

# 1. Liniear Process Flow

## **Kelebihan dari model waterfal/linear sequential model :**

- Mudah diaplikasikan karena urutan-urutan pengerjaan sudah sering dipakai
- Cocok untuk software berskala besar dan yang bersifat umum
- Langkah-langkahnya sangat sekuensial, pengerjaan proyek akan mudah dikontrol dan terjadwal dengan baik.
- Tiap tahap memiliki metode untuk menghasilkan suatu dokumen yang bisa diserahkan ke pemakai
- Dokumen yang dihasilkan tiap tahap bisa di spesifikasikan secara jelas dan mendetail

## **kekurangan dari model waterfal/linear sequential model :**

- Persyaratan system harus digambarkan dengan jelas.
- Rincian proses harus benar-benar jelas dan tidak boleh berubah-ubah.
- Sulit untuk mengadaptasi jika terjadi perubahan spesifikasi pada suatu tahapan pengembangan
- Terjadinya pembagian proyek menjadi tahap-tahap yang tidak fleksibel, karena komitmen harus dilakukan pada tahap awal proses
- Sulit untuk mengalami perubahan kebutuhan yang diinginkan customer
- 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. Iterative Process Flow**

### **Incremental Process Models :**

Keunggulan dari Incremental Model :

- Personil bekerja optimal
- Pihak konsumen dapat langsung menggunakan dahulu bagian-bagian yang telah selesai dibangun. Contohnya pemasukan data karyawan
- Mengurangi trauma karena perubahan sistem. Klien dibiasakan perlahan-lahan menggunakan produknya bagian per bagian
- Memaksimalkan pengembalian modal investasi konsumen

Kekurangan dari Incremental Model :

- Cocok untuk proyek berukuran kecil (tidak lebih dari 200.000 baris coding)
- Mungkin terjadi kesulitan untuk memetakan kebutuhan pengguna ke dalam rencana spesifikasi masing-masing hasil increment
- Dapat menjadi build and Fix Model, karena kemampuannya untuk selalu mendapat perubahan selama proses rekayasa berlangsung

## **3. Evolutionary Process Flow**

### **Keuntungan Evolutionary Model**

- Pengguna mendapatkan kesempatan untuk bereksperimen dengan sistem sebagian dikembangkan
- Banyak sebelum versi penuh bekerja dilepaskan, Membantu menemukan kebutuhan pengguna yang tepat
- Banyak sebelum sistem kerja sepenuhnya dikembangkan. Core modul bisa diuji secara menyeluruh: mengurangi kemungkinan kesalahan dalam produk akhir.

### **Kekurangan Evolutionary Model**

- Seringkali, sulit untuk membagi masalah menjadi unit-unit fungsional yang dapat diimplementasikan secara bertahap dan disampaikan.
- Evolutionary model ini berguna untuk masalah yang sangat besar, mana lebih mudah untuk menemukan modul untuk implementasi tambahan

Evolutionary model mempunyai dua jenis yaitu prototype model dan spiral model.

#### **a. Prototype model**

##### **Kelebihan prototype model :**

- Jika menggunakan model ini maka dapat menghemat waktu pengembangan
- Akan terjalin komunikasi yang baik antara kostumer dan pengembang
- Pengembang dapat bekerja lebih baik dalam menentukan kebutuhan pelanggan
- User mnegetahui apa yang diharapkan sehingga penerapan menjadi lebih mudah.
- Dalam pengembangan sistem user bisa berpartisipasi secara aktif.

##### **Kekurangan prototype model :**

- Proses analisa dan perancangan dalam prototype model terlalu singkat
- Biasanya kurang fleksible dalam menghadapi perubahan
- Mengesampingkan alternatif pemecahan masalah.
- Prototype yang dihasilkan tidak selamanya mudah dirubah
- Prototype terlalu cepat selesai

#### **b. Spiral model**

Spiral model merupakan gabungan antara prototype model dan waterfall model. Spiral model didefinisikan dalam aktivitas kerangka kerja oleh team software engineering.

##### **Kelebihan dari model spiral :**

- Lebih cocok untuk pengembangan sistem dan perangkat lunak skala besar
- Pengembang dan pemakai dapat lebih mudah memahami dan bereaksi terhadap resiko setiap tingkat evolusi karena perangkat lunak terus bekerja selama proses

- Dapat disesuaikan agar perangkat lunak bisa dipakai selama hidup perangkat lunak komputer.
- Lebih cocok untuk pengembangan sistem dan perangkat lunak skala besar
- Pengembang dan pemakai dapat lebih mudah memahami dan bereaksi terhadap resiko setiap tingkat evolusi karena perangkat lunak terus bekerja selama proses .

**Kekurangan dari model spiral :**

- Sulit untuk meyakinkan pemakai (saat situasi kontrak) bahwa penggunaan pendekatan ini akan dapat dikendalikan
- Memerlukan tenaga ahli untuk memperkirakan resiko, dan harus mengandalkannya supaya sukses
- Belum terbukti apakah metode ini cukup efisien karena usianya yang relatif baru
- Terlalu banyak memikirkan resiko yang akan terjadi
- Masih jarang digunakan
- Metode ini lambat dan mahal karena setiap tahapan yang dilalui harus menikutsertakan pemesan

## **4. Parallel Process Flow**

Model RAD atau Rapid Application Development

**Kelebihan dari model RAD atau Rapid Application Development adalah**

- 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 dari model RAD atau Rapid Application Development adalah**

- 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

**Urutan Software Process yang terbaik menurut analisis saya ialah :**

1. Liniear Process Model
2. Parallel Process Model
3. Evolutionary Process Model
4. Iterative Process Model