

Rapport de projet

Application Java de gestion de stages

Groupe G6 : ANTOINET Dorian, DELPY Quentin et SCHINELLA Johan

Table des matières

Rapport de projet	1
Introduction.....	2
I – Interface	3
1) Connexion.....	3
2) Inscription.....	3
3) Page principale	3
4) À propos	3
II – Base de données.....	4
1) Table « Utilisateur »	4
2) Table « Entreprise ».....	4
3) Table « OffreStage ».....	4
III – DAO.....	4
1) Classe « Dao ».....	4
2) Relation entre les classes « Utilisateur » et « UtilisateurDao »	5
3) Relation entre les classes « Entreprise » et « EntrepriseDao ».....	5
4) Relation entre les classes « OffreStage » et « OffreStageDao »	5
IV – xxx.....	5
V – xxx.....	5
Conclusion	5

Introduction

I – Interface

1) Connexion

Lors du lancement de l'application, l'utilisateur a accès à une interface de connexion à partir de laquelle il peut se connecter ou s'inscrire.

Pour se connecter, l'utilisateur doit saisir son identifiant et son mot de passe dans les champs correspondants. Si l'identifiant ainsi que le mot de passe saisis sont corrects, l'utilisateur sera redirigé vers la page principale du projet. Sinon, un message d'erreur apparaîtra et la connexion sera annulée.

2) Inscription

Lors de l'inscription, l'utilisateur saisit son identifiant et son mot de passe. Il choisit également son statut : « étudiant(e) » ou « entreprise ». Ensuite, le nouveau compte créé est inséré dans la base de données et l'utilisateur est redirigé vers l'interface de connexion.

En cas d'erreur (par exemple, un identifiant déjà pris ou un champ vide), un message d'erreur apparaît et indique la raison de l'échec de l'inscription.

3) Page principale

Le menu principal change en fonction du type d'utilisateur (étudiant(e) ou entreprise). Un seul point commun : tout le monde peut consulter les offres de stage et accéder au menu « À propos ».

Les étudiants ont accès à un menu leur permettant de consulter les offres de stage proposées par les entreprises. Ils peuvent y voir le nom des entreprises, le lieu du stage, l'adresse e-mail du recruteur, l'intitulé et le domaine de l'offre, la durée (en jours) ainsi qu'un descriptif.

Il est possible de changer l'offre affichée sur la page à l'aide de boutons « suivant » et « précédent » en bas de la fenêtre. On peut également postuler à l'offre avec le bouton « postuler ».

Les entreprises ont, elles, accès à deux menus supplémentaires : ils leurs servent à créer de nouvelles offres de stages et à insérer des entreprises dans la base de données.

Pour créer une offre de stage, il faut saisir le nom de l'entreprise, le domaine de l'offre, l'intitulé, la date de début du stage, la durée (en jours) et un descriptif.

Pour compléter la création de l'offre de stage, il suffit d'appuyer sur le bouton « »

Pour créer une entreprise dans la base de données, il faut saisir le nom, l'adresse (rue et numéro), la ville et son code postal, l'adresse e-mail du recruteur de l'entreprise ainsi que son numéro de téléphone, et enfin le secteur d'activité de l'entreprise.

Une fois la saisie effectuée, l'utilisateur en mode entreprise pourra s'enregistrer dans la base de données et ainsi être accessible par les autres utilisateurs.

4) À propos

Ce menu permet d'ouvrir une fenêtre surgissante qui donne des informations générales sur le projet, notamment l'équipe de développement ainsi qu'un lien GitHub.

II – Base de données

1) Table « Utilisateur »

Elle définit les utilisateurs qui se connectent pour accéder aux offres de stage. Un utilisateur est caractérisé par un numéro d'identifiant unique, son identifiant de connexion et son mot de passe crypté. Est également stocké un booléen qui détermine si l'utilisateur est un(e) étudiant(e) ou une entreprise.

Utilisateur
+ id : integer
+ login : string
+ pass : string
+ estEntreprise : boolean

2) Table « Entreprise »

Il s'agit des entreprises ajoutées via l'application par des utilisateurs « entreprise ». Une entreprise est caractérisée par un numéro d'identifiant unique, une raison sociale, une adresse e-mail, une rue, une ville, un code postal, un numéro de téléphone et un secteur d'activité.

Il s'agit d'une entité qui possède des droits similaires à un utilisateur. Elle peut en plus proposer de nouvelles offres de stage et créer une nouvelle entreprise.

Entreprise
+ id : integer
+ raisonSociale : string
+ mail : string
+ ville : string
+ rue : string
+ codePostal : string
+ tel : string
+ secteurActivite : string

3) Table « OffreStage »

La table « OffreStage » contient les offres de stage émises par les entreprises. Une offre de stage est caractérisée par un numéro d'identifiant unique, un libellé, une description du stage, le domaine du stage, la date de début, la durée et enfin un booléen qui indique si l'offre de stage est encore valide ou non.

OffreStage
+ id : integer
+ libelle : string
+ description : string
+ domaine : string
+ dateDebut : date
+ duree : integer
+ estValide : boolean

III – DAO

Le patron de conception DAO (« Data Access Object », « objet d'accès aux données ») a été utilisé pour les requêtes SQL, ce qui permet de faciliter leur appel ainsi que leur utilisation, tout en séparant des responsabilités qui ajoutent une couche d'abstraction au niveau de la persistance des entités. Le DAO est notamment utilisé dans les architectures logicielles objet.

1) Classe « Dao »

La classe abstraite « Dao » implémente les principales fonctions en lien avec la base de données utilisées par les sous-classes. On y retrouve les principales fonctions de recherche, d'ajout, de mise à jour et de suppression d'objet depuis un paramètre tel que l'objet ou son identifiant :

Méthode de recherche : `public abstract T find (int id)`

Méthode de création : `public abstract T create (T objet)`

Méthode de mise à jour : `public abstract boolean update (T objet)`

Méthode de suppression : `public abstract boolean delete (T objet)`

2) Relation entre les classes « Utilisateur » et « UtilisateurDao »

La classe « Utilisateur » contient les attributs qui permettent de gérer une entité d'utilisateur. Il s'agit du numéro d'identifiant unique, de l'identifiant de connexion, du mot de passe, du statut (administrateur, étudiant(e) ou entreprise), et, en fonction du statut, la liste des offres de stages auxquelles l'utilisateur a postulé ou alors la liste des entreprises qu'il a créées.

On emploiera différents constructeurs pour l'entité en fonction de la situation (inscription, connexion...).

L'entité « Utilisateur » dispose de deux méthodes : une pour postuler à une offre de stage, et une autre pour créer une entreprise.

La classe « UtilisateurDao » permet d'exécuter les principales requêtes liées aux entités « Utilisateur ». Certaines de ces requêtes sont initialisées dans la classe mère « Dao ». Parmi ces requêtes, en plus de celles déjà implémentées, on retrouve :

Recherche : `public Utilisateur find (String login)`

Récupération de liste d'utilisateurs : `public ArrayList<Utilisateur> findAll ()`

Postuler à une offre : `public boolean postuler (Utilisateur u, OffreStage o)`

Récupération de liste d'offres : `public ArrayList<OffreStage> findAlloffresStage (Utilisateur u)`

3) Relation entre les classes « Entreprise » et « EntrepriseDao »

4) Relation entre les classes « OffreStage » et « OffreStageDao »

IV – xxx

V – xxx

Conclusion