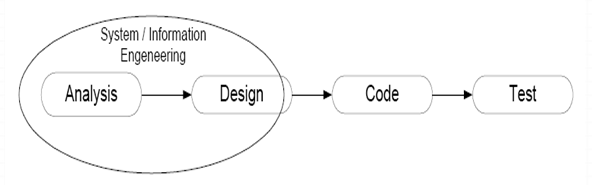
**MODEL PROSES**

Model proses : representasi abstrak dari proses pembangunan proyek. (urutan – urutan pembangunan perangkat lunak)

1. **Waterfall Model**

Disebut juga dengan model linier sekuensial



* Tiga tahap pada model proses ini diakhiri dengan dokumentasi.
* Model ini sering disebut Document Driven Software Process.
* **Kelebihan :**
  + Semua kebutuhan sistem dapat didefinisikan secara utuh dan explisit.
  + mudah diaplikasikan karena urutan-urutan pengerjaan sudah sering dipakai.
  + cocok untuk software berskala besar dan yang bersifat umum.
  + pengerjaan proyek akan mudah dikontrol dan terjadwal dengan baik.
* **Kekurangan :**
  + ketika ada masalah, maka proses berhenti.
  + karena pendekatannya secara sequential, maka tiap tahap harus menunggu hasil dari tahap sebelumnya.
  + jika tahap tidak dikerjakan, maka resourcenya tidak dibutuhkan lagi.

1. **Rapid Prototyping Model**

Model prototyping atau Metode Prototype adalah metode pengembangan sistem perangkat lunak (SLDC) di mana prototipe dibangun, diuji dan kemudian dikerjakan ulang seperlunya sampai hasil yang dapat diterima dicapai dari sistem atau produk yang lengkap dapat dikembangkan.

* 1. Model ini digunakan untuk mengeksplorasi kebutuhan customer.
  2. Dengan melihat prototype, customer dapat memberikan masukan - masukan kepada pengembang sehingga prototype yang dibangun dapat mendekati kebutuhan.

(digunakan untuk mengembangkan aplikasi, bukan membangun aplikasi)

* 1. Model ini sering dikenal sebagai "**Code Driven Software Process**"  dan merupakan proses perangkat lunak yang banyak digunakan pada saat ini.
  2. **Kelebihan :**Memilki produktivitas yang lebih baik.
  3. **Kekurangan :** Kelengkapan fungsi dari sistem dan keterpaduan sistem kurang baik.

1. **Spiral Model**
   1. Model ini sering disebut dengan "**Risk Driven Software Process**"
   2. Digunakan untuk proyek skala besar dengan memperhatikan resiko, dilihat dari segi finansial maupun keamanan.
   3. Merupakan kombinasi Linear Sekuensial, Prototyping dan Risk analysis.
   4. Pada tahap akhir dibuat hasil dokumenta analisis resiko.

RUP: pendekatan pengembangan perangkat lunak yang dilakukan berulang ulang, focus pada arsitektur, dan lebih diarahkan berdasarkan penggunaan kasus

* Inception : tahap yang berfokus pada pemodelan bisnis serta mendefinisikan kebutuhan akan sistem yang dibuat
* Elaboration : focus pada perencanaan arsitektur sistem dan mendeteksi resiko yang mungkin terjadi pada arsitektur yang dibuat.
* Construction : focus pada pengembangan sistem dan pengimplementasian sistem yang berfokus pada kode program
* Transition : focus pada deployment atau instalasi sistem agar dapat dimengerti user