

Formation Éco-conception pour les développeurs : coder des applications web et mobiles sobres et conformes au RGENS

Réf. SF34755

Mesurez l'empreinte de vos applications avec EcoIndex et GreenFrame, allégez votre code du front-end au back-end et au mobile, et faites la preuve de leur conformité aux critères techniques du RGENS !

3 j
Durée

2 550 € HT
Inter / participant

0
Sessions à venir

Objectifs pédagogiques

- ✓ Identifier les critères techniques du RGENS applicables à une application et les principales sources d'impact environnemental côté code
- ✓ Diagnostiquer l'empreinte environnementale d'une application avec EcoIndex, GreenFrame et les outils de développement du navigateur
- ✓ Optimiser le front-end web : poids des ressources, JavaScript, mode de rendu
- ✓ Optimiser le back-end : API, accès aux données, traitements et configuration de l'hébergement
- ✓ Appliquer les bonnes pratiques d'éco-conception propres aux applications mobiles
- ✓ Automatiser le contrôle de la sobriété dans le cycle de développement (analyse statique, mesure en intégration continue, budgets environnementaux)
- ✓ Évaluer la conformité d'une application aux critères techniques du RGENS et produire un plan d'action priorisé

Public visé & prérequis

Public visé Architectes logiciels, Développeurs

Prérequis- Pratiquer le développement web (HTML, CSS, JavaScript) et au moins un langage back-end - Connaître les bases du protocole HTTP et des bases de données - Des notions de développement mobile sont utiles mais non indispensables - Recommandé : avoir suivi « Éco-conception des services numériques – Niveau 1 » ou une Fresque du numérique

Programme détaillé

Éco-conception côté code : enjeux utiles au développeur et cadre RGENS

- Où se situent les impacts : terminaux, réseaux et centres de données ; poids prépondérant de la fabrication des terminaux et rôle du code dans leur obsolescence
- Sobriété et performance : pourquoi une application rapide n'est pas forcément sobre ; effets rebond ; différence entre mesure et estimation
- RGENS : origine, portée réglementaire (loi REEN, acteurs publics, appels d'offres), structure en thématiques et critères avec priorité, difficulté et applicabilité
- Critères techniques du RGENS : repérage des thématiques qui concernent le développeur (architecture, UX/UI, contenus, frontend, backend, hébergement, algorithmie)
- Unité fonctionnelle et budget environnemental : rapporter l'impact à un service rendu et fixer des seuils par page ou par parcours
- Repères internationaux : la Software Carbon Intensity (SCI) comme méthode de calcul de l'intensité carbone d'un logiciel par unité fonctionnelle

Activités pratiques :

- Découverte de l'application fil rouge (front web, API, base de données, client mobile) et de son architecture
- Repérage des critères techniques du RGENS applicables à l'application fil rouge

Mesurer et estimer l'empreinte d'une application

- Outils de développement du navigateur : requêtes, données transférées et décodées, bridage réseau, vue adaptative, Lighthouse

- EcoIndex : principe de calcul (taille du DOM, nombre de requêtes, poids de la page), note de A à G, équivalents eau et gaz à effet de serre
- Déclinaisons d'EcoIndex : site web, extension GreenIT-Analysis, outil en ligne de commande, plugin Lighthouse
- GreenFrame : mesure d'un scénario utilisateur scripté, estimation de la consommation du navigateur, du réseau et des conteneurs serveur et base de données
- Limites et complémentarité des outils : page isolée ou parcours, front seul ou pile complète, lecture critique des scores
- Protocole de mesure reproductible : choix des pages et parcours représentatifs, configuration, documentation des résultats

Activités pratiques :

- Mesure de référence de l'application fil rouge : EcoIndex sur les pages clés, GreenFrame (ou équivalent) sur un parcours utilisateur
- Comparaison des résultats obtenus avec plusieurs outils et analyse des écarts
- Définition d'un budget environnemental pour une page et un parcours de l'application

Front-end web sobre

- Images : formats modernes, compression, dimensionnement adapté, attribut srcset et chargement différé natif
- Vidéo, audio, polices et PDF : lecture automatique proscrite, compression, sous-ensembles de polices ou polices système, alternatives légères
- JavaScript et dépendances : poids embarqué, tree-shaking, découpage du code, remplacement de bibliothèques par des fonctionnalités natives
- HTML et CSS natifs : éléments details et dialog, formulaires et autocomplétion, requêtes média, plutôt que des composants scriptés
- Modes de rendu : génération statique, rendu côté serveur, rendu côté client, Single Page Application, et leur impact sur le transfert et le calcul
- Contenus tiers et activation à la demande : façades pour les vidéos, cartes et widgets, limitation des scripts tiers
- Mobile first, amélioration progressive et compatibilité avec les terminaux anciens et les connexions lentes
- Critères de choix d'un framework : poids embarqué, mode de rendu, dépendances, plutôt qu'un palmarès

Activités pratiques :

- Optimisation des pages clés de l'application fil rouge (ressources, JavaScript, composants natifs) et nouvelle mesure EcoIndex
- Remplacement d'un contenu tiers par une façade à activation volontaire
- Comparaison de l'empreinte d'une même page en rendu statique et en rendu côté client

Back-end, API, données et hébergement

- Configuration HTTP : compression, en-têtes de cache côté client, versions récentes du protocole HTTP, redirections maîtrisées
- API sobres : pagination, sélection des champs, regroupement des appels, cache côté serveur
- Accès aux données : requêtes efficaces, problème des requêtes N+1, index, volumétrie, conservation et purge des données
- Traitements : traitements asynchrones et par lots, planification des tâches lourdes, choix des algorithmes et des structures de données
- Algorithmie et IA : questionner le recours à un modèle, dimensionner l'entraînement et l'inférence, mesurer leur coût
- Architecture et hébergement : dimensionnement, conteneurs, extinction des environnements inutilisés, choix de la région et de l'hébergeur, décommissionnement

Activités pratiques :

- Optimisation d'un endpoint et d'une requête de l'application fil rouge, puis nouvelle mesure du parcours avec GreenFrame (ou équivalent)
- Audit de la configuration HTTP de l'application (compression, cache, protocole) et corrections
- Mise en place d'une règle de purge ou d'archivage des données

Applications mobiles sobres

- Poids de l'application et des mises à jour : ressources embarquées, dépendances, téléchargement à la demande
- Durée de vie des terminaux : prise en charge des anciennes versions de système et des terminaux d'entrée de gamme
- Réseau : regroupement des appels, cache local, mode hors ligne, synchronisation différée
- Énergie : capteurs et géolocalisation, tâches de fond, notifications, animations
- Outils de profilage : profileurs d'énergie, de réseau et de mémoire d'Android Studio et de Xcode
- Application native, hybride ou PWA : choisir selon le besoin réel

Activités pratiques :

- Profilage d'un parcours du client mobile fil rouge et identification des postes de consommation
- Revue de la configuration réseau et des tâches de fond d'une application mobile, avec propositions de corrections

Industrialiser la sobriété dans le cycle de développement

- Analyse statique : SonarQube et règles d'éco-conception du Green Code Initiative (creedengo), patterns et anti-patterns de code
- Mesure automatisée : EcoIndex en ligne de commande et GreenFrame sur les parcours clés dans la chaîne d'intégration continue
- Budgets environnementaux : seuils de non-régression sur le poids, les requêtes et le score, alertes et suivi dans le temps
- Backlog et définition de fini : critères d'acceptation sobres, prise en compte des critères RGESEN dans les user stories
- Partager les résultats : argumentaire auprès de l'équipe et du produit, communication sans greenwashing

Activités pratiques :

- Analyse statique de l'application fil rouge et correction des anti-patterns relevés
- Ajout d'une mesure automatisée avec seuil bloquant dans le pipeline (ou dans un script local)

Atelier : conformité de l'application aux critères techniques du RGESEN

- Grille de vérification : pour chaque critère technique, statut applicable, conforme ou non conforme, preuve associée (mesure, configuration, code)
- Mesure après optimisation : gains EcoIndex et GreenFrame par rapport à la mesure de référence, lecture honnête des résultats
- Contribution du développeur à la déclaration d'écoconception : éléments techniques à fournir au référent du service
- Plan d'action technique : priorisation des critères non conformes selon l'effort et l'impact

Pourquoi choisir Sparks Formation ?

20 ans d'expertise IT, des formateurs experts et des sessions garanties contre l'annulation.

20 ans
d'expertise

1 068+
formations

11 178
apprenants formés/an

+67
Net Promoter Score

Avis des participants

4,9/5 — 39 avis

Maxime V. — session très agréable, dense mais toujours agréable. formateur très à l'écoute, prends le temps de répondre aux questions. on apprend l'utilisation des fonctions mais aussi les processus mis...

Cécile L. — Très bon instructeur qui explique toujours en montrant du code, aucune slide et une présentation vivante ce qui permet de rester intéressé et concentré tout au long !...

Ils nous font confiance

Cette année notre filière Développement Web & Fullstack a accompagné plus de 111 entreprises dont :

ARPEGE APSIDE TOP CEGEDIM CONSERVATOIRE NATIONAL DES ARTS ET METIERS ECONOCOM APPS, CLOUD & DATA ORANGE BUSINESS
ORANGE BUSINESS SERVICES SA SOPRA STERIA ADVENTIEL AGATE AGENCE ALPINE DES TERRITOIRES +101 autres

Financement & certifications

Cette formation est éligible aux financements OPCO et FNE-Formation. Nos équipes vous accompagnent dans le montage de votre dossier.



Réservez votre place dès maintenant

20 ans d'expertise · 1 000+ formations · NPS +67 · réponse sous 24h ouvrées

0 805 950 800

demande@sparks-formation.com