

République Algérienne Démocratique & Populaire Ministère de  
l'enseignement supérieur et de la recherche scientifique  
UNIVERSITE MOULOUD MAMMERI DE TIZI-OUZOU



FACULTE DU GENIE ELECTRIQUE ET D'INFORMATIQUE  
DEPARTEMENT D'INFORMATIQUE

# **Mémoire de fin d'études de MASTER PROFESSIONNEL**

Domaine : **Mathématiques et Informatique**  
Filière : **Informatique**  
Spécialité : **Ingénierie des systèmes d'information**  
**Thème**

**Conception et réalisation d'une application JavaEE**

**Cas : E-learning**

**Réalisé par :**

**Melle: Sadat Lynda**

**Mr : Cherfioui Djamel**

**Encadré par :**

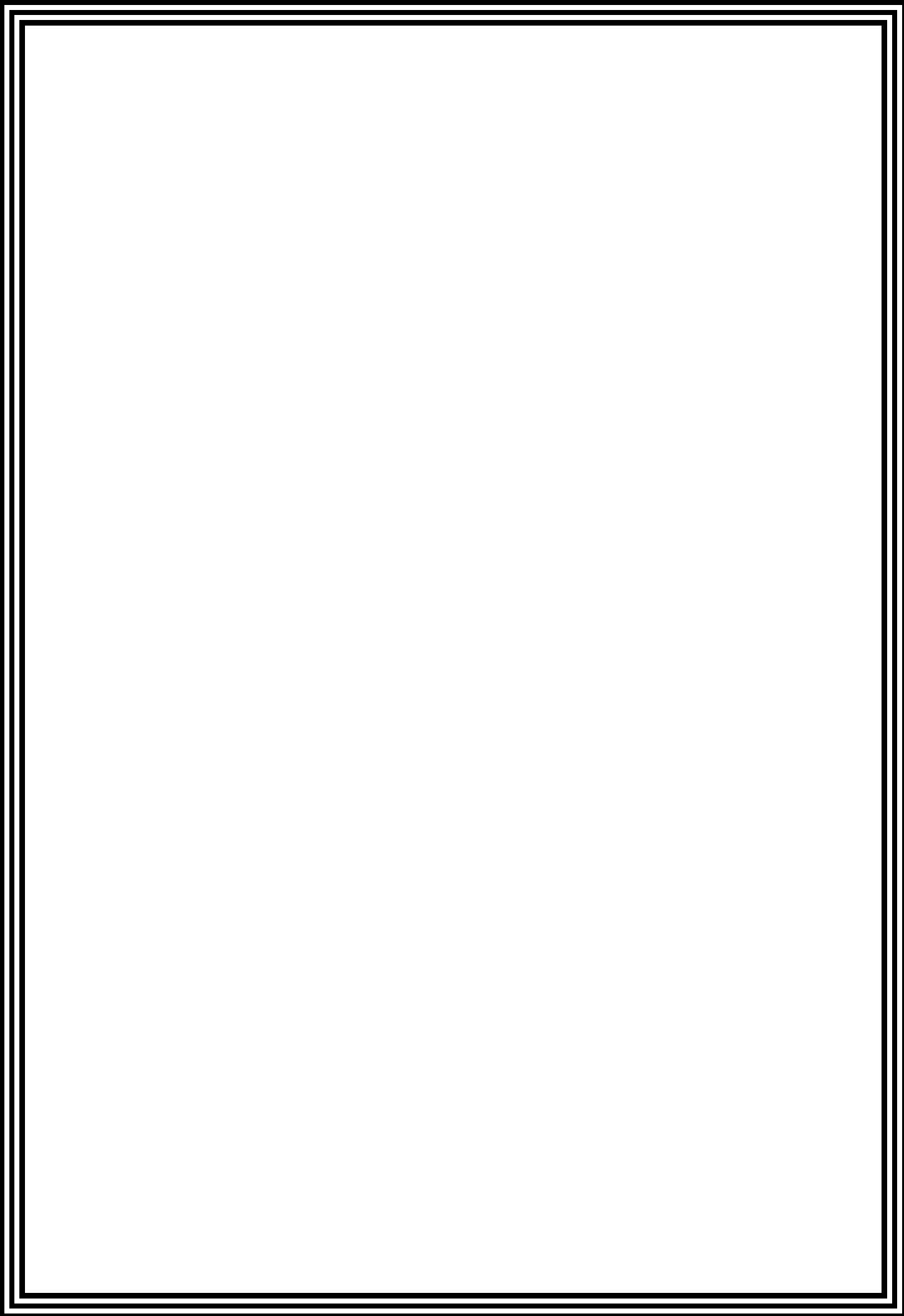
**Mr: Kerbich Mhand**

**Jury composé de :**

**Président : Bourkache.G**

**Examineur : Rabahellah.K**

**Année universitaire : 2016-2017**



# *Remerciements*

Nous tenons à remercier vivement notre promoteur Mohand Kerbich qui nous a fait confiance et s'est mise à notre entière disposition pour le déroulement de notre travail.

Nous remercions chaleureusement les membres du jury pour l'honneur qu'ils nous font en acceptant de juger ce mémoire de fin d'études.

Enfin, nous remercions toutes les personnes ayant contribué de près ou de loin au bon accomplissement de notre travail.

# *Dédicaces*

*Je dédie ce travail à :*

A mes très chers parents.

A toute ma famille.

A mon binôme.

A tous mes amis.

**Lynda Sadat.**

# *Dédicaces*

*Je dédie ce travail à :*

A mes très chers parents.

A toute ma famille.

A mon binôme.

A tous mes amis.

**Cherfioui Djamel.**

## Introduction générale

### CHAPITRE I : PRESENTATION DE LA TECHNOLOGIE JAVA EE

I.1	Introduction.....	2
I.2	Conception architecturale [1] .....	2
I.2.1	Architecture logicielle.....	2
I.2.2	Utilité d'une architecture logicielle.....	2
I.2.3	Structuration des applications .....	3
I.2.3.1	Vue en couches « vue logique ».....	3
I.2.3.2	Vue en niveaux « vue physique ».....	6
	Architecture Client/serveur [2] .....	7
I.3	Java Entreprise Edition.....	10
I.3.1	Présentation de JAVA EE [3] .....	10
I.3.2	L'architecture de JAVA EE.....	10
I.3.3	Les Couches logicielles Java EE [4].....	11
I.3.4	Spécification de la plateforme Java EE.....	12
I.3.4.1	Les java persistances APIs.....	12
	Les composants .....	12
	Services d'infrastructures .....	15
	Services de communication.....	16
I.3.4.2	Les serveurs d'application.....	16
	Le rôle d'un serveur d'application .....	17
I.3.5	Les avantages d'utiliser Java EE.....	18
I.4	Conclusion .....	19

## CHAPITRE II : ETAT DE L'ART DU E-LEARNING

II.1	Introduction.....	20
II.2	La formation à distance [5].....	20
II.2.1	Qu'est-ce que la FAD.....	20
II.2.2	Pourquoi la FAD.....	21
II.3	Formation Ouverte et à Distance (FOAD) .....	21
II.4	Différences entre la formation ouverte et la formation à distance .....	21
II.5	Présentation du E-learning.....	22
II.5.1	Définition du E-learning [6].....	22
II.5.2	Pourquoi l'E-Learning .....	23
II.5.3	Applications du E-Learning .....	23
II.6	Les différentes formes de la formation en ligne.....	24
II.6.1	Formation exclusivement en ligne (sans tutorat).....	24
II.6.2	Formation exclusivement en ligne (avec tutorat) .....	24
II.6.2.1	Asynchrone .....	25
	Les principaux avantages de la formation asynchrone.....	25
II.6.2.2	Synchrone .....	25
	Les principaux avantages de la formation synchrone.....	25
II.6.3	Formation mixte (avec tutorat en ligne) .....	25
	Les principaux avantages de la formation mixte.....	26
II.7	Outils de communication .....	26
II.8	Les Plate-forme de formation à distance ou LMS (Learning Management System) [7].....	27
II.8.1	Les acteurs d'un LMS et leurs rôles.....	28
	Concepteur (auteur ou créateur de cours) .....	28
	Formateur (ou tuteur).....	28
	Apprenant.....	28
	Administrateur.....	28

Orienteur .....	28
Evaluateur .....	28
II.8.2 Exemple de plateforme d'apprentissage en ligne .....	29
II.8.2.1 WebCT [8].....	29
II.8.2.2 Moodle (Modular Object-Oriented Dynamic Learning Environment) [9] .....	30
II.9 Les enjeux du E-learning.....	32
II.10 Conclusion .....	33

## CHAPITRE III : ANALYSE ET CONCEPTION

III.1 Introduction.....	34
III.2 Phase d'Initialisation .....	34
III.3 Phase d'Elaboration.....	34
III.3.1 Identification des acteurs.....	34
III.3.2 Diagramme de contexte .....	36
III.3.3 Identification des cas d'utilisations .....	36
III.3.4 Définition des itérations.....	39
III.4 Construction.....	39
III.4.1 Réalisation de l'itération 1.....	40
III.4.1.1 Diagrammes de séquence pour itération 1 .....	40
Description textuelle des cas d'utilisation.....	41
III.4.1.2 Diagrammes de classe pour l'itération 1 .....	46
III.4.2 Réalisation de l'itération 2.....	47
III.4.2.1 Diagrammes de séquence pour itération 2 .....	47
III.4.2.2 Diagrammes de classe pour l'itération2.....	55
III.4.3 Réalisation de l'itération 3.....	56
III.4.3.1 Diagrammes de séquence pour itération 3 .....	56
III.4.3.2 Diagrammes de classe pour l'itération 3 .....	58



III.5	Conclusion .....	59
-------	------------------	----

## CHAPITRE IV: LA REALISATION

IV.1	Introduction.....	60
IV.2	Les outils de développement .....	60
IV.2.1	GlassFishServer.....	60
IV.2.2	WampServer .....	61
IV.2.3	PhpMyAdmin.....	61
IV.2.4	NetBeans [10].....	62
IV.3	Les langages de programmations.....	64
IV.3.1	JAVA EE.....	64
IV.3.2	XHTML :(Extensible Hyper texte Markup Language) .....	64
IV.4	Le modèle Relationnel.....	65
IV.5	Le modèle physique de données .....	65
IV.6	Présentation de quelques interfaces .....	69
IV.7	Conclusion .....	72

## Liste des figures

### Chapitre I :

<b>Figure I.1:</b> Le Modèle a 5 couches .....	3
<b>Figure I.2 :</b> Le modèle vue contrôleur .....	5
<b>Figure I.3:</b> Dialogue client-serveur .....	8
<b>Figure I.4 :</b> Conception a deux niveaux.....	8
<b>Figure I.5 :</b> Conception a trois niveaux .....	9
<b>Figure I.6 :</b> Architecture à multi niveaux.....	9
<b>Figure I.7 :</b> Architecture et composants de Java EE.....	12

### Chapitre III :

<b>Figure III.1:</b> Diagramme de contexte.....	36
<b>Figure III.2:</b> Diagramme de cas d'utilisation globale.....	37
<b>Figure III.3:</b> Diagramme de séquence du cas d'utilisation : « Authentification ».....	41
<b>Figure III.4 :</b> Diagramme de séquence de cas d'utilisation « ajouter formateur » .....	44
<b>Figure III.5 :</b> Diagramme de classe pour l'itération 1 .....	46
<b>Figure III.6 :</b> Diagramme de séquence de cas d'utilisation « inscription ».....	47
<b>Figure III.7 :</b> Diagramme de séquence de cas d'utilisation « ajouter un cours » .....	49
<b>Figure III.8 :</b> Diagramme de séquence de cas d'utilisation « consulter un cours ».....	51
<b>Figure III.9 :</b> Diagramme de séquence de cas d'utilisation « faire un QCM ».....	53
<b>Figure III.10 :</b> Diagrammes de classe pour l'itération2.....	55
<b>Figure III.11 :</b> Diagramme de séquence de cas d'utilisation « envoyer un message ».....	56
<b>Figure III.12 :</b> Diagrammes de classe pour l'itération 3.....	58

### Chapitre IV :

<b>Figure IV.1:</b> Interface Glass Fish .....	60
<b>Figure IV.2 :</b> Interface de WampServer.....	61
<b>Figure IV.3:</b> Interface de PhpMyAdmin.....	62

<b>Figure IV.4 :</b> Interface NetBeans.....	63
<b>Figure IV.5:</b> Interface page d'accueil .....	69
<b>Figure IV.6:</b> Interface page d'authentification .....	70
<b>Figure IV.7:</b> interface « ajouter un cours ».....	70
<b>Figure IV.8:</b> interface « ajouter un QCM » .....	71

## **Introduction générale**

Depuis que l'Internet a été adopté et développé comme un moyen de communication par les établissements d'enseignement dans les années 1970, les universitaires ont été conscients de leur potentiel massif en tant qu'outil d'apprentissage. Au cours des dernières années, les gouvernements des pays développés et sous-développés sont devenus de plus en plus enthousiasmés par les possibilités d'apprentissage en ligne pour offrir une éducation rentable, facilement accessible et toujours actuelle à tous les âges et milieux sociaux, quel que soit le temps et la géographie.

A l'heure où l'apprentissage tout au long de la vie est considéré essentiel au succès continu de la société moderne. Le « e-Learning » est actuellement adapté comme la seule solution viable au problème de la fourniture des ressources nécessaires pour faciliter l'apprentissage à long terme.

Bien que beaucoup ait été dit et écrit sur le sujet de l'apprentissage en ligne, il y a peu de conclusions définitives à tirer de lui. Les livres sont écrits, des groupes Internet sont formés et des conférences ont lieu, mais nous semblons toujours incapables de définir vraiment comment, quand ou où l'e-learning devrait être utilisé. Dans le but de mieux interpréter ce concept il nous a été proposé mettre en place une application d'enseignement à distance, tout en optant pour la technologie Java Enterprise Edition et ceci afin de profiter de langage Java et des avantages de sa plateforme discuté plus loin dans ce projet.

Afin de parvenir à atteindre les objectifs attendus nous avons réparti notre travail en quatre parties

**Chapitre I :** Présentation de la technologie Java Enterprise Edition.

**Chapitre II :** Etat de l'art du E-Learning.

**Chapitre III :** Analyse et conception.

**Chapter IV:** La Réalisation.

## I.1 Introduction :

De nos jours, le génie logiciel nous offre des nouvelles technologies qui ont les potentialités de révolutionner le monde des conceptions des systèmes informatiques, un bon logiciel ne verra jamais le jour sans utiliser une bonne architecture logicielle qui respecte le critère de fiabilité, sécurité, portabilité ....

Dans ce chapitre on va étudier une technologie d'architecture logicielle (Java Entreprise Edition) qui est nécessaire pour la réalisation des applications des entreprises.

## I.2 Conception architecturale [1] :

### I.2.1 Architecture logicielle :

L'architecture logicielle décrit d'une manière symbolique et schématique l'organisation d'un système et sa décomposition en sous-systèmes ou composants.

- Elle se consacre à structurer et à concevoir une application à partir de ses spécifications fonctionnelles
- Elle structure et décompose de façon logique chaque application en couches
- Elle introduit les notions et concepts de découpage en couches, modules, composants, design patterns et Framework.

### I.2.2 Utilité d'une architecture logicielle :

**Compréhension** : facilite la compréhension des grands systèmes complexes en donnant une vue de haut-niveau de leur structure et de leurs contraintes. Les motivations des choix de conception sont ainsi mises en évidence

**Réutilisation** : favorise l'identification des éléments réutilisables, parties de conception, composants, caractéristiques, fonctions ou données communes

**Construction** : fournit un plan de haut-niveau du développement et de l'intégration des modules en mettant en évidence les composants, les interactions et les dépendances. Elle doit permettre aux développeurs de travailler sur des parties individuelles du système en isolation

**Evolution** : met en évidence les points où un système peut être modifié et étendu. La séparation composant/connecteur facilite une implémentation du type « Plug-and-Play ».

**Analyse** : offre une base pour l'analyse plus approfondie de la conception du logiciel, analyse de la cohérence, test de conformité, analyse des dépendances.

### I.2.3 Structuration des applications :

En architecture logicielle la structuration d'un système peut être conçue selon différentes vues :

#### I.2.3.1 Vue en couches « vue logique » :

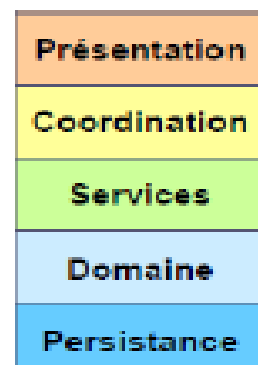
Elle se concentre sur la modélisation des principaux éléments d'architecture et mécanisme logiciels, Elle comprend en particulier les modèles d'analyse et de conception du système, le modèle de référence est le modèle à 5 couches, qui s'applique aux applications munies d'une interface graphique manipulant des données persistantes. La structuration des applications en couches permet

- De maîtriser la complexité des applications (développement, échanges entre les applications, interactions entre objets).
- D'optimiser les temps de développement, en factorisant certaines briques applicatives.
- D'isoler les problématiques d'enchaînements de processus en définissant et en délimitant le périmètre du contrôle de l'intégrité transactionnelle (locale et distribuée).
- De favoriser la communication :
  - À l'intérieur d'une application, en structurant les échanges entre les différentes couches.
  - Entre les applications en précisant les principes de communication liée aux couches de diverses applications.

#### ❖ Le modèle en 5 couches :

La structuration des applications se traduit par une décomposition logique de chaque application en 5 couches ou Chaque couche a ses pro  
située en dessous d'elle :

- La couche Présentation
- La couche Contrôleur
- La couche Services
- La couche Domaine
- La couche Persistance



**Figure I.1** : Le modèle à 5 couches.

**✓ La couche Présentation :**

La couche Présentation gère et assure l'affichage de l'interface graphique utilisateur ou les interfaces Homme-Machine (IHM : fenêtres, pages, composants graphiques...)

Elle intègre principalement :

- La gestion du domaine visuel
- L'interaction avec les utilisateurs
- L'interception des événements utilisateurs et l'appel à la couche Contrôleur
- La gestion du multicanal (web, voix, mobile, fax)
- Les services de portail (agrégation d'IHM, bouquets de services)
- Les services d'impression (impressions PDF, gestion de Template...)

On distingue trois catégories d'IHM pour les applications interactives :

- **Client léger :**

Dans ce modèle, aucun déploiement n'est réalisé sur le poste client à l'exception d'un navigateur Web, Les différents écrans de l'application sont générés en temps réel côté serveur et téléchargés par le poste client.

- **Client lourd :**

Dans ce modèle, l'ensemble des écrans de l'application sont stockés ou générés sur le poste client et doivent avoir été déployés sur celui-ci préalablement à l'exécution, ce type de client n'impose à priori pas de restriction sur le contenu et l'ergonomie des écrans.

- **Client riche (Smart Client) :**

Ce modèle constitue un compromis entre le client léger et le client lourd il présente une ergonomie comparable à celle d'un client lourd tout en limitant les problématiques de déploiement inhérentes à ce dernier.

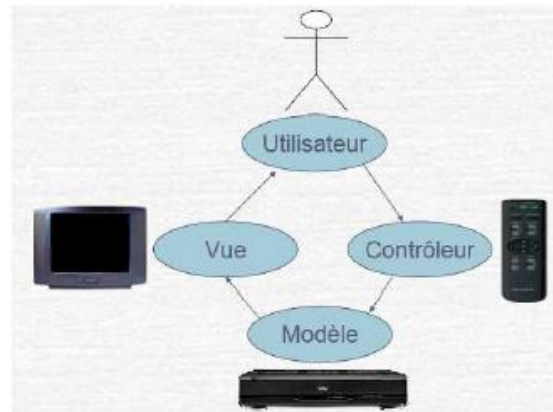
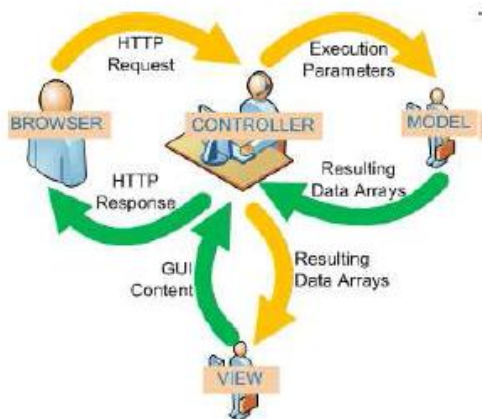
**✓ La couche Contrôleur :**

La couche Contrôleur gère le contrôle de la cinématique des écrans et l'invocation des appels de services, elle détecte les erreurs et les exceptions qui peuvent être levées, les habilitations et les droits d'accès. L'architecture applicative de gestion des interactions utilisateur est généralement mise en œuvre autour du motif de conception MVC (Modèle-Vue-Contrôleur)

**Modèle-Vue-Contrôleur(MVC) :**

En anglais design pattern, il s'applique particulièrement bien à la conception d'applications Java EE.

- Le Modèle représente l'ensemble des composants qui sont chargés de réaliser des appels à la couche services et de mettre les résultats de l'appel à la disposition de la Vue
- La Vue représente l'interface utilisateur.
- Le Contrôleur gère la synchronisation entre la Vue et le Modèle. Le contrôleur réagit aux actions de l'utilisateur en effectuant les actions nécessaires sur le Modèle. Il surveille les modifications du modèle et informe la Vue des mises à jour nécessaires.



**Figure I.2 :** Le modèle vue contrôleur.

**✓ La couche Services :**

La couche Services correspond aux traitements qu'effectue l'application, elle représente l'implémentation de la logique des cas d'utilisation (use-case fonctionnels), elle doit implémenter la logique métier, et gérer les services métiers qui enchaînent des règles métiers (processus métier) et des appels à la couche Domaine.



**✓ La couche Domaine :**

La couche Domaine gère l'intégrité du modèle « métiers ». Elle recense les objets métiers manipulés par l'application, elle intègre principalement :

- La gestion des règles métiers « élémentaires » (sans état, sans processus).
- La fourniture des moyens d'accès à l'information (SGBDR, Mainframe...).
- Le respect des propriétés transactionnelles de la couche persistance.

**✓ La couche Persistance :**

Elle contient un niveau d'abstraction de données les DAO (Data Access Object) qui prennent en charge l'accès à la source de données (SGBDR, fichiers XML, ...). Elle offre les fonctionnalités de base qui permettent :

- De créer, rechercher, modifier et supprimer des composants objets métiers dans le respect des propriétés transactionnelles classiques.
- D'utiliser le mécanisme de projection objet vers relationnel (zapping Objet / Relationnel) qui consiste en la transformation de la représentation des données en une représentation objet.
- D'offrir le support, des contextes transactionnels issus de la couche domaine.

**✓ La couche transverse :**

La couche sécurité n'est pas une couche isolée mais transverse aux autres couches, elle offre un service de sécurité :

- Authentification des utilisateurs et contrôle des habilitations au niveau des services IHM, sécurisation des traitements (authentification, habilitations grosse maille et habilitations fines...).
- Sécurisation des échanges, sécurisation des données.

**I.2.3.2 Vue en niveaux « vue physique » :**

La vue en niveaux donne une vision plus « physique » de la structuration de l'application. Les niveaux (ou tiers) peuvent être répartis physiquement sur différents composants matériels. On identifie un changement de « niveau » dès qu'un module logiciel doit passer par un intermédiaire de communication (middleware) pour en invoquer un autre. Si l'utilisation du middleware est en général transparente pour les développeurs, elle n'est pas sans impact sur

l'architecture. Pour cela la maîtrise des caractéristiques (clients/serveurs) s'avère indispensable.

### **Architecture Client/serveur [2] :**

L'architecture client/serveur désigne un mode de communication entre plusieurs ordinateurs d'un réseau qui distingue un ou plusieurs clients du serveur, chaque logiciel client peut envoyer des requêtes à un serveur. Un serveur peut être spécialisé en serveur d'applications, de fichiers, de terminaux, ou encore de messagerie électronique.

Les acteurs principaux d'une architecture Client-Serveur sont en général au nombre de Trois :

#### **Le client :**

Processus demandant l'exécution d'une opération à un autre processus serveur par l'envoi d'un message contenant le descriptif de l'opération à exécuter et attendant la réponse à cette opération par un message en retour.

#### **Le serveur :**

Processus accomplissant une opération sur demande d'un client et transmettant la réponse à ce dernier.

#### **Le middleware :**

Ensemble des services logiciels construits au-dessus d'un protocole de transport afin de Permettre l'échange de requêtes et des réponses associées entre client et serveur de manière transparente.

### **❖ Fonctionnement d'un système client/serveur :**

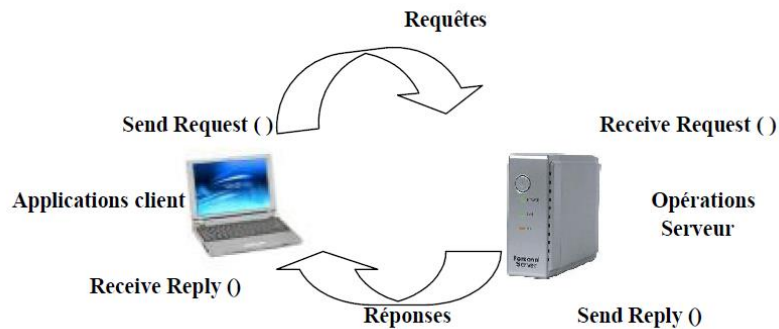
Le client/serveur est avant tout une technique de dialogue entre deux processus, qui s'appuient sur l'envoi de requêtes et de réponses en sens inverse. Ces deux processus ne sont pas identiques mais forment un système coopératif.

- Le client émet une requête vers le serveur grâce à son adresse IP et le port, qui désigne un service particulier du serveur
- Le serveur reçoit la demande et répond à l'aide de l'adresse de la machine client et son port.

La communication entre le client et le serveur peut être en mode synchrone ou asynchrone.

**Mode synchrone :** Dans ce type de dialogue, le client ne peut pas émettre d'autre requête pendant que le serveur exécute une opération pour lui. Autrement dit, les commandes d'émission et de réception sont bloquantes.

**Mode asynchrone :** Dans ce type de dialogue, au moins une des commandes d'émission ou de réception est non bloquante, il est donc géré par une file d'attente. Ce dialogue permet au client d'effectuer une autre requête pendant que le serveur exécute une opération pour lui.



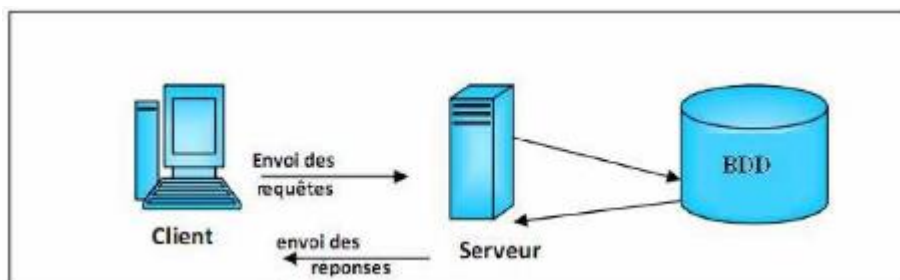
**Figure I.3 :** Dialogue client serveur

#### ❖ Les architectures à niveaux du client/serveur :

Le terme « *niveau* » est utilisé pour décrire un choix architectural grâce à un découpage de la logique applicative entre des clients et des serveurs. Cette répartition des charges a donné naissance à deux modèles d'architectures de base pour le client/serveur :

##### ✓ Le modèle deux-tiers :

Dans ce modèle, l'interface graphique se situe sur le poste client et la base de données est localisée sur le serveur. Pratiquement, toute la charge applicative s'exécute sur le client qui devient de ce fait un client lourd (c'est le modèle orienté client).



**Figure I.4 :** Conception a deux niveaux

### ✓ Le modèle trois-tiers :

Dans ce modèle, il existe un niveau intermédiaire. L'architecture est généralement partagée entre le client (demandeur de ressources), le serveur d'application (chargé de fournir la ressource mais faisant appel à un autre serveur) et le serveur secondaire (généralement un serveur de bases de données fournissant le service au premier serveur). Le client devient ainsi léger.

Ce modèle répartie donc la charge applicative en trois :

- Les clients qui exécutent la logique de l'interface graphique utilisateur.
- Le serveur d'application qui exécute la logique applicative.
- La base de données (c'est le modèle orienté serveur).

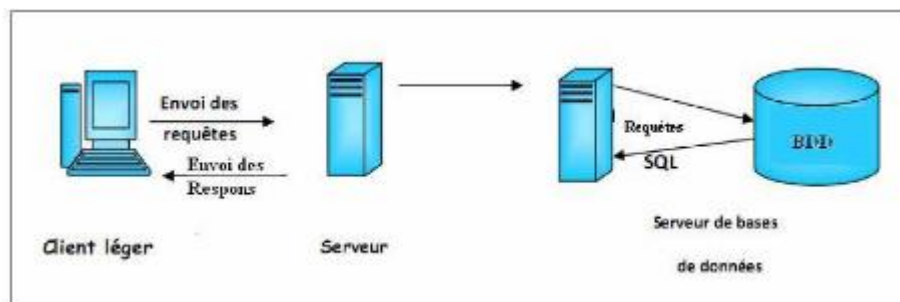


Figure I.5 : Conception a trois niveaux

### ✓ L'architecture multi-tiers (à base de composants) :

Dans l'architecture 3-tiers, pour répondre à une demande de service émise par un client, un serveur peut utiliser les services d'un ou plusieurs autres serveurs afin de fournir son propre service (d'où le multi-tiers). C'est le modèle N tiers qui est mise en œuvre dans le cadre des projets web.

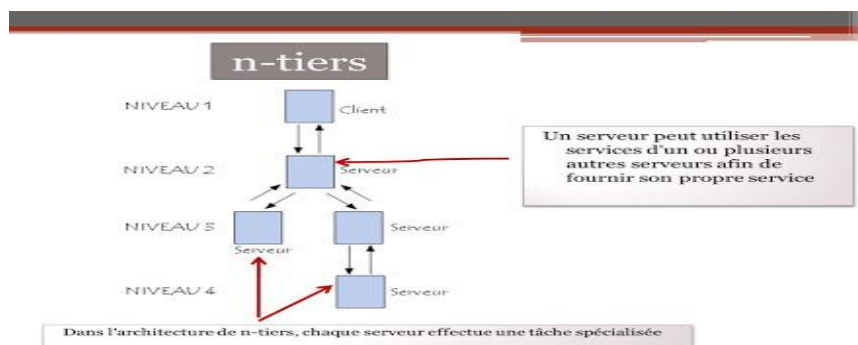


Figure I.6 : Architecture à multi niveaux

### **I.3 Java Entreprise Edition :**

#### **I.3.1 Présentation de JAVA EE [3] :**

Java EE est une version de java destinée aux applications d'entreprise, proposée par la société Sun, portée par un consortium de sociétés internationales, dans le but de définir un standard de développement d'applications d'entreprises multi-niveaux, basées sur des composants. Pour désigner l'ensemble constitué des services (API) offerts et de l'infrastructure d'exécution, nous parlons généralement de « plate-forme Java EE ».

JAVA EE comprend notamment :

- ❖ Les spécifications du serveur d'application, c'est-à-dire l'environnement d'exécution, Java EE définit les rôles et les interfaces pour les applications ainsi que l'environnement dans lequel elles seront exécutées.

- ❖ Des services au travers d'API, c'est-à-dire des extensions Java indépendantes permettant d'offrir en standard un certain nombre de fonctionnalités. Sun fournit une implémentation minimale de ces API appelée Java EESDK (Java EE Software Développement Kit).

JAVA EE s'appuie entièrement sur le Java, il bénéficie des avantages et inconvénients de ce langage, ce qui engendre une bonne portabilité et une maintenabilité du code. L'architecture JAVA EE s'appuie sur des composants différents, interchangeables et distribués, ce qui permet :

- D'étendre l'architecture d'une façon simple.
- Un système qui repose sur Java EE peut posséder des mécanismes de haute Disponibilité, afin de garantir une bonne qualité de service.
- La facilité de la maintenabilité des applications.

#### **I.3.2 L'architecture de JAVA EE :**

JAVA EE ajoute de nombreuses couches de niveau entreprise au-dessus de la plateforme Java Standard Edition. Chaque couche est conçue pour supporter une différente technologie de développement.

**1-Technologie web application :** est une technologie liée à la création des interfaces web dynamique, par exemple : JSP (Java Server Page) et servlet.

**2-Technologie entreprise application :** est une technologie liée à la logique de business : EJB (Entreprise Java Bean), java Mail, JMS (Java Message Service) ...etc.

**3-Technologie web services** : est une technologie nécessaire au développement des applications adhérentes au paradigme SOA (Service Oriented Architecture). Elle permet l'échange des données entre application et systèmes hétérogènes dans des environnements distribués.

**4-Technologie management et sécurité** : est une technologie liée à la gestion de la technologie d'entreprise afin de réaliser l'accès et l'échange d'informations entre machine et service distribués : JAAS (Java Authentication and Authorization Service), JCA (Java Connector Architecture).

### **I.3.3 Les Couches logicielles Java EE [4] :**

La plateforme Java EE utilise un modèle d'application distribuée multi-tiers pour des applications d'entreprise. La logique d'application est divisée en composants selon la fonction. Les divers composants qui forment l'application Java EE sont installés sur des machines différentes en fonction du niveau de l'environnement Java EE multi tiers auquel le composant d'application appartient.

L'architecture d'une application Java EE se découpe idéalement en au moins trois couches :

#### **❖ La couche interface utilisateur ou couche présentation.**

C'est la couche qui dialogue avec l'utilisateur, via une interface graphique swing, une interface console ou une interface web. Cette couche permet par l'intermédiaire de la couche dite « business » d'envoyer et de recevoir des informations de la couche d'accès aux données. La présentation peut être exploitée sous plusieurs formes : client lourd (application graphique ou textuelle), client léger (navigateur web), applications mobiles

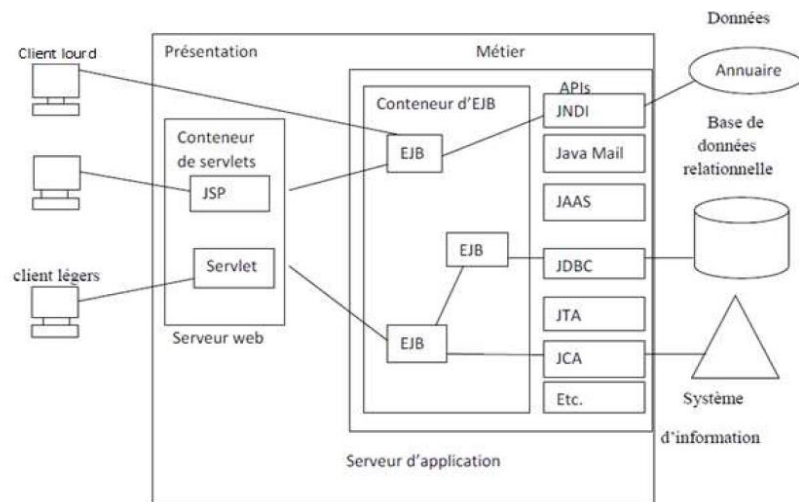
#### **❖ La couche métier :**

C'est la partie de l'architecture 3tiers qui implémente le métier de l'entreprise. Ce niveau permet le traitement des interactions entre les utilisateurs et le système tout en tenant compte des règles de gestion et de contrôle de l'entreprise.

Cette couche met en œuvre des services métiers et applicatifs, elle permet de faire un relais entre la couche de présentation et celle d'accès aux données.

### ❖ La couche d'accès aux données ou couches de persistances :

Cette couche regroupe les informations manipulées par la plateforme Java EE, ces informations peuvent être sous différentes formes telles que les bases de données, des fichiers et tout autre type de stockage de données (texte, XML, Bases de Données, Excel).



**Figure I.7 :** Architecture et composants de Java EE

### I.3.4 Spécification de la plateforme Java EE :

Java EE compte des spécifications qu'on peut regrouper en deux grandes parties :

- Les APIs.
- Les serveurs d'application.

#### I.3.4.1 Les java persistances APIs :

Les APIs sont des extensions Java, forment une collection d'outils logiciels permettant d'offrir en standard un certain nombre de fonctionnalités. Sun fournit une implémentation minimale de ces APIs nommée Java EE SDK (Java EE SOFTWARE DEVELOPMENT KIT), ces APIs peuvent être regroupées en trois groupes principaux :

- Composants.
- Services d'infrastructures.
- Services de communication

- **Les composants :**

Un composant est un fichier simple, contenant généralement du code compilé. Il est possible de créer des composants avec la majorité des langages. Aussi il peut être vu comme unité regroupant un certain nombre de fonctionnalités qui peuvent être appelées depuis un programme externe ou client.

- ❖ **Composants exécutés sur le client (client et applet) :**

Peuvent être application client ou applet.

- **Client web :**

Un client web est appelé client léger qui n'a généralement pas de requête de BDD, l'utilisation de client web permet de charger les opérations lourdes au niveau de l'entreprise Bean et sont exécutées sur le serveur Java EE ou ils peuvent exploiter les avantages des technologies Java EE (sécurité, la vitesse, les services...) sur le serveur, un client a besoin de deux fonctionnalités principales :

- Les pages web dynamiques avec les différents langages à balisage (HTML, XML,...), ces pages sont générés par les composants web au niveau du client web.
- Le navigateur web qui affiche les pages reçues par le serveur.

- **Application client**

L'application client fonctionne au niveau de la couche métier, et peut accéder directement à l'entreprise Bean, généralement les utilisateurs possèdent une interface graphique(GUI) pour exécuter l'application client (en utilisant l'API Swing ou Abstract Windows toolkit).

- **Applet :**

Une applet est une petite application java qui a les caractéristiques de pouvoir s'exécuter via un navigateur supportant java ou dans une l'applet viewer du JDK. En effet, la plupart des navigateurs aujourd'hui dispose d'une machine virtuelle capable d'interpréter les byte code inclus dans une page HTML.

- ✓ **Composants exécutés sur le serveur :**

Les composants web sont des servlets et les pages jsp (java server page) :

- **Servlet**

Une servlet est un composant coté serveur, considérée comme une classe java qui permet une communication avec un protocole « requête/réponse ». C'est un programme qui s'exécute coté serveur. Elle reçoit une requête du client, elle effectue des traitements et renvoie le résultat, l'utilisation des servlets permet la génération des pages web dynamiques qui utilisent



le protocole HTTP, ou aussi n'importe quel autre protocole.

✓ **Les pages JSP (Java server page) :**

Les JSPs : ( java Server Pages) est une technologie qui permet d'écrire très facilement des pages dynamiques en insérant des portions de code java dans une page html, destiné à l'origine à la présentation des données, une page JSP permet de définir le design d'une page (en HTML) et d'y intégrer les données (grâce aux portions de code java). Cette technologie favorise donc la séparation de l'apprentissage et du contenu, d'un point de vue plus technique, une page java est donc constituée de code HTML dans lequel sont insérées des codes java délimitées par des balises particulières, l'intégralité du code est portable puisque tout comme un code java, il sera compilé et exécuté comme Servlet.

A la différence d'une Servlet, il sera par contre compilé au vol (lors de chaque invocation par un moteur de JSP. Enfin, les pages JSP offrent les Possibilités d'importer des classes java et d'accéder à des objets java.

❖ **Composants métier(EJB) :**

✓ **Entreprise JavaBeans(EJB) :**

Est une architecture de composants logiciels cotés serveur pour la plateforme de développement Java EE. Cette architecture propose un cadre pour créer des composants distribués (c'est-dire déployés sur des serveurs distants) écrit en langage de programmation Java hébergés au sein d'un serveur applicatif permettant de représenter des données (EJB dite entité), de proposer des services avec ou sans conservation d'état entre les appels (EJB dit session), ou encore d'accomplir des tâches de manière asynchrone (EJB dit message). Tous les EJB peuvent évoluer dans un contexte transactionnel.

L'architecture EJB prévoit de manipuler les composants EJB dans un environnement distribué, assurant une complète comptabilité avec tous les types de clients : servlet, page jsp, autre EJB, applet...

Pour assurer la communication réseau entre différents composants distribués les EJB utilisent un mécanisme basé sur RMI. Ce mécanisme est implémenté dans une API qu'offre à une machine virtuelle java les mécanismes nécessaires pour invoquer des méthodes sur d'autre JVM, qu'elles soient ou non sur le même réseau.

**✓ Les différents types d'EJB :**

Il existe trois types des EJB :

- **Les beans entités :**

Ils correspondent à des objets réels tels que ces objets sont généralement des enregistrements persistants stockés dans une BDD. Les beans entités peuvent être des moyens de modélisation des concepts métier pouvant être désignés par des noms, par exemple un bean entité peut représenter un client ou un objet stocké.

- **Les beans session :**

Ceux sont une extension de l'application client et ils sont responsables de la gestion des processus ou des tâches.

- **Les beans orientés message :**

Un bean session fournit une interface distante afin de définir les méthodes que l'on peut invoquer, alors que bean orienté message attend des messages asynchrones spécifiques auquel il répond, donc généralement, un EJB récupère également les données stockées, les traite et les envoie au programme du client.

- **Services d'infrastructures :**

On trouve plusieurs services d'infrastructure on définit les APIs suivantes :

**✓ JDBC « Java Database Connectivity »:**

C'est une API qui permet aux programmes java d'interagir avec les bases de données SQL. Les serveurs fournissent en plus un mécanisme qui permet de réutiliser les connexions créées avec les utilisations.

**✓ JCA « Java EE Connector Architecture »:**

C'est une API de connexion au système d'information de l'entreprise notamment tel que les ERP (Entreprise Resource Planning).

**✓ JMX « Java Management Extension » :**

Permet de développer des applications WEB de supervision d'applications à partir des extensions.

- **Services de communication :**

- ✓ **Les conteneurs :**

Un conteneur est une entité qui fournit aux composants la gestion de cycle de vie, la sécurité, les services de déploiement et d'exécution. Chaque type de conteneur (EJB, WEB, JSP...) fournit aussi des services spécifiques à ces composants.

Il existe plusieurs types de conteneurs qu'on peut citer :

- a) Conteneur web :**

Il est composé de deux grandes parties : un moteur de servlets et un moteur de JSP, ce conteneur permet d'exécuter les composants JSP et les servlets de l'application Java EE.

- b) Conteneur EJB :**

Ce type de conteneur permet de gérer l'exécution d'EJB.

- c) Conteneur d'application client**

Il permet de gérer l'exécution des composants de l'application client. Il s'exécute sur le client.

- d) Conteneur d'applets**

C'est une composition de deux parties : navigateur web et java plug-in qui fonctionnent ensemble sur le client, donc ce conteneur permet de gérer l'exécution des applets.

#### **I.3.4.2 Les serveurs d'application :**

Les API Java EE ont été projetés pour travailler avec un particulier type de serveur appelé Java EE Application Server. Un serveur d'application est une couche software résidente sur une machine serveur qui fournit les services que la technologie Java EE nécessite. Il y a plusieurs serveurs d'applications. Parmi les produits commerciaux on rappelle Bean Web Logic, IBM Websphere, Sun Application Server, etc. Parmi les produits libres le plus connu est JBOSS. Les différences principales entre les différents serveurs d'applications sont relatifs aux opérations de déployé, clustering, ...etc.

Par contre toutes les fonctionnalités qui concernent strictement le fonctionnement des applications Java EE adhèrent aux spécifications proposées par le Sun, un serveur d'application s'installe et se lance comme un normal serveur web (de fait il met à disposition des services typiques d'un serveur web). En plus il possède un panneau d'administration accessible par un poste distant à travers lequel on peut définir les différents services.

Les serveurs d'application se sont développés depuis la création de Java EE. Il existe principalement 2 grandes catégories de serveurs :

- **Open Source :**

Évolue grâce à la communauté par exemple Tomcat : Apache, Jonas : ObjectWeb .

- **Propriétaire :**

Évolue selon l'éditeur par exemple WebSphere : IBM, WebLogic :

**Le rôle d'un serveur d'application :**

Le rôle d'un serveur d'application est d'héberger des applications distribuées, fabriquées à base de composants Java (Servlet, JSP, EJB) et de les rendre accessible à des clients web (navigateurs) et à des applications d'entreprise écrites en java. Le serveur d'application doit être performant et fiable pour satisfaire aux exigences des applications d'entreprise, le serveur offre différents services utiles pour les applications :

**a. Service de nommage :**

C'est le service le plus important dans un serveur d'application implémentant l'interface. SPI (Service Provider Interface) est utilisée généralement le protocole LDAP (Light weight Directory Access Protocol). Ce dernier fournit des mécanismes de recherche, de localisation et de liaison et il est différent du DNS.

**b. Services de gestion de transaction :**

Il permet de gérer des transactions locales mais surtout il prend en charge la gestion des transactions distribuées, indispensable pour les applications Java EE.

**c. Service de gestion de la disponibilité des applications :**

Il permet de gérer un grand nombre d'accès et de requêtes simultanés.

**d. Service de sécurité :**

L'administrateur du serveur d'application peut configurer des domaines de sécurité en définissant des utilisateurs (nom, mot de passe), des groupes d'utilisateurs, puis les permissions accordées sur différents éléments par rapport aux composants et/ou aux ressources de l'application Java EE

**e. Service d'administration :**

L'administrateur a la possibilité de consulter, de modifier, et d'ajouter des éléments à la configuration des applications d'entreprise et du serveur d'application.

**f. Service d'accès aux données :**

Ce service est une implémentation des extensions de l'API JDBC

**g. Service de gestion de messages :**

C'est une implémentation de l'API JMS, il permet de gérer une file de messages de traitements asynchrones.

**I.3.5 Les avantages d'utiliser Java EE :**

L'utilisation de Java EE pour développer et exécuter une application représente plusieurs avantages :

- Une architecture d'application basée sur les composants qui permet un découpage de l'application et donc une séparation des rôles lors du développement.
- La possibilité de s'interfacer avec le système d'information existant grâce à de nombreuses API : JDBC, JNDI, JMS, JCA...
- La possibilité de choisir les outils de développement et les serveurs d'applications utilisés qu'ils soient commerciaux ou libres.

**I.4 Conclusion :**

Dans ce chapitre, nous avons abordé le concept de l'architecture logiciel ainsi on a défini la mise en œuvre et le fonctionnement de l'architecture client /serveur, on a aussi présenté l'architecture et les composant de la technologie java entreprise édition en définissant certain concepts qui seront utilisées pour le développement de notre application, dans le chapitre qui suit nous allons présenter un état d'art de E-Learning.

## **II.1 Introduction :**

Les bouleversements technologiques ont donné naissance à beaucoup de formes d'assistance à la formation, aujourd'hui, on parle d'E-Learning. Ce type de formations permet de développer les compétences des apprenants, tout en rendant le processus d'apprentissage indépendant du temps et du lieu. Afin de mieux comprendre ce concept

Nous allons introduire dans ce chapitre l'état de l'art du E-learning.

## **II.2 La formation à distance [5] :**

Les premiers cours « à distance » sont apparus, avec la révolution industrielle, à la moitié du 19ème siècle en Angleterre et aux Etats-Unis, Certains auteurs attribuent à Isaac Pittman la mise en application du premier cours par correspondance en Angleterre autour des années 1840. Ces modalités nouvelles de cours se distinguaient de l'enseignement traditionnel par une séparation physique entre l'enseignant et l'étudiant, le contact s'établissant par correspondance.

Cela fait plus d'un siècle que la formation à distance a fait son apparition sous la forme d'enseignement par correspondance. L'expression la plus utilisée est celle d'EAD (Enseignement À Distance) qui désigne une situation où la transmission des connaissances se fait en dehors de la relation face-à-face Enseignant-Apprenant.

Cette expression est couramment utilisée par les formateurs car elle caractérise un point de vue enseignant. Un terme plus général pour caractériser cette situation est FAD (Formation À Distance) qui permet d'inclure à la fois le terme d'Enseignement à Distance (point de vue enseignant) et Apprentissage à Distance (point de vue apprenant).

### **II.2.1 Qu'est-ce que la FAD :**

La formation à distance est un système de formation qui permet de se former sans se déplacer sur le lieu de formation et sans la présence physique d'un formateur. La transmission des connaissances et les activités d'apprentissage se situent en dehors de la relation directe en face à face, dite « en présentiel » entre l'enseignant et l'apprenant.

La formation à distance recouvre plusieurs modalités : cours par correspondance et e-learning.

### **II.2.2 Pourquoi la FAD :**

La formation à distance est un système de formation très souple, tant dans l'espace que dans le temps. Le stagiaire peut apprendre à son rythme, chez lui, dans un cyber café ou ailleurs, quand il a un moment de libre.

La FAD s'adapte bien à la diversité des besoins en formation. Notamment dans les domaines techniques et industriels où l'évolution rapide des technologies exige de plus en plus la création de circuits directs entre formateurs et apprenants.

C'est un enseignement à distance qui permet à chacun de se former en toute autonomie tout en bénéficiant du support et des conseils d'experts à tout moment.

### **II.3 Formation Ouverte et à Distance (FOAD) :**

Il existe aujourd'hui une définition officielle donnée à la FOAD dans la circulaire du 20 juillet 2001 de la Délégation générale à l'emploi et à la formation professionnelle (DGEFP) : il s'agit d'un dispositif souple de formation, organisé en fonction de besoins individuels ou collectifs, qui comprend des apprentissages individualisés, l'accès à des ressources et à des compétences locales ou à distance, et qui n'est pas exécuté nécessairement sous le contrôle permanent d'un formateur.

La FOAD s'appuie sur les NTIC (Nouvelles Technologies de l'Information et de la Communication), notamment l'Internet, l'Intranet et l'Extranet pour fournir une formation à distance favorisant l'individualisation des formations (les apprenants ont de plus en plus besoin d'un apprentissage centré sur eux-mêmes) et les communications entre formateurs. On parle aussi de E-Learning (Ou apprentissage en ligne ou électronique) ou de E-Formation.

### **II.4 Différences entre la formation ouverte et la formation à distance :**

La formation à distance n'a rien à voir avec la formation ouverte, puisqu'au départ le terme « Ouvert » signifiait « ouvert sans diplôme », c'est-à-dire accessible à tout le monde. Lorsque l'Open Université a été créée, on pouvait s'y inscrire sans prérequis, sans diplômes initiaux. La notion de « formation à distance » intègre à son tour deux notions distinctes : l'enseignement à distance et l'apprentissage à distance. L'apprentissage dépend d'abord et



avant tout de l'apprenant, alors que l'enseignement n'implique pas nécessairement que l'apprenant apprenne.

La formation ouverte, l'exemple type est le centre de ressources. Les termes « formation ouverte et à distance » représentent des choses extrêmement différentes et des dispositifs extrêmement variés, qui de plus ont évolué au cours du temps. Ces dispositifs n'ont qu'un point commun entre eux : ils brisent la règle des trois unités de la tragédie classique, qui est aussi celle de l'enseignement traditionnel.

S'il y a rupture de l'unité de lieu, nous parlerons de formation à distance, s'il y a rupture de l'unité de temps et d'action, même dans un même lieu, nous parlerons de formation ouverte.

S'il y a rupture des deux, on parlera de formation ouverte et à distance.

## **II.5 Présentation du E-learning :**

### **II.5.1 Définition du E-learning [6] :**

Le E-learning est un processus d'apprentissage à distance s'appuyant sur des ressources multimédias, qui permet à une ou plusieurs personnes de se former à partir de leur ordinateur en utilisant les nouvelles technologies de l'information et de la communication(NTIC). Les supports multimédias utilisés peuvent combiner du texte, des graphismes en 2 ou 3 dimensions, du son, de l'image, de l'animation et même de la vidéo.

Plusieurs termes sont utilisés pour traduire le terme E-Learning

#### **E-Formation :**

E-Formation est à l'origine un sous-ensemble de la FOAD, qui s'appuie sur les réseaux électroniques. Aujourd'hui le concept d'E-Learning est de plus en plus employé, attestant de l'évolution fondamentale de ce domaine de formation.

#### **Apprentissage en ligne :**

Mode d'apprentissage basé sur l'utilisation des nouvelles technologies, qui permet l'accès à des formations en ligne, interactives et parfois personnalisées, diffusées par l'intermédiaire d'Internet, d'un intranet ou autre média électronique, afin de développer les compétences, tout en rendant le processus d'apprentissage indépendant de l'heure et de l'endroit.

### II.5.2 Pourquoi le E-Learning :

Le choix du E-Learning s'explique par le fait de vouloir abolir les notions de temps et d'espace, les apprenants ne doivent pas se trouver tous en un même lieu, à des moments bien déterminés. De plus, la qualité de la formation et l'efficacité de son processus peuvent être considérablement accrues par l'interactivité en permettant l'adaptation du contenu de la formation à la situation pédagogique de l'apprenant.

Enfin, la formation à distance revêt plusieurs avantages économiques notamment :

- La possibilité d'amortir sur un plus grand nombre d'apprenants les coûts liés à la création de contenus.
- L'automatisation des fonctions de gestion de l'enseignement (administration des cours, contrôle des connaissances, etc.)
- Les économies réalisées sur les déplacements des apprenants et des enseignants
- Les économies réalisées par l'utilisation optimale des formateurs, qui n'interviennent qu'en cas de difficulté et dont le coût peut être réparti sur un plus grand nombre d'apprenants.

Ces avantages peuvent largement compenser les coûts d'acquisition, de maintenance et de développement des logiciels utilisés lors de la mise en œuvre de l'E-Formation.

### II.5.3 Applications du E-Learning :

Aujourd'hui, le E-Learning est lié spécialement à deux secteurs :

- Des grandes entreprises qui l'utilisent avec le KM (Knowledge Management ou management des connaissances) pour former et actualiser les connaissances de ses employés de manière plus rapide.
- Des universités qui offrent des alternatives de formation à distance.

L'univers du E-Learning s'étend à des différents prestataires :

- Les entreprises qui offrent des solutions technologiques : infrastructure, plates-formes.
- Les sociétés qui fournissent des contenus pédagogiques qu'elles ont dans l'étagère.
- Les sociétés qui développent un contenu pédagogique sur mesure.

## **II.6 Les différentes formes de la formation en ligne :**

La formation en ligne se présente sous une multitude de formes. Cette particularité facilite son intégration aux stratégies de formation continue au près des apprenants.

On distingue deux types de formation en ligne :

### **II.6.1 Formation exclusivement en ligne (sans tutorat) :**

Ce type de formation d'avantage individualisé, est proche du service fourni par un CD ou d'une vidéo de formation. C'est un modèle d'apprentissage sans formateur, par lequel l'apprenant communique avec différents serveurs pour chercher de l'information et accéder à des modules d'autoformation sur des sites Web, des CD ou DVD.

#### **Avantage :**

Le grand avantage est que l'apprentissage se fait au rythme de l'apprenant, qui peut avoir différentes façons de parcourir son cours.

L'apprenant peut s'auto évaluer, et faire des pauses lorsqu'il en ressent besoin.

Formation moins coûteuse, elle tend à devenir gratuite.

#### **Inconvénients :**

Une formation qui n'apporte pas beaucoup de valeur ajoutée. En effet elle comporte des inconvénients sur le plan de la qualité pédagogique.

### **II.6.2 Formation exclusivement en ligne (avec tutorat) :**

C'est le modèle de référence des « portails de formation », plutôt destiné au grand public.

L'acheteur paie en ligne sa formation. Un tuteur lui est attribué qui lui propose un programme de travail. Dans ce cas, la formation peut être :

**II.6.2.1 Asynchrone :**

La formation asynchrone est une méthode d'apprentissage s'adaptant aux disponibilités de l'apprenant, celui-ci a accès à un ou plusieurs instruments (exemples : vidéo, texte, logiciel d'apprentissage virtuel) qu'il utilisera à sa guise et il n'est pas obligé d'être connecté en même temps que les autres apprenants et le formateur. Cependant, Le rythme de l'apprentissage est géré par le formateur. L'échange des documents et la participation au discours se feront par voie indirecte (courriel, forum de discussion).

**Les principaux avantages de la formation asynchrone :**

- Les apprenants évoluent à leur propre rythme.
- Ils peuvent adapter l'ordre dans lequel ils appréhendent les éléments du cours.
- Ils peuvent revoir et approfondir certains aspects du cours à leur guise.

**II.6.2.2 Synchrone :**

Dans la formation synchrone les apprenants et le formateur doivent être connectés en même temps. C'est aussi un modèle collectif. Cette pratique demande un équipement de vidéoconférence en salle ou sur poste de travail pour que la diffusion soit en direct.

**Les principaux avantages de la formation synchrone :**

La dynamique de groupe s'installe plus rapidement ;

On peut rapidement créer du contenu prêt à diffuser ;

Utilisation de langage oral et visuel.

**II.6.3 Formation mixte (avec tutorat en ligne) :**

C'est plutôt le modèle des écoles et organismes de formation. En mettant en ligne le contenu de la formation, des tests, des évaluations et un tutorat, cela permet de réduire le temps du présentiel et d'individualiser la formation.

Ce modèle de formation combine donc les éléments de l'apprentissage en ligne et de l'apprentissage traditionnel en classe. Afin de faciliter l'intégration des connaissances acquises à travers une formation.

**Les principaux avantages de la formation mixte :**

L'acquisition des connaissances est meilleure avec la formation mixte ;

Les classes virtuelles en temps réel permettent aux apprenants d'assister aux cours n'importe où.

**II.7 Outils de communication**

Généralement les plates-formes SGC (Système de Gestion de Cours) intègrent un certain nombre d'outils de communication, la liste des outils disponibles dépend de la richesse du SGC. Ces outils sont souvent très utilisés sur le Web et ne sont en rien spécifiques au E-Learning, ils font simplement partie intégrante du SGC et ne nécessitent pas de configuration spécifique sur les postes des apprenants (par exemple, il n'est pas nécessaire de configurer un client messagerie pour profiter de ce service).

On peut citer :

- **La messagerie** (C'est un espace de communication entre les membres du groupe (stagiaires et tuteurs). Ce système permet l'envoi et la réception de mails avec ou sans fichier attaché. Il peut être interne à la plateforme et ne nécessite pas d'avoir un e-mail personnel)
- **Le forum** (Le forum, pouvant être public ou réservé à un groupe, permet aux stagiaires et aux tuteurs : de poster des messages qui seront accessibles à l'ensemble des membres du groupe de formation, de répondre aux messages déjà postés et ainsi engager une discussion sur un sujet donné).
- **Le Chat** (Il permet à l'ensemble des membres du groupe de discuter en temps réel)
- **Les documents pédagogiques partagés** : Une zone commune peut être utilisée pour mettre à disposition du groupe, différents documents, Documents sous forme numérique (Word, Excel, PDF, etc.) pour l'ensemble du groupe.
- **Les news** (La possibilité de créer des news ce qui permettra, par exemple, d'informer ses apprenants de nouveaux rendez-vous, ou de leur communiquer diverses informations)

- **La visioconférence** : Equivalente à l'intervention du formateur en salle face à un groupe, mais ici chacun est devant son ordinateur, à domicile, au travail, ou un centre de ressources...
- **Les sondages** : Le sondage permettra de poser des questions ouvertes ou de proposer des choix de réponses. Ultérieurement, les réponses obtenues sur un cours en particulier pourront être analysées.
- **Le Bloc-notes** : C'est un espace privé à chaque utilisateur qui peut y noter toutes les informations qu'il souhaite : ce qu'il a retenu de sa formation, des actions à mener...etc. Ces notes sont archivées par date et il est possible de les compiler dans une même note.
- **Les Glossaires** (Des glossaires peuvent être créés, destinés à l'ensemble des utilisateurs ou un groupe en particulier. Les différents termes avec leur signification sont enregistrés, un moteur de recherche permet aux utilisateurs de faciliter la recherche d'un terme dans les glossaires).
- **Les Faq** : Cet espace permet aux utilisateurs de trouver éventuellement la réponse à des questions récurrentes.

## **II.8 Les Plate-formes de formation à distance ou LMS (Learning Management System) [7] :**

Une plate-forme de formation à distance est un logiciel d'apprentissage, combiné avec les technologies du Web, qui fournit des fonctionnalités à différents utilisateurs (Apprenants, Formateurs, Auteurs) afin de conduire et d'automatiser le processus de formation à distance d'apprenants. En d'autres termes, c'est un logiciel permettant la consultation à distance d'objets pédagogiques, le télé tutorat et la centralisation de l'apprentissage sur l'individu. Ces plateformes permettent le développement et la distribution de ressources et d'activités pédagogiques sur un environnement d'apprentissage en ligne. La maintenance du système et la gestion des droits d'accès des apprenants aux cours sont assurés par un Administrateur.

### II.8.1 Les acteurs d'un LMS et leurs rôles :

Le nombre d'acteurs et leurs fonctions diffèrent d'une plate-forme à une autre, selon les objectifs et les fonctionnalités offertes par chacune d'entre elles. Certaines se sont contentées de trois acteurs, à savoir l'apprenant, le formateur/créateur de cours et l'Administrateur.

Cependant, six rôles principaux ont été distingués en observant un ensemble de plateformes de formation à distance. Ces rôles sont :

**Concepteur (auteur ou créateur de cours) :** Il établit (pour l'apprenant) un parcours de formation personnalisé et individualisé. Il construit, adapte et maintient un système d'apprentissage. Il peut aussi avoir le rôle de présentateur. Ce rôle consiste à rendre disponible les informations pour l'apprentissage, en d'autres termes, la présentation des informations, la gestion des médias, la clarification des contenus en réponse à des questions, l'analyse et l'évaluation des contenus des documents...etc.

**Formateur (ou tuteur) :** Il accompagne les apprenants durant leurs parcours pédagogiques. Il a la tâche de motiver, d'orienter et d'évaluer les apprenants, comme il les assiste dans l'utilisation des ressources et du matériel mis à leur disposition.

**Apprenant :** Son rôle consiste à transformer les informations en connaissances. Pour cela, il doit gérer ses activités et son temps, exploiter les ressources qui lui sont indiquées de façon à optimiser la quantité d'information qu'il peut en tirer, résoudre des problèmes, échanger des informations et des idées avec d'autres apprenants via les outils de communication, s'auto évaluer et présenter des travaux qui serviront de référence à son formateur pour son évaluation.

**Administrateur :** Un administrateur installe et assure la maintenance du système, il s'occupe des tâches d'administration comme les inscriptions des acteurs dans la plateforme, il gère les droits d'accès, assure la gestion des ressources pédagogiques, ce qui consiste en la classification, l'indexation et l'administration des matériaux pédagogiques.

**Orienteur :** Son rôle est de gérer les acteurs, les événements, la création des cursus de formation, des individus et des groupes. Il s'occupe des inscriptions des apprenants dans une formation donnée et de la constitution de groupes de travail.

**Evaluateur :** Il s'occupe de l'évaluation des groupes et des individus et ceci grâce aux informations fournies par la plate-forme.

### **II.8.2 Exemple de plateforme d'apprentissage en ligne :**

Nous donnons quelques exemples de Système de gestion des cours (S.G.C) :

#### **II.8.2.1 Web CT [8] :**

Web CT (Web course Tools), est une plateforme développée par l'université canadienne British Columbia. Il s'agit d'un environnement d'enseignement/apprentissage destinée à développer et diffuser des télé Cours et à faciliter l'intégration d'un ensemble d'outils de communication, d'évaluation, de gestion, etc.

Web CT est l'une des plates formes commerciales de téléformation les plus vendues dans le monde. Elle est traduite en plusieurs langages utilisés par des milliers d'instruction. Dans WebCT, chaque acteur humain (administration, concepteur de cours, tuteur et apprenant) dispose d'un espace de travail sous forme d'une interface (accessible via un mot de passe) regroupant un ensemble d'outils propre à son profil.

Les principales fonctionnalités de Web CT sont regroupées dans le tableau suivant :



Espace administrateur	Espace enseignant	Espace apprenant
Gestion d'utilisateurs (profils, données, ...)  Suivi du bon fonctionnement technique de la plate-forme.	Création et suivi des cours  Création des groupes et des forums de discussion  consultations des notes et statistiques à l'aide d'un carnet de notes.	Découverte des contenus.  Lancement des activités.  Communication/collaboration et soutien à l'apprentissage(calendrier, courriel, forums, messagerie instantanée).

**Tableau 1 : Fonctionnalités de Web CT.**

### **II.8.2.2 Moodle (Modular Object-Oriented Dynamic Learning Environment) [9]:**

Moodle est une plate-forme d'apprentissage en ligne sous licence open source servant à créer des communautés d'apprenants autour de contenus et d'activités pédagogiques.

À un système de gestion de contenu (SGC), Moodle ajoute des fonctions pédagogiques ou communicatives pour créer un environnement d'apprentissage en ligne : c'est une application permettant de créer, par l'intermédiaire du réseau, des interactions entre des pédagogues, des apprenants et des ressources pédagogiques.

Ces systèmes de e-formation sont aussi des dispositifs de « formation ouverte et à distance » (FOAD) ou, de manière plus académique, « environnements d'apprentissage médiatisé ».

#### **Principales caractéristiques :**

- Elle présente de nombreuses caractéristiques : forums, gestionnaire de ressources, tests et neuf modules clé en main (Devoirs, chat, sondages, glossaires, journal, étiquettes, leçons, wiki, tests, base de données, blogues, flux RSS...). Elle intègre aussi un module de création de tests d'entraînement. Les questions créées avec ce module peuvent être mutualisées et réutilisées dans différents contextes d'épreuve.
- Moodle a été créée de manière modulaire : elle permet de répondre aux besoins d'un formateur isolé comme d'une institution académique. Aujourd'hui, le développement

de Moodle est fortement influencé par les demandes de la communauté d'administrateurs et d'utilisateurs (enseignants, pédagogues) de Moodle. On peut développer de nouveaux modules facilement puisque Moodle s'appuie sur le PHP, le langage de script le plus utilisé dans le monde de l'Internet.

- Elle présente une interface très conviviale avec les formateurs, apprenants et administrateurs. Sa prise en main s'effectue en moins d'une heure. Elle est très facile à installer. Son coût d'hébergement est très faible (100 à 200 euros). La plateforme Moodle a été créée dès le début dans une perspective mondiale et donc multilingue. Chaque installation comporte un pack de près de 90 langues. On peut changer à loisir la langue de la plate-forme. Plusieurs langues peuvent être utilisées simultanément dans les divers cours d'une même plate-forme.
- Des filtres permettent d'utiliser facilement des fichiers multimédias ou des expressions mathématiques au sein des pages Moodle.
- Des rapports d'usage détaillés pour chaque apprenant permettent de superviser les efforts d'apprentissage.
- La communauté Moodle (plusieurs milliers d'individus) est fortement structurée à travers de nombreux forums présents sur le site moodle.org : il s'agit de communautés de pratiques centrées sur des problématiques précises.
- Moodle facilite la gestion dynamique d'un cours avec le calendrier. Chaque cours représente comme un portail composé de blocs que le tuteur peut afficher à sa guise tout au long du déroulement du cours, évitant ainsi une surcharge informationnelle.

### **Communautés de pratique :**

Le site moodle.org héberge de nombreuses communautés de pratiques qui participent au développement de la plate-forme.

- « **Using Moodle** » : la communauté d'« utilisation de Moodle » se présente comme une vingtaine de forums qui traitent des problèmes d'utilisation généraux, d'installation, etc.
- « **Moodle Exchange** » propose un lieu virtuel où on peut échanger des cours Moodle dans une perspective communautaire libre. On peut installer un cours Moodle dans une plate-forme en quelques minutes via la procédure de restauration.

- « **Moodle Documentation Project** » a pour mission de produire des guides pour développeurs, créateurs de cours et apprenants dans toutes les langues.
- « **Moodle for Business Uses** » se concentre sur les applications d'affaire de Moodle.
- « **Moodle for Languages teaching** » regroupe les individus intéressés par l'enseignement des langues.

## II.9 Les enjeux du E-learning :

Les enjeux du E-Learning sont notamment de :

- Rendre plus efficaces, plus solides, plus adaptés les processus d'apprentissage et l'accès à la connaissance.
- Bénéficier des avantages des technologies éducatives (interactivité, simulation).
- Bénéficier des avantages de la formation à distance (plus grande autonomie, élimination de contraintes...) tout en éliminant les inconvénients de déshumanisation.

Le E-learning trouve de nombreuses applications à la condition de réunir les critères pédagogiques et organisationnels de l'efficacité. Pour les responsables de formation, au plan pratique, l'efficacité du e\_Learning réside dans :

- La personnalisation de la formation.
- Le gain de temps.
- La réduction de coûts.
- La démultiplication des thèmes de formation.
- L'efficacité de l'apprentissage.

Au plan organisationnel, on observe également :

- Une standardisation des supports d'auto-formation et une approche qualité.
- Une ouverture des sources d'approvisionnement des offres de formation : acheter devient un acte important.
- Des perspectives élargies de diffusion auprès des salariés, des fournisseurs, des clients. Tout investissement peut être immédiatement valorisé à l'échelle la plus large.

**II.10 Conclusion :**

Nous avons vu dans ce chapitre des généralités sur le E-Learning, ce qui nous a permis de déduire que l'E-Learning promet de révolutionner la formation en réduisant les contraintes de temps et d'espace, ce qui peut conduire à une diffusion rapide du savoir. Les outils et les plates-formes de E-Learning progressent rapidement et visent à donner aux enseignants et élèves les mêmes réflexes que ceux qui existent dans le monde physique.

Nous entamerons dans le chapitre suivant l'analyse et la conception de notre application d'enseignement à distance.

## **II.1 Introduction :**

Les bouleversements technologiques ont donné naissance à beaucoup de formes d'assistance à la formation, aujourd'hui, on parle d'E-Learning. Ce type de formations permet de développer les compétences des apprenants, tout en rendant le processus d'apprentissage indépendant du temps et du lieu. Afin de mieux comprendre ce concept

Nous allons introduire dans ce chapitre l'état de l'art du E-learning.

## **II.2 La formation à distance [5] :**

Les premiers cours « à distance » sont apparus, avec la révolution industrielle, à la moitié du 19ème siècle en Angleterre et aux Etats-Unis, Certains auteurs attribuent à Isaac Pittman la mise en application du premier cours par correspondance en Angleterre autour des années 1840. Ces modalités nouvelles de cours se distinguaient de l'enseignement traditionnel par une séparation physique entre l'enseignant et l'étudiant, le contact s'établissant par correspondance.

Cela fait plus d'un siècle que la formation à distance a fait son apparition sous la forme d'enseignement par correspondance. L'expression la plus utilisée est celle d'EAD (Enseignement À Distance) qui désigne une situation où la transmission des connaissances se fait en dehors de la relation face-à-face Enseignant-Apprenant.

Cette expression est couramment utilisée par les formateurs car elle caractérise un point de vue enseignant. Un terme plus général pour caractériser cette situation est FAD (Formation À Distance) qui permet d'inclure à la fois le terme d'Enseignement à Distance (point de vue enseignant) et Apprentissage à Distance (point de vue apprenant).

### **II.2.1 Qu'est-ce que la FAD :**

La formation à distance est un système de formation qui permet de se former sans se déplacer sur le lieu de formation et sans la présence physique d'un formateur. La transmission des connaissances et les activités d'apprentissage se situent en dehors de la relation directe en face à face, dite « en présentiel » entre l'enseignant et l'apprenant.

La formation à distance recouvre plusieurs modalités : cours par correspondance et e-learning.

### **II.2.2 Pourquoi la FAD :**

La formation à distance est un système de formation très souple, tant dans l'espace que dans le temps. Le stagiaire peut apprendre à son rythme, chez lui, dans un cyber café ou ailleurs, quand il a un moment de libre.

La FAD s'adapte bien à la diversité des besoins en formation. Notamment dans les domaines techniques et industriels où l'évolution rapide des technologies exige de plus en plus la création de circuits directs entre formateurs et apprenants.

C'est un enseignement à distance qui permet à chacun de se former en toute autonomie tout en bénéficiant du support et des conseils d'experts à tout moment.

### **II.3 Formation Ouverte et à Distance (FOAD) :**

Il existe aujourd'hui une définition officielle donnée à la FOAD dans la circulaire du 20 juillet 2001 de la Délégation générale à l'emploi et à la formation professionnelle (DGEFP) : il s'agit d'un dispositif souple de formation, organisé en fonction de besoins individuels ou collectifs, qui comprend des apprentissages individualisés, l'accès à des ressources et à des compétences locales ou à distance, et qui n'est pas exécuté nécessairement sous le contrôle permanent d'un formateur.

La FOAD s'appuie sur les NTIC (Nouvelles Technologies de l'Information et de la Communication), notamment l'Internet, l'Intranet et l'Extranet pour fournir une formation à distance favorisant l'individualisation des formations (les apprenants ont de plus en plus besoin d'un apprentissage centré sur eux-mêmes) et les communications entre formateurs. On parle aussi de E-Learning (Ou apprentissage en ligne ou électronique) ou de E-Formation.

### **II.4 Différences entre la formation ouverte et la formation à distance :**

La formation à distance n'a rien à voir avec la formation ouverte, puisqu'au départ le terme « Ouvert » signifiait « ouvert sans diplôme », c'est-à-dire accessible à tout le monde. Lorsque l'Open Université a été créée, on pouvait s'y inscrire sans prérequis, sans diplômes initiaux. La notion de « formation à distance » intègre à son tour deux notions distinctes : l'enseignement à distance et l'apprentissage à distance. L'apprentissage dépend d'abord et

avant tout de l'apprenant, alors que l'enseignement n'implique pas nécessairement que l'apprenant apprenne.

La formation ouverte, l'exemple type est le centre de ressources. Les termes « formation ouverte et à distance » représentent des choses extrêmement différentes et des dispositifs extrêmement variés, qui de plus ont évolué au cours du temps. Ces dispositifs n'ont qu'un point commun entre eux : ils brisent la règle des trois unités de la tragédie classique, qui est aussi celle de l'enseignement traditionnel.

S'il y a rupture de l'unité de lieu, nous parlerons de formation à distance, s'il y a rupture de l'unité de temps et d'action, même dans un même lieu, nous parlerons de formation ouverte.

S'il y a rupture des deux, on parlera de formation ouverte et à distance.

## **II.5 Présentation du E-learning :**

### **II.5.1 Définition du E-learning [6] :**

Le E-learning est un processus d'apprentissage à distance s'appuyant sur des ressources multimédias, qui permet à une ou plusieurs personnes de se former à partir de leur ordinateur en utilisant les nouvelles technologies de l'information et de la communication(NTIC). Les supports multimédias utilisés peuvent combiner du texte, des graphismes en 2 ou 3 dimensions, du son, de l'image, de l'animation et même de la vidéo.

Plusieurs termes sont utilisés pour traduire le terme E-Learning

#### **E-Formation :**

E-Formation est à l'origine un sous-ensemble de la FOAD, qui s'appuie sur les réseaux électroniques. Aujourd'hui le concept d'E-Learning est de plus en plus employé, attestant de l'évolution fondamentale de ce domaine de formation.

#### **Apprentissage en ligne :**

Mode d'apprentissage basé sur l'utilisation des nouvelles technologies, qui permet l'accès à des formations en ligne, interactives et parfois personnalisées, diffusées par l'intermédiaire d'Internet, d'un intranet ou autre média électronique, afin de développer les compétences, tout en rendant le processus d'apprentissage indépendant de l'heure et de l'endroit.

### II.5.2 Pourquoi le E-Learning :

Le choix du E-Learning s'explique par le fait de vouloir abolir les notions de temps et d'espace, les apprenants ne doivent pas se trouver tous en un même lieu, à des moments bien déterminés. De plus, la qualité de la formation et l'efficacité de son processus peuvent être considérablement accrues par l'interactivité en permettant l'adaptation du contenu de la formation à la situation pédagogique de l'apprenant.

Enfin, la formation à distance revêt plusieurs avantages économiques notamment :

- La possibilité d'amortir sur un plus grand nombre d'apprenants les coûts liés à la création de contenus.
- L'automatisation des fonctions de gestion de l'enseignement (administration des cours, contrôle des connaissances, etc.)
- Les économies réalisées sur les déplacements des apprenants et des enseignants
- Les économies réalisées par l'utilisation optimale des formateurs, qui n'interviennent qu'en cas de difficulté et dont le coût peut être réparti sur un plus grand nombre d'apprenants.

Ces avantages peuvent largement compenser les coûts d'acquisition, de maintenance et de développement des logiciels utilisés lors de la mise en œuvre de l'E-Formation.

### II.5.3 Applications du E-Learning :

Aujourd'hui, le E-Learning est lié spécialement à deux secteurs :

- Des grandes entreprises qui l'utilisent avec le KM (Knowledge Management ou management des connaissances) pour former et actualiser les connaissances de ses employés de manière plus rapide.
- Des universités qui offrent des alternatives de formation à distance.

L'univers du E-Learning s'étend à des différents prestataires :

- Les entreprises qui offrent des solutions technologiques : infrastructure, plates-formes.
- Les sociétés qui fournissent des contenus pédagogiques qu'elles ont dans l'étagère.
- Les sociétés qui développent un contenu pédagogique sur mesure.



## **II.6 Les différentes formes de la formation en ligne :**

La formation en ligne se présente sous une multitude de formes. Cette particularité facilite son intégration aux stratégies de formation continue au près des apprenants.

On distingue deux types de formation en ligne :

### **II.6.1 Formation exclusivement en ligne (sans tutorat) :**

Ce type de formation d'avantage individualisé, est proche du service fourni par un CD ou d'une vidéo de formation. C'est un modèle d'apprentissage sans formateur, par lequel l'apprenant communique avec différents serveurs pour chercher de l'information et accéder à des modules d'autoformation sur des sites Web, des CD ou DVD.

#### **Avantage :**

Le grand avantage est que l'apprentissage se fait au rythme de l'apprenant, qui peut avoir différentes façons de parcourir son cours.

L'apprenant peut s'auto évaluer, et faire des pauses lorsqu'il en ressent besoin.

Formation moins coûteuse, elle tend à devenir gratuite.

#### **Inconvénients :**

Une formation qui n'apporte pas beaucoup de valeur ajoutée. En effet elle comporte des inconvénients sur le plan de la qualité pédagogique.

### **II.6.2 Formation exclusivement en ligne (avec tutorat) :**

C'est le modèle de référence des « portails de formation », plutôt destiné au grand public.

L'acheteur paie en ligne sa formation. Un tuteur lui est attribué qui lui propose un programme de travail. Dans ce cas, la formation peut être :

**II.6.2.1 Asynchrone :**

La formation asynchrone est une méthode d'apprentissage s'adaptant aux disponibilités de l'apprenant, celui-ci a accès à un ou plusieurs instruments (exemples : vidéo, texte, logiciel d'apprentissage virtuel) qu'il utilisera à sa guise et il n'est pas obligé d'être connecté en même temps que les autres apprenants et le formateur. Cependant, Le rythme de l'apprentissage est géré par le formateur. L'échange des documents et la participation au discours se feront par voie indirecte (courriel, forum de discussion).

**Les principaux avantages de la formation asynchrone :**

- Les apprenants évoluent à leur propre rythme.
- Ils peuvent adapter l'ordre dans lequel ils appréhendent les éléments du cours.
- Ils peuvent revoir et approfondir certains aspects du cours à leur guise.

**II.6.2.2 Synchrone :**

Dans la formation synchrone les apprenants et le formateur doivent être connectés en même temps. C'est aussi un modèle collectif. Cette pratique demande un équipement de vidéoconférence en salle ou sur poste de travail pour que la diffusion soit en direct.

**Les principaux avantages de la formation synchrone :**

La dynamique de groupe s'installe plus rapidement ;

On peut rapidement créer du contenu prêt à diffuser ;

Utilisation de langage oral et visuel.

**II.6.3 Formation mixte (avec tutorat en ligne) :**

C'est plutôt le modèle des écoles et organismes de formation. En mettant en ligne le contenu de la formation, des tests, des évaluations et un tutorat, cela permet de réduire le temps du présentiel et d'individualiser la formation.

Ce modèle de formation combine donc les éléments de l'apprentissage en ligne et de l'apprentissage traditionnel en classe. Afin de faciliter l'intégration des connaissances acquises à travers une formation.

**Les principaux avantages de la formation mixte :**

L'acquisition des connaissances est meilleure avec la formation mixte ;

Les classes virtuelles en temps réel permettent aux apprenants d'assister aux cours n'importe où.

**II.7 Outils de communication**

Généralement les plates-formes SGC (Système de Gestion de Cours) intègrent un certain nombre d'outils de communication, la liste des outils disponibles dépend de la richesse du SGC. Ces outils sont souvent très utilisés sur le Web et ne sont en rien spécifiques au E-Learning, ils font simplement partie intégrante du SGC et ne nécessitent pas de configuration spécifique sur les postes des apprenants (par exemple, il n'est pas nécessaire de configurer un client messagerie pour profiter de ce service).

On peut citer :

- **La messagerie** (C'est un espace de communication entre les membres du groupe (stagiaires et tuteurs). Ce système permet l'envoi et la réception de mails avec ou sans fichier attaché. Il peut être interne à la plateforme et ne nécessite pas d'avoir un e-mail personnel)
- **Le forum** (Le forum, pouvant être public ou réservé à un groupe, permet aux stagiaires et aux tuteurs : de poster des messages qui seront accessibles à l'ensemble des membres du groupe de formation, de répondre aux messages déjà postés et ainsi engager une discussion sur un sujet donné).
- **Le Chat** (Il permet à l'ensemble des membres du groupe de discuter en temps réel)
- **Les documents pédagogiques partagés** : Une zone commune peut être utilisée pour mettre à disposition du groupe, différents documents, Documents sous forme numérique (Word, Excel, PDF, etc.) pour l'ensemble du groupe.
- **Les news** (La possibilité de créer des news ce qui permettra, par exemple, d'informer ses apprenants de nouveaux rendez-vous, ou de leur communiquer diverses informations)

- **La visioconférence** : Equivalente à l'intervention du formateur en salle face à un groupe, mais ici chacun est devant son ordinateur, à domicile, au travail, ou un centre de ressources...
- **Les sondages** : Le sondage permettra de poser des questions ouvertes ou de proposer des choix de réponses. Ultérieurement, les réponses obtenues sur un cours en particulier pourront être analysées.
- **Le Bloc-notes** : C'est un espace privé à chaque utilisateur qui peut y noter toutes les informations qu'il souhaite : ce qu'il a retenu de sa formation, des actions à mener...etc. Ces notes sont archivées par date et il est possible de les compiler dans une même note.
- **Les Glossaires** (Des glossaires peuvent être créés, destinés à l'ensemble des utilisateurs ou un groupe en particulier. Les différents termes avec leur signification sont enregistrés, un moteur de recherche permet aux utilisateurs de faciliter la recherche d'un terme dans les glossaires).
- **Les Faq** : Cet espace permet aux utilisateurs de trouver éventuellement la réponse à des questions récurrentes.

## **II.8 Les Plate-formes de formation à distance ou LMS (Learning Management System) [7] :**

Une plate-forme de formation à distance est un logiciel d'apprentissage, combiné avec les technologies du Web, qui fournit des fonctionnalités à différents utilisateurs (Apprenants, Formateurs, Auteurs) afin de conduire et d'automatiser le processus de formation à distance d'apprenants. En d'autres termes, c'est un logiciel permettant la consultation à distance d'objets pédagogiques, le télé tutorat et la centralisation de l'apprentissage sur l'individu. Ces plateformes permettent le développement et la distribution de ressources et d'activités pédagogiques sur un environnement d'apprentissage en ligne. La maintenance du système et la gestion des droits d'accès des apprenants aux cours sont assurés par un Administrateur.

### II.8.1 Les acteurs d'un LMS et leurs rôles :

Le nombre d'acteurs et leurs fonctions diffèrent d'une plate-forme à une autre, selon les objectifs et les fonctionnalités offertes par chacune d'entre elles. Certaines se sont contentées de trois acteurs, à savoir l'apprenant, le formateur/créateur de cours et l'Administrateur.

Cependant, six rôles principaux ont été distingués en observant un ensemble de plateformes de formation à distance. Ces rôles sont :

**Concepteur (auteur ou créateur de cours) :** Il établit (pour l'apprenant) un parcours de formation personnalisé et individualisé. Il construit, adapte et maintient un système d'apprentissage. Il peut aussi avoir le rôle de présentateur. Ce rôle consiste à rendre disponible les informations pour l'apprentissage, en d'autres termes, la présentation des informations, la gestion des médias, la clarification des contenus en réponse à des questions, l'analyse et l'évaluation des contenus des documents...etc.

**Formateur (ou tuteur) :** Il accompagne les apprenants durant leurs parcours pédagogiques. Il a la tâche de motiver, d'orienter et d'évaluer les apprenants, comme il les assiste dans l'utilisation des ressources et du matériel mis à leur disposition.

**Apprenant :** Son rôle consiste à transformer les informations en connaissances. Pour cela, il doit gérer ses activités et son temps, exploiter les ressources qui lui sont indiquées de façon à optimiser la quantité d'information qu'il peut en tirer, résoudre des problèmes, échanger des informations et des idées avec d'autres apprenants via les outils de communication, s'auto évaluer et présenter des travaux qui serviront de référence à son formateur pour son évaluation.

**Administrateur :** Un administrateur installe et assure la maintenance du système, il s'occupe des tâches d'administration comme les inscriptions des acteurs dans la plateforme, il gère les droits d'accès, assure la gestion des ressources pédagogiques, ce qui consiste en la classification, l'indexation et l'administration des matériaux pédagogiques.

**Orienteur :** Son rôle est de gérer les acteurs, les événements, la création des cursus de formation, des individus et des groupes. Il s'occupe des inscriptions des apprenants dans une formation donnée et de la constitution de groupes de travail.

**Evaluateur :** Il s'occupe de l'évaluation des groupes et des individus et ceci grâce aux informations fournies par la plate-forme.

**II.8.2 Exemple de plateforme d'apprentissage en ligne :**

Nous donnons quelques exemples de Système de gestion des cours (S.G.C) :

**II.8.2.1 Web CT [8] :**

Web CT (Web course Tools), est une plateforme développée par l'université canadienne British Columbia. Il s'agit d'un environnement d'enseignement/apprentissage destinée à développer et diffuser des télé Cours et à faciliter l'intégration d'un ensemble d'outils de communication, d'évaluation, de gestion, etc.

Web CT est l'une des plates formes commerciales de téléformation les plus vendues dans le monde. Elle est traduite en plusieurs langages utilisés par des milliers d'instruction. Dans WebCT, chaque acteur humain (administration, concepteur de cours, tuteur et apprenant) dispose d'un espace de travail sous forme d'une interface (accessible via un mot de passe) regroupant un ensemble d'outils propre à son profil.

Les principales fonctionnalités de Web CT sont regroupées dans le tableau suivant :

Espace administrateur	Espace enseignant	Espace apprenant
Gestion d'utilisateurs (profils, données, ...)  Suivi du bon fonctionnement technique de la plate-forme.	Création et suivi des cours  Création des groupes et des forums de discussion  consultations des notes et statistiques à l'aide d'un carnet de notes.	Découverte des contenus.  Lancement des activités.  Communication/collaboration et soutien à l'apprentissage(calendrier, courriel, forums, messagerie instantanée).

**Tableau 1 : Fonctionnalités de Web CT.****II.8.2.2 Moodle (Modular Object-Oriented Dynamic Learning Environment) [9]:**

Moodle est une plate-forme d'apprentissage en ligne sous licence open source servant à créer des communautés d'apprenants autour de contenus et d'activités pédagogiques.

À un système de gestion de contenu (SGC), Moodle ajoute des fonctions pédagogiques ou communicatives pour créer un environnement d'apprentissage en ligne : c'est une application permettant de créer, par l'intermédiaire du réseau, des interactions entre des pédagogues, des apprenants et des ressources pédagogiques.

Ces systèmes de e-formation sont aussi des dispositifs de « formation ouverte et à distance » (FOAD) ou, de manière plus académique, « environnements d'apprentissage médiatisé ».

**Principales caractéristiques :**

- Elle présente de nombreuses caractéristiques : forums, gestionnaire de ressources, tests et neuf modules clé en main (Devoirs, chat, sondages, glossaires, journal, étiquettes, leçons, wiki, tests, base de données, blogues, flux RSS...). Elle intègre aussi un module de création de tests d'entraînement. Les questions créées avec ce module peuvent être mutualisées et réutilisées dans différents contextes d'épreuve.
- Moodle a été créée de manière modulaire : elle permet de répondre aux besoins d'un formateur isolé comme d'une institution académique. Aujourd'hui, le développement

de Moodle est fortement influencé par les demandes de la communauté d'administrateurs et d'utilisateurs (enseignants, pédagogues) de Moodle. On peut développer de nouveaux modules facilement puisque Moodle s'appuie sur le PHP, le langage de script le plus utilisé dans le monde de l'Internet.

- Elle présente une interface très conviviale avec les formateurs, apprenants et administrateurs. Sa prise en main s'effectue en moins d'une heure. Elle est très facile à installer. Son coût d'hébergement est très faible (100 à 200 euros). La plateforme Moodle a été créée dès le début dans une perspective mondiale et donc multilingue. Chaque installation comporte un pack de près de 90 langues. On peut changer à loisir la langue de la plate-forme. Plusieurs langues peuvent être utilisées simultanément dans les divers cours d'une même plate-forme.
- Des filtres permettent d'utiliser facilement des fichiers multimédias ou des expressions mathématiques au sein des pages Moodle.
- Des rapports d'usage détaillés pour chaque apprenant permettent de superviser les efforts d'apprentissage.
- La communauté Moodle (plusieurs milliers d'individus) est fortement structurée à travers de nombreux forums présents sur le site moodle.org : il s'agit de communautés de pratiques centrées sur des problématiques précises.
- Moodle facilite la gestion dynamique d'un cours avec le calendrier. Chaque cours représente comme un portail composé de blocs que le tuteur peut afficher à sa guise tout au long du déroulement du cours, évitant ainsi une surcharge informationnelle.

### **Communautés de pratique :**

Le site moodle.org héberge de nombreuses communautés de pratiques qui participent au développement de la plate-forme.

- « **Using Moodle** » : la communauté d'« utilisation de Moodle » se présente comme une vingtaine de forums qui traitent des problèmes d'utilisation généraux, d'installation, etc.
- « **Moodle Exchange** » propose un lieu virtuel où on peut échanger des cours Moodle dans une perspective communautaire libre. On peut installer un cours Moodle dans une plate-forme en quelques minutes via la procédure de restauration.



- « **Moodle Documentation Project** » a pour mission de produire des guides pour développeurs, créateurs de cours et apprenants dans toutes les langues.
- « **Moodle for Business Uses** » se concentre sur les applications d'affaire de Moodle.
- « **Moodle for Languages teaching** » regroupe les individus intéressés par l'enseignement des langues.

## II.9 Les enjeux du E-learning :

Les enjeux du E-Learning sont notamment de :

- Rendre plus efficaces, plus solides, plus adaptés les processus d'apprentissage et l'accès à la connaissance.
- Bénéficier des avantages des technologies éducatives (interactivité, simulation).
- Bénéficier des avantages de la formation à distance (plus grande autonomie, élimination de contraintes...) tout en éliminant les inconvénients de déshumanisation.

Le E-learning trouve de nombreuses applications à la condition de réunir les critères pédagogiques et organisationnels de l'efficacité. Pour les responsables de formation, au plan pratique, l'efficacité du e\_Learning réside dans :

- La personnalisation de la formation.
- Le gain de temps.
- La réduction de coûts.
- La démultiplication des thèmes de formation.
- L'efficacité de l'apprentissage.

Au plan organisationnel, on observe également :

- Une standardisation des supports d'auto-formation et une approche qualité.
- Une ouverture des sources d'approvisionnement des offres de formation : acheter devient un acte important.
- Des perspectives élargies de diffusion auprès des salariés, des fournisseurs, des clients. Tout investissement peut être immédiatement valorisé à l'échelle la plus large.

**II.10 Conclusion :**

Nous avons vu dans ce chapitre des généralités sur le E-Learning, ce qui nous a permis de déduire que l'E-Learning promet de révolutionner la formation en réduisant les contraintes de temps et d'espace, ce qui peut conduire à une diffusion rapide du savoir. Les outils et les plates-formes de E-Learning progressent rapidement et visent à donner aux enseignants et élèves les mêmes réflexes que ceux qui existent dans le monde physique.

Nous entamerons dans le chapitre suivant l'analyse et la conception de notre application d'enseignement à distance.

### **III.1 Introduction :**

Après avoir vu dans le chapitre précédents l'état de l'art de E-Learning nous allons à présent entamer le développement de notre application, nous avons opté pour une démarche de conception orientée objet (RUP, Rational Unified Process), en se Basant sur la modélisation UML, tout en respectant les étapes suivantes :

- Initialisation : vision approximative de la finalité du projet, estimation globales.
- Elaboration : vision plus élaborée, implémentation itérative de l'architecture noyau, Résolution des risques élevés, identification de la plupart des besoins et du périmètre réel, Estimation plus réalistes.
- Construction : implémentation itérative des éléments qui présentent des risques et préparation du déploiement.
- Transition : déploiement.

### **III.2 Phase d'Initialisation :**

Une analyse approximative de notre projet permet de desseller les points suivants :

Notre application est un système d'enseignement à distance très répondeur de nos jours.

Notre application est de type Client/serveur.

Notre application ne présente pas de risque de sécurité d'où on estime que le système est abordable et réalisable.

### **III.3 Phase d'Elaboration :**

Notre projet porte sur la mise en place d'une application client/serveur 3-tiers, pour un système d'enseignement à distances. Dans le but de maîtriser la gestion de la pédagogie (cours, qcm, test) et la gestion des préinscrit l'application doit offrir un Environnement interactif et convivial aux différents acteurs de ce service.

#### **III.3.1 Identification des acteurs**

Après l'analyse approximative de notre projet on distingue les acteurs suivant or pour chaque acteur on a identifié les fonctionnalités lui permettant d'accomplir son rôle dans le système :

➤ **Administrateur :**

L'administrateur représente l'acteur chargé de réguler l'activité sur le système, il assure un ensemble d'actions tels que :

- Gestion des comptes utilisateurs (apprenants et formateurs) ;
- Gestion des sessions de pré-inscriptions des acteurs ;
- .

➤ **Formateur :**

Le formateur est la personne chargée de suivre les apprenants durant le processus d'apprentissage et cela en :

- Assurant le suivi des cours et exercices ;
- Communiquant et animant ses apprenants via la messagerie interne, la conversation.

➤ **Apprenant :**

L'apprenant est toute personne qui suit un cours dans son propre espace réservé, il peut aussi :

- Consulter et télécharger des cours, des exercices ;
- Communiquer avec les autres apprenants et son formateur via les outils collaboratifs (Messagerie interne) ;
- Faire des exercices afin d'évaluer son niveau de compréhension.

➤ **Visiteur :**

Le visiteur est toute personne qui visite le site sans être ni un apprenant ni un formateur.

L'ensemble d'actions qu'un visiteur peut effectuer sont :

- Se préinscrire en tant qu'apprenant ;
- Se préinscrire en tant que formateur ;
- Contacter l'administrateur pour d'éventuelles informations.

### III.3.2 Diagramme de contexte :

Le diagramme de contexte est un modèle conceptuel de flux qui offre une vision Globale des interactions entre le système et délimiter le champ de l'étude. Le diagramme de contexte pour notre système est donné par la figure suivante :

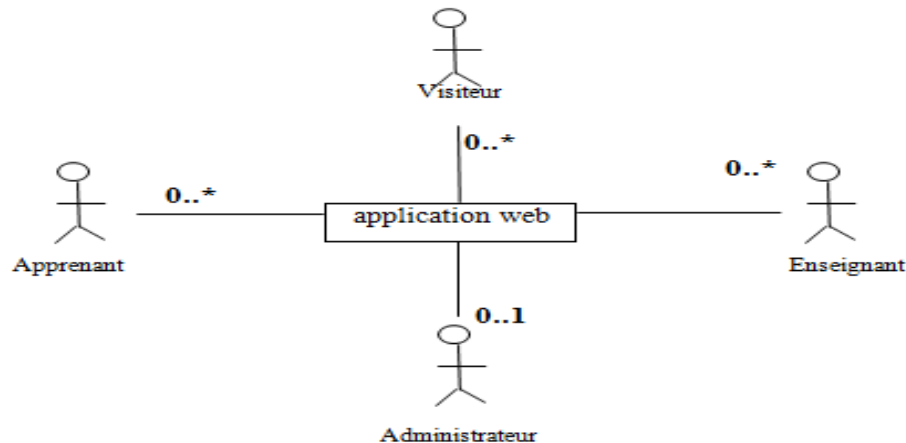
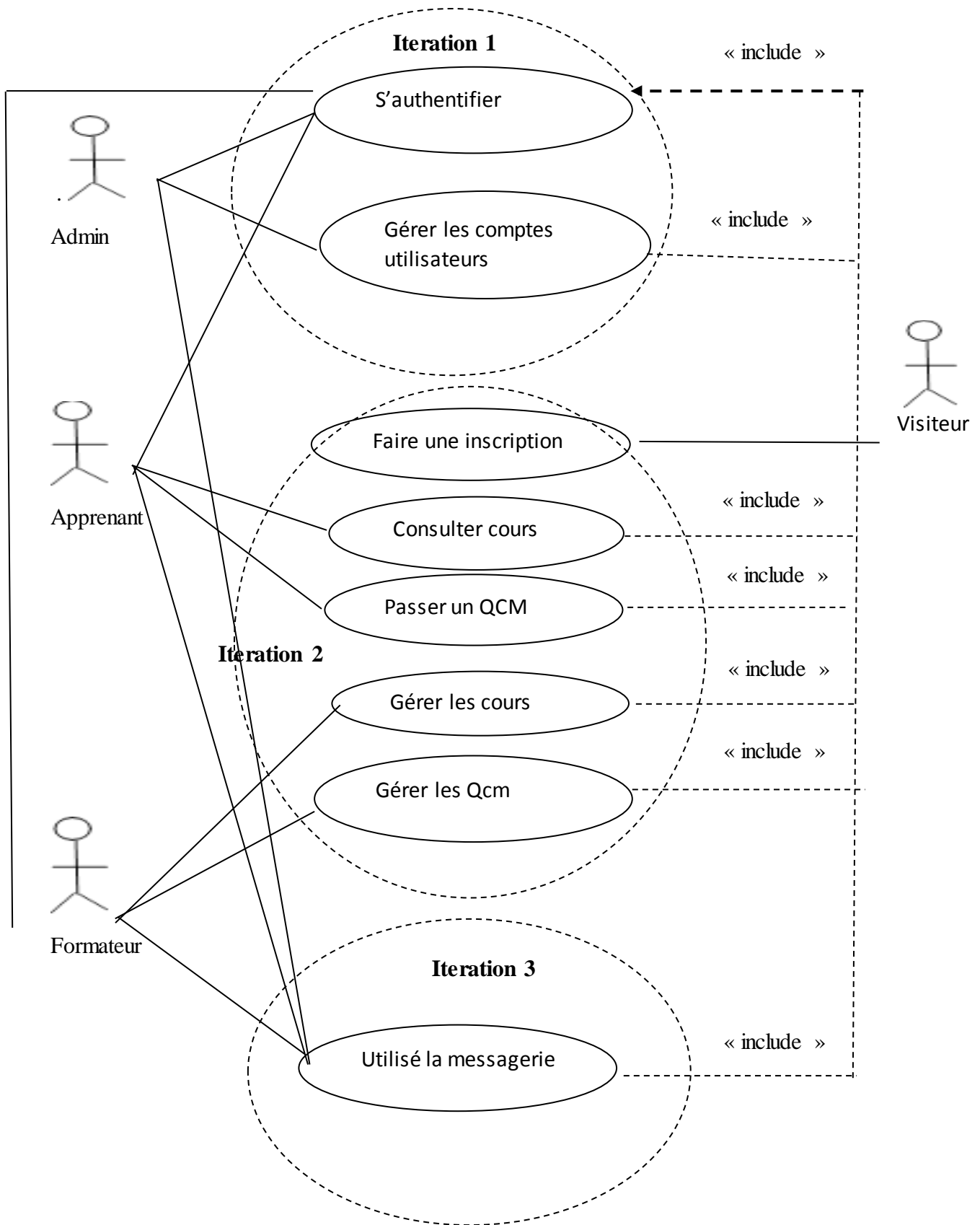


Figure III.1 : Diagramme de contexte

### III.3.3 Identification des cas d'utilisations :

Un cas d'utilisation (use case) modélise une interaction entre le système Informatique à développer et un utilisateur ou acteur interagissant avec le système. Plus précisément, un cas d'utilisation décrit une séquence d'actions réalisées par le système qui produit un résultat observable pour un acteur.

Représentation des cas d'utilisation globale :

**Figure III.2 :** Diagramme de cas d'utilisation globale

**Cas d'utilisation relatif à l'administrateur :**

- S'authentifier
- Gérer les utilisateurs
  - Ajouter un utilisateur
  - Supprimer un utilisateur
  - Modifier l'information de l'utilisateur
  - Consulter la liste des utilisateurs
- Utiliser la messagerie
  - Envoyer un message
  - Consulter les messages envoyés
  - Supprimer les messages

**Cas d'utilisation relatif à l'apprenant**

- S'authentifier
- Consulter cours
- Consulter un qcm
- Utiliser la messagerie
  - Envoyer un message
  - Consulter les messages envoyés
  - Supprimer les messages

**Cas d'utilisation relatif au formateur**

- S'authentifier
- Gérer les cours
  - Ajouter un cours
  - Modifier un cours
  - Supprimer un cours
- Gérer les qcm
  - Ajouter un qcm
  - Modifier un qcm
  - Supprimer un qcm
- Utiliser la messagerie
  - Envoyer un message
  - Consulter les messages envoyés
  - Supprimer les messages

**Cas d'utilisation relatif au visiteur**

- Inscription comme apprenant

**III.3.4 Définition des itérations :**

Le critère de définition des itérations se base sur les deux points suivants :

1. Ressemblance entre cas d'utilisation.
2. Dépendance entre cas d'utilisation.

**III.4 Construction :**

Pour chaque itération réalisée précédemment, nous allons suivre un cycle de vie en Cascade pour aboutir à un diagramme de classe en passant par un diagramme de séquence.



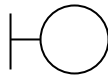
**III.4.1 Réalisation de l'itération 1 :*****III.4.1.1 Diagrammes de séquence pour itération 1 :***

Ce sont des diagrammes d'interactions qui mettent l'accent sur le classement chronologique des messages. Sur un diagramme de séquence, il est possible de représenter de manière explicite les différentes périodes d'activité d'un objet au moyen d'une bande rectangulaire.

De nombreux concepts sont mis en œuvre dans ce diagramme :

**Les objets de type interface** : l'interface entre l'acteur et le système, par exemples Application web, des pages web complètes.

**L'icône utilisée :**



**Les objets de type entité** : sont des objets décrits dans un cas d'utilisation, qui peuvent apparaître dans de nombreux cas d'utilisation, et généralement sa durée de vie dépasse celle de toute Interaction où il participe.

**L'icône utilisée :**

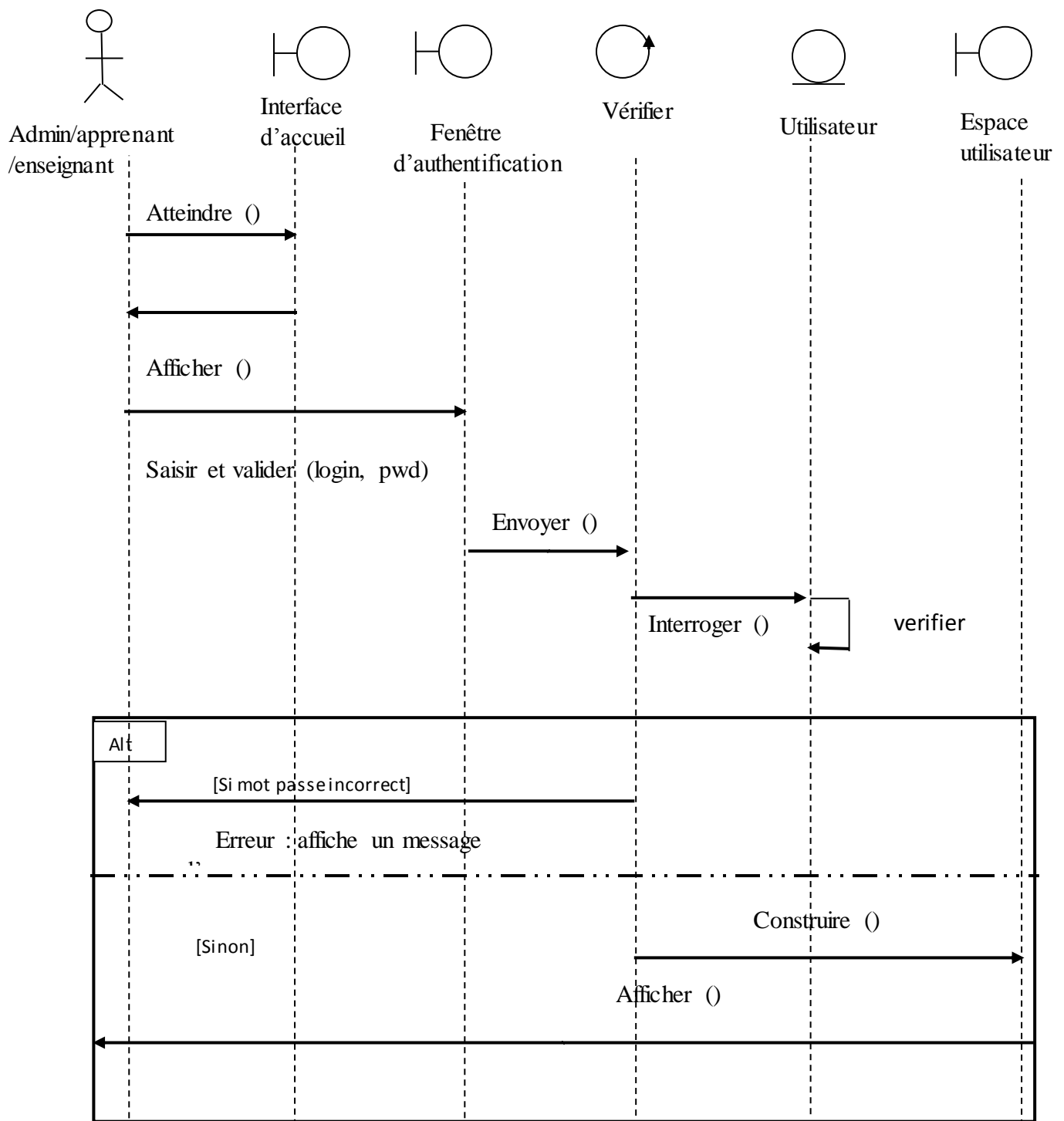


**Les objets de type contrôlent** : représentent les processus, c'est-à-dire les activités Système, ces objets dirigent les activités des objets entité et d'interface.

**L'icône utilisée :**

**L'icône utilisée :**



**Diagramme de séquence du cas d'utilisation : « Authentification » :****Figure III.3 :** Diagramme de séquence du cas d'utilisation : « Authentification » :

**Description textuelle des cas d'utilisation :**

**Définition d'un scénario :** un scénario décrit comment le système doit être utilisé, c'est un chemin logique traversant les cas d'utilisation (c'est l'exécution pas à pas d'un cas d'utilisation).

La Description textuelle des cas d'utilisation nous permet d'avoir une idée sur le fonctionnement de chaque cas d'utilisation, on la présente comme suit :

- Sommaire d'identification contient :
  - Le titre du cas d'utilisation.
  - Le but à atteindre du cas d'utilisation.
  - Les acteurs qui interviennent au cas d'utilisation.
- Les prés-conditions : elles décrivent dans quel état doit être le système avant que ce cas d'utilisation ne soit déclenché.
- Scénario nominal : qui décrit les différents enchainements du cas d'utilisation ainsi que les exceptions pouvant se produire.
- Les postes conditions : Elles décrivent l'état du système à l'issue des différents scénarios.

**Description de cas d'utilisation « Authentification » :****Sommaire d'identification****Titre** : Authentification.**Buts** : Connexion d'un utilisateur au Système.**Acteur** : administrateur, étudiant, formateur.**Pré conditions** :

L'utilisateur saisit ses droits d'accès (nom d'utilisateur et mot de passe).

**Scénario nominal** :**Enchaînement nominal** :

1. L'utilisateur atteint sa page d'authentification ;
2. Le système répond en affichant le formulaire d'identification en lui proposant de créer un compte, de s'identifier pour accéder à son propre espace

**Enchaînement alternatif (A)** : L'utilisateur possède un compte :

L'enchaînement démarre après le point 2 de L'enchaînement nominal.

3. L'utilisateur remplit le formulaire d'identification en tapant son login et mot de passe et valide en cliquant sur le bouton « Login » ;
4. Le système compare les données saisies par l'utilisateur avec celles de la base de données.

**Enchaînement alternatif (A1)** : Login et mot de passe corrects :

L'enchaînement démarre après le point 4 du l'enchaînement nominal

5. Le système charge l'espace approprié à l'utilisateur.

**Enchaînement alternatif (A2)** : Login ou mot de passe incorrect :

L'enchaînement démarre après le point 4 du l'enchaînement nominal

5. Le système redirige vers la page d'authentification.

**Enchaînement alternatif (B)** : crée un nouveau compte :

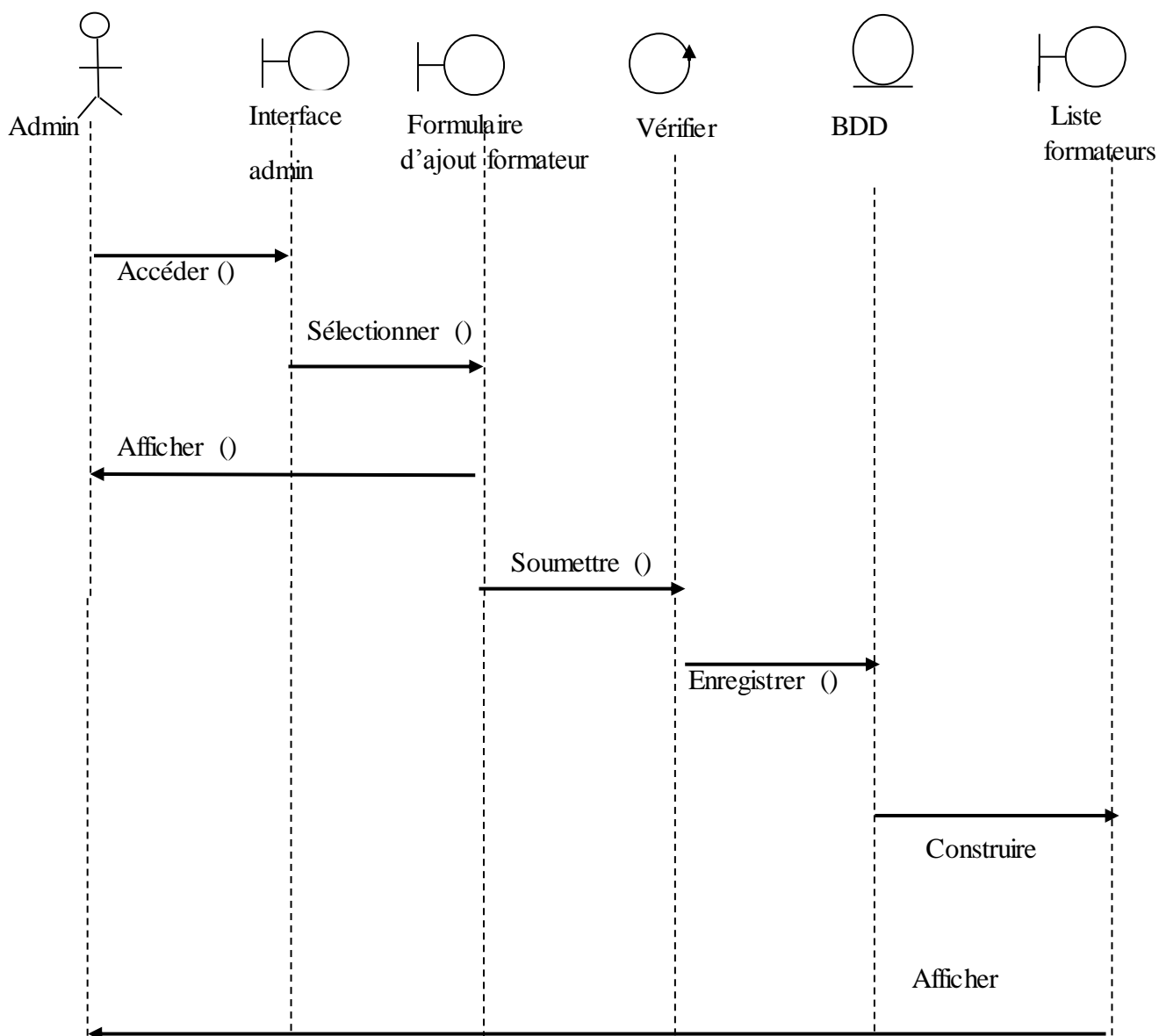
L'enchaînement alternatif démarre à l'étape 2

3. le système redirige vers la page d'accueil en incluant le formulaire de préinscription.

**Post conditions** :

L'utilisateur se connecte au système et peut ainsi accéder aux rubriques correspondant à son profile

**Diagramme de séquence du cas d'utilisation « ajouter un formateur » :**



**Figure III.4 :** Diagramme de séquence de cas d'utilisation « ajouter formateur »

**Description de cas d'utilisation « Ajouter un formateur » :****Sommaire d'identification :**

<b>Titre</b> : Ajouter un formateur <b>Buts</b> : inscrire un formateur. <b>Acteur</b> : administrateur
---

**Pré conditions :**

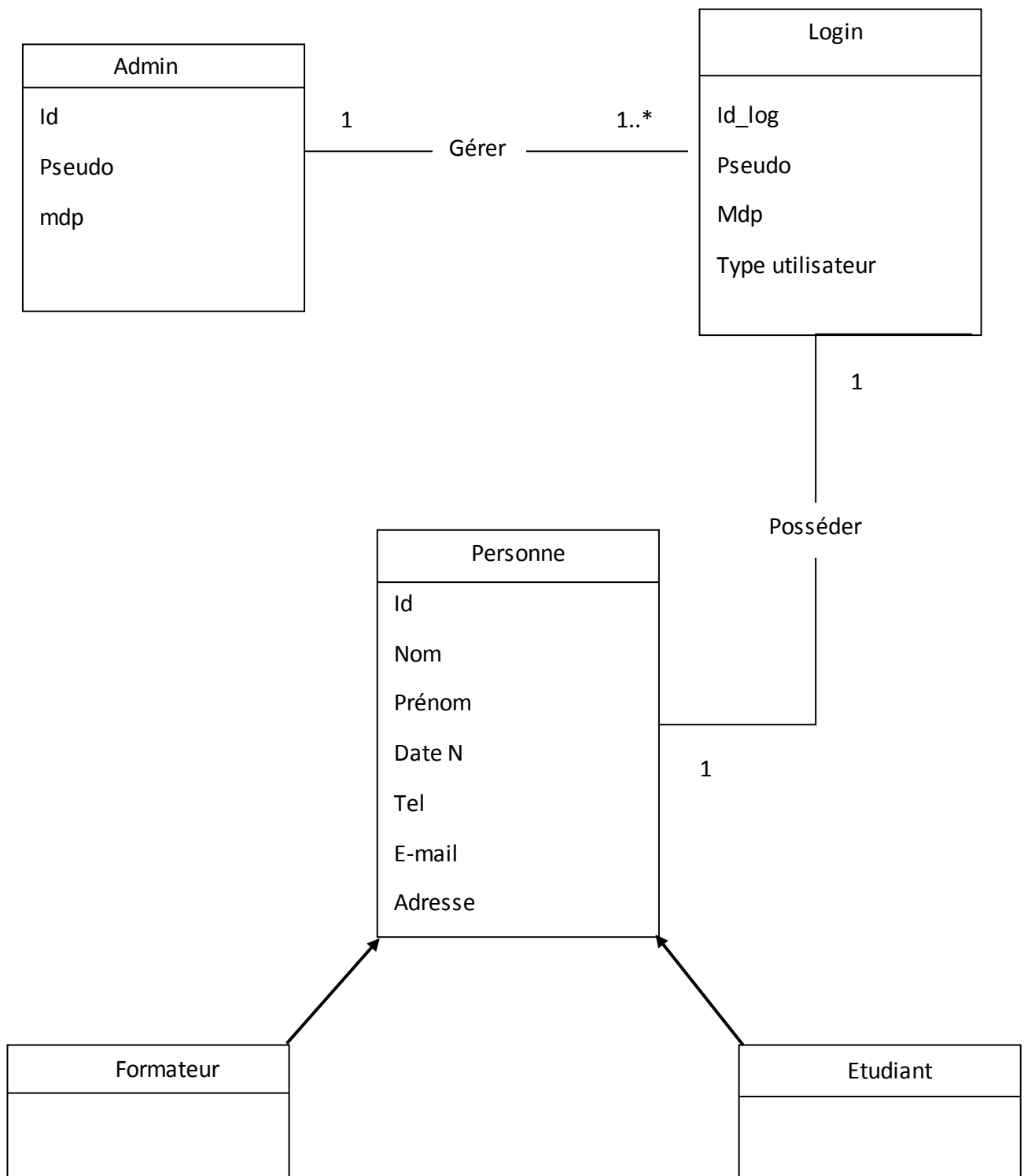
L'administrateur saisit ses droits d'accès (nom d'utilisateur et mot de passe).

**Scénario nominal :**

<b>Enchaînement nominal</b> : après authentification 1. L'administrateur accède à son espace 2. L'administrateur clique sur le lien ajouter formateur 3. Le formulaire d'ajout s'affiche 4. L'administrateur saisie les champs et valide 5. Le système affiche le formateur ajouté avec la liste des formateurs après vérification des données saisie par l'administrateur.
--

**Post conditions :**

Le formateur est ajouté à la base de données.

*III.4.1.2 Diagrammes de classe pour l'itération 1 :***Figure III.5 :** Diagramme de classe pour l'itération 1

## III.4.2 Réalisation de l'itération 2 :

## III.4.2.1 Diagrammes de séquence pour itération 2 :

Diagramme de séquence du cas d'utilisation « inscription » :

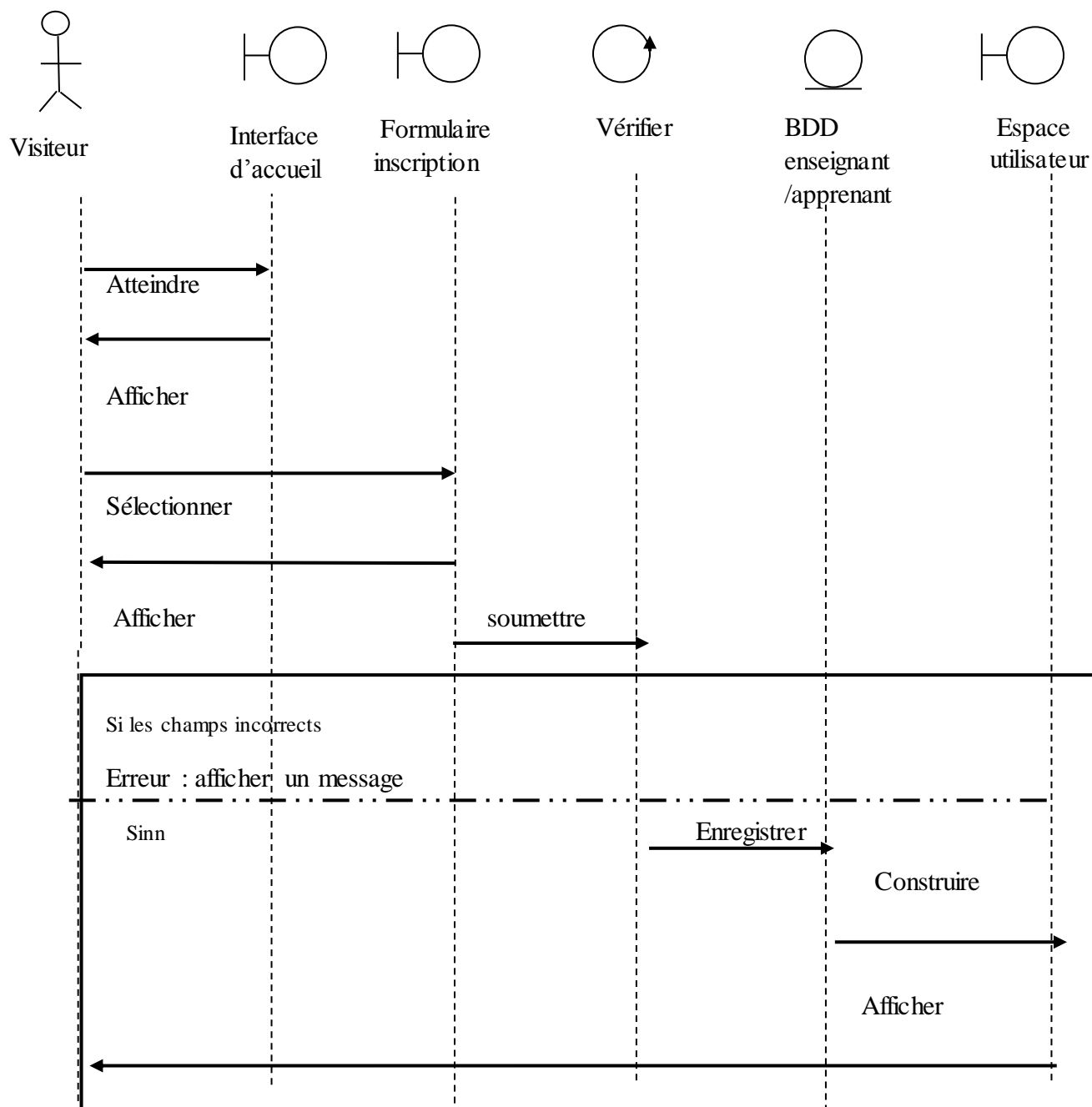


Figure III.6 : Diagramme de séquence de cas d'utilisation « inscription »



**Description textuelle de cas d'utilisation « Inscription » :****Sommaire d'identification :****Titre** : inscription.**Buts** : inscription comme étudiant.**Acteur** : visiteur**Pré conditions :**

Le visiteur fait une tentative d'inscription

**Scénario nominal :****Enchaînement nominal :**

1. Le visiteur atteint le site ;
2. Le système affiche la page d'accueil.

**Enchaînement alternatif (A) :**

L'enchaînement démarre après le point 2 de L'enchaînement nominal.

3. Le visiteur clique sur le lien apprenant.
4. Le système affiche la page d'authentification.
5. Le visiteur choisit « inscription » et le système affiche le formulaire d'inscription.
6. Le visiteur remplit le formulaire de préinscription et clique sur le bouton « inscription » pour le valider
7. Le système Vérifie le remplissage des champs.

**Enchaînement alternatif A1 : erreur dans le formulaire**

L'enchaînement démarre après le point 7 de l'enchaînement alternatif A

8. Le système signale les erreurs et propose de les corriger.

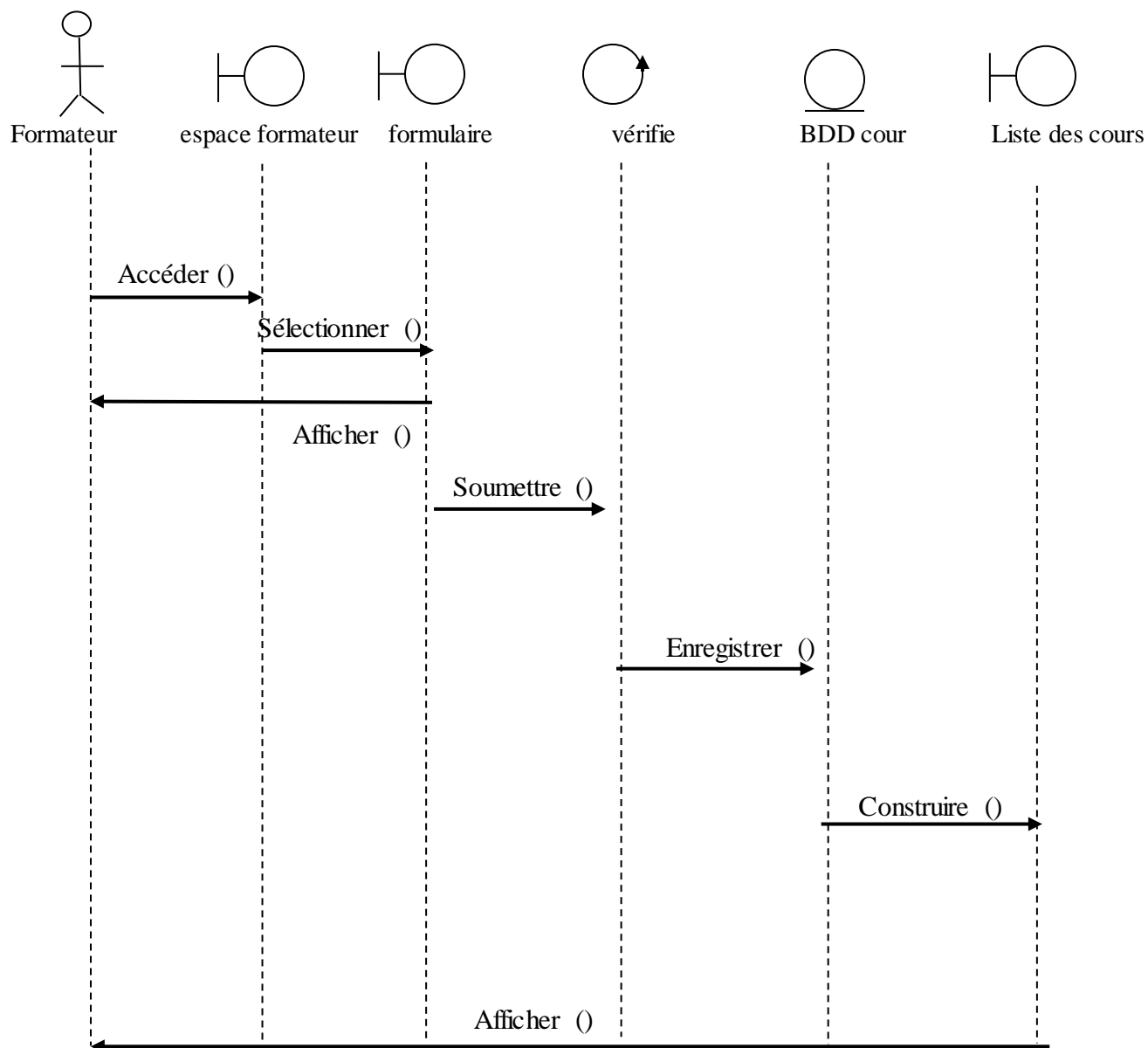
**Enchaînement alternatif A2 : pas d'erreur dans le formulaire**

L'enchaînement démarre après le point 7 de l'enchaînement alternatif A

9. Le système enregistre l'utilisateur autant qu'apprenant ou formateur préinscrit,

**Post conditions :**

Le visiteur est inscrit et peut accéder à son espace

**Diagramme de séquence du cas d'utilisation « Ajouter un cours »****Figure III.7 :** Diagramme de séquence de cas d'utilisation « ajouter un cours »

**Description de cas d'utilisation « Ajouter un cour » :****Sommaire d'identification :****Titre** : Ajouter cours**Buts** : augmenter la liste des cours.**Acteur** : formateur**Pré conditions :**

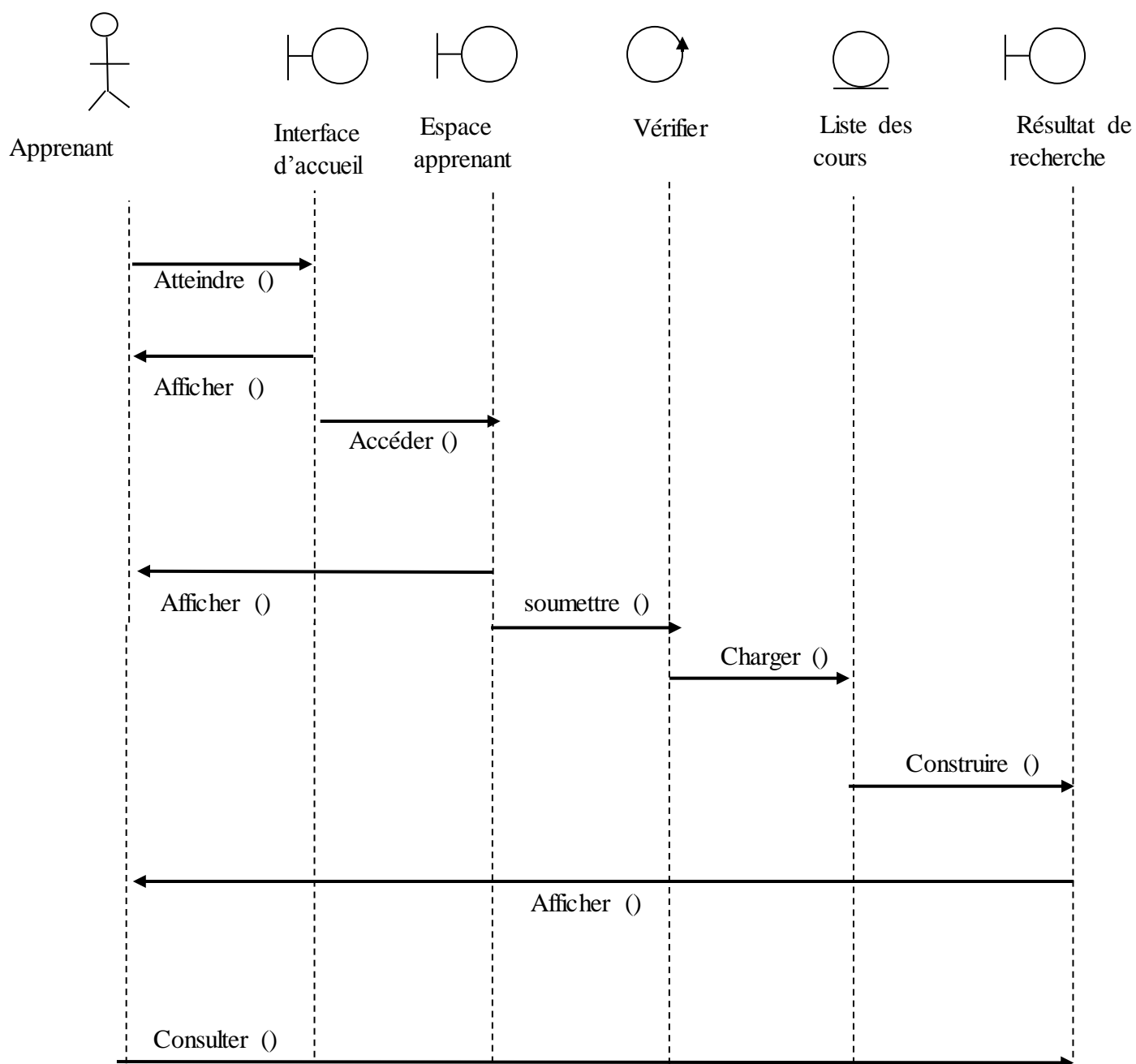
Formateur saisit ses droits d'accès (nom d'utilisateur et mot de passe).

**Scénario nominal :****Enchaînement nominal :**

6. Le formateur atteint le site ;
7. Le système affiche la page d'accueil.
8. Le formateur accède à son espace
9. Le formateur clique sur le lien ajouter cours
10. Le formulaire d'ajout s'affiche
11. Le formateur saisie les champs et valide
12. Le système affiche le cours ajouté avec la liste des cours existant après vérification des données saisie par le formateur.

**Post conditions :**

Le cours est ajouté à la base de données.

**Diagramme de séquence du cas d'utilisation « consulter cours » :****Figure III.8 :** Diagramme de séquence de cas d'utilisation « consulter un cours »

**Description de cas d'utilisation « consulter cour » :****Sommaire d'identification****Titre** : suivre une séance d'enseignement.**Buts** : auto formation.**Acteur** : étudiant /apprenant.**Pré conditions :**

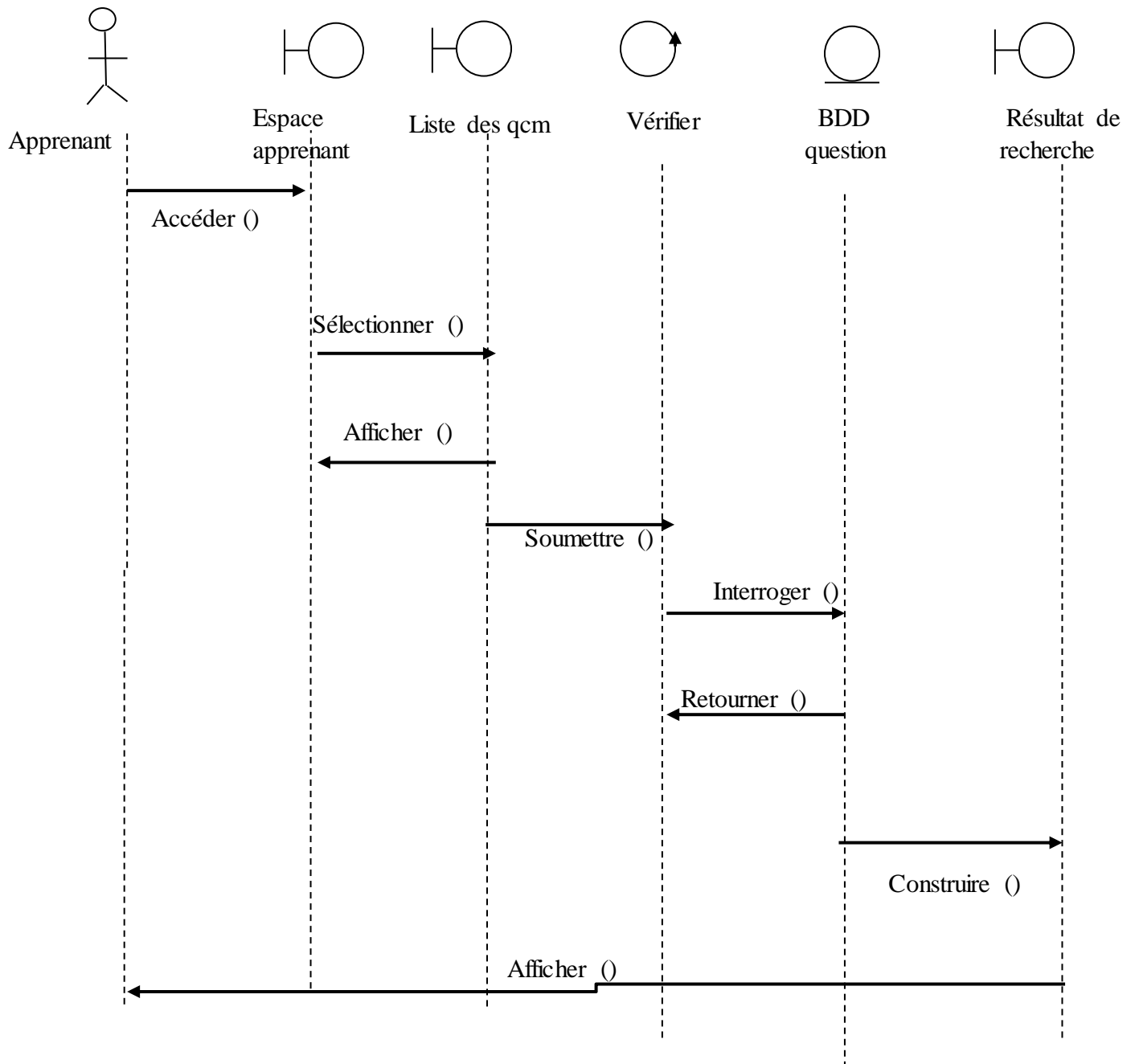
L'utilisateur saisit ses droits d'accès (nom d'utilisateur et mot de passe).

**Scénario nominal :****Enchaînement nominal :**

1. Après identification, l'apprenant atteint sa page d'accueil.
2. Le système répond en affichant la page d'accueil. Sur la page se trouvent les liens hypertextes suivants : Cours.
3. L'apprenant clique sur le lien « cours » ;
4. Le système charge depuis la base de données les cours existants.
5. L'apprenant télécharge son cours.
6. Le système affiche les cours téléchargées
7. L'apprenant consulte son cours

**Post conditions :**

L'utilisateur accède à son espace et télécharges les cours

**Diagramme de séquence du cas d'utilisation « faire qcm » :****Figure III.9 :** Diagramme de séquence de cas d'utilisation « faire un QCM »

**Description de cas d'utilisation « faire un qcm » :****Sommaire d'identification****Titre** : faire un qcm.**Buts** : auto évaluation.**Acteur** : étudiant /apprenant.**Pré conditions :**

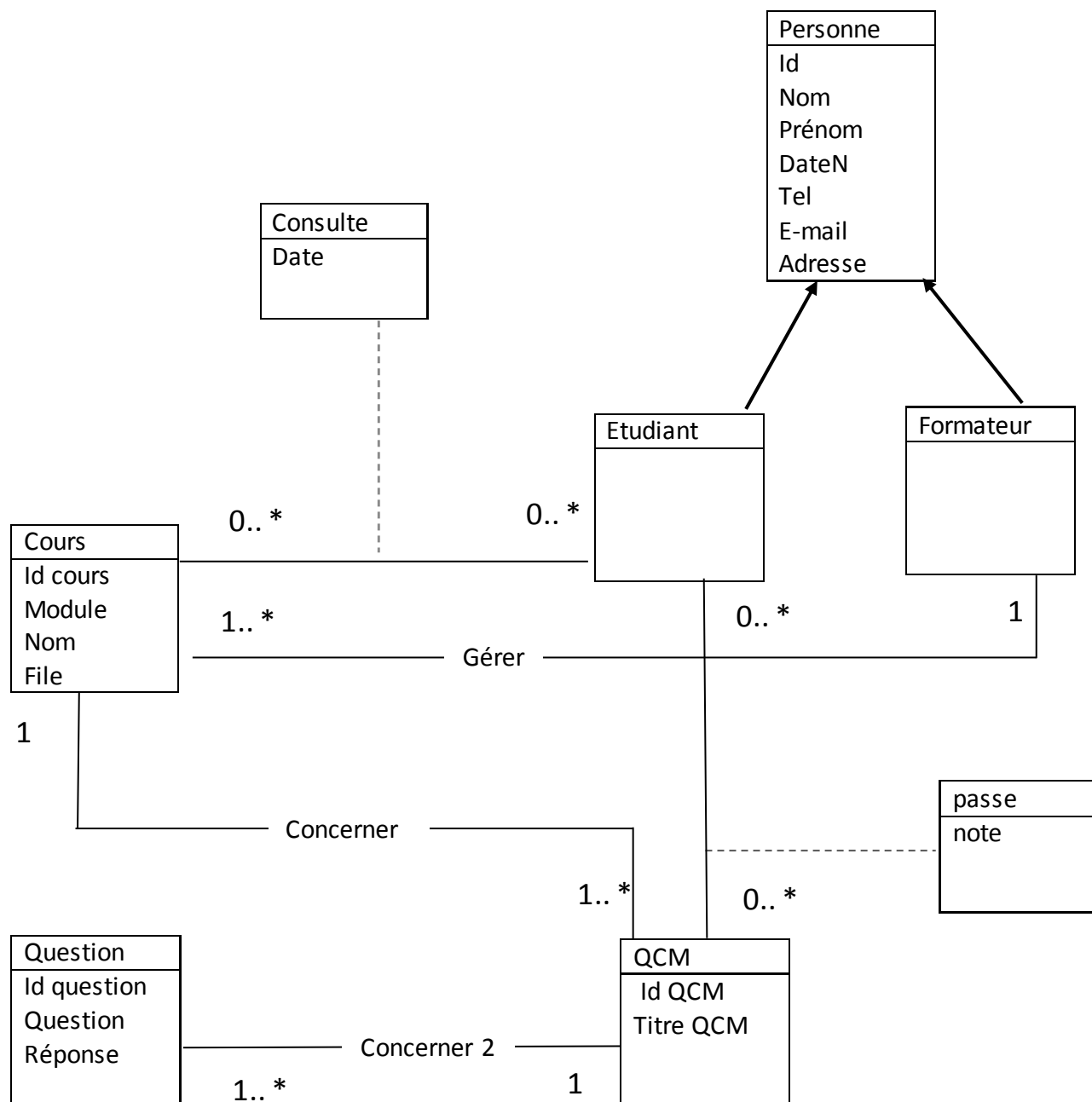
L'utilisateur saisit ses droits d'accès (nom d'utilisateur et mot de passe).

**Scénario nominal :****Enchaînement nominal :**

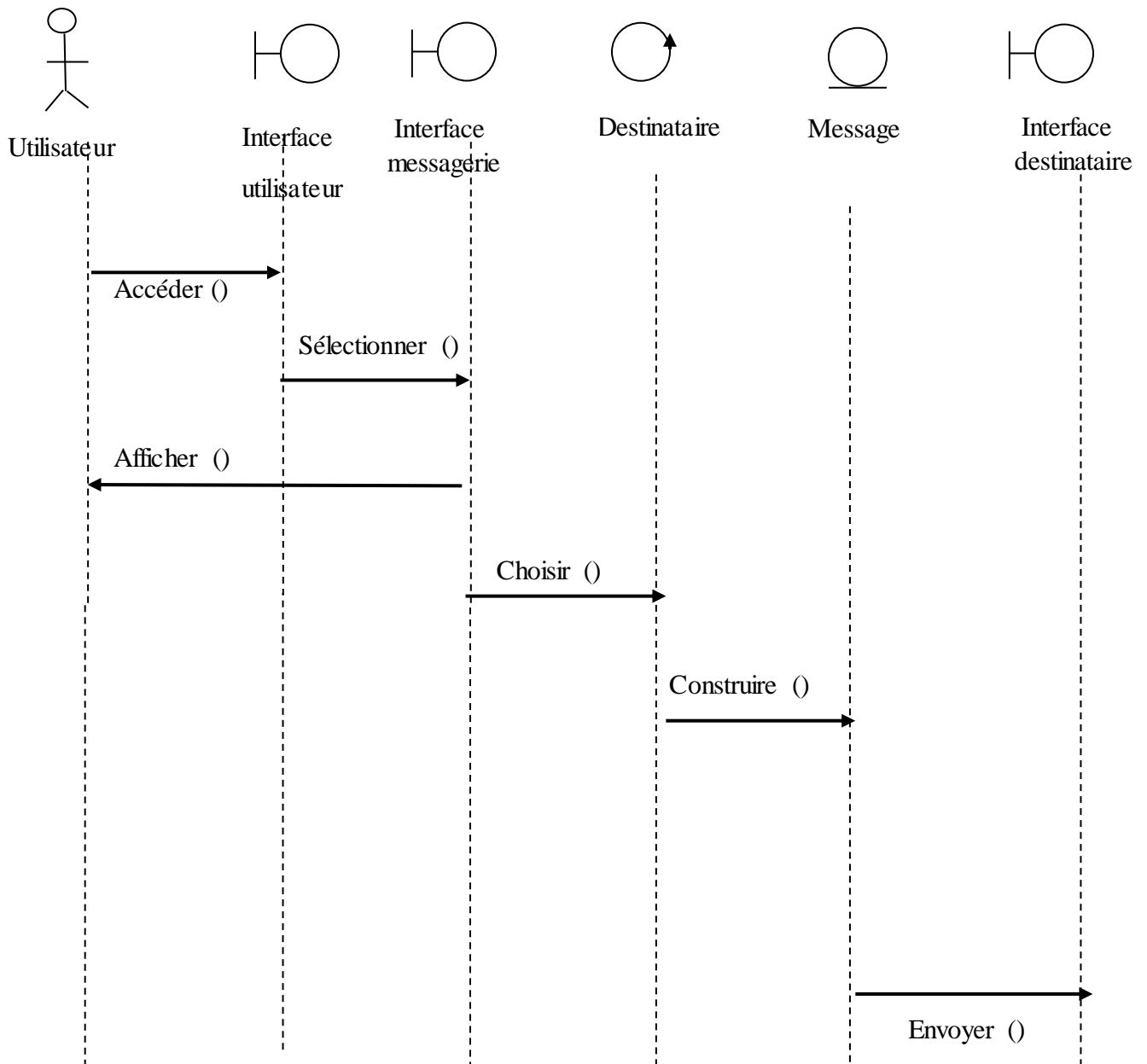
8. Après identification, l'apprenant atteint sa page d'accueil.
9. Le système répond en affichant la page d'accueil. Sur la page se trouvent le lien hypertexte suivant : test.
3. L'apprenant clique sur le lien « test ».
4. L'apprenant choisit le module dont il veut tester
5. Le système charge la liste des qcm.
6. L'apprenant choisit un qcm dans la liste.
7. Le système affiche le contenu du qcm.
8. L'apprenant répond au qcm et l'envoie au système pour le corriger.
9. Le système affiche la note obtenue par l'apprenant.

**Post conditions :**

L'utilisateur réussie à avoir une note d'évaluation

*III.4.2.2 Diagrammes de classe pour l'itération2 :***Figure III.10** : Diagrammes de classe pour l'itération 2



**III.4.3 Réalisation de l'itération 3 :***III.4.3.1 Diagrammes de séquence pour itération 3 :***Diagramme de séquence du cas d'utilisation « envoyer un message » :****Figure III.11 :** Diagramme de séquence de cas d'utilisation « envoyer un message »

**Description de cas d'utilisation** « envoyer un message » :

**Sommaire d'identification** :

**Titre** : envoyer un message

**Buts** : communication entre utilisateurs

**Acteur** : administrateur/formateur /apprenant

**Pré conditions** :

L'utilisateur saisit ses droits d'accès (nom d'utilisateur et mot de passe).

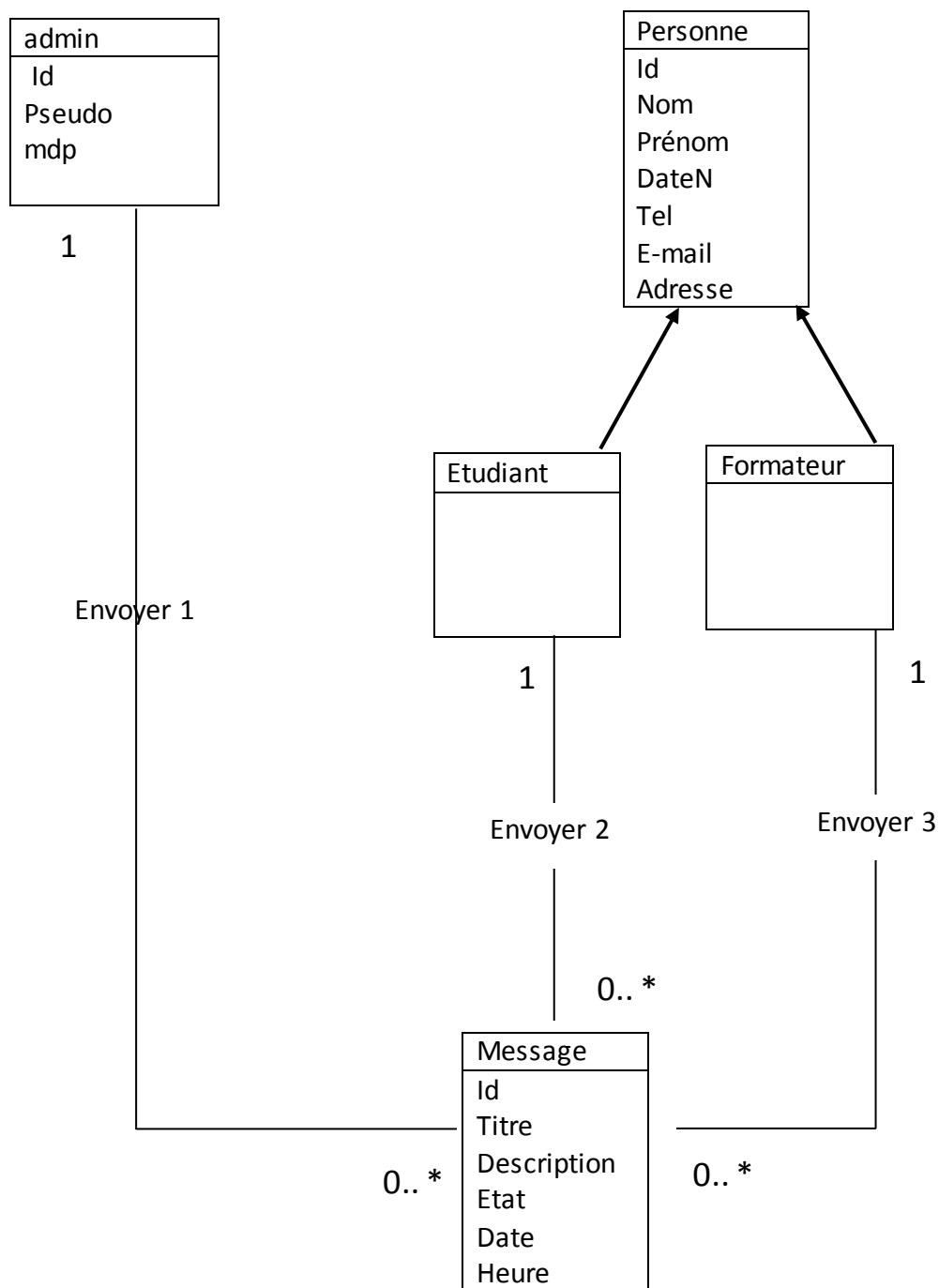
**Scénario nominal** :

**Enchaînement nominal** : après authentification

1. L'utilisateur accède à son espace
2. L'utilisateur choisie l'interface messagerie
3. L'utilisateur clique sur le lien envoyer un message
4. L'utilisateur choisie le destinataire et rédige le message et cliquez sur envoyé
5. Le destinataire reçoit le message dans son espace messagerie.

**Post conditions** :

L'utilisateur a envoyé son message.

*III.4.3.2 Diagrammes de classe pour l'itération 3 :***Figure III.12 :** Diagrammes de classe pour l'itération 3

**III.5 Conclusion :**

Dans ce chapitre nous avons utilisé le processus unifié RUP pour l'analyse et la conception de notre application en utilisant le langage de modélisation UML. Nous avons commencé par la définition des besoins qui ont été traduits par la suite en formalismes graphiques à l'aide des diagrammes offerts par le langage UML s'appuyant sur l'analyse. La partie conception a mis en évidence les différents composants de l'application et cela dans le but de la construction de notre application.

Dans le chapitre suivant, qui touche à la réalisation, nous essayerons de construire notre application par itération.

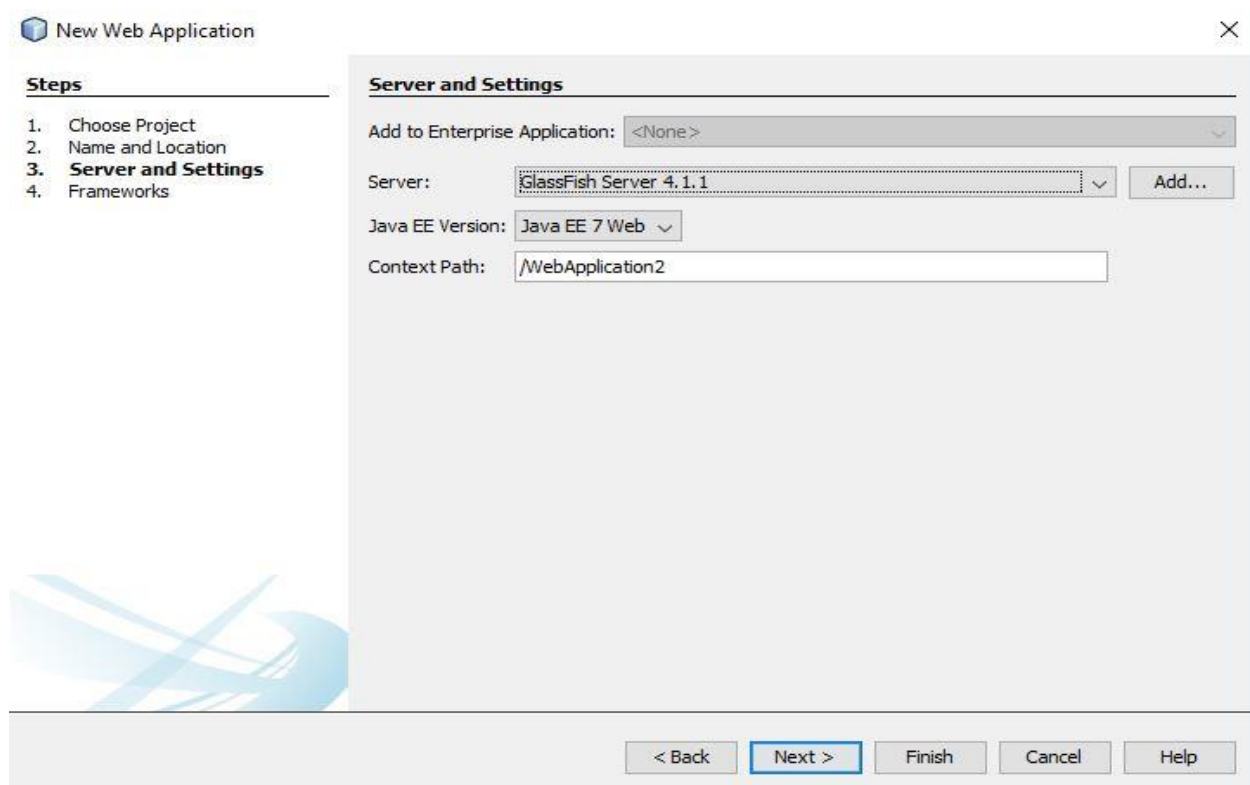
## IV.1 Introduction

Après avoir présenté dans le chapitre précédent les différentes étapes d'analyse et conception, nous allons présenter dans ce dernier chapitre l'environnement de développement, les outils qui ont servi à la réalisation de notre application, afin d'illustrer quelques fonctionnalités de l'application.

## IV.2 Les outils de développement :

### IV.2.1 GlassFishServer :

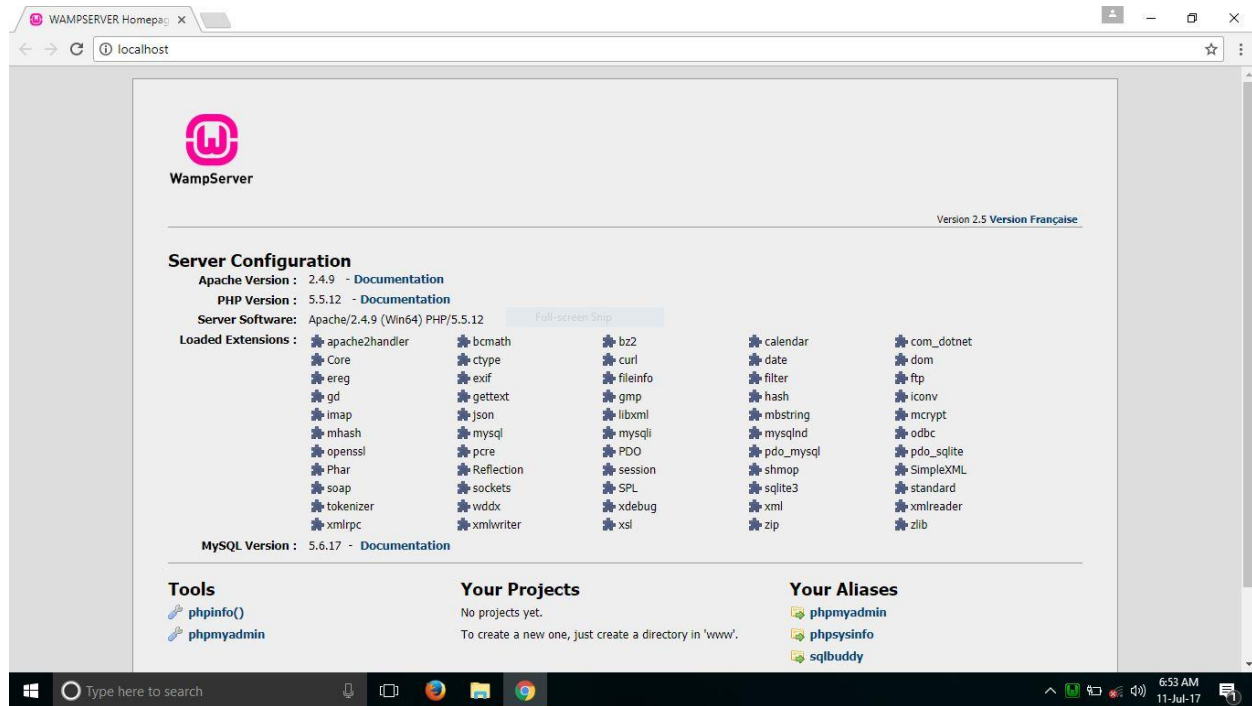
C'est un logiciel, écrit en java, qui permet de faire le mapping entre Objets Java et objets stockés en base relationnelle en assurant la persistance, et aussi s'occupe du transfert des classes Java dans les tables de la BDD et des types de données dans les types de données SQL.



**Figure IV.1:** Interface GlassFish

### IV.2.2 WampServer :

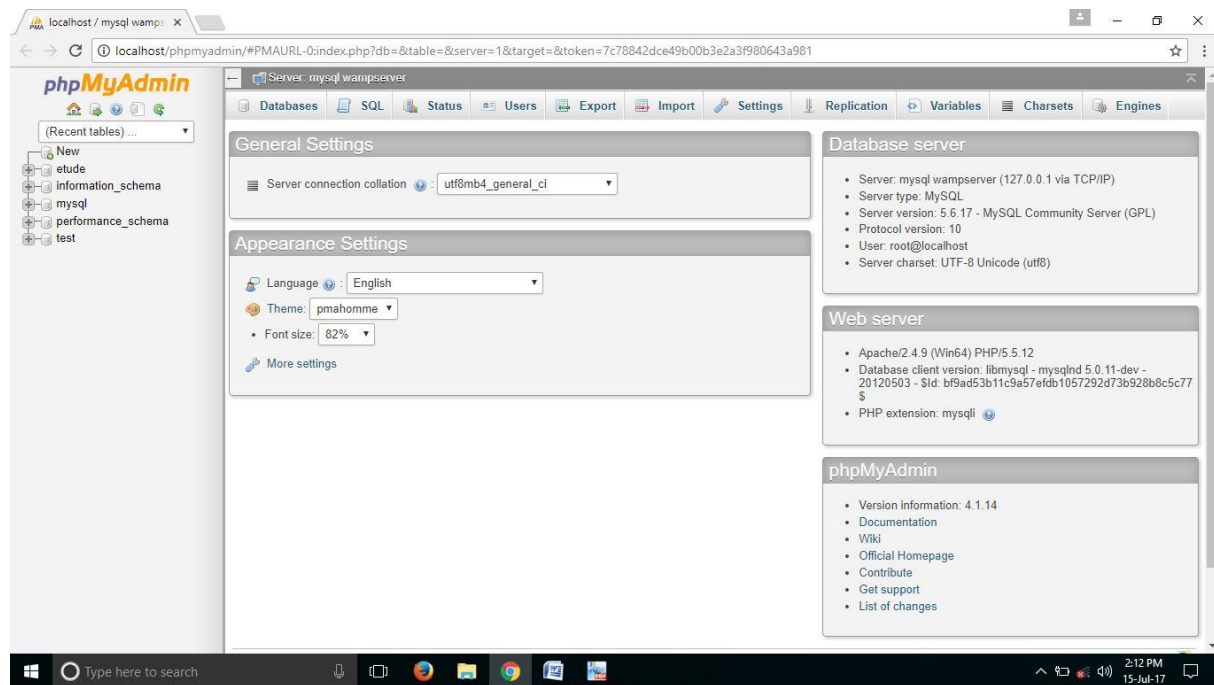
WampServer est une plate-forme de développement Web sous Windows pour des applications web dynamiques. Est un environnement comprenant deux serveurs (Apache etMySQL), un interpréteur de script (PHP) et PHPMyAdmin pour l'administration web des bases de données.



**Figure IV.2 :** Interface de WampServer

### IV.2.3 PhpMyAdmin :

PhpMyAdmin est une application web permettant de gérer un serveur de bases de données MySQL. Cette interface est écrite en PHP et s'appuie sur le serveur Apache. La figure suivante montre une copie d'écran de la page d'accueil de PhpMyAdmin.



**Figure IV.3 :** Interface de PhpMyAdmin

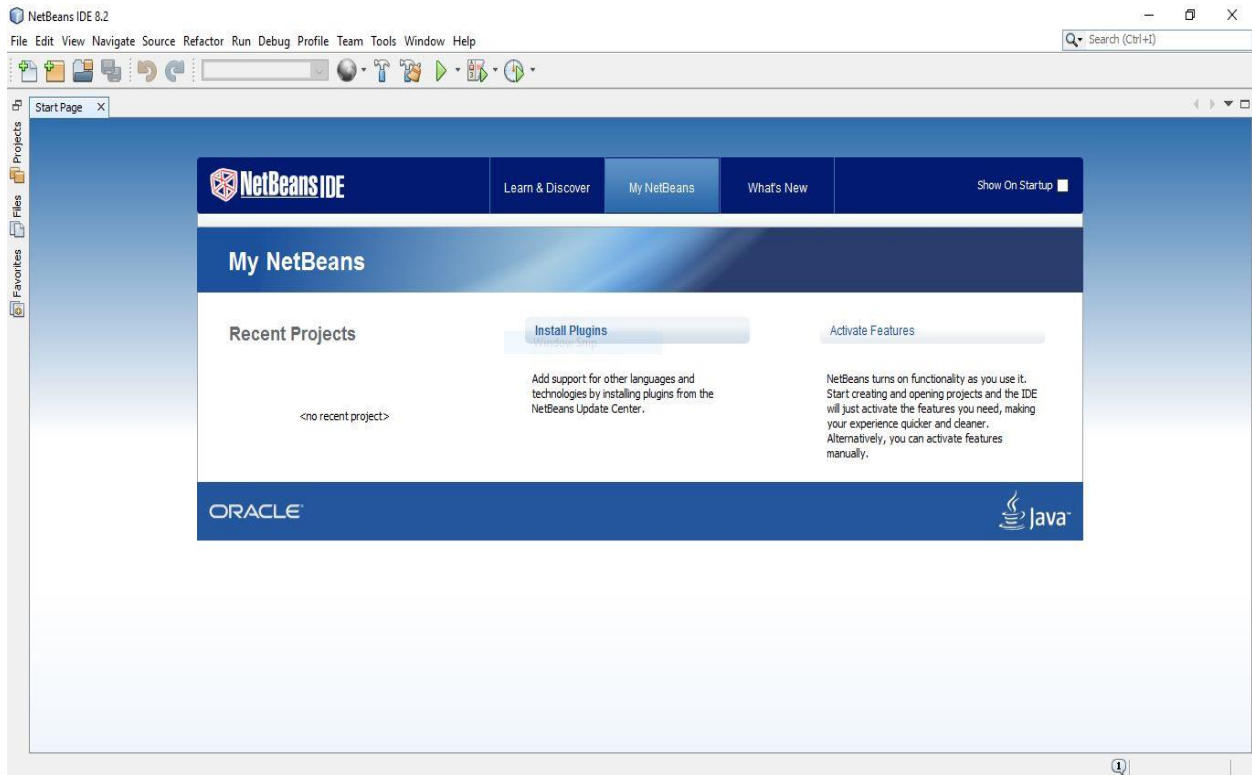
#### IV.2.4 NetBeans [10] :

NetBeans est un environnement de développement intégré (EDI), placé en open source par Sun en juin 2000 sous licence CDDL et GPLv2 (Common Development and Distribution License). En plus de Java, NetBeans permet également de supporter différents autres langages, comme Python, C, C++, JavaScript, XML, Ruby, PHP et HTML. Il comprend toutes les caractéristiques d'un IDE moderne (éditeur en couleur, projets multi-langage, refactoring, éditeur graphique d'interfaces et de pages Web).

Conçu en Java, NetBeans est disponible sous Windows, Linux, Solaris (sur x86etSPARC), Mac OS X ou sous une version indépendante des systèmes d'exploitation (requérant une machine virtuelle Java). Un environnement Java Development Kit JDKest requis pour les développements en Java.

NetBeans constitue par ailleurs une plateforme qui permet le développement d'applications spécifiques (bibliothèque Swing (Java)). L'IDE NetBeans s'appuie sur cette plateforme.

L'IDE NetBeans s'enrichit à l'aide de plugins. Il comprend toutes les caractéristiques d'un IDE moderne (coloration syntaxique, projet multi-langage, refactoring, éditeur graphique d'interfaces et de pages web, etc.).



**Figure IV.4 :** Interface de NetBeans



### **IV.3 Les langages de programmations :**

#### **IV.3.1 JAVA EE**

Java Enterprise Edition, ou **Java EE** (anciennement **J2EE**), est une spécification pour la technique Java de Sun plus particulièrement destinée aux applications d'entreprise. Dans ce but, toute implémentation de cette spécification contient un ensemble d'extensions au *Framework* Java standard (JSE, Java Standard Edition) afin de faciliter la création d'applications réparties.

#### **IV.3.2 XHTML : (Extensible Hyper texte Markup Language)**

XHTML est un langage balisé servant à l'écriture de pages du World Wide Web. XHTML est le successeur de HTML (de l'anglais HyperText Markup Language), XHTML respectant la syntaxe définie par XML, plus récente et plus simple que la syntaxe définie par SGML respectée par HTML.

#### IV.4 Le modèle Relationnel :

Le passage vers le modèle relationnel donne lieu à un modèle logique de données qu'on écrit sous une forme relationnelle comme suit :

Login (**Id\_log**, pseudo, mdp, type, id\_utilisateur).

Etudiant (**Id\_etu**, nom, prenom, dateN, tel, mail, adresse, id\_log).

Formateur (**Id\_form**, nom, prenom, dateN, tel, mail, adresse, module, id\_log).

Cours (**Id\_cours**, module, nom, file, id\_form, id\_QCM).

Message (**id\_msg**, titre, description, etat, date, heure, id\_expediteur, id\_recepteur).

Qcm (**Id\_QCM**, titre, id\_cours).

Question (id\_question, question, réponse, id\_qcm).

Consulter (id\_cours, id\_etu, date)

Passer (id\_QCM, id\_etu, note)

#### IV.5 Le modèle physique de données :

Ce modèle nous donne la représentation physique de l'ensemble des tables de la base de données du système étudié.

##### Description des tables :

###### ➤ La table « login » :

Nom de champ	désignation	Types de données	Clé(s)
<b>Id_log</b>	Identifiant de compte	Int(11)	Primaire
Pseudo	Nom d'utilisateur	Varchar (100)	
Mdp	Mot de passe	Varchar(100)	
<u>Id personne</u>	Identifiant de l'utilisateur	Int(11)	Etrangère

➤ **La table « Etudiant » :**

Nom de champ	désignation	Types de données	Clé(s)
<b>Id_etu</b>	Identifiant de l'étudiant	Int(11)	Primaire
Nom	Nom de l'étudiant	Varchar (30)	
Prenom	Prénom de l'étudiant	Varchar(30)	
dateN	Date de naissance de l'étudiant	Date	
Tel	Numéro de téléphone de l'utilisateur	Varchar(10)	
Email_etu	Email de l'étudiant	Varchar(45)	
Adresse_etu	Adresse de l'étudiant	Varchar(45)	
<u>Id_log</u>	Identifiant du compte	Int(11)	Etrangère

➤ **La table « Formateur » :**

Nom de champ	Désignation	Types de données	Clé(s)
<b>Id_form</b>	Identifiant du formateur	Int(11)	Primaire
Nom	Nom du formateur	Varchar (30)	
Prenom	Prénom du formateur	Varchar(35)	
dateN	Date de naissance du formateur	Date	
Tel	Numéro de téléphone du formateur	Varchar(10)	
Email_form	Email du formateur	Varchar(45)	
Adresse_form	Adresse du formateur	Varchar(45)	
Id_log	Identificateur du compte	Int(11)	Etrangère

➤ **La table cours :**

Nom de champ	Désignation	Types de données	Clé(s)
<b>Id_cours</b>	Identifiant du cours	Int(11)	Primaire
module	Titre de module	Varchar (30)	
Nom	Titre de cour	Varchar(35)	
File	L'url de cours	Varchar (45)	
Id_form	Identifiant de formateur	Int(11)	Etrangère
Id_QCM	Identifiant de QCM	Int(11)	Etrangère

➤ **La table « Message » :**

Nom de champ	Désignation	Types de données	Clé(s)
<b>Id_msg</b>	Identifiant du message	Int(11)	Primaire
Titre	Titre de message	Varchar (30)	
description	Description de message	Varchar(30)	
etat	Etat de message	Varchar (30)	
Date	Date de message	date	
heure	Heure de message	Int(11)	
<u>id_expediteur</u>	Identifiant de l'expéditeur	Int(11)	Etrangère
<u>id_recepteur</u>	Identifiant de formateur	Int(11)	Etrangère

## ➤ La table « QCM » :

Nom de champ	Désignation	Types de données	Clé(s)
<b>Id_QCM</b>	Identifiant du QCM	Int(11)	Primaire
Titre	Titre de QCM	Varchar (30)	
Id_cours	Identifiant du cours	Int(11)	Etrangère

## ➤ La table « Question » :

Nom de champ	Désignation	Types de données	Clé(s)
<b>Id_question</b>	Identifiant question	Int(11)	Primaire
question	Description	Varchar(100)	
réponse	Réponse	Varchar(30)	
id_qcm	Identifiant de QCM	Int(11)	Etrangère

## ➤ La table « consulter »

Nom de champ	Désignation	Types de données	Clé(s)
Id_cours	Identifiant du cours	Int(11)	
Id_qcm	Identifiant du QCM	Int (11)	
Date	Date de consultation	Date	

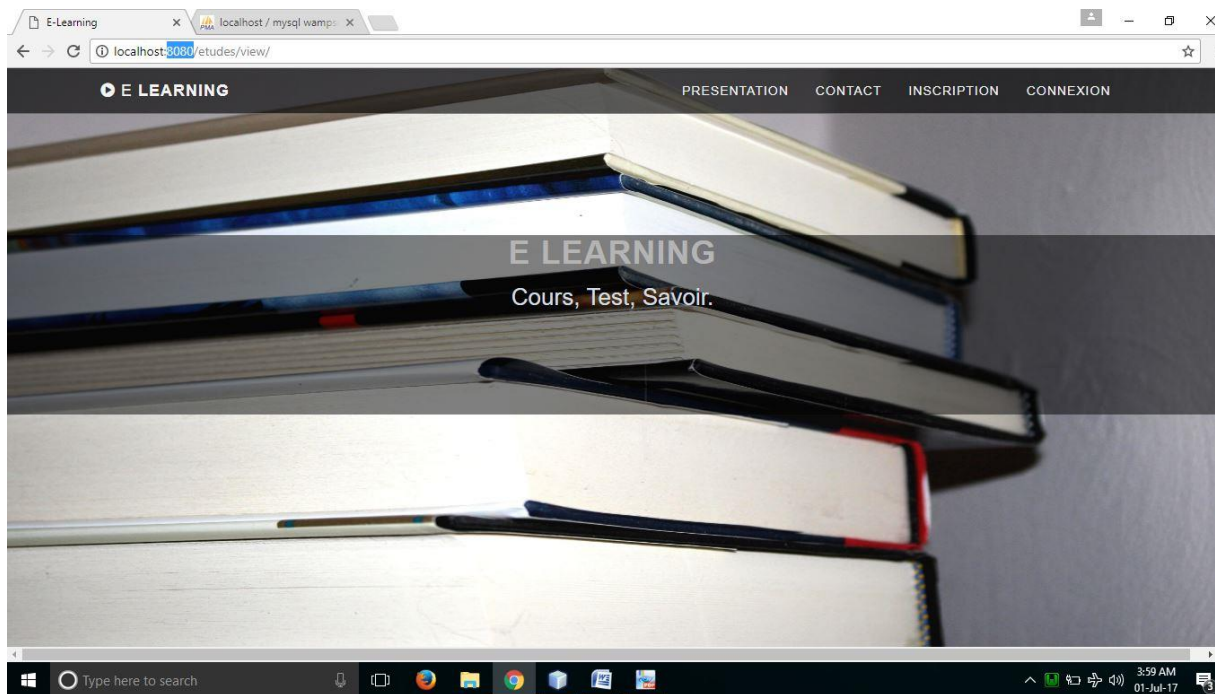
➤ **La table « passer »**

Nom de champ	Designation	Types de données	Clé(s)
Id_QCM	Identifiant du QCM	Int(11)	
Id_etu	Identifiant de l'étudiant	Int(11)	
Note	La note obtenue	Int (5)	

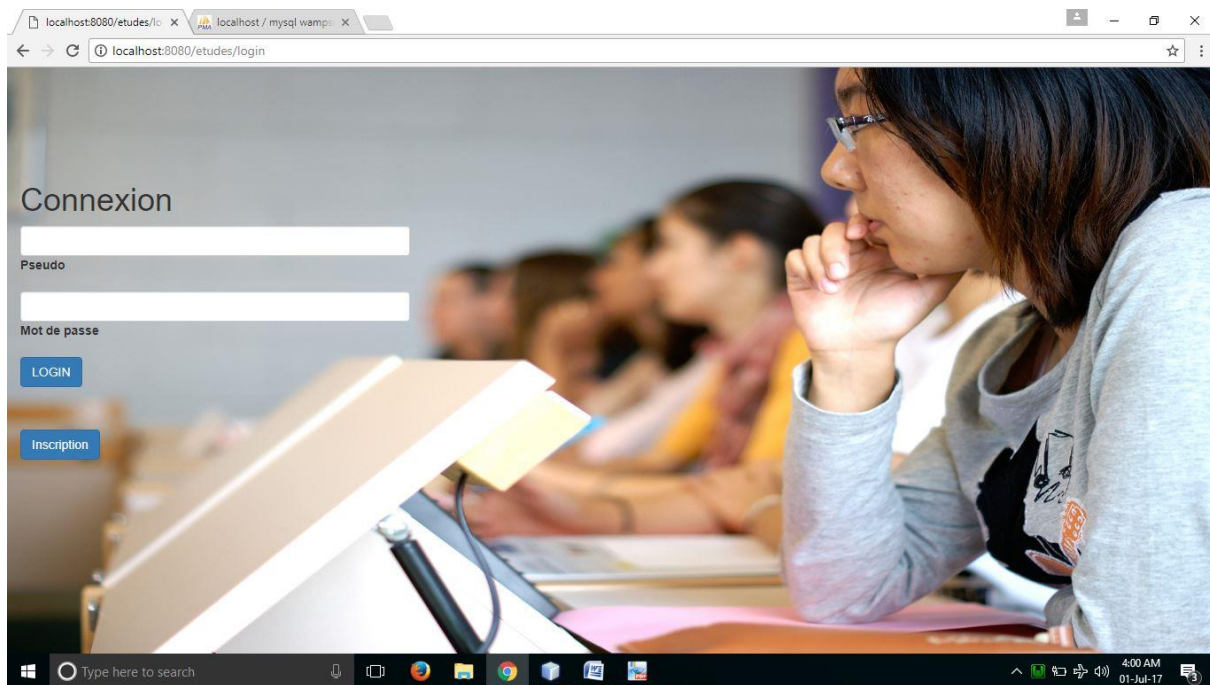
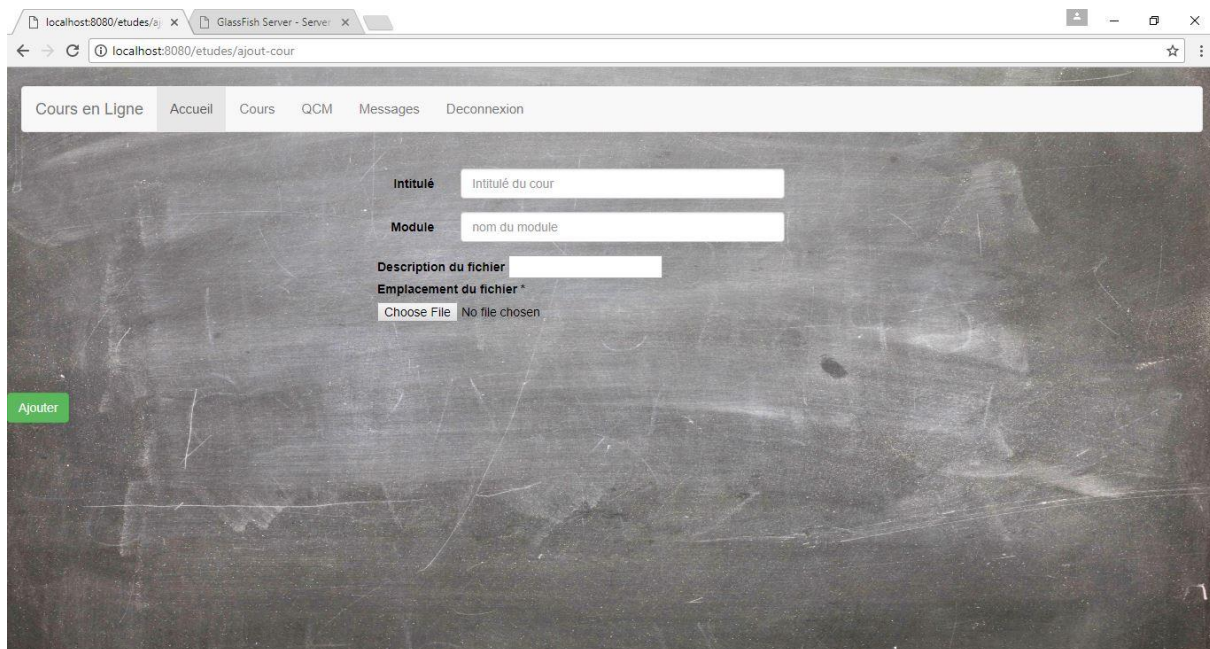
## IV.6 Présentation de quelques interfaces :

Nous n'allons présenter quelques interfaces :

### Présentation d'interface d'accueil :



**Figure IV.5:** Interface page d'accueil.

**Présentation d'interface authentification :****Figure IV.6:** Interface page d'authentification**Présentation d'interface « ajouter un cours » :****Figure IV.7:** interface « ajouter un cours »

**Présentation d'interface « ajouter un QCM » :**

localhost:8080/etudes/ajout-qcm

Cours en Ligne Accueil Cours QCM Messages Deconnexion

Titre du Qcm Titre du QCM

Question Question 1

☐ vrai ☐ faux

Question Question 2

☐ vrai ☐ faux

Question Question 3

☐ vrai ☐ faux

Question Question 4

☐ vrai ☐ faux

Question Question 5

☐ vrai ☐ faux

Ajouter

**Figure IV.8:** interface « ajouter un QCM »



**IV.7 Conclusion :**

Dans ce chapitre nous avons présenté les outils et l'environnement développement de L'application. Nous avons en outre explicité les composants de la base de données, puis pour terminer nous avons présenté quelques interfaces de l'application.

## **Conclusion générale :**

La réalisation de ce travail nous a permis d'approfondir nos connaissances en système d'information et de mettre pratique les théories acquises durant notre cursus universitaire, ainsi :

- D'avoir les éléments de base d'une nouvelle discipline qui ne cesse de se développer aussi bien dans le secteur public que dans le secteur privé : l'e-learning dit aussi enseignement à distance.
- D'acquérir de nouvelles connaissances sur les langages HTML, Java EE, JavaScript
- De maîtriser les concepts et les outils de la méthode de conception UML. Ainsi que la démarche de conception orientée objet (RUP, Rational Unified Process).
- D'avoir une expérience sur l'utilisation des logiciels tels que NetBeans, wamp server
- De travailler avec le système de gestion de base de données MySQL, le serveur Apache.

Pour mener à terme notre projet, nous avons jugé nécessaire de présenter quelques généralités sur la technologie Java EE, le concept client/serveur, ainsi que les technologies de développement d'application java existantes. Ensuite, nous avons présenté l'état de l'art de E-Learning passant à l'étape de conception et de réalisation afin d'aboutir à la mise en œuvre de notre application.

En guise de perspectives, les possibilités d'évolutions de ce projet sont nombreuses, a l'heure ou le e-learning envahi le marché de l'éducation et promet un fort changement des habitudes de travail des utilisateurs et des formateurs nous espérons que notre projet puisse être un outil modèle facilitant le développement d'applications dédiées à l'enseignement à distance pour les promotions à venir.

## **Annexe A : langage de modélisation UML**

### **1. Introduction :**

Aujourd'hui l'approche objet occupe une place prépondérante dans le génie logiciel, or la conception de toute application logicielle doit être traitée d'une manière précise et détaillée, précéder par une analyse profonde et bien réfléchie, vue qu'elle est le reflet du futur système avant même sa réalisation. Dans le but de formaliser l'analyse et la conception de notre système nous avons opté pour la méthodologie UML, un langage graphique de modélisation s'appuyant sur des normes très structurantes.

### **2. Définition UML[11] :**

UML (Unified Modeling Language) est un langage de modélisation unifié, un standard conçu pour l'écriture de plans d'élaboration de logiciels. Il peut être utilisé pour visualiser, spécifier, construire et documenter les détails d'un système logiciel. Issue d'un travail de fond mené par les trois compères James Rumbaugh, Ivar Jacobson et Grady Booch en 1995, la méthode dite unifiée a été ensuite consacrée par l'organisme international de normalisation OMG (*Object Management Group*) en 1997 avec la diffusion de la première version de la norme : UMLV1.1.

### **3. Quelques notions de base :**

#### **Acteur :**

Un acteur représente un ensemble de rôles joués par une personne, une machine ou un autre système, il peut interagir avec un cas d'utilisation et participer à son scénario, il est représenté par un petit bonhomme avec son rôle inscrit dessous :

#### **Acteur**



**Taches :** est un ensemble de fonctions qu'un acteur bien défini peut accéder.

**Scénario :** c'est une succession particulière d'enchaînement, s'exécutant du début à la fin du cas d'utilisation.

## **4. Présentation générale des diagrammes d'UML :**

UML dans sa version 2 propose treize diagrammes qui peuvent être utilisés dans la description d'un système. Ces diagrammes sont regroupés dans deux grands ensembles :

### **4.1 Les diagrammes structurels :**

Ces diagrammes, au nombre de six, ont vocation à représenter l'aspect statique d'un système (classes, objets, composants...).

**Diagramme de classe** : Ce diagramme représente la description statique du système en intégrant dans chaque classe la partie dédiée aux données et celle consacrée aux traitements. C'est le diagramme pivot de l'ensemble de la modélisation d'un système.

**Diagramme d'objet** : Le diagramme d'objet permet la représentation d'instances des classes et des liens entre instances.

**Diagramme de composant** : Ce diagramme représente les différents constituants du logiciel au niveau de l'implémentation d'un système.

**Diagramme de déploiement** : Ce diagramme décrit l'architecture technique d'un système avec une vue centrée sur la répartition des composants dans la configuration d'exploitation.

**Diagramme de paquetage** : Ce diagramme donne une vue d'ensemble du système structuré en paquetage. Chaque paquetage représente un ensemble homogène d'éléments du système (classes, composants...).

**Diagramme de structure composite** : Ce diagramme permet de décrire la structure interne d'un ensemble complexe composé par exemple de classes ou d'objets et de composants techniques. Ce diagramme met aussi l'accent sur les liens entre les sous-ensembles qui collaborent.

### **4.2 Les diagrammes de comportement :**

Ces diagrammes représentent la partie dynamique d'un système réagissant aux événements et permettant de produire les résultats attendus par les utilisateurs. Sept diagrammes sont proposés par

UML :

**Diagramme des cas d'utilisation** : Ce diagramme est destiné à représenter les besoins des utilisateurs par rapport au système. Il constitue un des diagrammes les plus structurants dans l'analyse d'un système.

**Diagramme d'état-transition (machine d'état)** : Ce diagramme montre les différents états des objets en réaction aux événements.

## **Annexe A : langage de modélisation UML**

**Diagramme d'activités :** Ce diagramme donne une vision des enchaînements des activités propres à une opération ou à un cas d'utilisation. Il permet aussi de représenter les flots de contrôle et les flots de données.

**Diagramme de séquence :** Ce diagramme permet de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets.

**Diagramme de communication (anciennement appelé collaboration) :** Ce diagramme est une autre représentation des scénarios des cas d'utilisation qui met plus l'accent sur les objets et les messages échangés.

**Diagramme global d'interaction :** Ce diagramme fournit une vue générale des interactions décrites dans le diagramme de séquence et des flots de contrôle décrits dans le diagramme d'activités.

**Diagramme de temps :** Ce diagramme permet de représenter les états et les interactions d'objets dans un contexte où le temps a une forte influence sur le comportement du système à gérer.

- ❖ Pour le cas de notre application nous avons utilisées les diagrammes suivant :
  - ✓ les diagrammes de cas d'utilisation.
  - ✓ les diagrammes de séquences.
  - ✓ les diagrammes de classes.

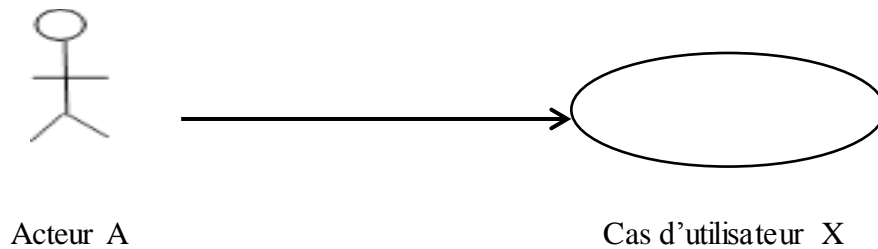
### **5. Les diagrammes de cas d'utilisation :**

- Il s'agit de la solution UML pour représenter le modèle conceptuel( est de comprendre et structurer les besoins du client.)
- Les use cases permettent de structurer les besoins correspondants d'un système.
- Ils centrent l'expression des exigences du système sur ses utilisateurs : ils partent du principe que les objectifs du système sont tous motivés.
- Ils se limitent aux préoccupations "réelles" de solutions d'implémentation et ne forment pas un inventaire fonctionnel du système.
- Ils identifient les utilisateurs du système (acteurs) et leur interaction avec le système.
- Ils permettent de classer les acteurs et

## **Annexe A : langage de modélisation UML**

- Ils servent de base à la traçabilité des exigences d'un système dans un processus de développement intégrant UML.

### **Exemple :**



### **5.1 Intérêt des cas d'utilisation :**

- Le but de la conceptualisation est de comprendre et structurer les besoins du client.
- Il ne faut pas chercher l'exhaustivité, mais clarifier, filtrer et organiser les besoins
- Une fois identifiés et structurés, ces besoins définissent le contour du système à modéliser (ils précisent le but à atteindre)

Permettent d'identifier les fonctionnalités principales (critiques) du système.

- Le modèle conceptuel doit permettre une meilleure compréhension du système.
- Le modèle conceptuel doit servir d'interface entre tous les acteurs du projet.
- Les besoins des clients sont des éléments de traçabilité dans un processus intégrant UML.

## **Annexe A : langage de modélisation UML**

### **6. Les diagrammes de séquences :**

C'est un diagramme d'interaction. Il représente un ensemble d'objets et leurs relations , avec les messages qu'ils échangent.(aspect chronologique des messages).

Dans un diagramme de séquence, deux types de messages peuvent être distingués :

- **Message synchrone**

Dans ce cas l'émetteur reste en attente de la réponse à son message avant de poursuivre ses actions. La flèche avec extrémité pleine symbolise ce type de message. Le message retour peut ne pas être représenté car il est inclus dans la fin d'exécution de l'opération de l'objet destinataire du message. .

- **Message asynchrone**

Dans ce cas, l'émetteur n'attend pas la réponse à son message, il poursuit l'exécution de ses opérations. C'est une flèche avec une extrémité non pleine qui symbolise ce type de message.

### **7. Les diagrammes de classes :**

Le diagramme de classe constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est centrée sur les concepts de classe et d'association. Chaque classe se décrit par les données et les traitements dont elle est responsable pour elle-même et vis-à-vis des autres classes.

### **8. Conclusion :**

L'UML, comme l'on a vu à travers ce chapitre, ne propose pas une démarche objet mais une notation adapté au monde de aux techniques de modélisation objet. La notation UML peut s'adapter a tous les projets informatiques.

### **1. Introduction :**

Processus unifié (up) Le processus unifié est un processus de développement logiciel : il regroupe les activités à mener pour transformer les besoins d'un utilisateur en système logiciel. Caractéristiques essentielles du processus unifié :

- Le processus unifié est à base de composants,
- Le processus unifié utilise le langage UML (ensemble d'outils et de diagramme),
- Le processus unifié est piloté par les cas d'utilisation,
- Centré sur l'architecture.
- Itératif et incrémental

### **2. RUP (Rational Unified Process ) [12] :**

**RUP®** (*Rational Unified Process*) est un processus basé sur une approche disciplinée afin de bien maîtriser l'assignation des tâches et la responsabilisation des différents acteurs participant au cycle de développement du logiciel. RUP a pour objectif principal de faire appliquer les bonnes pratiques de développement aux entreprises, ce qui confère au produit final une meilleure qualité.

Dérivée de UP, Commercialisée par IBM en 1998, RUP se veut être un modèle évolutif qui doit être configuré pour pouvoir être utilisé en intégrant les contraintes, les spécificités et l'historique de l'organisation qui l'adopte.

#### **2.1 Méthode de développement logiciel :**

- itérative.
- Incrémentale
- pilotée par les cas d'utilisation.
- centrée sur l'architecture et la réduction des risques
- Produit de qualité

#### **2.2. Phases de RUP :**

##### **2.2.1. Phase d'initialisation :**

- Définir la portée du projet
- Spécification
- Affectation des tâches



## **Annexe B : Présentation de RUP**

- Evaluation des risques
- Cycle de vie

### **2.2.2 Phase d'élaboration :**

- Planifier le projet,
- spécifier les fonctionnalités
- construire l'architecture
- Spécification du produit
- Conception de l'architecture
- Planification activités-ressources

### **2.2.3 Phase de construction :**

- Construire le produit
- Implémentation du produit
- Tests
- produit opérationnel

### **2.2.4 Phase de transition :**

- Transition du produit vers les utilisateurs
- Livraison
- Formation
- Qualité

## **2.3.Cycle de vie de RUP :**

### **✓ Exigence :**

Détermination des besoins

- fonctionnels (ce que l'on attend du système)
- Non fonctionnels (fiabilité, temps de réponse, environnement distribué, etc.)

### **✓ Analyse et conception :**

Evoluer depuis la spécification des besoins jusqu'à une solution informatique Analyse besoins fonctionnels Conception intègre aussi les besoins non fonctionnels

### **✓ Implémentation :**

## Annexe B : Présentation de RUP

Transcription dans un langage de programmation ou de base de données .

Utilisation de composants existants

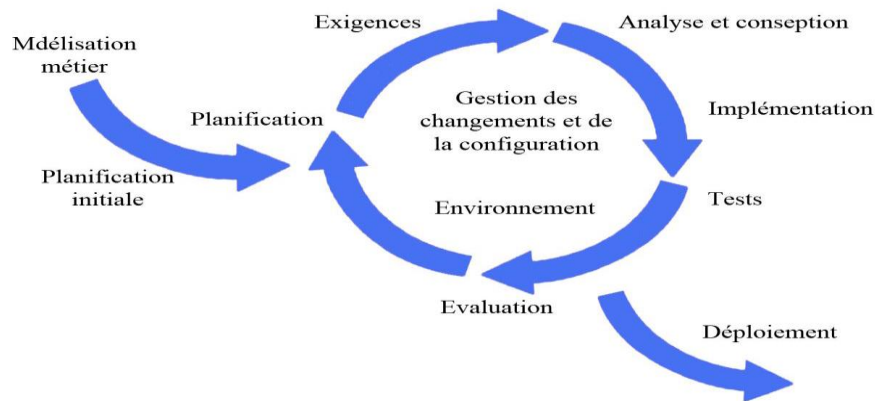
### ✓ **Test :**

Estimer si les besoins sont satisfaits. S'il y a des erreurs/défauts à corriger Renforcer et stabiliser l'architecture

### ✓ **Déploiement :**

Distribuer le logiciel dans son environnement opérationnel, Installation, test Formation des utilisateurs, Migration des données

Cycle de vie de RUP :



## 2.4 Différence entre incrémental et itératif

### ✓ **RUP est incrémental**

- Le produit final est livré en incréments

Chaque incrément livre une fonctionnalité ayant une valeur pour l'utilisateur final

Chaque incrément consiste en un ensemble de cas d'utilisation

Les incréments doivent être de courte durée (quelques semaines à quelques mois)

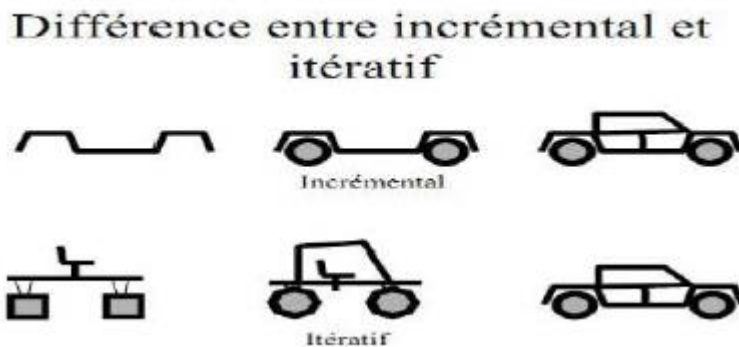
Les incréments sont des mini projets dans un projet

## **Annexe B : Présentation de RUP**

### **✓ RUP est itératif**

Une itération est une séquence d'activités qui répond à un plan et à des critères d'évaluation et qui produit une version exécutable.

### **Exemple :**



### **❖ Forces de la méthode RUP**

Traçabilité à partir des Uses Cases jusqu'au déploiement

Approche basée sur l'architecture

Gestion des risques dans les projets

### **❖ Faiblesses de la méthode RUP**

Coût de personnalisation souvent élevés

Très axé processus

### **3. Conclusion :**

Cette partie nous a permis de prendre connaissance le processus RUP (rationnel unified process) est une méthode de développement pour logiciel orienté objet, c'est une méthode générique itérative et incrémentale

## Bibliographie / webographie

- [1] [http://fr.wikipedia.org/wiki/Architecture\\_logicielle](http://fr.wikipedia.org/wiki/Architecture_logicielle)
- [2] <http://portail.org/webntic/Client-Serveur-presentation-12.html>
- [3] Java Entreprise Edition (JEE) Développer les applications web avec  
JEEThierry Groussard, François-Xavier Sennesal édition Eni, 14/10/2015
- [4] mémoire promotion 2012/2013 réalisé par : Amrane Samia et Amrane hassina  
(conception et réalisation d'une application Java EE pour la gestion de la  
scolarité).
- [5] La FOAD: qu'est-ce que c'est ? <http://www.chez.com/capfoad/chapfoad.htm>
- [6] E-Learning: Mode d'emploi. <http://www.demos.fr/lexique.asp>
- [7] Eric Ecoutin, Mise en œuvre des plates-formes pour la formation ouverte et à  
distance, janvier 2001
- [8] <http://MonWebCt.fr>
- [9] <http://www.moodle.org>
- [10] <http://fr.wikipedia.org/wiki/netbeans>
- [11] UML2 : Analyse et conception par : Joseph Gabay, David Gabay, Edition  
Dunod, Paris, 2008
- [12] Méthode de développement « RUP ».pdf, par Michel Winter :M2  
MIAG,NTDP