

Test myConnectedCompany

Introduction

Le but de ce test sera d'évaluer votre compétence d'apprentissage, d'autonomie et votre logique.

Vous pouvez me demander de l'aide, comme si j'étais votre collègue. Je suis disponible par email et sur Discord (Voir Ressource).

Pour cela vous devez créer une api REST en suivant le backlog fournis, le langage utilisé n'est pas important, cependant, si vous voulez utiliser le starter-kit, vous devez utiliser du Javascript. L'utilisation de git avec une solution en ligne est obligatoire dans votre projet et faire des commits construits et explicites (avec un message de commit expliquant ce que vous avez fait dans celui-ci), vous m'enverrez le lien du repository en ligne en tant que rendu.

Avant d'envoyer votre rendu, vous pouvez y rajouter une note de debriefing sur le projet.(ressentie, difficulté, compréhension)

P-S: Ne cherchez pas à faire compliqué et n'utilisé pas quelque chose que vous ne comprendrez pas.

Backlog

Projet : MyTaskBoard (Trello like back-end only)

STORY 0 :

EN TANT QUE développeur

JE VEUX produire du code propre et le mettre sur git

AFIN DE qu'il soit compréhensible par d'autres développeurs. Le mettre sur un repository en ligne, type: Github, Gitlab, Bitbucket ou autres.

Si vous n'utilisez pas le starter-kit:

STORY 1 :

EN TANT QUE développeur

JE VEUX initialiser le projet

AFIN D' avoir une base pour commencer

DoD .

Si vous utilisez le starter-kit :

STORY 1 :

EN TANT QUE développeur

JE VEUX comprendre l'architecture du projet

AFIN DE me familiariser avec le projet

DoD .

STORY 2 :

EN TANT QUE développeur

JE VEUX pouvoir requêter un élément sur la route /cards

AFIN D' avoir mon premier controller

DoD Vous pouvez récupérer un tableau vide par le biais d'une méthode GET.

STORY 3 :

EN TANT QUE développeur

JE VEUX avoir le CRUD d'une carte

AFIN D' avoir une base pour les cartes

DoD Récupérer une carte par rapport à un ID.

Pouvoir récupérer toutes les cartes.

Créer une carte et générer un ID.

Pouvoir mettre à jour une carte.

Pouvoir supprimer une carte par rapport à un ID.

Model d'une carte :

- Nom de la carte
- Membres (ne sera pas utiliser tout de suite)
- Description
- Etiquettes
- Checklist
- Durée d'une tâche ou Date de fin de la tâche (Deadline / milestone)
- Date et Temps de la mise à jour de la carte
- Date et Temps de création de la carte

STORY 4 :

EN TANT QUE développeur

JE VEUX avoir le CRUD d'une liste

AFIN D' y mettre les cartes

DoD Faire le CRUD des listes.

Model d'une liste :

- Nom de la liste
- Tableau qui contiendra des cartes
- Date et Temps de création de la liste

STORY 5 :

EN TANT QUE développeur

JE VEUX créer une carte dans une liste

AFIN DE pouvoir trier mes tâches

DoD Une liste contient des cartes.

STORY 6 :

EN TANT QUE développeur

JE VEUX pouvoir mettre à jour une carte

AFIN DE pouvoir la changer de liste

DoD Vous devez récupérer une carte par le biais de son ID ou de son nom, puis transférer l'objet de la carte dans une nouvelle liste.

STORY 7 :

EN TANT QUE développeur

JE VEUX pouvoir créer un tableau

AFIN DE créer un tableau par projet.

DoD Vous devez créer un tableau de projet qui contiendra des listes de tâches. Un tableau de projet contient une liste de tags qui pourront être utilisés par les tâches.

Modèle de Tableau :

- Nom du tableau
- Tableau qui contient des listes
- Date et Temps de la mise à jour du tableau
- Date et Temps de création du tableau
- Tableau d'étiquettes

STORY 8 :

EN TANT QUE développeur

JE VEUX ajouter des utilisateurs

AFIN D' attribuer des tâches aux utilisateurs.

DoD Créer le CRUD utilisateur,

Modèle d'utilisateur :

- Pseudo
- Email
- Date et Temps de l'utilisateur
- Tableau dans lequel je suis membre ou propriétaire (Attention, ne pas faire la feature en rapport tout de suite, story 9)

STORY 9 :

EN TANT QUE développeur

JE VEUX qu'un tableau ait un propriétaire et qu'il puisse ajouter des utilisateurs

AFIN D' avoir une équipe et partager mon projet.

DoD Refactoriser la création des tableaux pour qu'on puisse y ajoute un utilisateur. L'ID du tableau sera ajouté dans le tableau d'appartenance de l'utilisateur. Un propriétaire peut ajouter un ou des membres dans son tableau.

STORY 10 (Bonus) :

EN TANT QUE développeur

JE VEUX créer un log pour chaque activité

AFIN DE suivre son avancement et les modifications.

DoD Créer un service qui va générer les logs par rapport aux tableaux, aux tâches et aux utilisateurs.

STORY 11 (Bonus) :

EN TANT QUE développeur

JE VEUX mettre en place une BDD

AFIN D' enregistrer des données et de les rendre persistante

DoD Refactoriser le code pour le rendre compatible à la mise en base de données.

STORY 12 (Bonus) :

EN TANT QUE développeur

JE VEUX avoir un système d'authentification

AFIN DE sécuriser mon API

DoD Ajouter l'enregistrement de compte avec un chemin register, avec un mot de passe, qui sera hasché (chiffré) dans la base de données.

Un chemin pour le login qui va comparer le mot de passe donner par l'utilisateur avec celui de la BDD et lui retourner un JWT afin de pouvoir utiliser les différentes routes qui seront sécurisées.

L'ajout d'un middleware vous permettant de verifier le JWT à l'appel d'une route.

Ressources

- Contact:
 - Discord : Sauronox#0117
 - Email : alan.drieux@myconnectedcompany.com
- Apprendre NodeJS / JS:

- <https://grafikart.fr/formations/nodejs>
- <https://openclassrooms.com/fr/courses/6390246-passez-au-full-stack-avec-node-js-express-et-mongodb/6466231-demarrez-votre-serveur-node>
- <https://openclassrooms.com/fr/courses/6175841-apprenez-a-programmer-avec-javascript/6278450-enregistrez-vos-donnees-avec-des-types-de-donnees>
- https://www.youtube.com/watch?v=s_VMwHFSjXY&list=PLuWyq_E05_AI83Z2JdSPdJ-81QPvI3cxC

- [Apprendre git](#)
- [Documentation ExpressJS](#)
- [Documentation DB ORM sequelize](#)
- [Tutoriel utiliser mysql avec nodejs](#)
- Docker :
 - [Presentation de Docker by Grafikart](#)
 - <https://openclassrooms.com/fr/courses/2035766-optimisez-votre-deploiement-en-creant-des-conteneurs-avec-docker>
 - <https://grafikart.fr/tutoriels/docker-stack-web-635>

Si vous ne trouvez pas votre bonheur, google vous aidera.

Starter-kit

Le starter kit est fait en JavaScript avec le technologie Nodejs et le micro framework Express. En plus un docker-compose est disponible pour la partie bonus du backlog.

Prérequis

- [NodeJS](#) version : 16.x.x(Direct Download)
- [yarn](#) version : 1

Pour lancer le projet

Pour windows ▢

Nodejs version Manager (Optionel) : - <https://github.com/coreybutler/nvm-windows#node-version-manager-nvm-for-windows>

Récupérer le projet et l'installer (être dans le même repertoire que le dossier exo) `bash cd exo/starter-kit yarn yarn start`

Nodejs version Manager (Optionel) : `bash yarn global add n`

Récupérer le projet et l'installer (être dans le même repertoire que le dossier exo) `bash cd exo/starter-kit yarn yarn start`

Lancer le docker-compose

- Prérequis avoir docker engine ou desktop:
 - [windows](#)
 - [macos](#)
 - [linux docker-compose](#)

Dans votre console de commande :

```
cd exo/starter-kit
# en premier plan
docker-compose up
# en arriere plan
docker-compose up -d
# voir les logs
docker-compose logs
```