**LAPORAN**

**EKSPLORASI PROBLEM YANG TERDAPAT**

**DI WEB DEVELOPMENT: *Z-INDEX***

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



**Disusun oleh:**

Nama : Alisha Nara Chandrakirana

NIM : 221524033

Kelas : 3B

Kelompok: 3

**Mata Kuliah:**

Nama Mata Kuliah : Pengembangan Web

Kode Mata Kuliah : 21TI3005

**Dosen Pengampu:**

Joe Lian Min, M.Eng.

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**PROGRAM STUDI D4 TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2024**

**Laporan Hasil Analisis Eksperimen Z-Index dalam Web Development**

1. **Pendahuluan**

Dalam pengembangan web, pengelolaan lapisan elemen (layering) seringkali dilakukan dengan menggunakan properti `z-index` dalam CSS. Properti ini memungkinkan developer untuk mengatur urutan tampilan elemen berdasarkan nilai z-index yang lebih tinggi akan tampil di depan. Namun, ada beberapa kasus khusus di mana properti CSS lain seperti `transform`, `opacity`, dan `filter` dapat menciptakan konteks stacking baru, yang menyebabkan z-index berperilaku tidak seperti yang diharapkan. Laporan ini membahas hasil eksperimen yang dilakukan untuk memahami dampak penggunaan properti `filter` terhadap perilaku z-index.

1. **Identifikasi Masalah**

Masalah utama yang diidentifikasi adalah bahwa elemen dengan z-index lebih rendah bisa saja tampil di depan elemen dengan z-index lebih tinggi ketika properti `filter` dengan nilai selain default diterapkan. Masalah ini menciptakan kebingungan saat mengatur layering elemen dalam pengembangan antarmuka pengguna.

1. **Tujuan Eksperimen**

Tujuan dari eksperimen ini adalah:

1. Memahami bagaimana properti `filter` mempengaruhi perilaku z-index.
2. Mengidentifikasi kondisi di mana stacking context baru terbentuk.
3. Menemukan solusi yang tepat untuk mengelola layering elemen ketika menggunakan properti `filter`.
4. **Metodologi Eksperimen**

Metode yang digunakan dalam eksperimen ini melibatkan:

1. Membuat dua elemen HTML dengan nilai z-index berbeda.
2. Menggunakan properti `filter` pada salah satu elemen dan memeriksa dampaknya terhadap urutan layering.
3. Menguji hasil eksperimen di berbagai browser (Chrome, Firefox, Edge) untuk memastikan konsistensi perilaku.
4. **Pelaksanaan Eksperimen**

Eksperimen dilakukan dengan langkah-langkah sebagai berikut:

1. **Menyiapkan Lingkungan Pengujian:** Membuat file HTML sederhana dengan dua elemen yang memiliki nilai z-index berbeda.
2. **Menambahkan Properti Filter:** Menambahkan properti `filter: blur(2px)` pada elemen dengan z-index lebih rendah.
3. **Observasi Hasil:** Mengamati urutan tampilan elemen pada berbagai browser.
4. **Pengujian Alternatif:** Menghilangkan atau memindahkan properti `filter` ke elemen anak untuk melihat perbedaan hasil.
5. **Analisis Hasil Eksperimen**

Hasil eksperimen menunjukkan bahwa:

1. Ketika properti `filter` diterapkan pada elemen, elemen tersebut menciptakan konteks stacking baru. Ini berarti elemen tersebut diatur ulang dalam urutan tampilan yang berbeda, meskipun memiliki z-index lebih rendah, elemen tersebut dapat muncul di depan elemen dengan z-index lebih tinggi yang tidak memiliki properti `filter`.
2. Efek ini konsisten di berbagai browser (Chrome, Firefox, Edge), menunjukkan bahwa ini adalah perilaku yang diharapkan berdasarkan spesifikasi CSS.
3. **Kesimpulan**

Berdasarkan hasil eksperimen, dapat disimpulkan bahwa penggunaan properti `filter` dapat menyebabkan elemen untuk membentuk konteks stacking baru, sehingga mempengaruhi urutan z-index. Oleh karena itu, ketika menggunakan `filter` atau properti serupa lainnya, perlu diperhatikan dampaknya terhadap layering elemen.

1. **Saran**

Beberapa saran yang dapat diambil dari eksperimen ini adalah:

1. **Gunakan Properti dengan Hati-Hati:** Saat menggunakan properti `filter`, `opacity`, atau `transform`, pahami bahwa properti ini menciptakan konteks stacking baru. Pastikan untuk menguji hasilnya di beberapa browser.
2. **Pertimbangkan Alternatif:** Jika properti tersebut menyebabkan masalah layering, pertimbangkan untuk mengaplikasikan efek pada elemen anak, bukan pada elemen induk, untuk menghindari terbentuknya konteks stacking baru.
3. **Debugging dengan Tools:** Gunakan alat debugging yang disediakan oleh browser untuk memeriksa konteks stacking dan z-index elemen-elemen di halaman Anda.

---

Laporan ini diharapkan dapat membantu dalam memahami dampak penggunaan properti CSS tertentu terhadap perilaku z-index dan memberikan panduan untuk mengatasi masalah terkait dalam pengembangan web.