

## **Abstrak**

Aplikasi Fraud Deterrence Propeller (FDP) dikembangkan untuk mendeteksi dan mencegah kecurangan dalam laporan keuangan di berbagai instansi. Penelitian ini berfokus pada penerapan *Client-Side Rendering* (CSR) menggunakan *framework Next.js* dalam pengembangan aplikasi FDP, serta dampaknya terhadap performa aplikasi. CSR memungkinkan aplikasi memuat dan me-render konten di sisi *client*, yang secara teori dapat meningkatkan pengalaman pengguna dengan tampilan yang lebih cepat dan dinamis. Namun, dalam kasus laporan keuangan yang sering kali melibatkan data besar, CSR dapat berdampak pada *load time* halaman. Hasil pengujian *load time* menunjukkan variasi yang signifikan berdasarkan volume data. Dari delapan halaman yang diuji, tujuh halaman memenuhi *Non-Functional Requirement* (NFR-01) dengan waktu muat di bawah 3 detik. Namun, halaman "Riwayat Aktivitas Pengguna" tidak memenuhi standar ini, dengan median 5,15 detik dan rata-rata 5,32 detik pada *Google Chrome*, serta median 4,26 detik dan rata-rata 4,30 detik pada *Microsoft Edge*. Hal ini disebabkan oleh volume data yang besar. Sebaliknya, halaman "Kelola Akun Admin" memiliki *load time* yang lebih cepat, dengan median 1,03 detik dan rata-rata 1,07 detik. Berdasarkan hasil pengujian, untuk halaman yang menampilkan data dalam jumlah besar, disarankan mempertimbangkan penerapan *pagination* pada metode *Server-Side Rendering* (SSR). Keterbatasan waktu pengembangan juga mengakibatkan beberapa *Functional Requirement* (FR) prioritas *Low* tidak diimplementasikan. Secara keseluruhan, CSR terbukti efektif untuk halaman dengan volume data yang tidak terlalu besar, namun membutuhkan strategi tambahan untuk menangani data dalam jumlah besar.

**Kata kunci:** *fraud, kecurangan laporan keuangan, website, next.js, client-side rendering, load-time test*