

# Dasar Quality Assurance

SDLC



# Software Development Life Cycle (SDLC)

- ☐ Introduction SDLC
- ☐ Waterfall
- ☐ Prototype
- ☐ Agile Scrum

# Software Development Life Cycle (SDLC)



**Introduction SDLC**



Waterfall



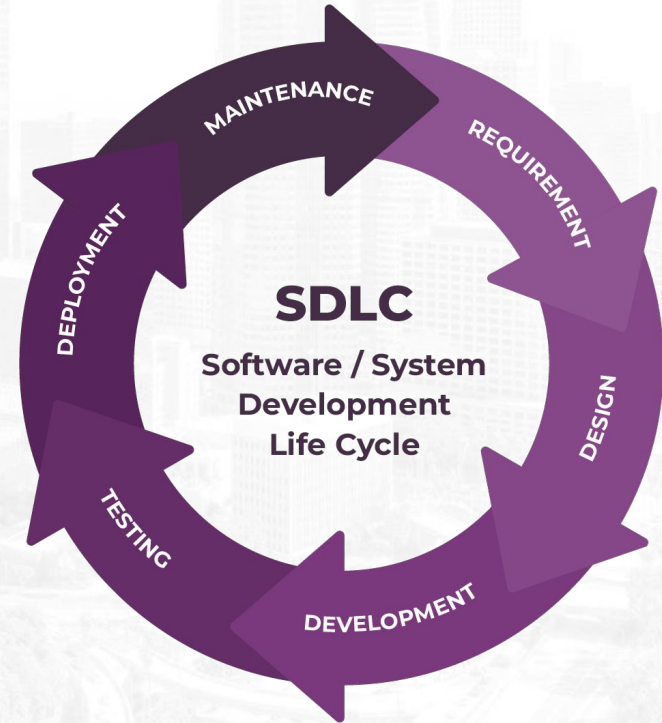
Prototype



Agile Scrum

# Apa itu SDLC

SDLC (*Software Development Life Cycle*) adalah proses pembuatan dan pengubahan sistem serta model dan metodologi yang digunakan untuk mengembangkan sistem rekayasa perangkat lunak.



# FUNGSI SDLC



## **Mengorganisir Pengembangan**

SDLC membantu mengorganisir langkah-langkah pengembangan perangkat lunak agar menjadi lebih terstruktur, mudah dikelola, dan terukur.



## **Mengontrol Proyek**

SDLC membantu dalam mengontrol proyek pengembangan perangkat lunak, termasuk mengidentifikasi risiko, mengelola perubahan, dan menjaga jadwal dan anggaran proyek.



## **Meningkatkan Kualitas**

SDLC memastikan bahwa perangkat lunak yang dihasilkan memiliki kualitas yang tinggi melalui tahap-tahap pengujian dan validasi yang berulang.



## **Meningkatkan Kerjasama**

SDLC membantu mengintegrasikan berbagai tim dan disiplin ilmu yang terlibat dalam pengembangan perangkat lunak, seperti pengembang, pengujian, analis bisnis, dan manajemen proyek.



# TUJUAN SDLC

1

SDLC membantu memastikan bahwa perangkat lunak yang dihasilkan memiliki kualitas tinggi dengan menjalani tahap-tahap pengujian dan validasi yang komprehensif.

3

**Mengoptimalkan Biaya dan Waktu:** SDLC membantu dalam pengaturan anggaran dan jadwal proyek, sehingga meminimalkan biaya dan memastikan penyelesaian tepat waktu.

5

**Memastikan Kepuasan Pengguna:** Dengan merancang perangkat lunak sesuai dengan kebutuhan pengguna dan mengujinya dengan baik, SDLC membantu meningkatkan kepuasan pengguna akhir.

2

**Meminimalkan Risiko:** Dengan merencanakan dan mengelola proses pengembangan, SDLC membantu mengidentifikasi dan mengurangi risiko yang terkait dengan proyek perangkat lunak.

4

**Meningkatkan Transparansi:** SDLC membantu dalam komunikasi yang lebih baik antara tim pengembangan, manajemen proyek, dan pemangku kepentingan lainnya.

# Software Development Life Cycle (SDLC)

- ☒ Introduction SDLC
- ☐ Waterfall
- ☐ Prototype
- ☐ Agile Scrum

# Software Development Life Cycle (SDLC)

☐ Introduction SDLC



☐ **Waterfall**

☐ Prototype

☐ Agile Scrum



# Waterfall

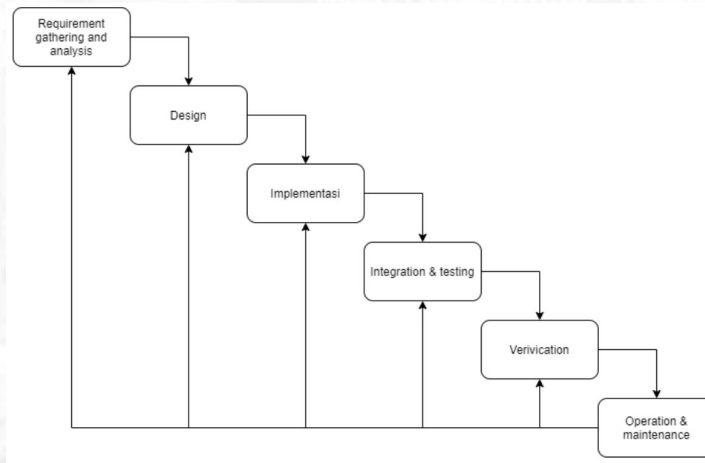
Metode waterfall adalah metode kerja tertua yang menekankan fase-fase yang berurutan dan sistematis. Disebut waterfall karena proses mengalir satu arah “ke bawah” seperti air terjun. Metode waterfall ini harus dilakukan secara berurutan sesuai dengan tahap yang ada.

## 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.

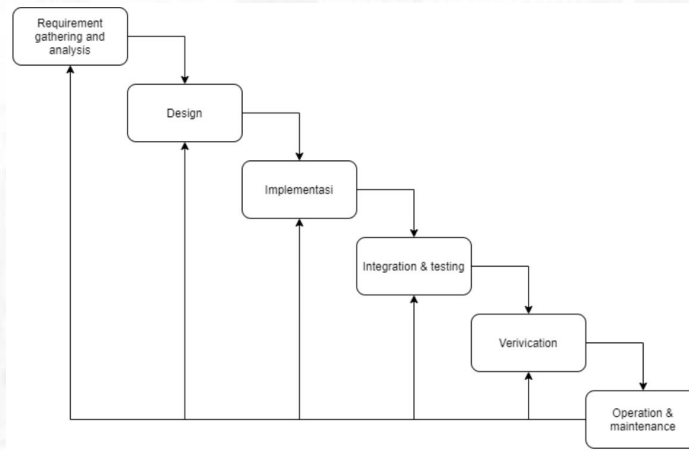
## Integration & Testing

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.

# Waterfall

## 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 dan kekurangan Metode Waterfall

### Kelebihan

1. Memiliki proses yang teratur, 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.

### Kekurangan

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. Masalah tidak bisa diselesaikan sebelum mencapai tahap maintenance.

# Software Development Life Cycle (SDLC)

☐ Introduction SDLC

☒ **Waterfall**

☐ Prototype

☐ Agile Scrum

# Software Development Life Cycle (SDLC)

☐ Introduction SDLC

☐ Waterfall



☐ **Prototype**

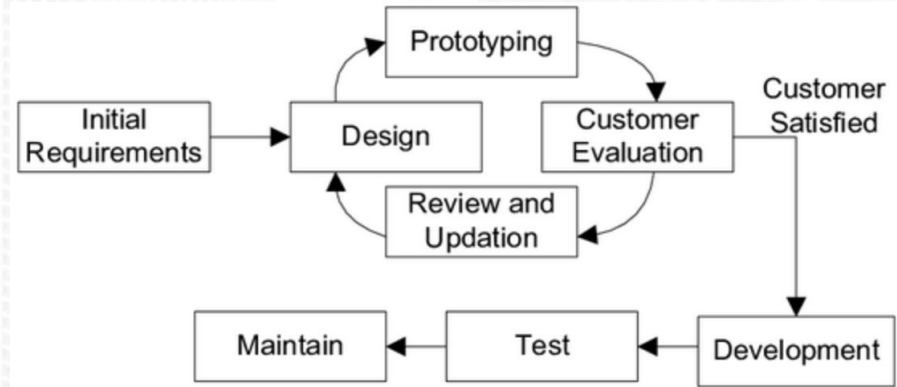
☐ Agile Scrum

# Prototype

Metode prototipe adalah pendekatan dalam SDLC (Software Development Life Cycle) di mana perangkat lunak dikembangkan secara iteratif dengan membuat prototipe atau versi awal dari perangkat lunak yang berfungsi.

Prototipe ini digunakan untuk menggambarkan fitur dan fungsi yang diharapkan dari perangkat lunak kepada pengguna akhir dan pemangku kepentingan lainnya.

Tujuan utama dari metode prototipe adalah untuk mendapatkan umpan balik dini dan memastikan bahwa perangkat lunak yang dikembangkan memenuhi kebutuhan dan harapan pengguna sebelum versi finalnya dibangun.





# Prototype

## Kelebihan

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

## Kekurangan

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

# Software Development Life Cycle (SDLC)

- ☐ Introduction SDLC
- ☐ Waterfall
- ☒ **Prototype**
- ☐ Agile Scrum

# Software Development Life Cycle (SDLC)

☐ Introduction SDLC

☐ Waterfall

☐ Prototype



☐ **Agile Scrum**

# AGILE



Produk yang berfungsi dengan baik dan dianggap sebagai pendekatan pengembangan yang sangat realistis. Itulah mengapa metode Agile lebih sering digunakan untuk pengembangan software.

Nilai utama dalam Development Agile adalah memungkinkan tim untuk memberikan nilai lebih cepat, dengan kualitas dan prediksi yang lebih baik, dan bakat yang lebih besar untuk merespons perubahan. Jadi, Bisa disimpulkan jika Agile merupakan model development jangka pendek yang memerlukan adaptasi cepat dan pengembangan terhadap perubahan dalam bentuk apapun. Nah, setelah kita tahu bahwa Agile itu cocok untuk pengembangan software, ternyata pada prakteknya ada banyak jenis model turunan dari metode Agile, salah satunya adalah SCRUM.

## Keuntungan Agile

- Keterlibatan Stakeholder
- Transparansi
- Release Awal dan Mudah Diprediksi
- Biaya & Jadwal Bisa Ditentukan
- Memungkinkan Perubahan
- Berfokus Pada Nilai Bisnis
- Fokus pada Pengguna
- Meningkatkan Kualitas

## Proses Dalam Sprint

# Scrum

**Scrum adalah** kerangka kerja yang digunakan untuk mengimplementasikan pengembangan Agile. Scrum lebih mengutamakan pada kekuatan kolaborasi tim dalam development software.

### Sprint

Sebuah sprint dimulai dengan tim menganalisa user story dari release backlog ke dalam sprint backlog. Idealnya sprint planning dipimpin oleh scrum master.

### Daily Scrum

Idealnya daily scrum dipimpin oleh scrum master, berlangsung sekitar 15 menit. Selama Daily Scrum, para anggota akan membahas pekerjaan yang diselesaikan pada hari sebelumnya, pekerjaan yang direncanakan untuk hari berikutnya dan masalah yang dihadapi selama sprint. Daily Scrum juga berguna untuk memonitor progress dari kemajuan tim setiap harinya.

### Sprint Retrospective

Kegiatan ini juga dipandu oleh scrum master, berlangsung sekitar 2-4 jam dan membahas apa yang telah dicapai tim dalam sprint terakhir dan pelajaran apa yang didapat.



# Scrum Artific

## User Stories

Penjelasan singkat tentang fungsionalitas fitur yang akan dikembangkan.

## Product Backlog

Kumpulan fitur yang diambil untuk produk scrum. Product owner menyiapkan dan memelihara backlog produk, scrum team dapat menambahkan atau melengkapi dengan persetujuan dari product owner.

## Sprint Backlog

Kumpulan fitur atau daftar tugas yang harus diselesaikan dalam sprint. Backlog dikelola oleh tim namun penentu prioritasnya adalah product owner.

## Release Backlog

Kerangka waktu di mana jumlah iterasi selesai. Product owner berkoordinasi dengan scrum master untuk memutuskan fitur mana yang harus ditargetkan untuk rilis lebih dulu

## Sprint

Periode waktu yang ditentukan untuk menyelesaikan fitur, diputuskan oleh product owner dan scrum team, biasanya 1-4 minggu

## Burndown chart

Bagan burn-down mewakili kemajuan keseluruhan dari pekerjaan yang sedang berlangsung dan pekerjaan yang diselesaikan selama proses berlangsung.

# Software Development Life Cycle (SDLC)

- ☐ Introduction SDLC
- ☐ Waterfall
- ☐ Prototype
- ☒ **Agile Scrum**



**Terimakasih**