

Conception d'une API REST pour Blizzard



Ca y est, c'est l'apothéose, Blizzard, l'entreprise qui a créé l'univers Warcraft (entre autre) fait appel à vous pour travailler sur une API à destination des joueurs.

Qu'ils soient sur l'appli via leur smartphone, sur le site via leur ordinateur ou qu'il s'agisse d'un logiciel tiers, tous pourront communiquer avec votre API qui centralisera les données.

Vous êtes placé sur les ordres d'un chef de projet. Celui-ci étant très pressé (il faut le comprendre, LE match commence dans 30 minutes et il doit prendre la route), il vous donne des indications que vous notez sur un bouton de papier en même temps que vous marchez vers le parking.

Ce qu'il m'a dit

"Ton API devra permettre de stocker des personnages pour World Of Warcraft, tu sais, le jeu.

On veut juste pouvoir ajouter un perso avec son pseudo, sa classe et son niveau.

Évidemment on peut récupérer toutes les infos d'un perso, le modifier, le supprimer, juste en donnant son pseudo et sa classe, en GET ou en POST selon le cas.

Cela signifie donc que deux persos ne peuvent pas avoir le même pseudo / classe.

On veut aussi pouvoir récupérer tous les persos d'un compte en passant l'ID de celui-ci.

Pour récupérer les infos d'un perso ou tous les persos d'un compte, tout le monde peut le faire. Par contre, faut que ce soit seulement celui qui a créé un perso qui puisse faire la modification et la suppression,... et l'admin, évidemment, lui, il peut tout faire même si c'est pas son perso.

Un admin ne peut pas créer de perso, juste modifier et supprimer.

Alors pour l'admin, c'est simple, dans toutes ses requêtes, il y aura un token qui contiendra une clé appelée "adminToken". L'admin aura juste à se logger sur ton API (email / password). On lui renverra son token avec un expire de 1an.

Pour les joueurs, c'est simple, pour ajouter leur perso :

- soit ils ont déjà créé un perso et ils ont donc un token à te transmettre
- soit ils vont t'envoyer leur username et leur password en même temps que les infos du perso. Toi tu vas call l'api blizzard (j'tenvoie les logs par mail sur la route) pour voir si leur compte est bon. S'il est bon, l'api blizzard te renverra un token avec l'ID du compte du joueur, comme ça tu pourras créer leur perso et leur fournir un token avec leur ID de compte.

Pour les prochaines requêtes, ils n'auront plus leur username/password à te transmettre mais juste leur token qui contient leur ID de compte !

Pour les requêtes de modification et de suppression, ils auront évidemment déjà un token.

Il faudrait aussi me logger dans un fichier quand quelqu'un fait une requête sur ton API avec un mauvais username / password pour son compte blizzard.

Tu écriras juste une ligne comme ça :

2023-01-01 17:00:05 : tentative de connexion invalide

Bien sûr, tu me feras une jolie documentation avec Swagger.

Ton API fera juste des retours textuels pour l'instant quand elle créera un perso, le modifiera, le supprimera.

Bon courage, je sais que t'es le meilleur (ou la meilleure, je n'ai jamais su si tu étais une fille ou un garçon) !

”

L'email du chef de projet

Re,

voici la documentation : <https://backend-tp-final-nodejs.agence-pixi.fr/api-docs/>

Notre clé secrète pour décoder le token renvoyée est la suivante :

SR1wKQYqITLVWZSIYkot3xTu0qdZuWDn

Je te conseille d'utiliser AXIOS pour faire ton call API, tu trouveras de la doc simple sur internet.

Deux comptes pour tester :

Eluvie / xPdd^5a9Jn

Ärth / nj564qsda;sdz

Les classes de wow si tu ne les connais pas : guerrier, paladin, chasseur, voleur, prêtre, chaman, mage, démoniste, moine, druide, chasseur de démons, chevalier de la mort ou évocateur.

En création de joueur, si quelqu'un envoie autre chose que ça, on lui retourne une erreur.

Allez, bon match !

L'évaluation du TP FINAL

Tout est décrit dans le brief de votre chef de projet, il n'a rien oublié, alors décortiquez le bien :-)

Vous êtes attendu sur votre capacité à créer et documenter une API sans que l'on vous liste les endpoints d'une manière précise et scolaire, et donc à savoir interpréter un brief client.

Votre API sera testée sur chaque route.

Votre API sera testée en envoyant un token adminToken (une fake String) qui devra me permettre de modifier / supprimer n'importe quel personnage.

Votre API sera testée en simulant un joueur qui voudrait supprimer un autre joueur.

Votre API sera testée sur toutes les fonctionnalités supplémentaires demandées.

Question pour 1 point supplémentaire

Votre API a une faille.

Il est possible d'insérer une valeur qui n'est pas autorisée pour une donnée en base.

Quelle donnée et comment peut-on le faire ?

Modalités de rendu

Veuillez déposer votre code dans un repository Github et m'envoyer le lien sur Teams.

Votre repository doit être public.

Le dernier commit doit avoir été fait au plus tard à 16h50.

Tout commit réalisé après cet horaire entraînera une perte de point (-2 points / heure de retard).

Plagiat

En cas de plagiat avéré d'un projet à l'autre (cela se voit assez vite), des pénalités conséquentes de points seront appliquées. N'envoyez pas votre code à quelqu'un d'autre :)