

BAB I PENDAHULUAN

A. Latar Belakang

Di era digital yang semakin maju, situs web telah menjadi sangat penting untuk interaksi global, penyebaran informasi, dan transaksi pengguna. Situs web merupakan sekumpulan dokumen *Hypertext Markup Language* (HTML) statis yang dibangun untuk memudahkan setiap orang berbagi informasi, selama terhubung ke dalam jaringan internet [1]. Terdapat beberapa acuan yang digunakan sebagai bahan pertimbangan untuk menentukan baik buruknya kualitas sebuah *website*. Sebagai bahan pertimbangan yaitu: kecepatan akses, isi mudah dibaca, dan tata letak atau desain yang konsisten [2]. Kompleksitas situs web menimbulkan tantangan bagi pengembang dalam hal kinerja dan pengalaman pengguna. Semakin banyak data atau konten yang akan ditampilkan, maka hal ini ikut mempengaruhi kinerja kecepatan dari sebuah browser untuk melakukan render sebuah halaman dalam melakukan penyajian [3]. Seorang web desainer dituntut tidak hanya berfokus pada layout desain melainkan harus dapat memaksimalkan kinerja, kecepatan dari website yang dibuat. Kecepatan dan responsivitas situs web secara signifikan memengaruhi retensi pengguna, tingkat konversi, dan persepsi merek [4].

Untuk mengoptimalkan kinerja aplikasi web, pemilihan teknik rendering sangat penting. Berdasarkan dokumentasinya, *Next.js* adalah *framework* baru yang dirancang untuk mengatasi keterbatasan React dalam *Server-side Rendering*. *Next.js* dibangun berdasarkan *Framework React*, yang berarti mengambil keuntungan dari React dan menambahkan beberapa fitur [5]. *Next.js* adalah Library *JavaScript* ringan yang digunakan untuk mengembangkan aplikasi statis dan aplikasi yang dirender di server, *Next.js* hadir dengan konfigurasi minimal dan fleksibel yang dapat diperluas tergantung pada tujuan aplikasi [6]. *Next.js* merupakan *framework full-stack*. Artinya, *Next.js* merupakan *framework front-end* untuk membangun tampilan *Website* sekaligus *framework back-end* untuk menangani proses rendering dan pengelolaan *database* [7]. *Next.js* menggunakan direktori folder sebagai metode routing untuk halaman web. Dengan menggunakan direktori halaman, *Next.js* menyediakan halaman dengan automatic routing [6]. *Next.js* menawarkan banyak fitur, termasuk Server-side rendering (SSR), pemisahan kode otomatis, Static site generation (SSG), Client-side routing (CSR), dukungan CSS dan Sass, plugin dan integrasi [5].

Next.js Secara default melakukan pra-render setiap halaman. Ini berarti bahwa *Next.js* menghasilkan HTML untuk setiap halaman terlebih dahulu, alih-alih semuanya dilakukan oleh *JavaScript* sisi klien [8]. Pra-rendering pada *Next.js*

terdapat pada metode rendering SSR (*Server-side Rendering*), dan SSG (*Static Site Generation*), Hal ini menghasilkan kinerja dan SEO yang lebih baik [9]. *Next.js* juga menawarkan Metode rendering Sisi Client atau lebih sering disebut CSR (*Client-side Rendering*). Pada dasarnya *Next.js* memperbolehkan penggunaan lebih dari satu metode rendering dalam satu halaman, ini mempermudah Development untuk memilih metode rendering yang efektif dan sesuai kebutuhan . Memutuskan antara SSR, CSR, dan SSG memerlukan pertimbangan tujuan aplikasi, kompleksitas halaman, dan kebutuhan interaktivitas. Pemahaman mendalam sangat penting karena setiap teknik melibatkan *trade-off*. SSR baik dalam SEO namun mempengaruhi kinerja server, CSR memperlambat waktu muat awal, namun pemuatan halaman selanjutnya akan lebih cepat serta tidak membebani Server [10], dan SSG lebih andal dalam kecepatan tetapi kurang optimal untuk *database* yang membutuhkan waktu *real-time* [11].

Penelitian ini menganalisis kinerja *Client-side Rendering*, *Server-side Rendering*, dan *Static Site Generation* pada situs Filmku berbasis *Next.js* menggunakan metode *loading time*. Berfokus pada kecepatan memuat halaman, kecepatan menyelesaikan rendering halaman (*finish*), dan pengalaman pengguna, penelitian ini bertujuan untuk memberikan panduan praktis bagi pengembang. Dengan menganalisis dampak performa dan pengalaman pengguna teknik rendering di *Next.js*, Penelitian ini memiliki potensi untuk berkontribusi pada pengembangan aplikasi web yang lebih baik di masa depan

B. Rumusan Masalah

Berdasarkan latar belakang di atas, yang menjadi rumusan masalah dalam penelitian ini yaitu:

1. Bagaimana performa loading web metode *Server-side Rendering* (SSR), *Client-side Rendering* (CSR), *Static Site Generation* (SSG)?
2. Bagaimana perbandingan performa loading web metode *Server-side Rendering*, *Client-side Rendering*, dan *Static Site Generation* pada *Next.js*?

C. Batasan Masalah

Penelitian ini memiliki beberapa batasan agar fokus dan ruang lingkup penelitian dapat terjaga dengan baik:

1. Aplikasi *Website* dibangun menggunakan *framework Next.js* versi 13.
2. *Software* yang digunakan untuk pengujian performa *Website* adalah DevTolls Crome versi 118.0.5993.90.
3. Aplikasi berbasis *Framework Next.js* yang di jalankan pada ubuntu server versi 22.04
4. Penelitian ini hanya mendasarkan pada time load browser client.
5. Tidak melakukan pengukuran pada konsumsi *resource* di sisi server.

Dengan adanya batasan-batasan ini, penelitian diharapkan dapat menghasilkan analisis yang lebih fokus dan sesuai dengan tujuan yang telah ditetapkan

D. Tujuan Penelitian

Adapun tujuan yang ingin dicapai dari penelitian ini adalah sebagai berikut:

1. Mengetahui performa loading web metode *Server-side Rendering* (SSR), *Client-side Rendering* (CSR), dan *Static Site Generation* (SSG).
2. Mengetahui perbandingan performa loading web metode *Server-side Rendering*, *Client-side Rendering*, dan *Static Site Generation* pada *framework Next.js*.

E. Manfaat Penelitian

Adapun manfaat yang dapat diperoleh dari penelitian ini antara lain :

1. Bagi Peneliti
Sebagai sarana bagi penulis dalam mengimplementasikan ilmu dan keterampilan yang didapat selama kuliah, khususnya mengenai sistem *rendering* pada *Website* berbasis *Next.js*. Dan dengan melakukan analisis ini peneliti dapat mengetahui mana metode *rendering* yang lebih tepat digunakan dalam *Website* Filmku yang berbasis *Next.js*.
2. Bagi Instansi yang bersangkutan
Hasil penelitian ini dapat digunakan sebagai referensi dalam pengembangan *Website*.
3. Bagi Peneliti selanjutnya
Bagi peneliti selanjutnya yang mengangkat isu tentang Metode *Rendering* dalam *Website* terutama pada *framework Next.js* dapat dijadikan bahan referensi dan pembandingan untuk penelitian selanjutnya.