

**République Algérienne Démocratique et Populaire**

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

---

**CENTRE UNIVERSITAIRE DE MILA**

**INSTITUT DES SCIENCES ET DE LA TECHNOLOGIE**

**Mémoire de fin d'étude**

Présenté pour l'obtention du diplôme de

**LICENCE ACADEMIQUE**

Domaine : **Mathématiques et Informatique**

Filière : **Informatique**

Spécialité : **Informatique général**

### **Thème**

**« La Conception et Le Réalisation d'un Site Web Dynamique pour la communication entre les enseignants et les étudiants en ligne » Outils utilisés : le langage UML, PHP, MySQL, JAVA Script.**

Presenté par : - Benabdelkader Samra  
- Djaafar Moufida

Sous la direction de : Hettab Abdelkamel

**Année universitaire 2010-2011**

Le Travail présenté dans ce mémoire est principalement consacré à la création d'un site Web dynamique, offrant la possibilité aux étudiants et aux enseignants pour communiquer entre eux. Ce site doit permettre de valoriser à l'échelle internationale de la recherche universitaire, pour passer à l'implémentation de notre projet on a fait une étude de cas en utilisant les diagrammes d'UML parce qu'il est adapté à toutes les phases du développement et compatible avec toutes les techniques de réalisations, et en suivant le processus de développement UP pendant toutes les phases du projet.

On a exprimé des outils techniques récents pour réaliser notre site web.

Cette expérience était très utile pour nous sur tous pour l'expérimentation des méthodes et les techniques de traitement.

Finalement, le diplôme que nous venons d'obtenir sera pour nous une ouverture sur le domaine de recherche en informatique, et une expérience afin de faire part à d'autres stagiaires de notre domaine.

***MOTS-CLÉS :***

Internet, Site web, Web.

UP :( Unified Process).

UML :( Unified Modeling language).

PHP, EasyPHP, MySQL, Adobe Dreamweaver, HTML.

The work presented in this memoir is mainly dedicated to the creation of dynamic web site to offer to the students and teachers the possibility to communicate with each other. This site should help promoting the international university research; to pass to the implementation of our project we made a study of cases using UML diagrams because it is adapted in all development phases and compatible with all the techniques of realizations following the process of development UP during all project phases.

We have expressed recent technical tools to realize of our web site.

This experience was very useful to as especially in the experiment of the methods and techniques of treatment.

Finally, the diploma which we are going to get will be an opening for us on the field of computer science research and an experience to share our field with other students.

***KEYS-WORDS:***

Internet, web Site, Web.

UP :( Unified Process).

UML :( Unified Modeling language).

PHP, EasyPHP, MySQL, Adobe Dreamweaver, HTML.

## « Tableau de matières »

<b>Introduction général</b>	
Introduction général	1
<b>Partie 1 : Etat de l'art UML et les applications web</b>	
<b>Chapitre 01 : L'internet et les applications web</b>	
1.1 Introduction	4
1.2 L'internet	4
1.3 Définitions sur le Web	5
1.3.1 WWW	5
1.3.2 Site web	6
1.3.3 Application Web	7
1.3.3.1 définition	7
1.3.3.2 Histoire	8
1.3.3.3 Création	8
1.3.3.4 Utilisations	8
1.3.3.5 La technique de base	9
1.3.3.6 Technologie	10
1.4 Conclusion	13
<b>Chapitre 02 : UML et le processus unifié (UP)</b>	
2.1 Introduction	15
2.2 UML	15
2.2.1 UML le langage de modélisation unifié	15
2.2.2 UML est un langage graphique	15
2.2.3 UML est un support de communication	15
2.2.4 UML n'est pas une méthode	15
2.2.5 UML est basé sur un méta- modèle	16
2.2.6 UML est une norme	16
2.2.7 UML est une unification des méthodes	16
2.2.8 Diagramme d'UML	16
2.3 Processus unifié (UP)	18
2.3.1 Définition	18
2.3.2 Catégories	18
2.3.3 Les phases	19
2.4 Une méthode simple et générique	20
2.4.1 Identification des besoins et spécification des fonctionnalités	21
2.4.2 Phase d'analyse	22
2.4.3 Phase de conception	24
2.5 Conclusion	25
<b>Partie 2 : La conception de l'application</b>	
<b>Chapitre 03 : identification des besoins</b>	
3.1 Introduction	27
3.2 Identifier les acteurs	27
3.3 Diagramme de contexte	28
3.4 Diagramme de cas d'utilisation	29
3.5 Les fiches descriptives	30
3.5.1 Fiche descriptive du cas d'utilisation « créer compte »	30
3.5.2 Fiche descriptive du cas d'utilisateur « s'identifier »	30





« Liste des figures »

<b>Introduction général</b>	
<b>Partie 1 : Etat de l'art UML et les applications web</b>	
<b>Chapitre 01 : L'internet et les applications web</b>	
Figure 1.1 : Le site web statique	7
Figure 1.2 Le site web dynamique	7
<b>Chapitre 02 : UML et le processus unifié (UP)</b>	
Figure 2.1 : l'itération dans le UP	18
Figure 2.2 : les différentes perspectives qui définissent un modèle d'architecture	19
Figure 2.3 : les besoins sont modélisés par un diagramme de cas d'utilisation	21
Figure 2.4 : Les diagrammes de séquence système illustrent la description textuelle des cas d'utilisation	21
Figure 2.5 : Une maquette d'IHM facilite les discussions avec les futurs utilisateurs	22
Figure 2.6 : La phase d'analyse du domaine permet d'élaborer la première version du diagramme de classes	22
Figure 2.7 : Le diagramme de classes participantes effectue la jonction entre les cas d'utilisation, le modèle du domaine et les diagrammes de conception logicielle	23
Figure 2.8 Les diagrammes d'activités de navigation représentent graphiquement l'activité de navigation dans l'IHM	23
Figure 2.9 : Les diagrammes d'interaction permettent d'attribuer précisément les responsabilités de comportement aux classes d'analyse	24
Figure 2.10 : Chaîne complète de la démarche de modélisation du besoin jusqu'au code.	24
<b>Partie 2 : La conception de l'application</b>	
<b>Chapitre 03 : identification des besoins</b>	
Figure 3.1: Diagramme de contexte	28
Figure 3.2: Diagramme de cas d'utilisation	29
Figure 3.3: Diagramme de séquence système cas d'utilisation « créer compte »	34
Figure 3.4: Diagramme de séquence système cas d'utilisation « s'identifier »	35
Figure 3.5: Diagramme de séquence système cas d'utilisation « Mettre document »	36
Figure 3.6: Diagramme de séquence système cas d'utilisation « Participer »	36
Figure 3.7: Diagramme de séquence système cas d'utilisation «Changer mot de passe ».	37
Figure 3.8: Diagramme de séquence système cas d'utilisation « Télécharger »	38
Figure 3.9: Diagramme de séquence système cas d'utilisation « s'inscrire »	38
<b>Chapitre 04 : Phase d'analyse</b>	
Figure 4.1 Diagramme de domaine	42
Figure 4.2 Diagramme de classe participante « Mettre document »	43
Figure 4.3 Diagramme de classe participante «Télécharger »	43
Figure 4.4 Diagramme de classe participante «Participer »	43
Figure 4.5 Diagramme de classe participante «Changer mot de passe »	44
Figure 4.6 Diagramme de classe participante «s'identifier »	44
Figure 4.7 Diagramme de classe participante «créer compte »	44
Figure 4.8 Diagramme de classe participante «s'inscrire »	45
Figure 4.9 Diagramme d'activité de navigation «Créer compte »	45

Figure 4.10	Diagramme d'activité de navigation «s'identifier »	46
Figure 4.11	Diagramme d'activité de navigation «mettre document »	47
Figure 4.12	Diagramme d'activité de navigation «Participer »	48
Figure 4.13	Diagramme d'activité de navigation «Changer mot de passe »	49
Figure 4.14	Diagramme d'activité de navigation «Télécharger »	50
Figure 4.15	Diagramme d'activité de navigation «s'inscrire »	51
<b>Chapitre 05 : Phase de conception</b>		
Figure 5.1	: diagramme d'interaction « créer compte »	53
Figure 5.2	: diagramme d'interaction « s'identifier »	54
Figure 5.3	: diagramme d'interaction « mettre document »	55
Figure 5.4	: diagramme d'interaction «participer »	56
Figure 5.5	: diagramme d'interaction «changer mot de passe »	57
Figure 5.6	: diagramme d'interaction «Télécharger »	58
Figure 5.7	: diagramme d'interaction «s'inscrire »	59
Figure 5.8	Diagramme de classe de conception « Mettre document »	59
Figure 5.9	Diagramme de classe de conception «Télécharger »	60
Figure 5.10	Diagramme de classe de conception «Participer »	60
Figure 5.11	Diagramme de classe de conception «Changer mot de passe »	60
Figure 5.12	Diagramme de classe de conception «s'identifier »	61
Figure 5.13	Diagramme de classe de conception «créer compte »	61
Figure 5.14	Diagramme de classe de conception «s'inscrire »	61
Figure 5.15	: Diagramme de classe	62
<b>Partie 03 : La réalisation de l'application</b>		
<b>Chapitre 06 : Implémentation</b>		
Figure 6.1	: l'architecture de PHP	68
Figure 6.2	: Carte de site	70
Figure 6.3	: L'interface de notre application	71
Figure 6.4	: L'interface « s'inscrire »	72
Figure 6.5	: L'interface « mettre document »	73
Figure 6.6	: L'interface « emploi du temps »	74
<b>Conclusion général</b>		





**INTRODUCTION  
GÉNÉRAL**

Aujourd'hui, parmi tous les réseaux, Internet peut être considéré comme le réseau spécialisé dans l'informatique, son apport prend de l'empierre de jour en jour, d'où son but n'est plus comme à l'origine, de transmettre quelques lignes à partir de messageries ; mais d'échanger des documents électroniques, des données informatisées, des informations économiques, des schémas, des sonsí etc.

En quelques années, la notoriété d'Internet est ainsi passée de la simple découverte à une explosion de services intéressants les professionnels comme les particuliers. Par exemple, dans le domaine économique, Internet se présente comme un outil d'information de tout premier plan offrant au entreprise de nombreux services interactifs : marketing en direct, publicités, affiche commercialí

Ce phénomène (Internet) d'interaction offre des possibilités considérables et accroit les chances de communication à travers un lieu virtuel qui regroupe des pages Internet présentées par des personnes ou des organisations, c'est **le site web**.

Les anciens sites web permettaient des applications simples supportant la navigation et la recherche d'information dans un cadre restreint pour une simple consultation (lecture seul). Aujourd'hui leur évolution a permit de supporter des applications possiblement distribuées avec des opérations affectant le contenu (information) et l'état de la navigation ainsi que des opérations intégrées dotant d'une capacité de recherche prédéfinie des hypermédia , d'où la convergence entre le domaine de l'hypermédia et les systèmes d'information qui adoptent de plus en plus la navigation comme paradigme d'interaction fondamental.

## Objectif du travail

Notre travail consiste à réaliser un site web dynamique pour la communication entre les enseignants et les étudiants des Centre Universitaire de Mila.

Les grands objectifs du projet sont comme suit :

- Permettre aux enseignants et étudiants de communiquer entre eux à n'importe quel moment et à n'importe quel lieu :
  - Permettre aux enseignants de mettre des cours, des examens, des solutions des examens, des documents, de répondre aux questionsí etc.
  - Permettre aux étudiants de télécharger les cours et les documents, de poser des questions et de des suggérer propositions í etc.
- Permettre aux visiteurs de visiter le site et de télécharger n'importe quel document.

En effet, le développement de notre application web repose sur une conception qui utilise des diagrammes du langage UML en suivant un processus de développement UP (une démarche simplifiée). La réalisation pratique fait appel à des outils technologiques récents tels que PHP et MYSQL. L'implémentation est effectuée sur une plate-forme permettant de discuter et de communiquer.

## **Contenu du mémoire**

Le mémoire est constitué d'une introduction générale, de trois parties, et une conclusion générale.

### **La première partie :**

**Chapitre 01-** L'internet et les applications web : ce chapitre a pour objectif essentiel la présentation de l'internet et les différentes applications web, Word Wide Web

**Chapitre 02-** L'outil de modélisation (UML) Et Le processus de développement (UP): dans le deuxième chapitre, nous définissons quelques concepts jugés nécessaire sur UML, et le processus unifié (UP).

### **La deuxième partie :**

**Chapitre 03-** Identification des besoins: dans ce chapitre, nous identifions les acteurs et les cas d'utilisations en proposant les diagrammes de contexte, des cas d'utilisations et de séquence système ainsi que les fiches descriptives.

**Chapitre 04-** Phase d'analyse: ce chapitre sert à déterminer l'analyse du domaine en suivant la démarche suivante : En commençant par l'identification des concepts du domaine, ensuite nous proposons le modèle de domaine, les diagrammes de classes participantes et les diagrammes d'activités de navigation.

**Chapitre 05-** Phase de conception: dans ce chapitre nous représentons les diagrammes d'interaction et le diagramme de classes de conception.

### **La troisième partie :**

**Chapitre 06-** implémentation et réalisation : dans ce chapitre nous abordons les outils de développement tel que : PHP, MYSQL, HTML, EasyPHP, et Adobe DreamWeaver.CS3

Nous terminons ce mémoire par une conclusion général.



**PARTIE 1**  
**ETAT DE L'ART**  
**UML ET LES**  
**APPLICATIONS**  
**WEB**



**CHAPITRE 01 :**  
**L'INTERNET ET**  
**LES**  
**APPLICATIONS**  
**WEB**

## 1.1 Introduction :

Dans ce chapitre, on va présenter d'une façon générale l'internet et les applications web. Plus précisément, ce chapitre vise à répondre aux questions suivantes :

Qu'est ce que l'internet ?

Qu'est ce qu'une application web ?

## 1.2 L'Internet :

L'Internet est un réseau mondial de réseaux informatiques, ayant chacune une adresse unique, qui utilise la technologie du système téléphonique et les liaisons par satellite pour communiquer des informations à toutes les 24 heures par jour en utilisant un logiciel standard commun. ordinateurs autonomes et privés des réseaux locaux dans les universités, entreprises et autres organisations communiquer les uns avec les autres plus haute vitesse de transmission appartenant dorsales et exploités par les fournisseurs de services Internet. [1]

La communication s'effectue via un modem ou une connexion réseau permanente. L'Internet se compose essentiellement d'un réseau des petits réseaux et des ordinateurs autonomes. La plupart des fournisseurs de services se connecter à un ou plusieurs autres fournisseurs de services, étendant ainsi la portée de leurs réseaux. Avec des organisations gouvernementales, les organisations commerciales et les particuliers autour des fournisseurs globe se connecter à Internet, et un service sans cesse croissante de nombre de réseaux auxquels ils ont des liens directs, l'Internet est de plus en plus de façon exponentielle. Il n'y a pas d'autorité centrale qui contrôle l'Internet. Il ya cependant des organisations comme l'Internet Engineering Task Force (IETF [2]) qui développent et établissent des normes pour création et l'utilisation des applications Web et des organisations comme l'International Internet Preservation Consortium (IIPC [3]) qui établissent des normes pour la préservation du web ressources. L'épine dorsale Internet est la propriété de sociétés privées travaillant ensemble.

Projects Agency (ARPA) pour communiquer un certain nombre d'installations informatiques à travers le pays. ARPA parrainé la recherche sur la façon de relier le département américain de la Défense a financé l'Advanced Research ordinateurs géographiquement éloignés de partager des données et des ressources. Parce que le



département américain de la Défense a été très préoccupé par une attaque nucléaire, l'ARPA Network a été conçu pour continuer à travailler même si une partie du réseau a été bombardé. Il a été conçu de telle sorte que dès qu'une partie du réseau cesse de fonctionner, le réseau réachemine automatiquement les fonctionnalités et la circulation de l'information de la partie pour le reste du réseau. Le début des années 1970 a été consacré à l'élaboration de normes pour le transfert de données. Ces normes sont connues sous le nom de protocoles. Leur but était de favoriser le partage des données entre ordinateurs fonctionnant sur le même réseau pour assurer l'interconnexion des différents ordinateurs sur les réseaux. TCP/ IP (Transmission Control Protocol et Internet Protocol) qui a été le protocole de base de l'Internet depuis 1983 a été élaboré entre 1973 et 1978. Ce réseau informatique développé au cours des années 1970 et 1980. Au début, seuls militaires institutions, organismes gouvernementaux et les universités ont été connectés à communiquer et à partager des informations et des ressources. Au cours de la fin des années 1980 et début des années 1990 plus entreprises et les particuliers ont rejoint le réseau pour communiquer les uns avec les autres lorsque le gouvernement américain a commencé à se retirer de la gestion du réseau. Cela a permis commerciales entités de vendre l'accès Internet pour le grand public. Depuis lors, l'Internet est devenu une épine dorsale de communication qui fournit différents services à ses utilisateurs, comme le World Wide Web, E-mail, forums de discussion, Les listes de diffusion, de vidéoconférence, les groupes de discussion et à des intranets.

### 1.3 Définitions sur le Web :

C'est l'appellation la plus courte et la plus commune pour désigner le World Wide Web (WWW). [4]

#### 1.3.1 WWW :

Système d'information multimédia, basée sur des liens hypertextes [4] crée en 1993 pour les besoins du Centre européen pour la recherche nucléaire (CERN). Le 1<sup>er</sup> Navigateur Web (NCSA Mosaic) a été créé la même année par le National Center for Supercomputing Applications (NCSA).il permettant de naviguer d'un site à un autre sur Internet.

Pour le moment, le Web intègre pratiquement l'ensemble des services présents sur les réseaux. Il a tellement simplifié le travail sur internet que même les utilisateurs n'ayant aucune expérience informatique apprennent immédiatement à se servir de son interface

graphique. Le client web sur l'ordinateur local (browser ou navigateur) s'adresse à un serveur web du réseau.

Le principe du web, conçu par Tim Berners-Lee en 1990 au CERN, s'appuie sur le fait que de grandes masses d'information en réseau peuvent être représentées par une arborescence. Le point de départ du travail sur le web est la racine d'un serveur. Celle-ci contient en général un grand nombre de liens et de références.

### 1.3.2 Site web :

Un **site Web** est un ensemble de pages Web hyperliées entre elles et mises en ligne à une adresse Web [7]. On dit aussi **site Internet** par métonymie, les pages n'est pas figé mais variable en fonction de l'heure, du visiteur ou d'autres paramètres.

Il existe deux types des sites web :

#### a) **Site web Statique:**

Le serveur web (HTTP) se contente d'envoyer des fichiers stockés sur disque dur.

- **Avantage :**

- Permet de réduire le coût de l'hébergement.
- Le niveau nécessaire pour la réalisation de ce type de site est faible.
- Rapidité d'affichage.
- Optimisation moteur grandement facilité.
- Ce type de site est parfaitement exploitable sur disquette (1,4Mo) ou sur CDROM, - - DVD, clé USB, í

- **Désavantage :**

- Peut évolutif.
- Chaque modification coûte du temps, beaucoup de temps parfois.



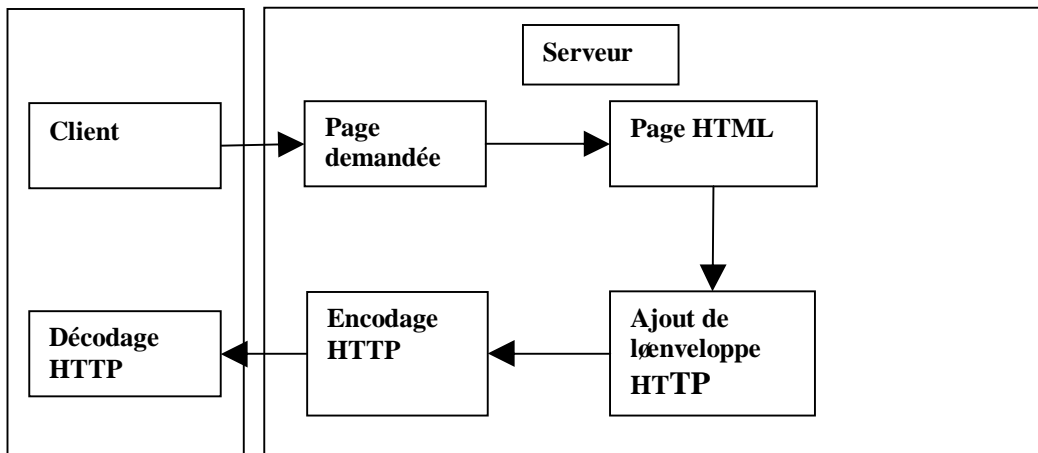


Figure 1.1: Le site web statique [5]

**b) Site web dynamique (application web):**

Le serveur web (HTTP) exécute un ou plusieurs programmes qui vont renvoyer des données, que le serveur web retransmet au navigateur.

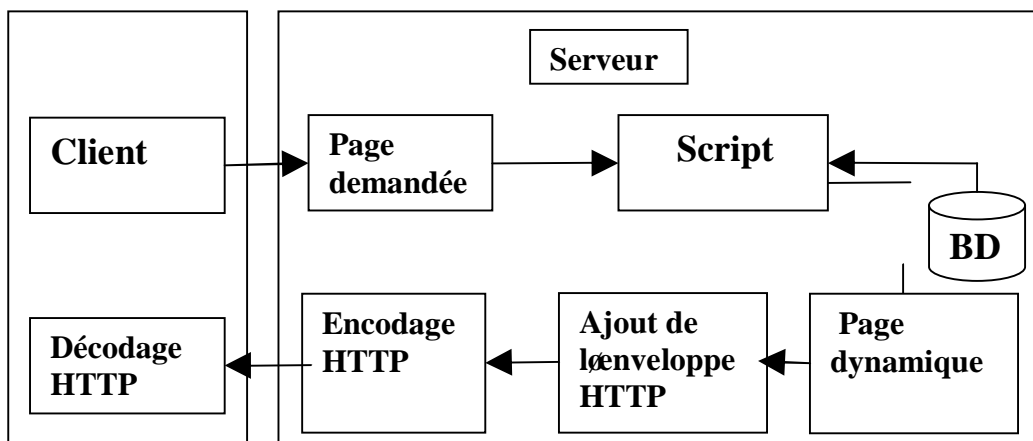


Figure 1.2 Le site web dynamique [5]

**1.3.3 L'Application Web:**

**1.3.3.1 Définition :**

En informatique, une **application Web** (site Web dynamique) est un **logiciel** applicatif manipulable grâce à un **navigateur Web** (liens, cliqués, formulaires, cookies...). De la même manière que les **sites Web**, une application Web est généralement placée sur un **serveur**(base de données SQL, fichiers de configuration...) et se manipule en actionnant des **widgets** à l'aide d'un navigateur Web, via un **réseau informatique** (Internet, intranet, réseau local, etc.) [6]

**Quelques exemples sur les applications :**

Les systèmes de gestion de contenu, les wikis, les blogs, les moteurs de recherches, les logiciels de commerce électronique, les jeux en ligne, les logiciels de forum et peut rajouter les appareils réseau tels par exemple les routeurs.

### 1.3.3.2 Histoire :

Grace au www, les utilisateurs peuvent se partager des documents et des images plus rapidement que via le **courrier électronique** et plus facilement que via les fichiers. La **norme industrielle CGI** qui a été publié en 1995 par le NCSA spécifie quelles sont les modalités d'extension d'un serveur Web pour le brancher avec un logiciel applicatif par exemple un moteur de recherche.

L'évolution très rapide de la technologie des applications web été entre 1994 et 2000, Rich Internet Application (1re application web) c'est une application Web qui offre une maniabilité et une ergonomie équivalente à une application client. Le terme Rich Internet Application est apparu la première fois dans une publication de Macro média en 2002 [6].

### 1.3.3.3 Création :

Les applications web sont souvent créées par des équipes composées à la fois de développeurs et de designers. Le développement nécessite la connaissance des différents langages utilisés dans les technologies du Web : **HTML** pour la présentation des pages, **CSS** pour la charte graphique, **JavaScript, Java** ou **Action Script** pour les automatismes exécutés par le client, ainsi qu'un langage tels que par exemple **Java, PHP, C#** ou **VB Script** pour les automatismes exécutés par le serveur.

### 1.3.3.4 Utilisations :

Les moteurs de recherche, le web mail, le commerce électronique et les jeux en ligne sont apparus grâce à la technologie des applications web.

Un **web mail** est une application web pour recevoir et envoyer du courrier électronique.

Un **système de gestion** de contenu est une application web qui présente des documents. La présentation des documents est similaire à celle d'un site web, cependant les documents sont générés par le logiciel lors de chaque demande. Le système de gestion de contenu effectue les traitements nécessaires à la mise en forme et la présentation des documents.

Un **weblog** est un système de gestion de contenu où des éléments de contenu sont présentés dans l'ordre chronologique de leur date de création.

Un **wiki** (de l'hawaïen wiki wiki qui signifie vite) est un système de gestion de contenu qui vise à simplifier la création collaborative des documents. Il autorise plusieurs personnes à effectuer des modifications simultanées, et il est équipé d'espaces de discussion.

Un **site web** marchand est un système de gestion de contenu, où le contenu sont des annonces concernant des produits. Il est utilisé pour la vente par correspondance. Les visites et les opérations d'achat sont enregistrées à des fins de marketing. Les sites web marchands sont utilisés aussi bien pour la vente des produits d'une société que pour des ventes entre particuliers ou des ventes aux enchères.

Un **jeu** par navigateur est un jeu vidéo réalisé sous la forme d'une application web.

Un **logiciel de forum** permet des discussions ouvertes entre des utilisateurs : un utilisateur écrit un message, et ce message peut être lu par tous les autres utilisateurs. Les logiciels de forums sont parfois réalisés sous forme d'application web.

La **messagerie instantanée** (anglais chat) permet l'échange instantané de messages texte entre différents utilisateurs. Les logiciels de messagerie instantanée sont parfois réalisés sous forme d'application web. Les messages peuvent être transmis à un autre utilisateur du logiciel, ou un téléphone portable via le short message service (abrégé SMS).

**Google Maps** est une application web qui permet de consulter des cartes géographiques du monde entier.

**Facebook** est une application web qui permet à chaque utilisateur de se constituer un réseau social (amis, associés, personnes qui partagent les mêmes centres d'intérêt).

### 1.3.3.5 La technique de base :

Concernant la technologie **client-serveur**, le navigateur Web envoie des **requêtes** de **pages Web** au serveur, ce dernier répond par des pages peuvent être affichées par le navigateur web.

Dans une application Web, les pages sont créées de toute pièce par le logiciel lors de chaque requête. Chaque **hyperlien** contenu dans la page provoque l'envoi d'une nouvelle

requête, qui donnera en résultat une nouvelle page. Par opposition à un **site Web** statique où les pages sont des **fichiers** préalablement enregistrés.

Les pages Web contiennent divers **widgets** tels des **boutons poussoirs**, des **icônes** et des **zones de texte**, permettant la manipulation de l'application. Chaque manipulation d'un **bouton poussoir** provoque l'envoi d'une nouvelle requête. Les pages Web peuvent contenir des **applets**.

Contrairement à d'autres logiciels, une application Web mise en place sur un **serveur** est immédiatement utilisable par le consommateur sans procédure d'achat et d'**installation** sur son propre ordinateur, du moment que l'ordinateur du consommateur est équipé d'un navigateur Web et d'une connexion réseau. Ceci évite des interventions des **administrateurs système**, interventions qui sont souvent plus coûteuses que le logiciel lui-même. L'application Web est souvent mise à disposition du consommateur par l'éditeur du logiciel sur ses propres serveurs- technique appelée **Software as a Service**.

L'usage du navigateur Web comme partie **client** - un logiciel qui est disponible sur de nombreux **système d'exploitation** - assure la **portabilité** d'une application Web.

#### 1.3.3.6 Technologie :

Dans la technologie la plus courante, le logiciel applicatif est branché sur le **serveur web**, le tout parfois accompagné d'un serveur de **base de données**. L'ensemble est appelé **serveur d'application**.

Le **code source** du logiciel applicatif est placé directement dans des pages web. Ces pages sont stockées par le **serveur**. Lorsque le **client** demande une page, le serveur web va rechercher la page, puis exécute les instructions qu'elle contient. Ces instructions peuvent faire appel au serveur de base de données. Le serveur web transmet la page avec le résultat de l'exécution au **client**.

La transmission des informations entre le client et le serveur se fait selon le **protocole HTTP**, protocole également utilisé pour les sites web. Ce qui permet d'utiliser le même logiciel **client** - un **navigateur web**.

Les applications web font souvent usage du mécanisme des **cookies** : en réponse à une requête, le serveur envoie une information de repérage au client (le cookie). Puis le client va

lui renvoyer cette information lors de la prochaine requête. Le mécanisme est utilisé pour identifier le client et suivre les manipulations.

Les pages web peuvent en outre contenir des **applets**. Ce sont des morceaux de code source qui sera exécuté par le navigateur web après transmission de la page - contrairement à la majorité du code source qui est exécuté par le serveur web avant la transmission. **ActiveX**, **Java**, **Adobe Flash** et **Silverlight** sont des technologies utilisées pour les applets.

Le logiciel client - le **navigateur web** - est identique à celui utilisé pour consulter un **site web**. Les logiciels nécessaires pour exécuter les applets sont inclus dans les navigateurs - parfois sous forme de complément - **plugin**.

### ❖ **Logiciel serveur :**

Le logiciel serveur reçoit les demandes provenant du navigateur web, et effectue les traitements, puis envoie le résultat du traitement au client. Le serveur web peut être branché par **PHP**, ce dernier est un interprète qui permet l'exécution des logiciels écrits en langage PHP. Le résultat de cette opération est envoyé au serveur web, qui le transmettra au client. **WebObjects**, développé par **Apple**, est un logiciel de serveur web qui exécute le **code source** contenu dans les pages, code source écrit en langage **Java**.

**ASP** est une technologie développée par **Microsoft**, composée d'un interprète, qui se branche sur le logiciel serveur web **Internet Information Services**. L'interprète exécute le code source contenu dans les pages, code source écrit en langage **VB Script**.

**JSP** est une technologie développée par **Sun Microsystems**, composée d'un logiciel de serveur web auquel peut se brancher un logiciel applicatif écrit en langage **Java**. Le logiciel applicatif est composé de pages web contenant du code source préalablement **compilé**.

**ASP.Net** est une technologie développée par **Microsoft** en remplacement de **ASP** composée d'un logiciel qui se branche sur le logiciel de serveur web **Internet Information Services** et qui exécute le logiciel applicatif. Le logiciel applicatif est composé de pages contenant du code source écrit dans un langage qui utilise le **Framework** de programmation **.Net (Visual Basic .NET ou C#)**, le logiciel applicatif est préalablement **compilé**

**LAMP (Linux - Apache - MySQL - PHP)** est une suite de logiciels **open source** qui, utilisés ensemble, permettent de réaliser des serveurs d'application. Par définition, cette suite

est composée du **système d'exploitation Linux**, du logiciel **serveur web Apache**, de l'interprète **PHP**, et du **système de gestion de base de données MySQL**.

Les logiciels de serveur d'application sont également utilisés pour réaliser des **services web**.

### ❖ **Navigateur web :**

Le **navigateur web** est le logiciel qui affiche les résultats envoyés par le serveur, reçoit les manipulations de l'utilisateur et les transmet au **serveur**.

Une **applet** est un morceau de logiciel incorporé dans une page web, et exécuté par le navigateur web de l'ordinateur **client**. Lorsque l'utilisateur actionne un **widgets** placé dans une page web, l'applet peut alors modifier la présentation de la page (technique appelée **DHTML**), afficher des messages ou envoyer des **requêtes** au serveur d'application.

**JavaScript** , initialement appelé Live Script est un **langage de programmation** pour les applets, développé par **Netscape**. Les applets écrites dans ce langage sont exécutées par un interprète inclus dans le navigateur web. De nombreux navigateurs web ont un interprète JavaScript.

Les premiers navigateurs web équipés d'un **interprète** JavaScript présentaient des différences de dialecte qui font qu'une applet écrite pour un interprète en particulier n'est pas toujours comprise par un autre interprète. Pour pallier ce problème, **Ecma International** publie en 1999 la **norme industrielle** ECMA-262 **ECMAScript** qui spécifie la syntaxe que doivent comprendre les interprètes JavaScript.

Le **Document Object Model** (abrégé DOM) est un ensemble d'**objets** normalisé qui représente la page affichée, ainsi que le navigateur web. Le DOM est l'**interface de programmation** utilisée dans les applets pour effectuer des modifications sur la page.

**AJAX** est une technologie en langage **JavaScript** dans laquelle une **applet** communique avec le serveur d'application sans intervention de l'utilisateur (parcours d'un **hyperlien**). Cette technologie est souvent utilisée pour améliorer la maniabilité d'une application web. Voir aussi **Rich Internet Application**.

Java est un langage de programmation développé par Sun Microsystems, qui peut être utilisé pour les applets. Les applets écrites dans ce langage sont préalablement compilées, et

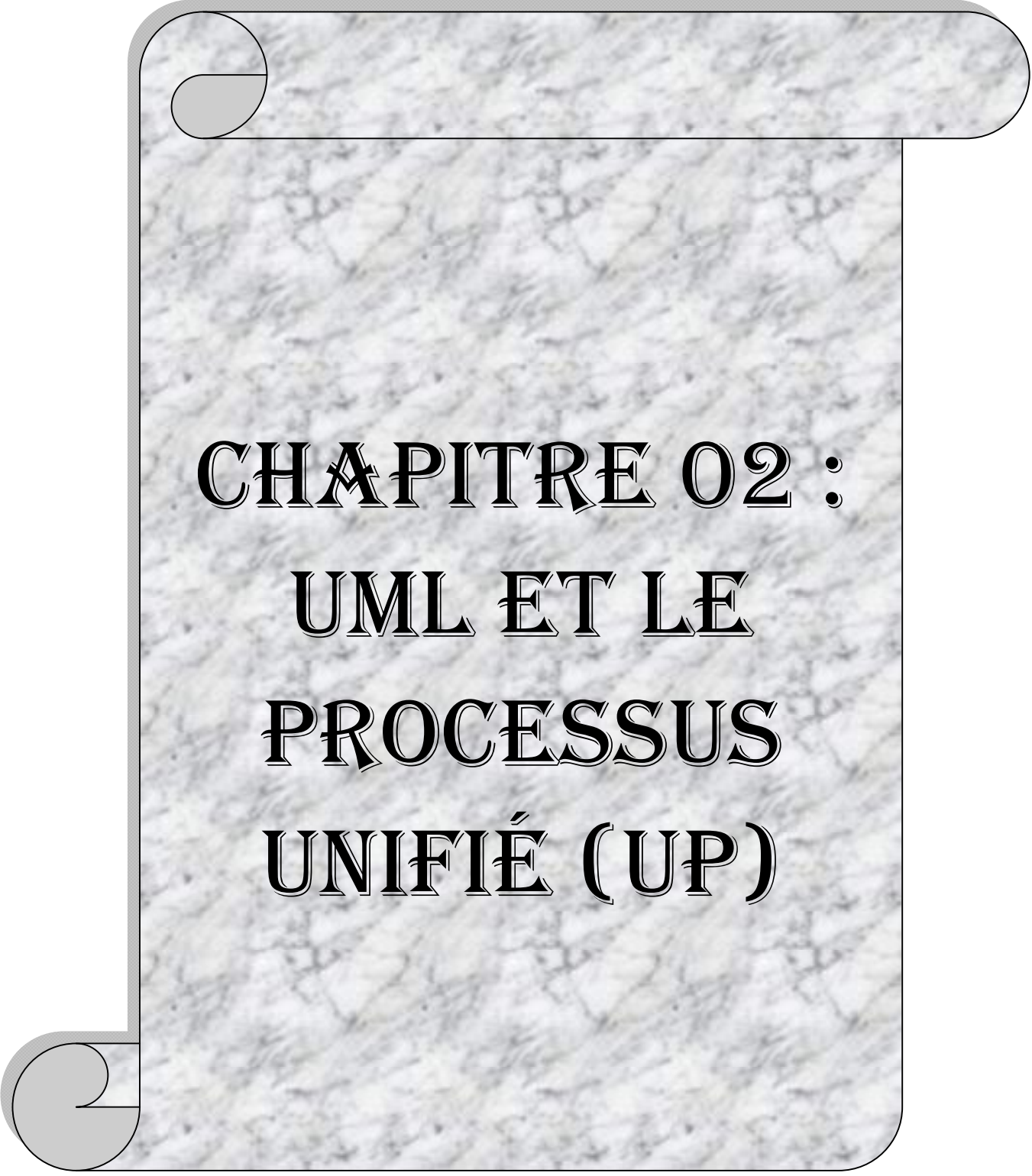
exécutées par un logiciel branché au navigateur web, le plug-in Java (traduction littérale : qui se branche dessus). Voir environnement d'exécution Java.

**ActiveX** est une technologie développée par **Microsoft** où des **composants logiciels** peuvent être inclus en tant qu'applet dans des pages web. Cette technologie nécessite un système d'exploitation **Windows** sur l'ordinateur **client**.

**Flash** est une technologie développée par **Adobe**. Un logiciel branché au navigateur, le plug-in Flash permet d'afficher des animations, des **images vectorielles**, des vidéos, et exécuter des applets. Il comporte un interprète pour des applets en langage de programmation Action Script, un langage similaire à JavaScript. Cette technologie est d'usage courant pour les jeux vidéo en ligne.

## 1.4 Conclusion :

Dans la fin d'étude de ce chapitre, on déduit que l'internet est développé de temps en temps et on a distingué qu'il existe plusieurs applications web.



**CHAPITRE 02 :**  
**UML ET LE**  
**PROCESSUS**  
**UNIFIÉ (UP)**



## 2.1 INTRODUCTION :

Dans ce chapitre on propose une présentation générale de UML, de UP, ses objectifs, ses domaines d'application. On va parler des concepts de base de UML. Principalement ce chapitre vise à répondre aux questions suivantes :

Qu'est-ce que UML ?

Qu'est-ce que UP ?

## 2.2 UML :

### 2.2.1 UML le langage de modélisation unifié :

C'est la forme contractée du langage unifié pour la modélisation. Il fournit les fondements pour spécifier, construire, visualiser et décrire les éléments d'un modèle. Il se base sur une sémantique précise et sur une notation graphique expressive. C'est un moyen d'exprimer des modèles en faisant abstraction de leur implémentation, c'est à dire le modèle fourni par UML est valable pour n'importe quel langage de programmation [8].

### 2.2.2 UML est un langage graphique :

C'est un ensemble de graphisme très complet. Sa notation graphique permet d'exprimer visuellement une solution du problème. L'aspect formel de sa notation limite les ambiguïtés et les incompréhensions. Son aspect visuel facilite la comparaison et l'évaluation des solutions [8].

### 2.2.3 UML est un support de communication :

Il facilite l'expression et la communication de modèles en fournissant un ensemble de symboles (la notation) et de règles qui régissent l'assemblage de ces symboles (la syntaxe et la sémantique) [8].

### 2.2.4 UML n'est pas une méthode :

UML n'est pas une méthode ou un processus UML développé pour permettre la modélisation des systèmes d'une manière standard et pas pour être une méthode de conception ou d'analyse, pour ça il lui manque la démarche.

### 2.2.5 UML est basé sur un méta- modèle :

Un modèle de plus haut niveau qui définit les éléments d'UML (les concepts utilisables) et leur mode d'utilisation (leur signification et leur mode d'utilisation).

### **2.2.6 UML est une norme :**

UML est devenu une référence en matière de langages de description des systèmes et d'aide à la conception.

### **2.2.7 UML est une unification des méthodes :**

UML est né de la fusion des trois méthodes ont le plus influencé la modélisation objet au milieu des années 90 :

- OMT de James Rumbaugh (General Electric) fournit une représentation graphique des aspects statique, dynamique et fonctionnel d'un système ;
- OOD de Grady Booch, définie pour le Department of Defense, introduit le concept de paquetage (package) ;
- OOSE d'Ivar Jacobson (Ericsson) fonde l'analyse sur la description des besoins des utilisateurs (cas d'utilisation, ou use cases).

### **2.2.8 Diagramme d'UML :**

- **Diagramme d'activité :**

Le diagramme d'activité n'est autre que la transcription dans UML de la représentation du processus telle qu'elle a été élaborée lors du travail qui a préparé la modélisation : il montre l'enchaînement des activités qui concourent au processus.

- **Diagramme de cas d'utilisation :**

Le diagramme de cas d'utilisation décrit la succession des opérations réalisées par un acteur (personne qui assure l'exécution d'une activité). C'est le diagramme principal du modèle UML, celui où s'assure la relation entre l'utilisateur et les objets que le système met en œuvre.

- **Diagramme de classe :**

Le diagramme de classe représente l'architecture conceptuelle du système : il décrit les classes que le système utilise, ainsi que leurs liens, que ceux-ci représentent un emboîtement

conceptuel (héritage, marqué par une flèche terminée par un triangle) ou une relation organique (agrégation, marquée par une flèche terminée par un diamant).

- **Diagramme d'objet :**

Le diagramme d'objet permet d'éclairer un diagramme de classe en l'illustrant par des exemples.

- **Diagramme de séquence :**

Le diagramme de séquence représente la succession chronologique des opérations réalisées par un acteur : saisir une donnée, consulter une donnée, lancer un traitement ; il indique les objets que l'acteur va manipuler, et les opérations qui font passer d'un objet à l'autre. On peut représenter les mêmes opérations par un *diagramme de collaboration*.

- **Diagramme de collaboration :**

Décrit les scénarios de chaque cas d'utilisation en mettant l'accent sur l'interaction des objets et les messages échangés.

- **Diagramme état-transition :**

Montre les différents états des objets en réaction aux événements.

- **Diagramme de composants :**

Représente les différents constituants logiciels d'un système (architecture logicielle d'un système).

- **Diagramme de déploiement :**

Décrit l'architecture technique d'un système (composants matériels du système informatique et leurs connexions).

- **Diagrammes de communication :**

Permet de mettre en évidence les interactions entre les objets du système étudié, on utilisera à de niveaux :

Phase d'analyse (représenter les interactions) et décrire les interactions

## 2.3 Processus unifié (UP):

### 2.3.1 Définition :

Le processus unifié est une méthode de prise en charge du cycle de vie d'un logiciel et donc du développement, pour les logiciels orientés objets [10]. C'est une méthode générique, itérative et incrémentale, contrairement à la méthode séquentielle **Merise** [9] (ou **SADT**).

PU vient compléter la systémique des modèles **UML**. Elle est le résultat final d'une évolution de l'approche d'Ericsson qui est au fondement d'une des premières méthodes de développement pour applications orientées objets, la méthode Objectory Process (1987). Objectory Process (version 1 à 3.8 en 1995) a elle-même servi de base à la société Rational pour la création de Rational Objectory Process (1997) (version 4.1), parente direct de RUP en 1998.

Les divers avatars d'UP sont : RUP, XUP, AUP, EUP, 2TUP, EssUP

### 2.3.2 Catégories :

#### a) UP est itératif :

L'itération est une répétition d'une séquence d'instructions ou d'une partie de programme un nombre de fois fixé à l'avance ou tant qu'une condition définie n'est pas remplie, dans le but de reprendre un traitement sur des données différentes.

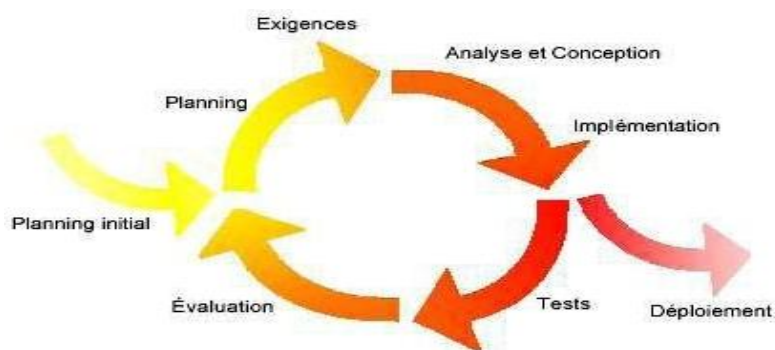


Figure 2.1 : l'itération dans le UP

#### b) UP est centré sur l'architecture :

Les différentes perspectives, indépendantes et complémentaires, qui permettent de définir un modèle d'architecture sont proposées par Ph. Kruchten.



Figure 2.2 : les différentes perspectives qui définissent un modèle d'architecture

### c) UP est piloté par les cas d'utilisation d'UML :

La satisfaction des besoins du client est le but principal d'un système informatique. Le processus de développement sera donc axé sur l'utilisateur. Les cas d'utilisation permettent d'illustrer ces besoins. Ils décrivent puis décrivent les besoins fonctionnels (du point de vue de l'utilisateur), et leur ensemble constitue le modèle de cas d'utilisation qui dicte les fonctionnalités complètes du système.

#### 2.3.3 Les phases :

##### a) Analyse des besoins :

L'analyse des besoins donne une vue du projet sous forme de produit fini. Elle porte essentiellement sur les besoins principaux (du point de vue de l'utilisateur), l'architecture générale du système, les risques majeurs, les délais et les coûts. On met en place le projet.

Elle répond aux questions suivantes :

Que va faire le système ? par rapport aux utilisateurs principaux, quels services va-t-il rendre ?

Quelle va être l'architecture générale (cible) de ce système ?

Quels vont être : les délais, les coûts, les ressources, les moyens à déployer ?

##### b) Elaboration :

L'élaboration reprend les éléments de la phase d'analyse des besoins et les précise pour arriver à une spécification détaillée de la solution à mettre en œuvre.

Elle permet de préciser la plupart des cas d'utilisation, de concevoir l'architecture du système et surtout de déterminer l'architecture de référence. Au terme de cette phase, les chefs de projet doivent être en mesure de prévoir les activités et d'estimer les ressources nécessaires à l'achèvement du projet.

Les tâches à effectuer dans la phase élaboration sont les suivantes :

- Créer une architecture de référence
- Identifier les risques
- Définir les niveaux de qualité à atteindre.
- Formuler les cas d'utilisation.
- Élaborer une offre abordant les questions de calendrier, de personnel et de budget

### c) Construction :

La construction est le moment où l'on construit le produit. L'architecture de référence se métamorphose en produit complet. Le produit contient tous les cas d'utilisation que les chefs de projet, en accord avec les utilisateurs ont décidé de mettre au point pour cette version.

### d) Transition :

Le produit est en version bêta. Un groupe d'utilisateurs essaye le produit et détecte les anomalies et défauts.

Cette phase suppose des activités (la formation des utilisateurs clients, la mise en œuvre d'un service d'assistance, la correction des anomalies constatées).

## 2.4 Une méthode simple et générique :

C'est une méthode qui se situe à mi-chemin entre UP (Unified Process), qui constitue un cadre général très complet de processus de développement, et XP (eXtreme Programming) qui est une approche minimaliste à la mode centrée sur le code. [12]

### 2.4.1 Identification des besoins et spécification des fonctionnalités :

➤ **diagramme de cas d'utilisation :**

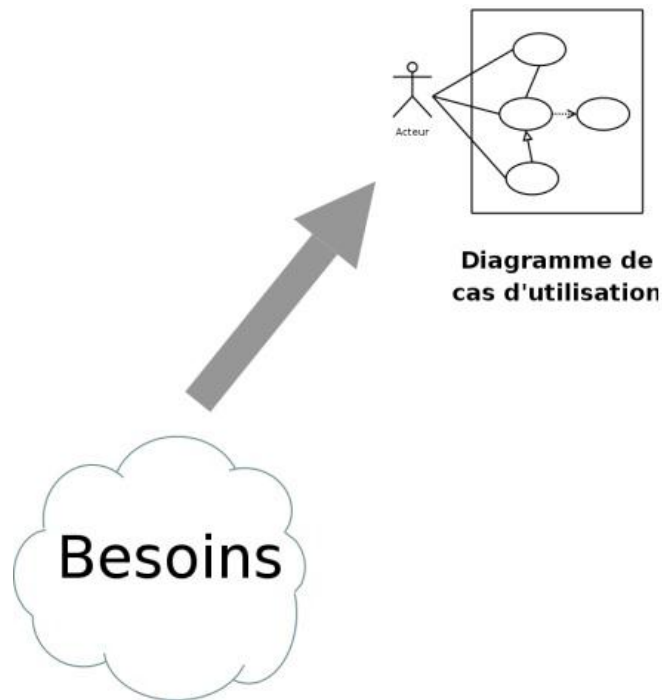


Figure 2.3 : les besoins sont modélisés par un diagramme de cas d'utilisation[12]

➤ diagramme de séquence système :

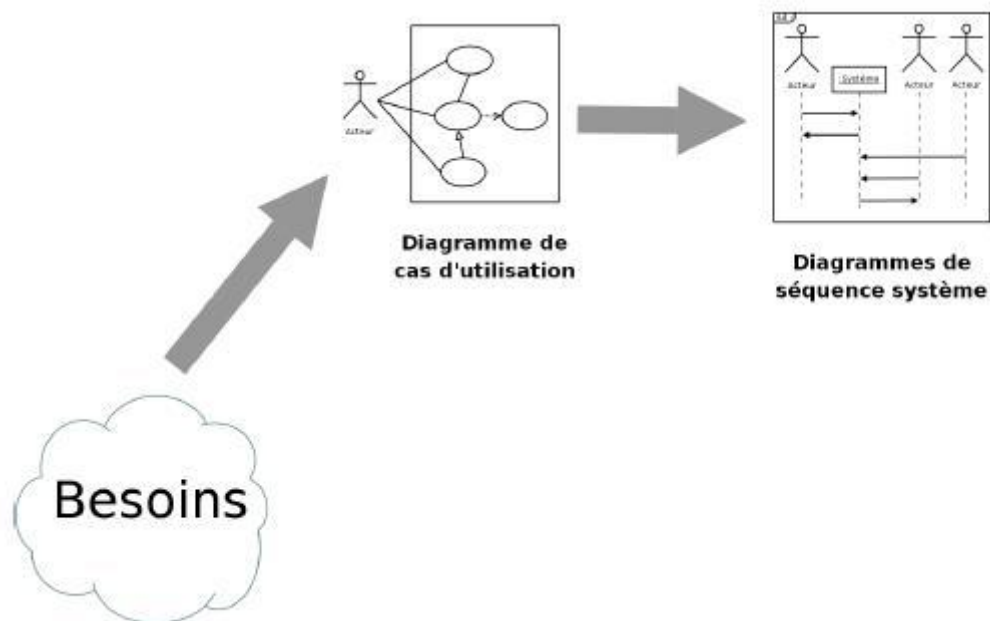


Figure 2.4 : Les diagrammes de séquence système illustrent la description textuelle des cas d'utilisation. [12]

➤ Maquette IHM :

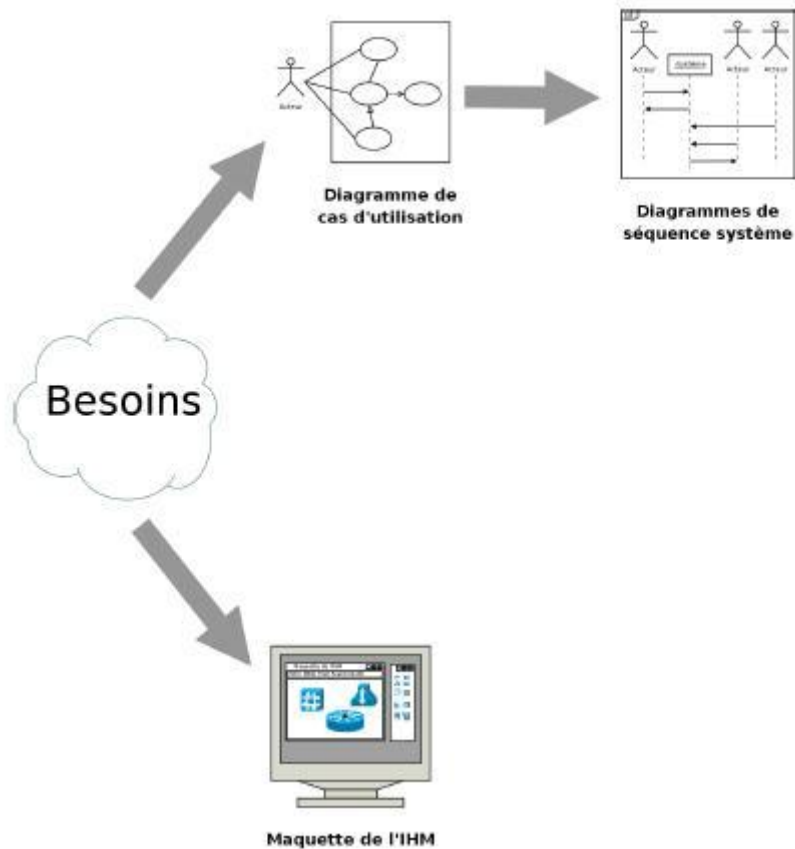


Figure 2.5 : Une maquette d'IHM facilite les discussions avec les futurs utilisateurs. [12]

## 2.4.2 Phase d'Analyse :

- diagramme de domaine :

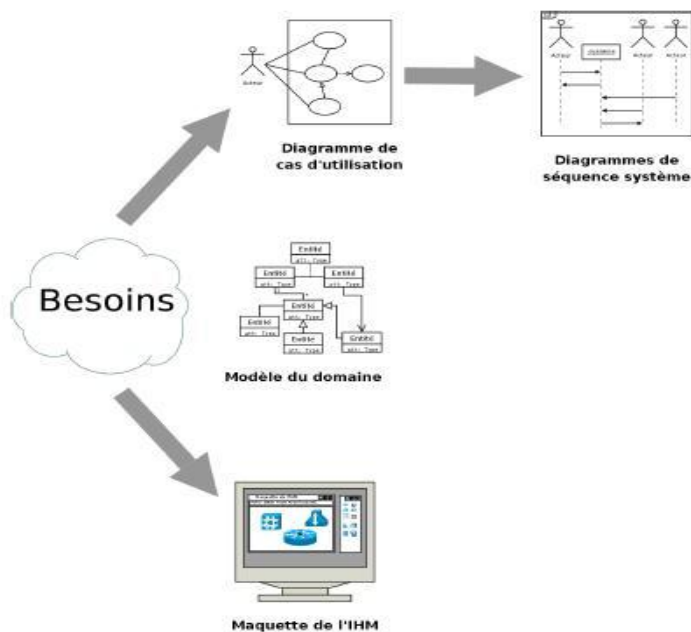


Figure 2.6 : La phase d'analyse du domaine permet d'élaborer la première version du diagramme de classes [12].

- diagramme des classes participantes :



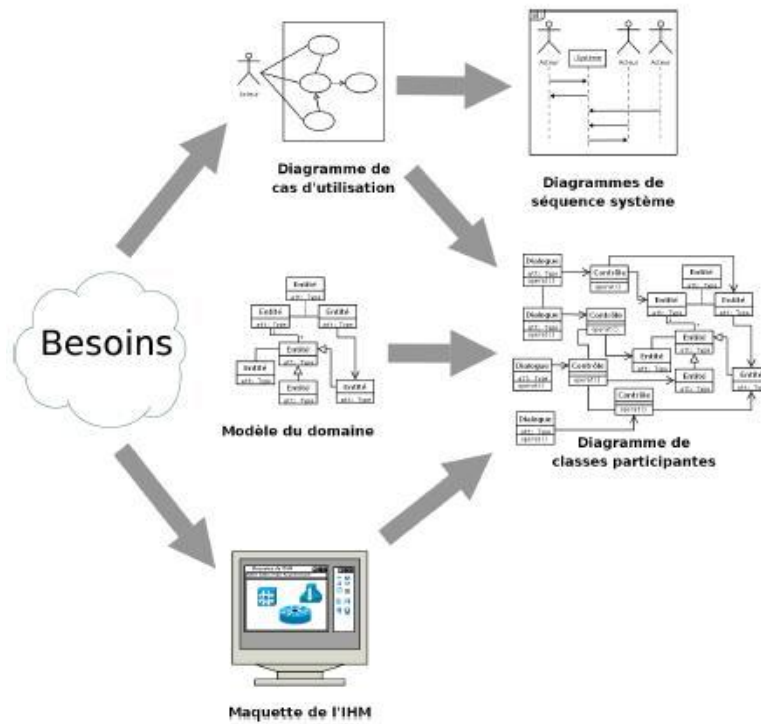


Figure 2.7 : Le diagramme de classes participantes effectue la jonction entre les cas d'utilisation, le modèle du domaine et les diagrammes de conception logicielle. [12]

➤ diagramme d'activité de navigation :

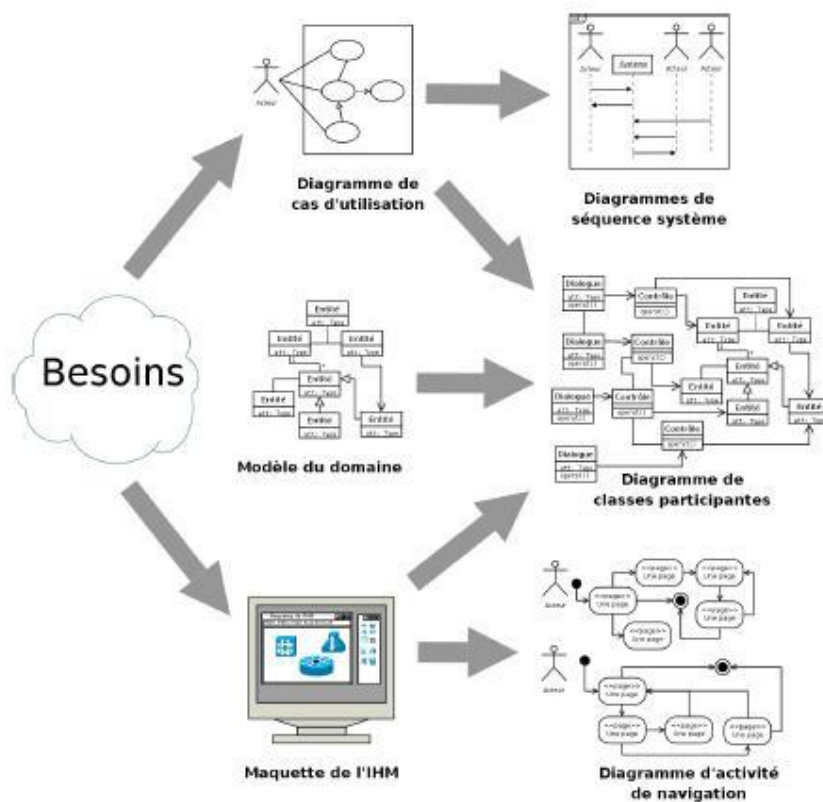


Figure 2.8 Les diagrammes d'activités de navigation représentent graphiquement l'activité de navigation dans l'IHM. [12]

### 2.4.3 Phase de conception

➤ **diagramme d'interaction :**

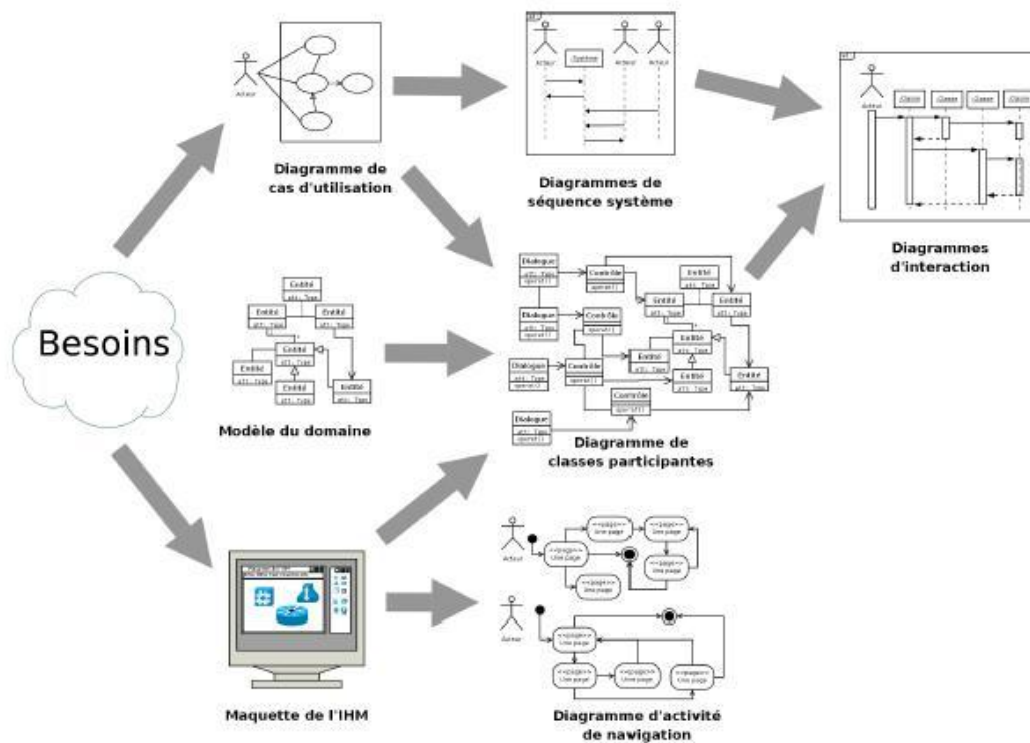


Figure 2.9 : Les diagrammes d'interaction permettent d'attribuer précisément les responsabilités de comportement aux classes d'analyse. [12]

➤ **Diagramme de classe de conception**

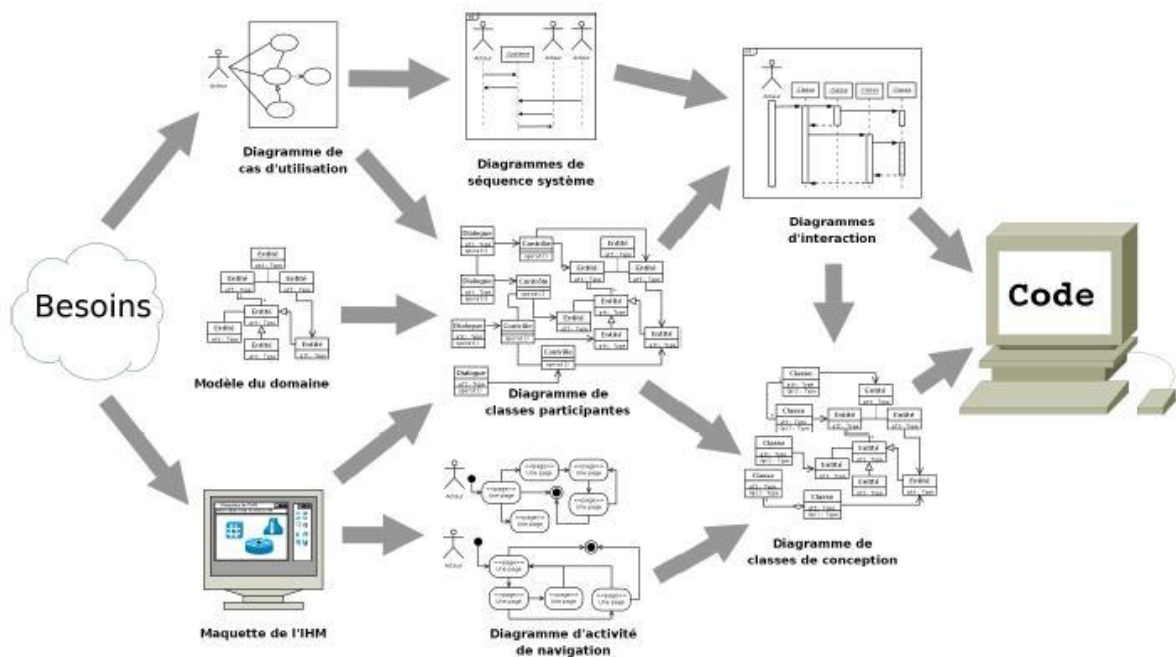


Figure 2.10 : Chaîne complète de la démarche de modélisation du besoin jusqu'au code. [12]

## 2.5 CONCLUSION :

UML est un moyen d'exprimer des modèles en faisant une abstraction de leur implémentation, i.e. que le modèle fourni par UML est valable pour n'importe quel langage de programmation. Il offre une manière de représenter le système selon différentes vues complémentaires grâce aux diagrammes.

Pour une pertinente conception UML doit suivre une démarche de développement. La démarche qu'on a suivi c'est UP (une approche simplifiée destinée pour les applications web).



**PARTIE 2**  
**LA CONCEPTION**  
**DE**  
**L'APPLICATION**



**CHAPITRE 03 :**  
**IDENTIFICATION**  
**DES BESOINS**

## 3.1 Introduction

Dans ce chapitre, nous allons présenter les différentes étapes conceptuelles de notre site web, alors, nous commencerons à identifier les *acteurs* qui interagiront avec le système, nous développerons un premier modèle UML de niveau *contexte*, pour pouvoir établir précisément les frontières fonctionnelles du système. Puis, nous identifions et nous décrivons les cas d'utilisation du système.

## 3.2 Identifier les acteurs

**Web master:** le personne responsable de créer des comptes utilisateurs (les comptes des enseignant et des étudiants) et d'enregistrés les dans la base de donnée, il est responsable aussi de maintenir le site.

**L'enseignant :** le personne qui participe dans le site selon le type d'utilisation « enseignant », il est responsable à mettre document, changer son mot de passe, s'identifier et participer.

**L'étudiant :** le personne qui participe dans le site selon le type d'utilisation « étudiant », il est responsable de télécharger les documents, de changer son mot de passe, de s'identifier et de participer.

**Visiteur :** l'internaute qui visite le site, il est responsable à télécharger et s'inscrire.



### 3.3 Diagramme de contexte

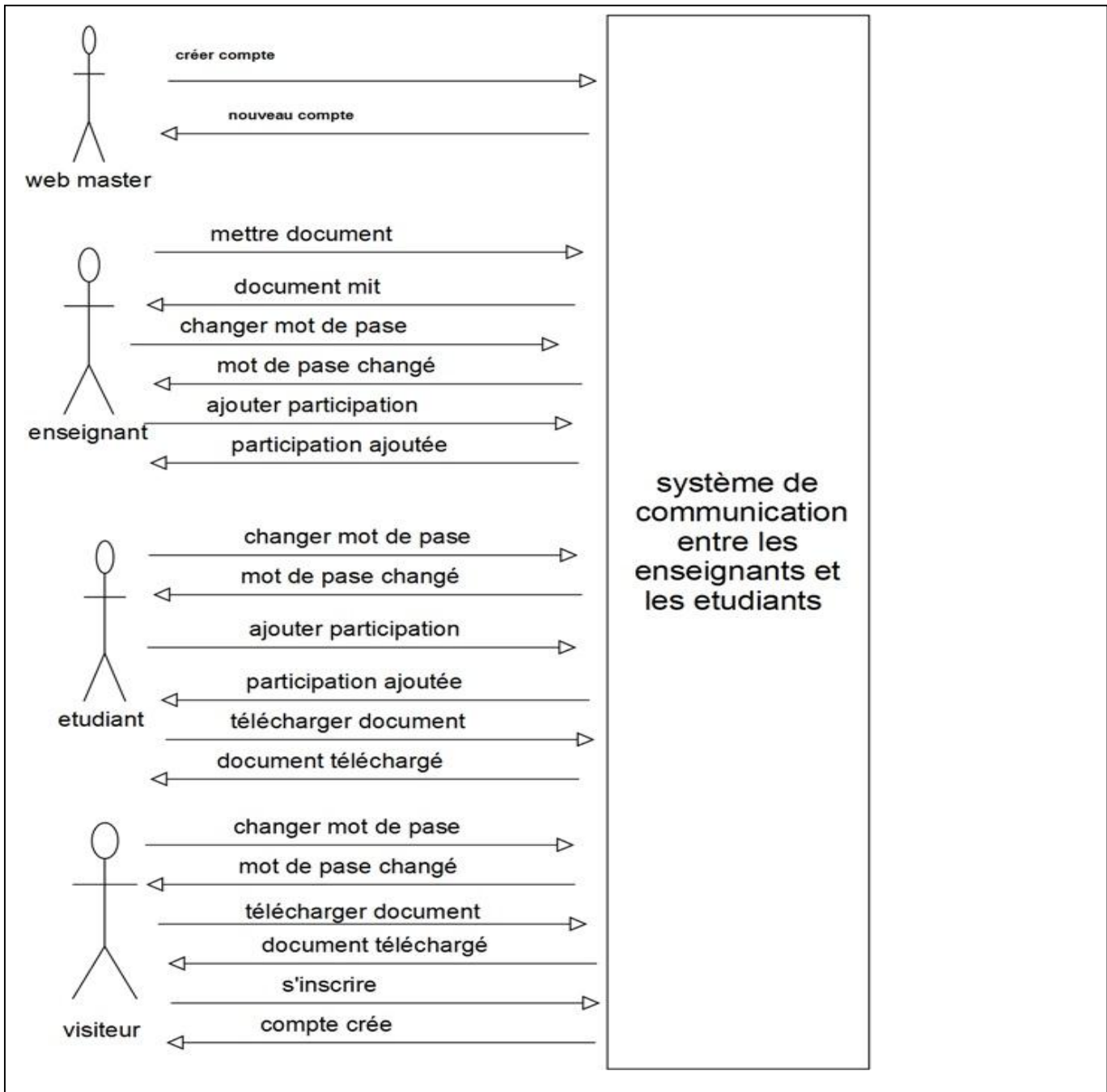


Figure 3.1: Diagramme de contexte

### 3.4 Diagramme de cas d'Utilisation

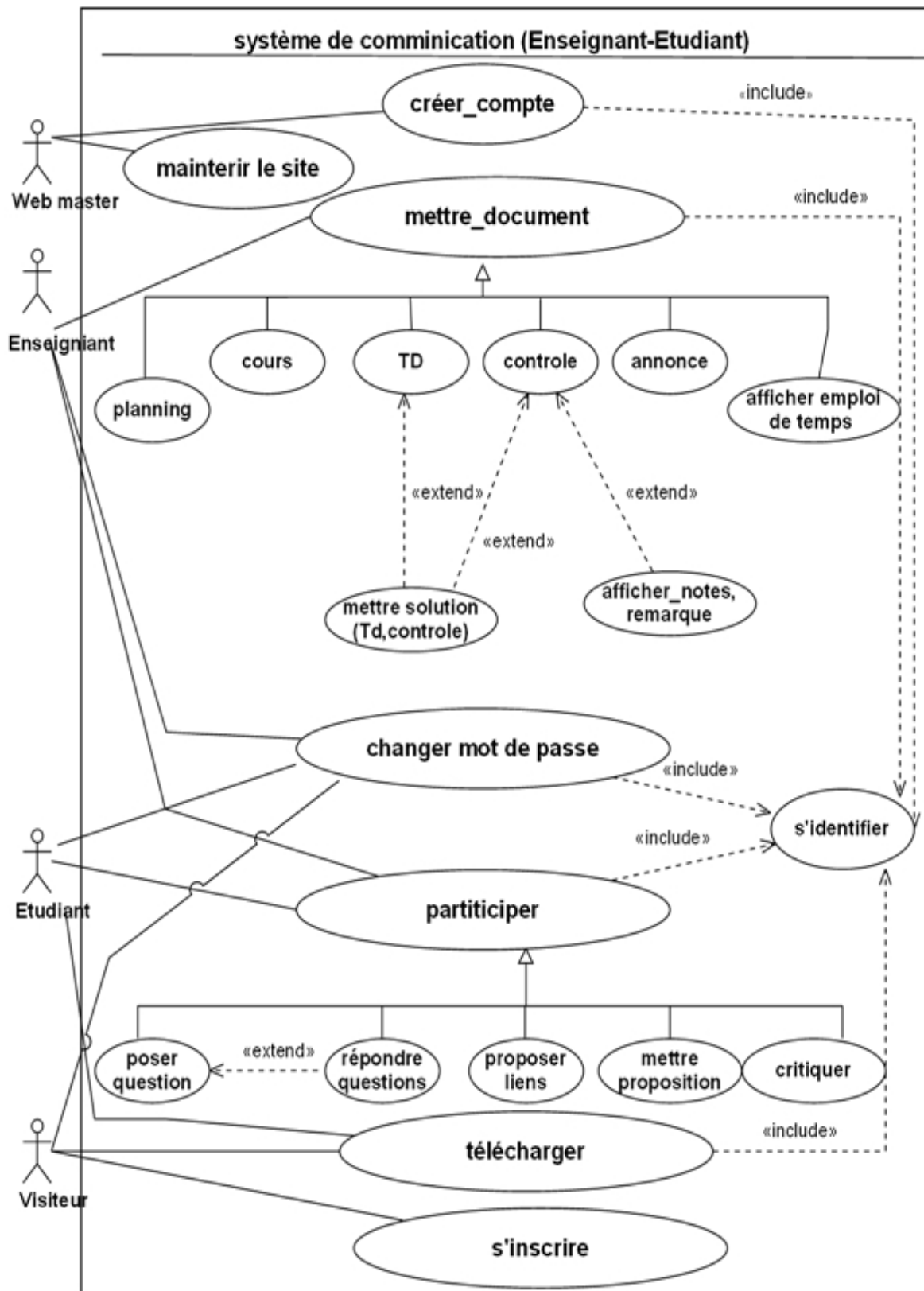


Figure 3.2: Diagramme de cas d'utilisation



## 3.5 Les fiches descriptives

### 3.5.1 Fiche descriptive du cas d'utilisation « créer compte »

Créer compte	
<b>Description</b>	Ce cas permet au web master de saisir les informations des étudiants et des enseignants dans la base de données.
<b>Acteur</b>	Web master
<b>Pré condition</b>	Le web master possède une liste d'étudiants et une autre d'enseignants
<b>Post condition</b>	Chaque enseignant ou étudiant a un compte pour visiter le site à n'importe quel moment.
<b>Nominal</b>	<p>Le web master remplit les informations de chaque étudiant et enseignant dans la base (nom, prénom, username, email, mot de passe, type d'utilisation).</p> <ol style="list-style-type: none"> <li>1. Le système enregistre ces informations dans la base de données.</li> </ol>
<b>Alternatif</b>	
<b>Exception</b>	

### 3.5.2 Fiche descriptive du cas d'utilisateur « s'identifier »

S'identifier	
<b>description</b>	Ce cas d'utilisation permet à l'étudiant, l'enseignant, visiteur de s'identifier pour avoir accès à leur espace.
<b>Acteur</b>	Etudiant, enseignant, visiteur.
<b>Pré condition</b>	Chaque utilisateur possède un compte.
<b>Post condition</b>	Le système est connu l'utilisateur et son type d'utilisation.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. Le système demande à l'étudiant ou l'enseignant ou visiteur de saisir l'username et le mot de passe.</li> <li>2. L'utilisateur saisit ses données.</li> <li>3. Le système vérifie les données de l'utilisateur et les valide.</li> <li>4. le système permet à l'utilisateur d'ouvrir son compte et entrer dans le site.</li> </ol>
<b>Alternatif</b>	L'username ou le mot de passe est erroné : retour au scénario nominal <b>1</b> .
<b>Exception</b>	L'utilisateur quitte le système.

### 3.5.3 Fiche descriptive du cas d'utilisation «mettre\_ document»

Mettre document	
<b>Description</b>	ce cas d'utilisation permet à n'importe quel enseignant de mettre des cours, des TD, des annonces, í dans chaque module.
<b>Acteurs</b>	Enseignant.
<b>pré condition</b>	l'enseignant a un (des) document(s) (cours, TD, annonces, í ) et il veut le(s) mettre dans le site.  l'enseignant est déjà s'identifié.
<b>Post condition</b>	le système affiche le message « votre document est met en sucée »
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. l'enseignant demande de mettre un (des) documents</li> <li>2. le système permet à l'enseignant de mettre le(s)document(s) dans le module qu'il est enseigné.</li> <li>3. L'enseignant met son document dans la zone de «mettre document » et il validé.</li> </ol>
<b>Alternatif</b>	<ol style="list-style-type: none"> <li>1. L'enseignant annule la mise des documents : revenir au scénario nominal 1.</li> <li>2. L'espace de la zone insuffisant pour le document :retour au scénario nominal 1.</li> </ol>
<b>Exception</b>	L'enseignant quitte le système

### 3.5.4 Fiche descriptive du cas d'utilisation «changer mot de passe»

Changer mot de passe	
<b>Description</b>	Ce cas permet à n'importe quel étudiant ou enseignant ou visiteur de changer son mot de passe.
<b>Acteur</b>	Enseignant, Etudiant, Visiteur.
<b>Pré condition</b>	L'utilisateur est déjà s'identifié.
<b>Post condition</b>	L'utilisateur a un compte avec un nouveau mot de passe.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur demande de changer son mot de passe.</li> <li>2. Le système permet à l'utilisateur de changer son mot de passe avec des conditions de la saisie bien sur.</li> <li>3. L'utilisateur saisit son nouveau mot de passe et le valide.</li> <li>4. Le système affiche que le mot de passe est changé.</li> </ol>

<b>Alternatif</b>	Le nouveau mot de passe est incorrect : retour au scénario nominal <b>1</b> .
-------------------	---

### 3.5.5 Fiche descriptive du cas d'utilisation « participer »

Participer	
<b>description</b>	Ce cas permet à un étudiant ou enseignant de participer dans le site (poser des questions, mettre des propositions, critiquer, proposer des liens d'études, répondre les questions, etc.).
<b>Acteur</b>	Étudiant, enseignant
<b>Pré condition</b>	L'utilisateur est déjà s'identifié.
<b>Post condition</b>	Les participations d'étudiant et d'enseignant seront affichées dans la zone de participation.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur demande d'ajouter une participation.</li> <li>2. Le système permet à l'utilisateur de mettre une participation.</li> <li>3. L'utilisateur écrit sa participation dans la zone et la valide.</li> <li>4. Le système confirme la validation de participation.</li> </ol>
<b>Alternatif</b>	L'utilisateur n'a pas validé sa participation : revenir au point <b>1</b> du scénario nominal.
<b>Exception</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur annule la participation et quitte le système.</li> <li>2. le type d'utilisation est visiteur alors le système affiche un message d'erreur.</li> </ol>

### 3.5.6 Fiche descriptive du cas d'utilisation « télécharger »

télécharger	
<b>description</b>	Ce cas permet à un étudiant ou un visiteur de télécharger tout ce qu'il veut
<b>Acteur</b>	Etudiant, visiteur
<b>Pré condition</b>	L'utilisateur est déjà s'identifié.
<b>Post condition</b>	Le système affiche un message le téléchargement est terminé
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur demande de télécharger un document.</li> <li>2. le système permet à l'utilisateur de télécharger un document.</li> <li>3. L'étudiant(ou le visiteur) sélectionne le module qu'il intéresse et ensuite, il télécharge le.</li> </ol>

<b>Alternatif</b>	L'étudiant(ou le visiteur) annule le téléchargement d'un document et veut télécharger un autre document (retour au scénario nominal).
-------------------	---

### 3.5.7 Fiche descriptive du cas d'utilisateur « s'inscrire »

S'inscrire	
<b>description</b>	Ce cas permet à n'importe quel visiteur d'inscrire dans le site.
<b>Acteur</b>	Visiteur.
<b>Pré condition</b>	
<b>Post condition</b>	Le système affiche des messages de confirmation de l'inscription.
<b>Nominal</b>	<ol style="list-style-type: none"> <li>1. le visiteur veut s'inscrire dans le site.</li> <li>2. le système donne le formulaire d'inscription au visiteur</li> <li>4. le visiteur saisit ses informations</li> <li>5. Le système affiche que l'inscription est réussite.</li> </ol>
<b>Alternatif</b>	<ol style="list-style-type: none"> <li>1. Les informations entrantes sont existées déjà (retour au scénario nominal 2).</li> <li>2. Erreur d'information (retour au scénario nominal 2)</li> </ol>
<b>Exception</b>	Le visiteur choisit de quitter le système.

## 3.6 Les diagrammes de séquence système

### 3.6.1 Créer compte

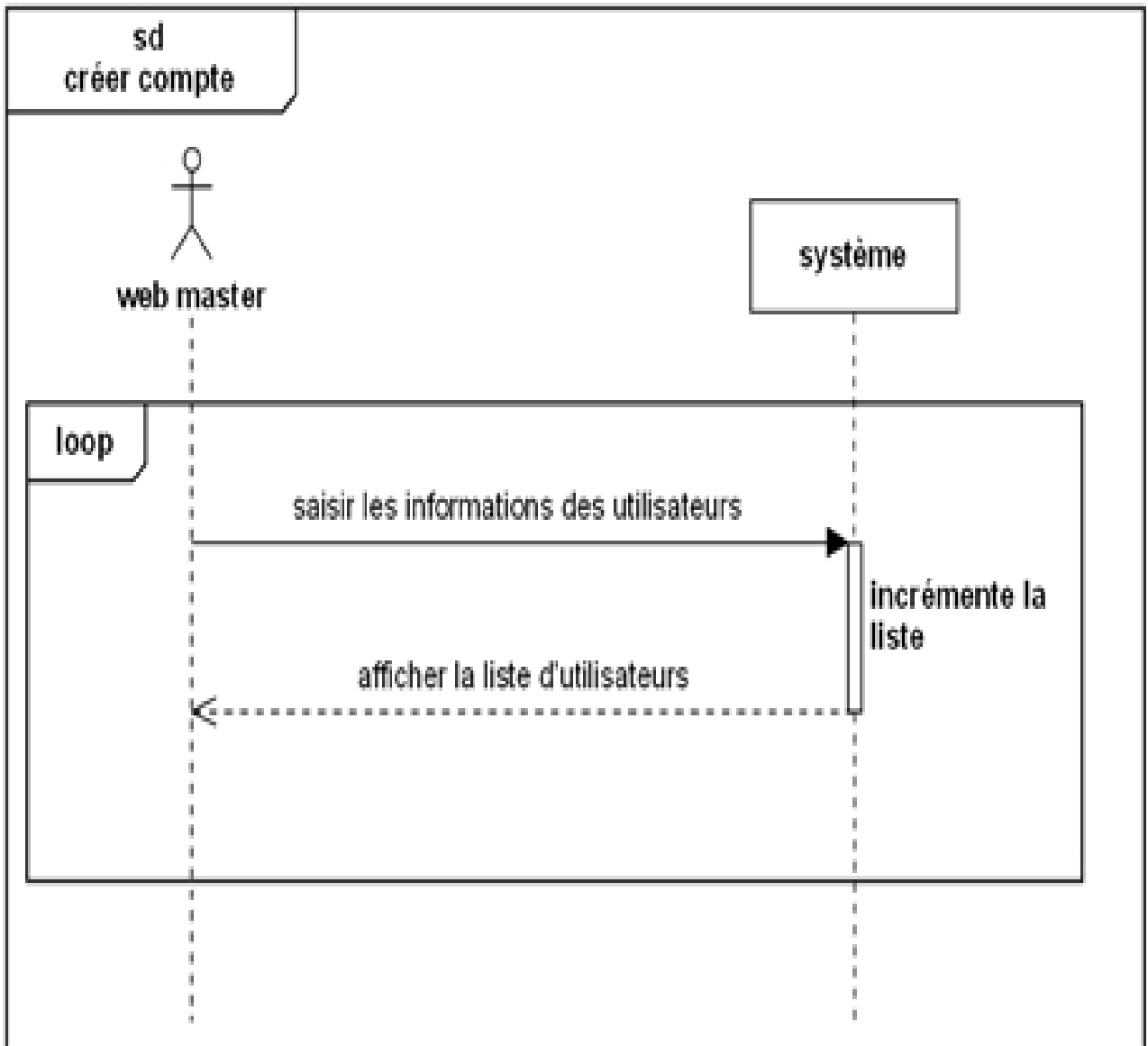


Figure 3.3: Diagramme de séquence système cas d'utilisation « créer compte »

### 3.6.2 SD Identifier

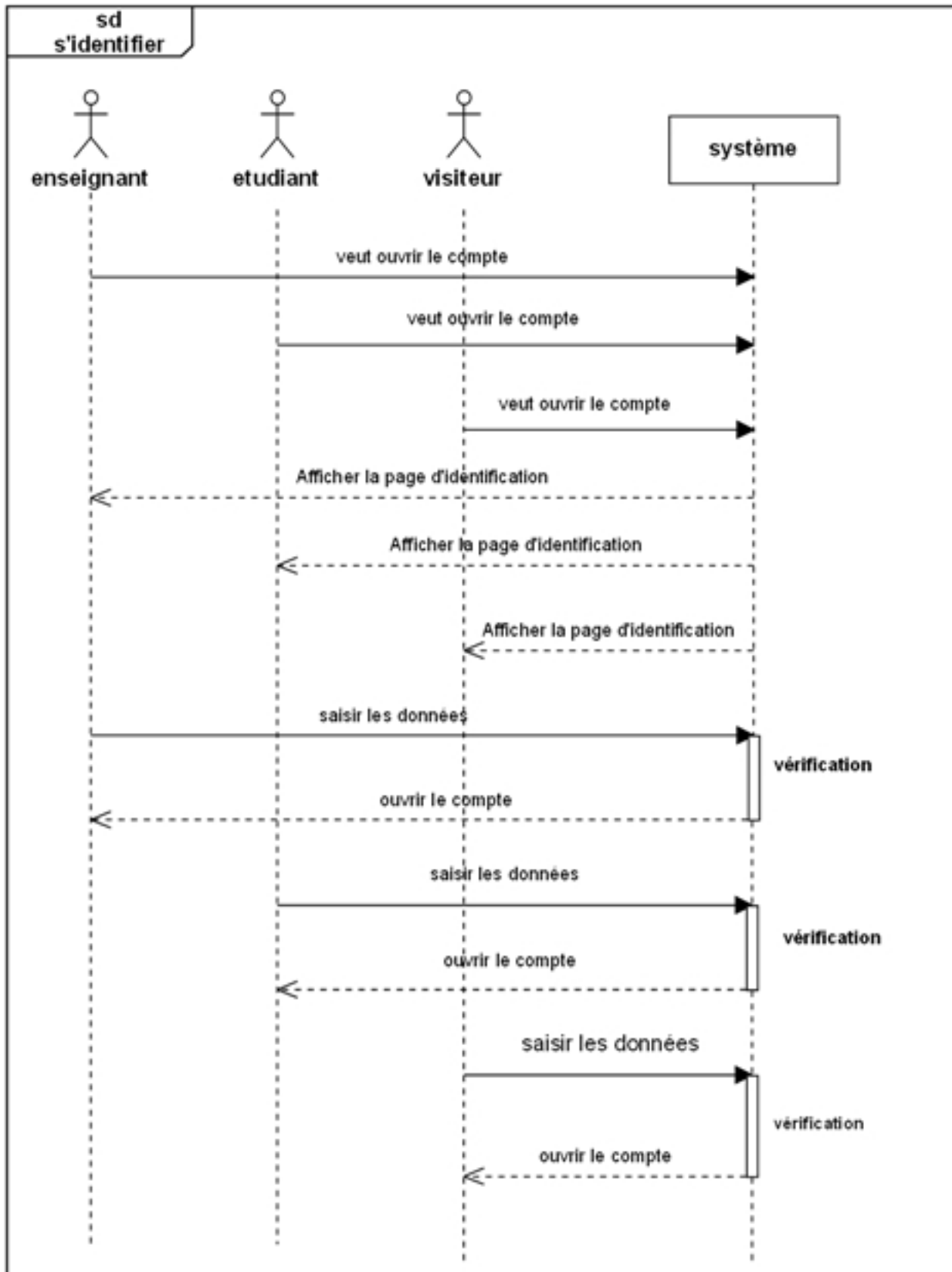


Figure 3.4: Diagramme de séquence système cas d'utilisation « s'identifier »

### 3.6.3 Mettre document

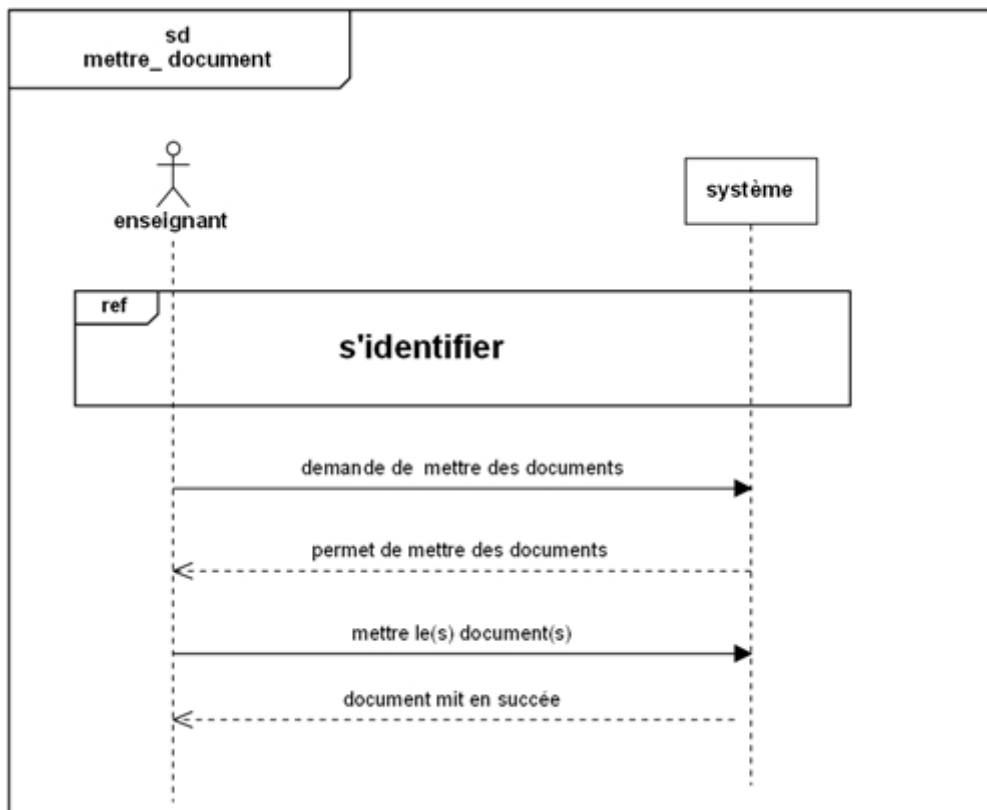


Figure 3.5: Diagramme de séquence système cas d'utilisation « Mettre document »

### 3.6.4 Participer

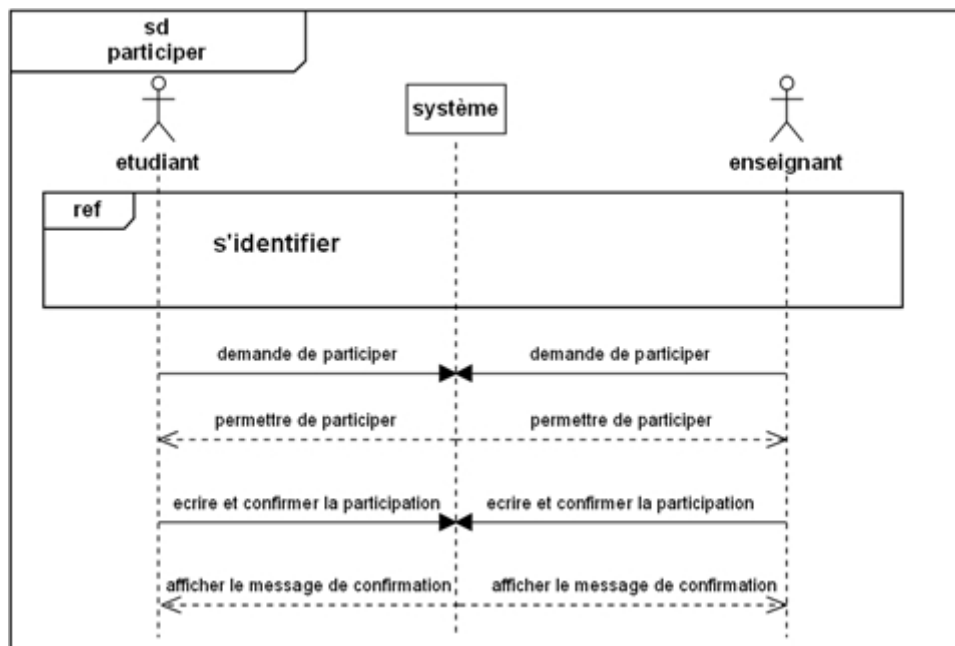


Figure 3.6: Diagramme de séquence système cas d'utilisation « Participer »

### 3.6.5 Changer mot de passe

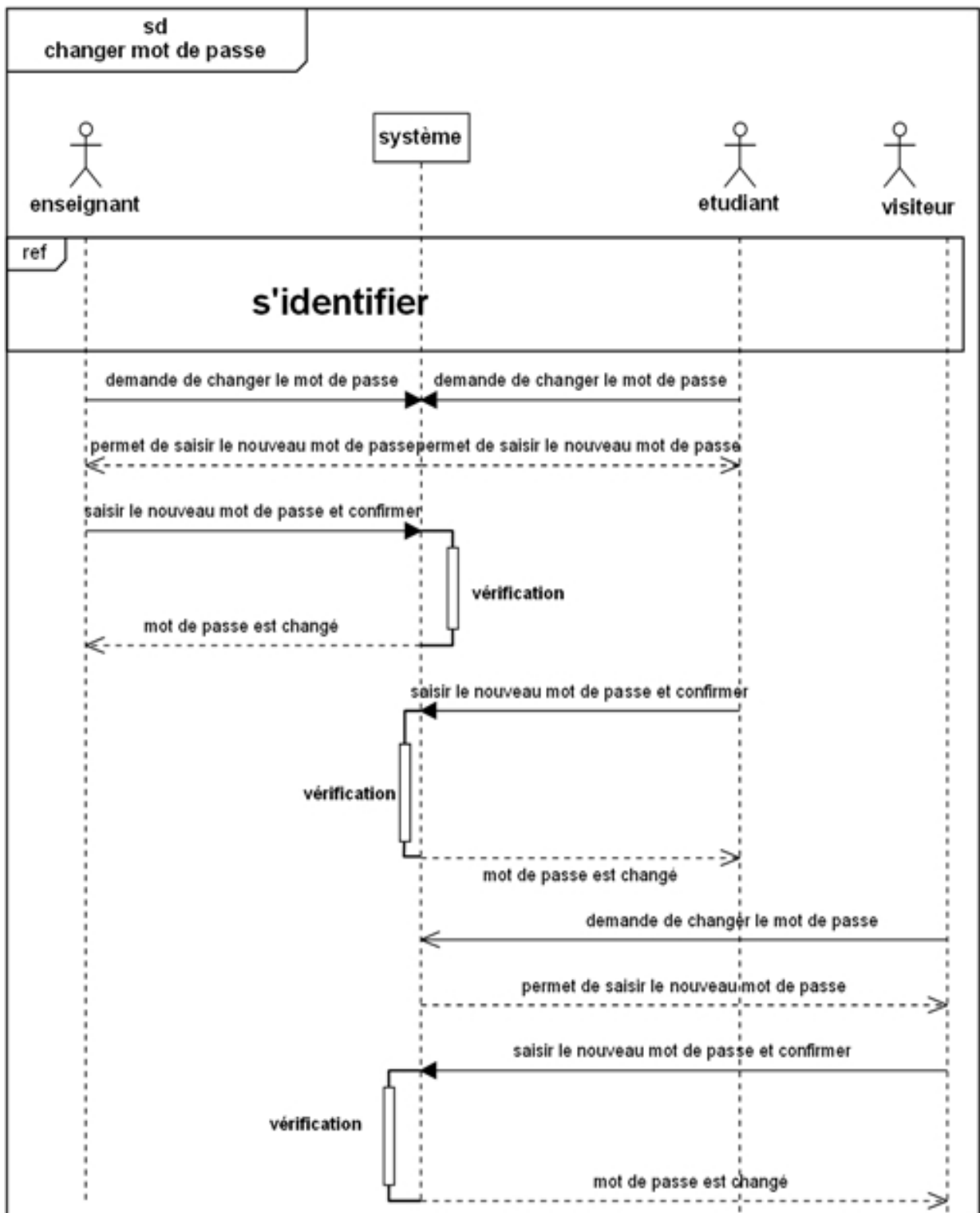


Figure 3.7: Diagramme de séquence système cas d'utilisation «Changer mot de passe »



### 3.6.6 Télécharger

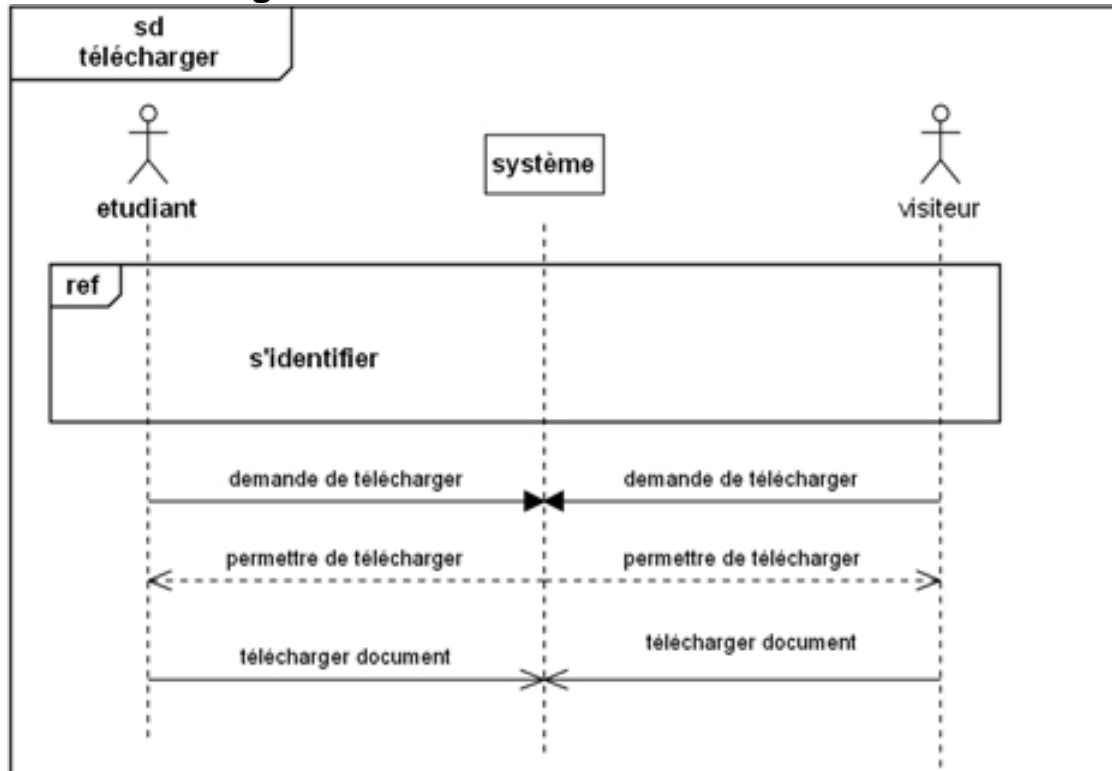


Figure 3.8: Diagramme de séquence système cas d'utilisation « Télécharger »

### 3.6.7 S'inscrire

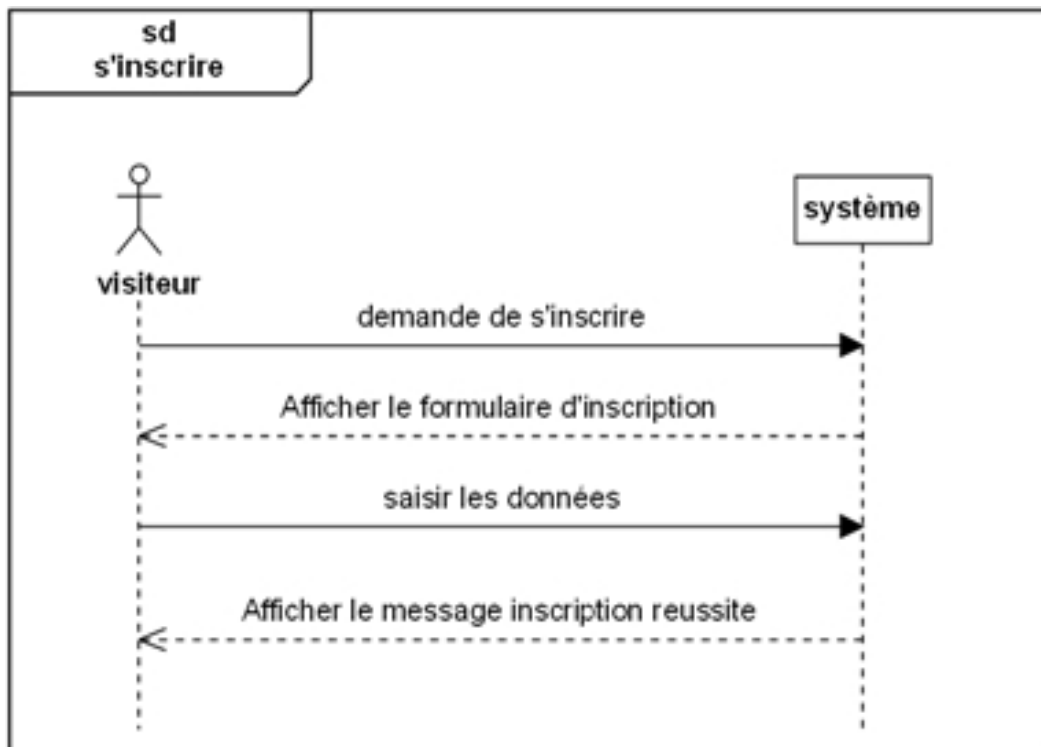


Figure 3.9: Diagramme de séquence système cas d'utilisation « s'inscrire »

## 3.7 Conclusion

La phase d'identification des besoins nous a permis de modéliser le contexte du système considéré comme une boîte noire en:

- Répertoriant les interactions entre acteurs et système.
- Représentant les interactions sur le diagramme de contexte dynamique.

Ainsi l'identification de besoin nous a permis aussi de modéliser les cas d'utilisations et les diagrammes de séquences système en prenant en considération que le système a toujours vu comme une boîte noire.

Ensuite cette phase prépare la phase d'analyse qui est l'objet de chapitre suivant.



**CHAPITRE 04 :**  
**PHASE**  
**D'ANALYSE**

## 4.1 Introduction

Dans ce chapitre nous allons élaborer du diagramme de classes qu'est le modèle de classes du domaine. Puis, Nous développerons les diagrammes de classes participantes qui effectuent la jonction entre, d'une part, les cas d'utilisation, le modèle du domaine et la maquette, et d'autre part, les diagrammes de conception logicielle que sont les diagrammes de séquence et le diagramme de classes de conception, enfin, nous représenterons l'activité de navigation dans l'interface du future application en produisant des diagrammes d'activités de navigation.

## 4.2 Analyse du domaine

### 4.2.1 Démarche

La conception objet demande principalement une description structurelle, statique, du système à réaliser, sous forme d'un ensemble de classes logicielles. Les classes candidats sont celle issues d'une analyse de domaine, effectivement ces concepts peuvent être identifiés directement à partir de la connaissance du domaine. Détaillons les différentes étapes de la démarche que nous allons suivre :

Identifier les concepts du domaine ;

Ajouter les associations entre classes ;

Ajouter les attributs.

#### 4.2.1.1 Identification des concepts de domaine

Nous allons prendre les cas d'utilisations.

❖ Créer compte :

- Web master

❖ Mettre document :

- Enseignant

❖ Changer mot de passe :

- Enseignant
- Etudiant
- Visiteur

❖ Identifier :

- Enseignant

- Etudiant
- Visiteur
- ❖ Participer :
  - Enseignant
  - Etudiant
- ❖ Télécharger :
  - Etudiant
  - Visiteur
- ❖ S'inscrire :
  - Visiteur

4.2.1.2 Modèle de domaine

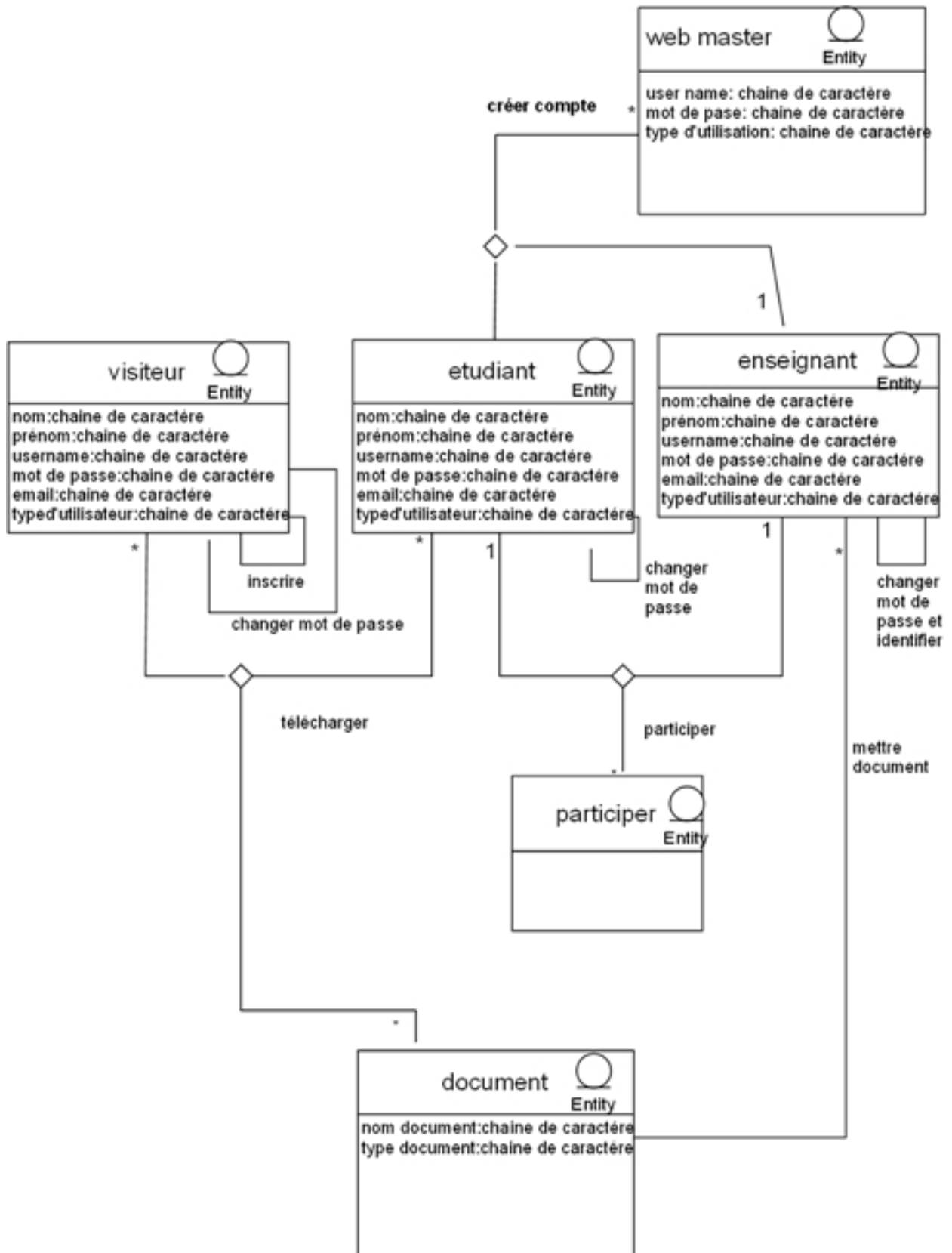


Figure 4.1 Diagramme de domaine

## 4.2.2 Les diagrammes de classes participantes

### 4.2.2.1 Mettre document

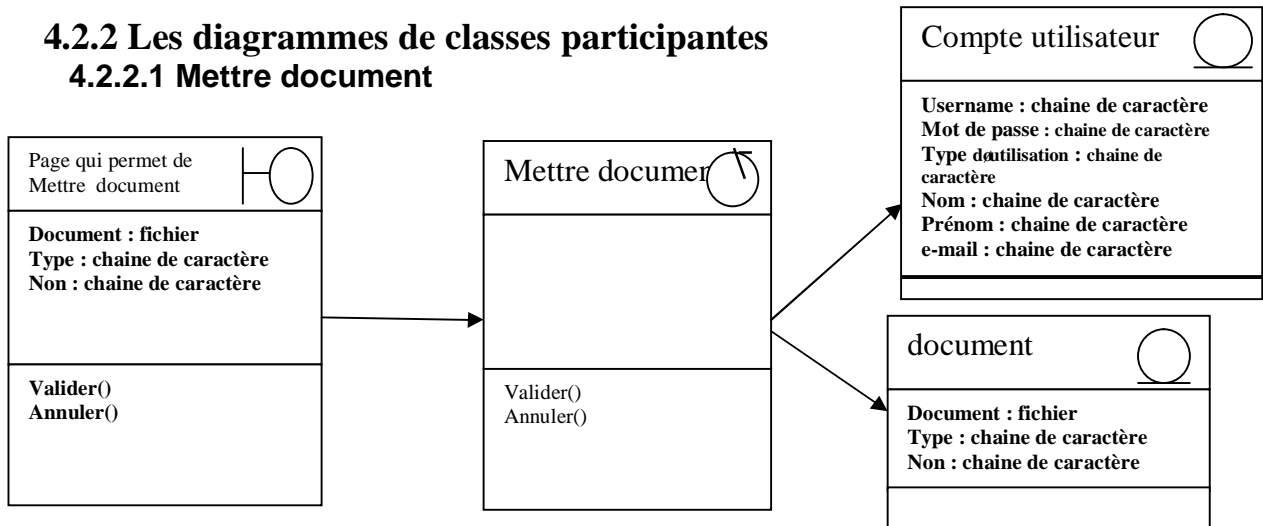


Figure 4.2 Diagramme de classe participante « Mettre document »

### 4.2.2.2 Télécharger

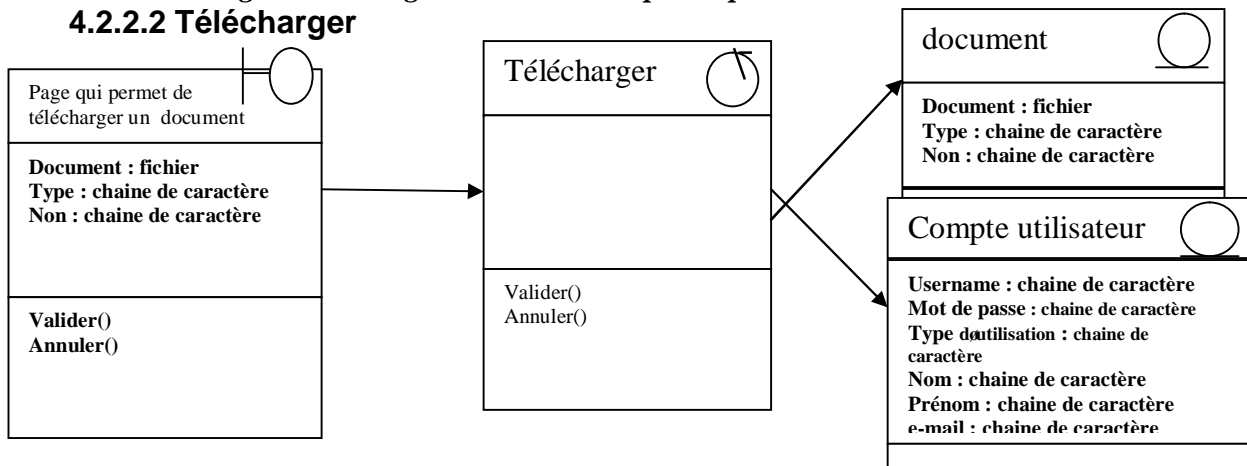


Figure 4.3 Diagramme de classe participante «Télécharger »

### 4.2.2.3 Participer

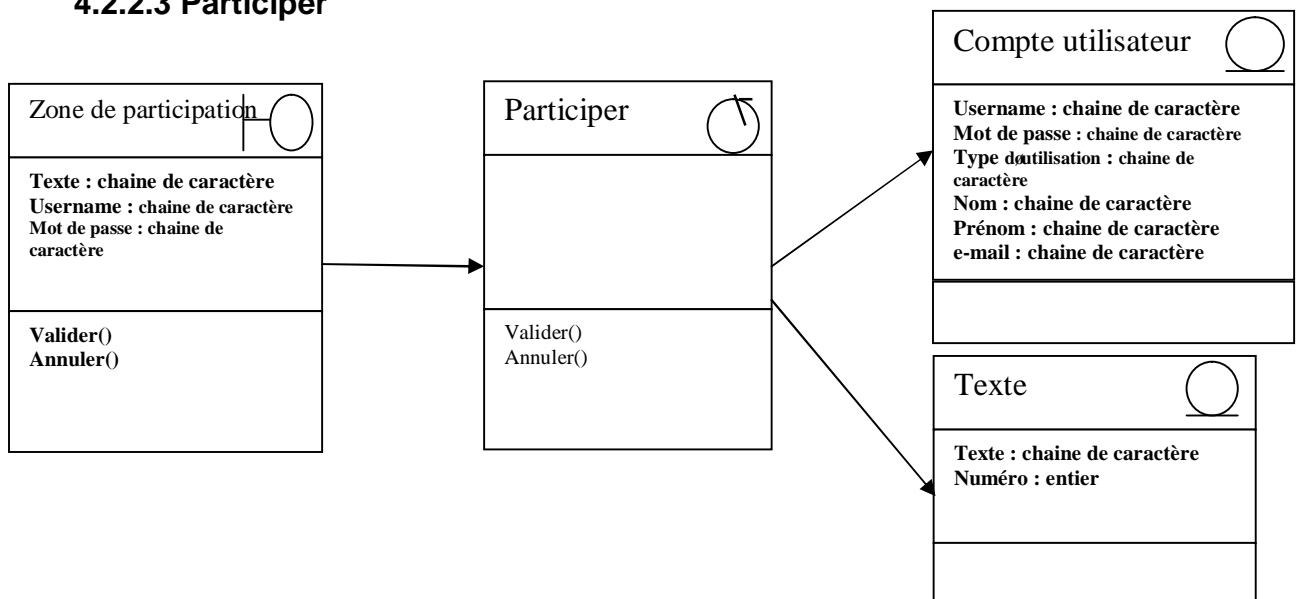


Figure 4.4 Diagramme de classe participante «Participer »

#### 4.2.2.4 Changer mot de passe

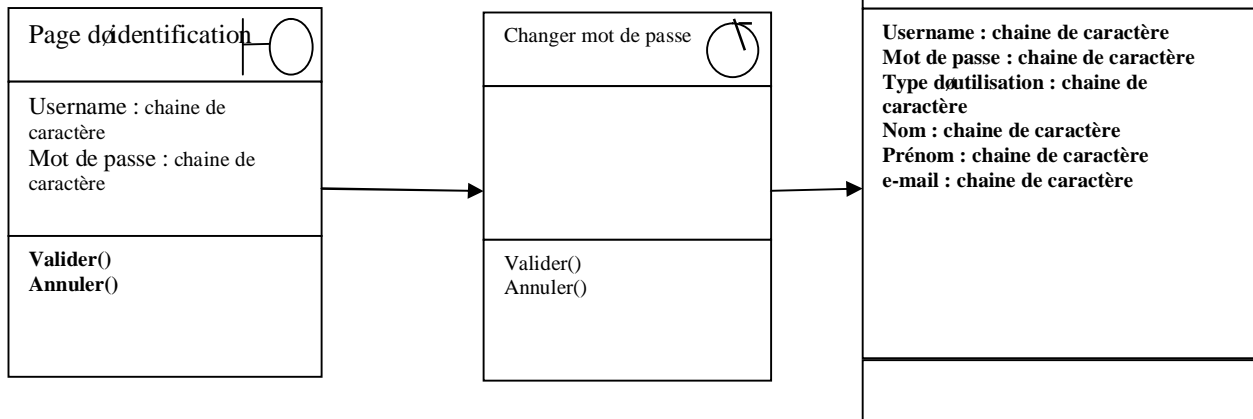


Figure 4.5 Diagramme de classe participante «Changer mot de passe »

#### 4.2.2.5 s'identifier

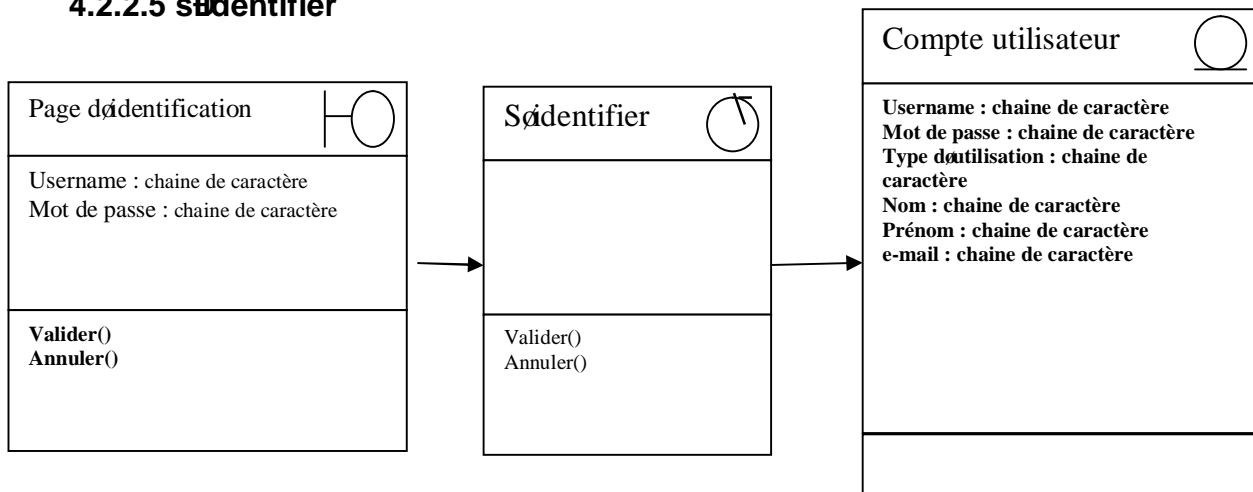


Figure 4.6 Diagramme de classe participante «s'identifier »

#### 4.2.2.6 créer compte

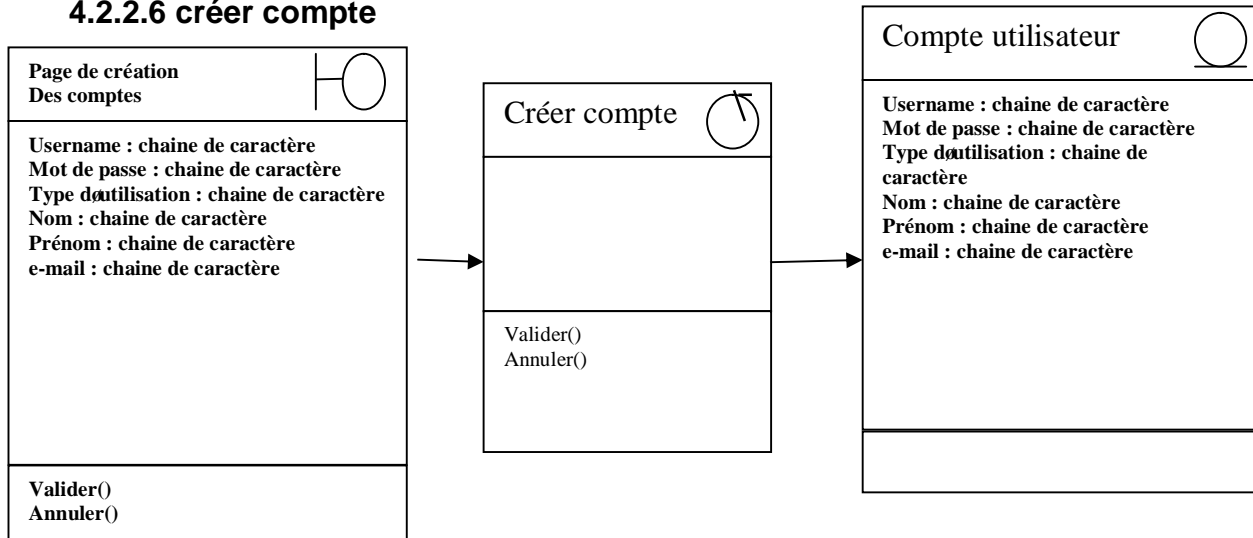


Figure 4.7 Diagramme de classe participante «créer compte »



### 4.2.2.7 sInscrire

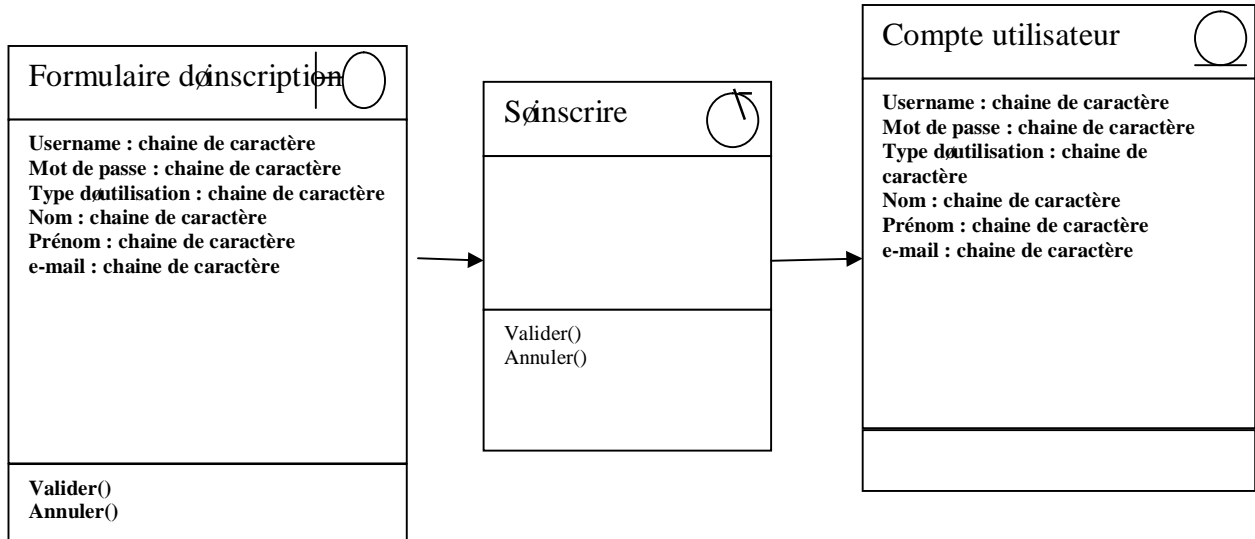


Figure 4.8 Diagramme de classe participante «sInscrire »

## 4.2.3 Les diagrammes d'activités de navigation

### 4.2.3.1 Créer compte

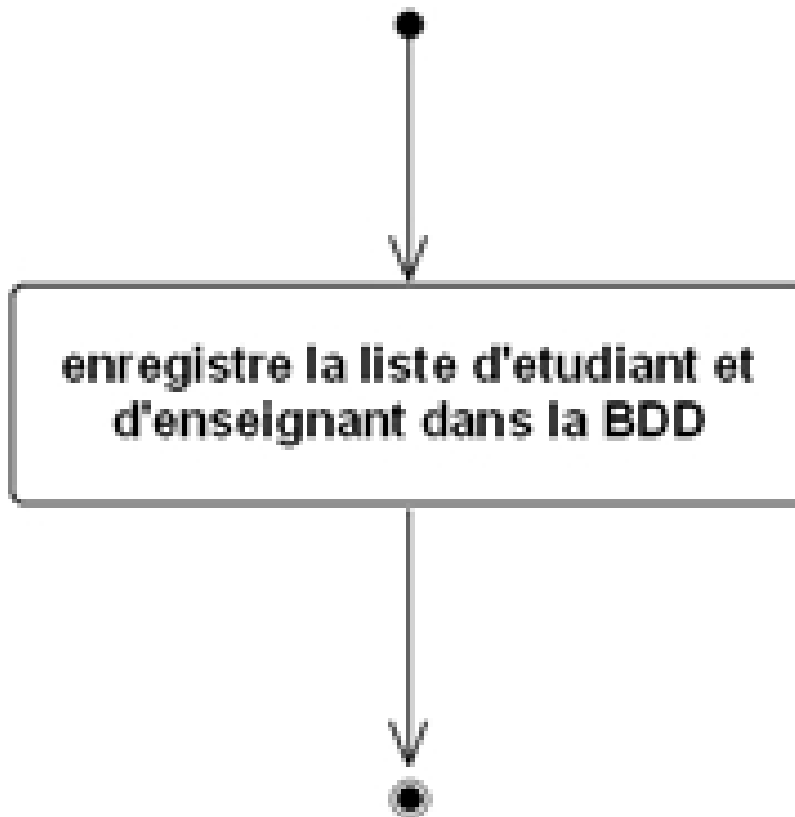


Figure 4.9 Diagramme d'activité de navigation «Créer compte »

### 4.2.3.2 SIdentifier

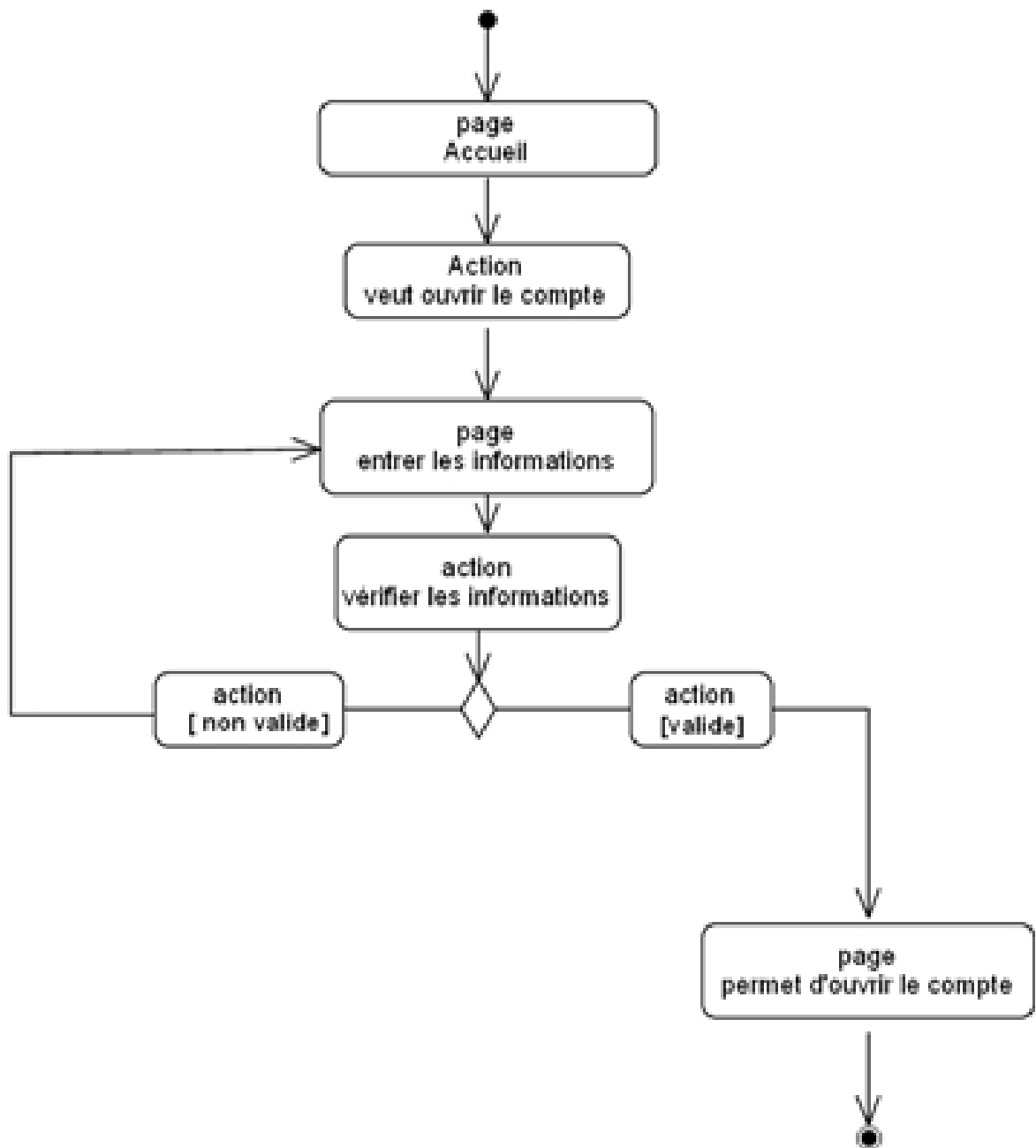


Figure 4.10 Diagramme d'activité de navigation «sIdentifier »

4.2.3.3 Mettre document

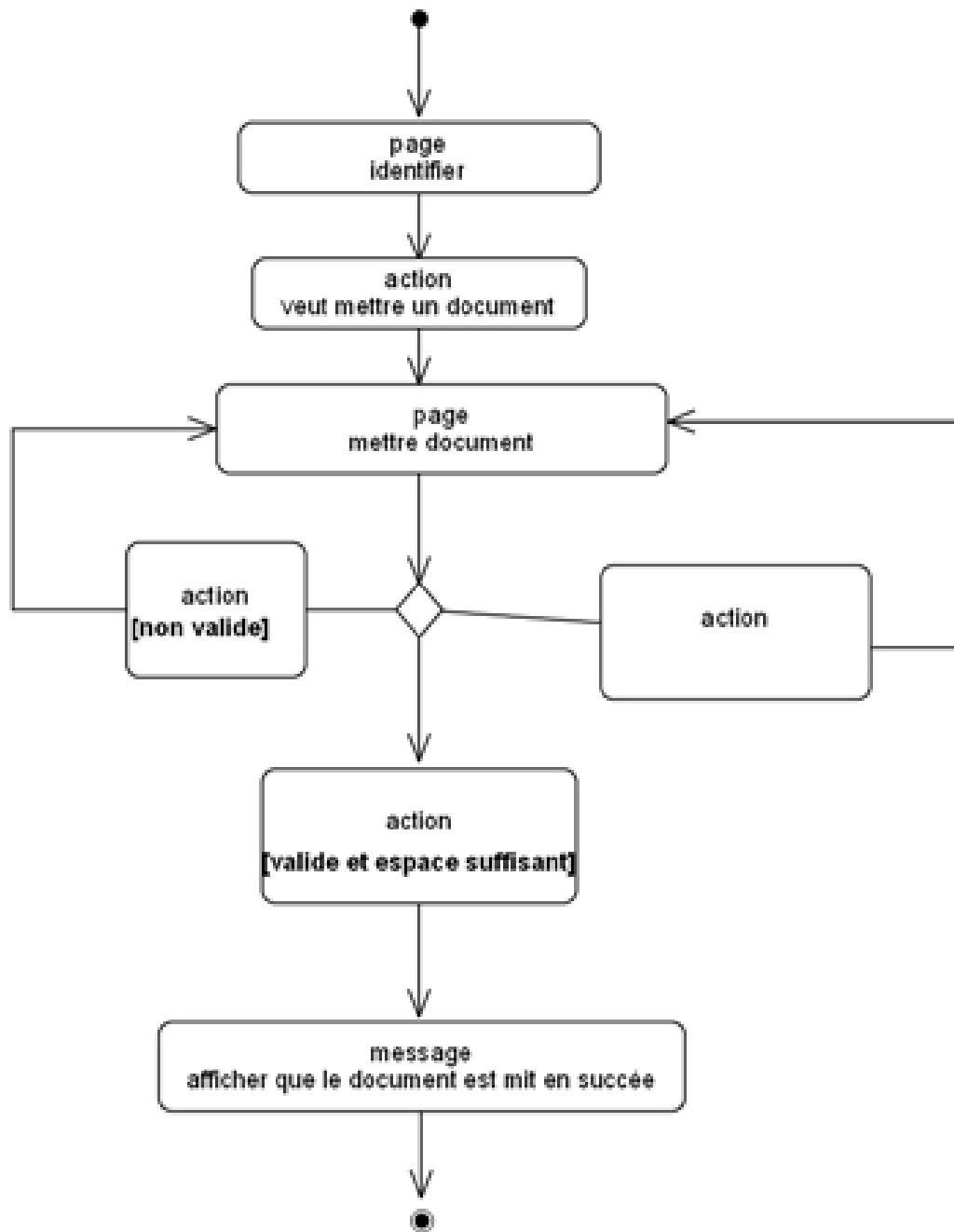


Figure 4.11 Diagramme d'activité de navigation «mettre document »

#### 4.2.3.4 Participer

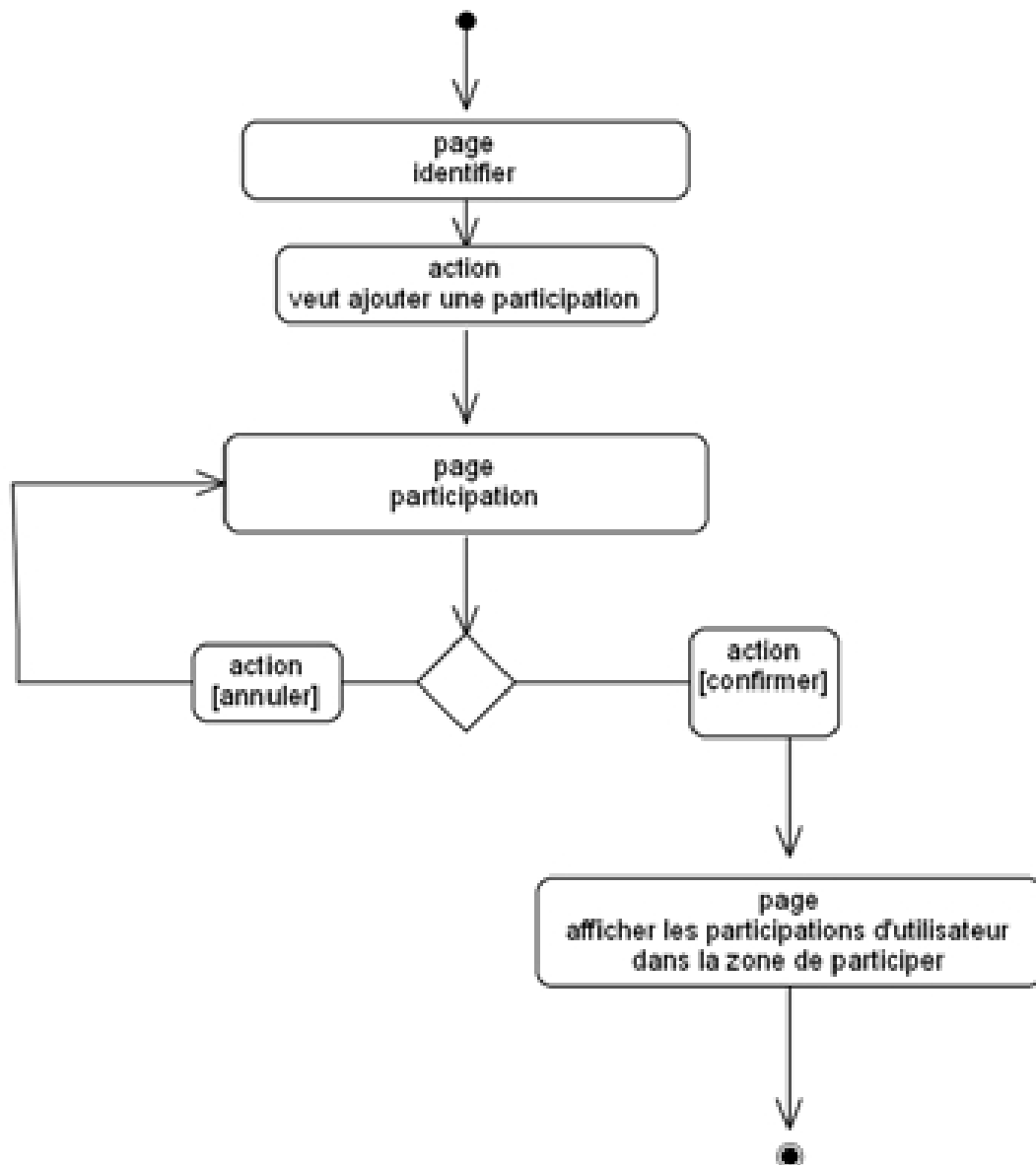


Figure 4.12 Diagramme d'activité de navigation «Participer »

## 4.2.3.5 Changer mot de passe

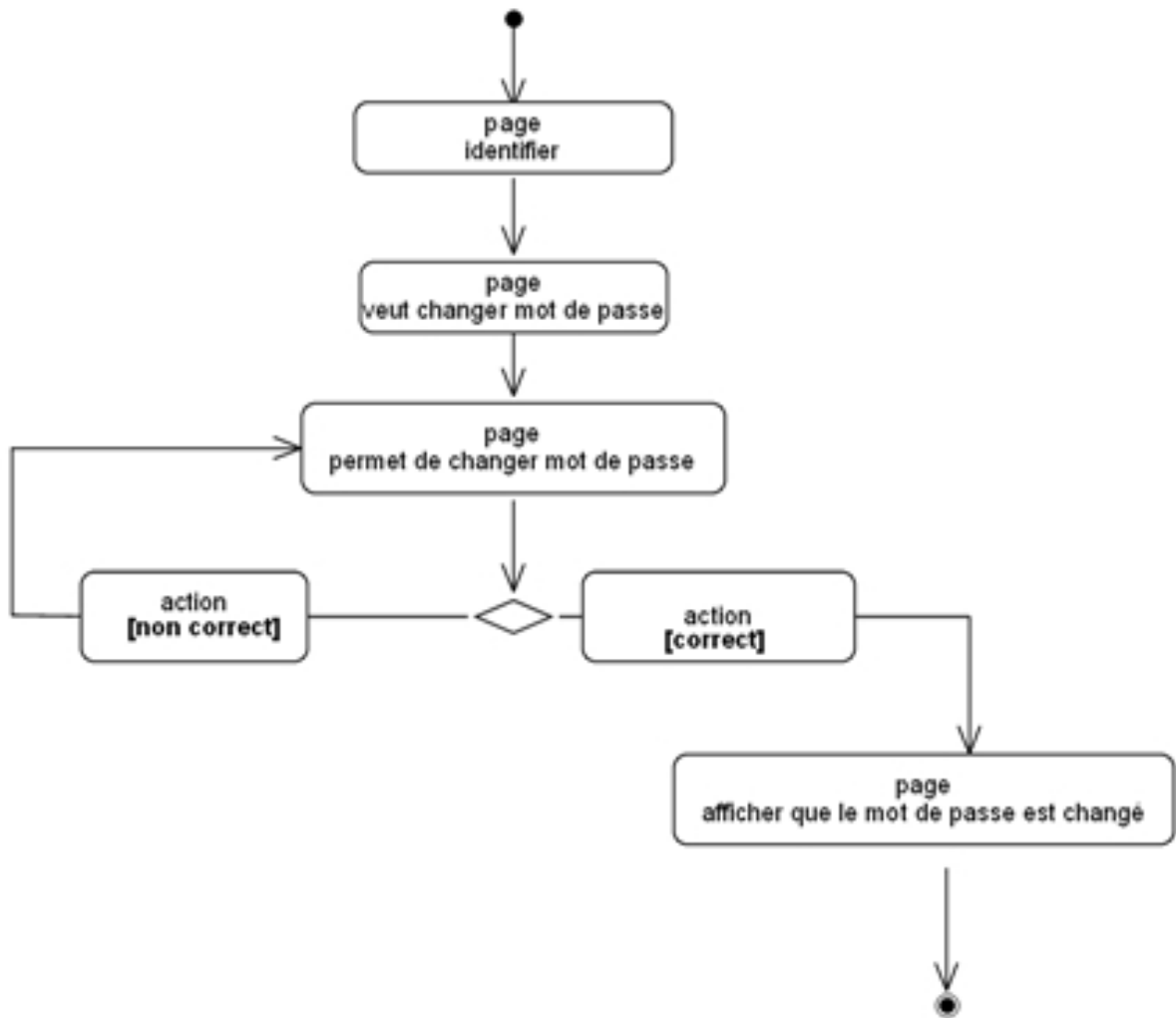


Figure 4.13 Diagramme d'activité de navigation «Changer mot de passe »

### 4.2.3.6 Télécharger

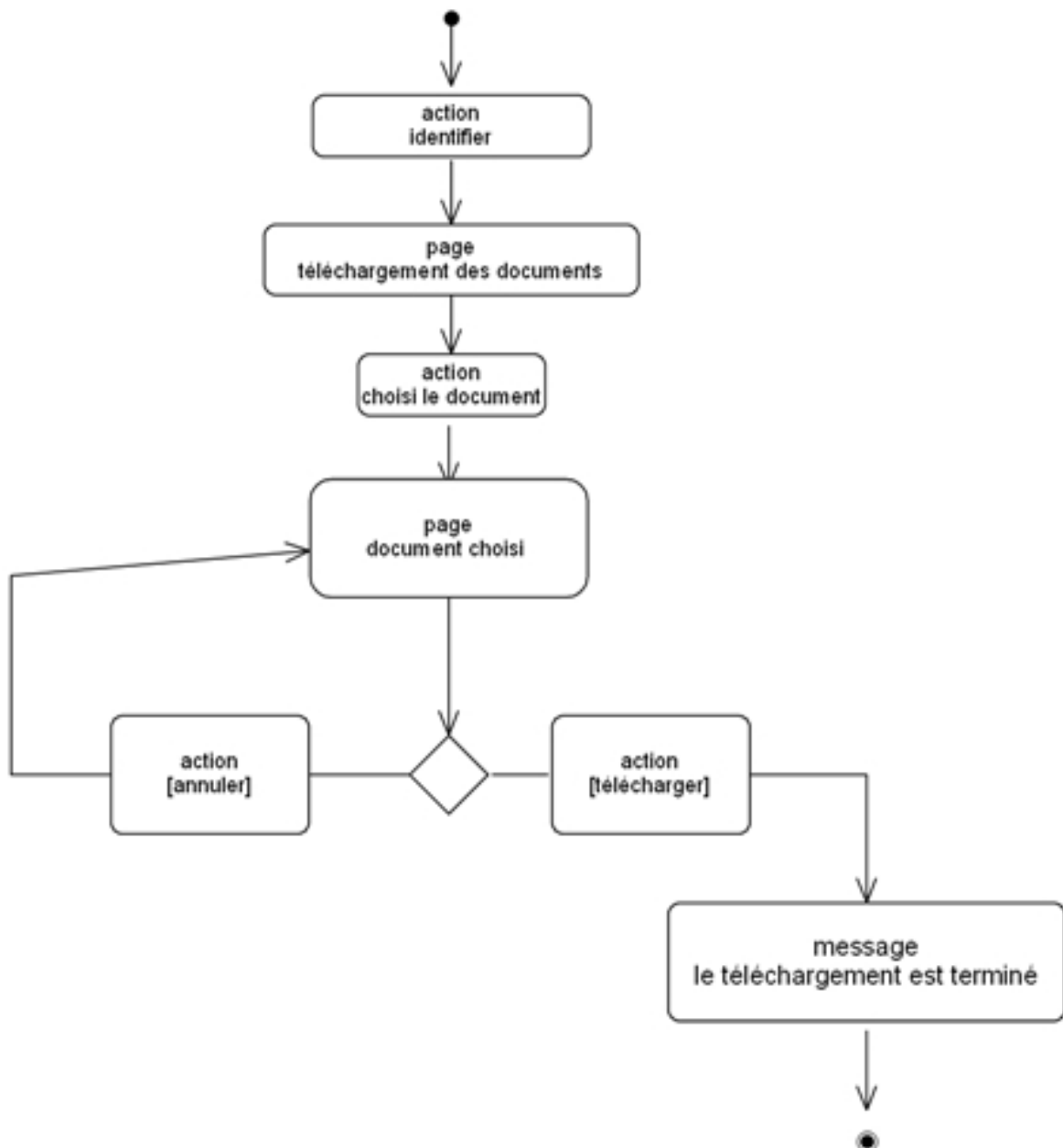


Figure 4.14 Diagramme d'activité de navigation «Télécharger »

#### 4.2.3.7 S'inscrire

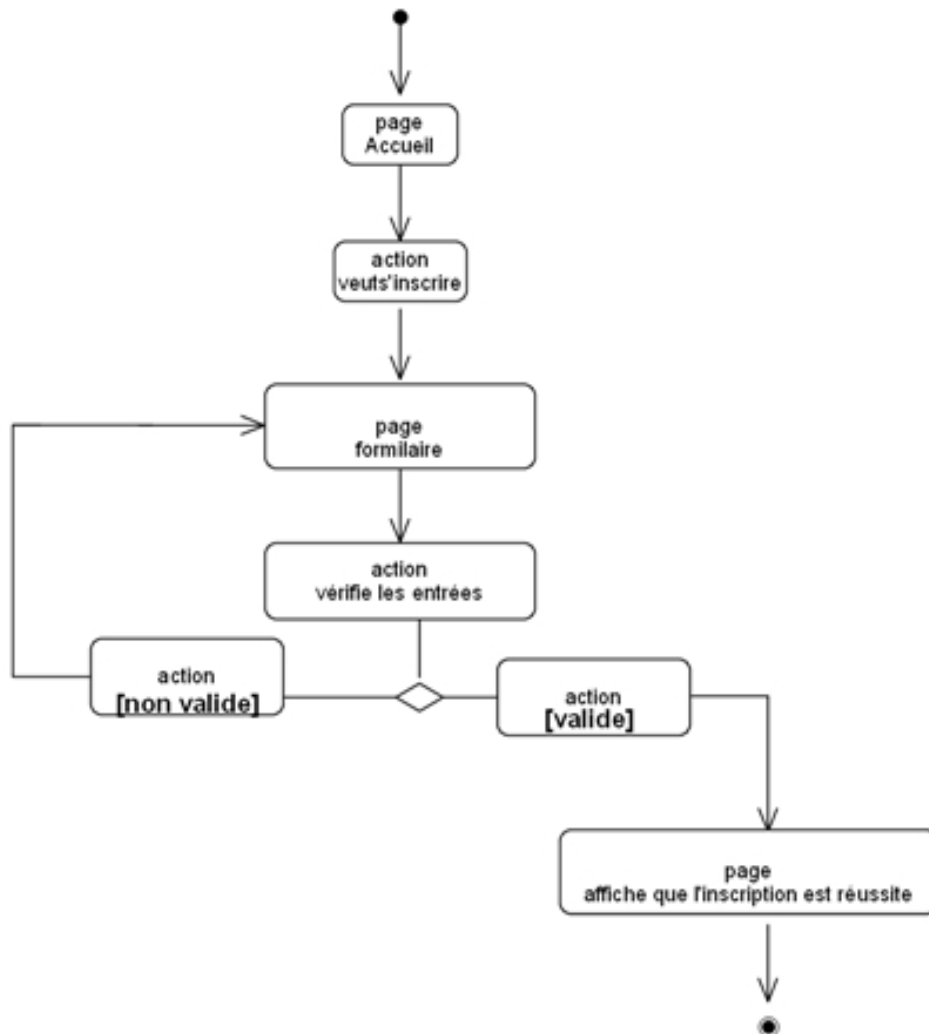


Figure 4.15 Diagramme d'activité de navigation «s'inscrire»

### 4.3 Conclusion

La phase d'analyse nous a permis de modéliser les concepts de domaine en:

- Représentant les différentes entités de domaine.
- Représentant les relations entre les classes d'entités, les classes de contrôles et les classes de dialogues.

Ainsi la phase d'analyse nous a permis aussi de modéliser les navigations sur l'interface via les diagrammes d'activités de navigation.

Ensuite cette phase prépare la phase de conception qui est l'objet de chapitre suivant.



**CHAPITRE 05 :**  
**PHASE DE**  
**CONCEPTION**



## 5.1 Introduction

Dans ce chapitre nous allons élaborer les diagrammes d'interactions qui nous permet d'attribuer précisément les responsabilités de comportement aux classes d'analyse du diagramme de classes participantes. Parallèlement, une première ébauche de la vue statique de conception, c'est-à-dire du diagramme de classes de conception, est construite et complétée.

## 5.2 Les diagrammes d'Interaction

### 5.2.1 Créer compte

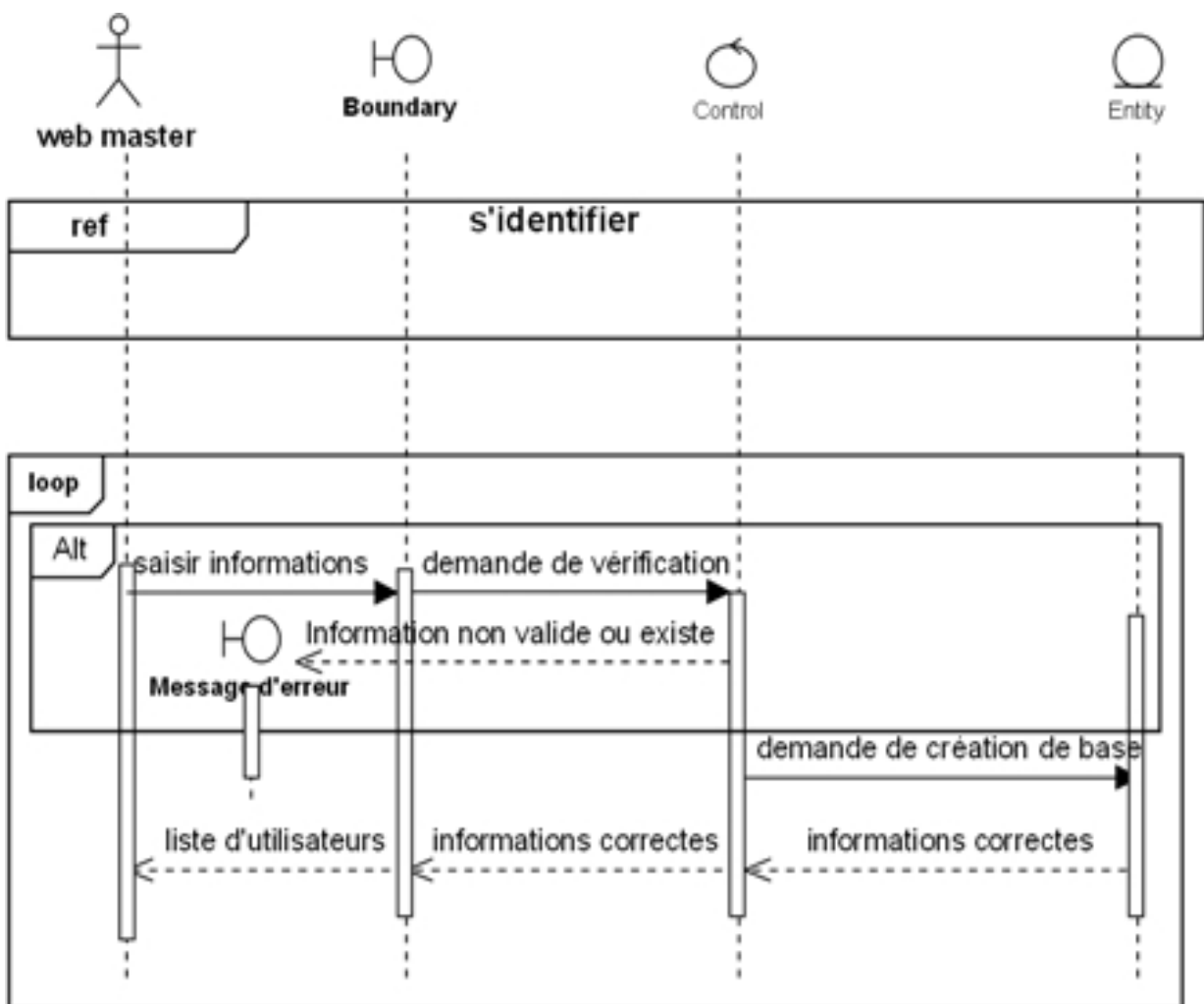


Figure 5.1 : diagramme d'interaction « créer compte »

### 5.2.2 SD Identifier

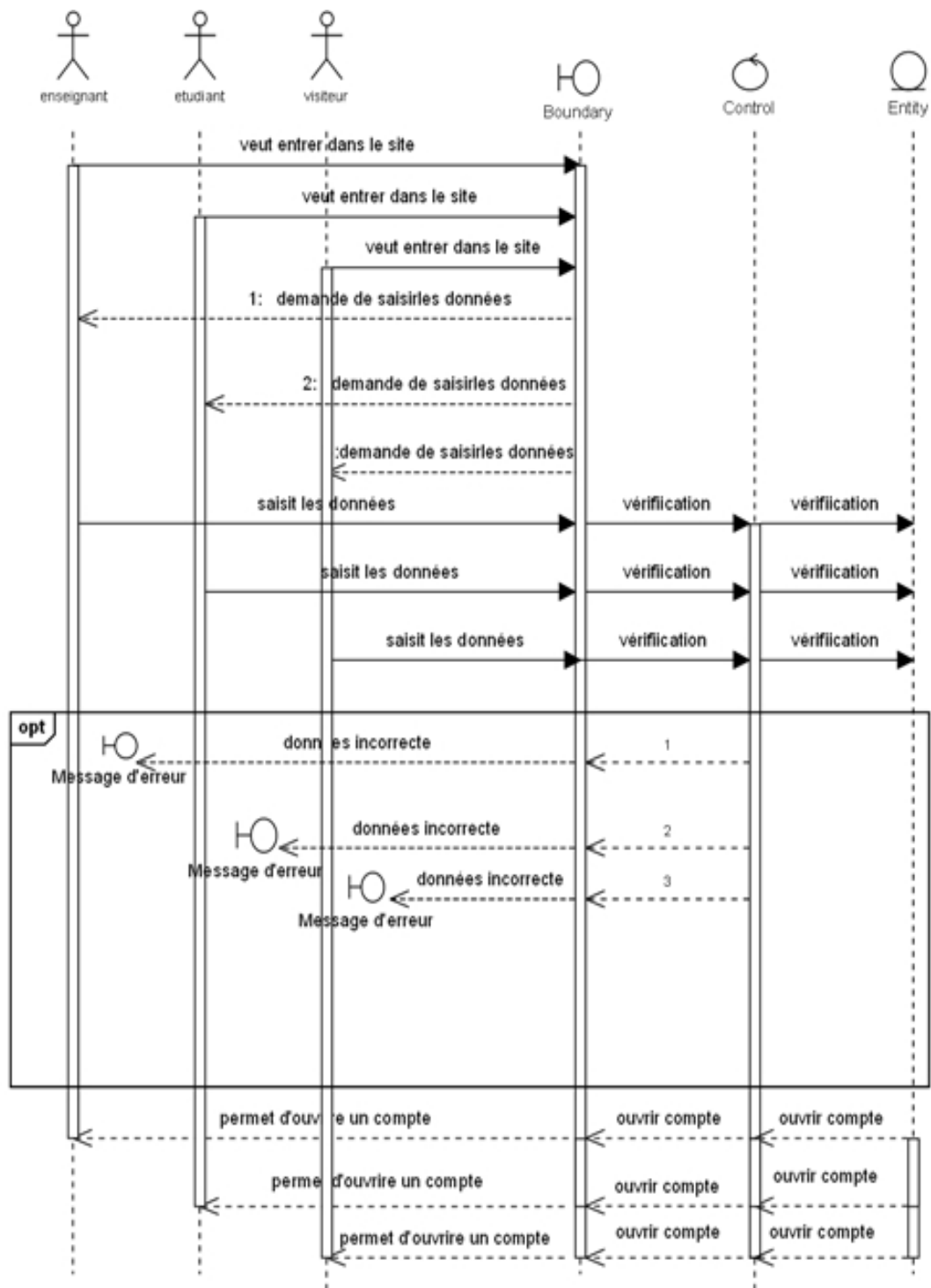


Figure 5.2 : diagramme d'interaction « s'identifier »

### 5.2.3 Mettre document

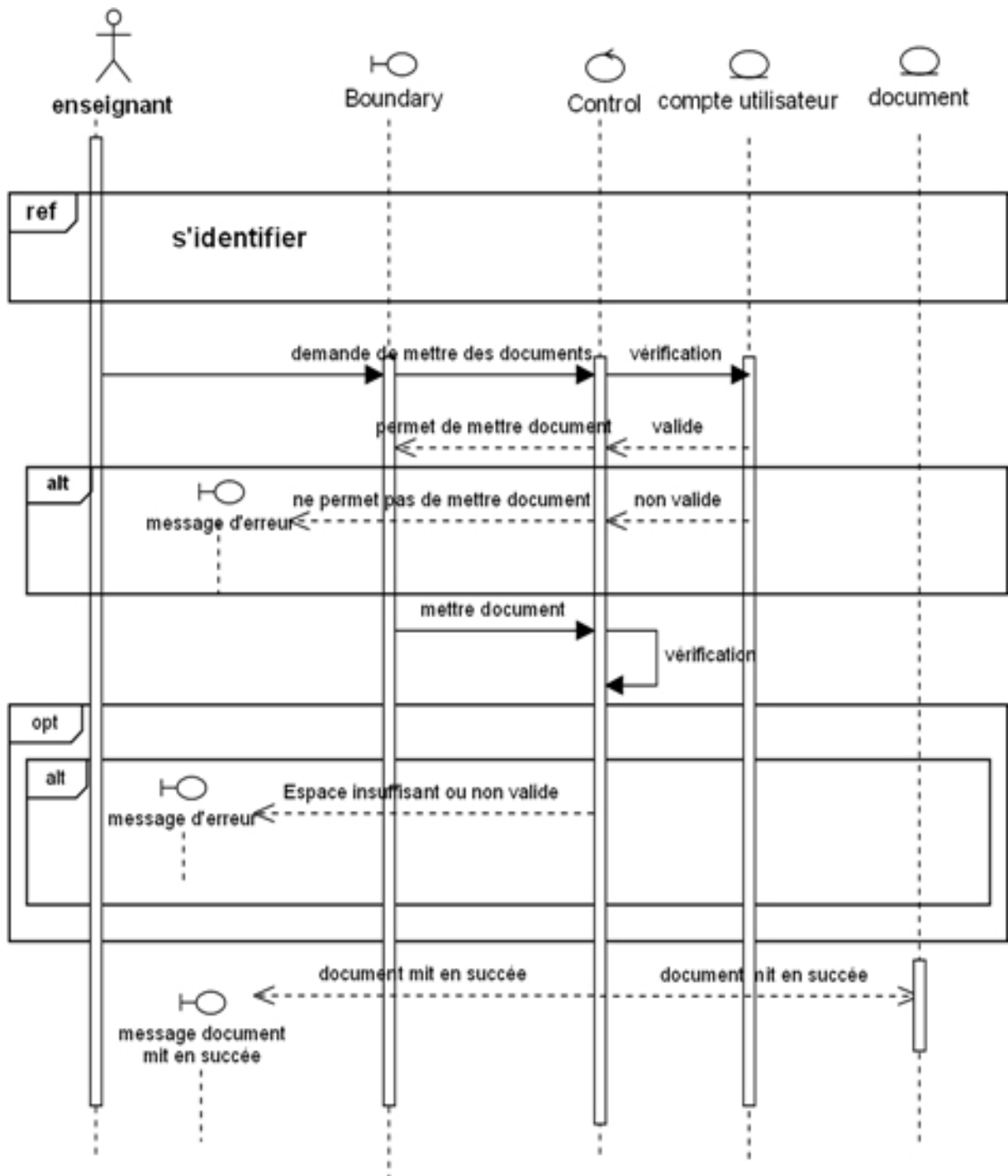


Figure 5.3 : diagramme d'interaction « mettre document »

### 5.2.4 Participer

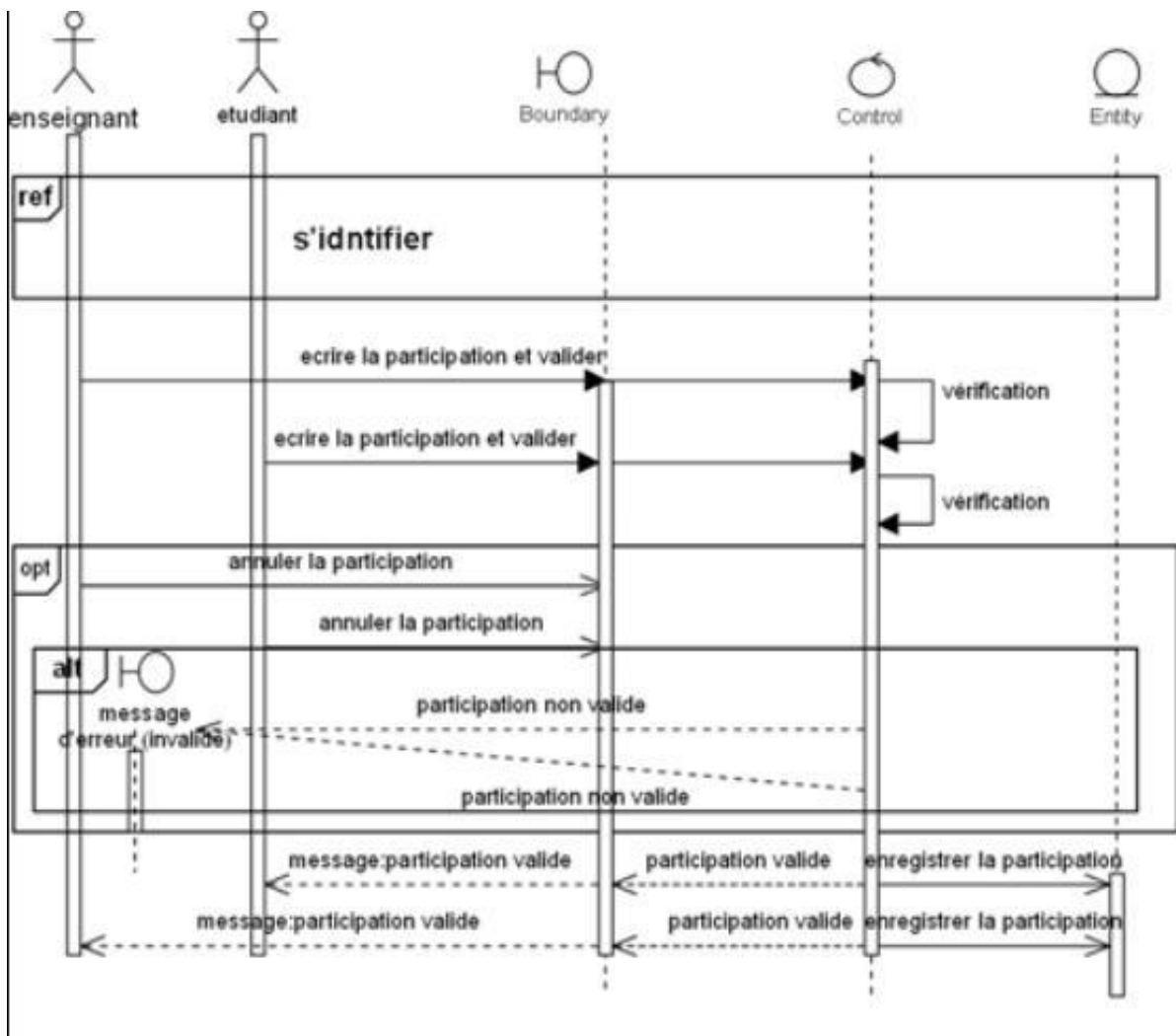


Figure 5.4 : diagramme d'interaction «participer »

### 5.2.5 Changer mot de passe

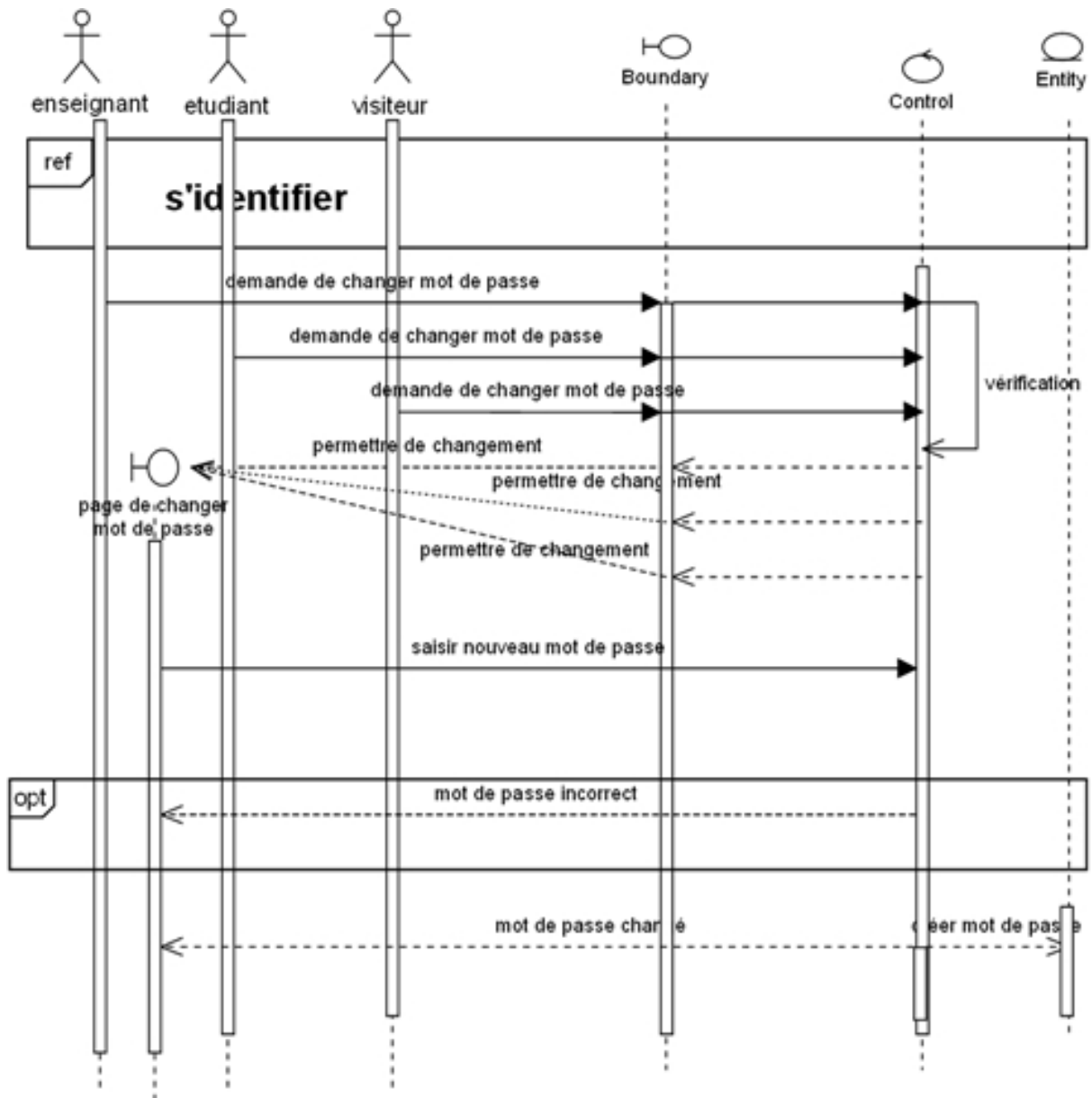


Figure 5.5 : diagramme d'interaction «changer mot de passe »

### 5.2.6 Télécharger

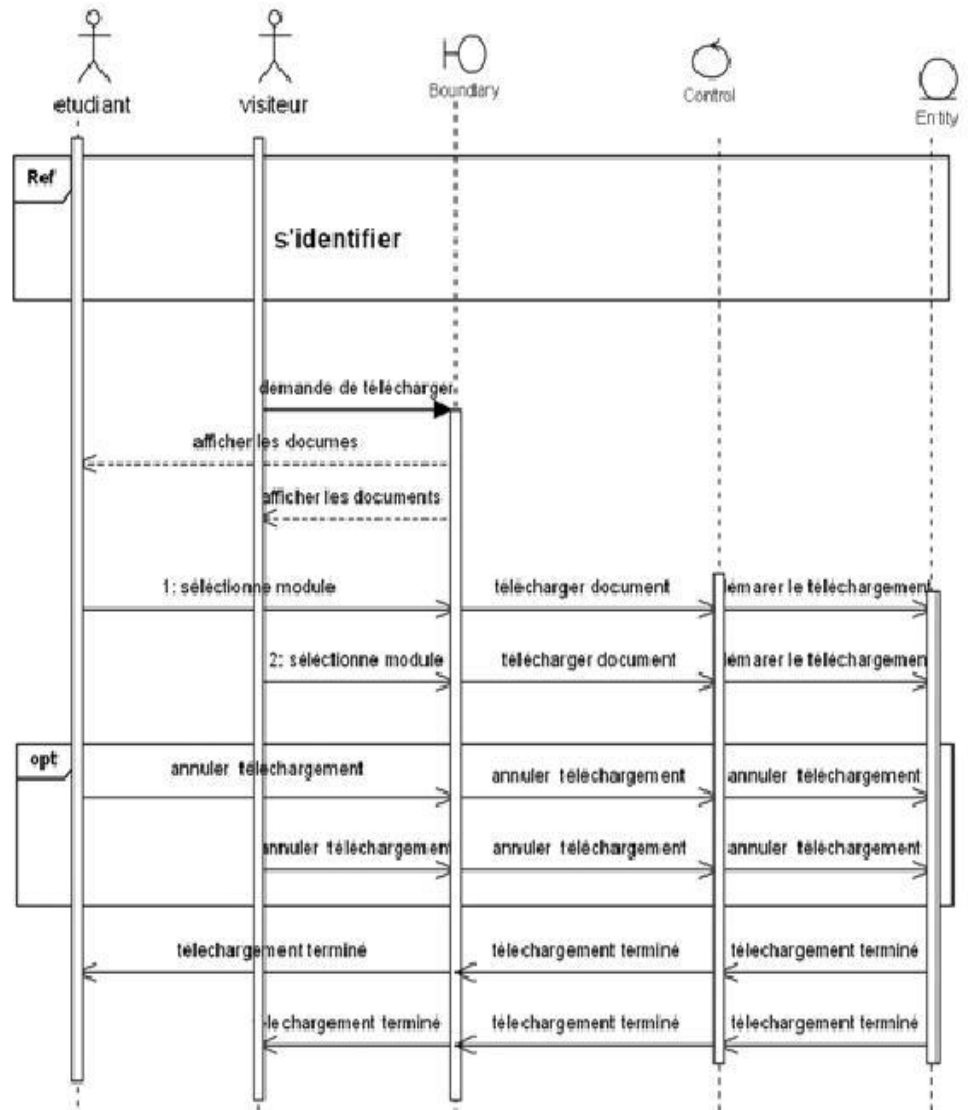


Figure 5.6 : diagramme d'interaction «Télécharger »

### 5.2.7 S'inscrire

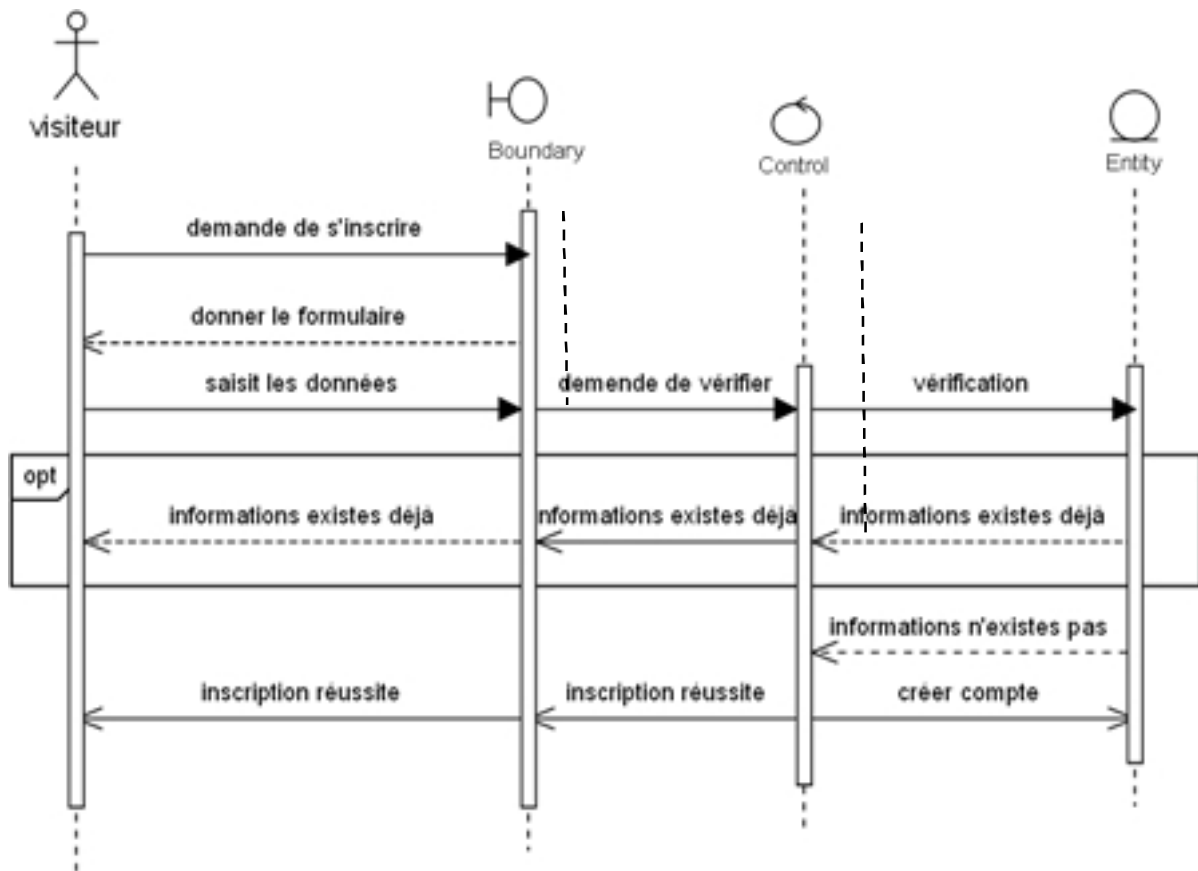


Figure 5.7 : diagramme d'interaction «s'inscrire »

## 5.3 Le diagramme de classes de conceptions

### 5.3.1 Mettre document

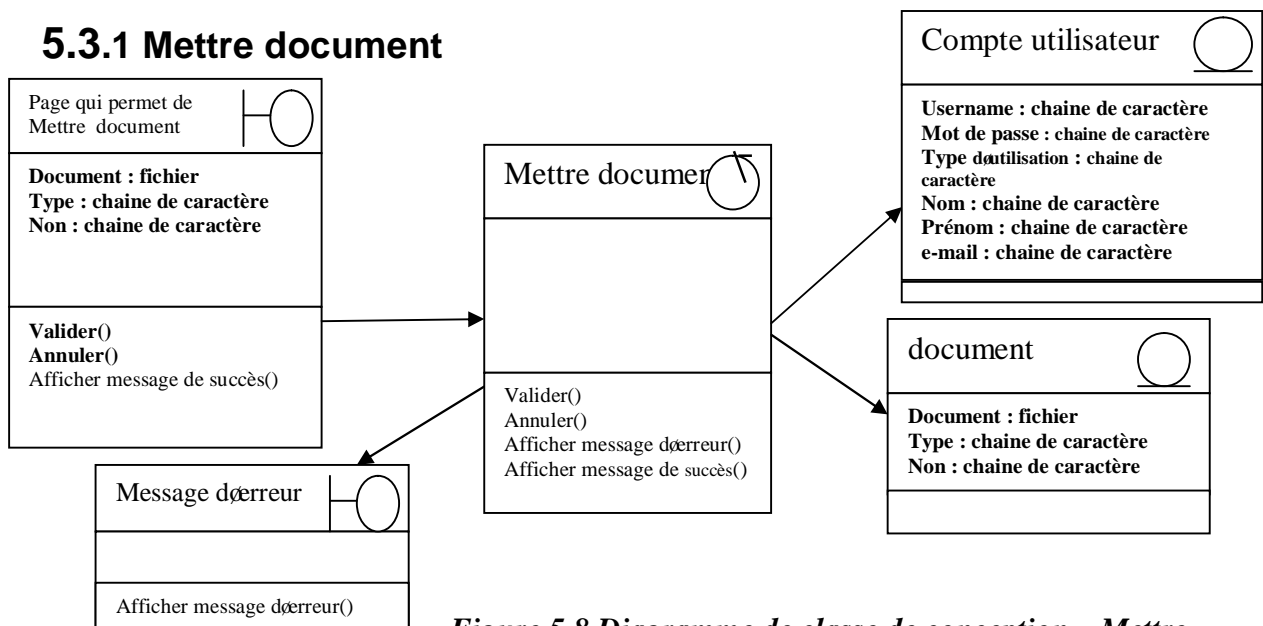


Figure 5.8 Diagramme de classe de conception « Mettre document »

### 5.3.2 Télécharger

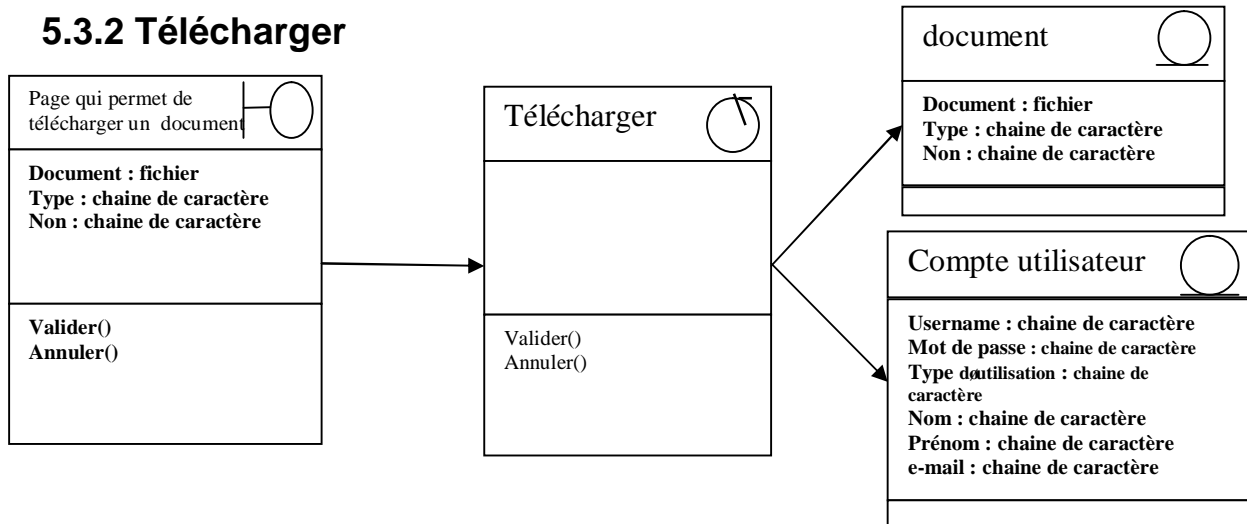


Figure 5.9 Diagramme de classe de conception «Télécharger »

### 5.3.3 Participer

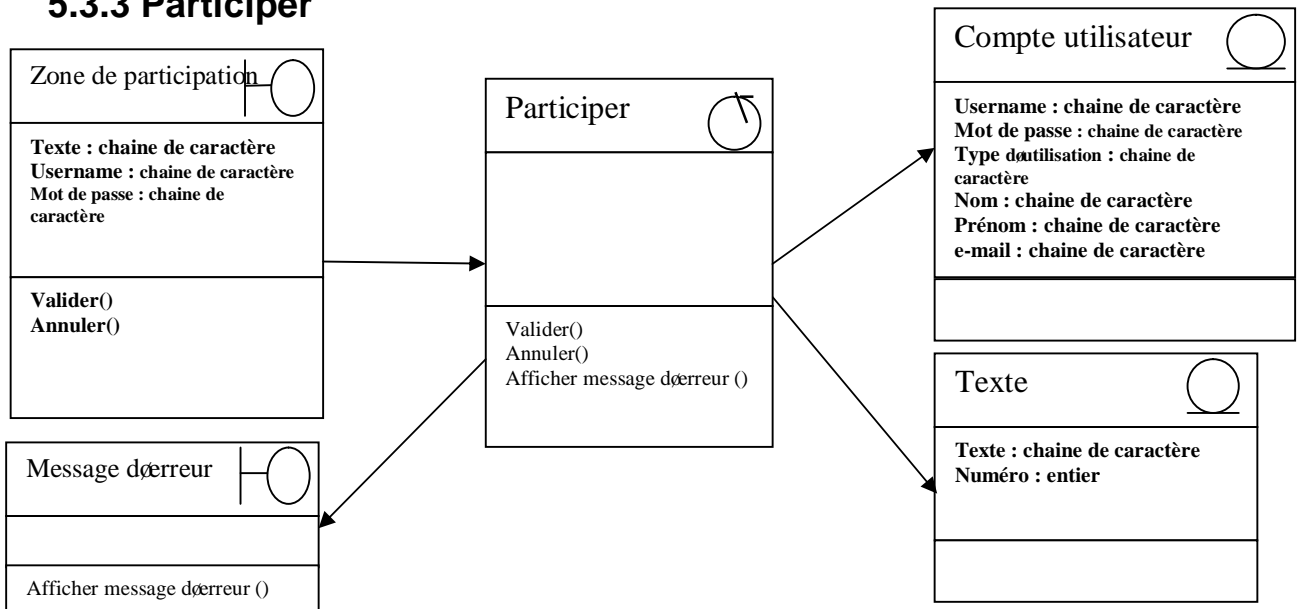


Figure 5.10 Diagramme de classe de conception «Participer »

### 5.3.4 Changer mot de passe

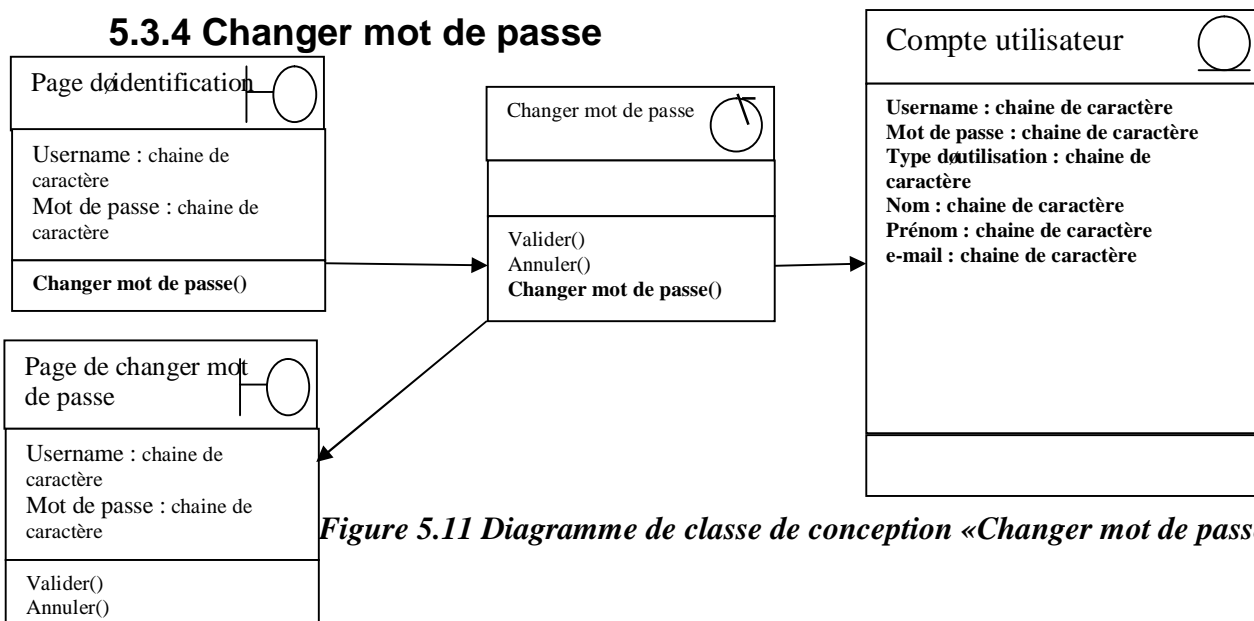


Figure 5.11 Diagramme de classe de conception «Changer mot de passe »



### 5.3.5 sØdentifier

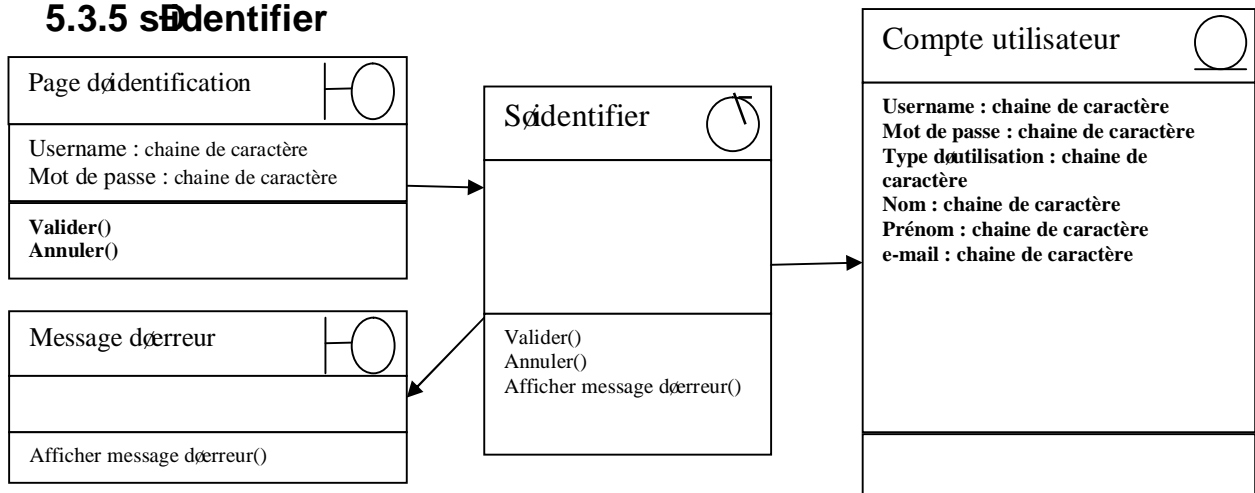


Figure 5.12 Diagramme de classe de conception «sØdentifier »

### 5.3.6 créer compte

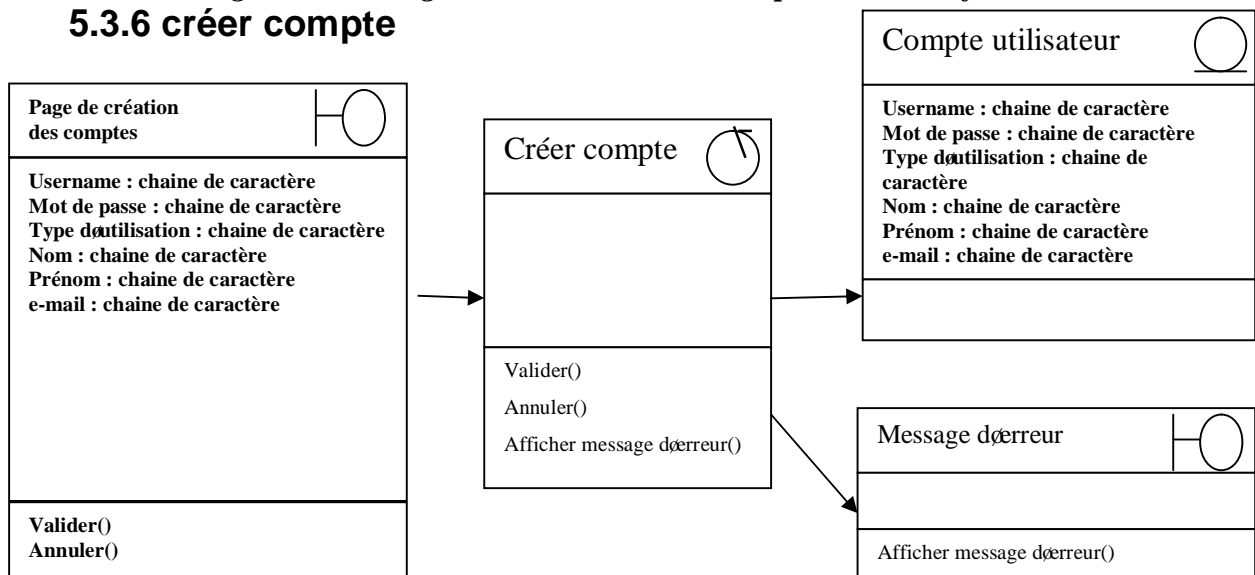


Figure 5.13 Diagramme de classe de conception «créer compte »

### 5.3.7 sØnscrire

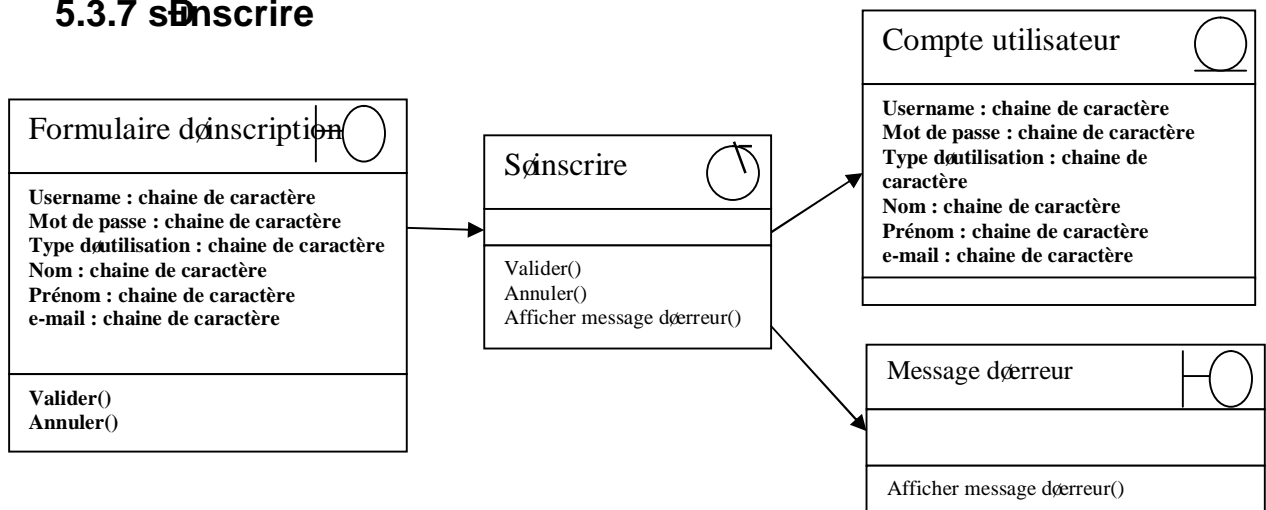


Figure 5.14 Diagramme de classe de conception «sØnscrire »

## 5.4 Diagramme de classe

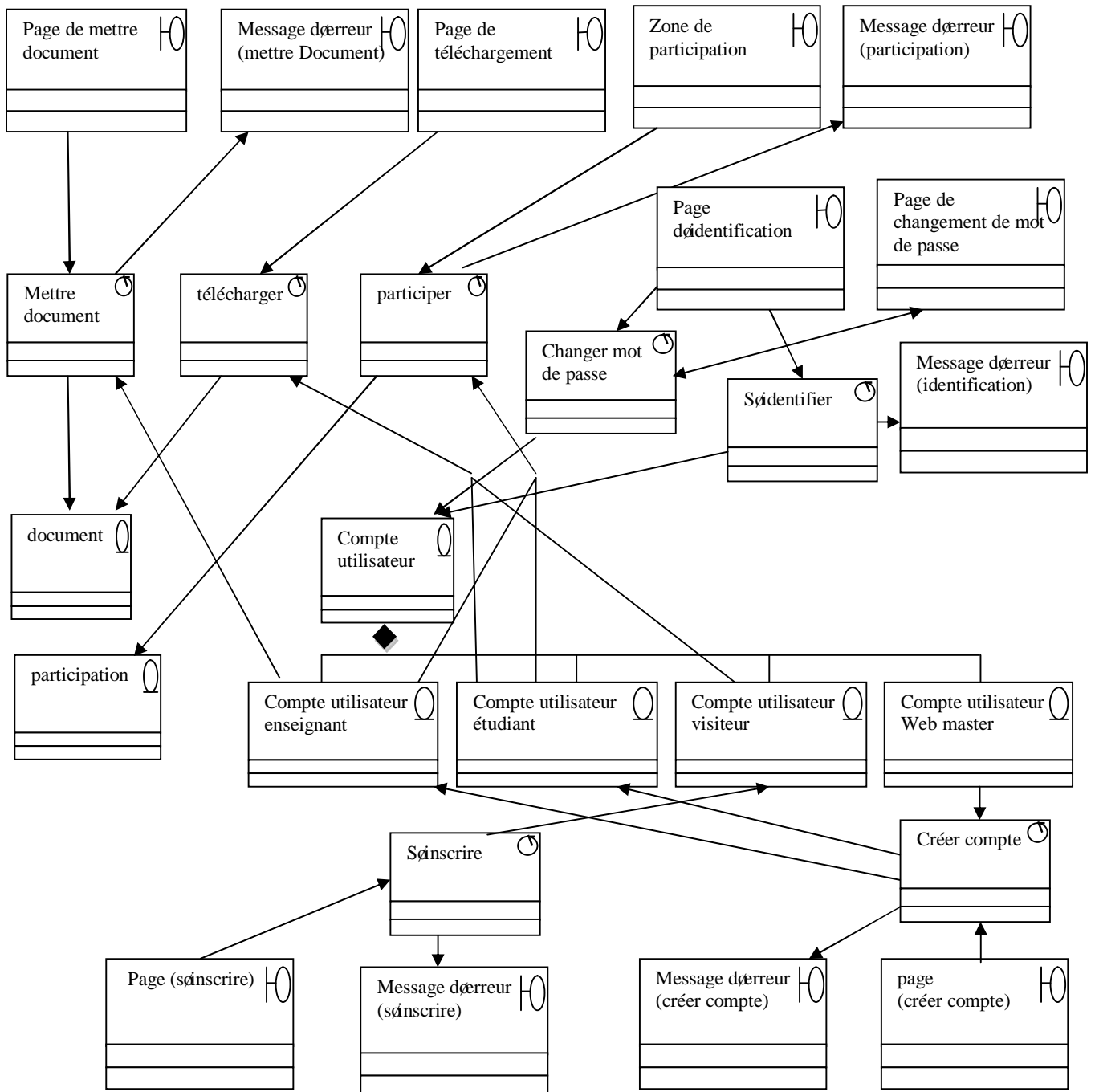


Figure 5.15 : Diagramme de classe

## 5.5 Conclusion

Dans ce chapitre, nous avons utilisé les diagrammes d'interaction et les diagrammes de classe de conception qui sont appartenent d'une démarche simplifiée de développement logiciel issue du UP.

La phase de conception a intégré des nouveaux classes et méthodes aux modèles d'analyse et la manière de naviguer entre ces classes de conception.

Le chapitre suivant présente les outils utilisées pour la phase implémentation et donne une première idée des différentes pages du site.



**PARTIE 3**  
**LA**  
**RÉALISATION**  
**DE**  
**L'APPLICATION**

The background of the page is a light-colored marbled paper with a complex, organic pattern of grey and white veins. The page is framed by a dark grey border with rounded corners. In the top-left and bottom-left corners, there are decorative scrollwork elements that appear to be part of a larger frame or binding.

# CHAPITRE 06 : IMPLÉMENTATION

## 6.1 Introduction

Chaque site web est besoin de plusieurs outils et langages pour réaliser, pour cela on a choisi quelques outils et langages parmi les langages qu'ils existent.

Dans ce chapitre on va essayer de justifier notre choix de ces langages.

## 6.2 Outils et langages utilisés

### 6.2.1 Pourquoi utiliser HTML ?

- HTML est indépendant de toute marque, conçu par des usagers pour des usagers, il est donc fait pour libérer et non pour rendre le client captif d'un produit particulier.
- Une page en HTML peut être lue par des ordinateurs de différentes marques pour obtenir essentiellement le même résultat.
- Sa structure distribuée (liens externes) permet de changer un élément sans rien changer à la page où il s'intègre. Le même élément peut être utilisé plusieurs fois sans devoir le répéter plusieurs fois.
- Lien réseau: On peut intégrer des éléments locaux dans une page personnelle ou destinée à des élèves et des éléments distants dont on intègre l'adresse (URL).
- L'hypertexte permet de gérer des informations très diversifiées sans devoir les stocker localement. [13]

### 6.2.2 Pourquoi utiliser PHP ?

PHP est libre et gratuit (il est distribué sous licence GNU/GPL). C'est un langage simple d'apprentissage, il ne nécessite pas de compilation et n'utilise pas de balises. Il est utilisable aussi bien sous plateformes linux que Windows.

PHP permet encore bien plus de choses comme demander d'afficher la date courante chaque fois que l'on consulte une page, de créer des forums, des annuaires, des livres d'or...

Le serveur web interprète les pages PHP et restitue au navigateur du code HTML, l'utilisateur n'a pas besoin d'un ordinateur très puissant pour les afficher. Par contre si l'on

souhaite gérer le code d'une page du coté du client il faudra utiliser du JavaScript, java ou encore Flash.

Comme le HTML il n'est pas nécessaire de disposer de logiciel particulier pour écrire le code de ses pages, le simple bloc-notes de Windows peut suffire.

#### ❖ Quelques avantages de PHP

- PHP possède plusieurs avantages significatifs :
- Des performances élevées,
- Des interfaces vers différents systèmes de bases de données,
- Des bibliothèques intégrées pour la plupart des tâches Web,
- Un faible coût,
- La simplicité d'utilisation,
- La portabilité,
- La disponibilité de son code source. [14]

#### 6.2.3 Pourquoi utiliser MySQL ?

Les principaux concurrents de MySQL sont PostgreSQL, Microsoft SQL Server et Oracle. Par rapport à tous ces produits, MySQL possède plusieurs avantages majeurs, dont :

- De performances élevées,
- Un coût réduit,
- Sa simplicité de configuration,
- Sa portabilité,
- L'accessibilité de son code source. [14]

### 6.2.4 Pourquoi utiliser CSS ?

CSS nous aide à garder l'information contenue dans notre document séparée des détails de sa présentation. Ces détails de présentation d'un document sont appelés son *style*. La séparation du style et du contenu permet :

- D'éviter des duplications.
- Une maintenance plus facile.
- D'utiliser le même contenu avec différents styles pour différents usages.

En général, avec, CSS, vous utilisez le langage de balisage pour décrire les informations contenues dans le document, et non son style. CSS est utilisé pour spécifier son style et non son contenu. [13]

### 6.2.5 Pourquoi utiliser ~~IE~~ Adobe DreamWeaver ?

Adobe DreamWeaver (anciennement Macromedia DreamWeaver) est un éditeur de site web de type WYSIWYG.

DreamWeaver fut l'un des premiers éditeurs HTML de type « tel affichage, tel résultat », mais également l'un des premiers à intégrer un gestionnaire de site (CyberStudio [GoLive](#) étant le premier). Ces innovations l'imposèrent rapidement comme l'un des principaux éditeurs de site web, aussi bien utilisable par le néophyte que par le professionnel. [15]

### 6.2.6 Pourquoi utiliser EasyPHP ?

Soit vous installez un serveur web en local, qui vous permettra de tester directement vos pages PHP. EasyPHP permet de réaliser simplement cette dernière solution.

EasyPHP permet également d'installer MySQL, une base de données, Cette dernière est un programme permettant de gérer une grande quantité de données en les organisant sous forme de tables.

On peut de plus accéder très facilement à une base de données MySQL à partir de PHP, ce qui permet de développer des sites web très performants et interactifs .EasyPHP joint PHPMYAdmin à MySQL, un outil écrit en PHP permettant de gérer vos bases de données



MySQL. En utilisant EasyPHP, vous pouvez installer un serveur web complet, qui vous permettra de faire tous vos tests de pages PHP en toute facilité. [16]

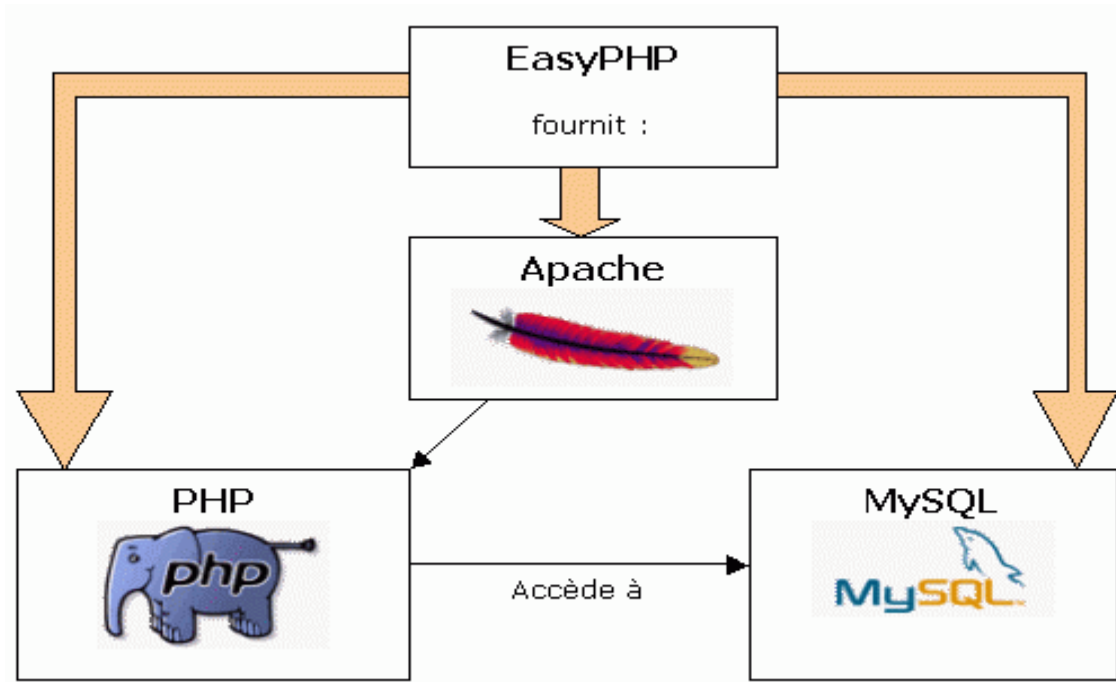


Figure 6.1 : l'architecture de PHP

### 6.2.7 Pourquoi utiliser UML ?

UML est un langage de modélisation, très répandu et très utile pour la conception des systèmes d'informations orientés objets. Il est utile pour les cas d'utilisation de l'application.

C'est vrai que UML n'est pas choisit automatiquement pour la conception mais il est déjà plus connu que merise et merise n'est pas orientée objet. Pour les applications web, je pense que UML exprime mieux la vue statique et dynamique du système d'information. [17]

## 6.3 Conception de la base de données

Nous donnons ici 4 après quatre règles (de R1 à R4) pour traduire un schéma conceptuel entité association ou UML en un schéma relationnel équivalent. Il existe d'autres solutions de transformation mais ces règles sont les plus simples et les plus opérationnelles :

**Transformation des entités/ classes :** La règle est simple R1 : \* Chaque entité devient une relation, identifiant de l'entité devient clé primaire pour la relation. \* Chaque classe du diagramme UML devient une relation. Il faut choisir les attributs de la classe pouvant jouer le

rôle d'identifiant. **Transformation des associations** Les règles de transformation que nous allons voir dépendent des cardinalités / multiplicités maximale des associations. Nous distinguons trois familles d'association : **Association 1..\*** : La règle est la suivante : R2 : Il faut ajouter un attribut de type clé étrangère dans la relation fils de l'association. L'attribut porte le nom de la clé primaire de la relation père de l'association. **Association \*.\*** : La règle est la suivante : R3 : L'association / classe- association devient une relation. La clé primaire de cette relation est la concaténation des identifiants des entités connectées à l'association. Chaque attribut devient clé étrangère si l'entité / classe connectée dont il devient une relation en vertu de la règle R1. Les attributs de l'association / classe- association doivent être ajoutés à la nouvelle relation. Ces attributs ne sont ni clé primaire, ni clé étrangère. **Association 1..1** : La règle est la suivante :

R4 : Il faut ajouter un attribut de type clé étrangère dans la relation dérivée de l'entité ayant la cardinalité minimale égale à zéro. Dans le cas du diagramme UML il faut ajouter un attribut de type clé étrangère dans la relation dérivée de la classe ayant la multiplicité minimale égale à un. L'attribut porte le nom de la clé primaire de la relation dérivée de l'entité / classe connectée à l'association. Si les deux cardinalités minimales égale à zéro, le choix est donné entre les deux relations dérivées de la R1. Si les deux cardinalités minimales égale à un, il est préférable de fusionner les deux entités / classe en une seule [18].

Document (code\_doc, document, type, nom)

Enseignant (code\_ens , username, mot de passe, type d'utilisation, nom, prénom, e-mail)

Etudiant (code\_étud , username, mot de passe, type d'utilisation, nom, prénom, e-mail)

Visiteur (code\_util , username, mot de passe, type d'utilisation, nom, prénom, e-mail)

Web master (code\_WM , username, mot de passe, type d'utilisation, nom, prénom, e-mail)

Texte (code\_Texte, texte, emplacement)

## 6.4 Carte de site

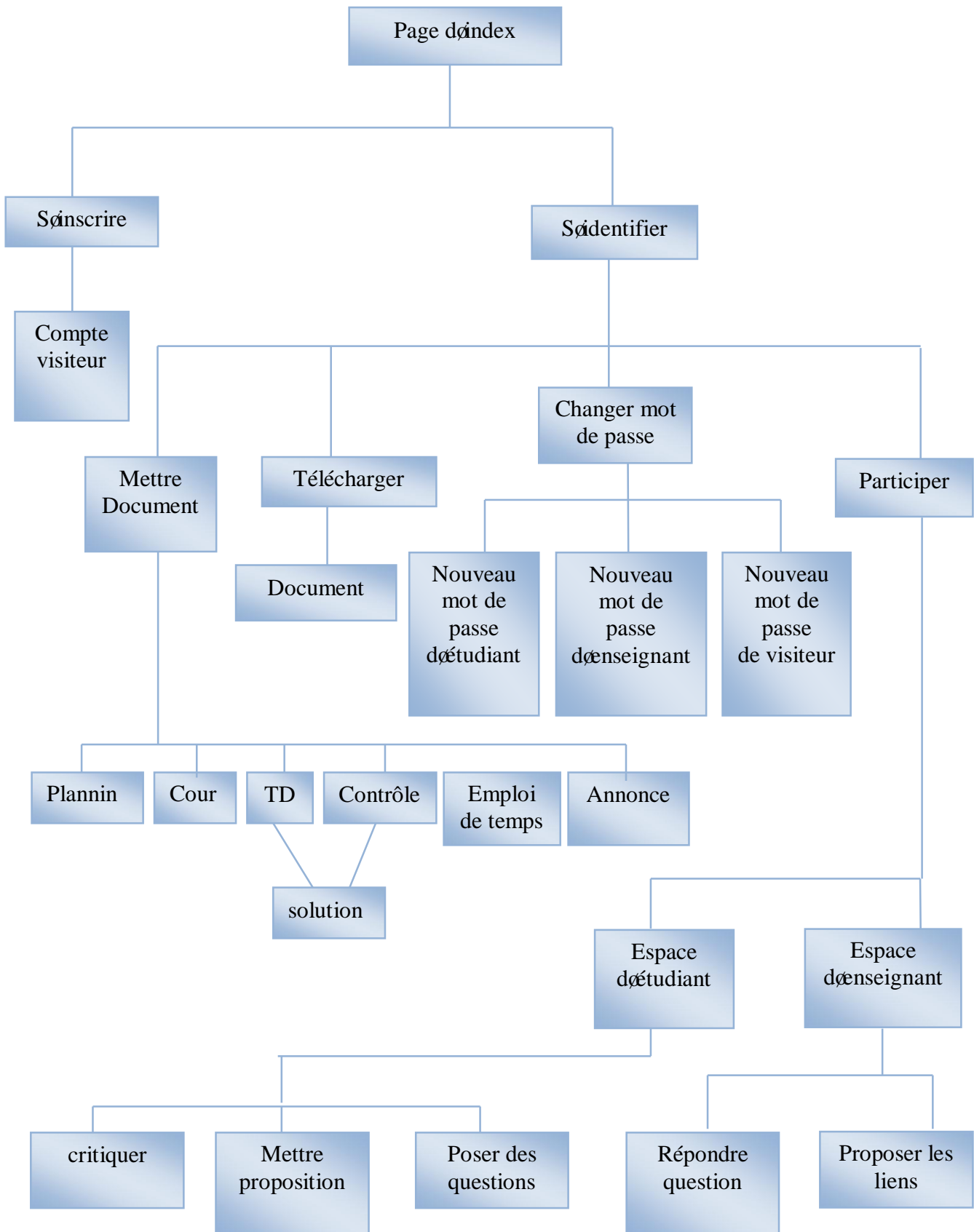


Figure 6.2 : Carte de site

## 6.5 Les interfaces de notre application

### 6.5.1 L'interface de notre application

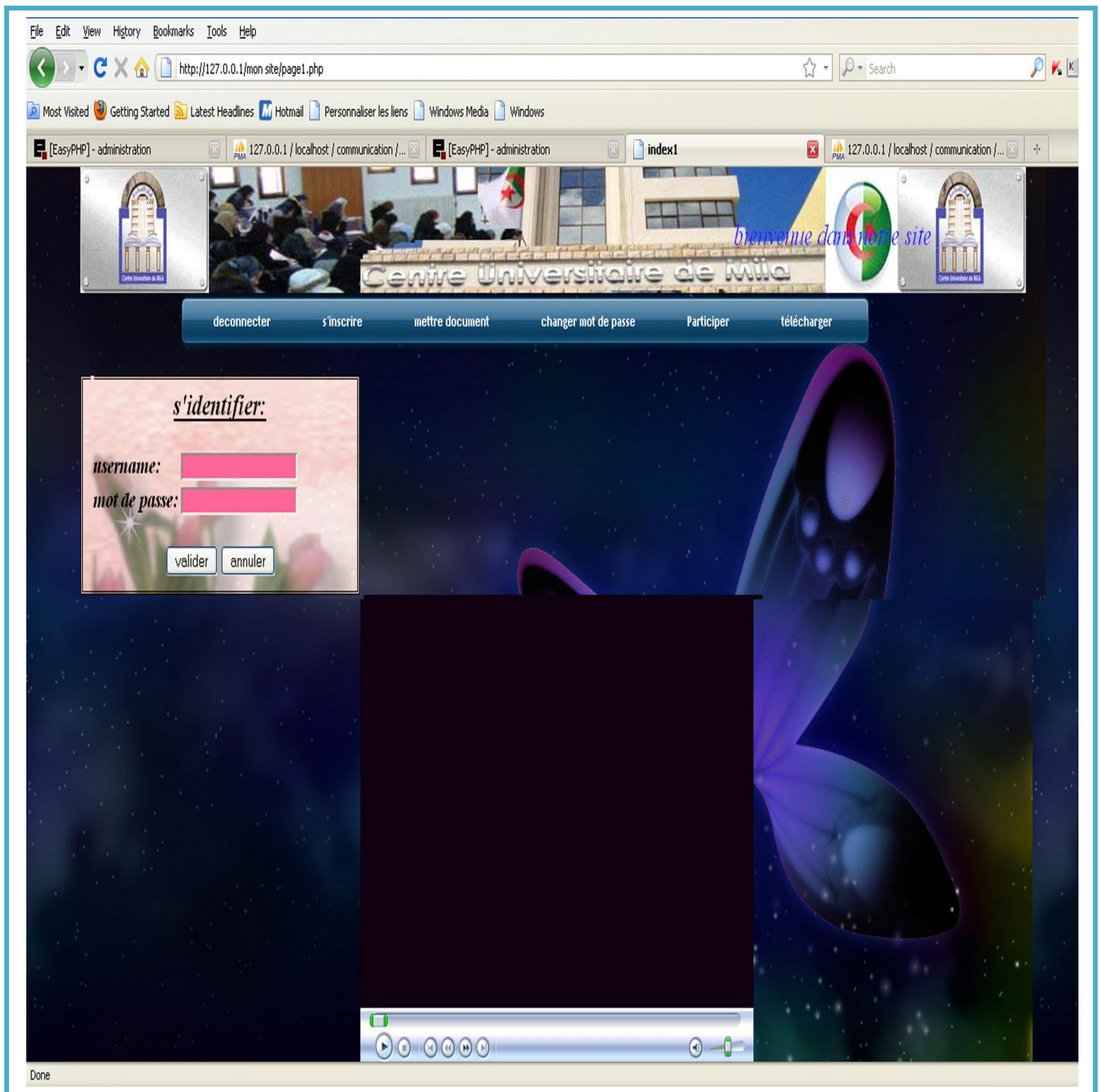


Figure 6.3 : L'interface de notre application



## 6.5.2 S'inscrire :

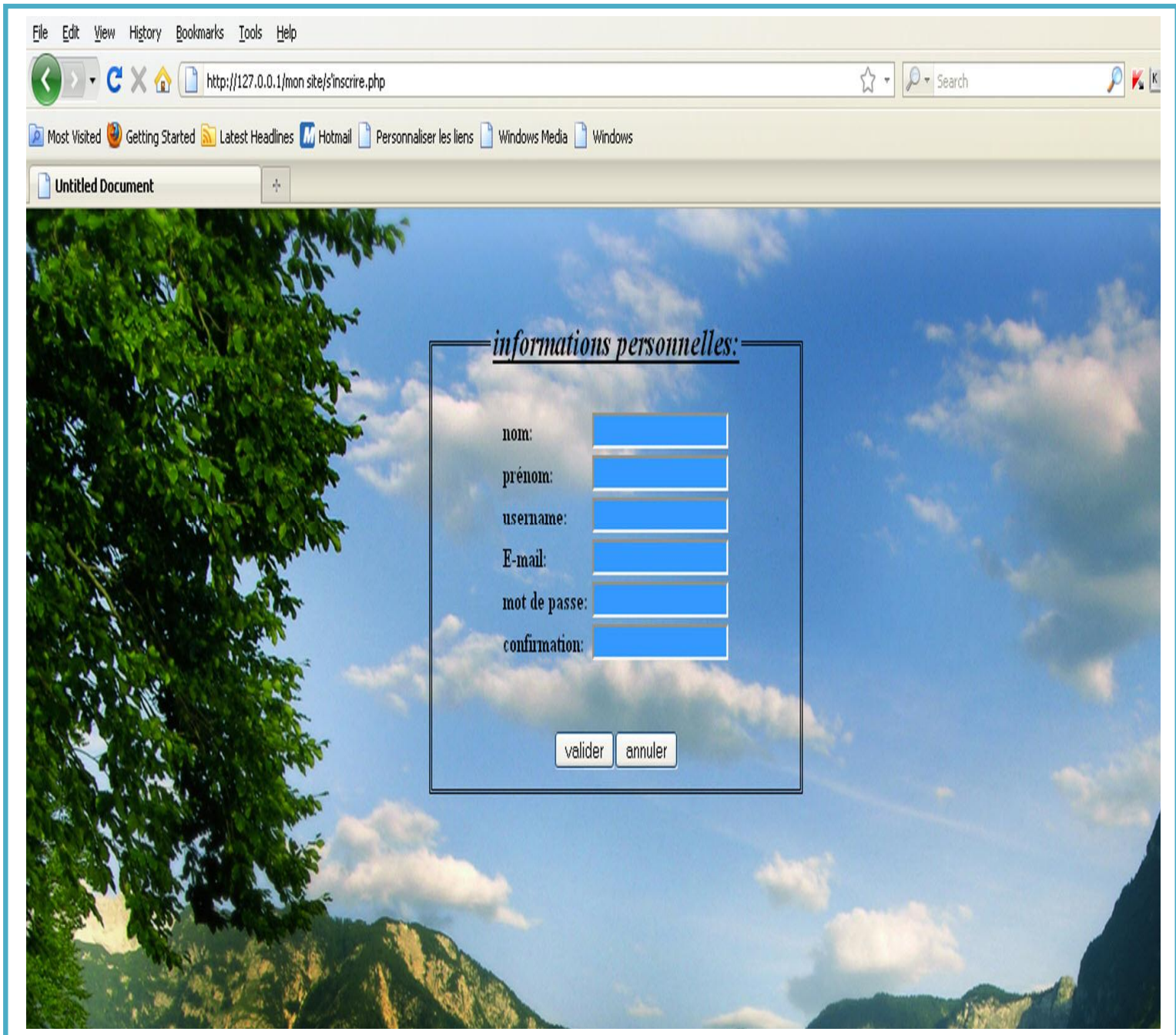


Figure 6.4 : L'interface «s'inscrire »

### 6.5.3 Mettre document (par exemple : mettre cours) :

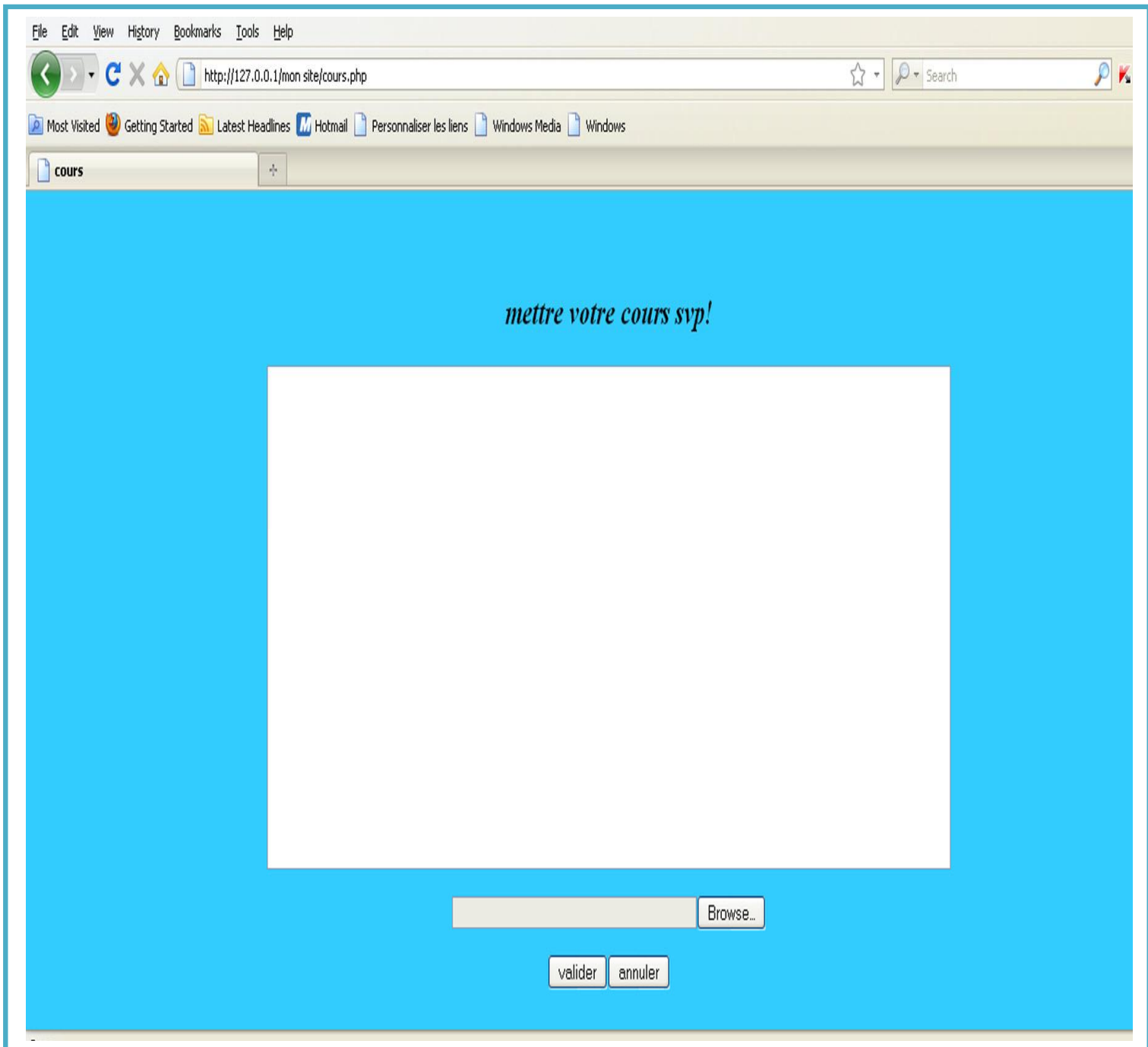


Figure 6.5 : L'interface «Mettre document »

### 6.5.4 Mettre document (par exemple : emploie-temps) :

mettre votre emploi du temps svp!

nom\_enseignant:

heures	8:00-9:30	9:30-11:00	11:00-12:30	12:30-14:00	14:00-15:30	15:30-17:00
samedi	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
dimanche	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
lundi	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
mardi	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
mercredi	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
jeudi	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

valider annuler

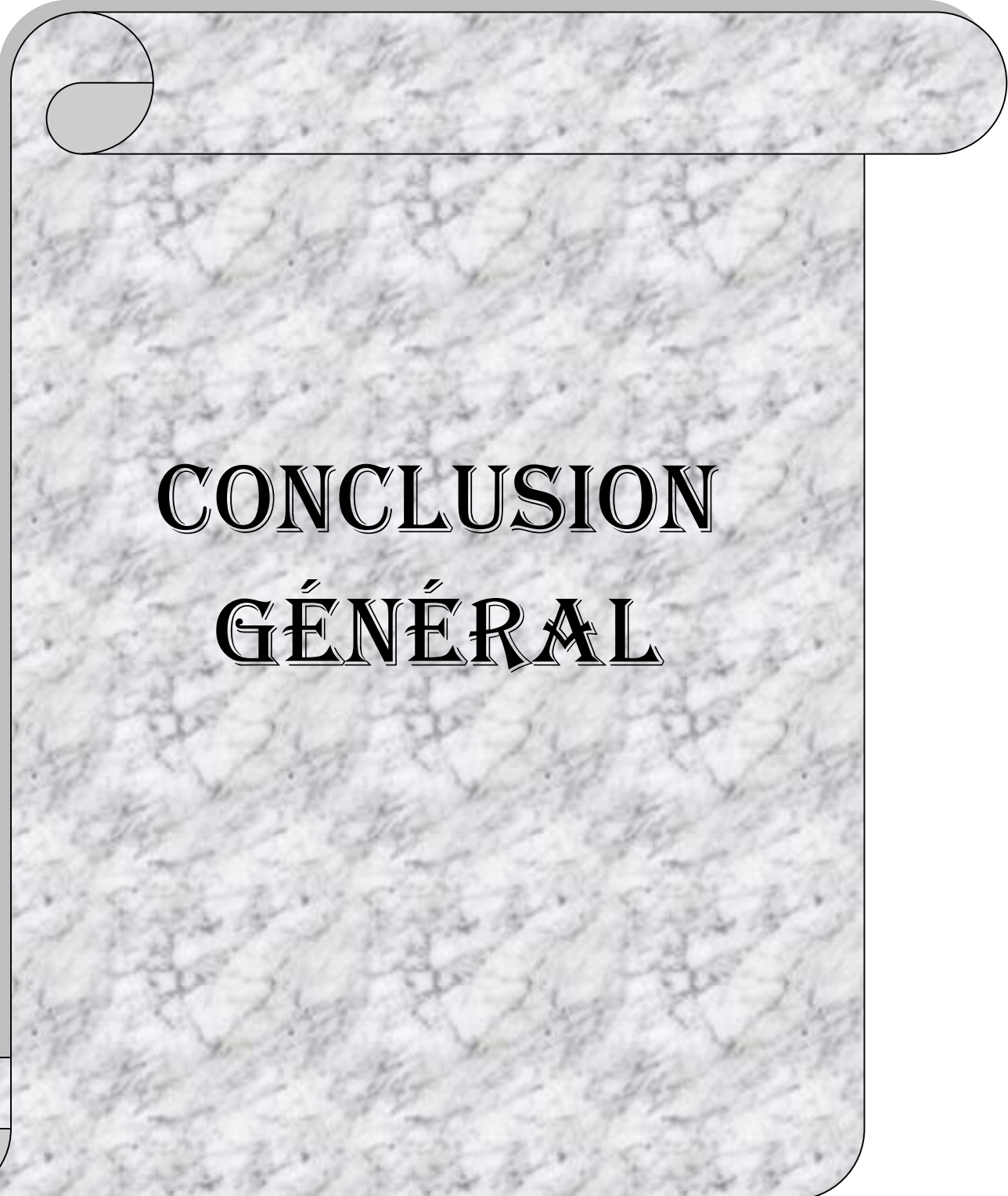
Done

Figure 6.6 : L'interface «emploi du temps»

## **6.6 Conclusion**

Pour réaliser notre site web, on a utilisé le langage PHP pour notre page du coté serveur et EasyPHP pour la création et la gestion de notre base de données son oublié le logiciel adobe DreamWeaver CS3 pour la création des pages du site.





**CONCLUSION  
GÉNÉRAL**

## Conclusion général

L'objectif de notre projet est la réalisation d'un site web dynamique pour une communication entre les enseignants et les étudiants qui permet ces derniers de communiquer entre eux et permet aussi aux visiteurs de participer par l'intermédiaire du site .

Afin de réaliser notre projet, on a utilisé pour la conception, les diagrammes du langage UML, parce qu'il est adapté pour les processus du développement.

Comme UML n'est pas une méthode, son utilisation exige un processus de développement, itératif et incrémental, centré sur les cas d'utilisation. On a suivi la méthodologie UP, car elle est caractérisée par les exigences d'utilisation d'UML. pour l'implémentation du projet abordé on a utilisé des techniques de développement récentes tel que PHP, MySQL, EasyPHP

Ainsi nous avons pu réussir à réaliser une application dynamique basé sur une conception adaptée à nos besoins.

Nous avons réalisé une grande partie de notre objectif.

Comme nous n'avons pas eu assez du temps, on n'a pas pu respecter toutes les critères ergonomiques pour réaliser notre site web.

Nous espérons, que notre travail a été bien détaillé et servira comme support pour les prochains étudiants. Néanmoins il est certain qu'une application web est extensible de nature et des améliorations peuvent être apportées à notre application.



**RÉFÉRENCES  
BIBLIOGRAPHIQUES**

## Í Bibliographies et webgraphiesÍ

- [1] : *Guide to the Internet and World Wide Web For Archivists and Records Managers*
- [2] : Pascal Roques *UML-Modéliser un site e-commerce version 2008*
- [3] : <http://www.ietf.org>
- [4] : Andreas Voss *Dictionnaire de l'informatique et de l'internet* 2001
- [5] : <http://crdp.ac-dijon.fr/sites-statiques-sites-dynamiques.html>
- [6] : [http://fr.wikipedia.org/wiki/Application\\_Web](http://fr.wikipedia.org/wiki/Application_Web)
- [7] : [http://fr.wikipedia.org/wiki/Site\\_Web](http://fr.wikipedia.org/wiki/Site_Web)
- [8] : [http://fr.wikipedia.org/wiki/Unified\\_Modeling\\_Language](http://fr.wikipedia.org/wiki/Unified_Modeling_Language)
- [9] : [Http://fr.wikipedia.org/wiki/Merise\(informatique\)](http://fr.wikipedia.org/wiki/Merise(informatique))
- [10] : <http://sabricole.developpez.com/uml/tutoriel/unifiedprocessus/>
- [11] : <http://www.netpreserve.org>
- [12] : Pascal Roques *UML-Modéliser un site e-commerce version 2008*
- [13] : <http://tecfa.unige.ch/themes/FAQ-FL/quest-ce-html/>
- [14] : <http://www.innomatiques.com/creation-site-web-php-mysql.php>
- [15] : <http://adobe-dreamweaver.softonic.fr/>
- [16] : <http://geronimo.developpez.com/EasyPHP/>
- [17] : <http://UML>
- [18] : Laurent AUDIBERT *UML 2 Édition 2007-2008*
- [19] : [https://developer.mozilla.org/fr/CSS/Premiers\\_pas/Pourquoi\\_utiliser\\_CSS](https://developer.mozilla.org/fr/CSS/Premiers_pas/Pourquoi_utiliser_CSS)