



**Conseillers en Gestion
et Informatique**

Mémoire de Projet de Fin d'Études

Pour l'Obtention du Diplôme

Ingénieur d'Etat

Filière

Génie Informatique

Projet : Michelin

Mise en place et alimentation d'un portail D3C par les données de facturation FAE afin de faciliter la lisibilité et la lecture des factures par les clients.

Par :

Mr Ilyas BENFATTOUM

Sous l'encadrement de :

Pr. Youssef BALOUKI (AIAC)

M. Youness HAJJI (CGI)

M. Khalid TAIB (CGI)

2018/2019

Dédicaces

Je dédie ce travail à :

Ma mère, à laquelle personne ne peut compenser toutes les peines endurées, les privations et les sacrifices qu'elle a consentis pour mon éducation et mon bien être.
Mon père, Pour tout ses sacrifices sans limite, et auquel je souhaite du profond de mon cœur qu'il soit fier de moi.

Mes deux frères, Pour leurs soutien et leurs encouragements.
Toute ma grande famille, Qu'elle trouve ici l'expression de ma profonde reconnaissance pour tout l'encouragement, les prières et le soutien.

Mes très chères amis, Je vous remercie d'avoir toujours été à l'écoute.
Tous mes Professeurs, Auxquels j'ai tout le respect et hommage pour avoir contribué à ma solide formation.

Tous ceux et celles que j'ai omis de citer, ceux qui m'aiment et qui ont cru en moi.

Remerciements

Au terme du stage de fin d'études effectué au sein de la Société CGI Technologies et solutions, j'adresse un remerciement spécial à M. BALOUKI , Professeur à l'AIAC, pour son encadrement, son soutien moral, ainsi que pour tous ses conseils instructifs et les remarques pertinentes durant toute la période de ce travail.

Je tiens également à remercier et exprimer ma profonde gratitude à mon encadrant à CGI, M. Hajji Youness de m'avoir offert l'opportunité d'effectuer mon stage de projet fin d'études au sein de son l'équipe, et aussi M. TAIB Khalid, pour leurs soutien, leurs conseils et leurs directives précieuses tout au long de ce projet.

Mes remerciements s'adressent également aux membres du jury qui ont accepté d'évaluer ce travail. J'aimerais toutefois, exprimer ma gratitude à toute l'équipe Michelin pour leur accueil et leur sympathie.

Que le corps professoral et administratif trouve ici mes sincères remerciements pour l'inestimable qualité de formation qui m'a été dispensée.

Une vive reconnaissance est exprimée à toutes les personnes qui ont contribué de près ou de loin à l'élaboration et à la réalisation de ce travail.

Résumé

Le présent document constitue une synthèse de mon projet de fin d'études, effectué au sein de CGI Technologies et solutions, plus particulièrement au sein de l'équipe Michelin. Ce projet a pour but la mise en place et l'alimentation d'un portail D3C par les données de facturation en provenance de FAE afin de faciliter la lisibilité et la lecture des factures par les clients.

Durant ce projet, j'ai eu pour mission, dans un premier temps, de cerner le sujet, de comprendre le métier du service, et de mener une étude concernant l'existant dans le but de conclure ses limites. J'ai fixé ensuite Scrum comme méthodologie du travail, puis j'ai mené une analyse globale des besoins permettant de réaliser le product backlog du projet suivie d'une étude technique des différents outils. Pour chaque sprint, une étude fonctionnelle suivie de la conception ont eu lieu avant d'aborder la réalisation et les tests. Le projet a été élaboré en 4 sprints :

- Le premier sprint est une étude fonctionnelle, qui comporte l'analyse des besoins fonctionnels et non fonctionnels.
- Le deuxième sprint est la conception en langage de modélisation objet UML.
- Le troisième comprend la mise en œuvre et l'implémentation de projet.
- Le quatrième et le dernière sprint comporte un jeu de tests.

Mots-clés : Facture, D3C, FAE, Application web, JEE, UML, Scrum.

Abstract

This document is a summary of my end-of-studies project, carried out within CGI Technologies and Solutions, more specifically within the Michelin project. This project aims at setting up and feeding a D3C portal with data coming from FAE to facilitate readability and reading of invoices by the customers.

During this project, my mission was initially to identify the subject, to understand the profession of service, and to conduct a study of the existing in order to conclude its limits. I then set Scrum as the methodology of the work, then I conducted a global needs analysis to realize the product backlog of the project followed by a technical study of the different tools. For each sprint, a functional study followed by the design took place before addressing the realization and the tests.

The project was developed in 4 sprints :

- The first sprint is a functional study, which includes the analysis of functional and non-functional needs.
- The second sprint is the design in UML object modeling language.
- The third one includes the implementation of the project.
- The fourth and last sprint has a test set.

Keywords : Invoice, D3C, FAE, Web Application, JEE, UML, Scrum.

Liste des abréviations

FAE	Facturation Europe
CLR	Commandes et bordereaux de livraison RT
BL	Bon de Livraison
API	Application Programming Interface
DAO	Data Access Object
IHM	Interface Homme Machine
JDBC	Java DataBase Connectivity
JEE	Java Enterprise Edition
MVC	Model-View-Controller
SGBD	Système de Gestion de Bases de Données
UML	Unified Modeling Language
XML	eXtensible Markup Language
XSD	XML Schéma Description

Liste des tableaux

3.1 Cas d'utilisation : Gestion documents	42
3.2 Cas d'utilisation : Gestion demandes	45
3.3 Cas d'utilisation : Demande de création de compte	47
3.4 Les termes du modèle de données	56
4.1 Les tests réalisés	85

Table des figures

1.1	Logo CGI Technologies and solutions	18
1.2	Logo utilisé lors du rachat de Logica par CGI	18
1.3	Les services de CGI	19
1.4	Quelques clients de CGI par secteur	20
1.5	Organigramme CGI Rabat	22
1.6	Logo de l'entreprise Michelin	23
1.7	Le Contrat Convergence entre CGI et Michelin	23
1.8	Le processus de Scrum [Ref3]	28
1.9	Les rôles SCRUM [Ref4]	29
1.10	Diagramme de GANTT	30
1.11	Diagramme d'ishikawa	31
2.1	La gestion des activités Michelin pour le périmètre de facturation	33
2.2	La solution proposée pour la gestion des activités Michelin	35
2.3	Le Product Backlog	36
3.1	Diagramme de cas d'utilisation global	41
3.2	Diagramme de cas d'utilisation du système	42
3.3	Prototype d'interface consulter document	43
3.4	Prototype d'interface télécharger document	44
3.5	Prototype d'interface consulter demandes	46
3.6	Prototype d'interface approuver demande	46
3.7	Prototype d'interface demande de création de compte	48

3.8 Le diagramme de séquence du scénario "Extraction des données et génération XML"	49
3.9 Le diagramme de séquence du scénario "Gestion demandes"	50
3.10 Le diagramme de séquence du scénario "Gestion documents"	51
3.11 Le diagramme de BPMN du processus de D3C	52
3.12 Le diagramme de BPMN du sys-générateur	53
3.13 Le diagramme de classes	54
3.14 Le modèle de données	55
3.15 Le diagramme de déploiement	57
4.1 Le Patron MVC	63
4.2 Le framework Spring	64
4.3 Framework Hibernate	64
4.4 Spring Security	65
4.5 Architecture technique de la solution	69
4.6 Squelette du batch d'extraction FAE	70
4.7 Structure du fichier de sortie	71
4.8 Le curseur CUR-DIST	72
4.9 Le JCL d'extraction des données	73
4.10 Le contenu fichier plat	73
4.11 Squelette du batch de génération XML	74
4.12 Le XSD DIST	75
4.13 Le JCL de génération XML	76
4.14 Le fichier XML généré	76
4.15 Page d'authentification	77
4.16 Page de ré-initialisation du mot de passe	78
4.17 Page de création de compte	78
4.18 Page de consultation des documents	79
4.19 Page de détails facture	80
4.20 Page de profile	81
4.21 Page des demandes de création	81
4.22 Page des demandes de modification	82

4.23 Page de consultation des clients	82
4.24 Page de modification des informations client	83
4.25 Bilan du projet en SonarQube	85
4.26 Capture d'un exemple de vulnérabilité	86
4.27 Structure fichier plat : bloc 1 et 2	88
4.28 Structure fichier plat : bloc 3	89
4.29 Structure fichier plat : bloc 4	90
4.30 Structure fichier plat : bloc 5 et 6	90
4.31 Le XSD DOC	91
4.32 Le XSD BRD	92
4.33 Le XSD CAI	93
4.34 Le XSD ECH	94
4.35 Le XSD TAXE	94

Table des matières

Introduction Générale	15
1 CONTEXTE GÉNÉRAL DU PROJET	17
Introduction	18
1.1 L'organisme d'accueil	18
1.1.1 Présentation de CGI	18
1.1.2 Services et les secteurs d'activités de CGI	19
1.1.3 Les clients de CGI par secteur	20
1.1.4 Les valeurs de CGI	21
1.1.5 CGI au Maroc	21
1.1.6 L'organigramme de CGI Rabat	22
1.1.7 Présentation du client MICHELIN	23
1.1.8 Le contrat CONVERGENCE	23
1.2 Périmètre du projet	24
1.3 Méthodologie de travail	25
1.3.1 Choix de la méthodologie	25
1.3.2 Choix de Scrum	26
1.3.3 Les piliers de SCRUM	26
1.3.4 Pilotage du projet avec SCRUM	27
1.3.5 Les rôles SCRUM	28
1.4 Planification du projet	29
1.4.1 Diagramme de GANTT	30
1.4.2 Gestion des risque	30
Conclusion	31

2 ANALYSE ET SPÉCIFICATION DES BESOINS	32
Introduction	33
2.1 Étude de l'existant	33
2.2 Problématique	34
2.3 Solutions proposées	34
2.4 Analyse des besoins	35
2.4.1 Identification des acteurs	35
2.4.2 Product Backlog :	36
2.4.3 Les besoins fonctionnels	37
2.4.4 Besoins non fonctionnels	37
Conclusion	38
3 CONCEPTION DU PROJET	39
Introduction	40
3.1 Présentation du langage de modélisation	40
3.1.1 Diagramme de cas d'utilisation	40
3.1.2 Diagrammes de séquence	48
3.1.3 Diagrammes de BPMN	51
3.1.4 Diagramme de classe	53
3.1.5 Modèle de données	54
3.1.6 Diagramme de déploiement	56
Conclusion	57
4 MISE EN PLACE DU PROJET	58
Introduction	59
4.1 Étude technique du projet	59
4.1.1 Présentation des technologies utilisés	59
4.1.2 Patrons de conception utilisés	62
4.1.3 Les frameworks adoptés	63
4.1.4 Les outils de développement utilisés	65
4.2 La réalisation du projet	68
4.2.1 Les programmes Batch	69

Table des matières	14
4.2.2 Présentation des interfaces de l'application	77
4.3 Les tests et validation	83
4.3.1 Les tests réalisés	83
4.3.2 La qualité du code	85
Conclusion	86
CONCLUSION GÉNÉRALE	87
ANNEXE	88
Bibliographie	95

Introduction Générale

Loin d'être un éphémère phénomène de mode, ou une tendance passagère, l'intégration des systèmes d'informations est fortement ressentie par l'ensemble des organisations. En effet, Il ne fait désormais plus aucun doute que la mise en place d'un système informatique améliore les performances, augmente le rendement des organismes et facilite la gestion du travail au sein de ces derniers.

Le client Michelin dispose d'une portail FAE qui est mis à contribution pour fournir les données de facturation, et il exprime clairement son besoin d'un outil pour stocker/gérer les informations fournies par FAE, mettre en forme et procéder à la distribution de ces données.

Consciente de l'importance d'un système informatisé, CGI m'a confié la construction d'un programme Batch, qui fait l'extraction des données à partir du système FAE et la mise en place et l'alimentation du portail D3C par les données déjà extraite afin de faciliter la lisibilité et lecture des factures par les clients de Michelin.

Dans cette optique, ma mission consiste principalement à appréhender le métier de CGI et bien analyser les besoins, avant de concevoir la solution à mettre en œuvre qui permettra de gérer les factures des clients.

Le premier travail couvrant la thématique de ce stage s'inscrit dans une politique de refonte générale du processus de gestion des factures. Plus particulièrement, elle vise l'extraction des données depuis FAE dans un fichier XML.

La seconde partie consiste à construire la portail D3C à travers laquelle les clients peuvent consulter leurs factures et les télécharger en format PDF à tout moment, la dernière partie est l'alimentation du portail par les données générés dans le fichier XML, choses qui étaient autrefois faites manuellement.

Le présent rapport est organisé en quatre chapitres :

- Chapitre 1 : consacré à la présentation du contexte général du projet, comportant une présentation de l'organisme d'accueil et le périmètre du projet.
- Chapitre 2 : aborde l'analyse et la spécification des besoins auxquelles doit répondre le projet.
- Chapitre 3 : consacré à la phase de conception du projet, incluant la modélisation du système avec UML.
- Chapitre 4 : consacré à la mise en place du projet qui présente les outils et les technologies du développement, ainsi que quelques interfaces de la solution et enfin les tests réalisés.

Chapitre 1

CONTEXTE GÉNÉRAL DU PROJET

Introduction

Ce chapitre présente le contexte général ainsi que les principaux objectifs du projet. Une présentation de l'organisme d'accueil sera en premier lieu, et ensuite une présentation du cadre général du projet tout en dévoilant la démarche de sa conduite.

1.1 L'organisme d'accueil

1.1.1 Présentation de CGI

Fondé en 1976 par Serge Godin et André Imbeau et établi à Montréal, CGI «Conseillers en gestion et informatique» est l'une des plus importantes entreprises qui offre une gamme complète de services-conseils en technologie de l'information(TI) et en management au monde.



FIGURE 1.1 – Logo CGI Technologies and solutions

Ils ont démarré l'entreprise dans la ville de Québec avec un seul client, un téléphone et beaucoup d'ambition. À partir de ce moment, ses revenus et sa taille n'ont cessé d'augmenter année après année. Actuellement, CGI compte approximativement 73000 membres travaillant dans le monde entier.

En 2012, CGI a réalisé sa plus grande acquisition à ce jour en se fusionnant avec Logica, une entreprise anglo-néerlandaise de services technologiques et d'entreprise. L'acquisition a accru sa présence, ses capacités et son expertise pour servir ses clients dans le monde entier. Aujourd'hui avec une présence dans 40 pays, une solide expertise dans tous ses



FIGURE 1.2 – Logo utilisé lors du rachat de Logica par CGI

marchés cibles et un éventail complet de services en technologies de l'information (TI), CGI est en mesure de répondre aux besoins d'affaires de ses clients partout, en tout temps.

1.1.2 Services et les secteurs d'activités de CGI

CGI offre un éventail complet de services, de solutions et d'expertise pour accélérer la transformation des activités de ses clients. La figure ci-dessous illustre les différents services du groupe CGI :



FIGURE 1.3 – Les services de CGI

CGI offre ses services complets dans des secteurs économiques cibles qui représentent plus de 90 des dépenses mondiales en TI. Ces secteurs sont les suivants :

- **Services financiers** : CGI aide les institutions financières, dont la majorité des plus grandes banques et des principaux assureurs, à réduire leurs coûts, à accroître leur efficacité et à améliorer leur service à la clientèle.
- **Gouvernements** : CGI offre du soutien à plus de 2 000 organisations gouvernementales pour réduire leurs coûts, améliorer leur efficacité ainsi que la qualité et la responsabilité des services publics tout en favorisant un engagement accru des citoyens.
- **Santé** : CGI aide plus de 1000 établissements de santé, hôpitaux et ministères de la santé, à mettre en œuvre des solutions TI pour améliorer les soins, les activités et les résultats.

- **Secteur manufacturier, distribution et détail** : CGI contribue à la transformation des activités de plus de 2 000 clients en améliorant leur efficacité et leur loyauté, en réduisant leurs coûts et en stimulant leur croissance durable.
- **Télécommunication et services public** : CGI aide 6 des 10 principaux fournisseurs mondiaux de services de télécommunications et 9 des 10 plus importants services publics en Europe à diversifier leurs sources de revenus, et à améliorer leur productivité et leurs services.
- **Pétrole et gaz** : CGI accompagne ses clients, dont 3 des 6 plus importantes entreprises pétrolières, à obtenir une valeur accrue de chacun des aspects de leur chaîne d'approvisionnement.

1.1.3 Les clients de CGI par secteur

Ci-dessous quelques clients et projets gérés par CGI notamment sur différents secteurs d'activité telle que le métier d'assurances, télécommunications, automobile, énergie et utilitaires, IT, et autres :



FIGURE 1.4 – Quelques clients de CGI par secteur

CGI a une couverture sectorielle large, travaillant sur des projets dans des domaines

extrêmement variés des finances à l'énergie en passant par les télécoms, elle est présente partout et permet à ses collaborateurs une spécialisation dans différents métiers.

1.1.4 Les valeurs de CGI

CGI a développé une culture d'entreprise fondée sur la participation à la vie de l'entreprise et sur l'attention qu'il faut accorder à chacun de ses membres.

Parmi les ingrédients clés d'une telle culture : la conception et le partage de valeurs communes.

À CGI, la croyance fondamentale est qu'une entreprise qui a un rêve inspirant, une grande intégrité, une philosophie de gestion humaine et des valeurs élevées augmente ses chances d'attirer des personnes de qualité et de répondre à leurs aspirations profondes.

CGI a souscrit à un certain nombre de principes ou d'idées maîtresses :

- Partenariat et qualité
- Objectivité et intégrité
- Entrepreneuriat et partage
- Respect
- Solidarité financière
- Responsabilité sociale

1.1.5 CGI au Maroc

CGI Maroc est le premier fournisseur francophone de services en TI offerts sur le continent (Nearshore) au Maroc, en collaboration avec plus de 35 clients actifs formés de grands noms français. Très actif dans le développement de l'IT et de l'innovation, la BU Maroc a démarré ses activités en 2004 et le lancement du business local été en 2009.

CGI Maroc possède actuellement trois bureaux travaillant conjointement à la fourniture de services aussi bien offshore que destinés au marché local. il possède aussi une double certification : ISO 9001 et ISO 27001 et fournit des prestations de services de plusieurs formats. Parmi ceux-ci nous trouvons principalement :

- BPO : Le Business Process Outsourcing, est l'externalisation d'une partie de l'activité de l'entreprise vers un prestataire extérieur, un sous-traitant.

- Intégration : Amélioration des performances, des organisations de premier plan qui comptent sur les services d'intégration de systèmes de CGI.
- TMA : La Tierce Maintenance Applicative est la maintenance appliquée à un logiciel, et assurée par une expertise externe dans le domaine des technologies de l'information et de la communication.

1.1.6 L'organigramme de CGI Rabat

Un organigramme est une représentation schématique des liens et des relations fonctionnels, organisationnels et hiérarchiques qui existent entre les éléments et les individus d'une organisation formelle (association, entreprise, réseau, etc.), d'un programme, etc. et met en évidence sa structure organisationnelle.

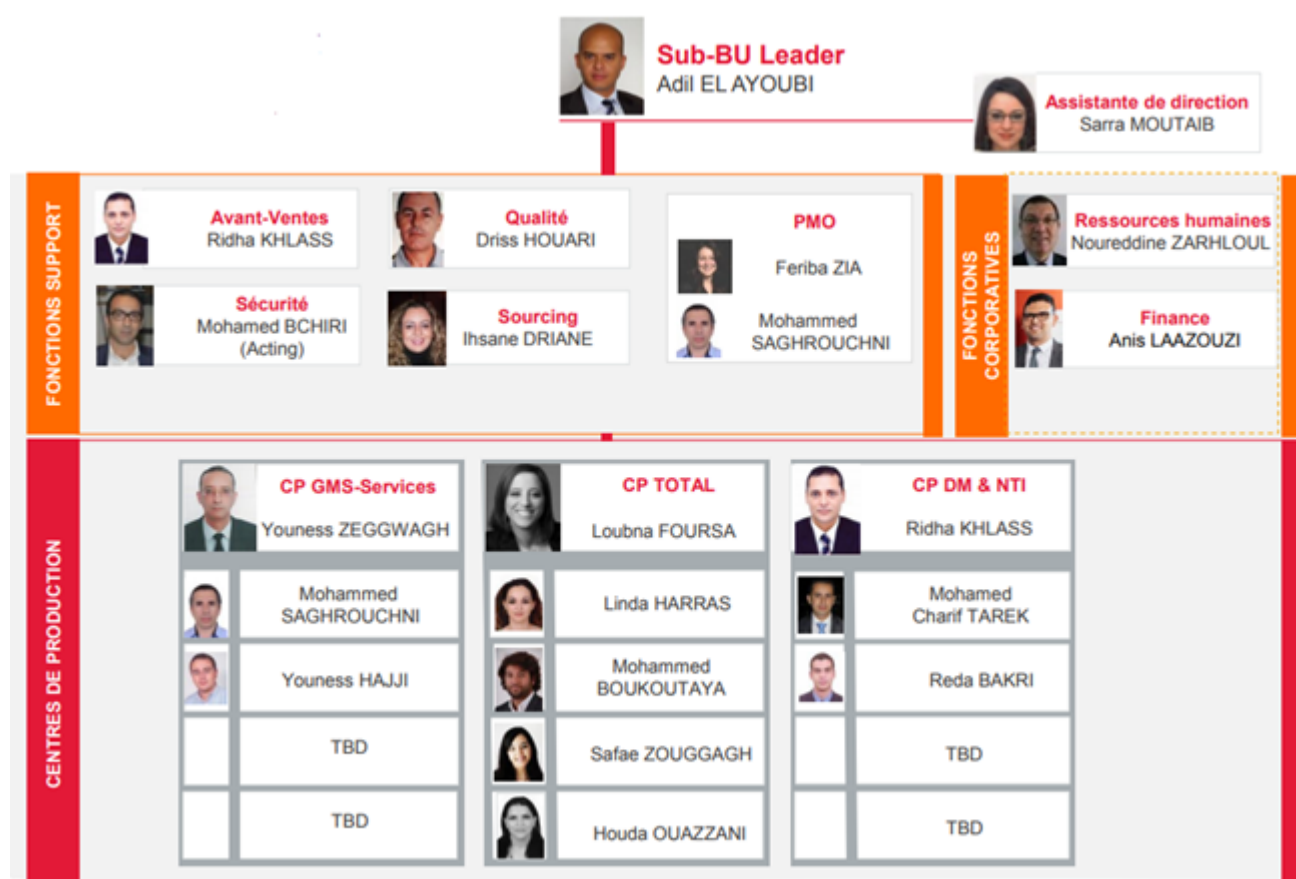


FIGURE 1.5 – Organigramme CGI Rabat

1.1.7 Présentation du client MICHELIN

Mon sujet de stage de fin d'études est une solution dont le client Michelin est le bénéficiaire, à savoir que j'effectue mon stage avec l'équipe « Michelin » au sein de CGI. Michelin est un fabricant français de pneumatiques qui conçoit et commercialise ses produits pour le marché de l'automobile, des camions, des deux roues, des avions, des engins civils et agricoles. Il a été fondé en 1889 par les frères André et Edouard Michelin près de la place des Carmes, là où est aujourd'hui implanté le siège social du groupe.



FIGURE 1.6 – Logo de l'entreprise Michelin

Michelin est implanté industriellement dans 18 pays et est présent commercialement dans plus de 170 pays. Il est spécialisé dans la fabrication et la commercialisation de pneumatiques. Il publie des cartes routières et des guides gastronomiques ou touristiques. Il propose aussi des services d'aide à la mobilité ViaMichelin sur des supports numériques. La vente de pneumatiques sous les marques Michelin, BFGoodrich, Kleber, Uniroyal et Taurus, représente plus de 80 de son chiffre d'affaires.

1.1.8 Le contrat CONVERGENCE

CGI a été choisie par Michelin à titre de partenaire privilégié pour le soutien et le développement de l'ensemble de ses applications en matière de chaîne d'approvisionnement, de marketing, de vente, et de finances d'entreprise, et ce, pour ses activités en Europe et en Amérique du Nord.



FIGURE 1.7 – Le Contrat Convergence entre CGI et Michelin

CGI soutiendra le programme de transformation de Michelin grâce à des solutions faciles

d'utilisation et mises rapidement à la disposition des utilisateurs.

L'objectif est de faciliter les tâches quotidiennes, tout en assurant la sécurité et l'intégrité des systèmes. Il s'agit notamment de faciliter l'optimisation du pilotage de la production de Michelin ainsi que d'accélérer les lancements de nouveaux produits.

En vertu de ce contrat, CGI interviendra sur des missions de services-conseils, de conception, de maintenance et de soutien des applications informatiques de Michelin en Europe et en Amérique du Nord. CGI apportera son savoir-faire en méthodologie agile et en intelligence d'affaires ainsi que ses compétences en technologies de l'information, par l'intermédiaire de ses équipes françaises et américaines situées à Clermont-Ferrand en France et à Greenville (Caroline du Sud), aux États-Unis.

1.2 Périmètre du projet

Les clients de Michelin sont partout dans le monde, répartis sur différents régions et pays. Faciliter les divers services pour chaque client et avoir un accès aux différents documents des clients, dans un seul et même portail, est l'objectif de ce projet.

Pour mener une analyse des besoins détaillée, une réunion était faite avec les utilisateurs concernés et nous avons pu déduire que certaines fonctionnalités restent nécessaires pour l'activité concernée.

La solution cible se compose de 4 éléments principales :

- Extraction des données et génération du fichier XML
- La gestion des documents
- La gestion des demandes (création/modification)
- La gestion des clients

En tant que client, l'application envisagée doit permettre :

- L'authentification
- La demande de création du compte
- La consultation des documents
- Le filtrage des documents selon des critères (avoir, facture, date ...)
- La consultation de ses propres profils

- La modification des ses propres informations
- Le téléchargement des documents

En cas de plantage d'une des chaînes de l'interface, une société serait absente du flux vers D3C ; Un mécanisme d'alerte doit être réfléchi, pour que D3C puisse alerter ses correspondants.

En tant que administrateur, il doit :

- S'authentifier
- Consulter la liste de tous les clients
- Approuver ou décliner les demandes de création des comptes
- Approuver ou décliner les demandes de modification des informations

1.3 Méthodologie de travail

Les méthodologies de travail vous aide à accomplir chaque étape de votre projet, de la planification à la mise en œuvre, dans un souci d'efficacité et rentabilité et afin de réussir votre projet en respectant les délais, le budget et les ressources.

Choisir une méthodologie pour conduire un projet permet à tous les acteurs de travailler efficacement ensemble, en suivant des règles clairement définies.

Voici 6 méthodes de gestion de projet à connaître.

- La méthode en cascade
- La méthode cycle en V
- Les méthodes agile
- La méthode PERT
- La méthode Prince2
- La méthode du chemin critique[Ref1]

1.3.1 Choix de la méthodologie

La société CGI a choisi les méthodes Agiles pour la gestion et le suivie de ses projets. Les méthodes Agiles caractérisent un mode de gestion des projets informatiques privilégiant le dialogue entre toutes les parties prenantes, clients, utilisateurs, développeurs

et autres professionnels du projet, la souplesse en cours de réalisation, la capacité à modifier les plans et la rapidité de livraison.

Notre équipe travaille selon la méthode SCRUM qui est considérée comme méthode agile. Les projets qui suivent la méthode agile SCRUM sont divisés en plusieurs cycles de travail relativement court que l'on appelle sprints. Ces derniers peuvent durer d'une à trois semaines, voire un mois, et permettent aux membres de l'équipe d'évaluer régulièrement les progrès liés au projet et de planifier les prochaines étapes de développement.

En effet, la méthode SCRUM se base avant tout sur un socle fixe de rôles, responsabilités et réunions qui ne changent jamais, tout en assurant une gestion flexible et adaptative des projets. Cela a l'avantage de rassurer les équipes lors de certaines phases de développement qui peuvent habituellement s'avérer chaotiques.

1.3.2 Choix de Scrum

Le choix de Scrum comme une méthodologie de pilotage pour ce projet s'est basé sur les atouts de ce dernier, qui peuvent être résumés comme suit :

- Scrum est centrée sur le produit.
- Progrès incrémental facilement mesurable et clairement visible.
- Les développeurs fixeront le rythme, ne sont pas surchargés de travail, et bénéficient d'un rôle accru.
- La charge du travail est réglable, basée sur la capacité de l'équipe et la priorité des tâches à réaliser.
- Les questions sont généralement découvertes avant qu'ils ne deviennent endémiques.
- Les tâches ont tendance à être granulaire, et par conséquent, plus facilement testables.
- La grande capacité d'adaptation au changement grâce à des itérations courtes.
- Flexibilité et créativité du travail avec facilité de transfert de connaissance.
- Probabilité du succès du projet est élevée.

1.3.3 Les piliers de SCRUM

Scrum repose sur 3 principaux piliers, qui se résument comme suit :

- **La transparence** : Scrum met l'accent sur le fait d'avoir un langage commun entre l'équipe et le management, qui doit permettre à tout observateur d'obtenir rapidement une bonne compréhension du projet.
- **L'inspection** : Scrum propose de faire le point sur les différents artefacts produits à intervalle régulier, afin de détecter toute variation indésirable.
- **L'adaptation** : Si une dérive est constatée pendant l'inspection, le processus doit alors être adapté. Scrum fournit des rituels, durant lesquels cette adaptation est possible. Il s'agit de sprint planning, de daily scrum, et de sprint review.

1.3.4 Pilotage du projet avec SCRUM

La première étape consiste à effectuer une première planification de l'itération ou Sprint Planning dans le jargon des développeurs. Cette réunion fera ressortir les éléments prioritaires de la liste des exigences fonctionnelles du produit. Chaque exigence représente une User Story ou "histoire utilisateur".

En accord avec le client, aussi appelé Product Owner, les premières livraisons devraient être effectuées à la fin de cette itération (qui dure de 2 à 4 semaines suivant le nombre des user stories présentes dans le Backlog). Le backlog est l'ensemble des US à développer durant l'itération en cours.

Une autre réunion appelée Revue de Sprint ou Sprint Review est organisée à la fin de chaque Sprint durant laquelle les développeurs présentent au client les fonctionnalités développées. Ce dernier pourra ainsi tout de suite donner son feedback, ce qui présente l'avantage de gagner beaucoup de temps et d'ajuster les fonctionnalités ou les méthodes de travail le cas échéant.

Vient ensuite une rétrospective de Sprint ou Sprint Rétrospective qui permet à tous les acteurs d'améliorer des choses et de s'améliorer également. Une autre particularité de la méthode Scrum est la réalisation de mêlées quotidiennes ou Daily Scrum qui permettent à l'équipe de développeurs de synchroniser leur travail. Cette réunion qui ne dure pas plus de 15 minutes permet à chacun de déterminer ce qu'ils ont réalisé depuis la dernière mêlée, de fixer ce qu'ils auront à terminer avant la prochaine Daily Scrum et d'identifier les obstacles qui pourraient les bloquer. Ci-après, une figure résumant le processus de Scrum.[Ref2]

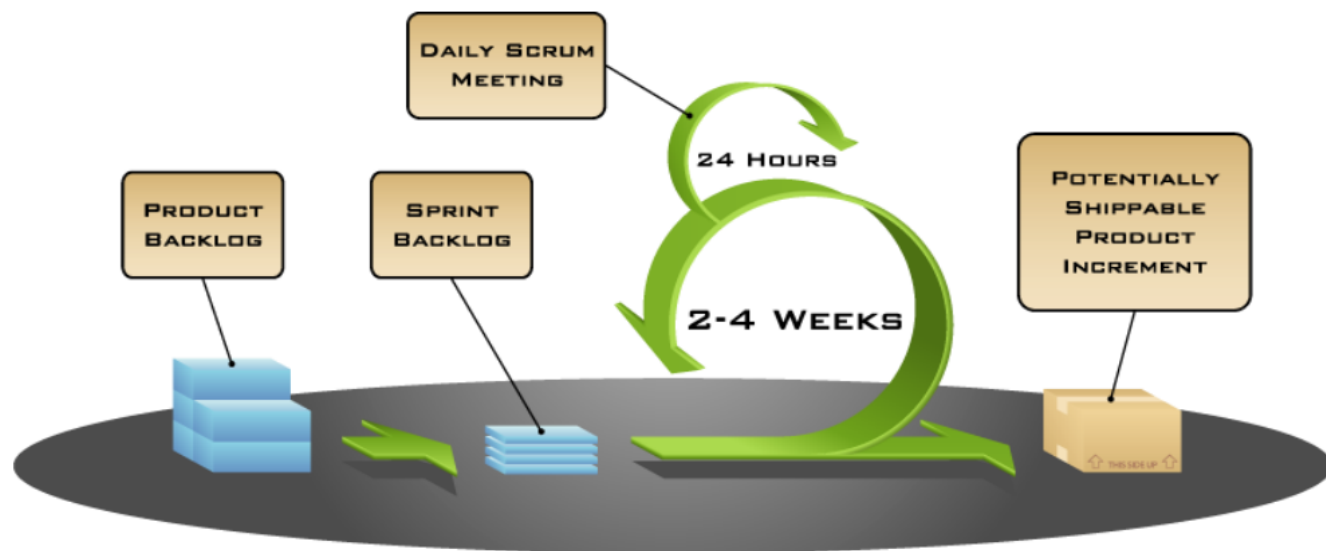


FIGURE 1.8 – Le processus de Scrum [Ref3]

1.3.5 Les rôles SCRUM

La méthode SCRUM définit trois rôles pour un projet, Dans la figure suivante, on voit donc déjà bien que des rôles précis sont attribués à chacun pour l'avancement du projet : chacun occupe un rôle spécifique, ce qui est l'une des meilleures briques de base pour avancer dans un projet.

- Le Product Owner qui porte la vision du produit à réaliser et travaille en interaction avec l'équipe de développement. Il s'agit généralement d'un expert du domaine métier du projet.
- L'Equipe de Développement qui est chargée de transformer les besoins exprimés par le Product Owner en fonctionnalités utilisables. Elle est pluridisciplinaire et peut donc encapsuler d'autres rôles tels que développeur, architecte logiciel, DBA, analyste fonctionnel, ingénieur système.
- Le Scrum Master qui doit maîtriser Scrum et s'assurer que ce dernier est correctement appliqué. Il a un rôle de coach à la fois auprès du Product Owner et auprès de l'équipe de développement. Il doit donc faire preuve de pédagogie. Il est également chargé de s'assurer que l'équipe de développement est pleinement productive. Généralement le Scrum Master est le chef de projet.

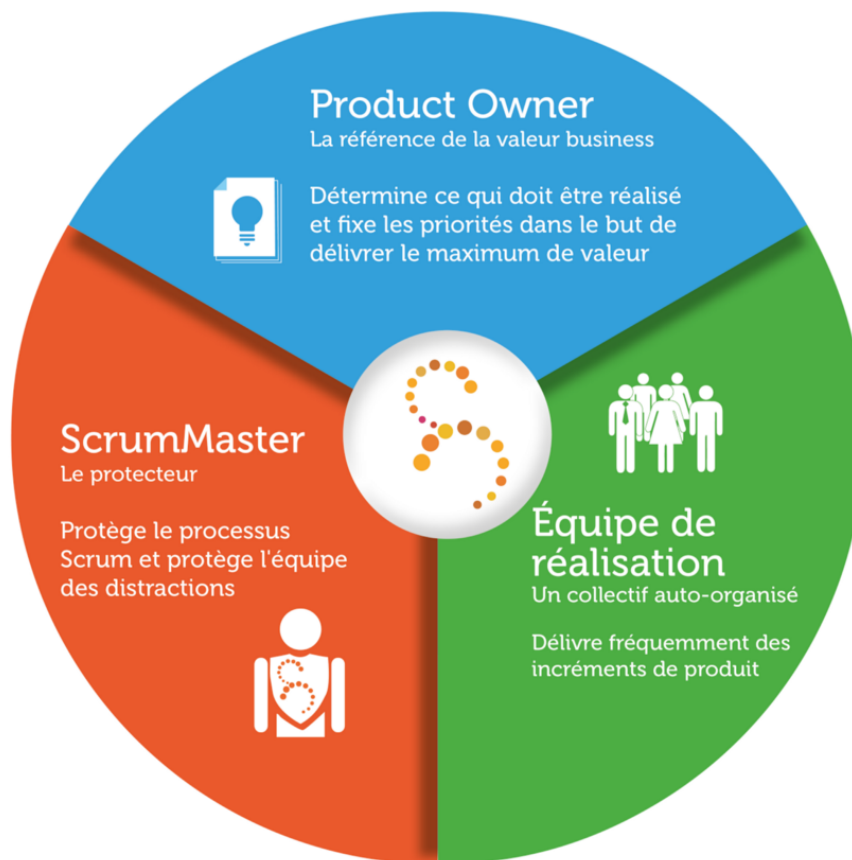


FIGURE 1.9 – Les rôles SCRUM [Ref4]

1.4 Planification du projet

La planification de projet vise à établir des prévisions raisonnables et à prévoir l'ordonancement des phases sur le plan des délais pour la mise en œuvre des travaux d'ingénierie et la gestion du projet. Ces prévisions sont indispensables à une gestion de projet efficace. Le planning élaboré se divise en quatre parties importantes :

1. **Formation** : Cette phase contient la formation en langage COBOL, ainsi que les formations en ligne (E-learning) et aussi les installations nécessaires pour le déroulement du stage.
2. **Montée en compétences** : Cette phase a permis de se familiariser avec l'environnement de travail et de maîtriser les différents outils et cela en travaillant sur des Workshops, résoudre des problématiques et aussi travailler sur des tâches qui ont

été attribuées par notre encadrant.

3. **Travail sur le projet** : Cette partie comprend quatre étapes :

- Étude de l'existant.
- Spécification des exigences.
- Conception du projet.
- Développement du projet.

4. **Rédaction du rapport** : Cette partie était intrinsèque aux autres parties.

1.4.1 Diagramme de GANTT

Parmi les outils de planification de projet, il ya le diagramme de GANTT, un outil qui permet de planifier un projet et rendre plus simple le suivi de son avancement. La figure suivante illustre l'enchaînement et la durée prévisionnelle des différentes activités et tâches du projet.

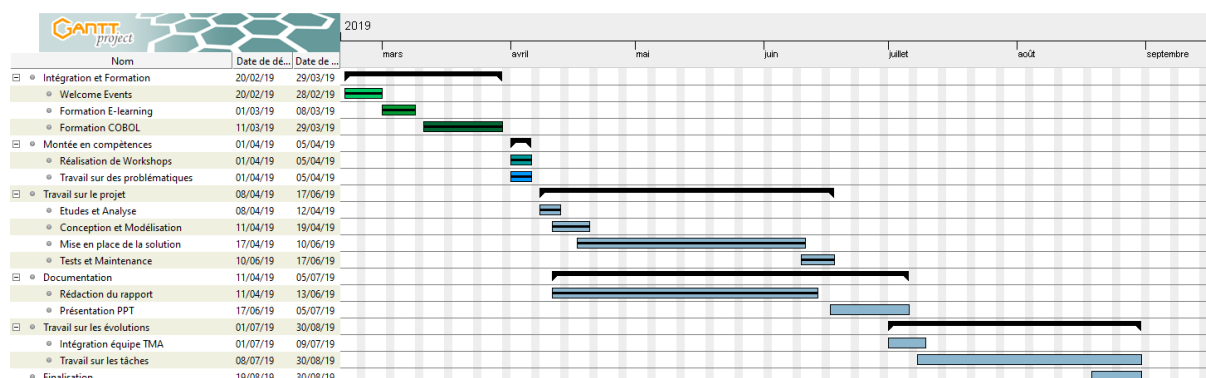


FIGURE 1.10 – Diagramme de GANTT

1.4.2 Gestion des risque

Le risque, c'est par définition un danger éventuel plus ou moins prévisible qui peut affecter l'issu d'un projet, la gestion des risques s'avère essentielle pour réagir en cas de survenance d'événements venant déstabiliser le projet.

Il est nécessaire de se poser la question des aléas qui peuvent survenir au cours de cette mission et des solutions qu'on y pourra apporter.

Durant les différentes étapes de réalisation de ce projet, nous étions face à plusieurs risques qui nous empêchaient, potentiellement, d'atteindre notre objectif. Nous avons pu les identifier et les réduire à travers des actions correctives et préventives.

Le diagramme d'Ishikawa ou 5M suivant permet d'analyser les grandes catégories de causes pour parvenir à un effet particulier. Les catégories de causes commencent toutes par la lettre M, ce qui permet de les mémoriser facilement [Ref5].

- Machines : il s'agit du matériel nécessaire au projet.
- Main-d'oeuvre : le personnel qui participe au projet.
- Méthodes : les procédures existantes, les modes d'emploi utilisés.
- Matières : tout ce qui est consommable et utile au projet ou à l'objectif du projet.
- Milieu : l'environnement physique et humain pouvant influencer sur le projet.

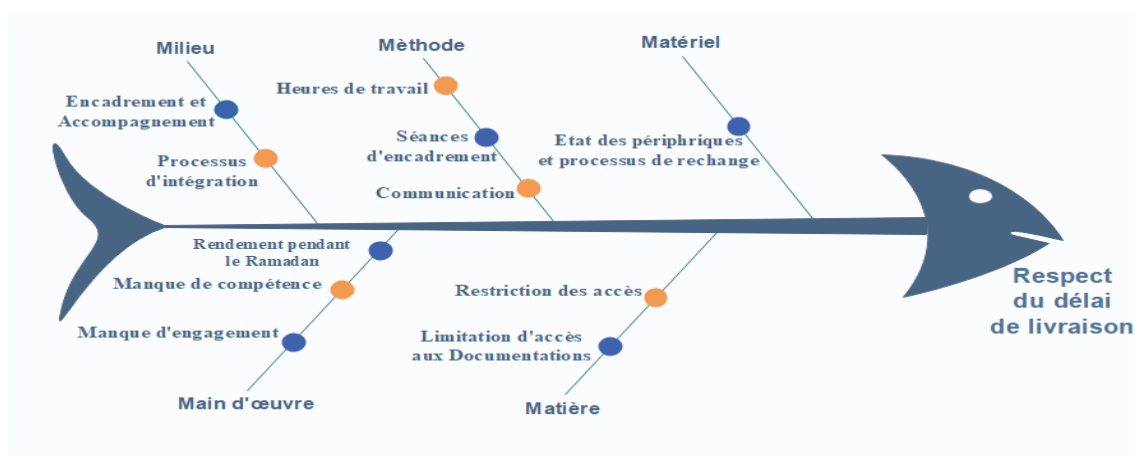


FIGURE 1.11 – Diagramme d'ishikawa

Conclusion

Ce premier chapitre a présenté l'organisme d'accueil, CGI Technologies et Solutions ainsi que son domaine d'activité en plus une présentation sur le client Michelin. Ensuite, il a situé le projet dans son contexte générale et détaillé l'organisation et la méthodologie de travail adoptée par les équipes de CGI et la planification suivie pour mener à bien le projet. Le 2ème chapitre va aborder l'étude fonctionnelle de notre système.

Chapitre 2

ANALYSE ET SPÉCIFICATION DES BESOINS

Introduction

La mise en place d'une solution nécessite d'abord une étude de l'existant et une étude des besoins fonctionnels et techniques des utilisateurs.

D'abord on va commencer par dégager les besoins fonctionnels et techniques, et ensuite une analyse générale des problématiques sera effectuée.

2.1 Étude de l'existant

Avant de mettre en place notre solution, le client Michelin gérait ses activités par des applications qui génèrent des factures sous forme d'un fichier Excel.

Le schéma ci-dessous nous montre la gestion des activités Michelin pour le périmètre de facturation :

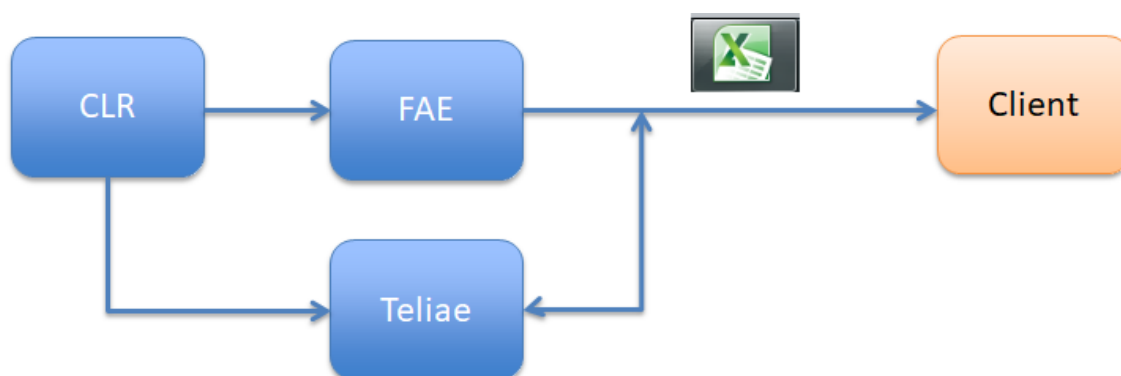


FIGURE 2.1 – La gestion des activités Michelin pour le périmètre de facturation

CLR c'est le système dans lequel les clients de Michelin introduisent leurs commandes, il envoie des flux BL vers l'application de facturation FAE, et des données vers TELIAE. FAE par la suite envoie un flux Excel directement au client, sur la base des informations des factures. Ce flux d'envoi est mixé avec Teliae pour récupérer le numéro du Delivery Detail Document (D3). Après le client reçoit par email le fichier Excel contenant ses documents.

2.2 Problématique

D'après l'étude de l'existant on constate qu'il n'existe pas un outils sécurisé qui regroupe l'ensemble des documents faites par le client ou un moyen rapide pour distinguer entre les factures et les avoirs.

Par conséquence ils trouvent des difficultés pour accéder ou visualiser leurs documents ils doivent lire tout le document pour savoir s'il s'agit d'une facture ou avoir. En plus la lecture de toutes les factures/avoirs sous formes d'un fichier Excel engendre un coût considérable en termes de temps.

Concernant l'outil Excel, on sait très bien qu'il est idéale pour créer des tableaux, des graphiques et appuyer des présentations PowerPoint ou PDF mais il faut bien admettre qu'il comporte plusieurs désavantages, on cite parmi eux :

- Les fichiers sont très volumineux ce qui pose des problèmes lors de l'envoi des fichiers aux clients.
- Absence de sécurité des données.
- Manque de lisibilité

2.3 Solutions proposées

Dans le but d'avoir un récapitulatif des documents et d'améliorer la lisibilité des factures émises par FAE, et au vue des limites du système existant, toutes les raisons décrit précédemment ont impliqués la mise en place d'un portail D3C et l'alimentation par les données en provenance de FAE afin de faciliter la lecture des factures par les clients et aussi sécuriser les données.

Après avoir reçu le BL par FAE, il envoie les informations des factures dans un fichier plat, et après sa mise en forme, le parsing des données au format XML est fait, ensuite le flux XML est envoyé vers D3C. Ce flux d'envoi est mixé avec Teliae pour récupérer le numéro du Delivery Detail Document (D3). Et enfin toutes les documents seront disponible dans D3C auprès du client.

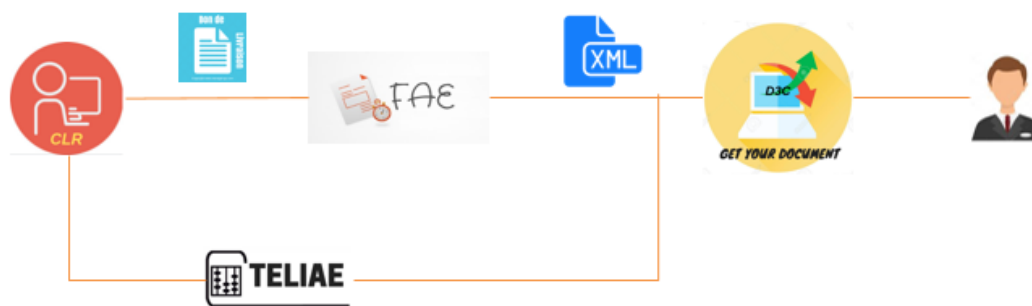


FIGURE 2.2 – La solution proposée pour la gestion des activités Michelin

2.4 Analyse des besoins

Cette étape consiste à analyser et décortiquer les besoins pour en tirer une représentation en diagrammes formalisés.

2.4.1 Identification des acteurs

La capture des besoins fonctionnels, une des phases primordiales de la méthode Scrum qui consiste essentiellement en l'identification des acteurs qui vont interagir avec le système. Au début de chaque sprint de Scrum, on définit quelles seront les fonctionnalités à développer durant ce sprint, on commence toujours par identifier les acteurs pour pouvoir identifier leurs besoins.

Les acteurs principaux du système sont :

- **Les clients** : sont les clients de Michelin, les utilisateurs de l'application peuvent uniquement utiliser les fonctionnalités de l'application après l'authentification. Cela signifie que les utilisateurs doivent dans un premier temps faire une demande de création de compte pour obtenir leur identifiant et mot de passe à travers l'administrateur.
- **L'administrateur** : c'est le client Michelin, il a le contrôle total sur l'application. Uniquement l'administrateur peut voir toutes les informations de chaque utilisateur et peut éditer dans toutes ces dernières.
L'administrateur a la possibilité de créer de nouveaux utilisateurs, d'approuver ou décliner leurs demandes ainsi que d'attribuer des mots de passes.

2.4.2 Product Backlog :

L'approche SCRUM propose de commencer d'abord par lister les exigences du client afin de produire le Product Backlog sous forme de liste d'item ou User Story.

Cette liste contient tout ce qui pourrait être requis dans le produit et est unique source des besoins pour tous les changements à effectuer sur le produit.

La figure suivante résume le backlog produit du projet.

Les numéros mentionnés dans chaque tâche indiquent une estimation en jours pour la réalisation de cette dernière, tandis que les lettres : M, S et C proviennent de l'acronyme de la méthode MoSCoW qui aide à décider les priorités des tâches, à savoir que M signifie Must have, S pour Should have, C pour Could have et W pour Would have.

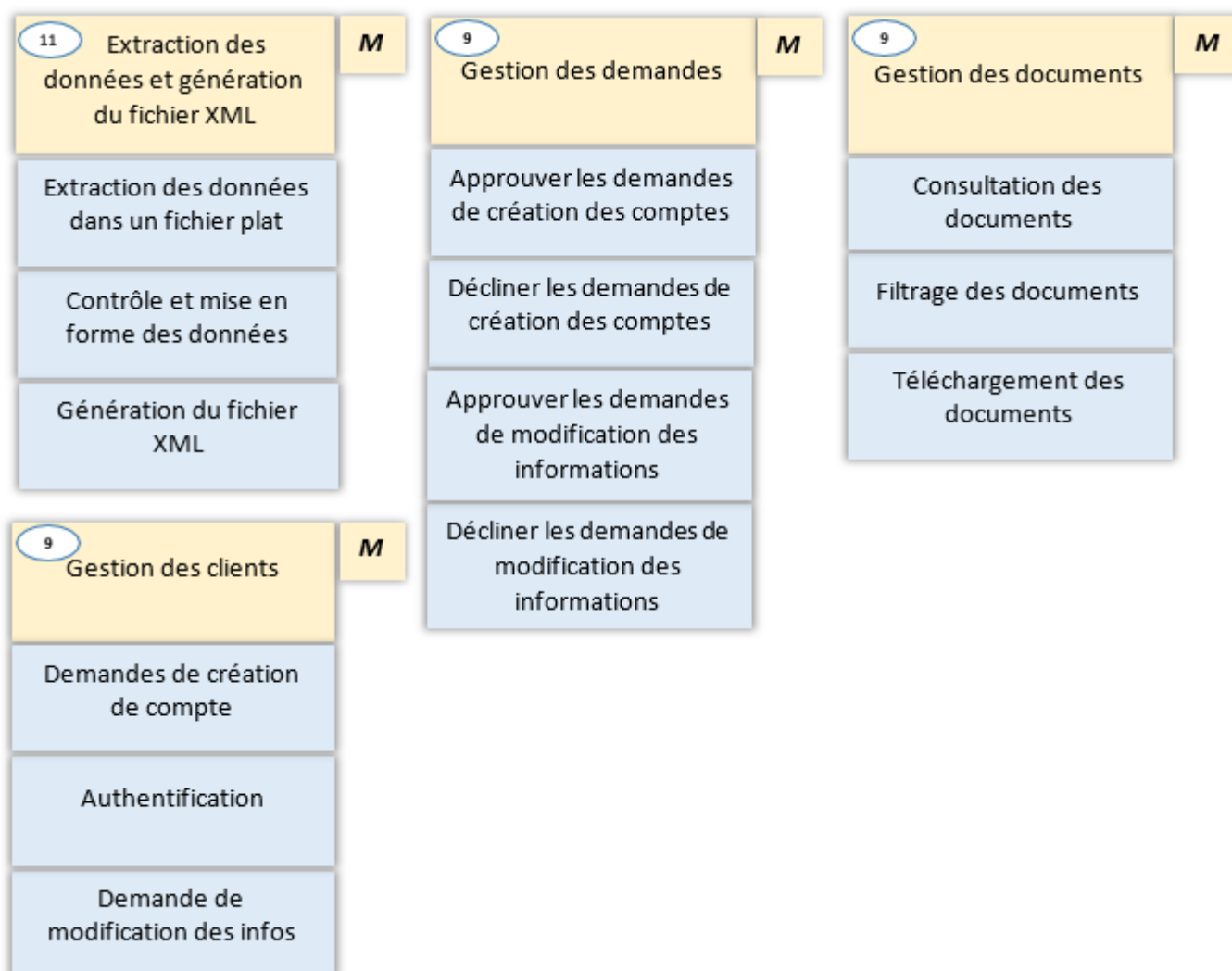


FIGURE 2.3 – Le Product Backlog

2.4.3 Les besoins fonctionnels

Extraction des données de FAE : En effet, notre client a exprimé son besoin d'extraire tous les données des factures nécessaire en provenance du FAE et les mettre dans un fichier plat pour organiser les données selon une structure spécifique .

Génération du fichier XML : Les données contenant dans le fichier plat sont alors générés dans un fichier XML après la vérification du champs "DateEmissionASF" qui doit être null, ce fichier sera donc un point de pont entre les deux différents environnements.

Alimentation D3C : Le portail réalisé dispose de son propre base de donnée, qui sera alimenté par les données générés dans le fichier XML, c'est pour ça le système PELICAN s'occupe du fichier XML et fait le chargement de ces données dans D3C.

Gestion des clients : Avant toute chose, le client doit exprimer son besoin de créer un compte pour accéder au portail, et après l'approbation de l'admin, il a le droit donc de s'authentifier et par la suite modifier ses informations.

Gestion des demandes : l'administrateur de D3C, peut gérer les demandes de création des comptes et les demandes de modification des informations faites par les clients. En cas d'approbation des demandes déclenche automatiquement l'envoi des notifications au client lui informe que sa demande est traité avec succès.

Gestion des documents : dans D3C, les clients dispose d'un espace où ils peuvent consulter leurs documents que ce soit facture ou avoir, ils peuvent aussi les télécharger.

2.4.4 Besoins non fonctionnels

En plus de la testabilité et le fonctionnement avec un optimum de temps, l'application doit garantir les besoins suivants :

- **La sécurité :** La sécurité de l'application s'appuie sur l'authentification des utilisateurs et la non répudiation, à savoir que la 1ère repose sur la vérification du login/mot de passe et la 2ème assure qu'un utilisateur ne peut pas nier avoir effectué une tâche au niveau de l'application.

- **La navigabilité** : l'application doit disposer d'un menu simple qui peut servir à l'utilisateur pour mieux interagir avec l'application.
- **La clarté** : le code doit être clair pour permettre des futures évolutions ou améliorations.
- **L'intégrité** : l'application doit garantir l'intégrité et la cohérence des données à chaque mise à jour et à chaque insertion.

Conclusion

Ce chapitre a permis de faire une analyse de l'existant pour détecter les problématiques engendrées. Ensuite, les exigences fonctionnelles et techniques des utilisateurs clés de la solution ont été définies. Afin d'y répondre, une conception de la solution s'avère nécessaire, et c'est ce qui fera l'objet du chapitre suivant.

Chapitre 3

CONCEPTION DU PROJET

Introduction

La phase de conception vient répondre à la question «comment réaliser la solution ?». Elle se fait suite à la capture et l'analyse des exigences fonctionnelles et techniques. Dans le présent rapport, cette phase de conception s'articule autour des différents diagrammes UML élaborés afin de modéliser le système à mettre en place.

3.1 Présentation du langage de modélisation

Le langage UML (Unified Modeling Language ou langage de modélisation unifié) est destiné à l'architecture, la conception et la mise en œuvre de systèmes logiciels complexes par leur structure aussi bien que leur comportement. Il a été pensé pour être un langage de modélisation visuelle commun, et riche sémantiquement et syntaxiquement. L'UML a des applications qui vont au-delà du développement logiciel, notamment pour les flux de processus dans l'industrie [Ref6].

3.1.1 Diagramme de cas d'utilisation

Un diagramme de cas d'utilisation [Ref6] est un graphe d'acteurs, un ensemble de fonctionnalités englobés par la limite du système, des relations (ou associations) de communication (participation) entre les acteurs et les cas d'utilisation, et des généralisations de ces cas d'utilisation.

Un acteur : est un ensemble cohérent de rôles joués par des entités externes (utilisateur, dispositif matériel ou autre système) qui interagissent avec le système.

Un acteur est représenté par l'icône stéréotype standard d'homme bâton (stick man) avec dessous le nom de l'acteur, ou par une classe avec le mot-clé actor.

Un cas d'utilisation : représente une unité cohérente d'une fonctionnalité fournie par un système (ou sous-système ou classe) spécifiée par une séquence d'actions que le système peut exécuter en interagissant avec les acteurs du système.

Un cas d'utilisation est représenté par une ellipse avec dedans le nom du cas d'utilisation, et éventuellement un stéréotype, des attributs, des opérations, et des points d'extension.

Le diagramme de cas d'utilisation global de notre solution est présenté ci-dessous, les acteurs sont représentés par le bonhomme à gauche. Nous pouvons bien remarquer qu'il y a 3 acteurs principales qui peuvent effectuer des actions spécifiques et bien définies dans le système, ces acteurs sont l'administrateur du système(notre client Michelin), les clients qui sont les clients de Michelin et le système générateur.

Chaque acteur peut effectuer des actions spécifiques et bien définies dans le système. Nous remarquons que tous les actions dans ce système ne peuvent être effectuées qu'après authentification ce qui signifie l'aspect sécurité déjà mentionné auparavant.

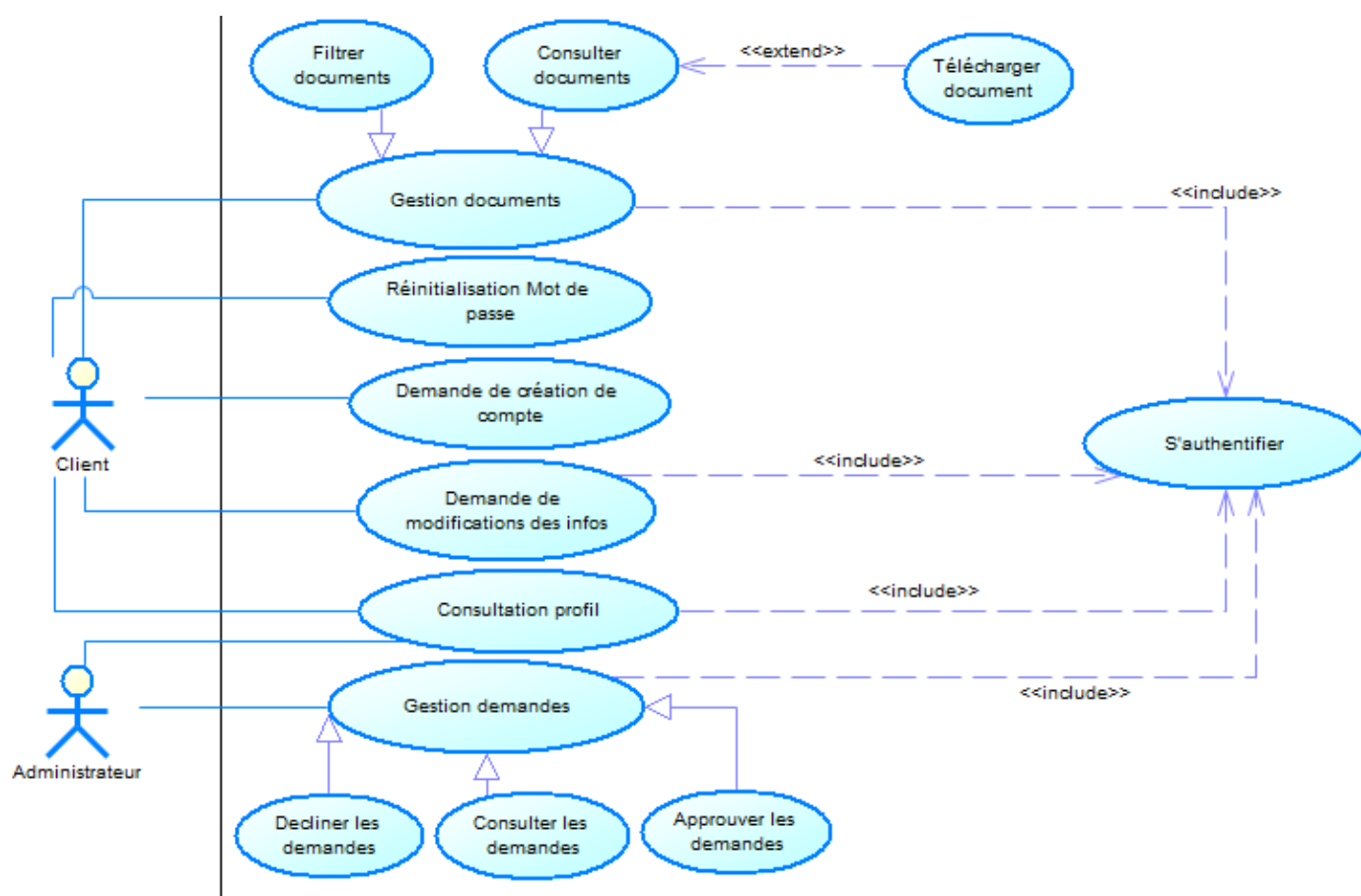


FIGURE 3.1 – Diagramme de cas d'utilisation global

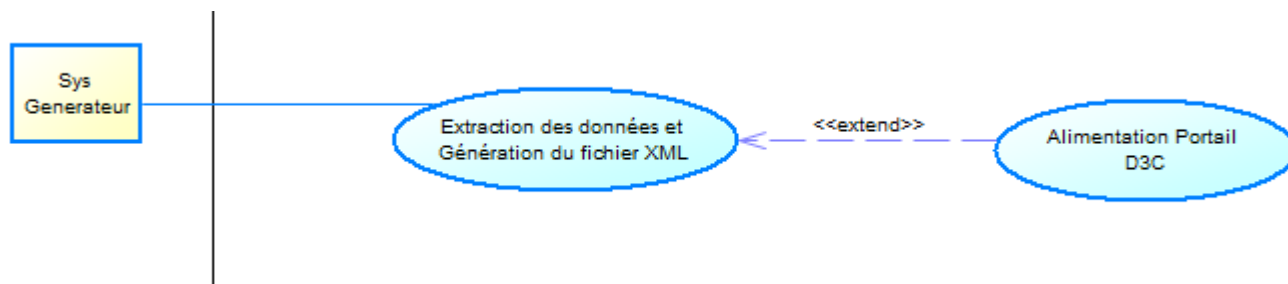


FIGURE 3.2 – Diagramme de cas d'utilisation du système

Description des cas d'utilisation :

- Cas d'utilisation : Gestion documents

Nom	Gestion documents
Acteur(s)	Client
Description	La gestion des documents se fait par les clients, il inclut la consultation, filtrage et téléchargement du document.
Pré-condition	L'authentification du client
Post-condition	Le téléchargement du document
Scénario nominal	<ol style="list-style-type: none"> 1. Le client choisit l'option consulter les documents 2. Le système affiche l'ensemble des documents effectués 3. Le client choisit une facture/avoir spécifique 4. Le système affiche le document choisie 5. Le client choisir de télécharger le document 6. Le système affiche un message de confirmation
Scénario d'exception	2.a Le client veut filtrer les documents selon un critère : fait appel au cas d'utilisation interne «Filtrer les documents».

TABLE 3.1 – Cas d'utilisation : Gestion documents

Prototypage des interfaces : Cette technique consiste à préparer quelques interfaces graphiques de l'application en utilisant un outil de conception des

prototypes afin de mesurer le degré de satisfaction du client par rapport à la compréhension du projet.

L'interaction qui se produit entre l'utilisateur final et le développeur, à la suite de la discussion sur ces interfaces, permet d'ajuster les besoins et de les concevoir de manière précise et exacte. En effet, les interfaces graphiques font que l'utilisateur final soit plus interactif, précis et le pousse à mieux s'exprimer.

Suivant le scénario nominal, la figure suivante présente un exemple de prototype de la gestion des documents.

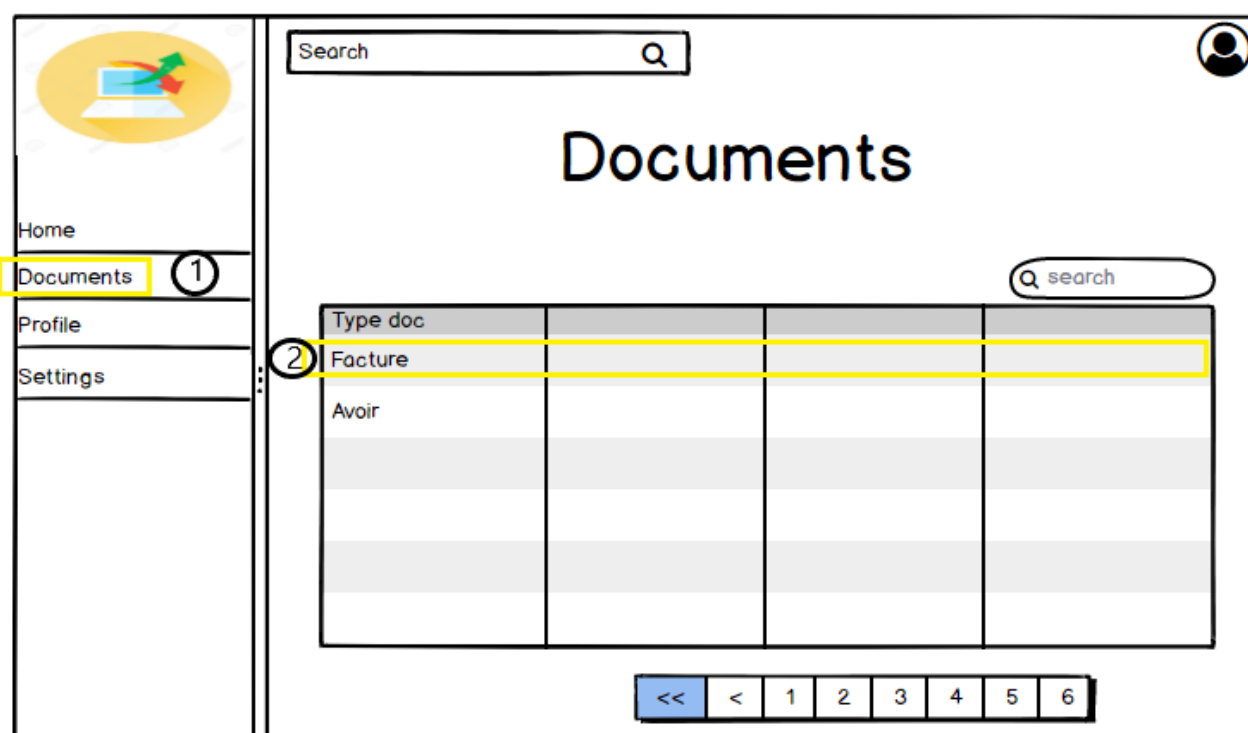


FIGURE 3.3 – Prototype d'interface consulter document

D'après la figure ci-dessus on voit que notre client exprime son besoin de façon que dès l'authentification de l'utilisateur client, il peut alors aller consulter tout ses documents les classer alpha numériquement pour chaque colonnes ainsi que chercher directement et avoir le résultat.

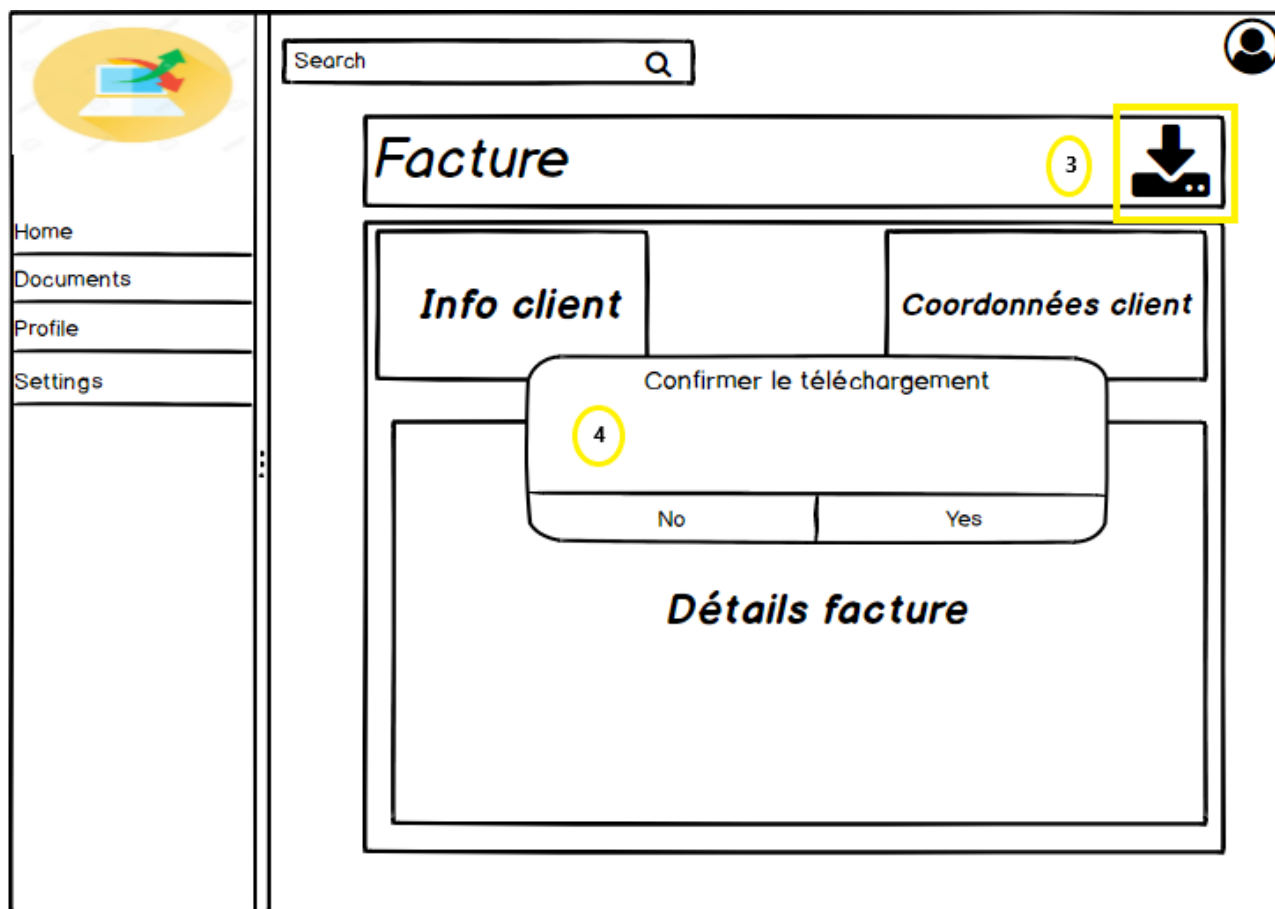


FIGURE 3.4 – Prototype d'interface télécharger document

La deuxième maquette traduit le besoin de récupérer les informations souhaités en format papier électronique, cela sera effectué à travers un bouton en haut qui donne la main à télécharger le document cherché auparavant, ainsi qu'une bulle de confirmation apparaît pour confirmer notre choix de téléchargement. Les informations de clients s'organisent de façon que à gauche on trouve ses informations, et à droite ses coordonnées et en corps de facture les détails de cette facture.

- **Cas d'utilisation : Gestion demandes**

Nom	Gestion demandes
Acteur(s)	Administrateur
Description	La gestion des demandes se fait par l'administrateur, il peut voir l'ensemble des demandes de création ou de modification faites par les clients et il a le choix soit d'approuver ou de décliner la demande.
Pré-condition	L'authentification de l'administrateur
Post-condition	l'envoi d'une notification au client
Scénario nominal	<ol style="list-style-type: none"> 1. L'administrateur choisit l'option consulter les demandes 2. Le système affiche l'ensemble des demandes effectués 3. L'administrateur choisit une demande spécifique 4. Le système affiche la demande choisie 5. L'administrateur approuve ou décline la demande
Scénario d'exception	2.a S'il n'y a pas de demande : le système affiche "il y a pas de nouvelle demande"

TABLE 3.2 – Cas d'utilisation : Gestion demandes

Prototypage des interfaces : Suivant le scénario nominal, la figure suivante présente un exemple de prototype de la gestion des demandes.

Pour la gestion des demandes on a présenté des maquettes création et modification, l'administrateur doit avoir la possibilité de consulter les deux types de demandes et naviguer entre eux , puis classer alphabétiquement les champs des noms , prénoms etc..

chaque ligne doit contenir un champs Action qui a deux bottons pour approuver et décliner comme on voit dans la maquette ci dessous , une autre solution existe en cas de clique sur un client pour approuver plus de détails.

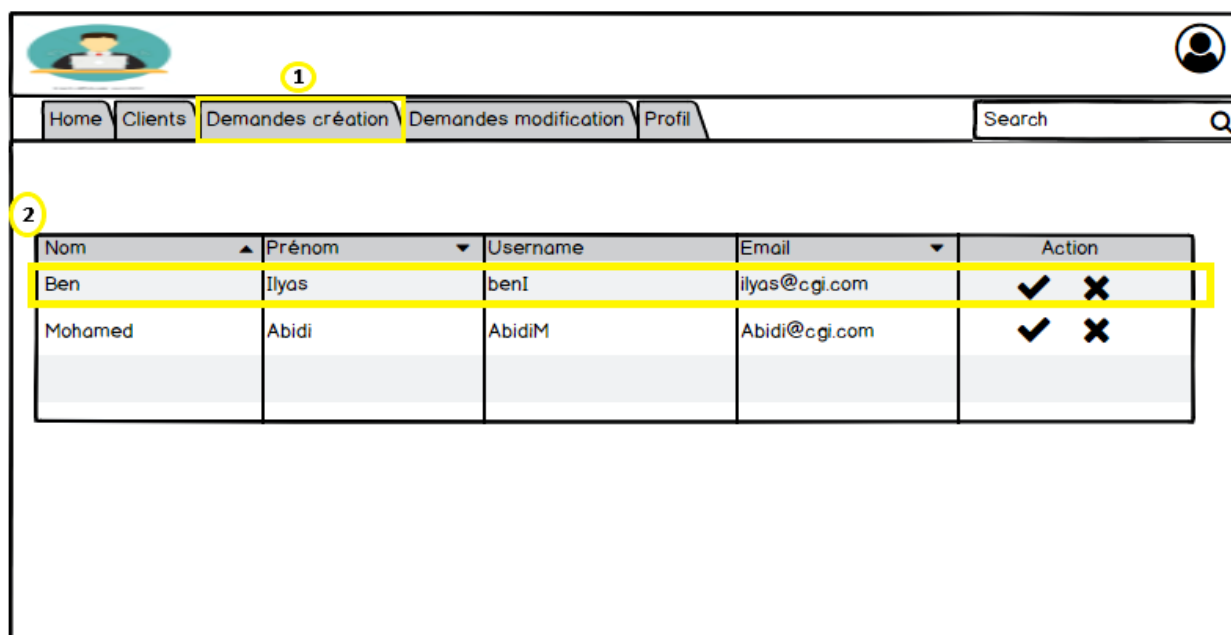


FIGURE 3.5 – Prototype d'interface consulter demandes

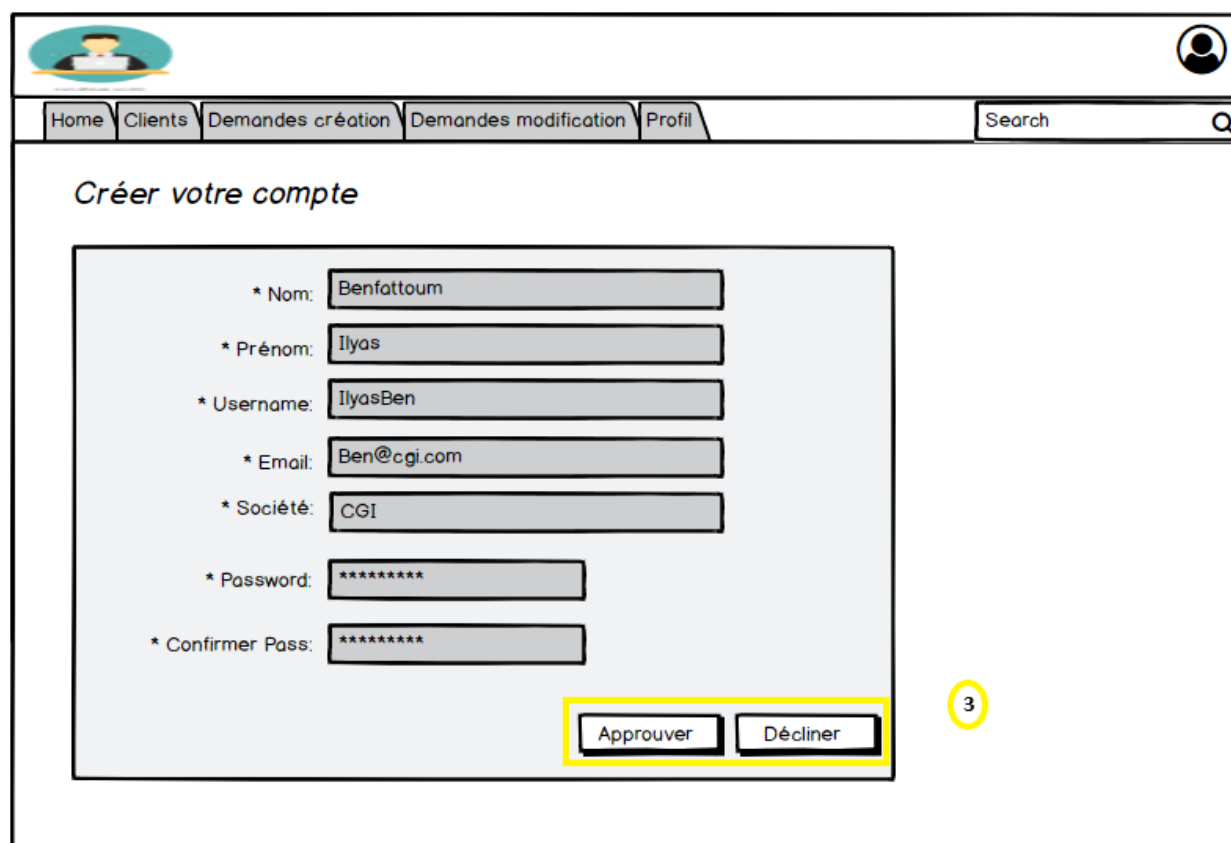


FIGURE 3.6 – Prototype d'interface approuver demande

- **Cas d'utilisation : Demande de création de compte**

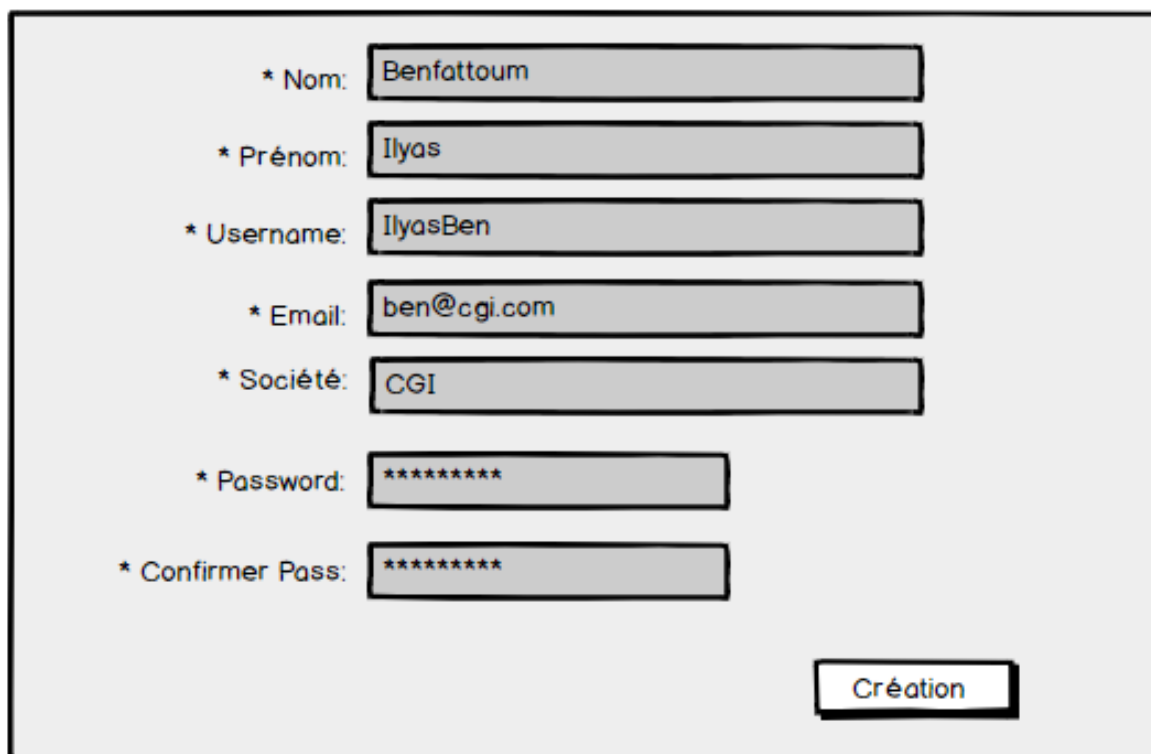
Nom	Demande de création de compte
Acteur(s)	Client
Description	La demande de création de compte se fait par les clients qui ne possèdent pas de compte et ils doivent attendre l'approbation de l'administrateur pour qu'ils puisse accéder à leurs compte.
Pré-condition	Le client ne doit pas avoir déjà un compte.
Scénario nominal	<ol style="list-style-type: none"> 1. Le client accède au portail 2. Le client demande la création du compte 3. Le système affiche le formulaire à remplir pour la création 4. Le client saisit ses informations et les envoie à l'administrateur 5. Le système affiche un message de confirmation
Scénario d'exception	4.a Un champ obligatoire est vide : le système affiche un message d'erreur indiquant que le champ est obligatoire

TABLE 3.3 – Cas d'utilisation : Demande de création de compte

Prototypage des interfaces : Suivant le scénario nominal, la figure suivante présente un exemple de prototype de la demande de création de compte.

Dans cette figure on trouve une maquette qui contient un formulaire ayant toutes les informations nécessaire pour créer un compte , tout client qui souhaitera créer une demande de création il devra alors passer par cette interface , et cette même demande qu'on a parlé à propos lors de présentations de maquettes précédente de gestion des demandes laquelle nécessite l'intervention de l'administrateur.

Créer votre compte



Prototype d'interface de création de compte. Le formulaire est intitulé "Créer votre compte" et contient les champs suivants :

- * Nom: Benfattoum
- * Prénom: Ilyas
- * Username: IlyasBen
- * Email: ben@cgi.com
- * Société: CGI
- * Password: *****
- * Confirmer Pass: *****

Un bouton "Création" est visible en bas à droite.

FIGURE 3.7 – Prototype d'interface demande de création de compte

3.1.2 Diagrammes de séquence

Un diagramme de séquence est un document graphique qui montre pour des scénarios de cas d'utilisation précis, les événements générés et les interactions entre objets en se basant sur des messages ordonnés. Chaque message transitant sur un lien est symbolisé par une flèche porteuse d'une expression. La lecture se fait de haut en bas, et l'ordre chronologique doit respecter ce sens[Ref7].

Voilà en ce qui suit la présentation de quelques diagrammes de séquence pour 3 sprints parmi ceux élaborés lors des phases de conception du projet.

3.1.2.1 Diagrammes de séquence du scénario "Extraction des données et génération XML"

La figure suivante présente le diagramme de séquence du scénario "Extraction des données et génération XML" qui fait parti du premier Sprint. Ce dernier met en évidence l'ensemble des interactions entre le système générateur et le système FAE.

En effet, après l'extraction des données des factures du FAE, on les met dans un fichier plat et on vérifie ensuite si la DateEmissionASF est null puis on génère le fichier XML.

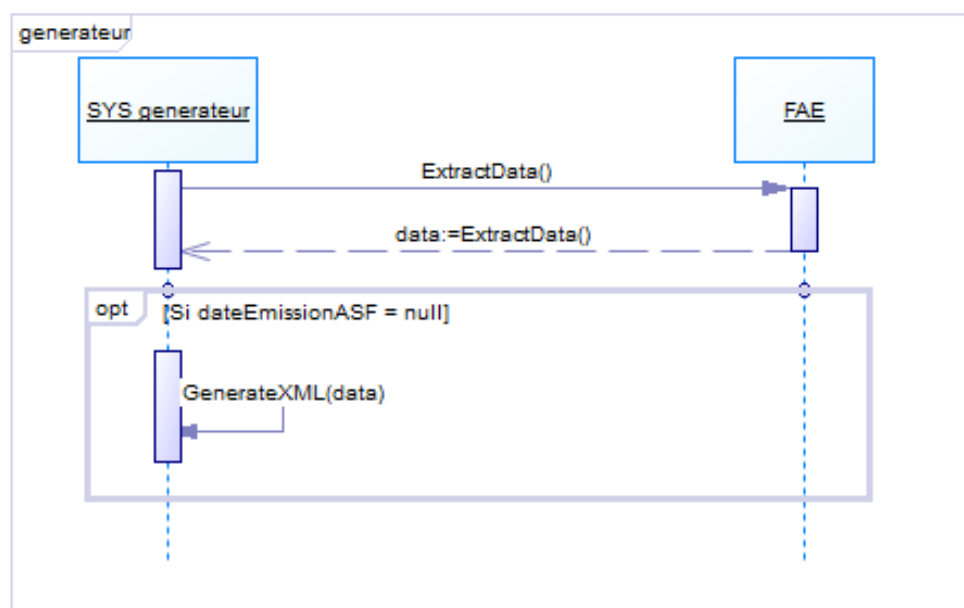


FIGURE 3.8 – Le diagramme de séquence du scénario "Extraction des données et génération XML"

3.1.2.2 Diagrammes de séquence du scénario "Gestion demandes"

La figure suivante présente le diagramme de séquence du scénario "Gestion demandes" qui se situe dans le 2ème Sprint. Ce dernier met en évidence l'ensemble des interactions entre l'administrateur d'une part et le portail D3C d'autre part.

L'administrateur dans un premier temps demande de consulter toutes les demandes soit de création de compte ou de modification des informations, après que le système lui affiche ses demandes, il choisit une qui va la traiter et par la suite il vérifie les informations s'ils sont juste ou non et puis il approuve ou décline.

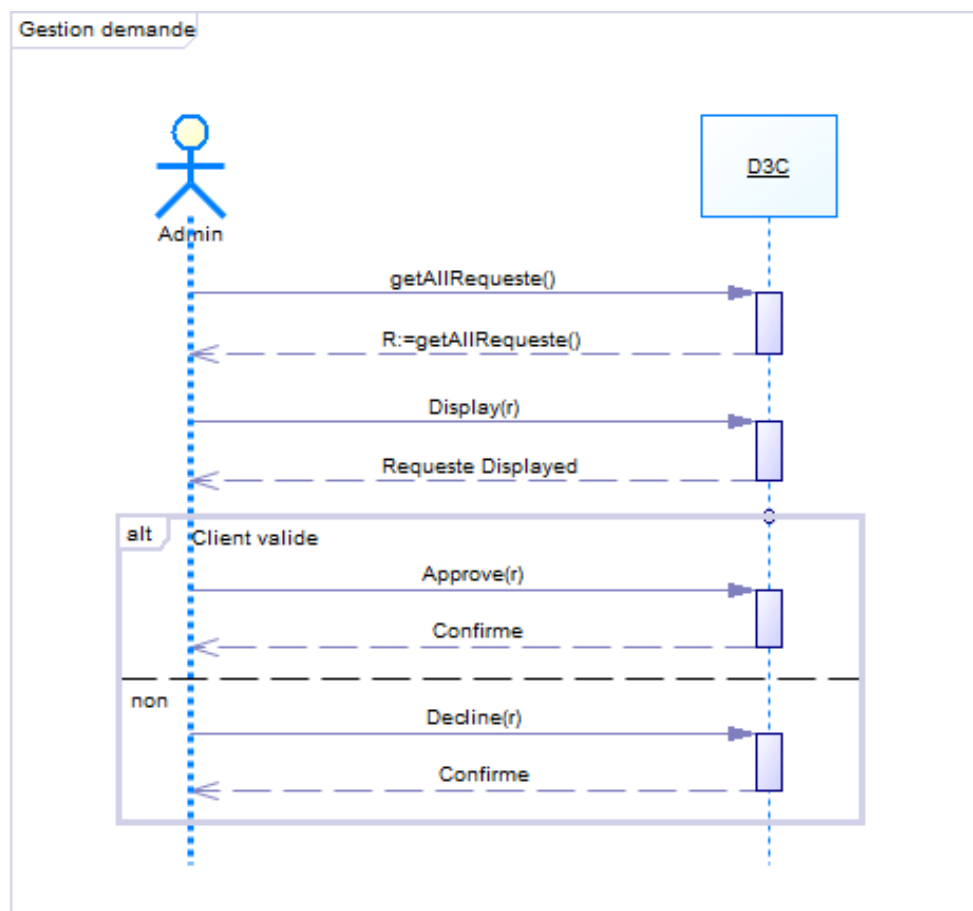


FIGURE 3.9 – Le diagramme de séquence du scénario "Gestion demandes"

3.1.2.3 Diagrammes de séquence du scénario "Gestion documents"

La figure suivante présente le diagramme de séquence du scénario "Gestion documents" qui se situe dans le 3ème Sprint. Ce dernier met en évidence l'ensemble des interactions entre le client d'une part et le portail D3C d'autre part.

Le client dans un premier temps demande de consulter toutes ses documents, les factures et les avoirs, après que le système lui affiche ses documents, il peut faire une recherche sur une facture précise ou de choisir une qui va la traiter et par la suite il vérifier les informations contenu dans cette facture s'ils sont juste ou non et puis il peut la télécharger.

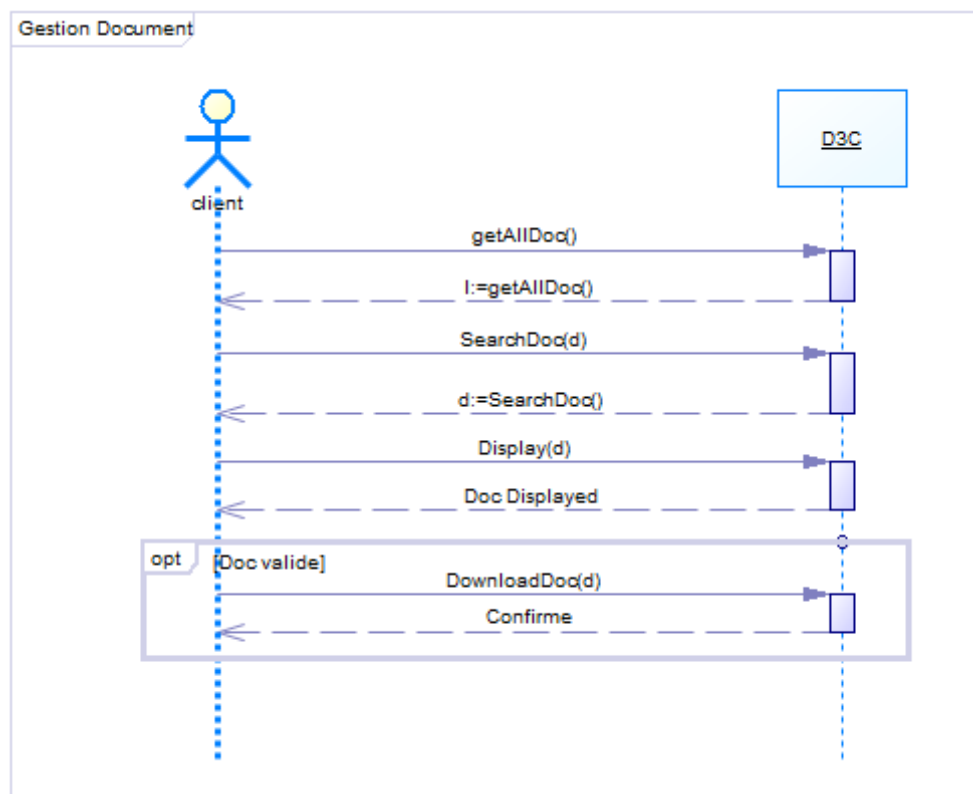


FIGURE 3.10 – Le diagramme de séquence du scénario "Gestion documents"

3.1.3 Diagrammes de BPMN

Business Process Modeling Notation (BPMN) est une représentation graphique permettant de définir des processus métier dans un flux d'informations. c'est un standard pour la modélisation de processus métier qui fournit une notation graphique permettant de définir des processus métier dans un diagramme de processus métier (BPM), basé sur une technique d'organigrammes très proche de celle utilisée par les diagrammes d'activité UML.

L'objectif de BPMN est de supporter la gestion des processus métier pour les utilisateurs technique et métier, tout en étant capable de représenter des sémantiques complexes de processus [Ref8].

Il existe 3 objets de base : les tâches, les évènements et les connecteurs (branchement). Ces objets font partie de la catégorie objets de flux.

Une tâche : est un élément indivisible. Elle représente une action. Chaque tâche a un début et une fin et donc une tâche ne peut débuter que si la tâche précédente est

terminée.

Le branchement : sert à représenter la condition de routage entre le(s) flux en entrée et le(s) flux en sortie. Le branchement n'est pas une tâche et n'effectue aucune action.

Les évènements : servent à identifier un état particulier dans le processus. Ils n'effectuent aucune tâche. Il existe 3 types d'évènements sont début, intermédiaire et fin.

3.1.3.1 Diagrammes de BPMN du processus de D3C

Le pool des utilisateurs est subdivisé en deux parties : pool d'administrateur et pool de client.

La figure suivante représente le diagramme BPMN du processus de D3C.

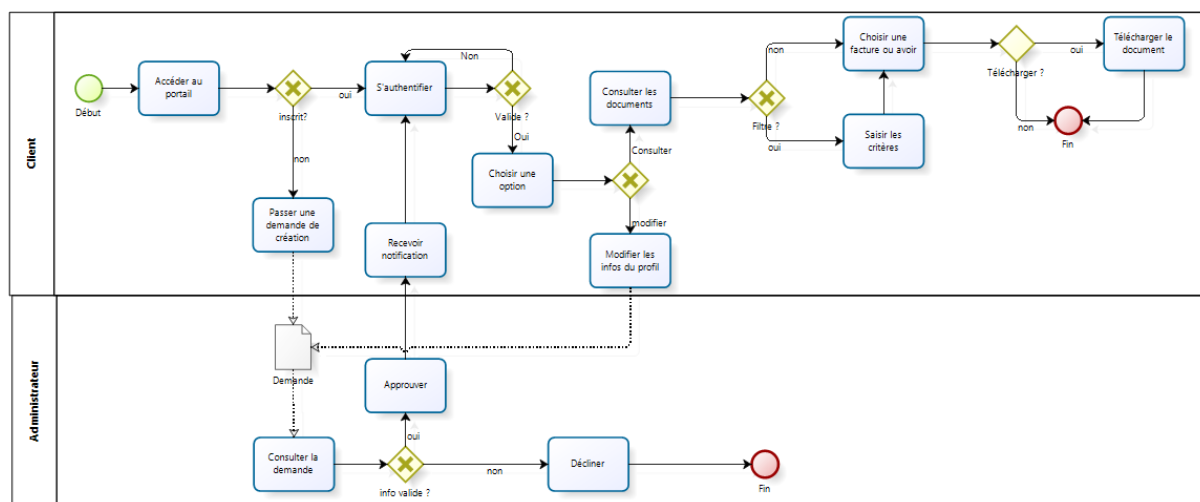


FIGURE 3.11 – Le diagramme de BPMN du processus de D3C

3.1.3.2 Diagrammes de BPMN du processus Sys-générateur

La figure suivante représente le pool du système qui illustre le processus du sys-générateur.

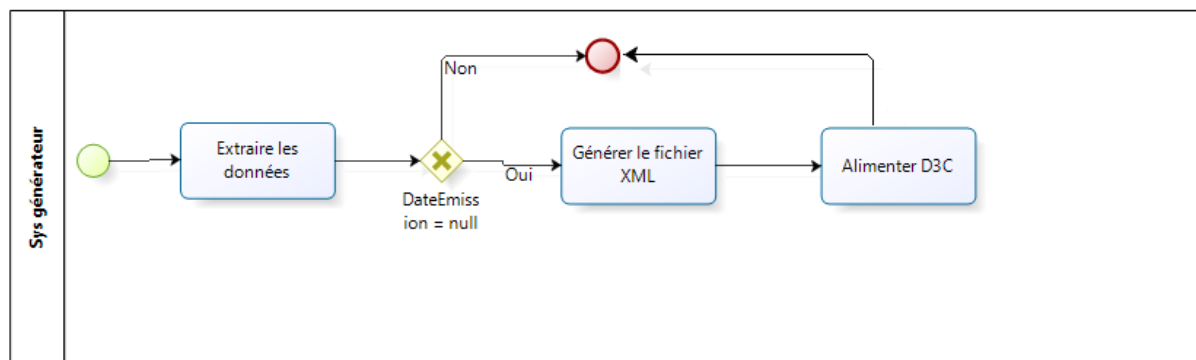


FIGURE 3.12 – Le diagramme de BPMN du sys-générateur

3.1.4 Diagramme de classe

Le diagramme de classe est un diagramme faisant partie des diagrammes structurels et est un des diagrammes d'UML le plus utilisé du fait de sa notation syntaxique riche. Il représente la structure d'une application orientée objet en montrant les classes et les relations qui s'établissent entre elles. C'est une classification qui représente un ensemble d'objets partageant les mêmes spécifications de propriétés, de contraintes et de sémantique [Ref9].

Pour notre application, les principales classes qui jouent le rôle sont Client, Administrateur, Demande et Document, le diagramme de la figure suivante décrit le diagramme de classe utilisé dans cette application, on voit que les deux classes client et administrateur hérite de la classe user, évitant les redondance des attributs et aussi pour faciliter la maintenance lorsque on va penser à d'autres acteurs, l'administrateur peut consulter et traiter les demandes que ça soit création ou bien de modification, ainsi le client peut effectuer les deux types de demandes et de plus consulter ses documents de types avoir ou bien facture.

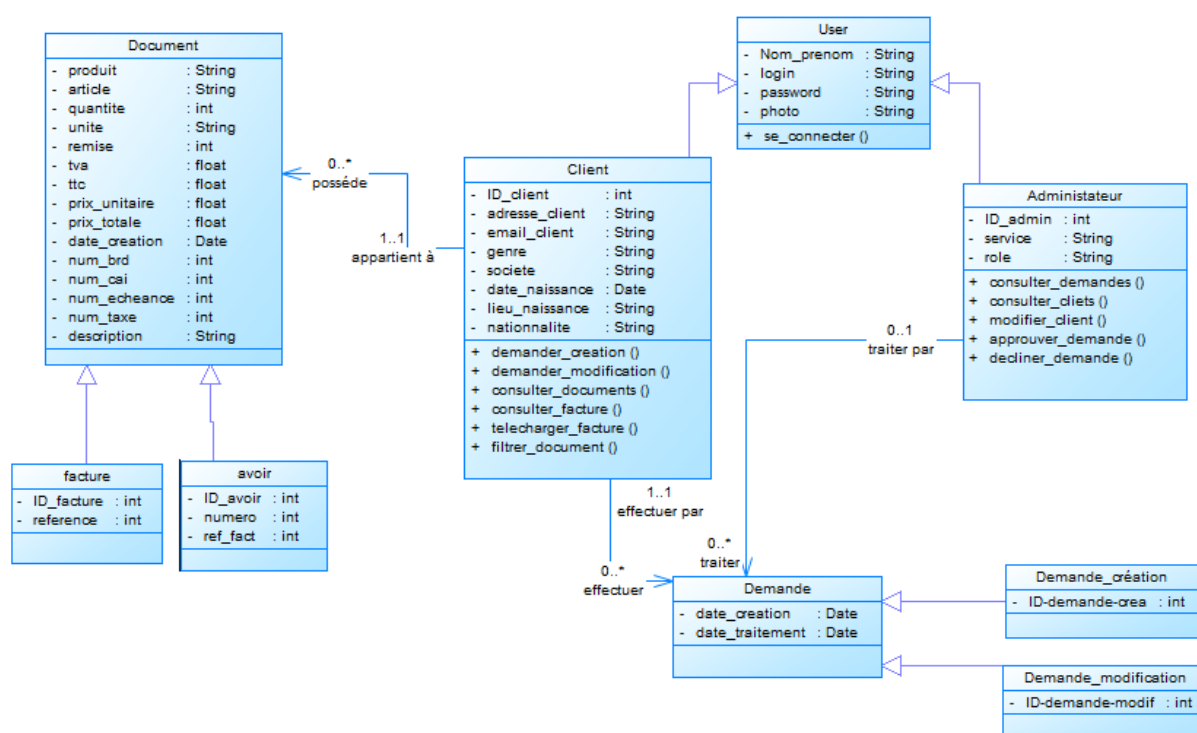


FIGURE 3.13 – Le diagramme de classes

3.1.5 Modèle de données

un modèle de données est un modèle qui décrit de façon abstraite comment sont représentées les données dans une organisation métier, un système d'information ou une base de données. Ce terme modèle de données peut avoir deux significations : Un modèle de données théorique, i.e. une description formelle ou un modèle mathématique. Un modèle de données instance, i.e. qui applique un modèle de données théorique pour créer un modèle de données instance [Ref11].

Puisque les tables de FAE sont toujours existants, et c'est à nous de les analyser, re-documenter et récupérer tous les éléments de conceptions, nous avons adopté la technique de rétro-ingénierie.

La **rétro-ingénierie** [Ref12] est un processus d'analyse de la version opérationnelle de ce composant qui vise à en reconstruire les spécifications techniques et fonctionnelles. La rétro-ingénierie a comme but la redocumentation, la conversion, la maintenance ou l'évolution d'anciennes applications. La redocumentation a pour but de produire une vue alternative d'un modèle, au même niveau d'abstraction.

Table	Libellé
FUAPPT00	APPLICATION GÉNÉRANT DES BL
FUPART00	PARAMÈTRES
FUDATT00	PARAMÈTRES DATE
FUDOCT00	DOCUMENT COMPTABLE
FUDEAT00	DEVISES
FUBRDT00	BORDEREAU
FULBDT00	LIGNE BORDEREAU
FUARCT00	CODE ARTICLE COMMERCIAL
FUASFT00	ADRESSE SPÉCIFIQUE FACTURATION DES DOCUMENTS ÉMIS
FULDET00	DÉTAIL ÉCHÉANCE DE LIGNE BORDEREAU
FUNDOT00	NATURE DE DOCUMENT
FUPCCT00	POINT DE CONTACT CLIENT
FUMTRT00	MOTIF DE RÉGULARISATION
FUTCLT00	CONDITIONS TAXE DE LA LIGNE BORDEREAU
FUTDOT00	TYPES DES DOCUMENTS
FUSTCT00	SOCIETE COMMERCIALE DU GROUPE MICHELIN
FULPRT00	CLASSIFICATION LPR
FUBAET00	ADRESSES EXCEPTIONNELLES BORDEREAUX
FUPAYT00	PAYS
FUMRQT00	MARQUES DE VÉHICULE

TABLE 3.4 – Les termes du modèle de données

3.1.6 Diagramme de déploiement

Un diagramme de déploiement décrit la disposition physique des ressources matérielles qui composent le système et montre la répartition des composants sur ces matériels. Chaque ressource étant matérialisée par un nœud, le diagramme de déploiement précise comment les composants sont répartis sur les nœuds et quelles sont les connexions entre les composants ou les nœuds. Les diagrammes de déploiement existent sous deux formes :

spécification et instance [Ref10].

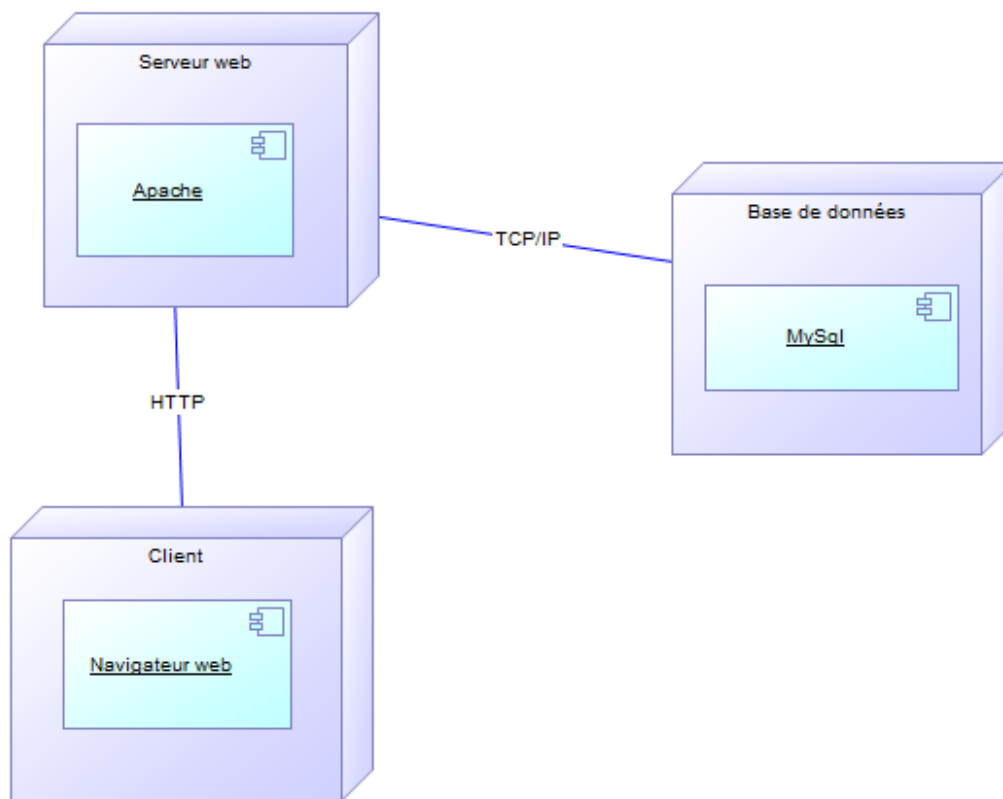


FIGURE 3.15 – Le diagramme de déploiement

Conclusion

Dans ce chapitre, nous avons décrit la conception globale de notre système, à travers les différents diagrammes de UML ainsi que le modèle de données du système existant. Dans le chapitre suivant, nous abordons la mise en place de la solution.

Chapitre 4

MISE EN PLACE DU PROJET

Introduction

Une fois les exigences fonctionnelles clairement identifiées, les besoins et le périmètre clairement cernés et la conception réalisée, on passe à la phase de la mise en place du système. Ce chapitre résume les différentes technologies et outils utilisés lors de cette phase. Il présente également quelques captures d'écrans illustrant les interfaces homme/machine offertes par l'application.

4.1 Étude technique du projet

En ce qui concerne la première partie du projet, il s'agit d'un programme Batch qui est réalisé avec le langage COBOL et exécuté par JCL qui a le rôle d'exécution des Batch. Le système de base des données utilisée dans le mainframe est DB2.

Par rapport au 2ème partie, l'application envisagée est constituée de quatre couches. La couche présentation est réalisée à l'aide du Bootstrap qui permet de réaliser des interfaces IHM ergonomiques et conviviales. La couche métier est réalisée à l'aide du framework Spring qui implémente le concept d'injection des dépendances. Il permet de gérer les objets métiers et les dépendances entre ces objets. La couche d'accès aux données est réalisée à l'aide du framework Hibernate. Le stockage des données est assuré par une base de données MySQL.

4.1.1 Présentation des technologies utilisés

4.1.1.1 Justification des choix

Dans notre contexte, le choix des outils de développement se justifie par les points suivants :

- **Le budget :** L'équipe que nous avons intégrée n'était pas apte à investir un budget assez important dans les outils de développement de cette application. De ce fait, notre choix s'est porté sur les outils open source susceptibles potentiellement de répondre à notre besoin.
- **Compétences personnelles :** le choix du langage COBOL était le sujet de stage

principalement et le choix du langage pour le développement de l'application était un choix de compétences personnelles, je suis familier avec le Java. A comparer avec un autre langage, je l'opte en question de préférence.

- **L'ouverture sur de nouvelle technologie** : ma curiosité sur les nouvelles technologies, ainsi qu'une petite étude sur les outils de tendance sur le marché m'ont convaincus que le Java en premier temps était une bonne décision pour l'application de back-end ainsi que l'utilisation du framework associé « Spring MVC » est la meilleurs des décisions.
- **L'innovation** : l'esprit d'innovation, l'ouverture, la flexibilité et l'encouragement de mes encadrants m'ont permis de mieux choisir mes outils de développement et m'ouvrir encore plus sur la tendance du marché.

4.1.1.2 COBOL (Common Business Oriented language)

COBOL est le langage de programmation le plus utilisé depuis l'invention des ordinateurs. Il a été créé au début des années 60 pour les besoins de l'administration américaine, qui avait besoin d'un langage standardisé pour la programmation des applications de gestion.

Il est surtout utilisé sur des ordinateurs centraux, mainframes, COBOL est de plus en plus développé sur des micro-ordinateurs compte tenu des coûts élevés d'utilisation des ordinateurs centraux.

- C'est un langage très facile à lire. Un programme en COBOL peut aisément être modifié par un autre programmeur.
- Écriture proche du langage naturel (anglais) – on lui reproche d'ailleurs souvent son manque de concision.
- COBOL est adapté aux problèmes de gestion (puissantes instructions d'E/S).
- Le principal inconvénient du COBOL est la longueur de sa codification.[Ref13]

4.1.1.3 JCL (Job Control Language)

JCL (Job Control Language) est un langage permettant de décrire des tâches (unités de travail) sur les systèmes d'exploitation MVS, OS / 390 et VSE, qui s'exécutent sur des

ordinateurs IBM S/390 à gros serveur (ordinateurs centraux ; mainframe). Ces systèmes d'exploitation allouent leurs ressources en temps et en espace au nombre total de tâches lancées sur l'ordinateur. Les emplois à leur tour se décomposent en étapes. Toutes les instructions requises pour exécuter un programme particulier constituent une étape de travail. Les tâches sont des unités de travail en arrière-plan (parfois appelées par lots) qui s'exécutent sans intervention de l'utilisateur (par exemple, les tâches d'impression). En outre, le système d'exploitation gère les demandes d'utilisateurs interactives (au premier plan) qui initient des unités de travail. En général, les travaux de premier plan ont la priorité sur les travaux de fond. [Ref14]

4.1.1.4 JEE (Java Enterprise Edition)

JEE est une plate-forme fortement orientée serveur pour le développement et l'exécution d'applications distribuées. Elle est composée de deux parties essentielles :

Un ensemble de spécifications pour une infrastructure dans laquelle s'exécutent les composants écrits en Java : un tel environnement se nomme serveur d'applications.

Un ensemble d'API qui peuvent être obtenues et utilisées séparément. Pour être utilisées, certaines nécessitent une implémentation de la part d'un fournisseur tiers.

L'utilisation de J2EE pour développer et exécuter une application offre plusieurs avantages :

- une architecture d'applications basée sur les composants qui permet un découpage de l'application et donc une séparation des rôles lors du développement.
- la possibilité de s'interfacer avec le système d'information existant grâce à de nombreuses API : JDBC, JNDI, JMS, JCA ...
- la possibilité de choisir les outils de développement et le ou les serveurs d'applications utilisés qu'ils soient commerciaux ou libres.

L'architecture JEE permet ainsi de séparer la couche présentation, correspondant à l'interface homme-machine (IHM), la couche métier contenant l'essentiel des traitements de données en se basant dans la mesure du possible sur des API existantes, et enfin la couche de données correspondant aux informations de l'entreprise stockées dans des fichiers, dans des bases de données relationnelles ou XML... [Ref15]

4.1.1.5 XML (Extensible Markup Language)

EXtensible Markup Language qui signifie langage de balisage extensible, est un langage informatique qui sert à enregistrer des données textuelles. Ce langage, grosso-modo similaire à l'HTML de par son système de balisage, permet de faciliter l'échange d'information sur l'internet. Contrairement à l'HTML qui présente un nombre finit de balises, le XML donne la possibilité de créer de nouvelles balises à volonté.

Les avantages du XML sont multiples :

- Lisibilité : il est facile pour un humain de lire un fichier XML car le code est structuré et facile à comprendre.
- Disponibilité : ce langage est libre et un fichier XML peut être créer à partir d'un simple logiciel de traitement de texte.
- Interopérabilité : Quelques soit le système d'exploitation ou les autres technologies, il n'y a pas de problème particulier pour lire ce langage.
- Extensibilité : De nouvelles balises peuvent être ajoutée à souhait.[Ref16]

4.1.2 Patrons de conception utilisés

4.1.2.1 Patron MVC

Une application web doit assurer la réalisation des opérations complexes correspondant au métier du concepteur de l'application, et elle doit au même temps proposer à l'utilisateur des interfaces graphiques riches et conviviales.

Il faut séparer au maximum la partie logique et la partie graphique de l'application, afin de donner un aspect parallèle au développement dans le projet. Un autre bénéfice de cette séparation est de faciliter les inévitables mises à jour de l'application. Le modèle MVC répond parfaitement à ce besoin de séparation des différentes parties du développement. C'est un patron qui regroupe les fonctions nécessaires en trois catégories :

- Un modèle (modèle de données) représente les traitements des données, et les interactions avec la base de données, etc
- Une vue (présentation) présente les résultats renvoyés par le modèle, et reçoit toute action de l'utilisateur. Ces différents événements sont envoyés au contrôleur. La vue

n'effectue pas de traitement, elle se contente d'afficher les résultats des traitements effectués par le modèle et d'interagir avec l'utilisateur.

- Un contrôleur (logique de contrôle) prend en charge la gestion des événements de synchronisation pour mettre à jour la vue ou le modèle et les synchroniser. Il reçoit tous les événements de l'utilisateur et déclenche les actions à effectuer.[Ref23]

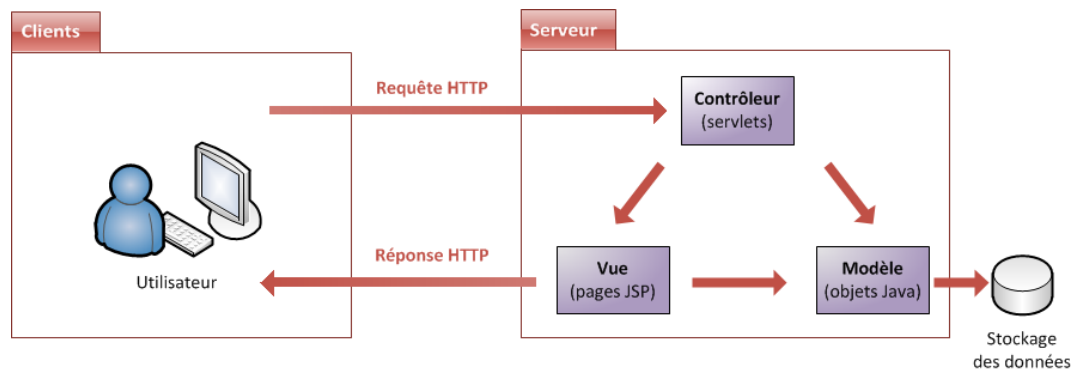


FIGURE 4.1 – Le Patron MVC

4.1.2.2 Patron DAO

Le pattern DAO (Data Access Object) permet de faire le lien entre la couche métier et la couche persistante, ceci afin de centraliser les mécanismes de mapping entre le système de stockage et les objets. Il permet aussi de prévenir un changement éventuel du système de stockage de données.

La couche persistante correspond, en fait, au système de stockage et la couche métier correspond aux objets, mapper la base de données. Le pattern DAO consiste à ajouter un ensemble d'objets dont le rôle sera d'aller lire, écrire, modifier, supprimer dans le système de stockage. Cet ensemble d'objet s'appelle la couche DAO.[Ref20]

4.1.3 Les frameworks adoptés

4.1.3.1 Le framework Spring

Spring est un framework libre et très riche, un conteneur dit «léger», c'est à dire une infrastructure similaire à un serveur d'application Java EE. Il permet de structurer, d'améliorer et de simplifier l'écriture d'application Java EE [Ref17].



FIGURE 4.2 – Le framework Spring

Dans l'application, Spring permet de remédier au problème de mélange du code d'accès à la base de données avec le code de traitement d'une façon nette et parfaite.

4.1.3.2 Le framework Hibernate

Hibernate est une surcouche de JDBC qui lui ajoute une dimension objet. Le terme mapping objet/relationnel (ORM) décrit la technique consistant à faire le lien entre la représentation objet des données et sa représentation relationnelle, basée sur un schéma SQL [Ref18].



FIGURE 4.3 – Framework Hibernate

Concrètement, Hibernate permet de lier/mapper un objet défini en Java avec une table dans une base de données, via un fichier déclaratif de mapping. Le système peut s'occuper de la création des tables en fonction des fichiers de configuration et mettre aussi à jour les tables si nécessaire lors d'un changement dans l'un des fichiers de configuration.

Hibernate possède plusieurs moyens pour effectuer des requêtes. Il est possible d'exprimer des requêtes en SQL, ou en HQL (Hibernate Query Language : langage propre à Hibernate) ou encore en critères orientés objet. On notera que pour les utilisateurs d'Eclipse, il existe un plugin : Hibernate Synchroniser, qui permet de faciliter l'utilisation d'Hibernate.

4.1.3.3 Spring Security

Spring Security propose un modèle de sécurité éprouvé, stable et performant, à même de répondre à l'ensemble des attentes : sécurisation des URLs, des méthodes et des instances d'objets, fourniture de nombreux filtres, par exemple, permettant l'authentification par formulaire ou l'authentification automatique par cookie [Ref19].



FIGURE 4.4 – Spring Security

Pour des besoins de sécurité basiques, Spring Security propose des fonctionnalités faciles à mettre en œuvre, grâce notamment à son schéma XML dédié. Pour des besoins plus avancés, il est nécessaire d'appréhender des mécanismes plus avancés du Framework. Spring Security étant très peu intrusif, sa configuration est indépendante de celle des autres BeansSpring. C'est là son principal atout. La sécurité est en effet une notion transversale, qui ne doit pas avoir d'impact sur l'application en elle-même.

4.1.4 Les outils de développement utilisé

Cette partie est réservée aux principaux outils utilisés durant les phases de la conception et réalisation du projet. Ces outils regroupent ceux ayant servi pour le développement et la production du code source et d'autres ayant servi pour la gestion de projet.

4.1.4.1 Eclipse

Eclipse est un IDE, Integrated Development Environment, c'est-à-dire un logiciel qui simplifie la programmation en proposant un certain nombre de raccourcis et d'aide à la programmation. Développé par IBM, ce logiciel est gratuit et disponible pour la plupart des systèmes d'exploitation. Son intérêt se situe dans la facilitation de la mise en place de différents Framework [Ref21].

4.1.4.2 Apache Tomcat

Apache Tomcat est un conteneur de servlet Java EE. Issu du projet Jakarta, Tomcat est désormais un projet principal de la fondation Apache. Tomcat implémente les spécifications des servlets et des JSP de Sun Microsystems. Il inclut des outils pour la configuration et la gestion, mais peut également être configuré en éditant des fichiers de configuration XML. Comme Tomcat inclut un serveur HTTP interne, il est aussi considéré comme un serveur HTTP [Ref22].

4.1.4.3 Apache Maven

Apache Maven est un outil de gestion et d'automatisation de production des projets logiciels Java en général et Java EE en particulier. Maven est géré par l'organisation Apache Software Foundation. L'outil était précédemment une branche de l'organisation Jakarta Project. L'objectif recherché est de produire un logiciel à partir de ses sources, en optimisant les tâches réalisées à cette fin et en garantissant le bon ordre de fabrication [Ref21].

4.1.4.4 MySQL

MySQL [Ref21] est un système de gestion de bases de données relationnelles qui fait partie des logiciels de gestion des bases de données les plus utilisés dans le monde, autant par le grand public (application web principalement) que par les professionnels. Nous avons choisi MySQL comme serveur de bases de données relationnelles, parce qu'il est :

- OpenSource.
- Multithread et multiutilisateurs.
- Très courant surtout pour les applications Web.
- Facilement intégrable dans un environnement Apache Tomcat.
- Pas gourmand en ressource mémoire et en espace disque.

4.1.4.5 DB2

DB2 est un produit de base de données IBM. C'est un système de gestion de base de données relationnelle (SGBDR). DB2 est conçu pour stocker, analyser et extraire efficace-

ment les données. Le produit DB2 est étendu à la prise en charge des fonctions orientées objet et des structures non relationnelles avec XML. Il utilise le langage SQL tout comme Oracle, PostgreSQL ou MySQL. Il est déployé sur les Mainframes, systèmes UNIX, Windows, Mac/OS et Linux. Il existe également une version allégée pour les ordinateurs de type Palm [Ref24].

4.1.4.6 WampServer

WampServer est une plate-forme de développement Web sous Windows. Elle permet de développer des applications Web dynamiques à l'aide du serveur Apache2, du langage de scripts PHP et d'une base de données MySQL. Elle possède également PhpMyAdmin pour gérer plus facilement les bases de données. Contrairement aux autres solutions, WampServer permet de reproduire fidèlement le serveur de production. Une fois la base installée, on a la possibilité d'ajouter autant de versions d'Apache, MySQL et PHP que nous le souhaitons [Ref25].

4.1.4.7 PhpMyAdmin

PhpMyAdmin est une application Web de gestion pour les systèmes de gestion de base de données MySQL réalisée en PHP et distribuée sous licence GNU GPL.

Il s'agit de l'une des plus célèbres interfaces pour gérer une base de données MySQL sur un serveur PHP. De nombreux hébergeurs, qu'ils soient gratuits ou payants, le proposent ce qui permet à l'utilisateur de ne pas avoir à l'installer.

Cette interface pratique permet d'exécuter, très facilement de nombreuses requêtes comme les créations de table de données, les insertions, les mises à jour, les suppressions, les modifications de structure de la base de données [Ref26].

4.1.4.8 Balsamiq Mockup

Balsamiq Wireframes est un outil d'interface graphique rapide qui reproduit l'expérience de l'esquisse sur un bloc-notes ou un tableau blanc, mais à l'aide d'un ordinateur.

Cela vous oblige vraiment à vous concentrer sur la structure et le contenu, en évitant de longues discussions sur les couleurs et les détails qui devraient intervenir plus tard dans le processus [Ref27].

4.1.4.9 PowerDesigner

PowerDesigner (Anciennement PowerAMC), le n1 de la modélisation de données, est un outil tout-en-un de modélisation d'entreprise et de gestion des métadonnées destiné à documenter l'architecture d'entreprise. PowerDesigner vous accompagne à toutes les étapes de la mise en œuvre de votre projet d'Architecture d'Entreprise, de l'urbanisation des processus à la représentation physique de vos données [Ref28].

4.1.4.10 LaTeX

LaTeX est un langage créé pour séparer le fond de la forme lors de la création d'un document ou d'une publication. En somme, il décrit comment il veut hiérarchiser l'information.

En résumé, LaTeX est un langage de description donnant à l'auteur les moyens d'obtenir des documents mis en page de façon professionnelle sans avoir à se soucier de leur forme. La priorité est donnée à l'essentiel : le contenu [Ref29].

4.1.4.11 SonarQube

Sonarqube, anciennement Sonar, est un logiciel open source qui permet de tester et de mesurer la qualité du code source de façon continue d'un projet informatique.

Sonarqube va pouvoir analyser aussi bien un site internet qu'une application mobile ou Java. Cet outil est utilisé par les équipes de développement donc pour gérer la qualité du code source. Sonar a été développé dans l'objectif de rendre la gestion de la qualité du code accessible au plus grand nombre sans être le développeur [Ref30].

4.2 La réalisation du projet

Notre projet est composé principalement de deux parties, la première concerne la réalisation des deux batchs cobol, l'un de l'extraction des données et l'autre de la génération XML, et la deuxième partie concerne la réalisation du portail D3C. En ce qui suite, les deux parties sont présentées séparément.

4.2.1 Les programmes Batch

Commençons par la première partie du projet, cette partie explique généralement la structure des programmes cobol ainsi que les règles de gestion respecté pour la constitution du batch.

Le schéma suivant représente l'architecture technique de la solution qui retrace les plus grandes spécificités technique dans notre périmètre, cela se résume dans ce qui suit :

- Extraction des données au quotidien.
- Conversion des données au format XML.
- Encodage des données en UTF8 pour les caractères spéciaux.
- Envoie du flux par PELICAN à D3C.

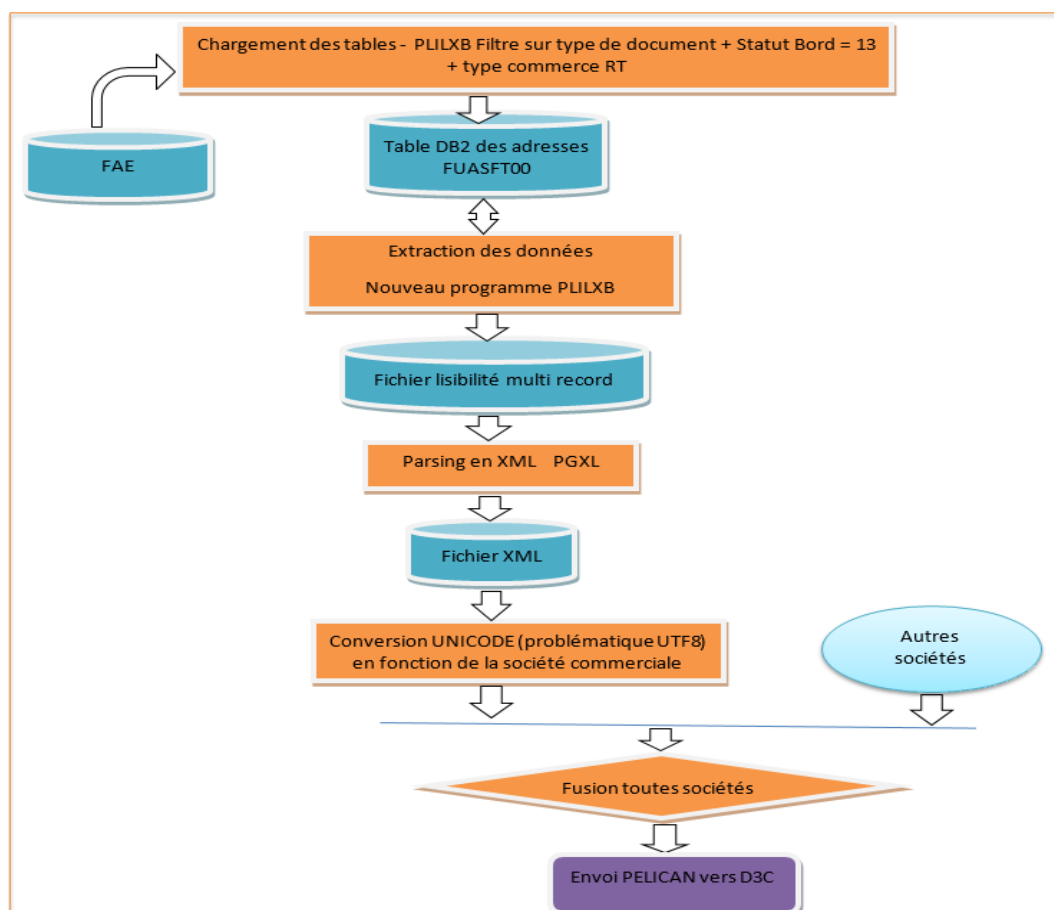


FIGURE 4.5 – Architecture technique de la solution

4.2.1.1 Batch d'extraction des données de FAE

Ce batch est le premier à exécuter afin de préparer le fichier plat avec une structure prédéfinis. Il doit donc charger les tables de la base de données, sélectionner les données en respectant les règles de sélection, prévoir une modification de la table FUASFT00 si nécessaire, sortir le fichier plat avec la bonne structure.

La figure suivante représente le squelette de traitement de ce batch.

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT FACTURE.PFE.PRGXML(PLILXB) - 01.23 Columns 00001 00072
***** ***** Top of Data *****
000001 *-----*
000002 * S Q U E L E T T E   D E   T R A I T E M E N T   B A T C H *
000003 *               I N T E R N A T I O N A L *
000004 *-----*
000005 *
000006 *   PROJET           : MICHELIN *
000007 *
000008 *   APPLICATION      : LISIBILITE FACTURE *
000009 *
000010 *   PROGRAMME        : P L I L X B *
000011 *
000012 *-----*
000013 *
000014 *   PERIODICITE D'EXECUTION : AU QUOTIDIEN *
000015 *
000016 *   ENTREE : *
000017 *       LECTURE DE LA BASE DE DONNEES : TABLES *
000018 *
000019 *   SORTIE : *
000020 *       FIPLAT - FICHER PLAT *
000021 *       FIERROR - FICHER DES ERREURS *
000022 *       CREXEC - COMPTE RENDU D'EXECUTION *
000023 *
000024 *-----*
000025 *               JOURNAL DES MISES A JOUR *
000026 *
000027 *-----*
000028 *               DESCRIPTION ! LE ! PAR *
000029 *-----*
000030 * CREATION !28/05/19! ILYAS BENFATTOUM *
000031 *-----*
000032 *
000033 * IDENTIFICATION DIVISION. *
000034 *-----*
000035 *
000036 * PROGRAM-ID. PLILXB. *
000037

```

FIGURE 4.6 – Squelette du batch d'extraction FAE

Les règles générales de l'extraction

Pour bien répondre au cahier des charges, on a développé un programme suivant les spécifications suivantes en respectant les résultats d'analyse :

- En début de traitement, l'initialisation de toutes les variables s'avère nécessaire.
- Chargement des tables et contrôle sur l'ouverture des curseurs.
- La date APP-EMI-DT est initialisée à la valeur Null; Ceci pour permettre de sélectionner les documents qui ne sont pas encore émis vers LISIFACT.
- Accéder à la table DOC pour récupérer le DOC-NTU puis le DOC-TYP ; Ne charger en table que les documents de type 1 ou 2 (Facture / Avoir).

- Accéder à la table BRD avec le DOC-NO et la DOC-CRE-DT ; Ne charger en table que si le Bordereau est au statut 13 (facturé) et si le type de commerce = RT.
- Charger les adresses exceptionnelles de chaque bordereau.
- Charger toutes les lignes bordereau, les échéances et les taxes de chaque ligne.
- Prévoir une modification de la date APP-EMI-DT par la date comptable du jour afin de ne re-sélectionner pas les documents dans les prochains lancements.

La structure du fichier de sortie

Le fichier de données est typé en fonction des informations portées par l'enregistrement

- 01 Adresses Facturation
- 02 Document (on peut avoir n docs pour une adresse de facturation)
- 03 Bordereau (on peut avoir n BL pour un doc) – on a une adresse de livraison par BL
- 04 CAI (on peut avoir n CAI pour un BL)
- 05 Echéance (on peut avoir n échéances pour un BL)
- 06 TAXE (on peut avoir n Taxes pour un BL)

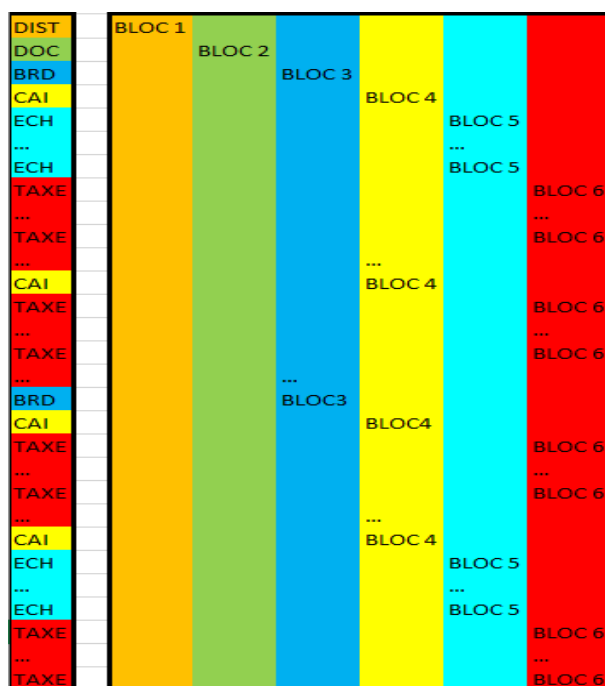


FIGURE 4.7 – Structure du fichier de sortie

Stratégie d'accès aux données

Pour extraire les données, et respecter la structure imposée ci-dessus avec la diversité de ces données, il est primordial de mettre en place une stratégie d'accès aux données pour bien gérer les différents curseurs qui se complète et respecte les différentes relations et correspondances entre les tables.

Le curseur est nécessaire pour le stockage du résultat de la requête de sélection qui est généralement plusieurs lignes dans notre cas.

Exemple d'ouverture et de fetch sur un curseur :

```

000744
000745 *-----*
000746 *          --- TRAIT. ---  CURSEURS  ---*
000747 *-----*
000748
000749 *OUVR-CUR-DIST SECTION.
000750
000751 EXEC SQL OPEN CUR-DIST END-EXEC.
000752 IF SQLCODE NOT = +0 AND +100
000753 MOVE 'ERREUR OPEN CUR-DIST' TO WS-MSG-ERR-SQL
000754 PERFORM ERRO-SQL
000755 ELSE
000756 IF SQLCODE = +0
000757
000758 *
000759 *          *-----*
000760 *          *          FETCH SUR CUR-DIST          *
000761 *          *-----*
000762 EXEC SQL FETCH CUR-DIST
000763 INTO
000764         , :STE-COM-C
000765         , :FACN-PCC-ID
000766         , :FACN-PCC-RS
000767         , :FACN-ADR1
000768         , :FACN-ADR2
000769         , :FACN-PAYS-LIB
000770         , :WS-DOC-NO
000771         , :WS-DOC-CRE-DT
000772
000773 END-EXEC
000774
000775 IF SQLCODE NOT = +0 AND +100
000776 MOVE 'ERREUR FETCH CUR-DIST' TO WS-MSG-ERR-SQL
000777 PERFORM ERRO-SQL
000778 ELSE
000779 IF SQLCODE = +100
000780 ***** CUR-DIST IS EMPTY *****
000781 DISPLAY " LE CURSEUR ( CUR-DIST ) EST VIDE "
000782 MOVE '00' TO WO-TAG
000783 PERFORM ECRITURE-FIPLAT
000784 PERFORM CLOSE-CUR-DIST
000785
000786 ELSE
000787 MOVE STE-COM-C TO WS-STE-COM-C
000788 PERFORM UPDATE-ASF
000789 MOVE '01' TO WO-TAG
000790 PERFORM ECRITURE-FIPLAT
000791 *
000792 OPEN AND FETCH --- CURSOR DOC
000793 EXEC SQL OPEN CUR-DOC END-EXEC
000794

```

FIGURE 4.8 – Le curseur CUR-DIST

L'exécution du Batch

Pour un traitement répétitif, afin de sélectionner que les documents du jour, nous avons prévu un lancement quotidien et automatique de nos deux batchs avec les caractéristiques suivantes :

- 6 passages par semaine, tous les jours sauf le lundi matin
- Heure de passage : vers 1h30 du matin

Pour exécuter un programme batch COBOL, nous devons créer un JCL et le soumettre. Ce JCL nous permet de définir les données à utiliser et/ou les fichiers d'entrée et/ou les fichiers de sortie, définir le programme à exécuter et faire la liaison avec le load du programme.

La figure suivante représente le JCL d'exécution pour le programme PLILXB d'extraction des données.

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT FACTURE.PFE.PRGXML(JCLDB2) - 01.10 Columns 00001 00072
***** ***** Top of Data *****
000001 //JCLXDB2 JOB 'TEST',CLASS=A
000002 //*
000003 //STEP1 EXEC PGM=PLILXB
000004 //STEPLIB DD DSN=FACTURE.PFE.LOAD,DISP=SHR
000005 //* FICHER DE SORTIE
000006 //FILEP DD DSN=FACTURE.PFE.FP,DISP=SHR
000007 //FIERR DD DSN=FACTURE.FILE.ERROR,DISP=SHR
000008 //*
000009 //SYSOUT DD SYSOUT=*
000010 //*
000011 //SYSPRINT DD SYSOUT=*
000012 //*
000013 //SYSTSIN DD *
000014 RUN PROGRAM(PLILXB) PLAN(PLAND) -
000015 LIB('FACTURE.PFE.LOAD')
000016 END
000017 /*
*****

```

FIGURE 4.9 – Le JCL d'extraction des données

Comme indiqué au début, la sortie de ce programme est un fichier plat structuré avec des tags selon le type d'enregistrement. (Pour plus de détails : voir l'annexe)

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT FACTURE.PFE.FP Columns 00001 00072
***** ***** Top of Data *****
000001 01B07M0000514BARANYA GUMISZERVIZ PÉCSVÁRADI ÚT 2/ 1. 760
000002 021TERHELES HF546074 2019-04-15HUF2019-04-15
000003 03VD E PCR49586311 2019-04-15 2019-04-15H8355 M
000004 040001817598MICHELIN TSMPASSENGER CAR SUMMER
000005 05 43289.002019-06-05
000006 06HU PROD BAS 27% 8997.00 0.27000
000007 01B07M0000516BAUMGARTNER GUMISZERVIZ CSABA ÚT 9.. 9700
000008 021TERHELES HF546075 2019-04-15HUF2019-04-15
000009 03VD E PCR49570311 2019-04-15 2019-04-12H8210 MK
000010 040001452329MICHELIN TSMPASSENGER CAR SUMMER
000011 05 155682.002019-06-05
000012 06HU PROD BAS 27% 32510.00 0.27000
000013 021TERHELES HF546076 2019-04-15HUF2019-04-15
000014 03VD E PCR49570911 2019-04-15 2019-04-15H8237 MK
000015 040001674619MICHELIN TSMPASSENGER CAR SUMMER
000016 05 128463.002019-06-05
000017 06HU PROD BAS 27% 26711.00 0.27000
000018 021TERHELES HF546077 2019-04-15HUF2019-04-15
000019 03VD E PCR49572511 2019-04-15 2019-04-15H8322 MK
000020 040001668455TIGAR TSMPASSENGER CAR SUMMER
000021 05 57264.002019-06-05
000022 06HU PROD BAS 27% 11683.00 0.27000
000023 00FIN
***** ***** Bottom of Data *****

```

FIGURE 4.10 – Le contenu fichier plat

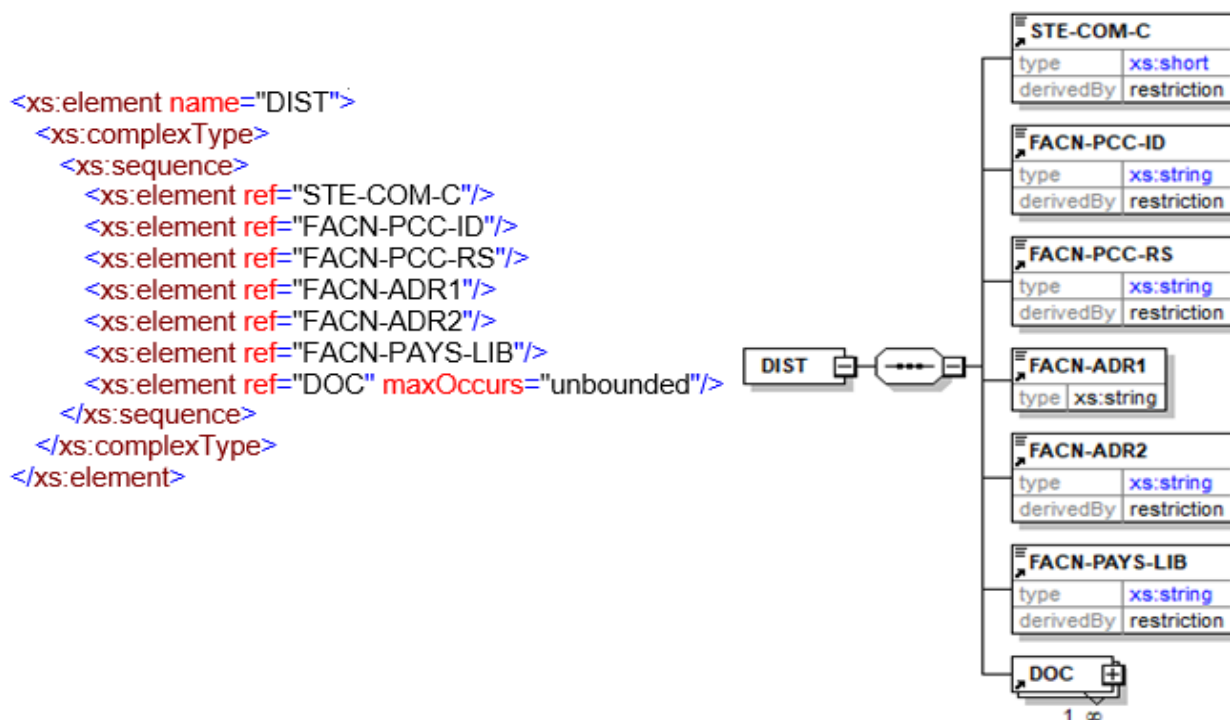


FIGURE 4.12 – Le XSD DIST

La figure précédente illustre le schéma XML ou XSD avec une vue graphique de bloc DIST qu'il doit être respecté afin que le fichier XML soit valide.

Comme on voit d'après la figure, le tag DIST doit être conforme au schéma de telle façon qu'il comporte 7 éléments, dans lequel le dernier élément peut se reproduire plusieurs fois. Les autres figures des autres blocs sont dans l'annexe, à la fin du rapport.

L'exécution du Batch

L'exécution de ce deuxième batch pour générer le XML est représentée dans le JCL suivant, il prend en entrée le fichier plat et comme sortie le fichier XML et le step ICONV pour la conversion en UTF-8, mentionnant que la conversion peut se faire au sein du programme, d'une manière manuelle si on a plus de configuration à faire, à savoir un type des caractères spéciaux d'une langue spécifique.

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT FACTURE.PFE.PRGXML(JCLXML1) - 01.08 Columns 00001 00072
***** ***** Top of Data *****
000001 //JCLMBR JOB 'TEST',CLASS=A
000002 //STEP1 EXEC PGM=PGXL
000003 //STEPLIB DD DSN=FACTURE.PFE.LOAD,DISP=SHR
000004 //ENTREE DD DSN=FACTURE.PFE.FP,DISP=SHR
000005 //SORTIE DD DSN=FACTURE.PFE.FXML,DISP=SHR
000006 //SYSOUT DD SYSOUT=*
000007 //*
000008 //* CONVERSION UTF-8
000009 //ICONV EXEC PROC=EDCICONV,
000010 // INFILE=FACTURE.PFE.FXML,
000011 // OUTFILE=FACTURE.XML.UTF8,
000012 // FROMC='IBM-037',
000013 // TOC='IBM-1047'
000014 //SYSPRINT DD SYSOUT=*
000015 //*

```

FIGURE 4.13 – Le JCL de génération XML

Et voila donc une partie du fichier XML généré à partir du fichier plat et qui est conforme au schéma XML expliqué avant :

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
EDIT STAGE.PFE.FXML Columns 00001 00072
***** ***** Top of Data *****
==MSG> -Warning- The UNDO command is not available until you change
==MSG> your edit profile using the command RECOVERY ON.
000001 <DIST><STE-COM-C>807</STE-COM-C><FACN-PCC-ID>M0000514</FACN-PCC-ID><FACN
000002 <DOC><DOC-TYP>1</DOC-TYP><DOC-TYP-LIB>TERHELES</DOC-TYP-LIB><DOC-NO>HF54
000003 <BRD><APP-C>VD</APP-C><EMI-L-IDR>E</EMI-L-IDR><BORD-NO> PCR495863</BORD
000004 <CAI><BORD-LG-NO>0001</BORD-LG-NO><CAI-ID>817598</CAI-ID><MRQ-LIB>MICHEL
000005 <ECH><ECH-MT> 43289.00</ECH-MT><ECH-DT>2019-06-05</ECH-DT></ECH>
000006 <TAXE><TAX-COND-LIB>HU PROD BAS 27%</TAX-COND-LIB><TAX-MT> 8997.
000007 <DIST><STE-COM-C>807</STE-COM-C><FACN-PCC-ID>M0000516</FACN-PCC-ID><FACN
000008 <DOC><DOC-TYP>1</DOC-TYP><DOC-TYP-LIB>TERHELES</DOC-TYP-LIB><DOC-NO>HF54
000009 <BRD><APP-C>VD</APP-C><EMI-L-IDR>E</EMI-L-IDR><BORD-NO> PCR495703</BORD
000010 <CAI><BORD-LG-NO>0001</BORD-LG-NO><CAI-ID>452329</CAI-ID><MRQ-LIB>MICHEL
000011 <ECH><ECH-MT> 155682.00</ECH-MT><ECH-DT>2019-06-05</ECH-DT></ECH>
000012 <TAXE><TAX-COND-LIB>HU PROD BAS 27%</TAX-COND-LIB><TAX-MT> 32510.
000013 <DOC><DOC-TYP>1</DOC-TYP><DOC-TYP-LIB>TERHELES</DOC-TYP-LIB><DOC-NO>HF54
000014 <BRD><APP-C>VD</APP-C><EMI-L-IDR>E</EMI-L-IDR><BORD-NO> PCR495709</BORD
000015 <CAI><BORD-LG-NO>0001</BORD-LG-NO><CAI-ID>674619</CAI-ID><MRQ-LIB>MICHEL
Command ==> Scroll ==> CSR
F1=Help F2=Split F3=Exit F5=Rfind F6=Rchange F7=Up
F8=Down F9=Swap F10=Left F11=Right F12=Cancel

```

FIGURE 4.14 – Le fichier XML généré

4.2.2 Présentation des interfaces de l'application

Cette partie est essentiellement consacrée à la présentation des principales interfaces du système sous forme de captures d'écrans.

4.2.2.1 Page authentification

Une fois lancée, l'application affiche à l'utilisateur la page d'authentification, l'utilisateur doit introduire son login et son mot de passe pour pouvoir accéder aux différentes fonctionnalités du portail.

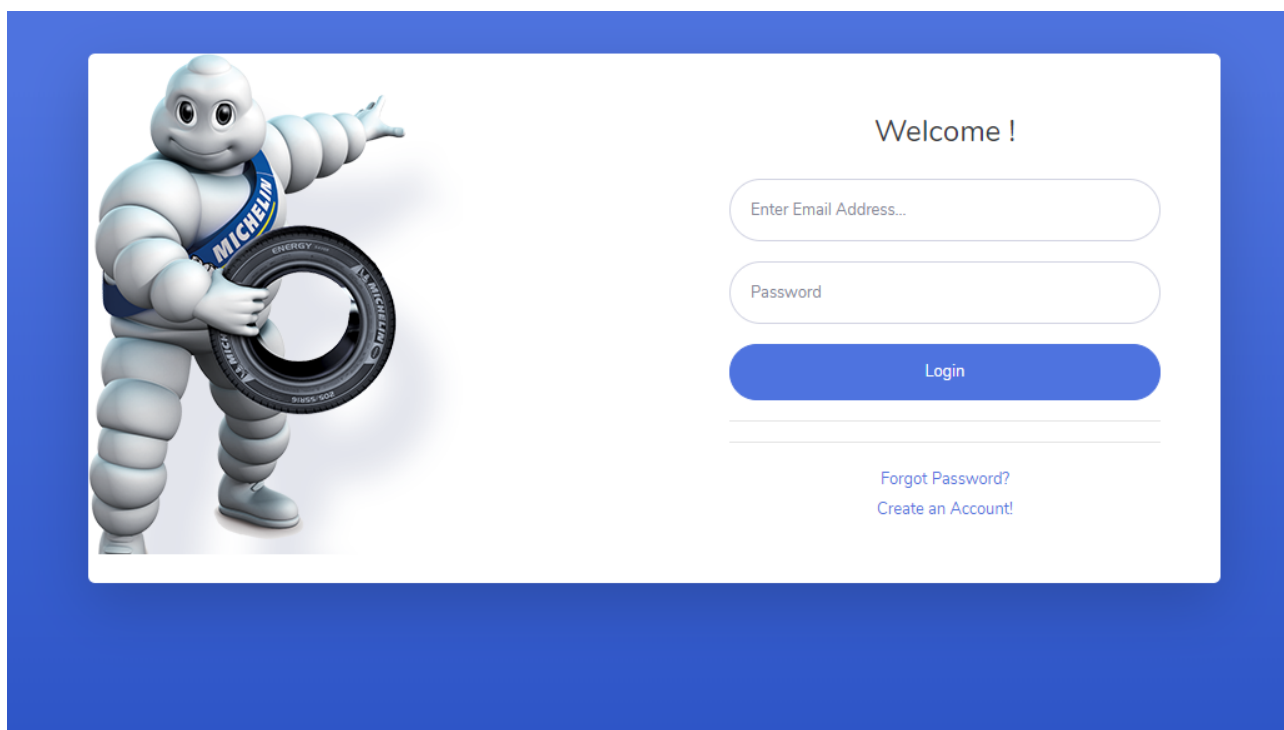


FIGURE 4.15 – Page d'authentification

Le système vérifie les informations saisis, s'ils sont valides, l'utilisateur accède au page d'accueil, sinon s'il arrive pas a se connecter, il peut réinitialiser son mot de passe par l'envoi de son email à l'administrateur pour lui envoyer son nouveau mot de passe.

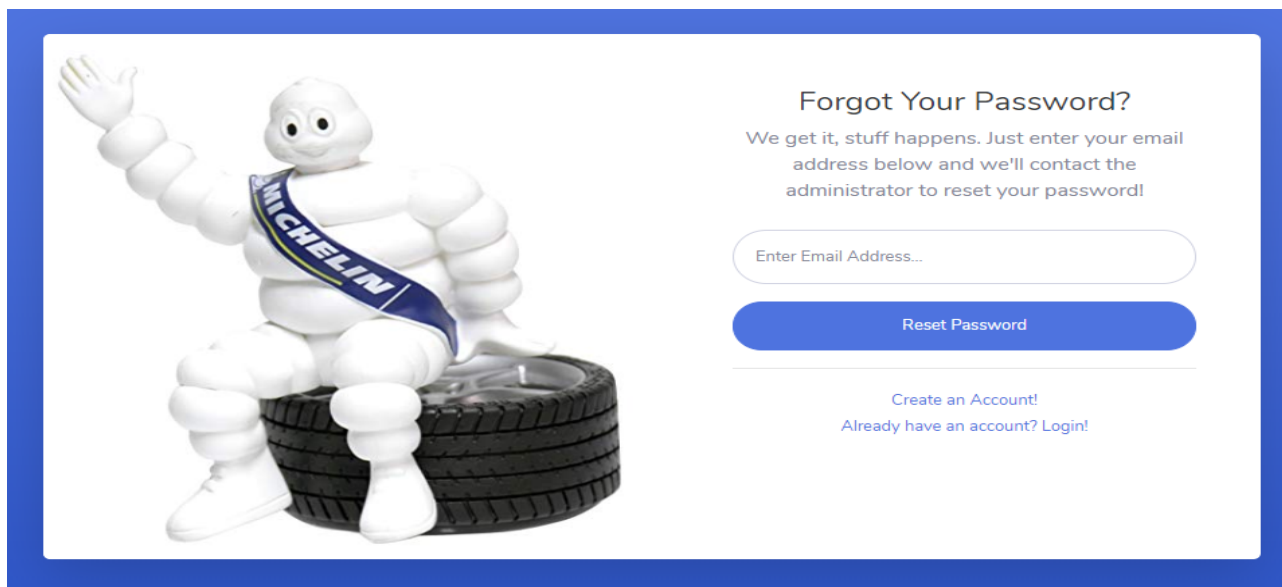


FIGURE 4.16 – Page de ré-initialisation du mot de passe

4.2.2.2 Page de création de compte

Au cas où l'utilisateur n'a pas de compte, il a juste à faire une demande de création du compte qui va être traité et validé par l'administrateur, et il va recevoir par la suite son login et mot de passe par email.

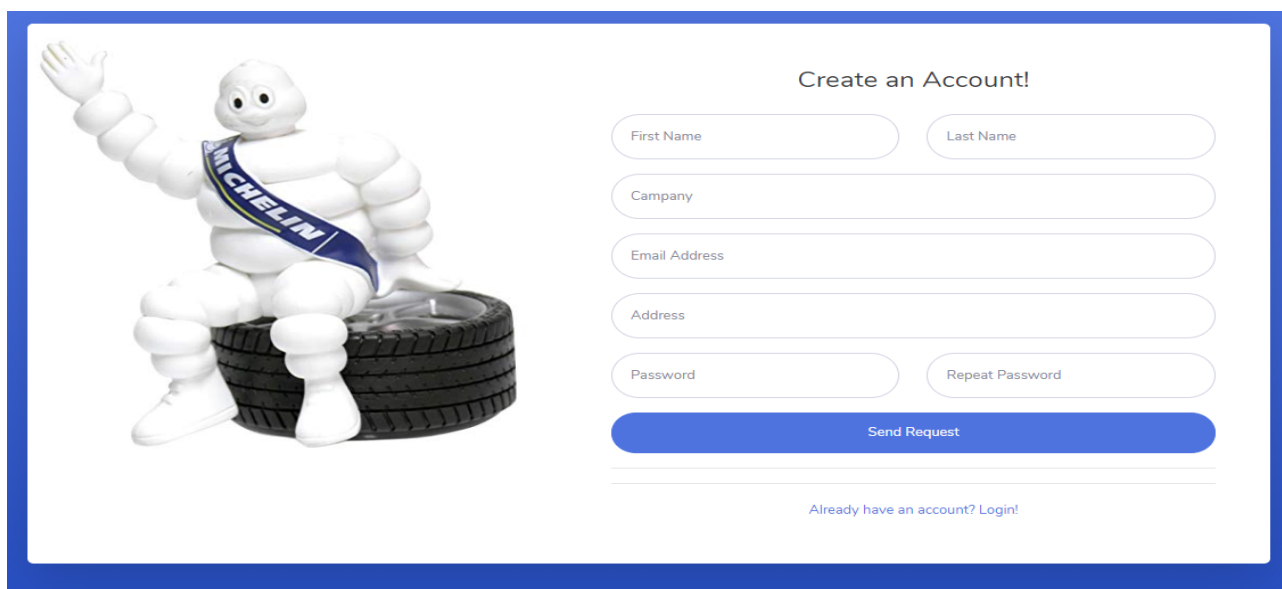


FIGURE 4.17 – Page de création de compte

4.2.2.3 Page de consultation des documents

Cette page permet aux clients de consulter tous ses documents que ce soit les factures ou les avoirs.

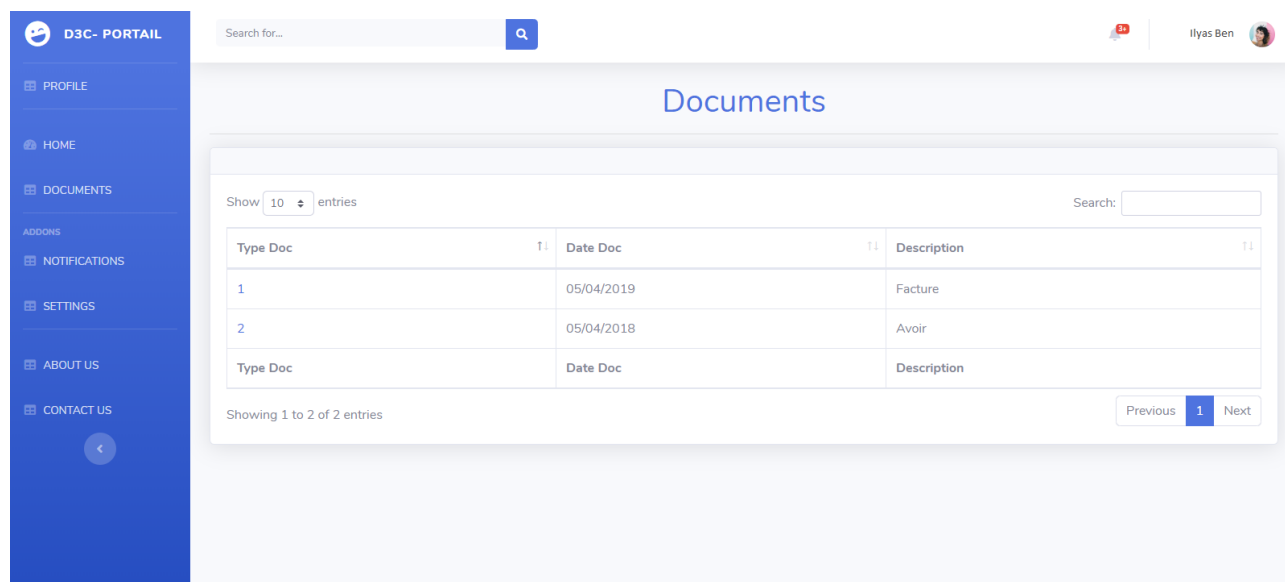


FIGURE 4.18 – Page de consultation des documents

4.2.2.4 Page de détails facture

Comme on peut voir, la liste des documents affichés est trié selon le type du document (facture/avoir), à chaque fois que le client clique sur un document, il accède directement au détails de cette facture/avoir.

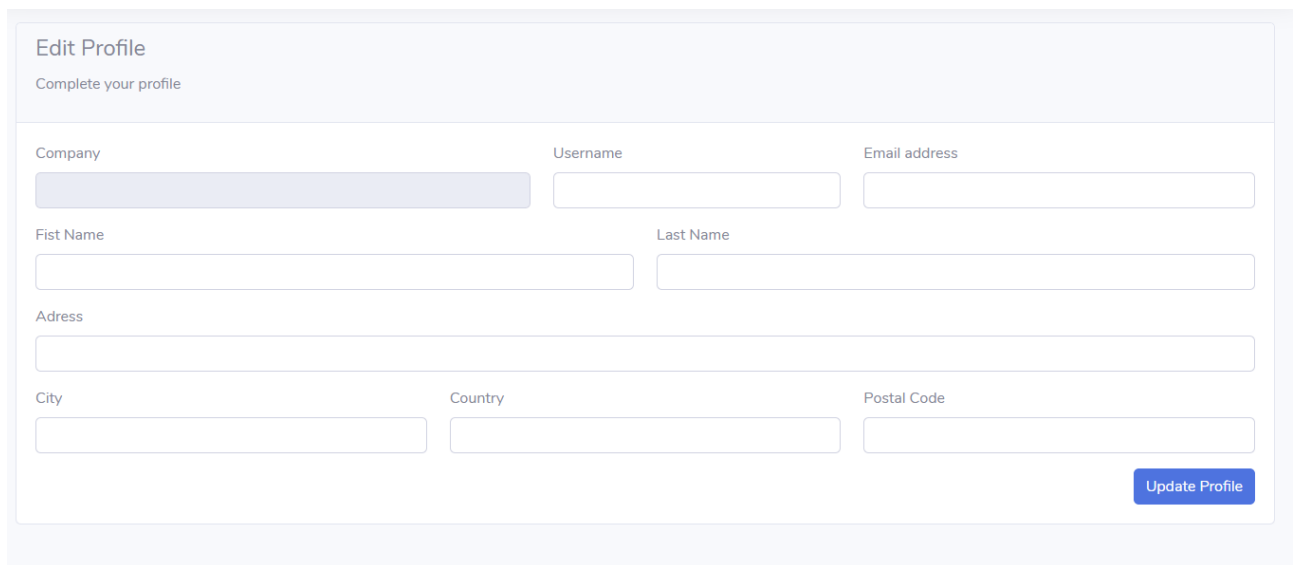
En effet, au haut de la facture, il se trouve des informations concernant le client et ses coordonnées, et juste en bas, les détails du facture concernant les prix, les remises et la quantité etc, il peut même la télécharger en cliquant sur le bouton bleu en haut.

FACTURE					Download document
PROJECT	Website development			Company Name	
CLIENT	John Doe			455 Foggy Heights,	
ADDRESS	796 Silver Harbour, TX 79273, US			AZ 85004, US	
EMAIL	john@example.com			(602) 519-0450	
DATE	August 17, 2015			company@example.com	
DUE DATE	September 17, 2015				
SERVICE	DESCRIPTION	PRICE	QTY	TOTAL	
Design	Creating a recognizable design solution based on the company's existing visual identity	\$40.00	26	\$1,040.00	
Development	Developing a Content Management System-based Website	\$40.00	80	\$3,200.00	
SEO	Optimize the site for search engines (SEO)	\$40.00	20	\$800.00	
Training	Initial training sessions for staff responsible for uploading web content	\$40.00	4	\$160.00	
				SUBTOTAL	\$5,200.00
				TAX 25%	\$1,300.00
				GRAND TOTAL	\$6,500.00
NOTICE: A finance charge of 1.5% will be made on unpaid balances after 30 days.					

FIGURE 4.19 – Page de détails facture

4.2.2.5 Page de profile

A travers cette page, l'utilisateur peut modifier ses propres informations, mais ils seront pas modifiés tout de suite parce qu'il fallait que l'administrateur approuve la demande. Juste après l'approbation le client reçoit une notification lui informe que son demande est acceptée.

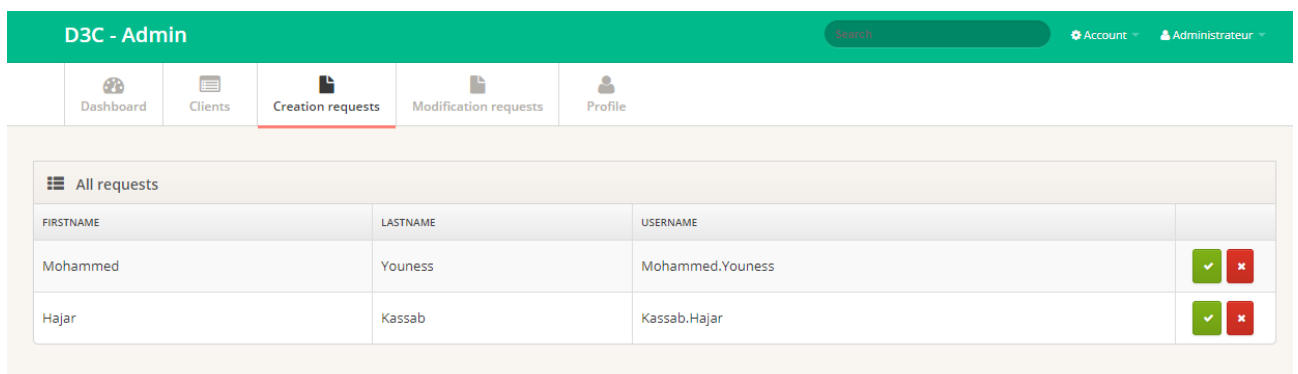


The 'Edit Profile' form is titled 'Edit Profile' with the subtitle 'Complete your profile'. It contains several input fields: 'Company' (a wide text box), 'Username' (a text box), 'Email address' (a text box), 'Fist Name' (a text box), 'Last Name' (a text box), 'Adress' (a text box), 'City' (a text box), 'Country' (a text box), and 'Postal Code' (a text box). A blue 'Update Profile' button is located at the bottom right of the form.

FIGURE 4.20 – Page de profile

4.2.2.6 Page des demandes de création

Concernant l'administrateur, il a tout a fait une interface différente. En accédant à son espace, il peut voir toutes les demandes de création de compte qu'il a déjà approuvée, ainsi que les nouvelles qu'il doit les traiter. une fois traité, le login et le mot de passe est envoyé par email au client pour qu'il puisse accède à son espace.



The 'D3C - Admin' interface features a green header with a search bar and user information. Below the header is a navigation bar with icons for 'Dashboard', 'Clients', 'Creation requests' (highlighted), 'Modification requests', and 'Profile'. The main content area shows a table titled 'All requests' with columns for 'FIRSTNAME', 'LASTNAME', 'USERNAME', and a status column with green checkmark and red X icons.

FIRSTNAME	LASTNAME	USERNAME	
Mohammed	Youness	Mohammed.Youness	<input checked="" type="checkbox"/> <input type="checkbox"/>
Hajar	Kassab	Kassab.Hajar	<input checked="" type="checkbox"/> <input type="checkbox"/>

FIGURE 4.21 – Page des demandes de création

4.2.2.7 Page des demandes de modification des informations

La même chose pour les demandes de modification des informations des clients, sauf que ici après l’approbation directement une notification est envoyé au client lui informe que son demande est traitée.

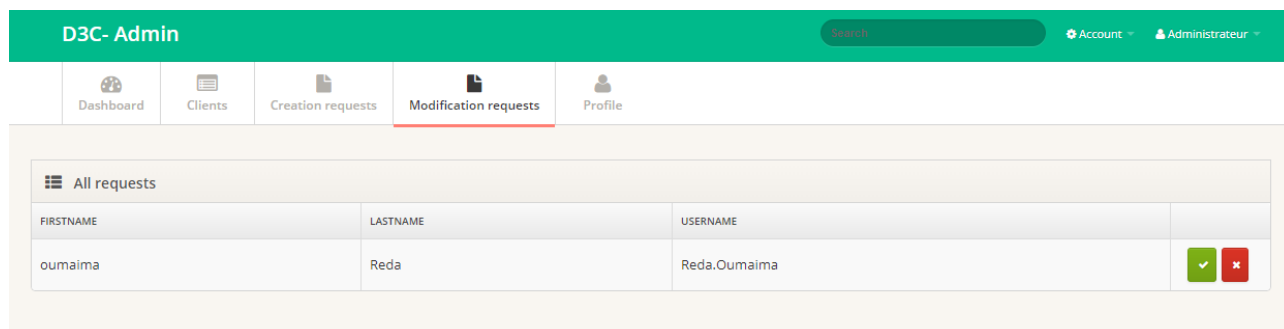


FIGURE 4.22 – Page des demandes de modification

4.2.2.8 Page de consultation des clients

Cette page permet au l’administrateur la consultation de tous les clients , avec la possibilité de modifier ses informations avec le botton vert à droite .

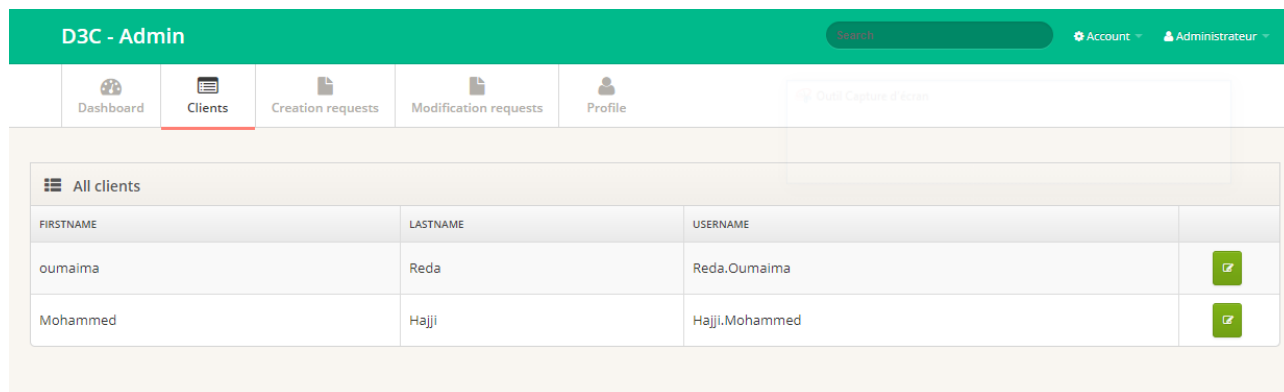


FIGURE 4.23 – Page de consultation des clients

Et après pour modifier ses informations on se redérige vers l'interface ci dessous.

The screenshot displays the 'D3C - Admin' web application interface. At the top, there is a green header bar with the text 'D3C - Admin' on the left, a search bar in the center, and 'Account' and 'Administrateur' on the right. Below the header is a navigation menu with five items: 'Dashboard', 'Clients' (which is highlighted with a red underline), 'Creation requests', 'Modification requests', and 'Profile'. The main content area is titled 'Account' and contains a section for 'Edit informations'. This section includes several input fields: 'Username' (pre-filled with 'Reda.Oumaima' and a note stating it cannot be changed), 'First Name' (pre-filled with 'Oumaima'), 'Last Name' (pre-filled with 'Reda'), 'Email Address' (pre-filled with 'Reda.Oumaima@cgi.com'), 'Password', and 'Confirm'. At the bottom of the form are 'Save' and 'Cancel' buttons.

FIGURE 4.24 – Page de modification des informations client

4.3 Les tests et validation

4.3.1 Les tests réalisés

La phase des tests est une phase très importante du cycle du développement d'un programme, il permet de vérifier la cohérence des résultats du traitement avec les résultats attendus, ce qui donnera une preuve concrète pour légitimer la mise en production du programme pour le mettre en place dans le système. Voici ci-dessous les tests que nous avons fait après exécution du programme :

Libellé	Description	Résultat attendu	Résultat obtenu
Ouverture des fichiers	Lors de l'ouverture d'un fichier, le statut du fichier est mis à jour par une valeur selon différents cas, 0 si l'ouverture est bien déroulée, si le statut est différent de zéro, nous générons et nous écrivons l'erreur dans un fichier de gestion des erreurs	Ouvrir le fichier et écrire les données extraites dans ce fichier	Exécution correcte
Connexion à la base des données	Lors de l'ouverture d'un curseur, la variable SQLCODE est mis à jour par une valeur selon différents cas, 0 si l'ouverture est bien déroulé et 100 si l'ouverture du curseur est impossible	Ouvrir le curseur et exécuter la requête	Exécution correcte
Gestion des dates	Les dates doivent respecter un format bien précis pour ne risque pas d'arrêter le programme	Les dates sont conformes au format donné	Exécution correcte
Gestion des clients	Les clients ont besoin de créer un compte et de modifier ses informations personnelles facilement	Envoyer une demandes de modification et demande la création de compte	Exécution correcte
Gestion des demandes	l'administrateur reçoit les demandes de création et de modification, il a la possibilité soit d'approuver si les informations sont valide ou de décliner sinon	Approuver ou décliner les demandes	Exécution correcte
Gestion des documents	Les clients ont la possibilité de consulter leurs facture/avoir et de les télécharger à tout moment	Consulter et télécharger la facture	Exécution correcte

TABLE 4.1 – Les tests réalisés

4.3.2 La qualité du code

Dans le cadre d'un projet de développement, la qualité du code est un élément très important. Connaître toutes les bonnes pratiques de développement d'un langage n'est pas forcément chose acquise pour tout l'ensemble des développeurs et encore moins respectées par tous les développeurs. Heureusement pour cela il existe de nombreux outils permettant de nous simplifier la tâche et SonarQube fait partie de ceux-là.

SonarQube est alors un logiciel libre permettant de mesurer la qualité de votre projet de plusieurs façons :

- * Respect des règles du code
- * Documentation du code
- * Analyse des tests unitaires mis en place
- * Duplication du code

La figure suivante est la bilan du code d'application sur SonarQube :

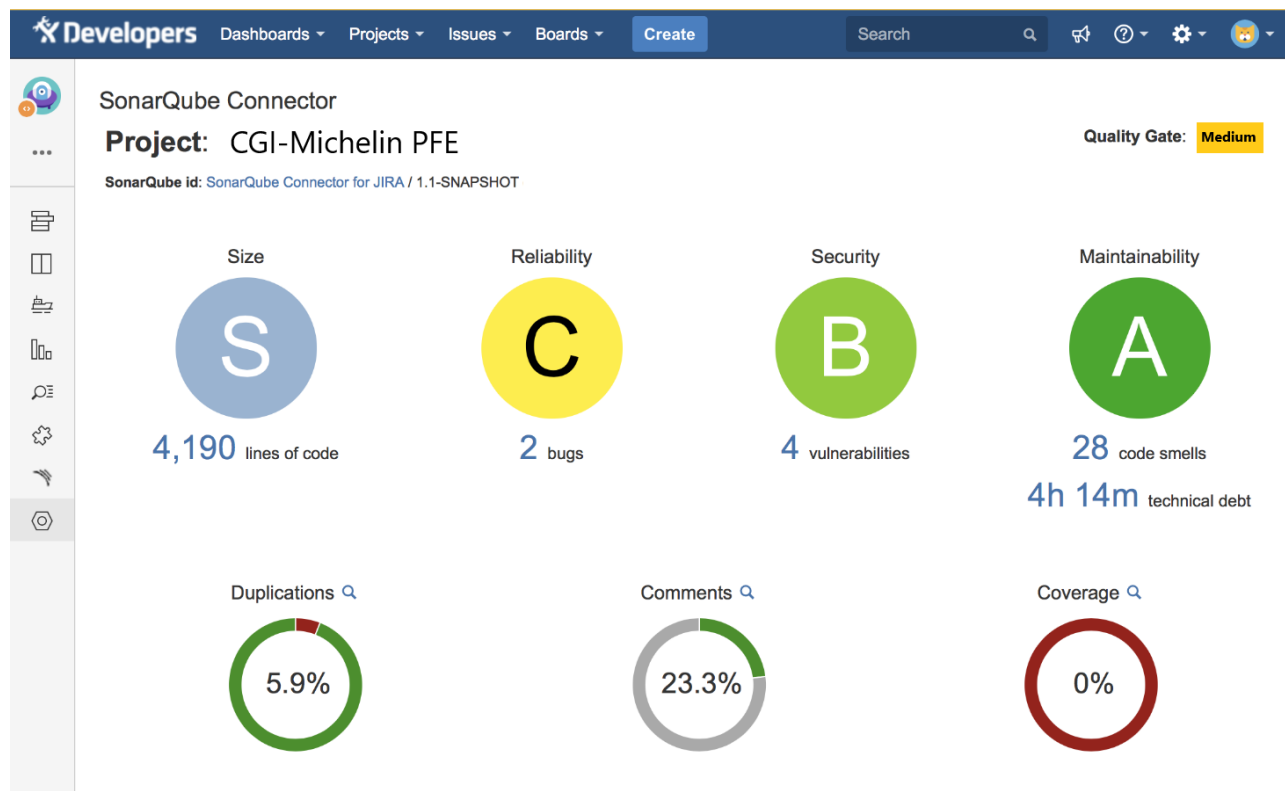


FIGURE 4.25 – Bilan du projet en SonarQube

La capture précédente est un résumé ,après correction, des principales informations sur le code , on voit que on a 4190 lignes de code , 2 bugs qui n'ont pas un effet sur l'exécution de l'application , 4 vulnérabilités de sécurité et y'a des duplications et commentaires et la qualité est moyenne , ces nombres sont très négligeables par rapport au premier analyse après que la qualité était 'failed' et en rouge.

Et voila un exemple de ces vulnérabilité détecté par sonar , il donne la ligne et la suggestion , ainsi que la cause qui est motionné ci dessous que les attaquateurs peut contrôler le filesystem path argument.

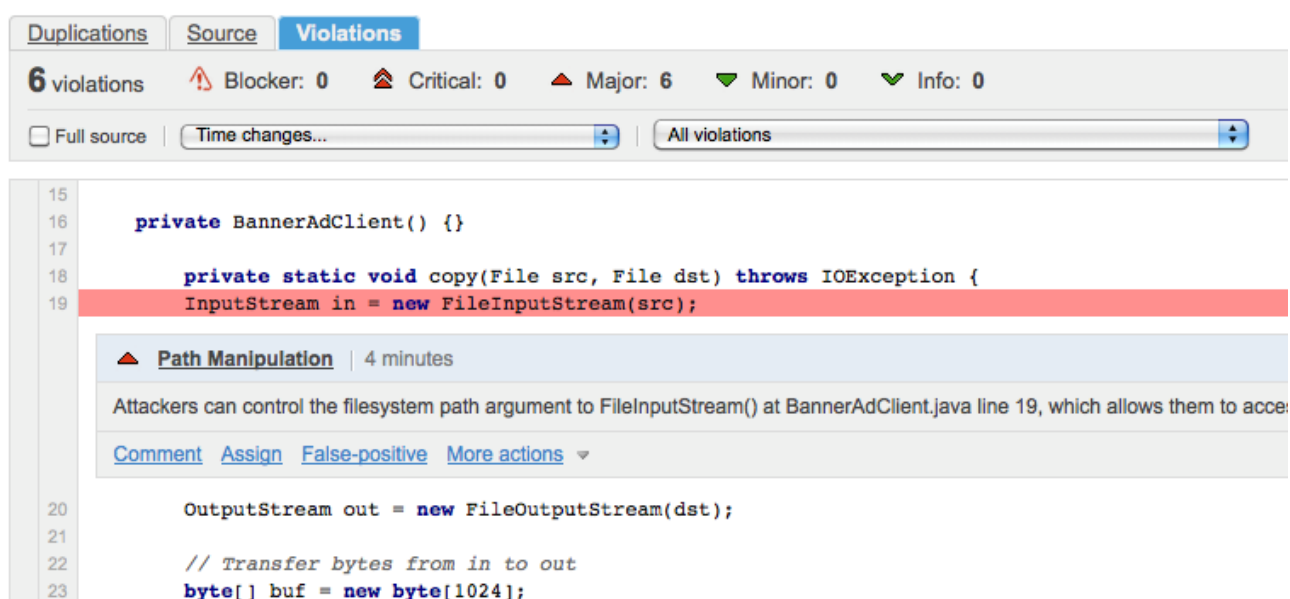


FIGURE 4.26 – Capture d'un exemple de vulnérabilité

Conclusion

Ce chapitre a décrit l'architecture technique du futur système et cité les différents outils et frameworks adoptés, on a présenté un aperçu sur quelques interfaces réalisées permettant l'interaction entre les utilisateurs et le système.

CONCLUSION GÉNÉRALE

Le projet de fin d'études, effectué au sein de la société CGI Maroc, consistait à mettre en place et alimenter une portail D3C par les données de facturation FAE.

Au terme de ce travail, on a pu implémenter la majeure partie des fonctionnalités du système cible. Pour atteindre cet objectif, une étude de l'existant et une extraction de ses faiblesses ont été réalisées, et les besoins fonctionnels auxquels le système futur devrait répondre ont été identifiés. on a ensuite entamé une étude conceptuelle globale selon la méthodologie SCRUM. L'étude technique nous a permis de décider les outils à utiliser lors de la réalisation de la solution.

Ensuite, on a présenté à notre client Michelin quelques maquettes pour se mettre en accord et s'assurer de leur satisfaction, puis dans la partie réalisation nous avons présenté différents interfaces graphiques accompagnées de leurs descriptions.

Malgré toutes les difficultés rencontrées au niveau du processus de développement et les contraintes de temps, nous avons réussi à réaliser la totalité de notre application tout en respectant l'aspect sécuritaire.

Par ailleurs, ce projet était une occasion intéressante pour raffiner mes capacités d'abstraction et de conception. En outre, ce projet m'a permis d'approfondir mes connaissances techniques et d'acquérir un esprit rigoureux d'analyse permettant d'apporter des solutions convenables aux problèmes rencontrés.

En termes de perspectives, nous avons proposés à notre client d'améliorer l'application en réalisant notre propre passerelle entre le portail D3C et le sys-générateur, et ce afin de faciliter l'envoi du fichier XML. Il est à signaler qu'un deuxième lot est programmé en perspectives dont le but est de gérer divers services dans la société (incidents, problèmes...).

ANNEXE

BLOC 1 - DIST - L = 500							
Champ		LG	Début	Fin	Type	Valeur / Description	XML
CAR02_1		2	1	2	X	"01"	
STE_COM_C	Code société commerciale	3	3	5	X	FUASFT00.STE_COM_C	DIST.STE-COM-C
FACN_PCC_ID	Identifiant PCC de facturation	8	6	13	X	FUASFT00.FACN_PCC_ID	DIST.FACN-PCC-ID
FACN_PCC_RS	Raison sociale	36	14	49	X	FUASFT00.FACN_PCC_RS	DIST.FACN-PCC-RS
FACN_ADR_1	Adresse 1	100	50	149	X	FUASFT00.FACN_ADR_1	DIST.FACN-ADR-1
FACN_ADR_2	Adresse 2	100	150	249	X	FUASFT00.FACN_ADR_2	DIST.FACN-ADR-2
FACN_PAYS_LIB	Date d'émission du document vers LISIFACT	36	250	285	X	FUPAYT00.PAYS_LIB	DIST.FACN-PAYS-LIB
FILLER		215	286	500	X		
BLOC 2 -DOC - L = 500							
Champ		LG	Début	Fin	Type	Valeur / Description	XML
CAR02_1		2	1	2	X	"02"	
DOC_TYP	Type de document	1	3	3	X	FUTDOT00.DOC_TYP	DIST.DOC.DOC-TYP
DOC_TYP_LIB	Libellé du type de document	11	4	14	X	FUTDOT00.DOC_TYP_LIB	DIST.DOC.DOC-TYP-LIB
DOC_NO	Numéro du document	11	15	25	X	FUDOCT00.DOC_NO	DIST.DOC.DOC-NO
DOC_CRE_DT	Date de création du document	10	26	35	X	FUDOCT00.DOC_CRE_DT	DIST.DOC.DOC-CRE-DT
DEV_C	Code devise	3	36	38	X	FUDOCT00.DEV_C	DIST.DOC.DOC-DEV-C
DOC_IMPT_DT	Date d'imputation du document	10	39	48	X	FUDOCT00.DOC_IMPT_DT	DIST.DOC.DOC-IMPT-DT
DOC_MT_TAX_SEUL		16	64	79	X		
PMT_TAX_IDR		1	80	81	X		
FILLER		435		500	X		

FIGURE 4.27 – Structure fichier plat : bloc 1 et 2

BLOC 3 - BRD - L = 500							
Champ		LG	Début	Fin	Type	Valeur / Description	XML
CAR02_1		2	1	2	X	"03"	
APP_C	Code application	3	3	5	X	FUBRDT00.APP_C	DIST.DOC.BRD.APP-C
EMI_L_IDR	Indicateur de lieu d'émission	4	6	9	X	FUBRDT00.EMI_L_IDR	DIST.DOC.BRD.EMI-L-IDR
BORD_NO	Numéro du bordereau	11	10	20	X	FUBRDT00.BORD_NO	DIST.DOC.BRD.CORD-NO
BORD_I	Indice du bordereau	1	21	21	X	FUBRDT00.BORD_I	DIST.DOC.BRD.BORD-I
BORD_VERS_NO	Numéro de version du bordereau	2	22	23	X	FUBRDT00.BORD_VERS_NO	DIST.DOC.BRD.BORD-VERS-NO
REF_BORD_DT	Date de référence du bordereau	10	24	33	X	FUBRDT00.REF_BORD_DT	DIST.DOC.BRD.REF-BORD-DT
ORI_DOC_NO	Numéro du document d'origine	11	34	44	X	FUBRDT00.DOC_NO FUBRDT00.ORI_DOC_NO	DIST.DOC.BRD.ORI-DOC-NO
DDT_CD_LIV_NO		3	45	47			
DDT_NO	Numéro de document de transport	6	48	53	X	FUBRDT00.DDT_NO	DIST.DOC.BRD.DDT-NO
DDT_DT	Date du DDT	10	54	63	X	FUBRDT00.DDT_DT	DIST.DOC.BRD.DDT-DT
AVIEXP_NO		35	64	98			
CDE_DT	Date de la commande	10	99	108	X	FUBRDT00.CDE_DT	DIST.DOC.BRD.CDE-DT
CDE_NO	Numéro de la commande	7	109	115	X	FUBRDT00.CDE_NO	DIST.DOC.BRD.CDE-NO
CD_PDC_NO		3	116	118	X		
CLT_CDE_C	Code commande du client	16	119	134	X	FUBRDT00.CLT_CDE_C	DIST.DOC.BRD.CLT-CDE-C
LIV_PCC_ID	Identifiant PCC de livraison	8	135	142	X	FUBRDT00.LIV_PCC_ID	DIST.DOC.BRD.LIV-PCC-ID
LIV_RS		36	143	178	X	FUBAET00.LIV_EXCP_RS	DIST.DOC.BRD.LIV-RS
LIV_ADR_1		100	179	278	X	FUBAET00.LIV_EXCP_ADR_1	DIST.DOC.BRD.LIV-ADR-1
LIV_ADR_2		100	279	378	X	FUBAET00.LIV_EXCP_ADR_1	DIST.DOC.BRD.LIV-ADR-2
LIV_PAYS_LIB		36	379	414	X	FUPAYT00.PAYS_LIB	DIST.DOC.BRD.LIV-PAYS-LIB
FILLER		86		500	X		

FIGURE 4.28 – Structure fichier plat : bloc 3

BLOC 4 - LBD - L = 500							
Champ		LG	Début	Fin	Type	Valeur / Description	XML
CAR02_1		2	1	2	X	"04"	
BORD_LG_NO	Numéro de ligne du bordereau	4	3	6	X	FULBDT00.BORD_LG_NO	DIST.DOC.BRD.CAI.BORD-LG-NO
CAI_ID	Code article international	6	7	12	X	FULBDT00.CAI_ID	DIST.DOC.BRD.CAI.CAI-ID
MRQ_LIB	Libellé marque	36	13	48	X	FUMRQT00.MRQ_LIB	DIST.DOC.BRD.CAI.MRQ-LIB
LPR_CLASSF_C	Code classification LPR	3	49	51	X	FUARCT00.LPR_CLASSF_C	DIST.DOC.BRD.CAI.LPR-CLASSF-C
CAI_LPR_LIB	Libellé classification LPR	36	52	87	X	FULPRT00.CAI_LPR_LIB	DIST.DOC.BRD.CAI.CAI-LPR-LIB
CAI_COM_DES	Désignation commerciale du CAI	36	88	123	X	UARCT00.CAI_COM_DES	DIST.DOC.BRD.CAI.CAI-COM-DES
ENS_NO	Numéro de l'ensemble	2	124	125	X	FULBDT00.ENS_NO	DIST.DOC.BRD.CAI.ENS-NO
ENS_TYP	Type d'ensemble	1	126	126	X	FUARCT00.ENS_TYP	DIST.DOC.BRD.CAI.ENS-TYP
PRD_LG_C	Code ligne produit	2	127	128	X	FULBDT00.PRD_LG_C	DIST.DOC.BRD.CAI.PRD-LG-C
CAC_NO	Numéro de la CAC	35	129	163	X	FULBDT00.CAC_NO	DIST.DOC.BRD.CAI.CAC-NO
FAE_EDI_C	Code EDI	35	164	198	X	FUARCT00.FAE_EDI_C	DIST.DOC.BRD.CAI.FAE-EDI-C
LG_REF_QT	Quantité de référence de la ligne	4	199	202	X	FULBDT00.LG_REF_QT	DIST.DOC.BRD.CAI.LG-REF-QT
BAS_PX	Prix de bas modifiée	13	203	215	X	FULBDT00.BAS_PX	DIST.DOC.BRD.CAI.BAS-PX
LG_PX_UNIT_HT	Prix unitaire hors taxe de la ligne	13	216	228	X	FULBDT00.LG_PX_UNIT_HT	DIST.DOC.BRD.CAI.LG-PX-UNIT-HT
LG_MT_HT		13	229	241			
LG_MT_TTC		13	242	254			
LG_AVT_MT_HT		13	255	267			
ENVT_CTB_MT_HT		13	268	280			
ENVT_CTB_LG_MT_TAX		13	281	293			
FILLER		207		500	X		

FIGURE 4.29 – Structure fichier plat : bloc 4

BLOC 5 - ECH - L = 500							
Champ		LG	Début	Fin	Type	Valeur / Description	XML
CAR02_1		2	1	2	X	"05"	
ECH_MT	Montant TTC de l'échéance	16	3	18	X	FUEDOT00.ECH_TTC_MT	DIST.DOC.ECH.ECH-TTC-MT
ECH_DT	Date de l'échéance	10	19	28	X	FUEDOT00.ECH_DT	DIST.DOC.ECH.ECH-DT
FILLER		472		500	X		
BLOC 6 - TAXE - L = 500							
Champ		LG	Début	Fin	Type	Valeur / Description	XML
CAR02_1		2	1	2	X	"06"	
TAX_COND_LIB	Libellé de la condition dans la langue	36	3	38	X	FUTCLT00.COND_LIB	DIST.DOC.BRD.CAI.TAXE.TAX-COND-LIB
TAX_MT	Montant de la taxe	16	39	54	X	FUTCLT00.TAX_MT	DIST.DOC.BRD.CAI.TAXE.TAX-MT
TAX_TX		6	55	60	X		
FILLER		440		500	X		

FIGURE 4.30 – Structure fichier plat : bloc 5 et 6

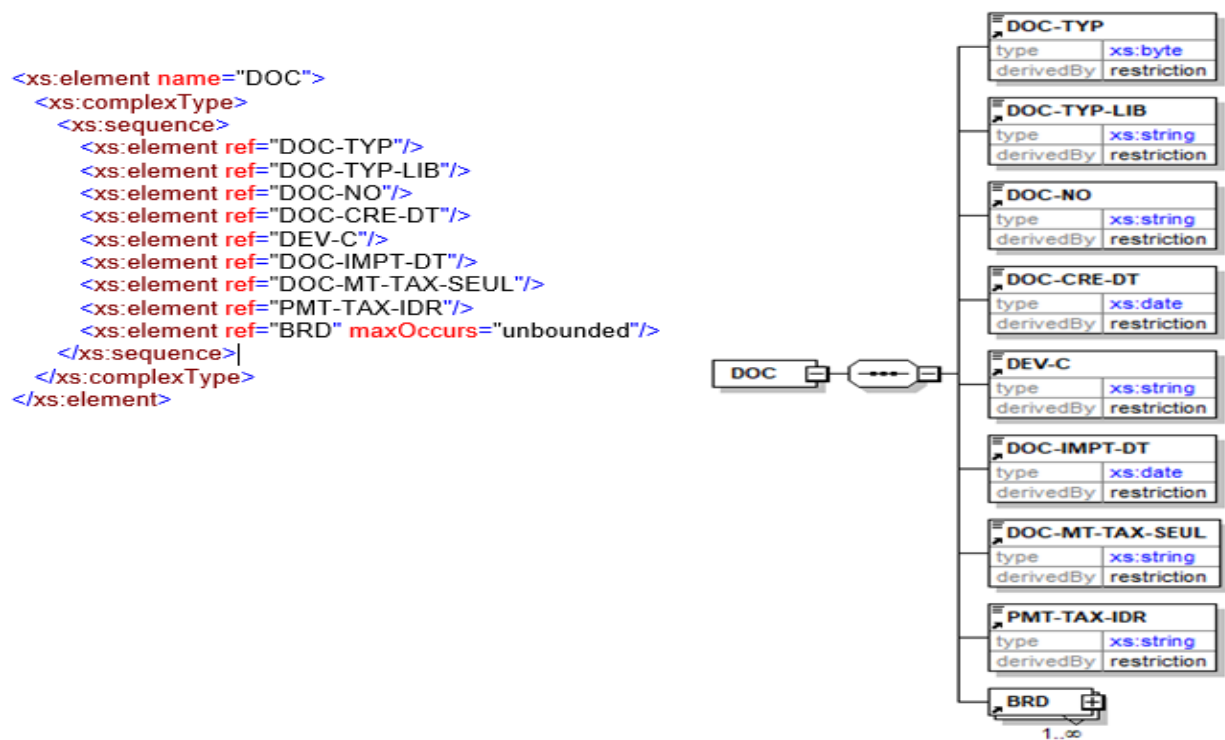


FIGURE 4.31 – Le XSD DOC

```

<xs:element name="BRD">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="APP-C"/>
      <xs:element ref="EMI-L-IDR"/>
      <xs:element ref="BORD-NO"/>
      <xs:element ref="BORD-I"/>
      <xs:element ref="BORD-VERS-NO"/>
      <xs:element ref="REF-BORD-DT"/>
      <xs:element ref="ORI-DOC-NO"/>
      <xs:element ref="DDT-CD-LIV-NO"/>
      <xs:element ref="DDT-NO"/>
      <xs:element ref="DDT-DT"/>
      <xs:element ref="AVIEXP-NO"/>
      <xs:element ref="CDE-DT"/>
      <xs:element ref="CDE-NO"/>
      <xs:element ref="CD-PDC-NO"/>
      <xs:element ref="CLT-CDE-C"/>
      <xs:element ref="LIV-PCC-ID"/>
      <xs:element ref="LIV-RS"/>
      <xs:element ref="LIV-ADR1"/>
      <xs:element ref="LIV-ADR2"/>
      <xs:element ref="LIV-PAYS-LIB"/>
      <xs:element ref="CAI" maxOccurs="unbounded"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

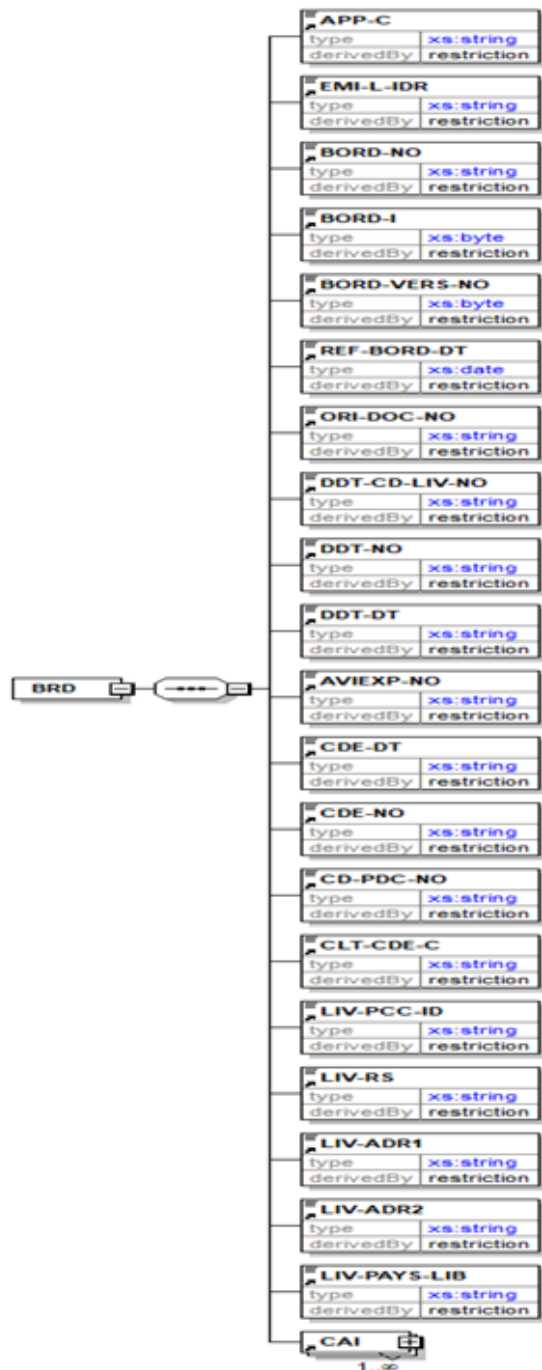


FIGURE 4.32 – Le XSD BRD

```

<xs:element name="CAI">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="BORD-LG-NO"/>
      <xs:element ref="CAI-ID"/>
      <xs:element ref="MRQ-LIB"/>
      <xs:element ref="LPR-CLASSF-C"/>
      <xs:element ref="CAI-LPR-LIB"/>
      <xs:element ref="CAI-COM-DES"/>
      <xs:element ref="ENS-NO"/>
      <xs:element ref="ENS-TYP"/>
      <xs:element ref="PRD-LG-C"/>
      <xs:element ref="CAC-NO"/>
      <xs:element ref="EAN-C"/>
      <xs:element ref="LG-REF-QT"/>
      <xs:element ref="BAS-PX"/>
      <xs:element ref="LG-PX-UNIT-HT"/>
      <xs:element ref="LG-MT-HT"/>
      <xs:element ref="LG-MT-TTC"/>
      <xs:element ref="LG-AVT-PX-UNIT-HT"/>
      <xs:element ref="LG-AVT-MT-HT"/>
      <xs:element ref="ENVT-CTB-MT-HT"/>
      <xs:element ref="ENVT-CTB-LG-MT-TAX"/>
      <xs:element ref="ECH"/>
      <xs:element ref="TAXE"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

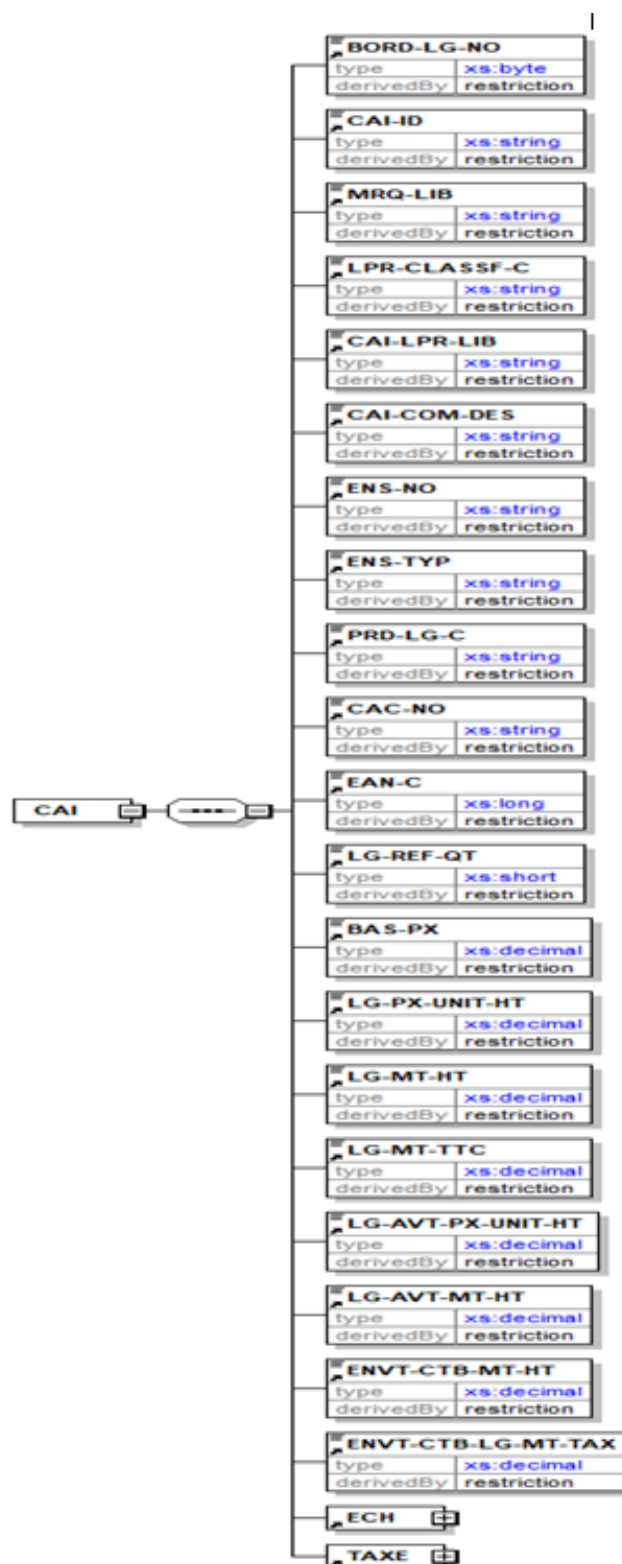


FIGURE 4.33 – Le XSD CAI

```

<xs:element name="ECH">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="ECH-MT"/>
      <xs:element ref="ECH-DT"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

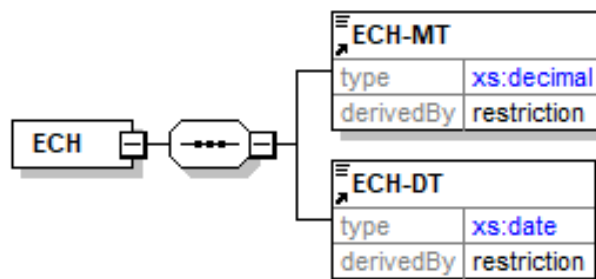


FIGURE 4.34 – Le XSD ECH

```

<xs:element name="TAXE">
  <xs:complexType>
    <xs:sequence>
      <xs:element ref="TAX-COND-LIB"/>
      <xs:element ref="TAX-MT"/>
      <xs:element ref="TAX-TX"/>
    </xs:sequence>
  </xs:complexType>
</xs:element>

```

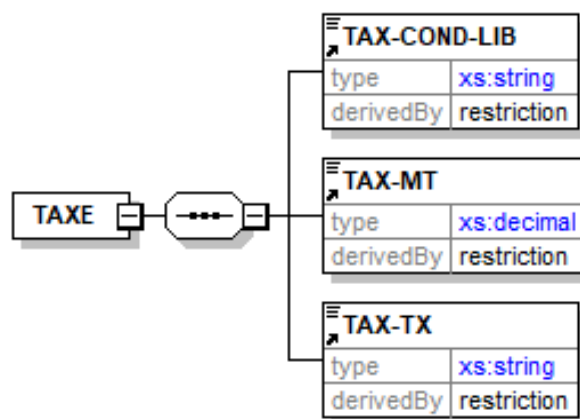


FIGURE 4.35 – Le XSD TAXE

Webographie

- [Ref1] <https://www.planzone.fr/blog/methodologies-gestion-projet>
- [Ref2] <https://fr.slideshare.net/ayoubDAMIR/rapport-de-stage-pfe-odoo-8-51637245>
- [Ref3] <https://fr.slideshare.net/gcollic/introduction-aux-mthodes-agiles-15590965>
- [Ref4] <https://medium.com/les-cahiers-agiles/au-del%C3%A0-de-l%C3%A9quipe-scrum-les-r%C3%B4les-dans-l-entreprise-agile-961d737a9e28>
- [Ref5] <http://www.blog-gestion-de-projet.com/comment-appliquer-le-diagramme-ishikawa-a-la-gestion-de-projet/>
- [Ref6] <https://www.labri.fr/perso/guibert/DocumentsEnseignement/UML.pdf>
- [Ref7] [http://laurent-audibert.developpez.com/Cours-UML/html/CoursUML.html#toc179\(28/03/2017\)](http://laurent-audibert.developpez.com/Cours-UML/html/CoursUML.html#toc179(28/03/2017))
- [Ref8] <https://www.modeliosoft.com/fr/technologies/bpmn.html>
- [Ref9] https://fr.wikiversity.org/wiki/Mod%C3%A9lisation_UML/Le_diagramme_de_classes
- [Ref10] <https://laurent-audibert.developpez.com/Cours-UML/?page=diagrammes-composants-deploiement>
- [Ref11] https://fr.wikipedia.org/wiki/Mod%C3%A9lisation_UML
- [Ref12] <https://staff.info.unamur.be/dbm/publication/1996/inforsid96.pdf>
- [Ref13] <http://www.labri.fr/perso/billaud/IUT/cobol.html>
- [Ref14] <https://searchdatacenter.techtarget.com/definition/JCL>
- [Ref15] <https://www.jmdoudoux.fr/java/dej/chap-j2ee-javaee.htm>
- [Ref16] <http://glossaire.infowebmaster.fr/xml/>
- [Ref17] <http://www.opentuto.com/quest-ce-que-le-framework-spring-2/>
- [Ref18] http://gardeux-vincent.eu/Documents/ProjetJEE/BVC_Hibernate_struts2/Hibernate/index.html
- [Ref19] <http://www.memoirepfe.fst-usmba.ac.ma/get/pdf/1314>
- [Ref20] <https://cyrille-herby.developpez.com/tutoriels/java/mapper-sa-base-donnees-avec-pattern-dao/>

[Ref21] <https://fr.slideshare.net/tommydonSalieri/rapport-sfe-med-aziz-chetoui-dhouha-melki>

[Ref22] https://lig-membres.imag.fr/plumejeaud/NFE107-fichesLecture/webserver-tomcat-jboss-jrun-jonas_DOC.pdf,

[Ref23] <https://www.irif.fr/~carton/Enseignement/InterfacesGraphiques/MasterInfo/-Cours/Swing/mvc.html>

[Ref24] https://www.tutorialspoint.com/db2/db2_introduction.htm

[Ref25] <http://www.wampserver.com/>

[Ref26] <https://fr.wikipedia.org/wiki/PhpMyAdmin>

[Ref27] <https://balsamiq.com/wireframes/>

[Ref28] <https://www.powerdesigner.biz/FR/powerdesigner/powerdesigner-features.html>

[Ref29] <https://www.supinfo.com/articles/single/6418-qu-est-ce-que-sonarque>