**Bayu Wicaksono**

**72130035**

1. **Model Konvensional**

Dalam Bab 1, proses didefinisikan sebagai kumpulan aktivitas kerja, tindakan, dan

tugas yang dilakukan ketika beberapa produk kerja yang akan dibuat. Masing-masing

kegiatan, tindakan, dan tugas berada dalam kerangka atau model yang mendefinisikan mereka terkait dengan proses dan dengan satu sama lain.

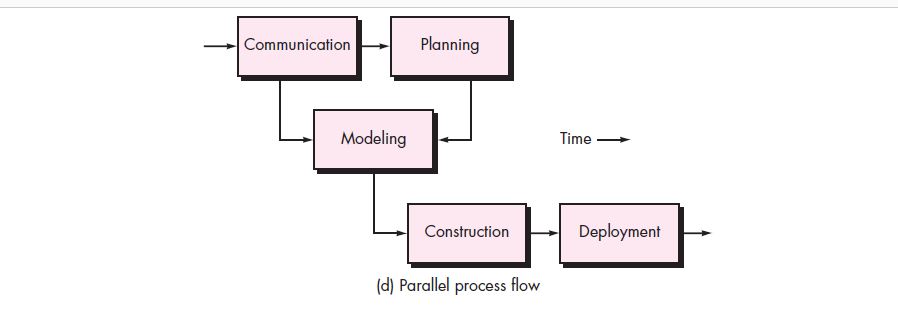
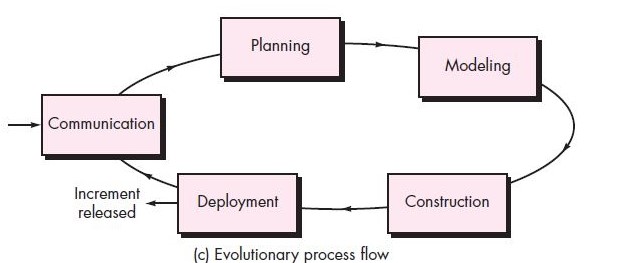
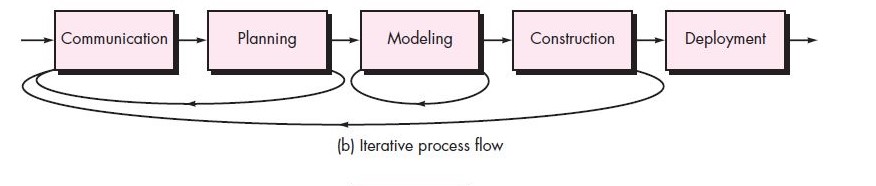
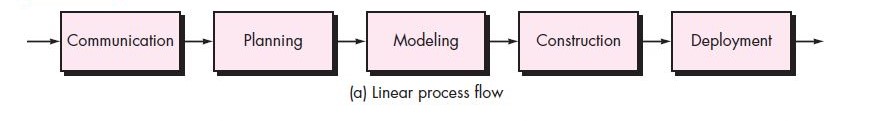
Proses perangkat lunak diwakili skematik pada Gambar 2.1. Mengacu pada Angka, setiap aktivitas kerangka dihuni oleh serangkaian tindakan rekayasa perangkat lunak.

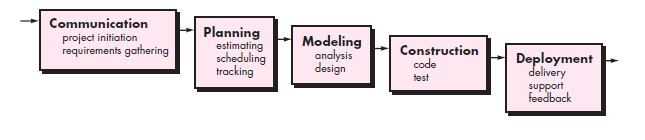
Setiap tindakan rekayasa perangkat lunak didefinisikan oleh satu set tugas yang mengidentifikasi pekerjaan tugas-tugas yang harus diselesaikan, produk pekerjaan yang akan diproduksi, kualitas poin jaminan yang akan diperlukan, dan tonggak yang akan digunakan untuk menunjukkan kemajuan.

kerangka proses generik untuk rekayasa perangkat lunak mendefinisikan lima kerangka kegiatan-komunikasi, perencanaan, pemodelan, konstruksi, dan penyebaran. Selain itu, satu set payung kegiatan-proyek pelacakan dan kontrol, manajemen risiko, jaminan kualitas, manajemen konfigurasi, ulasan teknis, dan lain-lain-yang diterapkan selama proses berlangsung.Bahwa salah satu aspek penting dari proses perangkat lunak belum

belum dibahas. Proses aspek-disebut ini aliran-menjelaskan bagaimana kerangka

kegiatan dan tindakan dan tugas yang terjadi dalam setiap kerangka  
Terdapat 4 jenis model, antara lain:  
•    Model aliran kerja (workflow): menunjukkan kegiatan pada proses bersama dengan input, output, dan ketergantungannya. Merepresentasikan pekerjaan manusia.  
•    Model aliran data (data flow): merepresentasikan proses sebagai suatu set kegiatan yang melakukan transformasi data. Menunjukkan bagaimana input ke proses, misalnya spesifikasi ditransformasi menjadi output, misalnya menjadi desain.  
•    Model peran/aksi: merepresentasikan peran orang yang terlibat pada PL dan kegiatan yg menjadi tanggung jawab mereka.

  
•    Model air terjun (waterfall): Mengambil kegiatan dasar seperti spesifikasi, pengembangan, validasi, dan evolusi dan merepresentasikannya sebagai fase-fase proses yang berbeda seperti spesifikasi persyaratan, perancangan perangkat lunak, implementasi, pengujian dan seterusnya



Model Waterfall adalah paradigma tertua untuk rekayasa perangkat lunak . Namun,

selama tiga dekade terakhir Di antara masalah yang kadang-kadang ditemui ketika model air terjun diterapkan adalah:

1. proyek nyata jarang mengikuti aliran sekuensial yang model mengusulkan. Meskipun model linier bisa mengakomodasi iterasi , ia melakukannya secara tidak langsung. Akibatnya perubahan dapat menyebabkan kebingungan sebagai hasil tim proyek
2. Hal ini sering sulit bagi pelanggan untuk menyatakan semua persyaratan eksplisit . Itu model waterfall memiliki kesulitan menampung alami ketidakpastian yang ada pada awal banyak proyek
3. Pelanggan harus memiliki kesabaran . Sebuah versi kerja dari program ( s ) akan tidak tersedia sampai akhir rentang waktu proyek . Sebuah kesalahan besar, jika tidak terdeteksi sampai program kerja ditinjau , bisa menjadi bencana .

Kelebihan dari Waterfall

Metode ini masih lebih baik digunakan walaupun sudah tergolong kuno, daripada menggunakan pendekatan asal-asalan. Selain itu, metode ini juga masih masuk akal jika kebutuhan sudah diketahui dengan baik.

Memberikan template tentang metode analisis, desain, pengkodean, pengujian, dan pemeliharaan.

Cocok digunakan untuk produk software yang sudah jelas kebutuhannya di awal, sehingga minim kesalahannya.

Cocok untuk system software berskala besar dan bersifat generik.

Pengerjaan project system akan terjadwal dengan baik dan mudah dikontrol.

Document pengembangan system sangat terorganisir, karena setiap fase harus terselesaikan dengan lengkap sebelum melangkah ke fase berikutnya. Jadi  setiap fase atau tahapan akan mempunyai dokumen tertentu.

Kualitas dari sistem yang dihasilkan akan baik. Ini dikarenakan oleh pelaksanaannya secara bertahap. Sehingga tidak terfokus pada tahapan tertentu.

Kekurangan dari Waterfall

Diperlukanya sebuah management proyek yang baik karena sangat berpengaruh karena Masalah kecil bisa beresiko bila tidak ditanganin atau diketahui dari awal pembuatan.

Persyaratan sistem harus digambarkan dengan jelas dan Rincian proses harus benar-benar jelas dan tidak boleh berubah-ubah.

Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakukan pada tahap awal proses.

Customer harus sabar untuk menanti produk selesai, karena dikerjakan tahap pertahap, menyelesaikan tahap awal baru bisa ke tahap selanjutnya.

Sulit untuk mengadaptasi jika terjadi perubahan spesifikasi pada suatu tahapan pengembangan.

Perubahan ditengah-tengah pengerjaan produk akan membuat bingung team work yang sedang membuat produk.

Adanya waktu menganggur bagi pengembang, karena harus menunggu anggota tim proyek lainnya menuntaskan pekerjaannya.

.

2.    Agile Proccces Model

Agility telah menjadi kata kunci di hari ini saat menjelaskan proses perangkat lunak modern. Semua orang tangkas . Sebuah tim tangkas adalah tim gesit dapat tepat menanggapi perubahan. Perubahan adalah apa pengembangan perangkat lunak sangat banyak tentang . Perubahan dalam perangkat lunak yang dibangun , perubahan anggota tim , perubahan karena teknologi baru , perubahan semua jenis yang mungkin berdampak pada produk yang mereka membangun atau proyek yang menciptakan produk. Dukungan untuk perubahan harus built-in segala sesuatu yang kita lakukan dalam perangkat lunak kita merangkul karena itu adalah jantung dan jiwa dari perangkat lunak . Sebuah tim tangkas mengakui bahwa perangkat lunak yang dikembangkan oleh individu yang bekerja dalam tim dan keterampilan orang-orang ini, kemampuan mereka untuk berkolaborasi merupakan inti untuk keberhasilan proyek

Agile Software development adalah salah satu metodologi dalam pengembangan sebuah perangkat  lunak.  Kata  Agile  berarti  bersifat  cepat,  ringan,  bebas  bergerak,waspada

Agile Software Development adalah cara membangun software dengan melakukannya dan membantu orang lain membangunnya sekaligus. Jadi agile model adalah metode konvensional untuk membangun berbagai jenis perangkat lunak dan berbagai macam tipe proyek pengembangan perangkat lunak, yang dapat melakukan pengiriman atau penyampaian hasil dari implementasi sistem melalui perangkat lunak dengan cepat.

Prinsip Agility

Agile Alliance (lihat [Agi03], [Fow01]) mendefinisikan 12 prinsip kelincahan bagi mereka yang

ingin mencapai kelincahan:

1. prioritas tertinggi kami adalah untuk memuaskan pelanggan melalui awal dan terus menerus pengiriman perangkat lunak yang berharga.
2. Selamat perubahan kebutuhan, bahkan larut dalam pembangunan. proses tangkas Perubahan harness untuk keunggulan kompetitif pelanggan.
3. Memberikan software bekerja sering, dari beberapa minggu sampai beberapa bulan, dengan preferensi untuk skala waktu yang lebih pendek.
4. Orang-orang bisnis dan pengembang harus bekerja sama setiap hari selama proyek.
5. Membangun proyek di seluruh individu termotivasi. Beri mereka lingkungan dan mendukung mereka butuhkan, dan mempercayai mereka untuk mendapatkan pekerjaan yang dilakukan.
6. Metode yang paling efisien dan efektif untuk menyampaikan informasi ke dan dalam tim pengembangan adalah tatap muka percakapan.
7. software Kerja adalah ukuran utama kemajuan.
8. proses Agile mempromosikan pembangunan berkelanjutan. Para sponsor, pengembang, dan pengguna harus mampu mempertahankan kecepatan konstan tanpa batas.
9. memperhatikan keunggulan teknis dan memper- tinggi desain yang baik berkelanjutan kelincahan.
10. Kesederhanaan-seni memaksimalkan jumlah pekerjaan tidak dilakukan-adalah penting.
11. arsitektur terbaik, persyaratan, dan desain muncul dari diri mengorganisir tim.
12. Pada interval reguler, tim mencerminkan tentang bagaimana menjadi lebih efektif, maka lagu dan menyesuaikan perilaku yang sesuai.

**a.**      **Tahapan *Agile Model***

1)      Perencanaan

2)      *Requirements analysis*

3)      *Design*

4)      *Coding*

5)      *Testing*

6)      Dokumentasi

**Kelebihan dari agile**

Meningkatkan kepuasan kepada klien.

Dapat melakukan review pelanggan mengenai software yang dibuat lebih awal.

Pembangunan system dibuat lebih cepat.

Mengurangi resiko kegagalan implementasi software dari segi non-teknis.

Jika pada saat pembangunan system terjadi kegagalan kerugian dari segi materi relatif kecil.

**Kekurangan dari agile**

Developer harus selalu siap dengan perubahan karena perubahan akan selalu diterima.

Agile tidak akan berjalan dengan baik jika komitmen tim kurang.

Tidak cocok dalam skala tim yang besar (>20 orang).

 Perkiraan waktu release dan harga perangkat lunak sulit ditentukan.