PROG3-TD2

On veut spécifier et implémenter l’API d’un match de football.

Compléter le formulaire suivant pour vos réponses : <https://forms.gle/Qs3MfZ8Jezz4Ynda6>

1er deadline : lundi 19 décembre avant le début des cours.

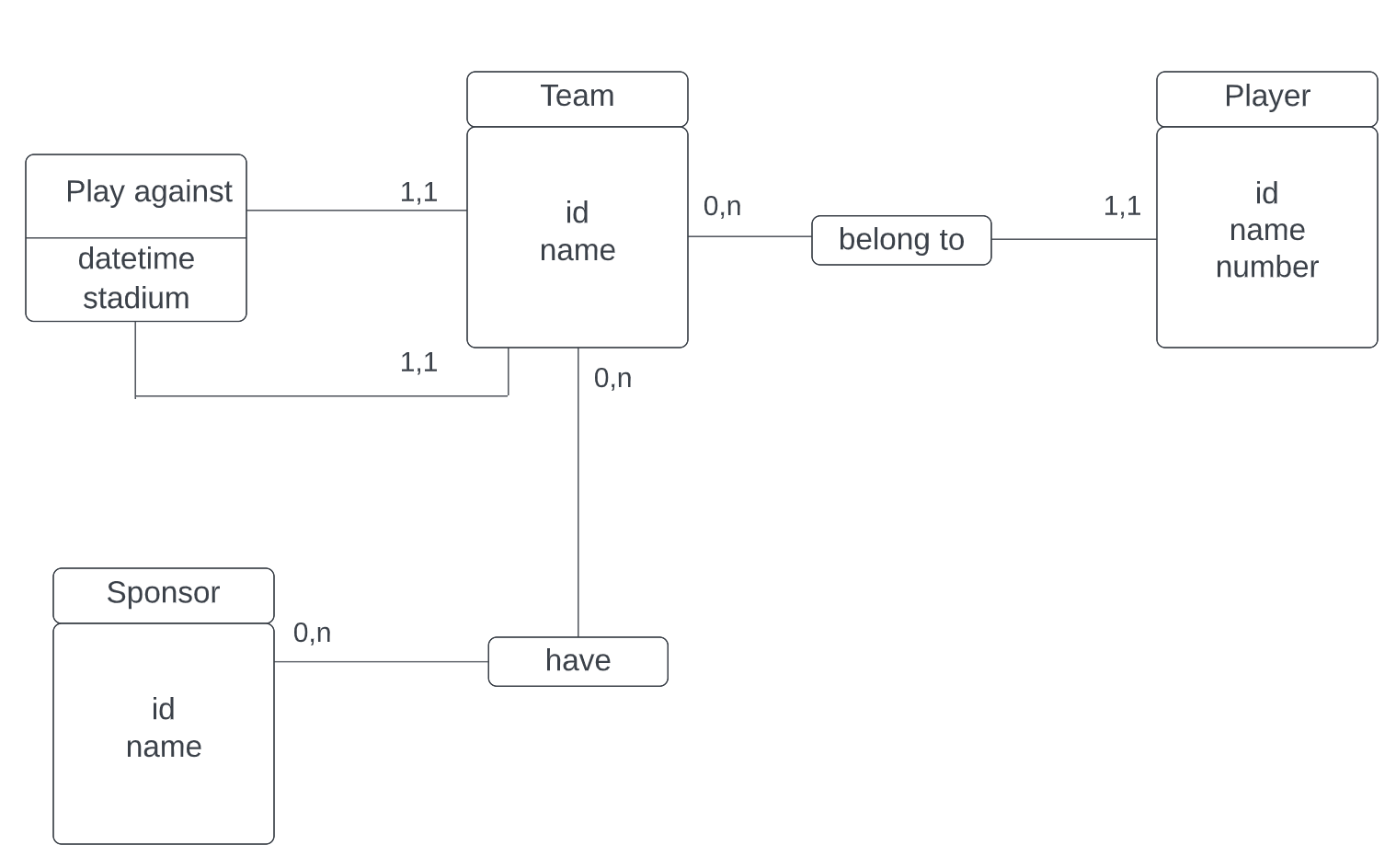
## Contexte :

Lors d’un match de foot, il y a deux équipes qui s’affrontent à une date et heure donnée et dans un stade donné.

* Chaque équipe peut posséder aucun ou plusieurs joueurs ainsi qu’aucun ou plusieurs sponsors.
* Un joueur ne peut appartenir qu’à une seule équipe et doit obligatoirement appartenir à une équipe.
* Un sponsor peut être le parrain de plusieurs équipes et doit obligatoirement au moins parrainer une équipe.

Voici un schéma qui modélise ces différentes relations :

**Figure 1 : MCD**



# Partie 1 :

## Objectifs :

* Revoir les bases de la modélisation et de la conception de données
* Réviser les bases du langage SQL

Q1. Est ce que mon modèle est correct par rapport au contexte établi ? Si ce n’est pas le cas, quelles corrections devrais-je effectuer ?

==>Elle n’est pas correct,

Entre Team et Sponsor devrait etre : Team 1,n —-have—-0,n Sponsor car un sponsor doit avoir obligatoirement une equipe a parrainer contrairement a une equipe.

Q2. De la figure 1, dégagez le modèle physique de données (MPD) en passant par le modèle logique des données (MLD). Utilisez <https://lucid.app/> pour créer vos diagrammes pour que vous puissiez les partager plus tard.

==> [lien vers lucidapp](https://lucid.app/lucidchart/fe637eb7-0ae4-4ec9-a3f2-01736589d241/edit?viewport_loc=402%2C-56%2C1711%2C1072%2C0_0&invitationId=inv_ba6ce54d-b929-4f5f-ac24-52bb9fbf65a4)

Q3.

1. Créez la base de données intitulée “prog3\_td2”.
2. Utilisez la base de données créée précédemment et écrivez les scripts SQL pour créer les tables correspondantes au MPD que vous avez généré dans un fichier appelé “V0\_1\_\_Initialize\_database.sql”.

Maintenant que vous avez modélisé le SI et que votre base de données est prête, vous allez accéder à vos données depuis votre application à travers Hibernate.

# Partie 2 :

## Objectifs :

* Se familiariser avec les concepts d’accès aux données à travers des objets : Hibernate ORM
* Réviser la configuration des scripts de migration (Flyway)
* Réviser la spécification d’une API REST suivant la spécification d’OpenAPI version 3.

Q1. Créer votre application Spring-Boot from scratch avec les dépendances nécessaires pour implémenter un service API REST, accéder à une base de données Postgresql et utiliser le script de migration avec Flyway.

Q2. Accéder aux tables que vous avez créées précédemment à travers des objets. Faites attention à bien établir les bonnes relations.

Q3. Spécifier l’API REST qui va retourner les données suivantes et mettez les dans votre projet (**docs/api.yml**) :

* Avoir la liste de tous les matchs, où chaque match comprend les deux équipes qui se confrontent ainsi que la date et l’heure du match. Notez que le stade ne sera PAS affiché.
* Pour chaque équipe, la liste des joueurs et la liste des sponsors sont affichées. La liste des joueurs affichés est ordonnée par leur numéro. Par exemple, le joueur Rakoto avec numéro 1 passera avant le joueur Rasoa avec numéro 2.
* Pour chaque joueur, seuls son ID et son nom sont affichés. Le numéro du joueur ne sera PAS affiché. Pour vérifier que l’ordre des numéros pour la liste des joueurs retournée est respecté, nous allons regarder les données directement dans la base de données.

Q4. Implémenter l’API spécifié précédemment.