

Projet PG219 2019

Introduction

Le projet doit être réalisé par trinômes. Il a pour but de développer une application de gestion de “todo”. Cette application se nommera **Remind**. Pour la réaliser, il sera nécessaire de mettre en oeuvre une architecture client serveur. L’application à développer comportera ainsi une partie serveur et un client mobile. La plate-forme de référence pour le développement client sera Android. Le code client doit être implémenté via la stack Cordova avec un framework SPA de votre choix, le code serveur via Node/Express/Mongo.

Le rendu final du projet est une archive nommée *pg219-2019-gN.zip* avec *N* étant le numéro de votre groupe. Cette archive doit contenir :

- Le code source de la partie serveur dans le répertoire src/serveur
- Le code source de la partie client dans le répertoire src/client
- Une documentation exhaustive de l’API REST du serveur au format OpenAPI dans le repertoire doc/serveur, incluant une génération au format HTML. Regardez du côté de l’outil [Swagger](#) pour cela.

Les fichiers sources doivent être correctement indentés, avec un nommage approprié des fonctions et variables. Un malus sera appliqué aux groupes ne respectant pas ces règles. Enfin faites particulièrement **attention à la copie**, nous utilisons un logiciel qui permet de détecter automatiquement les copier-coller, tout projet suspect sera étudié avec la plus grande attention...

Fonctionnalités

Inscription - authentification 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 authentification fournit un token (une clé) qui doit être conservée par le client et est fourni lors de la demande d’information. Une authentification 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 projets

Un utilisateur peut ajouter un projet. Un projet a vocation à contenir une collection de todo. Un projet dispose des informations suivantes :

suivantes:

- id (obligatoire)
- propriétaire (obligatoire)
- participants (obligatoire)
- titre (obligatoire)
- todos (obligatoire)

Un utilisateur doit pouvoir aussi modifier les informations associés à un projet. Finalement il doit aussi pouvoir supprimer un projet, après confirmation de l'utilisateur. Attention les utilisateurs ne peuvent agir que sur les projet dont ils sont propriétaires.

Ajouter/éditer/supprimer les todo

Un utilisateur peut ajouter un todo à un projet donné. Un todo contient les informations suivantes:

- id (obligatoire)
- propriétaire (obligatoire)
- titre (obligatoire)
- description (facultative)
- deadline (facultative)
- état (obligatoire, par défaut *en cours*)

La deadline est la date avant laquelle il faut accomplir le todo. L'état d'un todo peut être: *en cours* ou *effectué*.

Un utilisateur doit pouvoir aussi modifier les informations associés à un todo. Finalement il doit aussi pouvoir supprimer un todo. Attention les utilisateurs ne peuvent agir que sur les todo dont ils sont propriétaires. Par contre n'importe quel participant du projet peut changer l'état d'un todo.

Récupérer les projets et todo

Il doit être possible de récupérer les projets impliquant un utilisateur donné, ainsi que les todos associés. Par défaut seul les todo en cours doivent être affichés. Il faut faire ressortir les todos en retard.