REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE

Ministère de l'Enseignement Supérieur

et de la Recherche Scientifique

Université "Yahia FARES" Médéa

Faculté des Sciences

Département de Mathématiques et Informatique

Projet de fin d'étude

Présenté pour l'obtention du diplôme

De Master en Informatique

Spécialité : Ingénierie des Systèmes et Technologie de Web

Thème

Elaboration d'une plateforme de e-gouvernance à base des services web

Proposé et dirigé par :

Mr. TOUBAL El Bahi

Préparé et présenté par :

M^elle^. *MADANI Messaouda*

M^elle^. *MOHAMMEDI Amina*

Promotion 2018



Remerciement

Nous remercions tout d'abord, Allah qui nous a donné la force et le courage afin de parvenir à élaborer ce modeste travail.

Pour tous les bienfaits et les faveurs qu'il nous a attribuées et sans les quels nous ne serons pas arrivés là où l'on est.

Nous tenons aussi à remercier : Mr. Toubal El Bahi, notre encadreur, pour ses conseils, ses orientations, sa disponibilité et son encouragement qui nous ont permis de réaliser ce travail.

Nous remercions tous les enseignants de l'université Yahia Fares.

Nous tenons à exprimer notre gratitude à toutes les personnes qui nous consacrés à leur temps et de leur énergie afin de faciliter le déroulement de notre projet de fin d'étude.



Dédicace

Je remercie ALLAH

Tout puissant qui m'a donné toute la volonté et la

Patience pour aboutir à la réalisation de ce travail.

Je dédie mon travail à :

Ma très chère mère.

Mon très cher père.

Mes frères et mes sœurs.

Toute ma famille.

Tous mes amis.

Tous ceux qui me sont chers, …





Messaouda

Dédicace

Je dédie ce modeste travail à :

Mes parents,

Mes frères et sœurs,

Ma famille

Tous mes amis,

Tous mes collègues.





Amina

Résumé

L'objectif des gouvernements est de développer ses performances en suivant tous les développements dans différents domaines, notamment dans le domaine des technologies de l'information et de la communication (TIC), cela a rendu importante la création de gouvernements électroniques. Nous proposons donc dans notre projet la conception et l'implémentation d'une plateforme de gouvernement électronique (e-gouvernement) basé sur les services web.

La plateforme permet aux institutions de publier et de rechercher des services Web, ainsi que d'offrir d'autres services via la composition de services Web, en utilisant le langage de composition des services Web bpel, et Nous avons utilisé la technologie de cloud computing pour stocker des informations afin de faciliter leur utilisation et leur récupération.

Mot clé : e-gouvernement, services Web, la composition, bpel, Cloud computing.

الملخص

تهدف الحكومات إلى تطوير أدائها و ذلك من خلال مواكبة كل التطورات الحاصلة في مختلف المجالات ،لاسيما في مجال تكنولوجيا المعلومات والإتصال (TIC) ، هذا ما جعلها تهتم بانشاء حكومات الكترونية, لهذا نقترح في مشروعنا تصميم و انشاء منصة للحكومة الالكترونية (e-gouvernement) تعتمد على خدمات الواب.

تسمح هذه المنصة للمؤسسات بالنشر و البحث عن خدمات الواب ، بالاضافة الى تقديم خدمات اخرى من خلال تركيب هذه خدمات، وذلك بالاستعانة بلغة تركيب خدمات الواب bpel. واستخدمنا تقنية الحوسبة السحابية لتخزين المعلومات لتسهيل استخدامها .

الكلمات المفتاحية : الحكومة الالكترونية، خدمات الواب ، تركيب ، bpel، الحوسبة السحابية.

Résumé

Abstract

The objective of governments is to develop its performance by following all the developments in different fields, particularly in the field of information and communication technologies (ICT). which has made the creation of electronic governments important. We therefore propose in our project the design and implementation of an electronic government (e-government) platform based on web services.

The platform allows institutions to publish and search for Web services, as well as to offer other services through the composition of Web services, using the bpel web services composition language, and we used technology Cloud computing to store information to facilitate their use and recovery.

Keyword: e-government, web services, composition, bpel, cloud computing.

Sommaire

Résumé ---- i
Sommaire ---- i
Liste des figures ---- v
Liste des tableaux ---- vii
Liste des acronymes ---- viii
Introduction générale ---- 1
Chapitre I : gouvernement électronique « E-gouvernement » ---- 3
I.1. Introduction ---- 3
I.2. Définition de gouvernement ---- 3
I. 3. Définition de l'e-gouvernance ---- 3
I.4. Définition de l'e-gouvernement ---- 3
I.5. Les objectifs d'E-gouvernement ---- 4
I.6. Les éléments stratégiques de l'E-gouvernement ---- 4
I.6.1. Stratégies ---- 4
I.6.2. Sous-traitance ---- 4
I.6.3. Financement ---- 5
I.6.4. Culture et politiques de l'Etat ---- 5
I.6.5. Leadership ---- 5
I.6.6. Gestion centralisée/décentralisée de la technologie ---- 5
I.6.7. Evaluation de la performance ---- 5
I.7. Les domaines de l'e-gouvernement ---- 6
I.7.1. L'E-administration ---- 6
I.7.2. L'E-société ---- 7
I.7.3. L'E-démocratie ---- 7
I.8. Modèles d'E-gouvernement ---- 7
I.8.1. G2C : Gouvernement au citoyen ---- 8
I.8.2. G2B : Gouvernement au Entreprises ---- 9
I.8.3. G2G : Gouvernement au Gouvernement ---- 9
I.9. La mise en place de l'e-gouvernement ---- 9
I.10. Les avantages de l'e-gouvernement ---- 9
I.11. Risques et défis de développement de l'e-gouvernement ---- 10
I.11.1. L'accessibilité ---- 10

I.11.2. La sécurité ---- 10
I.11.3. La protection des renseignements personnels ---- 11
I.12. Exemples ---- 11
I.12.1. E-Gouvernement en Corée de sud ---- 11
I.12.2. E-gouvernement au Maroc ---- 12
I.13. Conclusion ---- 13
Chapitre II : Services web ---- 15
II.1. Introduction ---- 15
II.2. Définition de l'architecture SOA ---- 15
II.3. Caractéristiques de l'architecture SOA ---- 16
II.4. Les composants de SOA ---- 16
II.5. Les services Web ---- 18
II.5.1. Définition ---- 18
II.5.2. Caractéristiques des services web ---- 19
II.5.3. Standards de services Web ---- 19
II.5.4. Description du modèle fonctionnel des services web ---- 25
II.6. Composition des Services Web ---- 26
II.6.1 Définitions ---- 26
II.6.2 Cycle de vie d'une composition de services ---- 27
II.6.3 Types de composition de services Web ---- 28
II.6.4 Langages de composition de services web ---- 30
II.7. Conclusion : ---- 31
Chapitre III : Cloud Computing « l'informatique en nuage » ---- 33
III.1. Introduction : ---- 33
III.2. Historique ---- 33
III.3. Définition ---- 34
III.4. Les caractéristiques de Cloud ---- 35
III.5. L'architecture Cloud ---- 35
III.5.1. SaaS : ---- 36
III.5.2. PaaS : ---- 37
III.5.3. IaaS : ---- 38
III.6. Modèles de déploiement des Cloud ---- 39
III.6.1. Cloud privé ---- 39
III.6.2. Cloud public ---- 39

III.6.3. Cloud communautaire ---- 40
III.6.4. Cloud hybride ---- 40
III.7. Les enjeux du cloud computing ---- 41
III.8. Cloud Computing : avantages et inconvénients ---- 42
III.8.1. Avantages ---- 42
II.8.2. Inconvénients : ---- 42
III.9. Les principaux acteurs du Cloud : ---- 43
III.10. Conclusion ---- 44
Chapitre IV : Conception ---- 46
IV.1 Introduction ---- 46
IV.2 Approche proposée ---- 46
IV.3 L'architecture conceptuelle de la plateforme ---- 46
IV.4 Description des modules ---- 49
IV.4.1. Le module de gestion des utilisateurs ---- 49
IV.4.2 Le module de publication ---- 51
IV.4.3 Le module de découverte ---- 52
IV.4.4 Le module de composition ---- 52
IV.5 Les entités de la plateforme ---- 54
IV.6 Les types d'utilisateur ---- 54
IV.6.1. Administrateur de la plateforme ---- 54
IV.6.2 Administrateur d'institution ---- 55
IV.6.3 Fournisseur ---- 55
IV.6.4 Demandeur ---- 55
IV.7 Conclusion ---- 55
Chapitre V : Implémentation et étude de cas ---- 56
---- 56
V.1 Introduction ---- 57
V.2 Les outils et les langages utilisés en développement ---- 57
V.2.1 Power designer 16.5 ---- 57
V.2.2 langage JAVA ---- 57
V.2.3 NetBeans ---- 57
V.2.4 Java Server Face ---- 58
V.2.5 PrimeFaces ---- 58
V.2.6 JDOM ---- 58

V.2.7 Apache ODE ---- 59
V.2.8 Glass Fish Serer ---- 59
V.2.9 WampServer ---- 59
V.3 Schème globale de la plateforme ---- 60
V.4 La mise en œuvre de la plateforme ---- 61
V.4.1 Authentification ---- 61
V.4.2 Fenêtre d'accueil ---- 62
V.4.3 Fenêtre institution ---- 62
V.4.4 Fenêtre utilisateurs ---- 63
V.4.5 Fenêtre recherche ---- 63
V.4.6 Fenêtre publication ---- 64
V.4.7 Fenêtre composition ---- 64
V.5 Etude de cas ---- 65
V.5.1 Scenario de la composition ---- 65
V.5.2 Représentation de scénario à l'aide de notre outil ---- 66
V.6 Conclusion ---- 70
Conclusion générale ---- 71
Bibliographique ---- **Error! Bookmark not defined.**
ANNEXE ---- 76
Les diagrammes des bases de données ---- 76
Fonctionnement notre outil ---- 80

Liste des figures

Figure I.1 : Domaine du gouvernement en ligne---------- 7

Figure I.2 : Les interactions entre les principaux groupes dans l'e-gouvernance---------- 9

Figure II.1 : Principaux acteurs de SOA. ---------- 18

Figure II.2 : Schéma de fonctionnement du système unidirectionnel SOAP. ---------- 21

Figure II.3 : Structure de message SOAP. ---------- 22

Figure II.4 : L'en-tête du protocole de transport http. ---------- 22

Figure II.5: Entête de Message SOAP ---------- 23

Figure II.6 : Message SOAP requête. ---------- 23

Figure II.7 : La structure d'un document WSDL. ---------- 24

Figure II.8: Le contenu de l'annuaire UDDI. ---------- 26

Figure II.9: Modèle fonctionnel des services web. ---------- 26

Figure II.10 : Les Approches de la composition des services Web. ----------29

Figure II.11 : Vue générale de l'orchestration. ---------- 30

Figure II.12 : vue générale de la chorégraphie ---------- 30

Figure III.1 : NIST – Modèle visuel Cloud computing. ----------35

Figure III.2 : pile de services Cloud. ----------36

Figure III.3 : le modèle de service SaaS. ---------- 37

Figure III.4 : le modèle de service PaaS. ---------- 39

Figure III.5 : le modèle de service IaaS. ---------- 40

Figure III.6 : Modèles de déploiement Cloud Computing. ---------- 41

Figure IV.1 : L'architecture global de notre plateforme. ---------- 48

Liste des figures

Figure IV.2 : Fonctionnement de la plateforme ---------- 49

Figure IV.3 : module de gestion d'utilisateur. ---------- 50

Figure IV.4 : module de Publication. ---------- 51

Figure IV.5 : module de Composition ---------- 53

Figure IV.6 : Diagramme de classe de la plateforme. ---------- 54

Figure V.1 : Schème globale de la plateforme ---------- 60

Figure V.2 : page d'authentification. ---------- 61

Figure V.3 : la page d'accueil. ---------- 62

Figure V.4 : la page Institutions. ----------62

Figure V.5 : la page Utilisateurs. ---------- 63

Figure V.6 : la page Recherche. ---------- 63

Figure V.7 : la page Publication. ---------- 64

Figure V.8 : la page composition. ---------- 64

Figure V.9 : Schéma explicatif le scenario de cas d'étude ---------- 66

Figure V.10 : modélisation du scénario de recrutement. ---------- 66

Figure V.11 : fichier bpel de service recrutement. ---------- 67

Figure V.12: fichier wsdl de service recrutement. ---------- 67

Figure V.13: fichier deploy.xml de service recrutement. ---------- 68

Figure V.14 : fichier resultat.xml de service recrutement.---------- 69

Liste des tableaux

Tableau I.1 : les modèles d'E-Gouvernance ---------- 9

Tableau III.1: la différence entre les principaux d'acteurs de Cloud---------- 45

Liste des acronymes

Mot	Description
BPEL	Business Process Execution language
G2B	Government to Business
G2C	Government to citizen
G2G	Government to Government
HTTP	HyperText Transfer Protocol
IaaS	Infrastructure as a service
IBM	International Business Machines Corporation
Java EE	Java Enterprise Edition
JDOM	Java Document Object Model
JSF	Java server Fac
LISP	list Processing
MIT	Massachusetts Institute of Technology
NIST	National Institute of Standards and Technology
PaaS	Platform as a Service
SaaS	Software as a Service
SOA	Service oriented architectures
SOAP	Simple Object Access Protocol
TI	Technologies d'information
TIC	Technologies de l'information et de la communication

Liste des acronymes

UDDI	Universal Description Discovery and Integration
W3C	Consortium du World Wide Web
WSCDL	Web Services Choreography Description Language
WSCI	Web Services Choreography Interface
WSDL	Web Service Description Language
WSFL	Web Service Flow Language
XML	Extensible Markup Language

Introduction générale

Le gouvernement travaille à améliorer ses relations avec les citoyens, en améliorant les services qu'il fournit. Mais, actuellement fait face à des changements majeurs, surtout en matière de développement technologique et de la tendance des gens à l'utiliser. Donc, il s'est retrouvé obligée de suivre ces changements à travers l'utilisation de TIC. Il est passé d'un gouvernement qui s'appuie sur des moyens traditionnels à un gouvernement électronique(e-gouvernement).

Le gouvernement électronique s'appuie sur plusieurs moyens technologiques, afin de rendre le gouvernement plus responsable et transparent. Parmi les technologies dont le gouvernement électronique dépend pour améliorer ses services et qui rend le gouvernement plus crédible, Nous trouvons par exemple les services web, les clouds computing, etc.

Le gouvernement électronique utilise la technologie des services Web, pour diffuser et obtenir des informations sur les citoyens et fournir de bons services aux clients, les services web sont conçus pour être composés afin d'accomplir des buts complexes non réalisables par un seul service web. Les services web sont des briques de base logicielles s'affranchissant de toute contrainte de compatibilité logicielle ou matérielles, Ils le font en reprenant certains principes du Web (transport sur HTTP, formatage en XML) [19] [20].

L'e-gouvernement utilise un grand volume des données personnelles pour cela, La notion de cloud offre de nombreux avantages tels qu'un déploiement rapide, des économies de coûts, une évolutivité, une mise en service rapide, une flexibilité rapide, un accès réseau partout, une protection réseau supérieure, des solutions de reprise après sinistre à moindre coût et de stockage de données. Toutes ses caractéristiques ont rendu le gouvernement électronique dépendant d'eux pour stocker des informations.

L'objectif de notre projet est de créer une plateforme qui permet d'offrir des services aux clients. Les clients sont des institutions (des entreprises), le client peut publier, invoquer ou consulter les services web, et offrir d'autres services à travers la composition des services publiés dans la plateforme.

Nous avons divisé ce mémoire en cinq chapitres :

Chapitre I : dans ce chapitre nous allons illustrer la notion e-gouvernement, ainsi les objectifs principaux de cette dernière. Ensuite, nous allons présenter la relation d'e-gouvernement avec les citoyens et les entreprises.

Chapitre II : dans le deuxième chapitre, nous allons présenter la définition des services web , l'architecture de ces services, les caractéristiques, les standards, et Enfin, un aperçu sur la composition des services.

Chapitre III : dans le troisième chapitre, nous allons présenter les clouds computing, leurs caractéristiques, les modèle de services, les modèle de déploiements et les acteurs principales de cette notion.

Chapitre IV : dans le quatrième chapitre, nous allons présenter la conception de l'approche proposée de la plateforme et l'architecteur de cette dernière.

Chapitre V : dans le cinquième chapitre, nous allons représenter le schème global de notre plateforme, la mise en œuvre de cette dernière et l'étude de cas.

Chapitre I : gouvernement électronique « E-gouvernement »

I.1. Introduction

Actuellement, l'utilisation des technologies de l'information et de la communication est obligatoire pour améliorer le gouvernement. Par conséquent, un certain nombre de concepts reliés aux technologies d'information (TI) ont été de plus en plus utilisés, en particulier les notions de gouvernement électronique, appelé souvent e-gouvernement, à travers ce chapitre, nous allons présenter quelques notions sur l'e-gouvernement.

I.2. Définition de gouvernement

De manière générale, la gouvernance n'est pas un concept nouveau car elle fait partie du langage de la philosophie et des sciences politiques depuis Platon et Aristote [1].

Selon l'Office de la langue française la gouvernance correspond à la « manière d'orienter, de guider, de coordonner les activités d'un pays, d'une région, d'un groupe social ou d'une organisation privée ou publique » [2].

Selon James Rosenau définit la gouvernance comme « un ensemble de mécanismes de régulation dans une sphère d'activité, qui fonctionnent même s'ils n'émanent pas d'une autorité officielle » [3].

I. 3. Définition de l'e-gouvernance

La gouvernance électronique est la manière de diriger et d'administrer propre à un gouvernement en ligne, permettant à celui-ci, grâce à l'utilisation des TIC (technologies de l'information et de la communication), d'améliorer les services publics, de développer la responsabilisation et la transparence dans les relations entre les citoyens et l'administration, tout en favorisant le développement d'une société de l'information et le processus de démocratisation qui l'accompagne [4].

I.4. Définition de l'e-gouvernement

L'e-gouvernement est un concept qui est apparu dans l'administration publique vers la fin des années 1980[5]. Il existe plusieurs définitions sur l'e-gouvernement :

L'e-gouvernement est l'usage des TIC pour transformer le gouvernement en le rendant plus accessible aux citoyens, plus efficace et plus responsable. Cette approche très résumée, présente une conception particulière de l'impact des TIC sur le nouveau rôle de l'État (accessibilité, efficacité et responsabilité) [14].

L'e-gouvernement peut être défini comme l'utilisation des technologies de l'information et des communications (TIC) pour améliorer les activités des organisations du secteur public et de leurs agents [6].

I.5. Les objectifs d'E-gouvernement

L'objectif global de l'e-gouvernement est de permettre au gouvernement d'être plus orienté vers les résultats, plus efficace et plus sensible aux citoyens [7]. Mais, il y a d'autres objectives. On peut spécifier comme la suite :

- Transformer le gouvernement comme organisation plus responsable, transparente et efficace.
- Améliorer la collaboration entre les institutions gouvernementales et les clients.
- Fournir des meilleurs services, moins cout et plus fiables.
- Toujours fournir les informations en ligne sur le gouvernement.
- Donner aux utilisateurs le choix de décider quand, d'où et comment, ils souhaitent interagir avec le gouvernement.
- Être proactifs dans le processus de prise de décision du gouvernement.
- Des services simples, pour exécuter des transactions en ligne.

I.6. Les éléments stratégiques de l'E-gouvernement

Pour exécuter la stratégie de l'e-gouvernement. Il faut trouver les éléments suivants [7] :

I.6.1. Stratégies

Les stratégies sont essentielles pour la formulation de l'e-gouvernement parce qu'elles fournissent les objectifs pour les agences d'Etat et les gouvernements.

I.6.2. Sous-traitance

La procédure de sous-traitance qui consiste à contracter avec une entreprise extérieure pour la mise au point de projets est souvent controversée. La décision de sous-traiter pourrait dépendre de la complexité du nouveau programme.

I.6.3. Financement

Le financement est une question importante pour l'e-gouvernement parce que les projets IT (technologie d'information) sont coûteux et leur succès incertain. Les corps législatifs doivent choisir entre de multiples programmes aux cors de la procédure budgétaire et dans de nombreux cas, l'e-gouvernement est en compétition avec d'autres besoins pour les financements.

I.6.4. Culture et politiques de l'Etat

Les politiques et la culture d'un Etat peuvent soit freiner soit encourager le développement de l'e-gouvernement. Les agences publiques sont souvent opposées aux changements de technologies parce que celles-ci peuvent modifier les fonctions de l'employeur et des agences. Toutefois, les améliorations apportées par les nouvelles technologies ne provoquent pas, en règle générale, la suppression des agences.

I.6.5. Leadership

Un leadership solide permet d'exécuter les programmes d'e-gouvernement et piloter les améliorations apportées par les nouvelles technologies en encourageant et en promouvant les nouveaux projets.

I.6.6. Gestion centralisée/décentralisée de la technologie

Le degré de centralisation ou de décentralisation est un élément clé de la gestion de l'e-gouvernement parce qu'il détermine le niveau d'interaction des agences administratives participant dans les projets de technologies d'information. La centralisation du portail de la toile constitue une tendance commune dans de nombreux États et est souvent séparée des organes d'organisation de prises décisions de l'agence.

I.6.7. Evaluation de la performance

L'évaluation de la performance de l'E-gouvernement est très importante pour estimer le succès des programmes, identifier les défis et prendre en charge les difficultés liées à leur formulation et mise en œuvre.

I.7. Les domaines de l'e-gouvernement

De manière générale, l'e-gouvernement se divise en 3 grands blocs, illustré dans la figure suivante :

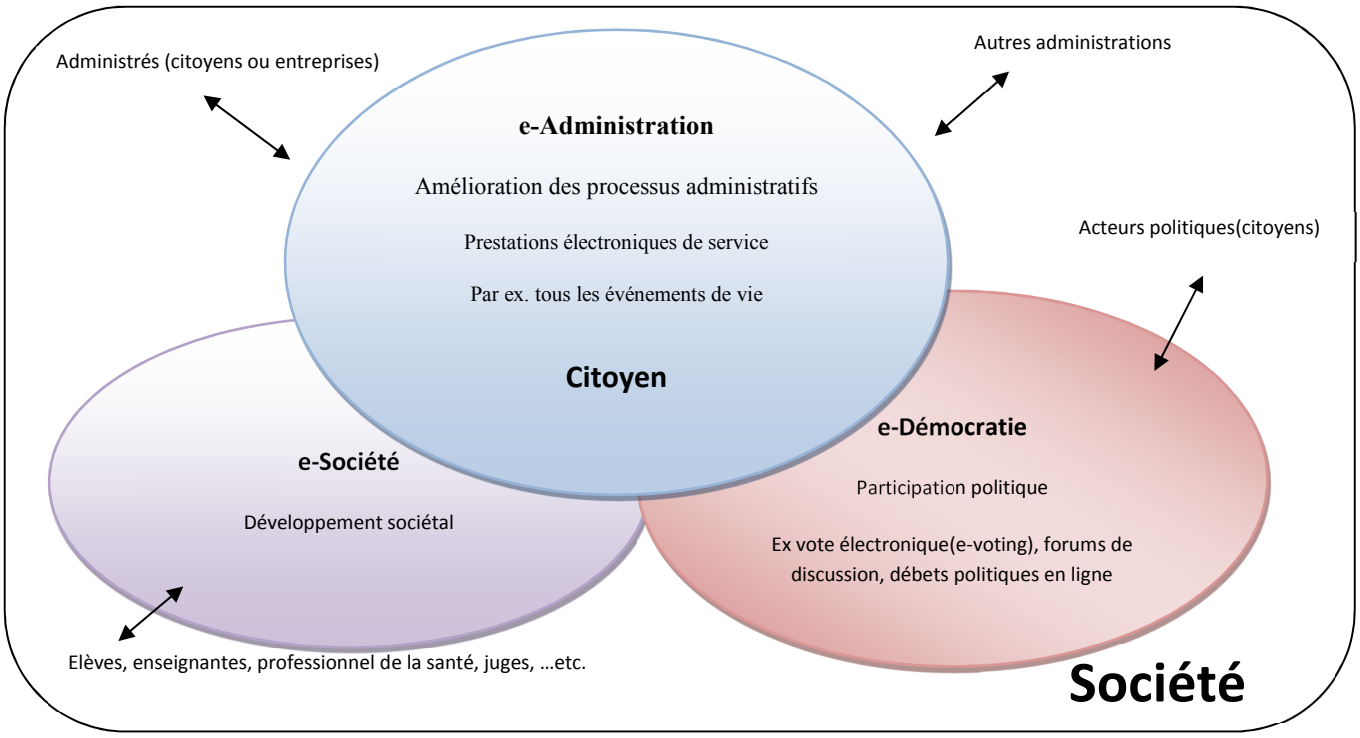


Figure I.1 : Domaine du gouvernement en ligne

I.7.1. L'E-administration

L'objectif global de l'E-administration est l'amélioration de la performance, de la cohérence, de la transparence et de la compétence interne de l'administration publique par l'informatisation de tous les services administratifs aussi bien au niveau central qu'au niveau décentralisé. L'e-administration vise particulièrement à [1] :

- Promouvoir l'accès à l'information des citoyens sur leurs droits et devoirs dans les domaines variés de la santé, de l'éducation, de l'assainissement et d'autres prestations de base des services publics.
- Assurer la transparence des informations, des décisions et des actions de l'Etat et des collectivités territoriales.

I.7.2. L'E-société

De telles initiatives portent, en particulier, sur les rapports entre les agences publiques et les autres institutions – d'autres agences publiques, les sociétés du secteur privé, les organisations à but non lucratif et communautaires. Comme pour les connexions entre citoyens, ces initiatives pourraient bel et bien incorporer les améliorations des procédures. Toutefois, elles pourraient inclure des attributions plus larges comme [7] :

- *Mieux travailler avec les entreprises* : amélioration de l'interaction entre le gouvernement et les entreprises. Cela pourrait comprendre la digitalisation de la réglementation sur l'approvisionnement et les services en direction des entreprises pour en améliorer la qualité, la commodité et le coût.
- *Développer les communautés :* renforcer les capacités sociales et économiques ainsi que le capital des communautés locales.
- *Renforcer les partenariats :* créer des groupements d'organisations pour atteindre les objectifs économiques et sociaux. Le secteur est presque toujours l'un des partenaires, même si occasionnellement, il n'agit qu'en facilitateur pour les autres.

I.7.3. L'E-démocratie

L'e-démocratie vise le développement et l'amélioration des relations avec le citoyen en tant qu'acteur politique et détenteur de la légitimité ainsi que les relations intergouvernementales, distingue 3 versions de la démocratie électronique : une version populiste, une version libérale, une version républicaine.

Son analyse de la littérature sur la démocratie électronique montre que celle-ci s'organise autour de 3 questions principales : l'information des citoyens. Le débat et la discussion et la délibération et la prise de décision publique [2].

I.8. Modèles d'E-gouvernement

Les trois principaux groupes cibles qui peuvent être distingués dans les concepts d'e-gouvernance sont gouvernement, les citoyens et les entreprises / groupes d'intérêt. Les objectifs stratégiques externes se concentrent sur les citoyens et les entreprises et les groupes d'intérêt, les objectifs internes se concentrent sur le gouvernement lui-même [8].

Les trois modèles présentés dans la table suivante :

Modèle	Abréviation
Gouvernement au citoyen	G2C
Gouvernement au Entreprises	G2B
Gouvernement au Gouvernement	G2G

Tableau I.1 : les modèles d'E-Gouvernance.

La relation entre ces trois groupes (gouvernement, les citoyens et les entreprises) est présentée dans la figure suivante :

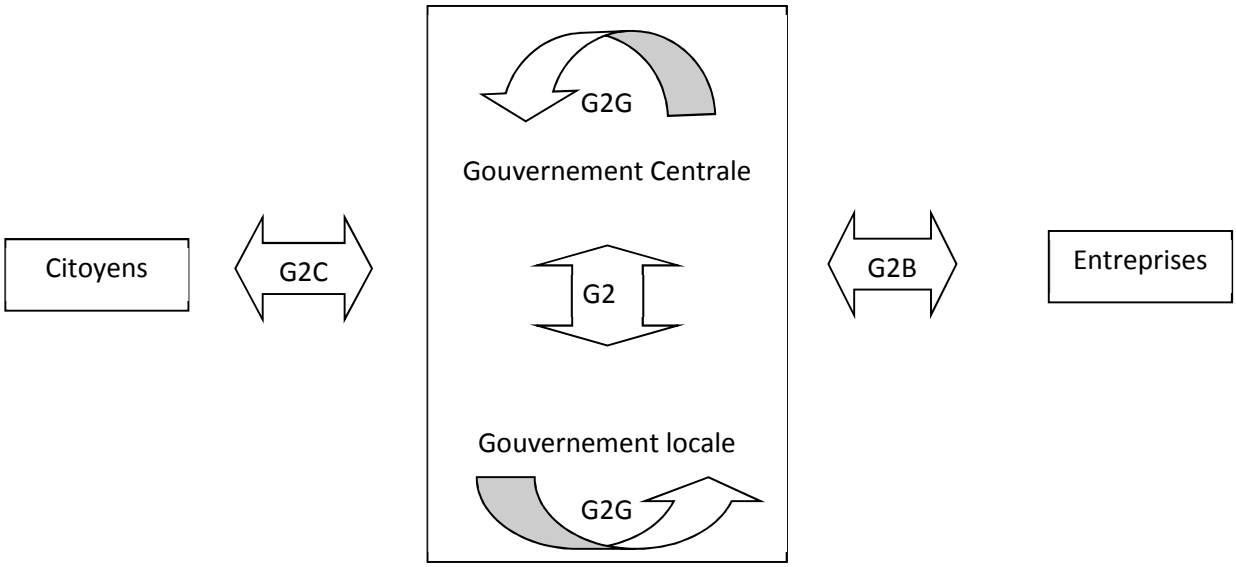


Figure I.2 : Les interactions entre les principaux groupes dans l'e-gouvernement.

I.8.1. G2C : Gouvernement au citoyen

Englobe les initiatives visant à faciliter l'interaction des gens avec le gouvernement en tant que consommateurs de services publics et en tant que citoyens. Cela comprend les interactions liées à la prestation de services publics, ainsi que la participation à la consultation et au processus de prise de décision [9].

I.8.2. G2B : Gouvernement au Entreprises

Englobe les transactions spécifiques aux affaires (par exemple les paiements effectués en ce qui concerne la vente et l'achat de biens et de services) ainsi que la fourniture en ligne de services axés sur les entreprises [9].

I.8.3. G2G : Gouvernement au Gouvernement

Implique le partage de données et la conduite des échanges électroniques entre acteurs gouvernementaux. Cela implique des échanges à la fois intra- et interinstitutionnels, ainsi que des échanges entre les niveaux nationaux, provincial et local [9].

I.9. La mise en place de l'e-gouvernement

La gouvernance électronique résulte d'une démarche structurée en cinq étapes [10] :

a. Présence émergente : c'est la présence sur le réseau par le biais d'un site officiel, d'un portail d'accès national, ou d'une page d'accueil officielle.

b. Présence accentuée : les services en ligne sont renforcés et incluent des bases de données.

c. Présence interactive : c'est la possibilité de télécharger des documents en ligne, d'accéder à des liens sécurisés, des services audio et vidéo... etc.

d. Présence transactionnelle : les utilisateurs peuvent effectuer des transactions en ligne.

e. Présence en réseau : établissement d'une relation entre le gouvernement et les utilisateurs de services électroniques se fondant sur les informations, le savoir et les services fournis par les institutions publiques. L'accent est mis sur le retour d'information.

I.10. Les avantages de l'e-gouvernement

Les principaux avantages d'un gouvernement électronique sont [14] :

- Le gouvernement produit plus de services au même coût ou à un moindre prix.
- Les opérations du gouvernement deviennent plus rapides – à la différence du traitement manuel qui repose sur le facteur humain qui exige un temps d'exécution plus long. Les nouvelles technologies de l'information permettent la production de résultats à un coût moindre en moins de temps.
- Le gouvernement produit des services de qualité supérieure. En effet, le gouvernement électronique assure une communication de qualité et une transparence de l'information.

- Les gouvernements deviennent inventifs et plus créatifs. En effet, le gouvernement électronique force les organismes gouvernementaux à rester à jour dans les technologies et trouver de façon continue des solutions novatrices pour développer les services publics. En outre, les lois et les réglementations doivent continuellement être mises à jour, simplifiées et débarrassées de tout superflu.
- Le gouvernement électronique peut aider à bâtir la confiance par l'engagement des citoyens, et ce de plus en plus dans les politiques publiques.

I.11. Risques et défis de développement de l'e-gouvernement

Il existe trios risques principaux de développement de l'e-gouvernement sont [5] :

I.11.1. L'accessibilité

La « fracture digitale » est une expression récente qui exprime l'accès inégal de la population aux médias électroniques dont l'Internet. Puisque le développement de l'e-gouvernement implique l'utilisation croissante de l'Internet, l'accessibilité de tous les citoyens doit être une préoccupation gouvernementale.

I.11.2. La sécurité

La sécurité technique des sites WEB constitue pour l'ensemble des navigateurs un facteur très important de confiance. Cette sécurité favorise l'utilisation des services électroniques, spécialement les transactions et les paiements en ligne.

Les organisations doivent mettre en place des mécanismes de contrôle efficaces pour s'assurer de la sécurité des systèmes d'information et des infrastructures des TIC. Depuis quelques années, ces mécanismes de contrôle se sont amplement développés. Voici des exemples de mécanismes de contrôle :

- Contrôle des infrastructures matérielles pour réduire les risques d'accidents, de sinistres naturels, de sabotage, de vandalisme, d'utilisation non autorisée, de destruction et de vols de données et de ressources matérielles.
- Contrôle des systèmes d'informations pour assurer la précision, la sécurité et l'exactitude des entrées, du traitement, des sorties et du stockage.

I.11.3. La protection des renseignements personnels

La protection des renseignements personnels constitue un élément majeur pour le développement du gouvernement en ligne.

I.12. Exemples

I.12.1. E-Gouvernement en Corée de sud

D'après les différents classements de « United Nation E-gouvernement », la Corée du Sud occupe constamment le premier rang mondial en matière de développement de l'e-gouvernement. Ce succès est le fruit de plusieurs projets et initiatives soigneusement conçus, planifiés et conduits durant les phases de réalisation [11].

Les deux objectifs de ce dernier, considéré comme le socle du gouvernement électronique en Corée du Sud, furent [11] :

- La construction de réseaux à large bande à l'échelle nationale.
- La promotion de l'informatisation.

Le site internet principal du gouvernement électronique a été développé, de sorte à regrouper la quasi-totalité des services à l'endroit des citoyens physiques et moraux, et ce au plan national et local [11]. Alors on a représenté ce site d'internet, par exemple :

I.12.1.1. G4C (Portail du Gouvernement)

A) Définition :

G4C est une e-fenêtre intégrée pour les services de l'administration civile où les citoyens coréens peuvent obtenir des orientations, postuler, lire et imprimer des informations administratives via l'internet [12].

B) Stratégie :

Coopération& Consensus : les agences concernées font des consultations pour les besoins du G4C. Après quoi, l'établissement de 'La Fenêtre Unique' fut présenté et un consensus trouvé.

G4C convivial :5,000 sortes de tâches administratives légales ont été catégorisées sur la base des informations accumulées à travers le réseau administratif.

G4C fiable : un grand effort a été fait pour assurer la sécurité du système et prévenir la falsification de document pour un G4C fiable. En outre, une fondation institutionnelle et légale a été créée pour soutenir le processus légal et institutionnel [12].

I.12.2. E-gouvernement au Maroc

Le Maroc a œuvré pour l'intégration de la gouvernance électronique à travers plusieurs projets et initiatives, preuve de la grande volonté politique des dirigeants du Royaume [11].

Selon le document publié par les nations unies « l'État de l'information pour les activités de développement en Afrique du nord », La volonté du Royaume du Maroc d'entrer dans l'ère de l'information s'exprime à travers la stratégie nationale e-Maroc, qui poursuit selon la commission économique pour l'Afrique attachée aux nations unies, Cinq objectifs [2] :

1. La généralisation des TIC.
2. Le déploiement accélérait des infrastructures.
3. L'accélération de la libéralisation et de la concurrence.
4. La redéfinition du rôle de l'État.
5. La mobilisation et la mise en place des moyens de la stratégie.

En consultant les anciens plans stratégiques, notamment le plan quinquennal TIC 1999-2003, qui était à la base de la stratégie e-Maroc 2010, on retrouve que celui-ci s'articulait autour des axes de développement suivants [2] :

- Élaborer une législation capable de répondre aux problématiques actuelles (protection des noms de domaines commerciaux, protection des données nominatives, besoins du commerce électronique, etc.).
- Amorcer un tournant au niveau de l'éducation (alphabétisation, intégration des outils informatiques et d'Internet dans les écoles, etc.).
- Réduire, à défaut d'éviter, la fracture numérique par des politiques qui favorisent le déploiement des TIC dans les régions et permettent leur utilisation même par les couches les plus démunies de la population.
- Financer la création d'entreprises de la nouvelle économie du savoir (capital risque, prêts au démarrage des entreprises à fort contenu technologique, etc.).

I.13. Conclusion

Dans ce chapitre, nous avons commencé par les définitions des termes gouvernement, l'e-gouvernance et l'e-gouvernement, ensuite nous avons déterminé les objectifs principaux et les éléments de stratégiques (la stratégie, financement, leadership, etc.) de l'e-gouvernement. Aussi, nous avons vu les domaines de l'e-gouvernement (e-administration, e-société et e-démocratie) et la relation entre les trois groupes (gouvernement, les citoyens et les entreprises) avec les conditions nécessaires pour assurer e-gouvernance.

Nous avons présenté aussi les avantages de-gouvernement et certains risques et défis de cette technologie. Enfin nous avons représenté quelques exemples sur l'e-gouvernement.

Chapitre II : Services web

II.1. Introduction

Aujourd'hui, on peut dire que le web qu'a était à l'origine un simple moyen de partage d'information. Et à cause de l'augmentation de volume de l'information, le web évolué rapidement et est devenu très complexe. Donc il faut développer des nouvelles technologies et nouveaux modèles d'interaction pour retourner l'information qui nous souhaitons. La naissance de concept de l'architecture orientée service a donné naissance à une autre technologie qui s'appelle les services web. Elles fournissent un lien entre les applications et les systèmes hétérogènes dans des environnements distribués. Elles permettent de réponde aux besoins des utilisateurs (application ou client). Mais lorsque les besoins sont complexes, il difficile de définir un service web précise. Dans ce cas, on peut composer plusieurs services pour jouer un rôle bien précis.

Dans ce chapitre, nous allons introduire un cadre théorique concernent l'architecture orientée services, ensuite nous présentons les services web, et finalement nous présentons la notion de la composition des services web.

II.2. Définition de l'architecture SOA

L'architecture SOA est un paradigme destiné à résoudre plusieurs problèmes, tel que d'hétérogénéité et interopérabilité des logiciels qui constituent le système d'information. Elle possède plusieurs définitions, selon leurs caractéristiques techniques qui renvoient toutes vers un seul sens :

Dans [15], l'auteur propose la définition suivante : L'architecture orientée service est un paradigme qui permet l'intégration d'application et ressources de manière flexible en :

- Représentant chaque application ou ressource sous la forme d'un service exposant une interface standardisée.
- Permettant à un service d'échanger des informations structurées (messages, document, 'objets métier').
- Coordonnant et organisant les services afin d'assurer qu'ils puissent être invoqués et utilisés de manière efficace.

D'un autre part [16] propose la définition suivante : « L'architecture orientée service est un ensemble de méthodes techniques, métiers, procédurales, organisationnelles et gouvernementales pour réduire ou éliminer les frustrations avec les technologies d'information, et pour mesurer quantitativement la valeur métier des technologies d'information, pendant la création d'un environnement métier agile pour un intérêt concurrentiel ».

Et le [16] propose la définition Technique suivante :« L'architecture SOA est un paradigme permettant d'organiser et d'utiliser des savoirs faires distribués pouvant être de domaines variés. Cela fournit un moyen uniforme d'offrir, de découvrir, d'interagir et d'utiliser des savoirs faires pour produire le résultat désiré avec des Préconditions et des buts mesurables ».

II.3. Caractéristiques de l'architecture SOA

L'architecture orienté service est caractérisée par [17] :

- Un couplage faible entre les services : implique qu'un service n'appelle pas directement un autre service.
- La réutilisation de service web.
- L'indépendance par rapport aux aspects technologiques : c-à-d. Les services avec cette architecture sont indépendants de ses plates-formes.
- La découverte des services disponibles dans les annuaires.
- La mise à l'échelle est rendue possible grâce à la découverte et à l'invocation des nouveaux services lors de l'exécution.

II.4. Les composants de SOA

L'architecture orientée service est basée sur trois acteurs principaux (Figure II.1) définis comme suit :

a. Le Fournisseur du service (Service fournisseur) : correspond au propriétaire du service web. Ses fonctions consistent à créer des services web en application et il le rend disponible pour tout le monde sur internet.

b. L'Annuaire (Service annuaire) : correspond à un registre de descriptions des services web disponibles. Ainsi que des facilites de recherche des services web a l'intention des clients.

c. Le Client (Service consommateur) : correspond au demandeur ou un consommateur qui fait la demande d'un service web.

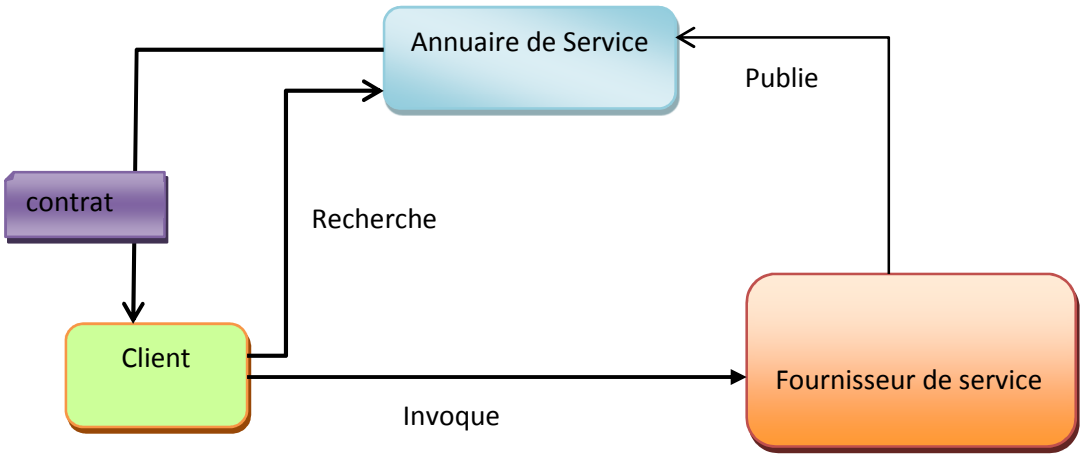


Figure II.1 : Principaux acteurs de SOA.

Les interactions de base entre ces trois composants incluent les opérations de publication, de recherche et de liens (Bind) d'opérations.

Pour garantir l'interopérabilité de ces trois opérations, des propositions de standards ont été élaborées pour chaque type d'interaction. Nous citons, notamment les standards émergents suivants :

- SOAP définit un protocole de transmission de messages basé sur XML.
- WSDL introduit une grammaire commune pour la description des services.
- UDDI fournit l'infrastructure de base pour la publication et la découverte des services [18].

II.5. Les services Web

Les services Web sont faits pour permettre la réalisation rapide et efficace de systèmes d'information distribués sur des réseaux en intégrant des applications existantes et nouvelles. Une architecture à base de services Web vise à permettre aux applications de communiquer facilement via Internet quelle que soit la plate-forme sous-jacente [19].

II.5.1. Définition

Les services web ont été proposés initialement par IBM et Microsoft, puis en partie standardisés par le W3C (le consortium du World Wide Web). Plusieurs définitions des services web ont été proposées dans la littérature :

Selon IBM :

« Les services web sont la nouvelle vague des applications web. Ce sont des applications modulaires, auto-contenues et auto-descriptives qui peuvent être publiées, localisées et invoquées depuis le web. Les services web effectuent des actions allant de simples requêtes à des processus métiers complexes. Une fois qu'un service web est déployé, d'autres applications (y compris des services web) peuvent le découvrir et l'invoquer » [19].

Selon W3C :

Le consortium W3C [1] définit un service web comme étant une application ou un composant logiciel qui vérifie les propriétés suivantes [20] :

- Il est identifié par un URI.
- Ses interfaces et ses liens (binding) peuvent être décrits en XML.
- Sa définition peut être découverte par d'autres services web.
- Il peut interagir directement avec d'autres services web à travers le langage XML et en utilisant des protocoles internet.

[1] <Http://www.w3.org/2002/ws/>

II.5.2. Caractéristiques des services web

Dans [19], plusieurs acteurs définissent les services web par des caractéristiques technologiques distinctives, les plus importantes sont :

- Un service web est une application logicielle qui est reconnue par un URI : le service web est caractérisé par un seul objet et une seule fonctionnalité, c'est un prolongement de la programmation orientée objet et à partir de cela, on peut faire la construction d'une application logicielle très large comportant plusieurs fonctionnalités, afin de sélectionner les fonctionnalités qui sont recherchées par les URI spécifiques.
- Capacité des interfaces et liaisons (bindings) d'être publiées, localisées et invoquées via le langage XML
- Capacité d'interagir avec les composantes des logiciels via des éléments XML avec l'utilisation des protocoles Internet standard.
- Composante logicielle légèrement couplée à interaction dynamique : un service web avec un programme qui l'invoque et appelé le consommateur de service web, et qui sont indépendants l'un de l'autre.

II.5.3. Standards de services Web

II.5.3.1 SOAP (Simple Object Access Protocol)

SOAP est un protocole d'échange de messages indépendants des plateformes, c'est un produit de Microsoft et IBM. La première version de SOAP a été acceptée par le W3C (Word Wide Web Consortium) en 2000 [21].

C'est le protocole de messagerie qui est devenu le standard utilisé pour les services web. Il définit la façon de faire des invocations de services et d'échanger des données structurées d'une manière standardisée en utilisant XML. Il se compose de trois parties : une enveloppe qui définit un Framework pour décrire le contenu d'un message, un ensemble de règles de codage, et une convention pour représenter les appels et les réponses [15].

SOAP se base généralement sur les deux standards HTTP et XML [22] :

- HTTP : est le protocole de transport des messages SOAP. Il constitue un bon moyen de transport pour sa réputation sur le web.
- XML pour structurer les requêtes et les réponses, indiquer les paramètres des méthodes, les valeurs de retour, et les éventuelles erreurs de traitements.

a. Modèle d'échange de messages en SOAP

Les messages SOAP sont des transmissions fondamentalement à sens unique (unidirectionnel) d'un expéditeur à un récepteur (figure II.2). Lorsqu'une transmission d'un message commence (E.g. invocation d'un service web), un message SOAP (ou document XML) est générée. Ce message est envoyé à partir d'une entité appelée le SOAP sender, localisé dans un SOAP nœud, celui-ci permet d'appeler une ou plusieurs fonctions du service [23].

Le SOAP utilise trois types de message qui sont :

- Appel (Call) obligatoire.
- Réponse (Response) facultatif.
- Erreur (fault) facultatif.

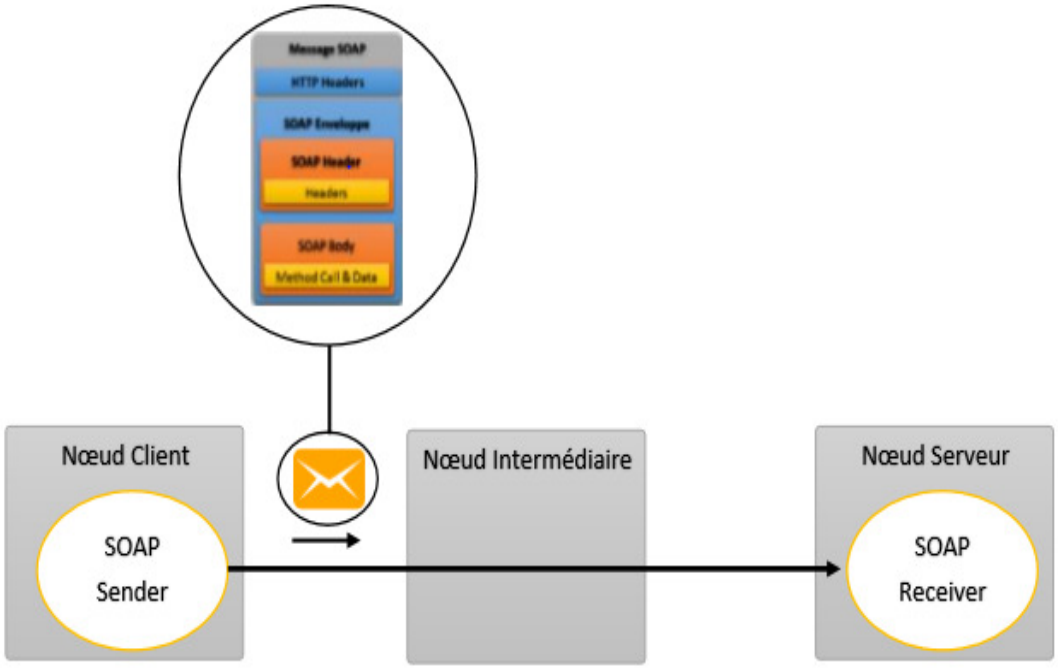


Figure II.2 : Schéma de fonctionnement du système unidirectionnel SOAP.

b. Structure d'un message SOAP : Un message SOAP est structuré de parties comme la suit (figure II.3) :



Figure II.3 : Structure de message SOAP.

- **L'en-tête du protocole de transport (http Headers)**: qui dépend de protocole de transport utilisé, par exemple si le protocole HTTP est utilisé, l'en-tête contient [17] :
  - La version de protocole HTTP utilisé.
  - La date de génération de message SOAP.
  - Le type d'encodage du contenu (généralement de type XML).

La figure (II.4) suivante clarifie la structure de L'en-tête du protocole de transport http du message :

```
1 POST /globalweather.asmx HTTP/1.1
2 Host: www.webservicex.net
3 Content-type: text/xml; charset=utf-8
4 Content-Length: length
5 SOAPAction: "http://www.webserviceX.NET/GetWeather"
```

Figure II.4 : L'en-tête du protocole de transport http.

- **L'enveloppe SOAP (SOAP Envelope) :** est la partie principale d'un message SOAP, elle se compose de deux sous-parties : la partie entête et la partie corps du message.

La figure suivante représente le premier partie (entête) [22] :

```
<soap:Header>

<!-- Élément USER : À destination du nœud RightManager -->
      <m:User xmlns:m= "http://www.monsite.com/rights/"
      soap:actor="http://www.monsite.com/rights/RightsManager" >
                    Thunderseb

      </m:User>

 <!-- Élément Session : À destination du nœud final -->

      <m:Session xmlns:m="http://www.monsite.com/session/"
      soap:mustUnderstand ="1" >
                  12AE3C
      </m:Session>

 <!-- Élément Lang : À destination du prochain nœud -->

      <m:Lang xmlns:m = "http://www.monsite.com/lang/"
      soap:actor ="http://schemas.xmlsoap.org/soap/next"
      soap:mustUnderstand = "0" >
                      FR
       </m:Lang>
<soap:Header/>
```

Figure II.5 : Entête de Message SOAP.

La figure montre un exemple de message SOAP requête d'un service web qui donne le factoriel d'un nombre entier [22].

```
<?xml version="1.0" encoding="UTF-8"?>
<S:Envelope xmlns:S="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:SOAP-ENV="http://schemas.xmlsoap.org/soap/envelope/">
    <SOAP-ENV:Header/>
    <S:Body>
        <ns2:Factoriel xmlns:ns2="http://leswebservice/">
            <number>6</number>
        </ns2:factoriel >
    </S:Body>
</S:Envelope>
```

Figure II.6 : Message SOAP requête.

II.5.3.2 WSDL (Web Service Description Language)

WSDL introduit une grammaire commune pour la description des services [20]. Le langage WSDL (W3C), est un langage basé sur XML permettant de décrire et de publier les interfaces et protocoles des services Web d'une manière standard. L'interface d'un service Web décrit en fait tout le fonctionnement d'un service Web, et cache tout le détail de l'implémentation du service Web, donc elle est indispensable pour pouvoir invoquer un service Web par une application cliente, ou un autre service Web permettant une utilisation indépendante de la plateforme utilisée ainsi que du langage utilisé [25].

Dans un fichier WSDL, chaque service web est une entité (entity), qui est définie par des ports (ports), qui sont les points finaux (endpoints) du service. Ces mêmes ports peuvent recevoir et répondre à un ensemble de messages définis par le type du port (portType). Chaque port est, en réalité, la liaison entre le portType et un protocole d'accès, qui nous dit comment les messages doivent être encodés et envoyés au port [26].

Un document WSDL est divisé en deux parties (partie abstract et partie concret), qui contient sept sections :

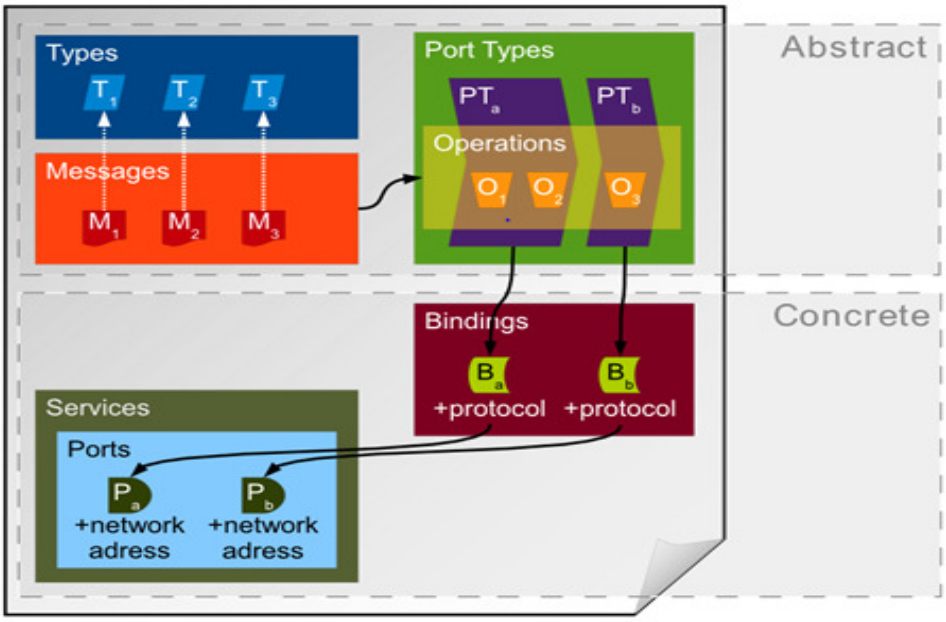


Figure II.7 : La structure d'un document WSDL**.**

Dans la suit, nous définissons chacun des éléments mis en évidence dans (la Figure II.7) :

a. Au niveau abstrait : la description du service web consiste à définir les éléments de l'interface du service web tel que les types de données (Data Types), les messages (Message), les types de ports (Port Type). Ces parties décrivent des informations abstraites indépendantes au contexte de mise en œuvre. On y trouve les types de données envoyées et reçues, les opérations utilisables et le protocole qui sera utilisé [18].

b. Au niveau concret, le service web est défini grâce aux éléments Bindings et Service & Port. Ces deux derniers décrivent des informations liées à un usage contextuel du service web. On y trouve l'adresse du fournisseur implémentant le service, et le service qui est représenté par les adresses des fournisseurs [18].

II.5.3.3 UDDI (Universal Description Discovery and Intégration L'annuaire des services)

La spécification UDDI constitue une norme pour les annuaires de services Web. Il fournit l'infrastructure de base pour la publication et la découverte des services Web. UDDI permet aux fournisseurs de présenter leurs services Web aux clients [27]. Il est Standard fondé sur XML. Il utilise SOAP pour la communication.

L'annuaire UDDI est accessible soit via un navigateur Web qui dialogue avec une application Web dédiée, interface spécifique à l'annuaire accédé, soit par programme, en utilisant l'API (Application Programming Interface).

L'API comporte deux groupes de fonctions :

- *Les fonctions de recherche (Inquiry API) :* navigation, recherche et consultation des informations de l'annuaire ;
- *Les fonctions de publication (Publishing API)* : publication, création, modification ou suppression des informations de l'annuaire [28].

✓ **Consultation de l'annuaire**

L'UDDI peut être vu comme un annuaire contenant (voir la figure II.8) :

- **Les pages blanches** comprennent la liste des entreprises ainsi que des informations associées à ces dernières (coordonnées, description de l'entreprise, identiants...).

- **Les pages jaunes** recensent les services Web de chacune des entreprises sous le standard WSDL.
- **Les pages vertes** fournissent des informations techniques précises sur les services fournis [29].

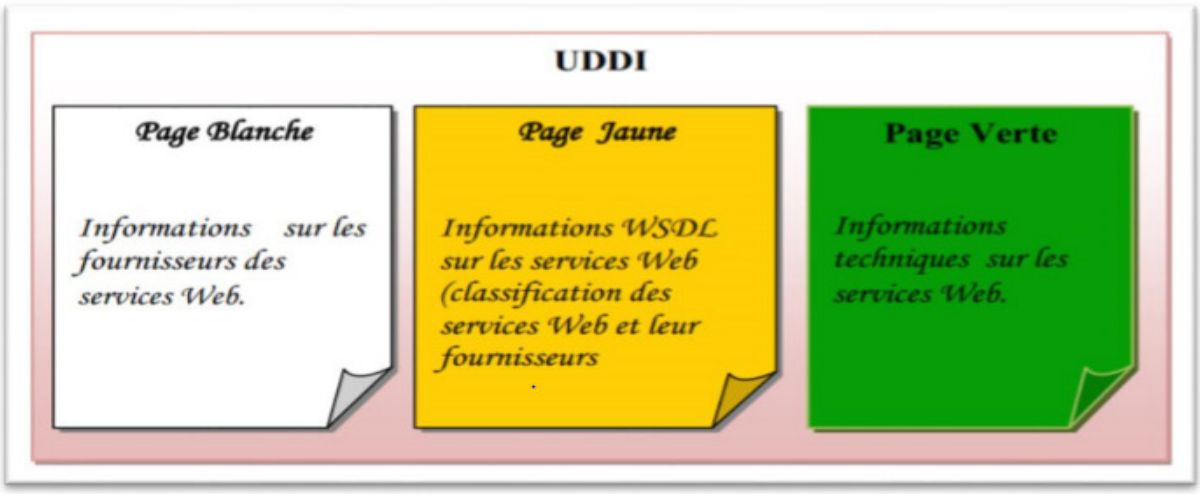


Figure II.8 : Le contenu de l'annuaire UDDI.

II.5.4. Description du modèle fonctionnel des services web

Le fonctionnement des services web passe par plusieurs étapes clarifiées dans (la figure II.9) :

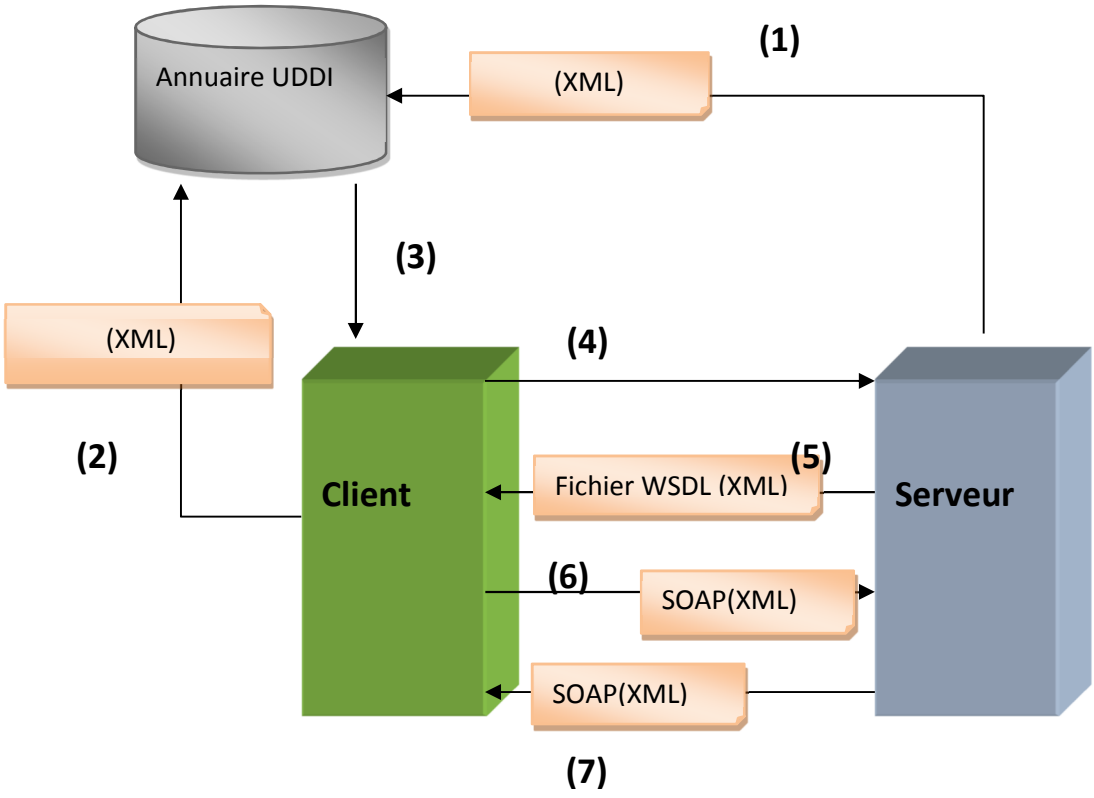


Figure II.9 : Modèle fonctionnel des services web.

(1) : le fournisseur de services web publie ses services web.

(2) : le client recherche un service web avec des caractéristiques.

(3) : l'annuaire a trouvé un service avec les caractéristiques de client, il envoie l'information du serveur qui héberge.

(4) : le client testé : quel est le format d'appel du service que tu proposes ?

(5) : voici mon contrat (WSDL).

(6) : j'ai compris comment invoquer ton service et je t'envoie un document XML représentant ma requête.

(7) : j'ai exécuté ta requête et je te retourne le résultat.

II.6. Composition des Services Web

La composition de services est considérée comme l'une des motivations les plus importantes du paradigme SOA, en effet si une application ou un client requièrent des fonctionnalités, et qu'aucun service n'est seul apte à les fournir, il devrait être possible de combiner ou de composer des services existants afin de répondre aux besoins de cette application ou de ce client. C'est ce que l'on appelle la composition de services

II.6.1 Définitions

La composition permet de combiner des services pour former un nouveau service dit composé ou composite. L'exécution d'un service composé implique des interactions avec des services partenaires en faisant appel à leurs fonctionnalités. Le but de la composition est avant tout la réutilisation de services (simples ou composés) et de préférence sans aucune modification de ces derniers [17].

II.6.2 Cycle de vie d'une composition de services

Le cycle de vie d'une composition de services Web reposant à partir de six activités [30].

- **L'encapsulation de services natifs** : cette première activité permet de s'assurer que tout service peut être appelé lors d'une composition, indépendamment de son modèle de données, de son format de message, et de son protocole d'interaction.
- **L'établissement d'accord d'externalisation :** cette seconde activité consiste à négocier, établir, et appliquer des obligations contractuelles entre les services.
- **L'assemblage de services composants :** cette activité permet de spécifier, à un haut niveau d'abstraction, l'ensemble des services à composer afin d'atteindre l'objectif attendu. Cet assemblage comporte une phase d'identification des services et de spécification de leurs interactions conformément aux descriptions et aux accords entre services.
- **L'exécution de services composants :** cette activité consiste en l'exécution des spécifications de la composition précédemment définies.
- **Le contrôle de l'exécution de services composites :** la phase de contrôle permet de superviser l'exécution de la composition en vérifiant, par exemple, l'accès aux services, les changements de statut, les échanges de messages. Ce contrôle permet de détecter des violations de contrats, de mesurer les performances des services appelés et de prédire des exceptions.
- **L'évolutivité des services :** cette dernière phase permet de faire évoluer la composition en modifiant les altérations de l'organisation de services, en utilisant de nouveaux services, ou en prenant en compte les retours de la phase de contrôle.

II.6.3 Types de composition de services Web

Dans la suite, nous avons présenté la composition des services web et la classification de différentes approches. (Voir la figure II.10) :

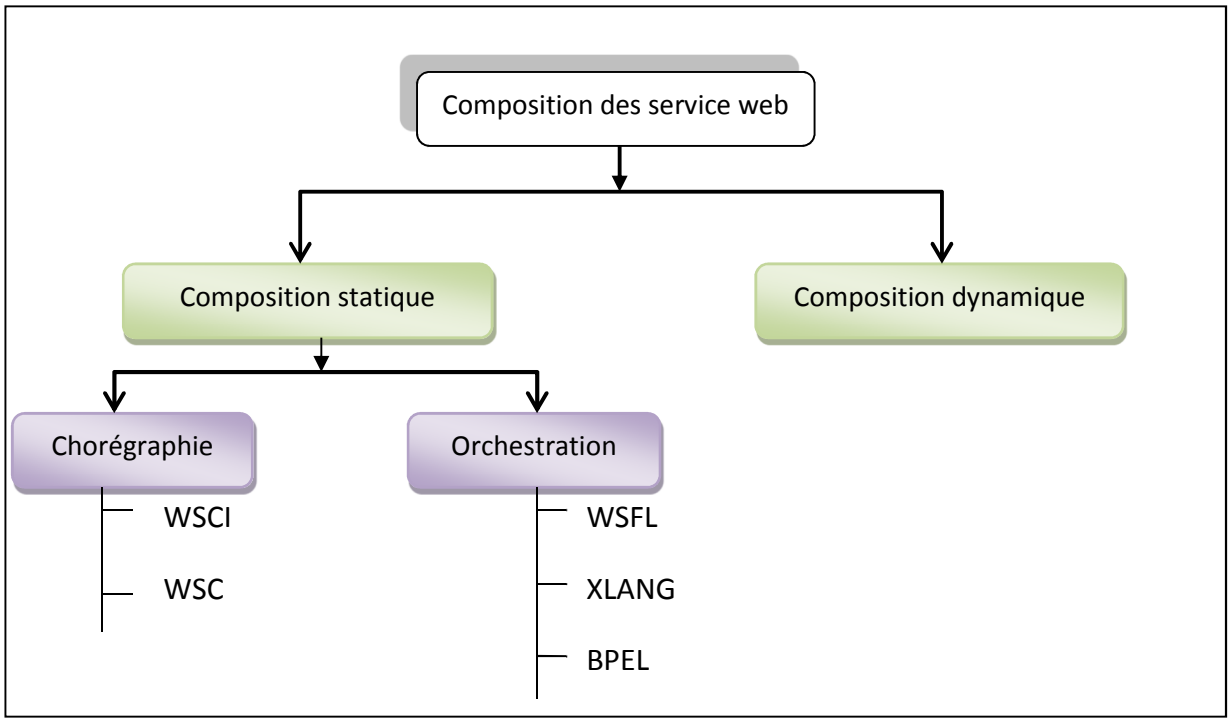


Figure II.10 : Les Approches de la composition des services Web.

II.6.3.1 Composition statique des services

Dans ce type de composition la construction d'un modèle abstrait des tâches qui doit être réalisée au cours de l'exécution de la composition se fait à la conception et/ou la compilation avant que la planification de la composition commence. Ce modèle abstrait est rien qu'une représentation d'un ensemble des tâches et des dépendances entre eux, ou chaque tâche correspond à un service Web. Les services Web participants à la composition sont choisis, liés ensemble, compilés puis déployés [31].

- **L'orchestration** : est un ensemble de processus exécutés dans un ordre prédéfini afin de répondre à un but. Ce type de composition permet de centraliser l'invocation des services Web composant. Chaque service est décrit en termes d'actions internes. Les contrats entre deux services sont constitués selon le processus à exécuter [30] (Voir la figure II.11).

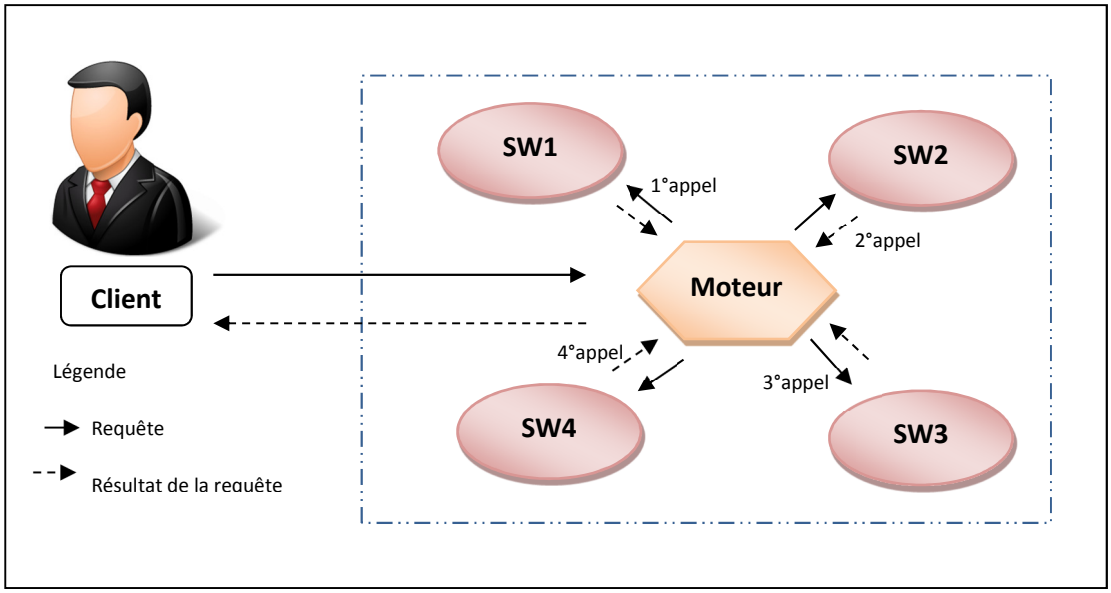


Figure II.11 : Vue générale de l'orchestration.

- **Chorégraphie**

La chorégraphie trace la séquence de messages pouvant impliquer plusieurs parties et plusieurs sources, incluant les clients, les fournisseurs, et les partenaires. La chorégraphie est typiquement associée à l'échange de messages publics entre les services web, plutôt qu'a un procédé métier spécifique exécuté par un seul partenaire[32] .

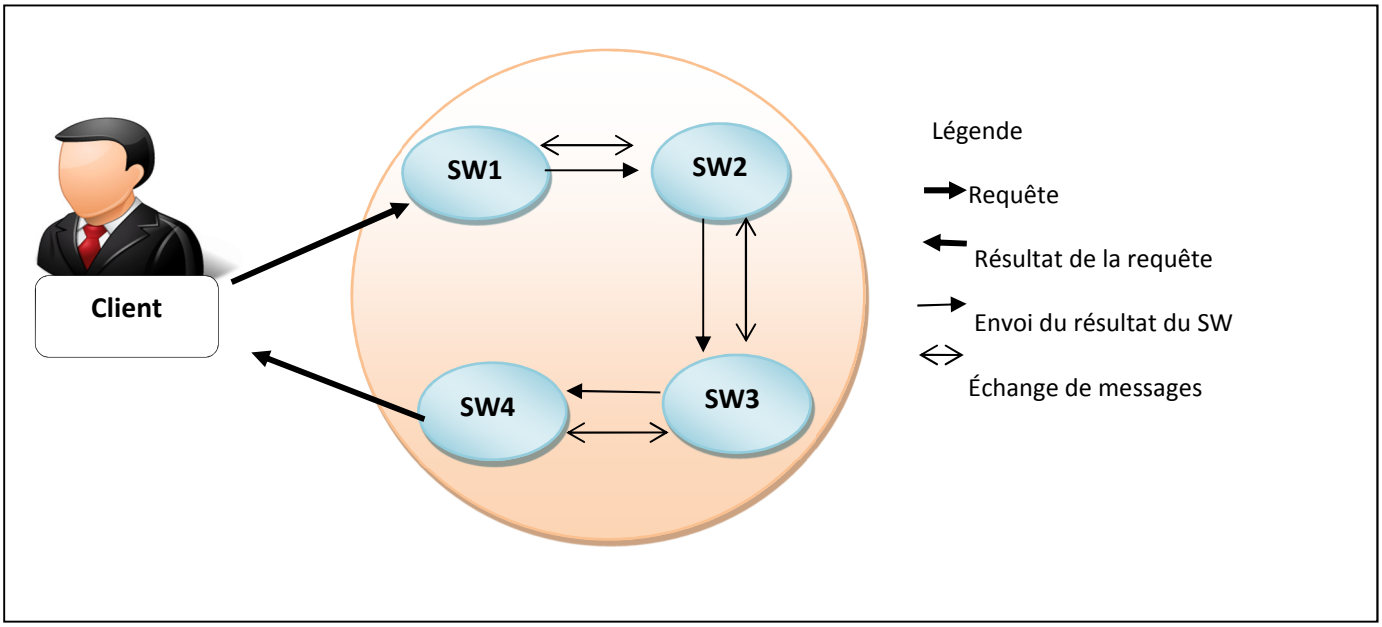


Figure II.12 : vue générale de la chorégraphie.

Il y a une différence importante entre l'orchestration et la chorégraphie de services web.

L'orchestration se base sur un procédé métier exécutable pouvant interagir avec les services web internes ou externes. L'orchestration offre une vision centralisée, le procédé est toujours contrôlé du point de vue d'un des partenaires métier.

La chorégraphie est de nature plus collaborative, chaque participant impliqué dans le procédé décrit le rôle qu'il joue dans l'interaction. Beaucoup de standards se sont intéressés initialement soit à l'orchestration soit à la chorégraphie [32].

II.6.3.2 Composition dynamique des services

La composition dynamique a lieu au moment de l'exécution. Elle permet de créer de manière autonome des services complexes en combinant des composants à la volée en fonction des demandes de l'utilisateur et du contexte. Elle évolue dans des environnements flexibles et ouverts où la sélection et la combinaison des composants sont effectuées à la demande

La technologie de la composition dynamique est généralement contestée par :

- Le grand nombre de services qui deviennent disponibles chaque jour.
- La nature volatile de services web (par exemple : ils peuvent disparaître, être modifiés où être temporairement indisponibles).
- Le nombre sans cesse croissant de fournisseurs de services [21].

II.6.4 Langages de composition de services web

Concerne la composition des services web plusieurs langages ont été proposés tel que :

II.6.4.1 WSCI (Web Services Choreography Interface)

Le langage WSCI a été proposé par Sun, SAP, BEA, et Intalio, et est devenu une note dans le W3C le 8 août 2002. La collaboration est coordonnée de manière décentralisée sans utiliser de processus principal, mais plutôt plusieurs processus distribués. Aussi basé sur XML, propose de se focaliser sur la représentation des services [19].

II.6.4.2 WSCDL (Web Services Choreography Description Language)

Est un langage de description de chorégraphie. Il définit une chorégraphie comme un contrat multipartenaire qui décrit, d'un point de vue global, le comportement commun observable des services participant à une collaboration [19].

II.6.4.3 WSFL (Web Service Flow Language)

IBM a, quant à elle, proposé WSFL pour décrire à la fois le modèle de flux et le modèle global d'un processus métier. Le modèle de flux correspondait au séquencement des activités du processus alors que le modèle global permettait la corrélation entre une activité et une instance d'opération de service Web. Une description WSFL pouvait enfin être exposée par le biais d'une interface WSDL afin de permettre une décomposition récursive d'un processus (utilisation d'une description WSFL comme activité d'une autre description WSFL) [27].

II.6.4.4 BPEL (Business Process Execution language)

Le langage BPEL, nommé aussi BPEL4WS (**B**usiness **P**rocess **E**xecution **L**angage **F**or **W**eb Services**)** ou WS-BPEL, est un standard OASIS d'orchestration de services Web. Ce langage est largement considéré comme étant le plus abouti du marché dans le cadre de la mise en œuvre des architectures orientées services. BPEL définit le processus, l'enchaînement et l'ordonnancement des actions qui seront exécutées par le moteur d'orchestration, agissant comme une machine virtuelle capable d'exécuter le code BPEL [33].

II.7. Conclusion :

Dans ce chapitre nous avons donné une vue générale sur les architectures orientées services (SAO) et les notions et les concepts concernent les services web, donc nous avons décrit, plus particulièrement les caractéristiques des services web et les standards qui leur sont associés, tel que le protocole SOAP, l'interface WSDL, et l'annuaire UDDI.

Ensuite nous avons présenté le problème de composition des services avec l'étude de cycle de vie de ce dernier. À la fin nous avons décrit les types de composition et de langages proposer pour l'implémentation de la composition.

Chapitre III : Cloud Computing

« l'informatique en nuage »

III.1. Introduction :

Aujourd'hui, l'information est disponible partout dans le monde à tout moment grâce à la technologie de l'internet qui se développe plus rapide et de manière progressive. Pour satisfaire l'évolution continue des besoins de différents types d'utilisateurs. Dans ces dernières années, il y a une nouvelle "tendance" qui s'appelle "Cloud Computing » qui s'appuie sur le WEB 2.0. Ce terme utilisé pour décrire plusieurs technologies servant à délivrer différents services, et faciliter le partage de ressources informatiques. Dans ce chapitre, nous allons définir clairement les concepts fondamentaux du Cloud Computing, ses avantages et inconvénients. Et les différentes plateformes disponibles.

III.2. Historique

Historiquement, Il n'y a pas de date-clé à laquelle nous puissions dire que le Cloud Computing est né ! [34] le concept de cloud computing est loin d'être nouveau, il est même présent depuis des décennies. On en trouve les premières traces dans les années 50 quand les premiers ordinateurs accessibles à distance sont apparus (mainframes). L'idée était de partager les ressources disponibles et de les utiliser de façon optimisée [35]. Après, John McCarthy (1927-2011) a estimé dans les années 1960 que « le calcul peut un jour être organisé comme un service d'utilité public » [36].

Cette notion de consommation a été aussi proposée en 1961, lors d'une conférence au MIT (Massachusetts Institute of Technology), par John McCarthy aussi connu comme l'un des pionniers de l'intelligence artificielle (dont il proposa le nom en 1955) et pour avoir inventé du LISP (« list processing » ou bien « traitement de listes ») en 1958. John McCarthy suggéra que la technologie informatique partagée (« timesharing ») pouvait construire un bel avenir dans lequel la puissance de calcul et même les applications spécifiques pouvaient être vendues comme un service public [34].

Le site web Salesforce.com fut le premier hébergeur de Cloud en 1999, suivi en 2002 par Amazon qui propose un ensemble d'hébergement d'application et de stockage. Amazon développe ses services en 2005 (Amazon Web Services) et en 2006 (ElasticCompute Cloud ou EC2). C'est en 2009 que l'utilisation réelle du Cloud survient avec l'arrivée sur le marché de sociétés comme Google (Google App Engine), Microsoft (Microsoft Azure), IBM (IBM Smart Business Service), Sun (Sun Cloud) et Canonical Ltd (Ubuntu Enter- prise Cloud) [37].

III.3. Définition

Le cloud computing possède plusieurs définitions dans la littérature. Permi ces définitions on trouve :

La définition de l'institut national des normes et de la technologie NIST (en anglais, National Institute of Standards and Technology) est : "Le Cloud computing est un modèle qui permet un accès réseau à-la-demande à un pool partagé de ressources informatiques configurables (Par exemple, réseaux, serveurs, stockage, applications et services) qui peuvent être rapidement provisionnées et publiées avec un effort de gestion minimale via un service d'interaction de fournisseur" [37].

Le Cloud computing offre plusieurs services différents, il permet l'accès à des services d'infrastructure (location de capacités de stockage ou de calcul), des services de plateforme (location d'environnements de développement préconfigurés) ou de services d'applications (location d'applications).

Ce modèle de nuage est composé de cinq caractéristiques essentielles, trois modèles de services et quatre modèles de déploiement [41] représente dans la figure suivante :

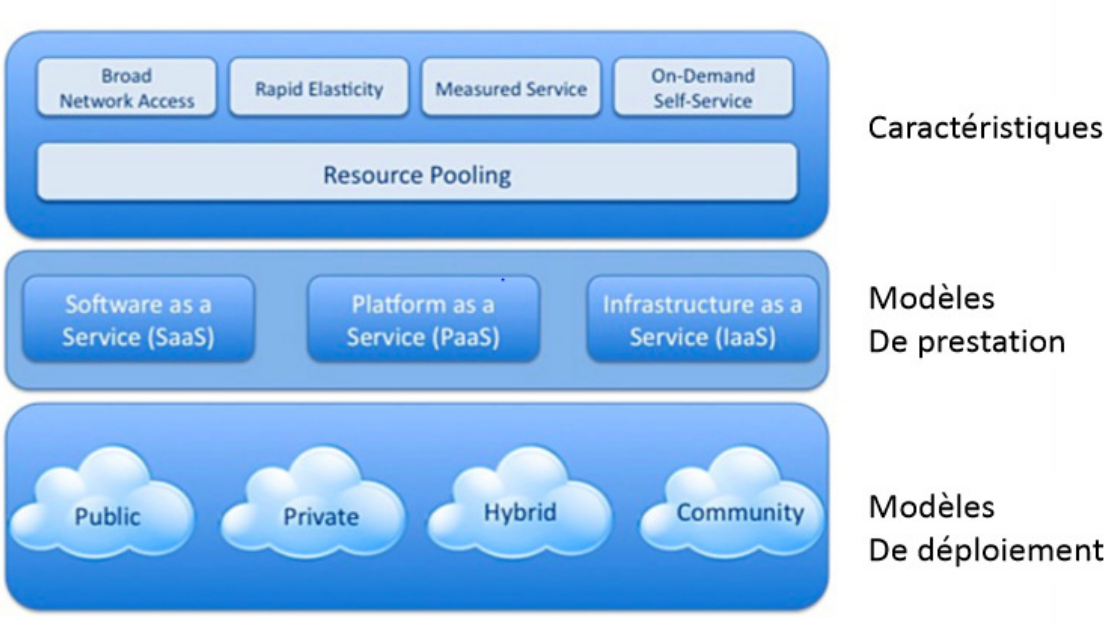


Figure III.1 : NIST – Modèle visuel Cloud computing.

III.4. Les caractéristiques de Cloud

Selon [35] les caractéristiques de Cloud sont :

• libre-service à la demande : les ressources sont disponibles à la demande du client, sans interaction humaine ou re-négociation.

• vaste accès réseau : une connexion internet est requise (ou au moins un accès réseau, dans le cas d'un Cloud local).

• mise en commun de ressources : ou virtualisation, les ressources physiques sont partagées avec d'autres clients et allouées dynamiquement.

• élasticité rapide : les ressources peuvent être échelonnées rapidement (et automatiquement).

• services mesurés : ou "pay-as-you-go", le client paye uniquement ce qu'il utilise.

III.5. L'architecture Cloud

Le Cloud se repose sur 3 modèles fondamentaux qui n'ont pas le même niveau de maturité : la figure (1) représente l'organisation en couches de la pile de Cloud de l'infrastructure physique jusqu'à niveau d'application. Ces niveaux d'abstraction peuvent également être considérés comme une architecture en couches [39] clarifie dans la figure suivent :

Modèle de service	Outils d'accès et de gestion	Services offerts
SaaS 	Navigateur Web	Réseaux sociaux, suites bureautiques, CRM, traitement vidéo.
PaaS 	Environnement de développement	Langages de programmation, systèmes, gestionnaire de données structurés.
IaaS 	Gestionnaire d'infrastructure virtuelle	Serveurs de calcul, stockage de données, Pare-feu.

Figure III.2 : pile de services Cloud.

III.5.1. SaaS :

Le niveau le plus haut, sa signification est « Software as a Service » en Français « L'application en tant que service ». Regroupe des services rendus accessibles aux utilisateurs finals au moyen d'un client léger (application web, application mobile, client bureau, ...) [41]. Le modèle SaaS permet de faciliter la navigation et l'exploitation des ressources informatiques sur l'Internet par ce que [42] :

- L'entreprise Cloud responsable à la mise à jour du logiciel. Donc la mise à jour se fait automatiquement.
- L'utilisation d'un logiciel sur le Cloud est assurée avec un contrat de niveau de service, qui assure la qualité de service, et les droits et les devoirs du client vis-à-vis de l'entreprise.
- Lorsque les logiciels sont installés et exécutés sur le Cloud, n'a pas à se soucier à compatibilité avec le côté matériel.

La figure suivante représente le modèle Software as a Service :

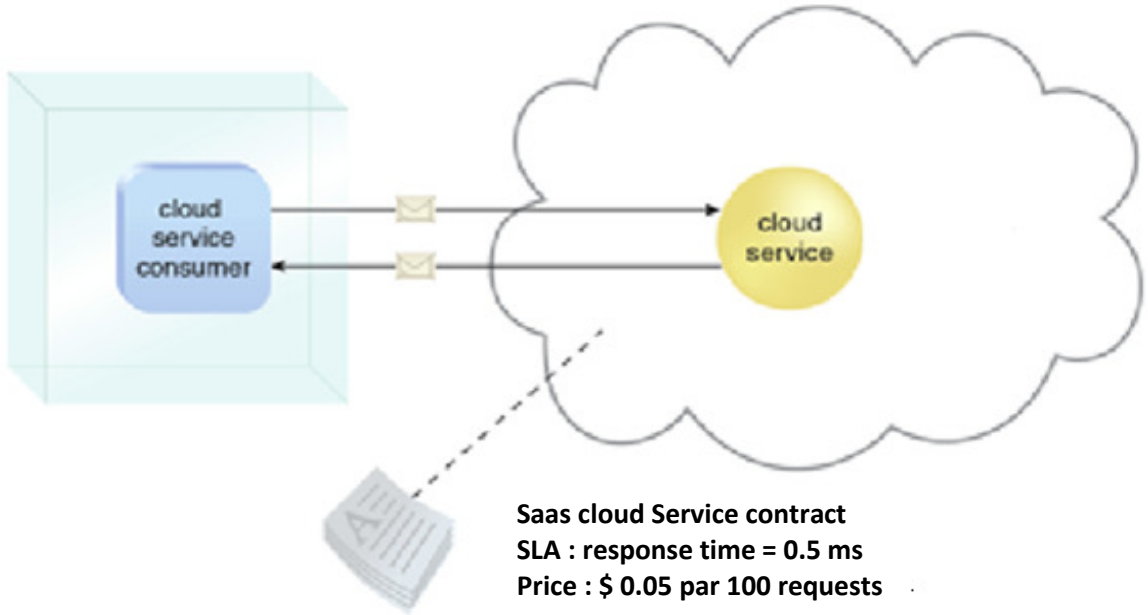


Figure III.3 : le modèle de service SaaS.

III.5.2. PaaS :

Qui signifie « Platform as a Service » en Français « plate-forme en tant que service ». La couche intermédiaire regroupe des services qui ciblent les concepteurs de services logiciels. Ce modèle représente une variante du modèle SaaS, sauf qu'au lieu d'offrir une application comme service il offre un environnement de développement d'applications comme service, ce modèle permet au développeur d'utiliser et réutiliser un large spectre d'outils, de composant logiciels, et de blocs de codes ce qui lui permet de développer rapidement son application. Avec ce modèle les fournisseurs des services Cloud offrent un ensemble d'outils et d'environnement de : développement, exécution, déploiement, distribution et de payement des applications crées au niveau de la plateforme. Le fournisseur de service d'une plateforme de développement est aussi destiné à produire et continuellement proposer et développer des standards ainsi que des outils de développement pour faciliter la tâche au programmeur et développeurs d'applications Cloud. Une plateforme de développement sur le Cloud permet de réduire considérablement les couts de déploiement des applications avec une grande rapidité de propagation vue que les fournisseurs des services Cloud placent assez de canaux de communications et de liaisons entre l'application développée et l'infrastructure de déploiement [42].

Parmi les fournisseurs des services PaaS on trouve : Microsoft avec AZURE, Google avec Google App Engine et Orange Business Services.

La figure suivante représente le modèle de service PaaS :

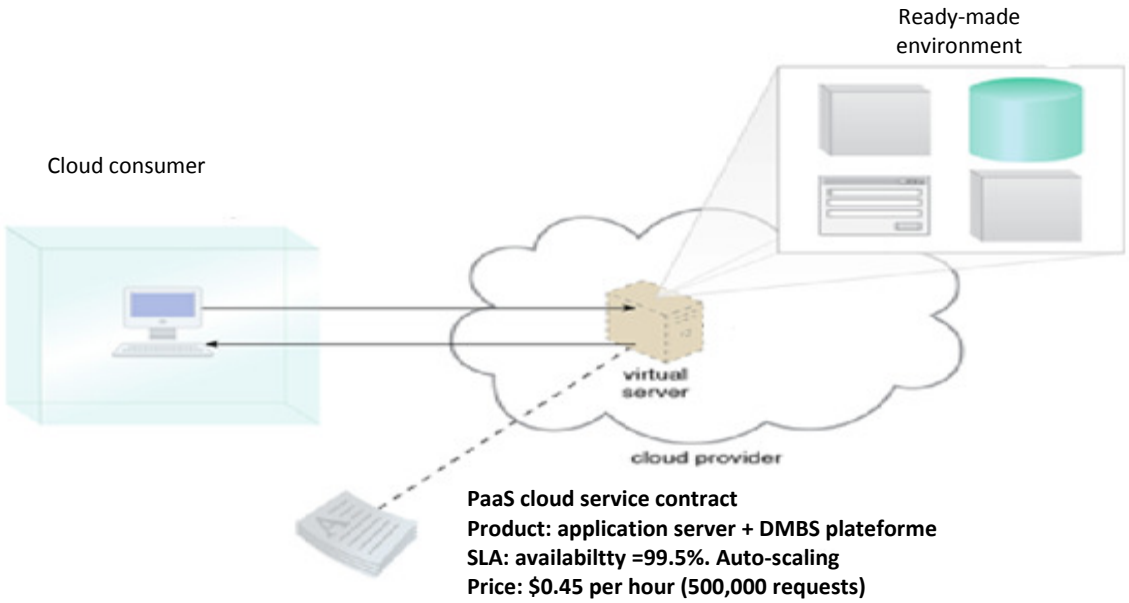


Figure III.4 : le modèle de service PaaS.

III.5.3. IaaS :

Qui signifie « Infrastructure as a service » en Français « L'infrastructure en tant que service ». L'infrastructure en tant que service : c'est le service de plus bas niveau. Il consiste à offrir un accès à un parc informatique virtualisé. Des machines virtuelles sur lesquelles le consommateur peut installer un système d'exploitation et des applications. Le consommateur est ainsi dispensé de l'achat de matériel informatique. Ce service s'apparente aux services d'hébergements classiques des centres de traitement de données, et la tendance est en faveur des services de plus haut niveau, qui fait d'avantage abstraction de détails Techniques [43].

Ce modèle est représenté dans la figure suivante :

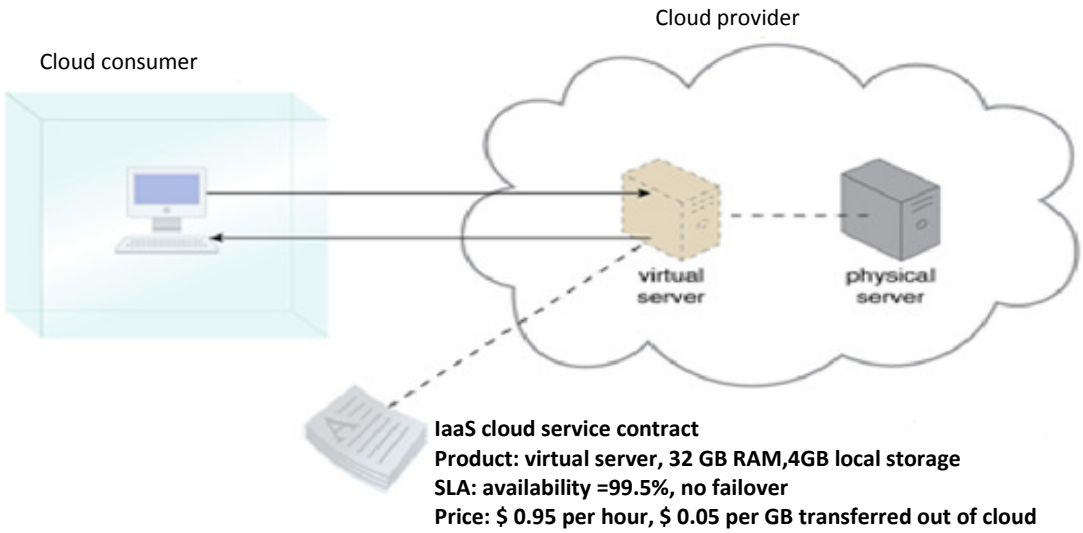


Figure III.5 : le modèle de service IaaS.

III.6. Modèles de déploiement des Cloud

Quatre grandes familles de Cloud existent en fonction du mode de déploiement, comme illustré dans (la figure III.6) :

III.6.1. Cloud privé

Ce type de Cloud a une vocation purement interne. La puissance de calcul, les capacités de stockage et le réseau sont accessibles uniquement en interne [48].

L'infrastructure d'un nuage privé n'est utilisée que par un unique client. Elle peut être gérée par ce client ou par un prestataire de service et peut être située dans les locaux de l'entreprise cliente ou bien chez le prestataire, le cas échéant. L'utilisation d'un nuage privé permet de garantir, par exemple, que les ressources matérielles allouées ne seront jamais partagées par deux clients différents [49].

III.6.2. Cloud public

Est réservé pour un usage externe. La solution est fournie à toute catégorie d'utilisateurs. L'infrastructure quant à elle reste à la charge de l'entreprise qui propose les services [44]. Le principal avantage pour un client est son exonération des frais d'investissement en infrastructures. Vu la prise en charge des coûts des matériels, applications

et réseaux par les fournisseurs Cloud, les clients peuvent créer et installer leurs environnements d'applications d'une manière rapide et moins coûteuse [38].

III.6.3. Cloud communautaire

Est un cas particulier dont le Cloud privé [48], constitué d'une infrastructure partagée entre plusieurs organisations ayant des préoccupations communes (justice, éducation, santé, industrie, culture, etc.). Il est très important de noter que c'est le seul modèle de Cloud qui garantit actuellement, la localisation et le contrôle total des données transitant sur le réseau [47].

III.6.4. Cloud hybride

L'infrastructure Cloud est une combinaison de deux ou plusieurs infrastructures Cloud distinctes (privée, communautaire ou publique) qui restent des entités à part entière mais sont liées entre elles par des technologies standardisées ou propriétaires afin de permettre la portabilité des données et des applications [45].

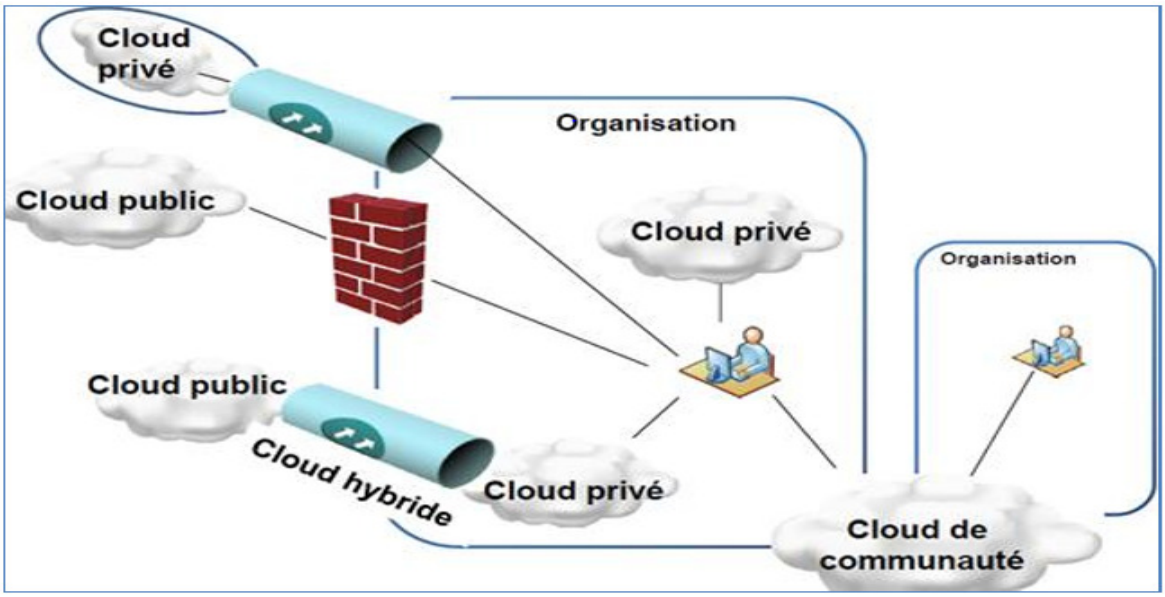


Figure III.6 : Modèles de déploiement Cloud Computing.

III.7. Les enjeux du cloud computing

Dans l'utilisation des Clouds il existe plusieurs contraintes, par exemple :

a. La Qualité de service :

Le premier principe de client est la qualité de service, c'est un concept de gestion qui a pour but d'optimiser les ressources d'un réseau (en management du système d'information) [50]. Afin d'assurer un certain niveau de qualité à leurs clients, les fournisseurs Cloud disposent d'un système de contrôle pour enregistrer les QdS (Qualité de Service) et mis à jour des Machines virtuelles (MV) du système [42].

b. Le transfert de donnés :

Le transfert de données dans Cloud soufre du même problème que celui pour Internet. Mais avec le Cloud le problème est plus grave puisqu' il s'agit de transférer des quantités gigantesques d'informations à partir d'entreprises qui génèrent des dizaines de tira octet chaque jour vers le Cloud. Par exemple avec une bande passante de 18 Mb/s, le transfert de 10 TO d'un Datacenter situé à Alger vers un Datacenter situé à Helsinki prend environ 54 jours. Amazon par exemple, propose le service AWS Import/Export destiné à transporter les disques durs de stockage aux Datacenter d'Amazon, afin de pourvoir réduire le temps de transfert des donnés [42].

c. La sécurité des Clouds :

La sécurité est une des premières préoccupations des entreprises qui transfèrent des éléments de leur infrastructure vers le Cloud. Il existe par ailleurs une contradiction, certaines entreprises craignent la perte d'informations capitales (brevet, données clients), les atteintes à leur réputation, les actes malveillants et les interruptions de services. Cependant, les entreprises reconnaissent que le Cloud peut améliorer leur sécurité car les prestataires doivent répondre aux exigences de sécurité et aux obligations réglementaires, tout en garantissant des performances et le respect des contrats de niveau de service [51].

III.8. Cloud Computing : avantages et inconvénients

III.8.1. Avantages

En adoptant la technologie de Cloud computing, on peut trouver plusieurs avantages. Parmi ces avantages, il y a selon [34] [39] [40] [42] :

- Réduction des couts d'utilisation : le modèle économique du Cloud permet aux clients de réduire et de contrôler leurs dépenses, puisqu'ils ne payent que ce qu'ils utilisent comme ressources Cloud.
- Réduction des couts de gestion : avec le Cloud les entreprises non plus à se soucier de la gestion des ressources
- Grande capacité de stockage
- Redondance de stockage de données : en cas un disque dur tombe en panne, les données restent en sécurité et disponibles automatiquement sur un autre disque et Système anti-désastre (la récupération des données et des applications).
- Gestion des mises à jour plus simple et rapide.
- Des ressources et services plus rapides à allouer et plus simple à utiliser avec la possibilité de partager les ressources
- Mobilité et accès facile : l'utilisateur peut à tout moment et à partir de n'importe quel appareil se connecter à ses applications (Any where – Any time).
- Augmentation de la puissance de calcul.
- Réduction des coûts de développement.

II.8.2. Inconvénients :

Le Cloud Computing n'a pas que des avantages, il possède aussi quelques inconvénients qui sont abordés dans [34], [39], [42]. Parmi ces inconvénients, il y a :

- Conformité réglementaire : lors du transfert des données du client vers le fournisseur du service Cloud, le client est seul responsable de l'intégrité et de la sécurité des données.
- Confidentialité des données (Vie privée et lois des confidentialités) : les lois qui régissent la préservation des données privées est différente d'un pays à un autre (par exemple les USA n'a pas les mêmes lois que L'union européenne), ce qui peut mettre les clients en situation de confusion par rapport aux services Cloud à utiliser, et cela

peut limiter le transfert de données d'une ressource Cloud vers une autre qui se trouve dans un autre pays.

- Identification des clients et Faible contrôlabilité : avec l'utilisation croissante du Cloud et l'utilisation multilocation de ces ressources, il devient de plus en plus difficile d'identifier par qui et de quel endroit les données ont été modifiées ?!
- Faible niveau de la qualité de service dans le réseau et dépendance des services.
- Manque de fonctionnalités d'audit.
- Nécessité d'un accès réseau constant.

III.9. Les principaux acteurs du Cloud :

Le stockage en ligne sur le Cloud est désormais entré dans les mœurs. Par conséquent, il y a plusieurs acteurs dans le Cloud, on a représenté quelques acteurs dans la table suivante :

	Caractéristique	**Compatibilité**
Google	-les utilisateurs de Gmail ou YouTube bénéficient directement d'un compte Google Drive. -15 GO espace de stockage gratuit. -si dépasse 15GO, le stockage payement. -Google Drive fonctionné en créant un dossier local sur le PC de l'utilisateur synchronisé avec sa version en ligne.	Windows, macOS, Android, IOS.
Microsoft	- 5 Go de stockage gratuit. -Vous pouvez aussi payer 2 € par mois pour obtenir 50 Go ou 69 € par an pour 1 To. -les fichiers et dossiers peuvent être créés sur le web. -Moteur de base de données sans verrou fournissant des disques SSD haute performance pour le cloud.	Windows, macOS, Android, IOS, Windows Phone.

Amazon	-5 Go de stockage gratuit. -l'abonnement à 70 € par an pour un espace de stockage illimité. -Ce Cloud est plus destiné à stocker des photos et des vidéos que des fichiers. Par exemple, il n'est pas possible de partager des dossiers mais seulement des fichiers individuels (et dans la limite de 25 fichiers d'un coup). -Amazon RDS : pour les bases de données relationnelles. -Amazon DynamoDB - Avec ses services de base de données NoSQL gérés.	Windows, macOS, Android, IOS.

Tableau III.1 : la différence entre les principaux d'acteurs de Cloud.

III.10. Conclusion

Dans ce chapitre, nous avons présenté le concept de Cloud Computing. Et on a commencé par l'historique et définition de cloud. Ensuite, nous avons abordé l'architecteur de Cloud, tel que ce dernier est composé : de cinq caractéristiques, trois modèles de service et quatre modèles de déploiement. Après avoir les enjeux du Clouds. En plus les avantages et les inconvénients. Et à la fin, on est terminé par quelque acteur principal de cloud.

Chapitre IV : Conception

IV.1 Introduction

Au niveau de ce chapitre nous allons présenter notre approche, commençons par la conception de notre plateforme, leur architecture conceptuelle de fonctionnement, les modules qui composent la plateforme, l'architecture globale, les utilisateurs qui peuvent utiliser la plateforme et les tâches effectuées sur la plateforme.

IV.2 Approche proposée

L'approche proposée est conçue pour la création d'une plateforme basée sur les services Web, qui soit des services simples ou des services composites. Dans notre plateforme nous proposons d'organiser la plateforme à des modules (module de gestion des utilisateurs, publication, recherche des service et module de composition), chaque module va interagir avec les autres modules pour garantir le bon fonctionnement de la plateforme. Chaque institution peut accéder à la plateforme soit pour la publication ou bien l'invocation des services d'une façon sécurisée. Les bases de données relatives à une institution sont stockées sur le cloud, pour faciliter l'accès et pour bénéficie de l'espace de stockage à cause de volumes des données stockées.

IV.3 L'architecture conceptuelle de la plateforme

La (figure IV.1) montre les différents composants et les entités participées à notre plateforme :

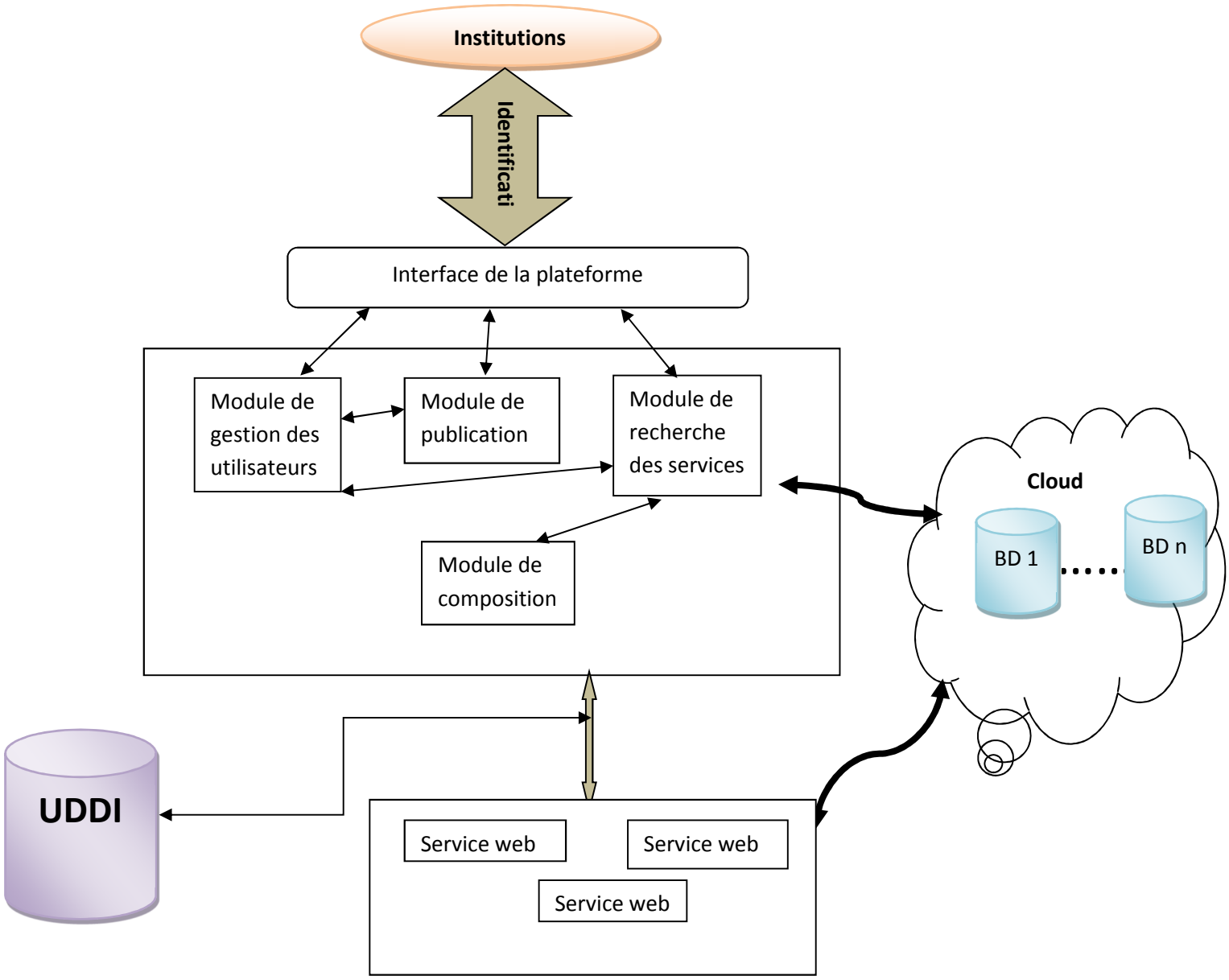


Figure IV.1 : L'architecture global de notre plateforme.

Comme illustré dans (la figure IV.1) l'architecture que nous proposons suppose l'accès de plusieurs institutions à notre plateforme. Elle est constituée d'un annuaire UDDI et quatre modules (module de gestion des utilisateurs, module de recherche des services, module de publication des services et module de composition), Ils interagissent entre eux pour permettre à la plateforme de fournir ses services. La publication des services n'est possible que par les utilisateurs authentifiés et autorisés. Ceci est réalisé par une interaction entre le module de gestion des utilisateurs et le module de publication de services. L'accès à la plateforme pour

servir des citoyens nécessite aussi une authentification, puis selon ses autorisations, un utilisateur peut invoquer des services, cette opération nécessite des interactions entre le module de gestion des utilisateurs, le module de découverte et le module de composition. Ce dernier, afin de fournir la composition adéquate, il interagit avec le module de découverte pour récupérer les profils des services web permettant la réalisation de la composition.

Elle connecte avec les Clouds pour récupérer les données.

Après l'explication de l'architecture de la plateforme on va présenter le fonctionnement de cette dernière par (la figure IV.2).

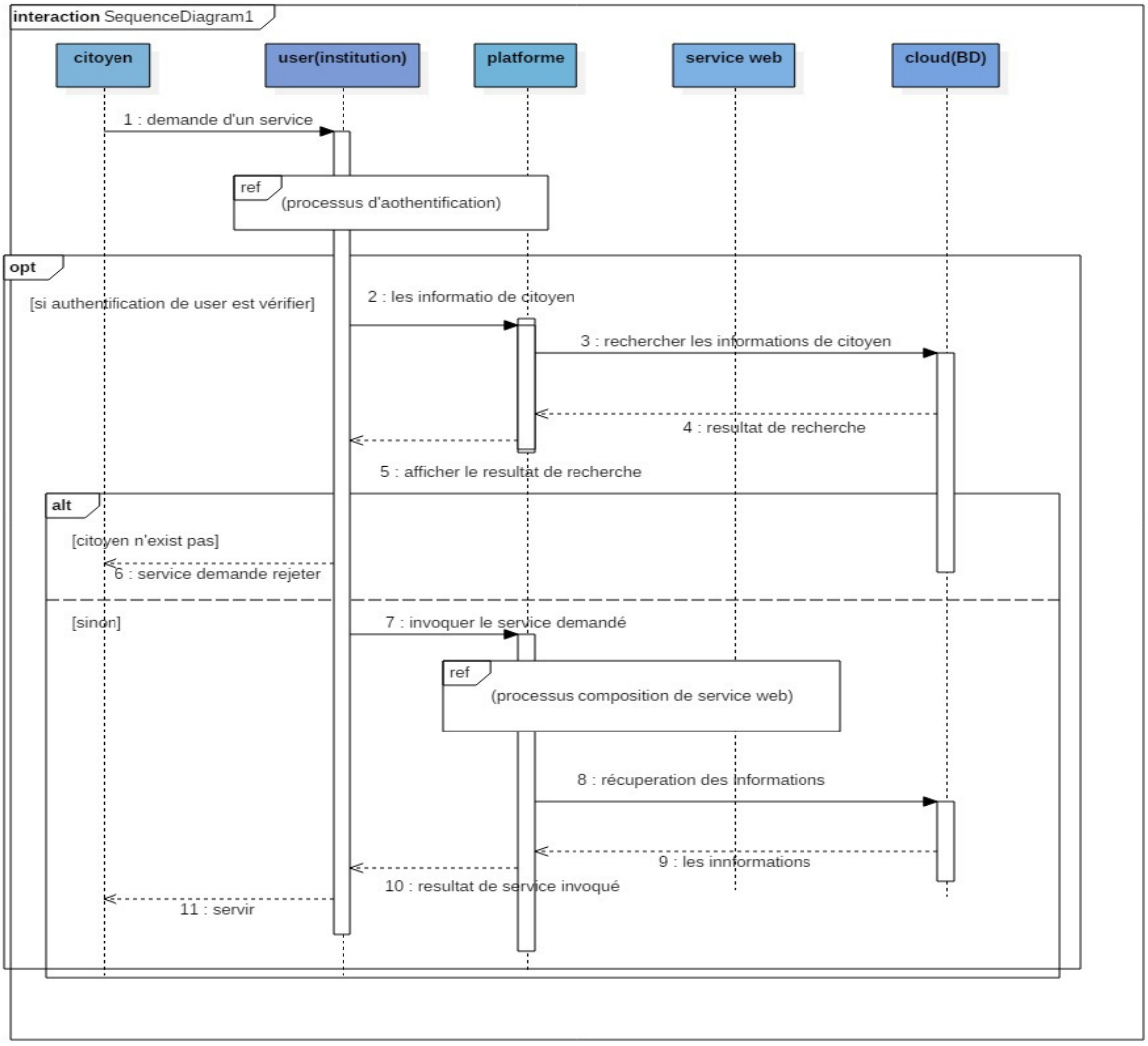


Figure IV.2 : Fonctionnement de la plateforme.

Comme illustré dans (la figure IV.2) notre plateforme fonctionne comme la suite : dans une première étape le citoyen demande un service à une institution. L'institution va contacter notre plateforme pour répondre à cette demande. Donc l'institution exécute le processus d'authentification pour accéder à la plateforme. Si l'authentification est valide, l'institution remplit les informations de citoyen et la plateforme recherche aux les informations relatives dans le cloud (Bd). Si les informations de citoyen n'existent pas, l'institution rejet le service demandé, sinon l'institution invoque les services demandés. Alors la plateforme va exécuter le processus de composition si les services existés ne sont pas satisfaits, puis récupérés les informations de citoyen du cloud, et affichée le résultat à l'institution, ce dernier serve le citoyen.

IV.4 Description des modules

IV.4.1. Le module de gestion des utilisateurs

Ce module géré par l'administrateur de la plateforme, il permet d'organiser l'accès à la plateforme par les utilisateurs des différentes institutions pour publier des services et servir des citoyens. Ce module inclut l'authentification et la gestion des autorisations pour la publication et l'accès aux services web. La figure suivante présente un exemple d'accès à la

plateforme :

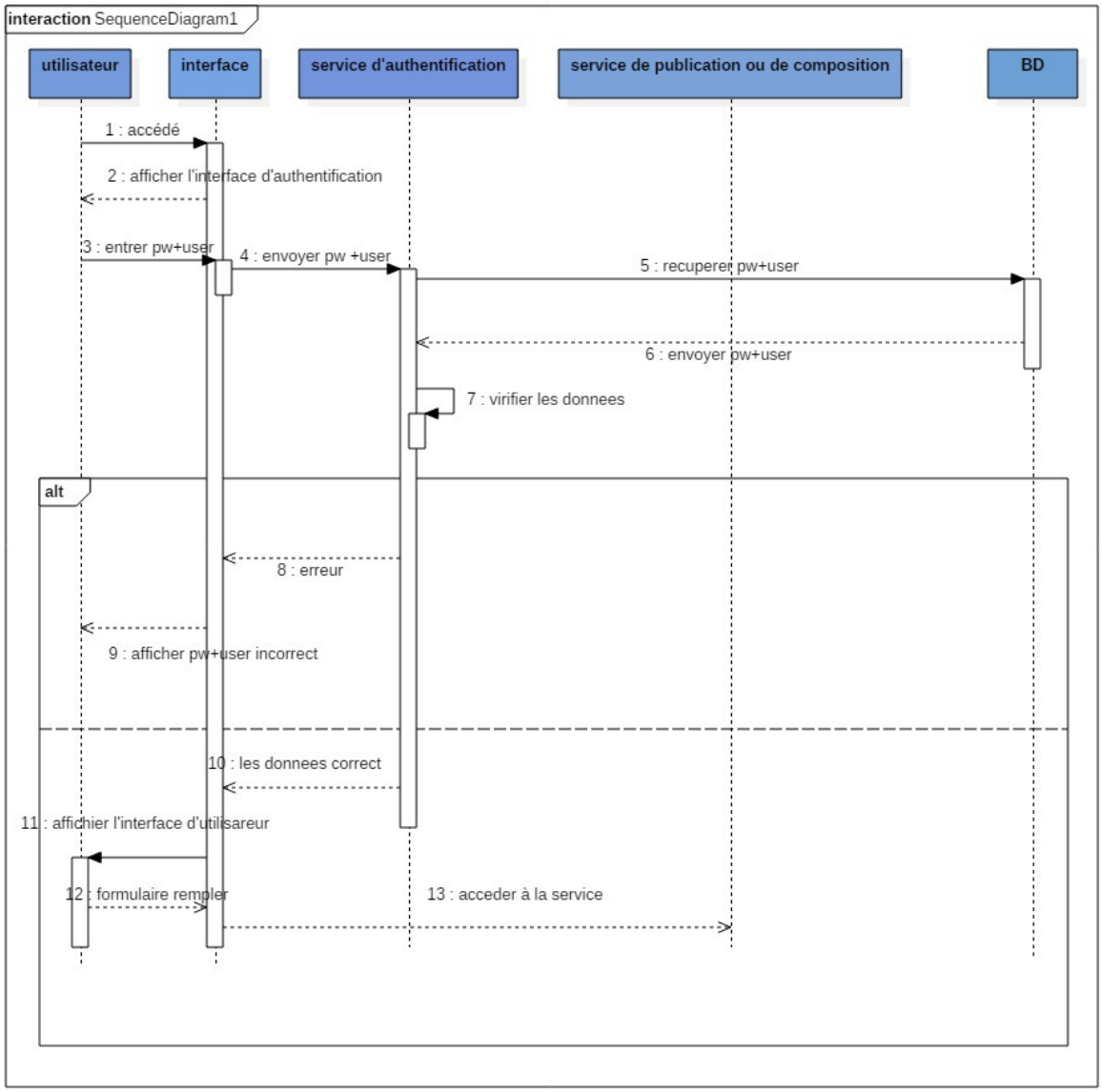


Figure IV.3 : module de gestion d'utilisateur.

Les points ci-dessous représentent les étapes pour l'accès à la plateforme :

1. L'utilisateur demande l'accès à la plateforme.
2. La plateforme affiche interface de l'authentification.
3. L'utilisateur ente son nom utilisateur et son mot de passe.
4. La plateforme récupère les donnes d'authentification qui se trouvent à la BDD.
5. La plateforme vérifie la validité de l'authentification.
6. Si l'authentification est incorrecte, la plateforme affiche un message d'erreur.

7. Sinon la plateforme affiche les services autorisés à l'utilisateur authentifié.
8. L'utilisateur sélectionne les services envisagés pour l'utiliser.

IV.4.2 Le module de publication

Sert d'interface entre la plateforme et le fournisseur du service web. Il permet à une institution de publier et de mettre à jour des services web dans l'annuaire UDDI, l'administrateur des services web de l'institution demande la publication ou la mise à jour des services web. La publication d'un service web permet de rendre des parties des informations personnelles des citoyens, relatives à l'institution et au service, disponible, dans le Cloud selon des autorisations. La publication est gérée par l'administration de l'institution et le module de publication est géré par l'administration de la plateforme. La figure suivante représente le processus de publication d'un service :

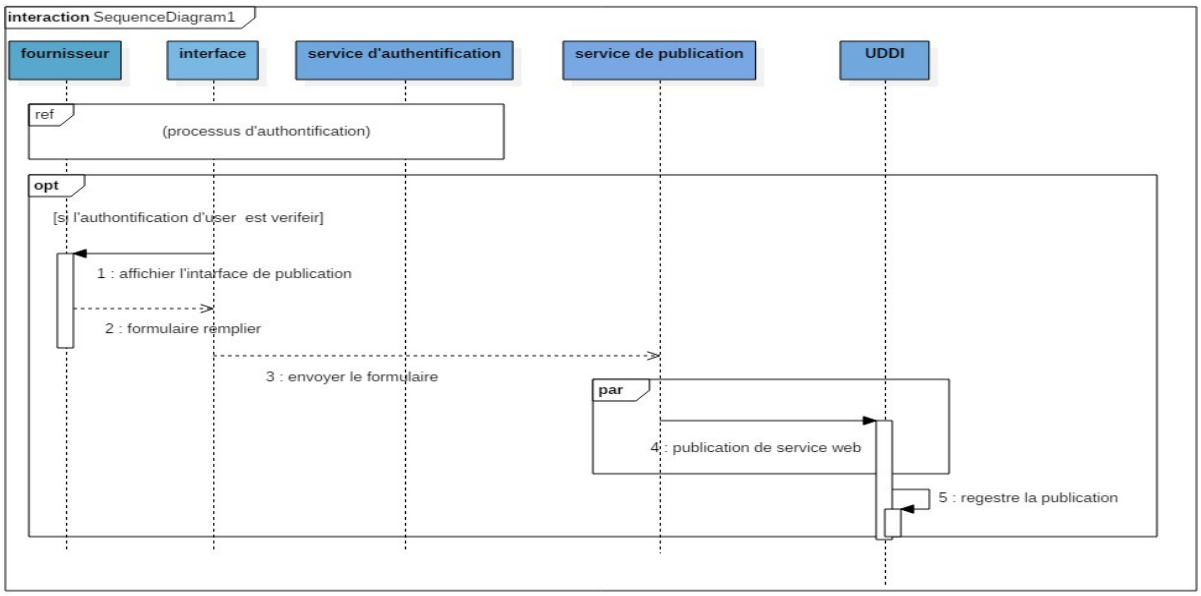


Figure IV.4 : module de Publication.

Ces points ci-dessous représentent les étapes pour publier un service Web :

1. Le fournisseur s'authentifier par son nom d'utilisateur et son mot de passe.
2. Le service d'authentification vérifie si le nom 'utilisateur et le mot de passe sont corrects.
3. La plateforme vérifie l'authentification.
4. Si l'authentification est vérifiée : la plateforme affiche l'interface de publication.

5. Le fournisseur remplie le formulaire de publication et publie le service dans.
6. La plateforme invoque le service web de publication.
7. L'annuaire UDDI registre la publication.

IV.4.3 Le module de découverte

Sert pour la recherche des services web qui a répondu aux besoins signalés dans sa requête (les besoins nécessaires pour la satisfaction de la requête du client), Il se base sur le profil du service web invoqué et les profils des services web publiés par les autres institutions. Ce module peut être utilisé pour la composition des services web.

IV.4.4 Le module de composition

Le module de composition combine, selon le résultat retourné par le module de découverte, les services web pour former le service invoqué. La composition des services permet de récupérer des parties des informations personnelles à partir des autres institutions que le service invoqué va les utiliser pour servir le client. La (Figure IV.5) illustre le scénario de composition :

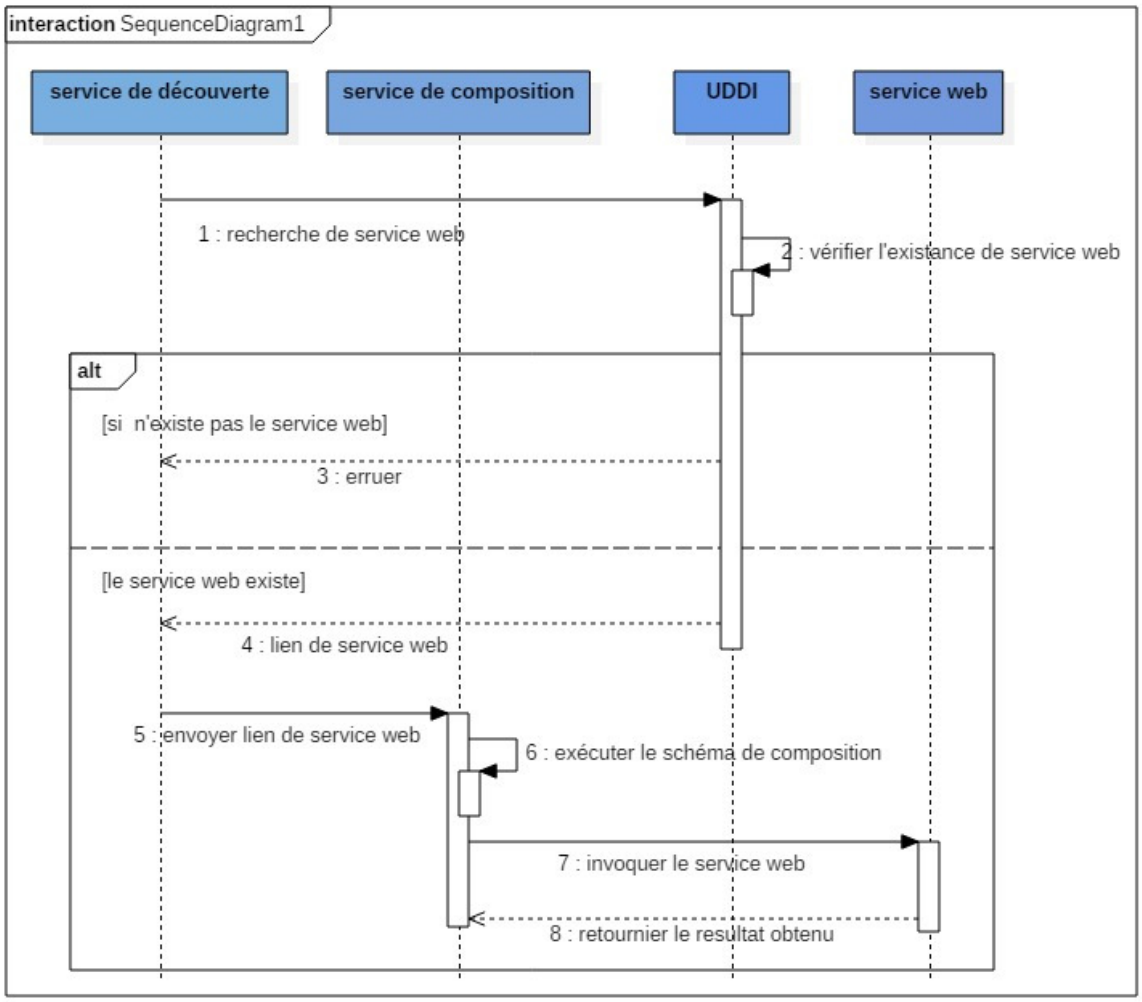


Figure IV.5 : module de Composition.

Ces points ci-dessous représentent les étapes de composition des services Web :

1. L'administrateur de la plateforme reçoit des demandes du service composé.
2. Après le processus de l'authentification et l'accès à la plateforme.
3. L'administrateur invoque le service de découvert pour rechercher si les services demandés disponible sur la plateforme.
4. Si les services sont indisponibles, la plateforme affiche un message d'erreur.
5. Sinon si les services sont disponibles, il récupère les liens.
6. Invoque le service composé et retourne le résultat obtenu.

IV.5 Les entités de la plateforme

La figure ci-dessous présente le diagramme de classes de notre plateforme :

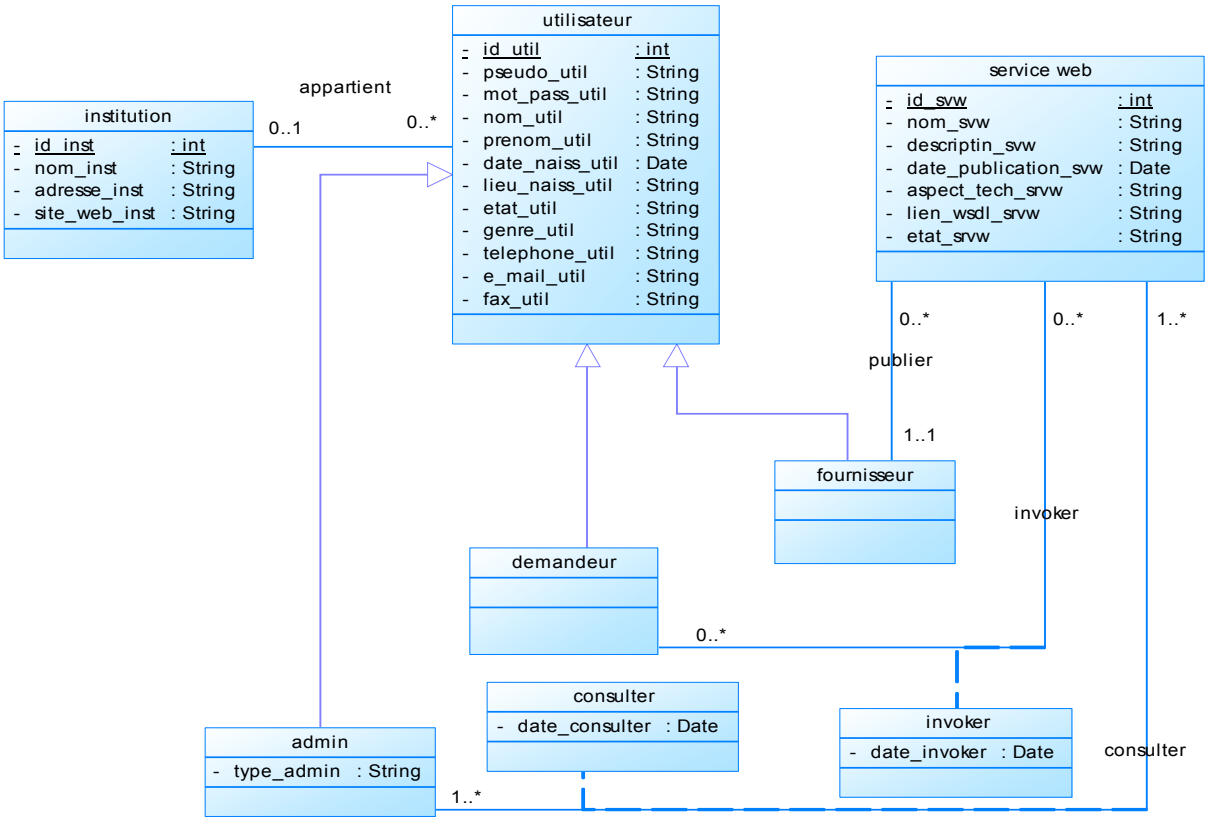


Figure IV.6 : Diagramme de classe de la plateforme.

IV.6 Les types d'utilisateur

Nous avons défini quatre types d'utilisateurs qu'ils peuvent utiliser la plateforme :

IV.6.1. Administrateur de la plateforme

C'est la personne qui a le droit d'accès à tous les ressources de la plateforme. Les tâches effectuées par l'administrateur sont :

- Ajouter ou modifier une institution.
- Ajouter ou modifier un administrateur d'institution.
- Rechercher des services web.
- Consulter un service web.
- Composer des services web.
- Activer ou désactiver un compte d'administrateur d'institution.
- Modifier son mot de passe.
- Modifier le mot de passe pour administrateur d'institution.

IV.6.2 Administrateur d'institution

Est un utilisateur moins prioritaire qu'un administrateur de plateforme. Pour utiliser la plateforme doit s'authentifier afin que la plateforme déterminée ses droits, les tâches effectuées par ce type d'utilisateur sont :

- Ajouter ou modifier un utilisateur de l'institution de cet administrateur.
- Consulter les services web publier de cette institution.
- Rechercher des services web.
- Modifier son mot de passe.
- Modifier les mots de passe de ses utilisateurs.

IV.6.3 Fournisseur

Est un utilisateur appartient à une institution. Pour utiliser la plateforme doit S'authentifier afin que la plateforme déterminer ses droits. Il accède à la plateforme pour publier ou modifier un service.

IV.6.4 Demandeur

Est un utilisateur appartient à une institution. Pour utiliser la plateforme doit S'authentifier afin que la plateforme déterminer ses droits. Il accède à la plateforme pour invoquer ou demander un service.

IV.7 Conclusion

Dans ce chapitre nous avons représenté l'approche proposée de la plateforme, comme nous avons présenté l'architecture conceptuelle, la description des modules et l'interaction entre ces modules.

Chapitre V : Implémentation et étude de cas

V.1 Introduction

Dans ce chapitre nous allons présenter l'implémentation et la réalisation de notre plateforme, mais avant de commencer l'implémentation de notre application, nous allons tout d'abord spécifier les outils utilisés qui nous ont semblés être un bon choix vu les avantages qu'ils offrent.

V.2 Les outils et les langages utilisés en développement

V.2.1 Power designer 16.5

Pour faire la modélisation de notre plateforme, on a choisi l'outil Power designer, c'est un logiciel permettant de réaliser tous les types de modèles informatiques, reste un des seuls qui permet de travailler avec la méthode Merise. Ainsi, ce logiciel, nous a aidé dans la réalisation de nos modèles UML. Il est à noter donc, que ce logiciel, permet d'améliorer la modélisation, les processus, le coût et la production d'applications.

V.2.2 langage JAVA

Pour le langage de programmation notre choix s'est porté sur le langage Java EE (Java Enterprise Edition), et cela parce que Java EE est une spécification pour la technique Java de Sun plus particulièrement destinée aux application d'entreprise, il permet de faciliter la création d'application réparties et d'application web, c'est un langage orienté objet simple et libre, il peut être utilisé sous Windows, sous Linux et sur d'autres plateformes sans aucune modification, enfin il possède une riche bibliothèque de classes comprenant des fonctions diverses telles que les fonctions standards, le système de gestion de fichiers, les fonctions multimédia et beaucoup d'autres fonctionnalités.

V.2.3 NetBeans

Pour le choix de développement, on a opté pour NetBeans car il possède de nombreux points fort qui son énorme succès dont les principaux sont :

- NetBeans IDE peut être installé sur tous les systèmes d'exploitation prenant en charge Java, Windows à linux, sur les systèmes Mac OS X.
- NetBeans est un environnement de développement intégré (IDE) open source pour le développement avec Java, PHP, C++ et d'autres langages de programmation.

V.2.4 Java Server Face

JSF est une technologie utilisée côté serveur dont le but est de faciliter le développement de l'interface utilisateur en séparant clairement la partie « interface » de la partie « métier » d'autant que la partie interface n'est souvent pas la plus compliquée mais la plus fastidieuse à réaliser.

JSF permet :

- Une séparation de la couche présentation des autres couches (MVC).
- Un mapping entre l'HTML et l'objet - un ensemble de composants riches et réutilisables.
- Une liaison simple entre les actions côté client de l'utilisateur (eventlistener) et le code Java côté serveur.
- Création de nouveaux composants graphiques.
- JSF peut être utilisé pour générer autre chose que du HTML (XUL, XML, WML, …).

Mais surtout, JSF (tout comme les autres Framework MVC) permet d'imposer des normes et une structuration dans un projet. Il est donc idéal dans le cadre d'un projet concernant plusieurs personnes afin qu'il soit cohérent.

V.2.5 PrimeFaces

Est une bibliothèque jar facile à ajouter dans un projet basé sue le Framework JSF pour obtenir un jeu de composants additionnel beaucoup avancé qui prend en compte les nouveautés des technologies du web surtout Ajax, les différents composants de PrimeFaces sont à découvrir en détail sur la page officielle de démo.

V.2.6 JDOM

Est une API open source Java son but est représenter et manipuler un document XML. Elle utilise des classes plutôt que des interfaces. Ainsi pour créer un nouvel élément, il faut simplement instancier une classe, nous avons utilisé cette API aussi pour manipuler les fichiers de description des web services.

V.2.7 Apache ODE

Est un projet Open Source écrit en Java qui implémente les spécifications WS-BPEL. Il ne fournit pas de support visuel pour la conception des processus

V.2.8 Glass Fish Serer

Glass Fish est un serveur d'application très utilisé dans le processus de développement d'applications, crée par Sun Microsystems en 2005. En effet, il permet le développement d'applications distribuées en utilisant les technologies comme : EJB, JPA, JSF et d'autres.

V.2.9 WampServer

Est une plateforme de développement Web sous Windows pour des applications Web dynamique à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MYSQL. Il possède également PHPMyAdmin pour gérer plus facilement les bases de données.

V.3 Schème globale de la plateforme

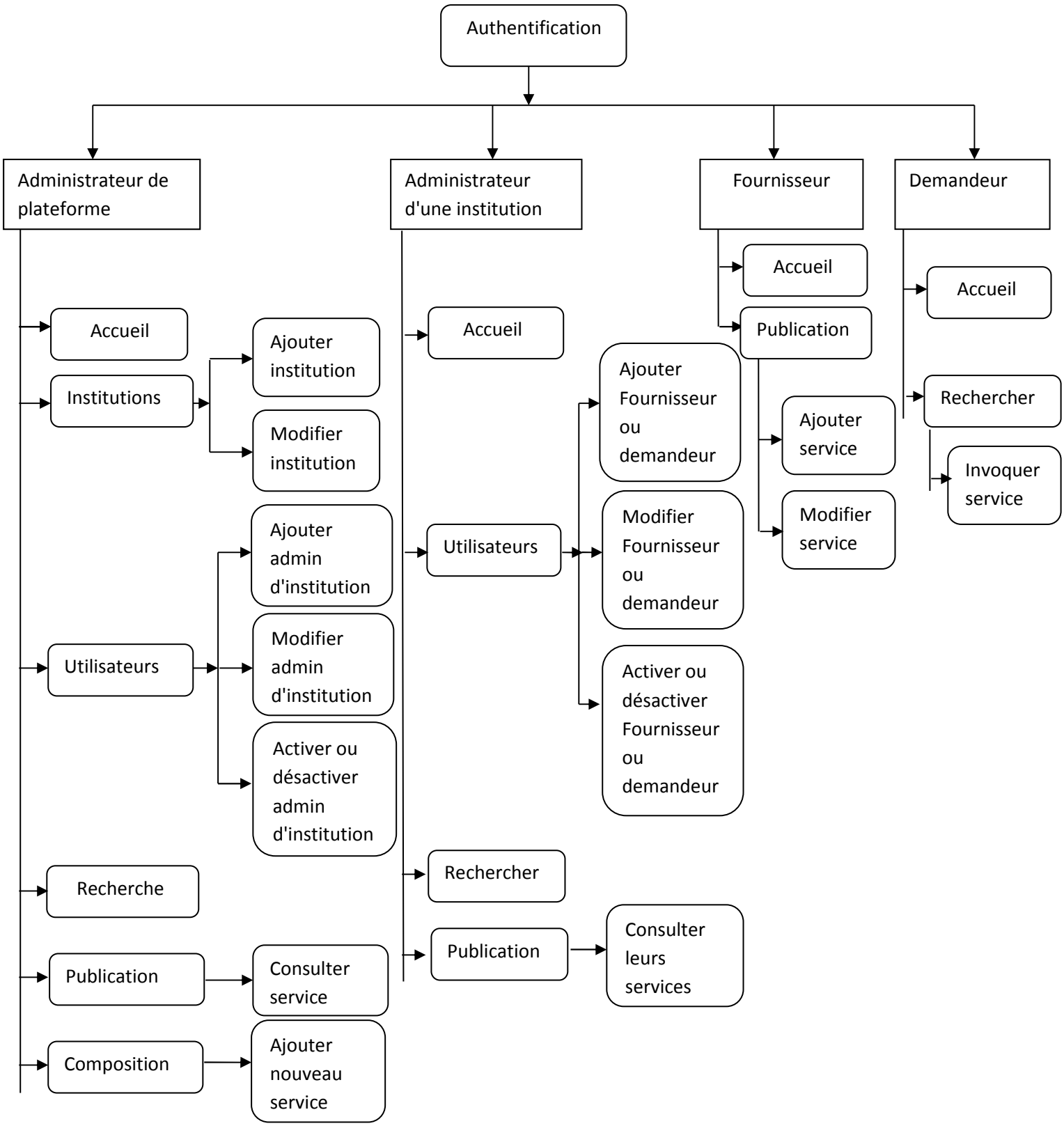


Figure V.1 : Schème globale de la plateforme.

V.4 La mise en œuvre de la plateforme

Dans ce qui suivant en illustre le fonctionnement de la plateforme pratiquement.

V.4.1 Authentification

Est une interface où on gère l'authentification des utilisateurs.la figure ci-dessous l'illustre.

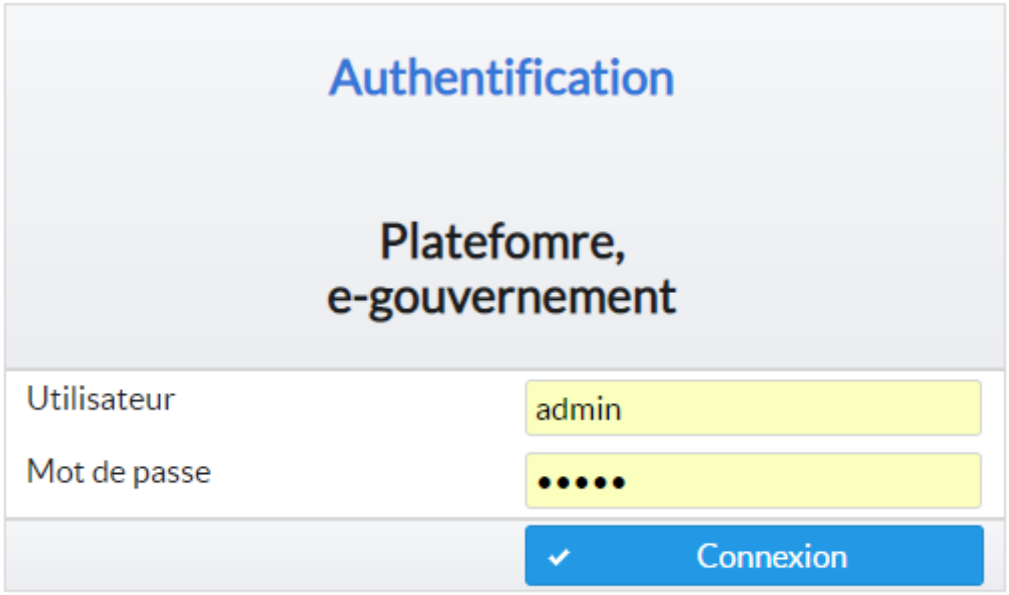


Figure V.2 : page d'authentification.

Commentaire :

- **Utilisateur :** ce champ pour saisir le nom d'utilisateur.
- **Mot de passe :** ce champ pour saisir le mot de passe d'utilisateur.

  On a 4 types d'utilisateur, selon le type d'utilisateur la plateforme s'affiche les pages autorisés.

V.4.2 Fenêtre d'accueil

Dans la suit on prend l'exemple d'authentification de l'administrateur de la plateforme :

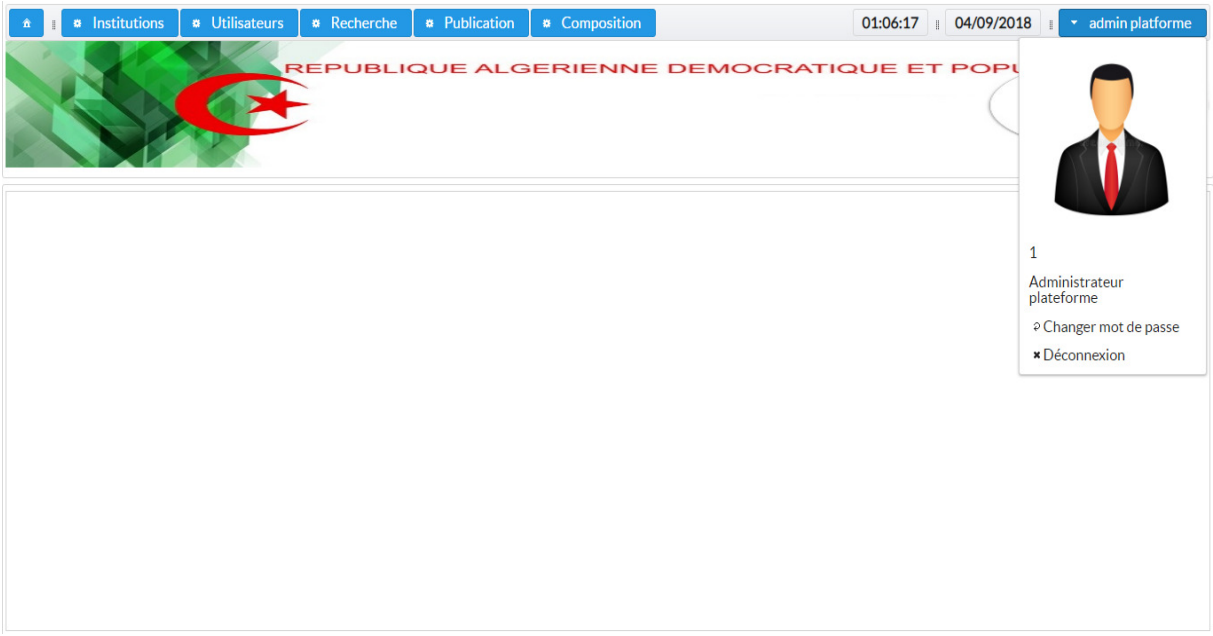
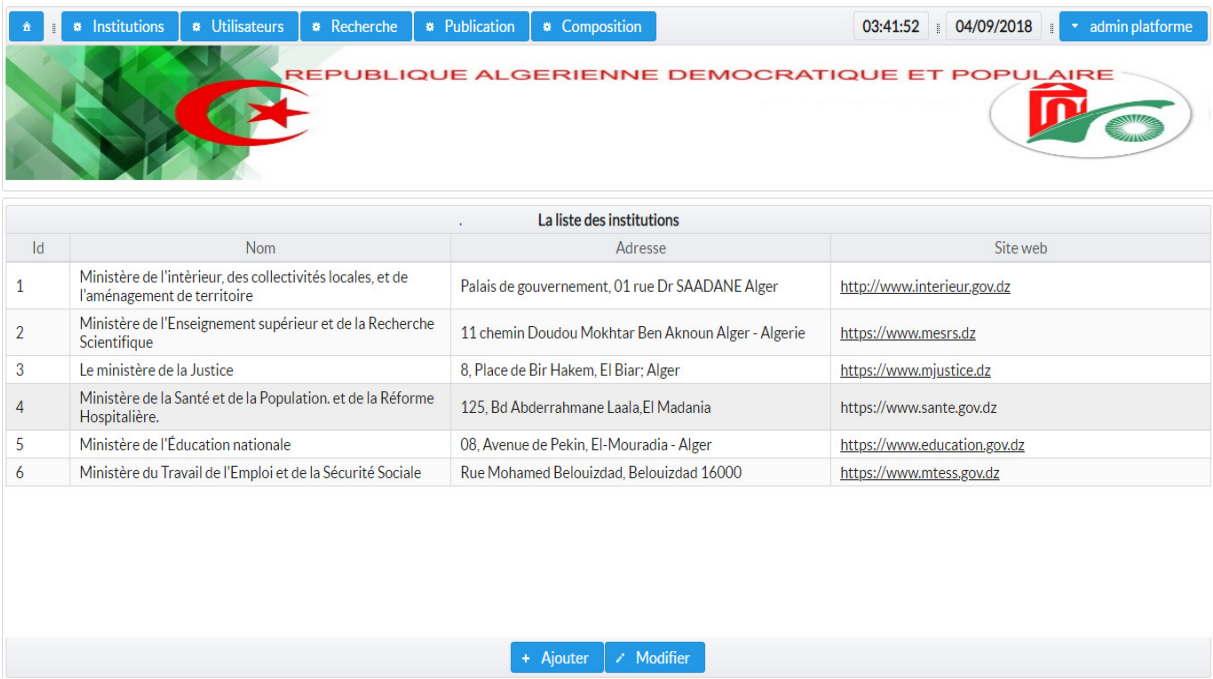


Figure V.3 : la page d'accueil.

V.4.3 Fenêtre institution

Cette interface affiche à l'utilisateur (administrateur) la liste des institutions participants à la plateforme. Et les deux boutons ajouter et modifier permet d'ajouter et de modifier une institution.



Id	Nom	Adresse	Site web
1	Ministère de l'intèrieur, des collectivités locales, et de l'aménagement de territoire	Palais de gouvernement, 01 rue Dr SAADANE Alger	http://www.interieur.gov.dz
2	Ministère de l'Enseignement supérieur et de la Recherche Scientifique	11 chemin Doudou Mokhtar Ben Aknoun Alger - Algerie	https://www.mesrs.dz
3	Le ministère de la Justice	8, Place de Bir Hakem, El Biar; Alger	https://www.mjustice.dz
4	Ministère de la Santé et de la Population. et de la Réforme Hospitalière.	125, Bd Abderrahmane Laala,El Madania	https://www.sante.gov.dz
5	Ministère de l'Éducation nationale	08, Avenue de Pekin, El-Mouradia - Alger	https://www.education.gov.dz
6	Ministère du Travail de l'Emploi et de la Sécurité Sociale	Rue Mohamed Belouizdad, Belouizdad 16000	https://www.mtess.gov.dz

Figure V.4 : la page Institutions.

V.4.4 Fenêtre utilisateurs

Cette interface affiche à l'utilisateur (administrateur) la liste des utilisateurs qui ayant un compte sur la plateforme et les opérations qui on peut effectuer avec ces utilisateurs.

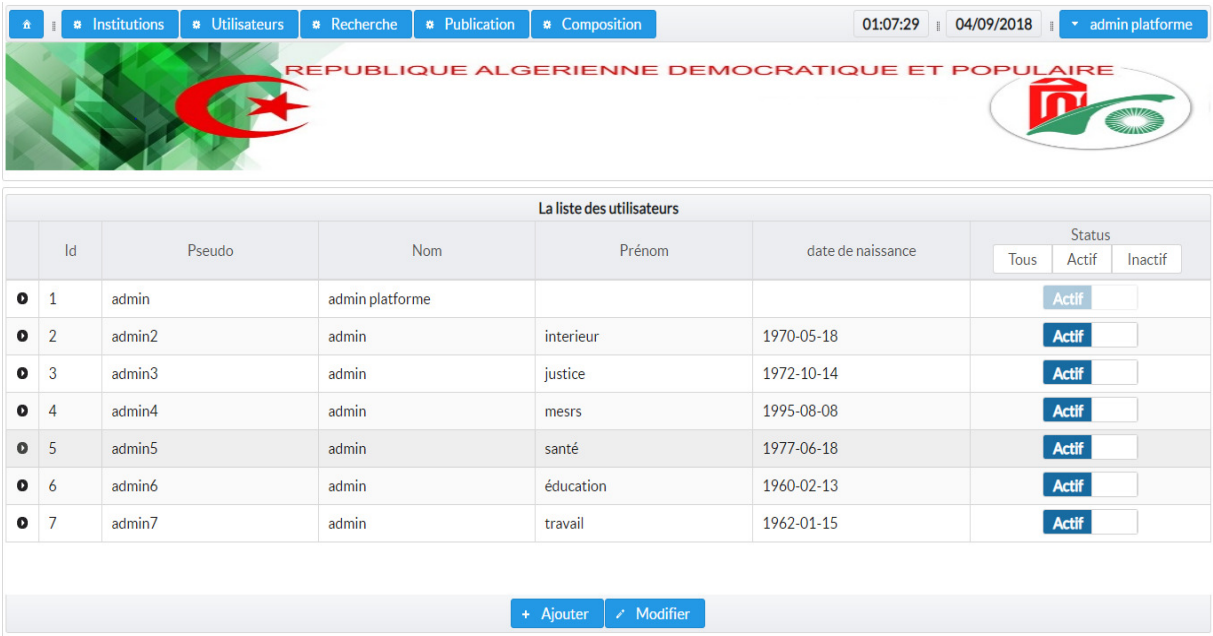


Figure V.5 : la page Utilisateurs.

V.4.5 Fenêtre recherche

Cette interface permet à l'utilisateur de faire une recherche à des services bien spécifiques soit par nom ou par description ou par date de publication et afficher sur le tableau.



Figure V.6 : la page Recherche.

V.4.6 Fenêtre publication

Cette interface affiche à l'utilisateur (fournisseur) la liste des services disponibles pour lui.et l'utilisateur peut Ajouter, Modifier ou consulter (administrateur) un service selon leurs droits.

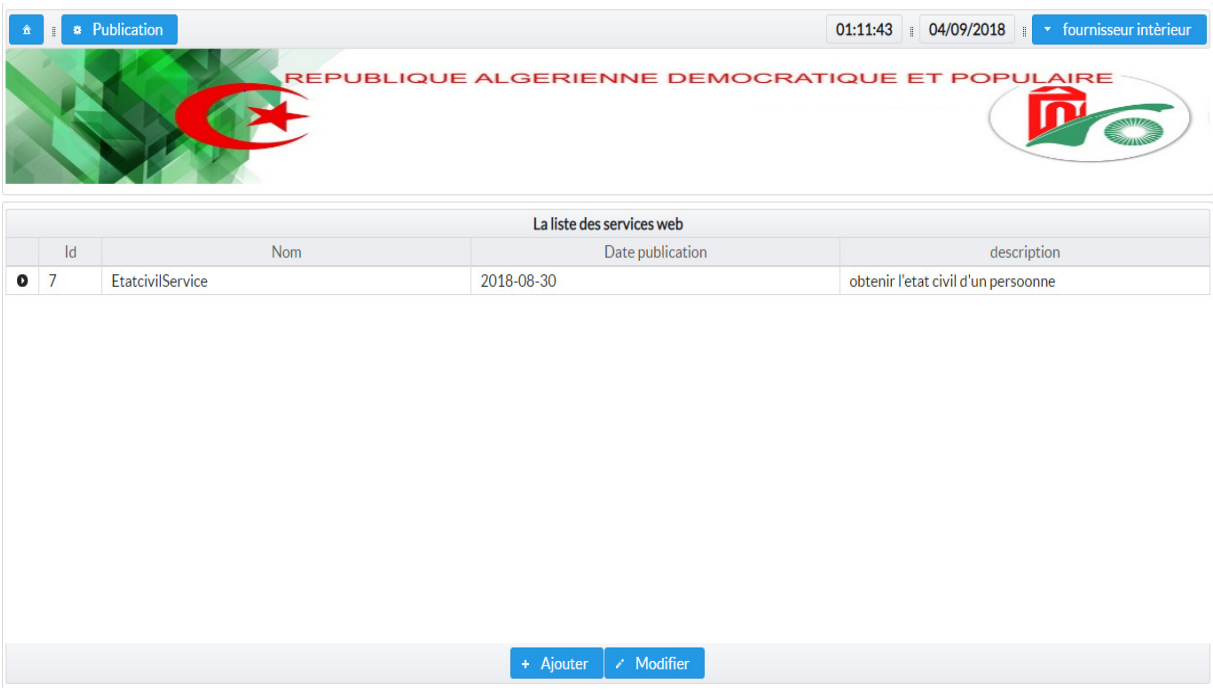


Figure V.7 : la page Publication.

V.4.7 Fenêtre composition

Cette interface offre à l'administrateur de la plateforme un outil permit de créer un service composer par le biais des services web existants (publier dans la plateforme).

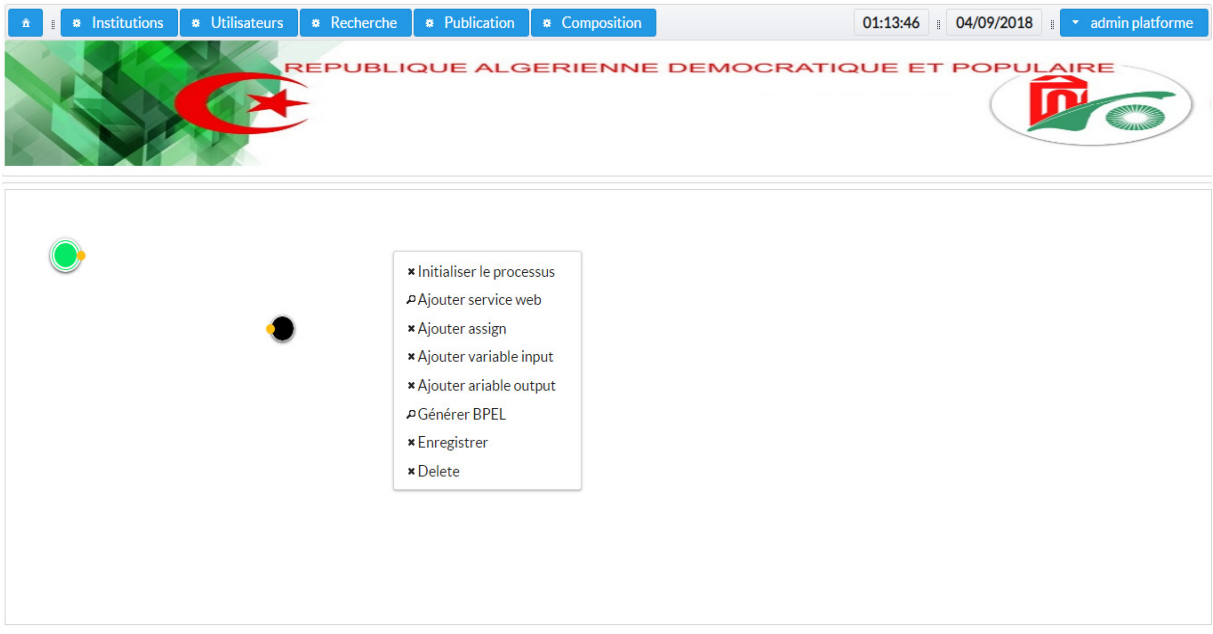


Figure V.8 : la page composition.

V.5 Etude de cas

Dans cette section nous allons présenter un scénario de composition des services web pour notre étude de cas.

V.5.1 Scenario de la composition

Afin d'illustrer la solution proposée dans ce travail, nous proposons le cas d'une institution qui veut faire un recrutement de personne, et pour trouver les compétences ou le profil requis par le poste à pouvoir. Un grand nombre d'informations sont pris en compte dans la procédure de recrutement (état civil, diplôme, casier juridique, état santé, …), on peut récupérer ces informations à partir d'autre institutions (ministère de l'intérieur, ministère de l'enseignement, ministère de la justice, ministère de santé …). Donc on a en besoin d'un intermédiaire entre l'institution qui peut faire le recrutement et les autres institutions qui récupèrent ces informations. En passant par notre plateforme, cette institution demande un nouveau service composé (recrutement). Une fois le module de gestion des utilisateurs assure que l'utilisateur de l'institution est bien authentifié et a les droits nécessaires, le module de composition fait appel au module de recherche pour récupérer les services nécessaires pour recrutement. Le résultat sera fourni par le service composite. Ce dernier va invoquer les services web (état civil, diplômes, casier judiciaire, ...) afin de fournir comme résultat les informations personnelles des candidats.

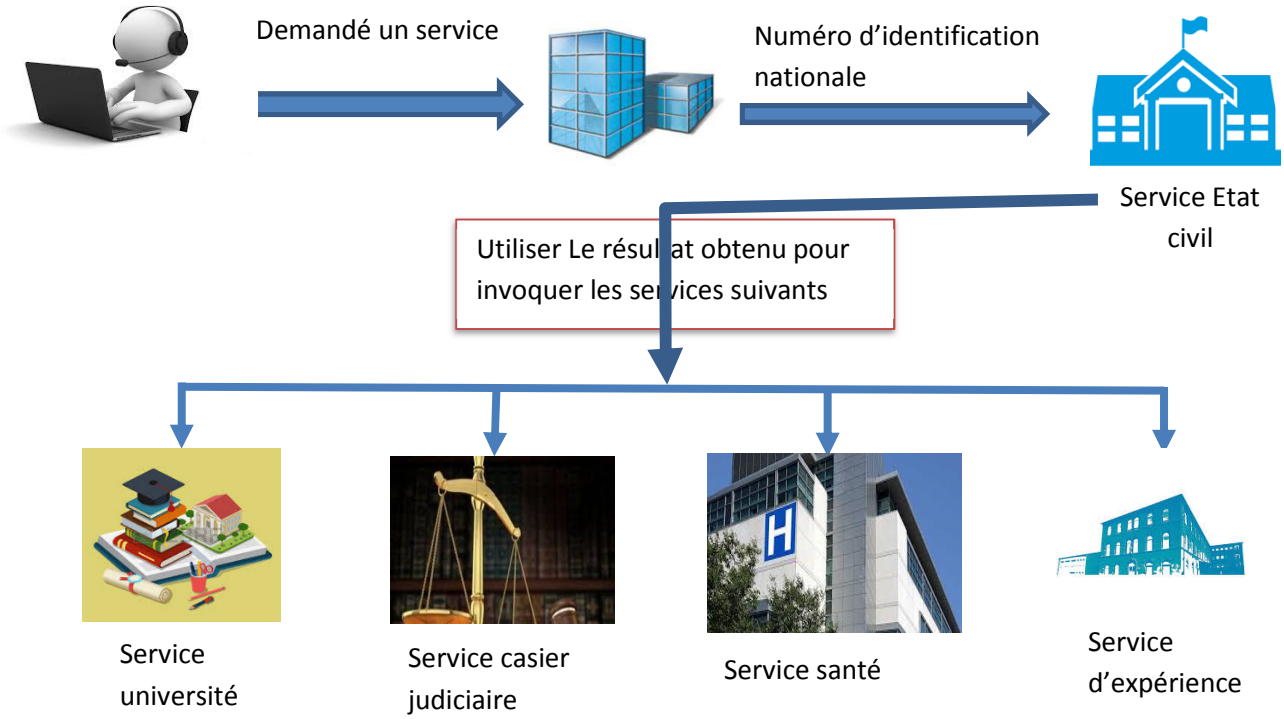


Figure V.9 : Schéma explicatif le scenario de cas d'étude.

V.5.2 Représentation de scénario à l'aide de notre outil

Nous allons faire la modélisation de l'orchestration avec notre outil pour fournir un nouveau service composite de notre scénario de recrutement représenté par la figure suivante :



Figure V.10 : modélisation du scénario de recrutement.

Les fichiers générés :

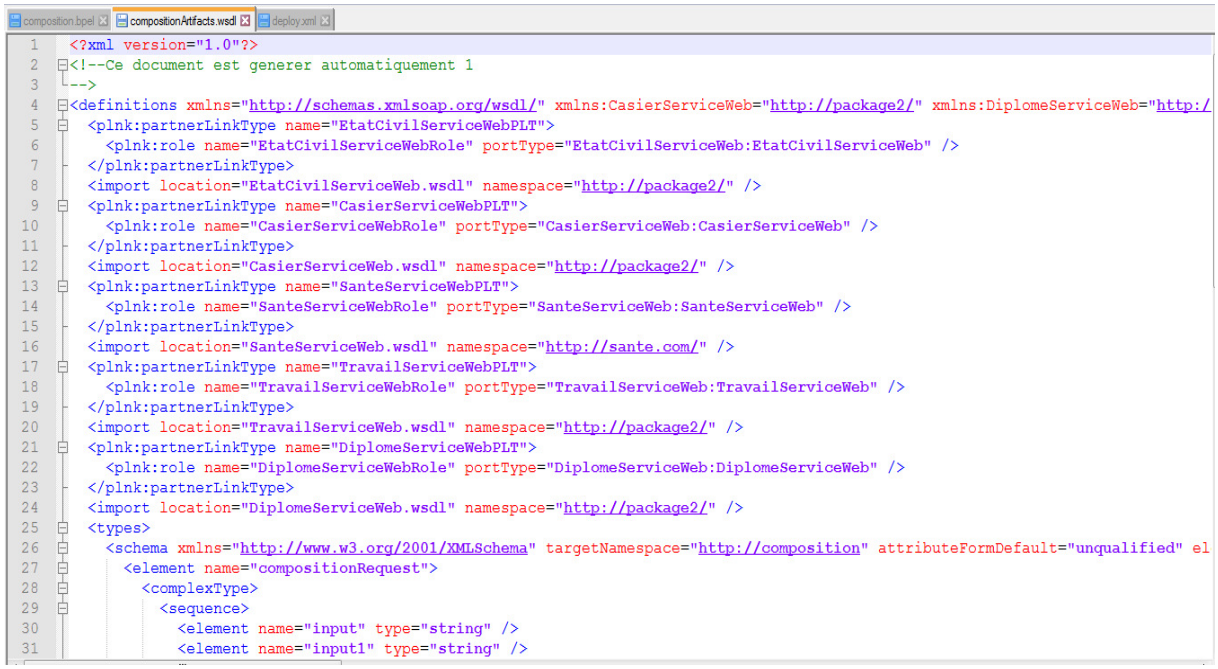
a. Un fichier « Compsition.bpel » qui décrit l'orchestration des services web invoquées.



```
composition.bpel
<!--Ce document est generer automatiquement
-->
<bpel:process xmlns:bpel="http://docs.oasis-open.org/wsbpel/2.0/process/executable" xmlns:CasierServiceWeb="http://package2/" x
  <bpel:import location="compositionArtifacts.wsdl" namespace="http://composition" importType="http://schemas.xmlsoap.org/wsdl/
  <bpel:partnerLinks>
    <bpel:partnerLink name="client" partnerLinkType="tns:composition" myRole="compositionProvider" />
    <bpel:partnerLink name="EtatCivilServiceWebPL" partnerLinkType="tns:EtatCivilServiceWebPLT" partnerRole="EtatCivilServiceWe
    <bpel:partnerLink name="CasierServiceWebPL" partnerLinkType="tns:CasierServiceWebPLT" partnerRole="CasierServiceWebRole" />
    <bpel:partnerLink name="SanteServiceWebPL" partnerLinkType="tns:SanteServiceWebPLT" partnerRole="SanteServiceWebRole" />
    <bpel:partnerLink name="TravailServiceWebPL" partnerLinkType="tns:TravailServiceWebPLT" partnerRole="TravailServiceWebRole"
    <bpel:partnerLink name="DiplomeServiceWebPL" partnerLinkType="tns:DiplomeServiceWebPLT" partnerRole="DiplomeServiceWebRole"
  </bpel:partnerLinks>
  <bpel:variables>
    <bpel:variable name="input" messageType="tns:compositionRequestMessage" />
    <bpel:variable name="output" messageType="tns:compositionResponseMessage" />
    <bpel:variable name="getEtatCivil" messageType="EtatCivilServiceWeb:getEtatCivil" />
    <bpel:variable name="getEtatCivilResponse" messageType="EtatCivilServiceWeb:getEtatCivilResponse" />
    <bpel:variable name="getCasier" messageType="CasierServiceWeb:getCasier" />
    <bpel:variable name="getCasierResponse" messageType="CasierServiceWeb:getCasierResponse" />
    <bpel:variable name="getEtatSante" messageType="SanteServiceWeb:getEtatSante" />
    <bpel:variable name="getEtatSanteResponse" messageType="SanteServiceWeb:getEtatSanteResponse" />
    <bpel:variable name="getTravail" messageType="TravailServiceWeb:getTravail" />
    <bpel:variable name="getTravailResponse" messageType="TravailServiceWeb:getTravailResponse" />
    <bpel:variable name="getDiplome" messageType="DiplomeServiceWeb:getDiplome" />
    <bpel:variable name="getDiplomeResponse" messageType="DiplomeServiceWeb:getDiplomeResponse" />
  </bpel:variables>
  <bpel:sequence name="main">
    <bpel:receive name="receiveInput" partnerLink="client" portType="tns:composition" operation="process" variable="input" crea
    <bpel:assign validate="no" name="debutToetatcivil">
      <bpel:copy>
        <bpel:from>
```

Figure V.11: fichier bpel de service recrutement.

b. Un fichier « CompositionArtifacts.wsdl » décrit les entrée, les sorties des service composé, l'url du service …etc.



```
composition.bpel  compositionArtifacts.wsdl  deploy.xml
<?xml version="1.0"?>
<!--Ce document est generer automatiquement 1
-->
<definitions xmlns="http://schemas.xmlsoap.org/wsdl/" xmlns:CasierServiceWeb="http://package2/" xmlns:DiplomeServiceWeb="http:/
  <plnk:partnerLinkType name="EtatCivilServiceWebPLT">
    <plnk:role name="EtatCivilServiceWebRole" portType="EtatCivilServiceWeb:EtatCivilServiceWeb" />
  </plnk:partnerLinkType>
  <import location="EtatCivilServiceWeb.wsdl" namespace="http://package2/" />
  <plnk:partnerLinkType name="CasierServiceWebPLT">
    <plnk:role name="CasierServiceWebRole" portType="CasierServiceWeb:CasierServiceWeb" />
  </plnk:partnerLinkType>
  <import location="CasierServiceWeb.wsdl" namespace="http://package2/" />
  <plnk:partnerLinkType name="SanteServiceWebPLT">
    <plnk:role name="SanteServiceWebRole" portType="SanteServiceWeb:SanteServiceWeb" />
  </plnk:partnerLinkType>
  <import location="SanteServiceWeb.wsdl" namespace="http://sante.com/" />
  <plnk:partnerLinkType name="TravailServiceWebPLT">
    <plnk:role name="TravailServiceWebRole" portType="TravailServiceWeb:TravailServiceWeb" />
  </plnk:partnerLinkType>
  <import location="TravailServiceWeb.wsdl" namespace="http://package2/" />
  <plnk:partnerLinkType name="DiplomeServiceWebPLT">
    <plnk:role name="DiplomeServiceWebRole" portType="DiplomeServiceWeb:DiplomeServiceWeb" />
  </plnk:partnerLinkType>
  <import location="DiplomeServiceWeb.wsdl" namespace="http://package2/" />
  <types>
    <schema xmlns="http://www.w3.org/2001/XMLSchema" targetNamespace="http://composition" attributeFormDefault="unqualified" el
      <element name="compositionRequest">
        <complexType>
          <sequence>
            <element name="input" type="string" />
            <element name="input1" type="string" />
```

Figure V.12: fichier wsdl de service recrutement.

c. Un fichier « deploy.xml ».

composition.bpel | compositionArtifacts.wsdl | deploy.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<deploy xmlns="http://www.apache.org/ode/schemas/dd/2007/03" xmlns:CasierServiceWeb="http://package2/" xmlns:DiplomeServiceWeb="h
  <process name="composition:composition">
    <active>true</active>
    <retired>false</retired>
    <process-events generate="all" />
    <provide partnerLink="client">
      <service name="composition:servicecompositon" port="EtatcivilTestPort" />
    </provide>
    <invoke partnerLink="EtatCivilServiceWebPL">
      <service name="EtatCivilServiceWeb:EtatCivilServiceWeb" port="EtatCivilServiceWebPort" />
    </invoke>
    <invoke partnerLink="CasierServiceWebPL">
      <service name="CasierServiceWeb:CasierServiceWeb" port="CasierServiceWebPort" />
    </invoke>
    <invoke partnerLink="SanteServiceWebPL">
      <service name="SanteServiceWeb:SanteServiceWeb" port="SanteServiceWebPort" />
    </invoke>
    <invoke partnerLink="TravailServiceWebPL">
      <service name="TravailServiceWeb:TravailServiceWeb" port="TravailServiceWebPort" />
    </invoke>
    <invoke partnerLink="DiplomeServiceWebPL">
      <service name="DiplomeServiceWeb:DiplomeServiceWeb" port="DiplomeServiceWebPort" />
    </invoke>
  </process>
</deploy>
```

Figure V.13 : fichier deploy.xml de service recrutement.

d. Un fichier « résultat.xml » décrit le résultat de composition de service recrutement :

```xml
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/" xmlns:q0="http://compositionService" xmlns:xsd="http://www.w3.org/2001/XMLSchema" xmln
  <soapenv:Body>
    <q0:compositionRequest>
      <q0:inputvar>1</q0:inputvar>
    </q0:compositionRequest>
  </soapenv:Body>
</soapenv:Envelope>
```

```
<soapenv:Envelope xmlns:soapenv="http://schemas.xmlsoap.org/soap/envelope/">
  <soapenv:Body>
    <compositionResponse xmlns="http://compositionService" xmlns:tns="http://compositionService" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
          <tns:outputvarsante>
      <nom_pat xmlns="">madani</nom_pat>
      <prenom_pat xmlns="">messaouda</prenom_pat>
      <date_naissance_pat xmlns="">1992-05-18T00:00:00+01:00</date_naissance_pat>
      <age xmlns="">26</age>
      <sexe xmlns="">femal</sexe>
      <adresse xmlns="">bir_ben_abed</adresse>
      <consultation xmlns="">
        <nom>medcin_1</nom>
        <prenom>MMMMM</prenom>
        <adresse>beni_slimane</adresse>
        <date_cons>2017-09-18T00:00:00+01:00</date_cons>
        <num_cons>0</num_cons>
        <nom_etab>ESSAHL_clinique_EURL</nom_etab>
        <adresse_etab>15_rue_du_20_oute_beni_slimane</adresse_etab>
        <num_telephine_etab>025515779</num_telephine_etab>
        <examen_supplimentaire>FNS</examen_supplimentaire>
      </consultation>
    </tns:outputvarsante>
          <tns:outputvarcasier>
      <nom xmlns="">madani</nom>
      <prenom xmlns="">messaouda</prenom>
      <lieuNissance xmlns="">beni_slimane</lieuNissance>
      <dateNissance xmlns="">1992-05-18T00:00:00+01:00</dateNissance>
      <adress xmlns="">bir_ben_abed</adress>
      <accusation xmlns=""/>
      <infraction xmlns=""/>
    </tns:outputvarcasier>
          <tns:outputvardiplome>
      <anneeDiplome xmlns="">2014/2015</anneeDiplome>
      <typeDiplome xmlns="">LMD</typeDiplome>
      <niveau_dep xmlns="">licence</niveau_dep>

       <nbr_annee xmlns="">3</nbr_annee>
       <nom_etudiant xmlns="">madani</nom_etudiant>
       <prenom_etudiant xmlns="">messaouda</prenom_etudiant>
       <adresse xmlns="">bir_ben_abed</adresse>
       <date_naissance xmlns="">1992-05-18T00:00:00+01:00</date_naissance>
       <lieu_naissance xmlns="">beni_slimane</lieu_naissance>
       <nom_spec xmlns="">informatique</nom_spec>
       <nom_dep xmlns="">mathematique_et_informatique</nom_dep>
       <nom_fac xmlns="">faculte_des_sciences</nom_fac>
       <nom_univ xmlns="">Université de médéa</nom_univ>
     </tns:outputvardiplome>
          <tns:outputvartravail>
      <nomP xmlns="">madani</nomP>
      <prenomP xmlns="">messaouda</prenomP>
      <dateNissance xmlns="">1992-05-18T00:00:00+01:00</dateNissance>
      <lieuNissance xmlns="">beni_slimane</lieuNissance>
      <adresse xmlns="">bir_ben_abed</adresse>
      <travail xmlns="">
        <dateDebutPer>2016-09-18T00:00:00+01:00</dateDebutPer>
        <dateFinPer>2018-05-18T00:00:00+01:00</dateFinPer>
        <nomEntreprise>entreprisexxxx</nomEntreprise>
        <activite>xxxxxxxxxx</activite>
        <nomProffition>xxxxxxxxxx</nomProffition>
        <salaireProffition>3000</salaireProffition>
        <duretravaillle>3</duretravaillle>
        <nomdeplome>licence_informatique</nomdeplome>
        <datedeplome>2015-06-18T00:00:00+01:00</datedeplome>
        <institution>univ_medea</institution>
      </travail>
     </tns:outputvartravail>
        </compositionResponse>
  </soapenv:Body>
</soapenv:Envelope>
```


Figure V.14 : fichier resultat.xml de service recrutement.

V.6 Conclusion

Ce chapitre a été consacré à la mise en œuvre et la validation de notre approche de créer une plateforme basée sur les services Web. Nous avons commencé par la représentation des outils et les langages utiliser durant le développement. Ensuite, nous avons présenté les détails des travaux expérimentaux de l'implémentation des différents modules de l'architecture ainsi que les composants qui les comportent. Enfin, nous avons exprimé aussi l'outil de la composition des services web qui nous avons développé. Cette étape a été suivie par une présentation de notre cas d'étude.

Conclusion générale

L'intégration et l'interopérabilité des services web dans un environnement de e-gouvernement (gouvernement électronique) constituent un problème clé et des exigences de base dans le développement d'applications orientées services en interaction qui doivent être résolues. Pour cette raison, notre mémoire s'inscrit dans cette problématique et avait pour objectif principal de résoudre les problèmes d'intégration et d'interopérabilité des services.

En ce qui concerne le déroulement du projet, deux étapes principales ont été nécessaires pour son développement.

La première étape de travail est consistée à créer une plateforme d'e-gouvernement. Le travail qui nous avons présenté se situe à l'intersection de quatre modules (module de gestion des utilisateurs, publication, recherche des service et module de composition), chaque module permet de donner des fonctions complémentaires à d'autres modules. La plateforme offrant un ensemble de services, tel que l'accès via le web à la consultation et récupération des informations d'une personne afin de faciliter le contacte administratif. Et faciliter la communication entre les institutions.

Dans une deuxième étape, nous avons proposé un outil de composition des services web et ces concepts relatifs, afin de l'adapter aux nouvelles fonctionnalités requises. Nous avons introduit la notion des services web composite et la façon dont nous les avons gérés. Cet outil proposé permit de générer automatiquement un fichier BPEL exécutable à partir d'une représentation graphique simple par l'administration de la plateforme.

Pour tester notre plateforme et notre outil nous avons proposé la réalisation d'une composition des services web concerne un scénario de recrutement. C'est pour ça nous avons proposé la conception et la réalisation des six services web et leurs bases de données (service d'état civil, service de santé, service de diplôme universitaire, service d'expérience professionnelle, service d'état juridique et service de scolarité).

Perspectives :

En ce qui concerne les perspectives de notre travail nous prévoyons de développer l'outil propose, et d'utiliser la composition dynamique dans cette plateforme.

Bibliographie

[1] Bouri, Sanhouidi et All. RAPPORT NATIONAL SUR LE DEVELOPPEMENT HUMAIN MADAGASCAR 2006.p 76, 81.

[2] SARI, Abdelhaq. Le rôle de la gestion des connaissances dans un projet d'e-gouvernance : cas de la stratégie e-Maroc 2010. 2006.p 9. 22. 68. 69.

[3] NDIAYE, Mamadou. E-gouvernance et démocratie en Afrique : le Sénégal dans la mondialisation des pratiques. 2006. Thèse de doctorat. Bordeaux 3.p 51.

[4] OUERGHI, Mohamed Saïd.TIC et gouvernance.2013. p 2.

[5] Gilles, St-Amant. Gouvernement en ligne cadre d'évolution de l'administration électronique. 2004.p 3. 11. 12.

[6] KETTANI, Driss et MOULIN, Bernard. L'e-gouvernement pour la bonne gouvernance dans les pays en développement : l'expérience du Projet eFez L'e-gouvernement pour la bonne gouvernance dans les pays en développement : l'expérience du Projet eFez. P 59.

[7] 13ème Réunion du Comité Intergouvernemental d'Experts (CIE) Mahe, Seychelles, 27-29 avril 2009. P 8. 9. 11. 12.

[8] TOUITOU, Mohammed, REKIMA, Faris, et BOUDEGHDEGH, Ahmed. Les Impacts Économiques et Sociaux de l'E-gouvernement.2017. p 319.

[9] TAZDAIT, Fatiha. L'e-gouvernance et collectivités locales : Etat des lieux dans la Daïra de Sidi-Aïch .2016. UNIVERSITE ABDERRAHMANE MIRA DE BEJAIA. p 37. 38.

[10] Aissata, SOW THIAM. E-GOUVERNANCE LES RELATIONS ETAT-CITOYENS A L'HEURE DU NUMERIQUE Panorama, Enjeux et Perspectives en Afrique. 2009.p 10

[11] AMADOU, Diabagate. E-Gouvernement : Modélisation d'un système d'information décisionnel intégrant les concepts de l'intelligence artificielle, de Big Data et de data science pour une gouvernance efficace des marchés publics. Thèse de doctorat. Université Abdelmalek Essaadi. 2016.p36,39.

[12] Gregory, Pokorny. E-Gouvernement de la Corée (Cas de Succès : Meilleurs pratiques). 2009.

[13] MEGDER, El, CHERKAOUI, C., SBIHI, B., et al. Le e-Gouvernement et la Modernisation du Secteur Public. In: 3rd International Conference: Sciences of Electronic Technologies of Information and Telecommunications. 2005.p 2.

[14] Georges, Labaki. Le Gouvernement Electronique : Visions et stratégies pour le cas libanais. 2011.

[15] BACHTARZI, Fayçal. Une Approche de Composition des Services Web Basée Transformation de graphes. 2014. Thèse de doctorat. Université Abdelhamid Mehri Constantine 2. p 8 .

[16] BOUCHEKARA, Med Kacem, ZIANI, Soheyb. La Découverte de services web à base De Dominance Probabiliste. 2015.Université Abou-bekr Belkaid Tlemcen. p 4,14.

[17] DAHA, Hanane, LIFA, Siham**.** Une approche formelle pour la composition des services web.2015. UNIVERSITÉ ECHAHID HAMMA LAKHDAR - EL-OUED. p 5,12,15.

[18] BEKKOUCHE, Amina. Composition des Services Web Sémantiques à base d'Algorithmes Génétiques.2012. Université Abou-bekr Belkaid Tlemcen. p 26, 27, 28, 35, 38.

[19] FDHILA, Walid. Décentralisation optimisée et synchronisation des procédés métiers inter-organisationnels. 2011. Thèse de doctorat. Université Henri Poincaré-Nancy I. p 26.

[20] KELLERT, Patrick et TOUMANI, Farouk. Les web services sémantiques. Web sémantique, Action spéci_que, 2003, vol. 32.p 2.

[21] CHEMAA, Sofiane. Une approche de composition de services Web à l'aide des Réseaux de Pétri orientés objet.2014. Thèse de doctorat. Abdelhamid Mehri Constantine 2. p 8,9,12,13 et 15.

[22] BENDADDA, Zeyneb Fatima Zahra, MOUSSOUS, Islam **.** Découverte hybride des services web à base Condorcet flou.2015. Université Abou Bakr Belkaid– Tlemcen. p 14,17.

[23] DAOUD Zahèra, MAALACHE, Nassima. Composition des services web sémantiques à base d'algorithmes Génétiques. 2013. Université Abou Bakr Belkaid– Tlemcen. p 11,14.

[24] BAKHTA, AMRANE. MÉTHODE DE RECHERCHE DE SERVICES WEB BASÉE SUR L'ANALYSE FORMELLE DE CONCEPTS. 2015. Thèse de doctorat. Université de La Manouba. p 12.

[25] Belkhiter, Khalid. Sélection des services Web à base d'algorithmes génétiques multi-objectifs.2012. Université Abou Bakr Belkaid– Tlemcen. p 19.

[26] ALKAMARI, Aniss. Composition de services web par appariement de signatures.2008. UNIVERSITÉ DU QUÉBEC À MONTRÉAL. p 37.

[27] MEFTAH, Mohammed Charaf Eddine. Une approche formelle pour les applications web 2.0. 2016. Thèse de doctorat. Université Mohamed Khider-Biskra. p 34.

[28] SALATGÉ, Nicolas. Conception et mise en œuvre d'une plate-forme pour la sûreté de fonctionnement des services Web. 2006. Thèse de doctorat. Institut National Polytechnique de Toulouse. p 31.

[29] BOULAHIA, NADIA. Conception et réalisation d'une application à base des services web d'une agence de voyage en ligne.2017. Université Abderrahmane Mira de Béjaia. p 21.

[30] Remadelia, Abdellah. Matallah, Imane. Composition des Services Web. 2016. Université de Djilali BOUNAÂMA Khemis Miliana. p 22,23.

[31] Benmedakhene, Boubekeur, Yahyaoui , Imed eddine. Composition des services Web : Etat de l'art.2016. Université Larbi Tébessi - Tébessa. p 16.

[32] JAMAL, Sonia. Environnement de procédé extensible pour l'orchestration-Application aux services web. 2005. Thèse de doctorat. Université Joseph-Fourier-Grenoble I. p 59.

[33] RABHI, Issam. Testabilité des services Web. 2012. Thèse de doctorat. Université Blaise Pascal-Clermont-Ferrand II. p 27.

[34] KARTIT, Zaid. Contribution à la sécurité du Cloud Computing : Application des algorithmes de chiffrement pour sécuriser les données dans le Cloud Storage. 2016.p24

[35] Elio, Goettelmann. Modélisation de processus métiers sensibilisés aux risques et déploiement en confiance dans le Cloud. 2015. Université de Lorraine. P 5.

[36] HAMZE, Mohamad. Autonomie, sécurité et QoS de bout en bout dans un environnement de Cloud Computing. 2015. Thèse de doctorat. Université de Bourgogne. P 6

[37] AOUAMEUR Zakia, TAHRINE Halima. Comparaison et mise en place des plateformes de Cloud Computing : OpenStack et Eucalyptus. 2013. Université Kasdi Merbah de Ouargla. P 1

[38] GUETMI, Nadir. Modèles de conception pour des applications collaboratives dans le Cloud. 2016. Thèse de doctorat. Chasseneuil-du-Poitou, Ecole nationale supérieure de mécanique et d'aérotechnique. P 14.15.

[39] FILALI, Fatima Zohra. Gestion de Confiance dans le cloud Computing. 2016. Université d'oran. P 13

[40] KHALIFA, MAIOUA et ADEL, MANSOURI. Approche basée Agents Mobiles intelligents dans un environnement de Cloud Computing. MASTER ACADEMIQUE. UNIVERSITE KASDI MERBAH OUARGLA. p 2. 22.

[41] PASTOR, Jonathan. Contributions à la mise en place d'une infrastructure de Cloud Computing à large échelle. 2016. Thèse de doctorat. Ecole des Mines de Nantes. P 28

[42] SAOULI, Hamza. Découverte de services web via le cloud computing à base d'agent mobile. 2015. Thèse de doctorat. Université Mohamed Khider Biskra. P 16. 17. 32. 33.

[43] BENKOUIDER SAHRAOUI SOUFYANE, Soufyane, MANSOURI, Yasmine, TOUAZI, Djoudi, et al. Etude de sécurité d'une extension Cloud. 2017. Thèse de doctorat. Universite de bejaia. P 11.

[44] GELIBERT, David. SMILI, Farid. La sécurité et la virtualisation. 2012. P 12

[45] PROBST, Thibaut. Évaluation et analyse des mécanismes de sécurité des réseaux dans les infrastructures virtuelles de cloud computing. 2015. Thèse de doctorat. INP Toulouse. P 8

[46] JUGANARU-MATHIEU, Mihaela. Cloud Computing. 2015. Ecole Nationale Supérieure des Mines de St Etienne. P 38

[47] BEN HMIDA, Slah. Etude, Conception et implémentation d'une architecture réseau sécurisée et un cloud privé. 2016. Thèse de doctorat. Université Virtuelle de Tunis. P 12

[48] FOULON, Denis. Etude et mise en place d'une plate-forme de Cloud computing privé avec vCloud Director 5.1.2, 2015. CONSERVATOIRE NATIONAL DES ARTS ET METIERS, CENTRE REGIONAL DE LYON. P 6

[49] AKBI, khalil. ZEHRI, mohammed. Etude et mise en place d'une solution Cloud computing privé au sein de l'université de ouargla.2013. UNIVERSITE KASDI MERBAH OUARGLA. P 9

[50] Kodzo Mawuessénam, PARKOO. La qualité de service au niveau IaaS : Vers l'utilisation du concept Software- Defined Networking (SDN).2015**.** Université Amadou Hampâté-Bâ de Dakar.

[51] BERKANI, Nassima. MOUSSAOUI, Salima. La sécurité des données dans le Cloud Computing.2016. Université de Béjaïa. P 19

ANNEXE

Les diagrammes des bases de données

Diagramme de diplôme _universitaire :

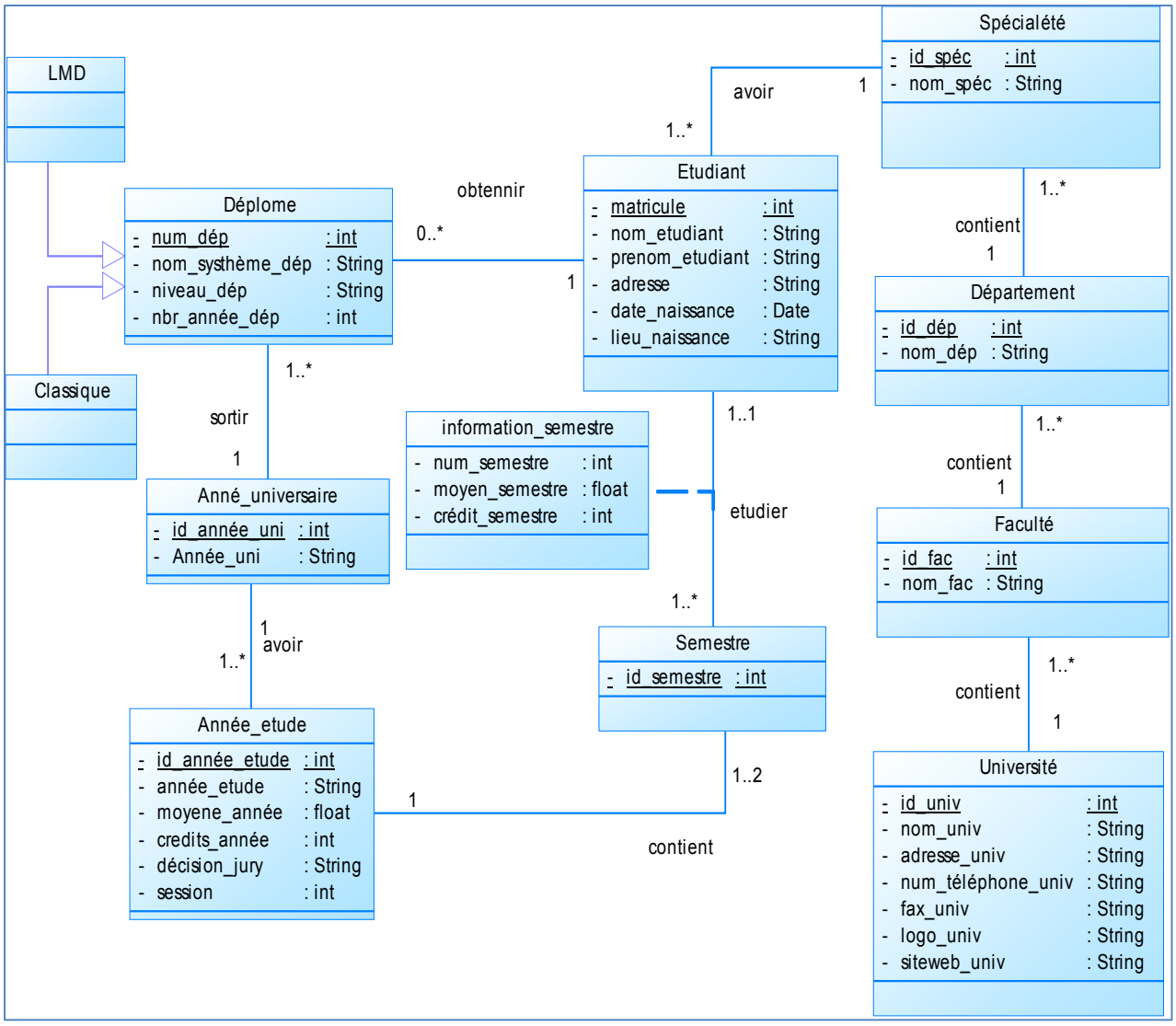


Diagramme de_sonté :

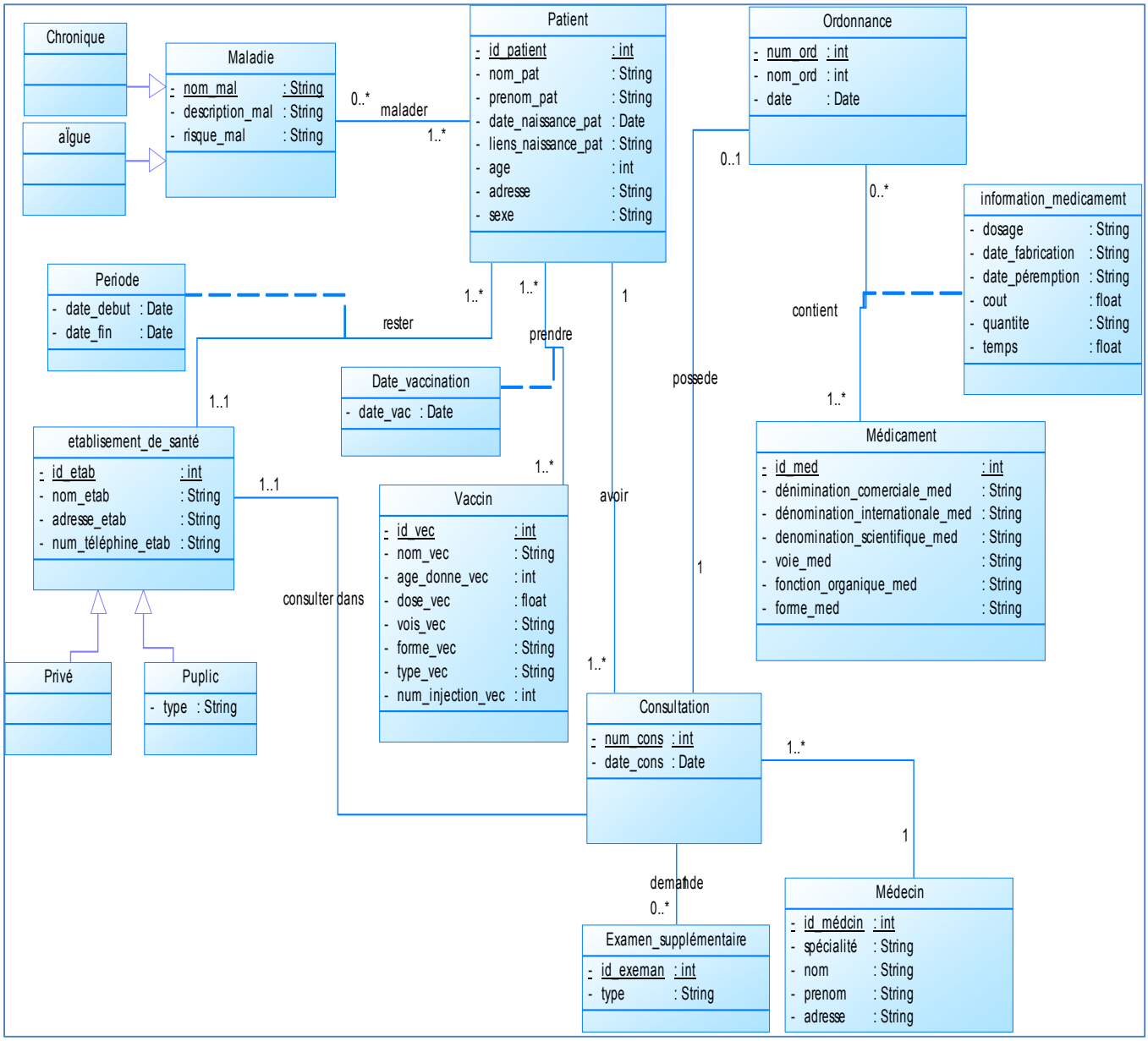


Diagramme de état civile :

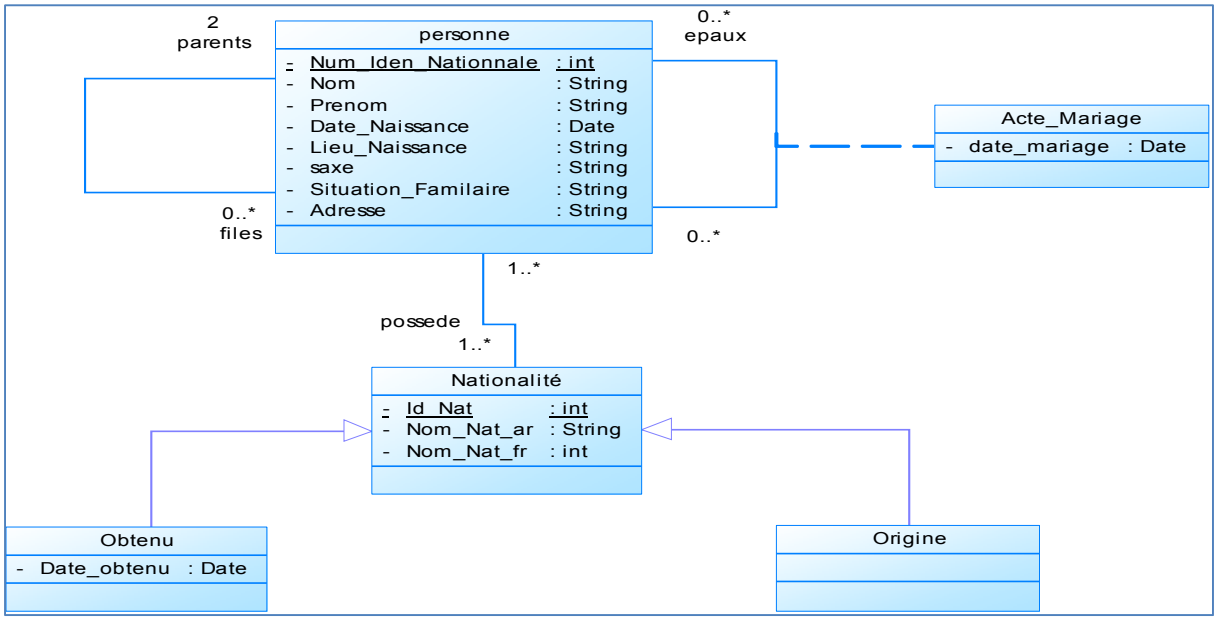


Diagramme scolaire :

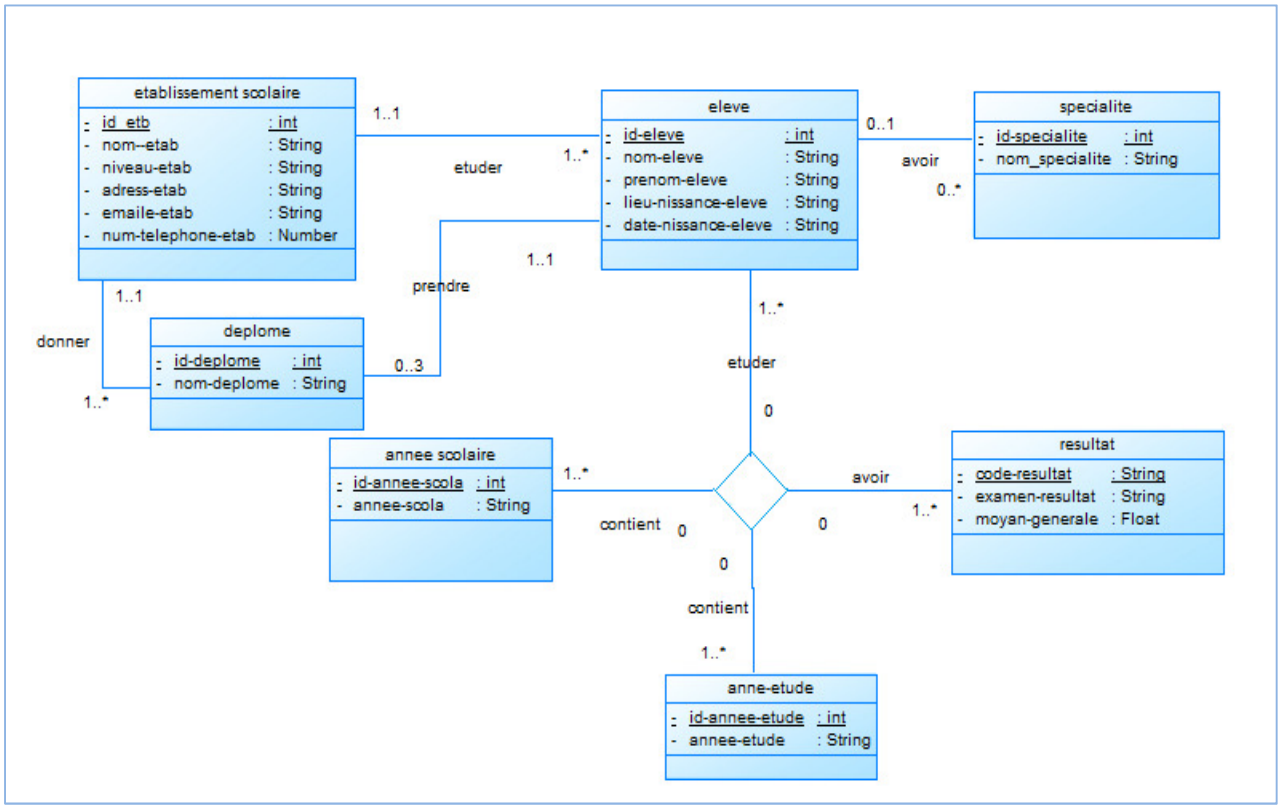


Diagramme de casier judiciaire :

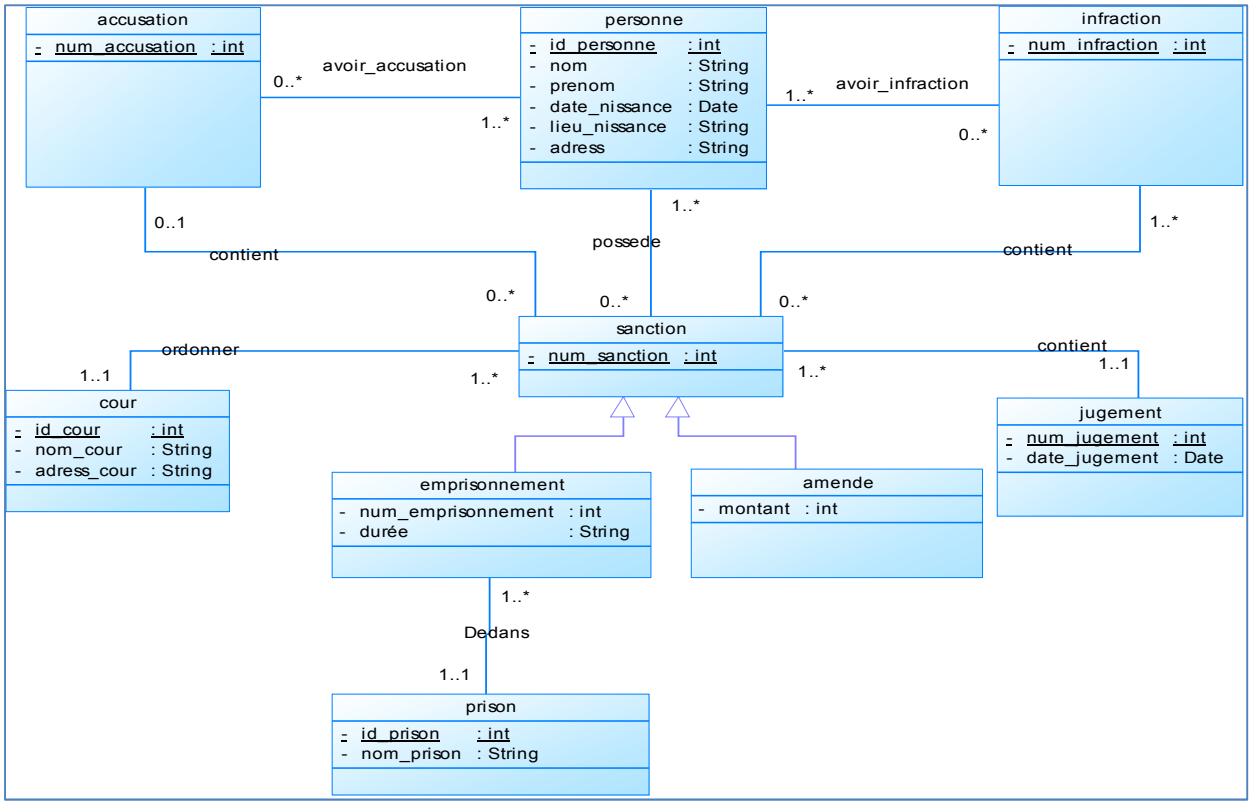
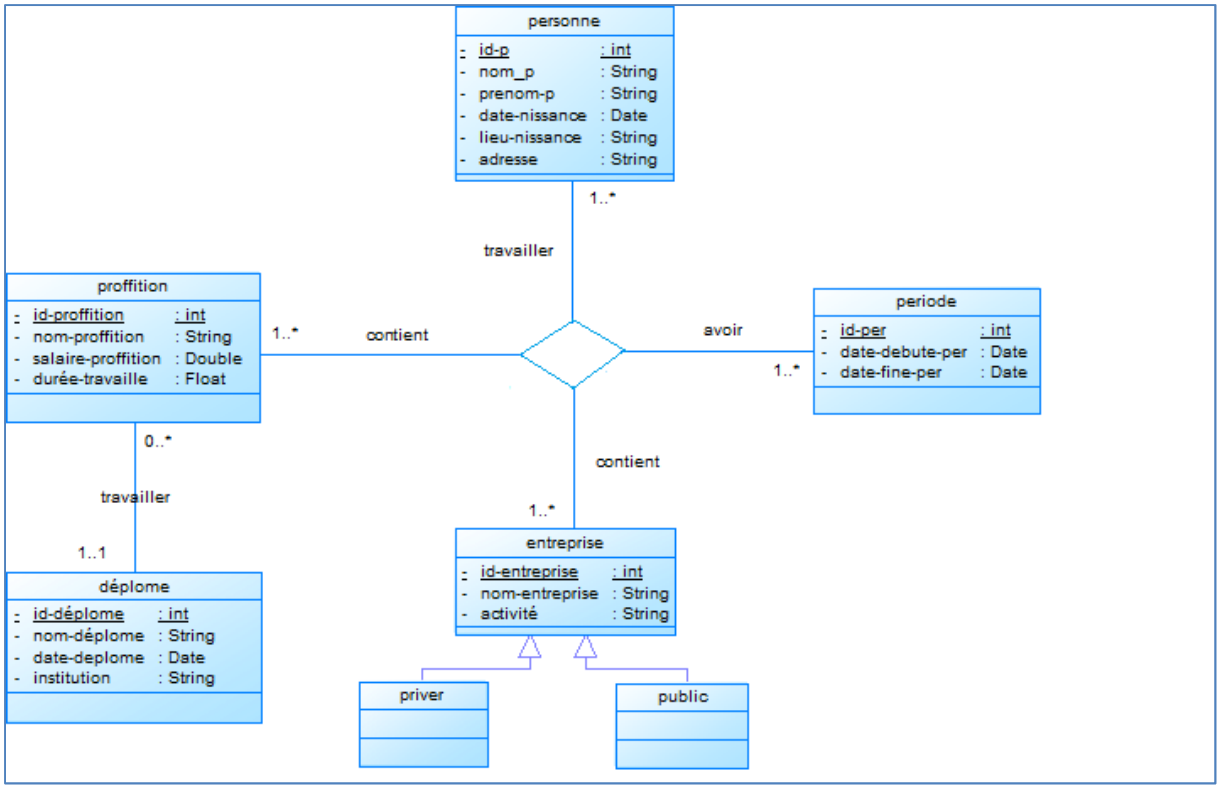


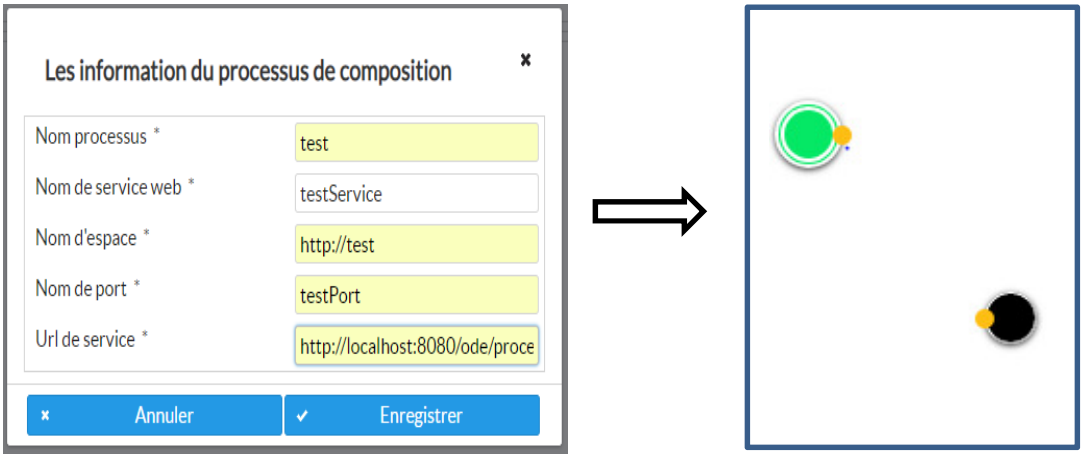
Diagramme de travaille :



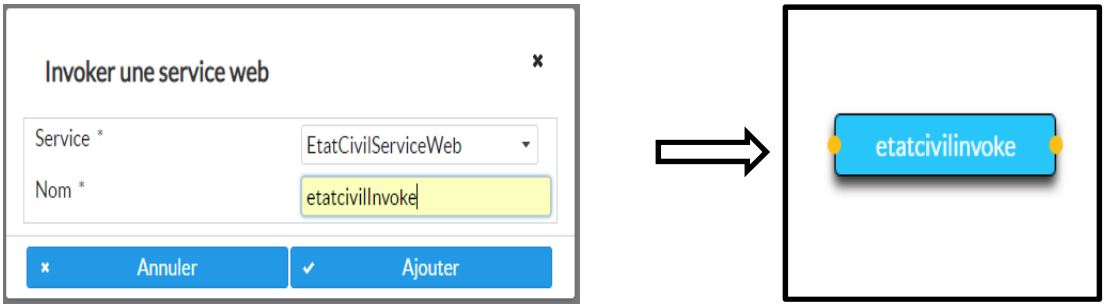
Fonctionnement notre outil

Dans la suit nous avons décrit la fonctionnement notre outil :

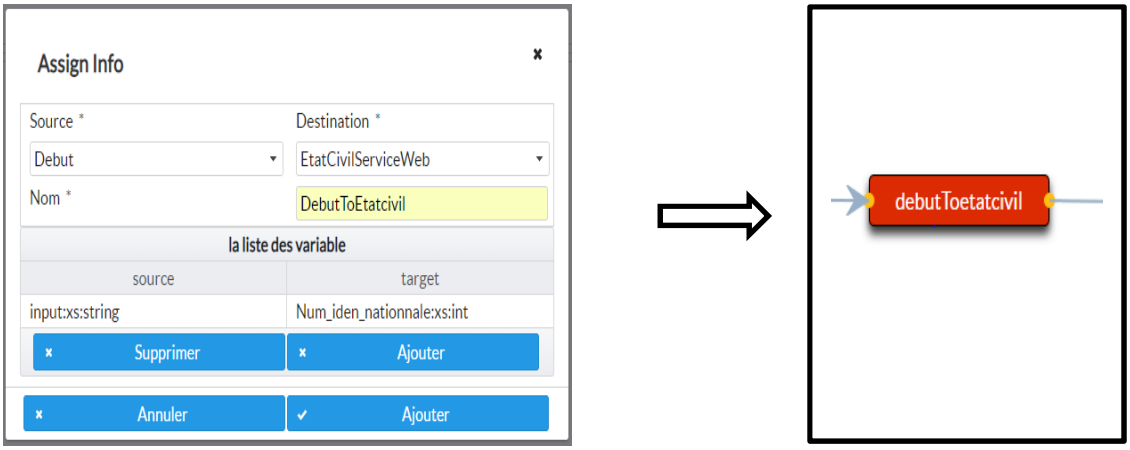
1- Créer un processus Bpel et saisir ses informations (un élément receive et reply sera créer automatiquement aprè l'ajoute d'un nouveau processus) :



2- Invoquer les service web existants dans la plateforme à l'aide du **Button « ajouter service web »** dans le menu contextuel, après l'ajoute de service web on récupère à partir du fichier WSDL les informations de ce service telle que les entrées, les sorties, nom de service, lien d'accès …etc. (on peut invoquer plusieurs service web en séquence et en parallèle dans la même processus BPEL) :



3- **Le Button « ajouter assign »** dans le menu contextuel permet d'ajouter un élément **assign** pour passer des paramètres au service web ou récupérer le résultat à partir de ce service.



4- **Le Button « générer BPEL »** permet de générer automatiquement un code BPEL exécutable à partir de diagramme qu'on a dessiner et le télécharger sous format d'un fichier zip, ce dernier contient plusieurs fichiers :

a. Les fichiers WSDL et XSD des services web invoquées.
b. Un fichier « nomService.bpel » qui décrit l'orchestration des services web invoquées.
c. Un fichier « nomServiceArtifacts.wsdl » décrit les entrée, les sorties des service composé, l'url du service …etc.
d. Un fichier « deploy.xml ».

