

UN SYSTEME DE
PAIEMENT POUR UNE
MONNAIE
SCRIPTURALE
IDENTIFIEE

Projet SOA, *suivi par Monsieur Henry BOCCON-GIBOD*

INTRODUCTION

Le suivi des émissions des masses monétaires est géré par des Banques Centrales. Ces masses monétaires sont composées de monnaies fiduciaire, scripturale et divisionnaire.

Pour mémoire, la monnaie fiduciaire est l'ensemble de la monnaie émise par un Etat sous la forme de billets de banques identifiés. La monnaie scripturale est matérialisée par une écriture comptable sur un chèque ou un compte en banque par exemple. Il s'agit de celle que l'on trouve sur les comptes bancaires et qui se déplace dans le cadre des flux (mouvements, échanges) monétaires.

La monnaie divisionnaire désigne les pièces de monnaie métalliques. Si elles ont perdu leur valeur intrinsèque, elles ont un coût de fabrication, ajusté de façon à dissuader leur contrefaçon, sans dépasser leur valeur nominale. Les variantes d'alliage et de composition de ces pièces n'ont donc qu'un but pratique. Les pièces sont généralement produites par le trésor public mais ne correspondent pas à une émission de monnaie : elles ne sont mises en circulation qu'après échange contre une quantité équivalente de billets.

Récemment la forme électronique de la monnaie est apparue sous la forme de cartes à puce de paiement rechargeables. Cette forme fonctionne de la même façon que la monnaie scripturale.

Des quatre formes de monnaie, à ce jour seule la monnaie fiduciaire est identifiée, par un code unique porté sur chaque billet de banque.

L'identifiant des billets est codifié de manière à pouvoir les référencer plus facilement mais également de pouvoir en repérer des falsifications. Ce code est composé d'une lettre, suivie de trois chiffres, puis une lettre et à nouveau un chiffre. La première lettre indique l'organisme qui a imprimé le billet, le nombre à trois chiffres qui suit cette lettre renvoie à la matrice utilisée à l'impression. Enfin, la position du billet sur la planche est indiquée par une lettre et un chiffre. Par exemple, un billet portant le numéro L068B6 a été imprimé par la Banque de France, avec la 6matrice, et il était situé sur la sixième ligne de la deuxième colonne sur la planche d'impression (une planche d'impression peut avoir jusqu'à dix colonnes et six lignes).

Aujourd'hui l'argent n'est donc identifié que sous cette forme de billets de banque. Les pièces de monnaie ainsi que les mouvements bancaires ne le sont pas. Et lorsqu'il est porté à un compte, l'argent n'apparaît que sous la forme d'une somme. Ce n'est que l'affectation d'un montant à un numéro de compte. Rien ne permet donc de discerner l'argent sous forme scripturale. Le contrôle de l'essentiel des masses monétaires et de leur valeur est de ce fait difficile ; en particulier une grande part de celles-ci est illégale.

L'absence d'identification de la monnaie divisionnaire porte moins à conséquence, du fait de la difficulté de son utilisation pour le paiement de grosses sommes, sous cette forme ceux-ci étant pragmatiquement soumis à des maximums réglementaires.

Dans les années soixante-dix le projet Arpanet développa ce qui devait devenir le protocole Internet, il dût affecter à chaque machine un identifiant, afin d'être reconnue comme nœud du réseau. Il fit gérer et partager par ces machines des tables contenant les listes des adresses identifiant chaque ordinateur connecté.

L'ICANN (Société pour l'attribution des noms de domaine et des numéros sur Internet), s'est chargée de la gestion des identifiants IP uniques ainsi que de la gestion des attributions des noms de domaines.

Ce qui a été techniquement réalisé pour l'organisation de l'Internet pourrait-il être transposé à l'émission monétaire ?

CAHIER DES CHARGES

Contexte

Aujourd'hui chacun des billets mis en circulation est identifié par une Banque Centrale, au moyen d'un code identifiant unique. La monnaie divisionnaire est émise en contre partie de monnaie fiduciaire. A la différence des monnaies fiduciaire et divisionnaire émises par les banques centrales, la monnaie scripturale, est émise principalement par les Etats sous forme de bons du Trésor, et par les établissements privés de crédit. .

Objectifs

L'objectif principal de l'application est de restituer aux banques centrales le contrôle de toute émission monétaire, quel qu'en soit la forme.

Il s'agit de créer le référencement et l'identification de la monnaie scripturale en unités de valeur monétaire. De la même façon que pour les billets, il s'agit de procurer la possibilité de convertir des unités de valeur monétaire auprès des banques, qui exercent le rôle de tiers de confiance des transactions en assurant le service d'échanges d'unités de valeurs qu'elles nécessitent .

Fonctions de l'application

Les principales fonctions de l'application sont :

- De réaliser toute transaction de paiement entre un acheteur et un vendeur, via leurs banques et tout organisme assurant le rôle de tiers de confiance.
- L'acheteur aura la possibilité de convertir la monnaie avant d'effectuer tout achat.
- La possibilité d'échanger de la monnaie scripturale auprès de banques, en collections d'unités nominales de différentes valeurs scripturale
- La possibilité de convertir de la monnaie scripturale en monnaie fiduciaire ou divisionnaire
- Le contrôle par les banques centrales des émissions de la monnaie scripturale par les banques privées et les Etats.

Détails de chaque fonction

- Les échanges de collections d'unités de valeur scripturale
- La possibilité de réaliser des virements, des dépôts, de régler par carte bancaire. Toutes ces transactions sont identifiées de façon unique à l'aide d'un serveur qui attribue des clés à chaque nouveau paquet.
- La possibilité de convertir de la monnaie. L'utilisateur peut sur demande, effectuer de l'échange de monnaie scripturale en monnaie fiduciaire ou divisionnaire. A partir d'un chèque, ou d'une carte de crédit il est possible d'obtenir des pièces et/ou des billets.
- Le moyen d'identifier l'émission de la monnaie scripturale

- Si l'acteur, souhaite avoir un historique de toutes ses transactions, cela devient possible. A partir du numéro d'émission, on peut déterminer s'il s'agit d'un virement, d'un dépôt, ou d'un paiement par carte de crédit, suivi de tous les détails nécessaires (montant, récepteur, date, nouveau solde).

EXPLICATIONS FONCTIONNELLES

Pour mettre en place ce projet, il est nécessaire d'avoir une application principale ainsi que plusieurs web services. Ils sont assez similaires dans le traitement des demandes émises par l'application.

Le système «Banque», permet aux utilisateurs de consulter, de faire des demandes de transactions ou de monnaie. Tous les comptes bancaires sont créés préalablement par un organisme financier. Le client ouvre un compte dans une banque de crédit en lui donnant toute information nécessaire d'identification. Une fois le compte ouvert, la banque lui donne les codes d'accès nécessaires à la consultation de son compte. La Banque fournit un premier web service, qui est celui de la consultation de compte.

Le système de la « Banque Centrale » a pour rôle de recevoir les demandes de transaction ou de monnaie en provenance de la banque de crédit. Par la suite, il les traite et renvoie une réponse à l'émetteur de l'opération ou un simple résultat. Ces requêtes concernent les virements, les dépôts, les paiements par carte bancaire et les demandes de conversion monétaire. A chaque demande valide, un numéro sera créé. La Banque Centrale fournit plusieurs web services, tels que ; faire un dépôt, faire un virement, convertir de la monnaie et régler par carte bancaire.

Le système de la Banque Centrale joue aussi à cette occasion le rôle de tiers de confiance pour sécuriser les transactions.

Le système des «Sociétés extérieures», concerne les commerces, les sociétés ou encore les centres administratifs. Tout utilisateur souhaitant régler par carte bancaire un achat dans un commerce, entraîne la création automatique d'un numéro de transaction par la Banque Centrale. Ce système fournit un web service permettant le paiement par carte bancaire.

D'un point de vue technique, la Banque centrale ainsi que les sociétés extérieures sont des « providers ». Un provider est une entité fournissant des services informatiques, au travers d'un réseau internet. Dans l'architecture proposée, l'application faisant appel à ce provider, doit respecter un modèle prédéfini en **WSDL** (*Web Services Description Language*). Ce modèle peut être considéré comme un contrat.

Lorsqu'il y a une demande de transaction, la requête est envoyée à la Banque centrale qui codifie la demande. Si la requête est acceptée, la transaction est codifiée sinon la demande est rejetée. Selon le type de transaction (dépôt, paiement par carte bancaire, conversion de monnaie), la codification de l'opération est différente. Cette demande est une requête **SOAP** (*Simple Object Access Protocol*). Elle permet la transmission de messages entre objets distants, ce qui veut dire qu'il autorise un objet à invoquer des méthodes d'objets physiquement situés sur un autre serveur. Le transfert se fait le plus souvent à l'aide du protocole HTTP.

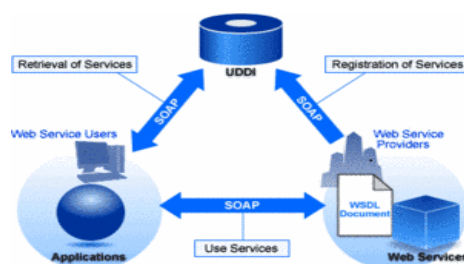


Schéma représentant le respect du contrat

CAS D'UTILISATIONS

Consulter un compte

1. L'utilisateur se connecte à l'application « Espace client », allez dans la rubrique « Consulter mon compte ».
2. Saisir son identifiant et mot de passe.
3. Valider.
4. Le système vérifie la tentative d'authentification.
5. Affiche les transactions liées au compte ainsi que la liste de billets disponible.

Erreurs

E3.1. L'identifiant ou le mot de passe est erroné. Un message d'information apparaît et invite l'utilisateur à ressaisir ces codes d'accès (trois tentatives possibles).
Retour à l'étape 2.

Faire un dépôt

1. L'utilisateur se connecte à l'application « Espace client », allez dans la rubrique « Faire un dépôt ».
2. Saisir son identifiant et mot de passe
3. Valider.
4. Le système vérifie la tentative d'authentification.
5. Saisir la liste de billets à déposer, ainsi le numéro de compte.
6. Valider.

Erreurs

E3.1. L'identifiant ou le mot de passe est erroné. Un message d'information apparaît et invite l'utilisateur à ressaisir ces codes d'accès (trois tentatives possibles).
Retour à l'étape 2.

E5.1 Les coordonnées bancaires sont erronées. Un message d'information apparaît et invite l'utilisateur à ressaisir ces codes d'accès (trois tentatives possibles).
Retour à l'étape 4.

Faire un virement

1. L'utilisateur se connecte à l'application « Espace client », allez dans la rubrique « Faire un virement ».
2. Saisir son identifiant et mot de passe
3. Valider.
4. Le système vérifie les identifiants saisis.
5. Saisir la somme à retirer, ainsi le numéro de compte concerné (numéro de compte, clé RIB, numéro de guichet).
6. Valider.
7. Un message informe l'utilisateur que le virement s'est réalisé correctement.

Erreurs

E3.1. L'identifiant ou le mot de passe est erroné. Un message d'information apparaît et invite l'utilisateur à ressaisir ces codes d'accès (trois tentatives possibles).

Retour à l'étape 2.

E5.1 Les coordonnées bancaires sont erronées. Un message d'information apparaît et invite l'utilisateur à ressaisir ces codes d'accès (trois tentatives possibles).

Retour à l'étape 4.

E5.1.1 Le solde est insuffisant. La transaction est annulée.

Convertir de la monnaie

1. L'utilisateur se connecte à l'application « Espace client », allez dans la rubrique « Convertir de la monnaie ».
2. Saisir son identifiant et mot de passe
3. Valider.
4. Sélectionner le type de monnaie de départ (s'il s'agit d'un chèque ou de monnaie billets et pièces).
5. Valider.

Erreurs

E3.1. L'identifiant ou le mot de passe est erroné. Un message d'information apparaît et invite l'utilisateur à ressaisir ces codes d'accès (trois tentatives possibles).

Retour à l'étape 2.

Chemins alternatifs

A4.1.1. L'utilisateur souhaite obtenir de la monnaie à partir d'un chèque.

A4.1.2. Il saisit le numéro du chèque bancaire.

A4.1.3. Valider.

A4.1.4. Le système de la Banque centrale rend de la monnaie.

A4.2.1. L'utilisateur souhaite obtenir de la monnaie à partir de billets.

A4.2.2. Il saisit la liste de billets.

A4.2.3. Le système valide la liste de billets.

A4.2.4. Le système de la Banque centrale rend de la monnaie.

Erreurs

E4.1.2.0 Le numéro du chèque bancaire est erroné. Un message d'information apparaît et invite l'utilisateur à ressaisir le numéro du chèque (trois tentatives possibles).

Retour à l'étape 4.1.2.

E4.2.3. Un numéro de billet est erroné. Un message d'information apparaît et invite l'utilisateur à ressaisir le numéro.

Retour à l'étape 4.2.2.

Régler par carte bancaire

1. L'utilisateur réalise un paiement par carte bancaire dans un commerce.
2. Il saisit son code.
3. Valide.
4. Le système financier du commerce se connecte au système de la banque du client pour autoriser ou non le paiement.
5. Le système de la banque, se connecte avec celui de la banque centrale pour créer une nouvelle transaction.
6. Le client retire sa carte. Fin du règlement.

Erreurs

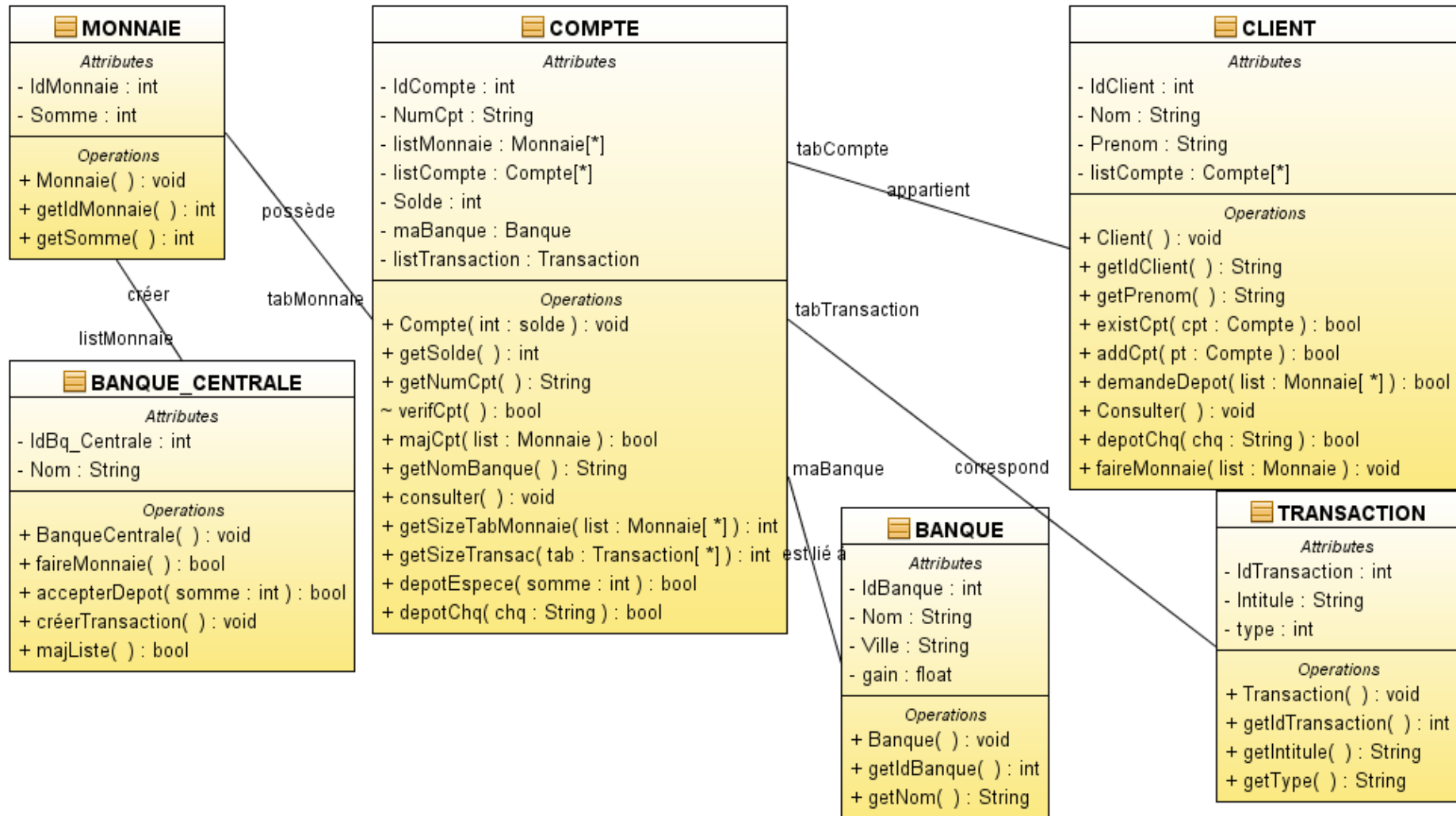
E3.1. Le client a saisi un code erroné. Il ressaisit son code (trois tentatives possibles, sinon fin de la transaction).

Retour à l'étape 2.

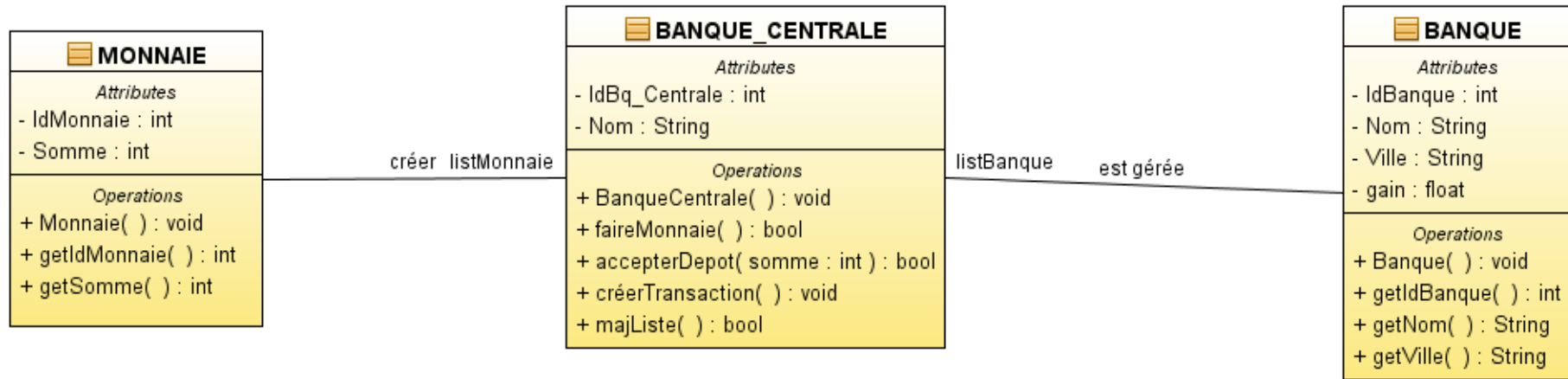
E4.1 Le système de la banque du client, refuse le paiement car le solde est insuffisant. Fin de la transaction.

Retour à l'étape 6.

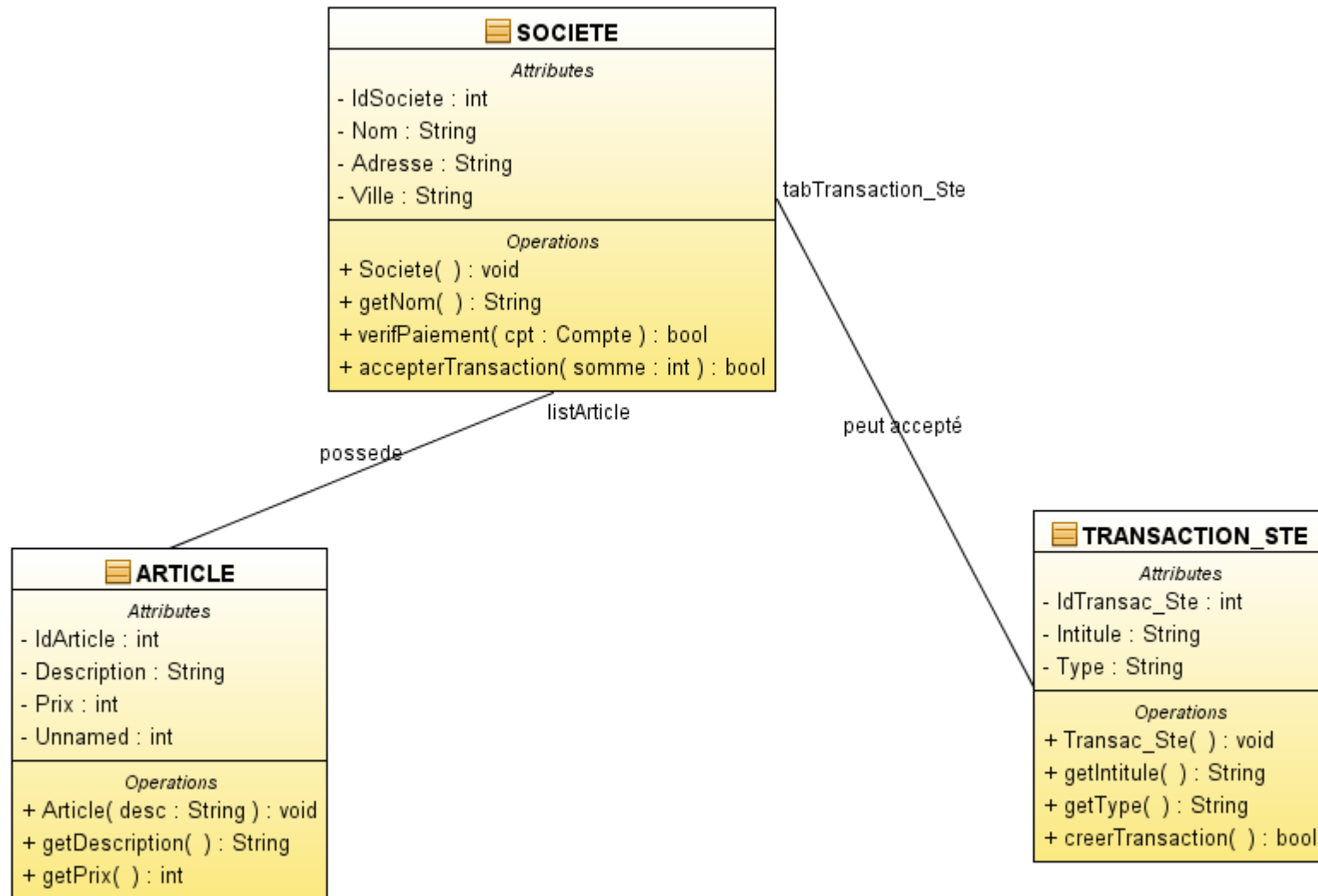
MODELE DE CLASSES « BANQUE »



MODELE DE CLASSES « BANQUE_CENTRALE »



MODELE DE CLASSE « SOCIETE »

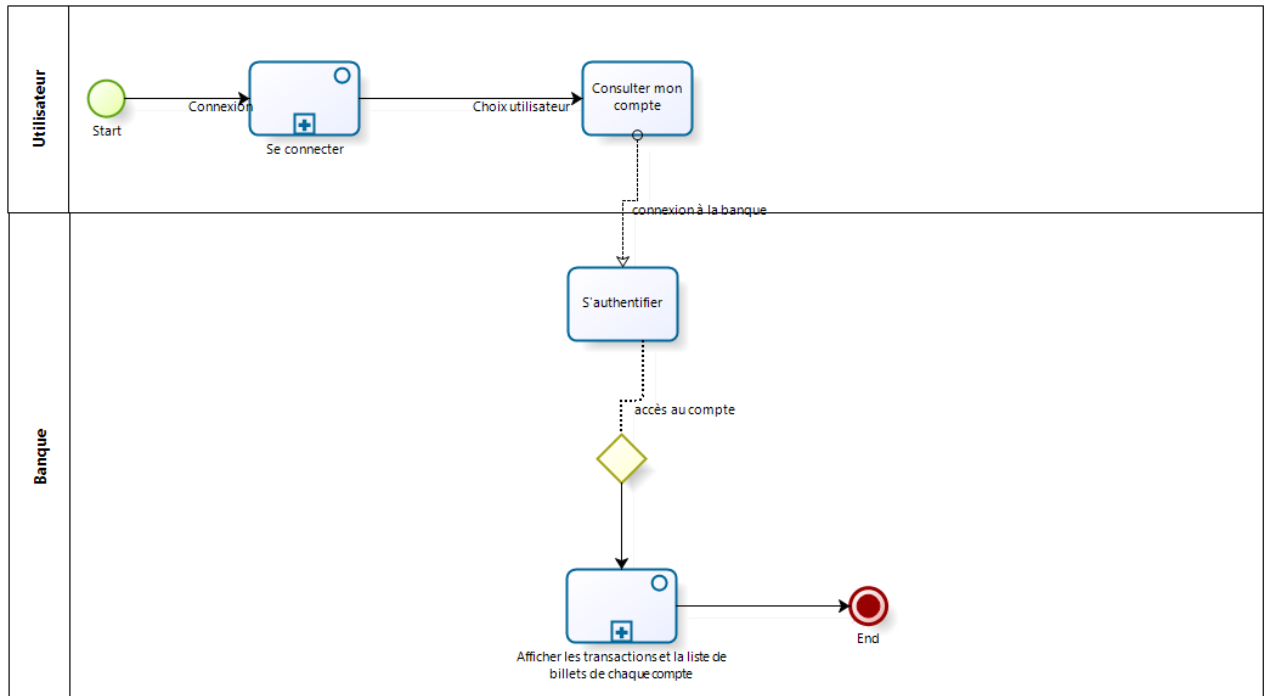


MODELISATION BPMN

La modélisation BPMN (Business Process Modeling Notation), est une notation graphique permettant de modéliser les processus métier. J'ai donc utilisé cette spécification pour représenter mes différents cas d'utilisations.

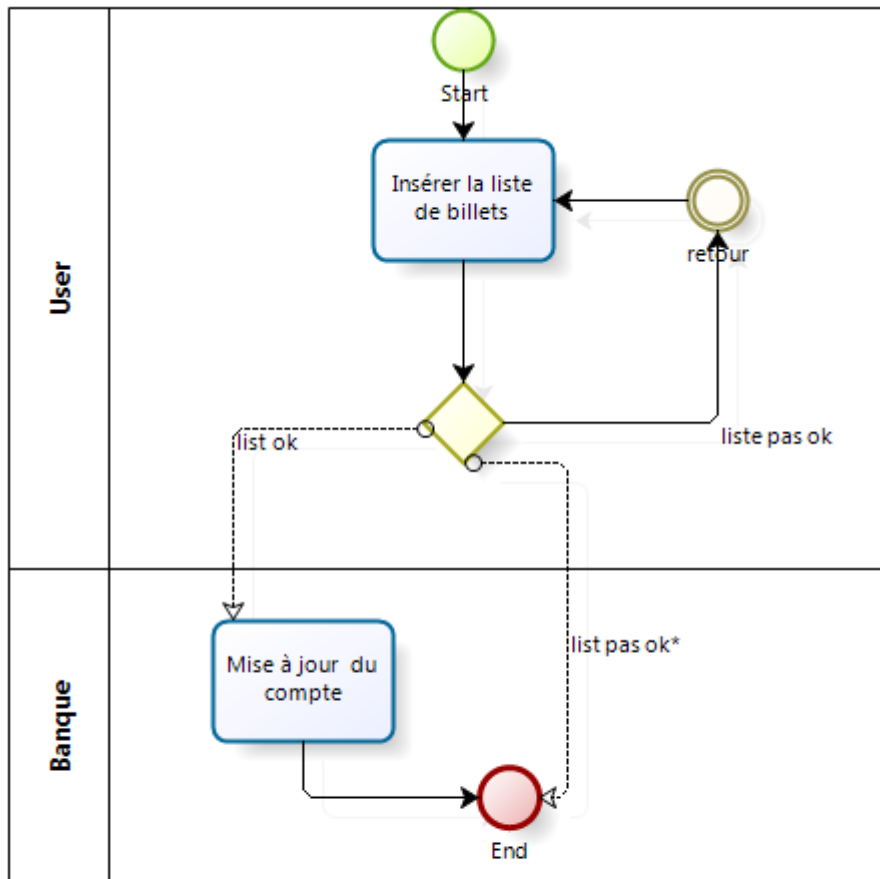
CONSULTER

Après s'être authentifié, l'utilisateur peut consulter l'état de son compte en saisissant les code d'accès fournit par son conseiller lors de l'ouverture du compte.



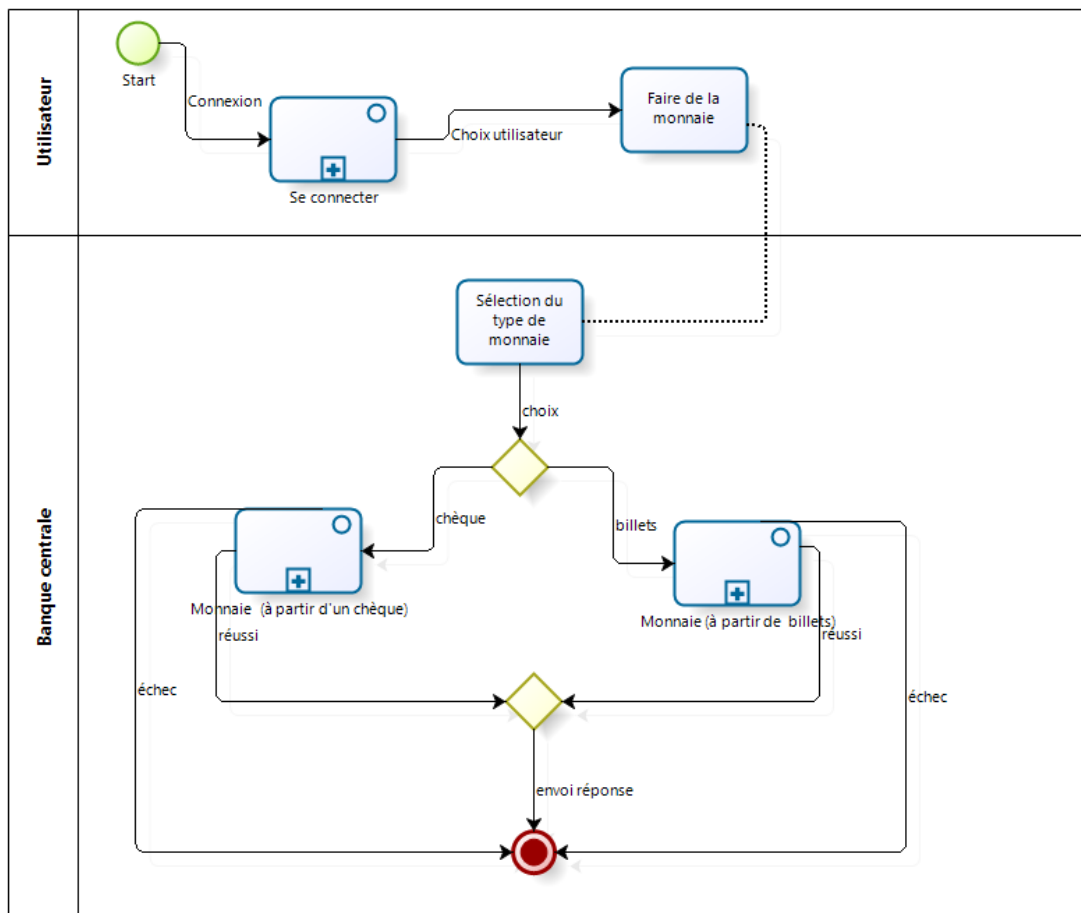
FAIRE UN DEPOT

L'utilisateur s'authentifie puis sélectionne « Faire un dépôt ». La demande est transférée à la banque centrale. Elle codifie la transaction. Le compte est ensuite mis à jour.

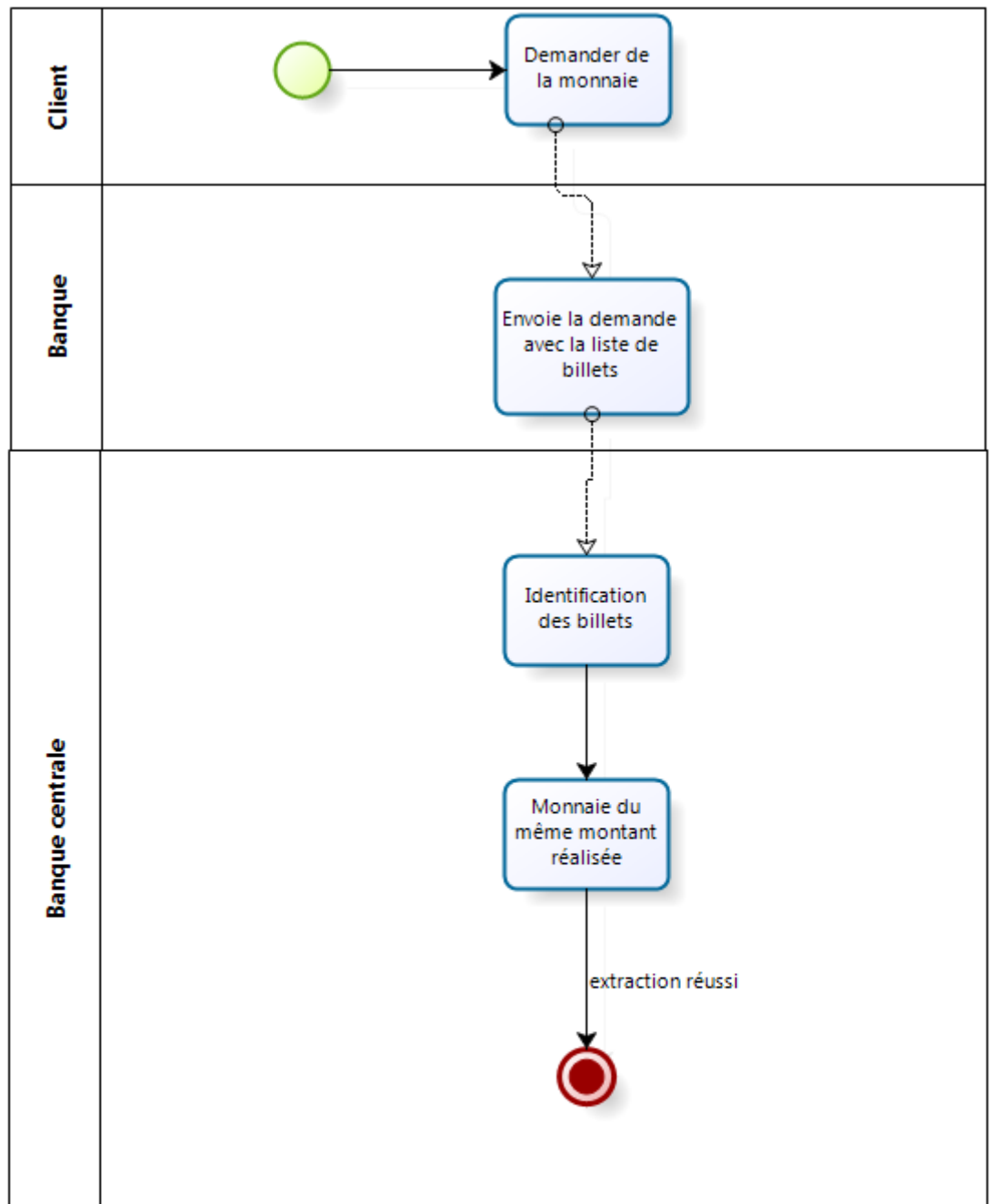


CONVERTIR DE LA MONNAIE

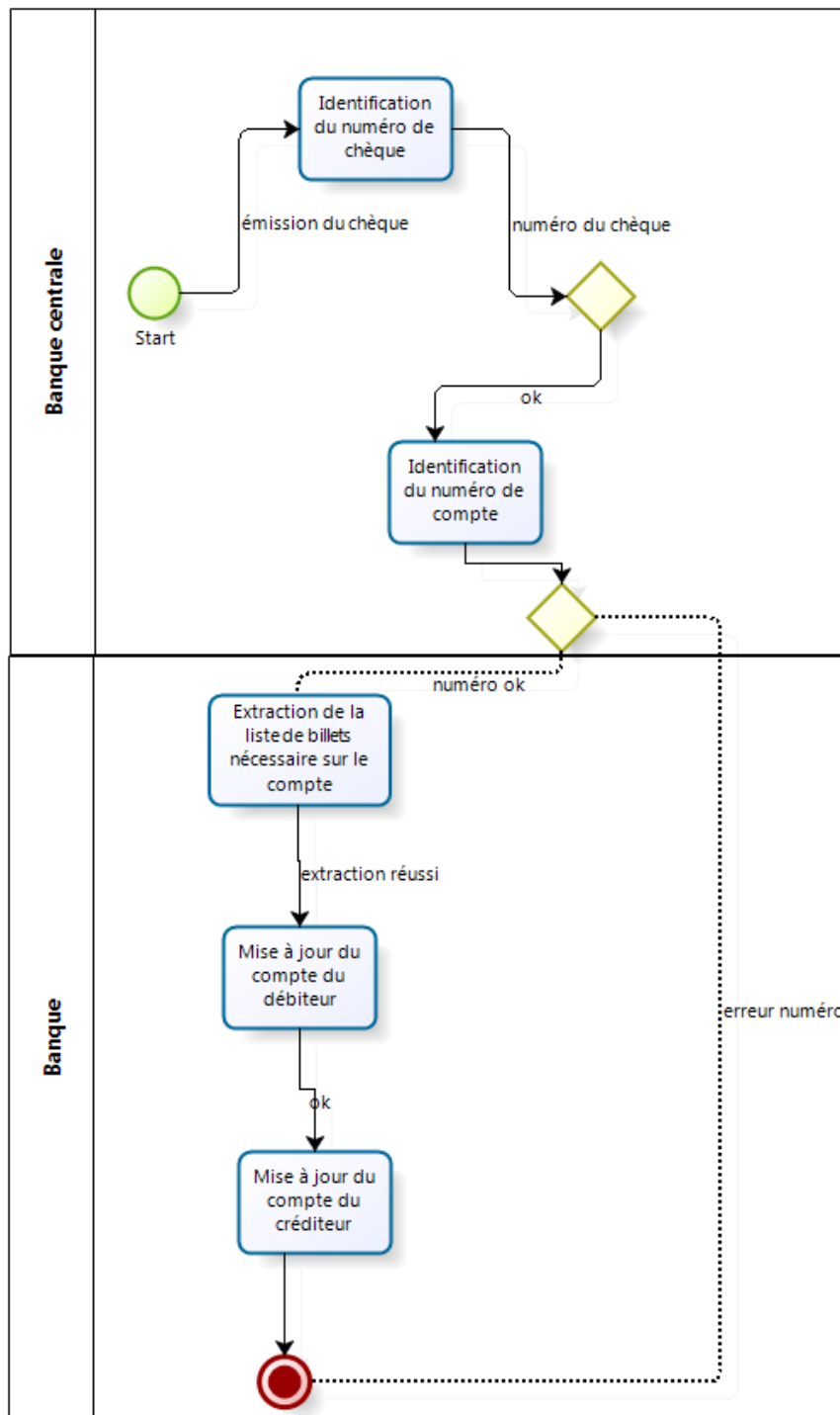
L'utilisateur est connecté au système de sa banque pour faire la demande.



- SOUS PROCESSUS
 - MONNAIE A PARTIR D'ESPECE






○ MONNAIE A PARTIR D'UN CHEQUE



APPLICATION

Pour réaliser cette application, j'ai utilisé les technologies suivantes :

Système d'exploitation	Windows 7 
IDE	NetBeans 6.7 
Serveur	Tomcat 
Plate-forme	J2EE

Il existe trois applications différentes à implémenter ; la Banque, la Banque Centrale et les Sociétés extérieures.

La technologie SOA est utilisée pour la génération des transactions suite à un mouvement bancaire autorisé tel qu'un virement, un dépôt ou un paiement.

L'application Banque possède une SOA (un bouton nommé [SOA Générer monnaie]) permettant d'effectuer de la monnaie auprès de la Banque Centrale.

L'application société disposera d'une SOA (un bouton nommé [SOA Générer transaction]) pour avoir la possibilité de créer des transactions liées au compte de l'acheteur après un achat sur son site, une seconde SOA ([SOA Générer monnaie]), pour réaliser de la monnaie auprès de la Banque Centrale.

De nos jours, le système monétaire est peu fiable et peu à tout moment basculer. En appliquant ce système, il est possible de limiter certaines crises.

En effet la présence de billets de virtuels permet à la Banque Centrale d'avoir des billets à disposition. En gardant ce principe, il est impossible d'avoir une « rupture » du côté de la Banque Centrale.

La monnaie scripturale, elle est également « virtualisée » comme les billets. Ainsi, cela réduit le nombre de fraude et augmente la monnaie en circulation.

A long terme, si ce système été conservé, cela pourrait devenir une solution d'amélioration du système actuel.

Cas du Portugal

Les banques au Portugal et dans d'autres pays européens sont confrontées à un manque de liquidités auquel les pouvoirs publics doivent rapidement trouver une solution.

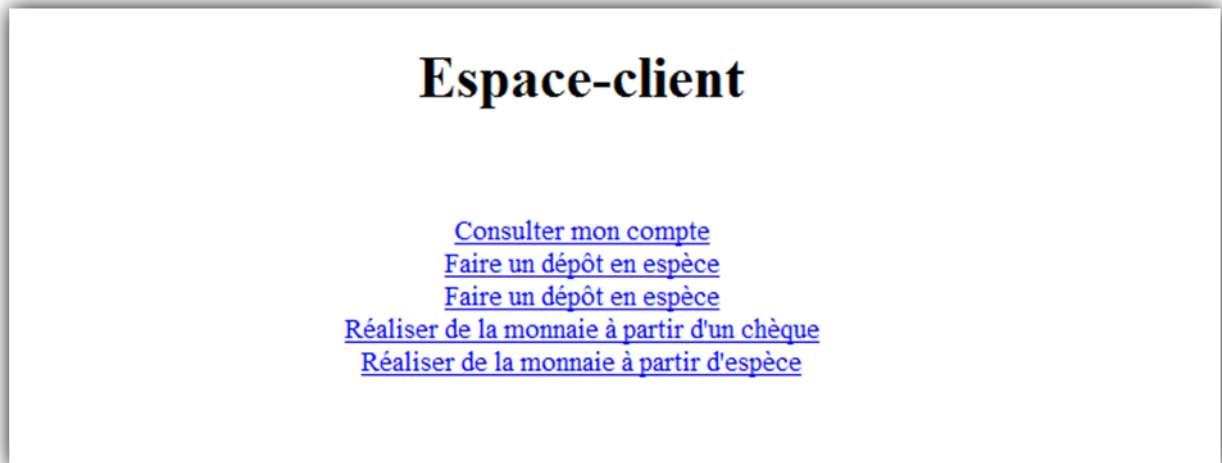
La **liquidité** d'un marché financier représente la capacité à acheter ou à vendre rapidement les actifs qui y sont cotés sans que cela ait d'effet majeur sur les prix. Plus un marché est liquide, plus il est aisé, rapide et peu coûteux d'y réaliser des transactions.

La solution de nos jours et la vente titre financier ou la levée de fonds pendant une durée indéterminée.

APPLICATION MOONAIE

Description d'un cas, où l'utilisateur se connecte à sa banque pour consulter son compte, et ensuite réaliser de la monnaie.

1. Page d'accueil Banque (Espace-Client)



Consulter



L'utilisateur peut revenir à l'étape précédente et choisir de convertir de la monnaie.

BANQUE CENTRALE

Aujourd'hui le,

<input type="text"/>	Billets de 5	<input type="button" value="[SOA CREER MONNAIE]"/>
<input type="text"/>	Billets de 10	<input type="button" value="[SOA CREER MONNAIE]"/>
<input type="text"/>	Billets de 20	<input type="button" value="[SOA CREER MONNAIE]"/>
<input type="text"/>	Billets de 50	<input type="button" value="[SOA CREER MONNAIE]"/>
<input type="text"/>	Billets de 100	<input type="button" value="[SOA CREER MONNAIE]"/>

Chaque bouton correspond à une servlet. La servlet appelée va créer le nombre de billets d'une somme spécifique. Un message d'informations apparaîtra informant l'utilisateur de la réussite de la fonction, ainsi que la visualisation de sa nouvelle liste de billets.

Faire un dépôt en espèce

DEPOT ESPECE

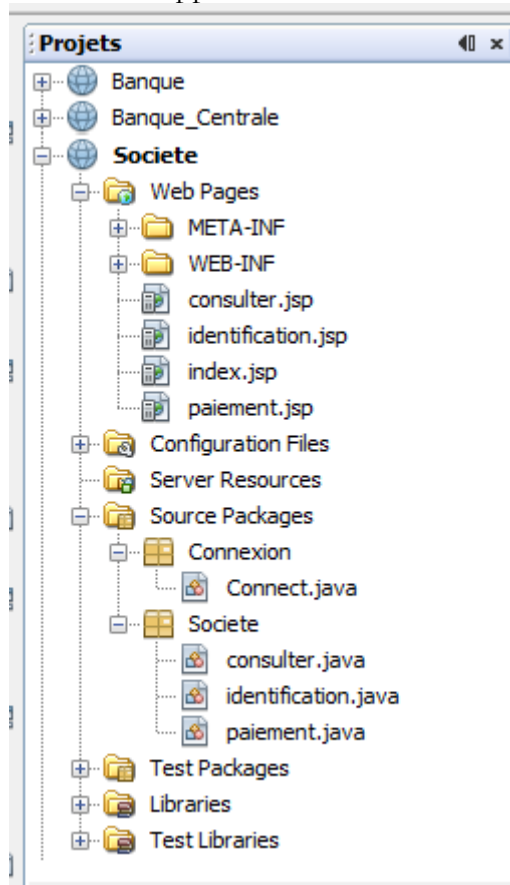
Compte numéro:

<input type="text"/>	Billets de 5
<input type="text"/>	Billets de 10
<input type="text"/>	Billets de 20
<input type="text"/>	Billets de 50
<input type="text"/>	Billets de 100

Mise à jour de la liste de billets du compte.

SCRIPTS

Schéma des applications



Script de connexion

```
protected void processRequest(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    try
    {
        // Elements de connection
        response.setContentType("text/html;charset=UTF-8");
        PrintWriter out = response.getWriter();
        Connection con = null;
        Statement stmt = null;
        ResultSet rs = null;
        response.setContentType("text/html");

        //get the variables entered in the form
        // String Id = request.getParameter("id");
        // String pwd = request.getParameter("pwd");

        // Pilote BDD
        Class.forName("com.mysql.jdbc.Driver");

        // Obtenir une connection vers la BDD
        String url = "jdbc:mysql://localhost/banque";
        con = DriverManager.getConnection(url, "root", "");
```

Récupération des données dans la base de données à l'aide de la commande « SELECT ».

```
//REQUETE récupérant les Id à partir du nom du client
String sql = "SELECT Date, Intitule,Somme,typetransaction.Nom FROM client, transaction, typetransaction " +
            "WHERE transaction.IdTransaction=typetransaction.IdType and client.IdClient=transaction.IdTransaction";
stmt = con.createStatement();
rs = stmt.executeQuery(sql);
```

Affichage des données dans un tableau

```
//Mise en forme de la page de retour
out.println("<h1 align=center>CONSULTATION</h1>");

out.println("<br/>");
out.println("<p align=center>Aujourd'hui le"+new Date().toString()+"</p>");

out.println("<br/>");
// Tableau de réponse
out.println("<table border=1 align=center>");
out.println("<tr>");
out.println("<td>Date</td>");
out.println("<td>Intitule</td>");
out.println("<td>Somme</td>");
out.println("<td>Type</td>");
out.println("</tr>");

while(rs.next())
{
    String d=rs.getString(1);
    String i=rs.getString(2);
    String s=rs.getString(3);
    String n=rs.getString(4);
    // Affichage des données dans le tableau
    out.println("<tr>");
    out.println("<td>"+d+"</td>");
    out.println("<td>"+i+"</td>");
    out.println("<td>"+s+"</td>");
    out.println("<td>"+n+"</td>");
    out.println("</tr>");
}

@Override
protected void doPost(HttpServletRequest request, HttpServletResponse response)
throws ServletException, IOException {
    processRequest(request, response);
}

/**
 * Returns a short description of the servlet.
 * @return a String containing servlet description
 */
@Override
public String getServletInfo() {
    return "Short description";
}
// credits field
```