

M1 : Ingénierie du Logiciel

UNIVERSITE PIERRE & MARIE CURIE (PARIS VI)

Examen Réparti 2eme partie

16 Mai 2013 (2 heures avec documents : tous SAUF ANNALES CORRIGÉES).

Barème indicatif sur 20,5 points (max 20/20).

1. Questions de cours

[2,5 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.

Q1.1 : Que définit un méta-modèle par rapport à un modèle donné ? (1 point)

Il définit l'ensemble des classes, des attributs qu'elles portent et des potentielles références (ou associations) entre classes. Le modèle ne contient que des instances de ces méta-classes.

Barème :

50% dès qu'on explique instance de

100% explication claire et complète, et/ou diagramme des méta niveaux à Video, Class... standard d'UML (au moins ils sont venus en cours même si ce n'est pas une réponse vraiment on-topic). On peut donner 75% pour une réponse moyennement claire.

Q1.2 : Qu'est-ce qui caractérise les méthodes agiles ? Citer une méthode agile. (1,5 point)

Agile manifesto cité en cours :

«

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on

the right, we value the items on the left more.”

Donc on peut citer réactivité, itérations courtes, moins de documentation, beaucoup de tests, confrontation directe client, rôles moins marqués au sein des équipes de dev...

On peut en citer d'autres empruntés plus spécifiquement à une méthode qualifiée d'agile, comme SCRUM, XP, ... comme la programmation en binôme ou le « daily Scrum » mais les réponses au dessus sont plus générales et répondent mieux à la question.

70% les valeurs de l'agile, i.e. 35% par point cité qui colle, on donne les 70 dès que 2 points au moins sont cités. On donne 70% si on arrive à citer le manifesto. Si on ne fait que décrire une méthode spécifique comme SCRUM, on donnera 30% donc ça fera 60% sur la question au total.

30% citer une méthode agile

2. Problème: eConf [12 Pts]

Le système eConf doit permettre de faciliter l'organisation de conférences scientifiques internationales.

Une conférence est gérée par un comité de programme, qui doit décider des articles qui seront acceptés pour être présentés à la conférence.

Les organisateurs de la conférence inscrivent la conférence sur le site, et constituent le comité de programme dans une étape que l'on ne traitera pas dans cet énoncé. On considérera donc que tous les membres du comité sont déjà inscrits sur le site. On se limite dans cet énoncé à la procédure de soumission.

La procédure de soumission d'un article est décomposée en trois phases : la soumission, la réponse, et la finalisation des articles acceptés. Chaque conférence définit des dates qui signalent la fin de chacune de ces phases, la date de limite de soumission est interprétée à GMT+11 (« anywhere on earth »).

Les auteurs désirant soumettre suivent un lien « soumettre » mis en place sur la page d'accueil de la conférence. Pour cela au moins l'un d'entre eux doit être inscrit sur le site.

Pour s'inscrire il faut fournir email, nom, prénom et institution. On est alors invité à choisir un login et un mot de passe. Le système teste si un compte associé à cet email existe déjà, et le cas échéant renvoie par email les identifiants de connexion à l'utilisateur distrait. Le compte est enfin activé en cliquant sur un lien envoyé à l'adresse email d'inscription.

Les comptes créés sur eConf permettent de gérer éventuellement plusieurs soumissions et/ou plusieurs rôles. Par exemple, une personne peut être auteur d'un article soumis dans la conférence A et membre du comité de programme d'une conférence B. Quand un utilisateur se connecte par la page d'accueil de eConf, on lui propose de choisir parmi les rôles dont il dispose actuellement celui qu'il veut incarner. Quand un utilisateur inscrit suit le lien « soumettre », son rôle bascule sur auteur, et on lui propose de remplir le formulaire de soumission pour un nouvel article.

Pour soumettre un article il faut fournir, un titre, un court résumé (l'abstract), et un fichier au format pdf (l'article lui-même). Si l'article est écrit à plusieurs (ce qui est souvent le cas) il faut aussi fournir la liste des auteurs. Les co-auteurs sont cités par nom, prénom et email. Le système fait une recherche pour trouver un compte qui correspondrait à chaque co-auteur et propose d'utiliser le compte associé le cas échéant.

L'ensemble des auteurs (qu'ils soient inscrits ou non) est notifié par email à chaque étape de la soumission. En particulier, quand le formulaire de soumission est validé, le système enregistre l'article et envoie un email à tous les auteurs. A partir de cette première soumission, le rôle auteur sur cet article sera proposé aux utilisateurs concernés au moment de la connexion.

La soumission d'un article doit être réalisée avant la date de limite soumission ; jusqu'à cette date, la soumission peut être consultée, modifiée et retirée. Passé cette date, il est encore possible pour les auteurs de retirer une soumission mais pas de la modifier. Les auteurs peuvent modifier la soumission en éditant le titre et le résumé et/ou en soumettant une nouvelle version de l'article en pdf. Les différentes versions soumises du pdf restent visible et ne sont pas effacées.

Dans la phase suivante du processus, les membres du comité de programme vont soumettre des évaluations sur les articles soumis. Tous les membres du comité de programme peuvent consulter la liste des articles soumis et accéder aux articles pour choisir ceux qu'ils souhaitent évaluer. Une évaluation est constituée de 4 notes entre 1 et 5, évaluant respectivement la qualité technique, l'originalité, la qualité de la rédaction et la pertinence par rapport aux thèmes de la conférence. Elle est accompagnée d'un texte expliquant les notes attribuées, à destination des auteurs de l'article, et éventuellement d'un second texte qui ne sera visible que des autres membres du comité de programme.

Quand les évaluations ont été soumises, commence la phase de discussion, où les membres du comité de programme peuvent éditer leur évaluation et discuter de l'article : une page de discussion visible uniquement des membres du comité est prévue à cet effet. Il porte un fil de discussion avec les messages datés que s'échangent les membres du comité.

A l'issue de cette étape, le comité de programme prend (offline) la décision d'accepter ou de rejeter les articles soumis, et inscrit ces choix dans le système. Les auteurs sont alors notifiés de la décision par un email type

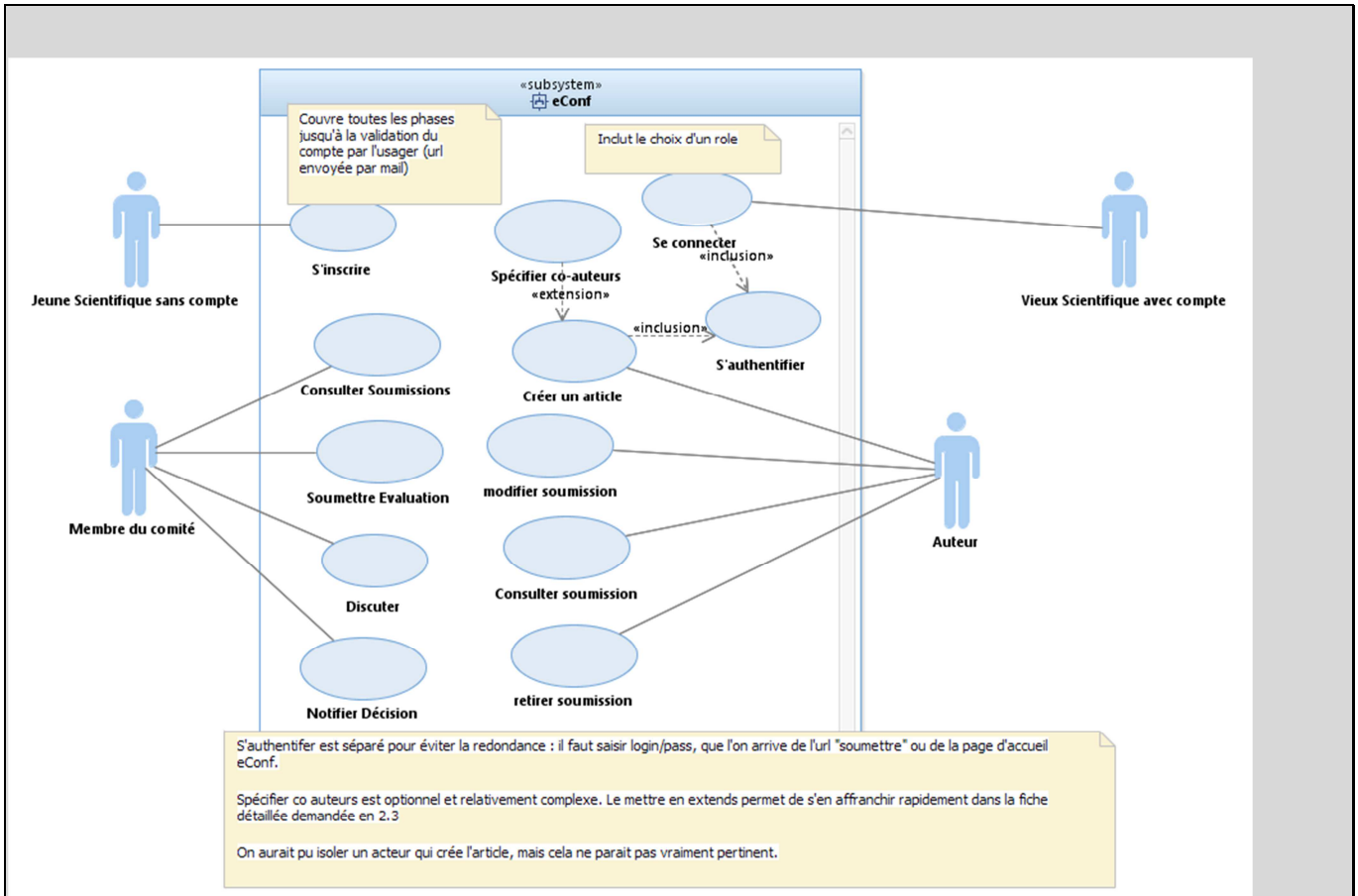
(accepté ou refusé) accompagné des évaluations fournies par les membres du comité. Les évaluations sont également visibles par les auteurs de l'article sur eConf.

Si l'article est accepté, les auteurs disposent d'une période pour modifier leur article, afin de prendre en compte les remarques des évaluations. Pendant cette période ils peuvent de nouveau mettre à jour le pdf de l'article.

A la date limite de soumission du manuscrit final, les articles sont collectés pour être ensuite envoyés à l'éditeur, mais on ne traitera pas cette étape.

Question 2.1 : (4 pts) Réalisez le diagramme de cas d'utilisation du système.

Commentez et/ou annotez un minimum le diagramme.



Barème TODO

Sur 130 à saisir tel quel dans la feuille.

Acteurs (*3) : 30% pour les 3 acteurs principaux. On peut fusionner les acteurs Jeune et vieux scientifique sans que ça pose de problème.

Use case * 10 : 10% par UC principal correctement identifié **et lié à un acteur adapté. 0% pour chaque use case mal lié ou mal modélisé**

+5 % : On peut faire hériter auteur et Membre du PC de Scientifique avec compte.

Par contre l'héritage PC member dérive de Auteur n'est pas adapté a priori, sauf s'il sert à factoriser certains use case (il faut que ce soit expliqué, car au niveau logique métier c'est incorrect). Si c'est expliqué on donnera les points + UC compte double consulter soumission vaut 20%.

Si les acteurs « acteur sans compte » et/ou « acteur avec compte » ne sont pas identifiés, mais que les UC « s'inscrire » et/ou « login » sont identifiés (et faisables par tous les acteurs, i.e. modélisation cohérente) on donnera les points sur ces UC (pas de double peine).

Choisir un rôle séparé de se connecter est légitime mais on ne compte que 10% pour les 2.

+10% extends/include ?

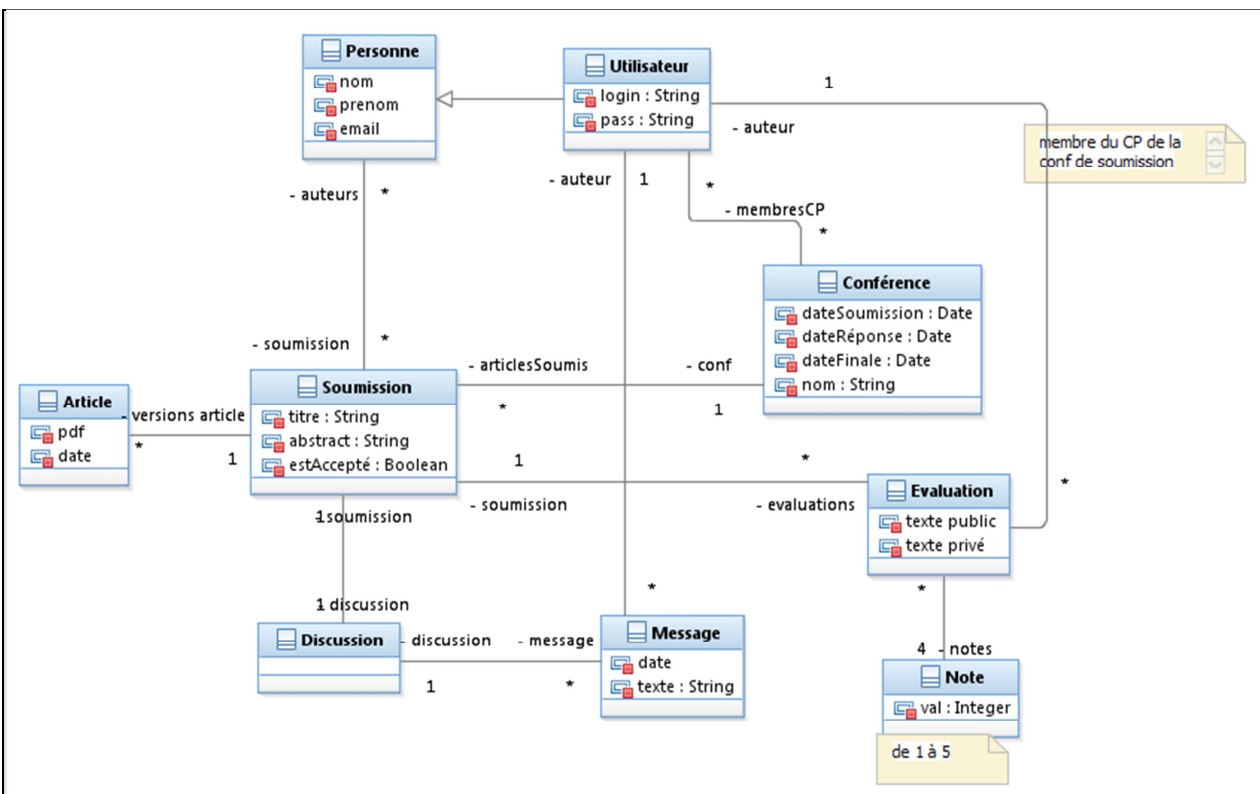
On pourra trouver pas mal de variantes correctes pour ...si la modélisation est correcte on donnera les 20%.

On ne sanctionne pas un use case « login », s'il est correctement modélisé, mais on ne donne pas de points non plus.

-10 à -20% par faute de syntaxe...

Question 2.2 : (4 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 : à affiner j'ai beaucoup 150 pourcent

20% Modélisation utilisateur/compte :

- 10% attributs : 5% identifiant/mdp et (5%) pour les trois attributs nom/prenom/email
- 10% Classe Personne matérialisée séparément

50% description de la classe soumission :

- 10% la classe (5%) titre/abstract (5%)
- 10% booleén est accepté
- 10% les versions de l'article correctement modélisé lien * sur soumission

- 20% lien avec les auteurs, on donne seulement 10% si le lien n'admet pas que les co-auteurs ne soient pas nécessairement inscrits

20% modélisation des discussions :

- 10% classe Discussion, liée à une soumission et à des messages
- 10% classe message avec un contenu et lié à un auteur.

30% La conférence

- 10% la classe (5%), avec des dates (5%)
- 10% liée aux * membres du comité de programme, éventuellement avec une indirection via une classe Comité de programme
- 10% lien 1-* sur les soumissions

30% les évaluations

- 10% Avec un auteur
- 10% avec deux textes
- 10% avec les 4 notes

-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 2.3 : (4 pts) Réaliser la fiche détaillée du ou des cas d'utilisation(s) couvrant les étapes qui permettent à un scientifique qui a déjà un compte de soumettre un article dont il est le seul auteur à la conférence « International Science Conference ISC'2013».

Titre : Créer un article

Acteur : Auteur

Hypothèse : aucune.

Pré : aucune

Post : La soumission d'un article par l'utilisateur/auteur est enregistrée..

Scénario :

1. L'utilisateur suit le lien « soumettre » mis en place sur la page d'accueil de la conférence.
2. Le système contrôle l'authentification, cf. cas d'utilisation « s'authentifier »
3. Le système affiche le formulaire de soumission
4. L'auteur saisit le titre, et un résumé
5. L'auteur attache un fichier pdf contenant l'article
6. L'auteur valide

7. Le système enregistre les informations sur l'article et affiche une confirmation.
8. Le système envoie un email type de confirmation à tous les auteurs de l'article.

Alternative A1 : spécifier co-auteurs

En SN5, si l'article est écrit à plusieurs, l'auteur saisit la liste des co-auteurs, cf cas d'utilisation « spécifier co-auteurs »

Alternative A2 : Problèmes de saisie

En SN7, si les informations sont incomplètes ou invalides, le système affiche un message indiquant la nature du problème. Retour en SN3.

Titre : s'authentifier

Acteur : Titulaire Compte, auteur

Hypothèse : aucune.

Pré : aucune

Post : L'utilisateur est identifié par le système.

Scénario :

1. Le système vérifie si l'utilisateur que l'utilisateur n'est pas encore connecté.
2. Le système affiche une page de saisie des identifiants login/pass
3. L'utilisateur saisit son login et son mot de passe et valide
4. Le système contrôle la validité des informations
5. L'utilisateur peut continuer à naviguer sur eConf en étant identifié comme un utilisateur inscrit.

A1 : Utilisateur déjà connecté

En SN1, si l'utilisateur est déjà connecté, aller directement en SN5.

A2 : Mot de passe erroné

A2.1 En SN4, si le mot de passe ne correspond pas au compte, le système propose de renvoyer ses identifiants par email à l'utilisateur

A2.2 Si l'utilisateur valide le système émet un email portant les informations de l'utilisateur.

A2.3 retour en SN2

Barème : sur 110% majoré à 100%

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

-10 à +10% cohérence globale du texte, utilisation correcte des champs Pré/Post/Scénario etc... En particulier, -10% si les préconditions/hypothèses sont testées dans le scénario.

+10% une pre ou post-condition identifiée (genre article enregistré).

+10 la connection : saisie des identifiants

+10 saisie titre/résumé

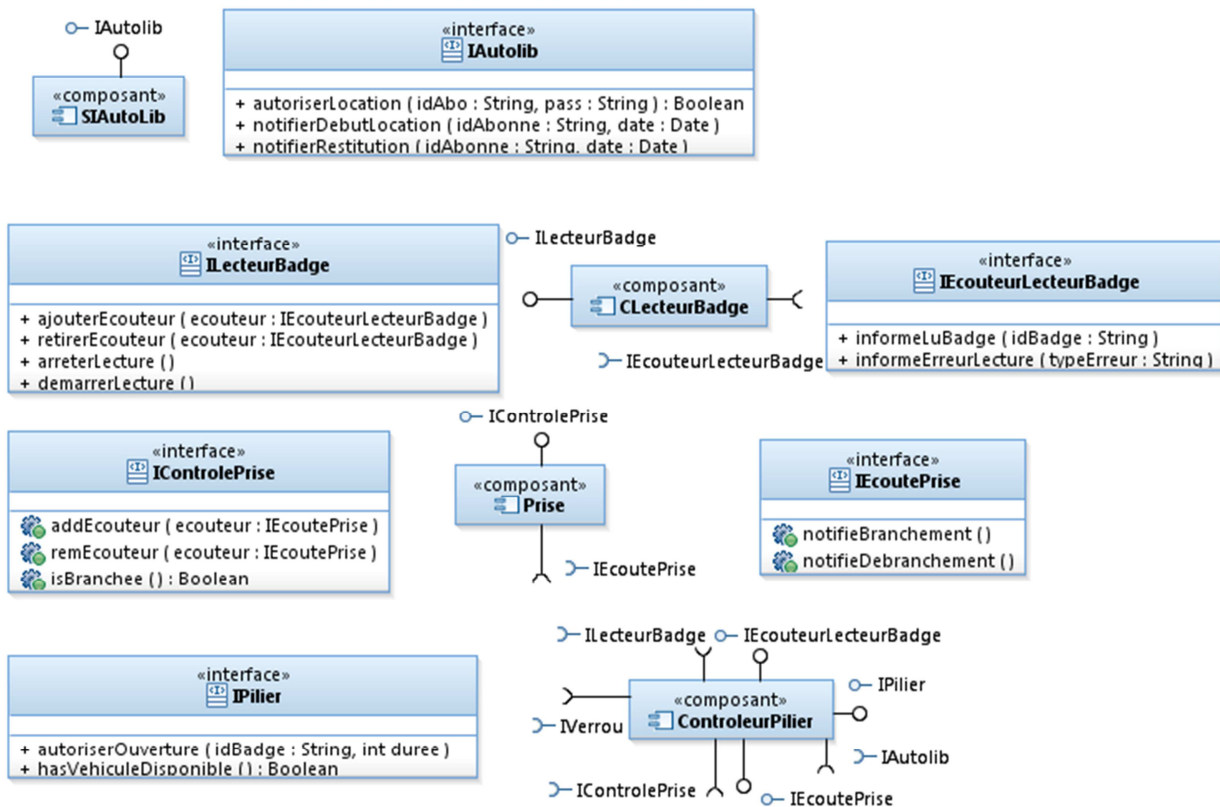
+10 les divers affichages du système (e.g. le système affiche un formulaire de soumission)

- +10 l'article lui-même (fichier à attacher). On devrait avoir une étape propre pour cela (distincte du tire/résumé) vu qu'attacher un fichier est un peu plus compliqué.
- +10 on voit la possibilité de saisir des co-auteurs (même si elle n'est pas pleinement spécifiée)
- +10 envoi d'un email aux utilisateurs
- +10 par alternative sur saisie incorrecte (identifiants et/ou fichiers/ infos diverses max 20%). Compter 0 si c'est indiqué « alternatif » mais que ça viole les post-conditions. +10 aussi si on identifie des cas exceptionnels genre date de soumission dépassée.
- +10 la modélisation est cohérente avec les use case de 2.1 (ne pas donner si on n'a pas répondu à 2.1)
- 10% le client 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 scénario
- 15% :Spécification d'étapes hors système comme étapes du scénario (e.g. l'agent téléphone à l'employé) .
- 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
- Jusqu'à -50% si le scénario fait apparaître des interactions entre des entités autres que les acteurs et le système (a priori bornes et voiture font partie du système, du moins ils génèrent des actions visibles)

3. Conception : Pilier de Chargement AutoLib [6 Pts]

NB : les descriptions suivantes sont reprises de l'examen de Janvier 2013.

On s'intéresse aux piliers qui contiennent les prises auxquelles sont connectées les BlueCar dans le système AutoLib. Un pilier est représenté au niveau logiciel par le composant ControleurPilier décrit ici avec ses dépendances. Les piliers de chargement sont constitués d'un lecteur de badge du même modèle que celui de la borne de station, d'un verrou, et d'une prise de courant uniquement accessible quand le pilier est déverrouillé. L'ensemble est piloté par un contrôleur de pilier, qui interagit avec le contrôleur de la borne de station.



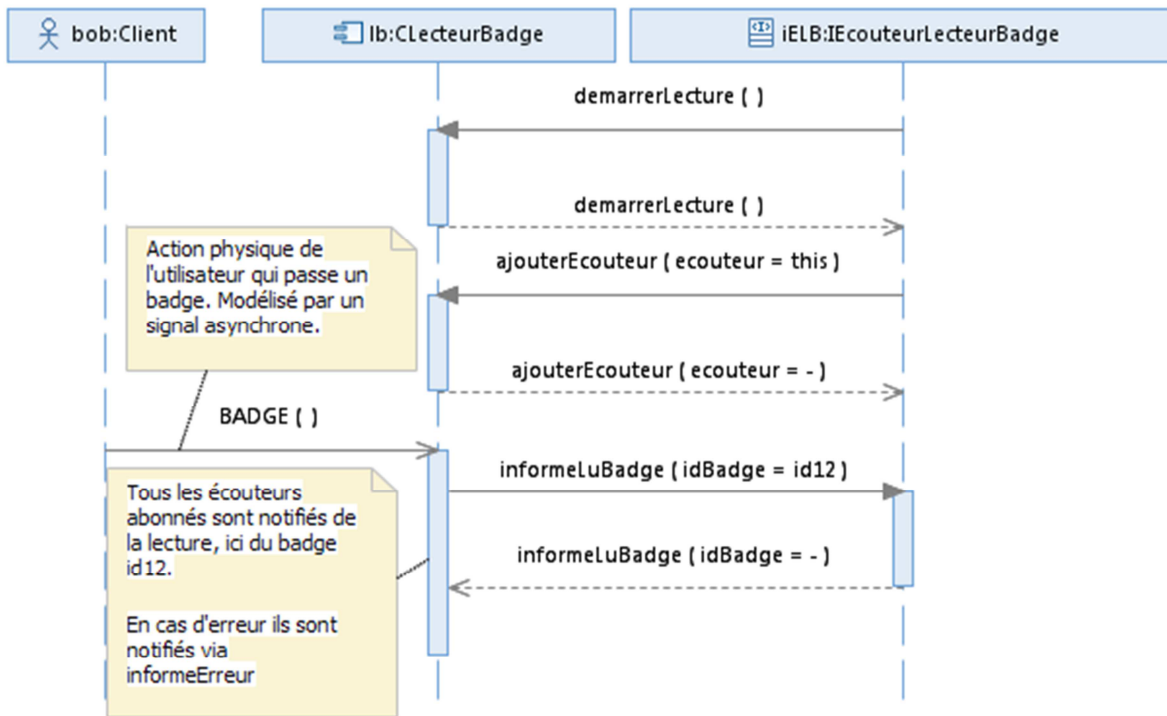
Les piliers de chargement sont des entités relativement autonomes qui bénéficient d’une connexion directe au SI Autolib. Le composant contrôleur d’un pilier de chargement peut prendre trois états :

1. Aucun véhicule n’est branché, le pilier attend qu’un utilisateur arbitraire utilise son badge pour débloquer le verrou et lui permettre de brancher son véhicule. Il notifie alors le SI Autolib’ du branchement en spécifiant la date et le numéro de l’abonné ayant branché le véhicule. Suite à un branchement, le pilier passe à l’état 2.
2. Un véhicule est branché et en charge, le pilier est verrouillé. Il attend de recevoir sur son interface de contrôle un ordre d’ouverture pour un abonné particulier pendant une certaine durée (timeout). Il passe alors à l’état 3.
3. Le pilier attend que l’abonné cité dans l’ordre d’ouverture présente son badge, ce qui déverrouille la borne et lui permet de débrancher le véhicule. Quand le véhicule est effectivement débranché, le pilier notifie le SI Autolib’ en lui fournissant le numéro d’abonné et la date. Le pilier revient alors à l’état 1. Si le timeout est rencontré sans que l’utilisateur n’ait badgé le pilier, il revient à l’état 2.

Une prise de courant peut être à un instant donné branchée à un véhicule ou débranchée et l’on peut interroger son état. Un écouteur abonné sur une prise de courant est notifié quand une voiture est branchée ou débranchée.

Le composant LecteurBadge est décrit comme suit :

- On peut démarrer ou arrêter la lecture. Quand la lecture est démarrée, le capteur RFID est sensibilisé, tout badge présenté devant le lecteur sera lu. Quand il est arrêté, passer un badge est sans effet.
- Les écouteurs s’abonnent au composant pour être informés quand un badge est passé sur le lecteur actif.
- Quand on passe un badge sur le lecteur actif, il tente de lire son identifiant. Si l’opération est un succès, tous les écouteurs abonnés en sont informés en invoquant « informeLuBadge » avec l’identifiant du badge lu en paramètre. Si l’opération de lecture échoue, c’est l’opération « informeErreurLecture » qui est invoquée, avec un message de diagnostic. Le lecteur reste actif, tant qu’on n’invoque pas arrêterLecture.



Question 3.1 : (2 pts)

A l'aide d'un diagramme de structure interne, montrez comment instancier ces composants pour constituer un pilier.

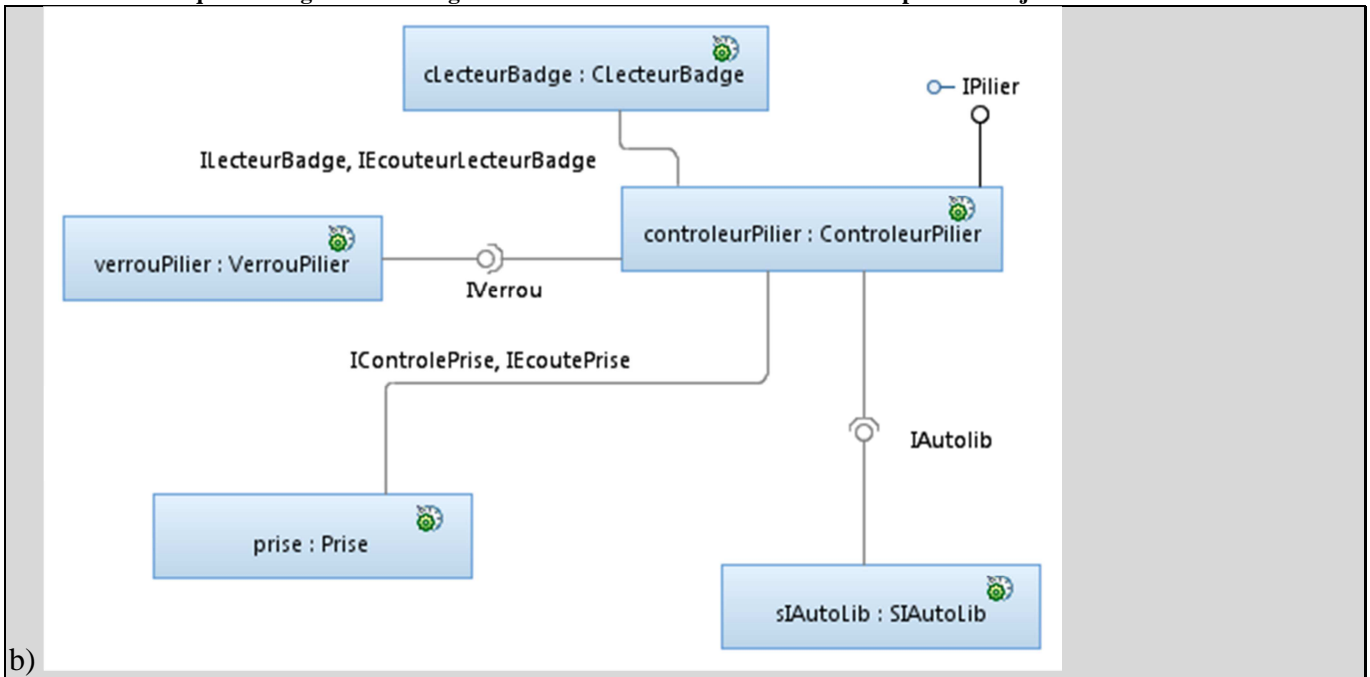
On attend une notation moins décomplexée que celle de RSA ici, les cycles étant représentés par deux connecteurs de dépendance correctement orientés.

Sur 100 :

- 40% : 4*10% par instance autre que controleurPilier
- 60% : 6*10% par lien correctement orienté et connecté

On ne compte pas IPilier (offert) modélisé ou non.

-20 ce ne sont pas clairement des instances.



Question 2.2 : (4 pts)

Représentez sur un diagramme de séquence de niveau composant les échanges permettant à un abonné porteur du badge 12 de retirer sa voiture. On démarrera la séquence par une invocation à l'opération « autoriserOuverture » du ContrôleurPilier. On détaillera les abonnements et désabonnements entre les composants.

Barème sur 100 :

Sur 20 les instances sont bien celles du diagramme précédent, on les donne si c'est des instances raisonnables (e.g. pas des lignes de vie typées IEcouteurLecteurBadge)

10 s'abonner au lecteur de carte

10 démarrer lecture NB : ça manque sur mon corrigé aussi, mais je n'ai pas le fichier pour m'aj la figure.

10 cleanup : dès qu'on voit au moins un arreterLecture ou remEcouteur dans la séquence.

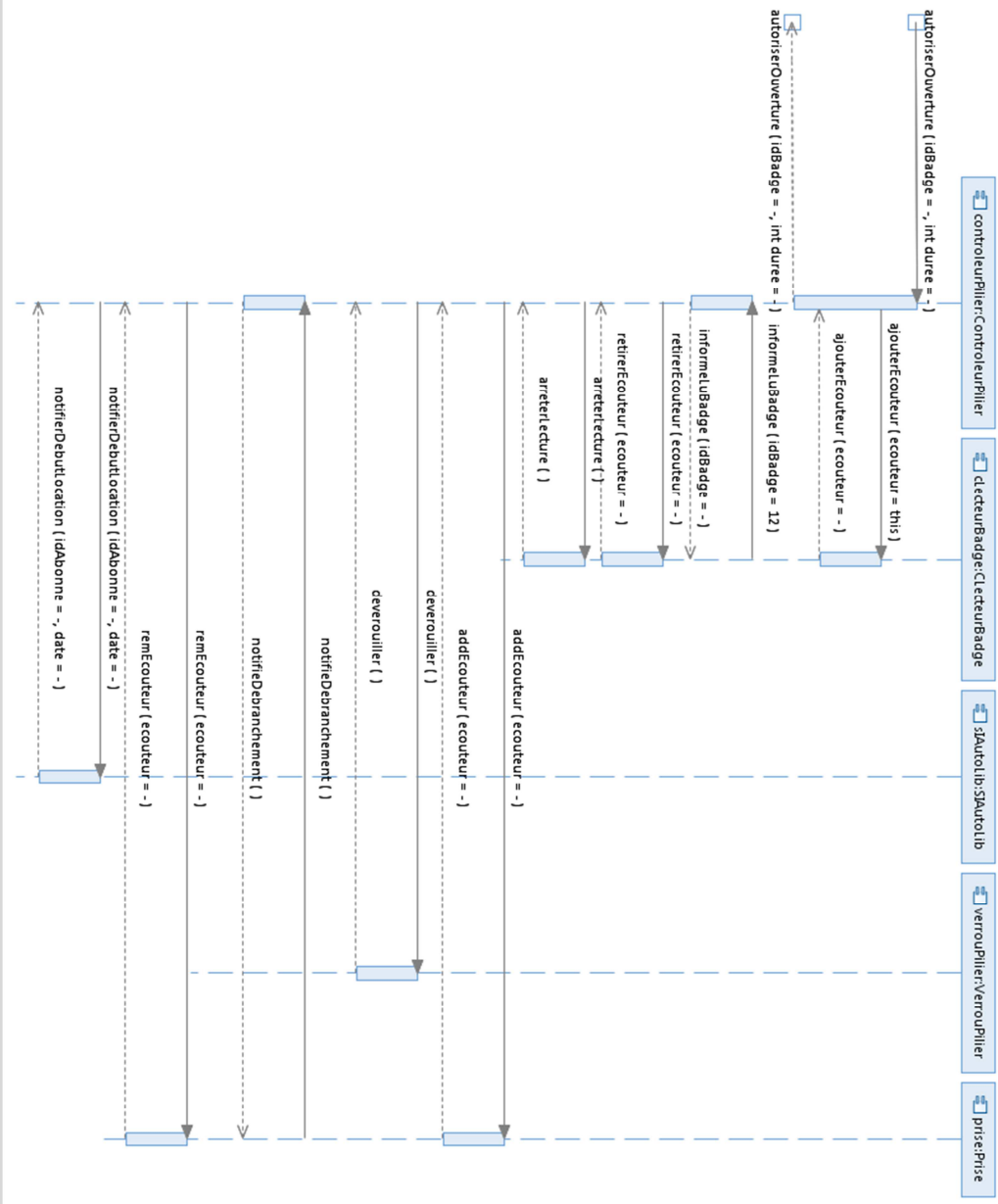
10 le lecteur carte notifie le contrôleur

10 déverrouiller pilier

10 s'abonner à la prise

10 recevoir la notification de la prise

10 notifier SI autolib



IPilier n'est pas demandé à ce stade.