

Abstract

Fraud Deterrence Propeller (FDP) application was developed to detect and prevent fraud in financial reports in various agencies. This research focuses on the implementation of Client-Side Rendering (CSR) using the Next.js framework in the development of the FDP application, and its impact on application performance. CSR allows applications to load and render content on the client-side, which in theory can improve user experience with faster and more dynamic displays. However, in the case of financial reports that often involve big data, CSR can have an impact on page load time. The load time test results show significant variations based on data volume. Of the eight pages tested, seven pages met the Non-Functional Requirement (NFR-01) with load times under 3 seconds. However, the “User Activity History” page did not meet this standard, with a median of 5.15 seconds and an average of 5.32 seconds on Google Chrome, and a median of 4.26 seconds and an average of 4.30 seconds on Microsoft Edge. This is due to the large volume of data. In contrast, the “Manage Admin Account” page has a faster load time, with a median of 1.03 seconds and an average of 1.07 seconds. Based on the test results, for pages that display large amounts of data, it is recommended to consider implementing pagination in the Server-Side Rendering (SSR) method. Limited development time also resulted in some Low priority functional requirements not being implemented. Overall, CSR proved to be effective for pages with modest data volumes, but requires additional strategies to handle large amounts of data.

Keywords: *fraud, financial statements fraud, website, next.js, client-side rendering, load-time test*