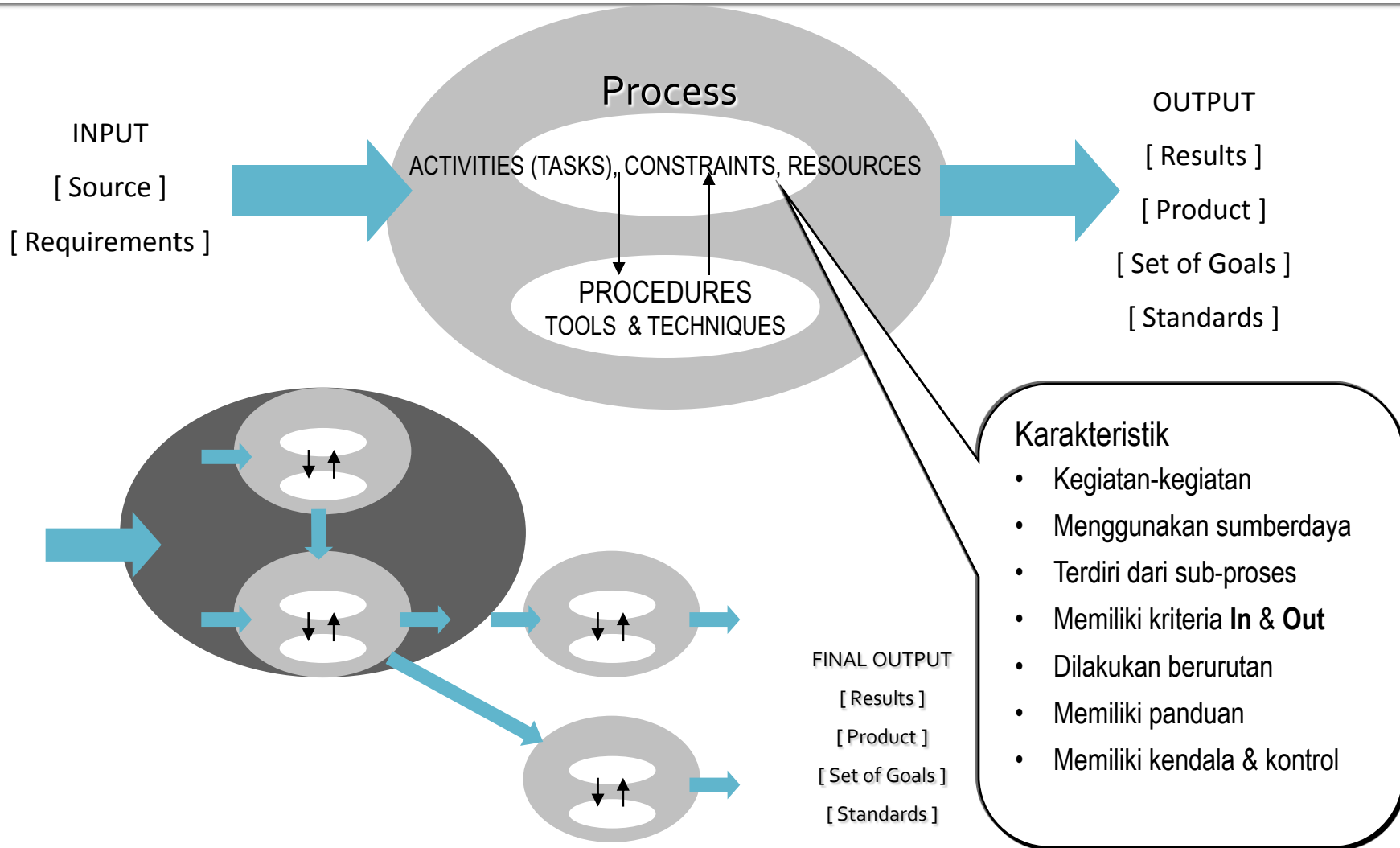


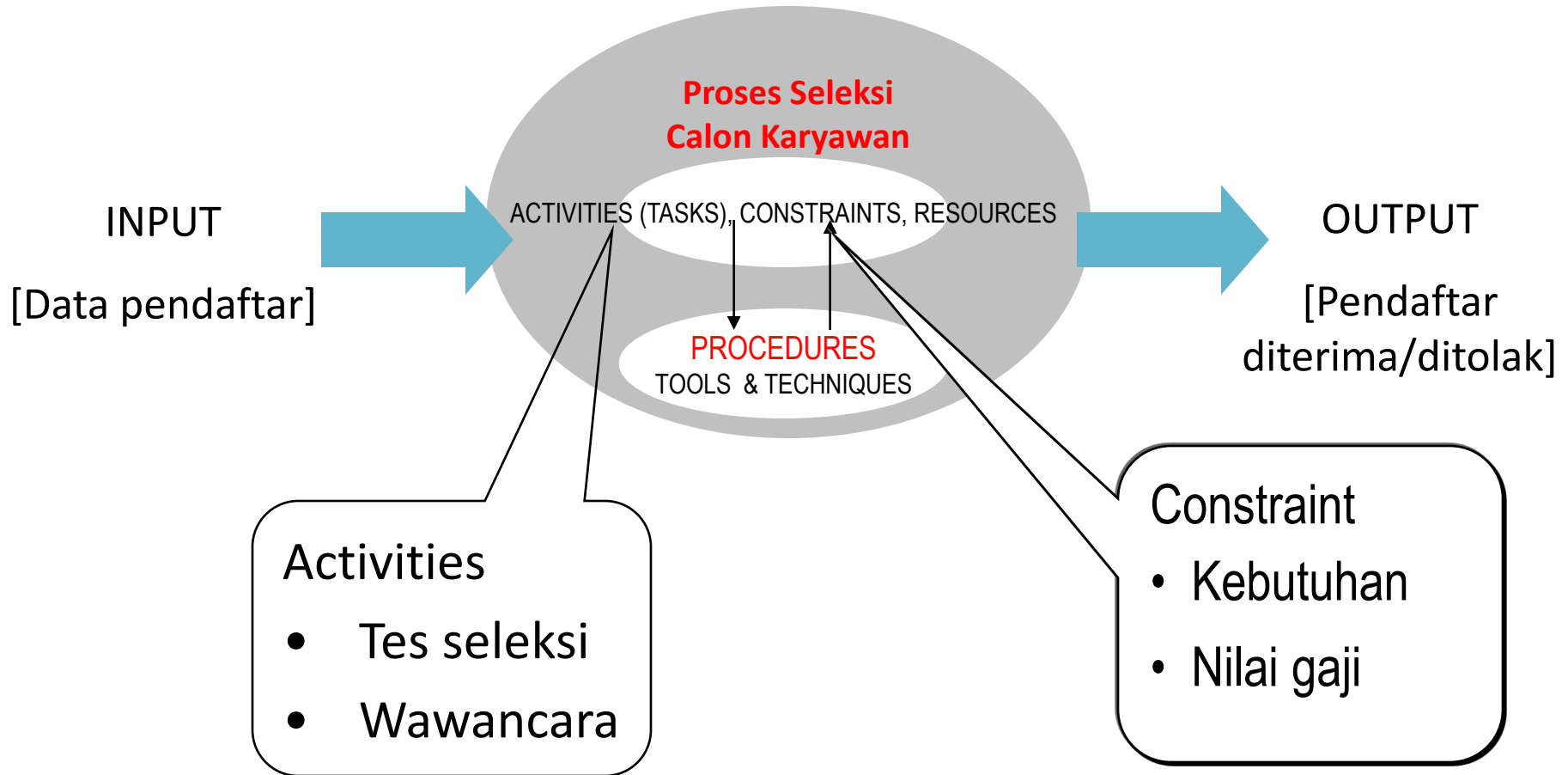
<http://www.brigidaarie.com>

Pemodelan Process & Life Cycle

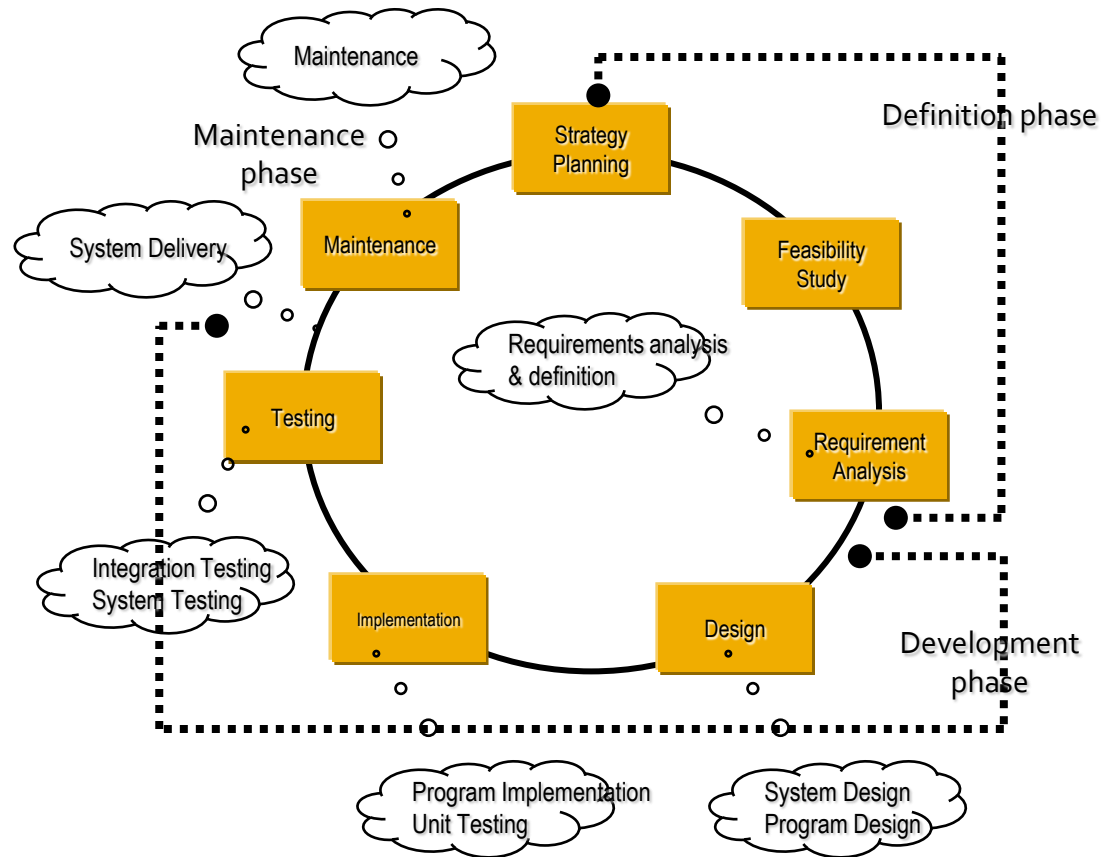
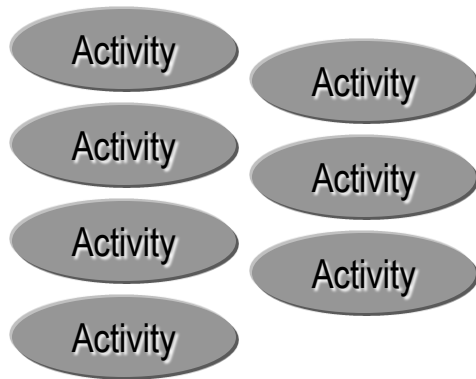
Proses



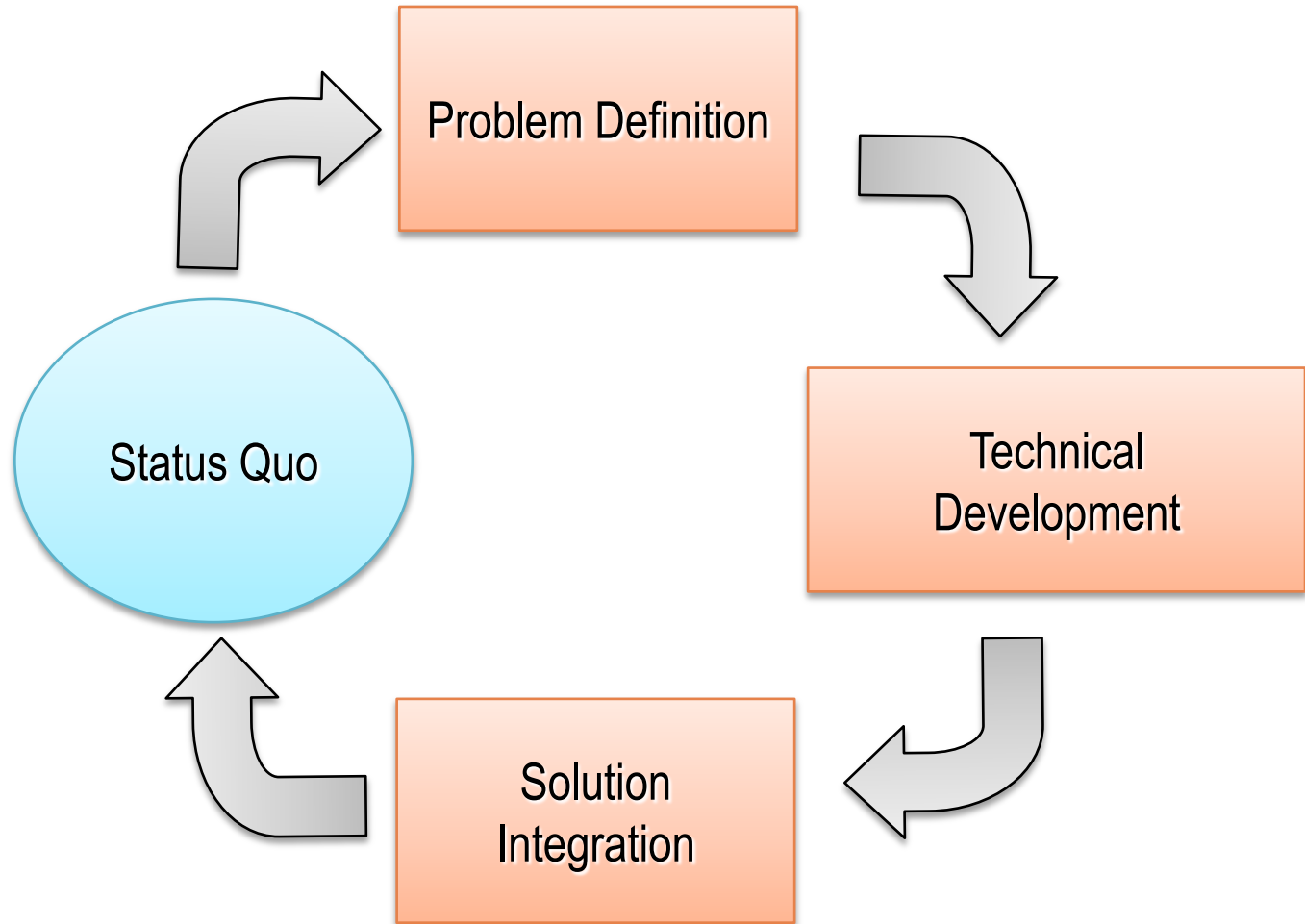
Contoh Proses



Proses dalam setiap tahap



Proses Software



Waterfall Model

- “V” Model

Prototyping Model

- Throw away
- Evolutionary

RAD Model

- Incremental Model
- Iterative Model

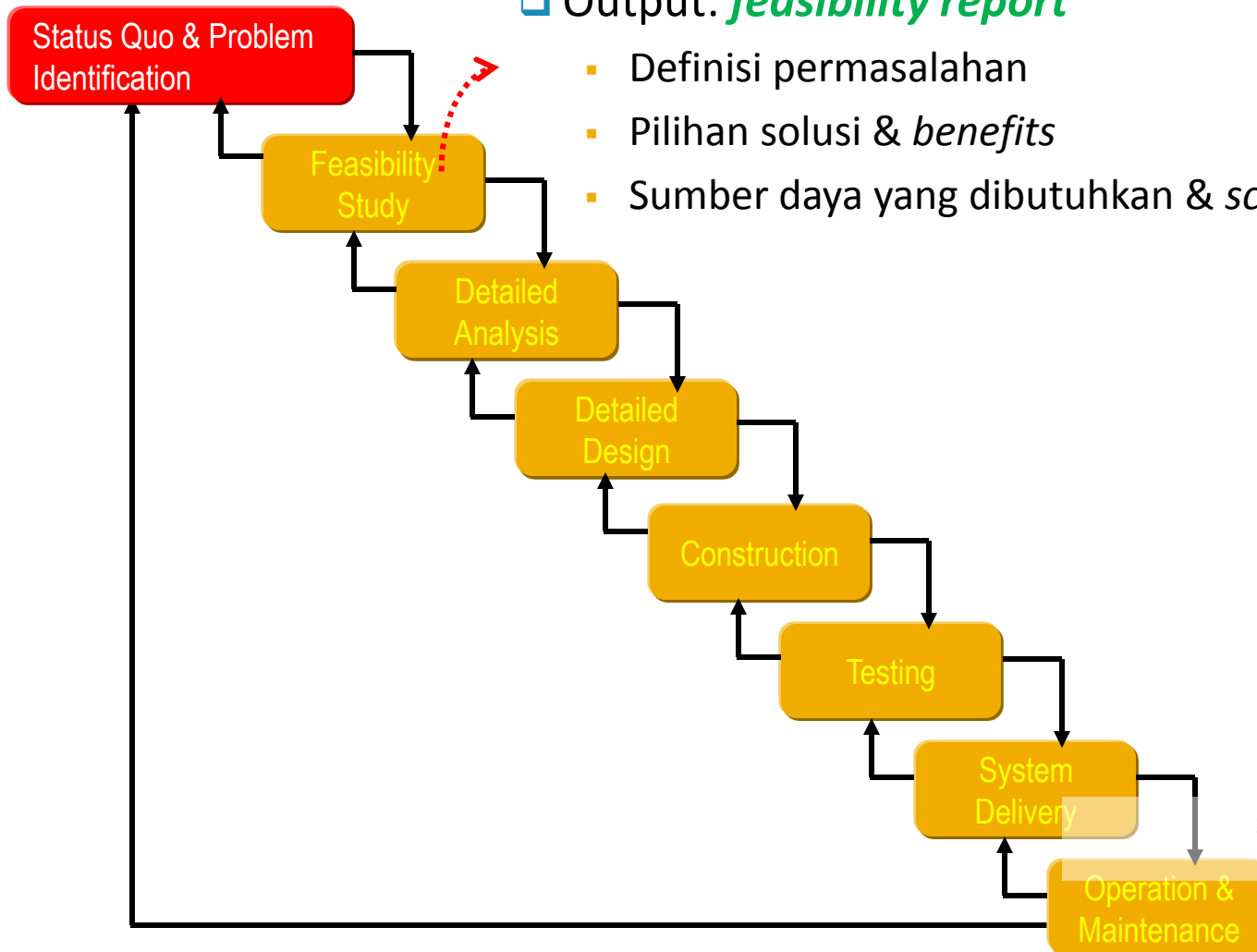
Spiral Model

Model Waterfall

- ❑ Evaluasi *costs* & *benefits*

- ❑ Output: *feasibility report*

- Definisi permasalahan
- Pilihan solusi & *benefits*
- Sumber daya yang dibutuhkan & *scheduling*



System / Information engineering
[Roger. S. Pressman]

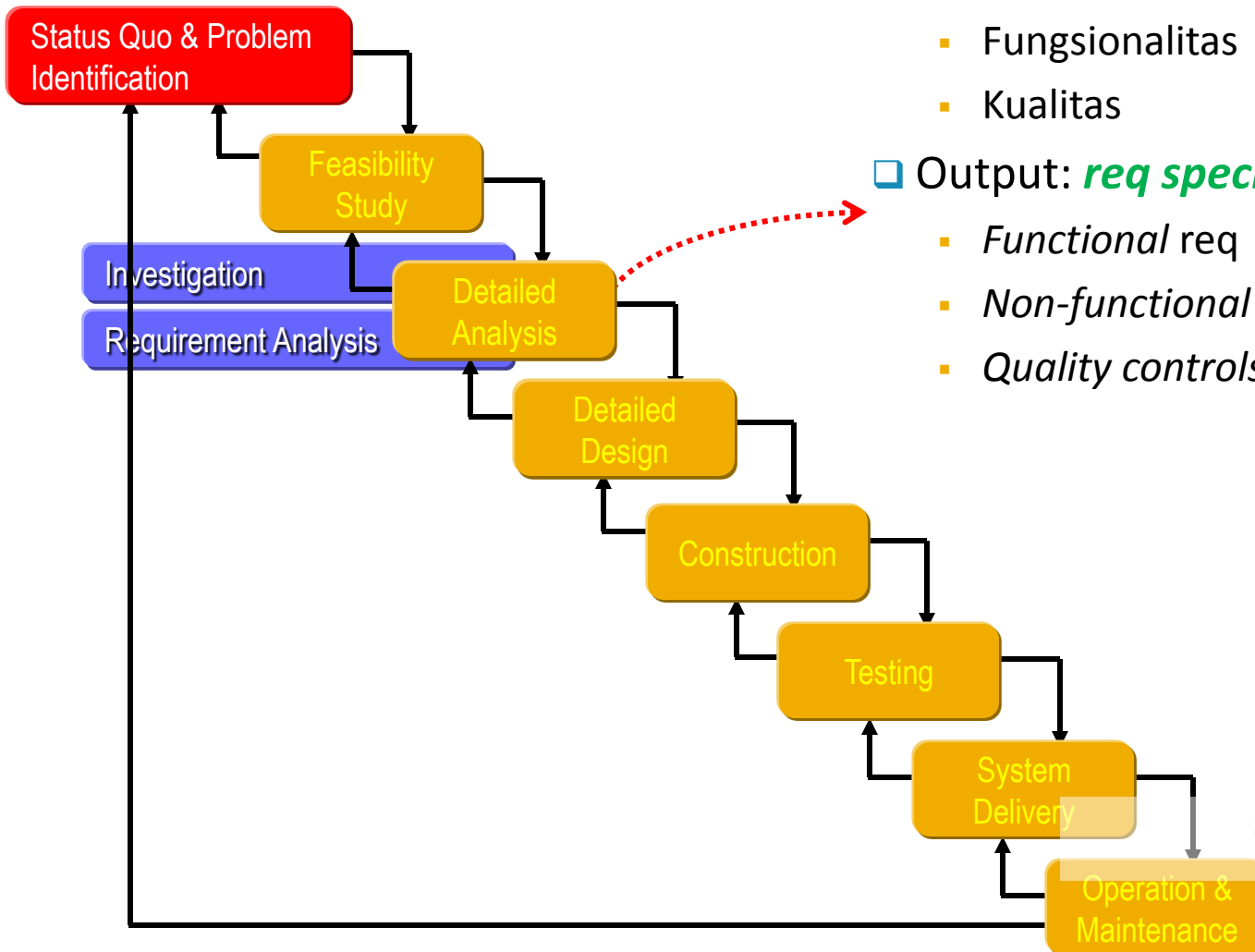
Model Waterfall

□ Identifikasi *requirements*

- Fungsionalitas
- Kualitas

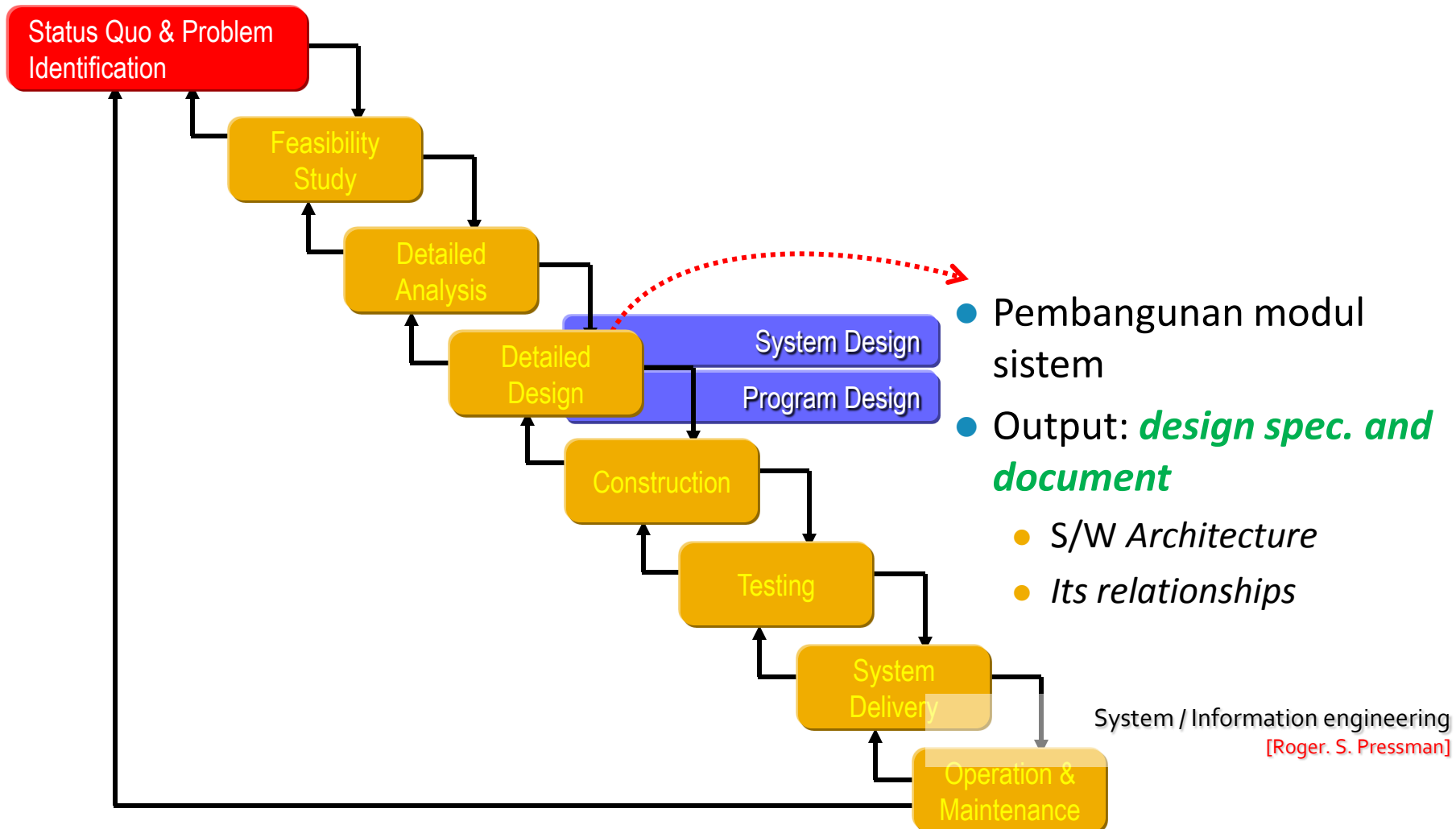
□ Output: *req specification document*

- *Functional req*
- *Non-functional req*
- *Quality controls*

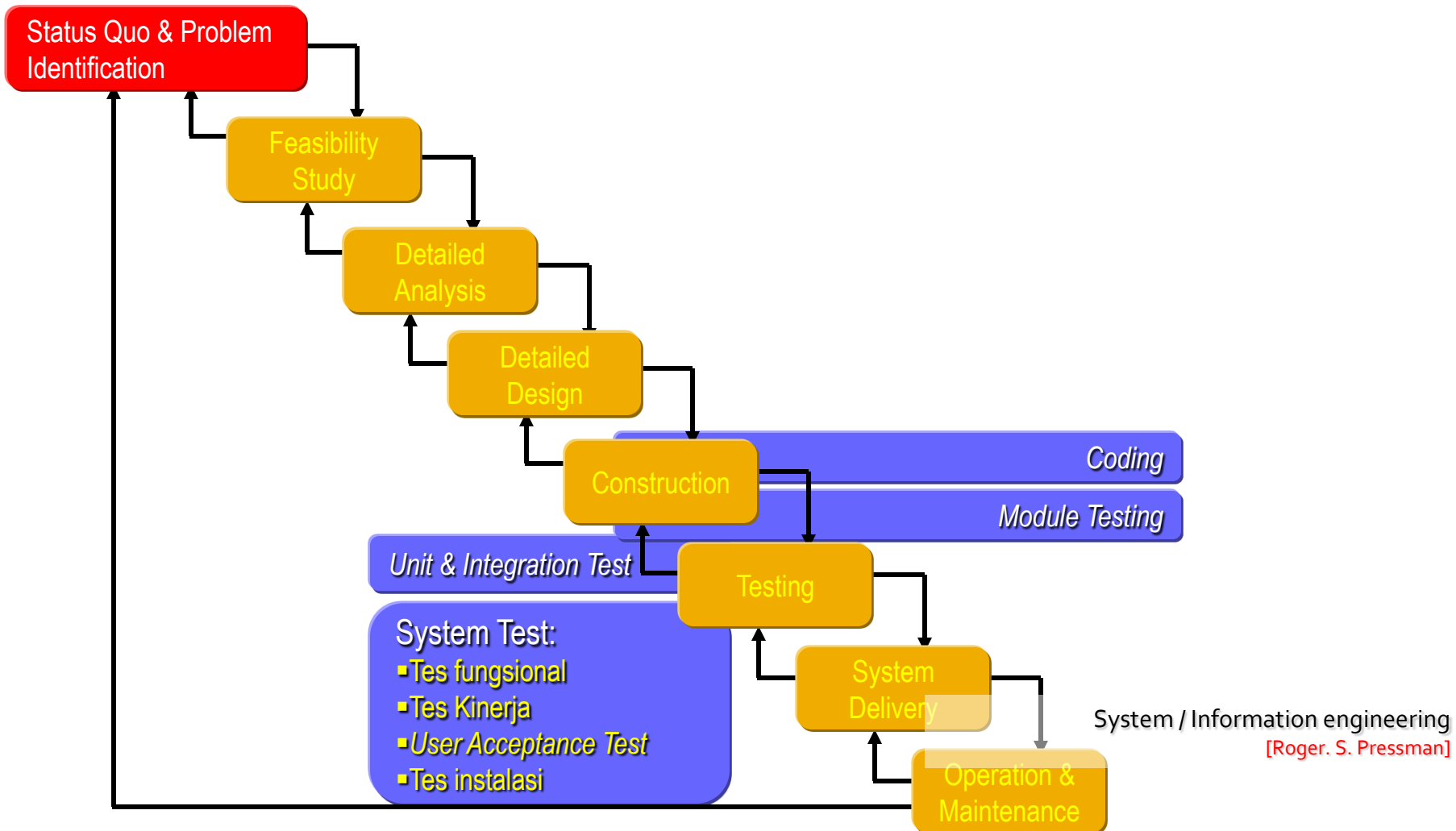


System / Information engineering
[Roger. S. Pressman]

Model Waterfall



Model Waterfall



Keuntungan Waterfall

Terperinci
dan detail

Cocok untuk
developer pemula

Dinyatakan
sebelum masuk
tahap berikutnya

Berfokus pada
dokumentasi

Kekurangan Waterfall

Jarang sesuai urutan yang direncanakan

Iterations → *confusion*

Ketidakpastian (Kesulitan memetakan SEMUA *user req*)

- Apakah sistem memenuhi harapan *user*?

Ketidakjelasan proses pengembangan

- Tidak ada pengecekan kembali

Produk baru muncul diakhir tahapan

Tidak terdapat pedoman, jika terjadi perubahan

Prototyping Model

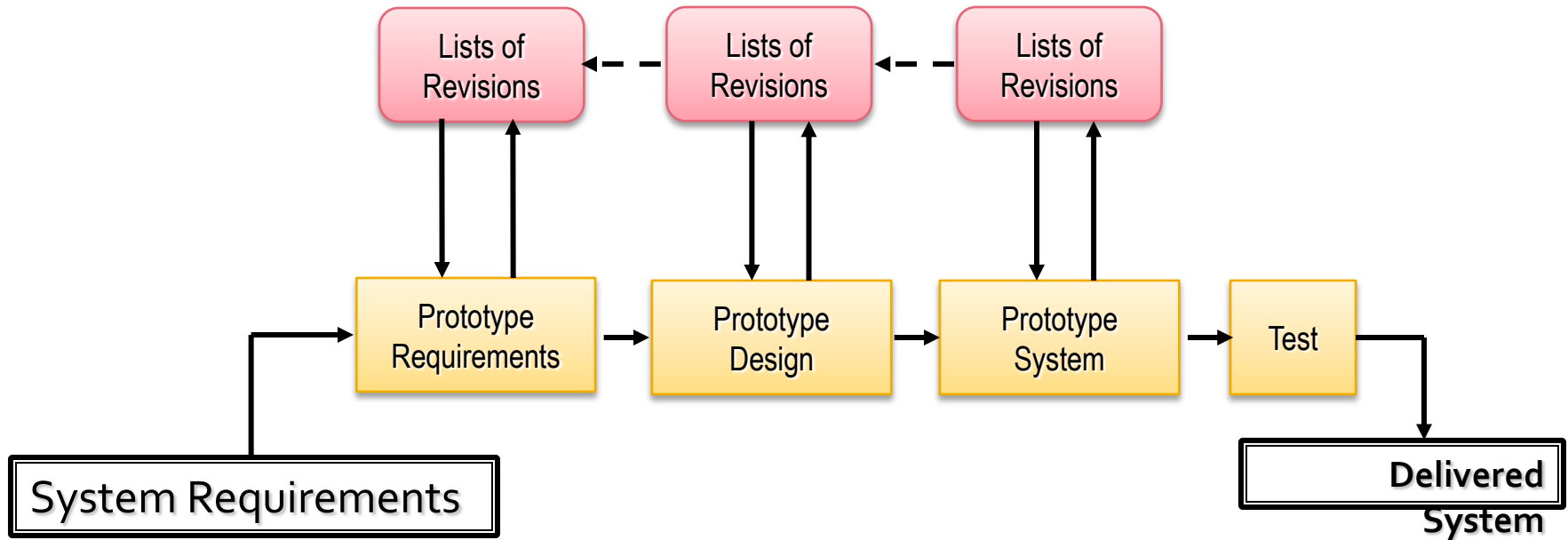
Pengembangan produk sebagian

- Pemeriksaan thd *Customers & developers*
- Memeriksa setiap aspek sistem yg diusulkan
- Kesesuaian dan Ketepatan

Cepat → Pembangunan (*all / part*)

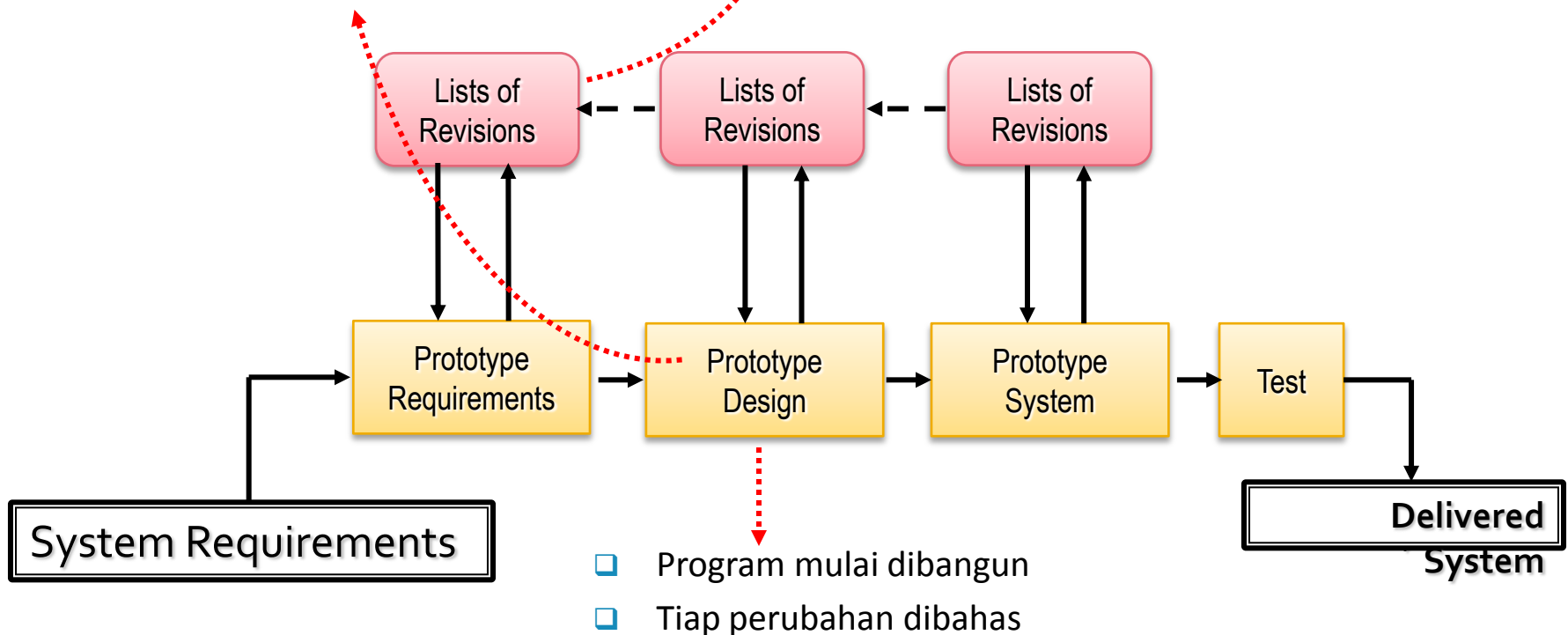
Pemahaman dan klarifikasi masalah

Prototyping Model

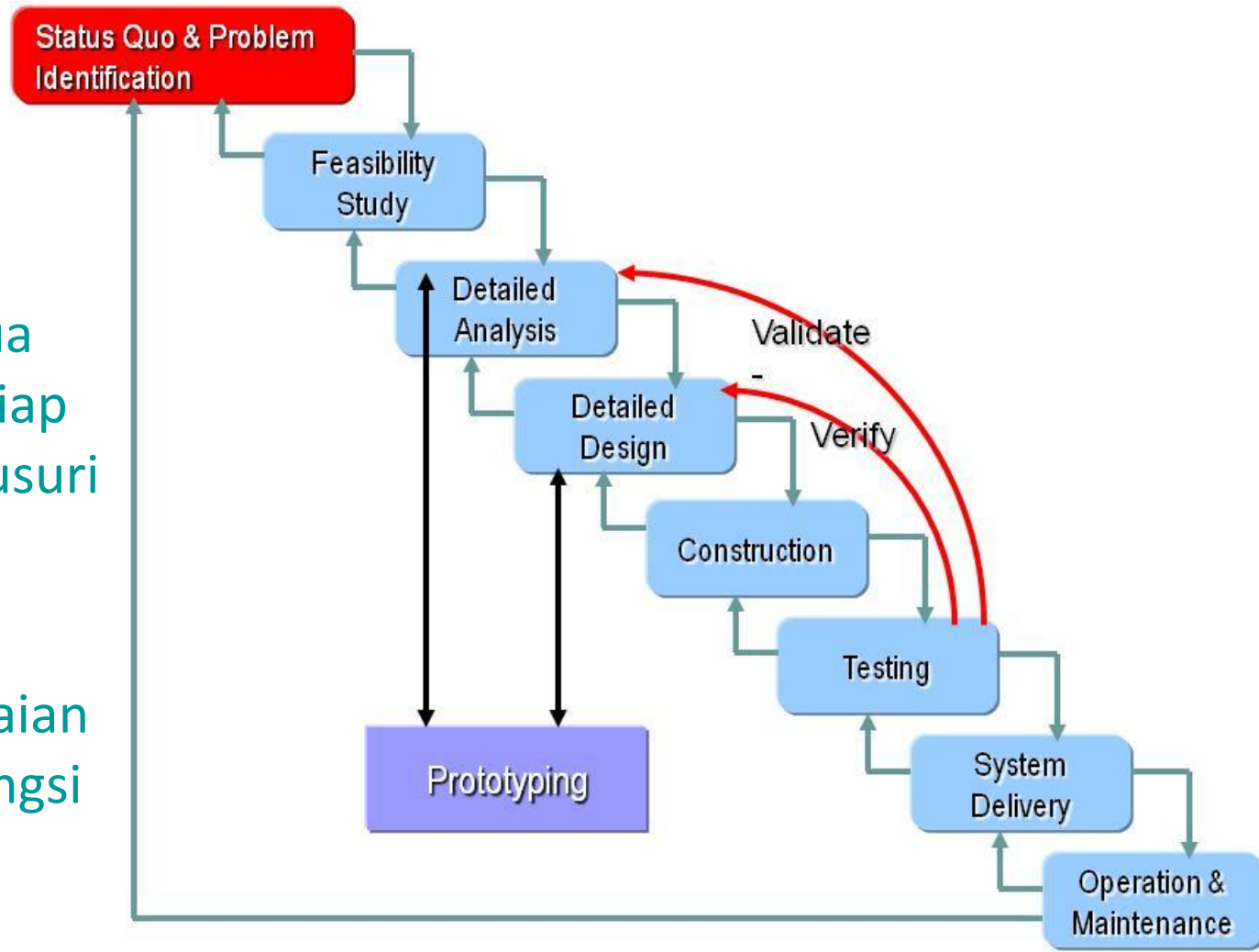


Prototyping Model

- ❑ Sediakan alternatif desain
- ❑ Direvisi sampai sesuai
- ❑ Alternatif desain mengungkap permasalahan
- ❑ Kembali dalam aktifitas req



Prototype



- Validation

Memastikan semua req terpenuhi, setiap fungsi dapat ditelusuri sesuai dengan req

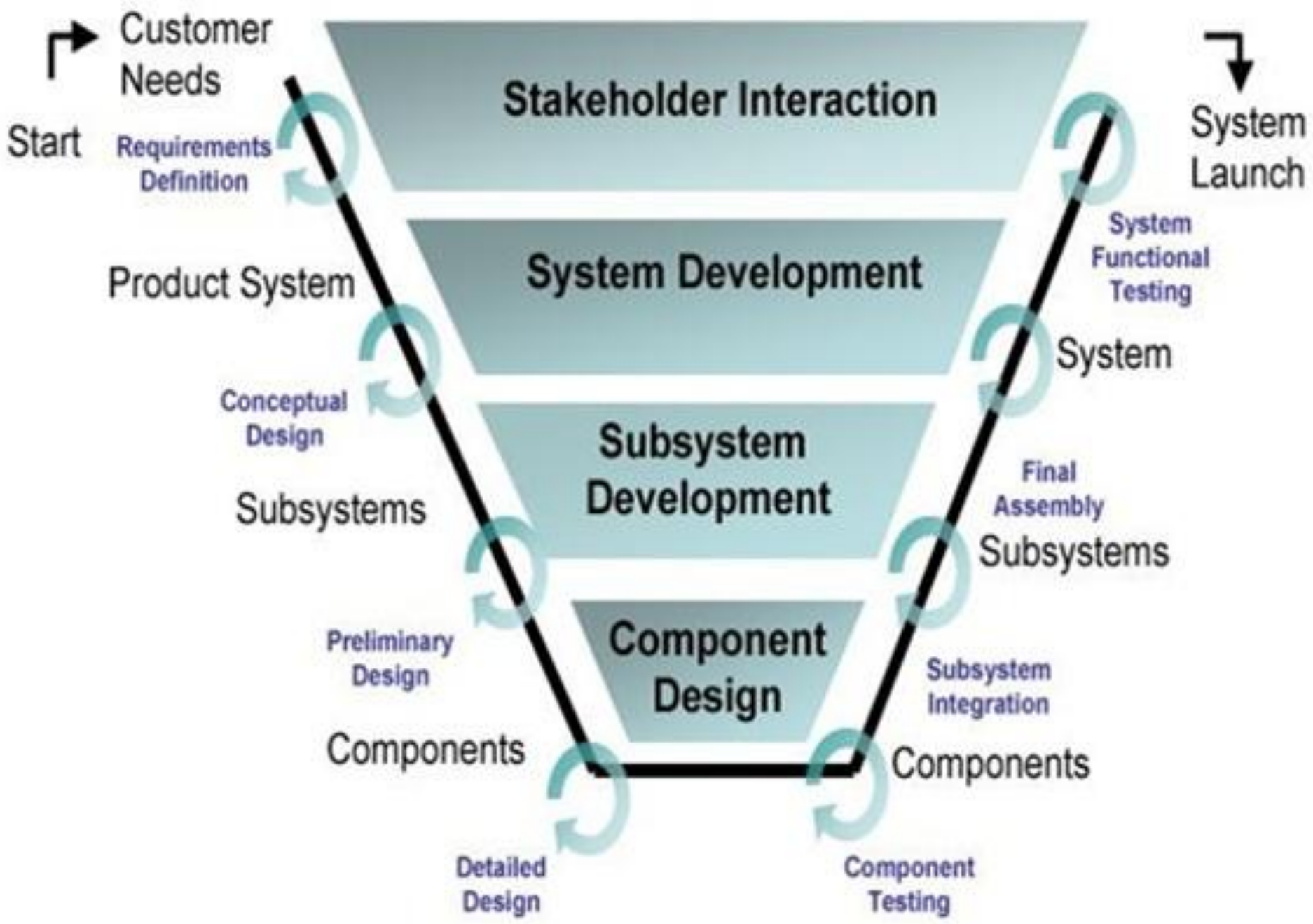
- Verification

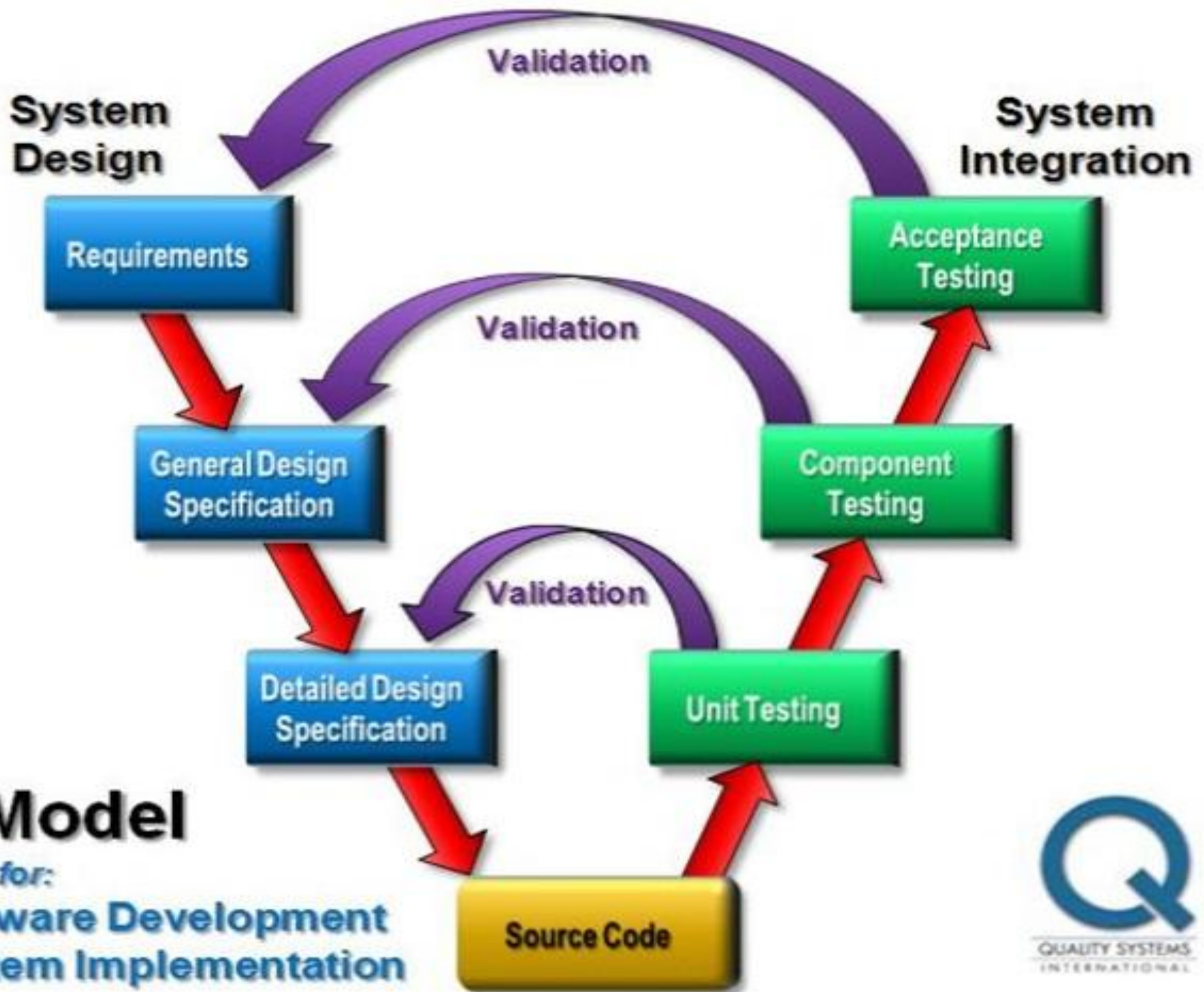
Mengecek kesesuaian masing-masing fungsi (*quality*)

V Model

Masalah yang ditemukan **langsung** diperbaiki, desain dan *coding* sebelum tesing dilakukan kembali

V-model fokus pada **aktifitas** dan **perbaikan** berkesinambungan





V-Model

Used for:
Software Development
System Implementation

Rapid Application Development

Incremental

- End-users melakukan evaluasi pada setiap tahap

Siklus pengembangan pendek

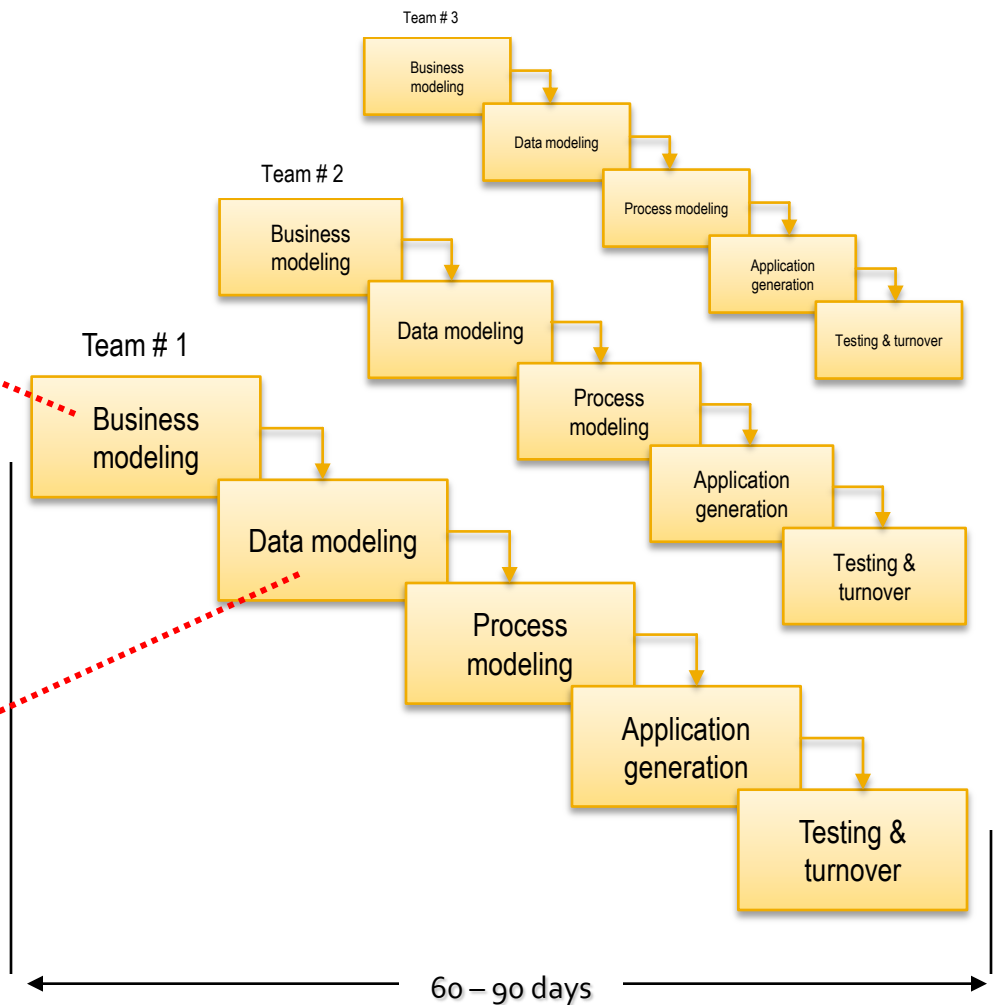
- Adaptasi
 - Model *linear (Waterfall)*
- Perkembangan yang pesat
 - Menggunakan pembangunan *component-based*
- Periode singkat
 - 60 – 90 hari

Rapid Application Development

□ Pemodelan fungsi bisnis (*information flow*)

- What info ? → business process
- What info is generated ?
- Who processes it ?
- Who generates it ?
- Where does info go to next ?

- Aliran informasi disempurnakan menjadi obyek data
- Karakteristik obyek data diidentifikasi
- Hubungan antara obyek data didefinisikan

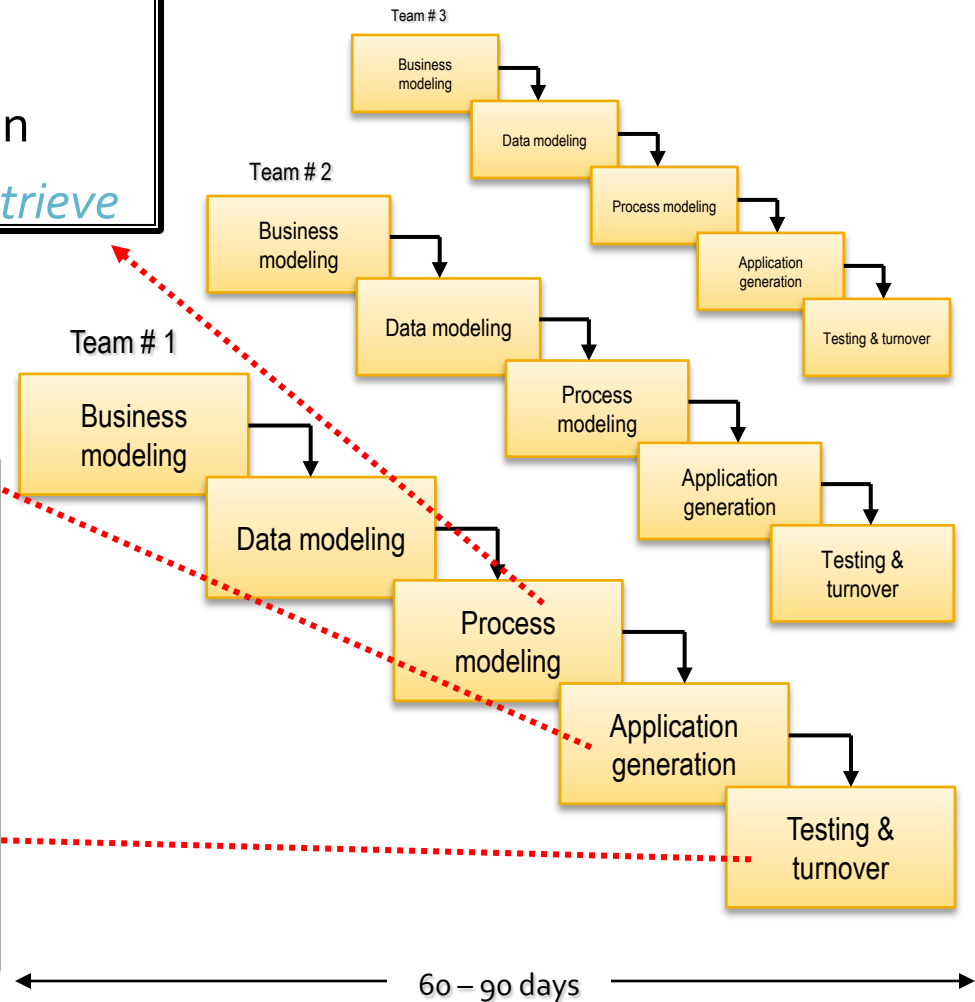


Rapid Application Development

- ❑ Obyek data berubah utk melaksanakan fungsi bisnis
- ❑ Dibuat deskripsi pemrosesan
 - *Add, Modify, Delete & Retrieve*

- ❑ Menggunakan 4th *Generation techniques* (misal: *Code generators*)
 - *Reusable components*
 - *Automated tools*

- ❑ Menekankan penggunaan kembali
 - Banyak komponen yg telah diuji
 - Komponen baru yg akan diuji



Incremental Model

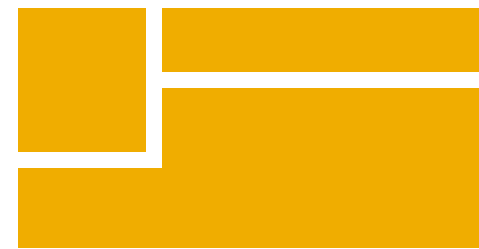
- ❖ *Requirements* **dipartisi menjadi subsistem** dengan fungsionalitasnya



Use Case 1



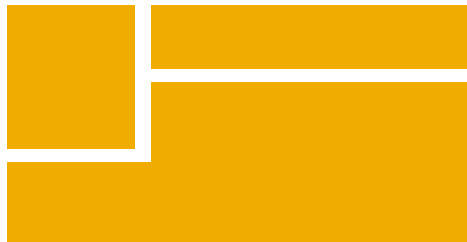
Use Case 1
Use Case 2



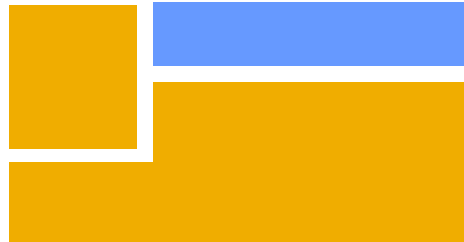
Use Case 1
Use Case 2
Use Case 3

Iterative Model

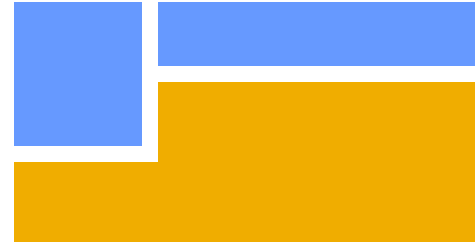
- ❖ Memberikan *full-system* diawal, kemudian **merubah fungsionalitas** dari setiap subsistem dengan setiap rilis terbaru



Use Case 1
Use Case 2
Use Case 3
(release 1)



Use Case 1
Use Case 2
Use Case 3
(release 2)



Use Case 1
Use Case 2
Use Case 3
(release 3)

Sesuai utk proyek singkat

Kesulitan pengujian minimum

- *Reusable components*

Pembangunan yg cepat

- *Automated tools*

Pendekatan *straight-forward*



Keuntungan-RAD

Kekurangan

Tidak sesuai utk proyek besar

- Melibatkan banyak tim yg bekerja

Komitmen penuh tim → *deadlines*

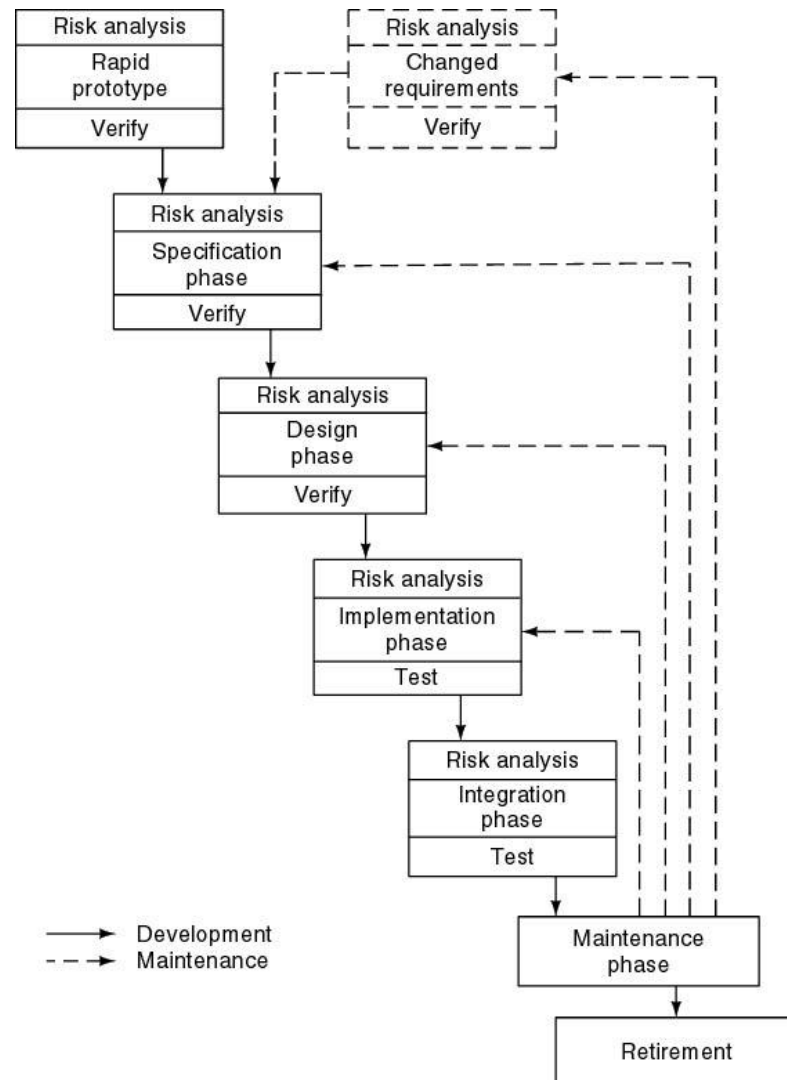
Tidak sesuai untuk:

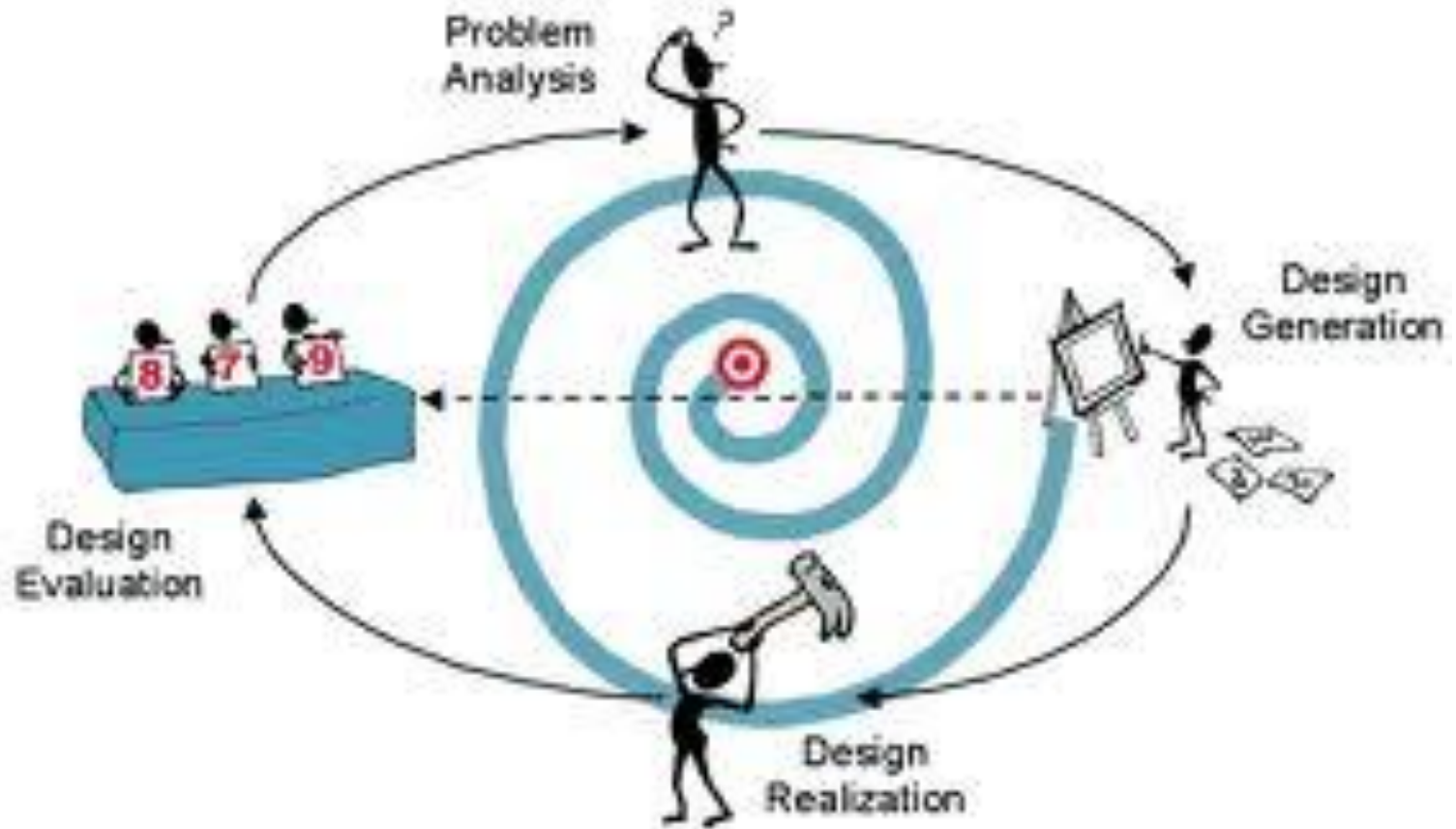
- *Systems* → *fine tuning*
- *System* → interaksi teknologi baru
- *System* → interaksi sistem yg lainnya

Spiral Model

Bentuk Sederhana

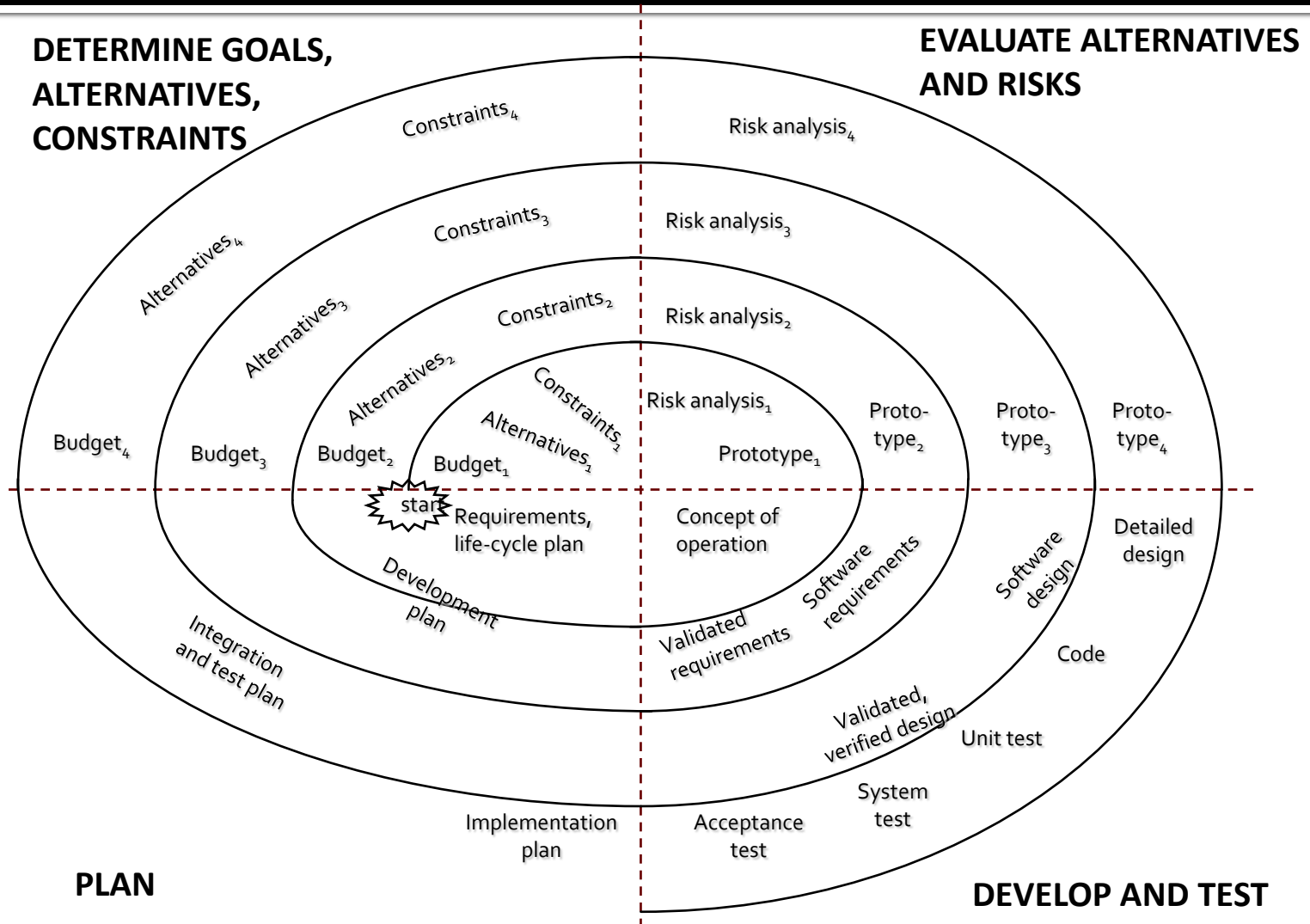
- Model *waterfall* **plus analisis resiko** sebelum setiap tahap
- Resiko tdk dpt diselesaikan -> **Proyek dihentikan**





Spiral Model

Spiral Model



Spiral Model

Pengaturan Tujuan

- Tujuan khusus untuk setiap fase diidentifikasi

Penilaian & Pengurangan Resiko

- Resiko dinilai dan kegiatan dilakukan utk mengurangi resiko

Pengembangan dan Validasi

- Model pengembangan dipilih

Perencanaan

- Proyek telah dikaji, selanjutnya dilakukan perencanaan