

Pengembangan Sistem Pendukung Keputusan

<http://www.brigidaarie.com>

Review Tugas

- Perusahaan barang tembikar Colonial memproduksi 2 produk setiap hari, yaitu :
 - mangkok
 - cangkir
- Perusahaan mempunyai 2 sumber daya yang terbatas jumlahnya untuk memproduksi produk-produk tersebut yaitu:
 - Tanah liat (120 kg/hari)
 - Tenaga kerja (40 jam/hari)
- Dengan keterbatasan sumber daya, perusahaan ingin mengetahui berapa banyak mangkok dan gelas yang akan diproduksi tiap hari dalam rangka memaksimalkan laba
- Kedua produk mempunyai kebutuhan sumber daya untuk produksi serta laba per item seperti ditunjukkan pada tabel

Produk	Tenaga kerja (jam/unit)	Tanah Liat (kg/unit)	Laba (Rp/Unit)
Mangkok	1	3	4000
Cangkir	2	2	5000

PEMBUATAN MODEL

- Menentukan Variabel Keputusan
- Menentukan Fungsi Tujuan
- Menentukan Fungsi Batasan
- Memecahkan Model
- Implementasi Model

VARIABEL KEPUTUSAN

- X_1 = jumlah mangkok yang diproduksi/hari
- X_2 = jumlah cangkir yang diproduksi/hari

FUNGSI TUJUAN

- Memaksimumkan

$$Z = 4000 X_1 + 5000 X_2$$

- Z = total laba tiap hari
- $4000 X_1$ = laba dari mangkok
- $5000 X_2$ = laba dari cangkir Dengan

BATASAN

- Batasan Tenaga Kerja

$$1 X_1 + 2 X_2 \leq 40$$

- Batasan Tanah Liat

$$3 X_1 + 2 X_2 \leq 120$$

- Batasan Non Negatif

$$X_1, X_2 > 0$$

MENGUBAH FUNGSI TUJUAN DAN BATASAN-BATASAN

- Fungsi Tujuan

$$Z = 4000 X_1 + 5000 X_2$$

Menjadi

$$Z - 4000X_1 - 5000X_2 = 0$$

- Batasan Batasan

$$1 X_1 + 2 X_2 \leq 40$$

$$3 X_1 + 2 X_2 \leq 120$$

- Solusi maksimalnya adalah $X_1 = 40$, $X_2 = 0$ dan $Z = 160000$

- Jika ini disubstitusikan ke persamaan

$$Z = 4000 X_1 + 5000 X_2$$

$$160000 = 4000 * 40 + 5000 * X_2$$

$$X_2 = 0$$

- Solusi maksimalnya adalah $X_1 = 40$, $X_2 = 0$ dan $Z = 160000$
- Ini berarti jumlah produksi mangkuk per hari adalah 40, jumlah produksi cangkir per hari adalah 0 dengan keuntungan yang akan diperoleh perusahaan sebesar Rp. 160.000,-
- Dari hasil ini, kita juga bisa mengetahui bahwa jam kerja yang terpakai adalah sebesar:

$$\begin{aligned} 1 X_1 + 2 X_2 &= 40 + 2 * 0 \\ &= 40 \end{aligned}$$

- Karena sumber daya jam kerja yang dimiliki adalah 40 jam, berarti semua sumber daya jam kerja dipakai untuk memproduksi.
- Sedangkan tanah liat yang dibutuhkan untuk produksi sehari sebesar:

$$\begin{aligned} 3 X_1 + 2 X_2 &= 3 * 40 + 2 * 0 \\ &= 120 \end{aligned}$$

- Karena sumber daya tanah liat yang tersedia di perusahaan sebesar 120 kg/hari, berarti semua sumber daya tanah liat dipakai untuk memproduksi.

SISTEM BANTU KEPUTUSAN - PRODUKSI

Keuntungan (Rp.) :

Mangkok Cangkir

Kebutuhan Tenaga Kerja (Jam) :

Mangkok Cangkir

Kebutuhan Tanah Liat (Kg) :

Mangkok Cangkir

Maksimal Sumber Daya

Jam Kerja Tanah Liat

Jumlah Produksi Mangkok : 40
Jumlah Produksi Cangkir : 0
Sisa Jam Kerja : 0
Sisa Tanah Liat : 0
Keuntungan : 160000

Development DSS

Pendahuluan

- Membangun sebuah DSS, apalagi yang besar, merupakan proses yang rumit.
- Melibatkan hal-hal :
 - teknis (hardware, software)
 - perilaku (interaksi manusia-mesin), dampak DSS pada individu.

Strategi Pengembangan

1. Tulis DSS dengan bahasa pemrograman umum : Pascal, Delphi, Java, C++ dll.
2. Menggunakan 4GL : financial-oriented language, data-oriented language.
3. Menggunakan DSS Generator : Excell.
4. Menggunakan DSS Generator khusus
5. Mengembangkan DSS dengan metodologi CASE

Tingkat Teknologi DSS

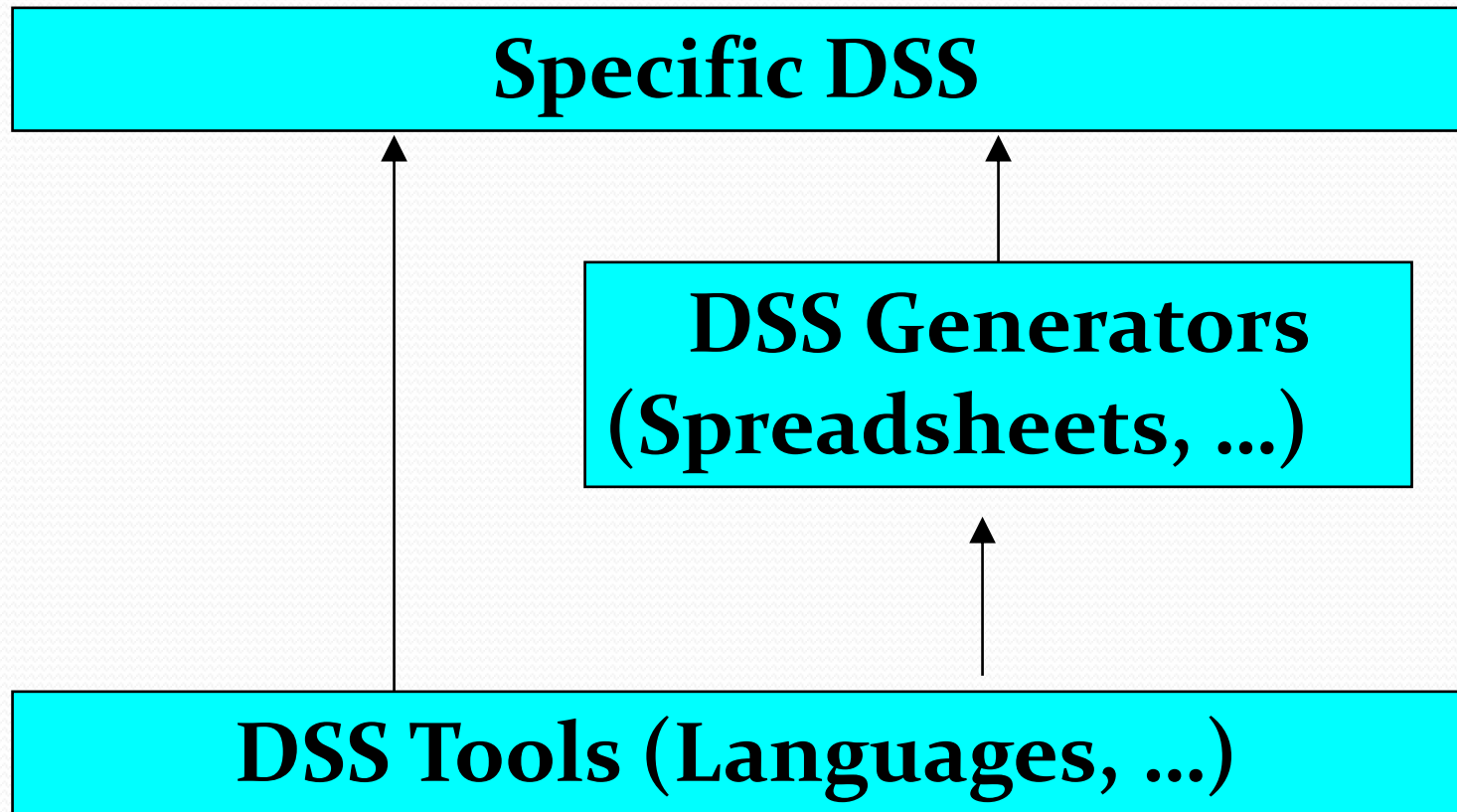
- Specific DSS (Application DSS)
 - “Final Product” atau aplikasi DSS yang nyata-nya menyelesaikan pekerjaan yang kita inginkan disebut dengan specific DSS (SDSS)
- DSS Generator
 - Software pengembangan terintegrasi yang menyediakan sekumpulan kemampuan untuk membangun specific DSS secara cepat, tak mahal, dan mudah. Contoh : Microsoft Excel



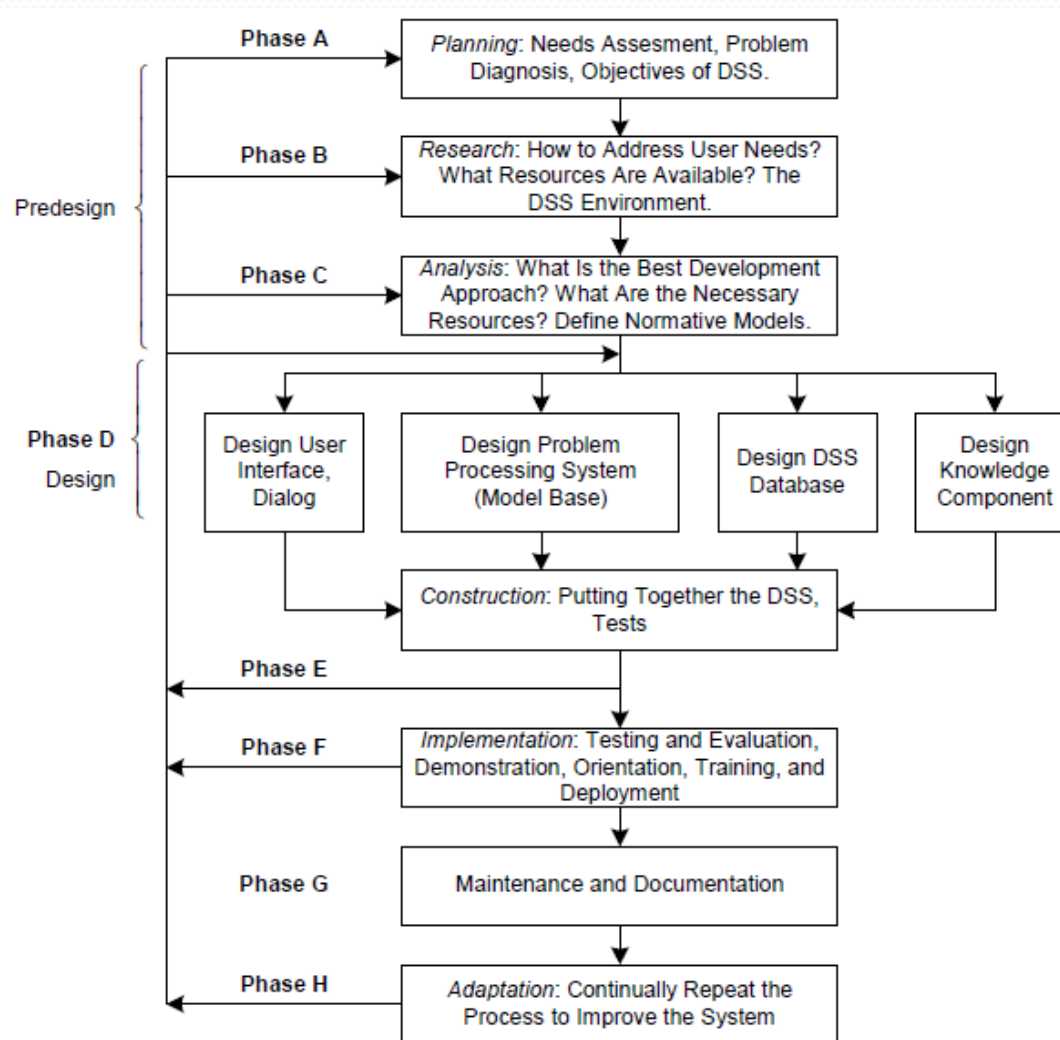
- DSS Tools

- Level terendah dari teknologi DSS adalah software utility atau tools. Elemen ini membantu pengembangan baik DSS generators atau SDSS.

Tingkat Teknologi DSS



Fase Pengembangan DSS



1. Perencanaan

- Perencanaan.
 - Merumuskan kerangka dan ruang lingkup SPK
 - Persyaratan unjuk kerja
 - Memilih konsep-konsep & menganalisis model pembuatan keputusan yang relevan dengan tujuan SPK.

Langkah ini menentukan pemilihan jenis SPK yang akan dirancang dan metode pendekatan yang dipergunakan.

2. Penelitian

Berhubungan dengan pencarian data serta sumber daya yang tersedia



3. Analisis & Perancangan Konsep

Penentuan teknik pendekatan yang akan dilakukan serta sumber daya yang dibutuhkan



4. Perancangan

- Melakukan perancangan ketiga subsistem utama SPK
 - Subsistem database
 - Model
 - Dialog.



5. Konstruksi

- Merupakan kelanjutan dari perancangan dimana ketiga subsistem yang telah dirancang digabungkan menjadi suatu SPK



6. Implementasi

- Menerapkan SPk yang dibangun. Pada tahap dilakukan testing, evaluasi, penampilan, orientasi, pelatihan dan penyebaran



7. Pemeliharaan

- Tahapan yang dilakukan terus menerus untuk mempertahankan keandalan sistem

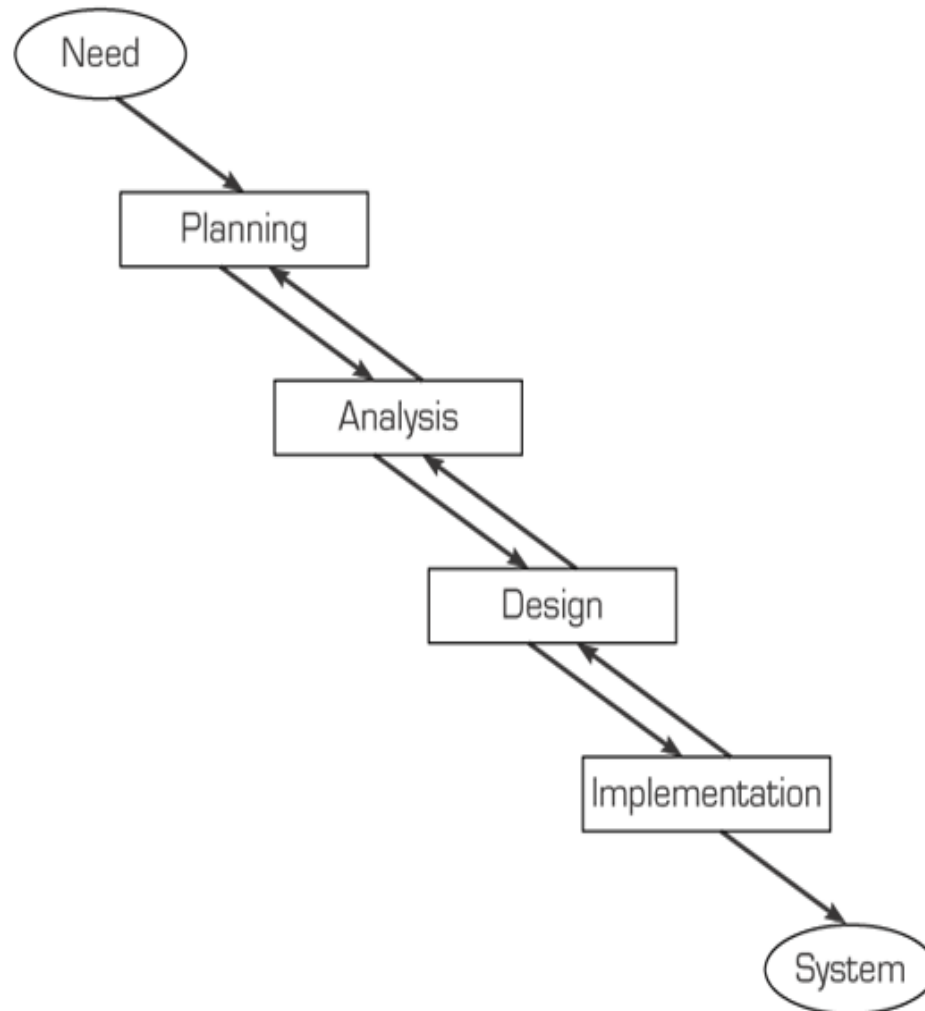


8. Adaptasi

- Melakukan pengulangan terhadap tahapan diatas sebagai tanggapan terhadap perubahan kebutuhan “pemakai”.



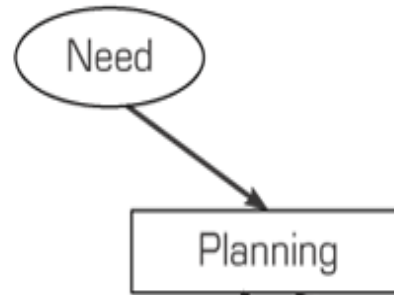
Traditional System Development Life Cycle



Need

PADI

PADI



- Perencanaan
 - Analisis studi kelayakan
 - Kelayakan teknis
 - Kelayakan biaya
 - Kelayakan organisasional

PADI

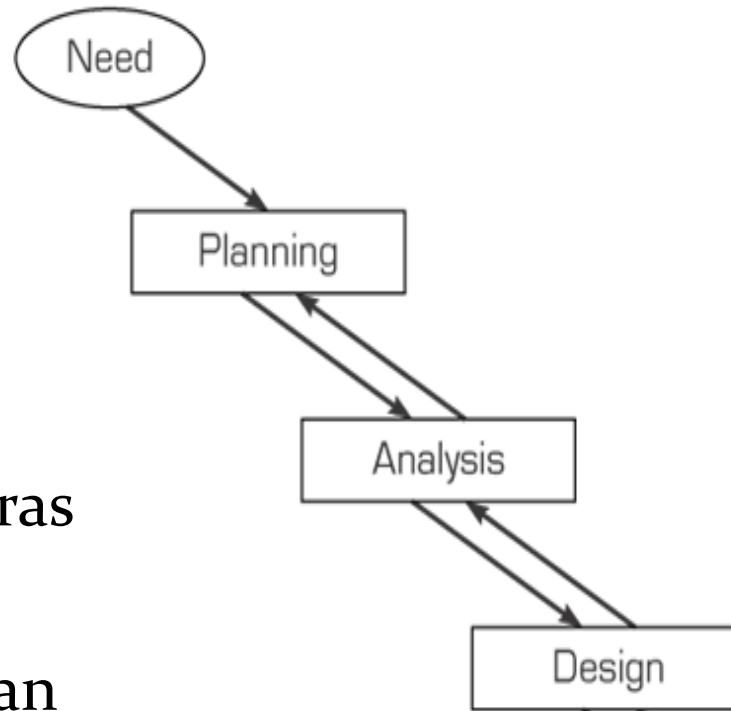
- Analisis

- Siapa pengguna??
- Apa yang akan dicapai sistem??
- Dimana sistem akan dijalankan??
- Kapan sistem akan dijalankan??



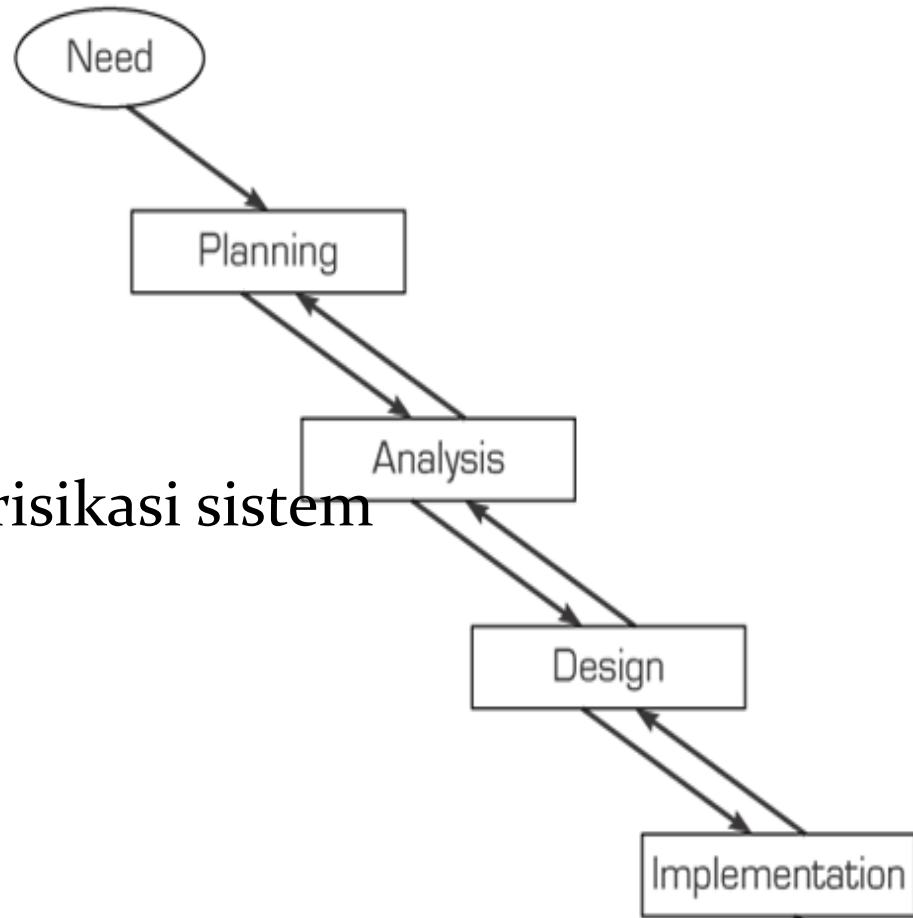
PADI

- Desain
 - Detail perangkat keras
 - Perangkat lunak
 - Infrastruktur jaringan
 - Antar muka pengguna
 - Form
 - Display
 - Program dan laporan
 - Database dan file

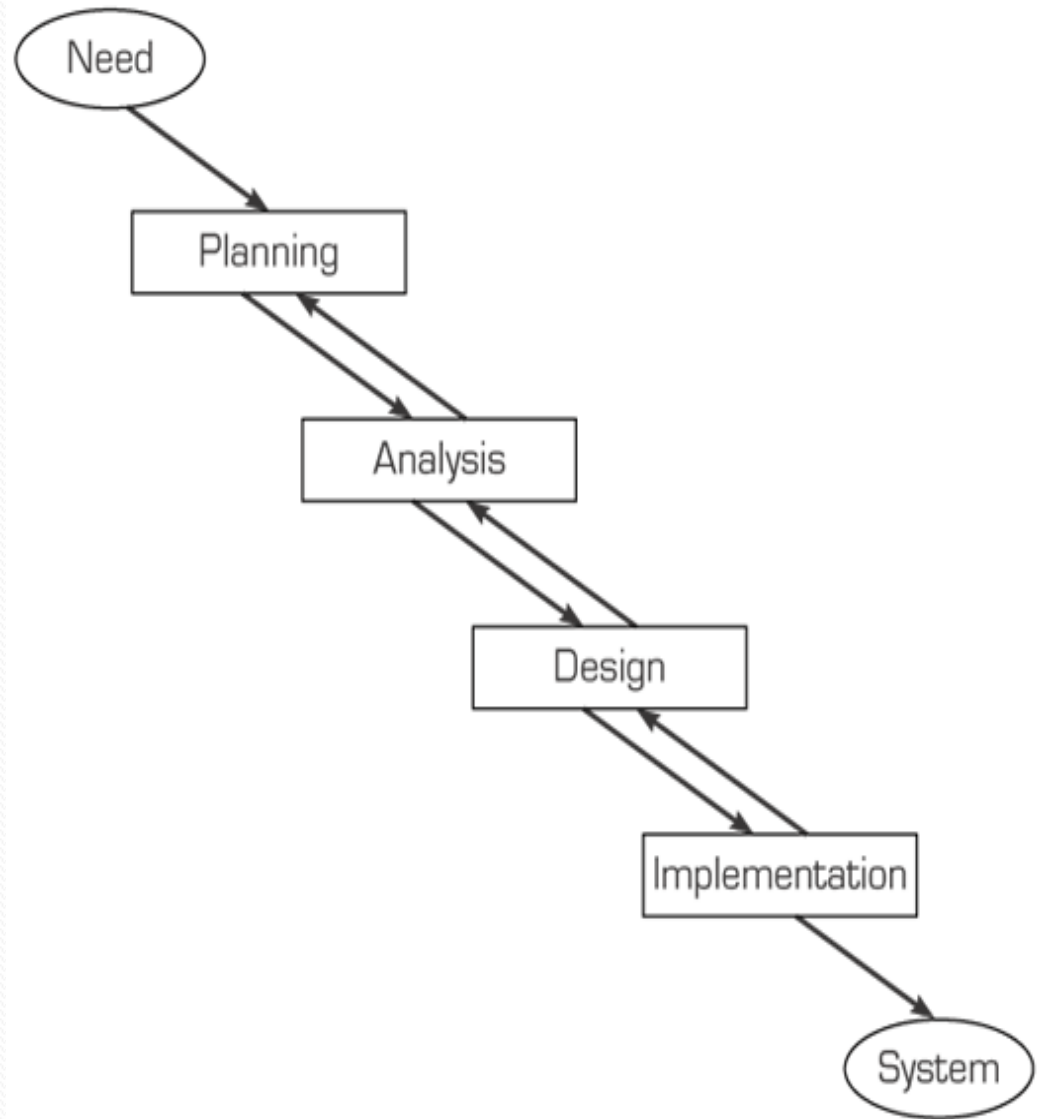


PADI

- Implementasi
 - Pengujian untuk verifikasi sistem



PADI



Peranti CASE

- Perangkat lunak aplikasi untuk membantu analisis sistem dalam mengerjakan setiap tahapan dalam pengembangan sistem.
 - Upper CASE
 - Menciptakan diagram sistem
 - Lower Case
 - Mengelola diagram dan menghasilkan kode untuk tabel database
 - Case Terintegrasi
 - Kombinasi Upper CASE dan Lower CASE

Manajemen Proyek

- Mengembangkan dari sistem yang berjalan
- Definisikan ruang lingkup
- Mengelola perubahan dan penambahan ruang lingkup secara bertahap
- Usahakan memperoleh dukungan dari top manajemen
- Menyusun jadwal tahapan dan anggaran berdasarkan goal yang realistis
- Ikutsertakan pemakai
- Lakukan dokumentasi sejak awal hingga akhir proyek

Manajemen Proyek

- Pemimpin tim perlu memiliki keterampilan yang baik
- PENTING untuk memahami faktor yang mendorong terjadinya kegagalan

Kegagalan Implementasi

- terbatasnya keikutsertaan stakeholder
- requirement tidak lengkap
- lingkup proyek bertambah terus menerus
- harapan kurang realistis
- personil kunci tidak melibatkan diri lagi
- kurangnya keahlian pelaksana
- SDM tidak memadai
- Teknologi baru

Faktor kegagalan dipengaruhi lingkungan atau proses

- Lingkungan
 - faktor budaya organisasi
 - tidak memperoleh dukungan dari top manajemen
 - attitude dari pemakai dan analis
 - kurangnya pengalaman pemakai
 - minimnya kemampuan tim pengembang sistem
- Proses
 - pendidikan,
 - Dukungan dan keterlibatan user
 - pelatihan

Metodologi Alternatif

1. Pengembangan secara Paralel
 - Mengembangkan bagian-bagian sistem secara paralel, selanjutnya diintegrasikan pada waktu akan diimplementasikan RAD
2. RAD (Rapid Application and Development)
 - Pengembangan yang cepat tetapi untuk fungsi-fungsi yang terbatas
 - Phase Development
 - Prototyping
 - Throwaway Prototyping

TRADITIONAL



RAD



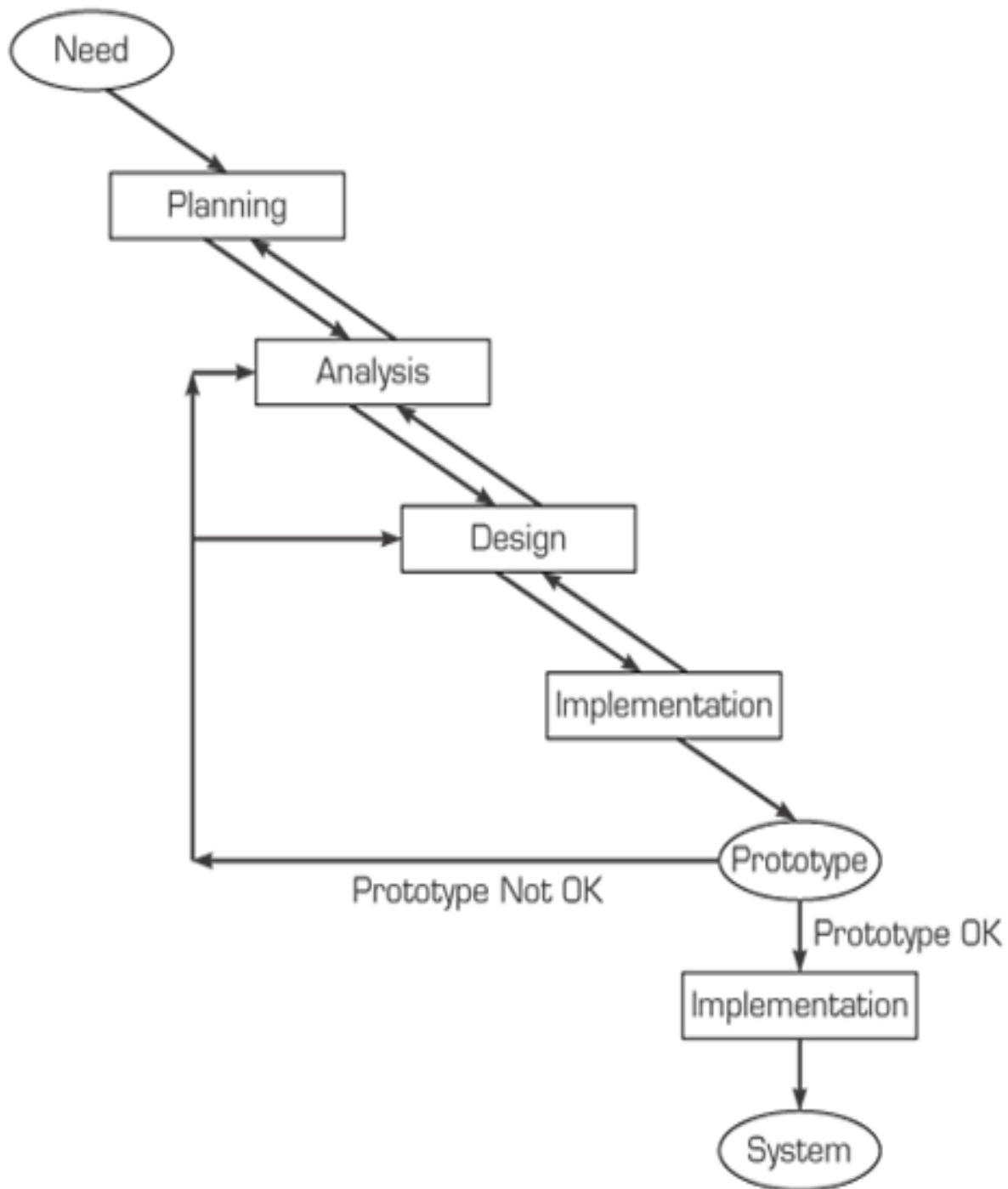
Phase Development

- Pengembangan secara berurutan bagian-bagian dari proyek



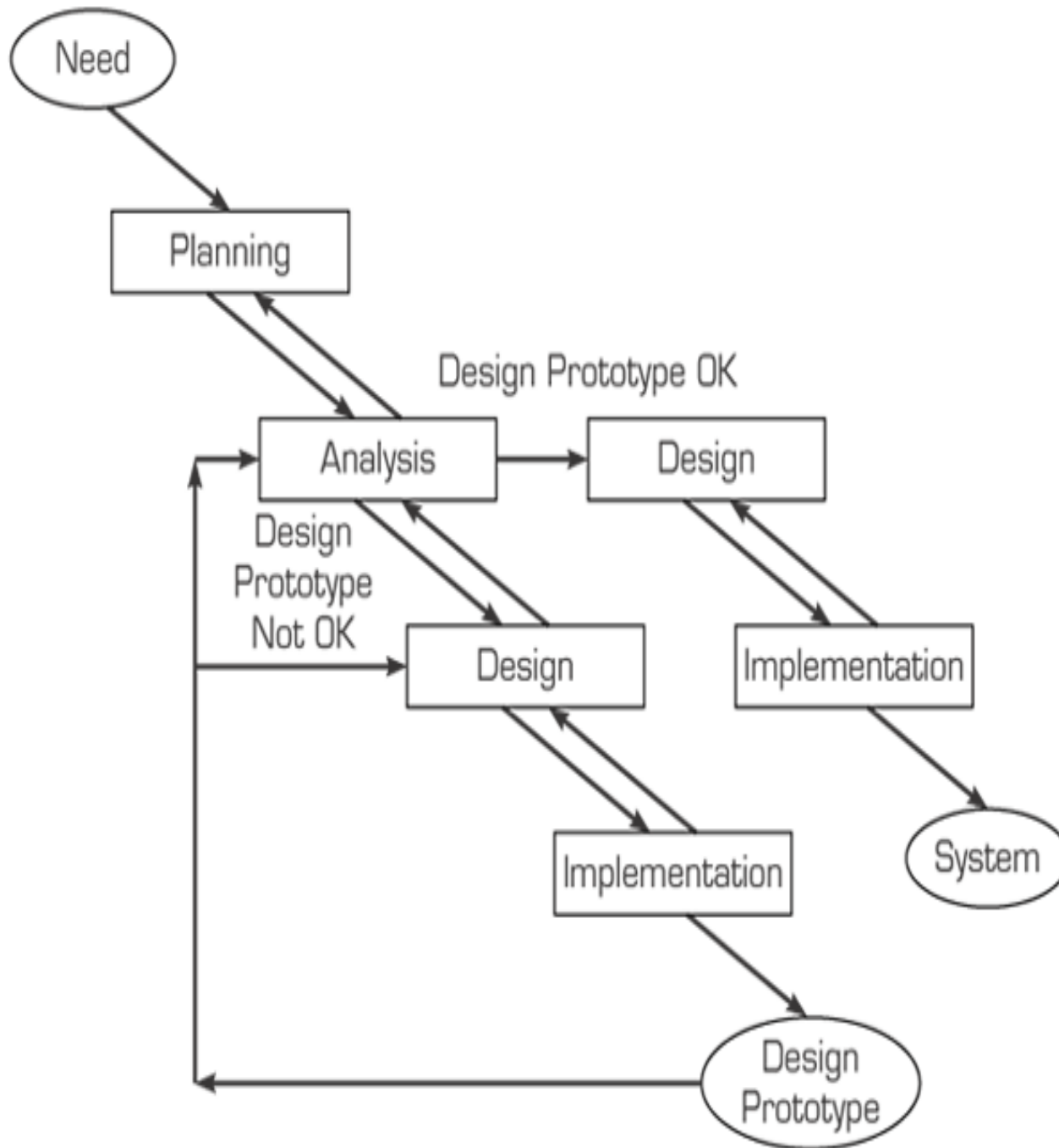
Prototyping


- Analisis, desain, implementasi secara bersama dan iteratif
- Hasil pengembangannya segera dan dapat dilihat pemakai; sehingga mereka dapat memberi masukan untuk menyempurnakan sistem



Throwaway Prototyping

- Gabungan SDLC dengan prototyping
- Pengembangan dimulai dari bagian yang paling sederhana dari sistem
- Setelah selesai prototip dapat dilanjutkan dengan pengembangan sistem sebenarnya



- 
- DSS harus dibangun dengan feedback yang pendek dan cepat dari pemakai untuk memastikan bahwa pengembangan telah berlangsung secara tepat

Keuntungan Prototyping

- Waktu pengembangan pendek/cepat
- Waktu reaksi pengguna pendek/singkat
- Meningkatkan pemahaman pengguna mengenai sistem, kebutuhan informasinya dan kemampuannya
- Biaya rendah

- Cara pengembangan DSS tidak dilakukan dengan membuat sistem yang akhirnya nanti "berjalan" seperti sistem tradisional
- Cara pengembangan DSS akan menghasilkan instalasi proses yang adaptif dimana pembuat keputusan dan set "kemampuan" sistem informasi berinteraksi untuk memecahkan masalah sambil merespon perubahan yang berasal dari berbagai sumber