

CPOO2 / PTUT - Projet e-commerce

Objectifs du projet :

Le but de ce projet est de développer un logiciel de vente de produits en ligne et de gestion de stock. Ce projet sert à la fois de PTUT (les parties Bonus vous permettent d'aller plus ou moins loin dans la réalisation) et de sujet de CPOO2 (44h, du 23/03 au 10/06). En Java Avancé (28h, du 4/04 au 8/06)), nous ferons aussi des TP portant sur ce sujet. Vous aurez 3 jours plein dédiés à ce projet (23-25 mai 2022).

Livrables attendus :

- Un dossier d'analyse et de conception en langage UML
- Un prototype fonctionnel pour chaque lot
- Un RETour d'EXpérience (RETEX), précisant les points forts et points faibles de votre choix d'organisation, les écarts entre la conception initiale et le prototype présenté.

Analyse fonctionnelle :

En binôme, vous devez proposer 2 interfaces distinctes:

- Un portail web pour un site e-commerce
- Une application bureautique pour la gestion de stock

1. Portail Web

Le portail web est le point d'entrée public sur la plateforme, celui que les clients vont utiliser au quotidien. Il doit donc être le plus ergonomique possible. Dans ce module, vous devez concevoir un site de e-commerce pour vendre des produits. Votre boutique en ligne doit permettre aux clients de :

- Parcourir le catalogue des produits classés en catégories et sous-catégories
- Consulter les caractéristiques d'un produit
- Créer un compte
- Donner son avis sur un produit
- Ajouter un produit au panier
- Payer une commande

Pour ce faire, il faut prévoir:

- Un module *front* accessible à tous: page de bienvenue, création de comptes, affichage des fonctionnalités du site e-commerce avec les promos et les dernières nouveautés, etc. ;
- Un module de consultation du catalogue des produits: moteur de recherche, page d'affichage du produit ;
- Une module de paiement avec la possibilité d'ajouter un bon de réduction (**Bonus**)
- Une module de dépôt d'un avis sur un produit. Chaque utilisateur peut déposer un avis avec une note comprise entre 1 et 5 étoiles et un champ de commentaires (**Bonus**)

2. Application de gestion de stock

Pour gérer le site e-commerce, on dispose d'un outil de gestion de stock qui sera une application Java, dédié au gestionnaire de stock. Il assure les fonctionnalités suivantes :

- Gestion du catalogue des produits : créer, modifier et supprimer un produit, affecter un produit à une catégorie ou une sous-catégorie ;
- Gestion des catégories et des sous-catégories : créer, modifier et supprimer des catégories et des sous-catégories ;
- Gestion d'opérations promotionnelles avec une date de début et de fin pour chaque promotion ;
- Gestion des alertes : définir un seuil pour chaque produit dans le stock. Un message d'alerte doit être affiché si le seuil d'un produit dans le stock est dépassé ;
- Gestion des comptes Client ;
- Modération des avis (**bonus**).

Spécifications particulières

- Il faut faire en sorte que le client puisse afficher les produits **en promotion uniquement**.
- Les utilisateurs non identifiés peuvent consulter le catalogue de produits. Seuls les utilisateurs identifiés peuvent passer une commande et ajouter un avis sur un produit.
- Les produits peuvent être triés et/ou filtrés par prix, couleurs, catégorie, etc.
- Chaque catégorie peut avoir une ou plusieurs sous-catégories.
- Chaque produit possède :
 - Un nom/ identifiant
 - Une catégorie et une sous-catégorie, à laquelle il appartient

- Un prix
- Une description
- Un avis (**bonus**)
- Un état s'il est en solde ou pas
- Un prix après solde s'il existe
- Un seuil d'alerte pour le stock
- Une quantité actuelle en stock

Selon le produit vendu dans votre magasin, d'autres informations peuvent être ajoutées (e.g., photo, modèle, année de fabrication, tailles disponibles, etc.)

- Un compte client possède:
 - Un nom
 - Un prénom
 - Une adresse email
 - Un mot de passe

Évaluation

- Les soutenances se feront en **CPOO2** et donneront lieu à une double évaluation :
 - (1) L'apport des IHM et de la conception sur l'implémentation, et surtout du retour d'expérience des binômes : qu'auront-ils retenu de cette 1^{ère} expérience de développement. Cette évaluation aura lieu en fin de module avec des soutenances, effectuées sur les dernières 6h.
 - (2) On évaluera d'autre part la qualité de réalisation de la partie Java et de la partie web, puis on moyennera. Ainsi chaque membre sera impliqué dans la réussite de son collègue.

La note **CPOO2** sera la moyenne de ces 2 évaluations (RETEX + moyenne des réalisations). On pourra individualiser l'évaluation, si nécessaire.

- L'évaluation de la conduite de projet (livrables) et la qualité de l'implémentation donnera une note de **PTUT**.
- Les TP de **progIHM** seront évalués par rapport aux concepts vus en IHM Java et qui auront été implémentés dans ces travaux.

Lots à réaliser

On propose la livraison de 2 lots, l'un pour le 3 mai, l'autre pour le 24 mai.

Lot n°1

Contenu : Ce lot concerne le développement du module « gestion du catalogue des produits » de l'application d'administration, et des modules « création de compte » et « consultation du catalogue » de l'interface Web.

Date limite de livraison : mardi 3 mai 2022 à 16h00

Livrable n°1

Le premier livrable est constitué d'un dossier de Gestion de Projet :

- la méthode de gestion de projet retenue (classique ou agile)
- un planning prévisionnel
- le découpage en tâches du projet et leur répartition au sein du groupe.

Livrable n°2

Le second livrable est le dossier de conception du lot (UML) :

- un diagramme de packages (par ex., un par partie : Java, Web)
- par package, le Diagramme de Classes du package (DCL) avec attributs et opérations d'analyse
- par package, un Diagramme de Cas d'Utilisation (DCU) avec les acteurs du système.

Livrable n°3

Le troisième livrable concerne la conception des IHM : développez les maquettes d'écran et leur enchaînement illustrant le fonctionnement choisi, à l'aide de l'outil de votre choix (WireFrame ou autres).

Livrable n°4

Le quatrième livrable correspond au **dossier technique** commun à l'application d'administration et à l'interface web. Il décrit :

- Le schéma de Base de Données, avec le script SQL de création de la base et l'insertion des données d'un jeu d'essai ;
- Un document détaillant comment seront gérés les droits d'accès à l'application.

Livrable n°5

Le cinquième livrable correspond au prototype de l'application d'administration, avec la gestion du catalogue (il y aura une 2ème version dans le lot suivant).

Livrable n°6

Le sixième livrable correspond au prototype de l'interface web, avec les modules de création de compte et de consultation du catalogue. Une page d'accueil basique avec un lien « se connecter/ se créer un compte » est suffisant.

Lot n°2

Contenu : Ce lot se concentre sur le développement du module « gestion d'opérations promotionnelles » de l'application d'administration et le module « paiement » de l'interface Web.

Date limite de livraison : Mardi 24 mai 2022 à 16h00

Livrable n°1

Ce livrable est l'évolution du dossier de conception UML.

Vous justifierez des changements apportés depuis la livraison du lot n°1. Vous ajouterez des Diagramme d'Activité (DIT) ou de séquences (DSQ) illustrant le fonctionnement des nouveaux cas d'utilisation, notamment ceux qui sont complexes ; éventuellement vous fournirez des scénarios au format texte associés aux différentes situations.

Ce livrable contient également les évolutions à apporter au schéma de base de données, et le script SQL permettant de réaliser ces modifications.

Livrable n°2

Ce livrable est l'évolution du dossier de conception d'IHM.

Vous ajouterez les maquettes des écrans d'administration de gestion d'opérations promotionnelles et du module « paiement » de l'interface web. Vous justifierez les changements éventuels apportés aux maquettes déjà livrées.

Livrable n°3

Ce livrable est la version 2 de l'application d'administration, contenant les fonctionnalités supplémentaires attendues.

Livrable n°4

Ce livrable est la version 2 de l'interface web, avec l'ajout des fonctionnalités attendues.

Livrable n°5

Ce dernier livrable est un document relatant votre RETEX relatif au développement de ce projet : un bilan et une réflexion *a posteriori* du travail d'analyse effectué, avec mention des *points forts* : qu'est-ce qui a permis d'aller plus vite, de faciliter la construction des classes, etc. et des *points faibles* : erreurs commises, incomplétude, éléments non utilisés. **L'implémentation pourra être différente de ce que vous aviez envisagé en analyse, mais les écarts devront être argumentés.**

Annexes

Données et jeux d'essai

Il existe un site appelé Generate Data, <http://www.generatedata.com/#generator> qui permet de créer un script SQL pour remplir automatiquement vos tables de données fictives (ID, noms, prénoms, etc.). Attention, ne passez pas du temps à remplir la BD et rédiger les rapports, c'est le reste qui est important pour ce module.

Guide méthodologique

- **L'utilisation d'un AGL** est conseillée pour la génération automatique de code. Pour la programmation Java et le développement web, vous prendrez un IDE.
- Le serveur web de l'IUT iutdoua-web.univ-lyon1.fr sera utilisé pour héberger le site web public avec **MariaDB** et **PHP7**.
- Pour les données persistantes, vous choisirez une implémentation par BD avec JDBC. Dans SQL*Plus, utiliser le *start* sur le script généré avec PowerAMC.
- L'application demandée comporte différents modules, mais tous relèvent de la même application: ils partagent des données communes (trouver lesquelles). Ne faites pas de développements indépendants. Notamment sous votre AGL de conception, les 2 interfaces (application d'administration et interface web) doivent faire partie du même projet.
- Le module CPOO2 a une durée de 44h. En mode traditionnel, **l'analyse requiert environ 16h**, la programmation 24h. En mode Agile, vous définissez la durée de sprint si vous faites du Scrum, et les éléments ciblés pour chaque itération. Si vous faites du Kanban, évaluez au mieux les dates de réalisation, et adaptez-les en fonction de l'avancée de chaque livrable. Dans tous les cas, vous pouvez utiliser le Kanboard mis à votre disposition dans le cadre du module AGL Test à l'adresse <https://aspe2021.bavori.tech/> pour organiser votre travail. Les dernières 6h sont consacrées à l'évaluation du travail réalisé par chaque binôme sous la forme de démonstrations, devant le groupe entier.
- Le travail concerne trois parties bien distinctes : l'analyse/conception, l'IHM et l'implémentation de l'application, qu'on retrouve dans les livrables de chaque lot.
- Comme le temps risque d'être court, il est important de **prioriser ce qui se valorise le mieux en présentation**, i.e. ce qui concerne le « métier » (gestion du catalogue des produits, gestion des comptes utilisateurs) plutôt que les parties « administratives » (publicité, promotion, etc).

- Pour faciliter le développement à deux, vous utiliserez un **système de gestion des versions** type CVS (*Concurrent Versions System*), par ex. GIT avec le GitLab de l'Université (la forge).
- Pour le développement WEB, seule l'utilisation de **Frameworks connus** est autorisée. Vous pouvez utiliser des librairies JS supplémentaires SI ET SEULEMENT SI elles vous font gagner du temps de développement. **Ne perdez pas de temps à apprendre Angular, Vue ou React si vous ne les maîtrisez pas.**
- **Gestion de projet** : le développement de cette application se fera selon la méthode de GP de votre choix : traditionnelle ou agile, dont vous devrez *rendre compte* pour la présentation finale. On veillera à constituer un **planning prévisionnel des tâches** pour la programmation, avec durée et affectation aux membres du groupe. Outre les tâches de codage des différentes parties, vous penserez aux tâches annexes comme la formation à un nouvel outil (ex. CVS), la création des tables, les tests, la préparation de la démonstration, etc.
Vous noterez ensuite le temps passé (à la louche) à réaliser chacune des activités pour pouvoir, en fin d'implémentation, **comparer avec un planning réel.**
- Les encadrants sont là en tant que « **client** » et pour vous aider avec UML, Java et le Web, pas pour diriger votre travail. Il y a plusieurs façons de voir et de procéder pour l'application proposée, chaque binôme construira son système selon son point de vue qu'il présentera lors de la démonstration finale.