

TP Technologies du web

Application à l'aide d'une API utilisant Spring Boot

Barbary Théo
Application Dofus Quest

Table des matières

Contexte	3
Introduction	3
Les différentes ressources et technologies utilisées	4
Le framework javascript	4
Les technologies imposées	4
Les autres ressources	4
Fonctionnalité et fonctionnement de l'application	5
Le sujet de l'application	5
Les fonctionnalités de l'application	5
Se connecter/Se créer un compte	5
Voir les quêtes que vous avez déjà réalisé	6
Voir toutes les quêtes et changer leur statu	6
Voir tous les succès	7
Option de recherche	8
Le fonctionnement de l'application	8
Lancement de l'application	8
Quelques explications sur le fonctionnement du code	9
La base de données	10
Les difficultés rencontrées et mon ressenti	11

Contexte

Le but de ce projet était de réaliser un site web utilisant différentes technologies qui nous étaient imposées. À savoir, une API basée sur les services REST grâce à Jersey et utilisant le framework Spring Boot, un framework javascript que nous avons la possibilité de choisir, et enfin une base de données utilisant hsqldb. De plus, le model choisit doit contenir au minimum 2 entités se faisant références.

Introduction

Ce rapport va vous expliquer dans un premier temps les différentes ressources et technologies qui ont été utilisées à la réalisation de ce projet, notamment le framework javascript choisi, et la raison de ce choix. Puis expliquer le sujet de l'application, les possibilités et les fonctionnalités qui ont été intégrées tout en vous montrant son fonctionnement. Ensuite, mettre au clair le model et la base de données, c'est-à-dire expliquer le cheminement qui a été utilisé pour trouver le schéma intégré dans l'application. Enfin, faire un point rapide sur les difficultés que j'ai rencontré durant la réalisation du projet.

Les différentes ressources et technologies utilisées

Le framework javascript

Comme dit dans l'introduction, nous avons la possibilité de choisir le framework javascript que nous allons utiliser pour la réalisation de l'application, par exemple jQuery, Angular, React, Vue.js, etc. Dans mon cas, jQuery aurait été un choix logique, vu que j'en ai déjà fait lors de la réalisation de projets personnels mais aussi, et surtout, c'est le framework que j'utilise en entreprise. Cependant, ce projet me donnait la possibilité d'en apprendre un nouveau, c'est pourquoi je me suis servi de React. J'ai choisi ce framework puisqu'en ce moment il est en pleine expansion, et de ce fait beaucoup de documentations m'étaient disponibles.

Les technologies imposées

Je vais très rapidement revenir sur ce point, même si je l'ai déjà introduit dans le contexte, pour détailler un peu plus ce qui était demandé. Comme dit précédemment donc, nous devons réaliser notre base de données à l'aide de hsqldb, et ainsi réaliser notre model grâce à Hibernate. Une application autonome et une API avec Spring Boot, les services REST avec Jersey, qui sera donc en lien avec mon framework javascript, permettant de faire du CRUD. Nous avons aussi la possibilité de créer deux projets différents, le premier correspondant à l'API et le deuxième qui est le front, cela permettrait une meilleure sécurité de l'application et des données. Cependant, cela n'était pas obligatoire et j'ai donc créé qu'un seul projet pour avoir un unique répertoire.

Les autres ressources

L'API a été réalisé avec l'outil de gestion Maven. Encore une fois, j'ai décidé d'utiliser ce dernier et non Gradle que j'avais déjà utilisé dans d'autres projets.

Fonctionnalité et fonctionnement de l'application

Le sujet de l'application

Cette dernière est une application pour la gestion de ses quêtes sur le jeu Dofus. Toutes les quêtes du jeu sont sauvegardées dans la base de données et des succès sont caractérisés par plusieurs quêtes.

Les fonctionnalités de l'application

Vous avez différentes fonctionnalités avec cette application, je vais vous les énumérer et vous les expliquer avec des illustrations :

Se connecter/Se créer un compte



The screenshot shows the user interface of the 'Dofus Quest' application. The top section has a green background on the left and a tan background on the right. On the green background, there is a yellow egg icon with a dollar sign, the text 'Bienvenue sur Dofus Quest', 'Organisation, Facilité, Fiabilité', and a paragraph of text explaining the site's purpose. On the tan background, there is the 'Dofus' logo and a cartoon character wearing a yellow hat with a blue '金' (gold) character. Below the green section, there are two brown boxes. The left box is titled 'Se connecter' and contains two input fields labeled 'Pseudo' and 'Mot de passe', and a 'Se connecter' button. The right box is titled 'S'inscrire' and contains two input fields labeled 'Pseudo' and 'Mot de passe', and an 'S'inscrire' button.

Bienvenue sur Dofus Quest
Organisation, Facilité, Fiabilité
Ce site vous offrira la possibilité d'avoir un suivi sur vos quêtes.
Vous avez simplement à vous connecter sur votre personnage, pour avoir accès à votre gestionnaire.
Vous pourrez aussi connaître toutes les quêtes du jeu qui peuvent être triées par niveau, localisation, etc..
Bon jeu !

Se connecter

Pseudo Pseudo
Mot de passe Mot de passe
Se connecter

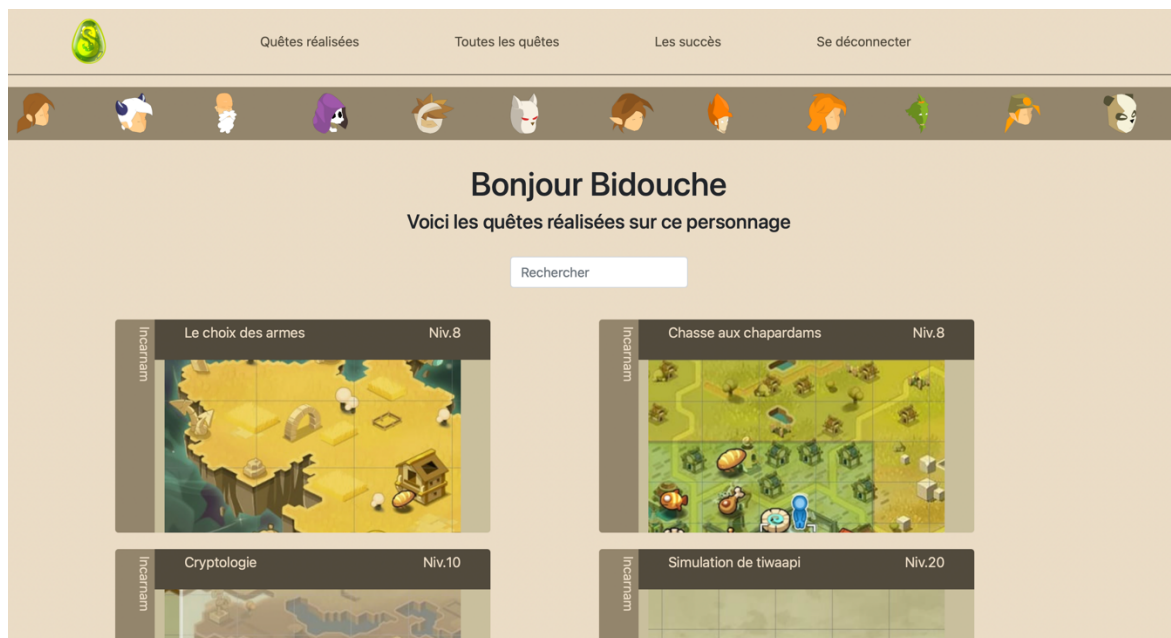
S'inscrire

Pseudo Pseudo
Mot de passe Mot de passe
S'inscrire

Vous voilà sur la page de connexion, ici vous pourrez vous connecter ou vous créer un compte qui caractérisera votre avancée sur un personnage. Lors du premier lancement de l'application, la base de données est directement initialisée avec un jeu de données, de ce fait vous pourrez vous connecter avec, par exemple, les identifiants suivants :

- Id : Bidouche
- Mdp : admin

Voir les quêtes que vous avez déjà réaliséé



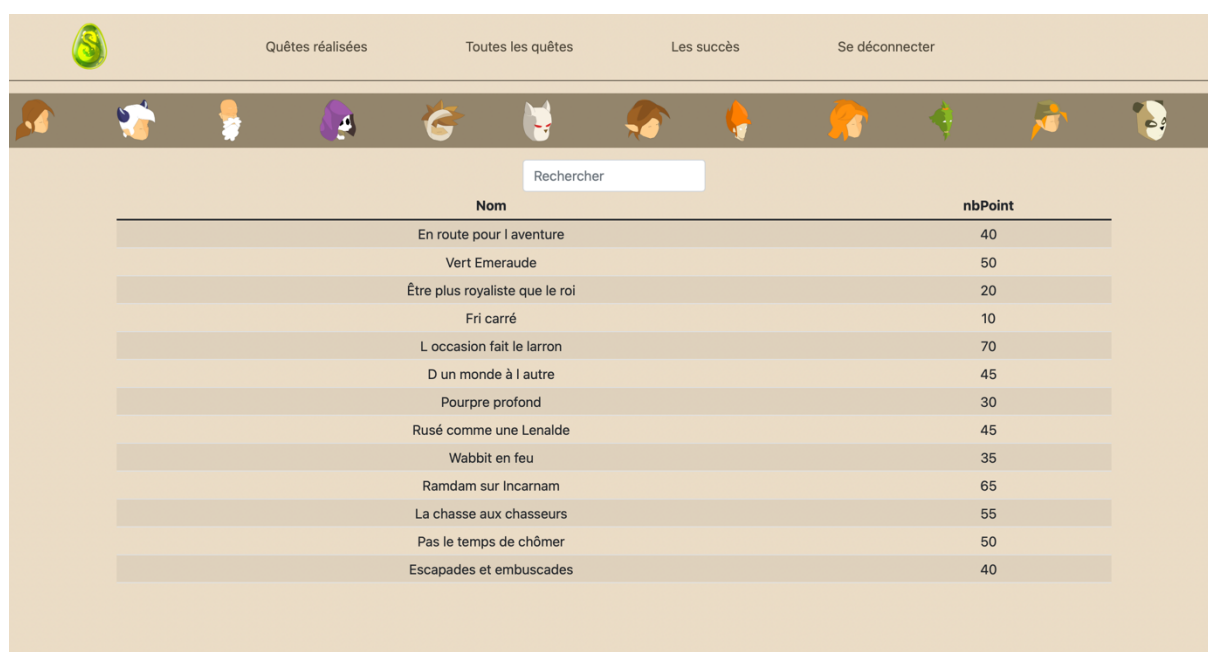
Une fois connecté ou vous êtes inscrit, vous arriver sur cette page. Ici vous pouvez voir les quêtes que vous avez déjà indiqué comme réalisées.

Voir toutes les quêtes et changer leur statu

	Nom	Lieu	Niveau	Succès
Indiquer faite	Transport peu commun	Incarnam	3	La chasse aux chasseurs
Indiquer faite	Des vestiges de legende	Incarnam	5	Wabbit en feu
Indiquer faite	Vu du ciel	Incarnam	6	En route pour l aventure
Indiquer faite	Mise a lepreuve	Incarnam	3	Ramdam sur Incarnam
Annuler	Champs de bataille	Incarnam	5	Fri carré
Annuler	Coups d epee dans l eau	Incarnam	6	L occasion fait le larron
Annuler	Decime-moi des bouftous	Incarnam	7	Vert Emeraude
Indiquer faite	Chasse aux chapardams	Incarnam	8	Pourpre profond
Annuler	Lecon d humilite	Incarnam	8	Pas le temps de chômer
Annuler	Des chafers qui marchent	Incarnam	9	D un monde à l autre
	La bache et la pioche	Incarnam	6	Ramdam sur Incarnam

Dans la barre de navigation, une fois cliqué sur « Toutes les quêtes », vous arrivez sur cette page. Sur cette dernière vous avez la possibilité de voir toutes les quêtes du jeu, leurs informations, etc. De plus, vous pouvez indiquer ces quêtes comme réalisées, ou annuler si finalement elles n'ont pas encore été faites.

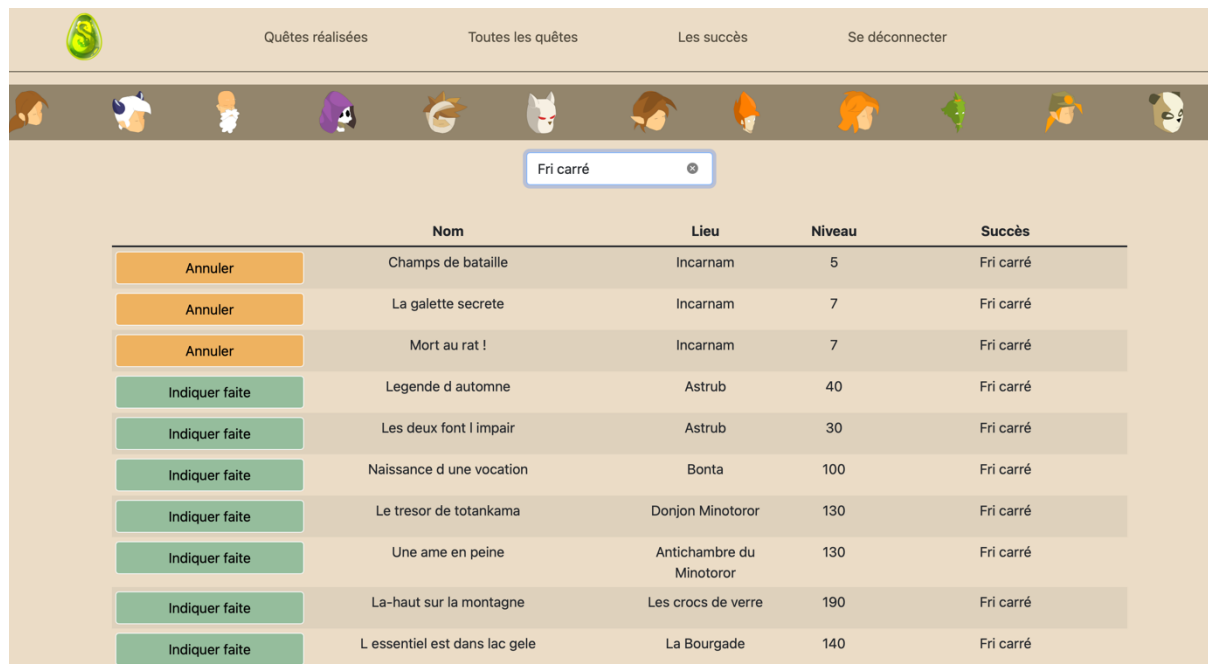
[Voir tous les succès](#)



Nom	nbPoint
En route pour l'aventure	40
Vert Emeraude	50
Être plus royaliste que le roi	20
Fri carré	10
L'occasion fait le larron	70
D'un monde à l'autre	45
Pourpre profond	30
Rusé comme une Lenalde	45
Wabbit en feu	35
Ramdam sur Incarnam	65
La chasse aux chasseurs	55
Pas le temps de chômer	50
Escapades et embuscades	40

Comme dit un peu plus tôt, un succès est défini par plusieurs quêtes. Cette page s'affiche une fois avoir cliqué sur « Les succès » dans la barre de navigation. Ici tous les succès sont affichés (malheureusement je n'avais pas accès à tous les succès du jeu).

Option de recherche



The screenshot shows the application interface with a search bar containing 'Fri carré'. Below the search bar is a table of quests. The table has five columns: Nom, Lieu, Niveau, and Succès. Each row represents a quest and includes an action button (Annuler or Indiquer faite).

	Nom	Lieu	Niveau	Succès
Annuler	Champs de bataille	Incarnam	5	Fri carré
Annuler	La galette secrete	Incarnam	7	Fri carré
Annuler	Mort au rat !	Incarnam	7	Fri carré
Indiquer faite	Legende d automne	Astrub	40	Fri carré
Indiquer faite	Les deux font l impair	Astrub	30	Fri carré
Indiquer faite	Naissance d une vocation	Bonta	100	Fri carré
Indiquer faite	Le tresor de totankama	Donjon Minotoror	130	Fri carré
Indiquer faite	Une ame en peine	Antichambre du Minotoror	130	Fri carré
Indiquer faite	La-haut sur la montagne	Les crocs de verre	190	Fri carré
Indiquer faite	L essentiel est dans lac gele	La Bourgade	140	Fri carré

Sur chaque page il y a la possibilité de rechercher les éléments par leurs informations, à savoir leur niveau, leur nom, leur localisation, etc.

Le fonctionnement de l'application

Lancement de l'application

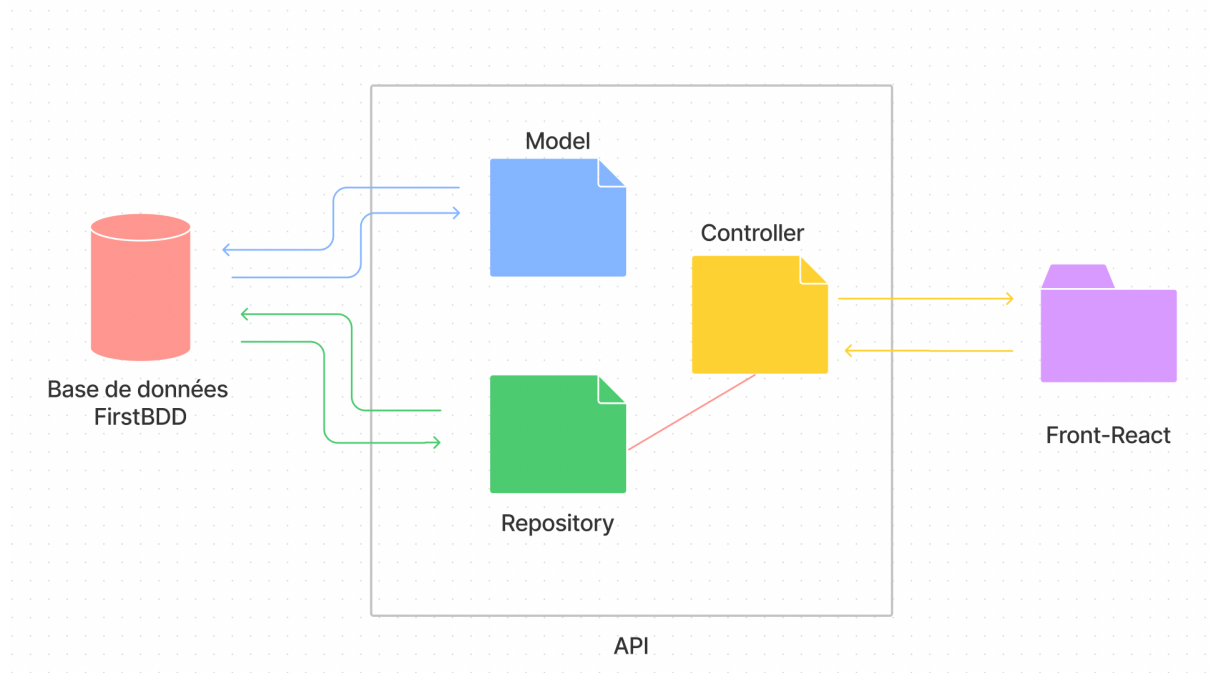
Pour lancer l'application différentes étapes sont demandées :

1. Au niveau de la base de données :
 - Créer la BDD (name : FirstBDD)
 - Se rendre dans le dossier lib du dossier hsql de base
 - Lancer la commande : `java -cp hsqldb.jar org.hsqldb.server.Server --database.0 file:mydb --dbname.0 FirstBDD`
2. Au niveau de l'API :
 - Exécuter le fichier DofusQuestProjectApplication dans le dossier `src/main/java/com.example.Dofus_Quest_Project`
3. Au niveau du front en React
 - Ouvrir un terminal dans le dossier front-react du projet
 - Lancer l'application avec la commande : `npm start`

Une fois cela réalisé vous pouvez vous rendre sur la page de connexion de l'application en indiquant « localhost:3000 » dans l'URL dans navigateur.

Quelques explications sur le fonctionnement du code

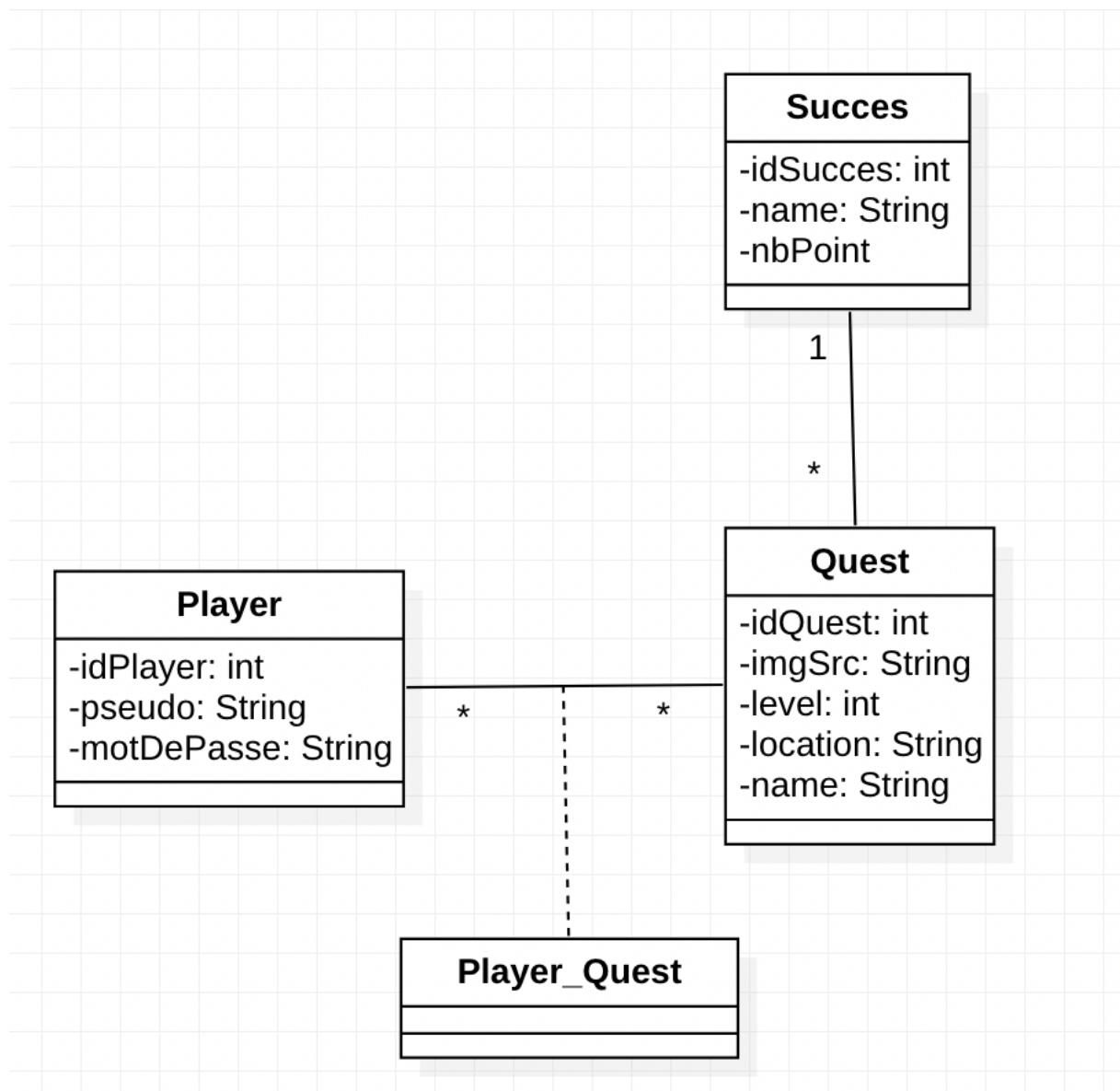
Voici une petite illustration du fonctionnement de l'application :



Lors du lancement de l'application, le model interagit avec la base de données pour la création de cette dernière avec la génération d'Hibernate. La base de données communique avec le model lors de la récupération des données pour directement les convertir en classe présente de ce modèle. Les repositories, liés au CRUD, communiquent avec la BDD pour les requêtes « SQL » et la récupération des données présentes dans cette base. Ces mêmes repositories sont utilisés et présents dans les contrôleurs. Les contrôleurs interagissent avec le front pour lui envoyer les données qu'ils auront récupéré grâce aux repositories. Le front interagira avec l'utilisateur directement, et pourra communiquer avec les contrôleurs grâce aux requêtes AJAX.

La base de données

La première question pour définir le modèle était « qu'est-ce que je veux proposer et afficher à l'utilisateur ? ». Ainsi j'ai dans un premier défini les fonctionnalités, précédemment expliquées, puis j'ai pu réfléchir à ce modèle que voici :



Cela nous donne un modèle très simple. Avec des « joueurs » qui ont des quêtes terminées ; des quêtes qui peuvent être faites par plusieurs joueurs. De plus, ces dernières appartiennent à un succès. Chacun des autres attributs sont les informations présentent dans le jeu.

Les difficultés rencontrées et mon ressenti

Ce projet était vraiment intéressant dans le sens où ça a permis de voir l'intégralité d'une application, à savoir la base de données, l'API et le front.

Cependant, j'ai pu rencontrer quelques difficultés, dans un premier temps j'étais seul à la réalisation de ce projet ainsi j'étais seul à la prise de décision alors qu'il peut toujours être intéressant d'avoir un deuxième avis. De plus, comme j'ai pu le dire un peu plus tôt, j'ai découvert un nouveau framework javascript, React, ce qui m'a posé des problèmes dans le sens où c'est une logique vraiment différente de jQuery, le framework avec lequel j'avais le plus l'habitude de travailler.

Autrement dans l'ensemble ça a été, vu que je travaillais déjà avec ces notions en entreprise, comme Spring Boot, service REST, les méthodes du CRUD, etc.