



جامعة العربي التبسي - تبسة
Université Larbi Tébessi - Tébessa

République Algérienne Démocratique et Populaire
Ministère de l'enseignement supérieur et de la
recherche scientifique

Université Larbi Tébessi - Tébessa

Faculté des Sciences Exactes et des Sciences de la Nature et de la Vie

Département : Mathématiques et Informatique



كلية العلوم الدقيقة وعلوم الطبيعة و الحياة
FACULTÉ DES SCIENCES EXACTES
ET DES SCIENCES DE LA NATURE ET DE LA VIE

Mémoire de fin d'étude

Pour l'obtention du diplôme de MASTER

Domaine : Mathématiques et Informatique

Filière : Informatique

Option : Systèmes d'information

Thème

**Tout en tant que ressource (*aaS) : Fondement et
illustration à travers l'internet des objets**

Présenté Par :

Fella trabelsi

Devant le jury :

Mr M. R. Laouar

Pr. Université Larbi Tébessa Président

Mr M. Y. Haouam

MCB Université Larbi Tébessa Examineur

Mr A. Metrouh

MCB Université Larbi Tébessa Encadreur

Date de soutenance : 22/06/2019

Dédicace

Je dédie ce travail

À mon très cher père, qui est toujours à mon côté à tout moment.

À ma très cher mère, que dieu ait pitié d'elle.

À ma famille.

À mes amis.

Felle trabelsi

Remerciement

Je remercie Dieu le tout-puissant qui m'a béni et m'a permis d'arriver là où je suis.

Je tiens à exprimer mon remerciement envers toutes les personnes qui ont contribué au bon déroulement et à l'aboutissement de ce mémoire.

Je tiens particulièrement à remercier mon encadreur **Dr. Abdelmalek Metrouh** pour mon avoir encadré et orienté durant mon travail.

Je remercie également :

Président de jury : **Pr. M R Laouar.**

Examineur : **Dr. M Y Haouam.**

D'avoir accepté d'examiner et d'évaluer mon travail.

Enfin, je tien à remercier mes proches et mes amis(e)s pour les moments partagés et les bonheurs, ainsi que ma famille.

Résumé

Nous présentons dans ce mémoire, tout en tant que ressource (* aaR) comme un paradigme pour la conception d'applications collaboratives sur le Web. Les ressources sont classées en ressources informatiques (traiter les données), consommées (capturer des données) et produites (extraire des données). Les ressources exposent des méthodes. L'appel approprié des méthodes est assuré par des restrictions. Les restrictions sur les ressources et les spécialisations en limité (plutôt qu'illimité), non partageable (ou partageable) et non renouvelable (ou renouvelable). Les restrictions permettent de différencier davantage les ressources des services et de mieux les contrôler.

Ce mémoire illustre le paradigme * aaR à travers une étude de cas dans le domaine sportif (natation).

Les mots clés : internet des objets, ressources, natation, Web services, RESTful .

Abstract

We present in this memoir, everything as a resource (*aaR) as a paradigm for the design of collaborative web applications. Resources are classified as computer resources (data processing), consumed (data capture) and produced (data extract). Resources expose methods, appropriate call of methods is ensured by restrictions. Restrictions on resources and specializations that are limited (rather than unlimited), not shareable (or shareable) and non-renewable (or renewable). Restrictions allow for greater differentiation of service resources and better control.

This memoir illustrates the * aaR paradigm with a case study in the field of sport.

Keywords: internet of things, resources, swimming, Web services, RESTful.

ملخص

نقدم في هذه المذكرة ، كل شيء كمورد (*aaR) كنموذج لتصميم التطبيقات التعاونية على الويب. يتم تصنيف الموارد في : موارد الحوسبة (معالجة البيانات) ، والموارد المستهلكة (التقاط البيانات) والمنتجة (المستخرجة من البيانات) ، ودورات حياة المورد يمكن تتبعها وفقا لفتتها. الموارد تعرض الأساليب ، ويضمن الاستدعاء المناسب للأساليب من قبل القيود. القيود المفروضة على الموارد والمتخصصين محدودة (وليست غير محدودة) وغير قابلة للمشاركة (أو قابلة للمشاركة) وغير قابلة للتجديد (أو قابلة للتجديد). تساعد القيود في زيادة التمييز بين الموارد والخدمات وتحسين التحكم.

هذه المذكرة توضح نموذج *aaR مع دراسة حالة لاستخدام إنترنت الأشياء (IoT) في مجال الرياضة .

الكلمات المفتاحية: إنترنت الأشياء ، موارد ، سباحة ، خدمات الويب , RESTful

Table de matière

Dédicace	1
Remerciement	2
Résumé.....	3
abstract.....	3
الملخص.....	4
Introduction générale	
Introduction générale	13
Chapitre 1 : Internet Des Objet « IoT »	
1. Introduction.....	15
2. concepts fondamentaux.....	16
2.1 Internet des objets ou « Internet Of Things (IOT) ».....	16
2.2.1 Définition	16
2.2 Ressource	17
2.2.1 Définition.....	17
3. Architecture de l'IoT	17
3.1. Architectures à trois et cinq couches	17
3.2. Architectures basées sur le Cloud computing	20
3.3. IoT social.....	21
3.3.1. Composants de base	21
3.3.2. Architecture représentative.....	22
4. Les domaines d'application d'IoT	23
4.1 Applications Internet des objets (IoT) pour la maison connectée /intelligente « Connected/Smart home ».....	23
4.2. Applications Internet des objets (IoT) pour les vêtements « Wearable ».....	27
4.3 Applications Internet des objets (IoT) dans le commerce de détail « Retail »	29
4.3.1 Modèle de consommation Retail 4.0 IoT	30
4.3.2 Modèle de détaillant Retail 4.0 IoT	30
4.4 Applications Internet des objets (IoT) pour les villes intelligentes « Smart Cities »	31
4.5 Applications de l'Internet des objets (IoT) dans les soins de santé « Healthcare »	33
4.5.1 Soins cliniques	33
4.5.2 Surveillance à distance	34
4.6 Applications de l'Internet des objets (IoT) en agriculture	35

4.7 Applications de l'Internet des objets (IoT) dans l'automobile / les transports « Automotive /Transportation »	36
4.7.1 Voiture autonome / autonome / sans conducteur « An autonomous car/ self-driving car/ driverless car » :	37
4.7.2 Voitures électriques « Electric cars ».....	37
4.8 Applications Internet des objets (IoT) pour l'automatisation industrielle.....	38
4.9 Gestion de l'énergie des applications Internet des objets (IoT) « Energy Management ».....	39
4.10 Application Internet des objets(IoT) dans le domaine de sport.....	41
Conclusion.....	43
Chapitre 02 : services Web	
1. Introduction.....	45
2. Services web.....	46
2.1 Définition	46
2.2 Architecture de service web (SOA)	47
2.2.1 Définition Métier	47
2.2.2 Définition Technique.....	47
2.3 Le modèle de services Web	49
2.3.1 Les Rôles dans une architecture de services Web.....	49
2.3.2 Opérations dans une architecture de services Web.....	50
2.3.3 Artéfacts d'un service Web	50
2.4 Les types de services Web	51
2.4.1 Basé sur SOAP(SOAP_based)	51
2.4.2 Basé sur REST(REST_based)	53
2.5 Pile des services web	55
2.6 Les technologies des services web	56
2.6.1 Protocole d'accès aux objets simple (SOAP)	56
2.6.1.1 Structure du Message SOAP	57
2.6.2 Langage de description des services Web (WSDL)	58
2.6.2.1 Structure d'un document WSDL	59
2.6.2.2 Les éléments principaux d'un document WSDL.....	59
2.6.3 Description universelle, protocole de découverte et d'intégration (UDDI) ..	60
2.7 Caractéristiques des services Web	61
2.7.1 Un service Web est une application logicielle qui est reconnue par un URI.....	61
2.7.2 Capacité des interfaces et liaisons d'être publiées, localisées et invoquées via le langage XML	62

2.7.3	Capacité d'interagir avec les composantes des logiciels via des éléments XML avec l'utilisation des protocoles Internet standards	62
2.7.4	Composante logicielle légèrement couplée à interaction dynamique	62
Conclusion.....		63
Chapitre 03 : Présentation du nouveau paradigme "Tout en tant que ressource" et illustration à travers une étude de cas		
1.	Introduction.....	65
La 1ère partie		66
2.	Internet des objets (tout en tant que ressource).....	66
2.1	Présentation du « Framework * aaR».....	66
2.1.1	La couche application	67
2.1.2	La couche ressources	67
2.1.3	La couche infrastructure	68
2.1.4	Les interactions entre couches.....	68
3.	Illustration de l'internet des objets, tout en tant que ressource.....	69
3.1	Les catégories des ressources	69
3.1.1	Ressources informatiques « computational ».....	70
3.1.2	Ressources consommées « consumed »	70
3.1.3	Ressources produites « produced »	71
3.2	Descripteur de ressources	71
3.2.1	Les méthodes utilise pour les ressources produites /consommées / informatiques	72
3.2.1.1	Ressources informatiques	72
3.2.1.2	Ressources consommées	73
3.2.1.3	Ressources produites	74
3.2.2	La capacité de ressource	74
3.2.3	Restrictions sur les méthodes des ressources	74
La 2ème partie		75
1.	La natation	75
1.1	Crawl :.....	76
1.1.1	Muscle du Crawl: les mouvements des bras	76
1.1.2	Muscle du Crawl : Mouvements des jambes	76
1.1.3	Muscle de crawl : Tronc	77
1.2	Dos	77
1.2.1	Muscle du Dos : les mouvements des bras :.....	77
1.2.2	Muscle du Dos : Mouvements des jambes	78

1.2.3	Muscle du Dos – Tronc	78
1.3	Papillon.....	79
1.3.1	Muscle du Papillon: les mouvements des bras	79
1.3.2	Muscle du Papillon: Mouvements des jambes	80
1.3.3	Muscle du Papillon-Tronc	80
1.4	Brasse	81
1.4.1	Muscle de la Brasse : les mouvements des bras	81
1.4.2	Muscle de la Brasse : les mouvements des jambes	82
1.4.3	Muscle de la Brasse-Tronc	83
	La 3ème partie	83
	Étude de cas.....	83
	Conclusion.....	87
	Conclusion générale	
	Conclusion générale et perspectives	89
	Bibliographie.....	90

Liste des figures :

Chapitre1 :

Figure 1: Architecture de IoT (A:three layers)(B :five layers).....	19
Figure 2 : Architecture de brouillard (fog) de passerelle (gateway) smart IoT.....	19
Figure 3 : Des exemples IoT pour la construction de propres maisons intelligentes.....	26
Figure 4 : Exemples de dispositifs IoT portables « wearable ».....	29
Figure 5 : Exemple de commerce de détail « Retail ».....	31
Figure 6 : Les applications d'IoT pour les villes intelligentes	32
Figure 7 : Des exemples des appareils des villes intelligentes.....	32
Figure 8 : Des exemples d'IoT pour le soin de santé.....	35
Figure 9 : Des exemples d'IoT en agriculture.....	36
Figure 10 : Des exemples d'IoT dans l'automobile / les transports.....	38
Figure 11 : Des exemples d'IoT pour l'automatisation industrielle.....	39
Figure 12 : Des exemples d'IoT de La gestion de l'énergie.....	40
Figure 13 : Des exemples d'IoT de domaine de sport.....	42

Chapitre2 :

Figure1 : Architecture de référence des services web.....	48
Figure 2: Rôles, opérations et artefacts des services Web.....	49
Figure 3 : Pille des services web.....	55
Figure 4: La relation entre XML/SOAP/WSDL/UDDI.....	56
Figure 5 : Structure de message SOAP.....	57
Figure 6 : Les composants des services web.....	58
Figure 7 : Structure d'un document WSDL.....	59

Chapitre3 :

La 1ère partie :

Figure 1 : Représentation de Framework *aaR	66
Figure 2 : Représentation de ressources informatiques	70
Figure 3 : Représentation de ressources consommées.....	70
Figure 4 : Représentation de ressources produites.....	71
Figure 5 : Modèle descripteur d'une ressource.....	72

La 2ème partie :

Figure 6: Crawl-Mouvements des bras	76
Figure 7 : Crawl-Mouvements des jambes.....	77
Figure 8 : Crawl -Tronc	77
Figure 9 : Dos-Mouvements des bras.....	78
Figure 10 : Dos-Mouvement des jambes.....	78
Figure 11 : Dos – Tronc.....	79
Figure 12 : Papillon: les mouvements des bras (phase 1).....	79
Figure 13 : Papillon: les mouvements des bras (phase 2).....	80
Figure 14 : papillon-les mouvements des jambes.....	80
Figure 15 : Papillon-Tronc.....	81
Figure 16 : Brasse : les mouvements des bras Phase Propulsive.....	81
Figure 17 : Brasse : les mouvements des bras Phase Dégagement.....	82
Figure18 : Muscle de la Brasse : les mouvements des jambes Phase Propulsive.....	82
Figure 19 : Brasse : les mouvements des jambes Phase Dégagement.....	83
Figure 20 : Brasse-Tronc	83

La 3ème partie :

Figure 21 : Les muscles principaux du nageur vu de dos.....	84
Figure 22 : Les muscles principaux du nageur vu de face.....	85
Figure 23 : L'emplacement des capteurs en tenue du nageur	86

Liste des tableaux

Chapitre 2 :

Table 1 : COMPARAISON DES APPROCHES DE CONCEPTION DE SERVICES WEB.....54

Chapitre 3 :

Table 1 : Application de restrictions aux méthodes de ressources.....75

Introduction générale

Introduction générale :

Internet où le réseau existant d'Internet se connectera aux objets du monde réel. Les objets peuvent inclure tous les objets, appareils ménagers, appareils, véhicules, etc., et lorsque ces objets se connectent à Internet dans une infrastructure spécifique via des protocoles standard, le système tout entier est alors appelé Internet des objets « IoT » [73-76]. Kevin Ashton a proposé l'Internet des objets en 1999, le Conseil national du renseignement des États-Unis définir l'internet des objets comme suit: «L'Internet des objets est l'idée générale des objets lisibles, reconnaissables, localisables, en particulier des objets de tous les jours. Adressables et contrôlables via Internet - que ce soit par RFID, LAN sans fil, réseau étendu ou tout autre moyen » [72]. L'Internet des objets est l'une des dernières avancées des technologies de l'information et de la communication, offrant une connectivité et une gestion mondiale des capteurs, des dispositifs, des utilisateurs et des informations [36,37]. Internet des objets à chaque fois fait beaucoup de défis pour prouver son importance dans différents domaines (domaine de santé, de sport, domaine d'agriculture, les maisons intelligentes, domaine de sécurité, etc.), L'objectif de l'Internet des objets est le monde intelligent, tous les objets dans le monde sont connectés et partagés.

Il y a un nouveau paradigme de l'internet des objets proposé par T. Baker et al. en 2018 [9], c'est « tout en tant que ressource (*aaR) », ce paradigme contient trois catégories de ressources (ressources consommées, ressources informatiques, ressources produites), les ressources consommées et produites ont des méthodes et capacités, les méthodes ont des restrictions.

Dans ce mémoire nous allons évoquer trois questions de recherches: -1: qu'est ce que l'internet des objets? -2: définir le tout en tant que ressource? -3: Comment appliquer ce paradigme dans le domaine sportif et précisément la natation. Ce mémoire est scindé en trois chapitres, nous avons parlé dans le 1^{ère} chapitre de l'internet des objets, définitions, domaines d'application et les différents architectures. Dans le 2^{ème} chapitre nous avons parlé des services Web: définitions, architecture, technologies des web services : SOAP, WSDL, UDDI et ses caractéristiques. Le 3^{ème} chapitre "contribution" contient trois parties : la 1^{ère} partie est une illustration du Framework (*aaR). La 2^{ème} partie est une étude de l'anatomie humaine (muscles du nageur) et la dernière partie est une étude de cas dans le domaine de la natation. L'objectif de ce mémoire est illustrer le Framework *aaR dans le domaine de sport (natation) avec une étude de cas.

Chapitre 1 :

Internet Des Objets « IoT »

1. Introduction

Kevin Ashton a d'abord proposé le concept de l'Internet des objets en 1999, et il l'a décrit comme un objet connecté interopérable identifiable de manière unique, doté de la technologie d'identification par radiofréquence (RFID : radio frequency identification) [66]. Le terme est étroitement associé à la RFID en tant que méthode de communication, bien qu'il puisse également inclure d'autres technologies de capteurs, des technologies sans fil ou des codes QR (Quick Response) [4].

L'intégration de plusieurs technologies et solutions de communication constitue le principal facteur habilitant de ce paradigme prometteur. Technologies d'identification et de suivi, réseaux de capteurs et d'actionneurs filaires et sans fil, protocoles de communication améliorés (partagée avec l'Internet de nouvelle génération) et l'intelligence distribuée pour les objets intelligents ne sont que les plus pertinentes [67]. Les bases de l'Internet des objets en tant que combinaison d'Internet et des technologies émergentes ont été discutées [68].

L'architecture du Framework IoT et les problèmes de conception des composants matériels et logiciels de l'IoT [69] ont été discutés. Ils ont élaboré les différents domaines d'application de l'IoT, tels que les villes intelligentes, les soins de santé, l'agriculture et les applications à l'échelle nanométrique [70].

Nous entamons ce chapitre par donner quelques définitions les plus communément adoptées par la communauté de l'IoT, ensuite nous présentons les différentes architectures et ses domaines d'application.

2. Concepts fondamentaux

2.1 Internet des objets ou « Internet Of Things (IOT) »

2.1.1 Définition :

L'Internet des objets « IdO » ou « IoT », est un système d'appareils informatiques interdépendants, de machines, d'objets, d'animaux ou de personnes numériques dotés d'identifiants uniques (UID) et la possibilité de transférer des données sur un réseau sans nécessité d'interaction d'homme à homme ou d'homme à ordinateur [1].

L'internet des objets représente les échanges d'informations et de données provenant de dispositifs présents dans la réalité vers le réseau internet. Il est considéré comme la troisième évolution de l'internet, baptisée web 3.0 qui fait suite à l'ère du web social. Surtout avec l'ajout d'une connexion internet qu'à donné une valeur supplémentaire en terme de fonctionnalité, d'information, d'interaction avec l'environnement ou d'usage [2].

Il fait référence à la connexion de périphériques (autres que les tarifs courants tels que les ordinateurs et les Smartphones) à Internet. Les voitures, les appareils de cuisine et même les moniteurs cardiaques peuvent tous être connectés via l'IoT [3].

L'Internet des objets est un concept informatique qui décrit l'idée que des objets physiques quotidiens sont connectés à Internet et puissent s'identifier à d'autres appareils. L'IoT est significatif car un objet qui peut se représenter numériquement devient quelque chose de plus grand que l'objet en lui-même. L'objet ne concerne plus uniquement son utilisateur, mais il est maintenant connecté aux objets environnants et aux données de la base de données. Lorsque de nombreux objets agissent à l'unisson, ils sont connus pour avoir une "intelligence ambiante"[4].

L'Internet des objets « IoT » est défini comme «Un réseau ouvert et complet d'objets intelligents capables d'auto-organiser, de partager des informations, des données et des ressources, de réagir et d'agir face aux situations et aux changements de l'environnement». L'Internet des objets est l'une des dernières avancées des technologies de l'information et de la communication, offrant une connectivité et une gestion mondiale des capteurs, des dispositifs, des utilisateurs et des informations [36,37].

2.2 Ressources :

2.2.1 Définition :

« Ensemble des moyens dont on dispose pour agir avec efficacité » [83].

« Ensemble des connaissances déclaratives et procédurales, des capacités fonctionnelles et Structurales, des aptitudes relatives aux différentes composantes de la conduite, qui constituent le répertoire caractérisant les pouvoirs d'un sujet » [84].

Dans les disciplines des Technologies de l'information et de la Communication (TIC) telles que l'informatique, les technologies de l'information et le génie logiciel, ressources signifient différentes choses. Il fait référence au matériel et aux logiciels (par exemple, réseau, serveurs, stockage, applications et services) dans le cloud computing [17], personne / machine qui exécute des tâches dans des processus métier [16], contenu Web dans l'architecture Web sémantique [18,19], etc. Malgré les multiples utilisations des ressources, elles abstraits certaines entités, physiques ou logiques, qui pourraient être découvertes, composées et consommées de manière à atteindre certains objectifs de l'entreprise. Il est à noter que de telles entités pourraient ne pas être abondantes et que, par conséquent, des restrictions sur leur utilisation sont jugées nécessaires [9].

Karkkainen et al. Ont Proposé une approche de gestion de l'information sur les produits utilisant une architecture "peer-to-peer" en association avec un référentiel de données centralisé [15]. Les données (considérées comme une ressource consommée) recueillies auprès de différentes parties prenantes sont transformées en informations sur les produits, puis partagées avec l'ensemble du réseau d'approvisionnement de partenaires. Cependant, la construction d'un référentiel centralisé pour intégrer une grande variété de données pourrait devenir coûteuse avec l'utilisation de nombreuses ressources (informatiques) [9].

3. Architecture de l'IoT :

Il n'existe pas de consensus unique sur l'architecture de l'IoT, qui fait l'objet d'un accord universel. Différentes architectures ont été proposées par différents chercheurs.

3.1. Architectures à trois et cinq couches [50] :

L'architecture la plus élémentaire est une architecture à trois couches (3-5), comme illustré à la figure 1. Elle a été introduite aux premiers stades de la recherche dans ce domaine. Il comporte trois couches, à savoir les couches de perception, de réseau et d'application.

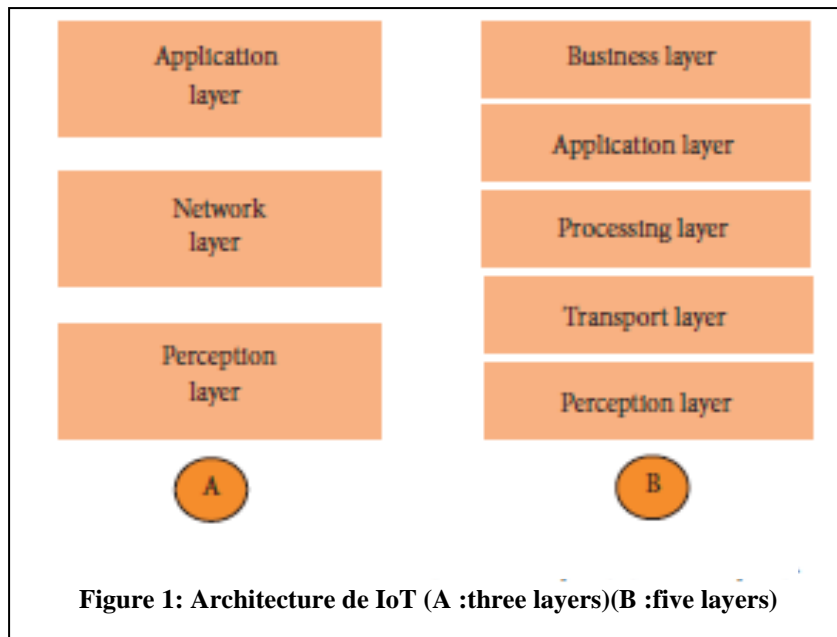
-
- (i) **La couche de perception:** est la couche physique, dotée de capteurs pour détecter et collecter des informations sur l'environnement. Il détecte certains paramètres physiques ou identifie d'autres objets intelligents dans l'environnement.
 - (ii) **La couche réseau:** est responsable de la connexion à d'autres objets intelligents, périphériques réseau et serveurs. Ses fonctionnalités sont également utilisées pour la transmission et le traitement des données de capteurs.
 - (iii) **La couche application:** est chargée de fournir des services spécifiques à l'application à l'utilisateur. Il définit diverses applications dans lesquelles l'Internet des objets peut être déployé, telles que les maisons intelligentes, les villes intelligentes et la santé intelligente.

L'architecture à trois couches définit l'idée principale de l'Internet des objets, mais elle n'est pas suffisante pour les recherches sur l'Internet des objets, car elle porte souvent sur des aspects plus fins de l'Internet des objets. Pour cette raison il existe beaucoup plus d'architectures en couches proposées dans la littérature. L'une est l'architecture à quatre couches, qui inclut en outre les couches de traitement et d'activité (3-6).

Les cinq couches sont les couches de perception, de transport, de traitement, d'application et de gestion. Le rôle des couches de perception et d'application est identique à celui de l'architecture à trois couches. La fonction des trois couches restantes :

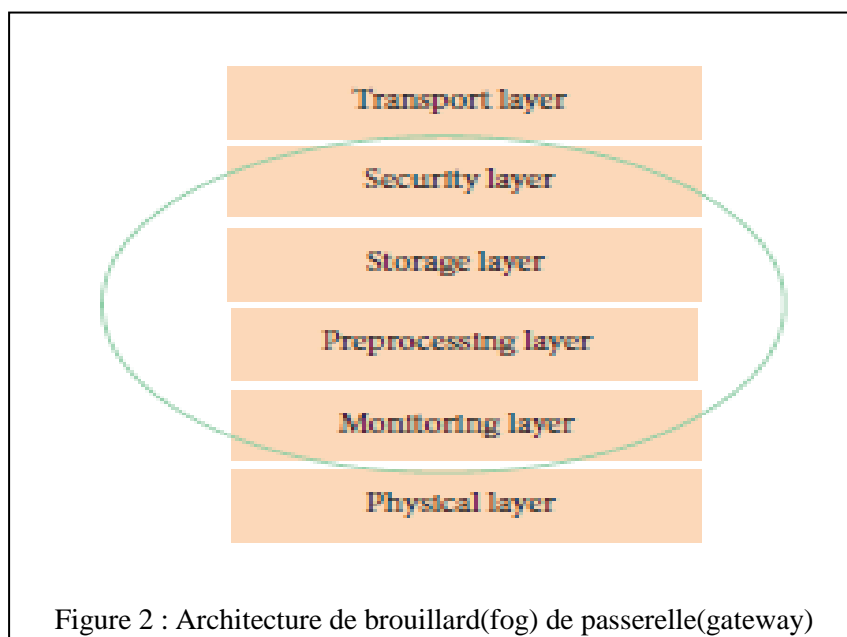
- (i) **couche transport:** transfère les données du capteur de la couche de perception à la couche de traitement et inversement via des réseaux tels que sans fil, 3G, LAN, Bluetooth, RFID et NFC.
- (ii) **couche traitement:** est également appelée couche middleware. Il stocke, analyse et traite d'énormes quantités de données provenant de la couche de transport. Il peut gérer et fournir un ensemble diversifié de services aux couches inférieures. Il utilise de nombreuses technologies telles que bases de données, informatique en nuage et modules de traitement de données volumineuses.
- (iii) **couche gestion:** gère l'ensemble du système IoT, y compris les applications, les modèles commerciaux et de profit, ainsi que la confidentialité des utilisateurs.

La figure suivante représentée l'architecture de IoT (A :three layers)(B :five layers) [50] :



Une autre architecture proposée par Ning et Wang [51] s’inspire des couches de traitement du cerveau humain. Il est inspiré par l’intelligence et la capacité des êtres humains à penser, ressentir, se souvenir, prendre des décisions et réagir à l’environnement physique. Il est constitué de trois parties. Le premier est le cerveau humain, analogue à l’unité de traitement et de gestion des données ou au centre de données. Deuxièmement, la moelle épinière est analogue au réseau distribué de nœuds de traitement de données et de passerelles intelligentes. Troisièmement, le réseau de nerfs, qui correspond aux composants et capteurs du réseau.

La figure suivante représentée « fog architecture of a smart IOT gateway » [50] :



3.2 Architectures basées sur le Cloud computing [50] :

Parlons maintenant de deux types d'architectures de systèmes: l'informatique dans le cloud. Cette classification est différente de la classification de l'Architecture à trois et cinq couches, qui a été effectuée sur la base de protocoles. En particulier, Il existe une certaine ambiguïté sur la nature des données générées par les dispositifs IoT et sur la nature du traitement des données. Dans certaines architectures de système, le traitement des données est effectué de manière centralisée par des ordinateurs dans le cloud.

Une telle architecture centrée sur le cloud garde le cloud au centre, les applications au-dessus de lui et le réseau d'objets intelligents en dessous [52]. Le cloud computing est primé parce qu'il offre une grande flexibilité et évolutivité. Il offre des services tels que l'infrastructure principale, la plate-forme, les logiciels et le stockage.

Les développeurs peuvent fournir leurs outils de stockage, leurs outils logiciels, leurs outils d'exploration de données et d'apprentissage automatique, ainsi que leurs outils de visualisation via le cloud.

Dernièrement, on assiste à une évolution vers une autre architecture système, à savoir le cloud computing [53,54], dans laquelle les capteurs et les passerelles de réseau effectuent une partie du traitement et de l'analyse des données. Une architecture de cloud [55] présente une approche en couches, illustrée à la figure 2, qui insère des couches de surveillance, de prétraitement, de stockage et de sécurité entre les couches physique et de transport. La couche de surveillance surveille l'alimentation, les ressources, les réponses et les services. La couche de prétraitement effectue le filtrage, le traitement et l'analyse des données du capteur. La couche de stockage temporaire fournit des fonctionnalités de stockage telles que la réplication, la distribution et le stockage de données. Enfin, la couche de sécurité effectue le cryptage / décryptage et garantit l'intégrité et la confidentialité des données. La surveillance et le prétraitement sont effectués à la périphérie du réseau avant l'envoi des données vers le cloud.

Souvent, les termes «cloud computing» et «calcul en périphérie» sont utilisés de manière interchangeable. Ce dernier terme précède le premier et est interprété comme étant plus générique. Le cloud computing à l'origine.

Cisco désigne les passerelles intelligentes et les capteurs intelligents, tandis que le cloud computing est légèrement plus pénétrante.

Ce paradigme envisage l'ajout de fonctionnalités de prétraitement de données intelligentes à des périphériques physiques tels que des moteurs, des pompes ou des éclairages.

L'objectif est d'effectuer autant que possible le prétraitement des données dans ces appareils, qu'on qualifie de périphérie du réseau. En termes d'architecture du système, le diagramme d'architecture n'est pas sensiblement différent de celui de la figure 2. Par conséquent, nous ne décrivons pas le cloud computing séparément.

Enfin, la distinction entre les architectures de protocole et les architectures système n'est pas très nette. Souvent, les protocoles et le système sont signés par un code. Nous allons utiliser la pile de protocoles IoT générique à 5 couches (diagramme d'architecture présenté à la figure 2) pour les architectures de cloud computing.

3.3 IoT social [50]:

Le nouveau paradigme: l'IoT social (SIoT). Considérer les relations sociales entre les objets de la même manière que les êtres humains forment des relations sociales (voir [56]).

Voici les trois facettes principales d'un système SIoT:

- (i) **Le SIoT est navigable:** commencer avec un appareil et naviguer à travers tous les appareils qui y sont connectés. Il est facile de découvrir de nouveaux appareils et services en utilisant un tel réseau social d'appareils IoT.
- (ii) **Un besoin de confiance (force de la relation):** est présent entre les appareils (semblable aux amis sur Facebook).
- (iii) Des modèles similaires à ceux utilisés pour étudier les réseaux sociaux humains peuvent également être utilisés pour étudier les réseaux sociaux des appareils IoT.

3.3.1 Composants de base:

Dans un environnement IoT social typique, traitez les appareils et les services comme des robots où ils peuvent créer et modifier des relations au fil du temps. Cela permettra aux appareils de collaborer de manière transparente et d'accomplir une tâche complexe. Pour qu'un tel modèle fonctionne, nous avons besoin de nombreux composants interopérables. Examinons quelques-uns des principaux composants d'un tel système :

(1) **ID:** nous avons besoin d'une méthode unique d'identification d'objet. Un ID peut être affecté à un objet en fonction de paramètres classiques tels que l'ID MAC, l'ID IPv6, un code produit universel ou une autre méthode personnalisée.

(2) **Méta information:** avec un ID, nous avons besoin de méta information sur le périphérique qui décrit sa forme et son fonctionnement. Cela est nécessaire pour établir des

relations appropriées avec le périphérique et également pour le placer de manière appropriée dans l'univers des périphériques IoT.

(3) Contrôles de sécurité: ceci est similaire aux paramètres de «liste d'amis» sur Facebook. Le propriétaire d'un périphérique peut imposer des restrictions sur les types de périphériques pouvant s'y connecter. Celles-ci sont généralement appelées contrôles de propriétaire.

(4) Découverte de services: ce type de système s'apparente à un nuage de services, dans lequel nous avons besoin de répertoires dédiés qui stockent les détails des périphériques fournissant certains types de services. Il devient très important de maintenir ces répertoires à jour pour que les périphériques puissent en apprendre davantage sur les autres.

(5) Gestion des relations: ce module gère les relations avec d'autres appareils. Il stocke également les types de périphériques auxquels un périphérique donné doit essayer de se connecter en fonction du type de services fournis.

(6) Composition des services: ce module pousse le modèle IoT social à un nouveau niveau. L'objectif ultime d'un tel système est de fournir des services mieux intégrés aux utilisateurs. Par exemple, si une personne a un capteur de puissance avec son climatiseur et que ce dispositif établit une relation avec un moteur d'analyse, il est alors possible que l'ensemble fournisse beaucoup de données sur les modes d'utilisation du climatiseur. Si le modèle social est plus étendu et qu'il y a beaucoup plus de périphériques, il est alors possible de comparer les données avec les modèles d'utilisation d'autres utilisateurs et de générer des données encore plus significatives. Par exemple, on peut dire aux utilisateurs qu'ils sont les plus gros consommateurs d'énergie de leur communauté ou de leurs amis sur Facebook .

3.3.2 Architecture représentative:

La plupart des architectures proposées pour le SIoT ont également une architecture côté serveur.

Le serveur se connecte à tous les composants interconnectés, agrège (compose) les services et agit comme un point de service unique pour les utilisateurs.

L'architecture côté serveur comporte généralement trois couches. La première est la couche de base qui contient une base de données qui stocke les détails de tous les périphériques, de leurs attributs, du méta information et de leurs données, des relations. La deuxième couche (couche Composante) contient du code permettant d'interagir avec les périphériques, d'interroger leur statut et d'utiliser un sous-ensemble d'entre eux pour effectuer un service. La couche supérieure est la couche d'application, qui fournit des services aux utilisateurs.

Du côté des périphériques (objets), nous avons généralement deux couches. La première est la couche objet, qui permet à un périphérique de se connecter à d'autres périphériques, de leur parler (via des protocoles normalisés) et d'échanger des informations. La couche objet transmet des informations à la couche sociale. La couche sociale gère l'exécution des applications des utilisateurs, exécute les requêtes et interagit avec la couche d'application sur le serveur.

4. Les domaines d'application d'IoT :

Les exemples d'internet des objets vont des maisons connectées intelligentes aux appareils portables en passant par les soins de santé. Il n'est pas faux de suggérer que l'IoT fait maintenant partie de tous les aspects de nos vies. Non seulement les applications Internet des objets améliorent le confort de nos vies, mais elles nous donnent également plus de contrôle en simplifiant la vie quotidienne et les tâches personnelles [6].

Avec le battage médiatique récente sur les perspectives d'avenir de l'IoT, les entreprises ont été obligées de prendre l'initiative de proposer des éléments de base de l'internet des objets, à savoir du matériel, des logiciels et un support, pour permettre aux développeurs de déployer des applications capables de connecter tout ce qui est dans l'internet [6].

Nous savons que le potentiel des marchés de l'Internet des objets est énorme, mais certains domaines vont mûrir beaucoup plus rapidement que les autres. Nous répertorions ici les domaines d'application de l'Internet des objets avec des exemples susceptibles de générer une croissance exponentielle [6].

4.1 Applications Internet des objets (IoT) pour la maison connectée / intelligente « Connected/Smart home »

Une maison intelligente est la maison ou ce milieu de vie doté d'une technologie permettant le contrôle automatique de tous les appareils ménagers / appareils ménagers et ce, à distance [10].

Dans les maisons intelligentes, l'utilisateur peut facilement surveiller et contrôler tous les appareils ménagers / appareils ménagers via Internet. Les appareils ménagers se connectent dans une architecture réseau appropriée prédéfinie et à l'aide de protocoles standard [11].

L'ensemble du système peut être divisé en deux parties: l'une comprend tous les dispositifs domestiques, les modules de commutation et le récepteur émetteur RF, et l'autre, l'interface, le processeur, le collecteur de données et le module GPRS (Le General Packet Radio Service) qui communiquera avec Internet [12].

Alerte: Le système de maison intelligente peut détecter son environnement et envoyer des alertes à l'utilisateur sur le périphérique ou le compte enregistré. L'alerte contient des informations liées aux données environnementales. Ces informations peuvent inclure le niveau de gaz différents dans l'environnement, la température, l'humidité, l'intensité lumineuse, etc. Une alerte peut être envoyée à l'utilisateur régulièrement à une heure prédéfinie. Les alertes peuvent être envoyées par courrier électronique, sous forme de message texte, de tweets ou de tout autre média social [14].

Moniteur: Une maison intelligente est capable de surveiller son environnement à l'aide de divers capteurs et flux de caméra. La surveillance est une fonction importante dans la mesure où elle suit toutes les activités d'une maison intelligente, qui constitue le principal besoin sur la base duquel toute action ultérieure peut être prise ou une décision prise. Par exemple, surveiller la température ambiante et envoyer une alerte à l'utilisateur pour qu'il allume le climatiseur si la température est supérieure au seuil [14].

Contrôle: Cette fonction de maison intelligente permet à l'utilisateur de contrôler différentes activités. Les activités peuvent inclure l'allumage / extinction des lumières, le climatiseur et électroménagers, verrouiller / déverrouiller les portes, ouvrir / fermer les fenêtres et les portes, etc. L'utilisateur peut contrôler des choses du même endroit ou d'un endroit distant. Elle permet même à l'utilisateur d'automatiser une activité telle qu'allumer / éteindre automatiquement le climatiseur lorsque la température ambiante est élevée / basse [14].

Intelligence: Intelligence ou Home Intelligence (HI) est la fonction la plus importante de la maison intelligente et fait référence au comportement intelligent de l'environnement de la maison intelligente. Cette fonction est liée à la prise de décision automatique lors de la survenue de divers événements. HI dépend du mécanisme d'intelligence artificielle (AI)

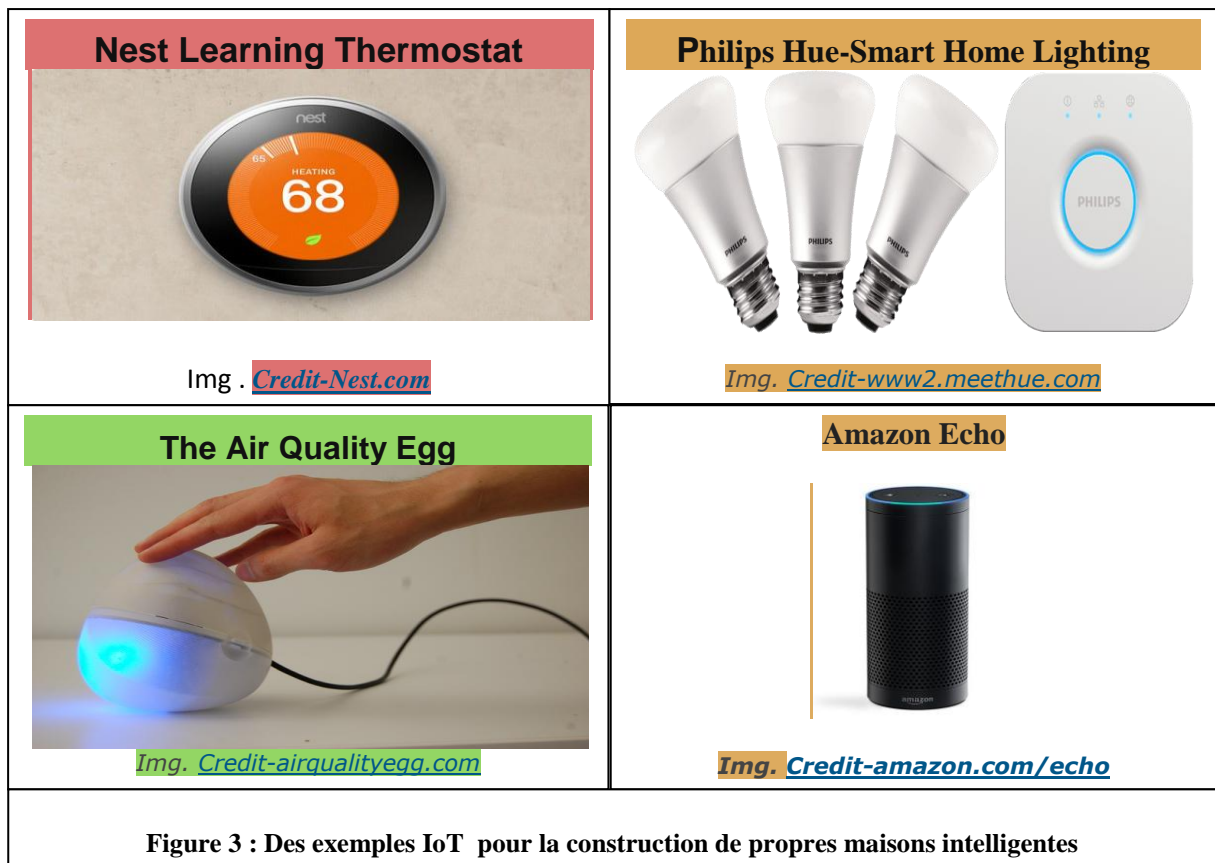
construit dans l'environnement de la maison intelligente. HI ne donne pas seulement le cerveau à la maison intelligente, il est également très important pour la sécurité dans une maison [13].

HI crée un environnement intégré dans la maison intelligente dans lequel le mécanisme d'IA peut identifier et réagir de manière appropriée en fonction de l'évolution des conditions et des événements. En identifiant des événements anormaux ou inattendus HI peut alerter l'utilisateur et fournir une réponse automatique immédiate si désiré. Certains scénarios illustrés consistent automatiquement à préparer le café dès son arrivée, à envoyer une alerte à chaque fois qu'une activité suspecte est détectée à la porte ou à l'intérieur de la maison, à commander automatiquement les produits en cas de pénurie de réfrigérateurs, à envoyer une notification à électricien / plombier chaque fois qu'un entretien est nécessaire, etc [14].

La définition de la maison connectée est différente pour différentes personnes. En termes simples, une maison intelligente est celle dans laquelle les appareils ont la capacité de communiquer les uns avec les autres, ainsi qu'avec leur environnement immatériel. Une maison intelligente donne au propriétaire la possibilité de personnaliser et de contrôler l'environnement de la maison pour une sécurité accrue et une gestion efficace de l'énergie. Des centaines de technologies IoT sont disponibles pour surveiller et construire des maisons intelligentes.

Les fabricants de produits de consommation tels que Belkin, Philips, Amazon et Haier se sont déjà imposés comme des sociétés de premier plan sur ce marché [6].

La figure suivante représentée des exemples d'Internet des objets pour la construction de vos propres maisons intelligentes [6] :



Éclairage intelligent: Un éclairage intelligent est utilisé pour réaliser des économies d'énergie en adaptant l'éclairage aux conditions ambiantes et en allumant / éteignant ou en atténuant l'éclairage en fonction des besoins de l'utilisateur, réduisant ainsi l'utilisation inutile d'énergie. Économiser de l'énergie aide également à réduire les coûts. L'éclairage intelligent peut être mis en œuvre avec un éclairage à semi-conducteurs (LED) ou des lampes IPenabled (contrôlées par Internet ou sans fil). L'éclairage intelligent détecte l'occupation, la température / humidité et le niveau de LUX dans l'environnement.

Appareils intelligents: Les appareils intelligents sont utilisés pour collecter des informations sur l'état des appareils et pour contrôler facilement les appareils depuis la pièce ou à distance. Il est également utilisé pour planifier des tâches à une heure et à une heure prédéfinies.

Pour l'intégration d'exécution entre les appareils. Les appareils intelligents permettent d'économiser de l'énergie et du temps.

Détection d'intrusion: La détection d'intrusion est utilisée pour alerter l'utilisateur par courrier électronique et message texte. L'application de détection d'intrusion peut également envoyer un rapport détaillé avec des images ou un clip audio / vidéo à l'utilisateur. L'objectif

principal de cette application est de surveiller l'activité suspecte dans la maison intelligente, d'alerter l'utilisateur et de prendre les mesures nécessaires à des fins de sécurité.

Détection de fumée / gaz: Cette application est utilisée pour détecter l'environnement de la maison intelligente pour une vie saine et peut également être utilisée pour la sécurité. Cette application est utilisée pour les techniques de détection optique, d'ionisation et d'échantillonnage de l'air. Il est capable de donner l'alerte à la caserne de pompiers à proximité en cas d'incendie et de fumée et à l'utilisateur par e-mail / SMS pour l'informer des risques pour la santé [14].

4.2 Applications Internet des objets (IoT) pour les vêtements « Wearable »

Les Wearables (Vêtements portables) sont la nouvelle vague du secteur informatique. C'est une nouvelle façon d'interagir entre un humain et un ordinateur lorsqu'un appareil est porté par l'être humain. L'appareil est actif et accessible à l'homme tout le temps. Les humains peuvent être gérés par l'être humain en donnant certaines commandes et peuvent également fonctionner en mode automatisé.

Le paradigme Wearable (Vêtements portables) suppose une interaction entre un appareil et l'homme d'une manière ou d'une autre. Nous pouvons définir un ordinateur portable comme un appareil que l'utilisateur porte aussi confortablement que ses vêtements, toujours connecté à son utilisateur et disponible pour divers besoins informatiques. L'informatique portable (Wearable computing) peut donc être comprise comme l'informatique utilisée qui répond aux exigences de l'informatique [20,21 ,22].

L'utilisation de l'informatique à porter n'est pas limitée à une exigence spécifique.

Cela dépend uniquement de la créativité des développeurs. Les applications IoT suivent le format standard pour les applications qui consistent en un microcontrôleur, une unité d'alimentation, des interfaces et capteurs pour la collecte de données. L'une des formes les plus visibles d'informatique portable (wearable computing) peut être vue comme des suiveurs d'activité comme LiktFitBit et Up de Jawbone, ou des appareils plus avancés comme Google Glass et Samsung Smart Gear. Ces appareils sont toujours connectés. Les dispositifs Vêtements portables (wearable) peuvent interagir avec

d'autres dispositifs IoT afin d'échanger des données et des informations en fonction des besoins de l'application [23].

Une large gamme de dispositifs portables (wearable) est utilisée dans le secteur de la santé. Les capteurs de condition physique et autres applications sont utiles pour collecter diverses données relatives à la santé des utilisateurs, telles que la fréquence cardiaque, les heures de sommeil, la température, la pression artérielle, etc., qui peuvent être surveillées en permanence par des professionnels de la santé afin de fournir de meilleurs services de santé[23].

Les appareils portables peuvent être classés en deux catégories largement [23] :

Wearables (Vêtements portables) passifs: Des wearables conçus pour être utilisés passivement collecter des données, qui est utilisé par certaines applications, sont appelées ordinateurs portables passives (passive wearables). Ils ne nécessitent pas d'interaction directe avec l'homme et sont étroitement associés à l'application utilisant les données générées. Certains exemples sont les compteurs de pas (step counters) et les dispositifs de surveillance de la fréquence cardiaque (heart rate monitoring devices).

Wearables (Vêtements portables) actifs: les vêtements portables (wearables) actifs tels que les montres et les lunettes intelligentes sont activement connectés aux utilisateurs finaux pour leur fournir des informations. Les wearables actifs constituent une source d'informations permettant de contrôler les utilisateurs finaux et les appareils mobiles qu'ils utilisent. Leur connectivité peut être étendue à d'autres appareils situés à proximité.

L'intégration de cloud, de capteurs portables (wearable sensors) et de smartphones a provoqué une éruption de données pouvant être utilisées dans diverses applications telles que les soins de santé personnalisés [23].

Les technologies vestimentaires sont l'une des tendances les plus en vogue dans l'internet des objets Apple, Samsung, Jawbone et beaucoup d'autres survivent dans une compétition acharnée.

Wearable IoT tech est un très grand domaine et consiste en un ensemble de périphériques. Ces dispositifs couvrent largement les exigences en matière de forme physique, de santé et de divertissement. La technologie Internet des objets pour les applications portables doit

impérativement être extrêmement économe en énergie ou à très basse consommation et de petite taille.

La figure suivante représentée des exemples de vêtements « wearable » [6] :



Figure 4 : Exemples de dispositifs IoT portables « wearable »

4.3 Applications Internet des objets (IoT) dans le commerce de détail « Retail »

Il y a une demande croissante d'innovation des consommateurs. Les consommateurs s'attendent à trouver plus d'outils interactifs et de soutien qui améliore l'expérience d'achat globale [85]. Un sondage réalisé par [86] auprès des clients indique que, selon un sondage d'Infosys, 86% des clients seraient prêts à payer 25% de plus pour une meilleure expérience de magasinage. L'Internet des objets fournira un environnement interconnecté dans lequel les produits et services sont hautement personnalisés pour chacun. En outre, cela procure un sentiment de flexibilité qui n'est pas connu dans le commerce de détail traditionnel. Il permet de connecter les mondes physique et numérique, permettant ainsi une interaction en temps réel avec le client [87] [30].

4.3.1 Modèle de consommation Retail 4.0 IoT :

IoT2c (IoT au consommateur) fait référence à l'utilisation de l'IoT par les consommateurs dans leur objectif quotidien. Ici, l'IoT peut inclure les Smartphones, les réfrigérateurs, les lave-linge, les téléviseurs, les capteurs de fitness et les appareils portables, qui sont des appareils grands publics. 63% des appareils grand public connectés à Internet sont des Smartphones. Cela peut aider le client dans son utilisation quotidienne et améliorer l'expérience client. Il fournit un accès rapide et efficace aux services. De plus, cela aide à résoudre les problèmes des consommateurs plus rapidement. Cela facilite également la commande automatique des produits [31].

4.3.2 Modèle de détaillant Retail 4.0 IoT :

IoT2B (IoT to business) est l'utilisation de l'IoT en entreprise. L'IoT peut être utilisé dans les magasins de détail. Il existe différentes méthodes pour utiliser l'IoT dans les magasins de vente au détail. Le modèle proposé comprend les parties suivantes: Caddies intelligentes, vérification automatisé des balises, Suivi et analyse de clientèle en temps réel, Mobilisation mobile, Smart Rack, Tarification dynamique, Sécurité intelligente, Reconnaissance des gestes du visage, Miroirs intelligents, Écrans intelligents [31].

Le potentiel de l'IoT dans le commerce de détail est énorme. Imaginez le cas où vos appareils électroménagers pourraient vous informer de la pénurie de fournitures ou même les commander tous seuls. Ce modèle de publicité intelligente basé sur la proximité de la vente au détail commence à devenir une réalité. Nous avons déjà des exemples d'applications Internet des objets dans le cadre de chaînes d'approvisionnement intelligentes. Applications de suivi des marchandises, d'échange d'informations en temps réel sur les stocks entre fournisseurs et détaillants et de livraison automatisée, toutes existantes mais de portée limitée [5].

La figure suivante représentée quelques exemples de commerce de détail « Retail » [6]:



4.4 Applications Internet des objets (IoT) pour les villes intelligentes « Smart Cities »

L'IoT utilise Internet pour fusionner diverses choses hétérogènes. En conséquence et pour fournir la facilité d'accès, tous les éléments existants doivent être lié à Internet. Cela s'explique par le fait que les villes intelligentes comprennent des réseaux de capteurs et que la connexion d'appareils intelligents à Internet est essentielle pour surveiller à distance leur traitement, notamment la surveillance de la consommation électrique afin d'améliorer l'utilisation de l'électricité, la gestion de la lumière, la climatisation. Pour atteindre cet objectif, les capteurs peuvent être étendus à divers endroits pour collecter et analyser des données en vue d'une amélioration de l'utilisation [33,34].

La figure suivante représentée les applications d'IoT pour les villes intelligentes [34] :

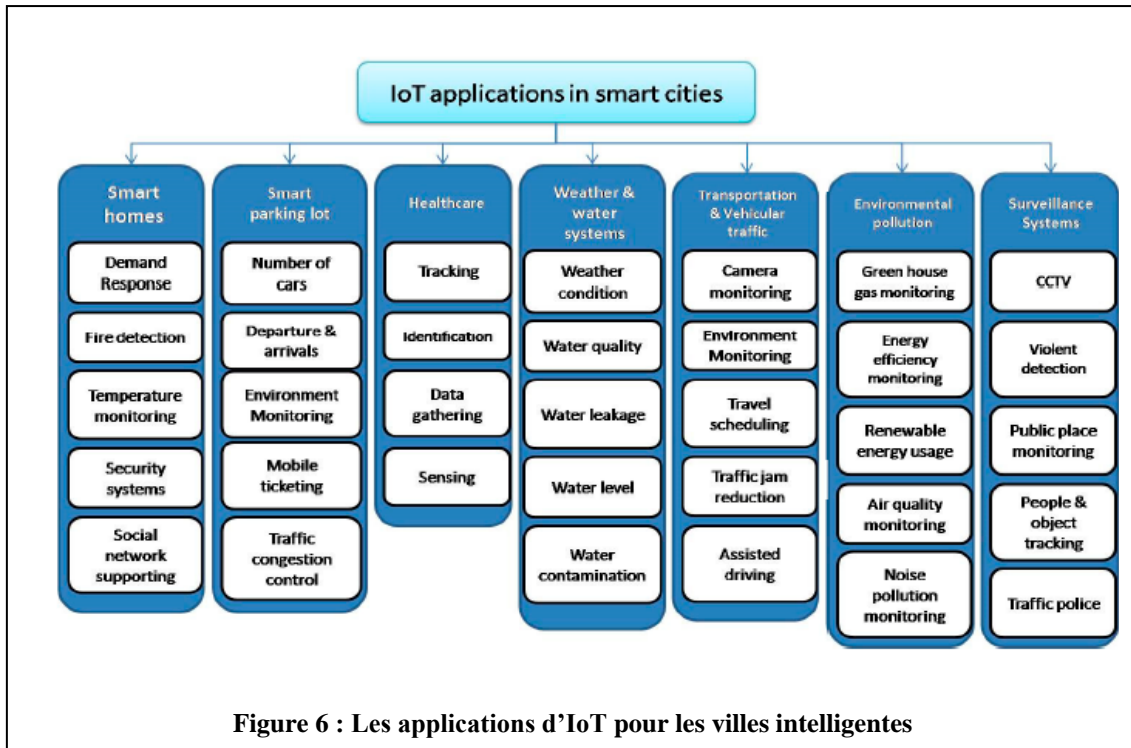


Figure 6 : Les applications d'IoT pour les villes intelligentes

Une surveillance intelligente, des transports plus sûrs et automatisés, des systèmes de gestion de l'énergie plus intelligents et une surveillance de l'environnement sont autant d'exemples d'applications Internet des objets pour les villes intelligentes. Les villes intelligentes sont les véritables solutions substantielles aux problèmes auxquels les gens sont généralement confrontés en raison de leur explosion, de la pollution, de la médiocrité des infrastructures et du manque d'approvisionnement en énergie [6].

La figure suivante représentée quelques exemples d'appareils IoT des villes intelligentes [6] :



Figure 7 : Des exemples des appareils des villes intelligentes

4.5 Applications de l'Internet des objets (IoT) dans les soins de santé « Healthcare » :

La dépendance des soins de santé vis-à-vis de l'IoT augmente chaque jour pour améliorer l'accès aux soins, améliorer la qualité des soins et, plus important encore, réduire le coût des soins [27,29].

Fondée sur les caractéristiques biologiques, comportementales, sociales et culturelles uniques d'un individu, la pratique intégrée du bien-être, des soins de santé et du soutien aux patients est appelée soins de santé personnalisés. Cela responsabilise chaque individu en respectant le principe de base des soins de santé, à savoir «les bons soins pour la bonne personne au bon moment», ce qui conduit à de meilleurs résultats et à une amélioration de la satisfaction, rendant ainsi les soins de santé rentables. Durable le service se concentre sur la prévention, la détection précoce de la pathologie et les soins à domicile au lieu de soins cliniques coûteux, et vérifie le bien-être général pour anticiper les besoins et assurer la conformité aux plans de soins de santé. Internet des objets promet de gérer la personnalisation des services de soins et peut maintenir une identité numérique pour chaque personne. Différents équipements sont utilisés dans le secteur de la santé pour communiquer et créer le système de système omniprésent. Les systèmes de soins de santé personnalisés basés sur l'IoT sont classés en fonction des soins cliniques et de la surveillance à distance [28,29].

4.5.1 Soins cliniques :

Les systèmes de surveillance non invasifs basés sur l'IoT sont utilisés pour les patients hospitalisés dont l'état physiologique nécessite une attention constante. Ces systèmes de surveillance utilisent des capteurs pour collecter des informations physiologiques qui sont analysées et stockées à l'aide de passerelles et du cloud. Ces informations sont ensuite envoyées sans fil aux soignants pour une analyse et un examen approfondis, ce qui évite aux professionnels de la santé de vérifier les signes vitaux du patient après des intervalles réguliers. Au lieu de cela, il fournit un flux d'informations automatisé continu. Ainsi, la qualité des soins est améliorée grâce à une attention constante, ce qui réduit le coût des soins et élimine la nécessité pour un soignant de participer activement à la collecte et à l'analyse de données [27,29].

4.5.2 Surveillance à distance :

Le manque d'accès rapide à des systèmes de surveillance de la santé efficaces peut conduire à de nombreux risques pour la santé qui ne sont pas détectés, ce qui constitue un problème auquel le monde entier est confronté. Cependant, les petites solutions sans fil puissantes connectées via l'IoT permettent aux patients de bénéficier d'une surveillance au lieu de l'inverse. Les données sur la santé des patients peuvent être capturées en toute sécurité en utilisant ces solutions. Une variété de capteurs et d'algorithmes complexes sont utilisés pour analyser les données puis les partager via une connectivité sans fil. Les professionnels de la santé peuvent alors faire des recommandations de santé appropriées à distance [29].

Shen [88] a étudié le fait que le système de soins de santé en ligne comprend principalement trois domaines: zone du corps, communication et réseautage, et service. Le domaine de la zone de corps est défini par un certain nombre de réseaux de zone de corps sans fil (WBAN), chacun correspondant à un utilisateur [5].

La fonctionnalité principale du domaine de la communication et de la mise en réseau consiste à relier la zone du corps et les domaines de service. Les technologies de communication sans fil évoluées (réseaux cellulaires, WiFi et WiMAX, par exemple) relient les passerelles WBAN à Internet et permettent communication de données mutuelle entre deux réseaux WBAN. Dans le domaine de service, une autorité de confiance gère un serveur en ligne chargé de recevoir, d'enregistrer et d'analyser les informations relatives à la santé de l'utilisateur [88].

Les soins de santé sont un secteur censé être fortement stimulé par l'avènement des applications de l'internet des objets. Les exemples d'IoT dans ce domaine sont nombreux [6].

La figure suivante représentée des exemples d'IoT pour le soin de santé [6] :

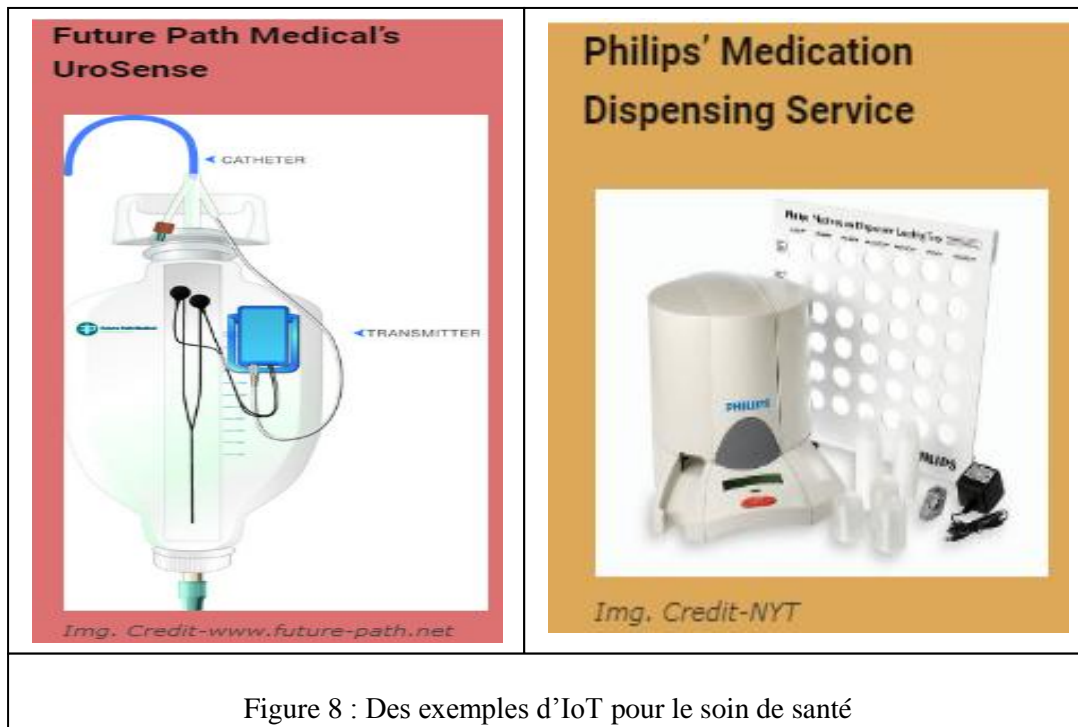


Figure 8 : Des exemples d'IoT pour le soin de santé

4.6 Applications de l'Internet des objets (IoT) en agriculture :

Le secteur de l'agriculture a besoin de solutions technologiques très institutionnelles et hautement évolutives. Les applications de l'Internet des objets peuvent fournir exactement la même chose aux agriculteurs [6].

Les appareils actuellement désignés comme IoT sont implémentés dans l'agriculture depuis de nombreuses années. Il existe principalement des solutions propriétaires dans lesquelles les dispositifs sont intégrés à des machines agricoles et donc leur utilisation est étroitement liée au fabricant de la machine. Un développement majeur dans ce domaine est attendu dans un proche avenir, en raison de nombreux projets axés sur des solutions ouvertes qui élimineraient les problèmes de compatibilité des appareils propriétaires [35].

Par exemple, le projet FarmBot ou Open Agriculture Initiative du MIT Media Lab («OpenAG») traitent tous deux de ces problèmes [35].

La figure suivante représenté des exemples d'IoT en agriculture [6],FramBot [35] :

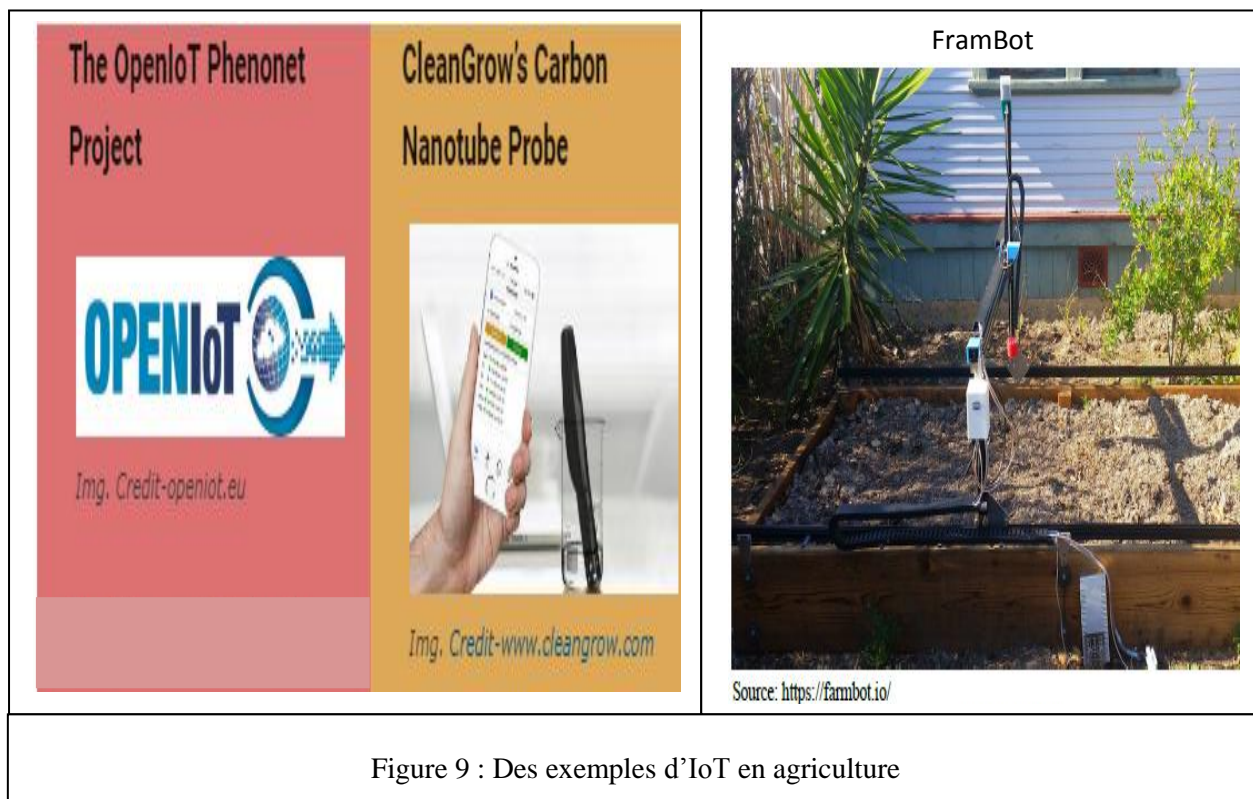


Figure 9 : Des exemples d'IoT en agriculture

4.7 Applications de l'Internet des objets (IoT) dans l'automobile / les transports « Automotive /Transportation »

Les voitures autonomes de Google sont connues de tous. L'IoT fait des voitures connectées une possibilité mais lentement. C'est un fait bien établi que toute nouvelle technologie met au moins deux ans à se propager dans l'industrie automobile traditionnelle. C'est pourquoi il n'y a pas un tel battage médiatique autour des exemples d'internet des objets connectés. Les entreprises et les startups annoncent toutes des technologies innovantes pour soutenir les plates-formes de voitures connectées [6].

L'automobile et les transports constituent l'un des secteurs les plus prometteurs pour l'Internet des objets (IoT). En convertissant de grandes quantités de données en connaissances utiles et exploitables, l'IoT peut aider à résoudre nombre des problèmes de sécurité, d'efficacité des transports et d'infrastructure de la société moderne. L'IoT permet des innovations telles que les voitures autonomes (autonomes), la gestion «intelligente» de la flotte et les infrastructures de transport intelligentes [38].

4.7.1 Voiture autonome / autonome / sans conducteur « An autonomous car/ self-driving car/ driverless car » :

Une voiture autonome /voiture autonome / voiture sans conducteur « An autonomous car/ self-driving car/ driverless car » [43,45] est un véhicule conçu pour détecter son environnement et naviguer sans intervention humaine. Avec les développements continus dans les voitures autonomes, la connectivité réseau entre la voiture et son environnement s'améliorera, offrant ainsi une toute nouvelle expérience de conduite. La voiture autonome alimentée par l'IoT est sous son propre contrôle une fois que la destination est prédéterminée et communique avec son environnement et le trafic en cours [40,45]. Les conducteurs de voitures autonomes peuvent profiter de la liberté d'utiliser leurs téléphones, ordinateurs portables et autres appareils sans crainte [45].

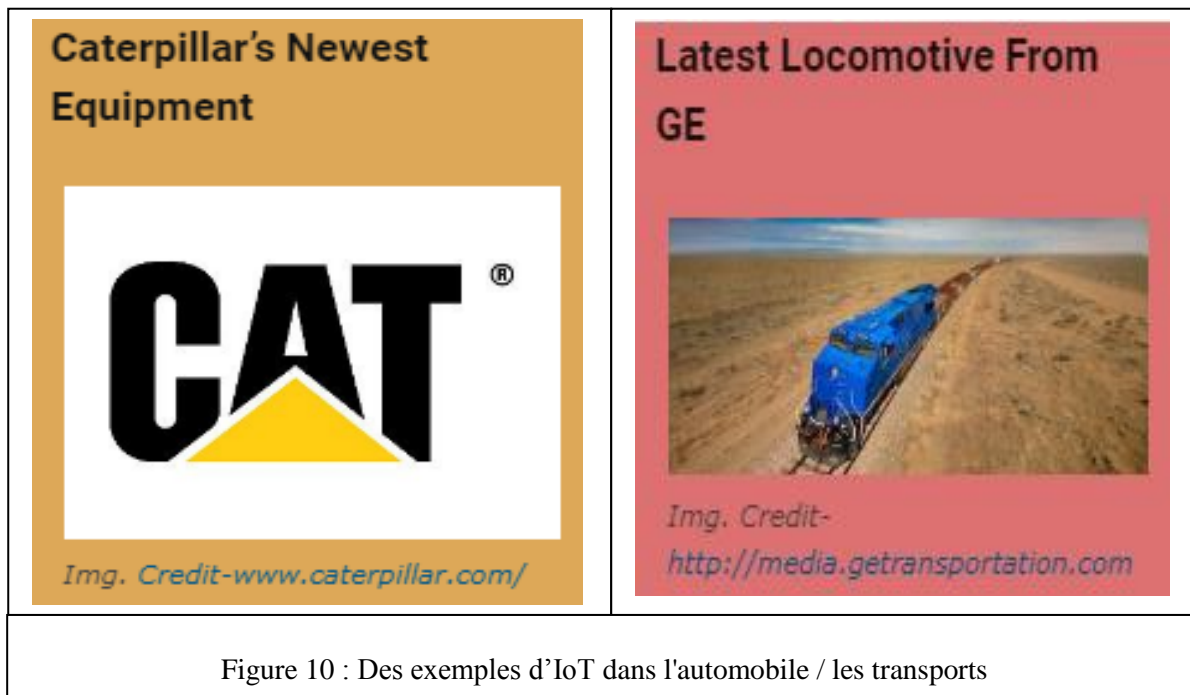
4.7.2 Voitures électriques « Electric cars »:

Une voiture électrique est une voiture alimentée par un moteur électrique [44,45], utilisant de l'énergie électrique. Stockés dans un ensemble de piles rechargeables [39,45]. Les voitures électriques gagnent en popularité en tant que alternative aux véhicules fonctionnant à l'essence ou au diesel, car ils sont plus durables, moins chers, plus faciles à maintenir et plus efficace [41,45].

Les voitures électriques deviennent plus courantes en raison de la disponibilité croissante de la technologie plug-in, qui a permis aux propriétaires de charger leurs voitures facilement à la maison, et plus puissantes batteries lithium-ion qui leur permettent de voyager sur de plus longues distances. Récemment, de nombreux pays ont commencé offrant des incitations aux stations de recharge publiques, faisant des voyages longue distance une option beaucoup plus économique pour les propriétaires de voitures électriques. De plus, certaines entreprises commencent à développer la recharge sans fil pour les voitures électriques, le processus de charge beaucoup plus facile [45].

Voitures électriques - les applications IoT sont : Solutions de recharge à domicile, Planification d'un emplacement de chargement, Station de charge la plus proche avec tarifs. État de charge de la batterie avec le temps de conduite estimé [45].

La figure suivante représenté des exemples d'IoT dans l'automobile / les transports [6] :



4.8 Applications Internet des objets (IoT) pour l'automatisation industrielle

L'Internet des objets industriel (IIoT) désigne l'application de l'Internet des objets (IoT) dans plusieurs secteurs tels que la fabrication, la logistique, le pétrole et le gaz, les transports, l'énergie / services publics, la chimie, l'aviation et d'autres secteurs industriels [24].

Avec Industrie 4.0, la quatrième révolution industrielle consiste à fusionner les domaines de l'automatisation et de l'information dans l'Internet industriel des objets, des services et des personnes. L'infrastructure de communication d'Industrie 4.0 permet aux appareils d'être accessibles sans obstacle dans l'Internet des objets industriel, sans sacrifier l'intégrité des sécurités et protection [25].

Le terme «Internet industriel» a été inventé par le géant industriel GE pour décrire la transformation industrielle dans le contexte connecté des machines, des systèmes cyber-physiques, de l'analyse avancée, de l'IA, des personnes, du cloud, etc. GE et l'Industrial Internet Consortium (IIC) ont décidé qu'IIoT était un synonyme d'Internet industriel. IIoT est sur le point d'apporter des opportunités sans précédent aux entreprises et à la société. Des

organisations comme IIC et IEEE travaillent d'arrache-pied pour définir et développer l'IIoT [26].

L'automatisation industrielle est l'une des applications les plus profondes de l'IoT. Grâce à l'infrastructure d'internet des objets reposant sur des réseaux de capteurs avancés, une connectivité sans fil, un matériel innovant et une communication machine à machine, les processus d'automatisation classiques des industries vont se transformer complètement. Les solutions d'automatisation IoT pour les industries de tous les grands noms tels que NEC, Siemens, Emerson et Honeywell sont déjà sur le marché [6].

La figure suivante représentée des exemples d'IoT pour l'automatisation industrielle [6] :



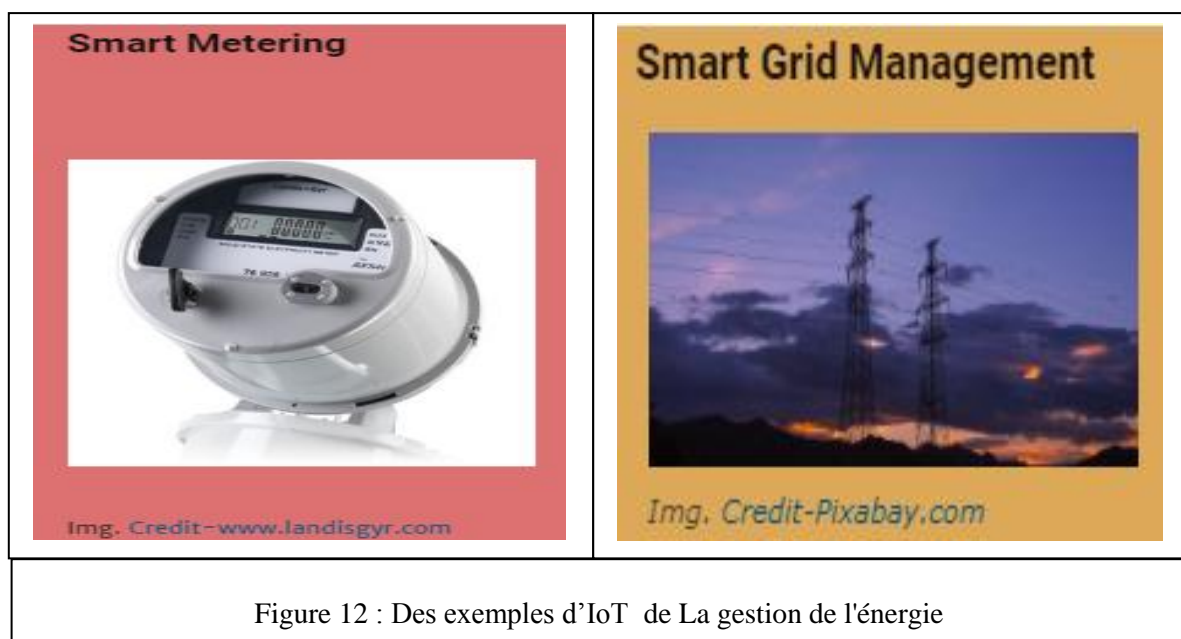
4.9 Gestion de l'énergie des applications Internet des objets (IoT) « Energy Management » :

La gestion de l'énergie comprend toutes les mesures prévues et mises en œuvre pour garantir une consommation d'énergie minimale pour l'activité en cours. La gestion de l'énergie influe sur les procédures organisationnelles et techniques, ainsi que sur les comportements, afin de réduire la consommation totale d'énergie en exploitation, d'utiliser économiquement les matériaux de base et les matériaux supplémentaires, ainsi que d'améliorer en permanence le rendement énergétique de l'entreprise. Un système de gestion de l'énergie enregistre

systématiquement les flux d'énergie et sert principalement de base aux investissements visant à améliorer l'efficacité énergétique. Un SME (système de management de l'énergie) efficace aide l'université à respecter les engagements pris dans sa politique énergétique et à améliorer de manière continue et systématique ses performances énergétiques. Le système de gestion de l'environnement englobe tous les éléments d'une organisation nécessaires à la création d'une politique énergétique et à la définition et à la réalisation d'objectifs stratégiques. Il comprend donc les structures organisationnelles et informationnelles nécessaires à la mise en œuvre de la gestion de l'énergie, y compris les ressources. Il élabore et met en œuvre la politique énergétique, avec la planification, l'introduction et l'exploitation, le suivi et la mesure, le contrôle et la correction, les audits internes, ainsi qu'une revue de direction régulière [32].

Les réseaux électriques du futur seront non seulement suffisamment intelligents, mais également extrêmement fiables. Le concept de réseau intelligent devient très populaire. L'idée de base des réseaux intelligents est de collecter des données de manière automatisée et d'analyser le comportement des consommateurs et des fournisseurs d'électricité afin d'améliorer leur efficacité ainsi que les coûts de l'utilisation de l'électricité [6].

La figure suivante représente des exemples d'IoT de La gestion de l'énergie [6] :



4.10 Application Internet des objets(IoT) dans le domaine de sport :

Le domaine du sport est en effet révolutionné par l'internet des objets. Il y a tellement d'innovations construites pour changer la façon dont les jeux sont joués et la façon dont les gens regardent leurs jeux préférés [8].

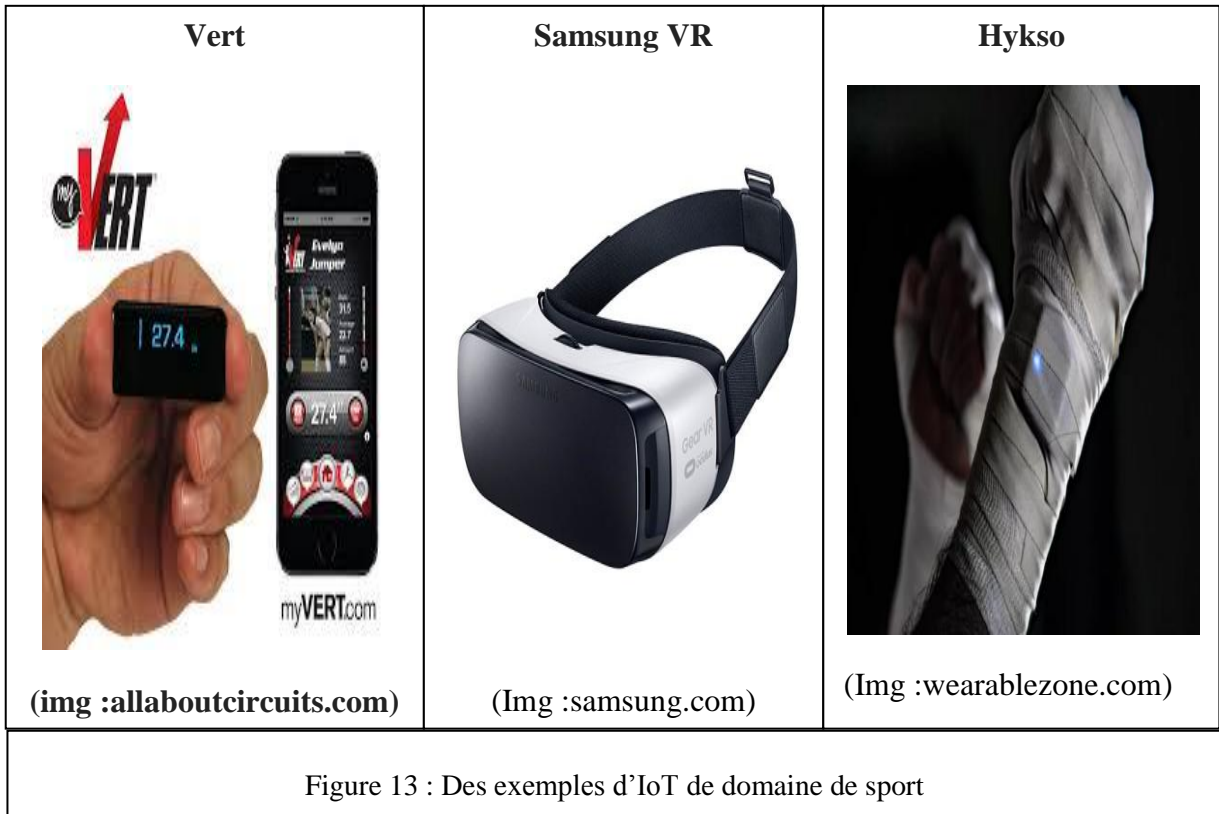
L'IoT ou l'Internet des objets est un concept qui est devenu de plus en plus excitant au fil des ans, avec la prédominance des périphériques intégrés compatibles avec la connectivité et prêts pour Internet. L'IoT est essentiellement un réseau d'objets physiques contenant une technologie intégrée qui peut détecter, traiter, agir et communiquer au sein du réseau et avec l'espace externe [8].

Quelques produits IoT récents dans le sport et le fitness vont maintenant être passés en revue. L'extension que ce travail couvre sera justifiée par les limitations liées aux produits existants et au système existant qui constitue la base de ce travail [7].

Les données pouvant être récoltées dans l'IoT dépendent des capteurs présents. Par conséquent, les applications des dispositifs IoT sont généralement basées sur l'interaction possible des processus avec les capteurs spécifiques présents [46]. L'Internet des objets trouve de plus en plus d'applications dans le sport et le fitness. Un appareil intelligent embarqué avec des accéléromètres, des gyroscopes, des magnétomètres et des capteurs de pression, par exemple sur le poignet d'un athlète, peut aider à compter les pas, à suivre les calories brûlées et à surveiller le rythme cardiaque. Dans le domaine de la santé et du bien-être, l'IoT permet la mise en œuvre d'ordinateurs portables et de correctifs pour la gestion des maladies. Tirer parti de l'approche latérale de l'IoT garantit également que les soins médicaux destinés aux personnes âgées et à la population en général seront moins chers et plus faciles à gérer avec ces nouveaux développements [47, 48, 49].

Actuellement, la plupart des fabricants de produits électroniques font d'énormes investissements pour être à la pointe de ce marché émergent. De plus en plus de capteurs sont raccordés aux engins et équipements de sport [7].

La figure suivante représentée des exemples d'IoT de domaine de sport [7] :



Conclusion

Dans ce chapitre nous avons défini l'internet des objets « internet of things (IoT) ». Nous avons également défini c'est quoi une ressource, nous avons parlé des différents architectures du l'internet des objets , nous avons rappelé les domaines d'application de l'internet des objet : l'automobile / les transports, l'automatisation industrielle, domaine de sport , domaine de santé « healthcare », agriculture, domaine d'IoT dans les villes intelligentes « smart cities », la maison connectée/intelligente « Connected/Smart home », Gestion de l'énergie « Energy Management » , dans le commerce de détail « Retail » .

Dans le chapitre suivant nous parlerons des services web, architectures, types et leurs technologies.

Chapitre 02 :

Services Web

1. Introduction

Les services Web sont des applications modulaires, auto-descriptives, autonomes, accessibles via Internet [91]. Le World Wide Web Consortium (W3C) [89] définit le service Web comme étant : « Un Web Service est un composant logiciel identifié par une URI, dont les interfaces publiques sont définies et appelées en XML. Sa définition peut être découverte par d'autres systèmes logiciels. Les services Web peuvent interagir entre eux d'une manière prescrite par leurs définitions, en utilisant des messages XML portés par les protocoles Internet ». Le web service utilise deux principales architectures SOAP_based (SOAP based web service) et REST_based (RESTful web service). Le SOAP_based utilise le protocole SOAP pour communiquer entre l'interface client et le fournisseur, Le REST_based proposé par Fielding en 2000[90] nommé REpresentational State Transfer. Il y a trois technologies principales de services web SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language), et UDDI (Universal Description, Discovery, and Interoperability).

Dans ce chapitre nous commençons par la définition de web service. En suite, nous avons parlé sur l'architecture de services web, la pile, les types, les différentes technologies de services web et les caractéristiques.

2. Services web :

2.1 Définition :

«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» [78].

Un service Web est une entité logicielle informatique qui est capable d'atteindre l'objectif de l'utilisateur par une invocation à distance. Les services Web permettent aux applications écrites dans différentes programmations langues pour interagir de manière transparente à travers des protocoles standards [57,58].

Le consortium W3C (World Wide Web Consortium) définit un service Web comme étant : « une application, ou un composant logiciel qui vérifie les propriétés suivantes » [77] :

- Il est identifié par un URI.
- Ses interfaces et ses liens 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 en Utilisant des protocoles Internet standards.

Un service Web est une interface qui décrit un ensemble d'opérations accessibles au réseau via une messagerie XML normalisée. Un service Web est décrit à l'aide d'une notion XML formelle standard, appelée description de service. Il couvre tous les détails nécessaires pour interagir avec le service, y compris les formats de message (qui détaillent les opérations), les protocoles de transport et la localisation. L'interface masque les détails d'implémentation du service, ce qui lui permet d'être utilisé indépendamment de la plate-forme matérielle ou logicielle sur laquelle il est implémenté, ainsi que du langage de programmation dans lequel il est écrit. Cela permet et encourage les applications basées sur les services Web à être couplées de manière lâche, orientées composant, implémentations multi-technologies. Les services Web remplissent une tâche spécifique ou un ensemble de tâches. Ils peuvent être utilisés seuls

ou avec d'autres services Web pour réaliser une agrégation complexe ou une transaction commerciale [59].

2.2 Architecture des services web (SOA) :

2.2.1 Définition Métier :

« 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. » [79].

2.2.2 Définition Technique

« L'architecture SOA est un paradigme permettant d'organiser et d'utiliser des savoir faire distribués pouvant être de domaines variés. Cela fournit un moyen uniforme d'offrir, de découvrir, d'interagir et d'utiliser des savoirs faire pour produire le résultat désiré avec des pré-conditions et des buts mesurables. » [80].

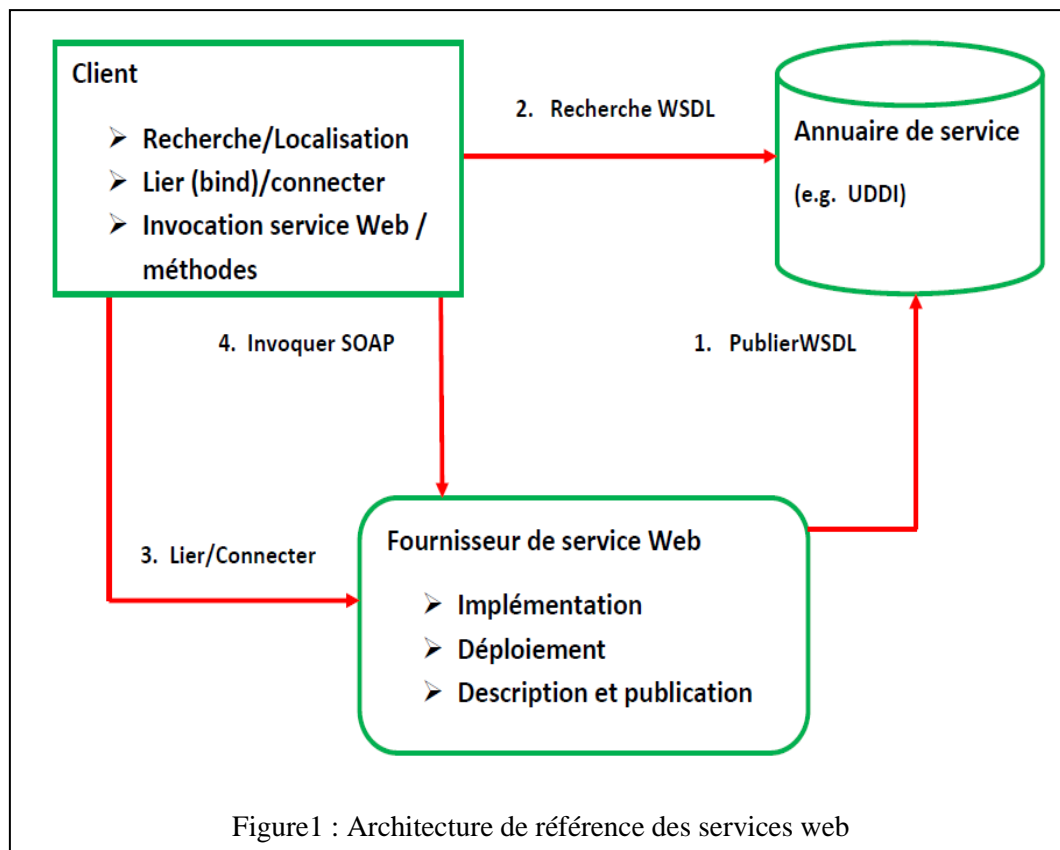
Du point de vue des applications, l'architecture orientée services permet le développement d'une nouvelle génération d'applications dynamiques ou composites. Ces applications permettent aux utilisateurs d'accéder à des informations et à des processus hétérogènes, et de les utiliser de différentes manières, notamment via le Web [81].

Du point de vue de l'infrastructure, l'architecture orientée services permet au service Web de simplifier l'intégration des applications et des systèmes, de recombinaison et de réutiliser les fonctionnalités des applications, et d'organiser les différentes phases du processus de développement, dans un cadre cohérent et unifié. En réalité, la philosophie des SOA décompose une application monolithique en une suite de services assurant la modularité dans leurs fonctionnalités [81].

L'architecture de services web appelé architecture de référence contient trois couches principale. Cette architecture vise trois objectifs importants [82] :

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

L'architecture de référence s'articule autour des trois rôles suivants [82]:



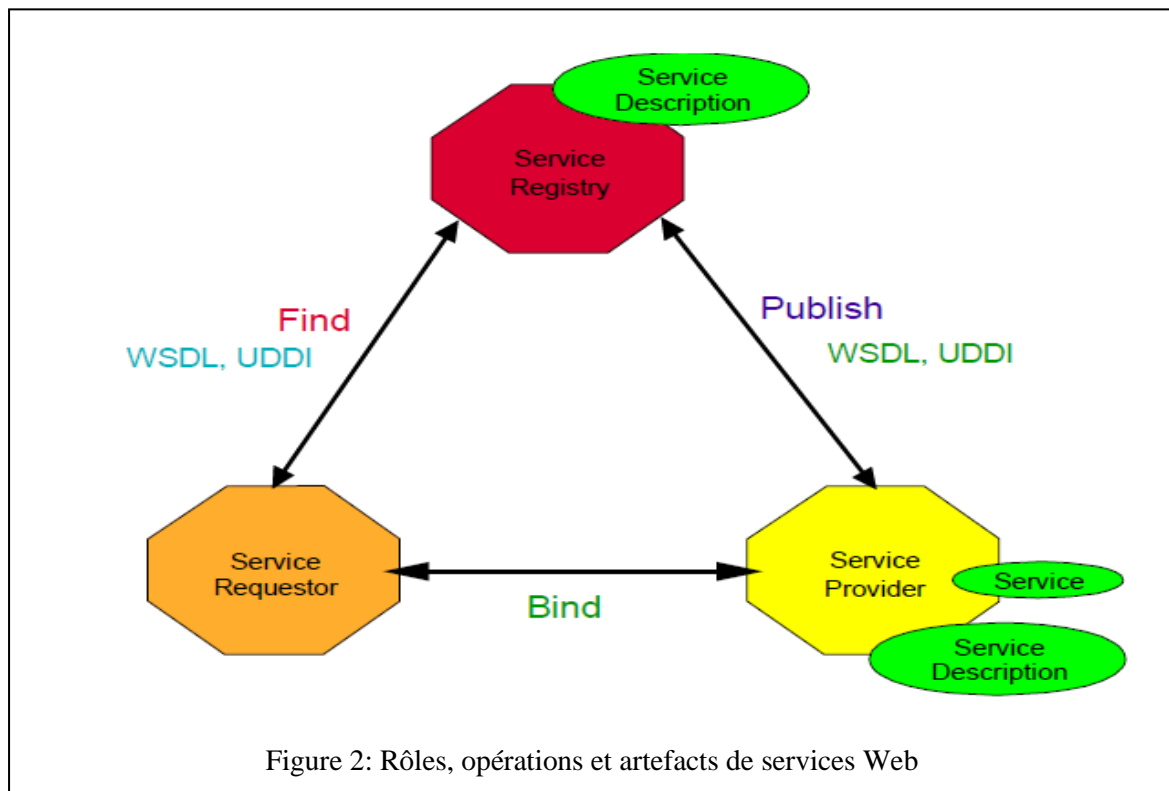
Le fournisseur de service : c'est le propriétaire du service Web. D'un point de vue technique, il est formé par la plateforme d'accueil du service Web, son rôle et la publication du service web à l'aide de formats standardisés (XML, WSDL-web service description language).

Le client : c'est lui qui va demander un service Web. D'un point de vue technique, il est composé par l'application qui va rechercher et sélectionner un service Web. L'application cliente peut être elle-même un service Web.

L'annuaire des services : Correspond à un registre de descriptions des services Web offrant des facilités de publication des services Web à l'intention des fournisseurs, ainsi que des facilités de recherche des services Web à l'intention des clients. Les interactions de base entre ces trois rôles incluent les opérations de publication, de recherche et de liens d'opérations.

2.3 Le modèle de services Web :

La figure suivante représentée les Rôles, les opérations et les artefacts de services Web[59] :



2.3.1 Les Rôles dans une architecture de services Web:[59]

- **Fournisseur de service:** du point de vue commercial, il s'agit du propriétaire du service. D'un point de vue architectural, il s'agit de la plate-forme qui héberge l'accès au service.
- **Demander de service:** d'un point de vue commercial, c'est l'activité qui exige que certaines fonctions soient remplies. D'un point de vue architectural, il s'agit de l'application qui recherche et appelle ou initie une interaction avec un service. Le rôle de demandeur de service peut être joué par un navigateur piloté par une personne ou un programme sans interface utilisateur, par exemple un autre service Web.
- **Registre de service:** il s'agit d'un registre consultable de descriptions de service dans lequel les fournisseurs de services publient leurs descriptions de service. Les demandeurs de service recherchent des services et obtiennent des informations de liaison (dans les descriptions de service) pour les services lors du développement pour la liaison statique ou lors de l'exécution

pour la liaison dynamique. Pour service statiquement lié demandeurs, le registre de services est un rôle facultatif dans l'architecture, car un fournisseur de services peut envoyer la description directement aux demandeurs de services. De même, les demandeurs de service peuvent obtenir une description du service auprès de sources autres qu'un registre de services, tel qu'un fichier local, un site FTP, un site Web, la publication et la découverte de services (ADS) ou la découverte de services Web (DISCO).

2.3.2 Opérations dans une architecture de services Web :

Pour qu'une application tire parti des services Web, trois comportements doivent avoir lieu: publication de descriptions de service, recherche ou recherche de descriptions de service et liaison ou appel de services sur la base de la description de service. Ces comportements peuvent se produire individuellement ou de manière itérative. En détail, ces opérations sont [59] :

- **Publier (publish):** pour être accessible, une description de service doit être publiée afin que le demandeur de service puisse la trouver. Le lieu où il est publié peut varier en fonction des exigences de l'application (voir «Service Publication» pour plus de détails).
- **trouver (find) :** dans l'opération de trouvez, le demandeur de service extrait directement une description de service ou interroge le registre de services pour connaître le type de service requis (voir «Découverte de services» pour plus de détails). L'opération de recherche peut être impliquée dans deux phases de cycle de vie différentes pour: le demandeur de service: au moment de la conception, pour extraire la description de l'interface du service pour le développement du programme, et au moment de l'exécution pour récupérer la liaison du service et la description de l'emplacement pour l'appel.
- **lier(Bind):** un service doit éventuellement être appelé. Dans l'opération de liaison, le demandeur de service appelle ou initie une interaction avec le service au moment de l'exécution en utilisant les détails de la liaison dans la description du service pour localiser, contacter et appeler le service.

2.3.3 Artéfacts d'un service Web :

- **Service:** lorsqu'un service Web est une interface décrite par une description de service, son implémentation est le service. Un service est un module logiciel déployé sur des plates-formes accessibles au réseau fournies par le fournisseur de services. Il existe pour être appelé ou pour

interagir avec un demandeur de service. Il peut également fonctionner en tant que demandeur, en utilisant d'autres services Web dans sa mise en œuvre [59].

• **Description du service:** la description du service contient les détails de l'interface et de la mise en œuvre du service. Cela inclut ses types de données, ses opérations, ses informations de liaison et son emplacement réseau. Cela pourrait également inclure la catégorisation et d'autres métadonnées pour faciliter la découverte et l'utilisation par les demandeurs de service. La description du service peut être publiée sur un demandeur de service ou sur un registre de services [59].

2.4 Les types de services Web :

Il y a deux types principaux de web service. Les services Web conformes à l'architecture SOA et utilisant le protocole SOAP pour communiquer entre l'interface client et le fournisseur sont appelés services Web SOAP. En 2000, **Fielding [90]** a proposé un nouveau style d'architecture pour les applications basées sur le réseau, appelé «REPresentational State Transfer (REST)». REST visait la généralisation des interfaces, l'évolutivité des interactions et le déploiement indépendant de composants logiciels. Services Web basés sur REST les principes sont appelés services Web RESTful [58] :

2.4.1 Basé sur SOAP(SOAP_based):

Les services Web basés sur SOAP sont conçus pour permettre des interactions de type RPC (remote procedure call) avec des systèmes distants. Dans ce style de conception, le fournisseur de services et les consommateurs potentiels doivent établir une compréhension commune de la syntaxe du service et des opérations qu'il offre. Chaque service Web basé sur SOAP possède sa propre interface unique et est décrit au moyen du langage WSDL (Web Services Description Language) [61]. Cette description est publiée dans un Universal public. Description Registre de découverte et d'intégration (UUDI). UDDI gère et gère les entrées de ces services Web et conserve une référence pour le fichier de description du service Web (document WSDL). XML est utilisé pour construire les blocs de base de la communication de services Web au moyen d'un protocole de messagerie XML, tel que SOAP (Simple Object Access Protocol) ou XML-RPC (XML-Remote Procedure Call). Les services Web basés sur SOAP n'exposent qu'un seul point de terminaison, par lequel les utilisateurs communiquent avec les fonctionnalités proposées. La force du protocole de messagerie SOAP vient de sa capacité à fonctionner dans des environnements hétérogènes et indépendamment de la plate-

forme sous-jacente. SOAP a un mécanisme de vérification de type rigide, par lequel SOAP effectue la plupart des vérifications de données standard. Les messages SOAP ne sont liés à aucun courant ni futur protocole de transport [58].

Les services Web basés sur SOAP ont plusieurs années de déploiement réussi au sein des entreprises. L'approche basée sur SOAP est fortement encouragée par les principaux éditeurs de logiciels qui proposent des solutions entièrement automatisées pour la migration des API existantes avec la génération de code SOAP. Bien que les services Web basés sur SOAP aient été largement adoptés par le secteur et pris en charge par presque tous les outils de développement, l'approche basée sur SOAP présente les limitations suivantes [60]:

- **Complexité:** Le déploiement d'un service basé sur SOAP nécessite une grande expérience du fait de la complexité de la pile de protocoles de services Web. De plus, la sérialisation et la désérialisation des demandes écrites en langues natives dans des messages SOAP est un processus fastidieux et gourmand en ressources, qui va à l'encontre des limites des périphériques mobiles.
- **Accessibilité et interface:** le service est exposé au public à l'aide d'une seule API de terminal. Par conséquent, toutes les fonctionnalités de service et les informations d'accès sont encapsulées dans le fichier de description de service. Par conséquent, toutes les opérations utilisent la méthode POST.
- **Interopérabilité:** chaque service Web a sa propre interface de service. Les informations de description sont uniques pour chaque service et sont exposées par un seul fichier WSDL. Une fois que le client a découvert le service, les informations de liaison incluses dans le fichier WSDL sont utilisées pour communiquer avec le service et de construire les requêtes. Chaque fois que ces liaisons changent, les communications et les demandes correspondantes doivent être modifiées en conséquence.
- **Performances:** un service Web basé sur SOAP est soumis à une surcharge. En raison de l'utilisation de XML et de la longueur des messages SOAP. De plus, le fichier WSDL et les messages SOAP incluent généralement des informations redondantes, ce qui augmente le trafic réseau et consomme plus de ressources.
- **Modèle de données:** l'approche basée sur SOAP masque le modèle de données derrière l'interface de service Web. Cette fonctionnalité impose au consommateur de services et au fournisseur de services de partager un modèle commun pour

communiquer. Cependant, les défenseurs de SOAP soutiennent qu'il est plus sûr et moins risqué de garder le modèle de données à l'écart des clients.

- **Évolutivité:** les messages SOAP sont interprétés uniquement en dehors du Web par différentes applications. Étant donné que les consommateurs et les fournisseurs doivent établir un terrain commun pour pouvoir communiquer, l'évolutivité est un problème car ils ne permettent pas une intégration adéquate avec le Web en tant que modèle d'information partagée.

2.4.2 Basé sur REST (REST_based):

Les services Web RESTful ont retenu l'attention de la communauté Web en raison de leur simplicité et de leur évolutivité. Les principaux fournisseurs de services Web tels que Google, Amazon, Yahoo et eBay ont adopté l'approche de services Web RESTful dans leurs services Web proposés. Les services Web RESTful [60] sont conformes aux concepts de REST afin d'éviter la dégradation des performances résultant de l'utilisation de SOAP et de XML. Les services conçus avec l'approche RESTful exposent leurs fonctionnalités en tant que ressources Web. Chaque ressource est adressée avec un URI unique. Un utilisateur peut accéder à la ressource souhaitée directement par l'URI associé ou par la fonctionnalité proposée via une structure hiérarchique. L'approche RESTful offre une grande flexibilité et évolutivité. Bien qu'elle soit étroitement associée au protocole HTTP, cette approche convient mieux aux domaines mobiles. Etudes d'évaluation de performance montrée que les services Web RESTful sont plus performantes que leurs homologues basés sur SOAP dans des environnements aux ressources limitées [58].

L'approche RESTful présente les avantages suivants par rapport à l'approche SOAP [58]:

- **Évolutivité:** L'approche RESTful hérite de l'évolutivité sous-jacente de HTTP.
- **Adressabilité :** les ressources (services) sont exposées et accessibles via un URI valide plutôt que de nécessiter un référentiel centralisé pour gérer la publication et la découverte. Chaque ressource a son propre URI unique, qui peut être récupéré pendant que l'utilisateur navigue via les connexions de lien entre les ressources.
- **Liens et connexions:** les ressources peuvent se lier les unes aux autres à l'aide d'hyperliens et le transfert d'état peut être géré via le renvoi aux liens.
- **Apatride:** Les demandes dans l'approche RESTful sont autonomes. Cette indépendance permet de supprimer les informations associées à une requête une fois

celle-ci effectuée. Les principes REST stipulent que les messages HTTP doivent être «auto-descriptifs», ce qui implique que tout nœud intermédiaire peut interpréter intégralement les messages, les comprendre et agir sur son contenu pour le compte de l'utilisateur.

- **Interface unifiée:** toutes les ressources sont gérées de manière dynamique par un ensemble limité de méthodes HTTP standard, à savoir GET, PUT, DELETE et POST. Tout client HTTP peut communiquer directement avec n'importe quel serveur HTTP sans autre configuration spéciale. En revanche, SOAP a besoin de client et de serveur (ou le consommateur et le fournisseur) doivent accepter les noms de méthodes, les types de données et le modèle d'adressage. La raison principale en est que SOAP est un Framework de protocole, alors que HTTP est un protocole d'application [57].

Le tableau suivant représente la COMPARAISON DES APPROCHES DE CONCEPTION DE SERVICES WEB [58] :

Caractéristiques (Feature)	SOAP-based	REST-based
Style d'architecture	Centré sur les services (Service-centric)	Centré sur les ressources (Resource-centric)
Couplage	Couplage étroit (Tightly coupled)	Couplage lâche (Loosely coupled)
Protocole de transport	Tout	HTTP seulement
Schéma d'accès	Seul point final (Single end-point)	URI pour chaque ressource
QoS (qualité de service)	Spécifications WS	Dépendant du transport (HTTP)
Invocation	RPC-like	Méthodes http
Interface	Interface pour chaque service Web	Navigateur web
La description	WSDL	Pas de norme
Modèle de données	Caché	Exposé
Représentation des données	XML	XML, JSON, etc.
L'évolutivité	Aucun	Hyperliens connectés
Sécurité	Basé sur WS-security	Basé sur HTTP

Table 1 : COMPARAISON DES APPROCHES DE CONCEPTION DE SERVICES WEB

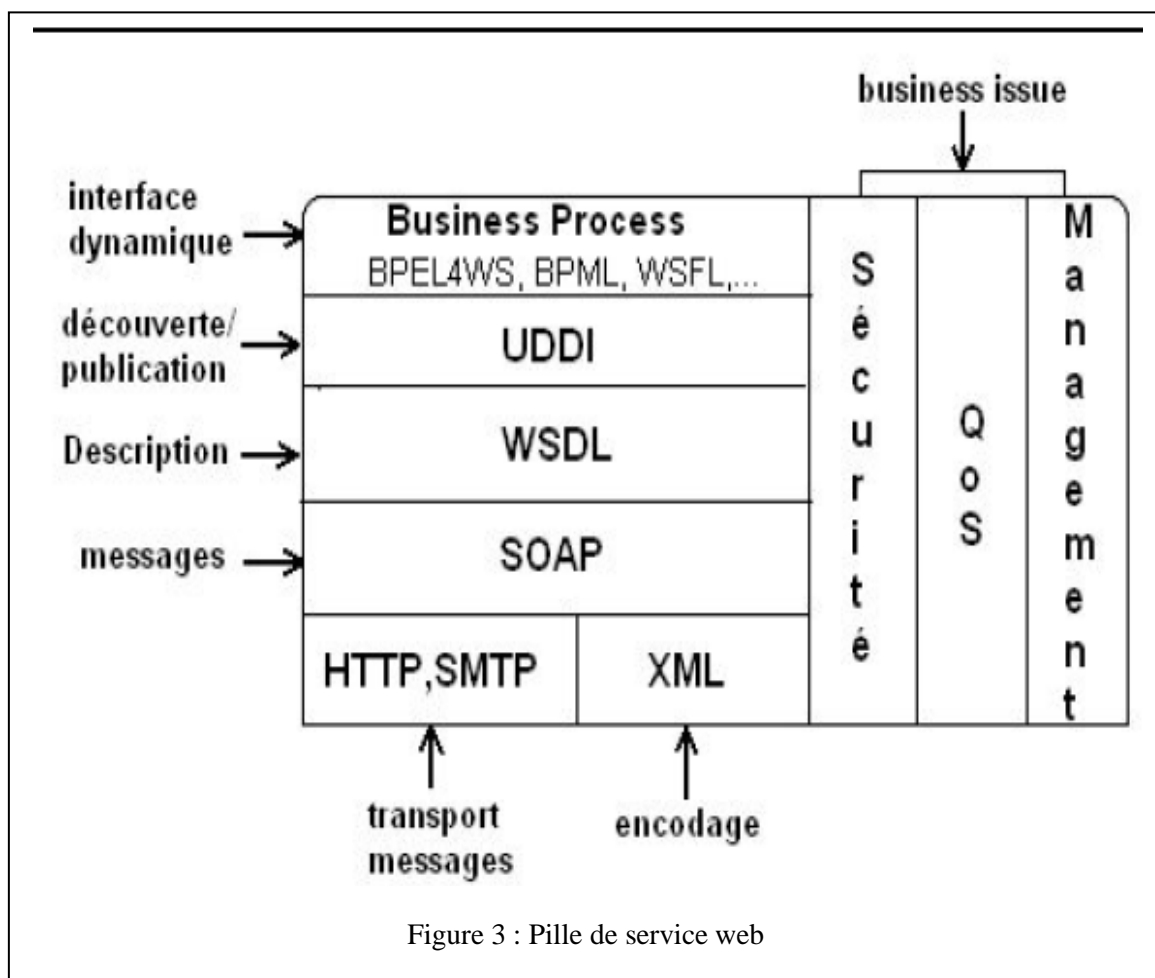
2.5 Pile de service web :

La pile est constituée de plusieurs couches, chaque couche s'appuyant sur un standard particulier. On retrouve, au-dessus de la couche de transport, trois couches s'appuyant sur les standards émergents SOAP, WSDL et UDDI.

L'infrastructure de Service Web de base définit les fondements techniques permettant de rendre les processus métiers (business processes) accessibles à l'intérieur d'une entreprise et au-delà même des frontières d'une entreprise.

Dans ce contexte deux types de couches permettent de la compléter : Les couches dites transversales (exemple: sécurité, administration, transactions et qualité de services (QoS)) qui rendent viable l'utilisation effective des Web-Services dans le monde industriel. Une couche Business processus qui permet l'utilisation effective des Web-Services dans le domaine du E-business [63].

La figure suivante représente la pile de service web [63] :

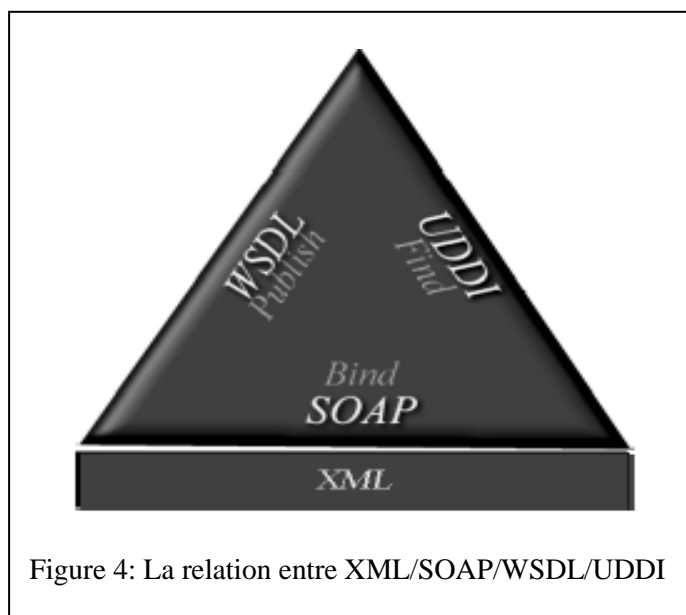


2.6 Les technologies de services web :

XML, SOAP, WSDL et UDDI sont les éléments fondamentaux pour déployer des infrastructures SOA basées sur des services Web [60].

- ✓ XML est la norme pour la représentation des données.
- ✓ SOAP spécifie la couche de transport pour envoyer des messages entre consommateurs et fournisseurs
- ✓ WSDL décrit les services Web.
- ✓ UDDI est utilisé pour enregistrer et rechercher des services Web.

Le figure suivant représenter la relation entre XML/SOAP/WSDL/UDDI [60] :



Les Services Web utilisant trois technologies fondamentales :

2.6.1 Protocole d'accès aux objets simple (SOAP) :

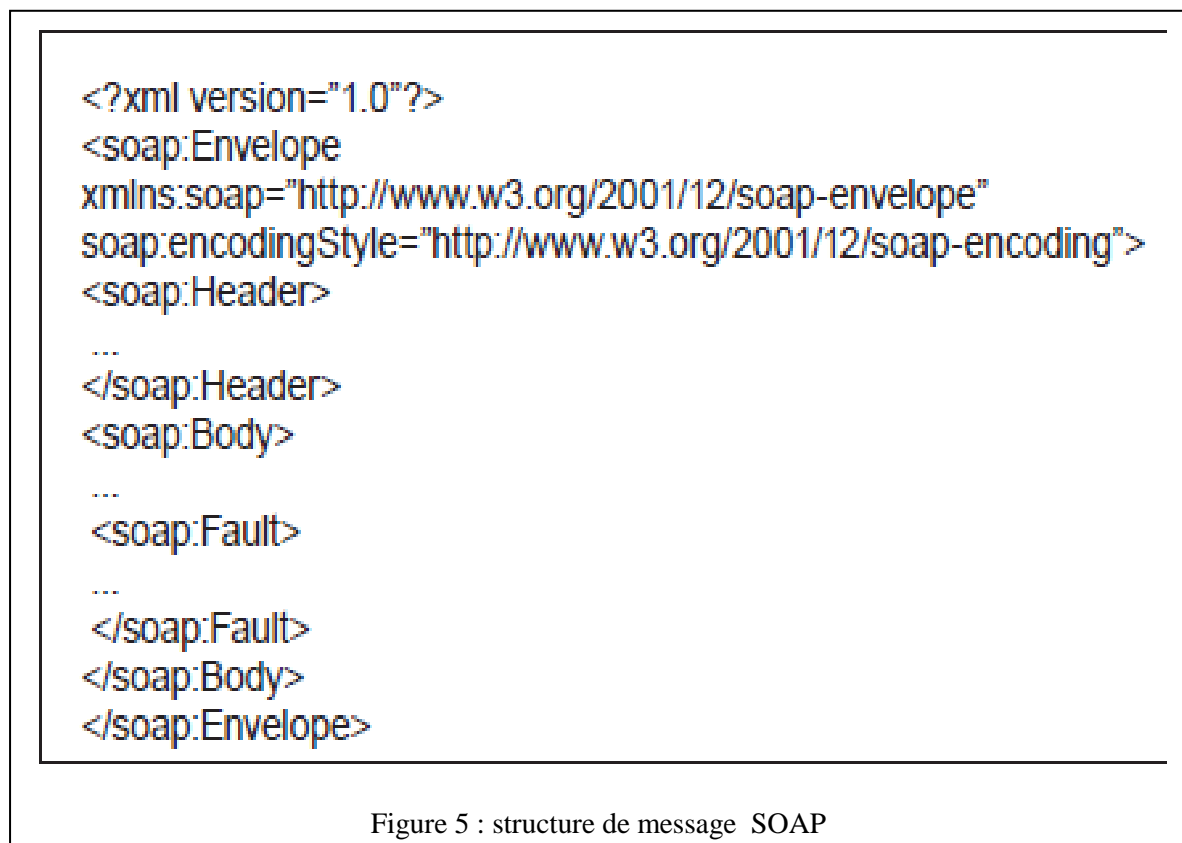
SOAP est un protocole de communication entre applications fondé sur XML, visant à satisfaire un double objectif : servir de protocole de communication sur les intranets, dans une optique d'intégration d'applications d'entreprise, et permettre la communication entre applications et Services Web, en particulier dans un contexte d'échanges inter entreprises. Conçu pour s'appuyer principalement sur HTTP, SOAP ne requiert pas de modification majeure aux serveurs Web déjà installés sur la Toile, tout au plus des extensions. SOAP permet également, grâce au support de XML Schéma, de s'interfacer avec des applications préexistantes en capturant leurs structures de données quelle que soit la complexité de ces dernières [61]. Ce protocole définit les types et les formats de messages XML pouvant être

échangés entre homologues dans un environnement décentralisé et distribué. L'un des principaux objectifs de SOAP est de créer un protocole de communication pouvant être utilisé par différentes applications développées à l'aide de différents langages de programmation, systèmes d'exploitation et plates-formes [60].

2.6.1.1 Structure du Message SOAP :

Les messages SOAP sont des messages XML définis dans une enveloppe SOAP. L'enveloppe SOAP contient l'en-tête et le corps SOAP. Tous les messages SOAP doivent avoir un corps, mais l'en-tête est facultatif. Le corps SOAP contient le message XML à transmettre. L'en-tête SOAP, si présent, contient généralement des informations qui sont importantes pour la sécurité, le routage ou le traitement approprié du destinataire du message [62].

La figure suivante représente la structure de message SOAP [60] :



Un message SOAP est un document XML qui doit avoir la forme suivante [64] :

- La structure de l'enveloppe SOAP qui définit une structure générale décrivant le contenu, le destinataire, et la nature du message.

- Les règles d'encodage SOAP qui définissent le mécanisme de sérialisation utilisé pour échanger des objets.
- SOAP RPC (SOAP remote-procedure calls) qui définit, pour les utilisations synchrones, une convention de représentation des appels et des réponses des procédures distantes.

La figure suivante représente les composants de service web :

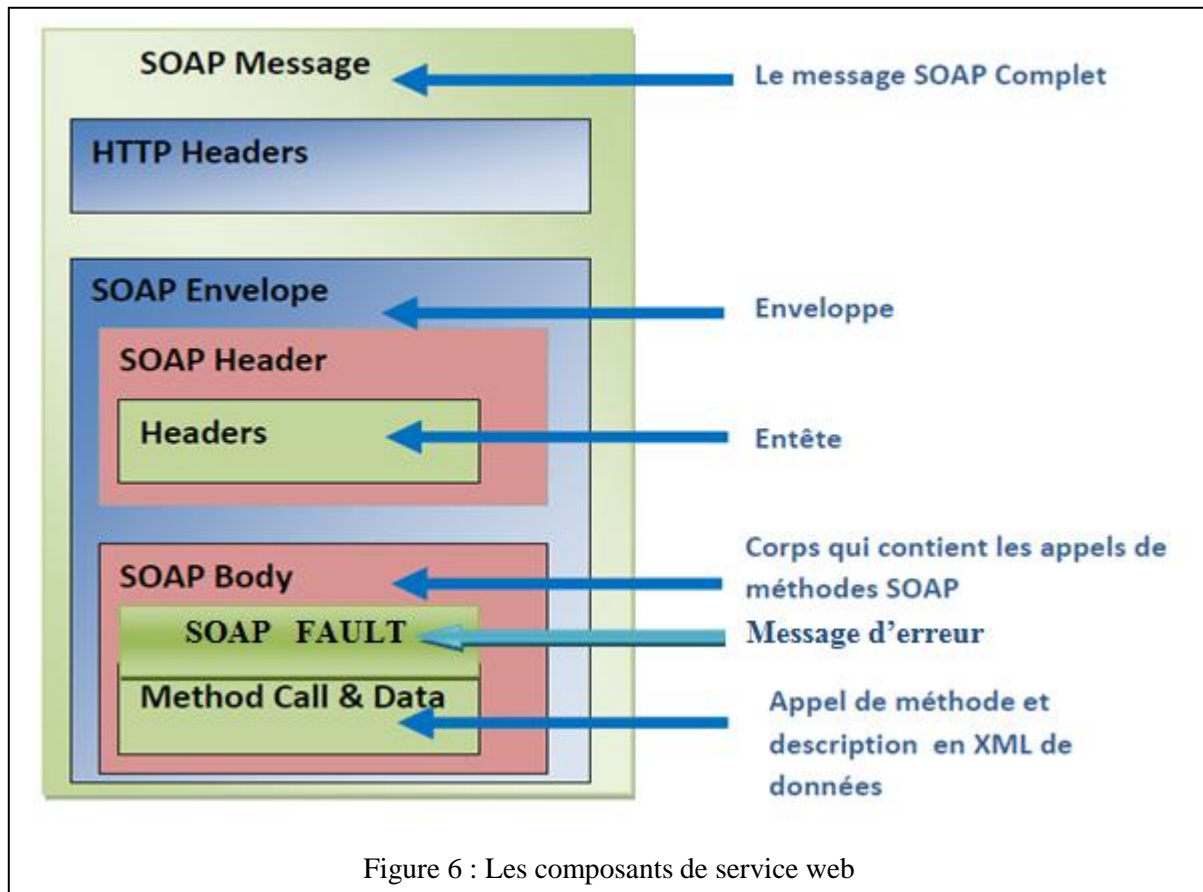


Figure 6 : Les composants de service web

2.6.2 Langage de description du service Web (WSDL) :

WSDL est le langage principal qui fournit un modèle et un format XML pour décrire les informations syntaxiques sur les services Web. C'est un langage XML standard du W3C permettant de spécifier l'interface d'un service Web. Cette norme permet de séparer la description de la fonctionnalité abstraite offerte par un service des détails concrets de son implémentation en définissant l'interface que les services Web fournissent aux demandeurs. La définition de l'interface (appelée type de port dans la version 1.x et appelée interface dans la version 2.0) donne les signatures pour toutes les opérations fournies, y compris le nom de l'opération, les entrées, les sorties et des défauts [60]. Conceptuellement similaire au langage

de définition d'interface (IDL) utilisé par l'architecture CORBA (Common Object Request Broker), WSDL décrit l'interface de programmation d'un service Web. La description comprend des détails tels que les définitions de type de données, les opérations prises en charge par le service, les formats de message d'entrée / de sortie, l'adresse réseau, les liaisons de protocole, etc [62].

2.6.2.1 Structure d'un document WSDL :

La figure suivante représentée la structure du document WSDL [63] :

```
< ?xml version="1.0" encoding="utf-8"? >
  <definitions >
    < types >  !-- Les types de données      </ types >
    <message >  !-- La structure du message  </message >
    <portType > !--L'interface d'accès au sw  </portType>
    <binding > ! -- Méthode d'accès          </binding >
    < service > ! -- Le fournisseur du service </service >
  </definitions>
```

Figure 7 : Structure d'un document WSDL

2.6.2.2 Les éléments principaux d'un document WSDL :

Le document WSDL décrit un service Web utilisant six composants principaux [58]:

<definitions> : C'est la racine de tout document WSDL, contient la définition du service. Cette balise contient les types d'éléments suivant:

- **<types>** / **</types>** est une définition de type de données XML qui décrit les conteneurs de données utilisés dans les échanges de messages. Le nom de l'élément n'a pas changé dans WSDL 2.0.
- **<message>/NA** est une représentation abstraite de l'information transmise. Généralement, un message contient une ou plusieurs parties logiques (paramètres).

Ces parties sont associées à une définition de type. Dans le squelette de WSDL 2.0, l'élément de message est supprimé en tant qu'élément global et la description des messages est encapsulée dans l'élément d'interface.

- **<portType>** / **<Interface>** est un composant important des documents WSDL, dans lequel un ensemble d'opérations abstraites (fonctions) pouvant être effectuées par le service Web est défini. Chaque opération est associée à un message d'entrée et / ou de sortie.
- **<binding>** / **<binding>** spécifie le protocole de communication et le format de données pour chaque opération et chaque message définis dans un élément portType / interface particulier.
- **<service>** / **<service>** est une opération composite qui regroupe plusieurs fonctions ou ports associés.

2.6.3 Description universelle, protocole de découverte et d'intégration (UDDI) :

Actuellement, les standards de l'industrie permettant d'enregistrer et de découvrir des services Web sont basés sur la spécification UDDI. Une fois qu'un service Web est développé, il doit être annoncé pour permettre la découverte. Le registre UDDI est supposé ouvrir la voie au succès de l'informatique orientée services, en exploitant la puissance d'Internet. Par conséquent, le mécanisme de découverte pris en charge doit être adapté à l'ampleur du Web en découvrant efficacement les services pertinents parmi des dizaines et des milliers (ou des millions selon les attentes du secteur) des services Web [60].

Le standard UDDI définit un service Web basé sur SOAP pour localiser les descriptions WSDL des services Web. Cette norme définit le contenu de l'information et le type d'accès fourni par les registres de services. Ces registres fournissent la publicité des services pouvant être invoqués par un client. UDDI peut stocker des descriptions sur les services Web internes d'une organisation et les services Web publics situés sur Internet [60].

L'UDDI est subdivisé en deux parties principales : partie publication ou inscription, et partie découverte. La partie publication regroupe l'ensemble des informations relatives aux entreprises et à leurs services. Ces informations sont introduites via une API d'enregistrement. La partie découverte facilite la recherche d'information contenue dans UDDI grâce à l'API SOAP [92].

UDDI est consultable de différentes manières [63] :

- **Les pages blanches** comprennent la liste des entreprises ainsi que des informations associées à ces dernières. ainsi, il contient des informations comme le nom de l'entreprise, ses coordonnées, la description de l'entreprise mais également l'ensemble des ses identifiants
- **Les pages jaunes** recensent les Web-Services de chacune des entreprises sous le standard WSDL.
- **Les pages vertes** fournissent des informations techniques précises sur les services fournis. Ces informations concernent les descriptions de services et de information de liaison ou encore les processus métiers associés.

UDDI Comporte cinq structures de données principales [63] :

- **BusinessEntity**: informations sur les entreprises qui offrent les services dans l'annuaire UDDI
- **BusinessService** : informations sur les services offerts par l'entreprise
- **BindingTemplate** : informations concernant le lieu d'hébergement du service
- **tModel** : informations concernant le mode d'accès au service
- **publisherAssertion** : assertions contractuelles entre partenaires dans le cadre des échanges d'exécution d'un service

2.7 Caractéristiques des services Web :

Plusieurs acteurs définissent les services Web par des caractéristiques technologiques distinctives, qui sont [77] :

2.7.1 Un service Web est une application logicielle qui est reconnue par un URI :

URI est la façon d'identifier un point de contenu sur le Web, que ce soit une page de texte, une vidéo, une image ou un programme. La forme la plus commune des URI est appelé Uniform Resource Locator (URL). Le service web est donc accessible en spécifiant son URI,

2.7.2 Capacité des interfaces et liaisons d'être publiées, localisées et invoquées via le langage XML :

Les principales tâches d'un service Web sont : La publication dans un registre, la localisation en interrogeant le registre qui l'héberge et l'invocation par un ou plusieurs services Web après sa localisation. Ces tâches sont réalisées via l'utilisation d'XML.

2.7.3 Capacité d'interagir avec les composantes des logiciels via des éléments XML avec l'utilisation des protocoles Internet standards :

Un service Web est créé pour être utilisé et interagir avec d'autres logiciels contrairement à une page Web, ou à une autre application qui n'utilise pas les services Web. C'est l'interopérabilité basée sur l'utilisation de l'XML et les protocoles Internet standards, par exemple, HTTP, SMTP, FTP, . . . etc.

2.7.4 Composante logicielle légèrement couplée à interaction dynamique :

Un service Web ayant un programme qui permet de l'invoquer est appelé consommateur de service Web. Le service Web et son consommateur sont indépendants l'un de l'autre. Si une modification est à faire sur le consommateur, on n'a pas besoin de connaître la machine, le langage de programmation, le système d'exploitation ou autre paramètre, afin d'établir à nouveau une communication entre le service Web et son consommateur. Le consommateur possède une fonctionnalité qui correspond à faire une localisation et une invocation sur le service Web, au moment de l'exécution du programme de service Web de manière automatique.

Conclusion

Dans ce chapitre nous avons défini les services web, leur architecture, le model de service web (les rôles, les artefacts et les opérations), en suite, nous avons parlé sur les types de services web (SOAP_based, REST_based), le pile de service web, ses technologies : SOAP (Simple Object Access Protocol), WSDL (Web Service Description Language), UDDI (Universal Description, Discovery, and Interoperability), et les caractéristiques des services web.

Dans le chapitre suivant nous allons présenter l'internet des objets "tout en tant que ressource", et nous allons expliquer comment nous avons appliqué ce paradigme dans le domaine sportif et précisément la natation

Chapitre 03 :

**Présentation du nouveau
paradigme "Tout en tant
que ressource" et
illustration à travers une
étude de cas**

1. Introduction

T.Baker et al. [9] ont proposé le nouveau paradigme "tout en tant que ressource" à travers l'internet des objets, ils présentent le Framework *aaR (couches applications, couche ressources et couche infrastructure), les catégories des ressources (informatiques, consommés, produites), les restrictions et les méthodes de ressources.

Dans ce chapitre nous avons répondu aux trois questions suivantes: -1: qu'est ce que l'internet des objets? -2: Définir le tout en tant que ressource? -3: Comment nous avons appliqué ce paradigme dans le domaine sportif et précisément la natation.

Ce chapitre est répartie en trois parties, la 1^{er} partie illustre l'internet des objets "tout en tant que ressource", dans la 2^{ème} partie nous avons parlé de l'anatomie humaine en rapport avec le domaine sportif et précisément la nation (Crawl, Dos, Brasse, Papillon) et dans la 3^{ème} partie nous appliquons notre vision des ressources dans le domaine sportif (natation).

Les objectifs de tout en tant que ressource selon T.Baker et al. [9] sont les suivants :

- ✓ développer un Framework * aaR fournissant les directives nécessaires à la création et à la gestion d'un écosystème de ressources.
- ✓ définir les ressources et les restrictions quelles que soient les applications à déployer sur cet écosystème
- ✓ illustrer la Framework *aaR avec une étude de cas sur l'Internet des objets (IoT) basée sur des ressources.

L'objectif de ce chapitre est d'appliquer le Framework *aaR dans le domaine de sport spécifiquement la natation pour illustrer tout en tant que ressource avec une étude de cas.

La 1ère partie :

2. Internet des objets (tout en tant que ressource)

2.1 Présentation du « Framework *aaR»

Pour représenter le Framework *aaR, on utilise trois couches qui permettent de distinguer les différents blocs et modules et de limiter les tâches particulières de ces blocs / modules dans chaque couche. Les couches sont application, ressources et infrastructures [9].

La figure suivante présente le Framework (*aaR) [9] :

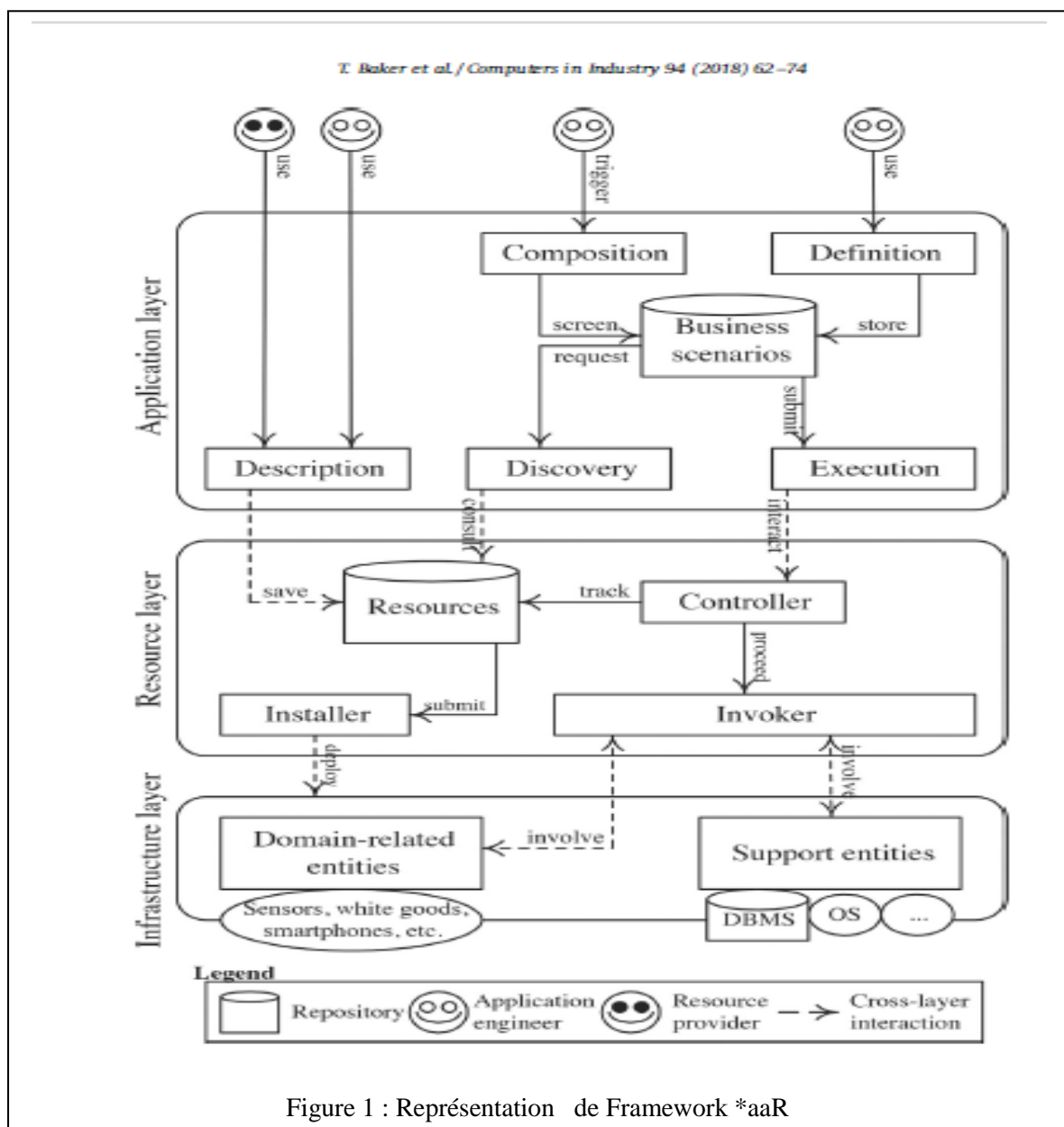


Figure 1 : Représentation de Framework *aaR

2.1.1 La couche application [9] :

La couche d'application se situe au sommet du Framework *aaR comme une interface entre toutes les ressources et tout le potentiel partie prenante qui mettra en place l'écosystème et / ou configurera (c'est-à-dire en ajustant si nécessaire) l'écosystème au moment de développer une application basée sur les ressources. Le Framework identifie deux catégories de parties prenantes: les fournisseurs de ressources liés à l'écosystème et les ingénieurs d'application liés aux applications (IoT).

Du point de vue de l'écosystème, il n'y a pas de bâtiment blocs à signaler dans la couche d'application. Au contraire, il y a un module d'opération qui est la description. Ce module aide les fournisseurs et les ingénieurs définissent les ressources pour écosystème et applications, respectivement.

Du point de vue de l'application, il y a un bloc de construction c'est le référentiel des scénarios d'entreprise. Ces scénarios guident le développement d'applications basées sur les ressources en termes de quelles ressources sont nécessaires et comment rassembler les ressources nécessaires. Les modules d'opération incluent la définition, la composition, la découverte et l'exécution. Le module de définition prend en charge les ingénieurs qui spécifient les scénarios de gestion stockés dans le référentiel approprié. Le module de composition identifie (une ou plusieurs) ressources selon des scénarios de gestion et avec l'aide du module de découverte qui consulte le référentiel de ressources de la couche ressources.

Enfin et surtout, le module d'exécution implémente les scénarios de gestion conformément à la demande du module de composition et une fois que les ressources sont confirmées pour une implication dans ces scénarios. Confirmer, c'est satisfaire les restrictions.

2.1.2 La couche ressources [9] :

La couche de ressources se situe entre la couche d'infrastructure et la couche application, aidant les parties prenantes à déployer des ressources sur l'infrastructure (à l'avantage des fournisseurs et des ingénieurs) conformément à la demande du module de description de la couche application ou à confirmer les ressources pour les scénarios de gestion (au bénéfice des ingénieurs) selon la demande du module de découverte de la couche application.

Du point de vue de l'écosystème, il existe un bloc de construction qui est le référentiel de ressources. Il contient la description des ressources de l'écosystème et des applications, ainsi que leur cycle de vie et restrictions. Il y a un module d'opération qui est l'installateur. Il déploie des ressources sur la couche infrastructure en les mappant sur des entités concrètes spécifiques dans la ressource couche.

Du point de vue de l'application, il n'y a pas de blocs bâtiment à signaler dans cette couche. Au contraire, les modules d'opération incluent contrôleur et invocateur. Le module de contrôle suit un cycle de vie de la ressource en fonction des changements / événements qui se produisent / surviennent dans l'écosystème et / ou les applications. Ce module aussi permet aux parties prenantes du Framework *aaR (à savoir, les fournisseurs et ingénieurs) pour appliquer des restrictions sur les ressources au moment de l'exécution. Le module invocateur implique des entités dans l'exécution du processus métier lors de l'approbation du module contrôleur.

2.1.3 La couche infrastructure [9] :

La couche infrastructure est au bas du Framework * aaR hébergeant deux types d'entités: support et domaine.

Du point de vue de l'écosystème, les entités de support agissent pour le compte de composants matériels (par exemple, serveurs) et logiciels (par exemple, OS) associés aux blocs fonctionnels et aux modules d'exploitation de l'écosystème / des applications.

Du point de vue de l'application, les entités liées au domaine agissent pour le compte de matériel spécialisé (capteurs, par exemple) et de logiciels (par exemple, régulateur de température) composants associés aux futures applications basées sur les ressources.

2.1.4 Les interactions entre couches :

Différentes interactions entre couches sont représentées dans la Figure 1 en utilisant des lignes en pointillées et sont résumées ci-dessous [9] :

- ✓ Le module de description de la couche d'application interagit avec le référentiel de ressources de la couche ressources afin de sauvegarder les descripteurs, les cycles de vie et les restrictions des ressources.

-
- ✓ Le module de découverte de la couche application interagit avec le référentiel de ressources de la couche de ressources afin de consulter les ressources nécessaires par rapport aux besoins des entreprises.
 - ✓ Le module d'exécution de la couche application interagit avec le contrôleur de la couche ressources pour s'assurer que les restrictions sur les ressources sont satisfaites avant d'utiliser ces ressources pendant l'exécution du processus métier.
 - ✓ Le module d'installation de la couche ressources interagit avec la couche infrastructure afin de déployer des supports / domaines liés entités associées aux composants matériels et logiciels de l'écosystème / des applications.
 - ✓ Le module invocateur de la couche ressources interagit avec la couche infrastructure afin d'impliquer le matériel approprié et composants logiciels, à travers leurs entités respectives, lors de l'exploitation de l'écosystème et / ou des applications.

3. Illustration de l'internet des objets, tout en tant que ressource:

3.1 Les catégories des ressources :

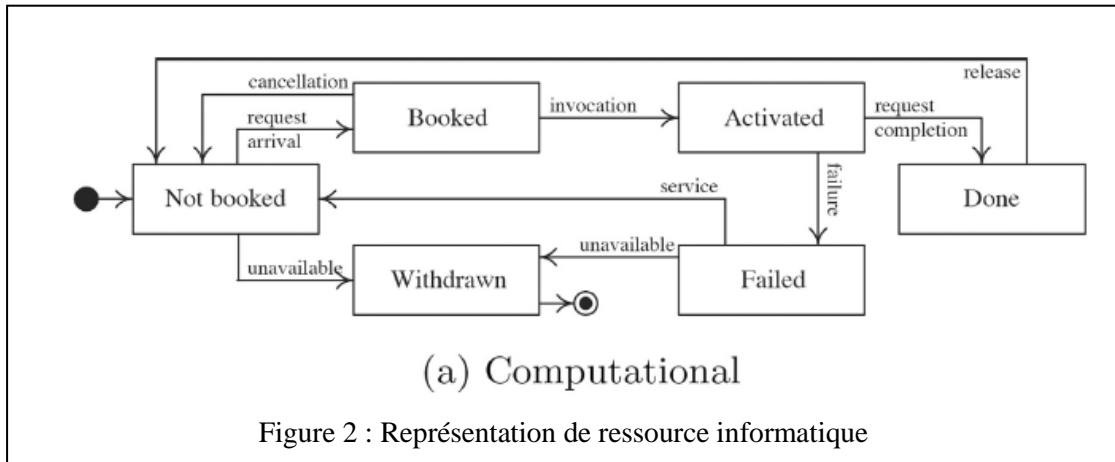
Les ressources divisées en trois classes : ressources informatiques, consommées et produites [9].

Une ressource informatique est définie comme un logiciel programme qui traite les intrants et produit des extrants, bien que les deux ne sont pas obligatoires (par exemple, un régulateur recevant la température et en ajustant le chauffage en conséquence). Nous définissons également une ressource consommée / produite en tant qu'entrée (s) / sortie (s) pouvant être liée à une ressource informatique au moment de l'exécution (par exemple, la température de la pièce produite par un capteur est consommée par un régulateur) [9].

Pour une meilleure compréhension de la manière dont les ressources sont utilisées et Pour opérer à la fois dans l'écosystème et dans les applications basées sur les ressources, nous définissons leurs comportements à l'aide de diagrammes d'état. Le comportement est un cycle de vie qui indique les états admissibles qu'une ressource emporte avec les chevauchements possibles entre les ressources (par exemple, une ressource produite pourrait devenir une ressource consommée) [9].

3.1.1 Ressources informatiques « computational » :

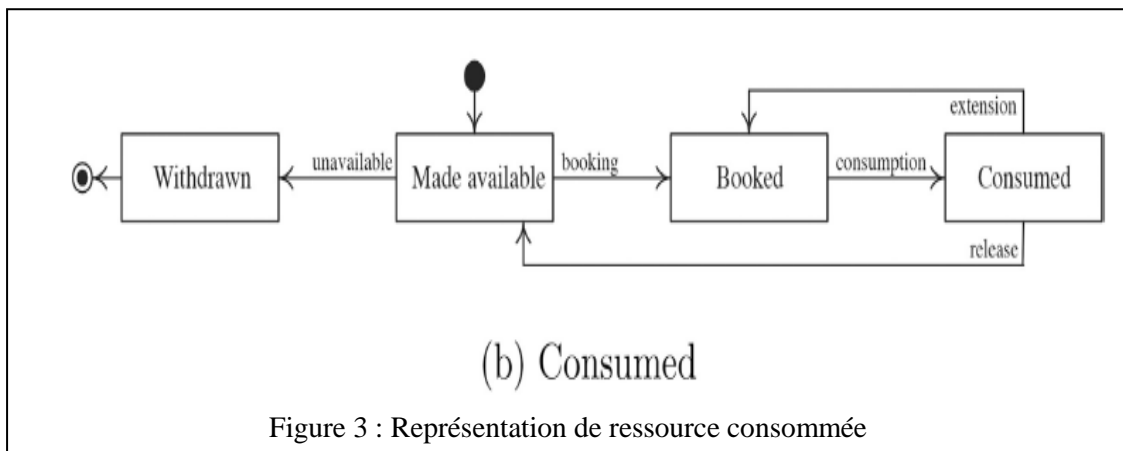
La figure suivante présente les ressources informatiques : [9]



- Les états d'une ressource informatique incluent non réservé « not-booked » (c'est-à-dire, inactif), réservé « booked » (c'est-à-dire dont l'activation est confirmée), activé « activated » (c'est-à-dire en cours d'activation), terminé « done » (c'est-à-dire une activation réussie), a échoué « failed » (c'est-à-dire une activation infructueuse) et retiré « withdrawn » (c'est-à-dire qu'ils ne sont plus disponibles). Certaines transitions incluent l'invocation réservée activée et l'échec indisponible retiré. [9]

3.1.2 Ressources consommés « consumed » :

La figure suivante présente les ressources consommées [9] :

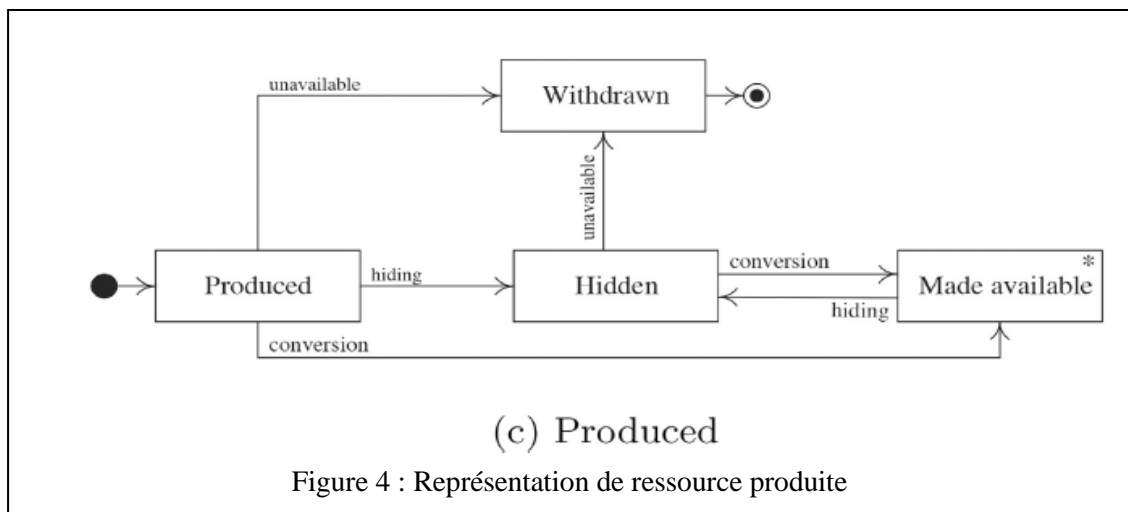


- Les états d'une ressource consommée incluent les éléments mis à disposition «made-available » (c'est-à-dire annoncé pour une consommation possible), réservé « booked » (c'est-à-dire confirmé pour la consommation par une ressource

informatique), consommé « consumed » (c'est-à-dire sous la consommation d'une ressource informatique), et retiré « withdrawn » (c'est-à-dire qu'elle n'est plus disponible). Certaines transitions incluent l'extension consommée réservée et la consommation réservée [9].

3.1.3 Ressources produites « produced » :

La figure suivante présente les ressources produites [9] :

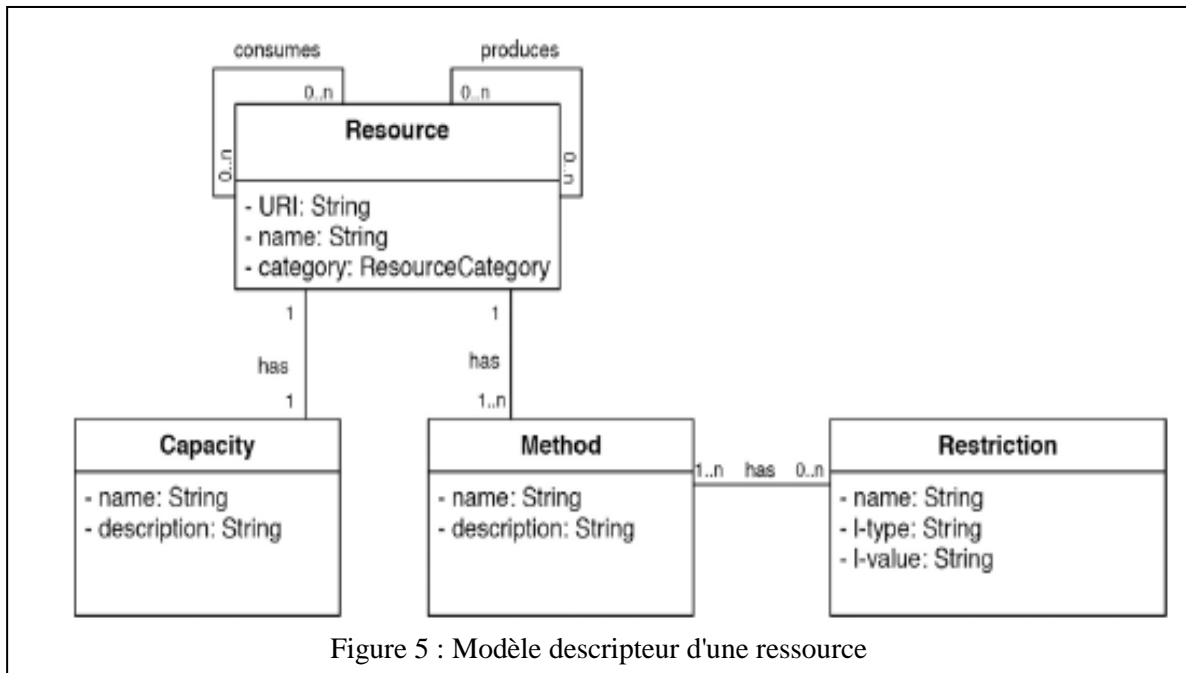


- Les états d'une ressource produite incluent « produced » (c'est-à-dire résultant de l'activation d'une ressource informatique), masqué « hidden » (c.-à-d. indisponible temporairement), mis à disposition « made-available » (c.-à-d. annoncé pour une consommation éventuelle), et retiré « withdrawn » (c'est-à-dire qu'elles ne sont plus disponibles). Certaines transitions incluent la dissimulation produite cachée et cachée indisponible retiré. [9]

3.2 Descripteur de ressources :

T. Baker et al. [9]. Proposent un modèle de descripteur de ressources (modèle non-UML) et utilisent certaines terminologies XML lors de la discussion de ce modèle. L'utilisation de ces terminologies facilite également la conversion du modèle en document XML.

La figure suivante présente le descripteur d'une ressource [9] :



Une ressource est composée d'un identifiant unique (par exemple, URI), d'un nom, d'une catégorie et de deux relations récursives (c'est-à-dire «consommée» et «produit») qui font référence aux connexions potentielles entre les ressources. De plus, une ressource a deux éléments: la capacité et la méthode. Chaque élément consiste en un nom et une description. Les descriptions multiples offrent un contenu lisible par l'homme qui permet de faciliter la recherche des ressources nécessaires dans le référentiel de ressources. Dans le modèle de descripteur, certaines méthodes sont associées à des restrictions consistant en un nom, un type et une valeur [9].

3.2.1 Les méthodes utilisées pour les ressources produites/ consommées/ informatiques :

3.2.1.1 Ressources informatiques [9]:

- ✓ Book (trs: request_arrival) permet de planifier l'invocation d'une ressource informatique.
- ✓ Activer (trs: invocation) permet d'appeler (lier) une ressource informatique par une autre ressource informatique ou par une application tierce.
- ✓ Release (trs: release) permet de rendre une ressource informatique disponible pour les activations à venir.

-
- ✓ Annuler « Cancel » (trs: annulation) permet de lâcher la réservation d'une ressource informatique après confirmation de la réservation.
 - ✓ Retirer « withdrawn » (trs: indisponible « unavailable ») permet de rendre une ressource informatique non disponible pour l'invocation.
 - ✓ Le service (trs: service) permet de maintenir et / ou réparer une ressource informatique dans le cadre de stratégies préventives et correctives.

La maintenance peut également concerner les méthodes de révision et / ou des restrictions en réponse à certains changements (par exemple, nouveaux frais d'activation).

3.2.1.2 Ressources consommées [9]:

- ✓ L'offre « Offer » (trs: disponible « available ») permet de rendre une ressource consommée disponible pour les ressources informatiques.
- ✓ Consommer « consume » (trs: consommation « consumption ») permet à une ressource informatique d'utiliser une ressource consommée.
- ✓ Book (trs: booking) permet à une ressource informatique de demander l'utilisation d'une ressource consommée. Cela aide la ressource consommée à répondre aux demandes de réservation d'autres ressources informatiques.
- ✓ annuler « Cancel » (trs: annulation « cancellation ») permet à une ressource informatique d'abandonner la demande d'utilisation d'une ressource consommée. Cela aide la ressource consommée à répondre à d'autres ressources de calcul « demandes de réservation ».
- ✓ Release (trs: release) permet à une ressource informatique de rendre une ressource consommée disponible après utilisation par d'autres ressources informatiques. Cela permet à la ressource de répondre aux demandes de réservation d'autres ressources informatiques.
- ✓ Extend (trs: extension) permet à une ressource informatique d'ajuster la réservation d'une ressource consommée en raison de contraintes de temps et / ou informatiques (par exemple, la date d'expiration).
- ✓ Retirer « Withdraw » (trs: indisponible « unavailable ») permet de rendre une ressource consommée indisponible pour les ressources de calcul. Cette indisponibilité pourrait être utilisée pour réviser des méthodes et / ou des restrictions en réponse à certains changements.

3.2.1.3 Ressources produites [9]:

- ✓ Convertir « convert » (trs: conversion) permet de rendre une ressource produite disponible pour d'autres ressources informatiques. La ressource produite sera traitée comme une ressource consommée.
- ✓ Cacher « Hide » (trs: hiding) permet de suspendre la disponibilité d'un produit ; la temporalité des ressources avant de rendre cette ressource totalement non disponible ou disponible en tant que ressource consommée.
- ✓ Retirer « withdraw » (trs: indisponible « unavailable ») permet de rendre une ressource produite indisponible (par exemple, se retirer).

3.2.2 La capacité des ressources :

La capacité est une manière de décrire le rôle d'une ressource dans l'écosystème ou dans une application. La capacité consiste à traiter (par exemple, analyser un échantillon de sang) une ressource informatique, à faire un résumé (par exemple, stocker les données d'un capteur) pour une ressource produite et à capturer (par exemple, convertir un fichier texte en un document XML) pour une ressource consommée [9].

3.2.3 Restrictions sur les méthodes des ressources :

T.Baker et al. [9] Proposent les restrictions suivantes sur les méthodes de ressources avec l'hypothèse que, sauf indication contraire, une méthode est par défaut illimitée « unlimited » (ul) et / ou partageable « shareable » (s):

Limité « limited » (l) : lorsque les appels à une méthode sont soumis à un seuil (par exemple, un nombre maximal) et / ou à un laps de temps spécifique (par exemple, avant juin 2017). Le principal motif de limitation est d'assurer la performance et / ou la disponibilité des ressources. Limité pourrait être un peu assoupli grâce à la limite, mais renouvelable (lr), qui permet d'étendre les appels à une méthode en contrepartie d'une taxe, par exemple.

Non-partageable « non-shareable » (ns): lorsque les appels simultanés à une méthode ne sont pas autorisés, la coordination de ces appels devient nécessaire. La principale raison d'être non partageable est d'assurer la cohérence des ressources.

La figure suivante représente les restrictions aux méthodes de ressources [9] :

Resource category	Resource method	Restrictions (l,ns,lr)
Computational	activate	l:applicable ns:applicable lr:applicable
	release	l:not-applicable ns:not-applicable lr:not-applicable
	book	l:applicable ns:not-applicable lr:applicable
	cancel	l:not-applicable ns:not-applicable lr:not-applicable
Consumed	book	l:applicable ns:applicable lr:applicable
	cancel	l:not-applicable ns:not-applicable lr:not-applicable
	release	l:not-applicable ns:not-applicable lr:not-applicable
	extend	l:applicable ns:applicable lr:applicable
	consume	l:applicable ns:applicable lr:applicable

Tableau 1 : Application de restrictions aux méthodes des ressources.

La 2ème partie :

1. La natation :

La natation c'est le sport que j'ai choisi pour appliquer la vision de T.Baker et al. (tout en tant que ressource) dans le domaine de sport. Il y a quatre principaux types de natation : Brasse, Dos, Papillon et Crawl.

Selon **la fédération de natation française**, les muscles utilisés dans chaque type de natation sont :

1.1 Crawl :

1.1.1 Muscle du Crawl: les mouvements des bras :

Phase 1 Propulsive haut du corps [65] :

- ✓ Faisceau claviculaire du grand pectoral, grand dorsal
- ✓ Biceps et triceps brachial, fléchisseurs du poignet
- ✓ Triceps brachial

Phase 2 Dégagement haut du corps [65] :

- ✓ Dectoïdes et coiffe des rotateurs (supra et infra-épineux, petit rond et sub-scapulaire)
- ✓ Stabilisateurs omoplate (petit pectoral, rhomboïdes, élévateur de scapula, faisceaux moyens et inférieurs du trapèze, grand dentelé)

La figure suivante représente Crawl-Mouvements des bras [65] :

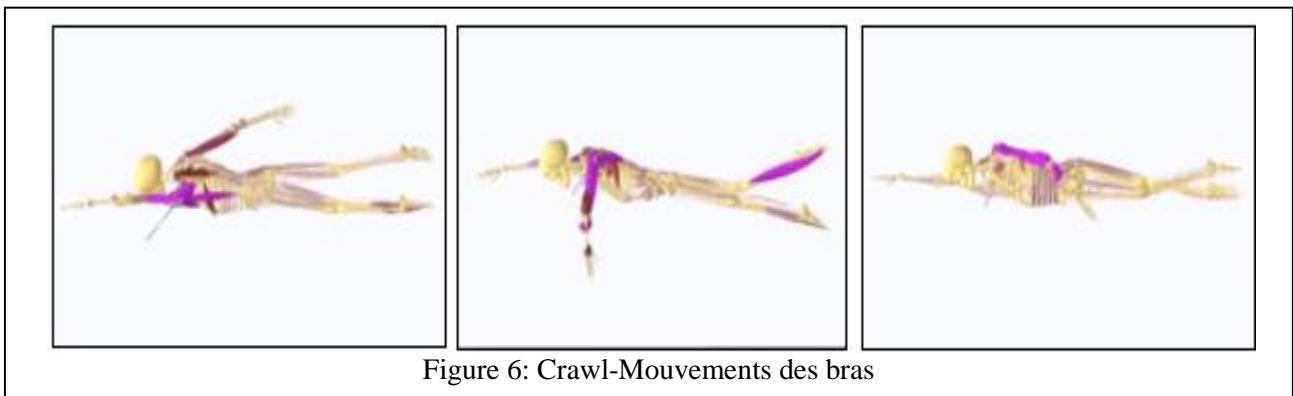


Figure 6: Crawl-Mouvements des bras

1.1.2 Muscle du Crawl : Mouvements des jambes

Phase Propulsive & Stabilisatrice bas du corps (battements) [65] :

- ✓ Phase descendante (Propulsive) : iliopsoas, droit de la cuisse, quadriceps
- ✓ Phase ascendante (Dégagement) : fessiers, ischio-jambiers
- ✓ Coulée
- ✓ 3 phases : gastrocnémien et soléaire (flexion plantaire)

La figure suivante représente Crawl-Mouvements des jambes [65] :

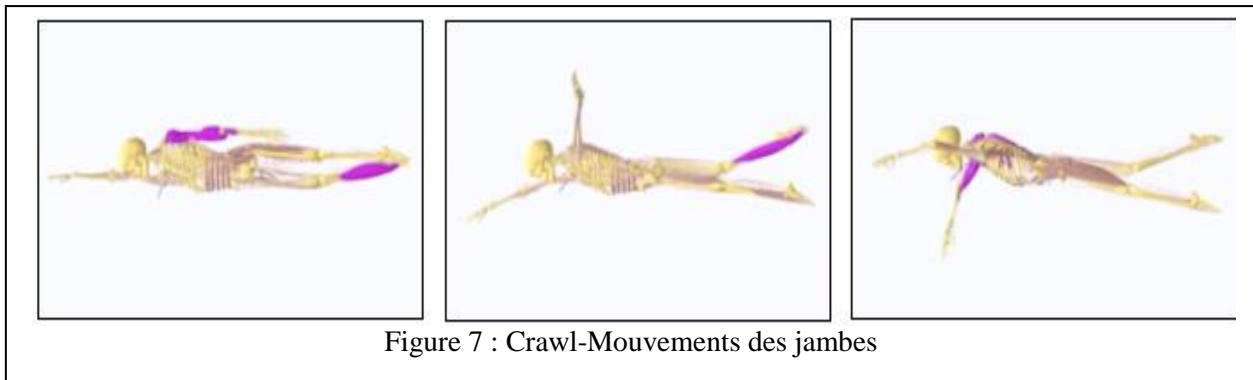


Figure 7 : Crawl-Mouvements des jambes

1.1.3 Muscle de crawl : Tronc

Phase Stabilisatrice du tronc [65] :

- ✓ Transverse de l'abdomen, droit de l'abdomen, obliques internes et externes, érecteurs du rachis

La figure suivante représente Crawl-Tronc [65] :

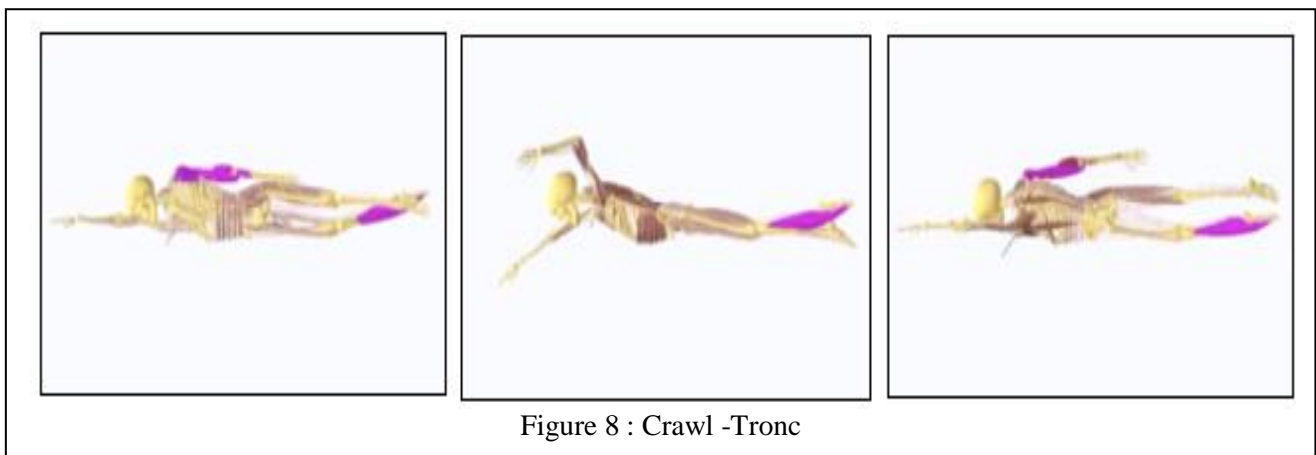


Figure 8 : Crawl -Tronc

1.2 Dos :

1.2.1 Muscle du Dos : les mouvements des bras :

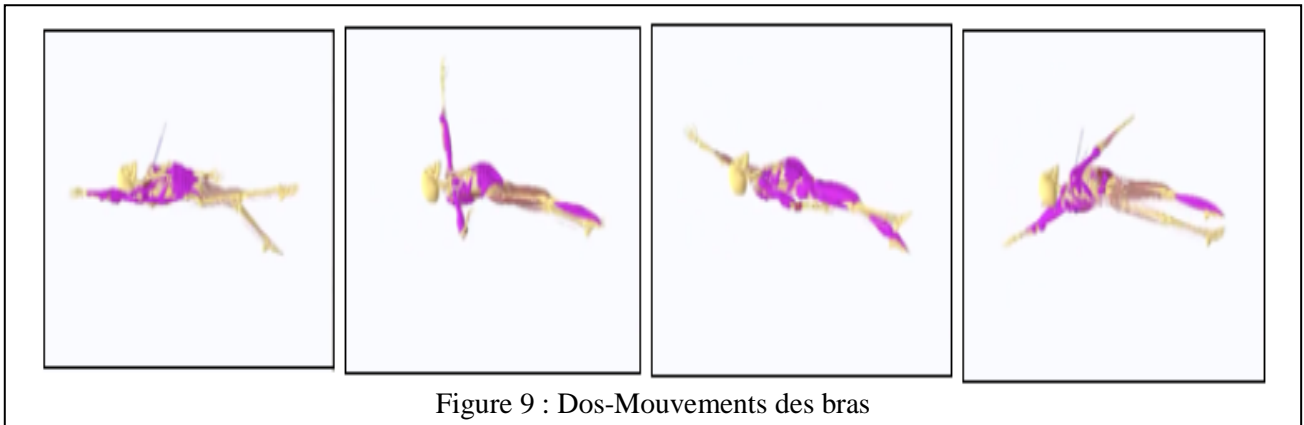
Phase 1 Propulsive haut du corps [65] :

- ✓ Grand dorsal ++, grand pectoral, triceps brachial
- ✓ Grand dorsal, Biceps brachial
- ✓ Triceps brachial

Phase 2 Dégagement haut du corps [65] :

- ✓ Deltoïdes et coiffe des rotateurs (supra et infra-épineux, petit rond et sub-scapulaire)

La figure suivante représente Dos-Mouvements des bras [65] :

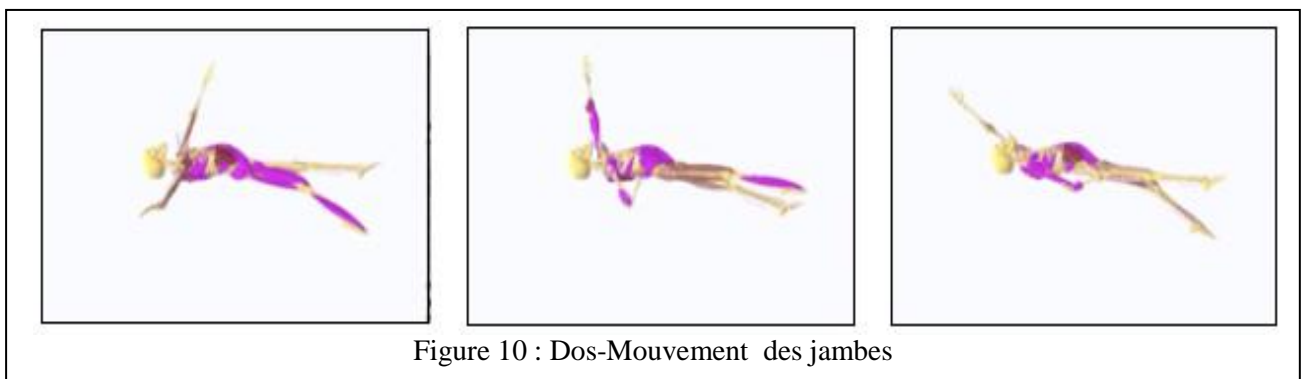


1.2.2 Muscle du Dos : Mouvements des jambes

Phase Propulsive bas du corps [65] :

- ✓ Phase dégagement (descendante) : illiopecteur, droit de la cuisse, quadriceps
- ✓ Phase propulsive (ascendante) : fessiers, ischio-jambiers
- ✓ Coulée
- ✓ 3 phases : gastrocnémien et solaire

La figure suivante représente les muscles du Dos : Mouvements des jambes Phase Propulsive [65] :

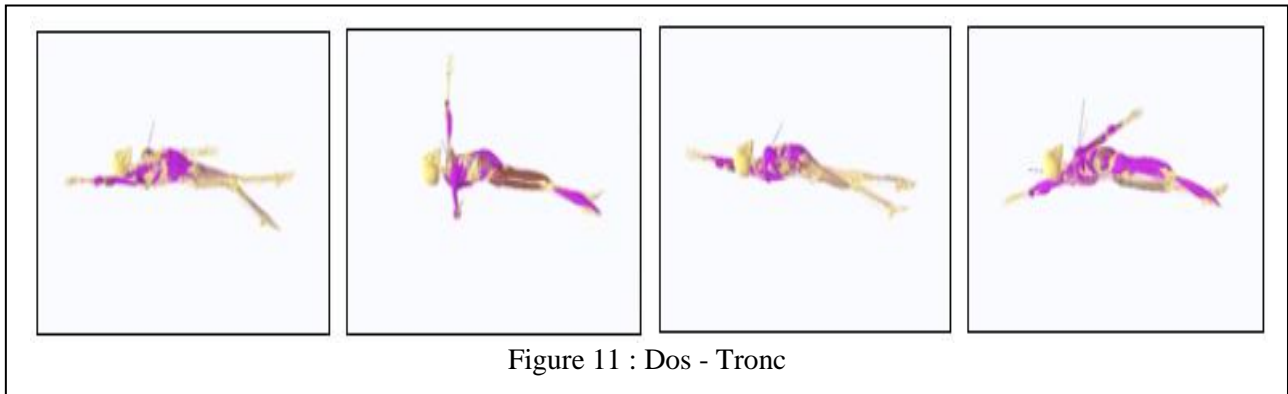


1.2.3 Muscle du Dos – Tronc

Phase Stabilisatrice du tronc [65] :

- ✓ Transverse de l'abdomen, droit de l'abdomen, obliques internes et externes, érecteurs du rachis

La figure suivante représente les muscles du Dos-Tronc [65] :



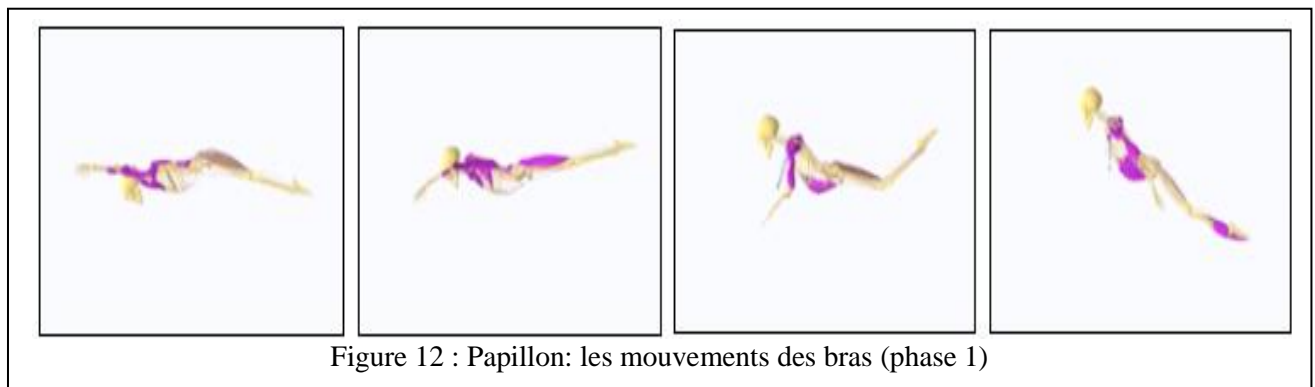
1.3 Papillon :

1.3.1 Muscle du Papillon: les mouvements des bras

Phase 1 Propulsive haut du corps [65] :

- ✓ Faisceau claviculaire du grand pectoral, grand dorsal
- ✓ Biceps et triceps brachial, fléchisseurs du poignet
- ✓ Triceps brachial++ (phase finale traction)

La figure suivante représente les muscles du papillon - les mouvements des bras (phase1) [65] :



Phase 2 Dégagement haut du corps [65] :

- ✓ Deltoïdes et coiffe des rotateurs (supra et infra-épineux, petit rond et sub-scapulaire)
- ✓ Ondulation : Muscles stabilisateurs du tronc et muscles paraspinaux

La figure suivante représente les muscles du papillon - les mouvements des bras (phase2) [65] :

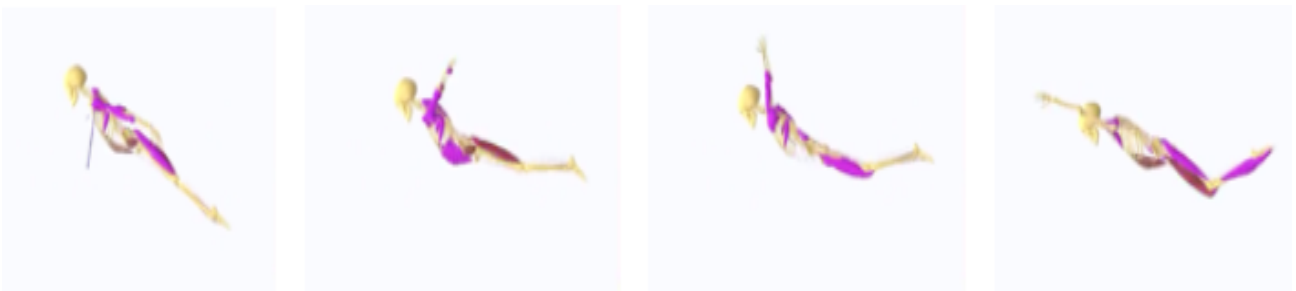


Figure 13 : Papillon: les mouvements des bras (phase 2)

1.3.2 Muscle du Papillon: Mouvements des jambes

Phase Propulsive bas du corps [65] :

- ✓ Phase descendante : iliopectiné, droit de la cuisse, quadriceps
- ✓ Phase ascendante : fessiers, ischio-jambiers
- ✓ Coulée
- ✓ phases : gastrocnémien et solaire (fléchion plantaire)

La figure suivante représente les muscles du papillon - mouvements des jambes [65] :

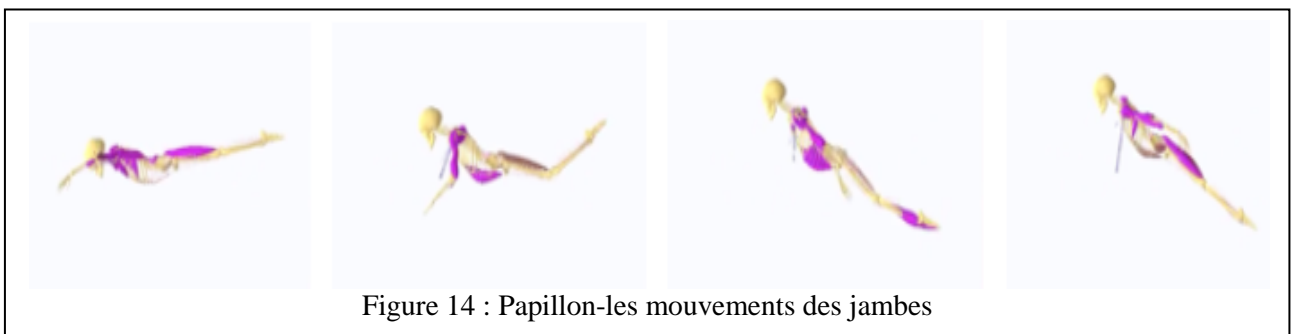


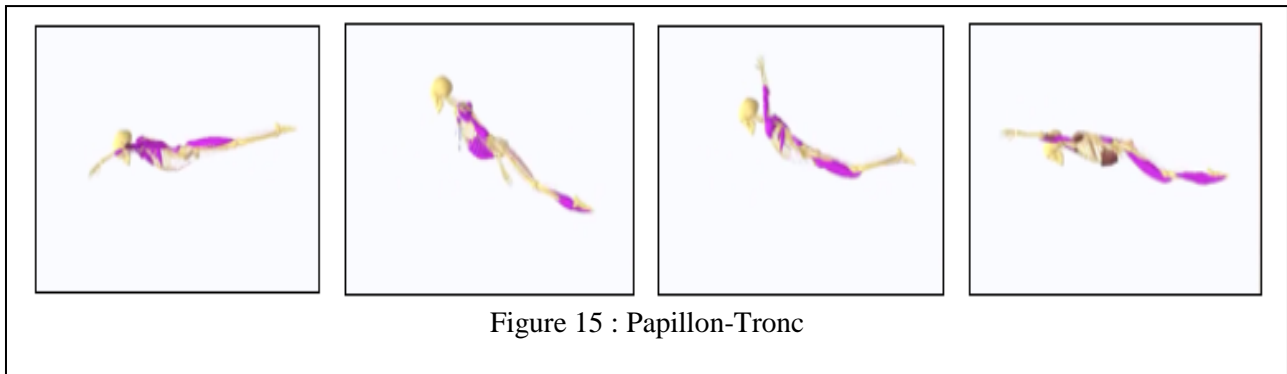
Figure 14 : Papillon-les mouvements des jambes

1.3.3 Muscle du Papillon-Tronc

Phase Stabilisatrice du tronc [65] :

- ✓ Stabilisateurs omoplate (petit pectoral, rhomboïdes, élévateur de scapula, faisceaux moyens et inférieurs du trapèze, grand dentelé)
- ✓ Transverse de l'abdomen, droit de l'abdomen, obliques internes et externes, érecteurs du rachis (muscles paraspinaux)

La figure suivante représente les muscles du Papillon-Tronc [65] :



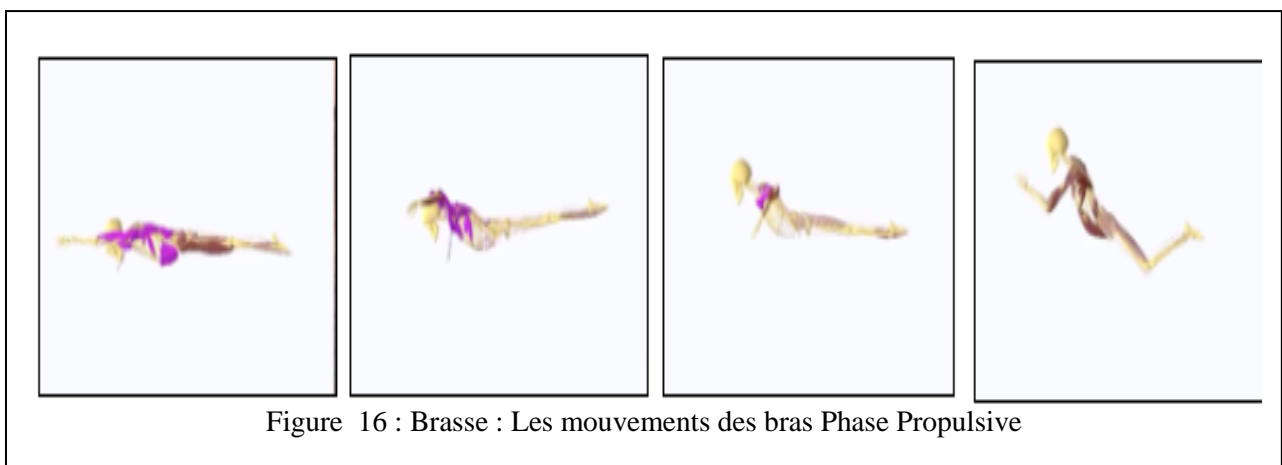
1.4 Brasse :

1.4.1 Muscle de la Brasse : les mouvements des bras :

Phase Propulsive haut du corps [65] :

- ✓ Faisceau claviculaire du grand pectoral et grand dorsal
- ✓ Grand pectoral ++ et grand dorsal ++
- ✓ Muscles paraspinaux

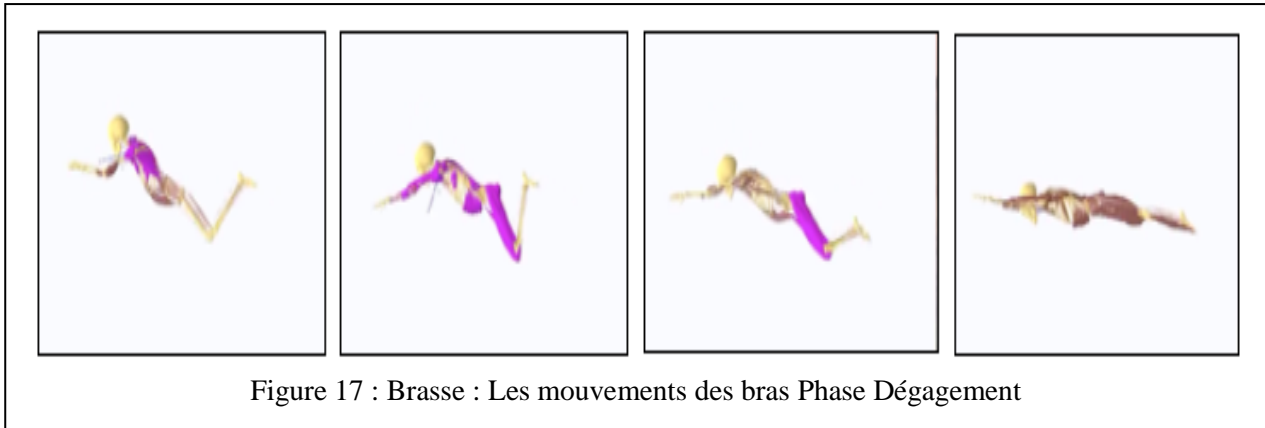
La figure suivante représente les muscles de la Brasse : les mouvements des bras Phase Propulsive [65] :



Phase Dégagement haut du corps [65] :

- ✓ Grand pectoral, faisceau antérieur du deltoïde, chef long du biceps brachial
- ✓ Triceps brachial

La figure suivante représente les muscles de la Brasse : les mouvements des bras Phase Dégagement [65] :



1.4.2 Muscle de la Brasse : les mouvements des jambes

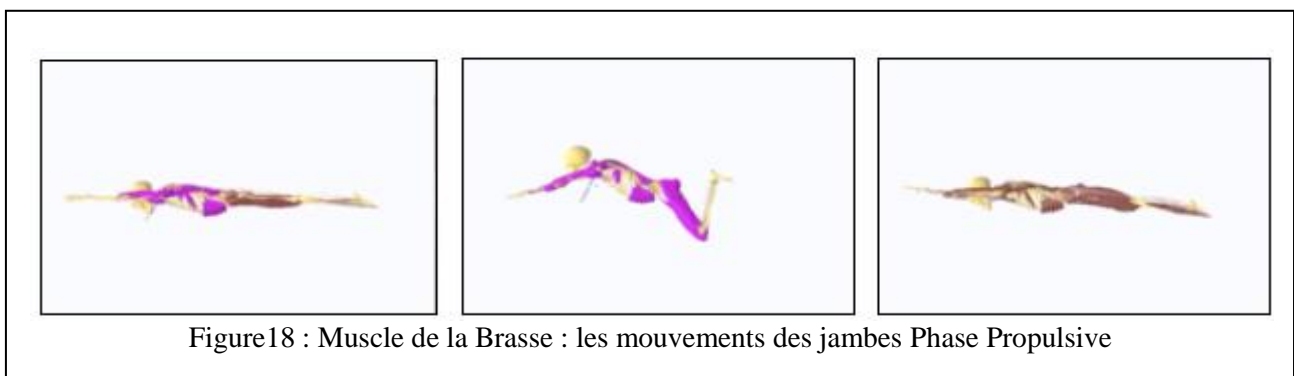
Phase Propulsive bas du corps (balayage extérieur) [65] :

- ✓ Abducteurs, fessiers et ischios jambiers (extension hanches et flexion genoux)
- ✓ Flexion dorsale des chevilles
- ✓ Droit de la cuisse et quadriceps (extension genoux)

Phase Propulsive bas du corps (balayage / intérieur) [65] :

- ✓ Adducteurs, fessiers et ischios jambiers (extension hanches et flexion genoux)
- ✓ Flexion plantaire des chevilles
- ✓ Droit de la cuisse et illiopsoas

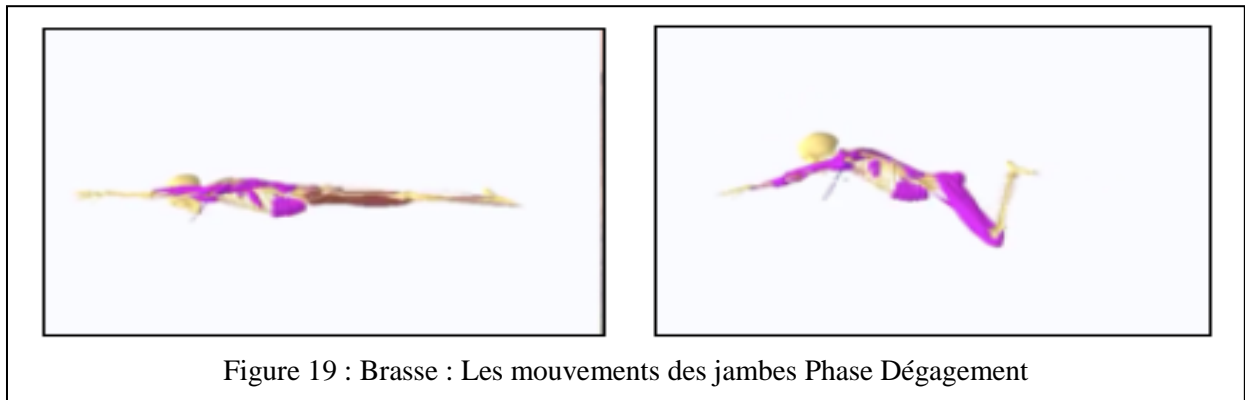
La figure suivante présente les muscles de la Brasse : les mouvements des jambes Phase Propulsive (balayage extérieur/ intérieur) [65] :



Phase Dégagement bas du corps [65] :

- ✓ Droit de la cuisse et illiopsoas (fléxion de hanche)
- ✓ Ischio-jambiers (flexion de genoux)

La figure suivante représente les muscles de la Brasse : les mouvements des jambes Phase Dégagement [65] :

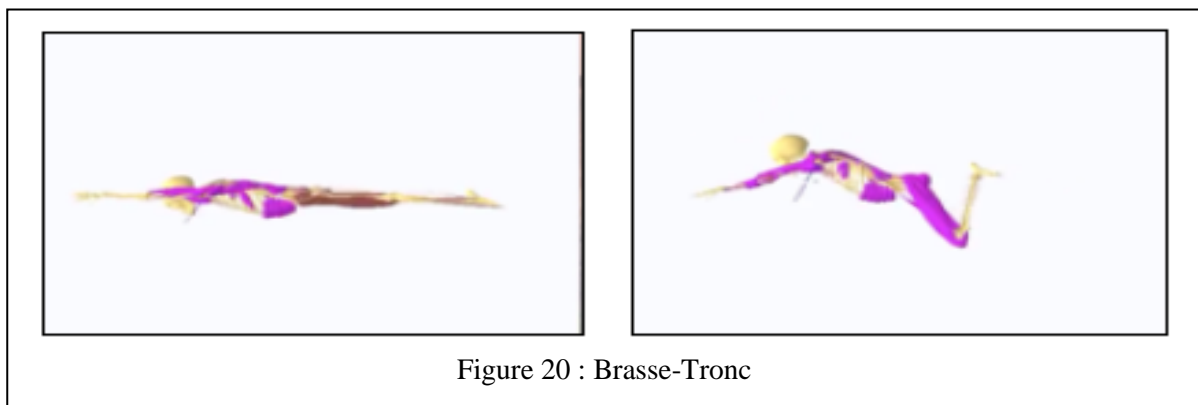


1.4.3 Muscle de la Brasse-Tronc :

Phase Stabilisatrice du tronc [65] :

- ✓ Muscles stabilisateurs de l'omoplate et muscles paraspinaux

La figure suivante représente les muscles de la Brasse-Tronc [65] :



La 3ème partie :

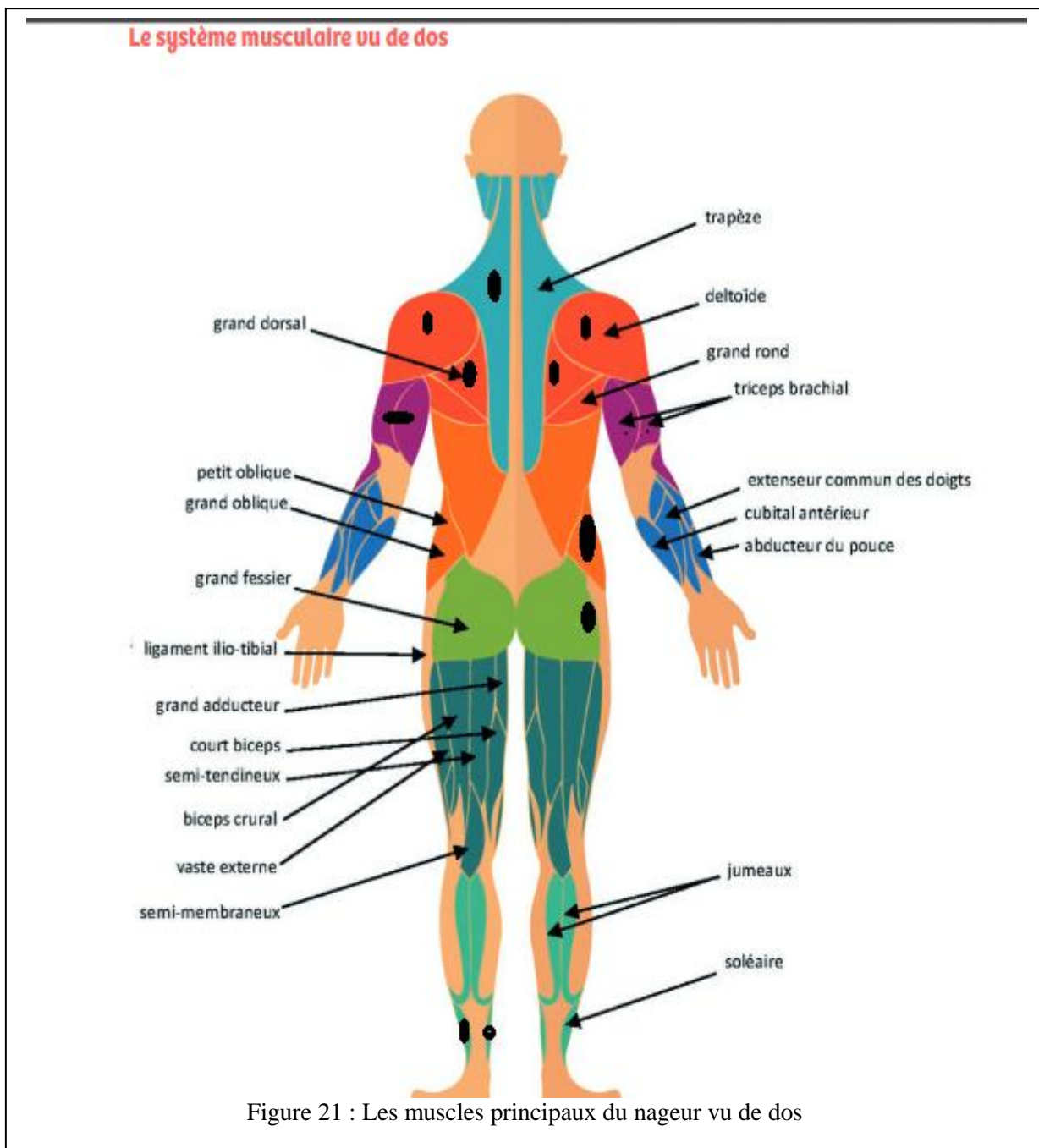
Etude de cas :

Dans cette partie, nous allons répondre à la question : comment appliquer la vision de T.Baker et al. [9] (tout en tant que ressource) dans le domaine sportif (natation). Nous nous sommes inspirés de la vision de T.Baker et al [9]. Dans le domaine de santé pour développer notre vision dans le domaine de sportif (natation). Trois catégories de ressources sont

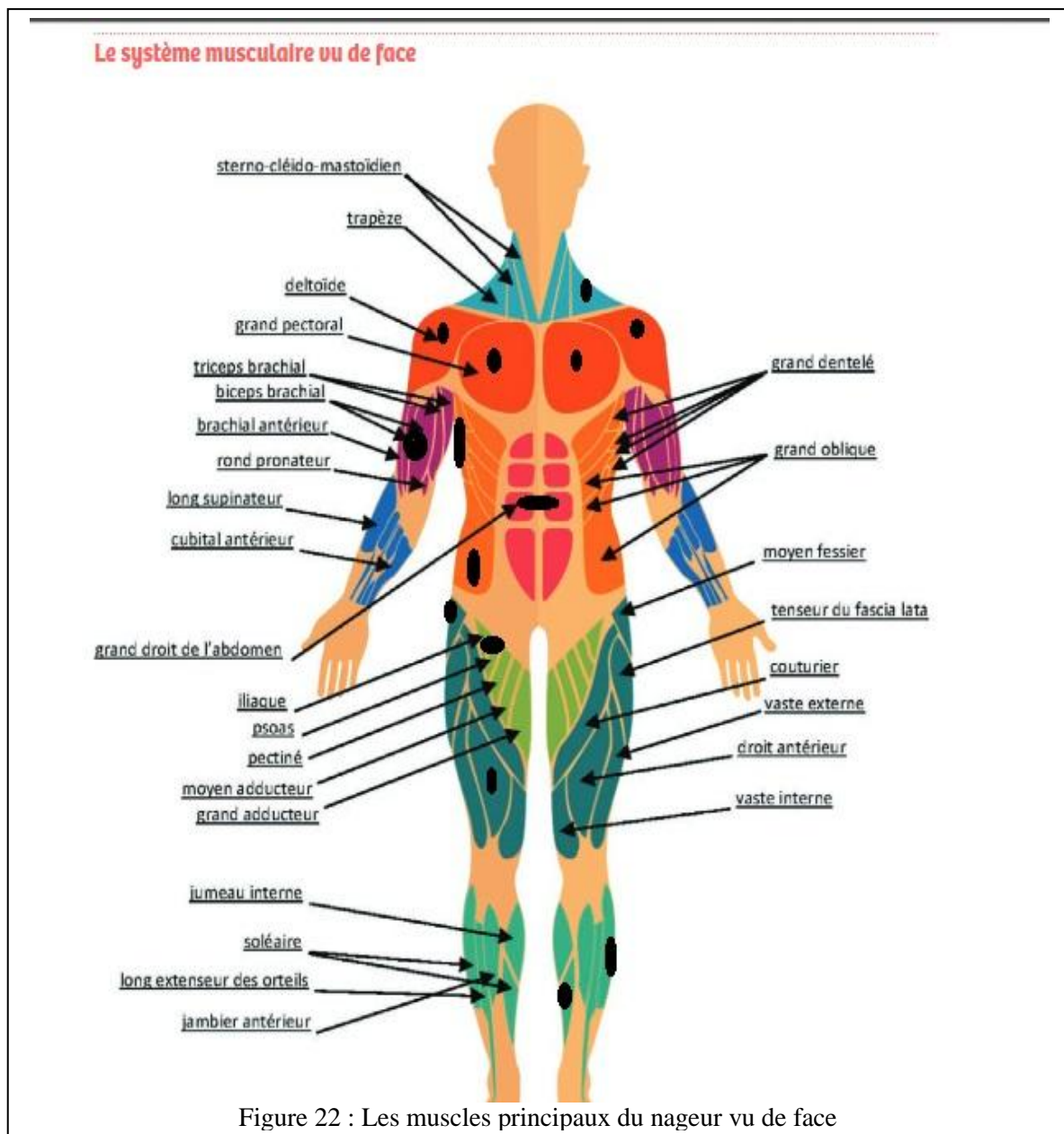
identifiées, comme nous l'avons déjà mentionné dans la 1^{ère} partie (ressources consommées, ressources informatiques et ressources produites).

Nous utilisons l'étude des muscles du nageur dans la 2^{ème} partie pour détecter les principaux muscles utilisés dans les quatre types de natation (Crawl, Papillon, Brasse, Dos), pour placer les capteurs dans la tenue du nageur.

La figure suivante représente la détection des principaux muscles du nageur vue de dos [71] :



La figure suivante représente la détection des principaux muscles du nageur vue de face [71] :



Les points en noire dans les deux figures précédentes (figure 21 et figure 22) représentent les principaux muscles utilisés par les nageurs dans les quatre types de natation.

Après l'identification des principaux muscles, nous utilisons les 2 figures précédentes (figure 21 et figure 22) pour placer les capteurs dans la tenue du nageur.

La figure suivante représente l'emplacement des capteurs de muscles dans la tenue du nageur :



Dans la figure 23 les points en blanc représentent l'emplacement des capteurs de muscles dans la tenue du nageur. Nous utilisons les capteurs de muscles pour capter des informations sur les muscles (ex : dans le cas de blessures, infections, ...).

Les informations captées sont utilisées comme ressources consommées qui sont envoyées pour quelles soient traitées par les ressources informatiques (la 2ème catégorie de ressources).

Les ressources informatiques (ex : REpresentational State Transfer (RESTful)) sont utilisées pour traiter les ressources consommées et fournir des résultats en tant que ressources produites.

Les ressources produites sont la 3ème catégorie de ressources qui sont les résultats de traitement des ressources consommées. Les résultats sont soit un fichier XML, un fichier TXT ou un fichier JSON. Les ressources produites peuvent être utilisées à leur tour comme des ressources consommées.

Conclusion

Ce chapitre est réparti en trois parties :

La 1^{er} partie présente le Framework *aaR « tout en tant que ressource » qui comporte trois couches (couche application, couche ressources, couche infrastructure), les ressources sont classées en : ressources informatiques, ressources consommées, ressources produites qui ont des cycles de vie traçables selon leur catégorie respective, nous avons mentionné les restrictions et les méthodes des ressources.

Le 2^{ème} partie est une étude de l'anatomie humaine (muscles du nageur) dans les quatre types de natation : Crawl, Papillon, Brasse, Dos. Chaque type de natation contient trois mouvements de muscles : mouvement de bras, mouvement de dos et de tronc.

La 3^{ème} partie est une étude de cas dans le domaine de la natation. Nous avons identifié les principaux muscles pour placer les capteurs dans la tenue du nageur.

Conclusion générale

Conclusion générale et perspectives :

L'objectif principale de ce mémoire est d'appliquer une étude de cas dans le domaine de sport (natation) pour illustrer l'internet des objets tout en tant que ressource (*aaR), pour aboutir à cet objectif, plusieurs points essentiels font partie de notre mémoire, tel qu'une illustration du paradigme "tout en tant que ressource", ce paradigme identifie trois types de ressources : ressources consommées, ressources informatiques et ressources produites. La connexion des trois catégories des ressources est simple: une ressource informatique invoquée à l'exécution peut consommer des ressources (existantes) et / ou produire (pas nécessairement de nouvelles ressources), l'analyse des ressources selon deux perspectives: la capacité pour que les ressources informatiques traitent les données, les données abstraites générées et les données consommées capturées et la restriction pour que l'accès aux ressources soit contrôlé en termes de limitation, de possibilité de partage et de renouvellement.

L'étude des muscles du nageur dans les quatre types de natation (Crawl, Brasse, Dos, Papillon) nous a permis d'identifier le placement exact des capteurs dans la tenue du nageur.

Il reste toujours des perspectives:

1. Appliquer cette étude de cas dans le monde industriel.
2. Illustrer tout en tant que ressource avec des études de cas dans différents domaines.

Bibliographie

- [1] Margaret Rouse. « internet of things (IoT) ». Agenda techtarget. June 2018. Accéder le sur : <https://internetofthingsagenda.techtarget.com/definition/Internet-of-Things-IoT> .
- [2] Alexandre Moussier. « L'internet des objet ». Slideshare. décembre 2014. Accéder le sur : <https://fr.slideshare.net/mousski/9-linternet-des-objets>.
- [3] Andrew Meola , « What is the Internet of Things? Meaning & Definition ». Business insider. Mai 2018. Accéder le sur: <https://www.businessinsider.com/internet-of-things-definition?IR=T>
- [4] « what is the internet of things (IoT)?: definition ». Techopedia. Accéder le sur: <https://www.techopedia.com/definition/28247/internet-of-things-iot>
- [5] Kiran Dewangan. dr.Mina Mishra. « Internet of Things for Healthcare:A Review » . International Journal of Advanced in Management, Technology and Engineering Sciences. Researchgate. Volume 8, Issue III. Mars 2018 .
- [6] Rahul. « IoT Applications With Examples ». 30Janvier 2016.
- [7] Echezona Felix Osigwe, PhD alar kuusik, « DEVELOPMENT OF A FORCE SENSOR FOR IOTSPORTS: THE BOAT ROWING APPLICATION » , La thèse de master, TALLINN UNIVERSITY OF TECHNOLOG, Faculty of Information Technology, Thomas Johann Seebeck, Department of Electronics, 23 mai 2016.
- [8] Dhrumit Shukla . « The Internet of Things and the Field of Sports. customer think ». September2017.
- [9] Thar Baker . EmirUgljanin. NouraFaci. MohamedSellami. ZakariaMaamar.EjubKajan. « Everything as a resource: Foundations and illustration through Internet-of-things ». Computers in Industry. Sciencedirect. Volume 94, Pages 62-74. January 2018.
- [10] Kang Bing, Liu Fu, Yun Zhuo, and Liang Yanlei, “Design of an Internet of Things-based Smart Home System”, The 2nd International Conference on Intelligent Control and Information Processing, July 2011, pp. 921-924 .
- [11] Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang, and Chengdong Li, “An IoT-based Appliance Control System for Smart Homes”, Fourth International Conference on Intelligent Control and Information Processing (ICICIP) June 9 – 11, 2013, pp. 744-747.

-
- [12] Mrs.Jyotsna P. Gabhane¹, Ms.Shradha Thakare, Ms.Monika Craig, Smart Homes System Using Internet-of-Things: Issues, Solutions and Recent Research Directions , International Research Journal of Engineering and Technology (IRJET) , Volume: 04, Issue: 05 , May 2017.
- [13] David Bregman, “Smart Home Intelligence - The eHome that Learns”, International Journal of Smart Home, Vol. 4, No. 4, October, 2010, pp. 35-46.
- [14] Timothy Malche, Priti Maheshwary, Internet of Things (IoT) for building Smart Home System.fevrier 2017. International conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud), researchgate.
- [15] M. Karkkainen, T. Ala-Risku, K. Främling, The product centric approach: a solution to supply network information management problems? Comput. Ind. 52 (2003).
- [16] R. Nick, W.M.P. van der Aalst, A.H.M. ter Hofstede, E. David, Workflow resource patterns: identification, representation, and tool support, Advanced Information Systems Engineering, volume 3520 of Lecture Notes in Computer Science, Springer, Berlin, Heidelberg, 2005.
- [17] M. Peter, G. Timothy, The NIST Definition of Cloud Computing. Technical Report 800-145, National Institute of Standards and Technology (NIST), 2011, September.
- [18] T. Berners-Lee, J. Handler, O. Lassila, The semantic web, Sci. Am. 284 (May (5)) (2001).
- [19] R.T. Fielding, Architectural Styles and the Design of Network-based Software Architectures, University of California, Irvine, 2000 (PhD thesis).
- [20] S. Cirani, M. Picone, “Wearable Computing for the Internet of Things”, IEEE, Pg 35-41. 2015.
- [21] Oscar D. Lara and Miguel A. Labrador, “A Survey on Human Activity Recognition using Wearable Sensors”.
- [22] Flammini, E. Sisinni, “Wireless Sensor Networking in the Internet of Things and Cloud Computing Era ”, Procedia Engineering 87 (2014) 672 – 679 .

-
- [23] Dr. Parul Verma , Dr. Ranjana Rajnish , Dr. Shahnaz Fatima , « Challenges: Wearable Computing for Internet of Things (IoT) », International Journal of Science and Research (IJSR) ,ISSN (Online): 2319-7064 , Volume 6 Issue 4, April 2017.
- [24] A. R. Sadeghi¹, C. Wachsmann, and M. Waidner, “Security and privacy challenges in industrial Internet of things,” Proceedings of the 52nd Annual Design Automation Conference, June 2015.
- [25] D. Schulz, « FDI and the industrial Internet of things », Proceedings of IEEE 20th Conference on Emerging Technologies & Factory Automation, 2015, pp. 1-8.
- [26] Matthew N. O. Sadiku, Yonghui Wang, Suxia Cui, and Sarhan M. Musa, « INDUSTRIAL INTERNET OF THINGS », International Journal of Advances in Scientific Research and Engineering (ijasre) , Vol.3 (11) , December-2017.
- [27] David Niewolny. 18 Oct 2013. «How the Internet of Things Is Revolutionizing Healthcare », Freescale Semiconductors.
- [28] Mikhail Simonov, Riccardo Zich, Flavia Mazzitelli. «Personalised healthcare communication in Internet of Things ».
- [29] Alok Kulkarni, Sampada Sathe, « Healthcare applications of the Internet of Things A Review », Alok Kulkar et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (5), pp 6229-6232 .2014.
- [30] B.G.J. ten Bok , « Innovating the Retail Industry; an IoT approach » , 7th IBA Bachelor Thesis Conference, Enschede, The Netherlands., University of Twente, The Faculty of Behavioural, Management and Social sciences . July 1st, 2016.
- [31] Athul Jayaram, « Smart Retail 4.0 IoT Consumer Retailer Model for Retail Intelligence and Strategic Marketing of In-store Products » , researchgate , February 2017.
- [32] Wen-Jye Shyr , Li-Wen Zeng , Chen-Kun Lin , Chia-Ming Lin , Wen-Ying Hsieh ,
« Application of an Energy Management System via the Internet of Things on a University Campus » , EURASIA Journal of Mathematics, Science and Technology Education, 2018, 14(5), 1759-1766, November 2017.

-
- [33] Botta, A.; de Donato, W.; Persico, V.; Pescapé, « A. Integration of Cloud computing and Internet of Things:A survey. Future Gener. Comput. Syst », 56, 684–700. **2016**.
- [34] Saber Talari , Miadreza Shafie-khah, Pierluigi Siano , Vincenzo Loia ,Aurelio Tommasetti and João P. S. Catalão , «A Review of Smart Cities Based on the Internet of Things Concept » , 23 March 2017 , mdpi/journal/energies , 10, 421.
- [35] M. Stočes, J. Vaněk, J. Masner, J. Pavlík, « Internet of Things (IoT) in Agriculture - Selected Aspects » , Agris on-line Papers in Economics and Informatics , researchgate , Volume VIII , , mars 2016.
- [36] Somayya Madakam, « Internet of Things: Smart Things »,International Journal of Future Computer and Communication, Vol. 4, No.4, August 2015.
- [37] J. Antonio et al, “Drugs interaction checker based on IoT,” Internet of Things (IOT), pp. 1-8, IEEE, 2010.
- [38] « The Vote Is In, Intel Corp : the vote is in : citizens support smart cities with driverless cars public service drones and surroundings that sense activities». intel newsroom. Février 2014.
- [39] Whitepaper « 2025 Every Car Connected: Forecasting the Growth and Opportunity », GSMA- SBD Février 2012.
- [40] Antonette Igbenoba. « autonomous-vehicles-and-the-internet-of-things ». information counts . novembre 2016 .
- [41] « The connected car of the future », agl discover: <https://aglsolar.com.au/blog/the-connected-car-of-the-future/>
- [42] C. Balasubramanian and D. Manivannan, “IoT Enabled Air Quality Monitoring System (AQMS)using Raspberry Pi”, Indian Journal of Science and Technology,Vol 9(39), DOI: 10.17485/ijst/2016/v9i39/90414, October 2016 ,ISSN (Print) : 0974-6846ISSN (Online) : 0974-5645.
- [43] Service Training Report- “Basics of Electric Vehicles”, Volkswagen Group of America, July 2013 .

-
- [44] Boon-Giin Lee and Wan-Young Chung, “A Smartphone-Based Driver Safety Monitoring System Using Data Fusion”, *Sensors* 2012, 12, 17536-17552; doi:10.3390/s121217536.
- [45] Ruhi Kiran Bajaj 1, Madhuri Rao 2, Himanshu Agrawal, 2018, « Internet Of Things (IoT) In The Smart Automotive Sector: A Review » , *IOSR Journal of Computer Engineering (IOSR-JCE)* e-ISSN: 2278-0661,p-ISSN: 2278-8727, PP 36-44, Conference on Recent Trends in Computer Engineering (CRTCE -2018).
- [46] Ray, P. (2015). Internet of Things for Sports (IoTSport): An architectural framework for sports and recreational activity. 2015 International Conference On Electrical, Electronics, Signals, Communication And Optimization (EESCO).
- [47] Pasluosta, C., Gassner, H., Winkler, J., Klucken, J., & Eskofier, B. (2015). An Emerging Era in the Management of Parkinson’s Disease: Wearable Technologies and the Internet of Things. *IEEE JOURNAL OF BIOMEDICAL AND HEALTH INFORMATICS*, 19(6), 1873.
- [48] Thibodeau, P. (2014). The Internet of Things for sports is quickly taking shape. *Computer World*. Retrieved 03 March 2016 .
- [49] Lightman, K. (2016). « Silicon Gets Sporty - Sensors invade tennis rackets, basketballs, soccer balls, boxing gloves and more ». *IEEE Spectrum*, 53(3 (INT)), 43-49.
- [50] Rajesh Khanna . 26 January 2017. « Internet of Things: Architectures, Protocols, and Applications ». *Journal of Electrical and Computer Engineering* .Volume 2017, Article ID 9324035, pages 25.
- [51] H. Ning and Z. Wang, “Future internet of things architecture: like mankind neural system or social organization framework?” *IEEE Communications Letters*, vol. 15, no. 4, pp. 461–463, 2011.
- [52] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, “Internet of Things (IoT): a vision, architectural elements, and future directions,” *Future Generation Computer Systems*, vol. 29, no. 7, pp. 1645–1660, 2013.
- [53] F. Bonomi, R. Milito, P. Natarajan, and J. Zhu, “Fog computing: a platform for internet of things and analytics,” in *Big Data and Internet of Things: A RoadMap for Smart Environments*, pp. 169–186, Springer, Berlin, Germany, 2014.

-
- [54] I. Stojmenovic and S. Wen, “The fog computing paradigm: scenarios and security issues,” in Proceedings of the Federated Conference on Computer Science and Information Systems (Fed- CSIS '14), pp. 1–8, IEEE, Warsaw, Poland, September 2014.
- [55] M. Aazam and E.-N. Huh, “Fog computing and smart gateway based communication for cloud of things,” in Proceedings of the 2nd IEEE International Conference on Future Internet of Things and Cloud (FiCloud '14), pp. 464–470, Barcelona, Spain, August 2014.
- [56] L. Atzori, A. Iera, and G. Morabito, “SIoT: giving a social structure to the internet of things,” IEEE Communications Letters, vol. 15, no. 11, pp. 1193–1195, 2011.
- [57] P. Prescod, “Roots of the rest/soap debate,” August 2002.
- [58] Khalid Elgazzar, Patrick Martin, Hossam Hassanein , « Mobile Web Services: State of the Art and Challenges » , (IJACSA) International Journal of Advanced Computer Science and Applications, Vol. 5, No. 3, 2014
- [59] Heather Kreger , « Web Services Conceptual Architecture (WSCA 1.0) » ,IBM Software Group , mai 2001
- [60] Cary Pennington, Jorge Cardoso, John A. Miller, Richard Scott Patterson, Ivan Vasquez. « Introduction to Web Services ». IGI Group. pp134 -154. 2007
- [61] JM. Chauvet, “Web Services avec SOAP, WSDL, UDDI, ebXML”, Eyrolles, 2002
- [62] Jaideep Roy and Anupama Ramanujan. « Perspectives : Understanding Web Services ». IEEE. décembre 2001.
- [63] HEDJERES Mouna. « Contribution à la Conception et à la mise en œuvre d'Architectures Orientées Services (SOA) dans un environnement coopératif, "MAeMSOS": Une plateforme Multi Agents Orientée Services pour la e-Maintenance ». ECOLE DOCTORALE D'INFORMATIQUE (STIC) .université de setif. avril 2011.
- [64] Rada, R, Mili, H, Bicknell, E, Blettner, M.: Development and application of a metric on semantic nets. In : IEEE Transactions on Systems, Man and Cybernetics .(1989) 17-30.
- [65] « La musculation du nageur ». Fédération française: Natation. Propose.2012
- [66] Shancang, L., Li, D. X., and Shanshan, Z. « The internet of things: a survey ». Information System Frontier , 17 (2), 243–259.2015.
-

-
- [67] Atzori, L., Lera, A., and Morabito, G. The internet of things: A survey. *Computer Network*, 54 (15), 2787–2805. (2010).
- [68] Korteum, G., Kawsar, F., Fitton, D., and Sundramoorthy, V. « Smart objects as building blocks for the Internet of Things ». *IEEE Internet Comput*, 1 (51), 44-51. (2010).
- [69] Gordana, G., Mladen, V., Nebojsa, M., Dragan, V., Igor, R., Slavica, T., and Milutin, R. « The IoT Architectural Framework, Design Issues and Application Domains ». *Wireless Personal Communications*, 92 (1), 127-148. (2017).
- [70] Bandyopadhyay, D., Sen, J. « Internet of Things: Applications and Challenges in Technology and Standardization ». *Wireless Personal Communication*, 58 (1), 49-69. (2011).
- [71] Hugues Oger. « La préparation physique des Jeunes sapeurs-pompiers de France ». 2018.
- [72] Melanie Swan. « *Review :Sensor Mania! The Internet of Things, Wearable Computing, Objective Metrics, and the Quantified Self 2.0* ». *Journal of Sensor and Actuator Networks*. J. Sens. Actuator Netw. 2012, 1, 217-253.
- [73] Gaurav Tripathi, Dhananjay Singh, and Antonio J. Jara, “A survey of Internet-of-Things: Future Vision, Architecture, Challenges and Service”, *IEEE World Forum on Internet of Things (WF-IoT)*, 2014, pp. 287-292 .
- [74] Ming Wang, Guiqing Zhang, Chenghui Zhang, Jianbin Zhang, and Chengdong Li, “An IoT-based Appliance Control System for Smart Homes”, *Fourth International Conference on Intelligent Control and Information Processing (ICICIP)* June 9 – 11, 2013, pp. 744-747 .
- [75] Vittorio Miori, and Dario Russo, “Domotic evolution towards the IoT”, *28th International Conference on Advanced Information Networking and Applications Workshops*, 2014, pp. 809-814 .
- [76] Sarita Agrawal, and Manik Lal Das, “Internet of Things – A Paradigm Shift of Future Internet Applications”, *International Conference on Current Trends in Technology*, December, 2011 .
- [77] Ethan Cerami, *Web Services Essentials*, édition O’Reilly, Février 2002.
- [78] Julien Ponge. *Model Based Analysis of Time-aware Web Services Interactions*. Thèse de Doctorat de l’Université Blaise Pascal - Clermont-Ferrand II, dans le cadre de l’Ecole Doctorale des Sciences pour l’Ingénieur, France, 2008.

-
- [79] Ben Margolisand, et Joseph Sharpe, SOA for the Busines Developer: Concepts, BPEL, and SCA, édition MC Pres.
- [80] Nicolai Josuttis, SOA in Practice, édition O'Reilly, Septembre 2007.
- [81] C. Devaux, Urbanisation et SOA : Quelques bonnes pratiques pour leur mise en oeuvre, Livre Blanc, Aubay, France, 2008.
- [82] Heather Kreger, Web Service Conceptual Architecture, édition IBM Software Group, Mai 2001.
- [83] M.PRADET. « Lapréparationphysique », INSEP-Publications,1996.
- [84] Bernard Boda .Michel Récopé. « PRÉPARATIONS AUX CONCOURS ».Laboratoire d'anthropologie des pratiques corporelles .UFRSTAPS Clermont-Ferrand. Revue EP.S n°231 Septembre-Octobre 1991.
- [85] Pantona, E. « Innovation Drivers in the Retail Industry ». *International Journal of Information Management*(34), 344-350. mars 2014.
- [86] Reddy, T. « 13 Retail Companies Using Data to Revolutionize Online & Offline Shopping Experiences ». Mai 2015.
- [87] Gregory, J. « *The Internet of Things: Revolutionizing the Retail Industry* ». Accenture. (2015).
- [88] Shen, X. Emerging technologies for e-healthcare. *IEEE Network* ,26 (5). (2012).
- [89] W3C World Wide Web Consortium; “Web Services Architecture”; W3C Working Group Note 11; February 2004.
- [90] Fielding, R. T. 2000, “Architectural Styles and the Design of Network-based Software Architectures”, PhD thesis, Department of Information and Computer Science, University of California, Irvine, USA, 2000.
- [91] Curbera, F., Nagy, W., Weerawarana, S. *Web services: Why and how*. Paper presented at the Workshop on Object-Oriented Web Services- OOPSLA 2001, Tampa, Florida.2001.
- [92] Eric Newcomere, Understanding Web Services Xml WSDL SOAP and UDDI, édition O'Reilly, 2004.