Waterfall

      Metode waterfall adalah suatu proses pembuatan situs web secara terstruktur dan berurutan dimulai dari penentuan masalah, analisa kebutuhan, perancangan implementasi, untegrasi, uji coba sistem, penempatan situs web dan pemeliharaan. Pembuatan situs web dengan metode ini sangat cocok dilakukan pada situs web berskala besar karena menyangkut manajemen dan sistem yang rumit.

      Metode ini membutuhkan pendekatan sistematis dan sekuensial dalam pengembangan perangkat lunak dan biasanya disebut juga dengan classic life cycle, dimulai dari tingkat sistem dan kemajuan melalui analisis, desain, coding, testing dan pemeliharaan.

      Rekayasa dan Pemodelan Sistem/Informasi (System/Information Engineering and Modeling) Karena perangkat lunak adalah bagian dari sistem yang lebih besar, pekerjaan dimulai dari pembentukan kebutuhan-kebutuhan dari semua elemen sistem dan mengalokasikan suatu subset ke dalam pembentukan perangkat lunak. Hal ini penting, ketika perangkat lunak harus berkomunikasi dengan hardware, orang dan basis data. Rekayasa dan pemodelan sistem menekankan pada pengumpulan kebutuhan pada level sistem dengan sedikit perancangan dan analisis.

Tahap – tahap pengembangan waterfall model

1. Analisis dan definisi persyaratan

    Pelayanan,batasan, dan tujuan sistem

    ditentukan melalui konsultasi dengan

    user.

2. Perancangan sistem dan perangkat

    lunak Kegiatan ini menentukan

    arsitektur sistem secara keseluruhan

3. Implementasi dan pengujian unit

    Perancangan perangkat lunak

    direalisasikan sebagai serangkaian

    program

4. Integrasi dan pengujian sistem Unit

    program diintegrasikan atau diuji

    sebagai sistem yang lengkap untuk

    menjamin bahwa persyaratan

         sistem telah terpenuhi

5. Operasi dan pemeliharaan

Pengertian uml

      UML (Unified Modeling Language) adalah sebuah bahasa untuk menetukan, visualisasi, kontruksi, dan mendokumentasikan artifact (bagian dari informasi yang digunakan atau dihasilkan dalam suatu proses pembuatan perangkat lunak. Artifact dapat berupa model, deskripsi atau perangkat lunak) dari system perangkat lunak, seperti pada pemodelan bisnis dan system non perangkat lunak lainnya

Jenis-jenis uml

      UML mendefinisikan diagramdiagram sebagai berikut:

1.  use case diagram

2.  class diagram

3.  statechart diagram

4.  activity diagram

5.  sequence diagram

6.  collaboration diagram

7.  component diagram

8.  deployment diagram

9.  Diagram konteks

10.Diagram batang

11.Entity Relationship Diagram (ERD)

12.Diagram lingkaran

13.Diagram alir data

Use Case Diagram

  Use case diagram menggambarkan fungsionalitas yang diharapkan dari sebuah sistem.

  Yang ditekankan adalah “apa” yang diperbuat sistem, dan bukan “bagaimana”.

  Sebuah use case merepresentasikan sebuah interaksi antara aktor dengan sistem.

  Use case merupakan sebuah pekerjaan tertentu, misalnya :

◦       login ke sistem,

◦       meng-create sebuah daftar belanja,

dan sebagainya

Class Diagram

  Class adalah sebuah spesifikasi yang jika diinstansiasi akan menghasilkan sebuah objek dan merupakan inti dari pengembangan dan desain berorientasi objek.

  Class menggambarkan keadaan (atribut/properti) suatu sistem, sekaligus menawarkan layanan untuk memanipulasi keadaan tersebut(metoda/fungsi).

Statechart Diagram

  Statechart diagram menggambarkan transisi dan perubahan keadaan (dari satu state ke state lainnya) suatu objek pada sistem sebagai akibat dari stimuli yang diterima.

  Pada umumnya statechart diagram menggambarkan class tertentu (satu class dapat memiliki lebih dari satu statechart diagram).

Activity Diagram

  Activity diagrams menggambarkan berbagai alir aktivitas dalam sistem yang sedang dirancang, bagaimana masing-masing alir berawal, decision yang mungkin terjadi, dan bagaimana mereka berakhir.

  Activity diagram juga dapat menggambarkan proses paralel yang mungkin terjadi pada beberapa eksekusi.

  Sebuah aktivitas dapat direalisasikan oleh satu use case atau lebih.

  Aktivitas menggambarkan proses yang berjalan, sementara use case menggambarkan bagaimana aktor menggunakan sistem untuk melakukan aktivitas.

Sequence Diagram

  Sequence diagram biasa digunakan untuk menggambarkan skenario atau rangkaian langkah-langkah yang dilakukan sebagai respons dari sebuah event untuk menghasilkan output tertentu.

  Diawali dari apa yang men-trigger aktivitas tersebut, proses dan perubahan apa saja yang terjadi secara internal dan output apa yang dihasilkan

Collaboration Diagram

  Collaboration diagram juga menggambarkan interaksi antar objek seperti sequence diagram, tetapi lebih menekankan pada peran masing-masing objek dan bukan pada waktu penyampaian message.

Component Diagram

  Component diagram menggambarkan struktur dan hubungan antar komponen piranti lunak, termasuk ketergantungan (dependency) di antaranya.

  Komponen piranti lunak adalah modul berisi code, baik berisi source code maupun binary code, baik library maupun executable, baik yang muncul pada compile time, link time, maupun run time.

  Umumnya komponen terbentuk dari beberapa class dan/atau package, tapi dapat juga dari komponen-komponen yang lebih kecil.

Deployment Diagram

  Deployment/physical diagram menggambarkan detail bagaimana komponen di-deploy dalam infrastruktur sistem, di mana komponen akan terletak (pada mesin, server atau piranti keras apa), bagaimana kemampuan jaringan pada lokasi tersebut, spesifikasi server, dan hal-hal lain yangbersifat fisikal

     Diagram konteks adalah diagram yang terdiri dari suatu proses dan menggambarkan ruang lingkup suatu sistem. Diagram konteks merupakan level tertinggi dari DFD yang menggambarkan seluruh input ke sistem atau output dari sistem

     Diagram batang (histogram) adalah gambaran mengenai suatu distribusi frekuensi, dimana untuk setiap kelas dinyatakan dalam skala horizontal (datar) dan frekuensinya dalam skala vertical (tegak) atau sebaliknya

     Dalam rekayasa perangkat lunak, sebuah Entity-Relationship Model (ERM) merupakan abstrak dan konseptual representasi data. Entity-Relationship adalah salah satu metode pemodelan basis data yang digunakan untuk menghasilkan skema konseptual untuk jenis/model data semantik sistem.

     Diagram lingkaran adalah diagram untuk menggambarkan atau mereprentasikan data sebagai gambar lingkaran. Karena representasinya sebagai lingkaran maka data harus membentuk penjumlahan dan tiap datum ( item data ) dapat dinyatakan sebagai persen ( memiliki porsi tertentu antara 0 hingga 100 ) terhadap data.

     Diagram Alir Data (DAD) atau Data Flow Diagram (DFD) adalah suatu diagram yang menggunakan notasi-notasi untuk menggambarkan arus dari data sistem, yang penggunaannya sangat membantu untuk memahami sistem secara logika, tersruktur dan jelas. DFD merupakan alat bantu dalam menggambarkan atau menjelaskan  DFD ini sering disebut juga dengan nama Bubble chart, Bubble diagram, model proses, diagram alur kerja, atau model fungsi.