

TP3 - PARTIE 1

420-W0V-SW - Développement d'applications Web transactionnelles

Pondération de toutes les parties : 17.5%

Travail à effectuer

Un magicien est entré en contact avec votre enseignant pour l'aider à utiliser une faille dans la dimension arcanique. Il prétend qu'une application web Restful API est l'outil dont il a besoin pour prendre le contrôle de cette faille. Cet outil permettra d'améliorer l'accès à la magie dans son monde. Nous allons utiliser tout ce que nous avons appris dans le cours pour essayer de l'aider.

Attentes

Pour ce travail, vous devez respecter les principes de l'architecture MVC. Puisque cette application sera remise au magicien à la fin du contrat(date de remise), une attention particulière devra être accordée à la qualité du code que vous allez produire.

Le magicien a spécifié que la meilleure façon pour créer des objets dans son monde était de créer des documents dans une base de données MongoDB. Votre application doit respecter cette condition.

Il est impératif d'implémenter une solution qui facilite la traduction des éléments lisible par les utilisateurs de l'application. Nous ne pouvons garantir si l'utilisateur parle anglais, français ou l'elfique. Il faut donc garder ce principe en tête lorsque vous allez utiliser des champs String.

Partie 1 : Programme de base

Classe Magicien

nom	type	description
id	String	identifiant du magicien. Il fait référence au _id de la base de données.
name	String	Le nom du magicien
appearance	Object	Objet javascript permettant à notre magicien de décrire son apparence. ex {hair: "long", hairColor: "green", robe: ...}
stats	Object	Objet javascript permettant à notre magicien d'avoir des caractéristiques ex: {strength:10, ...}
level	Number	Le niveau du magicien
schools	Array of String	Liste des écoles de magie dont le magicien est capable de lancer un sort. ex: ["illusion", "divination", ...]
alignment	String	Alignement du magicien. ex: "Chaotique bon" ou "Chaotic good"
spellbooks	Array of Grimoire	Liste des grimoires que le magicien peut utiliser.

Classe Grimoire

nom	type	description
id	String	identifiant du grimoire. Il fait référence au _id de la base de données.
name	String	Le nom du grimoire
schools	Array of String	Liste des écoles de magie disponible dans le grimoire
spells	Array of Spell	Liste des sorts du grimoire.
owner	Magicien	Propriétaire et créateur du grimoire. Peut être null.

Classe Sort

nom	type	description
id	String	identifiant du Sort. Il fait référence au _id de la base de données.
name	String	Le nom du sort
level	Number	Niveau du sort
school	String	École de magie du sort
effects	Array of Effect	Liste des effets du sort.

Classe Effet

nom	type	description
id	String	identifiant du Sort. Il fait référence au _id de la base de données.
description	String	Description de l'effet du sort.
school	String	École de magie de l'effet.
types	Array of String	Un effet peut avoir plusieurs types : "bon", "mauvais", "chaotique", etc. Dépend de l'effet. Ex : Un effet qui fait perdre les cheveux du lanceur du sort peut être considéré "mauvais" et/ou "chaotique". Un effet qui redonne des points de vie est considéré comme "bon".

Actions

Un Magicien peut:	
	Créer un sort (Aléatoire et/ou fait à la main)
	Créer un grimoire (Aléatoire et/ou fait à la main)
	Ajouter un sort à un grimoire
	Acquérir un grimoire existant
	Lancer un sort d'un de ses grimoires

Créer un sort

Le sort peut être généré de façon aléatoire. Si un élément est spécifié par le magicien, alors cet élément ne sera pas généré de façon aléatoire. Lorsqu'un magicien crée un sort, le niveau du sort ne peut pas être plus élevé que le niveau du magicien. L'école du sort doit se trouver dans les écoles du magicien. La quantité d'effets dépend du niveau du sort. Les effets secondaires d'un sort ne sont pas obligés d'être de la même école que le sort.

ex : {name: "Create water", school: "abjuration", level: 1, effects: [{d: "Create water", s: "abjuration", a: "create_water"}]}

Le sort créé n'aura aucun élément généré aléatoirement.

ex2: {name: "RandyRando spellosorum"}

le sort créé possède un niveau inférieur ou égal au magicien, une école parmi celles du magicien et un ou plusieurs effets aléatoires.

Créer un grimoire

Le grimoire peut être généré de façon aléatoire. Si un élément est spécifié par le magicien, alors cet élément ne sera pas généré de façon aléatoire. Lorsqu'un magicien crée un grimoire, les écoles de magie sont prises parmi les écoles de magie du magicien. La quantité de sorts du grimoire devrait dépendre du niveau du magicien.

Ajouter un sort à un grimoire

Un magicien peut ajouter un sort à un grimoire dont il est le propriétaire. Si l'école du sort ne figure pas parmi la liste des écoles du grimoire, la liste des écoles de magie du grimoire devra être mise à jour.

Acquérir un grimoire existant

Un magicien peut ajouter le grimoire d'un autre magicien dans sa liste de grimoires.

Lancer un sort d'un de ses grimoires

Un magicien peut lancer un sort qui se trouve dans l'un de ses grimoires. Pour lancer le sort, le sort doit appartenir à l'une des écoles de magie du magicien. Le magicien doit avoir un niveau égal ou supérieur au sort pour le lancer.

Logger

Pour la première partie du travail, vous devez ajouter un logger dans votre application. Vous devrez utiliser ce logger sur chacune des actions qu'un magicien peut poser.

Vous devez utiliser ce logger quand vous attrapez une erreur.