

PayBook : Alternative Trading System (ATS) with Paypal Express Checkout Guide Dev FR

(version chrysoberyl)

La version française est la source primaire pour toutes les traductions. Pour les non francophone ou les non anglophones, il existe des outils de traductions comme Google translate (http://translate.google.fr/translate_t#) pour lire des documents dans votre langue.

Problématique & Vision :

Créer un monde sans pauvreté pour permettre l'éducation, réduire les maladies, réduire la mortalité et aller vers des principes.

Le changement de paradigme

Le paradigme existant permet de résoudre certain problèmes, c'est pourquoi il a été accepté par le passé. Cependant, en ce moment, il ne répond pas à la demande de tout le monde. Donc, il y a eu une remise en question et une demande de changement.

J'ai étudié le marché des solutions existantes. N'étant pas satisfait (car si j'étais satisfait la problématique aurait été déjà résolue), j'ai donc décidé de créer ce logiciel. Car, à mon avis, il faut retirer progressivement les barrières qui entravent les échanges. (réf Traité instituant la Communauté européenne, 1957)

Le logiciel PayBook 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. Ce logiciel a été crée pour répondre au besoin de financement des agents économiques au niveau européen et mondial pour pouvoir financer, en autre chose, la recherche, l'éducation et l'innovation.

Principes & Qualités :

- Liberté, Citoyenneté, Responsabilité, Égalité, Solidarité,
- Ouvert, Réglementé, Prévisible, Non discriminatoire

Objectif :

Faire un projet pilote (fonctionnel, gratuit dans sa mise en place et dans son utilisation, légal, disponible (7/7j, 24/24h) et reposant sur une situation gagnant-gagnant) d'une plateforme de trading internationale à partir d'une API de paiement (Paypal Express Checkout) et de Give1Get2 pour permettre d'augmenter le solde des participants.

Réalisé :

Le logiciel a été testé dans la "PayPal Sandbox" avec des comptes premier pour les 3 cas possibles d'achat (inférieur, égal et supérieur) : et dans chacun des cas, cela a fonctionné parfaitement.

Exemple simple : lorsqu' 1 Euro est envoyé, 2 Euros sont reçus (facteur 2 : $1*2=2$). 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.

Cible :

Cible Utilisateur : Ce logiciel est été mis en oeuvre en ciblant principalement les utilisateurs du PSP utilisé parlant français ou anglais et ayant un minimum de compétences en finance (PayPal enregistre 141 millions de clients, selon PayPal). L'objectif n'étant pas de se limiter à cette catégorie mais de pouvoir toucher le plus grand nombre d' utilisateurs.

Cible Développeur : Les personnes ciblé pour le téléchargement et l'installation du logiciel sont les programmeurs (développeur web de préférence) et/ou entreprises voulant mettre en place une plateforme de trading.

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 à but non lucratif.

Les stratégies pour faire du business

Il existe 3 stratégies possibles pour 2 personnes : Perdre, pas de deal ou gagner. (soit 9 combinaisons en tout)

Cas pratique : Je ne veux pas perdre d'argent (destruction d'argent) Je ne veux pas perdre d'argent pour que quelqu'un en gagne. (un vol d'argent) Je veux changer. Je ne veux pas gagner d'argent pour que quelqu'un en perde. (un vol d'argent) Je ne veux pas gagner d'argent tout seul. (faux billet) Je veux gagner de l'argent et que quelqu'un d'autre en gagne aussi.

La seule solution qui me semble réalisable est donc : la stratégie gagnant-gagnant.

Business Model d'une plateforme de Trading en ligne avec une stratégie Gagnant-Gagnant :

5 parties : Fournisseur, Consommateur, Concurrent, Partenaire et Soi-même.

Fournisseur :

- Les prestataires de services de paiement (PSP) ont un business model qui fonctionne par les commissions (partie fixe et variable) sur les transactions de leur clients (environ 2%). Pour que leur profit augmente, ils veulent augmenter leur volume de transactions et à ce que les clients envoient de plus grosse somme.
- Leur principaux partenaires sont les marchands et les plateforme de trading qui leur permet

d'accroître leur volume de transactions et les montants via les utilisateurs qui les utilisent. Ceci se fait par l'augmentation de leur nombre de clients qui est proportionnel à l'accès de leur système d'informations : l'API (Application Programming Interface) qui permet à quiconque d'automatiser les paiements.

- Leur fournisseurs sont une ou plusieurs banques. Les messages financiers sont gérés électroniquement par la banque. Le PSP reçoit des confirmations via une API.

Consommateur :

- Les utilisateurs de l'internet veulent répondre à leur besoins. Les besoins de l'humanité sont récurrent (doivent être satisfait tout les jours). Ce besoin est soit un produit ou service, soit de l'argent. Les produits se trouvent chez les marchands en ligne (e-commerce). L'argent facilement gagné se trouve sur les plateforme de trading.

- Pour envoyer de l'argent, l'utilisateur à besoin d'un PSP. Et son PSP lui demande d'avoir un compte bancaire pour remplir son portefeuille électronique. Comparativement au PSP, l'internaute gagne plus par transaction si le profit est à sa valeur par défaut (maximum).

Concurrent, Partenaire et Soi-même :

- Le business model de la plateforme de trading est soit le même que celui du PSP (commissions par transactions), soit des frais d'entrée, soit des frais mensuels, soit des primes de parrainages (lien avec id) ou soit gratuit. (L'argent est gagné de la même manière que les utilisateurs.)

- Les fournisseurs des plateformes de trading sont les PSP avec leur API. (le logo est mis en évidence)

- Les utilisateurs consomment un service qui leur permet de déplacer de l'argent. La stratégie de déplacement de l'argent est définie soit par la plateforme de trading soit par les utilisateurs eux-même.

(dépend de la politique interne de la plateforme de trading)

- La plateforme de trading est une couche logiciel au dessus du PSP. La conception et l'évolution de ce produit informatique peut être en partie externalisé auprès d'une ou plusieurs personnes qui s'occupe d'un logiciel libre de plateforme de trading ce qui permettrait de réduire les coûts et le temps de développement.

Interopérabilité : 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. Approvisionnement : L'utilisateur peut 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 1 euros). (il faut aussi prendre en compte les frais de l'institution financière)

Facultatif (mais vivement recommandé) : Agrément financier : L'institution financière doit être

agrée par au moins une autorité de régulation.

Séparation des tests et du réel : 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. (POST ou GET)

2. B2B, B2C, C2B & C2C : Il faut que les paiements fonctionnent dans les 2 sens (marchand-marchand tout en étant accessible client-client, marchand-client et client-marchand), brièvement permettre le P2P. Le droit de rétractation dépend du statut des personnes effectuant les transactions ensemble et, cela est défini sur le site du PSP.

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

3. XML : 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 données dans des tables sql)

Pourquoi une relation avec un PSP :

La division des rôles : Le logiciel peut être vu comme un plugin qui interagit avec le logiciel principal (l'institution financière) pour lui apporter une nouvelle fonctionnalité. Le logiciel est un système ouvert qui envoie des informations à l'intérieur (historique) et à l'extérieur (ordre).

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

Dépendance : 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. 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 issuer) 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).

Le Choix du PSP :

On ne devrait pas effectuer un transfert d'argent (une transaction) avant d'avoir vérifié que l'on peut en sauvegarder la trace. Car sinon, la personne sera déçue d'avoir payé sans obtenir sa contrepartie.

Au vu de l'évolution rapide des systèmes de paiements, il est préférable que l'API gère les versions afin que les anciennes versions fonctionnent toujours.

La problématique de sécurité a aussi été un critère décisif.

L'API PayPal Express Checkout a été choisi car il respecte des contraintes.

<http://www.paypal.fr/presentation>

Paypal SandBox API : <https://developer.paypal.com/>

Il faut surclasser le compte pour acheter et vendre avec le système : Surclasser son compte > choisir Compte Premier (ou compte business).

Pour obtenir les informations d'identifications à l'API :

- Cliquer sur Préférences > Accès à l'API > Demander des informations d'authentification API > Demandez une signature API

(en anglais : My Account > Profile > API Access > Request API Credentials > API Signature > Agree and Submit)

- Copier-Coller l' Username, le Password et la Signature pour vous authentifier sur la plateforme.

== PayPal Sandbox ==

identifiant email paypal sandbox : me@mypreferredhost.com

mot de passe paypal sandbox : a

== alice (premier) ==

email PayBook : alice@mypreferredhost.com

mot de passe PayBook : ab

email Paypal : alice_1234524871_per@mypreferredhost.com

mot de passe Paypal : abc

Identifiant API : alice_1234524871_per_api1.mypreferredhost.com

Mot de passe API : PSHQHGGGRDMH2Y7F

Signature : Ajd8FF2KxieV-6BxPDy4odWHclXnAj13TA568AvD4KSTbwfGqKT84ljS

== bob (premier) ==

email PayBook : bob@mypreferredhost.com

mot de passe PayBook : a

email Paypal : bob_1228753395_per@mypreferedhost.com
mot de passe Paypal : ab

Identifiant API : bob_1228753395_per_api1.mypreferedhost.com
Mot de passe API : 5396CWRJP62HES74
Signature : AFcWxV21C7fd0v3bYYRYRCpSSRI31AXI353cqalhXTzu4SnL2g5ZYdjTa

Le diagramme des traitements des processus marchands du côté de l'institution financière a été copié dans une image jointe au logiciel. "Express Checkout flow.png" ("PayPal Sandbox User Guide" (en anglais) p41 : "Testing PayPal NVP APIs") version 2.3

Pour collecter de l'argent sur son compte paypal :
Se logger > page principale > Mon activité récente > état de la demande / action > Collecter / capturer > collecte par lots > tout afficher rechercher > tout cocher > vérifier les éléments sélectionnés > Collecter > Collecte par lot envoyée > Mon compte

L'institution financière peut laisser des contraintes sur l'intégration visuelle de ses logos relatif a la page d'accueil, au différents mode de paiements et aux options supplémentaires.

<https://www.paypal.com/fr/cgi-bin/webscr?cmd=xpt/Marketing/general/AcceptanceMarkLogos-outside>

Qu'est-ce que "PayBook" ?

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. Il n'y a pas de marchandises échangé sur le système. 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. Il n'y a pas de droit d'entrée. 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 pas 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). Finalement : 25 euros réel -> 50 euros réel. Ce qu'il fallait démontrer.

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.

Explication financière :

Bob envoie de l'argent à Alice. Bob est le débiteur. Alice est le créiteur. Si les frais variables et les frais fixes sont supérieur à ce qu'à reçu Alice, Bob et Alice sont chargé négativement donc 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.

Le processus marchand fonctionne de la manière suivante : Le vendeur échange un bien contre l'argent de l'acheteur. Le vendeur du bien ne le fait que contre une somme supérieur à ce que cela lui à coûté en premier lieu auprès de ces fournisseurs.

De la même façon, lorsque que une personne A fait une demande de paiement et reçoit l'argent d'une personne B. Cela agit comme un mécanisme de compensation qui permet à B placer à son tour une demande de paiement pour un prix à une valeur égale ou supérieur à ce que cela lui a coûté auprès de son prédécesseur.

Traditionnellement, c'est une relation à 2 personnes uniquement. Ce qui pose problème si l'une des 2 partie ne peut prêter ou rembourser. (Relation 1 à 1) Avec ce logiciel de marché alternatif, il y a de multiples offreurs et de multiples demandeurs. (Relation de plusieurs à plusieurs) Donc, cela diminue le risque (relation 1 à plusieurs dans les 2 sens) et il y a d'avantage de possibilités de succès.

Comme je viens de le démontrer, nous pouvons transformer une stratégie perdant-gagnant en une stratégie gagnant-gagnant. C'est 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.

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

Installation du logiciel

1. Acheter un nom de domaine (maplateformedetradingdexemple.com) chez un Registrar.
2. Obtenir un hébergement contenant un espace suffisant (environ : 60 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' :
 - remplacer email dans la table freepay_titre
 - remplacer les informations client dans la table freepay_client
5. Créer une base de données « paybook » (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) hôte, utilisateur, mot de passe, base de données.
8. Sur le serveur web, copier-coller la source modifié (avec les paramètres par défauts) en 7z et zip. Créer aussi un dossier /paybook/. Importer les fichiers via un FTP (ex fireFTP, une extension de Firefox) avec les paramètres de l'hébergeur ('params.php') dans le répertoire précédemment créé.
9. Lancer dans le navigateur <http://maplateformedetradingdexemple.com/paybook/>. La page d'index doit s'afficher sans message d'erreurs. 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 <https://www.google.com/webmasters/tools>)

Pré-requis :

Utilisateur :

- Le seul matériel nécessaire est un pc, un système d'exploitation, une connexion à Internet et un navigateur web.
- Multi-Plateforme : fonctionne sous Windows ou Linux avec Internet Explorer ou Firefox.
- La formation est gratuite. Elle constitue la documentation.

Développeur :

- 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 ($\geq 1.3.33$), Mysql ($\geq 4.1.9$), PHP ($\geq 4.3.10$) avec l'extension cURL installé et PhpMyAdmin($\geq 2.6.1$).

L'architecture est en trois tiers (donnée, business logic et présentation). L'architecture se base sur le projet Give1Get2. (<http://sourceforge.net/projects/give1get2/>) Pour approfondir, la documentation de Give1Get2 est à votre disposition. (http://give1get2.sourceforge.net/give1get2/nav_telechargement.php?option=documentation)

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.

Le code ISIN est de nouveau utilisé dans ce logiciel (ISO 6166). La codification mis en place est le ZZ afin de faire une distinction claire et qu'il n'y ai aucune ambiguïté avec les pays ou territoires ayant des valeurs mobilières, selon la norme ISO 3166-1.

Les status d'une transaction (états) :

- 2 : validé (Processed)
- 1 : test
- 0 : en attente (Pending)
- 1 : annulé (Cancelled)
- 2 : Échoué (Failed)
- 3 : Retour arrière (Chargeback)

Le process qui explique comment passer de "Give1Get2" vers la prochaine version "PayBook" :

Mashup Give1Get2 sans Moneybookers (pour l'architecture) + FreePay (pour les ajouts) +

Paypal Express Checkout (pour l'interconnexion) = PayBook ATS

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

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

<https://developer.paypal.com/>

<http://developer.paypal-portal.com/pdn/board?board.id=fr>

trouver un serveur de test

- supprimer les bdd de test existant pour éviter les cafouillages.

- créer base, utilisateur, droits

mettre a jour le logiciel ftp

- créer user ftp

- supprimer ftp test

créer un sous domaine.

- changer le logo interne (pour ne pas modifier les autres)

- Installer les scripts. + bdd

Supprimer la page par défaut si nécessaire

changer le nom du répertoire d'installation

configurer avec les paramètres (attention aux préfixes)

Interface :

- création de compte (email, mdp, API_username, API_password, API_signature)

 - importer script "client_ajouter.php" et son interface de FreePay

 - modifier nav haut pour le lien vers "client_ajouter.php"

 - ajouter à l'interface "API_username, API_password, API_signature"

 - modifier les traitements relatif à "API_username, API_password, API_signature"

 - modifier la bdd relatif à "API_username, API_password, API_signature"

 - ajouter un disclaimer

 - informations création de compte api vérifié par paypal (en tentant une commande)

 - changer url absolue et relative du répertoire courant

 - tout les messages d'erreurs

 - récupération des post en cas d'erreur

 - afficher erreur api paypal bien formaté

 - fonction API paypal commune à client_ajouter.php et client_modifier.php

 - optimiser la vérification du ACK

login (email, mdp)

- ajouter lien "ouvrir un compte"

- modifier répertoire par défaut

- vérification login et mdp freepay

déconnexion

- modifier répertoire par défaut

modification du compte (mdp, api)

- importer "client_modifier.php"

- créer interface dans fichier séparé

- modification traitement

info modification de compte api vérifié par paypal (en tentant une commande)
la personne change ses api credentials et ça marche toujours (selon paypal, on ne peut les modifier)

optimiser la vérification du ACK
afficher erreur api paypal bien formaté

menu

changer "mb_paiement_en_attente.php" en psp_ordre_afficher.php
changer le lien du menu

JavaScript tronc commun :

validation en JavaScript inscription (EN + FR) (plus long)
validation en JavaScript connexion (EN + FR) (le même en plus court)
message Warning javascript i18N
définir l'encodage en européen pour les fichiers UTF-8

suivre le guide d'intégration visuelle

changer le logo du PSP
suivre les guidelines
changer les balises méta (mot clé)
changer logo plateforme trading

Traitement :

créer compte paypal sandbox
créer des comptes premiers avec API
être connecté au Paypal sandbox pendant les tests

GetExpressCheckout

modifier table "freepay_achat_en_attente" avec token
modifier insertion avec token (pour le vendeur)
recupérer API_username, API_password, API_signature à partir du token (pour le vendeur)

Donner la signification de GetExpressCheckout et DoExpressCheckoutPayment.

GetExpressCheckout : obtenir des informations sur l'acheteur
DoExpressCheckoutPayment : obtenir des informations sur le paiement
être capable d'affirmer le succès ou l'échec
enregistrer la réponse ? -> Keep simple and stupid : ne pas enregistrer

traitement (status_report.php)

cas >= :

transfert d'argent ?
changement propriétaire ?
historique ?
nouveau tarif ?

cas < :

transfert d'argent ?
déduction ancien titre et garde ancien proprio
création nouveau titre et nouveau proprio
histo nouveau titre

bdd :

table "Freepay_titre" remplacer "merchand id" par info des API (username, password, signature)

modifier champs de la table

modifier requête

redirection vers Paypal

CP : lecture des répertoires pour la documentation (plus rapide que de taper à la main)

déplacer documentation officiel du PSP dans le répertoire "paypal" du projet

renommer les fichiers pour les conventions d'écritures

inclure aussi la doc de source externe citée dans la doc interne avec respect du copyright

document photo sur la fin de la pauvreté (trop lourd (8 Mo), abandonné)

supprimer :

mb_afficher.php

nav_transaction.php

services/ISIN/ISIN_verif.php

support/plan.php

support/moneybookers.php

vérifier : pas d'erreurs ne doit s'afficher quand on tape le nom de la page dans le navigateur

paiement/titre/titre_acheter_interface2.php

paiement/titre/titre_acheter_interface.php

support/client/client_modifier_interface.php

support/client/client_modifier.php

support/client/client_ajouter_interface.php

support/client/client_ajouter.php

support/client/connexion_interface.php

support/client/connexion.php

pay/status_report.php

titre/titre_consulter.php

tester :

succes >

succes =

succes <

écrire la documentation Give1Get2 (Dev, User, Slides)

- Guide utilisateur : faire les captures d'écran

EN

FR

- Guide Dev

- Slides

- changer le nom du projet pour la conformité
- mettre les fonctions paypal dans des fichiers séparés.
- Changer la devise : définir CURRENCYCODE=EUR dans SetExpressCheckout et DoExpressCheckoutPayment
- que ce passe t-il si celui qui paye sur le psp n'est pas celui authentifié sur le FSX ? A reçoit l'argent. B paye. B ne reçoit rien. C ne recoit la demande de paiement et ne paye rien.
- Spécifié que c'est le Name-Value Pair (NVP) API en cURL qui est utilisé.
- redirection sur le PSP avec la langue locale (FR ou EN)
- guide user avec screenshot en anglais
- respecter les conventions graphiques de la présentation des documents (Guide User EN et FR)

Marketing :

- PayPal training library
- s'inscrire sur guru.com
- répondre aux questions sur les forums

Web Performance Best Practices :

- Optimize images
- Serve resources from a consistent URL
- Avoid CSS expressions
- Combine external CSS
- Specify image dimensions
- Minimize redirects
- Put CSS in the document head
- Use efficient CSS selectors

L'architecture

Le code source est en français. Les commentaires du code source sont en français aussi. 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 attrait au support utilisateur est dans le dossier /support.

La documentation est dans le dossier /support/docs.

Tout ce qui concerne l'internationalisation est dans le dossier nommé "services/i18n".

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.

L'existence d'une demande de paiement peut se vérifier via l'historique.

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

La disponibilité d'une demande de paiement peut se vérifier en appliquant le filtre avec le numéro ISIN en paramètre.

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)

- freepay_achat_en_attente : liste des tentatives de transactions. (`num_transaction`, `num_titre`, `valeur_titre`, `profit_titre`, `date_attente`, `mail_acheteur`, `mail_vendeur`, `status`, `token`)
- freepay_client : information API client (`mailcli`, `mdpcli`, `API_username`, `API_password`, `API_signature`, `datecreationcompte`)
- freepay_titre : liste des titres en vente dans leur états actuels. (`numtitre`, `datetitre`, `valeurtitre`, `mailcli`)
- freepay_titre_transferer : liste des ordres effectués. (`datet`, `numtitre`, `mailcli`, `achattitre`, `tauxprofit`)

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

1. L'utilisateur a crée son compte chez le PSP, l'a approvisionné, a surclassé son compte et récupéré les informations d'API.
2. L'utilisateur crée un compte. (Lien sur la première page en haut à droite.)
3. L'utilisateur va sur la page d'achat et sélectionne un numero ISIN (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).

4. L'utilisateur est redirigé sur le PSP. Elle fait tout les étapes nécessaire au paiement. Et est retourné sur la plateforme de trading automatiquement.
5. L'argent a été transféré par le PSP et l'équivalent en demande de paiement aussi. (Le transfert de possession des titres se fait compte tenu des nouveaux chiffres envoyé par l'API afin d'éviter toute tentative de fraude.) La transaction en attente "pending" passe à "done" grâce au numéro de transaction récupéré du PSP. A chaque transaction, 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.

6. 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)

Le solde du compte de l'utilisateur est supérieur à ce qu'il était auparavant sans l'utilisation de la présente application. Ce qu'il fallait démontrer.

Chaque utilisateur peut répéter le processus autant de fois qu'il le veut sans restrictions.

7. L'utilisateur de déconnecte de la plateforme de trading (FSX) et du PSP.

Les vues :

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

Le Menu : Accueil, Historique des échanges, Paiement en ligne, Historique des transactions, Mes demandes de paiement, Contact, Documentation

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).

Sécurité

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

Test et Confirmation :

Les comptes de tests sont ouvert à l'initiative des clients. Des fonds de test sont donné gratuitement par le PSP.

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 ISIN" 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 récepteur (Bob) a t-il été notifié par email ?

Prochain développement de la plateforme :

FaceBook

Légalité :

Je pense que mon système est légal car j'ai fais des recherches en ce sens avant de le mettre à disposition de tous. Je considère qu'il va dans le sens des principes de l'union européenne. (http://europa.eu/scadplus/european_convention/objectives_fr.htm) . J'ai joint dans le dossier "support" > "docs" les documents de référence concernant le contexte législatif, juridique et réglementaire qui pourrait s'y rapporter.

Le site internet de la commission européenne est très instructif sur cette question. (http://ec.europa.eu/internal_market/top_layer/index_24_fr.htm) Rubrique : Commission européenne > Marché Intérieur > Le marché unique des services > Services financiers. Je ne suis pas complètement d'accord sur le choix de cette rubrique puisque la plateforme de trading offre un service gratuit (sans contrepartie) et ne gère pas d'argent (uniquement la confirmation que l'argent a été transféré).

Fonds d'investissements > Investissements alternatifs : il existe un Projet de Directive relative aux gérants de fonds dits « alternatifs ». Ce projet de directive peut encore être changé, et, la version finale ne va pas forcément s'appliquer à ce cas très précis.
http://ec.europa.eu/internal_market/investment/alternative_investments_fr.htm

Services de paiement > Monnaie électronique : La plateforme de trading ne crée pas de monnaie. Donc, cela ne concerne que le PSP.
http://ec.europa.eu/internal_market/payments/emoney/index_fr.htm

Services de paiement > e-Facturation : c'est PSP Moneybookers qui gère la facturation (il peut le désactiver) Uniquement une copie est gardé à des fins d'archive par la plateforme de trading (ou rien suivant le paramétrage)
http://ec.europa.eu/internal_market/payments/einvoicing/index_fr.htm

Conglomérats financiers : Selon la taille du système de trading PayBook et sa structure, il peut entrer ou ne pas entrer dans cette catégorie. C'est au choix du/des entrepreneurs selon les opportunités de fusion / acquisitions.

http://ec.europa.eu/internal_market/financialconglomerates/index_fr.htm

Commerce électronique : Cela dépend de ce qui est fait par les clients en contrepartie de l'argent envoyé. http://ec.europa.eu/internal_market/e-commerce/directive_fr.htm

Copyright :

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

Mes créations sont protégés au niveau Européen par la Directive 91/250/CEE du Conseil, du 14 mai 1991, concernant la protection juridique des programmes d'ordinateur.

(<http://eurlex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31991L0250:FR:HTML>)

Mes créations sont protégés internationalement par la Convention de Berne pour la protection des oeuvres 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)

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.

L' Organisation

L' Indefinite Lifespan Foundation est un organisation de bienfaisance à but non lucratif, non gouvernementale, dédié à réduire la mortalité humaine, à encourager l'extension de l'espérance de vie et la recherche du bonheur en utilisant principalement la médecine préventive, la santé publique et l'informatique. (En phase de création)

Contact :

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. 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 ne suis pas toujours disponible et donc, j'ai mis en place une documentation écrite. Dans le cadre d'une politique de transparence, j'ai aussi mis mon cv en pièce jointe afin de pouvoir

renseigner qui le souhaite sur mon identité et mes aptitudes professionnelles (ce qui laisse plusieurs moyens de me contacter). Sa lecture est facultative.

De plus, je suis ouvert à toute suggestion permettant d'améliorer le logiciel. S'il y a des bogues, pour que je puisse les corriger, il faut que j'en ai connaissance précise. 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. Ce retour servira de base à l'architecture de la prochaine version (qui contiendra l'existant + les corrections).

Enfin, si vous avez un problème de nature financière relatif à votre compte PayPal, vous pouvez vous adresser au service clientèle de PayPal. (https://www.paypal.com/fr/cgi-bin/webscr?cmd=_contact-phone)