

BAB I

PENGEMBANGAN SISTEM

A. Model Siklus Kehidupan Klasik

Model Sekuensial Linier sering disebut Model Air Terjun merupakan paradigma rekayasa perangkat lunak yang paling tua dan paling banyak dipakai. Model ini mengusulkan sebuah pendekatan perkembangan perangkat lunak yang sistematis dan sekuensial yang dimulai pada tingkat dan kemajuan sistem pada seluruh analisis, desain, kode, pengujian, dan pemeliharaan.

Tahapan-tahapan Model Sekuensial Linier

Model Sekuensial Linier mengikuti aktivitas-aktivitas yaitu:

1. Rekayasa dan Pemodelan Sistem/Informasi

Karena perangkat lunak merupakan bagian dari suatu sistem maka langkah pertama dimulai dengan membangun syarat semua elemen sistem dan mengalokasikan ke perangkat lunak dengan memperhatikan hubungannya dengan manusia, perangkat keras dan database.

2. Analisis Kebutuhan Perangkat Lunak

Proses menganalisis dan pengumpulan kebutuhan sistem yang sesuai dengan domain informasi tingkah laku, unjuk kerja, dan antar muka (*interface*) yang diperlukan. Kebutuhan-kebutuhan tersebut didokumentasikan dan dilihat lagi dengan pelanggan.

3. Desain

Proses desain akan menerjemahkan syarat kebutuhan ke sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuat coding. Proses ini berfokus pada : struktur data, arsitektur perangkat lunak, representasi interface, dan detail (algoritma) prosedural.

4. Pengkodean (Coding)

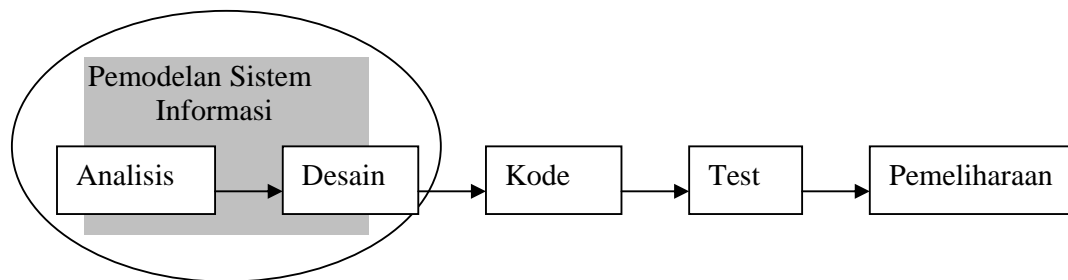
Pengkodean merupakan proses menerjemahkan desain ke dalam suatu bahasa yang bisa dimengerti oleh komputer.

5. Pengujian

Proses pengujian dilakukan pada logika internal untuk memastikan semua pernyataan sudah diuji. Pengujian eksternal fungsional untuk menemukan kesalahan-kesalahan dan memastikan bahwa input akan memberikan hasil yang aktual sesuai yang dibutuhkan

6. Pemeliharaan

Perangkat lunak yang sudah disampaikan kepada pelanggan pasti akan mengalami perubahan. Perubahan tersebut bisa karena mengalami kesalahan karena perangkat lunak harus menyesuaikan dengan lingkungan (peripheral atau sistem operasi baru) baru, atau karena pelanggan membutuhkan perkembangan fungsional atau unjuk kerja.



Keunggulan dan Kelemahan Model Sekuensial Linier

a. Keunggulan

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

b. Kelemahan

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

4. 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

B. Prototype

Prototyping merupakan salah satu metode pengembangan perangkat lunak yang banyak digunakan. Dengan metode prototyping ini pengembang dan pelanggan dapat saling berinteraksi selama proses pembuatan sistem.

Seing terjadi seorang pelanggan hanya mendefinisikan secara umum apa yang dikehendaknya tanpa menyebutkan secara detail output apa saja yang dibutuhkan, pemrosesan dan data-data apa saja yang dibutuhkan. Sebaliknya disini pengembang kurang memperhatikan efisiensi algoritma, kemampuan sistem operasi dan interface yang menghubungkan manusia dan komputer.

Untuk mengatasi ketidakserasian antara pelanggan dan pengembang , maka harus dibutuhkan kerjasama yang baik diantara keduanya sehingga pengembang akan mengetahui dengan benar apa yang diinginkan pelanggan dengan tidak mengesampingkan segi-segi teknis dan pelanggan akan mengetahui proses-proses dalam menyelesaikan sistem yang diinginkan. Dengan demikian akan menghasilkan sistem sesuai dengan jadwal waktu penyelesaian yang telah ditentukan.

Kunci agar model prototype ini berhasil dengan baik adalah dengan mendefinisikan aturan-aturan main pada saat awal, yaitu pelanggan dan pengembang harus setuju bahwa prototype dibangun untuk mendefinisikan kebutuhan. Prototype akan dihilangkan sebagian atau seluruhnya dan perangkat lunak aktual aktual direkayasa dengan kualitas dan implementasi yang sudah ditentukan

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 disepakati 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 Prototyping

Keunggulan prototyping adalah:

1. Adanya komunikasi yang baik antara pengembang dan pelanggan
2. Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan
3. Pelanggan berperan aktif dalam pengembangan sistem
4. Lebih menghemat waktu dalam pengembangan sistem
5. Penerapan menjadi lebih mudah karena pemakai mengetahui apa yang dikehendukannya.

Kelemahan prototyping adalah :

1. Pelanggan kadang tidak melihat atau menyadari bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan dan juga belum memikirkan kemampuan pemeliharaan untuk jangka waktu lama.

2. pengembang biasanya ingin cepat menyelesaikan proyek. Sehingga menggunakan algoritma dan bahasa pemrograman yang sederhana untuk membuat prototyping lebih cepat selesai tanpa memikirkan lebih lanjut bahwa program tersebut hanya merupakan cetak biru sistem .
3. Hubungan pelanggan dengan komputer yang disediakan mungkin tidak mencerminkan teknik perancangan yang baik

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

C. Model 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. Perangkat lunak dikembangkan dalam deretan pertambahan. Selama awal iterasi, rilis inkremental bisa berupa model/prototype kertas, kemudian sedikit demi sedikit dihasilkan versi sistem yang lebih lengkap.

Tahapan-Tahapan Model Spiral

Model spiral dibagi menjadi enam wilayah tugas yaitu:

1. Komunikasi pelanggan

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

2. Perencanaan

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

3. Analisis Resiko

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

4. Perencanaan

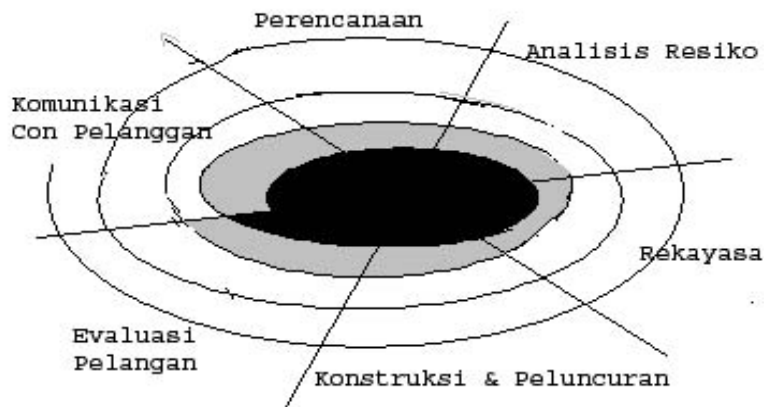
Yaitu tugas yang dibutuhkan untuk membangun satu atau lebih representasi dari aplikasi tersebut.

5. Konstruksi dan peluncuran

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

6. Evaluasi Pelanggan

Yaitu tugas-tugas untuk mendapatkan umpan balik dari pelanggan.



Dari gambar tersebut, proses dimulai dari inti bergerak searah dengan jarum jam mengelilingi spiral. Lintasan pertama putaran menghasilkan perkembangan spesifikasi produk. Putaran selanjutnya digunakan untuk mengembangkan sebuah prototype, dan secara progresif mengembangkan versi perangkat lunak yang lebih canggih. Masing-masing lintasan yang melalui daerah perencanaan menghasilkan penyesuaian pada rencana proyek. Biaya dan jadwal disesuaikan berdasarkan umpan balik yang disimpulkan dari evaluasi pelanggan. Manajer proyek akan menambah jumlah iterasi sesuai dengan yang dibutuhkan.

Kelebihan dan Kelemahan Model Spiral

a. Kelebihan model Spiral :

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.

b. Kelemahan model Spiral:

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

D. Rapid Application Development

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.

Tahap-Tahapan dalam RAD

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

1. Business 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

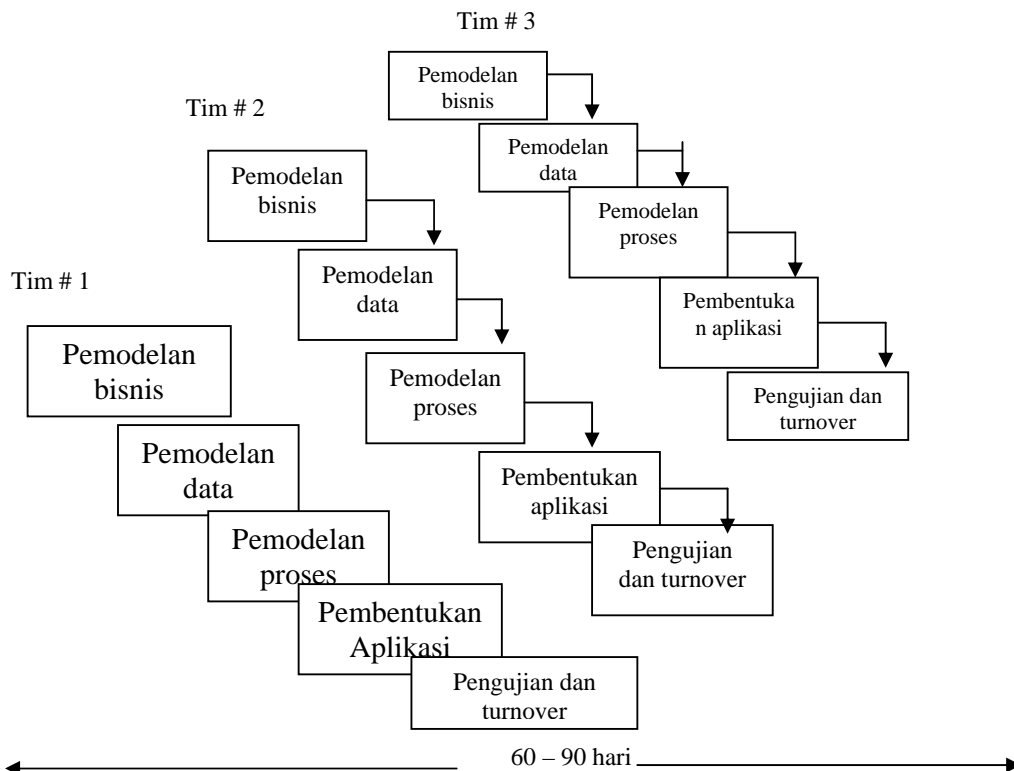
Aliran informasi pada fase data modelling ditransformasikan untuk mendapatkan aliran informasi yang diperlukan pada implementasi fungsi bisnis. Pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atau mendapatkan kembali objek data tertentu

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 Model RAD

a. Keunggulan Model RAD

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 .

b. Kelemahan Model RAD :

1. Proyek yang besar dan berskala, 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.

E. Object Oriented Technology

A. Pengantar Object Oriented Technology

Object Oriented Technology 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 sistem secara keseluruhan.

Dalam pengembangan sistem berorientasi objek ini , konsep-konsep dan sifat-sifat object oriented digunakan. Konsep-konsep tersebut adalah:

1. Kelas

Kelas adalah konsep OO yang mengencapsulasi/membungkus data dan abstraksi prosedural yang diperlukan untuk menggambarkan isi dan tingkah laku berbagai entitas. Kelas juga merupakan deskripsi tergeneralisir (misl template, pola, cetak biru) yang menggambarkan kumpulan objek yang sama.

2. Objek

Objek digambarkan sebagai benda, orang, tempat dan sebagainya yang ada di dunia nyata yang penting bagi suatu aplikasi. Objek mempunyai atribut dan metoda .

3. Atribut

Atribut menggambarkan data yang dapat memberikan informasi kelas atau objek dimana atribut tersebut berada.

4. Metoda/Servis/Operator

Metoda adalah prosedur atau fungsi yang tergabung dalam objek bersama dengan atribut. Metode ini digunakan untuk pengaksesan terhadap data yang terdapat dalam objek tersebut.

5. Message

Message adalah alat komunikasi antar objek. Hubungan antar objek ditentukan oleh problem domain dan tanggung jawab sistem.

6. Event

Event adalah suatu kejadian pada waktu yang terbatas yang menggambarkan rangsangan (stimulus) dari luar sistem.

7. State

State adalah abstraksi dari nilai atribut dan link dalam sebuah objek. State merupakan tanggapan dari objek terhadap event-event masukan.

8. Skenario

Skenario adalah urutan event yang terjadi sepanjang eksekusi sistem

Karakteristik-karakteristik yang terdapat dalam metode pengembangan sistem berorientasi objek adalah:

- **Encapsulation**

Encapsulation merupakan dasar untuk membatasi ruang lingkup program terhadap data yang diproses. Data dan prosedur dikemas dalam suatu objek sehingga prosedur lain dari luar tidak dapat mengaksesnya. Data akan terlindungi dari prosedur atau objek lain.

- **Inheritance**

Inheritance (pewarisan) adalah teknik yang menyatakan bahwa anak dari objek akan mewarisi data/atribut dan metode dari induknya langsung. Suatu kelas dapat ditentukan secara umum, kemudian ditentukan secara spesifik menjadi subkelas. Setiap subkelas mempunyai hubungan atau mewarisi semua sifat yang dimiliki kelas induknya dan ditambah dengan sifat nik yang dimilikinya.

- **Polymorphism**

Polymorphism menyatakan bahwa sesuatu yang sama dapat mempunyai bentuk dan perilaku berbeda. Polimorfisme juga menyatakan bahwa operasi yang sama mungkin mempunyai perbedaan kelas.

A. Tahapan-Tahapan Object Oriented Technology

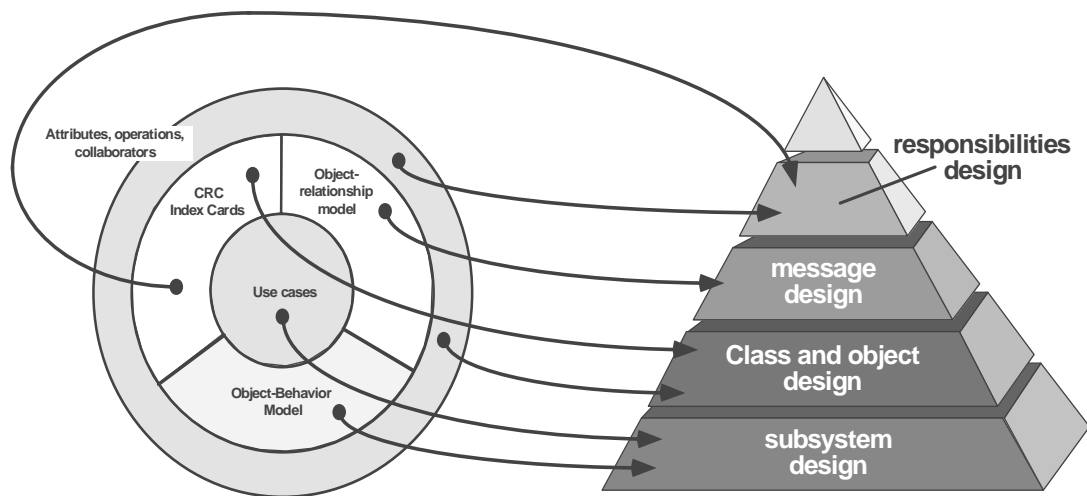
Pada Object Oriented Technology ada beberapa metode yang digunakan dalam pengembangan sistem. Salah satu yang terkenal adalah **OMT** (*Object Modelling Technique*) yang diciptakan oleh **Rambough**

Aktivitas-aktivitas yang dilakukan dalam OMT ini adalah:

- Model Objek
- Model Dinamis
- Model Fungsional

Dalam pengembangan sistem berbasis objek diperlukan tahapan proses analisis yang akan dilanjutkan dengan tahapan desain/perancangan sistem.

Untuk translasi model dari proses analisis ke proses desain dapat digambarkan berikut



THE ANALYSIS MODEL

THE DESIGN MODEL

Gambar 4 Translasi model OOA ke dalam model OOD

a. Langkah-langkah Proses OOA dalam metode OMT

1. Tentukan domain masalah
2. Bangun model objek
 - Identifikasi kelas yang relevan untuk masalah tersebut
 - Tentukan atribut dan asosiasi
 - Tentukan link objek
 - Organisasikan kelas objek dengan menggunakan pewarisan
3. Kembangkan Model Dinamis
 - Siapkan skenario
 - Tentukan event dan kembangkan penelusuran event untuk masing-masing skenario
 - Buatlah diagram aliran event
 - Kajiilah tingkah laku untuk konsistensi dan kelengkapannya.
4. Buatlah Model Fungsioanal untuk sistem tersebut
 - Identiikasikan input dan output
 - Gunakan aliran data untuk merepresentasikan transformasi aliran
 - Kembangkan msing-masing fungsi

b. Langkah-langkah Proses OOD dalam OMT

1. Lakukan Desain Sistem
 - Partissi model analisis ke dalam subsistem
 - Identifikasi konkurensi yang ditentukan oleh masalah
 - Alokasikan subsistem ke prosesor dan tugas.
 - Pilih strategi untuk manajemen data
 - Identifikasikan sumber daya globl dan mekanisme kontrol untuk mengakses

- Kajilah dan perhatikan *trade-offs*
2. Lakukan Desain Objek
 - Pilih operasi model analisis
 - Tentukan algoritma untuk masing-masing operasi
 - Pilih struktur data untuk setiap algoritma
 - Tentukan kelas internal
 - Kajilah organisasi kelas untuk mengoptimalkan akses ke data dan tingkatkan efisiensi komputasi
 - Rancang atribut kelas
 3. Implementasi mekanisme kontrol
 4. Sesuaikan struktur kelas untuk memperkuat pewarisan
 5. rancang pemesanan untuk mengimplementasikan hubungan objek asosiasi
 6. Kemas kelas-kelas dan asosiasi ke dalam modul

Keunggulan dan Kelemahan Object Oriented Technology

a. Keunggulan OMT

1. Uniformity

Penembang cukup menggunakan satu metodologi dari tahap analisis hingga perancangan. Dengan adanya perkembangan ke arah aplikasi GUI (*graphical User interface*) , OMT memungkinkan merancang user interface secara terintegrasi bersama dengan perancangan perangkat lunak sekaligus dengan perancangan basis data

2. Understandability

Kode-kode yang dihasilkan dapat diorganisasi ke dalam kelas-kels yang berhubungan dengan masalah sesungguhnya sehingga lebih mudah dipahami

3. Stability

Kode program yang dihasilkan relatif stabil sebab mendekati permasalahan sesungguhnya dilapangan

4. Reusability

Dimungkinkan penggunaan kembali kode-kode sehingga akan mempercepat waktu pengembangan perangkat lunak.

b. Kelemahan OMT

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 perancangan perangkat lunak

