

Analisis Perbandingan: Bootstrap 5 vs Tailwind CSS

1. Proses Pengembangan

Perbedaan utama yang saya rasakan adalah pada pendekatan membangun komponen.

- Menggunakan **Bootstrap 5** terasa seperti "merakit" dari komponen yang sudah jadi. Saya hanya perlu memanggil kelas `.card`, `.btn`, dan `.form-control`, dan Bootstrap langsung menyediakan styling-nya. Sedangkan menggunakan **Tailwind CSS**, saya harus "membangun" setiap komponen dari nol menggunakan kelas-kelas utilitas. Misalnya, untuk membuat card, saya harus mendefinisikan sendiri `bg-white`, `rounded-lg`, `shadow-md`, dan `overflow-hidden`.
- **Lebih Cepat (Setup Awal):** **Bootstrap 5** terasa jauh lebih cepat untuk setup awal dan membuat prototipe yang fungsional.
- **Kontrol Lebih Besar:** **Tailwind CSS** memberi saya kontrol yang jauh lebih besar atas styling. Saya bisa mengatur setiap detail kecil tanpa harus menulis CSS kustom atau "melawan" style default dari framework.

2. Ukuran File & Kustomisasi

Tampilan File HTML:

- File `portofolio-bootstrap.html` memiliki kode HTML yang lebih "**bersih**" (sedikit kelas). Kelasnya semantik dan mudah dibaca (contoh: `.card-body`).
- File `portofolio-tailwind.html` jauh lebih "**verbose**" (banyak kelas). Satu elemen `<input>` saja bisa memiliki 10-15 kelas utilitas hanya untuk styling.

Pendekatan Kustomisasi (berdasarkan Modul 6):

- **Bootstrap:** Kustomisasi utamanya dilakukan dengan menimpa (override) variabel Sass bawaan. Jika ingin mengubah warna `.btn-primary` secara global, saya harus mengubah variabel `$primary` di file Sass.
- **Tailwind:** Kustomisasi dilakukan melalui file `tailwind.config.js`. Saya bisa menambahkan warna baru, font, atau mengubah breakpoint dengan mudah tanpa menimpa file inti framework.

3. Kesimpulan

Berdasarkan pengalaman studi kasus ini:

- Saya akan memilih **Bootstrap 5** ketika saya perlu membuat proyek dengan cepat di mana desain yang unik bukanlah prioritas utama. Contohnya, membuat dashboard admin, halaman internal, atau prototipe cepat. Komponen siap pakai sangat menghemat waktu.
- Saya akan memilih **Tailwind CSS** ketika saya mengerjakan proyek yang membutuhkan desain kustom yang spesifik (misalnya, dari desainer UI/UX). Tailwind memungkinkan saya membangun desain tersebut dengan presisi tanpa harus "melawan" style bawaan, dan menjaga file CSS tetap kecil (saat menggunakan PurgeCSS) karena hanya utilitas yang dipakai yang akan di-bundle.