

Mise en place de la base de données

Le premier schéma de relations d'entités de notre site de tournoi ne contient pas les entités arbre et championnat, dont nous parlerons plus tard. Elles seront explicitées dans les second et troisième schémas, et correspondent à la mise en œuvre d'un type particulier de tournoi.

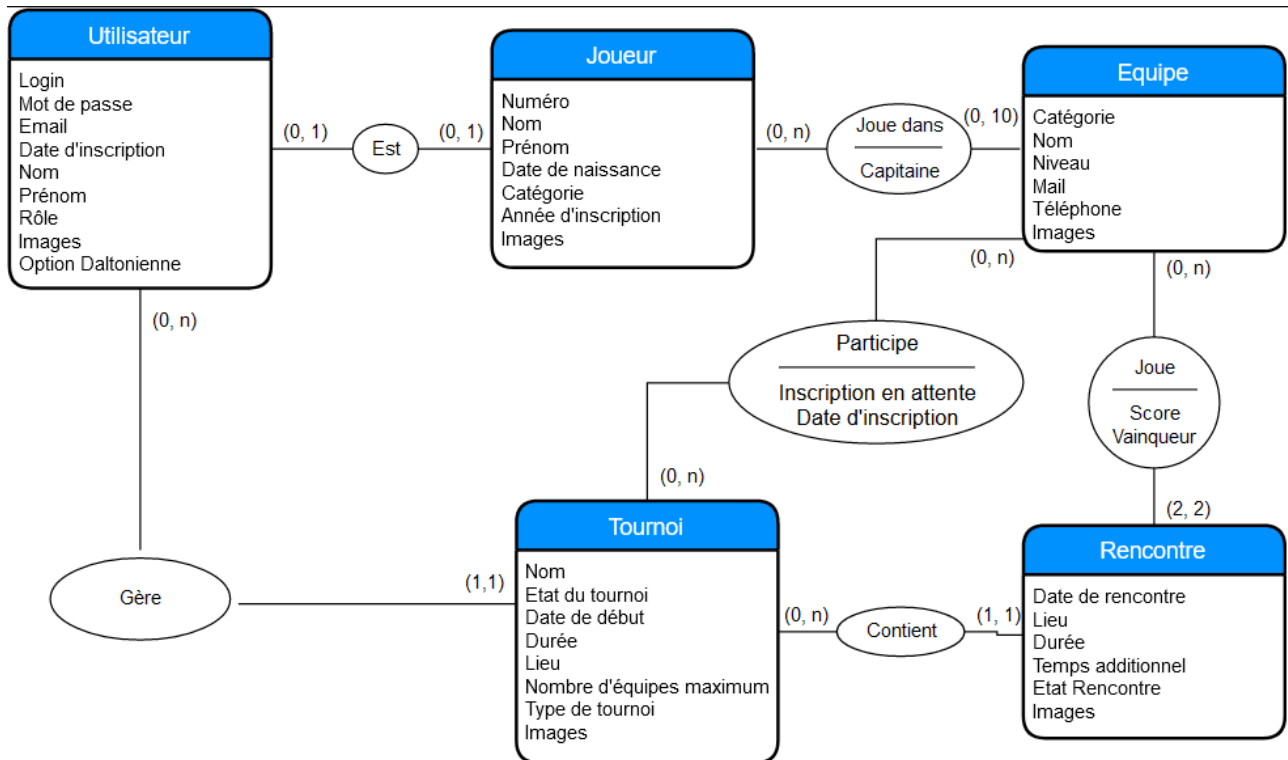


Figure 1: Schéma de relation entité-association

La première entité est l'utilisateur, ainsi que toutes ses informations. Il peut être joueur, mais aussi avoir un rôle qui définit sa position au cœur du site: "utilisateur", "gestionnaire" ou "administrateur". L'administrateur a accès à toutes les fonctionnalités, l'utilisateur peut s'inscrire et voir le déroulement des tournois, tandis que le gestionnaire peut gérer un ou plusieurs tournoi (tandis qu'un tournoi n'est géré que par un gestionnaire).

Le joueur peut donc être utilisateur ainsi qu'être membre d'une ou plusieurs équipes. Les équipes comprennent de 0 à 10 joueurs, dont l'un est capitaine.

L'entité tournoi contient entre 0 et n rencontres, et une liste d'équipes. La liste d'équipe contient pour chaque équipe un booléen : chaque équipe demande à s'inscrire à un tournoi, et le gestionnaire les accepte ou rejette. Le lien entre les rencontres et le tournoi sera fait ultérieurement par une entité dépendante du type de tournoi.

L'entité Rencontre est liée aux équipes par une cardinalité (2, 2): une rencontre est jouée par deux équipes, qui obtiennent chacune un score, et est gagnante ou perdante. Nous pouvons observer que dans la mise en œuvre de la base de données, puisque ce lien est constant, les références aux équipes scores et vainqueur ont intégré la table rencontre.

Nous avons hésité quant au comment représenter les listes d'équipes et de joueur dans une base de données. Initialement, nous avons tenté de créer des tables contenant autant d'attribut "équipeX" que le maximum d'équipes que nos fonctionnalités pourraient accepter, mais cela s'est avéré peu pratique à manipuler. Nous avons alors choisi de représenter les listes d'équipes et de joueurs au travers de deux tables liées par des références aux tables concernées. Pour les listes d'équipes, chaque tuple référence une équipe et un tournoi, permettant ainsi de trier dans un sens et dans l'autre aisément tous les tournois auxquels participe une équipe, ou toutes les équipes participant à un tournoi.

Dans un premier temps, nous avons choisi de nous concentrer sur le type de tournoi "Coupe", le type correspondant aux critères attendu, à savoir une représentation en arbre. Plus tardivement, nous avons ajouté un second type de tournoi, le "Championnat", dont l'information est contenue dans une seconde table construite de manière similaire à la table "Arbre" et que nous allons détailler dans ce second schéma.

Championnat
IdChampionnat INT AUTO INCREMENT
Rencontre INT REFERENCES Rencontre (IdRencontre)
Tournoi INT REFERENCES Tournoi (IdTournoi)
Journee INT
Resultat INT

Dans un tournoi "Championnat", l'interface entre les entités tournoi et rencontres est représentée par la table ci-contre. Chaque entrée référence donc les deux tables, et contient deux informations supplémentaires: Journée et Résultat. Les championnats sont en effet organisés par

journée, et à la différence des coupes, acceptent des matchs se soldant par une égalité entre les deux équipes. Le résultat symbolise cela, en codant l'information en fonction de l'équipe jouant à domicile (représentée par l'équipe 1 dans la table rencontre).

Ci-dessous, le schéma qui nous a servi de référence durant la phase de programmation. Lors de sa conception n'y figurait alors pas encore la table "Championnat", dont nous ne savions pas encore si nous l'insérerions. Afin de manipuler la base de données, nous avons d'abord utilisé mysql, puis sommes passés à PDO. Lors de l'implémentation de la base de données, nous avons décidé que toutes les clés primaires seraient insérées en auto-increment, c'est-à-dire que nous ne les préciserions pas lors de l'ajout de nouvelles valeurs; que les booléens seraient enregistrés sous forme de BIT(1), de valeur 0 ou 1.

La modélisation des rencontres d'un tournoi "Coupe" se fait au travers de la table arbre, qui référence le tournoi, les équipes participant à une rencontre, et une rencontre, mais aussi elle-même. Chaque nœud de l'arbre binaire correspond à une rencontre. Ses parents sont les noeuds/rencontres du tour précédents, et son fils est le noeud/rencontre du tour suivant qu'il engendre. Un attribut Hauteur permet d'enregistrer la hauteur du noeud, 0 correspondant à la finale, 1 à un noeud de demie-finale, etc.

Nous pouvons remarquer a priori que plusieurs attributs pourraient ne pas être contenus dans la table arbre: Equipe1/Equipe2 ainsi que Hauteur. Concernant Hauteur, elle pourrait être obtenue après plusieurs requêtes ou un calcul sur tous les tuples d'Arbre pour un Tournoi donné. Ce serait coûteux à mettre en place, plus que de retenir l'information dans la table. Quant à Equipe1/Equipe2, les deux attributs paraissent redondant vis à vis de la table Rencontre, qui les contiennent aussi. C'est une question de conception: dans un tournoi de type "Coupe", la table Arbre est construite

avant que les rencontres soient créées, ce qui demande de retenir l'emplacement des équipes. Et si nous mettions en place d'autres types de tournoi, ou des rencontres amicales, la table rencontre devait contenir l'information. Nous avons donc fait sciemment le choix d'une apparente redondance sur ces deux attributs.

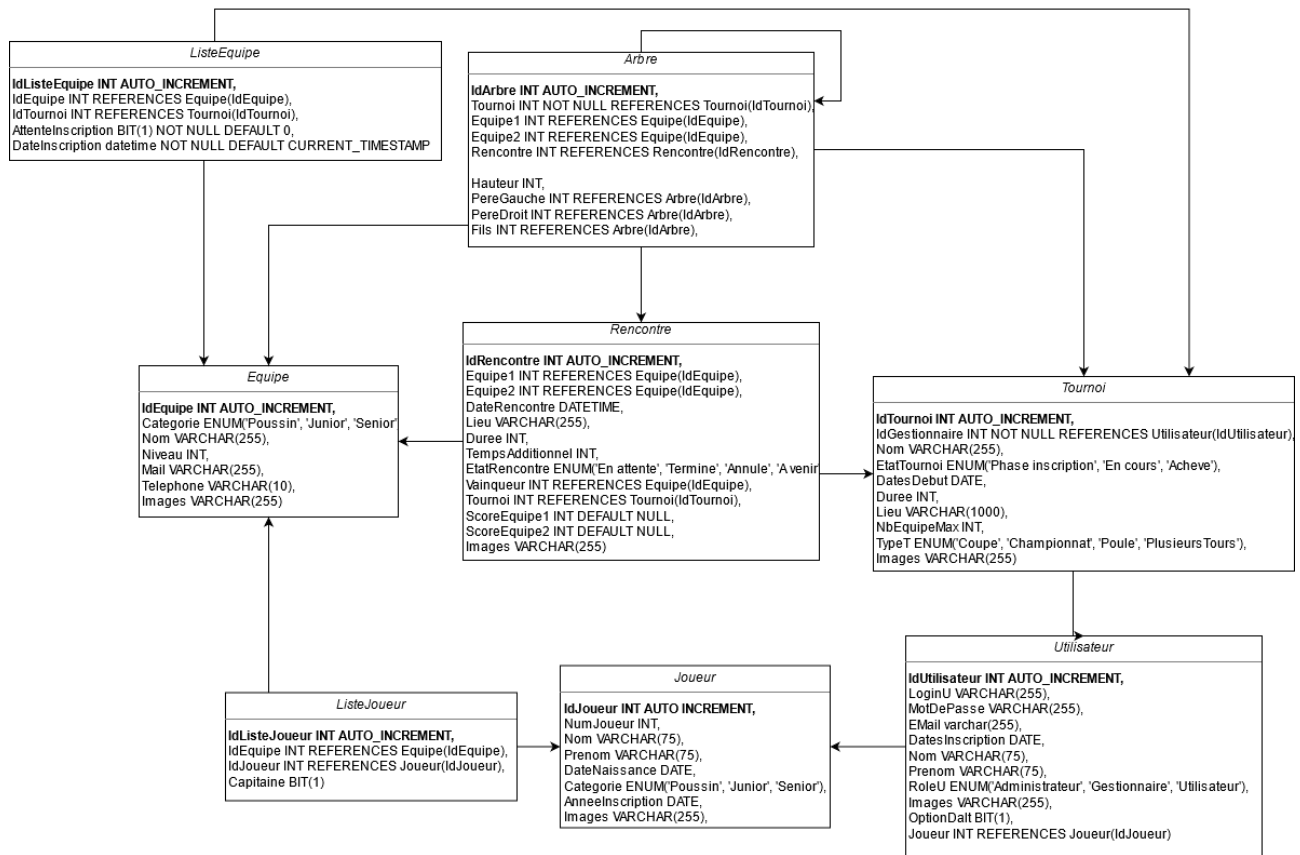


Figure 2: Table de référence pour la base de donnée