

Performance Comparison and Seo Optimization Between Laravel Blade and Laravel Inertia in The Development of The Muncak.Id Website in Indonesia

Maulana Hafez Ahyatara Tempariyawan¹, Mohammad Irham Akhar^{*2}, Nofiyati⁴, Sugeng Waluyo⁴

^{1,2,3,4}Informatics, Universitas Jenderal Soedirman, Indonesia

Email: ¹mohammad.akbar@unsoed.ac.id

Received : Aug 5, 2025; Revised : Sep 4, 2025; Accepted : Sep 9, 2025; Published : Sep 24, 2025

Abstract

Climbing mountains in Indonesia requires careful preparation, especially in understanding safe hiking routes and related information. A hiking route information website serves as a practical solution for hikers to access data easily and quickly. This study compares the performance and SEO optimization between Laravel Blade and Laravel Inertia in the development of the muncak.id website, which provides information on mountain hiking routes. The performance indicator tested is execution time, measured using Laravel Dusk. One of the test results for the hiking route access feature shows that Laravel Inertia Vue.js has an average execution time of 1.56 seconds, which is faster compared to Laravel Blade, which reached 3.89 seconds. In terms of SEO, both websites show good quality with no issues found. However, there are still opportunities for improvement in some areas marked as warnings and opportunities. Overall, Laravel Inertia with Vue.js proves superior in performance and consistency, as well as more effective in SEO optimization, making it the better choice for developing informative websites.

Keywords : *Operational Efficiency, Maintenance Optimization, School Facilities and Infrastructure.*

This work is an open access article and licensed under a Creative Commons Attribution-Non Commercial 4.0 International License



1. PENDAHULUAN

Mendaki gunung merupakan salah satu aktivitas alam terbuka yang populer di Indonesia, baik bagi masyarakat lokal maupun wisatawan asing [1]. Kegiatan ini tidak hanya memberikan tantangan fisik dan mental, tetapi juga memungkinkan pendaki untuk menikmati keindahan alam [2]. Namun, persiapan yang matang sangat diperlukan sebelum mendaki, mencakup pemahaman jalur, kondisi cuaca, serta perlengkapan yang harus dibawa. Berdasarkan data statistik tahun 2022, sebanyak 67% kasus pendaki hilang disebabkan oleh kesalahan manusia, 32% akibat cuaca buruk, dan kurang dari 1% berkaitan dengan kejadian metafisik [3]. Faktor utama yang menyebabkan pendaki tersesat meliputi kabut tebal, kelelahan fisik, serta penurunan daya konsentrasi yang mengakibatkan mereka terpisah dari rombongan. Oleh karena itu, ketersediaan informasi akurat tentang jalur pendakian sangat penting untuk meningkatkan keselamatan dan kenyamanan selama perjalanan.

Teknologi internet saat ini mempermudah akses informasi, termasuk informasi tentang jalur pendakian gunung. Salah satu media utama dalam penyebaran informasi ini adalah *website*, yang merupakan kumpulan halaman web saling berhubungan yang berisi berbagai informasi dalam bentuk teks, gambar, animasi, audio, dan video [4]. *Website* dapat diakses melalui berbagai perangkat seperti komputer, tablet, dan smartphone, sehingga menjadi sumber informasi utama bagi pendaki untuk mempersiapkan perjalanan mereka. Namun, informasi yang tersedia di internet sering kali tersebar di berbagai platform berbeda, sehingga pendaki harus mencari data dari berbagai sumber seperti prakiraan cuaca, jalur pendakian, dan peraturan gunung [5]. Hal ini menyebabkan ketidakpraktisan dalam

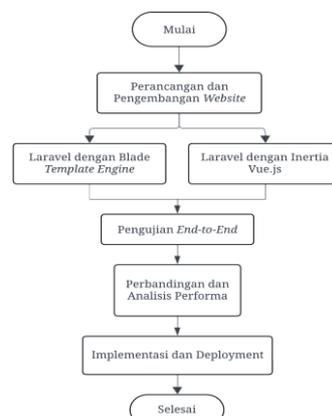
perencanaan pendakian, sehingga dibutuhkan sebuah platform terintegrasi yang dapat menyediakan informasi tersebut dalam satu tempat.

Dalam pengembangan *website* modern, pemilihan *framework* menjadi faktor penting yang menentukan performa dan efisiensi pengelolaan data [6]. Salah satu *framework* yang banyak digunakan adalah Laravel, sebuah *framework* berbasis PHP yang *open-source* dan dikembangkan untuk meningkatkan produktivitas dalam pengembangan aplikasi web [7]. Laravel memiliki berbagai fitur unggulan, seperti *template engine*, *routing*, ORM, dan *modularity*, yang membantu pengembang dalam membangun sistem yang lebih terstruktur dan efisien [8]. Selain performa, optimasi mesin pencari atau *Search Engine Optimization* (SEO) juga berperan penting dalam meningkatkan visibilitas *website* di halaman hasil pencarian. Kecepatan muat halaman menjadi salah satu faktor utama yang menentukan keberhasilan *website* di tengah persaingan digital [9]. Penerapan strategi SEO yang tepat membantu *website* lebih mudah dikenali oleh mesin pencari dan meningkatkan peringkatnya di hasil pencarian [10].

Dalam pengembangan *website* menggunakan Laravel, terdapat dua pendekatan utama, yaitu Laravel Blade dan Laravel Inertia. Laravel Blade merupakan *template engine* bawaan Laravel yang memungkinkan penyisipan kode PHP secara langsung serta menghasilkan tampilan yang efisien tanpa membebani performa aplikasi [7]. Di sisi lain, Laravel Inertia mengintegrasikan *framework* JavaScript modern, seperti Vue.js, untuk menciptakan aplikasi web yang lebih interaktif dan responsif [11]. Penggunaan Inertia memungkinkan navigasi antar halaman tanpa perlu memuat ulang seluruh halaman, meningkatkan pengalaman pengguna dalam berinteraksi dengan *website*. Studi sebelumnya telah membandingkan performa Laravel dengan *framework* lain seperti Express.js, Django, dan Flask, tetapi penelitian tersebut lebih berfokus pada aspek *server-side* tanpa mengevaluasi performa dari sisi pengguna [12], [13], [14]. Penelitian terkait Vue.js juga telah dilakukan, terutama dalam perbandingan dengan React.js [15], [16].

Berdasarkan latar belakang tersebut, penelitian ini bertujuan untuk membandingkan performa dan optimalisasi SEO antara Laravel Blade dan Laravel Inertia dalam pengembangan *website* informasi jalur pendakian gunung di Indonesia *muncak.id*. Aspek performa yang diuji mencakup kinerja sisi klien (*client-side performance*) yang mengukur seberapa cepat pengguna dapat berinteraksi dengan *website* melalui *browser*. Laravel Dusk digunakan sebagai alat uji *end-to-end* untuk mensimulasikan interaksi pengguna serta mengukur waktu eksekusi berbagai skenario sebagai indikator performa [17]. Dalam hal SEO, penelitian ini mengimplementasikan metode *on-page* SEO dan *technical* SEO untuk meningkatkan visibilitas *website* di mesin pencari [10]. Dengan adanya penelitian ini, diharapkan dapat memberikan wawasan dan rekomendasi bagi pengembang *website* dalam memilih teknologi yang lebih efisien dan efektif dalam hal performa serta optimalisasi SEO.

2. METODE PENELITIAN



Gambar 1. Metodologi Penelitian

Penelitian ini menggunakan metodologi yang terdiri dari beberapa tahap yang dapat dilihat pada Gambar 1.

2.1. Perancangan dan Pengembangan *Website*

Penelitian ini menggunakan metode pengembangan perangkat lunak Incremental, di mana sistem dikembangkan secara bertahap dengan setiap iterasi menghasilkan fitur yang dapat diuji dan diimplementasikan. Tahapan ini mencakup analisis kebutuhan sistem, perancangan dengan UML [18], pengembangan kode untuk dua versi *website* menggunakan Laravel Blade dan Laravel Inertia Vue.js, pengujian fungsionalitas dengan Laravel Dusk, serta iterasi dan perbaikan hingga tahap akhir *deployment* menggunakan HestiaCP di server NGINX.

2.2. Laravel dengan Blade *Template Engine*

Website pertama dikembangkan menggunakan Laravel dengan Blade sebagai *template engine* bawaan. Teknologi ini memungkinkan manipulasi tampilan menggunakan sintaks yang ringan dan efisien. Pengembangan melibatkan penggunaan Alpine.js dan jQuery untuk menangani interaksi pengguna, serta MySQL sebagai *database* utama. Dengan pendekatan ini, aplikasi tetap menggunakan metode *server-side rendering* untuk menyajikan halaman web.

2.3. Laravel dengan Inertia Vue.js

Website kedua dikembangkan menggunakan Laravel yang dikombinasikan dengan Inertia dan Vue.js untuk membangun aplikasi dengan pengalaman pengguna yang lebih interaktif. Inertia menggantikan Blade sebagai sistem tampilan dan memanfaatkan Vue.js untuk *rendering* halaman di sisi klien. Axios digunakan sebagai pengelola permintaan HTTP, sedangkan MySQL tetap menjadi *database* utama. Pendekatan ini memungkinkan navigasi yang lebih cepat tanpa perlu melakukan muat ulang halaman secara penuh.

2.4. Pengujian *End-to-End*

Pengujian *end-to-end* dilakukan untuk memastikan bahwa *website* berfungsi dengan baik sesuai skenario pengguna [17]. Laravel Dusk digunakan sebagai alat pengujian otomatisasi *browser* untuk menguji berbagai skenario, seperti proses autentikasi, akses informasi jalur pendakian, pengelolaan data, dan interaksi pengguna seperti memberikan ulasan. Setiap pengujian akan menilai seberapa responsif dan fungsional aplikasi dalam skenario nyata [11].

2.5. Perbandingan dan Analisis Performa

Hasil pengujian *end-to-end* akan dianalisis dengan membandingkan waktu eksekusi pada kedua versi *website*. Setiap *test case* dieksekusi sebanyak 10 kali, dan hasilnya diolah menggunakan statistik deskriptif, mencakup nilai minimum, maksimum, median, mean, serta standar deviasi. Data yang diperoleh kemudian divisualisasikan dalam bentuk tabel untuk memudahkan analisis performa Laravel Blade dan Laravel Inertia Vue.js.

2.6. Implementasi dan *Deployment*

Tahap akhir penelitian mencakup implementasi dan *deployment website*. *Website* akan di-hosting di server berbasis NGINX untuk meningkatkan efisiensi dalam menangani trafik pengguna. Dengan konfigurasi ini, performa *website* diharapkan tetap optimal saat digunakan oleh pendaki yang mencari informasi mengenai jalur pendakian di Indonesia.

3. HASIL DAN PEMBAHASAN

Pengembangan *website* pada penelitian ini menggunakan Model Incremental yang membagi pengembangan menjadi beberapa tahapan [19]. Model pengembangan ini memungkinkan pengembang untuk memeriksa dan memvalidasi setiap bagian sebelum mengintegrasikannya ke dalam sistem yang lebih besar, sehingga dapat mengidentifikasi dan memperbaiki masalah lebih awal dalam proses pengembangan [20]. Hasil penelitian ini adalah sebagai berikut.

3.1. Kebutuhan Pengguna

Dari hasil analisis yang dilakukan, didapatkan bahwa *website* muncak.id ini memiliki dua pengguna dengan penjelasan perannya yang dapat dilihat pada Tabel 1.

Tabel 1. Kebutuhan pengguna

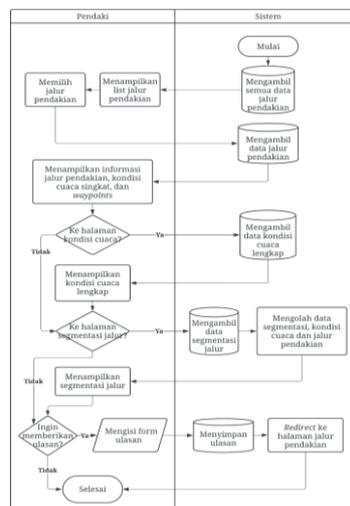
| Nama Pengguna | Peran Pengguna |
|----------------------------|---|
| Pengelola | Pengelola bertugas mengelola data yang ditampilkan di <i>website</i> muncak.id, termasuk data gunung, jalur pendakian, titik koordinat (point), data master wilayah, dan data pengguna. Pengelola juga memperbaiki atau memperbarui konten yang ditampilkan berdasarkan ulasan dari pendaki atau pengguna umum. |
| Pendaki atau Pengguna Umum | Pendaki atau pengguna umum berperan dalam mengakses informasi jalur pendakian yang tersedia di <i>website</i> . Selain itu, pengguna juga dapat memberikan ulasan terkait jalur pendakian yang telah disediakan. |

3.2. Desain

Tahap dilanjutkan dengan merancang desain sistem berdasarkan apa yang telah dijabarkan pada tahap sebelumnya. Pada tahap ini dibagi menjadi beberapa tahapan yaitu flowchart, use case diagram, sequence diagram, activity diagram, dan class diagram. Desain atau rancangan setiap tahapan adalah sebagai berikut.

3.2.1. Flowchart

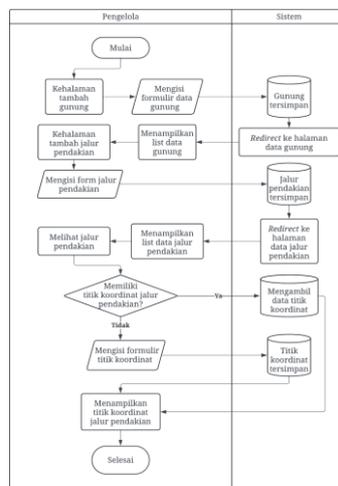
Flowchart membantu memahami urutan-urutan logika yang rumit dan panjang. Selain itu, Flowchart juga membantu mengkomunikasikan jalannya program kepada orang lain (bukan pemrogram) akan lebih mudah [8]. Flowchart pertama adalah flowchart untuk pengaksesan informasi jalur pendakian yang menggambarkan alur bagi aktor pendaki saat ingin memperoleh informasi mengenai jalur pendakian. Flowchart ini dapat dilihat pada Gambar 2.



Gambar 2. Flowchart akses informasi jalur pendakian

Pada Gambar 2, proses dimulai dengan pendaki melihat daftar seluruh jalur pendakian yang tersedia, lalu memilih salah satu jalur. Setelah itu, pendaki akan diarahkan ke halaman jalur pendakian, yang menampilkan informasi singkat seperti nama jalur, deskripsi, kondisi cuaca singkat, serta *waypoints* atau titik-titik penting. Pada halaman ini terdapat tombol yang memungkinkan pendaki untuk mengakses halaman kondisi cuaca lengkap dan segmentasi jalur. Selain itu, di halaman jalur pendakian, pengguna juga dapat melihat ulasan dari pendaki lain serta memberikan ulasan dengan cara mengisi formulir ulasan.

Selanjutnya, flowchart untuk pengelolaan data gunung dan jalur pendakian yang menggambarkan alur pengelolaan dalam mengelola data gunung dan jalur pendakian. Flowchart ini dapat dilihat pada Gambar 3.

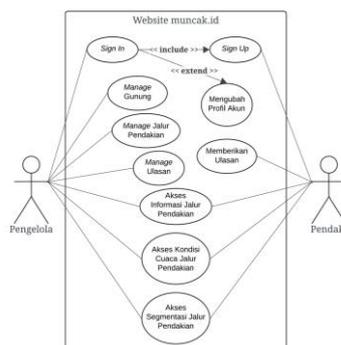


Gambar 3. Flowchart kelola gunung dan jalur pendakian

Pada flowchart yang ditampilkan pada Gambar 3, aktor pada flowchart ini adalah pengelola, yang memulai dengan mengisi formulir data gunung, kemudian menyimpan data tersebut. Dari data gunung yang telah ditambahkan, pengelola dapat menambahkan berbagai jalur pendakian dengan mengisi formulir jalur pendakian. Selanjutnya, pengelola dapat melihat jalur pendakian. Jika jalur pendakian baru, akan diarahkan untuk mengisi titik koordinat jalur pendakian tersebut. Jika jalur pendakian sudah ada, maka data akan ditampilkan.

3.2.2. Use Case Diagram

Use Case Diagram adalah sebuah diagram yang menggambarkan secara garis besar interaksi yang terjadi antara pengguna (user) dengan sistem. Use case diagram mendeskripsikan sebuah interaksi antara satu atau lebih aktor dengan sistem informasi yang akan dibuat [21]. Rancangan use case diagram pada *website* ini dapat dilihat pada Gambar 4.

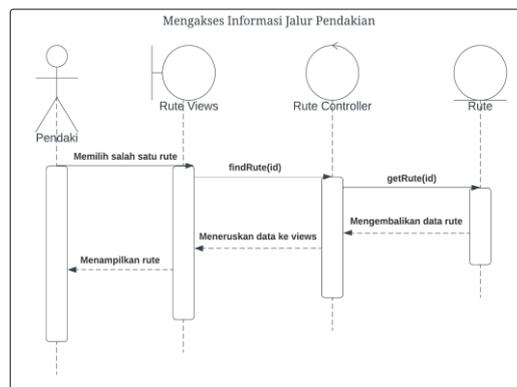


Gambar 4. Use case diagram

Pada use case diagram tersebut, terdapat dua aktor yang berinteraksi dengan *website*, yaitu pengelola dan pendaki. Aktor pengelola dapat melakukan *sign in* serta mengelola data yang ditampilkan pada *website*, seperti data gunung, jalur pendakian, dan ulasan. Kedua aktor memiliki akses untuk melihat informasi jalur pendakian, kondisi cuaca, dan segmentasi jalur. Selain itu, aktor pendaki juga memiliki fitur tambahan, yaitu dapat melakukan sign up dan *sign in*, serta memberikan ulasan.

3.2.3. Sequence Diagram

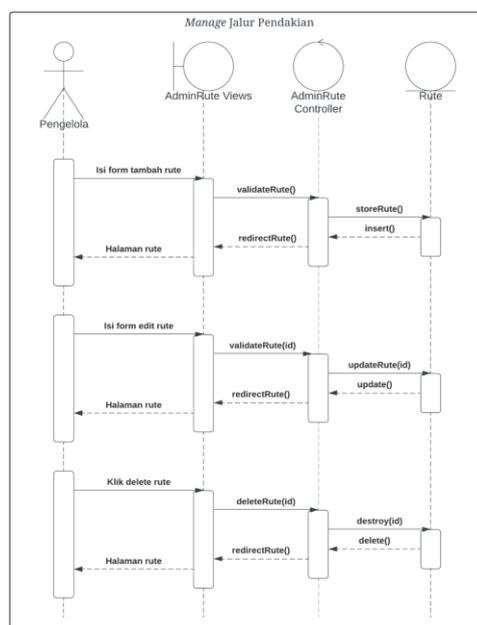
Sequence diagram merupakan sebuah diagram yang menunjukkan bagaimana pengguna berinteraksi dengan sistem serta menggambarkan apa yang terjadi di dalamnya saat sistem berjalan [22]. Sequence diagram pada *website* ini dibagi menjadi 2. Pertama, sequence diagram dari proses mengakses informasi jalur pendakian yang dapat dilihat pada Gambar 5.



Gambar 5. Sequence diagram akses informasi jalur pendakian

Pada Gambar 5 merupakan sequence diagram dari proses mengakses informasi jalur pendakian. Proses dimulai dengan pendaki memilih salah satu jalur pendakian dari daftar yang tersedia. Selanjutnya, sistem akan memproses informasi jalur pendakian dan menampilkannya kepada pengguna pada halaman jalur pendakian.

Selanjutnya, sequence diagram dari proses pengelolaan data jalur pendakian yang dibagi menjadi dilihat pada Gambar 6.

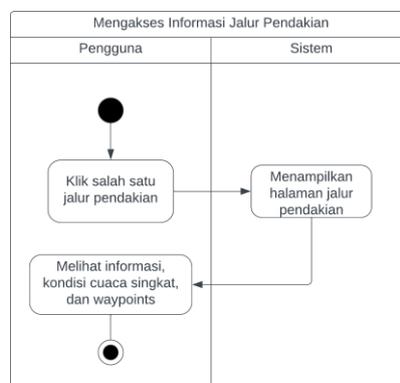


Gambar 6. Sequence diagram kelola jalur pendakian

Pada Gambar 6 merupakan sequence diagram dari proses pengelolaan data gunung yang dibagi menjadi 3 proses. Proses pertama adalah menambahkan data jalur pendakian baru dengan mengisi formulir jalur pendakian. Setelah formulir diisi, data tersebut akan disimpan ke dalam *database*, dan pengguna akan diarahkan kembali ke halaman utama jalur pendakian. Proses kedua adalah mengedit data jalur pendakian yang telah terdaftar, di mana data jalur akan diperbarui jika pengelola mengisi formulir edit dengan informasi baru. Proses terakhir adalah menghapus data jalur pendakian, dimana pengelola memilih salah satu data jalur pendakian yang ada, dan data tersebut akan dihapus dari *database*.

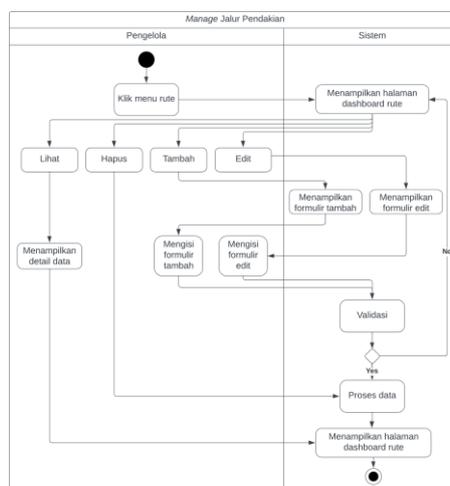
3.2.4. Activity Diagram

Activity diagram merupakan diagram yang menggambarkan tentang aktifitas yang terjadi pada sistem. Dari pertama sampai akhir, diagram ini menunjukkan langkah-langkah dalam proses kerja sistem yang dibuat [23]. Rancangan selanjutnya adalah activity diagram. Activity diagram *website* ini adalah sebagai berikut.



Gambar 7. Activity diagram akses jalur pendakian

Gambar 7 menunjukkan activity diagram untuk melihat jalur pendakian, dimulai dengan pengguna mengklik salah satu jalur pendakian. *Website* kemudian menampilkan halaman detail jalur pendakian, yang berisi informasi lengkap mengenai jalur tersebut, kondisi cuaca singkat, serta waypoint yang harus dilewati.



Gambar 8. Activity diagram kelola jalur pendakian

Gambar 8 menunjukkan activity diagram untuk mengelola jalur pendakian, yang dimulai dengan memilih menu pengelolaan jalur pendakian dan aplikasi akan menampilkan semua data jalur pendakian.

Pada halaman dashboard jalur pendakian, terdapat 4 menu lainnya untuk mengelola data tersebut. Menu pertama adalah menu lihat jalur pendakian yang digunakan untuk menampilkan data jalur pendakian yang ada. Menu kedua adalah menu hapus yang digunakan untuk menghapus data jalur pendakian. Menu ketiga adalah menu tambah jalur pendakian yang digunakan untuk menambahkan data jalur pendakian baru. Pengguna pertama-tama mengisi formulir tambah, dan jika sudah benar, data akan diproses dan disimpan di *database*. Menu terakhir adalah menu edit jalur pendakian yang digunakan untuk mengedit data jalur pendakian yang ada, dengan cara mengisi formulir edit. Jika formulir edit sudah benar, data akan diproses dan diperbarui.

3.2.5. Class Diagram

Desain terakhir adalah class diagram. Class diagram menggambarkan struktur sistem dari segi pendefinisian kelas-kelas yang akan dibuat untuk membangun sistem [21]. Class diagram dapat dilihat pada Gambar 9.



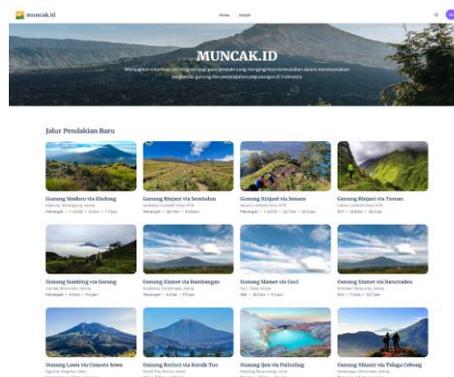
Gambar 9. Class diagram

Gambar 9 menunjukkan bahwa terdapat 9 class yang merupakan hasil perancangan dari activity diagram sebelumnya. Class-class tersebut adalah Gunung, Rute, Point, User, Ulasan, Provinsi, KabupatenKota, Kecamatan, dan Desa. Setiap class memiliki atribut masing-masing beserta perilaku (behavior) yang dapat dilakukan oleh class tersebut. Selain itu, setiap class juga memiliki relasi dengan class lainnya.

3.3. Coding

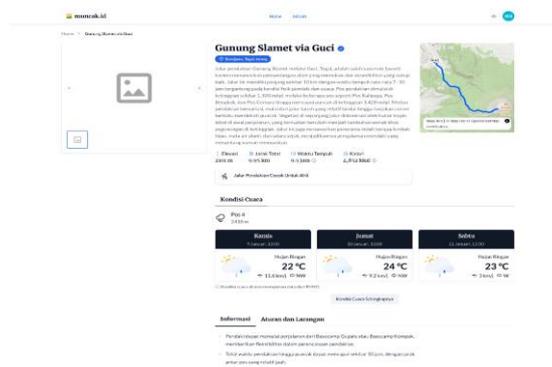
Setelah desain dibuat, tahap selanjutnya adalah implementasi desain ke dalam kode untuk menghasilkan *website*. Hasil dari tahap ini adalah dua *website* yang menggunakan teknologi berbeda, yaitu Laravel dengan Blade *template engine*, dan Laravel dengan Inertia Vue.js.

Halaman landing page merupakan halaman utama yang ditampilkan saat pengguna mengakses muncak.id. Halaman ini menyajikan informasi tentang *website* muncak.id serta jalur pendakian yang tersedia. Beberapa jalur pendakian yang ditampilkan mencakup informasi seperti nama, lokasi, waktu tempuh, jarak total, tingkat kesulitan jalur, rating, jumlah ulasan, serta gambar pendukung. Pengguna dapat mengklik salah satu jalur pendakian untuk mengetahui detailnya, yang kemudian akan mengarahkan mereka ke halaman khusus jalur pendakian yang dipilih. Tampilan halaman ini dapat dilihat pada Gambar 10.



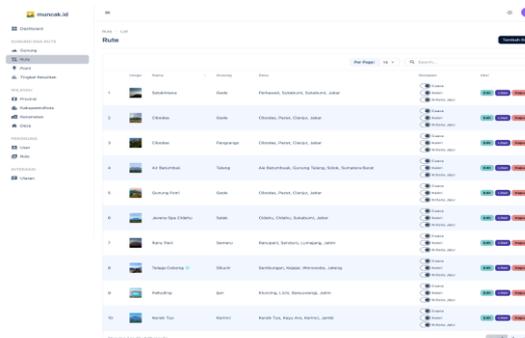
Gambar 10. Halaman landing page Laravel Blade

Halaman jalur pendakian adalah halaman yang menampilkan salah satu jalur pendakian yang tersedia di *website* muncak.id. Fungsi utama halaman ini adalah untuk menampilkan informasi detail mengenai jalur pendakian tersebut. Halaman ini dibagi menjadi beberapa bagian. Pada sisi kiri, terdapat galeri yang berisi gambar-gambar jalur pendakian terkait. Kemudian ada informasi singkat seperti nama, deskripsi, lokasi, waktu tempuh, ketinggian gunung, jarak total, jumlah kalori, dan peta yang menunjukkan bentuk jalur pendakian. Selain itu, terdapat bagian yang menampilkan prediksi cuaca singkat dan *waypoints*. Halaman ini dapat dilihat pada Gambar 11.



Gambar 11. Halaman jalur pendakian Laravel Inertia Vue.js

Halaman data jalur pendakian adalah halaman yang digunakan untuk menampilkan data jalur pendakian atau rute yang tersedia di *website* ini. Halaman ini hanya dapat diakses oleh pengelola atau pengguna yang memiliki peran 'admin'. Di halaman ini, ditampilkan informasi seperti nama jalur pendakian, nama gunung, desa, kesiapan rute dan gambar. Setiap jalur pendakian memiliki tombol untuk melihat, mengedit, dan menghapus data. Selain itu, tersedia tombol "Tambah Rute" yang berfungsi untuk menambahkan data jalur pendakian baru. Halaman ini dapat dilihat pada Gambar 12.



Gambar 12. Halaman kelola jalur pendakian Laravel Blade

3.4. Pengujian End-to-End

Setelah tahap coding atau implementasi selesai, dilakukan pengujian menggunakan Laravel Dusk sebagai *end-to-end testing tool*. Pengujian akses jalur pendakian dilakukan dengan tiga metode: mengakses informasi jalur, melihat prediksi cuaca, dan melihat segmentasi jalur. Pengujian dilakukan pada Gunung Sindoro via Kledung dengan menampilkan data sesuai fitur yang diuji. Metode pengujian ini dapat dilihat pada Gambar 13.

```

1 <?php
2
3 namespace Tests\Browser;
4
5 use App\Models\Route;
6 use Laravel\Dusk\Browser;
7 use PHPUnit\Framework\Attributes\Group;
8 use Tests\DuskTestCase;
9
10 class AccessHikingRouteTest extends DuskTestCase
11 {
12     protected $route;
13
14     public function setUp(): void
15     {
16         parent::setUp();
17         $this->route = Route::with('gunung_id,nama')->findOrFail(1);
18     }
19
20     #[Group('AccessHikingRoute')]
21     public function testAccessHikingRoute(): void
22     {
23         $this->browse(function (Browser $browser) {
24             $browser
25                 ->visit('/')
26                 ->waitForText("Gunung {$this->route->gunung->nama} via {$this->route->nama}")
27                 ->clickLink("Gunung {$this->route->gunung->nama} via {$this->route->nama}")
28                 ->waitForLocation("/jalur-pendakian/{$this->route->slug}")
29                 ->assertPathIs("/jalur-pendakian/{$this->route->slug}")
30                 ->waitForText("Gunung {$this->route->gunung->nama} via {$this->route->nama}");
31         });
32     }
33
34     #[Group('AccessHikingRoute')]
35     public function testAccessHikingRouteWeatherForecast(): void
36     {
37         $this->browse(function (Browser $browser) {
38             $browser
39                 ->visit('/')
40                 ->waitForText("Gunung {$this->route->gunung->nama} via {$this->route->nama}")
41                 ->clickLink("Gunung {$this->route->gunung->nama} via {$this->route->nama}")
42                 ->waitForLocation("/jalur-pendakian/{$this->route->slug}")
43                 ->assertPathIs("/jalur-pendakian/{$this->route->slug}")
44                 ->waitForText("Kondisi Cuaca")
45                 ->clickLink("Kondisi Cuaca Selengkapnya")
46                 ->waitForLocation("/jalur-pendakian/{$this->route->slug}/prediksi-cuaca")
47                 ->assertPathIs("/jalur-pendakian/{$this->route->slug}/prediksi-cuaca")
48                 ->waitForText("Prediksi Cuaca");
49         });
50     }
51 }

```

Gambar 13. Test case pengujian akses jalur pendakian

Hasil eksekusi dari pengujian akses jalur pendakian dapat dilihat pada Tabel 2.

Tabel 2. Hasil analisis pengujian akses jalur pendakian

| Laravel Blade (detik) | | | Laravel Inertia Vue.js (detik) | | | | | | | | |
|-----------------------|----------------------|------------------------|--------------------------------|----------------------|------------------------|--------|------|------------|------|--------|------|
| Akses Jalur Pendakian | Akses Prediksi Cuaca | Akses Segmentasi Jalur | Akses Jalur Pendakian | Akses Prediksi Cuaca | Akses Segmentasi Jalur | | | | | | |
| 4,04 | 1,48 | 1,78 | 2,04 | 0,74 | 1,07 | | | | | | |
| 4,25 | 1,52 | 1,93 | 1,53 | 0,77 | 0,94 | | | | | | |
| 4,08 | 1,42 | 1,74 | 1,47 | 0,75 | 0,74 | | | | | | |
| 3,78 | 1,49 | 1,73 | 1,54 | 0,76 | 0,74 | | | | | | |
| 3,74 | 1,41 | 1,74 | 1,46 | 0,76 | 0,95 | | | | | | |
| 3,19 | 1,43 | 1,77 | 1,57 | 0,74 | 0,92 | | | | | | |
| 3,95 | 1,51 | 1,77 | 1,53 | 0,83 | 0,83 | | | | | | |
| 4,02 | 1,59 | 1,73 | 1,52 | 0,76 | 0,86 | | | | | | |
| 3,88 | 1,61 | 1,83 | 1,49 | 0,77 | 0,9 | | | | | | |
| 3,95 | 1,73 | 1,84 | 1,48 | 0,75 | 0,88 | | | | | | |
| min | 3,19 | min | 1,41 | min | 1,73 | min | 1,46 | min | 0,74 | min | 0,74 |
| max | 4,25 | max | 1,73 | max | 1,93 | max | 2,04 | max | 0,83 | max | 1,07 |
| median | 3,95 | median | 1,50 | median | 1,77 | median | 1,53 | media n | 0,76 | median | 0,89 |
| mean | 3,89 | mean | 1,52 | mean | 1,79 | mean | 1,56 | mean | 0,76 | mean | 0,88 |

| | | | | | | | | | | | |
|-------|------|-------|------|-------|------|-------|------|-------|------|-------|------|
| stdev | 0,29 | stdev | 0,10 | stdev | 0,06 | stdev | 0,17 | stdev | 0,03 | stdev | 0,10 |
|-------|------|-------|------|-------|------|-------|------|-------|------|-------|------|

Berdasarkan Tabel 2 dan dan Tabel 3, rata-rata waktu eksekusi untuk Akses Jalur Pendakian menggunakan Laravel Blade adalah 3,89 detik, sementara Laravel Inertia Vue.js lebih cepat dengan 1,56 detik. Untuk Akses Prediksi Cuaca, Laravel Blade mencatat 1,52 detik, sedangkan Laravel Inertia Vue.js hanya 0,76 detik. Pada Akses Segmentasi Jalur, Laravel Blade memiliki rata-rata 1,79 detik, sementara Laravel Inertia Vue.js lebih unggul dengan 0,88 detik. Hasil ini menunjukkan bahwa Laravel Inertia Vue.js secara konsisten lebih cepat dibandingkan Laravel Blade.

Pengujian pengelolaan data jalur pendakian mencakup pembuatan, pengeditan, dan penghapusan data. Pengujian pembuatan dilakukan dengan *sign in* admin, menambah data jalur, dan menekan “Create.” Pengeditan dilakukan dengan memilih jalur, memperbarui informasi, lalu menekan “Edit.” Penghapusan dilakukan dengan menekan “Hapus” dan mengkonfirmasi penghapusan. Metode pengujian ini dapat dilihat pada Gambar 14.

```

1  <?php
2  namespace Tests\Browser;
3
4  use Laravel\Ui\Browser;
5  use PHPUnit\Framework\Attributes\Group;
6  use Tests\TestCase;
7
8  class ManageMuteTest extends DuskTestCase
9  {
10     protected $route;
11     protected $routes;
12     public function setUp(): void
13     {
14         parent::setUp();
15
16         $this->browse(function (Browser $browser) {
17             $browser->loginAs('');
18         });
19
20         $this->route = (object) [
21             'nama' => 'Mute',
22             'segmentasi' => '1',
23             'deskripsi' => 'Deskripsi Mute',
24         ];
25
26         $this->routeEdit = (object) [
27             'nama' => 'Mute Edit',
28             'segmentasi' => '2',
29             'deskripsi' => 'Deskripsi Mute Edit',
30         ];
31     }
32
33     #[Group('ManageMute')]
34     public function testCreate()
35     {
36         $this->browse(function (Browser $browser) {
37             $browser
38                 ->visit('/admin/rute')
39                 ->clickForText('Tambah Rute')
40                 ->clickLink('Tambah Rute')
41                 ->assertFormFilled('/admin/rute/create')
42                 ->assertFormFilled('create Rute')
43                 ->type('nama', $this->route->nama)
44                 ->type('segmentasi', $this->route->segmentasi)
45                 ->press('create')
46                 ->assertFormFilled('/admin/rute')
47                 ->assertFormFilled('create')
48                 ->assertFormFilled('/admin/rute')
49                 ->assertFormFilled('create')
50                 ->assertFormFilled('create')
51                 ->assertFormFilled('create')
52                 ->assertFormFilled('create')
53                 ->assertFormFilled('create')
54                 ->assertFormFilled('create')
55         });
56     }
57 }

```

Gambar 14. Test case pengujian kelola jalur pendakian

Hasil eksekusi dari pengujian akses jalur pendakian dapat dilihat pada Tabel 3.

Tabel 3. Hasil analisis pengujian kelola jalur pendakian

| Laravel Blade (detik) | | | Laravel Inertia Vue.js (detik) | | | | | | | | |
|-----------------------|-----------|-------------|--------------------------------|-----------|-------------|--------|------|--------|------|--------|------|
| Create Rute | Edit Rute | Delete Rute | Create Rute | Edit Rute | Delete Rute | | | | | | |
| 3,25 | 2,13 | 1,51 | 2,04 | 1,17 | 0,77 | | | | | | |
| 3,08 | 2,07 | 1,52 | 2,03 | 1,13 | 0,78 | | | | | | |
| 3,08 | 2,11 | 1,50 | 2,14 | 1,11 | 0,77 | | | | | | |
| 3,10 | 2,13 | 1,42 | 1,91 | 1,11 | 0,77 | | | | | | |
| 3,00 | 2,14 | 1,54 | 2,06 | 1,20 | 0,78 | | | | | | |
| 2,99 | 2,27 | 1,49 | 2,03 | 1,14 | 0,78 | | | | | | |
| 3,07 | 2,10 | 1,50 | 2,07 | 1,11 | 0,77 | | | | | | |
| 3,13 | 2,07 | 1,51 | 2,05 | 1,14 | 0,89 | | | | | | |
| 3,04 | 2,12 | 1,56 | 2,04 | 1,13 | 0,79 | | | | | | |
| 3,03 | 2,10 | 1,49 | 2,04 | 1,13 | 0,78 | | | | | | |
| min | 2,99 | min | 2,07 | min | 1,42 | min | 1,91 | min | 1,11 | min | 0,77 |
| max | 3,25 | max | 2,27 | max | 1,56 | max | 2,14 | max | 1,20 | max | 0,89 |
| median | 3,08 | median | 2,12 | median | 1,51 | median | 2,04 | median | 1,13 | median | 0,78 |
| mean | 3,08 | mean | 2,12 | mean | 1,50 | mean | 2,04 | mean | 1,14 | mean | 0,79 |

stdev 0,07 stdev 0,06 stdev 0,04 stdev 0,06 stdev 0,03 stdev 0,04

Pada pengujian pengelolaan jalur pendakian pada Tabel 4 dan Tabel 5, waktu rata-rata eksekusi fitur Create Rute menggunakan Laravel Blade adalah 3,08 detik, sedangkan Laravel Inertia Vue.js lebih cepat dengan 2,04 detik. Untuk Edit Rute, Laravel Blade mencatat rata-rata 2,12 detik, sementara Laravel Inertia Vue.js hanya 1,14 detik. Pada Delete Rute, Laravel Blade memiliki rata-rata 1,50 detik, sedangkan Laravel Inertia Vue.js lebih unggul dengan 0,79 detik. Hasil ini menunjukkan bahwa Laravel Inertia Vue.js lebih cepat dibandingkan Laravel Blade dalam semua fitur pengelolaan rute.

Berdasarkan hasil pengujian kinerja sisi klien menggunakan Laravel Dusk, dapat disimpulkan bahwa Laravel Inertia Vue.js selalu unggul dalam setiap fitur dibandingkan dengan Laravel Blade. Hal ini dapat dilihat dari perbedaan nilai rata-rata (mean) pada setiap pengujian.

3.5. Deployment

Kedua *website* yang telah dibangun dan diuji pada tahap sebelumnya di-*deploy* dengan bantuan control panel HestiaCP. HestiaCP adalah panel kontrol hosting berbasis web yang memudahkan pengelolaan server dengan antarmuka yang sederhana. Panel ini mendukung konfigurasi server menggunakan NGINX sebagai web server utama, serta menawarkan fitur-fitur seperti pengelolaan domain, *database*, dan keamanan [5]. Spesifikasi server yang digunakan dapat dilihat pada Tabel 4.

Tabel 4. Spesifikasi server

| No | Spesifikasi | Detail |
|----|----------------|-----------------------------|
| 1 | Sistem Operasi | Ubuntu 22.04.5 LTS |
| 2 | CPU | KVM 4 cores, 1 thread |
| 3 | RAM | 4GB |
| 4 | Storage | 60GB |
| 5 | Control Panel | Hestia Control Panel v1.9.2 |
| 6 | Web Server | NGINX |
| 7 | Database | MySQL v8.3 |
| 8 | PHP | PHP v8.3 |
| 9 | SSL | Let's Encrypt |

Website yang menggunakan Laravel Blade *template engine* di-*deploy* pada domain <https://muncak.id>, sedangkan *website* yang menggunakan Laravel Inertia Vue.js di-*deploy* pada domain <https://inertia.muncak.id>.

Setelah kedua *website* berhasil di-*deploy*, dilakukan optimalisasi SEO yang difokuskan pada aspek *On-Page* SEO [24] dan *Technical* SEO [25]. Analisis hasil optimalisasi ini menggunakan *tool* Screaming Frog SEO Spider [26], dengan hasil sebagai berikut.

| Issue Name | Issue Type | Issue Pr. | URLs | % of * |
|--|------------|-----------|------|--------|
| Content: Low Content Pages | Opport... | Medium | 45 | 76.31% |
| Page Titles: Over 561 Pixels | Opport... | Medium | 4 | 6.25% |
| Page Titles: Over 60 Characters | Opport... | Medium | 4 | 6.25% |
| Images: Over 100 KB | Opport... | Medium | 25 | 51.02% |
| Security: Missing Content-Security-Policy H... | Warning | Low | 122 | 100% |
| Meta Description: Over 985 Pixels | Opport... | Low | 29 | 45.31% |
| H2: Over 70 Characters | Opport... | Low | 30 | 46.88% |
| Security: Missing X-Frame-Options Header | Warning | Low | 122 | 100% |
| H2: Missing | Warning | Low | 6 | 9.38% |
| H1: Duplicate | Opport... | Low | 60 | 93.75% |
| Meta Description: Duplicate | Opport... | Low | 54 | 84.38% |
| H2: Duplicate | Opport... | Low | 54 | 84.38% |
| Meta Description: Missing | Opport... | Low | 6 | 9.38% |
| Content: Readability Very Difficult | Opport... | Low | 15 | 23.44% |
| Meta Description: Below 70 Characters | Opport... | Low | 27 | 42.19% |
| Meta Description: Below 400 Pixels | Opport... | Low | 27 | 42.19% |
| Content: Readability Difficult | Opport... | Low | 18 | 28.13% |
| Hefflang: Missing X-Default | Warning | Low | 64 | 100% |
| Meta Description: Over 155 Characters | Opport... | Low | 29 | 45.31% |
| Security: Missing HSTS Header | Warning | Low | 122 | 100% |
| Images: Missing Size Attributes | Opport... | Low | 49 | 100% |
| Security: Missing X-Content-Type-Options H... | Warning | Low | 122 | 100% |
| Security: Missing Secure Referrer Policy He... | Warning | Low | 122 | 100% |

Gambar 15. Analisis SEO Laravel Blade dan Inertia Vue.js

Berdasarkan Gambar 15, terlihat bahwa *website* yang dibangun menggunakan Laravel dengan Blade *template engine* menghasilkan 23 masalah yang terbagi menjadi 2 tipe, yaitu 7 *warning* dan 16 *opportunity*. Tidak ditemukan masalah bertipe *issue*, sehingga dapat disimpulkan bahwa *website* dengan Laravel Blade *template engine* memiliki kualitas SEO yang cukup baik. Namun, 7 *warning* tersebut dapat diperbaiki untuk meningkatkan kualitas SEO, dan 16 *opportunity* dapat dimanfaatkan lebih lanjut agar hasilnya lebih optimal.

Sedangkan *website* yang dibangun menggunakan Laravel dengan Inertia Vue.js menghasilkan 22 masalah yang terbagi menjadi 2 tipe, yaitu 8 *warning* dan 14 *opportunity*. Tidak ditemukan masalah bertipe *issue*, sehingga dapat disimpulkan bahwa *website* dengan Laravel Inertia Vue.js memiliki kualitas SEO yang cukup baik. Namun, 8 *warning* tersebut sebaiknya diperbaiki untuk meningkatkan kualitas SEO, dan 14 *opportunity* dapat dimanfaatkan lebih lanjut agar hasilnya lebih optimal.

Berdasarkan hasil optimalisasi dan analisis SEO, dapat disimpulkan bahwa kedua *website* memiliki format optimasi SEO yang sama, dengan perbedaan hanya pada sintaks yang digunakan. Hasil analisis menunjukkan bahwa keduanya memiliki kualitas SEO yang cukup baik, sebagaimana terlihat dari simulasi *crawling* menggunakan Screaming Frog SEO Spider yang tidak menunjukkan adanya masalah bertipe *issue*. Namun, masih terdapat 23 masalah pada Laravel Blade dan 22 masalah pada Laravel Inertia Vue.js yang termasuk dalam kategori *warning* dan *opportunity*, yang dapat diperbaiki untuk meningkatkan kualitas SEO lebih lanjut.

4. DISKUSI

Berdasarkan penelitian yang telah dilakukan, penelitian ini berhasil membangun dua *website* informasi jalur pendakian gunung di Indonesia menggunakan Laravel dengan dua pendekatan antarmuka yang berbeda, yaitu Blade *template engine* dan Inertia Vue.js. Dalam hal performa sisi klien, Laravel Inertia Vue.js menunjukkan hasil yang lebih baik dibandingkan Laravel Blade, terutama dalam menangani interaksi pengguna yang kompleks dan dinamis. Namun, Laravel Blade tetap menjadi pilihan yang lebih ringan dan cepat untuk *website* dengan interaksi minimal serta konten yang lebih statis.

Dalam aspek optimalisasi SEO, hasil penelitian menunjukkan bahwa kedua pendekatan memiliki kualitas SEO yang cukup baik dan tidak memiliki perbedaan signifikan dalam implementasinya, kecuali dalam sintaks yang digunakan. Pengujian menggunakan Screaming Frog SEO Spider menunjukkan bahwa baik Laravel Blade maupun Laravel Inertia Vue.js memenuhi standar *on-page* SEO dan *technical* SEO. Dengan demikian, pemilihan antara Blade atau Inertia Vue.js dalam pengembangan *website* dapat disesuaikan dengan kebutuhan proyek, terutama terkait kompleksitas interaksi dan dinamika konten.

Hasil penelitian ini sejalan dengan beberapa penelitian sebelumnya yang membandingkan performa *framework* dalam pengembangan web dan API. Studi sebelumnya menunjukkan bahwa *framework* berbasis JavaScript seperti Express.js memiliki performa yang lebih cepat dibandingkan Laravel dalam menangani request, tetapi Laravel tetap menjadi pilihan yang solid untuk pengembangan aplikasi dengan struktur yang lebih terorganisir. Selain itu, penelitian tentang perbandingan Blade dan

Vue.js juga menunjukkan bahwa Blade lebih cocok untuk aplikasi sederhana, sementara Vue.js lebih optimal untuk aplikasi yang lebih kompleks. Dengan demikian, penelitian ini memberikan kontribusi dalam memahami keunggulan dan keterbatasan masing-masing pendekatan dalam konteks pengembangan *website* berbasis Laravel.

5. KESIMPULAN

Kesimpulan dari penelitian ini adalah sebagai berikut. Dua *website* informasi jalur pendakian gunung di Indonesia berhasil dibangun menggunakan Laravel dengan dua pendekatan antarmuka yang berbeda, yaitu Blade *template engine* dan Inertia Vue.js. *Website* yang dikembangkan dengan Laravel Inertia Vue.js menunjukkan performa yang lebih unggul dalam pengujian kinerja sisi klien dibandingkan dengan *website* yang menggunakan Laravel Blade. Optimalisasi SEO pada kedua *website* relatif sama, dengan perbedaan hanya pada sintaks. Hasil crawling menggunakan Screaming Frog SEO Spider menunjukkan bahwa keduanya memiliki kualitas SEO yang cukup baik. Pengembangan *website* menggunakan Laravel dengan Blade *template engine* lebih cocok untuk *website* yang memiliki sedikit interaksi di sisi klien serta konten yang jarang berubah. Pengembangan *website* menggunakan Laravel dengan Inertia Vue.js lebih sesuai untuk *website* dengan interaksi sisi klien yang lebih banyak serta kompleksitas dan dinamika konten yang lebih tinggi.

DAFTAR PUSTAKA

- [1] S. Novianti, F. Farhah Zamilah, and T. Andrianto, "Perilaku Pendakian Gunung di Masa Kenormalan Baru (Analisis Perspektif Pendaki Gunung di Jawa Barat)," *Journal of Event, Travel and Tour Management*, vol. 2, no. 1, pp. 9–18, Sep. 2022, doi: 10.34013/jett.v2i1.785.
- [2] A. Hidayat *et al.*, "Optimalisasi Jalur Pendakian Cemara Sakti guna Menghidupkan Kembali Wisata Desa Batusari," *PROSIDING KAMPELMAS*, vol. 2, no. 2, 2023, [Online]. Available: <https://mounture.com/catatan-perjalanan/estimasi-waktu-mendaki-gunung-slamet-via->
- [3] A. Azis, Z. M. Putra, M. A. S. Azis, and A. D. Achmad, "Sistem Notifikasi untuk Mengurangi Resiko Tersesat saat Pendakian Menggunakan Received Strength Signal Indicator," *Jurnal Informatika dan Komputer) Akreditasi KEMENRISTEKDIKTI*, vol. 5, no. 3, 2022, doi: 10.33387/jiko.
- [4] J. Saptia Kurnia and F. Risyda, "Rancang Bangun Penerapan Model Prototype dalam Perancangan Sistem Informasi Pencatatan Persediaan Barang Berbasis Web," *Jurnal Sistem Informasi Universitas Suryadarma*, vol. 8, no. 2, 2021, doi: <https://doi.org/10.35968/jsi.v8i2.737>.
- [5] M. Anderton, "Perilaku Pendaki dalam Upaya Kelestarian dan Kebersihan Lingkungan Jalur Pendakian Gunung Kerinci di Kabupaten Kerinci," *Jurusan Geografi Fakultas Ilmu Sosial – UNP*, vol. 5, 2021.
- [6] A. Suprpto and D. Sasongko, "Evaluasi Performa Website Berdasarkan Pengujian Beban dan Stress Menggunakan LoadImpact (Studi Kasus Website IAIN Salatiga)," 2021. [Online]. Available: <https://iainsalatiga.ac.id>
- [7] Laravel, "Laravel Documentation." Accessed: Oct. 01, 2024. [Online]. Available: <https://laravel.com/docs/11.x>
- [8] D. Aipina and H. Witriyono, "Pemanfaatan Framework Laravel dan Framework Bootstrap pada Pembangunan Aplikasi Penjualan Hijab Berbasis Web," *Jurnal Media Infotama*, vol. 18, no. 1, p. 2022, 2022.
- [9] M. Najib, W. Rizqi, B. Zaman, and R. Hendradi, "Analisis Penerapan Metode Search Engine Optimization (SEO) On Page dalam Optimalisasi Konten Website Blog (Studi Kasus: Rumahginjal.id)," *Nusantara Journal of Computers and Its Application*, vol. 8, no. 2, pp. 47–57, 2023, doi: <http://dx.doi.org/10.36564/njca.v8i2.319>.
- [10] F. Alfiana *et al.*, "Apply the Search Engine Optimization (SEO) Method to determine Website Ranking on Search Engines," *International Journal of Cyber and IT Service Management*, vol. 3, no. 1, pp. 65–73, Mar. 2023, doi: 10.34306/ijcitsm.v3i1.126.

-
- [11] Inertia.js, “Inertia.js Documentation Website.” Accessed: Oct. 02, 2024. [Online]. Available: <https://inertiajs.com/>
- [12] A. Amarulloh, “Analisis Perbandingan Performa Web Service Rest Menggunakan Framework Laravel, Django, dan Node.js pada Aplikasi Berbasis Website,” *JURNAL TEKNIK INFORMATIKA STMIK ANTAR BANGSA*, vol. 9, no. 1, 2023.
- [13] W. Hadinata and L. Stianingsih, “Analisis Perbandingan Performa Restfull API antara Express.js dengan Laravel Framework,” *Jurnal Informatika dan Teknik Elektro Terapan*, vol. 12, no. 1, Jan. 2024, doi: 10.23960/jitet.v12i1.3845.
- [14] T. Purwanto, “Analisa Perbandingan Kinerja Rest Api Dengan Framework Flask, Laravel, Dan Express Js,” *Scientia Sacra: Jurnal Sains*, vol. 3, no. 4, 2023, [Online]. Available: <http://127.0.0.1:5000/users>
- [15] H. Jihadi and A. Fikhi Syarabil, “Perbandingan React.js dan Vue.js dalam Pengembangan Aplikasi Web Interaktif: Sebuah Studi Komparatif Penulis 1,” *IBI Kosgoro*, vol. 4, no. 2, pp. 70–79, 2023, doi: 10.55122/junsibi.v4i2.823.
- [16] F. Axza, F. Sofi'ie, and A. Qoiriah, “Analisis Perbandingan Framework Front-End Javascript React dan Vue Pada Pengembangan Website,” *Journal of Informatics and Computer Science*, vol. 05, 2023.
- [17] B. Prakosa, “Pemanfaatan Cypress untuk Pengujian End-to-End (Studi Kasus: Pengembangan Aplikasi Indicar),” Universitas Islam Indonesia, 2023.
- [18] S. Anardani, Y. Yunitasari, and K. Sussolaikah, “Analisis Perancangan Sistem Informasi Monitoring dan Evaluasi Kerjasama Menggunakan UML,” *Remik: Riset dan E-Jurnal Manajemen Informatika Komputer*, vol. 7, no. 1, pp. 522–532, Jan. 2023, doi: 10.33395/remik.v7i1.12070.
- [19] V. J. Sitepu *et al.*, “Perancangan Aplikasi Layanan Bimbingan Konseling Berbasis Website Menggunakan Metode Incremental,” *SNISTIK: Seminar Nasional Inovasi Sains Teknologi Informasi Komputer*, vol. 1, no. 1, 2023, [Online]. Available: <https://ejournal.ust.ac.id/index.php/SNISTIK/article/view/2929>
- [20] E. D. Wahyuni, F. Nurul Ramadha, T. T. Saputra, A. Hendra Maulana, and B. A. Nugroho, “Perbandingan Perancangan Sistem Informasi Akademik Menggunakan Metode Extreme Programming dan Incremental,” *JITET (Jurnal Informatika dan Teknik Elektro Terapan)*, vol. 12, no. 3, pp. 46418–46437, 2024, doi: 10.23960/jitet.v12i3.5424.
- [21] M. Puspitasari and A. Budiman, “Perancangan Sistem Informasi Manajemen Perpustakaan Menggunakan Metode FAST (Framework for the Application System Thinking) (Studi Kasus: SMAN 1 Negeri Katon),” *Jurnal Teknologi dan Sistem Informasi (JTSI)*, vol. 2, no. 2, pp. 69–77, 2021, [Online]. Available: <http://jim.teknokrat.ac.id/index.php/JTSI>
- [22] E. Arribe, D. Silpandi, and H. Mihardi, “Analisis Dan Perancangan Sistem Informasi Absensi Pada PT Wahana Persada Transport Menggunakan Metode Waterfall Dan UML (Unified Modelling Language),” *JJURSISTEKNI (Jurnal Sistem Informasi dan Teknologi Informasi)*, vol. 6, no. 1, pp. 373–381, 2024, [Online]. Available: <https://doi.org/10.52005/jursistekni.v6i1.266>
- [23] V. Valerino *et al.*, “Analisis dan Perancangan Sistem Informasi Aplikasi ‘Homify’ Menggunakan Unified Modeling Language (UML),” *Jurnal Nasional Komputasi dan Teknologi Informasi (JNKTI)*, vol. 7, no. 4, 2024.
- [24] A. A. Murtopo, M. Nursidik, and S. Syefudin, “Optimasi Search Engine Optimization (SEO) On Page untuk Meningkatkan Peringkat Website Hondasukabumi.com Di Google,” *INNOVATIVE: Journal Of Social Science Research*, vol. 4, no. 3, 2024.
- [25] R. Clark, *The Technical SEO Handbook*. Bounce Rank, 2023. [Online]. Available: <https://books.google.co.id/books?id=PtvfEAAAQBAJ&lpg=PT5&ots=kNYduFjCcD&dq=technical seo&lr&pg=PT11#v=onepage&q=technical seo&f=false>
- [26] ScreamingFrog, “Screaming Frog SEO Spider Website.” Accessed: Oct. 02, 2024. [Online]. Available: <https://www.screamingfrog.co.uk/seo-spider/>
-