

Muhammad Fahmi Zulfikar

Instalasi beberapa tools dibawah ini :

- Git
- Visual Studio Code
- Browser

Buat sebuah summary untuk apa yang sudah dipelajari dalam materi pengenalan software engineer sebelumnya.

Full Stack Development: pengembangan seluruh aplikasi baik dari sisi Front-End, Back-End, dan bahkan Client Side.

Scope Full Stack Development:

- FrontEnd Development
- BackEnd Development
- Database Management
- Version Control and Collaboration
- Mobile Development.

## **Front-End**

Dasar-dasar Front-End Web:

- HTML
- CSS
- Javascript

Framework Front-End:

- React
- Vue.Js
- AngularJS

## **Back-End**

Dasar-dasar Back-End Web:

- Server-Side Program Language (Python, Java, PHP)
- Server Framework (Flask, Express.Js, Laravel)
- Database Management (SQL(MySQL, PostgreSQL) dan NoSQL(MongoDB, Firebase))

Framework Back-End:

- Express.Js
- Laravel
- Spring
- Ruby on Rails

## **Database Management**

Dasar-dasar Database Management

- Database Management System (DBMS)
- Tipe Database (Relational SQL dan Non-Relational NoSQL)
- Bahasa Query (SELECT, INSERT, UPDATE, DELETE)

Framework Database Management yang populer:

- ORACLE
- PostgreSQL
- MySQL
- MongoDB
- Redis

## **Mobile Development**

Dasar-dasar Mobile Development

- Platform Mobile (Android Java atau Kotlin, iOS Swift atau Objective-C)
- IDE (Integrated Development Environment)

Framework Mobile Development yang populer:

- React Native
- Flutter

## **Pengembangan Aplikasi End-to-End**

Tahap-tahap:

1. Perencanaan dan Analisis (Pengumpulan kebutuhan, pemahaman tujuan aplikasi, Sasaran Pengguna, Lingkungan operasional)
2. Desain (UI, UX, Arsitektur Teknologi, Database, Framework)
3. Pengembangan Front-End (HTML, CSS, JavaScript, Framework Front-End)
4. Pengembangan Back-End (Python, Ruby, Java, Framework Back-End)
5. Integrasi dan Pengujian (Integrasi dengan API dan uji fitur dan bug fix)
6. Pemeliharaan dan Peningkatan (Bug fix, memperbarui fitur)

## **Collaboration dan Version Control**

Version Control: Sistem untuk melacak perubahan versi aplikasi selama pengembangan,

Contoh: Git, Mercurial

## **Tools Full Stack Dev yang Populer**

IDE - Code Editor

- Visual Studio Code (VSCode)

Version Control

1. Repository
  - GitHub
  - GitLab
  - Bitbucket
2. Git Tools
  - Sourcetree
  - GitLens

DBMS

- PostgreSQL
- MySQL
- ORACLE

- mongoDB
- Redis

#### API

- Postman
- Swagger

Dan masih banyak lagi mulai dari Tests and Debugging hingga desain UI/UX

### **Software Development Life Cycle (SDLC)**

proses yang terstruktur untuk mengembangkan perangkat lunak dari awal hingga selesai.

#### **Siklus SDLC**

1. Perencanaan dan Analisis (Identifikasi masalah atau kebutuhan)
2. Desain (perancangan secara rinci berdasarkan persyaratan)
3. Pengembangan (Implementasi rancangan perangkat lunak)
4. Pengujian (Memastikan perangkat lunak berfungsi sesuai persyaratan)
5. Penerapan (Implementasi rancangan perangkat lunak)
6. Pemeliharaan (memperbaiki bug dan meningkatkan fitur)

#### **Model-model SDLC**

- Waterfall Model (Pengerjaan hanya linier dan berurutan)
- V-Shaped Model (Menekankan pada pengujian)
- Prototype Model (Contoh awal sebelum mengembangkan versi final)
- Spiral Model (Menggabungkan elemen model spiral dengan pendekatan inkremental)
- Iterative Incremental Model (Pengulangan siklus pembangunan dan peningkatan perangkat lunak)
- Big Bang Model (semua tahapan pengembangan dilakukan tanpa perencanaan yang detail)
- Agile Model (berfokus pada pengiriman perangkat lunak secara berkala dan inkremental)

### **Design Thinking**

#### **Steps:**

1. Empathize: Understand User Needs
2. Define: Define the Problem
3. Ideate: Generate Ideas
4. Prototype: Build Quick and Iterative Solutions
5. Test: Gather User Feedback
6. Implement: Develop the Software

### **Basic Git**

Git: Sistem kontrol versi terdistribusi untuk:

- Melacak perubahan dalam kode mereka
- Berkolaborasi dengan anggota tim
- Mengelola revisi kode secara efektif.

Beberapa command git:

Command	Windows	Linux / macOS	Description
List Files	'dir'	'ls'	Menampilkan daftar file pada direktori
Change Directory	'cd'	'cd'	Mengganti direktori aktif
Make Directory	'mkdir'	'mkdir'	Membuat direktori baru
Remove Directory	'rmdir'	'rm -r'	Menghapus direktori
Delete Files	'del'	'rm'	Menghapus file
Copy Files	'copy'	'cp'	Salin file
Move/Rename	'move'	'mv'	Memindahkan file
Display File	'type'	'cat'	Menampilkan konten file
Display Text	'echo'	'echo'	Menampilkan teks
List Processes	'tasklist'	'ps'	List Running Processes
Terminate Processes	'taskkill'	'kill'	Memberhentikan proses

#### Jenis Version Control:

- Sistem Kontrol Versi Terpusat (Centralized Version Control System)
  - Hanya ada satu repository sentral yang berfungsi sebagai master untuk menyimpan seluruh history proyek.
  - Perubahan dilakukan ke repository master.
- Sistem Kontrol Versi Terdistribusi (Distributed Version Control System)
  - Setiap anggota tim memiliki salinan lengkap dari seluruh repository.
  - Setiap anggota memiliki salinan lengkap history perubahan, tidak hanya salinan terbaru.