

# EC06 – Rattrapage : Mini-chaîne CI/CD éco-conçue

## Contexte général

Dans le cadre du projet **RebootCamp**, l'épreuve de rattrapage consiste à mettre en place une **mini-chaîne d'intégration et de déploiement continu**, appliquée à une version simplifiée de l'application.

L'objectif est de démontrer la compréhension des **principes fondamentaux du DevOps** : automatisation, versionning, conteneurisation et documentation — tout en adoptant une approche **sobre et éco-responsable**.

L'apprenant doit prouver sa capacité à construire un pipeline CI/CD minimal mais cohérent, capable de lancer les tests, de construire une image Docker, et de déployer ou simuler un déploiement.

## Livrables attendus

Livrable	Détail
Dépôt Git	Organisation claire des branches ( <code>main</code> , <code>feature/...</code> )
Pipeline CI/CD	Fichier de configuration ( <code>.yaml</code> ) exécutant au moins 3 étapes
Dockerfile	Image conteneurisée fonctionnelle et optimisée
Script de déploiement	Automatisation basique du déploiement ou simulation
README	Explication du processus CI/CD et justification des choix éco-conçus

## Modalités d'évaluation

- **Type d'épreuve** : Mise en situation reconstituée sur ordinateur
- **Durée** : 4h
- **Nature** : Épreuve individuelle, sans oral
- **Critères évalués** :
  - C7.3 : Structuration du dépôt Git
  - C9.4 : Mise en place d'un pipeline CI/CD
  - C9.5 : Conteneurisation et déploiement sécurisé

## Recommandations

- Utiliser un **outil CI/CD accessible** (GitHub Actions ou GitLab CI)
- Intégrer **au moins trois jobs** dans le pipeline :
  - Lint / vérification du code
  - Build / test
  - Déploiement (ou simulation)
- Construire un **Dockerfile multistage** ou une image légère (basée sur Alpine, par ex.)
- Sécuriser les secrets via le CI/CD (variables d'environnement)
- Documenter le pipeline et expliquer les **choix d'éco-conception** :
  - Jobs parallèles limités
  - Caches réutilisés
  - Builds conditionnels
- Produire un **README clair** expliquant les étapes et les optimisations

## Exemples d'éléments attendus

- `.github/workflows/ci.yml` ou `.gitlab-ci.yml` avec étapes `lint` → `build` → `deploy`
- Un **Dockerfile** générant une image < 200 Mo
- Un script `deploy.sh` déployant localement (simulation possible via logs)
- Gestion des secrets ( `.env.dist` , variables protégées dans le pipeline)
- Une **documentation du pipeline** détaillant les optimisations et la sobriété des builds

## Nom de dossier attendu

- Une archive ZIP nommée `EC06R_NomPrenom.zip` contenant :
  - Le dépôt du projet

- Le fichier CI/CD ( `.yaml` )
- Le `Dockerfile` et les scripts associés
- Un `README.md` documentant la CI/CD et les optimisations

## Rappel pédagogique

Cette épreuve de rattrapage vise à valider la **compréhension concrète du cycle CI/CD**, en insistant sur :

- La mise en place d'un **pipeline fonctionnel et clair**
- La maîtrise de la **conteneurisation** et du **déploiement automatisé**
- L'intégration des **principes d'éco-conception DevOps** :
  - optimisation des images Docker,
  - réduction du nombre de jobs et du temps d'exécution,
  - documentation claire des bonnes pratiques.

Elle prépare à des environnements de production modernes, où l'efficacité et la sobriété énergétique vont de pair avec la fiabilité logicielle.