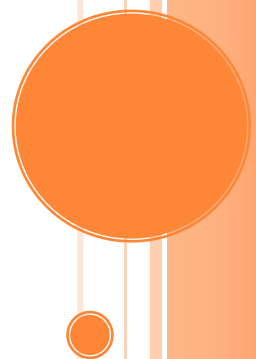


CAHIER DES CHARGES

MOOC'Line

Plateforme d'apprentissage et auto-évaluation

Par : Amadou Beidari, Seydou Diallo, Amadou Doumbia,
Ahmed El sayed, NguyenThi Ngoc Tam, Gayllord Yapou
28/02/2014



Cahier des charges

Plateforme d'apprentissage et auto-évaluation

<u>introduction</u>	2
But du projet :.....	2
<u>présentation générale</u>	3
Contexte.....	3
Produit et marché.....	3
<u>expression du besoins</u>	4
<i>Inscription</i> :	4
<i>Authentification</i> :	4
<i>Cours</i> :	4
<i>Recherche d'un cours</i> :	5
<i>Devoirs</i> :	5
<i>Evaluation</i> :	6
<i>Interface étudiant</i> :	6
<i>Interfaces professeurs</i> :	7
<u>besoin du produit</u>	8
<u>interface utilisateur</u>	9
<i>Inscription</i>	9
<u>description du système</u>	11
<i>Schéma fonctionnel de l'application</i> :	11
<i>Diagramme de classe</i> :	12
<i>La base de données</i> :	13
<u>scenario d'utilisation</u>	14
<i>Diagramme de séquence représentant la recherche d'un cours</i> :	16
<u>besoin interne</u>	17
<i>Environnement de développement</i> :	17
<i>Langage de développement</i> :	17
<i>Livraison</i> :	17
<u>annexes</u>	18

INTRODUCTION

Le projet que nous allons réaliser est une plateforme de cours en ligne pour apprentissage et auto-évaluation. La conception-réalisation de celui-ci nous place dans la situation d'un projet informatique reprenant le cadre de l'entreprise.

Dans ce document nous allons vous présenter les besoins du client, les spécifications du système ainsi que l'implémentation du projet et pour finir une partie test du logiciel.

But du projet :

Dans ce projet, l'objectif est de créer une plateforme de cours en ligne permettant aux professeurs d'universités de partager leurs cours.

Il vise un assez large public ainsi que plusieurs universités. Il doit permettre aux utilisateurs de suivre les cours auxquels ils se seront préalablement inscrits et d'accéder aux exercices concernant ces derniers, dans le but de valider des connaissances sur les sujets abordés.

Ce projet peut viser des personnes qui n'ont pas forcément le temps ou les moyens de suivre les cours des universités participantes; ainsi que des personnes qui ont tout simplement envie d'enrichir leurs connaissances personnelles.

Il devra être déployé sur une plateforme accessible à tous pour répondre aux besoins spécifiés tel que le stockage de cours, la mise en place d'exercice et d'auto-évaluation.

Un certain nombre de fonctionnalités devra être proposé afin de simplifier au mieux l'utilisation de la plateforme les professeurs.

PRESENTATION GENERALE

La plateforme de cours en ligne est interactif est ouverte à tous. L'utilisateur s'inscrit sur la plateforme. Après une authentification l'utilisateur peut choisir de s'inscrire à un cours en fonction des thèmes ou universités qu'il souhaite. L'un des principes de cette plateforme est de pouvoir valider des connaissances acquises dans un cours.

Contexte

Le contexte premier de la plateforme sera exclusivement éducatif, le but étant de permettre à un maximum de personnes d'avoir la possibilité de s'instruire malgré les contraintes qui s'opposaient à eux.

Ce projet est à l'image du réseau « FUN - France Université Numérique », la plateforme française de cours en ligne, ouverte depuis le 16 janvier 2014. Plusieurs cours numériques ont été ouverts pour les 100 000 inscrits de la plateforme. Des plateformes similaires sont disponibles aux Etats Unis depuis 2010 et affichent un nombre d'inscrits de plus de 6 millions.

Produit et marché

Plusieurs plateformes dans le genre sont déjà en ligne, mais l'une des fonctionnalités demandées par rapport à celles de base concernerait la correction automatique qui pourrait s'appliquer à des types d'exercices autres que de simples QCM, par exemple la correction de programme informatique.

EXPRESSION DU BESOINS

Inscription :

La plateforme gère deux types d'inscriptions. Il y a une inscription pour l'utilisateur qui veut accéder aux services de la plateforme. Celle-ci se fait avec un identifiant unique (son mail), son nom et un mot de passe.

Et une inscription pour un organisme qui souhaite intégrer la plateforme. Pour cela la plateforme propose un formulaire d'inscription en ligne.

Si l'inscription est acceptée, l'université pourra ajouter sa liste d'enseignants désirant mettre leurs cours en ligne.

Authentification :

Chaque utilisateur, qu'il soit étudiant, enseignant ou un organisme (ici représenté par un administrateur) doit s'authentifier pour pouvoir accéder aux services proposés selon l'utilisateur. Si l'authentification échoue on doit proposer à l'utilisateur de lui renvoyer son mot de passe par mail ou de lui donner la possibilité de s'inscrire s'il ne l'est pas encore.

Cours :

La plateforme propose un grand nombre de cours à catégorie variée tel que l'Art, l'informatique, la chimie...

Un cours est créé par un enseignant d'une école inscrite à la plateforme. Il est réparti en plusieurs parties, chaque partie comporte des chapitres. Ce chapitre est un document support. Le document devra être stocké sur la plateforme. Les formats du document sont :

- Format texte : pdf
- Format image : jpeg,png,bmp,gif
- Format vidéo : mpeg-4, avi,

Le document peut être créé sur la plateforme si le format est un format texte.

Le document peut être soit visionné sur la plateforme soit téléchargé. Un chapitre prend en général une à deux heures de cours. Une semaine de cours peut comprendre plusieurs chapitres afin d'atteindre un volume de 6 heures de travail personnel par cours et par semaine.

Chaque cours à une page spécifique sur la plateforme. La plateforme doit pouvoir laisser le choix au professeur du cours sur le contenu de la page du cours.

Recherche d'un cours :

Un étudiant peut procéder à la recherche d'un cours. Pour cela la plateforme propose une barre de recherche pour les cours mis en ligne.

La recherche peut être faite par nom de cours, par catégorie de cours ou encore par organisme.

Devoirs :

A chaque chapitre du cours sont associées des feuilles d'exercices. Il sont aussi créés par l'enseignant responsable du cours sur la plateforme. L'étudiant doit effectuer ces exercices et les rendre sur la plateforme avec une date limite de dépôt. Si les exercices ne sont pas rendus à temps, l'enseignant considérera que les exercices ne sont pas rendus.

A chaque partie de cours on associe des feuilles de révision. Elles prennent la forme d'un rendu de mémoire, d'un projet, elles sont aussi à déposer sur la plateforme par l'étudiant avec une date limite.

Et pour finir il y a l'examen final du cours. Il est constitué d'un ensemble d'exercices qui doivent être rendus avant la fin de la période d'enseignement du cours.

La plateforme doit supportés plusieurs formes d'exercices :

- Des questions à choix multiple (QCM)
- Des questions à réponse formatée (QRF) où la réponse doit être donnée dans un format bien précis qui sera indiqué dans la question.
- Des questions dont la réponse est un programme (P) dont les entrées et les sorties doivent respecter un format fixé par l'exercice.
- Des questions dont la réponse est libre (L) mais limitée par un nombre maximum de mots.

La plateforme permet de faire une sauvegarde des feuilles d'exercices et de faire une édition des feuilles non terminées.

Evaluation :

Pour chaque exercice, l'enseignant associe sa correction. Chaque rendu d'exercices est validé par un nombre de points, qui donnera une note finale à l'étudiant. La plateforme doit proposer plusieurs types de corrections selon le format de l'exercice:

- Une correction automatique est appliquée pour les exercices de type QCM, QRF et P.
- Une correction par l'enseignant pour les exercices à réponse libre (L) et parfois pour des exercices de type (P) pour évaluer le style de programmation.
- Une correction par pairs qui consiste à faire corriger une réponse par un autre étudiant inscrit au cours.

Le type de correction est indiqué par l'enseignant lors de sa création. Il indique aussi le nombre maximum de soumissions permise par l'étudiant avant la date limite. Le dernier score obtenu est conservé. La note est affichée immédiatement après la soumission quand la correction est automatique.

Lors de la création de la correction des feuilles des exercices, le professeur doit préciser le barème. Par exemple :

- Pour un exercice QCM, chaque choix de réponse doit avoir un nombre de points.
- Pour un exercice QRF, la réponse attendue doit être indiquée.
- Pour un exercice P, le professeur doit fournir un ensemble de fichiers incluant un fichier de type makefile avec deux entrées pour la compilation du programme et son test ; les autres fichiers peuvent être des bibliothèques ou des fichiers utilisés pour la compilation et le test.
- Pour un exercice L, le type de correcteur doit être indiqué.

Interface étudiant :

L'étudiant peut suivre plusieurs cours en même temps. Pour qu'il puisse suivre un cours il doit s'y inscrire. Un cours peut être suivi par session avec une date de début et une date de fin, ou de façon libre.

A sa connexion sur la plateforme, un tableau de bord est affiché avec chaque cours qu'il suit, les devoirs qu'il doit accomplir ainsi que la date limite de dépôt.

Pour les devoirs, la plateforme permet à l'étudiant de pouvoir sauvegarder ces devoirs et de les continuer si besoin. Il peut aussi les conserver avant de les soumettre pour pouvoir les modifier.

A la soumission d'un devoir, la note est affichée si la correction est automatique. Sinon la soumission est acquittée par message courriel. L'étudiant est notifié sur son tableau de bord lorsque la correction de son devoir a été faite.

Lorsque l'étudiant accède à la page de Cours choisie, les cours (support) qu'il doit suivre, les devoirs à faire ainsi que la date limite lui sont indiqués.

Interfaces professeurs :

Le professeur est responsable de l'ajout des supports de cours (les chapitres), la création des feuilles d'exercices et sa correction. Il peut rendre visibles ou non les cours et exercices sur la plateforme pour les étudiants inscrits à son cours.

Le professeur accède à son tableau de bord après son authentification, lui indiquant pour chaque cours qu'il encadre tous les devoirs soumis (corrigés automatiquement inclus). Le professeur peut obtenir pour chaque cours des statistiques sur le taux de réussite des exercices, des devoirs et des informations sur la cohorte en généra

BESOIN DU PRODUIT

Après une étude précise des besoins du client, nous avons choisi pour ce projet de vous fournir une application web ergonomique et accessible à un grand nombre d'utilisateurs.

Utilisabilité

En anticipant le nombre possible d'utilisateurs de l'application, la qualité de l'ergonomie est un facteur essentiel dans le choix d'une plateforme par rapport à une autre. Toute information de notre application sera disponible en maximum 3 clics de souris.

Composition

Notre application est composée de 3 parties :

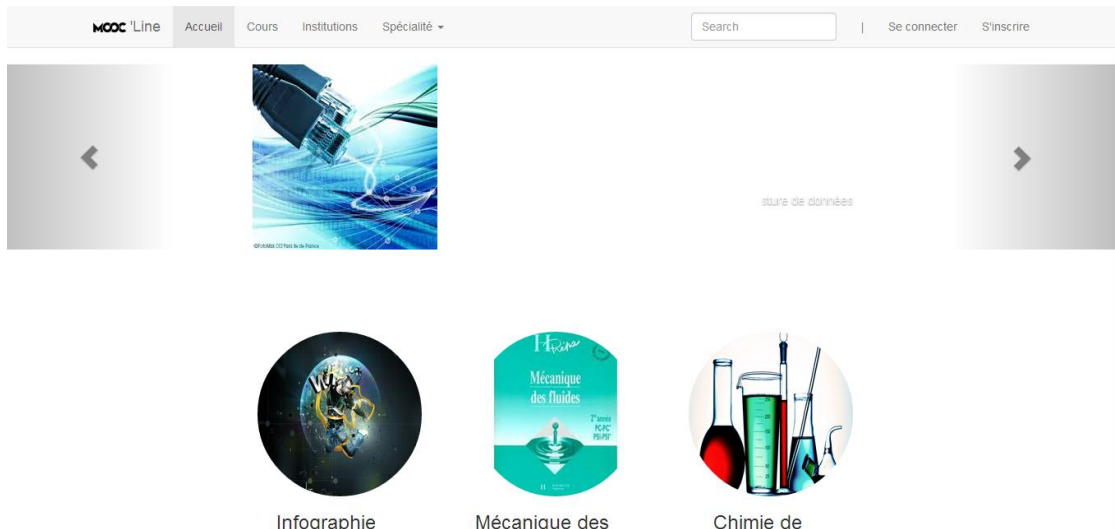
- D'une interface graphique pour l'affichage des différentes fonctionnalités du projet.
- D'une base de données pour le stockage des profil d'utilisateur et organisme, de lien des cours et des exercices.
- D'un serveur Web pour le partage de fichier et le stockage en base de données.

Portabilité

L'application devra fonctionner et avoir le même rendu graphique sur les 5 principaux navigateurs internet (Internet Explorer, Google Chrome, Safari, Firefox, Opéra).

INTERFACE UTILISATEUR

Voici la page d'accueil de notre application web. L'utilisateur peut s'inscrire, se connecter, voir les spécialités enseignées sur la plateforme. Il peut même procéder à une recherche de cours grâce à la barre de recherche dans la barre de navigation.



Inscription

Pour l'inscription d'un étudiant l'interface propose un formulaire d'inscription en demandant son mail, son nom, son pseudo, son mot de passe.

Pour l'inscription d'un organisme, la plateforme proposera un formulaire de contact en indiquant une demande d'inscription sur la plateforme.

MOOC Line

Accueil

Cours

Institutions

Spécialité ▾

Search

Se connecter

S'inscrire

Nous Contacter

Email

veuillez entrer votre Email

Nom & Prenom

veuillez entrer votre nom et prénom

Téléphone

veuillez entrer votre téléphone

Vous êtes

- Non - ▾

Sujet

- Select - ▾

Autres

Taille Organisation

nombre approximatif accepté

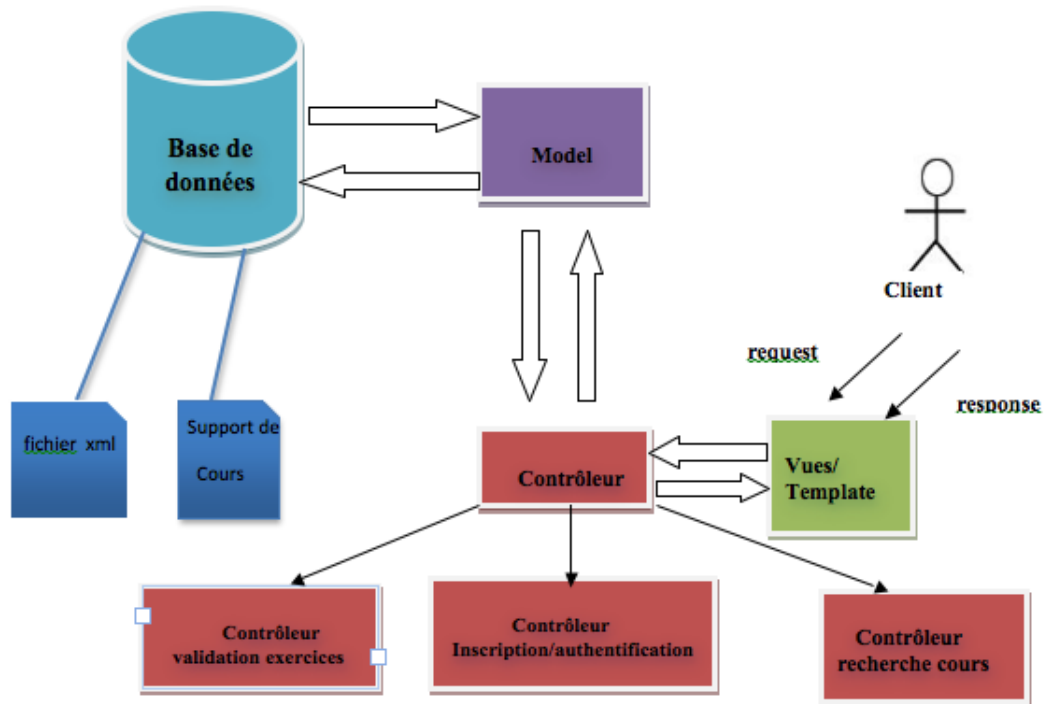
Message

Informations supplémentaires

Valider ➤

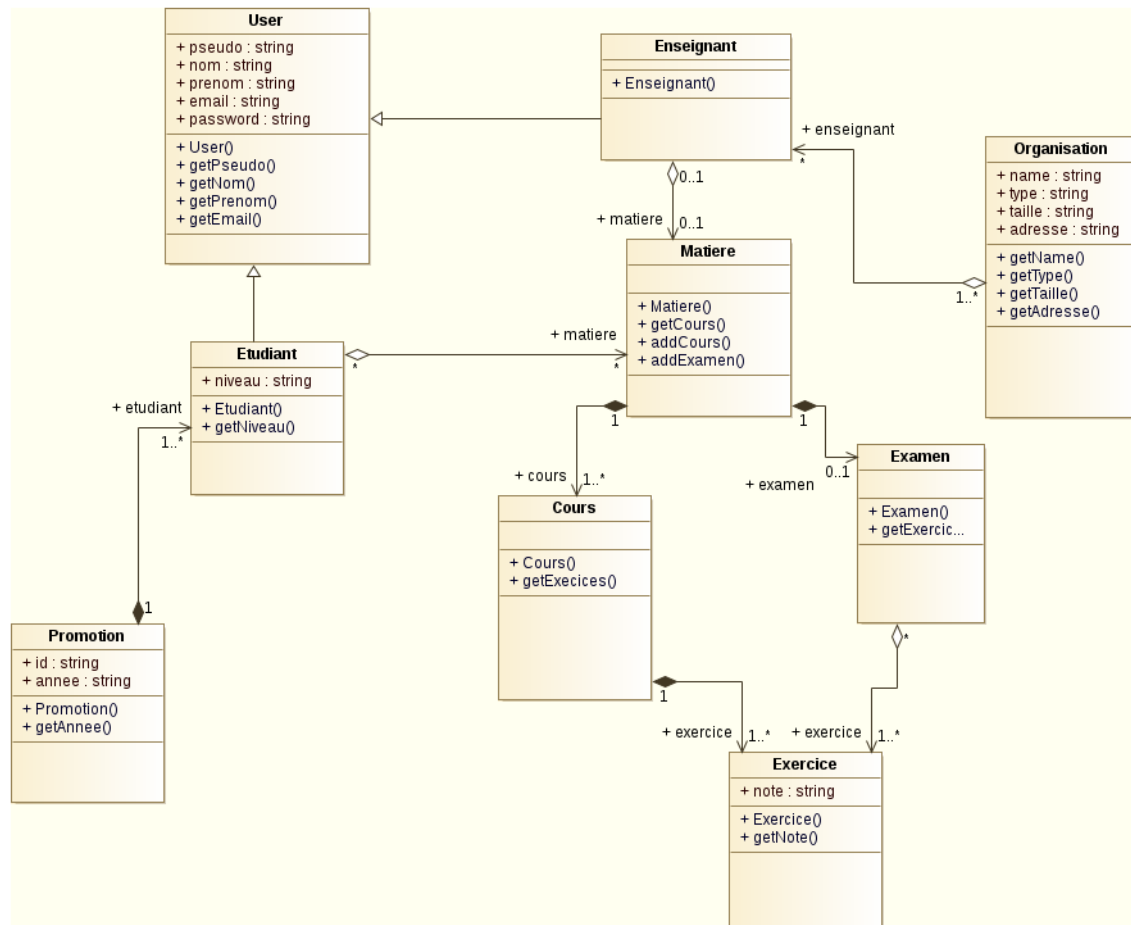
DESCRIPTION DU SYSTEME

Schéma fonctionnel de l'application :

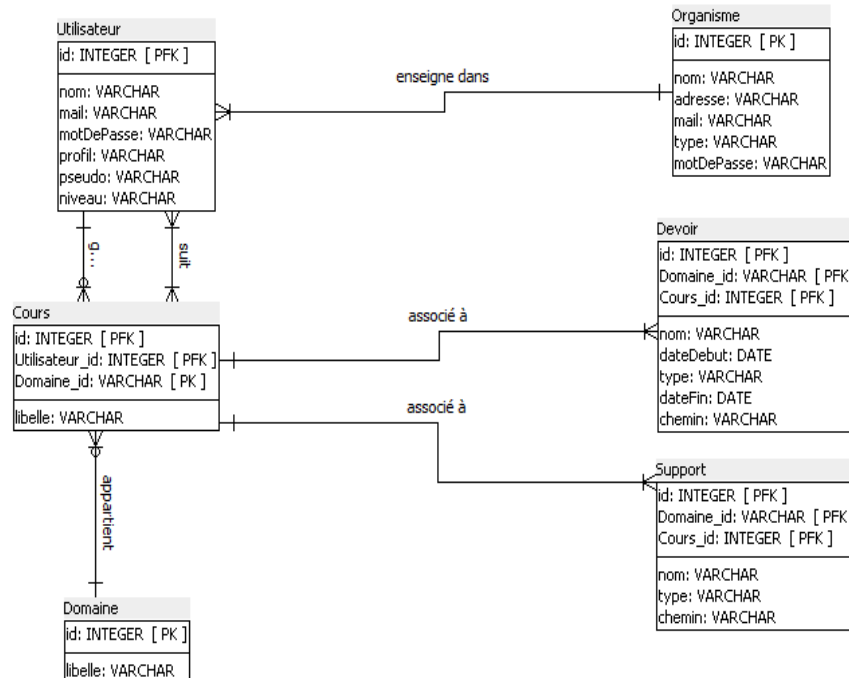


Nous avons choisi un patron MVC(Model-View-Controller).

Diagramme de classe:



La base de données:



La base de données est composée de 6 tables :

- La table Utilisateur regroupe tous les inscrits de la plateforme en tant qu'étudiant ou enseignant.
- La table Organisme regroupe tout les organisme tel que les universités, les écoles d'ingénieur les organismes gouvernementaux qui participe à la mise en place de cours sur la plateforme.
- La table cours représente les cours de la plateforme.
- La table domaine qui regroupe tous les domaines de cours sur la plateforme.
- La table Support associé à la table Cours représentant le support de cours.
- La table Devoir associé aussi à la table Cours qui représente les devoirs de chaque cours.

SCENARIO D'UTILISATION

Cas 1 :

- **Cas d'utilisation** : inscription d'un utilisateur sur la plateforme

Acteur : visiteur

Parties prenantes et intérêts :

- Visiteur : il veut pouvoir s'enregistrer sur la plateforme pour pouvoir suivre des cours.

Niveau : Objectif utilisateur

Pré-conditions : aucune

Post-conditions : Les données communiquées par le visiteur sont accessibles par le service administratif.

Scénario nominal

1. Le visiteur demande une inscription au système
2. Le système communique un formulaire d'inscription.
3. Le visiteur s'inscrit.
4. Le système lui confirme l'enregistrement des informations.

Cas 2 :

- **Cas d'utilisation** : inscription d'un organisme sur la plateforme

Acteurs : Organisme et l'Administrateur

Parties prenantes et intérêts :

- Organisme : il veut communiquer ses coordonnées afin d'être contacté ultérieurement pour faire une demande d'enregistrement sur la plateforme.
- Service Administratif : il veut pouvoir contacter, en temps voulu, les organismes ayant faite une demande d'inscription sur la plateforme.

Niveau : Objectif utilisateur

Post-conditions : Les données communiquées par le visiteur sont accessibles par le service administratif.

Scénario nominal

1. Le visiteur demande une inscription au système
2. Le système communique un formulaire de contact.
3. Le visiteur communique ces coordonnées.
4. Le système lui confirme l'enregistrement des informations.
5. Le système envoie la demande d'inscription au service administratif.
6. Le système administratif répond à l'organisme en question en lui fournissant un lien d'inscription.
7. Le système communique un formulaire d'inscription.
8. Le visiteur communique ces informations.
9. Le système lui confirme l'enregistrement des informations.
10. L'organisme demande une authentification pour pouvoir enregistrer ces enseignants.
11. Le système communique un formulaire d'identification.
12. L'organisme s'identifie.

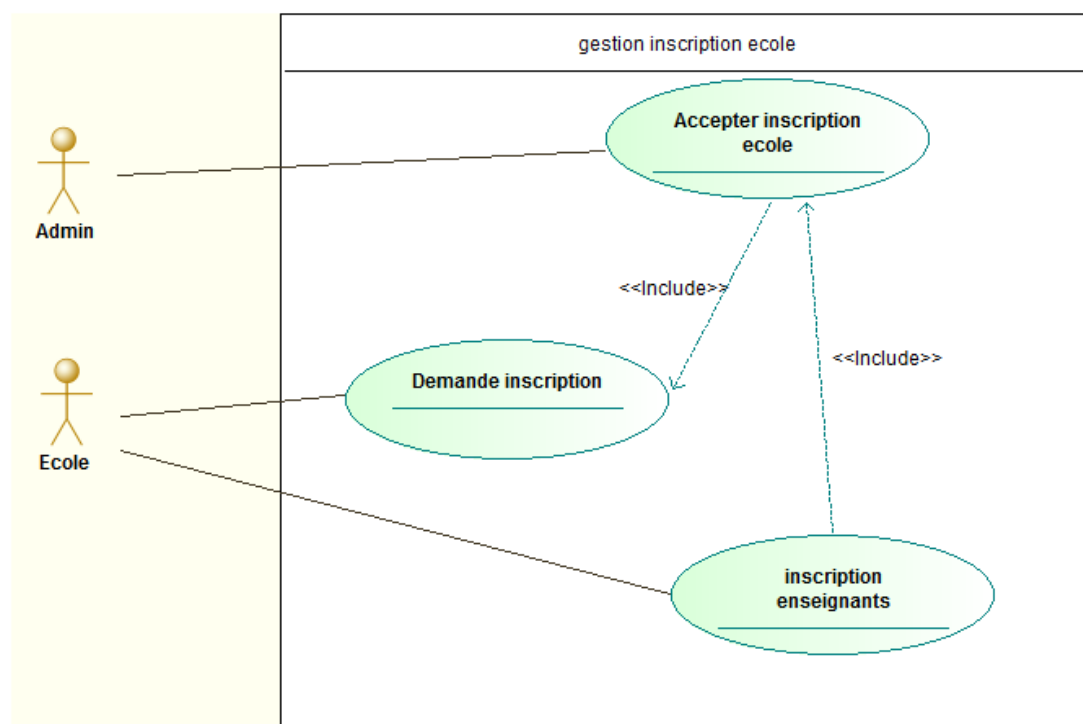
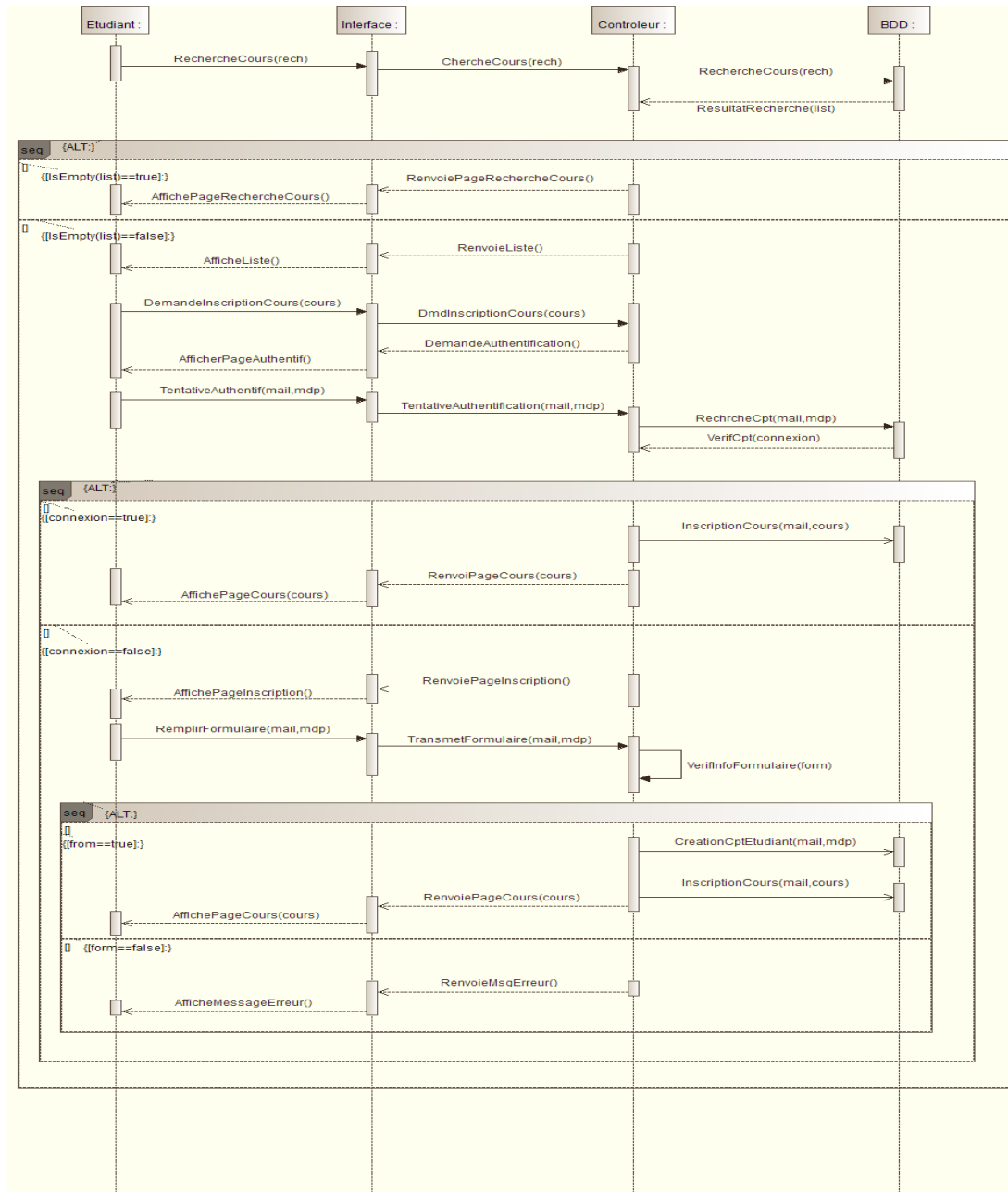


Diagramme de séquence représentant la recherche d'un cours :



BESOIN INTERNE

Environnement de développement :

- Emacs
- Notepad++
- SublimeText

Langage de développement :

- HTML et CSS pour la partie de l'interface graphique
- Php pour la partie contrôleur de l'application
- Mysql pour la base de données.

Livraison :

Une version fonctionnelle et déboguée devra être disponible pour le 6 mai 2014.

ANNEXES

Glossaire :

Environnement de développement : logiciel permettant d'écrire l'application

HTML : langage de balisage permettant la création de notre interface graphique.

CSS : feuille de style en cascade, un langage qui décrit la présentation du document html

PHP : langage de programmation

SGBD : Système de gestion de base de données destiné à stocker et à partager des informations dans une base de données.

MySQL : système de gestion de base de données.