

EC09 – Guide de bonnes pratiques

Contexte général

Dans le cadre du projet **RebootCamp**, cette épreuve consiste à rédiger un **guide interne de qualité logicielle**, à destination de l'équipe de développement.

Ce guide doit formaliser un ensemble de règles et recommandations permettant de garantir un **code propre, maintenable, sécurisé, sobre et bien documenté**. Il doit également intégrer des **bonnes pratiques d'éco-conception web**, en cohérence avec la thématique du projet.

Il pourra être utilisé en début de sprint, lors de revues de code, ou comme socle d'onboarding des nouveaux membres.

L'objectif est de structurer des pratiques professionnelles réutilisables dans tous les projets collaboratifs.

Livrables attendus

Livrable	Détail
Guide de bonnes pratiques	Rédigé en markdown ou PDF, structuré, clair, exploitable par une équipe
Exemples de conventions	Normes de code, Git, structure de projet, checklist de review
Plan de test / couverture	Stratégie de tests avec exemples d'outils ou de mise en œuvre

Modalités d'évaluation

- **Type d'épreuve** : Mise en situation reconstituée écrite
- **Durée** : 4h
- **Nature** : Épreuve individuelle, sans oral
- **Critères évalués** :

- C8.6 : Conventions de code et documentation
- C9.6 : Stratégie de tests
- C14.3 : Qualité logicielle globale (approche mesurable et réaliste)

Recommandations

- Éviter les guides trop théoriques : viser le **concret et applicable**
- Structurer le document avec des **titres clairs, checklist et exemples**
- Citer les **outils recommandés** : linters, formateurs, frameworks de tests, intégration Git
- Intégrer des recommandations d'**éco-conception** : lazy loading, compression d'images, gestion des requêtes, dark mode, optimisation des builds CI/CD
- Fournir des **exemples pratiques de bonnes et mauvaises pratiques**

Exemples d'éléments attendus

- Conventions de nommage (`camelCase` , `snake_case` , `PascalCase`...) et organisation des dossiers
- Normes Git (`feat/` , `fix/` , `hotfix/` , `release/`) et politique de merge (PR, rebase)
- Checklist de revue de code : lisibilité, duplication, sécurité (XSS, injections), testabilité, performance
- Tableau des types de tests : unitaires, intégration, E2E, avec outils (Jest, PHPUnit, Cypress...)
- Exigences de documentation minimale : `README.md` , commentaires pour modules critiques, schémas d'architecture
- Bonnes pratiques d'éco-conception : suppression du code mort, optimisation des images, monitoring de l'impact (`EcoIndex`, `Lighthouse`)

Nom de dossier attendu

- une archive ZIP nommée `EC09_NomPrenom.zip` contenant :
 - Le guide au format `.md` ou `.pdf`
 - Un fichier d'exemples ou un sous-dossier `exemples/`
 - Éventuellement un `README.md` expliquant les choix du guide

Rappel pédagogique

Cette épreuve vise à valider la **capacité à structurer et formaliser les pratiques de qualité logicielle** :

- Clarification des règles de code et d'organisation d'un projet collaboratif
- Mise en œuvre de stratégies de **tests réalistes et mesurables**
- Intégration des **bonnes pratiques d'éco-conception** au cœur du projet
- Valorisation de la **documentation technique** comme outil de collaboration

Elle donne aux apprenants les outils pour **travailler efficacement en équipe** et **maintenir un haut niveau de qualité et de durabilité** dans des projets professionnels.