

# Grille de correction détaillée – EC09

## Informations générales

- **Épreuve** : EC09 – Guide de bonnes pratiques
- **Durée** : 4h – Individuelle
- **Projet fil rouge** : FoodSafe – Rédaction d'un guide interne de qualité logicielle
- **Nature** : Épreuve écrite (guide + exemples)

## Critères d'évaluation et attentes

Critère	Intitulé officiel	Attentes côté correcteur	Points de vigilance
<b>C8.6</b>	Conventions de code et documentation	Guide propose normes claires de nommage, structuration de projet, bonnes pratiques de documentation (README, commentaires, schémas).	Vérifier que les conventions sont concrètes et applicables (pas de théorie abstraite).
<b>C9.6</b>	Stratégie de tests	Guide définit une stratégie de tests complète (unitaires, intégration, E2E). Outils recommandés précisés (Jest, PHPUnit, Cypress, etc.).	Vérifier réalisme : couverture minimale précisée, outils cohérents avec projet.
<b>C14.3</b>	Qualité logicielle globale	Guide présente une approche pragmatique de la qualité (checklists, exemples, revues de code, gestion des secrets, conformité RGPD).	Vérifier que le guide intègre la sécurité des données sensibles (santé), RGPD et accessibilité.

# Livrables à corriger

- **Guide de bonnes pratiques** : au format `.md` ou `.pdf`
- **Exemples concrets** : fichiers de conventions, checklists, exemples de code/config
- **README explicatif** (optionnel) : motivations des choix

## Barème indicatif (sur 20 points)

Axe évalué	Points
Conventions de code et doc (C8.6)	/7
Stratégie de tests (C9.6)	/7
Qualité logicielle globale (C14.3)	/6

**Tolérance** :  $\pm 2$  points selon la pertinence des exemples, la clarté de la rédaction et l'adéquation au contexte FoodSafe.

## Points positifs attendus

- Conventions précises de nommage ( `camelCase` , `PascalCase` , schémas SQL)
- Normes Git claires (branches `feat/` , `fix/` , `hotfix/` , règles de PR et rebase/merge)
- Checklist de revue de code exploitable (lisibilité, duplications, performance, sécurité, RGPD)
- Stratégie de tests structurée avec outils adaptés au projet (tests unitaires sur moteur, E2E sur recherche produit)
- Exigences minimales de documentation ( `README.md` , schémas d'architecture, commentaires pour modules critiques)
- Intégration des spécificités FoodSafe : gestion des données sensibles, anonymisation, sécurité des API, accessibilité WCAG

## Erreurs fréquentes à surveiller

- Guide trop théorique ou générique, non exploitable en contexte projet
- Conventions absentes ou floues (ex. “bien nommer les variables” sans exemple)
- Stratégie de tests trop vague ou irréaliste (pas de couverture mesurable)
- Oubli de la dimension RGPD / données sensibles dans les recommandations
- Checklists incomplètes ou sans valeur pratique
- Document mal structuré (texte dense, pas de titres clairs, pas de checklists)

## Rappel pédagogique

Cette épreuve doit confirmer que l'apprenant sait :

- **Structurer et formaliser** des pratiques de qualité logicielle concrètes et applicables
- Définir des stratégies de **tests réalistes, mesurables et adaptées au contexte sensible** (santé, RGPD)
- Produire une documentation qui serve réellement à l'équipe (guide utilisable, checklists prêtes à l'emploi)
- Tenir compte des enjeux spécifiques de FoodSafe : **sécurité, conformité et accessibilité**

Elle vise à ancrer les bonnes pratiques professionnelles qui garantissent la qualité et la pérennité d'un projet collectif.