

Projet informatique : Gestion d'un tournoi sportif

Notre application permet la création d'un tournoi sportif, quelque soit le sport voulu, ainsi que le suivi de ce tournoi. L'objectif étant de permettre à l'utilisateur de suivre, et ce à n'importe quel moment, l'avancée du tournoi créé.

I°/ Organisation du projet

Nous avons tout d'abord réalisé la partie analyse de notre sujet, essentielle afin de pouvoir coder l'application plus aisément. Nous avons fait trois diagrammes UML nous permettant de nous rendre compte du nombre de classes à créer et des différents éléments qui les relient. Ceci étant fait nous avons décidé de nous répartir les tâches : pendant que l'un créait les tables de la base de données(BDD) dont nous allions nous servir, l'autre commençait à écrire les premiers scripts permettant la création d'un tournoi. Une fois la BDD produite, nous avons regroupé nos fichiers et avons pu écrire les méthodes permettant à l'utilisateur de suivre le tournoi.

Une fois toutes les méthodes écrites, nous nous sommes réunis sur un même poste afin de remplir notre main et de donner notre point de vue sur les scripts de notre binôme.

II°/ Organisation des scripts

L'ensemble de nos scripts sont essentiellement basés sur des requêtes SQL, cela permettant de garder en mémoire toutes les données transmises par l'utilisateur et de les restituer plus facilement. Premièrement il était essentiel de pouvoir communiquer avec le créateur du tournoi pour connaître les informations dont nous avons besoin (sport, équipes, joueurs, etc) pour exécuter et gérer la suite de notre application. C'est pourquoi nous avons décidé de placer une classe utilitaire à part des autres et qui ne possède que ce rôle d'interface d'interlocution. Nous avons ensuite notre classe tournoi, classe mère possédant les attributs réutilisés dans nos classes filles qui elles permettent de créer trois tournois différents. En effet, nous avons réalisés qu'il existait trois types de tournois : les tournois à phase éliminatoire (ou à élimination directe) comme Roland-Garros, les championnats comme la ligue1 Conforama et les tournois associant les deux tel que la coupe du monde de football.

Nous avons également effectué une distinction entre les résultats que pouvaient ramener une équipe. Il en existe également trois : une distance (saut en longueur), un temps (marathon) ou des points (football). Les calculs des vainqueurs et des classements se font alors différemment, la plus grande distance l'emportant tout comme le temps le plus rapide.

Pour ce qui est de la création d'une interface, celui-ci n'a pas été notre priorité, notre application n'en ayant pas nécessairement besoin.

III°/ Exécution de l'application

Notre application se décompose en trois packages :

- le package tournoi
- le package connexion
- le package main

Le package tournoi possède toutes les classes permettant à l'utilisateur de créer et de suivre des tournois différents dans des sports différents. Ce sont ces classes qui possèdent toutes les méthodes retournant les informations recherchées et permettant d'identifier les vainqueurs des différents sports.

Le package connexion possède la classe Connexion uniquement utilisée pour se connecter à la BDD et la classe UtilitarianClass permettant d'interagir avec l'utilisateur.

Le package main possède une classe unique Main qui sera celle utilisée afin de lancer l'application. C'est à l'intérieur de celle-ci qu'interagissent les différentes méthodes du package tournoi.

Afin d'utiliser notre application, il suffit donc de lancer la méthode du fichier main.java et de répondre aux différentes questions posées. En revanche, même si nous avons renseigné les différents choix permettant de créer un tournoi à phase éliminatoire ou complet, seuls les choix 1/3/4 vous mèneront au bout puisque nous avons privilégié la réalisation totale d'un championnat à la construction partielle des trois tournois.

IV°/ Bilan du projet

1 – Les améliorations que nous aurions pu apporter

Nous avons émis beaucoup d'hypothèse nous permettant d'avancer plus rapidement dans la réalisation de notre code.

Nous avons notamment décidé de faire des tournois mixtes, le sexe des personnes ne faisant pas partie des données demandées. De la même façon, un athlète handicapé peut tout à fait courir à côté d'un athlète valide.

Nous aurions pu également rajouter des informations sur les performances des équipes : le goal average en football, les différentes performances réalisées en lancer de poids (nous ne prenons que la meilleure).

Nous avons omis de prendre en compte des joueurs protégés : en tennis, certains athlètes rentrent dans le tournoi directement en 16^{ème} de final au vu de leur classement. Il était plus simple pour nous d'avoir directement tous les joueurs commençant la compétition au même stade.

Le choix de l'utilisation des BDD et des requêtes SQL s'est fait car nous les manipulons plus aisément, et qu'il nous était alors possible d'avancer plus vite. En revanche, nous pensons que ce n'était pas le moyen le plus adapté et qu'il aurait été plus cohérent d'utiliser les liens possibles entre nos classes à partir des méthodes d'orienté objet de JAVA. Nous aurions tout de même du utiliser les bases de données afin de garder une liste des différents sports et des catégories en mémoire.

Pour ce qui est des catégories justement, nous avons décidé de les intégrer dans le nom pour éviter d'avoir à vérifier tout d'abord que le sport existe, puis la catégorie. Nous avons préféré faire une pierre deux coups et de tout vérifier et modifier d'un coup.

Les autres améliorations possibles se font sur l'organisation du tournoi en lui-même. Dans notre programme il est possible de gérer un seul tournoi à la fois. Une fois le tournoi terminé, les données propres à celui-ci sont effacées pour laisser la place à celles du tournoi suivant. Il n'est donc plus possible de retrouver les informations d'un tournoi terminé. Nous n'avons pas prit en compte non plus les problèmes de temporalité et de réservation des différents lieux de tournoi. Nous considérons que chaque match peut se faire à n'importe quel moment et qu'il trouve forcément un endroit où se réaliser.

Pour finir, il aurait été cohérent de prendre en compte l'état des sportifs et de pouvoir laisser les utilisateurs modifier les équipes en cas d'expulsion du tournoi ou de blessure.

2 – Ce que nous avons réalisé

Notre analyse montre un tournoi se décomposant de trois manières différentes : tournoi championnat, tournoi phase éliminatoire, tournoi mixte. Après les vacances, nous nous sommes rendus compte que nous ne pourrions pas réaliser toutes les méthodes pour les trois tournois et nous avons donc préféré nous concentrer sur le tournoi championnat afin qu'au moins un des trois fonctionne totalement. Nous avons donc réussi à créer une application où la mise en place d'un tournoi de type « championnat » ainsi que son suivi est possible. Il est donc possible à l'utilisateur de créer un tournoi avec le sport qu'il souhaite et de rentrer les équipes à inscrire.

Il pourra ensuite rentrer les résultats des matchs qu'il souhaite puis suivre le classement du championnat au fur et à mesure que les résultats sont rentrés dans la BDD par l'utilisateur. Il peut également afficher le classement en temps réel du championnat avec le nombre de points de chaque équipe. Bien sûr, toutes ces demandes pourront se faire seulement si l'utilisateur a rentré le bon code tournoi fourni par le créateur.

3 – Ce que nous a apporté le projet informatique

Premièrement, le fait de faire plein d'erreurs (ce à quoi nous n'avons pas échappé) permet de se rendre compte du type de celles-ci et de s'en rappeler pour les fois d'après. Nous avons vite remarqué que nous étions plus rapidement à l'aise pour aller chercher des données dans notre BDD après avoir subi plusieurs échecs. Une fois que nous tombons sur un type d'erreur et que nous trouvons sa solution, la fois d'après il est plus rapide de la résoudre.

Nous avons également pu nous confronter à des requêtes SQL ce qui nous a permis de nous remémorer les différentes manières d'effectuer des commandes dans ce langage. Il est assez intéressant de pouvoir être témoin du regroupement des différentes matières vu au cours de l'année, que ce soit l'UML, l'algorithmique ou encore les bases de données. Ce sont en plus des matières que nous avons vu assez tôt dans l'année et cela fait toujours du bien de se replonger dedans.

Un autre point positif à travailler sur un projet aussi important que celui-ci est de parvenir à gérer son temps de travail et de réussir à se répartir les tâches entre les membres du groupe. Nous étions partis à la base pour tout faire au fur et à mesure mais nous avons remarqué que nous manquerions de temps. Il nous était donc nécessaire de privilégier la bonne réussite d'une partie de notre projet à la réalisation totale mais partiellement réussie de celui-ci.