Pour améliorer le score de performance de votre site web selon le rapport Lighthouse, voici plusieurs optimisations que vous pouvez entreprendre :

**1. Réduire le JavaScript inutilisé**

* **Minimisation et report** : Identifiez et éliminez le code non utilisé dans vos lots de JavaScript, en ciblant spécifiquement les gros fichiers tels que react-icons et react-dom.development.js qui occupent un espace significatif sans contribuer au chargement initial. Utilisez des techniques de découpage de code pour charger les scripts uniquement lorsqu'ils sont nécessaires.
* **Chargement paresseux des modules** : Pour les modules react-icons qui ne sont pas immédiatement nécessaires pour le rendu initial, implémentez un chargement paresseux. Cela retarde leur chargement jusqu'à ce qu'ils soient effectivement requis, réduisant le temps de chargement initial et potentiellement améliorant des métriques telles que le Time to First Byte (TTFB).

**2. Minifier le JavaScript et le CSS**

* **Compression** : Appliquez la minification à la fois aux fichiers JavaScript et CSS. Cela réduit leur taille en supprimant les caractères inutiles (comme les espaces et les commentaires) sans affecter leur fonctionnalité.
* **Chargement efficace** : Rationalisez le processus de chargement en combinant les fichiers lorsque c'est possible pour réduire les requêtes HTTP, et utilisez des méthodes de compression telles que GZIP.

**3. Éliminer les ressources bloquant le rendu**

* **Incorporation du CSS/JS critique** : Intégrez de petites quantités de CSS et JavaScript critiques pour le rendu initial afin d'éviter qu'ils bloquent le rendu.
* **Chargement asynchrone** : Utilisez les attributs async ou defer pour les fichiers JavaScript non critiques afin d'éviter qu'ils ne bloquent la construction du DOM.

**4. Optimiser le chargement des polices**

* **Affichage des polices** : Utilisez font-display: swap; dans votre CSS pour assurer que le texte reste visible pendant le chargement des polices web, ce qui prévient le FOIT (Flash of Invisible Text).
* **Formats de polices efficaces** : Servez les polices dans des formats efficaces et chargez uniquement les ensembles de caractères utilisés sur le site pour réduire la taille des fichiers de polices.

**5. Améliorer le Largest Contentful Paint (LCP)**

* **Optimisation des images critiques** : Assurez-vous que les images contribuant au LCP sont de taille optimale, compressées et livrées dans des formats modernes. Le préchargement des images clés peut également accélérer leur rendu.
* **Rationalisation du chemin critique** : Analysez le chemin de rendu critique et réduisez sa complexité en minimisant le nombre de ressources qui doivent être analysées pour afficher la vue initiale.

**6. Réduire les décalages de mise en page**

* **Stabilisation de la mise en page** : Spécifiez les attributs de taille pour tous les médias (images, vidéos, etc.) et assurez-vous que les éléments de publicité ont un espace réservé pour éviter les décalages de mise en page.

**7. Mise en cache efficace des actifs**

* **Politique de cache** : Implémentez des stratégies de mise en cache HTTP longue durée pour les actifs statiques qui ne changent pas souvent, ce qui réduit les temps de chargement lors des visites ultérieures.

**8. Éviter les charges utiles réseau importantes**

* **Examen des tailles d'actifs** : Auditez régulièrement les tailles de vos actifs pour vous assurer qu'ils ne sont pas excessivement grands. Cela réduit les coûts de données pour les utilisateurs et améliore les temps de chargement.