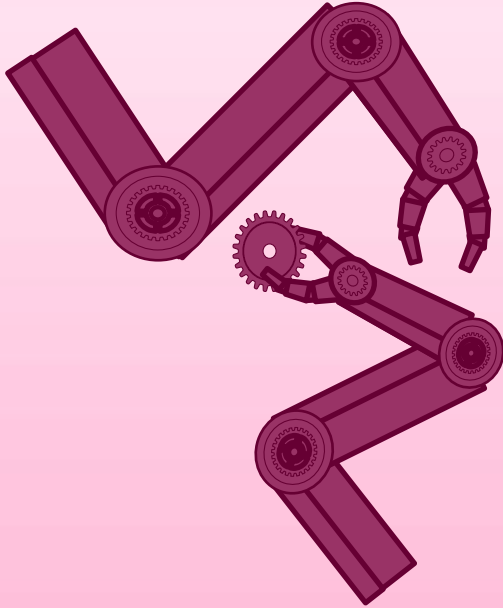
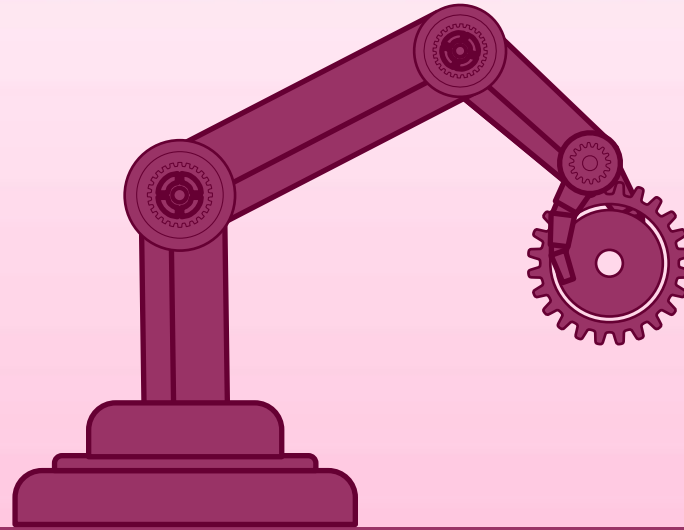




Pengembangan Sistem

Oleh:

Muhamad Muslih, ST., M.Kom



Mengapa Sistem Perlu Dikembangkan ?

Alasan mengapa sebuah sistem harus dikembangkan :

- ☐ Ketidakberesan dalam sistem lama
- ☐ Pertumbuhan organisasi : kebutuhan informasi yg semakin luas, volume pengolahan data meningkat, perubahan prinsip akuntansi
- ☐ Untuk meraih kesempatan – kesempatan dalam berbagai hal, seperti : peluang-peluang pasar, peningkatan pelayanan
- ☐ Adanya instruksi-instruksi (directive) dari pimpinan atau dari luar organisasi seperti peraturan.

Prinsip Pengembangan Sistem

- ❑ Sistem yg dikembangkan adalah untuk manajemen
- ❑ Sistem yg dikembangkan adalah investasi modal yg besar
- ❑ Melaksanakan tahapan kerja dan tugas-tugas yg harus dilakukan dalam proses pengembangan sistem
- ❑ Dokumentasi harus ada utk pedoman dalam pengembangan sistem

Tim Pengembangan Sistem

- ☐ Manajer Analis Sistem
- ☐ Ketua Analis Sistem
- ☐ Analis Sistem Senior
- ☐ Analis Sistem Junior
- ☐ Pemrogram Aplikasi Senior
- ☐ Pemrogram Aplikasi Junior

Tim Pengembangan Sistem

Manajer analisis system (manager of system analysis)

Disebut juga sebagai koordinator proyek dan mempunyai tugas dan tanggung jawab sebagai berikut:

- Sebagai ketua / koordinator team pengembangan sistem
- Mengarahkan, mengontrol dan mengatur anggota team pengembangan sistem yang lainnya.
- Membuat jadual pelaksanaan proyek pengembangan sistem yang akan dilakukan.
- Bertanggung jawab dalam mendefinisikan masalah, studi kelayakan, desain sistem dan penerapannya.
- Memberikan rekomendasi-rekomendasi perbaikan sistem.
- Mewakili team untuk berhubungan dengan pemakai sistem dalam hal perundingan-perundingan dan pemberian nasihat kepada manajemen dan pemakai system.
- Membuat laporan-laporan kemajuan proyek.
- Mengkaji ulang dan memeriksa kembali hasil kerja dari team.

Tim Pengembangan Sistem

- **Ketua analis system (lead system analyst)**
Biasanya menjabat sebagai wakil dari manajer analis sistem. Tugasnya adalah membantu tugas dari manajer analis sistem dan mewakilinya bila manajer analis sistem berhalangan.
- **Analist sistem senior (senior system analyst)**
merupakan analist sistem yang sudah berpengalaman
- **Analist system (System analyst)**
merupakan analist sistem yang cukup berpengalaman dan dapat bekerja sendiri tanpa bimbingan dari analist sistem senior.
- **Analist system junior (junior system analist)**
merupakan analist sistem yang belum berpengalaman dan masih membutuhkan bimbingan dari analist sistem yang lebih senior. Analist syistem junior ini juga sering disebut dengan analist syistem yang masih dilatih.

Tim Pengembangan Sistem

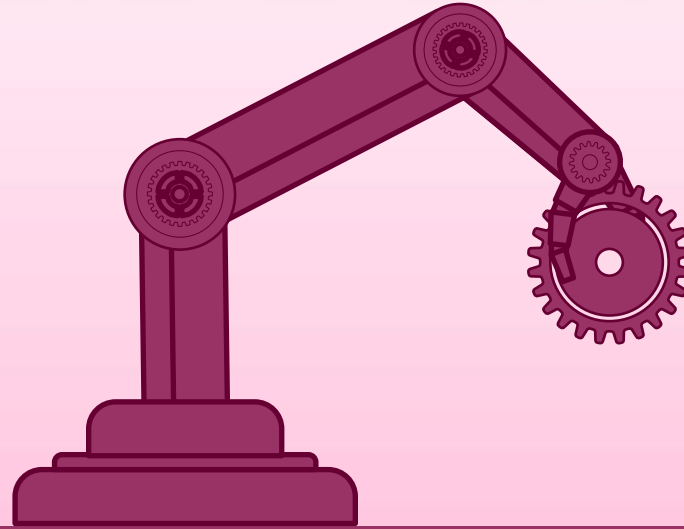
- **Pemrogram aplikasi senior (senior applications programmer)**
merupakan program komputer yang sudah berpengalaman dengan tugas merancang spesifikasi dari program aplikasi dan mengkoordinasi kerja dari pemrogram yang lainnya.
- **Pemrogram aplikasi (applicarions programmer)**
merupakan pemrogram komputer yang cukup berpengalaman dan dapat melakukan tugasnya tanpa harus bimbingan secara langsung lagi.
- **Pemrogram aplikasi junior (junior applications programmer)**
merupakan pemrogram komputer yang belum berpengalaman dan masih dibawah bimbingan dari pemrogram yang lebih senior.

Metodologi Pengembangan Sistem

Metodologi pengembangan system adalah suatu proses pengembangan system yang formal dan presisi yang mendefinisikan serangkaian aktivitas, metode, *best practices* dan *tools* yang terautomasi bagi para pengembang dan manager proyek dalam rangka mengembangkan dan merawat sebagai keseluruhan system informasi atau *software*.

Alasan perlunya Metodologi Pengembangan System!

- ☐ Menjamin adanya konsistensi proses.
- ☐ Dapat diterapkan dalam berbagai jenis proyek.
- ☐ Mengurangi resiko kesalahan dan pengambilan jalan pintas.
- ☐ Menuntut adanya dokumentasi yang konsisten yang bermanfaat bagi personal baru dalam timproyek.



Metode-Metode Pengembangan Sistem Informasi

❑ Metode System Development Life Cycle (SLDC)

Metode ini adalah metode pengembangan sistem informasi yang pertama kali digunakan makanya disebut dengan metode tradisional.

Tahap-tahap

- Melakukan survey dan menilai kelayakan proyek pengembangan sistem informasi.
- Mempelajari dan menganalisis sistem informasi yang sedang berjalan.
- Menentukan permintaan pemakai sistem informasi.
- Memilih solusi atau pemecahan masalah yang paling baik.
- Menentukan perangkat keras dan perangkat lunak computer.
- Merancang sistem informasi baru.
- Mengkomunikasikan dan mengimplementasikan sistem informasi baru.
- Memelihara dan melakukan perbaikan/peningkatan sistem informasi baru.

± SDLC

Plus

- Mudah diaplikasikan
- Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

Minus

- Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan
- Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk mengakomodasi ketidakpastian pada saat awal proyek.
- Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyek dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.
- Pengembang sering melakukan penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efisien.

± SDLC

Plus

- Mudah diaplikasikan
- Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

Minus

- Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan
- Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk mengakomodasi ketidakpastian pada saat awal proyek.
- Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyek dilalui. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.
- Pengembang sering melakukan penundaan yang tidak perlu karena anggota tim proyek harus menunggu tim lain untuk melengkapi tugas karena memiliki ketergantungan hal ini menyebabkan penggunaan waktu tidak efisien.

❑ Waterfall

Sering juga disebut model Sequential Linier. Metode pengembangan sistem yang paling tua dan paling sederhana. Cocok untuk pengembangan perangkat lunak dengan spesifikasi yang tidak berubah-ubah. Model ini menyediakan pendekatan alur hidup perangkat lunak secara sequential atau terurut dimulai dari analisa, desain, pengkodean, pengujian dan tahap pendukung

Tahap-Tahap :

- Analisis Kebutuhan Perangkat Lunak
- Desain
- Pembuatan Kode Program
- Pengujian
- Pendukung atau Pemeliharaan

± Waterfall

Plus

- Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
- Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.

Minus

- 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.
- Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

± Waterfall

Plus

- Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.
- Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi setiap fase atau tahapan akan mempunyai dokumen tertentu.

Minus

- 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.
- Pelanggan sulit menyatakan kebutuhan secara eksplisit sehingga tidak dapat mengakomodasi ketidakpastian pada saat awal pengembangan.

□ Prototyping

Prototyping adalah proses iterative dalam pengembangan sistem dimana requirement diubah ke dalam sistem yang bekerja (working system) yang secara terus menerus diperbaiki melalui kerjasama antara user dan analis.

Tahap-tahap

- Pengumpulan Kebutuhan
- Membangun Prototyping
- Menggunakan Sistem
- Mengkodekan Sistem
- Menguji Sistem
- Evaluasi Sistem
- Evaluasi Protootyping

± Prototyping

Plus

- Prototype melibatkan user dalam analisa dan desain.
- Punya kemampuan menangkap requirement secara konkret.
- Digunakan untuk memperluas SDLC.

Minus

- Proses analisis dan perancangan terlalu singkat.
- Mengesampingkan alternatif pemecahan masalah.
- Bisanya kurang fleksible dalam mengahdapi perubahan.
- Protitype yang dihasilkan tidak selamanya mudah dirubah dan cepat selesai.

± Prototyping

Plus

- Prototype melibatkan user dalam analisa dan desain.
- Punya kemampuan menangkap requirement secara konkret.
- Digunakan untuk memperluas SDLC.

Minus

- Proses analisis dan perancangan terlalu singkat.
- Mengesampingkan alternatif pemecahan masalah.
- Bisanya kurang fleksible dalam mengahdapi perubahan.
- Protitype yang dihasilkan tidak selamanya mudah dirubah dan cepat selesai.

❑ RAD (Rapid Application Development)

RAD adalah penggabungan beberapa metode atau teknik terstruktur. RAD menggunakan metode prototyping dan teknik terstruktur lainnya untuk menentukan kebutuhan user dan perancangan sistem informasi. Selain itu, RAD menekankan siklus perkembangan dalam waktu yang singkat (60 sampai 90 hari) dengan pendekatan konstruksi berbasis komponen.

Tahap-tahap

- Business Modelling
- Testing and Turnover
- Application Generation
- Process Modelling
- Data Modelling

± RAD

Plus

- RAD mengikuti tahapan pengembangan sistem seperti umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada (reusable object).
- Setiap fungsi dapat dimodulkan dalam waktu tertentu dan dapat dibicarakan oleh tim RAD yang terpisah dan kemudian diintegrasikan sehingga waktunya lebih efisien.

Minus

- Tidak cocok untuk proyek skala besar
- Proyek bisa gagal karena waktu yang disepakati tidak dipenuhi.
- Sistem yang tidak bisa dimodularisasi tidak cocok untuk model ini.
- Resiko teknis yang tinggi juga kurang cocok untuk model ini

❑ Spiral

Model spiral pada awalnya diusulkan oleh Boehm, adalah model proses perangkat lunak evolusioner yang merangkai sifat iteratif dari prototype dengan cara kontrol dan aspek sistematis model sequensial linier. Model iteratif ditandai dengan tingkah laku yang memungkinkan pengembang mengembangkan versi perangkat lunak yang lebih lengkap secara bertahap.

Tahapan-tahapan :

- Komunikasi Pelanggan
- Perencanaan
- Analisis Resiko
- Perekrayasaan
- Konstruksi dan Peluncuran
- Evaluasi Pelanggan

± Spiral

Plus

- Dapat disesuaikan agar perangkat lunak bisa dipakai selama hidup perangkat lunak komputer.
- Lebih cocok untuk pengembangan sistem dan perangkat lunak skala besar
- Pengembang dan pemakai dapat lebih mudah memahami dan bereaksi terhadap resiko setiap tingkat evolusi karena perangkat lunak terus bekerja selama proses
- Menggunakan prototipe sebagai mekanisme pengurangan resiko dan pada setiap keadaan di dalam evolusi produk.
- Tetap mengikuti langkah-langkah dalam siklus kehidupan klasik dan memasukkannya ke dalam kerangka kerja iterative.
- Membutuhkan pertimbangan langsung terhadap resiko teknis sehingga mengurangi resiko sebelum menjadi permasalahan yang serius.

Minus

- Sulit untuk menakutkan pelanggan bahwa pendekatan evolusioner ini bisa dikontrol.
- Memerlukan penaksiran resiko yang masuk akal dan akan menjadi masalah yang serius jika resiko mayor tidak ditemukan dan diatur.
- Butuh waktu lama untuk menerapkan paradigma ini menuju kepastian yang absolute

❑ Object Oriented Program

OOP merupakan cara pengembangan perangkat lunak berdasarkan abstraksi objek-objek yang ada di dunia nyata.

Dasar pembuatan adalah Objek, yang merupakan kombinasi antara struktur data dan perilaku dalam satu entitas. Filosofi Object Oriented sangat luar biasa sepanjang siklus pengembangan perangkat lunak (perencanaan, analisis, perancangan dan implementasi) sehingga dapat diterapkan pada perancangan sistem secara umum: menyangkut perangkat lunak, perangkat keras dan system secara keseluruhan.

Tahap-tahap :

- Model Objek
- Model Dinamis
- Model Fungsional

± Object Oriented Program

Plus

- Uniformity, OMT memungkinkan merancang user interface secara terintegrasi bersama dengan perancangan perangkat lunak sekaligus dengan perancangan basis data.
- Understandability, Kode-kode yang dihasilkan dapat diorganisasi ke dalam kelas-kelas yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah dipahami.
- Stability, Kode program yang dihasilkan relatif stabil sebab mendekati permasalahan sesungguhnya dilapangan.
- Reusability, Dimungkinkan penggunaan kembali kode-kode sehingga akan mempercepat waktu pengembangan perangkat lunak.

Minus

- Metode berorientasi objek merupakan konsep yang relatif baru sehingga belum ada standar yang diterima semua pihak dalam menentukan tool apa yang digunakan sebagai dasar analisis awal perancangan perangkat lunak.

❑ End-user Development (EUD)

Disini pengembangan dilakukan langsung oleh end-user. Keterlibatan langsung end-user sangat menguntungkan, karena memahami benar bagaimana sistem bekerja. Artinya tahap analisis sistem dapat dilakukan lebih cepat. Kelemahan adalah pada pengendalian mutu dan kecenderungan tumbuhnya “private” sistem informasi. Integrasi dengan sistem yang lain menjadi sulit.

Tahap-tahap

- Tahap inisiasi (initiation), Yaitu tahap dimana organisasi(perusahaan) mulai pertama kali mengenal teknologi informasi.
- Tahap ketularan (contagion)
- Tahap kendali (control)
- Tahap matang (mature)

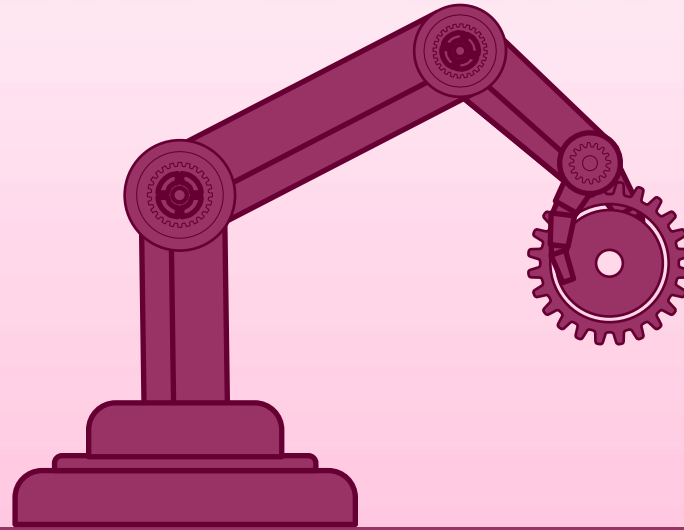
± End-user Development (EUD)

Plus

- Dapat menghindari permasalahan kemacetan di departemen sistem informasi.
- Kebutuhan pemakai sistem dapat lebih terpenuhi karena dapat dikembangkan sendiri oleh pemakai.
- Menambah atau meningkatkan partisipasi aktif pemakai dalam proses pengembangan sistemnya sehingga akan ada kepuasan sendiri dari pemakai sistem.
- Dapat menambah kualitas pemahaman pemakai terhadap aplikasi yang dikembangkan serta teknologi yang digunakan dalam sistem.

Minus

- Karena pemakai sistem harus mengembangkan aplikasinya sendiri, maka dalam hal ini pemakai sekaligus pengembang sistem dituntut untuk memiliki pemahaman mengenai teknologi informasi (computer literacy) serta pemahaman tentang pengembangan sistem informasi.
- End user computing memiliki resiko dapat mengganggu bahkan merusak sistem informasi di luar yang dikembangkan oleh pemakai sistem.
- End user computing pasti akan berhadapan dengan masalah kemampuan teknis pemakai sekaligus pengembang sistem.



Alat dan Teknik dalam Pengembangan Sistem

Alat Pengembangan Sistem

- Bagan Alir Dokumen (Mapping Chart) merupakan bagan alir yang menunjukkan arus dari laporan dan formulir atau dokumen
- Diagram Arus Data (Data Flow Diagram) digunakan pada metodologi structured, analisa dan desain. DFD sering digunakan untuk menggambarkan suatu sistem yang telah ada atau sistem baru yang akan dikembangkan lebih lanjut, DFD merupakan dokumentasi dari sistem yang baik.
- Kamus Data File (Data Dictionary) merupakan katalog fakta tentang data dan kebutuhan-kebutuhan informasi dari suatu sistem informasi.

Alat Pengembangan Sistem

- Relational Database merupakan hubungan dari file-file database yang digunakan dalam sistem yang dirancang. Penggambaran database relationship dilakukan pada proses normalisasi.
- Context Diagram merupakan diagram yang dapat menggambarkan hubungan keterkaitan antara sistem dengan pihak-pihak diluar lingkungan sistem dan posisi sistem dalam lingkungan tersebut. Pihak-pihak tersebut merupakan pihak-pihak yang membutuhkan informasi dan data bagi system. Hubungan keterkaitannya digambarkan sebagai aliran informasi dan data yang masuk kedalam sistem dan keluar sistem.
- Diagram Alir Informasi Diagram Alir Informasi adalah suatu diagram yang menggambarkan informasi yang mengalir dari suatu bagian ke bagian yang lainnya.
- Jadwal Implementasi Suatu jadwal yang menerangkan proses implementasi yang sudah direncanakan beserta perhitungan dari berbagai aktifitas

Alat Pengembangan Sistem

- **Use case diagram:** digunakan untuk mengetahui fungsionalitas tertentu dalam sebuah sistem.
- **Activity diagram :** diagram yang menunjukkan alur kerja sebuah bisnis atau proses operasinya.
- **Sequence diagram:** jenis diagram UML yang menunjukkan bagaimana interaksi yang terjadi dari objek ke objek beserta urutannya.
- **Class diagram :** diagram yang paling umum digunakan untuk melihat *class-class* dalam sebuah sistem berikut atribut, operasi, dan hubungan antar *class* yang ada.
- dll

Teknik Pengembangan Sistem

Teknik pengembangan sistem yang dimaksudkan dalam pengembangan sistem adalah teknik untuk mengumpulkan data dan menemukan fakta-fakta dalam kegiatan mempelajari sistem yang ada sehingga data-data yang diperoleh dapat diproses untuk dapat mengembangkan sistem. Adapun teknik-teknik yang digunakan adalah sebagai berikut :

- Teknik Wawancara
Wawancara memungkinkan analisis sistem mengumpulkan data secara langsung dengan pihak yang diwawancarai serta dapat langsung mengajukan pertanyaan-pertanyaan yang dirasa perlu dalam mengumpulkan data yang dibutuhkan.
- Teknik Observasi
Observasi adalah pengamatan langsung pada kegiatan yang sedang dilakukan
- dll

Tools Teknik Pemodelan Sistem

1. vEd-Graph Editor
2. ClickChartDiagram
3. Edraw MinMap
4. Pencil
5. Diagram Designer
6. Visio
7. Star UML
8. dll

Thank You