



**BORCELLE**

# **SDLC**

Presented by:

*Dussi Miss Deria*



# DEFENISI

SDLC adalah kependekan dari Systems development life cycle atau dalam bahasa Indonesia disebut siklus hidup pengembangan sistem

SDLC adalah siklus yang digunakan dalam pembuatan atau pengembangan sistem informasi yang bertujuan untuk menyelesaikan masalah secara efektif.

SDLC adalah tahapan kerja yang bertujuan untuk menghasilkan sistem berkualitas tinggi yang sesuai dengan keinginan pelanggan atau tujuan dibuatnya sistem tersebut.

SDLC menjadi kerangka yang berisi langkah-langkah yang harus dilakukan untuk memproses pengembangan suatu perangkat lunak.



# METODE / MODEL SDLC

○ WATERFALL

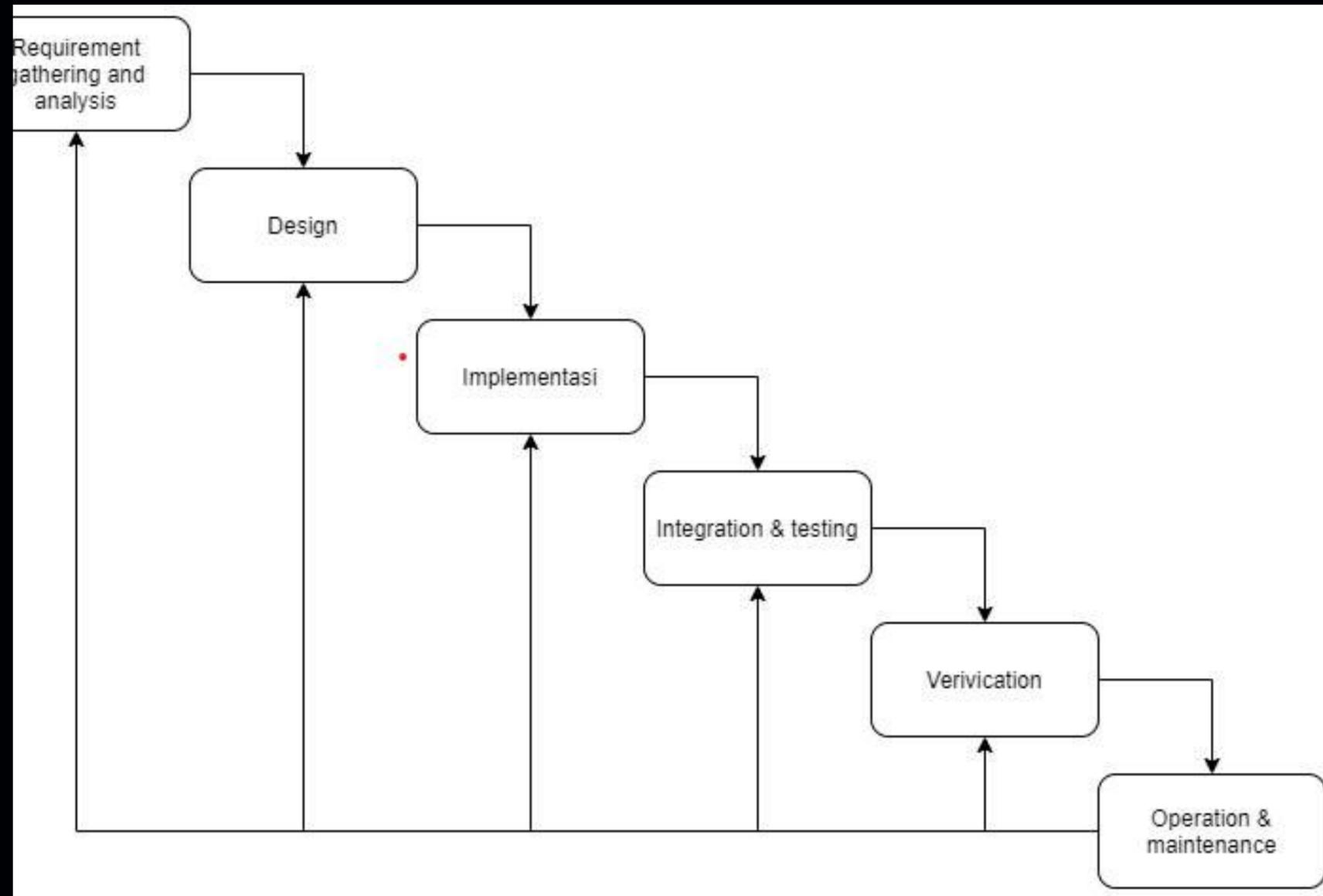
○ PROTOTYPE / ITERATIF

○ AGILE / SPIRAL

○ FOUNTAIN



# WATERFALL





# WATERFALL

Metode SDLC yang pertama adalah waterfall. Metode waterfall adalah metode kerja yang menekankan fase-fase yang berurutan dan sistematis. Disebut waterfall karena proses mengalir satu arah “ke bawah” seperti air terjun. Metode waterfall ini harus dilakukan secara berurutan sesuai dengan tahap yang ada.

# TAHAP-TAHAP PENGEMBANGAN DALAM METODE WATERFALL

## ○ Requirement gathering and analysis

Mengumpulkan kebutuhan secara lengkap untuk dianalisis dan mendefinisikan kebutuhan apa saja yang harus dicapai oleh program. Informasi dapat diperoleh melalui wawancara, diskusi, atau survey.

## ○ Design

Melakukan perancangan desain perangkat lunak sebagai perkiraan sebelum dibuatnya kode. Desain sistem dapat dibuat menggunakan Flowchart, Mind Map, atau Entity Relationship Diagram (ERD).

## ○ IMPLEMENTASI

Implementasi ini adalah tahap dimana seluruh desain yang sebelumnya sudah dibuat diubah menjadi kode-kode program. Kode yang dihasilkan masih berbentuk modul-modul yang harus digabungkan di tahap selanjutnya.



# TAHAP-TAHAP PENGEMBANGAN DALAM METODE WATERFALL

## ○ Integration & testing

Di tahap ini dilakukan penggabungan modul-modul yang sudah dibuat sebelumnya dan melakukan pengujian untuk mengetahui apakah perangkat lunak yang dibuat telah sesuai dengan desain dan fungsinya atau tidak.

## ○ Verification

Di tahap ini, pengguna atau klien yang langsung melakukan pengujian pada sistem, apakah sistem telah sesuai dengan yang disetujui atau belum sesuai.

## ○ Operation & maintenance

Tahap ini merupakan tahap terakhir dari model waterfall. Sistem yang sudah selesai dijalankan serta dilakukan pemeliharaan. Pemeliharaan berupa memperbaiki kesalahan yang tidak ditemukan pada langkah sebelumnya.



# KELEBIHAN VS KEKURANGAN

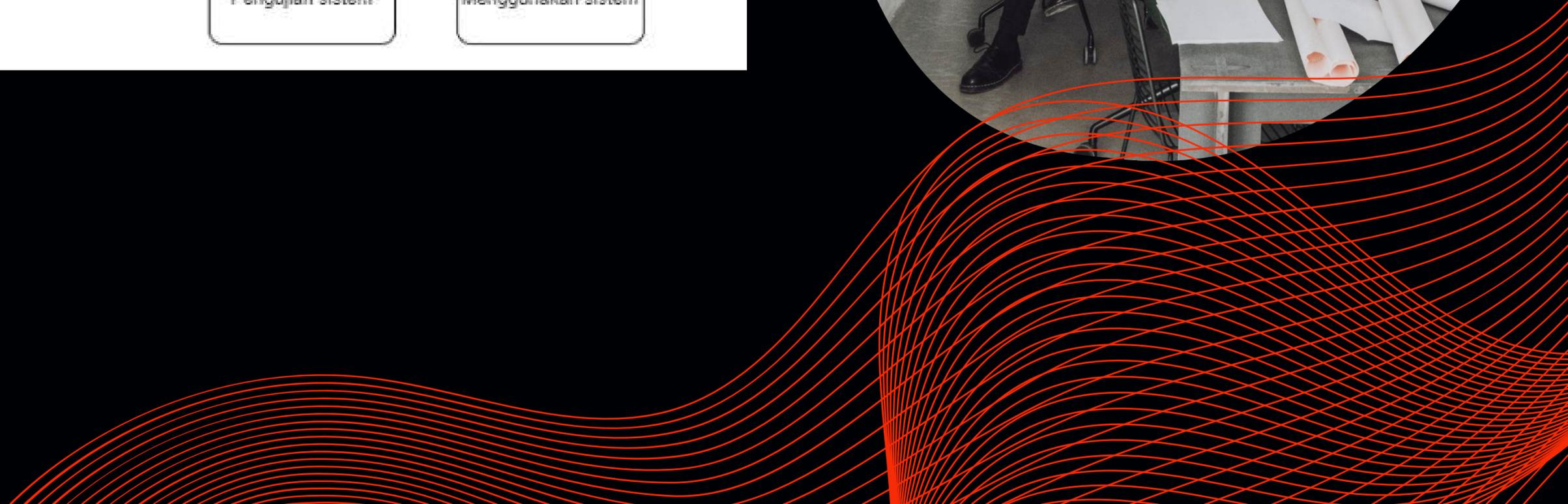
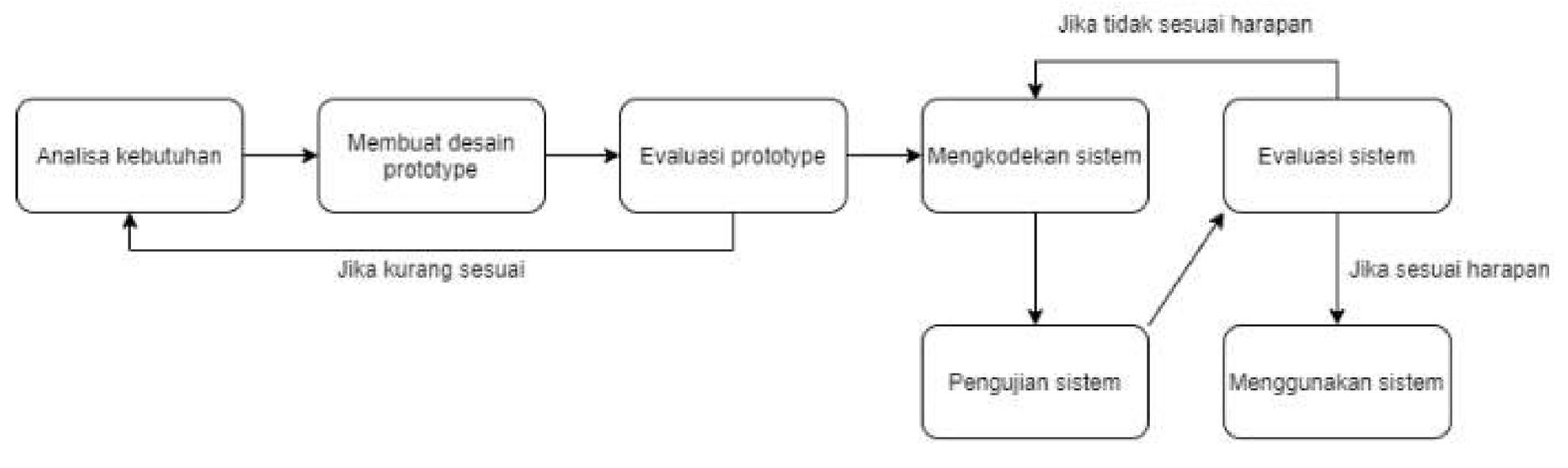


1. Memiliki proses yang terurut, sehingga pengerjaan dapat terjadwal dengan baik dan mudah.
2. Cocok untuk sistem dengan kompleksitas rendah (predictable).
3. Setiap proses yang dilakukan tidak dapat saling tumpah tindih.



1. Waktu pengerjaan relatif lebih lama, karena harus menunggu tahap sebelumnya selesai.
2. Biaya yang dibutuhkan lebih mahal karena waktu pengembangan yang dibutuhkan lebih lama.
3. Model waterfall ini kurang cocok untuk pengembangan proyek yang memiliki kompleksitas tinggi

# PROTOTYPE





# PROTOTYPE

Metode SDLC selanjutnya adalah prototype. Metode prototype adalah metode yang memungkinkan pengguna atau user memiliki gambaran awal tentang perangkat lunak yang akan dikembangkan, serta pengguna dapat melakukan pengujian di awal sebelum perangkat lunak dirilis.

Metode ini bertujuan untuk mengembangkan model menjadi perangkat lunak yang final. Artinya sistem akan dikembangkan lebih cepat dan biaya yang dikeluarkan lebih rendah. Metode prototype ini memiliki tahap-tahap yang harus dilakukan dalam pengembangan perangkat lunak.

# TAHAP-TAHAP PENGEMBANGAN DALAM METODE PROTOTYPE

## Analisa kebutuhan

Pada tahap ini pengembang melakukan identifikasi perangkat lunak dan semua kebutuhan sistem yang akan dibuat.

## Membuat prototype

Membuat rancangan sementara yang berfokus pada alur program kepada pengguna.

## Evaluasi prototype

Evaluasi dilakukan untuk mengetahui apakah model prototype sudah sesuai dengan harapan.



# TAHAP-TAHAP PENGEMBANGAN DALAM METODE PROTOTYPE

## Mengkodekan sistem

Jika prototype disetujui maka akan diterjemahkan ke dalam bahasa pemrograman yang sesuai.

## Pengujian sistem

Setelah perangkat lunak sudah siap, perangkat lunak harus melewati pengujian. Pengujian ini biasanya dilakukan dengan White Box Testing, Black Box Testing, dan lain-lain.

## Evaluasi sistem

Pengguna melakukan evaluasi apakah perangkat lunak sudah sesuai dengan apa yang diharapkan atau tidak. Jika ya, lakukan tahap selanjutnya. Jika tidak, ulangi tahap mengkodekan sistem dan pengujian sistem.

## Menggunakan sistem

Perangkat lunak yang telah diuji dan disetujui siap untuk digunakan.



# KELEBIHAN VS KEKURANGAN

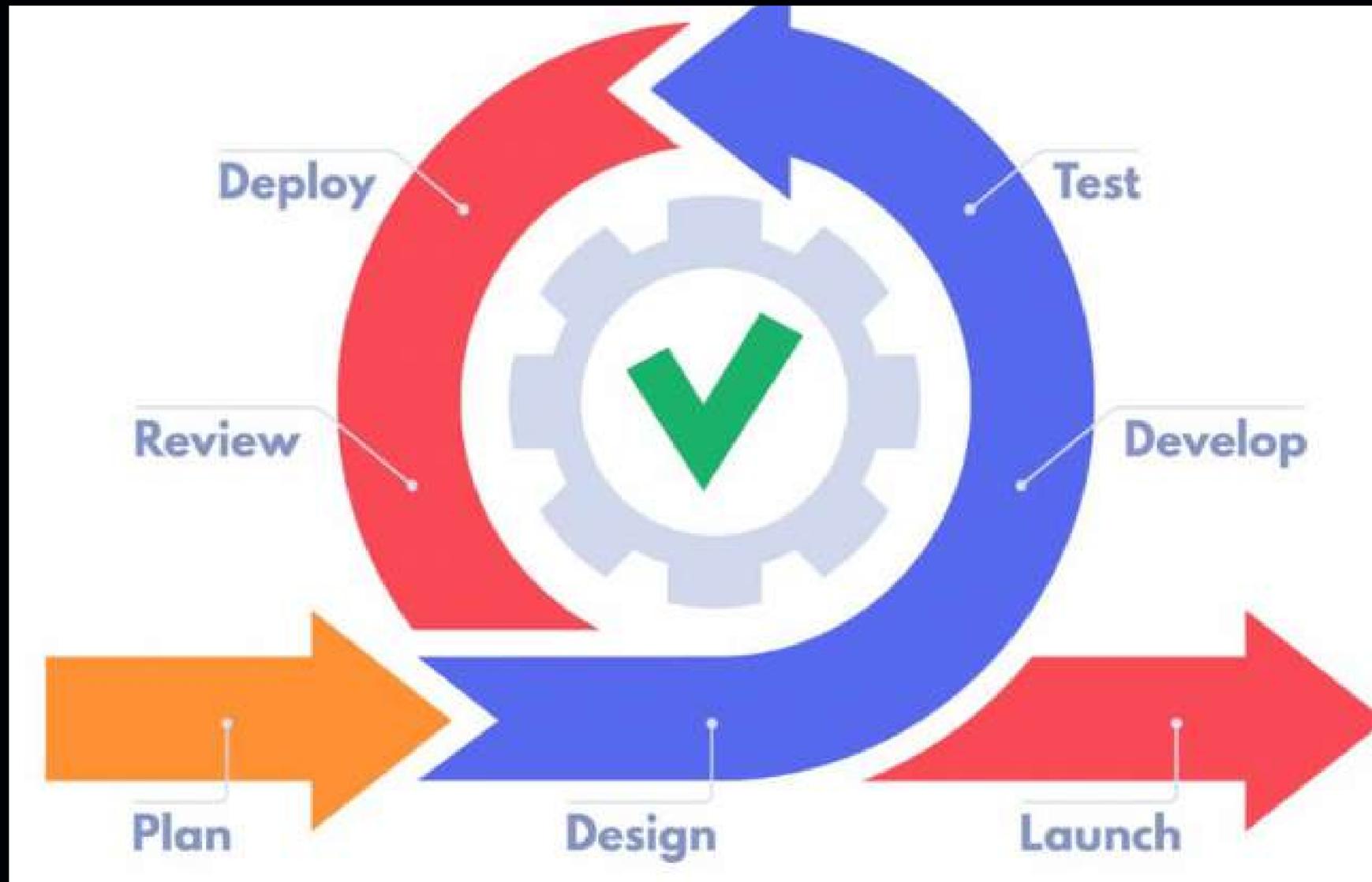


1. Mempersingkat waktu pengembangan perangkat lunak
2. Penerapan fitur menjadi lebih mudah, karena pengembang mengetahui apa yang diharapkan



1. Proses yang dilakukan untuk analisis dan perancangan terlalu singkat
2. Kurang fleksibel jika terjadi perubahan

# AGILE / SPIRAL





# AGILE / SPIRAL

Metode SDLC ketiga adalah agile. Metode agile adalah metode yang fleksibel di mana pengembangan dilakukan dalam jangka pendek. Namun diperlukan adaptasi yang cepat dari developer terhadap perubahan dalam bentuk apa pun.

# TUJUAN AGILE

## ○ High – value & working app system

Menghasilkan produk dengan kualitas yang baik, dan memiliki nilai jual tinggi.

## ○ Flexible & risk management

Pengembangan dapat dilakukan secara iteratif, berulang-ulang, dan dapat mengalami perubahan jika diperlukan.

## ○ Cost control & value – driven development

Pengembangan perangkat lunak dapat sesuai dengan kebutuhan pengguna dan tim dapat dengan cepat merespon kebutuhan, sehingga waktu dan biaya pembuatan dari perangkat lunak dapat dikendalikan.

## ○ High – quality production

Kualitas dari perangkat lunak tetap terjaga, meskipun waktu dan biaya lebih sedikit.



# TUJUAN AGILE

## **Flexible & risk management**

Meminimalisir terjadinya kesalahan pada program ataupun produk sebelum dilakukannya proses deploy aplikasi.

## **Collaboration**

Kolaborasi ini dilakukan oleh setiap tim pengembang untuk mendiskusikan feedback yang diberikan oleh klien.

## **Self – organizing, self – managing teams**

Pengembang diberikan akses untuk memajemen sendiri urusan software development. Seorang manajer hanya bertugas sebagai penghubung antara pengembang dengan klien sehingga dapat mengurangi terjadinya miss communication.



# KELEBIHAN VS KEKURANGAN

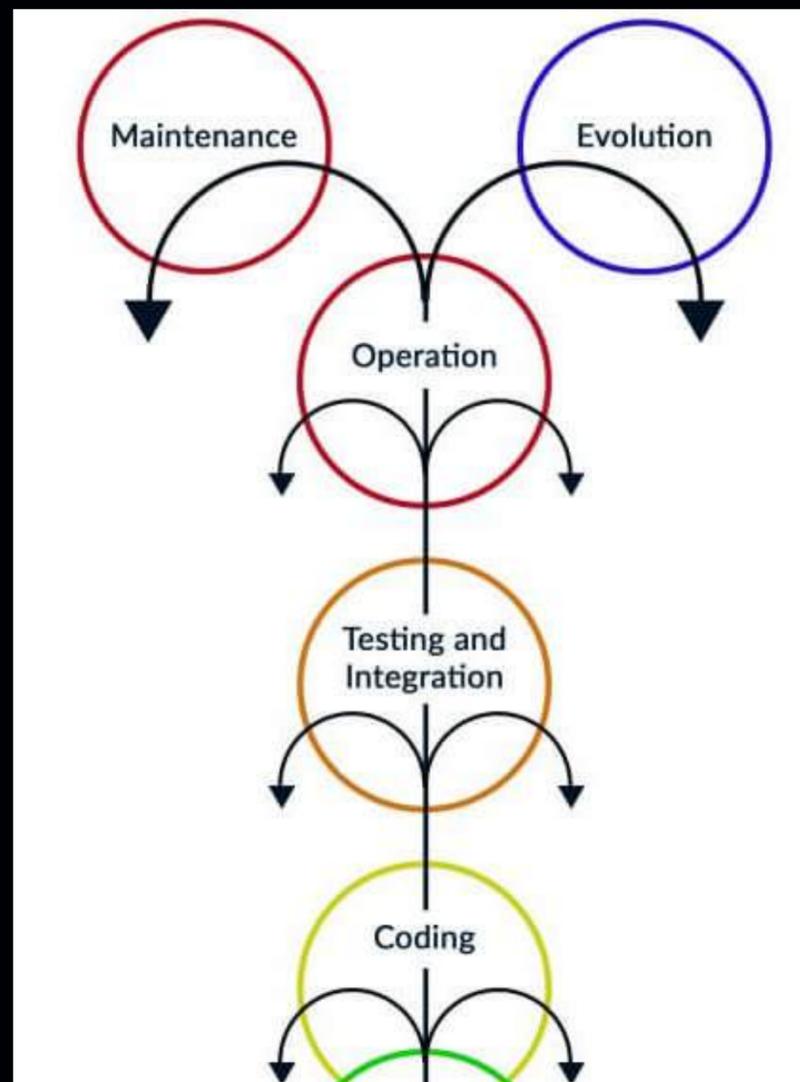


1. Perubahan dapat dengan cepat ditangani.
2. Proses pengembangan perangkat lunak membutuhkan waktu yang relatif cepat dan tidak memerlukan sumber daya yang besar.
3. Klien dapat memberikan feedback kepada pengembang dalam proses pembuatan program.



1. Metode ini kurang sesuai dengan tim yang besar (lebih dari 20 orang).
2. Tim harus selalu siap, karena perubahan dapat terjadi kapan saja.
3. Metode ini kurang cocok untuk tim yang berkomitmen untuk menyelesaikan proyek bersama-sama.

# FOUNTAIN





# FOUNTAIN

Metode SDLC yang terakhir adalah fountain. Metode fountain adalah perbaikan dari metode waterfall, di mana jenis tahapan masih sama. Namun beberapa jenis tahapan boleh didahulukan atau dilewati, tetapi ada tahapan yang tidak bisa dilewati, contohnya seperti kamu memerlukan design sebelum melakukan implementasi, jika hal tersebut dilewati maka akan ada tumpang tindih.

# TAHAP-TAHAP PENGEMBANGAN DALAM METODE FOUNTAIN

## User requirement specification

Mencari tahu apa saja yang dibutuhkan oleh pengguna dalam perangkat lunak yang sedang dikembangkan

## Software requirement specification

Penyesuaian perangkat lunak dari sisi pengguna.

## System design

Pembuatan desain sistem yang akan dibuat sebelum diimplementasikan.

## Program design

Pembuatan desain yang lebih sempurna dan hampir mendekati hasil akhir dari perangkat lunak.



# TAHAP-TAHAP PENGEMBANGAN DALAM METODE FOUNTAIN

## Implementation

Di tahap ini dilakukan implementasi sesuai dengan desain yang sudah dibuat di tahap sebelumnya.

## Program testing: unit

Dalam tahap ini dilakukan uji coba terhadap unit-unit yang dibutuhkan dalam perangkat lunak yang dikembangkan.

## Program testing: system

Di tahap ini dilakukan uji coba terhadap sistem dari perangkat lunak seutuhnya sebelum perangkat lunak digunakan.

## Program use

Dalam tahap ini dilakukan pengajaran kepada pengguna untuk menggunakan perangkat lunak yang telah dibuat.



# TAHAP-TAHAP PENGEMBANGAN DALAM METODE FOUNTAIN

## ○ Software maintenance

Biasanya dalam tahap ini dilakukan perawatan terhadap perangkat lunak yang sudah dibuat, perawatan dapat berupa update sistem atau perbaikan kesalahan atau bugs yang ada.



# KELEBIHAN VS KEKURANGAN



1. Memiliki proses yang terurut, sehingga pengerjaan dapat terjadwal dengan baik dan mudah.
2. Cocok untuk sistem dengan kompleksitas rendah (predictable).
3. Dapat melewati atau mendahului beberapa tahapan.
4. Setiap proses yang dilakukan tidak dapat saling tumpang tindih.



1. Waktu pengerjaan relatif lebih lama, karena harus menunggu tahap sebelumnya selesai.
2. Biaya yang dibutuhkan lebih mahal karena waktu pengembangan yang dibutuhkan lebih lama.
3. Model fountain ini kurang cocok untuk pengembangan proyek yang memiliki kompleksitas tinggi.

# TERIMA KASIH

Silakan diulang materi ini dirumah....

