Engineer

axe

le moteur d'accessibilité utilisé pour auditer ce site

Ce que c’est

axe, publié comme bibliothèque JavaScript à code source ouvert axe-core, est un ensemble de règles automatiques d’accessibilité exécutées dans une page Web. Il inspecte ce que le navigateur a rendu et signale chaque élément qui enfreint une règle. Deque Systems, entreprise spécialisée en accessibilité, le maintient sous Mozilla Public License 2.0. D’autres outils le réutilisent : la catégorie accessibilité de Lighthouse dans Chrome pondère ses audits selon les niveaux d’impact d’axe, et le guide de tests d’accessibilité de Playwright exécute axe par un petit paquet complémentaire.

Le moteur est un script unique. Un lanceur de tests ou une extension de navigateur le charge dans la page examinée et appelle `axe.run`. Le moteur parcourt le document, cadres imbriqués et DOM fantôme (shadow DOM) compris, en appliquant chaque règle dont les conditions sont remplies. Comme il lit la page après sa mise en page par le navigateur, il peut vérifier des éléments invisibles dans le seul code source : visibilité du texte, couleur de fond, caractère caché d’un élément. C’est aussi pourquoi certaines règles nécessitent un véritable navigateur. Le projet indique que le contraste des couleurs ne fonctionne pas dans JSDOM, le document simulé de nombreux tests unitaires.

Chaque règle porte des étiquettes. L’une précise sa version et son niveau WCAG, par exemple `wcag2a`, `wcag2aa`, `wcag21aa` ou `wcag22aa`, ou indique `best-practice` lorsqu’aucun critère de réussite ne l’exige. D’autres relient les règles à EN 301 549, Section 508 ou aux règles de test de conformité du W3C. Les étiquettes sont donc le moyen de décider ce qu’une exécution vérifie.

Le résultat n’est pas une réussite ou un échec unique, mais quatre listes :

- `violations` : les éléments ayant échoué à une règle, chacun avec un impact mineur, modéré, sérieux ou critique, un bref résumé de correction et un lien vers une explication plus longue.
- `passes` : les éléments ayant réussi, ce qui permet de constater qu'une règle a réellement été exécutée.
- `incomplete` , également appelé « needs review » (à examiner) dans la documentation : les éléments que le moteur n'a pu déclarer conformes ou non avec certitude et qu'une personne doit examiner.
- `inapplicable` : les règles sans rien à tester, comme celles des vidéos sur une page qui n'en contient pas.

Utilisez le concept pour prendre une décision

Utilisez « Qu'est-ce qu'axe ? » pour éclairer une décision réelle. Nous pouvons vous aider à relier ce concept au système que vous construisez ou achetez. Des interfaces rapides, utilisables sur téléphone et cohérentes dans tous les navigateurs de vos clients.

Découvrir nos prestations Contactez-nous →

Vérifiez les hypothèses derrière le terme

Appuyez-vous sur l'explication et les réalisations liées pour formuler les questions de votre projet. Nous pouvons examiner les hypothèses avec vous avant le choix d'un outil. Développement cartographique GIS full-stack — cartographie full-stack, de la base au navigateur

Développement cartographique GIS full-stack Contactez-nous →

Pourquoi c'est important

axe aide une équipe à décider quelles parties de l'accessibilité confier à une machine et si un échec doit empêcher une publication. Deux propriétés le rendent adapté. Il coûte peu à exécuter et peut donc vérifier chaque compilation plutôt qu'un audit annuel. Ses auteurs visent aussi explicitement zéro faux positif, hors bogues : tout cas incertain va dans la liste de revue plutôt que dans les violations. Un contrôle qui crie au loup finit désactivé ; celui qui ne rapporte que ce qu'il peut prouver peut faire échouer une compilation sans susciter de ressentiment.

Sa limite est la couverture. Deque indique qu'axe-core détecte automatiquement, en moyenne, 57 % des problèmes WCAG et renvoie les résultats incertains pour examen manuel. Ce chiffre décrit les problèmes détectés, pas la part des critères de succès qu'une machine peut évaluer.

Ce qui échappe relève surtout du jugement. axe peut confirmer qu'une image possède un texte alternatif, un bouton un nom et un champ une étiquette. Il ne peut pas dire si le texte décrit l'image, si le nom explique la fonction ou si la page

a du sens lue à voix haute dans l'ordre. Ces questions nécessitent une personne avec lecteur d'écran et clavier. Le temps gagné sur les contrôles mécaniques est du temps qu'elle peut leur consacrer.

Le choix des étiquettes est un compromis distinct. Les seules règles WCAG donnent un résultat correspondant clairement à une déclaration de conformité, et à rien d'autre. Ajouter les bonnes pratiques détecte des difficultés non nommées dans les WCAG, mais rencontrées par les utilisateurs, comme du contenu hors de toute région ou une page sans titre de premier niveau. En contrepartie, quelques échecs portent sur des points qu'aucun audit de conformité ne citerait.

À quoi cela ressemble en pratique

Deux pages de ce site montrent les responsabilités confiées au moteur.

Dans la suite de tests d'une plateforme, il accompagne quatre couches de tests, et un contrôle axe-core exige zéro violation WCAG 2.0 et 2.1 A et AA. L'accessibilité devient ainsi un échec de compilation plutôt qu'un constat d'audit plusieurs mois plus tard. Le moment compte : un défaut qui bloque la compilation est corrigé par la personne qui vient de l'écrire, pendant que la modification est encore ouverte.

Sur ce site, la page des crédits décrit le moteur auditant des pages représentatives, une par modèle dans chaque langue, selon WCAG 2.2 A et AA et dans les deux thèmes de couleurs, au sein du contrôle obligatoire plutôt que dans un rapport séparé. Les deux thèmes permettent de bloquer même une erreur de contraste propre au sombre. Choisir une page par modèle suppose une structure partagée par les pages qui l'utilisent : c'est ce qui a permis à 38 pages de représenter un site de 2 268 en octobre 2026. Les règles automatisées ne couvrent pas tous les jugements qu'exigent les WCAG. Le site tient donc un dossier par critère de succès plutôt qu'une simple note globale.

Où les problèmes surviennent

La plupart des échecs axe sont des échecs de confiance : une exécution qui vérifie moins qu'il n'y paraît, ou un résultat auquel on fait dire davantage.

Une liste d'étiquettes laisse des règles de côté. Les étiquettes ne sont pas cumulatives. `wcag2aa` ne contient que les règles correspondant aux critères AA de WCAG 2.0 : s'y limiter omet toutes les règles A. La documentation API demande explicitement `wcag2a` et `wcag2aa`. De même, `wcag21aa` ne contient que les règles correspondant aux critères ajoutés par 2.1, et les seules étiquettes WCAG n'exécutent jamais une règle `best-practice`. Les descriptions indiquent que les règles WCAG 2.2 sont désactivées par défaut : une exécution sans étiquettes les omet, leur étiquette les active. Nommez chaque étiquette à couvrir et conservez la liste près de l'affirmation qu'elle étaye :

```
const results = await axe.run(document, {
    runOnly: {
        type: "tag",
        values: ["wcag2a", "wcag2aa", "wcag21a", "wcag21aa", "wcag22aa"],
    },
});
```

Une règle absente des quatre listes n'a pas du tout été exécutée. Relisez donc les listes à chaque modification de configuration.

Une exécution propre est prise pour une page accessible. Zéro violation signifie seulement que les règles exécutées n'ont prouvé aucun échec. Cela ne dit rien de la justesse des textes alternatifs, de l'ordre cohérent du focus ou de la possibilité de terminer une tâche avec un lecteur d'écran. Considérez ce résultat comme un minimum et planifiez les vérifications impossibles pour lui.

La liste de revue est abandonnée. Un contrôle qui ne bloque que sur les violations laisse passer `incomplete` sans que personne les voie. Le moteur a examiné ces éléments sans pouvoir trancher : ce sont précisément ceux qui nécessitent le plus une personne. Envoyez-les à quelqu'un qui les lira ou faites-les aussi bloquer la compilation.

La page n'est testée que dans un état. Le moteur examine le rendu au moment de son passage. Menu fermé, boîte de dialogue non ouverte, message apparaissant seulement après une saisie invalide et autre thème de couleurs échappent tous à une exécution au chargement initial. Placez la page dans chaque état important et relancez le moteur dans chacun.

Le moteur s'exécute là où rien n'est dessiné. Un document simulé n'a ni mise en page ni couleur calculée, et le projet indique que le contraste n'y fonctionne pas. Les tests unitaires avec axe détectent encore les problèmes structurels ; le contraste et tout aspect visuel relèvent d'un véritable navigateur.

Une règle est désactivée pour pouvoir publier. La couper pour toute l'exécution parce qu'un composant la déclenche masque ses défauts partout ailleurs, y compris sur les pages à venir. Excluez plutôt l'élément concerné, avec une note indiquant pourquoi et quand cette exclusion devra disparaître.

Pour aller plus loin

- axe-core — code source du moteur, principes annoncés et navigateurs pris en charge.
- Description des règles — chaque règle avec impact et étiquettes, classée par WCAG, bonnes pratiques, AAA, expérimentation et obsolescence.
- API axe-core — étiquettes, option `runOnly`, inclusion ou exclusion de parties de page et structure des quatre listes.
- Tests d'accessibilité — guide Playwright pour exécuter axe dans les tests navigateur, notamment après interaction.

Glossaire

Choisissez selon vos contraintes réelles

Présentez le processus de travail, les interfaces existantes et les utilisateurs qui ont besoin du changement. Nous pouvons définir une livraison testable et une transmission permettant à votre équipe d'assurer la maintenance des éléments livrés.

Consulter les tarifs Contactez-nous →

Rendez la terminologie concrète

Demandez comment « Qu'est-ce qu'axe ? » s'applique à votre organisation. Un exemple concret tiré de votre système actuel suffit pour engager la discussion.

Découvrir nos prestations

welcome@engineer.company +45 50 10 61 84 Copenhague, Danemark

Emportez cette page avec vous

Fiche de contact

Créé avec soin et amour par D.

CVR 45969703 · Tagensvej 105C, 3. th, 2200 København N

Droit d'auteur © 2025 – aujourd'hui · Engineer ApS · Tous droits réservés.

Imprimé depuis https://engineer.company/fr/glossary/axe/