

Pengembangan Perangkat Lunak



Pengembangan Perangkat Lunak

Perancangan Perangkat Lunak

Pengkodean Perangkat Lunak

Pengujian Perangkat Lunak



Sumber Perangkat Lunak

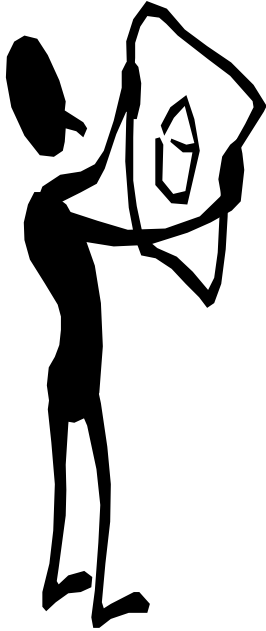
- ↪ Perangkat Lunak Komersial (Vendor)
- ↪ Perangkat Lunak In House /Customized Software

Perangkat Lunak Komersial (Vendor)

- Kelebihan** : Implementasi Cepat, Penghematan Biaya, Estimasi Biaya dan waktu, Reliabilitas.
- Kekutranan** : Kesesuaian Rancangan Sistem yang Tidak baik, Ketergantungan pada Vendor, Biaya Tidak Langsung dari Kerusakan SDLC
- Penilaian** : Kinerja Pengoperasian, Dokumentasi, Kemudahan pembelajaran, Kemudahan penggunaan (pengendalian dan penanganan kesalahan, dukungan)

Penyeleksian :

Manfaat Terbesar & Biaya Termurah



Cara :

- Tentukan angka penilaian Total dan Biaya Total
- Tentukan bobot penilaian relatif untuk setiap faktor kinerja umum (Nilai dasar 100)
- Beri penilaian untuk setiap faktor kinerja (proposal, benchmark, penilaian terpublikasi) - (skala : 1 = jelek s/d 10 = sangat bagus)
- Hitung Skor (Bobot x Penilaian)
- Hitung Angka penilaian total (Jumlah Skor)
- Hitung biaya perangkat penilaian (Harga : Total penilaian)

Soal :

Vendor A menawarkan Paket Aplikasi “A” dengan data sbb :
Penilaian Vendor (6), Kinerja pengoperasian (7), Dokumentasi (8),
Kemudahan belajar (7), Kemudahan Pemakaian (5), Kendali dan
penanganan Kesalahan (4), Dukungan (10).

Harga : \$ 22,700



Vendor B menawarkan Paket Aplikasi “B” dengan data sbb :
Penilaian Vendor (8), Kinerja pengoperasian (8),
Dokumentasi (9), Kemudahan belajar (6), Kemudahan
Pemakaian (6), Kendali dan penanganan Kesalahan (6),
Dukungan (8).

Harga : \$ 27,690

Penilaian Bobot yang digunakan adalah sbb :

Penilaian Vendor (10), Kinerja pengoperasian (20), Dokumentasi (10),
Kemudahan belajar (20), Kemudahan Pemakaian (10), Kendali dan
penanganan Kesalahan (20), Dukungan (10).

Paket Aplikasi mana yang anda PILIH ??????

Perangkat Lunak In House / Customized Software

Mengikuti SWDLS (Software Development Life Cycle)

Rancangan (Design)

Kode (Code)

Uji (Test)

Rancangan (Design)

Bagan terstruktur, SE/Pseudocode,
Tabel/pohon keputusan, Kamus data,
Diagram W/O, Diagram JSD dsb.

Kode (Code)

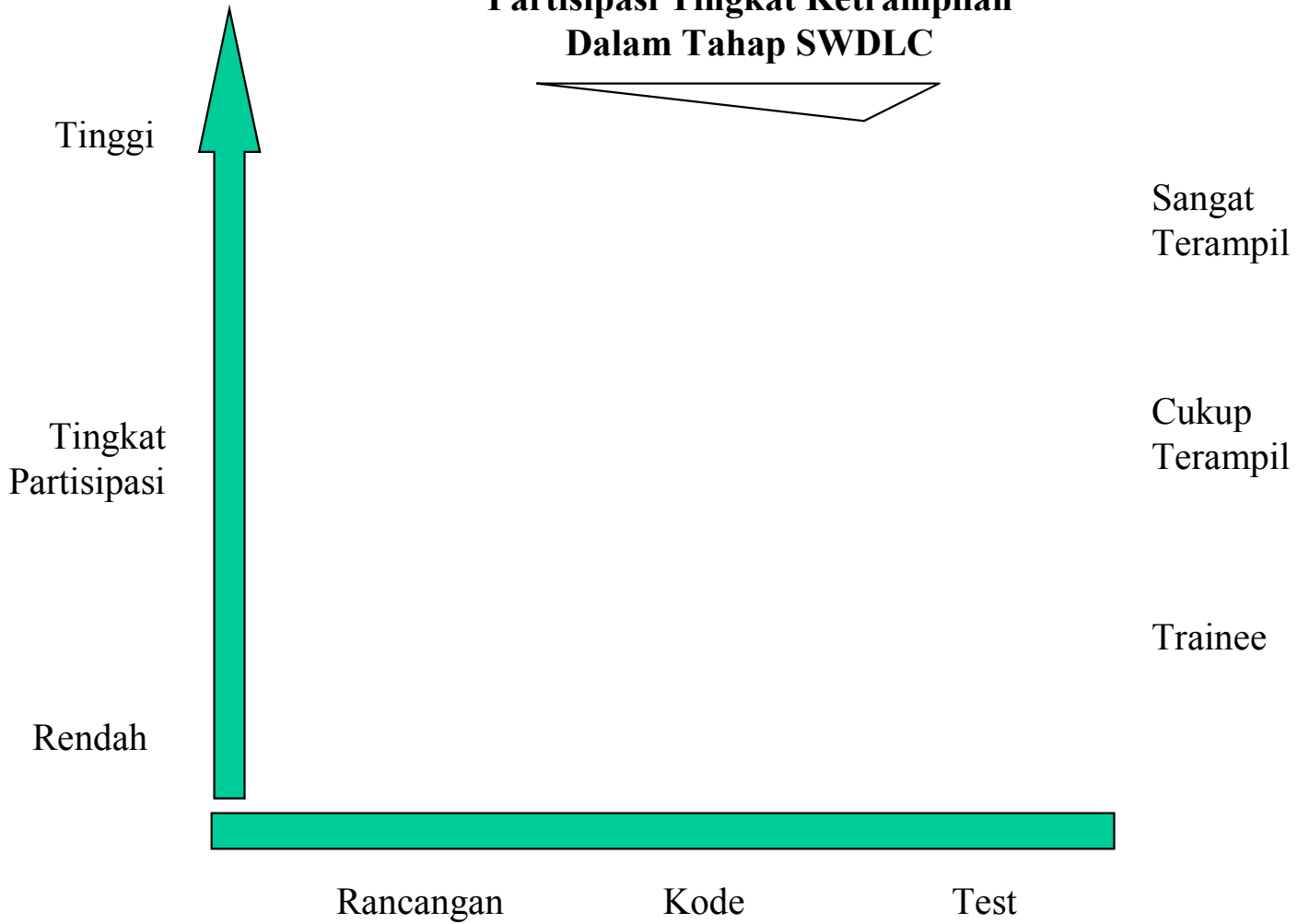
3GL, 4GL, Pemrograman Object

Uji (Test)

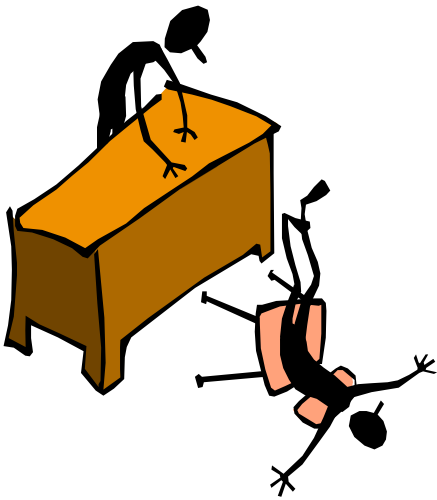
White Box, Black Box



Partisipasi Tingkat Ketrampilan Dalam Tahap SWDLC



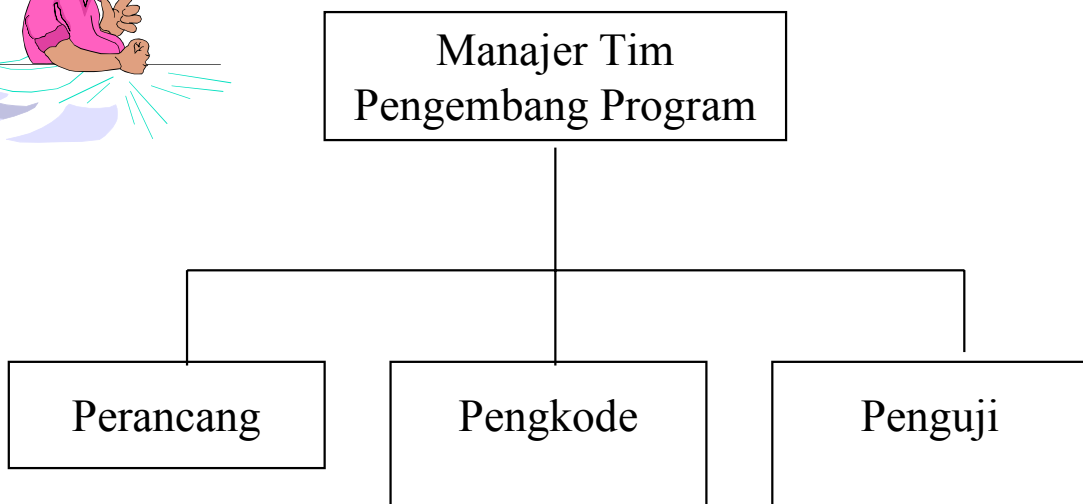
Pengorganisasian Proyek Pengembangan Perangkat Lunak (Komunikasi, Integrasi, koordinasi)



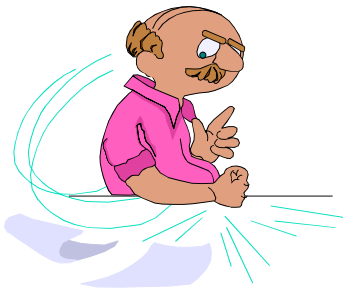
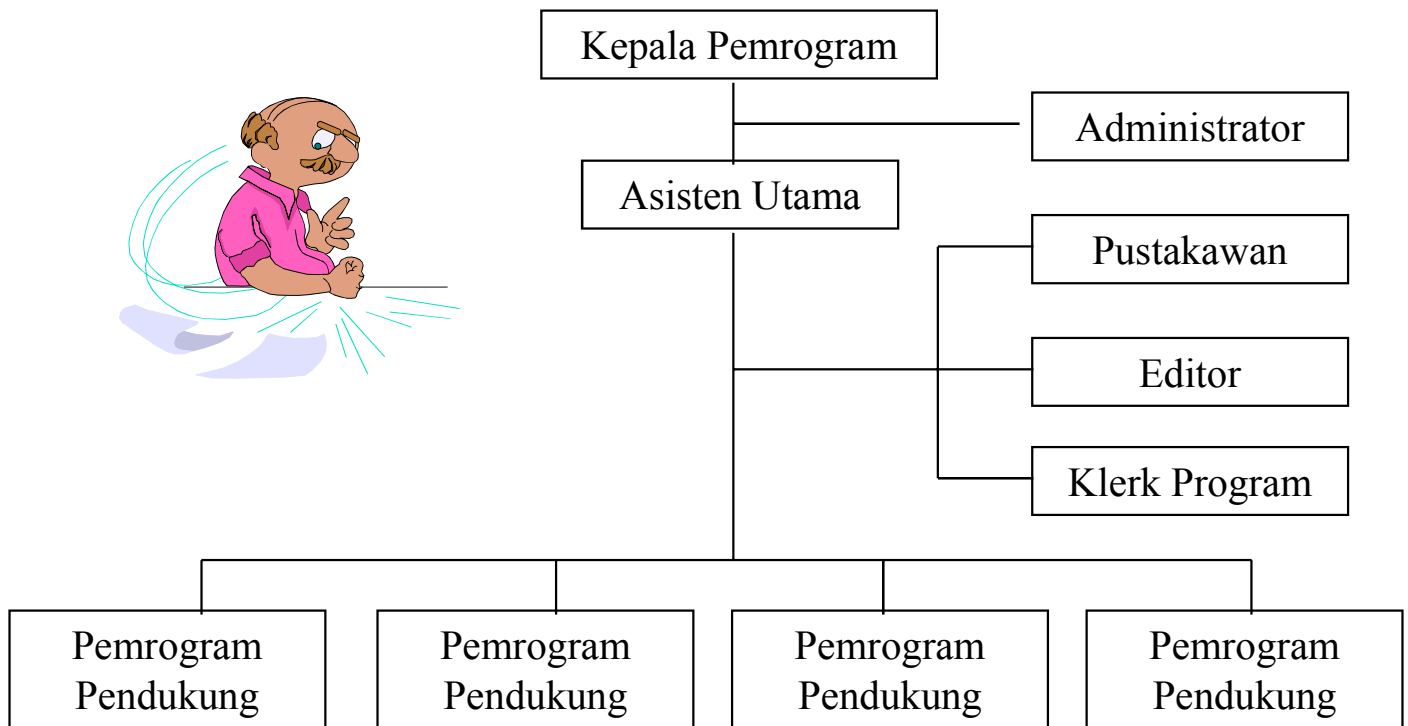
Pendekatan Organisasi

- ↪ Tim pengembangan program
(*program development team*)
- ↪ Tim programmer kepala (*chief programmer team*)
- ↪ Tim pemrograman bersama
(*egoless programming team*)

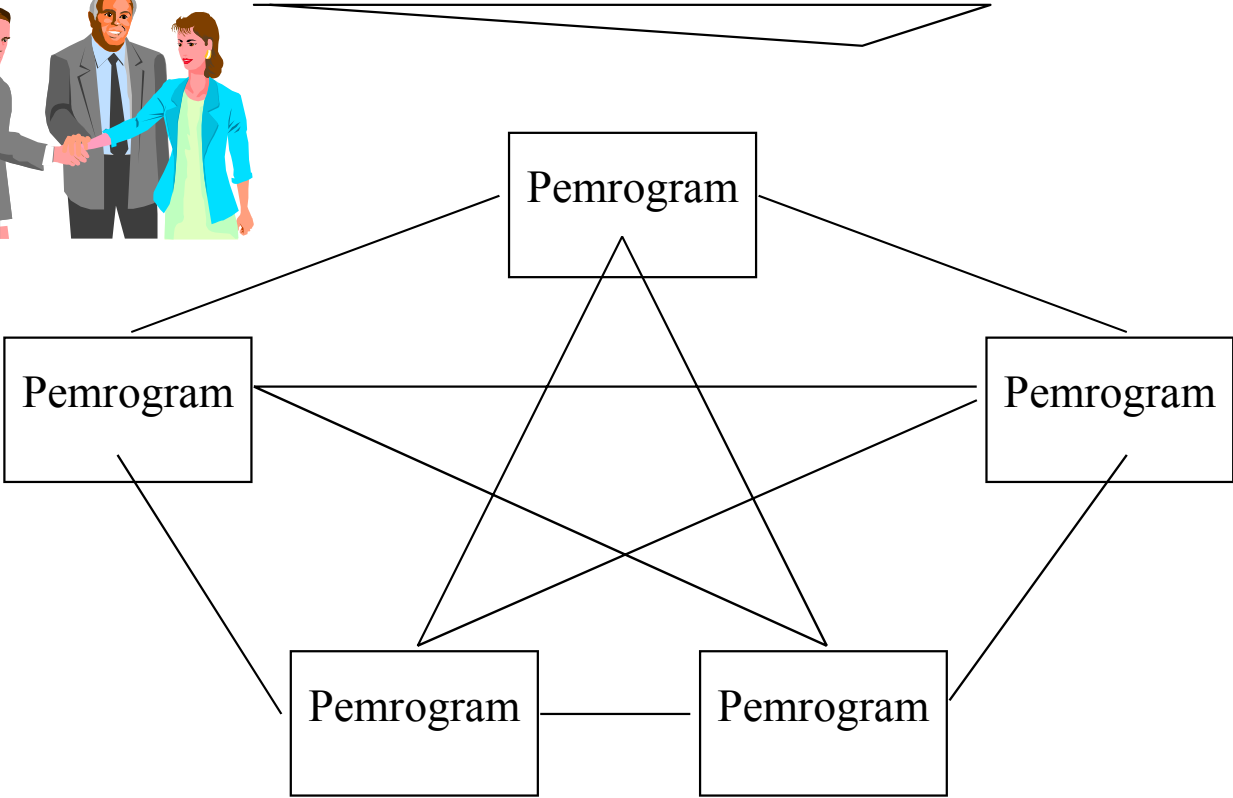
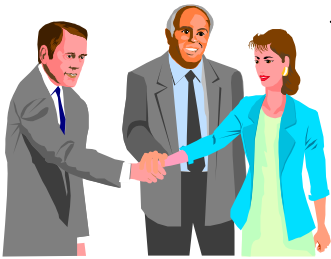
Program Development Team



Chief Programmer Team



Egoless Programming Team



Produktifitas

Pengembangan Perangkat Lunak

$$\text{Produktivitas} = \frac{\text{Output Yang Dihasilkan}}{\text{Input Yang Dikonsumsi}}$$

Metrik Lines Of Executable Code

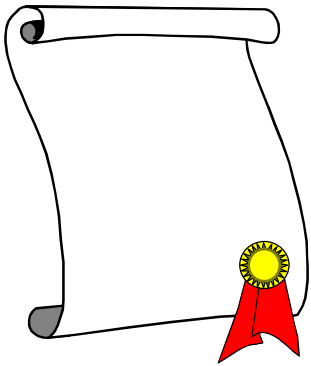
Mudah ditetapkan dan dibahas secara jelas, diakui secara luas, mudah diukur, mudah digunakan untuk estimasi.

Metrik Point Function

Jumlah input, output, query, file logik, interface

$$\text{Produktivitas} = \frac{\text{Jumlah titik fungsi yang dihadirkan}}{\text{Jumlah person-month}}$$

Kualitas Perangkat Lunak



Dimensi Pemakai :

- ↪ Kinerja Pengoperasian Keseluruhan
- ↪ Kemudahan Pembelajaran
- ↪ Pengontrolan dan Penanganan Kesalahan
- ↪ Dukungan dari Pembuat dan Pemelihara

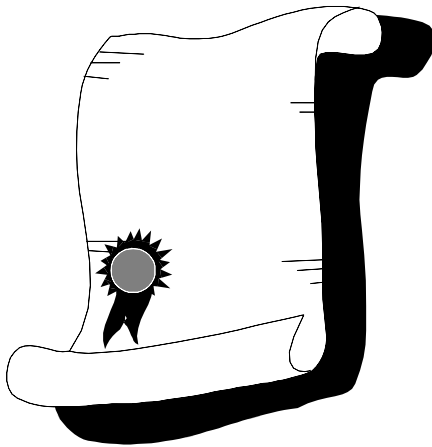
Dimensi Rancangan

(MURRE) :

- ↪ Maintainability
- ↪ Usability
- ↪ Reusability
- ↪ Realibility
- ↪ Extendability

Dimensi Nilai Tambah (PDM)

- ↪ Productivity
- ↪ Diferensiasi
- ↪ Management



**Jaminan Kualitas
(quality Assurance)**
adalah Proses merancang
kualitas ke dalam sistem

Pengendalian Kualitas (quality Control)

Proses yang memastikan bahwa kualitas sistem telah
terpenuhi/tercapai.

Perancangan Perangkat Lunak

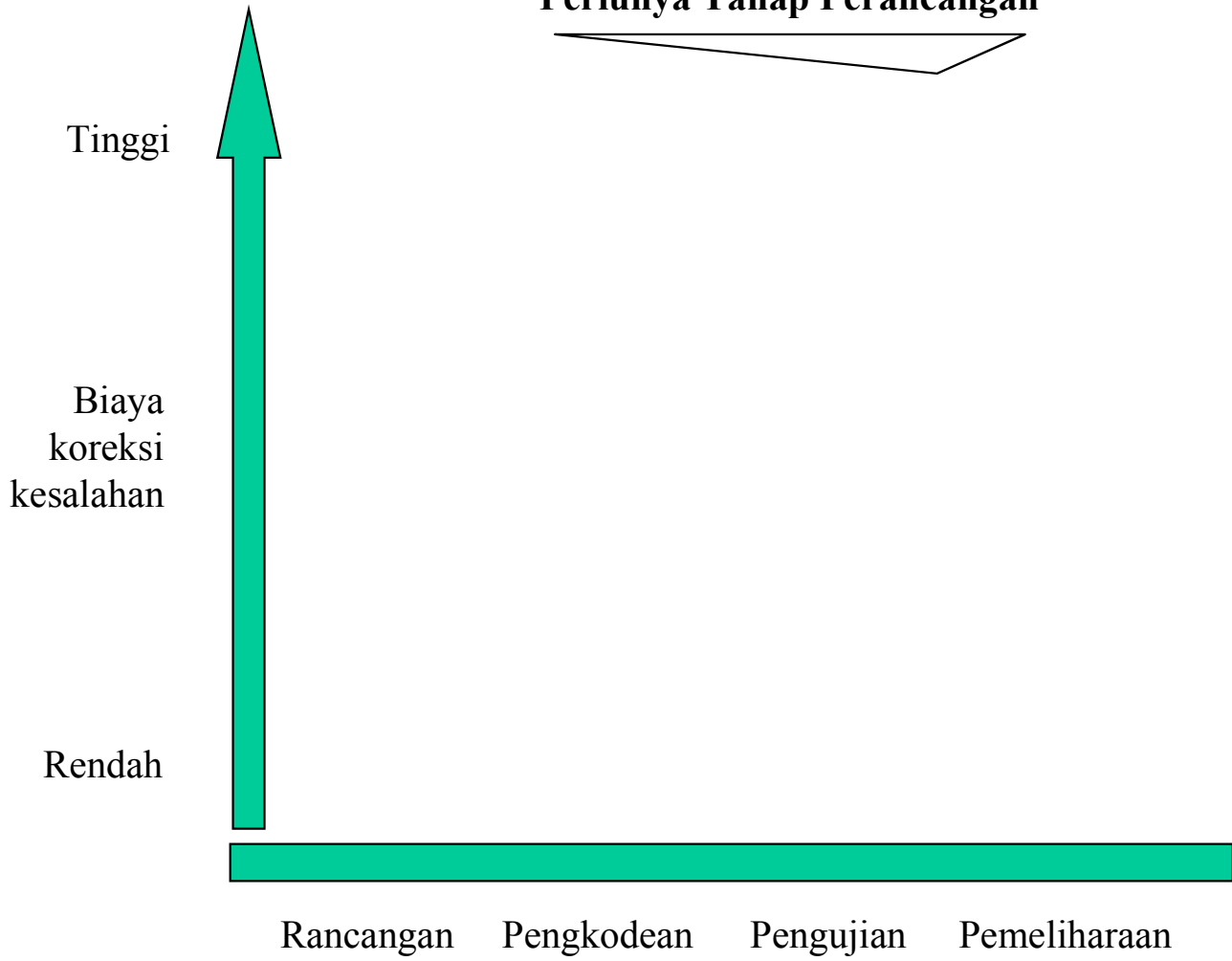


Perancangan
Terstruktur



Perancangan
Berorientasi Obyek

Perlunya Tahap Perancangan

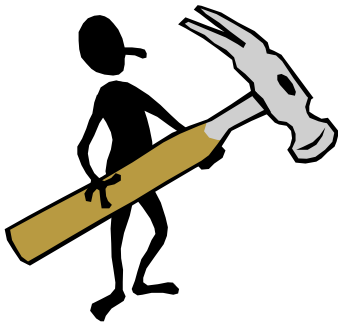


Perancangan Terstruktur

Orientasi pada Proses dan Data

Alat yang digunakan :

DFD, ERD, STD, Bagan Terstruktur,
SE/Pseodocode, W/O



Karakteristik Rancangan :

Modul disusun secara Hirarkis

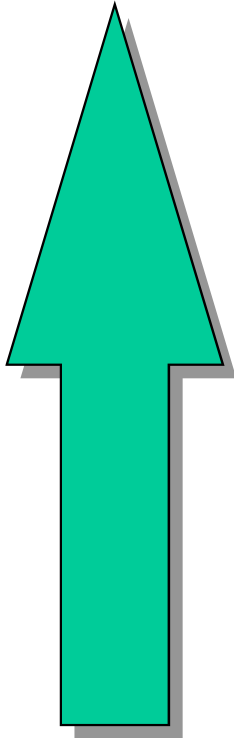
Menggunakan logika Call /Perform-based

Menggunakan Alur Kendali (top to bottom / bottom to top)

Repetisi dalam satu modul

Konsep kendali standart (urut, seleksi, repetisi)

Loose



Tight

Coupling

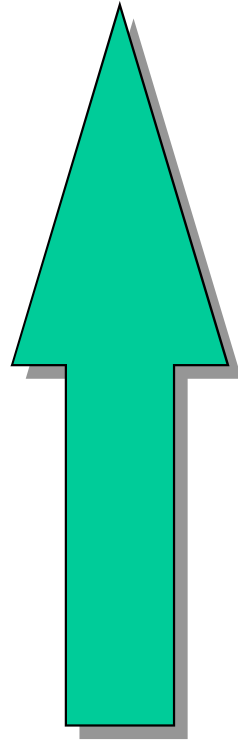
Data

Stamp

Control

Common

Content

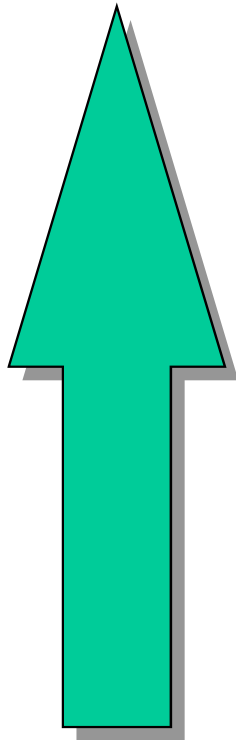


Best

Worst

Quality

Strongest



Weakest

Cohesion

Functional

Sequential

Communication

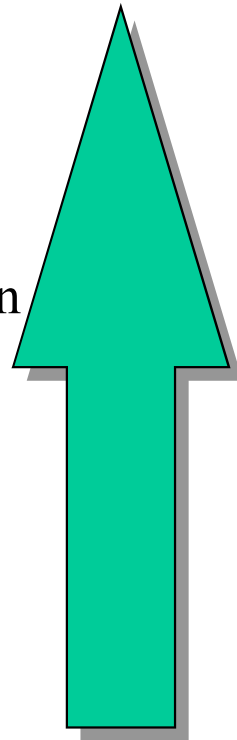
Procedural

Temporal

Logical

Coincidental

Best



Worst

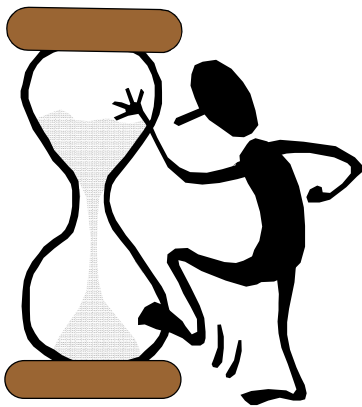
Quality

Perancangan Berorientasi Obyek

Orientasi pada Obyek

Alat yang digunakan :

DFD, ERD, STD, Bagan Terstruktur,
SE/Pseudocode



Tahapan :

- ↪ Mendeskripsikan Obyek, Kelas, Atribut dan Daftar Operasi
- ↪ Memodelkan relasi antara Obyek dan Kelas
- ↪ Memodelkan Pewarisan
- ↪ Menggabungkan faktor MURRE

Pengcodean Perangkat Lunak



Bahasa Pemrograman yang digunakan :



Ciri	3GL	4GL
Keringkasan	Tidak	Menyesatkan
Efisiensi Mesin	Ya	Tidak
Kompilasi	Ya	Sebagian
Fungsionalitas	Lengkap	Terbatas
Kompabilitas	Ya	Dapat jadi masalah
Portabilitas	Ya	Sebagian
Batch & Real-time	Ya	Sebagian
Bahasa Standar	Ya	Tidak
Kebebasan penjual	Ya	Tidak
Pengkodean end user	Umumnya Tidak	u/ beberapa App Ad hoc
Efisiensi pengkodean	Mungkin superior	Ya
Prototipe	Ya	Ya
Kendali terpasang	Ya	Sulit
Mendukung SQL	Sebagian	Sebagian
Mudah mendokumentasi	Ya	Tidak
Ketersediaan pemrogram	Ya	Sebagian
Kemudahan pemeliharaan	Ya (u/ terstruktur)	Umumnya Tidak

Jenis Bahasa OOP :

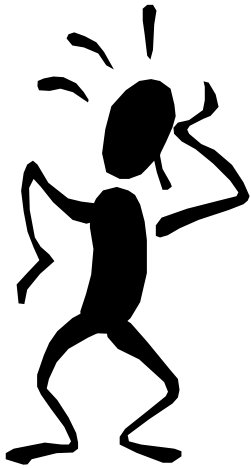
Murni :

Contoh : Smalltalk, Eiffel



Hybrid :

Contoh : C++, Pascal, Cobol



Saat Ini :

Orientasi pengembangan OOP pada sistem operasi berbasis GRAFIK.

Contoh : VBasic, Delphi, VFoxpro,



Pengujian Perangkat Lunak

Metode merancang test case :

White Box Testing

Black Box Testing

Kesalahan :

Fatal (Crash, Logika, Hang)

Serius (Output tidak Benar)

Minor

Area Uji :

Field

Record

File

Entry Data

Kendali

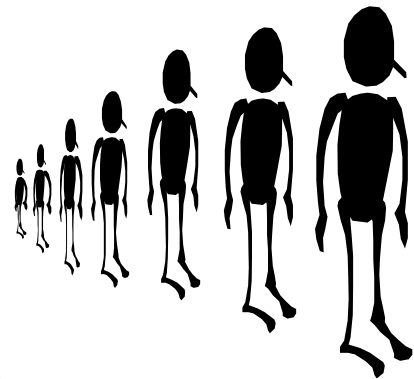
Arus Program

TAHAPAN PENGUJIAN :

Pengujian Modul



Pengujian Integrasi



Pengujian Sistem :

- ↪ Pengujian Pemulihan
- ↪ Pengujian Keamanan
- ↪ Pengujian Stress

Pengujian Penerimaan :

- ↪ Pengujian Alfa
- ↪ Pengujian Beta

Skenario Pengujian :

- ↪ Pengembangan test case dan strategi Pengujian
- ↪ Pelaksanaan test
- ↪ Pelaporan kesalahan yang terdeteksi
- ↪ Pengoreksian kesalahan
- ↪ Peramalan tingkat reliabilitas



Jenis Laporan Kesalahan :

- ↪ Saran
- ↪ Kesalahan rancangan
- ↪ Kesalahan pengcodean
- ↪ Kesalahan dokumentasi
- ↪ Query