

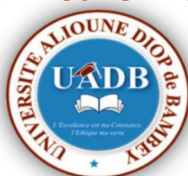
République du Sénégal



Un Peuple – Un But – Une Foi

Ministère de l'Enseignement Supérieur de la Recherche et de l'innovation

Université Alioune DIOP de Bambey



UFR Sciences Appliquées et Technologies de l'Information et de la

Communication (SATIC)

Département Technologies de l'Information et de la Communication (TIC)

Licence Professionnelle

Développement et Administration d'Applications Web (D2AW)

## PROJET DE MÉMOIRE

**Mise en place d'une plate-forme  
centralisée de gestion d'objets  
connectés**

**Encadreur(s) :**

*Dr. Fatoumata Baldé Kassé*

**Présenté par :**

*Adja Batta Ba*

*Demba Diallo*

Année académique 2018-2019



# **Dédicaces**

## ***À nos très chers parents***

*Nous vous devons ce que nous sommes aujourd'hui, grâce à votre amour, votre patience et vos innombrables sacrifices. Vous n'avez pas cessé de nous offrir les conditions favorables pour étudier. Que ce modeste travail, soit pour vous une petite compensation et reconnaissance pour tout ce que vous avez fait. Que Dieu, vous préserve et vous procure santé et longue vie afin que nous puissions à notre tour vous combler.*

## ***Un grand hommage à notre père Ibrahima Ba***

*Que le paradis soit ta demeure éternelle!*

## ***À nos très chers frères et sœurs***

*Selly Ba, Ouleye Ba ,Riche Ba et Oumar Ba*

*Aucune dédicace ne pourrait exprimer assez profondément ce que nous ressentons envers vous. Nous vous disons tout simplement, merci, nous vous aimons.*

## ***Aux familles Sow ,Cissé et Ndiaye.***

## ***À nos très chers ami(e)s***

*En témoignage de l'amitié sincère qui nous lie et les bons moments passés ensemble. Nous vous dédions ce travail en vous souhaitant un avenir radieux et plein de réussites.*

## **Remerciements**

*Nous remercions en premier lieu ALLAH le tout puissant de nous avoir donné non seulement le courage mais aussi la force et la volonté nécessaire pour la réalisation de ce modeste travail.*

*Nos vifs remerciements s'adressent à nos parents, nos frères et sœurs pour leur soutien moral et leur encouragement.*

*Nous remercions nos chers amis qui sont toujours présents et fidèles.*

*Nous tenons à exprimer nos profondes gratitude et nos sincères remerciements à notre encadreur Dr. Fatoumata Baldé Kassé pour la haute qualité de son encadrement, son suivi, sa disponibilité et ses conseils. Sans vous, la réalisation de ce mémoire n'aurait pas eu lieu. Encore une fois, merci beaucoup.*

*Nous adressons nos remerciements aux membres du jury qui nous ont fait l'honneur d'évaluer, d'examiner et d'enrichir notre modeste travail.*

*Notre reconnaissance va particulièrement à l'ensemble des enseignants du département Informatique de l'Université Alioune Diop de Bambey pour toutes les connaissances qu'ils nous ont transmises tout au long de notre formation.*

*Enfin nous remercions tous ceux qui ont contribué de près ou de loin à la réalisation de ce travail.*

## **Résumé**

La vie moderne est bouleversée par les technologies de l'information et de la communication TIC, particulièrement le réseau internet et les services qu'il propose: réseau sociaux, E-commerce, E-learning,... pour ne citer que quelques exemples. Dans ce contexte, il y-a lieu de citer l'avènement des IoT (Internet of Things). Cette nouvelle technologie est basée sur l'association d'équipements embarqués et logiciels intelligents connectés sur internet offrant une multitude d'applications et de services allant du transport et la supervision jusqu'au bien être des individus. Il existe de nos jours de nombreuses plate-formes et applications pour l'IoT. Dans une tentative de profiter de cette occasion, ce mémoire de licence présente une plate-forme centralisée de gestion de réseau d'objets connectés. Basé sur une stratégie de centralisation, ce modèle propose un schéma de récupération des données, leur stockage, leur traitement mais aussi et surtout une bonne gestion de ses abonnés.

Mots clés: internet , IoT, plate-forme, objets connectés, données, récupération, stockage, traitement, abonnés.

# Table des matières

Dédicaces.....	i
Remerciements.....	ii
Résumé.....	iii
Liste des figures.....	iv
Liste des tableaux.....	v
Liste des sigles.....	vi
<b>Introduction.....</b>	<b>1</b>
<b>Chapitre 1: Internet des objets.....</b>	<b>3</b>
1.1 Définition.....	3
1.2 Caractéristiques.....	4
1.4 Évolutions technologiques.....	6
1.5 Avantages.....	9
1.6 Inconvénients.....	12
1.7 Enjeux de l'IoT.....	12
<b>Chapitre 2: Conception de la plate-forme.....</b>	<b>13</b>
2.1 Analyse des besoins.....	11
2.2 Rôle des acteurs et Description des fonctionnalités.....	15
2.3 Diagrammes de cas d'utilisation.....	16
2.4 Description des différents scénarios et gestion des exceptions.....	19
2.5 Diagrammes de séquences.....	21

<b>Chapitre 3: Réalisation de la plate-forme.....</b>	<b>24</b>
3.1 Diagramme de classe.....	22
3.2 Technologies utilisées.....	25
3.3 Présentation de la plate-forme.....	35
Conclusion.....	40
Références.....	41

## **Liste des figures**

Figure 1-1 : Les objets de l'IoT.....	4
Figure 1-2 : architecture de l'IoT.....	5
Figure 1-3 : processus de traitement des données issues des objets.....	6
Figure 1-4 : maison connecté (smart house).....	8
Figure 1-5 : Première voiture autonome.....	8
Figure 1-6 : piège à souris connecté .....	9
Figure 1-7 : les bennes connectés .....	10
Figure 1-8 : brosse à dent connectée.....	11
Figure 1-9 : les clés connectés .....	11
Figure 2-1 : diagramme de cas d'utilisation des utilisateurs.....	17
Figure 2-2: diagramme de cas d'utilisation de l'administrateur.....	18
Figure 2-3 : diagramme de cas d'utilisation de l'abonné.....	18
Figure 2-4: diagramme de séquence de l'authentification.....	22
Figure 2-5: diagramme de séquence du demande d'abonnement.....	23
Figure 3-1 : diagramme de classe.....	25
Figure 3-2: Logo Java EE.....	26
Figure 3-3: Logo Java Mail.....	31
Figure 3-4: Logo JDBC.....	31
Figure 3-5: Logo JavaScript.....	33



Figure 3-6: Logo WampServer.....	33
Figure 3-7: Logo PowerAmc.....	34
Figure 3-8: Architecture de l'application.....	35
Figure 3-9: Page d'accueil de la plate-forme.....	36
Figure 3-10: Page d'authentification de la plate-forme.....	36
Figure 3-11: Espace administrateur.....	37
Figure 3-12: Espace Abonné.....	37
Figure 3-13: Informations sur les objets d'un abonné.....	38
Figure 3-14: Graphe des données issues des capteurs.....	38
Figure 3-15: Graphe des données issues des capteurs.....	38

## **Liste des tableaux**

Table 1-1 : liste des sigles.....	13
Table 2-1 : les acteurs, leur fonctionnalité et leur rôle.....	16
Table 2-2 : scénario du diagramme de séquence «authentification».....	20
Table 2-3 : scénario du diagramme de séquence «demande abonnement»..	21

## Liste des sigles

	Significations	
Sigles	Français	Anglais
IoT	Internet des objets	Internet of Things
IP	Protocole Internet	Internet Protocol
http	Protocole de Transfert Hypertexte	Hypertext Transfer Protocol
RFID	radio-identification	Radio Frequency Identification
Wifi	réseau local sans fil	Wireless Fidelity
IA	Intelligence artificielle	-
VR	Réalité Virtuelle	Virtual Reality
LoraWan	Réseau Etendu à Longue Portée	Long Range Wide-area network
UML	Langage de Modélisation Unifié	Unified Modeling Language
JEE	Entreprise Java Edition	Java Enterprise Edition
J2SE	Java Standard Edition	Java Standard Edition
JSP	Java Server Pages	Java Server Pages
EJB	Entreprise Java Bean	Enterprise Java Bean
JVM	Machine Virtuelle Java	Java Virtual Machine
IDE	Environnement de	integrated development

	Développement	environnement
JDBC	Connexion Base de Données en Java	Java DataBase Connectivity
JTA	Transaction d'API Java	Java Transaction API
JNDI	Nommage java et interface de répertoire	Java Naming and Directory Interface
JCA	Architecture de connecteur java	Java Connector Architecture
JAAS	Service d'authentification et d'autorisation java	Java Authentication and Authorization Service
JMS	Java Message Service	Java Message Service
JSTL	Page du serveur java bibliothèque de balise standard	Java server page Standard Tag Library
HTML	Langage de balisage hypertexte	Hypertext Markup Langage
MVC	Modèle Vue Contrôleur	Model View Controller
URL	Emplacement des ressources uniformes	Uniform Resouce Location
XML	Langage de balisage extensible	eXtensible Markup Langage
SQL	Langage de requêtes structurées	Structured Query Language
MUA	Agent de courriel électronique	Mail User Agent
MTA	Agent de transfert de	Mail Transfer Agent

	courrier	
SGBD	Système de gestion de base de données	-

Table 1-1: liste des sigles

# **Introduction**

## **➤ Contexte et Problématique**

Depuis quelques années déjà, nous entendons parler de “l’internet des objets”. Cette expression n’est pas nouvelle: elle fut adoptée dès 1999 par le pionnier britannique de la technologie Kevin Ashton qui travaillait en tant qu’assistant chef de marque chez Procter & Gamble. Depuis, lorsque nous parlons de révolution, l’internet des objets ne cesse de revenir sur le premier plan. Au départ, beaucoup de personnes ne pensaient pas que les objets connectés deviendraient réalité. Aujourd’hui, c’est devenu un fait. L’internet des objets va révolutionner le monde dans le sens où une grande partie des objets avec lesquels nous interagissons aujourd’hui seront connectés à l’avenir, et cela offrira un bien-être aux individus. L’internet des objets est à ses premiers pas et il ne cesse de croître au fil des années.

Vu l’estimation énorme du nombre d’objets connectés en 2020 (50 milliards) et la faible limite de leur capacité de stockage, il s’impose alors de prendre des mesures.

Pour répondre à cette nouvelle ère technologique nous avons décidé de mettre en place une plate-forme de gestion centralisée d’objets connectés avec des fonctionnalités plus innovantes: car il y a exactement deux ans un projet du même genre avait été traité par des étudiants mais au niveau statique.

## **➤ Objectifs poursuivis**

L’objectif de notre projet est la réalisation d’une plate-forme de gestion centralisée répondant aux besoins suivants:

- héberger des objets connectés
- communiquer avec un réseau d’objets connectés
- gérer des abonnés (propriétaires des objets) partout au Sénégal
- générer un rapport de consommation d’énergie pour chaque abonné
- permettre aux abonnés de:

- ✧ manipuler leurs objets à distance
- ✧ visualiser l'état de leurs objets
- ✧ suivre l'évolution de leur consommation d'énergie.

## ➤ **Structure du document**

Ce travail se repartie en trois grandes parties que nous détaillerons comme suit :

Tout d'abord, nous allons définir l'internet des objets, citer ses fonctionnalités, ses différents domaines d'applications, expliquer ses acteurs, mentionner ses avantages et ses inconvénients.

Ensuite, nous aborderons la conception du projet, citer les différents acteurs et les fonctionnalités qu'ils accomplissent et élaborer les différents modèles.

Enfin, nous parlerons de la réalisation de la plate-forme, citer les différentes entités de la base de données à travers un diagramme de classes, énumérer les différents outils technologiques utilisés et présenter quelques interfaces de la plate-forme avant de passer à la conclusion .





Figure 1-1: Les objets de l'IoT [futura tech, 2015]

## **1.2 Caractéristiques**

L'IoT n'est pas une technologie mais il est un concept qui s'appuie sur un ensemble de technologies qui ont évolué au cours des dernières décennies et qui ont comme objectif à rendre possible l'IoT.

L'un des fondements de l'IoT repose sur l'aptitude des objets à communiquer entre eux, quelle que soit leur nature, leur origine et leur localisation. Interopérabilité et ouverture sont deux mots clés de l'Internet des objets.

En effet, toutes les technologies que l'homme a développées peuvent trouver dans le concept d'IoT une justification nouvelle c'est pourquoi ses domaines d'activités sont nombreux.

On retrouve deux caractéristiques de technologies: matérielles et logicielles, qui offrent un bon fonctionnement des composants et des coûts de plus en plus réduits aux objets des capacités de traitement, de mémoire et de communication nécessaires à l'IoT.

D'un point de vue conceptuel, l'Internet des objets affecte, à chaque objet une identification unique sous forme d'une étiquette lisible par des dispositifs mobiles sans fil, afin de pouvoir communiquer les uns avec les autres. Ce réseau crée une passerelle entre le monde physique et le monde virtuel.

D'un point de vue technique, l'IoT consiste à une identification numérique directe et normalisée (adresse IP, protocole http...) d'un objet physique grâce à un système de communication sans fil (puce RFID, Bluetooth ou WiFi).

## **1.3 Fonctionnement**

Les objets de l'environnement de l'IoT permettent de collecter, stocker et transmettre des données issues du monde physique. Ce sont des sources de données, qui possède au minimum un identifiant unique attaché à une identité ayant un lien direct ou indirect avec Internet.

On distingue deux types d'objet :

Les objets passifs : ils utilisent généralement un tag (puce RFID, code barre 2D). Ils ont une capacité de stockage faible (de l'ordre du kilooctet) et permettent de jouer le rôle d'identification. Ils peuvent parfois, dans le cas d'une puce RFID, d'embarquer un capteur (température, humidité) et être réinscriptibles.

Les objets actifs : ils peuvent être équipés de plusieurs capteurs, avec une grande capacité de stockage, capables d'accomplir des calculs et être en mesure de communiquer sur un réseau.

- **Capter** permet de transformer une grandeur physique analogique en un signal numérique.
- **Concentrer** permet d'interfacer un réseau spécialisé d'objet à un réseau IP standard.
- **Stocker** permet de rassembler des données brutes, produites en temps réel, arrivant de façon non prévue.
- **Présenter** permet de collecter les informations de façon compréhensible par l'Homme, en lui offrant un moyen d'agir et/ou d'interagir.

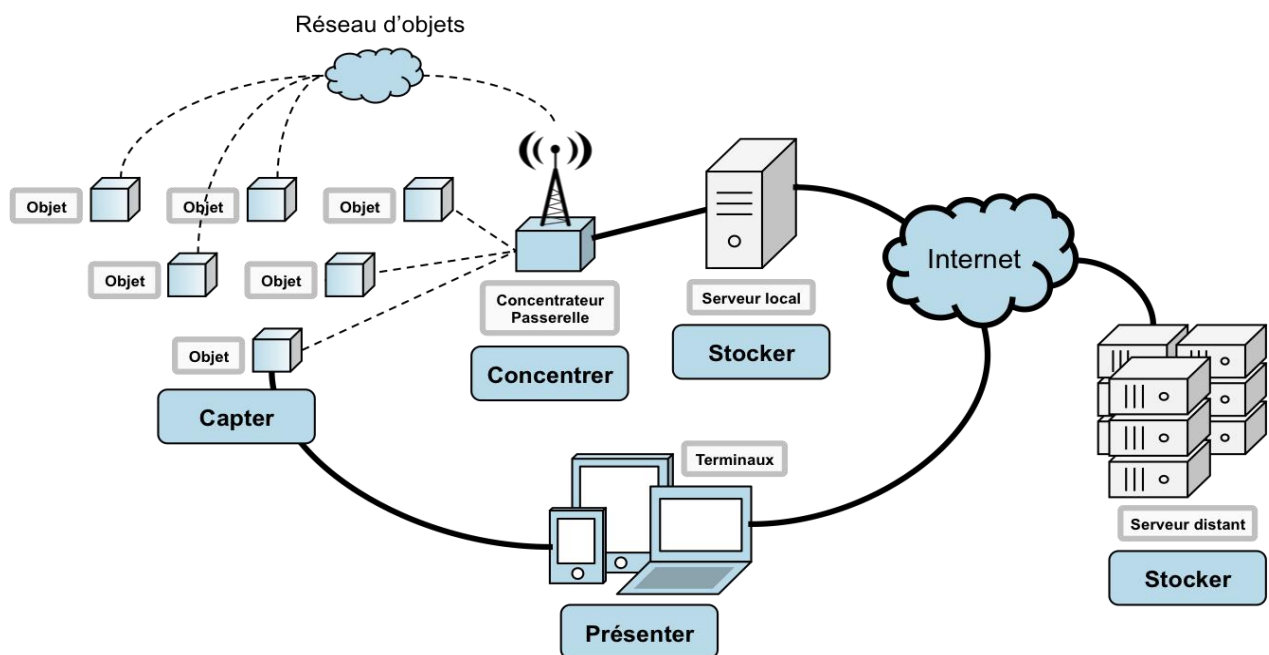


Figure 1-2: architecture de l'IoT [Plouin G et Colomer N, 2011]

**Le traitement des données** : est un processus qui peut intervenir à tous les niveaux de la chaîne, depuis la capture de l'information jusqu'à sa restitution. Une stratégie pertinente, et commune quand on parle d'Internet des objets, consiste à stocker l'information. Cette stratégie est possible aujourd'hui grâce à des architectures distribuées, capables d'emmagasiner de

grandes quantités d'information tout en offrant la possibilité de réaliser des traitements complexes en leur sein.

**La transmission des données** : est un processus qui intervient à tous les niveaux de la chaîne. Deux réseaux supportent des transmissions le réseau local de concentration (utilise ANT, ZigBee et Zwave..) et le réseau WAN ( WiFi, réseaux cellulaires. . . ).

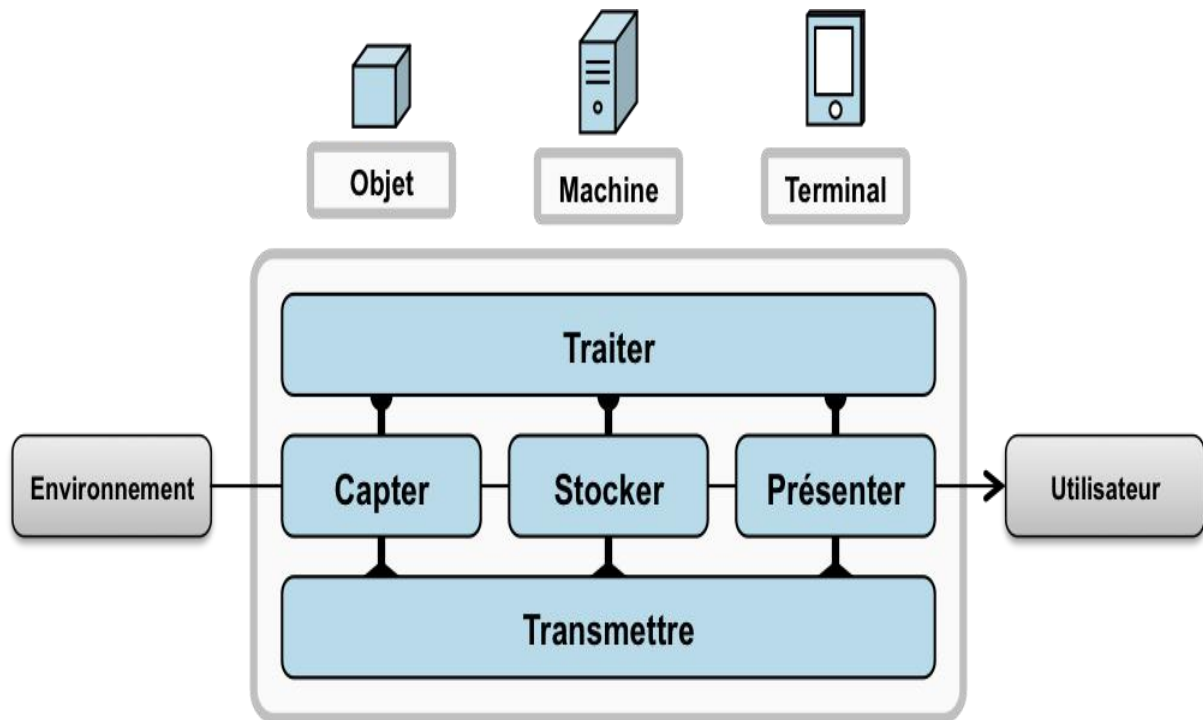


Figure 1-3: processus de traitement des données issues des objets [Plouin *et al.*, 2011]

## 1.4 Évolutions technologiques

Le terme Internet des objets est utilisé pour la première fois en 1999 par Kevin Ashton, employé de Procter & Gamble (P&G), qui désigne alors le lien entre la technologie RFID et l'Internet.

Le premier objet connecté est commercialisé en 2003 par la firme Violet. C'est la lampe DAL, équipée de 9 leds qui s'allument en fonction des événements. Deux ans plus tard, en 2005, Violet lance le célèbre lapin Nabaztag. Racheté par Mindscape puis Aldebaran Robotics en 2013, il disparaîtra en 2015.

**IoT + data + IA** : l'Internet des objets n'est donc pas une technologie récente mais elle est aujourd'hui associée à d'autres technologies en plein développement telles que l'intelligence artificielle, la big data ou la VR (Virtual Reality) en langue française «Réalité Virtuelle». En effet, l'expression « réalité virtuelle » ou multimédia immersif ou réalité simulée par ordinateur est une technologie informatique qui simule la présence physique d'un utilisateur dans un environnement artificiellement généré par des logiciels. La réalité virtuelle crée un environnement avec lequel l'utilisateur peut interagir. La réalité virtuelle reproduit donc artificiellement une expérience sensorielle, qui peut inclure la vue, le toucher, l'ouïe et l'odorat (visuelle, sonore ou haptique). La réalité virtuelle est à ne pas confondre avec la réalité augmentée. La réalité augmentée ajoute des éléments virtuels dans un environnement réel alors que la réalité virtuelle crée virtuellement un environnement réel ou imaginaire. [Davis *et al.*, 2019]

C'est cette synergie de technologie qui révolutionnera nos usages. Ainsi, dans un avenir proche nous vivrons dans un monde où nous pourrons non seulement interagir avec tous nos appareils à distance et en temps réel, mais nos appareils pourront également interagir entre eux, s'échanger des données et les analyser pour nous proposer toujours plus de services et de fonctionnalités parfaitement adaptés à nos besoins. Preuve de la révolution à venir, Microsoft ouvre à Munich un laboratoire spécialisé dans l'IoT et l'intelligence artificielle.

On ne peut améliorer que ce qu'on peut mesurer. Or, l'IoT peut tout mesurer à distance, de manière instantanée et automatisée. C'est en cela que c'est une technologie révolutionnaire.

**La domotique.** Cette technologie permet de gérer à distance de nombreuses fonctionnalités de la maison : ouverture, fermeture des volets, mise en marche du système de chauffage, d'appareils électriques, de la lumière, etc.



Figure 1-4: maison connecté (smart house) [ LEGENDRE IMMOBILIER, 2018]

**La voiture autonome.** La voiture autonome est une voiture qui peut se déplacer sans l'intervention d'un conducteur. Les premiers à l'avoir lancée sont Google, en 2010. De nombreux freins empêchent encore sa démocratisation, tels le coût, la lenteur actuelle des véhicules et les problèmes liés à la sécurité.



Figure 1-5: Première voiture autonome [Zaffagni Marc, 2017]

**Le piège à souris connecté.** Son usage est industriel. Il permet de savoir à distance quand une souris s'est fait prendre, on peut donc intervenir de manière ciblée, sans avoir à vérifier tous les pièges à souris d'un bâtiment à intervalles réguliers et sans risquer de laisser une souris dans son piège.



Figure 1-6: piège à souris connecté [Maxime, 2015]

**Les bennes connectées.** Aujourd'hui, on fonctionne avec un système de ramassage de poubelles régulier, quelque soit le taux de remplissage des bennes à ordures. Le fait d'avoir des bennes connectées permet d'optimiser l'enlèvement des ordures qui, plutôt que de se faire à intervalles réguliers, se fait dès qu'il y en a besoin et seulement s'il y en a besoin.



Figure 1-7: les bennes connectés [EASY SECURITY, 2019]

### **La brosse à dent connectée.**

Elle collecte des données sur votre façon de vous brosser les dents et les analyse. Cela permet d'inculquer de bonnes habitudes de brossage de dents aux enfants et pourquoi pas à leurs parents.



Figure 1-8: brosse à dent connectée [Oral B, 2014]

### **Les clés connectées.**

Avouez-le, ça vous est forcément déjà arrivé de chercher vos clés partout au moment de quitter la maison. Avec les clés connectées, ça ne vous arrivera plus jamais puisque votre téléphone vous permettra de les localiser où qu'elles soient !



Figure 1-9: les clés connectés [Ishrak Jahan , 2018]

**Focus sur LoraWan, technologie française.** LoRa est un protocole IoT très longue portée, inventé en France. L'acronyme signifie Long Range Wide-area network (réseau étendu à longue portée). C'est un réseau sans fils basse consommation, bidirectionnel, bas débit et longue portée, particulièrement adapté aux applications de l'IoT qui ne nécessitent pas un débit élevé. Cette technologie permet de recevoir et d'envoyer des informations ou des commandes pour un coût bien moindre que le cellulaire. De plus, LoRa utilise des fréquences libres et gratuites, n'importe qui peut donc déployer et monétiser un réseau LoRaWan sans avoir à acheter la connectivité à une entreprise publique. Nul doute que les solutions de type LoraWan contribueront à la démocratisation de l'IoT et à son adoption de masse. [Nadai Cécile, 2015]

## **1.5 Avantages**

L'IOT est une révolution technique qui va intégrer les objets connectés, permettre l'apparition



d'un réseau ubiquitaire et nous donner un avantage quotidien. C'est un concept ayant des répercussions sur les technologies et la société dans beaucoup de domaines: des secteurs privés, étatiques, industriels et sanitaire... Elle permettra de rendre l'environnement connecté et pouvoir communiquer avec lui, à l'avenir nous serons informés de l'état du sol, de l'humidité et de la quantité de lumière reçu, ce simple cas, permet de nous donner un aperçu global sur son potentiel et sur ces avantages.

Prenons par exemple le cas de la domotique. En gérant les volets selon la saison, ainsi que le chauffage, le système domotique vous permet d'économiser de l'énergie, et donc de l'argent, même si au départ on ne recherchait que le confort en plus. La consommation d'énergie peut être suivie très finement, qu'il s'agisse de votre consommation d'électricité, d'eau, ou même de gaz. Le simple fait d'activer l'alarme en partant va passer le chauffage en mode économie, et éteindre toutes les lampes et les appareils restés en veille, réduisant ainsi votre consommation d'énergie en votre absence. Et ceci sans aucune action de votre part. Ce pendant toute technologie à des bienfaits mais aussi des méfaits.

## **1.6 Inconvénients**

L'IOT gère nos données personnelles, en effet, les objets connectés produisent de grande quantités d'information et le traitement de cette masse de données implique de nouvelle préoccupation notamment autour de la confidentialité et de la sécurité.

## **1.7 Enjeux de l'IoT**

L'Internet of Things est une vague technologique qui transforme en profondeur entreprises, économie et société. Au-delà des objets du quotidien (smartphones, domotique), l'IoT bouleverse d'abord la façon dont les entreprises, administrations et collectivités fonctionnent via ses capacités de connectivité et temps-réel, et apporte en effet son lot d'innovations significatives : [Club IoT, 2016]

**La consommation électrique.** Les objets connectés consomment beaucoup d'énergie. Comment pourra-t-on faire face aux besoins énergétiques des 20 milliards d'objets connectés en circulation d'ici 2050 ? Si des solutions existent, telle que l'utilisation de piles rechargeables à l'énergie solaire ou l'adoption de technologies utilisant moins d'énergie que d'autres, cela reste un enjeu majeur.

**La gestion des capteurs.** Aujourd'hui, nous ne possédons que peu d'objets connectés. Mais lorsque chacun d'entre nous possédera 10 ou 20 objets connectés, la gestion des capteurs nécessaires à leur fonctionnement deviendra problématique. Il sera indispensable de développer un système de supervision aussi automatisé que possible et de repenser l'architecture et l'infrastructure de l'IoT.

**L'uniformisation des standards techniques.** L'IoT est un marché très fragmenté qui ne désigne pas une seule technologie mais des dizaines de standards fonctionnant différemment. En terme d'expérience utilisateur, c'est très complexe. Or, il ne peut y avoir d'adoption de masse d'une technologie trop complexe à maîtriser. Il faut donc effacer cette complexité, soit par l'uniformisation des pratiques, soit en équipant les box internet de plusieurs technologies prenant en charge tous les objets connectés de la maison.

**La gestion des données.** À l'avenir, l'IoT va créer un océan de data gérable uniquement grâce à l'IA. À titre d'exemple, sur un avion de ligne, il y a 20 000 capteurs et une simple turbine d'A380 fournit 10G de données à chaque vol ! Comment les gérer et donner du sens à ces milliards de données ? Il faudra donc compter sur l'intelligence artificielle et notamment le machine learning pour y parvenir. [Nadai Cécile, 2015]

## **Chapitre 2: Conception de la plate-forme**

Pour la modélisation de notre plate-forme, nous avons jugé nécessaire de ne présenter que les diagrammes de cas d'utilisation, quelques diagrammes de séquences et le diagramme de classes.

### **2.1 Analyse des besoins**

Elle consiste à identifier l'ensemble des fonctionnalités du système sommairement définies dans un cahier des charges, dans l'optique de mieux comprendre son sujet en décomposant ainsi le problème en sous problèmes élémentaires faciles à résoudre. Dans cette partie nous allons donc présenter les besoins principaux couverts par notre plate-forme. C'est-à-dire l'architecture fonctionnelle et applicative que peut offrir notre plate-forme aux utilisateurs ainsi que ses limites pour garantir la réussite et l'efficacité de notre projet.

Notons aussi que l'analyse des besoins fait apparaître les différents acteurs jouant chacun un rôle prépondérant et pouvant interagir avec le système en question.

#### **2.1.1 Besoins fonctionnels**

Les besoins fonctionnels constituent les actions exécutables par notre plate-forme. Une application a pour vocation première de gérer les cas d'utilisations qui ne sont rien d'autre que l'ensemble des fonctionnalités du système. Ainsi, nous pouvons comprendre par-là que le système ne doit pas tout autoriser pour plus de sérieux et de crédibilité. Ces derniers sont :

- Ajouter supprimer et modifier des objets connectés.
- Ajouter supprimer et modifier les réseaux des objets connectés.
- Ajouter supprimer et modifier les abonnés de notre plate-forme.
- Ajouter supprimer et modifier les stations de base de notre réseau.
- Ajouter supprimer et modifier les zones de couverture de notre réseau.
- Ajouter supprimer et modifier les bâtiments qui sont dans les zones de couverture de notre réseau.

- Ajouter supprimer et modifier les localisations qui sont dans les bâtiments.
- Visualiser l'état des objets.
- Allumer ou éteindre un objet.
- Visualiser la consommation d'énergie d'un objet à l'instant t.
- Générer un rapport sur la consommation d'énergie d'un objet par rapport à son temps d'activité.
- Envoyer des messages d'alertes aux abonnés quand leur consommation d'énergie est très élevée.

### **2.1.2 Besoins non fonctionnels**

Ce sont les besoins qui concerne notre système et qui ont souvent des contraintes internes ou externes. Les contraintes constituent alors les bornes, les limites ou le domaine de définition d'une application web. Elles sont intéressantes une fois qu'on sait les gérer car elles permettent d'éviter les bugs. Ces derniers sont:

- La fiabilité de la communication entre les données reçues et la plate-forme.
- La cohérence des données reçues.
- La rapidité des traitements: Avec le nombre important des transactions quotidiennes, il est impérativement nécessaire que la durée d'exécution des traitements s'approche le plus possible du temps réel.
- Garantir la performance à travers ses fonctionnalités c'est-à-dire la plate-forme répond à toutes les exigences des usagers d'une manière optimale.
- Garantir un rapport positif entre les interfaces et les besoins des utilisations.

L'ensemble de ces besoins nous permettra de passer à la phase d'analyse des cas d'utilisation.

## **2.2 Rôle des acteurs et Description des fonctionnalités**

Un acteur représente un rôle d'un utilisateur qui interagit avec le système que vous modélisez. L'utilisateur peut être un utilisateur humain, une organisation, une machine ou un autre système externe.

Notre système compte deux principaux acteurs (administrateur et abonné) qui doivent

nécessairement s'authentifier pour avoir accès à leurs pages dédiées.

<b>Acteurs</b>	<b>Cas d'utilisation</b>	<b>Rôles</b>
<b>Utilisateurs</b>	- Authentification	- Connexion - Déconnexion
<b>Administrateur</b>	- Gérer (comptes utilisateurs, zones, bâtiments, localisations, objets connectés)  - Visualiser détails objets  - Valider ou Invalider les demandes d'abonnement  - Envoyer le rapport de consommation d'énergie à l'abonné	- Mise à jour
<b>Abonné</b>	- Lister (zones, bâtiments, localisations, objets connectés)  - Visualiser (les objets et leur statistiques)  - Manipuler les objets	- Client

Table 2-1: les acteurs, leur fonctionnalité et leur rôle

## **2.3 Diagrammes de cas d'utilisation**

En langage UML, les diagrammes de cas d'utilisation modélisent le comportement d'un système et permettent de capturer les exigences du système. Les diagrammes de cas d'utilisation décrivent les fonctions générales et la portée d'un système. Ces diagrammes identifient également les interactions entre le système et ses acteurs. Les cas d'utilisation et les acteurs dans les diagrammes de cas d'utilisation décrivent ce que le système fait et comment

les acteurs l'utilisent, mais ne montrent pas comment le système fonctionne en interne. Les diagrammes de cas d'utilisation illustrent et définissent le contexte et les exigences d'un système entier, ou des parties essentielles d'un système. [ibm, 2017]

La figure ci-dessous représente le diagramme de cas d'utilisation d'un utilisateur. En effet, un utilisateur c'est quelqu'un qui interagit avec le système, il peut être un administrateur ou un abonné. La figure 2-10 illustre les cas d'utilisations que l'administrateur et l'abonné ont en commun.

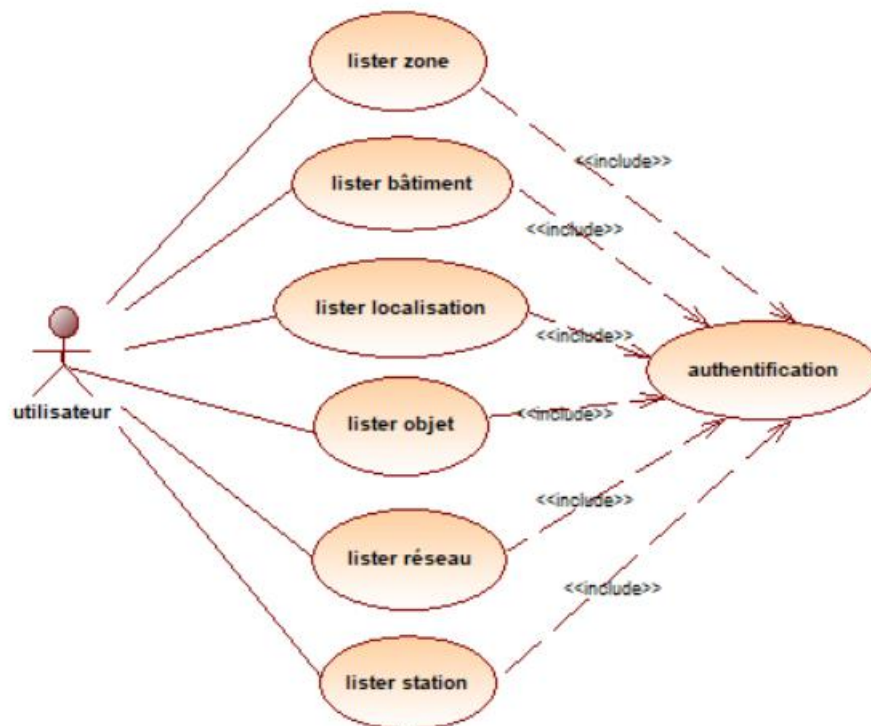


Figure 2-1: diagramme de cas d'utilisation d'un utilisateur

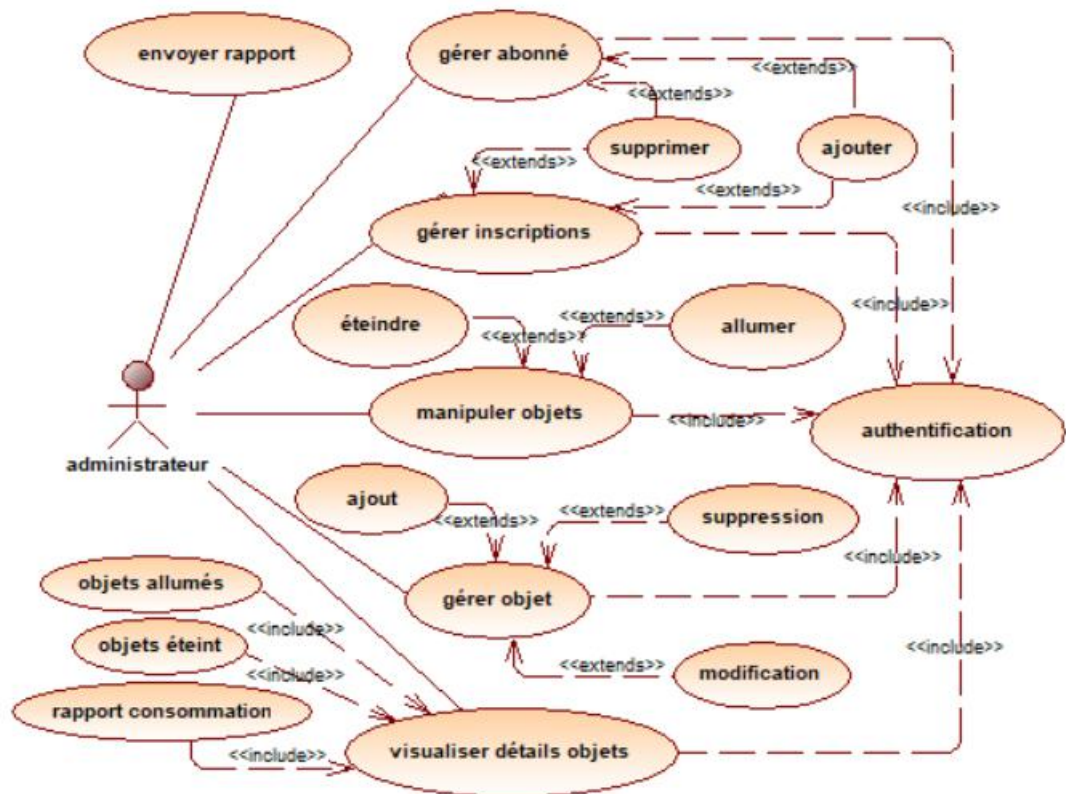


Figure 2-2: diagramme de cas d'utilisation de l'administrateur

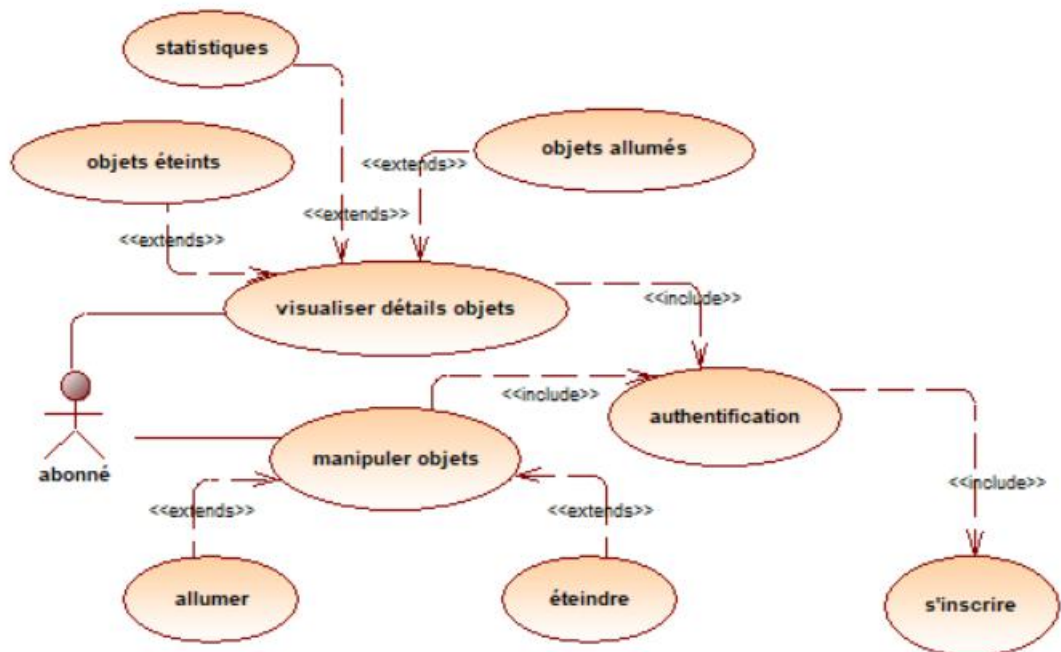


Figure 2-3: diagramme de cas d'utilisation de l'abonné

## **2.4 Description des différents scénarios et gestion des exceptions**

Dans ce qui suit, nous décrirons de façon détaillée certains cas d'utilisations identifiés précédemment en recensant de façon textuelle toutes les interactions entre les acteurs et le système.

### **➤ Cas d'utilisation Authentification**

Sommaire d'identification		
Titre du cas d'utilisation	Authentification	
Résumé	L'authentification permet d'accéder à des fonctionnalités réservées à un type d'utilisateur donné.	
Acteurs	Administrateur, Abonné	
Description des scénarios		
Préconditions	- Application accessible.	
Scénario nominal	1. L'utilisateur accède à la page d'authentification.  2. L'utilisateur choisit sa catégorie (administrateur ou abonné).  3. Le système affiche le formulaire d'authentification.	4. L'utilisateur saisit son nom d'utilisateur et son mot de passe.  5. Le système vérifie l'existence du compte.  6. Le système renvoie l'interface correspondante.
Enchaînements d'erreur	5a. Aucun compte correspondant au couple nom d'utilisateur/mot de passe indiqué :  le système lève une exception ; le cas d'utilisation se termine en échec.	



Postconditions	L'utilisateur est authentifié et accède aux fonctionnalités qui lui sont dédiées.
----------------	---

Table 2-2: scénario du diagramme de séquence «authentification»

➤ **Cas d'utilisation Demande abonnement**

Sommaire d'identification		
Titre du cas d'utilisation	Effectuer prés-inscription	
Résumé	Le client demande à être un abonné de la plate-forme à travers une prés-inscription .	
Acteurs	Demandeur abonnement	
Description des scénarios		
Préconditions	- Application accessible.	
Scénario nominal	1. Le client demande le formulaire d’ajout.  2. Le système renvoie le formulaire.  3. Le client remplit le formulaire.	4. Le client valide et envoie la demande.  5. Le système lui envoie un message confirmant l’action et envoie la demande à l’administrateur.
Enchaînements d'erreur	4a. Le client annule la demande:  le cas d'utilisation se termine en échec.	

Postconditions	Une nouvelle demande est ajoutée à la liste.
----------------	--

Table 2-3: scénario du diagramme de séquence «demande abonnement»

Pour documenter les cas d'utilisations, la description textuelle est indispensable car elle seule permet de communiquer facilement avec les utilisateurs et de s'entendre sur la terminologique employée.

En revanche, le texte présente des limites puisqu'il est difficile de montrer comment les enchaînements se succèdent. Il est donc recommandé de compléter la description textuelle par un ou plusieurs diagrammes dynamiques UML (diagramme de séquence, d'activité, d'état-transition ou encore de collaboration etc)

## **2.5 Diagrammes de séquences**

Dans les modèles UML, une interaction est un comportement qui représente la communication entre un ou plusieurs participants. Un diagramme de séquence est un diagramme d'interaction UML qui modélise les messages transmis entre les participants, comme les objets ou les rôles, ainsi que les structures conditionnelles et de commande, comme les fragments combinés. [ibm, 2017]

### **➤ Diagramme de séquence “authentification”**

Le premier scénario pour l'utilisateur consiste à s'authentifier auprès du système. La chronologie de ce scénario est représentée par la figure ci-dessus.

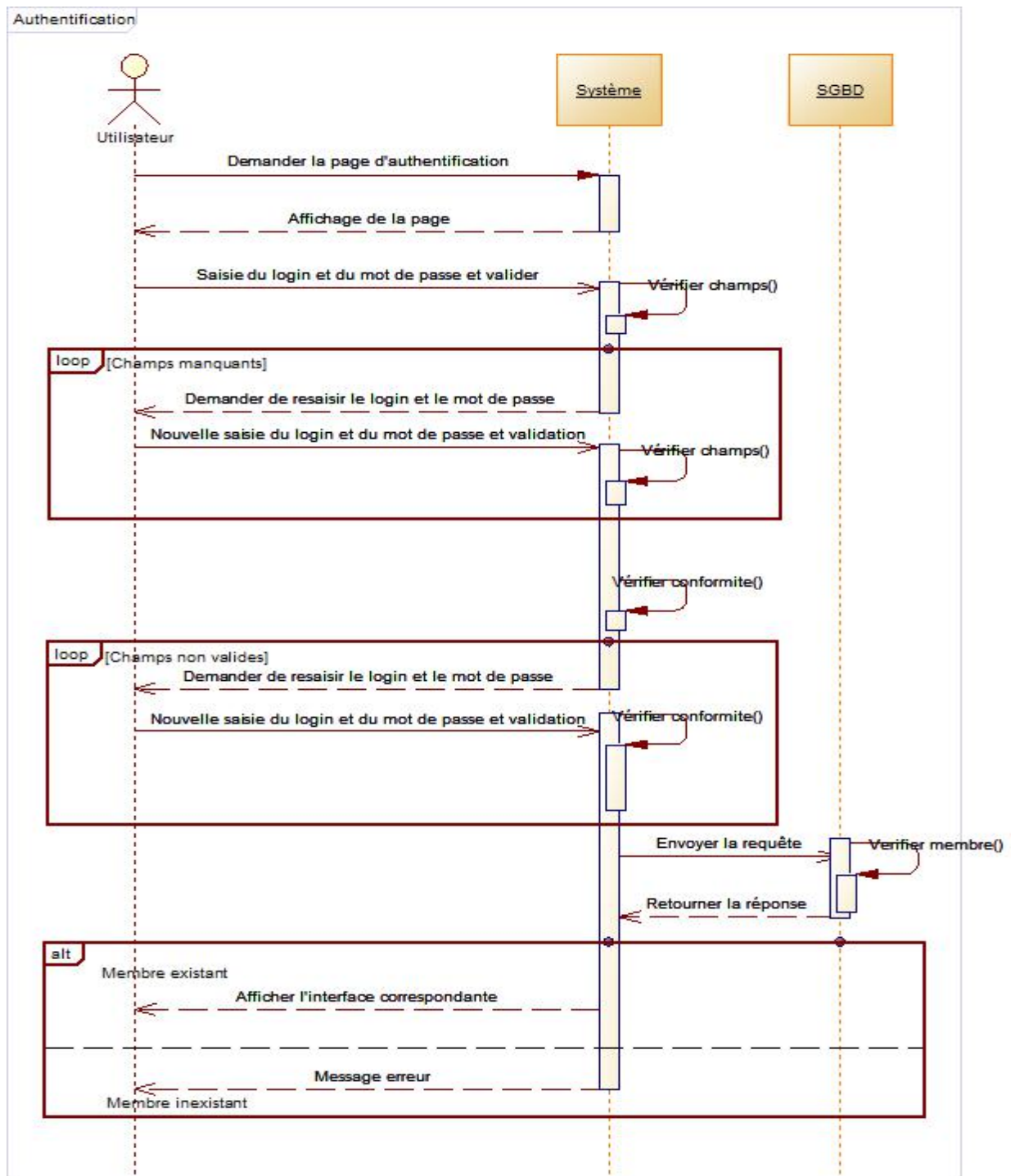


Figure 2-4: diagramme de séquence de l'authentification

### ➤ Diagramme de séquence “demande abonnement”

Le scénario demande abonnement se fait selon la chronologie ci-après.

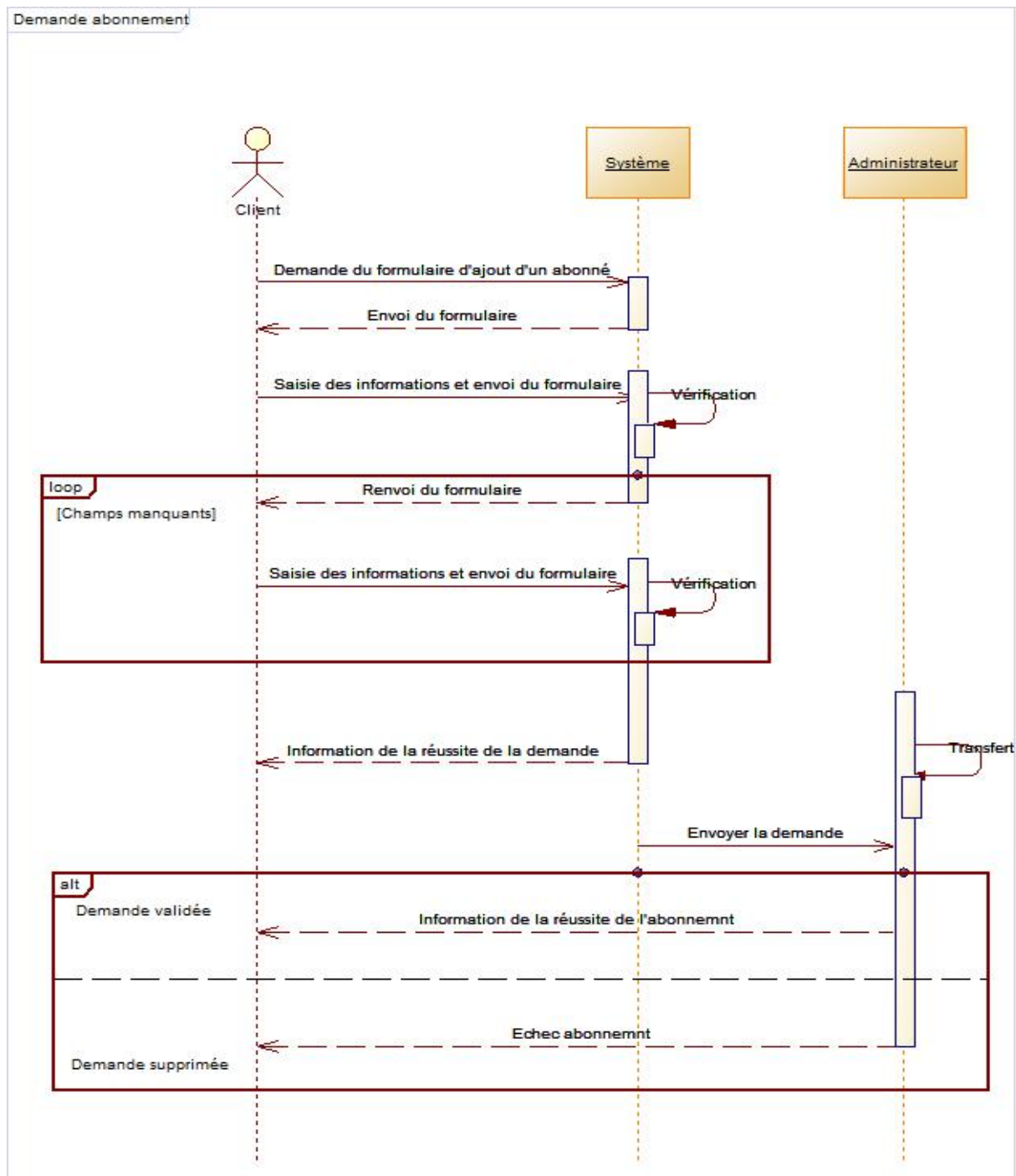


Figure 2-5: diagramme de séquence du demande d'abonnement

# **Chapitre 3: Réalisation de la plate-forme**

Ce chapitre est consacré à la partie pratique de la réalisation de la plate-forme. Il comporte une présentation du diagramme de classe et des outils de développement utilisés ainsi qu'une brève présentation de la plate-forme. Le choix de nos outils de développement s'est fondé principalement sur leur gratuité et l'open source. Effectivement, en implémentant une solution qui se base sur des technologies gratuites et open source, nous avons plus de chance d'éviter toutes sortes de problèmes liés aux licences, contrats, etc., réduisant ainsi les coûts.

## **3.1 Diagramme de classe**

**QUI ?**

Les diagrammes de cas d'utilisation modélisent à **QUOI** sert le système. Le système est composé d'objets qui interagissent entre eux et avec les acteurs pour réaliser ces cas d'utilisation :

Les diagrammes de classes permettent de spécifier **QUI** intervient à l'intérieur du système

- Il spécifie également quels liens peuvent entretenir les objets du système Chaque nouveau diagramme répond à une question différente :
- Ils sont complémentaires et pas du tout interchangeable
- Ils doivent être cohérents les uns avec les autres
- C'est en répondant à toutes les questions qu'on définit complètement un système.

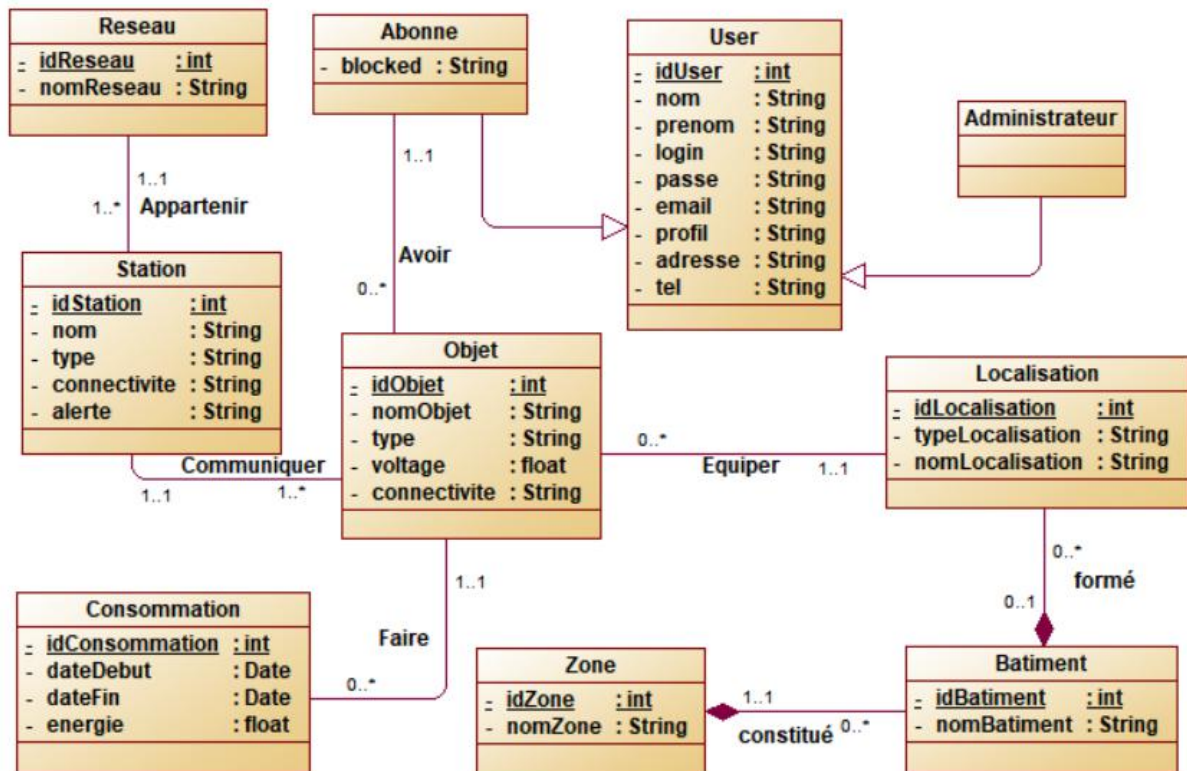


Figure 3-1: diagramme de classe

## 3.2 Technologies utilisées

Dans cette section, nous présentons le langage de programmation Java EE que nous avons d'ailleurs adopté pour réaliser la plateforme et les différentes API qui ont servis à son développement sans oublier les logiciels de modélisation comme Power AMC etc.

### 3.2.1 Java EE

JEE est un ensemble de technologies, créées par SUN Microsystems, permettant de réaliser des applications WEB mais aussi des applications standalone, en utilisant le langage JAVA. Cette édition est dédiée à la réalisation d'applications pour entreprises. JEE est basée sur J2SE (Java 2 Standard Edition) qui contient les API de bases de java.



Figure 3-2: Logo Java EE [Globlehi Philemon, 2019]

JEE est une plate-forme fortement orientée serveur pour le développement et l'exécution d'applications distribuées.

Elle est composée de deux parties essentielles :

- un ensemble de spécifications pour une infrastructure dans laquelle s'exécute les composants écrits en java : un tel environnement se nomme serveur d'application.
- un ensemble d' API qui peuvent être obtenues et utilisées séparément. Pour être utilisées, certaines nécessitent une implémentation de la part d'un fournisseur tiers.

Cet ensemble regroupe les servlets, JSP, services web,....

JEE fournit un ensemble d'API permettant de construire des sites WEB avec la technologie JAVA. Pour cela, on utilise un serveur d'application JEE:

Un serveur d'application permet de charger et d'exécuter les servlets, EJB, ... dans une JVM.

C'est une extension du serveur web.

Ce serveur d'application contient entre autre un moteur de servlets qui se charge de manager les servlets qu'il contient.

L'utilisation d'un IDE est obligatoire comme Eclipse ou JBuilder, ou NetBeans. Dans ce projet, nous avons utiliser NetBeans, vu sa gratuité et sa robustesse. [Ndong Joseph, 2007]

J2EE regroupe un ensemble d'API pour le développement d'applications d'entreprise. Ces API peuvent être regroupées en trois grandes catégories :

- les composants : Servlet, JSP, EJB
- les services : JDBC, JTA/JTS, JNDI, JCA, JAAS
- la communication : RMI-IIOP, JMS, JavaMail

Toutefois nous nous limiterons à la présentation des API utilisés pour la conception de la plateforme. Parmi ces derniers nous avons:

### ➤ JSP/JSTL

JSP est une technologie de JEE qui permet d'insérer du code Java dans du HTML qui sera interprété au moment de la compilation par le serveur JEE. Néanmoins l'insertion directe de code Java dans le code HTML étant déconseillé et assez peu lisible, JSP fait intervenir des balises permettant d'accéder aux variables des paramètres passés à la page web. JSP peut être combiné avec JSTL, qui est une extension de JSP qui permet d'effectuer des actions un peu plus développées que l'accès aux variables, on peut par exemple réaliser des boucles conditionnelles, ou appeler des fonctions sur des chaînes de caractère, ... Il faut tout de même savoir que JSP et JSTL restent du Java, mais « masqué » derrière des macros.

Voici une comparaison de JSP/JSTL face à du Java simple :

```
1 <c : forEach var=" ab" items="{ nomAbonne }" >
```

```
2 <c : out value="{ ab}" />
```

```
3 </c : forEach>
```

Classe Java faisant office de contrôleur dans le modèle MVC

```
1 List<Integer>id =
```

```
2 ( ArrayList<Integer>) request . getAttribute ( " id" );
```

```
3 for ( int i = 0 ; i < list . size ( ) ; i++){
```

```
4 out . println ( id. get ( i ) ) ;
```



5 }

On remarque que le premier code est plus abstrait que du Java. Les pages JSP étant destinées à un designer par exemple, cette technologie est plus abordable que du Java. Dans le précédent exemple, on peut voir une balise avec le préfixe « c », qui est le préfixe d'appel à la bibliothèque *Core* de JSTL qui fournit les actions de base, comme le parcours de liste (ligne 1 : `forEach`), l'encodage d'URL, l'affichage de variables (ligne 2 : `out`)... JSTL est composé, dans sa version 1.1, de 5 bibliothèques permettant de réaliser des actions orientées sur certains domaines.

Les autres bibliothèques sont :

- Format : les opérations sur les dates
- XML : le traitement de balises XML
- SQL : les interactions avec les bases de données
- Fonctions : les manipulations des chaînes de caractères

#### ➤ Servlet

Un servlet est une classe JAVA côté serveur qui reçoit des données HTTP et qui opère un ou des traitements et devant respecter les contraintes de ce protocole HTTP.

Exemple:

```
package servlet;
```

```
import java.io.IOException;
```

```
import javax.servlet.ServletException;
```

```
import javax.servlet.annotation.WebServlet;
```

```
import javax.servlet.http.HttpServlet;
```

```
import javax.servlet.http.HttpServletRequest;
```

```
import javax.servlet.http.HttpServletResponse;
```

```
**
```

```
*
```

```

* @author LENOVO

*/

@WebServlet(name = "ExempleServlet", urlPatterns = {"/ExempleServlet"})

public class ExempleServlet extends HttpServlet {

    protected void processRequest (HttpServletRequest request, HttpServletResponse response)

        throws ServletException, IOException{

        response.setContentType("text/html;charset=UTF-8")

        /* TODO output your page here. You may use following sample code. */

        out.println("<!DOCTYPE html>");

        out.println("<html>");

        out.println("<head>");

        out.println("<title>Servlet ExempleServlet</title>");

        out.println("</head>");

        out.println("<body>");

        out.println("<h1>Servlet ExempleServlet at " + request.getContextPath() + "</h1>");

        out.println("</body>");

        out.println("</html>");

    }

}

@Override

protected void doGet(HttpServletRequest request, HttpServletResponse response)

    throws ServletException, IOException {

```

```

processRequest(request, response);

}

@Override

protected void doPost(HttpServletRequest request, HttpServletResponse response)

throws ServletException, IOException {

    processRequest(request, response);

}

}

```

On remarque deux méthodes dans cette classe : doGet et doPost, la première répond par HTTP à l'envoi d'une request GET, la seconde à l'envoi d'une request POST. Comme l'on veut que dans les deux cas la servlet réponde, doPost renvoie à doGet. [Wikibooks, 2016]

### ➤ **Java Mail**

Le courrier électronique repose sur le concept du client/serveur. Ainsi, l'utilisation d'e-mails requiert deux composants :

- un client de mails (Mail User Agent : MUA) tel que Outlook, Messenger, Eudora, ...
- un serveur de mails (Mail Transport Agent : MTA) tel que SendMail

Les clients de mails s'appuient sur un serveur de mails pour obtenir et envoyer des messages. Les échanges entre clients et serveurs sont normalisés par des protocoles particuliers.



Figure 3-3: Logo Java Mail [techplanet, 2018]

JavaMail est une API qui permet d'utiliser le courrier électronique (e-mail) dans une application écrite en Java (application cliente, applet, servlet, EJB, ... ). Son but est d'être facile à utiliser, de fournir une souplesse qui permette de la faire évoluer et de rester le plus indépendant possible des protocoles utilisés. [Doudoux Jean-Michel, 2017]

#### ➤ JDBC

JDBC est une API Java disponible depuis la version 1.1 du JDK. Pour note, JDBC est un nom déposé et non un acronyme, même si en général on lui donne la définition suivante : **Java DataBase Connectivity**.



Figure 3-4: Logo JDBC [Valdes Amaury, 2011]

Cette API est constituée d'un ensemble d'interfaces et de classes qui permettent l'accès, à partir de programmes Java, à des données tabulaires (c.-à-d. triées sous forme de table ou de tableur). Par données tabulaires, on entend généralement des bases de données contenues dans des SGBD relationnels. Mais, JDBC n'est pas restreinte à ce type de source de données. On peut aussi accéder à des sources de données sous forme de fichiers (fichiers XML par exemple).

**L'API JDBC permet, entre autres :**

- l'établissement d'une connexion avec le SGBD ;
- l'envoi de requêtes SQL au SGBD, à partir du programme Java ;
- le traitement, au niveau du programme, des données retournées par le SGBD ;
- le traitement des métadonnées de la connexion, de l'instruction ou des résultats ;
- le traitement des erreurs retournées par le SGBD lors de l'exécution d'une instruction.

[FAQ JDBC, 2015]

### **3.2.2 JavaScript**

Pour les interfaces utilisateurs nous avons souhaité de l'interactivité, nous avons donc opté pour JavaScript. C'est un langage de programmation de script qui s'exécute généralement dans un navigateur, il est compatible avec les plus répandus, sous réserve d'une version relativement récente. Il a également l'avantage de s'exécuter côté client et donc de limiter la charge du serveur. De plus, il existe beaucoup de frameworks et de bibliothèques pour JavaScript, que nous avons largement utilisé pour le développement de l'application, et plus particulièrement deux:

- JQuery : Un framework très utile pour les interactions JavaScript/HTML
- Highcharts : Une bibliothèque pour réaliser des graphiques interactifs



Figure 3-5: Logo JavaScript [Cleanpng, 2018]

### **3.2.3 WampServer**

WampServer est une plate-forme de développement Web sous Windows pour des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Il possède également PHPMysqlAdmin pour une gestion plus facilement des bases de données. [Wikipédia, 2019]



Figure 3-6: Logo WampServer [Wikipédia, 2019]

### **3.2.4 POWER AMC**

POWER AMC est l'un des outils majeurs de modélisation des données et des processus. Il a été créé par la société Sybase, mais est désormais propriété de SAP. [Sap PowerDesign, 2017]



Figure 3-7: Logo PowerAmc [Sap PowerDesign, 2017]

### **3.2.4 UML**

Le Langage de Modélisation Unifié, de l'anglais Unified Modeling Language (**UML**), est un langage de modélisation graphique à base de pictogrammes conçu pour fournir une méthode normalisée pour visualiser la conception d'un système. Il est couramment utilisé en développement logiciel et en conception orientée objet.



### 3.3 Présentation de la plate-forme

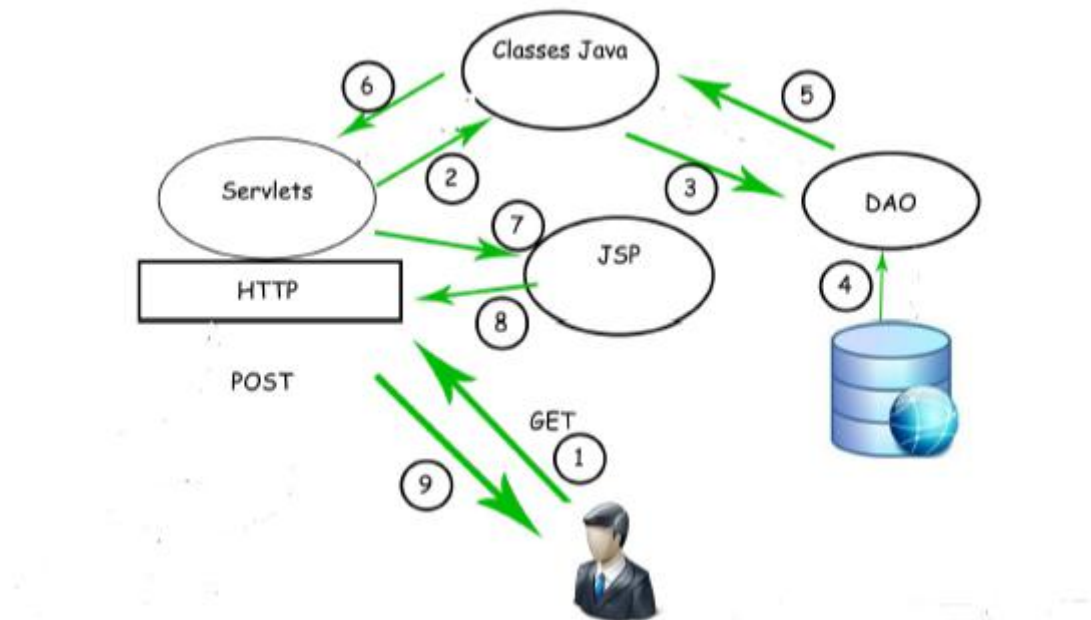


Figure 3-9: Architecture de l'application

Le schéma ci-dessus représente la structure générale de l'application ainsi que le parcours des informations dans celle-ci en fonction des utilisations. Lorsqu'un utilisateur interagit avec l'application, il passe toujours à travers les servlets via le protocole HTTP. Les servlets, interfaces entre l'utilisateur et le reste de l'application, sont des contrôleurs. En fonction de la page appelée une servlet différente peut-être appelée, puis celle-ci en fonction des paramètres envoyés par le navigateur va interagir avec certaines classes de l'application (étape 2). Ces classes réalisent des traitements, ils peuvent aller de la simple conversion à la récupération de données dans la base de données via le DAO (étapes 3, 4 et 5) et retourne le résultat à la servlet (étape 6). La servlet redirige vers la page JSP correspondante avec les éventuelles données traitées (étape 7) puis la page est renvoyée vers l'utilisateur via HTTP (étape 8 et 9). [Garnier *et al.*, 2014]

La fenêtre ci-dessous constitue la page d'accueil de la plate-forme. Elle permet aux utilisateurs d'accéder à la page de connexion ou d'effectuer une prés-inscription.





Figure 3-10: Page d'accueil de la plate-forme

Voici la page d'authentification de notre plate-forme! Elle gère l'accès aux comptes utilisateurs.

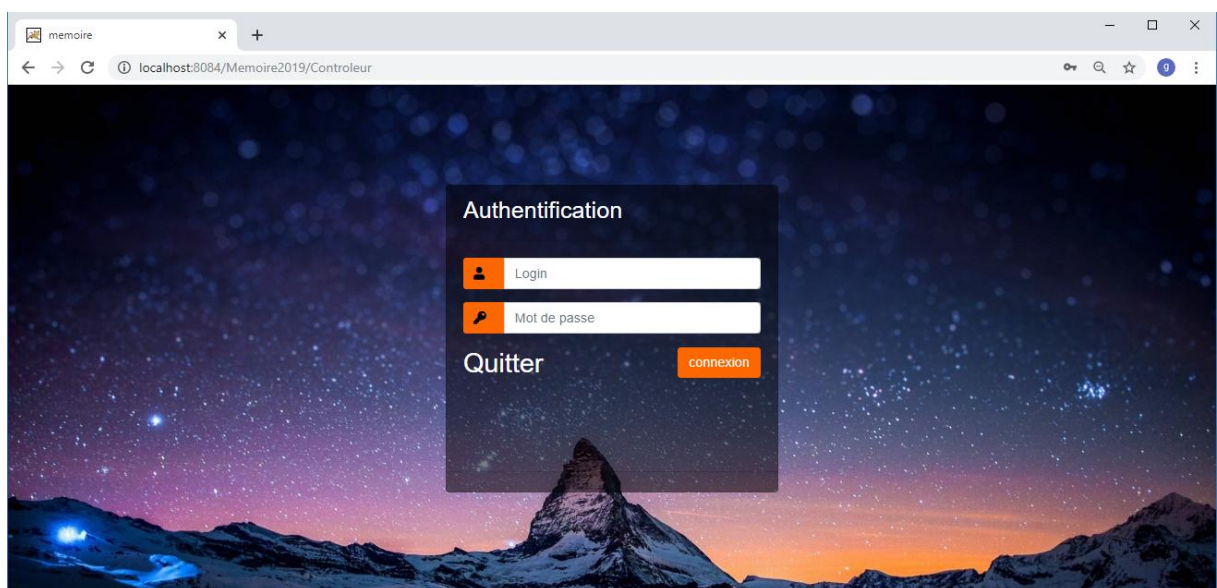


Figure 3-11: Page d'authentification de la plate-forme

La figure suivante est la page d'accueil de l'administrateur. Elle lui permet d'effectuer plusieurs tâches comme: consulter les demandes d'abonnement, faire des ajouts (abonné, objet, zone, bâtiment, localisation, station, réseau...) et de lister les ajouts qu'il a effectués.

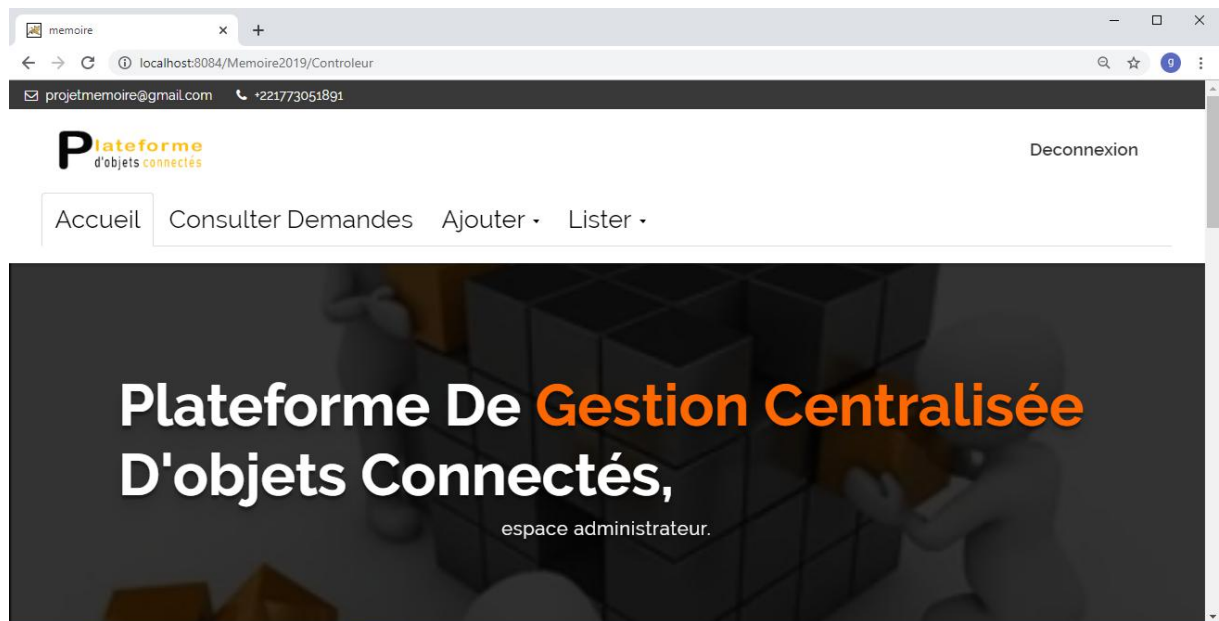


Figure 3-12: Espace administrateur

La fenêtre ci après est la page d'accueil de l'abonné. Elle lui offre certaines fonctionnalités comme visualiser les objets et les détails les concernant (statistiques), les manipuler. Cette fenêtre permet également à l'abonné de mettre à jour ses informations.

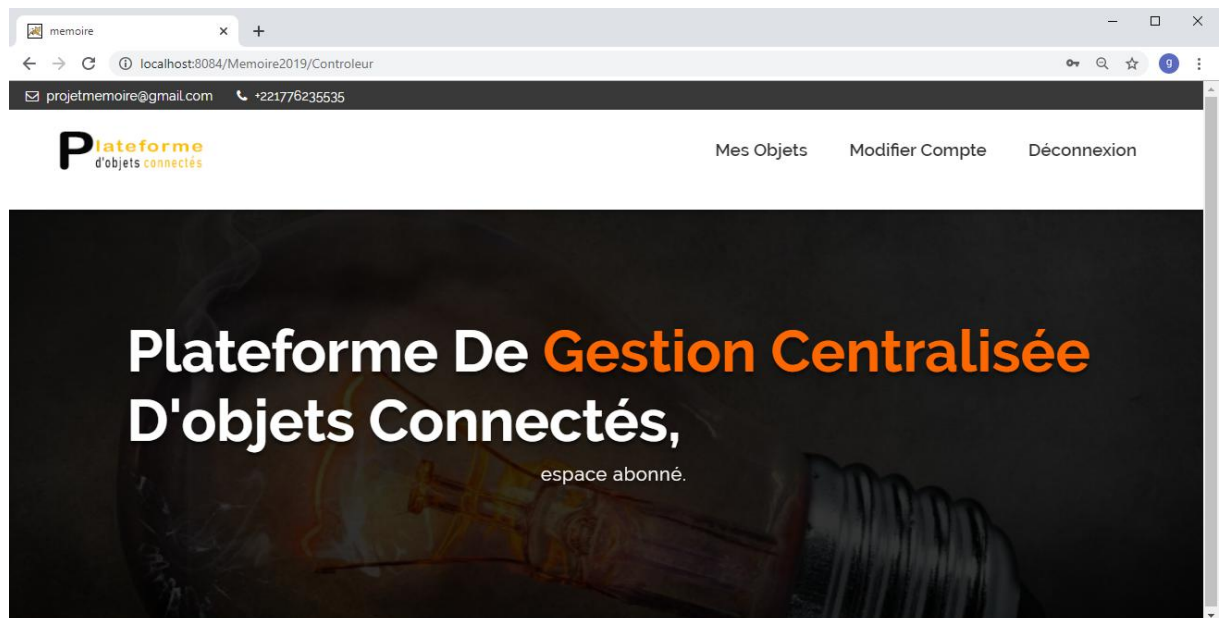


Figure 3-13: Espace Abonné

Cette fenêtre permet à l'abonné de voir ses objets actifs et il a la possibilité de les éteindre. Inversement, il peut avoir accès aux objets éteints et les rallumer. L'abonné peut également

visualiser les statistiques des objets c'est-à-dire il peut suivre l'évolution de sa consommation d'énergie.

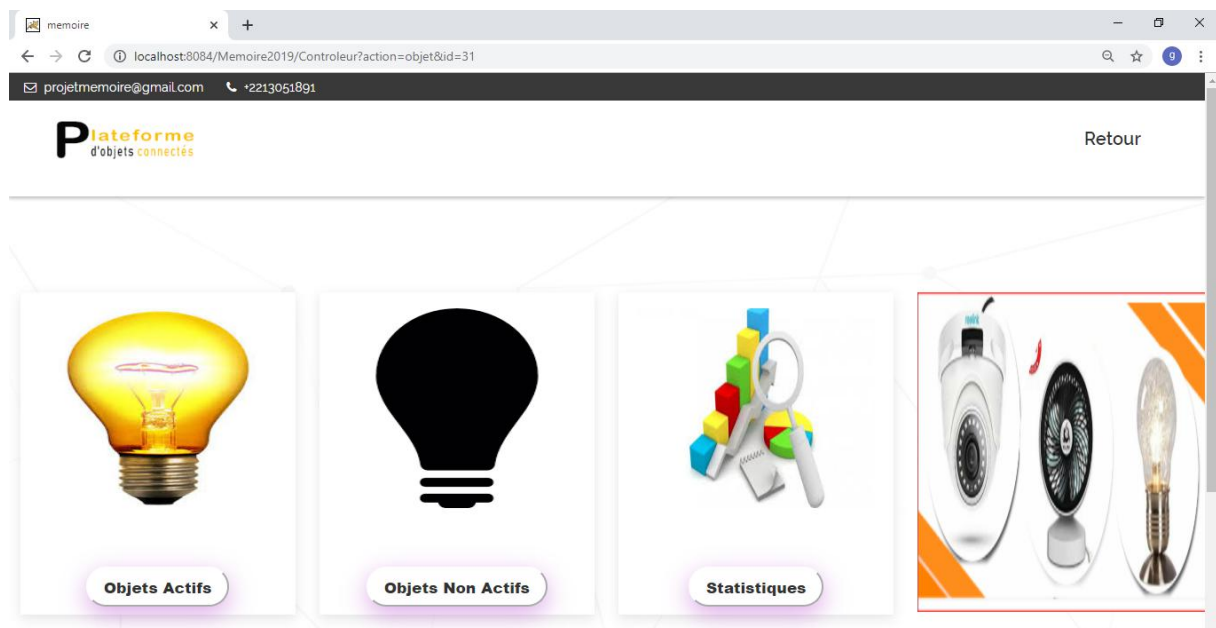


Figure 3-14: Informations sur les objets d'un abonné

La fenêtre ci-dessous montre l'évolution de la consommation d'énergie des objets d'un abonné donné.

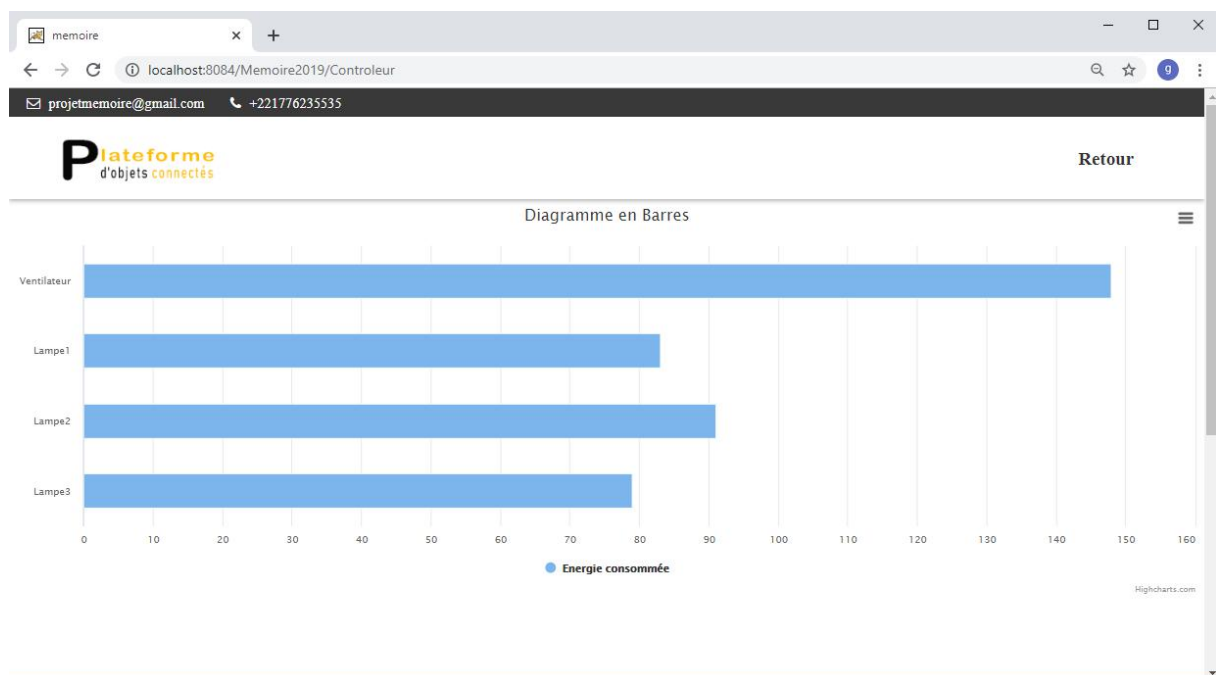


Figure 3-15: Graphe des données issues des capteurs

Ce sont entre autre les fenêtres de notre plate-forme et les fonctionnalités qu'elles gèrent. La création de ces interfaces nous a été très instructif en matière de développement et de programmation grâce aux technologies utilisées citées un peu plus haut.

# **Conclusion**

Toutes les fonctionnalités décrites dans ce mémoire apportent confort et bien-être. Le fait que les objets équipés dans nos maisons et bureaux soient en outre plus sûrs, capables de communiquer et dotés d'un minimum d'autonomie. A travers le projet nous avons appris à nous connaître plus en profondeur. Nous avons réalisé quelque chose de concret, une plate-forme gérant des abonnés et leurs objets. Nous avons également présenter les données issues de ces objets à travers des représentations graphiques. Grâce à ce projet, nous avons découvert le monde de la pratique. Jusqu'à maintenant nous n'avons étudié que la théorie, et il s'avère qu'il s'agit bien de deux mondes différents. L'architecture fonctionnelle ainsi que les méthodes de représentation de la connaissance du domaine ont été présentées en détail, nous avons vu les progrès accomplis dans le domaine de l'IoT depuis son apparition, jusqu'à aujourd'hui. Nous avons également décrit l'évolution de ces technologies dans notre quotidien. On peut dire que l'IoT est constitué d'un système révolutionnaire voué à évoluer encore plus dans le futur. Ce système permet de contrôler l'entièreté d'une localisation à distance. Mais à l'heure actuelle, l'installation d'un tel système est encore très chère donc peu de gens sont équipés de cela. Mais, dans le futur, les prix vont sans doute devenir plus abordables et les gens pourront donc peut-être se permettre de profiter de cette technologie.

En conclusion L'IoT «vend du rêve» : elle nous faciliterait la vie, elle nous permettrait de faire plus de choses et de mieux vivre. Surtout, elle nous donne l'impression que l'on pourrait tout contrôler et tout maîtriser chez soi. Notre projet pour l'avenir vise une diversification des objets, c'est-à-dire héberger d'autres variétés d'objets en plus de ceux que nous avons déjà (lampe, ventilateur) et aussi intégrer de nouvelles fonctionnalités à la plate-forme qui s'en doute intéresseront les abonnés. Ne serait-il pas souhaitable de faire des simulations sur des projets réelles (maison, entreprise, établissement, ferme, jardin, ...) dans le but d'atteindre une nouvelle innovation?

# Références

## ➤ Bibliographie

Pierre-Jean Benghozi, Sylvain Bureau, Françoise Massit-Folea. «L'Internet des objets. Quels enjeux pour les Européens ?». 2008. fhal-00405070. Disponible sur «<https://hal.archives-ouvertes.fr/hal-00405070/document>». [Page consultée le 09 juillet 2019]

Joseph Ndong. «Développement WEB avec JEE (*JavaEnterprise Edition*.)&Tomcat».2007[Cours]

Arthur Garnier, Emmanuel Nataf. «Développement d'une plate-forme de supervision d'un réseau de capteurs». Réseaux et télécommunications [cs.NI]. 2014. hal-01059019. Disponible sur «<https://docplayer.fr/23309959-Developpement-d-une-plate-forme-de-supervision-d-un-reseau-de-capteurs.html>». [Page consultée le 17 Août 2019]

## ➤ Wébographie

*futura tech*, «Internet des objets», *futura-sciences (en ligne)*, 14 Mai 2015. Disponible sur «<https://www.futura-sciences.com/tech/definitions/internet-internet-objets-15158/>»[Page consultée le 02 Juillet 2019]

*Guillaume Plouin, Nicolas Colomer*, «Modèles d'architectures de l'Internet des Objets», *Le Blog des Octos(en ligne)*,13 Septembre 2011.Disponible sur :«<https://blog.octo.com/modeles-architectures-internet-des-objets/>»[Page consultée le 09 Juillet 2019]

*Erik Davis, James Wagner, Ben Lawson, Andras Kemeny*, «Réalité virtuelle», *wikipédia(en ligne)*, 16 Décembre 2019.Disponible sur « [https://fr.wikipedia.org/wiki/R%C3%A9alit%C3%A9\\_virtuelle](https://fr.wikipedia.org/wiki/R%C3%A9alit%C3%A9_virtuelle) » [Page consultée le 17 Décembre 2019]

*LEGENDRE IMMOBILIER*, «La domotique, c'est quoi?», *LEGENDRE IMMOBILIER: L'essentiel sur la domotique (en ligne)*, Jeudi 15 mars 2018. Disponible sur «<https://www.legendre-immobilier.com/actualites/lessentiel-sur-la-domotique>»[Page consultée le 09 Juillet 2019]

*Marc Zaffagni*, «Google tente une première mondiale de la voiture autonome», Kafunel (en ligne), 26 Juillet 2017. Disponible sur «<https://www.kafunel.com/google-tente-une-premiere-mondiale-de-la-voiture-autonome>»[Page consultée le 09 Juillet 2019]

*Maxime*, «Pourquoi réinventer le piège à souris?», RADAR Connect (en ligne), 01 Mai 2015. Disponible sur «<https://www.rentokil.fr/blog/radar-connect-reinvention-du-piege-a-souris/>»[Page consultée le 09 Juillet 2019]

*EASY SECURITY*, «la première benne semi-remorque totalement connectée au monde.», BENALU (en ligne), 2019. Disponible sur «<https://www.benalu.com/easysecurity-une-premiere-mondiale/>»[Page consultée le 09 Juillet 2019]

*Oral B*, «Oral B Brosse à Dents Connectée Bluetooth», Frandroid (en ligne), 21 février 2014. Disponible sur «[https://www.frandroid.com/events/mwc/196098\\_oral-b-presentera-sa-brosse-dents-connectee-au-mwc](https://www.frandroid.com/events/mwc/196098_oral-b-presentera-sa-brosse-dents-connectee-au-mwc)»[Page consultée le 09 Juillet 2019]

*Jahan Ishrak*, «Internet of things IoT», SUPINFO (en ligne), 26 2018 Août à 15:01:42. Disponible sur «<https://www.supinfo.com/articles/single/7358-internet-of-things-iot>»[Page consultée le 18 Octobre 2019]

*Club IoT*, «Les enjeux de l'IoT», siapartners (en ligne), 13 Avril 2016. Disponible sur «<https://www.club-iot.sia-partners.com/enjeux>»[Page consultée le 20 Octobre 2019]

*Nadai Cécile*, «Focus sur l'Internet of Things (IoT)... l'essentiel à savoir», welcom to the jungle (en ligne), 06 juin 2015. Disponible «[www.welcometothejungle.com](http://www.welcometothejungle.com)»[Page Consultée le 13 Octobre 2019]

*ibm*, «Diagramme de cas d'utilisation et Diagramme de séquence», Knowledge center (en ligne). 03 décembre 2017. Disponible sur «<https://www.ibm.com/support/knowledge>»[Page consultée le 29 Octobre 2019]

*Globlehi Philemon*, «Les API Java EE», code divoire-(en ligne), 14 Mai 2019.Disponible sur «<https://medium.com/code-divoire>»[Page consultée le 12 Novembre 2019]

*Wikibooks*, «Programmation JEE/Servlets», Wikibooks (en ligne), 3 Août 2016.Disponible sur «[https://fr.wikibooks.org/wiki/Programmation\\_JEE/Servlets#Les\\_servlets](https://fr.wikibooks.org/wiki/Programmation_JEE/Servlets#Les_servlets)»[Page consultée le 07 Octobre 2019]

*techplanet*, «Java Send Email», techplanet (en ligne), 19 Juillet 2018.Disponible sur «<https://techeplanet.com/java-send-email/>» [Page consultée le 07 Octobre 2019]

*Doudoux Jean-Michel*, «JavaMail», Développons en java (en ligne), 07Juillet 2017.Disponible sur «<https://www.jmdoudoux.fr/java/dej/chap-javamail.htm>»[Page consultée le 07 Octobre 2019]

*Valdes Amaury*, «JDBC Statement – Batch Example», Developer's Corner (en ligne), 22 Mai 2011.Disponible sur «<https://avalides.com/jdbc-statement-batch-example/>»[Page consultée le 07 Octobre 2019]

*FAQ JDBC*, «Généralités», Developpez.com (en ligne), 3 juin 2015.Disponible sur «<https://java.developpez.com/faq/jdbc?page=Generalites>»[Page consultée le 07 Octobre 2019]

*Cleanpng*, «JavaScript Logo», Cleanpng.com(en lign), 25 juin 2018..Disponible sur «<https://www.cleanpng.com/png-javascript-web-development-logo-script-clipart-1375846/>»[Page consultée le 14 Novembre 2019]

*Wikipédia*, «WampServer» Wikipdia (en ligne), 27 Novembre 2019.Disponible sur «<https://fr.wikipedia.org/wiki/WampServer>»[Page consultée le 14 Novembre 2019]

*Sap PowerDesign*, «PowerAMC» Futura (en ligne), 02 février 2017.Disponible sur «<https://www.futura-sciences.com/tech/telecharger/sap-powerdesigner-poweramc-305>»[Page consultée le 05 Novembre 2019]

*Wikipédia*, «UML» Wikipdia (en ligne), 17 Octobre 2019.Disponible sur «<https://fr.wikipedia.org/wiki/WampServer>»[Page consultée le 30 Novembre 2019]