

Examen 2eme session

12 Juin 2015 (2 heures avec documents : tous SAUF ANNALES CORRIGÉES).

Barème *indicatif* sur 20 points

1. Questions de cours

[5 Pts]

Répondez de façon précise et concise aux questions.

Barème : VALABLE sur toutes les questions de cours : -25 à -50% si la réponse inclut la bonne idée, mais qu'elle est noyée dans des infos ou autres réponses fausses/inappropriées.

QC1. Expliquez la différence entre une dépendance structurelle et une dépendance fonctionnelle. Laquelle préférer en général ?

QC2. Peut-on dire qu'une bonne conception est une conception qui définit une solution répondant aux besoins spécifiés en analyse (ni plus, ni moins) ?

QC3. Quelles sont les différences entre les classes d'analyse et les classes définies en conception ?

QC4. Décrivez succinctement les étapes de la méthodologie définie par le standard UML de l'OMG.

QC5. Expliquez les similarités et les différences entre un modèle et un méta-modèle.

Une seule note sur 100 à saisir.

QC1. 10% décrire dépendance structurelle, 10% fonctionnelle, 5% on préfère fonctionnelle

QC2. 20% Non, ce n'est pas le seul critère. Une bonne conception doit aussi être flexible, robuste, efficace...

QC3. 10% les classes de conception servent de base aux classes de réal.

10% Les classes d'analyse capturent la structure du problème.

QC4. 15% non, UML n'est pas une méthode

QC5. 10% : Un Modèle est instance d'un MM. 10% Un MM est un modèle.

2. Analyse : EB6 vente aux enchères [10 Pts]

Une société désire mettre en place un système de ventes aux enchères, nommé « eB6 » permettant à des millions d'acheteurs et de vendeurs d'acheter et de vendre n'importe quel objet à travers le monde. Vous avez la charge de réaliser la phase d'analyse de ce système.

N'importe quel utilisateur peut parcourir le site web d'eB6 sans être inscrit. En effet, les utilisateurs ont uniquement besoin de s'inscrire pour acheter et pour vendre des objets. Avant de commencer à vendre sur eB6, les utilisateurs doivent ouvrir un compte vendeur pour qu'eB6 vérifie leur sérieux. eB6 propose deux façons d'ouvrir un compte vendeur :

- une façon instantanée sur internet : il suffit de fournir les informations de la carte de crédit du vendeur (la carte ne sera pas débitée, mais uniquement utilisée à des fins de vérification de l'identité du vendeur sauf si vous demandez à eB6 de prélever également les frais de vente sur cette carte) ;
- une méthode par courrier postal qui prend 2 à 3 jours et permet de recevoir un code de confirmation par courrier à la maison.

Quelle que soit la façon d'ouvrir un compte vendeur, il faut fournir les coordonnées personnelles du vendeur (nom, prénom, adresse postale, téléphone, email) ainsi que les coordonnées bancaires (soit les informations de la carte bancaire soit un RIB). Pour vendre un objet, le vendeur doit remplir le formulaire de mise en vente qui donnera les informations utiles à l'acheteur pour savoir s'il souhaite acheter l'objet. Ce formulaire contient:

- un titre clair et accrocheur ;
- une description complète indiquant aussi bien les qualités que les défauts pour éviter les questions et les litiges ;
- une photo fidèle de l'objet ;
- un prix attractif ;
- dans le cas d'une vente par enchère, la durée de la vente (3, 5, 7 ou 10 jours) en pensant à inclure un week-end pour toucher les acheteurs du week-end.
- Les catégories correspondantes à l'objet.
- Les mots-clés correspondants à l'objet

Il existe deux façons de mettre en vente un objet :

- une vente directe qui permet à n'importe quel acheteur d'acheter directement l'objet
- une vente par enchère qui permet aux acheteurs de placer des enchères jusqu'à la fin de la durée de la vente. A la fin de la durée de la vente, l'acheteur ayant placé la dernière enchère pourra alors acheter l'objet.

Le site web eB6 propose deux moyens aux utilisateurs pour trouver un objet :

- Naviguer par catégories : Depuis la page d'accueil du site, il est possible de naviguer dans les catégories et les sous catégorie des objets afin de trouver des objets.
- Naviguer par mot-clé : Depuis la page d'accueil du site, il est possible de taper un mot-clé dans le moteur de recherche. Les objets correspondants au mot-clé seront alors présentés.

Une fois que les utilisateurs ont trouvé l'objet de leurs rêves, ils peuvent, s'ils sont des acheteurs, soit enchérir dessus, soit l'acheter immédiatement en fonction du format de vente choisi par le vendeur de l'objet. Pour s'inscrire, un acheteur doit obligatoirement donner ses informations personnelles et choisir un login et un mot de passe. Pour enchérir, l'acheteur doit saisir le montant maximum qu'il est prêt à payer pour cet objet. eB6 va enchérir pour l'utilisateur jusqu'à ce montant maximum en cas de surenchère d'un autre acheteur. A la fin d'une vente (directe ou par enchère), le vendeur peut accepter ou refuser la vente. Si celle-ci est acceptée, elle pourra être considérée comme un contrat qui lie l'acheteur et le vendeur. Pour acheter un objet immédiatement, si le vendeur propose ce format, il suffit d'accepter le prix du vendeur et l'objet est à acheter immédiatement.

Pour payer un objet que l'acheteur a acheté sur eB6, il faut entrer en contact avec le vendeur, lui demander le montant total (frais de port inclus) s'il n'est pas déjà précisé dans l'annonce et lui envoyer directement l'argent via un des moyens de paiement qu'il accepte. Pour contacter le vendeur, il faut utiliser l'e-mail du vendeur présenté en fin d'enchères. Le vendeur n'enverra l'objet qu'une fois le paiement reçu.

Une fois l'objet reçu l'acheteur peut laisser une évaluation du vendeur (constitué d'une note sur 5 et d'un commentaire) et réciproquement le vendeur peut évaluer son client.

Question 2.1 : (2,5 pts) Réalisez le diagramme de cas d'utilisation du système. Commentez et/ou annotez un minimum le diagramme.

Barème TODO

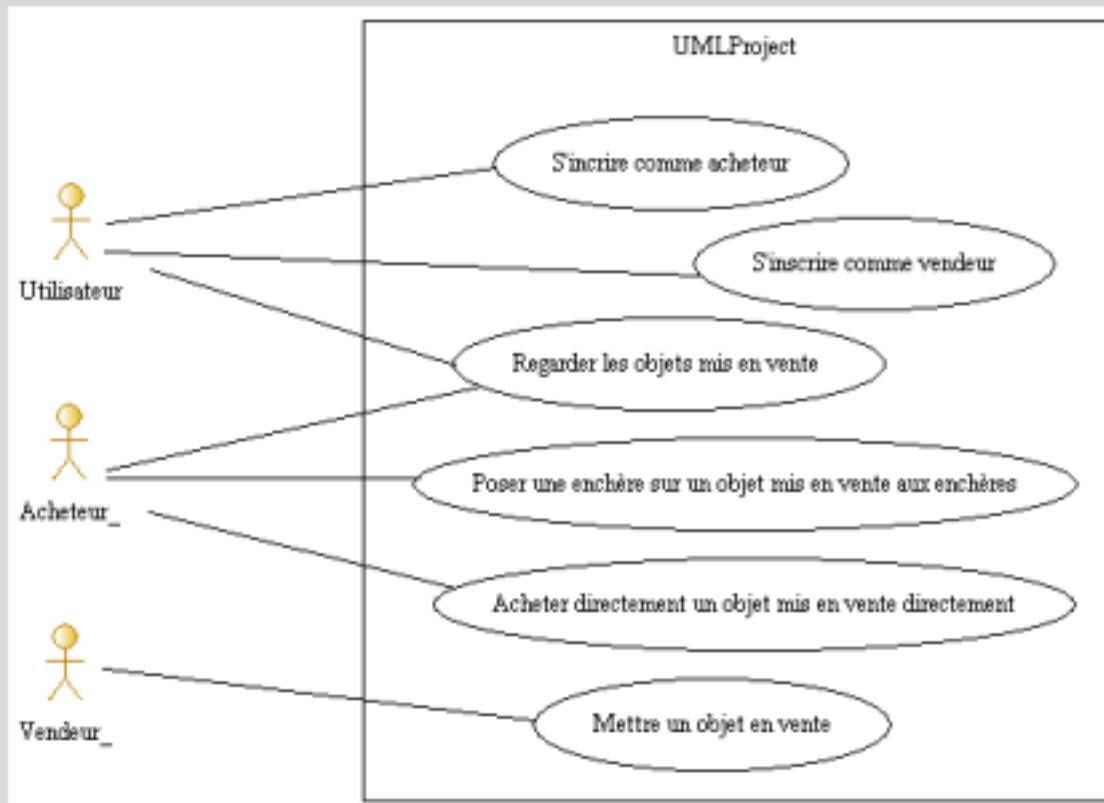


Diagramme de cas d'utilisation :

Les trois acteurs sont l'acheteur, le vendeur et l'utilisateur. Il n'est pas nécessaire de positionner les liens d'héritage entre acheteur, vendeur et utilisateur mais cela n'est pas non plus une erreur.

Les cas d'utilisation absolument fondamentaux : « s'inscrire comme vendeur », « s'inscrire comme acheteur », « parcourir les objets mis à la vente », « déposer un objet à la vente », « acheter un objet », « laisser une évaluation sur le vendeur », « laisser une évaluation sur l'acheteur ».

Il faut soit découper le cas « déposer un objet à la vente » en deux cas « vente directe » et « vente par enchère » soit préciser les différentes variantes dans le texte expliquant le diagramme.

Il faut soit découper le cas « acheter un objet » en deux cas « achat direct » et « achat par enchère » soit préciser les différentes variantes dans le texte.

La communication entre le vendeur et l'acheteur n'est pas un cas d'utilisation.

Il n'y a pas d'administrateur.

L'authentification ne doit pas apparaître.

Sur 100 :

10% par acteur => 30%

10% par UC « absolument fonadamental) => *7 = 70%

Acteurs :

-10 à -20% par faute de syntaxe...

-10% par « use case » modélisé qui n'en est pas un.

Question 2.2 : (2,5 pts) Réaliser la fiche détaillée du ou des cas d'utilisation(s) couvrant les étapes qui permettent à un vendeur de mettre en vente (directe) un article.

TODO

Titre : Mettre en vente directe

Acteur : Vendeur

Hypothèse : aucune.

Pré : Le vendeur a créé un compte et est authentifié

Post : Un nouvel objet est mis en vente, consultable par tous.

Scénario :

1. Le vendeur sélectionne l'action « mettre en vente directe »

2. Le système affiche le formulaire de description : Ce formulaire contient:

- un titre clair et accrocheur ;
- une description complète indiquant aussi bien les qualités que les défauts pour éviter les questions et les litiges ;
- une photo fidèle de l'objet ;
- un prix attractif ;
- Les catégories correspondantes à l'objet.
- Les mots-clés correspondants à l'objet

3. Le vendeur peut remplir les champs dans l'ordre qu'il souhaite.

4. Le vendeur valide

5. Le système enregistre la description de l'objet et la publie sur le site.

Alternative A1 : modifier saisie

En SN3 le vendeur peut choisir de modifier une saisie

Exception E1 : Annulation utilisateur

En SN3 ou SN4 l'utilisateur peut annuler, retour à l'interface principale.

Barème : sur 100%

Cette question est très délicate à corriger. Il faut donc vérifier les points suivants.

+15% une pré condition sur l'existence de sections ou au moins du livre identifiée

+15% une post condition au moins est spécifiée exprimant que l'enchaînement est créé

+15 les divers affichages du système (e.g. le système affiche un formulaire de soumission)

+20 On doit voir : toutes les données du CdC

+10% : une ALT au moins

+10% : une exception « annuler »

+15% on a spécifié comment on attache une photo, ou comment les mots clés sont proposés dans une liste... Des éléments d'IHM/interaction sont détaillés.

-10 % par incohérence du texte, utilisation incorrecte des champs Pré/Post/Scenario etc... En particulier, -10% si les préconditions/hypothèses sont testées dans le scenario, ou si une ALT viole les post-conditions.

-10% le vendeur n'a pas l'initiative (déclencheur)

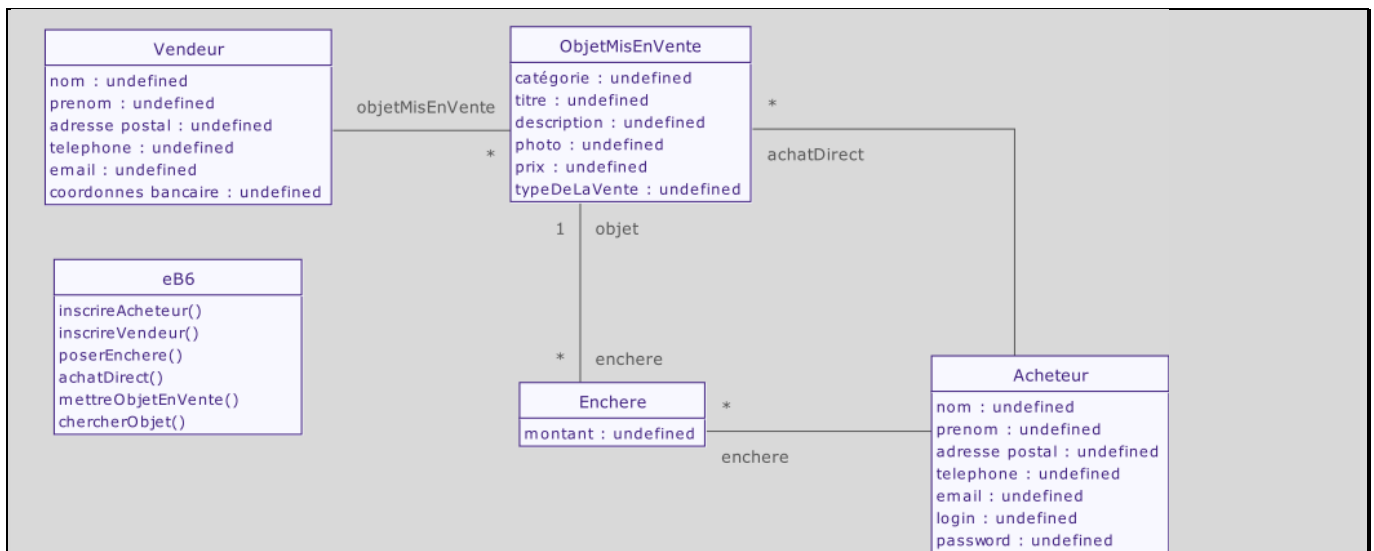
-10% on ne sait pas clairement qui du système ou de l'acteur fait l'action dans une étape du scenario

-15% : Spécification d'étapes hors système comme étapes du scenario (e.g. l'agent téléphone à l'employé) ou si si le scénario fait apparaître des interactions entre des entités autres que les acteurs et le système

-10% à -30% les séquences sont mal expliquées/peu détaillée (ou compter 5 au lieu de 10% quand c'est mal expliqué la séquence)

Jusqu'à -50% si les pré et post condition sont incohérentes avec le scénario nominal

Question 2.3 : (3,5 pts) Proposez un diagramme de classes d'analyse pour ce problème. On ne détaillera pas les opérations de la classe (fictive) représentant le système.



Barème : sur 100

15% classe vendeur, avec ses données perso + bancaires

15% classe acheteur, idem

10% vendeur lié à * ventes

10% une vente comporte la description de l'objet (cat, titre, photo, prix)

10% un objet lié à * MotClé, matérialisés comme une classe

10% on distingue les ventes directes et aux enchères (héritage par exemple)

10% la vente aux enchères comporte une date de fin/ un état

20% une classe enchère, a une date, un prix, un acheteur, une cible (objet en vente mode enchère)

-15% si associations orientées, compositions etc...

-15% si opérations sur les classes
 -10% à 20% pour toute autre faute ou aberration
 -10% aucun commentaires, aucune note.

Question 2.4 : (1,5 pts) Rédigez un test de validation couvrant partiellement le cas d'utilisation décrit en 2.2.

TV06 : Test nominal « mise en vente directe »
 Contexte : Le vendeur « Toto » est loggé (TV03)
 Entrée : « Vélo rouge », « TBE », « velo.jpg », « 20 eu », « cycles »
 Scenario :

1. Le vendeur sélectionne mettre en vente
2. Il saisit le titre « vélo rouge »,
3. Il saisit la description « TBE »
4. Il saisit le prix « 20 euros »
5. Il attache le fichier d'image « velo.jpg »
6. Il sélectionne « cycles » parmi les mots-clés proposés
7. Il valide

Résultat attendu : le système affiche une confirmation et crée un objet en vente

Moyen de Vérification : visuel pour la confirmation, Le « vélo rouge » doit être visible en browsant par « mot clé » dans l'interface principale.

Barème :

- Cohérence Q2.2 : 50% = toutes les actions décrites doivent être réalisables à la lecture de 2.2, on ne décrit dans le test que les action utilisateur sauf le R.A. Le R.A. doit être cohérent avec les post-conditions du use case. Si incohérence quelconque 0%.
- 30% reproductible : le contexte doit être suffisamment posé pour être reproductible, le verdict clair et non ambigu.
- 20% moyen de vérification autre que « visuel ».
- -20 à -50% non-respect des rubriques, incohérence du texte, illisible/incompréhensible

3. Conception : Suivi des commandes [5 Pts]

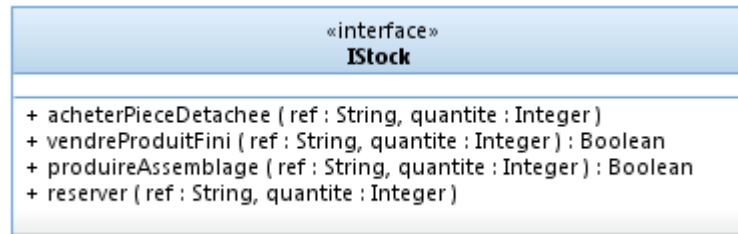
Rappels d'analyse : Nous considérons une application permettant la gestion de stocks et de production d'une usine de cycles. Un produit est muni d'une référence unique, d'une description et d'un prix. Certains produits sont des pièces détachées, achetées par l'usine auprès de fournisseurs. D'autres produits sont des pièces assemblées dans l'usine, en utilisant d'autres produits comme constituants.

Le travail de conception a déjà été bien abordé en Janvier. Récapitulons un peu ce que l'on a obtenu.

Le composant **CStock** gère et maintient à jour trois listes de produits :

- Les produits **disponibles** sont immédiatement en stock et n'ont pas été réservés.
- Les produits **réservés** sont présents dans le stock, mais font l'objet d'une réservation pour honorer une commande

- Les produits **demandés** ne sont pas présents dans le stock, il faudra les acheter auprès d'un fournisseur (produit de type pièce détachée) ou les produire dans l'usine (produit de type pièce assemblée).



Il offre l'interface suivante :

- `acheterPiecDetachee` : permet d'ajouter *quantité* occurrences de la pièce détachée portant la référence *ref* au stock. Cette opération est invoquée quand le fournisseur livre des pièces à l'usine. Si le stock contient 70 **demandes** pour *ref*, et si on en achète 100 unités, le stock contiendra 70 produits **réservés** et 30 produits **disponibles**.
- `vendreProduitFini` : permet d'enregistrer la sortie d'usine (et donc du stock) de *quantité* occurrences d'un produit assemblé fini, pour honorer les commandes. Les produits à vendre doivent impérativement être **réservés** dans le stock, ou l'opération échoue et rend « false ».
- `produireAssemblage` : permet d'enregistrer la fin d'une opération de production, c'est-à-dire l'assemblage de *quantité* occurrences d'une pièce assemblée *ref* dans l'usine. Les pièces nécessaires à l'assemblage sont retirées du stock, en prenant d'abord dans les pièces **réservées**, puis dans les pièces **disponibles**. L'opération échoue et rend « false » si les constituants ne sont pas présents dans le stock. L'ajout des produits nouvellement assemblés suit la même règle que l'achat de pièces détachées : on bascule de **demandé** à **réservé**, l'excédent éventuel devient **disponible**.
- `reserver` : permet de préparer la satisfaction des commandes, tous les produits figurant sur une commande sont réservés quand la commande est enregistrée. Les éléments passent de **disponible** à **réservés**, s'il n'y a pas assez de produits en stock, le reste est ajouté sous la forme de demandes. Les nouvelles demandes de produits assemblés nécessitent récursivement de réserver les constituants de l'assemblage.

Exemple :

Catalogue : Une « roue » est constituée de 50 « rayon » et 1 « jante ». « rayon » et « jante » sont des pièces détachées.

1. Initialement le stock contient :

- **Disponible** : 10 jantes, 3 roues
- **Réservé** et **Demandé** sont vides

2. Une nouvelle commande contenant 4 roues est saisie. On réserve donc 4 « roue » ; le stock devient :

- **Disponible** : 9 jantes
- **Réservé** : 1 jante, 3 roues
- **Demandé** : 1 roue, 50 rayon

3. On achète 1000 rayon au fournisseur ; le stock devient

- **Disponible** : 950 rayon, 9 jantes
- **Réservé** : 50 rayon, 1 jante, 3 roues
- **Demandé** : 1 roue

4. On produit l'assemblage d'une roue ; le stock devient :

- **Disponible** : 950 rayons, 9 jantes
- **Réservé** : 4 roue

• **Demandé** : vide

5. On vend les 4 roues ; le stock devient :

- **Disponible** : 950 rayons, 9 jantes
- **Réservé** et **Demandé** sont vides

Question 3.1 (2,5 points) On souhaite que l'interface graphique du gestionnaire de commande (composant CIHM) soit toujours à jour vis-à-vis de l'état du stock. On veut donc que le composant CStock signale ses changements d'état à tout autre composant intéressé (ici CIHM). Plus précisément, on souhaite notifier les composants abonnés à chaque modification du nombre de pièces **réservées**, sous la forme d'un tuple < REF, QTE >. REF est une référence d'article, par exemple « rayon ». QTE est une quantité entière, qui peut être positive ou négative.

Dans l'esprit du DP Observer, où le stock joue le rôle de sujet (observable), définissez la ou les interfaces utiles pour réaliser ce besoin. Représentez sur un diagramme de composant CIHM et CStock.

On va donc avoir 2 interfaces :

20% ISujet : addObs(IObs o), remObs(IObs o) + éventuellement notifyObs(), mais souvent on ne la met pas dans l'interface (elle est typiquement "protected"). Au minimum addObs doit être présent.

20% IObs : notifyChange(String ref, int qte) => on doit voir les données a priori

30% Donc CStock offre : IStock et ISujet et utilise IObs

30% Et CIHM : utilise IStock et ISujet et offre IObs

Question 3.2 (2,5 points) Sur un diagramme de séquence, expliquez les interactions entre CIHM et CStock, pour couvrir les étapes 1 et 2 du scénario exemple. On modélisera les abonnements en particulier.

CIHM->CStock : addObs(this)

CIHM->CStock : reserve("roue",4)

(Imbriqué) CStock->CIHM : notify("roue",3)

(Imbriqué) CStock->CIHM : notify("jante",1)

20% les lignes de vies (instances)

20% par message, bien positionné