

Audit & Refonte Technique "Web Performance"

Modalité : Par groupe de 2 étudiants.

Contexte : Vous récupérez un site web existant ("La Panthère") qui est visuellement fonctionnel mais techniquement désastreux (Lent, inaccessible, invisible pour Google).

Mission : Vous agissez en tant qu'experts techniques. Vous devez auditer le site, lister ses failles, puis intervenir sur le code pour le rendre performant, accessible et "SEO-friendly".

Le repository : <https://github.com/Loise/IUT-panthere>

Le site : <https://loise.github.io/IUT-panthere/>

1. Phase 1 : L'Audit Technique (Analyse)

Avant de toucher à une seule ligne de code, vous devez diagnostiquer l'existant.

Utilisez les outils vus en cours (Google Lighthouse, WAVE, W3C Validator, PageSpeed Insights) pour analyser le site fourni.

Livrable attendu : "Rapport d'Audit"

Ce document doit lister **20 problèmes techniques** classés par catégorie :

- **SEO** : (Ex: Balises manquantes, structure Hn incohérente...)
- **Accessibilité** : (Ex: Contrastes, navigation clavier, attributs ARIA/Alt...)
- **Performance** : (Ex: Poids des images, scripts bloquants...)

Pour chaque problème, vous devez indiquer :

1. Le constat (Capture d'écran ou explication).
2. L'impact (Pourquoi est-ce grave ?).
3. La solution technique préconisée.

Utilisez le fichier xls fourni à remplir

2. Phase 2 : La Refonte (Développement)

Vous devez "Forker" le dépôt GitHub du projet initial et appliquer vos corrections

Objectifs Techniques :

1. **Nettoyage HTML** : Passez d'un code "div soup" à un HTML5 sémantique (header, main, footer, nav, section). Validez le code au W3C.

2. **Optimisation des Assets :**
 - Convertissez toutes les images en **WebP**.
 - Redimensionnez les images aux tailles d'affichage réelles.
 - Mettez en place le loading="lazy".
3. **SEO Technique :**
 - Rédigez des <title> et meta description uniques et pertinents pour chaque page.
 - Ajoutez des **Données Structurées (JSON-LD)** pour le balisage "LocalBusiness" ou "Organization".
4. **Accessibilité :**
 - Assurez un contraste suffisant (textes/fonds).
 - Rendez le site navigable au clavier.
 - Ajoutez des alternatives textuelles (alt) pertinentes.

3. Phase 3 : Le Rapport de Preuve (Comparatif)

Une fois le site corrigé et déployé (via GitHub Pages), vous devez prouver l'efficacité de votre travail.

Livrable attendu : "Rapport de Performance" (PDF)

Ce document simple doit montrer le "Avant / Après" pour chaque page principale :

- Capture d'écran du score Lighthouse **avant / après**.
- Capture d'écran du score Wave **avant / après**
- Bilan rapide : Quelles corrections ont apporté le plus de gain ?

4. Modalités de Rendu

Sur Moodle, un seul membre du binôme dépose un fichier ZIP contenant :

1. **Le Rapport d'Audit**
2. **Le Rapport de Performance Avant/Après**
3. Un fichier texte avec :
 - Les noms du binôme.
 - Le lien du **Repository GitHub** (Code source).
 - Le lien du **Site en ligne** (GitHub Pages).

5. Grille d'évaluation Étudiant (Global)

Pôle	Attendus Principaux	Points
Qualité de l'Audit	Capacité à identifier les erreurs critiques (SEO, Accessibilité, Perf) et à proposer des solutions	5 pts

	viables.	
Qualité du Code	HTML Sémantique, Git propre (Commits), Structure du projet, Absence de code mort.	4 pts
Corrections SEO & Accessibilité	Balisage Hn logique, Meta tags, JSON-LD, Contrastes, Navigation clavier, Attributs Alt.	6 pts
Performance	Score Lighthouse > 90/100 (Vert). Images WebP et compressées. Chargement instantané. Score Wave	5 pts