

Examen Janvier 2010 (durée 2h) 2eme Session

Tous documents autorisés
Barème *indicatif* sur 21 points

Questions de Cours (3 points)

1. En SCRUM, quelles sont les trois questions auquel chacun répond lors de la mêlée quotidienne.
2. UML permet-il de modéliser entièrement une application ? Quels aspects sont les plus difficiles à modéliser ?
3. Expliquez la différence entre diagramme et modèle.

Gestion de Questions à Choix Multiples : QCM (18 points)

L'université souhaite mettre en place un système permettant l'évaluation des étudiants grâce à des QCM. Le système doit permettre aux enseignants de définir des QCM, et aux étudiants de passer les QCM. A la fois les enseignants et les étudiants peuvent ensuite consulter les résultats obtenus.

Pour définir une question, l'enseignant spécifie le texte de la question, et au moins trois réponses possibles. Chaque réponse est composée d'un texte et d'un verdict (réponse bonne ou fausse). Les questions sont gérées séparément des questionnaires, afin de permettre de définir un nouveau QCM en réutilisant des questions.

Pour définir un QCM, l'enseignant spécifie un titre pour le QCM et y ajoute les questions qu'il veut y voir figurer. L'outil lui permet de choisir des questions existantes ou d'en définir de nouvelles à la volée. L'outil contrôle également qu'une question donnée apparait au plus une fois dans un QCM. Enfin l'enseignant peut modifier le mode de notation du QCM: il affecte une valeur pour les réponses justes, fausses et vides. Le paramétrage par défaut est : réponse juste +1, réponse fausse -3, réponse vide 0. La note est obtenue en divisant le score par le nombre de questions et en ramenant à une note comprise entre 0 et 20.

Pour définir une épreuve, l'enseignant choisit un QCM existant (ou en crée un à la volée), définit une date et horaire (début/fin) où le QCM sera actif et donne l'intitulé du module (e.g. MI017-2010oct).

Les étudiants inscrits, à partir de l'heure de début de l'épreuve, peuvent se connecter pour répondre au QCM. Ils se connectent avec leurs login/mot de passe habituels, et le système interroge la base de l'université (e.g. dbUFR, donc déjà existant) pour contrôler ces identifiants et s'assurer que l'étudiant est bien inscrit au module. Pour chaque question, ils peuvent cocher une ou plusieurs réponses. Quand l'étudiant a fini, ou que la date de fin spécifiée par l'enseignant est atteinte, les réponses de l'étudiant sont enregistrées.

Passée la date de fin du questionnaire, enseignants et étudiants peuvent accéder aux résultats. Les étudiants n'accèdent qu'à leur propre note et à leur copie, mais les enseignants peuvent voir sous une forme tabulaire les notes de tous les étudiants (ainsi que leur moyenne...), et peuvent cliquer sur un étudiant pour accéder à sa copie.

PHASE ANALYSE (8 points)

Question 1 (2,5 points)

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

Question 2 (3 points)

Proposez un diagramme de classes d'analyse pour ce problème.
On ne détaillera pas les opérations de la classe (fictive) représentant le système.

Question 3 (2,5 points)

Réaliser la fiche détaillée du ou des cas d'utilisation(s) permettant à un étudiant de répondre à un questionnaire.

PHASE CONCEPTION (10 points)

Question 4 (1,5 points)

On souhaite découper l'application en trois composants aux responsabilités distinctes :
Le composant **Questionnaires** est chargé de stocker les questions et questionnaires définis par les enseignants. Le composant **Epreuves** est chargé de gérer la création des épreuves et leur bon déroulement (donc de gérer le comportement décrit en question 3): affichage des questions, collecte des réponses, contrôle des dates de début/fin...A la fin de l'épreuve, **Epreuves** transmet les réponses des étudiants au composant de Notation. Enfin le composant **Notation** prend en charge la partie stockage des réponses proposées par les étudiants, ainsi que la gestion du calcul des notes et l'affichage des copies.

Quelles dépendances entre composants sont introduites par cette architecture ? Représentez les composants sur un diagramme de composants, avec leurs interfaces offertes et requises.
On ne demande pas encore à cette question de détailler les opérations de ces interfaces.

Question 5 (3 points)

Représentez par un diagramme de séquence de niveau intégration (où les lignes de vie représentent des composants) les interactions permettant au composant **Notation** de calculer la note d'une copie. On modélisera la situation où l'on souhaite obtenir la note de l'étudiant « e007 » pour l'épreuve « QCM-124 ». Cet étudiant a coché pour la question 1 les réponses **a** et **b** (correct, **a** et **b** attendus !) et pour la question 2 la réponse **a** (faux car incomplet, on attendait **a** et **c**). Le QCM ne comportait que deux questions. On modélisera explicitement sur le diagramme les valeurs des paramètres sur les messages.

Fournissez par déduction la signature des opérations des interfaces définies en question 4 que vous avez utilisées dans cette question.

Question 6 (3 points)

Proposez à l'aide d'un diagramme de classes une réalisation possible du composant **Notation**.
On fera attention à la cohérence avec les réponses aux questions 4 et 5.

Question 7(2,5 points)

On souhaite s'assurer du fait que le barème positionné par l'enseignant influe bien sur la note. De quel niveau de test s'agit-il ? Modélisez une configuration du système permettant ce test (diagramme de structure interne niveau système). Vous décrirez le rôle des instances qui apparaissent sur ce diagramme.