**PENGEMBANGAN WEB (TEORI)**

**LAPORAN EKSPERIMEN: POLYFILLS, PROGRESSIVE ENHANCEMENT, DAN GRACEFUL DEGRADATION**

*Laporan ini disusun untuk memenuhi tugas 3 mata kuliah Pengembangan Web (Teori)*



Disusun oleh kelompok B4:

Asri Husnul Rosadi 221524035

Faris Abulkhoir 221524040

Mahardika Pratama 221524044

Muhamad Fahri Yuwan 221524047

Najib Alimudin Fajri 221524053

Sarah 221524059

Septyana Agustina 221524060

Dosen Pengampu:

Joe Lian Min, M.Eng.

**JURUSAN TEKNIK KOMPUTER DAN INFORMATIKA**

**PROGRAM STUDI D4 TEKNIK INFORMATIKA**

**POLITEKNIK NEGERI BANDUNG**

**2024**

# **DAFTAR ISI**

[**DAFTAR ISI** i](#_Toc176207706)

[**A.** **IDENTIFIKASI PROBLEM** 1](#_Toc176207707)

[**B.** **DESKRIPSI PROBLEM** 1](#_Toc176207708)

[**C.** **METODOLOGI EKSPERIMEN** 1](#_Toc176207709)

[**D.** **PELAKSANAAN EKSPERIMEN** 3](#_Toc176207710)

[**E.** **ANALISIS HASIL EKSPERIMEN** 3](#_Toc176207711)

# **IDENTIFIKASI PROBLEM**

Dalam pengembangan aplikasi web, salah satu tantangan utama adalah memastikan bahwa aplikasi dapat berjalan dengan baik di berbagai jenis browser, terutama mengingat perbedaan dukungan fitur antara browser modern dan browser lama. Polyfills, Progressive Enhancement, dan Graceful Degradation adalah tiga pendekatan yang dapat membantu menangani masalah kompatibilitas antar browser. Dalam eksperimen ini, kita akan mengevaluasi ketiga pendekatan tersebut untuk memahami bagaimana mereka dapat digunakan dalam pengembangan aplikasi web yang mendukung berbagai jenis browser.

# **DESKRIPSI PROBLEM**

Ketiga pendekatan yang akan dieksplorasi adalah:

1. Polyfills: Menyediakan solusi untuk menambahkan dukungan terhadap fitur JavaScript yang tidak tersedia di browser lama.
2. Progressive Enhancement: Fokus pada membangun fungsionalitas dasar terlebih dahulu dan kemudian menambahkan fitur-fitur canggih hanya jika browser mendukungnya.
3. Graceful Degradation: Membangun aplikasi dengan fitur modern dan memastikan ada fallback untuk browser yang tidak mendukung fitur tersebut.

# **METODOLOGI EKSPERIMEN**

1. Desain Eksperimen

Eksperimen ini bertujuan untuk membandingkan efektivitas **Polyfills**, **Progressive Enhancement**, dan **Graceful Degradation** dalam hal kompatibilitas cross-browser. Pengujian dilakukan dengan sebuah proyek web sederhana yang memiliki fitur modern seperti Array.from() yang hanya didukung oleh browser modern.

1. Variabel Eksperimen
   * Variabel Bebas: Pendekatan yang digunakan untuk kompatibilitas (Polyfills, Progressive Enhancement, Graceful Degradation).
   * Variabel Terikat (Dependent Variable): Kemampuan aplikasi berjalan di berbagai browser (browser lama dan modern).
   * Variabel Terkendali (Controlled Variables): Kode sumber aplikasi dan konfigurasi sama untuk setiap pendekatan.
2. Prosedur Eksperimen
   1. Tahap Persiapan:
      1. Menyiapkan proyek web dengan file index.html, style.css, dan main.js.
      2. Membuat implementasi Polyfills, Progressive Enhancement, dan Graceful Degradation..
   2. Pengujian Polyfills:
      1. Jalankan aplikasi di browser lama (IE 11) dan browser modern (Chrome) untuk melihat apakah polyfill bekerja.
   3. Pengujian Progressive Enhancement:
      1. Matikan JavaScript di browser dan cek apakah fungsionalitas dasar form masih bekerja.
   4. Pengujian Graceful Degradation:
      1. Uji aplikasi di browser lama untuk melihat apakah fallback pada fitur modern bekerja dengan baik.
3. Alat dan Bahan
   * Alat: Browser modern (Chrome, Firefox) dan browser lama (Internet Explorer 11 atau Safari 8).
   * Bahan: Kode Sumber: index.html, main.js, style.css, dan file polyfill (array-from-polyfill.js).
4. Pengukuran dan Pengamatan
   * Pengukuran:
     + Waktu yang diperlukan untuk menerapkan setiap metode.
     + Kecepatan dan performa selama rendering halaman.
     + Keakuratan terjemahan dan format tanggal, waktu, mata uang, dan nomor..
   * Pengamatan:
     + Kemudahan konfigurasi dan penggunaan di setiap metode.
     + Apakah metode mendukung RTL untuk bahasa seperti Arab.

# **PELAKSANAAN EKSPERIMEN**

1. Tahap Persiapan

Siapkan proyek web sederhana dengan file index.html, style.css, dan main.js. Implementasikan fitur modern (Array.from()) di dalam JavaScript dan sertakan polyfill untuk browser lama.

1. Pengujian dengan Polyfills
   1. Menjalankan Aplikasi: Jalankan aplikasi di Internet Explorer 11 dan Chrome terbaru.
   2. Catat Hasil: Browser lama menggunakan polyfill dan fitur modern berjalan tanpa masalah.
2. Pengujian dengan Progressive Enhancement
   1. Menjalankan Aplikasi: Matikan JavaScript di browser dan lihat apakah form masih dapat digunakan tanpa JavaScript.
   2. Catat Hasil: Fitur form tetap dapat berfungsi dengan baik meskipun tanpa JavaScript, menunjukkan bahwa pendekatan ini bekerja.
3. Pengujian dengan Graceful Degradation
   1. Menjalankan Aplikasi: Uji aplikasi di browser lama tanpa dukungan untuk fitur modern.
   2. Catat Hasil: Aplikasi tetap berjalan dengan baik dan menampilkan fallback ketika fitur modern tidak didukung.

# **ANALISIS HASIL EKSPERIMEN**

1. Hasil Pengamatan
   1. Polyfills:
      1. Browser Lama: Fitur modern seperti Array.from() berjalan dengan baik menggunakan polyfill.
      2. Browser Modern: Fitur berjalan langsung tanpa memerlukan polyfill.
      3. Catatan: Memerlukan tambahan script polyfill, tetapi tetap menjaga kompatibilitas di browser lama.
   2. Progressive Enhancement:
      1. Browser dengan JavaScript: Fitur modern ditambahkan jika didukung, dan form dapat digunakan dengan fungsionalitas penuh.
      2. Browser tanpa JavaScript: Form tetap dapat digunakan tanpa JavaScript, dengan fungsionalitas dasar tetap berjalan.
      3. Catatan: Efektif dalam memastikan fungsionalitas dasar tetap tersedia tanpa harus menambah polyfill.
   3. Graceful Degradation:
      1. Browser Modern: Fitur modern berjalan dengan baik.
      2. Browser Lama: Fitur modern tidak berfungsi, tetapi aplikasi tetap berjalan tanpa error, hanya fitur tambahan yang hilang.
      3. Catatan: Fallback yang disediakan memastikan pengalaman yang memadai di browser lama.
2. Analisis Pros dan Cons
   1. Polyfills:
      1. Pros: Memungkinkan fitur modern digunakan di browser lama.
      2. Cons: Memerlukan penambahan script eksternal, meningkatkan ukuran bundle.
   2. Progressive Enhancement:
      1. Pros: Memastikan fungsionalitas dasar di semua browser, tanpa ketergantungan pada JavaScript.
      2. Cons: Fitur canggih hanya tersedia di browser modern.
   3. Graceful Degradation:
      1. Pros: Aplikasi tetap berjalan meskipun fitur modern tidak didukung.
      2. Cons: Pengalaman pengguna di browser lama kurang optimal.
3. Kesimpulan

Eksperimen ini menunjukkan bahwa Polyfills, Progressive Enhancement, dan Graceful Degradation masing-masing memiliki kekuatan dan kelemahan dalam menangani kompatibilitas browser. Polyfills sangat berguna untuk memastikan fitur modern dapat bekerja di browser lama, tetapi menambah overhead. Progressive Enhancement memastikan fitur dasar tersedia di semua kondisi, sedangkan Graceful Degradation menjaga agar aplikasi tetap fungsional meski tanpa fitur modern.

Pendekatan terbaik bergantung pada tujuan proyek dan audiens target. Jika dukungan penuh untuk browser lama diperlukan, polyfills adalah solusi yang tepat, sementara Progressive Enhancement lebih cocok untuk memastikan fungsionalitas dasar selalu tersedia..