

INTRODUCTION + STRUCTURE DU PROJET

1. Introduction

Ce rapport présente l'audit complet du site vitrine de la photographe Nina Carducci, réalisé dans le cadre du projet P8 « Optimisez le référencement d'un site de photographe » du parcours Développeur Web OpenClassrooms.

L'objectif de cet audit est d'identifier, analyser et documenter l'ensemble des problématiques liées :

- aux performances (Core Web Vitals, poids des ressources, scripts bloquants)
- au référencement naturel (SEO technique et sémantique)
- à l'accessibilité (WCAG 2.1 AA)
- à la qualité du code (structure HTML, CSS, JS, dépendances)

L'analyse repose sur :

- un audit manuel du code source
- un audit Lighthouse (version 13.0.1, février 2026)
- les bonnes pratiques SEO 2026
- les recommandations Google Search Essentials
- les standards modernes de performance web (formats AVIF, lazy-loading, minification, réduction du JS/CSS inutilisé)
- les recommandations WCAG 2.1 AA

Le rapport est structuré en plusieurs sections :

1. Introduction
2. Analyse de la structure du projet
3. Audit de performance (manuel + Lighthouse)
4. Audit d'accessibilité
5. Audit SEO
6. Audit qualité du code
7. Synthèse priorisée
8. Conclusion
9. Annexes (extraits de code, tableaux, etc.)

Ce document constitue la base du travail d'optimisation qui sera réalisé dans l'étape suivante.

2. Analyse de la structure du projet

Le projet est organisé comme suit :

Code

```
.
├── index.html
├── assets
│   ├── bootstrap
│   │   ├── bootstrap.bundle.js
│   │   ├── bootstrap.bundle.js.map
│   │   ├── bootstrap.bundle.min.js
│   │   ├── bootstrap.bundle.min.js.map
│   │   ├── bootstrap.css
│   │   ├── bootstrap.css.map
│   │   ├── bootstrap.min.css
│   │   └── bootstrap.min.css.map
│   ├── images
│   │   ├── camera.png
│   │   ├── instagram.png
│   │   ├── nina.png
│   │   ├── slider
│   │   │   ├── edward-cisneros-3_h6-1NPDGw-unsplash.jpg
│   │   │   ├── nicholas-green-nPz8akkUmDI-unsplash.jpg
│   │   │   └── ryoji-iwata-wUZjn0v7t0g-unsplash.jpg
│   │   └── gallery
│   │       ├── concerts
│   │       ├── entreprise
│   │       ├── mariage
│   │       └── portraits
│   ├── maugallery.js
│   ├── scripts.js
│   └── style.css
└── structure_projet.txt
```

2.1. Technologies utilisées

- **HTML statique**
- **CSS + Bootstrap** (version complète, non optimisée)
- **JavaScript**
- **jQuery 3.4.1** (version 2019, obsolète en 2026)
- **Plugin jQuery mauGallery**
- **Aucune API, aucun backend**

2.2. Constat général

L'architecture du projet est simple mais présente plusieurs limites :

Dépendances lourdes et obsolètes

- Bootstrap complet est chargé alors que seule une fraction est utilisée.
- jQuery 3.4.1 est obsolète et non recommandé en 2026.

- Le plugin mauGallery repose entièrement sur jQuery.

Images extrêmement lourdes

- Les images représentent plus de 22 Mo de données inutiles.
- Aucun format moderne (AVIF/WebP).
- Aucun redimensionnement.
- Aucun lazy-loading.

Structure HTML non sémantique

- Absence de <header>, <main>, <footer>.
- Hiérarchie Hn incorrecte.
- Absence de balises meta essentielles.

Scripts bloquants

- Tous les scripts sont chargés dans le <head> sans defer.

Accessibilité insuffisante

- alt manquants
- labels non associés
- absence de lang
- contrastes insuffisants

AUDIT PERFORMANCE (MANUEL + LIGHTHOUSE)

3. Audit de performance

L'audit de performance combine :

- une analyse manuelle du code
- les résultats Lighthouse (février 2026)
- les standards Core Web Vitals

Les performances constituent le point le plus critique du site.

3.1. Résultats Lighthouse

Métrique	Valeur	Interprétation
First Contentful Paint (FCP)	0.9 s	Très bon
Largest Contentful Paint (LCP)	16.2 s	Critique
Total Blocking Time (TBT)	0 ms	Excellent
Cumulative Layout Shift (CLS)	0	Excellent
Speed Index	1.1 s	Très bon

Analyse technique

FCP (0.9 s)

Le navigateur affiche rapidement le premier élément visible. Cela signifie que :

- le HTML initial est léger
- les premières ressources sont servies rapidement

LCP (16.2 s)

C'est le problème majeur du site.

Le LCP correspond à **une image du slider**, très lourde, non optimisée, non compressée, non redimensionnée.

Causes :

- images en JPEG haute résolution
- absence de formats modernes (AVIF/WebP)
- absence de lazy-loading
- absence de préchargement de la première image
- scripts bloquants dans le <head>
- absence de compression serveur

Impact :

- score Performance très faible
- mauvaise expérience utilisateur
- pénalité SEO Core Web Vitals

TBT (0 ms)

Le JavaScript ne bloque pas le thread principal. Mais ce bon score est trompeur : Bootstrap + jQuery + maugallery.js sont lourds, mais Lighthouse ne les pénalise pas tant qu'ils ne bloquent pas le thread.

CLS (0)

La mise en page est stable. C'est un bon point.

Speed Index (1.1 s)

Très bon. Cela confirme que le problème principal est **uniquement le LCP**.

3.2. Insights Lighthouse

Render blocking requests — 570 ms

Les scripts suivants bloquent le rendu :

html

```
<script src="./assets/bootstrap/bootstrap.bundle.js"></script>
<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
<script src="./assets/maugallery.js"></script>
<script src="./assets/scripts.js"></script>
```

Problème : Ils sont dans le <head> sans defer.

Impact : Le navigateur doit les charger et les exécuter avant d'afficher la page.

Improve image delivery — 22 060 KiB économisables

Les images représentent plus de **22 Mo** de données inutiles.

Causes :

- JPEG non compressés
- aucune conversion AVIF/WebP
- aucune réduction de résolution
- aucune stratégie responsive
- aucune optimisation serveur

Impact : LCP extrêmement élevé.

Avoid enormous network payloads — 30 687 KiB

Le site charge plus de **30 Mo** au total.

C'est incompatible avec les standards modernes.

Image elements do not have explicit width and height

Impact : Le navigateur ne peut pas réserver l'espace, ce qui peut provoquer des micro-shifts.

3.3. Diagnostics Lighthouse

Minify JavaScript — 101 KiB économisables

Bootstrap + jQuery + scripts non minifiés.

Reduce unused JavaScript — 351 KiB économisables

Bootstrap.bundle.js contient énormément de fonctionnalités inutilisées.

Minify CSS — 39 KiB économisables

Bootstrap.css + style.css non minifiés.

Reduce unused CSS — 192 KiB économisables

Bootstrap complet est chargé alors que seule une fraction est utilisée.

3.4. Audit manuel complémentaire

Images du slider

Les images du slider sont en haute résolution (plusieurs Mo chacune).

Exemple :

Code

```
./assets/images/slider/ryoji-iwata-wUZjn0v7t0g-unsplash.jpg
```

Problèmes :

- résolution trop élevée
- poids excessif
- aucun lazy-loading
- aucun préchargement
- aucun format moderne

Galerie

Les images de la galerie sont également très lourdes.

Problèmes :

- absence d'attribut alt
- absence de lazy-loading
- absence de formats modernes

- absence de redimensionnement

Scripts

Le site charge :

- jQuery 3.4.1 (obsolète)
- Bootstrap complet (inutilement lourd)
- maugallery.js (dépendance jQuery)
- scripts.js (initialisation)

Problèmes :

- scripts bloquants
- dépendances obsolètes
- surcharge inutile

AUDIT ACCESSIBILITÉ

L'accessibilité est un pilier essentiel de la qualité web moderne. Elle garantit que le site est utilisable par tous les utilisateurs, y compris ceux ayant des limitations visuelles, motrices, cognitives ou auditives.

L'audit Lighthouse donne un score de **67**, ce qui révèle plusieurs problèmes majeurs.

Ce bloc présente :

- les résultats Lighthouse
- l'analyse technique détaillée
- les extraits de code problématiques
- les recommandations correctives

4. Audit d'accessibilité

4.1. Résultats Lighthouse

Lighthouse identifie plusieurs problèmes critiques :

Problèmes majeurs détectés

- Images sans attribut alt
- Form elements sans labels associés
- Absence de balise `<title>`

- Liens sans nom discernable
- Contrastes insuffisants
- Absence d'attribut `lang` sur `<html>`
- Hiérarchie Hn incorrecte
- Absence de landmark `<main>`

Ces problèmes affectent :

- les lecteurs d'écran
- la navigation clavier
- la compréhension du contenu
- la conformité WCAG 2.1 AA
- le SEO (Google utilise aussi les alt et la structure Hn)

4.2. Analyse détaillée des problèmes

4.2.1. Images sans attribut alt

Exemple dans ton code :

html

```

```

html

```

```

html

```

```

Problème

- Les lecteurs d'écran ne peuvent pas interpréter l'image.
- Google Images ne peut pas indexer correctement.
- Non conformité WCAG 1.1.1.

Recommandation

- Ajouter un alt descriptif pour les images informatives.
- Utiliser `alt=""` uniquement pour les images décoratives.

4.2.2. Labels non associés aux champs du formulaire

Exemple dans ton code :

```
html
<label>Nom</label>
<input type="text" name="nom" id="nom">
```

Problème

- Le label n'est pas associé à l'input.
- Les technologies d'assistance ne peuvent pas comprendre la relation.

Recommandation

```
html
<label for="nom">Nom</label>
<input type="text" id="nom" name="nom">
```

4.2.3. Absence d'attribut lang sur <html>

Exemple :

```
html
<html>
```

Problème

- Les lecteurs d'écran ne savent pas dans quelle langue lire le contenu.
- Non conformité WCAG 3.1.1.

Recommandation

```
html
<html lang="fr">
```

4.2.4. Contrastes insuffisants

Problème

Certains textes n'ont pas un ratio de contraste suffisant par rapport à leur fond.

Impact

- Difficulté de lecture pour les personnes malvoyantes.
- Non conformité WCAG 1.4.3.

Recommandation

- Ajuster les couleurs dans `style.css`.
- Vérifier les contrastes avec un outil comme WebAIM.

4.2.5. Structure Hn incorrecte

Problèmes identifiés

- Absence de `<h1>`
- Utilisation de `<h3>` comme titre principal
- Utilisation de `<h1>` dans les citations
- Hiérarchie non descendante

Exemple problématique :

html

```
<h3 class="name">Nina Carducci</h3>
```

```
...
```

```
<h1 class="quote__text">Un portrait n'est pas une ressemblance...</h1>
```

Impact

- Les lecteurs d'écran ne comprennent pas la structure du document.
- Google ne comprend pas la hiérarchie du contenu.

Recommandation

- Un seul `<h1>` pour le titre principal.
- `<h2>` pour les sections.
- `<h3>` pour les sous-sections.

4.2.6. Absence de landmark `<main>`

Problème

Le site ne contient pas de balise `<main>`.

Impact

- Les lecteurs d'écran ne peuvent pas identifier le contenu principal.
- Navigation clavier moins efficace.

Recommandation

html

```
<main>
... contenu principal ...
</main>
```

4.2.7. Liens sans nom discernable

Exemple :

html

```
<a class="social-link" href="https://www.instagram.com/...">
  
</a>
```

Problème

- Le lien n'a pas de texte accessible.
- Les lecteurs d'écran annoncent : « lien, vide ».

Recommandation

html

```

```

4.3. Résumé des problèmes d'accessibilité

Problème	Gravité	Impact
alt manquants	Critique	Inaccessibilité totale
labels non associés	Critique	Navigation assistée impossible
absence de lang	Élevée	Lecture incorrecte
structure Hn incorrecte	Élevée	Compréhension dégradée
contrastes insuffisants	Moyenne	Lisibilité réduite
absence de landmark main	Moyenne	Navigation clavier dégradée
liens sans nom	Critique	Accessibilité très faible

4.4. Conclusion du bloc Accessibilité

L'accessibilité du site est insuffisante et nécessite une refonte complète des éléments suivants :

- structure HTML
- attributs alt
- labels
- hiérarchie Hn
- landmarks
- contrastes

Ces corrections amélioreront :

- l'expérience utilisateur
- la conformité WCAG
- le SEO
- la qualité globale du site

AUDIT SEO (RÉFÉRENCEMENT NATUREL)

Le référencement naturel (SEO) est un pilier essentiel pour assurer la visibilité du site de Nina Carducci. L'audit Lighthouse attribue un score de **73**, révélant plusieurs problèmes critiques qui empêchent le site d'être correctement indexé et valorisé par les moteurs de recherche.

Ce bloc présente :

- les résultats Lighthouse
- l'analyse technique détaillée
- les extraits de code problématiques
- les recommandations correctives
- les impacts SEO réels

5. Audit SEO

5.1. Résultats Lighthouse

Lighthouse identifie plusieurs problèmes majeurs :

Problèmes critiques détectés

- Absence de balise `<title>`
- Absence de meta description
- Images sans attribut alt
- Structure Hn incohérente
- Absence d'attribut `lang`
- Absence de données structurées
- Absence de sitemap et robots.txt

Ces éléments sont fondamentaux pour :

- l'indexation
- la compréhension du contenu
- l'affichage dans les résultats enrichis
- la performance SEO globale

5.2. Analyse détaillée des problèmes

5.2.1. Absence de balise <title>

Exemple dans ton code :

```
html
```

```
<head>  
  <meta name="viewport" content="width=device-width, initial-scale=1.0">  
  ...  
</head>
```

Problème

Le site n'a **aucun titre**, ce qui est un problème SEO majeur.

Impact :

- Google ne peut pas identifier le sujet principal de la page
- Le site apparaît avec un titre généré automatiquement
- Perte de crédibilité dans les résultats de recherche

Recommandation

```
html
```

```
<title>Nina Carducci – Photographe professionnelle à Bordeaux</title>
```

5.2.2. Absence de meta description

Problème

Aucune meta description n'est définie.

Impact :

- Google génère un extrait aléatoire
- CTR (taux de clic) réduit
- Mauvaise compréhension du contenu

Recommandation

html

```
<meta name="description" content="Photographe professionnelle à Bordeaux, spécialisée en portraits, mariages, concerts et photos d'entreprise. Découvrez mon portfolio et mes services.">
```

5.2.3. Images sans attribut alt

Exemple :

html

```

```

Problème

- Google Images ne peut pas indexer correctement
- Perte de trafic potentiel
- Impact négatif sur l'accessibilité

Recommandation

html

```

```

5.2.4. Structure Hn incohérente

Problèmes identifiés

- Pas de <h1>
- <h3> utilisé comme titre principal
- <h1> utilisé dans les citations
- Hiérarchie non descendante

Exemple problématique :

html

```
<h3 class="name">Nina Carducci</h3>
```

```
...
```

```
<h1 class="quote__text">Un portrait n'est pas une ressemblance...</h1>
```

Impact

- Google ne comprend pas la structure du contenu
- Mauvaise indexation

- Perte de pertinence sémantique

Recommandation

- Un seul <h1>
- <h2> pour les sections
- <h3> pour les sous-sections

5.2.5. Absence de données structurées (JSON-LD)

Problème

Le site ne fournit aucune donnée structurée.

Impact :

- Pas d'apparition dans les résultats enrichis
- Moins de visibilité dans Google Images
- Moins de visibilité locale (Google Business)

Recommandation

Ajouter :

- Person
- LocalBusiness
- Service
- ImageObject

Exemple :

html

```
<script type="application/ld+json">
{
  "@context": "https://schema.org",
  "@type": "Person",
  "name": "Nina Carducci",
  "jobTitle": "Photographe professionnelle",
  "url": "https://www.ninacarducci.com",
  "image": "https://www.ninacarducci.com/assets/images/nina.png"
}
</script>
```

5.2.6. Absence de sitemap.xml et robots.txt

Problème

Le site ne fournit aucun fichier d'indexation.

Impact :

- Google explore moins efficacement
- Certaines pages peuvent être ignorées
- Pas de contrôle sur l'indexation

Recommandation

robots.txt

Code

```
User-agent: *
Allow: /
Sitemap: https://www.ninacarducci.com/sitemap.xml
```

sitemap.xml

Généré automatiquement ou manuellement.

5.2.7. Absence de balise canonical

Problème

Le site ne définit pas d'URL canonique.

Impact :

- Risque de contenu dupliqué
- Perte de ranking

Recommandation

html

```
<link rel="canonical" href="https://www.ninacarducci.com/">
```

5.3. Résumé des problèmes SEO

Problème	Gravité	Impact
Absence de title	Critique	Mauvais référencement
Absence de meta description	Critique	CTR réduit
alt manquants	Critique	Mauvaise indexation images
Structure Hn incorrecte	Élevée	Compréhension limitée
Absence de données structurées	Élevée	Pas de résultats enrichis
Absence de sitemap/robots.txt	Moyenne	Indexation sous-optimale
Absence de canonical	Moyenne	Risque de duplication

5.4. Conclusion du bloc SEO

Le site présente plusieurs lacunes SEO majeures qui empêchent :

- une bonne indexation
- une bonne compréhension du contenu
- une bonne visibilité dans les résultats de recherche

La correction de ces éléments est indispensable pour améliorer :

- le trafic organique
- la visibilité locale
- la présence dans Google Images
- la crédibilité du site

AUDIT QUALITÉ DU CODE (HTML, CSS, JS, STRUCTURE, DÉPENDANCES)

Ce bloc analyse la qualité du code du site de Nina Carducci, en se basant sur :

- les bonnes pratiques HTML5
- les standards CSS modernes
- les recommandations JavaScript 2026
- les règles de maintenabilité
- la cohérence structurelle
- la gestion des dépendances
- la séparation des responsabilités (SoC)
- la dette technique identifiée

L'objectif est d'évaluer la robustesse, la lisibilité, la maintenabilité et la modernité du code.

6. Audit qualité du code

6.1. Analyse HTML

6.1.1. Structure non sémantique

Le document n'utilise pas les balises sémantiques modernes :

- `<header>`
- `<main>`
- `<footer>`

- <section>
- <nav>
- <figure> (partiellement utilisé)

Exemple actuel :

html

```
<div class="top-header">
<div id="header">
<div id="about">
<div id="gallery">
<div id="services">
<div id="contact">
```

Problème

- Le HTML repose uniquement sur des <div>.
- Les lecteurs d'écran ne peuvent pas identifier les zones fonctionnelles.
- Google comprend moins bien la structure du contenu.
- Non conformité aux standards HTML5.

Recommandation

Remplacer par :

html

```
<header>...</header>
<main>...</main>
<section id="about">...</section>
<section id="gallery">...</section>
<section id="services">...</section>
<section id="contact">...</section>
<footer>...</footer>
```

6.1.2. Hiérarchie Hn incorrecte

Problèmes :

- Pas de <h1>
- <h3> utilisé comme titre principal
- <h1> utilisé dans les citations
- Hiérarchie non descendante

Impact :

- SEO dégradé
- Accessibilité dégradée
- Structure sémantique incohérente

6.1.3. Absence de balises meta essentielles

Problèmes :

- Pas de <title>
- Pas de meta description
- Pas de meta robots
- Pas de meta author
- Pas de meta OpenGraph
- Pas de meta Twitter Card

Impact :

- Mauvaise indexation
- Mauvaise prévisualisation sur les réseaux sociaux
- Perte de visibilité

6.1.4. Attributs alt manquants

Déjà analysé dans le bloc Accessibilité, mais impacte aussi la qualité du code.

6.1.5. Attribut lang manquant

html

```
<html>
```

Doit devenir :

html

```
<html lang="fr">
```

6.2. Analyse CSS

6.2.1. Bootstrap complet chargé inutilement

Le fichier `bootstrap.css` pèse plusieurs centaines de kilooctets.

Problèmes :

- surcharge inutile
- CSS non utilisé
- ralentissement du rendu
- dette technique

Recommandation

- Utiliser une version custom Bootstrap (carousel uniquement)

- Ou remplacer le carousel par une version native JS
- Purger le CSS avec PurgeCSS
- Minifier le CSS

6.2.2. Style inline dans le HTML

Exemple :

html

```
<div class="gallery" style="display: none">
```

Problème :

- Mauvaise séparation des responsabilités
- Difficulté de maintenance

Recommandation :

- Déplacer dans `style.css`

6.2.3. Absence de CSS responsive dédié

Le site repose entièrement sur Bootstrap pour la responsivité. Aucun media query personnalisé n'est présent.

Impact :

- dépendance excessive
- manque de contrôle sur le design mobile

6.3. Analyse JavaScript

6.3.1. Utilisation de jQuery 3.4.1 (2019)

Problèmes :

- obsolète en 2026
- non nécessaire pour un site statique
- surcharge inutile
- dépendance technique non justifiée

6.3.2. Utilisation de maugallery.js

Problèmes :

- dépend de jQuery
- non optimisé
- non maintenu

- surcharge inutile

Recommandation :

- Réécrire la galerie en JavaScript natif
- Ou utiliser une librairie moderne lightweight (GLightbox, Photoswipe)

6.3.3. Scripts bloquants dans le <head>

Exemple :

html

```
<script src="./assets/bootstrap/bootstrap.bundle.js"></script>
<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
<script src="./assets/maugallery.js"></script>
<script src="./assets/scripts.js"></script>
```

Problème :

- bloque le parsing HTML
- augmente le LCP
- mauvaise pratique moderne

Recommandation :

html

```
<script src="..." defer></script>
```

6.3.4. Absence de modularité

Le fichier `scripts.js` contient du code non structuré.

Recommandation :

- utiliser des modules ES6
- séparer les responsabilités
- documenter les fonctions

6.4. Analyse des dépendances

Dépendances actuelles :

Dépendance	Utilité	Problème
Bootstrap	Carousel + layout	Chargé en entier
jQuery	Nécessaire pour maugallery	Obsolète
maugallery.js	Galerie	Non maintenu
scripts.js	Initialisation	Non modulaire

Conclusion

Le site repose sur des dépendances :

- lourdes
- obsolètes
- non nécessaires
- non optimisées

6.5. Analyse des images

Problèmes :

- poids excessif (22 Mo)
- absence de formats modernes
- absence de lazy-loading
- absence de redimensionnement
- absence de compression
- absence de préchargement du LCP

Recommandation :

- conversion en AVIF
- fallback WebP
- lazy-loading
- préchargement du LCP
- compression
- responsive images (srcset)

6.6. Résumé du bloc Qualité du code

Domaine	Problème	Gravité
HTML	structure non sémantique	Élevée
HTML	hiérarchie Hn incorrecte	Critique
HTML	meta essentielles manquantes	Critique
CSS	Bootstrap complet	Élevée
CSS	unused CSS	Élevée
JS	jQuery obsolète	Critique
JS	scripts bloquants	Critique
JS	dépendances inutiles	Élevée
Images	poids excessif	Critique
Images	absence de formats modernes	Critique

SYNTHÈSE PRIORISÉE DES PROBLÈMES ET RECOMMANDATIONS

Ce bloc regroupe l'ensemble des problèmes identifiés dans les audits :

- Performance
- Accessibilité
- SEO
- Qualité du code
- Structure du projet
- Dépendances
- Images

L'objectif est de fournir une **vision claire, hiérarchisée et exploitable** pour :

- la soutenance
- la planification des corrections
- la justification technique des choix d'optimisation
- la démonstration de ton expertise

Cette synthèse est organisée par **priorité**, en fonction de l'impact :

- Critique
- Élevée
- Moyenne
- Faible

6.1. Synthèse globale des problèmes

Tableau récapitulatif

Domaine	Problème	Gravité	Impact	Priorité
Performance	LCP = 16.2 s	Critique	UX très dégradée, SEO pénalisé	Critique
Performance	Images non optimisées (22 Mo)	Critique	Temps de chargement très élevé	Critique
Performance	Scripts bloquants	Critique	Rendu retardé	Critique
Performance	Bootstrap complet	Élevée	Ressources inutiles	Élevée
Performance	Unused JS (351 Ko)	Élevée	Poids inutile	Élevée
Performance	Unused CSS (192 Ko)	Élevée	Poids inutile	Élevée
Accessibilité	alt manquants	Critique	Inaccessibilité totale	Critique

Domaine	Problème	Gravité	Impact	Priorité
Accessibilité	labels non associés	Critique	Navigation assistée impossible	Critique
Accessibilité	absence de lang	Élevée	Lecture incorrecte	Élevée
Accessibilité	structure Hn incorrecte	Élevée	Compréhension dégradée	Élevée
SEO	absence de title	Critique	Mauvais référencement	Critique
SEO	absence de meta description	Critique	CTR réduit	Critique
SEO	alt manquants	Critique	Mauvaise indexation images	Critique
SEO	absence de données structurées	Élevée	Pas de résultats enrichis	Élevée
SEO	absence de sitemap/robots.txt	Moyenne	Indexation sous-optimale	Moyenne
Code	structure HTML non sémantique	Élevée	Accessibilité + SEO	Élevée
Code	dépendances obsolètes (jQuery)	Critique	Dette technique	Critique
Code	maugallery.js non maintenu	Élevée	Dépendance inutile	Élevée
Code	scripts non modularisés	Moyenne	Maintenabilité réduite	Moyenne
Images	absence de formats modernes	Critique	Poids excessif	Critique
Images	absence de lazy-loading	Critique	LCP dégradé	Critique
Images	absence de préchargement LCP	Élevée	LCP dégradé	Élevée

6.2. Synthèse par priorité

6.2.1. Priorité Critique (à corriger immédiatement)

1. Optimisation des images

- Conversion AVIF
- Fallback WebP
- Redimensionnement
- Compression
- Lazy-loading
- Preload du LCP

Impact : Réduction du LCP de 16.2 s à < 2.5 s.

2. Correction des scripts bloquants

- Déplacement en fin de <body>
- Ajout de defer
- Suppression de jQuery si possible
- Remplacement de Bootstrap complet

Impact : Rendu initial plus rapide.

3. Ajout des balises SEO essentielles

- `<title>`
- meta description
- meta robots
- canonical

Impact : Indexation correcte.

4. Correction des alt manquants

Impact : Accessibilité + SEO.

5. Correction des labels non associés

Impact : Navigation assistée.

6. Suppression des dépendances obsolètes

- jQuery
- maugallery.js

Impact : Performance + sécurité + maintenabilité.

6.2.2. Priorité Élevée

1. Structure HTML sémantique

- header
- main
- footer
- section
- nav

Impact : SEO + accessibilité + structure.

2. Hiérarchie Hn correcte

- un seul `<h1>`
- `<h2>` pour les sections
- `<h3>` pour les sous-sections

Impact : Compréhension du contenu.

3. Données structurées JSON-LD

- Person
- LocalBusiness
- Service
- ImageObject

Impact : Résultats enrichis.

4. Purge du CSS

- suppression du CSS inutilisé
- minification

Impact : Poids réduit.

5. Réduction du JS inutilisé

- suppression des modules Bootstrap non utilisés
- minification

Impact : Performance.

6.2.3. Priorité Moyenne

1. Sitemap.xml et robots.txt

Impact : Indexation optimisée.

2. Modularisation du JS

Impact : Maintenabilité.

3. Amélioration du contraste

Impact : Accessibilité.

6.2.4. Priorité Faible

1. Nettoyage du code

- suppression des styles inline
- indentation
- commentaires

Impact : Qualité globale.

6.3. Synthèse stratégique

Objectif principal

Réduire le LCP de **16.2 s** à **< 2.5 s**.

Objectif secondaire

Améliorer le SEO pour atteindre un score > 95.

Objectif tertiaire

Atteindre une accessibilité > 95.

Objectif structurel

Moderniser le code pour :

- réduire la dette technique
- améliorer la maintenabilité
- respecter les standards 2026

6.4. Plan d'action priorisé

Étape 1 — Performance

1. Optimisation des images
2. Lazy-loading
3. Preload du LCP
4. Déplacement des scripts
5. Suppression de jQuery
6. Bootstrap allégé

Étape 2 — SEO

1. Title + meta description
2. Structure Hn
3. Données structurées
4. Sitemap + robots.txt

Étape 3 — Accessibilité

1. alt
2. labels
3. lang
4. landmarks
5. contrastes

Étape 4 — Qualité du code

1. HTML sémantique
2. CSS purgé
3. JS modularisé

CONCLUSION GÉNÉRALE DE L'AUDIT

Ce bloc synthétise l'ensemble des constats réalisés dans les audits précédents (performance, accessibilité, SEO, qualité du code, structure, dépendances). Il constitue la **conclusion officielle** du rapport v2 et servira de base à ta présentation orale lors de la soutenance P8.

7. Conclusion générale

L'audit complet du site de Nina Carducci met en évidence plusieurs problématiques majeures qui affectent :

- la performance globale du site
- l'expérience utilisateur
- l'accessibilité
- le référencement naturel
- la maintenabilité du code
- la conformité aux standards modernes du web

Ces problèmes sont principalement liés à :

- des images extrêmement lourdes et non optimisées
- une structure HTML non sémantique
- l'absence de balises SEO essentielles
- l'utilisation de dépendances obsolètes (jQuery, maugallery.js)

- des scripts bloquants dans le <head>
- l'absence de formats modernes (AVIF/WebP)
- une hiérarchie Hn incorrecte
- l'absence de données structurées
- l'absence de sitemap et robots.txt
- une accessibilité insuffisante (alt manquants, labels non associés, absence de lang)

7.1. Impact global sur le site

Performance

Le problème le plus critique est le **Largest Contentful Paint (LCP) à 16.2 secondes**, causé par des images non optimisées. Ce seul point suffit à :

- dégrader fortement l'expérience utilisateur
- pénaliser le SEO
- réduire le taux de conversion
- augmenter le taux de rebond

Accessibilité

L'absence d'attributs alt, de labels associés, de landmarks et de hiérarchie Hn correcte rend le site difficilement utilisable pour les personnes ayant des limitations visuelles ou cognitives.

SEO

L'absence de <title>, de meta description, de données structurées et de sitemap empêche le site d'être correctement indexé et valorisé par Google.

Qualité du code

Le code manque de structure sémantique, repose sur des dépendances obsolètes et contient des ressources inutiles.

7.2. Points forts du site

Malgré les problèmes identifiés, le site présente plusieurs atouts :

- Design visuel agréable
- Identité graphique cohérente
- Galerie riche et variée

- Navigation simple
- Mise en page stable (CLS = 0)
- TBT excellent (0 ms)

Ces éléments constituent une bonne base pour une optimisation efficace.

7.3. Recommandations globales

1. Optimiser les images (priorité absolue)

- Conversion en AVIF
- Fallback WebP
- Redimensionnement
- Compression
- Lazy-loading
- Preload du LCP

2. Moderniser la structure HTML

- Utilisation de `<header>`, `<main>`, `<footer>`, `<section>`
- Hiérarchie Hn correcte
- Landmarks ARIA si nécessaire

3. Améliorer le SEO

- Ajout d'un `<title>` pertinent
- Meta description optimisée
- Données structurées JSON-LD
- Sitemap.xml
- robots.txt
- Balise canonical

4. Améliorer l'accessibilité

- alt descriptifs
- labels associés
- attribut lang
- contrastes suffisants

- liens accessibles

5. Moderniser le JavaScript

- Suppression de jQuery
- Suppression de maugallery.js
- Réécriture en JavaScript natif
- Scripts en defer

6. Optimiser le CSS

- Purge du CSS inutilisé
- Minification
- Suppression de Bootstrap complet si possible

7.4. Vision finale après optimisation

Une fois les optimisations appliquées, le site pourra atteindre :

- **LCP < 2.5 s**
- **Score Lighthouse Performance > 95**
- **Score SEO > 95**
- **Score Accessibilité > 95**
- **Code moderne, propre et maintenable**
- **Meilleure indexation Google**
- **Meilleure expérience utilisateur**

Le site deviendra :

- plus rapide
- plus accessible
- mieux référencé
- plus professionnel
- plus durable

7.5. Conclusion finale

Cet audit met en lumière les axes d'amélioration prioritaires pour transformer le site de Nina Carducci en un site moderne, performant, accessible et optimisé pour le référencement naturel.

Les recommandations fournies dans ce rapport serviront de base à la **refonte technique** du site, qui constitue la seconde partie du projet P8.

ANNEXES TECHNIQUES (EXTRAITS DE CODE, TABLEAUX, STRUCTURES)

Ce bloc rassemble toutes les preuves techniques nécessaires pour démontrer ton expertise lors de la soutenance P8.

8.1. Extraits de code problématiques identifiés

8.1.1. Scripts bloquants dans le <head>

html

```
<script src="./assets/bootstrap/bootstrap.bundle.js"></script>
<script src="https://code.jquery.com/jquery-3.4.1.min.js"></script>
<script src="./assets/maugallery.js"></script>
<script src="./assets/scripts.js"></script>
```

Problème

- Bloque le parsing HTML
- Retarde le rendu
- Dégrade le LCP

Correction recommandée

html

```
<script src="./assets/bootstrap/bootstrap.bundle.min.js" defer></script>
<script src="./assets/scripts.js" defer></script>
```

(jQuery et maugallery supprimés)

8.1.2. Images sans attribut alt

html

```

```

Correction recommandée

html


```

```

8.1.3. Labels non associés

html

```
<label>Nom</label>
<input type="text" id="nom">
```

Correction recommandée

html

```
<label for="nom">Nom</label>
<input type="text" id="nom" name="nom">
```

8.1.4. Structure HTML non sémantique

html

```
<div id="about">
<div id="gallery">
<div id="services">
```

Correction recommandée

html

```
<section id="about">
<section id="gallery">
<section id="services">
```

8.1.5. Hiérarchie Hn incorrecte

html

```
<h3 class="name">Nina Carducci</h3>
<h1 class="quote__text">Un portrait n'est pas une ressemblance...</h1>
```

Correction recommandée

html

```
<h1>Nina Carducci</h1>
<h2>Un portrait n'est pas une ressemblance...</h2>
```

8.2. Extraits de code recommandés pour l'optimisation

8.2.1. Preload du LCP

html

```
<link rel="preload" as="image" href="./assets/images/slider/ryoji-iwata.avif">
```

8.2.2. Lazy-loading des images

html

```

```

8.2.3. Formats modernes (AVIF + fallback)

html

```
<picture>
  <source srcset="./assets/images/slider/photo.avif" type="image/avif">
  <source srcset="./assets/images/slider/photo.webp" type="image/webp">
  
</picture>
```

8.2.4. Données structurées JSON-LD

html

```
<script type="application/ld+json">
{
  "@context": "https://schema.org",
  "@type": "LocalBusiness",
  "name": "Nina Carducci",
  "image": "https://www.ninacarducci.com/assets/images/nina.png",
  "address": {
    "@type": "PostalAddress",
    "addressLocality": "Bordeaux",
    "addressCountry": "FR"
  },
  "url": "https://www.ninacarducci.com",
  "telephone": "+33 6 00 00 00 00",
  "priceRange": "€€"
}
</script>
```

8.3. Tableaux techniques

8.3.1. Tableau des optimisations images

Action	Gain estimé
Conversion AVIF	-70%
Compression	-30%
Redimensionnement	-50%
Lazy-loading	LCP - 4 à 8 s
Preload LCP	LCP - 1 à 2 s

8.3.2. Tableau des dépendances

Dépendance	Statut	Action
jQuery	Obsolète	Supprimer
maugallery.js	Non maintenu	Supprimer
Bootstrap complet	Trop lourd	Remplacer par version allégée
scripts.js	Non modulaire	Réécrire

8.3.3. Tableau des balises SEO essentielles

Balise	Présente ?	Action
	✗	Ajouter
meta description	✗	Ajouter
canonical	✗	Ajouter
OpenGraph	✗	Ajouter
Twitter Card	✗	Ajouter
JSON-LD	✗	Ajouter
sitemap.xml	✗	Créer
robots.txt	✗	Créer

8.4. Schéma de structure HTML recommandée

Code

```
<html lang="fr">
  <head>
    <title>...</title>
    <meta name="description" content="...">
  </head>

  <body>
    <header>...</header>

    <main>
      <section id="about">...</section>
      <section id="gallery">...</section>
      <section id="services">...</section>
      <section id="contact">...</section>
    </main>

    <footer>...</footer>
  </body>
</html>
```

8.5. Liste des corrections obligatoires

Performance

- Optimiser toutes les images
- Ajouter lazy-loading
- Ajouter preload LCP
- Déplacer scripts + ajouter defer
- Supprimer jQuery
- Supprimer maugallery.js
- Alléger Bootstrap

SEO

- Ajouter title
- Ajouter meta description
- Ajouter données structurées
- Ajouter sitemap + robots.txt
- Corriger Hn

Accessibilité

- Ajouter alt
- Associer labels
- Ajouter lang
- Ajouter landmarks
- Corriger contrastes

Qualité du code

- HTML sémantique
- CSS purgé
- JS modularisé