**Vuejs 3 TodoList Santarelli**

Effectuer un CRUD de taches Santarelli complet sur deux projets : un backend en nodeJs (JavaScript) et un frontend en vue 3

Les taches ont une image, un titre, une description, un statut et un type de tache

Base de données : MongoDB

Il faut affilier un user à une tache

**Modele mongoDB :**

Task

Status\_Task

Type\_task

User

Autres si besoin

**Structure à respecter :**

Models

Controllers

Routes

Services

**Convention typographique** : Snake case, en anglais

Connexion login en Passport JWT et API\_KEY pour l’accès API

Utiliser des variables d’environnements avec les fichiers .env

Renvoie des statuts api en gardant les conventions

Routes dans les fichiers routes (ainsi que les restrictions d’accès)

Appel en BD dans les fichiers services,

Pour l’enregistrement d’image, utiliser Gridsfstorage

Structure à respecter coté Backend :

Folder :

* App
  + Api
    - Config
      * Db.config.js
      * Grid\_sf\_storage.js
      * Passport.js
    - Controllers
    - Models
    - Routes
    - Services
    - Utils
      * Mails
* .env
* Node\_modules
* .gitignore
* Package.json
* Server.js

Quand une tache est créée, envoyer un email au user en charge

Quand le statut d’une tache est modifié, envoyé un email au user en charge

Quand une tache est terminée, envoyer un email au user en charge

FrontEnd :

Struture a respecter :

* Public
  + Src
    - Assets(image, icon, etc)
    - Compoments
    - Css
    - Router
    - Store
      * Requests
    - Views
    - App.view
    - Main.js
    - .env

Le design est libre.

Il faut un minimum d’une navbar avec connexion déconnexion.

Un accès seulement en lecture seule si on n’est pas connecté, et un accès lecture écriture si on est connecté.

Il y a seulement le créateur de la tâche, et la personne en charge de la tache qui peut modifier ou supprimer une tache

Il faut donc un formulaire de création d’une tache, une vue de listing de toutes les taches, avec un filtre sur toutes les colonnes, ainsi qu’un sort sur toutes les colonnes.

Une vue d’une tache, qui peut être modifiée ou supprimée (si c’est le bon user)

En css : responsive

Le code doit être commit et push régulièrement sur git via gitkraken