

# Perbandingan Performa *Client Side Rendering* pada Implementasi *Static Website* (Studi Kasus Barokah RentCar)

Miftakhul Ulum Khasanah<sup>1</sup>, I Made Suartana<sup>2</sup>

<sup>1,2</sup> Program Studi S1 Teknik Informatika, Universitas Negeri Surabaya

<sup>1</sup>[miftakhul.20015@mhs.unesa.ac.id](mailto:miftakhul.20015@mhs.unesa.ac.id)

<sup>2</sup>[madesuartana@unesa.ac.id](mailto:madesuartana@unesa.ac.id)

**Abstrak**— Internet berfungsi sebagai sistem global yang menghubungkan berbagai entitas di seluruh dunia, menyediakan informasi melalui *website* yang diakses menggunakan *browser* seperti *Google Chrome* dan *Mozilla Firefox*. Dalam era teknologi informasi yang pesat, performa *rendering website* menjadi krusial untuk pengalaman pengguna dan tingkat retensi mereka. Penelitian ini menganalisis dan membandingkan performa *rendering Client-Side Rendering (CSR)* dan *Incremental Static Regeneration (ISR)* dalam pengembangan *website* statis, dengan studi kasus Barokah RentCar. CSR melakukan *rendering* di sisi klien menggunakan sumber daya perangkat, sedangkan ISR memungkinkan respon instan dari *server* tanpa *rendering* tambahan. Tujuan penelitian ini adalah menemukan solusi optimal untuk meningkatkan performa *website* statis menggunakan *framework Next.js*, dengan mempertimbangkan kecepatan tampilan, tingkat interaktivitas, dan persepsi kecepatan akses pengguna. Hasil pengujian menggunakan *tools Google Lighthouse* menunjukkan bahwa teknologi *rendering ISR* memberikan kontribusi signifikan dalam performa *website* statis yang responsif dan efisien baik dalam mode *desktop* maupun *mobile*.

**Kata Kunci**— CSR, ISR, *Rendering*, *Google Lighthouse*.

## I. PENDAHULUAN

*Interconnection-networking* (Internet) adalah sistem global jaringan komputer yang saling terhubung, menghubungkan berbagai entitas di seluruh dunia<sup>[1]</sup>. Secara analogi, internet mirip dengan perpustakaan raksasa yang menyimpan beragam informasi yang dibutuhkan oleh masyarakat<sup>[2]</sup>. Informasi ini tersebar melalui *website*, yang berfungsi sebagai tempat penyimpanan dan akses informasi bagi masyarakat global yang mengaksesnya menggunakan berbagai aplikasi peramban (*browser*). *Website* adalah perangkat lunak yang terdiri dari berbagai halaman yang menampilkan konten dalam berbagai bentuk, seperti teks, audio, dan gambar, yang dapat diakses oleh pengguna di seluruh dunia. Setiap halaman dalam sebuah *website* dapat berisi informasi, multimedia, dan interaksi yang disusun secara dinamis untuk memenuhi kebutuhan penggunanya. Untuk mengakses halaman-halaman *website* ini, diperlukan program peramban web (web *browser*) seperti *Google Chrome* atau *Mozilla Firefox*. Selain itu, perangkat komputer atau perangkat lainnya harus terhubung ke internet untuk memungkinkan transmisi data dari *server* web ke perangkat pengguna, sehingga konten dapat ditampilkan dengan benar dan interaktif sesuai dengan desain dan fungsi yang diinginkan.

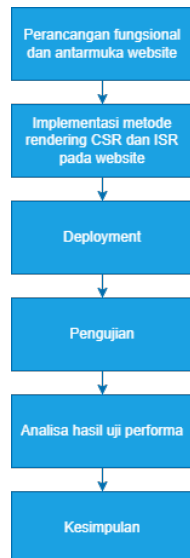
Dalam era perkembangan teknologi informasi yang pesat, aplikasi berbasis *website* menjadi sangat penting untuk memenuhi kebutuhan pengguna. Keberhasilan suatu aplikasi web tidak hanya tergantung pada fungsionalitasnya, tetapi juga pada performa *rendering* yang optimal<sup>[5]</sup>. Penelitian ini dipicu oleh kebutuhan untuk melakukan analisis yang lebih terperinci terhadap performa *rendering*, khususnya pada teknologi *Client Side Rendering (CSR)* dan *Incremental Static Regeneration (ISR)*. Menurut artikel “*Optimizing Client-Side Performance*”<sup>[6]</sup>, performa *rendering* menjadi faktor kunci dalam pengalaman pengguna dan dapat mempengaruhi tingkat retensi pengguna.

Salah satu tantangan utama dalam pengembangan *website* adalah waktu render halaman yang lambat. Hal ini dapat disebabkan oleh kompleksitas desain, ukuran konten yang besar, atau proses *rendering* yang tidak optimal<sup>[7]</sup>. Masalah ini dapat mengakibatkan penurunan kecepatan akses *website*, frustrasi pengguna, dan bahkan berdampak negatif pada citra perusahaan.

Untuk mengatasi permasalahan performa *website* yang kurang optimal, dua metode *rendering* tersebut menawarkan pendekatan yang berbeda. *Client Side Rendering (CSR)* memungkinkan penggunaan sumber daya perangkat untuk melakukan *rendering* di sisi klien, sedangkan *Incremental Static Regeneration (ISR)* memungkinkan respon instan dari *server* tanpa proses *rendering* tambahan<sup>[8][9]</sup>. Dengan menganalisis kedua metode ini secara komprehensif, penelitian ini bertujuan untuk menemukan solusi terbaik untuk meningkatkan performa pada sebuah *website* statis sesuai dengan kebutuhan dan tujuan proyek, khususnya dalam pengembangan *static website*. Faktor-faktor seperti tampilan *website* yang cepat dan optimal, tingkat interaktivitas yang tinggi, dan persepsi kecepatan akses yang maksimal menjadi pertimbangan utama. Oleh karena itu, penelitian ini tertarik untuk melakukan perbandingan mekanisme *rendering* antara *Client Side Rendering (CSR)* dan *Incremental Static Regeneration (ISR)*. Studi kasus yang dipilih adalah *website* Barokah RentCar, dengan harapan menemukan hasil optimal terkait proses *rendering* dalam pengembangan *website* menggunakan *framework Next.js* yang merupakan *framework* dari *library React.js*.

## II. METODE PENELITIAN

Penelitian ini melalui beberapa tahapan terstruktur yang ditunjukkan pada diagram alur penelitian di Gbr. 1.



Gbr. 1 Alur Penelitian

#### A. Perancangan Fungsi dan Antarmuka Website

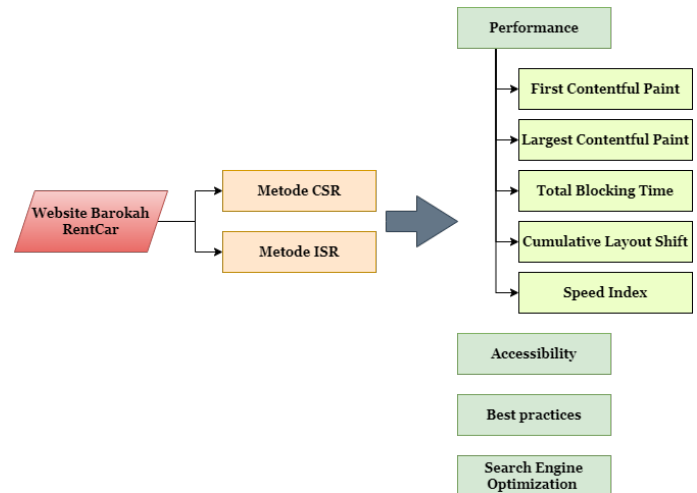
Dalam penelitian ini dilakukan percobaan dalam bentuk studi kasus yang bertujuan untuk menunjukkan pengaruh suatu perlakuan tertentu terhadap fenomena atau perubahan yang terjadi akibat perlakuan tersebut<sup>[10]</sup>.

Penelitian ini menggunakan studi kasus *website* Barokah RentCar yang memuat informasi terkait penyewaan mobil di kawasan Mojokerto.

Informasi ini mencakup daftar mobil yang disewakan, testimonial, *FAQ*, dan kontak person, yang dikombinasikan dengan metode *rendering website*. Metode *rendering website* yang digunakan pada percobaan ini adalah *Client Side Rendering* (CSR) dan *Incremental Static Regeneration* (ISR).

#### B. Perancangan Fungsi dan Antarmuka Website

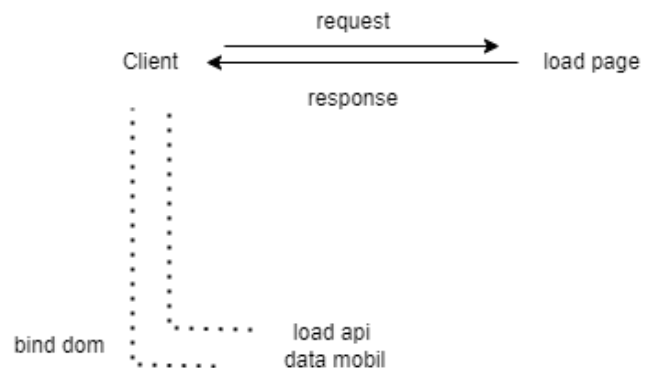
Implementasi metode *rendering Client Side Rendering* (CSR) dan *Incremental Static Regeneration* (ISR) dilakukan pada *website* Barokah RentCar menggunakan *framework NextJS*.



Gbr. 2 Implementasi Sistem

Gbr. 2 menunjukkan kerangka penelitian yang dilakukan. Dalam penelitian ini, aplikasi *browser Google Chrome* digunakan sebagai alat untuk mengakses *website*, sedangkan untuk pengujian digunakan tolos *Google Lighthouse*.

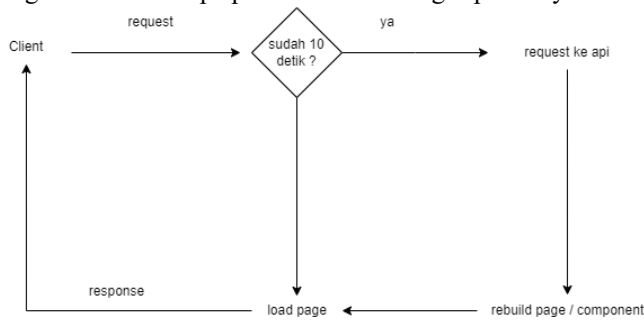
Dalam tahap pembangunan *website* Barokah RentCar, implementasi CSR menggunakan *framework Next.js*, di mana komponen *React* digunakan untuk setiap halaman yang dibuat pada *website*. Sementara itu, ISR juga menggunakan *framework Next.js*, dengan penambahan logika bisnis seperti pengambilan data dari API atau manipulasi data konten. Pada tahap ini, *Next.js* menyediakan berbagai fitur bawaan seperti *prerendering*, di mana halaman *website* dibuat secara statis pada waktu pembangunan.



Gbr. 3 Proses kerja CSR

*Client Side Rendering* (CSR) adalah metode *rendering* di mana konten halaman web dirender di sisi klien menggunakan *JavaScript*. Proses ini dimulai ketika pengguna mengakses URL, dan peramban mengirimkan permintaan HTTP ke *server* web. *Server* merespons dengan mengirimkan kerangka dasar HTML dan *file JavaScript* yang diperlukan. Peramban kemudian mengunduh dan mengeksekusi *JavaScript* tersebut, yang akan membuat permintaan tambahan ke *server backend* untuk mendapatkan data dinamis. Setelah data diterima, *JavaScript* memprosesnya dan merender konten halaman,

memungkinkan pengguna untuk melihat dan berinteraksi dengan halaman tanpa perlu memuat ulang sepenuhnya.



Gbr. 4 Proses kerja ISR

Sedangkan *Incremental Static Regeneration (ISR)* adalah metode *rendering* di mana konten halaman web diperbarui secara periodik tanpa memerlukan pembaruan penuh. Proses ini dimulai ketika pengguna mengakses halaman web, yang kemudian memicu permintaan ke *server*. *Server* memeriksa apakah halaman tersebut sudah ada dalam *cache* dan apakah halaman tersebut sudah lebih dari 10 detik sejak terakhir diperbarui. Jika ya, *server* akan memperbarui halaman secara statis di latar belakang sambil menyajikan versi yang ada kepada pengguna. Hal ini memungkinkan halaman untuk tetap segar dan *up-to-date* tanpa mengganggu pengalaman pengguna, karena mereka selalu melihat konten yang tersedia dengan cepat sementara pembaruan terjadi di belakang layar.

### C. Deployment

Pada tahap ini, *website* yang telah selesai dibangun akan dipublikasikan sehingga dapat diakses secara *online* oleh pengguna. Dalam penelitian ini, platform *deployment* yang akan digunakan adalah *Netlify*. Dengan menggunakan *Netlify*, URL *website* yang diperoleh akan menjadi kunci dalam penelusuran melalui mesin pencari. Hal ini penting untuk memastikan *website* mudah dijangkau oleh pengguna dan memudahkan dalam analisis SEO, karena URL yang terbaca dengan baik dan mengandung kata kunci yang relevan dapat meningkatkan visibilitas *website* dalam hasil pencarian. Diharapkan dengan langkah ini, *website* dapat menarik lebih banyak audiens dan memberikan pengalaman yang optimal bagi pengguna *online*.

### D. Pengujian

Tahap ini melibatkan pengujian performa aplikasi web statis yang telah dibuat dengan menggunakan tools *Google Lighthouse* untuk mengetahui seberapa cepat performa dari kedua metode *rendering website* yaitu *Client Side Rendering (CSR)* dan *Incremental Static Regeneration (ISR)*. Pengujian performa aplikasi web akan dilakukan dalam dua mode perangkat: *desktop* dan *mobile*, dengan tujuan untuk menentukan performa optimal di masing-masing jenis perangkat. Proses pengujian dimulai dengan pemilihan perangkat yang mewakili pengguna rata-rata, mencakup berbagai model *desktop* dan ponsel dengan ukuran layar dan resolusi berbeda. Lingkungan pengujian juga harus

mencerminkan kondisi nyata, termasuk variasi koneksi jaringan. Pengujian akan mencakup aspek fungsional dan non-fungsional, memastikan semua fitur berfungsi dengan baik dan aplikasi efisien dalam penggunaan sumber daya.

Parameter yang digunakan sebagai acuan untuk mengukur kecepatan *rendering* antara kedua metode *rendering* tersebut meliputi: *Performance*, *Accessibility*, *Best Practices*, dan *Search Engine Optimization (SEO)*.

1) *Performance score* adalah rata-rata dari hasil beberapa skor, metrik yang lebih berbobot memiliki pengaruh lebih besar pada skor performa secara keseluruhan, diantara metrik penyusunnya adalah:

- First Contentful Paint*, untuk mengukur waktu *responsivitas* suatu halaman web,
- Largest Contentful Paint*, untuk menentukan waktu yang dibutuhkan oleh *browser* dalam me-render elemen konten utama terbesar,
- Total Blocking Time*, bertujuan untuk mengukur waktu total yang dihabiskan oleh halaman web dalam menghadapi pemblokiran *input* yang dapat dilakukan pengguna selama proses *rendering* halaman *website*,
- Cumulative Layout Shift*, merupakan metrik yang digunakan dalam mengukur stabilitas visual halaman web dengan mengidentifikasi seberapa sering elemen-elemen pada halaman web berpindah atau bergeser dari posisi awalnya selama proses *rendering*,
- Speed Index*, untuk mengukur kecepatan *loading* halaman web dengan cara mengevaluasi seberapa cepat konten halaman muncul dalam *viewport* pengguna selama proses *rendering* oleh *browser*.

Dalam proses pengujiannya, *Google Lighthouse* menggunakan parameter yang terdapat pada model *Web Vitals* dengan tambahan satu parameter yaitu *Speed Index*. Menurut dokumentasi resmi *Google Lighthouse* ([developer.chrome.com/docs/lighthouse/](https://developer.chrome.com/docs/lighthouse/)), Tabel 1 merupakan parameter yang diukur beserta kategori waktu *rendering* pada *Google Lighthouse*.

TABEL 1  
PARAMETER DAN PENGKATEGORIAN NILAI PADA GOOGLE LIGHTHOUSE

| Aspek                           | Kategori        | Waktu Rendering        |
|---------------------------------|-----------------|------------------------|
| <i>First Contentful Paint</i>   | Cepat (Hijau)   | 0 – 1,8 seconds        |
|                                 | Sedang (Kuning) | 1,9 – 3 seconds        |
|                                 | Lambat (Merah)  | > 3 seconds            |
| <i>Largest Contentful Paint</i> | Cepat (Hijau)   | 0 – 2,5 seconds        |
|                                 | Sedang (Kuning) | 2,6 – 4 seconds        |
|                                 | Lambat (Merah)  | > 4 seconds            |
| <i>Total Blocking Time</i>      | Cepat (Hijau)   | 0 – 200 milliseconds   |
|                                 | Sedang (Kuning) | 201 – 600 milliseconds |
|                                 | Lambat (Merah)  | > 600 milliseconds     |
| <i>Cumulative Layout Shift</i>  | Cepat (Hijau)   | 0 – 0,1 %              |
|                                 | Sedang (Kuning) | 0,2 – 0,25 %           |
|                                 | Lambat (Merah)  | > 0,25 %               |
| <i>Speed Index</i>              | Cepat (Hijau)   | 0 – 3,4 seconds        |
|                                 | Sedang (Kuning) | 3,5 – 5,8 seconds      |
|                                 | Lambat (Merah)  | > 5,8 seconds          |
| <i>Final Score</i>              | Cepat (Hijau)   | 90 – 100               |

|  |                 |         |
|--|-----------------|---------|
|  | Sedang (Kuning) | 50 – 89 |
|  | Lambat (Merah)  | 0 – 49  |

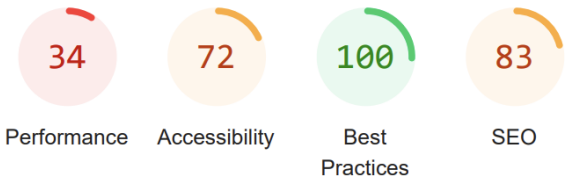
Menurut sumber yang sama, setelah nilai pengujian diperoleh, langkah selanjutnya *Google Lighthouse* mulai menghitung nilai kinerja secara keseluruhan dengan pembobotan yang ada pada Tabel II.

TABEL II  
BOBOT PARAMETER

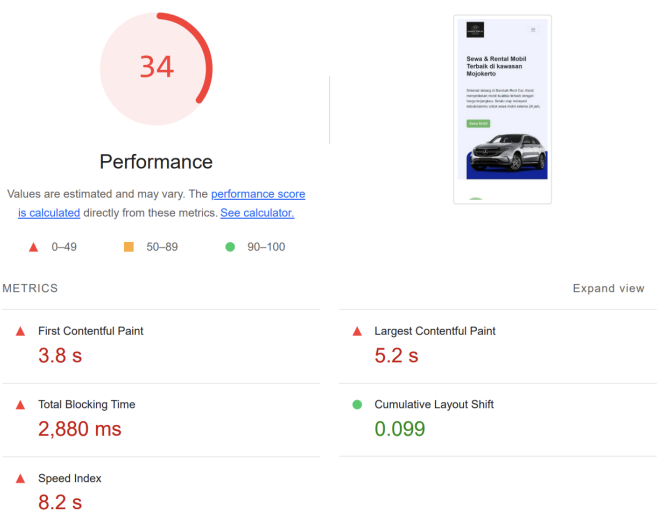
| Aspek Performance        | Bobot |
|--------------------------|-------|
| First Contentful Paint   | 10 %  |
| Largest Contentful Paint | 25 %  |
| Total Blocking Time      | 30 %  |
| Cumulative Layout Shift  | 25 %  |
| Speed Index              | 10 %  |
| Total                    | 100   |

- 2) *Accessibility*, poin penting dalam penilaian ini adalah mengenai aksesibilitas halaman *website* untuk semua pengguna, termasuk mereka dengan disabilitas. Penilaian ini mencakup elemen HTML yang dapat diakses dengan teks deskriptif untuk tombol dan link, kontras warna yang memadai untuk keterbacaan, navigasi yang sepenuhnya dapat dilakukan dengan *keyboard*, dan teks alternatif pada gambar untuk membantu pengguna dengan pembaca layar.
- 3) *Best Practice*, penilaian ini mengevaluasi apakah halaman web mengikuti praktik terbaik untuk pengembangan web, mencakup tiga aspek utama: untuk keamanan, memastikan tidak ada *JavaScript* yang rentan atau tidak aman, dan validasi HTML serta CSS untuk mencegah kesalahan yang dapat mempengaruhi *render* halaman atau menyebabkan masalah fungsi pada berbagai *browser*.
- 4) *Search Engine Optimization*, untuk mengoptimalkan *website* agar lebih mudah ditemukan oleh mesin pencari yang dipengaruhi oleh beberapa strategi dan teknik termasuk penggunaan kata kunci, konten berkualitas, pengoptimalan struktur, dan pembangunan tautan yang baik.

Setelah mengevaluasi dan mengklasifikasikan seluruh parameter, *Google Lighthouse* menampilkan output keseluruhan dari setiap parameter dan hasil evaluasi akhir bobot parameter. Gbr. 5 dan Gbr. 6 merupakan contoh keluaran dari *Google Lighthouse*:



Gbr. 5 Contoh hasil parameter pengujian pada Google Lighthouse



Gbr. 6 Contoh hasil pengujian pada Google Lighthouse

E. Analisa Hasil Perbandingan Uji Performa

Pada tahap ini, hasil pengujian yang dilakukan pada langkah sebelumnya dianalisis. Hasil analisis ini digunakan untuk mengetahui kinerja *website* yang dibuat. Setelah seluruh data dari kedua metode terkumpul, maka dapat diambil kesimpulan berdasarkan hasil analisis. Hasil analisis mengukur parameter yang ditetapkan sebagai nilai untuk setiap metode rendering.

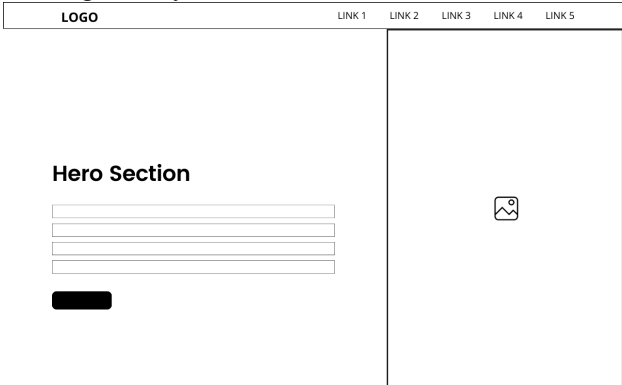
III. HASIL DAN PEMBAHASAN

A. Hasil Implementasi

Penelitian ini menghasilkan *website* statis yang dirancang untuk mengukur pengaruh teknologi *Client Side Rendering* (CSR) dan *Incremental Static Regeneration* (ISR) terhadap performa *rendering* dan pengalaman pengguna. *Website* ini digunakan untuk mengevaluasi bagaimana kedua teknologi mempengaruhi kecepatan *loading* halaman, interaktivitas, dan efisiensi *rendering*.

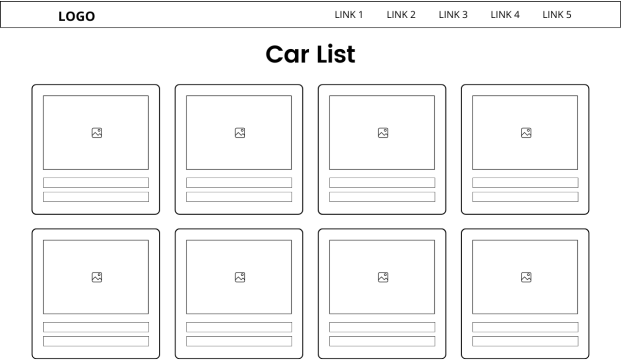
Berikut merupakan desain *wireframe website* yang terbagi menjadi beberapa *section*.

1) Rancangan Wireframe Website



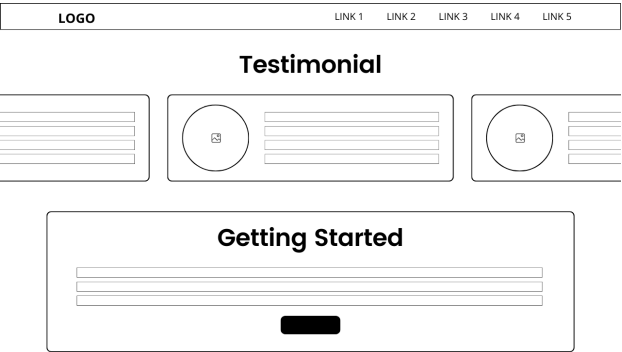
Gbr. 7 Wireframe Hero Section

Gbr. 7 merupakan tampilan rancangan *Hero Section* pada *website* Barokah RentCar baik dari tampilan *desktop* maupun *mobile*. *Hero section* disini termasuk area *visual* utama yang pertama kali dilihat oleh pengunjung ketika mengunjungi halaman web. Dalam section ini terdapat judul, teks, dan gambar untuk mendeskripsikan *website*.



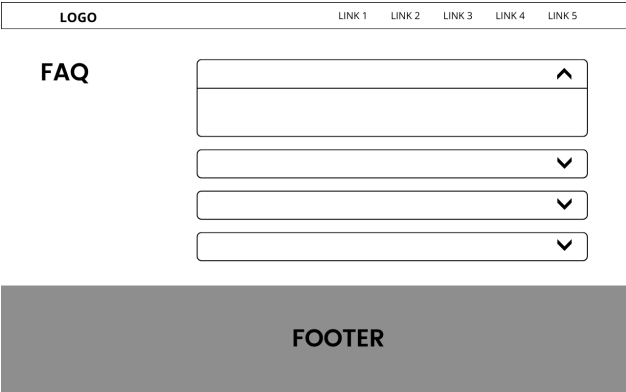
Gbr. 8 Wireframe Car List

Gbr. 8 merupakan tampilan rancangan *Car List Section* pada *website* Barokah RentCar baik dari tampilan *desktop* maupun *mobile*. Dalam *section* ini menampilkan beberapa *card* berisi daftar mobil yang siap untuk di sewa dilengkapi dengan informasi terkait mobil tersebut yang diantaranya *type* mobil, harga sewa/24 jam, transmisi, dan tahun.



Gbr. 9 Wireframe Testimonial

Gbr. 9 merupakan tampilan rancangan Testimonial *section* pada *website* Barokah RentCar menampilkan ulasan dan pengalaman positif dari pelanggan yang telah menggunakan layanan Sewa Mobil. *Section* ini bertujuan untuk membangun kepercayaan dan memberikan bukti nyata mengenai kualitas layanan kami. Testimonial yang ditampilkan mencakup pujian mengenai kondisi mobil yang terawat, pelayanan yang ramah dan profesional, serta kemudahan proses penyewaan. Dengan menampilkan testimonial, dapat memberikan keyakinan kepada calon pelanggan bahwa Barokah RentCar adalah pilihan terbaik untuk kebutuhan rental mobil mereka.



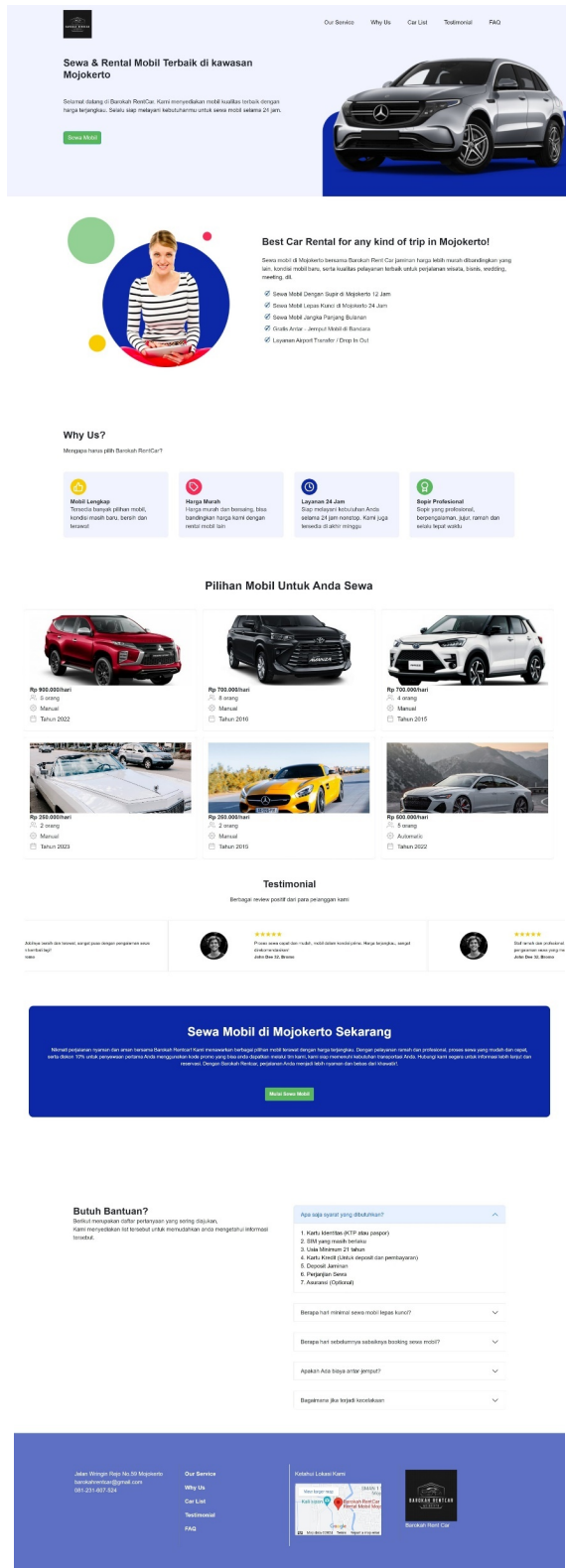
Gbr. 10 Wireframe FAQ dan Footer

Gbr. 10 di atas merupakan tampilan rancangan *Frequently Asked Questions (FAQ) Section* pada *website* Barokah RentCar yang menyediakan jawaban atas pertanyaan-pertanyaan yang paling sering diajukan oleh pelanggan. *Section* ini bertujuan untuk memberikan informasi yang jelas dan cepat mengenai berbagai aspek layanan Barokah RentCar, seperti prosedur penyewaan, persyaratan dokumen, metode pembayaran, kebijakan bahan bakar, dan penanganan masalah teknis. Dengan adanya FAQ *section*, diharapkan dapat membantu pelanggan menemukan jawaban atas pertanyaan mereka dengan mudah, meningkatkan pengalaman pengguna, dan mengurangi kebutuhan untuk menghubungi layanan pelanggan.

2) Hasil Implementasi

Dari tampilan semua *section* yang sudah dirancang diatas dihasilkan tampilan *website* statis yang telah dikembangkan menjadi seperti Gbr. 11



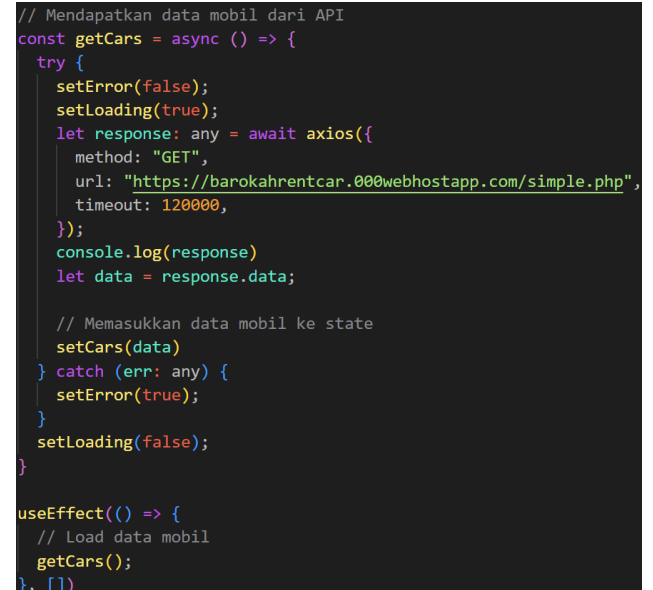


Gbr. 11 Landing page Barokah RentCar

### 3) Implementasi Teknologi

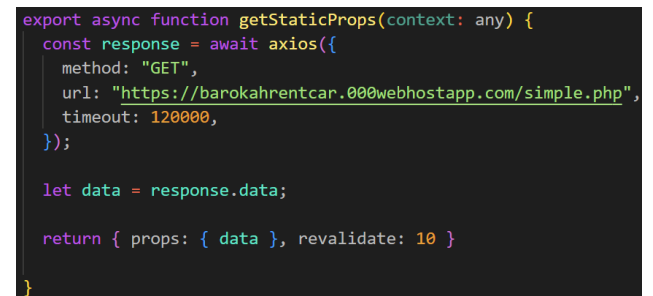
Dalam pembangunan *website* Barokah RentCar, digunakan dua teknologi yang membedakan *website*

utama dengan *website* pembandingnya. *Website* utama diimplementasikan dengan *Client Side Rendering* (CSR) sedangkan *website* pembanding menggunakan teknologi rendering *Incremental Static Regeneration* (ISR).



Gbr. 12 Code implementasi CSR

Pada metode *rendering* CSR di *Next.js* pertama kali dilakukan *loading page* yang kemudian *website* akan melakukan *fetch data* dari *server*. Lalu setelahnya data akan disimpan ke sebuah *state* dan akan ditampilkan ke klien. Dapat dilihat pada Gbr. 15 di atas, *response* di simpan pada *state cars* menggunakan fungsi *setCars*. Untuk dapat menampilkan data *cars* tersebut, kita perlu melakukan *passing state* ke komponen *CarsList*.



Gbr. 13 Code implementasi ISR

Metode *rendering* ISR hanya dapat berjalan pada *Next.js page router*. Pertama perlu membuat fungsi dengan nama *getStaticProps* untuk melakukan *fetching* data. Seperti yang dapat dilihat pada Gbr. 16 di atas, pada *return* data di fungsi *getStaticProps* terdapat *attribute revalidate*, *attribute* ini berfungsi sebagai penanda setiap berapa detik sekali komponen tersebut perlu dilakukan *re-build*.

### B. Pengujian

Setelah diperoleh seluruh hasil dari proses pengujian selama 1 bulan (empat minggu), maka dihitung rata-rata setiap parameter dan hasil akhir dari empat proses pengujian sebelumnya. Berikut adalah rata-rata hasil pengujian *website* Barokah RentCar menggunakan teknologi *Client Side Rendering* (CSR) dan *Incremental Static Regeneration* (ISR) pada mode *desktop* dan *mobile*.

#### 1) Hasil Pengujian CSR

TABEL III  
HASIL UJI CSR MODE DESKTOP

| Aspek                | Pengujian CSR dengan mode desktop |          |          |          | Rata-rata |
|----------------------|-----------------------------------|----------|----------|----------|-----------|
|                      | Uji ke-1                          | Uji ke-2 | Uji ke-3 | Uji ke-4 |           |
| FCP                  | 4,9 s                             | 3,1 s    | 1,2 s    | 1,2 s    | 2,6       |
| LCP                  | 5,1 s                             | 4,4 s    | 1,6 s    | 1,4 s    | 3,125     |
| TBT                  | 0 ms                              | 550 ms   | 950 ms   | 1130 ms  | 657,5     |
| CLS                  | 0,003                             | 0,249    | 0,003    | 0,003    | 0,065     |
| Speed Index          | 5,9 s                             | 6,7 s    | 7,5 s    | 2,2 s    | 5,575     |
| Final Score Performa | 57                                | 23       | 53       | 59       | 48        |
| Accessibility        | 72                                | 72       | 72       | 72       | 72        |
| Best Practices       | 93                                | 74       | 96       | 96       | 89,75     |
| SEO                  | 80                                | 80       | 80       | 80       | 80        |

Dari Tabel III, rata-rata hasil pengujian *website* Barokah RentCar dengan teknologi *rendering* CSR mode *desktop* mendapatkan nilai pada aspek *First Contentful Paint* (FCP) rata-rata memerlukan 2,6 *seconds* yang masuk ke dalam kategori sedang, aspek *Large Contentful Paint* (LCP) rata-rata memerlukan 3,125 *seconds* yang masuk ke dalam kategori sedang, aspek *Total Blocking Time* (TBT) rata-rata memerlukan 675,5 *milliseconds* yang masuk kedalam kategori lambat, aspek *Cumulative Layout Shift* (CLS) rata-rata menunjukkan 0,065 yang masuk ke dalam kategori cepat, aspek *Speed Index* rata-rata memerlukan 5,575 *seconds* yang masuk ke dalam kategori lambat, sehingga keseluruhan *final score* pada parameter Performa menunjukkan 48 yang masuk ke dalam kategori lambat. Kemudian rata-rata pengujian pada parameter yang lain yaitu diantaranya *Accessibility* mendapatkan nilai 72; 89,75 untuk *Best Practices*; dan 80 untuk SEO.

TABEL IV  
HASIL UJI CSR MODE MOBILE

| Aspek | Pengujian CSR dengan mode mobile |          |          |          | Rata-rata |
|-------|----------------------------------|----------|----------|----------|-----------|
|       | Uji ke-1                         | Uji ke-2 | Uji ke-3 | Uji ke-4 |           |
| FCP   | 3,8 s                            | 2,6 s    | 3,0 s    | 3,3 s    | 3,175     |
| LCP   | 5,2 s                            | 3,9 s    | 4,9 s    | 5,9 s    | 4,975     |

|                      |         |         |         |         |       |
|----------------------|---------|---------|---------|---------|-------|
| TBT                  | 2880 ms | 2740 ms | 1690 ms | 6970 ms | 3570  |
| CLS                  | 0,099   | 0,099   | 0       | 0       | 0,049 |
| Speed Index          | 8,2 s   | 6,1 s   | 5,5 s   | 7,2 s   | 6,75  |
| Final Score Performa | 34      | 47      | 46      | 36      | 40,75 |
| Accessibility        | 72      | 72      | 72      | 72      | 72    |
| Best Practices       | 100     | 100     | 100     | 100     | 100   |
| SEO                  | 83      | 77      | 83      | 83      | 81,5  |

Dari Tabel IV, hasil rata-rata pengujian *website* Barokah RentCar dengan teknologi *rendering* ISR pada mode *mobile* adalah sebagai berikut: *First Contentful Paint* (FCP) rata-rata memerlukan 3,175 detik, termasuk dalam kategori lambat; *Large Contentful Paint* (LCP) rata-rata memerlukan 4,975 detik, juga masuk dalam kategori lambat; *Total Blocking Time* (TBT) rata-rata memerlukan 3.570 milidetik, termasuk dalam kategori lambat; *Cumulative Layout Shift* (CLS) rata-rata 0,049, termasuk dalam kategori cepat; *Speed Index* rata-rata memerlukan 6,75 detik, masuk dalam kategori lambat. Sehingga nilai keseluruhan pada parameter Performa adalah 40,75, termasuk dalam kategori lambat. Selain itu, nilai rata-rata untuk parameter lainnya adalah 72 untuk *Accessibility*, 100 untuk *Best Practices*, dan 81,5 untuk SEO.

#### 2) Hasil Pengujian ISR

TABEL V  
HASIL UJI ISR MODE DESKTOP

| Aspek                | Pengujian ISR dengan mode desktop |          |          |          | Rata-rata |
|----------------------|-----------------------------------|----------|----------|----------|-----------|
|                      | Uji ke-1                          | Uji ke-2 | Uji ke-3 | Uji ke-4 |           |
| FCP                  | 1,3 s                             | 0,9 s    | 1 s      | 0,7 s    | 0,975     |
| LCP                  | 1,4 s                             | 1,7 s    | 1,7 s    | 1,3 s    | 1,525     |
| TBT                  | 0 ms                              | 140 ms   | 160 ms   | 120 ms   | 105       |
| CLS                  | 0,364                             | 0        | 0        | 0        | 0,091     |
| Speed Index          | 2,3 s                             | 2,2 s    | 1,7 s    | 1 s      | 1,8       |
| Final Score Performa | 70                                | 85       | 86       | 95       | 84        |
| Accessibility        | 72                                | 72       | 72       | 72       | 72        |
| Best Practices       | 96                                | 74       | 93       | 96       | 89,75     |
| SEO                  | 80                                | 80       | 80       | 80       | 80        |

Dari Tabel V, rata-rata hasil pengujian *website* Barokah RentCar dengan teknologi *rendering* ISR mode *desktop* mendapatkan nilai pada aspek *First Contentful Paint* (FCP) rata-rata memerlukan 0,975 *seconds* yang masuk ke dalam kategori cepat, aspek *Large Contentful Paint* (LCP) rata-rata memerlukan 1,525 *seconds* yang masuk ke dalam kategori cepat, aspek *Total Blocking Time* (TBT) rata-rata memerlukan

105 *milliseconds* yang masuk kedalam kategori cepat, aspek *Cumulative Layout Shift* (CLS) rata-rata menunjukkan 0,091 yang masuk ke dalam kategori cepat, aspek *Speed Index* rata-rata memerlukan 1,8 *seconds* yang masuk ke dalam kategori sedang, sehingga keseluruhan *final score* pada parameter Performa menunjukkan 84 yang masuk ke dalam kategori sedang cenderung cepat. Kemudian rata-rata pengujian pada parameter yang lain yaitu diantaranya *Accessibility* mendapatkan nilai 72; 89,75 untuk *Best Practices*; dan 80 untuk SEO.

TABEL VI  
HASIL UJI ISR MODE MOBILE

| Aspek                       | Pengujian ISR dengan mode mobile |          |          |          | Rata-rata |
|-----------------------------|----------------------------------|----------|----------|----------|-----------|
|                             | Uji ke-1                         | Uji ke-2 | Uji ke-3 | Uji ke-4 |           |
| FCP                         | 2,7 s                            | 2,6 s    | 3,4 s    | 3,7 s    | 3,1       |
| LCP                         | 8 s                              | 8,8 s    | 6,7 s    | 8,2 s    | 7,925     |
| TBT                         | 630 ms                           | 680 ms   | 10 ms    | 20 ms    | 335       |
| CLS                         | 0                                | 0        | 0,097    | 0        | 0,024     |
| <i>Speed Index</i>          | 5,7 s                            | 6,4 s    | 8,4 s    | 4,5 s    | 6,25      |
| <i>Final Score Performa</i> | 70                               | 49       | 60       | 66       | 61,25     |
| <i>Accessibility</i>        | 72                               | 72       | 72       | 72       | 72        |
| <i>Best Practices</i>       | 100                              | 100      | 93       | 100      | 98,25     |
| SEO                         | 83                               | 83       | 83       | 83       | 83        |

Dari Tabel VI, rata-rata hasil pengujian *website* Barokah RentCar dengan teknologi *rendering* ISR pada mode **mobile** adalah sebagai berikut: *First Contentful Paint* (FCP) rata-rata membutuhkan waktu 3,1 *seconds*, termasuk dalam kategori cepat; *Largest Contentful Paint* (LCP) rata-rata membutuhkan waktu 7,925 *seconds*, termasuk dalam kategori lambat; *Total Blocking Time* (TBT) rata-rata membutuhkan waktu 335 *milliseconds*, termasuk dalam kategori lambat; *Cumulative Layout Shift* (CLS) rata-rata 0,024, termasuk dalam kategori cepat; *Speed Index* rata-rata membutuhkan waktu 6,25 *seconds*, termasuk dalam kategori lambat. Sehingga nilai keseluruhan pada parameter Performa adalah 61,25, termasuk dalam kategori lambat. Selain itu, nilai rata-rata untuk parameter lainnya adalah 72 untuk *Accessibility*, 98,25 untuk *Best Practices*, dan 83 untuk SEO.

### C. Pembahasan

TABEL VII  
PERBANDINGAN RATA-RATA HASIL ANALISIS MODE DEKSTOP

| Aspek | CSR      | ISR     |
|-------|----------|---------|
| FCP   | 2,6 s    | 0,975 s |
| LCP   | 3,125 s  | 1,525 s |
| TBT   | 0,6575 s | 0,105 s |
| CLS   | 6,5 %    | 9,1 %   |

|                             |         |       |
|-----------------------------|---------|-------|
| <i>Speed Index</i>          | 5,575 s | 1,8 s |
| <i>Final Score Performa</i> | 48      | 84    |
| <i>Accessibility</i>        | 72      | 72    |
| <i>Best Practices</i>       | 89,75   | 89,75 |
| SEO                         | 80      | 80    |

Hasil penelitian pada mode *desktop* menunjukkan beberapa perbandingan antara teknologi ISR dan CSR pada *website* Barokah RentCar. Pada aspek *First Contentful Paint* (FCP), di mana konten pertama kali dimuat, ISR lebih unggul dengan waktu rata-rata 0,975 detik dibandingkan CSR yang memerlukan 2,6 detik.

Untuk aspek *Largest Contentful Paint* (LCP), yang mengukur waktu memuat konten terbesar pada halaman, ISR juga lebih unggul dengan waktu rata-rata 1,525 detik dibandingkan CSR yang membutuhkan 3,125 detik.

Pada aspek *Total Blocking Time* (TBT), yang menunjukkan waktu blokir konten, ISR memiliki keunggulan dengan rata-rata waktu 552,5 milidetik, lebih cepat dibandingkan CSR yang membutuhkan 657,5 milidetik (0,6575 detik).

Namun, pada aspek *Cumulative Layout Shift* (CLS), yang mengukur jarak perpindahan posisi teks atau gambar saat dimuat, CSR sedikit lebih unggul dengan rata-rata perpindahan 6,5% dibandingkan ISR yang mencapai 9,1%.

Dalam hal *Speed Index*, yang mengukur kecepatan muat konten visual pertama, ISR lebih unggul dengan waktu rata-rata 1,8 detik, sedangkan CSR memerlukan 5,575 detik.

Penilaian keseluruhan Performa dari *Google Lighthouse* menunjukkan bahwa ISR jauh lebih unggul dengan *final score* 84 dibandingkan CSR yang hanya memperoleh score 48. Meskipun demikian, pada parameter *Accessibility*, *Best Practices*, dan SEO, kedua teknologi tersebut memiliki nilai yang sama, yaitu masing-masing 72 untuk *Accessibility*, 89,75 untuk *Best Practices*, dan 80 untuk SEO.

TABEL VIII  
PERBANDINGAN RATA-RATA HASIL ANALISIS MODE MOBILE

| Aspek                       | CSR     | ISR     |
|-----------------------------|---------|---------|
| FCP                         | 3,175 s | 3,1 s   |
| LCP                         | 4,975 s | 7,925 s |
| TBT                         | 3,57 s  | 0,335 s |
| CLS                         | 4,9 %   | 2,4 %   |
| <i>Speed Index</i>          | 6,75 s  | 6,25 s  |
| <i>Final Score Performa</i> | 40,75   | 61,25   |
| <i>Accessibility</i>        | 72      | 72      |
| <i>Best Practices</i>       | 100     | 98,25   |
| SEO                         | 81,5    | 83      |

Hasil penelitian pada mode *mobile* menunjukkan beberapa perbandingan antara teknologi ISR dan CSR pada



website Barokah RentCar. Pada aspek First Contentful Paint (FCP), di mana konten pertama kali dimuat, ISR sedikit lebih unggul dengan waktu rata-rata 3,1 detik dibandingkan CSR yang memerlukan 3,175 detik.

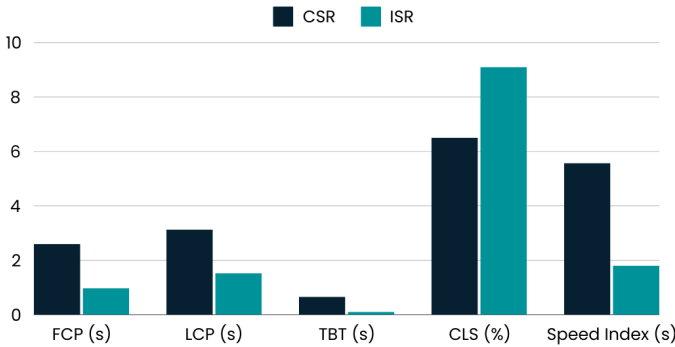
Dalam hal Largest Contentful Paint (LCP), yang mengukur waktu yang diperlukan untuk memuat konten terbesar pada halaman, CSR lebih unggul dengan waktu rata-rata 4,975 detik dibandingkan ISR yang membutuhkan 7,925 detik.

Untuk aspek Total Blocking Time (TBT), ISR memiliki keunggulan dengan rata-rata waktu 3.235 milidetik, lebih cepat dibandingkan CSR yang membutuhkan 3.570 milidetik (3,57 detik).

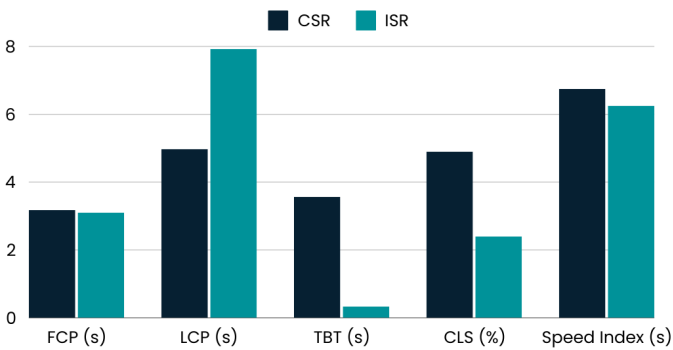
Dalam hal Cumulative Layout Shift (CLS), yang mengukur jarak perpindahan posisi teks atau gambar saat dimuat, ISR sedikit lebih unggul dengan rata-rata perpindahan 2,4% dibandingkan CSR yang mencapai 4,9%.

Pada aspek Speed Index, yang mengukur kecepatan muat konten visual pertama, ISR sedikit lebih unggul dengan waktu rata-rata 6,25 detik dibandingkan CSR yang memerlukan 6,75 detik.

Penilaian keseluruhan Performance dari Google Lighthouse menunjukkan bahwa ISR jauh lebih unggul dengan final score 61,25 dibandingkan CSR yang hanya memperoleh score 40,75. Pada parameter Accessibility, Best Practices, dan SEO, kedua teknologi memiliki nilai yang hampir sama. Nilai Accessibility adalah 72 untuk kedua teknologi. Pada parameter Best Practices, teknologi CSR memiliki nilai 100, sedangkan ISR memiliki nilai 98,25. Dalam hal SEO, CSR memperoleh nilai 81,5, sementara ISR sedikit lebih unggul dengan nilai 83.



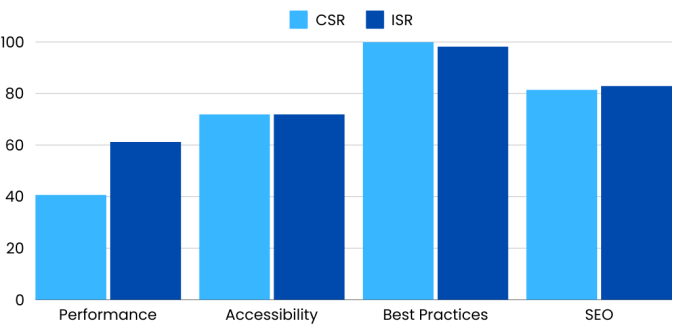
Gbr. 14 Grafik Perbandingan metrix performa CSR dan ISR mode desktop



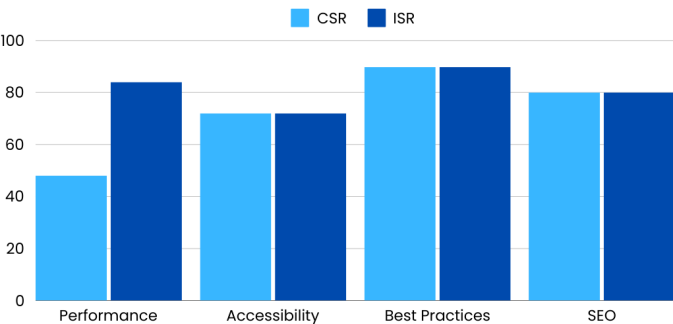
Gbr. 15 Grafik Perbandingan metrix performa CSR dan ISR mode mobile

Pada Gbr. 14, perbandingan grafik pengujian mode desktop menunjukkan bahwa teknologi ISR lebih unggul karena membutuhkan waktu yang lebih singkat dalam empat aspek yaitu FCP, LCP, TBT, dan Speed Index, sementara teknologi CSR lebih unggul hanya dalam satu aspek yaitu CLS.

Pada Gbr. 15, perbandingan grafik pengujian mode mobile juga menunjukkan bahwa teknologi ISR tetap lebih unggul karena membutuhkan waktu yang lebih singkat dalam empat aspek yaitu FCP, TBT, CLS, dan Speed Index, sedangkan teknologi CSR hanya lebih unggul dalam satu aspek yaitu LCP.



Gbr. 16 Grafik Perbandingan tiap parameter CSR dan ISR pada mode desktop



Gbr. 17 Grafik Perbandingan tiap parameter CSR dan ISR pada mode mobile

Pada Gbr. 16, perbandingan grafik pengujian mode *desktop* menunjukkan bahwa penilaian keseluruhan performa dari *Google Lighthouse* juga menunjukkan bahwa teknologi ISR memiliki nilai lebih tinggi dibandingkan dengan teknologi CSR. Namun, pada parameter *Accessibility*, *Best Practices*, dan SEO, kedua teknologi memiliki nilai yang sama.

Pada Gbr. 17, perbandingan grafik pengujian mode *mobile* menunjukkan bahwa pada parameter *Accessibility*, kedua teknologi memiliki nilai yang sama. Namun, pada parameter *Best Practices*, teknologi CSR sedikit lebih unggul daripada ISR, sedangkan pada parameter SEO, teknologi ISR lebih unggul daripada CSR. Meskipun demikian, penilaian keseluruhan performa dari *Google Lighthouse* menunjukkan bahwa teknologi ISR memiliki nilai lebih tinggi dibandingkan dengan teknologi CSR.

#### IV. KESIMPULAN

Penelitian ini bertujuan untuk membandingkan performa *rendering* antara teknologi *Client Side Rendering* (CSR) dan *Incremental Static Regeneration* (ISR) pada *website* statis. Parameter yang dihitung diambil dari model pengujian *Web Vitals* dengan bantuan *Google Lighthouse*, sehingga dapat disimpulkan sebagai berikut.

- 1) Analisis dimulai dengan membuat dua aplikasi yang identik. Studi kasus yang digunakan adalah *website* Barokah RentCar, yang melakukan *rendering* dengan banyak data. Kedua aplikasi dibuat menggunakan teknologi CSR dan ISR. Pengujian performa dilakukan menggunakan *Google Lighthouse* dengan lima aspek: *First Contentful Paint* (FCP), *Speed Index*, *Total Blocking Time* (TBT), *Largest Contentful Paint* (LCP), dan *Cumulative Layout Shift* (CLS).
- 2) Hasil penelitian menunjukkan bahwa teknologi ISR lebih unggul dalam empat aspek, yaitu FCP, LCP, TBT, dan *Speed Index*, dibandingkan teknologi CSR yang unggul dalam satu aspek, yaitu CLS. Selain itu, penilaian keseluruhan performa dari *Google Lighthouse* juga menunjukkan bahwa teknologi ISR memiliki nilai lebih tinggi dibandingkan teknologi CSR. Namun, pada parameter *Accessibility*, *Best Practices*, dan SEO, kedua teknologi memiliki nilai yang sama.
- 3) Hasil penelitian menunjukkan bahwa teknologi ISR tetap lebih unggul dalam empat aspek, yaitu FCP, TBT, CLS, dan *Speed Index*, dibandingkan teknologi CSR yang unggul dalam satu aspek, yaitu LCP. Pada parameter *Accessibility*, kedua teknologi memiliki nilai yang sama. Namun, pada parameter *Best Practices*, teknologi CSR sedikit lebih unggul dari ISR, sementara pada parameter SEO, teknologi ISR lebih unggul dari CSR. Meskipun demikian, penilaian keseluruhan performa dari *Google Lighthouse* menunjukkan bahwa teknologi ISR memiliki nilai lebih tinggi dibandingkan teknologi CSR.

#### REFERENSI

- [1] Anggriawan, F. (2019). PENGEMBANGAN LEARNING MANAGEMENT SYSTEM (LMS) SEBAGAI MEDIA PEMBELAJARAN UNTUK SEKOLAH MENENGAH SEDERAJAT. JTR: Jurnal Tata Rias. <https://core.ac.uk/download/pdf/295413661.pdf>
- [2] Ardiyanto, R., Ardianto, E., & Author, C. (2024). Analisa Peformasi Metode Rendering Website: Client Side, Server Side, dan Incremental Static Regeneration. In Computer Science (CO-SCIENCE (Vol. 4, Issue 1). <http://jurnal.bsi.ac.id/index.php/co-science>
- [3] Apriyanti, W., Erni, Syahlanisyiam, M., Anggraini Y., Gunawan S., Tyas Arinanto, R., Adnan Fauzan, R., Tuto Suban, N., Asmat, & Agung, A. (2022). SOSIALISASI PENGGUNAAN INTERNET YANG SEHAT BAGI ANAK-ANAK DI YAYASAN DOMYANDHU. 1(1), 2963-3486.
- [4] Abdulghani, T., Gozali, M. (2019). Sistem Konsultasi dan Bimbingan Online Berbasis Web Menggunakan Webrtc (Studi Kasus : Fakultas Teknik Universitas Suryakancana). Media Jurnal Informatika, 11(2). <https://jurnal.portalpublikasi.id/index.php/AJP/article/view/7/5>
- [5] Srinivacan. (2023). Optimizing Client-Side Performance: A Comprehensive Guide To Improving User Experience. Diakses pada tanggal 5 Februari 2024. <https://www.lambdatest.com/blog/optimizing-client-side-performance/>
- [6] Nofiyati, Nugroho, A., Wijayanto, B. (2022). EVALUATION OF THE QUALITY OF ACADEMIC INFORMATION SYSTEM UNSOED USING ISO 9126 AND MEAN OPINION SCORE (MOS). Jurnal Teknik Informatika (JUTIF), 3(3). <https://doi.org/10.20884/1.jutif.2022.3.3.366>
- [7] Herman, & Geovanny, A. (2022). ANALISIS RENDERING PERFORMA ANTARA SERVERSIDE DAN CLIENT SIDE PADA WEB APPLICATION. Jurnal Ilmiah Betrik, 13(03), 2339-1871. <https://ejournal.pppmitpa.or.id/index.php/betrik/article/view/38/44>
- [8] Nordstrom, C., Dixelius, A. (2023). Comparisons of Server-side Rendering and Client-side Rendering for Web Pages.
- [9] Patel, V. (2023). Analyzing the Impact of Next.JS on Site Performance and SEO. International Journal of Computer Applications Technology and Research, 12(10). 10.7753/IJCATR1210.1004
- [10] Abraham, I., & Supriyati, Y. (2022). DESAIN KUASI EKSPERIMEN DALAM PENDIDIKAN: LITERATUR REVIEW. Jurnal Ilmiah Mandala Education (JIME), 8(3), 2442-9511. <https://doi.org/10.36312/jime.v8i3.3800/http>