



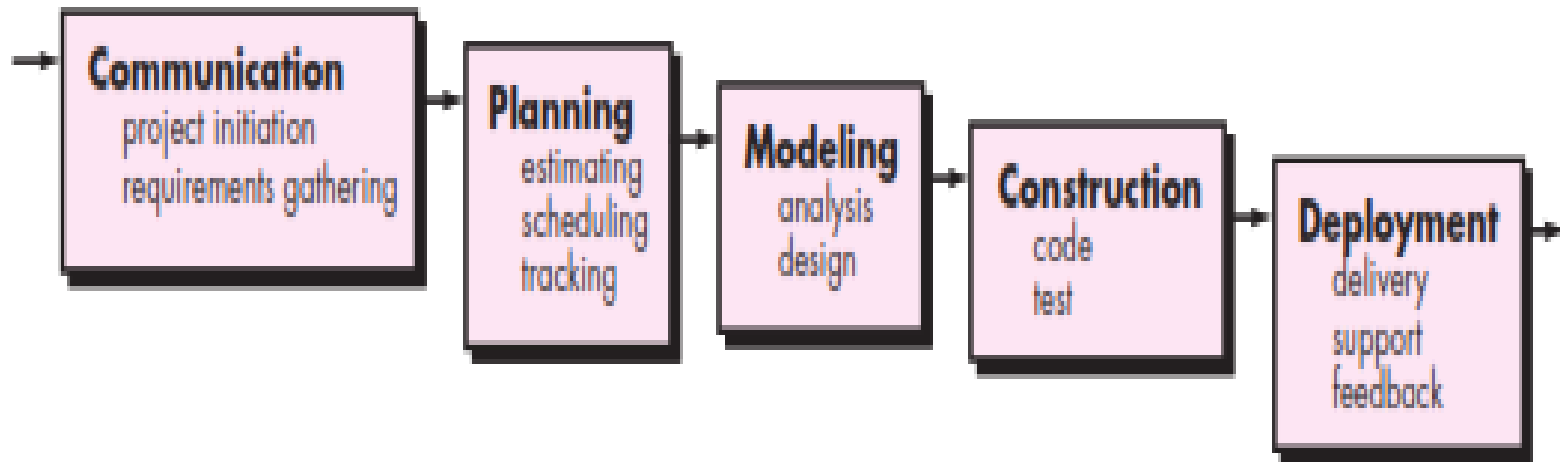
Metodologi Pengembangan Sistem Informasi Tradisional

Pertemuan 5

Macam-Macam Model Tradisional

1. Model Waterfall
2. Model Inkremental Proses
3. Model Proses Evolusi
4. Model Concurrent
5. Model Rational Unified Process

Waterfall



Gambar 1. Model Waterfall

- Model Waterfall terkadang disebut sebagai siklus hidup klasik.
- Tahapan Model Waterfall: Communication, Planning, Modelling, Contruction, Deployment.

Tahapan Waterfall

- **Communication (Project Initiation & Requirements Gathering)**
Sebelum memulai pekerjaan yang bersifat teknis, sangat diperlukan adanya komunikasi dengan customer demi memahami dan mencapai tujuan yang ingin dicapai. Hasil dari komunikasi tersebut adalah inisialisasi proyek seperti menganalisis permasalahan yang dihadapi dan mengumpulkan data-data yang diperlukan, serta membantu mendefinisikan fitur dan fungsi dari aplikasi. Pengumpulan data-data tambahan bisa juga diambil dari jurnal, artikel, paper dan internet.
- **Planning (Estimating, Scheduling, Tracking)**
Tahap berikutnya adalah tahapan perencanaan yang menjelaskan tentang estimasi tugas-tugas teknis yang akan dilakukan, resiko-resiko yang dapat terjadi, sumber daya yang diperlukan dalam membuat sistem, produk kerja yang ingin dihasilkan, penjadwalan kerja yang akan dilaksanakan, dan tracking proses pengerjaan sistem.

Tahapan Waterfall (Count...)

- Modeling (Analysis & Design)

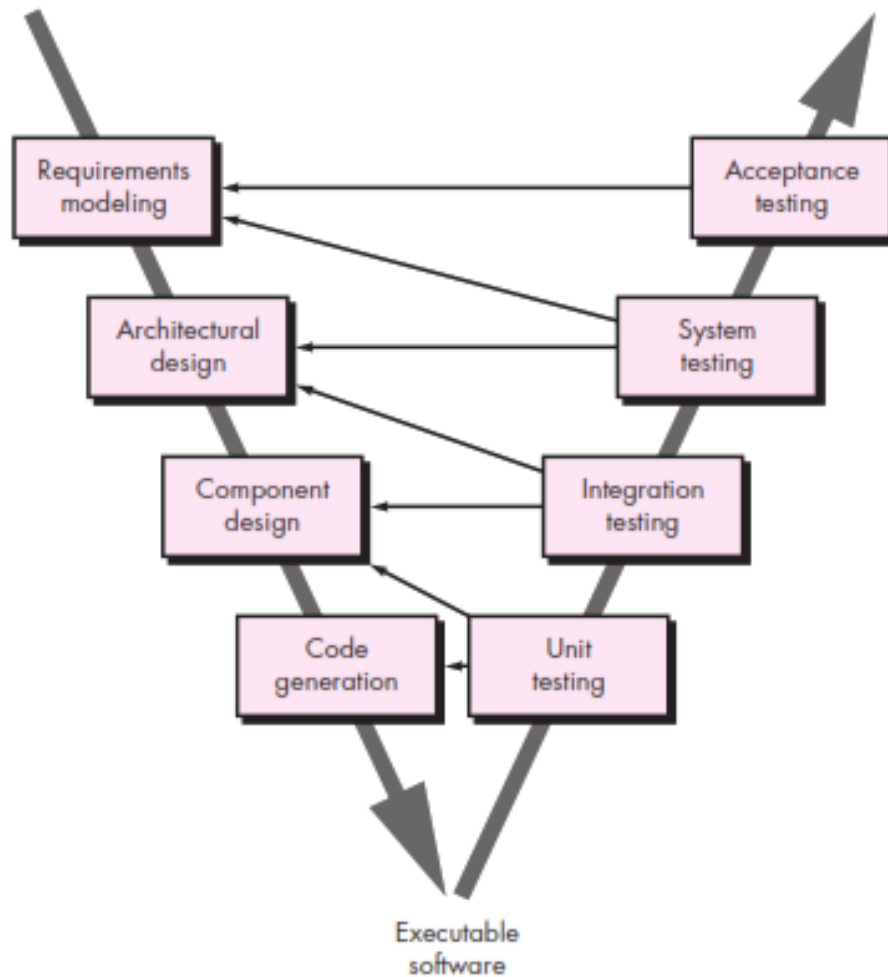
Tahapan ini adalah tahap perancangan dan pemodelan arsitektur sistem yang berfokus pada perancangan struktur data, arsitektur software, tampilan interface, dan algoritma program. Tujuannya untuk lebih memahami gambaran besar dari apa yang akan dikerjakan.

- Construction (Code & Test)

Tahapan ini merupakan proses penerjemahan bentuk desain menjadi kode atau bentuk bahasa yang dapat dibaca oleh mesin. Setelah pengkodean selesai, dilakukan pengujian terhadap sistem dan juga kode yang sudah dibuat. Tujuannya untuk menemukan kesalahan yang mungkin terjadi untuk nantinya diperbaiki.

- Deployment (Delivery, Support, Feedback)

Tahapan terakhir ini merupakan tahapan implementasi software ke customer, perbaikan software, evaluasi software, dan pengembangan software berdasarkan umpan balik yang diberikan agar sistem dapat tetap berjalan dan berkembang sesuai dengan fungsinya.



Gambar 2. Model Variasi Waterfall

- Merupakan perluasan model waterfall
- Tahapannya mirip dengan model waterfall
- Prosesnya dilakukan dengan percabangan

Tahapan Model V

1. Requirement Analysis & Acceptance Testing

Tahap Requirement Analysis sama seperti yang terdapat dalam model waterfall. Keluaran dari tahap ini adalah dokumentasi kebutuhan pengguna.

Acceptance Testing merupakan tahap yang akan mengkaji apakah dokumentasi yang dihasilkan tersebut dapat diterima oleh para pengguna atau tidak.

2. System Design & System Testing

Dalam tahap ini analisis sistem mulai merancang sistem dengan mengacu pada dokumentasi kebutuhan pengguna yang sudah dibuat pada tahap sebelumnya. Keluaran dari tahap ini adalah spesifikasi software yang meliputi organisasi sistem secara umum, struktur data, dan yang lain. Selain itu tahap ini juga menghasilkan contoh tampilan window dan juga dokumentasi teknik yang lain seperti Entity Diagram dan Data Dictionary.

3. Architecture Design & Integration Testing

Sering juga disebut High Level Design. Dasar dari pemilihan arsitektur yang akan digunakan berdasar kepada beberapa hal seperti: pemakaian kembali tiap modul, ketergantungan tabel dalam basis data, hubungan antar interface, detail teknologi yang dipakai.

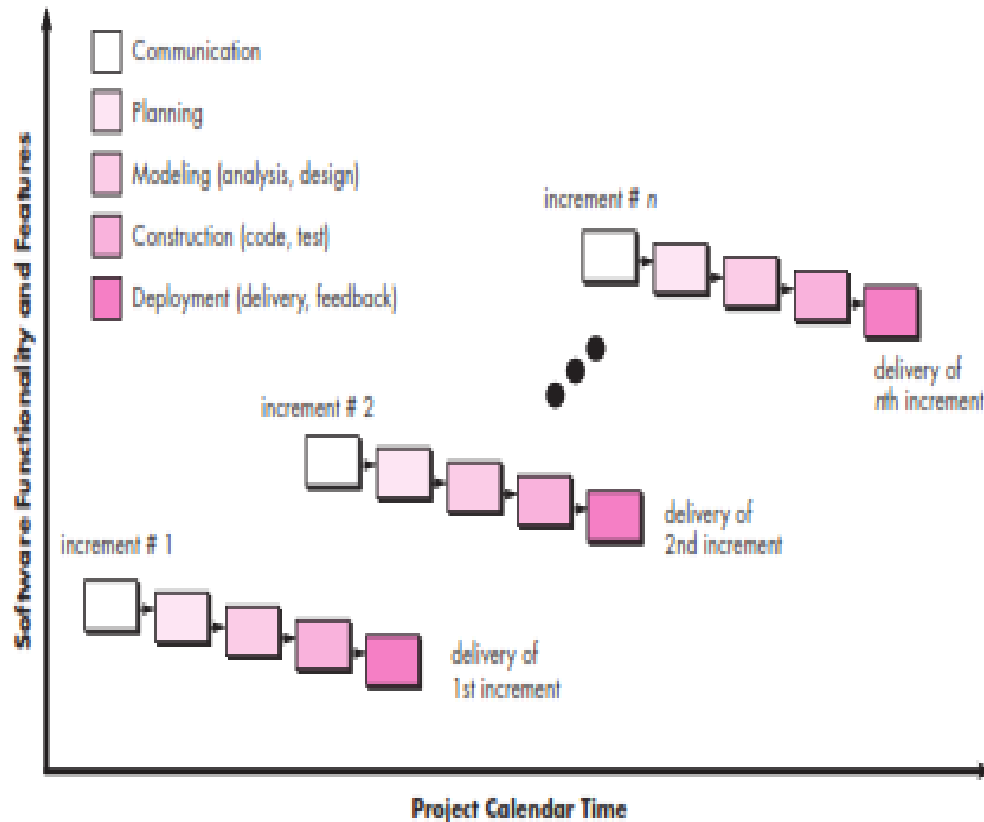
4. **Module Design & Unit Testing**

Sering juga disebut sebagai Low Level Design. Perancangan dipecah menjadi modul-modul yang lebih kecil. Setiap modul tersebut diberi penjelasan yang cukup untuk memudahkan programmer melakukan coding. Tahap ini menghasilkan spesifikasi program seperti: fungsi dan logika tiap modul, pesan kesalahan, proses input-output untuk tiap modul, dan lain-lain.

5. **Coding**

Dalam tahap ini dilakukan pemrograman terhadap setiap modul yang sudah dibentuk.

Inkremental Proses



Gambar 3. Inkremental Proses

- Model yang merupakan gabungan dari elemen aliran proses linear dan paralel.
- Setiap urutan liner menghasilkan “peningkatan” yang dapat dikirim dari perangkat lunak dengan cara yang mirip dengan peningkatan yang dihasilkan oleh aliran proses evolusi.

Tahapan Inkremental Proses

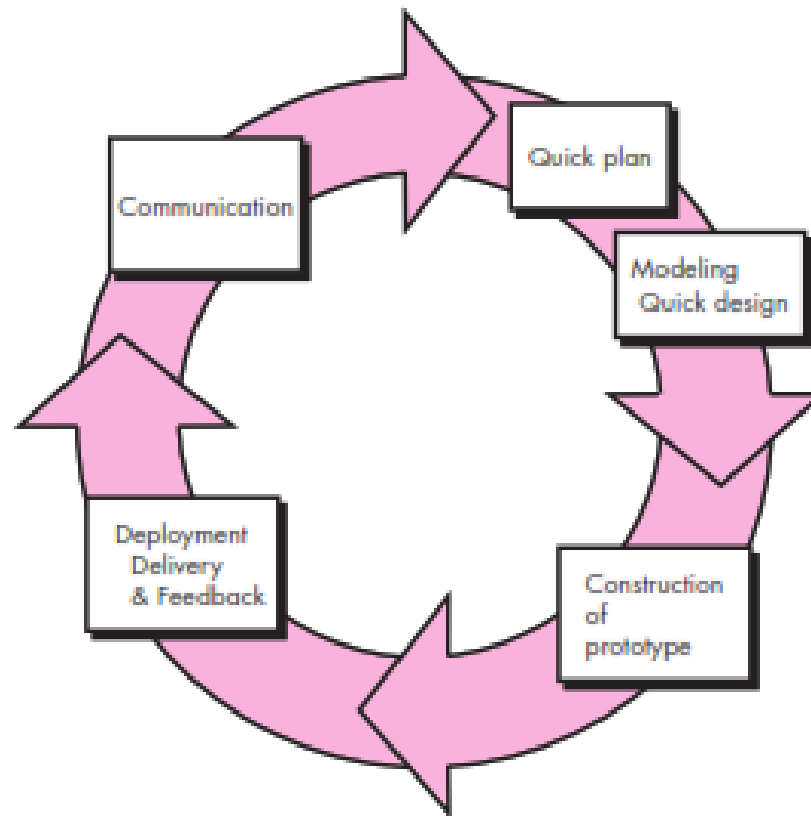
1. Kombinasikan element-element dari waterfall dengan sifat iterasi/perulangan element-element dalam waterfall dikerjakan dengan hasil berupa produk dengan spesifikasi tertentu, kemudian proses dimulai dari fase pertama hingga akhir dan menghasilkan produk dengan spesifikasi yang lebih lengkap dari yang sebelumnya. Demikian seterusnya hingga semua spesifikasi memenuhi kebutuhan yang ditetapkan oleh pengguna.
2. Produk hasil increment pertama biasanya produk inti (core product), yaitu produk yang memenuhi kebutuhan dasar. Produk tersebut digunakan oleh pengguna atau menjalani review/pengecekan detil. Hasil review tersebut menjadi bekal untuk pembangunan pada increment berikutnya. Hal ini terus dikerjakan sampai produk yang komplit dihasilkan. Model ini cocok jika jumlah anggota tim pengembang/pembangun PL tidak cukup.
3. Mampu mengakomodasi perubahan secara fleksibel.
4. Produk yang dihasilkan pada increment pertama bukanlah prototype, tapi produk yang sudah bisa berfungsi dengan spesifikasi dasar.

Proses Evolusi

Model evolusi bersifat iteratif. Model ini ditandai dengan cara yang memungkinkan Anda untuk mengembangkan versi perangkat lunak yang semakin lengkap

- Model Prototype
- Model Spiral

Model Prototype



Gambar 4. Model Prototype

Tahapan Prototype

1. Mendengarkan Pelanggan

Pada tahap ini dilakukan pengumpulan kebutuhan dari sistem dengan cara mendengar kebutuhan dari pelanggan. Untuk membuat suatu sistem yang sesuai kebutuhan, maka harus diketahui terlebih dahulu bagaimana sistem yang sedang berjalan untuk kemudian mengetahui masalah yang terjadi.

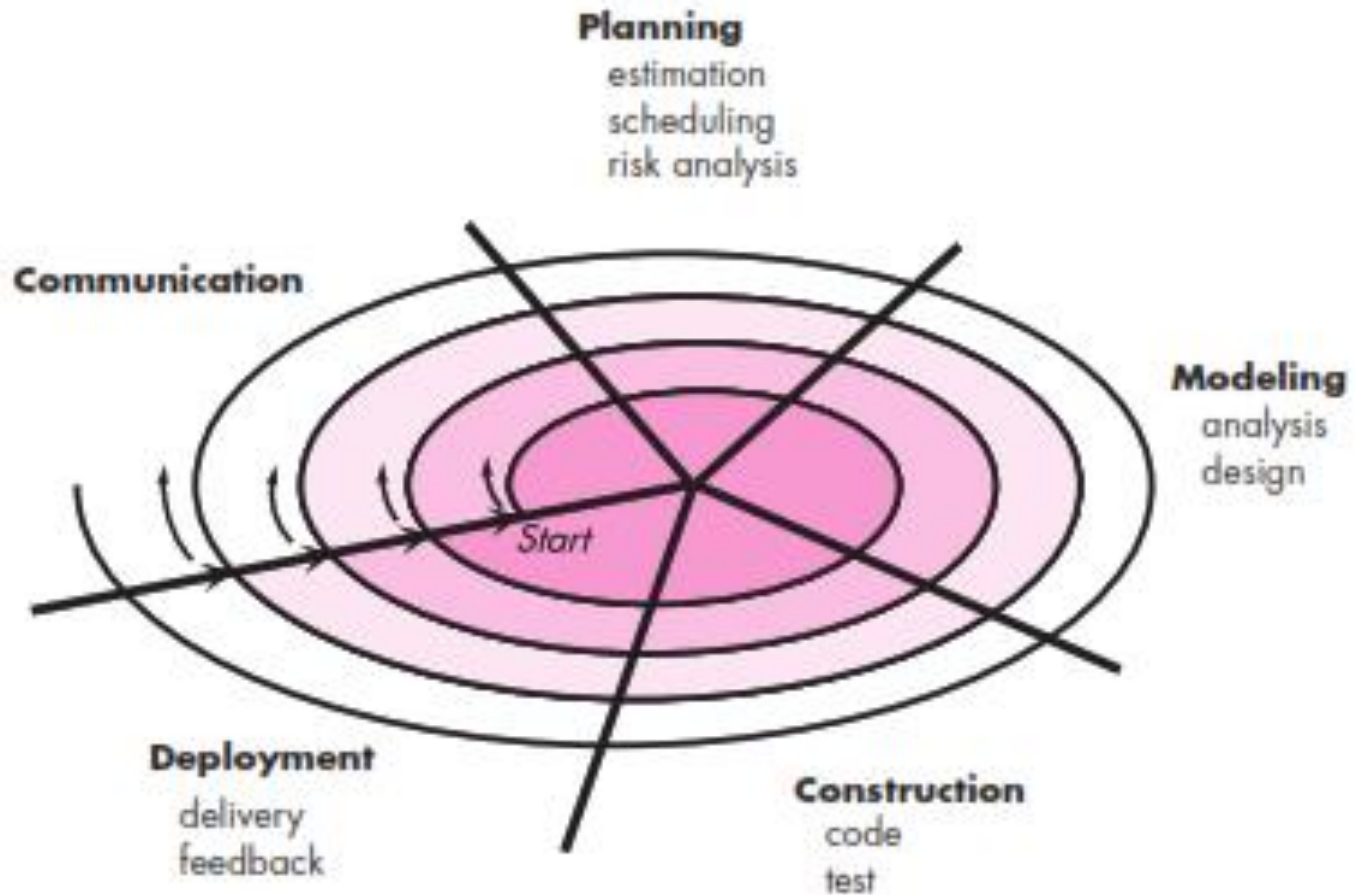
2. Merancang dan Membuat Prototype

Pada tahap ini, dilakukan perancangan dan pembuatan prototype system. Prototype yang dibuat disesuaikan dengan kebutuhan sistem yang telah didefinisikan sebelumnya dari kebutuhan pelanggan atau pengguna.

3. Uji Coba

Pada tahap ini, prototype dari sistem di uji coba oleh pelanggan atau pengguna. Kemudian dilakukan evaluasi kekurangan-kekurangan dari kebutuhan pelanggan. Pengembangan kemudian kembali mendengarkan keluhan dari pelanggan untuk memperbaiki prototype yang ada.

Model Spiral



Gambar 5. Model Spiral



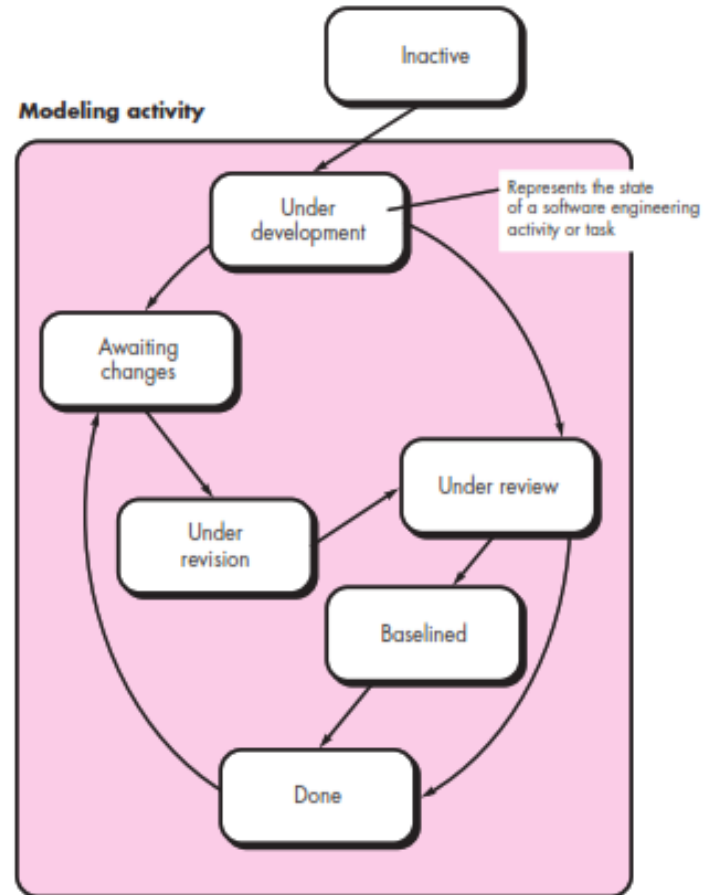
Fase Model Spiral

- Communication
- Planning (Estimation, Schedulling, Risk Analysis)
- Modeling (Analysis Design)
- Construction (Code & Testing)
- Deployment (Delivery, & Feedback)

Setiap Loop dibagi menjadi beberapa sector:

1. **Objective settings** (menentukan tujuan): menentukan tujuan dari fase yang ditentukan. Batasan-batasan pada proses dan produk sudah diketahui. Perencanaan sudah disiapkan. Resiko dari proyek sudah diketahui. Alternatif strategi sudah disiapkan berdasarkan resiko-resiko yang diketahui, dan sudah direncanakan.
2. **Risk assessment and reduction** (Penanganan dan pengurangan resiko): setiap resiko dianalisis secara detil pada sektor ini. Langkah-langkah penanganan dilakukan, misalnya membuat prototype untuk mengetahui ketidakcocokan kebutuhan.
3. **Development and Validation** (Pembangunan dan pengujian): Setelah evaluasi resiko, maka model pengembangan sistem dipilih. Misalnya jika resiko user interface dominan, maka membuat prototype User Interface. Jika bagian keamanan yang bermasalah, maka menggunakan model formal dengan perhitungan matematis, dan jika masalahnya adalah integrasi sistem model waterfall lebih cocok.
4. **Planning**: Proyek dievaluasi atau ditinjau-ulang dan diputuskan untuk terus ke fase loop selanjutnya atau tidak. Jika melanjutkan ke fase berikutnya rencana untuk loop selanjutnya.

Concurrent

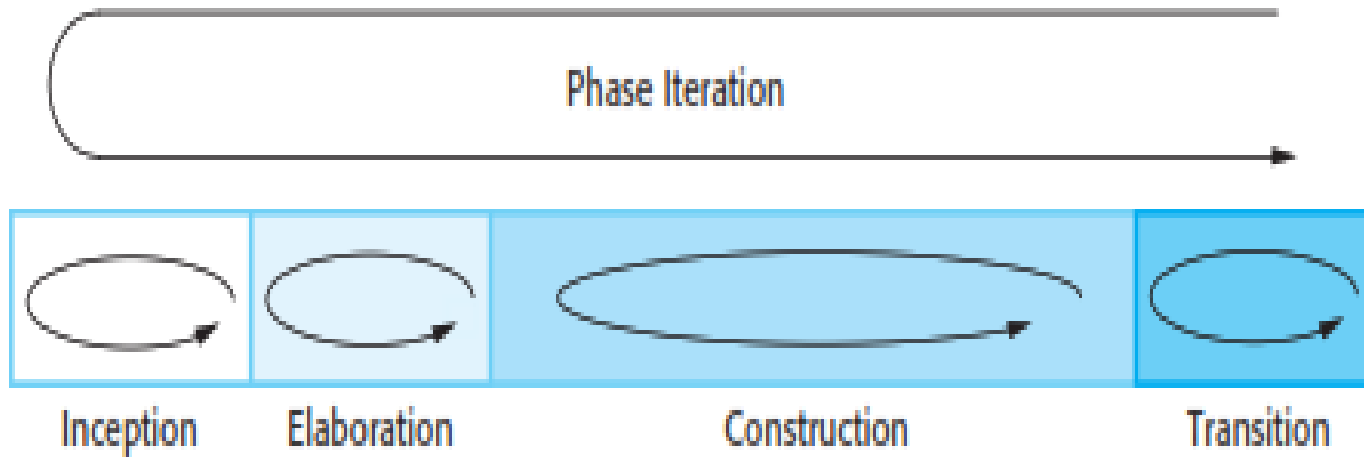


Gambar 6. Model Concurrent

Concurrent (Count..)

- Pemodelan Concurrent mendefinisikan serangkaian acara yang akan memicu transisi dari negara ke negara untuk masing-masing kegiatan rekayasa perangkat lunak, tindakan, atau tugas.
- Pemodelan Concurrent ini berlaku untuk semua jenis pengembangan perangkat lunak dan memberikan gambaran yang akurat tentang keadaan proyek.
- Setiap kegiatan, tindakan, atau tugas pada jaringan berjalan bersamaan dengan kegiatan, tindakan, atau tugas lain .
- Event yang dihasilkan pada satu titik dalam memicu transisi jaringan proses antara states.

Rational Unified Process



Gambar 7. Model Rational Unified Process (RUP)

Langkah-langkah Pada RUP

1. **Inception**, tujuan dari fase awal adalah untuk membangun kasus bisnis untuk sebuah sistem. Anda harus mengidentifikasi semua entitas eksternal (orang dan sistem) yang akan berinteraksi dengan sistem dan menentukan interaksi ini. Kemudian menggunakan informasi ini untuk menilai kontribusi yang diberikan sistem kepada bisnis. Jika kontribusi ini kecil, maka proyek dapat dibatalkan setelah fase ini.
2. **Elaboration**, tujuan fase elaborasi adalah untuk mengembangkan pemahaman masalah domain, membangun kerangka kerja arsitektur untuk sistem, mengembangkan rencana proyek, dan mengidentifikasi risiko proyek utama. Setelah menyelesaikan fase ini, Anda harus memiliki model persyaratan untuk sistem, yang mungkin berupa serangkaian kasus penggunaan UML, deskripsi arsitektur, dan rencana pengembangan untuk perangkat lunak

Langkah-langkah Pada RUP (Count..)

3. **Contruction**, tahap konstruksi melibatkan desain sistem, pemrograman, dan pengujian. Bagian dari sistem dikembangkan secara paralel dan terintegrasi selama fase ini. Setelah menyelesaikan fase ini, Anda harus memiliki sistem perangkat lunak yang berfungsi dan dokumentasi terkait yang siap dikirim ke pengguna.
4. **Transition**, fase terakhir dari RUP berkaitan dengan pemindahan system dari komunitas pengembangan ke komunitas pengguna dan membuatnya bekerja di lingkungan nyata. Ini sesuatu yang diabaikan dalam sebagian besar model proses perangkat lunak tetapi, pada kenyataannya, merupakan kegiatan yang mahal dan terkadang bermasalah. Setelah menyelesaikan fase ini, Anda harus memiliki sistem perangkat lunak terdokumentasi yang berfungsi dengan benar di lingkungan operasionalnya.