

# Projet Développement Avancé

## Introduction

L'objectif est de vous accompagner dans la conception d'une API pilotée par les tests, intégrant les aspects CI/CD, avec **Node.js**, **MongoDB**, **JWT** côté serveur et **React** côté client. Pour cela nous réaliserons une application de géocaching. Le géocaching est une chasse au trésor moderne où les participants utilisent des appareils GPS ou des smartphones pour cacher et chercher des conteneurs appelés "géocaches" ou simplement "caches" à des emplacements spécifiques marqués par des coordonnées GPS.

Jeudi 20 février, chaque étudiant devra rendre son projet en indiquant un lien sur un dépôt github sur projet intitulé geocaching + un lien d'hébergement déposé dans ce tableau partagé :

[https://drive.google.com/drive/folders/1UvMRszVu\\_wNeCgemEO3tA09pyvJiAhPg?usp=sharing](https://drive.google.com/drive/folders/1UvMRszVu_wNeCgemEO3tA09pyvJiAhPg?usp=sharing)

Votre dépôt doit contenir :

- Le code source de la partie serveur dans le répertoire /serveur (incluant les tests, le script de création de la base de données et de déploiement).
- Le code source de la partie client dans le répertoire /client (incluant les tests)
- Une documentation de l'API REST du serveur au format OpenAPI dans le répertoire /serveur/doc, incluant une génération au format HTML. Regardez du côté de l'outil [Swagger](#) pour cela.
- Un fichier ReadMe.md contenant la liste des fonctionnalités implémentées.

Pensez à commenter et clarifier votre code.

Vendredi 21 février, chaque étudiant aura 10 min pour présenter son travail et réaliser une démo.

## Fonctionnalités

### Inscription - authentication des utilisateurs

Il doit être possible de s'inscrire sur le serveur. Les informations qui doivent être fournies en cas d'inscription sont :

- e-mail (unique dans la base)
- mot de passe

Pour s'authentifier il faut fournir son adresse mail et son mot de passe. Une authentication fournit un token (une clé) qui doit être conservée par le client et est fourni lors de la demande d'information. Une authentication est valable pour une durée maximale de 24h.

Un utilisateur peut se déconnecter. Pour mettre en place des tokens sécurisés, regardez du côté de [JWT](#).

## Ajouter/éditer/supprimer des geocaches

Un joueur peut ajouter une cache comprenant les attributs suivants :

- id (obligatoire)
- coordonnées GPS (obligatoire)
- créateur (obligatoire)
- difficulté (obligatoire)
- description (facultative)

Un joueur doit pouvoir aussi modifier les informations associées à une cache. Attention les joueurs ne peuvent agir que sur les caches dont ils sont propriétaires.

## Récupérer les geocaches :

Il doit être possible de récupérer les positions de toutes les caches. Par défaut seules les caches à proximité du joueur doivent être affichées.

Une fois qu'un joueur trouve ou non une géocache il doit avoir la possibilité de l'indiquer sur le site avec un commentaire.

Il faut faire ressortir les caches déjà trouvées par le client sur une carte. Regardez de du côté de [leaflet](#) pour l'affichage des cartes.

## Extensions :

Voici une liste des extensions que vous pouvez programmer après avoir réalisé les fonctionnalités de base. Chaque extension donnera lieu à des points bonus.

Affichez les classements

- des meilleurs clients,
- des caches les plus populaires et
- celles qui sont le plus rarement trouvées.

Offrez la possibilité d'ajouter des photos dans

- la description de la geo cache.
- le profil des joueurs (avatar : images représentant l'utilisateur)