau Seluruh Atau Seluruh Isi Buku Ini Dengan Cara Apapun, Termasuk Dengan Cara. Penggunaan Mesin Foto Copi, Tanpa Izin Sah Dari Penerbit.

ISBN: 9786236853948

KATA PENGANTAR



الحمد لله رب العالمين. والصلاة والسلام على سيدنا محمد وعلى اله وأصحابه اجمعين

Dengan segala kerendahan hati, penulis sampaikan puji syukur kepada Allah SWT berkat rahmat dan karuniaNya penulis dapat menuliskan buku Analisis dan Perancangan Sistem Informasi

Buku ini adalah buku yang membahas topik matakuliah inti di beberapa program studi bidang informatika diantaranya Sistem Informasi, Teknologi Informasi maupun Teknik Informatika. Dengan harapan buku ini dapat menjembatani mahasiswa dan para penggiat serta pengembang sistem informasi mampu meningkatkan kemampuan analisis, pengkajian dan penelaahaan lebih mendalam.

Penulis juga mengucapkan banyak terimakasih atas dukungan dan bantuan para pimpinan, rekan-rekan dosen, sejawat sebidang informatika baik dari kalangan akademisi maupun praktisi pengembang sistem informasi serta pengarang buku-buku tentang Analisis dan Perancangan Sistem Informasi baik dari terbitan luar maupun dalam negeri atas terselesainya buku ini. Semoga buku ini dapat membantu dan mendukung tercapainya tujuan belajar mengajar dan penyebaran ilmu pengetahuan bidang sistem informasi.

Buku ini telah melakukan transformasi dan perubahan dari masukan, rekomendasi serta revisi dalam menyempurnakan materi hingga mengupgrade ke materi terbaru. Namun Penulis juga menyadari masih adanya kekurangan dan keterbatasan pada buku ini, maka penulis tetap mengharapkan kritik dan saran dari berbagai pihak agar buku ini bisa terus dikembangkan ke versi revisi berikutnya dikemudian hari.

Akhir kata semoga segala upaya yang penulis lakukan ini bermanfaat bagi kita semua dan Semoga Allah SWT berkenan memberikan berkahnya sehingga semua harapan dan cita-cita penulis dapat terkabulkan. Amin Ya Robbal Alamin.

Medan, Desember 2021

Ilka Zufria, M.Kom

DAFTAR ISI

Kata Pengantar	i
Kata pengantar Editor	ii
Daftar Isi	iii
BAB I KONSEP DASAR SISTEM	1
1.1 Pengertian Sistem	2
1.2 Karakteristik Sistem	4
1.3 Pengertian Subsystem.....	5
1.4 Sistem Yang Buruk	6
1.5 Beberapa Konsep sistem yang penting	7
1.6 Pengertian Sistem Informasi	8
1.7 Tipe-tipe sistem informasi	11
BAB II STAKEHOLDER DALAM SISTEM INFORMASI	14
2.1 Stakeholder.....	14
BAB III ANALISIS DAN PERANCANGAN SISTEM	18
3.1 Pengertian Analisis dan Perancangan Sistem	18
3.2 Metodologi pengembangan Sistem.....	19
3.3 Kelemahan dari SDLC tradisional.....	23
BAB IV PENDEKATAN-PENDEKATAN PENGEMBANGAN SISTEM	24
4.1 Structured analysis dan structured design	24
4.2 Object Oriented Analysis and Design (OOAD)	24
4.3 Prototyping	25
4.4 Joint Application Design (JAD)	26
4.5 Participatory Design.....	27
4.5 Meningkatkan Produktifitas Pengembangan Sistem Informasi	27

BAB V ANALISIS SISTEM	29
5.1 Definisi Analisis Sistem	29
5.2 Definisi Desain Sistem	29
5.3 Analisa sistem informasi	29
5.4 Pendekatan-pendekatan analisis model Driven	30
5.5 Arti penting tahapan Analisis	31
BAB VI PENDEFINISIAN MASALAH	32
6.1 Pendefinisian masalah	32
6.2 Sasaran dan batasan sistem informasi	32
6.3 Analisis PIECES	33
BAB VII PENENTUAN KEBUTUHAN SISTEM	52
7.1 System requirement (Kebutuhan Sistem)	52
7.2 Tipe-tipe Kebutuhan Sistem	52
7.3 Teknik Pengumpulan Requirement	57
BAB VIII ANALISA USE CASE	60
8.1 Analisa Use Case	60
8.2 Peranan Use case	60
BAB IX PROCESS MODELLING	70
9.1 Process model	70
9.2 Data Flow Diagrams	70
9.3 Menggambarkan proses bisnis dengan menggunakan DF	73
BAB X DATA MODELLING	84
10.1 Data Model	84
10.2 The Entity-Relationship Diagram (ERD)	84
BAB XI DESAIN SISTEM	102
11.1 Fase Desain	102
11.2 Strategi Desain	102

BAB XII DISAIN ARSITEKTUR	107
12.1 Desain Arsitektur	107
12.2 Elemen dari desain arsitektur	107
12.3 Tujuan desain arsitektur	108
12.4 Pilihan arsitektur	109
BAB XIII HIRARCHY INPUT OUTPUT CHART (HIPO)	113
13.1 HIPO	113
13.2 Jenis Diagram HIPO	115
BAB XIV PEDOMAN DISAIN INTERFACE	118
BAB XV IMPLEMENTASI SISTEM	120
15.1 Implementasi sistem	120
15.2 Pengetesan Sistem	122
BAB XVI PEMELIHARAAN SISTEM	126
16.1 Pengertian Pemeliharaan Sistem	126
16.2 Prosedur-Prosedur Pemeliharaan Sistem	128
16.3 Alat-Alat Pemeliharaan Sistem	129
16.4 Mengatur Pemeliharaan Sistem	129
16.5 Langkah-Langkah Pemeliharaan Sistem	129
16.6 Mengembangkan Perubahan Sistem Manajemen	130
BAB XVII AUDIT SISTEM INFORMASI	131
17.1 Pendahuluan	131
17.2 Pengertian Audit Sistem Informa	132
17.3 Tujuan Audit Sistem Informasi	132
17.4 Waktu Audit Sistem Informasi	134
17.5 Spesialis Audit Sistem Informasi	136
17.6 Summary Audit Sistem Informasi	138
Daftar Pustaka	140
Daftar Riwayat	145

BAB I

Konsep Dasar Sistem

1.1 Pengertian Sistem

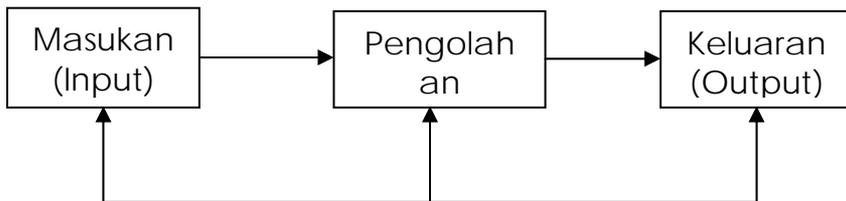
Definisi sistem berkembang sesuai dengan konteks dimana pengertian sistem itu digunakan. Disini akan diberikan beberapa definisi sistem secara umum:

- Kumpulan dari bagian-bagian yang bekerja bersama-sama untuk mencapai tujuan yang sama
 - Contoh
 - Sistem tatasurya
 - Sistem pencernaan
 - Sistem Transportasi umum
 - Sistem Otomotif
 - Sistem Komputer
 - Sistem Informasi
- Sekumpulan dari objek-objek yang saling berelasi dan berinteraksi dan hubungan antar objek bisa dilihat sbg 1 kesatuan yang dirancang untuk mencapai 1 tujuan

Dengan demikian secara sederhana sistem dapat diartikan sebagai suatu kumpulan atau himpunan dari unsur atau variable-variabel yang saling teroganisasi, saling berinteraksi dan saling bergantung sama lain. *Murdick dan Ross (1993)* mendefinisikan sistem sebagai seperangkat elemen yang digabungkan satu dengan lainnya untuk suatu tujuan bersama. Sedangkan definisi sistem dalam kamus *Webster's Unbrigid*

adalah elemen-elemen yang saling berhubungan membentuk satu kesatuan atau organisasi.

Scott (1996) mengatakan sistem terdiri dari unsur-unsur seperti masukan (*input*), pengolahan (*processing*), serta keluaran (*output*). Ciri pokok sistem menurut Gasperspert ada empat, yaitu sistem itu beroperasi dalam suatu lingkungan, terdiri atas unsur-unsur, ditandai dengan saling berhubungan dan mempunyai satu fungsi atau tujuan utama (Nugroho, 2005)¹.

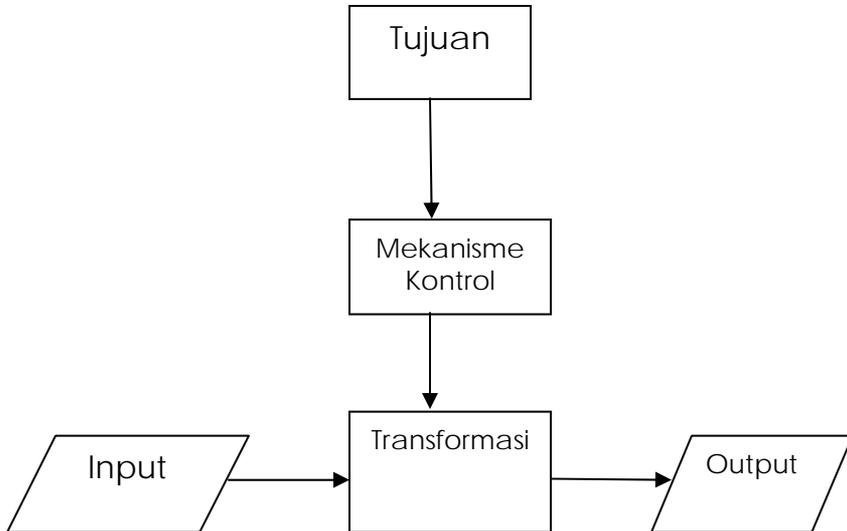


Gambar. 1.1 Model sistem

Gambar diatas menunjukkan bahwa sistem atau pendekatan sistem minimal harus mempunyai empat komponen, yakni masukan, pengolahan, keluaran dan, balikan atau control.

Sementara Mc. Leod (1995) mendefinisikan sistem sebagai sekelompok elemen-elemen yang terintegrasi dengan maksud yang sama untuk mencapai suatu tujuan. Sumberdaya mengalir dari elemen output dan untuk menjamin prosesnya berjalan dengan baik maka dihubungkan mekanisme control. Untuk lebih jelasnya elemen sistem tersebut dapat digambarkan dengan model sebagai berikut :

1] Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Edisi: Revisi, Informatika, Bandung



Gambar. 1.2. *Model hubungan elemen-elemen sistem*

Banyak ahli mengajukan konsep sistem dengan deskripsi yang berbeda namun pada prinsipnya hamper sama dengan konsep dasar sistem umumnya. *Schronderberg (1971)* dalam *Suradinata (1996)* secara ringkas menjelaskan bahwa sistem adalah

1. Komponen-komponen sistem saling berhubungan satu sama lainya.
2. Suatu keseluruhan tanpa memisahkan komponen pembentukanya.
3. Bersama-sama dalam mencapai tujuan.
4. Memiliki input dan output yang dibutuhkan oleh sistem lainnya.
5. Terdapat proses yang merubah input menjadi output.
6. Menunjukkan adanya entropi
7. terdapat aturan
8. Terdapat subsistem yang lebih kecil.
9. terdapat deferensiasi antar subsistem.
10. Terdapat tujuan yang sama meskipun mulainya berbeda.

1.2. Karakteristik Sistem.

Untuk memahami atau mengembangkan suatu sistem, maka perlu membedakan unsur-unsur dari sistem yang membentuknya. Berikut ini karakteristik sistem yang dapat membedakan suatu sistem dengan sistem lainnya.

1. Batasan (*boundary*) : Penggambaran dari suatu elemen atau unsur mana yang termasuk didalam sistem dan mana yang diluar sistem.
2. Lingkungan (*environment*) : Segala sesuatu diluar sistem, lingkungan yang menyediakan asumsi, kendala dan input terhadap suatu sistem
3. Masukan (*input*) : Sumberdaya (data, bahan baku, peralatan, energi) dari lingkungan yang dikonsumsi dan dimanipulasi oleh suatu sistem.
4. Keluaran (*output*) : Sumber daya atau produk (informasi, laporan, dokumen, tampilan layer computer, barang jadi) yang disediakan untuk lingkungan sistem oleh kegiatan dalam suatu sistem.
5. Komponen (*component*) : Kegiatan-kegiatan atau proses dalam suatu sistem yang mentransformasikan input menjadi bentuk setengah jadi (*output*). Komponen ini bisa merupakan subsistem dari sebuah sistem.
6. Penghubung (*interface*) : Tempat dimana komponen atau sistem dan lingkungannya bertemu atau berinteraksi.
7. Penyimpanan (*storage*) : Area yang dikuasai dan digunakan untuk penyimpanan sementara dan tetap dari informasi, energi, bahan baku dan sebagainya. Penyimpanan merupakan suatu media penyangga diantara komponen tersebut bekerja dengan berbagai tingkatan yang ada dan memungkinkan komponen yang berbeda dari berbagai data yang sama.

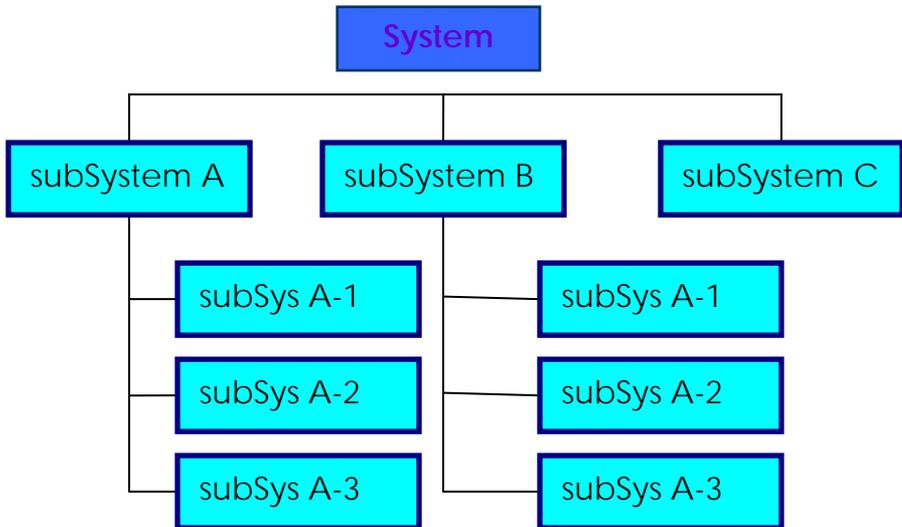
1.3 Pengertian Subsistem

Suatu sistem yang kompleks biasanya tersusun atas beberapa subsistem. Subsistem bisa dijelaskan sebagai sebuah sistem dalam sistem yang lebih besar. Sebagai contoh :

Automobile adalah sistem yang terdiri dari beberapa subsistem:

- Sistem mesin
 - Sistem Body
 - Sistem Roda
- Setiap sub sistem bisa terdiri dari beberapa sub-sub-systems.
- Sistem mesin: sistem karburator, sistem generator, sistem bahan bakar dan lain-lain

Untuk lebih jelasnya bisa dilihat pada bagan berikut ini:



Gambar 1.3 Gambaran subsistem dalam sistem

1.4 Sistem Yang Buruk

Untuk menghindari pengembangan suatu sistem yang buruk maka perlu diketahui beberapa ciri-ciri dari sistem yang buruk:

- Tidak memenuhi kebutuhan user
- Performance buruk
- Reliabilitas rendah
- Kegunaan rendah
- Contoh-contoh kesulitan:
 - Tidak terjadwal
 - Tidak ada rencana anggaran
 - Bisa jalan = 100% over budget atau jadwal

1.6.1. Komponen Sistem Informasi

Stair (1992) menjelaskan bahwa sistem informasi berbasis komputer (CBIS) dalam suatu organisasi terdiri dari komponen-komponen berikut :

- **Hardware**, yaitu perangkat keras komponen untuk melengkapi kegiatan memasukan data, memproses data dan keluaran data.
- **Software**, yaitu program dan instruksi yang diberikan kekomputer.
- **Database**, yaitu kumpulan data dan informasi yang diorganisasikan sedemikian rupa sehingga mudah diakses pengguna sistem informasi.
- **Telekomunikasi**, yaitu komunikasi yang menghubungkan antara pengguna sistem dengan sistem komputer secara bersama-sama kedalam suatu jaringan kerja yang efektif.
- **Manusia**, yaitu personil dari sistem informasi, meliputi manajer, analis, programmer, operator dan bertanggungjawab terhadap perawatan sistem.

Prosedur, yakni tata cara yang meliputi strategi, kebijakan, metode dan peraturan-peraturan dalam menggunakan sistem informasi berbasis komputer.

Pendapat *Burch dan Grudnistki (1986)*, sistem informasi terdiri dari komponen-komponen diatas disebutnya dengan istilah blok bangunan (*building block*), yaitu blok masukan (*input block*), blok model (*model block*), blok mkeluaran (*output block*), blok teknologi (*technology block*) dan blok kendali (*control block*). Sebagai suatu sistem, keenam blok tersebut masing-masing saling berinteraksi satu dengan yang lainnya membentuk satu kesatuan untuk mencapai sarasanya (Tim Politeknik Bandung, 2009)².

[2] Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung

1. **Blok Masukan.** Input mewakili data yang masuk kedalam sistem informasi. Input disini termasuk metode-metode dan media untuk menangkap data yang akan dimasukan yang dapat berupa dokumen-dokumen dasar.
2. **Blok Model.** Blok ini terdiri dari kombinasi prosedur, logika dan model matematika yang akan memanipulasi data input dan data yang tersimpan di basis data dengan cara tertentu untuk menghasilkan keluaran yang diinginkan.
3. **Blok Keluaran.** Produk dari sistem informasi adalah keluaran yang merupakan informasi yang berkualitas dan dokumentasi yang berguna untuk semua tingkat manajemen serta semua pemakai sistem.
4. **Blok Teknologi.** Teknologi merupakan kotakalat (tool-box) dalam sistem informasi. Teknologi digunakan untuk menerima input, menjalankan model, menyimpan dan mengakses data, menghasilkan sekaligus mengirimkan keluaran dan membantu pengendalian dari sistem secara keseluruhan.
5. **Blok Basis Data.** Basis Data (Data Base) merupakan kumpulan dari data yang saling berhubungan satu dengan lainnya, tersimpan diperangkat keras komputer dan digunakan perangkat lunak untuk memanipulasinya.
6. **Blok Kendali.** Pengendalian perlu dirancang dan diterapkan untuk meyakinkan bahwa hal-hal yang dapat merusak sistem dapat dicegah ataupun bila terlanjur terjadi kesalahan-kesalahan dapat langsung cepat diatasi.

Sedangkan menurut pendapat *Davis (1995)* sistem informasi manajemen terdiri dari elemen-elemen berikut :

1. Perangkat keras komputer (*hardware*).
2. Perangkat Lunak (*software*), yang terdiri dari perangkat lunak sistem umum, perangkat lunak terapan, program aplikasi.
3. Database.
4. Prosedur.
5. Petugas pengoperasian.

1.6.2 Computer Based Information System

Istilah Computer Based Information System (CBIS), sebenarnya mengacu kepada sistem informasi yang dikembangkan berbasis teknologi komputer.

Computer-based Information System = Hardware + Software + People + Procedures + Information

Dalam modul ini, CBIS selanjutnya akan disebut sebagai sistem informasi saja.

1.7 Tipe-tipe sistem informasi

CBIS biasanya dibedakan menjadi beberapa tipe aplikasi, yaitu :

- **Transaction Processing Systems (TPS)**
- **Management Information Systems (MIS)**
- **Decision Support Systems (DSS)**
- **Expert System and Artificial Intelligence (ES &AI)**

1.7.1 Transaction Processing System

TPS adalah sistem informasi terkomputerisasi yang dikembangkan untuk memproses sejumlah besar data untuk transaksi bisnis rutin (Tim Politeknik Bandung, 2009)².

- 1 Mengotomasi penanganan data-data aktifitas bisnis dan transaksi. Yang bisa dianggap sebagai kejadian diskrit dalam kehidupan organisasi :
- 2 Data setiap transaksi ditangkap
- 3 Transaksi di verifikasi untuk diterima atau ditolak
- 4 Transaksi yang telah di validasi disimpan untuk pengumpulan data berikutnya.
- 5 Laporan bisa dihasilkan untuk menyediakan rangkuman dari setiap transaksi
- 6 Transaksi bisa dipindah dari 1 proses ke proses yang lainnya untuk menangani seluruh aspek bisnis

1.7.2 Management Information System

Management Information System (MIS) atau Sistem informasi Manajemen adalah sebuah sistem informasi pada level manajemen yang berfungsi untuk membantu perencanaan, pengendalian dan pengambilan keputusan dengan menyediakan resume rutin dan laporan-laporan tertentu. SIM mengambil data mentah dari TPS dan mengubahnya menjadi kumpulan data yang lebih berarti yang dibutuhkan manager untuk menjalankan tanggung jawabnya. Untuk mengembangkan suatu SIM diperlukan pemahaman yang baik tentang informasi apa saja yang dibutuhkan manager dan bagaimana mereka menggunakan informasi tersebut.

[2] Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung

1.7.3 Decision Support System

Sistem informasi pada level manajemen dari suatu organisasi yang mengkombinasikan data dan model analisa canggih atau peralatan data analisis untuk mendukung pengambilan yang semi terstruktur dan tidak terstruktur. DSS dirancang untuk membantu pengambilan keputusan organisasional. DSS biasanya tersusun atas :

- Basis Data (bisa diekstraksi dari TPS/MIS)
- Model grafis atau Matematis yang digunakan untuk proses bisnis
- Use interface yang digunakan oleh user untuk berkomunikasi dengan DSS

1.7.4 Expert System

Representasi pengetahuan yang menggambarkan cara seorang ahli dalam mendekati suatu masalah. ES lebih berpusat pada bagaimana mengkodekan dan memanipulasi pengetahuan daripada informasi (misalnya aturan if...then). B iasanya ES bekerja sebagai berikut:

- User berkomunikasi dengan sistem menggunakan interaktif dialog
- ES menanyakan pertanyaan (yang akan ditanyakan seorang pakar), dan pengguna memberikan jawaban.
- Jawaban digunakan untuk menentukan aturan mana yang dipakai, dan ES sistem menyediakan rekomendasi berdasarkan aturan yang telah disimpan.
- Seorang knowledge engineer bertanggung jawab pada bagaimana melakukan akuisisi pengetahuan, sama seperti seorang analis tetapi dilatih untuk menggunakan teknik yang berbeda.

BAB 2

Stake Holder dalam Sistem Informasi

2.1 Stake Holder

Stake Holder adalah orang yang memiliki kepentingan tertentu pada suatu kegiatan bisnis (Tim Politeknik Bandung, 2009)².

Di dalam pengembangan sebuah sistem informasi stake holder dapat dibedakan menjadi:

- Manager SI
- System analyst pada pengembangan sistem
- Programmer dalam pengembangan sistem
- End user dalam pengembangan sistem
- Supporting end user
- Business manager
- Teknisi SI lainnya

2.1.1 Manager SI

Manager dalam departemen Sistem informasi memiliki peranan secara langsung dalam proses pengembangan sistem jika organisasi yang ditanganinya berskala kecil. Manager SI berperan dalam mengalokasikan dan mengawasi proyek pengembangan sistem daripada terlibat langsung dalam proses pengembangan sistem. Ada beberapa manager SI pada departemen SI yang berskala besar:

- Manager untuk keseluruhan departemen SI biasa disebut sebagai Chief Information Officer dan berada dibawah president atau direktur perusahaan.

[2] Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung

- Setiap divisi dalam departemen SI juga memiliki seorang manager misalnya manager pengembangan SI, Manager operasi ,manager programmer SI dan lain-lain

2.1.2 Systems Analysts

Sistem analis merupakan individu kunci dalam proses pengembangan sistem. Sistem analis mempelajari masalah dan kebutuhan dari organisasi untuk menentukan bagaimana orang, data, proses, komunikasi dan teknologi informasi dapat meningkatkan pencapaian bisnis. Seorang sistem analis juga merupakan orang yang paling bertanggung jawab pada proses analisa dan perancangan sistem informasi.

Seorang sistem analis yang sukses harus memiliki beberapa skill.

Keahlian analisa

- Memahami organisasi
- Keahlian memecahkan masalah
- Pemahaman sistem, untuk melihat organisasi dan sistem informasi sebagai sebuah sistem.

Keahlian teknis

- Memahami potensi dan limitasi dari suatu teknologi

Keahlian Managerial

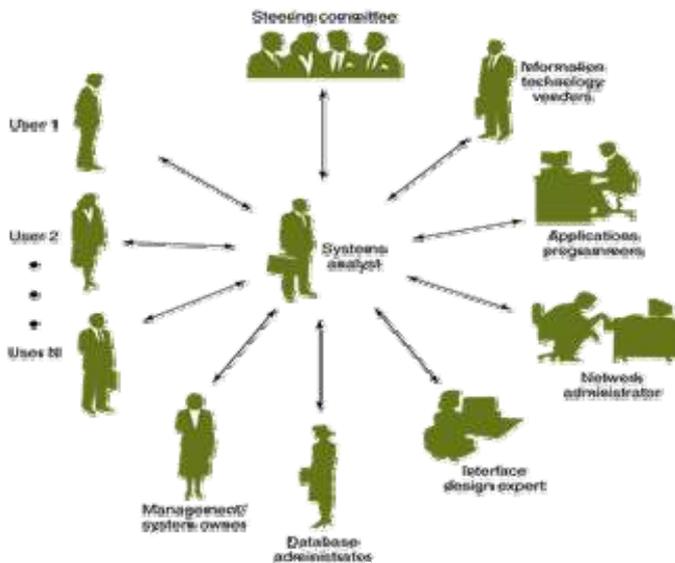
- Kemampuan untuk mengatur proyek, sumber daya resiko dan perubahan.

Interpersonal skills

- Kemampuan untuk berkomunikasi secara aktif baik tertulis maupun lisan
- Sangat membantu untuk komunikasi dengan end user, sistem analis maupun programmer

Adapun tanggung jawab dari seorang sistem analis meliputi :

1. Pengambilan data yang efektif dari sumber bisnis
2. Aliran data menuju ke komputer
3. Pemrosesan dan penyimpanan data dengan komputer
4. Aliran dari informasi yang berguna kembali ke proses bisnis dan penggunaanya



Gambar 2.1 Posisi sistem analis di antara stack holder yang lain

2.1.3 Programmer

Programmer mengubah Spesifikasi yang diberikan oleh sistem analis ke dalam instruksi yang bisa dijalankan oleh komputer. Langkah mngubah ke dalam kode yang bisa dijalankan komputer ini disebut coding. Code generator telah dikembangkan untuk menghasilkan kode dari spesifikasi yang telah dibuat, menghemat waktu dan biaya. Tujuan dari penggunaan CASE (Computer Aided Software Engineering) adalah untuk menyediakan beberapa code generator yang secara otomatis menghasilkan 90% atau lebih dari spesifikasi sistem normal yang diberikan oleh programmer secara normal.

2.1.4 Business manager

Kelompok lain dalam pengembangan sistem adalah manajer bisnis misalnya kepala bagian atau kepala departemen atau eksekutif perusahaan. Manajer-manajer ini penting karena mereka memiliki kekuatan pendanaan pengembangan sistem dan mengalokasikan sumber daya yang diperlukan untuk keberhasilan proyek.

2.1.5 Teknisi lainnya

Masih banyak lagi teknisi lain yang terlibat dalam pengembangan sistem diantaranya:

- database administrator
- Ahli network dan telekomunikasi

BAB 3

Analisa dan Perancangan Sistem

3.1 Pengertian Analisa dan Perancangan Sistem

Analisa sistem didefinisikan sebagai bagaimana memahami dan menspesifikasi dengan detail apa yang harus dilakukan oleh sistem. Sedangkan sistem desain diartikan sebagai menjelaskan dengan detail bagaimana bagian-bagian dari sistem informasi diimplementasikan. Sehingga Analisa dan desain sistem informasi (ANSI) bisa didefinisikan sebagai: Proses organisasional kompleks dimana sistem informasi berbasis komputer diimplementasikan. Sehingga bisa diringkas sebagai berikut: (Langer, Arthur M., 2008)³

Analysis: mendefinisikan masalah

– **From requirements to specification**

Design: memecahkan masalah

–**From specification to implementation**

Kenapa tahapan-tahapan ini penting?

- Kesuksesan suatu sistem informasi tergantung pada analisa dan perancangan yang baik
- ANSI telah digunakan secara luas di berbagai industri (teknologi yang telah teruji)
- Bagian dari karir dalam dunia IT, menawarkan banyak kesenangan dan tantangan serta gaji yang tidak rendah
- Kenaikan permintaan akan keahlian sistem analis

[3] Langer, Arthur M. 2008. *Analysis and Design of Information Systems 3rd edition*. Springer.

Secara singkat berdasarkan pendekatan ANSI, seorang sistem analios adalah orang yang bertugas:

- Bagaimana membangun sistem informasi
- Bagaimana menganalisa kebutuhan dari sistem informasi
- Bagaimana merancang sebuah Sistem Informasi berbasis komputer
- Bagaimana memecahkan masalah dalam organisasi melalui sistem informasi

3.2 Metodologi pengembangan Sistem

Proses-proses standard yang digunakan untuk membangun suatu sistem informasi meliputi langkah-langkah berikut ini:

- Analisa
- Desain
- Implementasi
- Maintenance

Pada perkembangannya, proses-proses standar tadi dituangkan dalam satu metode yang dikenal dengan nama Systems Development Life Cycle (SDLC) yang merupakan metodologi umum dalam pengembangan sistem yang menandai kemajuan dari usaha analisa dan desain. SDLC meliputi fase-fase sebagai berikut:

1. Identifikasi dan seleksi proyek
2. Inisiasi dan perencanaan proyek
3. Analisa
4. Desain
 - Desain logikal
 - Desain Fisikal
5. Implementasi
6. Maintenance

3.2.1 Identifikasi dan seleksi proyek

Langkah pertama dalam SDLC keseluruhan informasi yang dibutuhkan oleh sistem diidentifikasi, dianalisa, diprioritaskan dan disusun ulang. Dalam langkah ini dilakukan beberapa hal diantaranya:

- Mengidentifikasi proyek-proyek yang potensial
- Melakukan klasifikasi dan me-rangking proyek
- Memilih proyek untuk dikembangkan.

Adapun sumber daya yang terlibat adalah *user, sistem analis, manager yang mengkoordinasi proyek*

Aktivitas yang dilakukan meliputi: *mewawancarai manajemen user, merangkum pengetahuan yang didapatkan, dan mengestimasi cakupan proyek dan mendokumentasikan hasilnya*

Output: Laporan kelayakan berisi definisi masalah dan rangkuman tujuan yang ingin dicapai

3.2.2 Inisiasi dan perencanaan proyek

Dalam tahapan ini Proyek SI yang potensial dijelaskan dan argumentasi untuk melanjutkan proyek dikemukakan. Rencana kerja yang matang juga disusun untuk menjalankan tahapan-tahapan lainnya. Hasil dari tahapan ini adalah :

Langkah-langkah detail-rencana kerja-high level system requirement-penugasan untuk anggota tim.

3.2.3 Tahapan Analisa

Fase ketiga dalam SDLC dimana sistem yang sedang berjalan dipelajari dan sistem pengganti diusulkan. Dalam tahapan ini dideskripsikan sistem yang sedang berjalan, masalah dan kesempatan didefinisikan, dan rekomendasi umum untuk bagaimana memperbaiki, meningkatkan atau mengganti sistem yang sedang berjalan diusulkan. Tujuan utama dari fase analisis adalah untuk memahami dan mendokumentasikan kebutuhan

bisnis (business need) dan persyaratan proses dari sistem baru. Ada 6 aktifitas utama dalam fase ini:

- Pengumpulan informasi
- Mendefinisikan sistem requirement
- Membangun prototype untuk menemukan requirement
- Memprioritaskan requitement
- Menyusun dan mengevaluasi alternatif
- Mereview requiremen dengan pihak manajemen

3.2.4 Tahapan Desain

Pada tahapan ini deskripsi dari requirement yang telah direkomendasikan diubah ke dalam spesifikasi sistem physical dan logical.

Logical Design

Bagian dari fase desain dalam SDLC dimana semua fitur-fitur fungsional dari sistem dipilih dari tahapan analisis dideskripsikan terpisah dari platform komputer yang nanti digunakan. Hasil dari tahapan ini adalah :

- Deskripsi fungsional mengenai data dan proses yang ada dalam sistem baru
- Deskripsi yang detail dari spesifikasi sistem meliputi:
 - Input
 - Output
 - Process

Physical design

Pada bagian ini spesifikasi logical diubah ke dalam detail teknologi dimana pemrograman dan pengembangan sistem bisa diselesaikan. Adapun output dari sistem ini adalah :

BAB 4

Pendekatan-Pendekatan Pengembangan Sistem

Disamping metode tradisional SDLC, ada beberapa metode yang dikembangkan untuk melengkapi kelemahan-kelemahan yang ada dalam metode SDLC. Metode-metode itu antara lain:

- Structured analysis and structured design
- Object oriented analysis and design
- Prototyping
- Joint Application Design (JAD)
- Participatory design

4.1 Structured analysis dan structured design

Pendekatan ini lebih berfokus pada bagaimana mereduksi waktu dan maintenace dalam pengembangan sistem.

Pendekatan ini juga langsung mengintegrasikan perubahan jika diperlukan.

4. 2 Object Oriented Analysis and Design (OOAD)

Pendekatan baru untuk pengembangan sistem, sering disebut sebagai pendekatan ketiga setelah pendekatan yang berorientasi data dan berorientasi proses. OOAD adalah metode pengembangan sistem yang lebih menekankan pada objek

dibandingkan dengan data atau proses. Ada beberapa ciri khas dari pendekatan ini yaitu object, Inheritance dan object class

Object adalah struktur yang mengenkapsulasi atribut dan metode yang beroperasi berdasarkan atribut-atribut tadi. Objek adalah abstraksi dari benda nyata dimana data dan proses diletakkan bersama untuk memodelkan struktur dan perilaku dari objek dunia nyata.

Object class adalah sekumpulan objek yang berbagi struktur yang sama dan perilaku yang sama.

Inheritance, Properti yang muncul ketika tipe entitas atau object class disusun secara hirarki dan setiap tipe entitas atau object class menerima atau mewarisi atribut dan metode dari pendahulunya (Simon Bennet, 2005)⁵

4.3. Prototyping

Prototyping adalah proses iterative dalam pengembangan sistem dimana requirement diubah ke dalam sistem yang bekerja (working system) yang secara terus menerus diperbaiki melalui kerjasama antara user dan analis. Prototype juga bisa dibangun melalui beberapa tool pengembangan untuk menyederhanakan proses. Prototyping merupakan bentuk dari Rapid Application Development (RAD). Beberapa kerugian RAD: (Bowman, Kevin, 2004)⁴

- RAD mungkin mengesampingkan prinsip-prinsip rekayasa perangkat lunak
- Menghasilkan inkonsistensi pada modul-modul sistem
- Tidak cocok dengan standar
- Kekurangan prinsip reusability komponen

[4] Bowman, Kevin. 2004. *System Analysis: A Beginner's Guide*. Palgrave Macmillan.

[5] Simon Bennet, Steve McRobb, Ray Farmer, 2002, *Object Oriented Systems Analysis and Design Using UML 2nd*, McGraw Hill

Prototype methodology

- Analisis bekerja dengan tim untuk mengidentifikasi requirement awal untuk sistem
- Analisis kemudian membangun prototype. Ketika sebuah prototype telah selesai. User bekerja dengan prototype itu dan menyampaikan pada analisis apa yang mereka sukai dan yang tidak mereka sukai.
- Analisis kemudian menggunakan feedback ini untuk memperbaiki prototype
- Versi baru diberikan kembali ke user
- Ulangi langkah-langkah tersebut sampai user merasa puas

Keuntungan prototype

- Prototype melibatkan user dalam analisa dan desain
- Punya kemampuan menangkap requirement secara konkret daripada secara abstrak
- Untuk digunakan secara standalone
- Digunakan untuk memperluas SDLC

4.4 Joint Application Design (JAD)

Pada akhir 1970 an personil pengembangan sistem di IBM mengembangkan proses baru untuk mengumpulkan requiremen SI dan mereview desain dengan nama JAD. JAD adalah proses terstruktur dimana user, manager dan analisis bekerja bersama-sama selama beberapa hari dalam 1 pertemuan bersama untuk mengumpulkan requiremen sistem yang akan dibangun.

BAB 5

Analisis Sistem

5.1 Definisi Analisis Sistem

Analisis sistem adalah sebuah istilah yang secara kolektif mendeskripsikan fase-fase awal pengembangan sistem. Analisis sistem adalah teknik pemecahan masalah yang menguraikan bagian-bagian komponen dengan mempelajari seberapa bagus bagian-bagian komponen tersebut bekerja dan berinteraksi untuk mencapai tujuan mereka (Nugroho, Adi, 2005)⁶

5.2 Definisi Desain Sistem

Sebuah teknik pemecahan masalah yang saling melengkapi (dengan analisa sistem) yang merangkai kembali bagian-bagian komponen menjadi sistem yang lengkap-harapannya, sebuah sistem yang diperbaiki. Hal ini melibatkan penambahan, penghapusan, dan perubahan-perubahan bagian relatif pada sistem awal (aslinya).

5.3 Analisa sistem informasi

Fase-fase pengembangan dalam pembangunan sistem informasi yang utamanya difokuskan pada masalah dan persyaratan- persyaratan bisnis, terpisah dari teknologi apapun yang dapat atau akan digunakan untuk mengimplementasikan solusi pada masalah tersebut.

[6] Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Edisi: Revisi, Informatika, Bandung

5.4 Pendekatan-pendekatan analisis model Driven

Model Driven adalah analisis yang menggunakan gambar untuk mengkombinasikan masalah-masalah, persyaratan-persyaratan, dan solusi-solusi bisnis. Contoh model yang sering dipakai adalah flowchart dan DFD. Contoh dari analisis dengan pendekatan model driven adalah :

- Analisis terstruktur
- Rekayasa informasi
- Analisa berorientasi objek

5.4.1 Analisis Terstruktur

Dulu adalah salah satu pendekatan formal pertama untuk analisis sistem informasi. Analisis ini terfokus pada aliran data dan proses bisnis dan perangkat lunak. Analisis ini disebut proses oriented. Analisis terstruktur sederhana dalam konsep. Para analis menggambarkan serangkaian proses dalam bentuk diagram alir data (Data flow diagram) yang menggambarkan proses yang ada atau yang diusulkan bersama-sama dengan input, output dan file mereka.

5.4.2 Rekayasa informasi dan pemodelan data

Rekayasa informasi dulu terfokus pada struktur data yang tersimpan pada sebuah sistem. Jadi dikatakan data centered. Model-model data dalam rekayasa informasi disebut entity relationship. Untuk lebih jelasnya akan dibahas pada modul berikutnya. Rekayasa informasi dikatakan berpusat pada data karena menekankan pada pembelajaran dan analisa persyaratan data sebelum persyaratan-persyaratan proses. Hal ini didasarkan pada tingkat kepercayaan bahwa data dan pengetahuan adalah sumber daya perusahaan yang harus direncanakan dan dipelihara. Akibatnya analis menggambarkan model data mentah ke dalam hubungan entitas, baru kemudian membuat diagram alir data yang menjelaskan proses-proses

yang terjadi. Rekayasa informasi sebenarnya berusaha mensinkronkan pemodelan data dengan pemodelan proses. Perbedaannya hanya pada analisis terstruktur digambarkan pemodelan proses terlebih dahulu, baru pemodelan data.

5.4.3 Analisa berorientasi objek

Selama 30 tahun, kebanyakan pendekatan pengembangan sistem telah memisahkan pengetahuan(data) dari proses. Teknik objek muncul untuk memisahkan menghilangkan pemisahan data dan proses ini. Sebaliknya data dan proses spesifik yang membuat, membaca, memperbarui atau menghapus data itu diintegrasikan ke dalam konstruksi yang disebut data atau objek. Satu satunya cara untuk membuat, membaca, memperbarui atau menghapus data adalah dengan cara proses perlekatan (embeded) yang disebut metode. OOA adalah teknik yang model driven yang mengintegrasikan data dan proses yang disebut objek. Model OOA adalah gambar-gambar yang mengilustrasikan objek-objek sistem dari berbagai perspektif, seperti struktur, perilaku dan interaksi antar objek. Contoh yang paling terkenal adalah UML (Unified Modelling Language).

5.5 Arti penting tahapan Analisis

Banyak sistem informasi bagus yang akhirnya ditinggalkan user karena sistem analisis tidak punya pengertian yang benar tentang organisasi. Tujuan dari analisa sistem adalah menghindari kondisi ini, dengan menjawab pertanyaan-pertanyaan berikut:

- Siapa yang akan menggunakan sistem?
- Sistem yang dikembangkan akan seperti apa?
- Apa yang bisa dikerjakan sistem?
- Kapan sistem akan digunakan?

BAB 6

ANALISA SISTEM

(Bagian 2)

6.1 Pendefinisian masalah

Pada tahapan analisis sistem, analis mempunyai tugas mendefinisikan masalah sistem, melakukan studi kelayakan dan menganalisis kebutuhan sistem yang akan dikembangkan. Terdapat 3 pertanyaan kunci yang harus dijawab, yaitu :

- Apa masalah yang harus diselesaikan dengan sistem informasi?
- Apa penyebab masalah tersebut?
- Siapa pemakai akhir dari sistem?

Masalah yang dipelajari analis sistem adalah masalah yang dihadapi pemakai. Dengan mempelajari masalah ini, maka analis bekerjasama dengan pemakai untuk mendapatkan permasalahan secara kasar. Langkah-langkah yang harus dijalankan adalah:

- Mendefinisikan batasan dan sasaran
- Mendefinisikan masalah yang dihadapi pemakai
- Mengidentifikasi penyebab masalah dan titik keputusan
- Mengidentifikasi pemakai akhir
- Memilih prioritas penanganan masalah
- Memperkirakan biaya dan manfaat secara kasar
- Membuat laporan hasil pendefinisian masalah

6.2 Sasaran dan batasan sistem informasi

Sasaran sistem informasi adalah peningkatan kinerja, peningkatan efektifitas informasi, penurunan biaya, peningkatan keamanan aplikasi, peningkatan efisiensi dan peningkatan pelayanan pada pelanggan. Penyimpangan dari keenam sasaran inilah yang menimbulkan masalah pada sistem informasi. Batasan sistem adalah lingkungan yang membatasi aplikasi, misalnya peraturan-peraturan siapa yang boleh menggunakan sistem, dan siapa yang tidak boleh.

Masalah dalam sistem informasi

Masalah dalam sistem informasi adalah kondisi atau situasi yang menyimpang dari sasaran sistem informasi, bahkan menyimpang dari sasaran organisasi atau perusahaan, misalnya kinerja mengalami penurunan, informasi tidak efektif, sistem informasi tidak aman. Biasanya masalah dinyatakan dalam pertanyaan misalnya:

- Apakah sistem informasi ini dapat meningkatkan kinerja
- Apakah sistem informasi dapat menurunkan biaya
- Apakah sistem informasi bisa meningkatkan keamanan
- Apakah sistem informasi bisa menurunkan pemborosan
- Apakah sistem informasi bisa meningkatkan penjualan
- Apakah sistem informasi bisa meningkatkan pelayanan

6.3 ANALISIS PIECES

Untuk mengidentifikasi masalah, maka harus dilakukan analisis terhadap kinerja, informasi, ekonomi, keamanan aplikasi, efisiensi dan pelayanan pelanggan. Panduan ini dikenal dengan PIECES analysis (performance, Information, economy, Control,

eficiency dan Services). Dari analisis ini biasanya didapatkan beberapa masalah utama. Hal ini penting karena biasanya yang muncul dipermukaan bukan masalah utama, tetapi hanya gejala dari masalah utama saja.

a. Analisis Kinerja

Masalah kinerja terjadi ketika tugas-tugas bisnis dijalankan dan tidak mencapai sasaran. Kinerja diukur dengan jumlah produksi dan waktu tanggap. Jumlah produksi adalah jumlah pekerjaan yang bisa diselesaikan selama jangka waktu tertentu. Bagian pemasaran kinerjanya diukur berdasarkan volume pekerjaan atau pangsa pasar yang diraih atau citra perusahaan.

Waktu tanggap adalah keterlambatan rata-rata antara suatu transaksi dengan tanggapan yang diberikan kepada transaksi tersebut.

b. Analisis Informasi

Informasi merupakan komoditas krusial bagi pemakai akhir. Kemampuan sistem informasi dalam menghasilkan informasi yang bermanfaat dapat dievaluasi untuk menangani masalah dan peluang untuk mengatasi masalah tersebut. Dalam hal ini meningkatkan kualitas informasi tidak dengan menambah jumlah informasi, karena terlalu banyak informasi juga menghasilkan masalah baru. Situasi yang membutuhkan peningkatan informasi meliputi:

- Kurangnya informasi mengenai keputusan atau situasi yang sekarang
- Kurangnya informasi yang relevan mengenai keputusan ataupun situasi sekarang
- Kurangnya informasi yang tepat waktu
- Terlalu banyak informasi
- Informasi tidak akurat

Informasi juga dapat merupakan fokus dari suatu batasan atau kebijakan. Sementara analisis informasi memeriksa output sistem, analisis data meneliti data yang tersimpan dalam sebuah sistem. Permasalahan yang dihadapi meliputi:

Data yang berlebihan. Data yang sama ditangkap dan/atau disimpan di banyak tempat.

Kekakuan data. Data ditangkap dan disimpan, tetapi diorganisasikan sedemikian rupa sehingga laporan dan pengujian tidak dapat atau sulit dilakukan.

c. analisis ekonomi

Ekonomis barangkali merupakan motivasi paling umum bagi suatu proyek. Pijakan dasar bagi kebanyakan manajer adalah biaya atau rupiah. Persoalan ekonomis dan peluang berkait dengan masalah biaya. Adapun hal-hal yang perlu diperhatikan dapat disimak berikut ini:

Biaya

- Biaya tidak diketahui
- Biaya tidak dapat dilacak ke sumber
- Biaya terlalu tinggi

Keuntungan

- Pasar-pasar baru dapat dieksplorasi
- Pemasaran saat ini dapat diperbaiki
- Pesanan-pesanan dapat ditingkatkan.

d. Analisis keamanan

Tugas-tugas bisnis perlu dimonitor dan dibetulkan jika ditemukan kinerja yang dibawah standar. Kontrol dipasang untuk meningkatkan kinerja sistem, mencegah atau mendeteksi kesalahan sistem, dan menjamin keamanan data, informasi dan persyaratan. Adapun hal-hal yang perlu diperhatikan adalah:

Keamanan atau kontrol yang lemah

- Input data tidak diedit dengan cukup
- Kejahatan (misalnya penggelapan atau pencurian) terhadap data
- Etika dilanggar pada data atau informasi-mengacu pada data atau informasi yang diakses orang yang tidak berwenang.
- Data tersimpan secara berlebihan, tidak konsisten pada file-file atau database-database yang berbeda.
- Peraturan atau panduan privasi data dilanggar (atau dapat dilanggar)
- Error pemrosesan terjadi (oleh manusia, mesin atau perangkat lunak)
- Error pembuatan keputusan terjadi.

Kontrol atau keamanan berlebihan

- Prosedur birokratis memperlamban sistem
- Pengendalian mengganggu para pelanggan atau karyawan
- Pengendalian berlebihan menyebabkan penundaan pemrosesan.

e. analisis Efisiensi

Efisiensi menyangkut bagaimana menghasilkan output sebanyak-banyaknya dengan dengan input yang sekecil mungkin. Untuk melihat apakah efisiensi dari suatu sistem baik atau tidak dapat dengan melihat indikator-indikator berikut ini:

- Orang, mesin atau komputer membuang-buang waktu
- Data secara berlebihan di input atau disalin
- Data secara berlebihan di proses
- Informasi secara berlebihan dihasilkan
- Orang, mesin atau komputer membuangh
- Usaha yang dibutuhkan untuk tugas-tugas terlalu berlebihan
- Material yang dibutuhkan untuk tugas-tugas terlalu berlebihan.

f. Services

Analisa untuk menilai kualitas dari suatu sistem dapat dilihat dari kriteria-kriteria berikut ini :

- Sistem menghasilkan produk yang tidak akurat
- Sistem menghasilkan produk yang tidak konsisten
- Sistem menghasilkan produk yang tidak dipercaya
- Sistem tidak mudah dipelajari
- Sistem tidak mudah digunakan
- Sistem canggung untuk digunakan
- Sistem tidak fleksibel

Berikut ini contoh hasil analisa PIECES yang dilakukan pada sistem informasi penggajian yang ada pada BDK (Balai Diklat Keagamaan) Medan

A. ANALISIS KINERJA (PERFORMANCE)

Adalah kemampuan didalam menyelesaikan tugas bisnis dengan cepat sehingga sasaran dapat segera tercapai. Kinerja diukur dengan jumlah produksi (*throughput*) dan waktu jawaban (*respon time*) dari suatu system.

System yang dikembangkan ini akan menyediakan *throughput* dan *respon time* yang memadai untuk kebutuhan manajemen pada BDK Medan.

◆ Kelayakan Teknis

- ⇒ Apakah teknologi yang dibutuhkan sudah tersedia?
- ⇒ Apakah teknologi yang akan digunakan ini dapat berintegrasi dengan teknologi yang sudah ada?
- ⇒ Apakah sistem yang sudah ada dapat dikonversikan ke sistem dengan teknologi baru?
- ⇒ Apakah organisasi memiliki orang yang menguasai teknologi baru ini?

◆ Kelayakan Operasional

- Aspek Teknis
 - ⇒ Apakah sistem dapat memenuhi tujuan organisasi untuk mendapatkan informasi yang dibutuhkan?
 - ⇒ Apakah sistem dapat diorganisasikan untuk menghasilkan informasi pada saat yang tepat untuk setiap orang yang membutuhkannya?
- Aspek (psikologis) penerimaan oleh orang-orang yang ada di dalam organisasi
 - ⇒ Apakah sistem baru memerlukan restrukturisasi organisasi dan bagaimana akibat restrukturisasi ini terhadap orang-orang yang ada di organisasi?
 - ⇒ Apakah diperlukan pelatihan atau pelatihan ulang?
 - ⇒ Apakah personil di dalam organisasi dapat memenuhi kriteria untuk sistem baru?

Analisa Net Present Value (NPV)

Present Value :

Nilai sekarang dari penerimaan (uang) yang akan didapat pada tahun mendatang.

Net Present Value :

Selisih antara penerimaan dan pengeluaran per tahun.

Discount Rate :

Bilangan yang digunakan untuk men-discount penerimaan yang akan didapat pada tahun mendatang menjadi nilai sekarang.

Discount rate dapat dilihat dari tabel discount rate yang ditentukan oleh tingkat suku bunga (i) dan tahun (t).

Discount rate pada tahun ke-5 dengan interest rate 10% adalah :
0,621

Untuk menghitung discount rate ini dapat digunakan rumus :

$$d = 1/(1+i)^t$$

d = discount rate

i = Interest rate

t = tahun

Return On Invesment

Return on invesment adalah besarnya keuntungan yang bisa diperoleh (dalam %) selama periode waktu yang telah ditentukan untuk menjalankan proyek, untuk menghitungnya digunakan rumus:

$$ROI = \frac{\text{TotalManfaat} - \text{TotalBiaya}}{\text{TotalBiaya}}$$

Jika dinyatakan dalam persen (%)

$$ROI = \frac{\text{Totalmanfaat} - \text{TotalBiaya}}{\text{TotalBiaya}} \times 100\%$$

Berikut ini contoh perhitungan analisa biaya dan manfaat untuk pengembangan sebuah sistem pembelajaran berbasis komputer di sebuah SMP untuk melengkapi sistem pembelajaran yang sudah ada:

Adapun metode-metode untuk melakukan analisis biaya dan manfaat diantaranya sebagai berikut:

1. Analisis Periode Pengembalian (*Payback Period*)
2. Analisis Pengembalian Investasi (*Return On Investment*)
3. Analisis Nilai Sekarang Bersih (*Net Present Value*)

Berdasarkan para rincian biaya dan manfaat pada tabel di atas, maka dapat dilakukan analisis biaya dan manfaat sebagai berikut:

1. Analisis *Payback Period*

Perhitungan analisis *Payback Periode* (waktu pengembalian investasi):

	Total Biaya Pengadaan Sistem pada tahun 0	: Rp. 25.000.000
	<i>Proceed</i> pada tahun I	: <u>Rp. 8.750.000</u>
(-)		
	Sisa Biaya Sistem pada tahun I	: Rp. 16.250.000
	<i>Proceed</i> pada tahun II	: <u>Rp. 11.337.500</u>
(-)		
	Sisa Biaya Sistem pada tahun II	: Rp. 4.912.500
	Sisa	= $\frac{4.912.500}{15.427.125} \times 1 \text{ tahun} = 0,31 \text{ tahun}$

Payback Periode sudah dapat dicapai pada tahun ke-3, secara detailnya adalah 2,31 tahun. Dari fakta tersebut, dapat disimpulkan bahwa yang ditanamkan pada rancangan sistem ini akan mencapai titik impas pada waktu 2,31 tahun, yang berarti bahwa pada tahun ke-3 mulai dapat mengambil keuntungan dari sistem tersebut.

3. Analisis *Net Present Value* (NPV)

Perhitungan Analisis NPV:

$$\text{NPV} = - \text{nilai proyek} + \frac{\text{Proceed 1}}{(1+i \%)^1} + \frac{\text{Proceed 2}}{(1+i \%)^2} + \dots + \frac{\text{Proceed n}}{(1+i)^n}$$

Apabila diaplikasikan pada penerapan sistem yang baru, maka perhitungan nilainya dengan tingkat bunga diskonto 15% per tahun adalah:

$$\text{NPV} = - 25.000.000 + \frac{8.750.000}{(1+0,15)^1} + \frac{11.337.500}{(1+0,15)^2} + \frac{15.427.625}{(1+0,15)^n}$$

$$\text{NPV} = 1.325.059,587$$

Dari hasil perhitungan diatas, dapat disimpulkan bahwa jumlah keuntungan yang diterima sekarang pada tahun ke-3 apabila sistem ini diterapkan adalah : Rp. 1.325.059,587 atau NPV lebih besar dari 0, maka proyek tersebut layak dilaksanakan.

BAB 7

PENENTUAN KEBUTUHAN SISTEM

7.1 System requirement (Kebutuhan Sistem)

Tujuan dari fase analisis adalah memahami dengan sebenar- benarnya kebutuhan dari sistem baru dan mengembangkan sebuah sistem yang mewadahi requirement tersebut-atau memutuskan bahwa sebenarnya pengembangan sistem baru tidak dibutuhkan. Penentuankebutuhan sistem merupakan langkah yang paling crucial dalam tahapan SDLC. Kebutuhan Sistem bisa diartikan sebagai berikut: Pernyataan tentang apa yang harus dikerjakan oleh sistem Pernyataan tentang karakteristik yang harus dimiliki sistem (Tim Politeknik Telkom Bandung, 2009)⁸

7.2 Tipe-tipe Kebutuhan Sistem

Kebutuhan Fungsional (Functional requirement)

Functional requirement adalah jenis requirement yang berisi: Proses-proses yang harus dilakukan oleh sistem Informasi-informasi yang harus ada di dalam sistem

Nonfunctional Requirements

Adalah tipe requirement yang berisi properti perilaku yang dimiliki oleh sistem, meliputi:

- ❖Operasional
- ❖Performance
- ❖Keamanan
- ❖Politik dan budaya

[8] Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung

Berikut ini contoh sebuah dokumen kebutuhan sistem untuk pengembangan sistem informasi perpustakaan berbasis komputer :

FUNCTIONAL REQUIREMENT

❖Sistem harus dapat melakukan entri buku yang berhubungan dengan pendataan buku:

- a. User bisa memasukkan berbagai jenis buku beserta dengan kode buku, kategori buku, judul buku, penerbit, pengarang, jumlah halaman, ISBN buku dan lain-lain
- b. User bisa menambahkan koleksi buku baru yang akan dikeluarkan
- c. User bisa menghitung berapa jumlah koleksi buku yang ada secara keseluruhan
- d. User dapat menampilkan berapa jenis atau kategori koleksi buku yang ada beserta dengan jumlahnya
- e. User dapat menampilkan jenis buku berdasarkan kategori tertentu beserta jumlah bukunya
- f. User dapat mencari koleksi buku berdasarkan pengarang, penerbit, judul, dan lain-lain
- g. User dapat menampilkan nama buku yang paling sering di pinjam
- h. User dapat mengganti beberapa kode buku dan lain-lain yang berhubungan dengan buku
- i. User dapat menghapus beberapa jenis buku, judul buku dan lain-lain yang sudah rusak atau hilang
- j. User dapat mengurutkan buku berdasarkan kodenya, judul buku, jumlah halaman, dan lain-lain
- k. User dapat menampilkan buku yang belum diberi kode
- l. User dapat mengelompokkan buku berdasarkan pengarang dan judul buku apa saja yang telah ditulis yang ada di perpustakaan

❖System harus dapat melakukan pendataan anggota

- a. User dapat memasukkan data anggota baru dengan memasukkan kode anggota, nama, alamat, nomor telepon
- b. User dapat menampilkan biaya pendaftaran anggota yang harus di bayarkan
- c. User dapat menampilkan print kartu anggota baru
- d. User dapat menampilkan menghitung berapa jumlah anggota yang ada secara keseluruhan
- e. User dapat menampilkan data jumlah anggota yang berada pada daerah kecamatan yang sama
- f. User bisa menambahkan data anggota baru

- g. User bisa merubah data anggota apabila alamat atau nomor telepon anggota berubah
- h. User dapat menampilkan anggota yang paling sering meminjam serta kategori buku yang di pinjam
- i. User dapat menampilkan data peningkatan atau penurunan anggota baru setiap tahunnya

❖Sistem harus dapat melakukan transaksi peminjaman:

- a. User dapat mencatat semua transaksi peminjaman
- b. User dapat memasukan data-data dari anggota baik nama, alamat, nomor telepon dan lain-lain
- c. User dapat memasukkan jumlah buku yang di pinjam
- d. User dapat memasukkan judul buku, nama pengarang, dan nama penerbit
- e. User dapat menampilkan tanggal kembali buku yang di pinjam
- f. User dapat mengetahui apakah anggota sudah mengembalikan buku yang di pinjam

- a. Waktu untuk transaksi peminjaman buku dibatasi 2 menit
- b. Waktu untuk transaksi pengembalian buku di batasi 1 menit
- c. Jumlah transaksi peminjaman dan pengembalian

7.3 Teknik Pengumpulan Requirement

Dalam menyusun requirement, ada beberapa teknik yang biasa digunakan. Pada uraian berikut kita akan membahas beberapa teknik pengumpulan requiremen yang biasa digunakan.

Interview

Interview atau wawancara adalah teknik pengumpulan requirement yang paling umum digunakan. Langkah-langkah dasar dalam teknik interview adalah:

- a. Memilih target interview
- b. Mendisain pertanyaan-pertanyaan interview
- c. Persiapan interview
- d. Melakukan interview
- e. Follow up hasil interview

Joint Application Development

Proses kelompok terstruktur yang terfokus untuk menentukan requiremen, melibatkan tim proyek, pengguna, dan manajemen bekerja bersama-sama, teknik ini sangat berguna untuk mereduksi waktu pengumpulan informasi sampai 50%.

Questionnaires (Kuisisioner)

Kuisisioner adalah sekumpulan pertanyaan tertulis, dan sering melibatkan banyak orang. Kuisisioner bisa dilakukan paper based atau secara elektronik. Biasanya sampel dipilih untuk mewakili populasi tertentu. Setelah hasil kuisisioner diperoleh diperlukan analisa untuk mengambil data yang sesuai dengan keperluan pengumpulan requirement.

Analisa Dokumen

Teknik ini dilakukan dengan mempelajari material yang menggambarkan sistem yang sedang berjalan. Biasanya dokumen yang diamati berupa form, laporan, manual kebijakan, grafik organisasi

Observasi

Teknik ini dilakukan dengan melakukan pengamatan secara langsung pada proses-proses yang sedang berjalan. Hal ini penting karena kadang-kadang user atau manager tidak dapat mengingat secara keseluruhan apa yang mereka lakukan dan menceritakan kembali ke analis.

BAB 8

ANALISA USE CASE

8.1 Analisa Use Case

Use case adalah metode berbasis teks untuk menggambarkan dan mendokumentasikan proses yang kompleks. Use case menambahkan detail untuk requirement yang telah dituliskan pada definisi sistem requirement. Use case dikembangkan oleh sistem analis bersama-sama dengan user. Pada tahapan selanjutnya berdasarkan use case ini analis menyusun model data dan model proses (Simon Bennet, 2002)⁹

8.2 Peranan Use case

Semua kemungkinan tanggapan terhadap suatu kejadian didokumentasikan. Use case sangat berguna ketika situasi yang dianalisa sangat kompleks. Sebuah use terdiri dari elemen-elemen sebagai berikut:

1. Informasi Dasar
 - a. Nama, jumlah dan deskripsi singkat
 - b. Trigger-kejadian yang menyebabkab adanya usecase
 - c. Eksternal trigger: trigger yang berasal dari luar sistem
 - d. Temporal Trigger: kejadian yang berbasis waktu
 - e. Sudut pandang use case harus konsisten
2. Input-output utama
 - a. Asal dan tujuan
 - b. Tujuan harus lengkap dan komprehensif

[9] Simon Bennet, Steve McRobb, Ray Farmer, 2002, *Object Oriented Systems Analysis and Design Using UML 2nd*, McGraw Hill

3. Detail

- a. Harus ada detail dari langkah-langkah yang harus dilakukan berikut data masukan dan keluarannya.

Proses pengembangan Use Case

1. Identifikasi use case utama
2. Identifikasi setiap langkah dalam setiap use case
3. Identifikasi elemen-elemen dalam setiap langkah
4. Konfirmasikan use case
5. Ulangi langkah-langkah diatas secara iteratif

Langkah 3: Identifikasi elemen-elemen dalam setiap langkah

Aktifitas	Pertanyaan tipikal yang diajukan
Untuk setiap langkah, identifikasi setiap langkah, input dan outputnya	Tanyakan How tentang setiap langkah ini: <ul style="list-style-type: none"> • Bagaimana orang mengetahui kapan melakukan langkah ini? • Laporan/form apa yang dihasilkan dari langkah ini? • Laporan/form apa yang dibutuhkan langkah ini? • Apa yang terjadi jika laporan/form ini tidak ada?

Langkah 4: Konfirmasikan use case

Aktifitas	Pertanyaan tipikal yang diajukan
Untuk setiap langkah, validasi bahwa semuanya lengkap dan benar	Minta user untuk mengeksekusi proses-proses yang tertulis dalam use case.

Berikut ini contoh use case yang diterapkan pada proses pemilihan CD pada sebuah rental CD:

BAB 9

PROCESS MODELLING

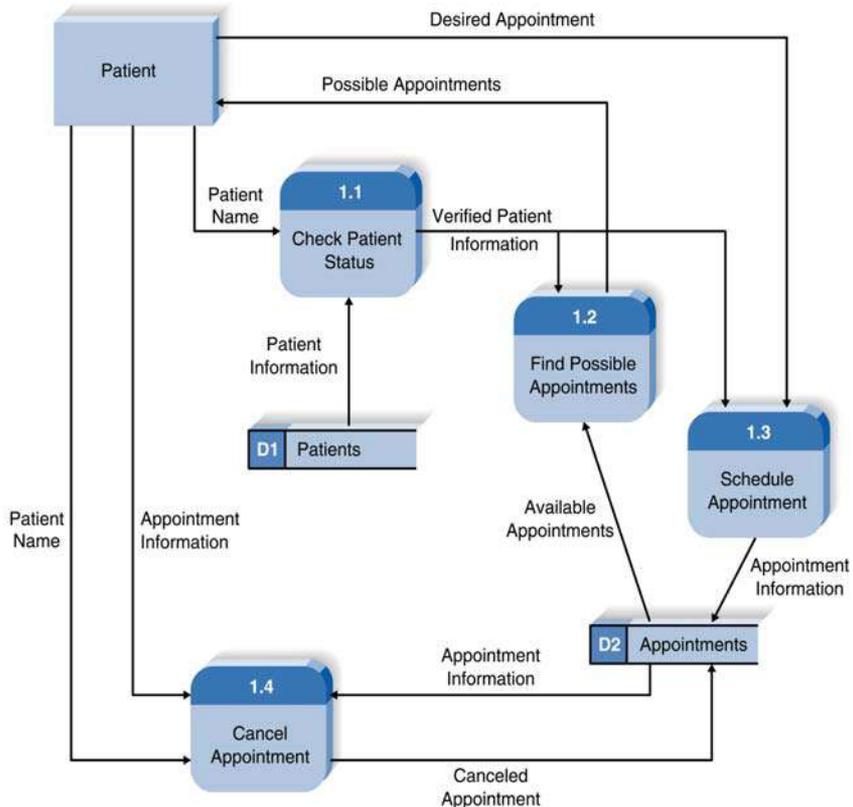
9.1 Process model

Proses modelling adalah cara formal untuk menggambarkan bagaimana bisnis beroperasi. Mengilustrasikan aktivitas-aktivitas yang dilakukan dan bagaimana data berpindah diantara aktifitas-aktifitas itu. Ada banyak cara untuk merepresentasikan proses model, cara yang populer adalah dengan menggunakan data flow diagram (DFD). Ada 2 jenis DFD, Logical DFD yang menggambarkan proses tanpa menyarankan bagaimana mereka akan dilakukan. Kedua adalah Fisikal DFD yang menggambarkan proses model berikut implementasi pemrosesan informasinya (Nugroho, Adi, 2005)¹⁰

9.2 DATA FLOW DIAGRAMS

Berikut ini contoh sebuah DFD yang menggambarkan proses pendaftaran pasien pada rumah sakit:

[10] Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Edisi: Revisi, Informatika, Bandung

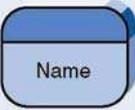
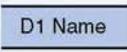


Gambar 9.1 DFD pendaftaran pasien pada rumah sakit

Untuk membaca suatu DFD kita harus memahami dulu , elemen-elemen yang menyusun suatu DFD. Ada 4 elemen yang menyusun suatu DFD yaitu:

Proses

Aktivitas atau fungsi yang dilakukan untuk alasan bisnis yang spesifik, biasa berupa man ual maupun terkomputerisasi.

Data Flow Diagram Element	Typical Computer-Aided Software Engineering Fields	Gane and Sarson Symbol	DeMarco and Yourdan Symbol
Every <i>process</i> has A number A name (verb phrase) A description One or more output data flows Usually one or more input data flows	Label (name) Type (process) Description (what is it) Process number Process description (Structured English) Notes		
Every <i>data flow</i> has A name (a noun) A description One or more connections to a process	Label (name) Type (flow) Description Alias (another name) Composition (description of data elements) Notes		
Every <i>data store</i> has A number A name (a noun) A description One or more input data flows Usually one or more output data flows	Label (name) Type (store) Description Alias (another name) Composition (description of data elements) Notes		
Every <i>external entity</i> has A name (a noun) A description	Label (name) Type (entity) Description Alias (another name) Entity description Notes		

Gambar 9.2 Elemen-elemen dari DFD dan lambangnya

9.3 Menggambarkan proses bisnis dengan menggunakan DFD

Proses bisnis biasanya terlalu kompleks untuk ditunjukkan dalam 1 DFD. Dekomposisi adalah proses untuk

menggambarkan sistem dalam hirarkie dari diagram DFD. Diagram anak menggambarkan proses yang lebih detail dibandingkan dengan diagram induk. Harus ada proses Balancing untuk menjamin informasi yang disajikan dalam satu level dari suatu DFD secara akurat direpresentasikan pada DFD level berikutnya. Adapun hirarki dari suatu DFD dapat dilihat pada diagram berikut ini:

Context Diagram: DFD pertama dalam proses bisnis. Menunjukkan konteks dimana proses bisnis berada. Menunjukkan semua proses bisnis dalam 1 proses tunggal (proses 0). Context Diagram juga menunjukkan semua entitas luar yang menerima informasi dari atau memberikan informasi ke sistem.

Level 0 Diagram: Menunjukkan semua proses utama yang menyusun keseluruhan sistem. Level ini juga menunjukkan komponen internal dari proses 0 dan menunjukkan bagaimana proses-proses utama direlasikan menggunakan data flow. Pada level ini juga ditunjukkan bagaimana proses-proses utama terhubung dengan entitas eksternal. Pada level ini juga dilakukan penambahan data store.

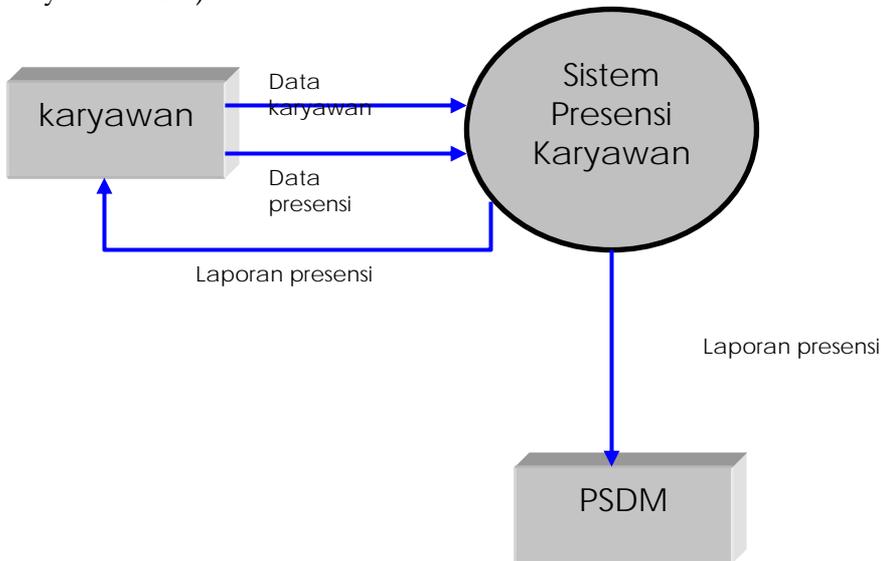
Level 1 Diagrams: Umumnya diagram level 1 diciptakan dari setiap proses utama dari level 0. Level ini menunjukkan proses-proses internal yang menyusun setiap proses-proses utama dalam level 0, sekaligus menunjukkan bagaimana informasi berpindah dari satu proses ke proses yang lainnya. Jika misalnya proses induk di pecah, katakanlah menjadi 3 proses anak, maka 3 proses anak ini secara utuh menyusun proses induk.

Level 2 Diagrams: Menunjukkan semua proses yang menyusun sebuah proses pada level 1. Bisa saja penyusunan DFD tidak mencapai level 2 ini. Atau mungkin harus dilanjutkan ke level berikutnya (level 3, level 4...dst)

Berikut ini contoh penyusunan DFD untuk sistem presensi karyawan yang menggunakan webcam untuk meng-capture wajah karyawan yang melakukan presensi. Sistem ini bekerja dengan prinsip pengenalan wajah. Perhatikan bahwa diagram yang dipakai adalah diagram yang mengacu pada metode De Marco-Jordan.

Context Diagram

Pada context diagram, sistem digambarkan dengan sebuah proses saja, kemudian diidentifikasi entitas luar yang berinteraksi dengan proses tunggal tadi. Didapatkan 2 entitas luar yaitu karyawan dan bagian PSDM (Personalia dan Sumber Daya Manusia).



Gambar 9.4. context diagram untuk sistem presensi

Keterangan Gambar :

Karyawan : entitas luar karyawan FST-UIN Sumatera Utara
Medan yang diwajibkan melakukan presensi

PSDM : entitas luar Personalia dan Sumber Daya Manusia

Data Karyawan: Meliputi No induk, Nama, Bagian, Jabatan, Gaji

Data Presensi: Meliputi wajah karyawan yang melakukan presensi

Laporan presensi: Laporan presensi karyawan perbulan berisi no induk, nama, bulan, tahun dan total potongan.

DFD Level 1

Pada level ini, proses tunggal dari context diagram dipecah menjadi 3 proses yang lebih terperinci yaitu proses input data karyawan, presensi dan pembuatan laporan presensi.

Data Presensi: Meliputi wajah karyawan yang melakukan presensi

Data Presensi2: Meliputi id presensi, citra wajah karyawan, tanggal dan waktu presensi dilakukan.

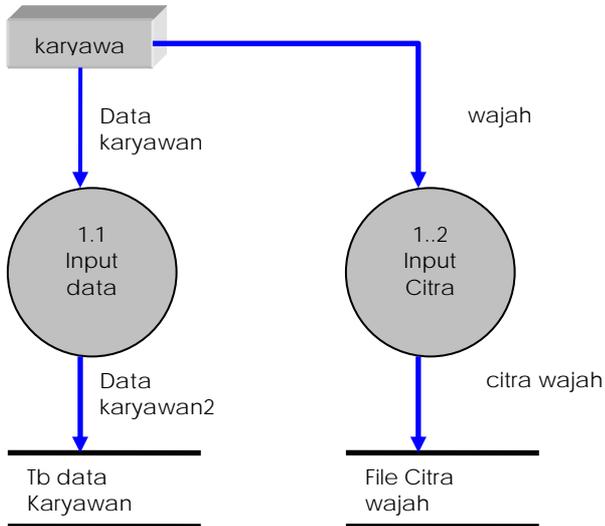
Tb Karyawan : Tabel Karyawan, untuk menyimpan data karyawan

Tb Presensi : Tabel Presensi untuk menyimpan data presensi

Laporan presensi: Laporan presensi karyawan perbulan berisi no induk, nama, bulan, tahun dan total potongan.

DFD Level 2

Pada DFD Level 2, proses Input data Karyawan diperinci menjadi 2 proses yaitu proses input data, digunakan untuk menginputkan data karyawan dan proses input citra, proses meng-capture wajah karyawan sehingga didapatkan citra wajah. Pada Level ini didapatkan juga beberapa penyimpanan data yaitu Tb Data Karyawan untuk menyimpan data-data karyawan dan File Citra Wajah Karyawan untuk menyimpan citra wajah karyawan.



Gambar 9.6 DFD Level 2 untuk proses Input data Karyawan

Untuk proses Presensi, dapat diperinci menjadi proses input citra yang menerima input wajah dari karyawan yang dicapture dengan webcam, Image matching yang bertugas mencari citra wajah yang sesuai pada database dan Transaksi presensi yang memproses data-data saat transaksi karyawan dilakukan.

BAB 10

DATA MODELLING

10.1 DATA MODEL

Data model adalah cara formal untuk menggambarkan data yang digunakan dan diciptakan dalam suatu sistem bisnis. Model ini menunjukkan orang, tempat atau benda dimana data diambil dan hubungan antar data tersebut. Data modelling juga dibedakan menjadi 2: *Logical data model*: Menunjukkan pengaturan data tanpa mengindikasikan bagaimana data tersebut disimpan, dibuat dan dimanipulasi. *Physical data model*: Menunjukkan bagaimana data akan disimpan sebenarnya dalam database atau file. Penyusunan model data harus seimbang dengan model proses. Salah satu cara pemodelan data adalah dengan ERD (Entity Relationship Diagram) (Nugroho, Adi, 2005)¹¹

10.2 THE ENTITY-RELATIONSHIP DIAGRAM (ERD)

Apakah ERD itu?

- Gambar yang menunjukkan informasi dibuat disimpan dan digunakan dalam sistem bisnis.
- Entitas biasanya menggambarkan jenis informasi yang sama
- Garis yang menghubungkan antar entitas menunjukkan hubungan antar data

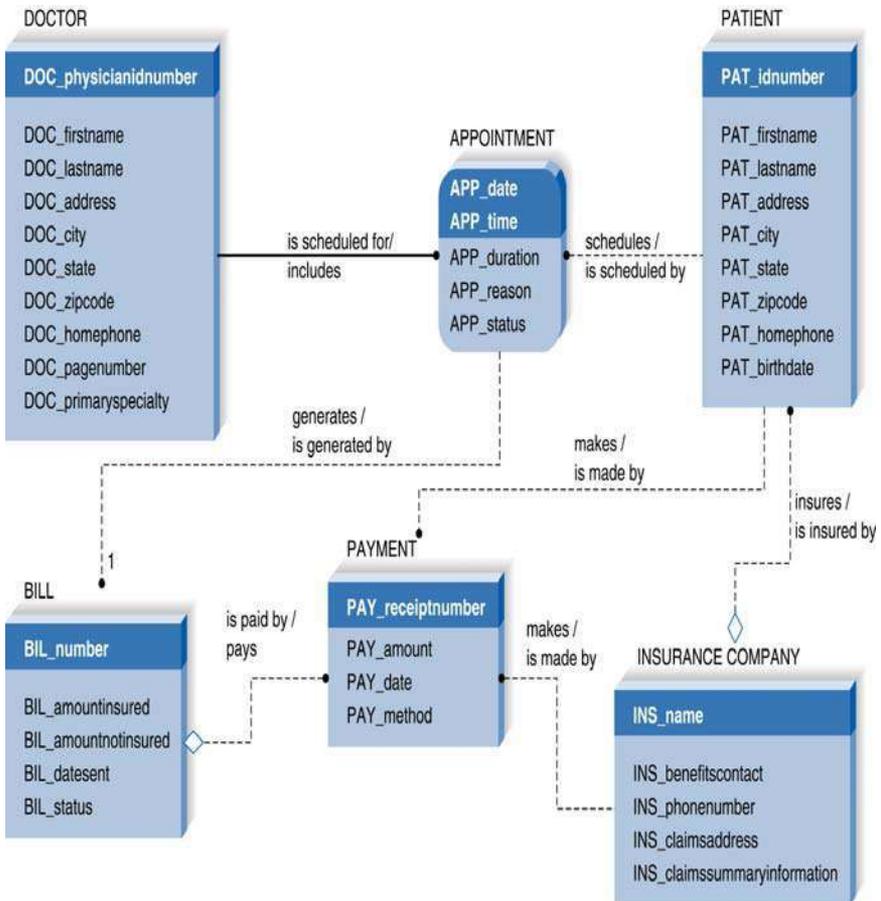
[11] Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Edisi: Revisi, Informatika, Bandung

- ERD juga bisa digunakan untuk menunjukkan aturan-aturan bisnis

Menggunakan ERD untuk menunjukkan aturan bisnis

- Aturan bisnis adalah batasan yang harus diikuti ketika sistem beroperasi.
- Simbol ERD hanya menunjukkan satu instance dari entitas harus ada sebelum instance lain dari suatu entitas. Sebagai contoh: Seorang dokter harus ada sebelum perjanjian ketemu dengan dokter dibuat.
- Simbol ERD dapat menunjukkan ketika salah satu instance dari suatu entitas dapat direlasikan dengan satu atau lebih instance dari entitas lainnya. Contohnya: satu dokter bisa memiliki banyak pasien, satu pasien bisa jadi hanya memiliki satu dokter utama.
- Simbol ERD juga menunjukkan ketika eksistensi dari instance suatu entity adalah opsional untuk sebuah relasi dengan instance dari suatu entitas. Contohnya: Pasien mungkin memiliki atau mungkin tidak memiliki biaya asuransi.

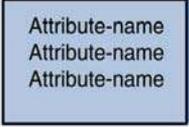
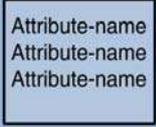
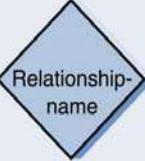
Berikut ini contoh sebuah ERD



Gambar 10.1 Contoh Diagram ERD

Elemen-elemen ERD

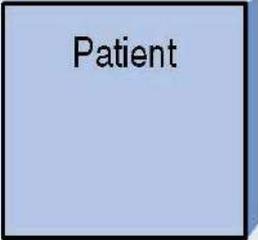
Seperti data flow diagram, ERD juga menggunakan simbol-simbol khusus untuk menggambarkan elemen-elemen ERD. Berikut ini simbol-simbol yang digunakan dalam ERD:

	IDEFIX	Chen	Information Engineering
<p>An ENTITY:</p> <ul style="list-style-type: none"> ✓ Is a person, place, or thing ✓ Has a singular name spelled in all capital letters ✓ Has an identifier ✓ Should contain more than one instance of data 	<p>ENTITY-NAME</p> 	<p>ENTITY-NAME</p> 	<p>ENTITY-NAME</p> 
<p>An ATTRIBUTE:</p> <ul style="list-style-type: none"> ✓ Is a property of an entity ✓ Should be used by at least one business process ✓ Is broken down to its most useful level of detail 	<p>ENTITY-NAME</p> 		<p>ENTITY-NAME</p> 
<p>A RELATIONSHIP:</p> <ul style="list-style-type: none"> ✓ Shows the association between two entities ✓ Has a parent entity and a child entity ✓ Is described with a verb phrase ✓ Has cardinality (1 : 1, 1 : N, or M : N) ✓ Has modality (null, not null) ✓ Is dependent or independent 	<p><u>Relationship-name</u></p>		<p><u>Relationship-name</u></p>

Gambar 10.2 Elemen-elemen dari ERD

Keterangan :**Entity**

Entitas bisa berupa orang, kejadian, atau benda dimana data akan dikumpulkan. Untuk menjadi sebuah entity, suatu objek harus menampilkan beberapa kali event. Sebagai contoh : Jika sebuah firma hanya memiliki 1 gudang, maka gudang tersebut bukan entitas. Tetapi jika perusahaan memiliki banyak gudang, maka gudang bisa menjadi entitas suatu entitas jika perusahaan ingin menyimpan data untuk setiap instance dari gudang.

Entity	Example Instances
 A light blue rectangular box with a black border, containing the word "Patient" in black text.	John Smith Susan Jones Peter Todd Dale Turner Pat Turner

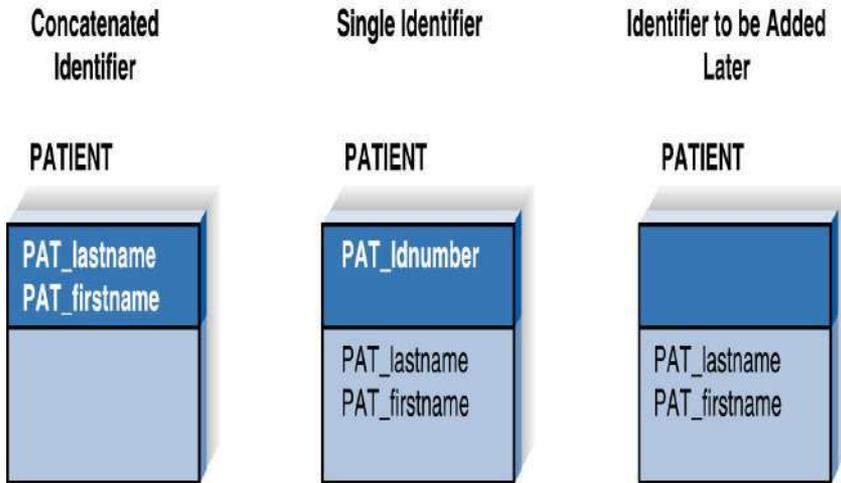
Gambar 10.3 Contoh Entitas dan Instance

Atribut

- Informasi yang diambil tentang sebuah entitas
- Hanya yang digunakan oleh organisasi yang dimasukkan dalam model
- Nama attribute harus merupakan kata benda
- Kadang nama entitas diletakkan di depan nama atribut untuk ketelitian.

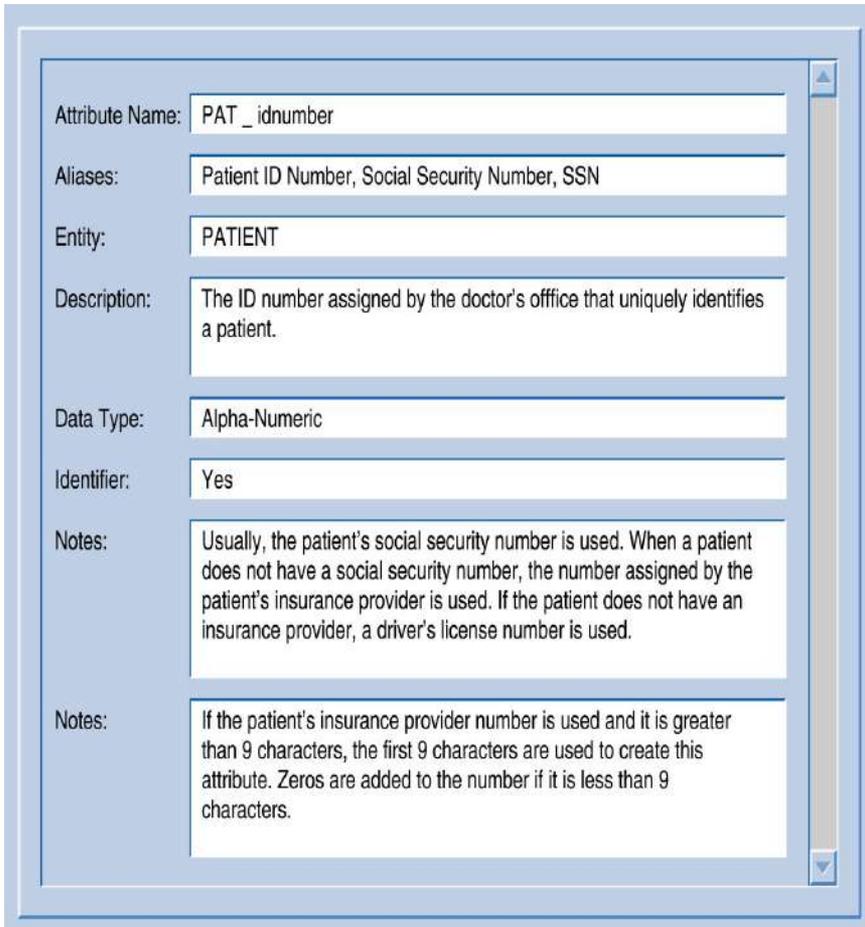
Identifier

- Satu atau lebih atribut dapat menjadi identifier entitas, yang secara unik mengidentifikasi setiap instance dari entitas.
- Concatenated identifier (identifier gabungan) terdiri dari beberapa atribut.
- Identifier bisa saja artifisial, seperti dengan membuat ID number
- Identifier tidak akan dikembangkan sampai fase desain.



Gambar 10.4 tipe-tipe identifier

Berikut ini contoh penggunaan identifier dalam sebuah case repository untuk suatu atribut



The image shows a software window titled "Case repository" for the attribute "PAT_idnumber". The window has a light blue border and a vertical scrollbar on the right side. It contains several fields with labels and text:

Attribute Name:	PAT_idnumber
Aliases:	Patient ID Number, Social Security Number, SSN
Entity:	PATIENT
Description:	The ID number assigned by the doctor's office that uniquely identifies a patient.
Data Type:	Alpha-Numeric
Identifier:	Yes
Notes:	Usually, the patient's social security number is used. When a patient does not have a social security number, the number assigned by the patient's insurance provider is used. If the patient does not have an insurance provider, a driver's license number is used.
Notes:	If the patient's insurance provider number is used and it is greater than 9 characters, the first 9 characters are used to create this attribute. Zeros are added to the number if it is less than 9 characters.

Gambar 10.5 Case repository untuk atribut patient_SSN

Relationships

- Hubungan antar entitas
- Entitas pertama dalam relationship disebut entitas induk, entitas kedua disebut sebagai entitas anak.
- Relationship harus mempunyai nama yang berupa kata kerja
- Relationship berjalan 2 arah

Sebagai contoh, jika dimiliki dua entitas yaitu buku dan toko buku maka bisa dibuat beberapa relationship, diantaranya:

- Toko buku memesan buku
- Toko buku menampilkan buku
- Toko buku menstock buku
- Toko buku menjual buku
- Toko Buku mengembalikan buku.

Relationship *memesan*, *menampilkan*, *menstock*, *menjual* dan *mengembalikan* mendefinisikan hubungan yang relevan antara buku dan toko buku.

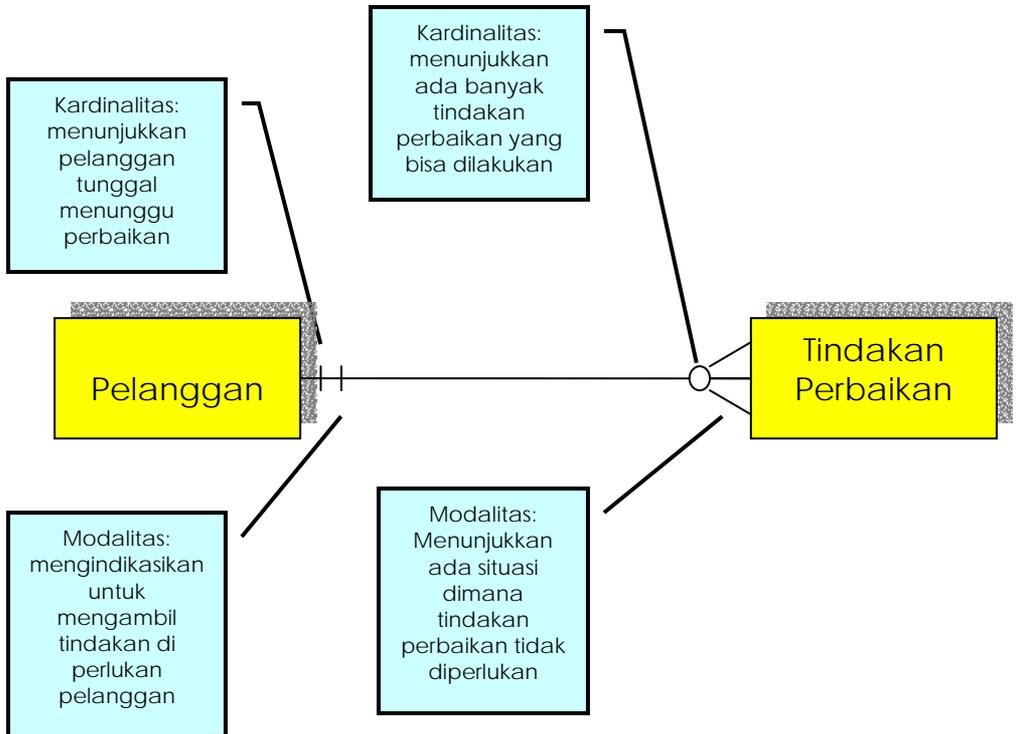
Cardinalitas

- Kardinalitas mengacu pada berapa kali instance dari satu entitas dapat berelasi dengan instance lain di entitas yang berbeda.
- Satu instance dalam 1 entitas mengacu pada satu dan hanya satu instance pada entitas lainnya (1:1)
- Satu instance dalam suatu entitas mengacu ke satu atau lebih instance yang berelasi (1:N)
- Satu atau lebih instance dalam suatu entitas mengacu pada satu atau lebih instance pada entitas yang berelasi (M:N).

Modalitas

- Mengacu pada apakah suatu instance dari entitas anak dapat ada tanpa suatu relasi dengan instance dari entitas induk atau tidak.
- **Not Null** berarti bahwa suatu instance pada entitas yang berelasi harus ada untuk suatu instance dari entitas lain untuk disebut valid
- **Null** berarti bahwa tidak ada instance dalam entitas yang berelasi yang diperlukan untuk instance pada relasi lain untuk dikatakan valid.

Berikut ini contoh penggunaan kardinalitas dan modalitas dalam suatu Relationship:



Data Dictionary dan Metadata

- Metadata adalah informasi yang tersimpan yang berisi komponen dari model data
- Metadata disimpan dalam data dictionary sehingga bisa dibagi dengan developer dan user melalui SDLC
- Data dictionary yang lengkap dan bisa dibagi membantu meningkatkan kualitas dari sistem yang sedang dikembangkan.

berikut ini frame umum dari sebuah metadata:

Sebuah entitas yang bisa dideskripsikan sebagai:	Sebuah Atribut yang bisa dideskripsikan sebagai:	Sebuah Relationship yang bisa dideskripsikan sebagai:
Nama Definisi Catatan Khusus Kontak User Kontak Analis	Nama Deskripsi Alias Contoh nilai Nilai yang bisa diterima Format Tipe Catatan khusus	Frase kata kerja Entitas induk Entitas anak Definisi Kardinalitas Modalitas

Gambar 10.6 Frame Metadata

MEMVALIDASI ERD

Untuk membuat ERD, diperlukan latihan dan jam terbang, Ada beberapa pedoman yang perlu diperhatikan untuk membuat ERD diantaranya:

- Entitas harus memiliki banyak kejadian/realitas
- Hindari penggunaan atribut yang tidak perlu
- Berilah label yang jelas untuk semua komponen
- Pasangkan kardinalitas dan modalitas yang jelas dan benar
- Pecah atribut menjadi level serendah mungkin yang diperlukan

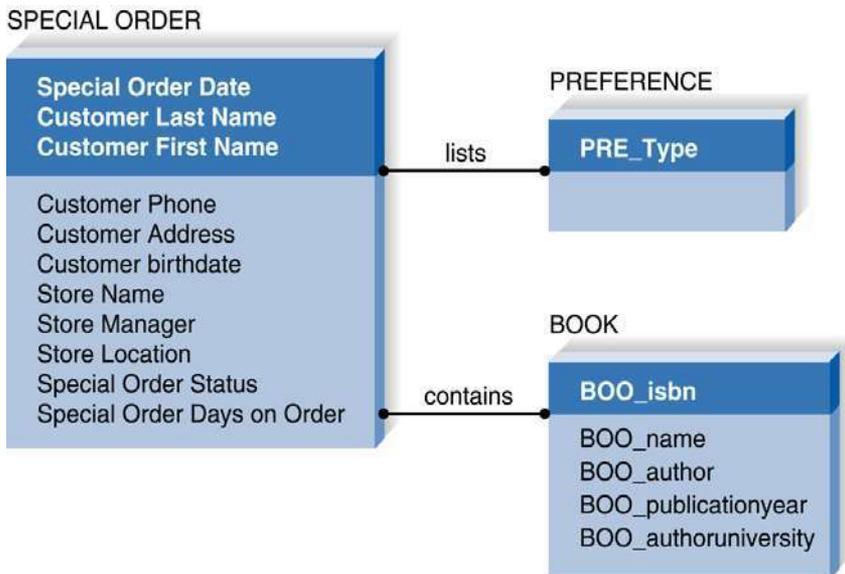
SPECIAL ORDER

Special Order Date
Customer Last Name
Customer First Name

Customer phone
Customer Address
Customer birthdate
Customer Book Preferences
Book ISBN1
Book Name1
Book Author1
Book Publication Year1
Book Author University1
Book ISBN2
Book Name2
Book Author2
Book Publication Year2
Book Author University2
Book ISBN3
Book Name3
Book Author3
Book Publication Year3
Book Author University3
Store Name
Store Manager
Store Location
Special Order Status
Special Order Days on Order

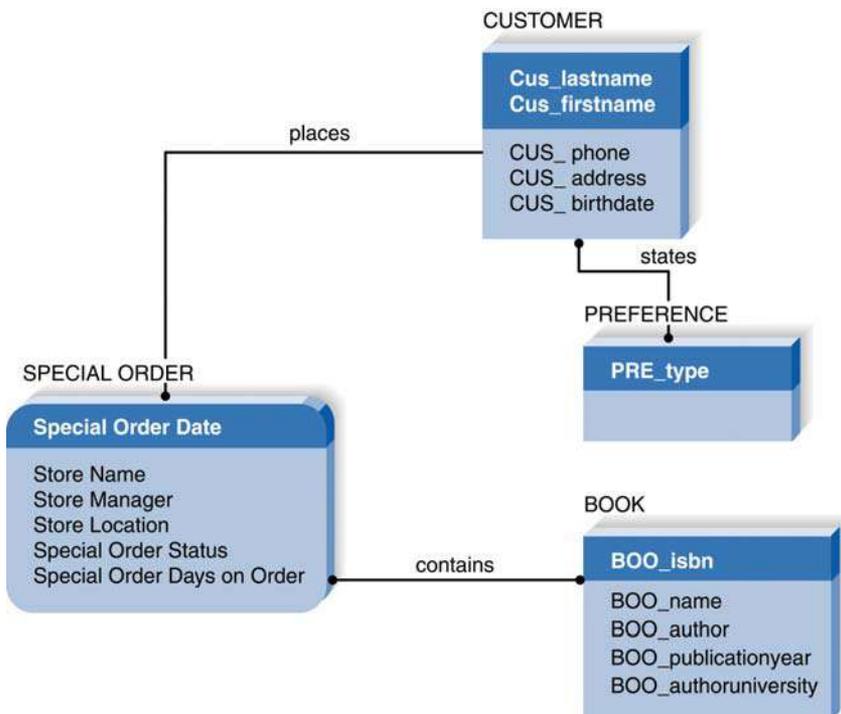
First Normal Form (1NF)

Cari kelompok-kelompok entitas yang berulang dan pisahkan ke dalam entitas yang berbeda.



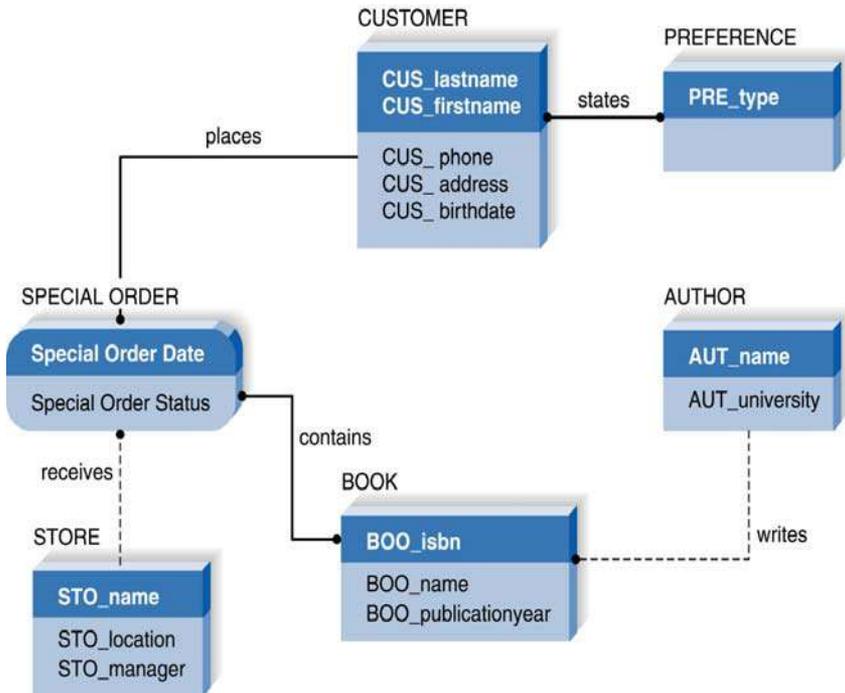
Second Normal Form (2NF)

Jika ada entitas yang memiliki identifier gabungan, cari atribut yang hanya bergantung pada identifier. Jika ditemukan pindahkan ke entitas baru.



Third Normal Form (3NF)

Cari atribut yang bergantung hanya pada atribut lain yang bukan merupakan identifier. Jika ditemukan pindahkan menjadi entitas baru, juga pindahkan atribut-atribut yang dirasa perlu dipindahkan.



Menyeimbangkan ERD dengan DFD

Semua aktifitas analisa merupakan aktifitas-aktifitas yang saling berkaitan, termasuk proses modelling dan data modelling. Proses model akan berisi dua hal data flow dan data store. Komponen data dalam DFD ini harus diseimbangkan dengan ERD dimana Data store diseimbangkan dengan entitas dan elemen data diseimbangkan dengan atribut. Untuk mempermudah, banyak tool CASE yang menyediakan fitur untuk mengecek ketidakseimbangan.

BAB 11

DESAIN SISTEM

11.1 FASE DESAIN

Fase desain adalah fase dimana didefinisikan bagaimana akan membangun sistem. Pada tahapan ini detail dari deskripsi teknis tentang bagaimana membangun sistem didefinisikan. Biasanya orang mengatakan hasil dari fase desain adalah spesifikasi sistem. Spesifikasi sistem adalah produk akhir dari fase desain, yang menjelaskan dengan tepat perancangan sistem seperti apa yang akan diimplementasikan oleh tim desain selama fase implementasi (Nugroho, Adi, 2005)¹².

Langkah-langkah Fase Desain:

Adapun langkah-langkah fase desain dapat dijelaskan sebagai berikut:

- Menyajikan alternatif desain (membuat sendiri, membeli atau outsourcing)
- Mengubah proses-proses logik dan model data ke dalam model fisik
- Merancang arsitektur sistem
- Membuat pemilihan hardware dan software
- Merancang bagaimana data akan disimpan
- Mendesain program untuk proses yang bersesuaian
- Membuat spesifikasi sistem

[12] Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Edisi: Revisi, Informatika, Bandung

Dalam fase desain sebaiknya dihindari kesalahan-kesalahan klasik seperti :

- Pengurangan waktu perancangan
- Penambahan fitur dari apa yang sudah ditentukan dalam fase analisis
- Mengganti tool dan teknologi pada pertengahan proyek.

11.2 STRATEGI DESAIN

Ada 3 strategi umum yang biasa dipakai dalam pengembangan sistem:

- *Custom development* : membangun sistem dari nol dalam rumah produksi sendiri
- *Membeli paket software* dan melakukan beberapa kostumisasi
- Melakukan *outsourcing* ke pihak ketiga.

Costum development

Strategi desain ini punya satu keunggulan utama, yaitu biaya pengembangannya dapat kita kendalikan sendiri dan relatif mudah. Rincian kelebihan dan keunggulannya adalah :

Kelebihan:

- Tingkat fleksibilitas dan krearifitas lebih tinggi

- Cocok dan konsisten dengan teknologi dan standar yang ada
- Bisa membangun kemampuan dan pengetahuan fungsional dari tenaga kerja yang kita miliki.

Kekurangan :

- Membutuhkan waktu dan usaha yang signifikan
- Mungkin membutuhkan keahlian atau pengetahuan fungsional yang belum dikuasai
- Kadang-kadang biaya bisa membengkak
- Sering membutuhkan waktu lebih banyak dari jadwal yang telah tersedia.
- Ada resiko kegagalan proyek.

Pembelian software Paket

Saat ini telah tersedia berbagai software paket untuk banyak kebutuhan bisnis. Dalam paket software, program yang digunakan telah dites dan diujikan serta bisa menghemat waktu dan biaya pengembangan. Tetapi software paket sangat jarang sesuai 100% dengan kebutuhan bisnis. Untuk mengatasi ini beberapa software paket memungkinkan user untuk melakukan kostumisasi. Kelemahan utamanya adalah paket software bisa jadi tidak bisa diintegrasikan dengan sistem yang sudah ada jika dikembangkan oleh perusahaan yang berbeda.

Outsourcing

Outsourcing adalah menyewa vendor luar, developer atau service provider untuk menyelesaikan tahapan desain . Keuntungan utama adalah mengurangi waktu produksi dan ada nilai tambah diantaranya bisa menerima lebih banyak proyek dalam periode yang sama dan melakukan outsourcing pada pihak-pihak ketiga. Tetapi strategi ini ada beberapa risikonya, diantaranya:

- Kehilangan informasi penting
- Kehilangan kendali akan pengembangan di masa depan
- Kehilangan kesempatan untuk belajar

Beberapa pedoman yang perlu diperhatikan untuk melakukan outsourcing adalah:

- Pelihara jalur komunikasi tetap terbuka dengan pihak ketiga.
- Definisikan dan stabilisasi requirement sebelum menandatangani kontrak
- Pandang kerjasama sebagai hubungan partnership
- Pilihlah vendor, developer dan service provider dengan hati-hati
- Tugaskan seseorang untuk memelihara kerjasama
- Jangan melakukan outsourcing apa yang tidak kita mengerti
- Tekankan pada requirement yang fleksibel, hubungan jangka panjang dan kontrak jangka pendek.

Memilih strategi desain

Dalam memilih strategi desain, pertimbangkan hal-hal berikut ini :

- Kebutuhan bisnis
- Pengalaman sendiri
- Kemampuan proyek
- Manajemen proyek
- Frame waktu yang tersedia

Untuk lebih jelasnya, perhatikan tabel berikut ini :

	Custom Development	Software Paket	Outsourcing
Business need	kebutuhan bisnis unik	kebutuhan bisnis umum	Kebutuhan bisnis bukan core/inti dari bisnis
In-house experience	Tenaga terlatih dan berkemampuan baik banyak dimiliki	Ikemampuan standar untuk kostumisasi dimiliki	SDM dengan kemampuan teknis yang terlatih tidak dimiliki
Project skills	Punya keinginan ingin meningkatkan Skill SDM yang dimiliki	Skill bukan hal atau isu yang strategis	Outsourcing adalah keputusan strategis
Project management	Memiliki manajer proyek yang handal dan teknologi yang telah terbukti	Manajer proyek bisa mengkoordinasikan usaha-usaha dari vendor	Proyek manajer berkemampuan tinggi pada level manajemen tertentu level
Time frame	Waktu yang tersedia fleksibel	Waktu yang tersedia pendek	Waktu yang tersedia bisa pendek bisa panjang

BAB 12

DISAIN ARSITEKTUR

12.1 Desain Arsitektur

Desain arsitektur adalah perencanaan bagaimana sistem akan di distribusikan di antara komputer-komputer yang ada dan software dan hardware apa yang akan digunakan untuk masing-masing komputer. Dalam desain arsitektur ada 2 hal yang perlu diperhatikan (Nugroho, Adi, 2005)¹³.

- Spesifikasi dari software dan hardware
- Deskripsi detail dari komponen software/hardware untuk mempermudah pihak yang akan membeli software/hardware tersebut.

12.2 Elemen dari desain arsitektur

Dari sudut pandang software komponen arsitektural dapat dibagi menjadi:

Data Storage : komponen yang digunakan untuk menyimpan data

Data access logic: prosedur yang digunakan untuk mengakses data yang disimpan dalam data storage.

Application Logic: bagian logika pemrosesan dalam lapis aplikasi

Presentation Logic : komponen pemrosesan tampilan dan perintah dari user.

[13] Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek*, Edisi: Revisi, Informatika, Bandung

12.3 Tujuan desain arsitektur

Tujuan dari desain arsitektur adalah untuk menentukan komponen software mana yang akan di-assign ke hardware yang tersedia. Adapun pilihan hardware yang tersedia adalah:

Klien :

Input/output device yang dioperasikan oleh user
PC, Laptop, peralatan mobile, telepon selular

Pilihan untuk Klien:

- Terminal
- Microcomputer
- Minicomputer

Server:

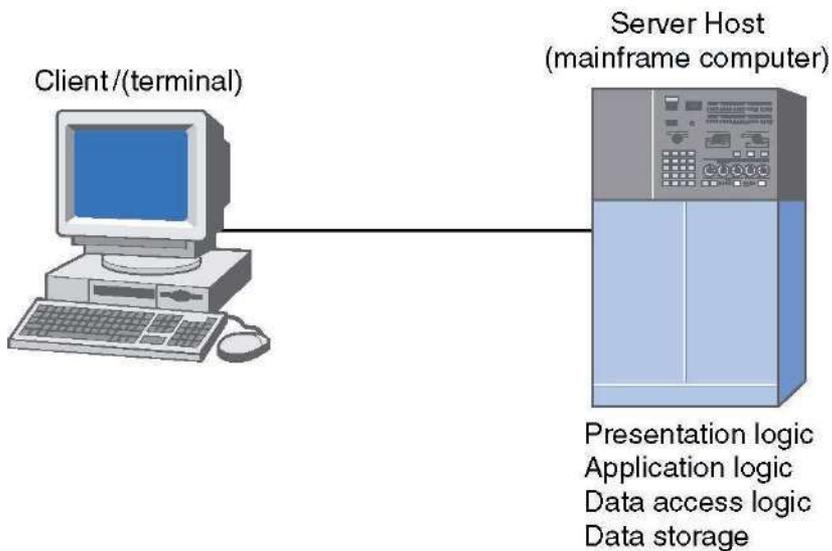
Komputer lebih canggih untuk menyimpan software
Bisa diakses oleh banyak user.

Pilihan untuk server :

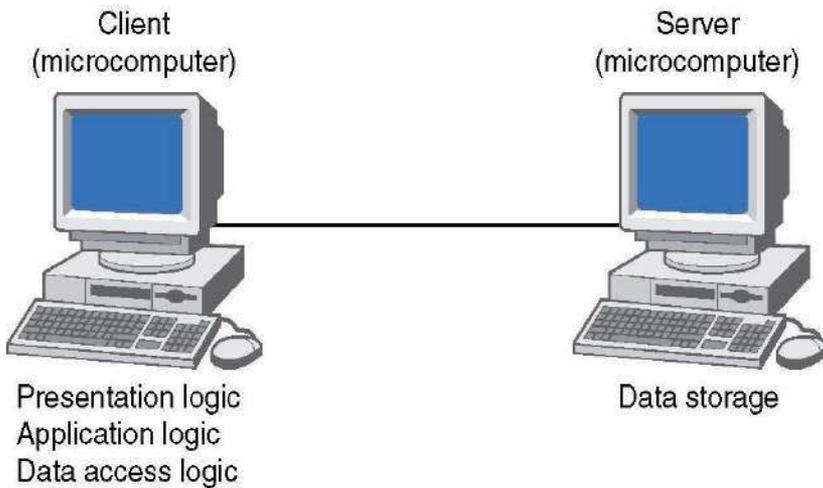
- Mainframe
- Microcomputer (PC)
- Terminal khusus seperti ATM, kiosk dll

12.4 Pilihan arsitektur

a. Server-Based Architecture

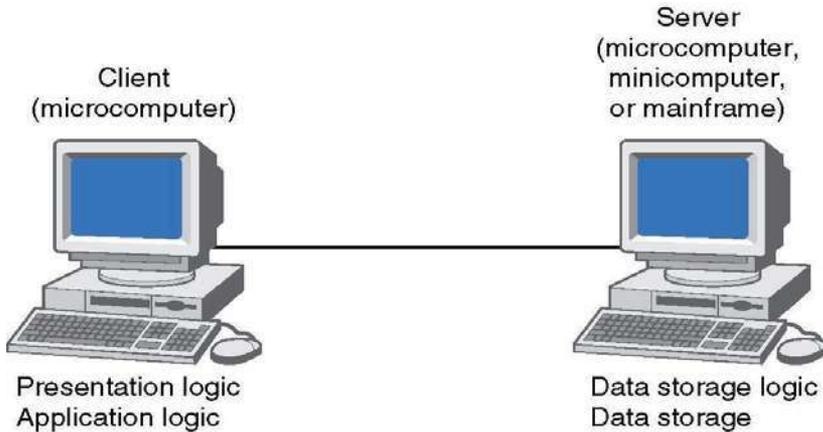


Pada arsitektur jenis ini, 4 komponen software diletakkan pada bagian yang sama yaitu server. Terminal yang tersedia hanya digunakan sebagai dumb terminal, perpanjangan layanan dari server saja. Pada arsitektur ini data relatif aman tetapi beban kerja server sangat berat, karena semua komponen dijalankan di server.

b. Client-Based Architecture

Pada arsitektur ini, server hanya berperan sebagai data storage. Komputer klien yang digunakan untuk mengolah data yang dikirim oleh server. Beban kerja server sangat ringan tetapi keamanan data sangat tergantung kinerja dari klien.

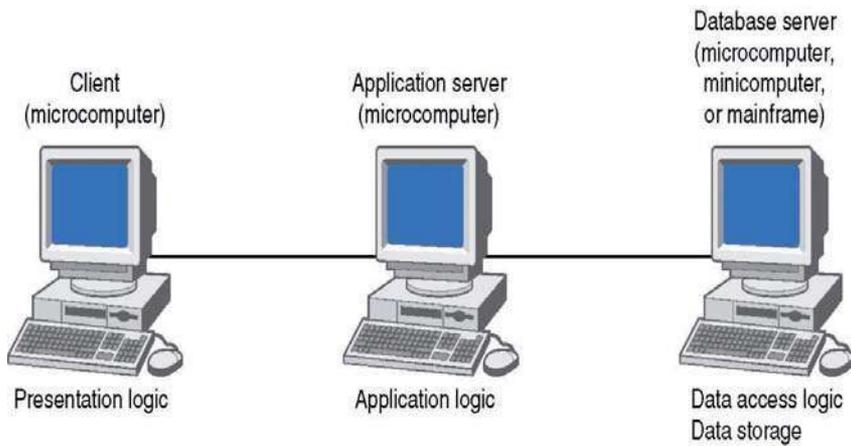
c. Client-Server Architecture (Two-Tiered)



Pada arsitektur ini data dan prosedur pengaksesan data dilayani oleh server dan masalah tampilan dan logika aplikasi dilayani oleh user. Pada arsitektur ini keamanan data cukup terjamin karena prosedur bagaimana mengakses data oleh klien ditangani oleh server. Arsitektur ini biasa disebut arsitektur 2-tier.

- ❑ Keunggulan
 - Scalable
 - Meningkatkan modularitas dari sistem berbasis web
 - Tidak ada titik pusat kegagalan
- ❑ Kelemahan
 - Kompleksitas
 - Perlu bahasa dan teknik pemrograman baru
 - Lebih kompleks untuk di update

d. Three-Tiered Client-Server Architecture



Pada arsitektur ini ditambahkan hardware untuk menjalankan lapis application logic, sehingga keamanan pengaksesan data menjadi lebih terjamin.

BAB 13

HIRARCHY INPUT OUTPUT CHART (HIPO)

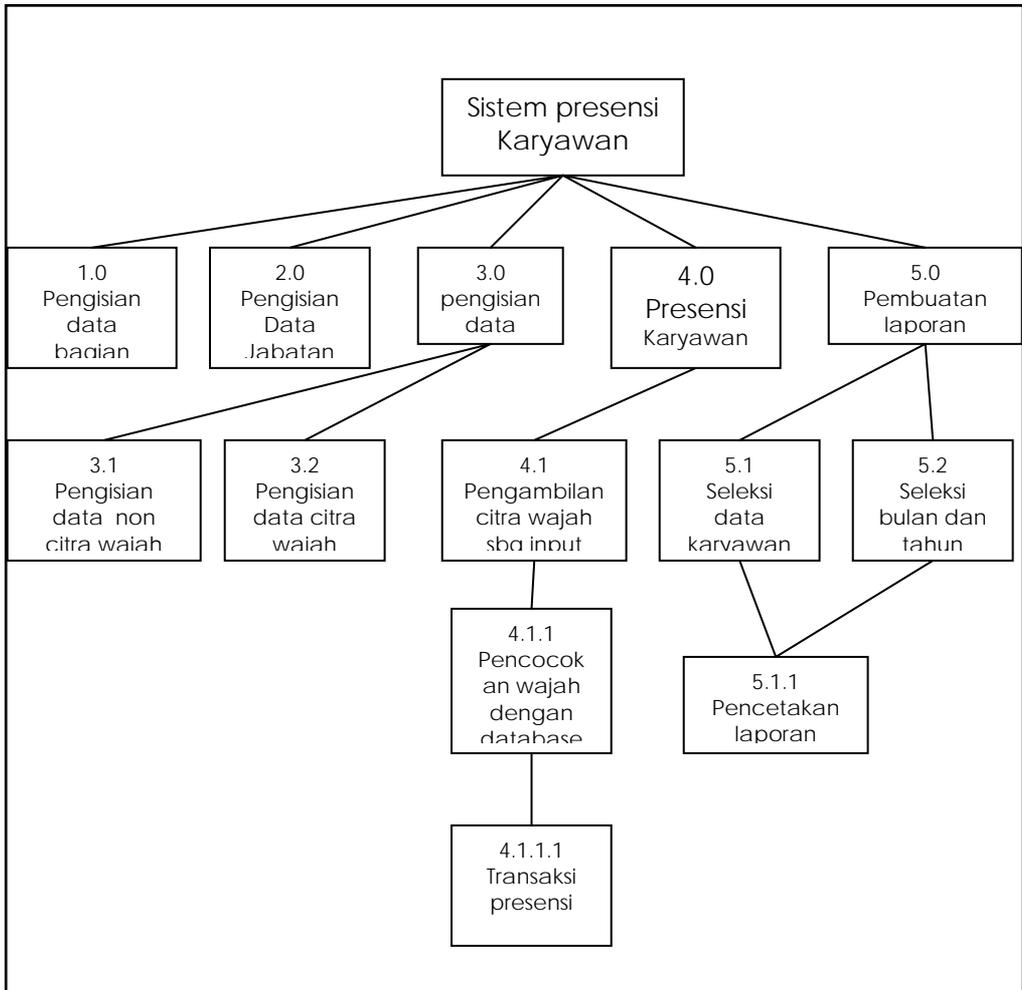
HIPO merupakan teknik untuk mendokumentasikan pengembangan suatu sistem yang dikembangkan oleh IBM. HIPO dapat digunakan untuk memenuhi kebutuhan dari beberapa pemakai yang menggunakannya untuk kepentingan yang berbeda-beda. Antara lain sebagai berikut:

- Seorang manajer dapat menggunakan dokumentasi HIPO untuk memperoleh gambaran umum sistem
- Seorang programmer menggunakannya untuk menentukan fungsi-fungsi dalam program yang dibuatnya.
- Programmer juga dapat menggunakan dokumen ini untuk mencari fungsi-fungsi yang dimodifikasi dengan cepat

Teknik ini mempunyai beberapa tujuan utama. Pertama dapat dibuat sebuah struktur yang menggambarkan hubungan antar fungsi dalam program secara hirarkis. (Tim Politeknik Telkom Bandung, 2009)¹⁴

Perhatikan gambar berikut ini:.

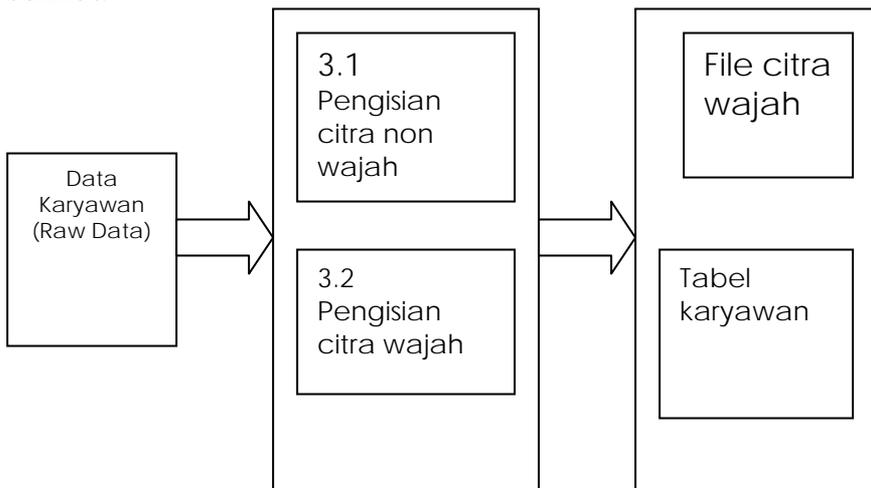
[14] Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung



Gambar 13.1 Diagram DIV dari HIPO

Sasaran kedua adalah untuk menentukan fungsi-fungsi apa saja yang harus ada dalam sistem yang dikembangkan. Dari gambar diatas terlihat jelas fungsi apa saja yang harus dibuat. Sasaran ketiga adalah untuk mendapatkan gambaran input dari fungsi

dan output apa yang dihasilkan. Sebagai contoh perhatikan modul fungsi no 3. Diagram HIPO yang bisa dibuat sebagai berikut:



Gambar 13.2 Diagram Ringkas

13.2 Jenis Diagram HIPO

Paket HIPO terdiri dari 3 jenis diagram yaitu diagram daftar isi visual (Visual Table of content). Diagram Ringkas (Overview diagram) dan diagram rinci (detail diagram)

Daftar Isi Visual (DIV)

Diagram ini memuat semua modul yang ada dalam sistem berikut nama dan nomornya, yang nantinya akan diperinci dalam diagram ringkas dan diagram rinci. Dalam DIV juga bisa

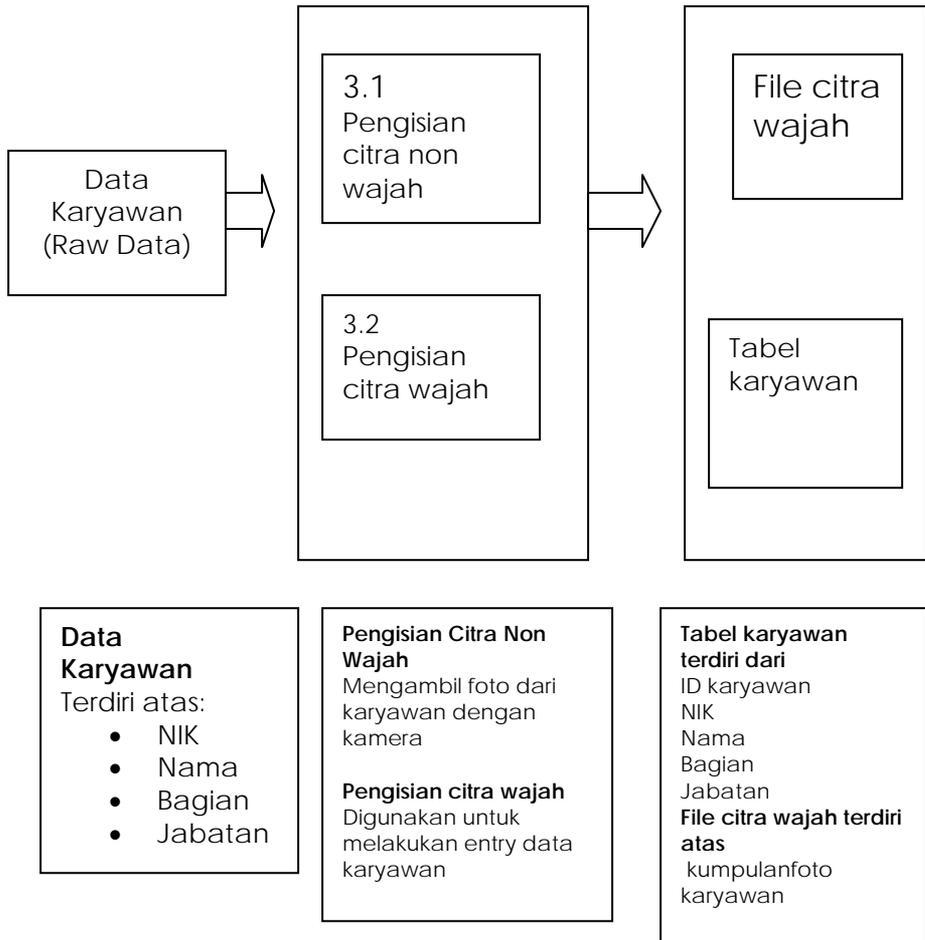
dilihat fungsi-fungsi utama yang menyusun sebuah sistem dan hubungan antar fungsi tersebut.

Diagram Ringkas

Diagram Ringkas menerangkan input, proses dan output dari sistem. Diagram ringkas menggambarkan input dan output dari fungsi-fungsi yang telah didefinisikan dalam Daftar Isi Visual. Perhatikan gambar 13.2 untuk contoh sebuah diagram ringkas.

Diagram Rinci

Diagram Rinci HIPO digunakan untuk memperinci input, proses dan output yang telah digambarkan dalam diagram ringkas. Input data dijelaskan field-field datanya secara detail. Untuk Fungsi, juga dideskripsikan proses apa yang dilakukan oleh fungsi-fungsi tersebut. Rincian field-field data output juga dijelaskan dengan lebih detail.



BAB 14

PEDOMAN DESAIN INTERFACE

14.1 User interface desain

- Beberapa aplikasi akan memiliki user interface yang sederhana, yang lain akan memiliki user interface yang kompleks.
- Akan ada 1 user interface untuk setiap pengguna
- Tujuan dari UI adalah untuk memungkinkan user menjalankan setiap tugas dalam user requirement
- Jadi dalam membangun sebuah user interface harus berdasar pada User Requirement

Prinsip-prinsip user interface yang baik:

- UI yang baik tidak mengharuskan pengguna untuk mengingat tampilan UI
- UI menampilkan apa yang dimengerti oleh user atau visualisasi dari keadaan dari sistem sekarang.
- hal yang harus dihindari :
- menampilkan terlalu banyak informasi dan terlalu banyak pilihan

- Menampilkan terlalu sedikit informasi, terlalu sedikit pilihan dan tanpa konteks
- Eksploitasi struktur menu standar yang sudah familiar dengan software yang sering digunakan user.

BAB 15

IMPLEMENTASI SISTEM

15.1 Implementasi sistem

Tahapan implementasi sistem dapat terdiri dari langkah-langkah berikut ini:

- Menerapkan rencana implementasi
- Melakukan kegiatan implementasi
- Tindak lanjut implementasi

15.1.1 Menerapkan rencana implementasi

Supaya kegiatan implementasi nantinya dapat beroperasi sesuai dengan yang diharapkan, maka suatu rencana implementasi . Sehingga rencana implementasi merupakan kegiatan awal dari tahap implementasi sistem. Rencana implementasi dimaksudkan untuk mengatur biaya dan waktu yang dibutuhkan. Dalam tahapan ini biaya yang dibutuhkan dikalkulasi dalam bentuk anggaran biaya. Anggaran biaya inilah yang digunakan untuk kendali terhadap biaya-biaya yang dikeluarkan. Waktu yang diperlukan untuk melaksanakan tahapan-tahapan pekerjaan juga perlu ditentukan disini. Lebih jauh tentang penjadwalan akan di bahas pada modul tersendiri.

15.1.2 Kegiatan Implementasi

Kegiatan implementasi dilakukan dengan dasar kegiatan yang sudah direncanakan dalam rencana implementasi. Kegiatan yang dilakukan dalam tahapan implementasi adalah :

- Pemilihan dan pelatihan personil

- Pemilihan tempat dan instalasi perangkat lunak dan perangkat keras.
- Pemrograman dan pengetesan program
- Pengetesan sistem
- Konversi sistem.

15.1.3 Pemilihan dan pelatihan personil

Personil yang dipilih dapat berasal dari 2 sumber yaitu karyawan yang telah ada dari perusahaan atau calon karyawan yang berasal dari luar.

Pelatihan karyawan

Ada beberapa pendekatan pelatihan yaitu:

- Ceramah/seminar
- Pelatihan prosedural
- Pelatihan tutorial
- Simulasi
- Latihan langsung di pekerjaan (on-the-job-training)

15.1.4 Pemilihan tempat dan instalasi perangkat lunak dan perangkat keras.

Jika peralatan baru akan dimiliki, maka tempat atau ruangan untuk hardware baru harus juga dipersiapkan. Sistem yang besar membutuhkan tempat dan lingkungan yang harus diperhitungkan, misalnya dengan pemasangan AC, penerangan, pendeteksi kebakaran, UPS, telepon dan lain sebagainya. Langkah selanjutnya adalah menginstall hardware dan softwarenya.

c.Integration testing

Tes interaksi dari modul-modul untuk menjamin bahwa mereka bekerja dengan baik. Integration test terdiri dari serangkaian tes sebagai berikut:

Ujicoba Antarmuka

Ujicoba setiap fungsi dari interface

Ujicoba skenario pengguna

Pastikan setiap skenario berjalan dengan baik

Ujicoba aliran data

Uji setiap proses dalam langkah per-langkah

Ujicoba interface sistem

Pastikan data mengalir antar proses

d.System testing

Tes untuk menjamin software bekerja dengan baik sebagai bagian dari keseluruhan sistem.

e.Requirements Testing

Pastikan bahwa integrasi sistem tidak menyebabkan kesalahan.

Tes ini terdiri dari serangkaian testing yaitu:

Usability Testing

Uji bagaimana sistem sudah bebas dari kesalahan dan mudah digunakan

Security Testing

Pastikan fungsi-fungsi keamanan berjalan dengan baik

Performance Testing

Pastikan bahwa sistem bekerja dengan baik pada aktivitas pekerjaan dengan volume besar

Documentation Testing

Uji apakah dokumen yang dihasilkan sudah benar dan sesuai dengan yang dibutuhkan

e. Acceptance testing

Tes untuk menjamin bahwa sistem telah melayani kebutuhan organisasi

• Alpha Testing

Tes ini dilakukan oleh user untuk menjamin bahwa mereka menerima sistem, tes dilakukan dengan menggunakan data tes .

• Beta Testing

Pengujian dengan menggunakan data riil, bukan data tes. User sesungguhnya memonitor kesalahan yang terjadi atau perbaikan yang dibutuhkan.

BAB 16

PEMELIHARAAN SISTEM

16.1 Pengertian Pemeliharaan Sistem

Pemeliharaan sistem informasi adalah suatu upaya atau tindakan untuk

- Memperbaiki
- Menjaga
- Menanggulangi
- Mengembangkan sistem yang ada.

Pemeliharaan ini di perlukan untuk meningkatkan efisiensi dan efektivitas kinerja sistem yang ada agar dalam penggunaannya dapat bekerja secara optimal (Tim Politeknik Telkom Bandung, 2009)¹⁵.

16.1.1 Sistem perlu dipelihara karena beberapa hal, yaitu :

1. Sistem memiliki kesalahan yang dulunya belum terdeteksi, sehingga kesalahan-kesalahan sistem perlu diperbaiki.
2. Sistem mengalami perubahan-perubahan karena permintaan baru dari pemakai sistem.
3. Sistem mengalami perubahan karena perubahan lingkungan luar (perubahan bisnis).
4. Sistem terinfeksi malware aktif
5. Sistem berkas corrupt
6. Perangkat keras melemah

16.1.2 Pemeliharaan sistem meliputi :

1. System Back-Up

Membuat Salinan/copy untuk data-data penting perusahaan yang ada pada computer user maupun server ke dalam backup storage (External Disk).

[15] Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung

2. System Optimization

Melakukan Defragmentasi data dan membuang sampah-sampah yang ada pada computer, serta memperbaiki kesalahan setting sehingga computer dapat berjalan normal.

3. System Rebuild

Membangun dan menata ulang kembali system yang rusak oleh faktor yang tidak disengaja, agar system dapat bekerja normal kembali.

4. System Upgrade

Menambah fungsi, memperbaharui system yang ada sesuai dengan kebutuhan pelanggan, serta melakukan testing stabilitas untuk hardware dan software.

5. Training dan Pelatihan

Memberikan Pengarahan dan konsultasi kepada operator computer, sehingga operator dapat mengoperasikan computer sesuai dengan prosedur pengoperasian komputer yang baik dan benar.

6. Update Anti Virus & Pembersihan Virus

Melakukan Update Definition file Anti Virus sehingga anti virus yang ada dapat memproteksi komputer dari serangan virus baik virus lam amaupun baru, dan juga melakukan scanning virus serta membersihkan komputer dari Virus.

7. System Security

Pemasangan Firewall dan sistem autentifikasi untuk pengamanan system dan data penting perusahaan dari orang luar yang tidak berkepentingan.

16.1.3 Jenis Pemeliharaan Sistem**1. Pemeliharaan Korektif**

Membuat perubahan pada sistem informasi untuk memperbaiki kesalahan yang terjadi pada saat disain, coding, atau implementasi

2. Pemeliharaan Adaptif

Pengubahan sistem untuk mengembangkan fungsionalitasnya agar dapat mengakomodasi perubahan kebutuhan bisnis atau untuk migrasi ke lingkungan operasi yang berbeda.

3. Pemeliharaan Perfektif

Pengembangan untuk meningkatkan kinerja proses atau kegunaan antarmuka, atau penambahan fitur sistem yang sebenarnya tidak terlalu diperlukan

4. **Pemeliharaan Preventif**

Pengubahan sistem untuk memperkecil peluang terjadinya kesalahan di masa yang akan datang.

16.2 Prosedur-Prosedur Pemeliharaan Sistem

Kemampuan pemeliharaan sistem adalah kapasitas personil pemeliharaan untuk melakukan pemeliharaan korektif, adaptif, penyempurnaan atau preventif. Semakin mudah suatu sistem dipelihara, semakin kecil pula tenaga dan biaya yang harus dikeluarkan untuk memelihara sistem.

Maintainability system bertambah jika sistemnya dirancang agar mudah dirubah. Aspek ini meliputi prosedur-prosedur berikut :

1. **System Development Life Cycle (SDLC) dan Software Development Life Cycle (SWDLC)**

Aplikasi yang profesional dalam SDLC dan SWDLC serta teknik maupun perangkat modelling yang baik akan mendukung keseluruhan sistem untuk meningkatkan maintainability system.

2. **Definisi Data Standard**

Tren ke arah sistem manajemen database relasional mendasari dorongan ke normalisasi data dan definisi data standard.

3. **Bahasa Pemrograman Standard**

Penggunaan bahasa pemrograman standard, misalnya visual basic atau C untuk Java

4. **Rancangan Moduler**

Programmer dapat mengganti modul jauh lebih mudah daripada jika ia berurusan dengan keseluruhan program

5. **Modul Yang Dapat Digunakan Kembali**

Modul biasa dari kode yang dapat digunakan kembali, dapat diakses oleh semua aplikasi yang memerlukannya.

6. **Dokumentasi Standard**

Diperlukan sistem, pemakai, perangkat lunak dan dokumentasi operasi yang standard sehingga semua informasi yang diperlukan untuk beroperasi dan pemeliharaan aplikasi khusus akan tersedia.

7. Kontrol Sentral

Semua program, dokumentasi dan data tes seharusnya diinstal dalam penyimpanan pusat dari sistem CASE (Computer Aided Software Engineering atau Computer Assisted Software Engineering)

16.3 Alat-Alat Pemeliharaan Sistem

Secara spesifik alat- alat yang digunakan dalam melakukan pemeliharaan sistem tidak selalu sama, alat tersebut digunakan hanya sesuai dengan kebutuhan sistem saja baik itu hardware maupun softwarena. Contoh tool paling umum yang digunakan dalam pemeliharaan sistem diantaranya: Registry Cleaner, Antivirus, firewall security, backup aplikasi dan sebagainya.

16.4 Mengatur Pemeliharaan Sistem

Dalam pemeliharaan sistem, kita perlu mengelola tindakan yang akan dilakukan sehingga proses pemeliharaan bisa terlaksana secara teratur dan terkontrol. Berikut caranya:

- Menetapkan Kegiatan Pemeliharaan Sistem
- Merekam kegiatan pemeliharaan sistem
- Menggunakan sistem perangkat lunak helpdesk
- Mengevaluasi aktivitas pemeliharaan sistem
- Mengoptimalkan program pemeliharaan sistem

16.5 Langkah-Langkah Pemeliharaan Sistem

1. Penggunaan Sistem

Yaitu menggunakan sistem sesuai dengan fungsi tugasnya masing-masing untuk operasi rutin atau sehari-hari.

2. Audit Sistem

Yaitu melakukan penggunaan dan penelitian formal untuk menentukan seberapa baik sistem baru dapat memenuhi kriteria kinerja. Hal semacam ini disebut penelaahan setelah penerapan dan dapat dilakukan oleh seorang auditor internal.

BAB 17

AUDIT SISTEM INFORMASI

17.1 Pendahuluan

Seiring dengan semakin banyak dan berkembangnya suatu organisasi, baik pada pemerintahan maupun lembaga swasta, yang mengandalkan sistem informasi untuk mendukung jalannya operasional sehari-hari, maka kesadaran akan perlunya dilakukan review atas pengembangan suatu sistem informasi semakin meningkat.

Resiko-resiko yang akan ditemukan bilamana terjadinya kegagalan dalam pengembangan sistem informasi diantaranya:

1. Biaya pengembangan sistem melampaui anggaran yang ditetapkan.
2. Sistem tidak dapat diimplementasikan sesuai dengan jadwal yang ditetapkan.
3. Sistem yang telah dibangun tidak memenuhi kebutuhan pengguna.
4. Sistem yang dibangun tidak memberikan dampak efisiensi dan nilai ekonomis terhadap jalannya operasi institusi, baik pada masa sekarang maupun masa datang.
5. Sistem yang berjalan tidak menaati perjanjian dengan pihak ketiga atau memenuhi aturan yang berlaku.

Untuk mengantisipasi hal-hal di atas, organisasi menginginkan adanya jaminan dari pihak yang berkompeten dan independen mengenai kondisi sistem informasi yang akan atau sedang mereka gunakan. Pihak yang paling berkompeten dan memiliki keahlian untuk melakukan review tersebut adalah Auditor Sistem Informasi.

2. Pengembangan dan perolehan program dilaksanakan sesuai dengan otorisasi khusus dan umum dari pihak manajemen
3. Modifikasi program dilaksanakan dengan otorisasi dan persetujuan dari pihak manajemen
4. Pemrosesan transaksi, file laporan dan catatan komputer lainnya telah akurat, jelas dan komplit.
5. Data sumber yang tidak akurat atau yang tidak memiliki otorisasi yang tepat diidentifikasi dan ditangani sesuai dengan kebijakan manajerial yang ditetapkan.
6. File data komputer telah akurat, lengkap dan terjaga integritas dan confidentiality-nya

Tujuan dari audit sistem informasi bisa dibagi menjadi dua kelompok utama, antara lain:

1. *Conformance* (kesesuaian) Kelompok audit sistem informasi ini bertujuan adalah fokus untuk memperoleh kesimpulan dari aspek kesesuaian yaitu Kerahasiaan (*Confidentiality*), Integritas (*integrity*), Ketersediaan (*Availability*) dan Kepatuhan (*Compliance*).
2. *Performance Kinerja*, kelompok tujuan audit sistem informasi ini berfokus pada memperoleh kesimpulan terhadap aspek kinerja yaitu Efektifitas (*Effectiveness*), Efisiensi (*Efficiency*), dan Keandalan (*Reliability*) (Langer, Arthur M. 2008)³.

Tujuan audit tersebut diatas berkaitan dengan komponen dari sistem informasi. Keterkaitan antara tujuan audit dan komponen sistem informasi dapat dilihat pada gambar 17.1.

^[3] Langer, Arthur M. 2008. *Analysis and Design of Information Systems 3rd edition*. Springer.

Manfaat Audit pada saat *pre-implementation system*, diantaranya:

1. Institusi dapat mengetahui apakah sistem yang telah dibuat sesuai dengan kebutuhan ataupun memenuhi *acceptance criteria*.
2. Mengetahui apakah pemakai telah siap menggunakan sistem tersebut.
3. Mengetahui apakah outcome sesuai dengan harapan manajemen.

Manfaat Audit pada saat *post-implementation system*, diantaranya:

1. Institusi mendapat masukan atas risiko-risiko yang masih ada dan saran untuk penanganannya.
2. Masukan-masukan tersebut dimasukkan dalam agenda penyempurnaan sistem, perencanaan strategis dan anggaran pada periode berikutnya.
3. Bahan untuk perencanaan strategis dan rencana anggaran di masa datang.
4. Memberikan *reasonable assurance* bahwa sistem informasi telah sesuai dengan kebijakan atau prosedur yang telah ditetapkan.
5. Membantu memastikan bahwa jejak pemeriksaan (*audit trail*) telah diaktifkan dan dapat digunakan oleh manajemen, auditor maupun pihak lain yang berwenang untuk melakukan pemeriksaan.
6. Membantu dalam penilaian apakah initial *proposed values* telah terealisasi dan saran tindak lanjutnya.

17.5 Spesialis Audit Sistem Informasi

Kegiatan audit sistem informasi hendaklah dilakukan oleh seorang spesialis audit sistem informasi. Spesialis audit sistem informasi mereview dan mengaudit seluruh area pada sistem informasi berkaitan dengan standard sesuai dengan organisasimisal profesional, hukum, aturan, kebijaksanaan organisasi, metodologi, pelaksanaan, integritas juga efektifitas biaya, kehandalan dan efisiensi (Kusrini, 2010)⁶.

Spesialisasi audit sistem informasi adalah:

1. Memimpin pelaksanaan audit dan review prosedur sistem. (diantaranya : sistem aplikasi, sistem perangkat keras, kebijaksanaan dan prosedur sekuriti, integritas Sistem Basis Data, perangkat lunak sistem, prosedur komunikasi dan jaringan komputer, operasi komputer) berdasarkan pertimbangan kepada hukum yang berkaitan dengan Sistem Informasi, kebijaksanaan organisasi, metodologi, praktek profesional, dan lainnya.
2. Memimpin pelaksanaan review sistem komputer dan informasi berkaitan dengan kehandalan, efisiensi, dan efektifitas biaya
3. Memimpin audit/review produktivitas sumber daya manusia teknologi Informasi
4. Merencanakan, menjadwal sumber daya untuk aktifitas audit.
5. Memimpin audit/review sistem informasi yang sedang dikembangkan untuk menjamin agar sesuai dengan standard
6. Mengembangkan dan merawat objektif dan prosedur audit organisasi

Hasil yang diharapkan dari kegiatan audit sistem informasi yang dilakukan oleh seorang spesialis audit sistem informasi adalah sebagai berikut:

[6] Kusrini, 2010, *Tinjauan Umum Mengenai Audit Sistem Informasi, Artikel*, STMIK-AMIKOM, Yogyakarta.

1. Dokumentasi obyektif , perencanaan, prosedur dan laporan audit.
2. Review secara berkala untuk memeriksa peningkatan kemampuan sistem

Untuk melaksanakan tugasnya dengan maksimal, Spesialis audit sistem informasi harus memiliki keterampilan sebagai berikut:

1. Siklus pengembangan sistem informasi
2. Analisis sistem dan teknik disain tingkat lanjutan serta metodologi pengembangan sistem.
3. Pemahaman yang baik, menangani sistem aplikasi, protokol komunikasi dan jaringan, dan operasi komputer.
4. Pemahaman yang baik mengenai standard kualitas nasional dan internasional, aturan-aturan sistem informasi, kebijaksanaan organisasi, metodologi dan lain sebagainya
5. Keterampilan interpersonal yang baik.
6. Harus dapat berargumentasi secara persuasif pada pertemuan formal dan informal.
7. Memiliki kemampuan menulis laporan dan teknis presentasi yang sangat baik.
8. Memiliki ketrampilan otomasi perkantoran (word processing, spreadsheet, graphics) yang sangat baik.
9. Metodologi audit yang sangat baik.
10. Kemampuan di atas raata-rata dalam mengobservasi detail dan terus menerus.

Selain pengetahuan dan keterampilan diatas seorang spesialis audit sistem informasi juga dituntut memenuhi syarat untuk akreditasi pribadi di bawah suatu sistem sertifikasi kualitas yang diakui secara internasional.

Salah satu sertifikasi profesional sebagai standar pencapaian prestasi dalam bidang audit, kontrol, dan keamanan sistem informasi yang telah diterima secara internasional adalah *CISA® (Certified Information Systems Auditor)* yang dikeluarkan oleh *ISACA (Information Systems Audit and Control Association)*.

17.6 Summary Audit Sistem Informasi

Audit sistem informasi dilakukan untuk menjaga agar sistem informasi dapat bertanggung melindungi aset milik organisasi, mampu menjaga integritas data, dapat membantu pencapaian tujuan organisasi secara efektif, serta menggunakan sumber daya yang dimiliki secara efisien.

Ada 6 kerangka tujuan yang harus diperhatikan dalam audit sistem informasi. Kerangka ini sangat berkaitan dengan komponen-komponen yang ada dalam sistem informasi.

Untuk melakukan audit sistem informasi, suatu organisasi dapat meminta kepada profesional audit sistem informasi yang terpercaya dengan pengetahuan dan keterampilan tertentu. Profesi audit sistem informasi ini telah memiliki badan akreditasi yang mengeluarkan sertifikasi yang berlaku secara internasional.

Daftar Pustaka

- [1]. Tim Politeknik Telkom Bandung, 2009, *Analisis dan Desain Sistem Informasi*, Politeknik Telkom, Bandung
- [2]. Nugroho, Adi, 2005, *Analisis dan Perancangan Sistem Informasi dengan Metodologi Berorientasi Objek, Edisi: Revisi*, Informatika, Bandung
- [3]. Langer, Arthur M. 2008. *Analysis and Design of Information Systems 3rd edition*. Springer.
- [4]. Bowman, Kevin. 2004. *System Analysis: A Beginner's Guide*. Palgrave Macmillan.
- [5]. Simon Bennet, Steve McRobb, Ray Farmer, 2002, *Object Oriented Systems Analysis and Design Using UML 2nd*, McGraw Hill
- [6]. Kusrini, 2010, *Tinjauan Umum Mengenai Audit Sistem Informasi, Artikel*, STMIK-AMIKOM, Yogyakarta.

TENTANG PENULIS



Ilka Zufria, M.Kom, Lulus S1 dan S2 di Universitas Putra Indonesia “YTPK” Padang Fakultas Ilmu Komputer (FILKOM) Program Studi Sistem Informasi April 2007 dan S2 di Program Studi Magister Ilmu Komputer konsentrasi Sistem Informasi April 2009. Saat ini aktif sebagai Dosen Tetap Program Studi Ilmu Komputer Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Sumatera

Utara Medan dan mengampu beberapa matakuliah diantaranya: Analisis Perancangan Sistem Informasi, Pemrograman Visual, Basis Data dan Rekayasa Perangkat Lunak, serta aktif juga menulis artikel ilmiah bidang informatika(komputer) pada jurnal-jurnal nasional terakreditasi dan internasional.

TENTANG EDITOR



Suendri, M.Kom. Lulus S1 di Program Studi Sistem Komputer Fakultas Ilmu Komputer Universitas Putra Indonesia “YPTK” Padang tahun 2009, Lulus S2 di Program Studi Magister Ilmu Komputer Konsentrasi Sistem Informasi tahun 2013. Saat ini adalah Dosen tetap Program Studi Sistem Informasi Fakultas Sains dan Teknologi Universitas Islam Negeri Sumatera Utara Medan. Mengampu matakuliah Pemrograman Berbasis Web Dasar dan Lanjutan.

ANALISIS DAN PERANCANGAN SISTEM INFORMASI

Analisis dan Perancangan Sistem Informasi (APSI) dimaknai sebagai proses organisasional kompleks dimana Sistem Informasi Berbasis Komputer(SIBK) diimplementasikan untuk membangun sistem informasi, menganalisis kebutuhan, merancang sistem informasi berbasis komputer dan memecahkan masalah dalam organisasi melalui sistem informasi.

Buku ini adalah buku yang membahas topik matakuliah inti di beberapa program studi bidang informatika diantaranya Sistem Informasi (SI), Teknologi Informasi maupun Teknik Informatika (TI). Dengan harapan buku ini dapat menjembatani mahasiswa dan para penggiat serta pengembang sistem informasi untuk mampu meningkatkan kemampuan analisis, pengkajian dan penelaahan lebih mendalam.

Buku ini juga disusun sebagai bahan bacaan bagi para akademisi dan praktisi bidang Teknologi Informasi dan Komunikasi (TIK) khususnya bidang informatika komputer. Dan juga bagi buku ini bermanfaat bagi mahasiswa program studi Sistem Informasi, Teknik Informatika, Teknologi Informasi, Rekayasa Perangkat Lunak dan Manajemen Informatika yang sedang menimba ilmu di Perguruan Tinggi.

Buku ini hadir dengan kajian dan cakupan diantaranya:

- * Konsep Dasar Sistem
- * Stakeholder dalam Sistem Informasi
- * Analisis dan Perancangan Sistem Informasi
- * Pendekatan-Pendekatan Pengembangan Sistem
- * Analisis Sistem
- * Pendefinisian Masalah
- * Penentuan Kebutuhan Sistem
- * Analisis Use Case
- * Process Modelling
- * Data Modelling
- * Disain Sistem
- * Disain Arsitektur
- * Hierarchy Input Output (HIPO) Chart
- * Pedoman Disain Interface
- * Implementasi Sistem
- * Pemeliharaan Sistem
- * Audit Sistem Informasi



Cv. Puskra Mitra Jaya
Jln. Williem Iskandar No – 2K/22 Medan
Tlpn. (061) 8008-8209 (0813-6106-0465)
Email: cvpusdikramitrajaya@gmail.com

ISBN 978-623-6853-94-8

