

PRAKTIKUM

Analisis dan Perancangan Sistem

Teknik Informatika Kelas G

Nama Mahasiswa

Muhammad Rizqon Maulana (215150201111022)

Dosen:

Denny Sagita Rusdianto, S.Kom., M.Kom.



Program Studi Teknik Informatika

Jurusan Teknik Informatika

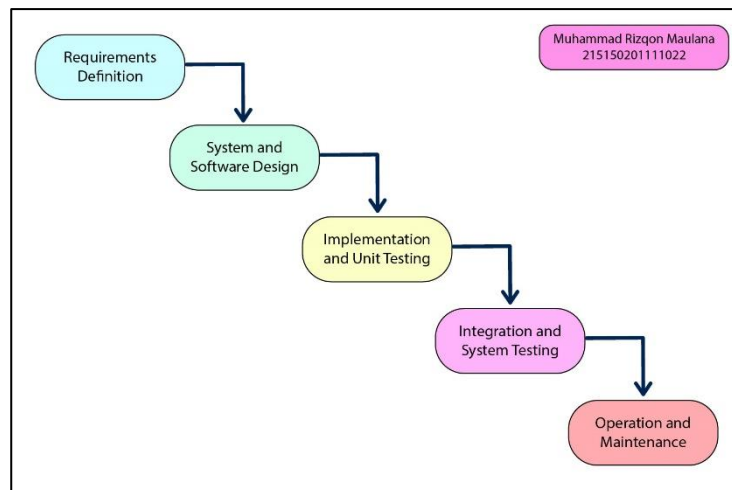
Universitas Brawijaya

2023

SOFTWARE DEVELOPMENT LIFE CYCLE

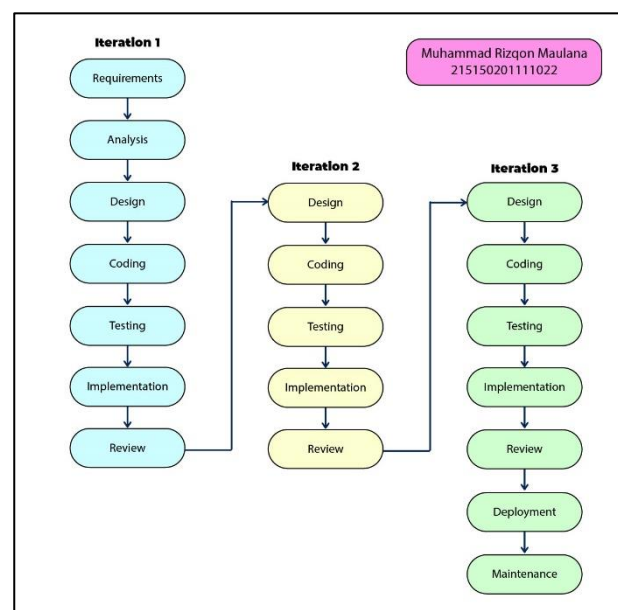
1. Sebutkan dan gambarkan 5 jenis SDLC yang anda ketahui.

a) Waterfall Model



Model SDLC ini adalah salah satu model tertua dan tersingkat dalam penerapannya. Dalam sistem ini, setiap tahap harus diselesaikan terlebih dahulu sebelum memulai tahap berikutnya. Setiap fase memiliki rencana kecil dan menurun ke fase berikutnya. Maka dari itu, fase ini dikatakan waterfall atau air terjun karena tiap fase memiliki ‘turunan’ kecil lainnya.

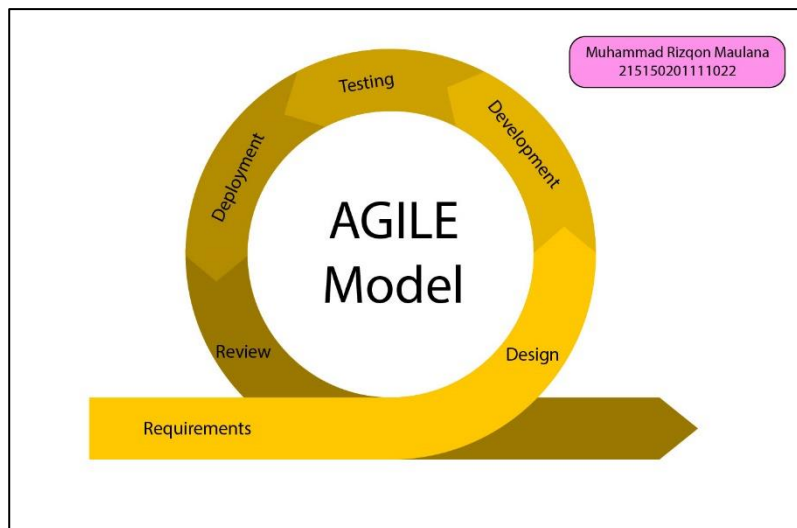
b) Iterative Model



Dalam model ini, pengembangan perangkat lunak dilakukan secara bertahap, dimulai dari tahap perencanaan, desain, implementasi, dan pengujian, kemudian dilakukan evaluasi

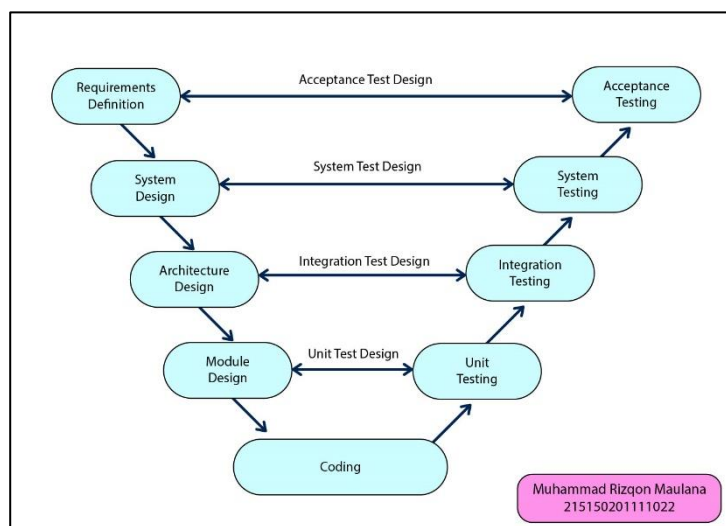
hasil dari tahap-tahap tersebut, dan proses pengembangan diteruskan kembali dengan perbaikan dan pengembangan tambahan yang diperlukan. Model ini memungkinkan pengembang perangkat lunak untuk lebih fleksibel dan adaptif terhadap perubahan kebutuhan dan spesifikasi pengguna, serta meminimalkan risiko kegagalan proyek pengembangan perangkat lunak.

c) Agile Model



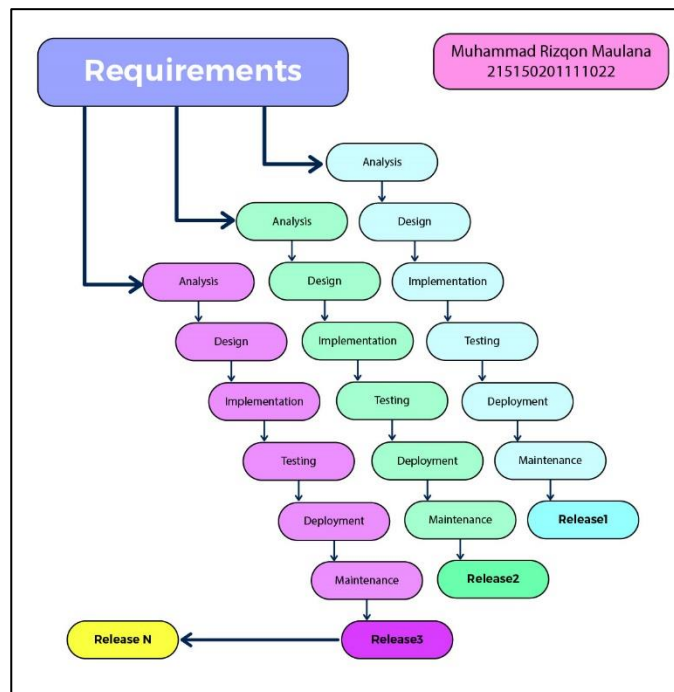
Suatu pendekatan dalam pengembangan perangkat lunak yang menekankan pada kolaborasi tim, pengiriman produk secara bertahap, dan responsif terhadap perubahan permintaan pelanggan. Dalam Agile Model, pengembangan produk dilakukan secara iteratif dan inkremental dengan siklus pengembangan yang terdiri dari perencanaan, analisis, desain, pengkodean, pengujian, dan evaluasi. Dalam Agile Model, pelanggan dan tim pengembang bekerja sama untuk menentukan kebutuhan dan prioritas, sehingga dapat memberikan nilai tambah pada setiap iterasi pengembangan produk. Model ini memungkinkan pengembang untuk lebih responsif terhadap perubahan kebutuhan dan keinginan pelanggan serta memaksimalkan nilai bisnis dan kepuasan pelanggan.

d) V-Model



Suatu model pengembangan perangkat lunak yang merupakan perluasan dari model waterfall. Model ini terdiri dari dua belahan, yaitu pengembangan dan pengujian. Pada tahap pengembangan, setiap tahap memiliki pasangan pengujian yang sesuai. Setiap tahap pengujian mencakup pengujian unit, integrasi, dan pengujian sistem. Pengujian dilakukan secara paralel dengan pengembangan, dan hasil pengujian menjadi acuan dalam mengevaluasi dan memastikan kualitas perangkat lunak. Hal ini membuat model ini sangat berorientasi pada kualitas, karena setiap tahap pengembangan memiliki tahap pengujian yang selaras dengannya. Model ini sangat cocok untuk proyek besar dengan jadwal yang ketat karena memberikan kualitas yang lebih baik dan meminimalkan kesalahan pada tahap pengujian.

e) Incremental Model



suatu model pengembangan perangkat lunak dimana pengembangan dilakukan secara bertahap dan terus-menerus, dimulai dari pengembangan modul atau bagian-bagian kecil dari perangkat lunak yang kemudian diintegrasikan satu sama lain hingga menjadi satu produk yang lengkap. Setiap tahap pengembangan atau iterasi biasanya menghasilkan produk yang siap untuk diuji coba dan digunakan, sehingga memberikan keuntungan untuk mengevaluasi progres dan menerima masukan dari pengguna. Model ini sangat cocok untuk proyek yang kompleks dan tidak jelas, karena memungkinkan perubahan dan penyesuaian kebutuhan selama pengembangan berlangsung.

2. Tuliskan kelebihan dan kelemahan pada masing-masing SDLC.

a) Waterfall Model

Kelebihan Waterfall Model:

- Mudah Diikuti: Waterfall model merupakan salah satu model SDLC yang mudah untuk diikuti karena mempunyai alur yang linear dan sekuensial.
- Menghasilkan Perangkat Lunak yang Dapat Diprediksi: Waterfall model menghasilkan perangkat lunak yang dapat diprediksi karena setiap tahap harus diselesaikan sebelum memulai tahap berikutnya.
- Mudah Dilacak: Karena setiap tahap memiliki output yang jelas dan harus diselesaikan sebelum memulai tahap berikutnya, maka proyek pengembangan perangkat lunak dengan menggunakan waterfall model mudah dilacak.

Kekurangan Waterfall Model:

- Kurang Fleksibel: Waterfall model kurang cocok untuk proyek perangkat lunak yang kompleks dan membutuhkan fleksibilitas dan adaptasi yang tinggi.
- Kesulitan dalam Menangani Perubahan Kebutuhan: Karena setiap tahap harus diselesaikan sebelum memulai tahap berikutnya, maka waterfall model kesulitan dalam menangani perubahan kebutuhan atau perubahan spesifikasi.
- Keterlambatan Identifikasi Kesalahan: Karena pengujian dilakukan setelah tahap implementasi selesai, maka identifikasi kesalahan atau bug terlambat ditemukan dan membutuhkan biaya yang lebih tinggi untuk memperbaikinya.

b) Iterative Model

Kelebihan Iterative Model:

- Fleksibel dan Adaptif: Iterative Model memungkinkan pengembang perangkat lunak untuk lebih fleksibel dan adaptif terhadap perubahan kebutuhan dan spesifikasi pengguna, karena setiap tahap dilakukan secara bertahap dengan evaluasi dan perbaikan di antara setiap tahap.
- Identifikasi Kesalahan Dini: Model ini memungkinkan pengembang untuk mengidentifikasi kesalahan atau bug lebih awal, sehingga perbaikan dapat dilakukan sebelum tahap pengujian.
- Meningkatkan Kepuasan Pengguna: Karena pengembang dapat melakukan perbaikan dan pengembangan tambahan pada setiap iterasi, maka kemungkinan hasil akhir perangkat lunak lebih sesuai dengan kebutuhan dan harapan pengguna.

Kekurangan Iterative Model:

- Memerlukan Biaya Lebih: Karena model ini melibatkan banyak iterasi dan evaluasi, maka pengembangan perangkat lunak menggunakan Iterative Model dapat memerlukan biaya yang lebih tinggi.
- Memerlukan Waktu Lebih Lama: Karena proses iterasi dan evaluasi dilakukan secara berulang-ulang, maka pengembangan perangkat lunak menggunakan Iterative Model dapat memerlukan waktu yang lebih lama dibandingkan dengan model SDLC lainnya.
- Memerlukan Tim yang Terampil: Model ini memerlukan tim yang terampil dan

berpengalaman untuk dapat melakukan evaluasi dan perbaikan secara efektif pada setiap iterasi.

c) Agile Model

Kelebihan Agile Model:

- Responsif terhadap perubahan: Agile Model memungkinkan pengembang untuk lebih responsif terhadap perubahan kebutuhan dan keinginan pelanggan, sehingga memungkinkan produk lebih sesuai dengan kebutuhan pelanggan dan nilai bisnis yang diharapkan.
- Kualitas Produk yang lebih baik: Dalam Agile Model, pengujian dan evaluasi produk dilakukan secara terus-menerus pada setiap iterasi, sehingga memungkinkan pengembang untuk mengidentifikasi masalah lebih awal dan memperbaikinya sebelum produk dirilis, sehingga meningkatkan kualitas produk.
- Kolaborasi Tim: Model ini mendorong kolaborasi antara pelanggan dan tim pengembang, sehingga memungkinkan pengembang untuk memahami kebutuhan pelanggan dan memberikan nilai tambah pada setiap iterasi pengembangan produk.

Kekurangan Agile Model:

- Memerlukan Tim yang Terampil: Agile Model memerlukan tim yang terampil dan berpengalaman, karena pengembangan produk dilakukan secara cepat dan efisien, sehingga memerlukan kemampuan yang tinggi dalam pengembangan perangkat lunak.
- Memerlukan Komunikasi yang Efektif: Model ini memerlukan komunikasi yang efektif dan transparan antara pelanggan dan tim pengembang, karena pengembangan produk dilakukan secara berulang-ulang dalam waktu yang relatif singkat.
- Tidak Cocok untuk Proyek Besar: Model ini lebih cocok untuk proyek kecil hingga menengah, karena jika digunakan untuk proyek besar, pengembangan produk bisa menjadi tidak terstruktur dan sulit dikelola.

d) V-Model

Kelebihan V-Model:

- Fokus pada Kualitas: V-Model sangat fokus pada kualitas karena setiap tahap pengembangan memiliki tahap pengujian yang sesuai, dan hasil pengujian menjadi acuan dalam mengevaluasi dan memastikan kualitas perangkat lunak.
- Meminimalkan Kesalahan: Dalam V-Model, pengujian dilakukan secara paralel dengan pengembangan, sehingga kesalahan bisa dideteksi lebih awal dan diminimalkan sebelum produk dirilis.
- Mudah Dipahami: V-Model relatif mudah dipahami dan diterapkan, sehingga cocok untuk proyek yang memiliki jadwal yang ketat.

Kekurangan V-Model:

- Tidak Fleksibel: V-Model kurang fleksibel dalam menghadapi perubahan kebutuhan, karena pengujian dilakukan secara berurutan dan tergantung pada tahap

pengembangan sebelumnya.

- Biaya yang Mahal: V-Model memerlukan biaya yang mahal untuk melakukan pengujian yang intensif dan melibatkan banyak sumber daya.
- Tidak Cocok untuk Proyek Kecil: V-Model lebih cocok untuk proyek besar yang memiliki jadwal yang ketat, sehingga kurang cocok untuk proyek kecil yang membutuhkan fleksibilitas dalam pengembangan.

e) Incremental Model

Kelebihan Incremental Model:

- Fleksibel: Incremental Model sangat fleksibel dalam menghadapi perubahan kebutuhan, karena pengembangan dilakukan secara bertahap dan terus-menerus. Hal ini memungkinkan tim pengembang untuk menyesuaikan kebutuhan dan mengubah arah pengembangan ketika diperlukan.
- Meminimalkan Resiko: Dalam Incremental Model, setiap tahap pengembangan menghasilkan produk yang siap digunakan. Ini memungkinkan pengguna untuk melakukan pengujian dan memberikan umpan balik, sehingga meminimalkan risiko kesalahan pada tahap pengembangan berikutnya.
- Mempercepat Time-to-Market: Incremental Model memungkinkan produk yang siap digunakan untuk dirilis pada setiap tahap pengembangan, sehingga mempercepat waktu pengiriman dan pemasaran.

Kekurangan Incremental Model:

- Memerlukan Rencana yang Baik: Incremental Model memerlukan perencanaan yang baik untuk menghindari potensi kegagalan dan kelemahan pada tahap pengembangan. Setiap tahap pengembangan harus direncanakan dengan baik agar dapat menghasilkan produk yang dapat digunakan dengan baik.
- Kompleksitas: Incremental Model dapat menjadi kompleks karena terdiri dari beberapa tahap pengembangan, yang masing-masing memerlukan pengujian dan validasi. Ini memerlukan pengelolaan proyek yang baik dan disiplin yang kuat untuk menghindari kegagalan.
- Biaya yang Mahal: Incremental Model memerlukan biaya yang mahal karena pengembangan dilakukan secara bertahap dan terus-menerus, dan memerlukan pengujian dan validasi setiap tahap pengembangan.