Candra Abdullah Khusni

A11.2019.12278

TUGAS 1 PPL

3 Model Software Engginering

1. Model Inkremental adalah proses pengembangan perangkat lunak di mana persyaratan dibagi menjadi beberapa modul mandiri dari siklus pengembangan perangkat lunak. Dalam model ini, setiap modul melewati fase persyaratan, desain, implementasi, dan pengujian. Setiap rilis modul berikutnya menambahkan fungsi ke rilis sebelumnya. Proses berlanjut

sampai sistem lengkap tercapai.



Analisis Fase

* Analisis kebutuhan: Pada fase pertama model inkremental, keahlian analisis produk mengidentifikasi persyaratan. Dan kebutuhan fungsional sistem dipahami oleh tim analisis kebutuhan. Untuk mengembangkan perangkat lunak di bawah model inkremental, fase ini memainkan peran penting.
* Desain & Pengembangan: Pada fase model Incremental SDLC ini, desain fungsionalitas sistem dan metode pengembangan selesai dengan sukses. Ketika perangkat lunak mengembangkan kepraktisan baru, model inkremental menggunakan fase gaya dan pengembangan.
* Pengujian: Pada model inkremental, fase pengujian memeriksa kinerja setiap fungsi yang ada serta fungsionalitas tambahan. Pada fase pengujian, berbagai metode digunakan untuk menguji perilaku setiap tugas.
* Implementasi: Fase implementasi memungkinkan fase pengkodean dari sistem pengembangan. Ini melibatkan pengkodean akhir yang dirancang dalam fase perancangan dan pengembangan dan menguji fungsionalitas dalam fase pengujian. Setelah menyelesaikan fase ini, jumlah produk yang bekerja ditingkatkan dan ditingkatkan hingga produk sistem akhir

Kelebihan :

* Kesalahan mudah dikenali
* Lebih mudah untuk menguji dan mendebug
* Lebih fleksibel
* Sederhana untuk mengelola risiko karena ditangani
* Klien mendapat fungsional penting lebih awal

Kekurangan :

* Perlu perancangan yang baik
* Total biaya tinggi
* Diperlukan antarmuka modul yang terdefinisi dengan baik

1. Model Prototyping adalah model pengembangan perangkat lunak di mana prototipe dibangun, diuji, dan dikerjakan ulang sampai prototipe yang dapat diterima tercapai. Ini juga menciptakan basis untuk menghasilkan sistem atau perangkat lunak akhir. Ini bekerja paling baik dalam skenario di mana persyaratan proyek tidak diketahui secara rinci. Ini adalah metode iteratif, coba-coba yang terjadi antara pengembang dan klien.



Analisis Fase :

* Pengumpulan dan alanisis persyaratan

Model prototyping dimulai dengan analisis kebutuhan. Pada fase ini, kebutuhan sistem didefinisikan secara rinci. Selama proses, pengguna sistem diwawancarai untuk mengetahui apa harapan mereka dari sistem.

* Desin cepat

Tahap kedua adalah desain pendahuluan atau quick design. Pada tahap ini dibuat desain sistem yang sederhana. Namun, itu bukan desain yang lengkap. Ini memberikan gambaran singkat tentang sistem kepada pengguna. Desain cepat membantu dalam mengembangkan prototipe.

* Bangun prototipie

prototipe sebenarnya dirancang berdasarkan informasi yang dikumpulkan dari desain cepat. Ini adalah model kerja kecil dari sistem yang diperlukan.

* Evaluasi pengguna awal

sistem yang diusulkan disajikan kepada klien untuk evaluasi awal. Ini membantu untuk mengetahui kekuatan dan kelemahan model kerja. Komentar dan saran dikumpulkan dari pelanggan dan diberikan kepada pengembang.

* Memperbaiki prototipr

Perlu memperbaiki prototipe sesuai dengan umpan balik dan saran pengguna.

Fase ini tidak akan berakhir sampai semua persyaratan yang ditentukan oleh pengguna terpenuhi. Setelah pengguna puas dengan prototipe yang dikembangkan, sistem final dikembangkan berdasarkan prototipe akhir yang disetujui.

* Menerapkan produk dan memlihara

Setelah sistem final dikembangkan berdasarkan prototipe akhir, sistem tersebut diuji secara menyeluruh dan disebarkan ke produksi. Sistem menjalani perawatan rutin untuk meminimalkan waktu henti dan mencegah kegagalan skala besar.

Kelebihan :

* Kurangi risiko persyaratan pengguna yang salah
* Baik di mana persyaratan berubah/tidak terikat
* Proses terlihat teratur membantu manajemen
* Mengurangi biaya Pemeliharaan.
* Kesalahan dapat dideteksi lebih awal karena sistem dibuat berdampingan.

Kekurangan :

* Prototipe yang tidak stabil/diimplementasikan dengan buruk sering kali menjadi produk akhir.
* Memerlukan kolaborasi pelanggan yang luas
* Sulit untuk mengetahui berapa lama proyek akan berlangsung.
* Mudah untuk kembali ke kode dan memperbaikinya tanpa analisis persyaratan, desain, evaluasi pelanggan, dan umpan balik yang tepat.

1. Arti Agile adalah cepat atau serbaguna." Model proses Agile " mengacu pada pendekatan pengembangan perangkat lunak berdasarkan pengembangan berulang. Metode tangkas memecah tugas menjadi iterasi yang lebih kecil, atau bagian tidak secara langsung melibatkan perencanaan jangka panjang. Lingkup proyek dan persyaratan ditetapkan pada awal proses pengembangan.



Fase model Agile :

* Pengumpulan persyaratan: Pada fase ini, Anda harus menentukan persyaratan. Anda harus menjelaskan peluang bisnis dan merencanakan waktu dan upaya yang diperlukan untuk membangun proyek. Berdasarkan informasi ini, Anda dapat mengevaluasi kelayakan teknis dan ekonomi.
* Rancang persyaratan: Ketika Anda telah mengidentifikasi proyek, bekerjalah dengan pemangku kepentingan untuk menentukan persyaratan. Anda dapat menggunakan diagram alur pengguna atau diagram UML tingkat tinggi untuk menunjukkan kerja fitur-fitur baru dan menunjukkan bagaimana hal itu akan diterapkan pada sistem Anda yang sudah ada.
* Konstruksi/ iterasi: Ketika tim mendefinisikan persyaratan, pekerjaan dimulai. Desainer dan pengembang mulai mengerjakan proyek mereka, yang bertujuan untuk menyebarkan produk yang berfungsi. Produk akan mengalami berbagai tahap peningkatan, sehingga mencakup fungsionalitas yang sederhana dan minimal.
* Pengujian: Pada fase ini, tim Quality Assurance memeriksa kinerja produk dan mencari bug.
* Deployment: Pada fase ini, tim mengeluarkan produk untuk lingkungan kerja pengguna.
* Umpan balik: Setelah merilis produk, langkah terakhir adalah umpan balik. Dalam hal ini, tim menerima umpan balik tentang produk dan bekerja melalui umpan balik.

Kelebihan :

* Pengiriman Sering
* Komunikasi tatap muka dengan klien.
* Desain yang efisien dan memenuhi kebutuhan bisnis.
* Setiap perubahan dapat diterima.
* Mengurangi total waktu pengembangan.

Kekurangan :

* Karena kurangnya dokumen formal, hal itu menciptakan kebingungan dan keputusan penting yang diambil di berbagai fase dapat disalahartikan kapan saja oleh anggota tim yang berbeda.
* Karena kurangnya dokumentasi yang tepat, setelah proyek selesai dan pengembang dialokasikan ke proyek lain, pemeliharaan proyek yang sudah selesai dapat menjadi kesulitan.