

REKAYASA PERANGKAT LUNAK

3 sks

Sri Rezeki Candra Nursari

reezeki2011.wordpress.com

Referensi

- Rekayasa Perangkat Lunak – Pendekatan Praktisi, Roger S. Pressman, Ph.D, Andi Yogyakarta, 2012 – Buku 1
- Rekayasa Perangkat Lunak – Pendekatan Praktisi, Roger S. Pressman, Ph.D, Andi Yogyakarta, 2012 – Buku 2
- Rekayasa Perangkat Lunak – Analisa Kebutuhan Dalam, Daniel Siahaan, Andi Yogyakarta, 2012

Materi

- Perancangan, pembuatan, pengujian dan perawatan perangkat lunak serta pemrograman dengan bahasa tingkat tinggi.
- Siklus hidup perangkat lunak, Waterfall model, V model, **Spiral model, Prototyping, prinsip dasar analisis kebutuhan, alur data, struktur data DSSD, Sistem Jackson, perangkat pembantu.**
- **Computer Aided Software Engineering (CASE) tools.**
- **Perancangan real time system.**
- **Analisis berorientasi objek, Pemodelan data, Metode formal, Pengantar perancangan dan implementasi**

Penilaian

- Tugas : 20%
- Absen : 5%
- UTS/open : 35%
- UAS/open : 40%

Total : 100%

PERANGKAT LUNAK DAN REKAYASA PERANGKAT LUNAK

Perbedaan Perangkat Lunak vs Rekayasa Perangkat Lunak

- **PERANGKAT LUNAK**

- Merupakan produk yang dibuat oleh para profesional perangkat lunak

- **REKAYASA PERANGKAT LUNAK**

- Merupakan suatu proses suatu kumpulan metode-metode dan sederatan perkakas yang memungkinkan para profesional mengembangkan perangkat lunak komputer

DEFINISI PERANGKAT LUNAK

1. Instruksi-instruksi (program komputer) yang ketika dijalankan menyediakan fitur-fitur, fungsi-fungsi dan kinerja-kinerja yang dikehendali
2. Struktur data yang memungkinkan program-program memanipulasi informasi
3. Informasi deskriptif pada hasil cetakan dan bentuk-bentuk maya yang menggambarkan pengoperasian dan penggunaan program-program

DEFINISI

PERANGKAT LUNAK

1. Pembuatan dan penggunaan prinsip-prinsip penting rekayasa agar pengguna dapat memperoleh perangkat lunak secara murah yang dapat diandalkan dan bekerja secara efisien pada mesin-mesin yang sesungguhnya

KARAKTERISTIK PERANGKAT LUNAK

1. Dapat dikembangkan, bukan diproduksi dalam konteks manufaktur
 - Kualitas tinggi dicapai melalui perancangan yang bagus, namun fase produksi pada suatu produk perangkat keras dapat menunjukkan masalah kualitas yang lebih kelihatan
2. Tidak mengalami kelelahan
 - Perangkat lunak tidak mengenal suku cadang, setiap terjadi kegagalan pada perangkat lunak akan diidentifikasi dan diterjemahkan kedalam kode-kode yang dapat dijalankan pada aras mesin
3. Dibuat berdasarkan spesifikasi yang diminta pengguna
 - Struktur data dan pemrosesan rinci dibutuhkan untuk membuat suatu antarmuka yang sesuai dengan keinginan pengguna

RANAH APLIKASI PERANGKAT LUNAK

1. Perangkat Lunak Sistem
 - Contoh : Sistem Operasi
2. Perangkat Lunak Aplikasi
 - Contoh : Pemrosesan → transaksi penjualan, produksi pada pabrik
3. Perangkat Lunak Rekayasa / Ilmiah
 - Contoh : Aplikasi algoritma untuk kebutuhan astronomi, vulkanologi, dll
4. Perangkat Lunak Yang tertanam
 - Contoh : Fungsi digital dalam mobil, sistem pengereman
5. Perangkat Lunak Lini Produk
 - Contoh : Pengolah kata, pengolah angka, pengolah data
6. Aplikasi WEB
 - Perangkat lunak yang berpusat pada jaringan komputer
7. Perangkat Kecerdasan Buatan
 - Contoh : Robotika, sistem pakar, dll

PERKEMBANGAN RPL

Tahun	Kejadian
1940an	Komputer pertama yang membolehkan pengguna menulis kode program langsung
1950an	Generasi awal interpreter dan bahasa macro Generasi pertama compiler
1960an	Generasi kedua compiler Komputer mainframe mulai dikomersialkan Pengembangan perangkat lunak pesanan Konsep Software Engineering mulai digunakan
1970an	Perangkat pengembang perangkat lunak Perangkat minicomputer komersial
1980an	Perangkat Komputer Personal (PC) komersial Peningkatan permintaan perangkat lunak
1990an	Pemrograman berorientasi obyek (OOP) Agile Process dan Extreme Programming Peningkatan drastis kapasitas memori Peningkatan penggunaan internet
2000an	Platform interpreter modern (Java, .Net, PHP, dll) Outsourcing

DEFINISI

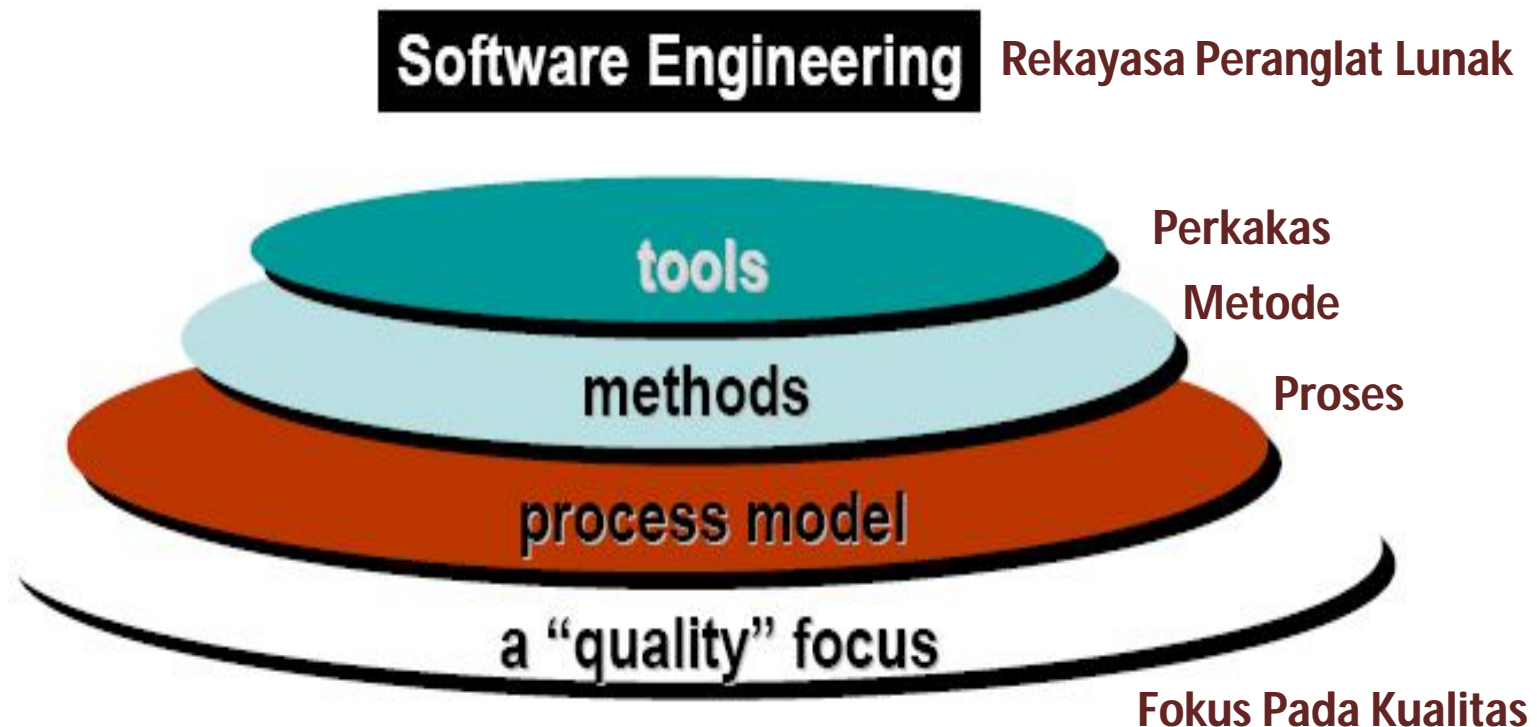
REKAYASA PERANGKAT LUNAK

1. Aplikasi dari suatu pendekatan yang sistematis, disiplin dan dapat diukur pada pengembangan, operasi dan perawatan perangkat lunak, yaitu: penerapan rekayasa pada perangkat lunak
2. Studi pendekatan-pendekatan seperti pada point (1)

LAPISAN

REKAYASA PERANGKAT LUNAK

- Rekayasa Perangkat Lunak (RPL) merupakan teknologi yang berlapis



REKAYASA PERANGKAT LUNAK

- Membutuhkan aktifitas-aktifitas kerangka kerja yang diterapkan secara berulang-ulang saat suatu proyek berjalan
- Aktifitas tersebut merupakan suatu proses rekayasa perangkat lunak
- Aktifitas tersebut : komunikasi → perencanaan → pemodelan → konstruksi → pengoperasian / penyerahan perangkat lunak
- Aktifitas-aktifitas kerangka kerja proses rekayasa perangkat lunak disempurnakan oleh sejumlah aktifitas-aktifitas yang bertindak

KEGIATAN-KEGIATAN PENYANGGA PADA UMUMNYA

1. Penelusuran dan kendali proyek perangkat lunak
 - Melakukan penilaian terhadap kemajuan proyek perangkat lunak yang sedang berjalan
2. Manajemen risiko
 - Menilai risiko-risiko yang mungkin berpengaruh pada hasil akhir proyek perangkat lunak
3. Penjaminan kualitas perangkat lunak
 - Memperjelas untuk memastikan kualitas perangkat lunak
4. Ulasan / tinjauan teknis
 - Menilai produk-produk hasil karya rekayasa

KEGIATAN-KEGIATAN PENYANGGA PADA UMUMNYA

5. Pengukuran

- Menjelaskan dan mengumpulkan tolok ukur dalam menyampaikan perangkat lunak yang memenuhi kebutuhan pengguna

6. Manajemen konfigurasi perangkat lunak

- Mengatur efek perubahan di seluruh perangkat lunak

7. Manajemen penggunaan ulang

- Memperjelas kriteria bagi penggunaan ulang

8. Persiapan produk kerja dan produksi

- Aktifitas-aktifitas yang diperlukan dalam

.

PRAKTIK

REKAYASA PERANGKAT LUNAK

1. Pahami permasalahannya (komunikasi dan analisa)
2. Rancang solusinya (pemodelan dan rancangan perangkat lunak)
3. Laksanakan rancangannya (pengaktifan kegiatan penulisan kode)
4. Periksa ketetapan hasilnya (pengujian dan penjaminan kualitas)

PRINSIP-PRINSIP UMUM REKAYASA PERANGKAT LUNAK

1. Alasan keberadaan perangkat lunak
2. Tetap sederhana
3. Pertahankan Visi
4. Apa yang anda buat, akan digunakan oleh konsumen/
pelanggan
5. Membuka diri terhadap masa depan
6. Rancanglah selangkah ke depan, sehingga dapat
digunakan kembali
7. Pikirlah!

MODEL-MODEL PROSES REKAYASA PERANGKAT LUNAK

MODEL PROSES GENERIK

- Proses rekayasa perangkat lunak merupakan sejumlah aktifitas-aktifitas kerja, tindakan-tindakan, serta pekerjaan-pekerjaan yang harus dilaksanakan saat produk kerja dibuat
- Masing-masing aktifitas berada dalam kerangka kerja / model yang mendefinisikan hubungan antara suatu proses dengan proses lainnya
- Masing-masing tindakan perangkat lunak didefinisikan menggunakan himpunan pekerjaan /kerangka kerja

KERANGKA KERJA PROSES PERANGKAT LUNAK

Proses Perangkat Lunak

Proses Kerangka Kerja

Aktifitas-Aktifitas Penyangga

Aktifitas Kerangka Kerja #1

Tindakan-tindakan Rekayasa Perangkat Lunak # 1.1.

Satuan pekerjaan

Pekerjaan, produk-produk kerja titik jaminan kualitas proyek

Tindakan-tindakan Rekayasa Perangkat Lunak # 1.t.

Satuan pekerjaan

Pekerjaan, produk-produk kerja titik jaminan kualitas proyek

Aktifitas Kerangka Kerja #n

Tindakan-tindakan Rekayasa Perangkat Lunak # n.1.

Satuan pekerjaan

Tindakan-tindakan Rekayasa Perangkat Lunak # n.s.

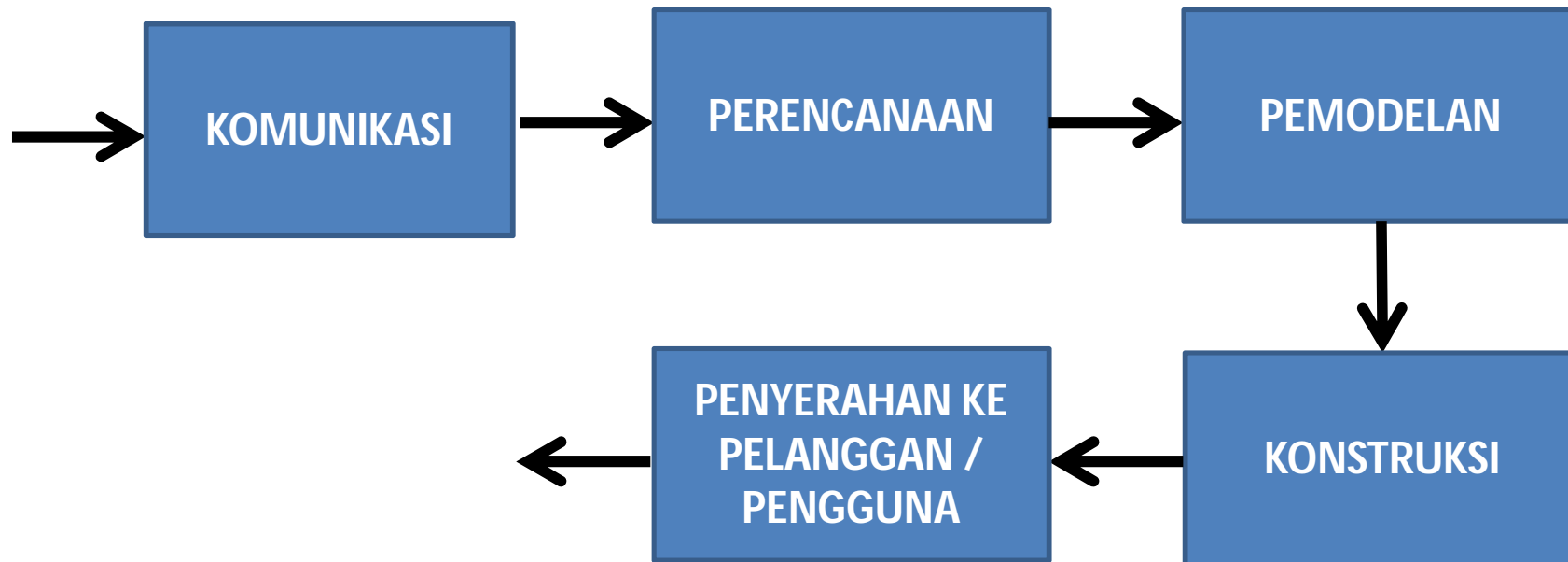
Satuan pekerjaan

Pekerjaan, produk-produk kerja titik jaminan kualitas proyek

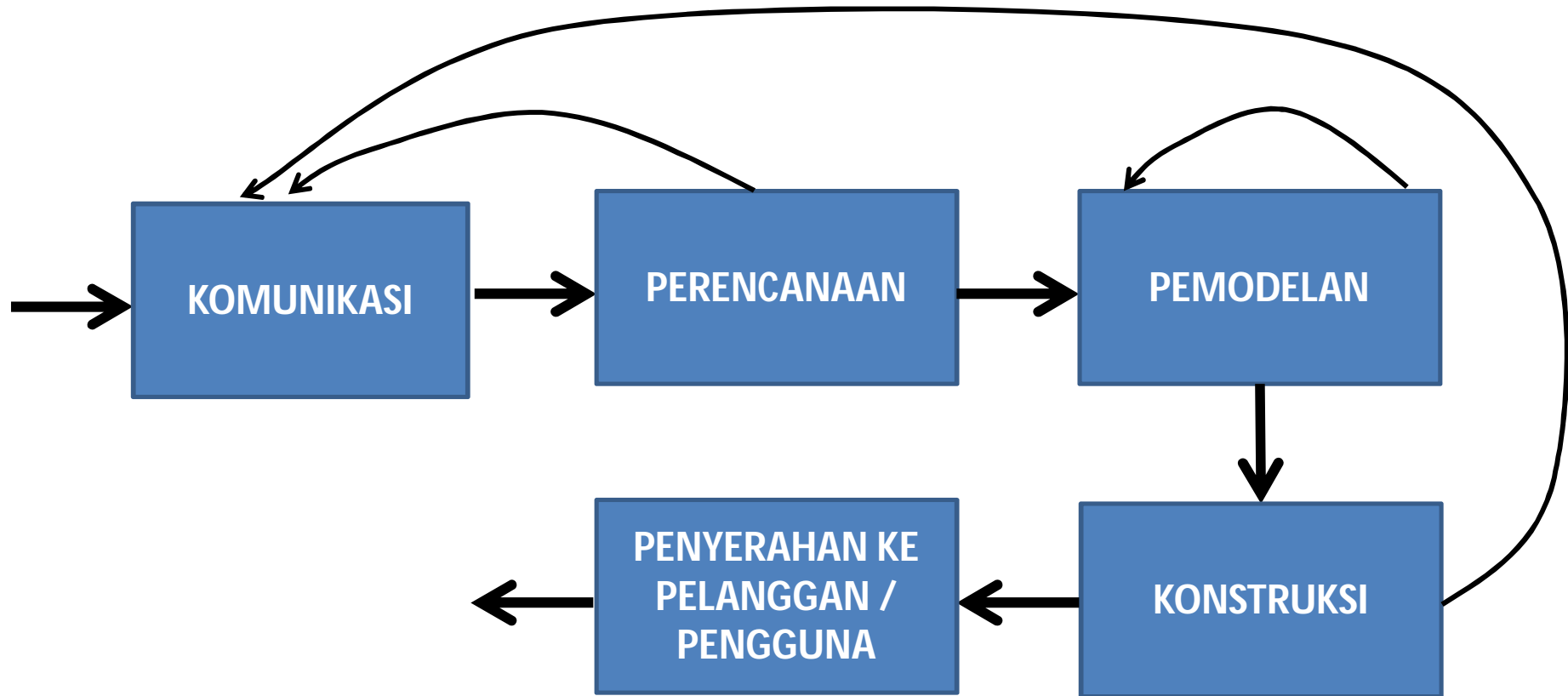
MACAM-MACAM ALIRAN PROSES REKAYASA PERANGKAT LUNAK

1. Aliran proses linier
2. Aliran proses iteratif
3. Aliran proses evolusioner
4. Aliran proses paralel

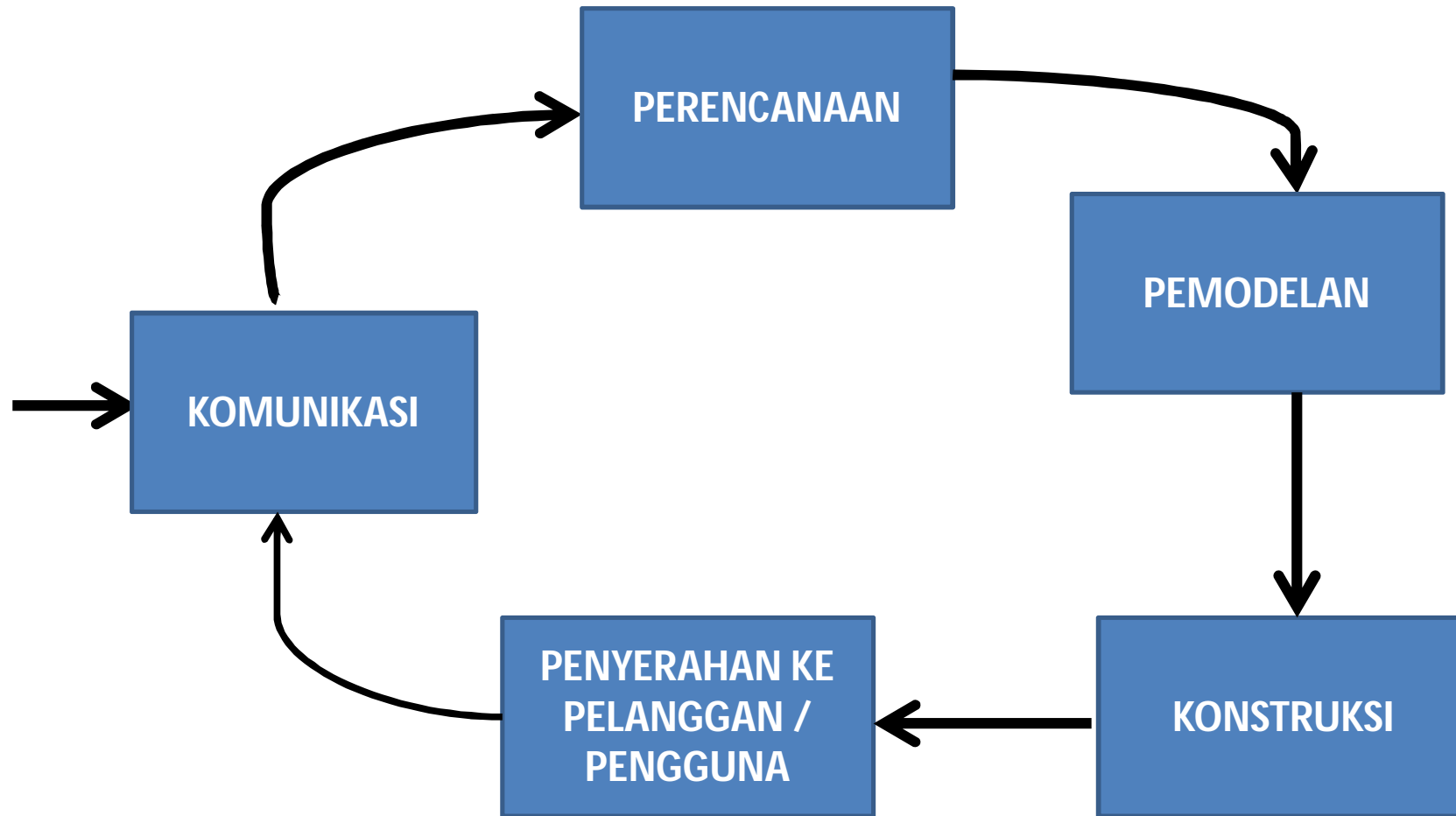
1. ALIRAN PROSES LINIER



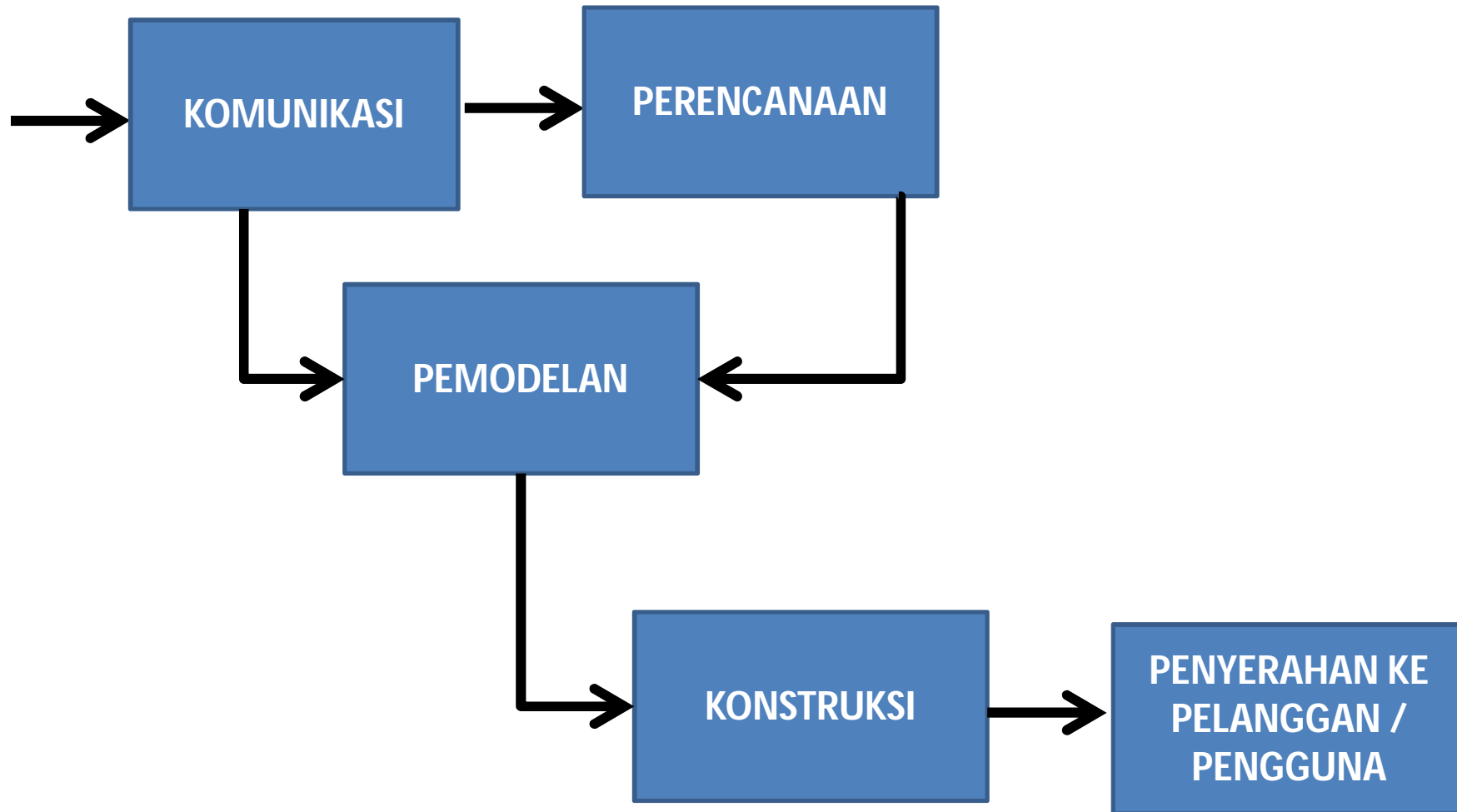
2. ALIRAN PROSES ITERATIF



3. ALIRAN PROSES EVOLUSIONER



4. ALIRAN PROSES PARALEL



TIPE POLA-POLA MENURUT AMBER PROSES REKAYASA PERANGKAT LUNAK (RPL)

1. Pola Tahapan

- Mendefinisikan suatu permasalahan yang berhubungan dengan aktifitas kerangka kerja tertentu untuk suatu proses

2. Pola Pekerjaan

- Mendefinisikan suatu permasalahan yang berhubungan dengan suatu aksi RPL atau tugas-tugas kerja dan relevan dengan RPL

3. Pola Fase

- Mendefinisikan urutan kerangka kerja yang terjadi di dalam suatu proses, meskipun saat aliran aktifitas secara keseluruhan secara alamiah bersifat iteratif

PENDEKATAN PENILAIAN PROSES RPL DAN PERBAIKAN USULAN, MENGGUNAKAN

1. SCAMPI (Standard CMMI Assessment Method for Process Improvement)
 - Menyediakan model penilaian dengan proses lima (5) langkah, yaitu : pemberian nilai-nilai awal (initiating) → melakukan diagnosa (diagnosing) → penetapan (establishing) → bertindak (acting) → belajar (learning)
2. CBA API (CMM-Based Appraisal for Internal Process Improvement)
 - Menyediakan teknik diagnosa untuk melakukan penilaian
3. SPICE (ISO/IEC15504)
 - Suatu standar yang mendefinisikan sejumlah spesifikasi kebutuhan untuk penilaian proses perangkat lunak
4. ISO 9001:2000 for Software
 - Suatu standar generik yang diterapkan untuk setiap organisasi

MODEL PROSES RPL

1. Model Air Terjun (Waterfall)
2. Model Proses Inkremental
3. Model Prototipe (Prototyping)
4. Model Proses Evolusioner
5. Model Spiral
6. Model RAD (Rapid Application Development)
7. Model-Model Konkuren
8. Model Metode Formal

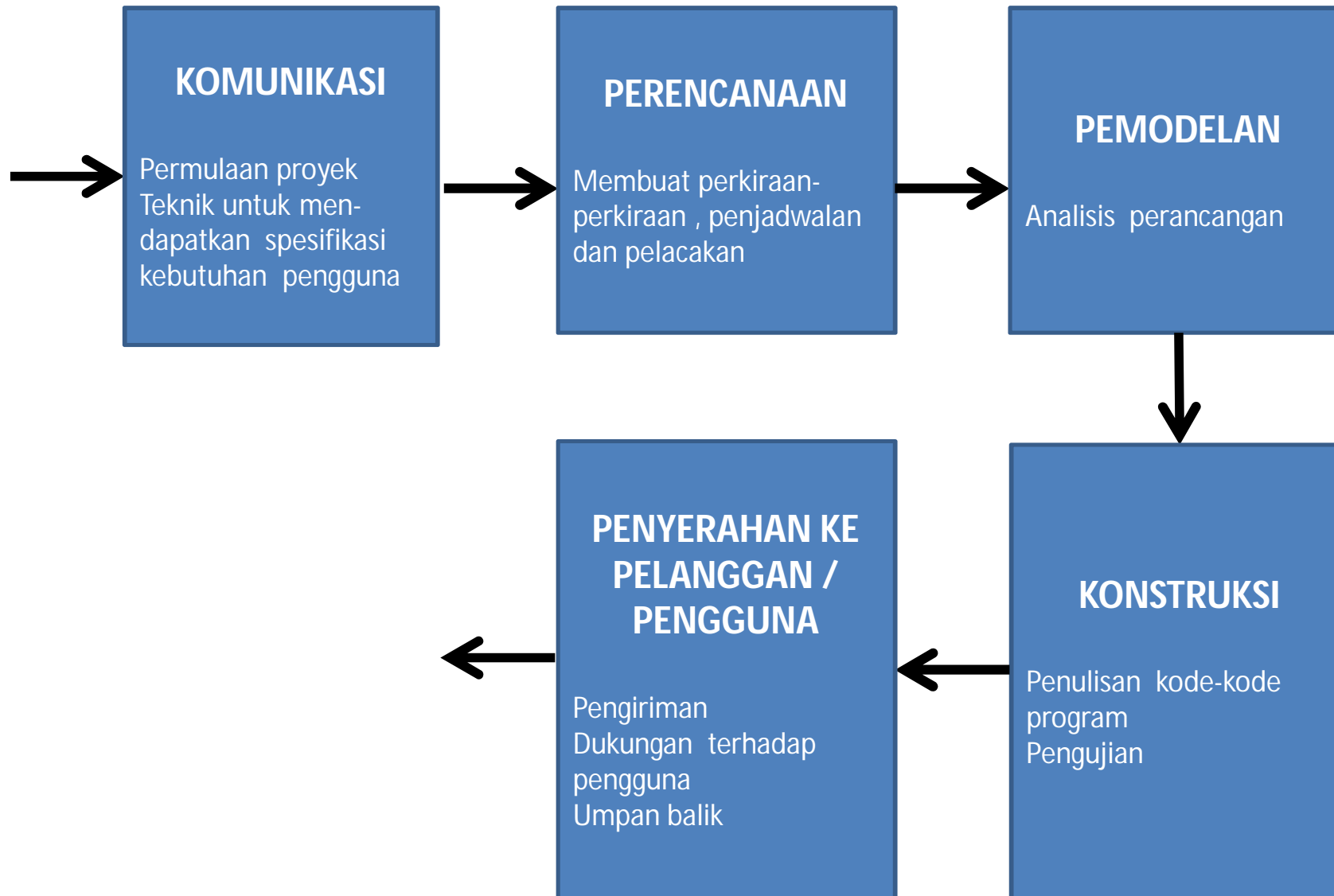
1. Metode Air Terjun / Waterfall

- Model Sekuensial Linier sering disebut Model Air Terjun merupakan paradigma rekayasa perangkat lunak yang paling tua dan paling banyak dipakai.
- Model ini mengusulkan sebuah pendekatan perkembangan perangkat lunak yang sistematis dan sekuensial.
- Variasi dari model waterfall/air terjun dinamakan sebagai **Model V (V-Model)**

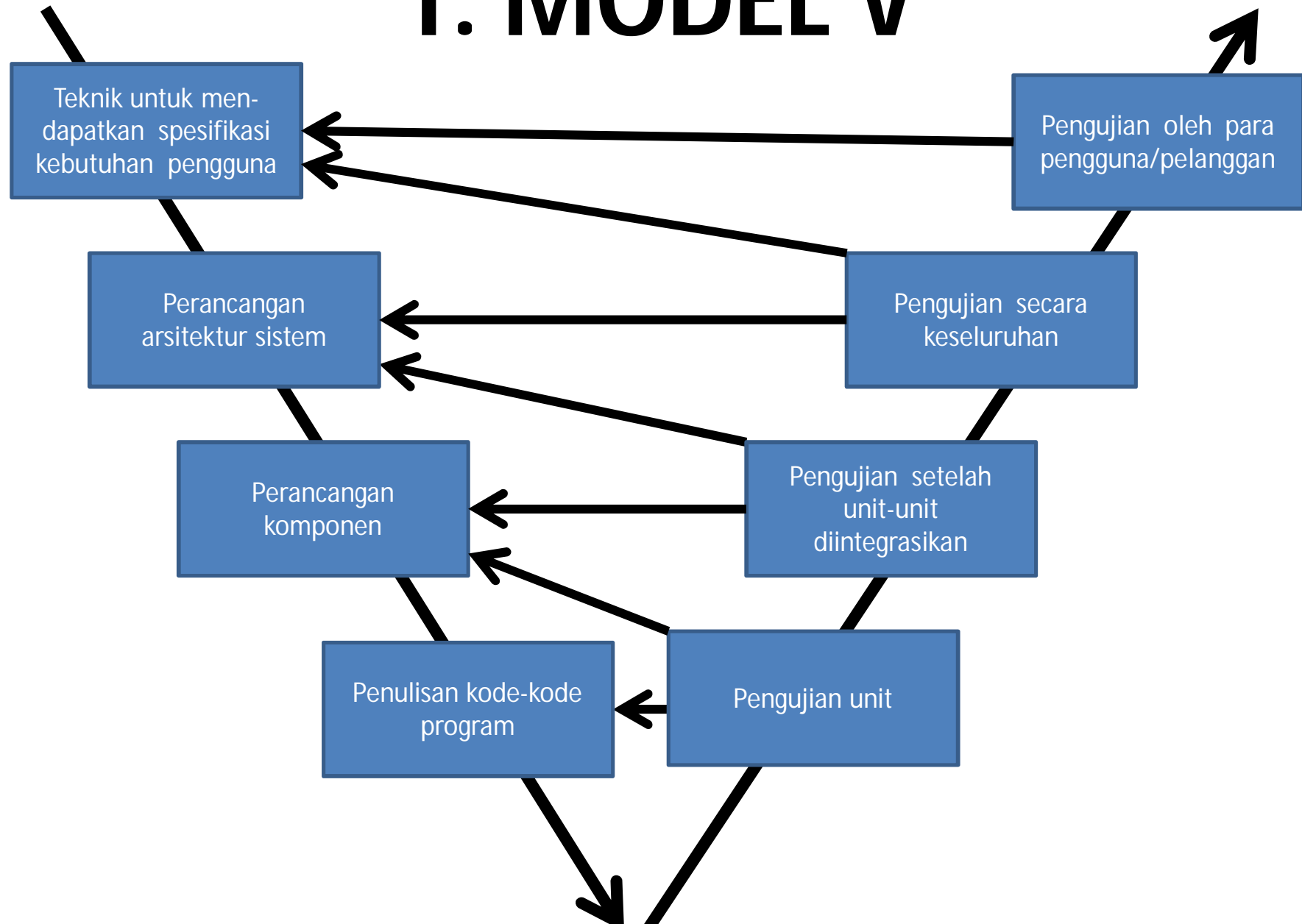
1. Metode Air Terjun / Waterfall

- Inti dari metode *waterfall* adalah pengerjaan dari suatu sistem dilakukan secara berurutan atau secara linear.
- Jadi jika langkah satu belum dikerjakan maka tidak akan bisa melakukan pengerjaan langkah 2, 3 dan seterusnya.
- Secara otomatis tahapan ke-3 akan bisa dilakukan jika tahap ke-1 dan ke-2 sudah dilakukan

1. MODEL WATERFALL



1. MODEL V



Perangkat Lunak Yang Dapat Digunakan Oleh Pelanggan/Pengguna

1. Kelebihan Waterfall

- Kelebihan :
 - Kualitas dari sistem yang dihasilkan akan baik.
 - Document pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya.
 - Mudah diaplikasikan.

1. Kekurangan Waterfall

- Kekurangan :
 - Diperlukan manajemen yang baik, karena proses pengembangan tidak dapat dilakukan secara berulang sebelum terjadinya suatu produk.
 - Kesalahan kecil akan menjadi masalah besar jika tidak diketahui sejak awal pengembangan.
 - Pengguna sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

Tugas 1

- Batas terakhir kirimkan ke email 2012 jam 24.00 wib ke dosen.it2011@gmail.com
- Easy Case
- Berdasarkan tugas Anda masing-masing buat dengan metode waterfall