

Projet Bases de Données

1 – Introduction

Le projet auquel vous allez participer a pour but de mettre en œuvre vos compétences en systèmes de gestion de bases de données relationnelles. De plus, vous vous essayerez à la programmation d'application utilisant une base de données (à travers des transactions). Le développement du démonstrateur sera fait en Java en utilisant l'API JDBC.

Le projet est à faire en **équipes de cinq** et donnera lieu à une réunion de suivi intermédiaire, ainsi qu'à une soutenance en fin de projet. La constitution des équipes et la remise des livrables (rapport final et code source) se fera sur l'application *Teide*.

L'évaluation se fera uniquement sur les aspects bases de données du projet en l'état au moment de la soutenance.

2 – Description de l'application

La société de commerce électronique destinée aux particuliers *Gange* souhaite mettre en place une base de données pour pouvoir mieux gérer les produits qu'elle propose à la vente. Les ventes se basent sur un mode d'enchères simple : les utilisateurs font des offres croissantes sur les produits disponibles et la 5^{ème} offre l'emporte. Le mode d'enchère n'a que peu d'importance, il doit pouvoir être changé facilement, et n'est pas connu des clients de *Gange*. Les produits sont catégorisés et un point important est la recommandation de catégories de produits en fonction des offres qui ont été faites précédemment par les utilisateurs.

Les produits ont un identifiant unique et sont décrits par un intitulé, un prix courant, un petit texte de description, l'URL d'une photo du produit, et un ensemble de couples caractéristique/valeur qui lui sont spécifiques (par exemple : <'taille', '44'> et <'matière', 'cuir'> pour une paire de chaussures). Un produit est classé dans une seule catégorie, identifiée par son nom. Une catégorie peut également faire partie d'une autre catégorie (par exemple : la catégorie « chaussures de sport » est une sous-catégorie de « chaussures »).

Les comptes utilisateurs ont un identifiant unique. Les clients sont identifiés par leur adresse email, un mot de passe (en clair, nous ne regardons pas les aspects sécurité ici), leur nom, leur prénom et leur adresse postale (pour les expéditions de produits). Un client peut parcourir le catalogue des produits, soit en se basant sur l'organisation des catégories, soit en suivant les catégories recommandées qui lui sont présentées. Un utilisateur peut faire une offre de prix sur un produit. Cette offre doit obligatoirement être plus élevée que le prix courant du produit concerné. Les offres sont identifiées par l'identifiant de produit et la date et l'heure de l'offre, et on précisera le prix proposé et le compte utilisateur concerné. Si l'offre est validée par l'utilisateur et l'application, alors le prix courant du produit devient le prix qui a été proposé. La cinquième offre emporte

l'enchère (encore une fois : les clients de *Gange* ne connaissent pas le principe d'enchères). L'achat est matérialisé en enregistrant la correspondance entre le produit et l'offre gagnante.

Les recommandations de catégories se basent sur l'historique d'offre et d'achat des utilisateurs. En priorité, les recommandations vont concerner **les catégories** pour lesquelles l'utilisateur a fait le plus d'offres sur des produits sans réussir à les acheter (par ordre décroissant du nombre d'offres). Ensuite, les recommandations concerneront **les catégories** pour lesquelles il y a eu le plus d'offres en moyenne par produit (avec ou sans achat) et ce quel que soit l'utilisateur (par ordre décroissant du nombre moyen d'offres par produit).

Nous devons enfin prendre en compte la RGPD, en particulier le droit à l'oubli. De ce fait, si un client demande à ce que l'on supprime ses données personnelles il est important de quand même conserver les offres qui ont été effectuées. En effet, il reste primordial pour *Gange* de savoir qu'un même client a effectué (et éventuellement remporté) un certain nombre d'enchères, même si nous ne savons plus qui il est.

3 – Travail à réaliser (15h encadrées + travail personnel, le tout ×5)

Le travail se découpe en quatre étapes, décrites ci-dessous, dont seules les deux premières sont obligatoirement séquentielles. L'étape 3 peut être découpée et parallélisée, et l'étape 4 peut être commencée dès le début du projet.

3.1 – Modélisation du problème

La modélisation se décompose en deux temps.

Dans un premier temps, vous aurez à **analyser le problème posé** pour en extraire les propriétés élémentaires, les dépendances fonctionnelles reliant ces propriétés, ainsi que tous les autres types de contraintes (contraintes de valeur, contraintes de multiplicité et contraintes contextuelles).

Vous devrez **proposer ensuite un schéma Entités/Associations** représentant les données nécessaires à l'application et leurs liens sémantiques (ce qui correspond à l'état cohérent de la base de données).

Le schéma Entités/Associations doit être totalement justifié par l'analyse. Expliquez vos choix de modélisation dans la documentation du projet.

3.2 – Implantation de la base de données

Vous devrez ensuite traduire le schéma Entités/Associations en un **schéma relationnel** décrivant les noms des relations obtenues, les noms et types de leurs attributs, ainsi que les contraintes à vérifier pour chacune des relations. **Vous devrez également justifier vos choix de traduction et expliquer les points difficiles.** Vous préciserez et justifierez la **forme normale de chacune des relations obtenues**.

Vous implanterez ensuite ce schéma relationnel sur le **SGBD Oracle disponible sur le serveur oracle1**. Vous devrez insérer suffisamment de données pertinentes pour la suite du projet.

3.3 – Analyse des fonctionnalités

Vous devrez définir les **requêtes SQL2** nécessaires pour réaliser les fonctionnalités suivantes en les **regroupant en transactions** maintenant la cohérence de la base de données conformément au schéma Entités/Associations et aux contraintes associées :

- **Parcours du catalogue**

Cette fonctionnalité inclura toutes les étapes nécessaires au parcours du catalogue :

1. Connexion du client à l'application avec son email comme identifiant et vérification du mot de passe.
2. Parcours des catégories et sous-catégories, ou des catégories recommandées, au choix. Les produits des catégories seront affichés par ordre décroissant du nombre d'offres qui ont été faites et par ordre alphabétique (un seul tri multicritère).
3. Consultation de la fiche complète de chacun des produits sélectionnés.

- **Enchères**

1. Proposition d'une offre de prix sur un produit disponible à la vente en respectant les règles décrites plus haut (cf. Section 2).
2. Si l'offre est la cinquième sur ce produit, conclure l'achat. Le produit n'est, dès lors, plus à la vente.

- **Droit à l'oubli**

Cette fonctionnalité doit permettre à un utilisateur de supprimer ses données personnelles (identifiant de compte, email, mot de passe, nom, prénom et adresse postale). Il ne pourra alors plus se connecter à notre système. Néanmoins les offres et achats déjà effectués par l'utilisateur en question devront être conservés et être associés à un nouvel identifiant de compte créé pour l'occasion.

Ces **requêtes et transactions doivent être testées** sur Oracle (**avec SQL*Plus**, Adminer ne gérant pas les transactions) pour en vérifier leur bon fonctionnement, **y compris pour des exécutions concurrentes**.

3.4 – Implantation des fonctionnalités

Les fonctionnalités précédemment étudiées devront être implantées dans un **démonstrateur programmé en Java/JDBC**. Nous vous recommandons l'implantation d'une **interface textuelle simple** (quelques menus et saisies utilisateurs/affichages). **L'évaluation ne portera que sur les aspects Base de Données** et pas sur le code Java.

4 – Déroulement du projet

Le projet sera constitué de 15 heures en séances encadrées et 3 heures réservées pour les soutenances.

4.1 – Séance encadrées

Comme toute épreuve pratique à l'Ensimag, **la présence aux séances encadrées est obligatoire et sera contrôlée**.

La première séance comprendra une présentation du projet, de JDBC et de l'analyse de fonctionnalités. Le travail se fera ensuite par équipe (créées préalablement via Teide). Vous pourrez poser des questions à votre encadrant à tout moment lors des séances, mais il jouera deux rôles alternativement : il sera **soit votre client** (qui connaît les besoins de *Gange*, mais ne connaît rien en base de données), **soit un expert en bases de données** (qui ne connaît pas et ne veut pas connaître les besoins applicatifs).

Posez bien vos questions 😊

4.2 – Outils

Vous disposez de trois outils principaux pour le bon déroulement du projet :

- **Chamillo** : vous y trouverez les documentations techniques pour accéder à Oracle et pour utiliser et vous procurer JDBC (exemple de code, documentation, etc.), des liens Internet utiles, ainsi qu'un forum dans laquelle vous trouverez les réponses aux questions fréquentes (à consulter souvent, donc).
- **Riot/Element** : outil hors-séance privilégié pour interagir avec votre encadrant (<http://element.ensimag.fr>).
- **Teide** : l'application de gestion de projet. Vous devrez utiliser Teide pour constituer vos équipes, déposer vos rendus (documentation du projet à déposer à chaque séance sans la valider, code source Java et SQL2, et supports pour la soutenance en fin de projet).

4.3 – Suivi

Le projet donnera lieu à une **réunion de suivi intermédiaire** entre l'encadrant et chacune des équipes (~4ème séance encadrée). Cette réunion sera provoquée par les équipes elles-mêmes ou par l'encadrant en fonction de l'avancement du travail. Les thèmes discutés lors du suivi concerneront **l'analyse et la modélisation Entités/Associations**.

4.4 – Livrables

Vous aurez trois livrables à fournir pour ce projet :

- **Documentation du projet** (un fichier PDF): Vous devrez maintenir la documentation du projet tout au long de son déroulement. La documentation doit comprendre :
 - L'analyse du problème sous forme de contraintes (DF, de valeur, de multiplicité, contextuelles). Expliquez vos choix sur les points difficiles.
 - La conception Entités/Associations noté en UML, comme vu en cours, en expliquant vos choix. Précisez bien les contraintes non représentées dans le schéma. Le schéma peut être réalisé avec l'outil Dia (disponible sur toutes les plateformes), n'importe quel outil de dessin vectoriel, voire même sur papier puis scanné, tant qu'il est lisible et respecte les notations graphiques vues en cours.
 - Sa traduction en relationnel en précisant les formes normales des relations et en justifiant vos choix. Les contraintes non implantables en relationnel devront également être listées.
 - L'analyse des fonctionnalités (transactions), leur implantation sous forme de requêtes SQL2, le tout bien commenté.
 - Un bilan du projet (organisation, points difficiles rencontrés, etc. en prenant du recul sur le travail effectué).

Un petit mode d'emploi de votre démonstrateur est également le bienvenu.

La qualité et la complétude de la documentation seront prises en compte dans la notation.

- **Sources Java et SQL2** (un fichier ZIP ou TGZ) : Vous devrez rendre en fin de projet un script SQL2 permettant de créer votre schéma relationnel, un script SQL2 permettant de peupler la base de données, l'implantation des fonctionnalités (squelette des transactions et requêtes SQL2), ainsi que le code source Java du démonstrateur.
- **Supports pour la soutenance** : Les slides prévus pour la soutenance devront être déposés sur Teide (visez 15 à 20 minutes).

4.5 – Soutenance finale

La soutenance **durera en tout 30 minutes** divisées de la façon suivante :

- 15 à 20 minutes pour que vous présentiez votre projet et faire une démonstration bien préparée (scénario, jeux de données en place, etc.) ;

- 10 à 15 minutes pour que le jury puisse vous faire un retour sur le projet et vous poser des questions. L'objectif de la soutenance est double : vous devez **convaincre le client que le produit répond bien à ses besoins**, et vous devrez **convaincre l'expert que votre équipe a fait du bon travail**. La soutenance doit donc reprendre les différents points de la documentation en insistant sur les points que vous jugerez les plus pertinents.

5 – Conclusion

Le projet se déroule en quatre semaines, soutenances incluses. **Cela va aller très vite. Une bonne organisation de l'équipe est primordiale.** Il est recommandé de **diviser les tâches dès le début du projet** et de toujours **garder les « candides » par tâche** (des personnes qui ne participent pas du tout à cette tâche), de façon à ce qu'ils puissent apporter **un regard critique** sur ce qui a été fait lors des réunions d'équipe.