**IFT2935 – Projet noté**

Tina Liu Lee (20092684)

Bojan Odobasic (952514)

Jean-Marc Prud’homme (20137035)

Jean-Daniel Toupin (20046724)

Michel Boyer

Université de Montréal

16 avril 2020

**Tâche 1. La modélisation**

Projet choisi : La coupe du monde des nations du football

**Diagram

Description automatically generated**

Détails :

On assume que les dates sont des attributs simples puisque l’implémentation se fera dans PostgreSQL.

* Coupe du monde
  + Chaque coupe a déjà un numéro d’édition unique et connu.
  + Une coupe peut avoir lieu dans plus d’un pays en même temps.
* Équipe
  + Il s’agit d’une entité faible car la composition des équipes d’une même nation change à chaque coupe. Cette même équipe ne participe qu’à une seule coupe.
  + Placement représente la position de l’équipe dans le classement final de la coupe.
* Personne
  + L’ajout d’un identificateur pour chaque personne s’avère nécessaire car il est difficile de garantir l’unicité avec les attributs personnels usuels.
* Collaborateur :
  + Une équipe peut avoir plusieurs collaborateurs. Les collaborateurs peuvent assister plusieurs équipes (puisqu’elles changent à chaque coupe).
  + On conserve la date de début de sa carrière qui peut être hors coupe du monde.
* Entraîneur :
  + Une équipe a un seul entraîneur. Les entraîneurs peuvent entraîner plusieurs équipes (puisqu’elles changent à chaque coupe).
  + On conserve la date de début de sa carrière qui peut être hors coupe du monde.
* Joueur :
  + Une équipe a plusieurs joueurs. Les joueurs peuvent faire partie de plusieurs équipes (puisqu’elles changent à chaque coupe).
  + On conserve la date de début de sa carrière qui peut être hors coupe du monde.
  + L’association entre le joueur et l’équipe de la coupe du monde permet de conserver l’équipe professionnelle usuelle du joueur (son équipe hors coupe).
* Stade :
  + Il est rare que deux stades portent le même nom, mais ce n’est pas impossible. L’ajout de la ville garantie l’unicité. Le pays n’offre pas la même garantie.
* Match :
  + Deux équipes d’une même coupe peuvent s’affronter plus d’une fois dans la même coupe. Cependant, ces matchs n’auront pas lieu la même journée. Il s’agit donc d’une entité faible déterminée par les deux équipes ainsi que la date.
* Arbitre :
  + Un arbitre peut gérer plusieurs matchs. Chaque match a un à quatre arbitres.
  + Chaque arbitre a un type (principal ou assistant) qui peut changer selon le match.
  + On conserve la date de début de sa carrière qui peut être hors coupe du monde.
* Sanction :
  + L’ajout d’un identificateur est essentiel car pour un même match, un joueur peut recevoir plusieurs sanctions de même couleur du même arbitre.

**Tâche 2. La transformation**

Coupe\_du\_monde (**edition**, date\_debut, date\_fin)

* Entité devient une table.
* L’édition est unique et devient la clé primaire. La date de début et a date de fin sont des clés candidates.

Pays\_enum (**pays**)

* Table représentant un type énuméré de pays. L’attribut pays agit comme clé.

Pays\_coupe (**#edition\_coupe**, **#pays**)

* Attribut multivalué de Coupe du monde devient une table.
* L’édition de la coupe est une clé étrangère. Le pays appartient au type énuméré et est une clé étrangère également.
* Chaque entrée est unique donc ses deux attributs forment la clé primaire.

Equipe (**#nation**, **#edition\_coupe,** #entraineur\_id, placement)

* Entité devient une table.
* Lien 1:1 avec Participe : on ajoute l’édition de la coupe à la table comme clé étrangère.
* Lien 1:1 avec Géré\_par : on ajoute l’identificateur de l’entraîneur à la table comme clé étrangère.
* La nation appartient au type énuméré de pays et est une clé étrangère.
* Il s’agissait d’une entité faible (l’équipe d’une nation change à chaque coupe). La nation de l’équipe et l’édition de la coupe forment la clé primaire.

Personne (**personne\_id**, nom, prenom, ddn, #pays\_natal, sexe)

* Entité devient une table.
* Le pays appartient au type énuméré et est une clé étrangère.
* L’identificateur est unique et devient la clé primaire.

Joueur (**#personne\_id**, joueur\_depuis)

* Entité devient une table.
* L’entité hérite de Personne : l’identificateur est la clé étrangère.

Entraineur (**#personne\_id**, entraineur\_depuis)

* Entité devient une table.
* L’entité hérite de Personne : l’identificateur est la clé étrangère.

Collaborateur (**#personne\_id**, expertise, collaborateur\_depuis)

* Entité devient une table.
* L’entité hérite de Personne : l’identificateur est la clé étrangère.

Arbitre (#**personne\_id**, arbitre\_depuis)

* Entité devient une table.
* L’entité hérite de Personne : l’identificateur est la clé étrangère.

Stade (**nom**, **ville**, #pays\_stade, capacite, annee\_construction)

* Entité devient une table.
* Le pays appartient au type énuméré et est une clé étrangère.
* Le nom et la ville forment la clé primaire.

Match (**date, #nation\_equipe\_1**, **#nation\_equipe\_2**, #edition\_coupe, rang, #nom\_stade, #ville\_stade, score\_equipe\_1, score\_equipe\_2)

* Entité devient une table.
* Lien 1:1 avec Organise : on ajoute l’édition de la coupe à la table comme clé étrangère.
* Lien 1:1 avec Accueille : on ajoute le nom du stade et la ville comme clés étrangères.
* Lien 1:1 avec Joue\_dans : on ajoute les clés des deux équipes comme clés étrangères, soit leurs nations respectives ainsi que l’édition de la coupe (qui est la même pour les deux équipes et la même que celle déjà ajoutée par l’association Organise).
* Il s’agissait d’une entité faible. Les deux nations ainsi que la date forment la clé primaire.

Sanction (**sanction\_id,** #joueur\_id, #arbitre\_id, #nation\_equipe\_1, #nation\_equipe\_2, #date\_match, couleur)

* Entité devient une table.
* Lien 1:1 avec Sanction\_survenue : on ajoute les clés du match comme clés étrangères, soit les deux nations et la date.
* Lien 1:1 avec Recoit\_sanction : on ajoute l’identificateur du joueur comme clé étrangère.
* Lien 1:1 avec Donne\_sanction: on ajoute l’identificateur de l’arbitre comme clé étrangère.
* L’identificateur est unique et devient la clé primaire.

Collaborateur\_equipe (**#collaborateur\_id , #nation\_equipe, #edition\_coupe**)

* Association sans lien 1:1 devient une table.
* Les clés primaires du collaborateur (son identificateur) ainsi que celles de l’équipe (sa nation et l’édition de la coupe) deviennent des clés étrangères et forment la clé primaire.

Equipe\_ligue\_enum (**equipe**)

* Table représentant un type énuméré d’équipes des ligues profesionnelles. L’attribut équipe agit comme clé.

Joueur\_equipe (#**joueur\_id**, **#nation\_equipe**, **#edition\_coupe** , position, numero\_dossard, #equipe\_ligue\_professionnelle)

* Association sans lien 1:1 devient une table.
* Les clés primaires du joueur (son identificateur) ainsi que celles de l’équipe (sa nation et l’édition de la coupe) deviennent des clés étrangères et forment la clé primaire.

Arbitre\_match (**#arbitre\_id**, **#date\_match, #nation\_equipe\_1, #nation\_equipe\_2,** type\_arbitre)

* Association sans lien 1:1 devient une table.
* Les clés primaires de l’arbitre (son identificateur) ainsi que celles du match (les deux nations et la date) deviennent des clés étrangères et forment la clé primaire.

**Tâche 3. La normalisation**

Nous assumons que toutes les tables sont déjà 1FN. Nous utilisons la définition de Zaniolo pour 3FN qui implique 2FN.

Coupe\_du\_monde (edition, date\_debut, date\_fin)

Diagram

Description automatically generatedDf = { (edition -> date\_debut, date\_fin), (date\_debut -> edition, date\_fin), (date\_fin -> edition, date\_debut) }

Il n’y a toujours qu’une seule coupe du monde en même temps. Edition, date\_debut et date\_fin sont donc toutes des clés de la relation. Puisque l’édition des coupes est un numéro débutant à 1 et qui incrémenté à chaque nouvelle coupe, c’est un candidat idéal comme clé primaire qui sera plus facile à utiliser que les dates. La table est déjà en 3FN et en FNBC car tous les déterminants sont des clés candidates.

Pays\_coupe (edition\_coupe, pays)

Df = { }

Aucune DF car les deux attributs forment l’unique clé. Elle est trivialement 3FN et FNBC.

Equipe (nation, edition\_coupe**,** entraineur\_id, placement)

Diagram

Description automatically generatedDf = { (nation, edition\_coupe-> entraineur\_id, placement), (entraineur\_id, edition\_coupe-> nation, placement), (placement, edition\_coupe-> nation, entraineur\_id) }

Un pays est représenté par une seule équipe dans une édition de coupe donnée. Son entraîneur est unique, tout comme son placement. On remarque alors que n’importe quel attribut associé avec l’édition de coupe détermine une équipe précise et donc le reste des attributs. Comme clé primaire, nous choisissons nation et edition\_coupe car c’est habituellement de cette manière qu’on identifie les équipes de la coupe du monde. La table est déjà en 3FN et en FNBC car tous les déterminants sont des clés candidates.

Personne (personne\_id, nom, prenom, ddn, pays\_natal, sexe)

Df ={ personne\_id -> nom, prenom, ddn, pays\_natal, sexe }

Il n’y a qu’un identificateur unique qui détermine tous les attributs. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Joueur (personne\_id, joueur\_depuis)

Df = { personne\_id -> joueur\_depuis }

Il n’y a qu’un identificateur unique qui détermine l’autre attribut. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Entraineur (personne\_id, entraineur\_depuis)

Df = { personne\_id -> entraineur\_depuis }

Il n’y a qu’un identificateur unique qui détermine l’autre attribut. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Collaborateur (personne\_id, expertise, collaborateur\_depuis)

Df = { personne\_id -> expertise, collaborateur\_depuis }

Il n’y a qu’un identificateur unique qui détermine les autres attributs. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Arbitre (personne\_id, arbitre\_depuis)

Df = { personne\_id -> arbitre\_depuis }

Il n’y a qu’un identificateur unique qui détermine l’autre attribut. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Stade (nom, ville, capacite, pays\_stade, annee\_construction)

Df = { nom, ville -> capacite, pays\_stade, annee\_construction }

Le nom et la ville déterminent uniquement le stade et ses autres attributs. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Match (date, nation\_equipe\_1, nation\_equipe\_2, rang, edition\_coupe, nom\_stade, ville\_stade, score\_equipe\_1, score\_equipe\_2)

Df= { date, nation\_equipe\_1, nation\_equipe\_2 --> rang, edition\_coupe, nom\_stade, ville\_stade, score\_equipe\_1, score\_equipe\_2 }

Il est tout à fait possible que deux équipes s’affrontent à nouveau dans une même coupe du monde dans le même rang. Cependant, deux équipes ne s’affronteront jamais deux fois dans la même journée. Il s’agit de la seule clé, tous les autres attributs en dépendent. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Sanction (sanction\_id, joueur\_id, arbitre\_id, nation\_equipe\_1, nation\_equipe\_2, date\_match, couleur)

Df = { sanction\_id -> joueur\_id, arbitre\_id, nation\_equipe\_1, nation\_equipe\_2, date\_match, couleur }

Il n’y a qu’un identificateur unique qui détermine tous les autres attributs. La table est déjà en 3FN et en FNBC car le seul déterminant est la clé primaire.

Collaborateur\_equipe (collaborateur id , nation\_equipe, edition\_coupe)

Df = { }

Aucune DF car il n’y a pas d’attribut particulier à cette association. Cependant nous avons besoin de cette relation pour joindre les collaborateurs au reste de la BDD par jointure.

Joueur\_equipe (joueur\_id, nation\_equipe, edition\_coupe, position, numero\_dossard, equipe\_ligue\_professionnelle)

Df = { joueur\_id, nation\_equipe, edition\_coupe -> position, numero\_dossard, equipe\_ligue\_professionnelle }

La raison nous avons besoin des 3 attributs pour définir position, dossard et équipe prof est parce qu’un même joueur peut jouer dans plusieurs équipes et une équipe a plusieurs joueurs. La raison pour laquelle nous avons besoin de joueur\_id + equipe pour définir position, dossard, et équipe professionnelle est que la position, le dossard et l’équipe professionnel actuelle au moment de la coupe peuvent changer entre chaque coupe.

Arbitre\_match (arbitre\_id, date\_match, nation\_equipe\_1, nation\_equipe\_2, type\_arbitre)

Df = { arbitre\_id, date\_match, nation\_equipe\_1, nation\_equipe\_2 -> type\_arbitre }

Un peu dans la même optique que Joueur\_equipe. Il y a plusieurs arbitres dans un match et un arbitre peu arbitrer plusieurs matchs. Alors nous avons besoin de définir match et arbitre au complet avant de savoir quel type d’arbitre il a été dans ce match. À partir de ça, nous pouvons savoir si l’arbitre était principal ou assistant.

**Schéma relationnel final**

Coupe\_du\_monde (**edition**, date\_debut, date\_fin)

Pays\_enum (***pays***)

Pays\_coupe (***#edition\_coupe***, #***pays***)

Equipe (#**nation**, **#edition\_coupe,** #entraineur\_id, placement)

Personne (**personne\_id**, nom, prenom, ddn, #pays\_natal, sexe)

Joueur (#**personne\_id,** joueur\_depuis)

Entraineur (#**personne\_id**, entraineur\_depuis)

Collaborateur (#**personne\_id**, expertise, collaborateur\_depuis)

Arbitre (#**personne\_id**, arbitre\_depuis)

Stade (**nom**, **ville**, #pays\_stade, capacite, annee\_construction)

Match (**date, *#nation\_equipe\_1*, #nation\_equipe\_2,** rang, #edition\_coupe, *#nom\_stade*, #ville\_stade, score\_equipe\_1, score\_equipe\_2)

Sanction (**sanction\_id,** #joueur\_id, #arbitre\_id, #nation\_equipe\_1, #nation\_equipe\_2, #date\_match, couleur)

Collaborateur\_equipe **(#collaborateur\_id, #nation\_equipe, #edition\_coupe**)

Equipe\_ligne\_enum (**equipe**)

Joueur\_equipe (**#joueur\_id**, **#nation\_coupe**, **#edition\_coupe,** position, numero\_dossard, #equipe\_ligue\_professionnelle)

Arbitre\_match (**#arbitre\_id**, **#date\_match, #nation\_equipe\_1, #nation\_equipe\_2,** type\_arbitre)

Nous pouvons remarquer que nous pouvons accéder à tous les informations de la BDD par jointure entre les relations. En partant de la relation Equipe, nous avons accès à n’importe quelle information que nous voulons par jointure.

**Tâche 4. L’implémentation**

**Tâche 5. Questions/réponses**

**Tâches 6 et 7. Interaction avec Java et utilisation d’un ORM (Hibernate)**

Pour exécuter l’application, il est nécessaire d’avoir une installation de Java 15 ainsi que de JavaFX 16. À partir du terminal, naviguer vers le dossier où se trouve le ficher JAR et exécuter :

java --module-path {CHEMIN\_VERS\_JAVAFX} --add-modules javafx.controls,javafx.fxml -jar Projet.jar

En remplaçant {CHEMIN\_VERS\_JAVAFX} par la localisation du dossier lib de JavaFX (inclusivement).

Pour compiler l’application, il faut avoir également Maven d’installé pour la gestion des dépendances. À partir du terminal, naviguer vers le dossier où se trouve le fichier *pom.xml* et exécuter :

mvn clean package

Le fichier résultant *IFT2935-Projet-1.0-SNAPSHOT-jar-with-dependencies.jar* du dossier *target* nouvellement créé peut être exécuté avec les instructions ci-haut.

Maven peut également être utilisé pour compiler et lancer l’application sans passer par un fichier JAR en exécutant plutôt :

mvn javafx:run

À noter qu’il est possible de changer les versions de Java et de JavaFX utilisées en modifiant les lignes 13 et 14 (pour Java) et les lignes 21 et 26 (pour JavaFx) du fichier *pom.xml*.

Graphical user interface, text, application, email

Description automatically generatedGraphical user interface, text, application, chat or text message

Description automatically generatedÀ l’ouverture de l’application, remplacer les informations de connexion par défaut si nécessaire puis cliquer sur *Enregister*. Cliquer ensuite sur un des quatre boutons pour faire apparaître la question et la réponse.