

Pertemuan 3

Metodologi Pengembangan Sistem Informasi

Tujuan :

1. Memahami metodologi pengembangan sistem (*System Development*) yang sesuai untuk sebuah proyek.
2. Memahami tugas-tugas yang perlu dilaksanakan dalam setiap tahapan proyek.
3. Memahami tugas dan dokumen yang dihasilkan selama fase pengembangan sistem.

Pendahuluan

- Pengembangan sistem informasi sering disebut sebagai proses pengembangan sistem (*System Development*).
- Pengembangan sistem didefinisikan sebagai: aktivitas untuk menghasilkan sistem informasi berbasis komputer untuk **menyelesaikan persoalan** (problem) organisasi atau **memanfaatkan kesempatan** (*opportunities*) yang timbul.

Pendahuluan (Cont.)

Banyak kegagalan sistem yang disebabkan karena Analisis sistem membuat sistem yang indah tanpa menjelaskan :

1. Bagaimana sistem dapat mendukung tujuan organisasi.
2. Bagaimana proses bisnis saat ini.
3. Bagaimana sistem informasi lainnya menjadi bernilai.

Sehingga analisis sistem harus melakukan analisis situasi bisnis, identifikasi kesempatan untuk pengembangan serta merancang sistem informasi untuk implementasinya.

Pendahuluan (Cont.)

Sebenarnya untuk menghasilkan sistem informasi tersebut terdiri dari:

- *System Analysis*: upaya mendapatkan gambaran bagaimana sistem bekerja dan masalah-masalah apa saja yang ada pada sistem
- *System Development*: langkah-langkah mengembangkan sistem informasi yang baru berdasarkan gambaran cara kerja sistem dan permasalahan yang ada

Metodologi Pengembangan Sistem

Proses-proses standard yang digunakan untuk membangun suatu sistem informasi meliputi langkah-langkah berikut ini:

- Analisa
- Desain
- Implementasi
- Maintenance

Pada perkembangannya, proses-proses standar tadi dituangkan dalam satu metode yang dikenal dengan nama *Systems Development Life Cycle* (SDLC) yang merupakan metodologi umum dalam pengembangan sistem yang menandai kemajuan dari usaha analisa dan desain.

System Develoment Life Cycle

Perencanaan (Planning)

Fase perencanaan merupakan proses fundamental untuk mengerti mengapa sistem informasi dibangun dan menentukan bagaimana team proyek akan membangunnya. Ada 2 tahapan :

1. Identifikasi dan seleksi proyek
2. Inisiasi dan perencanaan proyek

Analisa (Analysis)

Fase analisis akan menganalisis permasalahan lebih mendalam dengan menyusun studi kelayakan. Ada 6 dimensi :

1. Kelayakan Teknis

Menganalisis ketersediaan hardware, software dan pihak yang melaksanakan proses yang diperlukan.

2. Pengembalian Ekonomis

Menganalisis manfaat, penggunaan, dan potensi pengembalian secara ekonomis dari pengembangan sistem. Dengan memantau seberapa besar efisiensi yang dapat dilakukan, maka akan meningkatkan pendapatan dan laba perusahaan.

3. Pengembalian Non-ekonomis

Menganalisis manfaat, penggunaan, dan potensi dan keuntungan yang tidak dapat diukur secara finansial, spt : ketersediaan informasi secara akurat dan uptodate setiap saat, citra perusahaan, moral karyawan, layanan konsumen yang makin memikat dan posisi perusahaan dengan pesaing.

4. Hukum & Etika

Menganalisis apakah sistem yang dibuat akan beroperasi dengan batasan hukum dan etika dan kultur perusahaan.

5. Operasional

Menganalisis apakah sistem dapat diimplementasikan. Hal ini menyangkut analisis mengenai tempat, lingkungan dan sumber daya manusia yang akan mengoperasikan.

6. Jadwal

Menganalisis ketersediaan waktu sehingga sistem dapat selesai dan diimplementasikan.

Hasil dari tahapan ini berupa rekomendasi kelayakan sistem beserta dengan perkiraan biaya yang dibutuhkan.

Bila ditemukan ketidaklayakan dari hasil analisis maka perlu dilakukan kajian terhadap penyebab ketidaklayakan. Kemudian dilakukan pertimbangan secara cermat dan alternatif tindakan untuk mengatasinya.

Desain (Design)

Pada tahapan ini deskripsi dari requirement yang telah direkomendasikan diubah ke dalam spesifikasi sistem physical dan logical.

Logical Design

Bagian dari fase desain dalam SDLC dimana semua fitur-fitur fungsional dari sistem dipilih dari tahapan analisis dideskripsikan terpisah dari platform komputer yang nanti digunakan. Hasil dari tahapan ini adalah :

1. Deskripsi fungsional mengenai data dan proses yang ada dalam sistem baru.
2. Deskripsi yang detail dari spesifikasi sistem meliputi:
 - Input
 - Output
 - Process

Physical Design

Pada bagian ini spesifikasi logical diubah ke dalam detail teknologi dimana pemrograman dan pengembangan sistem bisa diselesaikan. Adapun output dari sistem ini adalah : Deskripsi teknikal. Deskripsi yang detail dari spesifikasi sistem meliputi: programs, files, network, system software.

Pada tahapan desain ada beberapa aktifitas utama yang dilakukan yaitu:

- Merancang dan mengintegrasikan network.
- Merancang Arsitektur aplikasi.
- Mendesain user interface.
- Mendesain sistem interface.
- Mendesain dan mengintegrasikan database.
- Memuat prototype untuk detail dari desain.
- Mendesain dan mengintegrasikan kendali sistem.

Implementasi (Implementation)

Tahapan kelima pada SDLC, dimana pada tahapan ini dilakukan beberapa hal yaitu:

- Coding
- Testing
- Instalasi

Output dari tahapan ini adalah : source code, prosedur pelatihan.

Operasional & Pemeliharaan (Operation & Support)

Sistem yang telah diuji dan dinyatakan bebas dari kesalahan mulai digunakan untuk proses bisnis yang sebenarnya. Selama sistem digunakan, tim teknis harus memperhatikan masalah pemeliharaan sistem untuk memelihara keutuhan data dan informasi perusahaan.

Pemeliharaan sistem secara rutin meliputi :

1. Penataan ulang database.
2. Memback-up data.
3. Scanning virus.

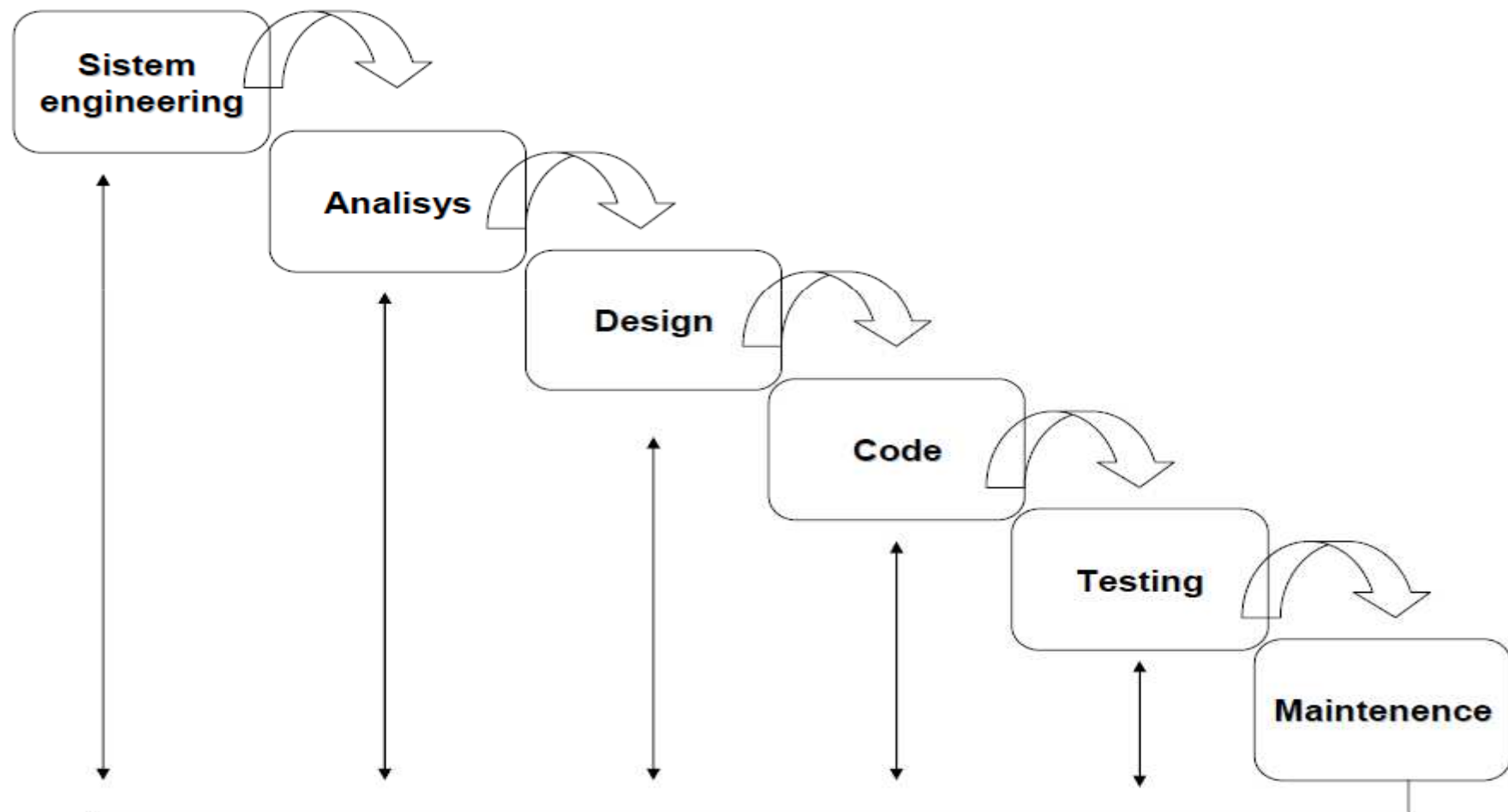
Metode Pengembangan Sistem

- Banyak metode pengembangan sistem yang tersedia.
- Metode yang paling dikenal disebut juga sebagai Waterfall.
- Metode-metode lain yang dikenal antara lain: Prototyping, Rapid Application Design, Spiral, dll.
- Pada bagian berikut akan dijelaskan metode-metode tersebut secara ringkas.

1. Waterfall

- Disebut juga siklus klasik (1970-an) dan sekarang ini lebih dikenal dengan sekuensial linier.
- Membutuhkan pendekatan sistematis dan sekuensial dalam pengembangan s/w.
- Dimulai dari analisis, desain, coding, testing dan pemeliharaan.

Waterfall



Keunggulan dan Kelemahan Waterfall

Keuntungan :

1. Mudah aplikasikan
2. Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

Kelemahan :

1. Jarang sekali proyek riil mengikuti aliran sekuensial yang dianjurkan model karena model ini bisa melakukan iterasi tidak langsung. Hal ini berakibat ada perubahan yang diragukan pada saat proyek berjalan.
2. Pelanggan sulit untuk menyatakan kebutuhan secara eksplisit sehingga sulit untuk mengakomodasi ketidakpastian pada saat awal proyek.
3. Pelanggan harus bersikap sabar karena harus menunggu sampai akhir proyek dilalui.
4. Sebuah kesalahan jika tidak diketahui dari awal akan menjadi masalah besar karena harus mengulang dari awal.

2. Prototyping

- Metode pengembangan perangkat lunak yang banyak digunakan.
- Pada metode ini pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem.
- Pengembang mengetahui dengan benar apa yang diinginkan pelanggan dengan tidak mengesampingkan segi-segi teknis dan pelanggan mengetahui proses-proses dalam menyelesaikan sistem yang diinginkan sehingga sistem yang dihasilkan sesuai dengan jadwal waktu penyelesaian yang telah ditentukan.

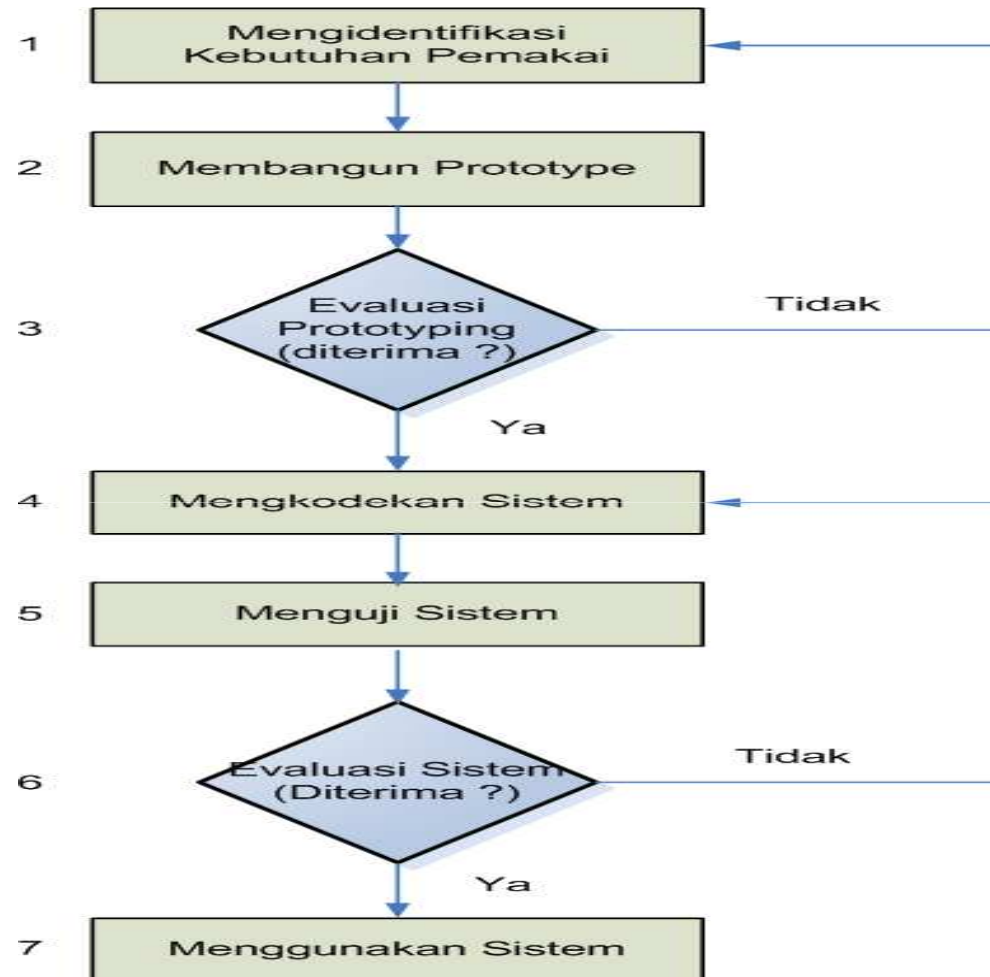


Diagram Metode Prototype

Tahapan-Tahapan Prototyping

Tahapan-tahapan dalam Prototyping adalah sebagai berikut:

1. Pengumpulan kebutuhan

Pelanggan dan pengembang bersama-sama mendefinisikan format seluruh perangkat lunak, mengidentifikasi semua kebutuhan, dan garis besar sistem yang akan dibuat.

2. Membangun prototyping

Membangun prototyping dengan membuat perancangan sementara yang berfokus pada penyajian kepada pelanggan (misalnya dengan membuat input dan format output).

3. Evaluasi prototyping

Evaluasi ini dilakukan oleh pelanggan apakah prototyping yang sudah dibangun sudah sesuai dengan keinginan pelanggan. Jika sudah sesuai maka langkah 4 akan diambil. Jika tidak prototyping direvisi dengan mengulangi langkah 1, 2, dan 3.

4. Mengkodekan sistem

Dalam tahap ini prototyping yang sudah di sepakati diterjemahkan ke dalam bahasa pemrograman yang sesuai.

5. Menguji sistem

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, harus dites dahulu sebelum digunakan. Pengujian ini dilakukan dengan White Box, Black Box, Basis Path, pengujian arsitektur dan lain-lain

6. Evaluasi Sistem

Pelanggan mengevaluasi apakah sistem yang sudah jadi sudah sesuai dengan yang diharapkan. Jika ya, langkah 7 dilakukan; jika tidak, ulangi langkah 4 dan 5.

7. Menggunakan sistem

Perangkat lunak yang telah diuji dan diterima pelanggan siap untuk digunakan .

Keunggulan dan Kelemahan Prototype

- Keunggulan:
 1. Komunikasi user dan pengembang intensif.
 2. User terlibat aktif dalam menentukan requirement.
 3. Waktu pengembangan relatif singkat.
 4. Implementasi mudah karena pemakai mengetahui dari awal apa yang akan diperolehnya.
- Kelemahan:
 1. Kemungkinan terjadi *shortcut* dalam pendefinisian masalah.
 2. Pemakai bisa terlalu berlebih menentukan requirement sehingga sulit dipenuhi.
 3. Kemungkinan tidak dihasilkan rancangan yang baik karena dokumentasi yang kurang rinci.

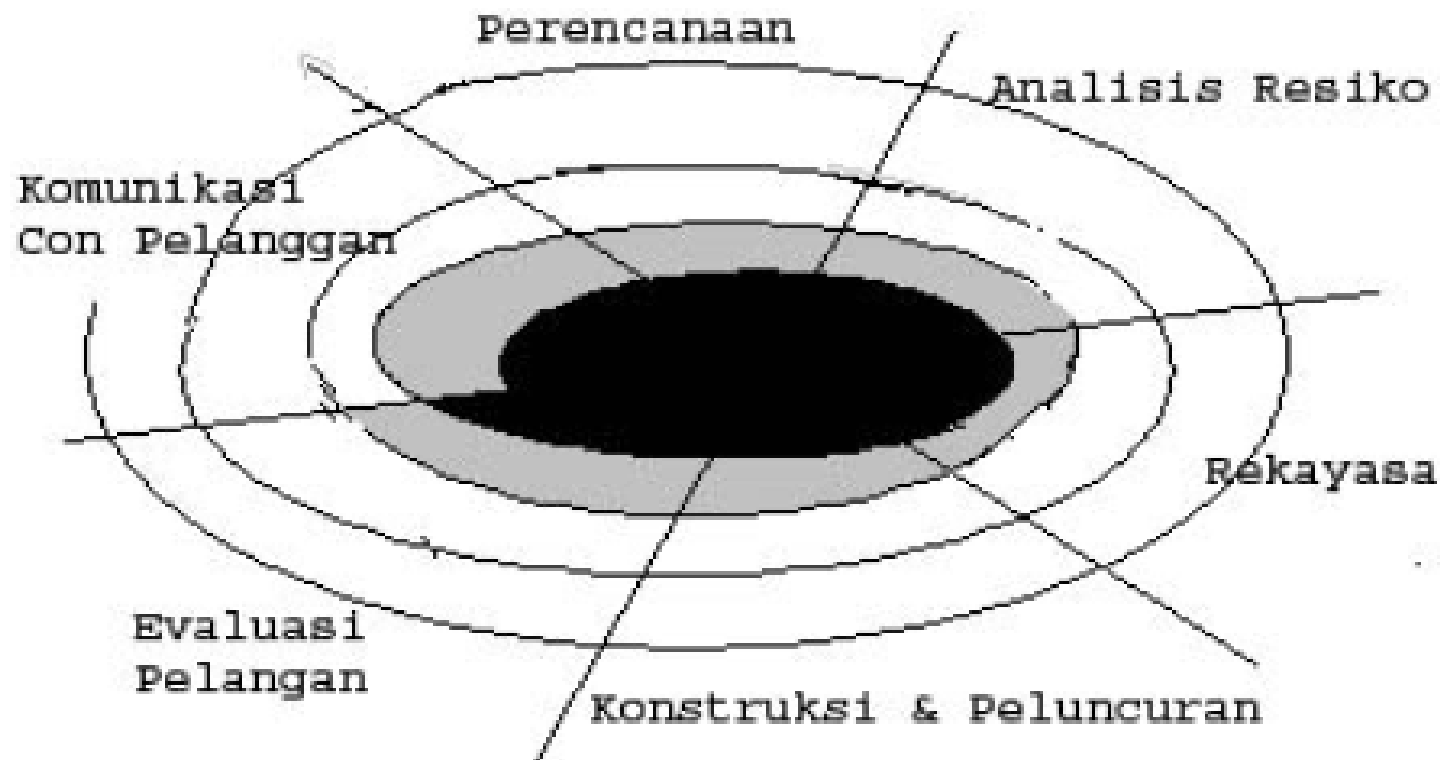
Prototyping bekerja dengan baik pada penerapan-penerapan yang berciri sebagai berikut:

1. Resiko tinggi yaitu untuk masalah-masalah yang tidak terstruktur dengan baik, ada perubahan yang besar dari waktu ke waktu, dan adanya persyaratan data yang tidak menentu.
2. Interaksi pemakai penting. Sistem harus menyediakan dialog on-line antara pelanggan dan komputer.
3. Perlunya penyelesaian yang cepat.
4. Perilaku pemakai yang sulit ditebak.
5. Sistem yang inovatif. Sistem tersebut membutuhkan cara penyelesaian masalah dan penggunaan perangkat keras yang mutakhir.
6. Perkiraan tahap penggunaan sistem yang pendek.

3. Spiral

- Metode ini dirancang secara revolusioner dengan tahapan yang jelas, tetapi terbuka bagi partisipasi customer untuk ikut serta menentukan pemodelan sistem.
- Metode ini lambat dan mahal karena setiap tahapan yang dilalui harus mengikutsertakan customer.
- Model ini merupakan perbaikan dari model waterfall dan prototype. Mengabungkan keuntungan model air terjun dan prototype dan memasukkan analisis resiko.
- Spiral melibatkan proses iterasi, dimana setiap iterasi bekerja pada satu level produk dimulai dari level prototype awal sampai pada level yang diinginkan. Setiap perpindahan level didahului analisis risiko.

Tahapan-Tahapan Metode Spiral



Tahapan-Tahapan Spiral

Tahapan-tahapan dalam Spiral adalah sebagai berikut:

1. Komunikasi pelanggan

Tugas-tugas untuk membangun komunikasi antara pelanggan dan kebutuhan-kebutuhan yang diinginkan oleh pelanggan.

2. Perencanaan

Tugas-tugas untuk mendefinisikan sumber daya, ketepatan, waktu, dan proyek informasi lain yg berhubungan.

3. Analisis Resiko

Tugas-tugas yang dibutuhkan untuk menaksir resiko manajemen dan teknis.

4. Perekayasaan

Tugas yang dibutuhkan untuk membangun satu atau lebih representasi dari aplikasi tersebut.

5. Konstruksi dan peluncuran

Tugas-tugas yang dibutuhkan untuk mengkonstruksi, menguji, memasang, dan memberi pelayanan kepada pemakai.

6. Evaluasi Pelanggan

Tugas-tugas untuk mendapatkan umpan balik dari pelanggan.

Keunggulan dan Kelemahan Spiral

Keunggulan:

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

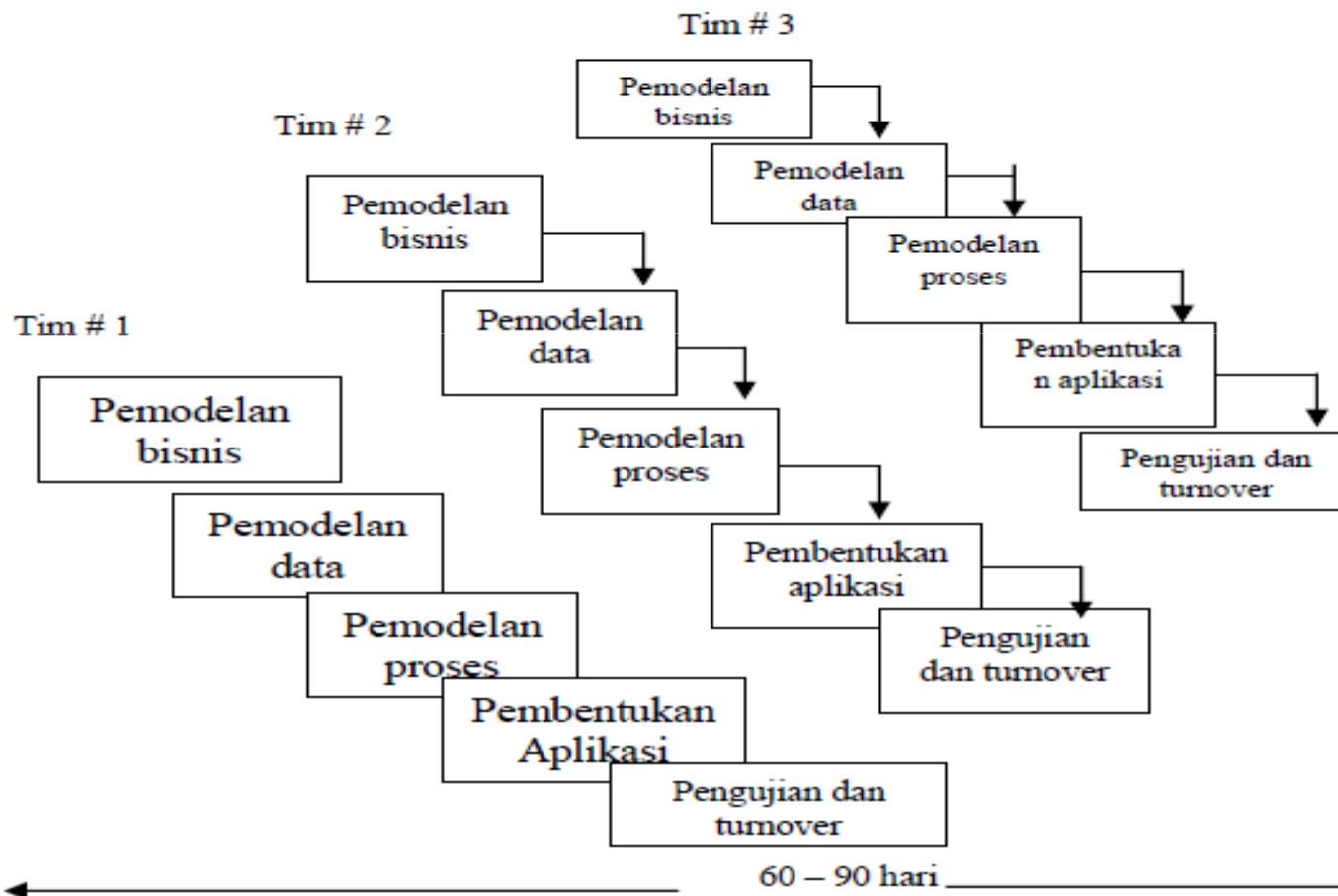
Kelemahan:

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

4. Rapid Application Design (RAD)

Rapid Application Development (RAD) adalah sebuah proses perkembangan perangkat lunak sekuensial linier yang menekankan siklus perkembangan dalam waktu yang singkat (60 sampai 90 hari) dengan pendekatan konstruksi berbasis komponen.

Tahapan-Tahapan Metode RAD



Tahapan-Tahapan Metode RAD

Metode RAD digunakan pada aplikasi sistem konstruksi, maka menekankan fase-fase sebagai berikut:

1. Bussiness Modelling

Fase ini untuk mencari aliran informasi yang dapat menjawab pertanyaan berikut:

Informasi apa yang mengendalikan proses bisnis?

Informasi apa yang dimunculkan?

Di mana informasi digunakan ?

Siapa yang memprosesnya ?

2. Data Modelling

Fase ini menjelaskan objek data yang dibutuhkan dalam proyek. Karakteristik (atribut) masing-masing data diidentifikasi dan hubungan antar objek didefinisikan

3. Process Modelling

Aliran informasi pada fase data modelling ditransformasikan untuk mendapatkan aliran informasi yang diperlukan pada implementasi fungsi bisnis. Pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atau mendapatkan kembali objek data tertentu.

4. Application Generation

Selain menggunakan bahasa pemrograman generasi ketiga, RAD juga memakai komponen program yang telah ada atau menciptakan komponen yang bisa dipakai lagi. Alat-alat bantu bisa dipakai untuk memfasilitasi konstruksi perangkat lunak.

5. Testing and Turnover

Karena menggunakan kembali komponen yang telah ada, maka akan mengurangi waktu pengujian. Tetapi komponen baru harus diuji dan semua interface harus dilatih secara penuh.

Keunggulan dan Kelemahan Spiral

Keunggulan:

1. Setiap fungsi mayor dapat dimodulkan dalam waktu tertentu kurang dari 3 bulan dan dapat dibicarakan oleh tim RAD yang terpisah dan kemudian diintegrasikan sehingga waktunya lebih efisien.
2. RAD mengikuti tahapan pengembangan sistem seperti umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada (*reusable object*) sehingga pengembang-pengembang tidak perlu membuat dari awal lagi dan waktu lebih singkat .

Kelemahan:

1. Proyek yang besar dan berkala, RAD memerlukan sumber daya manusia yang memadai untuk menciptakan jumlah tim yang baik.
2. RAD menuntut pengembang dan pelanggan memiliki komitmen dalam aktivitas rapid fire yang diperlukan untuk melengkapi sebuah sistem dalam waktu yang singkat. Jika komitmen tersebut tidak ada maka proyek RAD akan gagal.