

Analyste Programmeur

Projet Java

Laurent HUBERT

2017

Table des matières

| | | |
|----------|---|----------|
| 1 | Introduction | 1 |
| 2 | Contexte | 2 |
| 3 | Travail demande | 2 |
| 3.1 | Description generale | 2 |
| 3.2 | Description des modules | 3 |
| 3.2.1 | Module d'authentification en ligne de commande | 3 |
| 3.2.2 | Module d'exportation de données | 3 |
| 3.2.3 | Module d'importation de données | 4 |
| 3.2.4 | Module d'authentification via une interface graphique | 4 |
| 3.2.5 | Module de gestion | 4 |
| 3.3 | Structure du programme | 5 |
| 3.3.1 | Découpage par packages | 5 |
| 3.3.2 | Conventions de codage | 5 |
| 3.3.3 | Tests | 5 |
| 4 | Présentation du projet | 5 |
| 4.1 | Présentation | 5 |
| 4.2 | Mémo Technique | 6 |
| 4.3 | Fichiers | 6 |

1 Introduction

Ce projet reprend le travail fait sur la base de données CPA.

L'objectif de ce projet est de vous faire réaliser différents modules permettant l'importation, l'exportation, l'utilisation et l'exploitation des données de la base de données que vous avez réalisée.

2 Contexte

Vous et votre équipe avez mis en place une nouvelle base de données pour l'entreprise CPA.

Cette base de données, bien qu'opérationnelle n'est pas très utile sans la possibilité de manipuler facilement les données qui y sont stockées.

D'autant plus que le prototype de l'outil que vous avez fourni ne répond pas à tous les besoins actuels de la société.

3 Travail demande

3.1 Description generale

Vous allez commencer la réalisation d'un outil indépendant permettant la manipulation des données de votre BDD

Vous allez donc réaliser un prototype pour ces différents modules (dans la mesure de vos possibilités) :

1. un module d'authentification via une interface en ligne de commande
2. un outil d'exportation/sauvegarde des données
3. un outil d'importation de données
4. un module d'authentification via une interface graphique
5. un module au choix permettant de gérer soit :
 - ▷ les contacts
 - ▷ les données des utilisateurs
 - ▷ les factures
 - ▷ les stocks
 - ▷ ou toute autre donnée

Votre temps étant limité, vous devrez privilégier la qualité à la quantité : mieux vaut rendre un module le plus complet possible et le fiabiliser plutôt que tenter de tout faire avec rien qui ne fonctionne correctement.

C'est un prototype : une solution optimale n'est pas demandée. Un questionnement et une réflexion sur ce que pourrait être cette solution optimale l'est. Ce questionnement ne doit pas prendre le pas sur le développement du prototype.

Ces modules seront réalisés exclusivement à l'aide du langage Java : ils s'appuieront sur les requêtes SQL que vous avez déjà pu développer précédemment et que vous intégrerez dans des classes gérant l'accès au données.

Avant de commencer le développement, il vous est recommandé de réaliser un diagramme de classes décrivant l'organisation du module à développer.

3.2 Description des modules

3.2.1 Module d'authentification en ligne de commande

Ce module permettra à un utilisateur d'acquérir les droits pour l'accès et la modification des informations de la base de données.

Il demandera à l'utilisateur un login et mot de passe et affichera une réponse claire sur l'état de la connexion.

Ce module et/ou ses classes sera réutilisé dans les modules d'importation et d'exportation de données afin de garantir la sécurité des opérations d'import/export.

Vous êtes libre de choisir la méthode de votre choix :

- ▷ stockage des identifiants dans un fichier local
- ▷ stockage dans la base de données
- ▷ utilisation du système d'authentification MySQL
- ▷ autre

La sécurisation des informations pourra se faire via un chiffrement des données. Vous êtes libres de choisir la méthode qui vous semble la plus pertinente.

3.2.2 Module d'exportation de données

Ce module permettra de réaliser une sauvegarde complète et/ou partielle de la base de données au format CSV.

Le résultat de l'opération sera soit un seul, soit plusieurs fichiers. Vous êtes libres de choisir la structure qui vous convient.

Avant de réaliser toute opération, l'utilisateur devra s'authentifier.

L'utilisateur pourra préciser l'emplacement des fichiers cibles : soit en passant un paramètre à la ligne de commande, soit en répondant à une question du programme.

En cas de sauvegarde partielle, l'utilisateur pourra soit choisir la partie à sauvegarder. Si ce n'est pas le cas, la documentation devra préciser clairement quelles parties sont sauvegardées et lesquelles ne le sont pas.

La première ligne du fichier contiendra des noms de colonnes. Ces noms peuvent permettre d'associer les valeurs de la colonne correspondante à un champ d'une des tables de la base.

3.2.3 Module d'importation de données

Ce module permettra à un utilisateur d'importer un jeu de données en utilisant la ligne de commande. Cependant, les classes de ce module pourraient être réutilisées dans une interface graphique (à vous de bien organiser vos classes pour qu'elles soient réutilisables)

Avant de réaliser toute opération, l'utilisateur devra s'authentifier.

Les données en entrée seront issues d'un fichier Excel (format CSV) et/ou du module d'exportation de données (vous pouvez choisir parmi ces deux possibilités ou les deux).

L'utilisateur pourra préciser l'emplacement des fichiers cibles : soit en passant un paramètre à la ligne de commande, soit en répondant à une question du programme.

La première ligne du fichier contiendra les noms de colonnes. Ces noms peuvent permettre d'associer les valeurs de la colonne correspondante à un champ d'une des tables de la base.

Si possible, vous tiendrez compte du fait que l'ordre des colonnes peut varier.

Le type des données importées pourra être déterminé soit à partir du contenu du fichier (par rapport aux noms des colonnes par exemple), soit à partir d'un paramètre passé au programme (paramètre de la ligne de commande).

Le module devra fournir un compte-rendu précis de la réussite ou de l'échec de l'opération.

3.2.4 Module d'authentification via une interface graphique

Avant de réaliser ce module (et si ce n'est pas déjà fait), vous en réaliserez une maquette graphique.

Ce module reprend les mêmes fonctionnalités que celui en ligne de commande, à ceci près qu'il s'utilise via une interface graphique.

Les contraintes sont donc les mêmes.

Vous veillerez à ce que les informations, avertissements et messages d'erreur soient clairement affichés.

3.2.5 Module de gestion

Avant de réaliser ce module (et si ce n'est pas déjà fait), vous réaliserez une maquette graphique du module choisi parmi ceux décrits précédemment (paragraphe Description générale)

Ce module au choix reprendra une maquette graphique que vous aurez pu réaliser précédemment.

Ce module utilisera le module d'authentification graphique développé précédemment pour sécuriser l'accès aux données.

3.3 Structure du programme

3.3.1 Découpage par packages

Vous veillerez à l'organisation de vos classes sous forme de packages.

Vous devrez fournir une description succincte de cette organisation avec pour chaque package les fonctionnalités et domaines qu'il couvre.

Les classes d'accès aux données (celles qui réalisent les opérations d'accès à la BDD) seront clairement identifiées.

Les classes servant à la représentation et à la manipulation des données seront également identifiées.

3.3.2 Conventions de codage

Vous respecterez les conventions de nommage standards de Java.

Si vous faites le choix d'une convention particulière, vous l'expliquerez et la justifierez.

Les commentaires de génération de la Javadoc devront être au minimum présents.

3.3.3 Tests

Les tests réalisés (manuels ou automatiques) feront l'objet d'une description permettant de les rejouer sur n'importe quelle machine. Il vous est recommandé, avant la présentation, de lancer ces tests sur une machine n'ayant pas servi au développement du projet.

4 Présentation du projet

4.1 Présentation

Vous ferez une démonstration de votre programme. Vous en ferez une présentation orale. Le jury pourra éventuellement vous demander de lui montrer certaines parties de votre code et chaque membre de votre équipe pourra se voir poser quelques questions sur la partie qu'il a réalisée.

4.2 Mémo Technique

Vous rassemblerez également dans un mémo technique les éléments suivants :

- ▷ Les diagrammes des classes et autres schémas explicatifs sur le fonctionnement de votre programme ;
- ▷ La description de l'organisation des classes et packages ;
- ▷ Une description des tests réalisés ;
- ▷ Pour chaque module un mode d'emploi indiquant comment exécuter les commandes ou les fichiers ;
- ▷ La méthode de sécurisation des informations choisie et une critique de celle-ci (avantages, inconvénients, forces, faiblesses) ;
- ▷ Une maquette graphique des modules d'authentification et du module graphique que vous aurez choisis.

4.3 Fichiers

Vous fournirez également :

- ▷ Le résultat de la génération de la Javadoc ;
- ▷ Les jeux de tests (fichiers utilisés pour vos tests).