

Cahier de charge

- **Description du projet :**

Objectifs du projet :

L'objectif principal du projet est la création d'une application E-Banking qui permet d'effectuer les transactions bancaires en toute sécurité et rapidité.

Le projet a pour but de créer un lien solide entre l'agent de la banque et les clients. Grace à cette solution, on facilitera les transactions bancaires en suivant un processus sécurisé et fiable. Et En garantissant à chaque type d'utilisant un espace afin de bénéficier des fonctionnalités qui aura les privilèges à les exécuter.

2. Présentation du projet :

Le projet est pour simplifier la vie des clients de la banque et aider les banques à renforcer leur stratégie en respectant leurs spécificités et contraintes et exigences.

Ainsi, les objectifs principaux du projet sont :

Création d'un espace d'administration afin de paramétrer les devises et les variables globales de la solution.

Création d'un espace réservé à la banque afin de créer les enrôlements des clients et la gestion des clients.

Création d'un espace réservé au client afin de :

- Visualiser les comptes associés à son enrôlement
- Réserver un espace pour effectuer leurs virements multiple.
- Réserver un espace pour gestion de son compte.
- **Demander un rendez-vous avec le banquier**

~~Création d'un espace afin d'effectuer une vérification des transactions bancaires. —(pas la peine)~~

3. Expression des besoins a.

Besoins fonctionnels

Cette phase représente un point de vue « fonctionnel » de l'architecture système. Par le biais des cas d'utilisation, nous serons en contact permanent avec l'acteur du système en vue de définir les limites de celui-ci, et ainsi éviter de trop s'éloigner des besoins réels de l'utilisateur final.

Les principaux profils qui auront à utiliser sont les suivants :

- ❖ Client : Un utilisateur qui pourra y accéder à son compte où il aura la possibilité de visualiser ces Compte de facturation et son solde. Ainsi que d'effectuer des virements en saisissant le numéro de compte de bénéficiaires.
- ❖ Agent de la banque : Un utilisateur qui pourra y accéder à son propre espace afin d'enrôler des nouveaux clients ou bien de résilier un client déjà enrôlés, ainsi que de gérer les abonnées, il pourra soit activer ou suspendre ou lever la suspendu d'un client.
- ❖ Admin : Un utilisateur qui aura la possibilité de créer une nouvelle agence ainsi de gérer les agents. Ainsi que de contrôler les paramètres globales de l'application compris (Les précisions de solde, ainsi de contrôler les libellés des statuts des clients et leurs contrats).

b. Besoins non fonctionnels

Ce sont des exigences qui ne concernent pas spécifiquement le comportement du système mais plutôt identifient des contraintes internes et externes du système. Les principaux besoins non fonctionnels de notre application se résument dans les points suivants :

- ❖ Le code doit être clair pour permettre des futures évolutions ou améliorations. Et d'être paramétrable le plus possible afin d'assurer une simplicité lors changements sans être obliger de modifier le code source.
- ❖ L'ergonomie : l'application offre une interface conviviale et facile à utiliser.
- ❖ La sécurité : l'application doit respecter la confidentialité des données. Ainsi d'assurer une fiabilité lors de l'exécution des transactions bancaires...

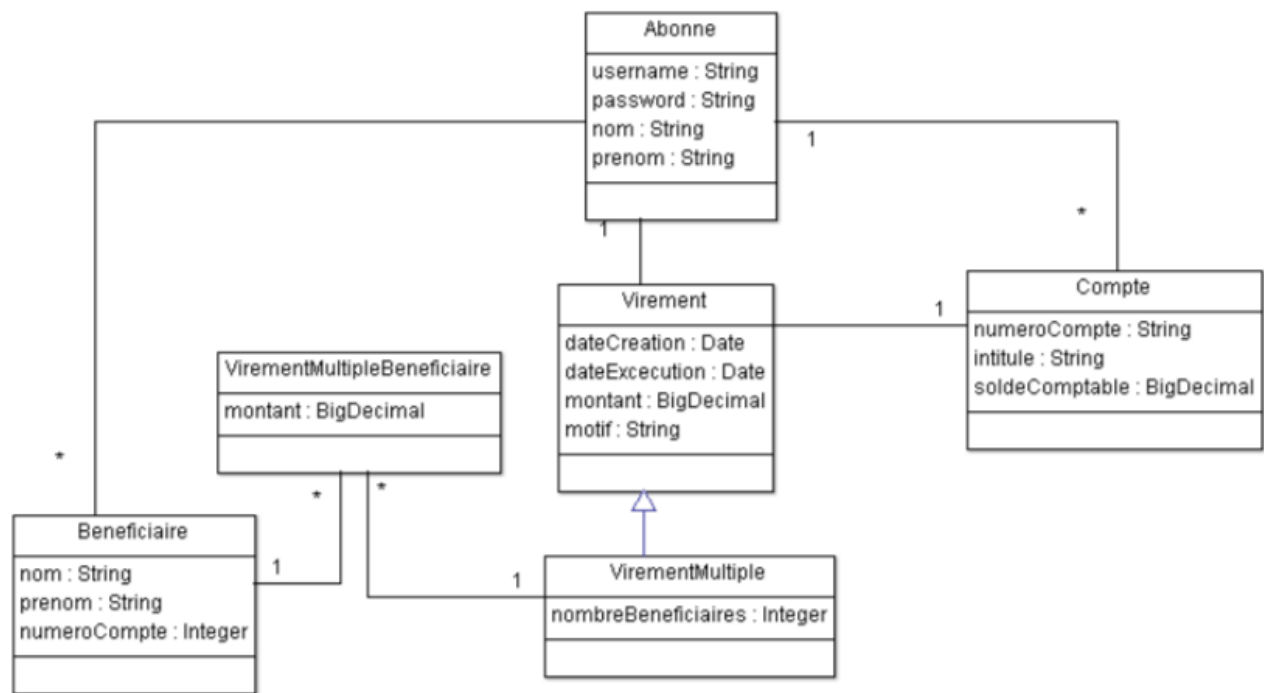
un virement multiple

introduction :

L'objectif est de mettre en place une petite application de gestion des virements multiples.

Un utilisateur (un client) après son authentification, saisie une demande de virement multiple, il remplit un formulaire de saisi, ensuite clique sur Enregistrer, si la saisie est correcte, il est redirigé vers la page de consultation (visualisation de la demande), dans cette page, il saisit son mot de passe et clique sur un bouton pour signer sa demande, si le mot de passe saisi est le même dans la base de données, il est redirigé vers une autre page avec un message de succès. Il peut aussi modifier sa demande ou de rechercher des demandes.

Diagramme de classe :



Écran de saisie (DEMO) :

Identifiant	Numéro du compte
11113	5444532102 ECHATOUT MOURAD
2109	55844103 BELMOUDEN MILOUD
2113	104077289873605412102986 ALAOUT SAID
2111	104653154878885412365472 ETTAHIRI MALIKA

Bénéficiaire	Montant	idb
ALAOUT SAID	5800	2113
BELMOUDEN MILOUD	14500	2109

Choisir un compte: 06377895 Date de création: 10/04/2013

Date d'exécution: 26/04/2013 Motif: VRT MUL

Nombre de bénéficiaires: 4 Montant: 44750

Le nombre de bénéficiaires saisi est différent au nombre de bénéficiaires sélectionnés. Le montant total doit être égal à la somme des montants des bénéficiaires.

Enregistrer

Ceci est l'écran de la DEMO, on propose au client, sa liste des bénéficiaires (un client ou un abonné a plusieurs bénéficiaires), ensuite il sélectionne une partie de ses bénéficiaires en ajoutant le montant de virement, on fait un contrôle sur les montants et le nombre de bénéficiaires, on affiche une liste déroulante des comptes de l'abonné, puis on enregistre la demande.

Vous pouvez mettre votre propre ergonomie, design, styles, façon de saisie ; le plus important est de respecter le modèle (Diagramme de classe).

Ensuite, on affiche un écran de consultation de virement saisi, avec quelques informations, la liste des bénéficiaires choisis, ainsi qu'un bouton signer avec une zone de saisie de mot de passe.

Mot de passe:  Confirmer et signer

L'action signer vérifie uniquement la validité du mot de passe saisi (doit être le même de l'utilisateur connecté), si le mot de passe est incorrect, on affiche un message d'erreur (En javascript, on met un alert(« Erreur »), sinon, on lui redirige vers un écran avec un message clair, en vert : « Votre demande a été signé ! »).

L'utilisateur peut rechercher les virements selon quelques critères : Compte, date de création, montant, motif, ...

Dans cette application, mettre le compte, la date de création, le montant et le motif.

Choisir un compte:

10000004

Date d'exécution:

Montant min:

8000

Montant max:

Statut:

Liste de vos virements multiples

id	dateCreation	Compte	Montant	Motif	Statut
15573	25/01/2013	10000004 REDA ALAQUI COMPTE AFFAIRES	11000,00	MTFF	Enregistré
15581	25/01/2013	10000004 REDA ALAQUI COMPTE AFFAIRES	11000,00	MTFGF	Confirmé et signé
16229	07/02/2013	10000004 REDA ALAQUI COMPTE AFFAIRES	15000,00	MTF	Enregistré

Page 1 of 1

Rechercher

Displaying 1 - 3 of 3

L'utilisateur peut aussi modifier sa demande.

Outils et versions :

- **back-end :**

- spring MVC : 4.3.2.RELEASE
- spring Security : 4.0.1.RELEASE
- spring data :1.10.2.RELEASE
- hibernate : 5.1.1.Final
- jackson : 2.8.0
- Maven : 3.0.0

- **Front end :**

- Angular