

Ancsi

Le guide d'éco-conception

2026



Pourquoi adopter l'éco-conception ?

Contexte



Le numérique représente $\approx 4\%$ des émissions mondiales de CO₂, autant que l'aviation civile.

Problème



Chaque fonctionnalité inutile ou mal conçue augmente la consommation énergétique.

Solution : l'éco-conception



L'essentiel des impacts se joue dès la conception.

Les impacts ne proviennent pas uniquement des centres de données ou des terminaux.

→ Concevoir des services utiles, optimisés et durables.

→ Privilégier la pertinence, la simplicité et l'inclusion, en réponse aux besoins réels plutôt qu'à la surenchère fonctionnelle.

→ Réduire l'extraction de ressources rares et la production de déchets électroniques.

Jusqu'à 70 % des impacts environnementaux se déterminent dès la phase de conception.

Bénéfices d'un service éco-conçu

Pour l'entreprise

Résilience accrue face à l'obsolescence matérielle ou aux pics de trafic.

Réduction de l'empreinte carbone et numérique.

Réponse aux attentes des clients et collaborateurs en matière de responsabilité environnementale.

Réduction des coûts d'infrastructure (hébergement, stockage, maintenance).



Pour l'utilisateur

Meilleure expérience utilisateur (temps de réponse, clarté, ergonomie).

Interfaces compatibles avec équipements anciens et connexions limitées.

Parcours et fonctionnalités simplifiés.

Services plus rapides et agréables.



Comprendre les impacts environnementaux du numérique

Cycle de vie d'un service numérique



Un service numérique n'est jamais virtuel. Derrière chaque application, site web ou plateforme, se cache un cycle de vie complet qui mobilise des ressources, de l'énergie, des matériaux, des personnes et des machines. L'éco-conception consiste à agir à chaque étape de ce cycle pour en réduire les impacts environnementaux sans sacrifier la qualité ni l'usage.

Le saviez-vous ? Jusqu'à 70 % des impacts environnementaux se décident à cette étape (GreenIT.fr)



Facteurs aggravants (!)**Code et contenus**

Un code mal optimisé génère des calculs et requêtes superflus, augmentant la charge serveurs et réseau. Des images, vidéos ou fichiers trop lourds créent une charge accrue pour les data centers, les réseaux et les terminaux.

Où se cache la pollution numérique ?

Les impacts ne viennent pas que des centres de données
Smartphone, PC, TV, tablettes...

Terminaux (≈ 50 %)

- La fabrication très polluante (métaux rares, forte consommation d'énergie).
- Un smartphone nécessite en moyenne 70 matériaux différents et demande environ 70kg de matières premières pour un appareil de 150g.

Réseaux (≈ 4 %)

Mobile, fibre, Wi-Fi → 4% des émissions sont liées à la transmission des données (4G/5G, fibre, WI-FI).

- Plus un service est "bavard" (requêtes inutiles, ressources lourdes), plus il sollicite le réseau.

Centres de données (≈ 46 %)

Cloud, hébergement, streaming, IA...

- 46% de l'impact carbone du numérique provient des data centers, en forte hausse depuis la prise en compte de l'hébergement hors France.
 - Les centres de données consomment en continu de l'énergie, notamment pour le fonctionnement des serveurs et leur refroidissement permanent.
- Etude ADEME-ARCEP publiée en novembre 2024.

Les 6 grands principes de l'écoconception numérique

L'écoconception numérique consiste à concevoir, développer et maintenir des services numériques en réduisant leurs impacts environnementaux tout au long de leur cycle de vie. Elle s'appuie sur des approches issues de l'ACV (Analyse du Cycle de Vie), des référentiels institutionnels (ADEME, RGEN) et des pratiques de conception responsable.

**UTILITÉ**

Concevoir un service qui répond réellement aux besoins des utilisateurs sans ajouter de fonctionnalités inutiles.

- Définir le périmètre fonctionnel minimal
- Supprimer ou éviter les features peu utiles
- Clarifier les parcours pour réduire la complexité

**SOBRIÉTÉ**

Diminuer le poids de tout ce qui transite (images, vidéos, scripts, animations, polices, effets)

- Compresser, redimensionner, mutualiser les médias
- Éviter les éléments décoratifs non essentiels
- Réduire les requêtes et appels techniques



PERFORMANCE

Rendre le service rapide, fluide et peu consommateur de ressources côté serveur comme côté utilisateur.

- Optimiser le code et les algorithmes
- Limiter les traitements superflus
- Réduire le temps de chargement et les ressources utilisées



FRUGALITÉ DES DONNÉES

Limiter la quantité de données manipulées tout au long du cycle de vie du service.

- Ne collecter que les données strictement nécessaires
- Nettoyer régulièrement logs, sauvegardes et fichiers anciens
- Éviter les duplications et la rétention inutile



SOUTENABILITÉ

Allonger la durée de vie du service, limiter l'obsolescence et faciliter la maintenance.

- Développer une architecture modulaire et stable
- Réduire la dette technique
- Prévoir des évolutions simples sans tout reconstruire



ACCESIBILITÉ

Garantir un accès fluide quel que soit le terminal, la capacité réseau, les contraintes ou handicaps.

- Compatibilité avec des appareils variés, même ancien
- Fonctionnement en connexion faible
- Respect des standards d'accessibilité

En quelques chiffres

40 %

des fonctionnalités développées dans les services numériques ne sont jamais utilisées

Une page web sobre peut charger jusqu'à 4 fois plus vite et consommer

80 % d'énergie en moins

Le Référentiel Général d'Écoconception des Services Numériques (RGESN)

Le RGESN est le référentiel français dédié à l'éco-conception des services numériques. Publié par la DINUM en 2022, il guide les équipes projet pour réduire l'impact environnemental tout en préservant qualité, accessibilité et performance.

Structure du RGESN



Près de 80 bonnes pratiques réparties en 9 grandes catégories : stratégie, spécifications, architecture, UX/UI, contenus, frontend, backend, hébergement, algorithmie.

- Applicables à tous types de services : sites web, applications, services transactionnels publics et privés.
- Ressources fournies : glossaire, outils d'auto-évaluation (ODS/Excel), modèle de déclaration d'écoconception.

Ses objectifs



- Fournir un cadre méthodologique national pour l'écoconception des services numériques.
- Encourager l'intégration de pratiques sobres et responsables dès la conception.
- Harmoniser les démarches en proposant un socle commun de critères reconnus.
- Faciliter l'auto-évaluation et le pilotage grâce à des outils simples et mobilisables par tous les métiers.

Réduire la charge utile : l'enjeu

L'une des premières actions concrètes en éco-conception numérique consiste à réduire la charge utile d'un service. C'est à dire l'ensemble des ressources numériques chargées pour faire fonctionner un service :

- Poids des pages (HTML, CSS, JS)
- Images, polices, vidéos, fichiers embarqués
- Appels API ou chargements de modules dynamiques
- Données collectées ou synchronisées en tâche de fond...

Pourquoi réduire la charge utile ?



- Diminuer la consommation énergétique sur toute la chaîne (serveur → réseau → utilisateur).
- Prolonger la durée de vie des terminaux (moins de besoins matériels).
- Accélérer les temps de chargement → meilleure UX.
- Réduire l'empreinte carbone de chaque page vue et de chaque requête.

En quelques chiffres

2,8 Mo

C'est le poids médian d'une page web en France

70 à 90 %

du contenu chargé n'est pas utile à l'utilisateur

Élément ↓ Bonnes pratiques

Images	→	Compresser, formats WebP/AVIF, limiter les décoratives
Polices	→	Réduire les variantes, charger uniquement l'utilisé
JS	→	Supprimer modules inutiles, différer les scripts
CSS	→	Nettoyer, éviter les frameworks trop lourds
API	→	Regrouper les appels, limiter les refresh auto
Multimédia	→	Pas de lecture auto, qualité adaptable
Contenus	→	Aller à l'essentiel, supprimer l'obsolète

Alléger un service numérique, ce n'est pas le "rendre pauvre", c'est le rendre plus efficace, plus rapide, plus accessible... et *moins polluant*.

Minimiser les dépendances

Limiter les dépendances permet de réduire le poids des pages, d'améliorer la vitesse de chargement et de garder un service plus simple, plus fiable et moins énergivore.

Outils trop puissants pour des besoins simples

Il arrive qu'un site vitrine utilise React ou Angular pour afficher... une page statique.

→ Résultat : plusieurs centaines de Ko de JavaScript chargés pour rien, un rendu initial ralenti et un code plus difficile à maintenir.

L'invisible qui pèse lourd



Beaucoup d'éléments « cachés » alourdissent discrètement le service :

- scripts marketing
- plugins sociaux ou outils de personnalisation
- chatbots tiers

Ils sont souvent chargés sur toutes les pages, même quand personne ne les utilise. Cela multiplie les requêtes externes, augmente la consommation réseau et élève les risques d'attaque.

Moins de dépendances = plus de performance, plus de sécurité et un budget mieux maîtrisé.

Privilégier des solutions open source légères permet de bénéficier d'outils plus transparents, pérennes et adaptés, tout en évitant des alternatives commerciales souvent surdimensionnées.

Objectifs ↓ Bonnes pratiques

Éviter les frameworks lourds pour des usages simples	→	Préférer du HTML/CSS/JS natif ou un micro-framework ; limiter l'usage de React/Angular à des applications justifiées.
Charger uniquement ce qui est nécessaire	→	Utiliser le lazy loading ou le tree shaking pour éviter d'importer des modules inutiles.
Réduire les appels à des services tiers	→	Supprimer les scripts inutiles (tracking, social, chat...) ou le bien joué ! s'active uniquement à la demande.
Intégrer localement les ressources utiles	→	Héberger en local les polices, scripts essentiels, images et vidéos au lieu de dépendre de CDN externes.
Évaluer chaque dépendance à l'entrée du projet	→	Intégrer la question "peut-on s'en passer ?" dans les choix techniques dès la conception.
Mettre en place une revue régulière des dépendances	→	Identifier les éléments obsolètes, jamais utilisés ou remplaçables par une alternative plus sobre.

En quelques chiffres

80 %

du JavaScript présent sur une page n'est jamais exécuté.

5 dépendances en moins = jusqu'à **2x** moins de requêtes.

Un plugin tiers peut ralentir

de **2.5** à **5** secondes le chargement d'une page.

Optimiser les contenus multimédias

Les contenus (images, vidéos, textes, documents) sont essentiels, mais aussi l'une des principales sources de pollution numérique. Les optimiser, c'est les rendre plus utiles et plus légers, sans perdre en qualité pour l'utilisateur.

Images : le poids caché des pages web

- ↓
- Les images représentent souvent 50 à 70 % du poids d'une page.
- Mauvais format = jusqu'à ×10 d'empreinte inutile.
 - Trop haute résolution, jamais affichée en taille réelle.
 - Images décoratives souvent... décoratives seulement.

Vidéos : puissantes mais très énergivores

- ↓
- Une vidéo consomme jusqu'à 20× plus de bande passante qu'un texte équivalent.
- Éviter les vidéos auto-lancées ou en fond.
 - Toujours proposer une version compressée.
 - Les rendre optionnelles et adaptées à la connexion de l'utilisateur.

Textes et documents : plus c'est clair, plus c'est léger

- ↓
- Un texte structuré se lit plus vite et se charge plus vite. Les documents téléchargeables doivent être allégés :
- PDF/PPT/Word trop lourds → mauvaise expérience + données gaspillées.
 - Un PDF compressé peut être 10× plus léger.

Objectifs

↓ Bonnes pratiques

- | | | |
|---|---|--|
| Réduire le poids des images | → | Utiliser des formats modernes (WebP, AVIF), compresser systématiquement, adapter la taille à l'affichage réel. |
| Supprimer les images inutiles | → | Ne pas afficher d'images décoratives sans valeur informative ; supprimer les doublons. |
| Éviter la lecture automatique des vidéos | → | Ne pas déclencher de vidéo sans action explicite ; ne pas utiliser de vidéos comme fond décoratif. |
| Proposer plusieurs qualités vidéo | → | Permettre à l'utilisateur de choisir la version la plus légère selon sa connexion ou son équipement. |
| Alléger les documents | → | Réduire le nombre de pages, compresser les images internes, éviter les effets visuels inutiles. |
| Privilégier le contenu natif plutôt que le document joint | → | Préférer un texte HTML clair à un PDF téléchargeable quand c'est possible. |
| Structurer les textes | → | Utiliser des titres, paragraphes courts, listes, liens explicites : plus rapide à lire, plus léger à charger. |

En quelques chiffres

50–70 %

du poids d'une page = les images.

Une vidéo auto-lancée = jusqu'à **20 Mo** par visite.

Un PDF compressé = jusqu'à **×10** plus léger sans perte.

Penser la performance comme levier environnemental

Un service plus rapide consomme moins d'énergie, réduit la charge des terminaux et des serveurs, limite les requêtes réseau et améliore la sobriété globale. La performance n'est donc pas seulement un enjeu UX : c'est un véritable levier environnemental.

La performance : un impact souvent sous-estimé



Améliorer les temps de chargement, ce n'est pas seulement optimiser l'expérience :

- Chaque milliseconde économisée réduit la consommation CPU.
- Diminue l'énergie nécessaire au rendu.
- Baisse la bande passante utilisée.

Un service lent entraîne au contraire des comportements à forte empreinte : rechargements répétés, abandon, multitâche... jusqu'à la surchauffe de terminaux anciens.

Inclure les usages bas débit et les terminaux modestes



Penser performance, c'est aussi favoriser l'inclusion. Un service éco-conçu doit fonctionner sur :

- Un smartphone en 3G
- Un PC ancien
- Une connexion dégradée

Concevoir un service rapide pour tous limite l'obsolescence matérielle et élargit l'accès au numérique.

Textes et documents : plus c'est clair, plus c'est léger



Un texte structuré se lit plus vite et se charge plus vite.

Les documents téléchargeables doivent être allégés :

- PDF/PPT/Word trop lourds → mauvaise expérience + données gaspillées.
- Un PDF compressé peut être 10× plus léger.

Objectifs



Bonnes pratiques



Réduire les temps de chargement	→	Optimiser les ressources (images, JS, CSS), utiliser des CDN sobres, activer le cache navigateur.
Limiter les requêtes HTTP	→	Regrouper les fichiers, différer les appels secondaires, supprimer les scripts non essentiels.
Accélérer le rendu initial	→	Utiliser le lazy loading pour les images ou composants non critiques ; charger d'abord l'essentiel.
Cibler les connexions lentes	→	Mettre en place un mode "économie de données" ou une version simplifiée activable selon la bande passante.
Tester sur des terminaux variés	→	Mesurer la performance sur mobile d'entrée de gamme, connexion 3G, ou ordinateur vieillissant.
Surveiller les indicateurs clés	→	Suivre les Core Web Vitals (LCP, FID, CLS), le poids total de la page, et les temps de réponse serveur.

Converger avec l'accessibilité, l'inclusion et l'accessibilité mobile

L'éco-conception ne vise pas seulement à réduire l'empreinte environnementale : elle rend aussi les services plus universels, équitables et inclusifs. Un service accessible est souvent plus sobre... et un service sobre est plus facilement utilisable par tous.

Accessibilité + sobriété : les mêmes leviers, les mêmes bénéfices



Bon contraste, structure claire, interface prévisible, chargement rapide : ce qui facilite la navigation des personnes en situation de handicap allège aussi les pages, réduit les scripts et simplifie l'expérience.

Les exigences du RGAA/WCAG rejoignent celles du RGSN :

- Contenus bien structurés → moins de balises décoratives et de CSS superflu
- Alternatives textuelles → meilleure compatibilité offline ou mode texte
- Navigation clavier fluide → code plus sémantique, plus robuste et plus léger

Inclusion numérique : couvrir la diversité des usages



Un service éco-conçu intègre les besoins des publics éloignés du numérique (personnes âgées, peu connectées, peu à l'aise avec la lecture ou en situation d'illectronisme). Privilégier la clarté, la progressivité et la simplicité profite aux plus vulnérables... et améliore l'expérience pour l'ensemble des utilisateurs.

Accessibilité mobile : une norme incontournable



60 % des usages web se font sur mobile. Un service trop lourd, difficile à manipuler ou mal adapté aux petits écrans sera vite abandonné. Optimiser pour le mobile, c'est automatiquement concevoir plus direct, plus léger et plus efficace.

Objectifs



Bonnes pratiques



Améliorer la lisibilité	→	Utiliser des tailles de texte suffisantes, un contraste élevé, une hiérarchie claire.
Réduire la complexité des parcours	→	Limiter les étapes, expliciter les actions, éviter les redondances et détours.
Assurer la compatibilité clavier / écran tactile	→	Vérifier les tailles de zones cliquables ($\geq 44px$), activer les focus visibles, éviter les hover-only.
Privilégier des contenus accessibles	→	Fournir des alternatives textuelles, éviter les contenus dépendants de la couleur ou du son.
Tester en condition réelle	→	Utiliser un terminal mobile basique, un lecteur d'écran, un zoom 200 %, un mode sombre, etc.
Adopter un design responsive et adaptatif	→	Adapter les interfaces aux petites résolutions sans surcharge visuelle ni perte de contenu.

En quelques chiffres...

1 Français / **5** est concerné par une forme de handicap (permanent ou temporaire).

60 % des connexions à internet se font sur mobile.

Une interface accessible est

30 à **50 %** plus rapide à utiliser pour tous.

Converger avec l'accessibilité, l'inclusion et l'accessibilité mobile

Un service éco-conçu doit vivre longtemps, évoluer sans s'alourdir et pouvoir disparaître proprement. Trop de services deviennent obsolètes faute d'entretien... ou continuent à consommer des ressources simplement parce qu'ils n'ont jamais été éteints.

Penser la maintenance dès le départ

Un code difficile à comprendre ou mal documenté génère rapidement de la dette technique :

- Plus de ressources mobilisées à chaque évolution,
- Délais rallongés
- Perte de maîtrise interne

Un service maintenable est naturellement plus sobre, plus fiable et plus pérenne.

Évoluer sans tout reconstruire



Les services changent : nouvelles fonctionnalités, obligations réglementaires, refontes... Sans cadre clair, cela crée du code mort, des dépendances oubliées, des données incohérentes. Anticiper les évolutions permet de conserver un service léger, cohérent et durable dans le temps.

Prévoir la fin dès la conception



Un service peut devenir inutile ou être remplacé. Sans procédure de fin de vie, il continue à :

- Consommer de l'énergie (serveurs, bases, logs)
- Mobiliser l'attention des équipes
- Ralentir les décisions

Prévoir un protocole d'extinction, une redirection ou une documentation de reprise fait partie intégrante de l'éco-conception.

Objectifs

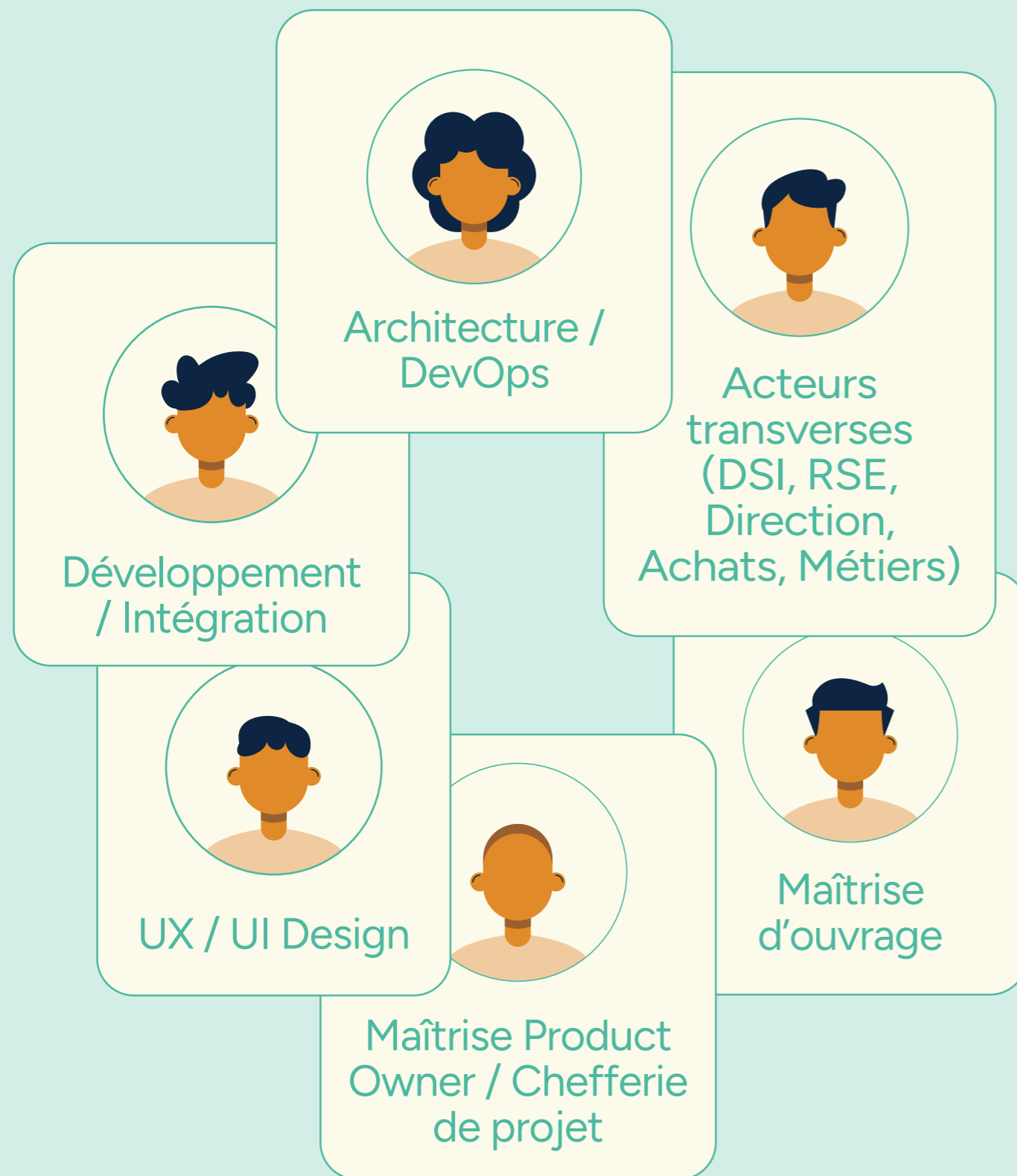


Bonnes pratiques



- | | | |
|--|---|---|
| Assurer la maintenabilité du code | → | Rédiger une documentation minimale, commenter le code, éviter les complexités inutiles. |
| Préparer l'évolutivité | → | Factoriser les composants, anticiper les cas d'usage futurs, limiter les interdépendances techniques. |
| Nettoyer régulièrement les contenus | → | Supprimer les fichiers obsolètes, les bases non utilisées, les logs anciens. |
| Réduire la dette technique | → | Instaurer des revues de code régulières, prioriser la correction des anomalies de structure. |
| Prévoir la désactivation | → | Définir un protocole de fin de vie : redirection, archivage, suppression. |
| Intégrer la durée de vie au cahier des charges | → | Poser la question : combien de temps ce service doit-il exister, et dans quelles conditions ? |

Rôles et responsabilités dans un projet éco-conçu



Maitrise d'ouvrage (MOA)

Rôle



Définir les besoins et cadrer le projet pour garantir sobriété, utilité et éco-conception dès le départ.

Responsabilités clés



- Cadrer un besoin réaliste et mesuré : éviter les fonctionnalités gadgets.
- Inclure des critères environnementaux dans les livrables (performance, accessibilité, pérennité).
- Impliquer les autres métiers dès le début : design, dev, contenus, RSE...

Mise en situation



Appel d'offres avec demandes multiples et complexes → MOA reformule le besoin centré sur les usages prioritaires, clarifie les populations cibles et demande un prototype simple.

Zone d'influence

Phase projet	Marge d'action	Impact potentiel
Cadrage initial	Définir besoins sobres, utiles et réalistes	Réduit la complexité inutile dès le départ
Cahier des charges	Intégrer sobriété, accessibilité, performance	Aligne tous les partenaires sur une ambition responsable
Sélection prestataires	Valoriser critères qualitatifs et Numérique Responsable	Encourage les pratiques vertueuses côté fournisseur
Pilotage/arbitrage	Refuser demandes hors périmètre	Prévient les effets rebond et ajouts tardifs coûteux
Bilan projet	Intégrer durabilité dans critères de réussite	Fait émerger projets sobres comme réussites stratégiques

Product Owner / Chefferie de projet

Rôle

Transformer les besoins métiers en livrables concrets tout en préservant sobriété et cohérence.

Responsabilités clés

- Prioriser les fonctionnalités selon leur valeur réelle.
- Arbitrer en faveur de la simplicité et limiter l'effet "scope creep".
- Coordonner les métiers pour aligner UX, technique et contenus sur l'objectif de sobriété.

Mise en situation

Ajout d'une vidéo décorative en milieu de sprint → PO évalue impact (poids, accessibilité, performance) et propose une alternative optimisée ou refuse si inutile.

Zone d'influence

Phase projet	Marge d'action	Impact potentiel
Backlog/priorisation	Classer selon valeur et impact environnemental	Réduit développements inutiles et énergivores
Sprints/itérations	Arbitrer solutions techniques et UX sobres	Évite choix lourds et difficilement réversibles
Communication inter-métiers	Assurer cohérence objectifs	Maintient alignement sur éco-conception
Gestion périmètre	Résister aux fonctionnalités gadgets	Limite complexité et effets rebond
Recette/validation finale	Intégrer critères performance et sobriété	Garantit produit final aligné avec Numérique Responsable

UX/ UI Design

Rôle

Concevoir des parcours clairs, utiles et accessibles, réduisant la charge cognitive et technique.

Responsabilités clés



- Simplifier les parcours et limiter les étapes.
- Privilégier lisibilité et hiérarchie visuelle.
- Limiter éléments graphiques superflus et intégrer accessibilité dès le design.

Mise en situation



Carrousel plein écran → remplacé par image optimisée unique ou illustration SVG et scripts réduits.

Zone d'influence

Phase projet



Marge d'action



Impact potentiel



Conception parcours

Simplifier chemins vers objectifs principal

Réduit temps et ressources nécessaires

Création maquettes

Choisir éléments graphiques légers

Diminue poids et accélère chargement

Accessibilité intégrée

Garantir lisibilité et navigation mobile

Améliore expérience et réduit refontes ultérieures

Validation design

Vérifier impact choix graphiques

Évite charges lourdes inutiles

Collaboration dev

Fournir assets optimisés

Assure mise en oeuvre fidèle et sobre

Développement / Intégration

Rôle



Écrire un code léger, maintenable et performant, tout en anticipant son évolution.

Responsabilités clés



- Optimiser le code : éviter duplications et code mort
- Limiter dépendances et privilégier solutions open source sobres
- Gérer chargement intelligemment : lazy loading, compression, cache
- Penser évolutivité et maintenance, collaborer avec les autres métiers

Mise en situation



Galerie d'images → utilisation de HTML/CSS natif ou JS minimal, images chargées uniquement à l'affichage et compressées.

Zone d'influence

Phase projet	Marge d'action	Impact potentiel
Choix techniques	Choisir solutions sobres et adaptées	Réduit poids et complexité
Écriture code	Établir Standards sobriété et maintenabilité	Améliore performance et évolutivité
Gestion ressources	Optimiser et compresser fichiers	Diminue bande passante et accélère chargement
Tests performance	Mesurer et corriger lenteurs	Évite optimisations coûteuses tardives
Documentation	Faciliter reprise	Réduit dette technique et prolonge durée de vie

Architecte / Devops

Rôle

Dimensionner et gérer l'infrastructure pour réduire consommation, augmenter résilience et durabilité.

Responsabilités clés

- Éviter surdimensionnement systématique
- Rationaliser et mutualiser environnements
- Optimiser stockage et sauvegardes
- Automatiser CI/CD avec discernement, choisir hébergement responsable

Mise en situation

Environnement de test oublié → réduire animations, structuration sémantique, mode sombre, suppression vidéos auto-lancées.

Zone d'influence

Phase projet	Marge d'action	Impact potentiel
Dimensionnement	Choisir une Infrastructure adaptée et scalable	Réduit consommation inutile et prolonge durée de vie
Environnements	Limiter/mutualiser instances	Diminue empreinte énergétique continue
Stockage	Avoir des Durées de rétention claires, nettoyage	Évite stockage massif inutile
Automatisation/CI-CD	Utiliser des pipelines sobres	Réduit charge serveur et gaspillage
Hébergement	Choisir de fournisseurs engagés	Cohérence stratégie Numérique Responsable

Acteurs transverses (DSI, RSE, Direction, Achats, Métiers)

Rôle

Rôle : Structurer les moyens, priorités et contraintes pour soutenir l'éco-conception.

Responsabilités clés



- DSI : standards techniques sobres
- RSE : stratégie Numérique Responsable et sensibilisation équipes
- Direction : impulsion et arbitrage en faveur de sobriété
- Achats : intégrer critères Numérique Responsable dans appels d'offres
- Métiers : besoins clairs, réalistes, mesurés

Mise en situation



Appel d'offres centré sur coût et délai → intégrer exigences éco-conception et valoriser prestataires responsables.

Phase projet	↓	Marge d'action	↓	Impact potentiel
--------------	---	----------------	---	------------------

DSI		Définir standards techniques sobres		Influence tous les projet
RSE		Intégrer éco-conception stratégie/formation		Acculture équipes et aligne projets
Direction		Arbitrer et donner les moyens		Légitime choix sobres face aux priorités
Achats		Établir Critères Numérique Responsable appels d'offres		Favorise prestataires responsables
Métiers		Définir Besoins fonctionnels clairs		Évite sur-fonctionnalité et recentre sur l'essentiel

Méthodes et outils pour mettre en œuvre l'écoconception

Pour être efficace, l'écoconception doit être intégrée dans la méthodologie de gestion de projet dès le départ. Intégrer l'éco-conception dans les méthodes projet (Agile, cycle en v, design thinking). On évite de corriger à la fin ce qu'on aurait pu éviter au début.

En cycle en V : une vigilance à chaque jalon



→	Phase	Question éco-conçue
→	Spécifications	Est-ce vraiment utile ?
→	Conception	Est-ce simple, accessible, léger ?
→	Développement	Est-ce optimisé ?
→	Tests	Est-ce performant et accessible ?
→	Mise en prod	Est-ce dimensionné au juste besoin ?

En Agile : la sobriété dans chaque sprint



- User stories avec critères NR
- Indicateurs simples (poids, temps de chargement)
- Rétrospectives pour ajuster

Exemple visuel :

"Une user story = valeur métier + valeur environnementale"

Les référentiels comme cadre de référence

L'éco-conception numérique repose sur des référentiels publics et partagés qui donnent un cadre commun. Ils permettent d'éviter les approches trop subjectives et d'ancrer les projets dans des pratiques reconnues. L'idée n'est pas de tous les appliquer intégralement, mais de s'en inspirer pour poser des repères clairs à chaque étape :

- Se repérer
- Parler le même langage
- Objectiver les décisions

↓ Les 4 référentiels clés 🔍

Référentiel	→	À quoi il sert
RGESN	→	Vision globale et gouvernance
GreenIT.fr	→	Pratique terrain web
GR491	→	Qualité et durabilité logicielle
RGAA / WCAG	→	Accessibilité & sobriété


Les outils de mesures

Les référentiels fixent des règles, mais pour progresser il faut aussi mesurer l'impact concret des services numériques. Plusieurs outils permettent aujourd'hui d'évaluer rapidement le poids, la performance et l'empreinte environnementale d'un site ou d'une application. Leur objectif n'est pas de donner une valeur absolue parfaite, mais de fournir des indicateurs pour améliorer en continu.

↓ Les outils essentiels 🔍


Outil	→	Sert à quoi
EcoIndex	→	Performance environnementale globale
Website Carbon	→	Estimation CO ₂
GreenIT Analysis	→	Audit éco-conception
Lighthouse	→	Performance & accessibilité

3 moments clés pour les utiliser




Au démarrage

établir un état des lieux de la performance et du poids des pages clés.



En cours de développement

tester régulièrement pour corriger les dérives au fil de l'eau.



En production

suivre quelques indicateurs simples (poids moyen, temps de chargement, score EcoIndex) pour mesurer les améliorations.

Les bonnes pratiques par contexte spécifique



Site web

Un site sobre est plus rapide, plus accessible et mieux référencé.



Application mobile

Une app éco-conçue respecte la batterie, le stockage et l'attention.



Cloud & API

Le cloud sobre, c'est moins d'appels, moins de stockage, moins d'infrastructures inutiles.



Documentation numérique

Un document sobre est plus lisible et plus durable.

↓ Réduire le poids des contenus

Web	→	images compressées, WebP, limiter les vidéos en autoplay
Mobile	→	limiter les assets embarqués, réduire les libs, mode basse consommation
Cloud/API	→	supprimer logs trop bavards, payloads trop lourds
Docs numériques	→	compression PDF, limiter les images HQ

↓ Compression PDF, limiter les images HQ

Web	→	moins de scripts, moins d'APIs appelées au scroll
Mobile	→	limiter les tâches en arrière-plan
Cloud/API	→	supprimer les appels récurrents inutiles
Docs numériques	→	limiter les liens dynamiques, charger seulement ce qui compte

↓ Nettoyer et archiver régulièrement

Web	→	contenus obsolètes, pages zombies
Mobile	→	anciennes versions, ressources inutiles
Cloud/API	→	environnements jamais supprimés, logs infinis
Docs numériques	→	versions multiples, surcharge documentaire

Mesurer, piloter et améliorer dans la durée

L'éco-conception n'est pas un livrable figé, mais une démarche continue. Les usages évoluent, les équipes changent, de nouvelles fonctionnalités apparaissent. Sans suivi, un service initialement sobre peut vite se charger en contenus, en scripts et en dépendances, jusqu'à perdre les bénéfices acquis.

Les indicateurs



Ce qu'on mesure doit être compris par tout le monde.

- Poids moyen des pages clés
- Nombre de requêtes
- Temps de chargement mobile
- Score EcoIndex
- Taux de suppression de contenus obsolètes

Une gouvernance simple suffit



3 piliers :

- Un ou deux référents éco-conception
- Des critères Numérique Responsable dans les comités projet
- Des jalons Numérique Responsable dans les checklists de validation

Progresser vaut mieux que viser la perfection



Structure en 3 blocs :

- Ce que disent les utilisateurs → parcours trop longs, contenus inutiles
- Ce que disent les métriques → pages lourdes, scripts inutiles
- Ce que disent les équipes → solutions plus simples possible

Fiche mémo 10 réflexes d'éco-conception numérique



Définir des besoins utiles

Ne développer que ce qui répond à un besoin réel et validé.



Réduire le poids des pages et applications

Compresser images, vidéos et fichiers ; supprimer les ressources inutiles.



Limiter les dépendances

Éviter les frameworks et scripts tiers superflus, privilégier des solutions légères.



Simplifier les parcours

Réduire le nombre d'étapes et d'écrans nécessaires pour atteindre un objectif.



Optimiser la performance

Accélérer le chargement (cache, lazy loading, réduction des requêtes).



Concevoir pour tous

Intégrer accessibilité et compatibilité mobile dès la conception.



Allonger la durée de vie

Produire un code maintenable, documenté et évolutif ; éviter la dette technique.



Nettoyer régulièrement

Supprimer contenus obsolètes, environnements de test inutiles, logs trop anciens.



Mesurer pour progresser

Suivre des indicateurs simples (poids, temps de chargement, EcoIndex).



Faire de l'éco-conception un critère de qualité

En faire un réflexe partagé par toutes les parties prenantes du projet.

Acensi

peut vous aider en cas de besoin.
contact@acensi.fr