CHAPITREIV

Modélisation et conception

1. Introduction

Dans ce chapitre nous allons présenter l'implémentation de notre application et la démarche de chaque fonctionnalité réalisée, nous allons développer tout au long de cette étude une application pour une banque qui permet de faire entrer une autre consultation depuis internet grâce à une application Web utilisant les JSP. Nous allons également réaliser les fonctionnalités Administrateur et Agent de la banque. L'objectif est de proposer aux clients une application de gestion de leurs comptes en ligne en leurs facilitant d'effectuer leurs opérations avec toutes sécurités.

Nous avons opté pour une architecture 3-tiers afin d'implémenter un ensemble de contrôle d'accès concernant le client. Pour le rôle agent et administrateur, nous avons choisi une architecture 2-tiers. Dans ce qui suit, nous allons présenter les différentes fonctionnalités de l'application.

2. Réalisation de l'application

Notre application se compose d'une partie Web concernant l'accès client, et d'une partie classique liée à l'Agent et à l'Administrateur.

2.1. Application Web 3-tiers

Cette partie est réalisée à l'aide de JSPs. L'objectif étant de faire une gestion dynamique de l'ensemble de fonctionnalités liées aux clients. Dans ce qui suit, nous présenterons les différentes vues de l'application.

2.1.1 Page d'accueil

Toutes les pages sont construites de la même manière et utilisent les mêmes éléments, c'est-à-dire :

•Un corps qui affiche les formulaires à remplir ou les informations à afficher.

• Un menu qui se trouve sur la gauche de la page et qui contient un certain nombre d'informations.

La figure suivante représente la page d'accueil de notre application Web.

Figure IV.1 : Page d'accueil.

2.1.2. Accès client

La figure suivante représente la page utilisée pour l'authentification du client.

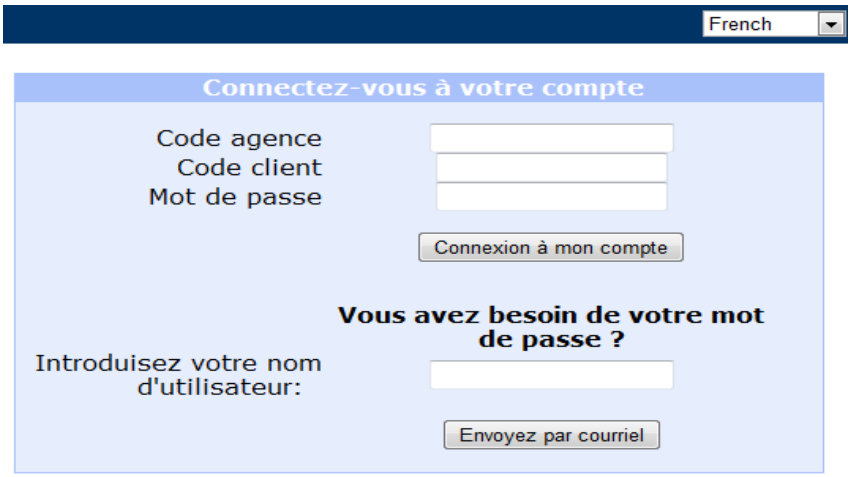


Figure IV.2 : Page d'authentification.

La page JSP vérifie si les codes correspondent aux informations d'un utilisateur existant dans la base de données. Cette page passe le contrôle à la page des opérations si le client a saisi correctement ses codes, sinon le contrôle reste dans la même page ou s'affiche un message d'erreur. Après trois essais erronés, le mot de passe sera désactivé pour des raisons de sécurité. C'est notre proposition de contrôle d'accès à ce niveau pour mieux gérer la sécurité de l'utilisateur. Maintenant et lorsque le client est connecté, l'application le redirige vers une autre JSP qui est représentée dans la figure suivante.

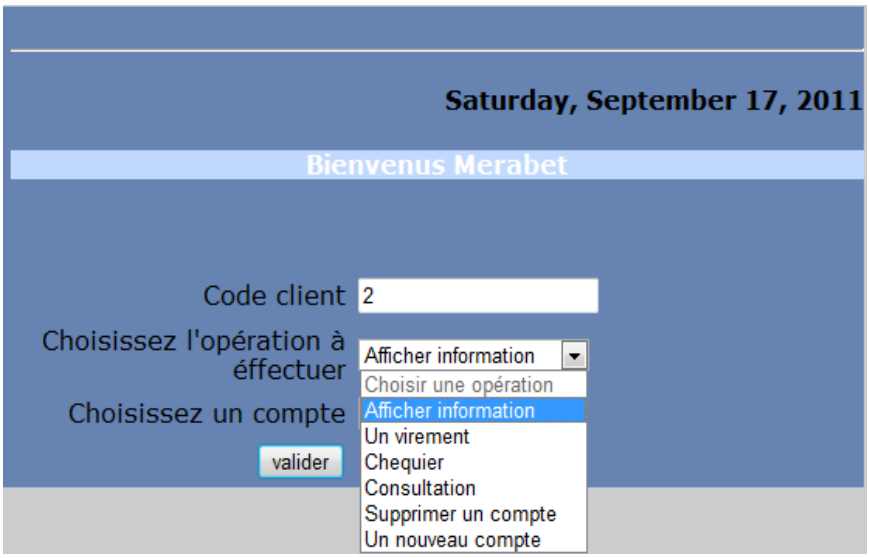


Figure IV.3 : Client authentifié.

Si le client choisit **« Afficher information »** la page suivante qui sera affichée :

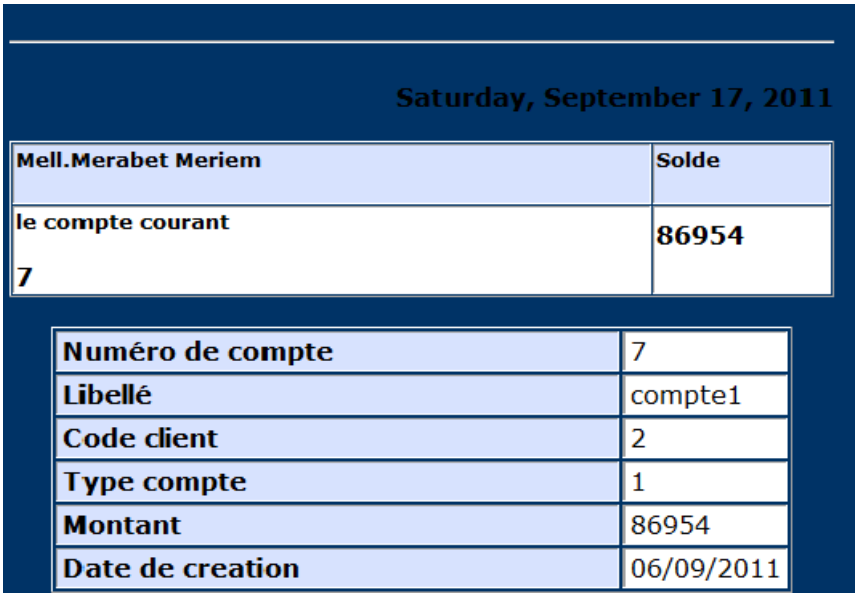
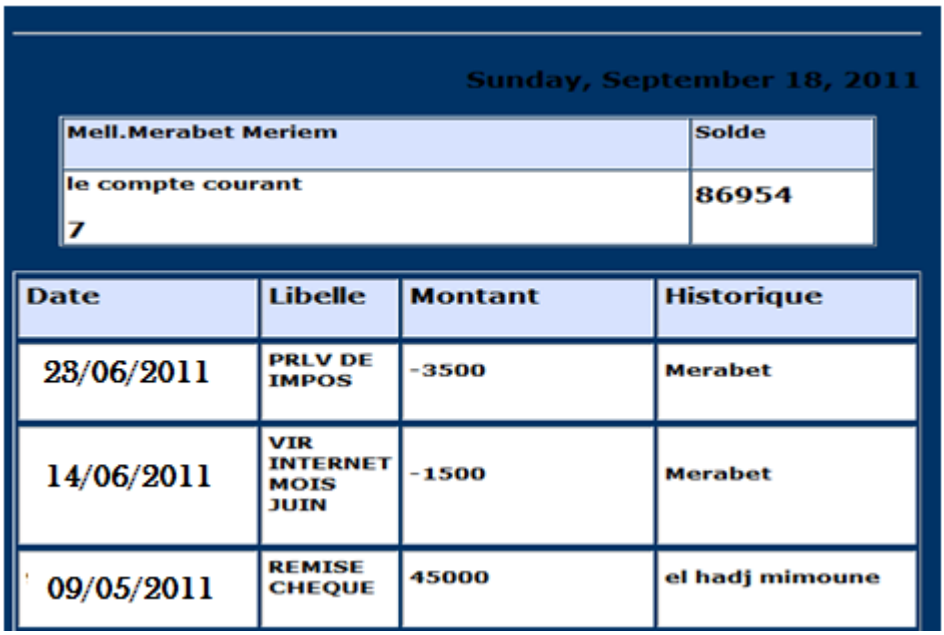


Figure IV.4 : Information d'un compte.

2.1.2.1 L'audit des accès de l'utilisateur

L'audit des accès est nécessaire pour déterminer l'utilisateur responsable de telle action dans la base de données, et comme les utilisateurs accèdent à partir d'un intermédiaire, il est difficile à un système audit de garder la trace et de corréler les activités qui peuvent être sensitives à la sécurité. Pour ce qui nous concerne, nous avons réalisé ce moyen de sécurité quand l'agent ou le client effectue une opération qui va être stocké dans la base de données et on a gardé la date ainsi que les différentes opérations nécessaires (libellé de l'opération, responsable d'opération, ..).

Donc lorsque le client choisit **« Consultation d'un compte »** la figure suivante sera affichée:



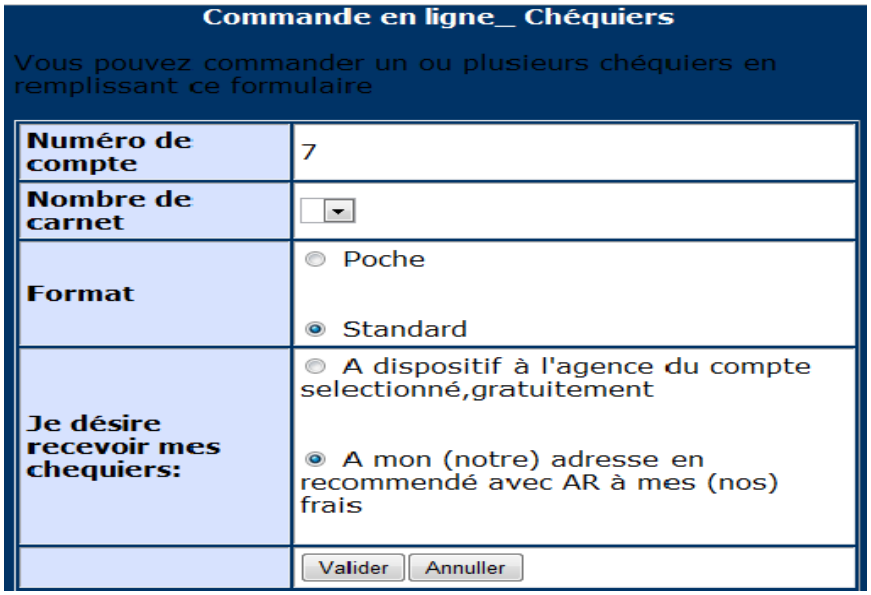
Sunday, September 18, 2011

Mell.Merabet Meriem	Solde
le compte courant 7	86954

Date	Libelle	Montant	Historique
23/06/2011	PRLV DE IMPOS	-3500	Merabet
14/06/2011	VIR INTERNET MOIS JUIN	-1500	Merabet
09/05/2011	REMISE CHEQUE	45000	el hadj mimoune

Figure IV.5 : Page d'historique.

Le client demande un chéquier par ce choix **« Demande chéquier »**, la figure suivante illustre cette opération :



Commande en ligne_ Chéquiers

Vous pouvez commander un ou plusieurs chéquiers en remplissant ce formulaire

Numéro de compte	7
Nombre de carnet	
Format	Poche Standard
Je désire recevoir mes chequiers:	A dispositif à l'agence du compte selectionné,gratuitement A mon (notre) adresse en recommendé avec AR à mes (nos) frais
	Valider Annuller

Figure IV.6 : Commande en ligne un chéquier.

Parmi les choix du client c'est choisir **« Un virement »** et le résultat sera:

Figure IV.7 : Effectuer un virement.

Pour les clients qui ne possèdent pas de compte, ils peuvent formuler leurs demandes via internet et à travers un formulaire bien spécifique. Le nouveau client est ajouté dans la base de données mais il reste toujours en attente jusqu'à la validation de l'administrateur.

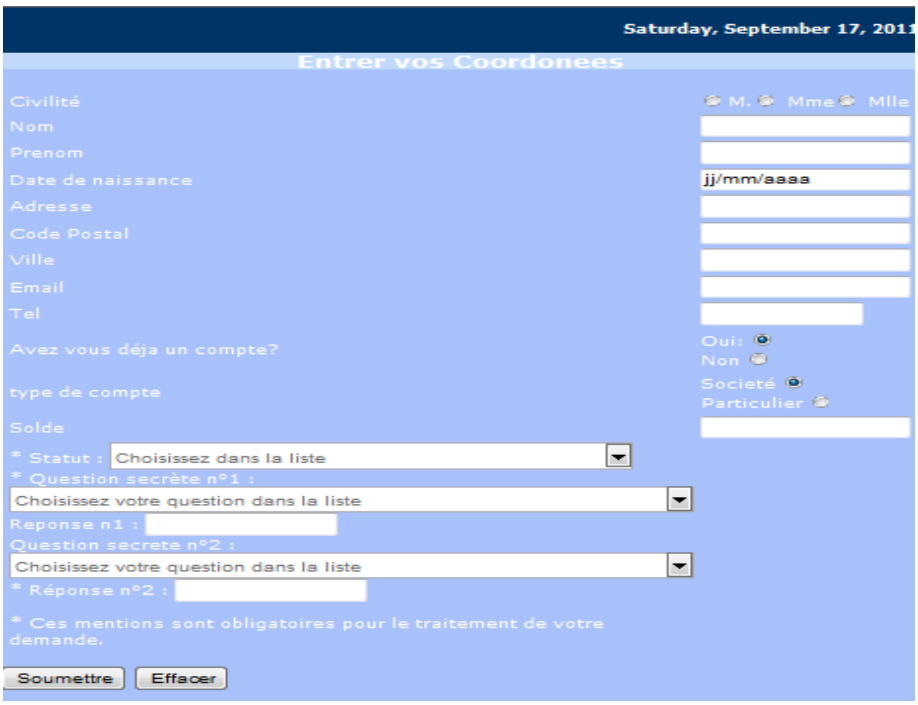


Figure IV.8 : Nouvelle inscription.

Lorsque le client demande de supprimer un compte, il va recevoir un message pour la confirmation de suppression et après la désactivation de ce compte, le message d'information suivant sera affiché :

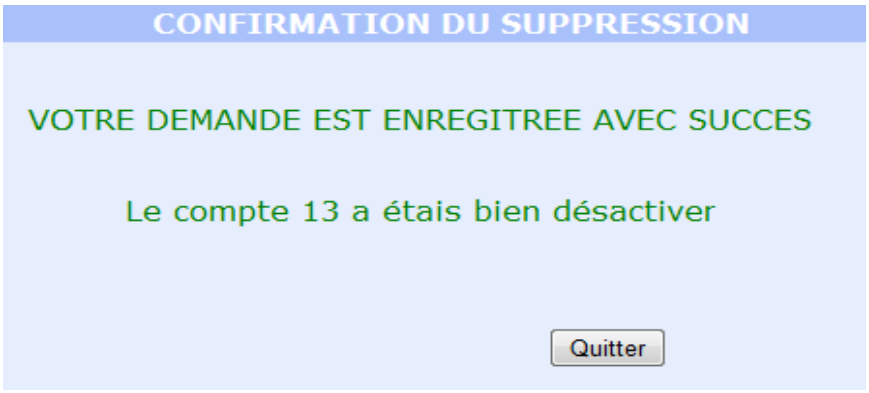


Figure IV.9 : Confirmation de suppression.

Le client peut cliquer sur le choix **trouver une agence**, une JSP affiche la liste des wilayas possédant des agences existantes dans la base de données. Puis une agence peut être sélectionnée parmi celles de cette wilaya.

Le choix de l'agence « Kiffane » de la wilaya de Tlemcen correspond à l'affichage suivant :

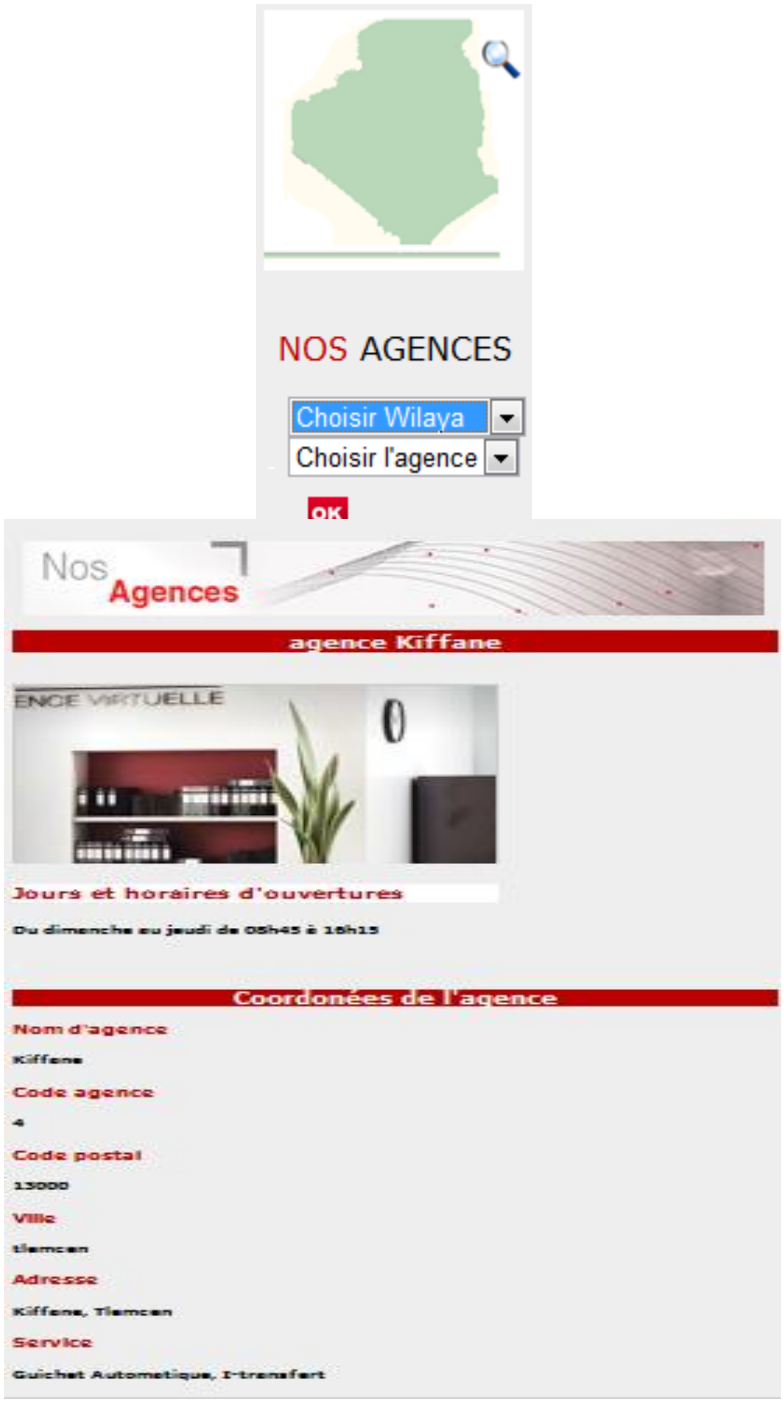


Figure IV.10 : Information sur une agence.

Dans ce qui précède nous avons présente notre première proposition de contrôle d'accès, le chiffrement qui sera détaillé dans la section suivante est transparente pour le client de notre application car c'est l'administrateur qui va crypter les mots de passes.

2.2. Application classique 2-tiers

Dans ce qui suit, nous présenterons notre deuxième partie de l'application qui concerne les fonctionnalités liées à l'Agent et à l'Administrateur. Mais avant tous s'intéressant à cette notion de rôle.

2.2.1 Rôle

Un rôle est le nom d'un groupe, tel que les gestionnaires. Après avoir établi des rôles bien spécifiques, on assigne l'administrateur et l'agent de la banque à des rôles bien défini. Puis, on accorde des autorisations à ces rôles. Chaque utilisateur appartenant à ce rôle hérite des autorisations qu'on a assignées. Les rôles sont par conséquent une façon efficace de gérer des autorisations pour les groupes d'utilisateurs. Le principal objectif des rôles consiste à faciliter la gestion des règles d'accès pour les groupes, tel que les administrateurs ou les agents. Une fois les rôles définis, nous pouvons créer les règles d'accès de notre application.

2.2.2 Accès de l'agent

Un agent dans une banque s'occupe de la relation avec le client. En fonction du profil du client, il va lui proposer des services adaptés à ses besoins et suivre l'évolution de ses opérations bancaires. Il suivra le client dans toutes ses démarches. L'agent a pour rôle de gérer le compte en banque des clients qui partagent avec lui le même code catégorie. Nous avons choisi cette clé afin d'implémenter le modèle par rôle.

La page d'identification est la plus simple du formulaire : c'est un formulaire de saisie avec trois champs, l'un pour choisir le code agence existant dans la BDD, l'autre pour la saisie du nom et le dernier pour la saisie du mot de passe.

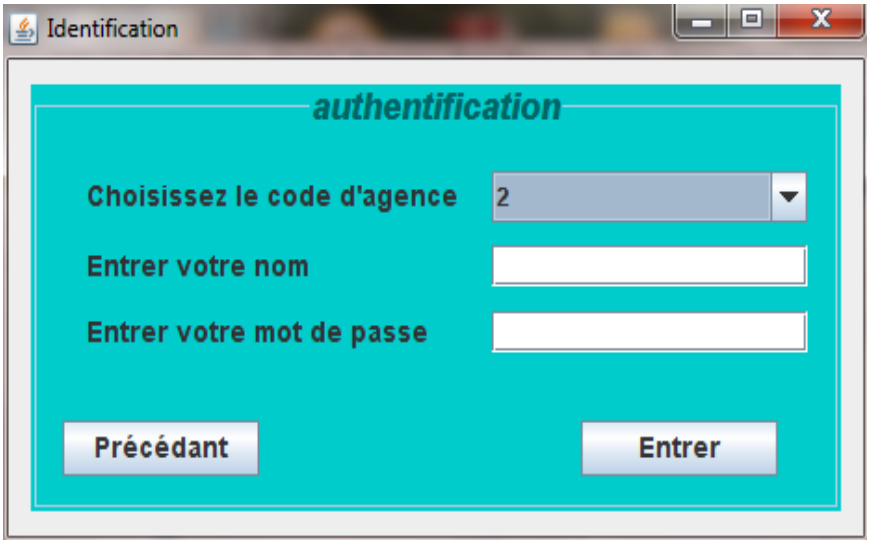


Figure IV.11 : Page d'authentification de l'agent.

Une fois l'agent connecté, la liste des clients qui partagent avec lui le même code catégorie sera affichée.

L'agent choisi le client, puis il va choisir les opérations, qu'il veut effectuer sur les comptes de ce dernier ; la figure suivante affiche la liste des clients.

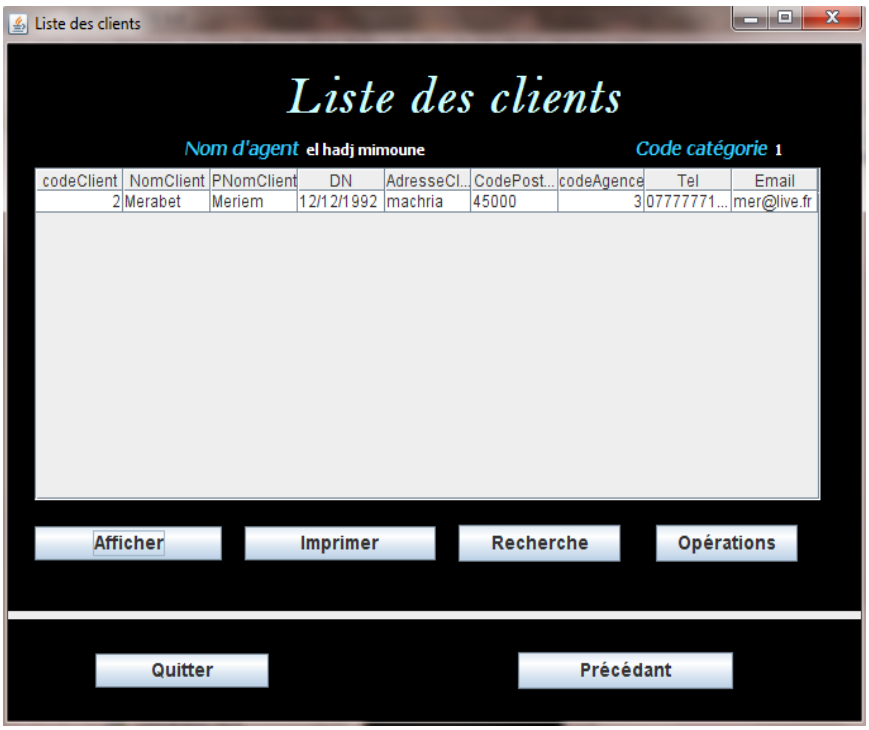


Figure IV.12 : Liste des clients selon la catégorie.

En cliquant sur « **Opérations** » après le choix d'un client l'interface suivante est affichée.

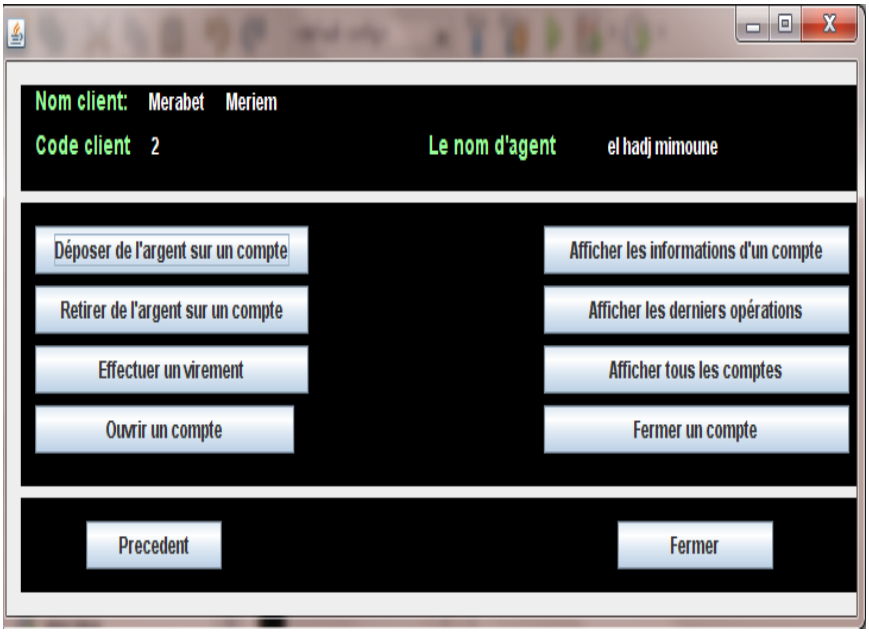


Figure IV.13 : Choisir une opération.

Lorsque l'agent choisit l'opération « **Retirer de l'argent sur un compte »** l'interface suivante sera affichée :

Figure IV.14 : Opération Retirer.

2.2.3 Accès administrateur

L'interface suivante représente l'authentification de l'administrateur.

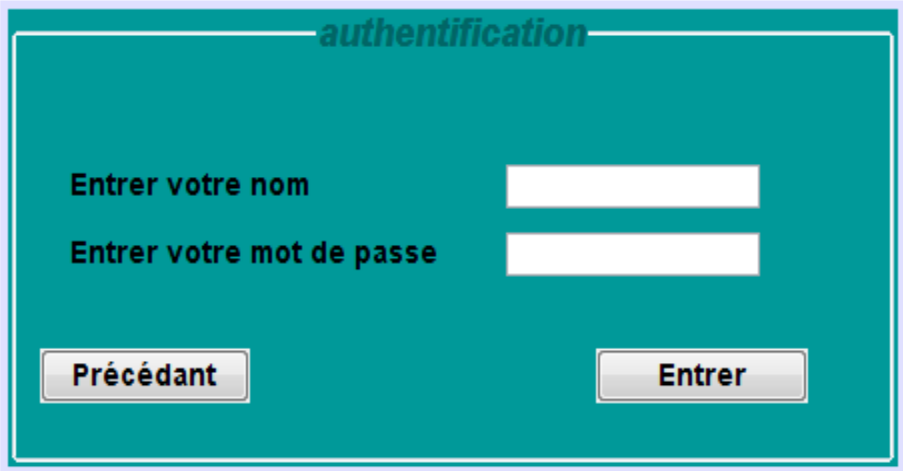


Figure IV.15 : Page d'authentification d'administrateur.

Après l'identification de l'administrateur, la figure suivante permet d'afficher les choix possibles pour ce dernier.

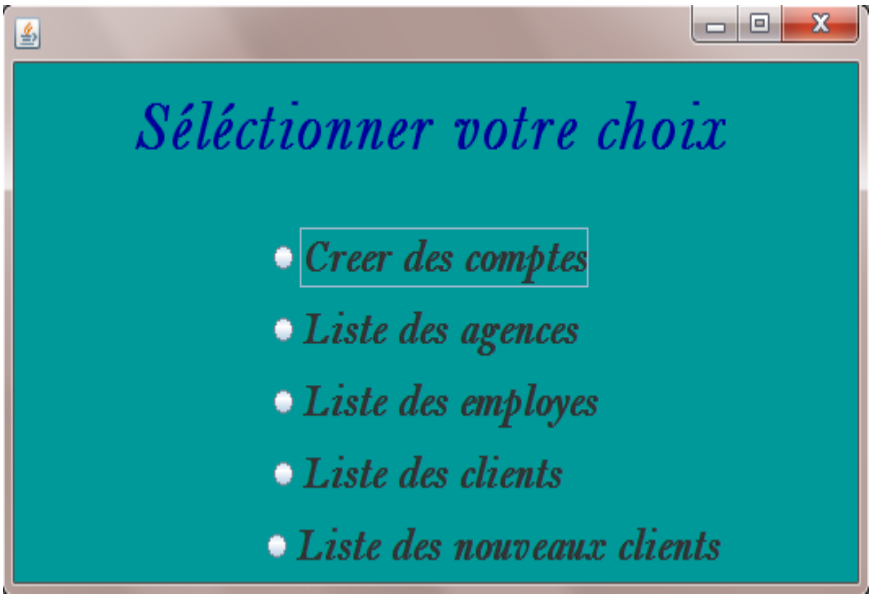


Figure IV. 16: Choix administrateur.

La figure suivante montre les agences disponibles pour cet administrateur.

Figure IV.17 : Liste des agences.

La figure suivante montre la liste des clients pour cet administrateur.

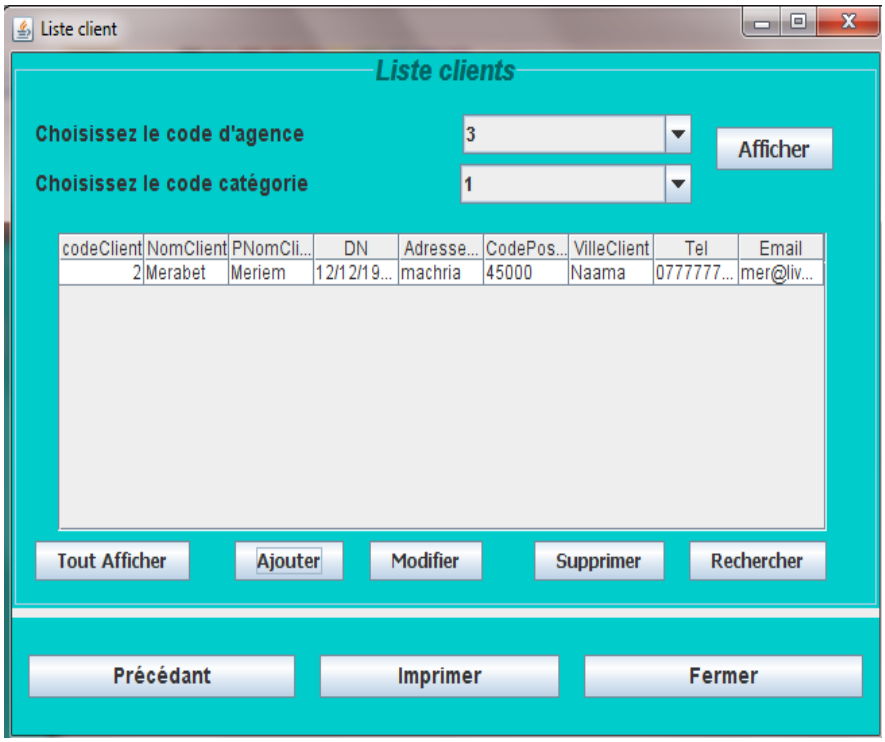


Figure IV.18 : Liste des clients.

Figure IV.19 : Liste des nouveaux comptes en attente de validation.

Pour le moment, nous avons réalisé un contrôle par rôle pour ce qui concerne l'Agent et l'Administrateur. Nous avons également, réalisé un contrôle d'accès pour le client qui consiste à désactiver un compte existant après 3 échecs d'authentification. Dans ce qui suit, nous proposons d'utiliser un algorithme de cryptage afin de coder les mots de passes des utilisateurs stockés dans la base de donnés afin de mieux protéger les informations des clients en cas de piratage de la base de données. Nous avons choisi le code de César pour réaliser cette opération.

2.2.3.1 Définition de code César

En cryptographie, le chiffrement par décalage, aussi connu comme le chiffre de César, est une méthode de chiffrement très simple utilisée par Jules César dans ses correspondances secrètes (ce qui explique le nom « chiffre de César »). Le texte chiffré s'obtient en remplaçant chaque lettre du texte clair original par une lettre à distance fixe, toujours du même côté, dans l'ordre de l'alphabet. Pour les dernières lettres (dans le cas d'un décalage à droite), on reprend au début. Par exemple avec un décalage de 3 vers la droite, A est remplacé par D, B devient E, et ainsi jusqu'à W qui devient Z, puis X devient A etc. Il s'agit d'une permutation

circulaire de l'alphabet. La longueur du décalage, 3 dans l'exemple évoqué, constitue la clé du chiffrement qu'il suffit de transmettre au destinataire — s'il sait déjà qu'il s'agit d'un chiffrement de César — pour que celui-ci puisse déchiffrer le message. Dans le cas de l'alphabet latin, le chiffre de César n'a que 26 clés possibles (y compris la clé nulle, qui ne modifie pas le texte). **[25]**

2.2.3.2 Cryptage de mot de passe de notre application

Nous avons utilisé le chiffrement de césar pour coder les mots de passe des utilisateurs. Cette tache est réalisée par l'administrateur.

Selon **la figure IV.8** qui est représente l'inscription d'un nouveau client, la validation d'un client est en attente du la validation par l'administrateur.

C'est l'administrateur qui doit fournir les mots de passe aux clients. Mais avant cette tache l'administrateur doit chiffrer chaque mot de passe en utilisant le code César.

La figure suivante indique la validation d'un client par un administrateur, il fournit les mots de passe dans le champ « Mot de passe » et le confirmer dans le champ « Confirmation de mot de passe », dans ce cas il a entré la chaîne de caractère « jamila » qui est cryptée en « RIUQTI » dans la case « mot de passe » de la base de données du client avec le code 11 qui est affiché dans la figure IV.21 ci-dessous.

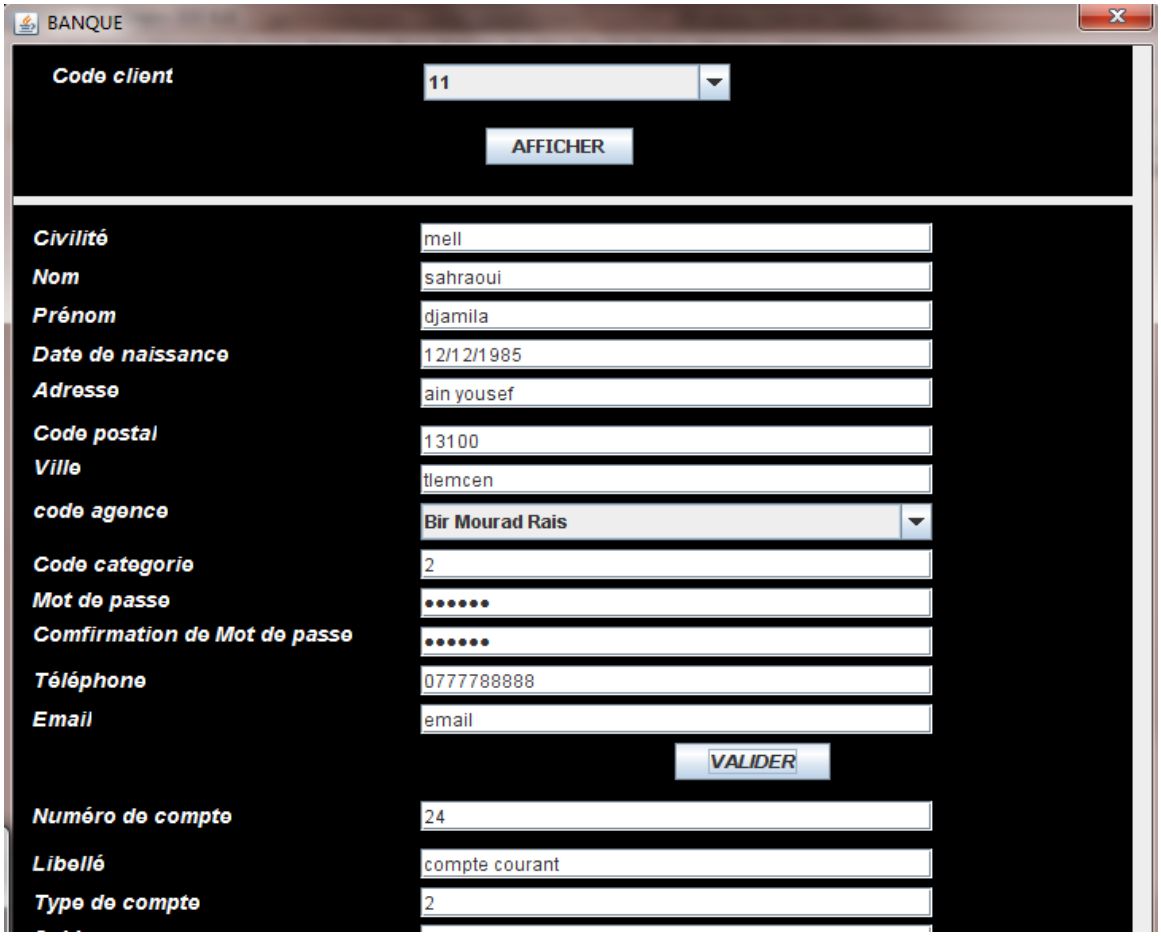


Figure IV.20 : Validation d'un client par un administrateur.

codeClient	NomClient	PNomClient	MotPass
2	Merabet	Meriem	OGT
11	sahraoui	djamila	RIUQTI

Figure IV.21 : Ajout d'un client avec le chiffrement de mot de passe.

3. Conclusion

Ce chapitre a été consacré à la présentation de notre application, cette dernière contient deux partie, la première est une application Web réalisée pour gérer les clients dans un environnement 3-tiers, nous avons proposé dans cette partie un contrôle d'accès sur le nombre de tentatives erronés d'authentification avant de désactiver un compte existant ainsi qu'un mécanisme de chiffrement afin de protéger les informations pertinentes de la base de données. Cette dernière

opération est totalement transparente par le client car elle est réalisée par l'administrateur.

La deuxième partie de notre application est une implémentation 2-tiers afin de gérer les fonctionnalités de l'Agent et de l'Administrateur, à ce stade nous avons proposé un contrôle par rôle, basé sure le modèle par rôle, nous avons également fait l'audit des opérations réaliser par le client.