

Nama GLORY HARDY FEBRIANDO
NIM 105223013
THT MODUL 6
TUGAS Analisis Perbandingan

1. Jelaskan perbedaan utama yang Anda rasakan saat membangun layout dan komponen menggunakan Bootstrap 5 vs Tailwind CSS. Manakah yang terasa lebih cepat untuk setup awal? Manakah yang memberi Anda kontrol lebih besar atas styling?

Perbedaan utama yang saya rasakan adalah pada filosofi dan tingkat kendali saat membangun:

Bootstrap 5 (Pendekatan Berbasis Komponen) Prosesnya terasa seperti merakit blok-blok yang sudah jadi. Saya lebih banyak menggunakan kelas-kelas komponen yang sudah memiliki style utuh, seperti `.card`, `.btn`, dan `.form-control`. Fokus saya adalah memilih "komponen apa" yang akan digunakan dan menyusunnya dalam sistem grid (`.row` dan `.col`).

Bootstrap terasa lebih cepat untuk setup awal.

Alasannya adalah setup standar Bootstrap hanya melibatkan penyalinan link CDN CSS dan JS ke dalam file HTML. Ini bisa langsung digunakan tanpa instalasi apa pun.

Tailwind CSS memberi Anda kontrol yang jauh lebih besar atas styling.

Alasannya, dengan Tailwind, Anda membangun style dari nol menggunakan kelas-kelas utilitas kecil. Jika Anda ingin bayangan, Anda tambahkan `shadow-lg`. Jika Anda ingin border bulat, Anda tambahkan `rounded-full`. Anda memiliki kendali penuh atas setiap properti CSS langsung dari HTML.

Tailwind CSS (Pendekatan Utility-First) Prosesnya terasa seperti membangun komponen dari nol menggunakan "alat-alat" kecil yang spesifik. Tidak ada kelas `.card` bawaan; saya harus membuatnya sendiri dengan menggabungkan banyak kelas utilitas seperti `bg-white`, `rounded-lg`, dan `shadow-md`. Hal yang sama berlaku untuk formulir, di mana setiap input dan button harus di-styling secara manual dari n

Ukuran File & Kustomisasi:

- Bandingkan tampilan file HTML. File mana yang memiliki kode HTML lebih bersih (sedikit kelas) dan file mana yang lebih "verbose" (banyak kelas)?

File yang Lebih "Bersih" (Sedikit Kelas): `Jelas portofolio-bootstrap.html`.

Karena Bootstrap menggunakan kelas berbasis komponen, saya hanya perlu menulis satu kelas seperti `.card` atau `.form-control` untuk menerapkan banyak style sekaligus.

File yang Lebih "Verbose" (Banyak Kelas): Jelas portofolio-tailwind.html. Karena ini adalah pendekatan utility-first, setiap style (seperti bg-white, rounded-lg, shadow-md, p-4, font-bold, text-blue-500, dll.) harus ditambahkan sebagai kelasnya sendiri. Ini membuat file HTML-nya dipenuhi oleh jauh lebih banyak kelas untuk membangun komponen yang secara visual terlihat sama.

2. Jelaskan bagaimana pendekatan kustomisasi berbeda antara kedua framework berdasarkan Modul 6

Berdasarkan Modul 6:

Kustomisasi Bootstrap 5:

Pendekatan kustomisasi di Bootstrap berfokus pada penggunaan komponen yang sudah jadi. menggunakan kelas seperti .card, .btn, dan .form-control yang sudah memiliki tampilan bawaan.

Kustomisasi lebih lanjut terbatas pada penggunaan kelas utilitas yang disediakan (seperti .text-primary atau .fw-bold) atau Anda harus menulis styling manual secara terpisah untuk menimpa (override) desain default-nya.

Kustomisasi Tailwind CSS:

Pendekatan kustomisasi di Tailwind adalah proses pengembangan itu sendiri. Modul ini menjelaskan bahwa membangun antarmuka pengguna (UI) dari nol. tidak diberi komponen jadi; Anda menggabungkan banyak kelas utilitas untuk menciptakan komponen sendiri.

Sebagai contoh, untuk membuat input formulir, Anda tidak menggunakan satu kelas, melainkan banyak kelas utilitas seperti .shadow, .border, .rounded, .w-full, .py-2, .px-3, dll..

3. Berdasarkan pengalaman studi kasus ini, kapan Anda akan memilih menggunakan Bootstrap 5? Dan kapan Anda akan memilih menggunakan Tailwind CSS untuk proyek di masa depan?

Berdasarkan pengalaman studi kasus ini:

Kapan Saya Akan Memilih Bootstrap 5

Saya akan memilih Bootstrap 5 untuk proyek-proyek di mana kecepatan pengembangan dan konsistensi adalah prioritas utama, dan desain yang unik bukanlah tuntutan utama.

~Prototyping Cepat: Jika klien meminta prototype fungsional dalam satu hari, Bootstrap adalah pemenangnya. Saya bisa merakit halaman menggunakan komponen yang sudah jadi.

~Proyek Internal: Sangat ideal untuk membangun dashboard admin atau panel backend, di mana fungsionalitas lebih penting daripada tampilan yang unik.

~Deadline Ketat: Ketika saya perlu membuat situs yang terlihat profesional dan cukup bagus dengan cepat, tanpa harus pusing memikirkan setiap detail styling.

Kapan Saya Akan Memilih Tailwind CSS

Saya akan memilih Tailwind CSS untuk proyek-proyek yang membutuhkan kontrol penuh atas desain dan menuntut tampilan yang unik.

~Desain Kustom (Pixel-Perfect): Ketika saya bekerja dengan desainer yang memberikan desain spesifik (seperti di Figma) dan saya harus mencocokkannya dengan sempurna. Saya tidak terkunci pada tampilan komponen bawaan.

~Interaktivitas & Kustomisasi: Ketika saya perlu menerapkan banyak style kustom, hover, dan efek transisi (seperti efek story ring pada foto profil atau kartu galeri yang terangkat). Melakukan ini di Tailwind jauh lebih mudah daripada harus menulis CSS tambahan untuk menimpa Bootstrap.

~Proyek Jangka Panjang: Untuk proyek yang akan sering diperbarui, pendekatan utility-first membuatnya lebih mudah untuk dimodifikasi tanpa merusak style di bagian lain.