**EKSPLORASI PROBLEM YANG TERDAPAT DI WEB DEVELOPMENT**

*Z-INDEX*

*Laporan ini disusun untuk memenuhi tugas mata kuliah Web Development*



Disusun oleh :

Muhammad Rizki N. 221524051

Kelompok 5

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**PROGRAM STUDI D4 TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2024**

**Laporan Hasil Analisis Eksperimen Terkait Permasalahan `z-index` dan Kantar Elemen Dalam Web Development:**

---

Laporan Hasil Analisis Eksperimen: Konflik Antar Elemen dalam Web Development terkait `z-index`

1. Pendahuluan

Dalam pengembangan web, salah satu tantangan yang sering ditemui adalah mengatur urutan tampilan elemen-elemen yang saling tumpang tindih. CSS menyediakan properti `z-index` untuk menentukan urutan elemen pada sumbu z (ke depan atau ke belakang). Namun, sering kali terjadi konflik antar elemen ketika `z-index` tidak berfungsi sesuai harapan, terutama ketika elemen-elemen tersebut berada dalam konteks tumpukan (stacking context) yang berbeda. Laporan ini membahas hasil eksperimen untuk memahami bagaimana konteks tumpukan dan `z-index` mempengaruhi urutan tampilan elemen.

2. Identifikasi Masalah

Masalah utama yang dihadapi dalam eksperimen ini adalah ketika dua elemen memiliki nilai `z-index` yang berbeda, namun tidak ditampilkan sesuai dengan urutan yang diharapkan. Hal ini sering terjadi karena adanya konteks tumpukan yang diciptakan oleh properti CSS tertentu seperti `transform`, `opacity`, dan `position`. Akibatnya, elemen dengan `z-index` lebih tinggi dapat tetap berada di bawah elemen lain dalam konteks tumpukan yang berbeda.

3. Tujuan Eksperimen

Tujuan dari eksperimen ini adalah untuk:

- Memahami bagaimana konteks tumpukan mempengaruhi perilaku `z-index`.

- Mengidentifikasi situasi di mana `z-index` tidak berfungsi seperti yang diharapkan karena konteks tumpukan.

- Menyediakan solusi yang dapat digunakan untuk menghindari konflik antar elemen yang melibatkan `z-index`.

4. Metodologi Eksperimen

Eksperimen ini dilakukan dengan membuat beberapa contoh kode HTML dan CSS di mana dua elemen yang saling tumpang tindih memiliki nilai `z-index` yang berbeda. Dalam setiap contoh, kondisi yang berbeda diterapkan, seperti penggunaan properti `transform`, `opacity`, dan `position` untuk menciptakan konteks tumpukan baru. Hasil dari eksperimen ini dianalisis berdasarkan bagaimana elemen-elemen tersebut ditampilkan di halaman web.

5. Pelaksanaan Eksperimen

Eksperimen dilaksanakan dengan langkah-langkah berikut:

1. Membuat dua elemen (`box1` dan `box2`) dalam sebuah container dengan posisi absolut.

2. Mengatur nilai `z-index` yang berbeda untuk kedua elemen, dengan harapan elemen dengan `z-index` lebih tinggi akan muncul di atas.

3. Menerapkan properti CSS seperti `transform`, `opacity`, dan `position` pada `box1` untuk menciptakan konteks tumpukan baru.

4. Mengamati hasil tampilan di browser dan mencatat apakah urutan elemen sesuai dengan nilai `z-index` yang diatur.

6. Analisis Hasil Eksperimen

Dari hasil eksperimen, ditemukan bahwa:

- Dalam kasus tanpa konteks tumpukan baru, `box2` dengan `z-index` lebih tinggi muncul di atas `box1` seperti yang diharapkan.

- Ketika properti `transform` atau `opacity` diterapkan pada `box1`, konteks tumpukan baru tercipta, dan meskipun `box2` memiliki `z-index` lebih tinggi dalam konteks global, `box1` tetap berada di atas karena `box2` berada dalam konteks tumpukan yang berbeda.

- Dalam situasi lain di mana `box2` juga memiliki konteks tumpukan baru dengan `z-index` lebih rendah, `box2` tetap berada di bawah `box1` meskipun secara global nilai `z-index`-nya lebih tinggi.

7. Kesimpulan

Eksperimen ini menunjukkan bahwa konflik antar elemen dalam hal `z-index` sering kali disebabkan oleh konteks tumpukan yang berbeda. Penggunaan properti CSS seperti `transform`, `opacity`, dan `position` dapat menciptakan konteks tumpukan baru yang mempengaruhi bagaimana `z-index` bekerja. Untuk mengatasi masalah ini, penting untuk memahami bagaimana konteks tumpukan terbentuk dan mempertimbangkan ini saat mengatur nilai `z-index` pada elemen-elemen yang saling tumpang tindih.

8. Saran

Berdasarkan hasil eksperimen, berikut beberapa saran untuk mencegah konflik antar elemen terkait `z-index`:

- Hindari Penggunaan Berlebihan Konteks Tumpukan Baru: Sebisa mungkin, hindari penggunaan properti seperti `transform`, `opacity`, dan `position` yang tidak perlu pada elemen yang seharusnya mengikuti urutan `z-index`.

- Periksa Kembali Konteks Tumpukan: Ketika menghadapi masalah `z-index`, periksa kembali konteks tumpukan dari elemen-elemen tersebut untuk memastikan mereka berada dalam konteks yang sama.

- Gunakan Teknik Debugging: Gunakan alat pengembang di browser untuk memeriksa konteks tumpukan dan urutan `z-index` untuk memahami bagaimana elemen-elemen ditampilkan di halaman.

Laporan ini memberikan panduan dan pemahaman yang lebih baik mengenai cara mengatasi konflik antar elemen terkait `z-index` dalam pengembangan web.