

REPUBLIQUE ALGERIENNE DEMOCRATIQUE
ET POPULAIRE

Ministère de l'Enseignement et de la
Recherche Scientifique

Université Mouloud Mammeri de Tizi-Ouzou



Faculté de Génie Electrique et d'Informatique
Département d'Informatique

Mémoire de fin d'études

En vue de l'obtention du diplôme de master II
Option : Système Informatique

Thème

**Intégration des technologies du web2.0 pour l'extension
d'un CMS**

Réalisé par:

M^{elle} DJEDDI Sabrina

Encadré par :

M^r R.AHMED-OUAMER

Année Universitaire: 2011/2012



Remerciements

*D'abord je tiens à remercier **ALLAH** de m'avoir accordé la force et le courage de mener à terme ce projet.*

Toute ma reconnaissance et toute ma gratitude vont vers mes professeurs qui m'ont appris le savoir.

*Je tiens à remercier sincèrement mon promoteur Monsieur **AHMED OUAMAR Rachid** qui s'est toujours montré à l'écoute et très disponible tout au long de la réalisation de ce mémoire, ainsi pour l'inspiration, l'aide et le temps qu'il a bien voulu me consacrer.*

Je remercie les membres du jury d'avoir accepté de juger mon travail.

J'exprime ma gratitude à tous les consultants rencontrés lors des recherches effectuées et qui ont accepté de répondre à mes questions avec gentillesse.

*Je n'oublie pas mes parents pour leur contribution, leur soutien et leur patience. Je tiens à exprimer ma reconnaissance envers **khali Mohammed** pour son aide précieuse et le temps qu'il m'a consacré.*

A tous mes proches qui ont toujours eu la patience de m'entendre et de me redonner confiance et qui ont compris l'importance d'un tel travail.



Dédicaces

*A mes très chers parents « Yemma » et « Vava »,
Témoignage d'affection et de grande reconnaissance,
Que Dieu les garde*

A toute ma famille,

A mes très chers frères Toufik, Hamid et Zizou

A ma grand-mère et mes très chères tantes Nouara, Hamida et Zohra

A mon oncle Abderahmane, sa femme et leurs enfants Dihia, Salah et Youcef

A mon oncle Malek, sa femme et leur fille Amel

A mon oncle Mohammed, sa femme Sabrina et leur perle Laeticia

A tous mes amis

A tous ceux que j'aime

A tous ceux qui ont participé de près ou de loin à la réalisation de ce travail.

*Avec l'expression de tous mes sentiments de respect,
Je dédie ce modeste travail.*

Résumé

Les contenus, au cœur des ressources du web, nécessitent de nouveaux outils pour prendre en compte les technologies du web2.0. Ces outils doivent être envisagés aussi bien sur le plan théorique en revisitant les modèles actuels de la recherche d'information, de l'indexation et de l'accès aux contenus, que sur le plan technique en proposant de nouvelles architectures et de nouvelles fonctionnalités basées sur les approches SOA qui tireraient profit des technologies telles que les services web.

Le web 2.0 doit également modifier en profondeur les usages car la participation à la production des contenus se fait de façon collaborative et nécessite alors de nouveaux outils qui apportent une valeur ajoutée dans la chaîne de production de contenu. Les CMS (Content Management System) vont ainsi connaître des évolutions significatives sur le plan des fonctionnalités et des caractéristiques techniques. Il s'agit alors de répondre notamment à certaines questions :

- Quelles sont les applications concrètes et les services qui pourraient être utilisés dans un CMS donné ?
- Comment rendre accessible à l'utilisateur final des fonctionnalités basées sur des technologies de très haut niveau ?
- Est-il possible d'intégrer des fonctionnalités de type sémantique à des sites déjà existants ?

Objectifs : Le but du sujet est tout d'abord de faire l'état de l'art des techniques du web2.0 pertinentes pour les CMS ainsi que sur les services web (découverte notamment). Il s'agit ensuite de proposer un modèle d'intégration qui permettra d'intégrer dans une application de nouvelles fonctionnalités (mises en œuvre à travers des composants) de façon dynamique dans son CMS. Ces fonctions pourraient être envisagées sous forme de services en exploitant les caractéristiques des architectures SOA.

Mots clés : Architecture Orientée Services (SOA), Services web, web sémantique, CMS (Content Management System), e-learning

Abstract

The contents in the heart of web resources, require new tools to take into account the web2.0 technologies. These tools should be considered both theoretically by revisiting existing models of information retrieval, indexing and access to content that the technical proposing new architectures and new features based approaches that would benefit SOA technologies such as web services.

Web 2.0 is also fundamentally change the practice because participation in content production is a collaborative manner and then requires new tools that bring added value in the chain of content production. CMS (Content Management System) and will undergo significant changes in terms of features and specifications. This is particularly so to answer some questions:

- What are the practical applications and services that could be used in a given CMS?
- How to make it accessible to the end user of features based on technologies of the highest level?
- Is it possible to integrate semantic type features to existing sites?

Objectives: The aim of the subject is first of all to the state of the art techniques web2.0 relevant to CMS as well as web services (including decouverte). It is then to propose a model of integration that will integrate into application new features (implemented through components) dynamically in the CMS. These functions could be considered as a service by exploiting the characteristics of SOA.

Keywords: Service Oriented Architecture (SOA), Web Services, Semantic Web, CMS (Content Management System), e-learning

sommaire

Introduction générale.....	1
-----------------------------------	----------

CHAPITRE I : ETAT DE L'ART SUR LES TECHNIQUE DU WEB2.0

Introduction	2
I. Le web et son évolution	2
I.1 Définition	2
I.2 évolution du web.....	2
II. Le web 2.0	3
II.1 Évolution du web 1.0 vers le web 2.0	3
II.2 qu'est ce que le web 2.0	5
II.3 Principes du web2.0.....	6
II.4 Le panorama des techniques et technologies utilisées par le web 2.0	7
II.4.1 Les nouveaux langages.....	7
II.4.2 Nouvelles architectures réseaux	11
II.4.3 Nouvelles applications.....	12
II.4.4 Les outils collaboratifs	13
II.5 Les sites web 2.0.....	16
II.6 Tendances et perspectives	17
Conclusion.....	18

CHAPITRE II : E-LEARNING ET SYSTEMES DE GESTION DE CONTENU

Introduction	19
I. qu'est ce que l'elearning	19
II. Principes du e-learning	20
III. formes du e-Learning	20
III.1 Formation exclusivement en ligne (sans tutorat)	20
III.2 Formation exclusivement en ligne (avec tutorat).....	20
III.2.1 Asynchrone.....	21
III.2 .2 Synchrone.....	21
III.2 .3 Mixte	21
III.2 .4 La Visio-formation	22
IV. Composantes d'un dispositif e-learning	23
IV.1 Les acteurs du dispositif.....	23
IV.2 Plates-formes d'apprentissage:	24

IV.2.1 Définition	24
IV.2.2 Les critères que doivent vérifier les plates-formes	25
IV.2.3 Outils de communication :	26
IV.2.4 Les fonctionnalités d'une plate-forme de E-Learning	28
IV.2.5 Exemples de plateformes d'e-learning.....	28
IV.3 Les contenus.....	29
IV.4 Stratégie pédagogique et tutorale	30
V. gestion de contenus d'apprentissage	31
V.1 Qu'est ce qu'un CMS	32
V.2 Fonctionnalités des CMS.....	33
V.3 Principe de fonctionnement d'un site web dynamique utilisant le CMS	33
VI. Normes et Standards pour la réutilisation des contenus pédagogiques élaborés	34
VI.1 objets pédagogiques et métadonnées	34
VI.1.1 Objets pédagogiques	34
VI.1.2 Méta-données	34
VI.2 Enjeux de la normalisation.....	35
VI.3 Les principaux standards.....	35
VII. Avantages et limites du e-learning	36
Conclusion.....	38

CHAPITRE III : WEB SERVICES & WEB SEMANTIQUE

Introduction	39
I. l'architecture orientée service (SOA)	39
I.1 Qu'est ce que SOA (Service Oriented Architecture).....	40
I.2 qu'est ce qu'un service	40
I.3 Les objectifs de l'architecture orientée services	41
I.4 Principe de l'architecture SOA	41
II. les web services : une instance de SOA	43
II.1 Qu'est ce qu'un web service.....	43
II.2 Les origines du web service	44
II.3 Caractéristiques technologiques du web service	45
II.4 Technologies des web services.....	46
II.4.1 XML (eXtensible Markup Language)	46
II.4.2 Le protocole SOAP (Simple Object Access Protocol)	47

II.4.3 Le langage WSDL (Web Service Description Language).....	49
II.4.4 Le protocole UDDI (Universal Description, Discovery and Integration)	51
II.5 Architecture des web services	52
II.5.1 L'architecture de base.....	52
II.5.2 L'architecture étendue	53
III. Le web sémantique.....	55
III.1 Objectifs du web sémantique	55
III.2 Les spécifications du Web Sémantique.....	56
III.2.1 Ontologie.....	57
III.2.2 RDF (Resource Description Framework)	57
III.2.3 OWL (Web Ontology Language).....	58
IV. Web services sémantique	59
IV.1 Approches Proposées Pour Les Services Web Sémantiques	61
IV.1.1 WSDL-S (Web Service Description Language-Semantic).....	61
IV.1.2 OWL-S (Ontology Web Language for Services).....	61
IV.1.3 DAML-S (DARPA Agent Markup Language for Services).....	61
Conclusion.....	62

CHAPITRE IV : ANALYSE ET CONCEPTION

Introduction	63
I. Analyse	63
I.1 Les problèmes rencontrés dans les systèmes de gestion de contenus (CMS).....	63
I.2 Présentation et objectifs de l'application	64
I.3 Les cas d'utilisation	65
I.3.1 Identification des acteurs et des fonctionnalités attendues	65
I.3.2 Identification des scénarios.....	67
II. Conception.....	71
II.1 La démarche d'élaboration du projet.....	71
II.2 Diagrammes représentatifs	72
II.2.1 Diagramme de contexte de l'application	72
II.2.2 Diagrammes de cas d'utilisation.....	73
II.2.3 Diagrammes de séquence	75
II.2.4 Diagramme de classe	80

III. Publication et découverte des services web	82
III.1 Services Web et e-Learning	82
III.2 Approche de Découverte de Services Web Liés aux Ressources Pédagogiques	83
III.2.1 Ontologie des services web OWL-S	83
III.2.2 Ontologie Associée à la Pédagogie Liée aux Ressources Pédagogiques	84
III.2.3 Ontologie du domaine à enseigner	85
III.2.4 Fonctionnalités de l'Infrastructure de Découverte de Services Web Liés aux Ressources Pédagogiques.....	86
Conclusion.....	88

CHAPITRE V : REALISATION

Introduction	89
I. Environnement de développement.....	89
I.1 Serveur Web Apache :	90
I.2 Serveur de base de données MySQL	90
I.3 Interface PHPMyAdmin	90
I.4 Le système d'exploitation Windows7.....	91
II. Les langages de programmation	91
II.1 PHP5 et son API intégré SOAP.....	91
II.2 Langage HTML	93
II.3 Le langage de requêtes « SQL (Structured Query Language)».....	93
II.4 JavaScript.....	93
II.5 CSS (Cascading Style Sheets)	94
II.6 XML (eXtensible Markup Language)	94
II.7 Ajax (Asynchronous JavaScript And XML)	95
III. Les outils de développement	96
III.1 Wamp server (Windows Apache MySQL, PHP5).....	96
III.2 Dreamweaver® 8.....	97
III.3 Zendstudio	98
IV. Présentation de quelques interfaces	100
IV.1 La page d'accueil de la plateforme	100
IV.2 La page des préinscriptions.....	100
IV.3 Page d'authentification	101

IV.4 L'interface d'édition de ressources pédagogiques (espace auteur) contenant les technologies du web 2.0	101
IV.5 page d'ajout de ressources pédagogique « cours »	102
IV.6 Page de consultation des ressources afin de les modifier ou supprimer « cours »	102
IV.7 Page « espace personnel de l'apprenant »	103
IV.8 Page « espace personnel du formateur »	103
IV.9 Page « espace personnel de l'administrateur »	104
Conclusion.....	104

Conclusion générale	105
----------------------------------	------------

liste des figures

Liste des figures

<i>Figure I.1 : Du Web 1.0 au Web 2.0</i>	<i>4</i>
<i>Figure I.2 : Schéma comparatif des deux approches</i>	<i>9</i>
<i>Figure II.1 : Fonctionnement d'un CMS</i>	<i>33</i>
<i>Figure III.1 : Les principes d'une SOA (Architecture applicative).....</i>	<i>42</i>
<i>Figure III.2: Structure d'un message SOAP</i>	<i>48</i>
<i>Figure III.3 : Structure d'un document WSDL</i>	<i>49</i>
<i>Figure III.4 : Architecture de référence des services web</i>	<i>53</i>
<i>Figure III.5 : Pile des web services</i>	<i>54</i>
<i>Figure III.6 : Pile du web service sémantique</i>	<i>55</i>
<i>Figure III.7 : Couches du Web Sémantique selon le W3C</i>	<i>57</i>
<i>Figure III.8 : Exemple de graphe RDF</i>	<i>58</i>
<i>Figure III.9 : Les services web sémantique.....</i>	<i>60</i>
<i>Figure IV.1 : Méthodologie de modélisation.....</i>	<i>72</i>
<i>Figure IV.2 :Diagramme de contexte.....</i>	<i>72</i>
<i>Figure IV.3 : Diagramme des cas d'utilisation « Administrateur ».....</i>	<i>73</i>
<i>Figure IV.4 : Diagramme des cas d'utilisation « Auteur»</i>	<i>74</i>
<i>Figure IV.5 : Diagramme de cas d'utilisation« Apprenant»</i>	<i>74</i>
<i>Figure IV.6: Diagramme des cas d'utilisation « formateur ».....</i>	<i>75</i>
<i>Figure IV.7: Diagramme de séquence « créer contenu pédagogique »</i>	<i>76</i>
<i>Figure IV.8: Diagramme de séquence « consulter contenu pédagogique ».....</i>	<i>77</i>
<i>Figure IV.9 : Diagramme de séquence « supprimer une ressource pédagogique</i>	<i>78</i>
<i>Figure IV.10 : Diagramme de séquence « modifier une ressource pédagogique ».....</i>	<i>79</i>
<i>Figure IV.11 : Diagramme de séquence « mise à jour du flux rss »</i>	<i>80</i>
<i>Figure IV.12 : Diagramme de classe «auteur».....</i>	<i>81</i>
<i>Figure IV.13:Les classe principale de l'ontologie OWL-S.....</i>	<i>83</i>
<i>Figure IV.14: Ontologie du domaine à enseigner</i>	<i>85</i>
<i>Figure IV.15:Architecture générale de découverte de service web</i>	<i>87</i>
<i>Figure V.1 : Diagramme de déploiement</i>	<i>89</i>
<i>Figure V.2 : Administration des bases de données à partir de PhpMyAdmin</i>	<i>91</i>
<i>Figure V.3 : Interface principale de WampServer.....</i>	<i>97</i>

Figure V.4 : Espace de travail de Dreamweave	98
Figure V.5 : Interface ZendStudio pour la génération des WSDL	99
Figure V.6 : page d'accueil de la plateforme	100
Figure V.7: page des préinscriptions	100
Figure V.8 : page d'authentification	101
Figure V.9: Interface d'édition	101
Figure V.10: Interface de publication des cours	102
Figure V.11 : Page de consultation des cours	102
Figure V.12 :Page « Espace apprenant»	102
Figure V.13 : Page « Espace formateur»	103
Figure V.14 : Page « Espace administrateur»	104

Introduction générale

Introduction générale

En quelques années, internet a révolutionné le monde de la communication et a touché un public de plus en plus nombreux et satisfait des besoins de plus en plus variés, de la mise en réseau de la recherche, à l'apprentissage à distance en passant par les services en ligne ou la communication et le partage de fichiers, photos, documents entre particuliers.

Le service qui a rendu l'Internet populaire auprès du grand public est le "World Wide Web" (toile d'araignée mondiale, abrégée en web) qui a commencé à se répandre en 1993.

Le web 2.0 est une évolution du web vers plus de simplicité et d'interactivité, c'est un support de communication très efficace qui touche la plus part d'individus dans le monde entier grâce à l'emploi d'ensemble d'applications qui font du web une sorte de système d'exploitation à part entière, et où les internautes deviennent des co-auteurs ou des Co-développeurs. Donc il est considéré comme un Phase du développement d'internet où les décideurs ont découvert qu'une fraction conséquente et croissante de la population s'en sert effectivement dans leur vie quotidienne.

L'avènement du concept de web sémantique a amélioré la manière de faire ses recherches sur Internet. Le web sémantique (plus techniquement appelé « Web de données ») permet aux machines de comprendre la sémantique, la signification de l'information sur le Web. Il permet ainsi aux agents automatisés d'accéder plus intelligemment aux différentes sources de données contenues sur le Web et de cette manière, d'effectuer des tâches (recherche, apprentissage, etc.) plus précises pour les utilisateurs.

L'objectif de notre travail consiste à intégrer les technologies du web 2.0 pour l'extension d'un CMS tout en exploitant l'architecture orienté service (SOA)

Pour mener à terme notre travail et par souci d'efficacité, nous avons répartis notre mémoire en cinq chapitres :

- Le premier chapitre concerne l'état de l'art sur le web 2.0
- Le second est consacré à l'e-learning ainsi qu'aux systèmes de gestion des contenus d'apprentissage
- Le chapitre qui suit traite les services web et le web sémantique
- Le chapitre suivant est consacré à l'analyse et à la conception de l'application
- Le dernier se charge de la partie réalisation.
- Ce rapport est suivi d'une conclusion générale.

Chapitre 1

Les technologies du web2.0

Introduction

L'internet est le plus grand réseau du monde fait appel à des efforts commerciaux, académiques et gouvernementaux

Le web ou World Wide Web est l'un des services majeurs de l'internet. Il se compose de serveurs web connectés à internet et stockant des pages web. Celles-ci sont des documents multimedia qui contiennent du texte, des images, des animations et des videos.

Le **Web 2.0** est une évolution du Web vers plus de simplicité et d'interactivité. Les internautes deviennent des co-auteurs ou des Co-développeurs, ils contribuent à l'échange d'informations et peuvent interagir (partager, échanger, etc.) de façon simple, à la fois avec le contenu et la structure des pages, mais aussi entre eux, créant ainsi notamment le Web social. L'internaute devient, grâce aux outils mis à sa disposition, une personne active sur la toile.

Le web2.0 n'est pas considéré comme une révolution technique mais comme une phase de développement d'internet d'un espace où les rares personnes qui y avaient accès mettaient en ligne des pages statiques qui sont rarement mise à jour où l'internaute n'est qu'un spectateur extérieur à l'information à un Internet plus interactif et participatif qui met l'individu au cœur du système avec des pages web qui changent chaque jour

I. Le web et son évolution

I.1 Définition

Le World Wide Web (toile d'araignée mondiale) est un système d'information hypermédia sur Internet. Il est fondé sur l'extension de la technique de l'hypertexte aux réseaux et a crée le premier réseau hypermédia répartie de couverture mondiale, donnant ainsi la possibilité d'accès a de très grandes quantités et variétés de document de façon très simple. Le Web repose sur une architecture Client/serveur distribuée et une interface de navigation offrant la puissance d'hypertexte en environnement graphique.

I.2 évolution du web

Au fil des années le web s'est modernisé et est passé par diverses versions parmi lesquelles on trouve :

- **Web 0.0** : Expression ironique désignant les phases de développement précédant l'existence réelle du Web, le fait que certaines personnes ne disposent pas d'internet ou un effet d'annonce sans aucun contenu.

- **Web 1.0** : Web statique (les sites web sont une collection de pages web statiques rarement mises à jour)
- **Web 1.5** : Web dynamique où des systèmes de gestion de contenu servaient des pages web dynamiques, créées à la volée à partir d'une base de données en constant changement
- **Web 2.0** : Web participatif, social et intelligence collective. Concept proposé par Tim O'Reilly en 2005.
- **Web 3.0** : Expression désignant la prochaine évolution majeure du Web. Attendue comme étant le Web sémantique; d'autres pensent que ce sera le Web3D. Finalement en 2011 le Web.3.0 désigne également le Web par P2P d'un ordinateur à l'autre
- **Web 4.0** : Pour Nova Spivack, patron de Radar Networks, désigne le WebOS, la possibilité de travailler avec des outils uniquement en ligne. Pour Joël de Rosnay ou Seth Godin, désigne le Web symbiotique, utilisé en permanence ; sans contester la pertinence de ce découpage

II. Le web 2.0

II.1 Évolution du web 1.0 vers le web 2.0

Le web 1.0 dans sa conception initiale comprenait des pages web statiques qui étaient rarement mises à jour, voir jamais. Une première évolution fut réalisée par des solutions se basant sur un web dynamique (parfois appelé web 1.5), où des systèmes de gestion de contenu(CMS) servaient des pages web dynamiques, créées à la volée à partir d'une base de données en constant changement. Le web était considéré principalement comme un outil de diffusion et de visualisation de données, où des aspects comme le nombre de pages vues et l'esthétique revêtaient une très grande importance.

En 1997, 1,1 million de personnes à peine étaient alors reliées à Internet en France, le plus souvent à l'aide de modems 56k au grésillement caractéristiques. Et le web était essentiellement composé de stocks de pages html statiques, peu ou pas mises à jour, auxquelles on accédait à l'aide d'annuaires les recensant par catégories.

L'explosion de la bulle internet en 2001 a définitivement marqué un tournant dans l'histoire du web. Beaucoup de gens ont à ce moment considéré que le web était une technologie surévaluée alors qu'au contraire, le fait qu'une bulle se forme puis éclate est un trait commun à toutes les révolutions industrielles. Ces soubresauts ont même caractéristiques

du moment où une technologie ascendante est prête à entrer dans une nouvelle phase. C'est en effet le moment où les simples prétendants arrivent à bout de souffle tandis que les points forts des premiers gagnants apparaissent : et c'est seulement à cet instant que l'on commence à comprendre ce qui distingue les premiers des seconds. [1]

Le concept de web 2.0 est apparu avec une conférence « brainstorming » entre O'Reilly et Medialive International. Dale Dougherty, pionnier du web et membre d'O'Reilly notait alors que bien loin de s'être effondré, le web n'avait jamais été aussi important et que nombre de nouveaux sites et applications à caractère innovant apparaissaient avec une régularité déconcertante. De plus, les quelques sociétés qui avaient survécu à l'hécatombe semblaient avoir quelque chose de commun.

La figure ci-dessous montre l'évolution du web 1.0 vers le web 2.0

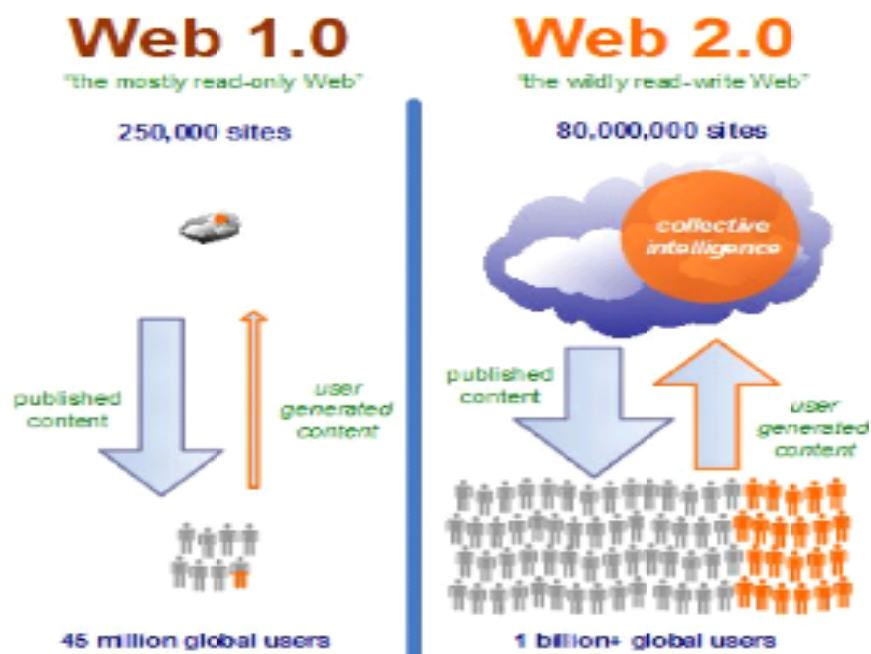


Figure I.1 : Du Web 1.0 au Web 2.0

L'évolution du web permettant de travailler exclusivement avec des applications en ligne et reposant sur des technologies fonctionnelles, rend le web2.0 accessible à des domaines variés (surtout commercial) tel que l'e-learning, la e-administration, la publicité...

II.2 qu'est ce que le web 2.0

Le Web 2.0 est un concept flou qui regroupe un certain nombre de tendances et chacun semble en avoir sa propre définition :

Anicet Mbida¹ « le web 2.0 n'est plus une collection de sites, mais une sorte de système d'exploitation pour applications Web. A terme, les services de Web 2.0 pourraient remplacer les applications bureautiques. »

Claude leblanc² introduit le concept de la manière suivante :

« Les blogs, les sites de réseaux sociaux ou encore les Wiki ont permis au citoyen lambda de passer du statut de simple récepteur à celui d'émetteur-récepteur, l'encourageant ainsi à s'investir dans la société. »

Guy Hervier³ restitue bien la confusion et la multiplicité des définitions qui sont associées à ce terme web 2.0 : "Le Web 2.0 est une sorte d'auberge espagnole où chacun peut apporter quelque chose ou trouver ce qu'il souhaite. Concept technologique pour les uns, évolution fonctionnelle du Web pour les autres, vaporware marketing pour d'autres encore..."

Wikipedia «L'expression**Web2.0** a été proposée pour désigner ce qui est perçu comme un renouveau du World Wide Web. L'évolution décrite est aussi bien technologique que d'utilisation. En particulier, on qualifie deWeb2.0 les interfaces permettant aux internautes d'interagir avec le contenu des pages, et entre eux.» Wikipédia-2008

De manière plus générale, on peut définir le web 2.0 comme une étape de l'évolution du web dont l'utilisateur et le partage d'information sont la clé de voute. Ce web est notamment caractérisé par l'apparition de nouveaux services multi-supports (ordinateur, pda, téléphone) favorisant l'interaction entre les internautes (blogs, wikis, les réseaux sociaux, le partage de vidéos...etc), et les informations (flux rss, tags...etc) grâce aux technologies Ajax (du javascript amélioré permettant de changer du contenu dynamique sans recharger la page web), xml et surtout à une meilleure maîtrise des technologies de web dynamique.

Le web 2.0 est donc à la fois l'émergence des technologies Web dites participatives, offrant la possibilité aux internautes de devenir acteurs du réseau.

De ce fait on peut résumer le "concept" Web 2.0 en deux aspects : l'un qui est technique et l'autre qui est plus lié à la communication et le partage.

- Au niveau technique le Web 2.0 correspond à ce que l'on peut appeler les interfaces enrichies (Ajax, Flash, Ruby on Rails...) Tous ces outils techniques, on pour but de faciliter la navigation et l'ergonomie.

- Au niveau communication cela correspond au partage et à la diffusion d'information (blog, RSS, utilisation d'API, portail communautaire, forum, folksonomie, wiki...) mais aussi au marketing viral.

II.3 Principes du web2.0

Le web vu comme une plate-forme de services :

On passe d'une collection de sites web à une plateforme informatique à part entière, fournissant des applications web aux utilisateurs.

Considérer les internautes comme co-développeurs des applications :

On passe ainsi de la notion de « logiciel produit » à celle de « logiciel service ».

Le service s'améliore quand le nombre d'utilisateurs augmente :

Le web 2.0 met à profit l'effet de la « longue traîne »

La richesse est dans les données :

O'Reilly envisage un mouvement « des données libres » s'opposer peu à peu à l'univers des données propriétaires.

Tirer parti de l'intelligence collective :

« l'implication des utilisateurs dans le réseau est le facteur-clé pour la suprématie sur le marché ».

Mettre en place des interfaces souples et légères fondées sur les nouveaux standards et protocoles du Web.

Le logiciel se libère du PC et va vers les objets nomades...[2]

II.4 Le panorama des techniques et technologies utilisées par le web 2.0

L'émergence du « user-generated » content (technologies participatives), ou l'internaute devient **émetteur-récepteur** et véritable acteur du réseau s'explique par les évolutions récentes des technologies Web dites **participatives**.

Pour implémenter ces nouvelles technologies les développeurs et concepteurs multimédias ont à leur disposition plusieurs **nouveaux langages de programmation**, principalement l'**AJAX**, le **XML** et le **Ruby**. Ces langages sont utilisés par les machines pour communiquer entre elles et de **nouvelles architectures** sont mises en place comme **REST** et **SOA** qui utilise le protocole **SOAP**. Ces architectures permettent aussi l'émergence de concepts jusqu'ici inconnus comme les **outils de travail collaboratif** tels que les **flux RSS**, les **atoms**, les **tags** et **Folksonomie** ... et les nouvelles applications telles que les **applications Internet riches** et les **mashups**. De plus, il faut également préciser qu'en plus du fond, les

sites Internet ont évolué sur la forme. Le Web2.0 a ainsi vu l'émergence de **nouvelles normes graphiques** pour les nouveaux sites Internet.

II.4.1 Les nouveaux langages

- **AJAX** (Asynchronous Javascript And XML)

Le terme « AJAX » est apparu pour la première fois dans un article de Jesse James Garret, sur le site de sa société Adaptive Path, le 18 février 2005.

L'article s'intitule *ajax : A New Approach to web Applications*. Il insiste sur le fait qu'Ajax n'est pas une technologie en tant que telle, mais la conjonction de technologies existantes pour une utilisation combinée novatrice.

L'idée de base : la plupart des applications web, c'est-à-dire des applications dont l'interface graphique est affichée dans un navigateur, offrent une interaction pauvre et un confort d'utilisation plutôt restreint, en particulier si on les compare aux applications classiques, installées sur les postes utilisateurs. Dans une application web, on reste le plus souvent positionner du carcan requête/réponse : pour interagir avec l'application, qu'il s'agisse de valider la saisie de données ou de réorganiser les éléments d'une liste, il faut avancer à petits pas, avec à chaque étape un aller-retour entre notre navigateur et le serveur, qui engendre un rechargement complet de la page.

Par ailleurs, les possibilités offertes à l'utilisateur pour exprimer une demande ou réaliser une action restent primaires : il est rare de voir un site proposer d'ajouter un produit au panier simplement en glissant-déplaçant le premier sur le second. Ces limitations habituelles auraient très bien pu être levées sans Ajax, comme ce fut d'ailleurs le cas sur une minorité de sites. Toutefois, en bouleversant notre conception de ce qu'il était possible de proposer comme interactions sur une page web, Ajax a naturellement remis sur le devant de la scène la question des interactions riches, notamment au travers du glisser-déplacer et des effets visuels. [3]

Ajax repose sur les technologies suivantes :

- ✓ XHTML pour assurer un balisage sémantique et cohérent du document, ce qui assure que celui-ci sera correctement représenté en mémoire et donc facilement manipulable
- ✓ CSS pour habiller ce balisage sémantique et élargir la gamme des effets visuels utilisables pour communiquer avec l'utilisateur
- ✓ DOM pour représenter le document en mémoire, afin d'en manipuler la structure et le contenu sans avoir, justement, à recharger toute la page.
- ✓ XML et, dans une moindre mesure, XSL/T, pour structurer les données échangées en coulisses entre la page déjà chargée et le serveur, et transformer si nécessaire ces données en contenu affichable.
- ✓ XMLHttpRequest, service originellement fourni par Microsoft dans MSIE depuis la version 5, jusqu'alors méconnu du grand public malgré un intérêt technique qui avait conduit à son implémentation ultérieure par les autres navigateurs. Le premier A de Ajax c'est lui : le moyen de communication asynchrone.
- ✓ JavaScript qui relie tout ces éléments entre eux.

Là où une page web classique nécessite un rechargement à chaque action, une page exploitant Ajax interagit avec le serveur en coulisses, sans se recharger entièrement. Un code JavaScript réagit à l'action de l'utilisateur en gérant un aller-retour interne avec le serveur, et met éventuellement à jour la page directement en manipulant le DOM de celle-ci. Le schéma suivant compare ces deux approches :

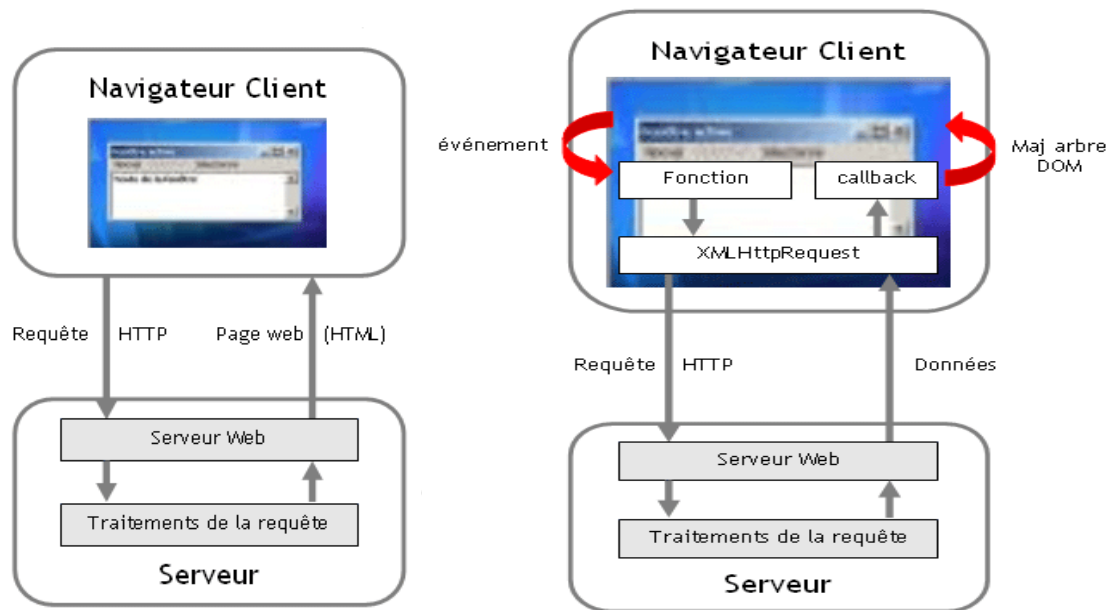


Figure I.2 : Schéma comparatif des deux approches

- **XML** (Extensible Markup Language)

Le langage XML (eXtended Markup Language) est un format général de documents orienté texte. Il s'est imposé comme un standard incontournable de l'informatique. Il est aussi bien utilisé pour le stockage de document que pour la transmission de données entre applications. Sa simplicité, sa flexibilité et ses possibilités d'extension ont permis de l'adapter à de multiples domaines allant des données géographiques au dessin vectoriel en passant par les échanges commerciaux. De nombreuses technologies se sont développées autour de XML et enrichissent ainsi son environnement.

Le langage XML dérive de SGML (Standard Generalized Markup Language) et de HTML (HyperText Markup Language). Comme ces derniers, il s'agit d'un langage orienté texte et formé de *balises* qui permettent d'organiser les données de manière structurée. [4]

XML impose et permet une séparation stricte entre :

- La structure : la définition de la structure d'un document,
- Le contenu : la description du contenu d'une occurrence du document, et
- La présentation : la « mise en page » du document. Les informations de présentation sont contenues dans un document séparé, formaté avec XSL (Extensible Stylesheet

Language) ou avec un document CSS (Cascading Style Sheet), qui sont référencés pour afficher le document de contenu.

Cette séparation apporte une indépendance du document par rapport aux évolutions technologiques matérielles et logicielles et ainsi une pérennité d'utilisation à court et moyen termes. C'est cette caractéristique de séparation entre structure, contenu et présentation qui va favoriser l'utilisation de XML dans l'échange d'informations entre applications informatiques.

- **Ruby on Rails**

Ruby est un langage de script orienté objet. Il a été conçu pour le web. L'une de ses principales caractéristiques est d'être intégralement orientée objet. Il s'agit d'un langage serveur, comme PHP, qui permet de créer des interfaces très facilement et très rapidement, au contraire de PHP qui nécessite de longues heures de développements. Parfaitement adaptée aux exigences des nouveaux services qui doivent bénéficier d'une grande réactivité, son essor est grandissant dans la communauté. Malgré ses nombreuses qualités, il manquait à ce langage un environnement de travail, un Framework puissant, exploitant ses nombreuses possibilités. Ruby on rails s'est présenté comme le candidat idéal. Il en résulte des sites qui proposent des interfaces riches en fonctionnalités, pourvues d'une forte interactivité, illustrant bien le concept web 2.0. [1]

- **CSS (Cascading Styles Sheets)]**

Le langage CSS (Cascading Style Sheets) adresse les problématiques d'affichage des pages HTML. Son objectif est de permettre la séparation claire entre la structure d'une page, par le biais du langage HTML, et sa présentation avec CSS. Fondé sur l'application en cascade de styles sur les éléments contenus dans les pages HTML, il permet d'appliquer des styles en fonction de la position des balises. [5]

- **XSL(eXtending Stylesheet Language)**

Un fichier XSL est une feuille de style CSS améliorée. Ce format permet la création de templates (modèles de pages) qui sont utilisées avec les données d'un fichier XML permettent l'affichage d'une page web complète. Les feuilles de styles XSL étendent l'effet des CSS à la création de calques ou de tableaux contenant un certain type d'information. De plus les développeurs peuvent programmer sommairement certains aspects, grâce à une syntaxe

fonctionnelle, permettant de créer des boucles afin d'automatiser certaines mises en page. On aura ainsi un fichier XML contenant les articles d'un journal, une feuille de style XSL positionnant cet article dans la page, et une feuille de style CSS gérant l'aspect du texte. [1]

II.4.2 Nouvelles architectures réseaux

- **REST** (REpresentational State Transfer)

REST est une technologie développée par Roy T. Fielding en 2000. C'est un modèle définissant l'architecture du web, bâtie sur quatre principes fondamentaux. Connaître l'URI (l'URI est l'identifiant de ressource Internet, généralement une adresse web) doit suffire pour accéder à l'information, le protocole HTTP doit être à même de fournir toutes les informations, chaque opération doit être auto suffisante et les standards hypermédias doivent être utilisés. Cette architecture n'est pas limitée aux sites Internet, elle est de plus en plus utilisée par les développeurs pour faire communiquer des machines entre elles. En effet, elle reste simple, compréhensive par des systèmes très différents. L'information de base, dans une architecture REST, est appelée ressource. Toute information qui peut être nommée est une ressource : un article d'un journal, une photo, un service ou n'importe quel concept. Dans un système hypermédia, une ressource est tout ce qui peut être référencé par un lien. Enfin, les composants de l'architecture manipulent ces ressources en transférant des représentations de ces ressources. Sur le web, on trouve aujourd'hui le plus souvent des représentations au format HTML ou XML. Par exemple, les services d'Amazon et de Flickr sont bâtis sur des architectures REST. [1]

- **SOA (Service Oriented Architecture)**

Une **architecture orientée services** (notée SOA pour *Services Oriented Architecture*) est une architecture logicielle s'appuyant sur un ensemble de services simples. L'objectif d'une architecture orientée services est donc de décomposer une fonctionnalité en un ensemble de fonctions basiques, appelées **services**, fournies par des composants et de décrire finement le schéma d'interaction entre ces services.

II.4.3 Nouvelles applications

- **Les RIA** (*Rich Internet Application*)

Rich Internet Applications (RIAs) sont des applications web qui offrent des caractéristiques similaires aux logiciels traditionnels installés sur un ordinateur. La dimension interactive et la vitesse d'exécution sont particulièrement soignées dans ces applications web

- **API (Application Program Interface)**

Les API sont des interfaces publiques qui permettent à des applications ou sites web de communiquer (intuitivement : envoyer, réceptionner des informations, extraire des données, ...) avec d'autres applications. Elle constitue une des bases techniques de la mise en place des web services.

L'API décrit les objets/fonctions exposés par le site web de façon à ce qu'un autre site web puisse ajouter, modifier, supprimer des données soit en appelant une méthode, soit en échangeant un fichier structuré comme le fichier XML

- **Les Mash-ups ou applications Web composites**

Les applications Web 2.0 s'appuient sur des technologies bien maîtrisées telles que Javascript, Xml, RSS, CSS, XHTML... mais qui évoluent dans leurs usages et qui se massifient. Le web se veut plus accessible et plus simple d'utilisation pour les usagers et les applications s'exécutent dès lors plutôt dans le navigateur internet que suite à une installation locale, tout en privilégiant interactivité et vitesse d'exécution.

Un des objectifs est de mieux faire communiquer les sites entre eux. Ainsi dans le spectre des nouveaux termes utilisés à propos du web 2.0 trouve-t-on celui de **Mash-Up (mixage ou applications composites)**, un outil permettant d'utiliser du contenu provenant de différents sites web pour alimenter son propre site. Les Mash-Up offrent la possibilité de créer de nouveaux services en mutualisant des données rendues disponibles librement, à travers des API (i.e. Application Programming Interface, ou web services), par les éditeurs eux-mêmes. [6]

- **Les web services**

Selon W3C (World Wide Web Consortium) « Un Web Service est un composant logiciel identifié par une URI, dont les interfaces publiques sont définies et appelées en XML. Sa définition peut être découverte par d'autres systèmes logiciels. Les services Web peuvent interagir entre eux d'une manière prescrite par leurs définitions, en utilisant des messages XML portés par les protocoles Internet »

II.4.4 Les outils collaboratifs

- **Les RSS (Really Simple Syndication)**

La syndication de contenu désigne la possibilité de republier sur un site Web de façon automatisée du contenu provenant d'un autre site Web. C'est un format d'échange de données, défini en XML, permettant de décrire un contenu Web, et plus généralement toute page qui présente un contenu mis à jour chronologiquement, à l'aide de balises spécifiques. Le format RSS est utilisé pour obtenir les mises à jour (texte, son, images, vidéo,...etc) d'un site web ou d'un blog au format XML. Un flux RSS contient le titre de l'information, sa brève description et un lien pour la consulter. Concrètement, ce sont des flux de données gratuits mis à la disposition par les sites. Ils incluent les titres des derniers articles publiés, leurs résumés et des liens vers les versions intégrales de ceux-ci, sur le site d'origine. Ces flux sont remis à jour à chaque fois qu'une nouvelle donnée est publiée sur le site. Ainsi, les dernières informations publiées sur le site du monde peuvent ainsi venir automatiquement compléter vos sources d'informations déjà agrégées via un logiciel de lecture de flux RSS

- **Del.icio.us**

Créé par Joshua Schachter, permet de stocker vos favoris (bookmarks, signets) sur un site web. Idéal pour garder la trace des choses intéressantes trouvées sur le net, mais aussi pour découvrir des sites web.

A la manière de flickr, où l'on étiquette des photos, la valeur ajoutée de Del.icio.us réside dans l'ajout de tags sur les signets

- **Tag et nuage de tags**

L'une des principales innovations du web 2.0 reste le développement de système de gestion de tags (étiquettes ou label) pour décrire les contenus présents sur un site. Un nuage de tags (ou tag cloud) permet d'avoir un aperçu des termes les plus représentés dans les articles, sous forme de mots clés (tags). Plus un mot est représenté avec une taille importante, plus il y'a d'articles étiquetés avec ce mot. Cet affichage des tags selon une taille de caractères proportionnée au poids d'usage du tag sur le site vous permet de repérer du premier coup d'œil les sujets traités les plus importants sur le site web (ou le blog, ou le réseau social...). Un nuage de tags compte ainsi en moyenne de 30 à 150 tags.

Leur utilisation a été popularisée par des sites de partage de photos comme Flickr, puisque c'est le co-fondateur de Flickr, Stewart Butterfield, qui les y a implémentés. Cela permet par exemple de voir sur un blog quels sont les mots clés le plus souvent cités dans les billets, et donc quels sont les thèmes les plus abordés. [7]

L'apparition des nuages de tags a apporté de nouvelles possibilités au lecteur pour naviguer (et fonctionner) par association d'idée.

Cette classification est appelée folksonomie (de l'anglais folksonomy, combinaison des mots folk le peuple, les gents et de taxonomy la taxonomie).

- **Folksonomie**

Le terme a été créé par l'architecte de l'information Thomas Vander Wall. La particularité des folksonomies étant que contrairement aux systèmes hiérarchiques de classification, les contributeurs d'une folksonomie ne sont pas contraints par une terminologie prédéfinie, mais peuvent adopter les tags qu'ils veulent pour classifier les contenus qu'ils produisent et leurs ressources. Autre particularité, pour un blog, un site Digg-like ou une plate-forme de partage telle que Flickr, sa classification est l'union des classifications par les différents contributeurs, par le choix de leurs mots clés, qui contribuent à faire apparaître en plus gros dans le nuage de tags les mots clés les plus populaires.

- **Les wikis**

Un wiki est un outil de travail collaboratif. C'est un site web librement modifiable par ses visiteurs, sans difficulté technique, et qui permet la libre circulation de l'information. Wikipedia, encyclopédie en ligne, est l'une des meilleures réussites du wiki sur internet.

Le wiki a été inventé en 1995 par **Ward Cunningham**, pour une section d'un site sur la programmation informatique qu'il a appelé WikiWikiWeb. Ce nom provient du mot hawaïen wikiwiki qui signifie « vite ». En effet, un des avantages du wiki est que la publication d'une information y est instantanée : l'article ou le commentaire ajouté n'est pas relu et approuvé par un modérateur avant d'être accessible en ligne. Toutes les personnes autorisées à modifier le contenu ont les mêmes droits de modification et jouissent d'une liberté d'action, qui n'est limitée que par la nécessité de ne pas compromettre l'intégrité technique du site. [7]

- **Les blogs**

Le web 2.0 se caractérise par la possibilité qu'il offre aux internautes de produire des contenus en ligne ; la communauté d'internautes pouvant compléter et modifier ou commenter les articles mis en ligne. Les blogs et les wikis, qui reposent sur des interfaces très faciles à utiliser, sont les deux outils symbolisant le mieux cette « prise de pouvoir » de l'internaute. Les médias participatifs sont aussi en train de s'affirmer.

Le terme de blog a été formé par l'association de deux termes, *web* (la toile) et *log* (journal). C'est pourquoi on parle aussi de *web log* ou de *weblog* ; Il désigne un outil de publication personnelle sur le net. S'ils comportaient, au début, essentiellement du texte et de l'image, les blogs se sont enrichis, avec des vidéos (vidéoblogs ou vblogs) et des sons (podcasts).

- **Les réseaux sociaux**

Les réseaux sociaux sont sans doute une des manifestations les plus connues du web 2.0, puisqu'ils permettent aux internautes d'être des participants actifs de la Toile. Cet ensemble d'applications internet de social networking (« services de réseautage social en ligne ») aide à relier des internautes entre eux, même disséminés géographiquement.

Outils de réseautage professionnel, de recherche de camarades de classes, de famille éloignée, de nouveaux amis par affinités, de promotion et de distribution pour les artistes,..., les réseaux sociaux constituent une des petites révolutions propres à internet participatif.

Grâce à une kyrielle de plate-formes et d'outils en ligne, les internautes se sont peu à peu appropriés ces réseaux sociaux, en les adaptant à leurs usages. [7]

II.5 Les sites web 2.0

Un site pourrait être appelé comme utilisant une approche Web 2.0 s'il contient un Certain nombre des techniques suivantes :

- l'utilisation de CSS, d'un balisage XHTML sémantiquement valide, et des microformats .
- les techniques d'applications riches telles que AJAX .
- la syndication et l'agrégation de contenu RSS/Atom .
- la catégorisation par étiquetage.
- l'utilisation appropriée des URL
- une architecture REST ou des services Web.

Parmi les exemples les plus connu de sites web2.0 on peut citer :

✓ **Facebook**

Initialement appelé TheFacebook, Facebook est aujourd'hui le plus grand réseau social du monde (la plus grande communauté virtuel du monde). FaceBook a été fondé par un ancien étudiant de Harvard Mark Zuckerberg qui l'a exécuté le 7 février 2004 comme un de ses projets passe-temps. Au départ, simple albums photo ou trombinoscope en ligne (facebook en anglais) réservé aux étudiants de Harvard et à la longue, étendu à toutes les universités et lycées Américains, Canadiens puis Européen et Asiatique il a été officiellement appelé FaceBook En août 2005.

✓ **Myspace**

Le site de socialisation MySpace a été créé en 2003 par Tom Anderson et Chris DeWolfe et appartient à Intermix Media, société de marketing sur internet. Il s'est imposé comme le réseau social musical de référence.

✓ **Twitter**

Twitter est un outil gratuit de réseaux social, par lequel l'internaute écrit à son réseau, en temps réel, par messageries instantanée ou par messagerie numérique. Ces updates sont qualifiées de tweets (gazouillis en anglais). Twitter a été lancé en mars 2006 à San Francisco au sein de la start-up Odeo Inc. Par Noah Glas.

✓ **Youtube**

Youtube est un site internet d'hébergement de vidéos à partir duquel les internautes peuvent envoyer, visionner et se partager des vidéos. Il a été créé en février 2005, Chad Hurley, Steve Chen et Jawed Karim.

✓ **Dailymotion**

Dailymotion est un site web français permettant de partager et de visionner des vidéos diverses et variées. Il est semblable en ceci à Youtube, mais va plus loin dans l'utilisation du web 2.0, notamment dans la gestion des tags.

Dailymotion a été créé en février 2005 par Benjamin Bejbaum et Olivier Poitrey. Il utilise comme la plupart de ces outils la technologie flash, il se démarque par une interface multilingue et un moteur d'encodage développé en interne. [1]

II.6 Tendances et perspectives

En termes d'évolution du Web2.0, on peut imaginer plusieurs choses. Premièrement, l'avènement du concept de Web sémantique va révolutionner la manière de faire ses recherches sur Internet. Deuxièmement, on peut concevoir une évolution de la syndication des données en combinant données et services afin de permettre de développer encore plus le concept de plateforme client/serveur où n'importe quelle application devient accessible via Internet. Enfin, si le terme de Web 3.0 doit exister un jour, cela ne pourra être qu'en tant qu'une évolution du Web2.0 avec des services encore plus poussés, plus intuitifs et surement le principe des MashUps exploités au maximum avec des sites concentrant un ensemble de services.

Conclusion

Un site moderne n'est plus dorénavant une somme de pages, mais plutôt une combinaison de techniques. Un webmaster doit être un technicien pour un site innovant.

Le web 2.0 et tout son apanage technique et technologie apparaît logiquement comme novateur dans sa modification fondamentale de la communication sur Internet. Le serment du Web 2.0 est donc d'ouvrir une ère nouvelle où les utilisateurs sont à la fois les contributeurs et les bénéficiaires. Dès lors la perception de l'intelligence collective prend tout son sens. Un point de départ où une multitude de nouveaux usages et de nouvelles applications sont encore à trouver.

Il n'en demeure pas moins que cette notion conceptuelle qu'est le Web 2.0 reste floue à définir mais aussi à cerner... Pour le moins paradoxal en soit puisque tout le monde en parle mais pourtant personne n'en saurait en donner une définition exacte sur laquelle tout le monde s'accorderait. Il s'agit surtout d'un ensemble d'enjeux économiques, de nouveaux services définis pour mais aussi autour de l'internaute consommateur.

Chapitre II

E-learning & systèmes de gestion des contenus

Introduction

Avec l'avènement des nouvelles technologies de l'information et de la communication (NTIC), nous devons, dès à présent, « penser apprentissage rapide et efficace », avec un minimum de problèmes d'organisation, de logistique et surtout de perte de temps. L'e-learning est la solution, C'est le nom donné actuellement à une phase importante de l'introduction des nouvelles technologies de l'information et de la communication dans la formation. Il s'agit d'une évolution rapide des technologies pour l'apprentissage, rendue possible par le développement planétaire de l'Internet.

Le e-learning est un domaine révolutionnaire, un changement dans le champ de l'apprentissage. Les instructions qu'il offre en ligne peuvent être fournies **n'importe quand et n'importe où** par une gamme très vaste de solutions d'apprentissages électroniques telles que les groupes de discussions, les cours virtuels « en live », vidéo et audio, Web chat,...

Un CMS (Content Management System ou système de gestion de contenu) est le cœur du système de formation à distance, c'est lui qui fait le lien entre les apprenants, les cursus, les ressources et les contenus présents dans le système. Son but est de simplifier la création et la gestion du contenu en ligne. Il permet une meilleure fréquence des mises à jour des ressources déjà publiées et à moindre coût.

I. qu'est ce que l'e-learning

L'e-learning définit tout dispositif de formation qui utilise un réseau local, étendu ou Internet pour diffuser, interagir ou communiquer. Ce qui inclut l'enseignement à distance, en environnement distribué (autre que l'enseignement par correspondance classique), l'accès à des ressources par téléchargement ou en consultation sur le Net. Il peut faire intervenir des outils synchrones ou de l'asynchrones, des systèmes tutorés, des systèmes à base d'auto formation, ou une combinaison des éléments évoqués. [8]

Parmi la terminologie couramment utilisée et ayant une signification proche, on peut lister:

- ✓ Formation à distance ou F.A.D.
- ✓ Enseignement à Distance ou E.A.D.
- ✓ Formation Ouverte à Distance ou F.O.A.D.
- ✓ E-Formation

- ✓ Formation en ligne
- ✓ Etc. [9]

II. Principes du e-learning

- L'organisation du contenu pédagogique interactif et l'adaptabilité du cursus pédagogique au public.
- L'utilisation des nouvelles techniques pour remplacer en tout ou en partie la présence physique de l'enseignant.
- La modification du rôle de l'enseignant et l'apprenant. L'enseignant est un tuteur qui fait le suivi des apprenants, les conseille, les suit dans leurs progression, les oriente, les évolue et les informe.
- La constitution des groupes d'apprenant : l'apprenant organise lui-même le déroulement de son apprentissage et devient acteur de sa formation. Il suit les cours à sa convenance et son rythme.

III. formes du e-Learning

III.1 Formation exclusivement en ligne (sans tutorat)

Ce type de formation est proche du service fourni par un cédérom ou d'une vidéo de formation. Cette formation n'apporte pas beaucoup de valeur ajoutée. On la trouve souvent sur les «portails de formation» aux Etats-Unis. Elle tend à devenir gratuite, certains portails faisant payer uniquement le tutorat en ligne.

III.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 :

III.2.1 Asynchrone

Situation durant laquelle l'échange avec les autres apprenants ou avec les tuteurs s'effectue via des modes de communication ne nécessitant pas de connexion simultanée forums de discussion, e-mails. Ce mode de formation repose sur un apprentissage dit autodirigé, avec des cours, des exercices et des évaluations automatisés, impliquant une certaine autonomie de l'apprenant. [8]

III.2.2 Synchrone

La formation synchrone, au contraire de la formation asynchrone, se caractérise par l'interaction directe et en temps réel entre les apprenants et le formateur. C'est la méthode qui s'approche le plus de la classe traditionnelle. Lorsque tous les apprenants d'un groupe sont simultanément en ligne avec leur formateur, échangent entre eux au moyen du clavardage, partagent des applications, visionnent les mêmes écrans ou encore reçoivent des images de vidéoconférence (*Webcast*), ils pratiquent l'apprentissage virtuel synchrone. [10]

- **Principaux avantages de la formation synchrone :**

- ✓ les apprenants interagissent intensivement à l'écran avec le formateur. Le langage oral et visuel est utilisé ;
- ✓ le modèle de la classe est familier aux apprenants et aux formateurs ;
- ✓ la dynamique de groupe s'installe plus rapidement ;
- ✓ le contenu de formation peut être rapidement créé et diffusé.

III.2.3 Mixte

L'apprentissage mixte, on le devine, emprunte aux différentes méthodes d'apprentissage. Cette méthode de formation combine donc les éléments de l'apprentissage en ligne et de l'apprentissage en classe traditionnelle.

Voici quelques exemples de formules d'apprentissage mixte :

- avant de bénéficier d'un enseignement traditionnel en classe, les apprenants reçoivent du matériel ayant recours aux technologies de l'information (Internet, cédéroms). Le temps de rencontre est diminué par cette préparation et la formation en classe se concentre sur les échanges approfondis ;

- entre deux séances de formation en classe, les participants poursuivent leur formation au moyen d'échanges virtuels structurés et d'outils de collaboration ;
- une classe se réunit en temps réel (mode synchrone : vidéoconférence, clavardage) et se poursuit en différé (forum de discussion, échange de courriels, didacticiel) ;
- de l'aide (*coaching*) est apportée aux apprenants, de façon virtuelle ou sur les lieux de travail, afin de faciliter l'intégration des connaissances acquises à travers une formation virtuelle.

- **Principaux avantages de la formation mixte :**

- ✓ des études en milieu universitaire ont démontré que l'apprentissage était plus élevé avec la formation mixte qui répond davantage aux différents styles d'apprentissage des apprenants ;
- ✓ les classes virtuelles en temps réel permettent aux apprenants d'assister aux cours à partir de plusieurs sites de formation ;
- ✓ lorsque l'apprenant peut enregistrer le déroulement du cours et sauvegarder les échanges entre participants, ces derniers peuvent revoir à leur guise les aspects qu'ils souhaitent réviser, ce qui constitue un véritable gain ;

En combinant l'utilisation des logiciels de formation standardisés et des formules d'échanges en classe et de travaux, on peut réduire considérablement la préparation à la formation et offrir tout de même des activités de formation valables. [10]

III.2 .4 La Visio-formation

C'est la formule la plus aboutie et sans doute la plus efficace en matière d'e-Learning. L'apprenant bénéficie d'un parcours personnalisé, défini par un tuteur qui intervient en direct et en image pour communiquer avec l'apprenant.

- **Avantages :** interactivité, personnalisation et visualisation du tuteur.
- **Inconvénient :** il faut posséder le matériel technique nécessaire (webcam, casque, carte vidéo) et une bande passante suffisamment large. La personnalisation entraîne un coût plus important pour l'entreprise (un tuteur pour chaque apprenant).

IV. Composantes d'un dispositif e-learning

Un dispositif d'apprentissage à distance se compose principalement de :

- Acteurs du dispositif :
 - une communauté d'apprenants
 - tuteurs ou animateurs
 - formateur
 - administrateur
 - Auteur
- plate-forme d'apprentissage
- contenus (textuels ou multimédia didactiques)
- stratégie pédagogique et tutorale

IV.1 Les acteurs du dispositif

- **Communauté d'apprenants**

Les apprenants consultent en ligne, téléchargent et impriment les ressources pédagogiques déposées par l'enseignant, réalisent les exercices, s'auto-évaluent, se font corriger des travaux à l'aide du tutorat mis en ligne, réalisent des tâches en collaboratif avec d'autres pour déboucher sur des documents écrits réalisés en commun.

- **Le tuteur**

A pour mission de suivre et accompagner les apprenants au cours de la formation. Il les aide à progresser, les motive, suit leur avancée, les relance, les guide, répond à leurs questions,... Le tuteur joue un rôle moteur dans la formation. La qualité du suivi permet d'entretenir la motivation de l'apprenant et d'éviter qu'il abandonne sa formation en cours de route. Le rôle du tuteur peut donc être occupé par le formateur, mais aussi par d'autres personnes qui garderaient un lien avec l'auteur du cours afin d'avoir de plus amples explications si une question complexe apparaissait. Il peut également être assuré par les co-apprenants.

- **Le formateur**

Le formateur crée des scénarios pédagogiques typé et individualisés de son enseignement et incorpore des ressources pédagogiques multimédias pour aider l'étudiant à apprendre tout en suivant ses activités.

- **L'administrateur**

Joue un rôle de coordination entre les enseignants, les étudiants et la plateforme et assure une communication dynamique autour de la vie estudiantine tout en jouant avec l'informatique un rôle technique de maintenance et d'évolution du système.

- **Auteur**

Il établit un parcours de formation personnalisé et individualisé. Il construit, adapte et maintient un système d'apprentissage. Il a pour rôle aussi la présentation des informations, la gestion des médias donnant ces dernières, la clarification des contenus en réponse à des questions, l'analyse et l'évaluation des contenus des documents.

IV.2 Plates-formes d'apprentissage:

IV.2.1 Définition :

La plateforme de formation (LMS : Learning Management System) met à la disposition des apprenants un ensemble de cours et de ressources accessibles à travers un réseau local d'entreprise ou à partir d'internet, en utilisant un navigateur classique. [11]

On rencontre également l'appellation LCMS, pour *Learning Content Management Systems*. Il s'agit de plates-formes hybrides intégrant des fonctionnalités de gestion de contenu (CMS - *Content Management Systems*) et des fonctionnalités de support à l'e-learning. La principale différence entre ces deux types d'outil est que les LCMS intègrent des « outils auteurs » (ou outils d'édition, *authoring tools* en anglais), c'est-à-dire des outils permettant de créer et d'organiser du contenu pédagogique. Les LMS n'intègrent pas ce type d'outil et se limitent à la gestion du parcours des apprenants dans des modules de cours créés séparément.

En poussant à l'extrême la vision des LCMS, et l'intégration d'outils simples à utiliser pour créer les cours, on arrive à la notion récente du *rapid e-learning*. Ce concept est mis en œuvre dans certains logiciels qui permettent de créer des supports de cours destinés à être utilisés comme support à l'e-formation. Ces cours sont en général de courte durée et concernent le plus souvent des thèmes situés à la frontière entre la formation et l'information (par exemple: la mise en place d'un nouveau logiciel, l'explication d'une nouvelle procédure interne, etc.).

IV.2.2 Les critères que doivent vérifier les plates-formes

- Le coût total de la mise en place de la plate-forme (gratuite ou non), dans le cas d'une plate-forme payante, vérifier les prestations fournies.
- Les besoins de maintenance de la plate-forme d'un point de vue strictement informatique, tout ce qui devra être géré par les administrateurs (voir également le coût en personnel de maintenance).
- La sécurité de la plate-forme d'un point de vue purement informatique (filtrer les IPs qui auront accès aux cours) et au niveau des cours (login, mot de passe).
- Le nombre d'utilisateurs maximum que pourra supporter la plate-forme.
- La possibilité de mettre en place des QCM (Questions à Choix Multiples) sur la plateforme, s'il est possible d'insérer une image, si le système de questions est interactif (avec animations), s'il y a possibilité de faire un tirage aléatoire des questions.
- Un forum de discussion dont l'accès sera étroitement lié à la sécurité interne de la plateforme.
- La technologie utilisée, du point de vue du langage pour pouvoir éventuellement faire évoluer le système et sa capacité d'accueil (XML, MySQL, PHP...).
- Un système de gestion de mails, pour qu'un formateur puisse éventuellement envoyer des mails à l'une de ses classes, également lui permettre de gérer des groupes.
- La façon de poster un cours.
- Les moyens requis pour consulter la plate-forme (browser, système d'exploitation...).
- La possibilité de diffuser les cours par vidéo soit en temps réel, soit en différé. **[12]**

IV.2.3 Outils de communication

Les plates-formes intègrent un certain nombre d'outils de communication, la liste des outils disponibles dépend de la richesse de la plate forme. Ces outils sont souvent très usités sur le Web et ne sont en rien spécifique au e-Learning, ils font simplement partie intégrante de la plate forme 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).

Parmi les outils de communication On peut trouver :

- **Le mail** : courriel ou e-mail ou courrier électronique : échanges de messages, via la messagerie électronique.
- **Le forum** : asynchrone, animé par un formateur ou animateur de site web (il a le nom de modérateur pour les forums du web), et fonctionne par messages écrits.
- **Les news** : permettent l'envoi d'un message électronique à un groupe d'abonnés ou d'inscrits.
- **La foire aux questions** : les questions de chaque apprenant et leurs réponses données par un formateur, voir un autre apprenant peuvent être consultées.
- **Le wiki** : c'est un outil qui permet de mettre en place un site web collaboratif avec une hiérarchie dans ses pages qui sont reliées par des hyperliens. Un membre peut donc consulter les pages et s'il en a besoin de modifier leur contenu ou tout simplement ajouter de nouvelles pages. Créer un site web mis à jour régulièrement devient alors très facile. Dans le cadre d'e-learning, il sera intéressant d'ouvrir un Wiki privé, réservé aux membres d'un groupe. Ils pourront ainsi bâtir une base de connaissances structurée qui peut être mise à jour ou corrigée facilement. Des systèmes d'authentifications permettent de sécuriser les pages et de ne les rendre éditables que par certains groupes ou même de bloquer les pages pour éviter des débordements.
- **Le blog** : est un outil qui permet de publier des articles rédigés par une ou plusieurs personnes. Les sujets peuvent varier d'un journal intime anonyme à un recensement de nouvelles sur un sujet particulier. Les articles sont généralement présentés du plus récent au plus ancien. Il est donc considéré que les visiteurs viennent régulièrement prendre connaissance des nouveaux articles publiés. Un système où les visiteurs peuvent ajouter un commentaire est dans la majorité des cas intégré ce qui permet à

chacun de réagir à un article donné. Dans le cas d'e-learning, les blogs pourraient être utilisés par un groupe. Ce qui permettrait à chacun de proposer des articles, de proposer des liens ou des nouvelles par le biais de cet outil. Les étudiants construisent donc leur savoir eux-mêmes à travers leurs recherches qu'ils pourront publier et consulter.

- **L'audioconférence et la vidéoconférence** : permettent à deux ou plusieurs personnes éloignées de communiquer entre elles en temps réel. Le son et l'image sont émis et réceptionnés. Cet outil permet dans une application d'e-learning d'offrir un apprentissage à distance. L'enseignant peut envoyer en même temps des schémas, images et autres ressources visuelles aux étudiants se situant sur différents sites. De plus, l'étudiant a la possibilité de prendre la parole pour poser des questions ou y répondre. Il peut être aussi utilisé pour la diffusion de cours ou de conférences. Dans ce cas, si ces transmissions sont enregistrées, on pourrait avoir une utilisation asynchrone de cet outil.
- **La messagerie instantanée** : permet des communications textuelles uniquement entre deux personnes. L'émetteur choisit son interlocuteur à partir d'une liste de présence des connectés à la session synchrone. Le message s'affiche immédiatement dans la machine du destinataire. La messagerie instantanée implique ainsi la fonctionnalité de paging.
- **Tableaux blancs partagés** : ces outils permettent de travailler à plusieurs sur des données textuelles, et /ou graphique en reprenant le concept de tableau sur lequel tout le monde peut écrire.
- **Les documents pédagogiques partagés** : Une zone commune peut être utilisée pour mettre à disposition du groupe, différents documents sous forme numérique (Word, Excel, PDF, etc.)
- **Les sondages** : 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.
- **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.

IV.2.4 Les fonctionnalités d'une plate-forme de E-Learning :

- **Gestion administrative :**

Configuration technique, ergonomique, gestion des droits d'accès.

- **Gestion des cours et des cursus :**

La plateforme permet aux formateurs d'intégrer les cours, d'attribuer différents par cursus aux apprenants et de suivre leurs progressions en récupérant les scores.

- **Gestion de classe virtuelle :**

Une classe virtuelle permet d'organiser les différents groupes d'apprenants pour planifier et animer les formations.

- **Gestion d'outils de communication :**

- ✓ **Forum :** Organisation de discussions entre apprenants et formateurs ;

- ✓ **Bibliothèque :** Mise en ligne de ressources par les formateurs, accessibles pour les apprenants

- ✓ **Espace collaboratif :** Mise en ligne et révision de ressources par les formateurs et les apprenants ;

- ✓ **Courriel :** Envoi de courrier électronique ;.

IV.2.5 Exemples de plateformes d'e-learning

- **Claroline**

Plateforme d'origine belge, initialement créée par l'Université de Louvain sous licence Open Source (GPL), puis étendue grâce à l'implication de nombreux réseaux de développeurs et de formateurs d'origine internationale. [12]

- **Moodle (Modular Object-Oriented Dynamic Learning Environment)**

La plate forme moodle est un LCMS (Learning Content Management System) c'est à dire (LMS+CMS). C'est une plateforme d'apprentissage en ligne servant à créer des communautés d'apprenants autour de contenus et d'activités pédagogiques.

Dotée d'un système de gestion de contenu (CMS) performant, Moodle a aussi des fonctions pédagogiques ou communicatives lui permettant de créer un environnement d'apprentissage en ligne. Ces fonctionnalités permettent de créer des interactions entre pédagogues, apprenants et ressources pédagogiques formant ainsi un réseau et parfois même une véritable communauté autour d'un thème choisi par les membres de la plateforme

(apprentissage d'un logiciel comme utilisation de la plateforme). C'est une particularité que l'on retrouve peu dans les autres plateformes étudiées.

Moodle fait partie des systèmes de e-formation qui sont appelés dispositifs de « Formation Ouverte et à Distance » (FOAD), pour favoriser un cadre de formation Socio-constructiviste. Ce courant de pensée affirme que les gens construisent activement leurs nouvelles connaissances en interagissant avec leur voisinage. Tout ce que vous lisez, voyez, entendez, ressentez et touchez est comparé à vos connaissances antérieures et si cela est viable dans votre monde mental, cela pourra former une nouvelle connaissance qui vous appartiendra. La connaissance est renforcée si vous pouvez l'utiliser avec succès dans un environnement plus large. Cela met l'accent sur le fait qu'il n'y a pas seulement un transfert d'information d'un cerveau à un autre, mais que tout est conditionné par l'interprétation et l'interaction avec un groupe social. [11]

IV.3 Les contenus

- **Les modules de cours**

L'e-learning étant supporté par l'informatique, la technologie n'est jamais très loin et la terminologie devient parfois complexe. Si l'on simplifie les choses, la première notion importante est la notion de *module de cours*. Les modules de formation e-learning sont des composants pédagogiques destinés à transmettre un savoir sur un sujet donné. La forme d'un module varie suivant le sujet abordé, mais comporte en général un enchaînement de séquences de textes, images, animations, sons, vidéos, simulations, etc. Bien utilisée, la diversité des médias employés sollicite les différentes facultés cognitives de l'apprenant et favorise la compréhension et la mémorisation des concepts abordés. L'apprenant lit, écoute et regarde les informations présentées. Il pilote son parcours à travers les différentes séquences qui composent un module (chapitres, sujets, etc.). Le parcours peut être plus ou moins dirigé, et par conséquent laisser plus ou moins de liberté à l'apprenant. Ce degré de liberté est notamment lié à l'ordre des séquences consultées (linéaire ou non).

En général un module de cours comporte des séquences interactives qui sollicitent davantage l'apprenant. Ce dernier doit par exemple répondre à une question, réaliser un exercice ou une simulation. A chaque fois, l'apprenant réalise ces actions à son rythme, en consultant si nécessaire les explications associées aux réponses bonnes ou mauvaises. La durée des modules varie selon la vitesse de parcours de l'apprenant.

- **Les tests**

Pour vérifier l'assimilation d'un cours, rien de tel que de tester l'apprenant. D'où l'intérêt de mettre des évaluations sous diverses formes, les bibliothèques existantes proposent un choix assez conséquent d'exercices, les plus couramment utilisés sont :

- ✓ le QCM (questionnaire à choix multiples),
- ✓ la question à choix unique,
- ✓ etc.

Associées à ces exercices, des fonctionnalités supplémentaires permettent d'enrichir l'évaluation :

- ✓ le feedback, permettant de répondre à l'apprenant en fonction de ses résultats,
- ✓ des timers, pour définir une enveloppe de temps pendant laquelle l'apprenant peut répondre,
- ✓ le tracking, permettant l'enregistrement des résultats, des tentatives effectuées, du temps passé sur un exercice. Il permet aussi d'individualiser le parcours en fonction de ces paramètres. Le tracking est en principe géré par le SGC, la communication entre les contenus et le SGC étant possible grâce aux standards utilisés. Il permet également au formateur de consulter les résultats obtenus et la difficulté qu'a pu éprouver l'apprenant sur les différentes évaluations.

IV.4 Stratégie pédagogique et tutorale

Toutes les méthodes pédagogiques peuvent être envisagées dans un dispositif de formation à distance : transmissive, démonstrative, interrogative, pédagogie de projet, méthode coopérative, ... Elles peuvent être utilisées alternativement selon les modules, les types de formation, les contextes,...

La formation à distance n'est pas synonyme d'auto-formation. Les objectifs visés actuellement tendent plutôt vers une acquisition de compétences plutôt que vers une acquisition de savoirs : « apprendre à apprendre » est une formule souvent utilisée. L'insistance est mise sur la qualité de l'accompagnement : l'apprenant est acteur de la construction de son savoir, le tuteur est un facilitateur. Il n'y a pas de dispositif idéal, valable pour tous. La réussite d'une formation à distance dépend de la capacité d'autonomie de l'apprenant, de la qualité du dispositif et du tutorat, du contexte et de la disponibilité, de la synergie émergente dans la classe virtuelle, ... Elle peut néanmoins être un palliatif tout à fait valable lorsque le contexte temporel, géographique ou budgétaire ne se prête pas à une

formation en présentiel. Elle peut aussi être utilisée dans un mode hybride (blended learning) alternant formation en présentiel et formation à distance.

V. gestion de contenus d'apprentissage

Les contenus, pour être pertinents, restent élaborés par les personnes compétentes, ce sont en principe les auteurs, professeurs et les formateurs. Ils peuvent être inclus en format natif (documents textes, PDF, HTML), et dans ce cas peuvent être mis à disposition directement par les personnes qui les ont élaborés. Le coût est alors moindre puisqu'ils ne nécessitent pas l'intervention de compétences autres que celles des personnes du service pédagogique.

Toutefois, cette façon de présenter les ressources pédagogiques ne peut s'appliquer qu'à certains types de contenus. Souvent, il sera nécessaire d'adapter le cours à une consultation individuelle de l'apprenant, notamment en y adjoignant des schémas, des séquences animées, des commentaires sonores, ceci afin de pallier à la non-présence physique du formateur et à capter davantage l'attention de l'apprenant. Par contre, l'adaptation des contenus a un coût, c'est peut-être actuellement le frein le plus significatif au développement du e-learning, bon nombre d'école ne dispose pas des moyens nécessaires à la mise en place de tels contenus.

Des logiciels générateurs de contenus qu'on appelle aussi **système de gestion des contenus (CMS)** existent, ils permettent à des personnes non informaticiennes de développer des contenus s'intégrant parfaitement au sein du LMS, de mettre en place des exercices d'évaluation de tous types (QCM, texte à trous, réorganisation d'objets, etc.) et ceci, sans développer une ligne de code. Ceci dit, l'aspect visuel concernant la présentation, les schémas, les animations ainsi que l'aspect sonore reste l'apanage des professionnels si l'on désire un contenu de qualité (infographiste, concepteur multimédia, speaker).

Les LCMS ("Learning Content Management Systems", ou "systèmes de gestion de contenus de formation") permettent, à travers l'utilisation d'une architecture technique, de créer et de gérer des contenus de "e-formation". Combinant les fonctionnalités des LMS (les "Learning Management Systems", ou "systèmes de gestion de la formation") et des CMS ("Content Management Systems"), ils forment des systèmes complets de création et de gestion de contenus adaptés aux problématiques de la formation ouverte et à distance

(FOAD). Comprendre le fonctionnement des LCMS et les modes de travail qu'ils induisent nécessite de se pencher sur la nature de leurs "ancêtres", les CMS.

V.1 Qu'est ce qu'un CMS

Les "Content Management Systems" forment une famille de solutions intégrées et "multi plates-formes" de complexités diverses, qui permettent de gérer facilement le contenu dynamique et rédactionnel d'un site web. Ce sont des **systèmes de publication en ligne**, c'est-à-dire de sites web constitués de deux espaces : un espace dit "public" ou "site public", qui est accessible aux internautes et propose l'accès aux contenus, et un "espace d'administration" ou "site privé" dont l'accès est réservé, qui contient le système de gestion et de publication des contenus. Ils offrent la possibilité à des non-techniciens de mettre en place des sites ou des applications Internet, en s'affranchissant des contraintes techniques et de développement. Cette gestion automatisée englobe les multiples facettes de l'édition de tout contenu web : tout d'abord, la création proprement dite, puis la validation des contenus créés, puis enfin leurs conditions de publication.

Les **contenus** disponibles sur Internet sont généralement distingués en deux catégories : soit il s'agit d'éléments dit "statiques", comme un fichier texte, une image, un fichier vidéo ou un cours au format Flash, par exemple, soit il s'agit d'éléments "dynamiques", c'est-à-dire puisés au sein d'une ou plusieurs bases de données, comme une liste de contacts, un catalogue de produits commerciaux ou une série de modules de formation. Le cœur d'un CMS est schématiquement constitué de deux noyaux imbriqués : un système de classement des contenus et un système de suivi de création et de publication de ces contenus. Le premier organise et classe les informations en les associant à des "méta données" prédéfinies, comme la date de rédaction, l'auteur, le titre, la thématique, etc. Il facilite l'accès aux données par des modes de recherche avancés. Le second offre un espace de "workflow" (chaîne de travail) qui formalise et organise les étapes du cycle de vie d'un document : création, validation, publication, archivage, mise à jour, suppression, etc.

La plupart des CMS disposent d'un **fil RSS** mis à jour automatiquement lors de la publication de nouveaux articles.

V.2 Fonctionnalités des CMS

- Ils permettent à plusieurs individus de travailler sur un même document ;
- Ils fournissent une chaîne de publication (*workflow*) offrant par exemple la possibilité de mettre en ligne le contenu des documents ;
- Ils permettent de séparer les opérations de gestion de la forme et du contenu ;
- Ils permettent de structurer le contenu (utilisation de FAQ, de documents, de blogs, de forums de discussion, etc.) ;
- création de cours ou de questionnaire, indexation des ressources, travail collaboratif, publication automatique. [13]

V.3 Principe de fonctionnement d'un site web dynamique utilisant le CMS

La figure ci-dessous explique le fonctionnement d'un site web utilisant un CMS

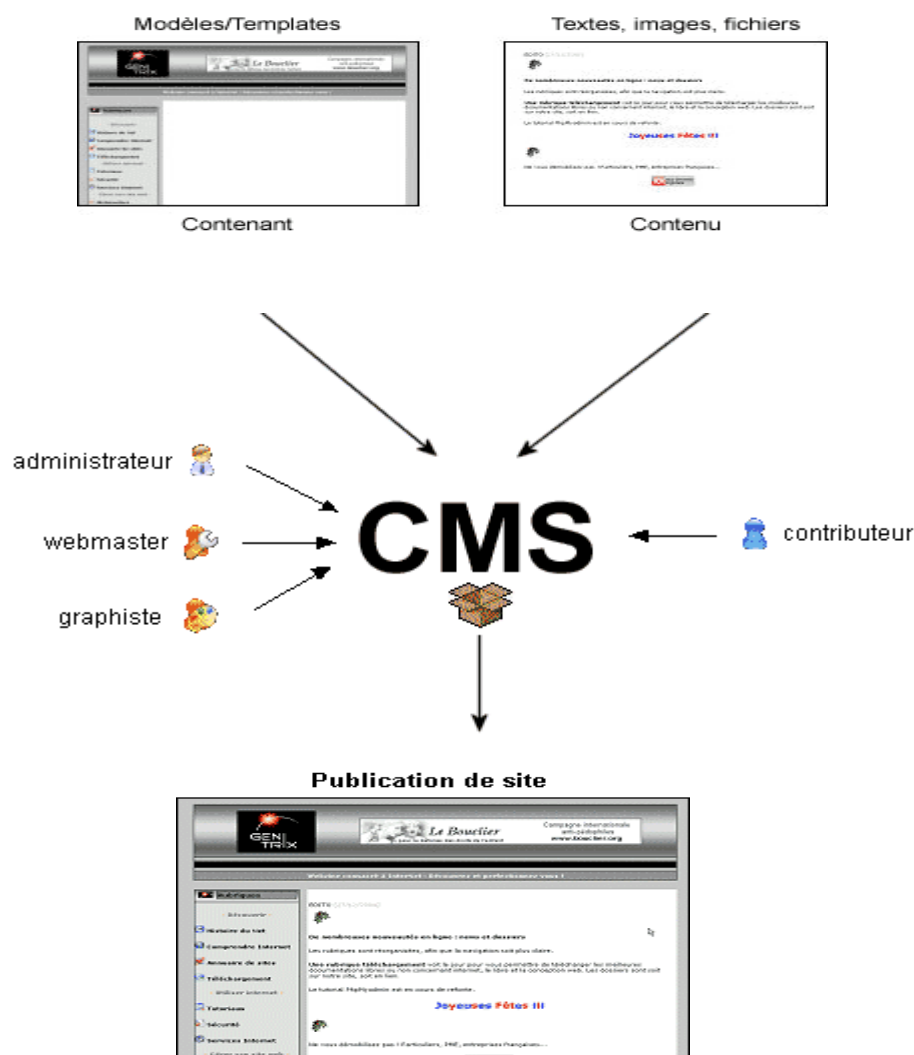


Figure II.1 : Fonctionnement d'un CMS [13]

VI. Normes et Standards pour la réutilisation des contenus pédagogiques élaborés

Les travaux de normalisation dans le domaine des technologies éducatives en sont à leurs débuts. Des perspectives d'ampleur ont été tracées par certains spécialistes et certains organismes. Mais, les premiers résultats commencent seulement à émerger. La majorité des travaux entamés dans les différents groupes se sont concentrés sur la notion d'objet pédagogique (Learning Objects) et de méta-données.

VI.1 objets pédagogiques et métadonnées

VI.1.1 Objets pédagogiques

Selon le **groupe de travail IEEE-LTSC**, un objet pédagogique peut être défini comme « *toute entité numérique ou non, qui peut être utilisée, réutilisée ou référencée lors d'une formation dispensée à partir d'un support technologique* » [14]. un objet pédagogique peut être : un texte, une image, une simulation, un document Word, PDF, HTML, etc. comme il peut aussi bien faire référence à une procédure ou des directives pour aider l'apprenant dans son parcours, à des activités conçues pour suivre les progrès d'un apprenant ou rendre compte de ses résultats. Il peut s'agir également d'un élément visant à faciliter l'interaction entre plusieurs systèmes de gestion de l'apprentissage ou des apprenants.

Beaucoup de travaux s'appuient sur XML pour structurer ces objets, et sur les méta-données pour décrire leurs contenus.

VI.1.2 Méta-données

Les méta-données ont pour but d'ajouter une information de nature sémantique aux objets pédagogiques de manière à obtenir une description aussi précise que possible. Etymologiquement, les méta-données sont des données sur les données, afin de donner des informations sur leurs contenus. Dans une bibliothèque, les méta-données sont les données qui nous permettent de localiser un ouvrage selon son auteur, son sujet, sa date de publication, etc.

En e-learning, les méta-données nous permettent de récupérer plus efficacement les diverses ressources éducatives sur Internet en facilitant la recherche par des descripteurs ou marqueurs.

VI.2 Enjeux de la normalisation

Les principaux enjeux de normalisation sont de rendre interopérable les techniques entre elles afin de faciliter l'usage des ressources éducatives quels que soient la plate-forme ou l'environnement technologique utilisés. À cet égard, les défis sont de plusieurs ordres. La littérature sur le sujet permet de relever les cinq suivants

- **Accessibilité** : permettre la recherche, l'identification, l'accès et la livraison de contenus et composantes de formation en ligne de façon distribuée;
- **Interopérabilité** : permettre l'utilisation de contenus et composantes développés par une organisation sur une plateforme donnée par d'autres organisations sur d'autres plates-formes;
- **Réutilisabilité** : permettre la réutilisation des contenus et composantes à différentes fins, dans différentes applications, dans différents produits, dans différents contextes et par différents modes d'accès;
- **Durabilité** : permettre aux contenus et composantes d'affronter les changements technologiques sans la nécessité d'une réingénierie ou d'un re-développement;
- **Adaptabilité** : permettre la modulation sur mesure des contenus et des composantes.

En contribuant à relever ces défis, la normalisation empêche le domaine de la formation en ligne de se cantonner dans une technicité fermée, basée sur des procédés obscurs. Elle permet plutôt l'ouverture du processus et le décroisement des pratiques en permettant aux dispositifs et aux contenus d'être interopérables dans un monde réseauté.

VI.3 Les principaux standards

Trois approches sont apparues successivement et ont débouché chacune sur des normes ou propositions de normes : LOM, SCORM et IMS-Learning Design.

Le terme « objet pédagogique » émerge au milieu des années 1990 au sein de consortiums internationaux ce qui amènera à proposer un standard, le **LOM** (Learning Object Metadata), au début des années 2000. L'objectif est alors de rentabiliser la production et de développer la réutilisation (point de vue économique). À la fin des années 1990, l'apparition des LMS et LCMS (plates-formes de FOAD) conduit à se poser la question de l'exploitation technique des objets pédagogiques avec internet, d'où l'apparition de **SCORM** au sein des consortiums « corporatistes » américains. L'objectif est de pouvoir exploiter la ressource dans un système informatique et d'en contrôler l'utilisation (point de vue technique).

Dix ans plus tard, la question qui prévaut est l'intégration des objets dans la conception de situations d'apprentissage : les activités et non les ressources deviennent centrales. D'où la création d'**IMS Learning Design (IMS LD)**, qui vise la conception de dispositifs de formation de qualité (point de vue pédagogique).

VII. Avantages et limites du e-learning

- **Les principaux avantages du e-learning**

- ✓ **Permet la formation « juste à temps »**

La formation virtuelle permet aux employés de se former et de se rafraîchir la mémoire en tout temps sur une foule de connaissances sans avoir à attendre la diffusion d'un cours donné à un moment bien précis.

- ✓ **Offre une formation en toute souplesse**

Les modes de diffusion de l'apprentissage virtuel offrent une gamme élargie de possibilités pour recevoir la formation : dans une salle de classe traditionnelle aménagée en conséquence, dans un bureau réservé spécifiquement à l'apprentissage individuel ou à des stations de travail aménagées un peu partout dans l'entreprise pour les employés n'ayant pas accès à un ordinateur, et même à la maison.

- ✓ **Contribue à réduire les coûts de formation**

L'apprentissage virtuel requiert un investissement initial important, mais son utilisation peut souvent engendrer des économies substantielles. Elles se réalisent principalement par la réduction des frais de déplacement des apprenants, des honoraires des formateurs, des pertes de temps de travail grâce à la diminution du temps requis pour l'apprentissage, des pertes de productivité en rendant accessible la formation sur les lieux de travail. Chaque projet de formation virtuelle aura une rentabilité différente d'après les critères de calculs énoncés précédemment. Toutefois, plusieurs entreprises autoriseront tout de même des projets d'apprentissage virtuel sans aucun retour sur l'investissement à cause d'une préoccupation particulière à un moment précis et considéreront ces projets plutôt comme un investissement stratégique.

- ✓ **Uniformise et personnalise l'apprentissage**

L'apprentissage virtuel assure à la fois l'uniformité de la formation et rend accessible aux employés des cours et des parties de cours selon les besoins exprimés par chacun.

- ✓ **Permet de diffuser l'apprentissage de manière simultanée à un large auditoire**

Que vous lanciez sur le marché un nouveau produit ou un nouveau service et que vous deviez former une multitude de personnes simultanément, à un moment précis, l'apprentissage virtuel est la stratégie toute désignée. En effet, vous pouvez joindre, au moment où vous choisissez, tous ceux qui ont accès à un ordinateur branché sur Internet ou sur votre intranet. Vous pouvez dès lors vous assurer que tous auront la même formation au même moment donné.

- **Inconvénients**

- ✓ **La diffusion de cours e-learning nécessite des équipements multimédia.**

L'équipement de l'entreprise doit permettre la diffusion des contenus des cours. Cela concerne notamment les postes utilisateurs (PC récents, logiciels installés compatibles, éventuellement réseau avec suffisamment de bande passante). Cet inconvénient tend à perdre de son importance au fil des années, étant donné l'évolution des technologies.

- ✓ **La mise en place de l'infrastructure technique et la création des contenus sont coûteuses.**

Un investissement est nécessaire au lancement d'une politique du e-learning. Cet investissement (coût fixe et risque important) remplace le paiement à la prestation de formation (coût variable et risque faible). Par ailleurs, le marché n'est pas encore consolidé et l'offre en matière de cours spécialisés reste limitée suivant les domaines.

- ✓ **L'accès à l'outil informatique est nécessaire.**

L'utilisation de l'outil informatique limite la diffusion du e-learning auprès d'une partie des collaborateurs. Ceci est un obstacle par exemple dans l'industrie.

- ✓ **L'e-learning limite les interactions entre les individus.**

Certains mécanismes de communication ne peuvent pas être reproduits (langage du corps par exemple), alors qu'ils jouent un rôle important dans la diffusion du savoir.

Conclusion

L'utilisation d'Internet ou plus généralement les Nouvelles Technologies de l'Information et de la Communication (NTIC) a conduit à l'adoption massive de nouveaux outils ce qui a conduit à la formation ouverte et à distance et à l'apparition des plates formes d'enseignement à distance basé sur l'échange d'information , l'interaction et surtout sur le travail collaboratif et coopératif.

L'intégration des technologies web 2.0 pour l'extension des CMS des plateformes e-learning et l'utilisation du web sémantique pour structurer l'information ou le contenu de formation permettent une plus grande interactivité entre les utilisateurs, une simplicité d'utilisation des plateformes ainsi qu'une facilité d'accès au contenu

Dans le prochain chapitre nous allons voir l'apport du web sémantique pour les plateformes de formation

Chapitre III

web services & web sémantique

Introduction

L'architecture orientée service (SOA) est de plus en plus utilisée dans les entreprises. Cette Architecture apporte beaucoup de nouveautés au monde des systèmes d'information et à l'informatique en général. Elle permet aux concepteurs de systèmes d'information d'organiser un ensemble de services interconnectés, accessible par une interface et des protocoles standards.

Les Web Services représentent l'implémentation la plus utilisée pour appliquer l'architecture SOA. Ils permettent la réalisation rapide et efficace de systèmes d'informations distribués sur internet ou intranet, en intégrant des applications existantes et nouvelles.

Un Web service sémantique prolonge les capacités d'un Web service en leur associant des concepts sémantiques afin de permettre une meilleure recherche, découverte, choix, composition et intégration.

Dans ce chapitre nous présentons l'architecture SOA ainsi que ses objectifs puis nous expliquons les concepts des web services et web sémantique ainsi que le résultat de leur convergence qui est les web services sémantique.

I. l'architecture orientée service (SOA)

Le système d'information (SI) d'une entreprise est habituellement représenté selon quatre niveaux d'architecture :

- L'architecture métier : processus métier de l'entreprise
- L'architecture fonctionnelle : blocs fonctionnels et flux d'information, supports à la réalisation des processus métier, indépendamment des technologies mises en œuvre
- L'architecture applicative : blocs applicatifs et échanges, supports à la réalisation des blocs fonctionnels et des flux, dépendant des technologies mises en œuvre dans le SI.
- L'architecture technique : infrastructure sur laquelle sont implémentés et exécutés les blocs applicatifs et leurs échanges, dépendant des contraintes de performance et de sécurité et de la qualité de services demandée.

Dans une vision SOA du SI, ce découpage reste tout à fait d'actualité, à la différence que la brique de base de construction du SI devient le **service**.

I.1 Qu'est ce que SOA (Service Oriented Architecture)

SOA n'est pas une technologie, ni une recette, encore moins un produit. C'est une façon de penser et de concevoir le Système d'Informations [15]

SOA est un style d'architecture logicielle pour lequel les processus métier de l'entreprise sont des composants logiciels paramétrables, orchestrant des tâches avec les acteurs de l'entreprise et des appels à des composants de services pour s'exécuter. [16]

I.2 qu'est ce qu'un service

Comme son nom le suggère, l'élément clé de SOA est le Service. Un Service est un composant logiciel distribué, exposant les fonctionnalités à forte valeur ajoutée d'un domaine métier [15].

D'un strict point de vue technique, un service possède les caractéristiques suivantes :

- Il propose une interface connue et pérenne.
- Il est invocable à distance (c'est une notion classique des architectures distribuées).
- Il est localisable (à terme, il existe donc un annuaire permettant aux clients de le localiser).
- Il est déployé physiquement sur un serveur.

Chaque service doit répondre aux caractéristiques suivantes pour être éligible dans une SOA :

- Couplage faible : un service doit être indépendant des autres services de sa famille.
- Réutilisation : chaque traitement métier doit être offert par un seul service. Ce besoin de réutilisation doit être traité au niveau d'une gouvernance SOA au sein de l'entreprise.
- Référencé : chaque service est référencé dans un annuaire de services.
- Stateless : l'exécution d'un service est non-interruptible et ne dépend pas de son exécution précédente.

I.3 Les objectifs de l'architecture orientée services

Les axes majeurs de la SOA sont :

- La réutilisation et la composition, permettent le passage de modules entre applications et les échanges inter-applicatifs ;
- La pérennité, qui implique notamment le support des technologies existantes et à venir;
- L'ouverture de l'interopérabilité, pour partager des modules applicatifs entre plates-formes et environnement ;
- La distribution, pour pouvoir utiliser ces modules à distance et les centraliser au sein de l'entreprise par exemple ;
- La performance, avec en priorité l'accent mis sur la montée en charge.
- De meilleures possibilités d'évolution (il suffit de faire évoluer un service ou d'ajouter un nouveau service).

I.4 Principe de l'architecture SOA

Une SOA est un véritable précepte d'architecture, dans laquelle des ressources applicatives hétérogènes se retrouvent façadées par les services, et pour laquelle les processus sont explicites et donc facilement maîtrisables et instrumentables.

Le principe de base de SOA est de publier (sur Internet, Extranet ou encore un Intranet) un ensemble d'offres (existant ou non) réalisant un service bien précis. Une application dans ce cadre appelle ces différents services quand elle en a besoin et peut être elle-même déployée comme un nouveau service (remplissant alors un plus grand rôle dans l'architecture métier).

Pour rendre interopérable les différentes plateformes basées sur l'architecture orientée service, il faut principalement :

- utiliser une API standard et normalisée de communication, aussi bien au niveau protocole de communication que de la mise à disposition des informations sur le réseau.
- coder les données de manières compatibles. Bien souvent les nouvelles architectures utilisent aujourd'hui le méta-langage XML pour coder leurs informations : il existe de nombreux analyseurs XML, disponibles pour un grand nombre de langages et de

plateformes, permettant ainsi le choix de cette technologie sans risquer de se retrouver incompatible.

Une SOA place au cœur de l'architecture d'un système d'information les services (qui sont ses briques de base de construction) puis les processus métier qui vont les orchestrer. Les applications mises à la disposition des acteurs sont alors construites par composition de processus et de services. Processus et Service sont de véritables composants logiciels, qui vont permettre de faire le liant entre la vision métier des Maîtrises d'Ouvrage et la vision technique des Maîtrises d'œuvre.

La figure ci-dessous illustre le principe de base de l'architecture orientée service :

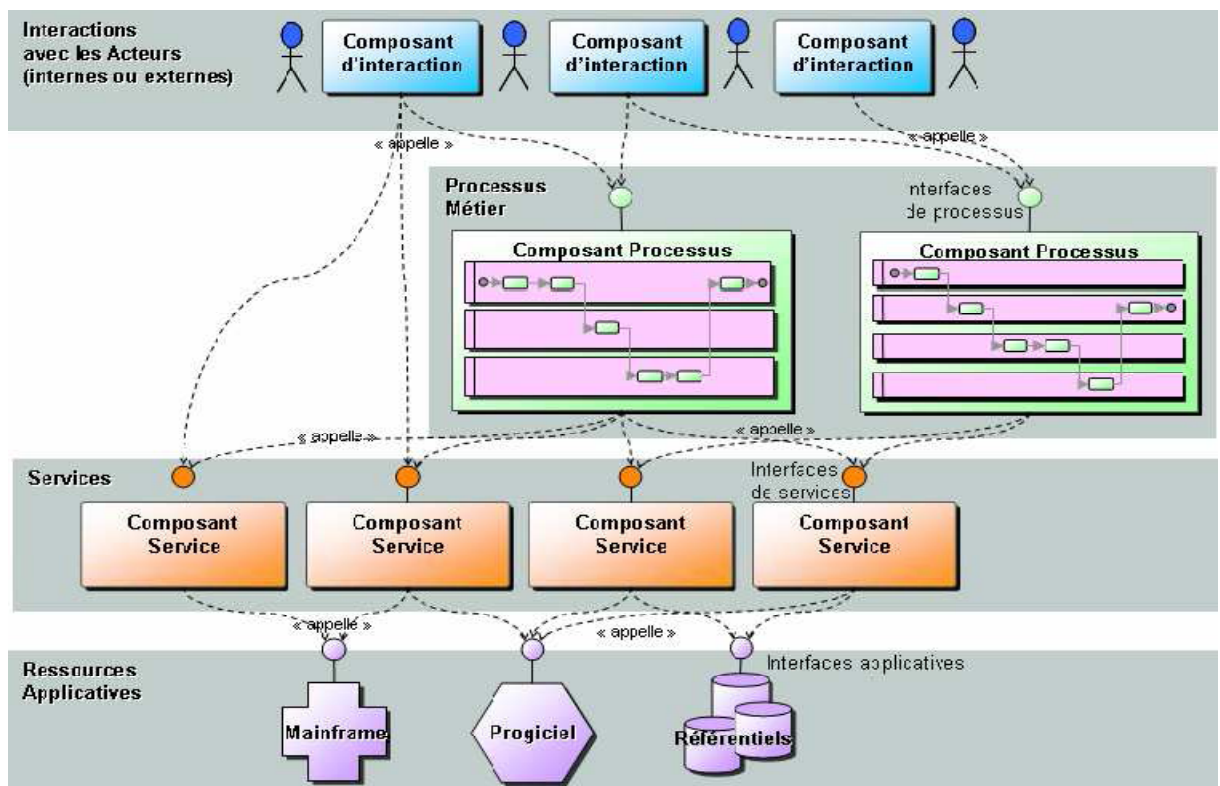


Figure III.1 : Les principes d'une SOA (Architecture applicative) [16]

Une SOA est organisée selon les niveaux suivants :

- **Interactions avec les acteurs :** ensemble des composants logiciels permettant aux acteurs d'interagir avec le système d'information et de réaliser les logiques applicatives. Les acteurs considérés ici sont soit des humains, pour lesquels on mettra typiquement en place des solutions de type portail permettant de traiter la diffusion multi-canal de l'information, ou des systèmes externes dans le cadre d'échanges B2B. Un composant d'interaction communique avec les processus et les services.

- **Processus Métier** : ensemble des composants logiciels permettant de réaliser les processus métier de l'entreprise par une orchestration de tâches automatiques ou humaines, correspondant réciproquement à l'appel d'opérations de services et à de l'attente de données de la part d'acteurs.
- **Services Métier** : ensemble des composants permettant de réaliser les services mutualisés de l'entreprise, et permettant de mettre en place des façades homogènes au-dessus des ressources applicatives. Typiquement les services seront décrits ici par WSDL. Chaque service expose des opérations, ces opérations sont appelées soit via les processus (dans le cadre d'une orchestration), soit directement par les interactions.
- **Ressources applicatives** : ensemble des ressources applicatives du SI exploitées par les services. Nous retrouverons donc ici les bases de données, les mainframes, les applications propriétaires existantes. Les services accèdent aux ressources via l'utilisation de connecteurs. Par exemple, dans le cas d'un accès java à une base de données, nous pourrions mettre en place une connexion JDBC. L'ensemble de ces composants logiciels permet au final de construire des applications dites composites.

II. les web services : une instance de SOA

II.1 Qu'est ce qu'un web service

Plusieurs acteurs définissent les web services par des caractéristiques technologiques distinctives, Une définition globale d'un web service (au niveau technologique) pourrait être :

Un web service est une application logicielle, identifiée par un URI (Uniform Resource Identifier), pouvant interagir avec d'autres composantes logicielles et dont les interfaces et association (binding) ont la capacité d'être publiées, localisées et invoquées [17] via les standards ouverts XML, SOAP, WSDL, UDDI et les protocoles de transport de l'internet. XML est utilisé pour représenter les données, SOAP pour transporter les données, WSDL pour décrire les services disponibles et UDDI pour lister les fournisseurs de services et les services disponibles. Ils sont les bases permettant de construire des systèmes distribués et ouverts sur internet, grâce à leur interface asynchrone utilisant des technologies indépendantes des plateformes et de leurs composantes réutilisables.

Les Web Services permettent de créer des outils capables d'interagir avec différents Web services et de proposer des fonctionnalités intégrées. C'est sur ce principe que fonctionnent les mashups. Les Web Services entrent directement dans la mouvance du **Web**

2.0 : les Web Services permettent aux applications d'être des composants interopérables et réutilisables.

II.2 Les origines du web service

Les efforts en vue de l'élaboration de normes communes pour le développement d'applications distribuées ont une longue histoire en informatique. En suivant la terminologie habituelle en génie logiciel, les technologies qui offrent la couche d'abstraction sur les moyens de communication sont souvent englobées sous le terme générique «intergiciel ».

De nombreuses tentatives antérieures pour établir une solution intergicielle unique standardisée ont complètement échoué ou ont été limité à utiliser dans le cadre des entreprises. En particulier il n'ya pas eu d'accord entre les fournisseurs sur la façon dont une telle standardisation devrait ressembler.

Au début des années 1980, Birell et Nelson ont présenté le RPC (Remote Procedure Calls), la première généralisation des technologies permettant d'invoquer les procédures dans les systèmes distribués. La définition de RPC a été faite d'une manière abstraite cachant les détails de mise en œuvre du développeur.

Les besoins croissants des exigences des applications du monde réel et la large utilisation du paradigme orienté objets ont abouti à plusieurs prolongations de RPC, tels que RPC transactionnels, Objects Brokers, Remote Method Invocation (RMI), Transaction Processing (TP), les moniteurs et le message orienté Intergiciel (MOM).

Des architectures complètes des intergiciels tels que CORBA introduites par l'OMG (Object Management Group) au début des années 1990 ont déjà autorisé des fonctionnalités avancées telles que la sélection dynamique et le binding de services. L'idée de créer un intergiciel assez robuste pour s'adapter à n'importe quelle situation ou problème a été mise en place dans la création de CORBA. Par exemple, quand l'Internet a été disponible pour l'utilisation civile, CORBA a été rapidement adapté pour travailler sur TCP/IP et le tunneling http. Techniquement, ceci a été possible grâce au protocole GIOP/IIOP. Cependant, il s'agit d'une adaptation et non d'une solution faite pour prendre en charge le contexte le l'internet. De plus, CORBA n'avait pas de consensus universel.

Microsoft a développé et stimulé l'utilisation de DCOM et des passerelles entre CORBA et DCOM ont été proposées, mais elles introduisaient plus de problèmes que de

solutions. Ainsi, pendant les années 90, on est passé de la période middleware à la période middlewar, où une compétition s'est déroulée entre CORBA et DCOM sans avoir ni gagnant ni perdant.

A la fin des années 90, les acteurs du monde informatique se sont aperçus que ni CORBA ni DCOM ne convenaient pour supporter les systèmes distribués sur internet. Plusieurs initiatives ont donc démarré pour créer une nouvelle solution plus adaptée à l'internet. En fait les acteurs du monde informatique avaient envie de supporter l'interopérabilité de façon uniforme et extensible. Ils se sont aussi rendu compte qu'un nouveau formalisme pour représenter l'information n'était pas suffisant pour développer les applications dans le contexte de l'internet : il fallait aussi changer de paradigme. Ainsi le paradigme service a été davantage développé et expérimenté pour prendre en charge les exigences de l'internet. Il fallait concevoir des solutions qui puissent être intégrées aux standards de l'internet, comme http et html ou utilisées avec eux. Parmi ces initiatives, nous citons ActiveXML, XML-RPC, SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language) et UDDI (Universal, Description, Discovery and Intégration). Toutes ces initiatives sont fondées sur XML comme norme pour représenter l'information. De plus la notion de service est présente dans l'ensemble des solutions proposées. Les webs services sont le résultat de la convergence de ces initiatives.

II.3 Caractéristiques technologiques du web service

Plusieurs acteurs définissent les web services par des caractéristiques technologiques distinctives. Certains éléments principaux nous paraissent importants :

- Une application logicielle identifiée par un URI (Uniform Resource Identifier). Un URI est la façon d'identifier un point de contenu sur le web, que ce soit un fichier texte, audio ou vidéo.
- Capacité des interfaces et liaisons (bindings) d'être publiées, localisées et invoquées via xml
- Un web service peut être publié dans un registre situé à l'intérieur ou à l'extérieur d'un système d'information. Un web services peut être localisé en interrogeant le registre qui l'héberge. Une fois localisé, un web service peut être invoqué (même par un autre web service) en envoyant une requête appropriée.

- Capacité d'interagir avec d'autres composantes logicielles via des éléments xml et utilisant les protocoles de l'internet.
- L'utilisation de protocoles standards de l'internet tels que HTTP (HyperText Transfert Protocole, le protocole du web), SMTP (Simple Mail Transfert Protocol, le protocole du courriel électronique) et xml.
- C'est une composante logicielle légèrement couplée à interaction dynamique. Par légèrement couplée, on veut dire que le web service et le programme (le consommateur de web services) qui l'invoque peuvent être modifiés indépendamment l'un de l'autre. Cela veut aussi dire que, contrairement à une composante logicielle qui serait fortement couplée, il n'est pas nécessaire de connaître la machine, le langage, le système d'exploitation ou tout autre détail, pour qu'une communication puisse avoir lieu. Cela offre une flexibilité qui permet aux entreprises d'éviter les coûts engendrés par l'intégration via des communications fortement couplées.

II.4 Technologies des web services

Le terme technologies de web services désigne un ensemble de technologies en évolution, basées sur des standards ouverts (non propriétaires) et aptes à la mise en œuvre d'architectures orientées services. Il s'agit, à la base, de technologies de communication entre applications réparties, qui garantissent l'interopérabilité de ces applications dont les implémentations sont hétérogènes.

XML, SOAP, UDDI, WSDL sont les technologies dominantes des web services. Fondamentalement, SOAP, WSDL et UDDI sont des technologies issues de l'intérêt parmi des membres de la communauté internet à développer un mécanisme RPC (Remote Procedure Call) pour échanger des documents XML sur le web.

II.4.1 XML (eXtensible Markup Language)

XML est une famille de technologies développées au sein du W3C (World Wide Web Consortium). Il est né de la tentative de mettre SGML (Standard Generalized Markup Language) sur le web. La première spécification de XML est apparue en février 1998 et se concentre sur les données, contrairement à HTML qui focalise sur la présentation. XML permet donc de transformer l'internet d'un univers d'information et de présentation de sites web statiques à un univers web programmable et dynamique, centré sur les données. XML est largement utilisé par les entreprises et supporté par les fabricants informatiques. Il est

indépendant des plateformes informatiques. Il est lisible par l'humain mais est destiné à être lu par la machine. [17]

XML constitue la technologie de base des architectures web services, il est un facteur important pour contourner les barrières techniques. En effet il apporte à l'architecture l'extensibilité et la neutralité vis-à-vis des plateformes et des langages de développement. De plus, grâce à la structuration, XML permet la distinction entre les données des applications et les données des protocoles permettant ainsi une correspondance facile entre les différents protocoles. L'interopérabilité entre les systèmes hétérogènes demande des mécanismes puissants de correspondance et de gestion des types de données des messages entre les fournisseurs et les clients. C'est une tâche où les schémas de type de données XML s'avèrent bien adaptés. Aujourd'hui, la technologie des web services est essentiellement basée sur XML ainsi que les différentes spécifications qui tournent autour, les espaces de noms, mes schémas XML et les schémas de type.

II.4.2 Le protocole SOAP (Simple Object Access Protocol)

C'est un protocole de dialogue par appels de procédures à distance entre objets logiciels. Sa syntaxe d'utilisation est fondée sur XML et ses commandes sont envoyées sur Internet par l'intermédiaire du protocole HTTP mais aussi SMTP et POP sous forme de texte structuré. Il permet aux systèmes objets distribués de solliciter et d'obtenir des services rendus par d'autres objets, il est moins lourd à mettre en œuvre que d'autres protocoles et c'est pour cela qu'il est de plus en plus adopté. [18]

Le protocole SOAP est une note du Consortium W3C dont Microsoft fait partie, mais qui n'est pas spécifique à Microsoft et Windows avec l'implication d'IBM.

➤ Structure d'un message SOAP

Les messages SOAP sont des transmissions fondamentalement à sens unique d'un expéditeur à un récepteur. Lorsqu'une transmission d'un message commence (par exemple l'invocation d'un web service), un message SOAP (ou document XML) est généré. Ce message est envoyé à partir d'une entité appelée le SOAP sender, localisé dans un nœud SOAP. Le message est transmis à zéro ou plusieurs nœuds intermédiaires (SOAP intermediates) et le processus sera fini lorsque le message arrive au SOAP receiver. Le chemin suivi par un message SOAP est nommé message path.

Un message SOAP est lui-même un document XML qui définit un protocole de communication structuré qui contient les éléments illustrés dans la figure suivante :

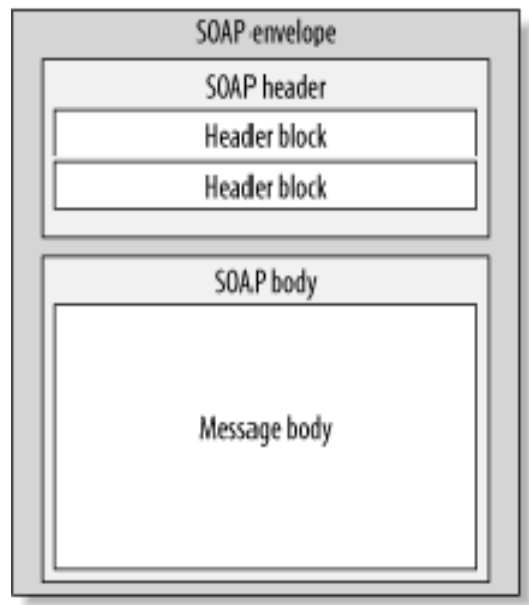


Figure III.2 : Structure d'un message SOAP [26]

- **Enveloppe SOAP (SOAP envelope)**

Un message SOAP est composé d'un élément obligatoire appelé SOAP enveloppe qui est la racine du document XML contenant le message SOAP. Elle est marquée par la balise `envelope`. L'enveloppe SOAP définit l'espace de nom (namespace : URI permettant de connaître la provenance de chaque balise) précisant la version supportée de SOAP. Cet espace de nom est standard et permet de différencier les éléments du schéma. L'enveloppe SOAP est constituée d'un entête facultatif (SOAP header) et d'un corps obligatoire (SOAP body).

- ✓ **Entête SOAP**

L'entête correspond à la balise `Header`, cette partie est facultative, elle doit être tout au début de l'enveloppe SOAP. Elle permet de transmettre des informations complémentaires par rapport à celle contenues dans le message, elle peut avoir plusieurs fils (SOAP blocks). Ces fils sont utilisés pour ajouter des fonctionnalités au message comme l'authentification et la gestion des transactions.

- ✓ **Corps SOAP**

Le corps SOAP contient l'information destinée au destinataire. Il doit fournir le nom de la méthode invoquée par une requête ou le nom de la méthode pour générer la réponse. Il

doit aussi fournir l'espace de noms correspondant au nom du service. Le corps SOAP peut contenir un SOAP fault. Ce bloque est utilisé pour transmettre l'information des erreurs durant le traitement du message.

II.4.3 Le langage WSDL (Web Service Description Language)

Le langage de description WSDL a été créé dans le but de fournir une description unifiée des services web. Il se présente comme un standard actuel dans ce domaine et il est, de plus, normalisé par le W3C. Son objectif principal est de séparer la description abstraite du service de son implémentation même. [19]

➤ Structure d'un document WSDL

Un document en WSDL est constitué de plusieurs définitions. Ces dernières définissent un service comme un groupe d'un ou plusieurs points finaux de réseau aussi appelés ports. Chaque port est associé à une liaison spécifique. C'est cette liaison qui détermine la manière dont un ensemble abstrait d'opérations et de messages est lié à un port selon un protocole particulier. Une liaison établit une correspondance entre un protocole donné et un type de port. Un type de port se compose d'une ou plusieurs opérations, qui représentent un ensemble abstrait d'opérations que peut réaliser le service. Chaque opération est constituée d'un ensemble de messages abstraits qui représentent les données communiquées au cours de l'opération. Chaque message contient une ou plusieurs données définies par des types

La figure suivante illustre la structure d'un document WSDL

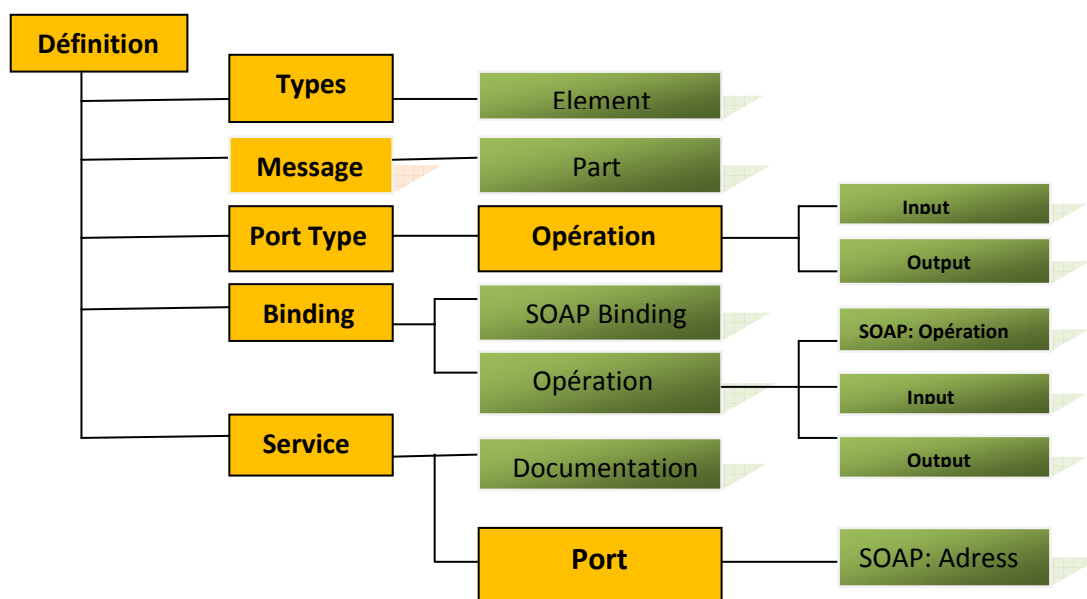


Figure III.3 : Structure d'un document WSDL

Le langage de description WSDL se comporte donc comme un langage permettant de décrire l'interface visible (ou publiée) du service web. Il décrit, à l'aide du langage de balises XML, les différents éléments du service :

- ✓ les messages (intervenant lors des échanges) et leurs types associés (pour gérer l'interopérabilité entre les différents intervenants de l'échange)
- ✓ les opérations (composées de un ou plusieurs messages)
- ✓ les liaisons et les ports de communication (permettant de lier les opérations à un protocole de transport sous-jacent)
- ✓ la description du service lui-même.

- **Les messages** : Ils sont composés de plusieurs parties. Ces parties correspondent, par exemple, dans le paradigme objet aux différents champs d'une structure. Ils sont décrits en XML et un type leur est associé. Les types de base XML peuvent être utilisés (entier, chaîne de caractères, etc.), mais également des types complexes définis dans un fichier WSDL (fichier identique à celui de la description du service ou externe).

- **Les opérations** : Cette partie associe les messages aux opérations (une opération peut être vue comme une méthode). Les différents messages étant les différents paramètres de cette méthode. Un mot clé permet de distinguer le mode :

- ✓ input : mode entrée ou paramètre de donnée
- ✓ output : mode sortie ou paramètre de résultat

Plusieurs opérations sont associées pour former un PortType, utilisé par la suite.

- **Les liaisons** : Elles permettent d'associer les opérations (ou un ensemble d'opérations identifiés par un PortType) au protocole de transport de niveau inférieur. Ainsi, dans le cas d'un service web utilisant le protocole de communication SOAP :

- ✓ pour chaque opération, on définit le type d'échange utilisé (mode document ou mode XMLRPC).
- ✓ pour chaque message (composant les opérations), on décrit l'espace de nom associé au type de message à transporter.

- **Le service** : Enfin, le service lui-même est décrit dans la partie service de cette description WSDL. Le mot clé port permet d'associer des adresses Internet (URL) à

chaque PortType, ces adresses seront utilisées par le client pour l'invocation du service.

II.4.4 Le protocole UDDI (Universal Description, Discovery and Integration)

UDDI est plus qu'un simple protocole : il fournit un protocole, une API et une plateforme permettant aux utilisateurs de services web, depuis n'importe quel système, de localiser dynamiquement à travers Internet les services web qu'ils désirent utiliser. Ceci passe par le biais d'annuaires qui peuvent être maintenus dans plusieurs optiques :

- par une seule et même personne pour son usage interne ;
- par un groupe d'utilisateurs regroupant des services web répondant à certains critères ;
- par un organisme comme base de données mondiale de services web.

UDDI, dans une optique d'interopérabilité, est basé sur les standards tels que HTTP, XML, XML Schema,... [20]

➤ Les différents rôles d'UDDI

UDDI fournit trois services de bases :

- **Publish** : ce service gère comment le fournisseur de service web s'enregistre lui-même ainsi que ses services (en utilisant UDDI).
- **Find** : ce service gère comment un client peut localiser le service web désiré (cela peut passer par des invocations de services web pour une utilisation automatique par un programme ou par une consultation d'annuaire en utilisant des mots clés).
- **Bind** : ce service gère également comment un client peut se connecter et utiliser le service web une fois celui-ci localisé.

Les entreprises qui fournissent ce service et hébergent un annuaire global UDDI sont appelées des **opérateurs UDDI**. Ils sont responsables de la synchronisation de l'information des annuaires : cette synchronisation d'annuaires s'appelle la réplication.

- #### ➤ Les structures de données UDDI:
- UDDI est un modèle d'information composé de structures de données persistantes appelées entités. Ces entités doivent être décrites en XML et sont stockées dans les différents nœuds UDDI.

Les différentes informations sont divisées en trois catégories :

- **pages blanches (White Pages)** : elles regroupent les informations sur les noms des entreprises publiant leurs services web, les moyens de les contacter, etc.
- **pages jaunes (Yellow Pages)** : elles regroupent les informations à propos de la classification des entreprises.
- **pages vertes (Green Pages)** : elles contiennent des informations techniques comme les services offerts par une entreprise, leur spécification, etc. [20]

II.5 Architecture des web services

L'architecture du web services met en relation différents constituants et technologies qui composent une pile du web services ou une implémentation fonctionnelle complète. Les implémentations valides comprennent des sous ensemble ou des parties de la pile, mais elles doivent au moins fournir les composant de l'architecture de base. Les composants et les technologies qui étendent l'architecture de base sont représentés au sein de l'architecture étendue.

II.5.1 L'architecture de base

Cette architecture vise trois objectifs importants :

- Identification des composants fonctionnels.
- Définition des relations entre ces composants.
- Etablissement d'un ensemble de contraintes sur chaque composant de manière à garantir les propriétés globales de l'architecture.

L'architecture de référence des web services s'articule autour des trois rôles suivants :

- ✓ **Le fournisseur de service** : correspond au propriétaire du service. D'un point de vue technique, il est constitué par la plateforme d'accueil du service
- ✓ **Le client** : correspond au demandeur de services. D'un point de vue technique, il est constitué par l'application qui va rechercher et invoquer un service. L'application cliente peut-être elle-même un web service.
- ✓ **L'annuaire des services** : correspond à un registre de descriptions de services offrant des facilités de publication de services à l'intension des fournisseurs ainsi que des facilités de recherche de services à l'intension des clients.

La figure suivante présente l'utilisation de technologies de web services par les acteurs fournisseurs et demandeurs :

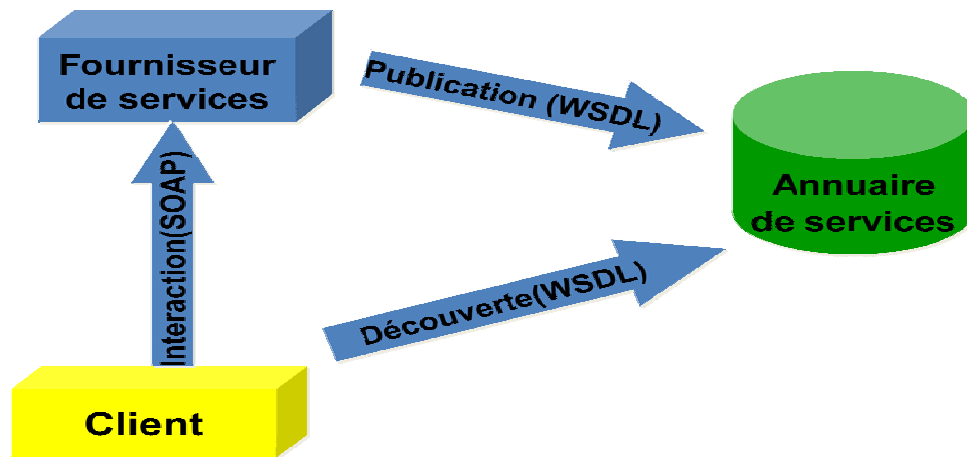


Figure III.4 : Architecture de référence des services web [21]

Pour garantir l'interopérabilité des trois opérations précédentes (publication, recherche et binding), des propositions de standards ont été élaboré pour chaque type d'interaction. Nous citons, notamment les standards émergents suivants : SOAP définit un protocole de transmission de messages basé sur XML. WSDL introduit une grammaire commune pour la description des services. UDDI fournit l'infrastructure de base pour la publication et la découverte des services.

II.5.2 L'architecture étendue

Une architecture étendue est constituée de plusieurs couches se superposant les unes sur les autres, d'où le nom de pile des web services. La pile est constituée de plusieurs couches, chaque couche s'appuyant sur un standard particulier. On retrouve, au-dessus de la couche de transport, les trois couches formant l'infrastructure de base décrite précédemment. Ces trois couches s'appuient sur les standards émergents SOAP, WSDL et UDDI.

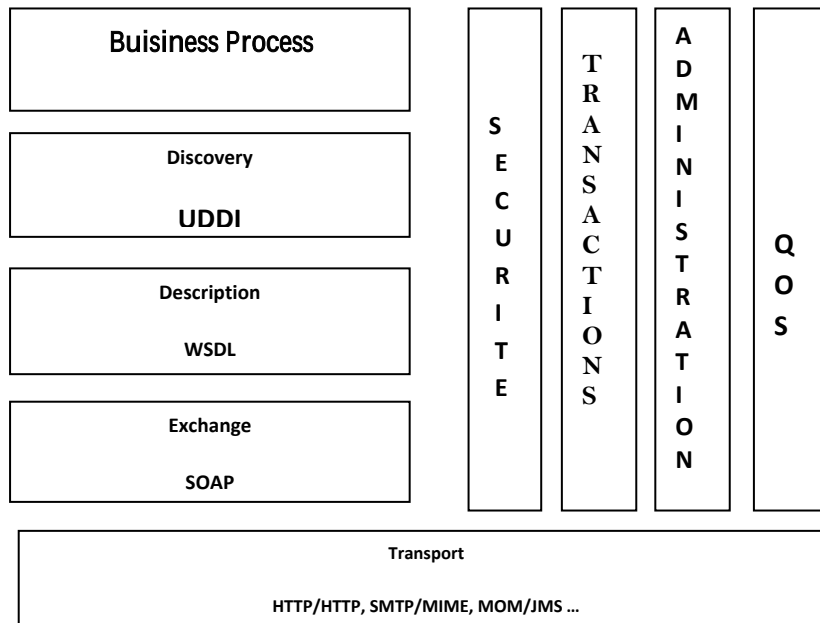


Figure III.5 : Pile des web services

L'architecture de base définit les fondements techniques permettant de rendre les business process accessibles à l'intérieur d'une entreprise et au-delà même des frontières d'une entreprise. Dans ce contexte deux types de couches permettent de la compléter : la couche business process et les couches transversales

- **La couche business process**

Cette couche supérieure permet l'intégration des web services. Elle représente le business process comme un ensemble de web services. De plus la description de l'utilisation des différents services composants ce processus est disponible par l'intermédiaire de cette couche.

- **Les couches transversales (Security, Transactions, administration, QoS)**

Ces couches supportent les différents standards mis à jour fréquemment, intervenant dans l'ensemble du processus d'un web services. Parmi ces derniers, nous pouvons mettre en relief la sécurité, l'administration, la qualité du web services.

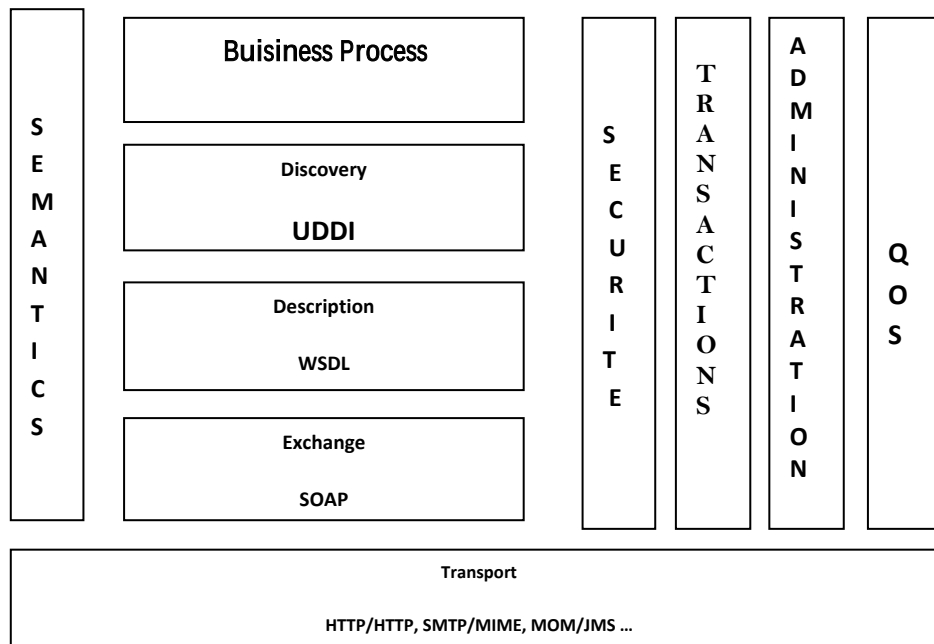


Figure III.6 : Pile du web service sémantique

Des travaux tentent d'intégrer le web sémantique dans ces couches transversales en ajoutant une couche verticale représentant le **web sémantique** et étant utilisable par les quatre couches horizontales représentant les standards

III. Le web sémantique

Le Web Sémantique est une vision du futur Web dans lequel, à l'information est donné un sens explicite facilitant ainsi aux machines le traitement et l'intégration des informations sur le Web. Le Web Sémantique sera construit sur la capacité de XML de définir des schémas de balisage personnalisés et sur la flexibilité de l'approche RDF pour représenter les données. Si les machines sont supposées faire des tâches de raisonnement utiles sur ces documents, le langage doit aller au-delà des sémantiques de base du RDF Schema. OWL a été conçu pour répondre à ce besoin pour un langage d'ontologie pour le Web. OWL est une partie d'une pile évolutive des recommandations W3C relativement au Web Sémantique. [24]

III.1 Objectifs du web sémantique

Un des principaux objectifs du Web sémantique est de permettre aux utilisateurs d'utiliser la totalité du potentiel du Web : ainsi, ils pourront trouver, partager et combiner des informations plus facilement.

Aujourd'hui tout le monde est capable d'utiliser des **forums**, d'utiliser des **réseaux sociaux**, de **chatter**, de faire des recherches ou même d'acheter différents produits. Néanmoins, il serait mieux que la machine fasse tout ceci à la place de l'homme, car actuellement, les machines ont besoin de l'homme pour effectuer ces tâches. La raison principale est que les pages Web actuelles sont conçues pour être lisibles par des êtres humains et non par des machines. Le Web sémantique a donc comme principal objectif que ces mêmes machines puissent réaliser seules toutes les tâches fastidieuses comme la recherche ou l'association d'informations et d'agir sur le Web lui-même.

III.2 Les spécifications du Web Sémantique

Il a été bien reconnu dans la communauté du Web Sémantique que les ontologies jouent un rôle clé dans la livraison du Web Sémantique en facilitant le partage d'information entre les communautés des humains et des agents logiciels. Les DTDs et les schémas XML peuvent être utilisés pour échanger les données entre les parties qui ont convenu les définitions avant l'échange.

Cependant, leur manque de sémantique ne permet pas aux personnes de créer de nouveaux vocabulaires XML. Le même terme peut être utilisé avec différents sens dans différents contextes et des termes différents peuvent être utilisés pour des articles ayant le même sens. RDF et RDF Schema ont essayé de résoudre ce problème en permettant d'associer des sémantiques simples aux identificateurs. Avec RDF Schema, une personne peut définir des classes qui ayant plusieurs sous-classes et super classes, et peut aussi définir des propriétés pouvant avoir des sous propriétés, de domaines ou d'intervalle. Dans ce sens RDF Schema est un langage d'ontologie pour le Web primitif. Pour atteindre l'interopérabilité entre les divers schémas développés de façon autonome, des sémantiques plus riches sont nécessaires.

Il semble clair que le Web Sémantique ne pourra voir le jour sans un minimum de standardisation. Différents consortiums et organismes mettent donc les acteurs autour d'une table pour définir les langages à utiliser dans le Web Sémantique. L'intérêt de cette approche est d'assurer des traitements uniformes sur des documents écrits dans ces langages. Les travaux de standardisation sont bien avancés. Durant le **10 février 2004**, OWL et RDF sont devenus des recommandations du W3C. RDF est utilisé pour représenter l'information et pour échanger la connaissance sur le Web. OWL est utilisé pour publier et partager les ensembles de termes (appelées les ontologies), en supportant les recherches Web avancées, la gestion de la connaissance et des agents logiciels. [24]

La figure suivante illustre les couches du web sémantique

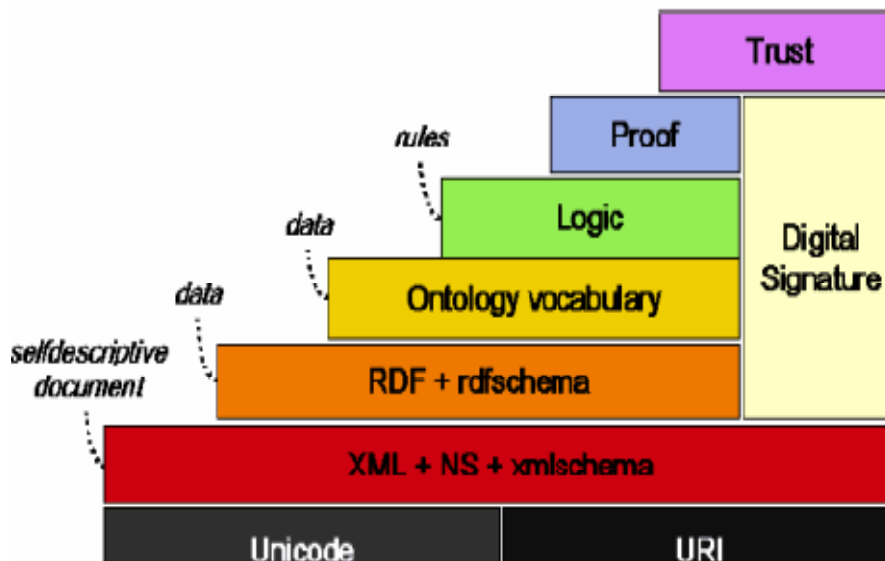


Figure III.7 : Couches du Web Sémantique selon le W3C [23]

III.2.1 Ontologie

Une ontologie est un modèle de connaissances organisé qui est spécifique à un sujet ou à un domaine. elles se composent d'éléments appelés "concepts, attributs, relations et instances": les concepts ou classes correspondent aux objets à organiser (ex: projets, personnes, produits, etc.); les attributs sont les caractéristiques de ces objets (ex: titre, adresse, couleur, etc.); les relations rapprochent deux objets ou un objet et une propriété; les instances correspondent aux données réelles.

III.2.2 RDF (Resource Description Framework)

RDF est un formalisme pour la description des méta-données. Il fournit l'interopérabilité entre les applications qui échangent des informations sur le web qui peuvent être compréhensibles par les machines. RDF augmente la facilité de traitement automatique des ressources web. Il peut être utilisé pour annoter des documents écrits dans des langages non structurés. La syntaxe de RDF repose sur le langage de balisage extensible XML. XML fournit une syntaxe pour encoder des données tandis que RDF fournit un mécanisme décrivant le sens des données. Un des buts de RDF est de rendre possible la spécification de la sémantique des données basées sur XML d'une manière standardisée et interopérable. [32]

Un document RDF est un ensemble de triplets de la forme < sujet, prédicat, objet >. Les éléments de ces triplets peuvent être des URIs (Universal Resource Identifiers), des littéraux ou des variables. Cet ensemble de triplet peut être représenté de façon naturelle par un graphe plus précisément un multi-graphe orienté étiqueté où les éléments apparaissant comme sujet ou objet sont des sommets, et chaque triplet est représenté par un arc dont l'origine est son sujet et la destination est son objet. Ce document sera codé en machine par un document RDF/XML, mais est souvent représenté sous une forme graphique comme le montre la figure ci-dessous.

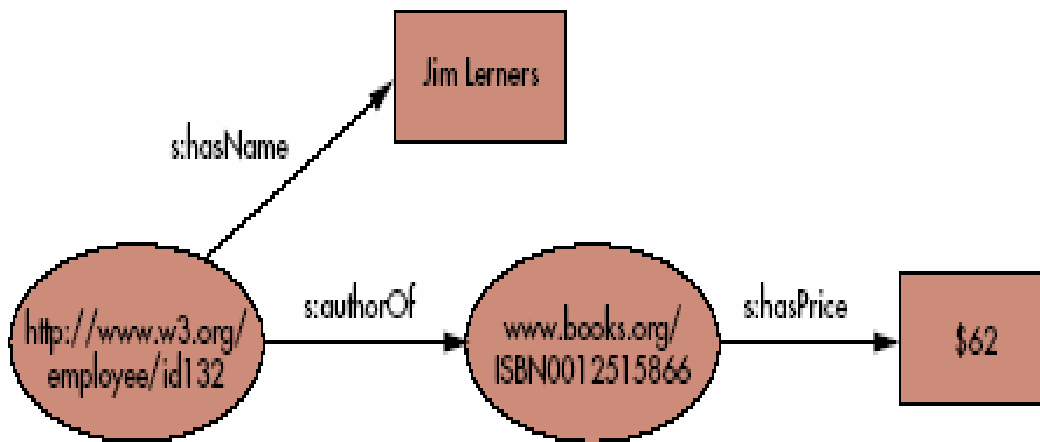


Figure III.8 : Exemple de graphe RDF [24]

III.2.3 OWL (Web Ontology Language)

Le langage OWL a été conçu pour être utilisé par les applications qui traitent le contenu de l'information au lieu de la présenter seulement aux êtres humains. OWL facilite grandement l'interopérabilité au niveau machine du contenu du Web plus que ce qui est déjà supporté par les XML, RDF et RDF Schema (RDF-S) en fournissant du vocabulaire supplémentaire avec des sémantiques formelles. [24]

OWL peut être utilisé pour représenter le sens des termes dans des vocabulaires ainsi que les relations qui existent entre ces termes. Cette représentation des termes et de leurs interrelations est appelée ontologie [24]. OWL décrit la structure d'un domaine en termes de classes et de propriétés comme les approches orientées objets.

OWL fournit des sous langages de plus en plus expressifs OWL Lite, OWL DL et OWL Full conçus pour faire un compromis entre son pouvoir expressif et son pouvoir de raisonnement

- **OWL Lite** : Supporte les utilisateurs qui ont besoin des hiérarchies de classifications et des caractéristiques de contraintes simples. Par exemple, lorsqu'il supporte les contraintes de cardinalité, il permet seulement les valeurs 0 et 1.
- **OWL DL** : plus complexe que OWL Lite, permettant une expressivité bien plus importante. OWL DL est fondé sur la logique descriptive (d'où son nom, OWL Description Logics), un domaine de recherche étudiant la logique, et conférant donc à OWL DL son adaptation au raisonnement automatisé. Malgré sa complexité relative face à OWL Lite, OWL-DL garantit la complétude des raisonnements (toutes les inférences sont calculables) et leur décidabilité (leur calcul se fait en une durée finie).[29]
- **OWL Full** : OWL Full est la version la plus complexe d'OWL, mais également celle qui permet le plus haut niveau d'expressivité. OWL Full est destiné aux situations où il est plus important d'avoir un haut niveau de capacité de description, quitte à ne pas pouvoir garantir la complétude et la décidabilité des calculs liés à l'ontologie. OWL Full offre cependant des mécanismes intéressants, comme par exemple la possibilité d'étendre le vocabulaire par défaut de OWL. [29]

IV. Web services sémantique

Les services Web sémantiques sont à la convergence de la technologie de services Web et du Web sémantique, ce sont des services Web à descriptions dotées de sémantique. Actuellement les services Web sont décrits par le langage WSDL qui permet de définir les opérations et les paramètres autorisés par le service Web. Cela est fait en attribuant des noms aux opérations et aux paramètres, puis associer ces paramètres à des types abstraits de données (string, char,...).

Cependant, le problème avec ce type de description est que lorsqu'on veut automatiser les divers aspects liés aux services Web (découverte, composition,...), l'agent manipulant les paramètres du service Web ne peut pas avoir la signification de ces données. Pour l'agent logiciel c'est juste des variables dénotées contenant de l'information. A ce stade, les services Web ne sont décrits qu'au niveau syntaxique. Un développeur voulant programmer une application cliente interagissant avec un service Web doit tout d'abord avoir connaissance de la syntaxe de sa description, l'interpréter, puis écrire le code client conforme aux paramètres

de la description du service Web. Cependant, un agent logiciel ne peut lire la description d'un service Web comme un humain, il peut avoir connaissance de la structure syntaxique de la description mais pas sa sémantique. Les services Web sémantiques sont des services Web décrits de telle sorte qu'un agent logiciel puisse interpréter les fonctionnalités offertes par ces services Web.

Un agent logiciel doit être capable de lire la description d'un service Web pour déterminer si le service Web fournit les fonctionnalités désirées, et s'il est lui-même capable d'utiliser ce service. Pour permettre cela, la description du service Web doit être complétée en information sémantique interprétable par machine. Les paramètres du service Web doivent être décrits de façon qu'un agent logiciel puisse avoir connaissance de leur signification. Cela est fait par la définition de vocabulaires organisés en ontologies.

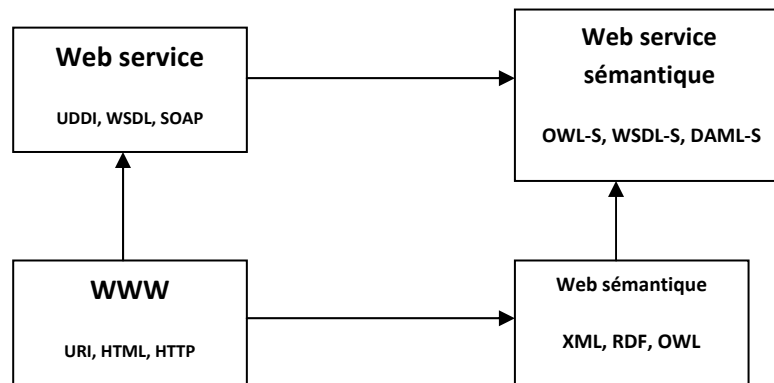


Figure III.9 : Les services web sémantique

Ainsi pour qu'un agent logiciel puisse avoir connaissance de la sémantique d'une description d'un service Web, il lui suffit juste d'accéder à une ontologie du domaine.

Un agent essayant d'interpréter la description du service Web aura juste à utiliser l'ontologie où les annotations sémantiques sont définies. En utilisant un moteur d'inférence, l'agent peut inférer les similitudes entre la sémantique employée pour décrire le service et la sémantique connue par l'agent.

La combinaison des technologies du Web sémantique à celles des Services Web permettra d'automatiser la découverte de services Web, c'est à dire localisation automatique des services Web qui fournissent une fonctionnalité particulière et qui répondent aux propriétés demandées par l'utilisateur. Pour pouvoir effectuer une découverte automatique, le procédé de

découverte devrait être basé sur la similitude sémantique entre la description déclarative, faite par l'utilisateur, du service demandé et celle du service offert.

IV.1 Approches Proposées Pour Les Services Web Sémantiques

IV.1.1 WSDL-S (Web Service Description Language-Semantic)

WSDL-S est un langage de description sémantique des services Web. Une description WSDL-S de service Web est une description WSDL augmentée de sémantique, cette sémantique est ajoutée en deux étapes : La première étape consiste à faire référence, dans la partie définition de WSDL, à une ontologie dédiée au service à publier. La seconde étape consiste à annoter les opérations de la définition WSDL de sémantique en ajoutant deux nouvelles balises : La balise *Action* et la balise *Contrainte*.

IV.1.2 OWL-S (Ontology Web Language for Services)

OWL-S est un langage de description et une ontologie OWL (Web Ontology Language) de services Web. La structuration de l'ontologie supérieure d'OWL-S est motivée par la nécessité de fournir trois types d'information essentiels pour un service, à savoir : Que fait le service ? Comment fonctionne le service ? Comment accède-t-on au service ?

IV.1.3 DAML-S (DARPA Agent Markup Language for Services)

DAML-S est un langage de description sémantique des services Web, il est constitué d'une ontologie lui permettant de décrire les services Web. Il permet l'automatisation de nombreuses tâches à savoir : la découverte et la sélection d'un service qui consistent à mettre en correspondance un demandeur de service avec un fournisseur qui détient ce service, l'invocation de ce service qui permet de réaliser l'appel effectif du service découvert, l'interopération et la composition de services qui autorisent la composition des services simples afin d'obtenir un service complexe qui répond aux exigences d'un client et la surveillance de l'évolution du processus qui sert à surveiller l'état d'accomplissement et de déroulement du service. DAML-S définit une ontologie particulière qui permet la description des propriétés des services Web et de les rendre disponibles au monde.

Conclusion

Avec la technologie d'Ontologie pour la représentation des connaissances, l'annotation des Web services pose plusieurs avenues de recherches. C'est un nouveau domaine de recherche et il reste encore beaucoup d'obstacle : par exemple les outils pour la réalisation et l'utilisation des ontologies, les langages, les méthodes de modélisation sémantique pour les Web services.

L'intégration du web sémantique avec les technologies des services Web peut former un Web interactif et intelligent.

Aujourd'hui, les services web sémantiques constituent une voie nouvelle permettant de mieux exploiter les services Web en automatisant autant que possible, les différentes tâches liées à ces services et à leur découverte. Dans ce chapitre nous avons présenté les principales approches actuelles pour la réalisation des services Web sémantique à savoir : WSDL-S, OWL-S et DAML-S. Cependant, il existe une nouvelle approche qui offre tout les outils nécessaires pour couvrir tout le cycle de vie d'un service Web sémantique et qui est la plus favorite, c'est l'approche WSMO.

Chapitre IV

Analyse & Conception

Introduction

Pour la réalisation d'une application informatique, il est indispensable de suivre une démarche méthodologique et rigoureuse pour planifier et concevoir l'application, en mettant en évidence tous les objectifs tracés pour la bonne élaboration du projet souhaité.

Pour notre part, nous avons opté pour la modélisation orientée objet avec le langage de modélisation le plus répandu UML, qui permet d'écrire avec un langage standardisé les plans d'élaboration et de construction de logiciels.

I. Analyse

I.1 Les problèmes rencontrés dans les systèmes de gestion de contenus (CMS)

Différents problèmes sont rencontrés dans les systèmes de gestion de contenus (CMS) des plateformes existantes parmi lesquels on trouve :

- Les utilisateurs souffrent de plus en plus de la lenteur des plateformes due au rechargement des pages web.
- Les utilisateurs doivent accéder à chaque fois aux plateformes pour voir les nouveaux contenus publiés (nouveau de la plateforme)
- Disponibilité des ressources pédagogiques et leurs échanges entre différentes plateformes : il existe sur le web un volume considérable de ressources pédagogiques, quelques plateformes d'apprentissage actuelles, contrairement aux systèmes traditionnels, offrent une certaine ouverture sur le web, d'où la nécessité de développer des spécifications descriptives de ces ressources pour faciliter leur localisation et réutilisation
- La diversité des standards (LOM, IMS, SCORM...) pour la description des ressources pédagogiques a rendu leur réutilisation difficile due à la non compatibilité des spécifications utilisées dans les diverses plateformes.

I.2 Présentation et objectifs de l'application

Les éléments clés sur lesquels nous nous sommes basés, sont la souplesse et la simplicité. Cette logique est sous-tendue par une interface simple et conviviale, respectant la tâche à accomplir, et adaptée à des utilisateurs novices en informatique.

Dans notre plate-forme d'e-learning, on propose comme solution pédagogique, la diffusion de ressources pédagogiques à travers le WEB à l'aide des web services tout en intégrant les technologies du web 2.0 dans le CMS de la plateforme (sites web disposant de fonctionnalités de publication ou **systèmes de publication en ligne**) qui permet de produire, stocker, diffuser ainsi que gérer les ressources pédagogiques.

Pour cela, notre travail consiste en :

- ✓ La structuration des fonctionnalités de l'auteur sous forme de web services tout en intégrant les technologies ajax permettra la réutilisation de ces fonctionnalités par d'autres plateformes et l'une des caractéristiques du web2.0 qui fait que le cms de la plateforme sera vu comme étant un ensemble de services web permet la création en ligne de contenu pédagogique par un simple appel ajax à un web service SOAP ce qui rend l'interface de l'auteur efficace et rapide en lui permettant de créer plusieurs contenus pédagogiques sans le rechargement de la page.
- ✓ intégration des **flux rss** qui seront mis à jour en fonction du contenu pédagogique ce qui permet de se tenir informé des publications de sites indépendants utilisant ces flux.
- ✓ La structuration des contenus pédagogiques produits sous forme de services web permet la construction d'un système de communication inter-plateformes pour partager l'information (Réutilisation et adaptabilité du contenu pédagogique).
- ✓ l'ajout de la sémantique aux services web permet d'affiner la recherche des ressources pédagogiques élaborées. Pour cela il faut ajouter aux informations existantes une couche de méta-données pour que les ordinateurs puissent l'exploiter.

I.3 Les cas d'utilisation

Étudions à présent les cas d'utilisation de notre application. Nous commencerons par définir les acteurs, puis nous nous attarderons sur leurs rôles dans l'application.

I.3.1 Identification des acteurs et des fonctionnalités attendues

Nous identifions principalement comme acteurs: Les auteurs, les apprenants, les formateurs ainsi qu'un administrateur qui gère les rôles de chacun d'eux.

➤ **Administrateur**

C'est le responsable de l'application et c'est lui qui gère la plate-forme et les autres acteurs en l'occurrence les apprenants, les auteurs et les formateurs.

L'application doit lui permettre de :

- S'authentifier puis accéder à son espace.
- Gérer les demandes de préinscriptions: gérer et répondre aux demandes faites par les visiteurs souhaitant devenir membres de la plate-forme (validation ou refus).
- Consulter les listes des inscrits: l'administrateur peut consulter à tout moment les listes des apprenants, formateurs et auteurs inscrits sur la plate-forme et accéder à leur profil et espace.
- Gérer des inscriptions : supprimer un inscrit
- Se déconnecter de son espace personnel

➤ **Visiteur**

On appelle visiteur toute personne qui navigue sur le site et qui n'a pas encore eu un login et un mot de passe pour se connecter à la plate-forme. Éventuellement, le visiteur peut solliciter une demande de préinscription en remplissant un simple formulaire afin d'obtenir un login et un mot de passe qui lui permettra de se connecter à l'application.

L'application doit lui permettre de :

- Accéder à la page d'accueil
- Demander une préinscription (en tant qu'apprenant, formateur ou auteur) : Le visiteur peut solliciter une demande de préinscription s'il le désire en remplissant un formulaire renfermant des informations personnelles telles que nom, prénom...etc et cela lui permettra de devenir apprenant, formateur ou auteur dans la plate-forme après

l'acceptation de l'administrateur et avoir les privilèges d'un apprenant, formateur ou auteur.

- Consulter l'aide : pour avoir plus d'information sur le fonctionnement de la plateforme

➤ **Auteur (Le producteur du contenu)**

C'est la personne chargée de créer le contenu pédagogique (cours, exercices, tp...)

L'application doit lui permettre de :

- S'authentifier et accéder à son espace.
- **Publier des cours** : mettre des cours en ligne pour que les apprenants puissent les consulter et cela en faisant des appels ajax au service de création de cours.
- **Publier des exercices** : mettre des exercices en ligne pour que les apprenants puissent les consulter et cela en faisant des appels au service de création d'exercices.
- **Publier des travaux pratiques** : mettre des tp en ligne pour que les apprenants puissent les consulter et cela en faisant des appels ajax au service de création de tp.
- Consulter la liste des ressources pédagogiques (cours, exercices, tp) disponibles en faisant des appels ajax au web service de consultation de ressources pédagogiques
- Supprimer des ressources pédagogiques par un simple appel ajax au web service de suppression de ressource pédagogique
- Modifier une ressource pédagogique
- Afficher une ressource pédagogique.
- **Se déconnecter de son espace personnel**

➤ **Apprenant**

L'apprenant est un visiteur ayant reçu une réponse favorable à sa préinscription (validation par l'administrateur) lui permettant de suivre des cours via la plateforme

L'application doit lui permettre de :

- S'authentifier et accéder à son espace.

- Consulter les cours disponibles sur la plate-forme
- Consulter les exercices disponibles sur la plate-forme
- Consulter les travaux pratiques disponibles sur la plate-forme
- Répondre à un exercice
- **Effectuer une recherche de ressources pédagogiques:** En formulant des requêtes de recherche, l'apprenant dispose d'une barre de recherche dans son espace personnel qui lui permet d'effectuer des recherches selon des critères personnalisés.
- **Se déconnecter de son espace personnel**

➤ **Formateur**

Tout visiteur désirant prendre en charge la formation des apprenants en ligne par le biais de la plate-forme peut soumettre une demande à l'administrateur. Une fois accepté il pourra utiliser les offres disponibles sur la plate-forme.

L'application doit lui permettre de :

- S'authentifier puis accéder à son espace.
- Déposer un support de cours.
- Proposer des exercices.
- Proposer des travaux pratiques
- Proposer des solutions aux exercices.
- consulter les cours disponibles sur la plateforme.
- consulter les exercices disponibles sur la plateforme.
- consulter les travaux pratiques disponibles sur la plateforme.
- **Gérer les travaux pratiques**
- **Se déconnecter de son espace personnel**

I.3.2 Identification des scénarios

Chaque tâche effectuée par un ou plusieurs utilisateurs sera décrite par un ensemble de scénarios. Un scénario est une description narrative du comment est ce que le système pourra être utilisé.

Acteurs	Taches	Scénarios
	T0 : Se connecter à la plateforme	S0 : Saisir l'URL du site

Visiteur		
	T1 : demande de préinscription.	S1 : cliquer sur le lien « apprenant » ou «formateur » ou « auteur » S2 : cliquer sur le lien « s’inscrire » S3 : Remplir le formulaire d’inscription et valider.
	T2 : Consulter l’aide	S4: Cliquer sur le lien «à propos»
Auteur	T0 : Se connecter à la plateforme	S0 : Saisir l’URL du site
	T1 : s’authentifier	S1 : Cliquer sue le lien « auteur » S2 : Remplir les champs login et mot de passe S3 : Connexion
	T2 : Publier des cours	S4 : Cliquer sur le lien « cours » S5 : Cliquer sur le lien «Publier un cours » S6 : Remplir les champs du formulaire S7 : publier le cours
	T3 : Publier des exercices	S8 : Cliquer sur le lien « exercice » S9 : Cliquer sur le lien «Publier un exercice » S10 : Remplir les champs du formulaire S11: publier l’exercice
	T4 : Publier des travaux pratiques	S12 : Cliquer sur le lien « exercice » S13: Cliquer sur le lien «Publier un exercice » S14: Remplir les champs du formulaire S15: publier l’exercice
	T5 : Consulter les cours	S16: Cliquer sur le lien « cours » S17 : Cliquer sur «consulter les cours »
	T6 : Consulter les exercices	S18: Cliquer sur le lien « exercice » S19 : Cliquer sur «consulter les exercies »
	T7 : Consulter les tp	S20: Cliquer sur le lien « tp»

		S21 : Cliquer sur «consulter les tp »
	T8 : modifier une ressource pédagogique (ex: cours)	S22: Cliquer sur le lien « cours » S23 : Cliquer sur «consulter les cours » S24: Cliquer sur «modifier» S25: modifier le cours et enregistrer les modifications
	T9 : supprimer une ressource pédagogique	S26: Cliquer sur le lien « cours » S27 : Cliquer sur «consulter les cours » S28 : Cliquer sur «supprimer»
	T10 : déconnexion	S29: cliquer sur le lien « déconnexion »
Formateur	T0 : Se connecter à la plateforme	S0 : Saisir l'URL du site
	T1 : S'authentifier.	S1: Cliquer sur le lien formateur S2 : Remplir les champs login et mot de passe S3 : Connexion
	T2 : proposer des supports de cours	S4 : Cliquer sur le lien « proposer des supports de cours » S5: Remplir les champs du formulaire S6 : publier le cours
	T3 : proposer des exercices	S7 : Cliquer sur le lien « proposer des exercices » S8 : Remplir les champs du formulaire S9: publier l'exercice
	T4 : proposer des tp	S10: Cliquer sur le lien « proposer des tps » S11 : Remplir les champs du formulaire S12: publier l'exercice
	T5 : consulter les cours	S13: Cliquer sur le lien « cours » S14 : Cliquer sur «consulter les cours »
	T6 : Consulter les exercices	S15: Cliquer sur le lien « exercice » S16 : Cliquer sur «consulter les exercices »

	T4 : proposer des solutions aux exercices	S17 : Cliquer sur le lien « consulter les exercices ». S18: Cliquer sur le lien « solution ». S19: écrire la solution et enregistrer
	T5 : consulter les tps	S14: Cliquer sur le lien « travaux pratiques » S15 : Cliquer sur «consulter les tp »
	T5 : déconnexion	S16: Cliquer sur le lien « déconnexion »
Administrateur	T0 : Se connecter à la plateforme	S0 : Saisir l'URL du site
	T1 : S'authentifier.	S1 : Cliquer sue le lien administrateur S2 : Remplir les champs login et mot de passe S3 : Connexion
	T2 : Consulter la liste des apprenants préinscrits	S4 : Cliquer sur « les préinscrits » S5 : Appuyer sur le bouton « afficher »
	T3: Consulter la liste des auteurs préinscrits	S6 : Cliquer sur « les préinscrits » S7 : Appuyer sur le bouton « afficher »
	T4 : Consulter la liste des formateurs préinscrits	S8 : Cliquer sur « les préinscrits » S9 : Appuyer sur le bouton « afficher »
	T5 : Accepter un préinscrit	S10 : Cliquer sur « les préinscrits » S11: Appuyer sur le bouton « afficher » S12: Cliquer sur le lien « accepter »
	T5 : Refuser un préinscrit	S13 : Cliquer sur « les préinscrits » S14 : Appuyer sur le bouton « afficher » S15: Cliquer sur le lien « refuser»
	T5 : Détailler un préinscrit	S16 : Cliquer sur « les préinscrits » S17: Appuyer sur le bouton « afficher » S19 : Cliquer sur le lien « détailler»
	T5 : Consulter la liste des apprenants inscrits	S20: Cliquer sur « les inscrits » S21 : Appuyer sur le bouton « afficher »
	T6 : Consulter la liste des auteurs inscrits	S22 : Cliquer sur « les inscrits » S23 : Appuyer sur le bouton « afficher »
	T7 : Consulter la liste des	S24 : Cliquer sur « les inscrits »

	formateurs inscrits	S25 : Appuyer sur le bouton « afficher »
	T8 : Supprimer un inscrit	S26 : Cliquer sur « les inscrits » S27 : Appuyer sur le bouton « afficher »
	T8 : Déconnexion	S16: Cliquer sur le lien « déconnexion »

II. Conception

La conception est une étape primordiale dans le cycle de vie d'un logiciel. Elle permet de décrire les solutions adoptées pour résoudre des problématiques d'une application ou d'un système.

Dans cette partie, nous présenterons quelques diagrammes correspondants à notre application.

II.1 La démarche d'élaboration du projet

La démarche d'élaboration de notre projet s'appuie sur la modélisation UML. Pour se faire nous avons adopté la démarche suivante :

- ✓ Le diagramme de contexte permet de spécifier le nombre d'instances d'acteurs connectés à ce système à un moment donné.
- ✓ A partir de la définition des besoins, nous avons identifiés les acteurs et leurs activités, à partir desquels nous avons déduit les cas d'utilisation.
- ✓ A partir des cas d'utilisation nous avons établie un ensemble de scénarios d'utilisation standard.
- ✓ Les diagrammes de séquences sont utilisés à identifier les classes nécessaires pour construire un diagramme de classes.
- ✓ En dernier lieu nous avons décrit l'architecture physique des composants matériels qui supportent l'exécution du système par un diagramme de déploiement.

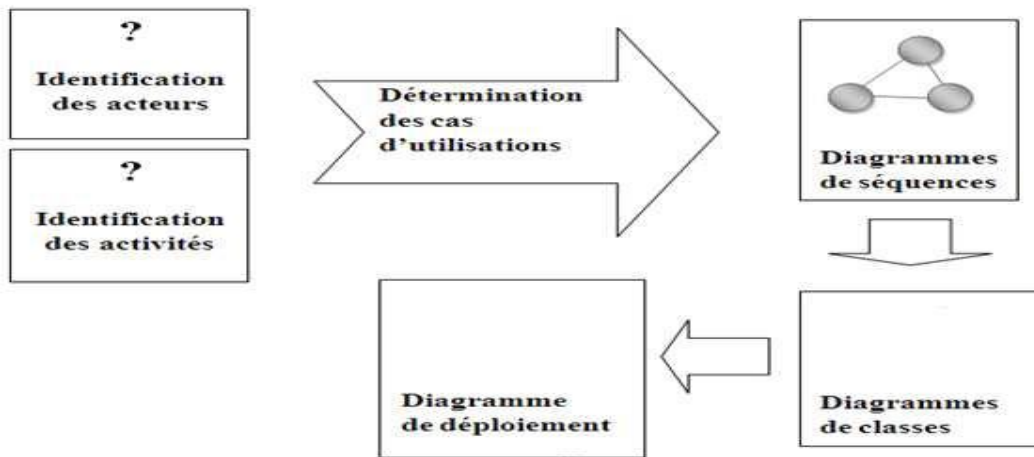


Figure IV.1: Méthodologie de modélisation

II.2 Diagrammes représentatifs

II.2.1 Diagramme de contexte de l'application

La définition des besoins présentée ci-dessus nous a permis de dégager le diagramme de contexte suivant

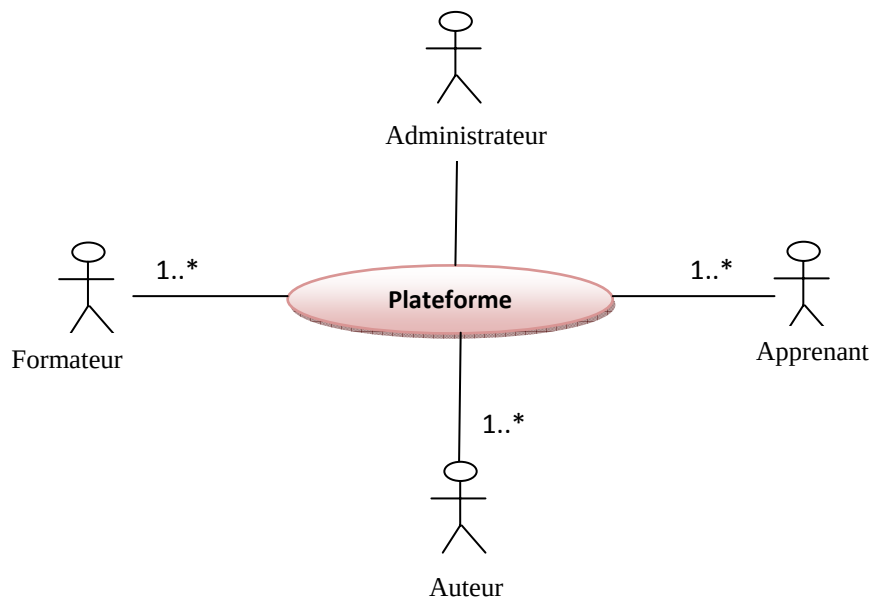


Figure IV.2 : Diagramme de contexte

II.2.2 Diagrammes de cas d'utilisation

Le diagramme de cas d'utilisation est un diagramme UML utilisé pour donner une vision globale du comportement fonctionnel de notre application. Ce diagramme représente une unité discrète d'interaction entre un utilisateur (homme ou machine) et un système. Dans un diagramme de cas d'utilisation, les utilisateurs qui sont appelés acteurs (actors) interagissent avec les cas d'utilisation (use cases).

➤ Diagramme de cas d'utilisation « Administrateur »

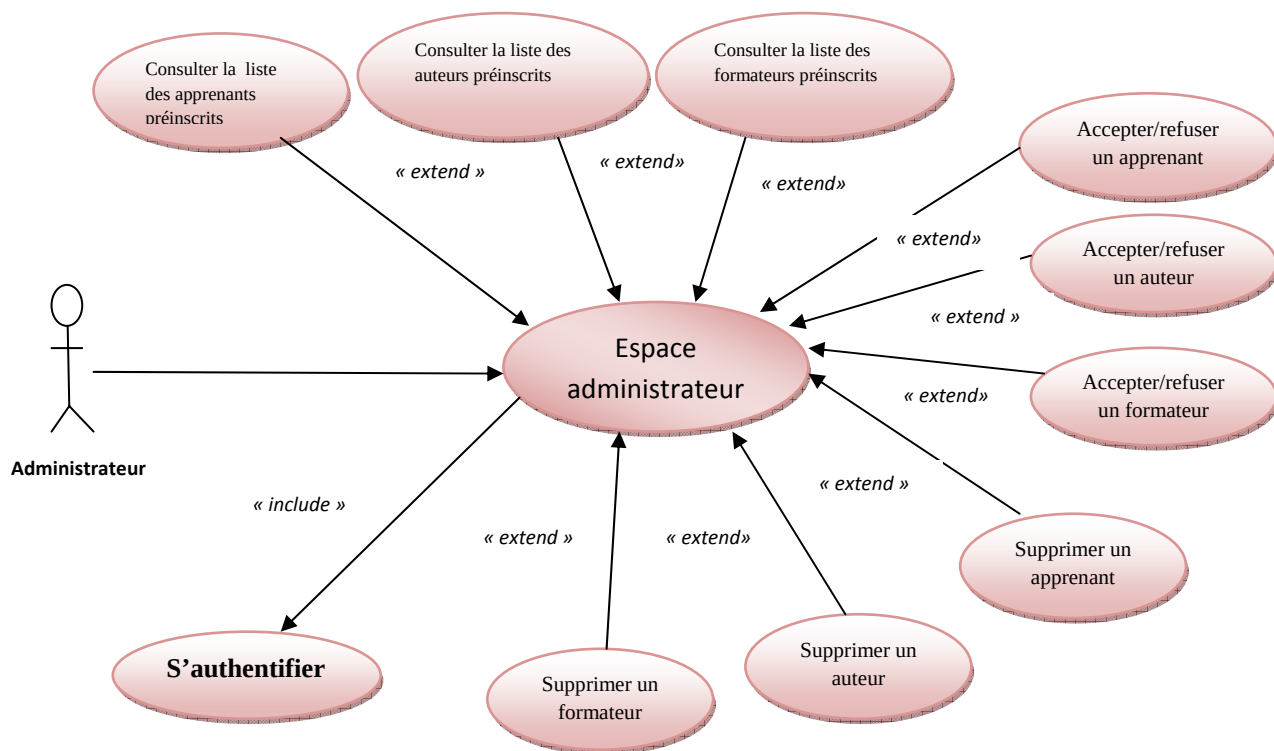


Figure IV.3 : Diagramme des cas d'utilisation « administrateur »

➤ Diagramme de cas d'utilisation « créateur de contenu (auteur) »

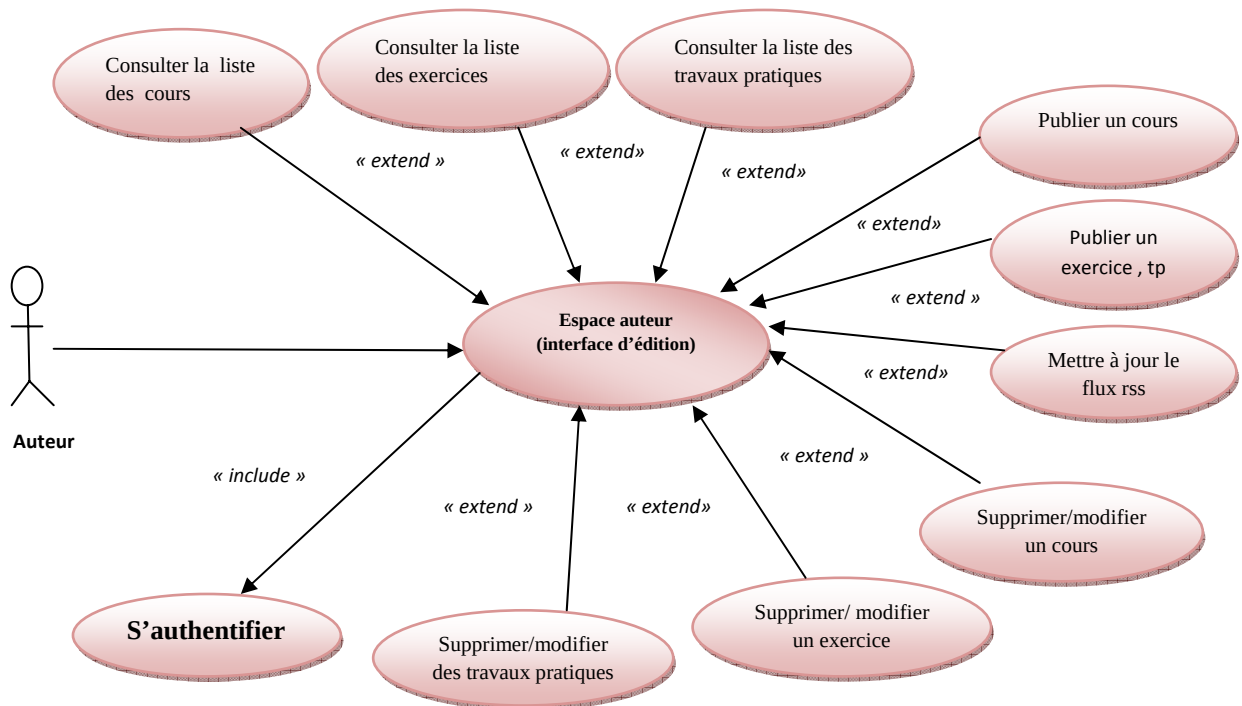


Figure IV.4 : Diagramme des cas d'utilisation « auteur»

➤ Diagramme de cas d'utilisation « apprenant »

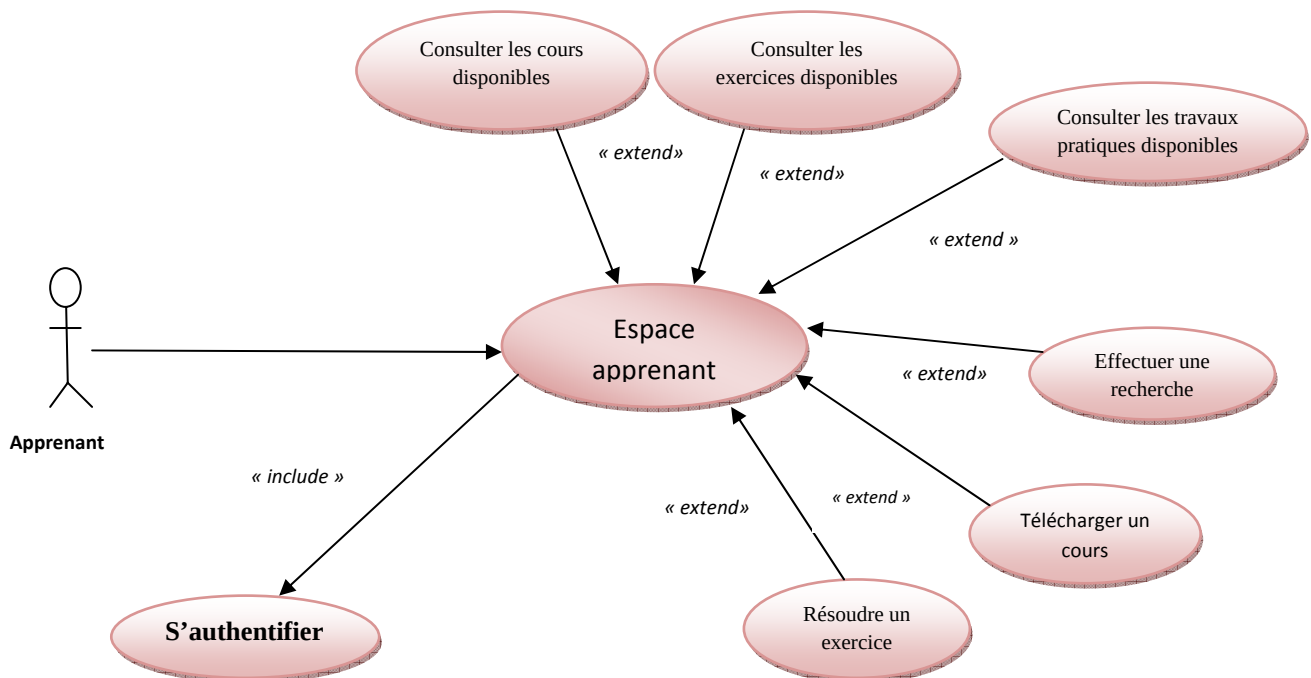


Figure IV.5 : Diagramme des cas d'utilisation « apprenant »

➤ Diagramme des cas d'utilisation « formateur »

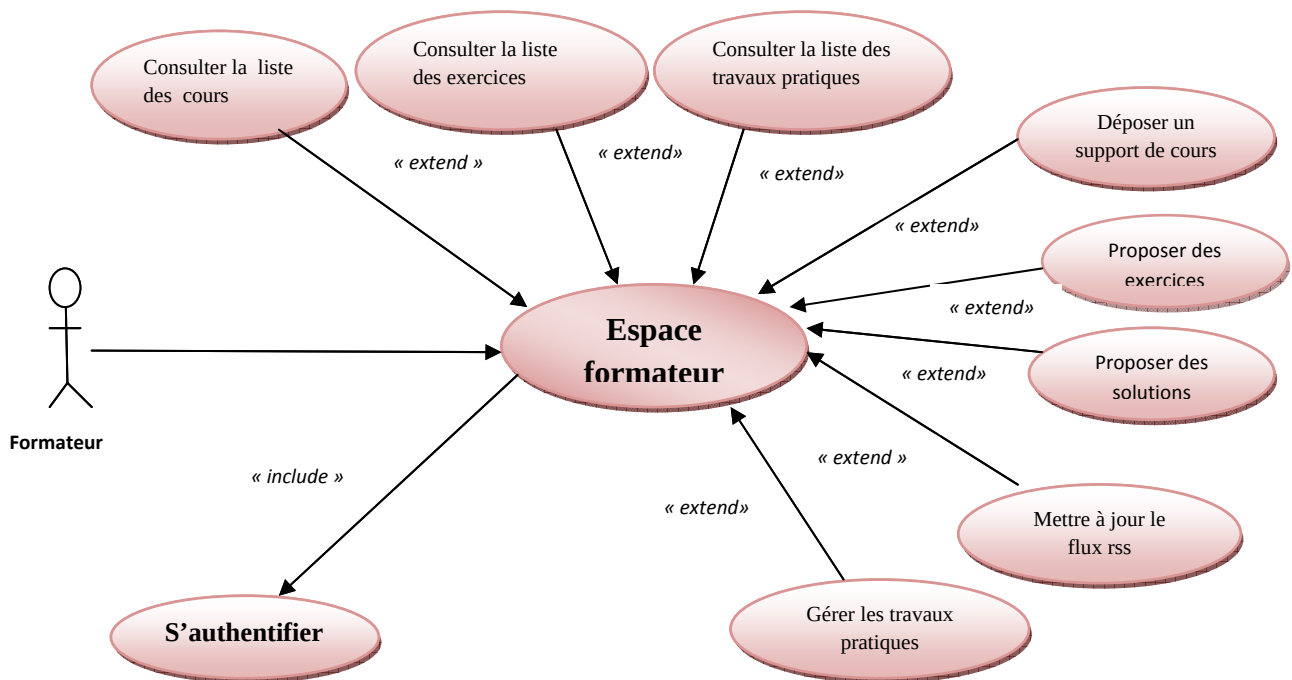


Figure IV.6 : Diagramme des cas d'utilisation « formateur »

II.2.3 Diagrammes de séquence

Les diagrammes de séquence offrent une vue dynamique du système par une spécification du comportement du système. Les diagrammes de séquence décrivent les interactions entre les objets. Ces interactions sont modélisées comme des échanges de messages. Contrairement au diagramme de collaboration, on n'y décrit pas le contexte ou l'état des objets, la représentation se focalise sur l'expression des interactions.

➤ Diagramme de séquence « créer contenu pédagogique »

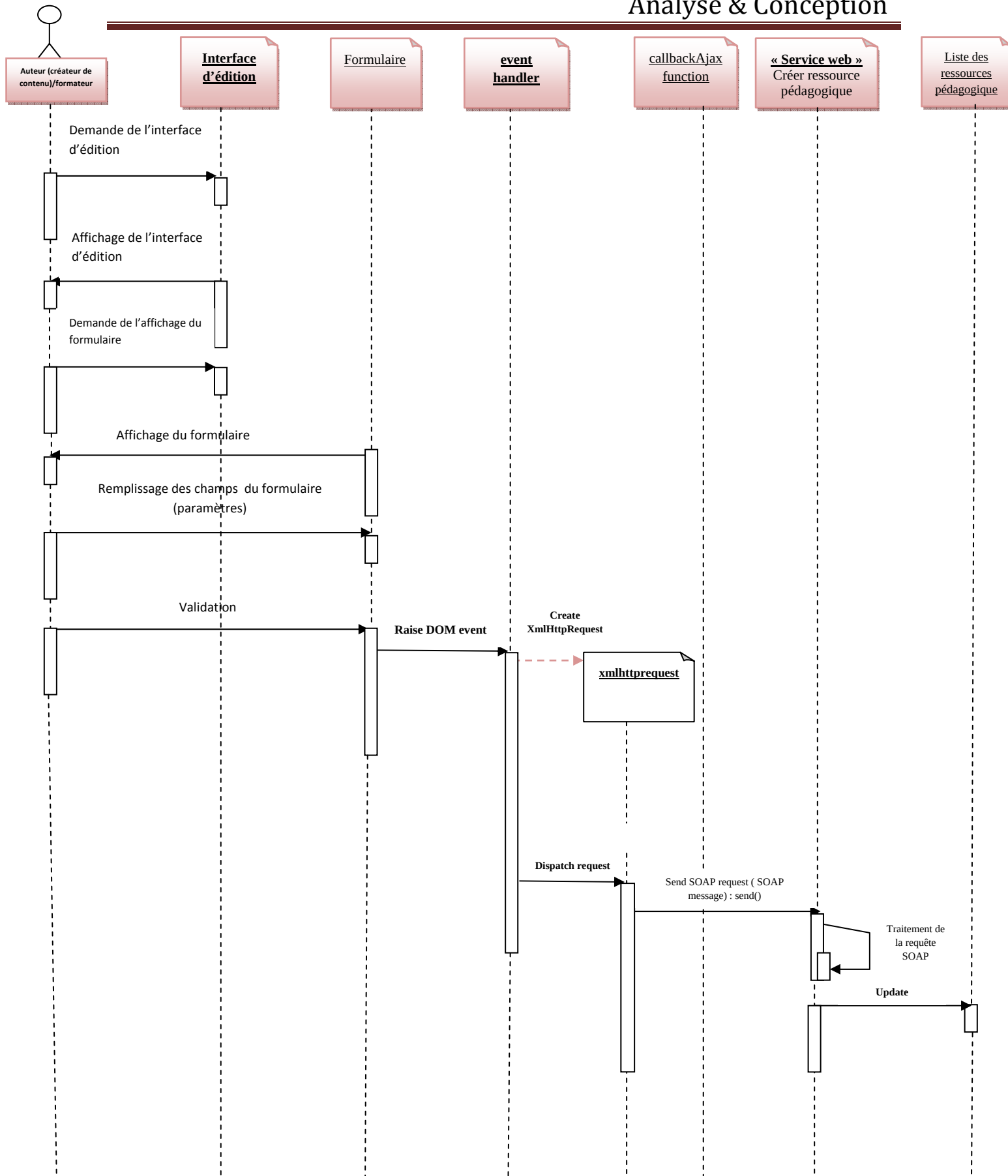


Figure IV.7: Diagramme de séquence « créer contenu pédagogique »

➤ Diagramme de séquence « consulter le contenu pédagogique »

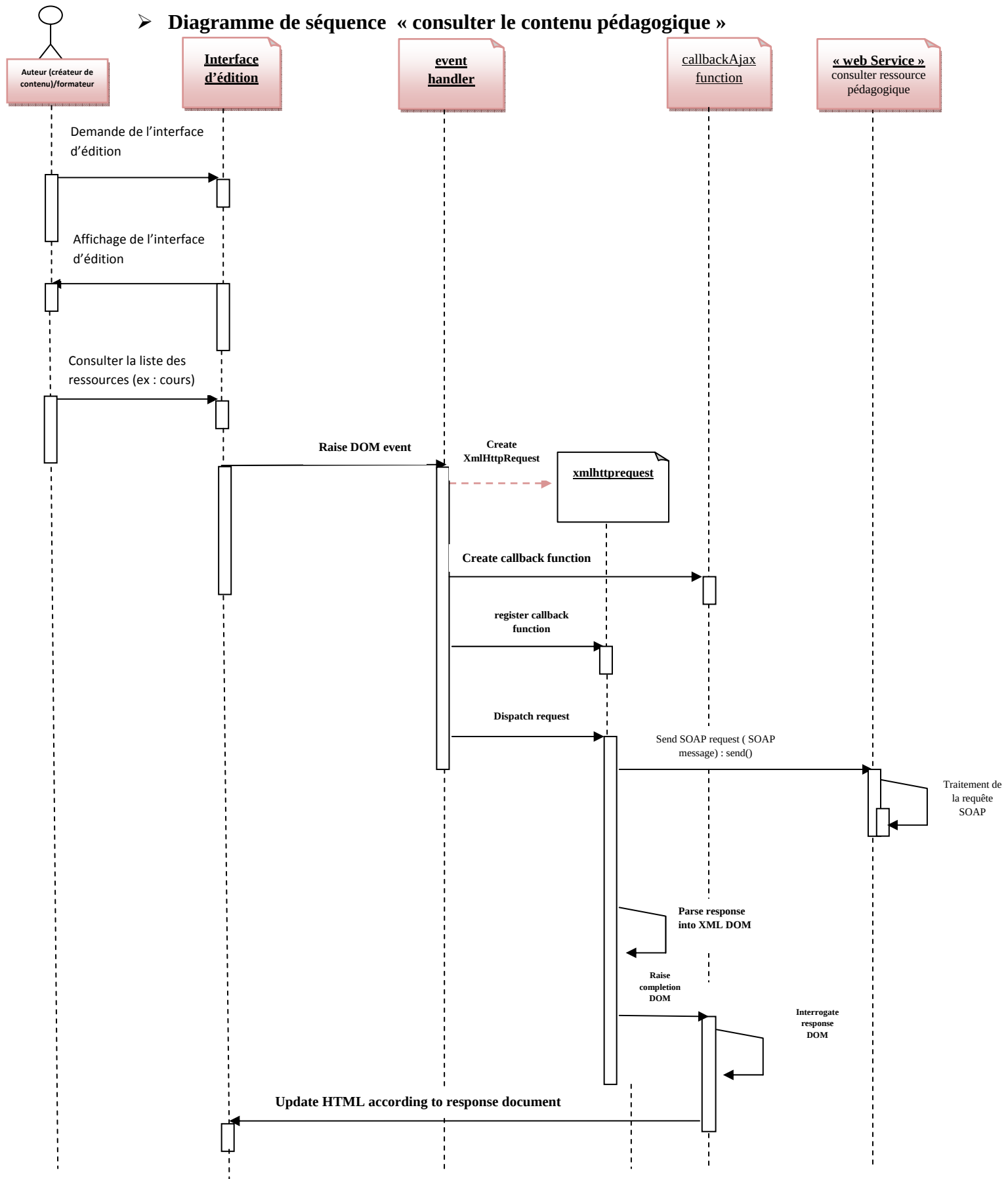


Figure IV.8: Diagramme de séquence « consulter contenu pédagogique »

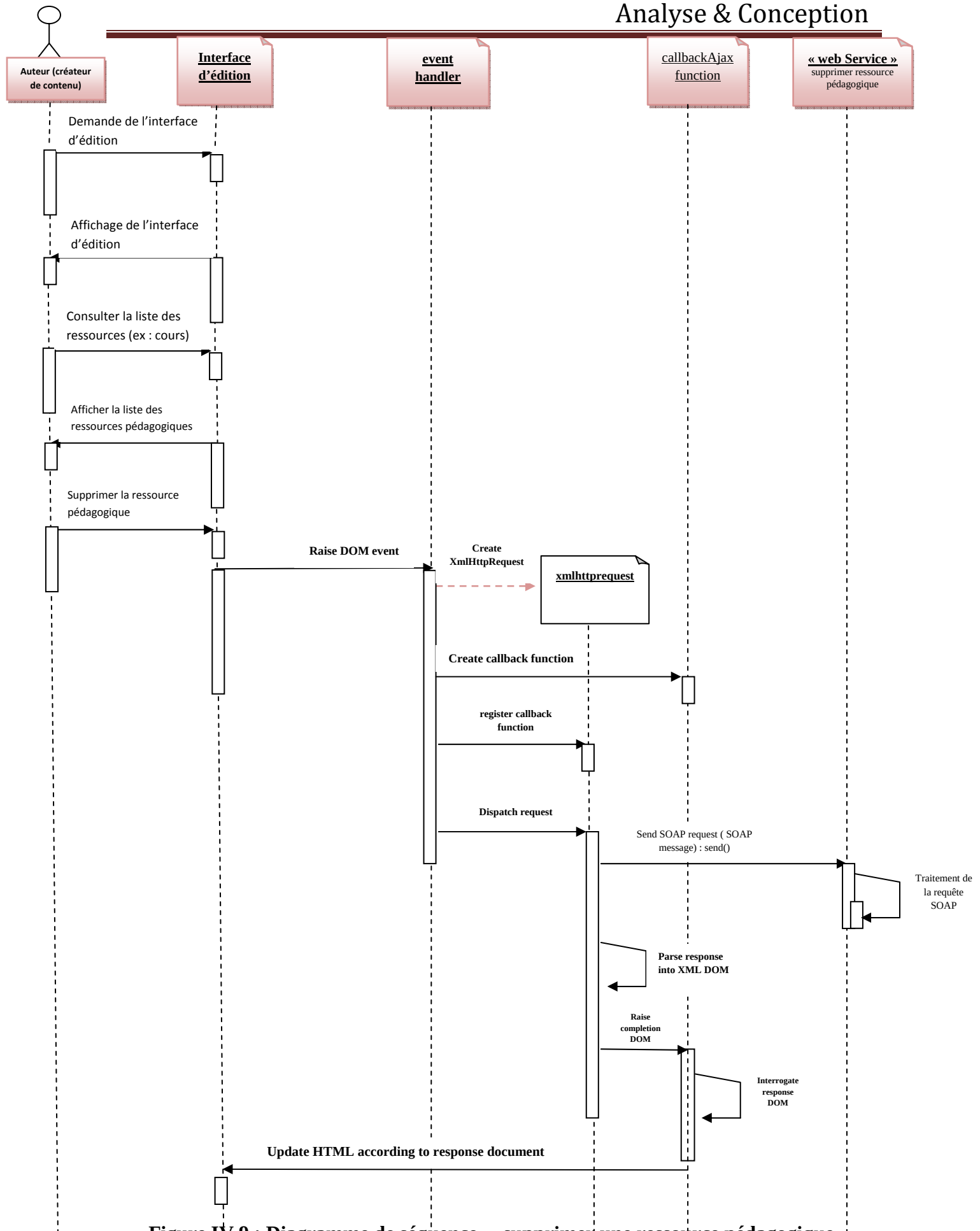


Figure IV.9 : Diagramme de séquence « supprimer une ressource pédagogique »

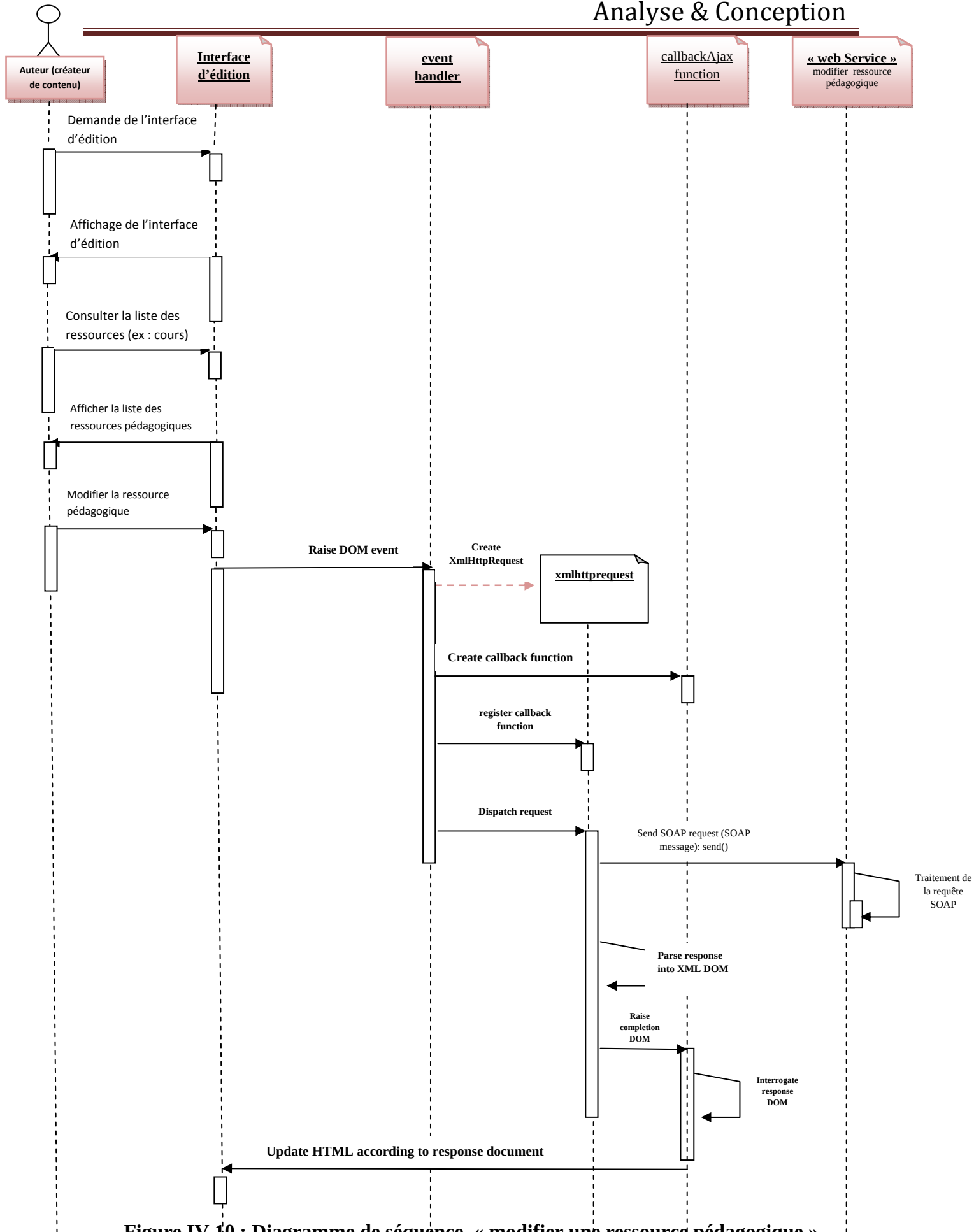


Figure IV.10 : Diagramme de séquence « modifier une ressource pédagogique »

➤ Diagramme de séquence « mise à jour du flux rss »

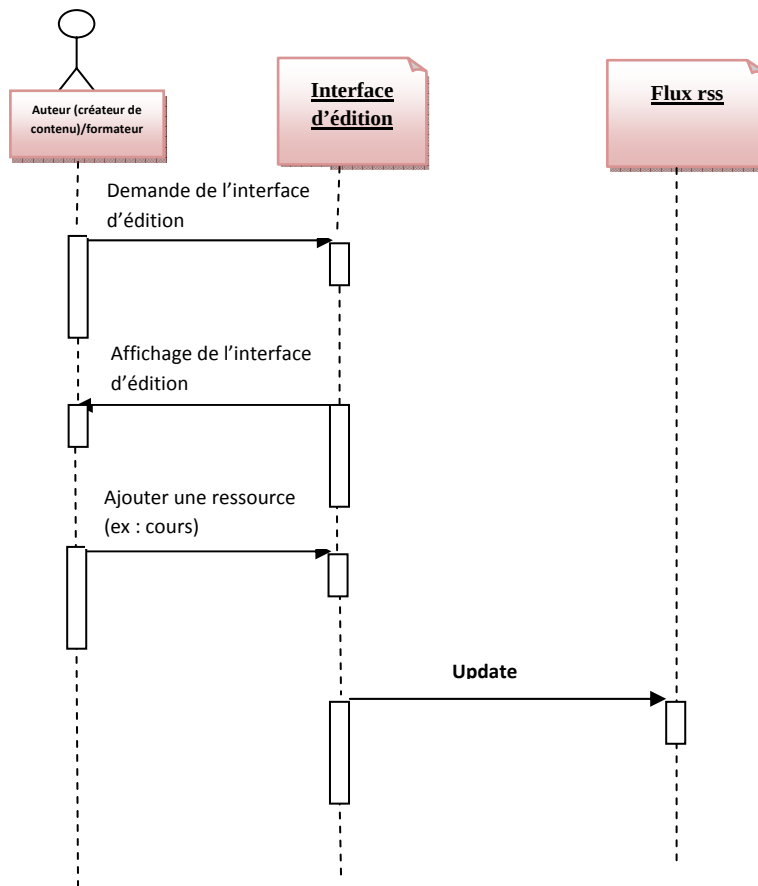


Figure IV.11: Diagramme de séquence « mise à jour du flux rss »

II.2.4 Diagramme de classe

Les diagrammes de classes expriment de manière générale la structure statique d'un système en termes de classes et de relations entre ces classes. De même qu'une classe décrit un ensemble d'objets, une association décrit un ensemble de liens ; les objets sont instances de classes et les liens sont instances des relations.



III. Publication et découverte des services web

L'intérêt de développer une infrastructure pour la découverte de services liés aux ressources pédagogiques est manifeste. Celle-ci permettra, grâce aux moteurs de services publics, de retrouver les ressources pédagogiques. Elle permettra également de retrouver les plates formes d'elearning disponibles dès lors que celles-ci sont décrites par des formalismes appropriés (WSDL, OWL-S ou tout autre langage standardisé). En outre, les ressources décrites sous forme de services web peuvent être intégrées dans n'importe quel système d'e-learning orienté services. [21]

III.1 Services Web et e-Learning

De nos jours les environnements d'e-learning sont distribués et basés sur des architectures orientées services. Les composants et ressources de ces systèmes sont implémentés à l'aide de services web. Cependant leur exploitation pose des problèmes de découverte et de localisation des services qui répondent le mieux aux besoins et spécificités des différents usagers. Les premières approches de découverte de services proposées dans la littérature étaient des approches syntaxiques. Avec l'émergence de la nouvelle génération du web, des approches sémantiques ont vu le jour. Celles-ci présentent une description sémantique de services web interprétables par une application dans le but d'assurer un degré d'automatisation élevé. Ceci a donné naissance à une nouvelle génération de langages pour le web, tels que RDF, OWL. Ceci a permis de développer des outils pour la description sémantique de services web comme OWL-S. Dans le cas des services pédagogiques, l'annuaire UDDI n'est pas approprié pour leur découverte car il ne permet pas le stockage des informations relatives à la sémantique du contenu pédagogique, d'où la nécessité d'une approche sémantique pour la description de ces services tel que OWL-S. Cependant OWL-S ne prend en considération ni l'aspect pédagogique du service, ni les besoins et spécificités du demandeur de service. Pour cela le développement d'une approche de découverte de services adaptés aux besoins et spécificités des usagers est nécessaire.

Celle-ci est basée sur deux ontologies :

- Une ontologie associée à la pédagogie liée aux ressources pédagogiques qui donne une description sémantique et pédagogique des services web associés aux ressources pédagogiques,
- Une ontologie du domaine à enseigner relative au domaine d'apprentissage. [21]

III.2 Approche de Découverte de Services Web Liés aux Ressources Pédagogiques

Cette approche est basée sur :

- La définition d'une ontologie qui est OWL-S+. OWL-S+ est une sous ontologie dans le « ServiceProfile » de l'ontologie des services web OWL-S, qui permet une description sémantique et pédagogique des services et qui assure le lien avec l'ontologie du domaine à enseigner.
- La définition de l'ontologie du domaine à enseigner qui précise les concepts du domaine d'apprentissage et les relations sémantiques qui existent entre eux. [32]

III.2.1 Ontologie des services web OWL-S

OWL-S est un langage de description de services basé sur XML utilisant le modèle des logiques de descriptions (une proposition d'ontologie des services Web). Il est aussi un langage de haut niveau pour la description et l'invocation des services Web dans lequel la sémantique est incluse et interface avec UDDI/WSDL/SOAP. [27]

Parmi les objectifs de OWL-S on peut citer :

- ✓ Découverte automatique des services par des agents
- ✓ Invocation automatique : détection et interprétation automatique des paramètres d'entrée sortie nécessaires à chaque service Web ainsi que les moyens de l'appeler.

OWL-S organise une description de service en quatre secteurs conceptuels: le «service model», le «profile», le «grounding», et le «service». [14] comme le montre la figure suivante :

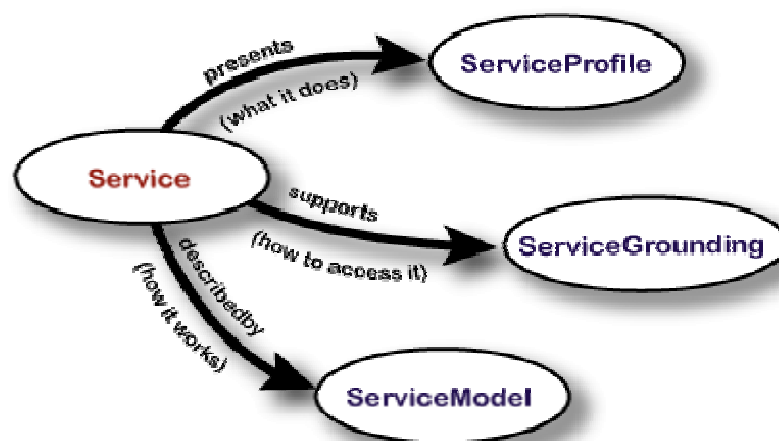


Figure IV.13: Les classes principales de l'ontologie OWL-S

- **Service model** : décrit comment un service accomplit ses tâches.
- **Profile** : fournit une description générale de Web Service pour qu'il puisse être publié et partagé pour faciliter la découverte de service. Les profils peuvent inclure non seulement les propriétés fonctionnelles (entrées, sorties, conditions préalables, et résultats) mais aussi les propriétés non fonctionnelles (nom de service, description des textes, information de contact, catégorie de service, et paramètres additionnels de service).
- **Grounding** : indique comment un service est appelé, en détaillant comment les processus atomiques dans le modèle de processus d'un service trace sur un protocole concret de transmission de messages. OWL-S a différents types de ServiceGrounding à utiliser, mais le seul type développé jusqu'ici est le WSDL grounding, qui permet à tous les Web services d'être marqués comme un Web Service Sémantique en utilisant OWL-S.
- **Service** : un service lie toutes les autres parts dans une seule unité qui peut être publié et appelée.

III.2.2 Ontologie Associée à la Pédagogie Liée aux Ressources Pédagogiques

Elle est définie comme une sous ontologie de l'ontologie "ServiceProfile". L'ontologie résultante baptisée OWL S+ utilise l'ontologie OWL-S et l'ontologie associée à la pédagogie liée aux ressources pédagogiques. Ses classes donnent une description du contenu pédagogique et sémantique du service ainsi que les caractéristiques techniques et financières jugées nécessaires pour la description des ressources pédagogiques. L'aspect technique englobe les informations concernant le matériel, la sécurité, la collaboration, la taille, le format et la durée d'apprentissage. L'aspect financier contient les sous-classes : coût de la ressource et ses droits d'utilisation. L'aspect pédagogique de la ressource comporte les classes suivantes :

- *Domaine*. Décrit les informations relatives au domaine d'apprentissage tels que :
 - Nom domaine : désignation du domaine d'apprentissage.
 - Description : description textuelle du domaine d'apprentissage.
- *Titre*. Indique le titre de la ressource
- *Description*. Donne une description textuelle de la ressource.
- *Langue*. Langue dans laquelle la ressource est décrite.
- *Concept*. Indique le concept enseigné par la ressource.

- *Type*. Décrit le type pédagogique de la ressource qui peut être une activité d'apprentissage, un scénario, un module d'apprentissage, un exercice, un test d'évaluation, un logiciel spécifique, une plate forme d'e-learning ...
- *Éléments connaissance concept*. Liste des éléments de connaissance associés au concept.
- *Poids concept*. Densité sémantique du concept dans la ressource pédagogique.
- *Concept pré-requis*. Définit l'ensemble des concepts pré-requis pour l'utilisation de la ressource pédagogique.
- *Élément pré-requis*. Définit l'ensemble des éléments de connaissance pré-requis nécessaires pour l'acquisition du concept enseigné par la ressource.
- *Acteur*. Décrit l'utilisateur destinataire de la ressource. C'est la superclasse des classes suivantes :
 - *Rôle* : Le rôle de l'utilisateur qui peut être selon les cas un apprenant, un formateur, un auteur-concepteur ou un administrateur.
 - *Profil initial* : Définit l'ensemble des concepts et éléments de connaissance qui doivent être déjà acquis par l'acteur apprenant.
 - *Profil final* : Définit l'ensemble des concepts et éléments de connaissance qui seront acquis par l'apprenant après l'utilisation de la ressource. [21]

III.2.3 Ontologie du Domaine à Enseigner

Une ontologie de domaine définit un vocabulaire commun pour les différents acteurs dans un domaine donné

Figure IV.14: Ontologie du domaine à enseigner [21]

III.2.4 Fonctionnalités de l'Infrastructure de Découverte de Services Web Liés aux Ressources Pédagogiques

L'architecture de découverte de services web liés aux ressources pédagogiques offre les deux principales fonctionnalités suivantes :

- La publication des ressources pédagogiques sous forme de services web dotés d'une description pédagogique et sémantique.
- La découverte et l'invocation des services web liés aux ressources pédagogiques qui répondent le mieux aux besoins et spécificités des différents usagers.

a) Publication des Services Web : Cette fonctionnalité permet à l'enseignant formateur de publier son cours sous forme de services web, et à l'administrateur-gestionnaire de publier les contenus des formations en ligne sous forme de services web internes (i.e. accessibles par les apprenants et les usagers inscrits) ou externes (i.e. utilisables par d'autres plates formes d'e-learning). En publiant un service, le fournisseur de services fournit les informations qui présentent le service ainsi qu'une description conforme au format défini dans l'ontologie associée à la pédagogie liée aux ressources pédagogiques. Le système crée alors une description WSDL contenant les informations nécessaires pour l'invocation de ce service et fait son enregistrement dans l'annuaire UDDI. Les informations relatives au service ainsi que celles en rapport avec le fournisseur de service sont enregistrées comme instances de l'ontologie OWL-S+. Toute ressource pédagogique aura ainsi une représentation dans l'ontologie OWL-S+. [21]

b) Découverte des Services Web. Cette fonctionnalité permet aux enseignants formateurs de consulter ou d'utiliser des ressources pédagogiques, et aux apprenants de trouver des ressources internes ou externes en rapport avec leurs activités d'apprentissage. La découverte des services web est basée ici sur deux ontologies : l'ontologie OWL-S+ et l'ontologie du domaine à enseigner permettant d'exploiter la couverture sémantique des concepts du domaine d'apprentissage. Le système détermine les classes à utiliser lors de la recherche en identifiant dans la requête introduite par l'utilisateur les concepts utilisés ainsi que les autres aspects de la recherche. [21]

La figure ci-dessous illustre l'architecture générale de découverte de services web liés aux ressources pédagogique

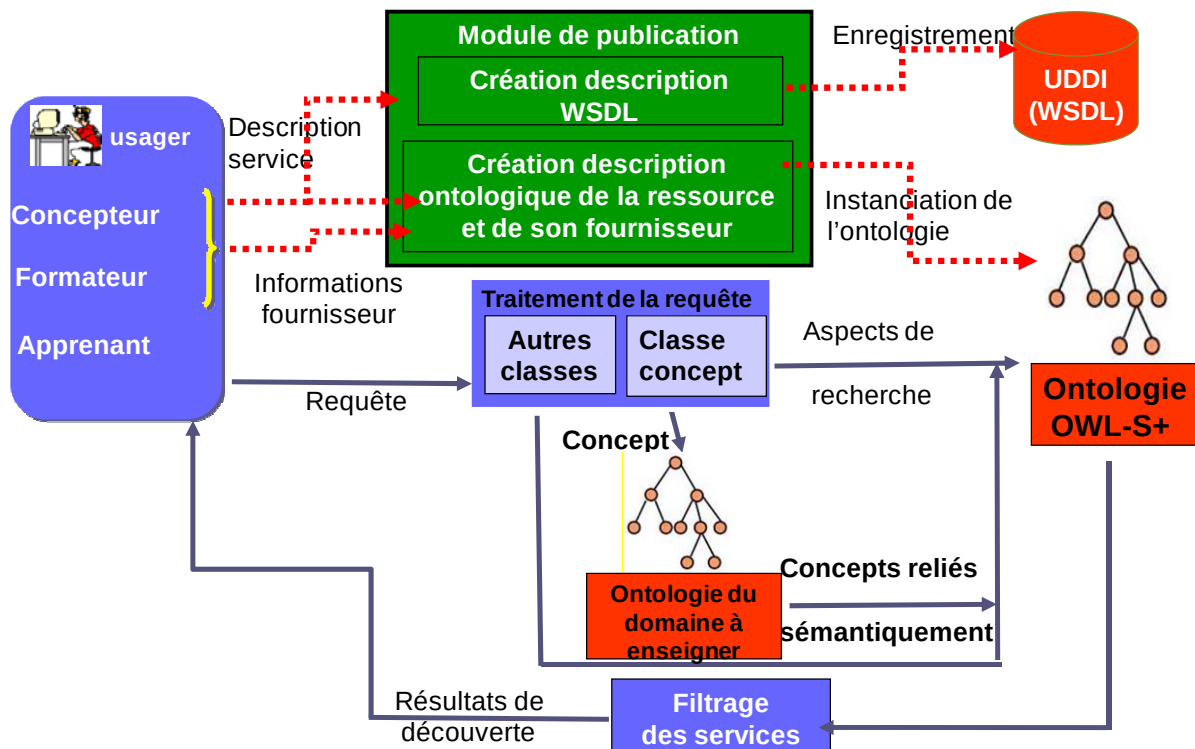


Figure IV.15: Architecture générale de découverte de services web [21]

La découverte de services consiste à trouver dans l'ontologie OWL-S+ les instances qui répondent directement aux aspects de la recherche et l'utilisation de l'ontologie du domaine à enseigner pour trouver les concepts liés sémantiquement à la classe concept considéré dans le but de les utiliser un à un comme instance de la classe concept (en gardant les mêmes autres aspects de la recherche introduits dans la requête) pour découvrir éventuellement d'autres services. Un filtrage est effectué sur les résultats de la découverte selon la densité sémantique des concepts associés aux ressources trouvées. Ce mécanisme permet ainsi de sélectionner les services qui répondent le mieux aux besoins et spécificités des usagers.

Conclusion

Dans ce chapitre, nous avons présenté en premier lieu une étude de l'existant pour le traitement des problèmes des plateformes existantes. Dans un second lieu nous avons fait une spécification des besoins en dégagant les cas d'utilisations correspondants puis nous avons entamé directement la phase de conception en présentant quelques diagrammes UML. Nous allons présenter dans le chapitre qui suit les différents outils utilisés et l'environnement de développement de notre application.

Chapitre V

Réalisation

Introduction

Une des étapes de la vie d'un projet, aussi importante que la conception est l'implémentation. Cette étape constitue la phase d'achèvement et d'aboutissement du projet. Pour accomplir cette tâche avec succès il faut savoir utiliser les outils adéquats et nécessaires. Ce choix d'outils peut influencer sur la qualité du produit obtenu et donc nécessite une attention particulière et doit se baser sur les besoins du projet et le résultat escompté. Ce chapitre présente alors l'environnement technique du travail ainsi que le choix pris en matière d'environnement logiciel.

I. Environnement de développement

Un diagramme de déploiement spécifie un ensemble de constructions qui peut être utilisé pour définir l'architecture d'exécution de systèmes. Le diagramme de déploiement permet de représenter l'environnement de développement de notre application.

Notre solution se base sur une architecture client/ serveur à trois tiers (trois niveaux) qui sont :

- ✚ Le premier niveau de cette architecture qui est le niveau présentation est constitué d'un navigateur Web.
- ✚ Le deuxième est le niveau applicatif (logique applicative) qui est pris en charge par le serveur Apache, et qui se compose de scripts écrits en PHP.
- ✚ Le troisième niveau, qui fournit au niveau intermédiaire les données dont il a besoin, est pris en charge dans notre cas par le SGBD MYSQL.

L'implémentation de notre système se base sur l'architecture illustrée dans la figure ci-dessous :

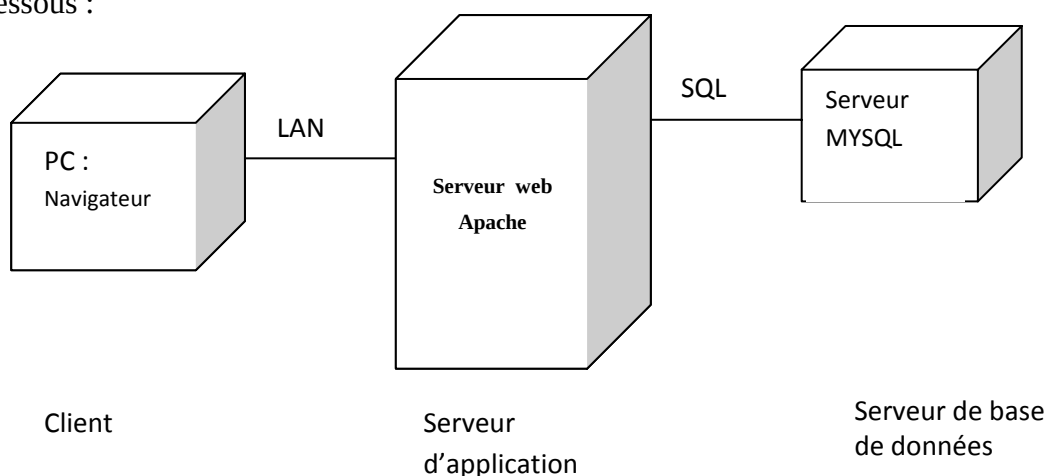


Figure V.1 : Diagramme de déploiement

I.1. Serveur Web Apache :

Tout développement de site Web requiert un serveur Web qui s'occupe du traitement des requêtes des clients, le transfert des pages HTML au browser et l'exécution des programmes sur la machine serveur.

Nous avons utilisé le serveur Web Apache, du fait qu'il est le serveur Web le plus utilisé, présentant un niveau de performances élevé pour des exigences matérielles modestes, en plus de sa gratuité et sa robustesse. [28]

I.2. Serveur de base de données MySQL :

MySQL est un système de gestion de base de données. Son rôle est de stocker et de gérer une grande quantité de données en les organisant sous forme de tables, et de permettre la manipulation de ces données à travers le langage de requête SQL, il est implémenté sur un mode client serveur avec du côté serveur : le serveur MySQL, et du côté client : les différents programmes et librairies.

MySQL se caractérise par sa rapidité et sa facilité d'utilisation, il est offert avec l'outil d'administration de base de données « phpmyadmin » par wamp server qui est un package d'installation facile de PHP, d'Apache et de MySQL.

I.3. Interface PHPMyAdmin

PHPMyAdmin est une application web qui permet de gérer un serveur de bases de données MySQL. Dans un environnement multiutilisateur, cette interface écrite en PHP permet également de donner à un utilisateur un accès à ses propres bases de données.

La figure suivante illustre l'interface de gestion des bases de données de MySQL PhpMyAdmin

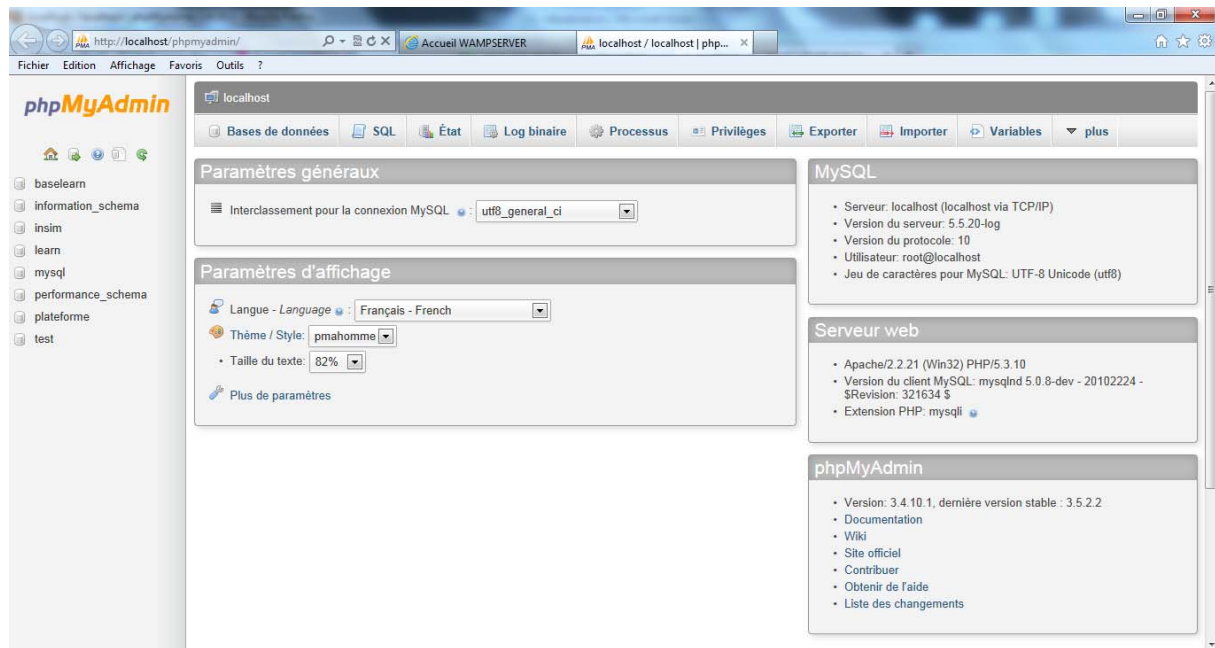


Figure V.2 : Administration des bases de données à partir de PhpMyAdmin

I.4 Le système d'exploitation Windows7

Windows 7 est le dernier en date des systèmes d'exploitation de la société Microsoft, sorti le 22 octobre 2009 et successeur de Windows Vista.

Cette nouvelle version de Windows reprend l'acquis de Windows Vista tout en apportant de nombreuses modifications, notamment par divers changements au niveau de l'interface et de l'ergonomie générale, un effort particulier pour la gestion transparente des machines mobiles et le souci d'améliorer les performances globales du système par rapport à son prédécesseur.

II. Les langages de programmation

II.1 PHP5 et son API intégrée SOAP

PHP est un langage de script libre principalement utilisé pour produire des pages web dynamiques via un serveur HTTP, mais pouvant également fonctionner comme n'importe quel langage interprété de façon locale, en exécutant les programmes en ligne de commande. PHP est un langage impératif disposant depuis la version 5 (php5) de fonctionnalités de modèle objet complètes et de programmation des services web soap en utilisant L'extension **php_soap** apparue avec PHP 5 qu'il suffit juste d'activer.

PHP5 intègre d'origine une API SOAP, permettant de créer des serveurs ou des clients, et ce de façon très simplifiée. En effet, PHP5 interprète de façon transparente les fichiers WSDL lors de la déclaration d'un client à un Web Service, et gère lui-même le formatage SOAP des messages envoyés au serveur (formatage XML, respect des règles du WSDL....).

Le serveur SOAP de PHP propose les mêmes facilités que pour le client, hormis pour la description WSDL qui doit être réalisé par un autre biais (dans notre cas on a utilisé zend studio pour la génération du WSDL). Dans ce cas, le serveur « sert » le Service WEB selon les spécificités décrites dans le WSDL....

Devant la diversité de produits de développement coté serveur, on peut se demander sûrement pourquoi PHP. La réponse est simple : Il intègre tous les avantages des autres solutions. Son code est rapide à produire et à exécuter. Il tourne sur différents serveurs (IIS, Apache, ...) et différents systèmes d'exploitation (UNIX, Windows et Macintosh). Il est gratuit. Il génère des fichiers PDF, DOC, HTML ou des images GIF, JPEG,...etc.

- **Portabilité**

Le programme PHP est indépendant des plates-formes (Linux, Unix, Windows, et MAC). Il tourne aussi sur différents serveurs (Apache, Microsoft IIS aussi, JRUN, etc.).

- **Efficacité**

Les scripts PHP offrent une méthode efficace de gestion des requêtes utilisateurs.

- **Réutilisation**

La réutilisation du code est le but de tout programmeur. La création de composants d'une application est l'une des méthodes de réutilisation de code, l'utilisation d'objets pour encapsuler des fonctionnalités partagées. PHP utilise ces deux méthodes. Il s'agit d'un langage orienté objet. C'est pour cela qu'il intègre la réutilisation de code.

- **Modularité**

Lors de développement d'une application complète coté serveur, les programmes peuvent rapidement gagner en volume et en complexité. Il vaut toujours mieux de diviser une application en différents modules chacun étant chargé d'une tâche définie. Ainsi, l'application est beaucoup plus facile à gérer et à comprendre. Les scripts PHP et les classes permettent de séparer l'application en différents modules, de la décomposer en niveaux et en tâches.

II.2 Langage HTML

Le langage HTML est un langage permettant de décrire la mise en page et la forme d'un contenu rédigé en texte simple.

Une page HTML est ainsi un simple fichier texte contenant des balises (parfois appelées marqueurs ou repères ou tags en anglais) permettant de mettre en forme le texte, les images, etc.

II.3 Le langage de requêtes « SQL (Structured Query Language) »

C'est un langage de manipulation de base de données mis au point dans les années 70 par IBM. Il permet notamment :

- La manipulation des tables : création, suppression, modification de la structure des tables.
- La manipulation des bases de données : sélection, modification et suppression d'enregistrements.
- La gestion des droits d'accès aux tables : contrôles des données et validation des modifications.

II.4 JavaScript

JavaScript est un langage de script très populaire, il permet de réaliser des traitements évolués dans les pages Web en offrant la possibilité d'utiliser les différents standards Web supportés par les navigateurs et d'interagir avec eux. Certaines spécificités de ces derniers restent néanmoins à prendre en compte afin de garantir leur portabilité.

Longtemps freiné par les incompatibilités entre les différentes implémentations des navigateurs, ce langage est devenu le fondement des applications Web 2.0. Il permet désormais de définir des traitements évolués en leur sein afin de gérer notamment l'interaction avec les utilisateurs et les échanges de données avec des applications ou des services accessibles par le biais d'Internet.

De solides connaissances sont nécessaires pour mettre en œuvre ce langage dans des applications de type Web 2.0, dont l'interface graphique est plus évoluée et interactive. Heureusement, des bibliothèques JavaScript ont été mises au point afin d'aider au développement de ce type d'application Web.

II.5 CSS (*Cascading Style Sheets*)

Les feuilles de styles simplifient le balisage HTML et soulagent grandement HTML des responsabilités de la représentation. Elles donnent le contrôle de la présentation des documents, informations sur les polices de caractères, alignement, couleurs etc. l'objectif est de tendre vers la séparation du contenu et de la présentation.

Les informations de styles peuvent être spécifiées pour un élément ponctuel ou pour des groupes d'éléments. Elles peuvent se trouver à l'intérieur du document HTML ou dans une feuille de style externe. Les mécanismes qui affectent une feuille de style à un document sont indépendants du langage de feuille de style.

Un fichier de feuille de style est un fichier dont l'extension est .css. Dans ce fichier, il est possible de définir les propriétés de différentes manières. Si l'identifiant est une classe, il est précédé d'un point, si c'est un identifiant unique dans le fichier XHTML, il est précédé d'un dièse. Dans le cas des propriétés attribuées à une balise, il suffit d'utiliser le nom de la balise seul. Les propriétés de la balise sont encadrées par des accolades et définies par la syntaxe suivante : Propriété : valeur ;

```
Balise{  
  Color: #FFFFFF ;}  
.classe{  
  Colo: #FFFFFF ;}  
#id{  
  Color: #FFFFFF}
```

II.6 XML (*eXtensible Markup Language*)

Recommandation du W3C depuis le 10 février 1998, le langage eXtensible Markup Language (XML) connaît depuis ses débuts un succès indéniable. Défini dès son origine comme un métalangage facilitant l'élaboration de langages à balises spécialisés, XML a conquis depuis de multiples formats, les nouveaux comme les anciens. Les normes strictes qui gèrent la syntaxe et la structure de XML rendent le langage et son utilisation plus aisés, favorisant notamment la factorisation des travaux de développement d'analyseurs syntaxiques. XML fournit donc un cadre de structuration des données qui peut être employé pour la définition rapide et sans ambiguïté syntaxique d'un format de document. [29]

II.7 Ajax (Asynchronous JavaScript And XML)

L'acronyme AJAX signifie **Asynchronous JavaScript And XML**. Il résume très bien l'enrichissement technique qu'apporte AJAX. Il est basé sur le modèle de développement des applications web actuelles. Ces applications utilisent le protocole HTTP, HTML et par extension XML.

JavaScript n'était jusque là qu'un langage de script chargé d'étendre les possibilités des pages web côté client.

AJAX enrichit ce JavaScript d'une nouvelle technologie : `xmlhttprequest`. Cet objet est la véritable évolution qui en entraîne d'autres. Avec AJAX, JavaScript revient à la mode, car couplé avec l'objet `xmlhttprequest`, ces possibilités redeviennent intéressantes.

La présence de XML n'est pas une surprise : XHTML, l'évolution XML du HTML, est un standard du web. Le DOM spécifie la manipulation de la structure sémantique d'une page, c'est à dire des balises XML. Le Web tend de plus en plus à utiliser XML : il est donc logique que l'objet chargé de communiquer avec le serveur reçoive les données sous forme XML. De plus, le traitement du flux par JavaScript sera facilité en utilisant le DOM. Cependant, le serveur peut aussi renvoyer des données sous forme HTML ou textuelle.

`XMLHTTPREQUEST` est un objet JavaScript qui permet d'établir une communication HTTP avec le serveur, en échangeant des données XML. Il existe un mode asynchrone pour l'objet. Ainsi, les traitements ne sont pas bloqués par l'attente du résultat. C'est appréciable si le flux de la requête est important. On peut donc utiliser l'objet pour effectuer des requêtes HTTP. La réponse du serveur sera lue par l'objet, et on pourra la traiter avec du JavaScript classique.

Ainsi, AJAX ajoute une couche technologique supplémentaire aux pages Web :

- ✓ HTML pour la structure sémantique des données.
- ✓ CSS pour la présentation des données.
- ✓ DOM (Document Object Model) pour la manipulation dynamique des données est une collection d'objets où chaque objet représente un élément structurel ou visuel d'une page web ou d'un document XML. Il est utilisé par JavaScript pour inspecter et modifier le contenu des pages Web.
- ✓ `XMLHttpRequest` pour échanger des données de manière asynchrone avec le serveur.

III. Les outils de développement

III.1 Wamp server (Windows Apache MySQL, PHP5)

WampServer est une plate-forme de développement Web sous Windows. Il permet de développer des applications Web dynamiques à l'aide du serveur Apache, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMyAdmin pour gérer plus facilement les bases de données.

Contrairement aux autres solutions, WampServer permet de reproduire fidèlement un serveur de production. Une fois la base installée, il est possible d'ajouter autant de versions d'Apache, MySQL et PHP que l'utilisateur souhaite.

WampServer dispose également d'un "TrayIcon" permettant de gérer et configurer simplement les serveurs, sans toucher aux fichiers de configuration.

Les fonctionnalités de WampServer sont très complètes et très intuitives, nous ne détaillerons donc pas ici leur utilisation.

Via un click gauche sur l'icône de WampServer, il est possible de :

- ✓ gérer les services d'Apache et MySQL
- ✓ passer en mode online/offline (accessible à tous ou limité à localhost)
- ✓ installer et changer de version d'Apache, MySQL et PHP
- ✓ gérer les paramètres de configuration des serveurs
- ✓ accéder aux logs
- ✓ accéder aux fichiers de configuration
- ✓ créer des alias

Via un click droit :

- ✓ changer la langue du menu de WampServer
- ✓ accéder directement à cette page

WampServer offre de plus la possibilité d'installer pratiquement toutes les versions d'Apache, PHP et MySQL existantes, de quoi reproduire fidèlement la configuration de votre serveur de production.

La figure suivante illustre l'interface principale de WampServer

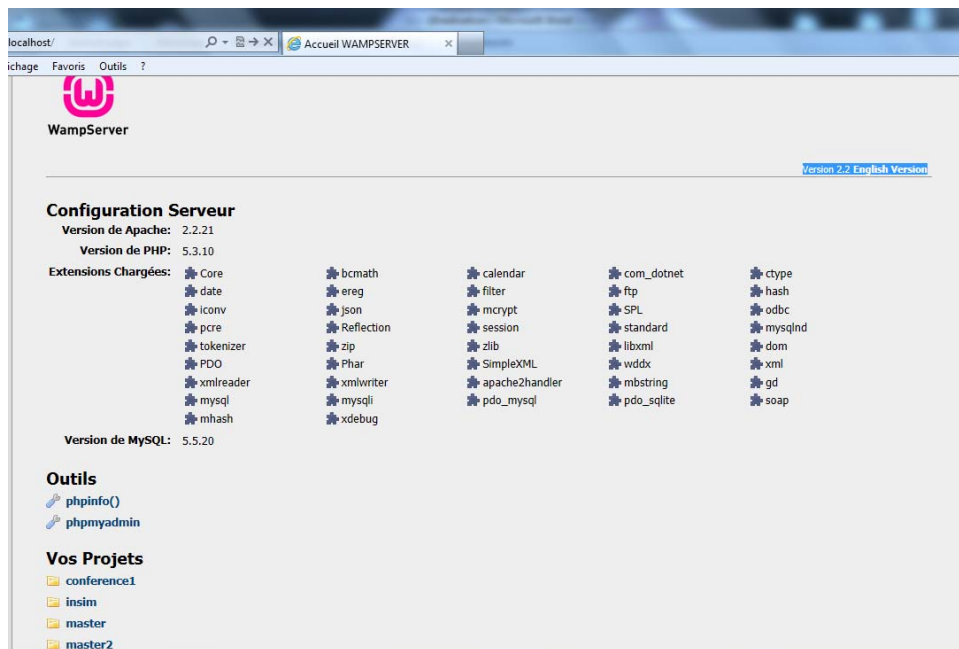


Figure V.3: Interface principale de WampServer

III.2 Dreamweaver® 8

Logiciel éditeur de texte créé par Macromedia (géré maintenant par Adobe) permettant la conception de sites web. Dreamweaver fonctionne en mode WYSIWYG (What You See Is What You Get), signifiant littéralement en français « ce que vous voyez est ce que vous obtenez » c'est-à-dire permet de composer visuellement le résultat voulu. Ce logiciel est très puissant d'utilisation, la partie Editeur de code du logiciel permet la modification rapide du contenu des pages, de leur mise en forme, de l'insertion d'images, de liens ou d'autres éléments habillant la page Web. La partie Gestion du site permet d'organiser tous les fichiers composant le site et de les publier sur le serveur distant pour qu'ils soient accessibles via Internet.[30]

L'interface du logiciel Macromedia Dreamweaver est illustrée dans la figure suivante :



Figure V.4: Espace de travail de Dreamweaver

III.3 Zendstudio

Zend Studio est le seul IDE (Integrated Development Environment - Environnement de développement intégré) conçu pour des développeurs professionnels et comprenant tous les composants de développement nécessaires au cycle de vie complet d'une application PHP. Grâce à un jeu complet d'outils d'édition, de débogage, d'analyse, de tests, d'optimisation et de base de données, Zend Studio accélère les cycles de développement et simplifie les projets complexes.

Zend Server est un serveur d'application web pour entreprise qui permet d'exécuter et de gérer des applications PHP exigeant un niveau de fiabilité et de sécurité élevé. Zend Server comprend une pile PHP complète et entièrement testée qui réduit considérablement le temps habituellement nécessaire au suivi, à l'installation, à la configuration et aux tests des bibliothèques et des pilotes PHP. Un cache et une accélération intégrées garantissent que vos applications PHP offrent un niveau de performance maximal et les fonctions intégrées de surveillance et de diagnostic vous permettent de détecter, d'isoler et de résoudre rapidement les pannes et les goulots d'étranglement.

Zend Studio et Zend Server sont conçus pour travailler ensemble, simplifiant ainsi le processus de développement. À partir de Zend Studio, les scripts PHP peuvent être déployés

sur un serveur local Zend. Puis, une fois le déploiement effectué, la fonction de surveillance de Zend Server capture les informations de débogage et d'analyse de l'origine des défaillances pour les utiliser dans Zend Studio.

Le fichier WSDL est en XML, et permet de décrire l'utilisation complète du service. C'est en fait un mode d'emploi, ni plus ni moins, et il permettra à de nombreux programmes ou même langages de programmations de découvrir automatiquement un service, et/ou de gérer les modalités d'échanges SOAP (port d'échange, protocole, encodage, etc), on pourrait presque le comparer à la DTD d'un fichier XML. Zend Studio et DIA permettent la réalisation assisté d'un tel fichier, sinon on peut se servir des spécifications W3C pour l'écrire à la main.

L'interface suivante illustre la génération d'un WSDL en utilisant ZendStudio

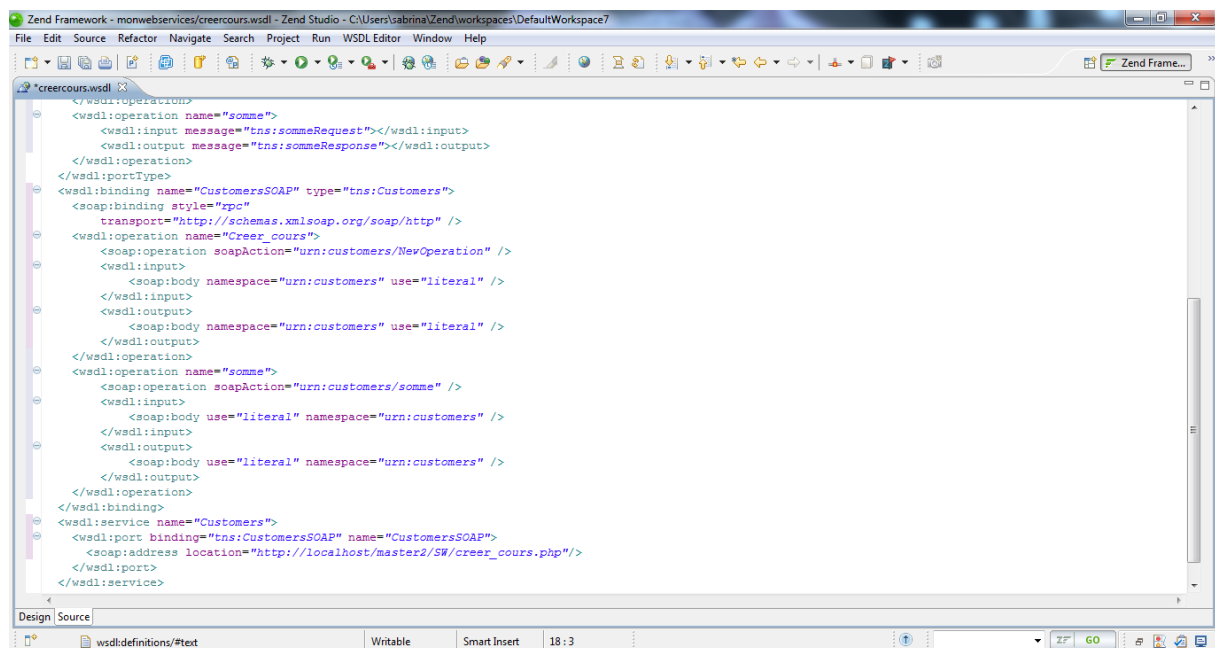


Figure V.5: Interface ZendStudio pour la génération des WSDL

IV. Présentation de quelques interfaces

IV.1 La page d'accueil de la plateforme



Figure V.6 : page d'accueil de la plateforme

IV.2 La page des préinscriptions



Figure V.7 : page des préinscriptions

IV.3 Page d'authentification



Figure V.8 : page d'authentification

IV.4 L'interface d'édition de ressources pédagogiques (espace auteur) contenant les technologies du web 2.0



Figure V.9 : Interface d'édition

IV.5 page d'ajout de ressources pédagogique « cours »

Figure V.10 : interface de publication de cours

IV.6 Page de consultation des ressources afin de les modifier ou supprimer « cours »

Figure V.11 : page de consultation des cours

IV.7 Page « espace personnel de l'apprenant »

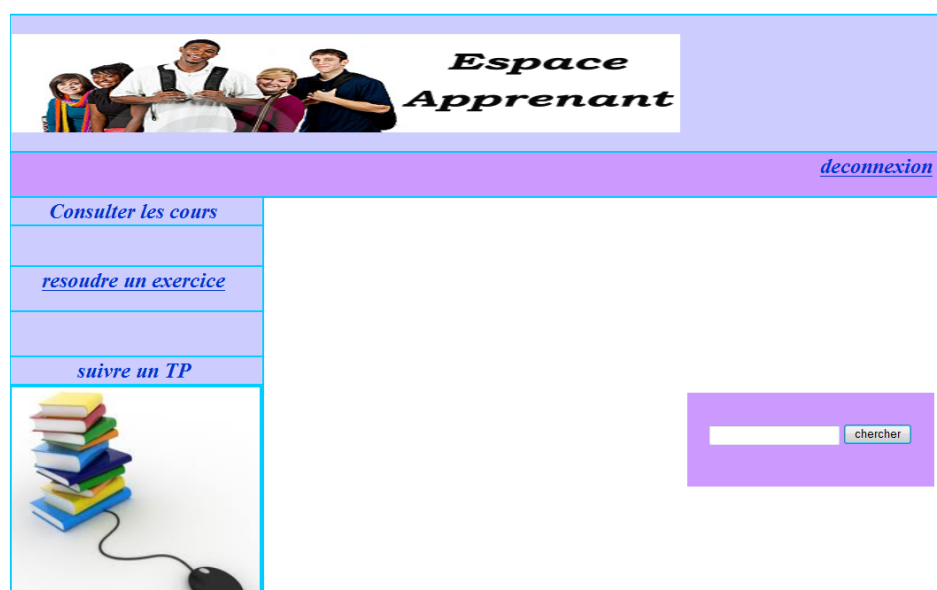


Figure V.12 : page « espace apprenant »

IV.8 Page « espace personnel du formateur »



Figure V.13: page « espace formateur »

IV.9 Page « espace personnel de l'administrateur »



Figure V.14 : Page « espace administrateur »

Conclusion

Dans ce chapitre nous avons en premier lieu présenté les différents outils de développement, les langages et les logiciels utilisés pour l'implémentation de l'application. Puis nous avons fait une illustration de quelques interfaces.

conclusion générale

Conclusion générale

L'objectif de notre projet de fin d'étude était d'intégrer les technologies web 2.0 pour l'extension des CMS (Content Management System) tout en exploitant les web services qui sont une instance de l'architecture orientée service (SOA). Le CMS que nous avons développé dans le cadre de notre projet permet la publication en ligne d'un ensemble de ressources pédagogiques que les utilisateurs peuvent diffuser, consulter et modifier

Le point de départ de la réalisation de ce projet était une récolte des informations nécessaires pour dresser un état de l'existant et présenter un aperçu sur la problématique. Par la suite, nous nous sommes intéressés à l'analyse et la spécification des besoins qui nous ont permis de distinguer les différents acteurs interagissant avec l'application visée. Puis la phase de conception dans laquelle nous avons fixé la structure globale de l'application.

Le dernier volet de notre projet était la partie réalisation qui a été consacrée à la présentation des outils de travail et des interfaces les plus significatives de l'application.

L'apport de ce travail a été d'une importance très considérable, en effet, il nous a permis de suivre une méthodologie de travail bien étudiée, d'approfondir nos connaissances dans plusieurs domaines tels que le e-learning et les technologies du web telles que le web 2.0, les web services ainsi que le web sémantique découvert notamment.

Comme perspective, nous envisageons de faire la découverte des web services de manière sémantique en utilisant les ontologies des web services.

Annexe

Introduction

La modélisation par objet saisit la structure statique d'un système, en montrant les objets dans le système, leurs relations, ainsi que les attributs et les opérations qui caractérisent chaque classe. Ce mode de conception de système prend son intérêt par rapport à la programmation traditionnelle puisque le modèle objet correspond étroitement au monde réel.

Il est par conséquent plus souple pour les modifications. UML (*unified modeling language*, que l'on peut traduire par “*langage de modélisation unifié*”) est une notation permettant de modéliser un problème de façon standard.

I. Les briques de base d'UML

La terminologie d'UML inclut trois sortes de briques :

- ✓ Des éléments.
- ✓ Des relations.
- ✓ Des diagrammes.

I.1 Eléments d'UML

I.1.1 Les éléments structurels

Les éléments structurels sont représentés par des noms dans les modèles UML. Ce sont les parties les plus statiques d'un modèle : ils représentent des éléments conceptuels ou physiques.

La classe

Elle représente un ensemble d'éléments qui partagent les mêmes attributs, les mêmes opérations, les mêmes relations et les mêmes sémantiques.

Classe1
-attribut1 : int = 3 -attribut2 : byte = 4
+opération1() : bool +opération2() : Classe1

Figure A1 : classes

✚ Les classes-associations

Il est possible de représenter une association par une classe pour ajouter, par exemple, des attributs et des opérations dans l'association. Une classe de ce type, appelée **classe associative** ou **classe-association**, possède à la fois les caractéristiques d'une classe et d'une association, et peut à ce titre participer dans d'autres relations dans le modèle.

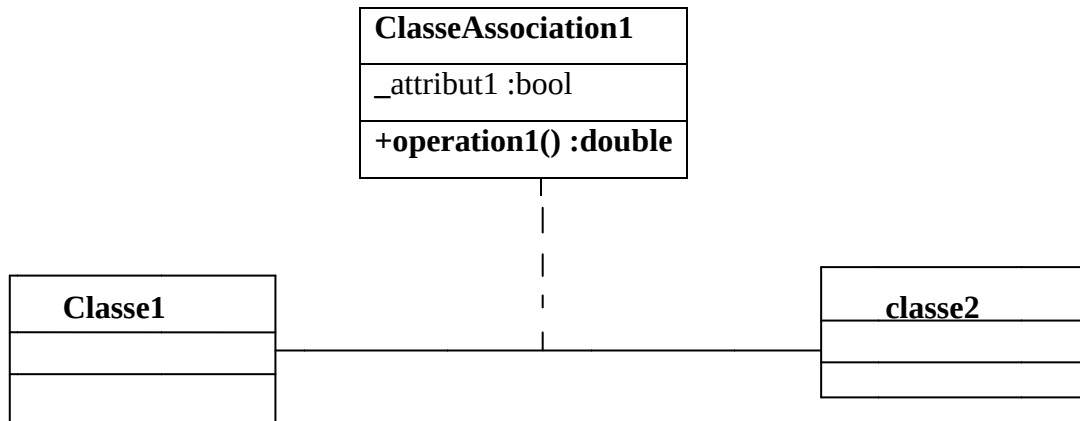


Figure A2 : classe-association.

✚ Les cas d'utilisation

Un cas d'utilisation est la description d'une séquence d'actions exécutées par un système pour conduire à un résultat qui peut être constaté par un acteur particulier. Il sert à structurer les éléments comportementaux d'un modèle et est réalisé par une collaboration comme le montre la **figure A3**, un cas d'utilisation est représenté par une ellipse en trait plein qui, en règle générale, contient seulement son nom.

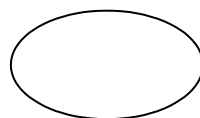


Figure A3 : Cas d'utilisation.

I.1.2 Les éléments comportementaux

Les éléments comportementaux représentent les parties dynamiques des modèles UML. Ce sont les verbes du modèle et ils représentent son comportement dans le temps et dans l'espace

Les interactions

Une interaction est un comportement qui comprend un ensemble de messages échangés au sein d'un groupe d'éléments, dans un contexte particulier, pour atteindre un but bien défini.

Le comportement d'un ensemble d'objets ou celui d'une opération individuelle peut être précisé par une interaction. Cette dernière implique un certain nombre d'éléments, y compris des messages, des séquences d'actions (comportement induit par un message) et des liens (relations entre des éléments). Comme le montre la **figure A4**, un message est représenté par une ligne fléchée, qui indique le nom de son opération.



Figure A4 : Message

I.1.3 Les éléments de regroupement

Les éléments de regroupements représentent les parties organisationnelles des modèles UML. Ce sont des boîtes dans lesquelles un modèle peut être décomposé. Il existe un seul type fondamental d'éléments de regroupement : le « paquetage ».

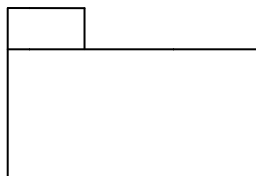


Figure A5 : Paquetage

I.1.4 Les éléments d'annotation :

Les éléments d'annotation représentent les parties explicatives des modèles UML. Ce sont les commentaires qui peuvent accompagner tout élément dans un modèle, à des fins de description, d'exploitation et de remarque. Il existe un type fondamental d'éléments d'annotation appelé « note » qui est simplement un symbole utilisé pour représenter les contraintes et les commentaires rattachés à un élément ou un ensemble d'éléments. Comme le montre la **figure A6**, une note est représentée par un rectangle écorné qui contient un

commentaire textuel ou graphique.

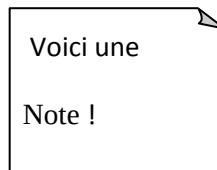


Figure A6 : Note

I.2 Relation dans UML [31]

I.2.1 Dépendance

C'est une relation sémantique entre deux éléments selon laquelle un changement apporté à l'un (élément indépendant) peut affecter la sémantique de l'autre (élément dépendant) comme le montre la **Figure A7**, une dépendance est représentée par une ligne en pointillés qui peut être fléchée ; elle comprend parfois une étiquette.

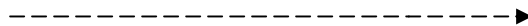


Figure A7 : Dépendance

I.2.2 Association

Une association est une relation structurelle qui décrit un ensemble de liens, un lien constituant une relation entre différents objets .L'agrégation est un type particulier d'association, qui représente une relation structurelle entre un tout et ses parties. Comme le montre la **Figure A8**, une association est représentée par une ligne; elle comprend parfois une étiquette et souvent d'autres décorations, comme la multiplicité et les noms de rôles.

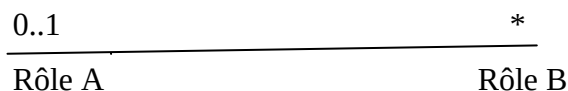


Figure A8 : Association

I.2.3 Généralisation

Une généralisation est une relation de spécialisation selon laquelle les attributs de l'élément spécialisé (l'enfant) peuvent se substituer aux attributs de l'élément généralisé (le parent). De cette manière, l'enfant partage la structure et le comportement du parent. Comme

le montre la **Figure A9**, une relation de généralisation est représentée par une flèche dont le trait est plein et dont la pointe creuse est dirigée vers le parent

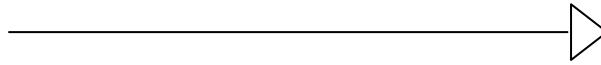


Figure A9 : Généralisation

I.2.4 Réalisation

La réalisation est une relation sémantique utilisée principalement soit pour indiquer qu'une interface est réalisée par une classe, soit pour indiquer qu'un cas d'utilisation est réalisé par une collaboration d'objets. On la représente par :

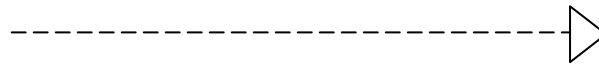


Figure A10 : Réalisation

III. Les diagrammes d'UML [19]

UML permet de définir et de visualiser un modèle, à l'aide de diagrammes.

III.1 Définition

Un diagramme est la représentation graphique d'un ensemble d'éléments que l'on représente en général par un graphe relié de sommets (éléments) et d'arcs (relations). Un diagramme est une projection dans un système.

Chaque type de diagramme UML possède une structure (les types des éléments de modélisation qui le composent sont prédéfinis).

Un type de diagramme UML véhicule une sémantique précise (un type de diagramme offre toujours la même vue d'un système).

Combinés, les différents types de diagrammes UML offrent une vue complète des aspects statiques et dynamiques d'un système.

Par extension et abus de langage, un diagramme UML est aussi un modèle (un diagramme modélise un aspect du modèle global)

UML repose sur neuf types de diagrammes à savoir:

1-Le diagramme de classe

Le diagramme de classes est un diagramme structurel qui exprime d'une manière générale la structure statique d'un système en termes de classes, d'interfaces et de collaborations, ainsi que leurs relations

2-Le diagramme d'objets

Le diagramme d'objets appelé aussi diagramme d'instance représente aussi une structure statique et montre des objets et des liens.

Les notations retenues pour les diagrammes d'objets sont dérivées de celles des diagrammes des classes ; les éléments qui sont des instances sont soulignés.

Pour le nom de l'objet on peut le désigner sous trois formes :

- Nom de l'objet : désignation directe et explicite du nom d'objet
- Nom de l'objet : **nom de la classe**: Désignation incluant le nom de la classe.
- Nom de la classe : désignation anonyme d'un objet d'une classe donnée.

3-Le diagramme de cas d'utilisation

Diagramme comportemental qui montre un ensemble de cas d'utilisation et d'acteurs ainsi que leurs relations.

4- Le diagramme de séquences

Diagramme comportemental qui montre que une interaction et met l'accent sur l'ordre chronologique des messages.

5-Le diagramme de collaboration

Diagramme comportemental qui montre une interaction et met l'accent sur l'organisation structurelle des objets qui envoient et reçoivent des messages.

6-Le diagramme d'états-transitions

Diagramme comportemental qui montre un automate à états finis et met l'accent sur le comportement d'un objet ordonnancé par les événements.

7-Le diagramme d'activité

Diagramme comportemental qui montre un automate a états finis et met l'accent sur le flot d'une activité à l'autre.

8-Le diagramme de composante

Diagramme structurel qui montre un ensemble de comportements ainsi que leurs relations.

9-Le diagramme de déploiement

Diagramme structurel qui montre un ensemble de nœuds ainsi que leurs relations

VII. L'extension d'UML pour le web

L'extension d'UML pour le Web définit un ensemble de stéréotype, d'étiquettes et de contraintes, qui rend possible la modélisation Web.

Un stéréotype

Est une extension du vocabulaire d'UML, il permet d'associer une nouvelle signification à un élément du modèle. On représente un stéréotype par une chaîne de caractères entre guillemet (« »).

Une étiquette

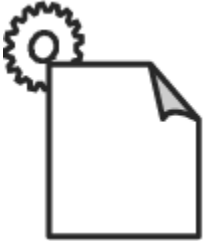
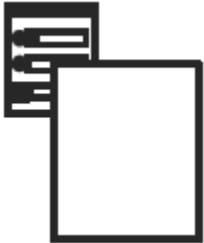
Est une extension des propriétés d'un élément, elle permet la description d'une nouvelle propriété d'un élément du modèle, on la représente par une chaîne de caractères entre chevrons (< >).

Une contrainte

Est une extension de la sémantique d'UML, elle édicte une règle que le modèle doit vérifier pour être qualifié <de bien formé>, elle est représentée par une chaîne de caractères accolades ({ }).

Le principal élément spécifique des applications Web étant les pages Web, plusieurs stéréotypes, qui lui sont destinés sont conçus tels que les cadres, les cibles et les formulaires qui représentent eux aussi des composants architecturalement significatifs dans un modèle.

 Les classes

Icône	La description
 Page serveur	C'est une page web contenant des scripts qui interagissent avec des ressources serveur. C'est le serveur qui est chargé de les exécuter.
 Page client	C'est une page web écrite en HTML, contenant par fois des scripts interprétés par le navigateur.
 Formulaire	Un formulaire est un ensemble de champs de saisie dans une page client. Les attributs de cette classe correspondent aux champs de saisie du formulaire.

Bibliographie

Références bibliographiques

- [1] **Dany Stéphane ZIE FOMEKONG** « Les apports du web 2.0 sur les futurs sites web, les enjeux dans la relation client et les impacts sur les modèles économiques » Université de Picardie Jules Verne - mémoire Master 2 Systèmes d'Information et Informatique Nomade (SIIN) 2009, p4-p7
- [2] **Véronique MESGUICH** « Qu'est-ce que le web 2.0 ? » Journée Juriconnexion 23 novembre 2006
INFOTHEQUE POLE UNIVERSITAIRE LEONARDE VINCI
- [3] **Christophe PORTENEUVE** « Bien développer pour le Web 2.0 » EROYLLES. Date de parution : 16/11/2006, p13
- [4] **Olivier CARTON** Cours XML » Support du cours XML en M2 Pro à l'Université Paris Diderot. Copyright © 2007-2012, p1
- [5] **Thierry TEMPLIER, Arnaud GOUGEON** « JavaScript pour le Web 2.0 » EROYLLES Collection Blanche janvier 2007, p202
- [6] **Thomas CHAIMBAULT** « Web 2.0 : l'avenir du web ? » Dossier documentaire .école nationale supérieure des sciences de l'information et des bibliothèques (Enssib) Septembre 2007, p32
- [7] **capucine COUSIN** « Tout sur le Web 2.0 » Etude (broché). Paru en 07/2010. dunodo
- [8] **Amine Boudefla** « Vers une Méthode pour la Mise en Place de Dispositifs E-learning »
MEMOIRE Pour l'obtention du diplôme de Magister en Informatique. Faculté des Sciences Département d'Informatique . Tlemcen 2010-2011, p7-p8
- [9] **Michel MIELNIKOFF** « Qu'est-ce que l' E-Learning ? » document .centre régional d'innovation et de transfert de technologie. CRITT-TTI, le 07/11/2005, p3
- [10] **Normand Saint-Hilaire, Michelle Larose, Hervé Pilon** « e-Learning Guide pratique de l'apprentissage virtuel en entreprise » TECHNOCompétences, nouvelle version 2003, p22
- [11] étude comparative de plates-formes de formation à distance dans le cadre du Projet @2L Apprentissage des logiciels libres dans L'enseignement Supérieur et la recherche. Octobre 2007, p6-p15

- [12] **B. Kouninef, M. Djelti, S.M. Rerbal** « CONCEPTION ET REALISATION D'UNE PLATE FORME E-LEARNING AVEC MIGRATION AU M-LEARNING » Institut des télécommunications d'Oran 2007, p2-p4
- [13] **Aoued BOUKELIF** « Plateformes collaboratives d'enseignement à distance (CMS, LMS et LCMS), systèmes de gestion de contenus (DMS, ECMS), Environnement Numériques de Travail (ENT), EIAH, groupware et bureau virtuel » Université Saad Dahlab de Blida , Journée scientifique sur la pédagogie universitaire en sciences et techniques. le 22 Mai 2011, p25-p26
- [14] **Jean-Philippe PERNIN** « Objets pédagogiques : unités d'apprentissage, activités ou ressources ? » Revue "Sciences et Techniques Educatives", Hors série 2003 " Ressources numériques, XML et éducation", éditions Hermès. avril 2003 p179-p210,
- [15] **Xebia livre blanc** « Comprendre et savoir utiliser un ESB dans le cadre d'une SOA » business integration architects . Copyright © Xebia 2007. P3
- [16] **Valtech livre orange** Urbanisation & Intégration de Systèmes « THINK SERVICE » Version 1.2. valtech, septembre 2007, p7
- [17] **Michel Leblanc** « les web services : définitions, technologies, acteurs, impacts sur les entreprises et problèmes » Rapport de projet CEL-6003- travail dirigé en commerce électronique Montréal le 14 novembre 2002, p22-p29
- [18] **Cyrielle Lablanche, Florens Seine, Sébastien Gastaud** « Les Web Services » Rapport de TE Université de Nice-Sophia Antipolis Licence d'Informatique 2004–2005, p22
- [19] **Erik Christensen, Francisco Curbera, Greg Meredith, and Sanjiva Weerawarana.** W3C «Web Services Description Language (WSDL) 1.1 », march 2001.
- [20] **Sylvain RAMPACEK** « Sémantique, interactions et langages de description des services web complexes » THÈSE de DOCTORAT l'Université de Reims Champagne-Ardenne Spécialité : Informatique présentée et soutenue publiquement le 10 novembre 2006, p29-p30
- [21] **Drifa ADDOUR, Rachid AHMED-OUAMER** « Actes des rencontres sur la recherche en informatique R2I 2011 » Tizi-Ouzou le 12-14 juin 2011 page 98-105
- [22] **W3C**, RDF/XML Syntax Specification (Revised). Dave Beckett (University of Bristol) 2004

- [23] **Philippe Laublet, Jean Charlet, Chantal Reynaud** « Introduction au Web sémantique »
Action spécifique 32 CNRS / STIC. Rapport de stage. V3 – décembre 2003, p3
- [24] **Sabri Boutemedjet** « Web Sémantique et e-Learning » cours .Université de montréal
2004, p8-p12
- [25] **W3C** «OWL Web Ontology Language Overview» Rédaction : Pierre Attar , 2004
- [26] **Doug Tidwell, James Snell, Pavel Kulchenko** «Programming Web Services with SOAP»
First Edition December 2001, p17
- [27] **PHAN Quang Trung** « Ontologies et Web services » Institut de la Francophonie pour
l'Informatique Rapport du travail d'intérêt personnel encadré Hanoï, juillet 2005, p20-
p28
- [28] **WAHIBA MEZEGHRANE** « mémoire de fin d'étude d'ingénieur en informatique
Thème « application E- administration pour la GAM assurance » année 2006.
- [29] **Xavier Lacot** «Introduction à OWL, un langage XML d'ontologies Web » clever age
Juin 2005, p8-p14
- [30] Manuel d'aide fourni avec le logiciel macromedia DREAMWEAVER 8.
- [31] **Laurent Audibert** « UML2_ de l'application à la pratique (cours et exercices) », Ellipses
dans la collection Info+
- [32] **BOUZIDI Zhor** « Dispositif de supervision et de partage d'expérience adapté à l'e-
learning » mémoire de MASTER II 2010-2011, p18-p64

Webographie

<http://www.w3.org/TR/2004/REC-rdf-syntax-grammar-20040210/>

<http://www.centre-inffo.fr/Les-Learning-Content-Management.html>

<http://www.w3.org/TR/2004/REC-owl-features-20040210/>

<http://www.gdzinfo.com/services-web-semantiques>

<http://www.eutech-ssii.com/ressources/1>

www.developpez.com

www.memoireonline.com

www.siteduzero.com

www.journaldunet.com

www.Commentcamarche.net

www.wikipedia.com

www.davidtouvel.com

www.w3.org

www.zend.com

