

Give 1 Get 2 - Give One Get Two

MOCHA - MoneyBookers Exchange

Vision :

Le logiciel Give1Get2 a été créé pour répondre au besoin de financement des agents économiques au niveau européen et mondial pour pouvoir financer, en outre, la recherche, l'éducation et l'innovation.

Ce logiciel est conçu pour construire un marché financier alternatif aux marchés financiers existants. Le logiciel est spécialisé dans la levée de fonds en ligne. Il facilite le libre échange dans l'économie internationale.

La plateforme ne prend pas de pourcentage sur les fonds qu'elle lève. En ce sens, une organisation qui l'utiliserait pourrait être du type "organisation non profitable".

Objectif :

Faire un projet pilote (fonctionnel, gratuit dans sa mise en place, légal, disponible (7/7j, 24/24h) et reposant sur une situation gagnant-gagnant) d'une plateforme de trading internationale (FSX) à partir d'une API de paiement (Moneybookers) et d'un logiciel financier libre (FreePay) pour permettre d'augmenter le solde des participants. (+1 Euro à chaque fois)

Business Model : Accord Gagnant-Gagnant tripartite

L'intérêt de l'internaute est de faire du profit pour pouvoir s'acheter des biens qui répondent à ses besoins. L'internaute a donc besoin d'une plateforme de trading.

Une plateforme de trading (comme le FSX) ne peut exister sans institution financière, sans l'API fournie par celle-ci, et sans ses utilisateurs.

Les systèmes de paiement ont un modèle économique fondé sur les commissions sur les transactions de leurs adhérents. (environ 2%) Le profit est directement proportionnel au volume de transactions. Pour augmenter leur profit, ils doivent augmenter leur volume de transactions. Ceci se fait par l'accès de leur système d'informations : l'API (Application Programming Interface) qui permet à quiconque d'automatiser les paiements.

Supplément au Business Model :

Traditionnellement, un projet qui utilise une API met le logo de l'API en évidence et un lien vers celle-ci.

"Refer a friend program" : le logo est un lien avec l'id du marchand qui a mis en place le serveur. Le marchand gagne 30% des frais engendré par ses utilisateurs parrainés.

Chaque utilisateur peut aussi parainer d'autres utilisateurs lors de la vente de titre (lien automatique).

Existant :

- FreePay : <http://freepay.fr/>

Fonctionnement de la plateforme de trading (version courte) :

Le système de trading repose sur une stratégie gagnant-gagnant. La plateforme de trading est un lieu de confrontation de l'offre de titres financiers et de la demande d'argent en application de la pensée du laissez-faire économique. C'est un jeu à somme nulle du point de vue de la plateforme de paiement mais pas du système de trading (1 euro titrisé symbolique émit pour l'amorçage, échange +1 +1 pour chaque partie à chaque transaction). C'est un cercle vertueux. C'est un système de personne à personne (P2P) complémentaire qui permet aux utilisateurs de passer des ordres sur une plateforme de paiement. Cela a été conçu de tel manière qu'il n'y a pas de risque de variation inverse. Puisqu'il n'y a pas d'ordre de vente, il ne peut y avoir de crack boursier. Tout les utilisateurs du système peuvent s'enrichir, mais pas au même moment. L'utilisateur peut donc devenir, selon ce qu'il en fait, un trader. (fr : Opérateur de marché)

Le profit est aussi paramétrable (de 0 à 100 %, exemple : 25 euros réel deviennent à 100% -> 50 euros titrisés). Cela permet à l'utilisateur de spéculer ou de ne pas le faire. Ce qui permet à l'utilisateur de transformer ses capitaux en plus de capitaux représenté par les titres. (Et ensuite transformer ses titres en capitaux par l'acte de vente, exemple : 50 euros titrisés -> 50 euros réel). D'où : 25 euros réel -> 50 euros réel. Ce qu'il fallait démontrer.

- Moneybookers : <http://moneybookers.com/>

Moneybookers Ltd est une société enregistrée au registre du Commerce de l'Angleterre et du Pays de Galles sous le No 4260907. Siège social: Welken House, 10-11 Charterhouse Square, London, EC1M 6EH.

Avertissement légal: réglementé par l'Electronic Commerce (EC Directive) Regulations 2002 & Consumer Protection (Distance Selling) Regulations 2000

(<http://www.moneybookers.com/app/help.pl?s=ecrcpr>)

Les frais : MoneyBookers prend des commissions variables par transactions.

- la FSA : <http://www.fsa.gov.uk/>

Numéro d'enregistrement de Moneybookers auprès de la FSA : 214225
(<http://www.fsa.gov.uk/register/firmSearchForm.do>)

Current status: Authorised

Effective Date: 05/02/2003

Address: 10 -11 Charterhouse Square, London, EC1M 6EH

Notices: UK authorised firms who have a deposit-taking permission do not usually need to obtain separate permission from the FSA to receive money from clients.

L'émission de monnaie électronique au Royaume Uni est une activité régulé par le Financial Services and Markets Act 2000 : http://www.opsi.gov.uk/acts/acts2000/ukpga_20000008_en_1

, par la directive européenne sur le e-commerce de 2002 (ISBN 0 11 042643 6 : <http://www.opsi.gov.uk/si/si2002/20022013.htm>) et par la directive européenne sur la monnaie électronique. (ci nommé E-Money Directive : Directive 2000/46/CE du Parlement européen et du Conseil du 18 septembre 2000 concernant l'accès à l'activité des établissements de monnaie électronique et son exercice ainsi que la surveillance prudentielle de ces établissements http://ec.europa.eu/internal_market/bank/e-money/index_en.htm)

Réalisé : L'objectif principal a été réalisé.

Le logiciel a été testé dans des conditions réelles avec des comptes personnels pour les 3 cas possibles d'achat (inférieur, égal et supérieur) : et dans chacun des cas, cela à fonctionné parfaitement.

Exemple simple : lorsqu' 1 Euro est envoyé, 2 Euros sont reçus (facteur 2 : $1*2=2$). La devise "Give 1. Get 2." a été choisie en ce sens. Le montant de départ est choisi par l'utilisateur et il peut utiliser le système autant de fois qu'il veut. exemple : 1234 Euros envoyé -> 2468 Euros reçus.

Le logiciel va vers la simplicité puisque, par rapport à la version précédente (FreePay), il soustrait les processus nécessaires à la gestion de l'argent. Le logiciel est un système ouvert qui envoie des informations à l'intérieur (historique) et à l'extérieur (ordre). Le logiciel peut être vu comme un plugin qui interagit avec le logiciel principal (l'institution financière) pour lui apporter une nouvelle fonctionnalité.

La division des rôles :

- Le logiciel permet de passer des ordres de paiement, d'échanger des ordres de paiement et de faire des demandes de paiement.
- L'institution financière transforme les capitaux entrant en monnaie électronique, exécute les paiements et sort les chèques.

Ce logiciel permet d'économiser le coût du capital initial (1 million d' Euros) nécessaire à la création d'une institution financière émettrice de monnaie électronique (en : E-money

issuier) au sein de l' Union Européenne. (ref : Article 4 paragraphe 1 de la Directive 2000 46 CE).

Indépendance : Chaque organisation qui met en place le logiciel Give1Get2 est autonome des autres organisations. Elle n'est dépendante que de l'institution financière qui permet de faire les paiements (Relation 1 à plusieurs).

Explication Économique : Le logiciel n'a pas pour but de créer de l'inflation.

D'après ce que je sais, il existe deux types d'inflation :

- L'inflation du niveau des prix : Une hausse des prix des biens et des services durant une période de temps donné. (source : Wikipédia) Le revenu augmente moins vite que la hausse des prix des produits. → diminution du pouvoir d'achat. Ce qui est problématique. Or, ce que propose Give1Get2, c'est d'augmenter le revenu des acteurs du système, d'où une augmentation du pouvoir d'achat. Il n'y a pas de services en vente sur la plateforme et l'utilisation y est gratuite. Les « produits » financiers en vente sont des demandes de paiements. Le prix d'achat est fixé par les acheteurs eux-même. S'ils décident d'acheter plus cher, c'est pour gagner plus.

- L'inflation de la masse monétaire : Lorsque la masse monétaire mondiale subit une création monétaire. L'argent en circulation augmente via les taux d'intérêts. Or, le logiciel Give1Get2 n'a pas vocation d'augmenter la masse monétaire ni de la diminuer. Il n'y a pas de taux d'intérêts non plus dans ce système. Car c'est uniquement du ressort des banques.

Le logiciel permet la circulation de la monnaie entre les acteurs économiques.

Les frais par transactions :

Prenons cet l'exemple : Bob veut envoyer 10 centimes d'EUR à Alice via Moneybookers.

Alice à un solde de 10.82 EUR. Bob a un solde de 78.19 EUR. Bob envoi 10 centimes à Alice. Le solde de Bob devient 78.09 EUR. Le solde de Alice passe à 10.92 EUR. C'est sans compter les frais variables (0.01) et les frais fixes (0.29). En fin de compte, le solde de Alice est de 10.62 EUR à la fin de l'opération.

Personnes	Alice		Bob	
Temps	Avant	Après	Avant	Après
Solde	10.82 EUR	10.62 EUR	78.19 EUR	78.09 EUR

On constate que le gain (10 centimes) est inférieur à la perte (30 centimes). Bob a été chargé négativement (-0.10 EUR, ce qui est normal). Alice a été aussi chargé négativement (total de -0.20 EUR). C'est donc une situation perdant-perdant.

Pour obtenir une situation perdant-gagnant, on peut mettre en place un quota d'argent minimum automatique à définir dans les paramètres pour tout achat via l'API. Testons : Bob (solde : 78.09 EUR), après mûres réflexions, veut envoyer 3 Euros à Alice (solde : 8.52 EUR) via Moneybookers. Bob a maintenant un solde de 75.09 EUR. Alice a un solde d'environ 11.12 EUR (8.52 +3.00 -0.10 -0.29).

Personnes	Alice		Bob	
Temps	Avant	Après	Avant	Après
Solde	8.52 EUR	11.12 EUR	78.09 EUR	75.09 EUR

Que constatons nous ? Bob a été chargé négativement (-3.00 EUR) tandis que Alice a été chargée positivement (différence de +2.60 EUR). Ce qui est une situation perdant-gagnant.

Comme je l'ai démontré dans mon [précédent post](#) sur mon blog, on peut transformer une situation perdant-gagnant en une situation gagnant-gagnant. Ce que nous allons faire maintenant.

Télécharger

Le logiciel se base sur une politique de transparence et de développement durable. La licence choisie est la GNU GPL. C'est un logiciel libre. De ce fait, il a été mis à disposition gratuitement sur SourceForge.net pour être téléchargé et installé sur des serveurs en ligne.

* Télécharger directement la solution complète (la plus à jour) :

<http://give1get2.com/give1get2.zip>

<http://give1get2.com/give1get2.7z>

* Miroir de téléchargement complet (version stable) :

<http://sourceforge.net/projects/give1get2/>

* [Moneybookers] Démonstration en action :

http://www.moneybookers.com/app/help.pl?s=m_gateway_demo

https://www.moneybookers.com/app/test_payment.pl

* [Moneybookers] La documentation en libre téléchargement:

http://www.moneybookers.com/merchant/fr/moneybookers_gateway_manual.pdf

http://www.moneybookers.com/merchant/fr/automated_payments_interface_manual.pdf

Installation du logiciel

1. Acheter un nom de domaine (maplateformedetradingdexemple.com) auprès d'un Registrar.
2. Obtenir un hébergement contenant un espace suffisant (environ : 50 Mo) et une bande passante suffisante (plusieurs Giga) en fonction du nombre d'utilisateur prévu. (et POP3, FTP et MySQL inclus)
3. Dé-compacter les fichiers précédemment téléchargé. (procédure ci-dessus)
4. Modifier le fichier 'scripts sql tables & champs.sql' : à la ligne 85, remplacer l'email (alice@give1get2.com) et l' id marchand (6173206) par votre email et votre id marchand obtenu auprès de moneybookers à l'ouverture d'un compte. Remplacer aussi le courriel par votre courriel à la ligne 108.
5. Créer une base de données « mocha » (sans les doubles quotes) dans votre panneau d'administration (généralement à l'URL <http://maplateformedetradingdexemple.com/phpmyadmin/>)

Créer un utilisateur et lui donner les droits d'accès à la base de données en lecture et écriture. (si ce n'est pas déjà fait automatiquement).
6. Cliquer sur l'onglet SQL, copier-coller les données du fichier 'scripts sql tables & champs.sql' dans le champ de saisie et cliquer sur Exécuter. Aucun message d'erreur ne doit s'afficher.
7. Modifier les valeurs par défaut par celles qui ont été fournies par l'hébergeur dans le fichier 'params.php' (sans les simple quotes) ligne 31 (hôte), ligne 32 (utilisateur), ligne 33 (mot de passe), ligne 34 (base de données)
8. Sur le serveur web, copier-coller la source modifié (avec les params par default) en [7z](#) et zip. Créer aussi un dossier /give1get2/ . Importer les fichiers via un FTP (ex [fireFTP](#), une extension de Firefox) avec les paramètres de l'hébergeur ('params.php') dans le répertoire précédemment crée.
9. Lancer dans le navigateur <http://maplateformedetradingdexemple.com/give1get2/>. La page d'index doit s'afficher sans message d'erreur. Les sources (7z et zip) doivent pouvoir être téléchargeable à partir d'un onglet 'documentation' ou 'téléchargement'.
10. Référencer votre site sur les moteurs de recherche (ex : <http://www.google.com/addurl/?continue=/addurl>)
11. Générer un sitemap XML, et le mettre à la racine (ex : <http://www.xml-sitemaps.com/>)
12. Optimiser votre site (ex: avec [Google Webmaster Tools](#))

Pré-requis :

Conçu en XHTML, CSS, JavaScript, PHP et SQL (CRUD). Nécessite un MySQL, phpMyAdmin, POP3 pour les mails, FTP et un navigateur web. A été testé et fonctionne avec Apache (>= 1.3.33), Mysql (>= 4.1.9), PHP (>= 4.3.10), PhpMyAdmin(>= 2.6.1).

L'architecture est en trois tiers (donnée, business logic et présentation). L'architecture se base sur le projet FreePay : <http://freepay.fr/freepay.zip> Pour approfondir, [la documentation](#) de freepay est à votre disposition.

Pour les non anglophones ou non francophone, il existe des outils de traductions comme [Google translate](#) pour lire des documents dans votre langue.

Le diagramme des traitements des processus marchands du côté de l'institution financière a été copié dans une image jointe au logiciel.

"moneybookers_payment_gateway_api_details_interaction_diagram.png"

Ce script se base également sur la documentation de l'API "Moneybookers Payment Gateway - Merchant Integration Manual" (en anglais)- Version: <6.5>. http://www.moneybookers.com/merchant/fr/moneybookers_gateway_manual.pdf (43 pages) Une nouvelle version est peut être sortie à l'heure ou vous lisez ces lignes, ce qui pourrait poser des problèmes.

Le script ne prend en compte que les champs obligatoires par mesure de simplification. Pour ce qui est des champs facultatifs : se référer à la documentation officielle.

Pour d'avantage d'informations sur moneybookers, vous pouvez consulter l'[article éponyme](#) sur mon blog.

Le code ISIN est de nouveau utilisé dans ce logiciel (ISO 6166).

Les statuts d'une transaction (états) :

- 2 : validé (processed)
- 1 : en attente (pending)
- 0 : test
- 1 : annulé (canceled)
- 2 : Échoué
- 3 : Retour arrière

Les Étapes du développement (le Comment)

Vision : Le module de paiement de FreePay et tout les modules de niveau inférieur sont remplacés par l'API externe de Moneybookers (institution financière validé par la FSA).

Conséquences : Le support du module de paiement est donc externalisé. Le nom du projet ne correspond plus. Les balises Méta ne correspondent plus. Les menus ne correspondent plus.

1. Copie de Freepay. Tout les modules non nécessaires sont retirés sauf le FSX. (+ vérifier que ça marche)
2. Implémentation de l'API Moneybookers (+ vérifier que ça marche)
3. Mashup des deux (Application composite)
 - création de l'interface spécifique
 - création des processus spécifique
 - création de la base de donnée spécifique (+ vérifier que ça marche via un simulateur de moteur de paiement)
4. Re-factorisation : Optimisation de la Mashup (+ vérifier que ça marche)
5. Publication sur internet

L'architecture

Le code source est en français. Les commentaires sont en français. Sauf pour les standards financiers qui sont en anglais. Le projet s'oriente vers une internationalisation (I18N).

Toutes les images sont dans un dossier spécifique (/images). Tout les CSS (Feuilles de style en cascade) sont dans un dossier spécifique (/style). Tout ce qui a attiré au support utilisateur est dans le dossier /support. La documentation est dans le dossier /support/docs.

La programmation est de type procédurale : les méthodes sont appelés dans un ordre spécifique.

La partie visible est composé de la page principale, de l'historique et de la documentation.

La limitation d'accès se fait grâce aux sessions pour la partie caché (processus de paiement).

Comme c'est un projet web, l'interface homme-machine est basé sur une architecture est de type client/serveur. Et, le serveur à une architecture en 3 tiers (base de données, traitements, présentation).

La base de donnée

Elle est composée de 4 tables par institution financière : ("scripts sql tables & champs.sql" présent à la racine)

-2 pour les titres (déjà existante dans le FSX) :

- liste des transactions passés effectués sur des titres.
- liste des titres en vente dans leur états actuels.

- 2 pour les ordres de paiements (l'avant et l'après) :

- liste des ordres passés sur les titres du FSX.
- liste des ordres de paiements renvoyé par l'api.

Elle est composé de 2 insertions dans ces tables correspondant à un titre et de son historique.

Les étapes du processus de paiement pour un utilisateur dans le FSX

L'utilisateur a crée et approvisionné un compte Moneybookers.
(<https://www.moneybookers.com/app/register.pl>)

1. Le lien vers la page de connexion est sur la première page en haut à droite. La connexion se fait avec l'Email et l'identifiant numérique (Customer ID) associé au compte Moneybookers.

(Visible en dessous du menu dans l'interface Moneybookers). (Pas de mot de passe, il est gérée par Moneybookers lors du paiement pour éviter une récupération externe)

2. L'utilisateur va sur la page d'achat et sélectionne un titre (définit aussi la valeur et le profit si besoin).

L'ordre d'achat est sauvegardé et accessible via le menu du même nom. Il récapitule l'état de la transaction (En cours / En attente, Échoue / Annulé ou Fini).

3. Selon les paramètres ("params.php") défini par l'administrateur, l'utilisateur est redirigé soit :

- sur le serveur de test (paiement/pay/mb_test_payment.php) et valide
- soit sur Moneybookers (<https://www.moneybookers.com/app/payment.pl>)

La redirection se fait grâce à un GET (passage de tout les paramètres obligatoires). La solution qui a été choisie est une redirection javascript. (<script type='text/javascript'>Code</script>)

4. Si la personne a donnée les bons paramètres, elle entre juste son mot de passe. Sinon, elle crée son compte.

5. Le paiement est annulé ou validé par l'utilisateur. Moneybookers transfère l'argent entre les parties. Moneybookers envoie un code de retour "HTTP 200" en POST uniquement à la

status_url précédemment envoyé ("paiement/pay/status_report.php") et redirige l'utilisateur vers la plateforme de FSX défini précédemment.

- Le logiciel utilise un couplage par données (passage de paramètres).

```
if( isset($_POST['mb_transaction_id']) ) { $mb_transaction_id =  
$_POST['mb_transaction_id']; }
```

- Les données sont ensuite filtrées par mesure de sécurité. ("status_report_filtre.php") Il vérifie que tout les champs sont renseignés, qu'ils ont le bon type, la bonne longueur et les bonnes données (technique de filtrage par liste blanche).

exemple : vérification que l'origine des données (IP) provient bien de l'institution financière pour éviter toute tentative frauduleuse d'envoi d'argent fictif. (attaque de type "man in the middle")

Le montant renvoyé par moneybookers peut être inférieur au montant indiqué dans la réservation (du fait de la prise de commission (fixe et variable) de la plateforme de paiement ou des paramètres GET qui peuvent être modifié en cours de route par l'internaute). Le transfert de possession des titres se fait compte tenu des nouveaux chiffres envoyé par l'API uniquement afin d'éviter ces problèmes.

6. Ensuite, si tout les critères sont validés alors les données sont stockés dans la table de l'api. (+ date actuelle) Et affichable au besoin par le fichier 'pay_liste.php' que le client retrouve sous ses yeux lorsqu'il retourne sur Give1Get2.

7. continuation de processus de transfert de possession (la transaction en attente "pending" passe à "done") grâce au numéro de transaction récupéré par Moneybookers précédemment envoyé)

A chaque vente de titre, l'argent est sauvegardé dans les comptes de l'institution financière. En cas de crise (comme une indisponibilité de la plateforme de titre), l'argent, lui, est toujours disponible.

8. L'utilisateur peut consulter ses titres et rafraichir la page (F5). Il peut aussi utiliser l'emailling pour accélérer le processus de paiement, puis :

- Le vendeur reçoit une notification par email du FSX l'informant de la cession du titre et de la reception de l'argent. (+ emetteur + argent + devise)
- L'acheteur reçoit une notification par email de l'institution financière. (+ argent + devise + indicatif produit)

9. Déconnexion de la plateforme de trading (FSX)

Les vues :

L'architecture des vues est commune à FreePay (header et footer en commun sur toutes les pages).

Le Menu : (Accueil, Historique des titres, Acheter un titre, Ordre Achat (la confirmation est visible si l'ordre a réussi), Consulter vos titres, Contactez-nous, Documentation)

L'utilisateur peut voir l'historique des transactions.

L'utilisateur peut acheter un titre.

L'utilisateur peut consulter la liste de ses ordres d'achat et, cliquer sur le numéro de la transaction pour voir les détails de la transaction (si celle-ci a réussi).

L'utilisateur peut consulter ses titres.

Respect des standards du W3C : a été validé XHTML 1.0 transitionnel et CSS 2.0 sous Mozilla Firefox, Internet Explorer et Safari.

Les contrôleurs :

Ce sont les même que ceux de FreePay. PHP et Javascript pour le côté serveur et client respectivement.

Les transactions qui n'aboutissent pas au bout d'1 journée sont classé comme ayant échoué. (status à -2).

Test et Confirmation :

Passer la séries de tests suivantes : Nécessite minimum 2 utilisateurs. (Alice et Bob)

Préparatif : Noter la situation financière existante des utilisateurs : "Solde en Euro" et "Solde de Titre" pour chacun d'eux.

Effectuer un achat et pour les 3 cas (inférieur, égal, supérieur), vérifier :

- Le solde de l'acheteur (Alice) a t-il diminué ?
- La valeur du titre de l'acheteur (Alice) a t-elle augmenté ?
- Le solde du récepteur (Bob) a t-il augmenté ?
- Les titres du récepteur (Bob) ont-ils diminué ?
- Le recepneur (Bob) à t-il été notifié par email ?

L'application a été conçu et testé avec CAL9000 (OWASP) pour être protégé contre les attaques de type Cross Site Scripting (XSS). (Top 10 des vulnérabilités 2007 selon l'OWASP)

Comment savoir si l'API du porte monnaie électronique d'une institution financière est compatible avec le FSX de FreePay ?

Sur l'institution financière :

1. Il faut pouvoir créer un compte.

2. Il faut pouvoir approvisionner son compte avec différents moyens de paiement (chèque, carte de crédit, transfert bancaire etc) et en retirer.

3. Il faut avoir un minimum de fonds sur son compte (solde à minimum 0.01 euros). (il faut aussi prendre en compte les frais de l'institution financière)

Facultatif (mais vivement recommandé) : l'institution financière doit être agréementé par au moins une autorité de régulation.

Les transferts entre client normaux et client de test sont interdits.

Sur le FSX :

1. Il faut pouvoir passer un ordre sur le FSX et arriver sur la plateforme de paiement.

2. Il faut que les paiements fonctionne dans les 2 sens (marchand-marchand tout en étant accessible client-client, marchand-client et client-marchand), brièvement permettre le P2P.

2 bis. (optionnel : mais c'est mieux de le faire) Il faut pouvoir activer l'automatisation des processus. (en xml)

3. Il faut que le site source (marchand / FSX) puisse obtenir une trace de la transaction de la part de la plateforme de paiement. (xml envoyé et enregistrable dans la base de donnée dans des tables sql)

Fonctionnalités en plus à implémenter (dans le futur) :

Prendre en compte le changement d'adresse du serveur. (sauvegarder toutes les requêtes, puis, changer la configuration et accepter les requêtes non passé à posteriori).

Besoin de garanties : solution envisagée : bots qui achètent automatiquement.

sécurité : si test activé : permettre des tests en local

sécurité : si paiement normal activé : désactivé en local

Le remboursement n'est pas pris en compte (pas de relation client mécontent donc remboursement)

+ multi devises

Optionnel : changer la monnaie par défaut (EUR) par une autre (exemple : USD) dans le fichier 'params.php' (sans les simple quotes)

+ solde du compte en interne (problème de sécurité)

Il serait avisé de pouvoir récupérer les anciennes transactions du FSX sauvegardé par Moneybookers pour une reconstruction de la base de donnée en cas de suppression.

Penser à vérifier que le numéro id est valide par rapport à l'email et les transactions passés.

Il faudrait pouvoir vérifier le mail et l'id auprès de Moneybookers avant de commencer des transactions.

Lorsque le titre a été sélectionné, il n'apparaît plus dans la liste des titres disponibles jusqu'à ce que la transaction soit acceptée ou annulée (environ 15 minutes max).

L'application doit prendre en compte le vol d'identité.

Import de contacts

Socialisation de l'application :

- FaceBook
- OpenSocial
- addthis (Social Bookmarking Sharing Button)

Clause de non responsabilité :

L'utilisation de la présente application à valeur d'acceptation de la clause de non responsabilité suivante : L'auteur décline toute responsabilité quant aux conséquences pouvant résulter de l'utilisation de cette application.

Script fourni sans aucune garantie.

Copyright :

Mon droit d'auteur est garanti par la Licence publique générale GNU. (<http://www.gnu.org/licenses/gpl.html>)

Mes créations sont protégées au niveau Européen par la Directive 91/250/CEE du Conseil, du 14 mai 1991, concernant la protection juridique des programmes d'ordinateur. (<http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31991L0250:FR:HTML>)

Mes créations sont protégées internationalement par la Convention de Berne pour la protection des œuvres littéraires et artistiques (géré actuellement par l' Organisation mondiale de la propriété intellectuelle (OMPI), organisme spécialisé au sein de l'ONU). (source : http://www.wipo.int/treaties/fr/ip/berne/trtdocs_wo001.html)

Contact :

Bien que je sois d'accord pour dire que le type de communication le plus efficace est le face à face, je reconnais aussi que je suis parfois indisponible et donc, j'ai mis en place une documentation écrite.

En tant que développeur, je recherche toujours un moyen de produire plus et à moindre coût. Ma motivation se base sur le fait que le logiciel fonctionne et qu'il est utile.

D'après mon expérience, pour améliorer un système, il faut que les utilisateurs puissent contacter l'auteur car l'amélioration de la plateforme se fait grâce aux boucles de rétroaction positive à l'initiative des utilisateurs. S'il y a des bogues, pour que je puisse les corriger, il faut que j'en ai connaissance précise.

Ce retour servira de base à l'architecture de la prochaine version (qui contiendra l'existant + les corrections). Je laisse donc toujours plusieurs moyens de me contacter, et de me connaître, dans mon CV, que je joins à mes créations.

Enfin, je suis ouvert à toute suggestion permettant d'améliorer le logiciel.

Si vous avez un problème de nature financière relatif à votre compte moneybookers, vous pouvez vous adresser au [service clientèle de Moneybookers](#).