

Compte-rendu TX n°4327 : Création d'un système de gestion des photocopies

Aurélien DUMAINE

Université de Technologie de Compiègne, 16 septembre 2012

Sujet Les étudiants de l'UTC doivent se procurer des photocopies de cours à des moments particuliers de l'année ce qui peut entraîner des saturations pour le service en charge de cette tâche. La TX a pour but de proposer un système informatique de gestion permettant de gérer et d'améliorer ce flux.

Responsables Sébastien CASTRIC, Camille BAROT

1 Contexte général

L'UTC voit chaque année son nombre d'étudiants augmenter. Malgré la démocratisation de l'informatique les enseignants aussi bien que les étudiants sont encore très attachés au support de cours papier. Un nombre très important de photocopies sont vendus durant les trois premières semaines de chaque semestre.

Afin d'acheter leurs photocopies les étudiants doivent faire la queue jusqu'à plusieurs heures (90 minutes d'attente ne sont pas rares) pour obtenir parfois un nombre parfois limité de photocopies. Ils doivent donc répéter plusieurs fois l'opération.

Ces longues files d'attente sont vraisemblablement dues à la durée trop importante de l'acte de paiement. Lorsqu'un étudiant souhaite payer par carte bancaire (très fréquent étant donnée l'impossibilité de payer en espèce), le vendeur est obligé de scanner un à un chaque photocopie, puis sélectionner le mode de paiement par carte bancaire. Il faut ensuite que l'étudiant dicte son nom complet (il n'y a aucune auto-complétion !) puis que l'agent recopie le montant total sur le TPE. Lors d'un paiement par chèque, le vendeur doit également saisir la banque ainsi que le numéro de compte et le numéro de chèque.

2 Environnement technique

L'UTC a mis en place il y a de nombreuses années une plate-forme de gestion de la scolarité développée en interne : Demeter. Elle s'appuie sur un serveur SGBD, un serveur SSO Oracle et un serveur J2E. Chaque étudiant a accès à un certain nombre d'informations et de procédures entièrement dématérialisées.

Depuis la mise en place de l'ENT en février 2009, les utilisateurs accèdent à cet outil au travers d'un portail ESUP connecté à un serveur SSO CAS. L'authentification est

aussi transmise au serveur SSO Oracle. Plusieurs applications se sont depuis ajoutées à cet environnement en s'appuyant sur des technologies garantissant l'homogénéité de l'ensemble du système d'information :

- la DRH a dématérialisé sa procédure de candidatures. Le lien présent dans le menu Demeter des étudiants les renvoie, après une authentification basée sur le serveur CAS à une application PHP.
- L'association Étuville a mise en vente les places pour le gala du 18 novembre 2011 en utilisant une du PHP, une authentification CAS ainsi que la charte graphique de Demeter et le service de paiement en ligne PayBox.
- J'ai moi-même déjà intégré l'authentification CAS au travers de la librairie phpCAS dans le site de l'association Écolo D'Ingénieur (EDI).

3 Architecture technique utilisée

Au regard des éléments précédants j'ai choisi d'implémenter ce système de gestion de polycopiés avec les technologies suivantes :

- réutilisation de la nouvelle charte graphique de Demeter (notamment les CSS)
- développement en PHP/HTML/Javascript
- authentification fondée sur la librairie phpCAS issue de Jasig
- structure MVC (modèle/vue/contrôleur)
- base de donnée SQL : utilisation de fonctions SQL standard permettant une intégration future à d'autres SGBD(dont Oracle).
- paiement en ligne avec PayBox

Je souhaite fournir un applicatif techniquement solide mais souple d'utilisation s'intégrant au maximum au reste du SI.

J'avais au départ envisagé d'utiliser le framework PHP Symfony2. Afin de maximiser les chances d'intégration de l'applicatif au système d'information de l'UTC, j'y ai remoncé. L'usage d'un tel framework apportait une lourdeur et des contraintes trop importantes par rapport à la taille du projet. Je n'ai pas non plus utilisé le paradigme objet du PHP.

4 Rencontre avec les parties prenantes

4.1 Avec Monsieur Franck HERTUS, vendeur à la BUTC, le 12 mars 2012

Monsieur HERTUS m'a appris que les étudiants en formation continue payaient un forfait polycopié à l'inscription. Le système doit donc générer une vente interne sur un registre particulier. La lourdeur en terme de manipulation et de gestion papier est importante. Nous essaierons de minimiser le temps passé à traiter de tels cas en cible tout en permettant toujours la génération de pièces justificatives papier afin de modifier à minima les habitudes de travail et donc de maximiser l'acceptation du système.

Monsieur HERTUS demandé également à ce qu'une liste déroulante soit présente afin de pouvoir saisir facilement les articles ne portant pas de code-barre (passage du TOEIC, livre de GE37, renouvellement des cartes étudiantes...).

La procédure de gestion des polys invendus en fin de semestre m'était également inconnue et génère une grosse perte de temps.

4.2 Avec Monsieur paul PIREDDA, imprimeur, le 14 mars 2012

Nous avons discuté des de la modélisation du système. Monsieur PIREDDA préfère saisir les nouveaux polycopiés lui-même et ne souhaite pas que les professeurs puissent déposer leur polycopiés eux-même par internet.

4.3 Parties prenantes non sollicitées

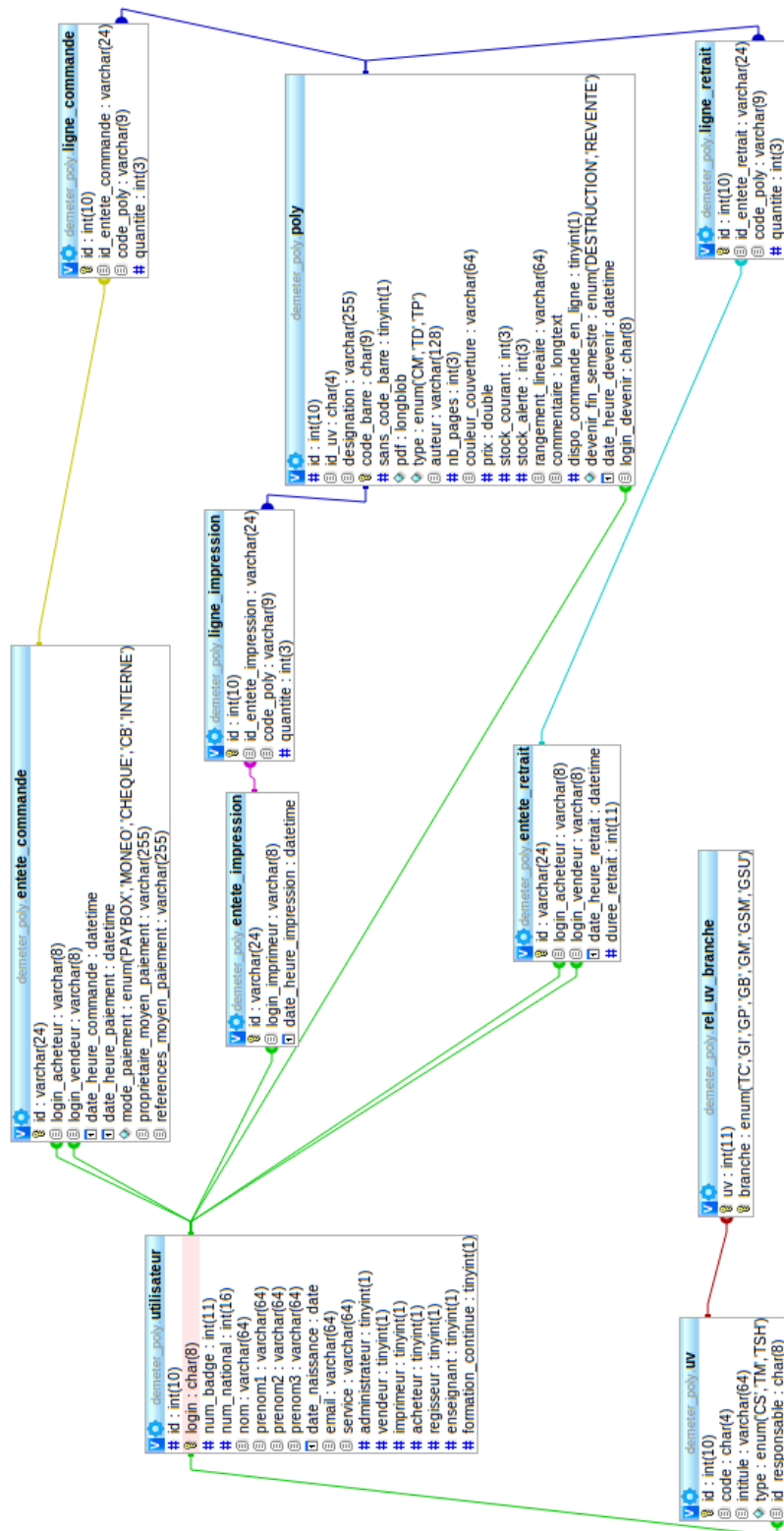
Je n'ai pas rencontré Madame Barbossa car le projet n'entrait que dans le cadre d'une TX. Je ne voulais pas lui faire perdre son temps étant donné le peu d'impact qu'aura la mise en place de ce système sur son travail . De même, je n'ai rencontré aucun membre de la DSI.

Échanger avec ces personnes sera indispensable avant une mise en place concrète du projet. Il faudra notamment déterminer les format d'inter-échange avec Arégie et/ou SIFACT.

Je pensais au départ remonter le processus métier de gestion des poly jusqu'au responsables d'UV. Ceux-ci pourraient déposer directement les PDF de leurs polycopiés sur l'application et renseigner le nombre d'exemplaires à imprimer. Étant donné l'impact d'un tel processus, je ne l'ai finalement pas retenu dans le cadre de cette TX. Il fait parti des pistes d'évolution proposées en fin de ce document. Une mise en place progressive est à préconiser.

Le profil enseignant ne sert actuellement qu'à demander au responsables d'UV si les invendus du semestre n doivent être remis en vente au semestre n+1 ou bien détruits. L'imprimeur peut générer des mails aux responsables d'UV les contenant un lien vers un formulaire dédié.

5 Modèle de données



6 Modélisation choisie

L'idée principale est de modéliser le processus de mise en vente des photocopiés :

1. L'imprimerie imprime de photocopié en fonction des indications laissées par l'enseignant. L'imprimeur valide et modifie le cas échéant le prix du photocopié calculé automatiquement selon le nombre de pages.
2. La boutique de la BU réceptionne les photocopiés.
3. Les étudiants autorisés (panel test lors du premier semestre) saisissent commandent et payent leurs photocopiés en ligne. Toute commande payée en ligne est définitive et non remboursable. L'étudiant peut faire autant de commandes qu'il le souhaite.
4. L'étudiant se présente à la boutique de la BU. Il peut consulter les stocks en temps réel afin de s'assurer que les photocopiés qu'il a payés sont disponibles.

L'étudiant pose son badge personnel sur le lecteur. Le vendeur a directement accès aux photocopiés commandés et payés en ligne qui n'ont pas encore été retirés. Selon les stocks de la boutique les quantités de polys qui vont être retirés sont pré-remplies. Le vendeur rassemble les polys disponibles en stock. Il vérifie que les quantités sont bonnes, les corrige le cas échéant et valide le retrait.

À tout moment le vendeur peut ajouter un poly absent de la liste pré-établie. Il doit pour cela ou le scanner avec une douchette code-barre ou bien saisir manuellement son code. Si aucun code n'est présent sur le produit, celui-ci apparaît dans une liste déroulante spécifique. Ce poly devra alors être payé au comptoir. Trois boutons/raccourcis-clavier sont disponibles : paiement par carte bancaire, paiement par carte Monéo, paiement par chèque. Le nom du porteur de carte est pré-rempli mais peut-être modifié.

Commander en ligne et payer au comptoir serait possible mais apporte de la confusion et une complexité inutile.

7 Sécurité

- L'utilisation de PDO évite les injections SQL
- Retour Paybox : Vérification de l'IP emmettant la requête de retour et à terme de la signature
- utilisation à minima du JavaScript afin de ne pas redoubler les contrôles côté serveur et de ne pas risquer une hétérogénéité des contrôles
- gestion des profils utilisateur
- cohérence de l'authentification avec tout le reste du SI : utilisation transparente du serveur SSO/CAS
- activation du HTTPS afin d'éviter les attaques man in the middle

8 Licence

Ce projet, comme convenu dans le sujet initial, est publié sous licence OpenSource AferoGPL. C'est une licence libre Copyleft.

Le code est disponible sur le dépôt git suivant : code.google.com/p/butc-poly

9 Optimisations mises en oeuvre

Afin de ne pas surcharger le système dès qu'un étudiant visualise le stock des photocopies, cette donnée est stockée en dur dans la table poly et non pas calculée en fonction du nombre d'exemplaires imprimés et du nombre d'exemplaires retirés. De plus, cette technique permet de gérer facilement les nouveaux et la revente d'invendus du semestre précédent. Il suffit d'initialiser les champs de la base avec le nombre d'exemplaires restants.

10 Évolution de l'ergonomie du poste de travail

On passe d'un seul poste de saisie à deux postes. Le but étant de limiter fortement le nombre de paiements au comptoir, il est conseillé de n'équiper qu'un seul des postes d'un lecteur de codes-barres et d'un TPE. En revanche les deux postes devront disposer d'un lecteur de carte RFID. Afin de limiter les contacts il est conseillé de disposer les lecteurs de carte étudiant sur le comptoir. L'étudiant posera lui-même sa carte à la demande du caissier. Étant donné le gain de temps engendré par le paiement en ligne, les vendeurs n'auront probablement plus besoin de s'asseoir. Il est donc conseillé de placer les écrans et périphériques de saisie en hauteur (écrans posés sur le comptoir) afin d'améliorer le confort visuel.

Actuellement, le montant à payer est transféré directement au TPE si l'on paye par carte bancaire (cela ne fonctionne pas pour le paiement par Monéo). Dans l'état actuel du projet, l'on perd cette automatisation. Le caissier doit donc saisir le montant à payer sur le TPE en cas de paiement par carte bancaire ou par Monéo. Il faut cependant noter qu'avec la possibilité de paiement en ligne (en cible 100 % des paiements), la fréquence de cette manipulation diminuera. Cette perte d'ergonomie est de toutes manières entièrement compensée par le report automatique de l'identité du porteur du moyen de paiement ; qui intervient lui sur tous les paiements.

11 Monté en charge et indicateur qualité

Une première période de tests est nécessaire afin de détecter les bugs restants mais aussi d'adapter l'application aux besoins utilisateurs non détectés jusqu'ici. Nous pourrions ensuite envisager d'utiliser cette application pour les paiements au comptoir une fois la période de rentrée de l'automne 2012 passée (à partir d'octobre 2012 par exemple). Nous pourrions en fonction des résultats envisager l'ouverture de la procédure de paiement en ligne pour les étudiants pour le semestre de printemps 2013. Il est possible d'expérimenter ce paiement en ligne avec un panel limité comme par exemple les étudiants de génie des procédés.

Dans une démarche qualitative il est nécessaire de chiffrer le gain de temps engendré entre un paiement au comptoir et un paiement en ligne :

Critère : Longueur de la file d'attente

Indicateur : Temps moyen passé au comptoir par étudiant

Mesure : delta entre l'heure de la pose de la carte d'étudiant et l'heure du retrait

L'autre objectif était de réduire le temps de gestion en dehors des heures d'ouverture. Ne pouvant difficilement quantifier le temps gagné nous ne présenterons pas d'indicateur chiffré. Néanmoins la génération des divers rapports PDF et des échanges informatisés avec Demeter, Arégie ou SIFACT apporteront probablement un gain de productivité.

La visibilité des stocks en temps réel permettra de faciliter la gestion en flux tendu entamée au semestre dernier. On évite ainsi une destructions trop importante d'inventus. L'imprimeur pourra ré-imprimer des polys dès qu'il verra le stock diminuer.

12 Captures d'écran au 16 septembre 2012

ENVIRONNEMENT DE DEMO [Profil acheteur](#) [Profil enseignant](#) [Profil imprimeur](#) [Profil vendeur](#) [Profil régisseur](#) [Profil admin](#)

BUTC - Vente en ligne des photocopies

UTILISATEUR : adumaine [Déconnexion](#) Si vous rencontrez un problème technique sur le site :

[Acheter des poly](#) [Historique](#)

Panier référence adumaine 16092012_120821

Code	Quantité	Prix Unitaire	Montant	Action
Votre panier est vide				

Aucun photocopié payé ne sera remboursé. Aucun photocopié retiré ne sera repris ou échangé.

Liste des poly disponibles

	CS	TM	TSH
TC			
GB			
GI		LO21C1P12 LB24C1P12 DIPLOEIC	
GM		LO21T1P12 LO22C1P12 LO22C5P12	
GP			
GSM			
GSU			

Aurélien DUMAINE (P12) - Code sous licence libre et copyleft [AfferroGPL](#) disponible gratuitement sur [ce dépôt Git](#).

Historique de adumaine

Synthèse

Code	Payés	Retirés	À retirer	Stock
DIPLTOEIC	3		3	2
LB24CIP12	3	2	1	0
LO21CIP12	6	4	2	23

Commandes/paiements

Commande n°adumaine_09072012_160623 (2012-07-09 16:06:43 par adumaine)

Code	Prix unitaire	Quantité	Total
LO21CIP12	3.85	1	3.85

Paiement : 3.85 par PAYBOX()

Commande n°adumaine_21062012_143322 (2012-06-21 14:33:39 par adumaine)

Code	Prix unitaire	Quantité	Total
LO21CIP12	3.85	3	11.55
LB24CIP12	3.15	1	3.15

Paiement : 14.7 par PAYBOX(2012-07-04 00:00:00)

Commande n°adumaine_21062012_111640 (2012-06-21 11:17:03 par adumaine)

Code	Prix unitaire	Quantité	Total
LO21CIP12	3.85	1	3.85

Paiement : 3.85 par PAYBOX(2012-07-11 00:00:00)

Commande n°adumaine_16062012_212838 (2012-06-16 21:28:48 par adumaine)

Retraits

Retrait n°adumaine_10082012_162745 (2012-08-10 16:27:45 par adumaine)

Code	Quantité
LO21CIP12	4
LB24CIP12	2

Liste des utilisateurs en formation continue (éligibles à la vente interne)

Utilisateur	Action
DUMAINE Aurélien	Facture

BUTC - Vente en ligne des photocopies

UTILISATEUR : adumaine [Déconnexion](#)Si vous rencontrez un problème technique sur le site : [?](#)[Liste des impressions](#)[Liste des polys](#)

Liste des bons d'impression

[Ajout impression](#)

Impression n°adumaine_10082012_162937 (2012-08-10 16:29:37 par adumaine)

Code	Quantité
DIPLOEIC	2

[Imprimer bon d'impression](#)[Supprimer bon d'impression](#)Aurélien DUMAINE (P12) - Code sous licence libre et copyleft [AfferoGPL](#) disponible gratuitement sur [ce dépôt Git](#).

BUTC - Vente en ligne des photocopies

UTILISATEUR : adumaine [Déconnexion](#)Si vous rencontrez un problème technique sur le site : [?](#)[Liste des impressions](#)[Liste des polys](#)

Liste des polys

[Ajout poly](#)

Code poly	UV	Type	Prix	Nb ex commandés	Nb ex retirés	Nb ex à retirer	Stock courant	Action
LO21C1P12	LO21	TM	3.85	13	4	9	23	Voir la fiche
LB24C1P12	LB24	TSH	3.15	6	2	4	0	Voir la fiche
DIPLOEIC	0		25	13			2	Voir la fiche
LO21T1P12	LO21	TM	2.2				0	Voir la fiche
LO22C1P12	LO22		5				0	Voir la fiche
LO22C5P12	LO22		6				0	Voir la fiche

[Envoyer les mails de gestion des inventus en fin de semestre aux responsables d'UV.](#)
[Envoyer les mails de rappels des polys payés mais non retirés aux étudiants.](#)Aurélien DUMAINE (P12) - Code sous licence libre et copyleft [AfferoGPL](#) disponible gratuitement sur [ce dépôt Git](#).

BUTC - Vente en ligne des photocopies

UTILISATEUR : adumaine [Déconnexion](#)

Si vous rencontrez un problème technique sur le site : [?](#)

[Vendre des polys](#) [Retirer des polys](#) [Brouillard de caisse](#) [Formation continue](#)

Saisie étudiant

NUMERO BUTC ETUDIANT DEMO : 81640

Aurélien DUMAINE (P12) - Code sous licence libre et copyleft [AfferoGPL](#) disponible gratuitement sur [ce dépôt Git](#).

BUTC - Vente en ligne des photocopies

UTILISATEUR : adumaine [Déconnexion](#)

Si vous rencontrez un problème technique sur le site : [?](#)

[Vendre des polys](#) [Retirer des polys](#) [Brouillard de caisse](#) [Formation continue](#)

Étudiant en cours



login : adumaine
NOM : DUMAINE
Prénom : Aurélien
Éligible vente interne : OUI
[Historique](#)

[Changer d'étudiant](#)

Bon de retrait


Code	Payés par adumaine	Retirés par adumaine	Retirables par adumaine	Stock	Retrait
DIPLTOEIC	3		3	2	2
LB24C1P12	3	2	1	0	0
LO21C1P12	6	4	2	23	2

[RETIRER \(F12\)](#)

Aurélien DUMAINE (P12) - Code sous licence libre et copyleft [AfferoGPL](#) disponible gratuitement sur [ce dépôt Git](#).

[Vendre des polys](#) [Retirer des polys](#) [Brouillard de caisse](#) [Formation continue](#)

Etudiant en cours



login : adumaine
NOM : DUMAINE
Prénom : Aurélien
Éligible vente interne : OUI
[Historique](#)

[Changer d'étudiant](#)

panier

Références du moyen de paiement :
Vide si Monéo, N°Cheque+N°Compte+Banque si chèque, N° de carte si CB
Propriétaire du moyen de paiement :

Code	Quantité	Prix Unitaire	Montant	Action
Votre panier est vide				

Saisie des polys

--Articles sans code barre--
Saisir le code barre :
Appuyer sur la touche TABulation du clavier pour passer directement à la zone de saisie des références du moyen de paiement.
CODE BARRE DU POLY DE DEMO : LB24C1P12

13 Installation

Modules à installer sur une distribution GNU/Linux : php5, php5-mysql, mysql-server, apache2, php5-curl, phpmyadmin

Afin d'obtenir la version la plus à jour du code, merci d'exécuter un **git clone** <https://code.google.com/p/butc-poly> dans le répertoire souhaité.

Étapes de configuration :

- attribution des droits
- configurer le https
- modifier le chemin du répertoire cgi-bin
- paramétrage de la base de données et IP_SREVEUR_BUTC dans global/config.php
- création de la structure de la base de donnée
- initialisation de la base de donnée en important les utilisateurs et les UV de Demeter
- positionner la variable ENVIRONNEMENT_DEMO à 0 dans global/config.php

Un navigateur web récent devra être installé sur les postes clients des vendeurs de la BUTC. L'utilisation du champ "date" de la norme HTML5 est utilisé pour sélectionner de manière légère et agréable la date du brouillard de caisse à imprimer. Chrome 21 ou supérieur gère correctement ce type de champ.

14 Coût et retour sur investissement

Coût de développement initial : 0 euros.

Coût matériel : 2 lecteurs RFID (2*50€ = 100 euros). Les deux PC ainsi que le TPE et le lecteur code-barre sont déjà présents.

15 Possibles évolutions futures

Se trouvent ci-après un certains nombre de propositions pour les développements futurs. Il est à noter que ces tâches sont totalement en-dehors du cahier des charge de cette TX. Ce dernier ayant été selon moi largement rempli.

15.1 Sécurité

- Vérification de la signature paybox en plus de l'IP d'origine de la requête
- Module de comparaison a posteriori des relevés de compte csv paybox et des paiements paybox en base
- Vérification a posteriori des signatures paybox + paiements dus et paiements effectifs paybox
- Mail automatique à l'acheteur et à l'admin lorsqu'il y a une erreur de paiement
- Utilisation des fonctions avancées de PDO(préparation des requêtes)
- Politique de backup distante régulière (load balancing inutile pour cette application non critique)

15.2 Économique

- Passer par un tiers de paiement en ligne moins onéreux ou bien limiter la possibilité de paiement en ligne au premier mois du semestre.

15.3 Ergonomique

- Afficher les UV par liste alphabétique afin de limiter les requêtes nécessaire à l'affichage de la boutique
- Transformer la liste déroulante des articles sans code-barre en boutons directement cliquables.
- Auto-complétion du titulaire du moyen de paiement si différent du titulaire de la carte d'étudiant (utilisation des datalist ou bien des websockets)
- Pouvoir saisir les données spécifiques aux chèques (banque + n° de compte) en fin de journée
- Pouvoir éditer un brouillard de caisse sur un intervalle de jours
- Calcul automatique du prix du poly en fonction du coût à la page, du nombre de page et du nombre d'emplacements gratuits pour les enseignants
- Aller directement sur retirer_poly ou vendre_poly selon qu'il y aie ou non des polys à retirer
- Interface Applet Java avec le TPE et l'imprimante chèque Ingenico
- Passer automatiquement en plein écran via JavaScript lorsque l'on est dans "boutique comptoir"
- Écrire une aide en ligne avec Scenarii
- Intégrer un lien contenant un hash directement dans les mails envoyés aux enseignants afin qu'ils n'aient pas à s'authentifier pour ordonner la remise en vente ou la destruction des invendus
- Interface de gestion des droits (attribuer les profils admin, enseignant, vendeur, imprimeur, acheteur, régisseur)

15.4 Base de donnée

- Script de bascule d'un semestre n au semestre n+1 (archivage des tables de vente, report des stocks de poly devant etre remis en vente...)
- Intégration à Demeter (webservice, vues spécifiques ?) pour l'import des utilisateurs, des inscriptions aux UV, des effectifs...
- Batch d'effacement des commande non payées
- Ne pas utiliser de RO non standard car cela engendre une perte de portabilité
- Ajouter un champ erreur_paiement dans la table entete_commande ?

15.5 Organisationnel

- Test de l'acceptation d'une interface de dépôt des pdf des polycopiés par les professeurs
- Création d'un alarme afin que l'imprimeur réimprime des polycopié dès que le stock est inférieur à 15 % des effectifs de l'UV.
- Génération de la page de garde avec code barre + concaténation avec le pdf du poly à sa création par le responsable de l'UV
- Les UV auxquelles sont inscrits les étudiants apparaissent en haut du tableau d'achat