

Pengembangan Aplikasi Website Pembelajaran untuk Solusi Persamaan Non-Linear Menggunakan Metode Numerik Newton-Raphson dengan Library React.js

Achmad Ardani Prasha^{1*}, Sabrina Laila Mutiara², Muhammad Arfan³, Erland Widyatamaka⁴, Mohamad Yusuf⁵

^{1,2,3,4,5} Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana, Jakarta, Indonesia

*Corresponden Author: 41523010005@student.mercubuana.ac.id

Abstract - This research focuses on developing an educational web application designed to enhance understanding and application of the Newton-Raphson, a fundamental numerical method for solving nonlinear equations. The application is built using the React.js library, allowing for a responsive and interactive user interface. The Newton-Raphson method, known for its efficiency in finding roots of nonlinear equations, is implemented in this application to provide quick and accurate solutions. Key features of the application include numerical visualization of the iteration process, and comprehensive tutorials and documentation on using the Newton-Raphson method. The application was evaluated by testing various nonlinear equations, demonstrating its effectiveness in improving users' conceptual understanding and practical skills. Future development will focus on integrating additional numerical methods and enhancing application performance for an optimal learning experience.

Keywords :

Newton-Raphson;
Nonlinear Equations;
Iteration;
Web Application;
React.js;
Numerical Method;

Article History:

Received: 11-07-2024

Revised: 21-08-2024

Accepted: 30-09-2024

Article DOI : 10.22441/collabits.v1i3.28141

1. PENDAHULUAN

Ketika mencari solusi suatu persamaan, pertanyaan utama yang muncul adalah apakah persamaan tersebut memiliki solusi dan apakah solusinya merupakan akar yang tepat. Persamaan non-linear seringkali tidak dapat diselesaikan dengan metode tradisional seperti pemfaktoran atau rumus eksplisit. Hal ini menuntut penggunaan pendekatan numerik yang dapat diimplementasikan dengan bantuan komputer untuk memproses perhitungan yang kompleks. Dalam konteks penerapan matematika di berbagai ilmu, seperti rekayasa sipil, pemecahan masalah teknik seringkali memerlukan solusi dari persamaan bentuk $f(x) = 0$ atau $f(t) = 0$ yang muncul dari model matematika yang merepresentasikan fenomena nyata. Pada dasarnya, metode untuk menemukan akar persamaan adalah mencari nilai-nilai x untuk $f(x) = 0$, yaitu menemukan titik potong pada grafik $y = f(x)$ dengan sumbu $-X$.^[1]

Metode Newton-Raphson adalah salah satu metode numerik yang paling terkenal untuk mencari akar persamaan non-linear. Metode ini dikenal karena kemampuannya mencapai konvergensi cepat terhadap akar yang diinginkan, asalkan estimasi awal cukup dekat dengan solusi sebenarnya. Metode ini menggunakan turunan pertama dari fungsi yang akan diselesaikan untuk memperkirakan akar berdasarkan titik awal yang

diberikan. Formula iteratif dasar dari Metode Newton-Raphson adalah:

Gambar 1. Formula iteratif dasar^[2]

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

Di mana $f(x)$ adalah fungsi yang akan dicari akarnya, adalah turunan pertama dari fungsi tersebut, dan x_n adalah perkiraan akar pada iterasi ke- n . Keberhasilan metode ini sangat bergantung pada pemilihan tebakan awal yang baik dan keberadaan turunan yang tidak nol di sekitar akar yang dicari.

Metode tradisional tidak digunakan untuk menyelesaikan masalah mencari akar-akar persamaan non-linear. Sebaliknya, metode numerik tertentu, yang kadang-kadang membutuhkan bantuan komputer dalam proses perhitungan, digunakan untuk menyelesaikan masalah ini, seperti melakukan iterasi, yang menggunakan teori kekonvergenan bilangan real. Salah satu contoh adalah persamaan non-linear $xe^{-x} + \cos(2x) = 0$, yang akarnya tidak dapat ditemukan dengan pemfaktoran atau rumus eksplisit. Untuk menyelesaikan persamaan semacam ini, metode iteratif seperti Newton-Raphson menjadi sangat relevan. Pada dasarnya, metode numerik adalah cara mengubah masalah matematika menjadi

formulasi yang dapat diselesaikan dengan operasi aritmatika.[3].

Merancang pemecahan masalah dengan metode numerik membutuhkan kemampuan untuk membuat ide, urutan operasi, atau langkah-langkah pemecahan yang sesuai dengan karakteristik komputer. Selain itu, untuk membuat program komputer yang dapat memecahkan masalah, diperlukan algoritma yang tepat.[4]. Menurut Prawirosusanto, algoritma adalah rangkaian tindakan atau langkah-langkah yang dapat dilakukan oleh komputer.[5]. Pemikiran untuk memperoleh ide dan algoritma, yang berkaitan dengan diagram alur dan langkah-langkah pemrograman, biasanya menjadi masalah dalam menyelesaikan persamaan dengan bantuan komputer. Sangat penting untuk memahami konsep dasar teoritis dari berbagai bidang; seperti analisis real, persamaan diferensial, teknik, statistika, karakteristik komputer, dan bahasa pemrograman yang digunakan saat menyusun algoritma ini.

Seiring dengan perkembangan teknologi informasi, kebutuhan akan aplikasi pembelajaran interaktif dan mudah diakses semakin meningkat. Aplikasi pembelajaran yang mampu memvisualisasikan proses iterasi Metode Newton-Raphson dapat sangat membantu dalam memperdalam pemahaman konsep matematika kompleks ini. Visualisasi dan interaktivitas memungkinkan pengguna untuk melihat langsung bagaimana perubahan dalam estimasi awal atau fungsi yang digunakan dapat mempengaruhi konvergensi solusi. Hal ini sangat berguna bagi siswa dan profesional yang ingin memahami dan menerapkan Metode Newton-Raphson dalam berbagai konteks.

Tujuan dari penelitian ini adalah untuk mengembangkan aplikasi pembelajaran interaktif yang menggunakan metode Newton-Raphson untuk menyelesaikan persamaan non-linear, dengan menggunakan library React.js sebagai dasar pengembangan antarmuka pengguna. Aplikasi ini dirancang untuk memvisualisasikan proses iterasi dan konvergensi solusi secara real-time, sehingga pengguna dapat memahami setiap langkah dalam proses penyelesaian persamaan non-linear.

State of The Art

Metode Newton-Raphson telah menjadi salah satu alat penting dalam analisis numerik dan memiliki sejarah panjang yang bermula dari era Isaac Newton. Metode ini dikenal karena kemampuannya dalam proses menemukan akar-akar persamaan non-linear secara efektif.

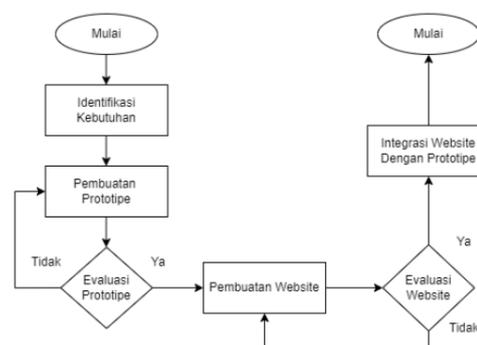
Pada tahun 1669, Isaac Newton menemukan cara untuk mencari akar dari sebarang fungsi yang memiliki turunan pertama, dan pada tahun 1685 John Wallis mempublikasikannya. Pada tahun 1690, Joseph Raphson mengubah dan menerbitkan versi yang lebih menarik dari metode ini, yang sekarang dikenal sebagai metode Newton-Raphson.[6]. Metode ini awalnya diciptakan untuk mencari hampir akar-akar dari nilai nol suatu fungsi bernilai real, dan namanya berasal dari kombinasi nama Newton dan Raphson.

Saat ini, banyak peneliti telah menggunakan metode Newton-Raphson untuk memecahkan masalah pencarian akar-akar persamaan non-linear. Salah satu contohnya adalah penelitian yang dilakukan oleh Qudeiri yang berkaitan dengan lembaran logam, juga dikenal sebagai kertas logam[7]. Dalam penelitian ini, mereka menggunakan metode Newton-Raphson untuk menyelesaikan model persamaan non-linear yang didasarkan pada rumus Lagrange.

2. METODOLOGI PENELITIAN

Penulis menggunakan metode prototyping untuk merancang sistem aplikasi dalam penelitian ini. Ini dilakukan secara bertahap, dimulai dengan mengidentifikasi kebutuhan, kemudian pembuatan prototipe, evaluasi prototipe, pengkodean aplikasi, pengujian aplikasi, dan evaluasi aplikasi, sebelum akhirnya, implementasi. Gambar 2 di bawah ini menampilkan diagram alir yang menunjukkan setiap tahap dari proses ini.

Gambar 2. Diagram yang menunjukkan alur penelitian



Pemilihan Teknologi

React.js telah dipilih karena kemampuan untuk membuat antarmuka pengguna yang dinamis dan responsif melalui pendekatan komponen yang memudahkan pengelolaan dan pengembangan fitur secara modular. Selain itu, React.js mendukung pembaruan efisien dengan virtual DOM, yang memastikan kinerja optimal dengan meminimalkan manipulasi DOM yang berat. Ditambah lagi, ekosistem yang luas dan dukungan komunitas yang aktif menyediakan berbagai pustaka dan alat bantu.

Desain Sistem

Dengan arsitekturnya, aplikasi ini memungkinkan pengguna memasukkan sistem persamaan non-linear apa pun yang mereka inginkan. Struktur aplikasi ini terdiri dari beberapa komponen utama. Pertama, ada antarmuka pengguna yang intuitif dan user-friendly, yang memungkinkan pengguna untuk memasukkan persamaan non-linear dengan mudah. Kedua, terdapat mesin pengolahan yang canggih, yang menggunakan metode Newton-Raphson untuk menyelesaikan persamaan tersebut.

Selain itu, aplikasi ini juga dilengkapi dengan modul output yang berfungsi untuk menampilkan langkah-langkah iterasi yang dilakukan selama proses penyelesaian, serta solusi akhir yang diperoleh. Desain ini dirancang dengan hati-hati untuk memastikan bahwa pengguna tidak hanya dapat memasukkan data dengan mudah, tetapi juga dapat memahami setiap tahap proses penyelesaian yang dilakukan oleh aplikasi. Hal ini memungkinkan pengguna untuk mengikuti perkembangan iterasi dan melihat bagaimana solusi akhir dicapai.

Selain fungsionalitas utamanya, aplikasi ini juga menyediakan penjelasan pembelajaran terkait metode Newton-Raphson untuk menyelesaikan persamaan non-linear. Fitur ini bertujuan untuk memberikan wawasan tambahan kepada pengguna tentang teori di balik metode yang digunakan, membantu mereka memahami konsep dan langkah-langkah yang terlibat dalam proses penyelesaian. Dengan demikian, keseluruhan desain aplikasi ini memastikan kemudahan penggunaan, pemahaman yang jelas tentang proses yang dilakukan, interpretasi hasil yang tepat oleh pengguna, serta pengetahuan tambahan tentang metode yang diterapkan.

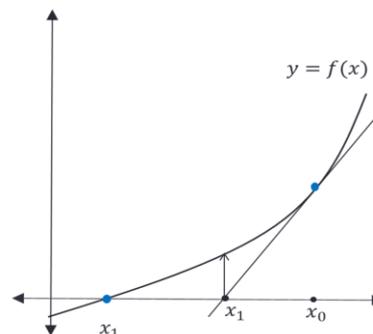
Metode Newton-Raphson

Metode Newton-Raphson adalah salah satu teknik iterasi yang paling efektif untuk menemukan akar dari suatu persamaan nonlinear. Metode ini didasarkan pada ide dasar bahwa garis singgung pada titik tertentu dari kurva fungsi akan melintasi sumbu x di titik di mana akar fungsi berada. Dengan menggunakan pendekatan ini, kita dapat menghasilkan tebakan awal yang lebih baik untuk mendekati solusi sebenarnya.

Metode Newton-Raphson bergantung pada nilai awal sebagai tebakan pertama dan memerlukan kemiringan atau gradien pada titik tersebut. Karena itu, jika nilai turunan di titik awal nol, metode ini tidak dapat digunakan. Metode Newton-Raphson didasarkan pada gagasan bahwa konvergensi barisan biasanya terjadi dengan cepat dan menghasilkan kesalahan kecil. Konsep teoritis dasar metode ini, secara geometris, didapat dari konvergensi barisan titik potong antara garis singgung kurva dan sumbu-X.

Gambar 3 menunjukkan konsep dasar penurunan geometri rumus Newton-Raphson.

Gambar 3. Garis singgung kurva gradien



Gradien garis singgung berdasarkan Gambar 3 x_r adalah

$$m = f'(x_r) = \frac{\Delta y}{\Delta x} = \frac{f(x_r) - 0}{x_r - x_{r+1}}$$

atau

$$f'(x_r) = \frac{f(x_r)}{x_r - x_{r+1}}$$

diperoleh rumus

$$x_{r+1} = x_r - \frac{f(x_r)}{f'(x_r)} ; f'(x_r) \neq 0$$

Ini menunjukkan bahwa antara sumbu-X dan garis singgung kurva di suatu titik terdapat deretan titik potong.

Pengujian

Pengujian dilakukan pada aplikasi untuk memastikan bahwa setiap fitur berfungsi sesuai dengan ekspektasi dan bahwa algoritma yang digunakan dapat memberikan solusi yang akurat. Proses dimulai dengan pengujian unit untuk memastikan bahwa setiap komponen aplikasi bekerja dengan benar secara individual. Setelah itu, pengujian integrasi dilakukan untuk memastikan bahwa semua komponen dapat bekerja sama dengan baik saat digabungkan ke dalam sistem yang utuh.

Selain pengujian unit dan integrasi, pengujian pengguna akhir juga merupakan bagian penting dari proses pengujian aplikasi. Pengujian ini melibatkan pengguna sebenarnya untuk mengevaluasi kegunaan aplikasi secara keseluruhan. Tujuannya adalah untuk mengidentifikasi dan memperbaiki masalah yang mungkin timbul terkait dengan pengalaman pengguna. Dengan mengumpulkan umpan balik dari pengguna akhir, pengembangan dapat membuat penyesuaian dan perbaikan yang diperlukan untuk memastikan bahwa aplikasi tidak hanya berfungsi dengan baik secara teknis, tetapi juga memberikan pengalaman pengguna yang optimal. Proses ini membantu memastikan bahwa produk akhir memuaskan dan memenuhi kebutuhan pengguna.

3. HASIL DAN PEMBAHASAN

Implementasi Metode

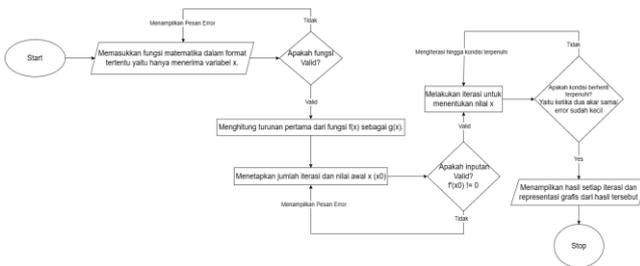
Kami menerapkan metode Newton-Raphson sebagai inti dari perhitungan matematis saat kita membuat aplikasi untuk menyelesaikan persamaan linear. Implementasi ini dilakukan dengan menggunakan bahasa pemrograman

JavaScript. Ini dipilih karena kemampuan pengetikan statis dan fitur pemrograman berorientasi objek, serta framework React.js untuk membangun antarmuka pengguna dan logika server-side.

1. Rancangan Algoritma

Algoritma Newton-Raphson dirancang untuk menemukan akar dari persamaan non-linear dengan menggunakan metode iteratif. Gambar 4 menampilkan flowchart yang mengilustrasikan langkah-langkah yang diambil dalam implementasi algoritma ini:

Gambar 4. Diagram Alur Proses Penghitungan



2. Penjelasan Algoritma

1. Input Fungsi: Pengguna memasukkan fungsi matematika dalam format tertentu yaitu hanya menerima variabel x .
2. Validasi: Memvalidasi fungsi yang diinput. Jika tidak valid, tampilkan pesan kesalahan.
3. Hitung Turunan: Menghitung turunan fungsi dengan rumus

$$y = ax^n$$

$$\frac{dy}{dx} = anx^{n-1}$$

4. Set Iterasi dan x awal (x_0): Pengguna menetapkan jumlah iterasi dan nilai awal (x_0) dengan ketentuan $f'(x_0) \neq 0$.

5. Periksa Validasi Input: Jika ada input yang tidak valid, tampilkan pesan kesalahan yang sesuai.
6. Lakukan iterasi: Jika semua input valid, mulailah proses perulangan:

- a. Tentukan x_1 sebagai titik potong garis singgung yang melalui $(x_0, f(x_0))$ dengan sumbu x . Untuk menentukan x_1 menggunakan formula

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}$$

- b. Selanjutnya dengan cara yang sama ditentukan nilai dari $x_2, x_3,$ dan seterusnya
- c. Iterasi dapat dihentikan bila dua iterasi beruntun menghasilkan hampiran akar yang sama atau ketika $f(x_i) = 0$

7. Tampilkan Hasil: Menampilkan hasil setiap iterasi, termasuk representasi grafik.

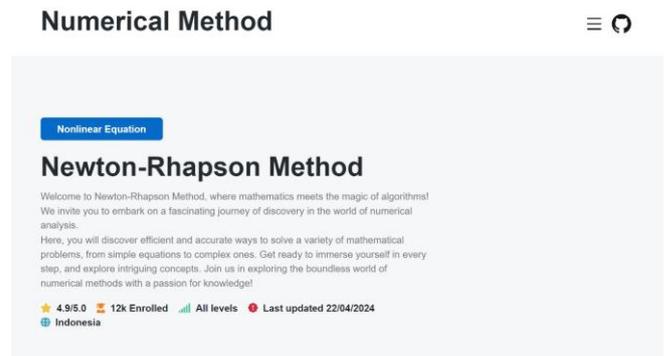
Tampilan Dari Aplikasi

Aplikasi website dapat diakses melalui tautan <https://nr-pro.achmadardani.me/>.

1. Layar Utama Aplikasi

Pengguna diberikan penjelasan singkat tentang prinsip dasar dan manfaat Metode Newton-Raphson di antarmuka utama.

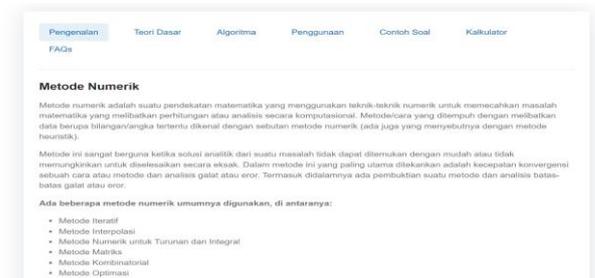
Gambar 5. Tampilan utama aplikasi



2. Tampilan Pengenalan

Pada bagian ini, halaman akan menampilkan berbagai permasalahan atau metode yang dapat diselesaikan menggunakan metode Newton-Raphson. Permasalahan-permasalahan ini mencakup berbagai jenis persamaan non-linear yang sering muncul dalam berbagai konteks, serta metode-metode yang dapat digunakan secara efektif untuk menyelesaikan masalah tersebut.

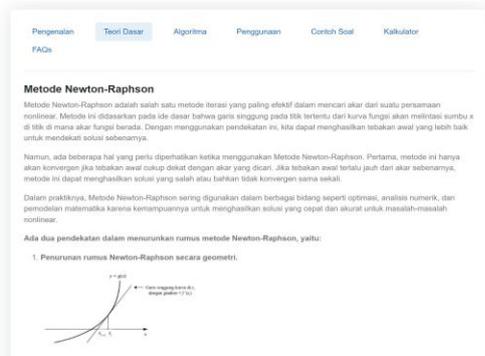
Gambar 6. Tampilan pengenalan



3. Tampilan Teori Dasar

Pada bagian ini, bertujuan untuk memberikan pemahaman yang jelas dan komprehensif mengenai metode Newton-Raphson, termasuk penerapannya, cara kerjanya, dan kapan metode ini akan konvergen atau gagal, serta bagaimana menghentikan iterasi ketika solusi sudah cukup akurat.

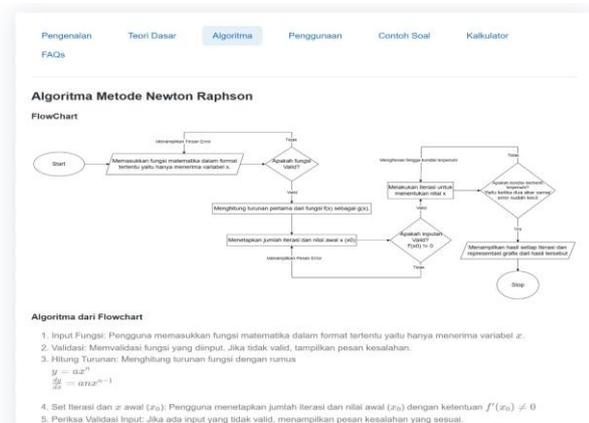
Gambar 7. Tampilan Teori Dasar



4. Tampilan Algoritma

Pada bagian algoritma ini, bertujuan memberikan panduan yang jelas dan sistematis tentang bagaimana algoritma dan flowchart Metode Newton-Raphson bekerja, memastikan setiap langkah dilakukan dengan benar untuk menemukan akar persamaan dengan efisien.

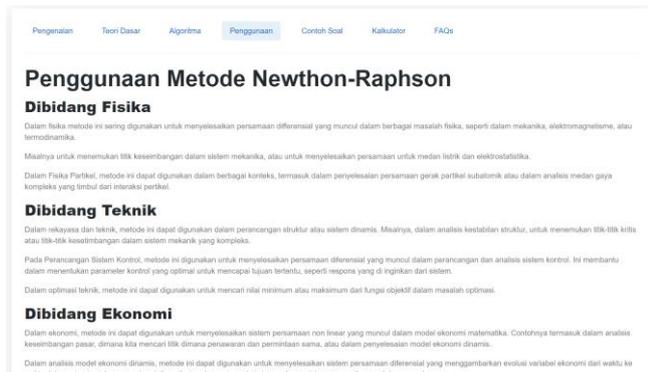
Gambar 8. Tampilan Algoritma



5. Tampilan Penggunaan

Pada bagian ini menjelaskan bagaimana Metode Newton-Raphson digunakan secara luas dan efisien dalam berbagai bidang seperti fisika, teknik, dan ekonomi untuk memecahkan berbagai jenis persamaan dan masalah kompleks.

Gambar 9. Tampilan Penggunaan



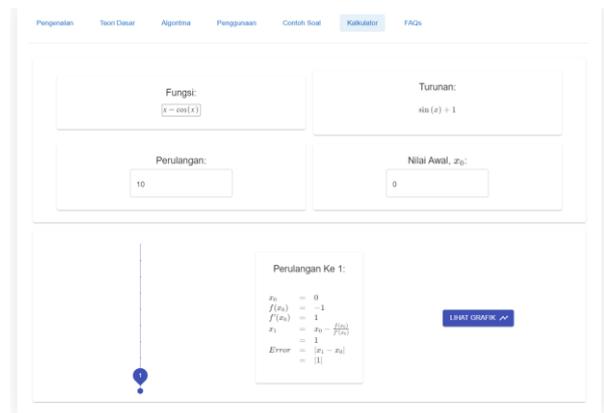
6. Tampilan Contoh Soal

Pada bagian ini, pengguna akan diberikan contoh soal beserta penyelesaiannya yang dituliskan secara detail. Tujuan ini memastikan bahwa setiap langkah dalam metode Newton-Raphson dijelaskan secara terperinci, sehingga pengguna dapat mengikuti dan memahami proses iteratif untuk menemukan akar persamaan non-linear dengan akurasi tinggi.

Gambar 10. Tampilan Contoh Soal

7. Tampilan Kalkulator

Gambar 11. Tampilan Kalkulator



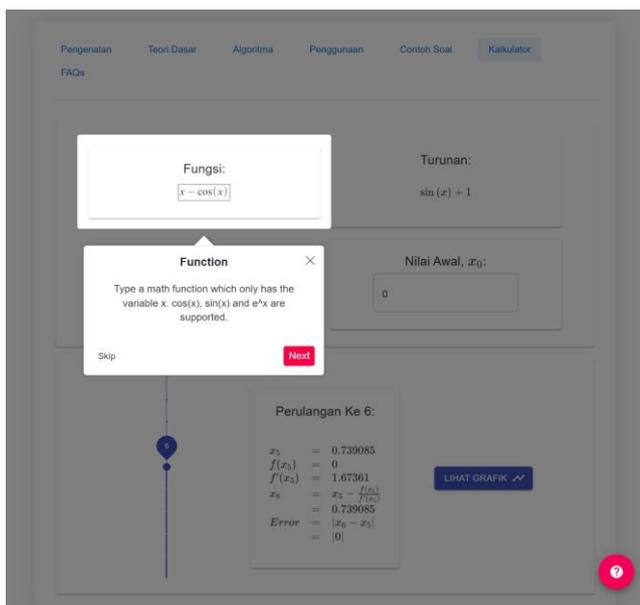
7.1 Fungsi

Pada bagian ini, pengguna dapat memasukkan nilai yang dicari dengan syarat-syarat tertentu.

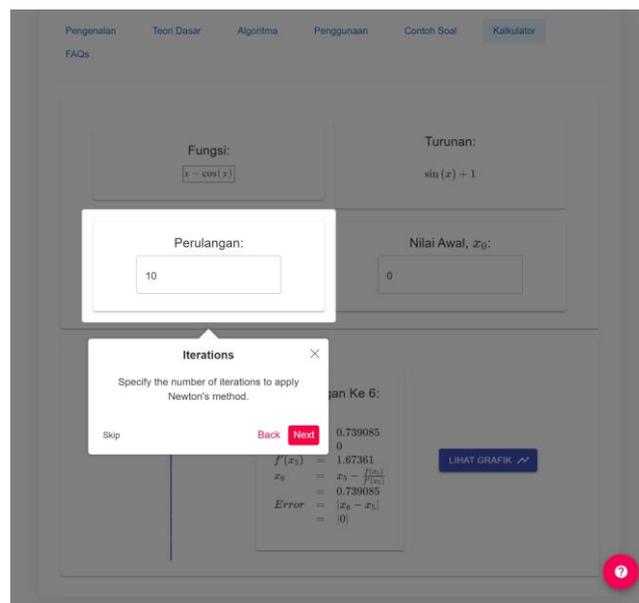
7.3 Perulangan

Pada bagian perulangan, pengguna dapat memasukkan nilai perulangan yang diinginkan.

Gambar 12. Fungsi



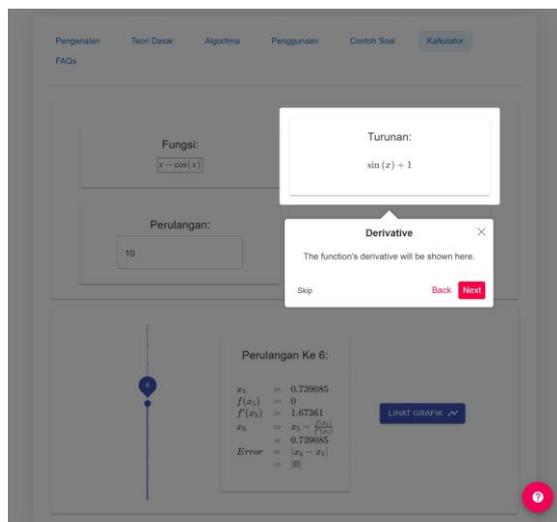
Gambar 14. Perulangan



7.2 Turunan

Bagian ini merupakan hasil turunan dari fungsi dengan nilai yang telah dimasukkan oleh pengguna.

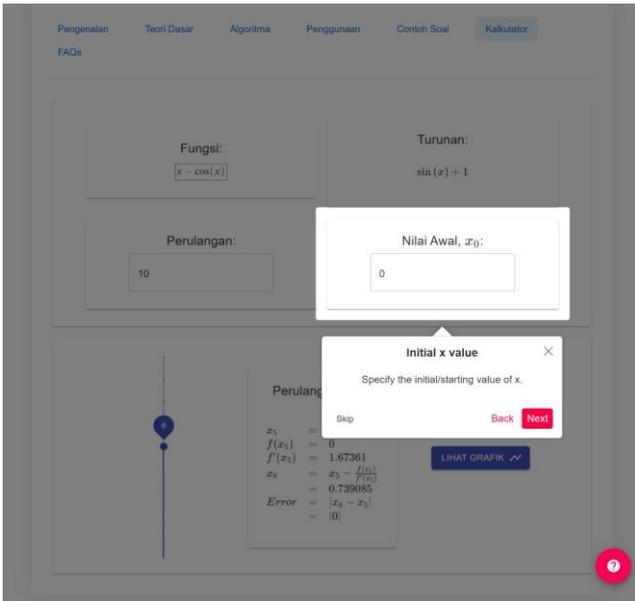
Gambar 13. Turunan



7.4 Nilai Awal

Pada bagian ini, pengguna dapat mengisi nilai awal dengan nilai yang diinginkan oleh pengguna.

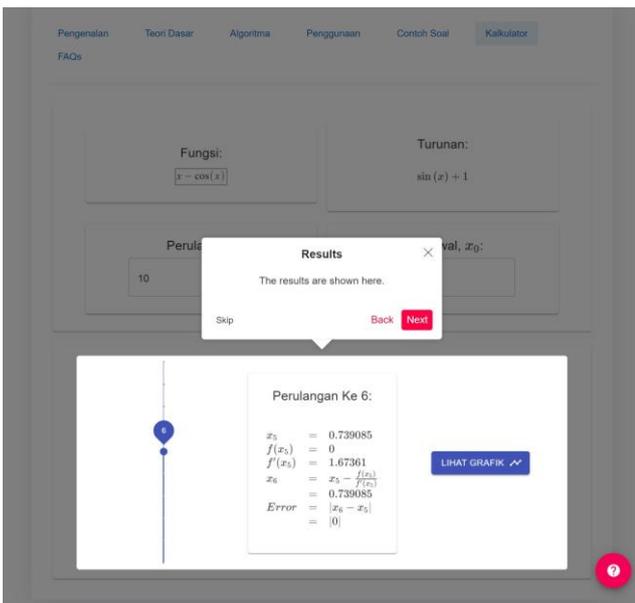
Gambar 15. Nilai Awal



7.6 Results

Bagian ini menampilkan hasil dari perhitungan menggunakan kalkulator ini.

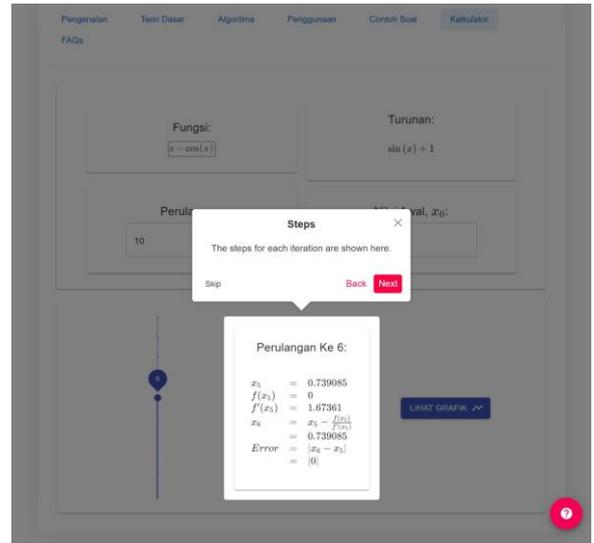
Gambar 16. Results



7.6.1 Steps

Bagian ini menunjukkan langkah-langkah perulangan sampai menemukan jawaban yang benar.

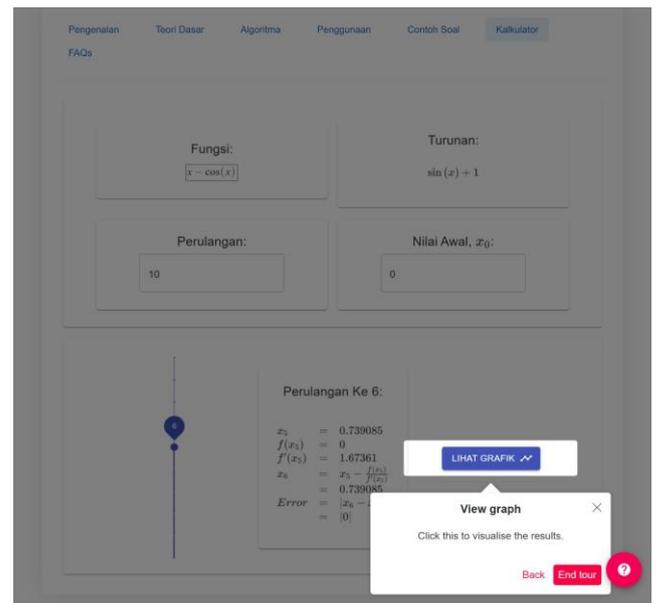
Gambar 17. Steps



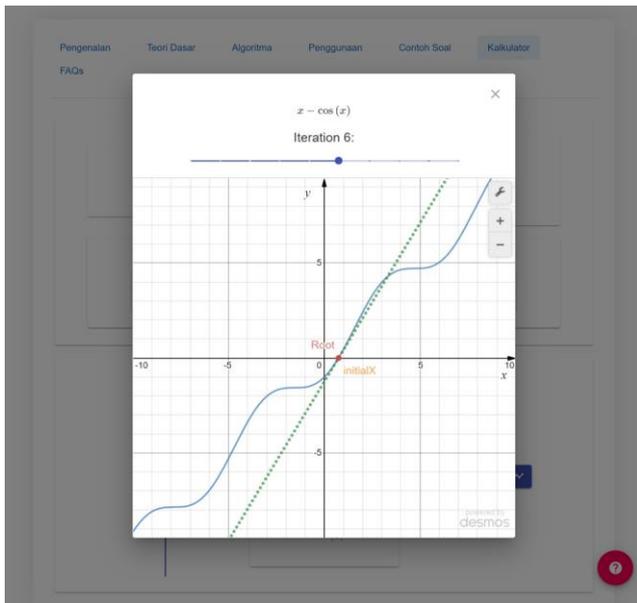
7.6.2 View Graph

Pada bagian ini, pengguna dapat melihat grafik dari tiap-tiap perulangan yang terjadi.

Gambar 18. View Graph



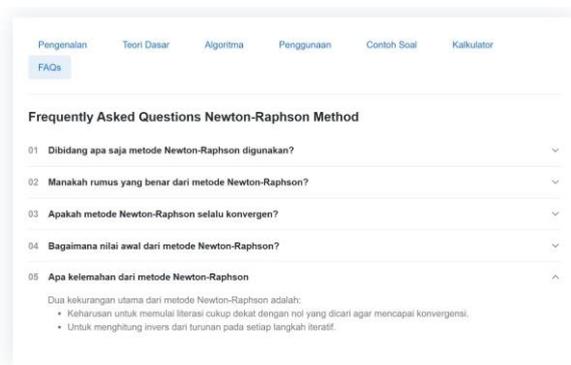
Gambar 18.1. View Graph



8. Tampilan FAQ

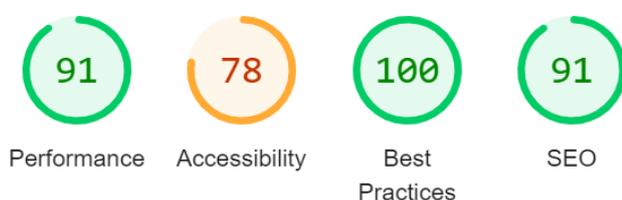
Pada bagian ini terdapat pertanyaan yang sering ditanyakan beserta jawaban dari pertanyaan tersebut yang dirancang untuk memberikan informasi dengan cara yang mudah dipahami dan komprehensif mengenai penggunaan, kelebihan, kelemahan, dan kondisi konvergensi metode Newton-Raphson, serta panduan praktis dalam memilih nilai awal untuk memulai iterasi.

Gambar 19. FAQ



Audit Website

Gambar 20. Ringkasan Audit Website



Gambar di atas menunjukkan hasil uji performa website menggunakan fitur Lighthouse pada Google Chrome. Lighthouse adalah alat otomatis open-source yang digunakan untuk meningkatkan kualitas halaman web. Ini berfungsi sebagai alat untuk mengukur dan mengaudit kualitas kinerja website, aksesibilitas, aplikasi web progresif, dan banyak lagi. Pada pertengahan 2018, Google memperkenalkan alat SEO bernama Lighthouse, yang menawarkan cara terbaik untuk meningkatkan kinerja dan kualitas halaman secara keseluruhan. Google Lighthouse berfungsi sebagai alat yang sepenuhnya otomatis yang memeriksa aksesibilitas dan kinerja.

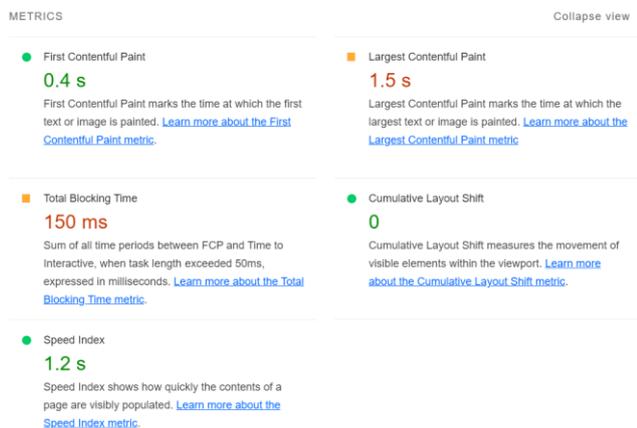
Hasil uji tersebut terdiri dari empat metrik utama, yaitu Kinerja (Performance), Aksesibilitas (Accessibility), Praktik Terbaik (Best Practices), dan Optimasi Mesin Pencari (SEO). Berikut adalah deskripsi dari masing-masing metrik yang diukur:

- Kinerja (Performance):** Kinerja mengukur seberapa cepat dan efisien sebuah website dalam memuat dan menampilkan konten kepada pengguna. Website memperoleh skor 91, yang menunjukkan bahwa situs web ini memiliki kecepatan dan efisiensi loading halaman yang luar biasa. Ini mencakup berbagai aspek seperti waktu muat konten utama, interaktivitas, dan stabilitas visual.
- Aksesibilitas (Accessibility):** Aksesibilitas mengukur seberapa baik sebuah website dapat diakses dan digunakan oleh semua pengguna, termasuk mereka yang memiliki disabilitas. Website memperoleh skor 78, yang menunjukkan bahwa terdapat beberapa area yang perlu diperbaiki untuk membuatnya lebih mudah bagi semua orang, termasuk orang dengan kebutuhan khusus. Ini mencakup aspek seperti penggunaan label yang benar pada elemen-elemen form, kontras warna yang memadai, dan navigasi yang dapat diakses melalui keyboard.
- Praktik Terbaik (Best Practices):** Praktik Terbaik mengukur sejauh mana sebuah website mematuhi standar dan rekomendasi terbaik dalam pengembangan web. Ini mencakup aspek-aspek seperti keamanan (misalnya, penggunaan HTTPS), penggunaan teknologi yang tepat (misalnya, menghindari penggunaan API yang sudah usang), dan adopsi standar web terbaru (misalnya, HTML5). Website memperoleh skor sempurna 100, yang menunjukkan bahwa website ini mematuhi semua praktik terbaik yang direkomendasikan dalam pengembangan web. Ini mencakup keamanan, penggunaan teknologi yang sesuai, dan adopsi standar web terbaru.
- Optimasi Mesin Pencari (SEO):** Optimasi Mesin Pencari mengukur seberapa baik sebuah website dioptimalkan untuk mesin pencari seperti Google. Website memperoleh skor 91, yang menunjukkan bahwa website ini dioptimalkan dengan baik untuk mesin pencari. Ini mencakup aspek-aspek seperti penggunaan meta tag yang tepat, struktur URL yang ramah mesin pencari, dan kecepatan loading yang baik. Secara keseluruhan, hasil uji Lighthouse ini menunjukkan bahwa website memiliki performa yang sangat baik dalam hal kinerja, praktik terbaik, dan SEO, namun masih

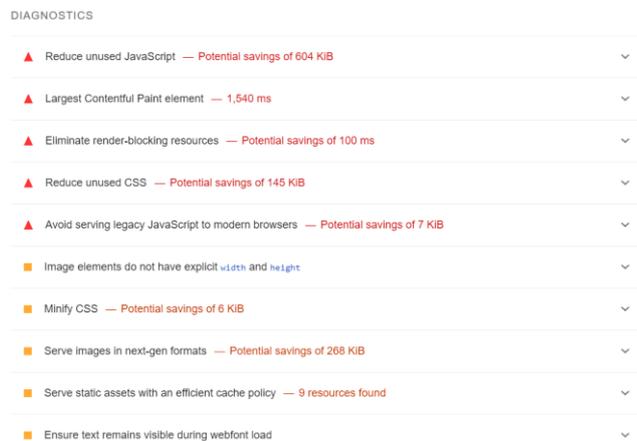
memerlukan beberapa perbaikan pada aspek aksesibilitas untuk mencapai performa yang optimal bagi semua pengguna.

Kinerja (Performance)

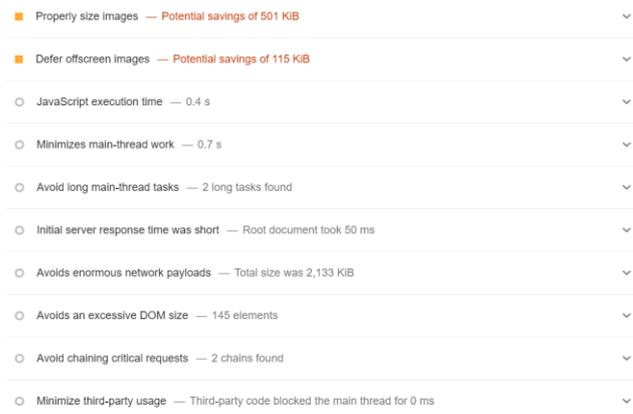
Gambar 21. Uji Performa



Gambar 22. Hasil Diagnosis Performa



Gambar 23. Hasil Diagnosis Performa (Lanjutan)



Kinerja halaman web dinilai berdasarkan beberapa metrik utama:

1. First Contentful Paint (FCP): 0.4 detik - Waktu ini menunjukkan seberapa cepat konten pertama muncul di layar setelah pengguna mulai mengakses halaman.
2. Largest Contentful Paint (LCP): 1.5 detik - Metrik ini mengukur waktu yang dibutuhkan untuk menampilkan elemen terbesar yang terlihat di viewport.
3. Total Blocking Time (TBT): 150 ms - Mengukur jumlah waktu ketika tugas panjang memblokir thread utama dan menunda interaksi pengguna.
4. Cumulative Layout Shift (CLS): 0 - Ini adalah metrik yang mengukur stabilitas visual halaman dengan menilai pergeseran tata letak yang tidak diharapkan.
5. Speed Index: 1.2 detik - Menunjukkan seberapa cepat konten halaman terlihat selama proses pemuatan.

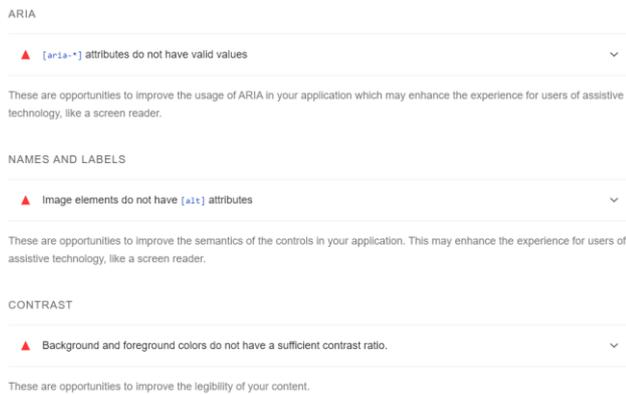
Rekomendasi Perbaikan:

- Kurangi JavaScript yang tidak terpakai: Penghematan potensial sebesar 604 KiB.
- Hapus sumber daya yang menghalangi rendering: Penghematan potensial sebesar 100 ms.
- Kurangi CSS yang tidak terpakai: Penghematan potensial sebesar 145 KiB.
- Meminimalkan CSS: Penghematan potensial sebesar 6 KiB.
- Gunakan format gambar generasi berikutnya: Penghematan potensial sebesar 268 KiB.
- Properly size images: Penghematan potensial sebesar 501 KiB.
- Defer offscreen images: Penghematan potensial sebesar 115 KiB.
- Meminimalkan kerja pada main-thread: 0.7 detik.
- Hindari tugas main-thread yang panjang: 2 tugas panjang ditemukan.

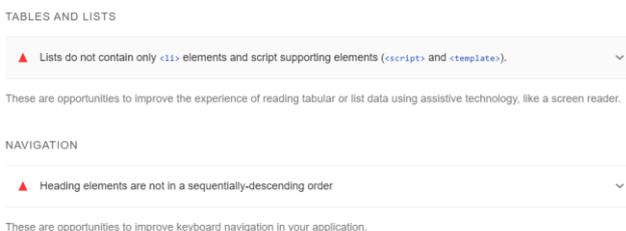
Kesimpulan: Kinerja keseluruhan situs cukup baik dengan skor 91, namun masih ada ruang untuk perbaikan terutama dalam pengoptimalan sumber daya dan pemuatan gambar.

Aksesibilitas (Accessibility)

Gambar 24. Hasil Diagnosis Aksesibilitas



Gambar 25. Hasil Diagnosis Aksesibilitas (Lanjutan)



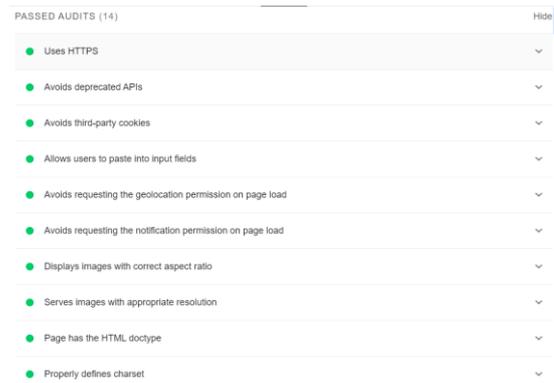
Aksesibilitas web dinilai dengan skor 78. Ada beberapa hal penting yang harus diperhatikan untuk meningkatkan aksesibilitas:

1. [aria-*] attributes: Atribut ARIA tidak memiliki nilai yang valid, yang penting untuk pengguna teknologi bantu seperti pembaca layar.
2. Elemen gambar tidak memiliki atribut [alt]: Ini mempengaruhi pengguna yang mengandalkan deskripsi teks untuk memahami konten gambar.
3. Rasio kontras: Warna latar depan dan latar belakang tidak sebanding, sehingga sulit dibaca.
4. Urutan heading tidak berurutan: Ini penting untuk navigasi menggunakan keyboard.
5. Daftar tidak hanya berisi elemen : Ini dapat mempengaruhi pengalaman membaca data tabular atau daftar menggunakan teknologi bantu.

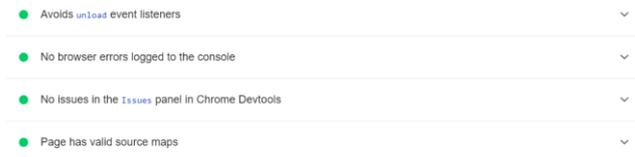
Kesimpulan: Untuk meningkatkan aksesibilitas, situs harus memperbaiki penggunaan atribut ARIA, menambahkan atribut alt pada gambar, meningkatkan rasio kontras warna, dan memastikan struktur heading yang tepat.

Praktik Terbaik (Best Practice)

Gambar 26. Hasil Diagnosis Praktik Terbaik



Gambar 27. Hasil Diagnosis Praktik Terbaik (Lanjutan)



Praktik Terbaik dinilai dengan skor sempurna 100. Ini menunjukkan bahwa situs mengikuti semua praktik keamanan dan kualitas yang direkomendasikan, termasuk:

- **Uses HTTPS:** Menggunakan HTTPS memastikan bahwa komunikasi antara pengguna dan server dienkripsi, menjaga keamanan data pengguna dari serangan man-in-the-middle. Ini adalah langkah penting untuk melindungi privasi pengguna dan integritas data.
- **Avoids deprecated APIs:** Menghindari penggunaan API yang sudah usang atau tidak didukung lagi penting untuk menjaga kompatibilitas dan kinerja aplikasi web. API yang usang bisa berhenti berfungsi sewaktu-waktu dan mungkin memiliki masalah keamanan.
- **Avoids third-party cookies:** Menghindari penggunaan cookie pihak ketiga dapat meningkatkan privasi pengguna. Cookie pihak ketiga sering digunakan untuk melacak pengguna di berbagai situs web, yang bisa menjadi masalah privasi.
- **Allows users to paste into input fields:** Memungkinkan pengguna untuk menempelkan teks ke dalam bidang input penting untuk kemudahan penggunaan dan aksesibilitas. Pembatasan ini dapat menghalangi pengguna, terutama mereka yang menggunakan alat bantu atau memiliki keterbatasan fisik.
- **Avoids requesting the geolocation permission on page load:** Tidak meminta izin geolokasi saat halaman dimuat membantu menjaga privasi pengguna dan menghindari pengalaman pengguna yang buruk. Pengguna cenderung merasa terganggu jika diminta izin geolokasi tanpa konteks yang jelas.
- **Avoids requesting the notification permission on page load:** Sama seperti izin geolokasi, tidak meminta izin notifikasi saat halaman dimuat dapat meningkatkan pengalaman pengguna. Pengguna biasanya lebih suka memberikan izin ini setelah mereka memahami mengapa diperlukan.

- Displays images with correct aspect ratio: Menampilkan gambar dengan rasio aspek yang benar memastikan tampilan yang konsisten dan estetis di seluruh perangkat. Ini mencegah distorsi gambar yang dapat mengganggu pengalaman visual pengguna.

- Serves images with appropriate resolution: Menyajikan gambar dengan resolusi yang tepat membantu mengoptimalkan waktu pemuatan halaman dan kualitas gambar. Gambar yang terlalu besar dapat memperlambat situs, sementara gambar dengan resolusi terlalu rendah dapat terlihat buram.

- Page has the HTML doctype: Menentukan doctype HTML di awal dokumen memastikan browser mem-parsing dan menampilkan halaman dengan standar yang benar, mengurangi kemungkinan terjadinya perbedaan tampilan antar browser.

- Properly defines charset: Mendefinisikan charset dengan benar (biasanya UTF-8) memastikan bahwa teks di halaman web ditampilkan dengan benar, termasuk karakter khusus dan simbol. Ini penting untuk kompatibilitas internasional dan mencegah masalah encoding.

- Avoids unload event listeners: Menghindari penggunaan pendengar acara unload membantu meningkatkan kinerja dan mencegah potensi kebocoran memori. Acara unload dapat menyebabkan masalah dengan navigasi halaman dan pengalaman pengguna yang buruk.

- No browser errors logged to the console: Tidak ada kesalahan yang tercatat di konsol browser. Ini berarti selama pengujian, tidak ditemukan error JavaScript atau masalah lainnya yang dicatat oleh konsol browser, yang menunjukkan bahwa halaman berjalan tanpa masalah yang mengganggu kinerja atau fungsionalitasnya.

- No issues in the Issues panel in Chrome Devtools: Tidak ada masalah yang ditemukan di panel Issues di Chrome DevTools. Panel Issues ini dirancang untuk membantu pengembang mengidentifikasi dan memperbaiki masalah yang berkaitan dengan keamanan, kinerja, aksesibilitas, dan lainnya. Tidak adanya masalah di panel ini menunjukkan bahwa halaman web sesuai dengan standar yang disarankan oleh Chrome DevTools dan telah dioptimalkan dengan baik.

- Page has valid source maps: Halaman memiliki source maps yang valid. Source maps adalah file yang memetakan kode sumber asli (seperti file JavaScript atau CSS yang telah dikompilasi/minified) ke dalam versi yang telah diubah (misalnya, minified). Ini memungkinkan pengembang untuk melakukan debug dengan lebih mudah karena mereka dapat melihat kode asli alih-alih versi yang telah diubah. Validnya source maps menunjukkan bahwa proses build dan minifikasi telah dilakukan dengan benar dan dapat digunakan untuk debugging jika diperlukan.

Kesimpulan: Situs mematuhi semua praktik terbaik yang direkomendasikan, menjamin keamanan dan kualitas yang baik.

Optimisasi Mesin Pencari (SEO)

Gambar 28. Hasil Diagnosis Optimisasi Mesin Pencari

(SEO)



Optimisasi Mesin Pencari (SEO) mendapat skor 91, dengan beberapa poin penting:

1. **Gambar tidak memiliki atribut [alt]:** Ini penting untuk SEO karena membantu mesin pencari memahami konten gambar.
2. **Format HTML yang baik:** memastikan bahwa format HTML memungkinkan crawler memahami konten dengan lebih baik.

Kesimpulan: Meskipun skor SEO tinggi, ada beberapa area yang bisa diperbaiki seperti penambahan atribut alt pada gambar untuk meningkatkan pemahaman konten oleh mesin pencari.

4. KESIMPULAN

Studi ini menghasilkan aplikasi web interaktif yang menyelesaikan persamaan non-linear dengan menggunakan teknik Newton-Raphson. Aplikasi ini dibangun dengan library React.js, yang memungkinkan pembuatan antarmuka pengguna yang responsif dan interaktif. Aplikasi ini menyediakan visualisasi numerik dari proses iterasi, analisis konvergensi, serta tutorial dan dokumentasi yang komprehensif mengenai penggunaan metode Newton-Raphson. Penggunaan React.js terbukti efektif dalam menciptakan antarmuka pengguna yang dinamis dan responsif, yang memudahkan pengelolaan dan pengembangan fitur secara modula. Aplikasi ini juga menunjukkan performa yang baik berdasarkan hasil audit dengan Google Lighthouse, dengan skor tinggi dalam hal kinerja, aksesibilitas, praktik terbaik, dan SEO. Aplikasi ini dapat menjadi alat yang sangat berguna bagi siswa dan profesional dalam memahami dan menerapkan metode Newton-Raphson dalam berbagai konteks. Diharapkan dengan adanya aplikasi ini, proses pembelajaran dan penerapan metode numerik dalam pemecahan masalah non-linear dapat menjadi lebih efektif dan efisien.

5. UCAPAN TERIMA KASIH

Chien Yoong Fong [8] atas kontribusinya yang berharga dalam proyek ini. Beliau telah menyediakan dan mengizinkan kami untuk menggunakan source code untuk bagian kalkulator dengan modifikasi. Kontribusinya sangat penting bagi keberhasilan penelitian ini.

DAFTAR PUSTAKA

- [1] Susila, I. 1994. "Teknik Penyelesaian Persamaan Non-Linear." *Jurnal Matematika dan Aplikasinya*, 10(1), 45-50.
- [2] Rochmad. 2013. *Aplikasi Metode Newton-Raphson Untuk Menghampiri Solusi Persamaan Non*

Linear. Jurnal MIPA. 36 (2): 193-200.

[3] Chapra, S. C., & Canale, R. P. 1991. *Numerical Methods for Engineers*. New York: McGraw-Hill.

[4] Conte, S. D., & de Boor, C. 1993. *Elementary Numerical Analysis: An Algorithmic Approach*. New York: McGraw-Hill.

[5] Prawirosusanto S. 1997. "*Pengantar Metode Numerik*." Yogyakarta: FMIPA UGM.

[6] D. M. Bressoud, "A Radical Approach to Real

Analysis", MAA, 2006.

[7] Qudeiri JA, FA Khadra, A Al-Ahmari & U Umar. 2013. Effect of Material and Geometrical Parameters on the Springback of Metallic Sheets. *Life Science Journal*. 10(2): 1531-1536.

[8] FongYoong. (n.d.). *Numerical-methods-js* [GitHub repository]. GitHub.
<https://github.com/FongYoong/numerical-methods-js>