

# MODEL PROSES PERANGKAT LUNAK

Oleh :

Rahmi Hidayati, S.Kom, M.Cs

# Perangkat Lunak Proses

- ▶ Serangkaian langkah pembuatan perangkat lunak berkualitas.
- ▶ Komponen-komponen produk perangkat lunak (program, dokumentasi dan data) adalah hasil dari perangkat lunak proses.
- ▶ Indikator berhasil tidaknya sebuah perangkat lunak proses adalah **KUALITAS**, **WAKTU Pengerjaan** dan **Kelayakan** produk.

# Software Engineering

*Software engineering* mencakup:

- **Perangkat Lunak Proses** : konteks implementasi metode, tatacara pengukuran kualitas dan akomodasi perubahan.
- **Metode Teknis** : *how-to* secara teknis, termasuk analisa kebutuhan, desain, konstruksi program, tes dan dukungan.
- **Pemanfaatan Tools Otomasi** : Dukungan semi-otomasi untuk proses dan metode.

CASE (*Computer Aided Software Engineering*) menggabungkan Perangkat Lunak, Perangkat Keras dan basisdata untuk membentuk lingkungan semacam CAD (*Computer Aided Design*) pada Perangkat Keras.

# CASE TOOLS

- ▶ CASE : *Computer Aided Software Engineering*
- ▶ Suatu peralatan baik *hardware* maupun *software* komputer yang digunakan untuk menyediakan pendukung otomatis dalam aktivitas pembangunan Perangkat Lunak (PL).
- ▶ Tujuan : meningkatkan produktivitas dalam proses pembangunan PL secara signifikan.

# *Kategori CASE*

CASE dikelompokkan dalam 2 kategori yaitu :

- ▶ ***Upper-CASE***

- ▶ Kakas untuk mendukung aktivitas-aktivitas proses awal analisis kebutuhan dan desain.

- ▶ ***Lower-CASE***

- ▶ Kakas untuk mendukung aktivitas-aktivitas selanjutnya seperti pemrograman, *debugging* dan pengujian.

# SDLC

## *(Software Development Life Cycle)*

### ▶ Spesifikasi

- ▶ Apa yang harus dilakukan oleh PL & batasan atau kendala pengembangannya

### ▶ Pengembangan

- ▶ Proses membuat sistem PL

### ▶ Validasi

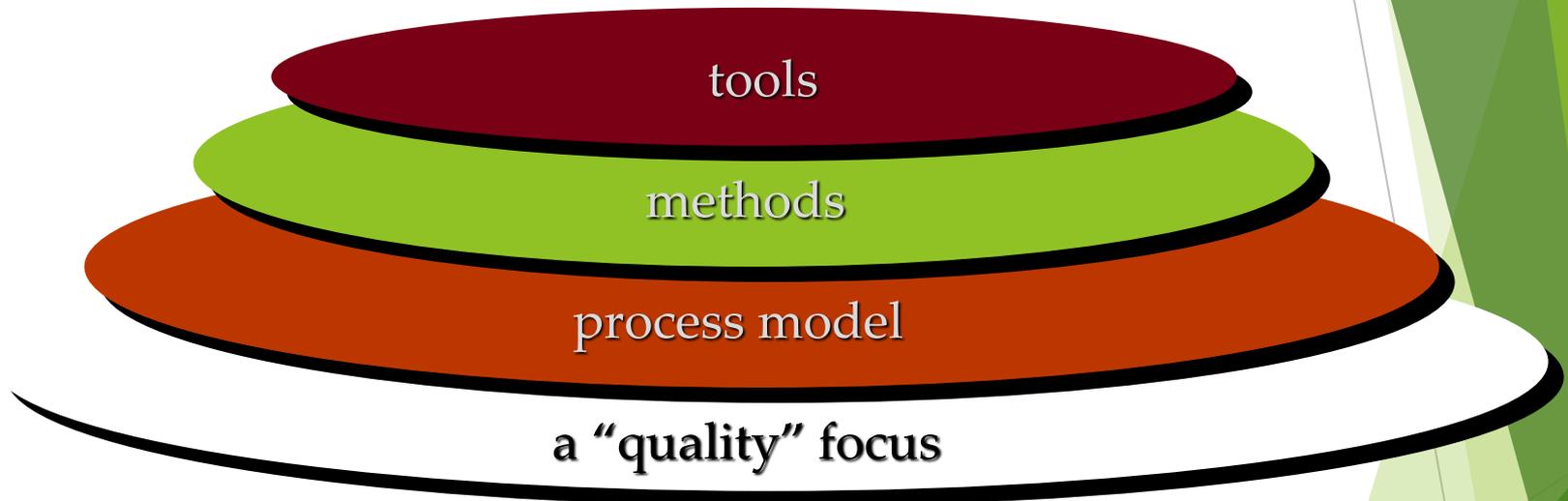
- ▶ Pengujian PL terhadap keinginan pengguna

### ▶ Evolusi

- ▶ Perubahan PL berdasarkan perubahan keinginan

# Teknologi Berlapis

Rekayasa Perangkat Lunak



# Pandangan Umum RPL

- ▶ Rekayasa : analisis, desain, konstruksi, verifikasi dan manajemen entitas teknis dan sosial.

Hal-hal penting untuk dijawab:

- ▶ APA masalah yang hendak diselesaikan?
- ▶ Apa saja KARAKTERISTIK kesatuan yang digunakan untuk menyelesaikan masalah?
- ▶ BAGAIMANA cara MEREALISASIKAN kesatuan dan solusinya?
- ▶ BAGAIMANA cara MEMBUAT kesatuan?
- ▶ PENDEKATAN apa yang digunakan untuk menemukan error selama perancangan dan pembuatan kesatuan?

# Tiga Fase Umum RPL

- ▶ Fase definisi, fokus pada pertanyaan “apa” (informasi, perencanaan proyek dan analisa kebutuhan).
- ▶ Fase pengembangan, fokus pada pertanyaan “bagaimana” (perancangan PL, membuat koding dan tes PL).
- ▶ Fase pemeliharaan, fokus pada “perubahan” :
  - ▶ Koreksi
  - ▶ Adaptasi
  - ▶ Peningkatan
  - ▶ Pencegahan

# Bingkai Kerja Proses

## Bingkai Kerja Proses

Aktivitas Bingkai Kerja

Tugas-tugas

Produk-produk

Point SQA (*Software Quality Assurance*)

Aktivitas Payung

# Aktivitas Payung

- ▶ Manajemen Proyek PL
- ▶ Review Teknik Formal
- ▶ Jaminan Mutu PL
- ▶ Manajemen Konfigurasi PL
- ▶ Persiapan dan Produksi Produk Pekerjaan
- ▶ Manajemen Penggunaan Kembali
- ▶ Pengukuran
- ▶ Manajemen Resiko

# PELAKU RPL

## ▶ **Manajer**

- ▶ Manajer Proyek, Manajer Konfigurasi, Manajer Penjamin kualitas PL

## ▶ ***Software Developer***

- ▶ Analis Sistem, Desainer, Programmer, Inspektor PL, Pengontrol Perubahan

## ▶ **Pendukung**

- ▶ Staf admin, Humas, Pencatat teknis, Admin database, Admin Jaringan.

# Model Proses

- ▶ Aktivitas-aktivitas bingkai kerja akan selalu diaplikasikan pada setiap *project*.
- ▶ Tugas-tugas pada setiap aktivitas akan bervariasi bergantung pada :
  - ▶ Tipe proyek
  - ▶ Karakteristik proyek
  - ▶ Penilaian umum, persetujuan tim proyek

# Pola-Pola Proses

- ▶ Pola-pola proses menentukan **sekelompok aktivitas, aksi, tugas-tugas pekerjaan, produk-produk pekerjaan** dan perilaku yang berkaitan.
- ▶ Contoh-contoh umum :
  - ▶ Komunikasi pelanggan (sebuah aktivitas proses)
  - ▶ Analisis (sebuah aksi)
  - ▶ Pengumpulan Kebutuhan (sebuah tugas proses)
  - ▶ *Review* sebuah produk kerja (sebuah tugas proses)
  - ▶ Model Desain (sebuah produk kerja)

## Proses Tim PL

- ▶ Setiap proyek diluncurkan menggunakan sebuah *script* yang mendefinisikan tugas-tugas yang harus diselesaikan.
- ▶ Tim diarahkan secara mandiri.
- ▶ Pengukuran dianjurkan.
- ▶ Pengukuran dianalisis dengan tujuan meningkatkan proses tim.

# Tujuan Utama Proses PL : *Kualitas Tinggi*

**Ingat:**

Kualitas Tinggi = Proyek pendek

**Mengapa?**

Sedikit/Tidak ada Pekerjaan ulang!

# Model Perangkat Lunak Proses

► Ragam pemodelan perangkat lunak proses :

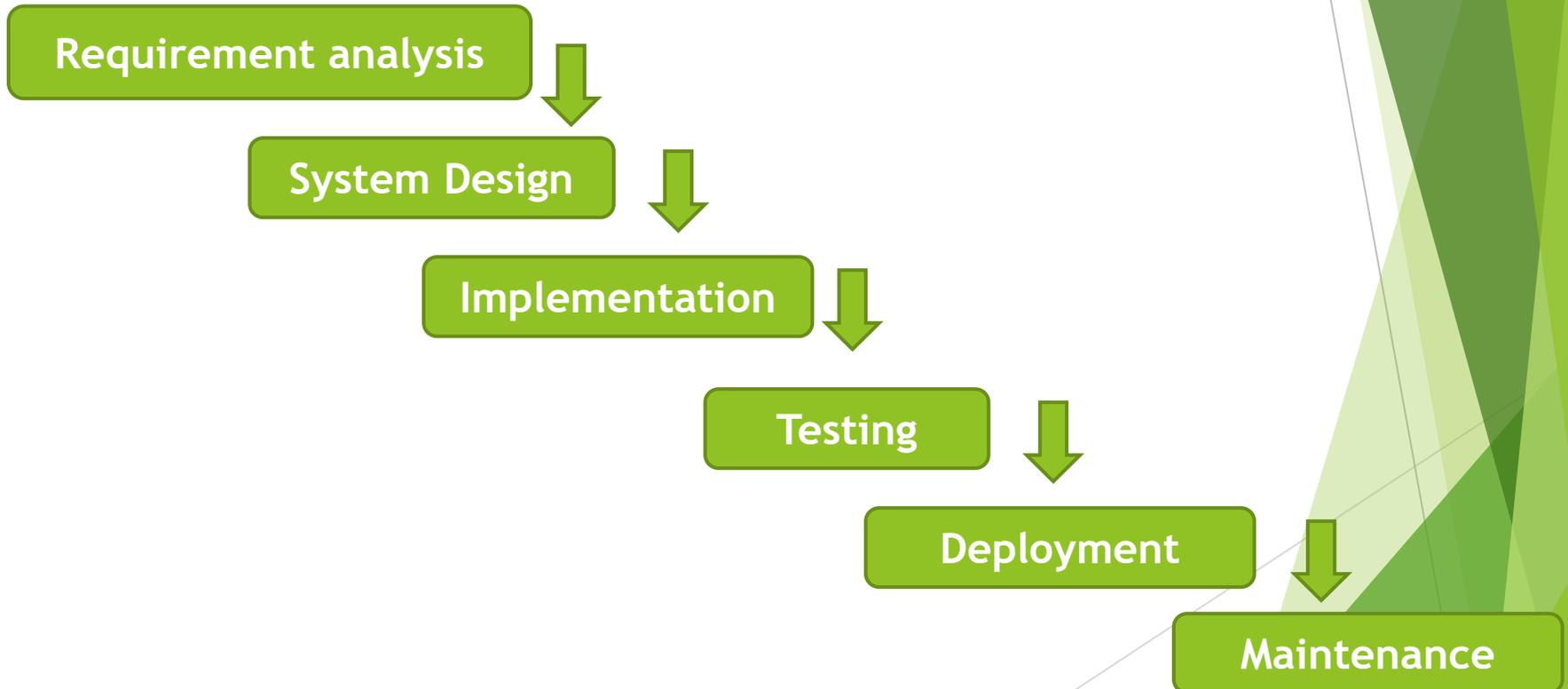
1. Model Sekuensial Linear (*waterfall*)
2. Model Prototype
3. Model RAD (*Rapid Application and Development*)
4. Model *Incremental*
5. Model Spiral

# Model Sekuensial Linier (*Waterfall*)

- ▶ Dianggap kuno, tetapi merupakan model yang paling banyak dipakai dalam *software engineering*.
- ▶ Model ini melakukan pendekatan secara sistematis danurut mulai dari level kebutuhan sistem menuju ke tahap analisis, desain, *coding*, *testing* dan *maintenance*.
- ▶ Disebut *waterfall* karena tahap demi tahap yang dilalui harus menunggu selesainya tahap sebelumnya dan berjalan berurutan.

# Model Sekuensial Linier (*Waterfall*)

- ▶ Berikut gambar model waterfall :



# Model Sekuensial Linier

## 1. *Requirement Analysis*

Seluruh kebutuhan *software* harus didapatkan dalam fase ini, kegunaan *software* yang diharapkan pengguna dan batasan *software*. Informasi dapat diperoleh melalui wawancara, survey atau diskusi. Informasi tersebut dianalisis untuk mendapatkan dokumentasi kebutuhan pengguna untuk digunakan pada tahap selanjutnya.

## 2. *System Design*

Tahap ini bertujuan untuk memberikan gambaran apa yang seharusnya dikerjakan dan bagaimana tampilannya. Tahap ini membantu dalam menspesifikasikan kebutuhan *hardware* dan sistem serta mendefinisikan arsitektur sistem secara keseluruhan.

# Model Sekuensial Linier

## 3. *Implementation*

Dalam tahap ini dilakukan pemrograman. Pembuatan *software* dipecah menjadi modul-modul kecil yang nantinya akan digabungkan dalam tahap berikutnya. Selain itu dalam tahap ini juga dilakukan pemeriksaan terhadap modul yang dibuat, apakah sudah memenuhi fungsi yang diinginkan atau belum.

## 4. *Integration & Testing*

Tahap ini dilakukan penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian untuk mengetahui apakah *software* yang dibuat telah sesuai dengan desainnya dan masih terdapat kesalahan atau tidak.

# Model Sekuensial Linier

## 5. *Operation & Maintenance*

*Software* yang sudah jadi setelah dijalankan memerlukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

# Kelebihan Model Sekuensial Linier

- ▶ Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
- ▶ Dokumen pengembangan sistem terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.
- ▶ Metode ini masih lebih baik digunakan walaupun sudah tergolong kuno, daripada menggunakan pendekatan asal-asalan.

# Kekurangan Model Sekuensial Linier

- ▶ 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 yang berakibat pada tahapan selanjutnya.
- ▶ Pelanggan harus sabar, karena pembuatan perangkat lunak akan dimulai ketika tahap desain sudah selesai. Sedangkan pada tahap sebelum desain bisa memakan waktu yang lama.