

Nama : Danang Widiyanto

Kelompok : 9

HomeWork Software Developer

## **SUMMARY**

### **Fullstack Development**

Pengembangan Full Stack (Full Stack Development) adalah pendekatan yang mencakup pengembangan seluruh aplikasi dari sisi depan (front-end) hingga sisi belakang (back-end), dan dalam beberapa kasus, hingga sisi klien (client-side). Ini mencakup beberapa konsep dan teknologi utama:

Definisi dan Scope Full Stack Development: Ini mencakup pembangunan antarmuka pengguna dengan HTML, CSS, dan JavaScript, penggunaan framework front-end seperti React atau Angular, pengembangan server dan aplikasi di sisi belakang menggunakan bahasa pemrograman server-side seperti Node.js atau Java, desain dan manajemen database, integrasi front-end dan back-end melalui API, serta penggunaan sistem pengendalian versi seperti Git.

- **Front-End Development:** Ini melibatkan pembuatan tampilan dan interaksi pengguna menggunakan HTML, CSS, dan JavaScript, serta penggunaan framework front-end.
- **Back-End Development:** Ini fokus pada pengembangan sisi server aplikasi, pengolahan permintaan dari sisi depan, dan penggunaan bahasa pemrograman server-side.
- **Database Management:** Ini mencakup desain dan pengelolaan basis data menggunakan teknologi database seperti MySQL atau MongoDB.

- **Integration of Front-End and Back-End:** Ini melibatkan koneksi antara komponen front-end dan back-end melalui API untuk berkomunikasi dengan server dan database.
- **Version Control and Collaboration:** Pengembang Full Stack menggunakan sistem pengendalian versi seperti Git untuk mengelola perubahan kode dan kolaborasi dalam tim pengembang.
- **Mobile Development:** Beberapa pengembang Full Stack juga memiliki kemampuan untuk mengembangkan aplikasi mobile menggunakan framework seperti React Native atau Flutter.
- **Dasar-Dasar Frontend Web Development:** Ini mencakup HTML, CSS, dan JavaScript sebagai bahasa utama dalam pembuatan antarmuka web.
- **Popular Frontend Framework:** Beberapa framework front-end yang umum digunakan dalam pengembangan Full Stack.
- **Dasar-Dasar Backend Development:** Ini melibatkan pengolahan permintaan dari pengguna, pengelolaan dan penyimpanan data di database, serta respons kepada pengguna.
- **Popular Backend Framework:** Beberapa framework back-end yang umum digunakan dalam pengembangan Full Stack.
- **Dasar-Dasar Database Management:** Ini mencakup konsep dan teknik pengelolaan data dalam aplikasi, termasuk tipe database dan bahasa query SQL.
- **Popular Database Management:** Beberapa jenis database yang umum digunakan dalam pengembangan Full Stack.

- **Dasar-Dasar Mobile Development:** Ini mencakup konsep pengembangan aplikasi mobile untuk berbagai platform.
- **Popular Mobile Framework:** Beberapa framework mobile yang digunakan dalam pengembangan Full Stack.
- **Version Control:** Penggunaan sistem pengendalian versi untuk melacak perubahan kode dan kolaborasi dalam tim.
- **Manfaat Version Control untuk Berkolaborasi:** Manfaat penggunaan sistem pengendalian versi dalam kolaborasi tim pengembang.
- **Tools sets Sebagai Full Stack Developer:** Berbagai alat yang digunakan oleh pengembang Full Stack, termasuk IDE, Version Control, DBMS, API, alat pengujian, dan lain-lain.
- **Roadmap Full Stack Development:** Rencana pengembangan karir sebagai Full Stack Developer.

## **SDLC & Design Thinking Implementation**

SDLC (Siklus Hidup Pengembangan Perangkat Lunak) adalah serangkaian proses yang terstruktur dan metodologi yang digunakan untuk mengembangkan perangkat lunak dari awal hingga selesai. Ini terdiri dari tahap-tahap seperti Perencanaan dan Analisis, Desain, Pengembangan, Pengujian, Penerapan, dan Pemeliharaan.

Beberapa model SDLC yang populer termasuk:

- **Model Air Terjun (Waterfall Model):** Model linier dan berurutan, cocok untuk proyek dengan persyaratan yang jelas dan stabil.
- **Model Berbentuk V (V-Shaped Model):** Model yang menekankan pengujian yang ketat dan sesuai dengan model Air Terjun.

- Model Prototipe (Prototype Model): Model yang menciptakan prototipe atau contoh awal sebelum pengembangan versi final.
- Model Spiral (Spiral Model): Model yang menggabungkan elemen model spiral dengan pendekatan inkremental, cocok untuk proyek besar dan kompleks.
- Model Iteratif Incremental (Iterative Incremental Model): Model yang menggabungkan pengembangan dalam tahapan kecil dan cocok untuk proyek dengan anggaran terbatas.
- Model Ledakan Besar (Big Bang Model): Model yang kurang terstruktur dan cocok untuk proyek kecil atau prototyping.
- Model Agile (Agile Model): Model yang berfokus pada pengiriman perangkat lunak secara berkala dan inkremental dengan fleksibilitas terhadap perubahan persyaratan.

Design Thinking Implementation adalah proses pengembangan yang berfokus pada pemahaman mendalam pengguna akhir dan kebutuhan mereka. Tahap-tahapnya mencakup Empathize (memahami kebutuhan pengguna), Define (menentukan masalah), Ideate (menghasilkan ide), Prototype (membangun solusi iteratif), Test (mengumpulkan umpan balik pengguna), dan Implement (mengembangkan perangkat lunak).

Integrasi Design Thinking dalam SDLC memastikan pengembangan perangkat lunak lebih berorientasi pada pengguna, intuitif, dan mampu memenuhi kebutuhan pengguna serta tujuan bisnis. Tahapan yang iteratif dalam Design Thinking memungkinkan perangkat lunak untuk beradaptasi dengan perubahan kebutuhan pengguna dan dinamika pasar.

### **Basic Git & Collaborating Using Git**

- Git adalah sistem kontrol versi terdistribusi yang digunakan untuk melacak dan mengelola perubahan dalam kode sumber proyek.
- Git memungkinkan pengembang untuk bekerja secara kolaboratif, mengelola repositori, dan mengelola revisi kode.

- Langkah-langkah dasar untuk memulai dengan Git termasuk menginstal Git di sistem Anda dan memahami perintah-perintah dasar seperti git init, git clone, git status, git add, git commit, git push, git pull, git branch, git checkout, git merge, git log, git remote, git fetch, git diff, git reset, dan lainnya.

## **Terminal and IDE**

- Terminal adalah antarmuka teks yang memungkinkan pengguna untuk berinteraksi dengan sistem operasi menggunakan perintah teks.
- Sejarah singkat terminal melibatkan perkembangan dari terminal fisik hingga terminal emulator di sistem operasi modern.
- Command line dasar melibatkan perintah-perintah teks yang digunakan untuk mengelola berkas dan sistem.

## **Installing, initializing and committing GIT**

- Git dapat diinstal pada berbagai sistem operasi seperti Windows, Linux, dan MacOS.
- Proses instalasi Git melibatkan mengunduh installer dan mengikuti langkah-langkah yang diberikan.
- Setelah menginstal Git, Anda dapat menginisialisasi repositori Git baru menggunakan git init dan melakukan commit pertama dengan pesan commit yang deskriptif.

## **Collaborating Using Git**

- Kolaborasi menggunakan Git melibatkan beberapa pengguna yang bekerja pada repositori yang sama.
- Peserta kolaborasi dapat mengklon repositori, membuat branch untuk pekerjaan mereka, melakukan perubahan, mengirimkan perubahan ke repositori jarak jauh, menangani konflik jika terjadi, membuat permintaan tarik (pull request), dan menggabungkan perubahan ke branch utama.

Simulasi kolaborasi menggunakan Git melibatkan pengaturan dasar repository, persiapan lokal, pembuatan branch, pembuatan perubahan, pengiriman perubahan ke remote, penanganan konflik, permintaan tarik, dan penggabungan.

Dengan menggunakan Git, pengembang dapat bekerja secara efisien dan kolaboratif pada proyek perangkat lunak, melacak perubahan, dan memastikan kualitas kode yang lebih baik.