



REKAYASA PERANGKAT LUNAK **REQUIREMENTS ANALYSIS FUNDAMENTALS**

I KADEK BAGUS FERISTHA
UDAYANA

Analisa Kebutuhan

- ▶ Adalah tugas rekayasa perangkat lunak yang menjembatani kesenjangan antara “system level software allocation” and “software design”
- ▶ Memungkinkan “system engineer” mengspesifikasikan fungsi dan unjuk kerja (performansi) perangkat lunak, menentukan antar muka perangkat lunak dengan elemen sistem lain, dan menetapkan kendala-kendala dalam proses pengdesainan perangkat lunak.
- ▶ Memungkinkan “Analyst” untuk menghaluskan alokasi perangkat lunak dan menggambarkan domain informasi yang akan disajikan dengan perangkat lunak.
- ▶ Memberikan gambaran kepada desainer tentang informasi dan fungsi yang dapat diterjemahkan ke dalam : data, arsitektur dan desain prosedur
- ▶ Memberikan kepada developer dan customer tentang perkiraan kualitas perangkat lunak yang akan dibuat / dikembangkan / dibangun.

TUGAS-TUGAS ANALISIS

Analisa keperluan / kebutuhan perangkat lunak dapat dibagi dalam 4 bidang usaha :

1. Pengenalan masalah :

- Analisis mempelajari spesifikasi sistem dan rencana proyek perangkat lunak.
- Penting untuk memahami perangkat lunak dalam konteks sistem dan meninjau ulang batasan perangkat lunak yang digunakan untuk generate perencanaan perkiraan.
- Komunikasi dalam proses analisa harus ditetapkan sehingga “masalah” dapat dikenali dengan pasti.

2. Evaluasi dan sintesa :

- Analisis harus melakukan evaluasi “aliran” dan “struktur” dari informasi, menghaluskan semua fungsi perangkat lunak, menetapkan karakteristik antarmuka, dan menemukan kendala-kendala dalam desain.
- Proses evaluasi dan sintesa berlangsung sampai analis dan customer merasa yakin bahwa perangkat lunak dapat dibuat spesifikasinya untuk tahap pengembangan.

3. Spesifikasi :

- Melakukan proses penentuan spesifikasi perangkat lunak, sehingga memudahkan pengerjaan pada tahap pengembangan.

4. Peninjauan ulang

- Proses meninjau kembali terhadap spesifikasi yang dihasilkan, sehingga diperoleh spesifikasi perangkat lunak yang lebih rinci dan jelas dengan maksud untuk menghasilkan perangkat lunak yang baik.

ANALIS

► **Analisis harus mempunyai kemampuan :**

1. Menganalisa konsep yang belum jelas
2. Menyerap fakta / informasi
3. Mengerti lingkungan pemakai
4. Menerapkan elemen sistem dari perangkat lunak maupun perangkat keras pada lingkungan pemakai
5. Berkomunikasi baik dalam bentuk tulisan maupun lisan.

PROBLEM AREAS

1. *Analisa keperluan merupakan aktivitas komunikasi yang dilakukan secara intensif*
2. *Permasalahan yang mungkin ditemukan pada proses analisa keperluan adalah :*
 - *Kesulitan untuk menggabungkan informasi yang didapat
 - *Penanganan permasalahan yang kompleks
 - *Perubahan-perubahan yang akan terjadi selama atau sesudah analisis.
3. *Hal-hal yang menyebabkan permasalahan pada tahap analisa kebutuhan :*
 - *Kurang komunikasi antara pemakai dan analis
 - *Teknik yang dipakai kurang baik, juga alat bantu yang digunakan tidak tepat.
 - *Kecendrungan mempersingkat waktu untuk melakukan analisa
 - *Gagal mempertimbangkan alternatif pemecahan masalah.

PRINSIP-PRINSIP ANALISA

1. DOMAIN INFORMASI

Domain informasi berisi 3 pandangan yang berbeda dari data yang diproses oleh program komputer :

a. Aliran informasi :

- ✓ Menggambarkan bagaimana perubahan data dari satu proses ke proses lain.

b. Isi informasi :

- ✓ Menggambarkan item-item data yang menyusun item yang lebih besar dan berisikan informasi yang lengkap
- ✓ Contoh :

Record mahasiswa terdiri dari item : nomor mahasiswa , nama mahasiswa , alamat mahasiswa , dan lain-lain

c. Struktur informasi :

- ✓ Menggambarkan organisasi data secara logika
- ✓ data-data disusun dalam bentuk tabel, hirarki, atau tree
- ✓ Yang diperhatikan pada struktur informasi adalah bagaimana data item yang ada saling berhubungan.

PRINSIP-PRINSIP ANALISA

2. PERMASALAHAN HARUS DIPARTISI

- ❑ Partisi dilakukan agar masalah yang besar bisa dimengerti dengan mudah
- ❑ Keuntungan :
 - ✓ Membantu meningkatkan pengertian permasalahan sampai detail / rinci
 - ✓ Memudahkan untuk penganalisaan
- ❑ Partisi horisontal dan partisi vertikal.

3. PANDANGAN LOGIKAL DAN FISIKAL

- ❑ Pandangan logikal :
 - ✓ Gambaran dari fungsi perangkat lunak yang diperlukan telah ditentukan dan informasi yang akan diproses tanpa memandang penerapannya sampai detail / rinci
- ❑ Pandangan fisikal :
 - ✓ Bagaimana penerapan fungsi pemrosesan dan struktur informasi

OBJECT-ORIENTED ANALYSIS

- ▶ Pendekatan object-oriented untuk pendefinisian masalah dan partisi cukup baik diterapkan sebagai bagian dari analisa kebutuhan
- ▶ Pendefinisian dari objek dan operasi adalah cara yang baik untuk memulai analisa terhadap fungsi dan domain informasi
- ▶ Objek : bisa dipandang sebagai suatu item informasi
- ▶ Operasi : sebagai suatu proses atau fungsi yang diterapkan pada satu / lebih objek
- ▶ Pendekatan analisa object-oriented

SOFTWARE PROTOTYPING

- ▶ Tujuan pembuatan “Software Prototyping” :
 - ✓ Membantu mengevaluasi apakah desain telah memenuhi spesifikasi fungsional maupun non fungsional
- ▶ Prototype diuji dan disempurnakan sebelum dilakukan produksi software yang sebenarnya.
- ▶ Dalam perekayasaan perangkat lunak, pembuatan prototype merupakan proses produksi
- ▶ Pembuatan prototype bisa membantu :
 - ✓ Pendefinisian dan spesifikasi desain yang akan dibuat
 - ✓ Pemilihan metode dan algoritma yang akan digunakan dalam desain
 - ✓ Pendefinisian spesifikasi interface dengan pemakai

SPECIFICATION

Prinsip-prinsip dalam penyusunan spesifikasi perangkat lunak :

1. Memisahkan fungsi dari implementasi
2. Spesifikasi sistem berorientasi kepada keperluan sistem
3. Spesifikasi harus memuat sistem dari perangkat lunak yang merupakan komponen
4. Spesifikasi harus termasuk di mana sistem akan dioperasikan
5. Spesifikasi sistem harus berupa model kognitif
6. Spesifikasi dapat dilaksanakan
7. Spesifikasi sistem harus bertoleransi terhadap ketidaklengkapan dan kemungkinan perluasan sistem
8. Spesifikasi harus dibatasi dan keterkaitannya longgar

REPRESENTATION

Digunakan untuk menjelaskan metoda yang dipakai untuk menganalisa kekuatan.

► **Petunjuk pembuatan representasi :**

1. **Format representasi dan isinya harus relevan dengan masalah yang akan dijelaskan**
2. **Informasi diisi dengan spesifikasi yang dapat bersarang**
3. **Menggunakan simbol / bentuk yang terbatas jumlahnya dan digunakan secara konsisten**
4. **Representari dapat direvisi**

SOFTWARE REQUIREMENTS SPECIFICATION

1. **Introduction**
 - a. System reference
 - b. Business objectives
 - c. S / W project constraints
2. **Information description**
 - a. Information flow representation
 - b. Information content representation
 - c. Information structure representation
 - d. System interface description
3. **Functional description**
 - a. Functional partitioning
 - b. Functional description
 1. Processing narrative
 2. Restrictions / limitations
 3. Performance requirements
 4. Design constraints
 5. Supporting diagrams
4. **Validation criteria**
 - a. Performance bounds
 - b. Classes of tests
 - c. Expected S / W response
 - d. Special considerations
5. **Bibliography**
6. **Appendi**

SEKIAN

"Terima Kasiiih.."

