

REPUBLIQUE DEMOCRATIQUE DU CONGO
ENSEIGNEMENT SUPERIEUR ET UNIVERSITAIRE

« E.S.U »

UNIVERSITE DE L'ASSOMPTION AU CONGO

U.A.C

B.P 104 BUTEMBO

E-mail : uacuniversite2018@gmail.org

Site : www.uaconline.edu.cd



FACULTE DES SCIENCES ECONOMIQUES ET DE GESTION
DEPARTEMENT DE L'INFORMATIQUE DE GESTION

*Service Web pour l'interopérabilité des services entre
l'Université de l'Assomption au Congo et des entreprises
tierces dans le suivi de stage des étudiants*

Par

KAVUSA KASOKO

*Mémoire présenté et défendu en vue de l'obtention du
titre de Licencié en Informatique de Gestion.*

Option: Conception des Systèmes d'Informations

Directeur : NSENGE MPIA Héritier
Chef des Travaux

Encadreur : KAMBALE KASAMBYA Moïse
Assistant 1

ANNEE-ACADEMIQUE 2022-2023

EPIGRAPHE

Dès que l'homme a commencé à se considérer comme la mesure de toute chose, le monde a commencé à perdre sa dimension humaine, et l'homme a commencé à en perdre le contrôle.

Vaclav Havel, *Disturbing the Peace, Conversation avec Karel Hvizdala*, 1986.

RESUME DU TRAVAIL

Notre étude s'est attelée sur l'interopérabilité des systèmes au moyen d'un service web. Elle s'est basée sur un système de cotation en ligne des stagiaires de l'UAC dans une entreprise tierce. Dans le cas échéant, nous avons ciblé les Bureau Diocésain des Œuvres Médicales (BDOM) où les stagiaires de l'UAC sont admis pour leur stage. Cette démarche a abouti à la création d'un serveur Java et d'un client C-Sharp pour échanger des ressources. Cette étude prouve en suffisance que pour tenir bon aucune entreprise ne peut plus se développer dans un monde clos. L'interopérabilité s'avère nécessaire et quasi obligatoire.

Mots clés : Interopérabilité, Service Web, Serveur ; WSDL, SOAP, HTTP

ABSTRACT

Our study focused on the interoperability of systems using a web service. It was based on an online rating system for UAC trainees in a third-party company. Where appropriate, we targeted the Diocesan Office of Medical works where UAC trainees are admitted for their training. This resulted in the creation of a Java server and a C-Sharp client to exchange resources. This study is proves enough that no company can continue to develop in a closed world. Interoperability is necessary and almost mandatory.

Key words : Interoperability, Web Service, Server, WSDL, SOAP, HTTP

DECLARATION

Je déclare que ce mémoire est mon travail original et n'a pas été précédemment publié ou soumis ailleurs pour l'obtention d'un diplôme. Je déclare également qu'il ne contient aucun matériel écrit ou publié par d'autres personnes, sauf lorsqu'il y est fait référence et que l'auteur est dûment identifié.

KAVUSA KASOKO

Matricule : IG645/2018

Date : Le 19 Septembre 2023

Je confirme par la présente que j'ai examiné le Mémoire de Licence de KAVUSA KASOKO et que je l'ai approuvé pour défense.

Date.....

NSENGE MPIA Héritier

REMERCIEMENTS

Au terme de notre parcours, nous tenons à remercier Dieu pour son assistance pendant tout ce temps de notre formation. Nous remercions également les Responsables de Province des Augustins de l'Assomption et ceux de l'UAC qui avaient mis confiance dans notre demande pour accepter que nous puissions retourner sur le bas de l'école malgré notre âge très avancé. Nous remercions, d'une manière particulière le Père NSENGE MPIA Héritier, *Chef des Travaux* et le Diacre KAMBALE KASAMBYA Moïse, *Assistant*, pour avoir accepté, malgré leurs multiples occupations, de diriger ce travail et surtout de s'être dévoués corps et âme afin que ce travail devienne ce que nous voyons aujourd'hui. Nous ne pouvons pas oublier nos parents KAMBALE KASOKO Célestin et KAHINDO NDENGWA Céline qui nous ont donné la vie et qui continuent à la soutenir d'une manière ou d'une autre. Que tous nos enseignants, nos camarades, nos amis et tous ceux qui nous connaissent, trouvent ici l'expression de notre gratitude.

KAVUSA KASOKO

DEDICACE

A la famille Kasuli Watokoyo

A tous les Augustins de l'Assomption

A tous les amoureux de la nouvelle technologie

A tous mes amis

SIGLES ET ABREVIATIONS

WS : Web Service

UAC : Université de l'Assomption au Congo

BDOM : Bureau Diocésain des Œuvres Médicales

WSDL : Web Services Description Language

SOAP : Simple Object Access Protocol

SOA : Service Oriented Application /Architecture

REST : Representational State Transfer

UML: Unified Modeling Language

XML : eXtensible Markup Language

PHP: Hypertext Preprocessor (historiquement : Personal Home Page tools)

HTTP: HyperText Transfer Protocol

UDDI : Universal Description Discovery and Integration

API: Application Programming Interface

JSON : JavaScript Object Notation

HTML : HyperText Markup Language

W3C : World Wide Web Consortium

OASIS : Organization for the Advancement of Structured Information Standards

ISEAB: Institut Supérieur Emanuel d'Alzon de Butembo

SGBD : Système de Gestion de Base de Données

SQL : Structured Query Language

JaxWS : Java API for XML Web Service

TFC : Travail de fin de Cycle

TABLE DE MATIERES

EPIGRAPHE	II
RESUME DU TRAVAIL.....	III
ABSTRACT.....	IV
DECLARATION	V
REMERCIEMENTS	VI
DEDICACE	VII
SIGLES ET ABREVIATIONS	VIII
TABLE DE MATIERES	IX
LISTE DES FIGURES.....	XI
PREMIER CHAPITRE : INTRODUCTION	1
1.1. CONTEXTUEL DE L'ETUDE	1
1.2. PROBLEMATIQUE	4
1.3. OBJECTIF PRINCIPAL.....	4
1.4. OBJECTIFS SPECIFIQUES	5
1.5. QUESTIONS SPECIFIQUES	5
1.6. CHOIX ET INTERET DU SUJET.....	5
1.7. JUSTIFICATION DE L'ETUDE	6
1.8. DELIMITATION DU SUJET	6
1.9. DIFFICULTES RENCONTREES	6
1.10. SUBDIVISION DU TRAVAIL.....	7
DEUXIEME CHAPITRE: REVUE DE LITTERATURE.....	8
2.0. INTRODUCTION	8
2.1. REVUE DE LITTERATURE THEORIQUE	8
2.1.1. <i>Service Web</i>	8
2.1.1.1. <i>Protocole soap</i>	9
2.1.1.2. <i>Protocole REST</i>	9
2.1.1.3. <i>WSDL</i>	10
2.1.1.4. <i>UDDI</i>	11
2.1.2. <i>Interopérabilité des services</i>	11
2.1.3. <i>Université de l'Assomption au Congo</i>	12
2.1.4. <i>Stage académique</i>	13
2.2. LITTERATURE EMPIRIQUE	14
CONCLUSION	17
TROISIEME CHAPITRE: METHODOLOGIE	18
3.0. INTRODUCTION	18
3.1. CONCEPTION DE LA RECHERCHE	18
3.2. ARCHITECTURE GLOBALE DU SYSTEME	20

3.3.	MODELISATION UML DU SYSTEME	21
3.3.1.	<i>Contexte du système</i>	21
3.3.2.	<i>Diagramme de cas d'utilisation</i>	22
3.3.3.	<i>Diagramme de séquences</i>	23
3.3.3.1.	Cas d'utilisation s'authentifier	23
3.3.3.2.	Cas d'utilisation rechercher	24
3.3.3.3.	Cas d'utilisation imprimer.....	25
3.3.3.4.	Cas d'utilisation saisir_cote	25
3.3.3.5.	Cas d'utilisation ajouter	26
3.3.4.	<i>Diagramme de classes</i>	27
3.3.4.1.	Schémas conceptuel	27
3.3.4.2.	Schéma relationnel	29
	CONCLUSION	30
	QUATRIEME CHAPITRE : ANALYSE DES DONNEES, RESULTATS ET DISCUSION	31
4.0.	INTRODUCTION	31
4.1.	RESULTATS DE LA RECHERCHE	31
4.1.1.	<i>Identifier les différents acteurs du système proposé</i>	31
4.1.2.	<i>Implémenter un serveur offrant des méthodes appelables</i>	33
4.1.3.	<i>Implémenter une application cliente</i>	37
4.1.4.	<i>Tester et valider les prototypes développés</i>	38
4.2.	DISCUSION DES RESULTATS	40
	CONCLUSION PARTIELLE	41
	CINQUIEME CHAPITRE : CONCLUSION GENERALE ET RECOMMANDATIONS	42
5.1.	INTRODUCTION	42
5.2.	CONCLUSIONS.....	42
5.3.	CONTRIBUTIONS AUX TRAVAUX PRÉCÉDENTS	43
5.4.	RECOMMANDATIONS.....	44
	BIBLIOGRAPHIE	45
A.	OUVRAGES	45
B.	ARTICLES.....	45

LISTE DES FIGURES

Figure 1: Architecture d'un Service Web basé sur le protocole SAOP	8
Figure 2: Structure du web service proposé	21
Figure 3: Diagramme de cas d'utilisation	22
Figure 4: Diagramme de séquence du cas d'utilisation se connecter.....	24
Figure 5: Diagramme de séquence du cas d'utilisation Rechercher	24
Figure 6: Diagramme de séquence du cas d'utilisation Imprimer	25
Figure 7: Diagramme de séquence du cas d'utilisation Saisir_cote.....	26
Figure 8: Diagramme de séquence du cas d'utilisation Ajouter.	26
Figure 9: Diagramme de classe	29
Figure 10: Classe étudiant	34
Figure 11: Getters et Setters	34
Figure 12: Serveur du web service.....	35
Figure 13: Serveur du web service.....	36
Figure 14: Client UAC affichant les résultats des stagiaires.....	37
Figure 15: Test de l'adresse WSDL.	38
Figure 16: Test de la méthode getListeStagiaires par SoapUI	39
Figure 17: Création du client .Net.....	40

Premier Chapitre : INTRODUCTION

1.1. Contextuel de l'étude

Les technologies des services Web connaissent une popularité croissante au sein des entreprises modernes vue la dimension d'interopérabilité entre différentes applications et plateformes que ces entreprises possèdent. Ceci constitue un grand potentiel dans le fonctionnement d'une entreprise offrant divers services aux autres entreprises partenaires. Ce potentiel découle du fait que les services Web sont basés sur des normes qui ont été largement acceptées et implémentées, telles que le protocole Simple Object Access Protocol (SOAP) et le Web Services Description Language (WSDL). Le grand nombre de produits et d'outils créés pour faciliter le développement des services Web a également contribué à cette évolution technologique d'interopérabilité entre applications¹.

Dans la littérature technologique existante, il y a lieu de remarquer à nos jours, de nombreuses approches pour construire des systèmes qui exigent l'interopérabilité. Chacune de ses approches existantes présente des avantages et des inconvénients particuliers en matière d'interopérabilité, et chacune fonctionne bien dans certaines circonstances, mais pas dans d'autres. Ce travail de mémoire a donc exploré les services Web, l'une des nombreuses technologies permettant de réaliser l'interopérabilité² afin de les appliquer dans un contexte d'interopérabilité des applications de l'Université de l'Assomption au Congo (UAC) et des entreprises tierces qui reçoivent des étudiants de l'UAC pour le stage académique.

Dans un environnement de services Web, il existe le fournisseur et les consommateurs. Le fournisseur, aussi appelé le serveur, offre une interface bien définie aux consommateurs, aussi identifié par utilisateurs ou clients. Cette interface comprend un ensemble d'opérations et de paramètres d'entrée/sortie typés. En général, les clients et les serveurs interagissent en utilisant un cadre de service web, qui fournit un ensemble de mécanismes garantissant le déploiement du service dans le serveur, avec une description de l'interface du service; le

¹ Cf. Chen S.C & Yang S.Y., "A Web Services, Ontology and Big Data Analysis Technology-Based Cloud Case-Based Reasoning Agent for Energy Conservation of Sustainability Science", *Applied Sciences*, Vol. 10, no. 4, 1387, 2020. <https://doi.org/10.3390/app10041387>.

² Cf. Devi Prasad Bhukya , Reeta Sony A.L & Gautam Muduganti, "On Web Services Based Cloud Interoperability", *International Journal of Computer Science Issues*, Vol. 9, no. 5, 2012, p. 234.

développeur de clients peut générer des artefacts côté client afin d'invoquer facilement les opérations du service et les applications client et serveur peuvent communiquer. Cette communication se fait en échangeant des messages SOAP produits par le service web au nom du client et du serveur³.

Les services web ont ainsi trouvé leur importance dans l'évolution croissante de besoin d'automatisation. Poussée par des processus commerciaux et associée à la nécessité d'une coopération interentreprises agile sur le Web, cette automatisation a alimenté l'intérêt pour l'architecture orientée services (SOA), car les processus commerciaux sont susceptibles d'impliquer un grand nombre de participants qui utilisent des technologies diverses et incompatibles. Le SOA est une approche qui permet, de ce fait, de concevoir, d'architecturer et de réaliser une intégration centrée sur l'interopérabilité, faiblement couplée⁴.

En revanche, dans le cadre de suivi des stagiaires de l'UAC par des entreprises tierces, nous avons ciblé le Bureau Diocésain des Œuvres Médicales (BDOM). Etant donné que les stagiaires après leur période de stage doivent être cotés par le BDOM, le scénario du service web peut être décrit comme suit :

1. On développe des services au BDOM contenant deux méthodes notamment celle retournant la cote de stage qu'un stagiaire a obtenu à la fin de son stage et celle contenant la liste des cotes de tous les stagiaires de l'UAC ayant passé leur stage au Bureau en une année académique X.
2. On implémente une application cliente au sein de l'UAC en n'importe quel langage de programmation qui consommera ces services offerts par le BDOM.

L'université, comme son nom le dit, forme des personnes qui seront lancées dans l'univers pour le décortiquer et produire ce qui sera appelé par la suite, la découverte. Celle-ci est souvent attendue pour améliorer les conditions de vie de l'être humain⁵. Toutefois, cette découverte ne vient pas de soi, c'est au bout d'un travail harassant, soutenu par la familiarisation au domaine à étudier, que des nouvelles notions à expérimenter arrivent à se

³ Cf. Sarah Hussein Toman, "Review of Web Service Technologies: REST over SOAP", *Journal of Al-Qadisiyah for Computer Science and Mathematics*, Vol. 12, no. 4, 2020, p. 21.

⁴ Cf. Sugyan Kumar Mishra & Anirban Sarkar, "Service-oriented architecture for Internet of Things: A semantic approach", *Journal of King Saud University – Computer and Information Sciences*, Vol. 34, no. 1, 2021. <https://doi.org/10.1016/j.jksuci.2021.09.024>.

⁵ Cf. Alfredo Moscardini, Rebecca Strachan & Vlasova T., "The role of universities in modern society", *Studies in Higher Education*, Vol. 47, no. 2, 2020. <https://doi.org/10.1080/03075079.2020.1807493>.

pointer à l'esprit. La familiarisation au domaine à étudier est souvent appelée période de stage. Que dire du stage ?

Le stage est une période pratique exigée des candidats à l'exercice de certaines professions libérales ou publiques. C'est aussi, selon un document consulté, une période pendant laquelle une personne exerce une activité temporaire dans une entreprise ou suit des cours en vue de sa formation. Le stage sert de longue date d'outil pédagogique ou de période de probation dans différents parcours de formation. Longtemps limité à certains cursus préparant à quelques professions comme la médecine, l'ingénierie, l'enseignement, ... cet instrument s'est rependu dans l'enseignement supérieur en commençant par les diplômes professionnels dans le commerce et le management. Désormais on trouve des stages dans des disciplines, des filières et des niveaux de plus en plus variés, et leur mise en œuvre est encouragée⁶.

Le stage est souvent encadré et évalué pour donner des indications justes au candidat chercheur. L'évaluation à l'université devient une question cruciale dans de nombreux pays. C'est ainsi que la qualité de la formation et de la recherche pourront effectivement dépendre de la bonne évaluation, comme le démontrent les différentes contributions regroupées dans un collectif d'auteurs, réunissant des chercheurs belges, canadiens, français, espagnols et suisses⁷.

Dans le cadre de l'UAC, chaque année, des étudiants du terminal au premier cycle et au second cycle, sont priés de passer une période considérable pour le stage dans une entreprise donnée. L'étudiant qui a effectué le stage est ainsi prié de ramener sa cote au jury chargé d'évaluer les cotes obtenues par les étudiants. De cette façon, il y a une franche collaboration entre ces différentes entreprises et l'UAC. Ces entreprises peuvent facilement offrir des services qui interopèrent avec des applications au sein de l'UAC, c'est-à-dire partager certaines ressources entre elles au moyen de quelques protocoles.

⁶ Cf. Karji A., et al, "Evaluation of an Interview-Based Internship Class in the Construction Management Curriculum: A Case Study of the University of Nebraska-Lincoln", *Education Sciences*, Vol. 10, no. 4, 109, 2020. <https://doi.org/10.3390/educsci10040109>.

⁷ Cf. Bedin Véronique (dir.), *L'évaluation à l'université*, Presse universitaire de Rennes, 2009, p. 2.

C'est ainsi que dans cette étude nous avons pensé développer un service web qui sera implémenté au sein du BDOM comme serveur contenant des méthodes appelables en distance pour retourner les cotes obtenues par des étudiants stagiaires de l'UAC ayant passé leur stage dans ce Bureau. Ce service web a été développé en Java, en usant du protocole SOAP et le client qui consommera ce service a été développé en C-Sharp, ce client a été déployé au sein de service académique de chaque département de l'UAC.

1.2. Problématique

C'est depuis des décennies que l'UAC largue des étudiants finalistes sur terrain pour effectuer leur stage de perfectionnement. Ces étudiants s'engagent dans diverses entreprises de Butembo et d'autres choisissent de s'éloigner de Butembo pour trouver des entreprises de leur choix ou pour trouver un terrain encore vierge. Le problème intervient quand il faut récupérer la cote de stage obtenue par l'étudiant à la fin de son séjour de stage. Cela peut prendre beaucoup de temps et cause parfois du retard à l'organisation du jury au sein de cette université.

Comment palier à cette difficulté ? Cette question a constitué l'objet de notre recherche. Cet objet nous a poussé à imaginer un système capable d'assouplir ce désagrément. Ainsi, un système d'interopérabilité entre l'entreprise qui accueille les étudiants de l'UAC et l'UAC au moyen des technologies de Web Service pourra aider à combler cette lacune. La création d'une architecture client-serveur s'est avérée optimale dans l'assurance d'interopérabilité des applications informatiques.

L'interopérabilité est une aptitude cruciale des entreprises, si elles souhaitent s'inscrire dans une dynamique d'intégration. Cette technique est désormais incontournable pour assurer la pérennité économique des entreprises⁸.

1.3. Objectif principal

L'objectif principal de cette étude a été de mettre en œuvre un système client-serveur basé sur l'architecture Service Oriented Application (SOA), utilisant le protocole SOAP pour assurer l'accès distant aux cotes de stage des étudiants de l'UAC via un service web.

⁸ Cf. Adalberto Polenghi et al., "Enterprise information systems interoperability for asset lifecycle management to enhance circular manufacturing", *IFAC-PapersOnLine*, Vol. 5, no. 1, 2021, p. 361.

1.4. *Objectifs spécifiques*

Les objectifs spécifiques de cette recherche ont été formulés comme suit :

1. Identifier les différents acteurs interagissant avec le système en utilisant le langage de modélisation Unified Modeling Language (UML).
2. Implémenter un serveur offrant des méthodes appelables (au sein du BDOM) en distance via le SOAP qui retourne les cotes de stage des étudiants de l'UAC.
3. Implémenter une application cliente au sein de l'UAC qui consomme les ressources offertes par le serveur se trouvant au BDOM.
4. Tester et valider les prototypes développés.

1.5. *Questions spécifiques*

Pour ce qui concerne cette recherche, ces questions nous ont servi de guide:

1. Quels sont les acteurs susceptibles d'interagir avec le système à développer?
2. Quelle technologie existante permet de créer des services web et des clients qui communiquent au moyen d'un langage simple tel que eXtensible Markup Language (XML)?
3. Quel objet PHP, en mode WSDL, peut être capable d'appeler les fonctions SOAP pour les consommer en distance via une technologie web ?
4. Quel est le degré de généralisation de ce prototype dans le contexte d'une autre entreprise partenaire de l'UAC ?

1.6. *Choix et intérêt du sujet*

Le choix de ce sujet s'est inséré dans la dynamique d'implémenter un système offrant la possibilité à l'UAC d'accéder aux cotes de ses étudiants en stage au BDOM, sans se préoccuper des technologies et des plateformes dans lesquelles ces cotes sont gérées au sein de ce Bureau. Plusieurs fois, certains étudiants n'ont pas pu obtenir leurs cotes de stage et même leur département n'a pas pu avoir accès à ces cotes auprès des entreprises où ces étudiants ont effectué leur stage. Ce désagrément a fait que ces étudiants n'ont pas pu passer à la première session car ne possédant pas de preuve de cotes obtenues pendant leur stage. Sur ce, cette application client/serveur basée sur le service web a l'intérêt d'offrir un système

assurant l'interopérabilité des services entre l'UAC et le BDOM, une des entreprises tierces dans lesquelles les étudiants de l'UAC effectuent leur stage professionnel.

1.7. Justification de l'étude

Ce système, une fois en marche, va alléger la tâche de transmission de la côte des stagiaires au niveau de l'UAC et cette dernière aura la facilité d'accès à ces résultats sans plus attendre beaucoup de temps. Ce système sera utilisé par l'UAC et l'entreprise hôte des stagiaires. Il pourra servir de modèle pour toutes les autres structures ayant les mêmes objectifs. Nous avons choisi le Bureau Diocésain des Œuvres Médicales puisqu'il est l'une des entreprises qui ont accueilli nos étudiants pour leur stage en son sein et reste ouvert à rendre ce service à notre institution. La technologie SOAP est la meilleure solution pour l'interopérabilité des systèmes sans tenir compte des plateformes et logiciels utilisés⁹.

1.8. Délimitation du sujet

Ce travail s'est limité au Bureau Diocésain des Œuvres Médicales (BDOM) qui reçoit en son sein des étudiants de l'UAC pour leur stage. Ce Bureau encadre et évalue leur performance afin de faire rapport au Superviseur qui vient de l'UAC. L'application serveur que nous avons déployé au niveau du BDOM pourra être déployée aussi dans d'autres entreprises où on aura un bon nombre de stagiaires venant de l'UAC. Le travail a porté essentiellement sur l'accès au service qui publie les cotes de stages des étudiants de l'UAC en stage dans cette entreprise.

1.9. Difficultés rencontrées

Notre difficulté principale a été le manque des documents à consulter pour réaliser ce travail. Il y a très peu de publication en français sur ce sujet et les quelques publications qui existent en anglais posent le problème de compréhension après traduction. Le système n'est pas encore usuel dans nos entreprises et cela pose un problème d'adaptation de ce nouveau système.

⁹ Cf. AHMAD, Saeed, ALI, Sehrish, WAQAR, Nazish, *et al.* Comparative evaluation of the maintainability of RESTful and SOAP-WSDL web services. In : *2023 International Conference on Business Analytics for Technology and Security (ICBATS)*. IEEE, 2023. p. 1-9.

1.10. Subdivision du travail

Notre travail est subdivisé en cinq chapitres. Le premier chapitre est l'introduction. Celui-ci nous donne un aperçu global sur notre sujet, sa problématique et les objectifs poursuivis. Le deuxième chapitre est la revue de littérature. Dans ce chapitre, il est question de cibler les termes clés de notre recherche et visiter certains auteurs qui abordent notre sujet d'une manière ou d'une autre, afin d'en déceler notre démarcation. Le troisième chapitre qui concerne la méthodologie aborde toutes les méthodes et techniques utilisées pour la réalisation de notre application. Le quatrième chapitre qui concerne l'analyse des données, les résultats et la discussion s'est préoccupé de la réalisation de notre application, de l'installation des outils et la production des documents XML et WSDL et enfin, le test de notre application. Le cinquième chapitre est la conclusion de notre travail. Celui-ci nous fournit la synthèse de notre travail et oriente vers les travaux futurs.

Deuxième chapitre: REVUE DE LITTERATURE

2.0. Introduction

Ce chapitre comprend trois grands points. Le premier aborde les notions clés sur le web service afin de situer notre étude dans un contexte théorique. Le second point, intitulé revue de littérature empirique, fait l'investigation des travaux existants ayant trait au nôtre afin d'y identifier des lacunes conceptuelles et contextuelles. Le dernier point illustre le cadre conceptuel de cette étude qui comprend les variables de la recherche.

2.1. Revue de littérature théorique

L'analyse de la littérature théorique se concentre sur les principaux articles qui examinent des aspects d'une théorie donnée ou qui utilisent la théorie des perspectives comme base pour d'autres domaines de recherche. Cette partie est consacrée principalement à analyser et expliciter certains mots clés de notre domaine. Voilà pourquoi dans cette étude, nous parlons des mots directement liés à l'interopérabilité dans le domaine du service web.

2.1.1. Service Web

Le service Web est une technologie qui permet à des applications de dialoguer à distance via Internet, et ceci indépendamment des plateformes et des langages sur lesquelles elles reposent. Le service web est un composant implémenté dans n'importe quel langage, déployé sur n'importe quelle plate-forme et enveloppé dans une couche de standards dérivés du Extensible Markup Language (XML). SOAP et REST sont deux types de services Web¹⁰. L'image ci-dessous illustre l'architecture de base d'un service web :

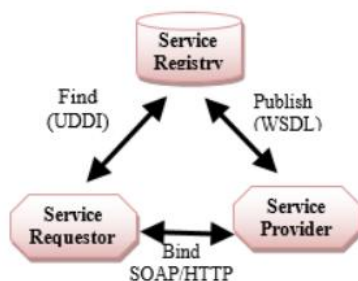


Figure 1: Architecture d'un Service Web basé sur le protocole SOAP

Source : Mohammad Ali A. Hammoudeh & Ajlan S. Al-Ajlan, "Implementing Web Services Using PHP Soap Approach", *International Journal of Interactive Mobile Technologies (iJIM)*, Vol. 14, no. 10, 2020, p. 37.

¹⁰ Cf. Sarah Hussein Toman, "Review of Web Service Technologies: REST over SOAP", *Journal of Al-Qadisiyah for Computer Science and Mathematics*, Vol. 12, no. 4, 2020, pp. 18-19.

Les services web sont des technologies qui permettent de collaborer avec les partenaires commerciaux externes d'une entreprise distribuée en développant des normes et des protocoles de communication émergents tels que SOAP, REST UDDI et WSDL via HTTP¹¹.

2.1.1.1. Protocole soap

Le protocole SOAP (Simple Object Access Protocol) est un protocole de messagerie. Il permet à des programmes qui s'exécutent sur des systèmes d'exploitation distincts (tels que Windows et Linux) de communiquer au moyen du protocole HTTP (HyperText Transfer Protocol) et de son langage, XML¹². Sachant que les protocoles Web sont installés et mis à disposition sur tous les systèmes d'exploitation dominants, le protocole HTTP et le langage XML constituent une solution toute trouvée qui permet aux programmes de s'exécuter et de communiquer sur les différents systèmes coexistant au sein d'un réseau. Le protocole SOAP spécifie exactement la manière de coder un en-tête HTTP et un fichier XML, afin que deux programmes, présent sur des ordinateurs différents, puissent s'appeler et se transmettre des informations. Le protocole SOAP spécifie également la manière dont le programme appelé, renvoie une réponse¹³.

2.1.1.2. Protocole REST

Le transfert d'état représentationnel (REST) a été proposé par Roy Fielding en 2000 dans sa thèse de doctorat. L'objectif de ce style était d'améliorer les performances, l'évolutivité, la simplicité, la modifiabilité, la visibilité de la communication, la portabilité et la fiabilité. Pour atteindre ces propriétés, Fielding a défini les six contrôles suivants : artefacts d'architecture liés au client et au serveur, absence d'état, capacité à mettre les données en cache, système à plusieurs niveaux, interface standard et, éventuellement, code à la demande¹⁴.

REST est un style architectural pour les services web. Dans REST, le serveur répond aux demandes du client en utilisant toutes les ressources autorisées dans les systèmes

¹¹ Cf. Mohammad Ali A. Hammoudeh&Ajlan S. Al-Ajlan, *Op.Cit.*, p. 35.

¹² Cf. Mohamed Shabbir A. Nabi, et al., "Suitability of Using SOAP Protocol to Secure Electronic Medical Record Databases Transmission", *International Journal of Pharmacology*, Vol. 6, 2010, p. 961.

¹³Cf. Kiah M.L.M., et al., "An Enhanced Security Solution for Electronic Medical Records Based on AES Hybrid Technique with SOAP/XML and SHA-1", *J.MedSyst*, Vol.37, 2013,9971.
<https://doi.org/10.1007/s10916-013-9971-2>.

¹⁴Cf. Adeel Ehsan et al., « RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions », *Applied Sciences*, Vol. 12, no 9 (janvier 2022): 4369. <https://doi.org/10.3390/app12094369>.

informatiques. Dans ses travaux en 2016, Grabis a comparé les API REST et SOAP en utilisant les métriques d'évaluation des logiciels et a découvert que REST est inférieur à SOAP en termes de temps de réponse¹⁵.

Ainsi pouvons-nous conclure que les deux technologies SOAP et REST sont conçues pour connecter des applications et utilisent principalement les protocoles et commandes HTTP tels que GET, POST et DELETE. Tous deux peuvent utiliser XML pour former les requêtes et les réponses. Cependant, SOAP dépend de XML par conception, tandis que REST peut également utiliser JSON, HTML et du texte brut. Toutefois, il faut bien noter la grande différence entre ces deux technologies. SOAP est un protocole d'échange de données XML, REST est un style d'architecture. SOAP est construit à partir d'appels de procédures à distance, tandis que REST est basé sur des ressources¹⁶.

2.1.1.3. WSDL

Web Services Description Language (WSDL) est un langage de description fondé sur XML. Il a été soumis au W3C (World Wide Web Consortium) comme standard industriel pour la description des services Web. WSDL tire sa puissance de deux principes architecturaux principaux : la possibilité de décrire un ensemble d'opérations métier et la possibilité de séparer la description entre deux unités de base. Ces unités fournissent une description des opérations et les détails du mode d'intégration de l'opération et des informations qui lui sont associées¹⁷.

Un document WSDL définit des services en tant que collections de points de contact de réseau ou de ports. Dans WSDL, les définitions abstraites des points de contact (endpoints) et des messages sont séparées de leur déploiement réseau concret ou des liaisons de format de données. Ainsi, la réutilisation des définitions abstraites est possible : messages, qui sont des descriptions abstraites des données échangées et types de ports, qui sont des collections abstraites d'opérations. Les spécifications de format de données et de protocole concret pour un type de port particulier constituent une liaison réutilisable. Un port est défini en associant

¹⁵ Sarah Hussein Toman, « Review of Web Service Technologies: REST over SOAP », s. d.

¹⁶ Stephen J. Bigelow et Gaétan Raoul, « Quels sont les types d'API et leurs différences ? », in LeMagIT, <https://www.lemagit.fr/conseil/Quels-sont-les-types-dAPI-et-leurs-differences>.

¹⁷ Cf. Sunyaev, "Web Services", in *Internet Computing*, Springer, Cham, 2020. https://doi.org/10.1007/978-3-030-34957-8_6.

une adresse réseau à une liaison réutilisable et une collection de ports définit un service. C'est pourquoi, un document WSDL est composé de plusieurs éléments¹⁸.

Un document WSDL est un fichier au format XML utilisant les éléments suivants : un **portType** qui définit le service web, en particulier les opérations qu'il réalise et le type de messages échangés ; un **message** qui comprend une ou plusieurs parties représentant les paramètres d'entrée ; un **types** qui définit les types de données utilisés par le service web et enfin un **binding** qui précise le protocole utilisé et le format de message¹⁹.

2.1.1.4. UDDI

Universal Description Discovery and Integration (UDDI) est l'équivalent des pages jaunes dans les services Web. Tout comme avec les pages jaunes classiques, avec UDDI on peut rechercher une entreprise offrant les services dont on a besoin et consulter le service offert. On peut également, proposer un service Web sans l'inscrire, mais si on souhaite toucher un public plus important, UDDI permet aux clients éventuels de trouver ces services. Un service de registre UDDI est un service Web qui gère des informations sur les fournisseurs de services, les implémentations de service et les métadonnées de service. Les fournisseurs annoncent leurs services Web sur le registre UDDI. Les consommateurs utilisent ensuite UDDI pour découvrir des services Web adaptés à leurs besoins et ainsi obtenir les métadonnées de service nécessaires pour consommer ces services²⁰.

Le standard UDDI défini par l'OASIS vise à décrire une manière standard de publier et d'interroger les services web proposés par un réseau, généralement au sein d'un service d'annuaire recensant les services web de l'organisation²¹.

2.1.2. Interopérabilité des services

L'interopérabilité est un terme informatique désignant des systèmes capables de s'adapter et de collaborer avec d'autres systèmes indépendants déjà existants ou encore à créer. Comme souligné plus haut, à l'heure actuelle aucune entreprise ne peut plus se prévaloir

¹⁸ Cf. Kiah M.L.M., et al., *Op.Cit.* <https://doi.org/10.1007/s10916-013-9971-2>.

¹⁹ Cf. *Idem*.

²⁰ Cf. Balaji N., et al., "Enhancements for UDDI using User Preferential Web Service Selection Model based on SLA", *International Journal of Engineering and Technology (IJET)*, Vol. 5, no. 5, 2013, p. 4057.

²¹ Cf. Balaji N., et al., *Op.Cit.*, pp. 4057-4058.

fonctionner dans un monde clos sans interaction avec d'autres. Tout le monde a besoin d'un appui extérieur pour mieux se développer²². Des entreprises et des particuliers se trouvent confrontés presque régulièrement aux problèmes d'interopérabilité numérique. Ces problèmes peuvent prendre plusieurs formes : un document de travail qui ne peut être lu qu'avec un seul logiciel, une machine incapable de communiquer avec une autre, un appareil qui ne permet d'acheter un produit ou une application que sur une plateforme spécifique, etc.²³

2.1.3. Université de l'Assomption au Congo

L'UAC est une université privée catholique sous la direction des Augustins de l'Assomption, dits Pères Assomptionnistes. Elle se situe en Ville de Butembo, Province du Nord-Kivu, République Démocratique du Congo (RDC). L'UAC, à ses débuts, était conçu comme un Campus dans le but d'assurer aux religieux Assomptionnistes et ceux d'autres congrégations la formation philosophique de niveau universitaire et leur permettre de continuer leur formation sacerdotale. Le Campus Saint Augustin de Bulengera devenu Institut Supérieur Emanuel d'Alzon de Butembo (ISEAB) a fini par ouvrir ses portes aux laïcs pour affronter les deux sections déjà en son sein, à savoir la philosophie et les sciences de développement. Plus tard il y aura la section de Gestion Informatique qui va ouvrir ses portes²⁴.

Devenue UAC, ces différentes sections vont être insérées dans les différentes facultés retenues par le ministère de l'Enseignement Supérieur et Universitaire, à savoir la faculté de lettres et sciences humaines (avec les départements de philosophie et de sciences de l'information et de la communication), la faculté de sciences économiques et de gestion (avec les départements d'informatique de gestion appliquée à la gestion des entreprises, d'informatique de gestion et anglais des affaires, de management général et de gestion des entreprises et organisation du travail), la faculté de psychologie et sciences de l'éducation (avec les départements de psychologie scolaire et de psychologie clinique), la faculté de sciences appliquées (avec les départements d'architecture, de génie civil, de génie informatique, de génie électrique et de génie mécanique) et la Faculté de sciences de

²² Cf. Chen Yuhan, et al., "Evaluation Methodology of Interoperability for the Industrial Domain: Standardization vs. Mediation", *Processes*, Vol. 11, no. 4: 1274, 2023. <https://doi.org/10.3390/pr11041274>.

²³ Cf. Paniagua Cristina, "Service Interface Translation. An Interoperability Approach", *Applied Sciences*, Vol. 11, no. 24, 11643, 2021. <https://doi.org/10.3390/app112411643>.

²⁴ Cf. Kambale Kasambya Moïse, et al., "Déploiement mobile d'un réseau de neurones profond pour l'identification des étudiants de l'UAC/ISDA en salle d'examens", *International Journal of Innovation and Applied Studies*, Vol. 39, 2023, p. 137.

développement (avec le département de développement rural, développement communautaire, science de développement des milieux ruraux et urbains)²⁵.

2.1.4. Stage académique

Le stage est une période d'activité durant laquelle un étudiant met en application les enseignements théoriques suivis en classe, dans un organisme d'accueil. Il ne peut en aucun cas exécuter une tâche régulière correspondant à un poste de travail permanent de l'entreprise. Cette période d'expérience professionnelle a une durée limitée. En d'autres termes, le stage est un moment passé dans une structure professionnelle par des étudiants et des diplômés désireux d'acquérir des compétences et une expérience dans un domaine particulier²⁶. Pourtant, les avantages exacts des stages et la manière dont ils sont obtenus font toujours l'objet d'un débat. Il ne fait guère de doute que les stages peuvent avoir un effet direct et positif sur un certain nombre d'indicateurs de carrière, du moins dans de bonnes conditions. Des études portant spécifiquement sur l'éducation ont montré que, par rapport à l'absence d'une telle expérience, les stages sont associés à une plus grande attractivité des candidats à l'emploi pour les recruteurs, à l'obtention plus rapide et plus facile d'un emploi par les diplômés, à des niveaux de salaire plus élevés ainsi qu'à une plus grande satisfaction au travail²⁷.

Selon une étude récente, le stage est l'occasion d'intégrer une expérience professionnelle dans un programme de formation de premier cycle en participant à un environnement de travail structuré et supervisé, avec des avantages tels que la possibilité de travailler dans un environnement professionnel ou lié à la carrière. En plus, le stage permet de sensibiliser les étudiants à leur carrière, leur donner l'occasion d'évaluer, de réfléchir et d'essayer un domaine professionnel, d'acquérir une expérience précieuse qui les aidera à trouver un futur emploi, de gagner de l'argent ou des crédits, d'établir des contacts

²⁵Cf. Kambale Kasambya Moïse, Héritier Mpia, et Vingi Baraka, « Déploiement mobile d'un réseau de neurones profond pour l'identification des étudiants de l'UAC/ISDA en salle d'examens », *International Journal of Innovation and Applied Studies* 39 (1 mars 2023): 136-55.

²⁶ Cf. Goplani Mala, Gupta Akash, & Sabhani Jewel, "A study on impact of internship on regular studies of undergraduate students", *International Journal of Advance and Innovative Research*, Vol. 7, no.1, 2020, p. 92.

²⁷ Cf. Jens F. Binder, et al., "The academic value of internships: Benefits across disciplines and student backgrounds", *Contemporary Educational Psychology*, Vol. 41, 2015, pp.73-74.
<https://doi.org/10.1016/j.cedpsych.2014.12.001>.

professionnels, de rendre l'apprentissage en classe plus intéressant, d'aider les étudiants à développer des compétences en matière de recherche²⁸.

2.2. Littérature empirique

La revue de littérature empirique concerne les différentes analyses documentaires en tant qu'éléments des travaux précédents pouvant permettre aux chercheurs d'améliorer leur compréhension par rapport à ces travaux antérieurs dans un domaine précis. Cette analyse leur permet d'identifier plus facilement les lacunes dans la documentation et ces recherches antérieures. Ce qui est le plus important encore, ce que les synthèses recueillies à ce niveau, peuvent remettre en question certaines hypothèses et normes établies dans un domaine ou sur un sujet donné, de desseller certaines erreurs à corriger pour un discours scientifique cohérent dans le futur²⁹.

Dans l'article intitulé « *Web Service for Document Management of University Degree Projects* », Camilo Sarmiento et les autres coauteurs se sont focalisés sur la gestion des documents fournis par un étudiant tels que les travaux de fin de cycle, les mémoires et les thèses. Ceci dans le but de faciliter le choix du sujet de travail sans pouvoir répéter les sujets déjà traités. Ils ont proposé de développer un service Web pour la gestion des fichiers générés dans le cadre du processus de remise des diplômes. Ce qui permettait d'administrer et de stocker les documents en toute sécurité et de manière à y accéder facilement³⁰.

Ce processus a été réalisé grâce à l'utilisation d'un service REST API intégré à la technologie client-serveur. Pour son développement, Django a été utilisé comme cadre d'application backend avec python et javascript comme langages de programmation. Le framework Django a été choisi parce qu'il est haut de gamme et qu'il gère une grande partie de la complexité du développement Web et facilite l'utilisation de la base de données PostgreSQL³¹. A l'issue de cette analyse il est possible de reconnaître que l'architecture client-serveur peut être considérée comme l'une des architectures les plus utilisées dans le développement des applications web et mobile. Pour l'avenir, il est souhaitable d'ajouter des

²⁸ Cf. Goplani Mala, Gupta Akash, & Sabhani Jewel, *Op.Cit.*, p. 92.

²⁹ Cf. Sascha Kraus et al., "Literature Reviews as Independent Studies: Guidelines for Academic Practice", *Review of Managerial Science*, Vol. 16, no. 8, 2022. <https://doi.org/10.1007/s11846-022-00588-8>.

³⁰ Cf. Camilo Sarmiento, et al., "Web Service for Document Management of University Degree Projects", *ICAIW 2022: Workshops at the 5th International Conference on Applied Informatics 2022*, 2022, Arequipa, pp. 214-215.

³¹ Cf. Camilo Sarmiento, et al., *Op.Cit.*, p. 219.

alertes pour des fichiers déjà téléchargés ou indiquer combien de fois le document a été téléchargé³².

Aftab et les autres coauteurs, poussés par le développement rapide de l'Université, ont écrit l'article intitulé « *An Implementation of Web Services for Inter-Connectivity of Information Systems* » pour réaliser un système qui facilite l'interopérabilité des sous-systèmes. Ces derniers désignaient les différents départements de leur université, à savoir, les campus étendus, la bibliothèque, les étudiants mais aussi les départements de gestion de l'information et de l'administration. Les agents dans ces départements doivent échanger et partager des informations. Pour implémenter ce système, il leur a fallu différents langages de programmation comme Java et C#, les différents systèmes de gestion de base de données (SGBD) comme SQL Server, MySQL et Oracle, mais aussi l'architecture orientée services (SOA) dans le but de gagner du temps dans la transmission de réponse pour atteindre un Service Web. Ce qui a abouti au système d'information qu'ils ont appelé i3 pour l'université de Sindh au Pakistan. Le système i3 a résolu le problème qui se posait fréquemment dans le processus de vérification d'un étudiant en provenance de différents départements. Ce processus s'était avéré nécessaire au moment de la délivrance du certificat final ou du diplôme à l'étudiant³³.

Comme méthodes et outils utilisés dans leur recherche, il y a SOA qui est souvent utilisé dans un système distribué, le protocole SOAP, et le langage XML. Ce dernier est utilisé par les applications pour échanger des informations via le protocole Internet HTTP. L'objectif principal des services SOA est de créer des services évolutifs, flexibles et faiblement couplés pour échanger des informations. I3 a résolu le problème d'interconnexion des départements de l'université Sindh³⁴. Le problème reste celui de surmonter la complexité d'interconnecter les applications de tous les départements qui fonctionnent sur différents types de plates-formes et de logiciels³⁵.

En Indonésie, il y avait un réel besoin, celui d'augmenter le nombre des publications. Par leur article intitulé « *A Database Integrated System Based on SOAP Web Service* », les

³²Cf. *Ibidem*, p. 220.

³³Cf. Aftab Chandio, et al., "An Implementation of Web Services for Inter-Connectivity of Information Systems", *International Journal of Computing and Digital Systems*, Vol. 3, no. 3, 2014, pp. 1-3.

³⁴Cf. *Ibidem*, pp. 3-4.

³⁵Cf. *Ibidem*, p. 6.

auteurs montrent comment le ministère a fait recours aux publications de fin d'étude comme en licence, en master et au doctorat pour augmenter le nombre des publications. Il y a eu aussi un encouragement des publications d'articles dans les revues et journaux mondiaux dans le même but de multiplier les publications. Le besoin était alors celui de constituer une base des données comme une bibliothèque numérique facile à consulter. Cette bibliothèque devait ainsi recenser toutes les publications afin de fournir un accès rapide aux chercheurs³⁶.

Pour la réalisation de cette bibliothèque les normes utilisées ont été XML et SOAP. La modélisation de la base des données a été faite au moyen des diagrammes d'UML et sa conception au moyen du SGBD MySQL. Dans son cadre conceptuel, on trouve le superviseur, l'enseignant, l'enseignant assistant, l'étudiant et l'invité. Cet article a fourni une compréhension totale de l'évolution des services WSDL et SOAP avec des clients qui utilisent PHP³⁷.

Dans leur article, « *Implementing Web Services Using PHP Soap Approach* », Hammoudeh et Ajlan visaient à élargir l'ingénierie actuelle des services Web pour répondre aux besoins spécifiques des cadres d'apprentissage en ligne. C'est là même l'objectif de cette méthodologie préconisée par ces auteurs. La caractéristique la plus importante de cette méthodologie est qu'elle permet aux clients de différents établissements, dans différents pays, de partager du matériel et de travailler ensemble en associant des cours, même s'ils sont hébergés sur un cadre d'apprentissage en ligne différent. Cet article a fourni une compréhension totale de l'évolution des services WSDL et SOAP et des clients qui utilisent PHP. Les variables utilisées sont le prestataire de services, le Registre des services spécialiste, le Demandeur de service, le langage de programmation PHP, la méthodologie utilisée qui est WSDL et SOAP³⁸.

Cet article indique les applications SOAP les plus utilisées aujourd'hui et garantit l'utilisation du package PHP SOAP pour le développement d'applications SOAP. Il ne se limite pas seulement à donner un bon aperçu de la façon dont les clients et les fonctions des

³⁶Cf. Havaluddin H., et al., "A Database Integrated System Based on SOAP Web Service", *TEM Journal*, Vol. 8, no. 3, 2019, p. 782.

³⁷Cf. Havaluddin H., et al, pp. 783-784.

³⁸Cf. Hammoudeh M.A.A. & Al-Ajlan, A.S., "Implementing Web Services using PHP Soap Approach. *International Journal of Interactive Mobile Technologies (iJIM)*, Vol. 14, no. 10, 2020, pp. 35–37. <https://doi.org/10.3991/ijim.v14i10.14391>.

serveurs communiquent, mais aussi et surtout de la façon dont ces messages XML sont analysés dans les variables PHP³⁹.

Eu égard les limites de la littérature reprise ci-haut, une littérature orientée vers le recensement des travaux publiés à l'université, l'interopérabilité entre les départements de l'université, l'accès facile à la bibliothèque numérique et l'apprentissage en ligne, notre travail va se focaliser sur l'interopérabilité entre l'UAC et les entreprises tierces dans le suivi et l'évaluation des stagiaires. Ces publications ne mentionnent pas spécifiquement une fiche d'évaluation. C'est cette dernière qui fait l'objet de notre recherche.

Conclusion

Comme souligné à l'introduction de ce chapitre, nous avons illustré de façon brève certains concepts liés à notre sujet de recherche dont l'interopérabilité des services, les protocoles SOAP et REST, etc. Cette revue de littérature théorique nous a aidés à comprendre plusieurs concepts liés au web service qui est le domaine de notre recherche. Après ces termes, nous avons revu certains travaux antérieurs qui se sont intéressés au service Web surtout dans le cadre de l'éducation et des institutions supérieures et universitaires. Enfin, s'en est suivi l'élaboration du cadre conceptuel de cette étude.

³⁹Cf. Hammoudeh M.A.A. & Al-Ajlan, A.S., *Op.Cit.*, p. 43.

Troisième Chapitre: METHODOLOGIE

3.0. *Introduction*

Avant d'entamer des constructions, tout architecte doit d'abord s'asseoir pour tracer des croquis et se représenter schématiquement ce qu'il veut réaliser comme œuvre. Le recours à la modélisation est depuis longtemps une pratique indispensable au développement logiciel, car un modèle est prévu pour arriver à anticiper les résultats du codage. Un modèle est en effet une représentation abstraite d'un système destiné à en faciliter l'étude et à le documenter⁴⁰. C'est l'étape qui va nous occuper tout au long de ce troisième chapitre consacré à la méthode à suivre.

Ce chapitre comprend trois grandes sections ayant quelques sous-sections pour certaines. La première section parle de la conception de notre système au moyen du langage UML. Elle nous présente en grosso modo certains diagrammes qui nous ont aidés à modéliser notre application. Ce qui nous a conduits à passer au prototypage comme méthode de développement du modèle qui sera à la longue déployé. La deuxième section est la présentation de la structure de notre application comme pour tout autre service web. Cette structure nous présente le processus d'utilisation du service web. La troisième section est la plus étendue avec plusieurs sous-sections. Elle s'occupe de la modélisation de notre système en présentant le contexte de notre préoccupation et en exprimant les besoins de l'utilisateur à travers plusieurs diagrammes proposés par le langage UML. Les diagrammes de cas d'utilisation, de séquence et de classe avec les schémas conceptuel et relationnel, nous donnent l'aperçu global de notre application.

3.1. *Conception de la recherche*

Cette étude a fait usage de plusieurs méthodes et matériels afin d'atteindre les objectifs définis dans le premier chapitre. Etant dans le domaine d'ingénierie, nous avons d'abord utilisé le langage unified modeling language (UML) pour modéliser notre prototype. UML est un langage de modélisation graphique et textuel destiné à comprendre et à décrire des besoins,

⁴⁰Cf. ROQUES Pascal, *Le cahier du programmeur. UML 2. Modéliser une application web*, 4^e édition, Paris, Eyrolles, s.d, p.2.

spécifier et documenter des systèmes, esquisser des architectures logicielles, concevoir des solutions et communiquer des points de vue⁴¹. UML est un langage de modélisation visuelle qui a été créé pour aider les développeurs et les ingénieurs logiciels à concevoir, à analyser et à documenter les systèmes logiciels. UML a évolué au fil des ans et plusieurs versions ont été publiées pour répondre aux besoins changeants du développement de logiciels⁴².

En d'autres termes, UML permet de mettre sur pied des modèles en tenant compte des besoins des utilisateurs. C'est de cette façon qu'UML rend ces artefacts évolutifs, sécurisés et robustes en exécution. Les diagrammes UML utilisés dans cette recherche sont : (a) Les cas d'utilisation qui constituent un moyen de recueillir et de décrire les besoins des acteurs du système. Ils peuvent être aussi utilisés par la suite comme moyen d'organisation du développement du logiciel, notamment pour la structuration et le déroulement des tests du logiciel. Le diagramme de cas d'utilisation permet de capturer le comportement d'un système, d'un sous-système, d'une classe ou d'un composant tel qu'un utilisateur extérieur le voit⁴³. (b) L'objectif du diagramme de séquence est de représenter les interactions entre objets, en indiquant la chronologie des échanges. Cette représentation peut se réaliser par les cas d'utilisation en considérant les différents scénarios associés. Le diagramme de séquence qui présente la coopération entre différents objets du système, a été lui aussi utilisé dans cette étude. Les objets sont définis et leur coopération est représentée par une séquence de messages entre eux. Les objets peuvent être connectés à des classes existantes ou bien être créés indépendamment de toute classe. Si les objets sont connectés à des classes, les messages peuvent être connectés à des opérations. Les diagrammes de séquence sont créés dans les interactions⁴⁴.

(c) Le diagramme de classes a été le dernier type de diagrammes UML utilisé dans cette étude. Le diagramme de classes est considéré comme le plus important de la modélisation orientée objet, il est le seul obligatoire lors d'une telle modélisation. Ce diagramme permet de donner la représentation statique du système à développer. Cette représentation est centrée sur les concepts de classe et d'association. Les traitements sont de point de vue des acteurs, le diagramme de classes en montre la structure interne. Ce

⁴¹Cf. *Ibidem*, p.4.

⁴²Cf. Xavier Blanc, et al., *UML 2 pour les développeurs*, Eyrolles, Paris, 2006, p. VII.

⁴³Cf. GABAY Joseph et GABAY David, *UML 2. Analyse et conception*, Dunod, Paris, 2008, p.61.

⁴⁴Cf. *Ibidem*, p.90.

diagramme permet de fournir une représentation abstraite des objets du système qui vont interagir, matérialisés par des opérations, alors que le diagramme de cas d'utilisation montre lui, un système pour réaliser les cas d'utilisation. Il est important de noter qu'un même objet peut très bien intervenir dans la réalisation de plusieurs cas d'utilisation. Les cas d'utilisation ne réalisent pas une partition des classes du diagramme de classes. Un diagramme de classes n'est donc pas adapté (sauf cas particulier) pour détailler, décomposer, ou illustrer la réalisation d'un cas d'utilisation particulier⁴⁵.

Au-delà de l'UML comme langage de modélisation, nous avons également utilisé le prototypage comme méthode de développement d'artéfact. La méthode du prototypage consiste à tester la faisabilité technique du futur produit. Adopter une méthode de prototypage permet d'effectuer différentes expérimentations pour le développement technique du produit, afin de pouvoir ajuster le tir et faire évoluer le produit en perspective. En fonction des résultats des tests et des analyses, on peut alors adapter les fonctions de services, les matériaux, la conception, l'assemblage des pièces, etc. et vérifier le fonctionnement global du produit avant la fabrication finale⁴⁶.

Les prototypes sont souvent utilisés dans la phase finale de test dans un processus de « *Design Thinking* » afin de déterminer le comportement des utilisateurs avec le prototype, de révéler de nouvelles solutions aux problèmes ou de déterminer si les solutions mises en œuvre ont été ou non couronnées de succès. Les résultats générés par ces tests sont ensuite utilisés pour redéfinir un ou plusieurs des problèmes identifiés lors des phases précédentes du projet et pour développer une compréhension plus solide des problèmes auxquels les utilisateurs peuvent être confrontés lorsqu'ils interagissent avec le produit dans l'environnement prévu⁴⁷.

3.2. *Architecture globale du système*

En informatique, l'architecture fait référence à une structuration ou organisation des différents éléments d'un système. Ces éléments sont conçus de façon à être interconnecté pour

⁴⁵Cf. GABAY Joseph et GABAY David, *Op.Cit.*, p.17.

⁴⁶Cf. Radka NACHEVA, "Prototyping approach in user interface development", in *2nd conference on innovative teaching methods (ITM 2017) 28-29 June 2017*, University of economics varna, Bulgaria, 2017, p. 80.

⁴⁷Cf. AMANO Tsuyoshi, et al., "Rethinking the prototyping process for applying design thinking to business model innovation", *Conference Proceedings of the Academy for Design Innovation Management*, Vol. 1, no. 1, 2019, p. 1208.

pouvoir collaborer et enfin réaliser le travail demandé. Certains éléments dépendent des autres dès qu'un processus est lancé. Pour mener à bon port notre recherche, nous avons proposé l'architecture ci-dessous afin de visualiser les différentes phases de notre application.

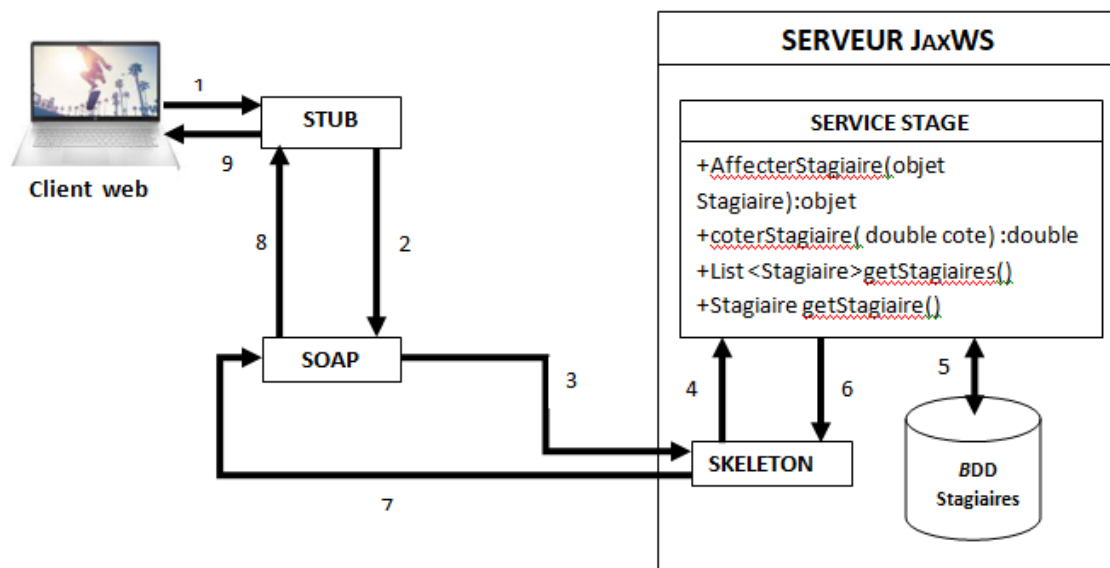


Figure 2: Structure du web service proposé

Cette figure résume le processus qui se crée quand un client veut consommer notre application service web. Le client soumet sa requête. Celle-ci est récupérée par le Stub qui la soumet, au format XML ou le WSDL, au protocole SOAP pour approbation. Le protocole SOAP par l'entremise du SKELETON la soumet au Serveur JaxWS. Ce dernier consulte sa Base de données et y puise une réponse conforme à la requête soumise. La réponse, via le SKELETON et toujours au format XML, est renvoyée au moteur de SOAP qui, par le service du STUB parvient au Client demandeur du Service.

3.3. Modélisation UML du système

3.3.1. Contexte du système

Le contexte de notre travail est tel qu'un serveur est implémenté au niveau du bureau diocésain des œuvres médicales (BDOM) où sont envoyés les stagiaires. Lors de la cotation de ces derniers, le client qui est l'UAC sollicite ce service au BDOM et celui-ci lui envoie la cote demandée. A part ces deux acteurs, il y a une intervention minime de l'étudiant, mais aussi d'un utilisateur quelconque et le suivi du système par un administrateur. Sont là les cinq

utilisateurs préconisés pour notre système et qui doivent interagir pour le fonctionnement du système.

3.3.2. Diagramme de cas d'utilisation

La figure ci-après illustre les différents acteurs et cas d'utilisation intervenant dans notre modélisation.

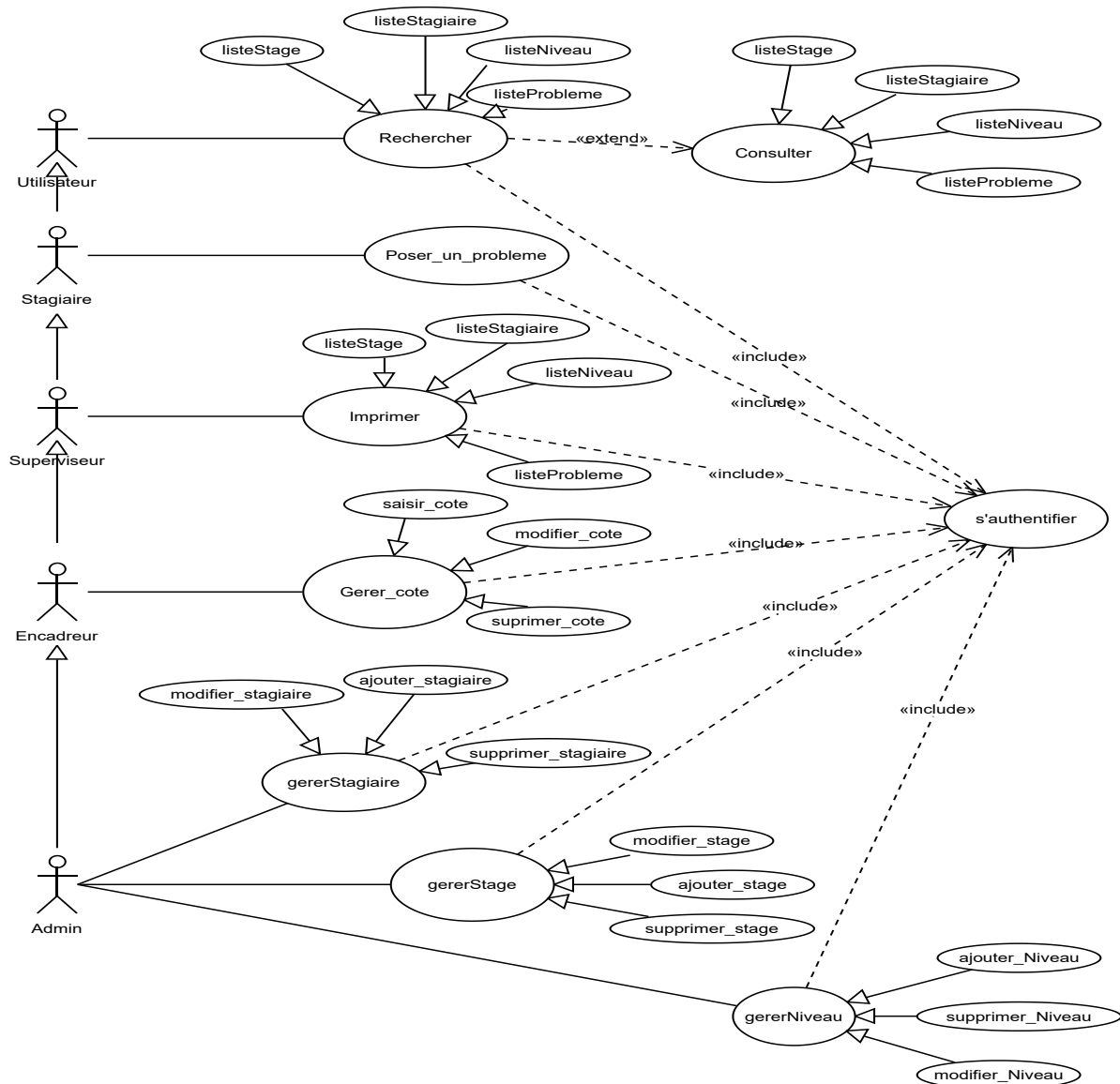


Figure 3: Diagramme de cas d'utilisation

Comme représenté dans la figure ci-haut, nous pouvons dire que notre système a cinq acteurs dont les rôles sont décrits comme suit :

1. **UTILISATEUR** : Il est toute personne qui exerce une fonction sur le système. Il s'authentifie dans le système, fait la recherche et consulte la liste des stagiaires, la liste des stages, la liste des niveaux et la liste des problèmes soumis par les utilisateurs.
2. **STAGIAIRE** : Il est celui qui doit être évalué pour sanctionner son travail au niveau du BDOM pendant la période de son stage. Il s'authentifie, fait la recherche, consulte et soumet un problème
3. **SUPERVISEUR** : Il est l'enseignant qui est chargé de récolter les cotes des stagiaires à partir du système. Il est au niveau de l'UAC comme client. Il s'authentifie, fait la recherche, consulte, soumet un problème et imprime la liste des stagiaires, la liste des stages, la liste des niveaux et la liste des problèmes soumis par les utilisateurs.
4. **ENCADREUR**: Il est celui qui suit les stagiaires au niveau du BDOM, il utilise le serveur pour gérer les cotes des stagiaires. Il s'authentifie, fait la recherche et consulte comme le fait le Superviseur, soumet un problème, imprime les listes publiées, saisit la cote du stagiaire, la modifie ou supprime.
5. **ADMINISTRATEUR** : Il est celui qui va gérer toutes les fonctionnalités du système. C'est lui qui enregistre ou modifie ou encore supprimer les différents stages, les différents étudiants, les différentes promotions, départements et facultés. Ce rôle peut être associé à celui de l'encadreur. Il s'authentifie, crée le compte des utilisateurs, modifie ce compte, le supprime, fait la recherche, consulte toutes les listes sur l'application, soumet un problème, imprime les listes publiées, saisit la cote du stagiaire, la modifie ou la supprime en cas de besoin.

3.3.3. Diagramme de séquences

Cette section présente toutes interactions des acteurs avec les objets du système. Il est question d'illustrer comment les cas d'utilisation exprimés ci-haut sont exécutés par des auteurs. Nous aurons dans ce point les cas d'utilisation s'authentifier, rechercher, imprimer, saisir_cote et ajouter.

3.3.3.1. Cas d'utilisation s'authentifier

La figure ci-dessous illustre l'authentification d'un utilisateur du système. C'est l'une des façons de sécuriser l'application contre des intrus malveillants. L'utilisateur envoie ses identifiants au système. Celui-ci regarde dans sa base des données s'il y a une conformité et

donne accès au client. Dans les cas contraire, il demande au client de vérifier ses identifiants avant de réessayer.

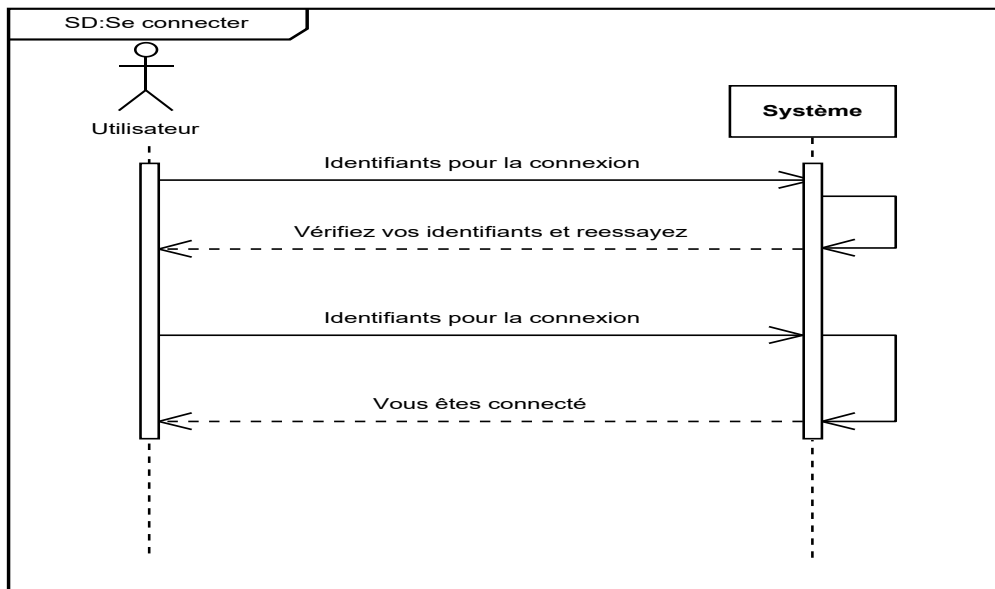


Figure 4: Diagramme de séquence du cas d'utilisation se connecter

3.3.3.2. Cas d'utilisation rechercher

La figure ci-après donne accès à l'utilisateur de faire sa recherche sur l'application. Cet accès est conditionné par l'authentification comme le démontre la figure précédente. C'est après cette étape que l'utilisateur peut procéder par la recherche de ce qu'il veut trouver sur le site.

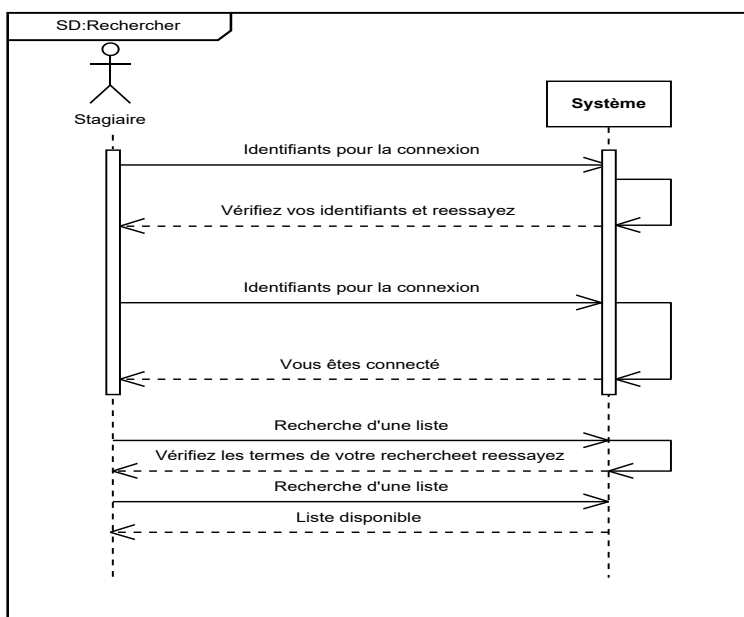


Figure 5: Diagramme de séquence du cas d'utilisation Rechercher

3.3.3.3. Cas d'utilisation imprimer

Comme le Superviseur a besoin de récolter la cote obtenue par le stagiaire, il peut avoir besoin de l'imprimer pour le jury de délibération. La figure ci-dessous montre comment il peut procéder. Après accès au système, il fait la recherche d'une liste et sélectionne celle dont il a besoin et l'imprime.

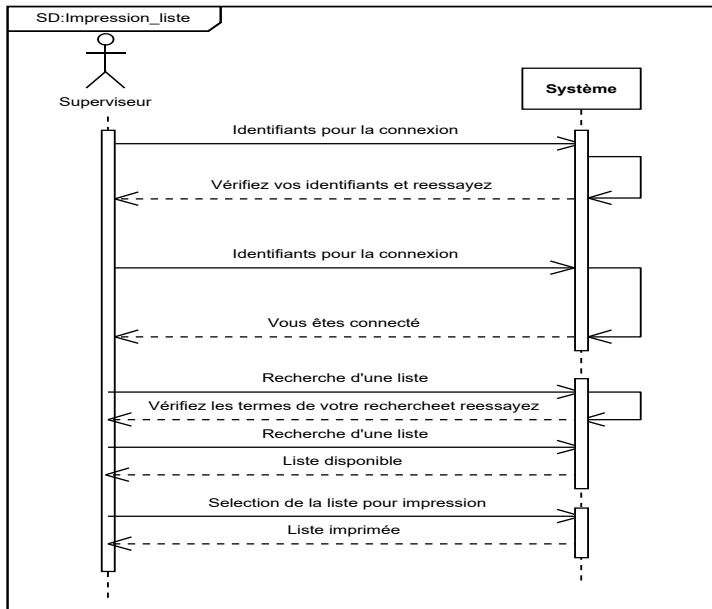


Figure 6: Diagramme de séquence du cas d'utilisation Imprimer

3.3.3.4. Cas d'utilisation saisir_cote

La figure reprise ci-dessous montre comment l'encadreur peut passer à l'évaluation des stagiaires. Comme tous les autres utilisateurs, il doit être identifié avant de faire quoi que ce soit sur le système. Après accès il peut consulter toutes les listes des stagiaires et sélectionner le stagiaire à évaluer. Il saisit sa cote avant de la valider.

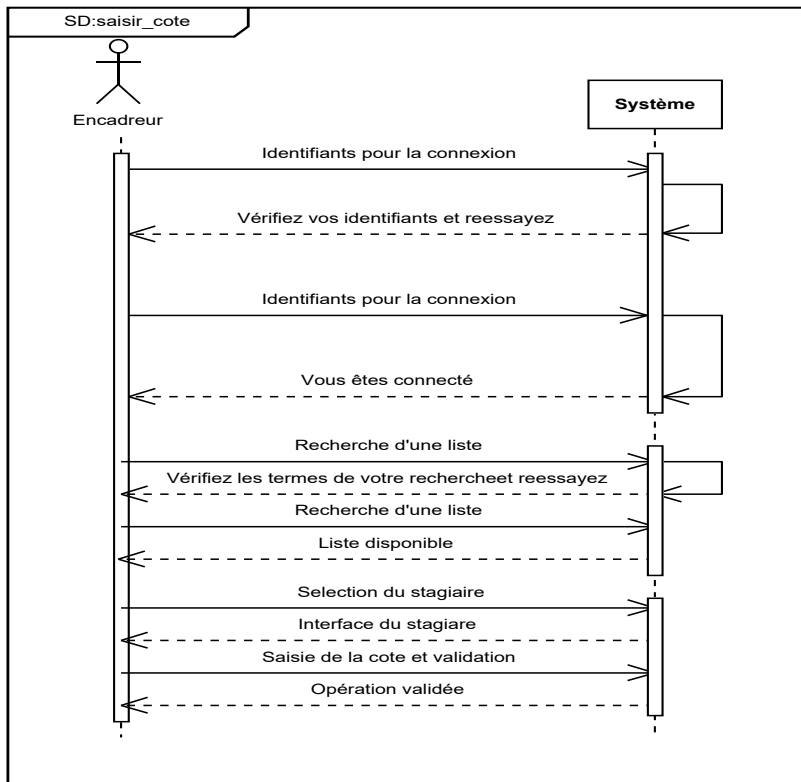


Figure 7: Diagramme de séquence du cas d'utilisation Saisir_cote.

3.3.3.5. Cas d'utilisation ajouter

La figure ci-dessous concerne l'administrateur du système. C'est lui qui est chargé de gérer tout le système. C'est lui qui crée des listes des stages, des stagiaires et aussi des niveaux. Il doit, lui aussi comme tous les autres utilisateurs, se connecter avant d'utiliser le système.

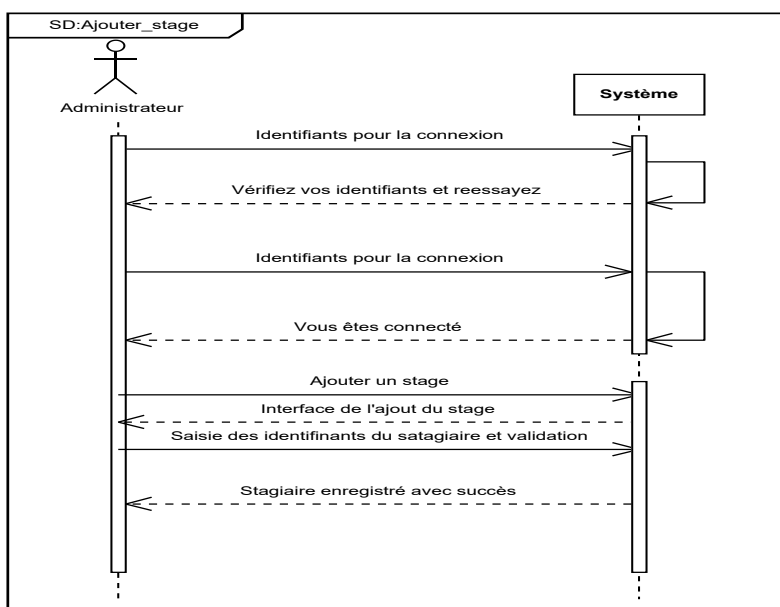


Figure 8: Diagramme de séquence du cas d'utilisation Ajouter.

3.3.4. Diagramme de classes

Dans le langage UML, une *classe* représente un objet ou un ensemble d'objets qui partagent une structure et un comportement communs. Les classes ou instances de classes, sont des éléments de modèle communs dans les diagrammes UML. Une classe identifie les attributs, les opérations, les relations et la sémantique que les instances ou objets de la classe possèdent. Chaque objet qui instancie une classe a ses propres valeurs d'attribut⁴⁸. Une classe est un type abstrait caractérisé par des propriétés comme les attributs et les méthodes communes à un ensemble d'objets et qui permettent de créer des objets ayant ces propriétés⁴⁹. Ceci nous conduit à entrevoir un schéma conceptuel de notre modèle qui tient compte de tous les acteurs de notre système avec leurs interactions leurs spécificités.

3.3.4.1. Schémas conceptuel

Aujourd'hui, l'approche UML semble remplacer les méthodes classiques utilisées en matière de conception de système d'information d'entreprise. En effet, UML permet la spécification des structures statiques et dynamiques des systèmes d'information et apparaît donc comme une approche complète dans le processus de conception et de réalisation de ces outils⁵⁰. Une association exprime une connexion sémantique bidirectionnelle entre deux classes. L'association est instanciable dans un diagramme d'objets ou de collaboration, sous forme de liens entre objets issus de classes associées. L'association représente une relation entre plusieurs classes. Elle correspond à l'abstraction des liens qui existent entre les objets dans le monde réel⁵¹. Les multiplicités ou cardinalités et les rôles des objets participant aux relations complètent la description d'une association. Les exemples d'associations sont donnés directement dans les diagrammes de classe d'UML. L'agrégation est une forme particulière d'association entre plusieurs classes. Elle exprime le fait qu'une classe est composée d'une ou plusieurs autres classes. La relation composant-composé ou la relation structurelle représentant l'organigramme d'une entreprise sont des exemples types de la relation d'agrégation⁵².

⁴⁸Cf. Maxim Sergievski & Ksenia Kirpichnikova, "Optimizing UML Class Diagrams", *ITM Web of Conferences* 18, 03003 (2018), 2018. <https://doi.org/10.1051/itmconf/20181803003>.

⁴⁹Cf. PIECHOCKI Laurent, *UML. Le langage de modélisation objet unifié*, Laurentpiechnocki. developpez. com, 2007, p.27.

⁵⁰Cf. DAVOINE Paule-Annick, *Modélisation UML et outils pour l'analyse spatiale*, Institut Polytechnique de Grenoble, Grenoble, p.2.

⁵¹Cf. PIECHOCKI Laurent, *Op.Cit.*, p. 29.

⁵²Cf. GABAY Joseph et GABAY David, *Op.Cit.*, p.3.

Pour notre cas, il y a une classe d'association appelée Affectation qui relie l'encadreur au programme et à une entreprise donnée. L'encadreur doit affecter un stagiaire si seulement il y a un programme le concernant et approuvé par une entreprise donnée. Cette dernière va être visitée par le Superviseur pour la récolte de l'évaluation du Stagiaire y affecté. Une deuxième classe association dénommée Inscription relie le Stagiaire à un Niveau, et celui-ci précise l'orientation du stagiaire. Une troisième association qui est l'Evaluation, relie l'Inscription à l'Affectation pour montrer que seul un stagiaire inscrit et affecté dans une entreprise à un semestre bien déterminé doit être évalué par le Superviseur.

D'autres classes participent à l'association pour réaliser le service demandé. Nous avons par exemple la classe programme qui détermine le programme arrêté par l'académie selon le niveau du stagiaire et soumis à des entreprises. La classe Superviseur contient l'identité de l'envoyé de l'UAC qui doit être chargé de récolter les cotes des stagiaires, de ce fait, un superviseur peut suivre un ou plusieurs stagiaires à la fois; la classe stagiaire contient elle, l'identité du stagiaire qui doit être évalué, ici un ou plusieurs stagiaires peuvent être inscrits dans un ou plusieurs niveaux ; la classe semestre indique à quel semestre le stagiaire a effectué son stage, une évaluation peut concerner un ou plusieurs semestres.

La figure ci-après récapitule toutes les classes que nous avons considérées pour implémenter notre système. Le diagramme comprend onze classes dont trois classes d'association telles que nous venons de le souligner dans les paragraphes précédents.

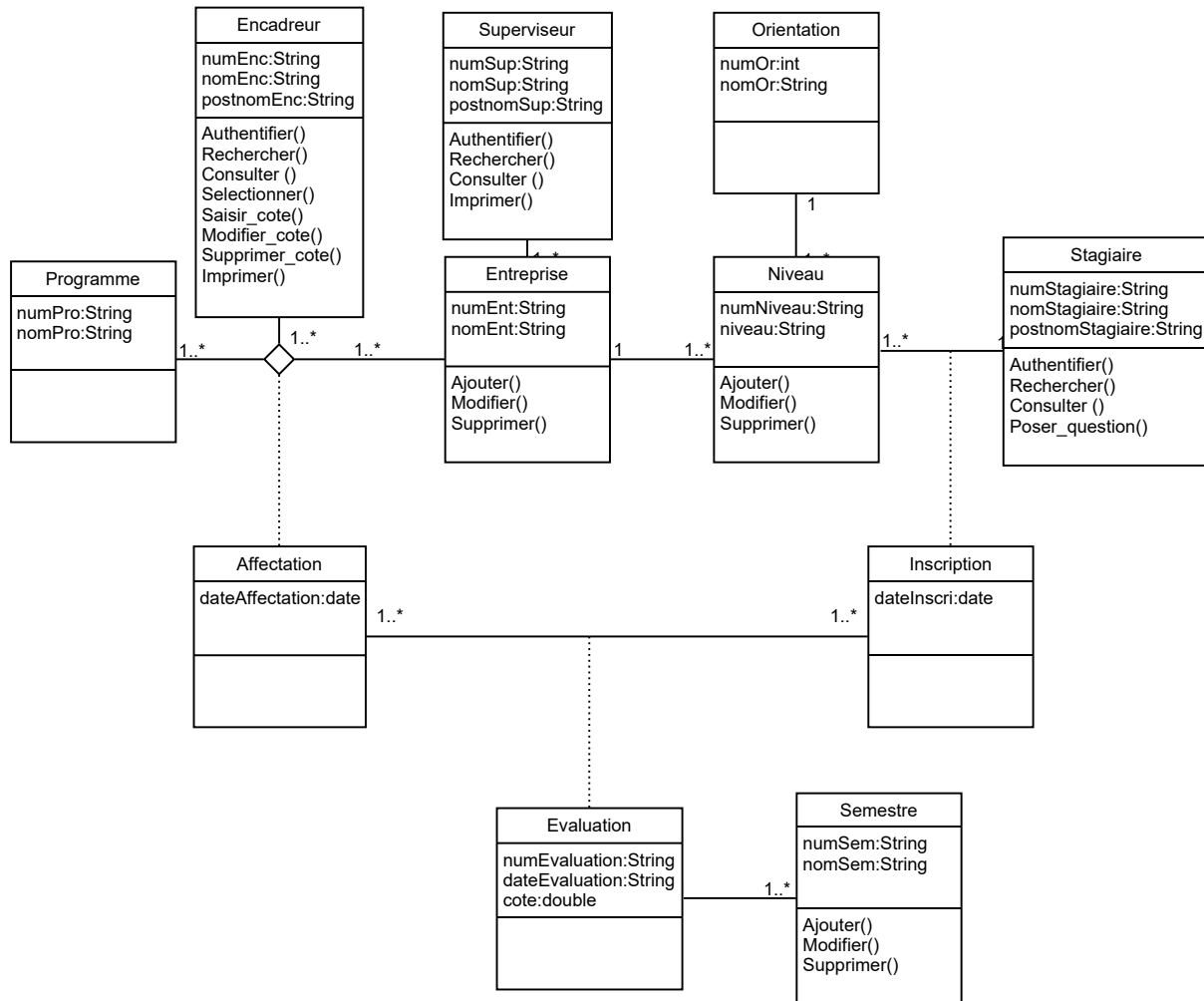


Figure 9: Diagramme de classe

3.3.4.2. Schéma relationnel

Ce schéma nous représente la structure interne de notre base de données telle qu'elle se présente au niveau de notre serveur. Ici sont reprises toutes les classes fonctionnelles avec leurs attributs. Ainsi dit, notre schéma se présente de la manière suivante :

Encadreur (numEnc, nomEnc, postnomEnc)

Entreprise (numEnt, nomEnt)

Superviseur (numSup, numEnt#, nomSup, postnomSup)

Stagiaire (numStagiaire, nomStagiaire, postnomStagiaire)

Programme (numPro, nomPro)

Niveau (numNiveau, niveau)

Inscription (numInscription, numNiveau#, numStagiaire#, dateInscription)

Affectation (numAffectation, numPro#, numEnc#, numEnt#, dateAffectation)

Orientation (numOrientation, numNiveau#, nomOrientation)

Semestre (numSemestre, nomSemestre)

Evaluation(numEvaluation, numAffectation#, numInscription#, numSemestre #,
dateEvaluation, cote)

Conclusion

Dans ce chapitre, nous avons présenté la méthodologie utilisée pour notre recherche. Nous avons adopté le langage UML. Nous avons, pour ce fait, utilisé les principaux diagrammes tels que : le diagramme de cas d'utilisation, le diagramme de séquence, et le diagramme de classe pour notre modélisation. Nous avons procédé aussi au modèle logique de données qui nous a permis d'exploiter notre base des données qui est une base de données relationnelle. C'est cette dernière qui est implémentée dans notre serveur et dont les résultats sont repris dans le quatrième chapitre consacré à la réalisation technique de notre système.

Quatrième chapitre : ANALYSE DES DONNEES, RESULTATS ET DISCUSION

4.0. Introduction

Ce quatrième chapitre est une sorte de mise en application de tout le processus du troisième chapitre qui nous a démontré, au moyen des diagrammes, la méthode à suivre pour réaliser notre projet. Il sera question ici de présenter les résultats obtenus après un long moment de recherche sur la mise au point d'un service web. Le but poursuivi est en fait, une implantation d'une application service web pour la publication des résultats des stagiaires. Voilà pourquoi ce chapitre va se concentrer à nous relater le processus qui a conduit au résultat final de notre travail.

Ce chapitre comprend deux grands points. Dans le premier point, il sera question de présenter les résultats de notre recherche, d'exposer les différentes étapes qui ont piloté le processus d'élaboration de notre Système. Tout cela se fera au moyen des certaines captures d'écrans qui visualisent le travail accompli. Le deuxième point sera celui de la discussion de nos résultats.

4.1. RESULTATS DE LA RECHERCHE

Dans cette section, nous présentons les résultats de nos recherches en fonction des objectifs spécifiques poursuivis dans cette étude tels que annoncés au premier chapitre. Au début de ce travail, nous nous sommes fixés quatre objectifs. Le premier consiste à identifier les acteurs de notre système, le deuxième est l'implantation du serveur au niveau du BDOM, le troisième consiste à configurer un client au niveau de l'UAC pour consommer les différents services de l'application déployée et le quatrième consiste à tester et valider les prototypes développés.

4.1.1. Identifier les différents acteurs du système proposé

Le premier objectif était d'identifier les différents acteurs qui interagissent avec le système en utilisant le langage de modélisation UML. Pour ce faire, nous nous sommes posé la question suivante: Quels sont les acteurs susceptibles d'interagir avec le système à développer? Nos recherches nous ont conduits à cibler cinq principaux acteurs qui doivent interagir pour utiliser notre système. Au troisième chapitre, nous avons présenté ces acteurs au

moyen de certains diagrammes pour montrer le rôle de chacun et les différentes interactions entre eux. Ainsi, ces acteurs repris ci-dessous sont ceux retenus dans cette étude.



Le premier acteur est l'utilisateur ordinaire. Celui-ci est toute personne qui a un compte sur l'application. Il doit s'identifier pour aboutir à son interface. C'est à partir des informations recueillies sur l'application qu'il peut choisir de changer de statut. Cet utilisateur doit avoir un identifiant et un mot de passe. Il peut faire la recherche et consulter les publications telles que les stages et les universités disponibles. Il ne peut rien faire d'autre. Il peut contacter l'administrateur par d'autres moyen de communication pour soumettre sa requête, il sera ainsi enregistré conformément à sa requête à l'administrateur.



Le deuxième acteur est le stagiaire lui-même. Celui-ci est comme tout simple utilisateur puisque lui aussi avant toute chose, il s'authentifie sur l'application au moyen de son Identifiant et de son mot de passe. C'est après qu'il peut passer à la recherche, à la consultation des publications et à soumettre son problème s'il y a besoin. Celui-ci a un identifiant comme clé primaire, un nom, un postnom, le genre et l'âge. Il est envoyé par l'UAC au BDOM pour passer son stage. Ce stagiaire est soumis à un programme proposé par l'UAC au BDOM. Il doit indiquer son domaine d'application, son orientation, le niveau de sa formation et son âge. Cet acteur peut interagir avec l'encadreur en soumettant un problème. Il peut s'agir du problème concernant son identification dans le logiciel ou le déroulement de son stage.



Le troisième acteur est le Superviseur. Celui-ci est mandaté par l'UAC pour le suivi des stagiaires sur leur lieu de stage. Il doit se rendre compte du déroulement du stage et c'est lui-même qui est chargé de récolter les points obtenus par chacun des stagiaires. Comme le deux précités, il doit s'authentifier avant de parvenir à son interface. Il peut consulter toutes les publications et imprimer la liste des stagiaires. Il a un identifiant et un nom. Pour réaliser son travail il n'a plus besoin, comme dans le passé, de faire des déplacements pour aller récolter la cote obtenue par l'étudiant ou de charger l'étudiant de remmener sa cote ; ce qui causait parfois de désengorgements énormes. La cote lui parvient directement dans son bureau au moyen de l'application implémentée.



Le quatrième acteur est l'encadreur. Celui-ci est détaché par l'entreprise pour suivre les stagiaires pendant leur période de stage. Comme tous les autres acteurs, il a un identifiant, un nom et un postnom. Il est chargé d'attribuer la cote au stagiaire soumis à responsabilité. Il peut consulter les publications, imprimer les listes publiées et faire la recherche d'un stagiaire ou d'un stage publié.



Le dernier acteur est l'administrateur. Celui-ci est chargé du suivi global de l'application. C'est lui qui s'occupe de l'enregistrement de tous les acteurs sur l'application. Il peut supprimer tout ce qui ne cadre pas avec la logique de l'application. Ce rôle peut être joué par l'encadreur.

4.1.2. Implémenter un serveur offrant des méthodes appelables

Le deuxième objectif spécifique de notre travail est d'arriver à implémenter un serveur qui offre des méthodes appelables (au sein du BDOM) en distance via le SOAP qui retourne les cotes de stage des étudiants de l'UAC. Pour satisfaire cet objectif, nous nous sommes posé cette question: Quelle technologie existante permet de créer des services web et des clients qui communiquent au moyen d'un langage simple tel que eXtensible Markup Language (XML)? Pour y parvenir, nous nous sommes servis du logiciel éclipse afin d'implémenter nos différentes classes. Ces différentes classes nous ont permis de concevoir notre serveur à partir des méthodes implémentées dans notre classe service comme nous pouvons le constater sur ces images.

Cette première image est une classe qui fournit les attributs du stagiaire tel que son matricule, son nom et sa cote. Ces attributs étant en privé nécessitent des getters et des setters pour être accessibles.

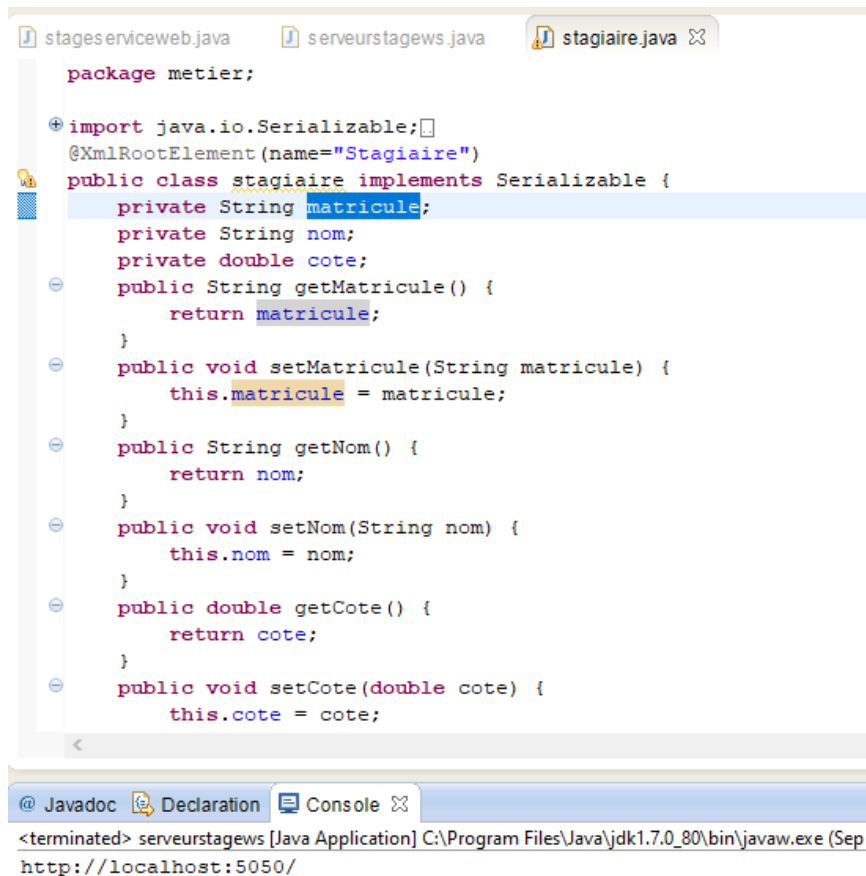


Figure 10: Classe étudiant

La classe est composée d'attributs qui déterminent cet objet. Et comme ces attributs sont en privé pour y accéder il nous a fallu générer des getters et des setters. Nous avons eu aussi besoin d'ajouter des annotations, comme *XmlRootElement*, pour que les objets puissent communiquer entre eux selon les normes du langage JaxWS que nous avons adopté. Cette deuxième image nous présente les différents getters et les setters de notre classe stagiaire dans le package métier.

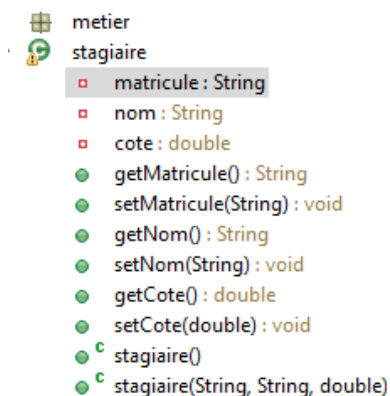


Figure 11: Getters et Setters

Comme souligné plus haut, une autre classe très capitale à cette étape est le serveur. Celui-ci est implanté au niveau du BDOM. Il fournit l'adresse du Web service et on peut déjà évaluer son efficacité par le concepteur au moyen des premiers tests appelés test boîte noir. C'est ce serveur qui fournit l'adresse qui pointe sur le document principal du web service, le WSDL écrit en langage XML.

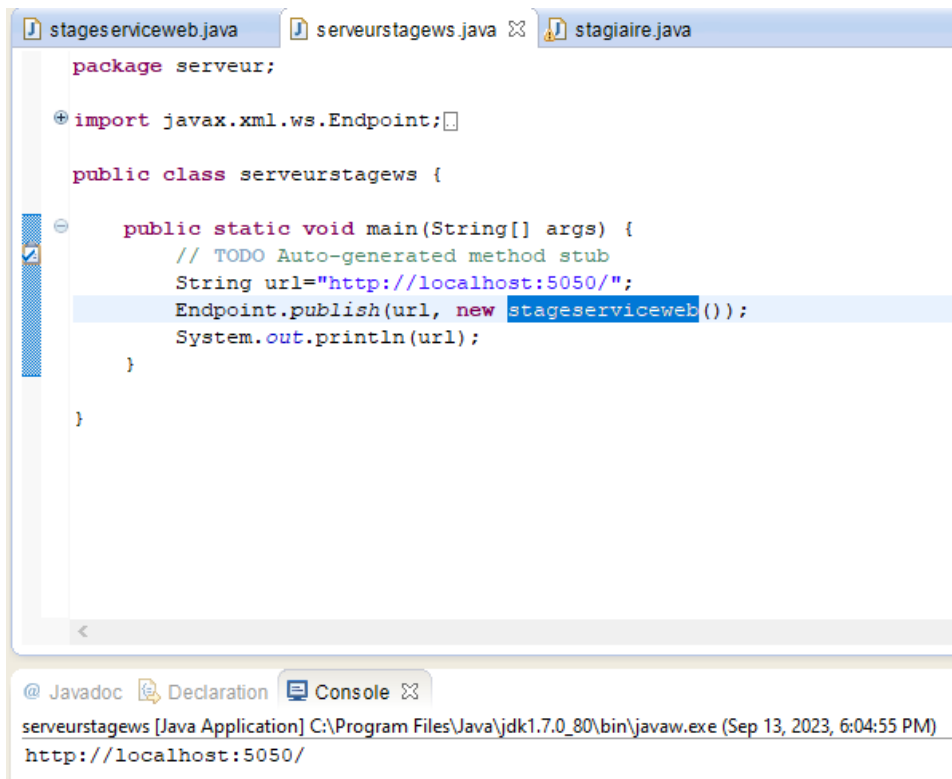


Figure 12: Serveur du web service

Nous venons de le souligner dans paragraphe précédent, le document WSDL est essentiel pour toute application Web Service. C'est lui qui constitue le contrat entre le fournisseur du service et le demandeur. Ce document comprend toutes les méthodes, appelées aussi opérations, qui sont autorisées pendant l'échange entre ces partenaires précités. Il indique aussi le port utilisé, le format de l'enveloppe SOAP qui doit être encapsulée et décapsulée pour être accessible, et les différents messages entrant et sortant. En poursuivant notre processus, nous sommes parvenus à ce document WSDL tel que le montre l'image ci-après.

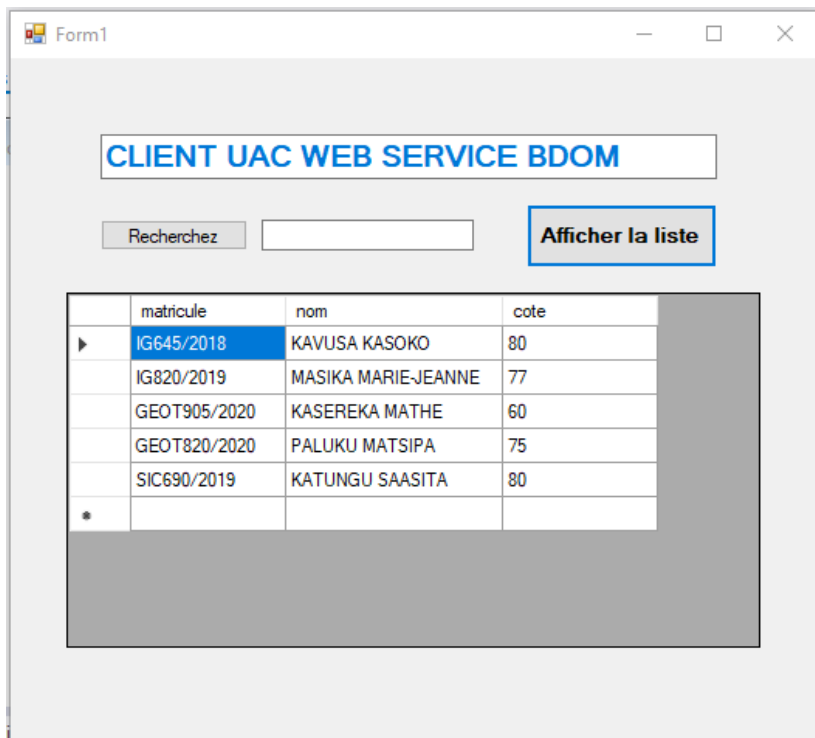


Figure 13: Serveur du web service

Sur l'image ci-dessus, nous constatons que notre Webservice comprend deux méthodes. La première, appelée *getstagiaire()*, est une méthode qui retourne un stagiaire. Le client entre le matricule du stagiaire et tout ce qui concerne le stagiaire s'affiche y compris la cote de stage obtenue. La deuxième méthode est appelée *getListStagiaires()*. Celle-ci consiste à afficher la liste de tous les stagiaires avec leurs cotes.

4.1.3. Implémenter une application cliente

Notre troisième objectif spécifique était de parvenir à implémenter une application cliente au sein de l'UAC qui soit capable de consommer les ressources offertes par le serveur qui se trouve au BDOM. Nous nous sommes posé la question de savoir quel objet, en mode WSDL, peut être capable d'appeler les fonctions SOAP pour les consommer en distance via une technologie web ? Dans la classe service, nous avons implémenté deux méthodes qu'utilise notre application. Notre application est en mesure d'afficher un stagiaire à partir de son numéro matricule. Elle a une deuxième méthode, celle d'afficher la liste de tous les stagiaires chacun avec son résultat. Ces méthodes peuvent être appelées par tous les clients qui sollicitent ce service, et peu importe le langage de programmation qu'ils utilisent. L'image ci-dessous nous présente le résultat obtenu par un client d'une entreprise (UAC) qui utilise C# comme langage de programmation. Même si le service web a été créé en Java, l'UAC le consomme en C-Sharp comme nous pouvons le voir sur l'image ci-après.



	matricule	nom	cote
▶	IG645/2018	KAVUSA KASOKO	80
	IG820/2019	MASIKA MARIE-JEANNE	77
	GEO905/2020	KASEREKA MATHE	60
	GEO820/2020	PALUKU MATSIPA	75
	SIC690/2019	KATUNGU SAASITA	80
*			

Figure 14: Client UAC affichant les résultats des stagiaires.

N.B: Dans la partie création des clients, nous avons développé un client en langage C-Sharp. Sur l'interface de ce client, le Superviseur peut saisir le matricule du stagiaire pour afficher sa cote obtenue pendant son stage. Il peut également cliquer sur le bouton *Afficher la liste* pour avoir la liste de tous les stagiaires, comme illustré ci-haut.

4.1.4. Tester et valider les prototypes développés

Pour être sûrs de nos résultats obtenus, nous sommes passés par diverses simulations. Nous avons faits les premiers tests au moyen de l'outil conçu pour ce genre de test, appelé SoapUI. Celui-ci nous a révélé que notre application marche très bien, comme cela se voit sur les images ci-dessous.

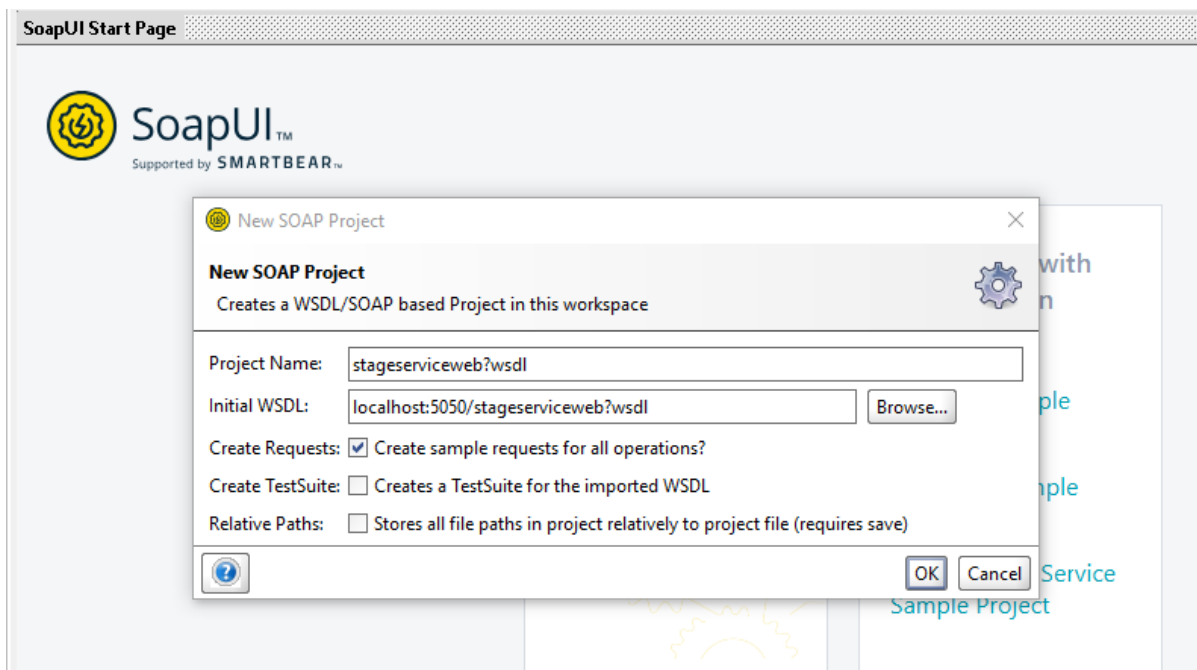


Figure 15: Test de l'adresse WSDL.

Ce test nous présente le web service correspondant à l'adresse indiquée avec toutes ses méthodes. C'est alors qu'il nous est possible de passer au test de ces méthodes qui affichent les stagiaires avec leurs résultats. Il y a donc échange des messages entre fournisseur du service et demandeur de ce service au moyen de l'enveloppe SOAP écrite en XML, comme nous pouvons le voir sur cette image.

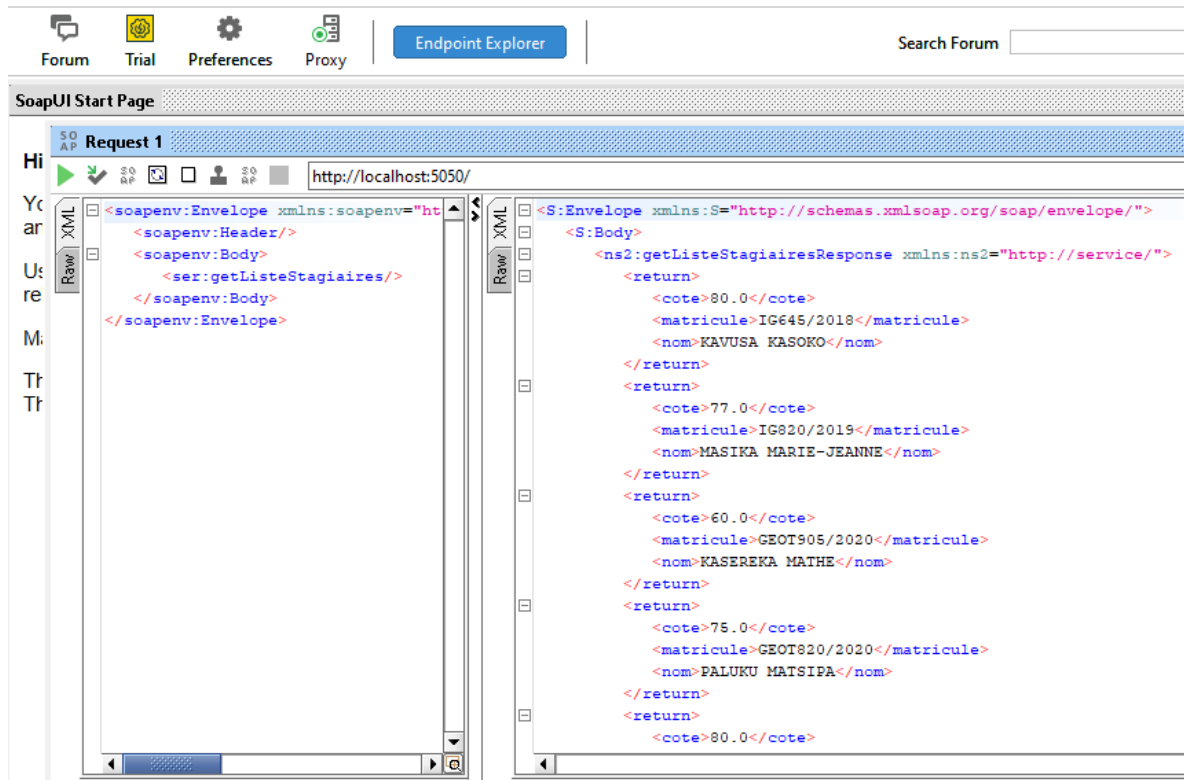


Figure 16: Test de la méthode `getListeStagiaires` par SoapUI

Ce test nous a alors poussés à créer notre utilisateur, le client UAC pour commencer à consommer ces ressources. Ce client est implémenté en C# comme le montre l'image ci-dessous.

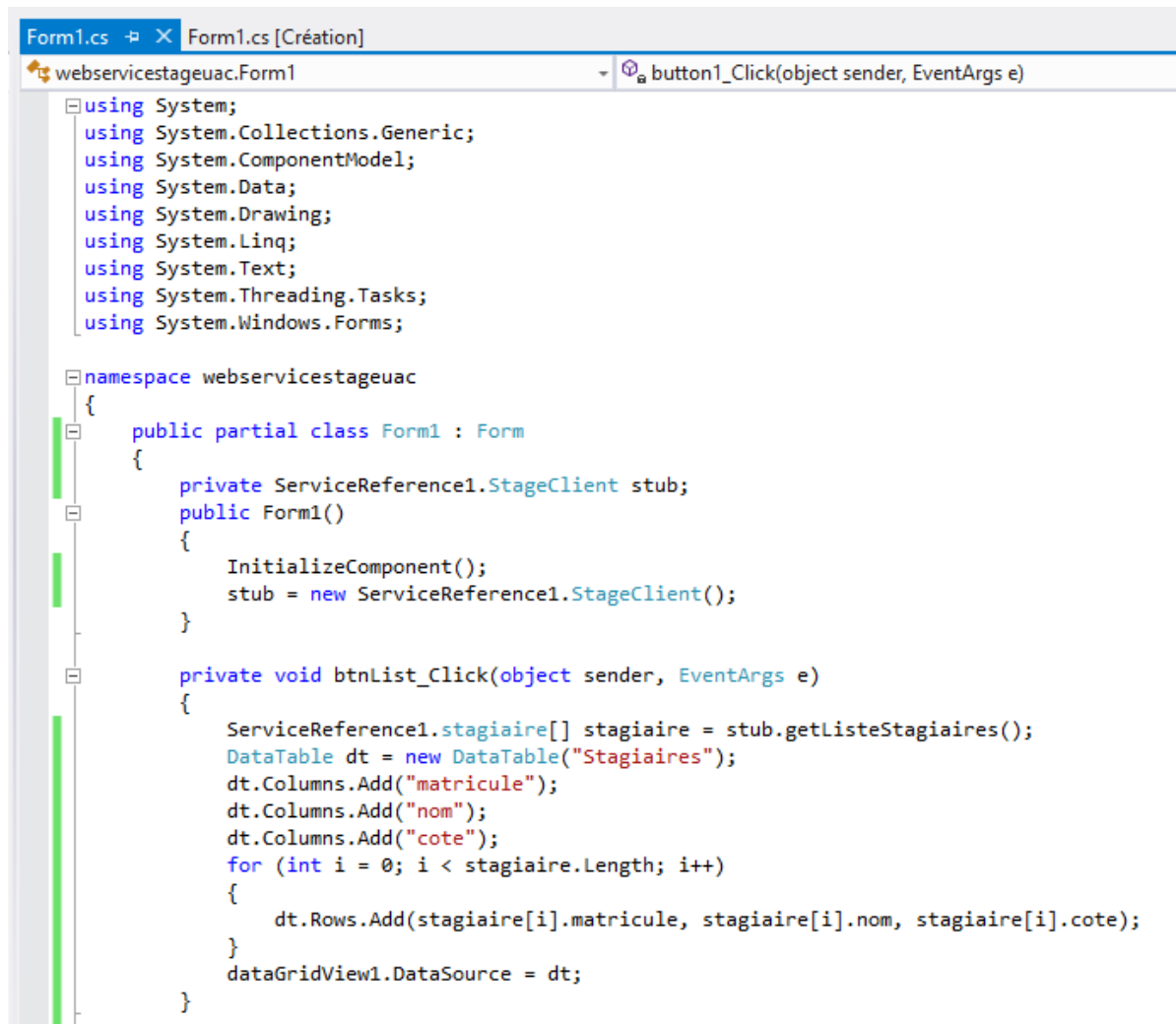


Figure 17: Création du client .Net

4.2. DISCUSSION DES RESULTATS

Ce travail, en tant que fruit d'une recherche scientifique, se situe dans le cadre des autres travaux précédents dans ce domaine des services web. Les travaux qui ont été à notre porté, comme indiqué dans notre premier chapitre, traitent des sujets différents au notre. Nous pouvons citer entre autres Camilo Sarmiento et les autres co-auteurs dans leur article intitulé « *Web Service for Document Management of University Degree Projects* », ont plus insisté sur la mise en ligne des catalogues des TFC, des mémoires et des thèses enfin que les finalistes ne reprennent plus les travaux déjà traités. De même Aftab et ses co-auteurs, dans leur article « *An Implementation of Web Services for Inter-Connectivity of Information Systems* », ont à leur tour créé un mécanisme d'interopérabilité des sous-systèmes dans le système principal de l'université. Etant donné qu'il y avait des campus étendus, il a fallu avoir un système d'intercommunication entre ces différents campus, les bibliothèques, les

départements et même les étudiants. Havaluddin H. et ses co-auteurs, dans leur article intitulé « *A Database Integrated System Based on SOAP Web Service* », ont été emmenés à mettre sur pied une base de données pour la publication des travaux de fin de cycle, des mémoires, des thèses et des articles dans les revues scientifiques. Le but était d'augmenter le nombre de publication dans leur pays, l'Indonésie. Pour finir, citons Hammoudeh et Ajlan qui, dans leur article « *Implementing Web Services Using PHP Soap Approach* », se sont focalisé sur un système d'apprentissage en ligne quel que soit le pays où se retrouve celui qui apprend et quel que soit le type de matériel qu'il utilise.

En somme, nous pouvons préciser que tous ces auteurs ont utilisés les mêmes technologies que nous avons adoptées. Ces technologies sont communes à tous les web services, que ce soit pour le protocole SOAP ou REST. Les technologies utilisées sont entre autres le protocole http, les langages XML, PHP, SQL, UML et JAVA, sans oublier l'élément principal sur lequel tourne le web service, à savoir le document WSDL. Par contre, nous constatons qu'aucun parmi ces auteurs, ne s'est soucié de la cotation en ligne des stagiaires comme nous venons de le réaliser dans ce travail. C'est cela qui marque notre démarcation d'avec nos prédécesseur.

CONCLUSION PARTIELLE

Dans ce chapitre nous avons cherché à appliquer tout ce que le troisième chapitre nous a proposé, à savoir, créer une application web service pour la cotation des stagiaires. Nous avons énuméré différents résultats obtenus conformément à nos quatre objectifs nous assigner au début de ce travail. Nous avons fini ce chapitre en en vérifiant nos résultats obtenus par rapport aux travaux précédents. C'est ainsi que dans le point qui suit, nous allons revenir amplement sur ce que nous avons réalisé et le travail qui reste à faire pour nos successeur dans ce domaine.

Cinquième chapitre : CONCLUSION GENERALE ET RECOMMANDATIONS

5.1. Introduction

Nous sommes pratiquement au terme de notre démarche scientifique. Depuis l'introduction de ce travail, nous avons traversé plusieurs étapes. Nous allons revenir de façon synthétique sur ces différentes étapes. Après ce petit tour d'horizon, nous allons donner notre point de vue sur l'orientation future pour les chercheurs intéressés par ce domaine d'interopérabilité. Autrement dit, dans ce chapitre, nous allons revenir en *grosso modo* sur les conclusions de nos différents chapitres, nous allons aussi présenter notre contribution aux travaux précédents avant de chuter sur quelques recommandations.

5.2. Conclusions

L'internet, depuis son apparition vers les années 90⁵³, ne cessent de se développer et de modifier le cours du monde. Chaque jour qui passe, il y a une nouveauté qui naît pour faciliter telle ou telle façon de faire. Notre étude s'est attelée sur une nouveauté, à savoir, l'interopérabilité des systèmes au moyen d'un service web. Et cela s'est basé sur un système de cotation en ligne des stagiaires de l'UAC dans une entreprise tierce. Dans le cas échéant, nous avons ciblé les Bureau Diocésain des Œuvres Médicales (BDOM). Le système mis sur pied est un produit d'un long processus. Ce dernier avait commencé son envol par l'intérêt accordé à ce nouveau système. Nous avons trouvé qu'il était nécessaire d'avoir cet outil pour notre université enfin de palier au problème de perturbations quant à l'obtention des résultats des stagiaires.

Comme nouveau domaine à l'heure actuellement, il nous a fallu élaguer le terrain par la définition de certains concepts remarquable en Web service mais aussi par la présentation de certains auteurs qui nous ont précédés sur ce terrain. Nous avons eu comme constat qu'à l'heure actuelle aucune entreprise ne peut plus se féliciter d'être enfermée sur elle-même. Il faut à nos jours mettre en pratiques les indications du célèbre historien Koyré dans son livre

⁵³ Cf. BECKOUCHE Pierre, La révolution numérique est-elle un tournant anthropologique?, *Le débat*, 2017, Vol. 193, no. 1, p. 153-166.

« Du monde clos à l'univers infini »⁵⁴. Nous devons comprendre qu'avec la mondialisation et actuellement avec la virtualisation de notre monde, on ne peut plus se taper la poitrine d'être seul dans ce qu'on fait. L'interopérabilité et même l'interdépendance sont capitales pour aller plus loin.

Pour arriver à réaliser notre application, il a fallu passer par plusieurs techniques. Nous avons d'abord pensé aux croquis qui visualisent notre œuvre. Ce qui s'est réalisé au moyen du langage UML avec ses différents diagrammes. Ces derniers nous ont donné une idée sur l'interconnexion et la collaboration de nos différentes classes. Ce qui a enfin donné lieu à la conception de notre Base de données par le biais du schéma relationnel.

C'est le schéma relationnel qui a fourni des bases pour notre modélisation technique de l'application. Plusieurs techniques sont intervenues à ce niveau pour concrétiser nos objectifs nous assigner et arriver à satisfaire les besoins de nos clients. Nous avons utilisé plusieurs outils pour créer les différentes classes à déployer. Nous avons créé le serveur pour nous fournir l'adresse du WSDL. Nous avons testé cette adresse avec l'outil SoapUI et procédé aux premiers tests de la fonctionnalité de notre web service. En fin, ce test du web service nous a conduits à créer notre client pour commencer à consommer ce web service.

5.3. Contributions aux travaux précédents

Nos objectifs ont été atteints puisqu'il nous a fallu aboutir à la réalisation de l'application capable de faire collaborer les deux entreprises ciblées, à savoir l'UAC et le BDOM. Les acteurs en interaction ont été identifiés dans le troisième chapitre consacré à la méthodologie utilisée. Cette méthode a été concrétisée dans le quatrième chapitre en répondant à nos différents objectifs relevés au premier chapitre. Le client créé au bout de cette démarche jouit complètement des ressources proposées par le web service sans tenir compte de son langage de création.

Cet outil vient suppléer au désagrément constaté pendant la collecte des résultats des stagiaires. C'est un ajout considérable pour l'UAC et ses partenaires, c'est-à-dire les entreprises qui collaborent avec l'UAC en accueillant des étudiants qui leur arrivent pour

⁵⁴ Cf. CANGUILHEM Georges, *Études d'histoire et de philosophie des sciences*, Vrin, 1983.

passer leur période de stage au sein de leurs entreprises. Un ajout aussi à toutes les entreprises désireuses de l'interopérabilité de leurs services.

5.4. Recommandations

Nous sommes certains que nous n'avons pas atteint le sommet de notre recherche dans ce domaine d'interopérabilité. Le terrain est encore très vaste à défricher. Du jour au lendemain, les entreprises sont en train de naître les unes après les autres. Ceci fait naître à son tour un besoin accru de collaboration pour que ces entreprises se maintiennent dans le temps. C'est un terrain encore ouvert à tous les chercheurs soucieux d'y apporter leur contribution. Le préalable est de connaître le fonctionnement de ces différentes entreprises enfin de les aider à quitter leur cloisonnement pour s'ouvrir à l'univers infini.

Nous recommandons à nos collègues chercheurs de découvrir davantage le système de la révolution scientifique actuelle, le Cloud Computing⁵⁵ et l'intégrer dans nos entreprises locales. Ceci semble encore un terrain vierge pour notre milieu. Nous sommes des humains et notre œuvre ne doit être qu'une œuvre humaine. Nous sollicitons nos successeurs dans la science à améliorer là où nous avons eu des lacunes pour parfaire cette œuvre.

⁵⁵ Cf. ATLAM Hany F., ALENEZI Ahmed, ALHARTHI Abdulrahman, *et al.* Integration of cloud computing with internet of things: challenges and open issues, In : *2017 IEEE international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCoM) and IEEE smart data (SmartData)*, IEEE, 2017, pp. 670-675.

Bibliographie

A. Ouvrages

1. BEDIN Véronique (dir.), *L'évaluation à l'université*, Presse universitaire de Rennes, 2009, p. 2.
2. BLANC Xavier, et al., *UML 2 pour les développeurs*, Eyrolles, Paris, 2006, p. VII.
3. CANGUILHEM Georges, *Études d'histoire et de philosophie des sciences*, Vrin, 1983.
4. DAVOINE Paule-Annick, *Modélisation UML et outils pour l'analyse spatiale*, Institut Polytechnique de Grenoble, Grenoble, p.2.
5. GABAY Joseph, GABAY David, *UML 2. Analyse et conception*, Dunod, Paris, 2008, p.61.
6. PIECHOCKI Laurent, *UML. Le langage de modélisation objet unifié*, Laurentpiechnocki. developpez. com, 2007, p.27.
7. ROQUES Pascal, *Le cahier du programmeur. UML 2. Modéliser une application web*, 4^e édition, Paris, Eyrolles, s.d, p.2.

B. Articles

1. AMANO Tsuyoshi, et al., "Rethinking the prototyping process for applying design thinking to business model innovation", *Conference Proceedings of the Academy for Design Innovation Management*, Vol. 1, no. 1, 2019, p. 1208.
2. ATLAM Hany F., ALENEZI Ahmed, ALHARTHI Abdulrahman, et al., Integration of cloud computing with internet of things: challenges and open issues, In : *2017 IEEE international conference on internet of things (iThings) and IEEE green computing and communications (GreenCom) and IEEE cyber, physical and social computing (CPSCom) and IEEE smart data (SmartData)*, IEEE, 2017, pp. 670-675.
3. BALAJI N., et al., "Enhancements for UDDI using User Preferential Web Service Selection Model based on SLA", *International Journal of Engineering and Technology (IJET)*, Vol. 5, no. 5, 2013, p. 4057.
4. BECKOUCHE Pierre, La révolution numérique est-elle un tournant anthropologique?, *Le débat*, 2017, Vol. 193, no. 1, p. 153-166.

5. BHUKYA Devi Prasad , Reeta Sony A.L & Gautam Muduganti, “On Web Services Based Cloud Interoperability”, *International Journal of Computer Science Issues*, Vol. 9, no. 5, 2012, p. 234.
6. BINDER Jens F., et al., "The academic value of internships: Benefits across disciplines and student backgrounds", *Contemporary Educational Psychology*, Vol. 41, 2015, pp.73-74.
7. CHANDIO Aftab, et al., "An Implementation of Web Services for Inter-Connectivity of Information Systems", *International Journal of Computing and Digital Systems*, Vol. 3, no. 3, 2014, pp. 1-3.
8. CHEN S.C & YANG S.Y., “A Web Services, Ontology and Big Data Analysis Technology-Based Cloud Case-Based Reasoning Agent for Energy Conservation of Sustainability Science”, *Applied Sciences*, Vol. 10, no. 4, 1387, 2020.
9. EHSAN Adeel et al., « RESTful API Testing Methodologies: Rationale, Challenges, and Solution Directions », *Applied Sciences* 12, no 9 (Janvier 2022): 4369.
10. HAMMOUDEH M.A.A. & Al-Ajlan, A.S., “Implementing Web Services using PHP Soap Approach. *International Journal of Interactive Mobile Technologies (iJIM)*, Vol. 14, no. 10, 2020, pp. 35–37.
11. HAVILUDDIN H., et al., “A Database Integrated System Based on SOAP Web Service”, *TEM Journal*, Vol. 8, no. 3, 2019, p. 782.
12. J.BIGELOW ET Stephen Gaétan Raoul, « Quels sont les types d’API et leurs différences ? », in LeMagIT, <https://www.lemagit.fr/conseil/Quels-sont-les-types-dAPI-et-leurs-differences>.
13. KAMBALE KASAMBYA Moïse, et al., "Déploiement mobile d’un réseau de neurones profond pour l’identification des étudiants de l’UAC/ISDA en salle d’examens", *International Journal of Innovation and Applied Studies*, Vol. 39, 2023, p. 137.
14. KAMBALE KASAMBYA Moïse, Héritier Mpia, et Vingi Baraka, « Déploiement mobile d’un réseau de neurones profond pour l’identification des étudiants de l’UAC/ISDA en salle d’examens », *International Journal of Innovation and Applied Studies* Vol.39 (1 mars 2023): 136-55.
15. KARJI A., et al, ”Evaluation of an Interview-Based Internship Class in the Construction Management Curriculum: A Case Study of the University of Nebraska-Lincoln”, *Education Sciences*, Vol. 10, no. 4, 109, 2020.
16. KIAH M.L.M., et al., “An Enhanced Security Solution for Electronic Medical Records Based on AES Hybrid Technique with SOAP/XML and SHA-1 ”, *J.MedSyst*, Vol.37, 2013, 9971.

17. KRAUS Sascha et al., "Literature Reviews as Independent Studies: Guidelines for Academic Practice", *Review of Managerial Science*, Vol. 16, no. 8, 2022.
18. MALA Goplani, AKASH Gupta, & JEWEL Sabhani, "A study on impact of internship on regular studies of undergraduate students", *International Journal of Advance and Innovative Research*, Vol. 7, no.1, 2020, p. 92.
19. MISHRA Sugyan Kumar & SARKAR Anirban, "Service-oriented architecture for Internet of Things: A semantic approach", *Journal of King Saud University – Computer and Information Sciences*, Vol. 34, no. 1, 2021.
20. MOSCARDINI Alfredo , STRACHAN Rebecca & Vlasova T., "The role of universities in modern society", *Studies in Higher Education*, Vol. 47, no. 2, 2020.
21. NACHEVA Radka, "Prototyping approach in user interface development", in *2nd conference on innovative teaching methods (ITM 2017) 28-29 June 2017*, University of economics varna, Bulgaria, 2017, p. 80.
22. PANIAGUA Cristina, "Service Interface Translation. An Interoperability Approach", *Applied Sciences*, Vol. 11, no. 24, 11643, 2021.
23. POLENGHI Adalberto et al., "Enterprise information systems interoperability for asset lifecycle management to enhance circular manufacturing", *IFAC-PapersOnLine*, Vol. 5, no. 1, 2021, p. 361.
24. SAEED Ahmad, ALI Sehrish, WAQAR, Nazish, et al. Comparative evaluation of the maintainability of RESTful and SOAP-WSDL web services. In : *2023 International Conference on Business Analytics for Technology and Security (ICBATS)*. IEEE, 2023. p. 1-9.
25. SARMIENTO Camilo, et al., "Web Service for Document Management of University Degree Projects", *ICAIW 2022: Workshops at the 5th International Conference on Applied Informatics 2022*, 2022, Arequipa, pp. 214-215.
26. SERGIEVSKI Maxim & KIRPICHNIKOVA Ksenia, "Optimizing UML Class Diagrams", *ITM Web of Conferences 18, 03003 (2018)*, 2018.
27. SHABBIR Mohamed, NABI A. et al., "Suitability of Using SOAP Protocol to Secure Electronic Medical Record Databases Transmission", *International Journal of Pharmacology*, Vol. 6, 2010, p. 961.
28. SUNYAEV, "Web Services", in *Internet Computing*, Springer, Cham, 2020.

29. TOMAN Sarah Hussein, "Review of Web Service Technologies: REST over SOAP", *Journal of Al-Qadisiyah for Computer Science and Mathematics*, Vol. 12, no. 4, 2020, pp. 18-19.
30. YUHAN Chen, et al., "Evaluation Methodology of Interoperability for the Industrial Domain: Standardization vs. Mediation", *Processes*, Vol. 11, no. 4: 1274, 2023.