

EC03 – Rattrapage : Back-end

“Progression et Badges”

Contexte général

Dans le cadre du projet **RebootCamp**, l'épreuve de rattrapage consiste à développer un **module back-end simplifié** dédié à la **gestion des progressions utilisateurs et des badges obtenus**, en utilisant deux bases de données : une **relationnelle (PostgreSQL)** et une **NoSQL (MongoDB)**.

L'objectif est de démontrer la capacité à manipuler ces deux types de stockage dans une logique cohérente : la base SQL gère les utilisateurs et leurs défis, tandis que la base NoSQL stocke les badges et l'historique de progression.

Le socle technique (Docker, authentification de base, environnement configuré) est fourni.

Livrables attendus

Livrable	Détail
Code source	Module en POO / MVC avec interaction SQL et NoSQL
Dump SQL et JSON	Exemples de données cohérentes pour PostgreSQL et MongoDB
Scripts de sauvegarde	<code>backup.sh</code> , <code>restore.sh</code> fonctionnels pour les deux bases
README	Explication des choix techniques, architecture et commandes utiles

Modalités d'évaluation

- **Type d'épreuve** : Mise en situation reconstituée sur ordinateur
- **Durée** : 4h

- **Nature** : Épreuve individuelle, sans oral
- **Critères évalués** :
 - C7.1 : Configuration de l'environnement de développement (Docker, terminal)
 - C8.1 : Programmation orientée objet, respect du pattern MVC
 - C9 : Sécurisation, optimisation du serveur
 - C10 : Utilisation d'une base relationnelle (PostgreSQL)
 - C11 : Utilisation d'une base NoSQL (MongoDB)
 - C12 : Justification des choix SQL/NoSQL
 - C13 : Mise en place d'un mécanisme de sauvegarde et restauration

Recommandations

- Définir une architecture claire (controllers , models , services)
- Utiliser des **migrations** pour PostgreSQL et des **scripts d'initialisation** pour MongoDB
- Sécuriser les accès via variables d'environnement (.env)
- Fournir un **dump cohérent** : utilisateurs, défis, badges
- Rédiger un **README précis** : explications techniques, choix, commandes principales
- Ajouter des **scripts de backup et restore** simples et fonctionnels

Exemples d'éléments attendus

- Une entité User et Challenge dans PostgreSQL
- Une collection Badges ou Progression dans MongoDB
- Une route /users/:id/progress combinant les deux sources
- Gestion d'un **score global calculé à partir des données SQL et NoSQL**
- Un docker-compose.yml complet (PostgreSQL, MongoDB, serveur)

Nom de dossier attendu

- Une archive ZIP nommée EC03R_NomPrenom.zip contenant :
 - Le dossier src avec le code source
 - Le fichier README.md
 - Les dumps /data/rebootcamp.sql et /data/mongo.json
 - Les scripts backup.sh et restore.sh

Rappel pédagogique

Cette épreuve de rattrapage permet d'évaluer la **maîtrise des bases du développement back-end moderne** :

- Structuration d'un projet MVC en POO
- Manipulation de **deux paradigmes de bases complémentaires** (SQL + NoSQL)
- Sécurisation et automatisation du travail avec Docker et scripts shell
- Capacité à **relier plusieurs sources de données dans une logique métier cohérente**

Elle conserve les objectifs fondamentaux de l'épreuve principale, dans un contexte plus ciblé autour de la **progression des utilisateurs et de la gamification écologique**.