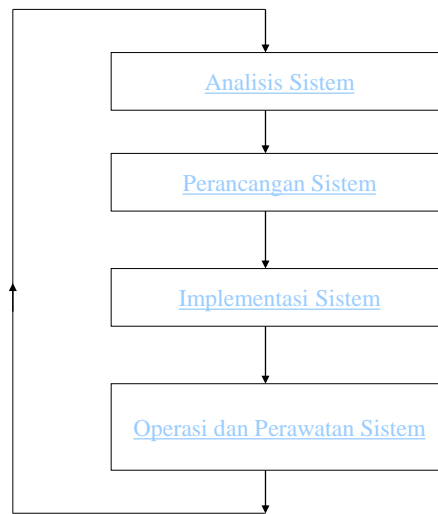


## Pengembangan Sistem Teknologi Informasi Metode SDLC

- ✓ Pengembangan sistem teknologi informasi (STI) dapat dilakukan dengan beberapa cara.
  - ▶ Pengembangan STI konvensional menggunakan metode siklus hidup pengembangan sistem atau *system development life cycle (SDLC)*.
  - ▶ Metode-metode alternatif (*alternative methods*) yang tersedia untuk mengembangkan STI adalah :
    1. Paket (*package*)
    2. Pembuatan prototip (*prototyping*)
    3. Pengembangan oleh pemakai akhir (*end user development* atau *end user computing*) dan
    4. *outsourcing*
- ✓ [Siklus Hidup Pengembangan Sistem](#)
- ✓ [Metodologi Pengembangan Sistem Tersruktur](#)

## Siklus Hidup Pengembangan Sistem

- ✓ Tahapan-tahapan dalam metode SDLC adalah :
  1. Analisis sistem (*system analysis*)
    - a. Studi pendahuluan
    - b. Studi kelayakan
    - c. Mengidentifikasi permasalahan dan kebutuhan pemakai
    - d. Memahami sistem yang ada
    - e. Menganalisis hasil penelitian.
  2. Perancangan sistem (*system design*)
    - a. Perancangan awal
    - b. Perancangan rinci.
  3. Implementasi sistem (*system implementation*)
  4. Operasi dan perawatan sistem (*system operation and maintenance*).
- ✓ [Kelebihan dan kekurangan Metode SDLC](#)



**Siklus hidup pengembangan sistem**

[Home](#)

## □ Analisis Sistem

- ✓ **Analisis sistem** (*system analyst*) adalah orang yang dididik khusus untuk mengembangkan sistem secara profesional.
- ✓ Tahap di analisis sistem terdiri dari kegiatan-kegiatan sebagai berikut :
  1. Studi pendahuluan
  2. [Studi kelayakan](#)
  3. Mengidentifikasi permasalahan dan kebutuhan informasi pemakai
  4. Memahami sistem yang ada
  5. Menganalisis hasil penelitian.

[Back](#)

## □ Studi Kelayakan (*Feasibility Study*)

- ✓ Studi kelayakan (*feasibility study*) terdiri dari lima macam kelayakan yang disebut dengan **TELOS**, yaitu :
  1. Studi kelayakan Teknologi
  2. Studi kelayakan Ekonomis
  3. Studi kelayakan Legal
  4. Studi kelayakan Operasi, dan
  5. Studi kelayakan Sosial
- ✓ Kelayakan ekonomis diperoleh jika manfaat dari STI lebih besar dari biaya-biaya dan dana yang dibutuhkan tersedia.

- ✓ Manfaat yang diperoleh dari STI dapat berbentuk :
  - **Manfaat-manfaat berujud (*tangible benefits*)**, merupakan manfaat-manfaat yang langsung dapat diukur dengan nilai uang.
  - **Manfaat-manfaat tidak berujud (*intangible benefits*)** merupakan manfaat-manfaat yang tidak langsung dapat diukur dengan nilai uang.
- ✓ Karena kelayakan ekonomis diukur dengan satuan uang, maka manfaat-manfaat tidak nampak harus dinilai uangkan. Beberapa metode tersedia untuk menilai uangkan manfaat-manfaat tidak nampak seperti metode nilai ekspektasi (*expected value*) dan metode skor.
- ✓ Metode **nilai ekspektasi (*expected value*)** dilakukan dengan mengidentifikasi **kejadian-kejadian (*outcomes*)** yang akan terjadi akibat dari manfaat tidak berujud dikalikan dengan probabilitas kemungkinan terjadinya.

[Back](#)

## ☐ Metode Nilai Ekspektasi (*Expected Value*)

- ✓ Metode **nilai ekspektasi** (*expected value*) dilakukan dengan mengidentifikasi **kejadian-kejadian** (*outcomes*) yang akan terjadi akibat dari manfaat tidak berujud dikalikan dengan probabilitas kemungkinan terjadinya.

Nilai ekspektasi kenaikan penjualan akibat kepuasan pelanggan

i	Kepuasan Pelanggan	Kenaikan Penjualan ( $O_i$ )	Probabilitas Terjadinya ( $p_i$ )	$O_i \times p_i$
1	Sangat puas	25% atau Rp5.000.000	50%	Rp2.500.000
2	Puas	20% atau Rp4.000.000	40%	Rp1.600.000
3	Cukup puas	10% atau Rp2.000.000	10%	Rp 200.000

- ✓ Nilai ekspektasi atau NE dihitung dengan menjumlahkan semua kejadian-kejadian (*outcomes*) dengan nilai probabilitas terjadinya sebagai berikut :

$$NE = \sum_{i=1}^k O_i \times p_i$$

- ✓ Maka besarnya nilai kepuasan pelanggan diperkirakan sebesar  
Rp2.500.000 + Rp1.600.000 + Rp200.000 = Rp4.300.000,-.

i	Ketidakpuasan Pelanggan	Penurunan Penjualan ( $O_i$ )	Probabilitas Terjadinya ( $p_i$ )	$O_i \times p_i$
1	Sangat tidak puas	30% atau Rp6.000.000	50%	Rp3.000.000
2	Tidak puas	20% atay Rp4.000.000	30%	Rp1.200.000
3	Cukup tidak puas	15% atau Rp3.000.000	20%	Rp600.000

- ✓ Besarnya nilai ketidakpuasan pelanggan diperkirakan sebesar  
Rp3.000.000 + Rp1.200.000 + Rp600.000 = Rp4.800.000,-. Total manfaat tidak nyata akibat kepuasan pelanggan dan ketidakpuasan pelanggan adalah Rp4.300.000 + Rp4.800.000 = Rp9.100.000,-.

[Back](#)

## Metode Skor

- ✓ Metode skor dilakukan dengan memberikan skor (*score*) dengan melihat pada dampak dari manfaat-manfaat tidak berujud yang diberikan oleh STI yang dikembangkan.
- ✓ Nilai skor akan lebih tinggi jika dampak dari manfaat tidak berujud lebih mengarah ke strategik dan nilai skor akan lebih kecil jika dampaknya lebih mengarah ke operasional.
- ✓ Satu poin skor kemudian dihubungkan dengan nilai uang.

Dampak dari manfaat tidak berujud	Skor
- Meningkatkan kompetisi (strategik)	100
- Meningkatkan keputusan manajemen puncak	80
- Meningkatkan keputusan manajemen menengah	60
- Meningkatkan keputusan manajemen bawah	40
- Meningkatkan efisiensi proses	20

- ✓ Jika satu poin skor diberi nilai Rp100.000,- maka besarnya manfaat tidak berujud kepuasan langganan yang dampaknya ke peningkatan kompetisi (strategik) adalah  $100 \times \text{Rp}100.000,- = \text{Rp}10.000.000,-$ .
- ✓ Setelah semua manfaat dan biaya-biaya dapat ditentukan, kelayakan ekonomis dapat dilihat di tabel berikut.

	Tahun 0	Tahun 1	Tahun 2	Tahun 3	Tahun 4
<b>Ia. Manfaat Berujud</b>					
Pengurangan biaya operasi	0	1,000,000	1,000,000	1,000,000	1,000,000
Pengurangan kesalahan program	0	2,000,000	2,000,000	2,000,000	2,000,000
Peningkatan penjualan	0	5,000,000	10,000,000	20,000,000	25,000,000
Pengurangan biaya sediaan	0	15,000,000	15,000,000	15,000,000	15,000,000
Pengurangan kredit tak tertagih	0	3,000,000	4,000,000	5,000,000	5,000,000
<b>Total Manfaat Berujud</b>	0	26,000,000	32,000,000	43,000,000	48,000,000

Sistem Teknologi Informasi BAB 9 – Copyright By Jogiyanto HM

	Tahun 0	Tahun 1	Tahun 2	Tahun 3	Tahun 4
<b>1b. Manfaat Tidak Berujud</b>					
Peningkatan Pelayanan Pelanggan	0	2,500,000	7,500,000	10,000,000	10,000,000
Peningkatan Kepuasan Pegawai	0	2,000,000	2,000,000	2,000,000	2,000,000
Peningkatan keputusan manajemen	0	2,500,000	4,000,000	6,000,000	7,000,000
<b>Total Manfaat Tidak Berujud</b>	0	8,000,000	14,500,000	19,000,000	20,000,000
<b>2a. Biaya Pengadaan (procurement costs)</b>					
Biaya pembelian perangkat keras	40,000,000	0	0	0	0
Biaya instalasi perangkat keras	500,000	0	0	0	0
Biaya manajemen dan staff	1,000,000	0	0	0	0
<b>Total Biaya Pengadaan</b>	41,500,000	0	0	0	0
<b>2b. Biaya persiapan operasi (start-up costs)</b>					
Biaya pembelian perangkat lunak sistem	1,000,000	0	0	0	0
Biaya reorganisasi	2,000,000	0	0	0	0
Biaya manajemen dan staff	1,000,000	0	0	0	0
<b>Total Biaya Persiapan Operasi</b>	3,000,000	0	0	0	0

Sistem Teknologi Informasi BAB 9 – Copyright By Jogiyanto HM

	Tahun 0	Tahun 1	Tahun 2	Tahun 3	Tahun 4
<b>2c. Biaya proyek (project related costs)</b>					
Biaya konsultan					
Analisis senior	20,000,000	0	0	0	0
Analisis junior	12,000,000	0	0	0	0
Pemrogram	10,500,000	0	0	0	0
Biaya perjalanan dan akomodasi	2,500,000	0	0	0	0
Biaya di Analisis Sistem					
Biaya pengumpulan data	1,000,000	0	0	0	0
Biaya dokumentasi	1,000,000	0	0	0	0
Biaya rapat	500,000	0	0	0	0
Biaya manajemen dan staff	1,000,000	0	0	0	0
Biaya di Disain Sistem					
Biaya dokumentasi	500,000	0	0	0	0
Biaya rapat	500,000	0	0	0	0
Biaya manajemen dan staff	1,000,000	0	0	0	0
Biaya di Penerapan Sistem					
Biaya pembuatan formulir baru	1,000,000	0	0	0	0
Biaya konversi data	750,000	0	0	0	0
Biaya latihan personil	1,000,000	0	0	0	0
Biaya manajemen dan staff	1,000,000	0	0	0	0
<b>Total Biaya Proyek</b>	54,250,000	0	0	0	0

	Tahun 0	Tahun 1	Tahun 2	Tahun 3	Tahun 4
<b>2d. Biaya Operasi</b>					
Biaya gaji (operator dan administrasi)	0	800,000	800,000	1,000,000	1,200,000
Biaya listrik	0	1,200,000	1,200,000	1,400,000	1,400,000
Biaya bahan habis pakai	0	1,500,000	1,500,000	1,500,000	1,500,000
Biaya manajemen	0	1,000,000	1,000,000	1,250,000	1,500,000
<b>Total Biaya Operasi</b>	0	4,500,000	4,500,000	5,150,000	5,600,000
<b>TOTAL ARUS KAS</b>	<b>98,250,000</b>	<b>29,500,000</b>	<b>42,000,000</b>	<b>56,850,000</b>	<b>62,400,000</b>
Payback period =	2.306 tahun				
NPV =	15,409,915				
IRR =	28.41%				

[Back](#)

## ❑ Perancangan Sistem

- ✓ Tahap perancangan sistem mempunyai dua tujuan utama :
  1. Memberikan gambaran secara umum tentang kebutuhan informasi kepada pemakai sistem secara logika.
  2. Memberikan gambaran yang jelas dan rancang bangun yang lengkap kepada pemrogram komputer dan ahli-ahli teknik lainnya.
- ✓ Tujuan perancangan sistem :
  1. Perancangan sistem secara logika (*logical system design*) atau perancangan sistem secara umum (*general system design*)
  2. Perancangan sistem secara terinci (*detail system design*)

[Back](#)

## □ General System Design

- ✓ Tujuan dari **perancangan sistem secara umum** (*general system design*) atau **perancangan sistem secara logika** (*logical system design*) atau **perancangan sistem secara konsep** (*conceptual system design*) adalah untuk memberikan gambaran secara umum kepada pemakai sistem tentang sistem teknologi informasi yang baru.
- ✓ Perancangan sistem secara umum lebih diarahkan kepada pemakai sistem untuk menyetujuinya ke perancangan sistem selanjutnya yaitu perancangan sistem secara terinci.
- ✓ Yang dirancang di tahap perancangan sistem secara umum adalah menggambarkan bentuk dari sistem teknologi informasinya secara logika atau secara konsep dan mengidentifikasikan komponen-komponen dari sistem teknologi informasinya.

[Back](#)

## □ Detail System Design

- ✓ Perancangan sistem secara terinci dirancang untuk menjawab pertanyaan bagaimana dan seperti apa bentuk dari komponen-komponennya.
- ✓ **Perancangan sistem secara terinci** (*detailed system design*) atau **perancangan sistem fisik** (*physical system design*) dimaksudkan untuk menggambarkan bentuk secara fisik dari komponen-komponen STI yang akan dibangun oleh pemrogram dan ahli teknik lainnya.

[Back](#)



## □ Implementasi Sistem

- ✓ **Implementasi sistem** (*system implementation*) adalah tahap meletakkan sistem supaya siap dioperasikan.
- ✓ Tahap implementasi sistem terdiri dari beberapa kegiatan sebagai berikut :
  1. Mempersiapkan rencana implementasi.
  2. Melakukan kegiatan implementasi.
    - a. Memilih dan melatih personil.
    - b. Memilih dan mempersiapkan tempat dan lokasi sistem.
    - c. Mengetes sistem.
    - d. Melakukan konversi sistem.
  3. Meninjau-lanjuti implementasi.

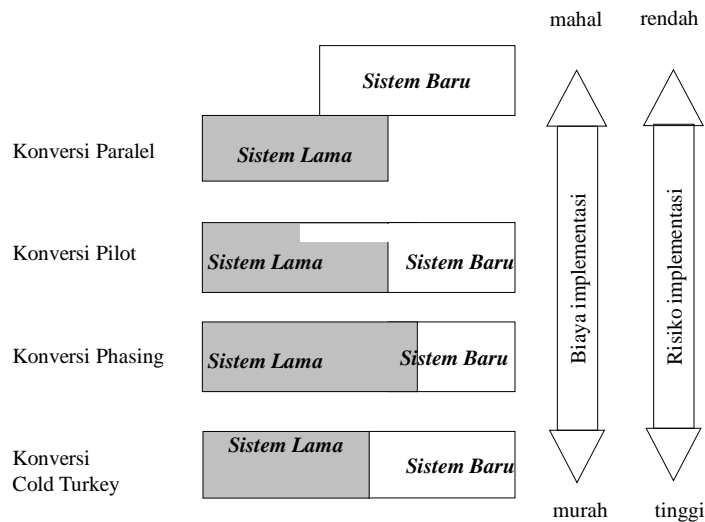
- ✓ Implementasi sistem juga merupakan proses mengganti atau meninggalkan sistem yang lama dengan sistem yang baru.
- ✓ Pendekatan atau strategi konversi yang ada adalah sebagai berikut ini.
  1. Konversi paralel.  
Pendekatan atau strategi **konversi paralel** (*parallel conversion*) dilakukan dengan mengoperasikan sistem yang baru bersama-sama dengan sistem yang lama selama satu periode waktu tertentu.
  2. Konversi pilot.  
Pendekatan atau strategi **konversi pilot** (*pilot conversion*) atau **pendekatan konversi lokasi** (*location conversion*) dilakukan bertahap pada suatu lokasi sebagai suatu percontohan dan jika berhasil dilanjutkan ke lokasi yang lainnya.

### 3. Konversi bertahap.

Pendekatan atau strategi **konversi bertahap** (*phasing conversion* atau *stepped conversion* atau *staged conversion* atau *phase-in conversion* atau *phased cut-over conversion*) dilakukan dengan menerapkan masing-masing modul dari sistem secara bertahap dan urut.

### 4. Konversi langsung.

Pendekatan atau strategi **konversi langsung** (*direct conversion* atau *direct cutover* atau *cold turkey conversion* atau *abrupt cutover*) dilakukan dengan mengganti sistem yang lama langsung dengan sistem yang baru.



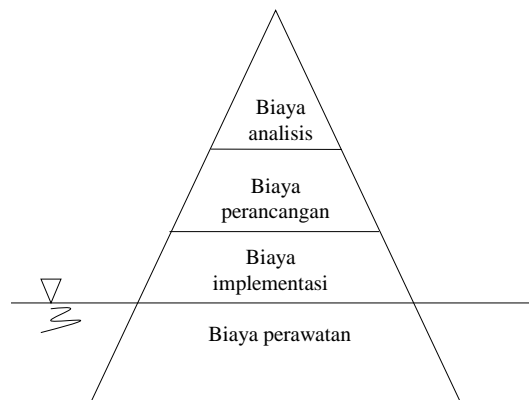
**Strategi konversi  
beserta biaya dan risiko yang harus ditanggungnya**

[Back](#)

## ❑ Operasi dan Perawatan Sistem

- ✓ Setelah sistem diimplementasi dengan berhasil, sistem akan dioperasikan dan dirawat. Tahap ini disebut dengan **operasi dan perawatan sistem** (*system operation and maintenance*).
- ✓ Sistem perlu dirawat karena :
  1. Sistem mengandung 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.
  4. Sistem perlu ditingkatkan.

- ✓ Biaya perawatan sistem sering diabaikan.
- ✓ Kenyataannya biaya perawatan sistem merupakan biaya yang cukup besar. Biaya perawatan sistem yang besar yang tidak disadari ini dapat digambarkan sebagai biaya di bawah gunung es berikut.



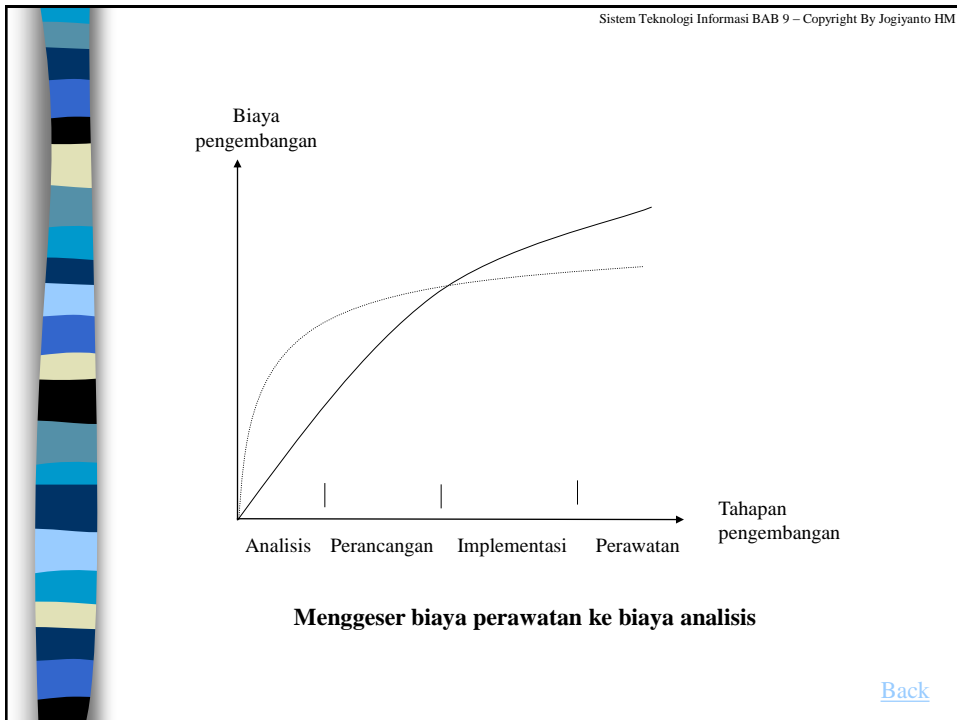
**Gunung es biaya perawatan**

- ✓ Biaya perawatan yang besar berkisar 20% sampai dengan 30% dari total biaya pengembangan sistem tiap tahunnya.
- ✓ Contoh kelayakan ekonomis di tabel 9.5 menunjukkan proyek pengembangan STI layak secara ekonomis dengan biaya perawatan yang belum dimasukkan.
- ✓ Misalnya biaya perawatan sebesar 20% dari total pengembangan sistem adalah sebesar  $20\% \times \text{Rp}54.250.000$  atau sebesar  $\text{Rp}10.850.000$  tiap tahunnya.

**Tabel 9.5 Kelayakan ekonomis jika biaya perawatan sistem dimasukkan**

	Tahun 0	Tahun 1	Tahun 2	Tahun 3	Tahun 4
Total Manfaat Berjud	0	26,000,000	32,000,000	43,000,000	48,000,000
Total Manfaat Tidak Berjud	0	8,000,000	14,500,000	19,000,000	20,000,000
Total Biaya Pengadaan	41,500,000	0	0	0	0
Total Biaya Persiapan Operasi	2,500,000	0	0	0	0
Total Biaya Proyek	54,250,000	0	0	0	0
Total Biaya Operasi	0	4,500,000	4,500,000	5,150,000	5,600,000
Biaya Perawatan		10,850,000	10,850,000	10,850,000	10,850,000
	-98,250,000	18,650,000	31,150,000	46,000,000	51,550,000
Payback period =	2,975 th				
NPV =	-7,996,559				
IRR =	15.53%				

- ✓ Hasil analisis ini menunjukkan bahwa proyek pengembangan STI tidak layak secara ekonomis, karena NPV bernilai negatif dan IRR dibawah bunga diskonto yang diminta.
- ✓ Besarnya biaya perawatan dapat dikurangi dengan cara menggeser biaya perawatan ke biaya analisis yaitu dengan cara lebih meningkatkan aktivitas di analisis supaya hasil dari analisis menjadi lebih baik.



Sistem Teknologi Informasi BAB 9 – Copyright By Jogiyanto HM

## Kelebihan dan Kekurangan Metode SDLC

✓ **Kelebihan-kelebihan dari metode ini adalah :**

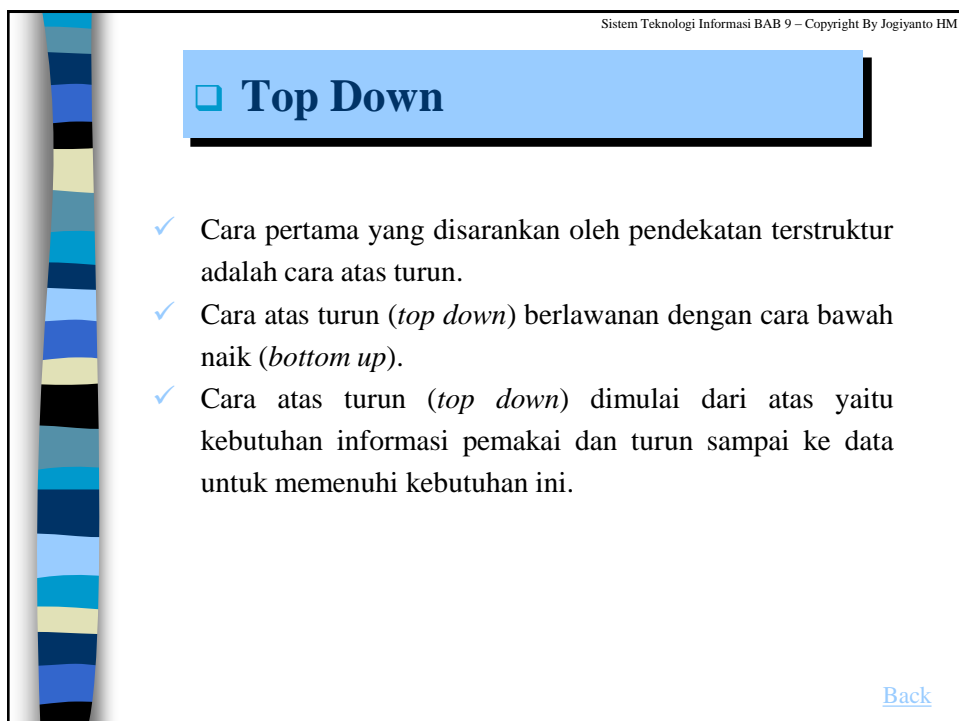
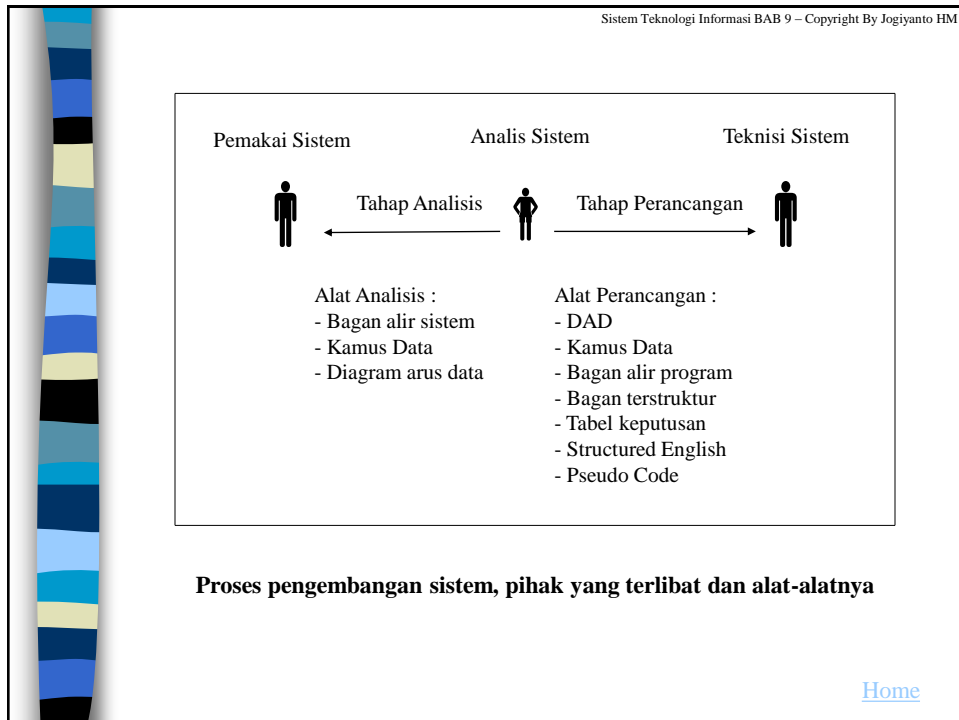
1. Menyediakan tahapan yang dapat digunakan sebagai pedoman mengembangkan sistem.
2. Akan memberikan hasil sistem yang lebih baik karena sistem dianalisis dan dirancang secara keseluruhan sebelum diimplementasikan

- ✓ **Kekurangan-kekurangan dari metode ini adalah :**
  1. Hanya menyediakan tahapan-tahapan saja, tetapi tidak menyediakan metodologi (cara dan alat-alat).
  2. Hasil dari SDLC sangat tergantung dari hasil di tahap analisis, sehingga jika terdapat kesalahan analisis, akan terbawa terus dengan hasil sistem yang kurang memuaskan.
  3. Dibutuhkan waktu yang lama untuk mengembangkannya karena sistem harus dikembangkan sampai selesai semua terlebih dahulu.
  4. Dibutuhkan biaya yang relatif lebih besar dibandingkan dengan metode lainnya.
  5. Hasil dari sistem tidak luwes untuk dimodifikasi karena perlu dilakukan analisis kembali.

[Back](#)

## Metodologi Pengembangan Sistem Tersruktur

- ✓ **Metodologi pendekatan tersruktur (*structured approach*)** memberikan cara [top down](#) dan [cara dekomposisi](#) dan beberapa alat pengembangan sistem.
- ✓ Beberapa **alat (*tools*)** diperlukan untuk metodologi pengembangan sistem terstruktur. Alat-alat yang tersedia untuk pendekatan ini diantaranya adalah:
  1. Bagan alir sistem ([system flow chart](#))
  2. Diagram arus data ([data flow diagram](#))
  3. Kamus data ([data dictionary](#))
  4. Bagan alir program ([program flow chart](#))
  5. Bagan terstruktur ([structured chart](#))
  6. *Structured english*
  7. [Pseudocode](#), dan
  8. Tabel keputusan ([decision table](#))



## □ Decomposition Approach

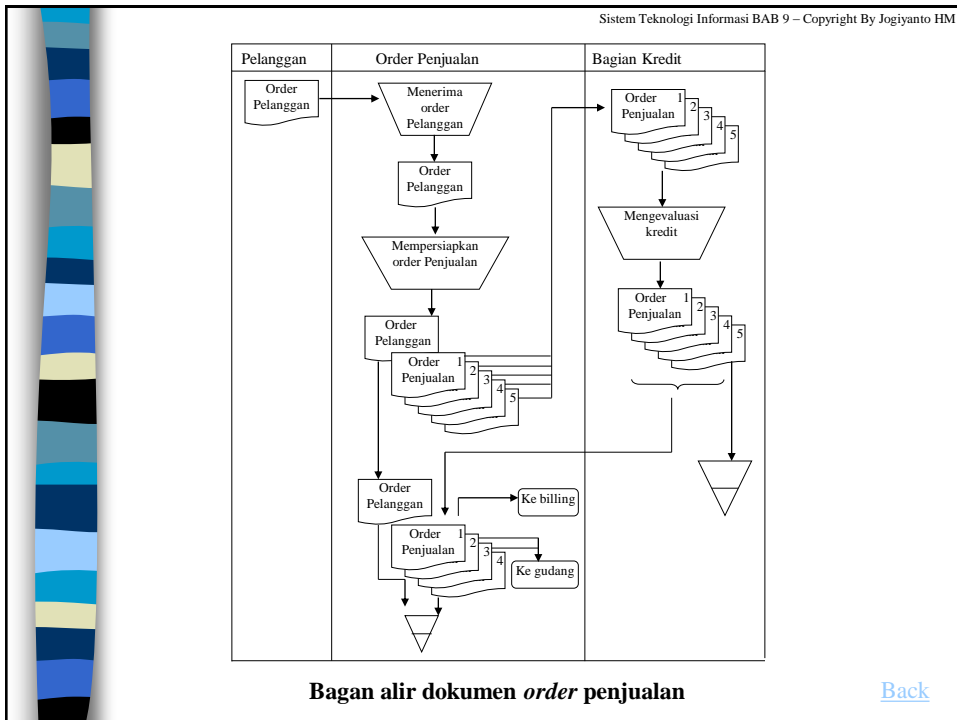
- ✓ Cara dekomposisi (*decomposition approach*) disebut juga dengan **cara moduler** (*modulair approach*) memecah sistem yang rumit menjadi beberapa bagian sistem yang disebut dengan modul-modul yang lebih sederhana
- ✓ Kebaikan cara ini adalah :
  - 1) Membuat sistem yang rumit menjadi mudah dipahami dalam bentuk-bentuk modul yang lebih sederhana,
  - 2) Dapat dilakukan pembagian kerja mengembangkan sistem sesuai dengan modul-modulnya,
  - 3) Sebagai dokumentasi yang baik untuk memahami sistem,
  - 4) Menyediakan **jejak audit** (*audit trail*) dan **proses menemukan kesalahan sistem** (*debugging*) yang baik jika sistem mempunyai beberapa kesalahan yang akan diperbaiki.

[Back](#)

## □ System Flow Chart

- ✓ Bagan alir sistem juga menunjukkan arus dari dokumen-dokumen yang ada di organisasi, sehingga disebut juga dengan nama **bagan alir dokumen** (*document flow chart*). Gambar berikut ini menunjukkan suatu bagan alir sistem.

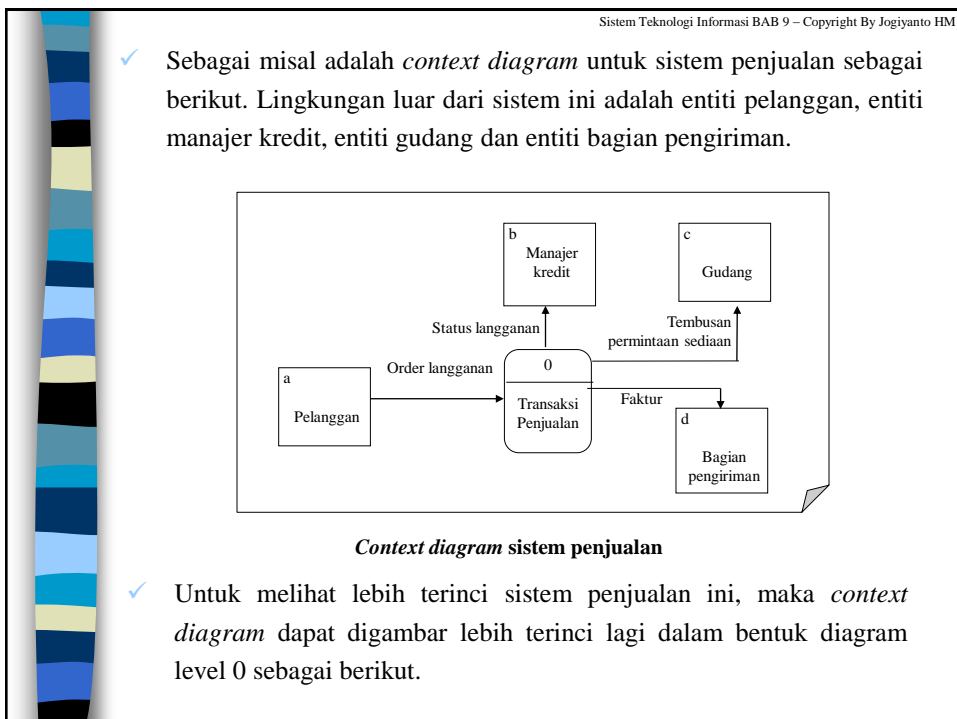
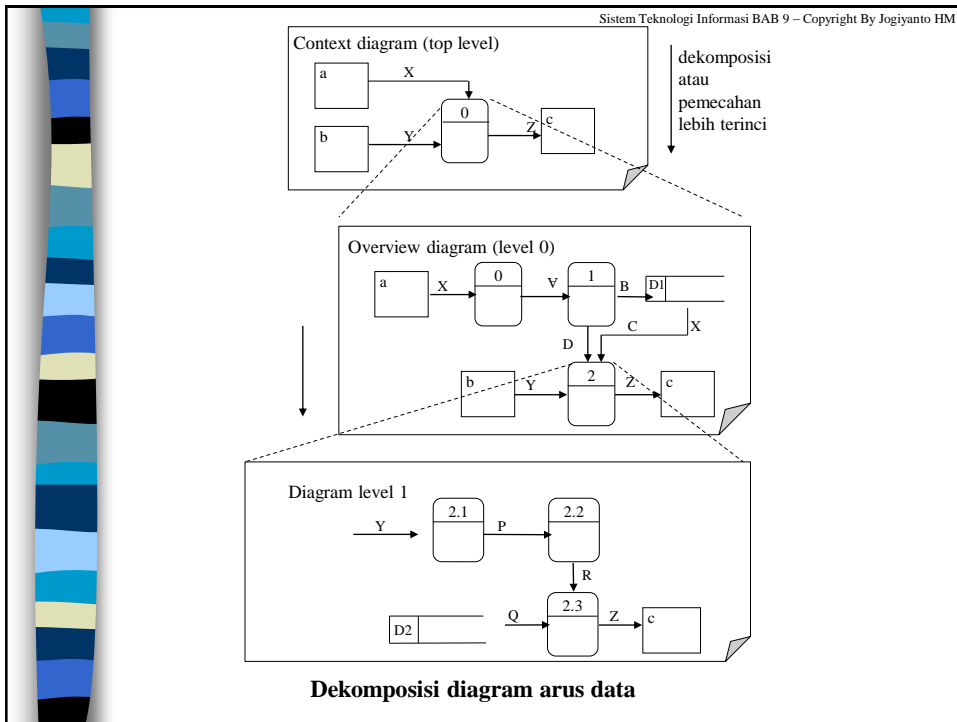


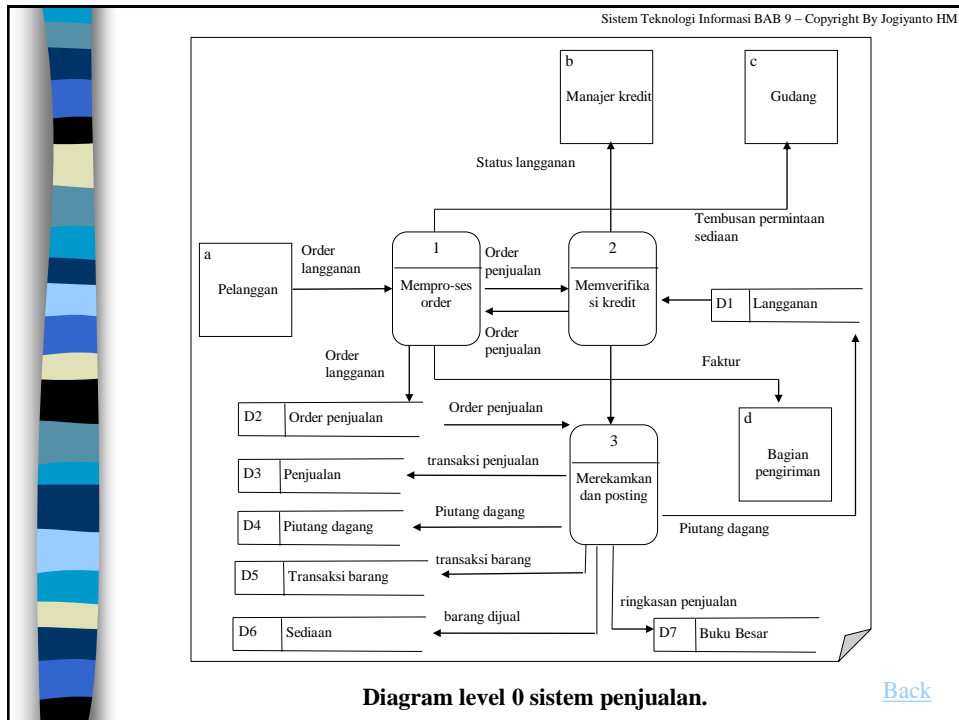


Sistem Teknologi Informasi BAB 9 – Copyright By Jogyanto HM

## □ Data Flow Diagram

- ✓ **Diagram arus data (DAD)** atau *data flow diagram (DFD)* menunjukkan data yang mengalir dari satu entiti ke entiti yang lain.
- ✓ Yang akan digambar pertama kali dalam DAD adalah **diagram level atas (top level diagram)** yang juga disebut dengan **diagram konteks (context diagram)**.
- ✓ Dari *context diagram* ini kemudian akan digambar menjadi lebih terinci lagi yang disebut dengan **overview diagram** atau **diagram level 0**.
- ✓ Dari diagram level 0 ini dapat dipecah-pecah kembali menjadi diagram-diagram yang lebih terinci menjadi **diagram level 1**, **diagram level 2** dan seterusnya sampai dianggap sudah cukup rinci untuk tidak dipecah kembali.





Sistem Teknologi Informasi BAB 9 – Copyright By Jogyanto HM

## ❑ Data Dictionary

- ✓ **Kamus data (KD)** atau *data dictionary (DD)* adalah katalog fakta tentang data yang mengalir di sistem.
- ✓ Kamus data ini menjelaskan atribut dari data yaitu tentang nama dari arus data, aliasnya, bentuk media data (dokumen dasar atau laporan atau layar komputer, variabel, parameter), arusnya (dari mana ke mana), penjelasannya, periode waktunya, volume datanya dan struktur datanya.

Sistem Teknologi Informasi BAB 9 – Copyright By Jogiyanto HM

**KAMUS DATA**

**Sistem :** Penjualan  
**Tahap :** Analisis Sistem  
**Nama data :** Tembusan Permintaan Sediaan  
**Alias :** Faktur, tembusan kredit, tembusan jurnal  
**Bentuk :** Dokumen cetakan komputer  
**Arus :** Proses 1 – Gudang  
**Penjelasan :** Tembusan faktur penjualan untuk meminta barang di gudang  
**Periode :** Setiap kali terjadi penjualan  
**Volume :** Volume rata-rata tiap hari 50 dan volume puncak adalah 75 kali

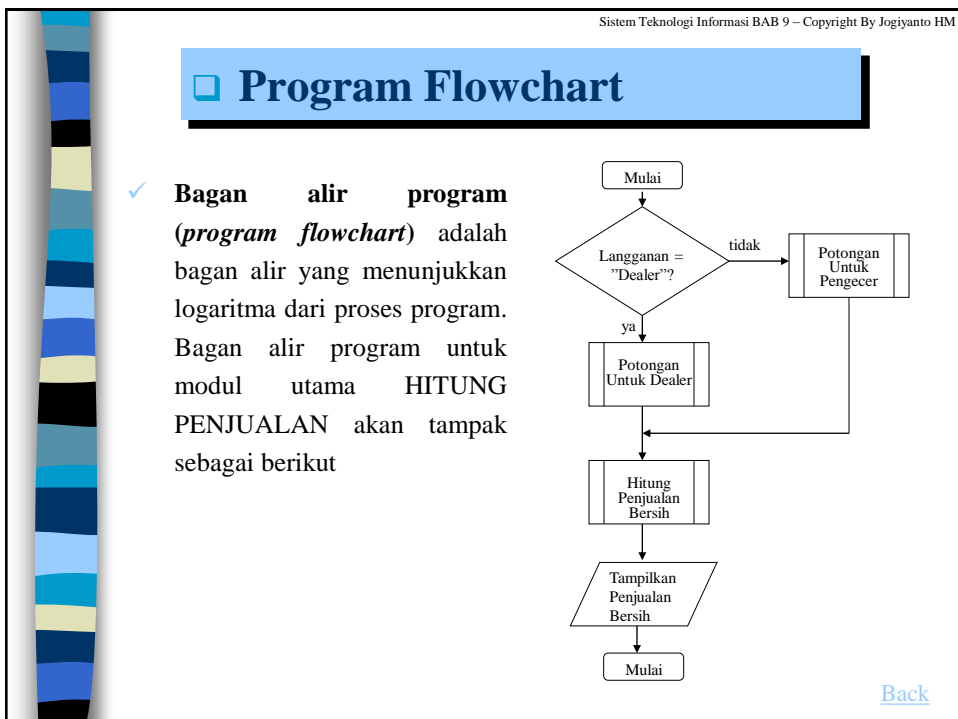
**Struktur Data:**

No.	Level	Nama Item Data	Tipe	Lebar	Range	Keterangan
1	01	KodeL.gn	C	5	-	Kode Langganan
2	01	Nama.Lgn	C	30	-	Nama Langganan
3	01	TglPenj	D	8	-	Tanggal Penjualan
4	01	NoFak	C	7	-	Nomor Faktur
5	01	(Kod Brg)	C	5	-	Kode Barang
6	01	NamaBrg	C	30	-	Nama Barang
7	01	Uni Jual 10	N	3	1 - 1 00	Unit dijual Maksimum 100
8	01	Harga at	N	10,2	-	Harga Satuan
9	01	TotalHarga	N	10,2	-	Total haraga per Barang
10	01	TotalPenj	N	10,2	-	Total Penjualan
11	01	PotPenj	N	10,2	0-15%	Potongan Penjualan
12	01	PajakPenj	N	10,2	-	Pajak Penjualan
13	01	TotalByr	N	10,2	-	Total dibayar
14	01	JenisPenj	C	1	T/K	Tunai/Kredit

Hak (C) Cipta oleh Jogiyanto HM

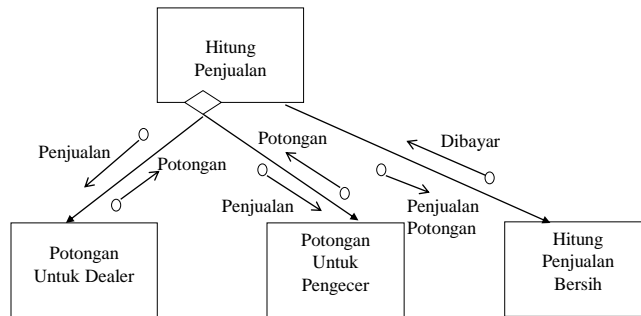
**Kamus data tembusan permintaan sediaan**

[Back](#)



## □ Structured Chart

- ✓ **Bagan tersruktur** (*structured chart*) digunakan untuk mendefinisikan dan mengilustrasikan hubungan elemen data dan elemen kontrol antar modul-modul sistem secara berjenjang. Contoh berikut ini menunjukkan bagan tersruktur untuk menghitung nilai penjualan.



**Bagan tersruktur menghitung nilai penjualan**

[Back](#)

## □ Pseudo Code

- ✓ *Pseudo* berarti imitasi atau mirip dan *code* berarti kode program, sehingga *pseudo code* dapat diartikan sebagai kode yang mirip dengan instruksi kode program komputer.
- ✓ Variasi lain dari *pseudo code* adalah *structured english*. Perbedaannya adalah jika *pseudo code* berbasis pada statemen kode program, *structured english* berbasis pada bahasa Inggris.
- ✓ Berikut ini contoh *pseudo code* untuk modul utama di aplikasi menghitung nilai penjualan.

```

if langganan adalah dealer then
    PotonganUntukDealer(Penjualan, Potongan)
else
    PotonganUntukPengecer(Penjualan, Potongan);

HitungPenjualanBersih (Penjualan, Potongan, Dibayar);
Write ('Penjualan Bersih = ', Dibayar);
  
```

[Back](#)

## Decision Table

- ✓ **Tabel keputusan (*decision table*)** : tabel yang digunakan sebagai alat bantu menyelesaikan logika penyeleksian kondisi di dalam program. Contoh dari tabel keputusan adalah sebagai berikut

	Aturan-aturan ( <i>rules</i> )												
	1	2	3	4	5	6	7	8	9	10	11	12	
<b>Kondisi-kondisi:</b>													
Apakah perusahaan mampu membayar utang-utangnya?	Y	Y	Y	Y	Y	Y	T	T	T	T	T	T	T
Apakah perusahaan telah meminjam terlalu banyak?	Y	Y	Y	T	T	T	Y	Y	Y	T	T	T	T
Bagaimana kemampuan laba perusahaan?	B	C	J	B	C	J	B	C	J	B	C	J	J
<b>Tindakan-tindakan:</b>				X									
Sangat sehat													
Sehat	X				X								
Cukup sehat		X				X	X			X			
Kurang sehat			X					X	X		X	X	

[Back](#)