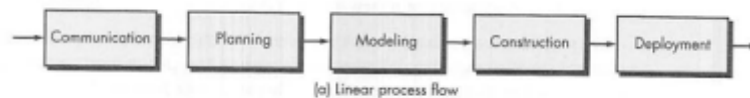


**Nama : Isna Khoiriyah**

**Kelas : MI-6/1421024155**

## **1. Linear Process Flow/Waterfall**

Model *Linear Sequential/Waterfall* merupakan paradigma rekayasa perangkat lunak yang paling tua dan paling banyak dipakai. Model ini bagus dipilih bila semua kebutuhan sudah siap jauh sebelum sistem akan dimulai.



*Melaksanakan masing-masing dari lima framework activity secara urut dimulai dengan communication dan mencapai puncaknya dengan deployment*

### **Kelebihan model *Linear Sequential/Waterfall* :**

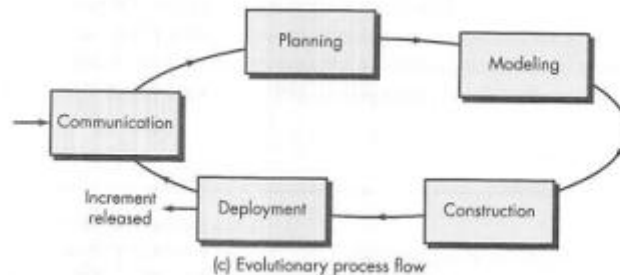
- Mudah diaplikasikan
- Menghasilkan sistem yang terstruktur dengan baik
- Memiliki tingkat visibilitas yang tinggi ( setiap tahap mempunyai dokumen yang jelas )

### **Kekurangan model *Linear Sequential/Waterfall* :**

- Susah dalam merespon kebutuhan customer
- Model ini hanya cocok jika kebutuhan customer sudah sangat jelas dan perubahan kebutuhan dibatasi
- Customer harus sabar untuk menanti produk selesai, karena dikerjakan tahap per tahap, menyelesaikan tahap awal baru bisa ke tahap selanjutnya
- 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. Evolutionary Process Flow/Prototyping

Model ini digunakan ketika user sudah bisa mendefinisikan kebutuhan umum tetapi tidak bisa secara detail menyatakan kebutuhan fungsi-fungsi fiturnya/pengembang tidak yakin dengan efisiensi algoritmanya.



*Menjalankan aktivitas dengan cara melingkar. Setiap sirkuit yang melalui lima kegiatan mengarah ke versi yang lebih lengkap dari perangkat lunak*

### Kelebihan model prototyping :

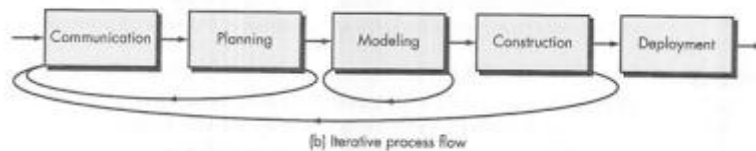
- Adanya komunikasi yang baik antara pengembang dan pelanggan
- Dapat menghemat waktu pengembangan
- User dapat berpartisipasi aktif dalam pengembangan sistem
- Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan

### Kekurangan model prototyping :

- Proses perancangan dan analisis terlalu singkat
- Pelanggan kadang tidak melihat bahwa perangkat lunak yang ada belum mencantumkan kualitas perangkat lunak secara keseluruhan
- Biasanya kurang fleksibel terhadap perubahan

## 3. Iterative Process Flow/Incremental

Dalam proses pembuatannya, model ini apabila di tengah jalan mendapat ide baru, maka ide tersebut menjadi versi lain sehingga framework process dimulai dari awal. Biasanya incremental model adalah inti produk dengan banyak tambahan ( modification user )



*Mengulang satu atau lebih kegiatan sebelum melanjutkan ke yang berikutnya*

#### **Kelebihan model incremental :**

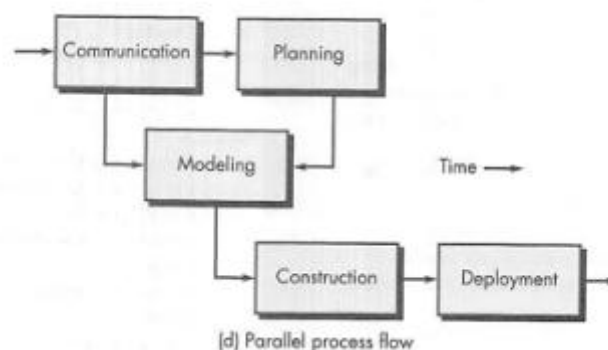
- Mampu mengkondisikan perubahan kebutuhan customer
- Pihak konsumen dapat langsung menggunakan dahulu bagian-bagian yang telah selesai dibangun
- Memaksimalkan pengembalian modal investasi konsumen

#### **Kekurangan model incremental :**

- Penambahan staff dilakukan jika hasil incremental akan dikembangkan lebih lanjut
- Hanya cocok untuk proyek dengan skala kecil
- Kemungkinan tiap bagian tidak dapat diintegrasikan

#### **4. Parallel Process Flow/RAD**

Model RAD merupakan model proses pengembangan perangkat lunak secara linear sequential yang menekankan pada siklus pengembangan yang sangat singkat.



*Menjalankan satu atau lebih kegiatan secara paralel dengan kegiatan lain. (Contoh modelling untuk satu aspek pada software akan dijalankan secara paralel dengan construction aspek lain pada software)*

### **Kelebihan model RAD :**

- Lebih efektif dari pendekatan *waterfall/sequential linear* dalam menghasilkan sistem yang memenuhi kebutuhan langsung dari pelanggan
- Cocok untuk proyek yang memerlukan waktu yang singkat

### **Kekurangan model RAD :**

- RAD tidak cocok digunakan untuk sistem yang mempunyai resiko teknik yang tinggi
- Membutuhkan orang yang banyak untuk menyelesaikan sebuah proyek berskala besar
- Pengembang dan customer harus punya komitmen yang kuat untuk menyelesaikan sebuah software
- Jika sistem tidak di bangun dengan benar maka RAD akan bermasalah
- Jika ada perubahan di tengah-tengah pengerjaan maka harus membuat kontrak baru antara pengembang dan customer.

*Jadi disini yang paling bagus menurut saya adalah Linier/Waterfall karena Model ini akan memudahkan para pengembang perangkat lunak untuk menetapkan tanggung jawab, memonitor penjadwalan proyek serta akuntabilitas peran personal dalam proyek perangkat lunak.*

- *Mudah untuk diaplikasikan*
- *Model waterfall ini sangat cocok untuk diterapkan jika semua kebutuhan sudah terpenuhi jauh sebelum model dibuat.*
- *Memberikan template tentang metode anaisis, pegkodean, pengujian, design, serta pemeliharaan. Dan lain-lain. Demikian tugas ini . terimakasihh .*