

# Pengembangan Sistem Informasi

## 1. Pendahuluan

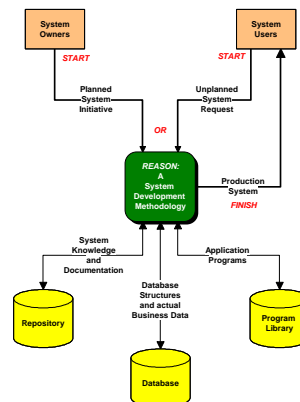
- Pengembangan sistem informasi sering disebut sebagai proses pengembangan sistem (*System Development*)
- Pengembangan sistem didefinisikan sebagai: adalah aktivitas untuk menghasilkan sistem informasi berbasis komputer untuk menyelesaikan persoalan (problem) organisasi atau memanfaatkan kesempatan (opportunities) yang timbul.

## Pendahuluan..... (lanjutan)

- Sebenarnya untuk menghasilkan sistem informasi tersebut terdiri dari:
  - *System Analysis*: upaya mendapatkan gambaran bagaimana sistem bekerja dan masalah-masalah apa saja yang ada pada sistem
  - *System Development*: langkah-langkah mengembangkan sistem informasi yang baru berdasarkan gambaran cara kerja sistem dan permasalahan yang ada

## 2. Konteks Pengembangan Sistem

### Information System Development



**The Context  
of System  
Development**

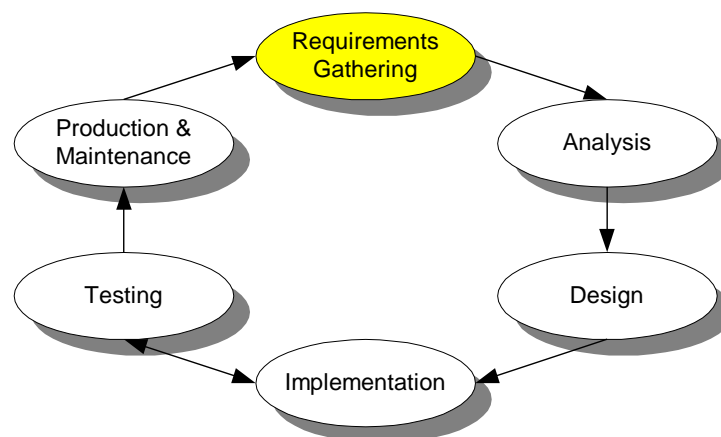
### 3. Metode Pengembangan Sistem

- Banyak metode pengembangan sistem yang tersedia
- Metode yang paling dikenal disebut juga sebagai System Development Life Cycle (SDLC) atau sering juga disebut sebagai *Water Fall Method*
- Metode-metode lain yang dikenal antara lain: Prototyping, Application Software, End-User Development, Outsourcing, dan lain-lain.
- Pada bagian berikut akan dijelaskan metode-metode tersebut secara ringkas.

5

LSIK - TI / Rajesri

### 4. Metode SDLC



Typical System Development Life Cycle (SDLC) diagram.

6

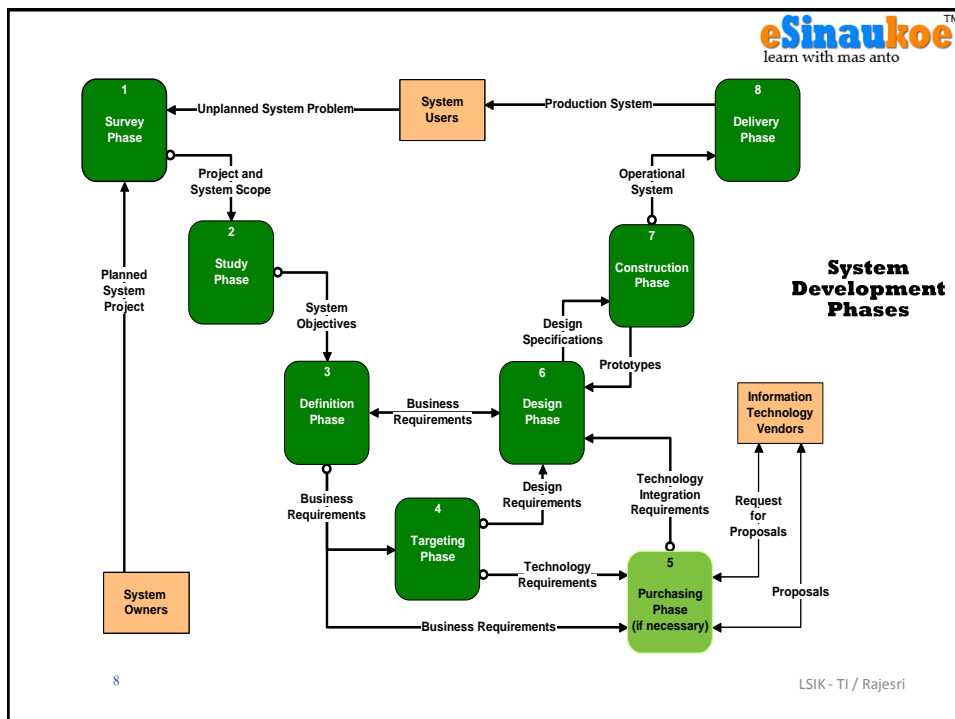
LSIK - TI / Rajesri

## SDLC (lanjutan)

- SDLC merupakan metode pengembangan sistem paling tua
- Sangat cocok untuk pengembangan sistem yang besar
- Tidak sesuai atau tidak terlalu disarankan untuk *small scale project* karena:
  - Resource intensive
  - Tidak fleksibel
  - Sulit untuk aplikasi dengan perubahan cara pengambilan keputusan yang cepat

7

LSIK - TI / Rajesri



8

LSIK - TI / Rajesri

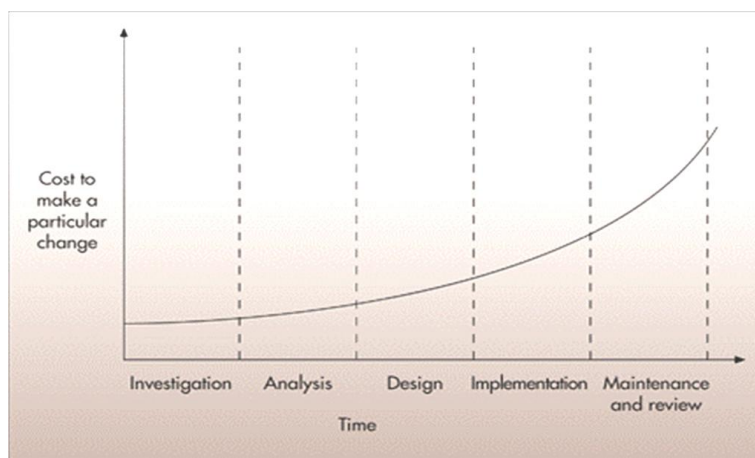
## SDLC (lanjutan)

Advantages	Disadvantages
<p>Formal review at the end of each phase allows maximum management control.</p>	<p>Users get a system that meets the needs as understood by the developers; this may not be what was really needed.</p>
<p>This approach creates considerable system documentation.</p>	<p>Documentation is expensive and time-consuming to create. It is also difficult to keep current.</p>
<p>Formal documentation ensures that system requirements can be traced back to stated business needs.</p>	<p>Often, user needs go unstated or are misunderstood.</p>
<p>It produces many intermediate products that can be reviewed to see whether they meet the users' needs and conform to standards.</p>	<p>Users cannot easily review intermediate products and evaluate whether a particular product (e.g., data flow diagram) meets their business requirements.</p>

9

LSIK - TI / Rajesri

## SDLC (lanjutan)



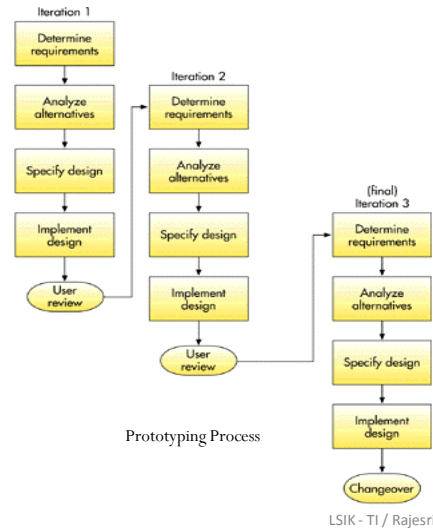
Cost to make a particular change in SDLC

10

LSIK - TI / Rajesri

## 5. Metode Prototyping

- Pendekatan iteratif dalam pengembangan sistem
- Dibuat prototype operasional sistem, gunakan data aktual, edit input, lakukan komputasi dan semua manipulasi sehingga dihasilkan output nyata
- Seperti membuat mock-up, coba, perbaiki, dst.



11

## ...metode prototyping

- Keuntungan:
  - Komunikasi user dan pengembang intensif
  - User terlibat aktif dalam menentukan requirement
  - Waktu pengembangan relatif singkat
  - Implementasi mudah karena pemakai mengetahui dari awal apa yang akan diperolehnya
- Kelemahan:
  - Kemungkinan terjadi *shortcut* dalam pendefinisian masalah
  - Pemakai bisa terlalu berlebih menentukan requirement sehingga sulit dipenuhi
  - Kemungkinan tidak dihasilkan rancangan yang baik

12

LSIK - TI / Rajesri

## 6. Metode Application Software

- Alternatif lain adalah dengan membeli software aplikasi yaitu paket software yang sudah jadi
- Misalkan membeli SAP, MSProject, dll.
- Digunakan untuk aplikasi yang bersifat umum, misalkan payroll, akunting, dll. Namun pada saat ini software yang berbasis enterprise secara keseluruhan banyak tersedia (*enterprise software*): Oracle, Baan, SAP, dll.
- Sangat sesuai jika perusahaan yang mengembangkan sistem kekurangan tenaga IT

## 7. Metode End-user Development

- Pengembangan dilakukan langsung oleh end-user.
- Menjadi semakin layak dengan tersedianya bahasa pemrograman yang mudah seperti MS Access, Delphi, dll.
- Keterlibatan langsung end-user sangat menguntungkan, karena memahami benar bagaimana sistem bekerja. Artinya tahap analisis sistem dapat dilakukan lebih cepat.
- Kelemahan adalah pada pengendalian mutu dan kecenderungan tumbuhnya "*private*" sistem informasi. Integrasi dengan sistem yang lain menjadi sulit.

## 8. Metode Outsourcing

- Dilakukan kontrak dengan pihak luar untuk menangani baik pengembangan maupun operasi & maintenance sistem.
- Menguntungkan dari sisi kecepatan memperoleh hasil dan biaya.
- Ada resiko tidak dapat mengendalikan sistem secara langsung dan masalah security.

## 9. Prinsip Dasar Pengembangan Sistem

- ▣ **Prinsip 1: Pemilik dan Pengguna Sistem Harus terlibat dalam pengembangan.**
  - ▣ Keterlibatan pemilik dan pengguna sistem (System Owner dan User) adalah keharusan yang mutlak untuk keberhasilan pengembangan sistem.
  - ▣ Pengembang sistem bertanggungjawab harus menyediakan waktu yang cukup untuk partisipasi pemilik dan pengguna sistem dan meminta persetujuannya untuk setiap langkah analisis dan pengembangan sistem.

## Prinsip Dasar Pengembangan Sistem

### ▣ **Prinsip 2: Gunakan Pendekatan Pemecahan Masalah**

- ▣ Metodologi yang digunakan dalam pengembangan sistem berbasis pendekatan bagaimana memecahkan masalah
- ▣ Langkah-langkah klasik pemecahan masalah adalah sbb:
  - ☞ Pelajari dan pahami masalah (opportunity, dan/atau directive) dan konteks dari sistem.
  - ☞ Definisikan kriteria atau ukuran solusi yang sesuai
  - ☞ Identifikasi alternatif-alternatif solusi dan pilih solusi terbaik.
  - ☞ Disain dan atau implementasikan solusi.
  - ☞ Observasi dan evaluasi dampak dari solusi dan sesuaikan solusi jika diperlukan.

## Prinsip Dasar Pengembangan Sistem

### (lanjutan prinsip 2)

- ▣ Ada kecenderungan untuk melewati langkah-langkah tersebut di atas atau melakukannya dengan kurang seksama.
- ▣ Akibat yang terjadi kemungkinan adalah:
  - ☞ Memecahkan persoalan yang salah
  - ☞ Kurang tepat dalam memecahkan persoalan
  - ☞ Mengambil solusi yang salah sama sekali

## Prinsip Dasar Pengembangan Sistem

### ▣ **Prinsip 3: Tentukan tahapan pengembangan**

- ▣ Pentahapan akan membuat proses pengembangan menjadi aktivitas-aktivitas yang lebih kecil yang lebih mudah dikelola dan diselesaikan.
- ▣ Tahapan pembuatan sistem harus dilakukan dengan urutan *top-to-bottom*.

## Prinsip Dasar Pengembangan Sistem

### ▣ **Prinsip 4: Tetapkan standard untuk pengembangan dan dokumentasi yang konsisten**

- ▣ Standard pengembangan sistem umumnya menjelaskan:
  - ↳ aktivitas
  - ↳ Tanggung jawab
  - ↳ Petunjuk dan kebutuhan pendokumentasian
  - ↳ Pemeriksaan kualitas
- ▣ Kegagalan pengembangan sistem akibat tidak tersedianya standard pendokumentasian merupakan hal yang banyak dijumpai dalam proyek pengembangan sistem

## Prinsip Dasar Pengembangan Sistem

### ▣ **Prinsip 5: Justifikasi sistem sebagai investasi**

- ▣ Sistem Informasi adalah sebuah investasi.
- ▣ Pada investasi maka terdapat dua hal yang harus diperhatikan, yaitu:
  - ☞ Untuk setiap masalah, kemungkinan ada beberapa alternatif solusi
  - ☞ Setelah alternatif-alternatif teridentifikasi, pengembang sistem harus mengevaluasi fisibilitas setiap kemungkinan alternatif solusi, terutama dikaitkan dengan *cost-effectiveness*.
    - **Cost-effectiveness** artinya semua biaya yang dikeluarkan untuk mengembangkan dan mengoperasikan sistem harus sebanding atau lebih sedikit dibandingkan dengan manfaat yang diperoleh dari sistem.
  - ☞ Cost-benefit analysis harus dilakukan.

21

LSIK - TI / Rajesri

## Prinsip Dasar Pengembangan Sistem

### ▣ **Prinsip 6: Jangan takut membatalkan atau merubah lingkup pekerjaan.**

- ▣ Keuntungan yang diperoleh dengan melakukan pentahapan pekerjaan pengembangan sistem adalah diperolehnya kesempatan untuk melakukan reevaluasi fisibilitas dari proyek pengembangan tersebut.
- ▣ Dalam jangka panjang membatalkan sama sekali proyek bisa lebih murah dibandingkan masalah besar pada implementasinya.
- ▣ Jika hal ini tidak dilakukan dapat terjadi pembengkakan biaya (*cost overruns*) yang sangat merugikan.

22

LSIK - TI / Rajesri

## Prinsip Dasar Pengembangan Sistem

### ▣ (lanjutan prinsip 6:)

- ▣ Pendekatan *creeping commitment* :
  - ☞ Titik-titik pemeriksaan (*checkpoints*) fisibilitas harus dibuat dalam proses pengembangan sistem.
  - ☞ Pada setiap titik pemeriksaan, semua biaya dianggap *sunk-cost* (artinya *irrecoverable*).
  - ☞ Pada titik tersebut proses re-evaluasi dilakukan untuk menentukan apakah proyek masih fisibel.
  - ☞ Dari hasil evaluasi tersebut, pengembang sistem dapat:
    - Membatalkan proyek jika dinilai tidak fisibel
    - Re-evaluasi biaya dan jadwal jika lingkup proyek berubah.
    - Pengurangan lingkup proyek jika anggaran dan jadwal tidak dapat ditambah sedangkan yang tersedia tidak mencukupi pencapaian semua tujuan yang telah ditetapkan.

23

LSIK - TI / Rajesri

## Prinsip Dasar Pengembangan Sistem

### ▣ **Prinsip 7: Bagi dan tundukkan**

- ▣ Semua sistem merupakan bagian dari sistem yang lebih besar (disebut *super-systems*).
- ▣ Begitu pula semua sistem terdiri dari bagian-bagian yang lebih kecil (disebut *subsystems*).
- ▣ Sistem harus dibagi menjadi subsistem-subsistem yang lebih kecil untuk memudahkan menyelesaikan persoalan dan membangun sistem yang lebih.
  - ☞ Dengan membagi persoalan besar (sistem) menjadi potongan kecil yang lebih mudah (subsistem), pengembang akan mudah menerapkan proses pemecahan masalah.

24

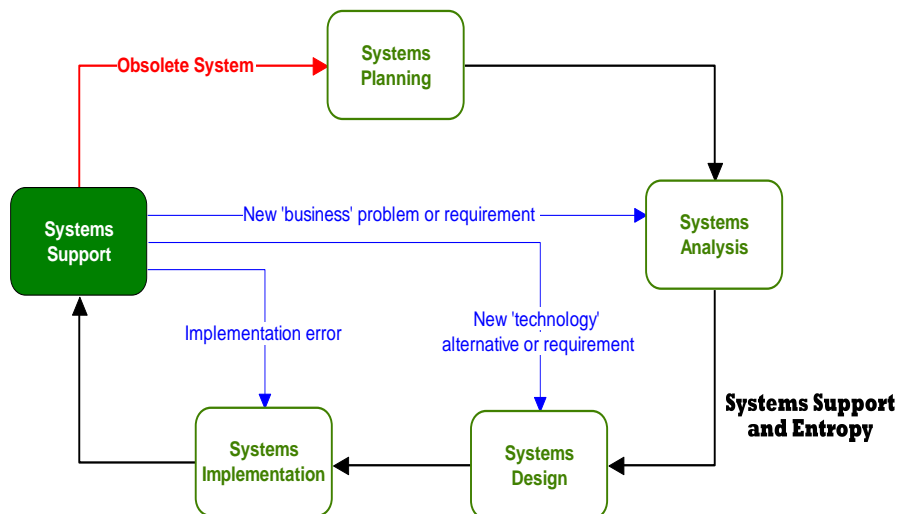
LSIK - TI / Rajesri

## Prinsip Dasar Pengembangan Sistem

- ▣ **Prinsip 8: Rancang sistem untuk pertumbuhan dan perubahan**
  - ▣ Banyak pengembang sistem yang terjebak dalam pengembangan sistem untuk memenuhi kebutuhan saat ini saja.
  - ▣ **Entropy** adalah pengertian yang dipakai untuk menjelaskan bahwa sistem secara alamiah akan mengalami penurunan.
    - ☞ Sistem bisa menjadi usang (*obsolete*) dan biaya untuk mengoperasikannya menjadi sangat besar.

25

LSIK - TI / Rajesri



26

LSIK - TI / Rajesri

## Prinsip Dasar Pengembangan Sistem

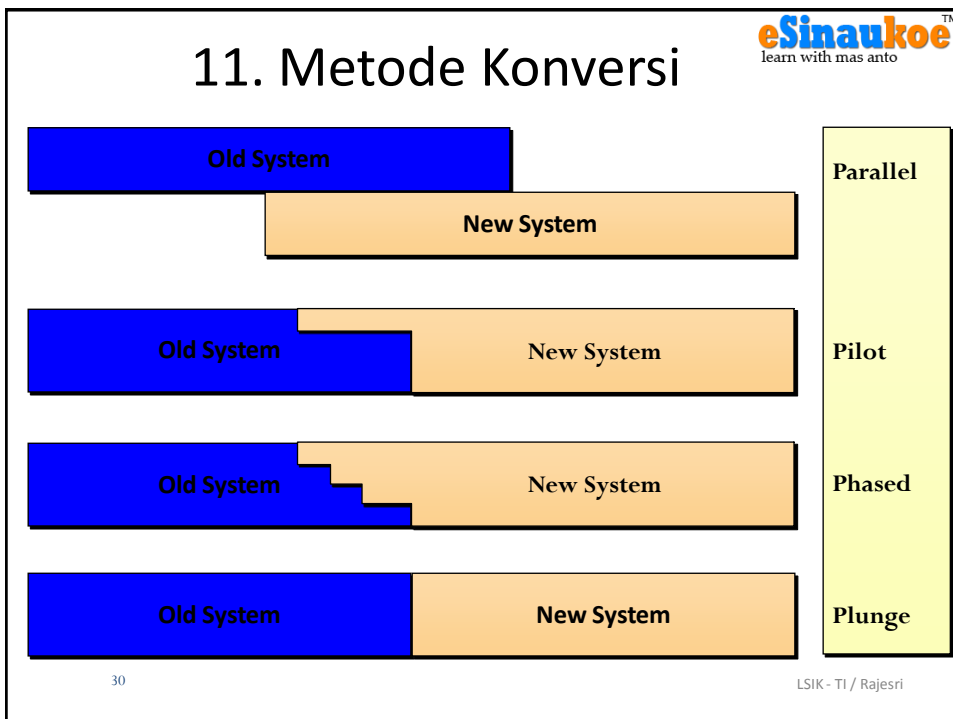
### ▣ (lanjutan Prinsip 8:)

- ▣ Sistem yang dirancang hanya untuk memenuhi kebutuhan saat ini akan sulit disesuaikan untuk menghadapi perubahan-perubahan.
- ▣ Perhatian harus sebanding antara memperhatikan sistem yang ada (sering disebut *legacy systems*), dan bagaimana memperkirakan arah pengembangan sistem yang baru.
- ▣ Fleksibilitas dan kemampuan beradaptasi tidak terjadi begitu saja tetapi harus dirancang secara sengaja di dalam sistem

## Prinsip Dasar Pengembangan Sistem

- **Libatkan pemilik dan pemakai sistem**
- **Gunakan pendekatan pemecahan masalah**
- **Buat pentahapan aktivitas**
- **Tetapkan standar pengembangan dan pendokumentasian yang konsisten**
- **Justifikasi sistem sebagai investasi**
- **Jangan takut membatalkan**
- **Bagi dan tundukkan**
- **Rancang sistem untuk pertumbuhan dan perubahan**

**Principles of  
Systems  
Development**



## 12. Fact Finding Techniques for Requirement Discovery



- Effective fact finding are crucial to the development of systems projects.
- System Requirements : specify what the information system must do or what property or quality the system must have.
  - Functional Requirements: what IS must do
  - Nonfunctional Requirements: specify a property or quality the system must have

LSIK - TI / Rajesri

## System Requirement Criteria:



- **Consistent** (not ambiguous or conflicting)
- **Complete** (describe all possible inputs and responses)
- **Feasible** (can be satisfied based on the available resources and constraints)
- **Required** (truly needed and fulfill the purpose of the system)
- **Accurate** (The requirements stated correctly)
- **Traceable** (directly map to the functions and features of the system)
- **Verifiable** (defined so that can be demonstrated during test)

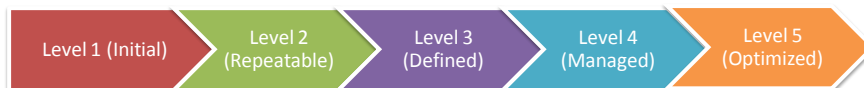
LSIK - TI / Rajesri

## Fact Finding Techniques

- **Sampling** of existing documentation, reports, forms, files, databases, and memos.
- **Research** of relevant literature, benchmarking of other's solution, and site visits.
- **Observation** of the current system in action and the work environment
- **Questionnaires** and surveys of the management and user community
- **Interviews** of appropriate managers, users, and technical staff
- **Joint Requirement Planning**

LSIK - TI / Rajesri

## 13. Capability Maturity Model



**Initial** → Inconsistent Methods

**Repeatable** → Consistent Project Management

**Defined** → Process is stable, predicable, and repeatable

**Managed** → Process Managed and Measured

**Optimized** → Continuous Process Improvement

LSIK - TI / Rajesri