Table des matières

DEDICACE 3

REMERCIEMENTS 4

TABLE DES MATIERES 5

LISTE DES ABREVIATIONS 7

TABLE DES FIGURES 8

LISTE DES TABLEAUX 11

خلاصة 12

RESUME 13

ABSTRACT 14

INTRODUCTION GENERALE 15

I. CONTEXTE GENERAL DU PROJET 17

1. ORGANISME D'ACCUEIL 17
A propos de Z SMART SERVICES *17*
2. MAITRE D'OUVRAGE « LA COMMUNE DE MARRAKECH » 18
3. PRESENTATION DU SUJET 19
4. CONDUITE DU PROJET 20
a. Processus de développement *20*
b. Planification *22*

CONCLUSION 22

II. ETUDE PRELIMINAIRE ET FONCTIONNELLE 24

1. ETUDE PRELIMINAIRE 24
Présentation du projet *24*
2. ETUDE DE L'EXISTANT 25
a. Description de l'existant : *25*
b. Critique de l'existant : *31*
3. LE CAHIER DES CHARGES 34
4. ETUDE FONCTIONNELLE 38
a. Capture des besoins fonctionnels *38*
b. Identification des acteurs *38*

CONCLUSION 40

III. ANALYSE ET CONCEPTION 42

1. ARCHITECTURE DE L'APPLICATION 42
2. MODELES DE CONCEPTION (DESIGN PATTERNS) 43
Définition *43*
3. LE MODELE MVC 43
a. MVC Modèle 1 ..... *43*
b. MVC Modèle 2 ..... *44*
c. Application du MVC Modèle 2 avec les composants du Framework JSF *44*
d. Le formalisme UML *46*
e. Le choix UML *46*
4. DIAGRAMME DE CAS D'UTILISATION 47
a. Description détaillée des cas d'utilisation *47*
5. DIAGRAMMES DE SEQUENCES 68

A. DIAGRAMME DE CLASSES 75

CONCLUSION 91

6. OUTILS TECHNIQUES 93

A. PLATEFORME JEE 93
i. Pourquoi la technologie JEE ? 93
ii. Architecture multi-tiers 93
iii. L'architecture JEE 94
B. DEFINITION DE FRAMEWORK 95
C. LE FRAMEWORK JSF 95
i. Principes 95
ii. Cycle de vie 96
iii. PrimeFaces 96
iv. Framework Facelets 97
D. LES ENTREPRISES JAVABEAN 97
E. SPECIFICATION JPA 99
Nouveauté de JPA 2.0 100
F. ENVIRONNEMENT TECHNIQUE 101
i. Git 101
ii. Netbeans 101
iii. Xampp 101
iv. GlassFish 101
v. Enterprise Architect 102
vi. Jasper Reports 102
vii. iText 102
viii. Gantt Project 102
ix. Trello 102
x. Balsamiq mockups 102

CONCLUSION 103

7. MISE EN ŒUVRE 105

A. LES DIFFICULTES RENCONTRES 105
i. Les difficultés au niveau de la conception 105
ii. Les difficultés au niveau technique 105
B. ECRANS DE L'APPLICATION 106
i. Page de connexion à e-Taxe 106
ii. Vue d'un régisseur communal 107
iii. Vue d'un responsable 108
iv. Vue d'un administrateur 108

CONCLUSION 132

CONCLUSION GENERALE ET PERSPECTIVES 133

BIBLIOGRAPHIE 134

WEBOGRAPHIE 134

ANNEXE A: INTRODUCTION A XP 135

Liste des abréviations

Acronyme	Détails
UML	Unified Modeling Language
JSF	Java Server Faces
EJB	Enterprise Java Bean
XP	eXtreme Programming
JPA	Java Persistence API
MVC	Model-View-Controller
JEE	Java Enterprise Edition
XML	eXtended Markup Language
LDAP	Lightweight Directory Access Protocol
UC	Use Case
SD	Sequence Diagram
JSP	Java Server Pages
HTML	HyperText Markup Language
GUI	Graphical User Interface
API	Advanced Programming Interface
JDBC	Java DataBase Connectivity
SQL	Structured Query Language
ORM	Object/Relationnal Mapping
CRUD	Create, Read, Update, Delete
JPQL	Java Persistence API
RMI	Remote Method Invocation
DB	DataBase
TNB	Terrains non bâtis
RC	Registre de commerce
TVA	Taxe sur la valeur ajoutée
CIN	Code Identité National

Table des figures

Figure 1 : Organigramme de la société 18
Figure 2 : Cycle de vie des itérations de XP 22
Figure 3 : Diagramme de Gantt du projet 22
Figure 4 : Les points forts d'e-Taxe 24
Figure 5 : Archive des dossiers des locaux dont les activités sont : Hôtellerie et Ryad 32
Figure 6 : Archive des dossiers des locaux dont les activités sont : Café, Snack, Restaurant classés en fonction des secteurs et annexes administratives 33
Figure 7 : Figure qui illustre le risque d'abîmer ou la perte d'un dossier 34
Figure 8 : Fonctionnalités principales d'e-TAXE 38
Figure 9 : Architecture standard d'une application EJB 42
Figure 10 : Schéma de MVC 1 43
Figure 11 : Schéma de MVC 2 44
Figure 12 : Implémentation de MVC2 dans JSF 45
Figure 1 : Diagramme de cas d'utilisation « Gestion redevables » 48
Figure 2 : Diagramme de cas d'utilisation « Gestion des locaux » 49
Figure 3 : Diagramme de cas d'utilisation « Affectation des taxes » 50
Figure 4 : Diagramme de cas d'utilisation « Paiement taxe boisson » 51
Figure 5 : Diagramme de cas d'utilisation « Déclaration d'existence » 52
Figure 6 : Diagramme de cas d'utilisation « Gestion taux boisson » 53
Figure 7 : Diagramme de cas d'utilisation « Paiement taxe séjour » 54
Figure 8 : Diagramme de cas d'utilisation « Paiement taxe 37 » 56
Figure 9 : Diagramme de cas d'utilisation « Paiement taxe 38 » 56
Figure 10 : Diagramme de cas d'utilisation « Paiement taxe TNB » 58
Figure 11 : Diagramme de cas d'utilisation « Gestion terrains » 59
Figure 12 : Diagramme de cas d'utilisation « Déclaration terrain » 60
Figure 13 : Diagramme de cas d'utilisation « Exonération » 61
Figure 14 : Diagramme de cas d'utilisation « Gestion taux TNB » 62
Figure 15 : Diagramme de cas d'utilisation « Gestion notifications » 63
Figure 16 : Diagramme de cas d'utilisation « Gestion utilisateur » 64
Figure 17 : Diagramme de cas d'utilisation « Gestion activité » 65
Figure 18 : Diagrammes des cas d'utilisations de chaque élément de l'adresse 66
Figure 31 : Diagramme de séquence de l'authentification 69
Figure 32 : Diagramme de séquence de la recherche d'un redevable via le site du tribunal commercial 70
Figure 33 : Diagramme de séquence de la création d'un redevable 70
Figure 34 : Diagramme de séquence de la création d'un local 71
Figure 35 : Diagramme de séquence du paiement d'une taxe trimestrielle 72
Figure 36 : Diagramme de séquence « Paiement taxe TNB » 73
Figure 37 : Diagramme de séquence de la préparation des notifications 74
Figure 38 : Diagramme de séquence « Statistiques » 75

Figure 39 : Diagramme de packages 76
Figure 40 : Diagramme de classes « Package Local » 77
Figure 41 : Diagramme de classes « Package boisson » 77
Figure 42 : Diagramme de classes « Gestion taxe boisson » avec les attributs 78
Figure 43 : Diagramme de classes « Package séjour » 79
Figure 44 : Diagramme de classes « Gestion taxe séjour » avec les attributs 80
Figure 45 : Diagramme de classes "Package Taxe 37" 81
Figure 46 : Diagramme de classes « Gestion taxe 37 » avec les attributs 81
Figure 47 : Diagramme « Package Taxe 38 » 82
Figure 48 : Diagramme de classes « Gestion taxe 38 » avec les attributs 83
Figure 49 : Diagramme « Package Taxe TNB » 84
Figure 50 : Diagramme de classes « Gestion taxe TNB » avec les attributs 85
Figure 51 : Diagramme de classe « Package Notification » 86
Figure 52 : Diagramme de classes « Gestion notifications » avec les attributs 87
Figure 53 : Diagramme de classes « Package Commune » 88
Figure 54 : Diagramme de classes « Gestion utilisateurs » avec les attributs 88
Figure 55 : Diagramme "Package Adresse" 89
Figure 56 : Diagramme de classes « Gestion adresses » avec les attributs 90
Figure 57 : Découpage de l'adresse 90
Figure 58 : Diagramme de classes global 91
Figure 59 : L'architecture globale de JEE 94
Figure 60 : Cycle de vie d'une requête JSF 96
Figure 61 : Architecture interne d'un serveur d'application 98
Figure 62 : Communication entre le serveur d'application et les fournisseurs EJB 99
Figure 63 : IHM d'authentification 106
Figure 64 : Menu (vue régisseur "Taxe boisson ") 107
Figure 65 : Menu principal (vue d'un régisseur "Taxe séjour") 107
Figure 66 : Menu principal (vue d'un régisseur "Taxe boisson") 107
Figure 67 : Menu (vue régisseur "Taxe boisson et séjour") 107
Figure 68 : Menu principal (vue d'un régisseur "Taxe boisson et séjour") 107
Figure 69 : Menu « vue responsable » 108
Figure 70 : Menu principal de l'application (Vue responsable) 108
Figure 71 : Menu principal de l'application (Vue administrateur) 108
Figure 72 : Menu (Vue administrateur) 109
Figure 73 : Menu « Local/Redevable » 109
Figure 74 : IHM création d'un redevable 109
Figure 75 : IHM création d'un local 110
Figure 76 : IHM affectation des taxes au local 110
Figure 77: IHM recherche d'un redevable 111
Figure 78 : IHM recherche d'un local 111
Figure 79 : IHM détail d'un local 112
Figure 80 : Menu « Taxe boisson » 112
Figure 81 : IHM Liste des paiements, taxe boisson (vue utilisateur) 113
Figure 82 : IHM critères de recherche (vue régisseur ou responsable) 113
Figure 83 : IHM paiement d'un trimestre « Taxe boisson » 114
Figure 84 : IHM simulation d'un paiement « Taxe boisson » 114
Figure 85 : IHM déclaration annuelle 115
Figure 86 : IHM déclaration d'existence 115

Figure 87 : Menu « Taxe séjour » 116
Figure 88 : IHM liste des déclarations d'existence 116
Figure 89 : IHM paiement d'un trimestre « Taxe séjour » 116
Figure 90 : IHM consultation du paiement « Taxe séjour » 117
Figure 91 : IHM simulation du paiement d'un trimestre « Taxe séjour » 117
Figure 92 : IHM déclaration annuelle « Taxe séjour » 118
Figure 93 : Menu « Taxe 37 » 118
Figure 94 : IHM paiement d'un trimestre « Taxe 37 » 119
Figure 95 : Simulation paiement d'un trimestre « Taxe 37 » 119
Figure 96 : Menu « Taxe 38 » 120
Figure 97 : IHM simulation paiement d'un trimestre « Taxe 38 » 120
Figure 98 : IHM paiement d'un trimestre « Taxe 38 » 120
Figure 99 : Menu « Taxe TNB » 121
Figure 100 : IHM paiement « Taxe TNB » 121
Figure 101 : IHM simulation d'un paiement « Taxe TNB » 121
Figure 102 : IHM de génération d'avis d'imposition « Taxe TNB » 122
Figure 103 : Menu « Utilisateur » 122
Figure 104 : IHM création d'un utilisateur (Régisseur) 122
Figure 105 : IHM création d'un utilisateur (Responsable) 123
Figure 106 : Menu « Statistiques » 123
Figure 107 : Exemple de graphe en barre de la taxe boisson 123
Figure 108 : Exemple de graphe linéaire de la taxe boisson et séjour 124
Figure 109 : Exemple de graphe en camembert des montants collectés en fonction des secteurs 124
Figure 110 : Exemple de graphe en camembert des montants collectés en fonction des taxes 125
Figure 111 : Graphe en camembert des montants collectés en fonction des catégories de séjour 125
Figure 112 : Graphe en camembert des montants collectés par activité 126
Figure 113 : Graphe du suivi de la saisie des taxes par les utilisateurs 126
Figure 114 : Menu « Notification » 127
Figure 115 : IHM préparation des notifications 127
Figure 116 : Menu « Utilitaire » 127
Figure 117 : IHM liste des activités 127
Figure 118 : IHM pour créer une activité 128
Figure 119 : IHM pour modifier une activité 128
Figure 120 : IHM gestion de l'adresse 128
Figure 121 : IHM création d'un secteur 128
Figure 122 : Bordereau de versement « Taxe boisson » 129
Figure 123 : Avis d'imposition « Taxe TNB » 130
Figure 124 : Première notification « Taxe séjour » 131

Liste des tableaux

Tableau 1 : Fiche technique de la société 17
Tableau 2 : Comparaison entre les différentes méthodes agiles 21
Tableau 3 : Taux de la redevance 31
Tableau 4 : Aperçu des droits d'administration 39
Tableau 1 : Description du cas d'utilisation « Gestion redevables » 48
Tableau 2 : Description du diagramme de cas d'utilisation « Gestion locaux » 49
Tableau 3 : Description du diagramme de cas d'utilisation « Affectation des taxes » 50
Tableau 4 : Description du cas d'utilisation « Paiement taxe boisson » 52
Tableau 5 : Description du cas d'utilisation « Déclaration d'existence » 53
Tableau 6 : Description du cas d'utilisation « Gestion taux boisson » 54
Tableau 7 : Description du diagramme de cas d'utilisation « paiement taxe séjour » 55
Tableau 8 : Description des diagrammes des cas d'utilisation « Paiement taxe 37 » et « Paiement taxe 38 » 57
Tableau 9 : Description du cas d'utilisation « Paiement taxe TNB » 59
Tableau 10 : Description du cas d'utilisation « Gestion des terrains » 60
Tableau 11 : Description du cas d'utilisation « Déclaration des terrains » 61
Tableau 12 : Description du cas d'utilisation « Exonération » 62
Tableau 13 : Description du diagramme de cas d'utilisation « Gestion taux TNB » 63
Tableau 14 : Description du diagramme « Gestion notification » 64
Tableau 15 : Description du diagramme de cas d'utilisation « Gestion utilisateur » 65
Tableau 16 : Description du diagramme de cas d'utilisation « Gestion activité » 66
Tableau 17 : Description des diagrammes de cas d'utilisation « Gestion adresse » 68
Tableau 22 : Classes du système par package 76

Rapport-Gratuit.com

خلاصة

في ظل النقلة النوعية التي يشهدها العالم في مجال استخدام التكنولوجيا الحديثة والسير نحو مجتمع تلبي فيه التقنية الحديثة جميع المتطلبات في مختلف القطاعات، عملنا على ارساء الاسس السليمة لتطبيق استراتيجية للعمل الالكتروني.

وسعيا منا لتحقيق هذه الاستراتيجية وتوظيف الاساليب التكنولوجية الحديثة وتوجيهها بما يتناسب مع طبيعة العمل في مصلحة الوعاء الضريبي بالجماعة، ويخدم اهدافها ويعود بالنفع والفائدة على المواطن بالدرجة الاولى، قمنا بتصميم نظام ادارة وتسيير مصلحة الوعاء الضريبي على درجة عالية من الدقة والكفاءة ليستجيب لأرقى معايير الجودة والتنافسية.

يهدف البرنامج الى احداث تطوير شامل في هذه المصلحة ورفع كفاءة وفعالية ادائها وتعزيز ادارتها ورفع مستوى الخدمات وجودة الوثائق المقدمة للمواطن من خلال رصد جميع قسائم الإيداع المضمنة بسجلات الوعاء الضريبي وجردها وتخزينها الكترونيا لتسهيل البحث وتسريع الولوج واصدار اي وثيقة. ويوفر البرنامج كذلك مؤشرات احصائية وتحليلية دقيقة وسريعة مما يتيح معرفة وضعية المدفوعات مع إمكانية إعداد الجداول السنوية وأيضا إنشاء الاقرارات والامر بالمداخيل.

Résumé

A la lumière de la révolution technologique qui se déroule dans le monde contemporain, et pour aller vers une société où la technologie répond à toutes les exigences modernes dans divers secteurs, Z SMART SERVICES a pensé à établir des bases solides pour l'application de la stratégie e-business.

Afin d'atteindre cette stratégie et d'utiliser les méthodes technologiques modernes et correspondantes à la nature du travail dans les bureaux de la commune de Marrakech, et servir ses objectifs pour apporter des avantages et de l'intérêt au citoyen d'abord, nous avons conçu un système d'administration et de gestion des services des taxes communales à un degré élevé de précision et d'efficacité pour répondre aux normes les plus élevées de la qualité et la compétitivité.

L'application e-TAXE vise à favoriser le développement global de ce service, et augmenter l'efficacité de sa performance en renforçant sa gestion, en élevant le niveau de service et la qualité des documents fournis aux citoyens. En effet, l'application prend en charge tous les bordereaux compris dans les registres de ce département, les inventorie et les stocke électroniquement. Cela permettra, d'un côté, de faciliter la recherche d'y accélérer l'accès, et d'un autre coté, de fluidiser la délivrance, le traitement et la gestion des paiements des taxes. Aussi, l'application opère tout en respectant la loi relative à la fiscalité des collectivités locales. Outre cela, e-TAXE fournit des indicateurs analytiques et statistiques précis et rapides permettant de connaitre et de suivre la situation du paiement et de la saisie des taxes par chaque utilisateur, la possibilité de préparer des tableaux annuels et aussi d'établir les notifications et les ordres de recettes.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Abstract

In the light of the technological revolution-taking place in the contemporary world and to move towards a society where technology meets all modem requirements in various sectors, we have worked to build a solid foundation for the implementation of the strategy e-business. To achieve this strategy and to use of modern and appropriate to the nature of work in the offices of community technological methods, and serve its purpose to bring benefits and interest to the citizen first. We have designed a system of administration and management of the municipal tax services with a high degree of exactness and efficiency to meet the highest standards of quality and competitiveness.

The e-TAXE application aims to foster the overall development of this service and increase the efficiency of its performance by strengthening its management, raising the level of service and the quality of documents provided to citizens. Indeed, the application supports all the slips included in the registers of this department, inventories them and stores them electronically. On the one hand, this will make it easier to seek to accelerate access to it and, on the other hand, to streamline the delivery, processing and management of tax payments. Also, the application operates while respecting the law on taxation of local authorities. In addition, e-TAXE provides accurate and timely analytical and statistical indicators that allow users to know and monitor the payment status and tax entry by each user, the ability to prepare annual tables and to establish notifications and the revenue orders.

Introduction générale

La taxe communale (ou impôt communal) peut être définie comme le prélèvement réalisé par voie d'autorité par la commune sur les moyens des personnes physiques et morales ou des associations se situant sur son territoire ou y ayant des intérêts, afin de l'utiliser à des services d'utilité publique.

Cet impôt est donc un transfert fait à la commune. Il est obligatoire et établi unilatéralement. Le contribuable est tenu de payer. L'impôt répond aussi au principe d'unité de caisse, il est affecté aux services d'utilité générale et représente un moyen de financement important des activités.

Le but de notre projet est d'informatiser le système de gestion du service d'assiette de l'impôt au sein de la commune de Marrakech. Ce service comprend la gestion de la taxe sur les débits des boissons, la taxe de séjour, la redevance d'occupation temporaire du domaine public communal pour un usage commercial, industriel ou professionnel, appelé taxe 37 , la redevance d'occupation temporaire du domaine public communal par les biens meubles ou immeubles liés à l'exercice d'un commerce, d'une industrie ou d'une profession, appelé taxe 38 et la taxes sur les terrains urbains non bâtis. Notre système permet aussi l'impression des différents bordereaux de versement, des avis d'imposition, et des notifications.

Le développement du projet e-TAXE se base sur la plateforme Java EE, avec l'usage du Framework JSF.

Le contenu de ce présent rapport est organisé comme suit :

Chapitre 1 intitulé « Contexte général du projet », qui présentera l'entreprise accueillante, le sujet de stage, la problématique traitée et les objectifs visés, suivis de la conduite adoptée pour le bon déroulement du projet.

Chapitre 2 intitulé « Etude préliminaire », qui présente la phase de l'étude de l'existant qui consiste à effectuer une étude critique pour en détecter les défaillances.

Chapitre 3 intitulé « Analyse et conception », ce chapitre est consacré à la partie conception de l'application. Après une brève présentation du langage de conception UML, le chapitre présente les diagrammes de conception adoptés lors de la phase d'élaboration.

Chapitre 4 intitulé « Outils techniques », qui projette l'ensemble des technologies utilisées lors de la phase de développement.

Chapitre 5 intitulé « Mise en œuvre », qui décrit le travail réalisé sous forme de prises d'écran de l'application développée ainsi que les difficultés rencontré avant de conclure et suggérer quelques perspectives.

CHAPITRE 1 :

Contexte général du projet

- Organisme d'accueil
- Rappel du sujet de stage
- Objectifs et résultats
- Conduite du projet

Ce premier chapitre décrit le contexte général du projet. Nous allons présenter en première section de ce chapitre l'organisme d'accueil, **Z SMART SERVICES**, puis nous présenterons le cadre dans lequel s'inscrit notre projet, les objectifs à atteindre durant le stage ainsi que la démarche suivie pour mener à bien notre travail.

I. Contexte général du projet

1. Organisme d'accueil

A propos de Z SMART SERVICES

Z SMART SERVICES est une société marocaine, spécialisée dans le développement dans le domaine d'informatique, y compris l'hébergement Web, la création de sites Internet, le référencement, la conception d'interfaces, la réalisation d'animations en flash, le développement d'applications personnalisées et la maintenance informatique. Son siège se trouve à Marrakech.

La société représente un groupe de professionnels en informatique ingénieurs, techniciens spécialisés, analystes programmeurs et Webmasters avec une expérience de plus d'un an dans ce domaine. La mission de la société est d'accompagner ses clients depuis l'identification des besoins jusqu'à la mise en œuvre et l'intégration de la solution quelques soit ces besoins : Hébergement, logiciels, installation de réseau, formations, et quelques soit l'activité du client : Entreprise commerciale, hôtel, pharmacie, cabinet comptable, travaux publiques, ... Toutes les solutions informatiques de la société sont évolutives selon les besoins et sont régulièrement mis à jour.

Fiche technique de la société

Dénomination	Z SMART SERVICES
Date d'implantation	2015
Siège social	11 Résidence bab lmanar Lot Maral Lahlou, Marrakech.
Forme juridique	Société à responsabilité limitée (S.A.R.L)
Secteur d'activité	Informatique
Capitale	100.000 DHs
Directeur général	M. Rachid SMAITI
Téléphone	+212 / (0) 524059716
Email	zsevent@gmail.com

Tableau 1 : Fiche technique de la société

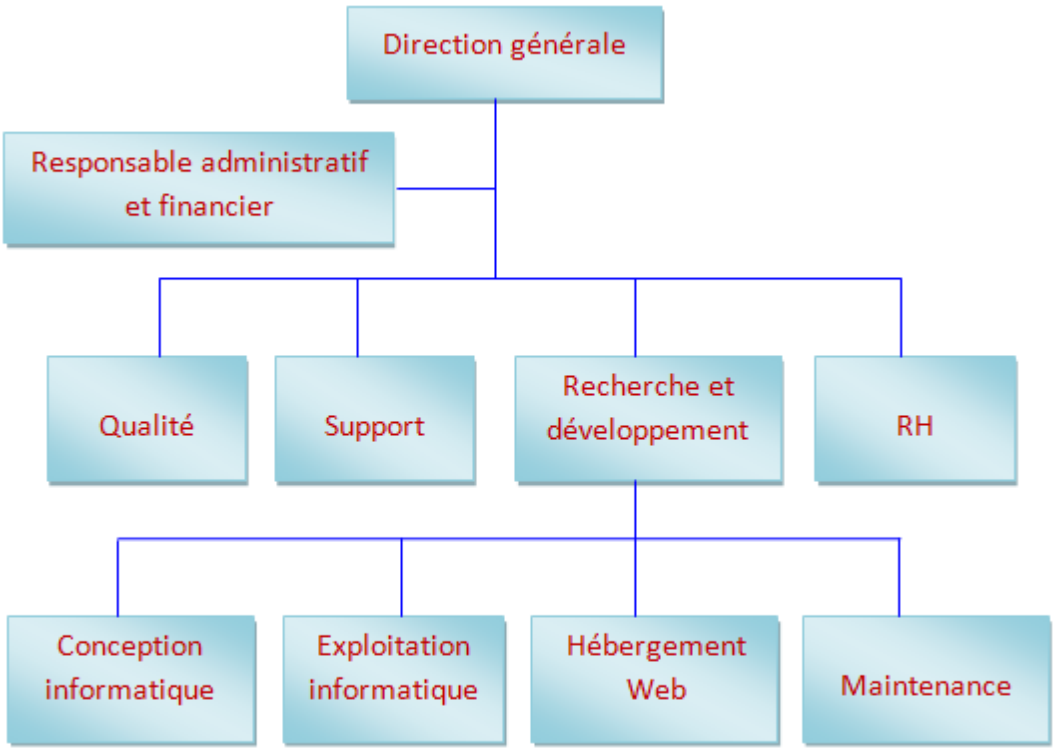


Figure 1 : Organigramme de la société

Notre mission dans le projet e-TAXE consiste d'une part à collecter les informations relatives à l'organisation de chaque taxe à travers des réunions avec le responsable du service d'assiette ainsi que quelque personnel du même service, et en se basant aussi sur les articles de la loi d'autre part, à effectuer l'analyse conceptuelle, et enfin la réalisation de la solution. C'est pour cela nous avons opéré dans le pôle de la « Recherche et développement » et plus précisément dans la « Conception informatique » et «l'Exploitation informatique » au sein d'une équipe de 3 personnes.

2. Maitre d'ouvrage « La commune de Marrakech »

Commune de Marrakech

Selon le dahir du 1er septembre 1959, la commune est une collectivité territoriale de droit public, dotée de la personnalité morale et de l'autonomie financière.

Notre application cible le personnel du service de l'assiette communal de Marrakech dans le but de couvrir les différents services offerts par ce dernier.

3. Présentation du sujet

Notre projet de fin d'études intitulé « e-TAXE » est une application web développée en JEE, pour la gestion des taxes communales.

1- e-Taxe remplit intégralement les conditions du cahier de charges des taxes communales :

- Respect de la loi N° 47-06 relative à la fiscalité des collectivités locales promulguée par le Dahir 1-07-195 du Kaada 1428 (30 novembre 2007)
- Respect de la loi N° 30-89 relative à la fiscalité des collectivités locales et de leurs groupements promulguée par le Dahir n° 1-89-187 du 21 rebia II 1410 (21 novembre 1989)
- Respect des différents modèles de la loi.
- Paiement des taxes ou déclaration annuelles des recettes.
- Notification des redevables et génération des ordres de recettes.
- Mise en place de différentes statistiques.

2- e-Taxe intègre des fonctionnalités avancées :

- Un accès sécurisé à la base de données.
- La gestion des privilèges des utilisateurs de l'application.
- Une base de données illimitée.
- Application évolutive avec les nouvelles lois.
- La possibilité de sauvegarder la base de données sur des supports externes afin de garantir des copies de la base de données pour la restauration dans le cas de sinistre ou de panne majeure du serveur accueillant la base de données.
- Interface ergonomique et intuitive.
- En-tête adaptable de la commune par rapport aux différents documents.
- Les bordeaux de versements, déclarations annuelles, déclaration d'existence, les lettres de notifications et les ordres de recettes sont implémentés au niveau de l'application pour qu'ils puissent être générés automatiquement en gardant les canevas qu'utilisait la commune.

- La sécurité dans ses deux axes : Système (accès sécurisé, technologie prouvée et mature, respect des standards lors de la programmation) et données (sauvegarde et récupérations de la base de donnée en cas de panne).

4. Conduite du projet

a. Processus de développement

Un mauvais choix du processus de développement peut conduire à un échec d'un projet. C'est pourquoi nous avons dressé une liste non exhaustive des différentes méthodes agiles utilisées dans un projet informatique. Le tableau 1 résume cette étude.

	Description	Points forts	Points faibles
RUP (Rational Unified Process)	✓ Méthodologie centrée sur l'architecture et couplée aux diagrammes UML ✓ Cible des projets de + de 10 personnes. ✓ Processus complet assisté par des outils exhaustifs.	✓ Itératif ✓ Spécifie le dialogue entre les différents intervenants du projet : les prototypes… ✓ Propose des modèles de documents et des canevas pour des projets types. ✓ Rôles bien définis, modélisation.	✓ Couteux à personnaliser. ✓ Très axé processus, au détriment du développement. ✓ Lourd, largement étendu, il peut être difficile à mettre en œuvre de façon spécifique. ✓ Convient pour les gros projets qui génèrent beaucoup de documentation.
XP (eXtreme Programming)	✓ Développement guidé par les besoins du client. ✓ Equipes réduites, centrés sur les développeurs (Binômes)	✓ Itératif ✓ Simple à mettre en œuvre ✓ Fait une large place aux aspects techniques. ✓ Builds journaliers ✓ Amélioration constante, adaptation aux modifications.	✓ Ne couvre pas les phases en amont et en aval de développement. ✓ Assez flou dans sa mise en œuvre : quels intervenants, quel livrables ? ✓ On se focalise sur l'aspect individuel du développement, au détriment d'une vue globale
FDD (Feature Driven Development)	✓ Application très «light » de RUP qui ne se concentre que sur les phases de design et de réalisation.	✓ Itératif ✓ Les rôles de chaque intervenant sont bien clarifiés.	✓ La mise en œuvre du développement n'est pas précisée, seule compte la gestion des ressources humaines

	✓ Petites équipes, itérations de 30 jours, réunions journalières.		
2TUP (Two Track Unified Process)	✓ S'articule autour de l'architecture. ✓ Propose un cycle de développement en Y. ✓ Cible des projets de toutes tailles	✓ Itératif et fait une large place à la technologie et à la gestion du risque. ✓ Définit des profils des intervenants, les livrables, les plannings, les prototypes.	✓ Plutôt superficiel sur les phases situées en amont et en aval du développement. ✓ Ne propose pas de document type.

Tableau 2 : Comparaison entre les différentes méthodes agiles

Le premier constat que nous pouvons faire est que ces méthodologies proposent de travailler de manière itérative. S'il y a une insistance sur l'itératif, c'est qu'il réduit la complexité des phases en travaillant par étapes successives et incrémentales.

Suite au cahier de charges retenu pour le projet et par constat de la subdivision de ce dernier en plusieurs sous tâches et dû au fait que ces tâches sont de courtes durées, nous avons opté pour la méthode XP. Les conditions pour utiliser cette méthode ne sont pas toutes réunies mais la majorité (celles décrites plus haut) nous a fait pencher sur ce choix.

Faisant partie des méthodes dites agiles, XP vise à réduire le cycle de vie du logiciel tout en accélérant son développement par la mise en place d'une version minimale puis en intégrant les fonctionnalités par un processus itératif basé sur l'avènement de nouvelles fonctionnalités.

(Voir annexe A).

L'origine de ces méthodes est l'instabilité technologique et le manque d'une spécification exhaustive des besoins. Le terme «agile» fait référence à la capacité d'adaptation aux changements de contextes et aux modifications des spécifications pendant le processus de développement.

La méthode XP fournit des pratiques permettant de développer dans des conditions optimales. Elle est basée sur les principes suivants :

- ✓ Le cycle de travail est court d'une durée de deux à trois semaines
- ✓ Les livraisons des versions sont à une fréquence élevée

Rapport-Gratuit.com

✓ L'équipe de développement travaille sur la base de binômes

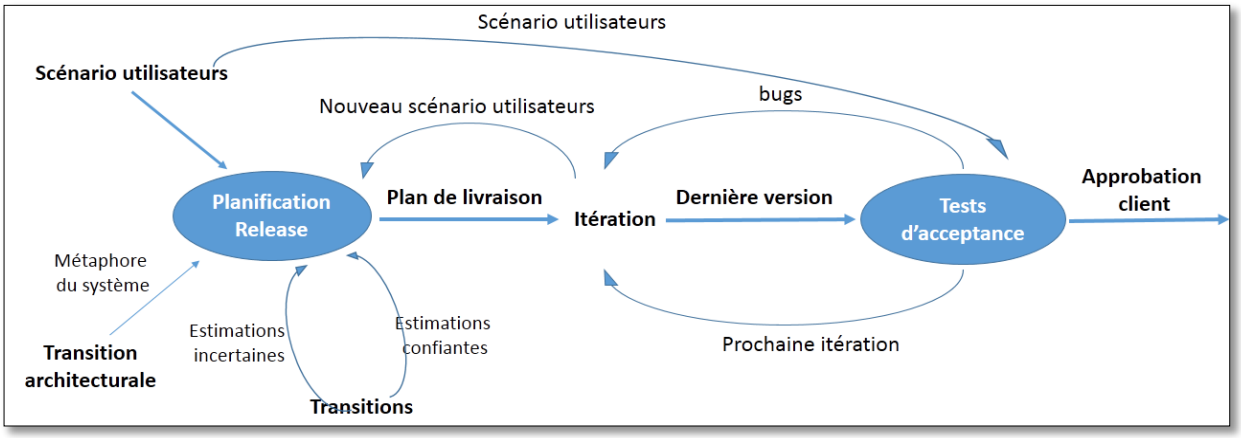


Figure 2 : Cycle de vie des itérations de XP

b. Planification

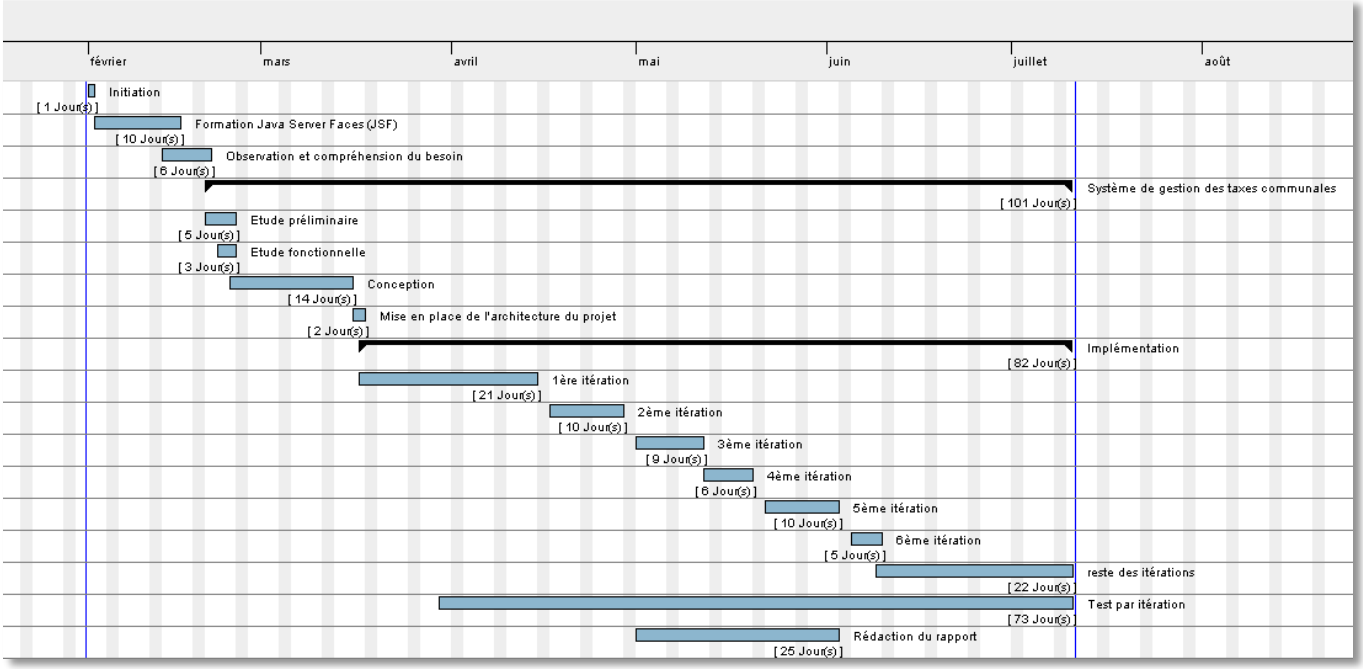


Figure 3 : Diagramme de Gantt du projet

Conclusion

Après avoir déterminé le contexte général du projet et élaboré un planning prévisionnel pour mettre en œuvre ce dernier, il convient maintenant d'entamer une analyse et spécification des besoins, avant d'aboutir à la conception et à la réalisation de la solution.

CHAPITRE 2 :

Etude préliminaire

- Etude de l'existant
- Le cahier des charges
- Capture des besoins fonctionnels
- Les besoins non fonctionnels

Ce chapitre présente l'étude préliminaire du projet, qui consiste à effectuer un premier repérage des besoins fonctionnels et techniques du projet.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

II. Etude préliminaire et fonctionnelle

1. Etude préliminaire

L'étude préliminaire (ou Pré-étude) est la toute première étape du processus XP. Elle consiste à effectuer un premier repérage des besoins fonctionnels et opérationnels, en utilisant principalement le texte, ou diagrammes très simples. Elle prépare les activités les plus formelles de captures des besoins fonctionnels et de capture techniques.

Présentation du projet

Notre projet de fin d'étude est intitulé e-TAXE. Il est proposé par la société Z SMART SERVICES pour la gestion des taxes communales en assurant le stockage, la consolidation et l'exploitation rentable des données comme le montre la « Figure 4 » ci-dessus.

Du point de vue fonctionnel, e-TAXE couvre de façon intégrale le besoin fonctionnel exprimé au niveau des bureaux de la commune tout en adoptant la même démarche du travail au sein de ces bureaux, ce qui minimise l'effort d'adaptation de l'ancien système d'information au nouveau. Ajoutons à cela un retour d'expérience qui a rendu le logiciel en respect de tous les textes juridiques édités dans le domaine.

Figure 4 : Les points forts d'e-Taxe

2. Etude de l'existant

a. Description de l'existant :

Dans cette section, nous allons définir chaque taxe en précisant les articles des lois qui organisent chacune d'eux.

- **La loi n° 47-06 relative à la fiscalité des collectivités locales :**

<u>Taxe sur les débits de boissons :</u>

Section 1 : Champ d'application

Article 64 : ***Personnes imposables***
La taxe sur les débits de boissons est due par les exploitants des cafés, bars et salons de thé et de manière générale, par tout débitant de boissons à consommer sur place.

Section 2 : Base imposable

Article 65 : ***Détermination de la base imposable***
La taxe sur les débits de boissons est assise sur les recettes, hors taxe sur la valeur ajoutée, réalisées sur la vente des boissons à consommer sur place et effectuées par les exploitants des établissements soumis à la taxe.

Section 3 : Liquidation de la taxe

Article 66 : ***Taux***
Le taux de la taxe est fixé, dans les formes et conditions prévues à l'article 168 ci-dessous, de 2% à 10% des recettes, hors taxe sur la valeur ajoutée, réalisées par l'établissement.

Section 4 : Obligations des redevables

Article 67 : D*éclarations d'existence et de recettes*
I. Les redevables sont tenus de déposer dans un délai maximum de trente (30) jours suivant la date du début de l'activité une déclaration d'existence auprès du service d'assiette de la commune dont relève ladite activité, établie selon un imprimé-modèle de l'administration.
II. - Les redevables sont tenus de déposer avant le premier avril de chaque année, auprès du service d'assiette communal, une déclaration des recettes, hors taxe sur la valeur ajoutée, réalisées au cours de l'année écoulée. Le montant de la taxe est versé spontanément à la caisse du régisseur communal trimestriellement avant l'expiration du mois suivant chaque trimestre, sur la base des recettes, hors taxe sur la valeur ajoutée, réalisées au cours de cette période et au vu d'un bordereau de versement établi selon un imprimé-modèle de l'administration.

<u>Taxe de séjour :</u>

Section 1 : - Champ d'application

Article 70 : ***Personnes et activités imposables***
La taxe de séjour est perçue dans les établissements d'hébergement touristiques appartenant à des personnes morales ou physiques et vient en sus du prix de la chambre.
On entend par « établissements d'hébergement touristiques » au sens du présent chapitre, les hôtels qui offrent en location des chambres ou des appartements équipés et meublés à une clientèle de passage ou de séjour, les clubs privés, les motels, les villages de vacances, les résidences touristiques, les maisons d'hôtes, les centres, ou palais des congrès, et tout établissement touristique portant statut des établissements touristiques.

Section 2 : Base imposable

Article 72 : ***Détermination de la base imposable***
La taxe est due par personne et par nuitée selon les tarifs fixés pour les différentes catégories d'établissements d'hébergement touristiques.

Section 3 : Liquidation de la taxe

Article 73 : ***Tarif***
Les tarifs de la taxe sont fixés, dans les formes et conditions prévues à l'article 168 ci-dessous, comme suit :
a) Maisons d'hôtes, centres ou palais de congrès et hôtels de luxe de 15 à 30 DH ;
b) Hôtels :
- 5 étoiles ..... de 10 à 25 DH ;
- 4 étoiles..... de 5 à 10 DH ;
- 3 étoiles..... de 3 à 7 DH ;
- 2 et 1 étoile.....de 2 à 5 DH ;
c) Clubs privés.....de 10 à 25 DH ;
d) Villages de vacances..... de 5 à 10 DH ;
e) Résidences touristiques..... de 3 à 7 DH ;
f) Motels, gîtes, relais et autres établissements touristiques.....de 2 à 5 DH.

Section 4 : Obligations des redevables

Article 74 : ***Déclaration du nombre de clients et de nuitées***
Les exploitants des établissements d'hébergement touristiques sont tenus de déposer avant le premier avril de chaque année une déclaration auprès du service d'assiette communal, d'après un imprimé-modèle établie par l'administration, comportant le nombre de clients ayant séjourné dans l'établissement pendant l'année écoulée ainsi que le nombre de nuitées.

Article 76 : ***Versement de la taxe***
Les exploitants des établissements d'hébergement touristiques, visés à l'article 70 ci-dessus, sont responsables du recouvrement de la taxe de séjour auprès des clients.
Les factures établies doivent faire apparaître distinctement le montant de la taxe.
Le montant de la taxe est versé spontanément à la caisse du régisseur communal, trimestriellement, avant l'expiration du mois suivant chaque trimestre, sur la base du nombre de clients ayant séjourné dans l'établissement et du nombre de nuitées et au vu d'un bordereau de versement, établi selon un imprimé modèle de l'administration.

Taxe sur les terrains urbains non bâtis :

Section 1 : Champ d'application

Article 39 : *Biens imposables*
La taxe sur les terrains urbains non bâtis porte sur les terrains urbains non bâtis situés à l'intérieur des périmètres des communes urbaines et les centres délimités disposant d'un document d'urbanisme, à l'exclusion des terrains nus affectés à une exploitation professionnelle ou agricole de quelque nature qu'elle soit dans la limite de cinq (5) fois la superficie des terrains exploités.

Article 40 : *Personnes imposables*
La taxe est due par le propriétaire et, à défaut de propriétaire connu, par le possesseur.
En cas d'indivision, la taxe est établie dans l'indivision à moins que chaque co-indivisaire ne demande que la taxe soit établie séparément sur sa quote-part. Dans ce cas les co-indivisaires restent solidairement tenus du paiement de la totalité du montant de la taxe.

Article 42 : E*xonérations totales temporaires*
Sont exonérés temporairement de la taxe sur les terrains urbains non bâtis :
- les terrains faisant objet d'une autorisation de lotir ou de construire pour une durée de trois (3) années à compter du premier janvier de l'année qui suit celle de l'obtention de l'autorisation de lotir ou de construire ;
- les terrains appartenant à des personnes physiques ou morales, qui font l'objet d'une autorisation d'aménagement ou de développement durant les périodes suivantes :
* trois (3) ans pour les terrains dont la superficie ne dépasse pas 30 hectares
* cinq (5) ans pour les terrains dont la superficie est supérieure à trente (30) hectares et ne dépassant pas cent
(100) hectares ;
* sept (7) ans pour les terrains dont la superficie est supérieure à cent (100) hectares.
Toutefois, à l'expiration des délais précités, le redevable qui n'a pas obtenu le certificat de conformité ou le permis d'habiter est tenu au paiement de la taxe due sans préjudice de l'application des pénalités et majorations prévues par les articles 134 et 147 ci-dessous.

Section 2 : Base imposable

Article 43 : *Détermination de la base imposable*
La taxe est assise sur la superficie du terrain au mètre carré. Chaque fraction de mètre carré étant comptée pour un mètre carré entier.

Section 3 : Liquidation de la taxe

Article 44 : *Annualité de la taxe*
La taxe sur les terrains urbains non bâtis est due pour l'année entière à raison des faits existants au premier janvier de l'année d'imposition.

Article 45 : *Tarif*
Les tarifs de la taxe sur les terrains urbains non bâtis sont fixés, dans les formes et conditions prévues à l'article 168 ci-dessous, comme suit :
- zone immeuble ..... 4 à 20 DH/m2 ;
- zones villa, zone logement individuel et autres zones ..... 2 à 12 DH/m2.
La taxe dont le montant est inférieur à cent (100) dirhams ne fait l'objet ni d'émission ni de paiement.

Article 46 : *Paiement de la taxe*
La taxe sur les terrains urbains non bâtis est payée spontanément à la caisse du régisseur communal avant le premier mars de chaque année.

Section 4 : Obligations des redevables

Article 47 : *Déclaration des terrains*
Les propriétaires ou les possesseurs des terrains urbains non bâtis soumis à la taxe ou exonérés doivent déposer avant le premier mars de chaque année une déclaration desdits terrains au service d'assiette communal, établie sur ou d'après un imprimé-modèle de l'administration, faisant ressortir tous les éléments de liquidation de la taxe.

Article 134 : *Sanctions pour défaut de déclaration, déclaration déposée hors délai ou suite à rectification*
I. - En cas de défaut de déclaration ou de déclaration déposée hors délai, le montant de la taxe exigible est majoré de 15%.
Toute déclaration incomplète, ou comportant des éléments discordants est assortie d'une majoration de 15% du montant de la taxe exigible sauf si les éléments manquants ou discordants sont sans incidence sur la base de la taxe ou sur son recouvrement.
Le montant de chacune des majorations prévues ci-dessus ne peut être inférieur à cinq cents (500) dirhams.
II. - En cas de rectification de la base d'imposition résultant de la déclaration, une majoration de 15% est établie sur le montant des droits correspondant à cette rectification sans préjudice de l'application des sanctions prévues à l'article 147 ci-dessous.
III. - Le complément de la taxe et les sanctions visées ci-dessus sont émis par voie de rôle.
Le taux de la majoration de 15% prévu ci-dessus peut être porté jusqu'à 100% quand la mauvaise foi du redevable est établie.
La majoration de 100% est applicable avec un minimum de cent (100) dirhams, sans préjudice de l'application de la pénalité et de la majoration prévues à l'article 147 ci-dessous.

Article 147 : *Sanctions pour paiement tardif de la taxe*
Une pénalité de 10% et une majoration de 5% pour le premier mois de retard et de 0,50% par mois ou fraction de mois supplémentaire est applicable au montant :
- des versements effectués spontanément, en totalité ou en partie, en dehors du délai prescrit, pour la période écoulée entre la date d'exigibilité de la taxe et celle du paiement.
Toutefois, ces majorations et pénalité ne s'appliquent pas à la taxe d'habitation ou à la taxe de services communaux lorsque le montant de la cote ou de la quote-part de la taxe exigible figurant au rôle n'excède pas mille (1000) dirhams pour chacune des deux taxes ;
- des impositions émises par voie de rôle ou ordre de recettes suite à rectification de la base d'imposition résultant de la déclaration, pour la période écoulée entre la date d'exigibilité de la taxe et celle de l'émission du rôle ou de l'ordre de recettes.
Par dérogation aux dispositions ci-dessus, les majorations prévues au présent article ne s'appliquent pas pour la période située au-delà des vingt-quatre mois (24) écoulés entre la date de l'introduction du recours du redevable devant la commission locale de taxation prévue à l'article 225 du Code Général des Impôts et celle de la mise en recouvrement du rôle ou de l'ordre de recettes comportant le complément de taxe exigible.

Pour le recouvrement du rôle ou de l'ordre de recettes, il est appliqué une majoration de 0,50% par mois ou fraction de mois de retard écoulé entre le premier du mois qui suit celui de la date d'émission du rôle ou de l'ordre de recettes et celle du paiement de la taxe.

Article 168 : ***Fixation des taux par arrêté***
Lorsque la présente loi ne détermine pas de taux ou de tarifs d'imposition fixes pour les taxes, qui y sont visées, ces tarifs et taux sont fixés par arrêté pris par l'ordonnateur de la collectivité locale concernée après approbation du conseil de ladite collectivité.
Toutefois, lorsque l'ordonnateur refuse ou s'abstient de prendre l'arrêté fixant les taux et tarifs des taxes et que ce refus ou cette abstention a pour effet de se soustraire aux dispositions de la présente loi ou de nuire à l'intérêt général de la collectivité, le ministre de l'Intérieur ou la personne déléguée par lui à cet effet, pour les régions, préfectures, provinces et communes urbaines et le gouverneur ou la personne déléguée par lui à cet effet, pour les communes rurales, peut après l'avoir requis fixer d'office les taux et tarifs de ces taxes.

Les notifications :

Section VII : Procédure De Rectification Des Impositions

- L'agent, désigné à cet effet par l'ordonnateur, vérifie les déclarations avant ou après l'établissement de l'ordre de recettes.

1 - Dans le cas où il est amené à rectifier les impositions initiales que celles-ci résultent des déclarations souscrites par le redevable ou d'une taxation d'office, il notifie à celui-ci par lettre recommandée avec accusé de réception les motifs, la nature et le montant détaillé du redressement envisagé et l'invite à produire ses observations dans un délai de 30 jours suivant la date de réception de la lettre de notification.
2 - Si les observations du redevable parviennent à l'agent, visé ci-dessus, dans le délai prescrit et si ce dernier les estime non fondées en tout ou en partie, il notifie au redevable, dans un délai maximum de 60 jours suivant la date de réception de la réponse du redevable, les motifs de son rejet partiel ou total ainsi que les impositions qui lui paraissent devoir être retenues en faisant connaître au redevable que ces impositions seront définitives s' il n' introduit pas son recours devant la commission dite, "Commission préfectorale ou provinciale du recours fiscal".

3 - Le recours du redevable est présenté sous forme de requête adressée à la commission visée ci-dessus par lettre recommandée avec accusé de réception dans un délai de 30 jours suivant la date de la notification de l'agent désigné ci- dessus. Le défaut de recours dans les délais précités est considéré comme une acceptation tacite de la décision de l'ordonnateur.

4 - Lorsque le redevable saisit la commission, celle-ci en informe l'ordonnateur concerné par lettre
recommandée avec accusé de réception dans le délai de trois mois suivant la date de réception du recours en lui communiquant la copie de la requête dont elle a été saisie et lui demande de lui transmettre le dossier fiscal afférent à la période litigieuse. Le défaut de communication du dossier fiscal dans le délai prescrit, est considéré comme une acceptation tacite de la décision de la commission.
La commission informe les deux parties de la date à laquelle elle tient sa réunion, 30 jours au moins avant cette date.
5 - Sont immédiatement émis par voie d'ordres de recettes, les droits supplémentaires et les pénalités y afférentes découlant des impositions établies :

*pour défaut de réponse ou de recours dans les délais prescrits par les dispositions des 1, 2, 3 ci- dessus ;

*pour accord partiel ou total conclu par écrit entre les parties au cours de la procédure de rectification ;

*après décision définitive de la commission.

- **La loi n° 30-89 relative à la fiscalité des collectivités locales et de leurs groupements :**

Taxe 37 :

Rubrique 40 10 20/37 : Redevance d'occupation temporaire du domaine public communal pour un usage commercial, industriel ou professionnel.

Article 185 : - La redevance d'occupation temporaire du domaine public communal pour usage commercial, industriel ou professionnel est instituée sur les occupations temporaires du domaine public communal pour l'exercice d'une activité commerciale, industrielle ou professionnelle.

Article 186 : - Le débiteur de la redevance est le bénéficiaire de l'autorisation d'occupation temporaire du domaine public communal.

Article 187 : - La redevance est calculée en tenant compte de la superficie occupée, de l'emplacement et de la nature de l'occupation.

Article 188 : - Le montant de la redevance est payable trimestriellement à la caisse du receveur de la régie communale.
Les trimestres partent du 1er janvier, 1er avril, 1er juillet et 1er Octobre. Tout trimestre commencé est compté pour un trimestre entier.

Taxe 38 :

Rubrique 40 10 20/38 : - Redevance d'occupation temporaire du domaine public communal par les biens meubles ou immeubles liés à l'exercice d'un commerce, d'une industrie ou d'une profession.

Article 189 : - La redevance d'occupation temporaire du domaine public communal pour usage lié à l'exercice d'un commerce, d'une industrie ou d'une profession porte sur les biens meubles et immeubles visés à l'article 192 ci-dessous, s'ils constituent un complément de l'activité commerciale, industrielle ou professionnelle exercée, à titre principal, à l'intérieur d'un local.

Article 190 : - La redevance est due par le bénéficiaire de l'autorisation d'occupation temporaire du domaine public communal.

Article 191 : - La redevance est assise sur la valeur locative qui sert de base de calcul de l'impôt des patentes. La valeur locative est fixée lors du recensement général des propriétés soumises à la taxe d'édilité.

Article 192 : Les taux de la redevance sont fixés comme suit :

Biens meubles et immeubles	Unité de Taxation	Taux de taxation par rapport à la valeur locative
Marquise, paravents des boutiques	mètre carré	0,5%
Store vertical suspendu à une marquise	mètre linéaire	0,20%
Borne	mètre linéaire	0,20%
Store de croisées pour abriter	mètre carré	0,10%
Vitrine	mètre carré	0,50%
Rampe d'illumination	mètre carré	0,10%
Etalage suspendu à la devanture ou au mur de façade	mètre carré	0,10%
Etalage devant la boutique	mètre carré	0,40%
Enseigne et pré enseigne : * Superficie inférieure à 0,5 m²	fixe	150 DH
* Pour le surplus de superficie	mètre carré	0,25%

Tableau 3 : Taux de la redevance

Article 193 : La redevance est recouvrée en même temps et dans les mêmes conditions que la redevance d'occupation temporaire du domaine public communal pour usage commercial, industriel ou professionnel.

b. Critique de l'existant :

Cette partie a pour but de dégager les insuffisances et les défaillances du système actuel relatif à la gestion des taxes communales dont on peut citer :

1- Travaux manuels élevés, lourds et pénibles qui se présentent d'une façon répétitive à savoir l'archivage, la mise en œuvre et la consultation des dossiers.

Rapport-Gratuit.com

Figure 5 : Archive des dossiers des locaux dont les activités sont : Hôtellerie et Ryad

2- Absence d'un outil de recherche rapide : pour chercher un redevable ou un bordereau, le régisseur doit faire une recherche manuelle par secteur et par activité, d'où une perte de temps.

Figure 6 : Archive des dossiers des locaux dont les activités sont : Café, Snack, Restaurant classés en fonction des secteurs et annexes administratives

3- Processus très long avec probabilité de perte des données : puisqu'un dossier englobe un ensemble de bordereaux qui contiennent les paiements d'un redevable de chaque trimestre et de chaque année. Il est possible qu'un bordereau appartient à un tel redevable soit déchiré ou endommagé.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Figure 7 : Figure qui illustre le risque d'abîmer ou la perte d'un dossier

4- Absence de la notion de confidentialité car n'importe quel agent du bureau du service de l'assiette ou du recouvrement peut consulter et même modifier les données du dossier.
5- Encombrement et non clarté des coordonnées d'un redevable qui contient plusieurs informations écrites manuellement.

3. Le cahier des charges

Notre projet consiste à réaliser une solution de gestion des taxes communales. Cette solution de gestion de projet doit être un outil développé sous l'architecture JEE, facile à administrer.

Contraintes :

La contrainte majeure de ce projet se situe en deux volets, le premier est de comprendre le métier des taxes communales. La deuxième contrainte c'est comment adapter notre solution pour qu'elle puisse répondre aux besoins de n'importe quelle commune.

Exigences fonctionnelles :

La solution contient 10 modules principaux :

- **Redevable/local**

A un instant donné un redevable exploite un locale (objet de taxation) pour exercer une activité commerciale/professionnelle ou industrielle. Pour une identification précise des locaux on a jugé pertinent de scinder l'adresse en quatre parties : Secteur (Arrondissement), annexe administrative, quartier, rue, zone et le complément d'adresse.

❖ Taxe boisson

Cette taxe est instituée sur les débits de boissons où se vendent et se consomment les boissons. La consommation des boissons sur le lieu où elles se vendent constitue une condition essentielle à l'imposition de cette taxe.

✓ **Les personnes imposables :**

- Sont assujettis à cette taxe les exploitants des débits de boissons (cafés, bars, salon de thé, etc.)
- Tous débiteurs de boisson consommée sur place
- Le redevable doit payer ses taxes trimestrielles (en se basant sur le chiffre d'affaire réalisé et la date de présentation) et déposer la déclaration annuelle. Ce module contient aussi la gestion des taux trimestrielles/annuelles, les taux de retard et la déclaration d'existence.

❖ Gestion de la taxe de séjour

Cette taxe est établie sur le séjour au sein des établissements d'hébergement touristique.

✓ **Les établissements d'hébergement touristique sont :**

- Les hôtels,
- Les résidences meublées
- Les clubs privés,
- Les pensions et villages de vacances,
- Les résidences touristiques,
- Les Riads, les maisons d'hôte,
- Les centres et les palais des congrès ainsi que tout établissement considéré comme institution touristique.

Un redevable qui occupe un locale marqué comme sujet de la taxe de séjour doit payer ses taxes trimestrielles (en se basant sur le nombre de nuitée réalisé, la date de présentation et la catégorie du locale en question : 0…6 étoile, luxe, maison d'hôte ou Riyad) et déposer la déclaration annuelle. Ce module contient aussi la gestion des taux trimestrielles/annuelles et les taux de retard.

❖ Gestion de la taxe 37

Redevance d'occupation temporaire du domaine public communal pour un usage commercial, industriel ou professionnel. Taxe 37 est une taxe trimestrielle dont la valeur dépend de la catégorie du locale, sa surface ainsi que la valeur locative.

Ce module contient aussi la gestion des taux trimestriels et les taux de retard qui diffèrent d'un secteur à un autre et d'une zone à une autre.

❖ Gestion de la taxe 38

Redevance d'occupation temporaire du domaine public communal par les biens meubles ou immeubles liés à l'exercice d'un commerce, d'une industrie ou d'une profession.

Taxe 38 est une taxe trimestrielle dont la valeur dépend de la catégorie de l'élément adhésive du locale et sa surface.

Ce module contient aussi la gestion des taux trimestriels et les taux de retard.

❖ Gestion de la taxe sur les TNB

La T.N.B est instituée sur la superficie des terrains non bâtis.

Elle s'applique à la superficie non bâtie qui dépasse de cinq fois la superficie couverte par l'ensemble des constructions.

S'il s'agit d'une propriété dans l'indivision, la taxe s'applique à la propriété dans son ensemble. à moins que chacun des propriétaires demande que la taxe soit instituée uniquement sur sa part, dans ce cas tous les propriétaires sont tenus de s'acquitter du montant total de la taxe, à titre solidaire

Sont exclus de l'application de cette taxe les terrains non bâtis réservés à une exploitation agricole ou professionnelle.

- **Les redevables**
  - ✓ La taxe sur les terrains non bâtis est établie au nom du propriétaire. A défaut de propriétaire connu, la taxe est instituée au nom de l'usufruitier.

Ce module contient aussi la gestion des taux et les taux de retard annuel ainsi que les exonérations temporaires.

❖ Gestion des notifications

Les redevables qui n'ont pas payé leur taxe doivent être notifié, en cas de non payement, une deuxième notification doit être envoyés au redevable en question. Si l'état de non-paiement persiste même si le redevable a reçu la deuxième notification, le responsable établira un ordre de recette qui va être envoyé au trésorier communale.

- **Statistiques**

Pour bien suivre le paiement trimestrielle/annuelle des diffèrent taxes la solution doit présenter un système pour comparer l'évolution du paiement des taxes trimestrielles en se basant sur plusieurs critères ainsi que le suivie de la saisie des taxes par le personnel de la commune.

- **Gestion des utilisateurs**

Ce module permet la gestion des utilisateurs de l'application ainsi que leurs droits d'accès, suivis de connexion/déconnexion des utilisateurs ainsi que le blocage d'un utilisateur.

Utilitaires :

- **Gestion des secteurs/annexes administrative/quartiers/rues et zones**

Ajouter/modifier/supprimer les secteurs/quartiers/annexes administrative /rues et les zones.

- **Gestion des activités :** Ajouter/modifier/supprimer les activités.
- **Export de la base de données (Backup)** : Cet utilitaire permet d'exporter la base de donnée pour éviter les problèmes de perte de données.

Exigences techniques :

L'application doit satisfaire les normes de la spécification JSF et doit implémenter le patron de conception MVC (Modèle-Vue-Contrôleur) durant son développement afin de respecter les différentes normes imposés et exigées par la spécification JSF.

- L'environnement de travail : Netbeans.
- Serveur de déploiement : Glassfish4.1.
- Serveur Web locale : XAMP
- Base de données : MySql

Les performances :

1- Sécurité :
   - Les données protégées par un système de cryptage.
   - Hachage du mot de passe en utilisant l'algorithme SHA-256.
   - Vérification des paramètres de la machine.
   - Application du protocole SSL sur la page d'authentification.
2- Simplicité.
3- Rapidité.
4- Ergonomique.
5- Fiabilité.
6- Confidentialité.
7- Une application client léger, ce qu'il faut c'est un appareil et une connexion au réseau local de la commune.

8- Multitude d'utilisateurs dont les droits sont totalement différents.

4. Etude fonctionnelle

a. Capture des besoins fonctionnels

La couverture des exigences fonctionnelles représente un pilier du logiciel, vu que le métier de gestion des taxes communales représente un secteur critique, sa structure connait une complexité inégalée à cause des évolutions des textes juridiques qui ont défini le contexte de ce métier depuis 1989 jusqu'à 2007. La figure suivante résume les axes fonctionnels que traite le logiciel.

.

Figure 8 : Fonctionnalités principales d'e-TAXE

b. Identification des acteurs

Un acteur d'un système est une entité externe à ce système qui interagit (saisie de données, réceptions d'informations,…) avec lui. Les acteurs permettent de cerner l'interface que le système va offrir à son environnement. Un acteur regroupe plusieurs utilisateurs qui ont

le même rôle. Et pour trouver un acteur il faudra identifier les différents rôles que vont devoir jouer ses utilisateurs.

Pour l'identification des acteurs, nous avons appliqué le principe comme suite : tout ce qui est à l'extérieur et interagit avec le système est un acteur, tout ce qui est à l'intérieur est une fonctionnalité à réaliser.

Notre application s'adapte à l'usage de chaque type d'utilisateur, chacun ayant accès aux fonctionnalités en rapport avec sa fonction.

Les différents acteurs du système étudié sont :

- **L'administrateur** : L'administrateur est l'utilisateur le plus haut dans la hiérarchie. C'est lui qui gère le système ainsi que les droits d'accès des régisseurs communaux et le responsable du service d'assiette.
- **Le responsable :** Le responsable qui gère toutes les taxes ainsi que les notifications, les activités, les secteurs et les statistiques.
- **Le régisseur communal :** Chaque employé gère une ou plusieurs taxes selon le service dont il appartient.

Module	Gestion	Administrateur	Responsable du service d'assiette	Régisseur communal
Local/Redevable	Redevables	Gestion	Gestion	Gestion
	Locaux	Gestion	Gestion	Gestion
	Affectation des taxes	Gestion	Gestion	Gestion
Taxe boisson	Paiement	Gestion	Gestion	Gestion
	Déclaration d'existence	Gestion	Gestion	Gestion
	Taux	Gestion	-	-
Taxe séjour	Paiement	Gestion	Gestion	Gestion
	Taux	Gestion	-	-
Taxe 37	Paiement	Gestion	Gestion	Gestion
	Taux	Gestion	-	-
Taxe 38	Paiement	Gestion	Gestion	Gestion
	Taux	Gestion	-	-
Taxe TNB	Terrain	Gestion	Gestion	Gestion
	Paiement	Gestion	Gestion	Gestion
	Déclaration des terrains	Gestion	Gestion	Gestion
	Taux	Gestion	-	-
	Exonération	Gestion	-	-
Notifications	Notification	Gestion	Gestion	-
Utilitaires	Activités	Gestion	Gestion	-
	Secteurs	Gestion	Gestion	-
Statistiques	Statistiques	Gestion	Gestion	-
Utilisateurs	Utilisateurs	Gestion	-	-

Tableau 4 : Aperçu des droits d'administration

Conclusion

Au terme de ce chapitre, nous avons présenté l'étude préliminaire du projet, qui sert à avoir un premier regard sur les besoins fonctionnels et techniques du projet.

Nous avons présenté également l'étude fonctionnelle du projet réalisé en respectant le processus de développement XP.

Après avoir bien spécifié les exigences fonctionnelles de notre système nous passerons à la phase d'analyse et conception où nous projetterons les exigences captures sur un modèle conceptuelle.

CHAPITRE 3 :

Outils techniques

- Architecture de l'application
- Démarche de conception
- Diagrammes de cas d'utilisation
- Diagrammes de séquence
- Diagramme de classes

Ce chapitre définit l'architecture et la technologie de base de la conception de notre système.

Rapport-Gratuit.com

III. Analyse et conception

1. Architecture de l'application

L'application est constituée de trois couches. Ces trois couches se conforment à l'architecture de couches fermées « closed layer architecture » (une couche peut communiquer seulement avec la couche qui lui est adjacente).

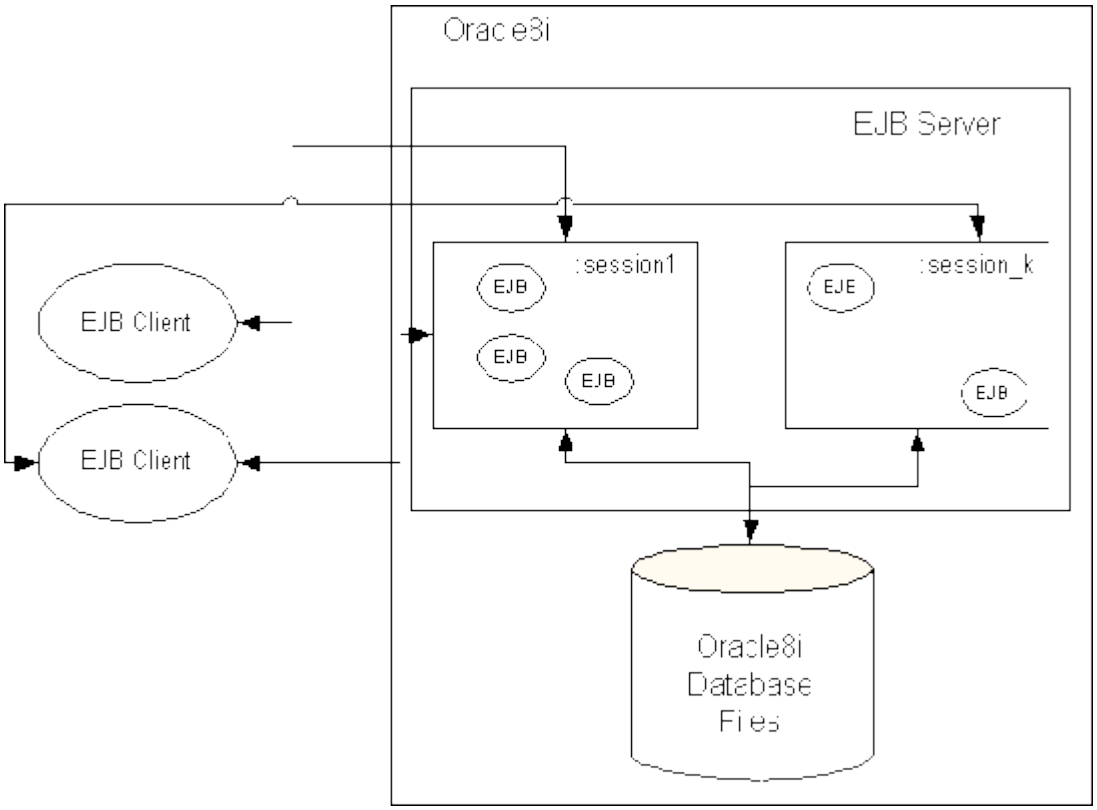


Figure 9 : Architecture standard d'une application EJB

Le concept de partitionner une application en couches et de garder toute la logique de l'application dans ces couches distinctes et séparées, a été introduite bien avant l'approche orientée objet. Ainsi une application est divisée en trois couches logiques, chacune traitant des fonctions spécifiques.

- **Présentation** : interface usager et présentation.
- **Logique du logiciel à produire** (besoins, services de l'entreprise) : les règlements de l'entreprise et la logique de l'application.
- **Logique des données** : Base de données et intégration des services de l'entreprise.

Ce concept nous permet de créer des composants indépendants et de les déployer sur des plates-formes différentes. En fait, ce concept est très utilisé dans le développement des applications multi-tiers. Plus tard, il fut adapté au modèle de conception Model-View-

Controller (MVC) qui est un modèle très commun pour développer des applications distribuées et multi-tiers.

2. Modèles de conception (Design Patterns)

Définition

La majorité des projets les plus structurés architecturalement utilisent des modèles de conceptions, bien que beaucoup de développeurs ne réalisent pas qu'ils les utilisent.

Bien souvent, les développeurs conçoivent des meilleurs pratiques et techniques de programmation durant leurs projets et ne réalisent pas qu'elles sont en effet des modèles de conception « Design Patterns », des approches réutilisables pour programmer.

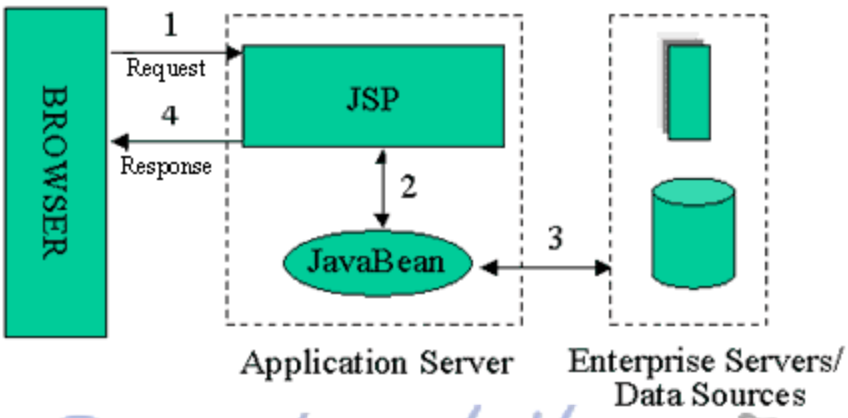
Ce qui nous emmène à définir un modèle de conception comme suit :

Un modèle de conception (Design Patterns) décrit une meilleure pratique, approche, solution prouvée qui permet de résoudre un problème commun ayant tendance à se reproduire, en mettant l'accent sur le contexte et sur les conséquences et l'impact de la solution.

3. Le modèle MVC

a. MVC Modèle 1

L'architecture du MVC Modèle 1 consiste en un navigateur web accédant directement à des pages JSP. Les pages JSP accèdent à des JavaBeans qui représentent le modèle de l'application, et décident de la prochaine page à présenter. Les applications implémentant ce modèle sont décentralisées parce que la page courante détermine la prochaine page à afficher. De plus, chaque page JSP ou Servlet possède ses propres paramètres obtenus à partir des requêtes « GET ou POST ». Dans le modèle 1, choisir la nouvelle page à afficher s'effectue dans un code de script, mais cet usage est considéré vraiment mauvais.



Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Figure 10 : Schéma de MVC 1

b. MVC Modèle 2

Cette architecture introduit un contrôleur Servlet entre le navigateur et les pages JSP ou le contenu des Servlet à délivrer. Le contrôleur centralise la logique d'expédier (dispatching) les requêtes vers la prochaine vue dépendamment de l'URL, de ses paramètres et l'état de l'application (Application State). Le contrôleur gère aussi la sélection de la vue (page de présentation), ainsi permettant de découpler les pages JSP et les Servlet.

Les applications implémentant le MVC Modèle 2 sont plus faciles à maintenir et à étendre, parce que les pages de présentations ne se réfèrent pas les unes aux autres directement.

Le Servlet contrôleur permet un seul point de contrôle, de sécurité et de login, et souvent encapsuler des données en une forme utilisable par les autres couches MVC.

Donc, le MVC modèle 2 est recommandé pour les applications interactives ce qui est le cas pour notre application.

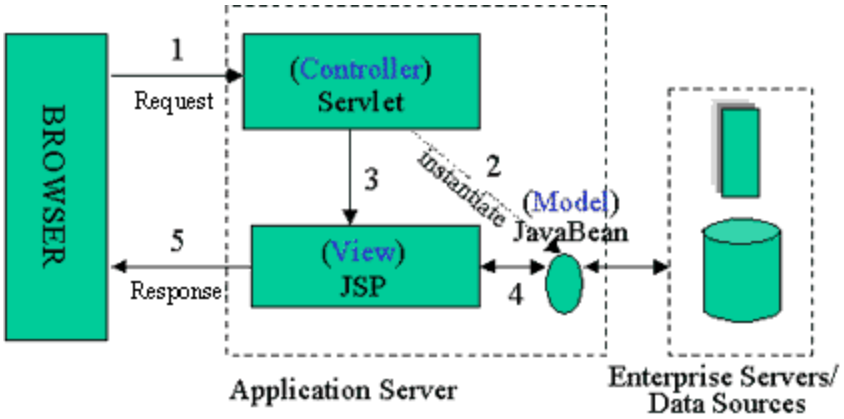


Figure 11 : Schéma de MVC 2

c. Application du MVC Modèle 2 avec les composants du Framework JSF

Le Framework JSF implémente les concepts du modèle Model-View-Controller en les organisant en des composants et concepts d'une application web, comme souligné dans la figure suivante :

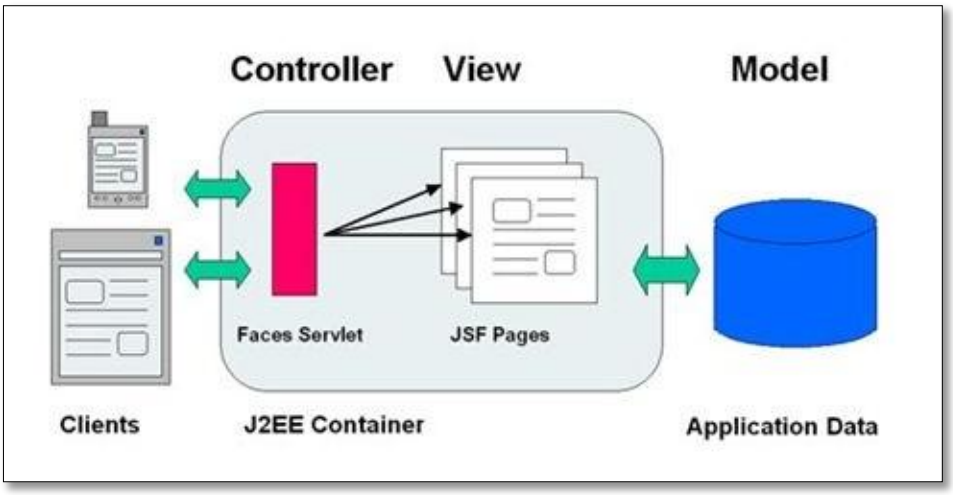


Figure 12 : Implémentation de MVC2 dans JSF

- **Modèle :**

La partie Modèle d'un système base sur le MVC-Model 2 peut être divisée en différents concepts :

- Les états internes du système.
- Les actions qu'on peut prendre pour changer ces états.

Généralement, une application représente les différents états du système comme un ensemble d'un ou plusieurs JavaBeans, avec les propriétés qui représentent les détails de ces états. De nos jours, les applications à vaste étendue représentent les actions (services) de la logique du logiciel à produire qui sont possibles dans un système par des méthodes qui pourront être à sont appelées à partir des informations sur les différents états. D'autres systèmes représentent les actions disponibles séparément par exemple comme un EJB de type session Bean.

- **La Présentation View :**

La vue sous JSF est constituée de pages XHTML. Afin de rendre la création de ces pages aisée par un designer et d'y éviter l'introduction de code Java. JSF et PrimeFaces proposent l'utilisation de plusieurs librairies comme Apache Tomahawk et Mojarra.

- **Le contrôleur :**

Le contrôleur est la partie du Framework qui fait le lien entre la vue et le modèle. C'est elle qui permet de gérer l'enchainement de la navigation. Dans JSF, le point central du contrôleur est un servlet javax.faces.webapp.FacesServlet, toutes les requêtes y aboutissent.

d. Le formalisme UML

UML, (Unified Modeling Language), que l'on peut traduire par "langage de modélisation unifié" est une notation permettant la modélisation d'un problème. Ce langage est né de la fusion de plusieurs méthodes existantes auparavant, et est devenu la référence en termes de modélisation objet.

Entre 1910 et 1990, de nombreux analystes ont mis au point des approches orientées objets, ainsi en 1994 il existait plus de 50 méthodes objet. Toutefois seules 3 méthodes ont véritablement émergé :

- La méthode OMT de Rumbaugh
- La méthode BOOCH'93 de Booch.
- La méthode OOSE de Jacobson (Object Oriented Software Engineering)

A partir de 1994. Rumbaugh et Booch (rejoints en 1995 par Jacobson) ont uni leurs efforts pour mettre au point la méthode unifiée incorporant les avantages de chacune des méthodes précédentes. La méthode unifiée à partir de la version 1.0 devient UML (Unified Modeling Language), une notation universelle pour la modélisation objet. UML 1.0 est soumise à l'OMG1 en janvier 1997 mais elle ne sera acceptée qu'en novembre 199 7dans sa version 1.1, date à partir de laquelle UML devient un standard international cette méthode représente un moyen de spécifier, représenter et construire les composantes d'un système informatique. En effet, la notation unifiée définit 9 diagrammes pour représenter les différents points de vue de modélisation. Ces diagrammes permettent de visualiser et de manipuler les éléments de modélisation.

Les diagrammes définis par UML, sont les suivants :

- ✓ Les diagrammes de cas d'utilisation : représentation des fonctions du système du point de vue de l'utilisateur
- ✓ Les diagrammes de séquence : représentation temporelle des objets et de leurs Les diagrammes d'activités représentation du comportement d'une opération en termes d'actions.
- ✓ Les diagrammes de composants : représentation du code en termes de modules, de composants et sur tous des concepts du langage ou de l'environnement d'implémentation.
- ✓ Les diagrammes de classes : représentation de la structure statique en termes de classes et de relations.

e. Le choix UML

- ✓ Vu que l'application doit être robuste, extensible et modulaire, une modélisation objet apparait la plus adaptée. En effet, l'objet a fait ses preuves dans la réalisation

d'applications temps réel. C'est pourquoi nous avons opté pour UML comme langage de modélisation.

- ✓ Ce choix peut être justifié également par plusieurs raisons La notation UML facilite la compréhension et la communication d'une modélisation objet.
- ✓ La notation UML, par définition, n'est pas spécifique à un langage de programmation objet, elle peut donc être utilisée avec n'importe quel langage tel que C#, J#, JAVA ou C++ UML est aujourd'hui un standard, adopté par les grands constructeurs de logiciel du marché. Durant notre étude du système, nous avons utilisé trois diagrammes d'UML, il s'agit du diagramme des cas d'utilisation, les diagrammes de séquences, et le diagramme de classes.

4. Diagramme de cas d'utilisation

Un cas d'utilisation représente un ensemble de séquences d'actions réalisées par le système et produisant un résultat observable intéressant pour un acteur particulier. Pour constituer les cas d'utilisation, il faut considérer l'intention fonctionnelle de l'acteur par rapport au système dans le cadre de l'émission ou de la réception de chaque message. En regroupant les intentions fonctionnelles en unités cohérentes, nous obtenons les cas d'utilisation.

a. Description détaillée des cas d'utilisation

Comme représenté sur le diagramme ci-dessous, les cas d'utilisations majeurs offerts par le système sont :

Remarque : Dans tous les diagrammes ci-dessous, l'authentification est un cas d'utilisation lié directement à l'acteur, d'où l'élimination des inclusions qui figuraient entre l'authentification et les autres cas d'utilisation.

<u>Module Local/Redevable :</u>

- ❖ **Gestion redevables :**

➢ Diagramme de cas d'utilisation

« Gestion redevables» :

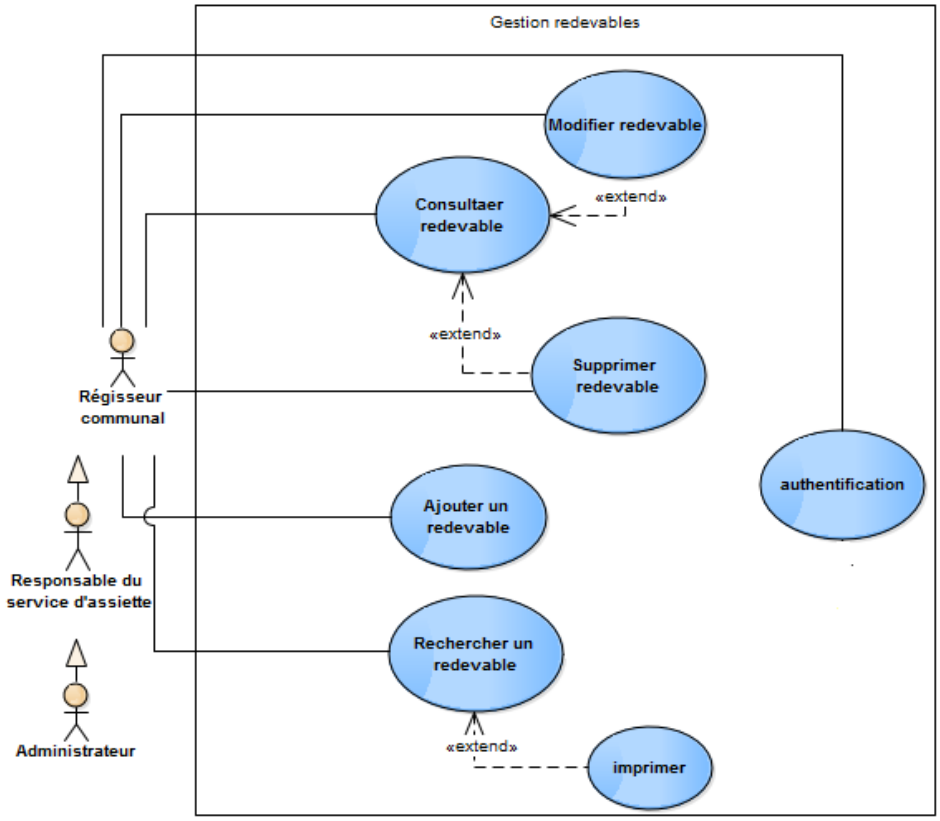


Figure 13 : Diagramme de cas d'utilisation « Gestion redevables »

- Description du diagramme du cas d'utilisation pour la gestion des « Redevables» :

Libellé	Acteur	Description	Précondition	Post condition
Ajouter un redevable		Création d'un nouveau redevable.	Néant	Néant
Modifier redevable		Modification du redevable concerné.	Néant	Néant
Supprimer redevable		Suppression du redevable concerné.	Vérifier l'inexistence des locaux associés	Néant
Consulter redevable	➢ Administrateur ➢ Responsable ➢ Régisseur	Affichage des différents redevables qui sont insérés dans la base de données dans un tableau	Néant	Néant
Rechercher un redevable		Effectuer une recherche sur le(s) redevable(s) par statut juridique, par RC/CIN, par nom et prénom et par raison social.	Néant	Néant
Imprimer		Imprimer le résultat de la recherche.	Vérifier que la liste à imprimer contient au moins un redevable	Néant

Tableau 5 : Description du cas d'utilisation « Gestion redevables »

❖ Gestion des locaux :

➢ Diagramme du cas d'utilisation «Gestion des locaux»

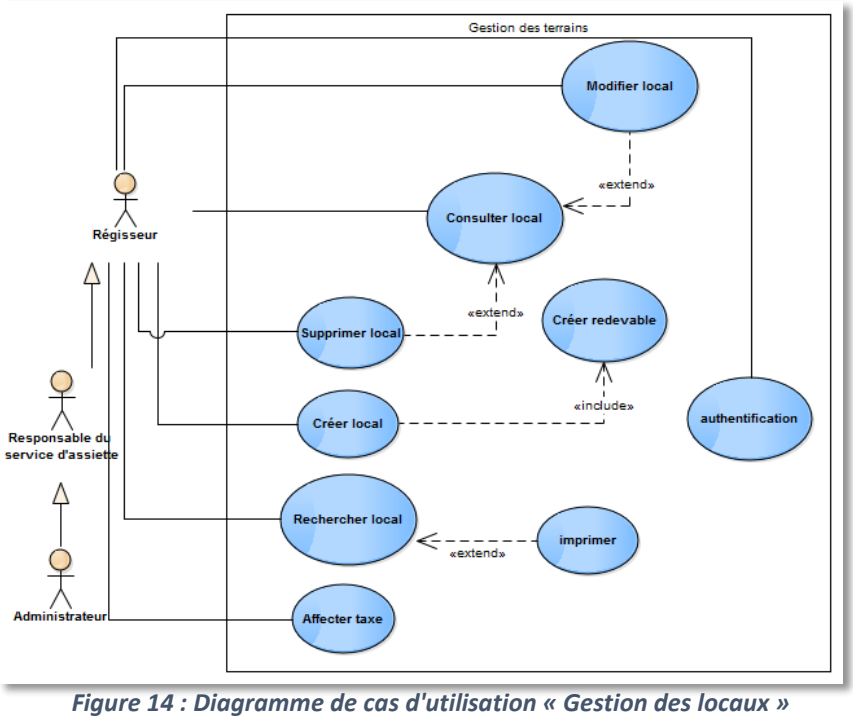


Figure 14 : Diagramme de cas d'utilisation « Gestion des locaux »

➢ Description du diagramme du cas d'utilisation pour «Gestion locaux» :

Libellé	Acteur	Description	Précondition	Post condition
Créer un local	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'un nouveau local.	Vérifier l'existence du redevable (propriétaire/gérant)	Néant
Modifier local		Modification du locale concerné.	Néant	Néant
Supprimer local		Suppression du locale concerné.	Vérifier que le local concerné n'as aucun paiement	Néant
Consulter local		Affichage des différents locaux qui sont insérés dans la base de données dans un tableau	Néant	Néant
Rechercher un locale		Effectuer une recherche sur le(s) locale (aux) selon plusieurs critères.	Néant	Néant
Imprimer		Imprimer le résultat de la recherche.	Vérifier que la liste à imprimer contient au moins un local	Néant

Tableau 6 : Description du diagramme de cas d'utilisation « Gestion locaux »

❖ Affectation des taxes :

➢ Diagramme du cas d'utilisation «Affectation des taxes» à un local :

✓ Les taxes concernées sont : La taxe boisson, séjour, 37 et 38

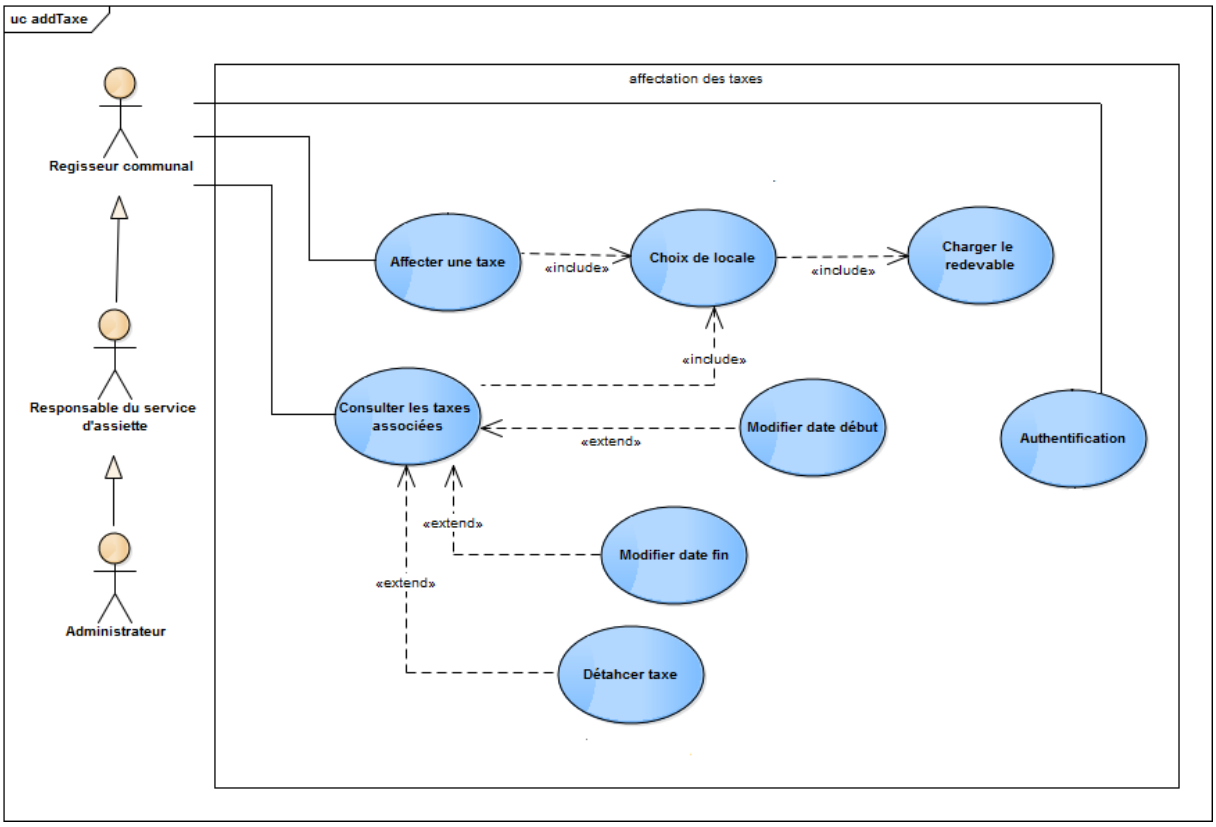


Figure 15 : Diagramme de cas d'utilisation « Affectation des taxes »

➢ Description du diagramme du cas d'utilisation pour «Affectation des taxes» :

Libellé	Acteur	Description	Précondition	Post condition
Affecter une taxe	➢ Régisseur ➢ Responsable ➢ Administrateur	Affecter une taxe (boisson, séjour, 37 ou 38) à un local.	-Récupération du local concerné. -Vérifier que ce local n'as pas la taxe à affecter	Modifier le local en affectant la taxe concerné
Choix de local		Choisir le local en question.	Néant	Néant
Consulter les taxes associées		Consulter les taxes associées à un local	Néant	Néant
Modifier la date début		Modifier la date de début d'activité d'un local	Vérifier qu'il est inférieur à la date fin s'il existe	Néant
Modifier la date fin		Modifier la date fin d'activité d'un local	Néant	Local n'est pas censé à payer après cette date
Détacher taxe		Détacher une taxe d'un local	Vérifier que le local n'a pas un paiement pour cette taxe	Modification du local en détachant la taxe

Tableau 7 : Description du diagramme de cas d'utilisation « Affectation des taxes »

Module taxe boisson :

❖ Paiement taxe boisson :

➢ Diagramme de cas d'utilisation « Paiement taxe boisson» :

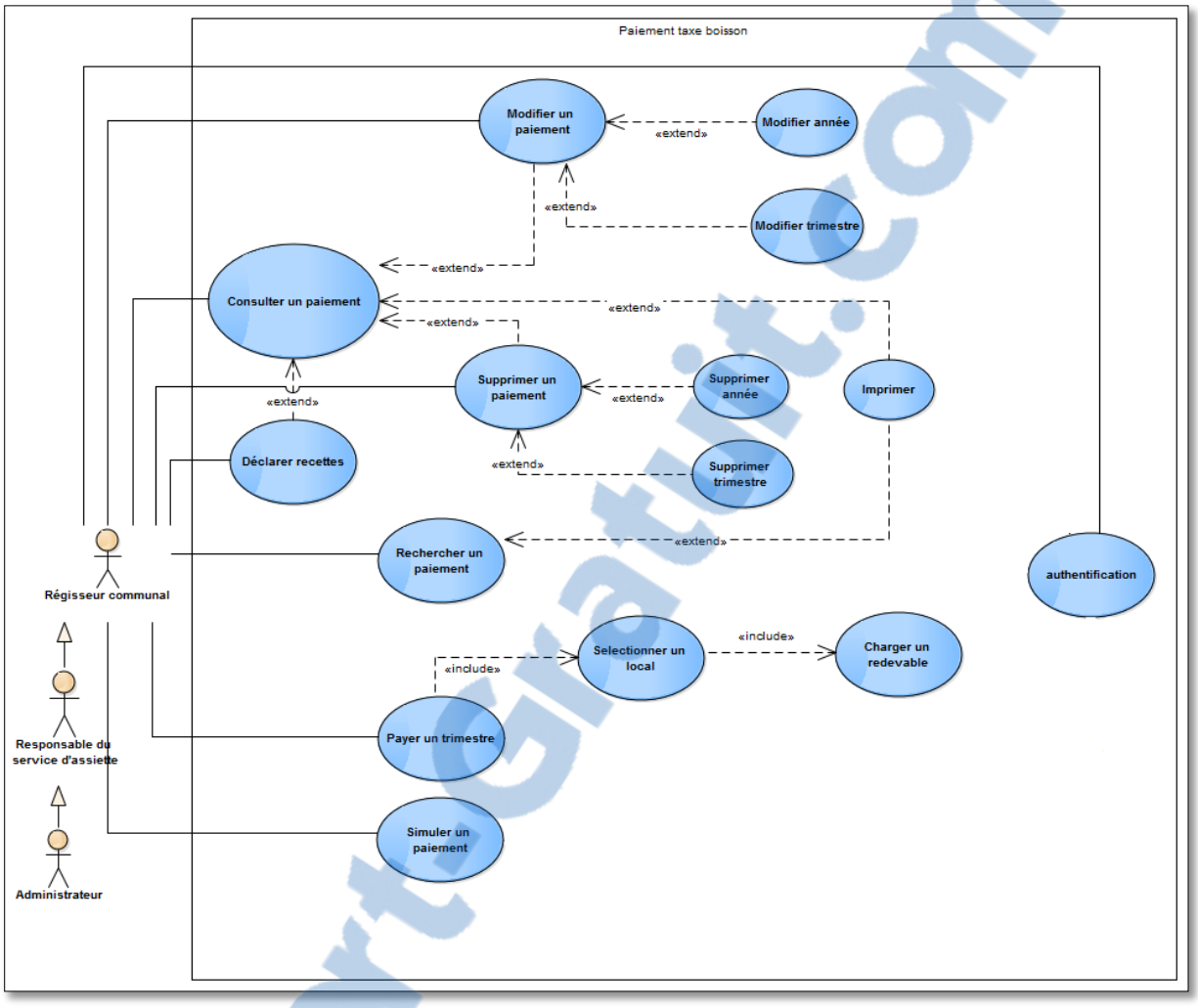


Figure 16 : Diagramme de cas d'utilisation « Paiement taxe boisson »

➢ Description du diagramme du cas d'utilisation pour la gestion des «Paiement taxe boisson» :

Libellé	Acteur	Description	Précondition	Post condition
Payer un trimestre	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'un nouveau paiement trimestriel.	- Récupérer le local concerné par le paiement - Validation des données (date présentation, année) - Génération du trimestre à payer - Vérification que le trimestre n'est pas payé	- Modifier le montant total de l'annualité auquel appartient le trimestre - Modifier le montant total payé par le local pour une taxe - Modifier l'année et le trimestre du dernier paiement du local concerné
Modifier un paiement		Modification du paiement concerné : modification de l'année ou bien d'un trimestre de l'année.	- Vérifier que le trimestre n'est pas payé	- Modifier le montant total annuel. - Modifier le montant total du local

RapportGratuit.com

				- Modifier l'année et le n° trimestre du dernier paiement du local
Supprimer un paiement		Suppression du paiement concerné.	Supprimer les éléments associés s'ils existent	- Modifier le montant total au niveau de l'année et le local concerné - Modifier l'année et le n° trimestre du dernier paiement du local
Consulter paiement		Affichage des différents paiements qui sont insérés dans la base de données dans un tableau	Néant	Néant
Rechercher un paiement		Effectuer une recherche sur le(s) paiement (s) selon plusieurs critères.	Néant	Néant
Imprimer		Imprimer le résultat de la recherche.	Vérifier que la liste à imprimer contient au moins un paiement	Néant
Simuler un paiement		Simulation d'un paiement trimestriel sans l'enregistrer dans la base de données.	Néant	Néant
Déclarer recettes		Effectuer une déclaration annuelle de l'année concernée.	Néant	Modification de la dernière année effective du paiement chez le local

Tableau 8 : Description du cas d'utilisation « Paiement taxe boisson »

❖ Déclaration d'existence :

➢ Diagramme de cas d'utilisation « Déclaration d'existence» :

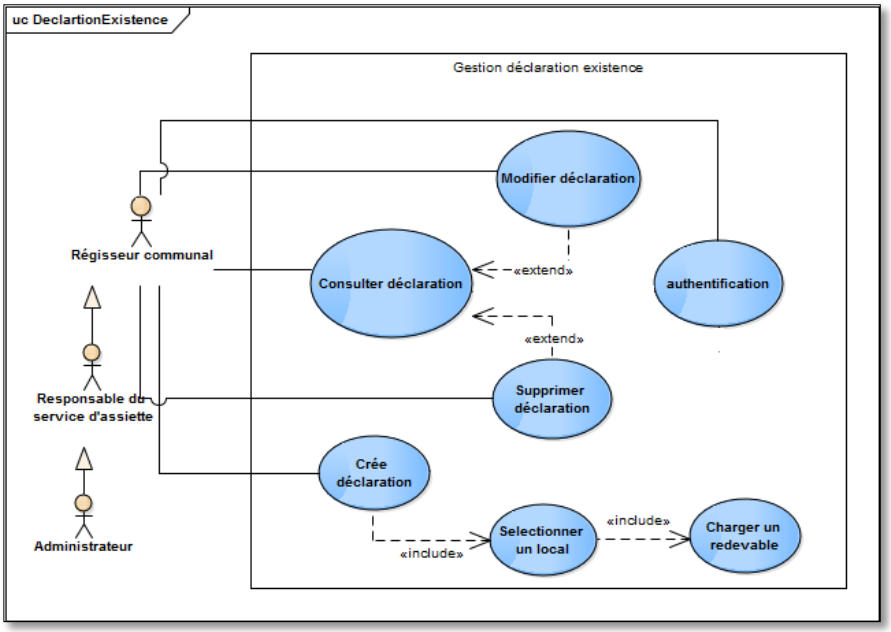


Figure 17 : Diagramme de cas d'utilisation « Déclaration d'existence »

- Description du diagramme du cas d'utilisation pour la gestion des «Déclaration d'existence» :

Libellé	Acteur	Description	Précondition	Post condition
Créer déclaration	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'une nouvelle déclaration d'existence en précisant un local après avoir chargé le redevable en question.	Vérifier l'inexistence de la déclaration	-Affecter la taxe boisson au local s'il n'a pas cette taxe. -Calcul d'une pénalité si la différence entre la date de déclaration et la date de début d'activité dépasse les 30 jours
Modifier déclaration		Modification de la déclaration concernée.	Néant	Calcul d'une pénalité si la différence entre la date de déclaration et la date de début d'activité dépasse les 30 jours
Supprimer déclaration		Suppression de la déclaration concernée.	Néant	Néant
Consulter déclaration		Affichage des différentes déclarations qui sont insérés dans la base de données dans un tableau	Néant	Néant

Tableau 9 : Description du cas d'utilisation « Déclaration d'existence »

❖ Taux taxe boisson :

- Diagramme de cas d'utilisation «Gestion taux boisson» :
  - ✓ On distingue trois types de taux à savoir :
    - Le taux appliqué lors du paiement de la taxe.
    - Le taux de retard appliqué lors d'un paiement hors délai.
    - Le taux appliqué lorsqu'il s'agit d'un défaut de déclaration.

Le diagramme qui suit est valable pour la gestion des trois taux :

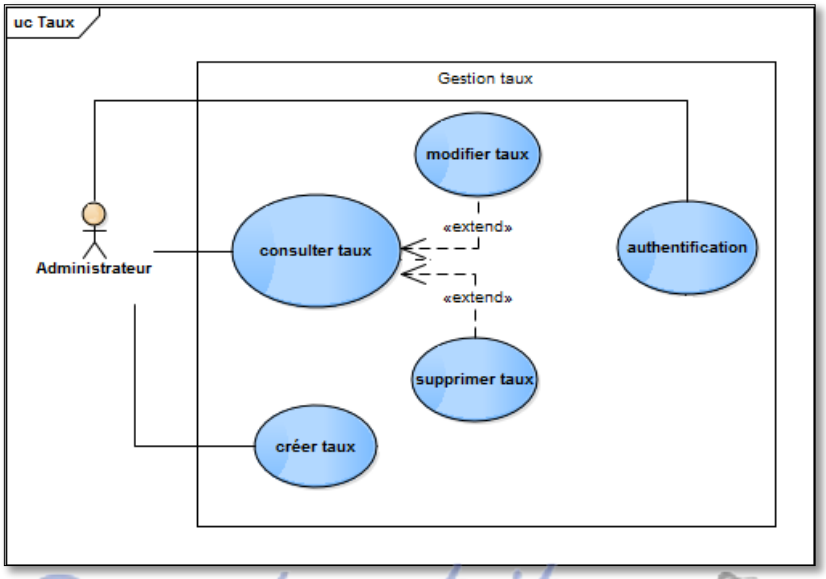


Figure 18 : Diagramme de cas d'utilisation « Gestion taux boisson »

Rapport gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

➢ Description du diagramme du cas d'utilisation pour la gestion des «Taux» :

Libellé	Acteur	Description	Précondition	Post condition
Créer taux	➢ Administrateur	Création d'un nouveau taux.	Néant	Néant
Modifier taux		Modification du taux concerné.	Néant	Néant
Supprimer taux		Suppression de taux concernée.	Vérifier que le taux n'est pas lié à un paiement	Néant

Tableau 10 : Description du cas d'utilisation « Gestion taux boisson »

Module taxe séjour :

❖ Paiement taxe séjour :

➢ Diagramme de cas d'utilisation « Paiement taxe séjour» :

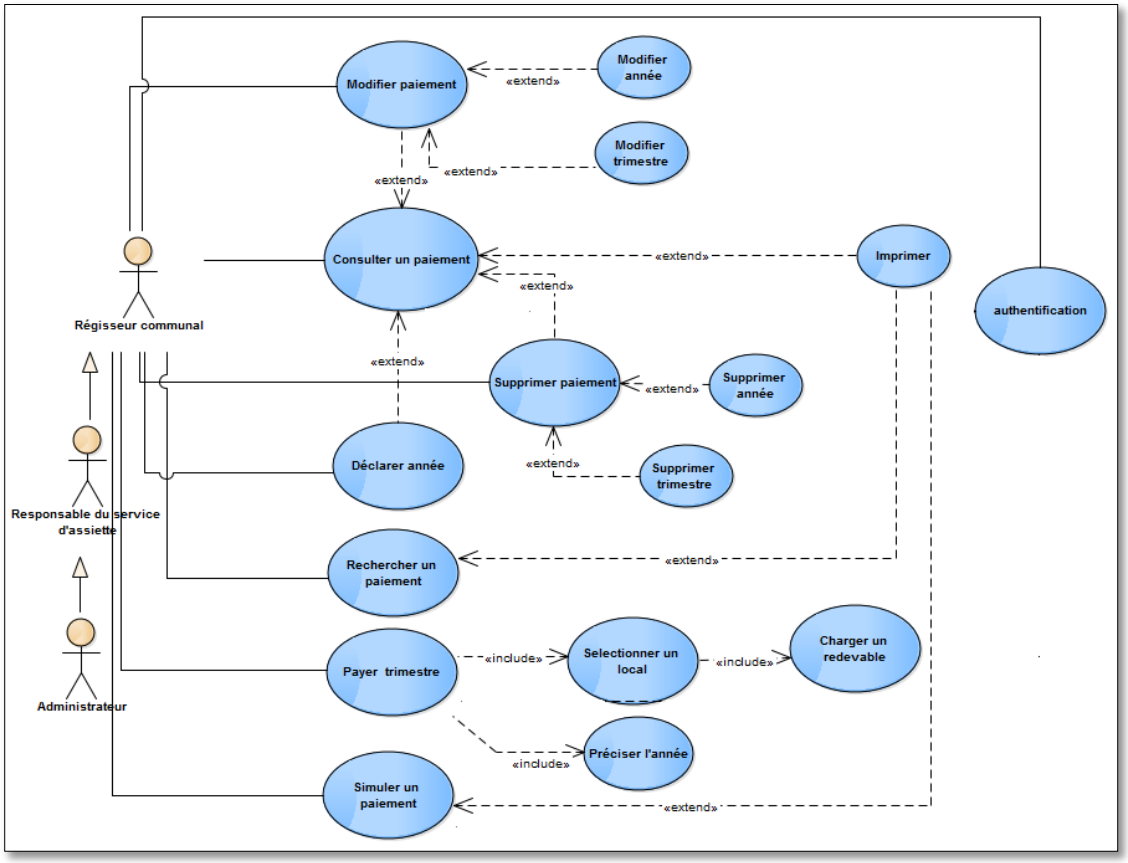


Figure 19 : Diagramme de cas d'utilisation « Paiement taxe séjour »

➢ Description du diagramme du cas d'utilisation pour la gestion des «Paiement taxe séjour» :

Libellé	Acteur	Description	Précondition	Post condition

Payer un trimestre	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'un nouveau paiement trimestriel.	- Récupérer le local concerné par le paiement - Validation des données (date présentation, année) - Génération du numéro de trimestre à payer - Vérification que le trimestre n'est pas payé	- Modifier le montant total de l'année auquel appartint le trimestre - Modifier le montant total payé par le local pour une taxe - Modifier l'année et le trimestre dernier paiement pour
Modifier un paiement		Modification du paiement concerné : modification de l'année ou bien d'un trimestre de l'année.	Vérifier que le trimestre n'est pas payé	- Modifier le montant total de la taxe annuelle - Modifier le montant total payé du local - Modifier l'année et le n° trimestre du dernier paiement du local
Supprimer un paiement	➢ Administrateur ➢ Responsable ➢ Régisseur	Suppression du paiement concerné.	Supprimer les éléments associés s'ils existent	- Modifier les montants totaux de l'annualité et du local - Modifier l'année et le trimestre du dernier paiement du local concerné.
Consulter paiement		Affichage des différents paiements qui sont insérés dans la base de données dans un tableau	Néant	Néant
Rechercher un paiement		Effectuer une recherche sur le(s) paiement (s) selon plusieurs critères.	Néant	Néant
Imprimer		Imprimer le résultat de la recherche.	Vérifier que la liste à imprimer contient au moins un paiement	Néant
Simuler un paiement		Simulation d'un paiement trimestriel sans l'enregistrer dans la base de données.	Néant	Néant
Déclarer recettes		Effectuer une déclaration annuelle de l'année concernée.	Néant	Modification de la dernière année effective du paiement chez le local

Tableau 11 : Description du diagramme de cas d'utilisation « paiement taxe séjour »

- Diagramme de cas d'utilisation «Gestion taux séjour» :
  - ✓ Nous gardons le même diagramme que la taxe boisson « Figure 18 » ci-dessus.

Modules taxes 37 et 38 :

- Diagramme de cas d'utilisation « Paiement taxe 37» :

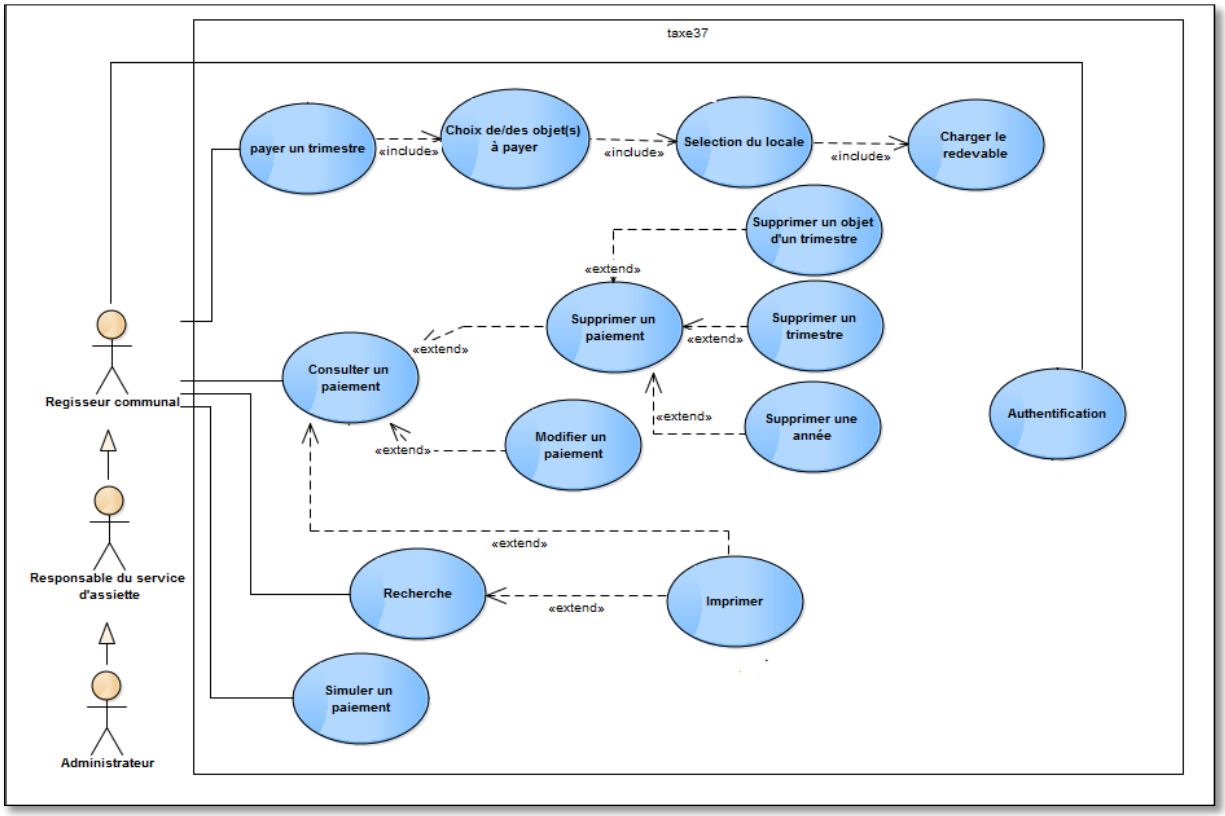


Figure 20 : Diagramme de cas d'utilisation « Paiement taxe 37 »

- Diagramme de cas d'utilisation « Paiement taxe 38»

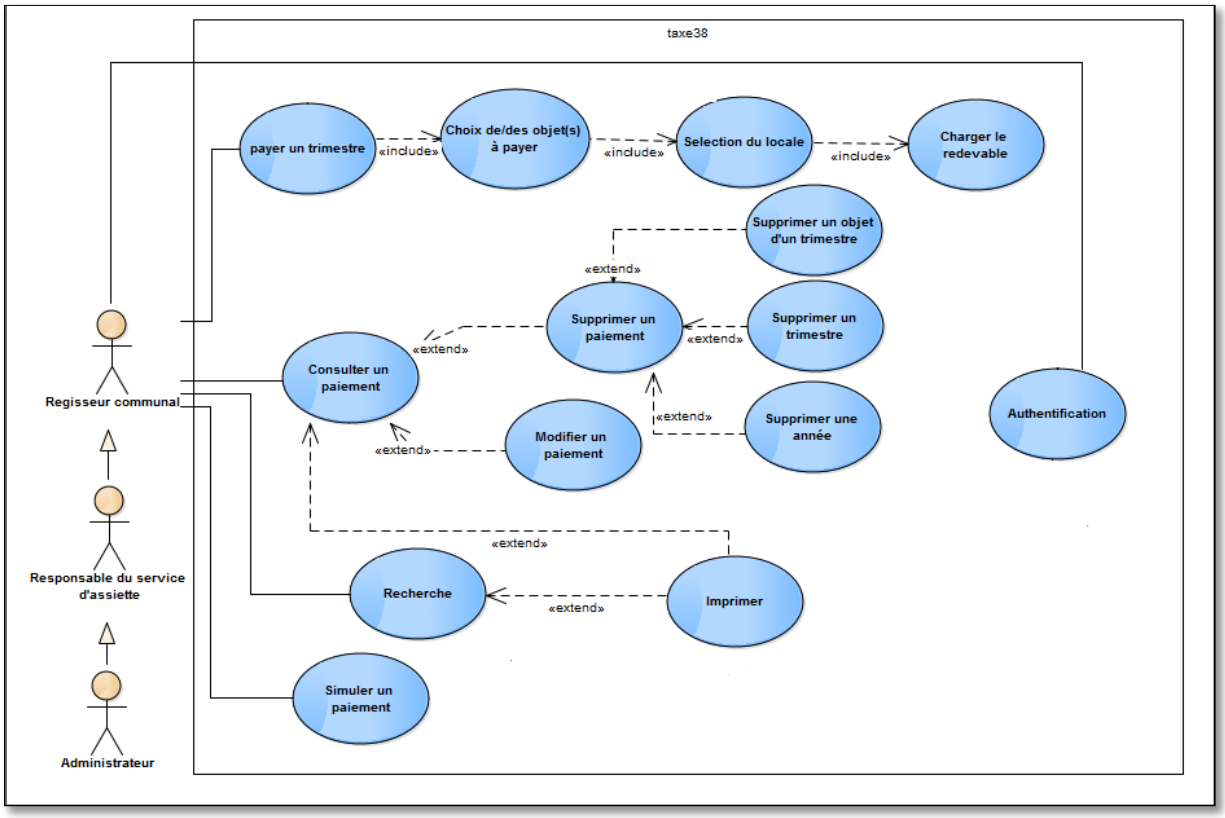


Figure 21 : Diagramme de cas d'utilisation « Paiement taxe 38 »

Les taxes 37 et 38 ont le même mode de paiement, la différence est au niveau des taux appliqué ainsi que les catégories des objets de ces taxes.

- Description des diagrammes des cas d'utilisation « Paiement taxe 37 » et « Paiement taxe 38 » :

Libellé	Acteur	Description	Précondition	Post condition
Payer un trimestre	➢ Régisseur ➢ Responsable ➢ Administrateur	Paiement d'un trimestre	- Récupérer local concerné par le paiement - Validation des données (date présentation, année) - Génération du n° trimestre à payer - Vérification que le trimestre n'est pas payé	- Modifier le montant total de l'annuel dont appartint le trimestre - Modifier le montant total payé par le local pour une taxe - Modifier l'année et le trimestre dernier paiement pour le local
Choix de/des objet(s) à payer		Choisir les objets 37 à payer	Néant	Néant
Consulter un paiement		Consultation de la liste des paiements annuels effectués	Néant	Néant
Supprimer un paiement		Supprimer le paiement d'un objet, un trimestre ou une année	Suppression des éléments associés	-Modification du montant total annuel - Modification du dernier trimestre et année de paiement au niveau du local
Modifier un paiement		Modifier un trimestre ou une année	Néant	- Modification du montant de la taxe
Recherche		Effectué une recherche selon des critères.	Néant	Néant
Imprimer		Imprimer un pdf de la recherche ou le bordereau de versement.	Vérifier que la liste à imprimer contient au moins un paiement	Néant

Tableau 12 : Description des diagrammes des cas d'utilisation « Paiement taxe 37 » et « Paiement taxe 38 »

Module Taxe TNB :

➢ Paiement TNB :

- ✓ Diagramme de cas d'utilisation « Paiement taxe TNB» :

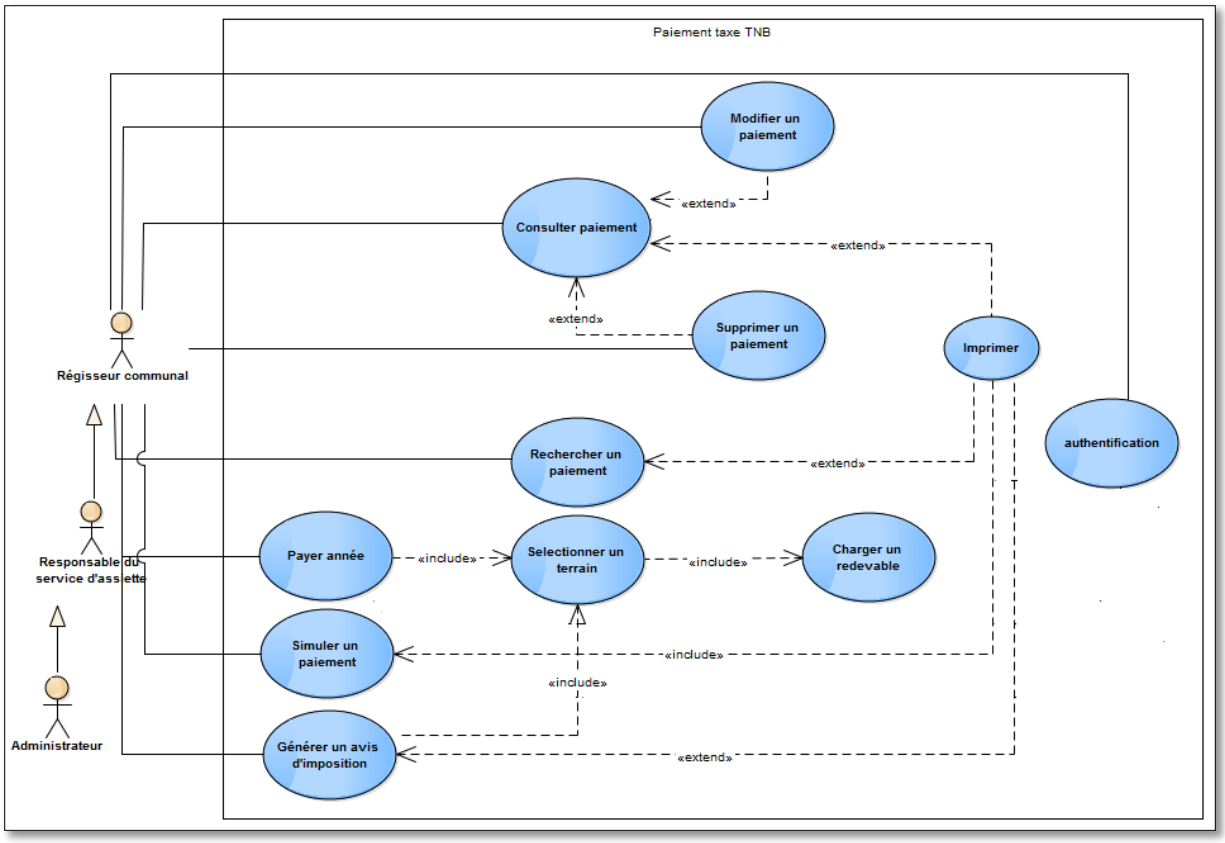


Figure 22 : Diagramme de cas d'utilisation « Paiement taxe TNB »

- Description du diagramme du cas d'utilisation pour «Paiement taxe TNB» :

Libellé	Acteur	Description	Précondition	Post condition
Payer année	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'un nouveau paiement annuel.	- Récupérer le terrain concerné par le paiement. - Vérifier que le terrain n'a pas de permis d'habiter -Vérifier que l'année à payer est supérieure ou égale à l'année d'imposition - Validation des données (date présentation, année) -Vérification de l'existence d'un paiement de même année.	- Modification du dernier paiement au niveau du terrain -Modification de la dernière année payée au niveau du terrain. - Modification du montant total des taxes payée au niveau du terrain.
Modifier un paiement		Modification du paiement concerné.	Néant	-Modification du montant de la taxe. - Modification du montant total des taxes payée au niveau du terrain. -Modification de la dernière année payée au niveau du terrain.
Supprimer un paiement		Suppression du paiement concerné.	Néant	-Modification de la dernière année payée au niveau du terrain.

				-Modification du montant total des taxes payées au niveau du terrain.
Consulter paiement		Affichage des différents paiements qui sont insérés dans la base de données dans un tableau	Néant	Néant
Rechercher un paiement		Effectuer une recherche sur le(s) paiement (s) selon plusieurs critères.	Néant	Néant
Imprimer		Imprimer le résultat de la recherche.	Vérifier que la liste à imprimer contient au moins un paiement	Néant
Simuler un paiement		Simulation d'un paiement sans l'enregistrer dans la base de données.	Néant	Néant
Générer un avis d'imposition		Génération d'un avis d'imposition contenant tous les années non payé d'un terrain à partir de l'année d'imposition.	- Récupérer terrain concerné. - Validation des données (date présentation) - Vérifier que le terrain n'a pas de permis d'habiter	Néant
Imprimer		Impression d'un avis d'imposition.	Vérifier que la liste à imprimer contient au moins un élément.	Néant

Tableau 13 : Description du cas d'utilisation « Paiement taxe TNB »

- Diagramme de cas d'utilisation « Gestion des terrains» :

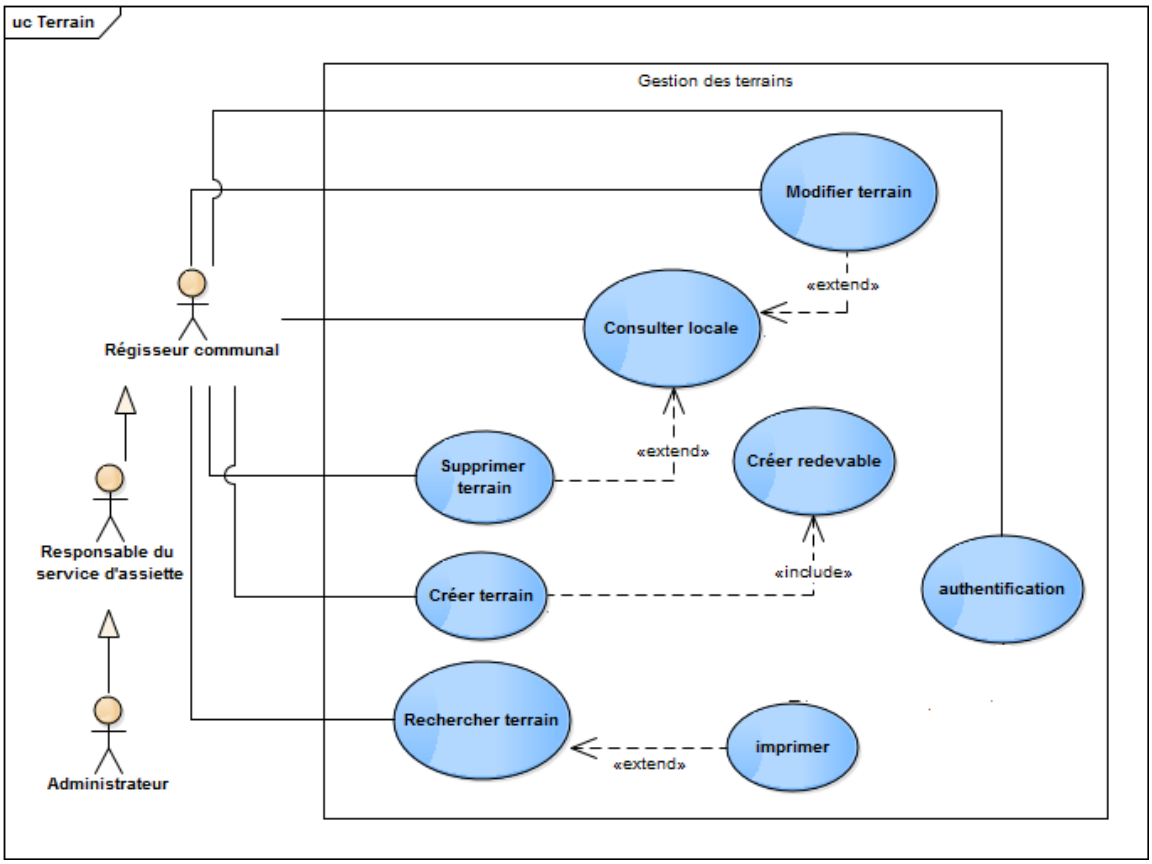


Figure 23 : Diagramme de cas d'utilisation « Gestion terrains »

➢ Description du diagramme du cas d'utilisation pour la gestion des «Terrains» :

Libellé	Acteur	Description	Précondition	Post condition
Créer un terrain	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'un nouveau terrain.	Vérifier l'existence du redevable et le récupérer.	Néant
Modifier terrain		Modification du terrain concerné.	Néant	Néant
Supprimer terrain		Suppression du terrain concerné.	Vérifier que le terrain n'a pas de paiements dans la base de données.	Néant
Consulter terrain		Affichage des différents terrains qui sont insérés dans la base de données dans un tableau.	Néant	Néant
Rechercher un terrain		Effectuer une recherche sur le(s) terrain (s) selon plusieurs critères.	Néant	Néant
Imprimer		Imprimer le résultat de la recherche.	Vérifier que la liste à imprimer contient au moins un élément.	Néant

Tableau 14 : Description du cas d'utilisation « Gestion des terrains »

❖ Déclaration des terrains

➢ Diagramme de cas d'utilisation « Déclaration terrain» :

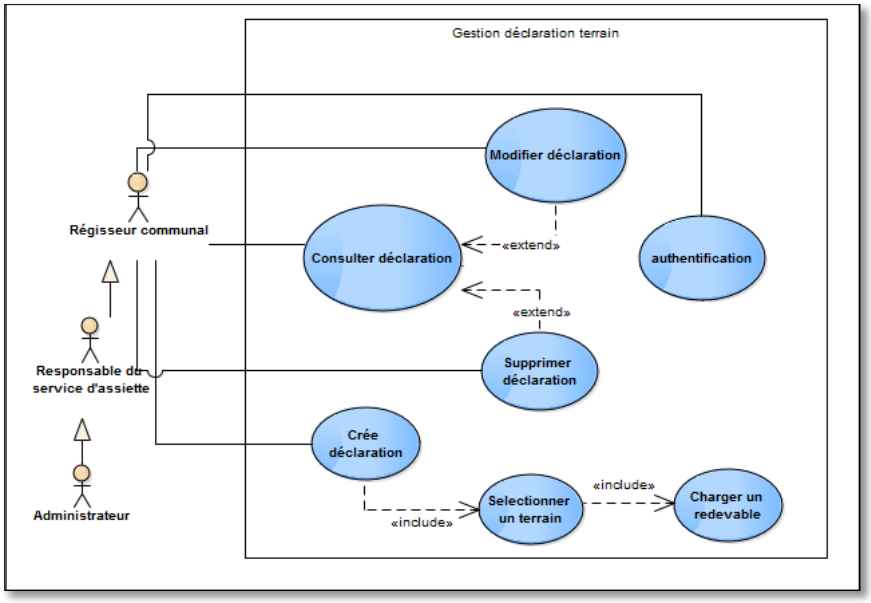


Figure 24 : Diagramme de cas d'utilisation « Déclaration terrain »

➢ Description du diagramme du cas d'utilisation «Déclaration des terrains» :

Libellé	Acteur	Description	Précondition	Poste condition
Créer déclaration	➢ Administrateur ➢ Responsable ➢ Régisseur	Création d'une nouvelle déclaration du terrain en précisant un terrain après avoir chargé le redevable en question.	- récupérer le terrain concerné par la déclaration	-Modifier la dernière année de déclaration au niveau du terrain.
Modifier déclaration		Modification de la déclaration concernée.	Néant	-Calcul de la majoration en cas de retard
Supprimer déclaration		Suppression de la déclaration concernée.	Néant	-Modifier la dernière année de déclaration au niveau du terrain.
Consulter déclaration		Affichage des différentes déclarations qui sont insérés dans la base de données dans un tableau	Néant	Néant

Tableau 15 : Description du cas d'utilisation « Déclaration des terrains »

❖ Exonération

➢ Diagramme de cas d'utilisation « Exonération» :

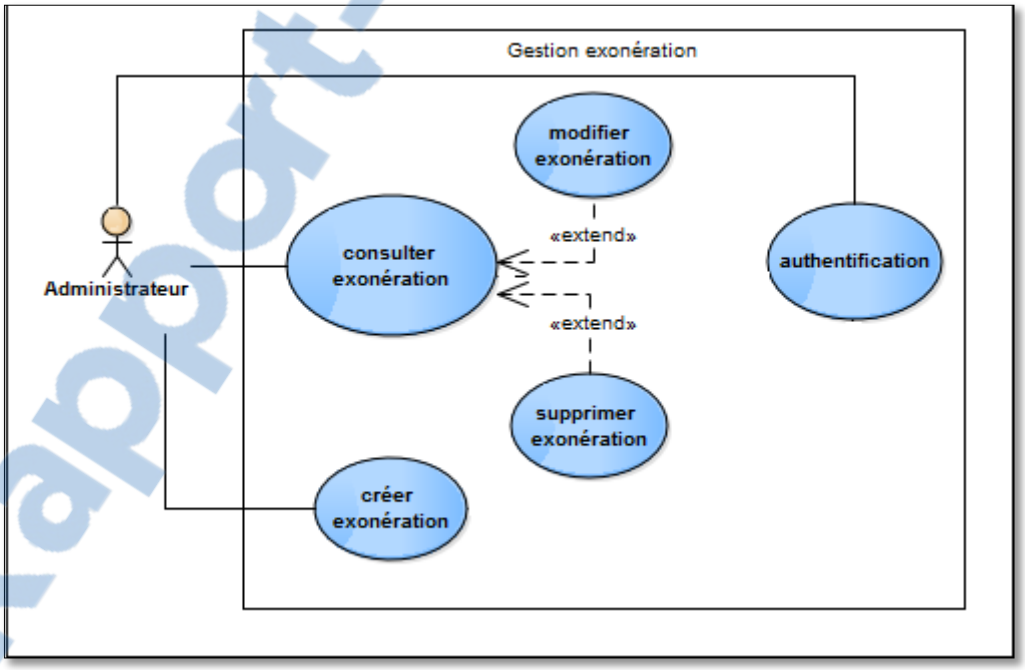


Figure 25 : Diagramme de cas d'utilisation « Exonération »

Rapport-Gratuit.com

➢ Description du diagramme du cas d'utilisation pour la gestion «Exonérations» :

Libellé	Acteur	Description	Précondition	Poste condition
Créer exonération	➢ Administrateur	Création d'une nouvelle exonération.	Néant	Néant
Modifier exonération		Modification de l'exonération concernée.	Néant	Néant
Supprimer exonération		Suppression de l'exonération concernée.	Néant	Néant

Tableau 16 : Description du cas d'utilisation « Exonération »

❖ Taux

➢ Diagramme de cas d'utilisation «Gestion taux TNB» :

✓ On distingue trois types de taux à savoir :

- Le taux appliqué lors du paiement de la taxe.
- Le taux de retard appliqué lors d'un paiement hors délai.
- Le taux de retard de déclaration lorsqu'il s'agit d'un défaut de déclaration de terrain.

Le diagramme qui suit est valable pour décrire la gestion des trois taux :

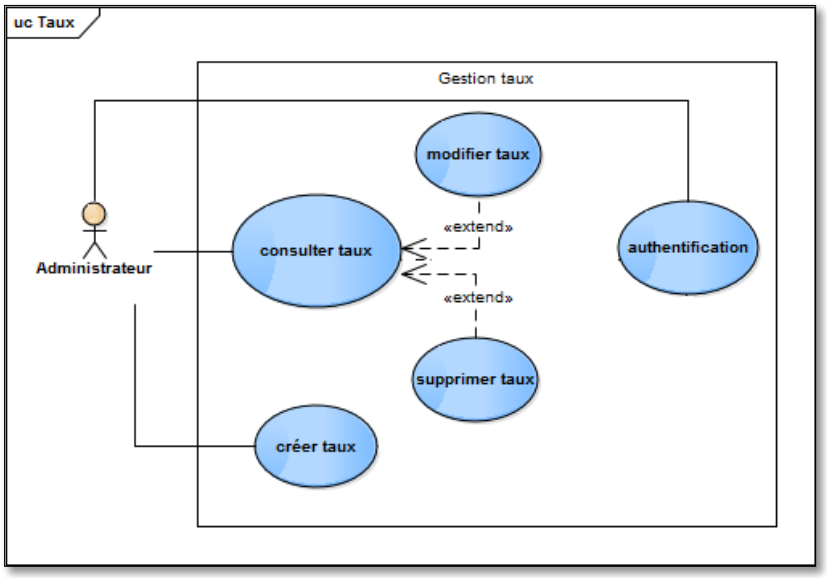


Figure 26 : Diagramme de cas d'utilisation « Gestion taux TNB »

➢ Description du diagramme du cas d'utilisation pour la gestion des «Taux» :

Libellé	Acteur	Description	Précondition	Poste condition
Créer taux	➢ Administrateur	Création d'un nouveau taux.	Néant	Néant
Modifier taux		Modification du taux concerné.	Néant	Néant

Supprimer taux		Suppression de taux concernée.	Vérifier que le taux n'est pas lié à un paiement	Néant

Tableau 17 : Description du diagramme de cas d'utilisation « Gestion taux TNB »

Module notification :

- Diagramme de cas d'utilisation « Gestion notification» :

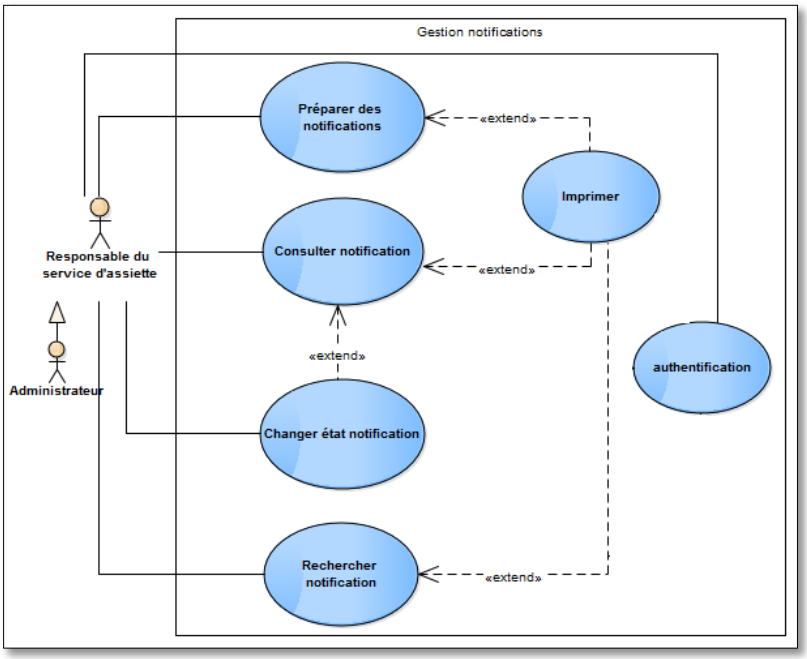


Figure 27 : Diagramme de cas d'utilisation « Gestion notifications »

- Description du diagramme du cas d'utilisation pour «Gestion notification» :

Libellé	Acteur	Description	Précondition	Post condition
Préparer des notifications	➢ Administrateur ➢ Responsable	Préparation des notifications en précisant le type de la taxe, numéro de notification, l'année à notifier et la date d'envoi.	-Rechercher les locaux dont la taxe est non déclaré (non payé dans le cas de la taxe TNB) -Calculer une estimation (montant de base, cas de la taxe TNB)	- Modification de la date dernière notification au niveau du local /terrain - Modifier le type de notification au niveau du local/terrain
Consulter notification		Affichage des différentes notifications déjà préparé dans un tableau	Néant	Néant
Changer état notification		Changement de l'état de la notification concerné à : Imprimé, non imprimé, reçu ou non reçu	Sélectionner une ou plusieurs notifications	Néant

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Rechercher notification		Rechercher une ou plusieurs notifications selon des critères.	Néant	Néant
Imprimer		Impression de la lettre de notification.	Vérifier que la liste des notifications contient au moins un élément	Néant

Tableau 18 : Description du diagramme « Gestion notification »

Module utilisateurs :

- Diagramme de cas d'utilisation « Gestion utilisateur» :

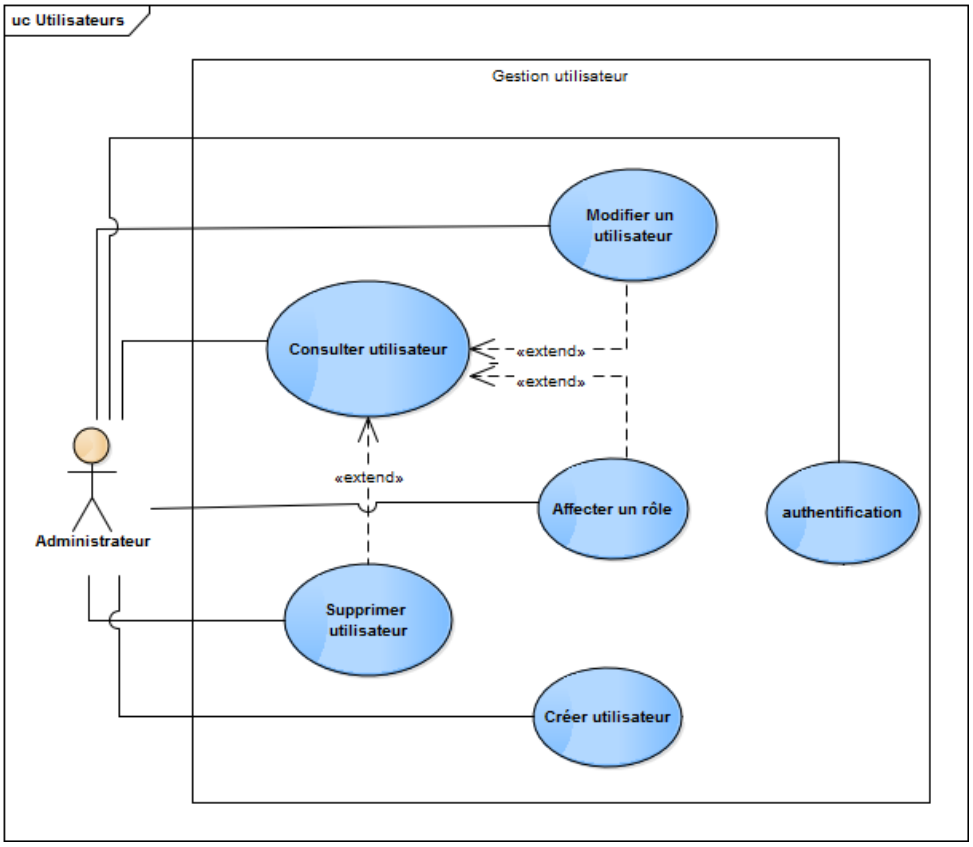


Figure 28 : Diagramme de cas d'utilisation « Gestion utilisateur »

- Description du diagramme du cas d'utilisation pour «Gestion utilisateurs» :

Libellé	Acteur	Description	Précondition	Post condition
Créer utilisateur	➢ Administrateur	Création d'un nouvel utilisateur.	Vérifier que le login n'est pas associé à un autre utilisateur	Néant

Consulter utilisateur		Affichage des différents utilisateurs enregistrés dans la base de données dans un tableau	Néant	Néant
Modifier utilisateur		Modification des données d'un utilisateur, blocage ou bien réinitialisation du mot de passe.	Néant	Néant
Affecter rôle		Affectation d'un rôle à un utilisateur	Néant	Néant
Supprimer utilisateur		Suppression de l'utilisateur concerné.	Vérifier que l'utilisateur n'a rien saisi	Néant

Tableau 19 : Description du diagramme de cas d'utilisation « Gestion utilisateur »

Module utilitaire :

❖ Gestion des activités :

➢ Diagramme de cas d'utilisation « Gestion activité» :

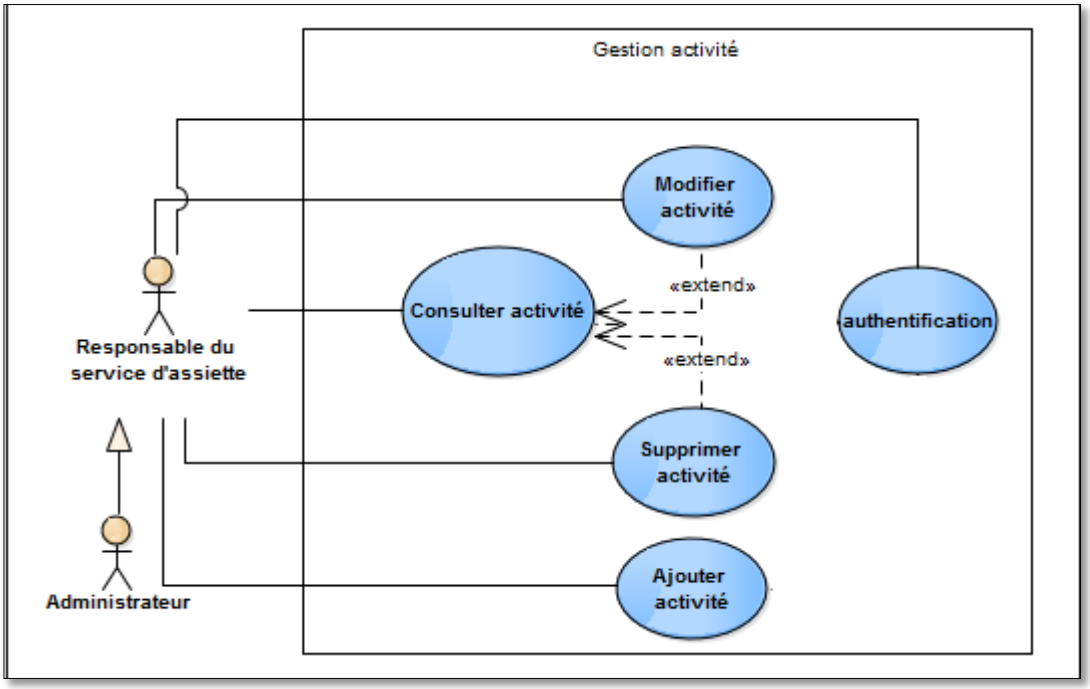


Figure 29 : Diagramme de cas d'utilisation « Gestion activité »

➢ Description du diagramme du cas d'utilisation pour «Gestion activité» :

Libellé	Acteur	Description	Précondition	Post condition
Ajouter activité	➢ Administrateur ➢ Responsable	Création d'une nouvelle activité.	Vérifier l'inexistence de l'activité à créer	Néant
Modifier activité		Modification de l'activité concernée.	Néant	Néant
Supprimer activité		Suppression d'activité concernée.	Vérifier qu'il n'est pas lié à un local	Néant

Consulter activité		Affichage des activités enregistrées dans la base de données dans un tableau.	Néant	Néant

Tableau 20 : Description du diagramme de cas d'utilisation « Gestion activité »

❖ Gestion des secteurs :

➢ Diagramme de cas d'utilisation « Gestion secteurs» :

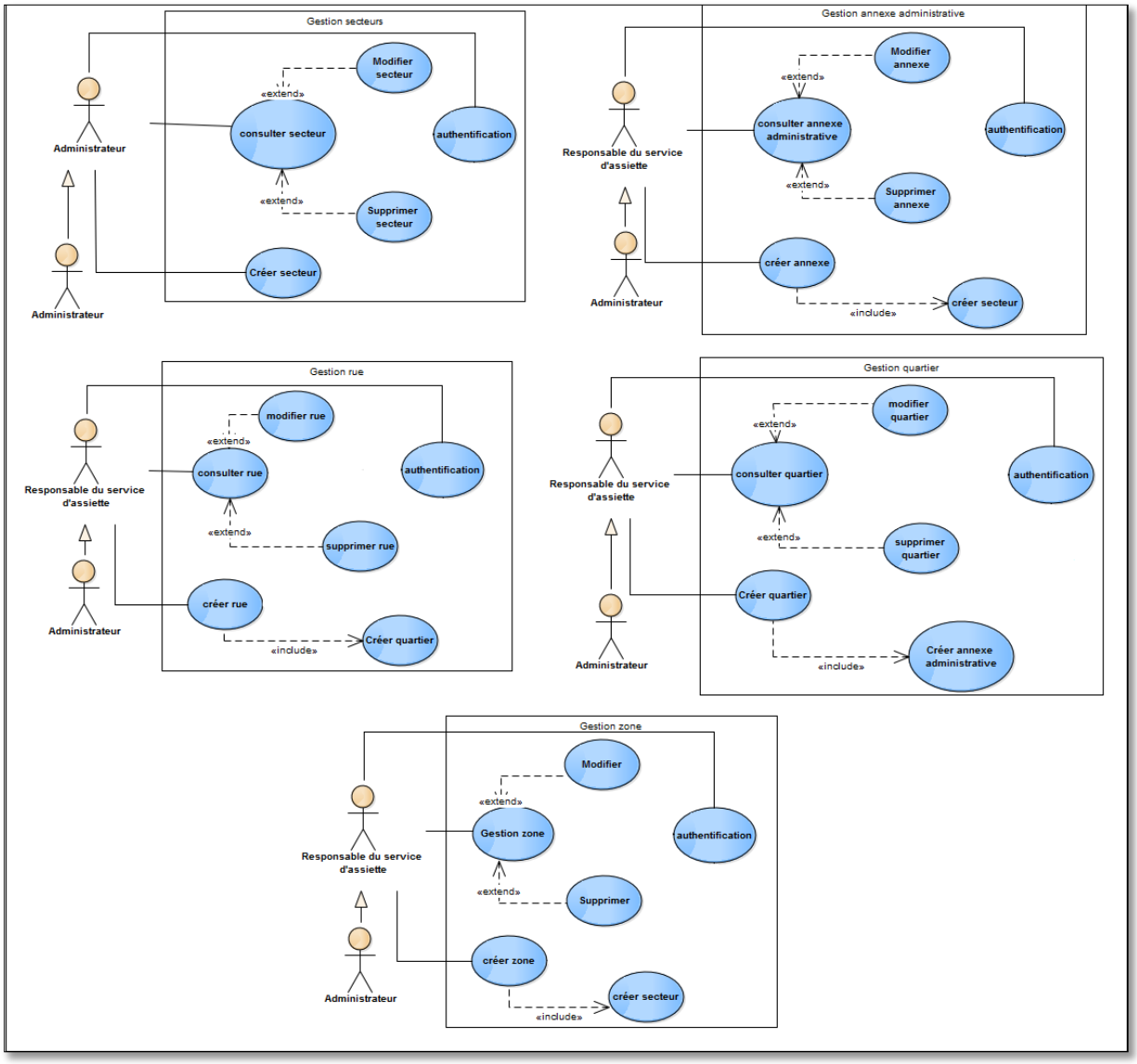


Figure 30 : Diagrammes des cas d'utilisations de chaque élément de l'adresse

➢ Description du diagramme du cas d'utilisation pour «Gestion adresse» :

Diagramme	Libellé	Description	Acteur	Précondition	Post condition
Gestion secteur	Créer secteur	Création d'un nouveau secteur.	♦ Administrateur ♦ Responsable	Vérifier l'inexistence du secteur	Néant
	Modifier secteur	Modification du secteur concerné		Néant	Néant
	Supprimer secteur	Suppression du secteur concerné.		Vérifier qu'il n'est pas lié à d'autres éléments	Néant
	Consulter secteur	Affichage de la liste des secteurs dans un tableau.		Néant	Néant
Gestion annexe administrative	Créer annexe	Création d'une nouvelle annexe administrative liée à un secteur.		-Vérifier l'inexistence de l'annexe administrative -Vérifier l'existence du secteur	Néant
	Modifier annexe	Modification de l'annexe administrative concernée.		Néant	Néant
	Supprimer annexe	Suppression de l'annexe administrative concernée.		Vérifier qu'il n'est pas lié à d'autres éléments	Néant
	Consulter annexe	Affichage de la liste des annexes administratives dans un tableau.		Néant	Néant
Gestion quartier	Créer quartier	Création d'un nouveau quartier lié à une annexe administrative.		-Vérifier l'inexistence du quartier -Vérifier l'existence de l'annexe administrative	Néant
	Modifier quartier	Modification du quartier concerné		Néant	Néant
	Supprimer quartier	Suppression du quartier concerné.		Vérifier qu'il n'est pas lié à d'autres éléments	Néant
	Consulter quartier	Affichage de la liste des secteurs dans un tableau.		Néant	Néant
Gestion rue	Créer rue	Création d'une nouvelle rue liée à un quartier.		-Vérifier l'inexistence de rue -Vérifier l'existence du quartier	Néant
	Modifier rue	Modification de la rue concernée.		Néant	Néant
	Supprimer rue	Suppression de la rue concernée.		Vérifier qu'il n'est pas lié à d'autres éléments	Néant
	Consulter rue	Affichage de la liste des rues dans un tableau.		Néant	Néant
Gestion zone	Créer zone	Création d'une nouvelle zone liée à un secteur.		-Vérifier l'inexistence de la zone	Néant

				-Vérifier l'existence du secteur	
	Modifier zone	Modification de la zone concernée.		Néant	Néant
	Supprimer zone	Suppression de la zone concernée.		Vérifier qu'il n'est pas lié à d'autres éléments	Néant
	Consulter zone	Affichage de la liste des zones dans un tableau.		Néant	Néant

Tableau 21 : Description des diagrammes de cas d'utilisation « Gestion adresse »

5. Diagrammes de séquences

Un cas d'utilisation décrit un ensemble de scénarios. Un scénario décrit une exécution particulière d'un cas d'utilisation du début jusqu'à la fin. Il correspond à une sélection d'enchaînements du cas d'utilisation. Il est clair que la combinatoire des enchaînements fait exploser le nombre de scénarios potentiels, nous ne pouvons pas donc tous les décrire. Alors les diagrammes de séquence que nous avons réalisée sont les suivants :

➢ SD1 : Authentification :

Vu la criticité des données relatifs aux taxes communales une authentification s'impose.

En fait, notre système permet le blocage des utilisateurs après trois tentatives erroné, et c'est l'administrateur qui peut débloquer l'utilisateur en question.

Ce diagramme représente les différentes étapes d'authentification par un régisseur, le responsable et l'administrateur procèdent de la même manière.

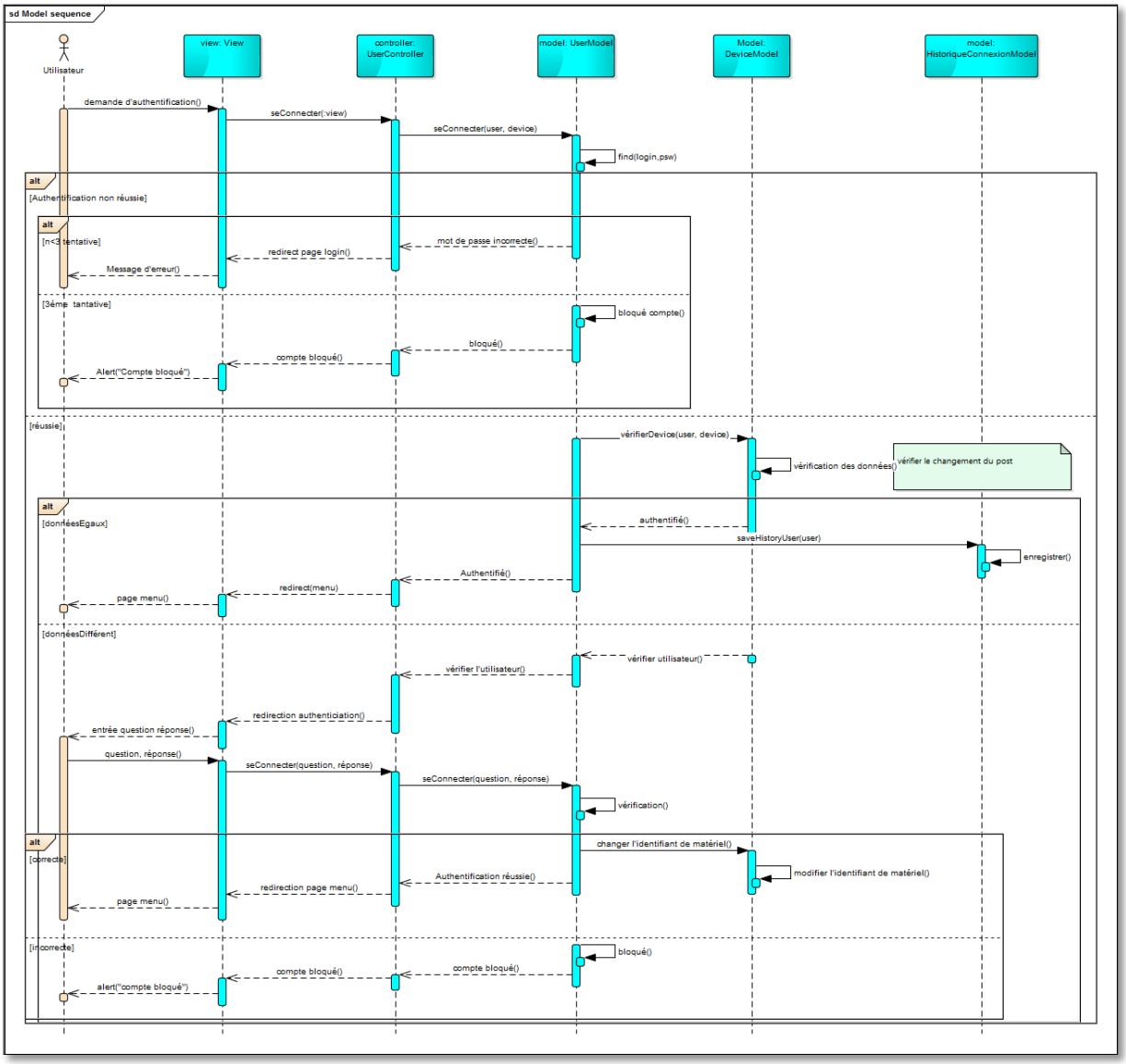


Figure 31 : Diagramme de séquence de l'authentification

➢ SD2 : Rechercher un redevable via le site du tribunal commercial :

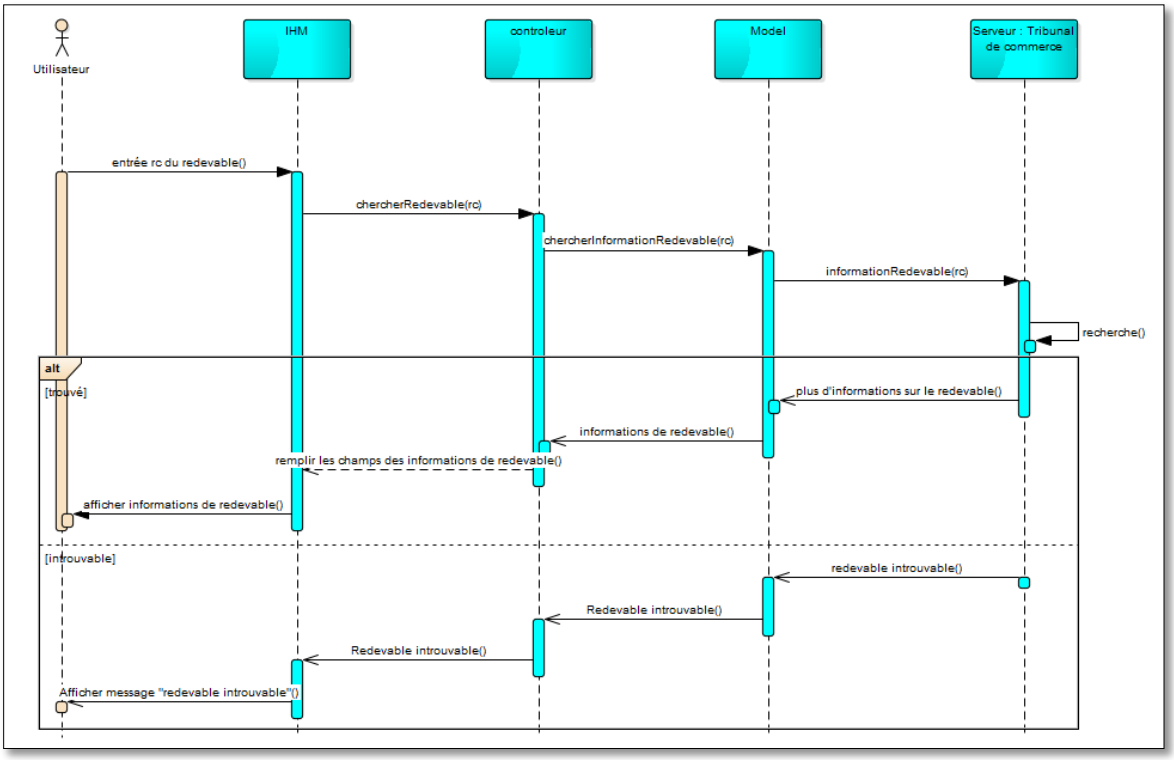


Figure 32 : Diagramme de séquence de la recherche d'un redevable via le site du tribunal commercial

➢ SD3 : Créer un redevable

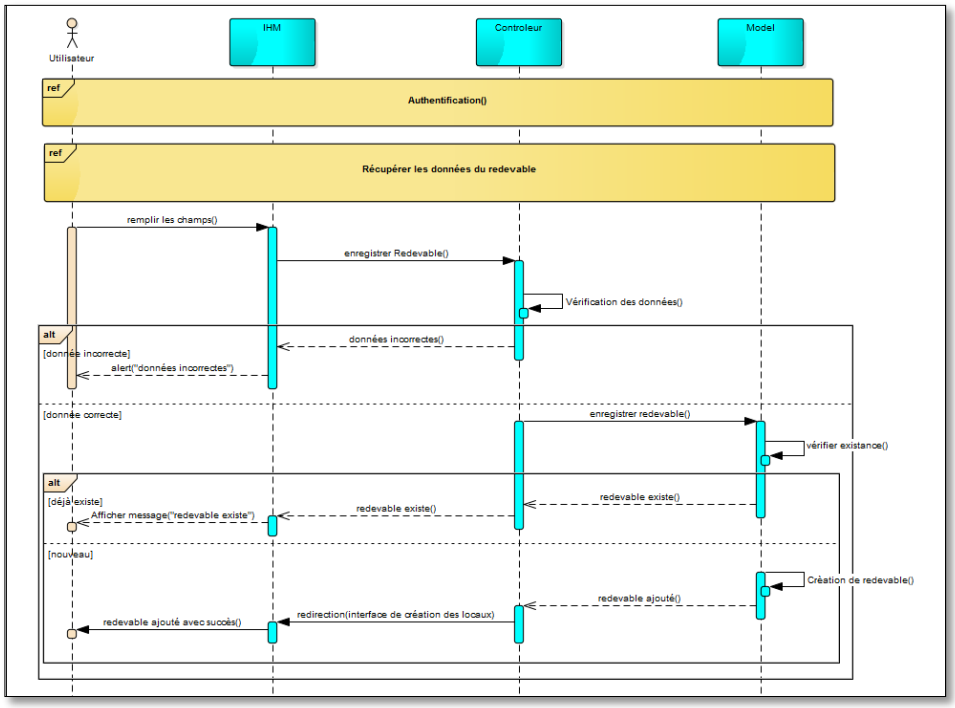


Figure 33 : Diagramme de séquence de la création d'un redevable

Ce diagramme ainsi que le diagramme « Figure 32 » ci-dessus, représentent le processus de création d'un redevable par un régisseur, le responsable et l'administrateur procèdent de la même manière.

➢ SD5 : Création d'un local :

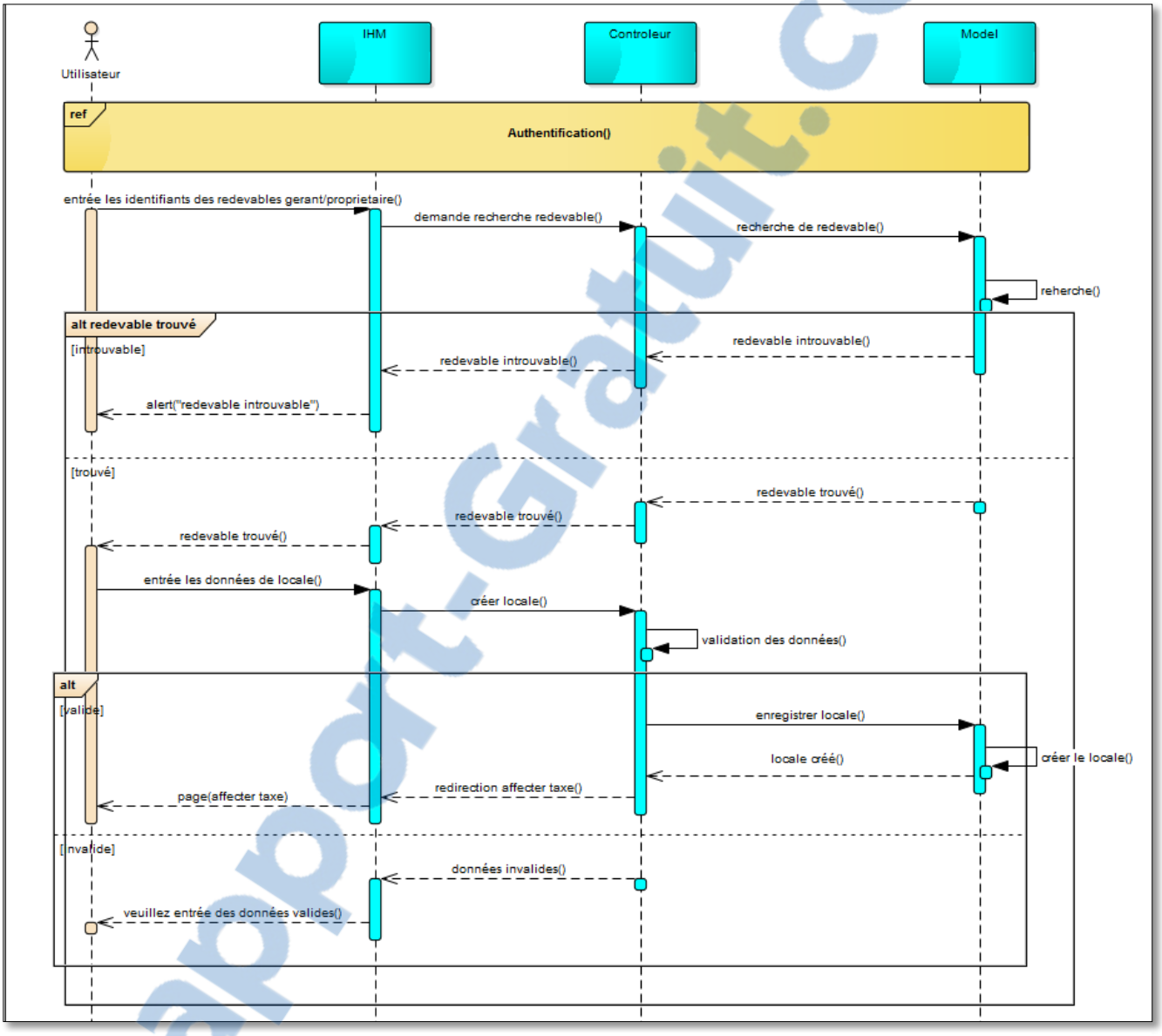


Figure 34 : Diagramme de séquence de la création d'un local

Ce diagramme représente le processus de création d'un local par un régisseur, le responsable et l'administrateur procèdent de la même manière.

Rapport-Gratuit.com

➢ SD6 : Paiement d'une taxe trimestrielle :

Le scenario décrit par ce diagramme est valable pour les taxes trimestrielles (taxe boisson, séjour, 37 et 38), sauf une différence au niveau des données à saisir, ainsi que les taux à appliquer.

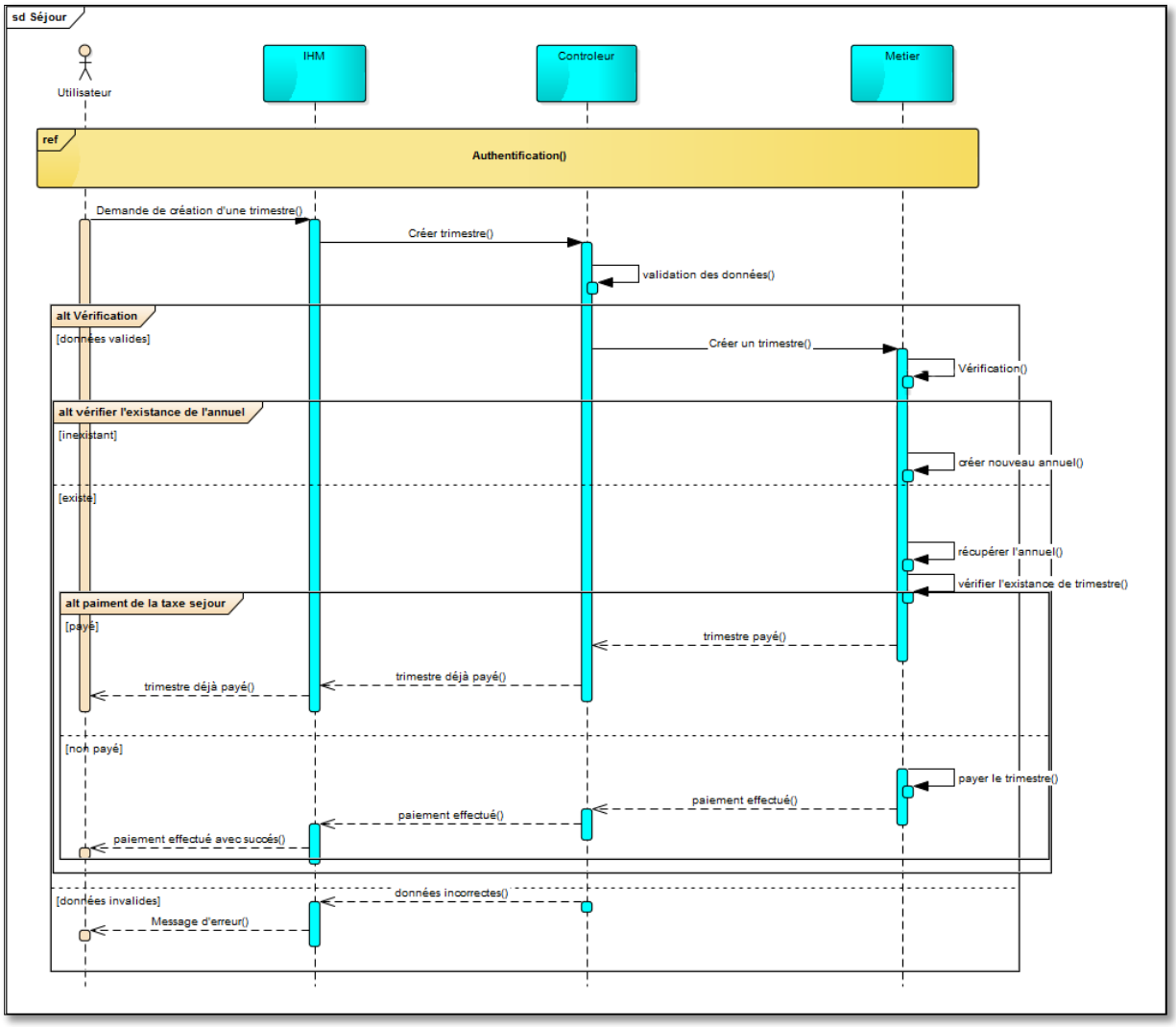


Figure 35 : Diagramme de séquence du paiement d'une taxe trimestrielle

➢ SD7 : Paiement d'une taxe annuelle (Taxe TNB) :

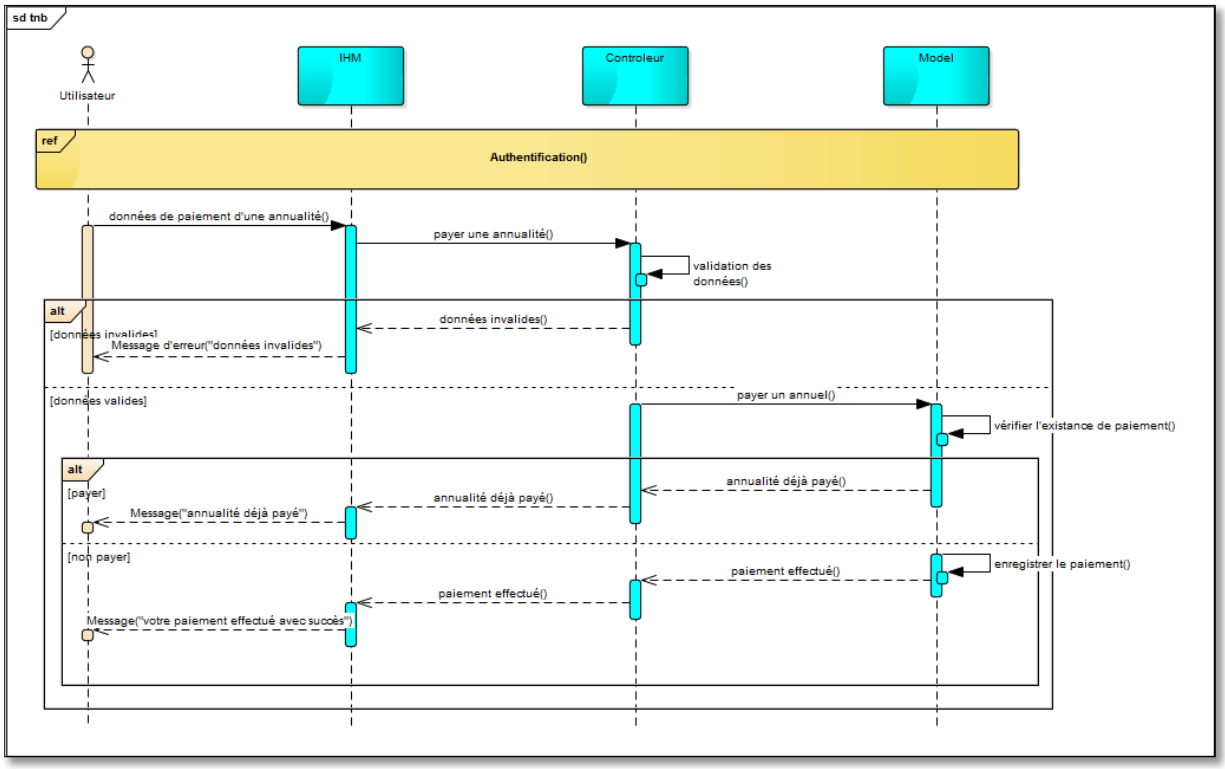


Figure 36 : Diagramme de séquence « Paiement taxe TNB »

Ce diagramme représente le processus du paiement annuel de taxe TNB par un régisseur, le responsable et l'administrateur procèdent de la même manière.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

➢ SD8 : Préparer une notification :

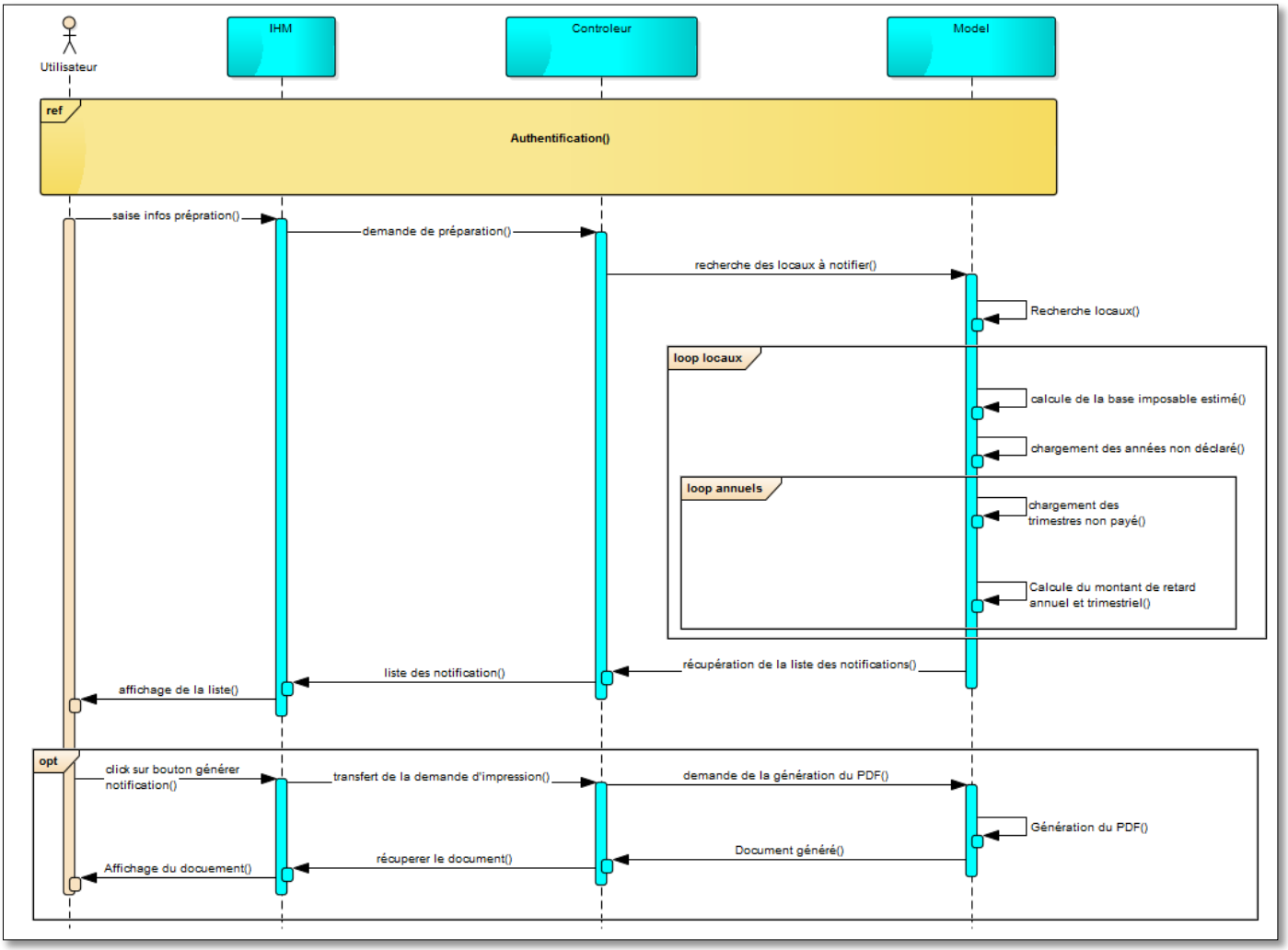


Figure 37 : Diagramme de séquence de la préparation des notifications

Ce diagramme représente le processus de la préparation des notifications par un responsable, l'administrateur procède de la même manière.

➢ **SD9 : Statistiques :**

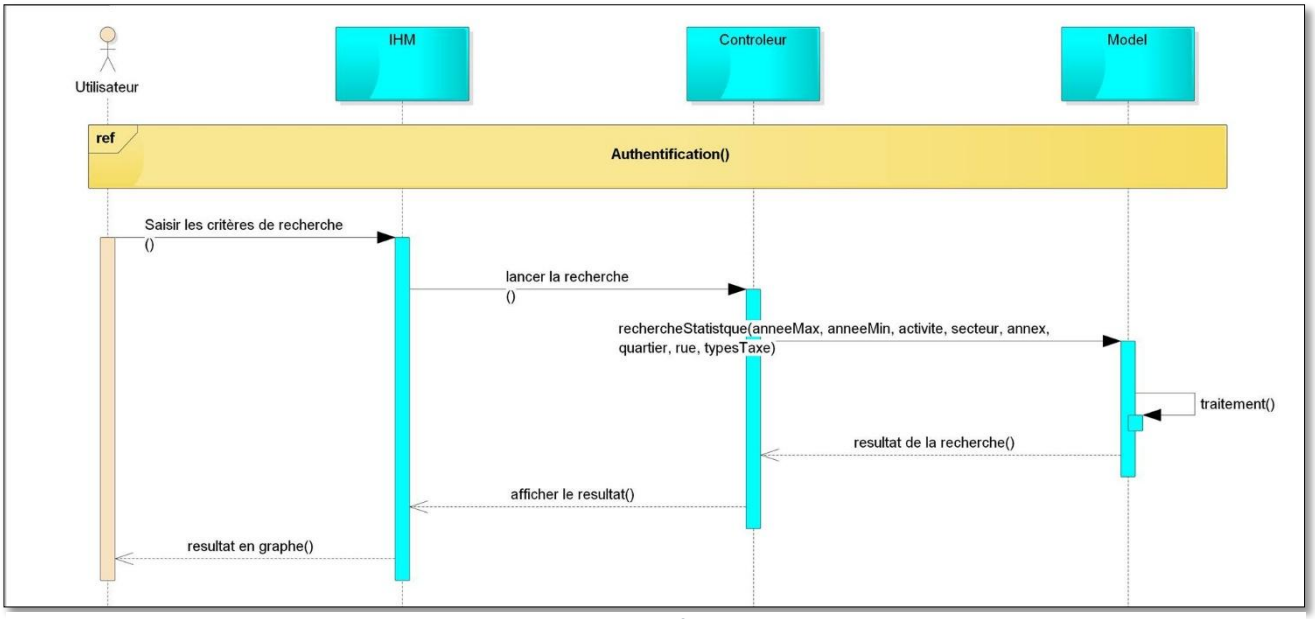


Figure 38 : Diagramme de séquence « Statistiques »

Ce diagramme représente le processus de la génération et consultation des graphes statistiques par le responsable et l'administrateur procèdent de la même manière.

a. Diagramme de classes

Le diagramme de classes permet de représenter l'aspect statique du système en termes de classes et des relations entre ces classes, en identifiant les différents attributs et méthodes qui reflètent respectivement les propriétés et les services offerts par notre système.

❖ **Découpage en packages :**

Le diagramme de packages/diagramme de paquetages permet de découper l'ensemble des classes (d'un modèle de classes) en sous-ensembles de classes en fonction de leurs dépendances, d'une tâche commune.

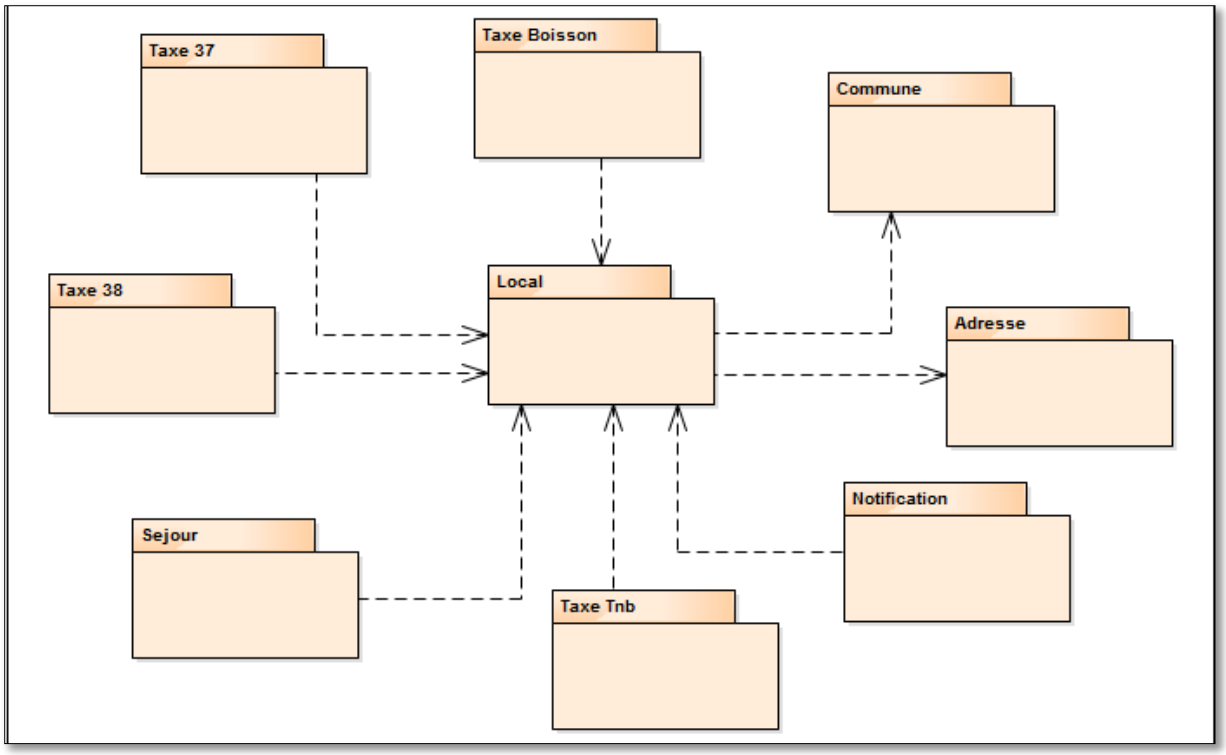


Figure 39 : Diagramme de packages

Liste des classes par package :

Package	Classes
Local	Local, Redevable, Activite
Adresse	Secteur, AnnexeAdministrative, Quartier, Rue, Zone, Polygone, PositionPolygone
Commune	Commune, Ville, User, Groupe, UserConnexionHistorique
Taxe boisson	TaxeBoissonAnnuelle,TaxeBoissonTrimestrielle, DeclarationExistance, DeclarationExistancePieceJointe, TauxRetardTaxeBoissonAnnuelle, TauxTaxeBoissonTrimestrielle, TauxRetardTaxeBoissonTrimestrielle, TauxRetardDeclarationExistance
Taxe Séjour	TaxeSejourAnnuelle, TaxeSejourTrimestrielle, TauxRetardTaxeSejourAnnuelle, TauxRetardTaxeSejourAnnuelleItem, TauxTaxeSejour, TauxTaxeSejourItem, TauxRetardTaxeSejourTrimestrielle, CategorieSerjour
Taxe 37	Taxe37Annuelle, Taxe37Trimestrielle, Taxe37TrimestrielleItem, TauxRetardTaxeSejourAnnuelle, TauxRetardTaxeSejourAnnuelleItem, TauxTaxe37, TauxTaxe37Item, TauxRetardTaxe37, TauxRetardTaxe37Item, Categorie37, Local37Item
Taxe 38	Taxe38Annuelle, Taxe38Trimestrielle, Taxe38TrimestrielleItem, TauxRetardTaxe38Annuelle, TauxRetardTaxe38AnnuelleItem, TauxTaxe38, TauxTaxe38Item, TauxRetardTaxe38, TauxRetardTaxe38Item, Categorie38, Local38Item
Taxe TNB	Terrain, TaxeAnnuelleTnb, TauxTaxeTnb, TauxTaxeTnbItem, TauxRetardTaxeTnb, TauxRetardTaxeTnbItem, Exoneration,ExonerationItem, CategorieHabitation, NotificationTnb,NotificationTnbItem
Notification	Notification, NotificationItem

Tableau 22 : Classes du système par package

Après avoir déterminé les packages du système il fallait élaborer un diagramme de classes pour chaque package pour bien illustrer la structure interne de chacun.

- **Package Local**

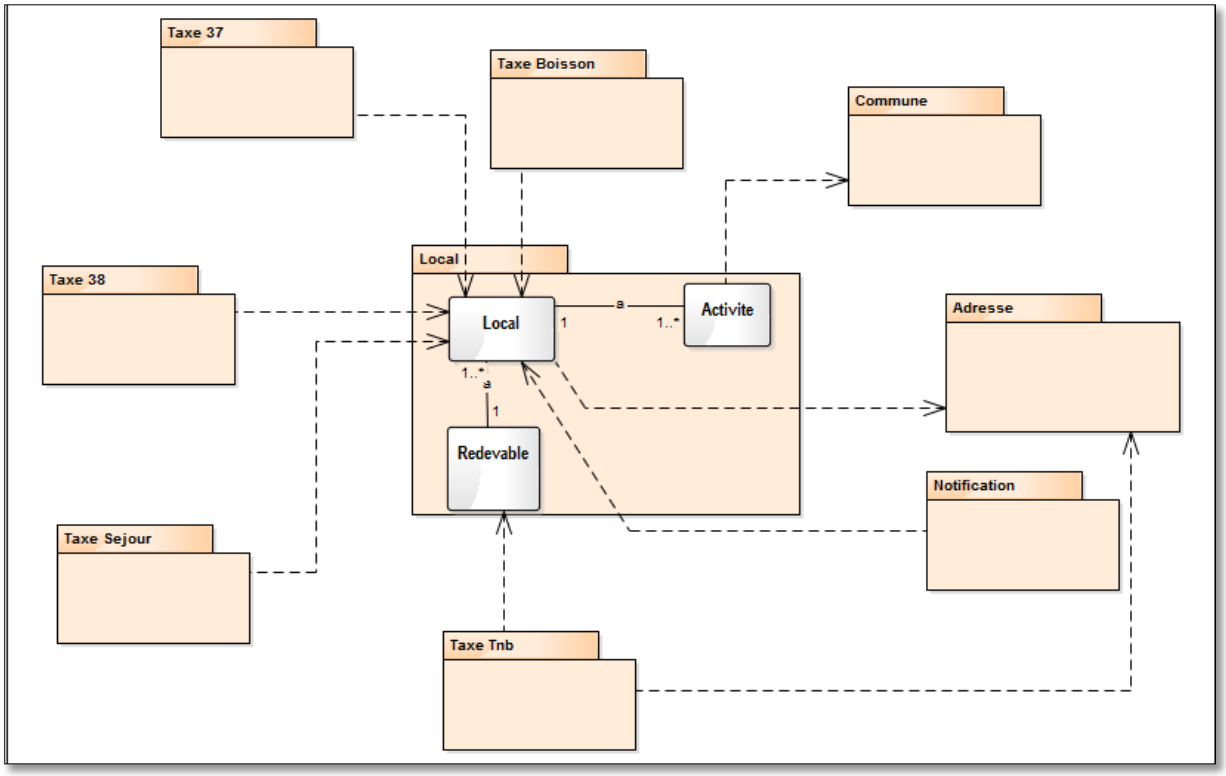


Figure 40 : Diagramme de classes « Package Local »

- **Package Boisson**

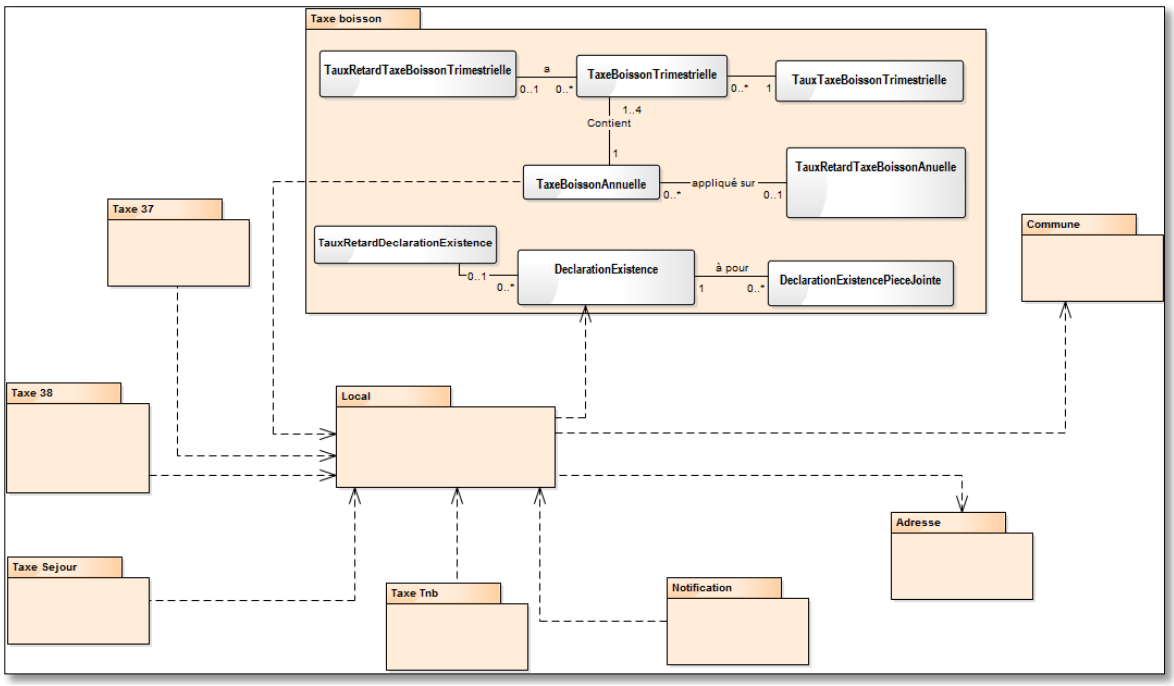


Figure 41 : Diagramme de classes « Package boisson »

➢ **Diagramme de classes taxe boisson**

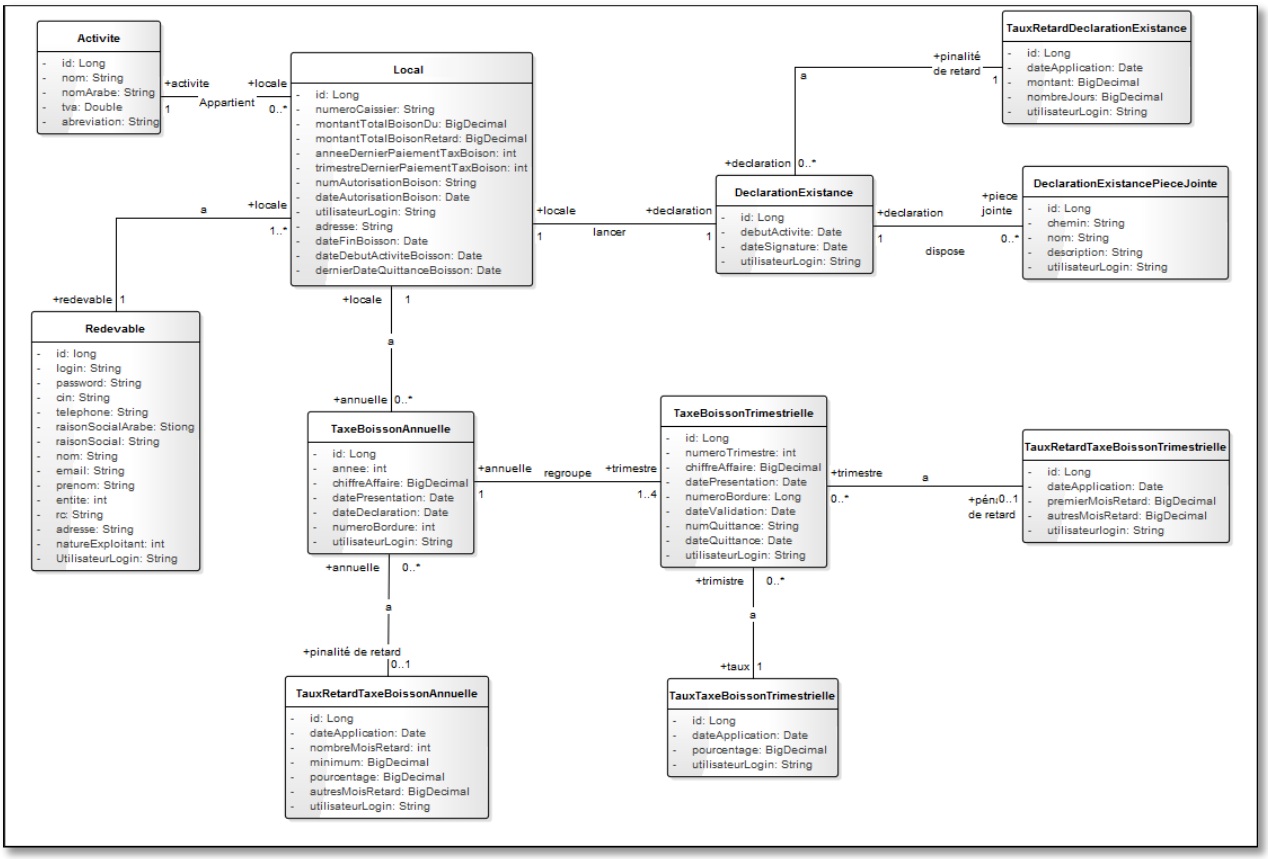


Figure 42 : Diagramme de classes « Gestion taxe boisson » avec les attributs

✓ **Description :**

Redevable :

Cette classe représente les éléments de base pour la création d'un redevable à savoir (nom, prénom, CIN, RC, raison sociale, téléphone…etc.), chaque redevable à un ou plusieurs locaux.

Local :

Cette classe représente les éléments de base pour la création d'un local à savoir (ICE, nom commercial, l'adresse ainsi que les montants totales de chaque taxe), un local peut payer une ou plusieurs taxe à la fois (Séjour ou Boisson ou 37,38).

TaxeBoissonAnnuelle :

Cette classe correspond à la taxe de boisson annuelle, le paiement de cette taxe consiste à préciser le redevable et locale concerné, l'année à payer, le chiffre d'affaire, la TVA selon l'activité du local (classe « Activite »), et la date de présentation, le numéro de quittance et la date de quittance. Le taux et le taux, représentés par la classe « TauxTaxeBoissonTrimetrielle »est appliqué pour calculer le montant de la taxe.

Chaque taxe de boisson annuelle est composée de quatre trimestres.

Un taux de retard est représenté par la classe « TauxRetardTaxeBoissonTrimestrielle » est appliqué pour calculer les majorations lors qu'il s'agit d'un paiement hors délai. Le taux de retard annuel, représenté par la classe « TauxRetardTaxeBoissonAnnuelle » est appliqué pour calculer une majoration, quand il s'agit d'un défaut de déclaration.

Le paiement d'un trimestre est représenté par la classe « TaxeBoissonTrimestrielle ».

DeclarationExistence :

Un local à une et une seule déclaration d'existence, qui est représentée par la classe « DeclarationExistence », cette classe peut avoir plusieurs pièces jointes, représentés par la classe « DeclarationExistencePieceJointe » qui contient les éléments d'un document attaché à la déclaration.

- La relation entre les taux et d'autres classes a été établi pour garder une trace de calcul

- **Package séjour**

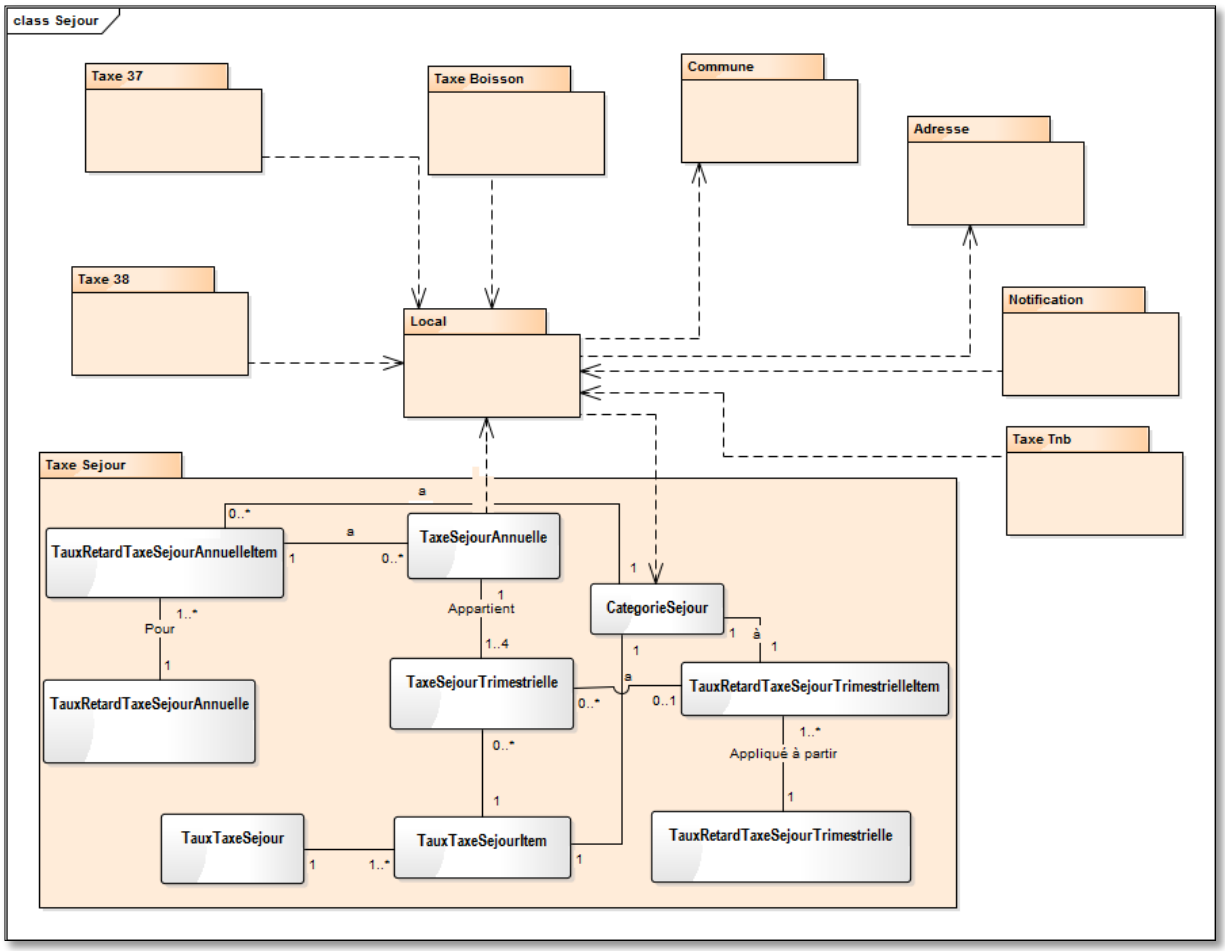


Figure 43 : Diagramme de classes « Package séjour »

- **Diagramme de classe taxe séjour**

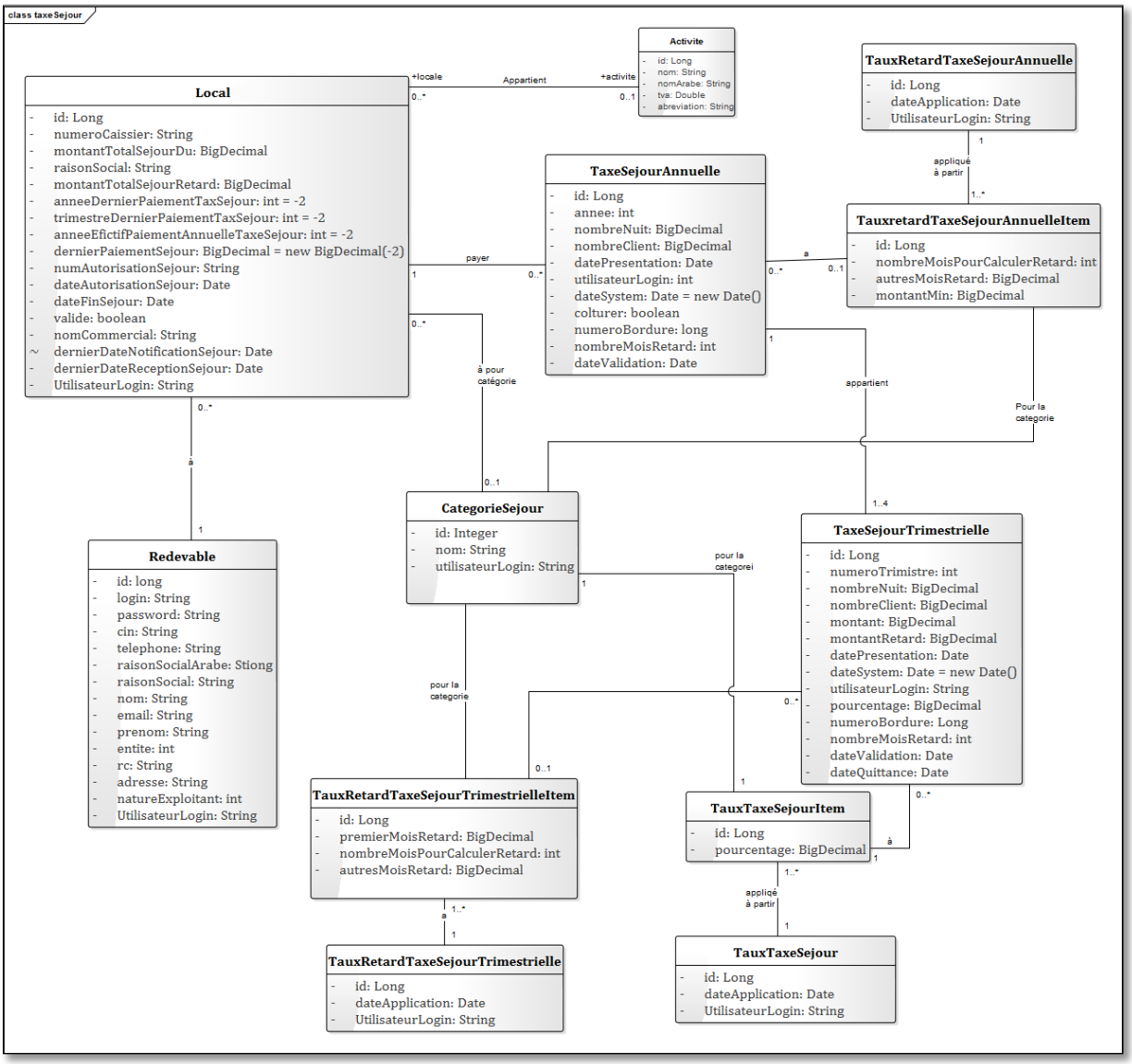


Figure 44 : Diagramme de classes « Gestion taxe séjour » avec les attributs

✓ Description :

<u>TaxeSejourAnnuelle :</u>

Cette classe correspond à la taxe de séjour annuelle, le paiement de cette taxe consiste à préciser le redevable et le locale concerné, l'année à payer, le nombre de nuitées, le nombre de clients, la TVA selon l'activité (Classe «**Activite** ») du local, la date de présentation, le numéro de quittance et la date de quittance.

Le taux appliqué pour calculer le montant de la taxe est défini en fonction de la catégorie de séjour (classe « **CategorieSejour** »). Le taux est représenté par la classe « **TauxTaxeSejourTrimestrielle** ». Le taux de retard trimestriel représenté par la classe « **TauxRetardTaxeSejourTrimestrielle** » est appliqué pour calculer une majoration quand il s'agit d'un paiement hors délai, chaque taxe de séjour annuelle est composée de quatre taxes de séjour trimestrielles (la classe « **TaxeSejourTrimestrielle** »).

Un taux de retard représenté par la classe « **TauxRetardTaxeAnnuelleItem** » est appliqué pour calculer une majoration, quand il s'agit d'un défaut de déclaration.

- La relation entre les taux et d'autres classes a été établi pour garder une trace de calcul.

➢ **Package Taxe 37**

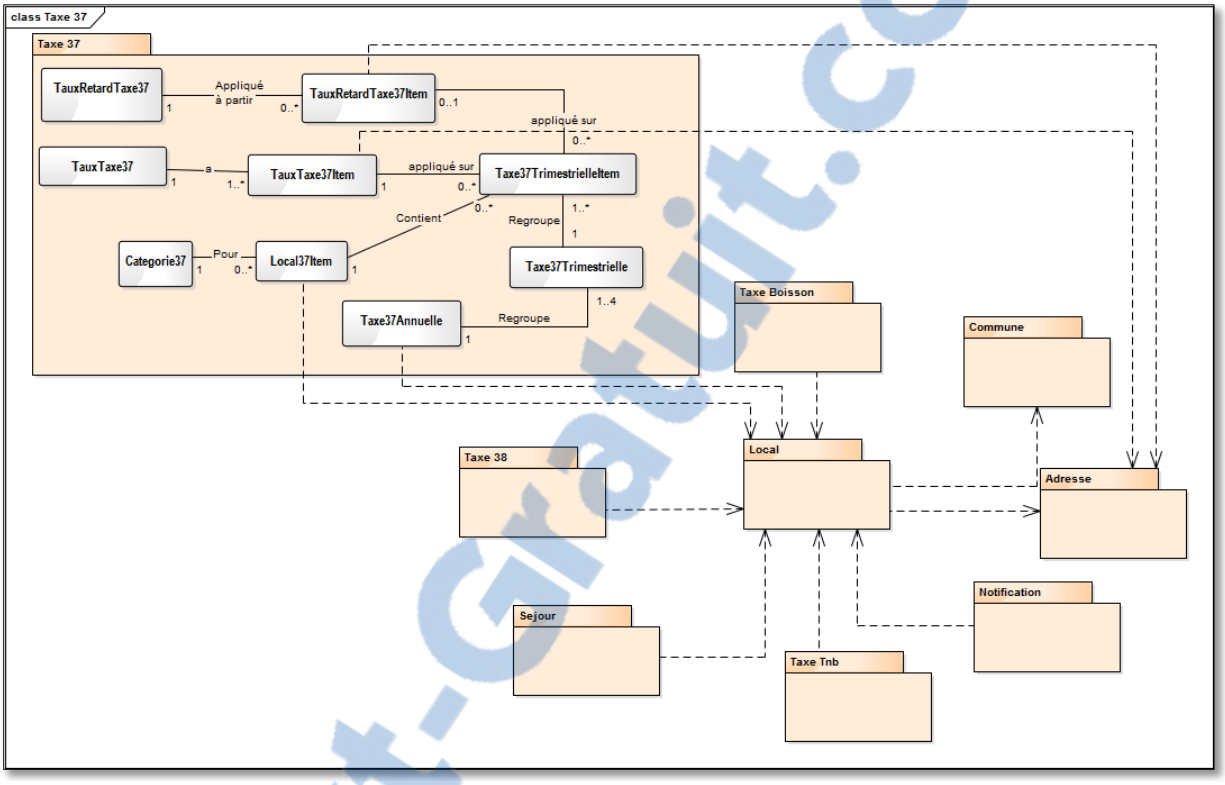


Figure 45 : Diagramme de classes "Package Taxe 37"

➢ **Diagramme de classes Taxe 37 :**

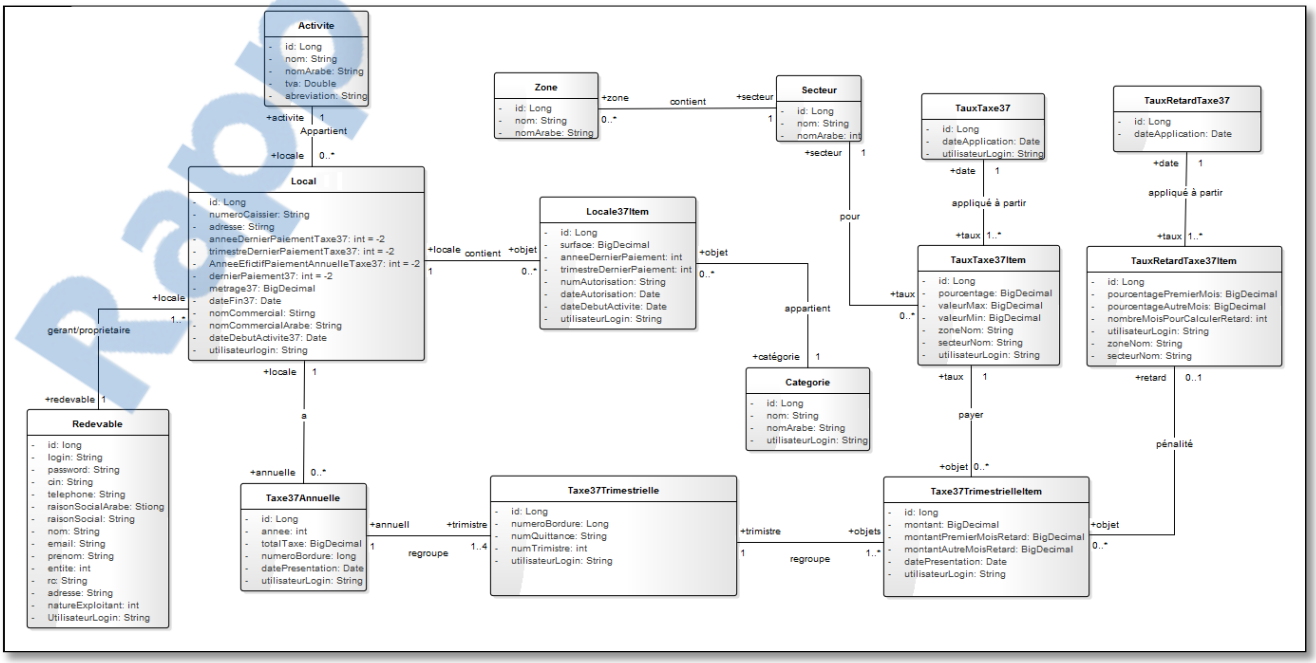


Figure 46 : Diagramme de classes « Gestion taxe 37 » avec les attributs

✓ Description :

Taxe37Annuelle :

Cette classe correspond à la taxe 37 annuelle, le paiement de cette taxe consiste à fournir le redevable et son local, la date de présentation ainsi que les éléments associés au local concerné (objet de taxe 37 représenté par la classe « **Local37Item** » et caractérisé par une catégorie, la surface et la valeur locative). Chaque taxe 37 annuelle est composée de quatre taxes 37 trimestrielles représenté par la classe « **Taxe37Trimestrielle** » et cette dernière elle-même est constituée des éléments représentés par la classe «**Taxe37TrimestrielleItem** » qui correspond au paiement de chaque objet de cette taxe « **Local37Item** ». Pour calculer le montant à payer, un taux est appliqué, ce dernier correspond à la classe « **TauxTaxe37Item**) » et « **TauxRetardTaxe37Item** » lorsqu'il s'agit du calcul de la majoration en cas d'un paiement hors délai, ce taux est précisé en fonction de la catégorie de l'élément lié au local ainsi que le secteur et/ou la zone du local en question.

- La relation entre les taux et d'autres classes a été établi pour garder une trace de calcul.

➢ Package Taxe 38

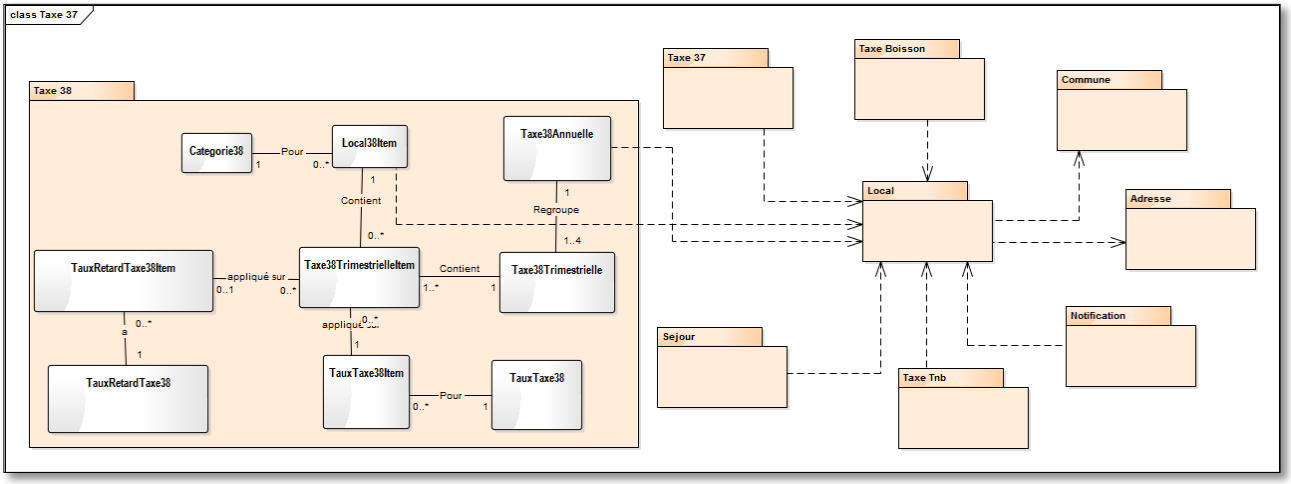


Figure 47 : Diagramme « Package Taxe 38 »

➢ **Diagramme de classes « Package Taxe 38 » :**

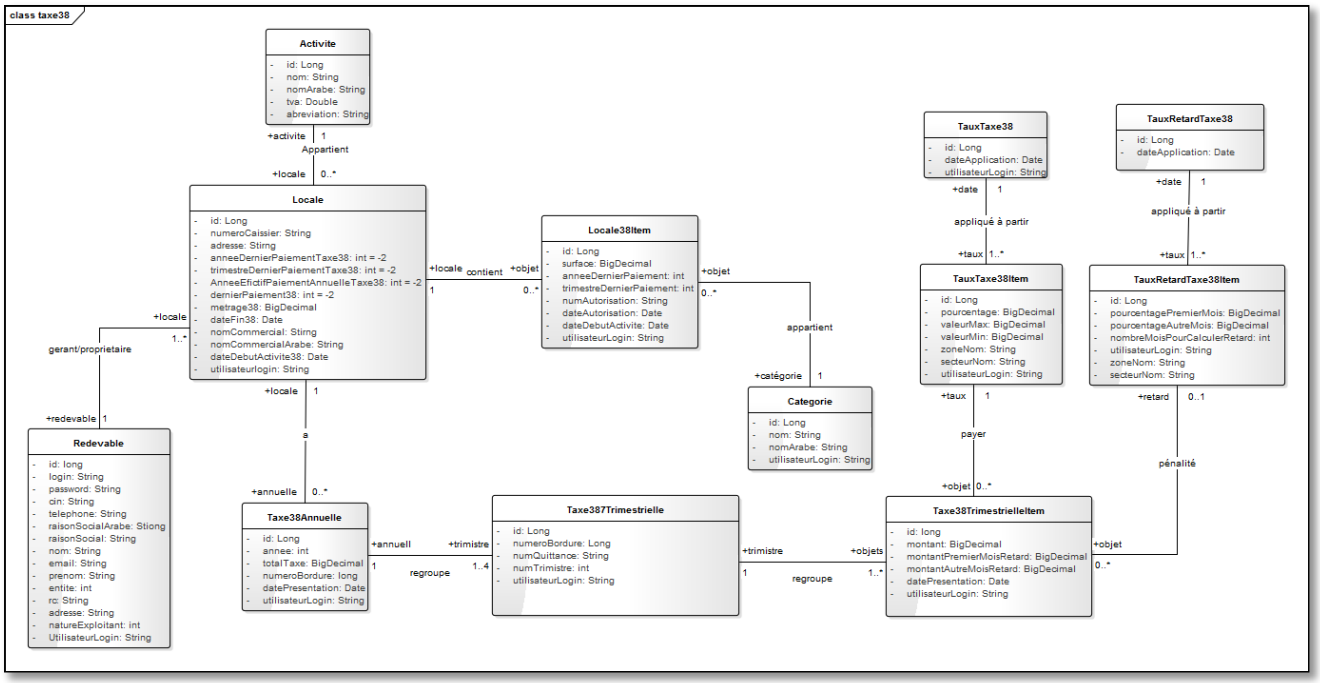


Figure 48 : Diagramme de classes « Gestion taxe 38 » avec les attributs

✓ **Description :**

<u>Taxe38Annuelle :</u>

Cette classe correspond à la taxe 38 annuelle, le paiement de cette taxe consiste à fournir le redevable et son local, la date de présentation ainsi que les adhésives (objet de taxe 38 représenté par la classe « **Local38Item** » et caractérisé par la catégorie et la surface).

Chaque taxe annuelle est composée de quatre taxes 38 trimestrielles représentés par la classe « **Taxe37Trimestrielle** » et cette dernière elle-même est constituée des éléments représentés par la classe « **Taxe37TrimestrielleItem** » qui correspond au paiement de chaque objet de cette taxe « **Local38Item** ». Lors du calcul du montant de la taxe, un taux est appliqué, ce dernier est précisé en fonction de la catégorie de l'adhésive et qui sont représentés par les classes « **TauxTaxe38Item** » et « **TauxRetardTaxe38Item** » lorsqu'il s'agit du calcul de la majoration en cas d'un paiement hors délai.

- La relation entre les taux et d'autres classes a été établi pour garder une trace de calcul.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

➢ Package Taxe TNB

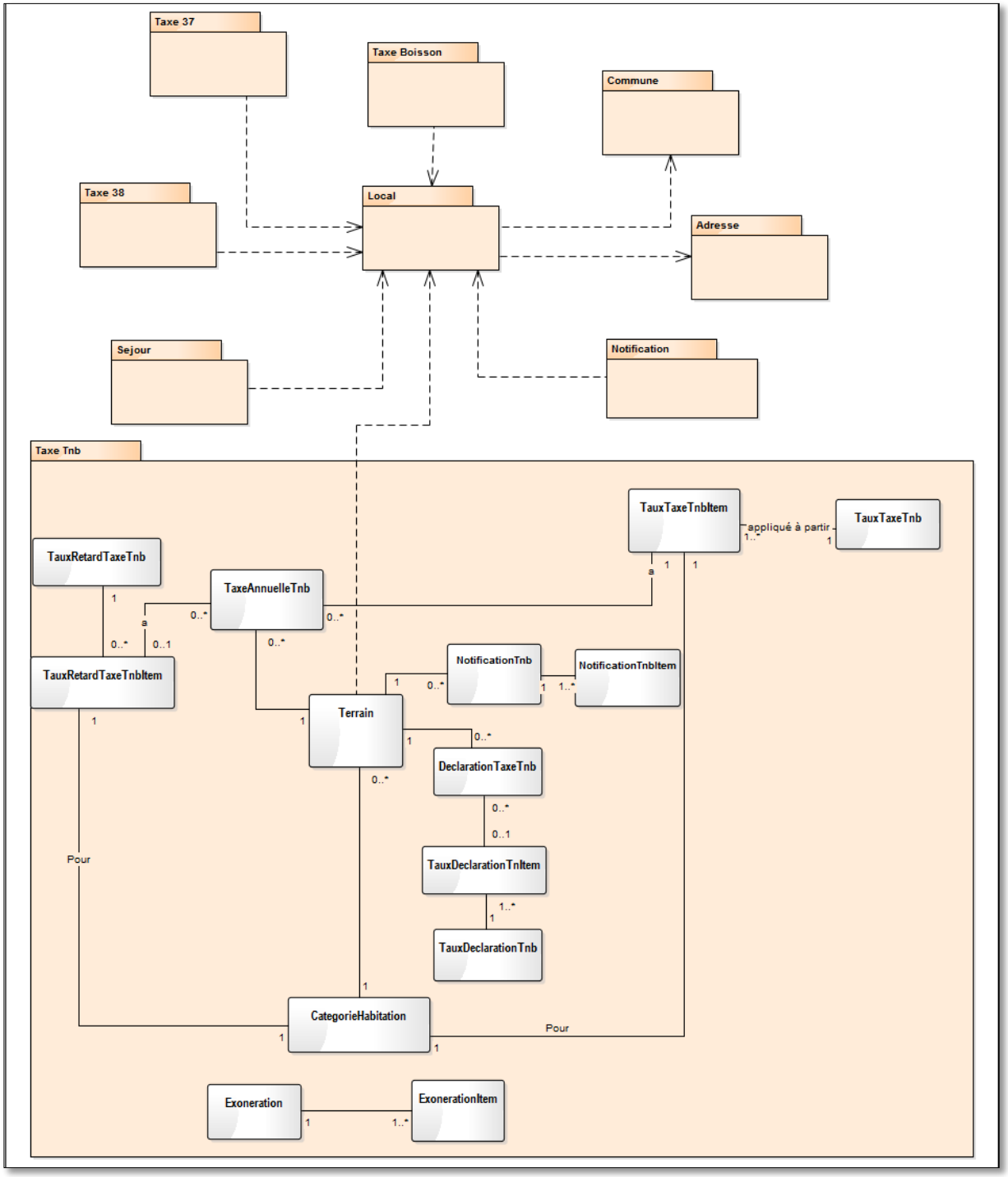


Figure 49 : Diagramme « Package Taxe TNB »

➢ **Diagramme de classes « Package Taxe TNB » :**

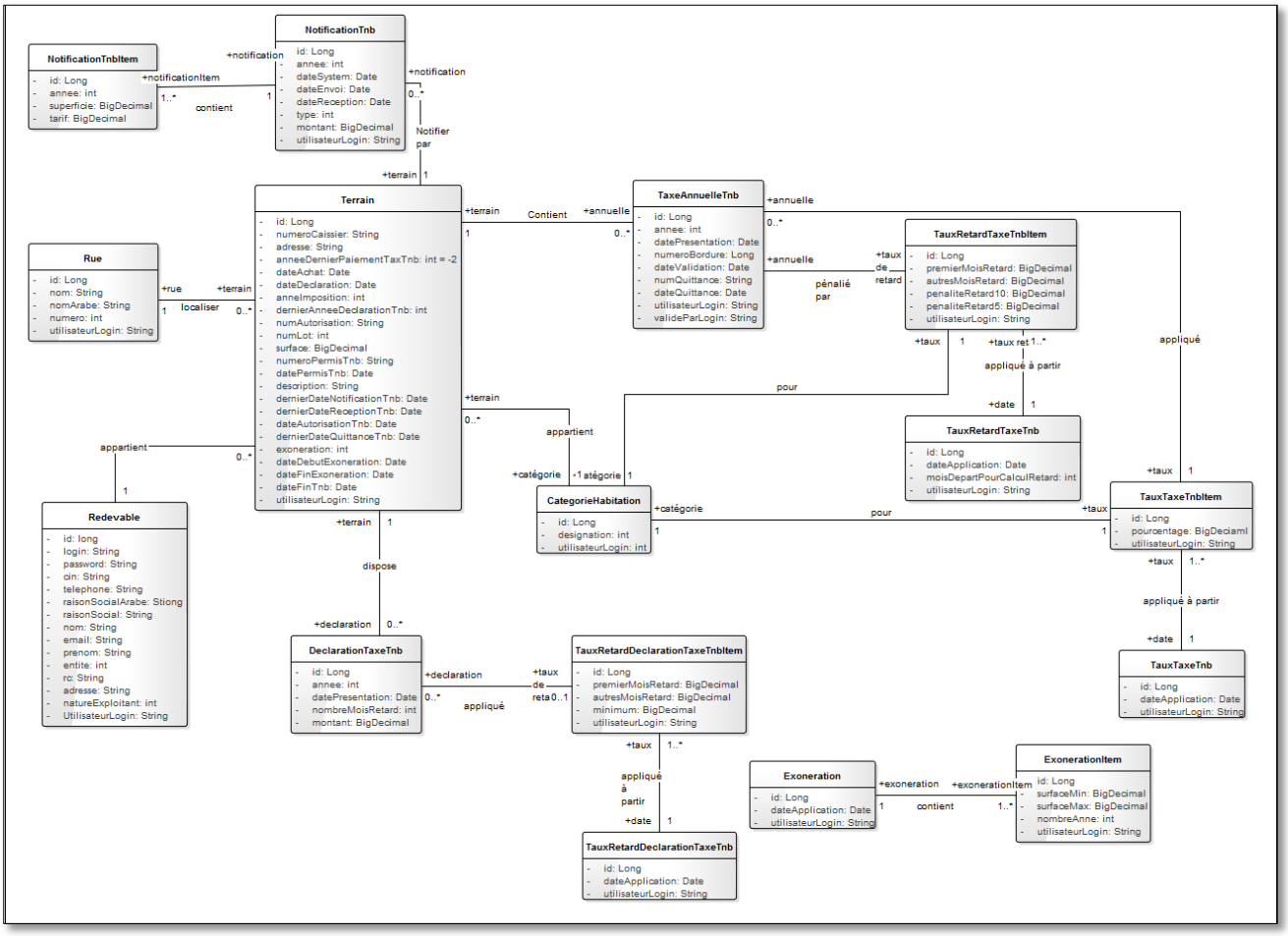


Figure 50 : Diagramme de classes « Gestion taxe TNB » avec les attributs

✓ Description :

<u>Terrain :</u>

Cette classe représente les éléments de base pour la création d'un terrain à savoir (Superficie, N° lot, date d'autorisation, année d'imposition, date d'achat...), un terrain appartient à une catégorie d'habitation «classe **CategorieHabitation** ». Chaque terrain à zéro ou plusieurs taxes annuelles représentées par la classe « **TaxeAnnuelleTnb** ».

<u>TaxeAnnuelleTnb :</u>

Cette classe représente les éléments de base pour la création d'une taxe de TNB (terrain non bâtis), à savoir le redevable et son local, année, date de présentation. Pour calculer le montant de la taxe, un taux est appliqué, ce dernier est précisé en fonction de la catégorie d'habitation du terrain concerné. Le taux représenté par la classe « **TauxTaxeTnbItem** » et « **TauxRetardTaxeTnb** » pour calculer la majoration lorsqu'il s'agit d'un paiement hors délai.

La déclaration annuelle d'un terrain est représentée par la classe « **DeclarationTaxeTnb** » avec un taux appliqué en cas de retard qui correspond à la classe « **TauxRetardDeclarationTnbItem** » qui est précisé en fonction de la catégorie d'habitation.

Le nombre d'année d'exonération est définit en fonction de la superficie du terrain, ces exonérations sont stocké dans la table qui correspond à la classe «**ExonérationItem** », cette dernière appartient à « Exonération » qui contient la date d'application desdites exonérations « **ExonerationItem** ».

- La relation entre les taux et d'autres classes a été établi pour garder une trace de calcul.

- **Package Notification**

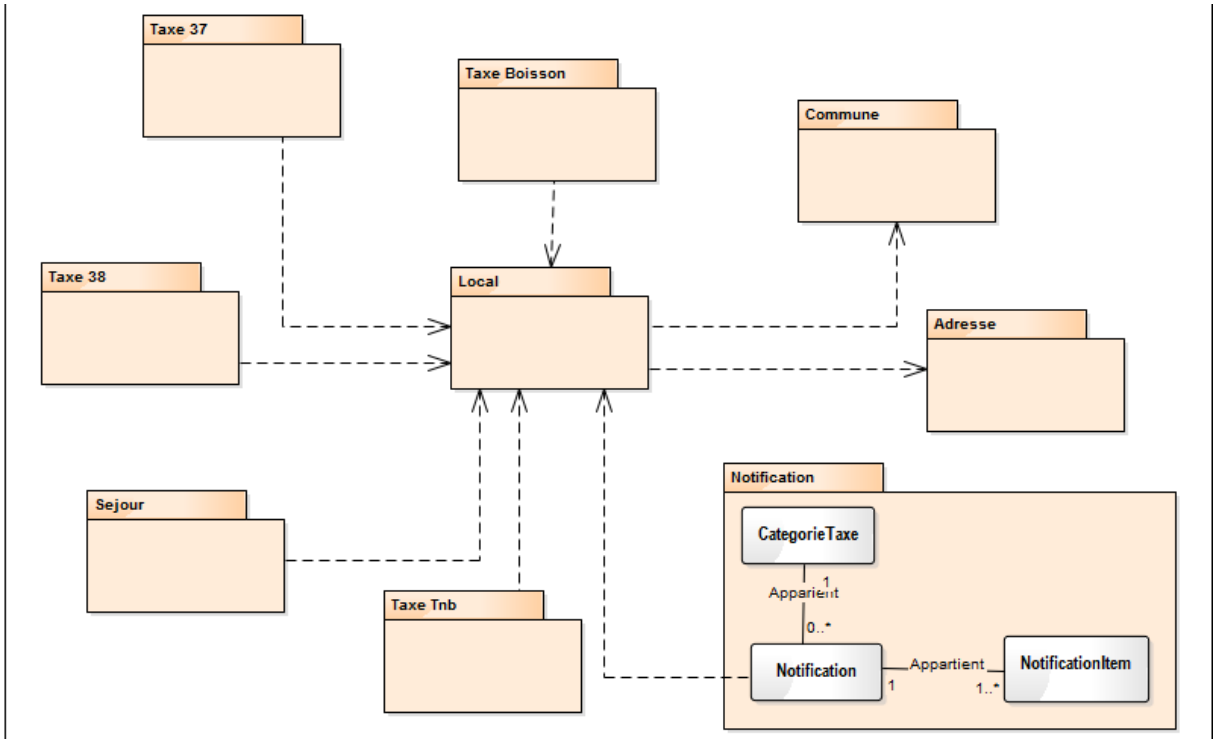


Figure 51 : Diagramme de classe « Package Notification »

- **Diagramme de classes « Package Notification »**

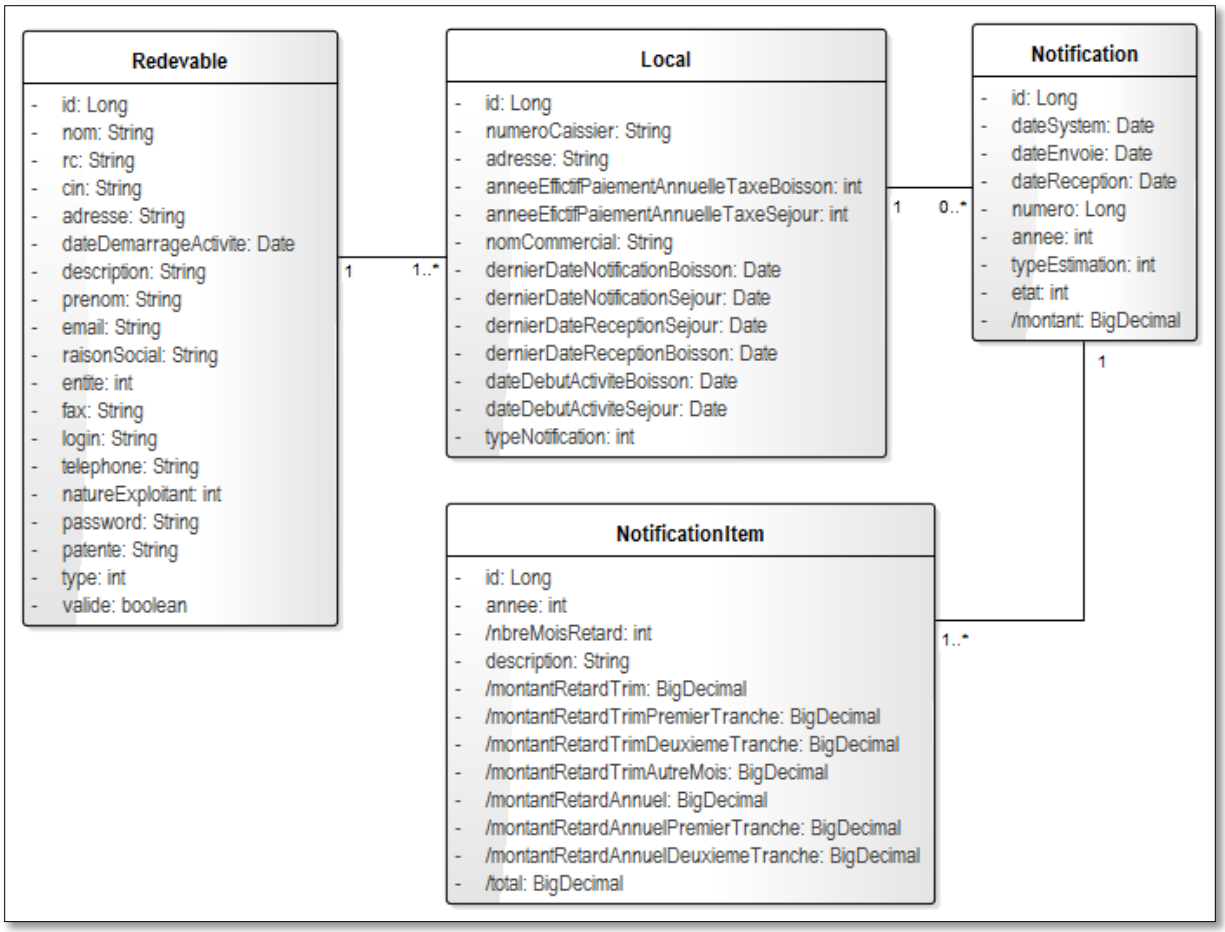


Figure 52 : Diagramme de classes « Gestion notifications » avec les attributs

✓ Description :

<u>Notification :</u>

Chaque local à une ou plusieurs notifications caractérisées par le numéro d'article, le type d'estimation, le montant estimé, l'année à notifier, et la date d'envoi et de réception. Chaque notification est composée des éléments (classe « **NotificationItem** ») qui représentent chaque année (non déclarer) notifié avec ses détails à savoir les trimestres non payé, le nombre de mois de retard et le montant de retard. Cette description est valable pour les notifications de la taxe TNB avec des classes intitulé « **NotificationTnb** » et « **NotificationTnbItem** » sauf qu'il s'agit de notifier les terrains dont des années ne sont pas payé.

➢ Package Commune

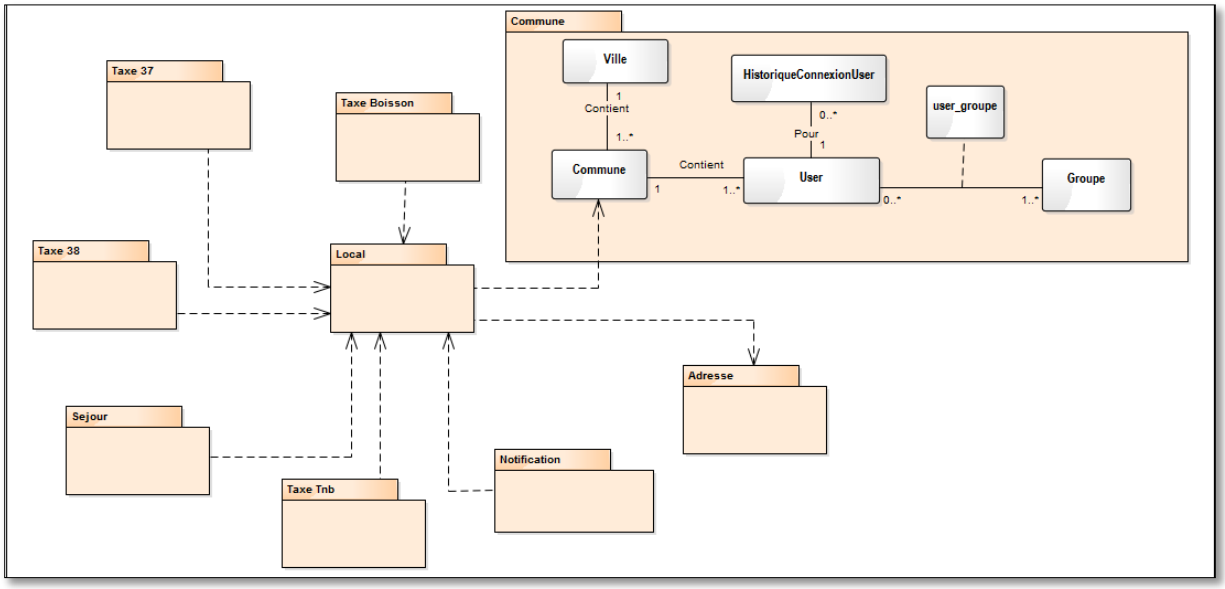


Figure 53 : Diagramme de classes « Package Commune »

➢ Diagramme de classes « Package Commune »

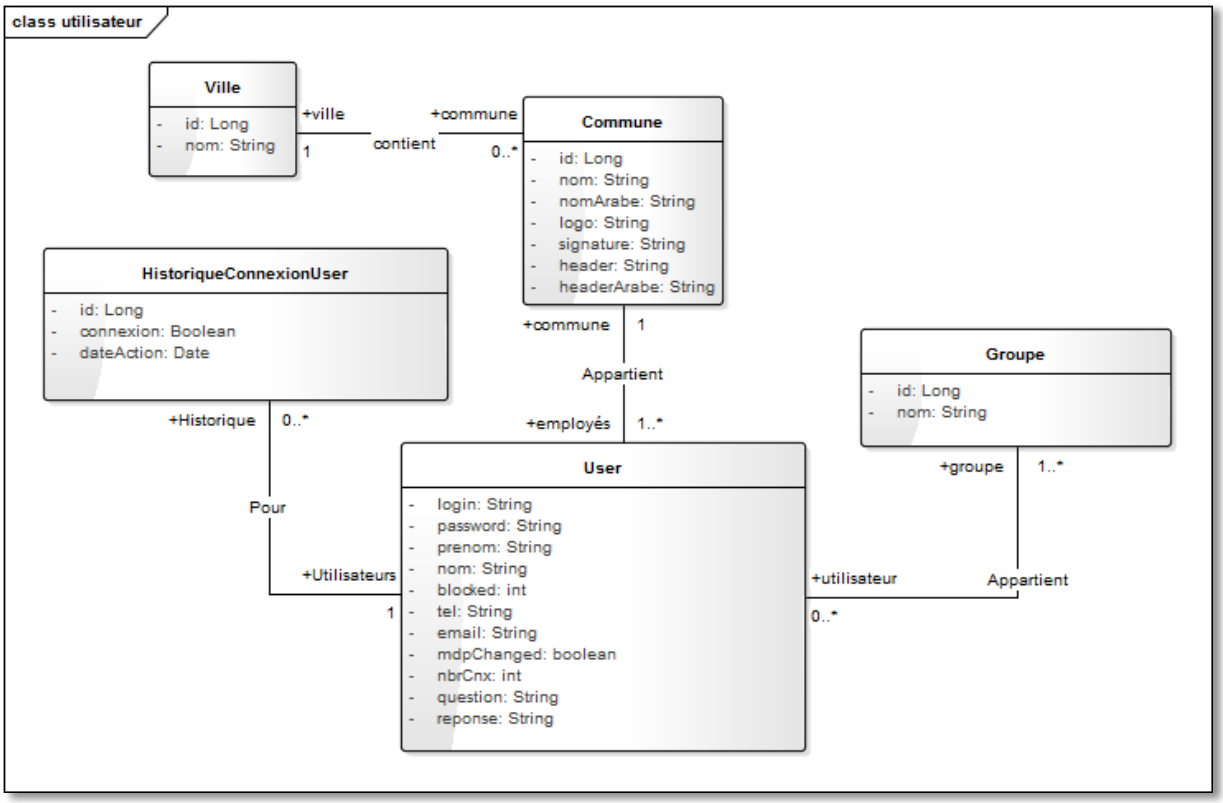


Figure 54 : Diagramme de classes « Gestion utilisateurs » avec les attributs

✓ Description :

Utilisateur :

Cette classe contient les informations d'authentification (login, mot de passe) des utilisateurs. Un utilisateur appartient à une commune et il a un historique de connexion/déconnexion. Chaque utilisateur a un rôle représenté par la classe « **Groupe** ».

Commune :

Cette classe représente la commune dans laquelle l'application sera opérationnelle, elle est caractérisée par le nom, l'index, la ville (classe Ville), ainsi que le logo, l'entête (header), qui seront présent au niveau des bordereaux ainsi que la signature du président pour les lettres de notification et les avis d'imposition.

➢ Package Adresse

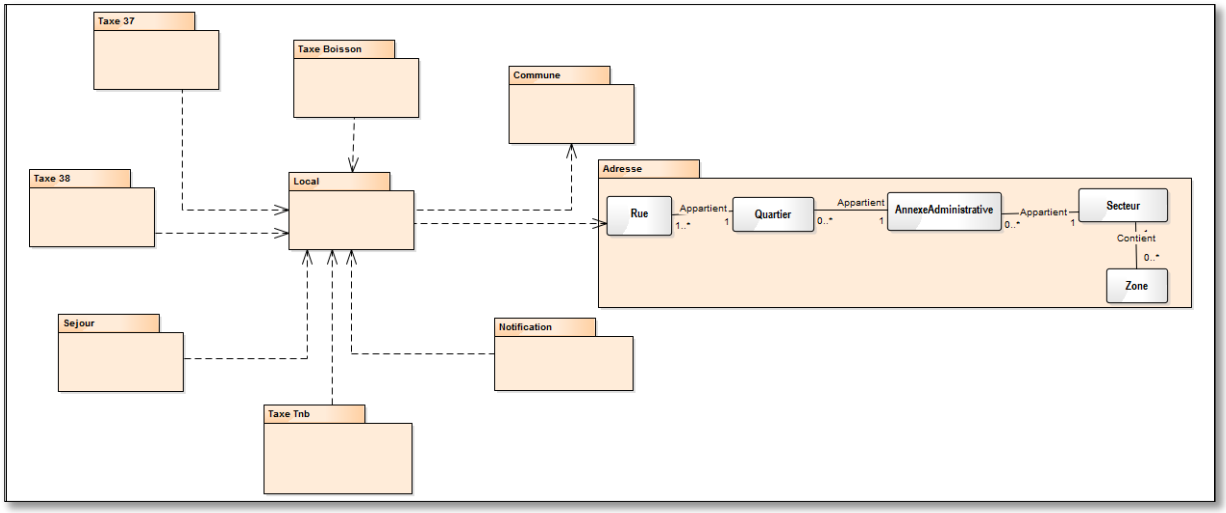


Figure 55 : Diagramme "Package Adresse"

➢ **Diagramme de classes « Package Adresse »**

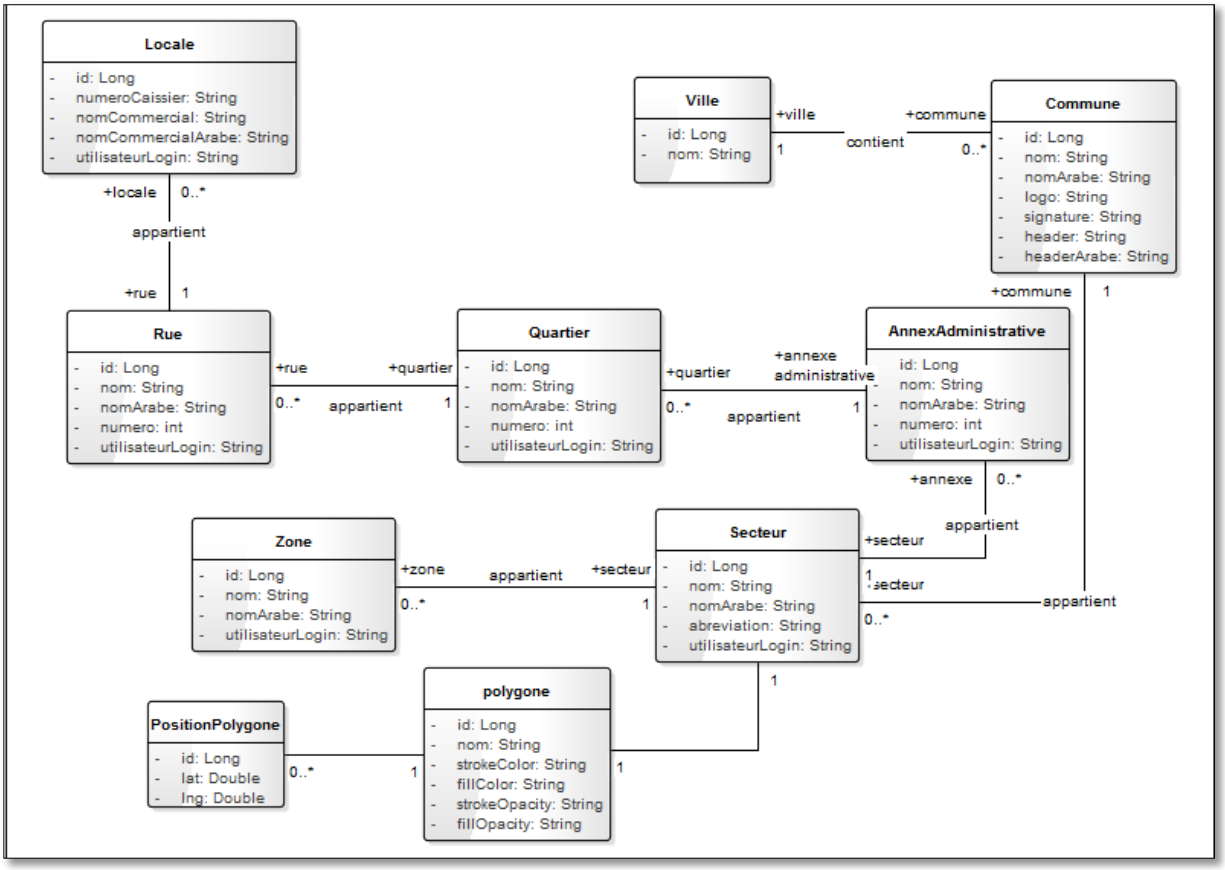


Figure 56 : Diagramme de classes « Gestion adresses » avec les attributs

✓ **Description :**

Le découpage de l'adresse des locaux est représenté par les classes (Secteur, Annexe Administrative, Quartier, Rue et Zone) selon le schéma ci-dessous.

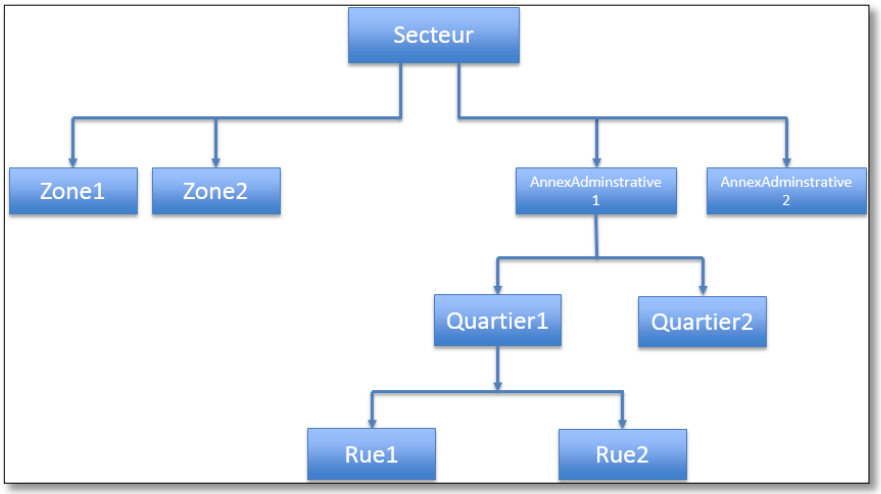


Figure 57 : Découpage de l'adresse

➢ **Diagramme global :**

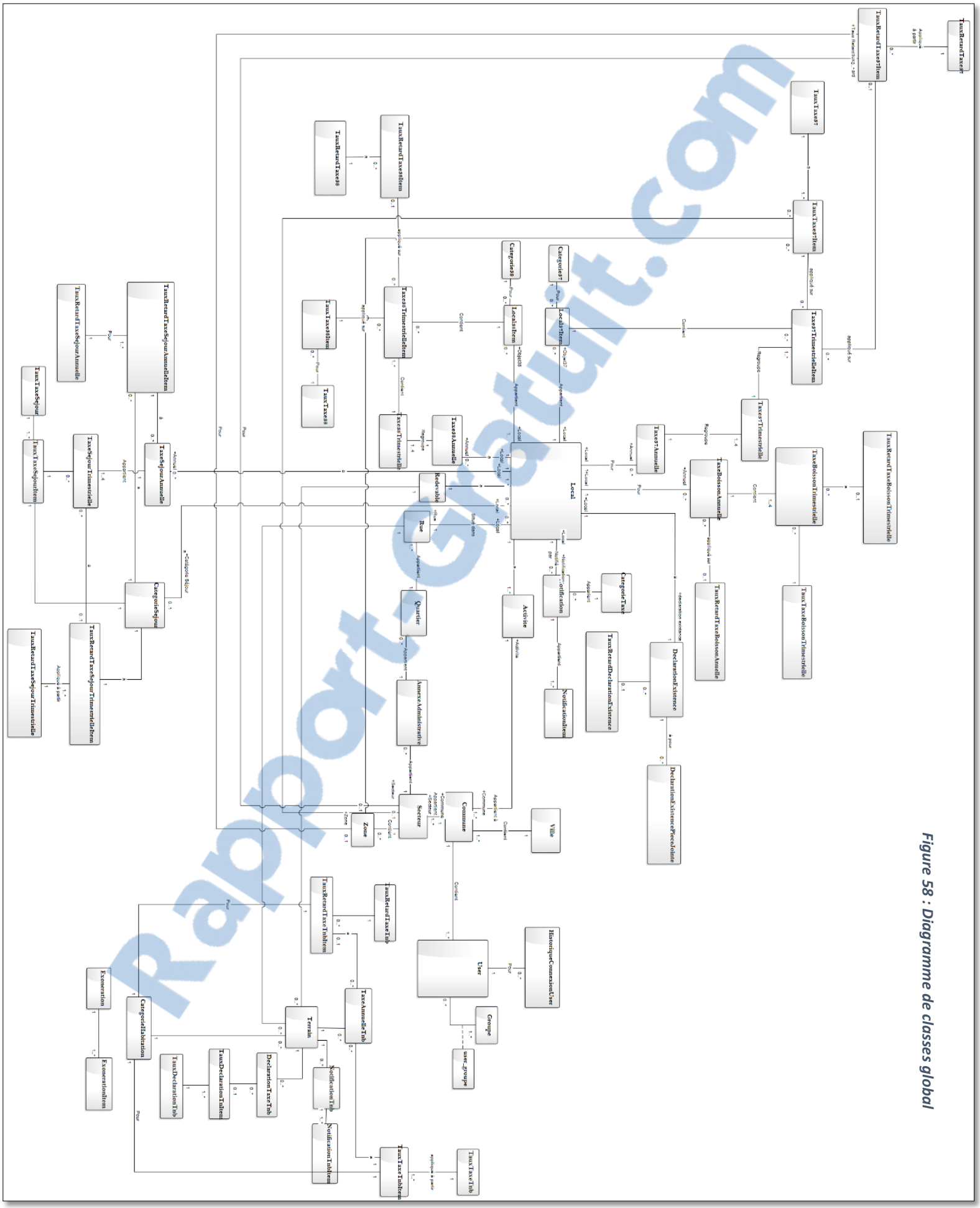


Figure 58 : Diagramme de classes global

Rapport-Gratuit.com

CHAPITRE 4 :

Outils techniques

- JEE
- JSF
- PrimeFaces 6.0
- EJB3
- JPA2

Ce chapitre traite l'architecture technique adoptée, une présentation des différents Framework utilisés.

6. Outils Techniques

1. Plateforme JEE

a. Pourquoi la technologie JEE ?

Les entreprises recherchent de plus en plus, des solutions qui permettent la diminution des couts et la réduction du temps de réponse, tout en respectant les standards de qualité logicielle.

Typiquement les applications qui répondent à ces besoins doivent combiner entre les systèmes d'informations existantes et les nouvelles fonctions métiers qui apportent des services à un large intervalle des utilisateurs. Ces services doivent être :

- ✓ Fortement disponibles, pour répondre aux besoins actuels de l'environnement métier.
- ✓ Sécurisées, pour protéger les utilisateurs et l'intégrité des données d'entreprise.
- ✓ Fiables et extensible, pour s'assurer que les transactions sont exécutées d'Une manière exacte et prompte.

Pour des raisons diverses, ces services sont conçus à base d'applications distribuées reposant sur plusieurs niveaux à savoir la couche client, la couche client, la couche données et une ou plusieurs niveaux à savoir la couche client, la couche donnée et une ou plusieurs couches intermédiaires.

b. Architecture multi-tiers

Le concept de l'architecture multi tiers vient pour remédier aux problèmes de la maintenance, de la réutilisation et de la montée en charge, ce qu'on appelle « scalabilité », d'une application. Les applications basées sur ce type d'architecture sont divisées en plusieurs niveaux logiques, chacun d'entre eux comportant un ensemble d'interfaces bien définies, à savoir la couche client, la couche données et une ou plusieurs couches intermédiaires.

Le niveau intermédiaire (logique applicative) est simplement constitué du code appelé par l'utilisateur (via le niveau présentation) pour extraire les données utiles. Le niveau présentation reçoit ensuite les données et les met en pages, en vue de leur affichage. Cette séparation de la logique applicative de l'interface utilisateur accentue considérablement la souplesse de conception d'une application. En effet, il est désormais possible de créer et de diffuser plusieurs interfaces utilisateurs sans modifier la logique applicative, à condition toutefois, que cette dernière soit dotée d'une interface strictement définie avec le niveau intermédiaire.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Enfin, le troisième niveau contient les données nécessaires à l'application. Il peut s'agir de toute source d'information présentée sous n'importe quelle forme : base de données relationnelle, ensemble de documents XML, ou encore service de répertoires tel qu'un serveur LDAP.

c. L'architecture JEE

C'est une architecture qui définit un modèle de programmation multi tires basé sur des composants. Avec ce modèle, des applications légères peuvent interagir facilement avec le système d'information et les composants implantant la logique d'entreprise sur des serveurs d'applications.

L'architecture JEE consiste en trois parties :

1- **Components** : les composants qui prennent en charge la présentation et la logique métier ;
2- **Containers** : fournit un contexte aux composants ;
3- **Connectons** : fournit l'accès aux sources de données de l'entreprise.

Cet environnement supporte trois types de composants d'applications, dont les définitions seront présentées par la suite, Applets, Servlets (y compris JSP) et Enterprise JavaBean.

La figure suivante illustre les trois parties constituant l'architecture JEE. Le noyau de serveur « JEE Sever Core » est un environnement de serveur d'application qui fournit les services de gestion de ressources et de transactions.

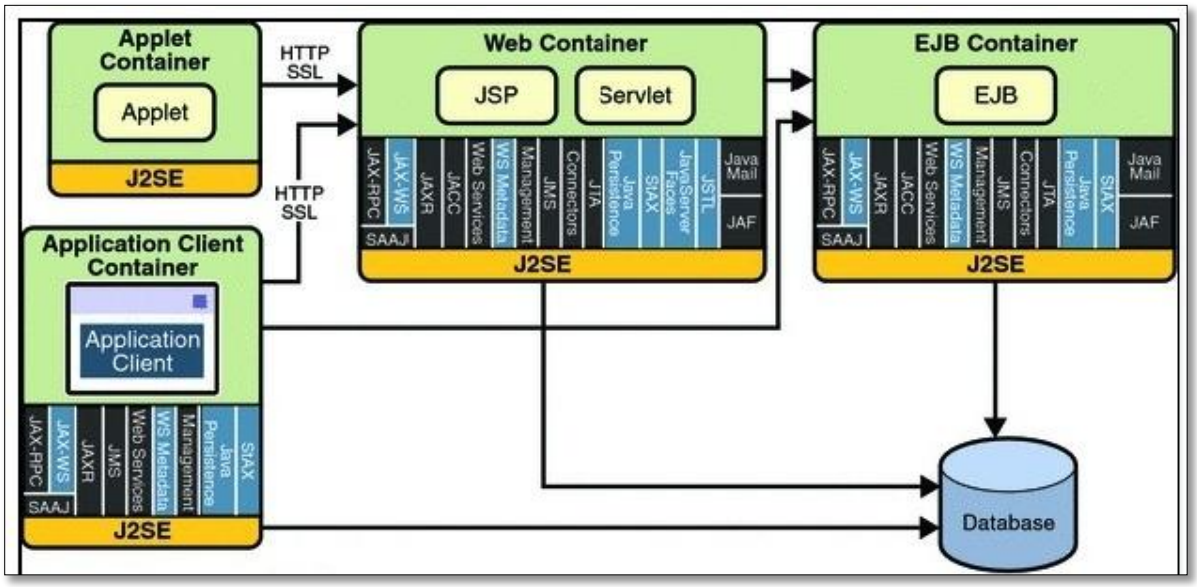


Figure 59 : L'architecture globale de JEE

2. Définition de Framework

Un Framework est un ensemble d'outils et de composants qui aident au développement d'application, pour un contexte donné.

Cela inclut :

- ✓ Un ensemble de Design Patterns,
- ✓ Un ensemble d'interfaces et implémentations,
- ✓ Des outils associés.

3. Le Framework JSF

JSF est une spécification permettant de développer des interfaces graphiques améliorées pour les applications Web basées sur la plate-forme JEE. Les JSF utilisent un modèle Orienté évènement. Elle propose dans son implémentation des composants graphiques réutilisables capables de gérer les états et les événements. Les JSF proposant aussi une libraire de balises à insérer dans les pages JSP.

a. Principes

Le Framework Java Server Faces (JSF) est basé sur le modèle MVC2, il fournit des APIs standard et des Taglibs utiles aux développeurs Java qui développement des GUI web. JSF définit un lot de Taglibs JSP qui génère des éléments HTML pouvant être « bindés » à des propriétés de Java Beans.

JSF est un standard JEE qui permet :

- ✓ Une séparation entre la couche présentation et les autres couches d'une application web.
- ✓ Une mise en place d'un mapping HTML/OBJET.
- ✓ La réutilisation de composants graphiques.
- ✓ Une gestion de l'état de l'interface entre les différentes requêtes.
- ✓ Une liaison entre les actions cotées « Client » et les actions des objets Java coté, « Serveur »
- ✓ Création de composants personnalisés.

b. Cycle de vie

Dans le graphe ci-dessous est illustré le cycle de vie d'une requête JSF :

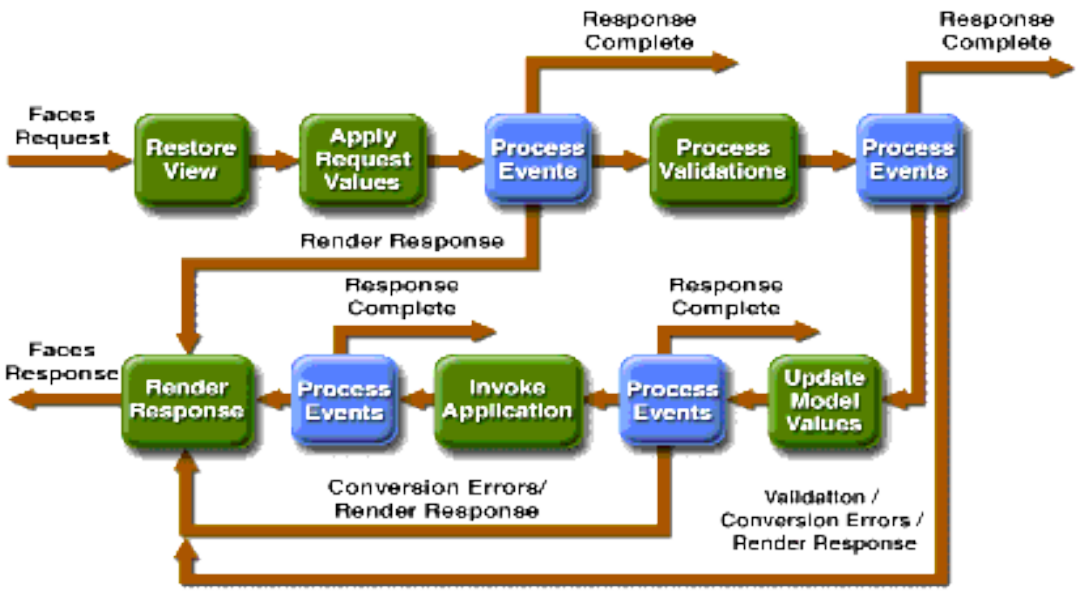


Figure 60 : Cycle de vie d'une requête JSF

Comme le montre le schéma, le cycle de vie d'une requête JSF est composé de 6 phases :

- ✓ **Restore View** : Restaure l'arbre des composants JSF associé à cette vue ou crée un nouvel arbre de composants.
- ✓ **ApplyRequest Values** : Mappe le contenu des attributs de la requête dans l'arbre des composants JSF,
- ✓ **Proccess validation** : Conversion des attributs si nécessaire et application des validations, une erreur de validation amène à la phase Render Response.
- ✓ **Update Model values** : Associe les valeurs de l'arbre des components aux attributs des backingsbeans.
- ✓ **Invoke application** : Diffuse certains évènements vers des listeners éventuels.
- ✓ **RenderResponse** : Sauvegarde l'état de l'arbre des composants et génère la réponse.

c. PrimeFaces

Prime Faces est une bibliothèque open source de composants JSF.

- **Côté serveur :** il est basé sur l'API standard de JSF à partir de la version 2.
- **Côté client :** les scripts, de Primefaces sont basés sur la librairie la plus populaire de javaScript **jQuery**.

En générale Primefaces vise à garder le traitement propre, rapide et léger.

d. Framework Facelets

Facelets est un Framework permettant de faire des vues de JSF basées sur le modèle html, il permet de coder des vues avec de simples balises XML, bien plus familières que le codage de jsp.

Il offre également un support à JSF pour l'utilisation de Template, ce qui réduit la quantité de code produite.

En fait, Facelets est basé sur une représentation de la page sous forme d'Ul Components (composants JSF) et traite les tags JSF (les Facelets supportent tous les composants JSF) et HTML comme étant des membres à part entière de l'arbre. Il permet également la création de composants réutilisable sans une seule ligne de code java.

Facelet est basé sur trois idées principales :

- ✓ L'intégration JSF : Facelet fournit une technologie performante JSF-centrique qui utilise le standard ViewHandler. Facelet intègre tous les composants JSF et fonctionne sans le cycle de vie de JSF,
- ✓ La composition : Facelet compile un arbre de composant JSF à partir de plusieurs vues. Il traite les balises dans le fichier comme des instructions de construction d'un arbre de composants JSF. Il peut compiler un ou plusieurs fichiers de vue en un seul arbre de composant,
- ✓ Les templates : Facelet supporte la définition de template de vues réutilisable dans plusieurs pages. Quand Facelet construit l'arbre de composant il parcourt le template et remplace dynamiquement son contenu ou utilise les valeurs par défaut dans le template. Le templating et la facilité de développement sont les points forts des facelets.

4. Les entreprises JavaBean

Les Entreprise JavaBeans (EJB) sont des composants Java qui peuvent être combinés pour créer des applications distribuées et modulaires. Le standard EJB décrit à la fois une spécification et un ensemble d'interfaces que doivent implanter les composants. Un des principes de conception d'une telle architecture est la séparation des concepts non-fonctionnels de niveau système (comme le multi-threading, la persistance, les transactions, la sécurité, etc) de la logique de l'application des programmes spécifiques nommés conteneurs EJB fournissent une plateforme d'accueil pour les EJB ainsi que les services de niveau système. Ces conteneurs EJB sont eux-mêmes inclus dans des serveurs d'applications qui peuvent accueillir tout un panel de conteneurs différents. De bien des manières, les EJBs ressemblent aux objets RMI, sur lesquels d'ailleurs ils reposent, en exposant leurs fonctionnalités à des clients distants au moyen d'interfaces.

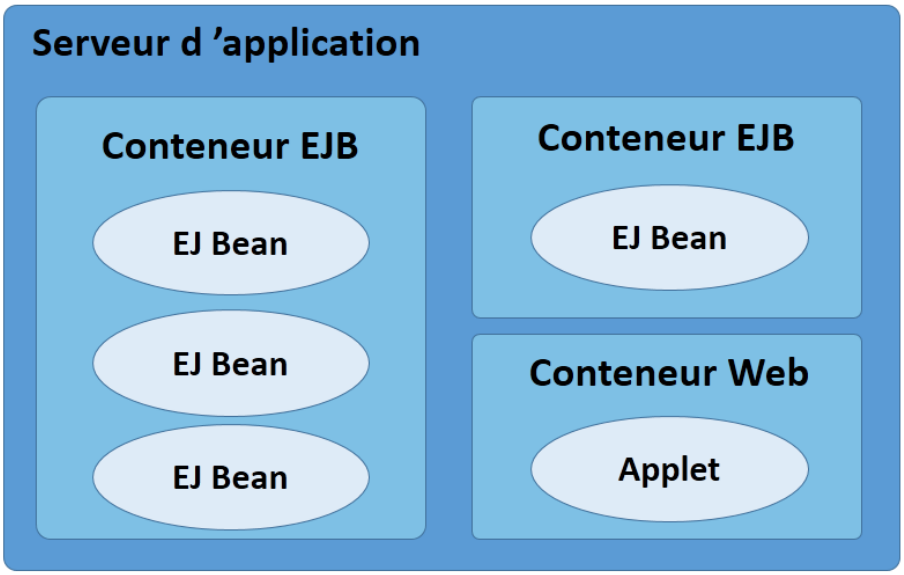


Figure 61 : Architecture interne d'un serveur d'application

Les EJBs sont aussi architecturés dans le but de promouvoir leur réutilisation au sein de diverses applications et fournir de la flexibilité dans leurs déploiements et leurs administrations selon les besoins des entreprises. Pour cela, il existe une claire séparation entre les divers acteurs entrant en jeu dans l'élaboration d'une application. Ces acteurs sont organisés en rôle

- ✓ les fournisseurs de Beans sont les concepteurs des EJBs.
- ✓ les assembleurs d'applications assemblent les EJBs pour créer une application.
- ✓ les déployeurs d'applications déploient les applications assemblées au sein d'un environnement de production.
- ✓ les administrateurs systèmes s'occupent de l'environnement de production, ils veillent à fournir à l'application toutes les ressources dont elle a besoin durant son exécution.

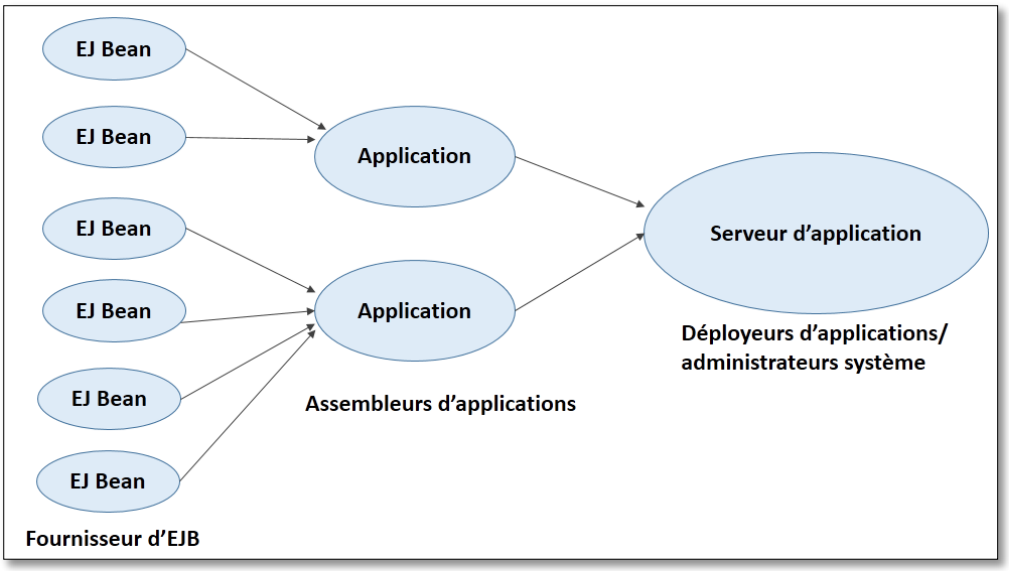


Figure 62 : Communication entre le serveur d'application et les fournisseurs EJB

Au sein d'une architecture Java EE, les EJB sont utilisés pour créer les services. Cette technologie ne s'arrête cependant pas à cette couche, mais permet aussi de créer l'abstraction de l'accès aux données. Ce sont les entités qui remplissent cette fonction.

Tout comme les beans sessions, entre le client et la logique métier, les beans entités forment la passerelle entre la logique applicative et les sources de données. Ils offrent une abstraction quasi complète du stockage des données, permettant à l'application de rendre persistantes ou de charger des données de manière totalement transparente.

5. Spécification JPA

JPA 1.0 a été créée avec Java EE 5 pour résoudre le problème de la persistance des données en reliant les modèles objets et relationnels. Avec Java EE 6, JPA 2.0 conserve la simplicité et la robustesse de la version précédente tout en lui ajoutant de nouvelles fonctionnalités. Grâce à cette API, vous pouvez accéder à des données relationnelles et les manipuler à partir des EJB (Entreprise Java Beans), des composants web ou même des applications Java SE.

JPA est une couche d'abstraction au-dessus de JDBC, qui fournit une indépendance vis-à-vis de SQL. Toutes les classes et annotations de cette API se trouvent dans le paquetage javax.persistence.

Ses composants principaux sont les suivants

- ✓ ORM, qui est le mécanisme permettant de faire correspondre les objets à des données stockées dans une base de données relationnelle.

- ✓ Une API gestionnaire d'entités permettant d'effectuer des opérations sur la base de données, notamment les opérations CRUD (Create, Read, Update, Delete). Grace à elle, il n'est plus nécessaire d'utiliser directement JDBC.

- ✓ JPQL ava Persistence Query Language) qui permet de récupérer des données à l'aide d'un langage de requêtes orienté objet.

- ✓ Des mécanismes de transaction et de verrouillage lorsque nous accédons de façon concurrente aux données, fournis par JTA (Java Transaction API). Les transactions locales à la ressource (non JTA) sont également reconnues par JPA.

- ✓ Des fonctions de rappel et des écouteurs permettant d'ajouter la logique métier au cycle de vie d'un objet persistant.

Nouveauté de JPA 2.0

Cette seconde version ajoute de nouvelles API, étend JPQL et intègre de nouvelles fonctionnalités :

- ✓ Les collections de types simples (String, Integer, etc.) et d'objet intégrables (embeddable) peuvent désormais être associées à des tables distinctes alors qu'auparavant nous ne pouvions associer que des collections d'entités.

- ✓ Les clés et les valeurs des associations peuvent désormais être de n'importe quel type de base, des entités ou des objets intégrables.

- ✓ L'annotation @orderColumn permet maintenant d'avoir un tri persistant.

- ✓ La suppression des orphelins permet de supprimer les objets fils d'une relation lorsque l'objet parent est supprimé.

- ✓ Le verrouillage pessimiste a été ajouté au verrouillage optimiste, qui existait déjà.

- ✓ Une toute nouvelle API de définition de requêtes a été ajoutée afin de pouvoir construire des requêtes selon une approche orientée objet.

- ✓ La syntaxe JPQL a été enrichie (elle autorise désormais les expressions case, par exemple).

- ✓ Les objets intégrables peuvent maintenant être embarquées dans d'autres objets intégrables et avoir des relations avec les entités.

- ✓ La notation pointée a été étendue afin de pouvoir gérer les objets intégrables avec des relations ainsi que les objets intégrables d'objets intégrables.

- ✓ Le support d'une nouvelle API de mise en cache a été ajouté.

L'implémentation de référence de JPA est EclipseLink, anciennement TopLink d'Oracle. Ce Framework est à la fois souple et puissant. Il supporte la persistance XML au travers de JAXB (Java XML Binding). Il est également désigné sous les termes de fournisseur de persistance ou, simplement de fournisseur. Il offre un ORM, un OXM (Object XML Mapping) et la persistance des objets sur EIS (Enterprise Information Systems) à l'aide de JCA (Java EE Connector Architecture).

6. Environnement technique

a. Git

C'est un logiciel de gestion de version décentralisé, ce qui signifie qu'il gère les modifications d'un projet sans en écraser toutes les parties.

b. Netbeans

NetBeans IDE est un environnement de développement intégré (EDI) de premier ordre pour Windows, Mac, Linux et Solaris. Le projet NetBeans consiste en un EDI Open Source et en une plate-forme d'application permettant aux développeurs de créer rapidement des applications Web, d'entreprise, de bureau et mobiles à l'aide de la plate-forme Java, ainsi que de JavaFX, PHP, JavaScript et Ajax, Ruby et Ruby on Rails, Groovy et Grails et C/C++. NetBeans est disponible en 23 langues dont la plupart de ces traductions a été réalisée par la communauté des utilisateurs

c. Xampp

XAMPP est un ensemble de logiciels permettant de mettre en place facilement un serveur Web local, un serveur FTP et un serveur de messagerie électronique. Il s'agit d'une distribution de logiciels libres (X (cross) Apache MariaDB Perl PHP) offrant une bonne souplesse d'utilisation, réputée pour son installation simple et rapide.

d. 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 Server4 (anciennement Sun Java System Application Server5 de Sun Microsystems). Sa partie Toplink persistence6 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.

La distribution dite Open Source Edition est placée sous double licence CDDL et GPLv2.

Rapport-Gratuit.com

GlassFish est certifié Java EE 5 (EJB3 + JPA + JSF + JAX-WS 2.x + ...) et Java EE 6 (EJB 3.1, CDI, JSF 2.0, JAX-RS 1.1, ...)

e. Enterprise Architect

Enterprise Architect est un logiciel de modélisation et de conception UML, édité par la société australienne Sparx Systems. Couvrant, par ses fonctionnalités, l'ensemble des étapes du cycle de conception d'application, il est l'un des logiciels de conception et de modélisation les plus reconnus.

f. Jasper Reports

JasperReports est outil 100% Open Source de génération d'états/ rapports développé en Java, pour des applications Java. Il se présente sous la forme de librairies à intégrer aux applications. Ces librairies permettent la visualisation ou l'export de données vers de multiples formats.

g. iText

iText est une interface de programmation partiellement à code source ouvert servant à créer et manipuler des documents PDF.

h. Gantt Project

GanttProject est un outil permettant de gérer les projets sur le modèle des diagrammes de Gantt.

i. Trello

Trello est un outil de liste de tâche bien pensé permettant de gérer ses projets efficacement.

j. Balsamiq mockups

Balsamiq est un outil permettant de créer facilement des prototypes d'IHM électronique.

Conclusion

Nous avons cerné dans ce chapitre les points clés de technologies et le Framework sélectionnés. Dans le chapitre suivant nous entamerons la phase d'implémentations de l'application.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

CHAPITRE 5 :

Mise en œuvre

- Les difficultés rencontrées
- Les captures d'écran les plus pertinent

Ce chapitre décrit les difficultés rencontrées durant la phase de la réalisation ainsi que les écrans les plus pertinents de l'application.

7. Mise en œuvre

1. Les difficultés rencontrés

a. Les difficultés au niveau de la conception

Parmi les difficultés que nous avons rencontrées est celle de l'analyse conceptuelle, durant cette phase, il fallait tout d'abord comprendre le besoin de la commune pour établir une conception optimale, mais la difficulté s'est manifestée lors de la phase de la réalisation suite à la mauvaise l'expression du qui nous pousse parfois à effectuer des changements au niveau du diagramme de classes.

Nous avons pensé à ce que la gestion des locaux soit centralisé dans la classe "Local", mais après avoir eu plus de détail sur la taxe TNB, nous avons constaté qu'on ne peut pas exploiter cette classe pour cette taxe, puisque l'objet de la taxe est un "Terrain" dont les informations diffèrent d'un local, c'est pour cela que nous avons rajouté la classe « Terrain » pour isoler le traitement des autres taxes.

Concernant les notifications, notre système de notifications génère des notifications pour la taxe Boisson, séjour et TNB. La taxe TNB subit un traitement particulier par rapport aux autres. En effet, la notification de la taxe boisson et séjour consiste à notifier les locaux qui n'ont pas déclaré une ou plusieurs années, de plus, le calcul des montants est sous forme d'une estimation sur la base du dernier paiement ou par voisinage (la moyenne des derniers paiements des locaux similaire). Or la notification de la taxe TNB consiste à calculer le montant brut et de notifier le redevable dans les années non payé, c'est pour cela, nous avons aussi isolé le traitement de cette dernière dans les classes « notificationTnb » et « notificationTnbItem ».

b. Les difficultés au niveau technique

Les difficultés techniques majeures que nous avons rencontrées durant ce projet résident essentiellement dans la nouveauté des technologies Open Source à savoir EJB3/JSF2

- **Problèmes encodage avec JSF :**

Le problème que nous avons rencontré est au niveau des enregistrements en arabe. Si nous insérons les lettres en arabe nous obtenons des symboles bizarre dans la base de données tel que « ???.. ». Sachant que lorsque nous insérons directement dans la base de données cela fonction bien. Mais à partir d'une page web nous tombons sur le même problème. L'encodage de la page étant UTF-8.

La solution que nous avons mis en place c'est l'encodage UTF-8 de chaque attribut lors de sa modification (SET) et le décodage lors de sa lecture (GET).

- **Jasper Report arabe :**

Nous avons utilisé Jasper Report 6.3.1 pour créer nos rapports (fichier JRXML et .Jasper).

Le problème que nous avons rencontré est quand on exporte le rapport via jasper Report en PDF, la page apparait vierge.

La solution que nous avons mis en place est l'installation du font « Arial », puisque il support la langue arabe.

2. Ecrans de l'application

En ce qui concerne les écrans de l'application nous nous restreindrons à présenter les écrans les plus pertinents.

a. Page de connexion à e-Taxe

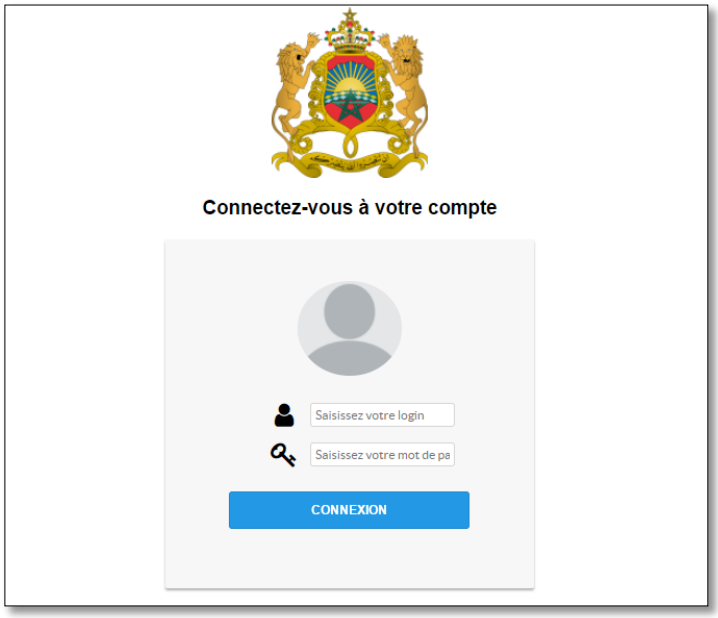


Figure 63 : IHM d'authentification

Cet écran vous propose d'accéder à l'application selon votre profile : Utilisateur Simple (Régisseur communal), Responsable du service d'assiette ou Administrateur. Après avoir fourni le nom d'utilisateur et le mot de passe, l'utilisateur accède au menu principal.

b. Vue d'un régisseur communal

Une fois succès de connexion, l'utilisateur est redirigé vers la page d'accueil, cette page offre à l'utilisateur un menu générale pour l'application.

Figure 66 : Menu principal (vue d'un régisseur "Taxe boisson")

Figure 65 : Menu principal (vue d'un régisseur "Taxe séjour")

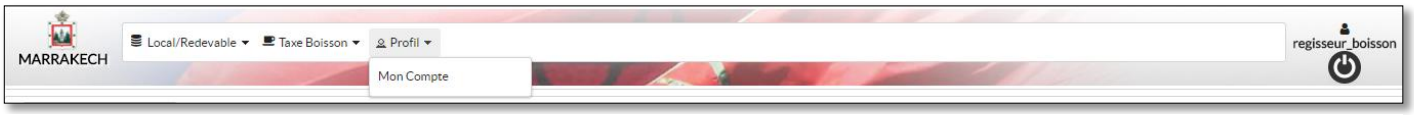


Figure 64 : Menu (vue régisseur "Taxe boisson ")

Figure 68 : Menu principal (vue d'un régisseur "Taxe boisson et séjour")

Figure 67 : Menu (vue régisseur "Taxe boisson et séjour")

c. Vue d'un responsable

Figure 70 : Menu principal de l'application (Vue responsable)

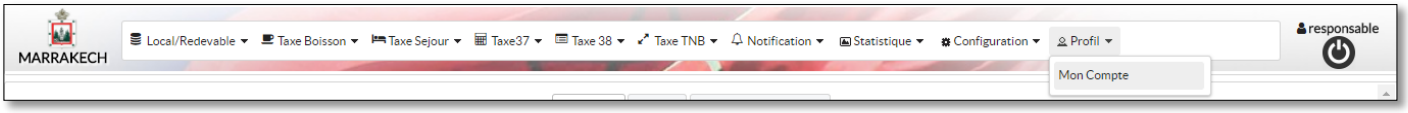


Figure 69 : Menu « vue responsable »

d. Vue d'un administrateur

Une fois succès de connexion, l'utilisateur est redirigé vers la page d'accueil, cette page offre à l'utilisateur un menu général pour l'application.

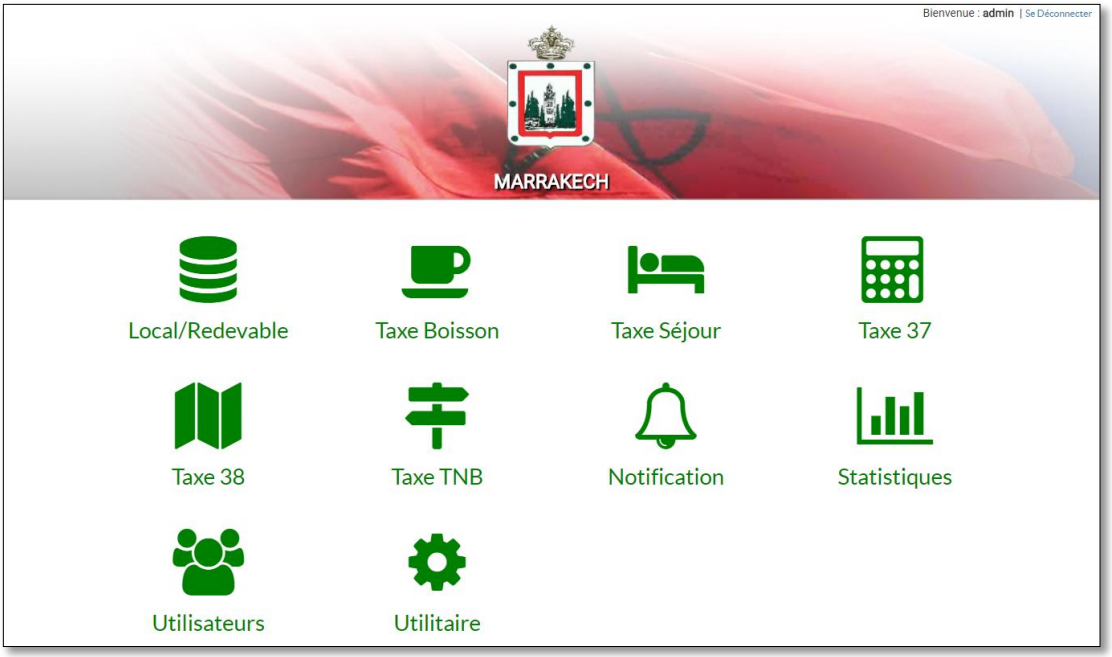


Figure 71 : Menu principal de l'application (Vue administrateur)

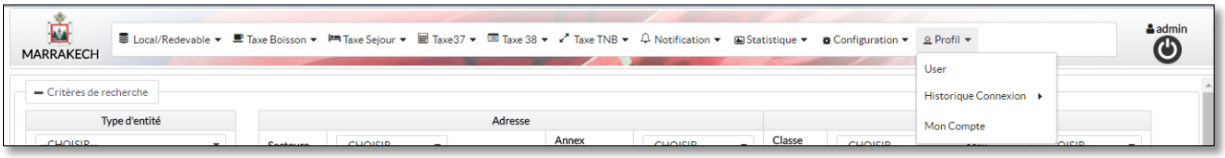


Figure 72 : Menu (Vue administrateur)

❖ Local/Redevable

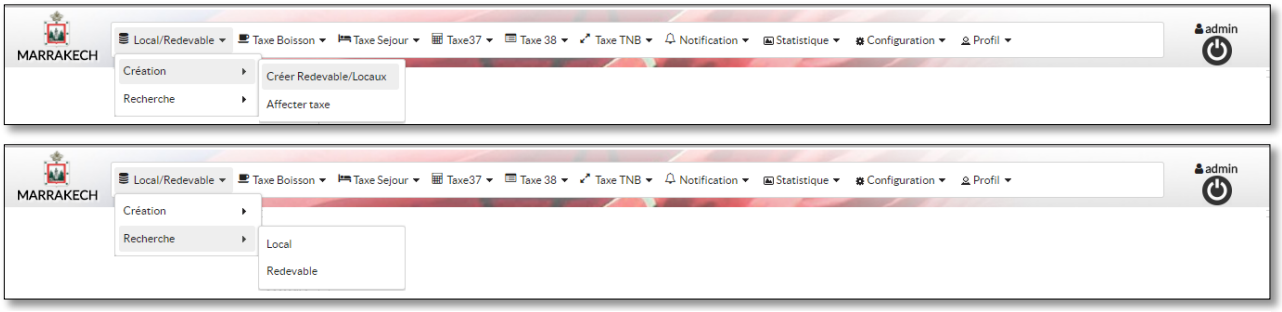


Figure 73 : Menu « Local/Redevable »

➢ IHM création d'un redevable :

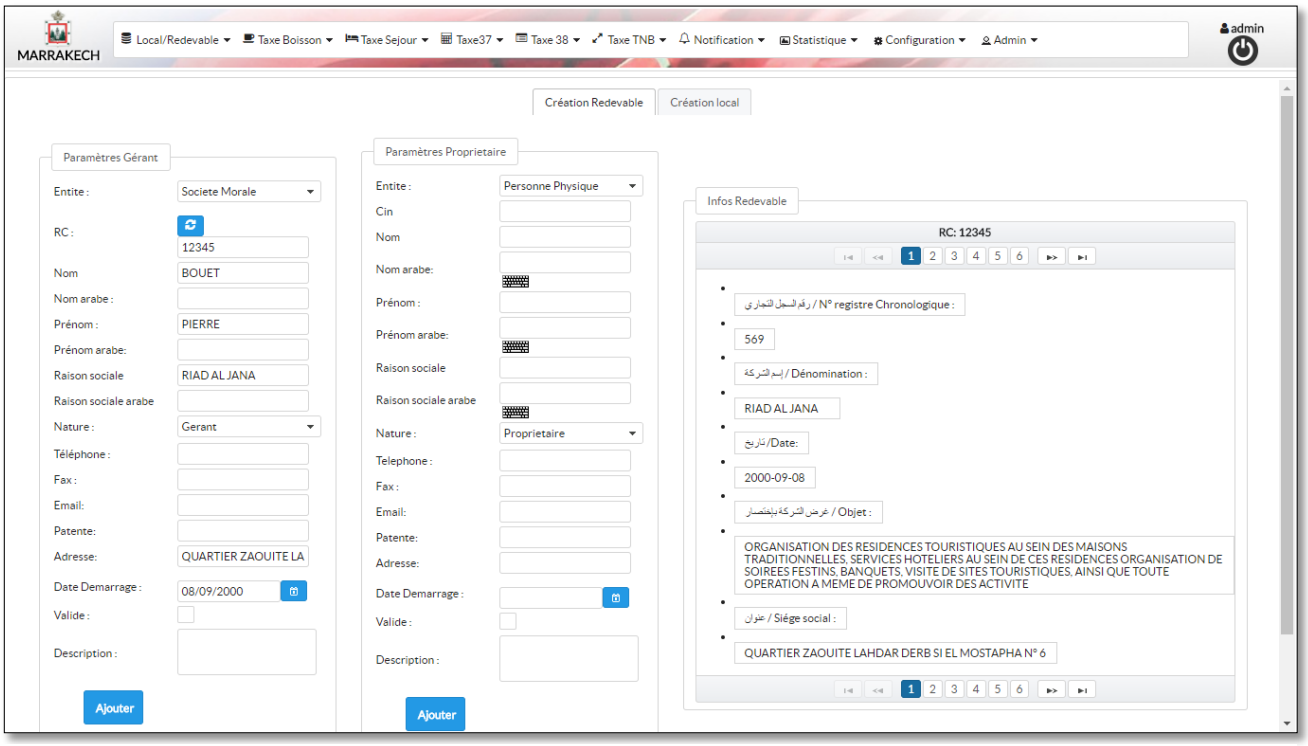


Figure 74 : IHM création d'un redevable

Une fois l'utilisateur complète la saisie des informations du redevable (Gérant et/ou propriétaire), il sera automatiquement redirigé vers l'IHM de la création d'un local pour saisir les informations du local.

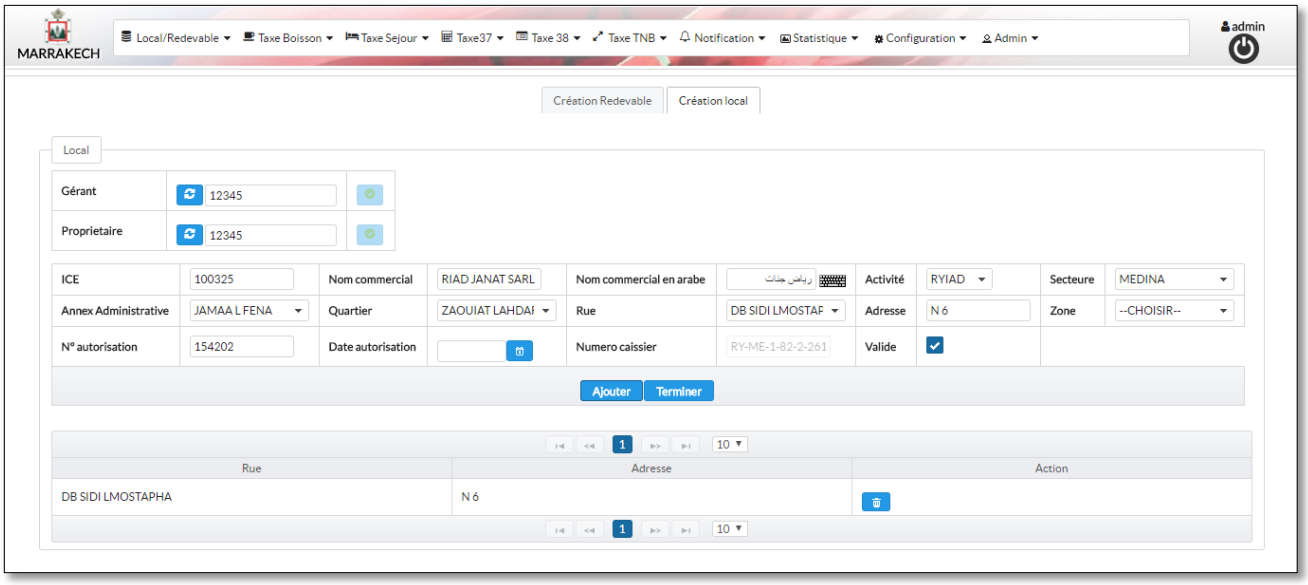


Figure 75 : IHM création d'un local

Après avoir compléter la saisie des informations du local, l'utilisateur est redirigé automatiquement vers l'IHM d'affectation des taxes, dans cet exemple nous avons affecté la taxe boisson et séjour au local concerné, l'utilisateur peut à tout moment accéder à cette interface pour affecter une taxe à un local bien précis.

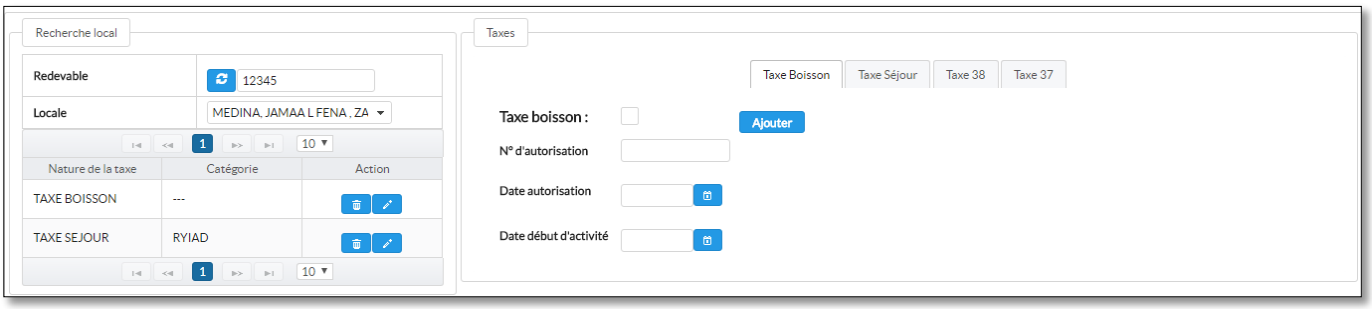


Figure 76 : IHM affectation des taxes au local

- IHM pour rechercher et consulter un redevable, voir ses détails, le modifier ou le supprimer

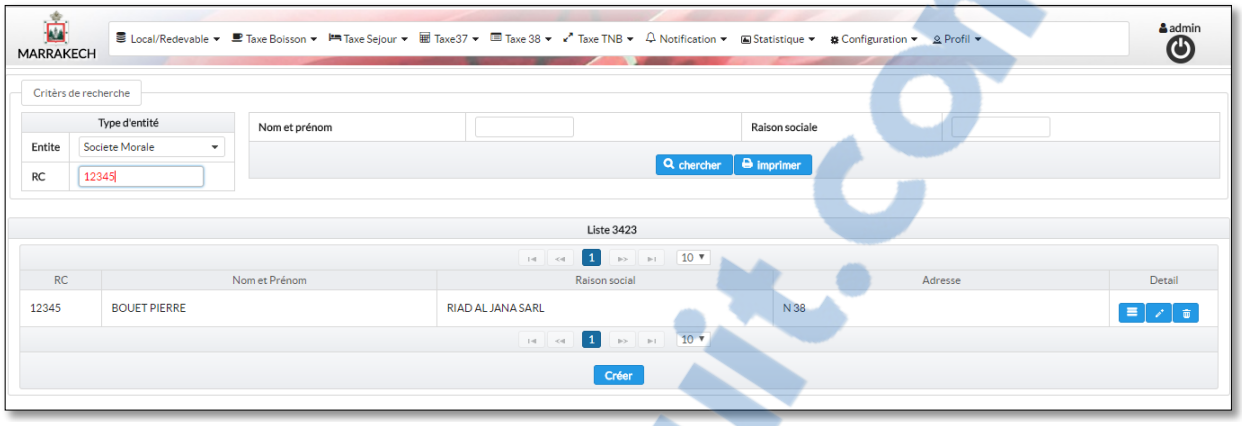


Figure 77: IHM recherche d'un redevable

Une recherche du local par RC du redevable que nous avons inséré

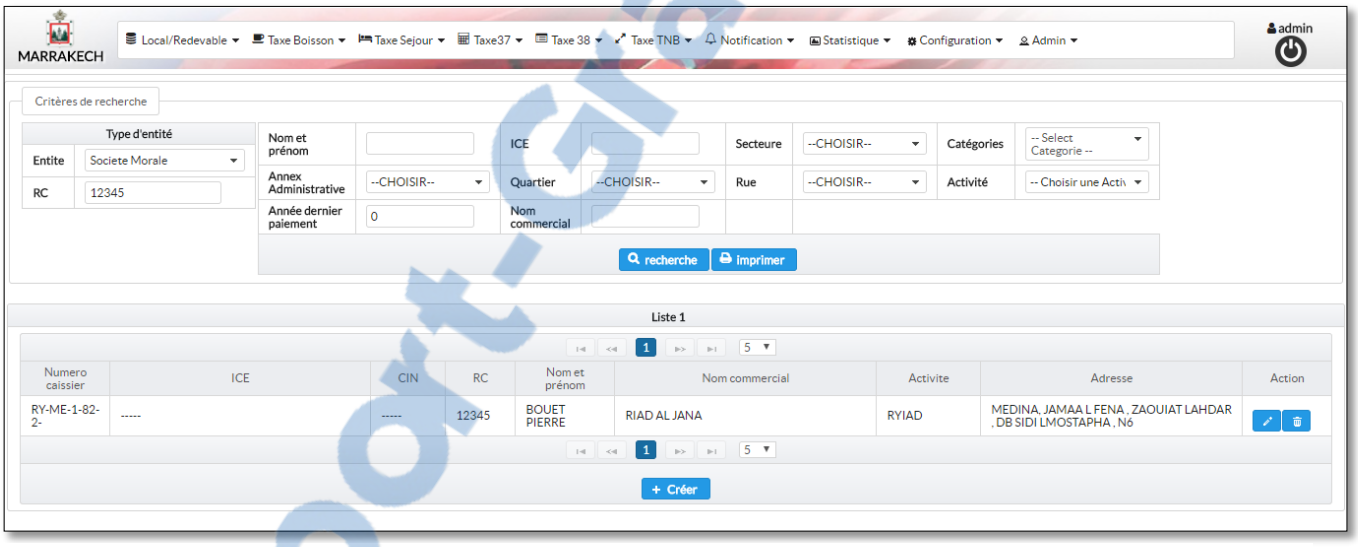


Figure 78 : IHM recherche d'un local

Rapport-Gratuit.com

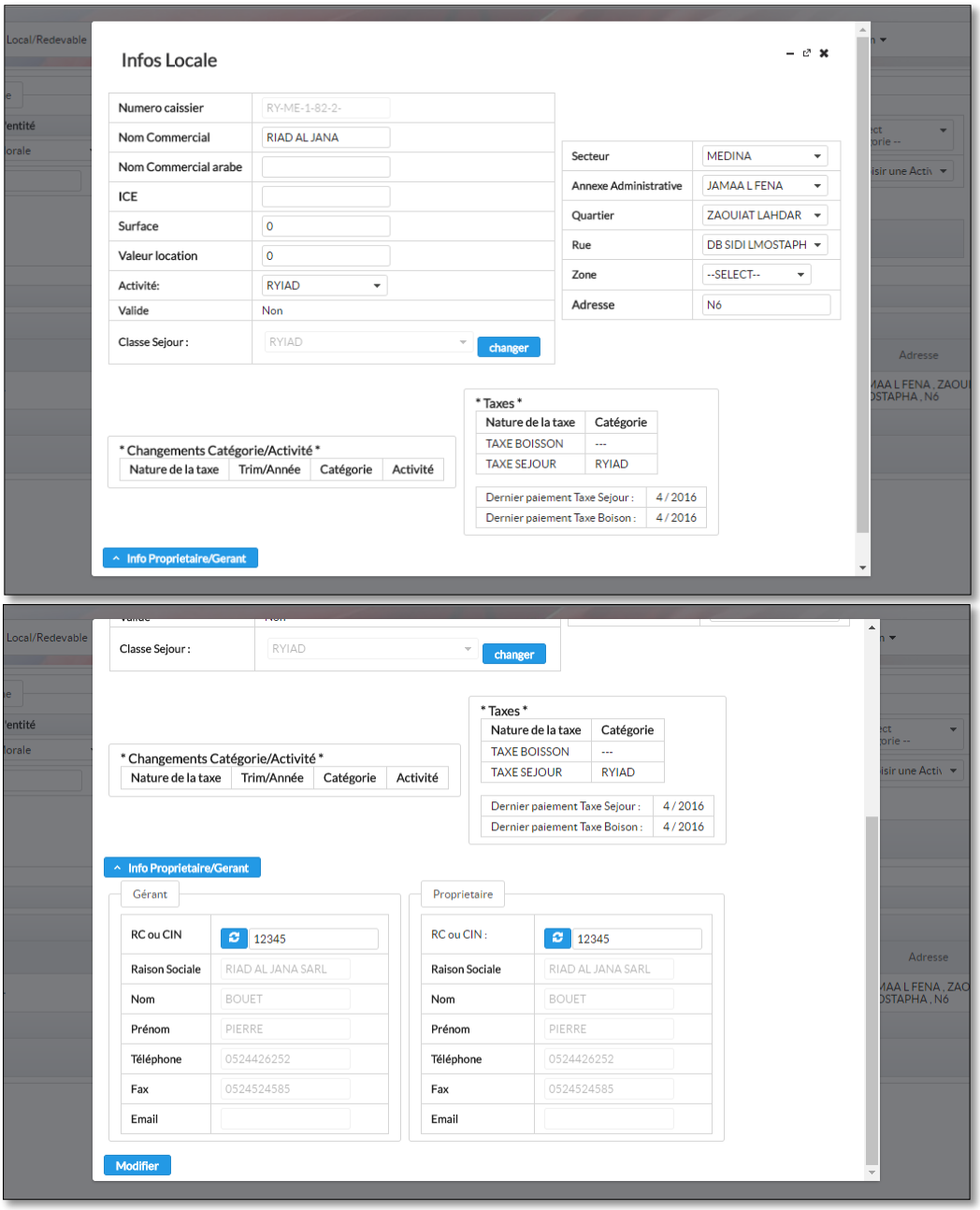


Figure 79 : IHM détail d'un local

❖ Taxe Boisson :

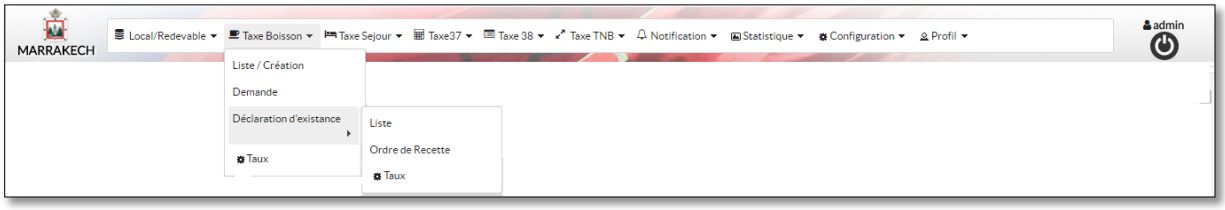


Figure 80 : Menu « Taxe boisson »

➢ IHM paiement taxe boisson (vue administrateur) :

La particularité de la vue utilisateur par rapport à la vue régisseur ou responsable, c'est qu'un administrateur peut effectuer une recherche en fonction des utilisateurs, l'état du paiement (valide/non valide) et la date du système contrairement à un régisseur ou un responsable« Figure 68 ».

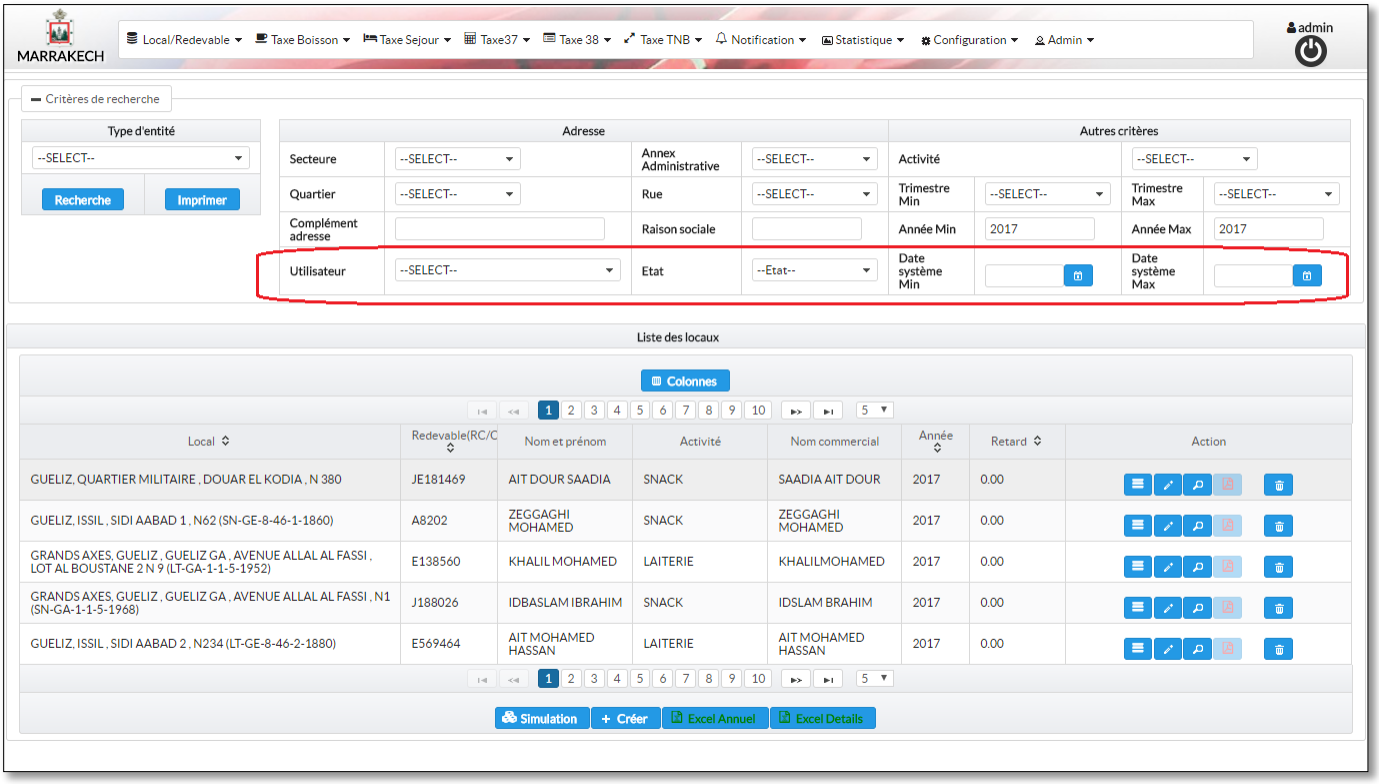


Figure 81 : IHM Liste des paiements, taxe boisson (vue utilisateur)

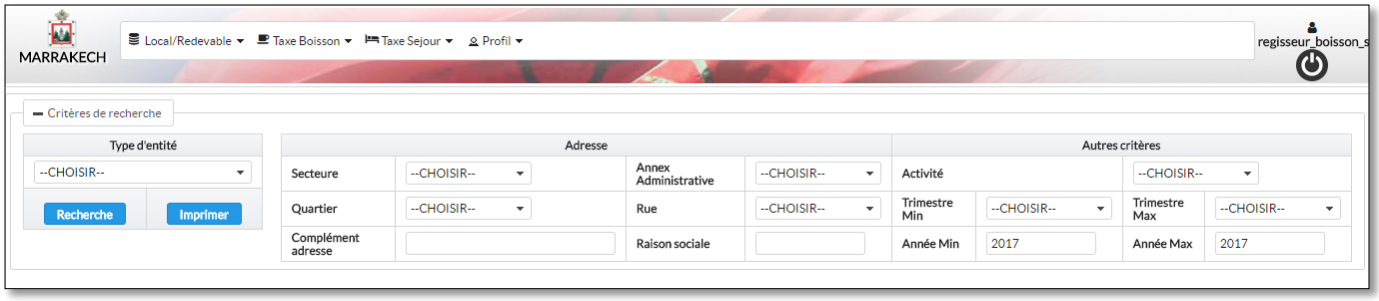


Figure 82 : IHM critères de recherche (vue régisseur ou responsable)

Pour payer un trimestre « Taxe boisson », l'utilisateur doit procéder comme suit :

1- Rechercher le redevable en fonction du RC ou CIN (selon son statut juridique).
2- Une fois le redevable trouvé, le système charge le(s) local/locaux associé à celui-là, et l'utilisateur choisi le local concerné.
3- Par la suite, il complète la saisie des informations relatives au paiement de la taxe à savoir :
   - Le chiffre d'affaire HT ou bien TTC, le système calculera l'un des deux en cas d'absence en fonction de la TVA liée à l'activité du local (exemple : RYAD = 10%).
   - La date de présentation du redevable.
   - Le numéro du trimestre à payer.
   - L'année, s'il s'agit du premier paiement.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

- Le numéro de quittance et la date de quittance.

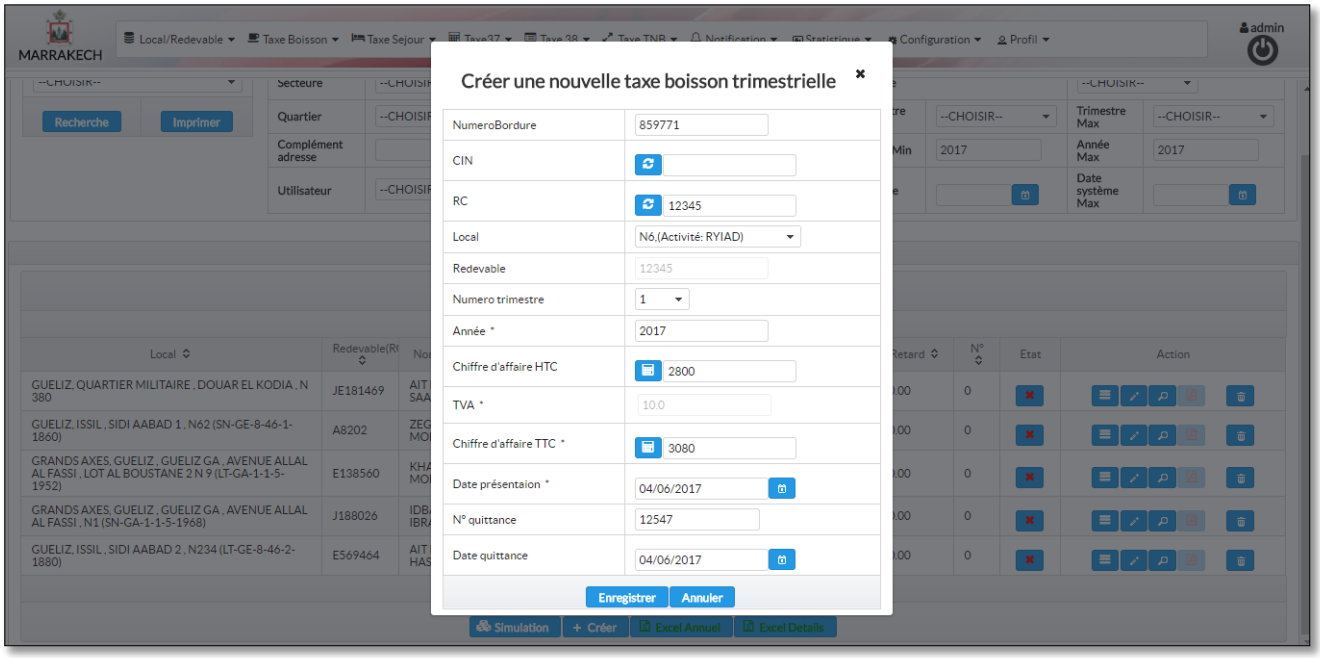


Figure 83 : IHM paiement d'un trimestre « Taxe boisson »

Notre application offre la possibilité d'effectuer une simulation d'un paiement sans l'enregistrer dans la base de données, et d'informer le redevable.

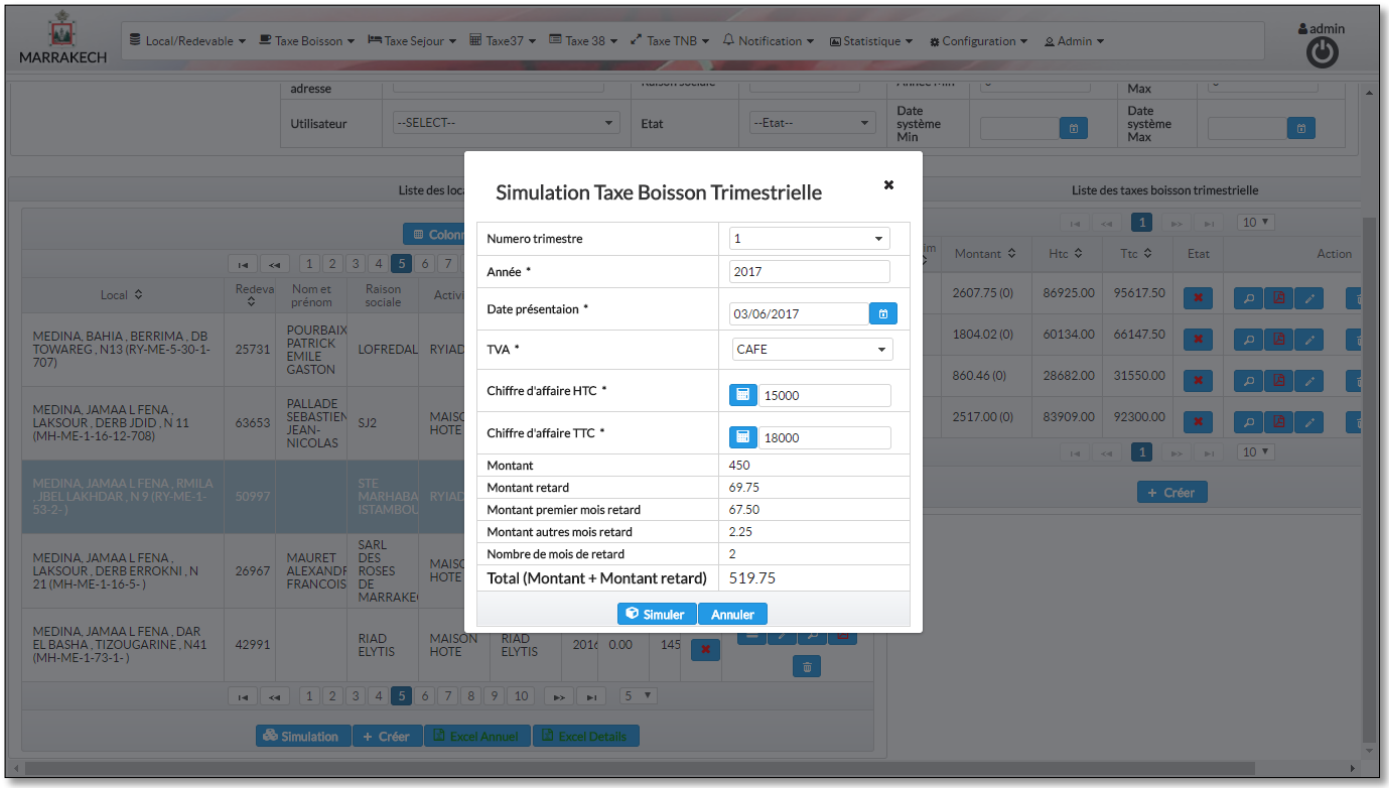


Figure 84 : IHM simulation d'un paiement « Taxe boisson »

La déclaration annuelle consiste à déclarer les chiffres d'affaires réalisées durant chaque trimestre, il suffit juste de saisir la date de présentation. Si le système détecte une différence entre le montant à payer et le montant versé (déclaré), ce dernier affiche cette différence et calculera une pénalité, comme le montre la figure ci-dessous

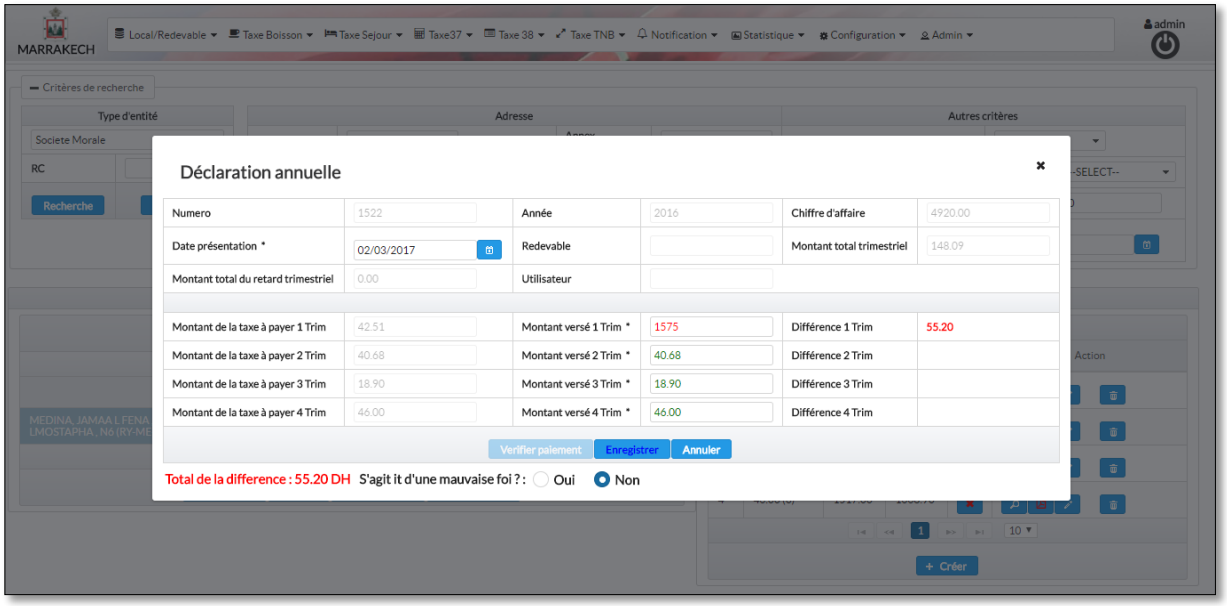


Figure 85 : IHM déclaration annuelle

Les redevables dont le local fait objet de la taxe boisson, doivent avant tout déposer une déclaration d'existence, les informations nécessaire pour effectuer cette déclarations sont comme le montre la figure ci-dessus

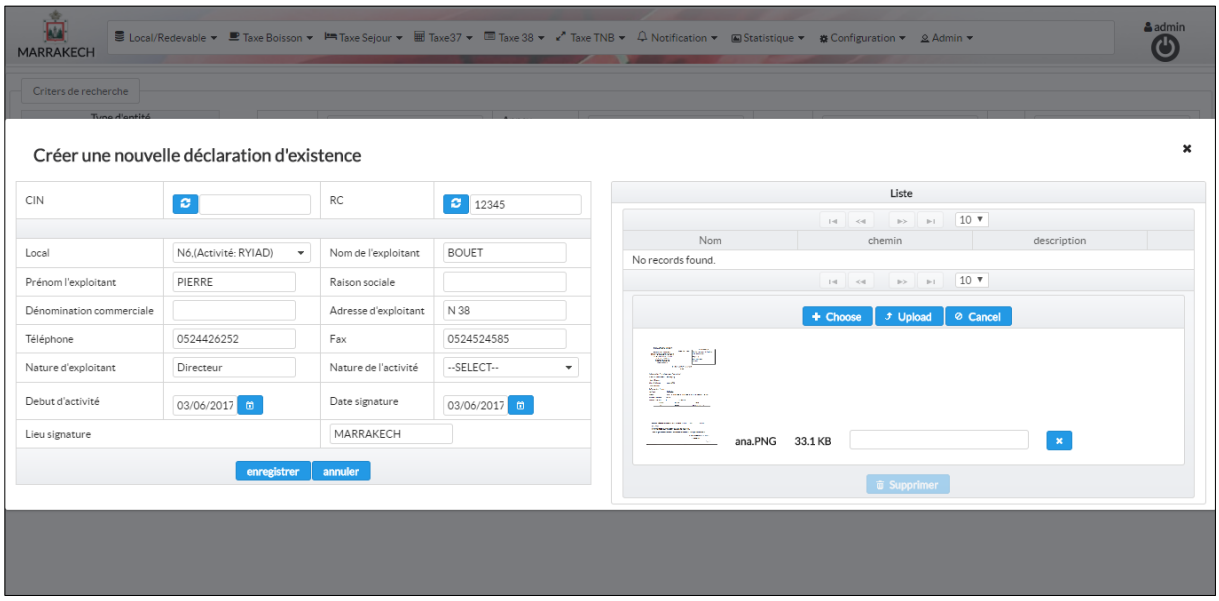


Figure 86 : IHM déclaration d'existence

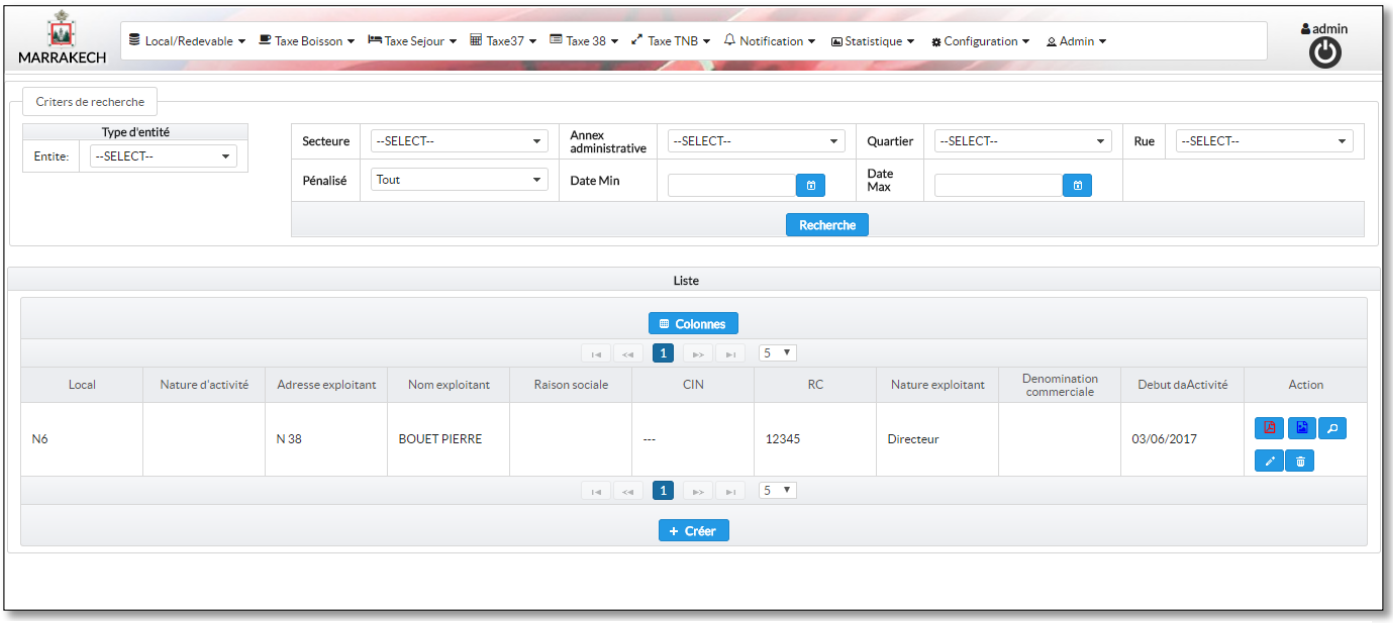


Figure 88 : IHM liste des déclarations d'existence

❖ Taxe de séjour :

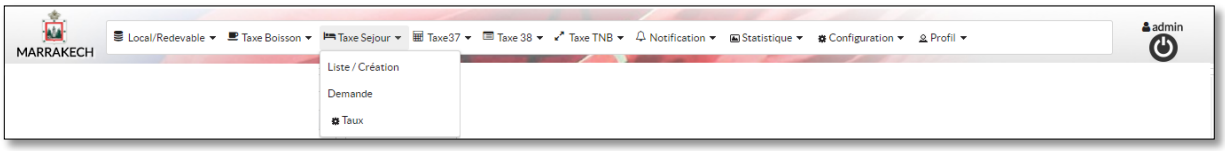


Figure 87 : Menu « Taxe séjour »

La procédure de paiement d'un trimestre taxe séjour, ressemble à celle de la taxe boisson, la différence est au niveau des données à saisir, dans ce cas, on s'intéresse au nombre de nuitées et le nombre de clients.

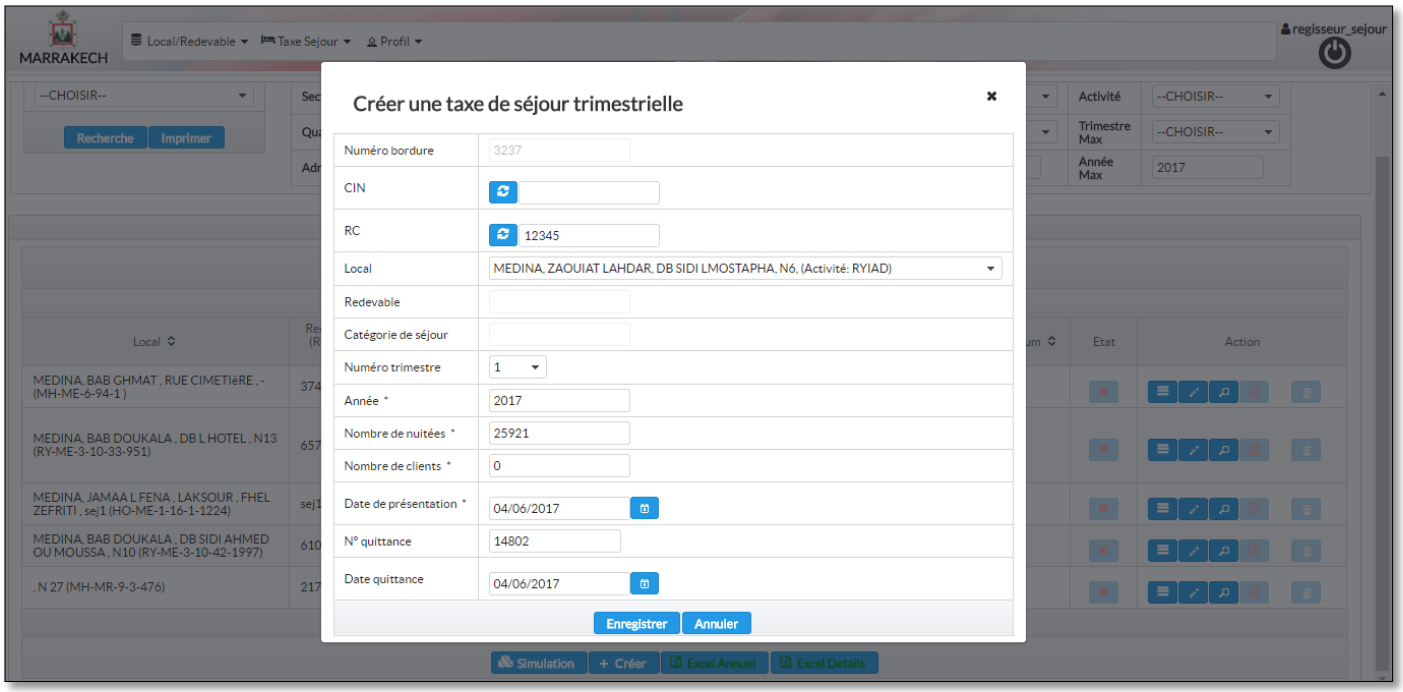


Figure 89 : IHM paiement d'un trimestre « Taxe séjour »

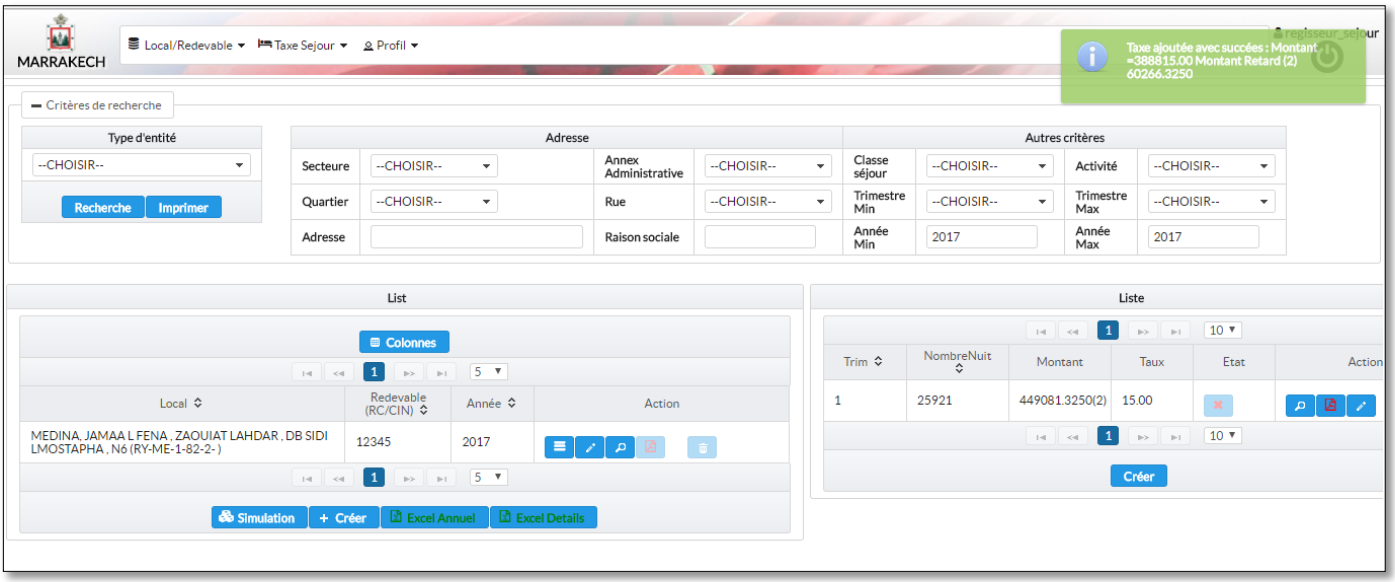


Figure 90 : IHM consultation du paiement « Taxe séjour »

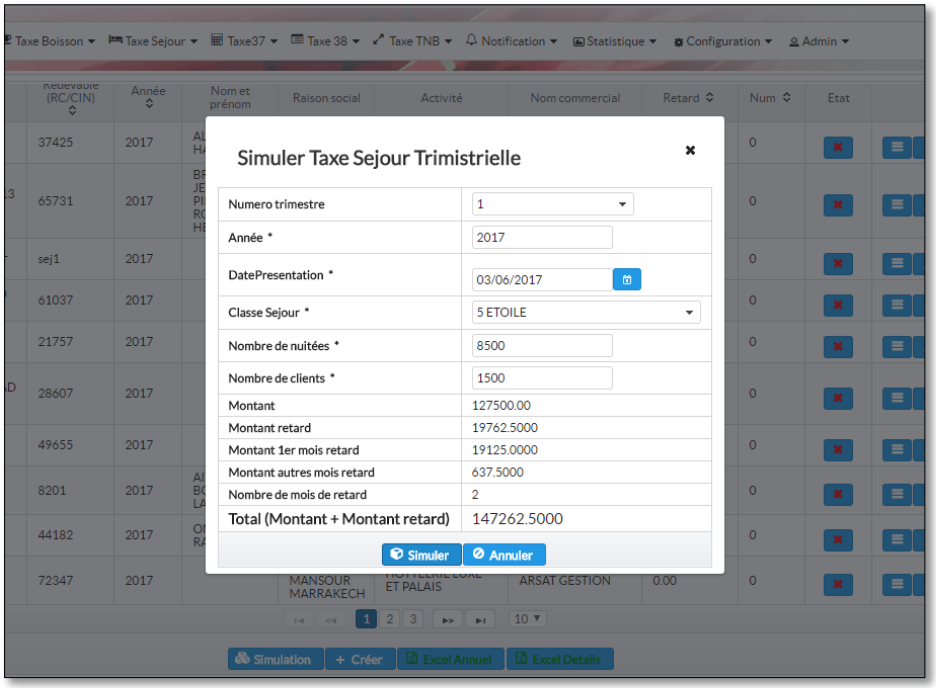


Figure 91 : IHM simulation du paiement d'un trimestre « Taxe séjour »

La déclaration annuelle « Taxe séjour » consiste à déclarer le nombre de nuitées, le nombre de client et le montant de chaque trimestre, il suffit juste de saisir la date de présentation. Si le système détecte une différence entre le nombre de nuitée saisi dans la BD et le nombre de nuitées déclaré, ce dernier affiche cette différence et calculera une pénalité, comme le montre la figure ci-dessous

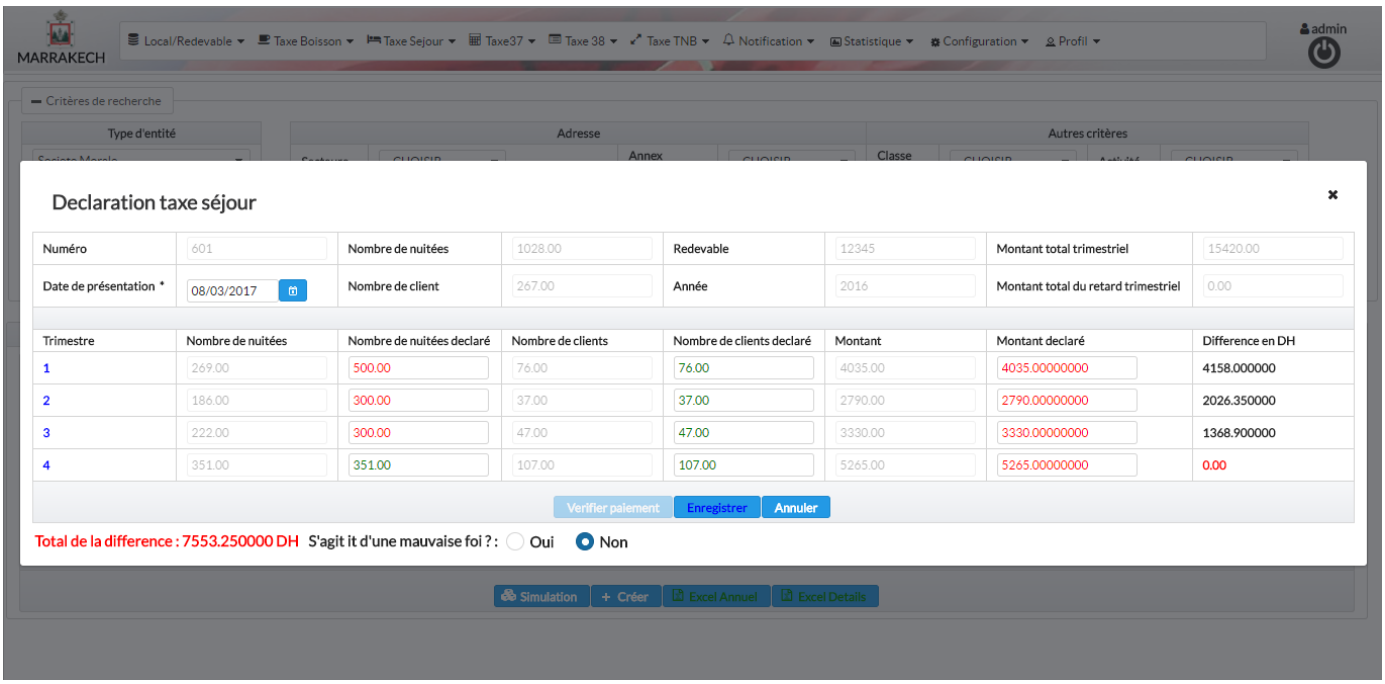


Figure 92 : IHM déclaration annuelle « Taxe séjour »

❖ Taxe de 37 :

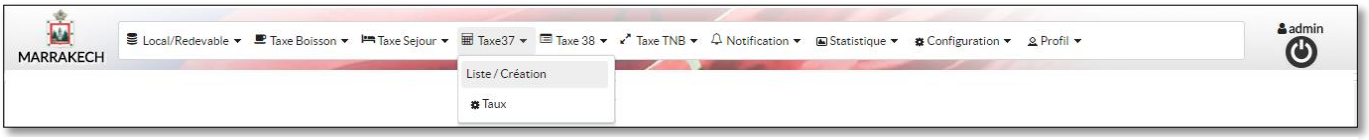


Figure 93 : Menu « Taxe 37 »

Pour payer un trimestre taxe 37, l'utilisateur doit procéder comme suit :

- Rechercher le redevable en fonction du RC/CIN, le système chargera le(s) local/locaux du redevable en question.

- L'utilisateur doit choisir le local concerné, le système affiche automatiquement la liste des objets liés à ce dernier avec leurs situation et la possibilité de cocher l'élément à payer comme le montre la figure ci-dessus.

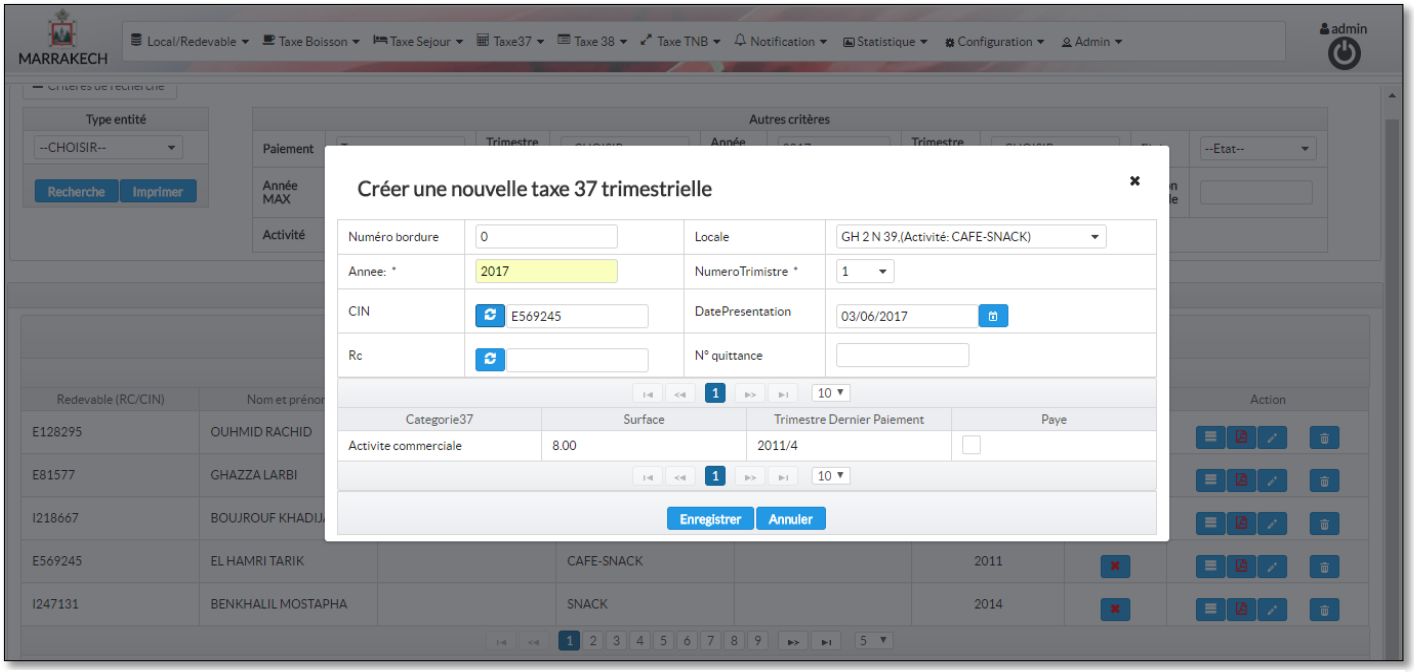


Figure 94 : IHM paiement d'un trimestre « Taxe 37 »

- IHM pour simuler le paiement d'un trimestre « Taxe 37 » :

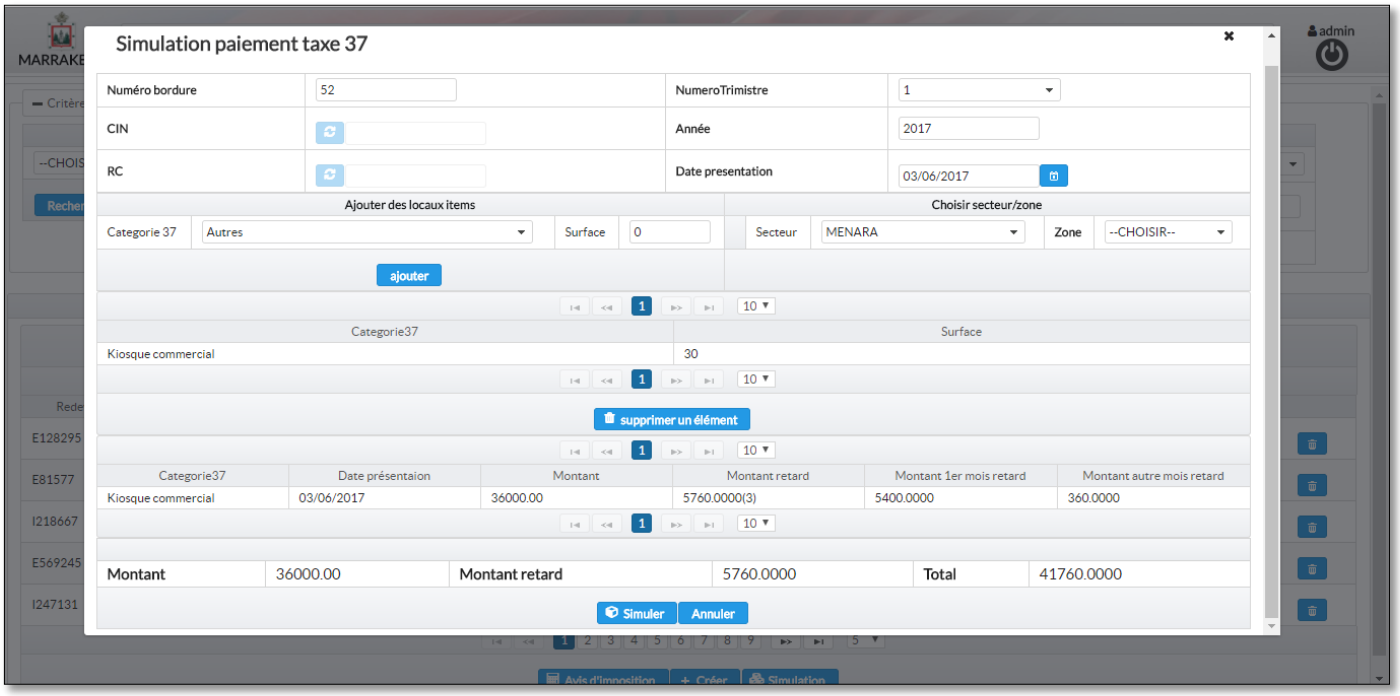


Figure 95 : Simulation paiement d'un trimestre « Taxe 37 »

❖ Taxe de 38 :

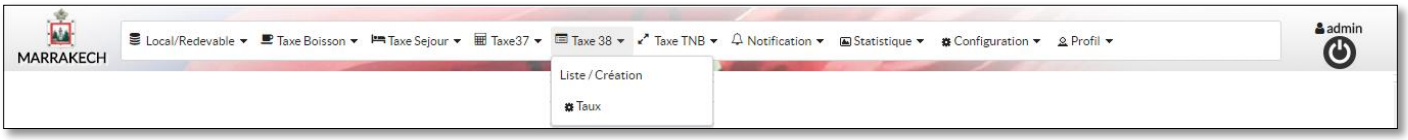


Figure 96 : Menu « Taxe 38 »

Pour payer un trimestre taxe 38, l'utilisateur doit procéder de la même manière que dans la taxe 38

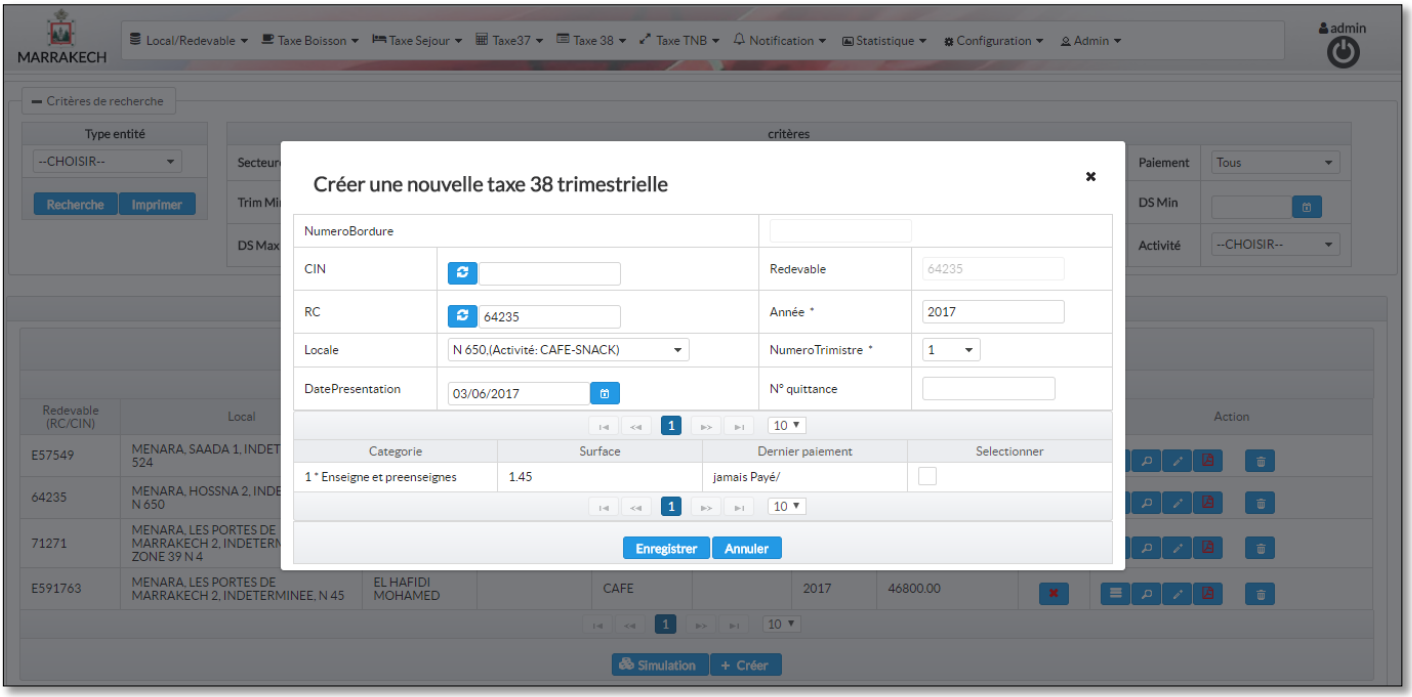


Figure 98 : IHM paiement d'un trimestre « Taxe 38 »

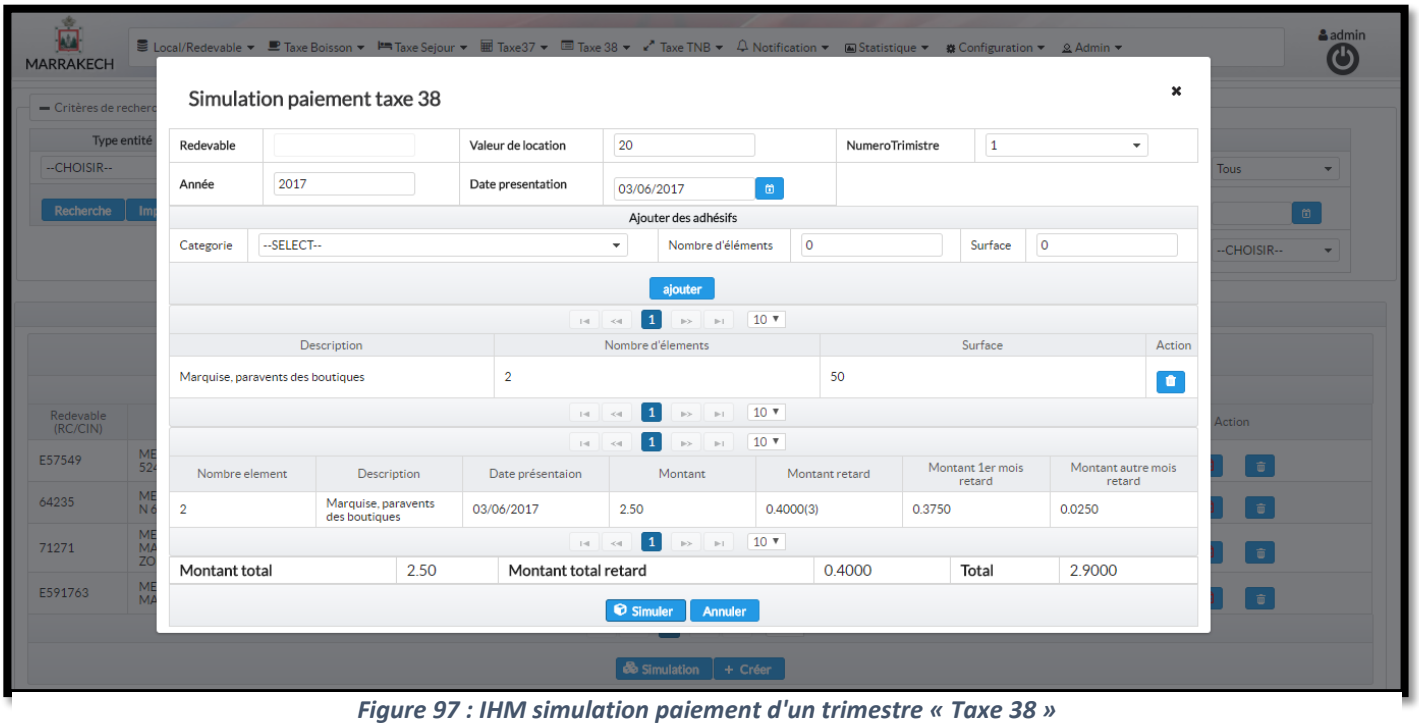


Figure 97 : IHM simulation paiement d'un trimestre « Taxe 38 »

❖ Taxe TNB :

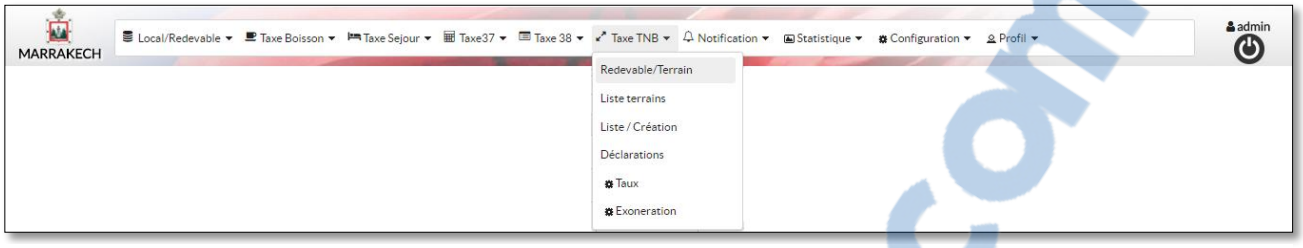


Figure 99 : Menu « Taxe TNB »

➢ IHM paiement annuelle d'une taxe TNB

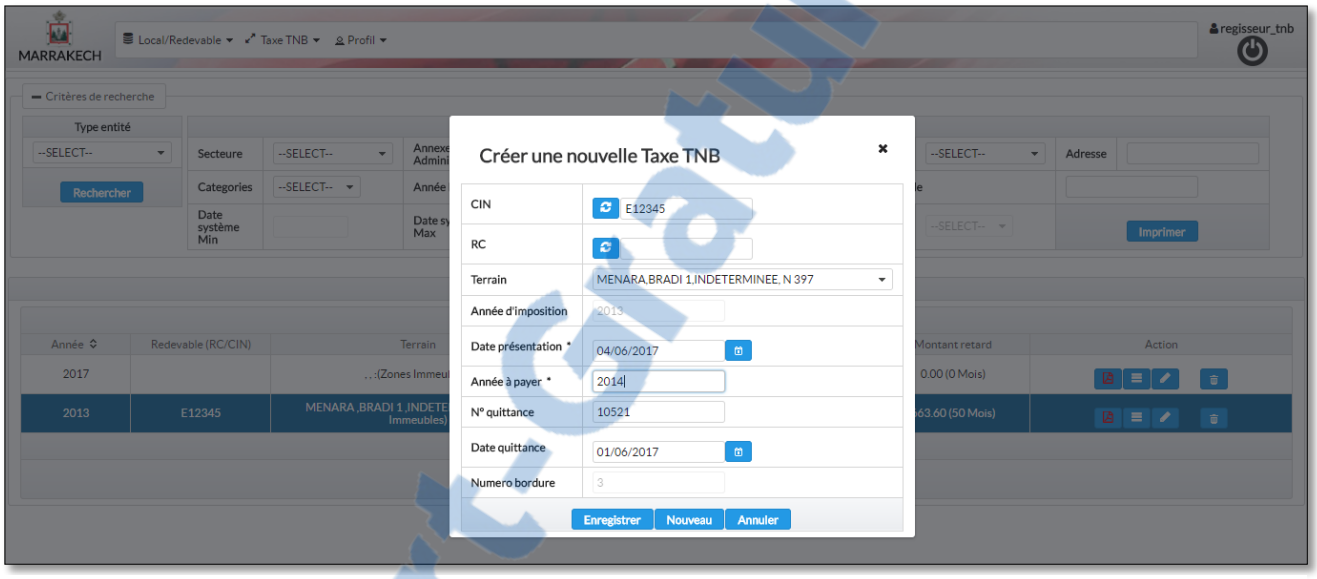


Figure 100 : IHM paiement « Taxe TNB »

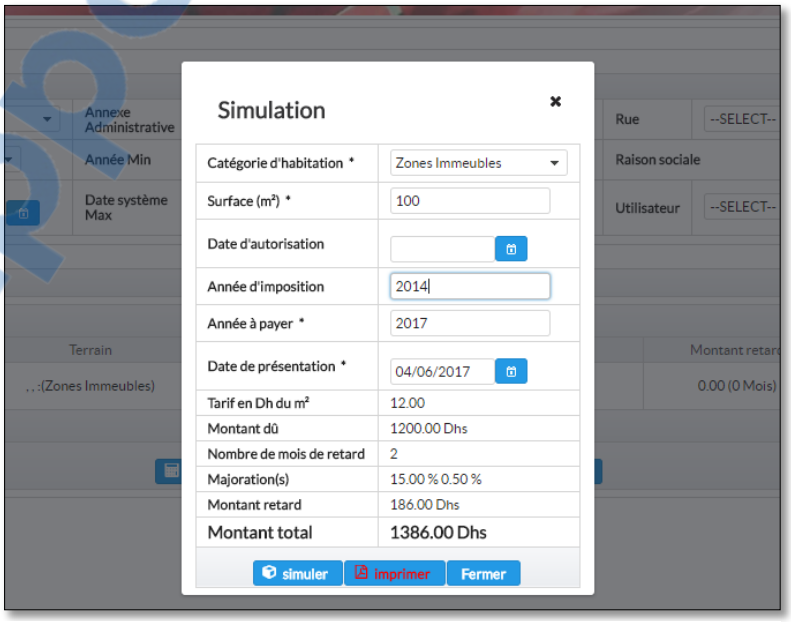
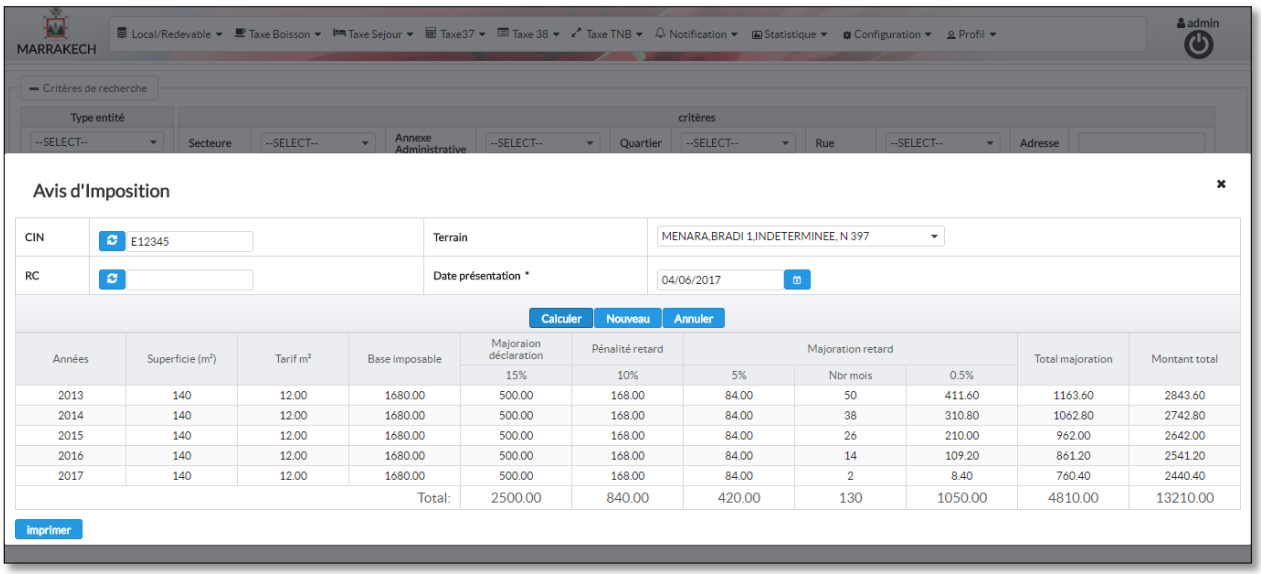


Figure 101 : IHM simulation d'un paiement « Taxe TNB »

Rapport-Gratuit.com

Apres avoir recherché le redevable concerné, le système charge son/ses terrain(s), l'utilisateur doit choisir le terrain en question et saisi la date de présentation



Années	Superficie (m²)	Tarif m²	Base imposable	Majoraion déclaration 15%	Pénalité retard 10%	Majoration retard 5%	Nbr mois	0.5%	Total majoration	Montant total
2013	140	12.00	1680.00	500.00	168.00	84.00	50	411.60	1163.60	2843.60
2014	140	12.00	1680.00	500.00	168.00	84.00	38	310.80	1062.80	2742.80
2015	140	12.00	1680.00	500.00	168.00	84.00	26	210.00	962.00	2642.00
2016	140	12.00	1680.00	500.00	168.00	84.00	14	109.20	861.20	2541.20
2017	140	12.00	1680.00	500.00	168.00	84.00	2	8.40	760.40	2440.40
			Total:	2500.00	840.00	420.00	130	1050.00	4810.00	13210.00

Figure 102 : IHM de génération d'avis d'imposition « Taxe TNB »

❖ Utilisateurs :

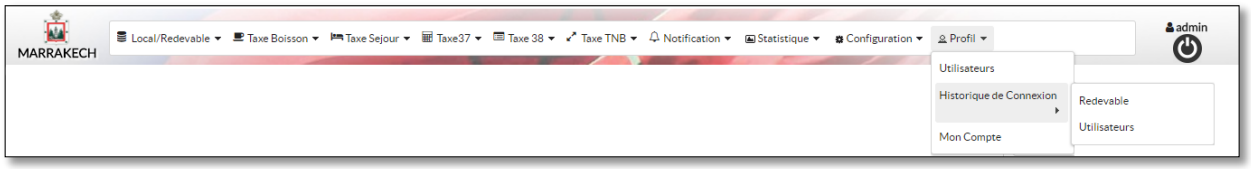


Figure 103 : Menu « Utilisateur »

➢ IHM création d'un utilisateur (Exemple d'un régisseur 38) :

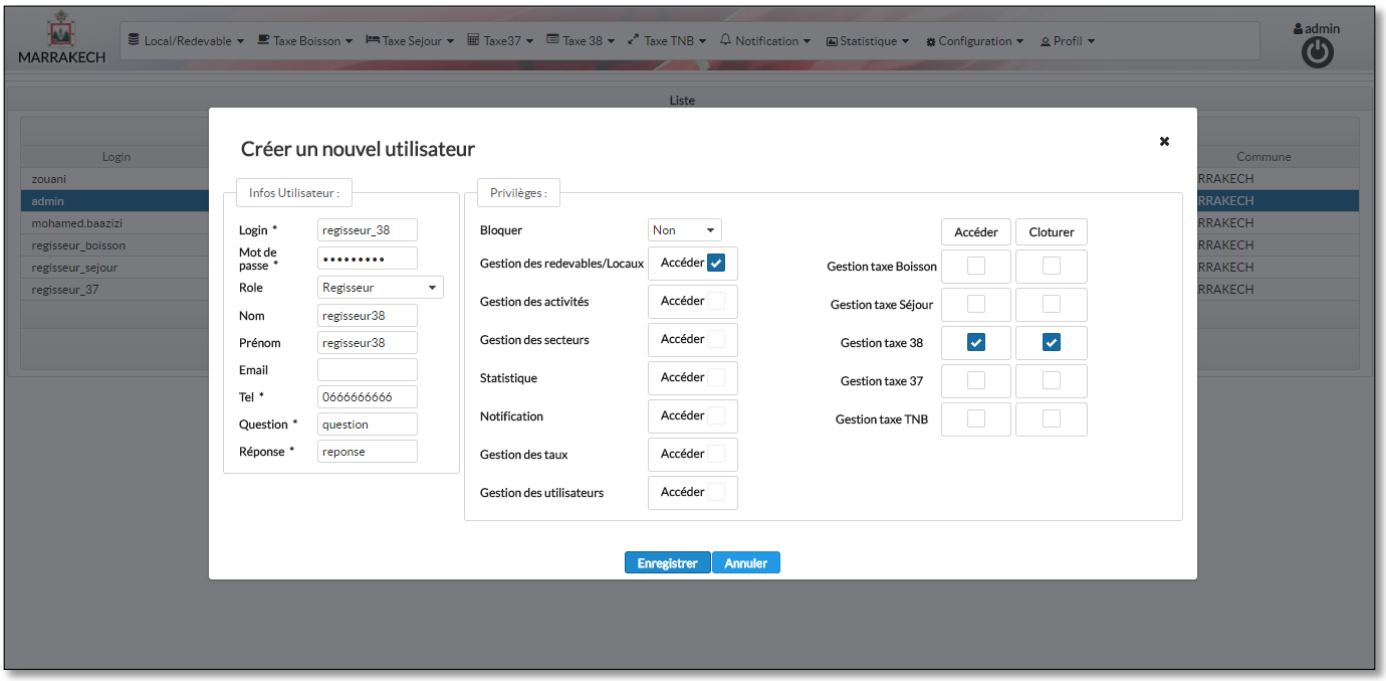


Figure 104 : IHM création d'un utilisateur (Régisseur)

➢ IHM création d'un utilisateur (Exemple du Responsable)

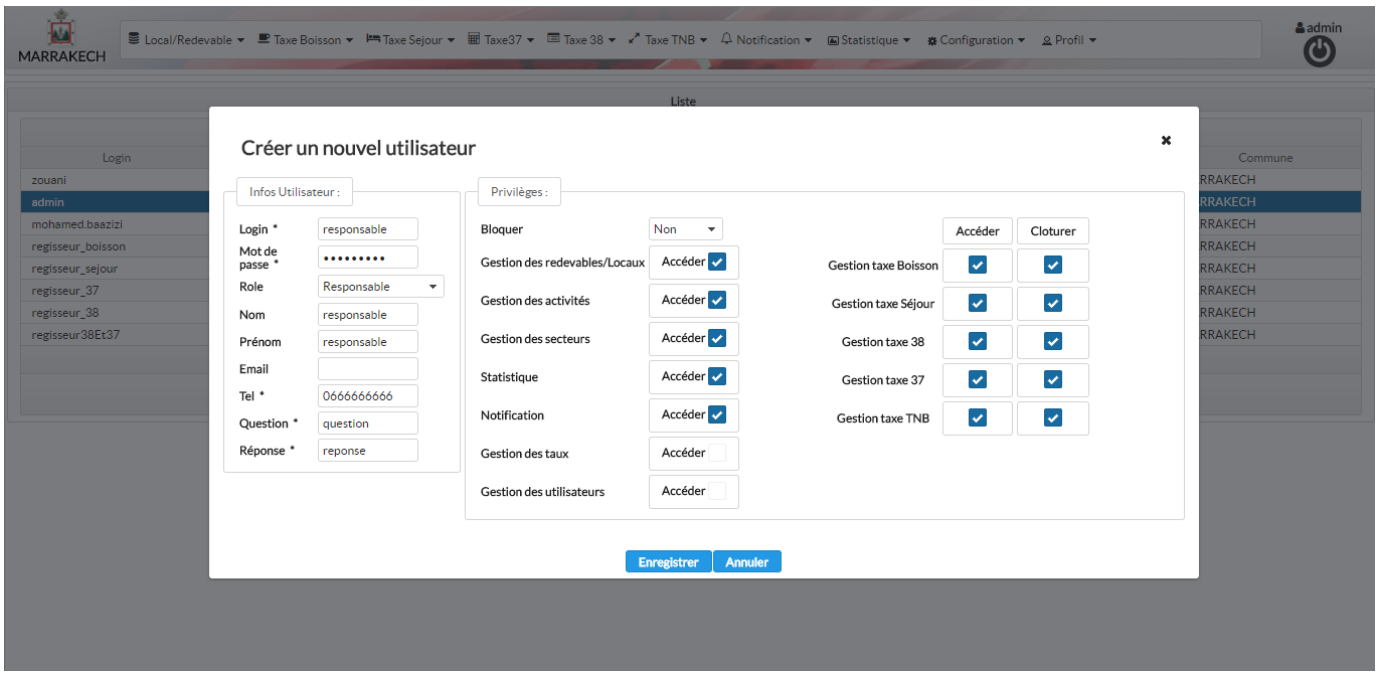


Figure 105 : IHM création d'un utilisateur (Responsable)

❖ Statistiques :

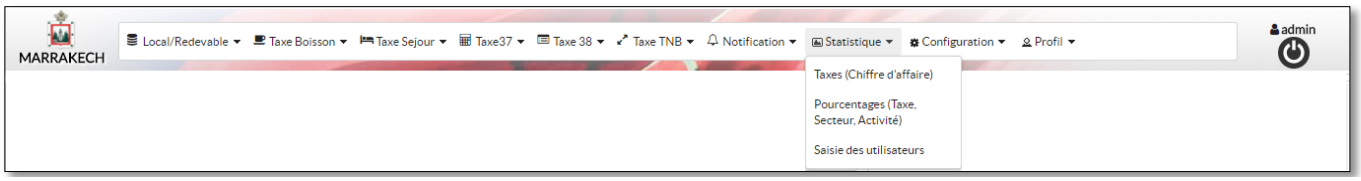


Figure 106 : Menu « Statistiques »

➢ IHM de la consultation de l'évolution du paiement de chaque taxe

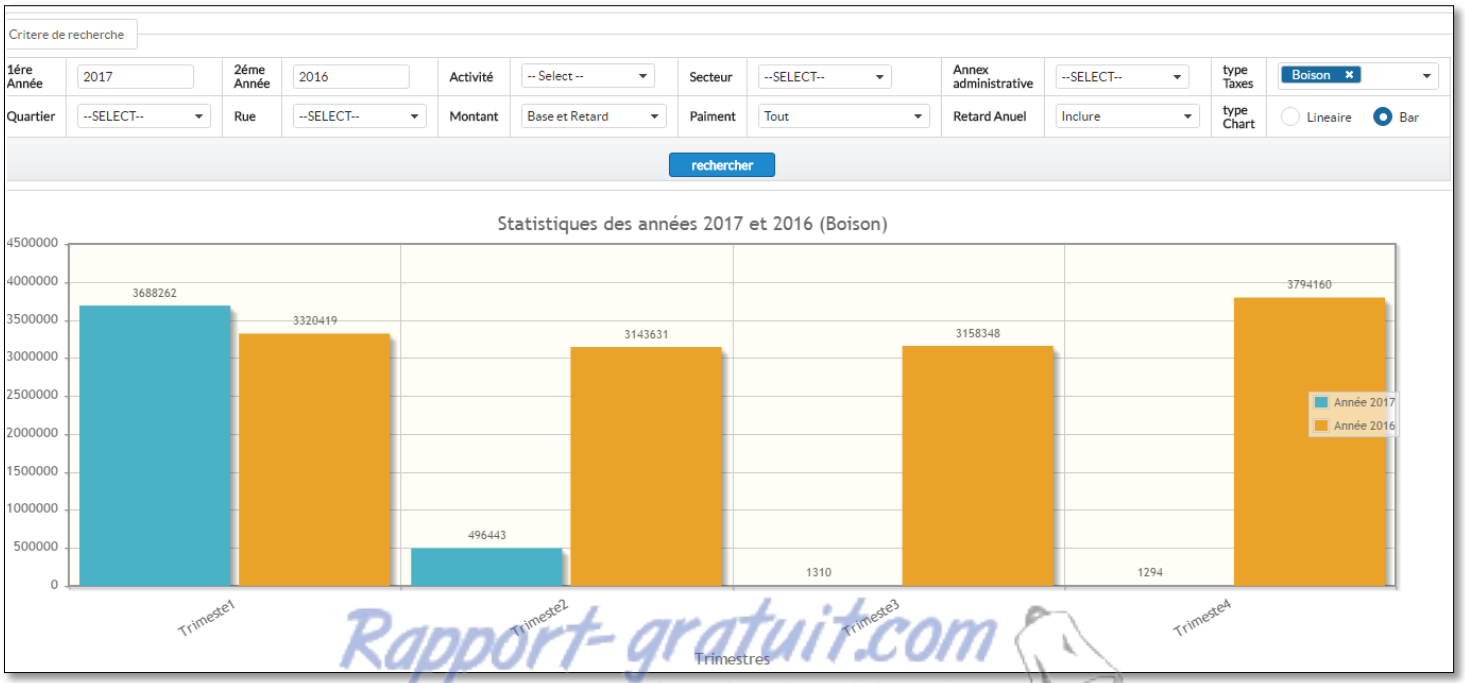


Figure 107 : Exemple de graphe en barre de la taxe boisson

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

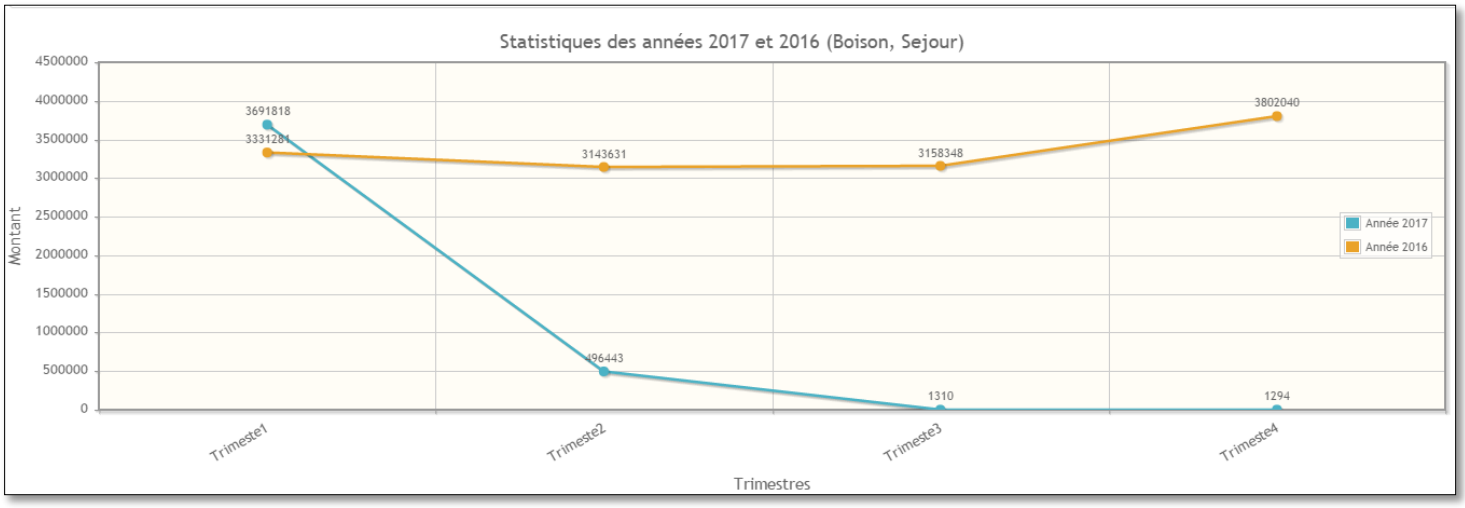


Figure 108 : Exemple de graphe linéaire de la taxe boisson et séjour

- IHM de la consultation des pourcentages des montants collecté pour chaque secteur selon le(s) type(s) des taxes choisi, ainsi que la date

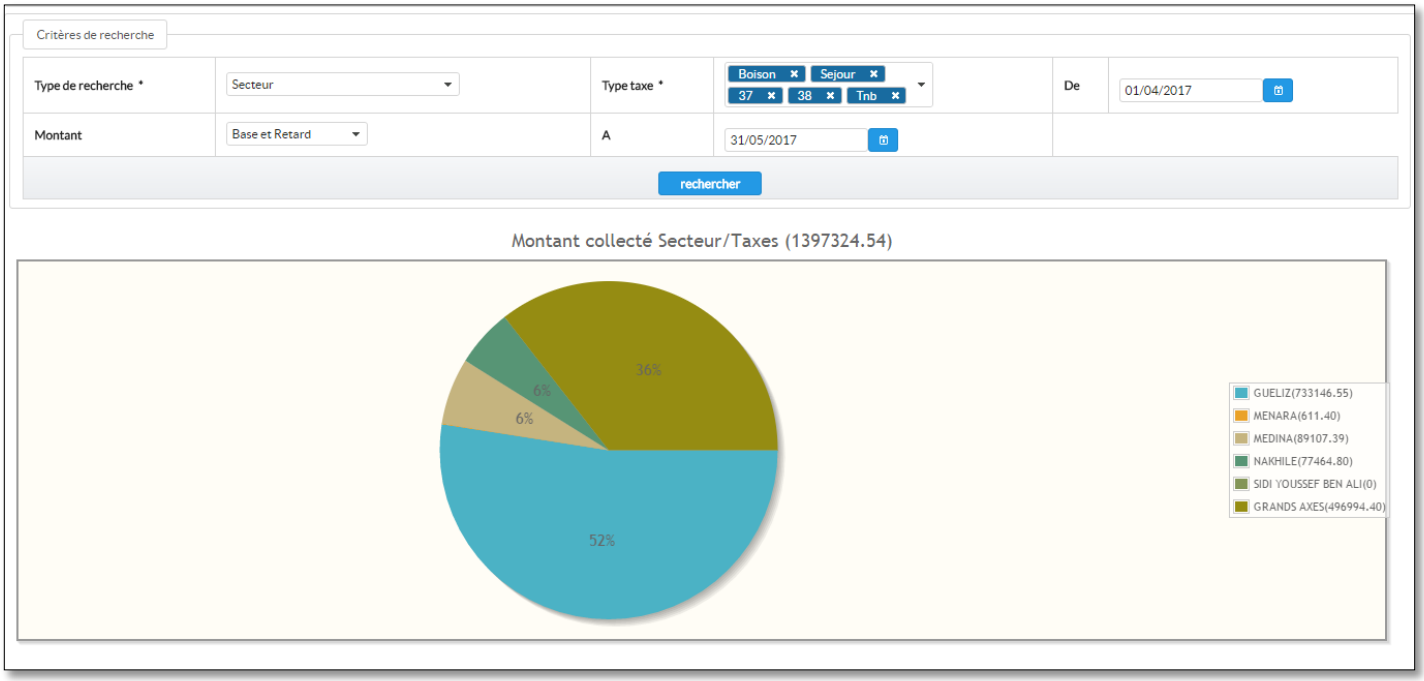


Figure 109 : Exemple de graphe en camembert des montants collectés en fonction des secteurs

- IHM pour consulter les pourcentages des montants collectés pour chaque taxe selon le(s) secteur(s) choisi, ainsi que la date. Cela entrainera la génération d'un graphe supplémentaire qui représente les montants collectés pour chaque catégorie de séjour « Figure » :

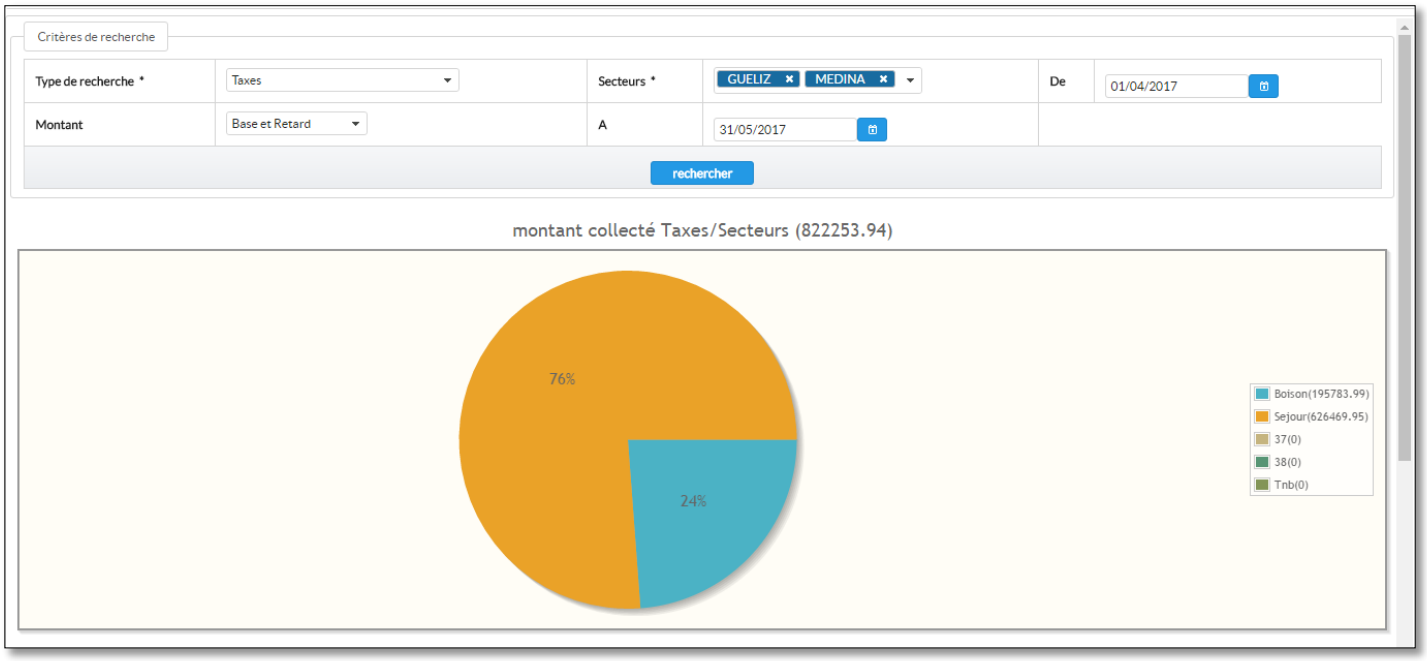


Figure 110 : Exemple de graphe en camembert des montants collectés en fonction des taxes

- IHM pour afficher le graphe des montants collectés pour chaque activité selon le(s) secteur(s) choisi :

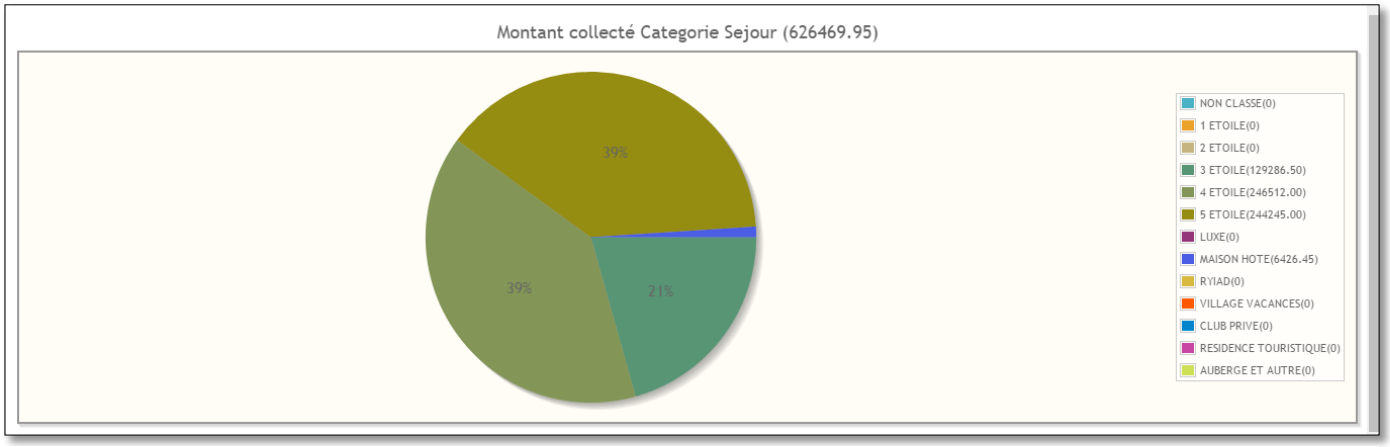


Figure 111 : Graphe en camembert des montants collectés en fonction des catégories de séjour

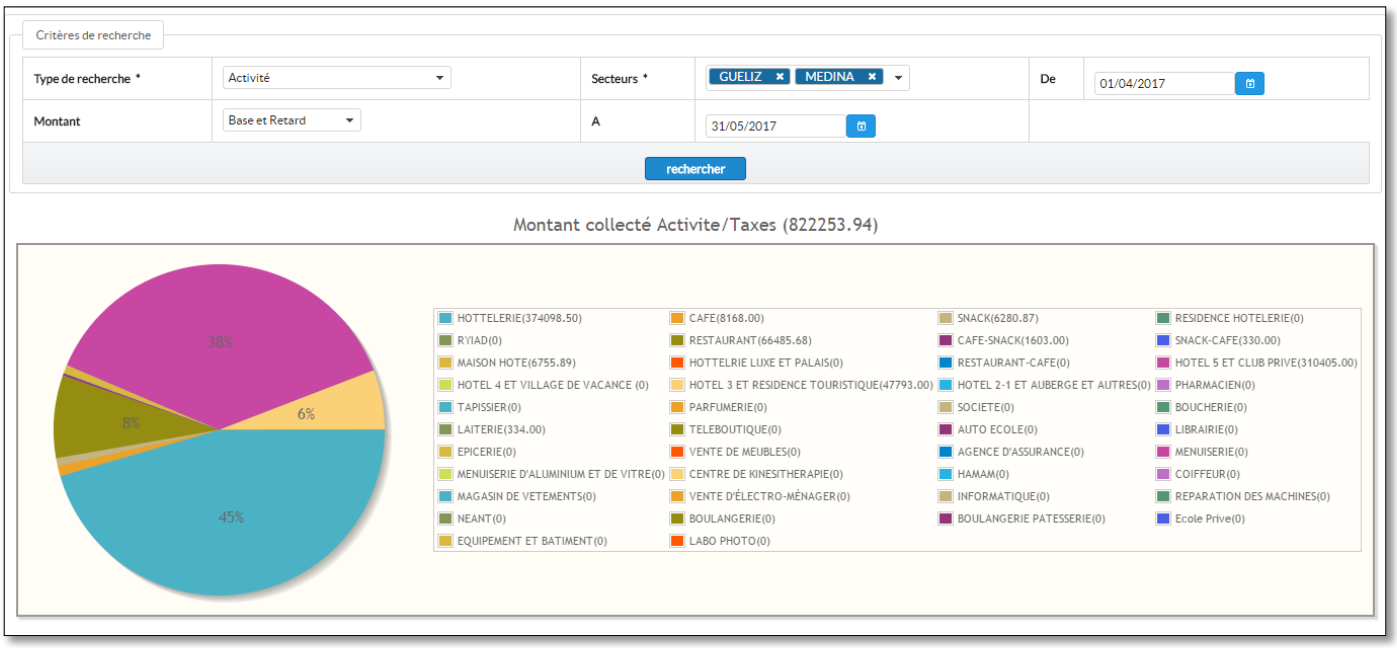


Figure 112 : Graphe en camembert des montants collectés par activité

- IHM pour le suivie de la saisie des paiements trimestriels et annuels par les utilisateurs (régisseurs communaux). Celui-ci est accessible par l'administrateur ou bien le responsable du service de l'assiette.

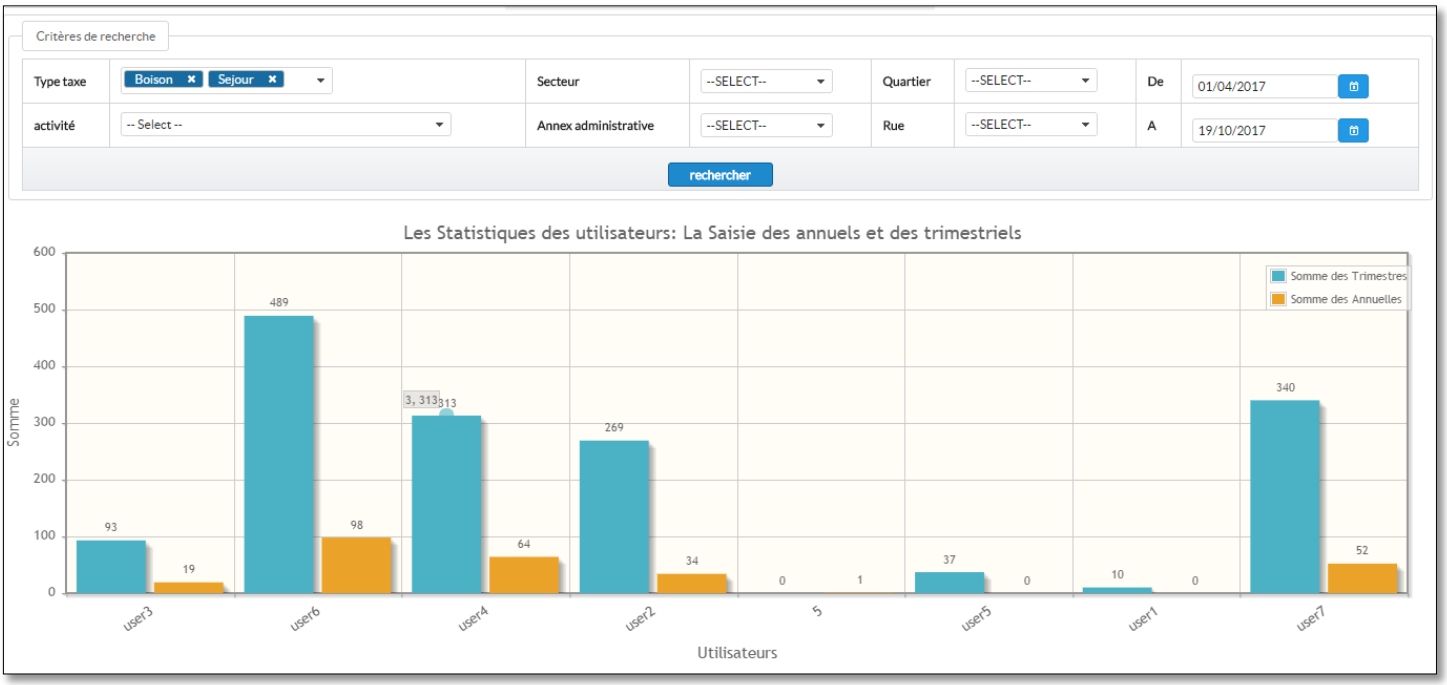


Figure 113 : Graphe du suivi de la saisie des taxes par les utilisateurs

❖ Notifications :

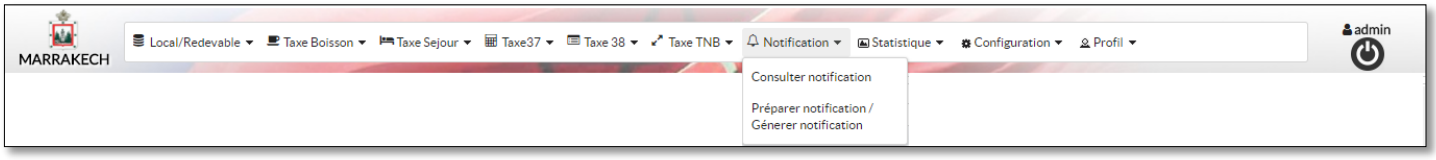


Figure 114 : Menu « Notification »

➢ IHM préparation des notifications (Exemple 1er notification taxe boisson)

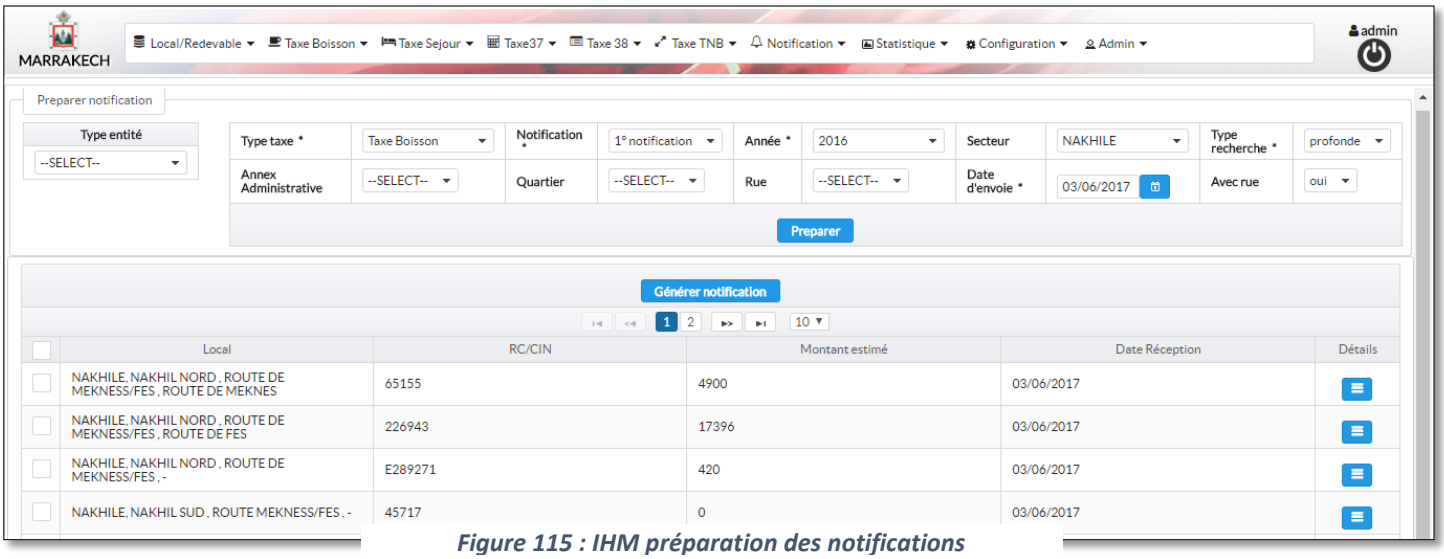


Figure 115 : IHM préparation des notifications

❖ Utilitaires :

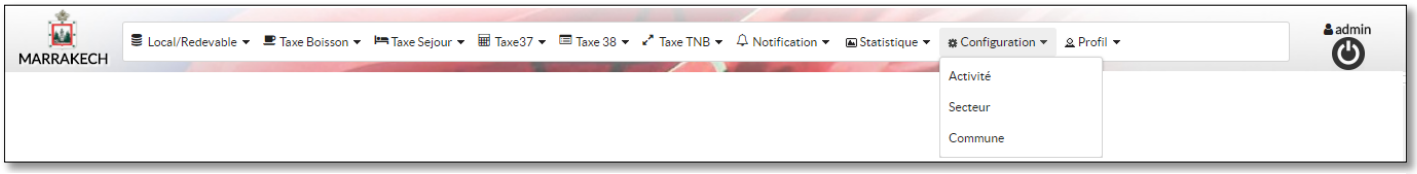


Figure 116 : Menu « Utilitaire »

➢ Gestion des activités

Liste des activités insérées dans la base de données :

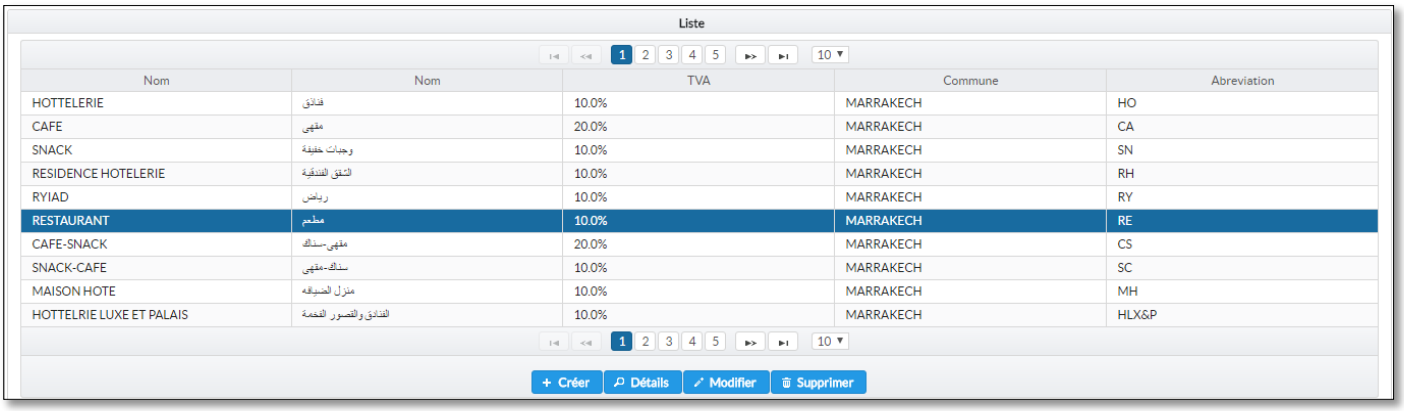


Figure 117 : IHM liste des activités

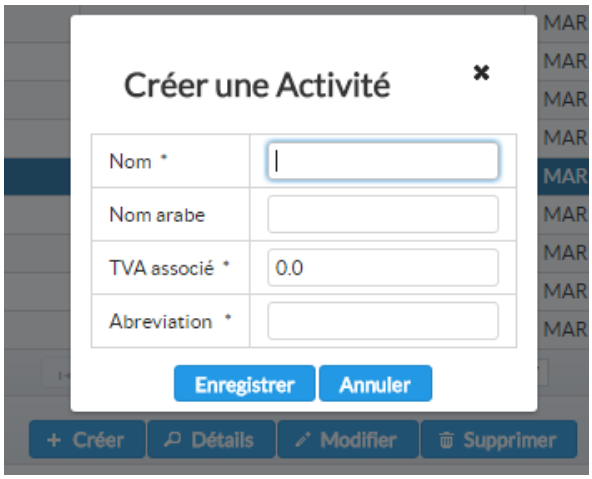


Figure 118 : IHM pour créer une activité

Figure 119 : IHM pour modifier une activité

➢ Gestion des secteurs

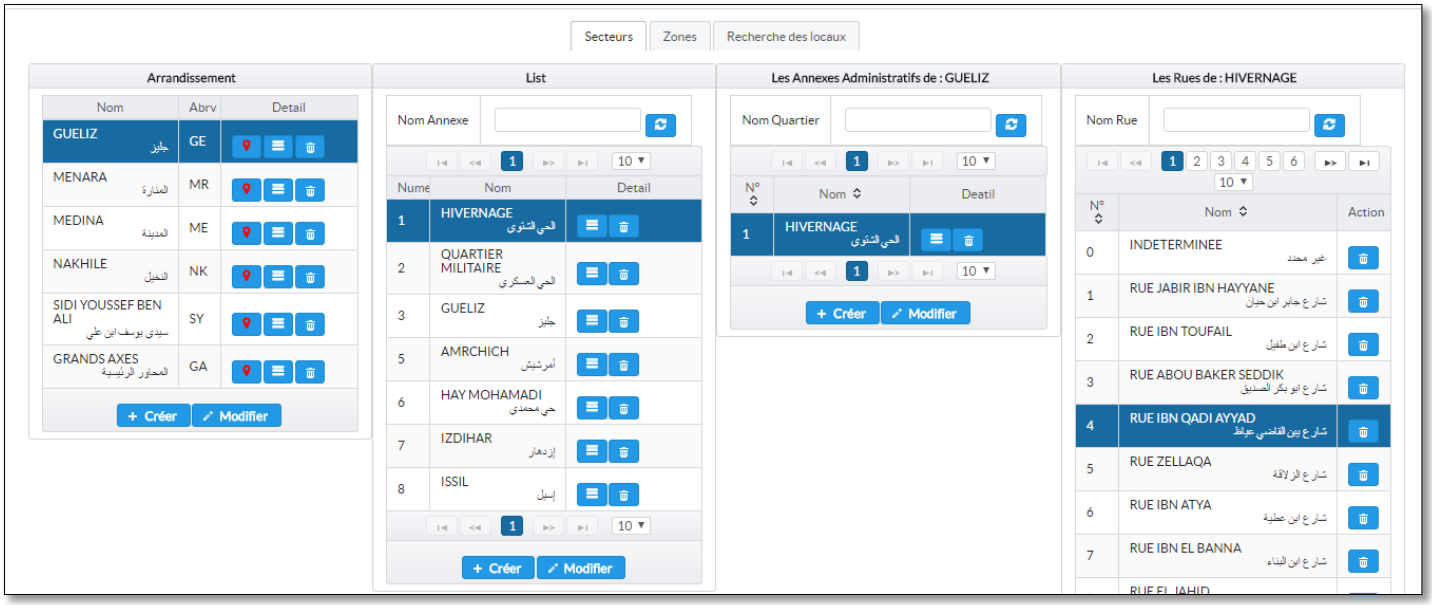


Figure 120 : IHM gestion de l'adresse

Figure 121 : IHM création d'un secteur

❖ Documents générés par le système

➢ Exemple d'un bordereau de versement « Taxe boisson »

ROYAUME DU MAROC
MINISTÈRE DE L'INTÉRIEUR
WILAYA DE LA RÉGION MARRAKECH-SAFI
PRÉFECTURE DE MARRAKECH
COMMUNE DE MARRAKECH
DIRECTION GÉNÉRALE DES SERVICES
DIVISION DE DÉVELOPPEMENT DES
RESSOURCES FINANCIERES

TAXE BOISSON
Trimestrielle

A COMPTER PAR
Service communal de l'assiette
N° de la déclaration : 11
Reçu : 60
Date de présentation : 29/04/2016
Pour l'ordonnateur
Signature :

Bordereau de versement

Trimestre 1 2016

Raison Social RIAD AL JANA SARL (Raison sociale)

Adresse: MEDINA,JAMAA L FENA,ZAOUIAT LAHDAR,DB SIDI LMOSTAPHA
N6

Cin ou Rc: 12345 (RC)

N° téléphone : 0524426252 FAX : 0524524585

Nature de l'exploitant : Gérant

Chiffre d'affaire	Montant	Taux	Mois retard	Montant Retard
1558.66	42.51	3.00	0	0.00

Arrêté le présent bordereau à la somme de (chiffre) : 43 dirhams.
En lettre : QUARANTE-TROIS DHs

Je soussigné certifie exactes les informations mentionnées sur le présent bordereau

A MARRAKECH le04/06/2017

Le Déclarant :

Figure 122 : Bordereau de versement « Taxe boisson »

➢ Exemple d'un avis d'imposition « Taxe TNB »

ROYAUME DU MAROC
MINISTÈRE DE L'INTÉRIEUR
WILAYA DE LA RÉGION MARRAKECH-SAFI
PRÉFECTURE DE MARRAKECH
COMMUNE DE MARRAKECH
DIRECTION GÉNÉRALE DES SERVICES
DIVISION DE DÉVELOPPEMENT DES RESSOURCES FINANCIERES

Exercice :
Article N° :1

Le président du conseil communal de
Marrakech
A

batou arbi

MENARA,ASKJOUR,BRADI 1

N 397

Deuxième avis d'imposition

RC ou CIN : E12345 **Titre foncier :** 12345 **Date d'autorisation :**

N° Lot : 220104 **Sperficie en m²** 140 **Date présentation :** 04/06/2017

Zone : Zones Immeubles **Année d'imposition :** 2013

Années	Sperficie en m²	Tarif Dh/m²	Base Imposable	Majoration defaut	Penalité de retard	Majorations retard			Total Majorations	Montant total
				15%	10%	5%	Nbr mois	0.5%		
2013	140	12.00	1680.00	500.00	168.00	84.00	50	411.60	1163.60	2843.60
2014	140	12.00	1680.00	500.00	168.00	84.00	38	310.80	1062.80	2742.80
2015	140	12.00	1680.00	500.00	168.00	84.00	26	210.00	962.00	2642.00
2016	140	12.00	1680.00	500.00	168.00	84.00	14	109.20	861.20	2541.20
2017	140	12.00	1680.00	500.00	168.00	84.00	2	8.40	760.40	2440.40
Total				2500.00	840.00	420.00	130	1050.00	4810.00	13210.00

Total 13210.00 **DHs**

En lettres : TREIZE MILLE DEUX CENT DIX DHs

Marrakech Le : 04/06/2017

LE PRESIDENT DU CONSEIL COMMUNAL DE MARRAKECH

Figure 123 : Avis d'imposition « Taxe TNB »

➢ Exemple d'une lettre de notification « 1er Notification taxe séjour »

Exercice :

Article N° : 295

Le président du conseil communal de Marrakech

A

DAR RHIZLANE

GUELIZ,HIVERNAGE,AV JNANE EL HARTI

-

Première notification

ROYAUME DU MAROC
MINISTÈRE DE L'INTÉRIEUR
WILAYA DE LA RÉGION MARRAKECH-SAFI
PRÉFECTURE DE MARRAKECH
COMMUNE DE MARRAKECH
DIRECTION GÉNÉRALE DES SERVICES
DIVISION DE DÉVELOPPEMENT DES RESSOURCES FINANCIERES

RC ou CIN : 12045 (RC)

Objet : Taxes de séjour. (1° notification)

J'ai l'honneur de vous rappeler que conformément aux articles 70 à 76 de la loi 47.06 relative à la fiscalité des collectivités locales et de leurs groupements, vous n'avez pas encore deposé vos déclarations de la taxe cible en objet pour les années :

2016,2014,2015

Je vous demande de bien vouloir nous adresser les déclarations en question suivie de paiement correspondant majoré des pénalitées de retard prévus par la dite loi et ce dans un délai de 30 jours à compter de la date de réception de la présente lettre .

Dans le cas contraire , nous serions dans le regret de vous signaler que vous serez taxé d'office, conformément à l'article 158 de la loi précitée.

Dans l'attente , veuillez agréer, monsieur l'expression de mes salutations distinguées.

Marrakech le : 04/06/2017

Le président du conseil communal de Marrakech

Figure 124 : Première notification « Taxe séjour »

Rapport-Gratuit.com

Conclusion

L'objectif de cette partie est d'illustrer la dernière partie du projet qui est la phase de réalisation dans laquelle nous avons présenté et décrit les outils et l'environnement de développement utilisés. Ensuite, il a été question de décrire l'architecture applicative adoptée. Quelques écrans de l'application ont été également présentés en plus de quelques rapports générés par le système.

Conclusion générale et perspectives

Le logiciel e-Taxe répond à la majorité des besoins des bureaux du service d'assiette communal de Marrakech, les agents des bureaux où le programme sera installé sont satisfait puisqu'on a respecté le principe « la simplicité fait la beauté » donc e-Taxe est simple et performant. On touche la performance du logiciel dans plusieurs points dont le plus grand bénéficiaire est le citoyen, parmi les points forts d'e-Taxe :

- Interface simple et facile à utiliser
- Backup et restauration de la base de données à tout moment
- Utilisation d'AJAX pour économiser le temps de rechargement des pages
- Recherche simplifié et rapide
- …

Mis à part le premier contact avec les besoins de la commune de Marrakech en terme de développement de solutions spécifiques, ce stage nous a permis non seulement d'avoir une vision plus claire sur le domaine des taxes, mais c'était aussi une occasion pour travailler sur un projet réel, professionnel et confidentiel.

Notre logiciel est extensible, on pourra ajouter de la communication par internet entre les différentes communes ou préfectures en utilisant le Web Service qu'on peut implémenter dans e-Taxe puisqu'on a utilisé EJB3.

Actuellement, notre solution est en cours de production, notons que nous avons déployé une version de test au niveau du serveur de la commune de Marrakech. Il nous reste la réalisation d'un module qui permet la « déclaration de chômage d'établissement » et la « déclaration de cession, cessation, transfert d'activité ou transformation de la forme juridique de l'établissement » selon les articles 15 et 16 de la loi 47.06.

Rapport-gratuit.com
LE NUMERO 1 MONDIAL DU MÉMOIRES

Bibliographie

- **[Jérôme LAFOSSE]** Java EE, Guide de développement d'applications web en Java, ENI Editions.
- **[Pierre-Alain Muller,Nathalie Gaernter]** Modélisation objet avec UML, Edition Eyrolles.
- **[Rachid ZIZAOUI, 2015]** Guide de la taxe sur les terrains urbains non bâtis, 19/10/2015.
- **[Cagatay Civici],** PrimeFaces User Guide 5.3, First Edition.

Webographie

- **[Oracle Documentation Home]** GlassFish Server 3.0.1 Reference Manual,

  Date de consultation : Mars 2017

  Disponible sur : @https://docs.oracle.com/cd/E19798-01/821-1758/index.html
- **[Java Docs]**

  Date de consultation : Février 2017

  Disponible sur : @http://docs.oracle.com/javase/8/

Annexe A : Introduction à XP

1. Les pratiques de programmation Au cœur des pratiques XP

Les pratiques de programmation que nous présentons ci-après permettent d'améliorer continuellement la conception et le code de l'application pour qu'elle reste toujours aussi claire et simple que possible :

- ✓ Conception simple (simple design) : les développeurs implémentent toujours la solution la plus simple qui puisse fonctionner. En particulier, ils n'inventent pas de mécanismes génériques si le besoin immédiat ne l'exige pas.

- ✓ Remaniement (refactoring) : les développeurs n'hésitent pas à revenir sur le code écrit pour le rendre plus «propre», le débarrasser d'éventuelles parties inutilisées, et le préparer à l'ajout de la fonctionnalité suivante. D'une manière plus générale, cette pratique propose une démarche de conception continue qui fait émerger la structure de l'application au fur et à mesure du développement.

- ✓ Développement piloté par les tests unitaires (test-first programming, unit tests, developer tests) : les développeurs écrivent des tests automatiques pour le code qu'ils produisent, et ce au moment même d'écrire le code en question. Cela leur permet d'une part de mieux cerner le problème avant d'écrire le code, et d'autre part de constituer progressivement une batterie de tests qui les autorise ensuite à apporter rapidement des changements dans l'application, tout en conservant une certaine sérénité.

- ✓ Tests de recette (acceptance tests, customer tests) : le client précise très explicitement ses besoins et les objectifs des programmeurs – en participant à la rédaction de tests de recette. Comme les tests unitaires, les tests de recette doivent être automatiques afin de pouvoir vérifier tous les jours la non-régression du produit.

2. Les pratiques de collaboration

Dans une équipe XP, tous les développeurs travaillent ensemble et interviennent sur la totalité de l'application. Cela garantit la qualité de cette dernière à travers les relectures croisées que cela engendre, mais cela rend également le travail plus motivant et offre une plus grande souplesse dans l'affectation des tâches. Les pratiques qui régissent cette organisation sont les suivantes :

- ✓ Programmation en binôme (pair programming) : lorsqu'ils écrivent le code de l'application, les développeurs travaillent systématiquement à deux sur la même machine – il s'agit là d'une forme «extrême» de relecture de code, dans laquelle les deux développeurs collaborent activement pour résoudre les problèmes qu'ils rencontrent. Les binômes changent fréquemment, ainsi chacun est amené à travailler tôt ou tard avec tous les autres membres de l'équipe.

- ✓ Responsabilité collective du code (collective code ownership) : tous les développeurs de l'équipe peuvent être amenés à travailler sur toutes les parties de l'application. De plus, ils ont le devoir d'améliorer le code sur lequel ils interviennent, même s'ils n'en sont pas les auteurs initiaux.

- ✓ Règles de codage (coding standards) : les développeurs se plient à des règles de codage définies par l'équipe elle-même, de manière à garantir l'homogénéité de leur code avec le reste de l'application, et ainsi à faciliter l'intervention d'autres développeurs.

- ✓ Métaphore (metaphor) : les développeurs n'hésitent pas à recourir aux métaphores pour décrire la structure interne du logiciel ou ses enjeux fonctionnels, de façon à faciliter la communication et à assurer une certaine homogénéité de style dans l'ensemble de la conception, l'idéal étant de décrire le système dans son ensemble par une métaphore unique.

- ✓ Intégration continue (continuous integration) : les développeurs synchronisent leurs développements aussi souvent que possible – au moins une fois par jour. Cela réduit la fréquence et la gravité des problèmes d'intégration, et permet de disposer à tout moment d'une version du logiciel qui intègre tous les développements en cours.

3. Les pratiques de gestion de projet

Les pratiques de programmation et de collaboration permettent de créer un contexte dans lequel une démarche de spécification et de conception purement itérative devient viable. Les pratiques suivantes montrent comment XP exploite cet avantage afin de s'assurer que l'équipe et son client restent en phase tout au long du projet, de façon à converger au plus tôt vers un produit adapté aux besoins du client :

- ✓ Livraisons fréquentes (frequent releases) : l'équipe livre des versions du logiciel à un rythme régulier, aussi élevé que possible – la fréquence précise étant fixée par le client. Cela permet à l'équipe comme au client de s'assurer que le produit correspond bien aux attentes de ce dernier et que le projet est sur la bonne voie.

- ✓ Planification itérative (planning game) : la planification du projet est réalisée conjointement par le client et l'équipe de développement, au cours de séances dédiées, organisées régulièrement tout au long du projet.

- ✓ Client sur site (on-site customer, whole team) : le client est littéralement intégré à l'équipe de développement pour arbitrer les priorités, et définir précisément ses besoins, notamment en répondant en direct aux questions des programmeurs et en bénéficiant du feedback immédiat d'une application aux livraisons fréquentes.

- ✓ Rythme durable (sustainable pace) : l'équipe adopte des horaires qui lui permettent de conserver tout au long du projet l'énergie nécessaire pour produire un travail de qualité et mettre en œuvre efficacement les autres pratiques.

4. Les quatre valeurs de XP

Si elles sont indispensables aujourd'hui pour mettre l'Extreme Programming en œuvre, les pratiques que nous venons de présenter ne suffisent pas pour autant à le définir. Il ne s'agit en définitive que de techniques, destinées à faire émerger un environnement de travail marqué par les quatre qualités érigées en valeurs par XP et qui en font l'essence : la communication, la simplicité, le feedback et le courage.

a. La communication pour une meilleure visibilité

Du point de vue d'XP, un projet de développement est avant tout un effort collectif de création, dont le succès dépend d'une part de la capacité de ses différents intervenants à s'accorder sur une vision commune de ce qui doit être produit, et d'autre part de leur capacité à synchroniser leurs actions individuelles pour atteindre l'objectif commun. Or, ces deux conditions dépendent en majeure partie de la qualité de la communication qui lie ces intervenants entre eux. Mais sur le fond, il n'y a finalement pas grand-chose de spécifique à XP sur ce point.

Ce qui démarque l'approche XP dans ce domaine, c'est l'accent mis sur la communication directe, sur le contact humain. Certes, la communication orale présente des faiblesses en termes de structuration de l'information et de traçabilité, mais elle tire sa force de sa Communication Feedback Simplicité Courage Respect simplicité et des interactions rapides entre les interlocuteurs qui permettent de converger rapidement sur les informations essentielles. En outre, le contact direct permet de véhiculer des informations beaucoup plus personnelles cela donne ainsi l'occasion d'identifier et de résoudre des blocages qui relèvent de positions individuelles, qui sans cela pourraient compromettre les chances de succès du projet. Au quotidien, la communication directe permet donc d'obtenir une «bande passante» nettement supérieure à l'écrit, et XP exploite largement cet avantage dans la plupart de ses pratiques. Ainsi, qu'il s'agisse de l'identification des besoins, de la planification, de la répartition des tâches ou encore de la programmation proprement dite, la plupart des

pratiques d'XP sont conçues pour amener les intervenants du projet à communiquer directement et résoudre les problèmes ensemble.
Cependant, si l'accent est placé sur la communication directe, la communication écrite n'est pas laissée de côté pour autant. Mais elle apparaît sous une forme différente : toutes les informations ayant trait à l'implémentation et la conception se retrouvent dans le code lui-même – dont la clarté fait l'objet de nombreux efforts – et dans la batterie de tests qui l'accompagne, et celles relatives aux besoins du client sont consignées d'une manière on ne peut plus formelle, sous la forme de tests de recette automatiques. Si le client souhaite obtenir des documents supplémentaires, il lui suffit de définir et de planifier les tâches correspondantes comme pour tout autre besoin.

b. La simplicité comme garantie de productivité

Une personne qui arrive sur un projet XP ne doit pas s'étonner de s'entendre répéter toujours les valeurs de la simplicité.
Cette simplicité touche d'abord à la réalisation même du logiciel, dans laquelle un soin particulier est apporté au code pour le débarrasser de toute complexité superflue. En particulier, les mécanismes de généricité ne sont encouragés que lorsqu'ils servent un besoin concret et immédiat, et en aucun cas un besoin futur plus ou moins imaginaire. Les développeurs sont donc invités à implémenter la chose la plus simple qui puisse marcher et, en ce qui concerne ces fameux mécanismes génériques, ou encore ces outils «magiques» qui en font plus que ce qu'on leur demande, ils n'auront pas besoin ! En revanche, simple ne veut pas dire simpliste : les solutions adoptées doivent être aussi simples que possible, mais toutes les duplications doivent être éliminées de sorte que chaque information, chaque mécanisme ne soit exprimé qu'une seule fois.

Cet effort de simplicité s'applique également au client, à qui l'équipe demandera de définir ses besoins et ses priorités avec une grande précision pour éviter d'implémenter des choses inutiles et pour pouvoir se focaliser sur les besoins réellement importants. La simplicité est également recherchée dans le choix des outils (de programmation ou de gestion de projet) et dans la méthode de travail elle-même.

En définitive, XP est fondé sur un pari : «faire simple un jour, avec la possibilité de revenir en arrière le lendemain si un besoin différent apparaît. L'éventuel coût supplémentaire induit sera, d'une part réduit parce que l'application est restée assez simple pour évoluer facilement, et d'autre part sera bien peu de choses au regard des économies réalisées à ne pas faire ce qui n'aurait servi à rien. » C'est au succès de ce pari qu'une équipe XP doit sa vitesse de développement et son ouverture au changement.

c. Le feedback comme outil de réduction du risque

Malgré ce que peuvent laisser supposer son nom et ses apparences immédiates, l'eXtreme Programming est avant tout un processus de réduction du risque dans le projet. Le risque est en effet soigneusement contrôlé à tous les niveaux, par la mise en place de boucles de feedback

qui permettent à l'équipe de développement, comme à son client, de savoir à tout moment dans quel état se trouve réellement le projet, et de pouvoir rectifier le tir au fur et à mesure pour mener le projet à son terme avec succès.

L'exemple le plus saillant de ce principe concerne la pratique des livraisons fréquentes, qui donne à tous les intervenants du projet un feedback régulier sur l'état d'avancement du projet et l'état réel du produit. Mais le feedback ne se borne pas à l'observation : la pratique de la planification itérative permet de tirer parti des informations recueillies pour, à la fois, améliorer la planification elle-même et faire converger le produit vers une solution mieux adaptée aux besoins réels du client.

L'activité de programmation fait également l'objet de divers mécanismes de feedback, tout d'abord à travers les tests unitaires mis en place, qui donnent aux développeurs des indications immédiates sur le fonctionnement du code qu'ils écrivent. Le feedback est également intégré à l'activité de conception, qui est menée tout au long de la réalisation pour s'appuyer sur l'état réel de l'application plutôt que sur l'idée que l'on pourrait en avoir a priori. Enfin, les développeurs s'appuient en permanence sur le feedback de leur binôme pour s'assurer de la validité et de la qualité du code qu'ils produisent. Ce feedback permanent est un facteur de qualité, puisque les intervenants du projet améliorent sans cesse leur travail à partir de l'expérience qu'ils accumulent. Mais c'est également un facteur de vitesse : une équipe XP peut programmer vite, tout en restant sereine parce qu'elle sait qu'elle dispose de nombreux mécanismes pour s'assurer que le projet reste sur la bonne voie.

d. Le courage de prendre les bonnes décisions

L'expérience fait apparaître que la mise en œuvre des pratiques XP requiert une certaine dose de cran. En effet, il faut du courage pour se lancer dans un projet sans avoir au préalable tout spécifié et conçu dans le détail, même si l'on sait que le processus suivi comporte de nombreux mécanismes de feedback. Il faut également du courage pour se borner à réaliser des choses simples, se focaliser uniquement sur les besoins du moment en se disant qu'on pourra adapter l'application à de nouveaux besoins, le moment venu. Il en faut aussi pour accepter de jeter une partie du code qui est devenue inutile, ou récrire une partie de code jugée trop complexe.

Enfin, il faut du courage pour appliquer les principes de communication et de feedback, en particulier lorsqu'il s'agit de maintenir une transparence complète, même lorsque les nouvelles ne sont pas bonnes, ou encore lorsqu'il s'agit de travailler ouvertement avec son binôme en acceptant de lui montrer nos propres limites ou lacunes. Plus généralement, nous avons pu constater qu'il faut du courage pour mettre en place des méthodes nouvelles sur des projets dont les enjeux sont toujours stratégiques.

الحمد لله الذي بحمده تتم الصالحات