

TUGAS
PROYEK PERANGKAT LUNAK
MODEL PENGEMBANGAN PERANGKAT LUNAK



Disusun oleh:
FIRMAN ASDIANSYAH
A11.2019.11862

FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO
SEMARANG
2022

DAFTAR ISI

DAFTAR ISI.....	2
3 Model Pengembangan Perangkat Lunak	3
1. Waterfall	3
Tahapan – tahapan Pengembangan Waterfall Development.....	3
Kelebihan Waterfall Development Model	4
Kekurangan Waterfall Development Model	4
2. Model Prototype.....	5
Teknik – teknik Prototyping	5
4 langkah yang menjadi karakteristik dalam proses pengembangan pada metode prototype	5
Tahapan – tahapan Proses Pengembangan dalam Model Prototype	5
Kelebihan Model Prototype	7
Kekurangan Model Prototype	7
3. Model Rapid Application Development (RAD).....	7
Tahapan – tahapan Proses Pengembangan dalam Model Rapid Application Development (RAD) ...	7
Kelebihan Model RAD	8
Kekurangan Model RAD	8
REFERENSI	9

3 Model Pengembangan Perangkat Lunak

Proses Pengembangan Perangkat Lunak (Software Development Process) adalah suatu penerapan struktur pada pengembangan suatu Perangkat Lunak (Software), yang bertujuan untuk mengembangkan sistem dan memberikan panduan untuk menyelesaikan proyek pengembangan sistem melalui tahapan-tahapan tertentu. Dalam prosesnya, terdapat beberapa paradigma model pengembangan sistem perangkat lunak, 3 diantaranya :

1. Waterfall

Model Sekuensial Linier atau sering disebut Model Pengembangan Waterfall, merupakan paradigma model pengembangan perangkat lunak 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 tahapan analisis, desain, kode, pengujian, dan pemeliharaan.

Tahapan – tahapan Pengembangan Waterfall Development

- **Rekayasa dan pemodelan sistem/informasi**
Langkah pertama dimulai dengan membangun keseluruhan elemen sistem dan memilah bagian-bagian mana yang akan dijadikan bahan pengembangan perangkat lunak, dengan memperhatikan hubungannya dengan Hardware, User, dan Database.
- **Analisis kebutuhan perangkat lunak**
Pada proses ini, dilakukan penganalisaan dan pengumpulan kebutuhan sistem yang meliputi Domain informasi, fungsi yang dibutuhkan unjuk kerja/performansi dan antarmuka. Hasil penganalisaan dan pengumpulan tersebut didokumentasikan dan diperlihatkan kembali kepada pelanggan.
- **Desain**
Pada proses Desain, dilakukan penerjemahan syarat kebutuhan sebuah perancangan perangkat lunak yang dapat diperkirakan sebelum dibuatnya proses pengkodean (coding). Proses ini berfokus pada struktur data, arsitektur perangkat lunak, representasi interface, dan detail algoritma prosedural.
- **Pengkodean**
Pengkodean merupakan proses menterjemahkan perancangan desain ke bentuk yang dapat dimengerti oleh mesin, dengan menggunakan bahasa pemrograman.
- **Pengujian**
Setelah Proses Pengkodean selesai, dilanjutkan dengan proses pengujian pada program perangkat lunak, baik Pengujian logika internal, maupun Pengujian

eksternal fungsional untuk memeriksa segala kemungkinan terjadinya kesalahan dan memeriksa apakah hasil dari pengembangan tersebut sesuai dengan hasil yang diinginkan.

- **Pemeliharaan**

Proses Pemeliharaan merupakan bagian paling akhir dari siklus pengembangan dan dilakukan setelah perangkat lunak dipergunakan. Kegiatan yang dilakukan pada proses pemeliharaan antara lain :

- **Corrective Maintenance**
yaitu mengoreksi apabila terdapat kesalahan pada perangkat lunak, yang baru terdeteksi pada saat perangkat lunak dipergunakan.
- **Adaptive Maintenance**
yaitu dilakukannya penyesuaian/perubahan sesuai dengan lingkungan yang baru, misalnya hardware, periperal, sistem operasi baru, atau sebagai tuntutan atas perkembangan sistem komputer, misalnya penambahan driver, dll.
- **Perfektive Maintenance**
Bila perangkat lunak sukses dipergunakan oleh pemakai. Pemeliharaan ditujukan untuk menambah kemampuannya seperti memberikan fungsi-fungsi tambahan, peningkatan kinerja dan sebagainya.

Kelebihan Waterfall Development Model

- Tahapan proses pengembangannya tetap (pasti), mudah diaplikasikan, dan prosesnya teratur.
- Cocok digunakan untuk produk software/program yang sudah jelas kebutuhannya di awal, sehingga minim kesalahannya.
- Software yang dikembangkan dengan metode ini biasanya menghasilkan kualitas yang baik.
- Dokumen pengembangan sistem sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya.

Kekurangan Waterfall Development Model

- Proyek yang sebenarnya jarang mengikuti alur sekuensial seperti diusulkan, sehingga perubahan yang terjadi dapat menyebabkan hasil yang sudah didapatkan tim pengembang harus diubah kembali/iterasi sering menyebabkan masalah baru.
- Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakukan pada tahap awal proses.
- Sulit untuk mengalami perubahan kebutuhan yang diinginkan oleh customer/pelanggan.

- Pelanggan harus sabar untuk menanti produk selesai, karena dikerjakan tahap per tahap, dan proses pengerjaannya akan berlanjut ke setiap tahapan bila tahap sebelumnya sudah benar-benar selesai.
- Perubahan ditengah-tengah pengerjaan produk akan membuat bingung tim pengembang yang sedang membuat produk.
- Adanya waktu kosong (menganggur) bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.

2. Model Prototype

Metode Prototype merupakan suatu paradigma baru dalam metode pengembangan perangkat lunak dimana metode ini tidak hanya sekedar evolusi dalam dunia pengembangan perangkat lunak, tetapi juga merevolusi metode pengembangan perangkat lunak yang lama yaitu sistem sekuensial yang biasa dikenal dengan nama SDLC atau waterfall development model.

Dalam Model Prototype, prototype dari perangkat lunak yang dihasilkan kemudian dipresentasikan kepada pelanggan, dan pelanggan tersebut diberikan kesempatan untuk memberikan masukan sehingga perangkat lunak yang dihasilkan nantinya betul-betul sesuai dengan keinginan dan kebutuhan pelanggan.

Perubahan dan presentasi prototype dapat dilakukan berkali-kali sampai dicapai kesepakatan bentuk dari perangkat lunak yang akan dikembangkan.

Teknik – teknik Prototyping

- Perancangan Model
- Perancangan Dialog
- Simulasi

4 langkah yang menjadi karakteristik dalam proses pengembangan pada metode prototype

- Pemilihan fungsi
- Penyusunan Sistem Informasi
- Evaluasi
- Penggunaan Selanjutnya

Metode ini menyajikan gambaran yang lengkap dari suatu sistem perangkat lunak, terdiri atas model kertas, model kerja dan program. Pihak pengembang akan melakukan identifikasi kebutuhan pemakai, menganalisa sistem dan melakukan studi kelayakan serta studi terhadap kebutuhan pemakai, meliputi model interface, teknik prosedural dan teknologi yang akan dimanfaatkan.

Tahapan – tahapan Proses Pengembangan dalam Model Prototype

- Pengumpulan kebutuhan

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

- **Membangun prototyping**

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

- **Evaluasi prototyping**

Evaluasi ini dilakukan oleh pelanggan, apakah prototyping yang sudah dibangun sudah sesuai dengan keinginan pelanggan atau belum. Jika sudah sesuai, maka langkah selanjutnya akan diambil. Namun jika tidak, prototyping direvisi dengan mengulang langkah-langkah sebelumnya.

- **Mengkodekan sistem**

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

- **Menguji sistem**

Setelah sistem sudah menjadi suatu perangkat lunak yang siap pakai, kemudian dilakukan proses Pengujian. Pengujian ini dilakukan dengan White Box, Black Box, Basis Path, pengujian arsitektur, dll.

- **Evaluasi Sistem**

Pelanggan mengevaluasi apakah perangkat lunak yang sudah jadi sudah sesuai dengan yang diharapkan. Jika ya, maka proses akan dilanjutkan ke tahap selanjutnya, namun jika perangkat lunak yang sudah jadi tidak/belum sesuai dengan apa yang diharapkan, maka tahapan sebelumnya akan diulang.

- **Menggunakan sistem**

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

Model Prototyping ini sangat sesuai diterapkan untuk kondisi yang beresiko tinggi di mana masalah-masalah tidak terstruktur dengan baik, terdapat fluktuasi kebutuhan pemakai yang berubah dari waktu ke waktu atau yang tidak terduga, bila interaksi dengan pemakai menjadi syarat mutlak dan waktu yang tersedia sangat terbatas sehingga butuh penyelesaian yang segera. Model ini juga dapat berjalan dengan maksimal pada situasi di mana sistem yang diharapkan adalah yang inovatif dan mutakhir sementara tahap penggunaan sistemnya relatif singkat.

Kelebihan Model Prototype

- Pelanggan berpartisipasi aktif dalam pengembangan sistem, sehingga hasil produk pengembangan akan semakin mudah disesuaikan dengan keinginan dan kebutuhan pelanggan.
- Penentuan kebutuhan lebih mudah diwujudkan.
- Mempersingkat waktu pengembangan produk perangkat lunak.
- Adanya komunikasi yang baik antara pengembang dan pelanggan.
- Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan.
- Lebih menghemat waktu dalam pengembangan sistem.
- Penerapan menjadi lebih mudah karena pelanggan mengetahui apa yang diharapkannya.

Kekurangan Model Prototype

- Proses analisis dan perancangan terlalu singkat.
- Biasanya kurang fleksibel dalam menghadapi perubahan.
- Walaupun pemakai melihat berbagai perbaikan dari setiap versi prototype, tetapi pemakai mungkin tidak menyadari bahwa versi tersebut dibuat tanpa memperhatikan kualitas dan pemeliharaan jangka panjang.
- Pengembang kadang-kadang membuat kompromi implementasi dengan menggunakan sistem operasi yang tidak relevan dan algoritma yang tidak efisien.

3. Model Rapid Application Development (RAD)

Rapid Application Development (RAD) adalah sebuah model proses pengembangan perangkat lunak sekuensial linier yang menekankan siklus pengembangan yang sangat pendek (kira-kira 60 sampai 90 hari). Model RAD ini merupakan sebuah adaptasi “kecepatan tinggi” dari model sekuensial linier dimana pengembangan cepat dicapai dengan menggunakan pendekatan konstruksi berbasis komponen.

Tahapan – tahapan Proses Pengembangan dalam Model Rapid Application Development (RAD)

• Business Modeling

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 ?

- **Data Modeling**

Aliran informasi yang didefinisikan sebagai bagian dari fase bussiness modeling disaring ke dalam serangkaian objek data yang dibutuhkan untuk menopang bisnis tersebut. Karakteristik (atribut) masing-masing objek diidentifikasi dan hubungan antar objek-objek tersebut didefinisikan.

- **Proses Modeling**

Aliran informasi yang didefinisikan di dalam fase data modeling ditransformasikan untuk mencapai aliran informasi yang perlu bagi implementasi sebuah fungsi bisnis. Gambaran pemrosesan diciptakan untuk menambah, memodifikasi, menghapus, atau mendapatkan kembali sebuah objek data.

- **Aplication Generation**

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

- **Testing dan Turnover**

Karena proses RAD menekankan pada pemakaian kembali, banyak komponen program telah diuji. Hal ini mengurangi keseluruhan waktu pengujian. Tetapi komponen baru harus diuji dan semua interface harus dilatih secara penuh.

Kelebihan Model RAD

- Lebih efektif dari Pengembangan Model waterfall/sequential linear dalam menghasilkan sistem yang memenuhi kebutuhan langsung dari pelanggan.
- Cocok untuk proyek yang memerlukan waktu yang singkat.
- Model RAD mengikuti tahap pengembangan sistem seperti pada umumnya, tetapi mempunyai kemampuan untuk menggunakan kembali komponen yang ada sehingga pengembang tidak perlu membuatnya dari awal lagi sehingga waktu pengembangan menjadi lebih singkat dan efisien.

Kekurangan Model RAD

- Model RAD menuntut pengembangan dan pelanggan memiliki komitmen di dalam aktivitas rapid-fire yang diperlukan untuk melengkapi sebuah sistem, di dalam kerangka waktu yang sangat diperpendek. Jika komitmen tersebut tidak ada, proyek RAD akan gagal.
- Tidak semua aplikasi sesuai untuk RAD, bila system tidak dapat dimodulkan dengan teratur, pembangunan komponen penting pada RAD akan menjadi sangat bermasalah.
- RAD tidak cocok digunakan untuk sistem yang mempunyai resiko teknik yang tinggi.
- Membutuhkan Tenaga kerja yang banyak untuk menyelesaikan sebuah proyek dalam skala besar.
- Jika ada perubahan di tengah-tengah pengerjaan maka harus membuat kontrak baru antara pengembang dan pelanggan.

REFERENSI

<http://roysarimilda.wordpress.com/2012/05/08/macam-macam-model-proses-rpl-dan-penyelesaian-kasus/>

<http://komandankempong.blogspot.com/2011/09/model-proses-rekayasa-perangkat-lunak.html>

http://id.wikipedia.org/wiki/Proses_pengembangan_perangkat_lunak