**PENDAHULUAN**

**Atribut Produk**

Suatu produk mempunyai nilai sebagai hasil dari kegunaan atau kinerja yag dapat memenuhi kebutuhan user. Faktor yang memberi nilai tertinggi dalam suatu produk adalah versalitas, kemudahan dalam penggunaan, keamanan, estetis dan reliabilitas.

Terdapat sepuluh hal yang menjadi atribut suatu produk adalah :

1. Kinerja
2. Reliability
3. Pelayanan
4. Maintanability
5. Garansi
6. Mudah digunakan
7. Penampilan
8. Merek
9. Kemasan
10. Model terakhir

Alasan utama untuk reliabilitas dan maintabilitas dalam proses rekayasa adalah untuk memperbaiki reliabilitas dan maintabilitas dari suatu produk atau sistem yang dibangun, dan hal ini menambah nilai suatu produk.

Selama kegiatan awal dalam mendisain, terdapat beberapa cara untuk meningkatkan nilai produk. Untuk masalah yang kritis, komponen rata-rata kegagalan yang cukup tinggi adalah masalah redudansi atau duplikasi dari fungsi, melakukan *derating* atau menurunkan pengoperasian sistem dibawah rata-rata beban yang dapat ditampung, pemilihan teknologi, dan mengurangi kompleksitas sistem.

**Karakteristik Perangkat Lunak**

Perangkat lunak merupakan elemen logika dan bukan elemen sistem fisik, sehingga perangkat lunak memiliki ciri, sebagai berikut :

1. Perangkat lunak dibangun dan dikembangkan, tidak dibuat dalam bentuk yang klasik,

Walaupun perkembangan antara perangkat keras dan perangkat lunak sangat ekuivalen, namun aktivitas diantara keduanya sangat berbeda.

1. Perangkat lunak tidak pernah usang,

Perangkat keras sangat mudah mengalami kegagalan, hal ini dapat dilihat dari kurva ‘bathtube’ berikut :



Pada kurva tersebut ditunjukkan bahwa perangkat keras mengalami laju kegagalan yang sangat tinggi pada awal hidupnya. Kegagalan tersebut disebabkan oleh perancangan atau cacat pada saat pembuatan. Cacat tersebut harus dikoreksi, sehingga laju kegagalan turun ke keadaan yang diharapkan/sangat rendah/steady state untuk beberapa waktu. Tetapi seiring perjalanan waktu, laju kegagalan bertambah lagi karena pengaruh lingkungan.

Perangkat lunak tidak rentan terhadap pengaruh lingkungan. Kesalahan yang tidak dapat ditemukan pada saat perancangan akan menyebabkan kegagalan yang sangat tinggi pada awal kehidupan perangkat lunak, tetapi hal ini dapat diperbaiki. Hal ini ditunjukkan gambar berikut :



Selama masa hidupnya, perangkat lunak mengalami perubahan sebagai bagian dari maintanence. Selama perubahan dibuat, kesalahan akan muncul kembali, yang menyebabkan kurva kegagalan akan naik kembali, sehingga diperlukan modifikasi kembali. Hal ini dapat dilihat dari kurva bathtube actual berikut :



Secara umum kurva bathtube menggambarkan waktu penggunaan produk pada suatu periode tertentu, yaitu :



|  |  |  |  |
| --- | --- | --- | --- |
| **Periode** | **Simbolisasi** | **Penyebab** | **Solusi** |
| Pembuatan | DFR  (Decreasing Failure Rate) | Defect,  rendahnya control kualitas, | Quality control,  Pengujian penerimaan, |
| Pemakaian | CFR  (Constant Failure Rate) | Human error | Redudancy,  User friendly, |
| Kadaluarsa | IFR  (Increasing Failure Rate) | Peningkatan kebutuhan, prosedur kerja baru, | Teknologi,  Modifikasi |

1. Sebagian besar perangkat lunak dibuat secara custom built, serta tidak dapat dirakit dari komponen yang sudah ada

**REKAYASA SISTEM**

Rekayasa sistem adalah kegiatan untuk melakukan perekayasaan (manipulasi/membuat/mengembangkan/modifikasi) sistem, dengan berfokus pada berbagai elemen analisis, perancangan dan pengorganisasian elemen tersebut menjadi produk, jasa atau teknologi untuk mentransformasi informasi atau control.

Proses rekayasa sistem disebut *rekayasa informasi* bila konteks kerja rekayasa berfokus pada perusahaan bisnis. Pada tahapan pembuatan produk maka proses pembuatannya disebut *rekayasa produk.*

**SISTEM BERBASIS KOMPUTER**

**(CBIS – Computer Based on Information System)**

Sistem berbasis computer adalah serangkaian atau tatanan elemen-elemen yang diatur untuk mencapai tujuan yang telah ditentukan melalui pemrosesan informasi.

Tujuan tersebut dapat berupa upaya mendukung fungsi bisnis atau untuk mengembangkan suatu produk yang dapat dijual untuk menghasilkan keuntungan bisnis. Sehingga yang menjadi inti dari CBIS adalah “membenahi” system informasi yang dianalisis, setelah Sistem Informasi sudah “sempurna” maka baru dibuat perangkat lunak yang akan mendukung pelaksanaan kerja system informasi yang dianalisis.

Elemen dari sistem yang berbasis computer :

****

**REKAYASA INFORMASI**

Tujuan global rekayasa informasi adalah untuk mengaplikasikan teknologi informasi dengan cara tertentu yang melayani dengan baik kebutuhan bisnis secara keseluruhan.

Information Engineering (IE) dimulai dengan menganalisis sasaran dan tujuan bisnis, memahami area bisnis yang bekerja sama untuk mencapai sasaran dan tujuan bisnis, dan menentukan kebutuhan informasi dari masing-masing area bisnis dan bisnis keseluruhan.

Tujuan khusus dari rekayasa informasi (information engineering – IE) adalah untuk menentukan arsitektur yang memungkinkan suatu bisnis menggunakan informasi secara efektif, dan mengimplementasikan semua arsitektur tersebut.

IE membuat transisi ke dalam domain RPL yang teknis – proses dimana sistem informasi, aplikasi dan program dianalisis, didesain dan dibangun.

Arsitektur yang harus dianalisis dan dirancang :

1. Arsitektur data (database)

Arsitektur data memberikan kerangka kerja untuk kebutuhan informasi dari bisnis atau fungsi bisnis

1. Arsitektur aplikasi

Arsitektur aplikasi melingkupi elemen-elemen dari suatu sistem yang mentrasformasikan objek ke dalam arsitektur data untuk keperluan bisnis. (perangkat lunak atau penggabungan peran manusia dengan prosedur bisnis yang belum diotomatisasi)

1. Infrastruktur teknologi

Infrastruktur teknologi menyangkut penggunaan perangkat keras dan perangkat lunak untuk mendukung aplikasi dan data. (Berupa computer, jaringan, telekomunikasi, storage).

Pemodelan arsitektur sistem didapat dari hirarki aktivitas rekayasa transformasi. WV dalam hirarki dicapai melalui perencanaan strategi informasi ( Information Strategy Planning – ISP). ISP memandang bisnis keseluruhan sebagai suatu entitas dan memisahkan domain-domain bisnis. (Contoh domain bisnis : rekayasa manufaktur, pemasaran, keuangan). ISP mendefinisikan objek data yang terlihat pada tingkat perusahaan, hubungan antar data, dan bagaimana data tersebut mengalir di antara domain bisnis.

**PERENCANAAN STRATEGI INFORMASI**

Langkah pertama rekayasa Informasi adalah perencanaan strategi informasi (Information Strategic Planning - ISP). Sasaran ISP adalah (menciptakan model data tingkat bisnis yang menetapkan objek data kunci dengan hubungan antar data) :

1. Menentukan sasaran dan tujuan bisnis strategis
2. Mengisolasi factor sukses kritis (Critical Success Factor – CSF) yang memungkiskan bisnis mencapai tujuan dan sasaran tersebut
3. Menganalisis pengaruh teknologi dan otomasi terhadap tujuan dan sasaran
4. Menganalisis informasi yang ada untuk menentukan peranan data dalam mencapai sasaran dan tujuan.

Sasaran adalah pernyataan umum yang bersifat strategis, sedangkan tujuan adalah menentukan bentuk kegiatan kuantitatif yang bersifat taktis.

Factor sukses kritis (Critical Success Factor – CSF) dapat dihubungkan dengan suatu sasaran atau tujuan individual. CSF harus ada bila sasaran atau tujuan akan dicapai sehingga perencanaan manajemen harus mengakomodasikan CSF. Salah satu contoh CSF, untuk bidang manufaktur :

* strategi manajemen kualitas total bagi organisasi
* motivasi dan pelatihan pekerja
* mesin dengan reliabilitas lebih tinggi
* bagian-bagian kualitas tinggi
* rencana penjualan untuk menyakinkan pemasok agar menurunkan harga
* ketersediaan staf rekayasa

Setiap area bisnis banyak menggunakan teknologi informasi, maka ISP harus mengidentifikasi teknologi apa yang sekarang ada dan bagaimana teknologi itu digunakan untuk mencapai sasaran dan tujuan.

Analisis pengaruh teknologi akan menguji sasaran dan tujuan, dan memberikan indikasi mengenai teknologi-teknologi yang akan berpengaruh langsung dan tidak langsung terhadap upaya mencapai sasaran dan tujuan dengan sukses. Perekayasa informasi menekankan pertanyaan-pertanyaan :

* seberapa kritis teknologi untuk mencapai sasaran bisnis
* apakah teknologi itu dapat diperoleh saat ini
* bagaimana teknologi akan mengubah cara bisnis yang dilakukan
* berapa biaya langsung dan tidak langsung
* bagaimana bisnis harus menyesuaikan atau memperluas sasaran dan tujuan untuk mengakomodasi teknologi.

**PEMODELAN SISTEM**

Faktor-faktor yang harus dipertimbangkan perekayasa :

1. Asumsi, digunakan untuk mengurangi jumlah kemungkinan (permutasi) dan variasi yang mungkin.
2. Penyederhanaan, digunakan untuk menciptakan model dengan waktu yang tepat.
3. Pembatasan (*Boundaries*), digunakan untuk membatasi lingkup sistem.
4. Batasan (*Constraint*), digunakan untuk menunjukkan cara dimana model tersebut diciptakan dan pendekatan yang dilakukan pada saat model diimplementasikan.
5. Preferensi, digunakan untuk menunjukkan arsitektur yang dipilih untuk semua data, fungsi dan teknologi.

**REKAYASA PRODUK**

Rekayasa Produk merupakan suatu aktivitas pemecahan masalah. Data produk, fungsi dan tingkah laku yang dinginkan, ditemukan, dianalisis dan dialokasikan ke dalam komponen rekayasa individual

Produk yang dihasilkan adalah perangkat lunak. Rekayasa Perangkat lunak merupakan suatu aktivitas pemecahan masalah. Data produk, fungsi dan tingkah laku yang dinginkan, ditemukan, dianalisis dan dialokasikan ke dalam komponen rekayasa individual

Rekayasa produk :



Awal dari sebagian produk dan sistem yang baru, dimulai dari konsep yang sama dari fungsi yang diperlukan. Hal ini mengharuskan perekayasa sistem harus membatasi persyaratan produk dengan mengidentifikasikan ruang lingkup fungsi dan kinerja yang diinginkan.

Sekali fungsi, kinerja, batasan dan interface dibatasi, maka perekayasa sistem bergerak maju ke bagian Alokasi (pembuatan scenario sistem). Untuk memilih alokasi yang paling efektif, serangkaian criteria trade off harus diaplikasikan pada masing-masing alternative tersebut.

**REKAYASA PERANGKAT LUNAK**

Rekayasa perangkat lunak adalah penetapan dan penggunaan prinsip-prinsip rekayasa yang tangguh/teruji dalam upaya memperoleh perangkat lunak secara ekonomis, handal dan bekerja efisien di mesin nyata, dan berkaitan dengan metode dan kaidah yang diperlukan dalam mengembangkan perangkat lunak untuk computer.

Sedangkan pengertian rekayasa perangkat lunak menurut IEEE :*Rekayasa perangkat lunak adalah pendekatan sistematis untuk pengembangan, operasi, pemeliharaan dan pemberhentian pemakaian perangkat lunak.*

Prinsip-prinsip Rekayasa perangkat lunak :

1. Kekakuan (*Rigor*),

Rekayasa yang dilakukan harus sesuai dengan keinginan user, walupun terkadang diperlukan kreativitas perekayasa untuk membuat perangkat lunak.

1. Resmi (formal)

Pemilihan salah satu metodologi/pendekatan perangkat lunak, berdampak pada harus dilaksanakannya aktivitas rekayasa sesuai dengan metodologi yang dipilih, serta notasi yang dipilih harus selalu konsisten digunakan

1. Pemisahan kepentingan

Berkaitan dengan apek-aspek persoalan : melebarnya focus kerja, kompleksitas sistem.

1. Abstraksi

Menggambarkan keseluruhan sistem dalam bentuk yang sederhana

1. Modularitas

Mendekomposisikan persoalan menjadi modul-modul independent sehingga memisahkan perhatian mengenai persoalan internal modul dan interaksi modul-modul dengan lingkungan luarnya.

**LAPISAN REKAYASA PERANGKAT LUNAK**

Rekayasa perangkat lunak merupakan aktivitas yang berlapis-lapisan, yaitu :

1. Lapisan fondasi berupa Proses.

Lapisan fondasi adalah lapisan proses pengembangan perangkat lunak yang merupakan perekat bagi lapisan-lapisan teknologi yang lain, serta yang memungkinkan pengembangan perangkat lunak yang rasional dan tepat.

Lapisan proses mendefinisikan kerangka kerja untuk sekumpulan proses pokok (KPA –Key Process Area)

1. Lapisan Metode

Lapisan Metode memberikan cara teknis dalam membangun perangkat lunak pada kegiatan penetapan kebutuhan, analisis, perancangan, pembangunan program, pengujian.

1. Lapisan Alat Bantu

Lapisan yang mendefinisikan alat Bantu yang digunakan untuk mendukung pelaksanaan proses dan metode.

**PROSES PERANGKAT LUNAK**

Penggambaran aktivitas dalam proses perangkat lunak :

Kerangka Kerja Proses Umum

Aktivitas Pendukung

Aktivitas Kerangka Kerja

Rangkaian Tugas

Tugas

Kejadian Penting

SQA

Kerangka kerja proses umum dibangun dengan mendefinisikan sejumlah aktivitas kerangka kerja yang dapat diaplikasikan ke proyek perangkat lunak, dan aktivitas pendukung.

Kerangka kerja terdiri dari sejumlah tugas rekayasa perangkat lunak, milestone (panduan kerja) dan penyerahan perangkat lunak serta jaminan kualitas. Sedangkan pada aktivitas pendukung memperhatikan proses jaminan kualitas, model proses dan manajemen konfigurasi perangkat lunak.

**ALAT BANTU PERANGKAT LUNAK**

Secara umum untuk menggambarkan keadaan sistem dapat digunakan Model analisis dan desain :

1. Berorientasi aliran data (proses)

Alat bantu yang digunakan :

* 1. Konteks Diagram – Data Flow Diagram – Kamus Data
  2. Control Flow Diagram – State Transition Diagram

1. Berorientasi Struktur Data

Alat Bantu yang digunakan :

* 1. Diagram Warnier Orr (DSSD - Data Structured System Development)
  2. JSD (Jackson System Development)
  3. Diagram ER – Normalisasi – Relasi Tabel

1. Berorientasi Objek

Alat Bantu yang digunakan :

* 1. OOAD – Object Oriented Analysis and Design (Coad dan Yourdon)
  2. OMT – Object Modelling Technique (Rumbaugh)
  3. OOSE – Object Oriented Software Engineering (Ivar Jacobson)