

# Projet Bases de Données et Conception Objet

---

## 1 – Introduction

Ce projet a pour but de mettre en œuvre vos compétences en systèmes de gestion de bases de données relationnelles et en conception objet. Ce projet vient en application du cours de « Principes des systèmes de gestion de bases de données » (PSGBD) et du cours d'« analyse, conception et validation des logiciels » (ACVL). Vous devez réaliser l'analyse, la conception et la programmation complète d'une application de gestion de tournois d'échecs utilisant une base de données. Vous devez notamment concevoir et implanter un schéma relationnel. Le programme doit être écrit en *Java*, et les accès à la base de données effectués grâce l'API *JDBC*.

Le projet est à faire en quadrinômes et donnera lieu à la remise d'un rapport, ainsi qu'à une soutenance comprenant, entre autres, une démonstration. L'évaluation concerne l'analyse, la conception et l'implémentation, à la fois sur les aspects bases de données et sur la partie programmation orientée objets.

## 2 – Description de l'application

L'application que vous devez développer doit permettre la gestion d'un tournoi d'échecs. Un tournoi se compose de plusieurs rencontres organisées en différents tours entre les joueurs qui se sont inscrits.

L'application comportera une section organisation de tournoi permettant d'inscrire les participants et d'organiser chacun des tours du tournoi (qualifications, quart de finale, demi finale et finale). Elle comportera également une section permettant de jouer une rencontre parmi celles du tour actif, les joueurs accédant alternativement à l'application pour jouer leurs coups. Plusieurs parties d'un même tour pourront être jouées en même temps en lançant l'application plusieurs fois. Tous les coups d'une partie sont enregistrés de façon à pouvoir rejouer la partie pour analyse a posteriori dans la troisième section de l'application.

Vous pouvez choisir de faire une interface textuelle ou graphique à votre convenance (le projet est tout de même à réaliser en temps limité). De plus, il n'est pas nécessaire pour cette version du logiciel que l'accès des personnes l'utilisant soit géré par un système d'identification et de mot de passe. Il vous est explicitement demandé de ne pas réaliser cela : la personne utilisant le logiciel sera simplement sélectionnée parmi la liste des personnes enregistrées.

### a) Organisation d'un tournoi

Les rencontres d'un tournoi se conforment au programme suivant:

- un *tour de qualifications*, lors duquel chaque participant doit rencontrer chacun des autres ;

- un *tour de quart de finale*, lors duquel les 8 joueurs ayant remportés le plus de victoires au tour de qualification se rencontrent deux à deux (1<sup>er</sup> et 8<sup>ème</sup>, 2<sup>nd</sup> et 7<sup>ème</sup>, 3<sup>ème</sup> et 6<sup>ème</sup>, 4<sup>ème</sup> et 5<sup>ème</sup> dans le classement) ;
- un *tour de demi-finale*, où les vainqueurs des deux premiers quarts de finale se rencontrent et les vainqueurs des deux derniers quarts de finale se rencontrent également ;
- une *finale* où s'affrontent les deux vainqueurs des demi-finales.

Un joueur est identifié par un numéro unique pour le tournoi et est décrit par son nom, son prénom, son adresse postale et sa date de naissance. Une rencontre entre deux joueurs est identifiée par le tour et le numéro de partie dans le tour (ex : 1er quart de finale). La couleur des pièces d'un joueur (blanc ou noir) pour une partie est tirée aléatoirement. Le vainqueur de la partie doit être mémorisé afin de faciliter la préparation du tour suivant.

### b) Jouer une partie

L'échiquier sera représenté par l'ensemble des pièces présentes (32 pièces en début de partie) et leurs positions. Précisément, une pièce est identifiée par un numéro unique pour la partie en cours et décrite par sa couleur, sa position courante sur une grille de 8x8 cases, ainsi que sa position précédente. Cette dernière information servira à valider le déplacement d'une pièce en fonction de son type et de sa couleur.

Nous allons considérer ici des règles du jeu d'échecs simplifiées par rapport aux règles officielles<sup>1</sup> :

- les blancs sont les premiers à jouer ;
- les pions ne se déplacent que d'une seule case, quelles que soient leurs positions ;
- pas de roque ;
- pas de prise en passant ;
- pas de promotion ;
- un joueur peut décider à tout moment d'abandonner la partie ;
- si les deux joueurs sont d'accord, une partie peut être annulée et rejouée depuis le début (exemple : cas du *pat*).

Avant qu'un joueur ne joue, il faut vérifier sa situation :

- vérifier s'il est en *échec* ;
- le cas échéant, vérifier s'il est *mat* :
  - vérifier les déplacements possibles du roi,
  - vérifier s'il est possible ou non de prendre la pièce qui le met en échec sans se démasquer (et se retrouver de nouveau en échec),
  - vérifier s'il est possible ou non de mettre un obstacle entre le roi et la pièce qui le met en échec sans se démasquer ;
- il n'est pas nécessaire de vérifier une situation de *pat*.

Chaque coup joué par un joueur donnera lieu à un certain nombre de vérifications :

- vérifier la validité de la nouvelle position par rapport à l'ancienne (**DOIT ETRE REALISE A L'AIDE DE CONTRAINTES SUR LE SCHEMA RELATIONNEL**) ;
- vérifier l'absence d'obstacle sur le déplacement, sinon le coup est invalide ;
- vérifier si le coup entraîne la prise d'une pièce adverse ;

1 Cf. [http://fr.wikipedia.org/wiki/%C3%89checs#R.C3.A8gles\\_du\\_jeu](http://fr.wikipedia.org/wiki/%C3%89checs#R.C3.A8gles_du_jeu)

- vérifier si le joueur ne se met pas en échec en jouant ce coup (démasquage d'une pièce adverse), sinon le coup est invalide ;

Ne pas oublier d'enregistrer le coup dans l'historique de la partie (coups numérotés dans l'ordre chronologique pour une partie et comprenant la position initiale et la position finale de la pièce manipulée par le coup).

### c) Rejouer une partie

Une partie jouée et terminée lors d'un tournoi doit pouvoir être rejouée pas à pas dans son intégralité. Il faut pour cela, conserver l'historique des coups au fur et à mesure qu'une partie se joue. On ne demande pas de pouvoir reculer dans l'historique d'une partie, ni de permettre de jouer une variante à partir d'un certain coup, mais juste de rejouer la partie pour l'analyser.

## 3 – Travail à réaliser (15h encadrées)

### 3.1 – Analyse

Cette étape consiste à analyser le problème posé.

- Réaliser l'analyse dynamique**, en identifiant les différents cas d'utilisation. Faire un diagramme de cas d'utilisation. Documenter le fonctionnement global du système par un ou plusieurs diagrammes états/transitions pertinents. Documenter les cas d'utilisation par des explications, *si nécessaire* illustrées par un ou plusieurs diagrammes de séquence.
- Réaliser l'analyse statique** : inventorier les données, identifier les dépendances fonctionnelles et les contraintes (invariants) entre les données persistantes.

### 3.1 – Conception de bases de données

- Élaboration du schéma conceptuel**. Construire, à partir de l'analyse statique un schéma Entités/Associations (ce schéma correspond à la modélisation objet du domaine). Expliquer ce schéma (notamment les points difficiles de sa construction). Préciser les contraintes d'intégrité non représentées dans le schéma Entités/ Associations.
- Conception de la base de données**. Traduire le schéma Entités/Associations en relationnel ; écrire les tables et les contraintes. Préciser et justifier la forme normale des relations, les contraintes d'intégrité implantées, ainsi que les contraintes ne pouvant être implantées en SQL (et qui sont donc à vérifier dans le code applicatif Java).
- Analyse des accès à la base de données**. Écrire les requêtes nécessaires aux différentes fonctionnalités à implanter. Délimiter les transactions permettant d'implanter les fonctionnalités tout en préservant la cohérence globale des données.

### 3.3 – Conception de l'application

- Architecture de l'application**. Proposer une architecture pour l'application. Préciser en particulier l'interface utilisateur ainsi que les liens avec la base de données. Proposer un diagramme de collaboration ou un diagramme de séquences illustrant un scénario typique pour :
  - une action de l'utilisateur sur l'interface.
  - le déclenchement et l'exécution d'une requête depuis l'application.
- Conception objet**. Faire un diagramme de classes logicielles.

### 3.4 – Implémentation

- a) **Instancier la base de données.** Elle doit être créée sur le serveur Oracle 11g disponible sur le serveur *ensibm*.
- b) **Programmer l'application en Java.** L'accès à la base de données se fera avec l'API JDBC.

### 3.5 – Bilan

Rédiger un bilan de ce projet (répartition des tâches, difficultés rencontrées, leçons retenues, *etc.*). Pour vous aider dans cette tâche, il vous est demandé de maintenir **un journal** de projet dans lequel vous préciserez (en quelques lignes) ce qui a été réalisé au cours de chaque séance encadrée de projet (ou en dehors de ces séances). Ce journal devra être déposé sur l'application *Teide* à la fin de chaque séance encadrée (déposé, mais non validé) et pourra être consulté à tout moment par vos encadrants tout au long du projet.

La **documentation finale** à rendre comporte quatre parties : l'analyse, la conception et l'implantation de la base de données, la conception de l'application, et le bilan.

## 4 – Soutenance (Vendredi 17 mai 2013)

Votre projet donne lieu à une soutenance de 20 à 25 minutes (inscription sur les créneaux horaires via l'application *Teide*), suivie de 10 à 15 minutes de questions, durant laquelle vous devrez :

- présenter votre modélisation entités-associations et son implantation en relationnel
- présenter votre architecture
- présenter le diagramme de classes logicielles
- faire une démonstration des fonctionnalités ; prévoir un scénario montrant le bon fonctionnement des transactions
- faire un bilan du déroulement du projet

L'objectif est de vendre **votre produit** (en montrant qu'il fait bien ce qu'on attend) **et votre équipe** de projet (en montrant que le produit a été bien construit).

**Note :** Il est très fortement recommandé de consulter régulièrement les pages du kiosk consacrées à ce projet.