**DASAR SISTEM INFORMASI**

***SDLC WATERFALL***



Disusun oleh:

**Epafroditus Kusuma Putra ( 2018140018 )**

**Danar Yugo Prakoso ( 3120510507 )**

**Mohammad Jawahir Alma’rifatullah ( 3120510501 )**

**D3 PJJ Teknik Informatika**

**Politeknik Elektronika Negeri Surabaya**

**2020**

**Apa itu SDLC Waterfall?**

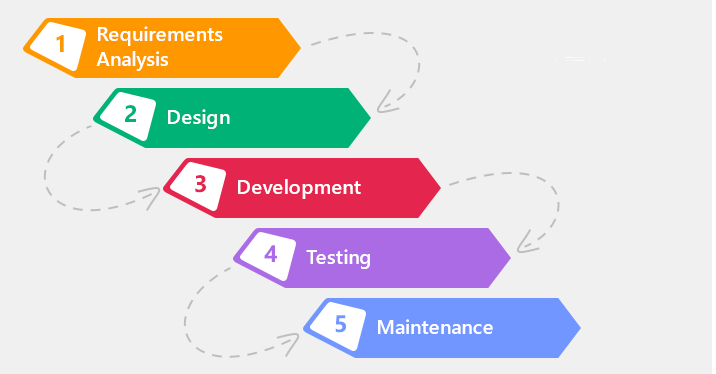
Semua *software developer*dalam perusahaan pastinya menggunakan SDLC mengembangkan *software* atau aplikasi. Sebelumnya apa itu SDLC atau *Software Development Life Cycle*? SDLC adalah tahapan-tahapan pekerjaan yang dilakukan oleh analis sistem dan *programmer*dalam membangun sistem informasi dan metode dalam mengembangkan sistem tersebut. Sistem yang dibangun dengan dengan menggunakan SDLC akan memudahkan dalam mengidentifkasi masalah dan merancang sistem sesuai kebutuhan dalam menyelesaikan permasalahan tersebut. Salah satu SDLC yang paling sering digunakan dalam pengembangan sistem yaitu SDLC *Waterfall.*



SDLC *Waterfall*sesuai namanya SDLC ini berkembang secara sistematis dari satu tahap ke tahap lain layaknya air terjun. Metode *waterfall* merupakan suatu metode dalam pengembangan *software*dimana pengerjaannya harus dilakukan secara berurutan yang dimulai dari tahap perencanaan konsep,pemodelan(*design*),implementasi,pengujian dan pemeliharaan.

**A.Tahapan Metode Waterfall**

Berikut ini merupakan tahapan tahapan pengembangan dalam metode *waterfall*.



**1.*Requirement Analysis***Pada tahap ini pengembang sistem diperlukan suatu komunikasi yang bertujuan untuk memahami *software* yang dibutuhkan pengguna dan batasan *software*. Informasi ini biasanya dapat diperoleh melalui wawancara, *survey* atau diskusi.

**2.*System Design***  
Pada proses desain, dilakukan penerjemahan syarat kebutuhan ke sebuah perancangan desainperangkat lunak yang dapat diperkirakan sebelum dibuatnya proses pengkodean (*coding*). Proses ini berfokus pada struktur data, arsitektur perangkat lunak, representasi *interface*, dan detail algoritma prosedural.  
Berikut contoh desain sistem yang biasanya dibuat dan digunakan.

* *Data Flow Diagram*(DFD).
* *Flowchart.*
* *Mind Map.*
* *Entity Relationship Diagram* (ERD).
* *Context Diagram, etc*.

**3.*Implementation***  
Pada tahap ini terjadi proses menerjemahkan perancangan desain ke bentuk yang dapat dimengerti oleh mesin, dengan menggunakan kode kode bahasa pemrograman. Kode program yang dihasilkan masih berupa modul-modul kecil yang nantinya akan digabungkan pada tahap berikutnya.

**4.*Integration & Testing***Di tahap ini dilakukan penggabungan modul-modul yang sudah dibuat dan dilakukan pengujian ini dilakukan untuk mengetahui apakah *software*yang dibuat telah sesuai dengan desainnya dan fungsi pada *software* terdapat kesalahan atau tidak.

**5.*Operation & Maintenance***  
Ini merupakan tahap terakhir dalam model *waterfall*. *Software*yang sudah jadi dijalankan serta dilakukan pemeliharaan. Pemeliharaan termasuk dalam memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya. Perbaikan implementasi unit sistem dan peningkatan jasa sistem sebagai kebutuhan baru.

**B.Kelebihan Metode Waterfall**

* *Waterfall* adalah model pengembangan paling handal dan paling lama digunakan.
* Cocok untuk sistem *software* dengan kompleksitas rendah (*predictable*).
* Pengerjaan *project* sistem akan terjadwal dengan baik dan mudah dikontrol.

**C.Kekurangan Metode Waterfall**

* Waktu pengembangan lama karena harus menunggu tahap sebelumnya selesai.
* Biaya juga mahal, hal ini juga dikarenakan waktu pengembangan yang lama.
* Karena tahap-tahapan pada *waterfall*tidak dapat berulang, maka model ini tidak cocok untuk pemodelan pengembangan sebuah proyek yang memiliki kompleksitas tinggi.
* Perubahan spesifikasi perangkat lunak yang terjadi ditengah alur pengembangan
* Sangat sulit bagi pelanggan untuk mendeskripsikan kebutuhan spesifikasi di awal pengembangan.Pelanggan sering kali membutuhkan contoh (prototype) untuk menjabarkan spesifikasi kebutuhan sistem lebih lanjut.
* Pelanggan tidak mungkin bersabar mengakomodasi perubahan yang diperlukan di akhir alur pengembangan.

Jadi, menggunakan metode *waterfall*akan memudahkan kita dalam melakukan manajemen dan merancang *requirement*karena*waterfall*mudah dipahami dan digunakan.

Sekian penjelasan tentang SDLC *Waterfall*, jika ada kesalahan atau saran bisa ditambahkan pada kolom komentar, terimakasih.

**Referensi :**

[Kelebihan dan Kekurangan Metode Waterfall dalam Pengembangan Sistem - DosenIT.com](https://dosenit.com/kuliah-it/teknologi-informasi/kelebihan-dan-kekurangan-metode-waterfall" \t "_blank)

[Waterfall apabila diartikan secara literature berarti air terjun. Namun demikian, bagi ilmu komputer dan juga teknologi…](https://dosenit.com/kuliah-it/teknologi-informasi/kelebihan-dan-kekurangan-metode-waterfall" \t "_blank)

[dosenit.com](https://dosenit.com/kuliah-it/teknologi-informasi/kelebihan-dan-kekurangan-metode-waterfall" \t "_blank)

<http://www.sistem-informasi.xyz/2017/04/pengertian-waterfall-sdlc.html>