**Eksperimen 5 – WEB DEVELOPMENT**

*Laporan ini disusun untuk memenuhi Tugas Mata Kuliah   
Web Development (Teori)*

A logo with a blue and orange design

Description automatically generated

**Disusun oleh:**

Nama : Yusuf

NIM : 221524062

Kelas : 3B – D4

Program Studi : D4 Teknik Informatika

**Mata Kuliah:**

Nama Mata Kuliah : Sistem Informasi

Kode Mata Kuliah: : 21TI3002

**SARJANA TERAPAN PROGRAM STUDI TEKNIK INFORMATIKA**

**TEKNIK KOMPUTER DAN INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2024**

Daftar Isi

[Pendahuluan 3](#_Toc176208797)

[Identifikasi Masalah 3](#_Toc176208798)

[Tujuan Eksperimen 3](#_Toc176208799)

[Metodologi Eksperimen 3](#_Toc176208800)

[Pelaksanaan Eksperimen 4](#_Toc176208801)

[Analisis Hasil Eksperimen 4](#_Toc176208802)

[Kesimpulan dan Saran 5](#_Toc176208803)

# Pendahuluan

Responsive design adalah pendekatan dalam desain web yang memungkinkan tampilan situs web untuk menyesuaikan secara dinamis dengan berbagai ukuran layar dan perangkat. Dalam era digital saat ini, di mana pengguna mengakses website melalui berbagai perangkat dengan ukuran layar yang bervariasi, responsive design menjadi sangat penting untuk memastikan pengalaman pengguna yang optimal.

Untuk mengimplementasikan responsive design, ada berbagai pendekatan yang dapat digunakan, salah satunya adalah dengan menggunakan vanilla CSS, di mana pengembang menulis CSS dari nol. Alternatif lain adalah dengan menggunakan framework seperti Tailwind CSS, yang menyediakan utility classes untuk mempercepat dan memudahkan proses pengembangan.

# Identifikasi Masalah

Pada proyek web development, salah satu tantangan utama adalah memastikan bahwa tampilan website tetap konsisten dan mudah digunakan di berbagai perangkat, dari ponsel hingga desktop. Masalah yang diidentifikasi adalah menentukan pendekatan mana yang lebih efektif dan efisien antara menggunakan vanilla CSS atau Tailwind CSS dalam mengimplementasikan responsive design.

# Tujuan Eksperimen

Tujuan dari eksperimen ini adalah untuk:

* Membandingkan efisiensi pengembangan responsive design antara vanilla CSS dan Tailwind CSS.
* Menganalisis perbedaan dalam jumlah kode yang diperlukan, kemudahan penggunaan, dan fleksibilitas dari kedua pendekatan tersebut.
* Menentukan pendekatan yang lebih baik dalam konteks waktu pengembangan dan pemeliharaan proyek.

# Metodologi Eksperimen

Eksperimen dilakukan dengan membangun sebuah layout sederhana yang responsif, yang memiliki dua kolom pada layar lebar dan satu kolom pada layar kecil. Proses ini dilakukan dua kali, satu kali menggunakan vanilla CSS dan satu kali menggunakan Tailwind CSS. Setiap langkah didokumentasikan, dan hasilnya dianalisis berdasarkan:

* Jumlah kode yang ditulis.
* Kemudahan dalam mengimplementasikan perubahan gaya.
* Fleksibilitas dalam penyesuaian tampilan.
* Waktu yang dibutuhkan untuk menyelesaikan tugas.

# Pelaksanaan Eksperimen

1. **Vanilla CSS:**
   * Menulis CSS untuk membuat container menggunakan flexbox, dengan kolom-kolom yang berubah menjadi baris pada layar yang lebih besar menggunakan media queries.
   * Mengatur gaya untuk setiap item dalam container, termasuk padding, margin, dan background.
   * Memastikan gaya yang diatur tetap konsisten di berbagai ukuran layar melalui media queries.
2. **Tailwind CSS:**
   * Menggunakan utility classes dari Tailwind untuk membuat container responsif dengan flexbox, mengatur perubahan dari kolom ke baris menggunakan kelas md:flex-row.
   * Menambahkan utility classes untuk padding, margin, background, dan elemen styling lainnya.
   * Mengamati perubahan responsif tanpa perlu menulis media queries secara manual.

# Analisis Hasil Eksperimen

* **Jumlah Kode:**
  + **Vanilla CSS:** Memerlukan lebih banyak kode karena harus menulis media queries dan pengaturan flexbox secara manual.
  + **Tailwind CSS:** Jauh lebih sedikit kode yang perlu ditulis, karena sebagian besar styling sudah tersedia dalam bentuk utility classes.
* **Kemudahan Penggunaan:**
  + **Vanilla CSS:** Membutuhkan pemahaman mendalam tentang CSS, flexbox, dan media queries.
  + **Tailwind CSS:** Lebih mudah dan cepat, terutama bagi pengembang yang sudah terbiasa dengan utility-first framework.
* **Fleksibilitas:**
  + **Vanilla CSS:** Memberikan kontrol penuh atas setiap elemen, tetapi lebih kompleks dan memakan waktu.
  + **Tailwind CSS:** Sangat modular dan mudah digunakan, namun mungkin memerlukan penyesuaian tambahan untuk kebutuhan yang sangat spesifik.
* **Waktu Pengembangan:**
  + **Vanilla CSS:** Memakan waktu lebih lama karena harus menulis dan menguji setiap bagian kode secara manual.
  + **Tailwind CSS:** Lebih cepat karena banyak gaya sudah disediakan dalam bentuk kelas yang siap pakai.

# Kesimpulan dan Saran

* **Kesimpulan:** Eksperimen menunjukkan bahwa Tailwind CSS menawarkan efisiensi yang lebih tinggi dalam pengembangan responsive design dibandingkan dengan vanilla CSS. Tailwind memungkinkan pengembang untuk menyelesaikan tugas dengan lebih cepat dan menulis lebih sedikit kode, tanpa mengorbankan kualitas tampilan akhir. Namun, vanilla CSS masih relevan dan berguna dalam situasi di mana kontrol penuh atas setiap detail desain sangat penting.
* **Saran:** Untuk proyek-proyek dengan timeline yang ketat dan kebutuhan desain yang tidak terlalu spesifik, disarankan untuk menggunakan Tailwind CSS. Namun, untuk proyek yang memerlukan kustomisasi yang mendalam dan detail, vanilla CSS bisa menjadi pilihan yang lebih baik. Pengembang sebaiknya mempertimbangkan kompleksitas proyek, kebutuhan tim, dan keahlian yang dimiliki sebelum memilih pendekatan yang paling sesuai.