

UNIVERSITÉ DE CERGY-PONTOISE

DÉCEMBRE 2018

ANALYSE DE PROJET  
Protocole de microtransactions par QRCode

BASE DE DONNÉES  
L3 INFORMATIQUE

GROUPE 16

AURÉLIEN OTTAVIANO - SOLÈNE BISCH -  
MAHFOUD AMHIYEN - AMIRA MAALLOUL

# Contents

<b>A</b>	<b>Cahier des charges</b>	<b>2</b>
A 1	Objectif de la solution . . . . .	2
A 2	Division de la solution en Entités composites . . . . .	2
A 3	L'application mobile . . . . .	2
A 4	Le programme client Machine de Change . . . . .	2
A 5	Le site internet . . . . .	3
A 6	Le programme serveur . . . . .	3
<b>B</b>	<b>Dictionnaire des données</b>	<b>4</b>
<b>C</b>	<b>Schéma E/A</b>	<b>5</b>
<b>D</b>	<b>Rôle des tables</b>	<b>6</b>
<b>E</b>	<b>Schéma relationnel</b>	<b>6</b>
<b>F</b>	<b>Jeu de données</b>	<b>6</b>
<b>G</b>	<b>Exemple de requêtes</b>	<b>10</b>
<b>H</b>	<b>Manuel d'utilisation</b>	<b>11</b>
H 1	Utilisation de l'application . . . . .	11
H 2	Site web . . . . .	12
<b>I</b>	<b>Organisation</b>	<b>13</b>
I 1	Répartition des tâches . . . . .	13
I 2	Planning . . . . .	13
<b>J</b>	<b>Conclusion</b>	<b>14</b>
<b>K</b>	<b>Annexes</b>	<b>14</b>

# A Cahier des charges

## A 1 Objectif de la solution

Le but de la solution à développer est de permettre à l'utilisateur de stocker de l'argent liquide sur un compte, et d'effectuer des transferts d'argent par le biais de QRcodes à scanner. La solution inclut un appareil permettant le transfert de pièces de monnaies vers ou depuis le compte. Pour des raisons de simplicité, la partie physique de la machine sera omise.

## A 2 Division de la solution en Entités composites

- Application mobile
- Programme client Machine de Change
- Programme Serveur
- Site internet

## A 3 L'application mobile

Elle doit permettre:

- La création et la connexion à un compte utilisateur
- L'accès à l'historique des transactions avec un détail sur le montant, la date, et le produit si il existe.
- La génération d'un QRcode permettant de recevoir un montant fixé au préalable
- La lecture d'un QRcode permettant le paiement d'un produit ou service
- La connexion à la machine pour retirer ou ajouter de l'argent sur le compte

L'application devra également contenir un panel de paramètres permettant de modifier les options de sécurité (Code PIN lors d'une transaction, Authentification à deux facteurs), de notifications (Via l'application ou par mail), ou d'affichage(Thème).

De plus, elle contiendra un panel de modification d'informations personnelles. Les informations stockées seront le nom complet, le pseudo, éventuellement une photo/un logo, et l'adresse e-mail.

Le mécanisme de connexion à une machine s'effectue en scannant le code machine et en rentrant le code PIN affiché sur son écran afin d'éviter qu'un utilisateur malveillant ne conserve le QRcode pour le photographier après sans être présent. Une fois connecté, l'utilisateur accède à une vue où il choisit entre retrait et dépôt, et où il peut rentrer le montant correspondant.

La vue principale contient le montant actuellement présent sur le compte ainsi qu'une timeline des transactions. Pour chaque transaction, on affiche le logo du vendeur, le nom de la transaction, le montant, la date, et le type de produit ou service. Elle contiendra également un bouton d'accès rapide à la vue de Scan, et à la vue de génération de QRcode.

La vue de Scan sera une vue de l'appareil photo avec un carré où placer le QRcode. Une fois un code reconnu, si il s'agit de celui d'une machine, on renvoie vers une vue de connexion, sinon on effectue la transaction.

La vue de création d'un QRcode est un formulaire demandant les informations requises pour la réalisation d'une transaction: Le montant, Le nom, et un message facultatif.

## A 4 Le programme client Machine de Change

Ce programme doit fonctionner sans intervention d'un utilisateur, excepté le montant inséré dans la machine que nous rentreront au clavier.

**Comportement du programme** La machine écoute le serveur, tout en affichant un QRCode encodant son identifiant, et lorsqu'elle enregistre une tentative de connexion, elle affiche un code PIN aléatoire et attend la réponse du serveur lui disant si le code fourni par l'utilisateur est correct ou non. Si elle reçoit une réponse positive, elle écoute si on lui envoie une demande de retrait ou des pièces en entrée, et quand l'utilisateur confirme son ordre elle l'effectue et repasse à l'état initial.

## A 5 Le site internet

Il sera constitué d'une page vitrine promouvant la solution, d'une page de contact, d'un espace utilisateur, et d'un espace administrateur.

**Espace utilisateur** Il reprendra sur la page suivant la connexion la fonctionnalité de timeline de l'application. Il disposera également d'une page permettant de générer des QRcodes, ainsi qu'un bouton permettant d'imprimer le QRcode généré. Le panneau de paramètre du site sera similaire à celui de l'Application, mais permettra également de générer un PDF résumant des dépenses triées et filtrées sur une certaine plage de temps.

**Espace Administrateur** Il est destiné à l'organisation qui administre le protocole. La page atteinte après la connexion sera une "page-menu" donnant accès aux différents services. Le site contiendra une liste des machines de change opérées, affichant le solde actuellement présent, un avertissement si le montant minimal souhaité ou si le montant maximum toléré sont atteints. On aura des options pour modérer les machines (bloquer ou limiter les dépôts/retraits, Afficher un message sur l'écran, ouvrir le "coffre-fort"). Si on clique sur une machine, on accède à l'historique des retraits et dépôts, accompagnés d'une courbe retraçant le solde de la machine. Une autre page donnera accès à l'historique global des transactions, avec des outils statistiques financiers (à définir: valeur du réseau en euros, taux de transfert moyen sur x,...). Finalement, on aura une page pour modérer les utilisateurs, notamment si on suspecte qu'ils aient un comportement frauduleux. On pourra les bloquer temporairement ou les expulser du réseau.

## A 6 Le programme serveur

Son rôle principal sera d'assurer la communication entre les autres composantes de façon fiable et sécuriser. Pour les détails concernant les échanges de données/requêtes, se référer au chapitre 2.

**Sécurité du serveur** Les données des requêtes, au lieu d'être envoyées en texte brut, seront encryptées avec un algorithme sha1 salé, afin d'éviter toute écoute non autorisée. Pour croître le niveau de sécurité, le sel peut varier sur les différents canaux de communication, et le serveur peut imposer un changement de sels aux clients.

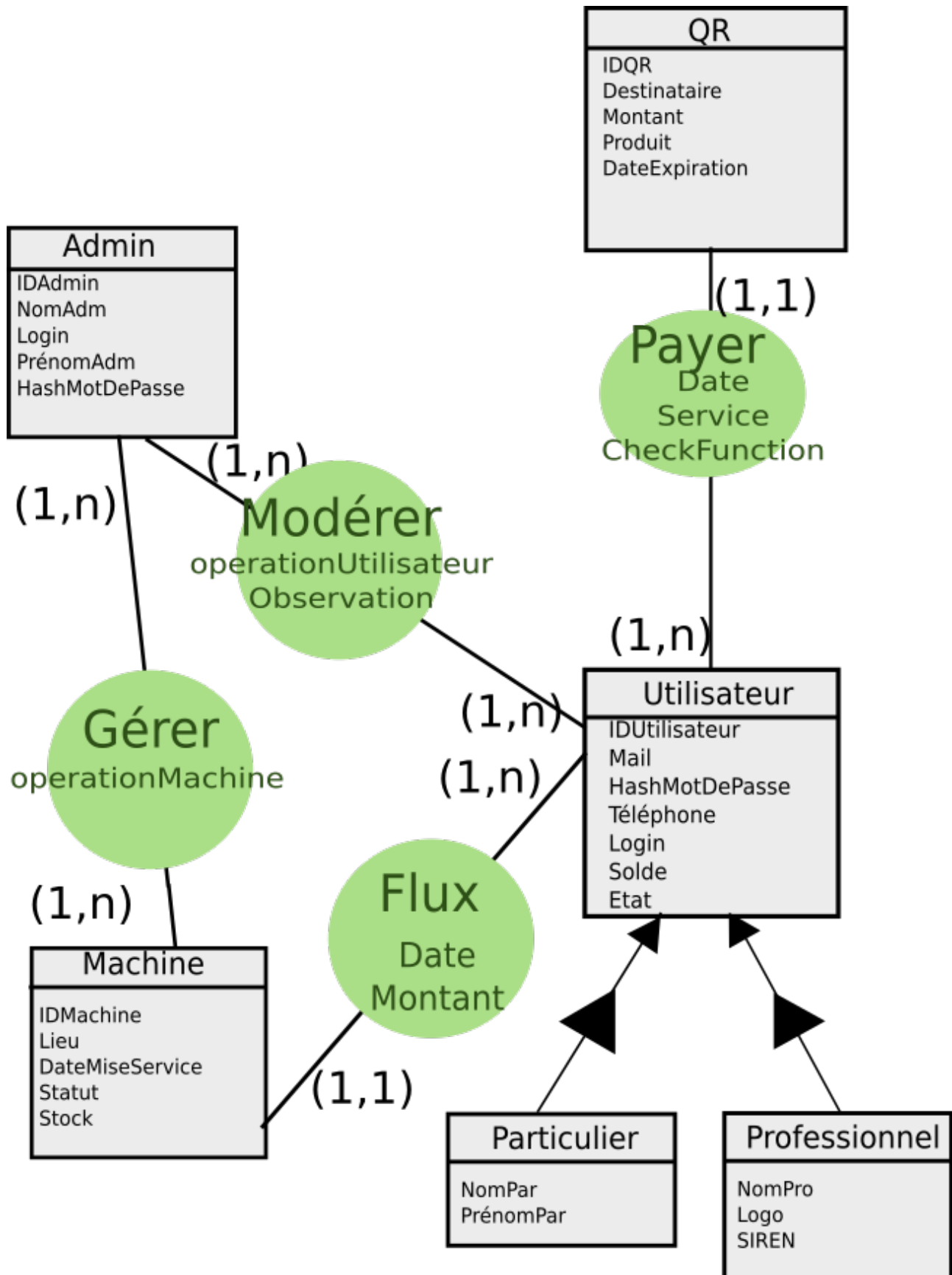
**Fiabilité du serveur** Le programme serveur tournera sur deux machines, l'une étant considérée comme principale, et l'autre comme secondaire. Leur contenu sera synchronisé, et un va-et-vient de requêtes de "check" sera effectué entre les deux machines. Si une requête n'obtient pas de réponse de la part de la machine principale dans un délai à définir, La seconde machine prendra le relais, éliminant les risques liés à une panne de courant par exemple.

## B Dictionnaire des données

NOM	TYPE	DÉFINITION
iDQR	SERIAL	Identifie le QR utilisé pour la transaction
destinataire	INTEGER	Receveur du paiement
montant	FLOAT	Montant de la transaction
produit	VARCHAR(100)	Nom du produit de la transaction
dateExpiration	TIMESTAMP	Vérifie si le QR code a déjà été utilisé ou non
idUtilisateur	SERIAL	Identifiant de l'utilisateur
mail	VARCHAR(100)	Email de l'utilisateur, qu'on utilisera pour lui envoyer les informations à propos de son compte
hashMotDePasse	VARCHAR(256)	Le mot du compte utilisateur crypté lui permettant de se connecter à son compte
téléphone	VARCHAR(10)	Numéro de téléphone de l'utilisateur
nomPro	VARCHAR(50)	Nom de l'utilisateur professionnel
nomPar	VARCHAR(50)	Nom de l'utilisateur particulier
prénom	VARCHAR(50)	Prénom de l'utilisateur particulier
login	VARCHAR(50)	Permet à l'utilisateur de se connecter à son compte
siret	VARCHAR(14)	Permet d'identifier le professionnel qui utilise notre système
idMachine	SERIAL	Permet d'identifier la machine
lieu	VARCHAR(200)	Lieu où la machine est installée
dateMiseService	DATE	La date de mise en service de la machine
stock	FLOAT	La somme d'argent dans la machine
idAdmin	SERIAL	Identifiant de l'administrateur
nomAdm	VARCHAR(50)	Nom de l'administrateur
prénomAdm	VARCHAR(50)	Prénom de l'administrateur
operationMachine	operationM	Le type d'opération que l'administrateur va effectuer sur la machine : il peut par exemple désactiver, ouvrir le coffre, modifier le plafond pour obtenir une alerte
operationClient	operationC	Le type d'opération que l'administrateur va effectuer sur l'utilisateur : il peut par exemple désactiver le compte d'un utilisateur, le supprimer ou le faire passer en compte professionnel.
date	DATE	Date de dépôt ou de retrait d'argent à partir d'une machine
montant	FLOAT	Montant retirer ou déposer dans une machine par un utilisateur
statut	status	Décrit l'état actuel de la machine : si elle fonctionne, en panne ou en réparation.
observation	VARCHAR(256)	Donne la raison de l'action entreprise par l'administrateur
IdDepot	SERIAL	Identifie le dépôt
idOperation	SERIAL	identifie le numéro de l'opération

## C Schéma E/A

Ci-dessus le schéma E/A



## D Rôle des tables

Pour des raisons de clarté on a renommé les tables : modérer par administrationClient, payer par transaction, gérer par administrationMachine.

- qr : Sert à la génération de qrCode utilisé lors des transactions.
- utilisateur : Répertorie les informations concernant l'utilisateur.
- particulier et professionnel : Permettent de faire la distinction entre les différents types d'utilisateurs, nous avons choisi de différencier les utilisateurs particuliers et les professionnels dans l'optique de faciliter les démarches administratives aux utilisateurs professionnels (payer les impôts par exemple).
- transaction : Regroupe toutes les transactions effectuées par les utilisateurs et permettra d'afficher l'historique de transaction pour chaque utilisateur.
- admin : Référence les administrateurs qui vont s'occuper de la gestion des machines et des utilisateurs.
- machine : Répertorie les machines installées.
- administrationMachine : Regroupe toutes les actions affectées par un administrateur sur une machine comme l'approvisionnement en argent, la vérification des stocks.
- administrationClient : Regroupe toutes les actions affectées par un administrateur sur un utilisateur comme la suspension d'un compte et le motif.
- depot : Regroupe tous les dépôts et retraits d'argent effectués par les utilisateurs. Sert à afficher l'historique pour chaque utilisateur.

## E Schéma relationnel

Ci-dessus le schéma relationnel:

- qr(idQR, destinataire, montant, produit, dateExpiration)
- utilisateur(idUtilisateur, mail, hashMotDePasse, telephone, login, etat, solde)
- particulier(#idUtilisateur, nomPart, prenom)
- professionnel(#idUtilisateur, nom, prenom)
- transaction(#idQR, #idUtilisateur, date, service)
- admin(idAdmin, nomAdm, prenomAdm, login, hashMotDePasse)
- machine(idMachine, lieu, dateMiseService, stock, statut)
- administrationMachine(idOperation, #idMachine, #idAdmin operationMachine)
- administrationClient(idOperation, #idAdmin, #idUtilisateur, operationUtilisateur, observation)
- depot(#iddepot, #idUtilisateur, #idMachine, date, montant)

## F Jeu de données

TABLE qr				
IdQR	Destinataire	Montant	Produit	DateExpiration
1	5	0,50	Banane	2018-10-15 14:10:25
2	3	1,20	Coca	2018-10-16 14:50:25
3	2	3,25	Repas RU	2018-10-19 12:16:25
4	1	7,80	Service	2018-10-31 15:10:25

TABLE utilisateur						
IdUtilisateur	Mail	HashMotDepasse	Téléphone	Login	Solde	Etat
1	vatef@test.fr	7c4a8d09ca3762af61e59520943dc26494f8941b	01.01.01.01.01	Vatef	0	maintenu
2	crist@service.fr	7c4a8d09ca3762af61e59520943dc26494f8941b	01.02.02.02.02	Crist5252	0	maintenu
3	bouffe@phonix.fr	7c4a8d09ca3762af61e59520943dc26494f8941b	01.03.03.03.03	bouffeurPhonix	0	maintenu
4	bob@outlook.fr	7c4a8d09ca3762af61e59520943dc26494f8941b	01.04.05.06.07	BobLeBG	0	maintenu
5	mah@service.net	7c4a8d09ca3762af61e59520943dc26494f8941b	01.08.09.10.11	Mahmoud	0	maintenu

TABLE particulier		
IdUtilisateur	Nom	Prénom
5	Mahmoud	Bellala
4	Bob	Guildarts
2	Crist	Lemarc



TABLE Professionnel		
IdUtilisateur	NomPro	SIRET
3	Phonix	14365925932571
1	Vatef	15395715863922

TABLE transaction			
IdQR	IdUtilisateur	Date	Service
1	4	2018-10-16	Banane
2	5	2018-10-15	Coca
3	2	2018-10-19	Repas RU
4	1	2018-10-31	Service

TABLE Admin			
IdAdmin	NomAdmin	PrénomAdmin	HashMotDePasse
1	Lemaire	Marc	8eb59d836774f3a2c0e77378e3401bcc96f54bf3
2	Tao	Jen	8eb59d836774f3a2c0e77378e3401bcc96f54bf3
3	Tram	Tuyet	8eb59d836774f3a2c0e77378e3401bcc96f54bf3

TABLE Machine				
IdMachine	Lieu	DatemiseEnService	Stock	Statut
1	Rotonde STM	2018-10-10	120.1	En service
2	Tour des Chênes	2018-10-12	130.5	En service
3	Neuville	2018-10-14	20.7	Maintenance

TABLE AdministrationMachine			
idOperation	IdAdmin	IdMachine	OpérationMachine
1	1	1	Dépot
2	2	2	Retrait
3	2	3	Maintenance

TABLE AdministrationClient				
IdOpération	IdAdmin	IdUtilisateur	OpérationClient	Observation
1	1	1	Suspendre	NULL
2	2	2	NULL	NULL
3	3	3	Bloquer	Tentative d'escroquerie

TALBE Dépôt			
IdUtilisateur	IdMachine	Date	Montant
3	2	2018-20-10	6,33

## G Exemple de requêtes

# H Manuel d'utilisation

## H 1 Utilisation de l'application

**Connexion** Afin de pouvoir utiliser l'application au mieux il faut :

- Se connecter à la connexion de la fac : uniquement u-cergy-wifi. L'application ne marchera pas avec les autres wifi (invités, personnels etc...)
- Ne pas oublier d'aller demander un jeton sur le lien : 10.40.128.22:8029. Sans le jeton, il est impossible de se connecter à l'application.
- Aller sur l'application, entrer vos identifiants et mots de passe (ou créer un compte si besoin).
- Vous pouvez à présent utiliser l'application sans problèmes.

**Génération d'un Qr Code** Afin de générer un qr code il faut :

- Cliquer sur le bouton "Generate" qui vous ramène sur une nouvelle fenêtre;
- Il faut remplir les champs de textes comme suit
  - \* Recipient : Le login de la personne à qui est destinée le service ou l'objet que vous comptez vendre.
  - \* Amount : Le prix de l'objet ou du service.
  - \* Service : Le nom du service ou de l'objet que vous vendez. Mettre Null ou service si vous n'avez pas envie de mettre de nom.
  - \* cliquer sur le Bouton "Generate". Si le nom de la personne que vous avez rentré est exacte (existe dans la base de donnée), un Qr code apparait en bas de la fenêtre. Sinon, l'application vous invite à vérifier le nom de la personne.



Vatef

Service

1.47

GENERATE

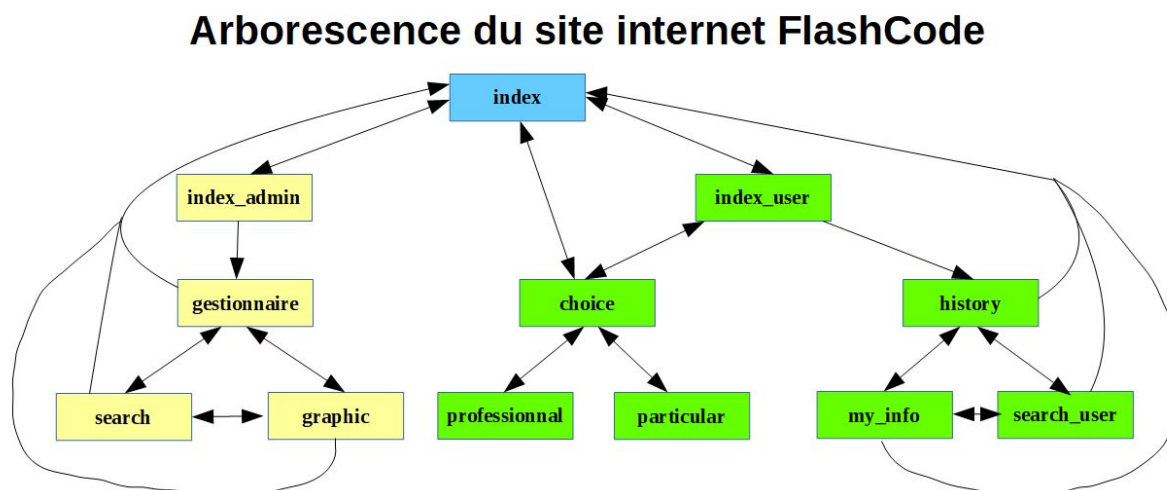


**Scanner un Qr Code** Pour scanner code, il suffit de :

- Cliquer sur le bouton "Scan" à partir de la page principale.
- Diriger votre smartphone vers le Qr Code, le Qr code sera automatiquement scanné. Si le Qr code est valable, vous serez directement redirigé vers la fenêtre principale (avec l'historique des transactions), sinon un message d'erreur va s'afficher et vous serez redirigé vers la fenêtre principale.

## H 2 Site web

### Plan du site



Pour vous rendre sur le site :

- Connecter vous au wii-fi de l'Université de Cergy-Pontoise : u-cergy-wifi.
- Demander un jeton en tapant <http://10.40.128.22:8029> dans votre navigateur.
- Taper l'URL <https://10.40.128.22/> mamhiyen dans votre navigateur

Une fois sur le site : Choisir votre mode de connexion :

- Si vous êtes gestionnaire
  - Cliquer sur la case "gestionnaire"
  - Rentrer vos identifiants puis vous accéderez aux pages dédiées à l'espace
  - Dans le but pédagogique de tester notre site voilà l'identifiant d'un gestionnaire :
    - \* Login : Lemaire
    - \* Mot de passe : A123456\*
- Si vous êtes utilisateur
  - Cliquer sur la case "utilisateur"
  - Si vous avez déjà un compte :
    - \* Cliquer sur la case "particulier" ou "professionnel" selon votre profil
    - \* Rentrer vos identifiants puis vous accéderez aux pages qui sont dédiées au profil de votre connexion
    - \* Dans le but pédagogique de tester notre site voilà les identifiants d'un particulier et d'un professionnel :
      - Particulier : Login : Crist5252 et Mot de passe : 123456
      - Professionnel : Login : Vatef et Mot de passe : 123456
  - Si vous n'avez pas de compte
    - \* Cliquer sur le lien "veuillez en créer un"
    - \* Choisir votre profil (particulier ou professionnel)
    - \* Compléter le formulaire et valider le en cliquant sur "submit"
    - \* Suivre les instructions de connexion précédentes "Vous êtes utilisateurs et vous avez déjà un compte"

## I Organisation

### I 1 Répartition des tâches

Mahfoud	Application mobile
Aurélien	Code C de la connexion C ↔ postgres + requêtes en C
Amira	Création de la base de donnée + jeu de données
Solène	Site Internet

### I 2 Planning

	Septembre	Octobre	Novembre	Décembre
Remise du Rapport présentation du projet (Contexte, Schémas, Dictionnaire de données)				
Code SQL de création + jeu de données				
Code C de la connexion C ↔ postgres				
Modification + remise rapport modifiés avec le jeu de données en plus				
Développement de l'application mobile				
Création du site internet				
Requêtes en C				
Vérification du bon fonctionnement + complétude du dossier final				
Remise Projet final				7 décembre
Oral du projet				12 décembre

# J Conclusion

## K Annexes

### K 1 Code Sql



DROP TABLE qr, utilisateur, particulier, professionnel, machine, transaction, admin, administrationClient, administrationMachine, depot CASCADE;

DROP TYPE state, status, operationC, operationM CASCADE;

```
CREATE TABLE qR(  
    iDQR SERIAL NOT NULL,  
    destinataire INTEGER NOT NULL,  
    montant FLOAT NOT NULL,  
    PRODUIT VARCHAR(100),  
    dateExpiration TIMESTAMP,  
    CONSTRAINT qr_pk PRIMARY KEY (iDQR)  
);
```

CREATE TYPE state AS ENUM ('Maintenu','Suspendu','Bloque');

```
CREATE TABLE utilisateur(  
    idUtilisateur SERIAL NOT NULL,  
    mail VARCHAR(100) NOT NULL,  
    login VARCHAR(50) NOT NULL,  
    telephone VARCHAR(10) NOT NULL,  
    hashMotDePasse VARCHAR(256) NOT NULL,  
    etat state,  
    solde FLOAT NOT NULL,  
    CONSTRAINT utilisateur_pk PRIMARY KEY (idUtilisateur)  
);
```

```
CREATE TABLE particulier(  
    idUtilisateur INTEGER NOT NULL,  
    nomPar VARCHAR(50) NOT NULL,  
    prenom VARCHAR(50) NOT NULL,  
    CONSTRAINT particulier_pk PRIMARY KEY (idUtilisateur)  
);
```

```
CREATE TABLE professionnel(  
    idUtilisateur INTEGER NOT NULL,  
    nomPro VARCHAR(50) NOT NULL,  
    siret VARCHAR(14) NOT NULL,  
    CONSTRAINT professionnel_pk PRIMARY KEY (idUtilisateur)  
);
```

```
CREATE TABLE transaction(  
    iDQR INTEGER NOT NULL,  
    idUtilisateur INTEGER NOT NULL,  
    date TIMESTAMP NOT NULL,  
    service VARCHAR(100),  
    CONSTRAINT transaction_fk1 FOREIGN KEY (idUtilisateur) REFERENCES  
utilisateur(idUtilisateur),  
    CONSTRAINT transaction_pk PRIMARY KEY (iDQR, idUtilisateur)  
);
```

```
CREATE TABLE admin(  
    idUtilisateur SERIAL NOT NULL,  
    login VARCHAR(50) NOT NULL,  
    password VARCHAR(50) NOT NULL,  
    CONSTRAINT admin_pk PRIMARY KEY (idUtilisateur)  
);
```

```

        idAdmin SERIAL NOT NULL,
        nomAdm VARCHAR(50) NOT NULL,
        prenomAdm VARCHAR(50) NOT NULL,
        hashMotDePasse VARCHAR(256) NOT NULL,
        CONSTRAINT admin_pk PRIMARY KEY (idAdmin)
    );

CREATE TYPE status AS ENUM ('En service','Desactive', 'Maintenance');

CREATE TABLE machine(
    idMachine SERIAL NOT NULL,
    lieu VARCHAR(200) NOT NULL,
    dateMiseService DATE,
    stock FLOAT NOT NULL,
    statut status,
    CONSTRAINT machine_pk PRIMARY KEY(idMachine)
);

CREATE TYPE operationM AS ENUM ('Depot','Retrait','Maintenance');

CREATE TABLE administrationMachine(
    idOperation SERIAL NOT NULL,
    idAdmin INTEGER NOT NULL,
    idMachine INTEGER NOT NULL,
    operationMachine operationM NOT NULL,
    CONSTRAINT administrationMachine_fk1 FOREIGN KEY (idAdmin) REFERENCES
admin(idAdmin),
    CONSTRAINT administrationMachine_fk2 FOREIGN KEY (idMachine) REFERENCES
machine(idMachine),
    CONSTRAINT administrationMachine_pk PRIMARY KEY (idOperation)
);

CREATE TYPE operationC AS ENUM ('Suspendu','Bloque','Maintenu');

CREATE TABLE administrationClient(
    idOperation SERIAL NOT NULL,
    idAdmin INTEGER NOT NULL,
    idUtilisateur INTEGER NOT NULL,
    operationClient operationC,
    observation VARCHAR(256),
    CONSTRAINT administrationClient_fk1 FOREIGN KEY (idAdmin) REFERENCES
admin(idAdmin),
    CONSTRAINT administrationClient_fk2 FOREIGN KEY (idUtilisateur) REFERENCES
utilisateur(idUtilisateur),
    CONSTRAINT administrationClient_pk PRIMARY KEY (idOperation)
);

CREATE TABLE depot(
    idDepot SERIAL NOT NULL,
    idUtilisateur INTEGER NOT NULL,
    idMachine INTEGER NOT NULL,
    date DATE NOT NULL,

```

```

montant FLOAT NOT NULL,
CONSTRAINT depot_fk1 FOREIGN KEY (idUtilisateur) REFERENCES
utilisateur(idUtilisateur),
CONSTRAINT depot_fk2 FOREIGN KEY (idMachine) REFERENCES
machine(idMachine),
CONSTRAINT depot_pk PRIMARY KEY (idDepot)
);
ALTER SEQUENCE admin_idadmin_seq RESTART WITH 1;
ALTER SEQUENCE administrationclient_idoperation_seq RESTART WITH 1;
ALTER SEQUENCE administrationmachine_idoperation_seq RESTART WITH 1;
ALTER SEQUENCE depot_iddepot_seq RESTART WITH 1;
ALTER SEQUENCE machine_idmachine_seq RESTART WITH 1;
ALTER SEQUENCE qr_idqr_seq RESTART WITH 1;
ALTER SEQUENCE utilisateur_idutilisateur_seq RESTART WITH 1;

INSERT INTO utilisateur(mail, login, telephone, hashmotdepasse,etat, solde)
VALUES('vatef@test.fr','Vatef',
'101010101','7c4a8d09ca3762af61e59520943dc26494f8941b', 'Maintenu', '0'),
('crist@service.fr','Crist5252',
'102020202','7c4a8d09ca3762af61e59520943dc26494f8941b','Maintenu', '0'),
('bouffe@phonix.fr','bouffeurPhonix',
'103030303','7c4a8d09ca3762af61e59520943dc26494f8941b','Maintenu', '0'),
('bob@outlook.fr','BobLeBG',
'104050607','7c4a8d09ca3762af61e59520943dc26494f8941b','Maintenu', '0'),
('mah@service.net','Mahmoud',
'108091011','7c4a8d09ca3762af61e59520943dc26494f8941b','Maintenu', '0');

INSERT INTO professionnel(idutilisateur, nompro, siret)
VALUES('1','Vatef','5181684329850'),
('3','phonix','0349203910392');

INSERT INTO particulier (idutilisateur, nompar, prenom)
VALUES ('2', 'Crist', 'Lemarc'),
('4','Bob', 'Guildart'),
('5','Mahmoud', 'Bellala');

INSERT INTO qR (destinataire, montant, produit, dateexpiration)
VALUES('3', '0.5', 'Banane', '2018-10-15 14:10:25'),
('3', '1.20', 'Coca', '2018-10-16 14:50:25'),
('2', '3.25', 'Repas RU', '2018-10-19 12:16:25'),
('1', '0.5', 'Service', '2018-10-31 15:10:25');

INSERT INTO transaction (idqr,idutilisateur,date, service)
VALUES('1','1','2018-10-15 15:10:33.000000','Banane'),
('2','5','2018-10-16 16:30:50.000000','Coca'),
('3','2','2018-10-19 12:32:50.000000','Repas RU'),
('4','3','2018-10-16 10:04:20.000000','Service');

INSERT INTO admin (nomadm, prenomadm, hashmotdepasse)
VALUES('Lemaire', 'marc', '8eb59d836774f3a2c0e77378e3401bcc96f54bf3'),
('Tao', 'Jen', '8eb59d836774f3a2c0e77378e3401bcc96f54bf3'),

```

```
('Tram', 'Tuyet', '8eb59d836774f3a2c0e77378e3401bcc96f54bf3');
```

```
INSERT INTO machine (lieu,datemiseservice,stock,statut)
VALUES('Rotande STM', '2018-10-10', '120.1', 'En service'),
('Chênes', '2018-10-12', '130.5', 'En service'),
('Neuville', '2018-10-14', '15.6', 'Maintenance');
```

```
INSERT INTO administrationMachine (idadmin,idmachine,operationmachine)
VALUES('1','1', 'Depot'),
('2','3', 'Retrait'),
('3','2', 'Maintenance');
```

```
INSERT INTO administrationClient (idadmin,idutilisateur,operationclient,observation)
VALUES('1', '1', 'Suspendu', NULL),
('3', '2', NULL, NULL),
('2', '3', 'Bloque', 'Tentative escroquerie');
```

```
INSERT INTO depot (idutilisateur, idmachine, date, montant)
VALUES('3','2','2018-10-20','6.33');
```