Proyek Perangkat Lunak

A11.4606

Di Susun oleh :

1. Daniel Muharom Apriyan Supratono

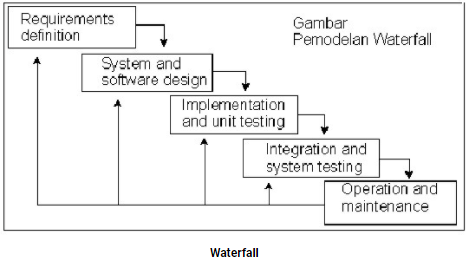
A11.2019.12102

Universitas Dian Nuswantoro

2022

3 Model Rekayasa Perangkat lunak

1. **Waterfall**



Model ini merupakan yang pertama digunakan untuk mengembangkan perangkat lunak. Model ini diperkenalkan oleh Royce pada tahun 1970.

Model ini mencakup spesifikasi persyaratan, perancangan sistem dan perangkat lunak, implementasi dan pengujian unit, integrasi dan pengujian sistem dan pemeliharaan

Tahapan metode **waterfall** :

1. Software Requiretments Analysis, yaitu pencarian kebutuhan yang difokuskan pada software tentu saja kebutuhan tersebut nantinya dapat dipahami oleh pengembang.
2. Design, yaitu pengaplikasian dari kedua kebutuhan diatas dalam bentuk sebuah arsit sebelum koding dimulai, ini menyangkut keseluruhan sistem seperti alur kerja sistem.
3. Implementasi, yaitu design yang sudah jadi diaplikasi kedalam sebuah bahasa pemrograman agar dapat dimengerti oleh komputer
4. Integrasi dan pengujian sistem, yaitu uji coba sistem yang sudah dibuat agar dapat disempurnakan kembalie)
5. Operasi dan Maintenance, yaitu pemeliharaan sistem yang sudah dilempar kepasar. Tujuan dari maintenance ini adalah agar sistem yang dibuat bisa sesuai dengan perkembangan pasar yang ada sehingga sistem yang dibuat dapat dipakai dalam jangkat waktu yang cukup panjang.

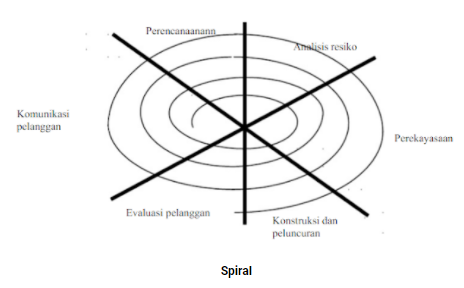
Kelebihan Waterfall :

pengerjaan proyek sistem akan dilakukan secara terstruktur karena dan mudah dikontrol.

Kekurangan Waterfall :

model ini bersifat kaku sehingga akan menyulitkan ketika melakukan perubahan.

1. **Sprial**

model proses software yang evolusioner yang merangkai sifat iteratif dari prototipe dengan cara kontrol dan aspek sistematis dari model sekuensial linier. Model ini berpotensi untuk **pengembangan** versi pertambahan software secara cepat.

**Tahapan dalam Spiral Model**

Dalam penerapan Model Spiral, terdapat lima tahapan untuk merealisasikan penggunaannya, yaitu sebagai berikut:

1. Tahap Liason

Tahap ini berhubungan dengan komunikasi antara pihak-pihak yang terlibat dalam pengembangan softaware (seperti: system analyst) dengan pelanggan (user). Tujuannya adalah memperbaiki dan mengembangan software sesuai kebutuhan dan keinginan hingga memuaskan pelanggan.

1. Tahap planning

Tahap perencanaan meliputi estimasi biaya yang digunakan, batas waktu, pengaturan jadwal, identifikasi lingkungan kerja, sumber-sumber informasi untuk melakukan iterasi (Teknik perulangan). Hasil dari tahapan ini adalah dokumen spesifikasi kebutuhan sistem dan bisnis.

1. Tahap analisis risiko

Tahap analisis reisiki berfungsi untuk mengidentifikasi resiko yang berpotensi akan terjadi dan menghasilkan solusi alternatif secara teknis dan manajemen saat strategi mitigasi (upaya untuk mengurangi resiko bencana) direncanakan dan diselesaikan.

1. Tahapan rekayasa (engineering)

Pada tahap rekayasa, beberapa kegiatan ini yang akan dilakukan, yaitu:

* Menguji, coding dan mengembangkan software
* Menginstal software
* Membuat prototype
* Mendesain dokumen
* Meringkas suatu pengujian software
* Membuat laporan atas kekurangan dari software agar segera diperbaiki

1. Tahap evaluasi

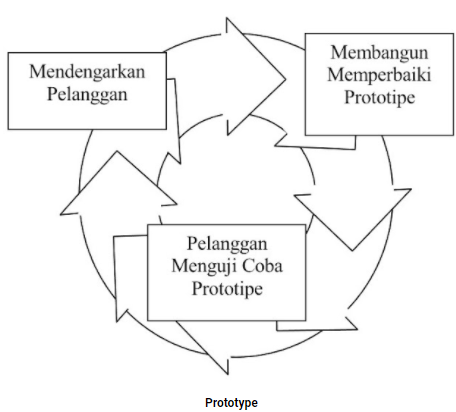
Pada tahap evaluasi, system analyst membutuhkan masukan dan tanggapan dari para user dalam mengevaluasi perangkat/produk yang diuji dan memastikan bahwa produk dibutuhkan sesuai ketentuan yang telah dibicarakan diawal dengan user. System analyst memastikan pelanggan puas dengan produk yang akan dihasilkan untuk menjawab persoalan bisnis mereka. Selain itu, system analyst harus tetap memantau resiko yang akan terjadi seperti faktor-faktor yang dapat menyebabkan cost overrun (pembengkakan biaya).

**Kelebihan dalam menggunakan model spiral :**

1. Pembangunan dan perubahan perangkat lunak yang terjadi dapat diselesaikan secara sistematis
2. Mudah dalam mengestimasi biaya karena proses pembuatan prototype yang jelas dan terencana dalam tahapan yang sistematis
3. Manajemen dan analisa risiko yang lebih cepat dan mudah
4. Mudah dalam melakukan perubahan kebutuhan dan dokumentasi
5. Produksi software bisa terjadi lebih cepat

**Kekurangan dalam menggunakan model spiral :**

1. Tidak cocok dan sulit diimplementasikan dalam projek kecil
2. Memakan waktu yang cukup lama
3. Membutuhkan best practice atau pengalaman sebelumnya karena proses yang sangat kompleks
4. Resiko dalam tahap planning cukup besar. Misalnya terjadi perbedaan dalam jadwal pengembangan dan anggaran belanja.
5. **Prototype**



metode pengembangan software yang cukup banyak digunakan. Dengan metode ini, pengembang dan pelanggan bisa saling berinteraksi selama proses pengembangan software. Hal ini tentu sangat menguntungkan dan semakin memudahkan dalam pembuatan perangkat lunak.

Metode Prototype adalah teknik pengembangan sistem yang menggunakan prototype untuk menggambarkan sistem sehingga klien atau pemilik sistem mempunyai gambaran jelas pada sistem yang akan dibangun oleh tim pengembang.

**Tahapan Dalam Metode Prototype**

**Tahap 1: Requirements Gathering and Analysis (Analisis Kebutuhan)**

Tahapan model prototype dimulai dari analisis kebutuhan. Dalam tahap ini kebutuhan sistem didefinisikan dengan rinci. Dalam prosesnya, klien dan tim developer akan bertemu untuk mendiskusikan detail sistem seperti apa yang diinginkan oleh user.

**Tahap 2: Quick Design (Desain cepat)**

Tahap kedua adalah pembuatan desain sederhana yang akan memberi gambaran singkat tentang sistem yang ingin dibuat. Tentunya berdasarkan diskusi dari langkah 1 diawal.

**Tahap 3: Build Prototype (Bangun Prototipe)**

Setelah desain cepat disetujui selanjutnya adalah pembangunan prototipe sebenarnya yang akan dijadikan rujukan tim programmer untuk pembuatan program atau aplikasi.

**Tahap 4: User Evaluation (Evaluasi Pengguna Awal)**

Di tahap ini, sistem yang telah dibuat dalam bentuk prototipe di presentasikan pada klien untuk di evaluasi. Selanjutnya klien akan memberikan komentar dan saran terhadap apa yang telah dibuat.

**Tahap 5: Refining Prototype (Memperbaiki Prototipe)**

Jika klien tidak mempunyai catatan revisi dari prototipe yang dibuat, maka tim bisa lanjut pada tahapan 6, namun jika klien mempunyai catatan untuk perbaikan sistem, maka fase 4-5 akan terus berulang sampai klien setuju dengan sistem yang akan dikembangkan.

**Tahap 6: Implement Product and Maintain (Implentasi dan Pemeliharaan)**

Pada fase akhir ini, produk akan segera dibuat oleh para programmer berdasarkan prototipe akhir, selanjutnya sistem akan diuji dan diserahkan pada klien. Selanjutnya adalah fase pemeliharaan agar sistem berjalan lancar tanpa kendala.

**Keuntungan dari Metode Prototype**

Pertama-tama penting untuk memahami metode prototipe yang paling baik digunakan ketika sistem yang diinginkan perlu memiliki banyak interaksi dengan pengguna akhir.

Saat menggunakan model jenis ini, kesalahan biasanya dapat dideteksi lebih cepat dan umpan balik pengguna yang lebih cepat tersedia untuk menghasilkan solusi yang lebih baik. Dalam metodologi ini model kerja dari sistem disediakan, pengguna mendapatkan pemahaman yang lebih baik tentang sistem yang sedang dikembangkan.

Developer bisa bekerja menentukan kebutuhan klien dengan baik, Efisiensi waktu tinggi dalam pengembangan sistem serta Lebih mudah dalam penerapannya karena klien mengetahui apa yang dibutuhkan.

**Kekurangan Metode Pembuatan Prototype**

Metode ini dapat meningkatkan kompleksitas. Rencana Anda mungkin mulai melampaui rencana awal Anda. Selain itu, Fokus pada prototipe terbatas dapat mengalihkan pengembang dari analisis lengkap proyek dengan benar. Namun itulah mengapa ada tahap penyempurnaan.

Klien terus menerus menambah requirement dari sistem, pengen dibuatkan yang seperti inilah seperti itulah, sehingga menambah kompleksitas pembuatan sistem. Sistem akan terhambat jika komunikasi kedua belah pihak tidak berjalan secara efektif.