

# Rapport de Génie Logiciel 2

MONGODIN Maxime; HIRACLIDES Nicolas;  
LESIEUR Christophe; KOWALEWSKI Matthieu

6 mars 2014

# Table des matières

<b>Introduction</b>	<b>2</b>
<b>1 Présentation du projet</b>	<b>3</b>
1.1 Expression des besoins . . . . .	3
1.2 Contrainte Plateforme . . . . .	3
1.3 Ressources Humaines . . . . .	4
1.4 Planning . . . . .	4
<b>2 Analyse</b>	<b>7</b>
2.1 Cas d'utilisation . . . . .	7
2.1.1 Diagramme Général . . . . .	7
2.1.2 Diagramme de cas d'utilisation pour les administrateurs .	8
2.1.3 Diagramme de cas d'utilisation pour les professeurs . . . .	10
2.1.4 Diagramme de cas d'utilisation pour les étudiants . . . .	13
2.2 Diagrammes de classe . . . . .	14
2.2.1 Diagramme de classe général . . . . .	14
2.2.2 Diagramme de classe relatif aux administrateurs . . . . .	15
2.2.3 Diagramme de classe relatif aux professeurs . . . . .	16
2.2.4 Diagramme de classe relatif aux étudiants . . . . .	17
<b>Conclusion</b>	<b>18</b>

# Introduction

Dans le cadre de l'école virtuelle de L'EISTI nommée AREL, l'école a besoin de gérer un certain nombre de questionnaires. L'objectif du premier projet génie logiciel était d'établir l'ensemble de la base de données permettant de gérer les données générées par ces questionnaires.

Dans ce deuxième projet génie logiciel, nous devons concevoir un logiciel permettant de gérer et de traiter des QCM. En d'autres termes, nous devons développer une interface capable de créer des QCM, d'y répondre et de stocker et exploiter ces données. Il nous faudra pour cela planifier le projet dans son intégralité en respectant le temps qui nous a été imparti.

# Chapitre 1

## Présentation du projet

### 1.1 Expression des besoins

Notre logiciel doit pouvoir gérer trois types d'utilisateurs : enseignants, élèves et administrateurs.

Les administrateurs ont comme fonction de :

- définir d'autres utilisateurs quel que soit leur rôle. Pour un professeur ils préciseront le domaine d'enseignement.
- définir une promotion (liste d'élèves).
- définir et modifier les modules enseignés à l'école en prenant en compte le nom du module, les modules prérequis et le syllabus du module.

Les professeurs doivent pouvoir :

- définir un QCM qui comprend des questions, des réponses prédéfinies (au moins deux par question) vraies ou fausses.
- chaque QCM peut être privé ou public. C'est-à-dire qu'un QCM privé n'est utilisable que par le professeur qui l'a créé. Le rendre public, le rend utilisable par tous les professeurs.
- créer une session de QCM avec dates de début et de fin, associée à un module et une promotion. On peut répondre à un QCM plusieurs fois seulement si cela est précisé par le professeur.
- consulter les résultats complets ou partiels des QCM à tout moment soit par élève, soit sous forme de statistiques (écart type, moyenne, ...)

Les élèves quant à eux peuvent :

- répondre à un QCM dans lequel il est inscrit entre les dates de début et fin du QCM et voir un message d'avertissement en dehors de ces dates.
- consulter les résultats d'un QCM auquel il a participé une fois que la session est fermée.

### 1.2 Contrainte Plateforme

Pour ce projet, nous avons comme contrainte d'implémenter le code sous Java version 1.7. Nous devons aussi utiliser comme méthode d'analyse et de conception l'UML avec la persistance des données. Nous utiliserons StarUML afin de réaliser les différents diagrammes. Les objets doivent communiquer entre

eux à travers des interfaces. la sauvegarde des données à la fin du programme ou sur commande pour les données modifiées uniquement est obligatoire.

### 1.3 Ressources Humaines

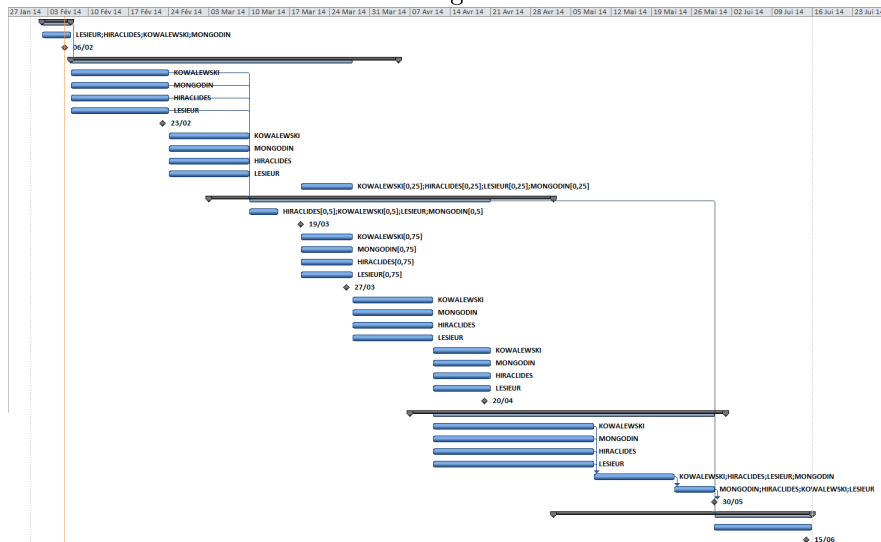
Notre équipe se compose de quatre membres qui sont les suivants :

- MONGODIN Maxime
  - Etudiant à l'EISTI de Cergy
  - connaissance informatique : C, C++, langage web, Ocaml, Pascal, Java.
- KOWALEWSKI Matthieu
  - Etudiant à l'EISTI de Cergy
  - connaissance informatique : C, langage web, Ocaml, Pascal, Java.
- HIRACLIDES Nicolas
  - Etudiant à l'EISTI de Cergy
  - connaissance informatique : C, langage web, Ocaml, Pascal, Java.
- LESIEUR Christophe
  - Etudiant à l'EISTI de Cergy
  - connaissance informatique : C, langage web, Ocaml, Pascal, Java.

### 1.4 Planning

Nous avons décidé d'attribuer à chaque membre de l'équipe une partie du projet qu'il devrait analyser, conceptualiser et implémenter du début à la fin du projet. Les découpages actuelles des tâches vont sans doute évoluer en fonction du temps et de l'avancement du projet, ainsi que pour respecter l'équité de la quantité de travail à fournir par chaque membre. Voici donc notre diagramme de Gantt :

FIGURE 1.1 – Diagramme de Gantt



Les différentes parties du diagramme sont détaillées ici :

FIGURE 1.2 – Cahier Des charges

<b>Cahier des charges</b>	<b>5 jours</b>	<b>Dim 02/02/14</b>	<b>Jeu 06/02/14</b>
Cahier des charges	5 jours	Dim 02/02/14	Jeu 06/02/14
jalón 1	0 jour	Jeu 06/02/14	Jeu 06/02/14

FIGURE 1.3 – Analyse

<b>Analyse</b>	<b>57 jours</b>	<b>Ven 07/02/14</b>	<b>Ven 04/04/14</b>
Diagramme de cas d'utilisation et de classe général	17 jours	Ven 07/02/14	Dim 23/02/14
Diagramme de cas d'utilisation et de classe professeur	17 jours	Ven 07/02/14	Dim 23/02/14
Diagramme de cas d'utilisation et de classe élève	17 jours	Ven 07/02/14	Dim 23/02/14
Diagramme de cas d'utilisation et de classe administrateur	17 jours	Ven 07/02/14	Dim 23/02/14
jalón 2	0 jour	Dim 23/02/14	Dim 23/02/14
Diagramme de classe avancé général	14 jours	Lun 24/02/14	Dim 09/03/14
Diagramme de classe avancé professeur	14 jours	Lun 24/02/14	Dim 09/03/14
Diagramme de classe avancé élève	14 jours	Lun 24/02/14	Dim 09/03/14
Diagramme de classe avancé administrateur	14 jours	Lun 24/02/14	Dim 09/03/14
vérification de l'analyse	9 jours	Mer 19/03/14	Jeu 27/03/14


FIGURE 1.4 – Conception

<b>conception</b>	<b>60 jours</b>	<b>Lun 03/03/14</b>	<b>Jeu 01/05/14</b>
Préparation de soutenance	5 jours	Lun 10/03/14	Ven 14/03/14
Soutenance	0 jour	Mer 19/03/14	Mer 19/03/14
Diagramme d'état d'activité général	9 jours	Mer 19/03/14	Jeu 27/03/14
Diagramme d'état d'activité professeur	9 jours	Mer 19/03/14	Jeu 27/03/14
Diagramme d'état d'activité élève	9 jours	Mer 19/03/14	Jeu 27/03/14
Diagramme d'état d'activité administrateur	9 jours	Mer 19/03/14	Jeu 27/03/14
jalón 3	0 jour	Jeu 27/03/14	Jeu 27/03/14
Diagramme d'état de séquence général	14 jours	Ven 28/03/14	Jeu 10/04/14
Diagramme d'état de séquence professeur	14 jours	Ven 28/03/14	Jeu 10/04/14
Diagramme d'état de séquence élève	14 jours	Ven 28/03/14	Jeu 10/04/14
Diagramme d'état de séquence administrateur	14 jours	Ven 28/03/14	Jeu 10/04/14
Diagramme d'état de transition général	10 jours	Ven 11/04/14	Dim 20/04/14
Diagramme d'état de transition professeur	10 jours	Ven 11/04/14	Dim 20/04/14
Diagramme d'état de transition élève	10 jours	Ven 11/04/14	Dim 20/04/14
Diagramme d'état de transition administrateur	10 jours	Ven 11/04/14	Dim 20/04/14
jalón 4	0 jour	Dim 20/04/14	Dim 20/04/14

FIGURE 1.5 – Implémentation

<b>Implémentation</b>	<b>55 jours</b>	<b>Lun 07/04/14</b>	<b>Sam 31/05/14</b>
Implémentation générale	28 jours	Ven 11/04/14	Jeu 08/05/14
Implémentation professeur	28 jours	Ven 11/04/14	Jeu 08/05/14
Implémentation élève	28 jours	Ven 11/04/14	Jeu 08/05/14
Implémentation administrateur	28 jours	Ven 11/04/14	Jeu 08/05/14
Mise en commun des implémentations	14 jours	Ven 09/05/14	Jeu 22/05/14
Batterie de test	7 jours	Ven 23/05/14	Jeu 29/05/14
jalón 5	0 jour	Ven 30/05/14	Ven 30/05/14

FIGURE 1.6 – Synthèse

 <b>synthèse</b>	<b>45 jours</b>	<b>Ven 02/05/14</b>	<b>Dim 15/06/14</b>
Version livrable du programme	17 jours	Ven 30/05/14	Dim 15/06/14
jalón 6	0 jour	Dim 15/06/14	Dim 15/06/14

## Chapitre 2

# Analyse

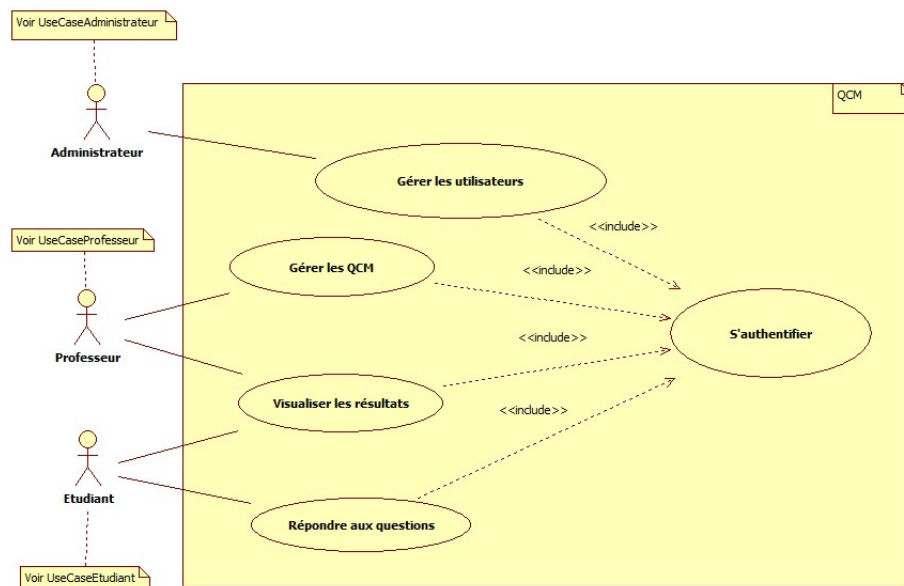
### 2.1 Cas d'utilisation

Nous avons modélisé le projet sous forme de différents diagrammes de cas d'utilisation afin de pouvoir définir l'ensemble des différents besoins que nous devons réaliser pour ce projet.

#### 2.1.1 Diagramme Général

Le premier diagramme est un diagramme plutôt général, il est peu détailler mais reprend globalement les différentes fonctionnalités demandées.

FIGURE 2.1 – Diagramme Général





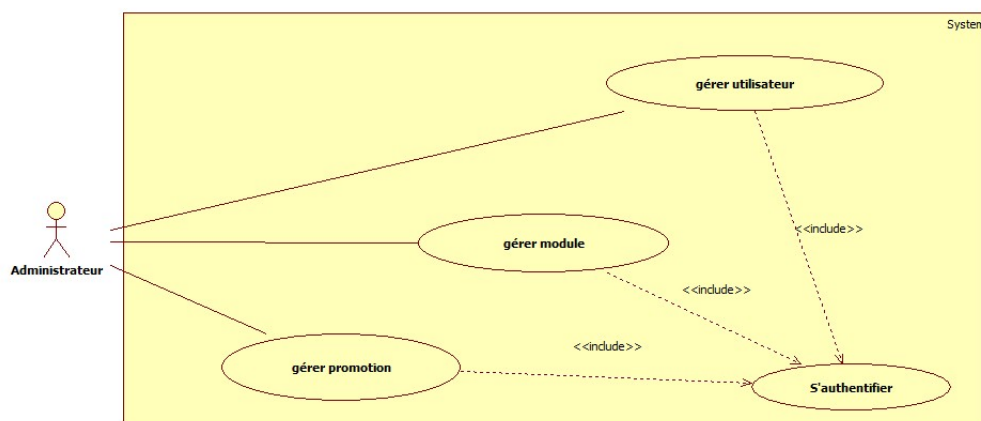
Fiche de descriptive : s'authentifier	
<b>Description</b>	Tous les utilisateurs doivent se connecter avant de réaliser les différentes actions qui leur sont attribuées
<b>Règle d'initiation</b>	Les utilisateurs doivent déjà avoir un compte d'authentification créé par un administrateur
<b>Règle de terminaison</b>	L'authentification se déconnecte à la fin de l'utilisation du logiciel ou à la déconnexion de ce dernier.
<b>Règle d'exception</b>	Aucune
<b>Relation</b>	Il s'agit d'un prérequis pour accéder à n'importe quelles autres fonctionnalités.

Les différents cas d'utilisations présentés sur le diagramme sont expliqués plus en détails par la suite.

### 2.1.2 Diagramme de cas d'utilisation pour les administrateurs

Ce diagramme est en rapport avec les administrateurs.

FIGURE 2.2 – Diagramme Administrateur



Fiche de descriptive : Gérer utilisateurs	
<b>Description</b>	L'administrateur doit pouvoir, via une authentification, créer un utilisateur
<b>Règle d'initiation</b>	L'administrateur doit avoir son nom d'utilisateur et son mot de passe valide pour s'authentifier. L'utilisateur créé ne doit pas déjà exister.
<b>Règle de terminaison</b>	Création d'un utilisateur dans la base de données.
<b>Règle d'exception</b>	Aucune
<b>Relation</b>	Ce cas inclus le cas s'authentifier Ce cas peut entraîner la précision du domaine d'enseignement dans le cas où un professeur est défini

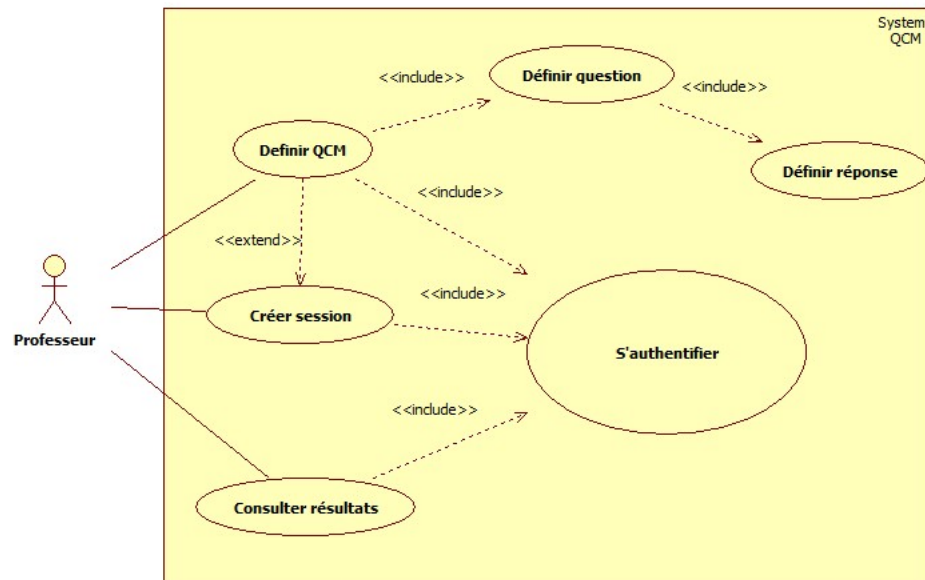
<b>Fiche de descriptive : Gérer module</b>	
<b>Description</b>	L'administrateur doit pouvoir, via une authentification, créer un module
<b>Règle d'initiation</b>	Le module créé ne doit pas déjà exister. Le module désigné comme prérequis doit exister. Pour modifier un module, ce dernier doit exister.
<b>Règle de terminaison</b>	Création d'un module dans la base de données.
<b>Règle d'exception</b>	Aucune
<b>Relation</b>	Ce cas inclus le cas s'authentifier Ce cas peut entraîner la précision des modules prérequis

<b>Fiche de descriptive : Gérer promotion</b>	
<b>Description</b>	L'administrateur doit pouvoir, via une authentification, créer une promotion
<b>Règle d'initiation</b>	La promotion créée ne doit pas déjà exister. Les élèves d'une promotion doivent exister.
<b>Règle de terminaison</b>	Création d'une promotion dans la base de données.
<b>Règle d'exception</b>	Aucune
<b>Relation</b>	Ce cas inclus le cas s'authentifier

### 2.1.3 Diagramme de cas d'utilisation pour les professeurs

Ce diagramme est en rapport avec les professeurs.

FIGURE 2.3 – Diagramme Professeur



<b>Fiche de descriptive : Définir QCM</b>	
<b>Description</b>	Création d'un ensemble de questions, elles-mêmes munies de plusieurs réponses fermées (qualitative nominale), et qui sont définies comme vraies ou fausses. Lors de la création du QCM, le professeur précise s'il est privé (seul le créateur peut l'utiliser) ou public (d'autres professeurs peuvent l'utiliser). Un libellé doit aussi être donné pour identifier ledit QCM.
<b>Règle d'initiation</b>	Le professeur est authentifié et choisit de créer un QCM. Une QCM possédant le même libellé ne doit pas déjà exister. Une QCM doit être composé d'au moins 1 question. Une question doit avoir au moins 2 réponses. Une réponse est soit vraie, soit fausse.
<b>Règle de terminaison</b>	Un objet QCM est créé. Le libellé est disponible dans la liste des QCMs disponible lors d'une session. Invite de création de session pour le QCM créé.
<b>Règle d'exception</b>	Aucune
<b>Relation</b>	Le cas d'usage 'S'authentifier' est inclus dans le cas d'usage 'Définir QCM' car il permet au professeur de s'authentifier. Les cas d'usage 'Définir question' et, par extension, 'Définir réponse', sont inclus dans 'Définir QCM' car ils sont des étapes nécessaires à la création d'un QCM.

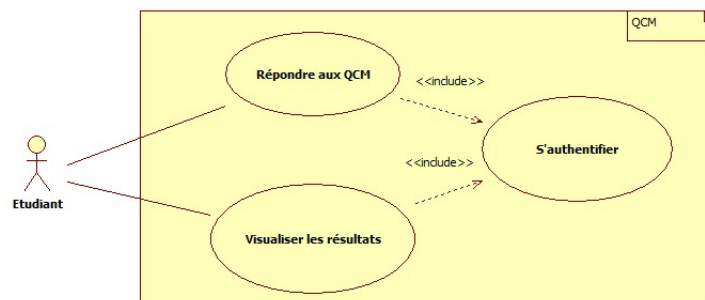
<b>Fiche de descriptive : Créer Session</b>	
<b>Description</b>	Création d'une session de QCM donné, limitée dans le temps par une date de début et une date de fin, toute deux précisées par le professeur. Cette session est associée à une promotion, pour savoir quels élèves doivent passer le QCM ; un module, pour savoir dans le cadre de quel enseignement ces élèves sont interrogés. Un nombre de répétitions (par défaut, 1) est aussi indiqué, pour donner le nombre maximum de fois où un élève donné a le droit de répondre au QCM.
<b>Règle d'initiation</b>	Le professeur doit être authentifié. Le QCM indiqué doit exister. La promotion doit exister et être non vide. Le module doit exister. La date de début doit être avant la date de fin. Un nombre de répétitions est attendu.
<b>Règle de terminaison</b>	Création d'un objet session. Création d'un objet résultats, associé à la session créée, qui stockera les réponses données par chaque élève lors de cette session.
<b>Règle d'exception</b>	Si le QCM indiqué n'existe pas, alors le professeur peut choisir d'en créer un. La date de début peut être la même que celle de fin : la session dure alors une unique journée. Si le aucune nombre de répétitions n'est précisé, celui-ci est fixé par défaut à 1.
<b>Relation</b>	Le cas d'usage 'S'authentifier' est inclus dans le cas d'usage 'Séfinir session' car il permet au professeur de s'authentifier. Le cas d'usage 'Définir QCM' est une extension de 'Définir session', car il peut être appelé au cours de celui-ci.

Fiche de descriptive : Consulter les résultats	
<b>Description</b>	Un professeur doit pouvoir consulter les résultats des sessions qu'il a créées. Ceux-ci peuvent être définitifs (session terminée), ou bien partiels (session toujours en cours). Deux types de résultats peuvent alors être visualisés : pour chaque élève, ses réponses ainsi que son score final, ou bien les statistiques de l'ensemble des élèves, à savoir la moyenne et l'écart-type de tous les scores, et la fréquence de bonnes réponses par question.
<b>Règle d'initiation</b>	Le professeur doit être authentifié. Le professeur voulant consulter les résultats d'une session doit être le créateur de cette session. La session doit avoir commencé (des résultats doivent être enregistrés pour être visualisés).
<b>Règle de terminaison</b>	Aucune
<b>Règle d'exception</b>	Si la session n'a pas encore commencé, un message affiche cet état de fait et invite le professeur à retenter le jour de début de session.
<b>Relation</b>	Le cas d'usage 'S'authentifier' est inclus dans le cas d'usage 'Consulter résultats' car il permet au professeur de s'authentifier.

#### 2.1.4 Diagramme de cas d'utilisation pour les étudiants

Ce diagramme est en rapport avec les étudiants.

FIGURE 2.4 – Diagramme Etudiant



<b>Fiche de descriptive : Répondre aux QCM</b>	
<b>Description</b>	L'étudiant peut, via une authentification, répondre à des QCM.
<b>Règle d'initiation</b>	L'étudiant doit être dans la promotion qui répond à ce QCM et il doit se connecter durant la session.
<b>Règle de terminaison</b>	Une fois que l'étudiant a répondu à un QCM, on doit noter l'étudiant comme ayant validé ce QCM, récupérer les réponses données et les stocker.
<b>Règle d'exception</b>	Affichage d'un message d'erreur si l'étudiant essaye de répondre aux QCM en dehors des dates de début et de fin du QCM
<b>Relation</b>	Répondre aux QCM nécessite de s'authentifier

<b>Fiche de descriptive : Visualiser les résultats</b>	
<b>Description</b>	L'étudiant peut, via une authentification, visualiser ses résultats.
<b>Règle d'initiation</b>	L'étudiant doit avoir répondu aux QCM, doit se connecter une fois la session terminée, et il ne doit pas pouvoir visualiser les QCM auquel il n'a pas répondu.
<b>Règle de terminaison</b>	Aucune
<b>Règle d'exception</b>	Aucune
<b>Relation</b>	Visualiser les résultats nécessite de s'authentifier

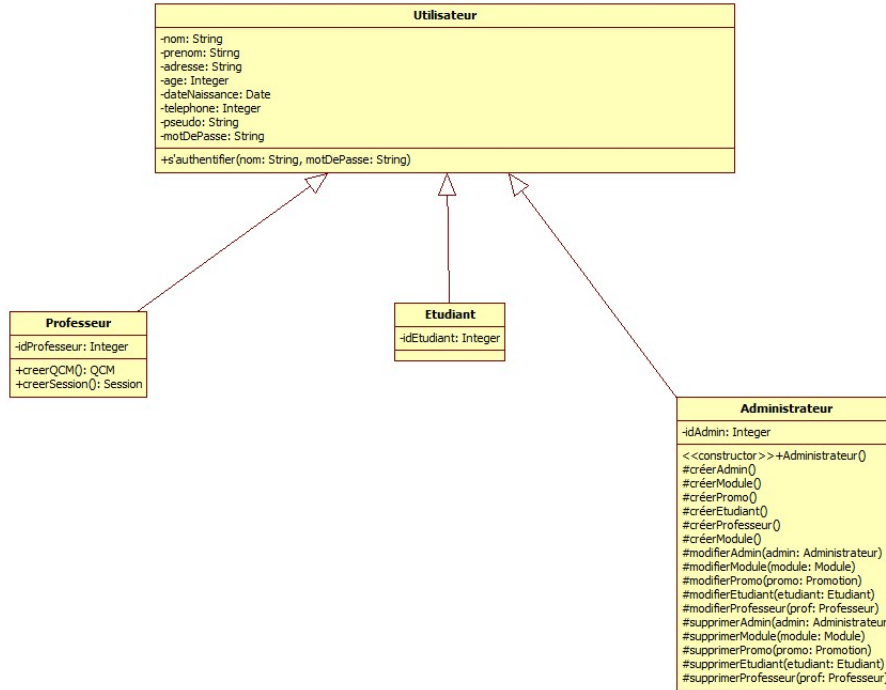
## 2.2 Diagrammes de classe

Le diagramme de classes est un schéma utilisé en génie logiciel pour présenter les classes et les interfaces des systèmes ainsi que les différentes relations entre celles-ci. Ce diagramme fait partie de la partie statique d'UML car il fait abstraction des aspects temporels et dynamiques.

### 2.2.1 Diagramme de classe général

Dans un premier temps nous nous intéressons à la super classe Utilisateur dont un certain nombre de propriétés va engendrer des attributs pour les différents utilisateurs du logiciel. Chacun des utilisateurs est détaillé dans la suite du rapport.

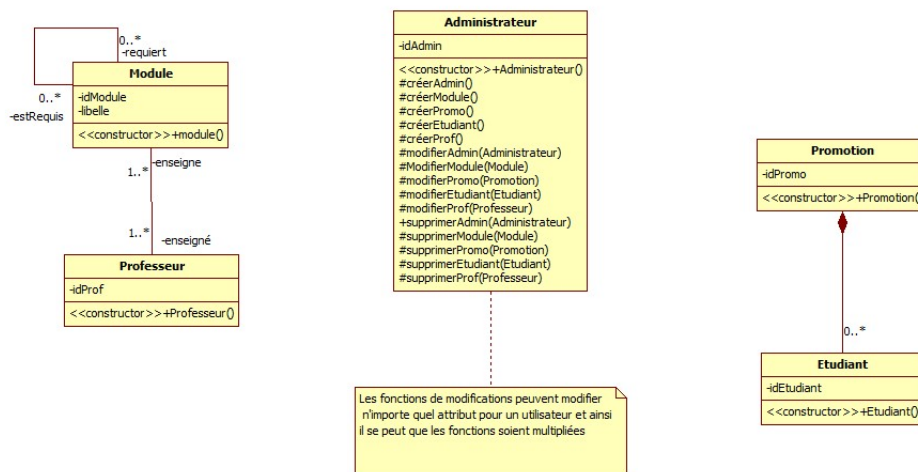
FIGURE 2.5 – Diagramme Administrateur



## 2.2.2 Diagramme de classe relatif aux administrateurs

Les administrateurs peuvent créer, modifier ou même supprimer un certain nombre d'éléments dans le logiciel. Un premier administrateur devra être ajouté manuellement afin de pouvoir générer un ensemble d'utilisateurs et de QCM.

FIGURE 2.6 – Diagramme Administrateur

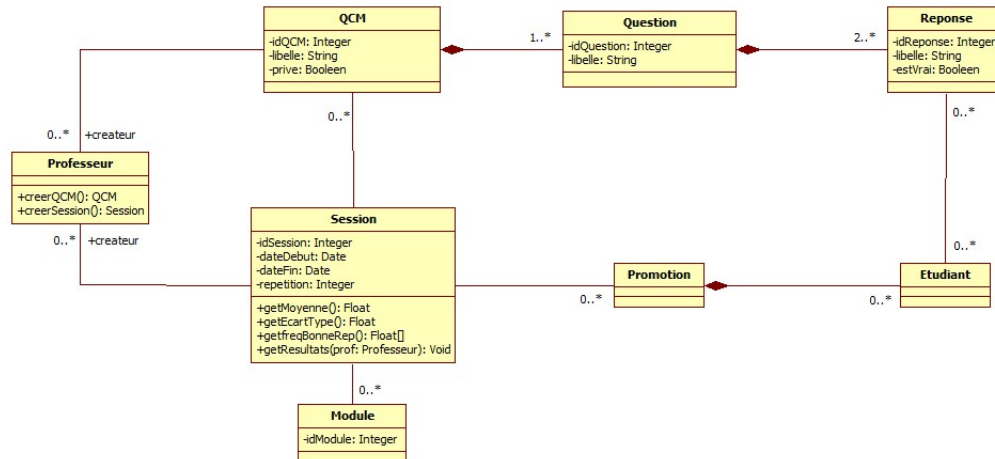




### 2.2.3 Diagramme de classe relatif aux professeurs

Le professeur peut créer un QCM ainsi que la session associée à ce QCM. On y retrouve ici un lien avec l'étudiant qui est détaillé dans le prochain diagramme.

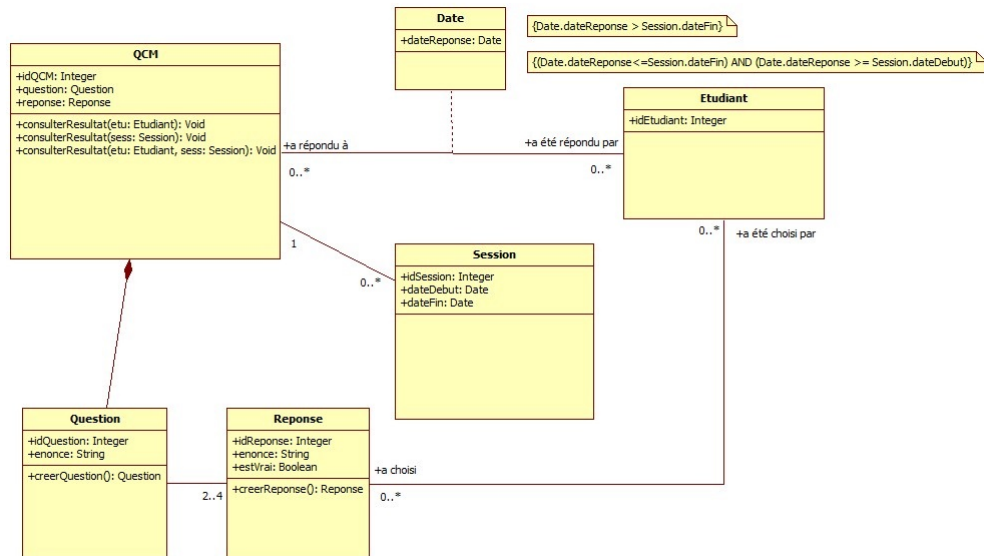
FIGURE 2.7 – Diagramme Professeur



## 2.2.4 Diagramme de classe relatif aux étudiants

L'étudiant va pouvoir répondre aux questions d'un ou plusieurs QCM dans la limite où la session est encore active. Si la session est terminée l'étudiant peut alors visualiser ses résultats.

FIGURE 2.8 – Diagramme Etudiant



# Conclusion

Pour conclure, ce livrable contient une partie de l'analyse du projet. Il va être modifier dans les semaines qui suivent pour être beaucoup plus précis et ainsi mieux cerner le problème initial. Les diagramme proposées ne sont pas définitif ils pourront être modifiés tout au long des prochains livrables.