lighthouse-ci-budgets

lighthouse-ci-budgets

> *Catalogue généré le 2026-05-11*

En une phrase

Configure Lighthouse CI dans ton pipeline pour qu'il bloque automatiquement les régressions performance, accessibilité ou SEO via des assertions et budgets définis dans un fichier de config.

Quand l'utiliser

- Sur un projet avec CI/CD (GitHub Actions, GitLab CI) et plusieurs contributeurs
- Quand tu veux te protéger contre les régressions de perf après un merge
- Pour fixer des seuils minimum acceptables (score perf > 80, LCP < 2.5s, etc.)
- Sur un projet client qui a un engagement de qualité contractuel
- Pour automatiser ce qui se fait sinon à la main avant chaque déploiement

Comment l'invoquer

- **Slash command** : `/lighthouse-ci-budgets`
- **Auto-invocation** : Oui — quand un projet a CI/CD ou plusieurs contributeurs

Description détaillée

Lighthouse CI est la version automatisable de Lighthouse (l'outil intégré dans Chrome DevTools). Au lieu de cliquer dans le navigateur après chaque déploiement, tu le branches sur ta pipeline et il tourne tout seul, comparant les résultats à des seuils que tu as définis. Si une régression dépasse le seuil, la pipeline échoue et le merge est bloqué.

Ce skill produit quatre choses concrètes : un fichier de config `lighthouserc.json` (ou `.lighthouserc.js` si tu préfères du JS) avec les assertions et budgets paramétrés, les scripts npm pour lancer Lighthouse CI en local et en CI (`lhci autorun`, `lhci collect`, `lhci assert`), un exemple GitHub Action prêt à coller dans `.github/workflows/lighthouse.yml`, et un set de **budgets initiaux réalistes** (taille JS, CSS, images, fonts) plus les seuils minimum (score perf, a11y, SEO, best practices).

Deux conseils d'usage importants : démarrer avec des seuils réalistes basés sur l'existant — si tu mets `perf > 95` alors que le site est à 72, tu vas être en CI rouge pour toujours — puis resserrer progressivement. Et activer le **multirun** (3 ou 5 exécutions par audit) pour réduire la variance naturelle de Lighthouse qui peut donner +/- 5 points entre deux runs.

Pour Thymon : utile sur les projets qui partent en production sérieuse et où tu veux dormir tranquille, moins pertinent sur un side project où tu fais déjà attention à la main.

Source

- **Plugin** : `perso (standalone)`
- **Nom interne** : `lighthouse-ci-budgets`
- **Fichier** : `/home/thymon/.claude/skills/lighthouse-ci-budgets/SKILL.md`

Revision #2
Created 2026-05-11 21:18:32 UTC by thymon
Updated 2026-05-11 21:36:17 UTC by thymon