

Examen Janvier 2012 (durée 2h) 2eme Session

Tous documents autorisés SAUF ANNALES corrigées
Barème *indicatif* sur 22 points

Questions de Cours (3 points)

1. Pourquoi l'obtention d'une certification qualité (CMMI, ISO 9001...) est-elle importante pour une entreprise ?

Car une certif aide à obtenir des contrats, surtout dans un domaine type appel d'offres public etc...
Ca augmente la confiance des clients plus généralement, particulièrement dans un contexte B2B.
C'est également nécessaire quand la taille de la structure augmente, d'avoir des processus d'entreprise maîtrisés.

50% réponse dans le sujet, on sait dire ce qu'est la qualité => augmenter la confiance
100% réponse convaincante

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

Pas de méthodo en UML !!
100% pas de méthodo
Sinon 0%

3. Quel est le rôle d'un diagramme de structure interne représentant des composants ?

Donner la topologie de connexion, l'instanciation des composants.
50% topologie/connections
50% instanciation

BDUfr: Gestion de notes (17 points)

L'université souhaite mettre en place un système permettant la collecte et l'analyse des notes d'unités d'enseignements (UE). Ce système appelé BDUfr doit remplir les objectifs suivants :

- Permettre la définition des UE et des inscriptions d'étudiants dans ces UE
- Permettre la définition par l'enseignant responsable d'une UE de contrôles correspondant à cette UE
- Permettre la saisie et l'édition de notes pour les contrôles d'une UE par l'enseignant responsable de l'UE
- Permettre la consultation par les étudiants des notes qu'ils ont obtenues dans les UEs où ils sont inscrits
- Permettre pour tout enseignant la consultation des notes obtenues par n'importe quel étudiant, et aussi la consultation des notes d'un groupe d'étudiants pour une UE.
- Permettre la bascule des notes finales dans la base nationale, étape nécessaire pour la délivrance des diplômes.

On fait les hypothèses suivantes sur le fonctionnement du système :

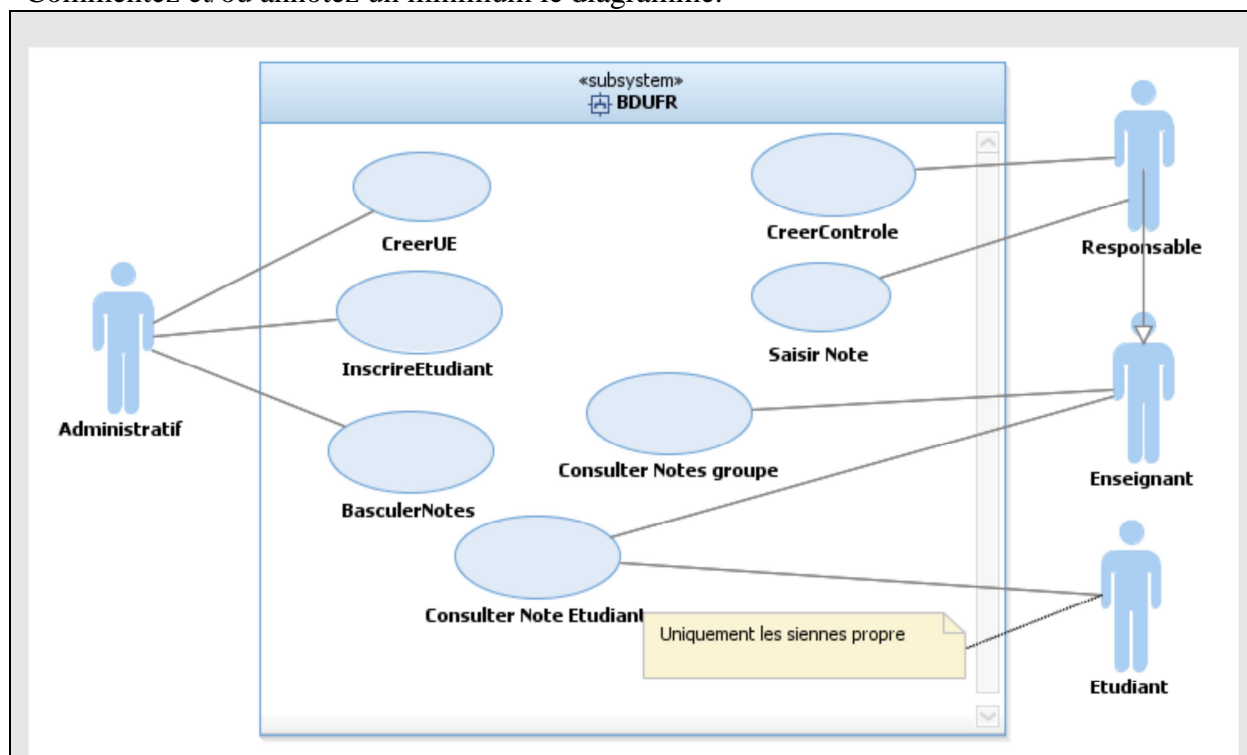
- Le processus d'authentification est géré via le LDAP (gestion centralisée des authentifications) donc il n'existe pas de phase de création de compte, et les logins/pass sont gérés au moment de la connexion au système. Chaque requête au serveur est donc estampillée par l'identifiant du demandeur. Le système doit cependant gérer les droits des différents utilisateurs ; on distingue de plus les catégories d'utilisateur « administratif », « enseignant » et « étudiant ».
- Chaque utilisateur est identifié par un numéro unique, qui est associé à son nom et prénom.

- La définition des UEs et l'inscription des étudiants aux UE sont réalisés par le personnel administratif. La définition de l'UE inclut la définition d'un enseignant responsable de l'UE, qui pourra créer des contrôles et saisir des notes.
- Un étudiant peut a priori s'inscrire à autant d'UEs qu'il le souhaite
- Pour définir un contrôle, le responsable d'UE donne son nom (par exemple : « Exam réparti 1 Novembre ») et sa catégorie : CC (contrôle continu), Examen, Partiel, TP, Note finale d'UE.
- Pour chaque contrôle défini dans l'UE, chaque étudiant inscrit possède une note, par défaut 0.
- La saisie des notes par l'enseignant responsable de l'UE peut se faire en éditant un formulaire web, ou en important une feuille de tableur (type CSV, i.e. comma separated values) qui porte dans la première colonne le numéro d'étudiant et dans la deuxième la note obtenue pour ce contrôle
- La bascule des notes dans la base nationale des notes universitaires est réalisée par le personnel administratif à l'issue des jurys. A dater de cette bascule, l'UE est notée « fermée » et l'ensemble des données qui y correspondent sont figées (protégées contre la modification).

PHASE ANALYSE (9 points)

Question 1 (2,5 points)

Réalisez le diagramme de cas d'utilisation du système.
Commentez et/ou annotez un minimum le diagramme.



Barème (sur 110):

Acteurs (*4) : 30% pour les 4 acteurs principaux

Admin, Etu : 5% chaque

Enseignant et responsable : 5% par acteur + 10% héritage

+10% (bonus) si acteur secondaire base nationale (U.C basculer note) bien modélisé.

Use case * 7 : 10% par UC correctement identifié **et lié à un acteur adapté**. **0% pour chaque use case mal lié ou mal modélisé**

On pourra trouver pas mal de variantes correctes pour la consultation des notes : de un à 3 use case selon que l'on distingue « consulter ses notes », « consulter note d'un étudiant », « consulter notes groupe » ou pas. Si on les a confondu, on doit trouver un commentaire qui explique qu'il y a plusieurs scénarios différents dans ce use case. Si la modélisation est correcte on donnera les 20%.

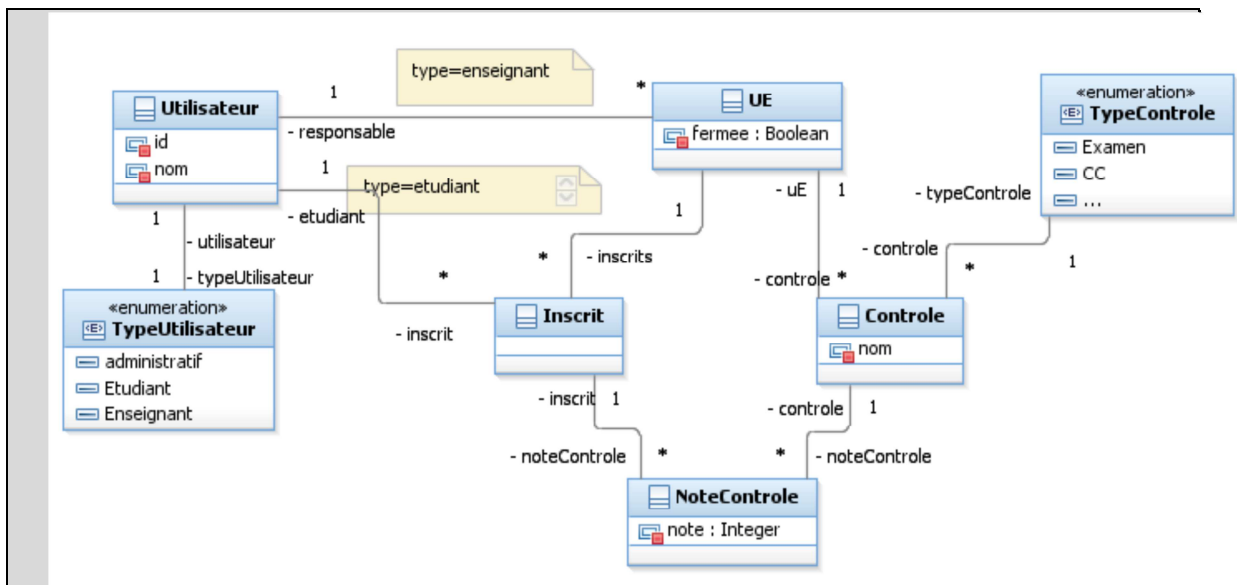
On ne sanctionne pas un use case « login », s'il est correctement modélisé, mais on ne donne pas de points non plus.

-10 à -20% par faute de syntaxe...

Question 2 (3,5 points)

Proposez un diagramme de classes d'analyse pour ce problème.

On ne détaillera pas les opérations de la classe (fictive) représentant le système.



Bareme :

Utilisateur peut aussi être modélisé avec de l'héritage.

Il y a beaucoup de façons de modéliser, utilisez votre talent naturel de notateur pour évaluer la pertinence de la réponse ☺

On doit trouver :

- 20% : Les utilisateurs 5%, leur nom 5%, leur id 5%, leur type 5%.
- 20% : Les UE 5%, avec un nom 5%, et un statut (fermée ou non) 10%
- 10% responsable d'UE (un enseignant)
- 10% inscrits à une UE (i.e. étudiants)
- 15% : Les contrôles (nom et type) 5%, liés à une UE 10%
- 25% : Les notes des étudiants liés à un étudiant et à un contrôle. La modélisation du fait qu'il doit être inscrit est difficile.

- 10% lien étudiant note (même si pas de corrélation avec inscription)
- 10% lien étudiant contrôle (on donne 5% si on lie à UE au lieu de contrôle)
- 5% classe note + valeur

Question 3 (3 points)

Réaliser la fiche détaillée du ou des cas d'utilisation(s) permettant à un enseignant de saisir les notes obtenues à un contrôle.

On doit trouver la spéc de charger note par note et (Alternatif sans doute) la spec de charger en bloc (CSV)

Hypothèse : L'enseignant responsable est connecté.

Pré : on a défini des contrôles

Post : Les notes sont affectées aux étudiants concernés

Scénario :

1. Le responsable navigue jusqu'à la page du contrôle
2. Il choisit l'action saisir notes
3. Le système affiche un formulaire avec les noms/numéros des étudiants inscrits dans l'UE et une case pour saisir la note (0 si pas de note actuellement)
4. Le responsable d'UE saisit/modifie les notes
5. Le responsable valide
6. Le système affiche les notes du contrôle qui viennent d'être saisies

Alternative A1 : Saisie par CSV

A1.1 : En SN.2, le responsable choisit l'option « charger depuis CSV ».

A1.2 : le système affiche un formulaire permettant de choisir le fichier portant les notes dans le système de fichier local

A 1.3 : l'utilisateur choisit le fichier portant ses notes et valide

A.1.4 : le système charge les notes à partir du fichier

Retour en SN6.

Exception E1 : Fichier incorrect

E1.1. en A1.4, si le fichier porte des données incorrectes, les notes ne sont pas chargées. Le système affiche un message indiquant la nature du problème.

E1.2. l'utilisateur valide

E1.3. le système affiche l'écran d'accueil du contrôle.

Barème :

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

+30% cohérence globale du texte, utilisation correcte des champs Pré/Post/Scenario etc...

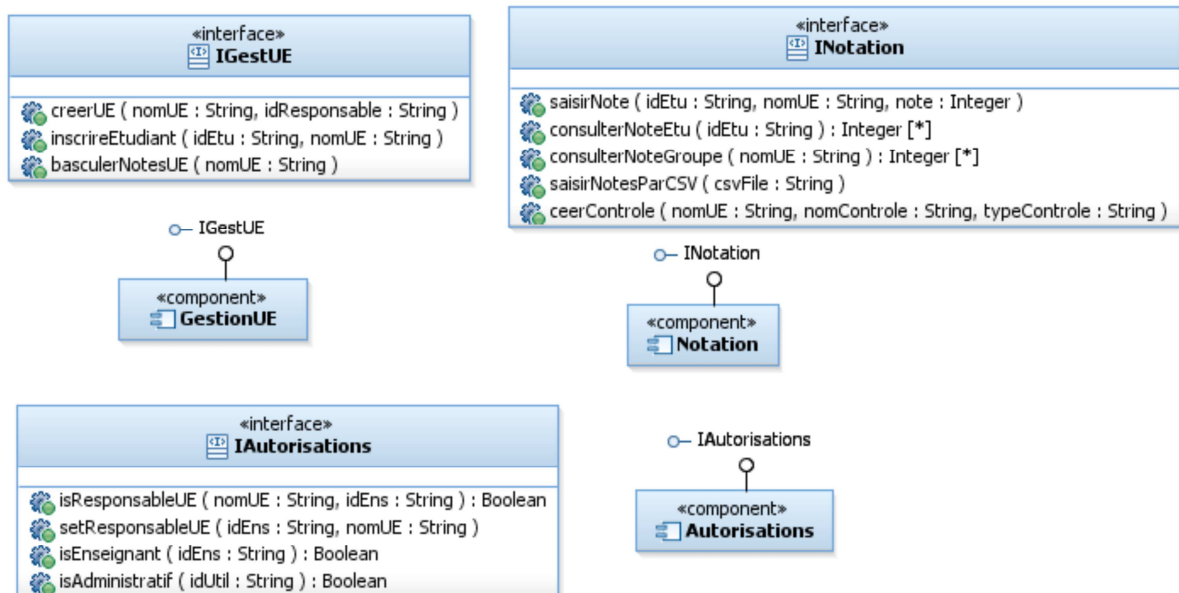
+10% l'acteur à l'initiative (déclencheur)

- +30% pour les étapes 2 à 6 correctement identifiées/couverte
- +20% pour l'alternative saisir notes par fichier
- +10% exception liée à un mauvais fichier

- 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) .
- 10% sur chaque séquence mal expliquée/peu détaillée
- 50% si on a découpé en plusieurs use case en question 1, mais que leur description détaillée n'est pas cohérente avec le diagramme.
- 50% si les pré et post condition sont incohérentes avec le scénario
- 50% si le scénario fait apparaître des interactions entre des entités autres que les acteurs et le système

PHASE CONCEPTION (8 points)

Dans une première étape de conception, on a identifié les composants et interfaces suivants :

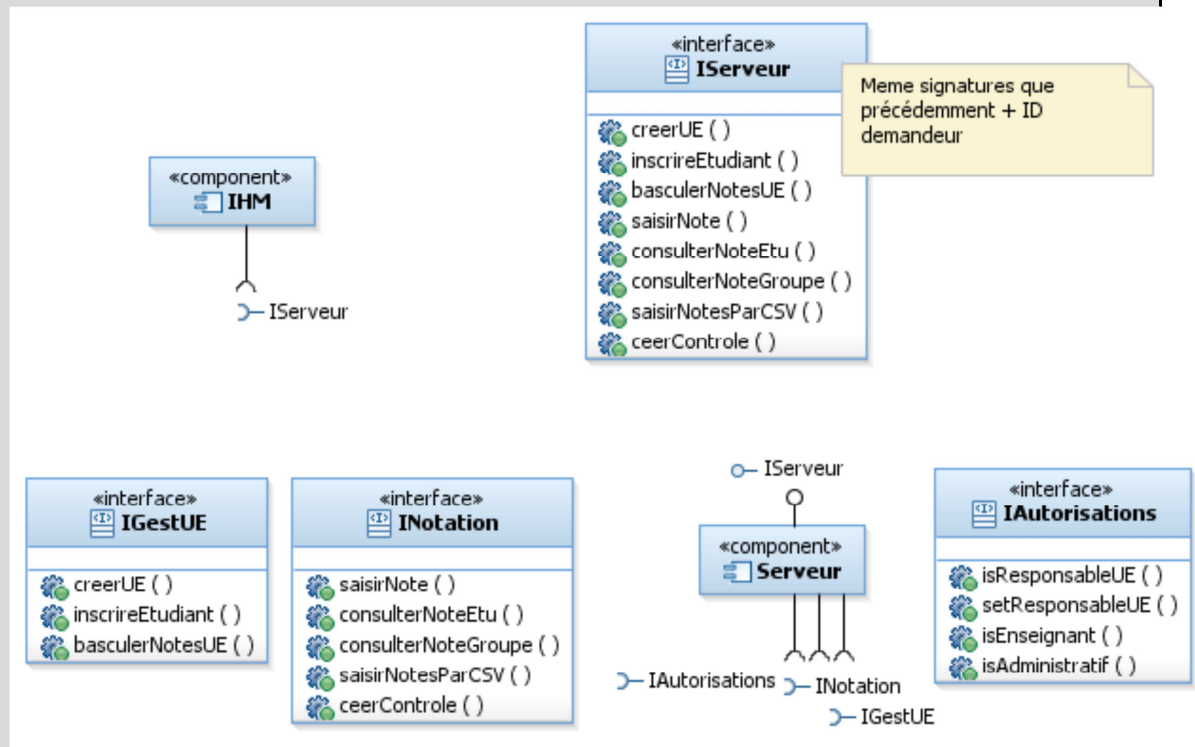


En plus de ces composants, on identifie un composant **Serveur** et un composant **IHM**. Le composant IHM représente la page web par laquelle les utilisateurs accèdent à l'application. Le composant **Serveur** intercepte toutes les demandes réalisées par l'IHM (qui portent en plus de leurs arguments propres l'identifiant de l'utilisateur qui fait la demande), contrôle les droits en s'appuyant sur le composant **Autorisations**, et transmet la demande au composant approprié (GestionUE ou Notation) pour le traitement.

Question 4 (1,5 points)

Modélisez sur un diagramme de composants les composants Serveur et IHM. On ajoutera au besoin des interfaces supplémentaires.

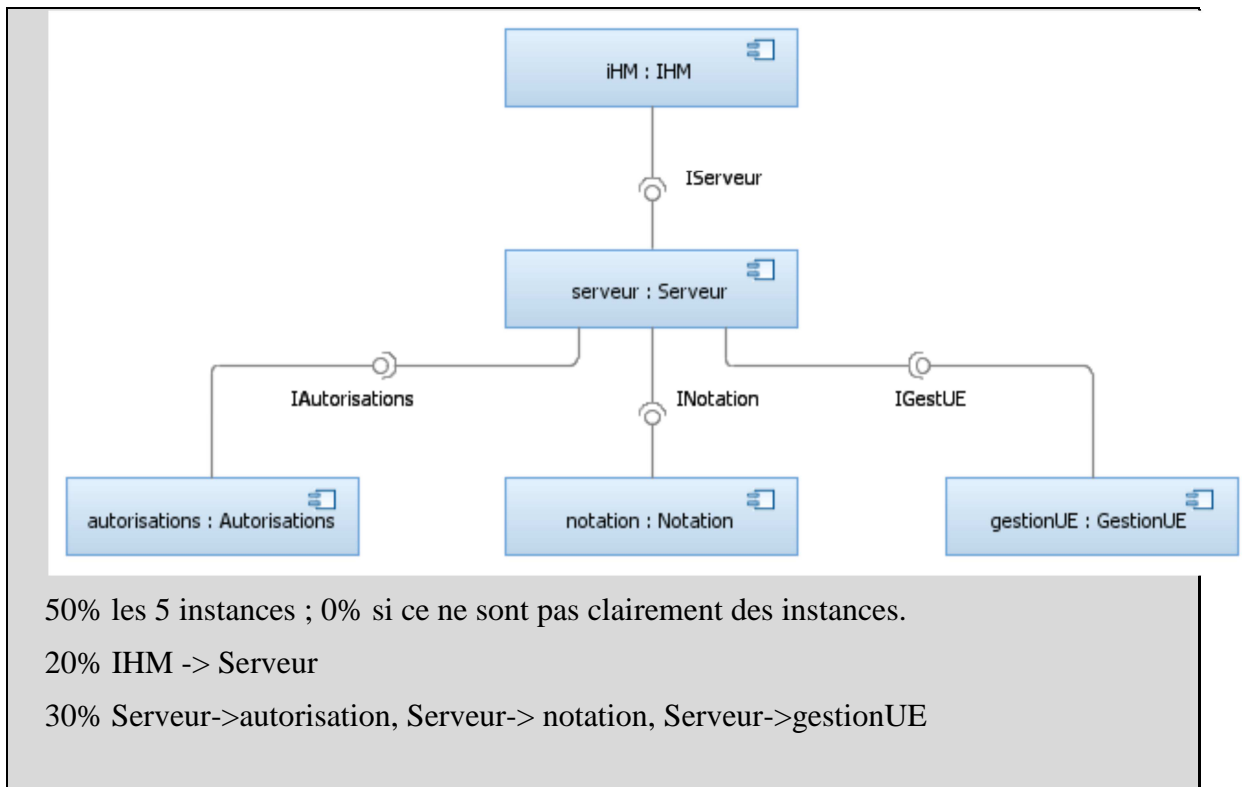
On doit trouver la spéc de charger note par note et (Alternatif sans doute) la spec de charger en bloc (CSV)



- 30% signatures avec demandeur posées dans une ou plusieurs interfaces
- 20% l'IHM n'utilise que le Serveur (itf requises). 0% si IHM offre une ou plusieurs itf.
- 40% Serveur requiert Autorisations (20%), Notation (10%) et GestUE (10%)
- 10% Serveur offre une ou plusieurs itf à l'IHM.
- 20% si Serveur requiert IHM

Question 5 (1,5 points)

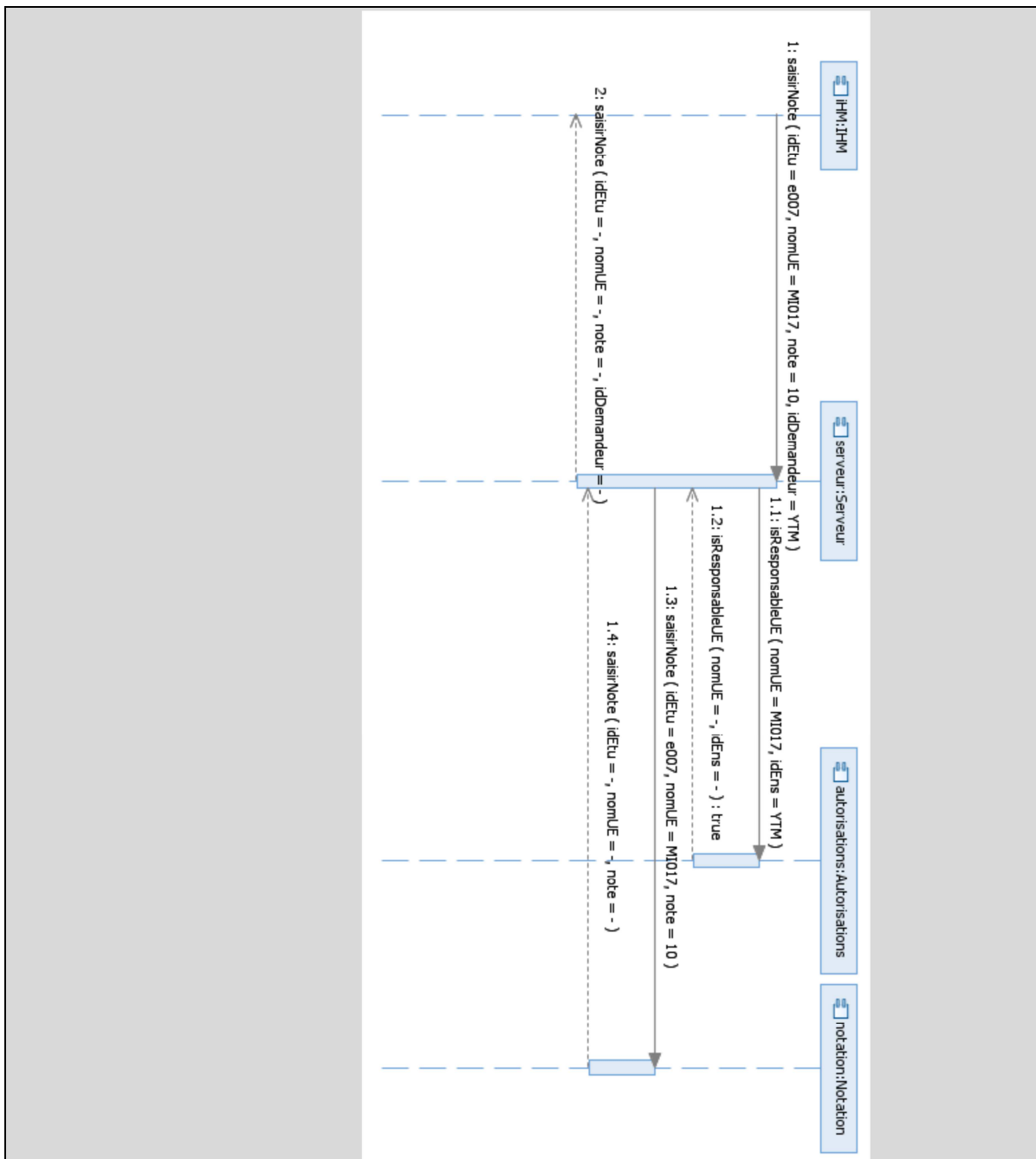
Modélisez à l'aide d'un diagramme de structure interne la configuration nominale des composants du système.



Question 6 (1,5 points)

Représentez par un diagramme de séquence de niveau intégration (où les lignes de vie représentent des composants) les interactions permettant à l'enseignant « YTM » d'affecter la note « 10/20 » à l'étudiant « e007 » pour l'ue « MI017-2011oct ».

Empty rectangular box for the answer.



10% on trouve les 4 lignes de vie

30% invocation de l'IHM sur le serveur avec note (10), id enseignant (YTM), idUE (MI017), idEtudiant (e007). 0% si IHM interagit avec d'autres composants que le Serveur.

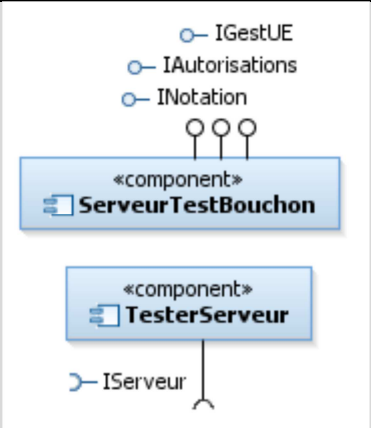
30% contrôle des droits. 0% si c'est l'IHM qui fait ce contrôle : cela permettrait de contourner la protection.

30% délégation sur Notation

-40% si on ne voit pas les paramètres.

Question 7 (0,75 point)

Quels éléments supplémentaires faut-il définir pour permettre de tester le composant **Serveur** en isolation ?



70% il faut définir un ou des bouchons pour les interfaces requises (/!\ cohérent avec avec Q4 et Q5)

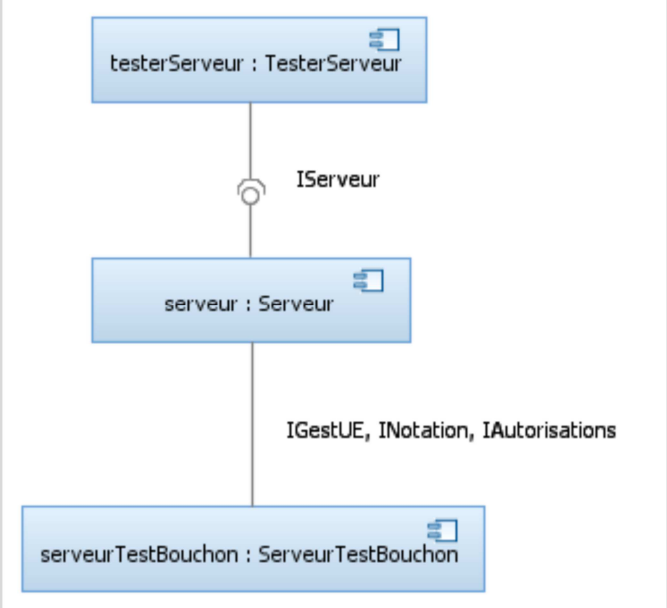
30% il faut un testeur

MAX 50% si on est incohérent avec Q4/Q5

Question 8 (0,75 point)

A l'aide d'un diagramme de structure interne, représentez une configuration des composants permettant de tester **Serveur**.

La question ne précise pas (malheureusement) qu'on attend un diagramme de composants.



20% le testeur

40% le composant à tester

40% le(s) bouchon

Maximum 50% si ce n'est pas clairement un diagramme de structure interne.

Maximum 50% si les liens sont mal orientés/nommés...

Question 9 (2 points)

Ecrivez un test d'intégration pour le composant **Serveur**.

Ce test d'intégration devra porter sur la séquence d'actions correspondant à la création d'une UE par un administratif, suivi de la création d'un contrôle par l'enseignant responsable de cette UE.

Précisez les données qu'il faut embarquer dans les éventuels composants/classes bouchon pour faire fonctionner ce test.

On veut voir :

```
cServeur.creerUE(admin1, YTM, MI017)
```

```
cServeur.creerControle(YTM, MI017, Examen2, Epreuve)
```

```
Assert ( cServeur.saisirNote(e007, YTM, 10/20 , MI017) ==true )
```

Selon ce que l'on a vu en Q6, il faut que le bouchon réponde correctement aux invocations.

40% une invocation aux opérations : creerUE, creerControle,

40% présence d'un résultat attendu : NB il faut être indulgent ici car l'API du sujet ne permet pas de tester les valeurs de retour.

20% données du bouchon cohérentes avec le test

On sera assez généreux à cette question, vu que l'énoncé avait quelques failles (quasi pas d'API en lecture sur les composants...).