Rapport d'intervention

Client : Nina Carducci

I. Performance du Site Avant et Après Optimisation

1. Avant Optimisation :

Lighthouse_ordinateur_avant.pdf
Lighthouse_mobile_avant.pdf

2. Après Optimisation :

Lighthouse_ordinateur_après.pdf
Lighthouse_mobile_après.pdf

II. Détails des Optimisations et Interventions effectuées

1. Optimisations des Images

Le projet comportait initialement 17 images pesant un total de 31,5 Mo.

Les modifications suivantes ont été apportées :

- Optimisation et compression des images pour réduire leur poids.
- Redimensionnement explicite des images pour correspondre aux dimensions affichées.
- Introduction du format d'image moderne WebP pour améliorer les performances.
- Chargement différé (lazy-load) des images d'arrière-plan.

Après les modifications, le poids total des images est de 558 ko, avec un gain d’environ 98%.

2. Améliorations SEO On-Page

a. SEO Contenu

- Sélection d'un mot-clé cible pour optimiser le contenu. (photographe évènementiel)
- Améliorations sémantiques on-page pour enrichir le contenu. (modification sémantique on-page)

b. SEO Balisage / Web Sémantique

- Ajout de l'attribut "lang" à la balise `<html>` pour spécifier la langue du contenu.
- Révision de la structure de balisage Hn pour améliorer la hiérarchisation des titres.
- Optimisation du web sémantique visible côté SERP (meta-title, description, robots).
- Ajout de balises `<strong>` pour mettre en évidence le contenu important.
- Vérification et amélioration de la cohérence web sémantique du DOM.

c. SEO Performances

- Report des scripts en fin de <body>.
- Report des éléments CSS non essentiels en fin de <body>.

3. Optimisations pour l'Accessibilité

a. Avant Optimisation

Rapport_accessibilité_Wave_avant.png

b. Après Optimisation

Rapport_accessibilité_Wave_après.png

- Ajout d'attributs "alt" pour toutes les images en vue d'une meilleure expérience pour les lecteurs d'écran.
- Association des libellés "name" au formulaire pour une navigation plus aisée avec les lecteurs d'écran.
- Ajout d'attributs "aria-label" aux liens externes pour fournir des informations contextuelles.
- Amélioration du contraste des éléments on-page pour respecter les normes d'accessibilité.

III. Ordre d'Intervention des Améliorations

(La liste des actions d'amélioration réalisées a été marquée comme "effectuées" conformément à l'ordre d'intervention initial)

1. ☑ Choisir un mot-clé cible
2. ☑ Ajouter un attribut lang à la balise `<html>`
3. ☑ Revoir le web sémantique visible côté SERP
4. ☑ Revoir la structure de balisage Hn
5. ☑ Apporter des améliorations sémantiques on-page
6. ☑ Ajout de balises `<strong>`
7. ☑ Vérifier la cohérence web sémantique du DOM
8. ☑ Réduire le poids des images
9. ☑ Redimensionner les images explicitement
10. ☑ Utiliser les nouveaux formats d'image (WebP)
11. ☑ Afficher les images d'arrière-plan au survol (lazy-load)
12. ☑ Ajouter des attributs alt aux images
13. ☑ Ajouter des aria-label aux liens externes
14. ☑ Associer le formulaire à des libellés name
15. ☑ Reporter les scripts non essentiels
16. ☑ Augmenter le contraste des éléments on-page
17. ☑ Vérifier le responsive design
18. ☑ Ajout des meta réseaux sociaux
19. ☑ Ajout des Rich Snippets locaux (Schema.org)
20. ☑ Correction du bug d’affichage CSS Portfolio
21. ☑ Correction du bug navigationnel des images focalisées (avant/après)
22. ☑ Épreuve Lighthouse
23. ☑ Épreuve Wave

Cahier de recette :

Bugs	Action	Résultat initial	Résultat après résolution	Statut
Bug d'affichage css portfolio	Correction du bug d'affichage côté code css.	Affichage incorrect des éléments du portfolio.	Affichage correct et cohérent du portfolio.	Résolu
Bug navigationnel des images focalisées (avant/après)	Correction du bug de navigation des images focalisées, côté script js	Navigation incorrecte entre les images avant et après l'optimisation.	Navigation fluide et correcte entre les images avant et après l'optimisation.	Résolu

Optimisations et actions	Action	Résultat initial	Résultat après résolution	Statut
Choisir un mot-clé cible	Sélection d'un mot-clé cible pour optimiser le contenu.	N/a	Mot-clé cible sélectionné.	Effectué
Ajouter un attribut "lang" à la balise <html>	Ajout de l'attribut "lang" pour spécifier la langue du contenu.	N/a	Attribut "lang" ajouté.	Effectué
Revoir le web sémantique visible côté serp	Optimisation du web sémantique visible côté serp (meta-title, description, robots).	N/a	Web sémantique optimisé pour la serp.	Effectué
Revoir la structure de balisage hn	Révision de la structure de balisage hn pour améliorer la hiérarchisation des titres.	N/a	Structure de balisage hn optimisée.	Effectué
Apporter des améliorations sémantiques on-page	Améliorations sémantiques on-page pour enrichir le contenu.	N/a	Contenu enrichi avec des améliorations sémantiques.	Effectué
Ajout de balises	Ajout de balises pour mettre en évidence le contenu important.	N/a	Balises ajoutées.	Effectué
Vérifier la cohérence web sémantique du dom	Vérification et amélioration de la cohérence web sémantique du dom.	N/a	Cohérence web sémantique vérifiée et améliorée.	Effectué
Réduire le poids des images	Optimisation et compression des images pour réduire leur poids.	Images pesant un total de 31,5 mo.	Poids total des images réduit à 558 ko.	Effectué
Redimensionner les images explicitement	Redimensionnement explicite des images pour correspondre aux dimensions affichées.	N/a	Images redimensionnées correctement.	Effectué
Utiliser les nouveaux formats d'image (webp)	Introduction du format d'image moderne webp pour améliorer les performances.	N/a	Format webp utilisé pour améliorer les performances.	Effectué
Afficher les images d'arrière-plan au survole (lazy-load)	Chargement différé (lazy-load) des images d'arrière-plan.	N/a	Chargement différé des images d'arrière-plan implémenté.	Effectué

Ajouter des attributs "alt" aux images	Ajout d'attributs "alt" pour toutes les images en vue d'une meilleure expérience pour les lecteurs d'écran.	N/a	Attributs "alt" ajoutés à toutes les images.	Effectué
Ajouter des "aria-label" aux liens externes	Ajout d'attributs "aria-label" aux liens externes pour fournir des informations contextuelles.	N/a	Attributs "aria-label" ajoutés aux liens externes.	Effectué
Associer le formulaire à des libellés "name"	Association des libellés "name" au formulaire pour une navigation plus aisée avec les lecteurs d'écran.	N/a	Libellés "name" associés au formulaire.	Effectué
Reporter les scripts non essentiels	Report des scripts en fin de <body>.	N/a	Scripts non essentiels reportés en fin de <body>.	Effectué
Augmenter le contraste des éléments on-page	Amélioration du contraste des éléments on-page pour respecter les normes d'accessibilité.	N/a	Contraste des éléments on-page amélioré.	Effectué

Note : toutes les actions prévues ont été effectuées avec succès, et les problèmes identifiés ont été résolus pour améliorer la performance, l'accessibilité et l'optimisation seo du site web de nina carducci.

```
<!DOCTYPE html>
<html lang="fr">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Nina Carducci | Photographe pour immortaliser vos évènements</title>
    <meta name="description" content="Explorez mon portfolio photos d'événements, mariages et concer
    <meta name="robots" content="index, follow">
    <!-- Open Graph / Facebook -->
    <meta property="og:type" content="website" />
    <meta property="og:title" content="Immortalisez vos évènements avec la photographie Nina Carducc
    <meta property="og:description" content="Plongez dans mon portfolio : des événements aux mariage
    <meta property="og:image" content="./assets/images/nina-resized.webp"/>
    <!-- Twitter -->
    <meta property="twitter:card" content="summary_large_image" />
    <meta property="twitter:title" content="Immortalisez vos évènements avec la photographie Nina Ca
    <meta property="twitter:description" content="Plongez dans mon portfolio : des événements aux ma
    <meta property="twitter:image" content="./assets/images/nina-resized.webp" />
    <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.3.1/dist/css/bootstrap.min.css" rel="styles
    <link rel="stylesheet" href="./assets/style.css">
    <script type="application/ld+json">
        {
          "@context": "http://schema.org",
          "@type": "LocalBusiness",
          "name": "Nina Carducci",
          "address": {
            "@type": "PostalAddress",
            "streetAddress": "68 avenue Alsace-Lorraine",
            "addressLocality": "Bordeaux",
            "postalCode": "33200"
```

LocalBusiness

All (1)

LocalBusiness

0 ERREUR 0 AVERTISSEMENT

@type	LocalBusiness
name	Nina Carducci
telephone	05 56 67 78 89
openingHours	Mo-Fr 10:00-19:00
address	
@type	PostalAddress
streetAddress	68 avenue Alsace-Lorraine
addressLocality	Bordeaux
postalCode	33200

WAVE
web accessibility evaluation tool

powered by
WebAIM

Styles: OFF ON

Summary

Summary | Details | Reference | Order | Structure | Contrast

17 Errors	1 Contrast Errors
7 Alerts	6 Features
16 Structural Elements	10 ARIA

The following apply to the entire page:

h3 Nina Carducci

À propos Galerie Service Contact

WAVE
web accessibility evaluation tool

powered by
WebAIM

Styles: OFF ON

Summary

Summary | Details | Reference | Order | Structure | Contrast

0
Errors

0
Contrast Errors

4
Alerts

22
Features

14
Structural Elements

10
ARIA

View details ›

Congratulations! No errors were detected! Manual testing is still necessary to ensure compliance and optimal accessibility.

The following apply to the entire page:

fr

Nina Carducci

À propos Galerie Service Contact

logo instagram

https://maximec31.github.io/Nina-Carducci/

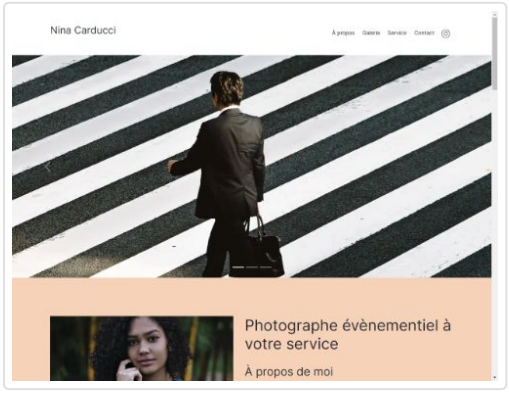
Performance	Accessibility	Best Practices	SEO	PWA
98	100	100	100	—

98

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 50–89 90–100

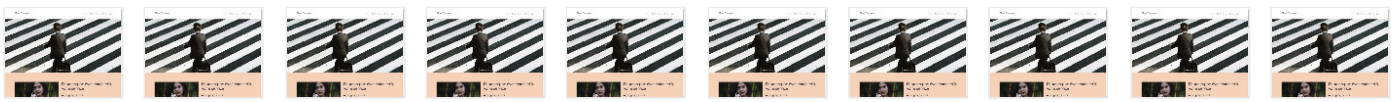


METRICS

Expand view

First Contentful Paint 0.8 s	Time to Interactive 0.8 s
Speed Index 0.8 s	Total Blocking Time 0 ms
Largest Contentful Paint 0.8 s	Cumulative Layout Shift 0.06

View Treemap View Original Trace



Show audits relevant to: All FCP TBT LCP CLS

OPPORTUNITIES

Opportunity	Estimated Savings
Eliminate render-blocking resources	0.52 s

These suggestions can help your page load faster. They don't directly affect the Performance score.

DIAGNOSTICS

- ▲ Image elements do not have explicit `width` and `height`
- ▲ Serve static assets with an efficient cache policy — 12 resources found
- ○ Avoid chaining critical requests — 7 chains found
- ○ Keep request counts low and transfer sizes small — 21 requests • 363 KiB
- ○ Largest Contentful Paint element — 1 element found
- ○ Avoid large layout shifts — 3 elements found

More information about the performance of your application. These numbers don't directly affect the Performance score.

PASSED AUDITS (33) Show

100

Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

ADDITIONAL ITEMS TO MANUALLY CHECK (10) Show

These items address areas which an automated testing tool cannot cover. Learn more in our guide on conducting an accessibility review.

PASSED AUDITS (20) Show

NOT APPLICABLE (24) Show

100

Best Practices

TRUST AND SAFETY

- Ensure CSP is effective against XSS attacks

GENERAL

- Detected JavaScript libraries

PASSED AUDITS (13) Show

NOT APPLICABLE (1) Show

SEO

These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on Core Web Vitals. Learn more.

ADDITIONAL ITEMS TO MANUALLY CHECK (1) Show

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (10) Show

NOT APPLICABLE (4) Show

PWA

These checks validate the aspects of a Progressive Web App. Learn more.

INSTALLABLE

- ▲ Web app manifest or service worker do not meet the installability requirements — 1 reason

PWA OPTIMIZED

- ▲ Does not register a service worker that controls page and `start_url`
- ▲ Is not configured for a custom splash screen Failures: No manifest was fetched.
- ▲ Does not set a theme color for the address bar.
  Failures: No manifest was fetched, No `<meta name="theme-color">` tag found.
- ○ Content is sized correctly for the viewport
- Has a `<meta name="viewport">` tag with `width` or `initial-scale`
- ▲ Does not provide a valid `apple-touch-icon`
- ▲ Manifest doesn't have a maskable icon No manifest was fetched

ADDITIONAL ITEMS TO MANUALLY CHECK (3)

Show

These checks are required by the baseline PWA Checklist but are not automatically checked by Lighthouse. They do not affect your score but it's important that you verify them manually.

- Captured at Sep 7, 2023, 12:07 AM GMT+2
- Initial page load
- Emulated Desktop with Lighthouse 9.6.8
- Custom throttling
- Single page load
- Using Chromium 111.0.0.0 with devtools

Generated by **Lighthouse** 9.6.8 | File an issue

https://maximec31.github.io/Nina-Carducci/

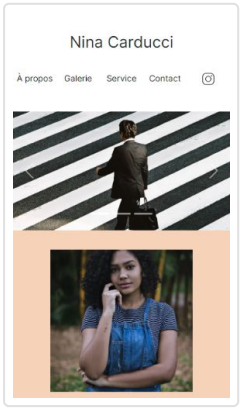
Performance	Accessibility	Best Practices	SEO	PWA
94	100	100	100	—

94

Performance

Values are estimated and may vary. The performance score is calculated directly from these metrics. See calculator.

▲ 0–49 50–89 90–100

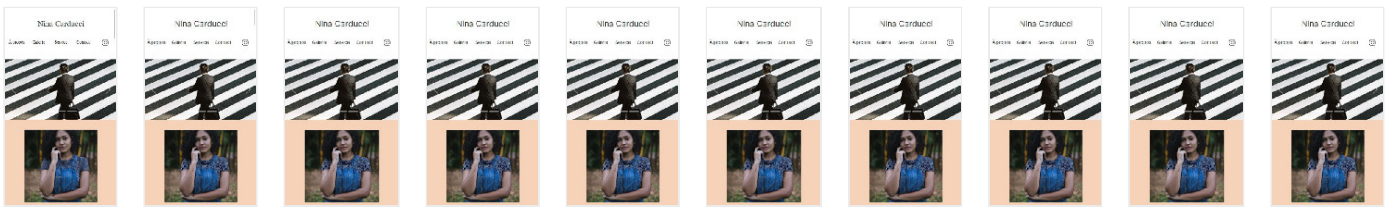


METRICS

Expand view

First Contentful Paint: 1.8 s	Time to Interactive: 1.8 s
Speed Index: 1.8 s	Total Blocking Time: 0 ms
Largest Contentful Paint: 2.8 s	Cumulative Layout Shift: 0.084

View Treemap View Original Trace



Show audits relevant to: All FCP TBT LCP CLS

OPPORTUNITIES

Opportunity	Estimated Savings

▲ Eliminate render-blocking resources 0.91 s

Resources are blocking the first paint of your page. Consider delivering critical JS/CSS inline and deferring all non-critical JS/styles. Learn more. FCP LCP

URL	Transfer Size	Potential Savings
…css/bootstrap.min.css (cdn.jsdelivr.net)	34.0 KiB	1,050 ms

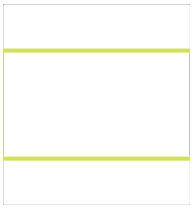
Reduce unused CSS 0.2 s

Reduce unused rules from stylesheets and defer CSS not used for above-the-fold content to decrease bytes consumed by network activity. Learn more. FCP LCP

URL	Transfer Size	Potential Savings
…css/bootstrap.min.css (cdn.jsdelivr.net)	34.0 KiB	32.4 KiB

Properly size images 0.15 s

Serve images that are appropriately-sized to save cellular data and improve load time. Learn more.

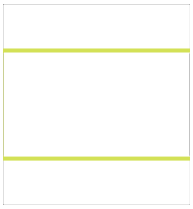
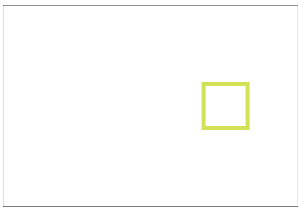
	URL	Resource Size	Potential Savings
 mg.d-block.w-100	i …img-720/photo-professionnel.webp (maximec31.github.io)	62.3 KiB	20.4 KiB

These suggestions can help your page load faster. They don't directly affect the Performance score.

DIAGNOSTICS

▲ Image elements do not have explicit `width` and `height`

Set an explicit width and height on image elements to reduce layout shifts and improve CLS. Learn more CLS

URL	
	img.d-block.w-100 …img-720/photo-professionnel.webp (maximec31.github.io)
	img …images/nina-resized.webp (maximec31.github.io)
	img …images/instagram.png (maximec31.github.io)
	img

Serve static assets with an efficient cache policy — 7 resources found

A long cache lifetime can speed up repeat visits to your page. Learn more.

URL	Cache TTL	Transfer Size
…img-720/photo-professionnel.webp (maximec31.github.io)	10 m	62 KiB
…img-480/concert%20(1).webp (maximec31.github.io)	10 m	35 KiB
…images/nina-resized.webp (maximec31.github.io)	10 m	29 KiB
…assets/maugallery.js (maximec31.github.io)	10 m	2 KiB
…assets/style.css (maximec31.github.io)	10 m	2 KiB
…images/instagram.png (maximec31.github.io)	10 m	1 KiB

URL	Cache TTL	Transfer Size
…assets/scripts.js (maximec31.github.io)	10 m	0 KiB

Avoid chaining critical requests — 7 chains found

The Critical Request Chains below show you what resources are loaded with a high priority. Consider reducing the length of chains, reducing the download size of resources, or deferring the download of unnecessary resources to improve page load. Learn more. FCP LCP

Maximum critical path latency: **190 ms**

Initial Navigation

- /Nina-Carducci/ (maximec31.github.io)
  - …css/bootstrap.min.css (cdn.jsdelivr.net) **- 30 ms, 34.00 KiB**
  - …assets/style.css (maximec31.github.io) **- 30 ms, 1.66 KiB**
  - /css2?family=… (fonts.googleapis.com)
    - …v12/UcCO3FwrK….woff2 (fonts.gstatic.com) **- 30 ms, 16.39 KiB**
    - …v13/rnCu-xNNw….woff2 (fonts.gstatic.com) **- 20 ms, 14.53 KiB**
    - …v13/rnCu-xNNw….woff2 (fonts.gstatic.com) **- 30 ms, 15.33 KiB**
    - …v13/rnCr-xNNw….woff2 (fonts.gstatic.com) **- 30 ms, 13.52 KiB**
  - …js/bootstrap.bundle.min.js (cdn.jsdelivr.net) **- 60 ms, 24.55 KiB**

Keep request counts low and transfer sizes small — 16 requests • 285 KiB

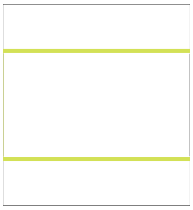
To set budgets for the quantity and size of page resources, add a budget.json file. Learn more.

Resource Type	Requests	Transfer Size
Total	16	285.1 KiB
Image	4	127.4 KiB
Font	4	59.8 KiB
Script	4	57.2 KiB
Stylesheet	3	36.6 KiB
Document	1	4.1 KiB
Media	0	0.0 KiB

Resource Type	Requests	Transfer Size
Other	0	0.0 KiB
Third-party	8	149.2 KiB

Largest Contentful Paint element — 1 element found

This is the largest contentful element painted within the viewport. Learn More LCP

Element
 img.d-block.w-100

Avoid large layout shifts — 5 elements found

These DOM elements contribute most to the CLS of the page. CLS

Element	CLS Contribution
 div#about-me	0.083
 p.name	0
a	0
a	0

Element	CLS Contribution
a	0

Avoid long main-thread tasks — 1 long task found

Lists the longest tasks on the main thread, useful for identifying worst contributors to input delay. Learn more TBT

URL	Start Time	Duration
/Nina-Carducci/ (maximec31.github.io)	803 ms	101 ms

More information about the performance of your application. These numbers don't directly affect the Performance score.

PASSED AUDITS (30) Hide

Defer offscreen images

Consider lazy-loading offscreen and hidden images after all critical resources have finished loading to lower time to interactive. Learn more.

Minify CSS

Minifying CSS files can reduce network payload sizes. Learn more. FCP LCP

Minify JavaScript

Minifying JavaScript files can reduce payload sizes and script parse time. Learn more. FCP LCP

Reduce unused JavaScript

Reduce unused JavaScript and defer loading scripts until they are required to decrease bytes consumed by network activity. Learn more. LCP

Efficiently encode images

Optimized images load faster and consume less cellular data. Learn more.

Serve images in next-gen formats

Image formats like WebP and AVIF often provide better compression than PNG or JPEG, which means faster downloads and less data consumption. Learn more.

Enable text compression

Text-based resources should be served with compression (gzip, deflate or brotli) to minimize total network bytes. Learn more. FCP LCP

Preconnect to required origins

Consider adding `preconnect` or `dns-prefetch` resource hints to establish early connections to important third-party origins. Learn more. FCP LCP

Initial server response time was short — Root document took 20 ms

Keep the server response time for the main document short because all other requests depend on it. Learn more. FCP LCP

URL	Time Spent
/Nina-Carducci/ (maximec31.github.io)	20 ms

Avoid multiple page redirects

Redirects introduce additional delays before the page can be loaded. Learn more. FCP LCP

Preload key requests

Consider using `<link rel=preload>` to prioritize fetching resources that are currently requested later in page load. Learn more. FCP LCP

Use HTTP/2

HTTP/2 offers many benefits over HTTP/1.1, including binary headers and multiplexing. Learn more.

Use video formats for animated content

Large GIFs are inefficient for delivering animated content. Consider using MPEG4/WebM videos for animations and PNG/WebP for static images instead of GIF to save network bytes. Learn more LCP

Remove duplicate modules in JavaScript bundles

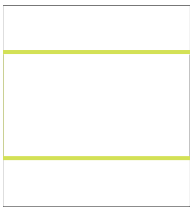
Remove large, duplicate JavaScript modules from bundles to reduce unnecessary bytes consumed by network activity. TBT

Avoid serving legacy JavaScript to modern browsers

Polyfills and transforms enable legacy browsers to use new JavaScript features. However, many aren't necessary for modern browsers. For your bundled JavaScript, adopt a modern script deployment strategy using module/nomodule feature detection to reduce the amount of code shipped to modern browsers, while retaining support for legacy browsers. Learn More TBT

Preload Largest Contentful Paint image

Preload the image used by the LCP element in order to improve your LCP time. Learn more. LCP

	URL	Potential Savings
 img. d-block. w-100	…img-720/photo-professionnel.webp (maximec31.github.io)	0 ms

Avoids enormous network payloads — Total size was 285 KiB

Large network payloads cost users real money and are highly correlated with long load times. Learn more. LCP

☑ Show 3rd-party resources (7)

URL	Transfer Size
…img-720/photo-professionnel.webp (maximec31.github.io)	62.4 KiB
…img-480/concert%20(1).webp (maximec31.github.io)	35.2 KiB
…css/bootstrap.min.css (cdn.jsdelivr.net)	34.0 KiB
/jquery-3.7.1.min.js (code.jquery.com)	30.0 KiB
…images/nina-resized.webp (maximec31.github.io)	28.9 KiB
…js/bootstrap.bundle.min.js (cdn.jsdelivr.net)	24.6 KiB
…v12/UcCO3FwrK….woff2 (fonts.gstatic.com)	16.4 KiB

URL	Transfer Size
…v13/rnCu-xNNw….woff2 (fonts.gstatic.com)	15.3 KiB
…v13/rnCu-xNNw….woff2 (fonts.gstatic.com)	14.5 KiB
…v13/rnCr-xNNw….woff2 (fonts.gstatic.com)	13.5 KiB

Avoids an excessive DOM size — 143 elements

A large DOM will increase memory usage, cause longer style calculations, and produce costly layout reflows. Learn more. TBT

Statistic	Element	Value
Total DOM Elements		143
Maximum DOM Depth	div.mg-prev	10
Maximum Child Elements	body	10

User Timing marks and measures

Consider instrumenting your app with the User Timing API to measure your app's real-world performance during key user experiences. Learn more.

JavaScript execution time — 0.3 s

Consider reducing the time spent parsing, compiling, and executing JS. You may find delivering smaller JS payloads helps with this. Learn more. TBT

Show 3rd-party resources (1)

URL	Total CPU Time	Script Evaluation	Script Parse
/Nina-Carducci/ (maximec31.github.io)	631 ms	159 ms	9 ms
/jquery-3.7.1.min.js (code.jquery.com)	176 ms	92 ms	4 ms

URL	Total CPU Time	Script Evaluation	Script Parse
Unattributable	123 ms	7 ms	0 ms

Minimizes main-thread work — 1.0 s

Consider reducing the time spent parsing, compiling and executing JS. You may find delivering smaller JS payloads helps with this. Learn more TBT

Category	Time Spent
Style & Layout	334 ms
Other	279 ms
Script Evaluation	270 ms
Parse HTML & CSS	44 ms
Rendering	25 ms
Script Parsing & Compilation	17 ms

All text remains visible during webfont loads

Leverage the font-display CSS feature to ensure text is user-visible while webfonts are loading. Learn more. FCP LCP

Minimize third-party usage — Third-party code blocked the main thread for 0 ms

Third-party code can significantly impact load performance. Limit the number of redundant third-party providers and try to load third-party code after your page has primarily finished loading. Learn more. TBT

Third-Party	Transfer Size	Main-Thread Blocking Time
Google Fonts	61 KiB	0 ms
…v12/UcCO3FwrK….woff2 (fonts.gstatic.com)	16 KiB	0 ms
…v13/rnCu-xNNw….woff2 (fonts.gstatic.com)	15 KiB	0 ms
…v13/rnCu-xNNw….woff2 (fonts.gstatic.com)	15 KiB	0 ms
…v13/rnCr-xNNw….woff2 (fonts.gstatic.com)	14 KiB	0 ms
JSDelivr CDN	59 KiB	0 ms

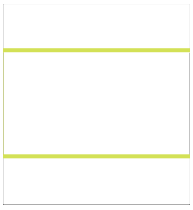
Third-Party	Transfer Size	Main-Thread Blocking Time
…css/bootstrap.min.css (cdn.jsdelivr.net)	34 KiB	0 ms
…js/bootstrap.bundle.min.js (cdn.jsdelivr.net)	25 KiB	0 ms
jQuery CDN	30 KiB	0 ms
/jquery-3.7.1.min.js (code.jquery.com)	30 KiB	0 ms

Lazy load third-party resources with facades

Some third-party embeds can be lazy loaded. Consider replacing them with a facade until they are required. Learn more. TBT

Largest Contentful Paint image was not lazily loaded

Above-the-fold images that are lazily loaded render later in the page lifecycle, which can delay the largest contentful paint. Learn more.

Element	
	img.d-block.w-100

Uses passive listeners to improve scrolling performance

Consider marking your touch and wheel event listeners as `passive` to improve your page's scroll performance. Learn more.

Avoids `document.write()`

For users on slow connections, external scripts dynamically injected via `document.write()` can delay page load by tens of seconds. Learn more.

Avoid non-composited animations

Animations which are not composited can be janky and increase CLS. Learn more CLS

Has a `<meta name="viewport">` tag with `width` or `initial-scale`

A `<meta name="viewport">` not only optimizes your app for mobile screen sizes, but also prevents a 300 millisecond delay to user input. Learn more. TBT

Avoids `unload` event listeners

The `unload` event does not fire reliably and listening for it can prevent browser optimizations like the Back-Forward Cache. Use `pagehide` or `visibilitychange` events instead. Learn more

Accessibility

These checks highlight opportunities to improve the accessibility of your web app. Only a subset of accessibility issues can be automatically detected so manual testing is also encouraged.

ADDITIONAL ITEMS TO MANUALLY CHECK (10) Hide

- The page has a logical tab order

  Tabbing through the page follows the visual layout. Users cannot focus elements that are offscreen. Learn more.

- Interactive controls are keyboard focusable

  Custom interactive controls are keyboard focusable and display a focus indicator. Learn more.

- Interactive elements indicate their purpose and state

  Interactive elements, such as links and buttons, should indicate their state and be distinguishable from non-interactive elements. Learn more.

- The user's focus is directed to new content added to the page

  If new content, such as a dialog, is added to the page, the user's focus is directed to it. Learn more.

- User focus is not accidentally trapped in a region

  A user can tab into and out of any control or region without accidentally trapping their focus. Learn more.

- Custom controls have associated labels

  Custom interactive controls have associated labels, provided by aria-label or aria-labelledby. Learn more.

- Custom controls have ARIA roles

Custom interactive controls have appropriate ARIA roles. Learn more.

Visual order on the page follows DOM order

DOM order matches the visual order, improving navigation for assistive technology. Learn more.

Offscreen content is hidden from assistive technology

Offscreen content is hidden with display: none or aria-hidden=true. Learn more.

HTML5 landmark elements are used to improve navigation

Landmark elements (<main>, <nav>, etc.) are used to improve the keyboard navigation of the page for assistive technology. Learn more.

These items address areas which an automated testing tool cannot cover. Learn more in our guide on conducting an accessibility review.

PASSED AUDITS (20) Hide

`[aria-*]` attributes match their roles

Each ARIA `role` supports a specific subset of `aria-*` attributes. Mismatching these invalidates the `aria-*` attributes. Learn more.

`[aria-hidden="true"]` is not present on the document `<body>`

Assistive technologies, like screen readers, work inconsistently when `aria-hidden="true"` is set on the document `<body>`. Learn more.

`[aria-*]` attributes have valid values

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid values. Learn more.

`[aria-*]` attributes are valid and not misspelled

Assistive technologies, like screen readers, can't interpret ARIA attributes with invalid names. Learn more.

Buttons have an accessible name

When a button doesn't have an accessible name, screen readers announce it as "button", making it unusable for users who rely on screen readers. Learn more.

ARIA IDs are unique

The value of an ARIA ID must be unique to prevent other instances from being overlooked by assistive technologies. Learn more.

Image elements have `[alt]` attributes

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. Learn more.

Form elements have associated labels

Labels ensure that form controls are announced properly by assistive technologies, like screen readers. Learn more.

`[user-scalable="no"]` is not used in the `<meta name="viewport">` element and the `[maximum-scale]` attribute is not less than 5.

Disabling zooming is problematic for users with low vision who rely on screen magnification to properly see the contents of a web page. Learn more.

`[aria-hidden="true"]` elements do not contain focusable descendents

Focusable descendents within an `[aria-hidden="true"]` element prevent those interactive elements from being available to users of assistive technologies like screen readers. Learn more.

The page contains a heading, skip link, or landmark region

Adding ways to bypass repetitive content lets keyboard users navigate the page more efficiently. Learn more.

Background and foreground colors have a sufficient contrast ratio

Low-contrast text is difficult or impossible for many users to read. Learn more.

Document has a `<title>` element

The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. Learn more.

`[id]` attributes on active, focusable elements are unique

All focusable elements must have a unique `id` to ensure that they're visible to assistive technologies. Learn more.

`<html>` element has a `[lang]` attribute

If a page doesn't specify a lang attribute, a screen reader assumes that the page is in the default language that the user chose when setting up the screen reader. If the page isn't actually in the default language, then the screen reader might not announce the page's text correctly. Learn more.

`<html>` element has a valid value for its `[lang]` attribute

Specifying a valid BCP 47 language helps screen readers announce text properly. Learn more.

Links have a discernible name

Link text (and alternate text for images, when used as links) that is discernible, unique, and focusable improves the navigation experience for screen reader users. Learn more.

Lists contain only `<li>` elements and script supporting elements (`<script>` and `<template>`).

Screen readers have a specific way of announcing lists. Ensuring proper list structure aids screen reader output. Learn more.

List items (`<li>`) are contained within `<ul>` or `<ol>` parent elements

Screen readers require list items (`<li>`) to be contained within a parent `<ul>` or `<ol>` to be announced properly. Learn more.

Heading elements appear in a sequentially-descending order

Properly ordered headings that do not skip levels convey the semantic structure of the page, making it easier to navigate and understand when using assistive technologies. Learn more.

NOT APPLICABLE (24) Hide

`[accesskey]` values are unique

Access keys let users quickly focus a part of the page. For proper navigation, each access key must be unique. Learn more.

`button`, `link`, and `menuitem` elements have accessible names

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

ARIA input fields have accessible names

When an input field doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

ARIA `meter` elements have accessible names

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

ARIA `progressbar` elements have accessible names

When a `progressbar` element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

`[role]`s have all required `[aria-*]` attributes

Some ARIA roles have required attributes that describe the state of the element to screen readers. Learn more.

Elements with an ARIA `[role]` that require children to contain a specific `[role]` have all required children.

Some ARIA parent roles must contain specific child roles to perform their intended accessibility functions. Learn more.

`[role]`s are contained by their required parent element

Some ARIA child roles must be contained by specific parent roles to properly perform their intended accessibility functions. Learn more.

`[role]` values are valid

ARIA roles must have valid values in order to perform their intended accessibility functions. Learn more.

ARIA toggle fields have accessible names

When a toggle field doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

ARIA `tooltip` elements have accessible names

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

ARIA `treeitem` elements have accessible names

When an element doesn't have an accessible name, screen readers announce it with a generic name, making it unusable for users who rely on screen readers. Learn more.

`<dl>`'s contain only properly-ordered `<dt>` and `<dd>` groups, `<script>`, `<template>` or `<div>` elements.

When definition lists are not properly marked up, screen readers may produce confusing or inaccurate output. Learn more.

- Definition list items are wrapped in `<dl>` elements

  Definition list items (`<dt>` and `<dd>`) must be wrapped in a parent `<dl>` element to ensure that screen readers can properly announce them. Learn more.

- No form fields have multiple labels

  Form fields with multiple labels can be confusingly announced by assistive technologies like screen readers which use either the first, the last, or all of the labels. Learn more.

- `<frame>` or `<iframe>` elements have a title

  Screen reader users rely on frame titles to describe the contents of frames. Learn more.

- `<input type="image">` elements have `[alt]` text

  When an image is being used as an `<input>` button, providing alternative text can help screen reader users understand the purpose of the button. Learn more.

- The document does not use `<meta http-equiv="refresh">`

  Users do not expect a page to refresh automatically, and doing so will move focus back to the top of the page. This may create a frustrating or confusing experience. Learn more.

- `<object>` elements have alternate text

  Screen readers cannot translate non-text content. Adding alternate text to `<object>` elements helps screen readers convey meaning to users. Learn more.

- No element has a `[tabindex]` value greater than 0

  A value greater than 0 implies an explicit navigation ordering. Although technically valid, this often creates frustrating experiences for users who rely on assistive technologies. Learn more.

- Cells in a `<table>` element that use the `[headers]` attribute refer to table cells within the same table.

  Screen readers have features to make navigating tables easier. Ensuring `<td>` cells using the `[headers]` attribute only refer to other cells in the same table may improve the experience for screen reader users. Learn more.

- `<th>` elements and elements with `[role="columnheader"/"rowheader"]` have data cells they describe.

  Screen readers have features to make navigating tables easier. Ensuring table headers always refer to some set of cells

may improve the experience for screen reader users. Learn more.

- `[lang]` attributes have a valid value

Specifying a valid BCP 47 language on elements helps ensure that text is pronounced correctly by a screen reader. Learn more.

- `<video>` elements contain a `<track>` element with `[kind="captions"]`

When a video provides a caption it is easier for deaf and hearing impaired users to access its information. Learn more.

100

Best Practices

TRUST AND SAFETY

- Ensure CSP is effective against XSS attacks

A strong Content Security Policy (CSP) significantly reduces the risk of cross-site scripting (XSS) attacks. Learn more

Description	Directive	Severity
No CSP found in enforcement mode		High

GENERAL

- Detected JavaScript libraries

All front-end JavaScript libraries detected on the page. Learn more.

Name	Version
Bootstrap	5.3.1
jQuery	3.7.1

PASSED AUDITS (13) Hide

Uses HTTPS

All sites should be protected with HTTPS, even ones that don't handle sensitive data. This includes avoiding mixed content, where some resources are loaded over HTTP despite the initial request being served over HTTPS. HTTPS prevents intruders from tampering with or passively listening in on the communications between your app and your users, and is a prerequisite for HTTP/2 and many new web platform APIs. Learn more.

Avoids requesting the geolocation permission on page load

Users are mistrustful of or confused by sites that request their location without context. Consider tying the request to a user action instead. Learn more.

Avoids requesting the notification permission on page load

Users are mistrustful of or confused by sites that request to send notifications without context. Consider tying the request to user gestures instead. Learn more.

Avoids front-end JavaScript libraries with known security vulnerabilities

Some third-party scripts may contain known security vulnerabilities that are easily identified and exploited by attackers. Learn more.

Allows users to paste into password fields

Preventing password pasting undermines good security policy. Learn more.

Displays images with correct aspect ratio

Image display dimensions should match natural aspect ratio. Learn more.

Serves images with appropriate resolution

Image natural dimensions should be proportional to the display size and the pixel ratio to maximize image clarity. Learn more.

Page has the HTML doctype

Specifying a doctype prevents the browser from switching to quirks-mode. Learn more.

Properly defines charset

A character encoding declaration is required. It can be done with a `<meta>` tag in the first 1024 bytes of the HTML or in the Content-Type HTTP response header. Learn more.

Avoids deprecated APIs

Deprecated APIs will eventually be removed from the browser. Learn more.

No browser errors logged to the console

Errors logged to the console indicate unresolved problems. They can come from network request failures and other browser concerns. Learn more

No issues in the `Issues` panel in Chrome Devtools

Issues logged to the `Issues` panel in Chrome Devtools indicate unresolved problems. They can come from network request failures, insufficient security controls, and other browser concerns. Open up the Issues panel in Chrome DevTools for more details on each issue.

Page has valid source maps

Source maps translate minified code to the original source code. This helps developers debug in production. In addition, Lighthouse is able to provide further insights. Consider deploying source maps to take advantage of these benefits. Learn more.

URL	Map URL
…js/bootstrap.bundle.min.js (cdn.jsdelivr.net)	…js/bootstrap.bundle.min.js.map (cdn.jsdelivr.net)

NOT APPLICABLE (1) Hide

Fonts with `font-display: optional` are preloaded

Preload `optional` fonts so first-time visitors may use them. Learn more

SEO

These checks ensure that your page is following basic search engine optimization advice. There are many additional factors Lighthouse does not score here that may affect your search ranking, including performance on Core Web Vitals. Learn more.

ADDITIONAL ITEMS TO MANUALLY CHECK (1)

Hide

- Structured data is valid

  Run the Structured Data Testing Tool and the Structured Data Linter to validate structured data. Learn more.

Run these additional validators on your site to check additional SEO best practices.

PASSED AUDITS (12)

Hide

- Has a `<meta name="viewport">` tag with `width` or `initial-scale`

  A `<meta name="viewport">` not only optimizes your app for mobile screen sizes, but also prevents a 300 millisecond delay to user input. Learn more. TBT

- Document has a `<title>` element

  The title gives screen reader users an overview of the page, and search engine users rely on it heavily to determine if a page is relevant to their search. Learn more.

- Document has a meta description

  Meta descriptions may be included in search results to concisely summarize page content. Learn more.

- Page has successful HTTP status code

  Pages with unsuccessful HTTP status codes may not be indexed properly. Learn more.

- Links have descriptive text

  Descriptive link text helps search engines understand your content. Learn more.

- Links are crawlable

  Search engines may use `href` attributes on links to crawl websites. Ensure that the `href` attribute of anchor elements links to an appropriate destination, so more pages of the site can be discovered. Learn More

- Page isn't blocked from indexing

  Search engines are unable to include your pages in search results if they don't have permission to crawl them. Learn more.

- Image elements have `[alt]` attributes

Informative elements should aim for short, descriptive alternate text. Decorative elements can be ignored with an empty alt attribute. Learn more.

Document has a valid `hreflang`

hreflang links tell search engines what version of a page they should list in search results for a given language or region. Learn more.

Document uses legible font sizes — 100% legible text

Font sizes less than 12px are too small to be legible and require mobile visitors to “pinch to zoom” in order to read. Strive to have >60% of page text ≥12px. Learn more.

Source	Selector	% of Page Text	Font Size
`Legible text`		100.00%	≥ 12px

Document avoids plugins

Search engines can't index plugin content, and many devices restrict plugins or don't support them. Learn more.

Tap targets are sized appropriately — 100% appropriately sized tap targets

Interactive elements like buttons and links should be large enough (48x48px), and have enough space around them, to be easy enough to tap without overlapping onto other elements. Learn more.

NOT APPLICABLE (2) Hide

robots.txt is valid

If your robots.txt file is malformed, crawlers may not be able to understand how you want your website to be crawled or indexed. Learn more.

Document has a valid `rel=canonical`

Canonical links suggest which URL to show in search results. Learn more.

PWA

These checks validate the aspects of a Progressive Web App. Learn more.

INSTALLABLE

▲ Web app manifest or service worker do not meet the installability requirements — 1 reason

Service worker is the technology that enables your app to use many Progressive Web App features, such as offline, add to homescreen, and push notifications. With proper service worker and manifest implementations, browsers can proactively prompt users to add your app to their homescreen, which can lead to higher engagement. Learn more.

Failure reason

Page has no manifest <link> URL

PWA OPTIMIZED

▲ Does not register a service worker that controls page and `start_url`

The service worker is the technology that enables your app to use many Progressive Web App features, such as offline, add to homescreen, and push notifications. Learn more.

▲ Is not configured for a custom splash screen Failures: No manifest was fetched.

A themed splash screen ensures a high-quality experience when users launch your app from their homescreens. Learn more.

▲ Does not set a theme color for the address bar.
Failures: No manifest was fetched, No `<meta name="theme-color">` tag found.

The browser address bar can be themed to match your site. Learn more.

Content is sized correctly for the viewport

If the width of your app's content doesn't match the width of the viewport, your app might not be optimized for mobile screens. Learn more.

Has a `<meta name="viewport">` tag with `width` or `initial-scale`

A `<meta name="viewport">` not only optimizes your app for mobile screen sizes, but also prevents a 300 millisecond delay to user input. Learn more. TBT

▲ Does not provide a valid `apple-touch-icon`

For ideal appearance on iOS when users add a progressive web app to the home screen, define an `apple-touch-icon`. It must point to a non-transparent 192px (or 180px) square PNG. Learn More.

▲ Manifest doesn't have a maskable icon No manifest was fetched

A maskable icon ensures that the image fills the entire shape without being letterboxed when installing the app on a device. Learn more.

ADDITIONAL ITEMS TO MANUALLY CHECK (3)

Hide

○ Site works cross-browser

To reach the most number of users, sites should work across every major browser. Learn more.

○ Page transitions don't feel like they block on the network

Transitions should feel snappy as you tap around, even on a slow network. This experience is key to a user's perception of performance. Learn more.

○ Each page has a URL

Ensure individual pages are deep linkable via URL and that URLs are unique for the purpose of shareability on social media. Learn more.

These checks are required by the baseline PWA Checklist but are not automatically checked by Lighthouse. They do not affect your score but it's important that you verify them manually.

- Captured at Sep 7, 2023, 12:05 AM GMT+2
- Initial page load
- Emulated Moto G4 with Lighthouse 9.6.8
- Slow 4G throttling
- Single page load
- Using Chromium 111.0.0.0 with devtools

Generated by **Lighthouse** 9.6.8 | File an issue