

**Ministère de l'Enseignement Supérieur et de la Recherche Scientifique**

**Université Saad Dahleb de Blida1**

**Faculté des sciences**

**Département informatique**



**MEMOIRE DE MASTER**

**En Informatique**

**Spécialité : Ingénierie du logiciel**

**THEME**

---

---

**Conception et réalisation d'une plateforme de service pour le système de gestion commercial de la distribution SONELGAZ**

---

---

**Réalisé par :**

**ISMAIL Yousra**

**RAHALI Meroua**

Soutenu le 08/10/2020 devant le jury composé de :

**Mme. CHERFA Imane**

**Mme. BOUTOUMI Bachira**

**Encadré par :**

**Mme. ARKAM Meriem**

**Mr. KAID Abdelhak**

Présidente

Examinatrice

**Promotion : 2019/2020**

## **Remerciements**

*On tient tout d'abord à remercier **ALLAH** le tout puissant, le tout miséricordieux, qui nous a donné la force et la patience d'accomplir ce modeste travail.*

*On voudra présenter nos remerciements à nos promoteurs, Mme **ARKAM** et Mr **KAID**. On a eu le privilège de travailler parmi eux et d'apprécier leurs sérieux, compétences et leurs sens du devoir qui nous ont énormément marqués. Leurs conseils, ainsi que leurs remarques pointues nous ont certainement permis de mener à bout notre mémoire de fin d'étude.*

*On tient aussi à remercier Mr **BENBARA** et Mr **SAIDANI** pour leurs aides précieuses. Veuillez trouver ici l'expression de nos respectueuses considérations et notre profonde admiration pour toutes vos qualités scientifiques et humaines. Ce travail est pour nous l'occasion de vous témoigner notre profonde gratitude.*

*Encore un grand merci à toute l'équipe de **ELIT** pour leurs soutiens et leurs conseils avisés qui ont contribué à l'aboutissement de ce rapport.*

*Nous tenons à remercier particulièrement nos très Chers Parents. Tous les mots du monde ne sauraient exprimer l'immense amour qu'on vous porte, ni la profonde gratitude qu'on vous témoigne pour tous les efforts et les sacrifices que vous n'avez jamais cessé de consentir pour notre instruction et notre bien-être.*

*Nous aimerions également exprimer notre gratitude aux membres du jury, devant qui nous avons l'honneur de présenter notre travail et qui ont pris la peine de le lire avec soin ce mémoire pour juger son contenu.*

*Merci à tous qui ont participé de près ou de loin à l'aboutissement de ce travail.*

## **Dédicaces**

*Je dédie ce modeste travail*

***A mes chers parents,***

*Que nulle dédicace ne puisse exprimer ce que je leur dois, pour leur bienveillance, leur affection et leur soutien. Trésors de bonté, de générosité et de tendresse, en témoignage de mon profond amour et ma grande reconnaissance « que DIEU vous garde ».*

***A mes chères sœurs et mes chers frères,***

*Je leur dédie ce modeste travail en témoignage de mon grand amour et ma gratitude infinie.*

***A toutes les personnes que j'aime et qui m'aiment,***

*Pour leur aide et leur soutien moral durant l'élaboration du travail de fin d'études et le comble de ma gratitude et respect vont vers mon fiancé et toute sa famille.*

***A mon binôme Yousra,***

*Avec qui j'ai partagé des moments inoubliables et avec qui j'ai pu réaliser ce travail grâce à son aide sa compréhension et sa présence.*

*Que DIEU vous protège tous.*

***RAHALI Meroua***

*Je dédie ce modeste travail*

***A mes Chers parents,***

*Qui sont les meilleurs parents dans ce monde, c'est grâce à leur soutien moral, leur encouragement et leur confiance en moi que j'ai pu mener à terme ce projet. Tout en témoignant de ma profonde gratitude et de mon incontestable reconnaissance pour tous leurs sacrifices et tout l'amour dont ils m'entourent.*

***A mon cher frère et A mes chères sœurs,***

*Je leur dédie ce modeste travail en témoignage de mon grand amour et ma gratitude infinie.*

***A mes chers proches,***

*Qui étaient toujours à mes côtés et m'ont été d'un soutien précieux. Je leur souhaite à eux aussi plein de succès dans leur vie.*

***A toutes les personnes que j'aime et qui m'aiment,***

***A mon binôme Meroua,***

*Avec qui j'ai partagé des moments inoubliables et avec qui j'ai pu réaliser ce travail grâce à son aide sa compréhension et sa présence.*

*Que DIEU les protège tous.*

***ISMAIL Yousra***

## Résumé

SONELGAZ est l'opérateur historique dans le domaine de la fourniture des énergies électriques et gazière en Algérie. Les systèmes d'information dans le groupe SONELGAZ deviennent de plus en plus complexes et hétérogènes. Il convient par conséquent de les intégrer afin de les faire communiquer et de les faire coopérer, alors l'intégration et l'interopérabilité des processus métiers constituent un problème clé et des exigences de base dans le développement d'applications orientées services en interaction qui doivent être résolues de toute urgence.

Dans ce contexte, notre objectif est de résoudre le problème d'interaction (intégration) des différentes fonctionnalités communes entre les directions de distribution (DD) et les centraliser au niveau SADEG, et de résoudre le problème d'interopérabilité des systèmes d'information existant dans SONELGAZ, alors nous avons adopté la combinaison d'une architecture orientée service et l'architecture d'intégration de l'entrepôt de données pour constituer un système adaptatif d'aide à l'intégration du système d'informations hétérogène et distribué de notre organisme, il est basé sur la notion des services web qui permette d'intégrer des services préexistants et indépendants pour construire de nouveaux services avec de nouvelles fonctionnalités.

**Mot clés :** Architecture orienté service, intégration, web service, interopérabilité, SONELGAZ, SADEG, SGC.

## **Abstract**

SONELGAZ is the historical operator in the field of electricity and gas supply in Algeria. The information systems in the SONELGAZ group are becoming more and more complex and heterogeneous. Therefore, they need to be integrated in order to communicate and cooperate, so integration and interoperability of business processes are a key issue and basic requirements in the development of interacting service-oriented applications that need to be solved urgently.

In this context, our objective is to solve the problem of interaction (integration) of the different common functionalities between the distribution departments (DD) and centralize them at SADEG level, and to solve the problem of interoperability of the existing information systems in SONELGAZ, then we adopted the combination of a service oriented architecture and the data warehouse integration architecture to build an adaptive system to help integrate the heterogeneous and distributed information system of our organization, it is based on the notion of web services that allows to integrate pre-existing and independent services to build new services with new functionalities.

Keywords : Service oriented architecture, integration, web service, interoperability, SONELGAZ, SADEG, SGC.

## Liste des tables

Tableau 1: Comparaison REST vs SOAP .....	26
Tableau 2: Description de cas d'utilisation « Gestion des données de base» .....	46
Tableau 3: Description de cas d'utilisation «Gestion des jobs de synchronisation» .....	47
Tableau 4: Description de cas d'utilisation « Acte de gestion» .....	48
Tableau 5: Description de cas d'utilisation « Gestion des tarifs » .....	49
Tableau 6: Le modèle relationnel de serviceSGC. ....	77
Tableau 7: le le modèle relationnel de parametrage. ....	112

## Liste des figures

Figure I-1: Architecture de médiation [4].....	7
Figure I-2: Etapes de traitement du premier niveau de construction d'un entrepôt de données [9]. .....	9
Figure I-3: Terminologies de Cloud Computing [7]. .....	10
Figure I-4: Les principes de SOA. [24] .....	15
Figure I-5: Les propriétés d'un service. [15].....	16
Figure I-6: Les types de services. [17] .....	18
Figure I-7: L'architecture orientée service web. [25].....	20
Figure I-8: Structure de message SOAP. [25].....	21
Figure I-9: Structure d'un document WSDL. [25].....	23
Figure II-1: L'organigramme de groupe SONELGAZ.[10].....	32
Figure II-2: Organisations des SD. [11] .....	33
Figure II-3: Hiérarchie de la SD. [10] .....	34
Figure II-4: L'architecture fonctionnelle du SGC. [11] .....	37
Figure II-5: L'architecture technique du SGC. [11].....	38
Figure II-6: Diagramme de cas d'utilisation globale.....	45
Figure II-7: Diagramme de cas d'utilisation « Gestion des données de base». .....	45
Figure II-8: Diagramme de cas d'utilisation « Gestion des jobs de synchronisation». .....	47
Figure II-9: Diagramme de cas d'utilisation « Acte de gestion». .....	48
Figure II-10: Diagramme de cas d'utilisation « Gestion des tarifs ». .....	49
Figure II-11: Diagramme de séquence « gestion des communes ». .....	52
Figure II-12: Diagramme de séquence « gestion des tarifs».....	54
Figure II-13: Diagramme de séquence du cas d'utilisation « centralisation de synchronisation des jobs de synchronisation».....	56
Figure II-14: Diagramme de séquence du cas d'utilisation « Gestion des actes».....	58
Figure III-1: L'architecture globale du système. ....	62
Figure III-2:Le schéma d'intégrateur. ....	63
Figure III-3: Diagramme de séquence de service web d'ajout. ....	64
Figure III-4: Diagramme de séquence de service web de modification. ....	65
Figure III-5: Diagramme de séquence de service web de suppression. ....	66
Figure III-6: Diagramme de séquence de service web de la recherche. ....	67
Figure III-7: Diagramme de classe « Service web Activité économique ». ....	69

Figure III-8: Diagramme d'activité d'ajout. ....	70
Figure III-9: Diagramme d'activité de modification. ....	71
Figure III-10: Diagramme d'activité de suppression. ....	72
Figure III-11: Diagramme d'activité de la recherche. ....	73
Figure III-12: Diagramme de classe de serviceSGC. ....	74
Figure III-13: Diagramme de classe de paramétrage. ....	75
Figure III-14: Modèle logique de données de ServiceSGC. ....	77
Figure III-15: Modèle logique de données de paramétrage. ....	78
Figure III-16: Diagramme de composant du système. ....	79
Figure IV-1: Architecture MVC. [28] .....	83
Figure IV-2: Logo Spring Boot. [29] .....	84
Figure IV-3: Logo de Netbeans. [30] .....	84
Figure IV-4: Logo PostgrsSQL. [31].....	85
Figure IV-5: Logo« java EE». [32] .....	86
Figure IV-6: Logo de « Visual Paradigm ». [33] .....	86
Figure IV-7: Logo de « Glassfish ». [34] .....	86
Figure IV-8: interface authentification. ....	87
Figure IV-9: interface d'accueil d'administrateur fonctionnel niveau siège. ....	88
Figure IV-10: interface d'accueil d'administrateur fonctionnel niveau régionale. ....	88
Figure IV-11: Interface d'accueil. ....	89
Figure IV-12: Interface de recherche d'une activité économique. ....	90
Figure IV-13: interface de consulter les activités économiques de la base centralisé. ....	90
Figure IV-14: interface pour consulter les activités économiques par DD précise. ....	91
Figure IV-15: interface pour ajouter une activité économique. ....	91
Figure IV-16: interface de résultat de l'ajout. ....	92
Figure IV-17: Affichage de toutes les activités économiques. ....	92
Figure IV-18: Interface pour la modification d'une activité économique. ....	93
Figure IV-19: Interface pour supprimer d'une activité économique. ....	93
Figure A-1: schéma de SGC. ....	108



## Liste des abréviations

**SGC** : Système Gestion Clientèle de Sonelgaz.

**SOA** : Architecture Orienté Service

**RUP** : Rational Unified Process.

**UML** : Unified Modeling Language.

**CAAS** : Communication as a Service.

**SAAS** : Software as a Service.

**PAAS** : Platform as a Service.

**IAAS** : Infrastructure as a service.

**VPN**: Réseau privé virtuel.

**SI** : Système d'Information.

**XML** : Extensible Markup Language.

**URL** : Uniform Resource Locator.

**QoS** : Quality of Service.

**SLA** : Service Level Agreement.

**HTTP** : HyperText Transfer Protocol.

**REST** : Representational State Transfer.

**SOPA** : Simple Object Access Protocole.

**WSDL** : Web Services Description Language.

**UDDI** : Universal Description Discovery and Integration.

**URI** : Uniform Resource Identifier.

**HTML** : Hyper Text Markup Language.

**JSON** : JavaScript Object Notation.

**PDL** : Point de livraison.

**JPEG** : Joint Photographic Experts Group.

**RPC** : Remote Procedure Call.

**SMTP** : Simple Mail Transfer Protocol.

**FTP** : File Transfer Protocol.

**RMI** : Remote Method Invocation.

**CORBA** : Common Object Request Broker Architecture.

**JMS** : Java Message Service.

**API** : Application Programmable Interface.

**EGA** : Electricité et Gaz d'Algérie.

**SONELGAZ** : Société Nationale de l'Electricité et du GAZ

**SD** : Société de Distribution.

**SDO** : Société de Distribution Ouest.

**SDC** : Société de Distribution Centre.

**SDA** : Société de Distribution d'Alger.

**SDE** : Société de Distribution Est.

**ELIT** : Djazair Information Technology.

**DSID** : Direction SI Activités Distribution.

**BT** : Basse Tension.

**HTA/B** : Haute Tension class A/B.

**BP** : Basse Pression.

**HP** : Haute Pression.

**FMS** : Facturation Multi-sites.

**ADM** : Administratifs.

**DMD** : Débit Maximal Demandé.

**PMD** : Puissance Maximale Demandée.

**TVA** : Taxe sur Valeur Ajoutée.

**CRMS** : Customer Relationship Management System.

**DD** : Direction de Distribution.

**DCM** : Directions Commerciales et Marketing.

**SGBD** : Système Gestion Base de Données.

**BDD** : Base De Données.

**UP** : Unified Process.

**XP** : Extrem Programing.

**MLD** : Modèle Logique de Données.

**MPD** : Modèle Physique de Données.

**MCD** : Modèle Conceptuel de Données.

**MVC** : Modèle, Vue et Contrôleur.

**IDE** : Environnement de Développement Intégré.

**CDDL** : Common Développement and Distribution License.

**SGBDRO** : Système de Gestion de Base de Données Relationnelle et Objet.

**ODBC** : Open Data base Connectivity.

## Table des matières

Introduction générale .....	2
<i>Chapitre I : Synthèse sur les différentes architectures de centralisation.....</i>	<i>4</i>
Introduction .....	5
I.1. Centralisation des systèmes d'information.....	5
I.2. L'interopérabilité et l'intégration des systèmes d'information.....	5
I.3. Les approches existantes de la centralisation .....	6
I.3.1. Intégration.....	6
I.3.2. Cloud Computing : .....	10
I.3.3. Architecture Orientée Services (SOA) .....	13
Conclusion.....	28
<i>Chapitre II : Etude de l'existant et analyse des besoins .....</i>	<i>29</i>
II.1. Étude de l'existant .....	30
II.1.1. Présentation de la société SONELGAZ .....	30
II.1.2. Diagnostic.....	39
II.1.3. Présentation du sujet .....	39
II.2. Analyse des besoins .....	41
II.2.1. Le langage UML.....	41
II.2.2. Identification des acteurs.....	43
II.2.3. Spécification des besoins fonctionnels.....	43
II.2.4. Diagramme de cas d'utilisation .....	44
II.2.5. Diagramme de séquences: .....	50
II.2.6. Spécification des besoins non fonctionnels .....	58
Conclusion.....	59
<i>Chapitre III : Conception.....</i>	<i>60</i>
Introduction .....	61
III.1. L'architecture proposée .....	61
III.2. Conception partie service web .....	63
III.2.1. Diagramme de séquence pour les services web.....	63
III.2.2. Diagramme de classe des services web.....	68
III.3. Conception détaillé.....	69
III.3.1. Diagramme d'activité .....	69
III.3.2. Diagramme de classe .....	73
III.3.3. Modèles Logiques de Données.....	76

III.3.4. Diagramme de composant.....	79
Conclusion.....	80
<i>Chapitre IV : Implémentation</i> .....	81
Introduction .....	82
IV.1. Architecture MVC.....	82
IV.2. Les environnements et les outils utilisés .....	83
IV.2.1.Framework de développement Spring Boot.....	83
IV.2.2.Outils de développement « NetBeans » .....	84
IV.2.3.Serveur de base de données Postgresql.....	85
IV.2.4. Plateforme de développement “Java EE” .....	85
IV.2.5. Outils de modélisation .....	86
IV.2.6. Serveur de déploiement « Glassfish » .....	86
IV.3. Description de l’interface de l’application .....	87
IV.3.1.Interface d’authentification .....	87
IV.3.2.Interface d’accueil .....	89
IV.3.3.Interface de gestion activité économique.....	89
<i>Conclusion générale et perspective</i> .....	95
Références Bibliographiques.....	98
<i>Annexe</i> .....	100

# *Introduction générale*

## Introduction générale

De nos jours, l'interopérabilité et l'intégration des applications sont devenues de plus en plus importantes dans les organisations, tout en rentabilisant les investissements consentis dans des systèmes généralement assez coûteux.

Dans ce contexte les organisations doivent à la fois connecter les nombreuses applications hétérogènes existantes, exploiter des données issues de systèmes d'information différents et définir de nouveaux processus tout en garantissant à terme la cohérence du système.

SONELGAZ est le leader national chargé de la production, du transport et de la distribution de l'électricité et du gaz en Algérie. La distribution est l'activité la plus rentable du groupe vu qu'elle réalise 44 % de son chiffre d'affaire. Cette activité est confiée à quatre filiales. Il s'agit des sociétés de distribution (SD). L'activité commerciale du SONELGAZ est distribuée sur le territoire national, soit 53 directions de distribution, et chacune avec sa propre base de donnée. Les SD disposaient d'un système de gestion de la clientèle SGC qui est déployé pour chaque direction, ce que rend l'administration fonctionnelle du système une tâche difficile.

Les architectures orientées services (SOA) constituent une réponse aux problématiques de SONELGAZ, en termes de description, de réutilisabilité, d'interopérabilité et de réduction de couplage entre les différents systèmes qui implémentent des fonctionnalités, et issues en général d'applications complexes, en cachant les détails d'implantation.

Dans notre travail, nous nous intéressons au cas des SOA basées sur les services Web. Un des apports principaux de ces architectures est la possibilité d'intégrer des services préexistants et indépendants pour construire de nouveaux services avec de nouvelles fonctionnalités.

Les services Web semblent donc être bien adaptés aux problèmes sous tendus par l'intégration des processus métiers. Ils permettent en effet une intégration centrée sur les services, où les fonctions automatiques sont mises à la disposition de tous les utilisateurs éventuels, quelles que soient la technologie et les spécifications d'interfaces qu'ils utilisent.

Pour bien présenter notre travail, nous avons structuré notre mémoire autour des axes suivants :

Chapitre 1 : ce chapitre décrit le rôle des systèmes centralisés et le but de la centralisation, on cite aussi les différentes approches centralisées et chacune on donne son explication, ainsi, nous avons fait une synthèse pour choisir l'approche adaptée.

Chapitre 2 : Le 2eme chapitre sera consacré à la présentation de la structure et l'organisation du groupe SONELGAZ, aux événements les plus marquants de son histoire, ainsi qu'à ses filiales. Il consacré aussi à la présentation de la spécification et l'analyse des besoins et nous choisirons la méthode RUP pour l'implémentation de processus unifié.

Chapitre 3 : dans ce chapitre, on présentera l'approche de notre solution de manière générale et on détaille notre solution munie de certains diagrammes UML.

Chapitre 4 : ce chapitre est dédié à la présentation des outils utilisés dans la réalisation de notre système ainsi que le résultat de son implémentation.



***Chapitre I : Synthèse sur les différentes  
architectures de centralisation***

## **Introduction**

Faut-il centraliser ou décentraliser les systèmes d'information pour créer un plus grand soutien aux différents processus métier? Différentes approches de centralisation et de décentralisation se sont produites au cours du siècle dernier, et il n'y a pas de réponse simple à la question. Ce chapitre donne au premier lieu un aperçu sur le rôle des systèmes de centralisation et sur quel but on centralise, on cite aussi les différentes approches centralisées et chacune on donne son explication. Au deuxième lieu, nous avons fait une synthèse pour choisir l'approche adaptée.

### **I.1. Centralisation des systèmes d'information**

Un système centralisé c'est un système où tout le monde dépend d'une même autorité, un serveur a priori dans le cas informatique. Bien qu'un système centralisé soit beaucoup plus simple à faire sur le plan technique (facile de trouver des gens ou des informations quand ils sont tous au même endroit). [1]

Cette évolution des systèmes d'information sont dues à différentes causes [1] :

- En effet, les entreprises et/ou organisations développent de nouveaux logiciels et applications tout en gardant les anciens pour des raisons non seulement de sécurité mais aussi de logistique et compréhension.
- D'autre part, les entreprises doivent, de plus en plus, ouvrir leurs systèmes d'information à leurs fournisseurs, partenaires ou clients qui se trouvent souvent dans un autre pays du monde ou qui ne possèdent pas les mêmes outils et systèmes d'information.
- Enfin, les données sont stockées dans de multiples formats, de protocoles et technologies de bases de données.

### **I.2. L'interopérabilité et l'intégration des systèmes d'information**

L'interopérabilité des systèmes d'information est un domaine d'intérêt en croissance, à cause du besoin continu d'intégration de nouveaux systèmes et des systèmes en évolution, en particulier dans le contexte des réseaux d'entreprises collaboratives. Un de ses enjeux est de faire collaborer et interagir des applications distribuées, autonomes et hétérogènes, et de permettre le partage d'information et de connaissances.[22]

### **I.3. Les approches existantes de la centralisation**

Parmi les différentes approches qui existent dans la centralisation nous avons choisi les trois approches suivantes.

#### **I.3.1. Intégration**

L'intégration de données est un ensemble de processus qui peuvent fusionner plusieurs sources de données via une interface unifiée. Après avoir éliminé tous les conflits entre les données et les avoir présentées de manière cohérente, dans les informations par une vue globale qui traite les interrogations des requêtes. [3]

Le système d'intégration peut être réalisé via des vues matérialisées (comme dans l'entrepôt de données) ou via des vues virtuelles (dans le système médiation). [3]

#### **A. Les approches d'intégration**

Nous présentons ci-dessous les deux approches d'intégration existantes : Approche virtuelle (l'approche médiateur) et Approche matérialisée (l'approche entrepôt de donnée).

##### **a. L'approche médiateur**

L'approche médiateur consiste à développer une application qui agit comme une interface entre les sources de données locale et l'application utilisateur. L'objectif est de donner l'impression d'interroger un système centralisé et homogène alors que les sources interrogées sont réparties, autonomes et hétérogènes.[4]

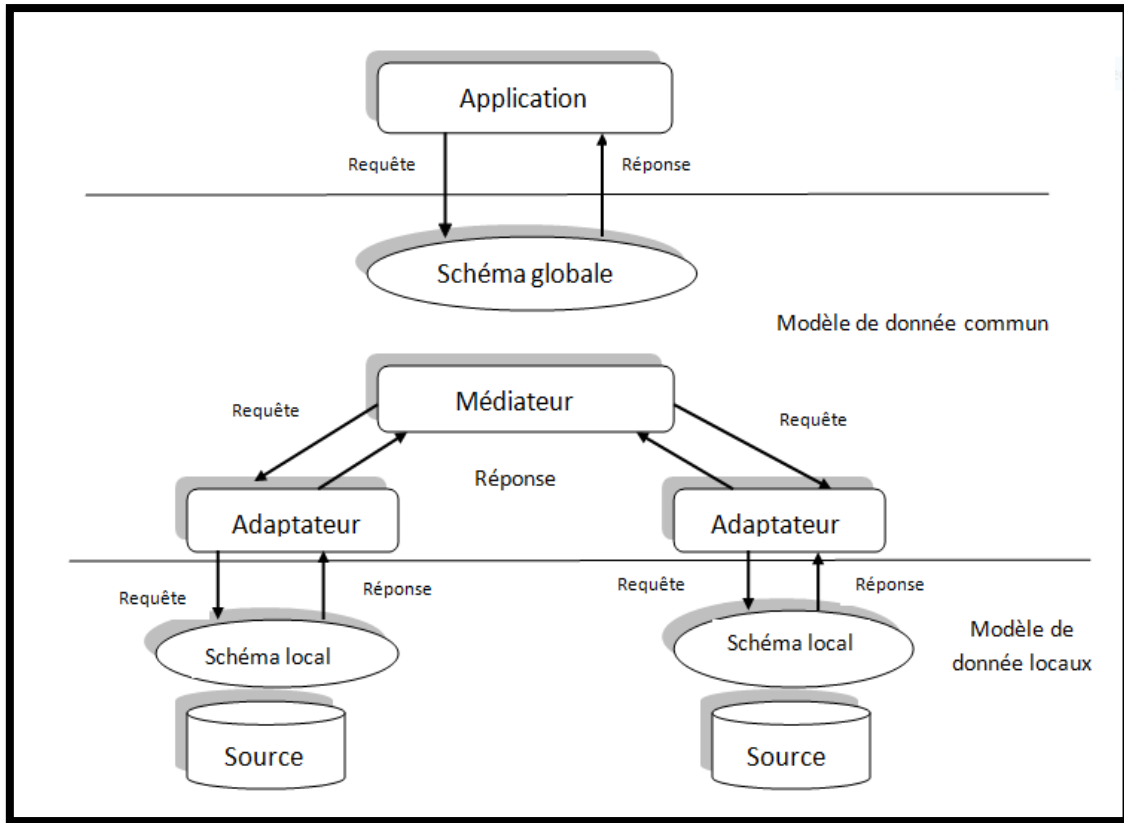


Figure I-1: Architecture de médiation [4].

Comme nous pouvons le voir sur la figure I-1, les systèmes de bases de données hétérogènes actuels adoptent une architecture distribuée qui consiste en plusieurs composants spécialisés. Un médiateur expose un schéma médiateur ou (schéma global) qui est une représentation intégrée des sources de données, son rôle est central. Les applications accèdent aux sources de données sous-jacentes à travers les médiateurs. Les médiateurs traitent les requêtes sur la représentation intégrée. Les composants essentiels d'un système de médiation sont [2] :

➤ **Adaptateur (Wrapper):**

Un adaptateur est donc un composant qui :

- Réalise la transformation entre le modèle de données dans lequel sont représentées les données de la source et le modèle choisi au niveau médiateur.
- Effectue la transformation entre les expressions de requêtes du niveau global en expressions compréhensibles par les sources. Inversement, il traduit les réponses des sources au format du niveau global.

- Assure les traitements spécifiques non disponibles au niveau local et nécessaire pour le niveau global (augmentation du pouvoir d'expression et de traitement de la source locale).

➤ **Méiateur (Mediator):**

Le rôle d'un médiateur est de collecter, nettoyer et fusionner les données attendues par le système de médiation et produites par les différents adaptateurs. C'est un module logiciel recevant directement la requête d'un usager et devant la traiter. Celui-ci doit localiser l'information nécessaire pour répondre à la requête, résoudre les conflits schématiques (lié au schéma c'est de donner même terme par des terminologies différentes) et sémantiques (exprime le même concept mais avec des significations différentes), interroger les différentes sources et intégrer les résultats partiels dans une réponse homogène et cohérente.[4]

**b. L'approche entrepôt de données**

Un entrepôt de données est une base de données regroupant une partie ou l'ensemble des données fonctionnelles d'une entreprise. son but est de fournir un ensemble de données servant de référence unique, utilisée pour la prise de décisions dans l'entreprise par le biais de statistiques et de rapports réalisés via des outils de reporting.[6]

➤ **Les étapes d'intégration**

Cette architecture consiste à centraliser physiquement, dans un entrepôt de données, l'ensemble de données consolidées à partir de diverses sources (catalogues électroniques, bases de données relationnelles, Web, etc.). [9]

Le processus de construction d'un système d'intégration matérialisé se compose en quatre étapes principales [9] :

- l'extraction des données des sources de données opérationnelles.
- la transformation des données aux niveaux structurel et sémantique.
- l'intégration des données.
- le stockage des données intégrées dans le système cible.

La figure I-2 résume l'enchaînement de ces étapes de traitement.

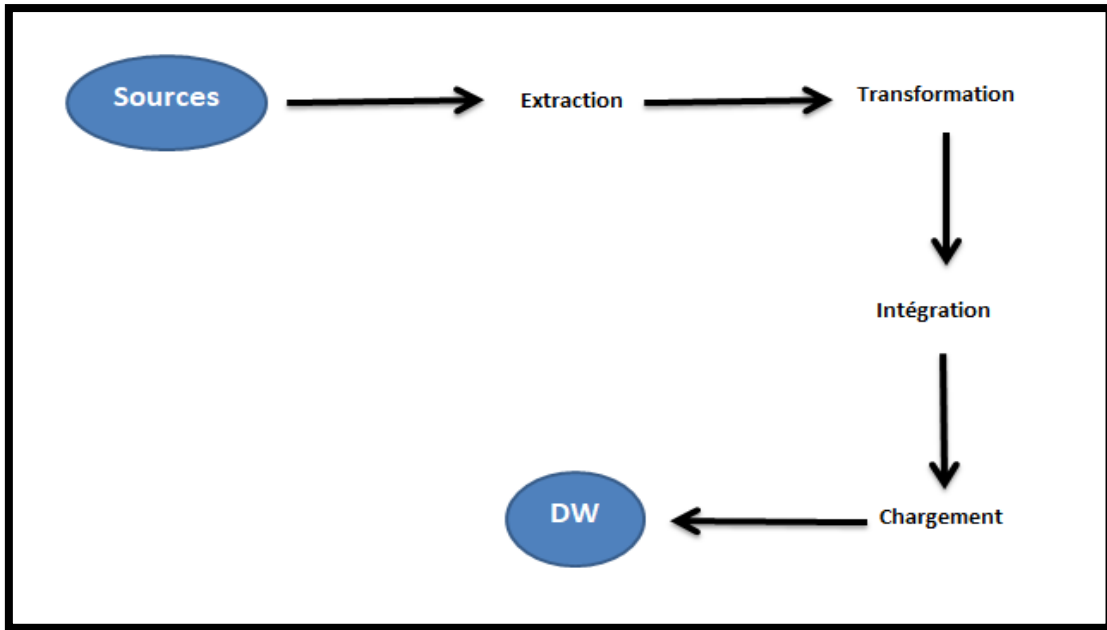


Figure I-2: Etapes de traitement du premier niveau de construction d'un entrepôt de données [9].

### B. Les avantages d'intégration

L'approche d'intégration est populaire et couramment utilisée car elle offre des avantages intéressants pour les utilisateurs des systèmes distribués [9]:

- L'intégration de la base de données consiste à combiner des informations provenant de diverses sources pour distribuer une version propre et consolidée dans toute l'entreprise.
- Pour rendre les données accessibles à plusieurs parties prenantes et applications client sans qu'il soit nécessaire de les dupliquer ou de les déplacer.
- Une synchronisation des données entre les applications.
- Une vue d'ensemble des clients.
- Une disponibilité des données pour les dirigeants et les métiers.
- Des analyses, des prévisions et des prises de décisions s'appuyant sur des données complètes et de qualité.

### C. Les limites d'intégration

Comme chaque approche, l'intégration comporte plusieurs limites dont [9] :

- La quantité massive de données peut être telle qu'elle est difficilement exploitable.
- L'intégration des données contient des informations qui ne sont pas toujours à jour. Cela est dû au fonctionnement des entrepôts de données, qui extraient régulièrement des informations d'autres bases de données. Si les données de ces bases de données

changent entre les extractions, les requêtes adressées à l'entrepôt de données n'obtiendront pas la vue la plus récente et la plus précise.

- lorsqu'on fait un schéma d'intégration, il faut que la base de données doit être faite, on ne peut rien ajouter après, à chaque mise à jour si on ajoute une nouvelle base de données ou on est fait faux schéma conceptuel, alors le schéma d'intégration doit être refait.

### I.3.2. Cloud Computing :

#### A. Définition

Le cloud computing est une technologie qui permet d'accéder aux données ou à l'infrastructure via une simple connexion Internet. Ces données sont gérées par le serveur distant et ne nécessitent ni installation ni configuration. [7].

#### B. Les différents types du Cloud Computing

On va décrire les services du cloud computing les plus importants sur la figure I-3 suivante [7] :

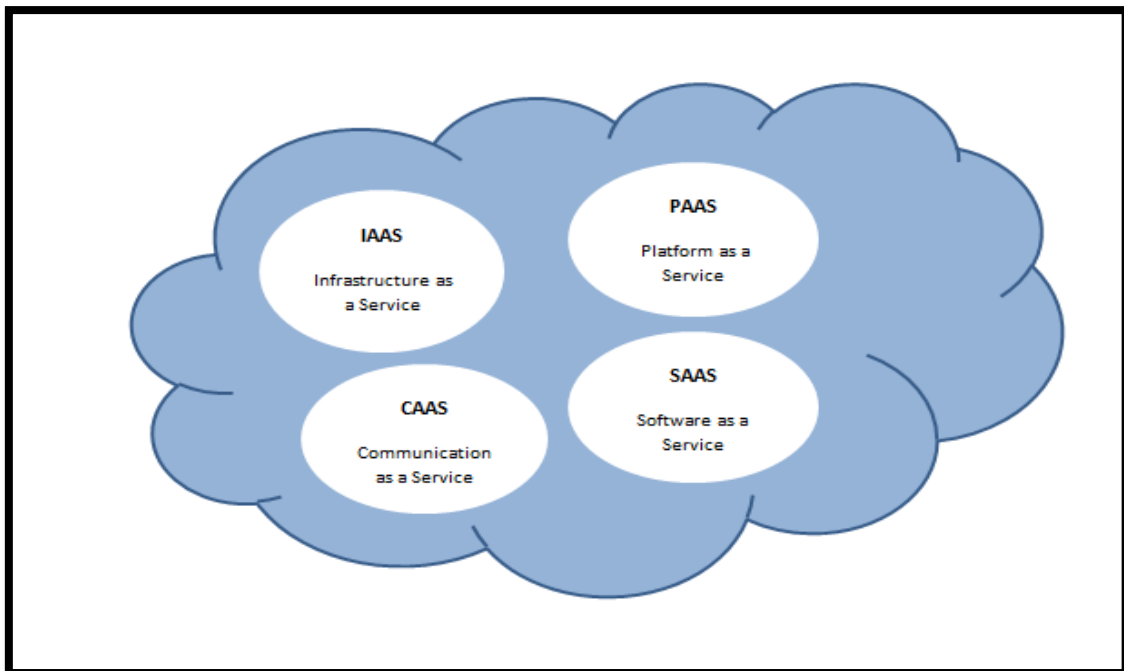


Figure I-3: Terminologies de Cloud Computing [7].

### **a) CAAS**

Il existe plusieurs significations de CAAS, mais la plupart des gens pensent que CAAS est très cohérent avec la communication en tant que service. CAAS comprend les moyens de fournir une communication en tant que service via Internet. En fait, ce n'est pas du tout nouveau, mais simplement utiliser des noms différents dans les services existants (VOIP, Web mail, etc.).

### **b) SAAS**

Le SAAS qui signifie « Software as a Service » est un modèle de déploiement d'application dans lequel un fournisseur loue une application clé en main à ses clients en tant que service à la demande au lieu de leur facturer des licences. De cette façon, l'utilisateur final n'a plus besoin d'installer tous les logiciels existants sur sa machine de travail.

### **c) PAAS**

« Plateforme as a Service », offre aux développeurs les outils nécessaires pour créer et héberger des applications web. La stratégie PaaS est conçue pour permettre aux utilisateurs d'accéder aux composants dont ils ont besoin afin de développer et d'implémenter rapidement des applications web ou mobiles sur Internet.

### **d) IAAS**

C'est le système informatique dans son ensemble qui est externalisé et hébergé par un prestataire. Celui-ci comprend la partie software (logiciels et applications) mais aussi la partie hardware (serveurs et réseaux). Les entreprises qui ne disposent pas de moyens suffisant pour acquérir leur propre matériel informatique peuvent tirer parti de cette technologie. La maintenance et les opérations de mises à jour sont intégralement prises en charge par le prestataire.

## **C. Les modèles de déploiement du Cloud Computing**

Les services de cloud computing se distinguent en trois catégories [7]:

- Le cloud privé (ou interne) : comme son nom l'indique, il est dédié à un seul utilisateur. L'avantage du cloud privé est son important niveau de sécurité, renforcé par une connexion VPN. Le Cloud privé est administré par l'entreprise elle-même ou un prestataire de services.



- Le cloud public (ou externe) : les ressources informatiques de l'entreprise sont stockées sur un serveur mutualisé, autrement dit partagé entre plusieurs clients, et accessibles par internet.
- Le cloud hybride (interne et externe) : l'entreprise utilise à la fois le cloud privé et le cloud public pour mettre en œuvre certaines activités. Par exemple, le cloud public est utilisé par les collaborateurs pour les tâches opérationnelles, tandis que le cloud privé sert à héberger le site web e-commerce de l'entreprise ou ses données financières, pour réduire le risque de piratage.

#### **D. Avantages du Cloud Computing**

Les avantages du cloud computing sont [7]:

- Souplesse d'évolution : il n'y a pas de logiciel à installer et l'accès se fait avec un simple navigateur web.
- Simplicité : l'entreprise cliente n'a plus besoin de développements coûteux et déplace la responsabilité du fonctionnement du service sur le fournisseur.
- Liberté de changement de service : le cloud computing étant généralement facturé à la demande ou par abonnement mensuel, il est très facile pour une entreprise d'arrêter le service si elle n'en a plus besoin ou si elle souhaite aller chez un concurrent.
- Coût : la force du cloud computing réside dans la possibilité de proposer le même service à un grand nombre d'utilisateurs, finalement, le coût de cloud computing sera donc très raisonnable.

#### **E. Les limites du Cloud Computing**

Cette technologie présente aussi plusieurs limites et sont présentés comme suit : [7]

- Confidentialité et sécurité des données : les données sont hébergées en dehors de l'entreprise. Le fournisseur de services héberge les données utilisateur de l'entreprise, ce qui peut présenter un risque potentiel pour l'entreprise d'abuser ou de voler ses données.
- Dépendance: si l'entreprise souhaite des fonctionnalités très spécifiques, il peut être difficile de convaincre le fournisseur de proposer ces fonctionnalités. Et en général, s'il y a un problème, l'entreprise est tributaire du service client d'un fournisseur. Il s'agit donc de choisir un fournisseur en qui l'on a confiance.
- Besoin d'un raccordement constant d'internet.

- Peut-être plus cher pour certain cas d'utilisation.
- On parle de cloud computing lorsqu'il est possible d'accéder à des données ou à des programmes depuis internet. Mais l'entreprise avec laquelle nous traitons notre projet a déjà un data center comme un espace de données virtuelle pour héberger leurs applications et données critiques.

Dans la partie qui suit, on présente une approche qui a essayé d'engendrer les limites des différentes approches présentées qui est le SOA.

### **I.3.3. Architecture Orientée Services (SOA)**

À cause de l'utilisation de la technologie hétérogène, le système d'information (SI) est devenu lourd et rigide, cela a conduit à un manque d'interopérabilité. Les experts de ce domaine ont découvert des nouvelles méthodes pour résoudre ces problèmes. Parmi les technologies qui sont apparues comme solution, c'était la notion «Architecture Orientée Services ». [13]

#### **A. Définition**

Il n'y a pas de définition exacte de l'architecture orientée services. En fait, plusieurs définitions ont été proposées, mais toutes conviennent que la SOA est un paradigme visant à résoudre les problèmes d'hétérogénéité et d'interopérabilité du logiciel qui constitue un système d'information.

*« L'architecture orientée service constitue un style d'architecture basée sur le principe de séparation de l'activité métier en une série de services. »*[12]

*«Les services dans une SOA sont les modules d'une fonctionnalité de l'application avec des interfaces exposées et qui sont invoquées par messages.»*[12]

*« Une façon de concevoir et d'implémenter les applications de l'entreprise qui possèdent un couplage faible, sont à grandes mailles et sont des services réutilisables, accessibles par des interfaces bien définies et indépendantes de toute plateforme. »*[13]

L'architecture SOA peut être utilisée dans un cadre interne à l'entreprise, ainsi les services peuvent être classés par domaine fonctionnel (achat, finance, marketing, etc.), mais surtout SOA présente un moyen efficace pour faciliter la communication avec d'autres entreprises [13].

## **B. Pourquoi SOA ?**

Au début les départements informatiques ont tenu un point de vue orienté application du monde. Ils ont alloué la plupart de leurs budgets à l'achat, le déploiement et la maintenance des applications. A l'époque, un grand pourcentage de budget a été alloué aux projets d'intégration pour fournir un soutien plus large et plus transparent pour les processus d'affaires. En effet, ces projets n'ont souvent pas atteint leurs promesses en termes de la rigidité de l'intégration câblés. Tout changement à une seule application rend le retour en arrière de l'infrastructure très coûteux. [20]

Après que les techniques conventionnelles de programmation ont été remplacées par le développement de modèle à objet, la programmation par composant est apparue et a permis un développement des applications basé sur cette technique de programmation, un nouveau concept architectural est apparu : l'architecture orienté service ou SOA (Service Oriented Architecture). [20]

En quelques années, SOA est devenue un thème majeur pour les systèmes d'informations d'entreprise. Plus qu'une nouvelle technologie ou méthode, c'est la convergence de plusieurs approches existantes, et l'émergence d'une forte adhésion des directions informatique et métier à un même objectif : une meilleure agilité des systèmes face aux transformations nécessaires. [20]

Cette architecture est apparue pour : [20]

- Gérer l'intégration entre des applications dans le SI.
- Gérer l'interopérabilité entre les applications.
- Construire un SI plus flexible.
- Réduire les coûts.
- Réagir rapidement à tous les changements.
- Réduire les doublons entre applications.

### **1) Principe du SOA**

Les principaux concepts intervenant dans l'architecture orientée service [24] :

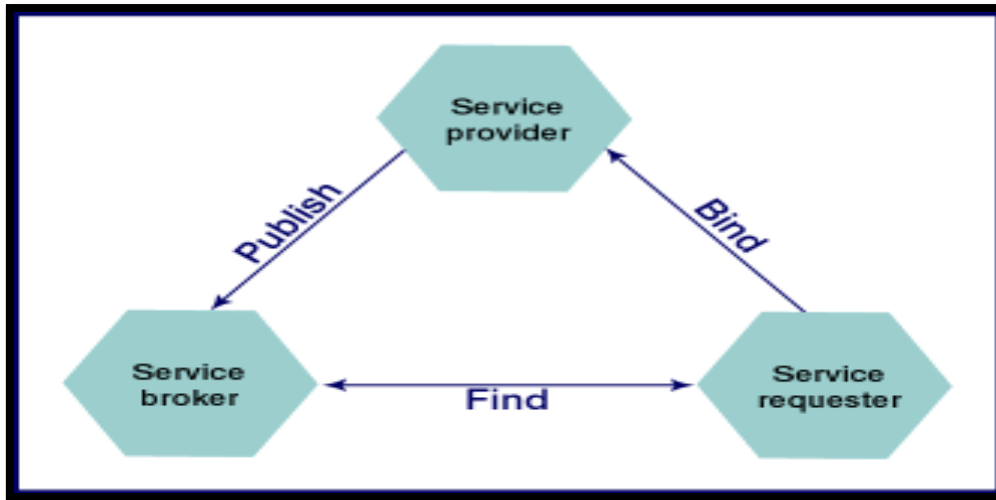


Figure I-4: Les principes de SOA. [24]

- a) **Provider (Fournisseur)** : le fournisseur de service met en application le service web et le rend disponible sur internet.
- b) **Requestor (Demandeur)**: c'est n'importe quel consommateur du service web. Le demandeur utilise un service web existant en ouvrant une connexion réseau et en envoyant une demande en XML (REST, XML-RPC, SOAP).
- c) **Annuaire service registry** : le registre de service est un annuaire de services. Le registre fournit un endroit central où les programmeurs peuvent publier de nouveaux services ou en trouver. Les interactions entre ces trois acteurs suivent plusieurs étapes :
  - **La publication du service** : le fournisseur diffuse les descriptions de ses services web dans l'annuaire.
  - **La recherche du service** : le client cherche un service particulier, il s'adresse à un annuaire qui va lui fournir les descriptions et les URL des services demandés afin de lui permettre de les invoquer.
  - **L'invocation du service** : une fois que le client récupère l'URL et la description du service, il les utilise pour l'invoquer auprès du fournisseur de services.

## C. Les services

### a. Définition

SOA propose un modèle d'architecture informatique basé sur l'émergence de la couche service. Ces services offrent une vision logique des processus et des données qui existent déjà

ou seront développés, et chaque service encapsule ces processus et données, masquant ainsi l'hétérogénéité du système d'information. Plusieurs définitions ont été proposées, parmi elles:

«Un service est un comportement défini contractuellement qui peut être implémenté et fourni par n'importe quel composant pour une utilisation par n'importe quel composant, basé uniquement sur le contrat».[14]

Un service peut se réduire à une simple tâche élémentaire comme il peut englober tout un processus métier. Un service doit disposer d'une interface qui permet de dialoguer avec lui. Par analogie, cette interface joue le rôle d'un contrat entre le fournisseur et le consommateur de service. Le contrat indique notamment la liste des opérations disponibles, la manière de les employer (entrées/sorties attendues), ainsi que la ressource physique (le serveur) sur laquelle se trouve. [14]

### b. Propriétés d'un service

Un service doit posséder les caractéristiques suivantes (voir la figure I-5) [15]:

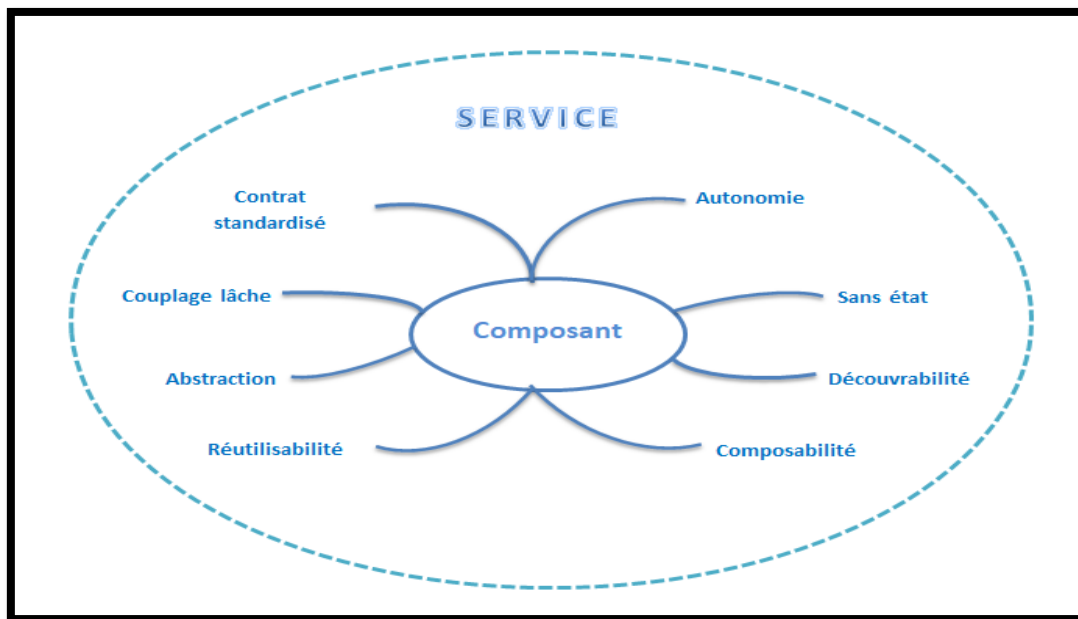


Figure I-5: Les propriétés d'un service. [15]

- **L'autonomie**

L'autonomie de service représente sa capacité à se gérer. Les services sont considérés comme autonomes s'ils ont la liberté et le contrôle de prendre leurs propres décisions sans approbation ou intervention externe. L'autonomie de service peut garantir une meilleure

fiabilité, performance et prévisibilité pendant l'exécution du service, en particulier en cas de réutilisation ou de combinaison de services.

- **Granularité**

Cette caractéristique décrit à quel point une architecture orientée service est découpée en services et sous-services. Plus la granularité est fine, plus il y a de services et sous-services.

- **Couplage lâche**

La propriété de couplage faible implique qu'un service n'appelle pas directement un autre service. En effet, les interactions sont gérées par une fonction d'orchestration. Alors le principe de couplage lâche est instauré par le biais du contrat de service, qui est l'élément de base autour duquel tourne la majorité des aspects conceptuels liés au couplage.

- **Interopérabilité**

L'adoption du concept de service permet aux différentes solutions du système informatique d'échanger des données et des fonctionnalités entre elles, même s'ils sont développés en différents langages.

- **Réutilisabilité**

Les services sont conçus de façon à ce qu'ils puissent être réutilisés ultérieurement. Le but est de minimiser la redondance et rendre le service réutilisable par les différentes applications du système d'information.

- **Composabilité**

Les services peuvent appeler ou utiliser d'autres services, le principe étant de décomposer les processus métiers en un certain nombre d'unités d'activité sous forme de services. La composition de ces éléments conduit à l'exécution de diverses tâches du processus métier.

- **Découvrabilité**

Pour appeler un service, vous devez savoir que le service existe, il existe donc un domaine public où vous pouvez trouver tous les services et les descriptions de chaque service.

- **Contrat standardisé**

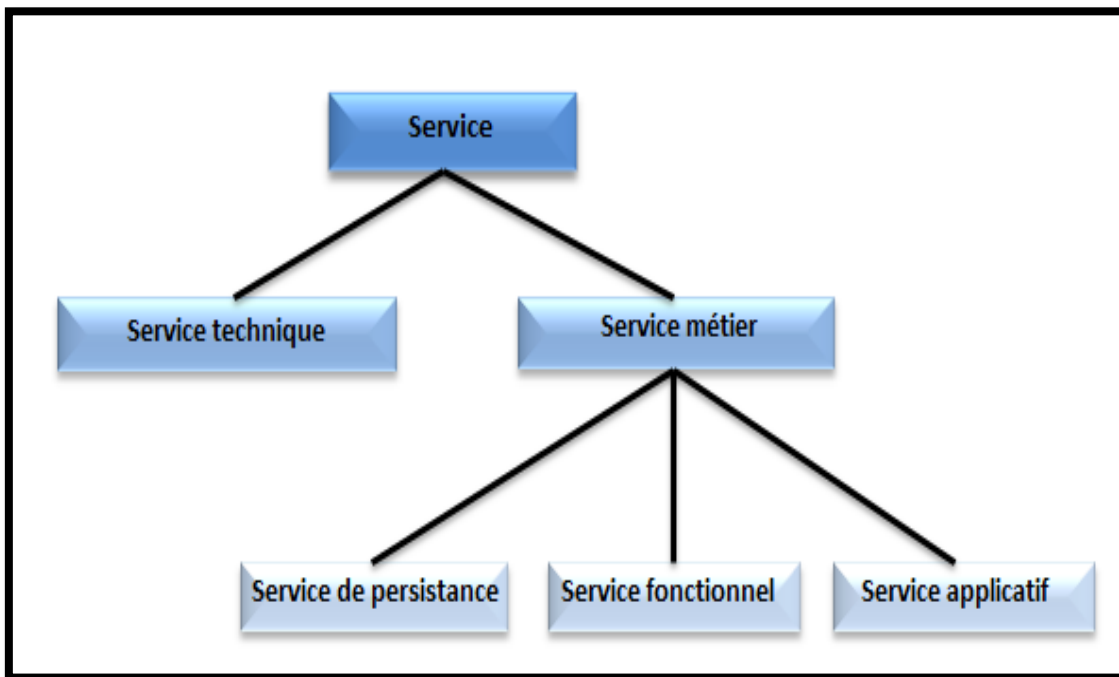
L'interaction entre un fournisseur et un consommateur de services se fait à travers une interface selon un contrat d'utilisation, évidemment on échange des données de nature métier.

- **Abstraction**

Le principe d'abstraction consiste à ne révéler d'un service que l'information nécessaire à son utilisation. Mis à part sa description dans le contrat de service, la logique ou l'implémentation du service est cachée.

**c. Les types de service**

Il existe deux principaux groupes de services [17] :



**Figure I-6: Les types de services. [17]**

➤ **Les services métiers**

Il fournit un type de service spécifique et correspond à la gamme de fonctions que nous voulons exposer aux consommateurs, quel que soit le choix de l'architecture de l'application.

Les services métiers sont divisés en trois catégories :

- **Service de persistance** : c'est le responsable de la création, de la recherche, de la mise à jour et de la suppression des informations de la base de données d'informations du système SI.
  - **Service fonctionnel** : sont utilisés pour encapsuler tout ou partie des règles et processus métier. Ils s'appuient généralement sur des services persistants pour accéder aux informations métiers requises. Le service fonctionnel est caractérisé par une granularité moyenne à forte et une réutilisabilité moyenne à forte.
  - **Service applicatif** : sert à coordonner les appels aux services fonctionnels. Sa granularité est forte et sa réutilisabilité est nulle.
- **Les services techniques**

Ils donnent l'accès à une ressource technique donnée (Messagerie, imprimante, Base de données relationnelle...etc.). Les services techniques sont génériques, c'est-à-dire lié à une catégorie de ressources. Ils sont caractérisés par une granularité moyenne à forte et une réutilisabilité forte.

#### **D. Les services web**

SOA est un concept architectural. Les services web sont une réalisation de la SOA, qui exploite XML et les protocoles internet courants (tels que HTTP) pour déployer des services pouvant être découverts à granularité grossière.

##### **a. Définition**

Il existe différentes définitions pour les services web. Certaines sont plus générales que d'autres. D'un point de vue technique La définition communément admise c'est la définition de W3C.

Le W3C définit le service web comme 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. [18]



«Un service web est un mécanisme qui tend à donner plus d'interactions pour permettre à deux entités hétérogènes (entreprises, clients, applications, etc. ...) de dialoguer au travers du réseau internet. Les logiciels écrits dans divers langages de programmation (C#, Visual Basic, Java, etc...), sur diverses plateformes (Linux, Windows, etc. ...) et avec diverses architectures peuvent employer des services web pour échanger des données à travers des réseaux informatiques. Chaque service web doit pouvoir être découvert et invoqué dynamiquement par les applications». [19]

## b. Type des services web

Sur le plan technique, les services web peuvent être mis en œuvre de différentes manières. Les deux types de services web abordés peuvent être distingués en tant que BIG web services et RESTful web services. [21]

### 1) BIG Web service

Les gros services web (BIG web service) sont basés sur le standard SOAP et contiennent souvent un WSDL pour décrire l'interface proposée par le service web. Les détails du contrat peuvent inclure des messages, des opérations, des liaisons et l'emplacement du service web [25] :

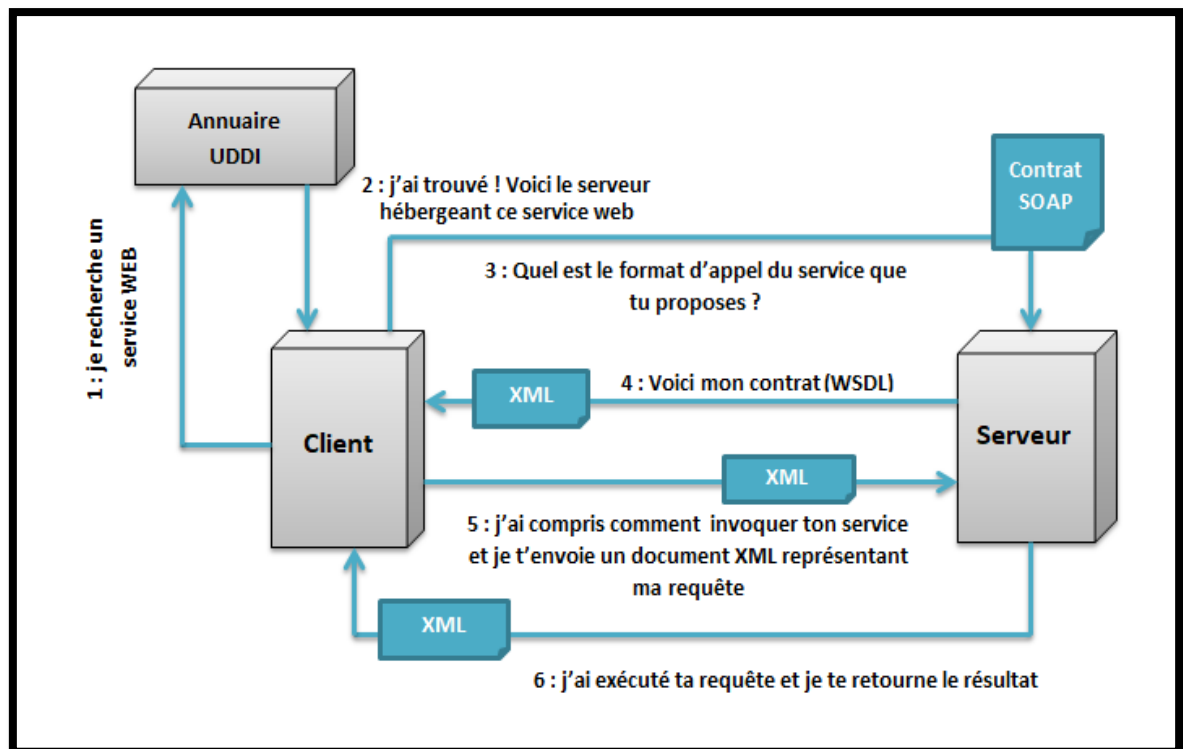


Figure I-7: L'architecture orientée service web. [25]

**i. Transport: SOAP (Simple Object Protocol communication)**

SOAP c'est un protocole de communication, proposé par W3C, Il est produit par Microsoft et IBM. Le but principal de SOAP est de transférer les messages qui sont écrits sur le format XML sur des environnements indépendants. SOAP est un protocole de type requête/réponse.

SOAP a ses propres avantages, car il définit sa propre sécurité appelée WS Security et est indépendant du langage et de la plate-forme, ce qui permet aux services web SOAP d'être écrits dans n'importe quel langage de programmation et exécutés sur n'importe quelle plate-forme.

SOAP a aussi ses propres inconvénients, comme sa lenteur, car il utilise un format XML qui doit être analysé pour être lu. Il définit de nombreuses normes à respecter lors du développement des applications SOAP. Donc, il est lent et consomme plus de bande passante et de ressources. SOAP utilise WSDL et n'a aucun autre mécanisme pour découvrir le service.

**La structure des messages SOAP**

Un message SOAP est un document XML, composé de deux parties, l'en-tête de protocole de transport et l'enveloppe SOAP.



**Figure I-8: Structure de message SOAP. [25]**

- **l'en-tête**
- **l'enveloppe**

Un message SOAP est contenu dans une enveloppe, La structure d'une enveloppe SOAP se compose de plusieurs parties :

**L'élément Header (entête):** contient des informations sur le traitement du message. Pour des services web simples, cette partie peut être vide ou absente mais pour des services plus complexes elle peut contenir des informations concernant les transactions, la sécurité, le routage, etc...

**L'élément Body :** contient les informations de la requête ou de la réponse, le body de la requête identifier l'objet distant, le nom de la méthode à exécuter et les éventuels paramètres ou des rapports d'erreur 'Fault'.

**L'élément Fault :** est optionnel et fournit des informations sur d'éventuelles erreurs (le type d'erreur, une description de l'erreur et l'adresse du serveur SOAP).

## ii. **Description : WSDL (Web Services Description Language)**

WSDL est utilisé pour fournir une description d'un service web afin de permettre son utilisation. Proposé par W3C, Il est produit par Microsoft et IBM.

Afin de permettre aux clients d'utiliser les services web, ce dernier doit les décrire en détail avant d'interagir avec le service. WSDL fournit cette description dans un document XML. WSDL joue un rôle important dans l'architecture de service en fournissant une partie description: il contient toutes les informations nécessaires pour appeler les services qu'il décrit.

WSDL est conçu pour être indépendant de tous protocoles. Ceci rend le standard WSDL flexible mais aussi plus complexe à comprendre. Comme SOAP et HTTP sont les deux protocoles les plus couramment utilisés pour implémenter les services web, le standard WSDL intègre un support de ces deux protocoles. Il y a deux types de documents WSDL :

- Le document WSDL décrivant l'interface du service.
- Le document WSDL décrivant l'implémentation du service.

## Structure d'un document WSDL

- **Types:** fournit la définition de types de données utilisés pour décrire les messages échangés entre le client et le serveur sous le format de XML schéma.
- **Messages:** représente une définition abstraite qui définit les données en cours de transmission entre les services web.
- **PortTypes:** combine plusieurs messages pour former une opération qui définit un couple message-entrée / message – sortie. Chaque opération à zéro ou un message en entrée, zéro ou plusieurs messages de sortie ou d'erreurs.
- **Binding:** c'est un élément qui spécifie une liaison entre un <portType> et un protocole concret (SOAP, HTTP...).
- **Port:** représente un point d'accès de services défini par une adresse réseau et une liaison.
- **Service :** une collection des <Ports>, contenant chacun un nom, une URL de point d'accès et une référence à une liaison donnée.
- **Opération:** c'est la description d'une action exposée dans le port.

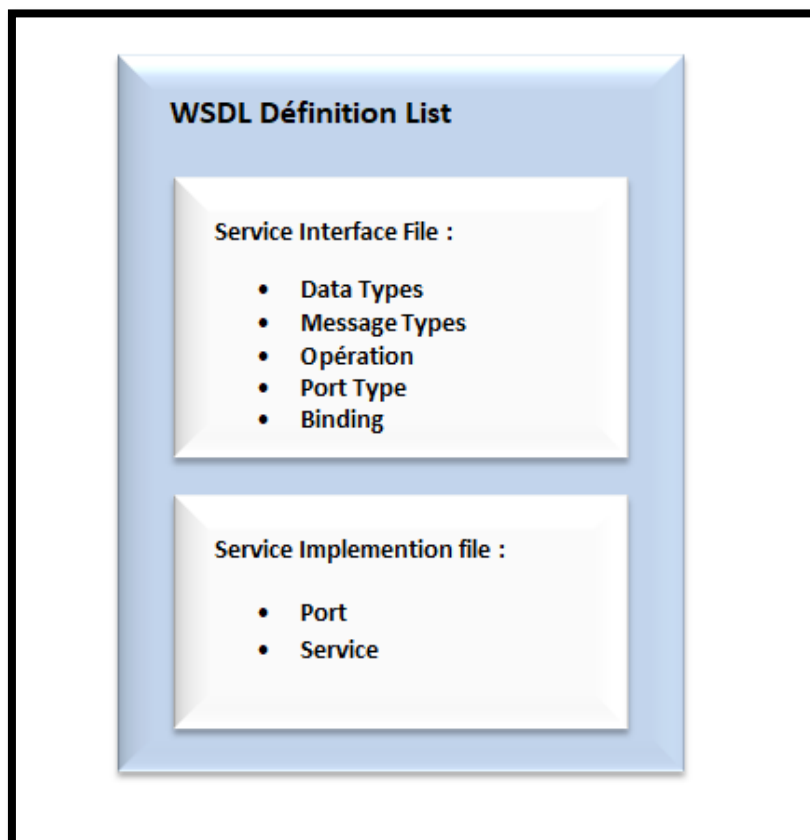


Figure I-9: Structure d'un document WSDL. [25]

### iii. Découverte : UDDI (Universal Description Discovery and Integration)

C'est un annuaire de publication de services, il se repose sur le protocole SOAP et de ce fait, les requêtes et les réponses sur l'annuaire sont des messages SOAP. Cet annuaire est une spécification pour permettre la publication et la recherche d'informations sur des entreprises et les services web qu'elles proposent. UDDI permet à une entreprise de s'inscrire dans l'annuaire, d'y enregistrer et de publier ses services web. Il est alors possible d'accéder à l'annuaire et de rechercher un service particulier.

UDDI comporte plusieurs catégories de données :

- Pages blanches : elles recensent les entreprises et contiennent des informations telles que le nom de l'entreprise, ses coordonnées, des descriptions accessibles aux clients, et des identifiants permettant de la retrouver par recherche.
- Pages jaunes : elles contiennent, au format WSDL, la description des services déployés par l'entreprise. Les services sont répertoriés par catégories.
- Pages vertes : elles fournissent des informations techniques détaillées sur les services.

### 2) Web services REST

De nombreux développeurs ont trouvé SOAP encombrant et difficile à utiliser. Par exemple, travailler avec SOAP en JavaScript signifie écrire une tonne de code pour effectuer des tâches extrêmement simples parce que vous devez créer à chaque fois la structure XML requise. REST offre une alternative plus simple. Au lieu d'utiliser XML pour faire une demande, REST repose sur une simple URL [21] :

#### i. Définition

REST (Representational State Transfer) est une architecture de services web, élaborée en 2000 par Roy Fielding l'un des créateurs du protocole HTTP du serveur Apache et d'autres travaux fondamentaux. Dans ce style d'architecture, les données et les fonctionnalités sont considérées comme des ressources et sont accessibles à l'aide d'un URI, généralement des liens sur le web. Les ressources sont exploitées en utilisant un ensemble d'opérations simples et bien définies. Le style architectural REST contraint une architecture client/serveur et il est conçu pour utiliser un protocole de communication sans état, généralement http. [21]

REST est un ensemble de règles qui vise à créer une application de web Service selon les quatre règles de base ci-dessous :

- Identification de ressource via URI.
- Interface uniforme basée sur les méthodes http.
- Messages auto-descriptifs.
- Être apatride.

## ii. Caractéristiques d'un service web REST

- **Identification de ressource via URI** : un service web RESTful expose un ensemble de ressources qui identifient les cibles de l'interaction avec ses clients. Les ressources sont identifiées par des URI, qui fournissent un espace d'adressage global pour la découverte des ressources et des services.[8]
- **Interface uniforme basée sur les méthodes http** : Cette caractéristique est définie par une interface entre le client et le serveur. Les ressources sont manipulées à l'aide d'un ensemble fixe de quatre opérations de création, de lecture, de mise à jour et de suppression : PUT, GET, POST et DELETE. PUT crée une nouvelle ressource, qui peut être ensuite supprimée en utilisant DELETE. GET récupère l'état actuel d'une ressource dans une représentation. POST transfère un nouvel état sur une ressource.[8]
- **Messages auto-descriptifs** : Les ressources sont découplées de leur représentation de sorte que leur contenu peut être consulté dans divers formats, tels que HTML, XML, JSON, texte brut, PDF, JPEG et autres. Les métadonnées sur la ressource sont disponibles et utilisées, par exemple détecter les erreurs de transmission, négocier le format de représentation approprié et effectuer une authentification ou un contrôle d'accès. [8]
- **Sans état** : un protocole sans état est un protocole de communication qui n'enregistre pas l'état d'une session de communication entre deux requêtes successives. La communication est formée de paires requête-réponse indépendantes et chaque paire requête-réponse est traitée comme une transaction indépendante. [8]

## 3) Comparaison entre SOAOP et REST

SOAP et REST sont les méthodes les plus utilisées pour échanger des données entre un client et un serveur. Il est difficile de comparer les deux méthodes, car elles ont des approches différentes. Comme expliqué dans les sections précédentes, REST est un style d'architecture utilisant la technologie du web tandis que SOAP définit un protocole qui permet d'appeler des

procédures à distance. En effet, REST est conseillé pour un service web orienté ressources qui a besoin d'utiliser les méthodes HTTP (CRUD). SOAP est plutôt utilisé lorsque le CRUD ne suffit pas, dès lors que des opérations plus complexes sont nécessaires.

Le tableau suivant est un résumé des caractéristiques principales de REST et SOAP.

	<b>REST</b>	<b>SOAP</b>
Concept	Style architectural	Protocole
Format d'échange des messages	Utilise XML ou JSON pour envoyer et recevoir des informations	Utilise WSDL pour communiquer entre le client et le serveur
Format d'appel	Appelle simplement les services via un chemin d'URL	Invoque les services en appelant la méthode RPC
Protocole utilisé	Uniquement le protocole http	Principalement HTTP, mais peut utiliser SMTP, FTP, ...
Résultat	Le résultat est lisible, ce qui est tout simplement XML ou JSON	Ne renvoie pas le résultat lisible par l'homme

**Tableau 1: Comparaison REST vs SOAP**

En conclusion, l'approche REST est viable (voire préférable) dans le cas de services Web avec les caractéristiques suivantes :

- Ils ne requièrent pas de garanties de sécurité au-delà de celles offertes par HTTPS
- Ils n'ont besoin d'être exposés que sur HTTP ou HTTPS (HyperText Transfer Protocol Secure).
- De nombreux développeurs ont trouvé SOAP encombrant et difficile à utiliser. Par exemple, travailler avec SOAP en JavaScript signifie écrire une tonne de code pour effectuer des tâches extrêmement simples parce qu'il est impératif de créer à chaque fois la structure XML requise.
- REST offre une alternative plus simple. Au lieu d'utiliser XML pour faire une demande, REST repose sur une simple URL.
- Contrairement à SOAP, REST n'a pas à utiliser XML pour fournir la réponse. Il est possible de trouver des services Web en REST qui ont comme sortie des données au format CSV (Command Separated Value), JavaScript Object Notation (JSON) ou encore Really Simple Syndication (RSS).

## I.4. Synthèse

Après un aperçu sur les différentes approches nous avons conclu :

- L'approche d'intégration de donnée présente des avantages tels que la combinaison des informations provenant de diverses sources pour distribuer une version propre et consolidée dans toute l'entreprise, pour rendre les données accessibles à plusieurs parties prenantes et applications client sans qu'il soit nécessaire de les dupliquer ou de les déplacer et pour une synchronisation des données entre les applications. Alors nous avons choisi cette approche comme solution pour créer une base de données globale ce qu'on appelle une base de données référentielle donc nous avons fait un système d'intégration avec un entrepôt de données pour des fins d'intégration et non pas décisionnel.
- L'entreprise avec laquelle nous traitons notre projet a déjà un data center comme un espace de données virtuelle pour héberger leurs applications et données critiques. Donc cette entreprise à un cloud privé qui représente le data center ce type de cloud est plus sécurisé, renforcé par une connexion VPN en effet, le cloud n'a pas été un choix, ça été imposé par l'organisme.
- Pour la troisième approche SOA est apparu rapidement comme l'approche la plus pratique pour l'intégration des systèmes d'information, permettant de construire une architecture de communication entre des applications hétérogènes. Pour cela nous avons proposé le SOA comme un service entre les utilisateurs et l'architecture d'intégration ce que nous avons mentionné dans la première approche. Ainsi nous avons travaillé avec l'approche REST car nous avons conclu qui est la préférables dans notre cas d'étude pour les services web.

Alors, nous avons adopté la combinaison d'une architecture orientée service et l'architecture d'intégration de l'entrepôt de données pour constituer un système hybride adaptatif d'aide à l'intégration du système d'informations hétérogène et distribué de notre organisme.



## **Conclusion**

Chacune de ces approches cités a bien marqué sa place dans la centralisation, bien évident, chacune a ses avantages et ses inconvénients. Par ailleurs, l'idée sous-jacente est de construire une approche hybride entre le SOA, Cloud et l'intégration. Dans ce contexte nous avons pu faire le tour des principaux aspects de base des trois approches de centralisation.

Cette approche utilise un ensemble de services assurant une fonctionnalité particulière et accessible via son interface. SOA permet la communication d'une manière transparente aux applications du système d'information, elle est capable de répondre rapidement aux nouveaux besoins d'un utilisateur. Ainsi les services web qui sont considérés comme les composants de base les plus utilisés.

***Chapitre II : Etude de l'existant et analyse  
des besoins***

## **Introduction**

L'étude de l'existant et l'analyse des besoins sont des étapes impératives pour commencer tout projet. Pour effectuer notre étude, nous nous sommes basées sur le cahier des charges fourni ainsi que les différents questionnaires déroulés avec les développeurs au niveau d'ELIT.

Dans cette partie, nous entamons par l'étude de l'existant, cette dernière comporte la présentation du groupe SONELGAZ, son historique, son organisation, la présentation du sujet dont nous exposerons la problématique qui se pose au niveau de la structure d'accueil et nous mettrons en avant les objectifs à viser, une étude approfondie sur le système SGC actuel de la SONELGAZ. Par la suite nous détaillerons les besoins fonctionnels et non fonctionnels de notre système.

### **II.1. Étude de l'existant**

L'étude de l'existant est une étape préliminaire nous permettant de prendre connaissance du système existant d'une manière approfondie, découvrir la structure d'accueil, d'explorer ses fonctionnalités, de comprendre son système et de délimiter le cadre général de notre travail. Nous commençons par un aperçu global du groupe SONELGAZ, et son historique.

#### **II.1.1. Présentation de la société SONELGAZ**

La société nationale de l'électricité et du gaz, SONELGAZ, est l'opérateur chargé de la production, du transport et de la distribution d'électricité, du transport et de la distribution du gaz en Algérie. Son statut lui permet aussi l'exportation des deux énergies vers l'étranger. Ayant le monopole de l'énergie dans le marché algérien depuis plus de 50 ans, et de par sa dimension économique, le groupe a connu une forte évolution managériale et organisationnelle. [10]

##### **A. Historique**

SONELGAZ fut créée en 1962, via l'ordonnance N°69-59 le 28 juillet 1969 qui dissout l'établissement public d'Electricité et Gaz d'Algérie (EGA). [10]

En 1969, SONELGAZ était déjà une entreprise de taille importante dont le personnel est de quelque 6000 agents. Elle desservait 700 000 clients. Dès sa mise en place, l'entreprise

a effectué, outre la vente d'énergie, l'installation et l'entretien d'appareils domestiques fonctionnant à l'électricité ou au gaz. [10]

En 1983, SONELGAZ s'est restructurée une première fois et a donné naissance à cinq (05) entreprises de travaux spécialisés ainsi qu'une entreprise de fabrication. C'est grâce à ces sociétés que SONELGAZ dispose actuellement d'infrastructures électriques et gazières répondant aux besoins du développement économique et social du pays. [10]

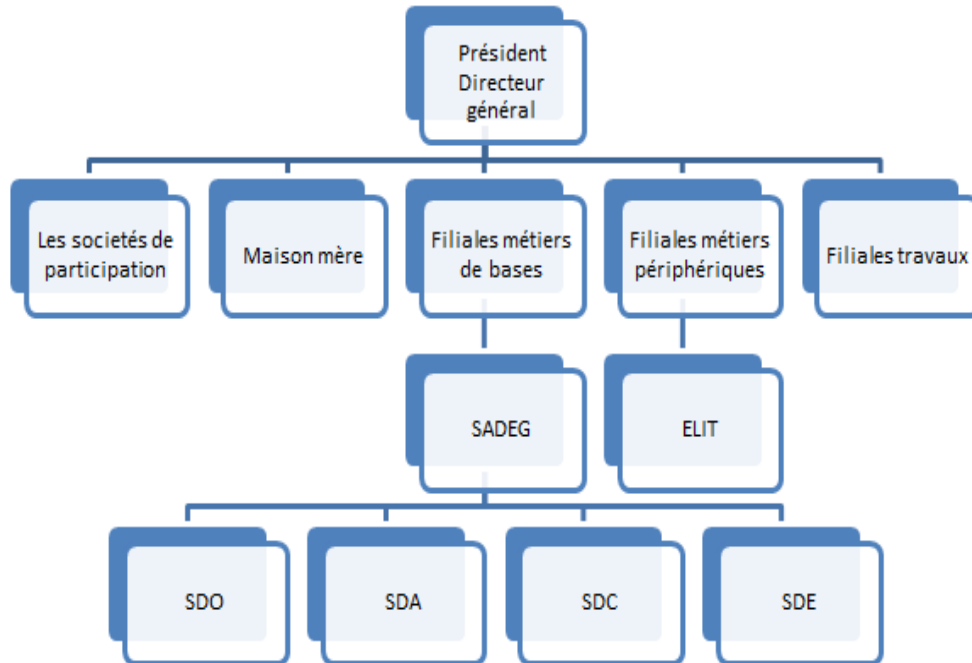
En 2006, cinq (05) autres sociétés sont créées. Durant cette même année, les cinq (05) entreprises de travaux ont réintégré le groupe. Au-delà de cette évolution, assurer le service public reste la mission essentielle de SONELGAZ et constitue le fondement de sa culture d'entreprise.[10]

## **B. Organisation du groupe SONELGAZ**

SONELGAZ s'est dotée d'une nouvelle macrostructure et d'une nouvelle organisation pour assurer ses nouvelles missions. Elle est érigée en groupe industriel constitué d'une maison mère, de 36 filiales et 8 sociétés en participation organisée en quatre pôles : [10]

- **Maison mère** : chargée du pilotage et de l'élaboration de la stratégie du groupe, de la coordination et du contrôle des filiales, de l'élaboration de la politique financière et du développement.
- **Filiales Métiers de base** : il existe 8 filiales. Ces dernières activent dans les domaines de la production, la gestion du réseau de transport, la gestion du système production transport, la distribution de l'électricité et du gaz.
- **Filiales Travaux** : elles sont spécialisées dans le domaine de la réalisation des infrastructures énergétique et l'établissement des réseaux.
- **Filiales Périphériques** : elles sont au nombre de quatorze, chargées des activités périphériques pour le groupe SONELGAZ telles que la maintenance d'équipement énergétiques, la distribution de matériels électriques et gaziers, le transport, la manutention exceptionnels et la recherche et le développement.
- **Sociétés en Participation** : SONELGAZ détient des participations dans des sociétés différentes dans le but de:
  - ✓ Intégrer la technologie et le savoir-faire.
  - ✓ Introduire l'expertise managériale dans le domaine de la gestion.

La figure II-1 permet d'observer la structure organisationnelle de SONELGAZ :



**Figure II-1: L'organigramme de groupe SONELGAZ.[10]**

Ce nouveau schéma organisationnel de la SONELGAZ, décidé par les pouvoirs publics, a induit à la création de la Société Algérienne de Distribution de l'Electricité et du Gaz, dénommée SADEG, c'est dans cette structure que notre projet a été initié.

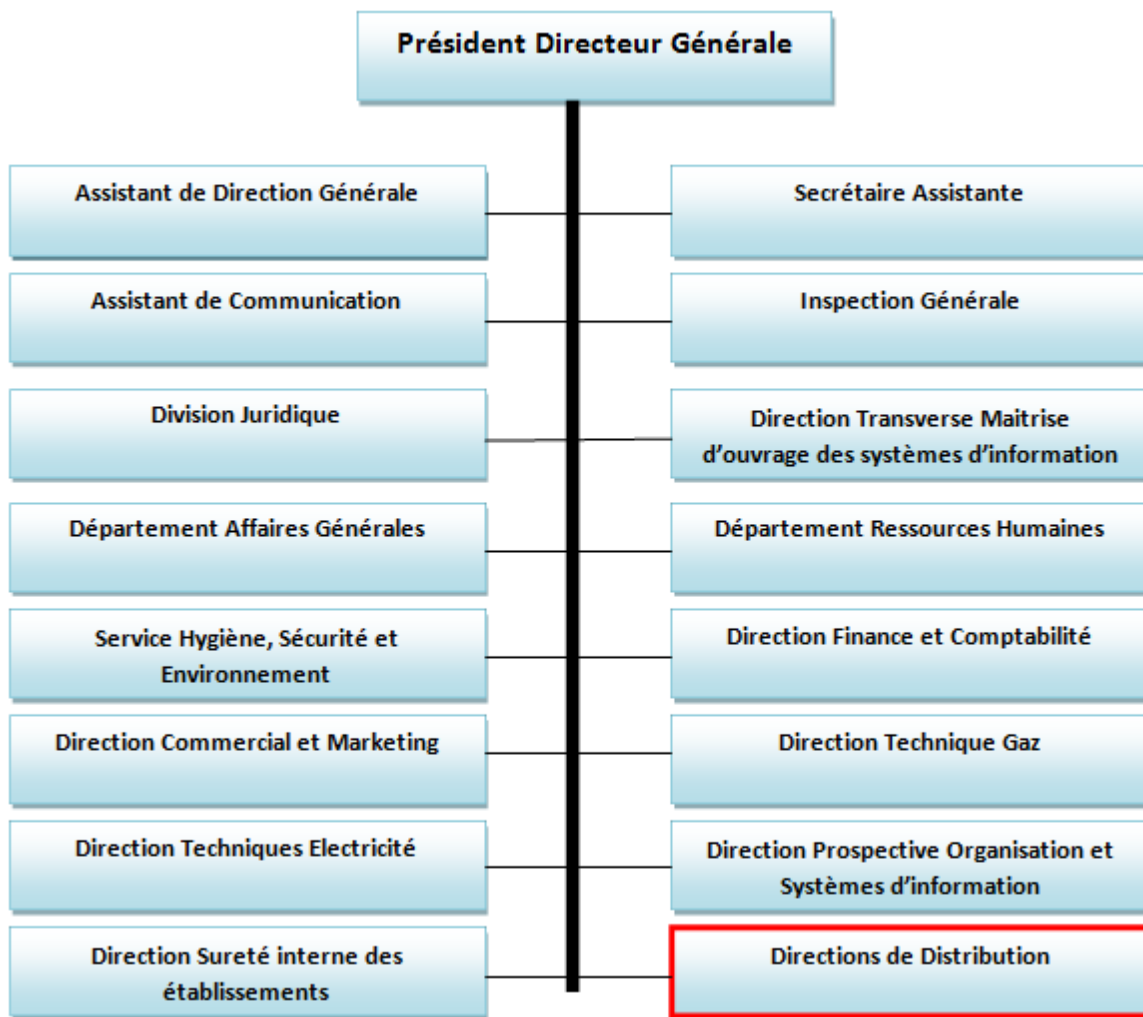
### **C. Présentation de la Société Algérienne de Distribution de l'Electricité et du Gaz(SADEG)**

L'un des métiers les plus importants du groupe SONELGAZ est la distribution de l'énergie électrique et gazière qui est réalisé par la SADEG (Société Algérienne de Distribution de l'Electricité et du Gaz), SPA, résultat de la fusion - absorption des sociétés SDC, SDE, SDO et SDA. Créée le 22 mai 2017.Cette société, selon l'organisation du groupe (voir figure II-1), est divisée en quatre filières qui sont les « Sociétés de Distribution (SD)». Les sociétés sont [10]:

- Sociétés de Distribution d'Electricité et du Gaz de l'Ouest (SDO).

- Sociétés de Distribution d'Electricité et du Gaz du Centre (SDC).
- Sociétés de Distribution d'Electricité et du Gaz d'Alger (SDA).
- Sociétés de Distribution d'Electricité et du Gaz de l'Est (SDE).

Une société de distribution possède la structure décrite par l'organigramme qui montre dans la figure II.2.



**Figure II-2: Organisations des SD. [11]**

Chaque société de distribution régit plusieurs directions de distribution qui gèrent, à leur tour, plusieurs agences commerciales qui présentent le point de contact direct avec le client. La figure II-3 illustre l'organisation autour de la fonction de distribution de l'électricité et du gaz. [10]

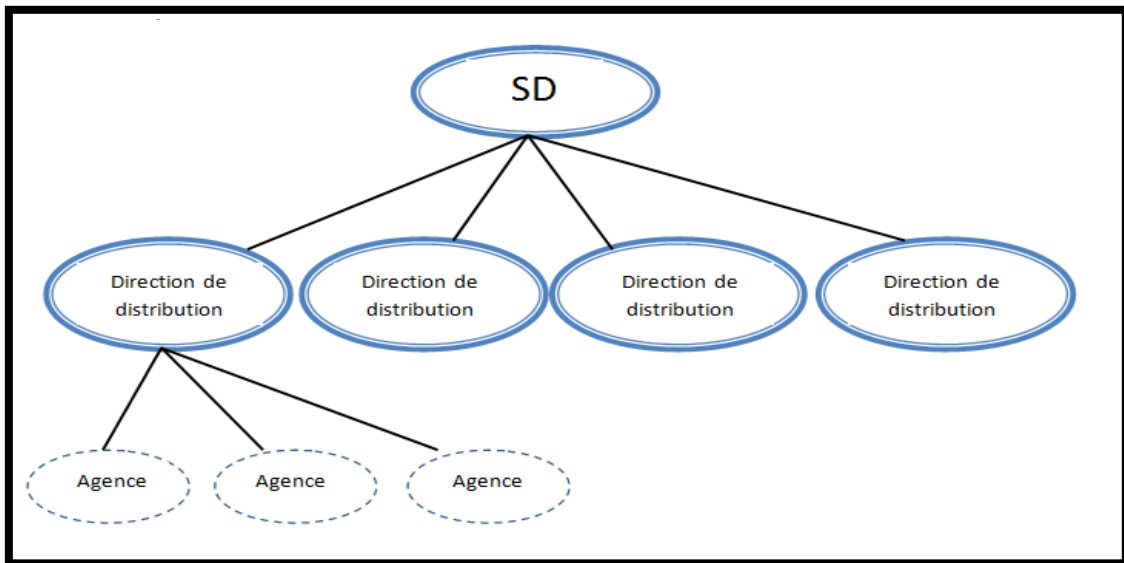


Figure II-3: Hiérarchie de la SD. [10]

#### a. Mission et tâches

La société de distribution d'électricité et du gaz a pour missions. [10]:

- L'exploitation et la maintenance du réseau de distribution de l'électricité et du gaz.
- Le développement des réseaux électricité et gaz permettant le raccordement nouveaux clients.
- La commercialisation de l'électricité et du gaz.
- Garantir la qualité et la continuité de service dans la distribution de l'électricité et du gaz.
- Garantir la sécurité des réseaux électricité et gaz.

#### b. La clientèle de la distribution

Les clients sont classifiés selon le type de l'énergie et la puissance d'alimentation comme suit [11] :

En électricité on distingue trois types de clients selon la tension :

• **Client « Basse Tension-BT »** : jusqu'à une puissance de 40KVA.L'abonné est considéré comme un client BT. La tension délivrée est 220 V ou 380 V.

• **Client « Haute Tension A-HTA »** : d'une puissance de 50 KVA jusqu'à 360 KVA. La tension d'alimentation est de 10 KV ou 30KV.

- **Client « Haute Tension B-HTB »** : la tension d'alimentation dépasse 60KV.

En gaz, les clients sont classifiés selon la pression :

- **Client « Basse Pression-BP »** : tout client alimenté sous une pression de 21 bars à travers un détendeur.
- **Client « Haute Pression-HP »** : tout client alimenté sous une pression supérieure à 4bars avec un poste de détente gaz.

En BT/BP (basse tension et basse pression), on distingue plusieurs types de clients :

- **Clients ordinaires** :
  - ✓ **Clients ménages** : il s'agit des clients domestiques.
  - ✓ **Clients non ménages** : il s'agit des clients non domestiques (Activités commerciales).
- **Clients FMS (Facturation Multi-sites).**
- **Clients ADM (administratifs)** : ce sont les établissements publics tels que les écoles, les hôpitaux, etc.
- **Clients particuliers**: il s'agit des clients ayant des activités sur plusieurs sites tels que les usines industrielles.

#### **D. Les systèmes d'informations déployées au niveau SADEG**

SONELGAZ est une entreprise très importante ait plusieurs systèmes d'informations opérationnels à sa charge, parmi elle [10]:

##### **a) Système de Gestion Finances et Comptabilité "HISSAB"**

ELIT HISSAB est un système de gestion automatisé qui vous aidera à tenir votre comptabilité tout en enregistrant un gain de temps considérable et une traçabilité infaillible. C'est un système permettant de gérer votre comptabilité en conformité avec les normes inscrites dans le nouveau système comptable et financier "SCF".

##### **b) Système de Gestion de la Clientèle SGC**

Le SGC ou gestion de la relation client de la SONELGAZ appelé aussi CRMS (nouvelle version du SGC avec nouveau design on ne sait pas encore quelles sont les nouvelles fonctionnalités) est une stratégie de gestion des relations et des interactions de la



SONELGAZ avec ses clients, dans le but de les fidéliser tout en leur offrant ou proposant des services. Ce dernier apportera une aide aux commerciaux pour mieux s'organiser, accéder à l'information interne utile et améliorer leur productivité. Il permettra ainsi de remonter d'un cran dans le pilotage avec un suivi des actions et des affaires en cours. Le SGC est un ensemble des modules qui gère la relation avec le client, depuis sa demande d'alimentation en électricité ou en gaz jusqu'à l'intégration et la signature du contrat reliant le client avec la SONELGAZ. [11]

Ce qui nous intéresse dans notre projet, c'est d'étudier l'amélioration du SGC et voir comment l'améliorer pour répondre aux besoins actuels.

### E. L'architecture fonctionnelle du SGC

Le SGC est composé de plusieurs modules, qui assurent de différentes fonctionnalités. Parmi les modules importants qui représentent dans la figure II-4, nous citons [11]:

- **Raccordement** : est un processus propre au métier des gestionnaires de réseaux. Il prend en charge toute création ou modification d'ouvrages sur le réseau de distribution.
- **Relève** : la relève consiste à collecter les index lus sur les cadrans des compteurs (d'électricité et du gaz) sur le lieu de consommation ou à distances et de constater toute anomalie et la consigner sur un rapport journalier de relève.
- **Facturation** : le module de facturation est chargé d'établir une facture d'énergie soit gaz ou électricité au client afin de lui permettre de régler sa consommation en énergie électrique et/ou gazière.

SONELGAZ distingue trois types de factures d'énergie :

- ✓ **Facture cyclique** : c'est une facture établie périodiquement en fonction du calendrier de relève, La facture est mensuelle pour les clients (HTA/HTB/HP/MP) contrairement aux clients (BP/BT) qui est trimestrielle.
- ✓ **Facturation hors cycle** : une facture hors cycle est élaborée suite à la demande d'un agent chargé des événements.
- ✓ **Facturation de prestation** : une facture de prestation est une facture hors cycle qui est générée uniquement dans le cas où le client choisit que les travaux liés à sa demande soient réalisés par SONLEGASZ.

- **Achat** : la société de production d'énergie dispose de plusieurs points de livraison, Ces derniers livrent l'énergie aux sociétés de distribution. Les achats d'énergie d'une SD correspondent à la quantité enregistrée à travers la mesure des compteurs installés au niveau de ses PDL, sachant qu'un PDL contient des index qui indiquent cette quantité brute prise par la SD, à partir de cette quantité la SD distribue l'énergie à ses clients.
- **Recouvrement** : le recouvrement constitue un enjeu commercial important vu son impact et son influence sur la trésorerie. Ce dernier consiste à suivre et mettre à jour l'état des créances et des contrats après présentations des factures cycliques.
- **Réclamation** : une réclamation est l'expression orale ou écrite d'une insatisfaction formulée par un client (ou par un citoyen d'une façon générale) à l'égard des produits et services offerts.
- **Gestion des évènements** : le processus gestion des évènements permet le suivi du traitement des demandes adressées par les clients ou décidées par le niveau opérationnel ou décisionnel, depuis leurs enregistrements jusqu'à leurs aboutissements.
- **Statistiques et Reporting** : fournis des statistiques aux décideurs concernant les activités liées à la gestion de la clientèle.

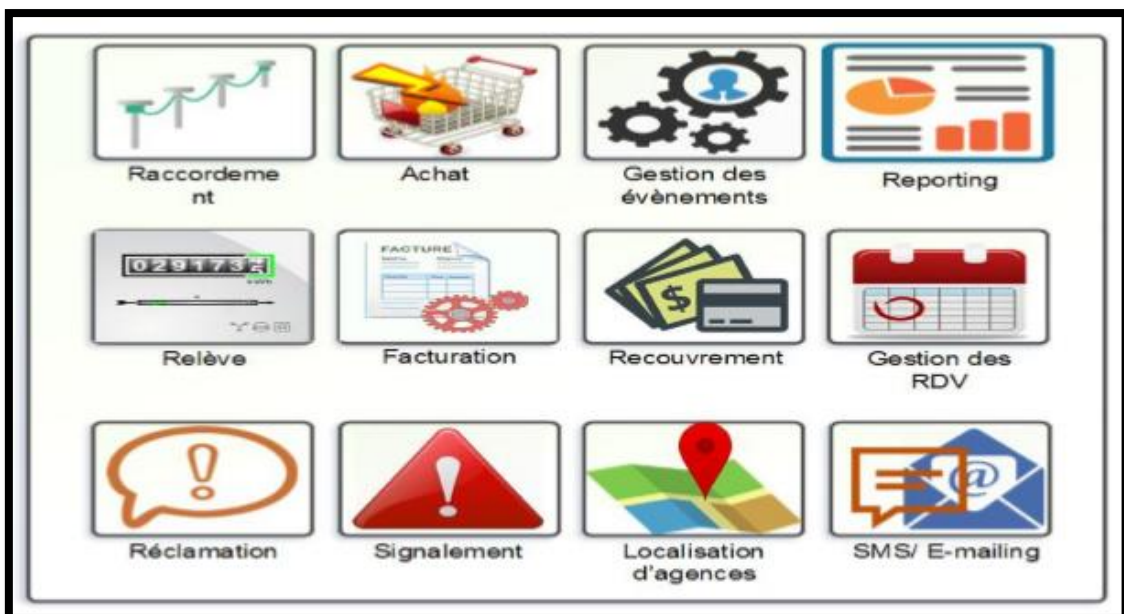


Figure II-4: L'architecture fonctionnelle du SGC. [11]

## F. L'architecture technique du SGC

La filiale ELIT suit une politique open source pour le développement des systèmes informatiques. Pour le SGC, l'équipe de la DSID a opté pour les technologies suivantes [11] :

- La plateforme Java EE pour le développement du SGC.
- Le SGBD PostgreSQL pour la BDD du SGC.

Le système SGC été conçu selon une architecture client-serveur léger comme montré sur la figure II-5:

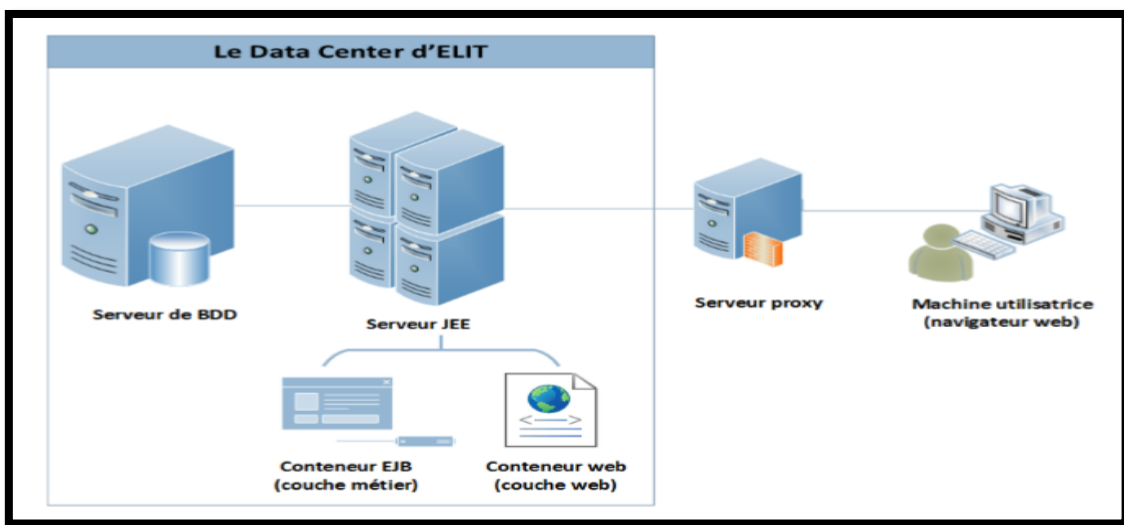


Figure II-5: L'architecture technique du SGC. [11]

## G. La BDD du SGC

La consolidation des données de la clientèle des SDs étant l'un des objectifs prioritaires du projet SGC, a limité le choix de l'équipe à une seule solution inévitable qui est la construction d'une BDD centralisée pour le SGC, regroupant toutes les données provenant des différentes agences commerciales du territoire national. Cette BDD contient environs 1000 tables et qu'est hébergée au niveau du data center d'ELIT (la BD est donnée dans l'annexe A).

Malheureusement la conception de cette base de données a été faite dans l'optique d'avoir d'un système centralisé, toutefois le déploiement réel est purement décentralisé, aucune connexion entre les différentes bases de données n'a été établie, chaque Direction de Distribution (DD) a sa propre base de données indépendamment des autres DD. [11]

### **II.1.2. Diagnostic**

Après plusieurs séances de travail avec l'organisme d'accueil et les ingénieurs, et suite à l'étude que nous avons effectuée sur les sources du système nous avons observé plusieurs anomalies et dysfonctionnements à savoir :

- Le déploiement et le fonctionnement réel du SGC posent beaucoup des problèmes, il ne répond plus à tous les besoins de l'entreprise, selon notre constat on a remarqué que le SGC est conçu comme un système centralisé, et malheureusement est déployé au niveau de chaque direction, en effet, il est dupliqué partout avec des bases de données complètement indépendantes et aucune liaison n'existe entre eux, on a 53 directions de distribution et chacune avec sa propre base de données.
- Manque d'interopérabilité entre le SGC et les autres solutions métiers tel que la comptabilité.
- L'utilisation de plusieurs langages de programmation dans les différents systèmes d'informations déjà déployé qui pose un problème d'interopérabilité et d'accès aux données (l'accès de manière transparente).
- Il y a des informations communes à travers toutes les directions de distribution que l'utilisateur global doit les voir, mais la vue globale de données n'existe pas, car les informations sont dupliquées.
- L'absence de synchronisation des données entre les différentes directions de distribution.

Ces constats et ces anomalies nous ont permis de ressortir un ensemble des problèmes qu'il faudra résoudre, nous les citerons en détail dans ce qui suit.

### **II.1.3. Présentation du sujet**

Le groupe SONELGAZ est l'opérateur historique et leader du domaine électrique et gazier en Algérie.

Depuis la création ELIT, elle s'est pris la tâche de développer les différents systèmes et solutions informatiques selon les besoins métiers de ses clients (les différentes filiales du groupe ainsi que les clients externes du groupe), ce qui a engendré le déploiement de ces derniers.

Au fil du temps, il a été nécessaire d'interfacer ces solutions entre eux afin de gérer l'information dans un seul système et la circuler entre les autres. Cette façon de faire est devenue une nécessité vitale pour assurer la bonne gestion. L'activité commerciale du SONELGAZ est distribuée sur le territoire national, soit 53 directions de distribution, et chacune avec sa propre base de données.

### **a) Problématique**

Les problèmes engendrés peuvent être résumés dans les points suivants :

- Le SGC (Système de Gestion Clientèle) est déployé pour chaque direction de distribution, ce que rend l'administration fonctionnelle (la gestion des tarifs, PMD, DMD, ....) du système une tâche difficile.
- Le partage de certaines données avec d'autres solutions métiers tel que le système comptable.
- Comment accéder aux solutions métiers (système comptable) de manière transparente tout en faisant abstraction des différents langages de programmation, et des différentes plateformes.
- L'absence de synchronisation des données entre les différentes directions de distribution.

### **b) Objectif**

Pour atteindre à notre objectif, nous avons procédé par l'étude approfondie de ce qui se passe dans ELIT : documents échangés, interviews des personnes clés, cahier de charge reporting commercial, étude des besoins pour déterminer les facteurs clés de succès de l'entreprise.

Alors l'objectif principal de notre travail est de développer une plateforme de service basée sur l'architecture SOA offrant au différent composant du système de gestion commercial les fonctionnalités suivantes :

- Centraliser la gestion des données de base du système à savoir (la gestion des tarifs, PMD, DMD,....).
- Synchroniser les données entre les différentes directions de distribution.

- Assurer la communication entre les différentes solutions métier et le système de gestion commercial (SGC).
- Assurer la sécurité de la plateforme de service proposée.

## II.2. Analyse des besoins

L'analyse des besoins nous permet de définir les besoins fonctionnels et non fonctionnels des différents acteurs intervenants dans notre projet, cette analyse ainsi que la conception seront effectuées selon la démarche RUP (Relation Unified Process) basée sur le langage UML.

### II.2.1. Le langage UML

UML est un langage visuel constitué d'un ensemble de schémas, appelés des diagrammes, qui donnent à chacun une vision différente du projet à traiter. UML nous fournit donc des diagrammes pour représenter le logiciel à développer : son fonctionnement, sa mise en route, les actions susceptibles d'être effectuées par le logiciel, . . . etc. Réaliser ces diagrammes revient donc à modéliser les besoins du logiciel à développer. [26]

Si UML permet de modéliser un système, il ne définit pas le processus d'élaboration des modèles.

- Dans quel ordre doit-on utiliser les types de diagrammes?
- A quel moment de la conception d'un système doivent-ils intervenir?

Seul un processus de développement peut répondre à ces questions.

#### 1) Le choix de la méthode

Une méthode de développement définit à la fois un langage de modélisation et la marche à suivre lors de la conception. Le langage UML propose uniquement une notation dont l'interprétation est définie par un standard, mais pas une méthodologie complète. Plusieurs processus de développement complets fondés sur UML existent, comme UP, RUP, UP agile, XP, 2TUP .....[27]

Parmi ses méthodes notre choix est porté sur la méthode RUP (Rational Unified Process).

➤ **Rational Unified Process (RUP)**

Processus unifié (PU) est une méthode de prise en charge du cycle de vie d'un logiciel et donc du développement, pour les logiciels orientés objets. [27]

RUP est l'une des plus célèbres implémentations de la méthode PU permettant de donner un cadre au développement logiciel. [27]

Le Rational Unified Process (RUP) est un processus de développement de logiciel, il permet d'affecter selon une approche disciplinée à l'affectation des tâches et des responsabilités au sein d'une organisation de développement. Son but est de guider pour une utilisation efficace d'UML : il précise de quel modèle on a besoin, pour quelles raisons on doit l'utiliser et comment le construire. Très complet, il couvre l'ensemble des activités, depuis la conception du projet jusqu'à la livraison de la solution. [27]

➤ **Caractéristiques essentielles du RUP**

Les Caractéristiques de RUP sont [27]:

- RUP est à base de composants.
- RUP utilise le langage UML.
- RUP est piloté par les cas d'utilisation : l'objectif principal d'un système logiciel est de rendre service à ses utilisateurs, il faut par conséquent bien comprendre les désirs et les besoins des futurs utilisateurs.
- RUP est centré sur l'architecture : dès le démarrage du processus, on aura une vue sur l'architecture à mettre en place. L'architecture d'un système logiciel peut être décrite comme les différentes vues du système qui doit être construit. L'architecture logicielle équivaut aux aspects statiques et dynamiques les plus significatifs du système.
- RUP est itératif et incrémental : l'itération est une répétition d'une séquence d'instructions ou d'une partie de programme un nombre de fois fixé à l'avance ou tant qu'une condition définie n'est pas remplie, dans le but de reprendre un traitement sur des données différentes.

➤ **Les phases du RUP**

Les phases du cycle de vie de la méthodologie RUP [27] :

- La création : évaluation des risques, architecture, planification.

- L'élaboration : spécification du besoin, validation, environnement de projet.
- La construction : production du logiciel et de la documentation support, tests.
- La transition : test système et utilisateur, correction, déploiement.

### II.2.2. Identification des acteurs

Un acteur représente un rôle joué par une entité externe (utilisateur humain, dispositif matériel ou autre système) qui interagit directement avec le système étudié, autrement dit un acteur peut consulter et/ou modifier directement l'état du système, en émettant et/ou en recevant des messages susceptibles d'être porteurs de données [26]. Dans le cadre de notre étude, nous avons distingué deux acteurs qui interagissent directement avec le système :

- **Administrateur fonctionnel principal:** Le directeur commercial au niveau du siège de la SADEG aura le rôle de l'administrateur fonctionnel principal, il aura comme tâche principale la gestion toutes les fonctionnalités de notre système au niveau siège(SADEG), qui en réalité un directeur commercial niveau siège.
- **Administrateur fonctionnel régional :** c'est un directeur commercial niveau régional de la SD, il aura comme tâche la consultation de notre système.

### II.2.3. Spécification des besoins fonctionnels

Pour pouvoir spécifier les besoins fonctionnels de notre étude, le développement de l'application est décomposé en parties, chaque partie est constituée de plusieurs fonctionnalités, Cette répartition a pour but de simplifier la construction et la compréhension de l'application. La présente plateforme doit satisfaire les besoins fonctionnels suivants :

- **Gestion des données de base :** concerne toute les données qui sont communes à toutes les directions de distribution, à savoir :
  - Gestion des tarifs.
  - Gestion des communes.
  - Gestion des secteurs économiques.
  - Gestion des activités économiques.
  - Gestion des PMD/DMD.
  - Gestion des banques.
  - Gestion des wilayas.



- **Gestion des jobs de synchronisation** : sont des procédures permettant la migration de donnée d'une application à une autre. Notre travail concerne:
  - L'installation des tâches de synchronisation.
  - L'activation et désactivation des tâches de synchronisation.
- **Gestion des actes de gestion** : sont des mouvements comptable associé à chaque montant qui existe sur une facture qui concerne :
  - L'ajout d'un mouvement.
  - La mise à jour d'un mouvement.
  - La consultation des mouvements.

#### **II.2.4. Diagramme de cas d'utilisation**

Un cas d'utilisation ou use case, représente un ensemble de séquences d'actions qui sont réalisées par le système et qui produisent un résultat observable, intéressant pour un acteur particulier. Chaque cas d'utilisation spécifie un comportement attendu du système comme un tout sans imposer le mode de réalisation de ce comportement. On l'identifie en recherchant les différentes interactions avec lesquelles un acteur utilise le système et en déterminant les services fonctionnels attendus du système. Il permet donc de structurer les besoins des utilisateurs. [26]

##### **A. Diagramme de cas d'utilisation globale**

Notre système doit atteindre son objectif principal tout en respectant les exigences.

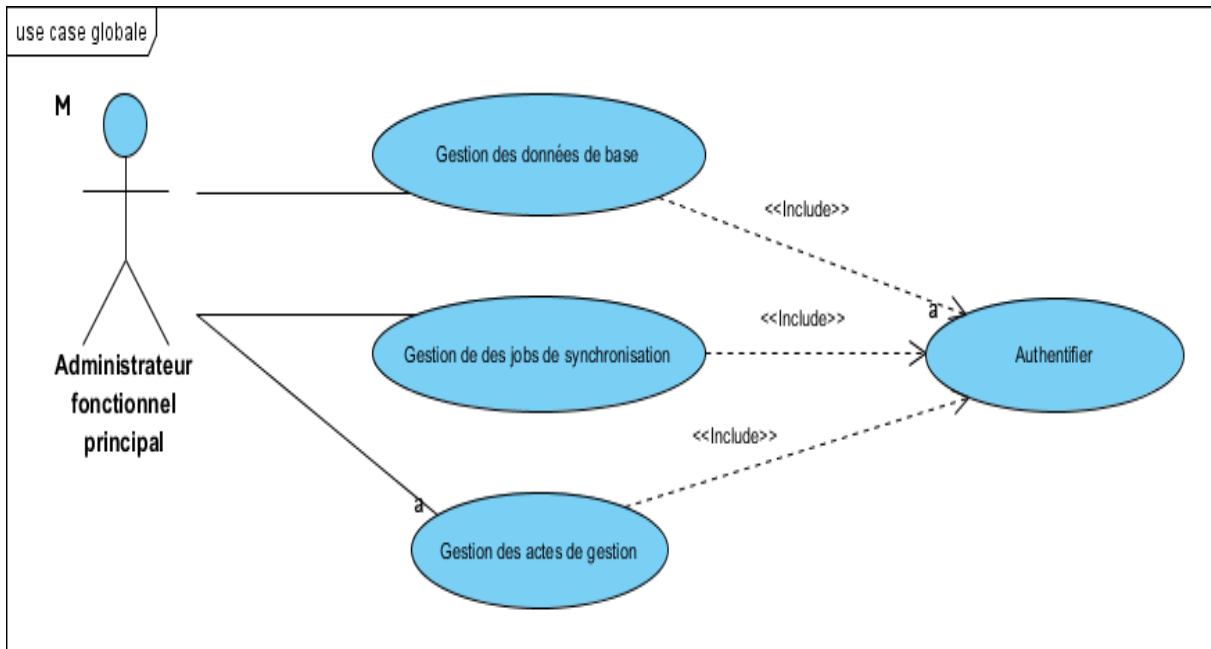


Figure II-6: Diagramme de cas d'utilisation globale.

## B. Diagrammes de cas d'utilisations Principaux

Les principaux diagrammes de cas d'utilisations sont:

### a) Diagramme de cas d'utilisation « Gestion des données de base»



Figure II-7: Diagramme de cas d'utilisation « Gestion des données de base».

<b>Acteur</b>	<b>Use case</b>	<b>Description</b>
Administrateur fonctionnel niveau siège	Gestion des tarifs	La gestion des tarifs concerne en réalité la mise à jour et la consultation des tarifs unitaires appliqués à tous les clients d'une catégorie donnée ou d'une région déterminé.
Administrateur fonctionnel niveau siège	Gestion des communes	La gestion des communes concerne la mise à jour et la consultation des communes qui détermine où réside le client.
Administrateur fonctionnel niveau siège	Gestion des secteurs économique	La gestion des secteurs économiques s'intéresse à la mise à jour et à la consultation des secteurs qui représentent une catégorie des clients.
Administrateur fonctionnel niveau siège	Gestion des activités économiques	La gestion des activités économiques concerne la mise à jour et la consultation des activités appartenant aux secteurs données
Administrateur fonctionnel niveau siège	Gestion des PMD/DMD	La gestion des PMD concerne l'ajout d'une valeur de puissance mise à disposition. La gestion des DMD concerne l'ajout d'une valeur de débit mis à disposition.
Administrateur fonctionnel niveau siège	Gestion des banques	La gestion des banques s'occupe la mise à jour des banques existantes à l'échelle nationale pour faciliter les relations avec les clients.
Administrateur fonctionnel niveau siège	Gestion des wilayas	La gestion des wilayas concerne la mise à jour et la consultation des wilayas qui détermine où réside le client.

**Tableau 2: Description de cas d'utilisation « Gestion des données de base»**

**b) Diagramme de cas d'utilisation « Gestion des jobs de synchronisation»**



Figure II-8: Diagramme de cas d'utilisation « Gestion des jobs de synchronisation».

Acteur	Use case	Description
Administrateur fonctionnel niveau siège	Installer un job	Créer un nouveau job qui est un déclencheur pour la migration de données.
Administrateur fonctionnel niveau siège	Modifier les informations d'un job	Modifier les informations de déclencheurs d'un job.
Administrateur fonctionnel niveau siège	Activer un job	Pour rendre le job en état en marche.
Administrateur fonctionnel niveau siège	Désactiver un job	Pour rendre le job en état non marche
Administrateur fonctionnel niveau siège et régional.	Consulter les jobs	Pour chercher des jobs précis

Tableau 3: Description de cas d'utilisation «Gestion des jobs de synchronisation»

c) Diagramme de cas d'utilisation « Acte de gestion»

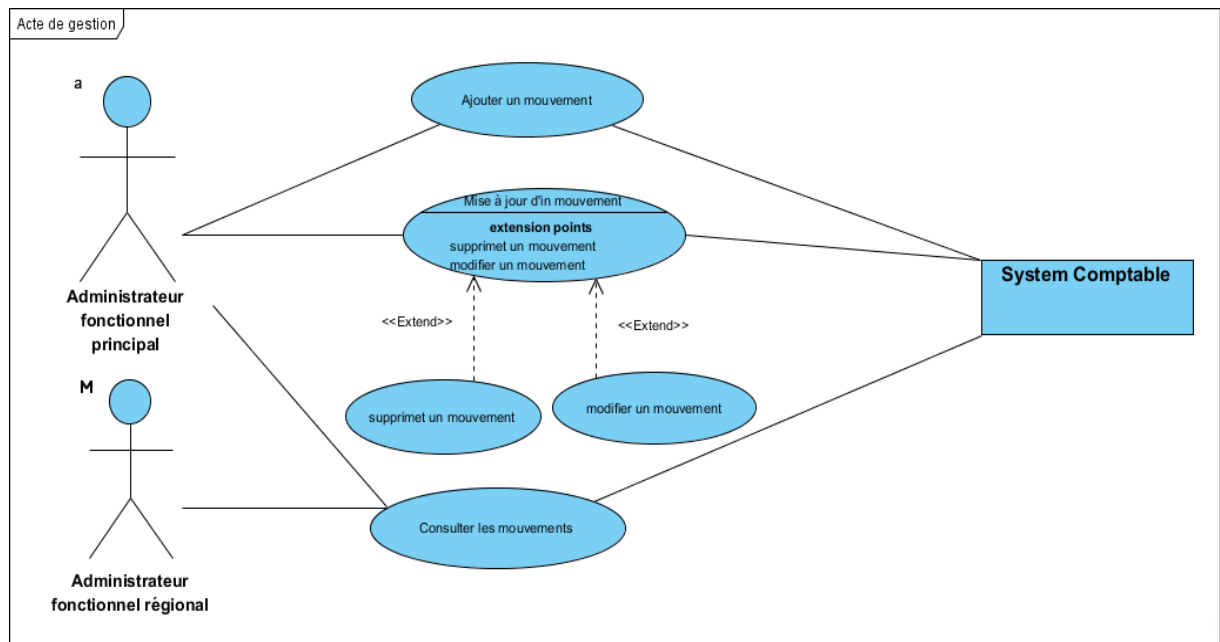


Figure II-9: Diagramme de cas d'utilisation « Acte de gestion».

Acteur	Use case	Description
Administrateur fonctionnel niveau siège	Ajouter un mouvement	Concerne en réalité la création des mouvements comptables associés à chaque montant qui existe sur une facture.
Administrateur fonctionnel niveau siège et régional	Consulter les mouvements	La consultation concerne la recherche des mouvements précis.
Administrateur fonctionnel niveau siège	Mise à jour d'un mouvement	S'intéresse à la suppression ou à la modification d'un mouvement spécifié.

Tableau 4: Description de cas d'utilisation « Acte de gestion»

### C. Diagramme de cas d'utilisation détaillés

On cite un seul diagramme de cas d'utilisation seulement pour comprendre les séquences d'actions qui sont réalisées par le système car toute la gestion de données de base est la même.

a) Diagramme de cas d'utilisation « Gestion des tarifs »

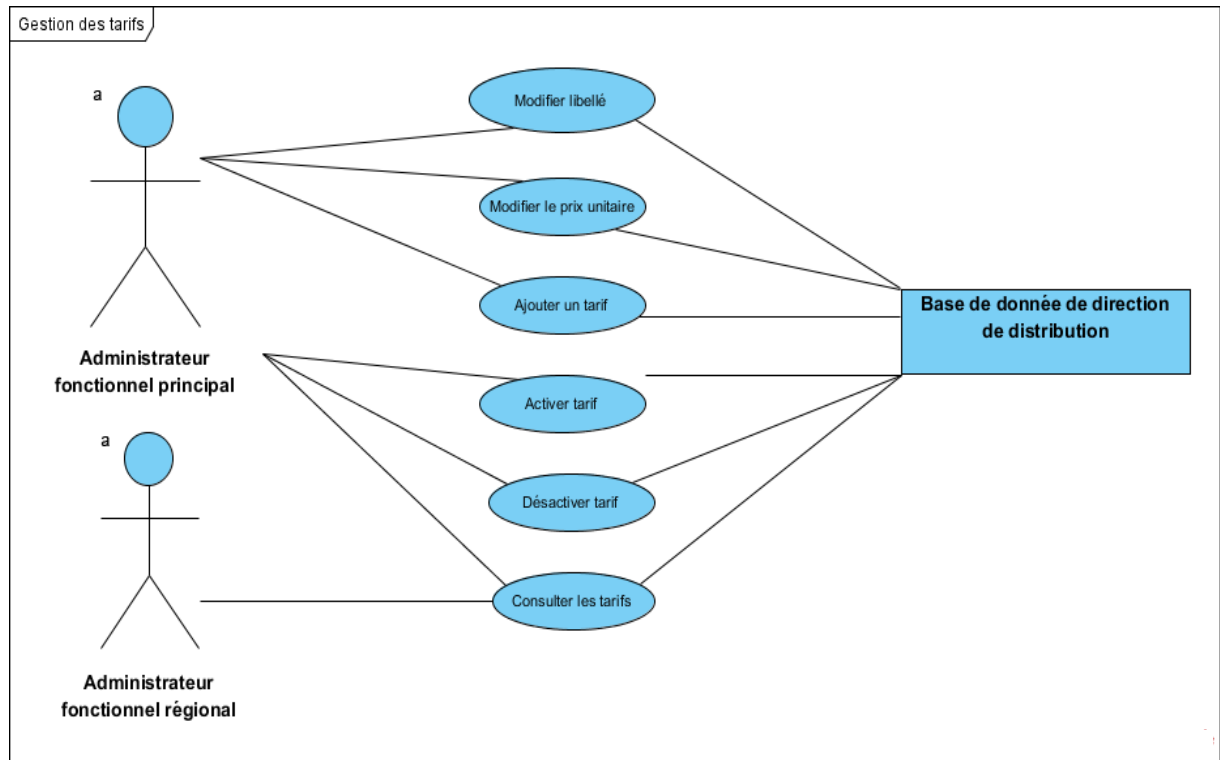


Figure II-10: Diagramme de cas d'utilisation « Gestion des tarifs ».

Acteur	Use case	Description
Administrateur fonctionnel niveau siège	Modifier libellé	Permettre à l'administrateur de modifier le libellé d'un tarif.
Administrateur fonctionnel niveau siège	Modifier le prix unitaire	Permettre à l'administrateur de modifier le prix unitaire d'un tarif.
Administrateur fonctionnel niveau siège	Ajouter un tarif	Permettre à l'administrateur d'ajouter un nouveau tarif.
Administrateur fonctionnel niveau siège	Activer un tarif	Permettre à l'administrateur d'activer un tarif.
Administrateur fonctionnel niveau siège	Désactiver un tarif	Permettre à l'administrateur de désactiver un tarif.
Administrateur fonctionnel niveau siège et régional	Consulter les tarifs	S'intéresse à la recherche des tarifs spécifiés.

Tableau 5: Description de cas d'utilisation « Gestion des tarifs »

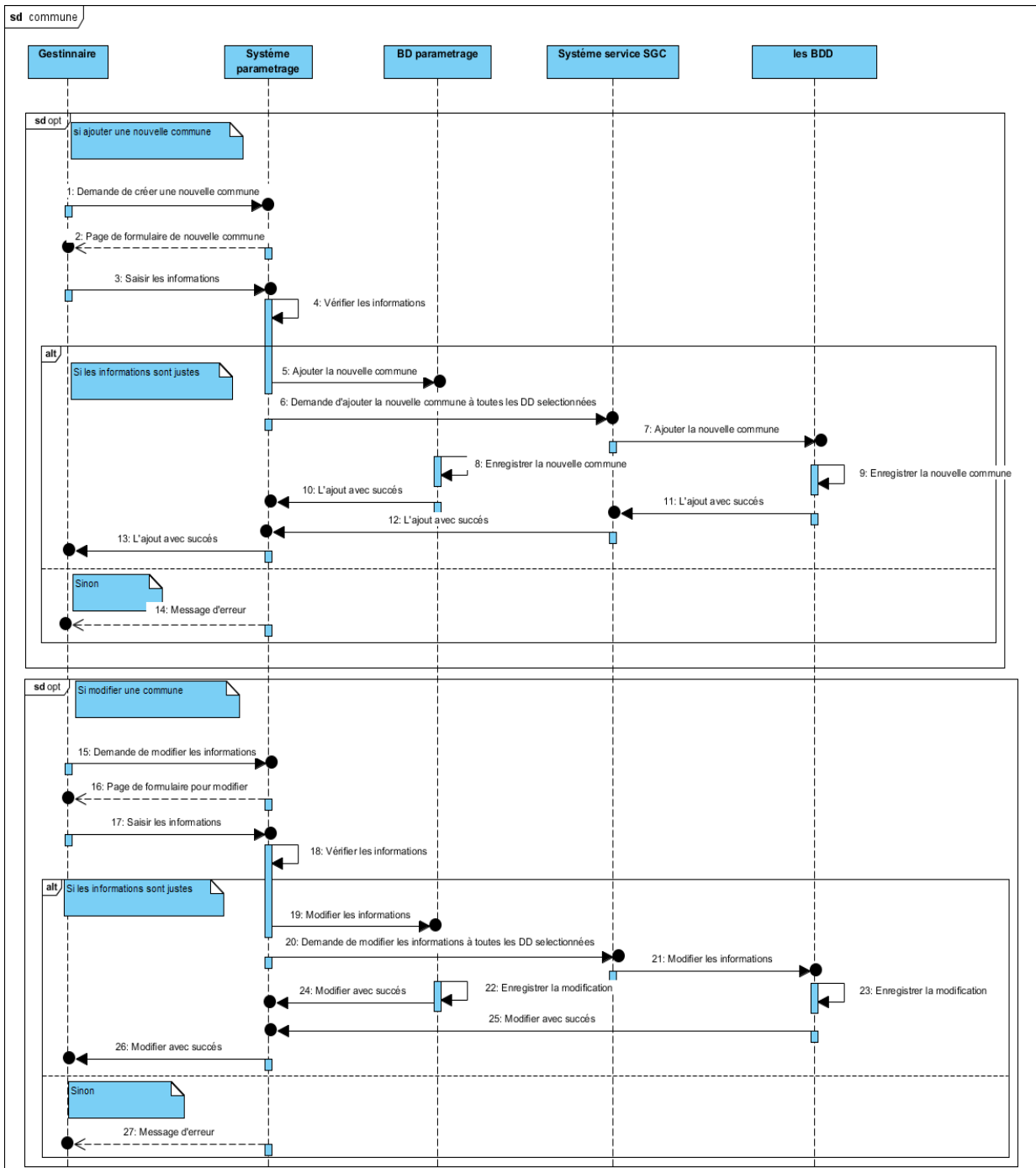
### **II.2.5. Diagramme de séquences:**

Maintenant on présente les diagrammes de séquences qui peuvent montrer l'interaction des séquences de messages entre les instances du système (classes, composants, sous-systèmes ou les acteurs), ce qui peut être réalisé grâce à des cas d'utilisation en considérant les différents scénarios associés.[26]

#### **A. Diagramme de séquence du cas d'utilisation « gestion données de base »**

On cite deux diagrammes de séquences seulement pour comprendre les interactions de messages entre les instances du système car toute la gestion de données de base est la même.

a. Diagramme de séquence « gestion des communes »





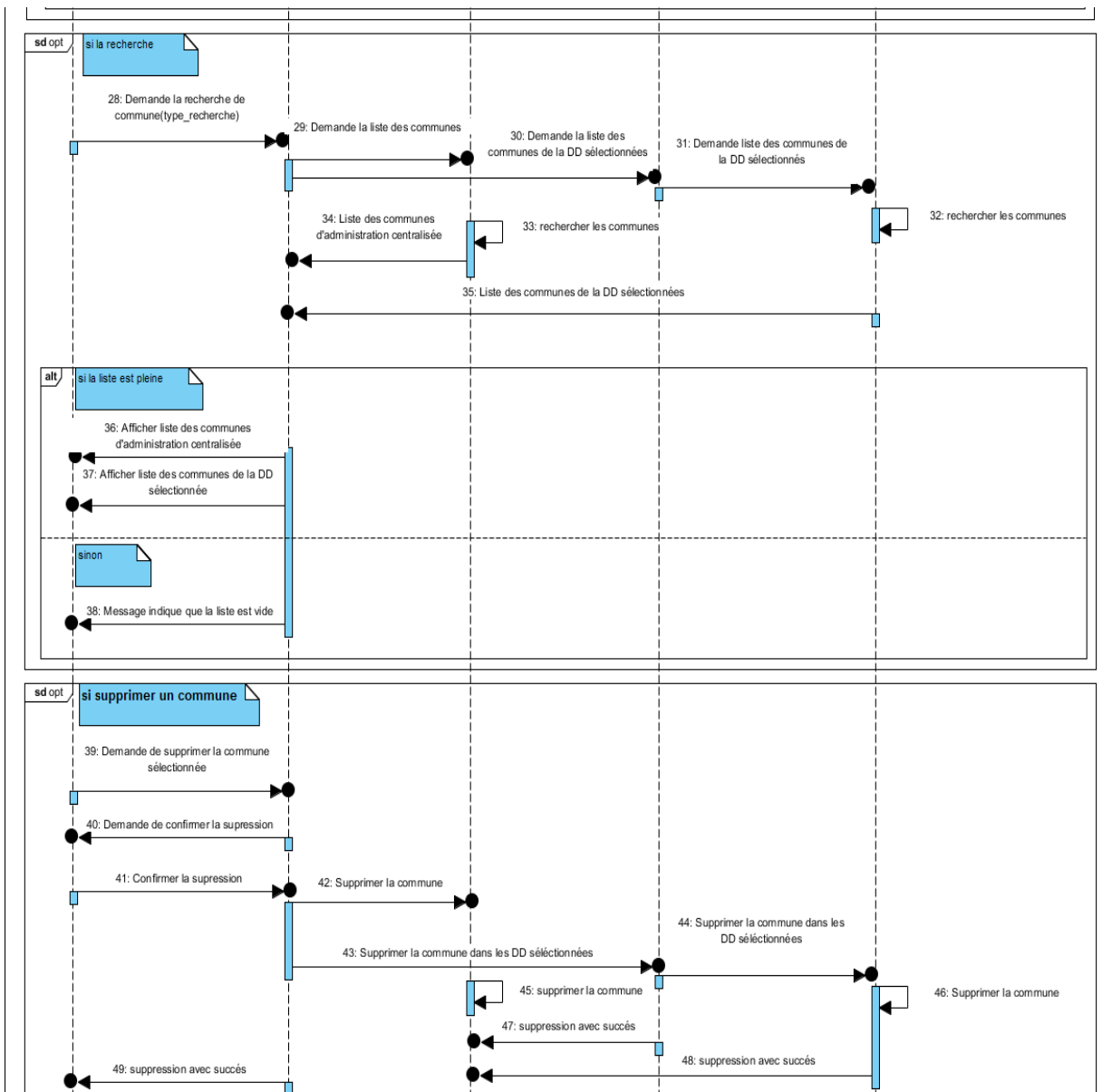
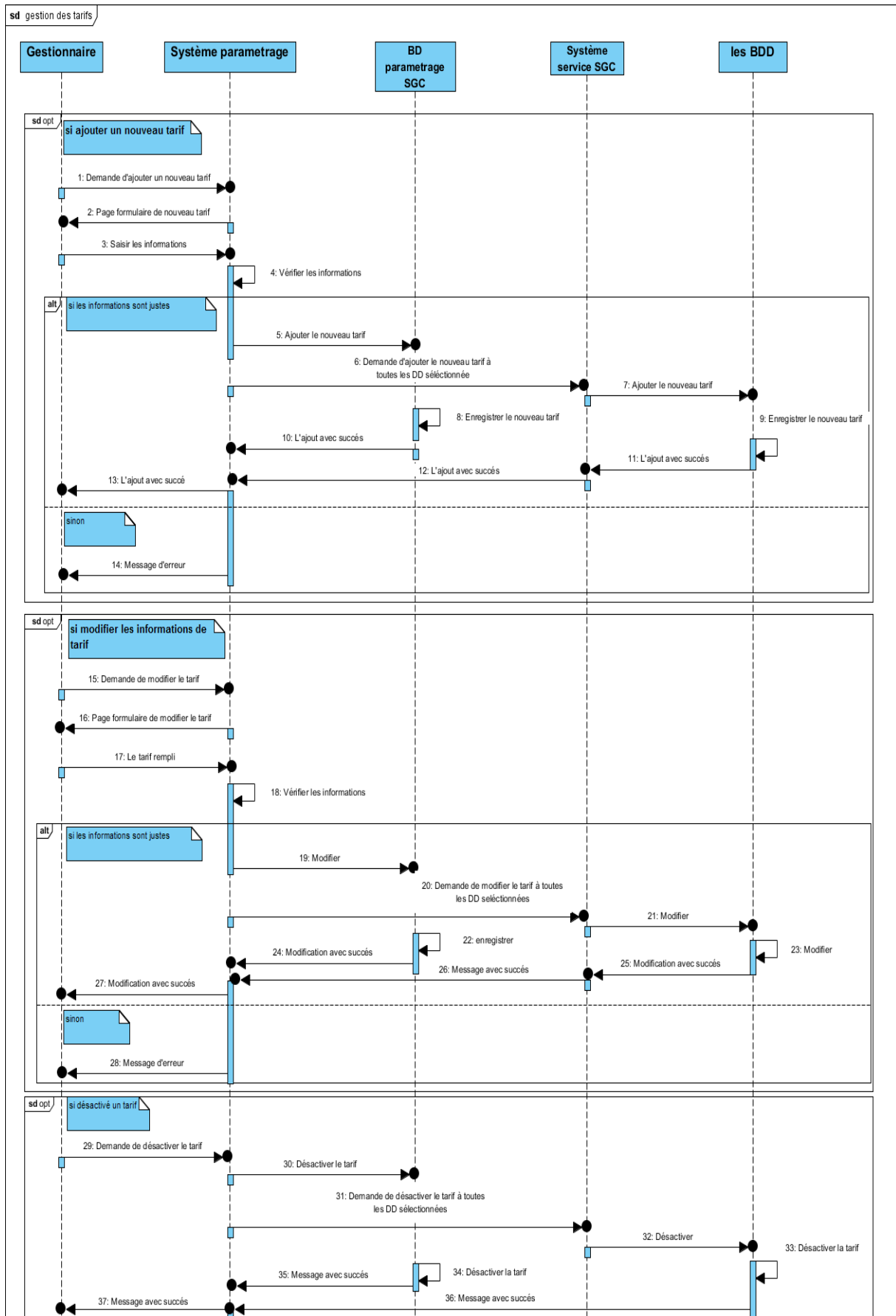


Figure II-11: Diagramme de séquence « gestion des communes ».

**b. Diagramme de séquence « gestion des Tarifs »**



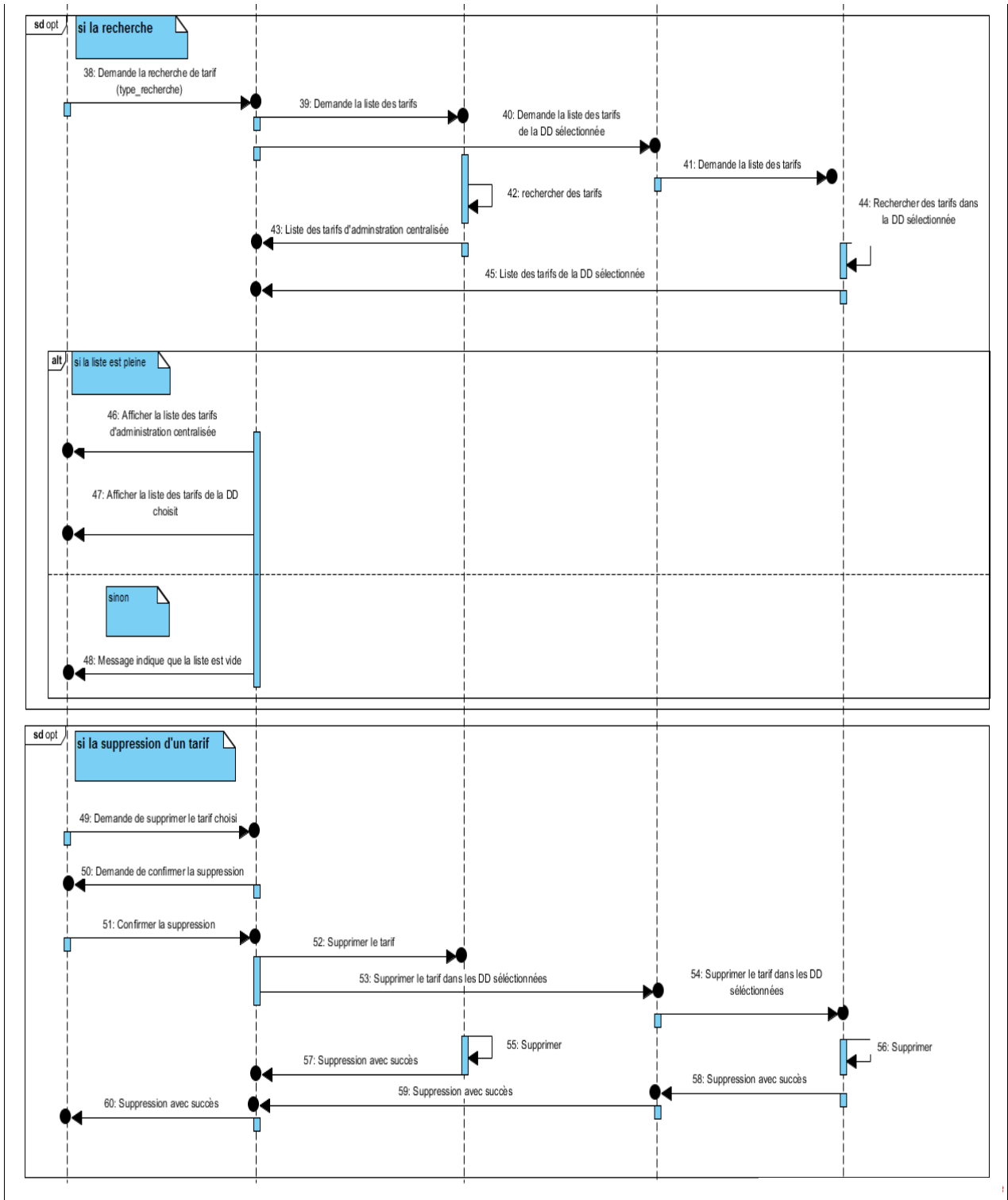
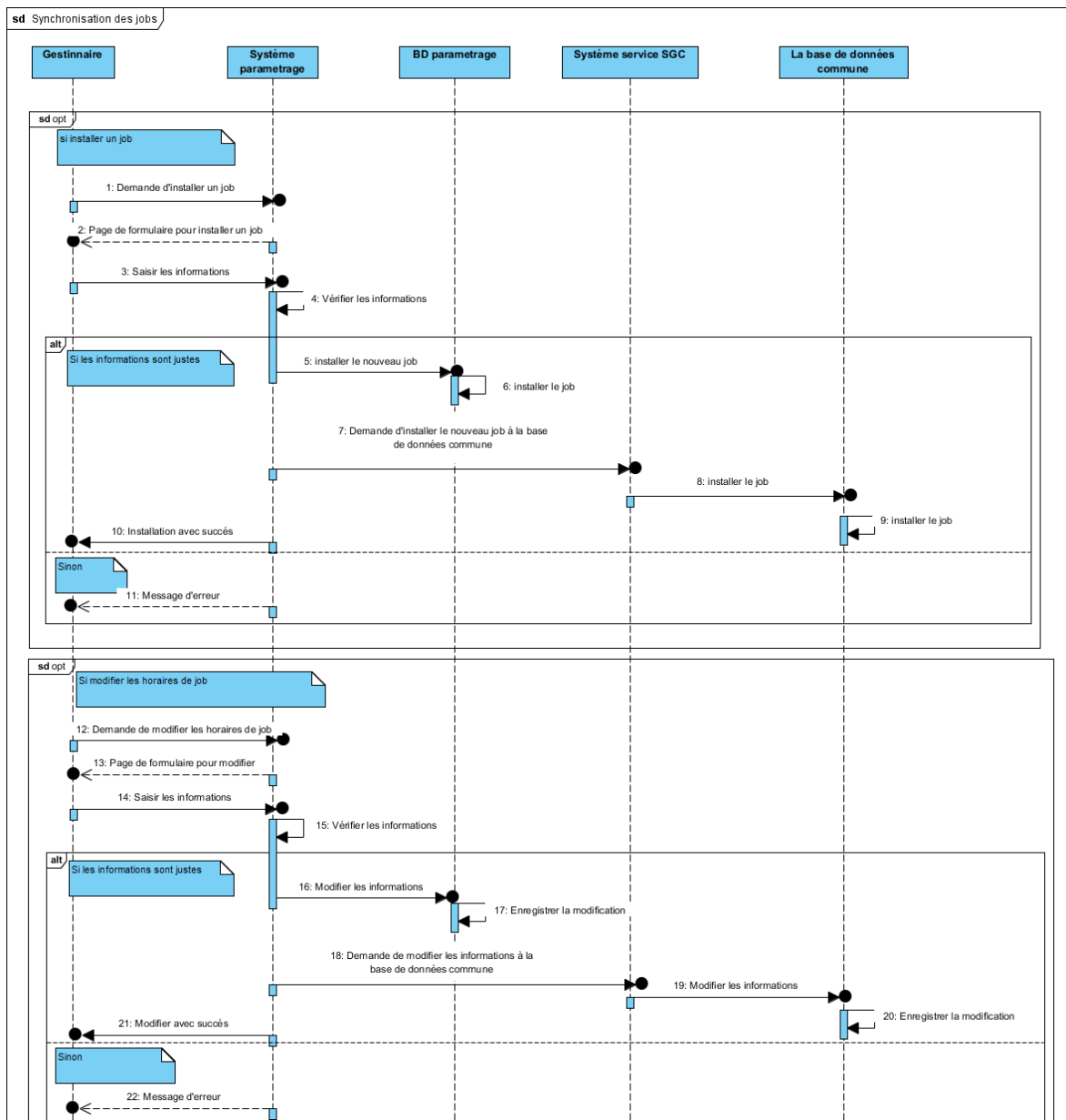
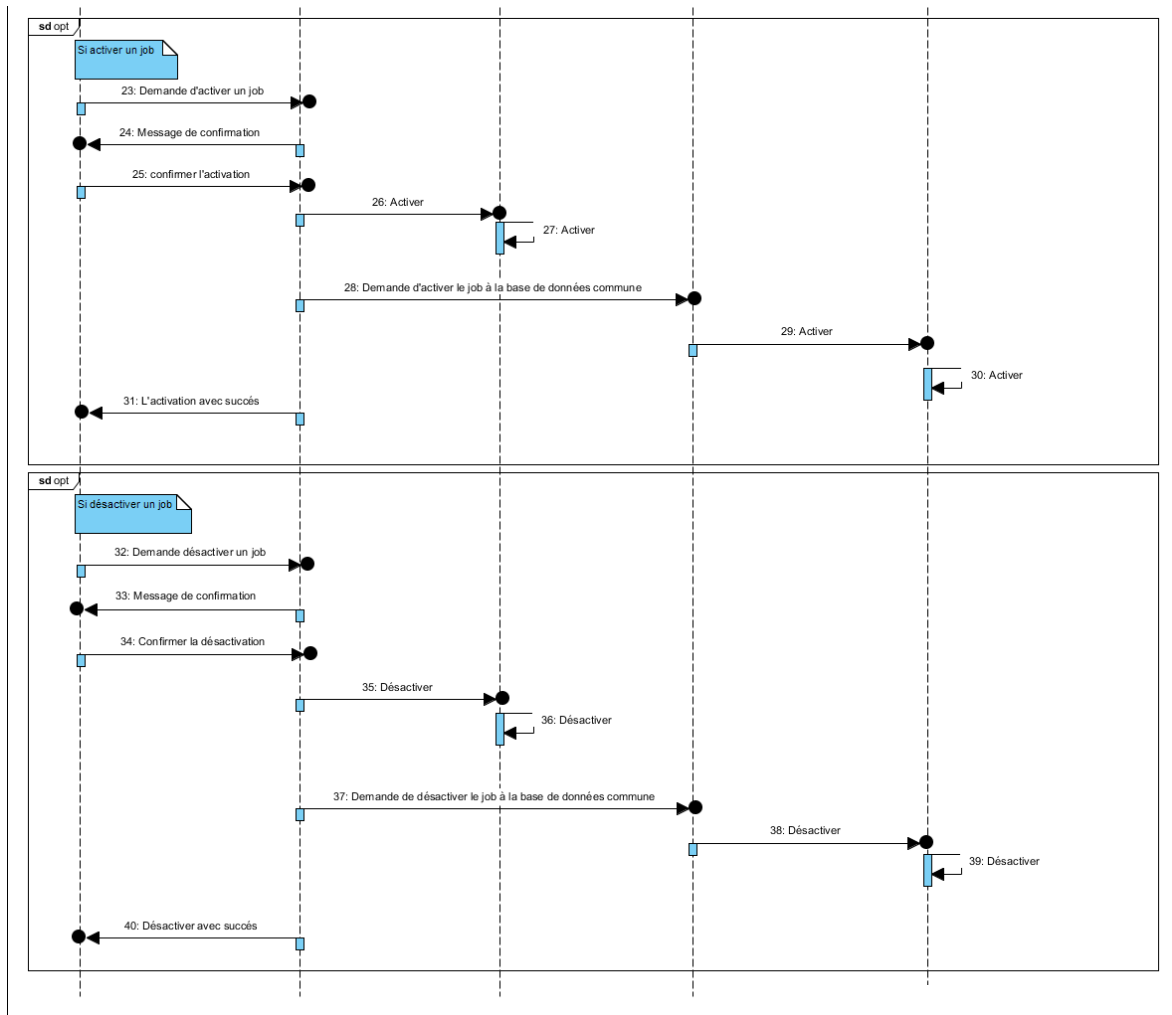


Figure II-12: Diagramme de séquence « gestion des tarifs».

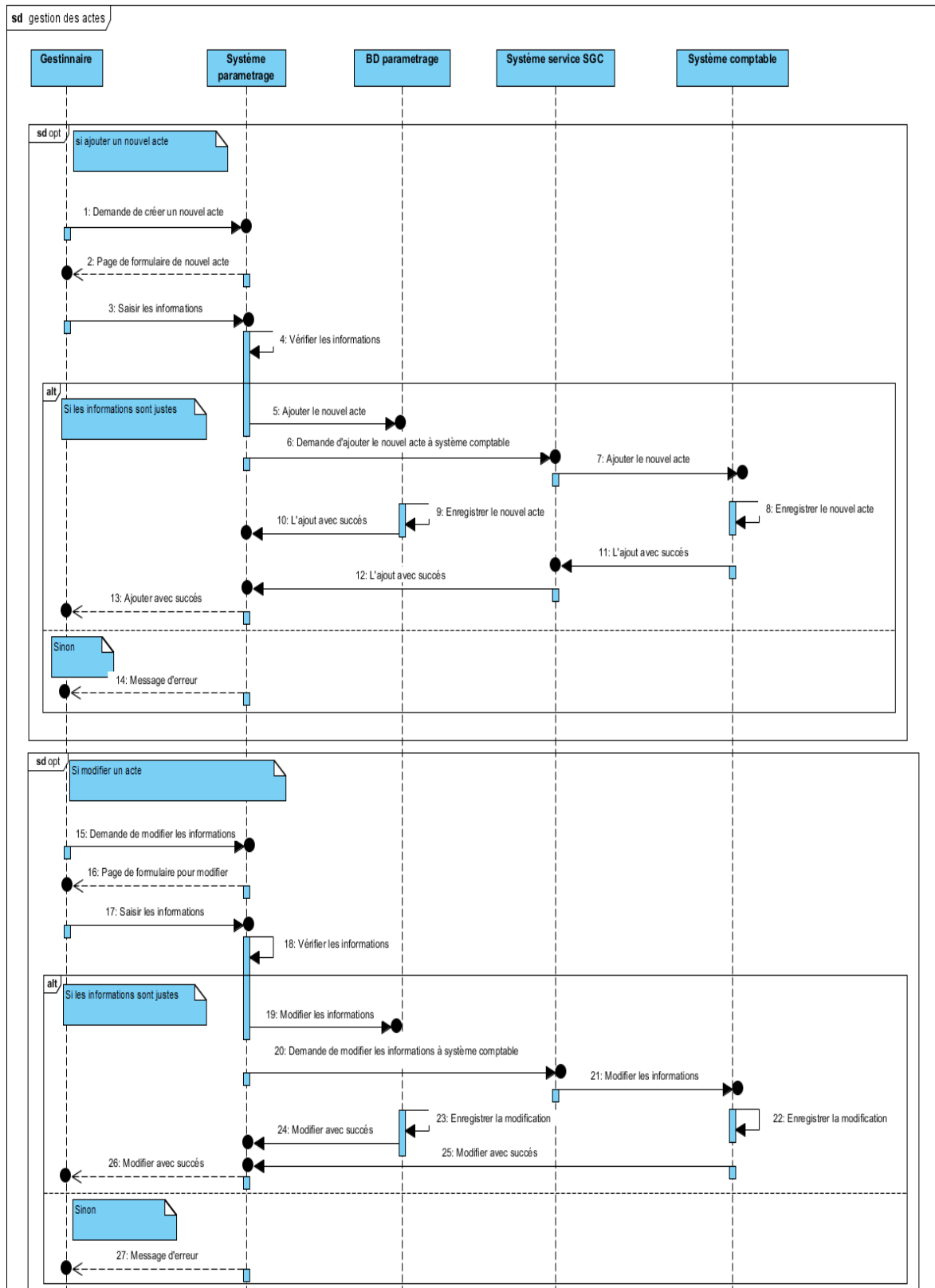
## B. Diagramme de séquence du cas d'utilisation « Gestion des jobs de synchronisation »





**Figure II-13: Diagramme de séquence du cas d'utilisation « centralisation de synchronisation des jobs de synchronisation ».**

C. Diagramme de séquence du cas d'utilisation « Gestion des actes »



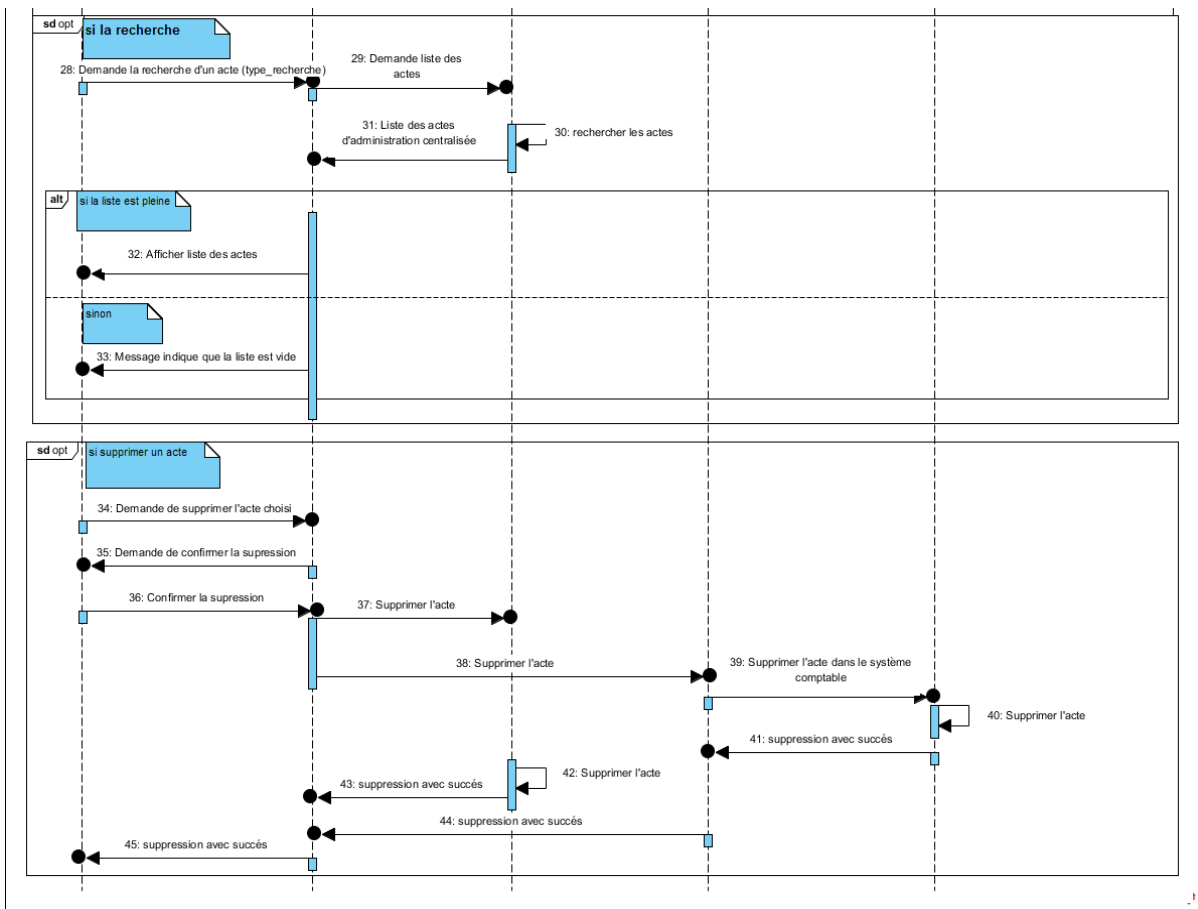


Figure II-14: Diagramme de séquence du cas d'utilisation « Gestion des actes».

### II.2.6. Spécification des besoins non fonctionnels

Après avoir déterminé les besoins fonctionnels, nous présentons ci-dessous l'ensemble des contraintes à respecter pour assurer la performance du système tout en respectant les exigences de l'utilisateur.

L'ensemble des extensions à réaliser doivent respecter les besoins suivants :

#### a. La Sécurité

- **Authentification :** l'accès à notre application requiert une authentification pour accorder une session au administrateur en gardant tous les informations être secrètes et sécurisés pour l'administrateur.
- **Gestion des rôles et droits d'accès:** le premier rôle est l'administrateur fonctionnel niveau siège du système qui a l'accès total aux différentes fonctionnalités et configurations de la plateforme. Pour le deuxième rôle est l'administrateur fonctionnel niveau région qui a l'accès à la consultation seulement de la plateforme.

### **b. La rapidité de traitement**

En effet, vu 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.

### **c. La performance**

Une plateforme de service doit être avant tout performante c'est-à-dire à travers ses fonctionnalités, répond à toutes les exigences des usagers d'une manière optimale.

### **d. La convivialité**

La future plateforme doit être facile à utiliser. En effet, les interfaces doivent être conviviales c'est-à-dire simples, ergonomiques et adaptées à l'utilisateur.

### **e. L'intégrité**

- La capture des erreurs d'entrée/sortie.
- Comment traiter les échecs d'interface de système, etc.
- Intégrité des données.
- Intégrité référentielle dans les tables de base de données et interfaces.

## **Conclusion**

Durant notre étude, nous avons consulté le cahier des charges du projet du reporting commercial « Refonte SGC ». Ce dernier nous a permis de mieux comprendre les fonctionnalités du SGC et avoir une idée sur les rapports demandés. La consultation des sites web du groupe SONELGAZ nous a permis de récupérer les organigrammes, comprendre l'organisation du groupe SONELGAZ et ses activités métiers ainsi que la SD.

La partie d'analyse a permis de livrer une spécification complète des besoins issus du diagramme de cas d'utilisation et séquence, de les structurer sous une forme qui facilite la compréhension et qui ébauche vers l'activité de conception. Le chapitre suivant définit la conception de notre système.



## *Chapitre III : Conception*

## **Introduction**

Comme dans tout projet, la phase d'analyse des besoins est suivie de celle de la conception.

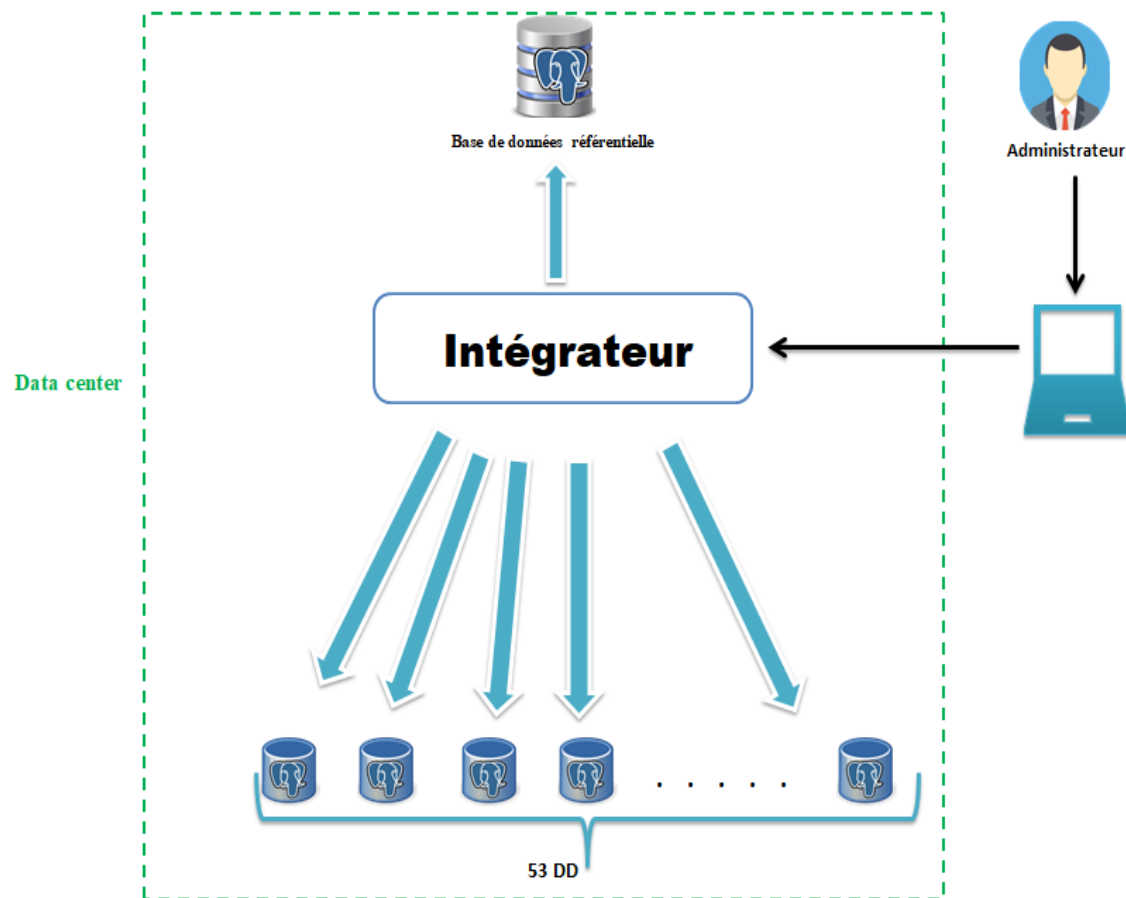
La modélisation d'un système avant sa réalisation, permet de mieux comprendre son fonctionnement. Alors que dans ce chapitre, et dans un premier lieu, on présentera l'architecture de notre solution de manière générale. Dans un deuxième lieu on détaille notre solution munie de certains diagrammes UML.

### **III.1. L'architecture proposée**

L'architecture que nous venons de présenter, a été pensée pour pallier aux limites de l'ancienne architecture du système SGC. Nous nous sommes basés sur une architecture orientée services pour limiter le problème d'interopérabilité entre le SGC et les autres solutions métiers et une autre architecture d'intégration pour éliminer le problème de centralisation des fonctionnalités communes entre les DD. La migration d'un système existant vers une architecture orientée services est très différente de la construction d'un nouveau système.

En effet, pour implémenter cette application qui répond à ces besoins nous avons deux schémas, le premier schéma de la base de données répartie ou nous avons adopté l'architecture qui est à l'intérieur du data center et qui montre dans la figure III-1.

Il nous est apparu important de prendre les fonctionnalités communes du niveau DD et on les centralise au niveau SD, c'est la centralisation de l'administration fonctionnelle du système de gestion de la clientèle c'est ce qui nous a permis d'accomplir un système d'intégration avec un entrepôt de donnée pour des fins d'intégration et non pas décisionnels.



**Figure III-1: L'architecture globale du système.**

Et le deuxième schéma qui représente l'intégrateur de notre système qui montre dans la figure III-2, l'idée qui est derrière notre intégrateur consiste à isoler chaque fonctionnalité ou un groupe de fonctionnalités dans un service, ce qui permet de distribuer plus librement la logique applicative pour chaque direction de distribution.

Notre solution devient alors un système constitué d'un certain nombre de petites unités indépendantes qui collaborent entre elles. Chacune ayant un cycle de vie et une responsabilité propre, est appelée un service.

Nous avons développé deux applications une application « paramétrage » et une autre « ServiceSGC ».

L'application « paramétrage » est une application consommateur de service (comme application cliente), dont l'objectif est de gérer notre fonctionnalité et les stocker dans une base de données référentielle.

L'application « ServiceSGC » est une application fournisseur de services est développé pour fournir les services de gestion de paramétrage et les stocker dans les différentes directions de distribution.

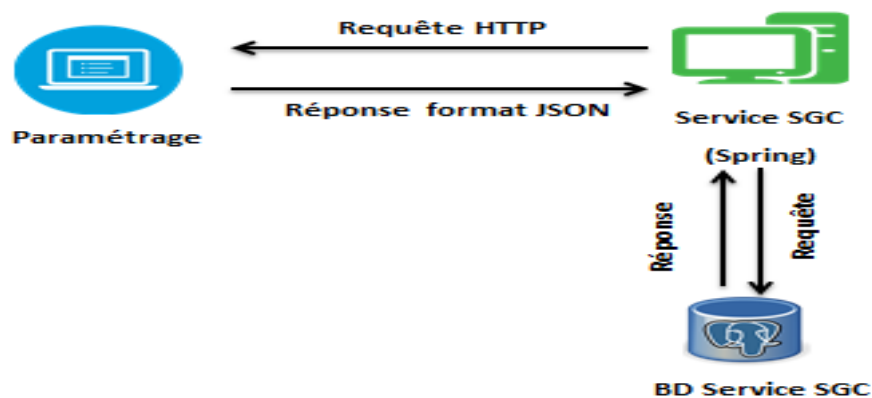


Figure III-2:Le schéma d'intégrateur.

## III.2. Conception partie service web

Nous avons implémenté plusieurs services web pour notre système afin d'effectuer les traitements sur les 53 bases de données de direction de distribution alors nous avons présenté nos service web par le diagramme de séquence et le diagramme de classe.

### III.2.1. Diagramme de séquence pour les services web

Pour montrer la démarche de chaque service web de notre application, nous avons utilisé le diagramme de séquence.

#### A. Service web d'ajout :

Ce service web est utilisé pour ajouter une des fonctionnalités de paramétrage, il l'ajout dans tous les DD sélectionnées par l'utilisateur.

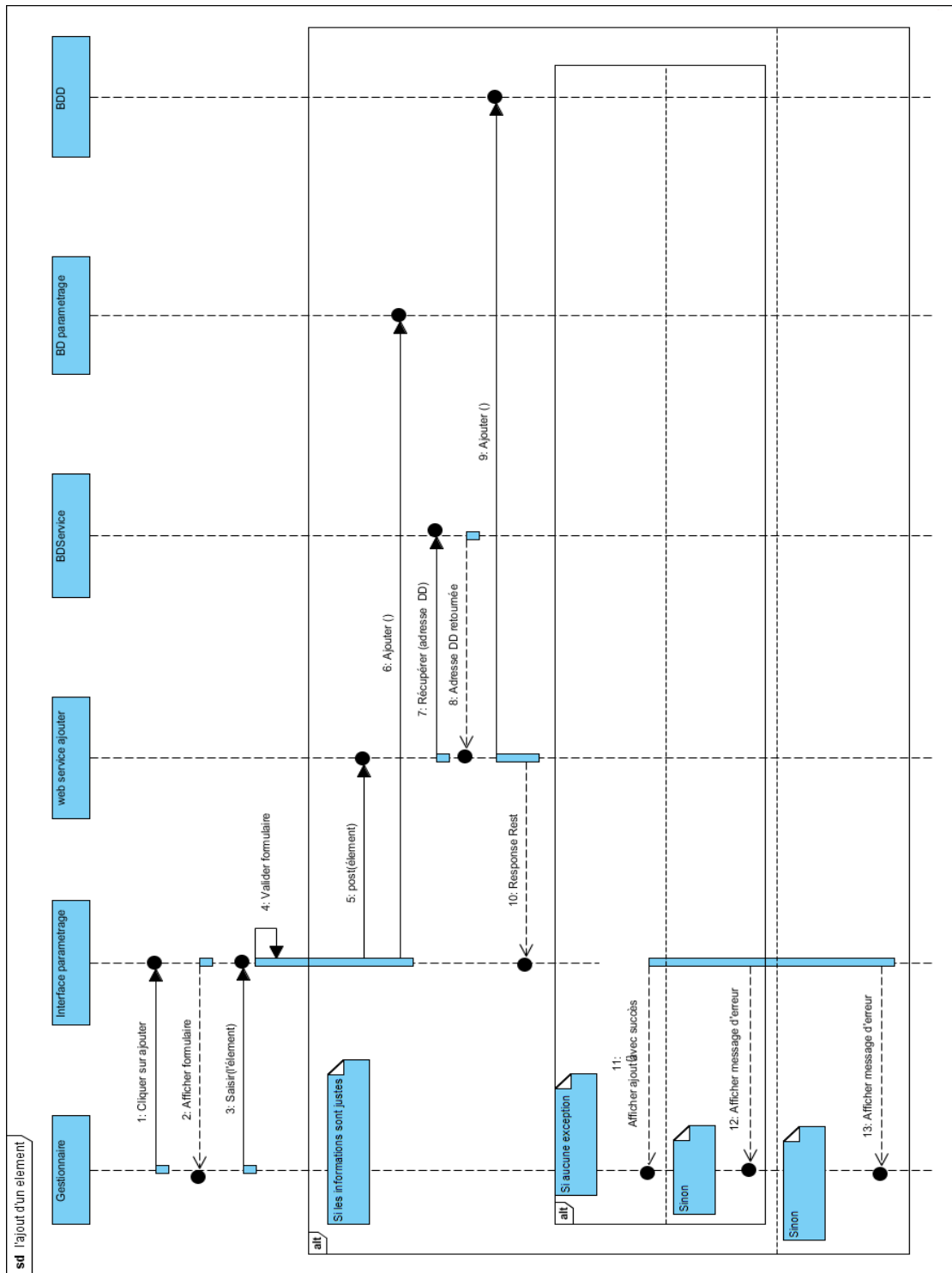


Figure III-3: Diagramme de séquence de service web d'ajout.

## B. Service web de modification

Ce service web est utilisé pour modifier des informations d'une des fonctionnalités de paramétrage, il la modifier dans tous les DD sélectionnés par l'utilisateur.

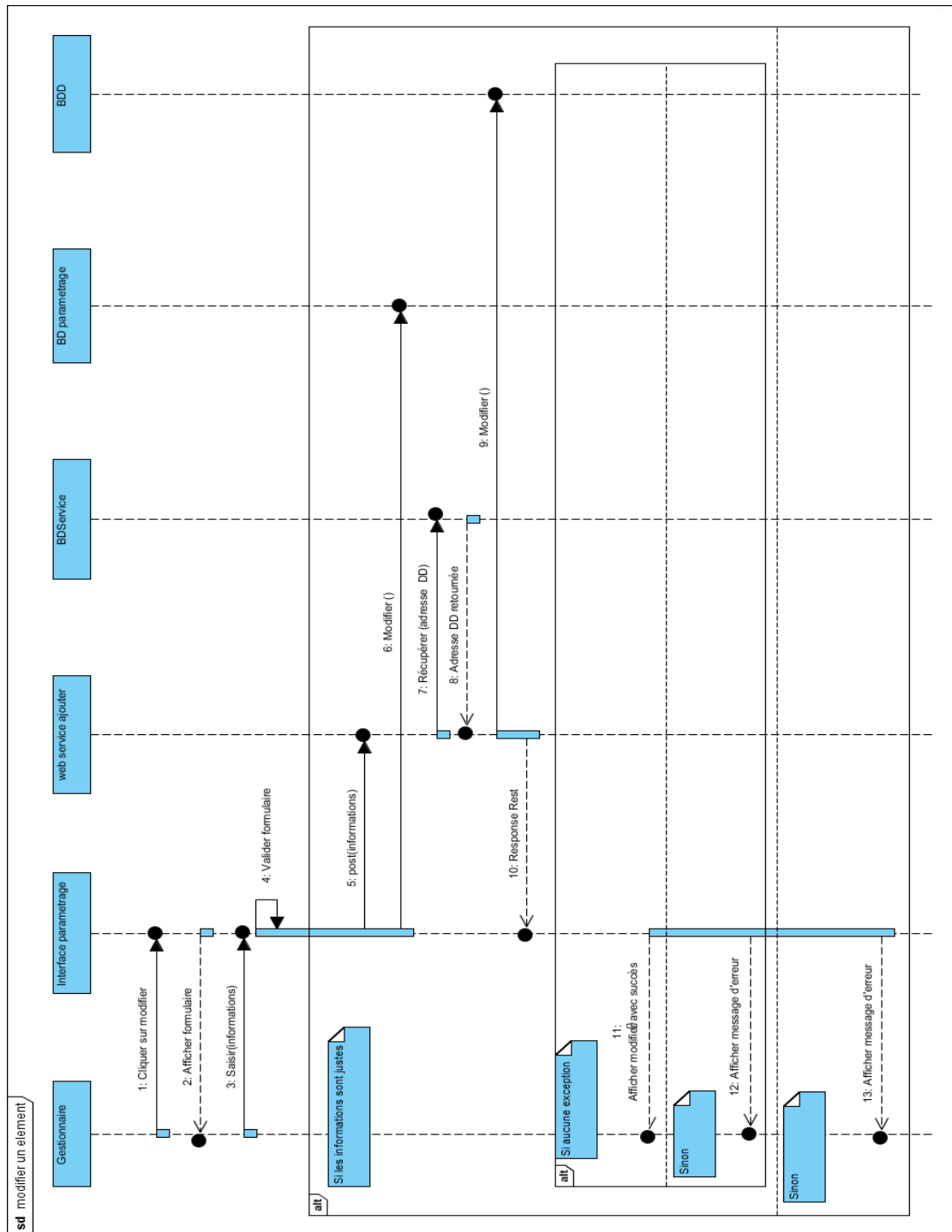


Figure III-4: Diagramme de séquence de service web de modification.

### C. Service web de Suppression

Ce service web est utilisé pour supprimer une des fonctionnalités de paramétrage, il la supprimer dans tous les DD sélectionnées par l'utilisateur.

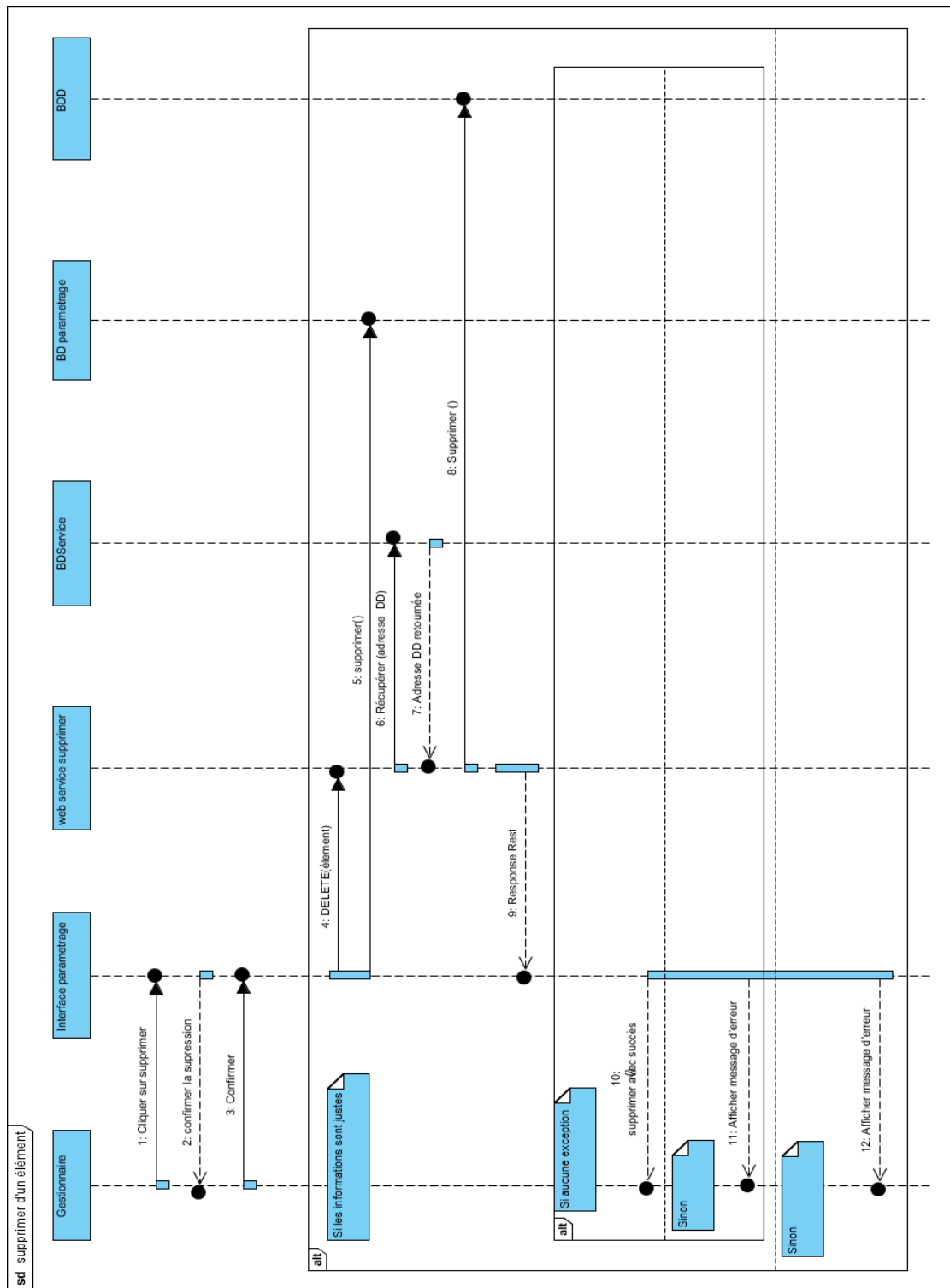


Figure III-5: Diagramme de séquence de service web de suppression.

### D. Service web de recherche

Ce service web est utilisé pour rechercher des informations d'une des fonctionnalités de paramétrage dans la DD sélectionnée par l'utilisateur.

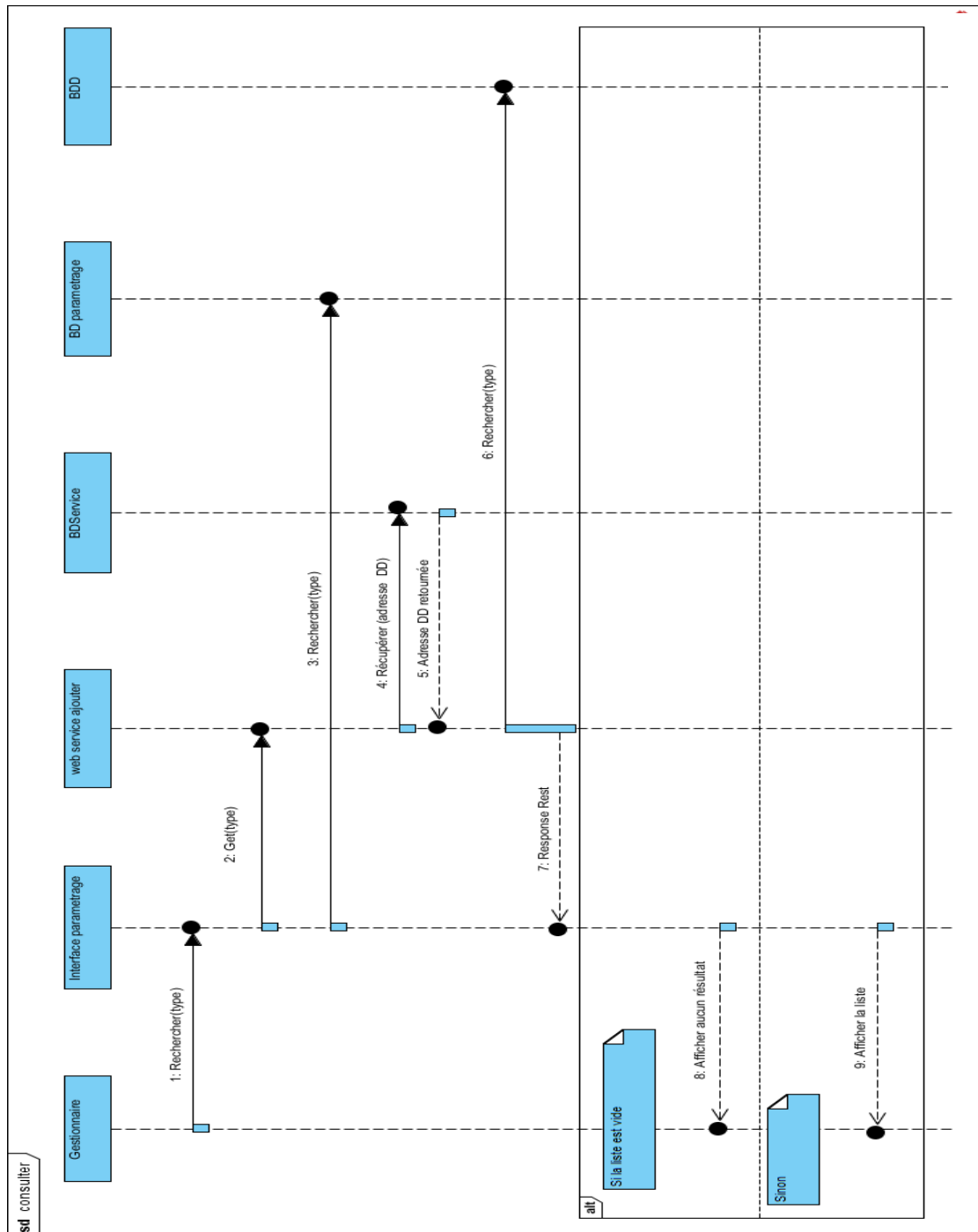


Figure III-6: Diagramme de séquence de service web de la recherche.



### **III.2.2. Diagramme de classe des services web**

Nous avons proposé ce diagramme de classe pour monter le fonctionnement de nos services web.

#### **Scenario narratif :**

Ce service web est utilisé pour ajouter, modifier, supprimer ou rechercher une activité économique. Ce service web est composé d'une entité activité économique qu'il même est composé de deux interfaces ou on définit toutes les fonctions qu'on veut utiliser et une configuration de la base de données BDService et une autre pour les DD.il comporte aussi un contrôleur ou on définit les API.

Afin de gérer l'activité économique dans toutes les DD à travers de récupérer les URL des DD à partir de « paramConnexion ».

Pour les autres gestions c'est le même scénario.

#### **Représentation UML :**

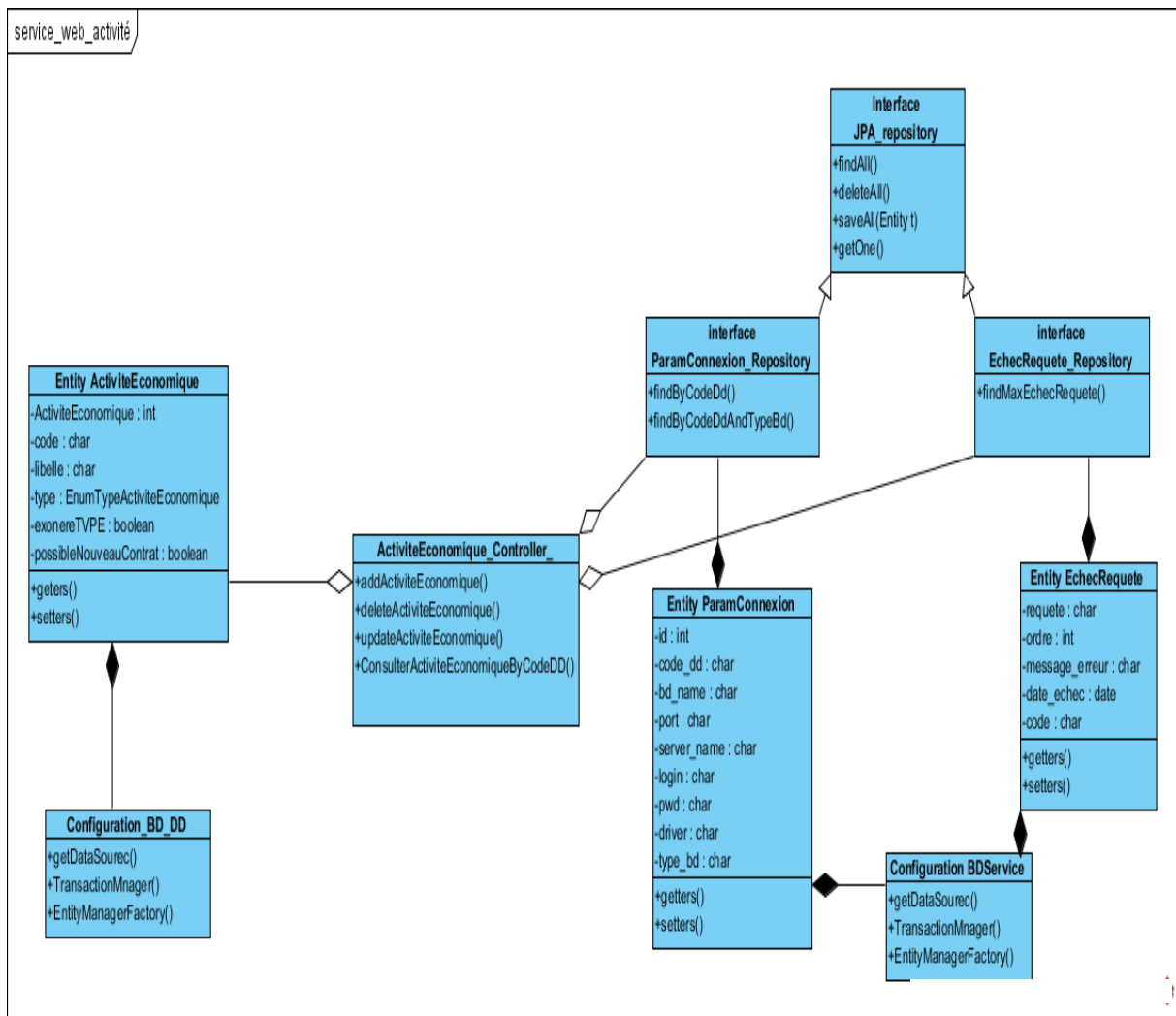


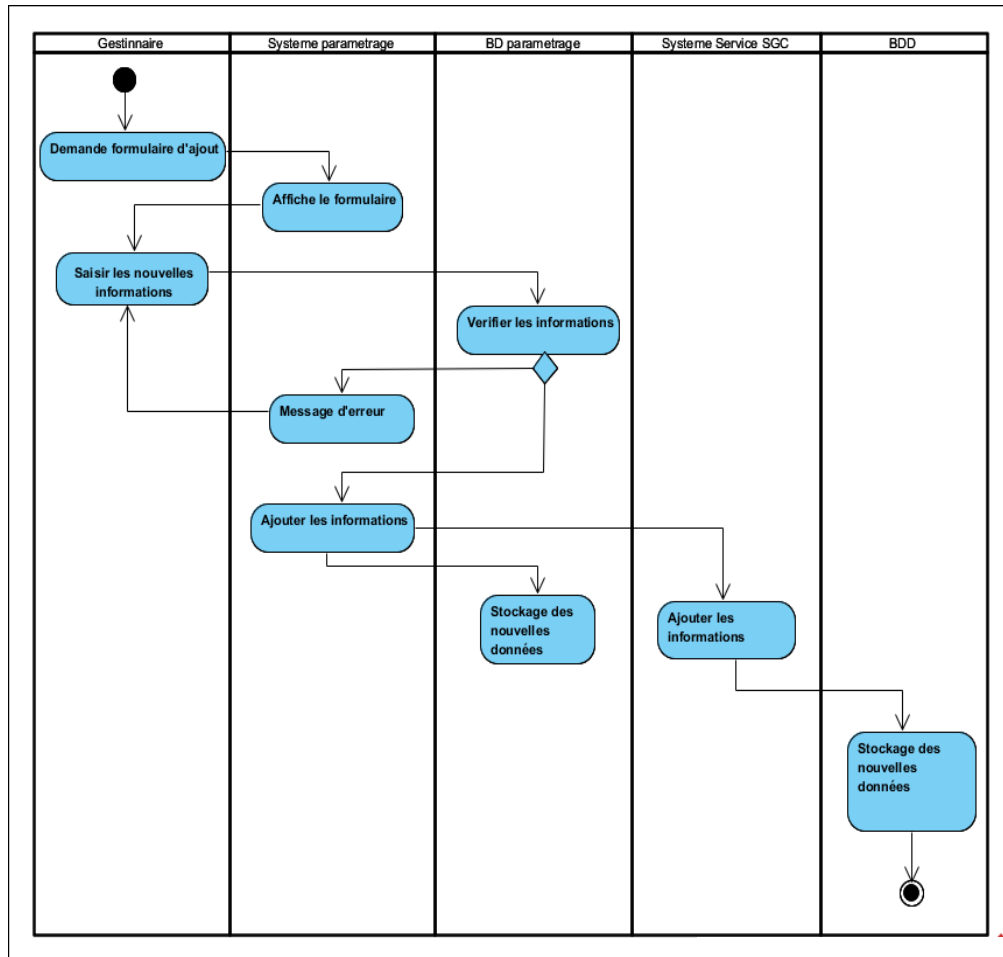
Figure III-7: Diagramme de classe « Service web Activité économique ».

### III.3. Conception détaillé

#### III.3.1. Diagramme d'activité

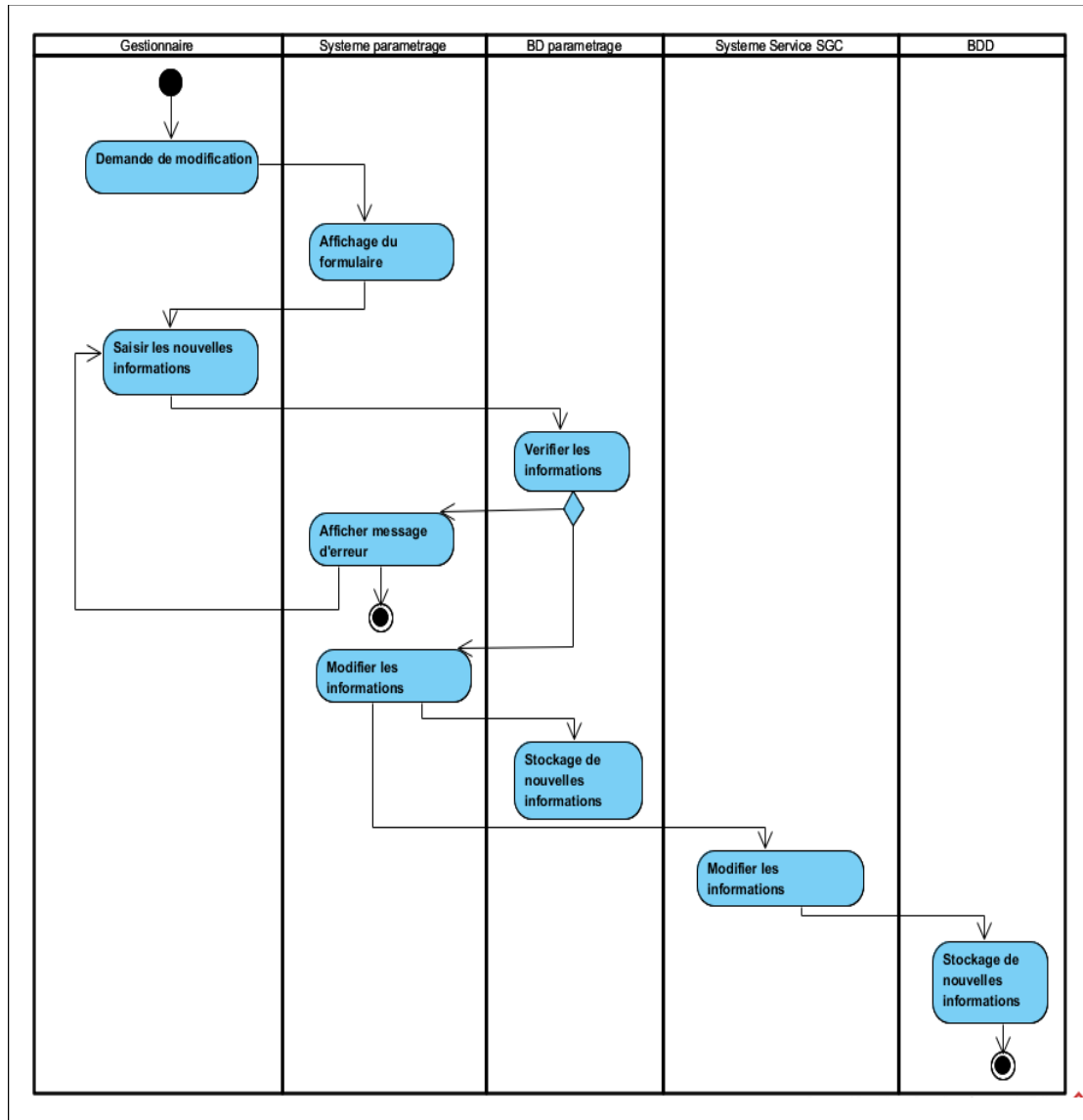
Les diagrammes d'activité sont des représentations graphiques des flux de travail d'activités et d'actions par étapes, se sont conçus pour modéliser les processus de calcul et les processus d'organisation [26]. On va voir dans ce sens les diagrammes de quatre processus importants à savoir : le processus d'ajout, suppression, modification, et la recherche, qui se répètent dans chaque gestion de notre système paramétrage.

**A. Diagramme d'activité d'ajout**



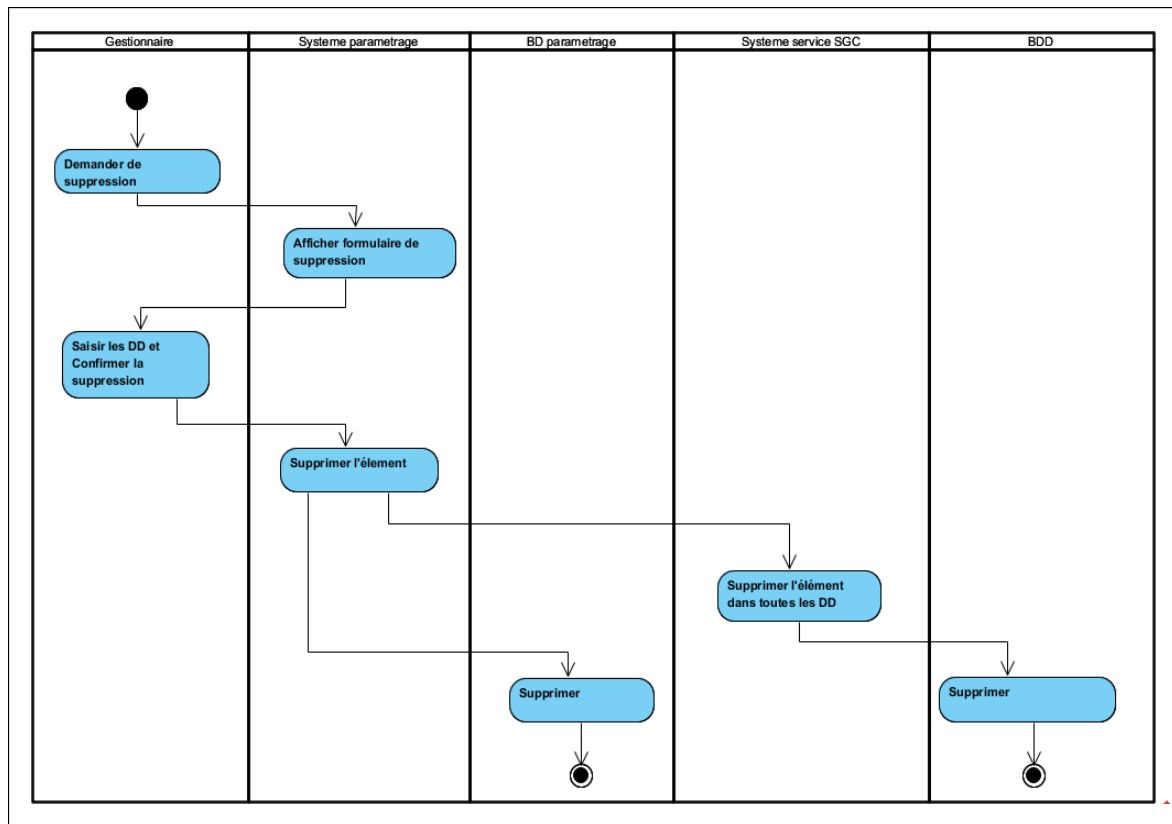
**Figure III-8: Diagramme d'activité d'ajout.**

**B. Diagramme d'activité de modification**



**Figure III-9: Diagramme d'activité de modification.**

**C. Diagramme d'activité de suppression**



**Figure III-10: Diagramme d'activité de suppression.**

**D. Diagramme d'activité de la recherche**

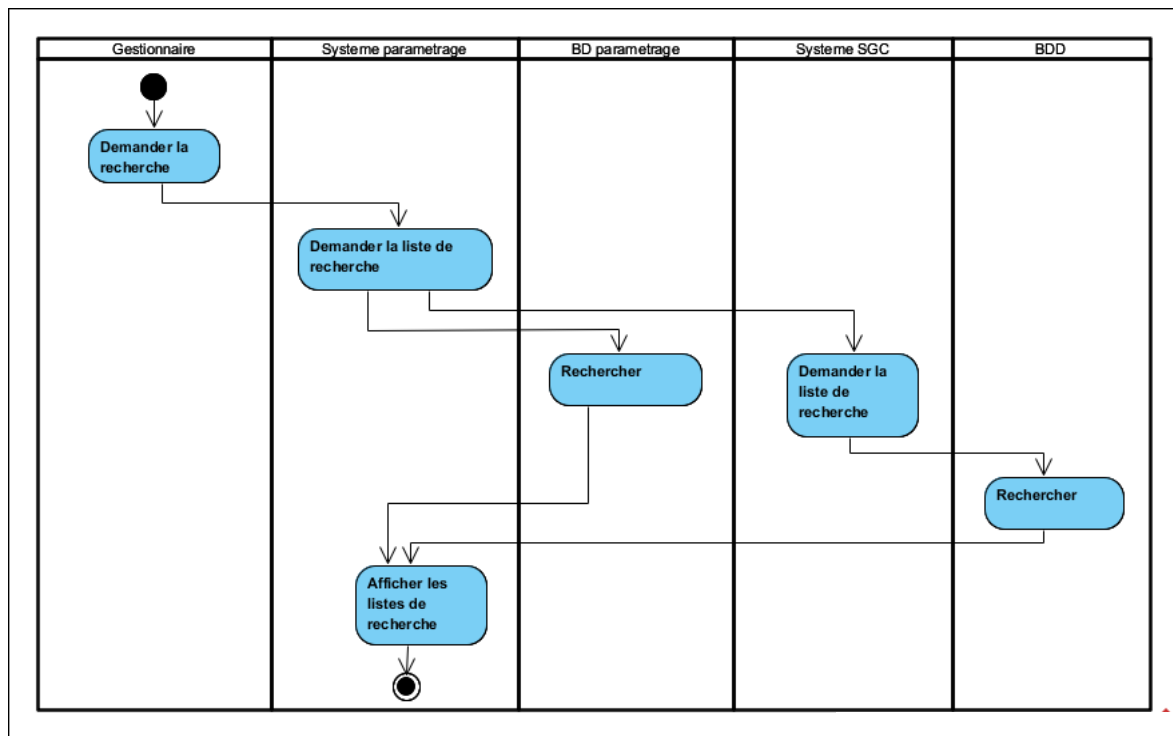
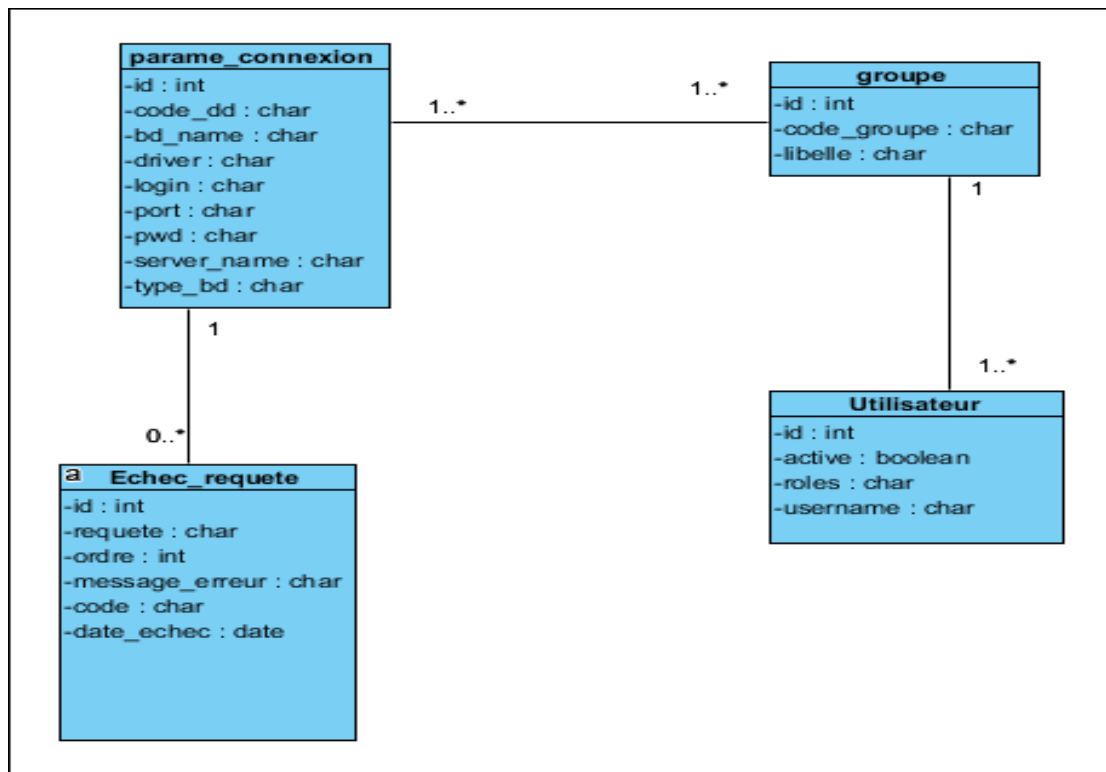


Figure III-11: Diagramme d'activité de la recherche.

### III.3.2. Diagramme de classe

Maintenant on s'intéresse à comment les fonctionnalités du système seront implémentés en reposant sur le diagramme de classe qui est le concept fondamental dans l'orienté objet qui vise à représenter les aspects métier, technique et les entités du système. La figure suivante montre le diagramme de classes d'application serviceSGC.



**Figure III-12: Diagramme de classe de serviceSGC.**

La base de données du SGC contient environ 1000 tables, donc pour obtenir une base de données centralisée, nous avons fait un nettoyage sur cette base qui est montrent ci-dessous :

- 1- Nous avons fait un aperçu sur la base de données SGC.
  - 2- Nous avons déterminé toutes les tables concernant l'administration fonctionnelle pour les centraliser.
  - 3- Nous avons éliminé les autres tables qui n'ont rien à voir avec l'administration fonctionnelle.
  - 4- Enfin, pour obtenir un schéma homogène, nous avons lié les tables entre elles.
- Donc, le diagramme de classe qui montre dans la figure III-12 représente notre résultat.

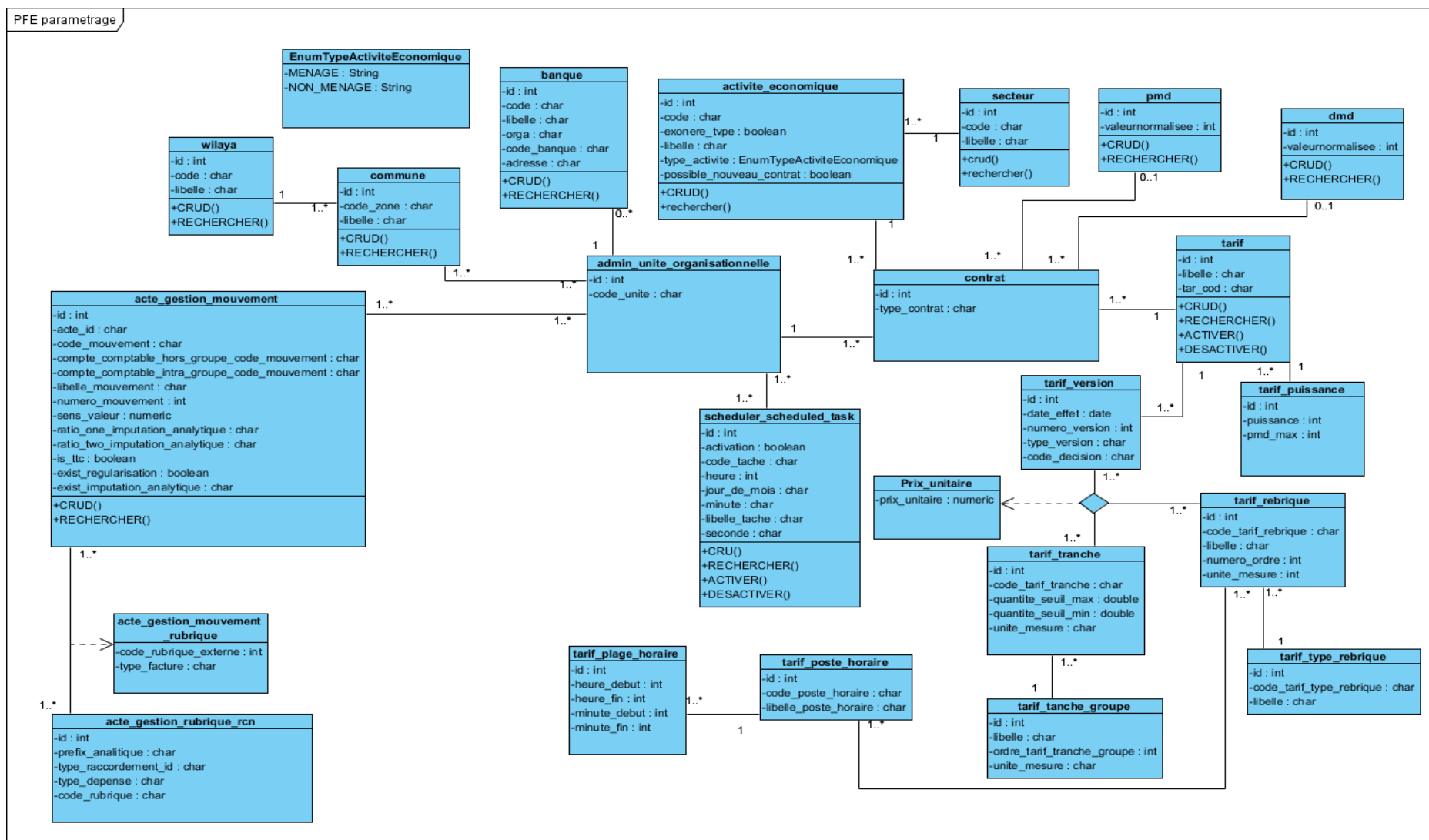


Figure III-13: Diagramme de classe de paramétrage.



### III.3.3. Modèles Logiques de Données

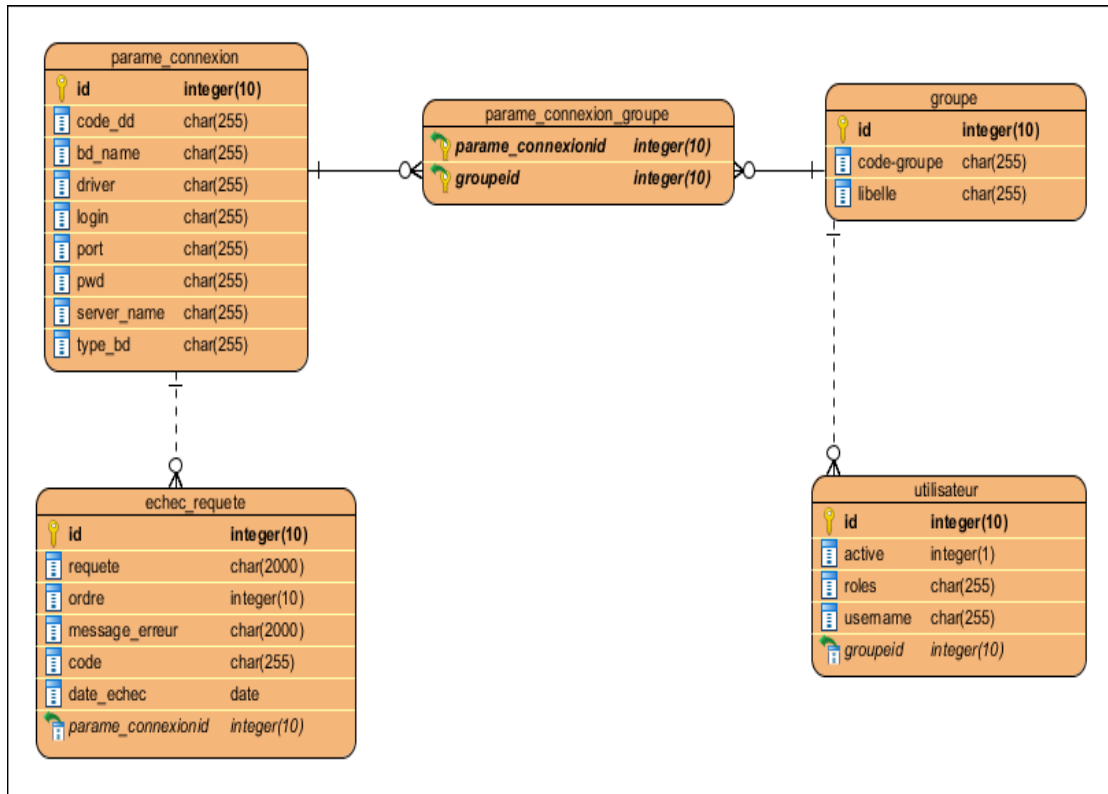
En appliquant les règles de passages à nos digrammes de classe, nous obtenons le schéma relationnel de parametrage (voir l'annexe B) et le schéma relationnel de serviceSGC suivant :

Les tables	Les champs	Observation
param_connexion	<u>Id</u> code_dd bd_name driver login port pwd server_name type_bd	-Identifiant de paramétrage de connexion -Le code de la direction de distribution -Le nom de la base de données de la direction -Le driver de la base de données -Le login pour la base de données -Le port de la base de données -Le mot de passe -Le nom de serveur -Le type de la base de données
Groupe	<u>Id</u> code_groupe libelle	-Identifiant de groupe -Code de groupe -Nom de groupe
param_connexion_groupe	<u>#param_connexion</u> <u>nid</u> <u>#groupeid</u>	-Identifiant de paramétrage e connexion -Identifiant de groupe
Utilisateur	<u>Id</u> active rôles username #groupeid	-Identifiant d'utilisateur -S'il a l'autorisation d'accès - le rôle de l'utilisateur -Le nom d'utilisateur -Identifiant de groupe
Echec_requete	<u>Id</u> requete ordre message_erreur code date_echec	-Identifiant de l'échec de la requête -Description de la requête -L'ordre de la requête par DD -Description de message d'erreur -Le code de module -La date de la requête

	#param_connexion nid	-Identifiant de paramétrage de connexion
--	-------------------------	--

**Tableau 6: Le modèle relationnel de serviceSGC.**

Les figures suivantes montrent le modèle logique de données de notre système :



**Figure III-14: Modèle logique de données de ServiceSGC.**



### III.3.4. Diagramme de composant

Le diagramme de composants sert à décrire l'organisation du système du point de vue des éléments logiciels (modules, données, paramètres...). Ce diagramme permet de mettre en évidence les dépendances entre les composants (qui utilisent quoi) [26]. L'illustration suivante présente le diagramme de composants du système :

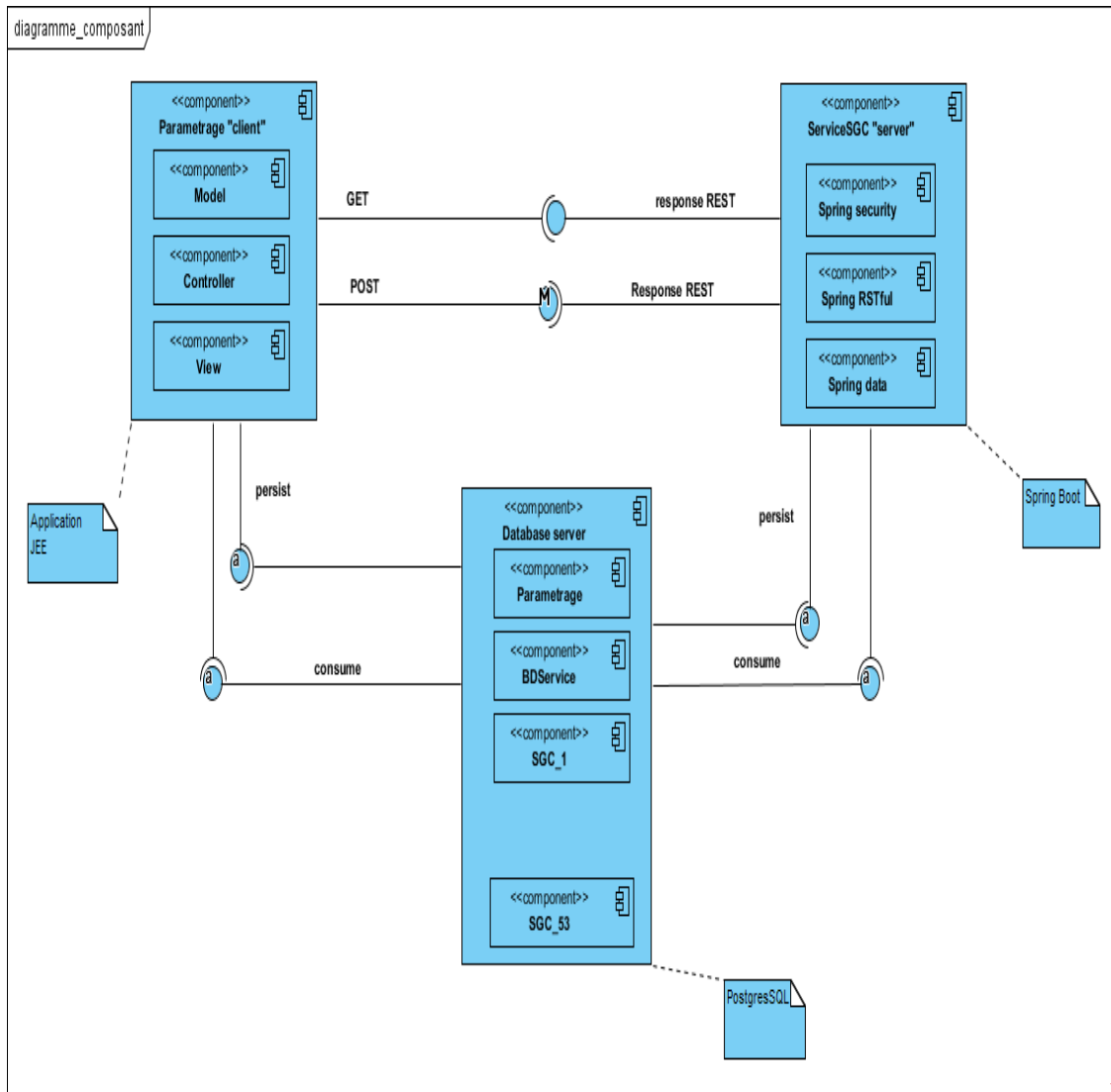


Figure III-16: Diagramme de composant du système.

## **Conclusion**

Dans ce chapitre, tout d'abord nous avons présenté l'architecture globale de notre système après nous avons détaillé le schéma d'intégrateur, tout en expliquant les différents composants de celle-ci. Ensuite, nous avons détaillé l'étude technique et conceptuelle qu'on a réalisée et suivie le long de la conception de notre architecture solution, et ça en utilisant le langage UML et des schémas explicatifs détaillés.

## *Chapitre IV : Implémentation*

## Introduction

Après avoir présenté l'architecture de notre système dans le chapitre précédent, Nous allons maintenant introduire les outils utilisés dans la réalisation de notre système et les environnements de développement, ainsi nous allons présenter les différentes interfaces pour expliquer le fonctionnement des fonctionnalités de ce système.

### IV.1. Architecture MVC

Il est primordiale à la réalisation de tout système informatique de choisir le modèle d'architecture qui lui sera adéquat pouvant assurer un bon fonctionnement, des meilleurs performances ainsi que la réutilisation et l'interconnexion fiable de ce système avec d'autres.

C'est à cet effet que nous optons pour le modèle MVC qui sera également très pratique pour gérer l'interaction entre les différents composants de notre application paramétrage.

L'architecture MVC (modèle, vue et contrôleur) est une architecture à trois couches utilisée pour la programmation client/serveur et d'interface graphique. [28]

C'est un modèle architectural très puissant qui intervient dans la réalisation d'une application. Il tire sa puissance de son concept de base qui est la séparation des données (modèle), de l'affichage (vue) et des actions (contrôleur). [28]

C'est trois couches sont décrites comme suit [28]:

- **Modèle** : il correspond aux données stockées généralement dans une base de données. Dans un langage orientée objet ces données sont exploitées sous forme de classes. Le modèle peut aussi agir sur la vue en mettant à jour ses données.
- **Vue** : ne contenant que les informations liées à l'affichage, la vue se contente d'afficher le contenu qu'elle reçoit sans avoir connaissance des données. En bref, c'est l'interface homme machine de l'application.
- **Contrôleur** : le contrôleur sert de base à récupérer les informations, de les traiter en fonction des paramètres demandés par la vue (par l'utilisateur), puis de renvoyer à la vue les données afin d'être affichées. C'est donc l'élément qui va utiliser les données pour les envoyer à la vue.

L'interaction entre ces trois couches est décrite à l'aide de la figure IV-1 suivante :

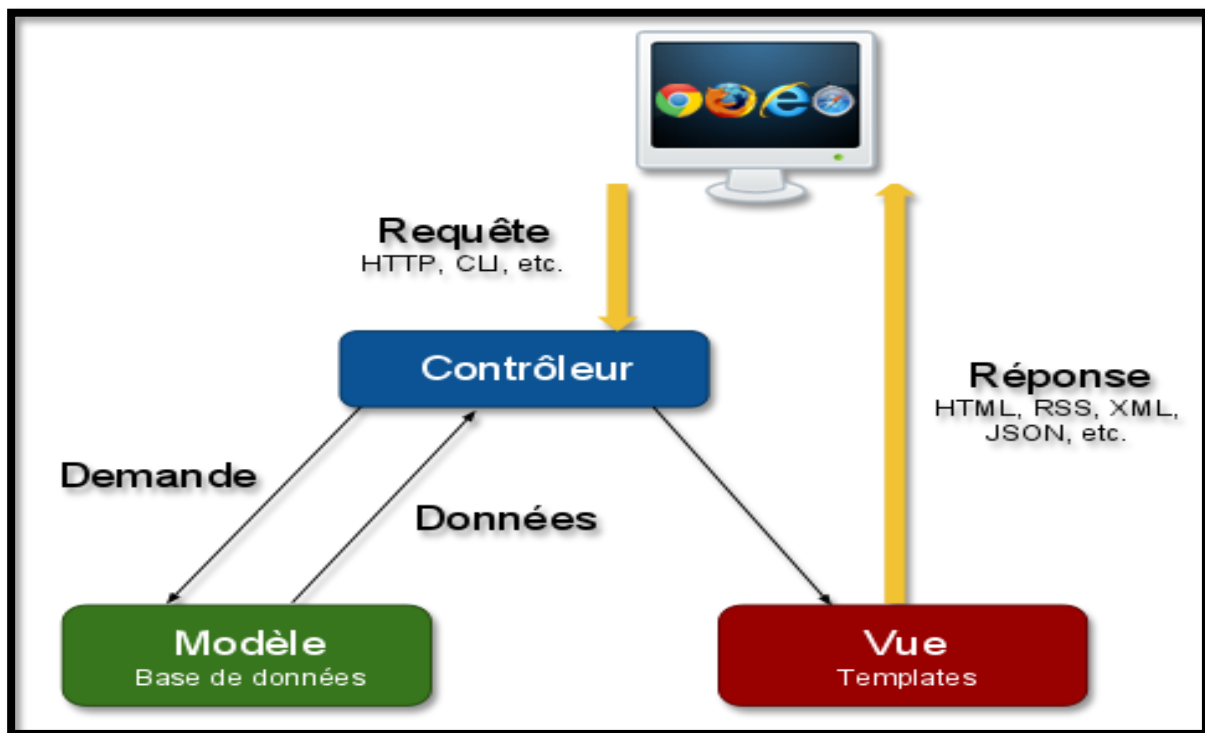


Figure IV-1: Architecture MVC. [28]

Les avantages apportés par l'architecture MVC sont [28] :

- La séparation des données de la vue et du contrôleur (ce qui permet une conception claire et efficace de l'application)
- Une indépendance des données, de l'affichage et des actions (ce qui donne plus de souplesse pour la maintenabilité et l'évolutivité du système).
- Un gain de temps de maintenance et d'évolution de l'application.

## IV.2. Les environnements et les outils utilisés

### IV.2.1. Framework de développement Spring Boot

L'application serveur a été développée sur la base du framework Spring Boot qui est un framework d'application et une inversion du conteneur de contrôle pour la plateforme JavaEE. Les principales fonctionnalités du framework peuvent être utilisées par n'importe quelle application Java. Il permet de créer des applications basées sur les microservices à base de REST. Spring Boot permet également de [29]:



- Faciliter le développement d'applications complexes.
- Faciliter à l'extrême l'injection des dépendances.
- Réduire à l'extrême les fichiers de configurations.
- Faciliter la gestion des dépendances Maven.
- Auto Configuration : la plupart des beans sont créés si le ou les jar(s) adéquats sont dans le classpath.
- Fournir un conteneur de servlet embarqué (Tomcat, Jetty)
- Créer une application autonome (jar ou war).



**Figure IV-IV-2: Logo Spring Boot. [29]**

#### **IV.2.2. Outils de développement « NetBeans »**

Est un environnement de développement intégré (IDE) pour Java, placé en open source par Sun en juin 2000 sous licence CDDL (Common Développement and Distribution License). Un outil pour les programmeurs pour écrire, compiler, déboguer et déployer des programmes. Il est écrit en Java - mais peut supporter n'importe quel langage de programmation. [30]



**Figure IVV-IV-3: Logo de Netbeans. [30]**

### IV.2.3. Serveur de base de données PostgreSQL

PostgreSQL est un système de gestion de base de données relationnelle et objet (SGBDRO). Ce système multiplateforme est largement connu et réputé à travers le monde, notamment pour son comportement stable. [31]

Il prend en charge les transactions, les sous-sélections, les déclencheurs, les vues, l'intégrité référentielle des clés étrangères et le verrouillage sophistiqué. [31]

Il fonctionne sur de nombreuses plates-formes, dont Linux, la plupart des versions d'UNIX, Mac OS X, Solaris, Tru64 et Windows. Il prend en charge aussi le texte, les images, les sons et la vidéo et comprend des interfaces de programmation pour C / C ++, Java, Perl, Python, Ruby et Open Database Connectivity (ODBC). [31]

C'est un système de gestion de base de données open source développé par une équipe mondiale de bénévoles. PostgreSQL n'est contrôlé par aucune société ou autre entité privée et le code source est disponible gratuitement. [31]



Figure IV-4: Logo PostgrsSQL. [31]

### IV.2.4. Plateforme de développement “Java EE”

Le choix de cette plateforme a été justement imposé par l'organisme, c'est d'ailleurs la plateforme de développement utilisée dans la majorité des projets de ELIT. Cette plateforme représente en fait un ensemble de technologies et d'APIs pour la plateforme Java conçus pour soutenir les applications d'entreprise. En général, il s'agit d'applications complexes, à grande échelle, distribuées, transactionnelles et hautement disponibles conçues pour répondre aux besoins d'affaires critiques. [32]



Figure IV-5: Logo « java EE ». [32]

#### IV.2.5. Outils de modélisation

Comme outil de modélisation UML nous avons utilisé l'outil professionnel Visual Paradigm de la version 13.0 qui est une solution complète du formalisme UML. [33]



Figure IV-6: Logo de « Visual Paradigm ». [33]

#### IV.2.6. Serveur de déploiement « Glassfish »

GlassFish est le nom du serveur d'applications Open Source Java EE 5 et désormais Java EE 7 avec la version 4.1 qui sert de socle au produit Oracle GlassFish Server (anciennement Sun Java System Application Server de Sun Microsystems). [34]

Sa partie Toplink persistance provient d'Oracle. C'est la réponse aux développeurs Java désireux d'accéder aux sources et de contribuer au développement des serveurs d'applications de nouvelle génération. [34]



Figure IV-7: Logo de « Glassfish ». [34]

### IV.3. Description de l'interface de l'application

Pour mieux comprendre le fonctionnement de l'application, les figures qui suivent décrivent la chronologie des principales fonctionnalités de l'application.

On an montrer une seule gestion c'est la gestion des activités économiques car les autres gestions sont les mêmes.

#### IV.3.1. Interface d'authentification

Tout d'abord une authentification est requise afin d'accéder au notre système.



**Figure IV-8: interface authentification.**

On a deux utilisateurs le premier est l'administrateur fonctionnel niveau siège du système qui a l'accès total aux différentes fonctionnalités et configurations de la plateforme, comme le montre la figureI IV-9 ci-dessous :

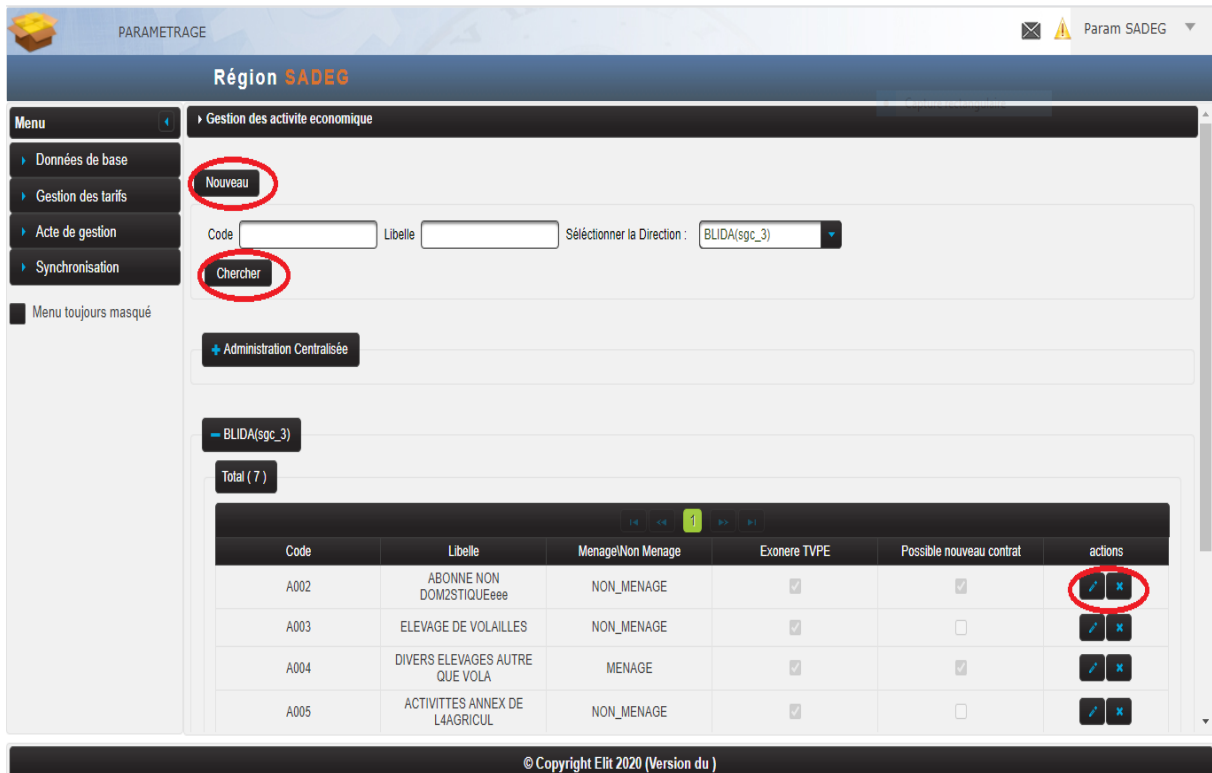


Figure IV-9: interface d'accueil d'administrateur fonctionnel niveau siège.

Pour le deuxième rôle est l'administrateur fonctionnel niveau région qui a l'accès à la consultation seulement de la plateforme, comme la montre la figure IV-10 ci-dessous :

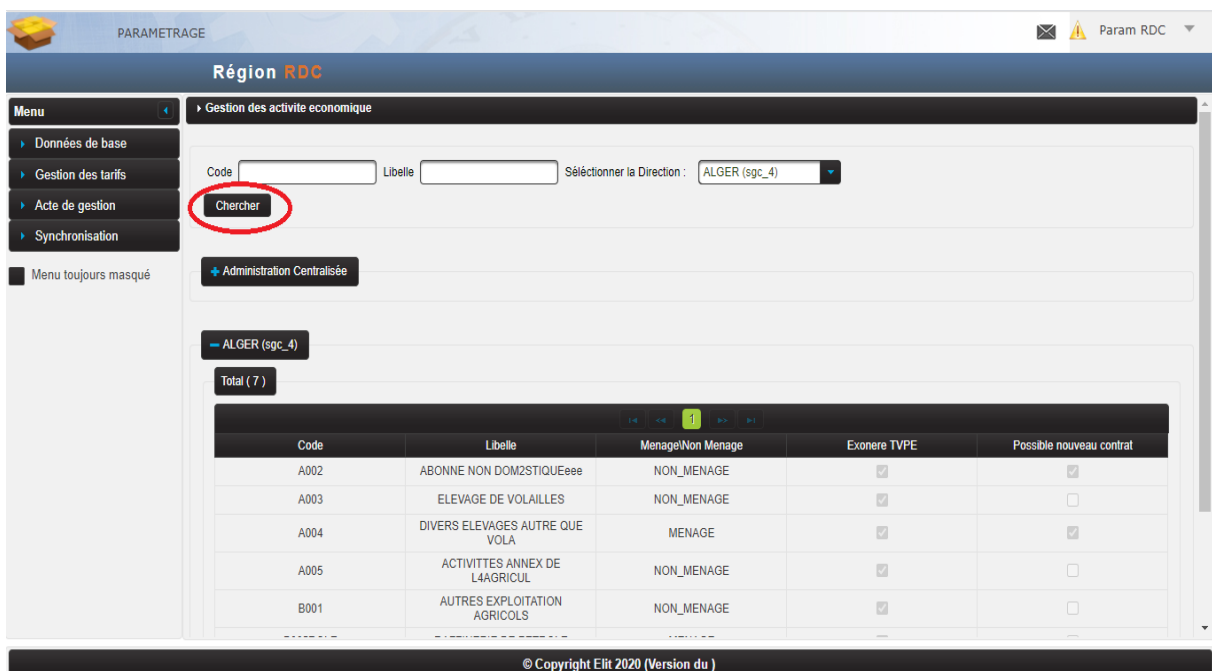


Figure IV-10: interface d'accueil d'administrateur fonctionnel niveau régionale.

### IV.3.2. Interface d'accueil

La figure IV-11 apparait une fois l'administrateur authentifié, permet de voir son accueil qui affiche le menu de toutes les gestions.

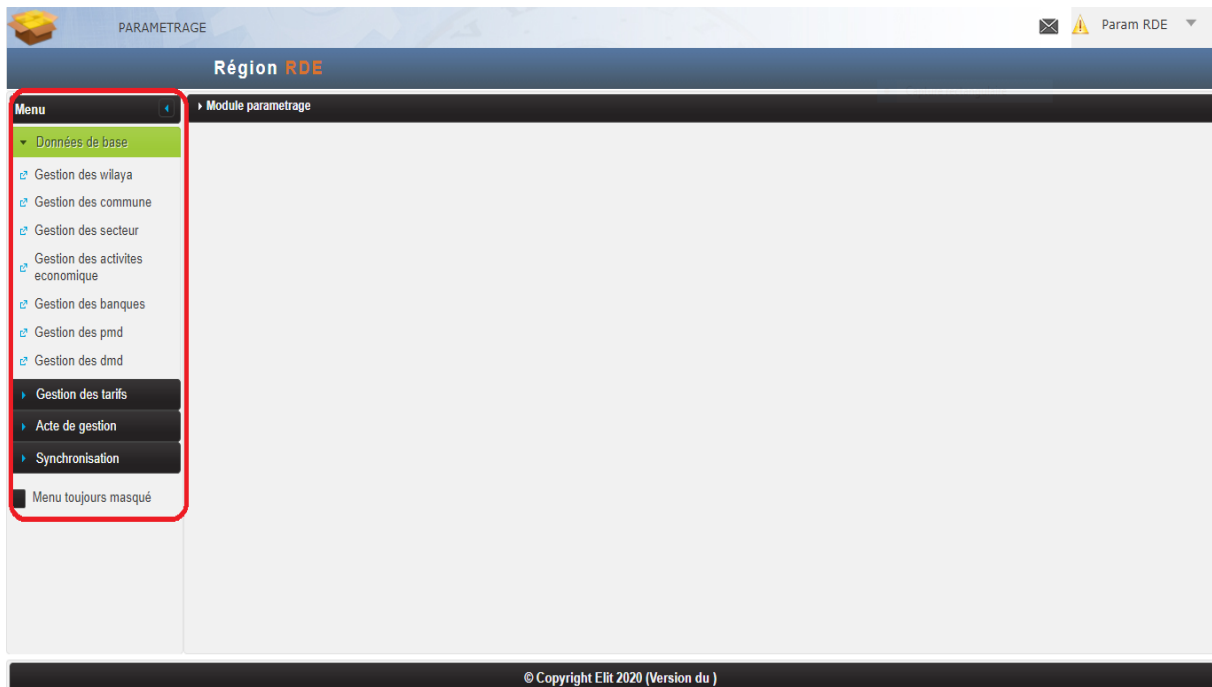


Figure IV-11: Interface d'accueil.

### IV.3.3. Interface de gestion activité économique

Parmi toutes les gestions nous avons choisi de présenter les interfaces de gestion activité économique.

#### A. La recherche d'une activité économique

Les trois figures suivantes (IV-12, IV-13, IV-14) représentent l'interface de gestion d'activité économique qui permet à l'administrateur de consulter toutes les activités économiques de la base centralisé ou de les DD et il permet aussi de rechercher des activités par DD, code ou libellé.

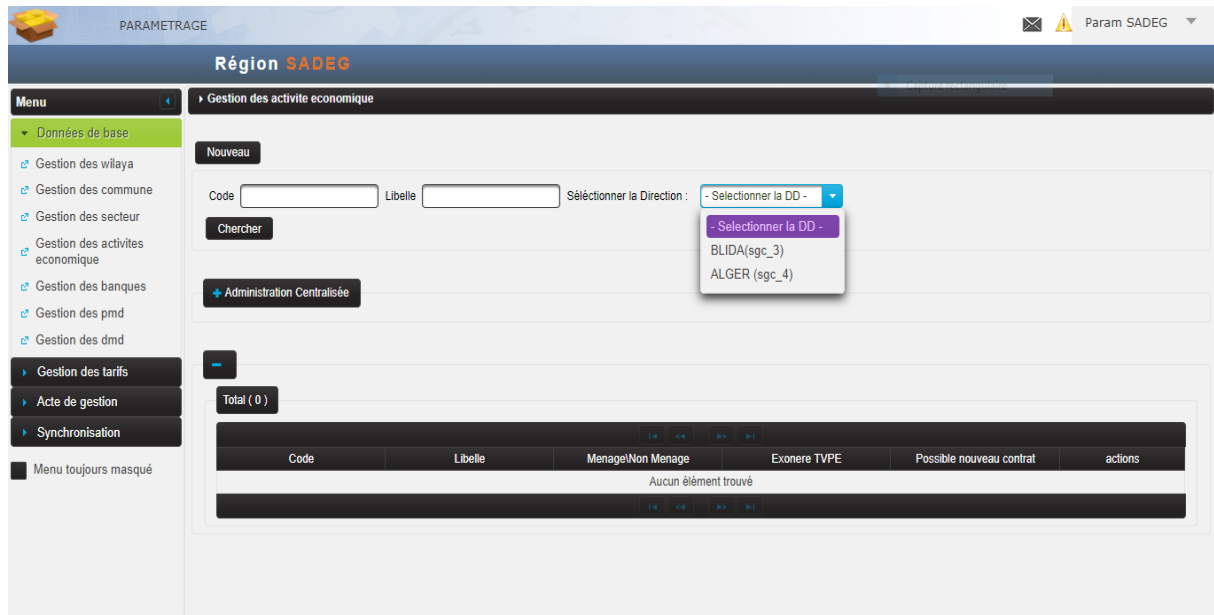


Figure IV-12: Interface de recherche d'une activité économique.

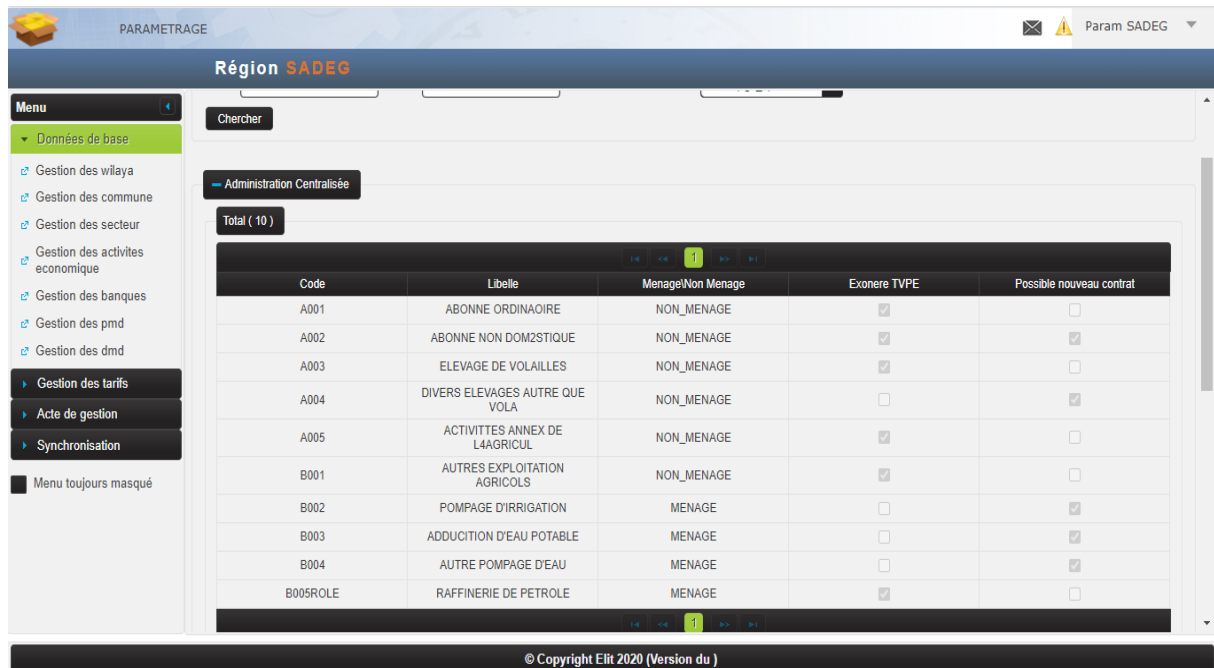


Figure IV-13: interface de consulter les activités économiques de la base centralisé.

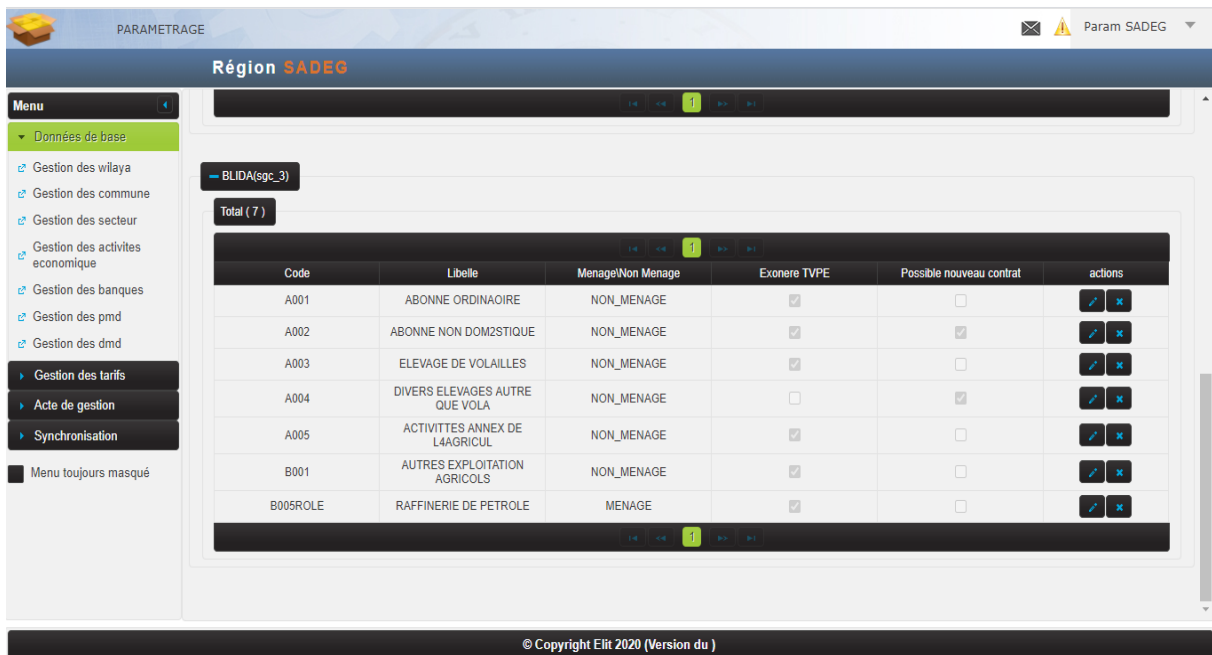


Figure IV-14: interface pour consulter les activités économiques par DD précise.

## B. L'ajout d'une activité économique

Pour ajouter une nouvelle activité économique en premier lieu l'administrateur doit saisir les informations de l'activité et en deuxième lieu il doit choisir les DD à ajouter comme démontré respectivement dans les figures qui suivent (IV-15 ,IV-16) :

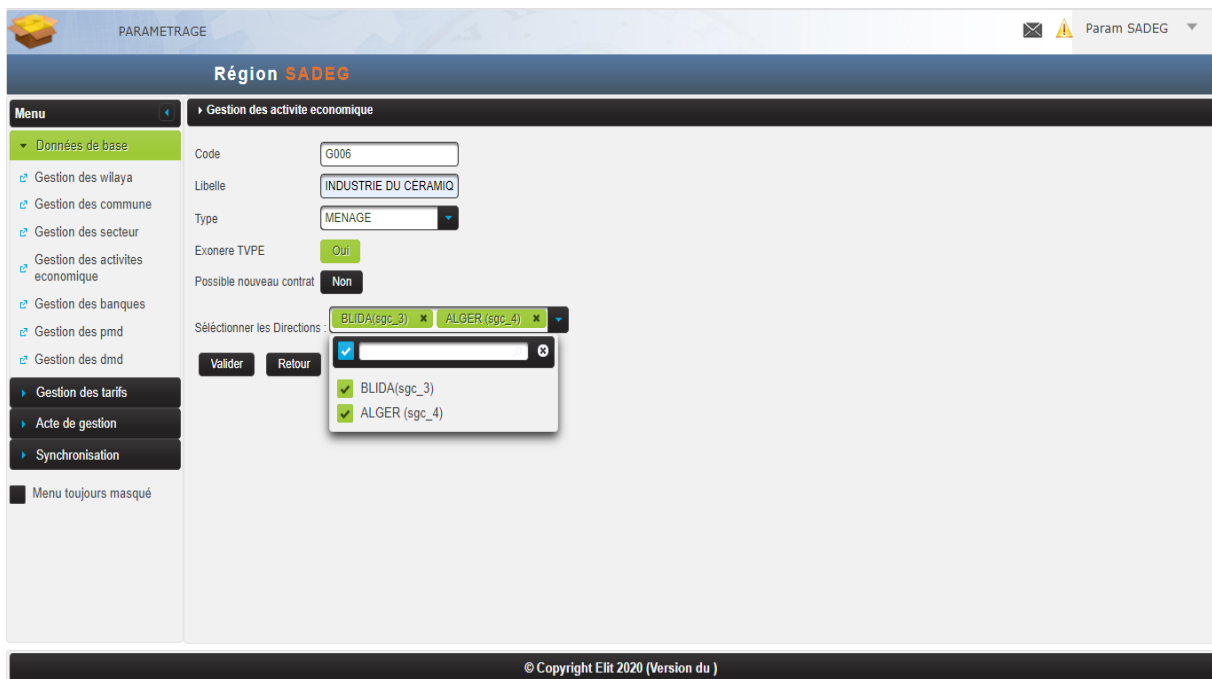


Figure IV-15: interface pour ajouter une activité économique.



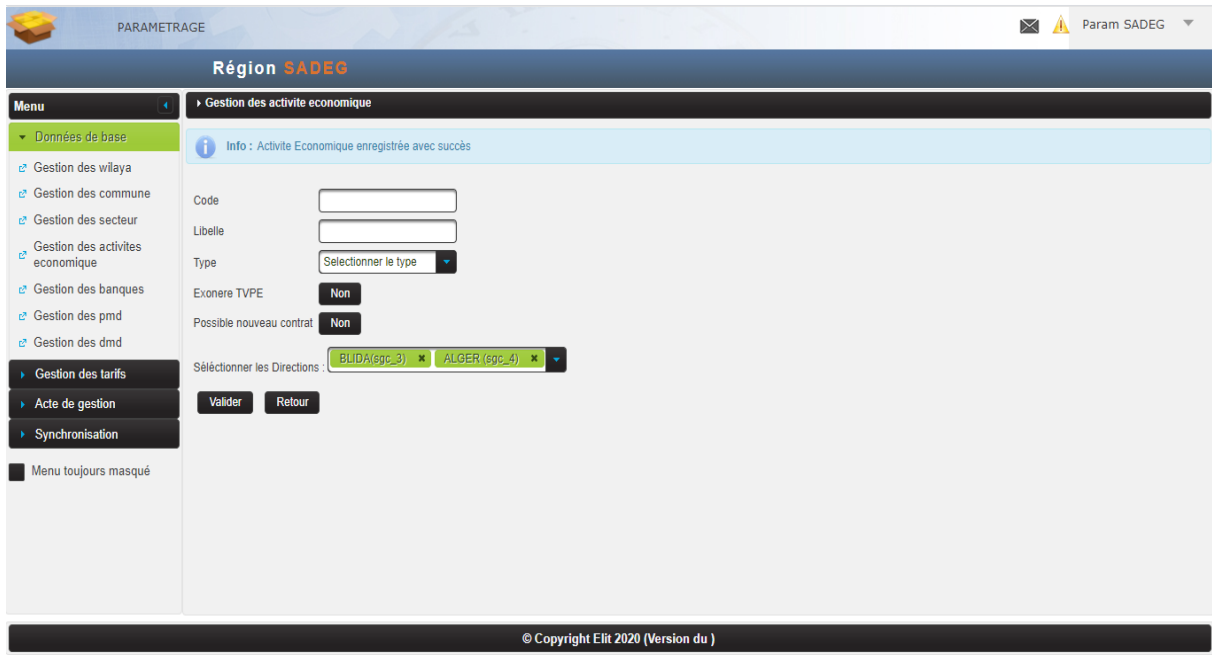


Figure IV-16: interface de résultat de l'ajout.

### C. La modification et la suppression d'une activité économique

Il y a deux boutons dans cette interface le premier bouton en rouge pour modifier une activité économique et le deuxième en vert pour la supprimer, comme illustré respectivement dans les figures qui suivent (IV-17, IV-18, IV-19):

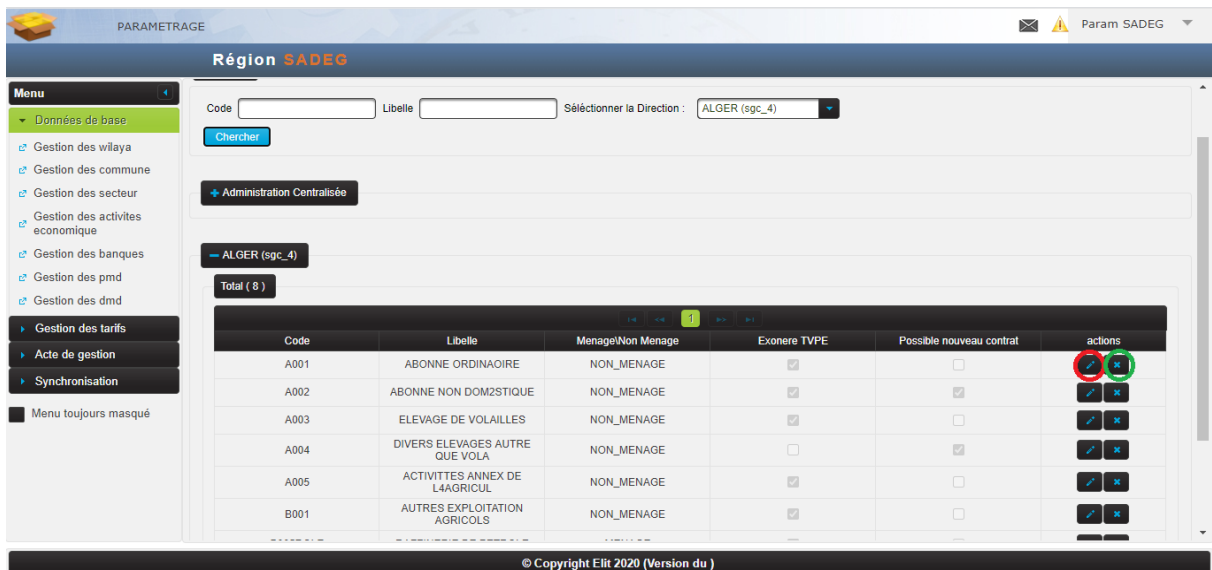


Figure IV-17: Affichage de toutes les activités économiques.



Figure IV-18: Interface pour la modification d'une activité économique.

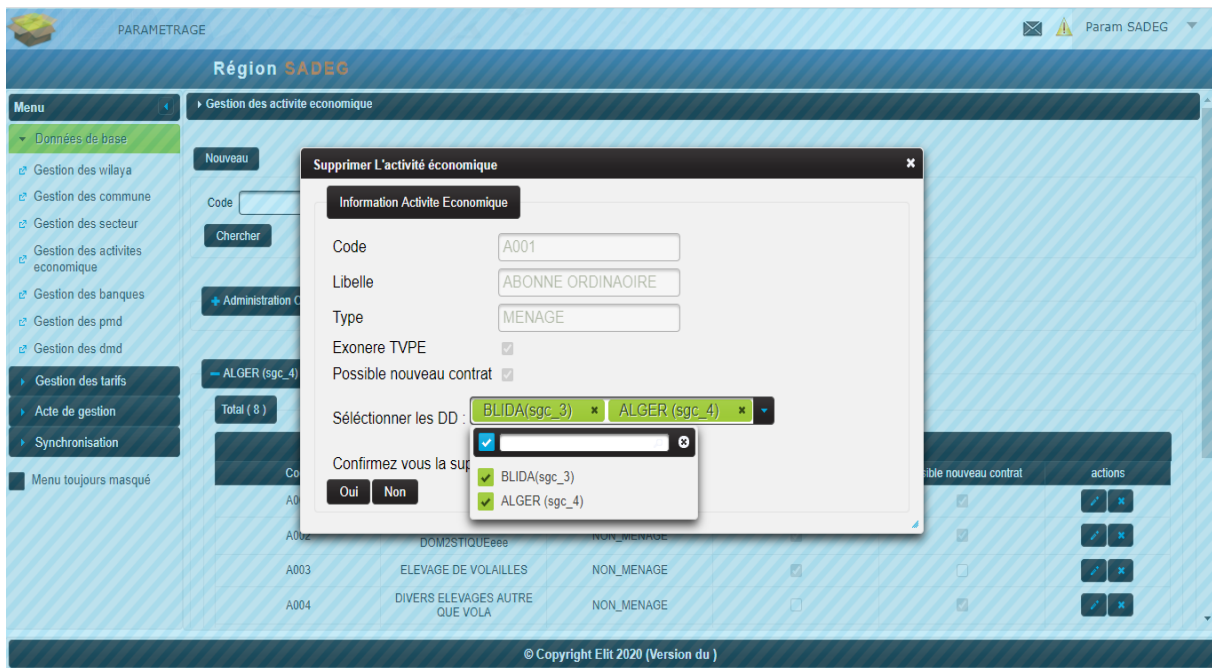


Figure IV-19: Interface pour supprimer d'une activité économique.

## **Conclusion**

Dans ce chapitre nous avons tout d'abord présenté les aspects et choix technologiques que nous avons choisis pour la réalisation et l'implémentation de notre architecture.

Nous avons ensuite présenté les différentes interfaces développées de notre application tout en exposant les fonctions qui constituent chaque espace. Nous avons abouti au terme de ce travail à un résultat conforme aux attentes des utilisateurs, et à un système qui reste extensible et évolutif.

## *Conclusion générale et perspective*

## Conclusion générale

Nous rappelons que notre travail adopte à l'architecture orientée service (SOA) comme modèle de développement favorise l'interopérabilité et l'intégration, et le but principal de ce travail est de développer une plateforme de service basée sur l'architecture SOA offrant au différent composant du système de gestion commercial pour cela on centralise les fonctionnalités communes entre les différents DD.

La réalisation de ce travail a nécessité au début de faire une collecte d'informations qui nous a permis à son tour de préparer une étude théorique au cours de laquelle nous avons étudié et critiqué l'existant. Cette étude nous a abouti à proposer quelques solutions pour choisir une approche selon notre problématique.

La solution retenue était étudiée profondément et nous avons pu grâce à cette étude déterminer les besoins fonctionnels et non fonctionnels de l'application. Arrivant à cette étape nous étions capables de passer à concevoir notre système.

La phase conception détaillée, a couvert tous les aspects conceptuels : statique et physique de l'application. Nous avons aussi, au cours de ce chapitre, défini l'architecture pour mettre en place notre système et on détaille notre solution munie de certains diagrammes UML.

Toutes les étapes sus indiquées nous ont abouti finalement à la dernière étape à savoir la réalisation. Nous avons débuté cette étape par la présentation de l'approche MVC, l'environnement matériel et logiciel utilisé pour l'implémentation du système. Puis, nous avons enchainé avec une série de captures d'écran illustrant le fonctionnement de l'application.

## Perspectives

Bien que les objectifs fixés soient atteints, nous pouvons envisager les améliorations suivantes :

- Améliorer l'aspect de sécurité des services web.
- Appliquer l'approche microservice au lieu les web services pour des fins de la rapidité des fonctionnalités et la localisation des bugs.
- Assurer la communication entre le SGC et d'autres solutions métiers tels que le système comptable.

## Références Bibliographiques

- [1] Hugoson, “Centralized versus Decentralized Information Systems: A Historical Flashback,” IFIP Adv. Inf. Commun. Technol., vol. 303, pp. 106–115, 2007.
- [2] G. Wiederhold, “Mediators in the Architecture of Future Information SystemsGio Wiederhold Abstract,” Processing, no. September 1991, pp. 1–36, 1992.
- [3] H. Telhaoui and A. Boutora, “Utilisation d’une ontologie linguistique (thésaurus) pour l’intégration des sources de données. ,” Université Dr. Tahar Moulay SAIDA, 2016.
- [4] F. Lahmar, “Une approche Hybride d’intégration de sources de données hétérogènes dans les datawarehouses,” Université Mentouri de Constantine, 2011.
- [5] B. Benmedakhene and I. Yahyaoui, “Composition des services Web : Etat de l’art,” Université Larbi Tébessi, 2016.
- [6] W. H. Inmon, “Building the data warehouse,” 2005.
- [7] J. SRINIVAS, S. R. VENKATA, and A. MOIZ QYSER, “Cloud ComputingBasics,” Creat. Smart Enterp., no. August 2013, pp. 141–171, 2012.
- [8] C. Pautasso and E. Wilde, “RESTful web services: principles, patterns, emerging technologies,” pp. 1359-1360, 2010.
- [9] M. Hacid, C. Reynaud, U. Claude, and B. Lyon, “L ’ intégration de sources de données,” 2004.
- [10] “Sonelgaz spa,” Consulté le 03/03/2020, à l’adresse <https://www.sonelgaz.dz/fr>
- [11] “Cahier de charge Reporting commercial « Projet Refonte SGC »,” version 0.3, réalisé le 26/11/2015, Direction commerciale et Marketing.
- [12] A. Siddiki, “Une Architecture Orientée Services ( SOA ) de Business Intelligence pour la gestion de l ’ éducation,”2012.
- [13] L. Madi, “D’une modélisation dans la conception d’un système d’information vers une architecture orientée services (SOA),” 2012.
- [14] G. Bieber and J. Carpenter, “Introduction to service-oriented programming (rev 2.1),” OpenWingsWhitepaper , pp. 1–13, 2001.
- [15] “SOA : Du composant au service,” Consulté le 1 avril 2020, à l’adresse <https://blog.engineering.publicissapient.fr/2009/08/11/soa-du-composant-au-service-la-composabilite/>
- [16] T. Yann, “Urbanisation & Intégration de Systèmes « THINK SERVICE »,” pp. 1–87, 2007.

- [17] P. DAEHNE, “Architecture SOA,” 2009.
- [18] D. Booth et al., “Web Services Architecture,” W3C, no. February, pp.1–98, 2004.
- [19] C. Lablanche, F. Seine, and S. Gastaud, “Les Web Services Rapport de TE,” 2005.
- [20] M. Borloz, “Gestion De Processus Avec Soa Et Bpm Dans Une Pme,” Univ. Fribg., 2013.
- [21] C. Pautasso, O. Zimmermann, and F. Leymann, “RESTful web services vs. ‘Big’ web services: Making the right architectural decision,” pp. 805–814, 2008.
- [22] A.Tebib, “Concepts et Outils pour l ’ Intégration et l ’ Interopérabilité des Services,” 2014.
- [23] T. Kiss et al., “MiCADO—Microservice-based Cloud Application-levelDynamicOrchestrator,” *Futur. Gener. Comput. Syst.*, vol. 94, pp. 937–946, 2019.
- [24] B. Benmedakhene and I. Yahyaoui, “Composition des services Web : Etat de l’art,” Université Larbi Tébessi, 2016.
- [25] D. Chappell, and T. Jewell, *Java Web Services*, O ’ REIL, 2002.
- [26] P. A. Muller and N.Gaertner, “Modélisation objet avec UML,” p. 420, 2000.
- [27] P. Kruchten, “Le Rational UnifiedProcess ,” p. 22, 2004.
- [28] N. Martin et al., “Une utilisation du modèle MVC pour une plate-forme de travail virtuel,” pp.131-138, 2020.
- [29] D. Jagodić, Ž. Jovanović, S. Randić, and D. Vujičić, “Java Spring Boot Rest Web Service IntegrationWith Java,” no. November, pp. 323–327, 2017.
- [30] “NetBeans,” Consulté le 5 août 2020, à l’adresse <https://netbeans.org/projects/www/>
- [31] “PostgreSql,” consulté le 5 août 2020, à l’adresse <https://www.postgresql.org/>
- [32] “JEE,” consulté le 6 août 2020, à l’adresse <https://www.asee.org/papers-and-publications/publications/jee>
- [33] “Visual-Paradim,” consulté le 6 août 2020, à l’adresse <https://www.visual-paradigm.com/>
- [34] “Glassfish,” consulté le 6 août 2020, à l’adresse <https://eclipse-ee4j.github.io/glassfish/>
- [35] N. Porteous, “ La construction du modèle logique d'un programme. Approches et pratiques en évaluation de programme,” pp. 87-105,2009.
- [36] P. Cloux, D. Doussot, and A. Geron, “LES ARCHITECTURES CLIENT-SERVEUR INTERNET ET INTRANET,” pp. 4–6,2000.



# *Annexe*

## Annexe A : le schéma de la base de données du SGC

Les figures suivantes représentent le schéma de la base de données SGC.

Tables (991)			
TESTEUR		activite_economique	
achat_adresse_pdl		adresse	
achat_auxiliaires		adresse_fact	
achat_bcc		adresse_lieu	
achat_certificat_homologation		affectation_type_structure_organisationnelle	
achat_comptage_elec		avoir_btcp_redressement_releve	
achat_comptage_gaz		banque	
achat_cooronnees_geo		cadran_htb	
achat_hist_certificat_homologation		cal_num_contrat_bt	
achat_hist_pdl_elec		cal_num_contrat_bt_set	
achat_hist_pdl_gaz		cataeg_client	
achat_marque		categ_sgc	
achat_mode_releve		categorie_client	
achat_motif_modification		client	
achat_motif_modification_comptagee		code_postal	
achat_motif_modification_comptageg		commune	
achat_nature_comptage		commune_refonte_sgc	
achat_partie_prenante		commune_unite_organisationnelle	
achat_pcs		compte_bancaire_ccp	
achat_pdl_elec		connexion	
achat_pdl_gaz		csv_releve	
achat_pression		devis	
achat_releve_elec		dmd	
achat_releve_gaz		droit_fixe	
achat_releve_retenue_gaz		droit_fixe_temp	
achat_tension		etat_affaire	
achat_type_pdl		etat_compteur	
acte_gestion_code_ap		etat_compteur_sgc	
acte_gestion_mouvement		etat_contrat	
acte_gestion_mouvement_rubrique		event_btcp_agent	
acte_gestion_mouvement_rubrique_rcn		event_btcp_anomalie_reprise	
acte_gestion_rubrique_rcn		event_btcp_comptage_updated	
actemouvement		event_btcp_data_avenant	
action		event_btcp_data_avoir_financier	
		event_btcp_data_branchment	
		event_btcp_data_changement_references	
		event_btcp_data_client	
		event_btcp_data_comptage	
		event_btcp_data_contrat	
		event_btcp_data_devis	
		event_btcp_data_devis_prestation	
		event_btcp_data_domiciliation	
		event_btcp_data_facture	
		event_btcp_data_frais	
		event_btcp_data_fusion_references	
		event_btcp_data_info_compteur	
		event_btcp_data_lieu_consommation	
		event_btcp_data_memoire	
		event_btcp_data_mutation	
		event_btcp_data_mutation_contrat	
		event_btcp_data_pmd_dmd	
		event_btcp_data_pose_compteur	
		event_btcp_data_prise_en_charge	
		event_btcp_data_realisation	
		event_btcp_data_registre	
		event_btcp_data_releve	
		event_btcp_data_rpe	
		event_btcp_data_tarif	
		event_btcp_deleted_facture	
		event_btcp_entreprise	
		event_btcp_evenement	
		event_btcp_info_object	
		event_btcp_info_user	
		event_btcp_journal	
		event_btcp_param_action	
		event_btcp_param_etat	
		event_btcp_param_motif	
		event_btcp_param_referentiel_prestation	
		event_btcp_param_select_etat	
		event_btcp_param_type	
		event_btcp_redressement	
		event_btcp_redressement_prestation	
		event_btcp_redressement_reference	
		event_btcp_redressement_registre	
		event_btcp_redressement_releve	
		event_btcp_suivi	
		event_btcp_tournee_verif	
		event_hp_donnees_prestation_redressement	
		event_hp_donnees_avoir_financier	
		event_hp_donnees_cession	
		event_hp_donnees_chiffrage	
		event_hp_donnees_client	
		event_hp_donnees_client_contr	
		event_hp_donnees_comptage	
		event_hp_donnees_contrat	
		event_hp_donnees_devis	
		event_hp_donnees_domiciliation	
		event_hp_donnees_etude	
		event_hp_donnees_facture	
		event_hp_donnees_lieu_consommation	
		event_hp_donnees_pmd	
		event_hp_donnees_prise_en_charge	
		event_hp_donnees_realisation	
		event_hp_donnees_redressement	
		event_hp_donnees_redressement_index	
		event_hp_donnees_registre	
		event_hp_donnees_releve	
		event_hp_donnees_rpe	
		event_hp_donnees_tarif	
		event_hp_donnees_taxe	
		event_hp_evenement	
		event_hp_info_object	

event_ha_param_action	event_hp_donnes_taxe
event_ha_param_etat	event_hp_evenement
event_ha_param_motif	event_hp_info_object
event_ha_param_select_etat	event_hp_redressement_comptage
event_ha_param_type	event_hp_redressement_registre
event_ha_referentiel_prestation	event_hp_redressement_releve
event_ha_suivi	event_hp_suivi_event
event_htb_data_avenant	event_ha_donnes_avoir_financier
event_htb_data_avoir_financier	event_ha_donnes_client
event_htb_data_client	event_ha_donnes_comptage
event_htb_data_comptage	event_ha_donnes_contest
event_htb_data_domociliation	event_ha_donnes_devis
event_htb_data_facture	event_ha_donnes_domociliation
event_htb_data_index_releve	event_ha_donnes_etude
event_htb_data_infos_contractuel	event_ha_donnes_facture
event_htb_data_pmd	event_ha_donnes_index_releve
event_htb_data_prestation	event_ha_donnes_infos_contractuel
event_htb_data_prise_en_charge	event_ha_donnes_lieu
event_htb_data_releve	event_ha_donnes_pmd
event_htb_data_repe	event_ha_donnes_prestation
event_htb_data_tarif	event_ha_donnes_prise_en_charge
event_htb_data_taxe	event_ha_donnes_rapport_tctp
event_htb_evenement	event_ha_donnes_recherche
event_htb_info_user	event_ha_donnes_redressement
event_htb_param_action	event_ha_donnes_redressement_prestation
event_htb_param_etat	event_ha_donnes_redressement_index
event_htb_param_motif	event_ha_donnes_redressement_releve
event_htb_param_select_etat	event_ha_donnes_releve
event_htb_param_type	event_ha_donnes_repe
event_htb_recherche	event_ha_donnes_tarif
event_htb_redressement	event_ha_donnes_taxe
event_htb_redressement_comptage	event_ha_donnes_transformateur
event_htb_redressement_index	event_ha_evenement
event_htb_redressement_prestation	event_ha_info_user
event_htb_redressement_reference	event_ha_infos_object
event_htb_redressement_releve	event_hp_infos_object
event_htb_redressement_taxe	event_mp_suivi_event
event_htb_referentiel_prestation	fact
event_htb_suivi	fact_affectation_lot_facture_hp
event_motif_mp_hp	fact_affectation_lot_facture_hta
event_mp_donnes_avoir_financier	fact_affectation_lot_facture_htb
event_mp_donnes_cession	fact_affectation_lot_facture_mp
event_mp_donnes_chiffrage	fact_btbp_acte_gestion
event_mp_donnes_client	fact_btbp_acte_mouvement
event_mp_donnes_comptage	fact_btbp_beneficiaire_aide_energie
event_mp_donnes_contractuelle	fact_btbp_beneficiaire_aide_energie_historique
event_mp_donnes_devis	fact_btbp_co_facture
event_mp_donnes_dmd	fact_btbp_contribution_facture
event_mp_donnes_domociliation	fact_btbp_detail_facture_contrat
event_mp_donnes_facture	fact_btbp_detail_tranche_facture
event_mp_donnes_info_generals	fact_btbp_droits_taxes_facture
event_mp_donnes_lieu_consommation	fact_btbp_element_facture_code_gestion
event_mp_donnes_prise_en_charge	fact_btbp_energie_compteurs
event_mp_donnes_realisation	fact_btbp_evenement
event_mp_donnes_redressement	fact_btbp_facture
event_mp_donnes_redressement_registre	fact_btbp_facture_contrat
event_mp_donnes_redressement_releve	fact_btbp_facture_devis
event_mp_donnes_registre	fact_btbp_facture_fms_lieu
event_mp_donnes_releve	fact_btbp_facture_par_page_memoire
event_mp_donnes_rgpe	fact_btbp_facture_unite_organ
event_mp_donnes_tarif	fact_btbp_filiale_aide_energie
event_mp_donnes_taxe	fact_btbp_memoire
event_mp_evenement	fact_btbp_memoire_unite_organ
event_mp_hp_param_action_mp_hp	fact_btbp_mouvement
event_mp_hp_param_chiffrage	fact_btbp_page_facture
event_mp_hp_param_etat_event	fact_btbp_page_memoire
event_mp_hp_param_select_etat	fact_btbp_prestation
event_mp_hp_param_type_event	fact_btbp_prestations_devis
event_mp_info_object	fact_btbp_prestations_frais_facture
event_mp_info_user	fact_btbp_rubrique_facture_btbp
	fact_btbp_seuil_avantage_energie

fact\_btbp\_soutien\_etat\_bt  
 fact\_btbp\_suivi\_facturation  
 fact\_btbp\_tranche\_tva  
 fact\_btbt\_prestations\_facture  
 fact\_facturation\_sous\_tache\_table\_hta  
 fact\_facturation\_tache\_table\_hta  
 fact\_facture\_hp  
 fact\_facture\_hta  
 fact\_facture\_htb  
 fact\_facture\_mp  
 fact\_facture\_releve\_hta  
 fact\_instruction\_calcul\_hp  
 fact\_lot\_facture\_btbp  
 fact\_lot\_facture\_hp  
 fact\_lot\_facture\_hta  
 fact\_lot\_facture\_htb  
 fact\_lot\_facture\_mp  
 fact\_lot\_memoire\_btbp  
 fact\_message\_facture\_hp  
 fact\_message\_facture\_hta  
 fact\_message\_facture\_htb  
 fact\_message\_facture\_mp  
 fact\_modele\_facture\_hta  
 fact\_modele\_facture\_htb  
 fact\_modele\_facture\_mp  
 fact\_modele\_format\_facture\_hp  
 fact\_modele\_format\_reference\_hp  
 fact\_modele\_format\_rubrique\_hp  
 fact\_modele\_format\_segment\_type\_facture\_hp  
 fact\_modele\_formule\_hta  
 fact\_modele\_formule\_htb  
 fact\_modele\_formule\_mp  
 fact\_modele\_instruction\_calcule\_htb  
 fact\_modele\_instruction\_calcule\_mp  
 fact\_modele\_instruction\_hta

fact\_modele\_parametre\_hta  
 fact\_modele\_reference\_hta  
 fact\_modele\_reference\_htb  
 fact\_modele\_reference\_mp  
 fact\_modele\_rubrique\_hta  
 fact\_modele\_rubrique\_htb  
 fact\_modele\_rubrique\_instruction\_htb  
 fact\_modele\_rubrique\_instruction\_mp  
 fact\_modele\_rubrique\_mp  
 fact\_modele\_segment\_facture\_hta  
 fact\_modele\_segment\_facture\_htb  
 fact\_modele\_segment\_facture\_mp  
 fact\_modele\_segment\_reference\_hta  
 fact\_modele\_segment\_reference\_htb  
 fact\_modele\_segment\_reference\_mp  
 fact\_modele\_segment\_rubrique\_htb  
 fact\_modele\_segment\_rubrique\_mp  
 fact\_reference\_facture\_hp  
 fact\_reference\_facture\_hta  
 fact\_reference\_facture\_htb  
 fact\_reference\_facture\_mp  
 fact\_reference\_formule\_hp  
 fact\_reference\_rubrique\_hp  
 fact\_reference\_rubrique\_instruction\_hp  
 fact\_rubrique\_facture\_hp  
 fact\_rubrique\_facture\_hta  
 fact\_rubrique\_facture\_htb  
 fact\_rubrique\_facture\_mp  
 fact\_segment\_facture\_hp  
 fact\_segment\_facture\_hta  
 fact\_segment\_facture\_htb  
 fact\_segment\_facture\_mp  
 fact\_table\_controle\_facturation\_htb  
 facteur\_k\_elec

facteur\_k\_gaz  
 facture\_double\_bdc  
 filtre\_gtc\_ctc  
 forme\_juridique  
 frais\_occ  
 hist\_cadrant\_mt  
 hist\_cadrant\_mt\_ancien  
 hist\_cadrant\_mt\_coef  
 hist\_cadrant\_mt\_nbpos  
 hist\_cadrant\_mt\_nouveau  
 histo\_historique\_consommation\_hp  
 histo\_historique\_consommation\_hta  
 histo\_historique\_consommation\_htb  
 histo\_historique\_consommation\_mp  
 histo\_historique\_consommation\_registre\_hp  
 histo\_historique\_consommation\_registre\_hta  
 histo\_historique\_consommation\_registre\_htb  
 histo\_historique\_consommation\_registre\_mp  
 index\_relev\_htb  
 integr\_avenant\_comptage\_hta  
 integr\_avenant\_hp  
 integr\_avenant\_hta  
 integr\_avenant\_htb  
 integr\_avenant\_mp  
 integr\_bp\_avenant  
 integr\_bp\_branchement  
 integr\_bp\_matiere  
 integr\_bp\_pression  
 integr\_bt\_avenant  
 integr\_bt\_branchement  
 integr\_bt\_calibre\_disjoncteur  
 integr\_bt\_cof\_lect  
 integr\_bt\_diff  
 integr\_bt\_nb\_conducteur  
 integr\_bt\_nb\_phase

integr\_bt\_nb\_pole  
 integr\_bt\_panneau  
 integr\_bt\_reglage\_disjoncteur  
 integr\_bt\_section  
 integr\_bt\_tension  
 integr\_bt\_type\_conducteur  
 integr\_btbp\_branch\_improd  
 integr\_btbp\_rpe  
 integr\_code\_gestion  
 integr\_coefficient\_altitude\_mp  
 integr\_comptage\_hp  
 integr\_comptage\_hta  
 integr\_comptage\_htb  
 integr\_comptage\_mp  
 integr\_compteur\_bt  
 integr\_compteurs\_bp  
 integr\_contrat\_bp  
 integr\_contrat\_bt  
 integr\_contrat\_bt\_bis  
 integr\_contrat\_hp  
 integr\_contrat\_hta  
 integr\_contrat\_htb  
 integr\_contrat\_mp  
 integr\_correcteur\_hp  
 integr\_correcteur\_mp  
 integr\_corress\_etat\_contrat  
 integr\_decision\_soutien\_etat  
 integr\_exoneration\_taxe\_avenant\_hp  
 integr\_exoneration\_taxe\_avenant\_hta  
 integr\_exoneration\_taxe\_avenant\_mp  
 integr\_exoneration\_taxe\_contrat\_hp  
 integr\_exoneration\_taxe\_contrat\_hta  
 integr\_exoneration\_taxe\_contrat\_mp  
 integr\_exoneration\_taxe\_htb  
 integr\_fms

integr\_fmms  
 integr\_fmms\_lieu\_consomation  
 integr\_hp\_prestation  
 integr\_hp\_rpe  
 integr\_hp\_rpe\_echantillon  
 integr\_hta\_facturation\_prestation  
 integr\_hta\_referentiel\_prestation  
 integr\_hta\_rpe  
 integr\_hta\_rpe\_echantillon  
 integr\_htb\_facturation\_prestation  
 integr\_htb\_rpe  
 integr\_htb\_rpe\_echelonnement  
 integr\_lieu\_consommation\_btbp  
 integr\_lieu\_consommation\_hp  
 integr\_lieu\_consommation\_htb  
 integr\_lieux\_consommation\_hta  
 integr\_lieux\_consommation\_mp  
 integr\_mp\_prestation  
 integr\_mp\_referentiel\_prestation  
 integr\_mp\_rpe  
 integr\_mp\_rpe\_echantillon  
 integr\_nature\_branchement  
 integr\_pdl\_hp  
 integr\_pdl\_hta  
 integr\_pdl\_htb  
 integr\_pdl\_mp  
 integr\_register\_hta  
 integr\_register\_mp  
 integr\_registre\_bp  
 integr\_registre\_bt  
 integr\_registre\_comptage\_hp  
 integr\_registre\_comptage\_htb  
 integr\_rejet\_contrat\_hta  
 integr\_rejet\_contrat\_mp  
 integr\_situation  
 integr\_soutien\_etat

integr\_soutien\_etat\_activite\_economique  
 integr\_taxe\_contrat\_activite\_economique  
 integr\_tc\_tp\_comptage\_hta  
 integr\_transformateur\_caracteristique  
 integr\_type\_compteur  
 integration  
 integration\_bp  
 integration\_bt  
 journal\_erreur\_job\_migration  
 journal\_erreur\_job\_rejet  
 journal\_error  
 journal\_stat\_job\_migration  
 jurid\_affaire\_atteinte\_tiers  
 jurid\_affaire\_cheque\_traite\_impaye  
 jurid\_affaire\_dossier\_precontentieu  
 jurid\_affaire\_fraude  
 jurid\_bordereau\_affaire\_precontentieux  
 jurid\_creance\_affaire\_precontentieux  
 jurid\_creance\_cheque\_traite\_impaye  
 jurid\_depot\_plainte\_atteinte\_tiers  
 jurid\_depot\_plainte\_cheque\_traite\_impaye  
 jurid\_depot\_plainte\_fraude  
 jurid\_detail\_affaire\_fraude  
 jurid\_detail\_bordereau\_affaire\_precontentieux  
 jurid\_dossier\_rcg  
 jurid\_etablissement\_juridique  
 jurid\_facture\_cheque\_traite\_impaye  
 jurid\_liste\_noire\_cheque\_impaye  
 jurid\_med\_atteinte\_tiers  
 jurid\_med\_cheque\_traite\_impaye  
 jurid\_med\_dossier\_precontentieu  
 jurid\_referentiel\_fraude  
 lc\_groupe  
 migration\_correction\_facture  
 migration\_facturation\_date\_paiement

migration\_mois  
 migration\_rcn\_devis  
 ministere  
 moteur\_moteur\_calcule\_parametrage  
 motif  
 pma\_coef\_relev\_htb  
 pmd  
 pression\_point\_livraison  
 rcn\_aff\_accord  
 rcn\_aff\_accord\_revision\_client  
 rcn\_aff\_affaire  
 rcn\_aff\_affectation\_lot\_facture\_rcn  
 rcn\_aff\_affectation\_mise\_en\_service  
 rcn\_aff\_affectation\_realisation  
 rcn\_aff\_autorisation  
 rcn\_aff\_chiffrage  
 rcn\_aff\_client\_lotissement  
 rcn\_aff\_client\_promoteur  
 rcn\_aff\_commun\_affaire  
 rcn\_aff\_commun\_chiffre\_affaire  
 rcn\_aff\_commun\_code\_ap  
 rcn\_aff\_commun\_demande  
 rcn\_aff\_commun\_documentclient  
 rcn\_aff\_commun\_info\_demande\_annul  
 rcn\_aff\_commun\_infoclient  
 rcn\_aff\_commun\_mandatitaire  
 rcn\_aff\_commun\_mandatitaire\_rcn\_aff\_pouvoirs\_accordes\_mandatitaire  
 rcn\_aff\_commun\_nature\_du\_site  
 rcn\_aff\_commun\_representant  
 rcn\_aff\_commun\_signaturetrav  
 rcn\_aff\_commun\_site\_raccordement  
 rcn\_aff\_commun\_type\_affaire  
 rcn\_aff\_commun\_type\_document  
 rcn\_aff\_constatation\_paiement  
 rcn\_aff\_contrat\_raccordement  
 rcn\_aff\_dgd

rcn\_aff\_document  
 rcn\_aff\_donnees\_avenant\_contrat  
 rcn\_aff\_donnees\_contrat\_fourniture  
 rcn\_aff\_donnees\_oet  
 rcn\_aff\_donnees\_ost  
 rcn\_aff\_engagement  
 rcn\_aff\_etude  
 rcn\_aff\_facture\_rcn  
 rcn\_aff\_famille\_facture  
 rcn\_aff\_htbhp\_accord  
 rcn\_aff\_htbhp\_affaire  
 rcn\_aff\_htbhp\_affectation\_etude  
 rcn\_aff\_htbhp\_affectation\_lot\_facture\_rcn\_htbhp  
 rcn\_aff\_htbhp\_annul\_dgd  
 rcn\_aff\_htbhp\_annul\_finale  
 rcn\_aff\_htbhp\_asc  
 rcn\_aff\_htbhp\_besoins\_energie  
 rcn\_aff\_htbhp\_capacite\_prod\_elec  
 rcn\_aff\_htbhp\_capacite\_prod\_gaz  
 rcn\_aff\_htbhp\_caracteristique\_besoins\_elec  
 rcn\_aff\_htbhp\_caracteristique\_besoins\_gaz  
 rcn\_aff\_htbhp\_caracteristique\_des\_moteurs  
 rcn\_aff\_htbhp\_compteur\_info  
 rcn\_aff\_htbhp\_contact  
 rcn\_aff\_htbhp\_demande\_facture\_sup  
 rcn\_aff\_htbhp\_document  
 rcn\_aff\_htbhp\_enregistrement\_dgdhtbhp  
 rcn\_aff\_htbhp\_enregistrement\_facture\_sup  
 rcn\_aff\_htbhp\_enregistrement\_ods  
 rcn\_aff\_htbhp\_envoi\_relance  
 rcn\_aff\_htbhp\_etude  
 rcn\_aff\_htbhp\_facturation\_dgd  
 rcn\_aff\_htbhp\_facture\_priliminaires  
 rcn\_aff\_htbhp\_facture\_rcnhtbhp  
 rcn\_aff\_htbhp\_index\_compteur  
 rcn\_aff\_htbhp\_information\_generales

## Annexe A : le schéma de la base de données du SGC

- rcn\_aff\_htbhp\_information\_techniques
- rcn\_aff\_htbhp\_justification\_paiement
- rcn\_aff\_htbhp\_lettre\_prise\_en\_charge
- rcn\_aff\_htbhp\_ligne\_secours
- rcn\_aff\_htbhp\_lot\_facture\_cnhtbhp
- rcn\_aff\_htbhp\_mise\_service
- rcn\_aff\_htbhp\_montent\_enc
- rcn\_aff\_htbhp\_offre\_raccordement
- rcn\_aff\_htbhp\_oms
- rcn\_aff\_htbhp\_paiement
- rcn\_aff\_htbhp\_paiement\_asc
- rcn\_aff\_htbhp\_paiement\_dgd
- rcn\_aff\_htbhp\_paiement\_facture\_preliminaire
- rcn\_aff\_htbhp\_paiement\_annulation\_dgd
- rcn\_aff\_htbhp\_poste\_transformateur\_client
- rcn\_aff\_htbhp\_programmation\_realisation
- rcn\_aff\_htbhp\_realisation
- rcn\_aff\_htbhp\_reception\_dgd
- rcn\_aff\_htbhp\_reception\_dgd\_final\_hp
- rcn\_aff\_htbhp\_reception\_dossier\_grt
- rcn\_aff\_htbhp\_reception\_dossier\_technique
- rcn\_aff\_htbhp\_suivi\_affaire
- rcn\_aff\_htbhp\_supervision
- rcn\_aff\_htbhp\_transmission\_dgd
- rcn\_aff\_htbhp\_transmissionavance
- rcn\_aff\_htbhp\_transmission\_dossier\_grt
- rcn\_aff\_htbhp\_transmission\_dossier\_technique
- rcn\_aff\_htbhp\_transmission\_facture
- rcn\_aff\_htbhp\_transmission\_facture\_sup
- rcn\_aff\_htbhp\_transmission\_mes
- rcn\_aff\_htbhp\_transmission\_offre
- rcn\_aff\_htbhp\_transmission\_oms
- rcn\_aff\_htbhp\_transmission\_paiement\_avance
- rcn\_aff\_info\_qualification
- rcn\_aff\_lot\_facture\_rcn
- rcn\_aff\_mise\_service

- rcn\_aff\_modification\_travaux
- rcn\_aff\_nature\_prise\_charge
- rcn\_aff\_paiement
- rcn\_aff\_pave\_facture
- rcn\_aff\_pouvoirs\_accordes\_mondataire
- rcn\_aff\_prise\_charge
- rcn\_aff\_programmation
- rcn\_aff\_realisation
- rcn\_aff\_rebrique\_facture
- rcn\_aff\_reception\_autorisation
- rcn\_aff\_sous\_rebrique\_etude
- rcn\_aff\_sous\_rebrique\_facture
- rcn\_aff\_suivi\_affaire
- rcn\_aff\_transmission\_ods\_contrat\_travaux
- rcn\_ag\_btbp\_accord
- rcn\_ag\_btbp\_affaire
- rcn\_ag\_btbp\_branchement
- rcn\_ag\_btbp\_client
- rcn\_ag\_btbp\_compteur
- rcn\_ag\_btbp\_contrat
- rcn\_ag\_btbp\_devis
- rcn\_ag\_btbp\_facture
- rcn\_ag\_btbp\_frais
- rcn\_ag\_btbp\_lieu
- rcn\_ag\_btbp\_paiement
- rcn\_ag\_btbp\_prestation
- rcn\_ag\_btbp\_realisation
- rcn\_ag\_btbp\_registre
- rcn\_ag\_btbp\_rpe
- rcn\_ag\_btbp\_suivi
- rcn\_ag\_btbp\_user
- rcn\_annul\_enregistrement\_dgdhtbhp
- rcn\_annul\_etat\_annulation\_htbhp
- rcn\_annul\_etat\_annulationbtbp
- rcn\_annul\_info\_annuleebtbp
- rcn\_annul\_info\_annuleehtbhp

- reclam\_recepisse
- reclam\_reclamation
- reclam\_reclamation\_hist
- reclam\_recours
- reclam\_referentiel\_delais
- reclam\_reliance
- reclam\_requerant
- reclam\_sous\_reclamation
- reclam\_sous\_reclamation\_hist
- reclam\_type\_requerant
- reclam\_user\_tracabilite
- reclam\_voie\_reception
- recouv\_btbp\_bordero\_presentation\_creance
- recouv\_action
- recouv\_attache\_commerciale
- recouv\_btbp\_bordero\_presentation
- recouv\_btbp\_creance
- recouv\_btbp\_creance\_sauv
- recouv\_btbp\_lot\_creance
- recouv\_btbp\_synchronisateur
- recouv\_etat\_contrat
- recouv\_facture\_travaux
- recouv\_hp\_creance
- recouv\_hp\_lot\_creance
- recouv\_hp\_suivi\_contrat
- recouv\_hta\_creance
- recouv\_hta\_lot\_creance
- recouv\_hta\_suivi\_contrat
- recouv\_htb\_creance
- recouv\_htb\_lot\_creance
- recouv\_htb\_suivi\_contrat
- recouv\_mp\_creance
- recouv\_mp\_lot\_creance
- recouv\_mp\_suivi\_contrat
- recouv\_pres\_hp\_bordereau\_remise
- recouv\_pres\_hp\_bordereau\_remise\_creance

- rcn\_annul\_motif\_annulationbtbp
- rcn\_annul\_motif annullhtbhp
- rcn\_annul\_paiement\_annulation\_dgd
- rcn\_annul\_reception\_situation\_dgd
- rcn\_annul\_transmission\_annulation\_dgd
- rcn\_annul\_transmission\_annulation\_grthtb
- rcn\_annul\_transmission\_situation
- rcn\_param\_action
- rcn\_param\_etat
- rcn\_param\_famille
- rcn\_param\_frais\_generaux
- rcn\_param\_motif
- rcn\_param\_pave
- rcn\_param\_revision\_bareme
- rcn\_param\_rubrique
- rcn\_param\_select\_etat
- rcn\_param\_sous\_rubrique
- rcn\_param\_type\_autre\_travaux\_rcn
- rcn\_param\_type\_processus\_rcn
- reclam\_accuse\_reception
- reclam\_affectation
- reclam\_categorie\_requerant
- reclam\_degre\_urgence
- reclam\_etat\_reclamation
- reclam\_etat\_sous\_reclamation
- reclam\_etat\_sous\_reclamation\_hist
- reclam\_info\_supp
- reclam\_motif\_naturemotif
- reclam\_motif\_reclamation
- reclam\_motif\_reference
- reclam\_nature\_motif
- reclam\_niveau\_prise\_en\_charge
- reclam\_notification
- reclam\_parametre\_gestion\_motif
- reclam\_piece\_jointe
- reclam\_qualite\_requerant

recouv\_pres\_hta\_bordereau\_remise  
 recouv\_pres\_hta\_bordereau\_remise\_creance  
 recouv\_pres\_htb\_bordereau\_remise  
 recouv\_pres\_htb\_bordereau\_remise\_creance  
 recouv\_pres\_mp\_bordereau\_remise  
 recouv\_pres\_mp\_bordereau\_remise\_creance  
 recouv\_sous\_traitant  
 recouvrement\_attache\_commercial\_travaux  
 recouvrement\_btbp\_document\_suivi  
 recouvrement\_btbp\_echeancier  
 recouvrement\_btbp\_lot\_recouvrement  
 recouvrement\_btbp\_notification\_reglement  
 recouvrement\_btbp\_parametrage  
 recouvrement\_btbp\_parametrage\_lot  
 recouvrement\_btbp\_registre\_index\_recouvrement  
 recouvrement\_btbp\_statut\_date  
 recouvrement\_btbp\_suivi\_creance  
 recouvrement\_btbp\_tranche  
 recouvrement\_creance\_situation  
 recouvrement\_creance\_travaux  
 recouvrement\_encaissement\_a\_tort  
 recouvrement\_facture\_identifie  
 recouvrement\_fms\_attache\_commercial  
 recouvrement\_fms\_bordereau\_remise  
 recouvrement\_fms\_bordereau\_remise\_creance  
 recouvrement\_fms\_creance  
 recouvrement\_fms\_demarchage  
 recouvrement\_fms\_demarchage\_creance  
 recouvrement\_fms\_document\_suivi  
 recouvrement\_fms\_echeancier  
 recouvrement\_fms\_notification\_reglement  
 recouvrement\_fms\_parametrage  
 recouvrement\_fms\_suivi  
 recouvrement\_fms\_suivi\_creance  
 recouvrement\_hp\_accord  
 recouvrement\_hp\_accord\_creance

recouvrement\_hp\_accord\_creance  
 recouvrement\_hp\_attache\_commercial  
 recouvrement\_hp\_bordereau\_remise  
 recouvrement\_hp\_bordereau\_remise\_creance  
 recouvrement\_hp\_creance  
 recouvrement\_hp\_demarchage  
 recouvrement\_hp\_demarchage\_creance  
 recouvrement\_hp\_document  
 recouvrement\_hp\_document\_creance  
 recouvrement\_hp\_echeancier  
 recouvrement\_hp\_lot\_creance  
 recouvrement\_hp\_notification\_reglement  
 recouvrement\_hp\_parametrage\_action  
 recouvrement\_hp\_parametrage\_etat  
 recouvrement\_hp\_suivi\_contrat  
 recouvrement\_hp\_suivi\_creance  
 recouvrement\_hp\_tranche  
 recouvrement\_hta\_accord  
 recouvrement\_hta\_accord\_creance  
 recouvrement\_hta\_attache\_commercial  
 recouvrement\_hta\_bordereau\_remise  
 recouvrement\_hta\_bordereau\_remise\_creance  
 recouvrement\_hta\_creance  
 recouvrement\_hta\_creance\_sauv  
 recouvrement\_hta\_demarchage  
 recouvrement\_hta\_demarchage\_creance  
 recouvrement\_hta\_document  
 recouvrement\_hta\_document\_creance  
 recouvrement\_hta\_echeancier  
 recouvrement\_hta\_lot\_creance  
 recouvrement\_hta\_notification\_reglement  
 recouvrement\_hta\_parametrage\_action  
 recouvrement\_hta\_parametrage\_etat  
 recouvrement\_hta\_suivi\_contrat  
 recouvrement\_hta\_suivi\_creance  
 recouvrement\_hta\_tranche  
 recouvrement\_htb\_accord

recouvrement\_htb\_accord\_creance  
 recouvrement\_htb\_attache\_commercial  
 recouvrement\_htb\_bordereau\_remise  
 recouvrement\_htb\_bordereau\_remise\_creance  
 recouvrement\_htb\_creance  
 recouvrement\_htb\_demarchage  
 recouvrement\_htb\_demarchage\_creance  
 recouvrement\_htb\_document  
 recouvrement\_htb\_document\_creance  
 recouvrement\_htb\_echeancier  
 recouvrement\_htb\_lot\_creance  
 recouvrement\_htb\_notification\_reglement  
 recouvrement\_htb\_parametrage\_action  
 recouvrement\_htb\_parametrage\_etat  
 recouvrement\_htb\_suivi\_contrat  
 recouvrement\_htb\_suivi\_creance  
 recouvrement\_htb\_tranche  
 recouvrement\_inventaire\_situations  
 recouvrement\_mp\_accord  
 recouvrement\_mp\_accord\_creance  
 recouvrement\_mp\_attache\_commercial  
 recouvrement\_mp\_bordereau\_remise  
 recouvrement\_mp\_bordereau\_remise\_creance  
 recouvrement\_mp\_creance  
 recouvrement\_mp\_creance\_sauv  
 recouvrement\_mp\_demarchage  
 recouvrement\_mp\_demarchage\_creance  
 recouvrement\_mp\_document  
 recouvrement\_mp\_document\_creance  
 recouvrement\_mp\_echeancier  
 recouvrement\_mp\_lot\_creance  
 recouvrement\_mp\_notification\_reglement  
 recouvrement\_mp\_parametrage\_action  
 recouvrement\_mp\_parametrage\_etat  
 recouvrement\_mp\_suivi\_contrat  
 recouvrement\_mp\_suivi\_creance  
 recouvrement\_mp\_tranche

recouvrement\_htb\_accord\_creance  
 recouvrement\_htb\_attache\_commercial  
 recouvrement\_htb\_bordereau\_remise  
 recouvrement\_htb\_bordereau\_remise\_creance  
 recouvrement\_htb\_creance  
 recouvrement\_htb\_demarchage  
 recouvrement\_htb\_demarchage\_creance  
 recouvrement\_htb\_document  
 recouvrement\_htb\_document\_creance  
 recouvrement\_htb\_echeancier  
 recouvrement\_htb\_lot\_creance  
 recouvrement\_htb\_notification\_reglement  
 recouvrement\_htb\_parametrage\_action  
 recouvrement\_htb\_parametrage\_etat  
 recouvrement\_htb\_suivi\_contrat  
 recouvrement\_htb\_suivi\_creance  
 recouvrement\_htb\_tranche  
 recouvrement\_inventaire\_situations  
 recouvrement\_mp\_accord  
 recouvrement\_mp\_accord\_creance  
 recouvrement\_mp\_attache\_commercial  
 recouvrement\_mp\_bordereau\_remise  
 recouvrement\_mp\_bordereau\_remise\_creance  
 recouvrement\_mp\_creance  
 recouvrement\_mp\_creance\_sauv  
 recouvrement\_mp\_demarchage  
 recouvrement\_mp\_demarchage\_creance  
 recouvrement\_mp\_document  
 recouvrement\_mp\_document\_creance  
 recouvrement\_mp\_echeancier  
 recouvrement\_mp\_lot\_creance  
 recouvrement\_mp\_notification\_reglement  
 recouvrement\_mp\_parametrage\_action  
 recouvrement\_mp\_parametrage\_etat  
 recouvrement\_mp\_suivi\_contrat  
 recouvrement\_mp\_suivi\_creance  
 recouvrement\_mp\_tranche

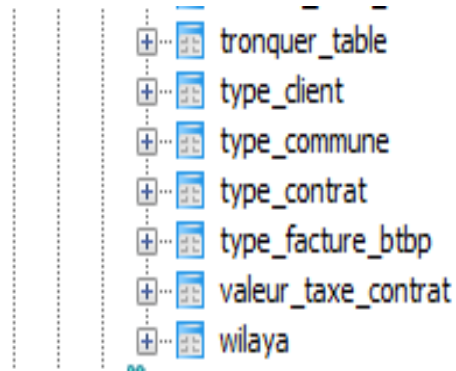
rejet\_pcx\_btbp  
 rejet\_register\_hta  
 rejet\_register\_mp  
 relev\_anomalie\_hp  
 relev\_anomalie\_releve\_hta  
 relev\_anomalie\_releve\_htb  
 relev\_anomalie\_releve\_mp  
 relev\_anomalie\_tele\_releve\_htb  
 relev\_btbp\_alert\_releve  
 relev\_btbp\_alert\_releve\_histo  
 relev\_btbp\_anomalie  
 relev\_btbp\_avoir\_non\_traite  
 relev\_btbp\_calendrier  
 relev\_btbp\_circuit  
 relev\_btbp\_coefficient\_modulateur  
 relev\_btbp\_commune\_fective  
 relev\_btbp\_compteur\_absant  
 relev\_btbp\_compteur\_coupe  
 relev\_btbp\_compteur\_defectueux  
 relev\_btbp\_compteur\_hors\_circuit  
 relev\_btbp\_compteur\_improductif  
 relev\_btbp\_compteur\_non\_trouve  
 relev\_btbp\_compteur\_remplace  
 relev\_btbp\_compteur\_retire  
 relev\_btbp\_donnees\_configuration\_calendrier  
 relev\_btbp\_fichier\_tsp  
 relev\_btbp\_historique\_consommation  
 relev\_btbp\_index\_bp  
 relev\_btbp\_index\_bp\_histo  
 relev\_btbp\_index\_bp\_sos\_27\_03  
 relev\_btbp\_index\_bt  
 relev\_btbp\_index\_bt\_histo  
 relev\_btbp\_index\_bt\_sos\_27\_03  
 relev\_btbp\_jour\_ferie  
 relev\_btbp\_oparation\_en\_cour  
 relev\_btbp\_refrentiel\_estimation

relev\_btbp\_releve  
 relev\_btbp\_releve\_anomalie  
 relev\_btbp\_releve\_histo  
 relev\_btbp\_tournee  
 relev\_btbp\_trace\_calendrier  
 relev\_btbp\_tsp\_abonne  
 relev\_btbp\_validation\_releve  
 relev\_calendrier\_hp  
 relev\_calendrier\_hta  
 relev\_calendrier\_htb  
 relev\_calendrier\_mp  
 relev\_compte\_rendu\_releve\_hp  
 relev\_compte\_rendu\_releve\_hta  
 relev\_compte\_rendu\_releve\_mp  
 relev\_compte\_rendue\_releve\_htb  
 relev\_compte\_rendue\_tele\_releve\_htb  
 relev\_contrat\_source\_releve\_hta  
 relev\_contrat\_source\_releve\_mp  
 relev\_date\_calendrier\_hp  
 relev\_date\_calendrier\_hta  
 relev\_date\_calendrier\_htb  
 relev\_date\_calendrier\_mp  
 relev\_detail\_compte\_rendu\_anomalie\_hp  
 relev\_detail\_compte\_rendu\_signale\_hp  
 relev\_detail\_consommation\_comptage\_htb  
 relev\_detail\_consommation\_registre\_hp  
 relev\_detail\_consommation\_registre\_htb  
 relev\_entree\_releve\_hta  
 relev\_entree\_releve\_mp  
 relev\_groupe\_hp  
 relev\_groupe\_hta  
 relev\_groupe\_htb  
 relev\_groupe\_mp  
 relev\_hta\_commune\_fective  
 relev\_index\_releve\_export\_hta  
 relev\_index\_releve\_hta

relev\_index\_releve\_hta  
 relev\_index\_releve\_mp  
 relev\_reference\_lieu  
 relev\_releve\_export\_hta  
 relev\_releve\_hp  
 relev\_releve\_hta  
 relev\_releve\_htb  
 relev\_releve\_mp  
 relev\_signale\_hp  
 relev\_signales\_detail\_consommation\_comptage\_htb  
 relev\_signales\_htb  
 relev\_source\_releve\_hta  
 relev\_source\_releve\_mp  
 relev\_tele\_releve\_htb  
 relev\_tele\_releve\_htb\_rejet  
 relev\_tournee\_hp  
 relev\_tournee\_hta  
 relev\_tournee\_htb  
 relev\_tournee\_mp  
 relev\_trace\_releve  
 relev\_tsp\_de\_cryptage  
 relev\_tsp\_comptes\_sftp  
 relev\_tsp\_fichier\_releve  
 relev\_tsp\_historique\_mouvement  
 relev\_tsp\_rejet  
 relev\_type\_anomalie\_tele\_releve\_htb  
 relev\_valeur\_control\_contrat\_changement\_hta  
 relev\_valeur\_control\_contrat\_hta  
 relev\_valeur\_control\_contrat\_mp  
 relev\_valeur\_control\_hta  
 relev\_valeur\_control\_mp  
 relev\_valeur\_control\_releve\_hta  
 relev\_valeur\_control\_releve\_mp  
 relev\_valeur\_control\_source\_releve\_hta  
 relev\_valeur\_control\_source\_releve\_mp  
 releve\_btbp\_id\_index\_bp  
 releve\_btbp\_id\_index\_bt

releve\_releve\_export\_hta  
 releve\_releve\_gaz  
 reponse\_reclam  
 report\_migr\_btbp  
 report\_migr\_fact\_hta  
 report\_migr\_fact\_mp  
 report\_migr\_htamp  
 report\_migr\_htbhp  
 scheduler\_scheduled\_task  
 scheduler\_task\_execution  
 secteur  
 sequence  
 sgc\_cadran  
 sgc\_contrat\_bt  
 sgc\_cumul\_fact  
 sgc\_detail\_fac\_bt  
 sgc\_histo\_conso  
 soutien\_etat  
 tar\_num\_cadr  
 tar\_numcadr  
 tarif  
 tarif\_plage\_horaire  
 tarif\_poste\_horaire  
 tarif\_prix\_unitaire  
 tarif\_puissance  
 tarif\_rebrique  
 tarif\_rebrique\_poste\_horaire  
 tarif\_tranche  
 tarif\_tranche\_groupe  
 tarif\_type\_rebrique  
 tarif\_version  
 taxe\_contrat  
 taxe\_habitation  
 tension\_point\_livraison  
 tronquer\_table  
 type\_dient  
 type\_commune





**Figure A-1:** schéma de SGC.

**Annexe B : Le modèle relationnel de parametrage.**

Les tables	Les champs	Type
Admin_unite_organisationnelle	<u>Id</u> Code	Integer(10) Character varying(255)
Banque	<u>Id</u> Code Orga code_banque adresse #admin_unite_organisationnelleid	Integer(10) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Integer(10)
acte_gestion_mouvement	<u>Id</u> code_mouvement compte_comptable_hors_groupe_code_mouvement compte_comptable_intra_groupe_code_mouvement libelle_mouvement sens_valeur ratio_one_imputation_analytique ratio_two_imputation_analytique is_ttc exist_regularisation exist_imputation_analytique	Integer(10) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Integer(10) Character varying(255)
acte_gestion_mouvement_rubrique	#acte_gestion_mouvementidactive #acte_gestion_rubrique_rcnid code_rubrique_externe type_facture	integer(10) integer(10) integer(10) Character varying(255)
acte_gestion_rubrique_rcn	<u>Id</u> code_rubrique type_raccordement_id	integer(10) Character varying(255) Character varying(255)

	type_depense	Character varying(255)
acte_gestion_mouvement_admin_unite_organisationnelle	#acte_gestion_mouvementid #admin_unite_organisationnelleid	integer(10) integer(10)
scheduler_scheduled_task	<u>Id</u> Activation code_tache heure jour_de_mois libelle_tache seconde	integer(10) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255) Character varying(255)
scheduler_scheduled_task_admin_unite_organisationnelle	#scheduler_scheduled_taskid #admin_unite_organisationnelleid	integer(10) integer(10)
commune	<u>Id</u> Code code_zone libelle #wilayaid	integer(10) Character varying(255) Character varying(255) Character varying(255) integer(10)
commune_unite_organisationnelle	#admin_unite_organisationnelleid #communeid	integer(10) integer(10)
contrat	<u>Id</u> type_contrat #pmdid #dmdid #tarifid	integer(10) integer(10) integer(10) integer(10) integer(10)
dmd	<u>Id</u> valeurnormalisee	integer(10) integer(10)
pmd	<u>Id</u> valeurnormalisee	integer(10) integer(10)
secteur	<u>Id</u> Libelle code	integer(10) Character varying(255) Character varying(255)

activite_economique	<u>Id</u> Code exonere_ttype libelle type_activite possible_nouveau_contrat #secteurid	integer(10) Character varying(255) Character varying(255) Character varying(255) Character varying(255) integer(10) integer(10)
tarif	<u>Id</u> Libelle tar_code	integer(10) Character varying(255) Character varying(255)
tarif_version	<u>Id</u> date_effet numero_version type_version code_decision #tarifid	integer(10) date integer(10) Character varying(255) Character varying(255) integer(10)
tarif_puissance	<u>Id</u> Tarif Puissance pmd_max #tarifid	integer(10) Character varying(255) integer(10) integer(10) integer(10)
tarif_prix_unitaire	<u>#tarif_versionid</u> <u>#tarif_rebriqueid</u> <u>#tarif_trancheid</u> prix_unitaire	integer(10) integer(10) integer(10) integer(10)
tarif_tranche	<u>Id</u> code_tarif_tranche quantite_seuil_max quantite_seuil_min unite_mesure #tarif_tranche_groupeid	integer(10) Character varying(255) numeric(20, 2) numeric(20, 2) Character varying(255) integer(10)
tarif_tranche_groupe	<u>Id</u>	integer(10)

	Libelle ordre_tarif_tranche_groupe unite_mesure	Character varying(255) integer(10) Character varying(255)
tarif_rebrique	<u>Id</u> code_tarif_rebrique libelle numero_ordre unite_mesure #tarif_type_rebriqueid	integer(10) Character varying(255) Character varying(255) Character varying(255) Character varying(255) integer(10)
tarif_type_rebrique	<u>Id</u> code_tarif_type_rebrique libelle	integer(10) Character varying(255) Character varying(255)
tarif_rebrique_tarif_poste_horaire	#tarif_rebriqueid #tarif_poste_horaireid	integer(10) integer(10)
tarif_poste_horaire	<u>Id</u> code_poste_horaire libelle_poste_horaire	integer(10) Character varying(255) Character varying(255)
tarif_plage_horaire	<u>Id</u> heure_debut heure_fin minute_debut minute_fin tarif_poste_horaireid	integer(10) integer(10) integer(10) integer(10) integer(10) integer(10)
wilaya	<u>Id</u> Code libelle	integer(10) Character varying(255) Character varying(255)

Tableau 7:le le modèle relationnel de parametrage.