

REPUBLIQUE ALGERIENNE DEMOCRATIQUE ET POPULAIRE  
MINISTERE DE L'ENSEIGNEMENT SUPERIEUR ET DE LA RECHERCHE SCIENTIFIQUE  
UNIVERSITE ABDERRAHMANE MIRA DE BEJAÏA  
FACULTE DES SCIENCES EXACTES  
DEPARTEMENT D'INFORMATIQUE



Mémoire de Fin de Cycle

En vue de l'obtention du diplôme de Master Professionnel en Informatique

Option : Génie Logiciel

Thème

---

Conception et réalisation d'une application Desktop de  
gestion pour la section RELEX de SONATRACH – Béjaïa

---

Réalisé par

M. TAGUILSA Rafik

Mlle. TIGHIDET Messilva

Soutenu le 04/07/2023 devant le jury composé de :

Président	Mr SIDER Abderrahmane
Examineur	Mme HAMZA Lamia
Encadrant	Mme AIT HACÈNE Souhila

Année Universitaire : 2022 — 2023

## ***Remerciements***

*Nous exprimons notre sincère gratitude envers notre encadrante, Mme. AIT HACENE Souhila, pour ses conseils inestimables, sa patience et son soutien constants tout au long du processus. Sa passion et son expertise ont été une source d'inspiration et ont grandement contribué à l'enrichissement de ce travail.*

*Nous souhaitons également remercier chaleureusement M. TIAB Ahmed, notre encadrant au sein de l'entreprise SONATRACH, pour nous avoir intégré et pour son appui tout au long de notre parcours.*

*Nous tenons à exprimer notre profonde reconnaissance envers M. KACHER Reda, responsable de la gestion relex, pour la confiance qu'il nous a accordée et pour le temps précieux qu'il nous a consacré tout au long de cette période.*

*Nous aimerions également adresser nos vifs remerciements aux membres du jury pour l'intérêt qu'ils ont porté à cette recherche, en acceptant d'examiner ce modeste travail, et pour leurs remarques constructives qui contribueront à enrichir cette étude.*

*Enfin, nous souhaitons remercier toute l'équipe pédagogique du département d'informatique de l'université de Bejaia pour leur contribution à l'enrichissement de nos connaissances.*

# *Dédicaces*

*Je dédie humblement ce modeste travail :*

*À ma chère mère, qui m'a soutenu inconditionnellement tout au long de mes études, et  
qui a été une source de force et d'inspiration.*

*À la mémoire précieuse de mon père, dont la bienveillance et les valeurs restent  
gravées en moi.*

*À ma petite sœur Meriem, pour sa présence constante, ses encouragements  
chaleureux et son affection.*

*À mes deux grands-mères, que Dieu leur accorde une longue vie, pour leur soutien et  
leurs prières incessantes. À mes tantes, oncles, cousins et cousines, qui ont toujours  
été présents et ont contribué à mon épanouissement. À mes amis, en particulier  
Massilia, Manel, Mélissa, Rahma, Maïssa Dina, et Houda, pour leur amitié sincère,  
leur soutien indéfectible et les moments inoubliables partagés. Enfin, à tous ceux qui  
m'entourent et qui ont apporté leur aide et leur appui, je vous suis reconnaissante à  
jamais.*

***Messilva.***

# *Dédicaces*

*Je dédie humblement ce modeste travail :*

*À ma chère mère, qui m'a soutenu inconditionnellement tout au long de mes études, et qui a été une source de force et d'inspiration. À la mémoire éternelle de mon père, dont le souvenir inspire mon chemin chaque jour.*

*À mes chers frères Nassim et Rami, ainsi qu'à ma sœur Lydia, pour leur présence constante, leurs encouragements chaleureux et leur affection.*

*À mes tantes, oncles, cousins et cousines, qui ont toujours été présents et ont contribué à mon épanouissement. À mes amis, en particulier Aris, Abdallah, Walid, Sami, Cherif, Merouane, Anis, Nassim, Idir et Youba pour leur amitié sincère, leur soutien indéfectible et les moments inoubliables partagés. Je vous aime tous infiniment et je vous remercie du fond du cœur pour tout ce que vous faites pour moi. Que nos liens restent forts et que notre bonheur soit toujours partagé.*

***Rafik.***

# Table des matières

<b>Table des figures</b>	<b>viii</b>
<b>Liste des tableaux</b>	<b>xi</b>
<b>Liste des abréviations</b>	<b>xii</b>
<b>Introduction générale</b>	<b>1</b>
<b>1 Étude préliminaire et analyse de l'existant</b>	<b>3</b>
1.1 Introduction . . . . .	3
1.2 Présentation de l'organisme d'accueil SONATRACH . . . . .	3
1.2.1 Historique de la SONATRACH . . . . .	4
1.2.2 Missions de la SONATRACH . . . . .	4
1.2.3 Organigramme de la SONATRACH . . . . .	5
1.2.4 Présentation de l'activité Transport par Canalisation (TRC) . . . . .	6
1.2.5 Présentation de la Région de Transport Centre (RTC) . . . . .	7
1.2.6 Présentation du département Moyens Généraux . . . . .	8
1.2.7 Présentation de la section concernée par l'étude : La section RELEX . . . . .	9
1.3 Étude de l'existant . . . . .	9
1.3.1 Processus de gestion documentaire suivi par la section RELEX . . . . .	10
1.3.1.1 Les documents manipulés . . . . .	10
1.3.1.2 Le cheminement des documents . . . . .	11
1.4 Critique de l'existant . . . . .	12
1.5 Solution proposée . . . . .	13

1.6	Conclusion . . . . .	14
<b>2</b>	<b>Analyse et conception</b>	<b>15</b>
2.1	Introduction . . . . .	15
2.2	Spécification des besoins . . . . .	15
2.2.1	Besoins fonctionnels . . . . .	15
2.2.2	Besoins non fonctionnels . . . . .	16
2.3	Identification des acteurs du système . . . . .	16
2.4	Conception . . . . .	17
2.4.1	Processus de développement . . . . .	17
2.4.1.1	Présentation de SCRUM . . . . .	17
2.4.1.2	Piliers de SCRUM . . . . .	18
2.4.1.3	Les rôles SCRUM . . . . .	18
2.4.1.4	Concepts de SCRUM . . . . .	19
2.4.1.5	La mise en œuvre de SCRUM . . . . .	21
2.4.2	Langage de modélisation UML . . . . .	23
2.4.2.1	Identification des cas d'utilisation . . . . .	23
2.4.2.2	Diagramme de cas d'utilisation . . . . .	26
2.4.2.3	Description textuelle des cas d'utilisation . . . . .	29
2.4.2.4	Diagramme de séquence . . . . .	33
2.4.2.5	Dictionnaire de données . . . . .	38
2.4.2.6	Diagramme de classes . . . . .	41
2.4.2.7	Passage du diagramme de classes au modèle relationnel . . . . .	43
2.5	Conclusion . . . . .	45
<b>3</b>	<b>Réalisation</b>	<b>46</b>
3.1	Introduction . . . . .	46
3.2	Outils de développement . . . . .	46
3.2.1	Environnements de développement . . . . .	46
3.2.1.1	Visual Studio Code . . . . .	46

3.2.1.2	DB Browser for SQLite . . . . .	47
3.2.1.3	Visual Paradigm . . . . .	47
3.2.1.4	Adobe Illustrator . . . . .	47
3.2.2	Langages de programmation . . . . .	48
3.2.2.1	HTML . . . . .	48
3.2.2.2	CSS . . . . .	48
3.2.2.3	JavaScript . . . . .	48
3.2.2.4	SQL . . . . .	48
3.2.3	Environnement d'exécution et Frameworks . . . . .	49
3.2.3.1	Node.js . . . . .	49
3.2.3.2	Electron JS . . . . .	49
3.2.3.3	Bootstrap . . . . .	50
3.2.4	Bibliothèques utilisées . . . . .	50
3.2.4.1	SweetAlert2 . . . . .	50
3.2.4.2	SQLite3 . . . . .	50
3.2.4.3	Font Awesome . . . . .	50
3.2.4.4	Mammoth . . . . .	51
3.2.4.5	Docxtemplater . . . . .	51
3.2.4.6	PizZip . . . . .	51
3.2.5	Outils de déploiement . . . . .	51
3.2.5.1	Electron Packager . . . . .	51
3.2.5.2	Inno Setup Compiler . . . . .	51
3.3	Présentation de quelques interfaces de l'application . . . . .	52
3.3.1	L'interface d'authentification . . . . .	52
3.3.2	L'interface d'accueil utilisateur . . . . .	53
3.3.3	L'interface de génération d'un bon de commande . . . . .	54
3.3.3.1	L'interface de génération automatique d'un bon de commande . . . . .	55
3.3.3.2	L'interface de génération manuelle d'un bon de commande . . . . .	56
3.3.4	L'interface de la liste des bons de commande . . . . .	59

3.3.5	L'interface d'édition d'un bon de commande . . . . .	60
3.3.6	L'interface d'établissement d'une facture . . . . .	61
3.3.7	L'interface de la liste des factures . . . . .	62
3.3.8	L'interface d'édition d'une facture . . . . .	63
3.3.9	L'interface du reporting . . . . .	64
3.3.10	L'interface de gestion des établissements . . . . .	66
3.3.11	L'interface de gestion des missionnaires . . . . .	67
3.3.12	L'interface de gestion du profil utilisateur . . . . .	68
3.4	Conclusion . . . . .	69
	<b>Conclusion générale</b>	<b>70</b>
	<b>Bibliographie</b>	<b>72</b>
	<b>Annexes</b>	<b>75</b>

# Table des figures

1.1	Organigramme de la SONATRACH [1]	5
1.2	Activités de la SONATRACH [1]	6
1.3	Organigramme de la Région Transport Centre [1]	7
1.4	Organigramme du département MOG	9
2.1	Méthode de gestion de projet SCRUM [5]	20
2.2	User stories de notre application.	21
2.3	Planification des releases.	22
2.4	Les type de diagrammes d'UML.	23
2.5	Diagramme de cas d'utilisation " Gestion du profil "	26
2.6	Diagramme de cas d'utilisation " Consulter la liste des missionnaires "	26
2.7	Diagramme de cas d'utilisation " Consulter la liste des établissements "	27
2.8	Diagramme de cas d'utilisation "Gérer les bons de commande".	27
2.9	Diagramme de cas d'utilisation " Établir une facture "	28
2.10	Diagramme de cas d'utilisation " Gérer les rapports "	28
2.11	Diagramme de séquence du cas d'utilisation " Générer un bon de commande hôtel automatiquement "	34
2.12	Diagramme de séquence du cas d'utilisation " Générer un bon de commande hôtel manuellement "	35
2.13	Diagramme de séquence du cas d'utilisation " Établir une facture hôtel "	36
2.14	Diagramme de séquence du cas d'utilisation " Générer un rapport hôtel "	37
2.15	Diagramme de classes.	42

3.1	Logo de notre application. . . . .	52
3.2	L'interface d'authentification. . . . .	53
3.3	L'interface d'accueil utilisateur. . . . .	54
3.4	Interface du choix de la méthode de génération du bon de commande. . . . .	55
3.5	L'interface de génération automatique d'un bon de commande. . . . .	55
3.6	Interface du choix du nombre de missionnaires à prendre en charge. . . . .	56
3.7	L'interface du formulaire de la génération manuelle d'un bon de commande. . . . .	57
3.8	Exemple d'un bon de commande d'hôtel généré. . . . .	57
3.9	Exemple d'une liste de missionnaires jointe au bon de commande. . . . .	58
3.10	Liste des bons de commande. . . . .	59
3.11	L'interface d'édition d'un bon de commande. . . . .	60
3.12	L'interface d'établissement d'une facture. . . . .	61
3.13	L'interface de la liste des factures. . . . .	62
3.14	Exemple d'une liste des factures générée. . . . .	63
3.15	L'interface d'édition d'une facture. . . . .	63
3.16	L'interface du reporting. . . . .	64
3.17	Exemple de resultats d'un rapport. . . . .	65
3.18	Exemple d'un document Word d'un rapport généré. . . . .	65
3.19	L'interface de la liste des établissements. . . . .	66
3.20	L'interface d'ajout d'un établissement. . . . .	66
3.21	L'interface de la liste des missionnaires. . . . .	67
3.22	L'interface d'ajout d'un missionnaire. . . . .	67
3.23	L'interface de gestion du profil utilisateur. . . . .	68
1	L'ordre de mission, source : L'entreprise Sonatrach. . . . .	75
2	Demande de prise en charge, source : L'entreprise Sonatrach. . . . .	76
3	Demande achat titre transport, source : L'entreprise Sonatrach. . . . .	77
4	Facture d'hôtel, source : L'entreprise Sonatrach. . . . .	78
5	Facture du transport aérien commercial, source : L'entreprise Sonatrach. . . . .	79
6	Bon de commande d'hôtel, source : L'entreprise Sonatrach . . . . .	80

7	Bon de commande du transport aérien commercial, source : L'entreprise Sonatrach. . .	81
8	Bon de commande du transport aérien affrété, source : L'entreprise Sonatrach . . . . .	82

# Liste des tableaux

2.1	Les rôles SCRUM . . . . .	19
2.2	Identification des cas d'utilisation . . . . .	25
2.4	Description textuelle du cas d'utilisation " Générer un bon de commande hôtel" . . . . .	30
2.5	Description textuelle du cas d'utilisation " Établir une facture hôtel" . . . . .	31
2.6	Description textuelle du cas d'utilisation "Générer un rapport hôtel" . . . . .	32
2.7	Dictionnaire de données . . . . .	41

# Liste des abréviations

<b>BDC</b>	Bon de commande
<b>CSS</b>	Cascading Style Sheets
<b>DB4S</b>	Data Base Browser For SQLite
<b>HTML</b>	HyperText Markup Language
<b>JS</b>	JavaScript
<b>MOG</b>	Moyens Généraux
<b>RELEX</b>	Relations Exterieurs
<b>SCRUM</b>	Systematic Customer Resolution Unraveling Meeting
<b>SGBD</b>	Système de Gestion de Base de Données
<b>SQL</b>	Structured Query Language
<b>TAA</b>	Transport Aérien Affrété
<b>TAC</b>	Transport Aérien Commercial
<b>UML</b>	Unified Modeling Language

# Introduction générale

L'informatique joue un rôle crucial dans tous les domaines de nos sociétés modernes. Son importance est indéniable et se fait ressentir dans de nombreux secteurs. Notamment, dans le fonctionnement des entreprises, en particulier dans le secteur industriel. Grâce aux avancées technologiques, elle a révolutionné les opérations commerciales, stimulant ainsi la productivité et l'efficacité. Au sein des entreprises, l'informatique offre des outils et des systèmes automatisés qui simplifient les tâches quotidiennes. Les logiciels de gestion d'entreprise permettent de superviser efficacement les opérations, de suivre les stocks, de gérer les ressources humaines et de prendre des décisions éclairées basées sur des données précises. Les systèmes de planification des ressources d'entreprise (ERP) centralisent les informations et facilitent la communication interne, améliorant ainsi la coordination entre les différents départements. Dans le cadre de l'amélioration continue des solutions informatiques au sein des organisations, nous avons réalisé un stage au sein de la section RELEX du service intérieur de SONATRACH. En tant que société nationale algérienne, SONATRACH se consacre à la recherche, à l'exploitation, au transport par canalisation, à la transformation et à la commercialisation des hydrocarbures et de leurs dérivés. Sa principale mission consiste à valoriser de manière optimale les ressources nationales d'hydrocarbures et à générer des richesses au service du développement économique et social du pays. L'objectif de ce mémoire était de proposer des solutions concrètes et pratiques pour améliorer l'efficacité, la qualité et la sécurité du processus de gestion au sein de la section RELEX de SONATRACH. À travers une étude approfondie de l'existant, une modélisation précise des besoins et la réalisation d'une solution informatique adaptée afin de contribuer à l'amélioration continue des processus de gestion et de soutenir le développement économique et social de l'entreprise.

Ainsi le présent mémoire est organisé de la manière suivante :

Chapitre 1 : Intitulé : « Etude préliminaire et analyse de l'existant », concerne la présentation de l'organisme SONATRACH et leurs différentes activités, l'étude de l'existant en particulier le processus de gestion documentaire, suivi des critiques et problématique liée au cas d'étude, et enfin la solution proposée.

Chapitre 2 : Intitulé « Analyse et conception », présente les besoins fonctionnels et non-fonctionnels, les processus de modélisation et de développement choisis, ainsi que les différentes phases de la conception.

Chapitre 3 : Intitulé «Réalisation», désigne la phase de concrétisation de notre application. Ce chapitre comprend la présentation de l'environnement de développement et les outils utilisés, et enfin la présentation des interfaces de l'application. Pour conclure, nous achevons ce mémoire par une conclusion générale des éléments clés de notre travail et en exposant les perspectives envisagées pour notre application.

# Chapitre 1

## Étude préliminaire et analyse de l'existant

### 1.1 Introduction

Dans le cadre de notre travail, nous entreprenons une étude préliminaire et approfondie de la section RELEX au sein de l'entreprise d'accueil SONATRACH. Une étape cruciale qui nous permettra de comprendre en détail le fonctionnement de la section, en identifiant les éventuelles lacunes et les opportunités d'amélioration. L'objectif principal de cette étude est de dresser un état des lieux complet en examinant de près les processus de travail, les documents manipulés par la section RELEX et le cheminement des informations. Cette démarche nous permettra de définir précisément la problématique à résoudre et de dresser un bilan de la situation actuelle, en recensant les anomalies constatées et en proposant les solutions appropriées.

### 1.2 Présentation de l'organisme d'accueil SONATRACH

SONATRACH «*Société Nationale pour la Recherche, la Production, le Transport, la Transformation, et la Commercialisation des Hydrocarbures et leurs dérivés* » est une entreprise publique algérienne, un acteur majeur de l'industrie pétrolière surnommé le major africain qui a pour mission fondamentale la valorisation optimale et sur le long terme des ressources nationales d'hydrocarbures en Algérie et à l'international. Outre ses activités dans le domaine du pétrole et du gaz, SONATRACH intervient également dans d'autres secteurs tels que la génération électrique, les énergies nouvelles et

renouvelables, ainsi que le dessalement d'eau de mer. Elle exerce ses métiers en Algérie et partout dans le monde, où des opportunités se présentent. La société est dirigée par un président-directeur général qui supervise l'ensemble du groupe SONATRACH, y compris ses activités opérationnelles, commerciales et de transport par canalisation [2].

### **1.2.1 Historique de la SONATRACH**

Après l'indépendance de l'Algérie, l'État a créé la Société nationale de transport et de la commercialisation des hydrocarbures, connue sous le nom de SONATRACH, le 31 décembre 1963. Dès ses débuts, SONATRACH a entrepris d'importants projets pour développer l'industrie des hydrocarbures en Algérie. En 1964, elle a construit le premier oléoduc algérien, suivi de la mise en service du premier complexe de liquéfaction de gaz naturel. Elle a également élargi ses activités en créant l'association coopérative ASCOOP en 1965, et a réalisé des découvertes majeures de pétrole et de gaz en Algérie.

En 1981, la société a été restructurée dans le cadre d'un schéma directeur approuvé pour une meilleure efficacité organisationnelle et économique. Cette restructuration a conduit à la création de 17 entreprises en 1984, telles que NAFTAL, NAFTEC, ASMIDAL, ENIP, ENPC... Ces dernières ont permis d'élargir les activités de SONATRACH [3].

### **1.2.2 Missions de la SONATRACH**

La Sonatrach a des missions variées confiées par l'état, son unique actionnaire, ainsi que des tâches spécifiques qu'elle exerce tant en Algérie qu'à l'étranger. Ses missions se résument comme suit :

- Satisfaire les besoins actuels et futurs de l'Algérie en hydrocarbures et produits pétroliers.
- Gestion complète des réseaux de transport, stockage et de chargement des hydrocarbures.
- Exploration des nouvelles opportunités à l'échelle mondiale.
- Développement international, partenariat et diversification du portefeuille d'activités.
- Étude, promotion et valorisation de toute autre forme et sources d'énergie [1].

### 1.2.3 Organigramme de la SONATRACH

Voici une représentation schématique de la structure organisationnelle de SONATRACH en Algérie, illustrant ses différentes branches opérationnelles sur le territoire national :

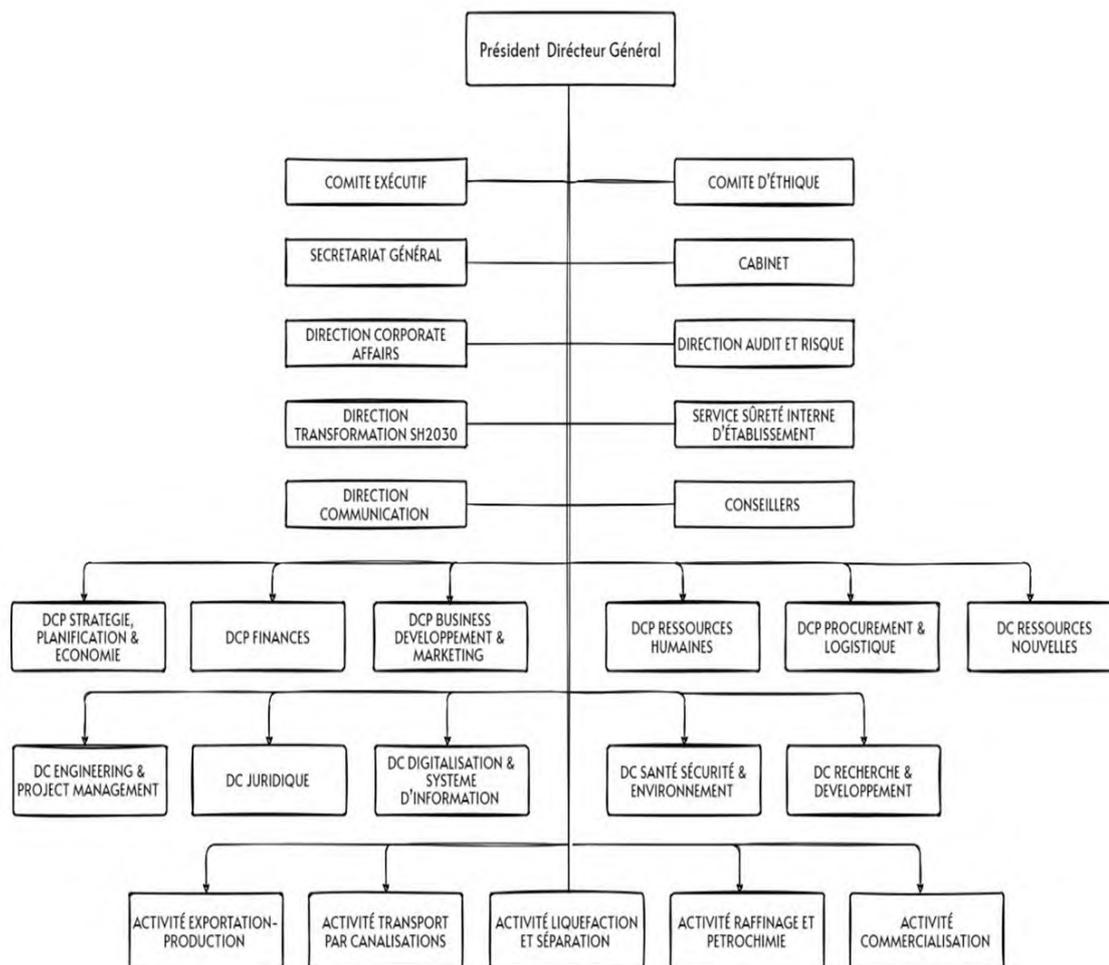


FIGURE 1.1 – Organigramme de la SONATRACH [1]

Notre stage s’est déroulé au sein de l’activité Transport par Canalisation (TRC) qui est spécifiquement exercée à Béjaïa. Cette activité constitue l’une des quatre activités principales pratiquées par SONATRACH [1].

Pour mieux visualiser ces activités, voici un schéma représentatif :

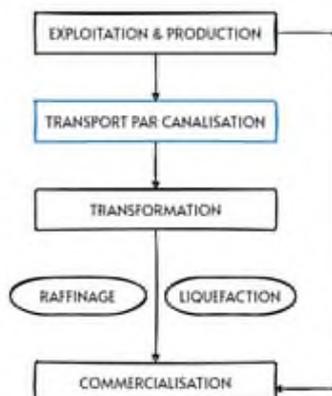


FIGURE 1.2 – Activités de la SONATRACH [1]

#### 1.2.4 Présentation de l'activité Transport par Canalisation (TRC)

L'activité Transport par Canalisation est chargée de l'exploitation des installations de transport des hydrocarbures : pétrole brut, gaz et condensant dans des conditions optimums de sécurité et de protection de l'environnement [2].

Ses principales missions sont :

- La gestion et exploitation des ouvrages concentrés et des canalisations de transport des hydrocarbures.
- La coordination et contrôle de l'exécution des programmes de transport en fonction des impératifs de production et de commercialisation
- La maintenance, intervention et protection des installations de transport
- Conduite d'études, réalisation et gestion de projets pour le développement du réseau de transport [2].

SONATRACH possède cinq régions de transport des hydrocarbures, à savoir :

- La région de transport Est 'Skikda'.
- La région de transport Centre 'Bejaia'.
- La région de transport Ouest 'Arzew'.
- La région de transport Sud (Hassi Messaoud)
- La région de transport Sud-Ouest (Hassi R'Mel) [1].

## 1.2.5 Présentation de la Région de Transport Centre (RTC)

La Région de Transport Centre (RTC) basée à Bejaïa est l'une des cinq directions régionales de la TRC chargée de l'exploitation et de la gestion des infrastructures de transport par canalisation, notamment des oléoducs et du gazoduc. Son rôle principal est d'assurer le transport efficace et sécurisé des hydrocarbures tels que le pétrole brut, le condensat et le gaz naturel depuis les pôles de production jusqu'aux zones de demande et de transformation. Grâce à son réseau étendu et à ses installations portuaires, la région facilite le chargement, le déchargement et le transport des hydrocarbures, contribuant ainsi à la stratégie globale de l'entreprise en matière de distribution et d'exportation des ressources énergétiques [4].

Notre stage s'est déroulé au sein du département Moyens Généraux, représenté en bleu dans l'organigramme d'organisation de la RTC suivant :

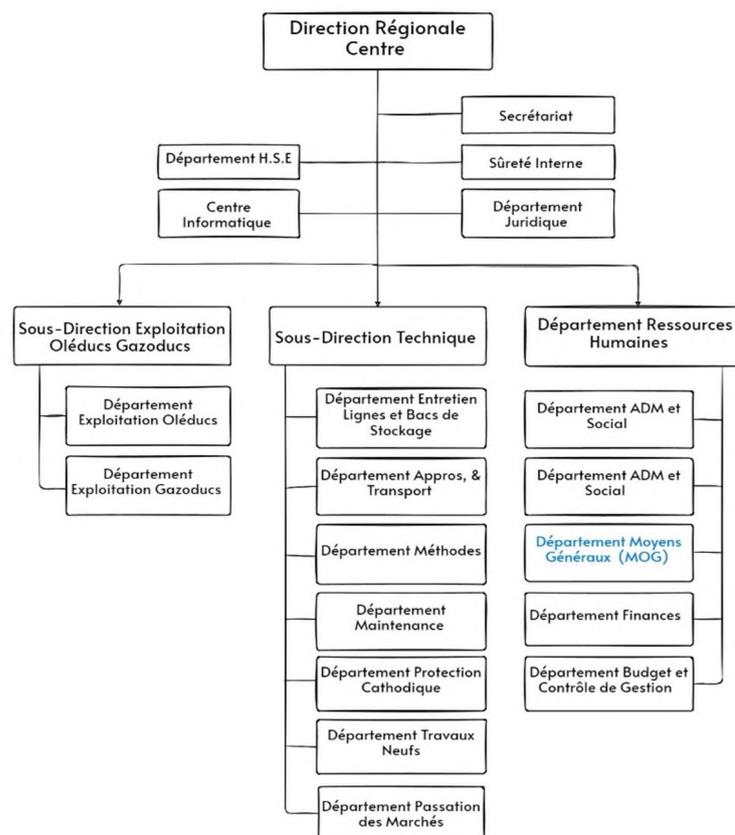


FIGURE 1.3 – Organigramme de la Région Transport Centre [1]

## **1.2.6 Présentation du département Moyens Généraux**

Le département Moyens Généraux (MOG) est l'entité de SONATRACH chargé de fournir un soutien logistique essentiel à l'organisation en répondant efficacement à ses besoins opérationnels, permettant ainsi aux autres départements de se concentrer sur leurs activités principales qui incluent :

- L'entretien des locaux, infrastructures et espaces de travail.
- L'approvisionnement en fournitures, équipements et matériels nécessaires aux activités de l'entreprise.
- L'organisation des services d'hébergement du personnel, de restauration ainsi que du transport aérien.
- La gestion des services de courrier et de reprographie.
- Le suivi des contrats de prestataires de services et de fournisseurs.

Le Département MOG se compose de trois services principaux :

- Le service Intérieur
- Le service Entretien Bâtiment
- Le service Intendance

Dans le cadre de notre étude, nous nous concentrons sur le service Intérieur qui se décline de cinq sections :

- La section Reprographie (matériel de document)
- La section Bureau d'ordre (courrier)
- La section Gestion magasin
- La section Economa
- La section RELEX

Nous focalisons spécifiquement notre analyse sur la section RELEX. Voici un schéma explicatif de son emplacement au sein du Département MOG.

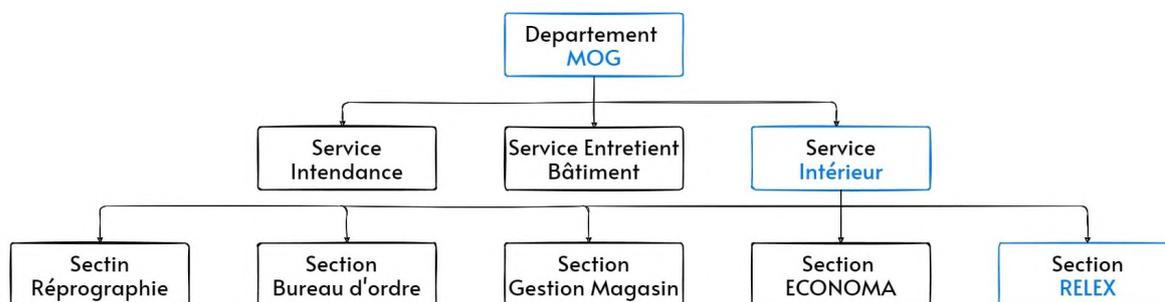


FIGURE 1.4 – Organigramme du département MOG

### 1.2.7 Présentation de la section concernée par l'étude : La section RELEX

Le section RELEX, qui est le diminutif de Relation Extérieur, se réfère aux relations entre l'organisation SONATRACH et des entités externes telles que les établissements hôteliers, les restaurants, les compagnies de transport aérien. Son rôle est d'entrer en contact avec ces établissements dans le but d'établir et de maintenir des relations avec ces partenaires externes afin de faciliter les besoins de voyage, d'hébergement, de restauration et autres services nécessaires pour les missions et les activités de l'organisation. La section RELEX est supervisée par le chargé de gestion des relations extérieures, qui a pour responsabilité de coordonner les activités liées à RELEX. Les principales fonctions de ce poste incluent la coordination de l'hébergement, de la restauration et du transport aérien des missionnaires, en collaborant avec des hôtels conventionnés et des partenaires de confiance.

## 1.3 Étude de l'existant

Cette partie de notre mémoire est dédiée à l'étude de la section RELEX de SONATRACH. Notre objectif est d'analyser en détail le fonctionnement actuel de la section, en identifiant ses faiblesses et les opportunités d'amélioration. Nous examinerons de manière approfondie la procédure de travail, les processus en place ainsi que les phases d'établissement des tâches au sein de RELEX. De plus, nous nous pencherons sur les documents manipulés, leur provenance, leur cheminement avant et après exploitation,

leur diffusion, ainsi que sur la gestion documentaire dans son ensemble. Cette analyse nous permettra de définir la problématique à résoudre en vue de proposer une solution informatique optimisée.

### **1.3.1 Processus de gestion documentaire suivi par la section RELEX**

Pour mieux comprendre le fonctionnement de la section RELEX, nous allons présenter les documents manipulés ainsi que leur cheminement.

Les références et informations utilisées dans cette présentation ont été obtenues grâce à la collaboration du responsable de la section RELEX de Sonatrach, qui nous a fourni les documents officiels.

#### **1.3.1.1 Les documents manipulés**

- Documents provenant à ce poste
  - L'ordre de mission : Il s'agit d'un document reçu de la part d'une structure interne de SONATRACH pour indiquer qu'il y a une mission à effectuer. Il peut inclure des détails tels que les personnes concernées et leur fonction, l'itinéraire, l'objet de la mission, sa durée, le moyen de transport [Annexe 1].
  - Demande de prise en charge : Ce document concerne l'hébergement et la restauration lors des missions. Il est établi par une structure interne de SONATRACH et doit être transmis au département des Moyens Généraux au moins 72 heures avant le départ des missionnaires accompagné de l'ordre de mission. Il contient des informations telles que la structure émettrice, le nom et le prénom des missionnaires, la période du séjour et il énumère les prestations qui seront prises en charge par SONATRACH pour les personnes concernées [Annexe 2].
  - Demande d'achat titre transport : Ce document concerne le transport aérien. Il est également établi par une structure interne de SONATRACH et doit être transmis au département des Moyens Généraux accompagné de l'ordre de mission. Il contient des informations telles que la structure émettrice, les personnes prises en charge, leur fonction, le type de vol, l'itinéraire, les dates et heures de départ et d'arrivée [Annexe 3].
  - Facture d'hôtel : Il s'agit d'une facture établie par l'hôtel après la prise en charge des missionnaires qui est ensuite transmise à la section RELEX pour vérification de conformité. La structure de la facture dépend de l'établissement émetteur et elle contient le numéro du bon

de commande associé ainsi que des informations concernant la facture telles que les frais à payer [Annexe 4].

— Facture du transport aérien commercial : Il s'agit d'une facture établie par la compagnie aérienne après le vol des missionnaires. Elle est également envoyée à la section RELEX pour vérification de conformité. Sa structure dépend également de la compagnie aérienne émettrice et contient le numéro du bon de commande associé ainsi que des informations concernant le vol telles que les frais du vol [Annexe 5].

• Documents sortants de ce poste :

— Bon de commande d'hôtel : Ce document est généré par le responsable de la section RELEX à partir des informations de la demande de prise en charge. Il est ensuite transmis aux établissements hôteliers afin de prendre en charge les missionnaires [Annexe 6].

— Bon de commande du transport aérien commercial : Ce document est généré par le responsable de la section RELEX à partir des informations de la demande d'achat titre transport. Il est ensuite transmis à la compagnie aérienne afin de prendre en charge les missionnaires pour leur voyage [Annexe 7].

— Bon de commande du transport aérien affrété : Ce document est généré par le responsable de la section RELEX. Contrairement au bon de commande du transport aérien commercial, il n'est pas transmis à une compagnie aérienne, car il concerne des vols organisés par l'entreprise elle-même [Annexe 8].

### **1.3.1.2 Le cheminement des documents**

Le processus de gestion documentaire au sein de la section RELEX de SONATRACH suit les étapes suivantes :

- Réception de la demande de prise en charge ou de la demande d'achat titre transport accompagné de l'ordre de mission en format papier.
- Génération du bon de commande en remplissant les informations requises extraites des documents reçus.
- Impression du document Word et remise aux signataires pour approbation.
- Transmission du bon de commande à l'hôtel, au restaurant ou à la compagnie aérienne concernée.

- Réception de la facture associée au bon de commande, vérification visuelle de sa conformité et association manuelle avec le bon de commande.

## 1.4 Critique de l'existant

Pendant notre stage au sein de la SONATRACH et en collaboration avec le responsable de la section RELEX pendant un mois, nous avons identifié trois problèmes majeurs qui affectent le fonctionnement de la section :

- **Surcharge documentaire et difficulté de gestion :**

Se traduit par une accumulation excessive de documents physiques et numériques au sein du bureau de la section RELEX. Cette surcharge documentaire entraîne des difficultés dans la recherche et l'accès aux documents pertinents, ainsi qu'une confusion générale qui peut entraîner un impact négatif sur la productivité, la réactivité et l'efficacité de la section.

- **Limitations matérielles, logicielles et saisie manuelle des informations :**

Ces limitations résultent des outils et des ressources dont dispose l'agent de la section RELEX pour accomplir ses tâches. Actuellement, L'utilisation de Word, qui est un outil général non spécifiquement adapté aux besoins de la section, entraîne une charge de travail considérable pour l'agent, ce qui entraîne une augmentation des risques d'erreurs, notamment lors de manipulations incorrectes des données. Ces erreurs peuvent avoir un impact financier important pour SONATRACH, car une seule erreur peut entraîner des dépenses considérables. De plus, la dépendance à la saisie manuelle aggrave la charge de travail des agents et peut entraîner des retards et des problèmes de gestion des informations.

- **Contraintes du processus de reporting :**

Concerne la méthode actuelle de reporting utilisée par l'agent de la section Relex lorsqu'il est confronté à des demandes de rapports de la part d'autres services de SONATRACH ou d'entités externes telles que les compagnies aériennes ou les hôtels. Actuellement, l'agent est contraint d'effectuer ces rapports manuellement, en se basant sur des calculs, des notes personnelles ou sa mémoire, ce qui augmente les risques d'erreurs et compromet la fiabilité des informations générés dans les rapports. De plus, l'absence d'un système automatisé de suivi limite sa capacité à analyser

ses actions sur une période donnée, que ce soit dans le cadre de collaborations avec des entités externes ou pour son propre suivi interne.

D'autres problématiques peuvent se manifester au sein de la section, telle que :

- Manque d'automatisation des tâches répétitives
- Problème d'accessibilité à l'information
- Manque de fiabilité des informations, notamment dans le processus de reporting

## **1.5 Solution proposée**

Pour remédier aux problèmes identifiés au sein de RELEX, il est essentiel de mettre en place une solution efficace en développant une application desktop monoposte de gestion intégrant des fonctionnalités essentielles visant à accélérer le flux de travail, de réduire les risques d'erreur et d'améliorer la productivité globale au sein de la section RELEX. Ces fonctionnalités incluent :

- Automatiser les tâches précédemment effectuées manuellement avec Word.
- Faciliter la génération, la gestion et la vérification des documents liés aux missions
- Stocker de manière efficace les informations liées aux missionnaires et aux établissements (hôtels, compagnies de transport aérien et restaurants) pour faciliter leur accès.
- Améliorer le processus de reporting en automatisant l'extraction rapide et précise des données pertinentes, et en générant des rapports automatisés et fiables.
- Assurer un accès facile et continu aux informations.

## **1.6 Conclusion**

Dans ce chapitre, nous avons effectué une analyse de l'existant en examinant à la fois l'organisme d'accueil SONATRACH et ses activités, ainsi que la section RELEX en détail. Cette analyse nous a permis de mettre en évidence les lacunes au sein de la section RELEX qui requièrent une amélioration et la nécessité de mettre en place une solution informatique adaptée aux besoins spécifiques de la section.

L'implémentation de cette solution permettra d'optimiser les processus de travail, d'accroître l'efficacité opérationnelle et de réduire les risques d'erreur. L'objectif principal est d'améliorer la productivité de la section RELEX en simplifiant la gestion documentaire et en automatisant les tâches répétitives ce qui favorisera une meilleure utilisation des ressources, une meilleure réactivité aux demandes et une amélioration globale de la performance organisationnelle.

# Chapitre 2

## Analyse et conception

### 2.1 Introduction

Dans ce chapitre, nous présenterons les étapes essentielles du développement de notre application. Nous débuterons par une spécification des besoins, en identifiant les besoins fonctionnels et non fonctionnels. Ensuite, nous mettrons en œuvre les processus de modélisation et de développement choisis, ainsi que les différentes phases de la conception.

### 2.2 Spécification des besoins

Une identification des besoins fonctionnels et non fonctionnels de notre application nous permettra de définir en détail les fonctionnalités requises ainsi que les contraintes techniques, garantissant ainsi une conception adaptée aux attentes de l'utilisateur et aux exigences du projet.

#### 2.2.1 Besoins fonctionnels

Voici les besoins fonctionnels identifiés pour notre système :

- Gérer le profil utilisateur : consulter les informations du profil et apporter des changements si nécessaires
- Gérer les établissements
- Gérer les missionnaires

- Générer un bon de commande (hôtel, TAA, et TAC)
- Éditer un bon de commande (hôtel, TAA, et TAC)
- Supprimer un bon de commande (hôtel, TAA, et TAC)
- Consulter la liste des bons de commande (hôtel, TAA, et TAC)
- Établir une facture (hôtel, TAC)
- Consulter la liste des factures (hôtel, TAC)
- Générer les rapports (hôtel, TAC)
- Filtrer les rapports (par période et par hôtel ou par lieu) pour les factures d'hôtels
- Filtrer les rapports (par période et par compagnie ou par structure) pour les factures du TAC
- Visualiser et imprimer les rapports
- Rechercher un bon de commande, facture, établissement ou un missionnaire

### 2.2.2 Besoins non fonctionnels

- **Sécurité** : l'application doit assurer la confidentialité des données utilisateur à travers une authentification sécurisée avec un mot de passe attribué initialement à l'utilisateur.
- **Performance** : le système doit garantir des temps de réponse rapides lors de l'exécution des actions et maintenir une bonne réactivité de l'application.
- **Utilisabilité et ergonomie** : l'application doit offrir une interface utilisateur intuitive, facile à utiliser et à comprendre.
- **Maintenabilité** : assurer un code source bien structuré, facilement compréhensible et documenté pour faciliter les modifications et les mises à jour de l'application.

## 2.3 Identification des acteurs du système

Dans notre système, le responsable de RELEX occupe le rôle d'acteur principal. En tant qu'unique utilisateur de l'application, il joue un rôle clé dans l'utilisation et la gestion du système, contribuant ainsi à la prise de décision et à la coordination efficace pour répondre aux besoins de la section RELEX.

Le responsable de RELEX est responsable de diverses tâches, notamment la consultation des informations, la gestion des établissements et des missionnaires, ainsi que la gestion des bons de commande,

factures et rapports.

## **2.4 Conception**

Pour concevoir une solution fonctionnelle et efficace répondant aux besoins de notre application, nous utiliserons UML comme langage de modélisation, offrant des outils et des diagrammes standardisés pour représenter les différentes composantes de notre application. En parallèle, nous adopterons la méthode SCRUM, une approche de développement agile, favorisant la collaboration et l'itération continue. En combinant UML et SCRUM, nous concevons une application bien structurée, flexible et adaptée aux besoins changeants de notre utilisateur.

### **2.4.1 Processus de développement**

Le développement logiciel est un processus complexe qui peut être difficile à gérer et à exécuter. Les méthodes de développement traditionnelles peuvent souvent être rigides et peu adaptées à un environnement de travail en constante évolution. C'est pourquoi de plus en plus d'entreprises se tournent vers des méthodes de développement agiles pour leur permettre de s'adapter rapidement aux changements et d'offrir un produit final de qualité.

Afin de répondre aux exigences du client de manière efficace et mener à bien notre projet tout en optimisant les délais de notre projet, nous avons opté pour la méthode agile "SCRUM". Dans cette section, nous décrirons les principes fondamentaux de cette méthode, son processus de travail ainsi que sa mise en œuvre pour assurer le succès de notre projet.

#### **2.4.1.1 Présentation de SCRUM**

SCRUM (Systematic Customer Resolution Unraveling Meeting) est une méthode agile de gestion de projet qui se concentre sur l'adaptation aux besoins du client et la productivité de l'équipe de développement. Elle permet aux développeurs de travailler étroitement avec les clients pour éviter le développement de systèmes qui ne seront pas utilisés. Son objectif phare est d'améliorer la productivité des équipes, tout en permettant une optimisation du produit grâce à des feedbacks réguliers avec les utilisateurs finaux [5].

### 2.4.1.2 Piliers de SCRUM

- **Transparence** : toutes les informations pertinentes doivent être visibles et accessibles à tous les membres de l'équipe SCRUM.
- **Inspection** : l'équipe SCRUM et les parties prenantes inspectent régulièrement les livrables pour s'assurer qu'ils sont en conformité avec les attentes et besoins des parties prenantes.
- **Adaptation** : l'équipe SCRUM doit être capable de s'adapter aux changements et d'ajuster son plan et sa manière de travailler en fonction de nouvelles informations ou de nouvelles priorités [6].

### 2.4.1.3 Les rôles SCRUM

L'équipe de développement SCRUM est composée de :

1. **SCRUM Master** : c'est le leader de l'équipe SCRUM, son rôle est de s'assurer que l'équipe de développement est autonome, auto-organisée et efficace et qu'elle suit les principes de SCRUM (la transparence, l'inspection et l'adaptation) et faciliter la collaboration entre les membres de l'équipe et avec les parties prenantes externes.
2. **Product Owner** : est le représentant de l'entreprise ou du client responsable de la définition et de la gestion du carnet de produit (Product Backlog), qui est une liste de toutes les fonctionnalités, les tâches et les exigences du projet. C'est lui qui s'assure de la conformité fonctionnelle du produit final et qui porte la vision globale de l'application.
3. **Équipe de développement** : elle est composée de professionnels multidisciplinaires qui travaillent ensemble pour réaliser les fonctionnalités décrites dans le carnet de produit. L'équipe de développement est auto-organisée et multidisciplinaire et doit posséder toutes les compétences nécessaires pour réaliser les tâches du projet.

Ensemble, ces trois rôles travaillent en étroite collaboration pour atteindre les objectifs fixés pour chaque sprint et pour livrer des fonctionnalités de haute qualité dans des délais serrés [6].

Les rôles pour notre projet sont répartis comme indiqués dans le tableau suivant :

Rôles SCRUM	Personnes affectées
Scrum Master	Mme. AIT HACÈNE Souhila
Product Owner	M. KACHER Reda
Équipe de développement	Mme. TIGHIDET Messilva et M. TAGUILSA Rafik

TABLE 2.1 – Les rôles SCRUM

#### 2.4.1.4 Concepts de SCRUM

Voici les 8 concepts de la méthode SCRUM :

- **Product Backlog** : Cette étape consiste à définir les fonctionnalités et les objectifs du projet sous forme de liste d'éléments à réaliser appelés backlog de produit. Cette liste est ordonnée en fonction de la priorité de chaque élément.
- **Sprint planning** : Au début de chaque sprint, l'équipe se réunit pour planifier les tâches à réaliser au cours du sprint. Elle sélectionne les éléments du backlog de produit qui seront traités pendant le sprint et définit les tâches à réaliser pour chaque élément.
- **Sprint Backlog** : Cette étape consiste à définir une liste des tâches à réaliser pour atteindre les objectifs du sprint, à partir des éléments sélectionnés lors de la réunion de planification de sprint.
- **Sprint** : C'est la période pendant laquelle l'équipe réalise les tâches planifiées lors de la phase de planification du sprint. Les sprints ont une durée fixe et courte (généralement de 1 à 4 semaines).
- **Daily SCRUM** : C'est des réunions organisées chaque jour par l'équipe afin de faire le point sur les tâches réalisées la veille, les tâches en cours et les tâches à réaliser le jour même. Cette réunion permet de suivre l'avancement du projet et de détecter rapidement les éventuels problèmes.
- **Sprint Review** : à la fin de chaque sprint, l'équipe organise une réunion de revue de sprint pour présenter les résultats obtenus. Cette réunion permet d'évaluer l'avancement du projet, de vérifier si les objectifs du sprint ont été atteints et d'obtenir des retours du client.

- **Sprint Retrospective** : après la revue de sprint, l'équipe se réunit pour une réunion de rétrospective. Elle évalue le sprint qui vient de se terminer, discute des problèmes rencontrés et des améliorations possibles.
- **Finished product** : cette étape est atteinte lorsque toutes les fonctionnalités prévues dans le backlog de produit ont été développées, testées et livrées.

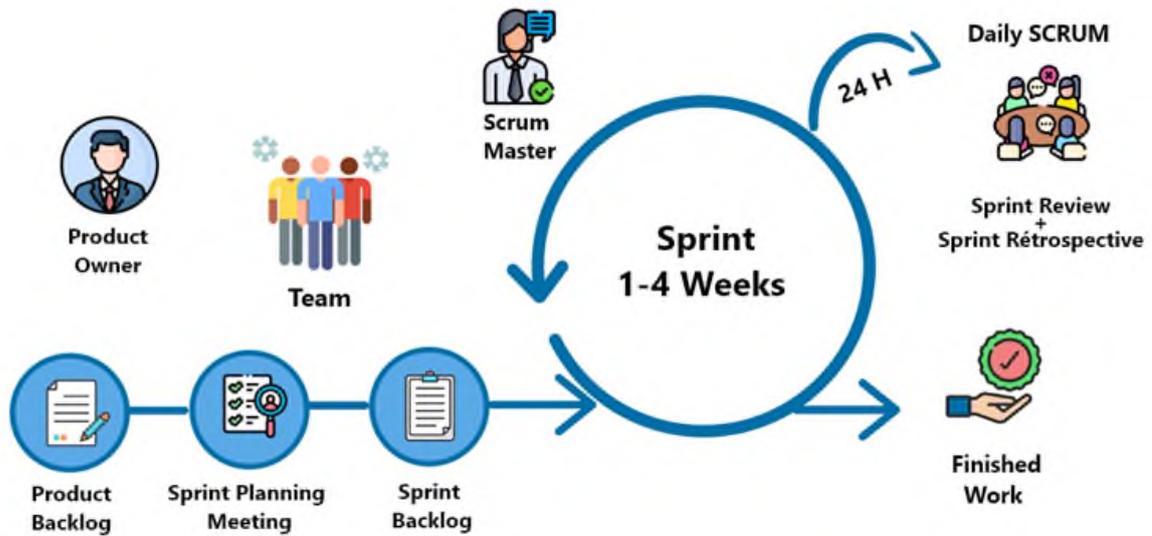


FIGURE 2.1 – Méthode de gestion de projet SCRUM [5]

### 2.4.1.5 La mise en œuvre de SCRUM

Maintenant que nous avons présenté les concepts fondamentaux de la méthodologie SCRUM, nous sommes prêts à l'appliquer à notre projet.

Après avoir échangé avec le client et analysé ses besoins, nous avons procédé à la définition des fonctionnalités à développer sous forme d'User Stories. Ces User Stories sont des phrases claires et concises qui utilisent un langage courant pour décrire chaque fonctionnalité de manière compréhensible.

En tant que responsable de RELEX, je veux :

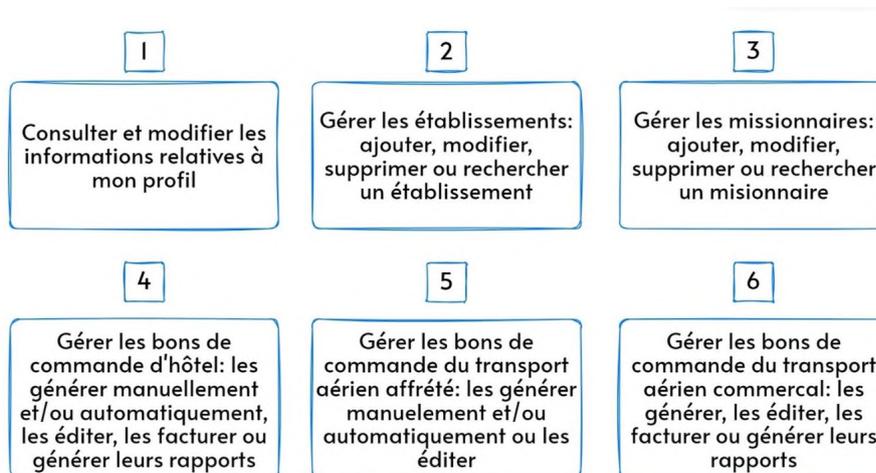


FIGURE 2.2 – User stories de notre application.

Après avoir identifié les user stories, nous avons réparti les tâches selon des sprints d'une durée limite de réalisation (en fonction de la tâche à réaliser) et nous avons créé un plan de releases illustré dans la figure 2.3 suivante :

User story	Sprint	Durée du sprint	
<b>01</b>	1	1 semaine	Mise en place de l'interface utilisateur pour afficher les informations de profil
	2	1 semaine	Ajout de la fonctionnalité de modification du profil
<b>02</b>	1	1 semaine	Création de la fonctionnalité d'affichage de la liste des établissements
	2	1 semaine	Implémentation des fonctionnalités de recherche, ajout, modification et suppression des établissements
<b>03</b>	1	1 semaine	Création de la fonctionnalité d'affichage de la liste des missionnaires
	2	1 semaine	Implémentation des fonctionnalités de recherche, ajout, modification et suppression des missionnaires
<b>04</b>	1	1 semaine	Implémentation de la fonctionnalité de génération manuelle des bons de commande d'hôtel
	2	2 semaines	Ajout de la fonctionnalité de génération automatique des bons de commande d'hôtel
	3	1 semaine	Intégration de la fonctionnalité d'édition des bons de commande d'hôtel
	4	2 semaines	Mise en place de la fonctionnalité de facturation des bons de commande d'hôtel
	5	2 semaines	Ajout de la fonctionnalité de génération de rapports pour les bons de commande d'hôtel
<b>05</b>	1	1 semaine	Implémentation de la fonctionnalité de génération manuelle des bons de commande du transport aérien affrété
	2	2 semaines	Ajout de la fonctionnalité de génération automatique des bons de commande du transport aérien affrété
	3	1 semaine	Intégration de la fonctionnalité d'édition des bons de commande du transport aérien affrété
<b>06</b>	1	1 semaine	Implémentation de la fonctionnalité de génération manuelle des bons de commande du transport aérien commercial
	2	1 semaine	Ajout de la fonctionnalité d'édition des bons de commande du transport aérien commercial
	3	2 semaines	Ajout de la fonctionnalité de facturation des bons de commande du transport aérien commercial
	4	2 semaines	Mise en place de la fonctionnalité de génération de rapports pour les bons de commande du transport aérien commercial

FIGURE 2.3 – Planification des releases.

## 2.4.2 Langage de modélisation UML

UML (Unified Modeling Language) est un langage de modélisation largement utilisé dans le domaine du développement logiciel. Il fournit un ensemble de notations et de techniques standardisées pour représenter graphiquement les différentes facettes d'un système logiciel. Grâce à ses nombreux diagrammes, UML offre une méthode visuelle puissante pour analyser, concevoir et documenter les logiciels de manière claire et compréhensible [7].

UML 2 comporte 14 types de diagrammes :

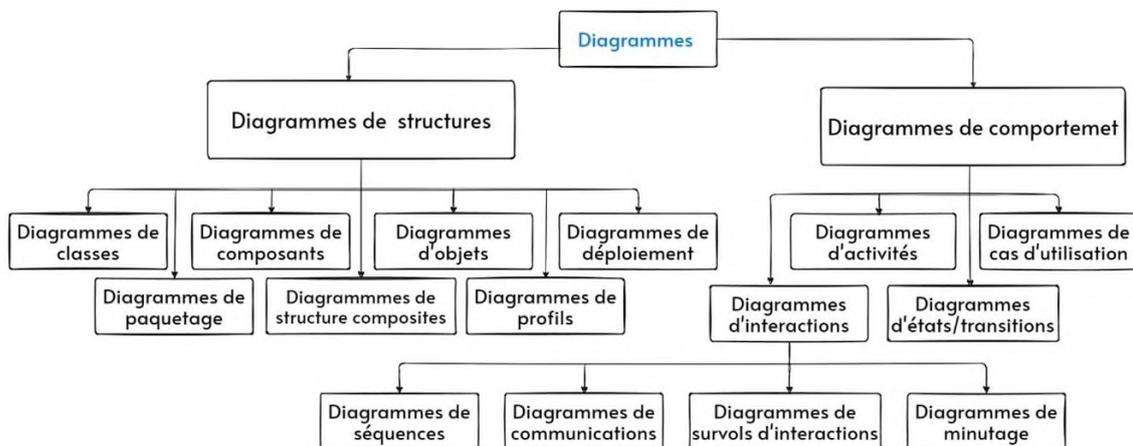


FIGURE 2.4 – Les type de diagrammes d’UML.

Pour la conception de notre projet, nous avons utilisé les diagrammes de cas d'utilisation, de séquence et de classes

### 2.4.2.1 Identification des cas d'utilisation

Le tableau 2.2 répertorie les divers cas d'utilisation de notre application :

Acteur	Cas d'utilisation		
Responsable de RELEX	S'authentifier		
	Consulter les informations du profil	Modifier les informations du profil	
	Consulter la liste des établissements	Rechercher un établissement	
		Ajouter un établissement	
		Modifier un établissement	
		Supprimer un établissement	
	Consulter la liste des missionnaires	Rechercher un missionnaire	
		Ajouter un missionnaire	
		Modifier un missionnaire	
		Supprimer un missionnaire	
	Gérer les bons de commande d'Hôtel	Consulter la liste des BDC	
		Générer un BDC automatiquement	
		Générer un BDC manuellement	
		Rechercher un BDC	
		Modifier un BDC	
		Supprimer un BDC	
	Gérer les bons de commande TAA	Consulter la liste des BDC	
		Générer un BDC automatiquement	
		Générer un BDC manuellement	
		Rechercher un BDC	
		Modifier un BDC	
		Supprimer un BDC	
	Gérer les bons de commande TAC	Consulter la liste des BDC	
		Générer un BDC automatiquement	
		Générer un BDC manuellement	
		Rechercher un BDC	
		Modifier un BDC	
Supprimer un BDC			

Responsable de RELEX	Établir les factures hôtel	
	Consulter la liste des factures hôtel	Rechercher une facture
		Modifier une facture
		Supprimer une facture
		Imprimer la liste des factures
	Établir les factures TAC	
	Consulter la liste des factures TAC	Rechercher une facture
		Modifier une facture
		Supprimer une facture
		Imprimer la liste des factures
	Gérer les rapports hôtel	Filtrer le rapport par date
		Filtrer le rapport par hôtel
		Filtrer le rapport par lieu
		Visualiser le rapport
		Générer le rapport
	Gérer les rapports TAC	Filtrer le rapport par date
		Filtrer le rapport par compagnie
		Filtrer le rapport par structure
		Visualiser le rapport
		Générer le rapport

TABLE 2.2 – Identification des cas d'utilisation

### 2.4.2.2 Diagramme de cas d'utilisation

Les cas d'utilisation servent à exprimer le comportement du système en matière d'actions et de réactions, le diagramme de cas d'utilisation comprend les acteurs, le système et les cas d'utilisation.

- Cas d'utilisation N°1 : Gestion du profil

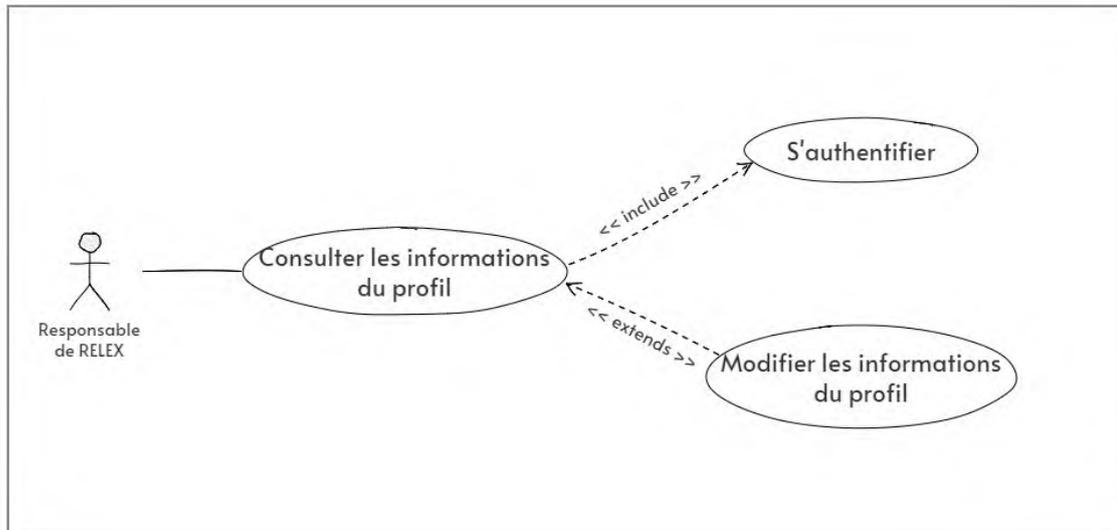


FIGURE 2.5 – Diagramme de cas d'utilisation " Gestion du profil ".

- Cas d'utilisation n°2 : Consulter la liste des missionnaires

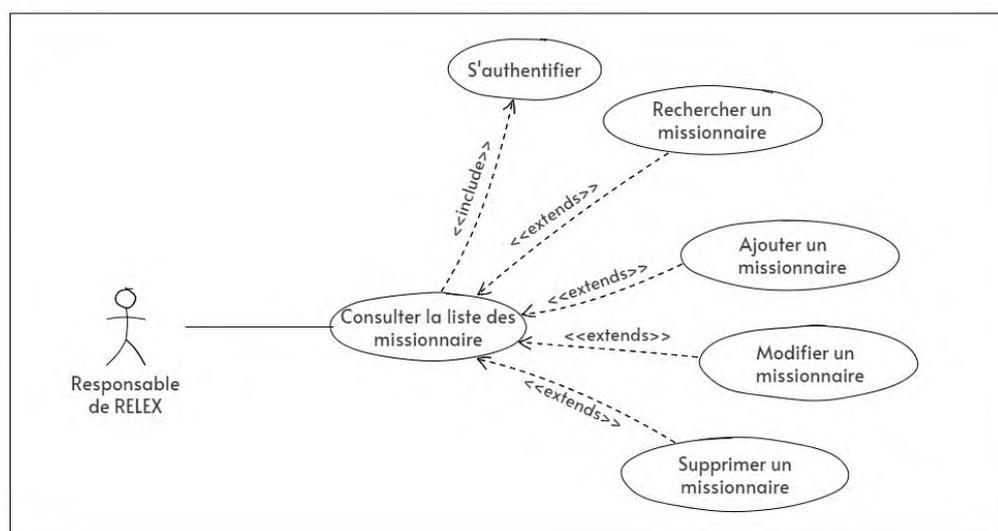


FIGURE 2.6 – Diagramme de cas d'utilisation " Consulter la liste des missionnaires ".

• Cas d'utilisation n°3 : Consulter la liste des établissements

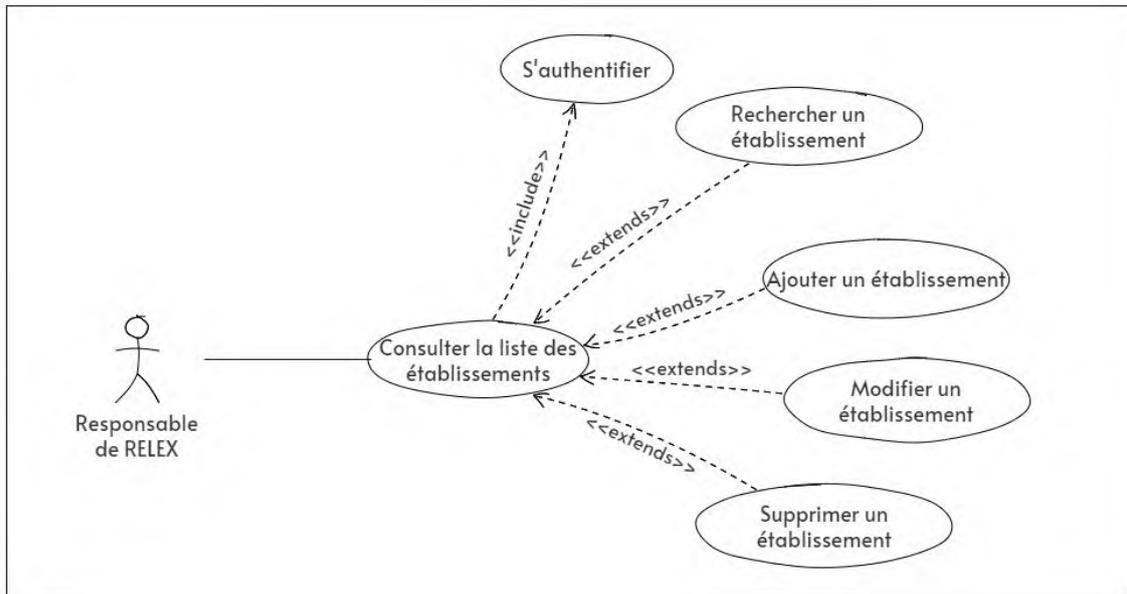


FIGURE 2.7 – Diagramme de cas d'utilisation " Consulter la liste des établissements " .

• Cas d'utilisation n°4 : Gérer les bons de commande

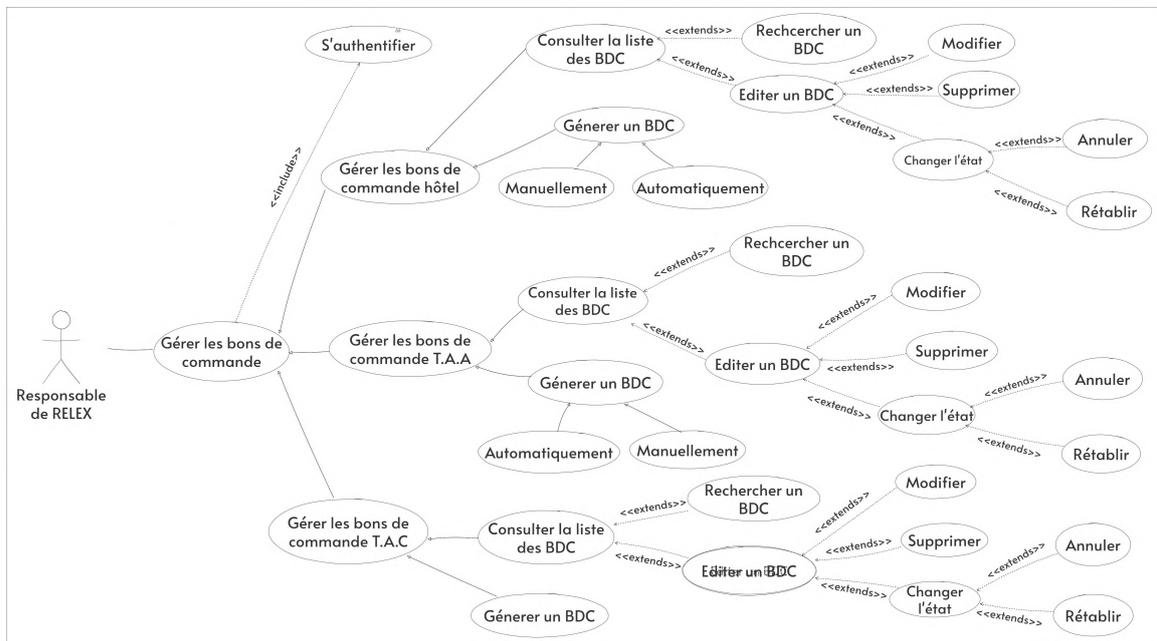


FIGURE 2.8 – Diagramme de cas d'utilisation "Gérer les bons de commande" .

• Cas d'utilisation n°5 : Établir une facture

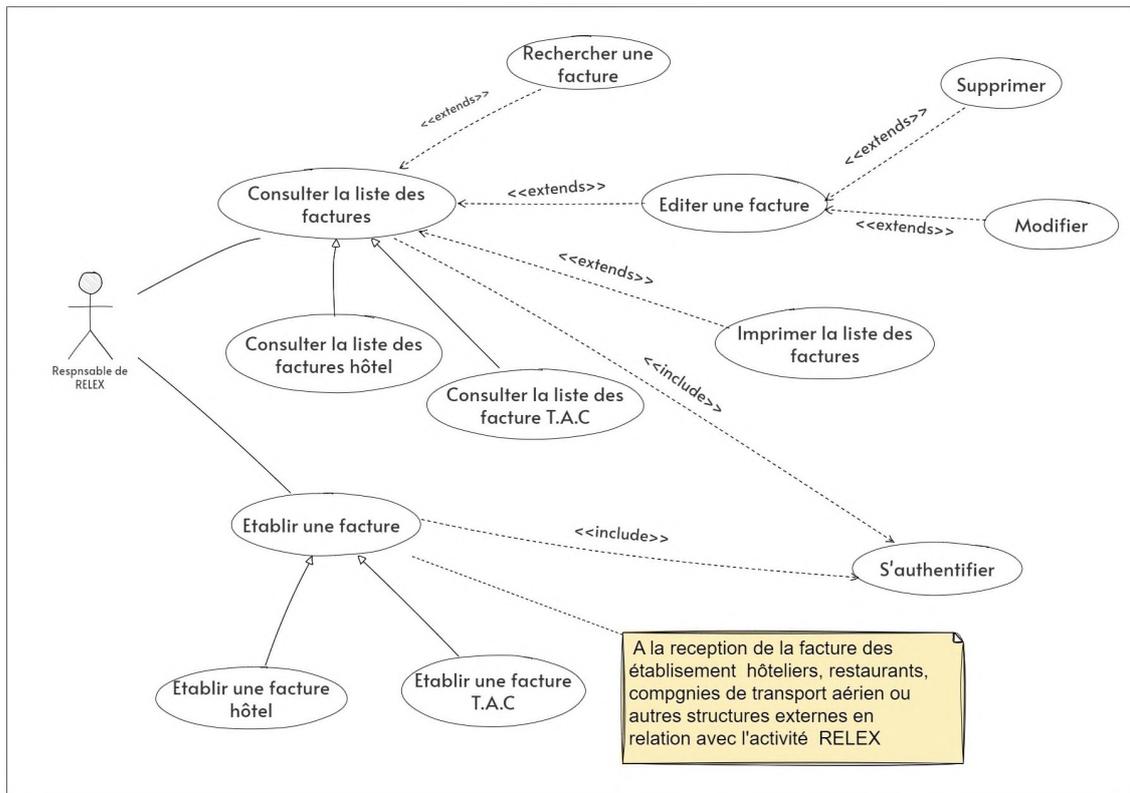


FIGURE 2.9 – Diagramme de cas d'utilisation " Établir une facture ".

• Cas d'utilisation n°6 : Gérer les rapports

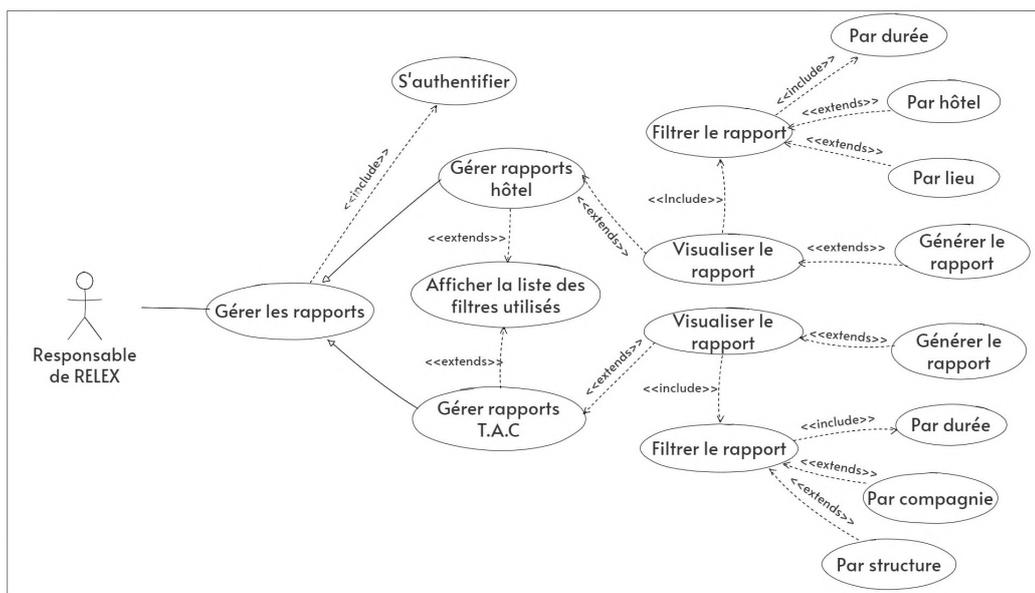


FIGURE 2.10 – Diagramme de cas d'utilisation " Gérer les rapports ".

### 2.4.2.3 Description textuelle des cas d'utilisation

Afin de détailler les fonctionnalités du système, nous présentons ci-dessous les descriptions textuelles des cas d'utilisation :

- Description textuelle du cas : Générer un bon de commande hôtel

<b>Désignation</b>	<b>Générer un bon de commande hôtel</b>
<b>Acteur principal</b>	Le responsable de Relex
<b>Objectif</b>	Générer un bon de commande hôtel
<b>Précondition</b>	L'utilisateur doit être authentifié
<b>Scénario nominal</b>	<ol style="list-style-type: none"><li>1. L'utilisateur clique sur le lien 'Générer BDC hôtel.</li><li>2. Le système affiche les options de création du bon de commande : automatique ou manuelle.</li><li>3. L'utilisateur choisit l'option de génération automatique.</li><li>4. Le système demande de sélectionner un fichier de bon de commande au format Word.</li><li>5. L'utilisateur insère une demande de prise en charge en format Word.</li><li>6. Le système affiche un formulaire de génération de bon de commande hôtel pré-rempli avec les informations récupérées du fichier Word.</li><li>7. L'utilisateur vérifie la pertinence des informations et clique sur le bouton 'Valider'.</li><li>8. Le système affiche une petite fenêtre pour demander à l'utilisateur de choisir un nom pour le document généré.</li></ol>

	<p>9. L'utilisateur choisit un nom et valide.</p> <p>10. Un message de succès est affiché indiquant que le bon de commande a été créé avec succès.</p> <p>11. Le fichier Word généré est affiché et l'utilisateur est redirigé vers la page de la liste des bons de commande d'hôtel déjà enregistrés.</p>
<b>Scénarios alternatifs</b>	<p><b>Scénario alternatif 1 :</b> Le scénario démarre au point 5 du scénario nominal</p> <p>3. L'utilisateur insère un fichier Word qui n'est pas une demande de prise en charge.</p> <p>4. Le système affiche un message d'erreur demandant à l'utilisateur d'insérer une demande de prise en charge valide.</p> <p>5. L'utilisateur insère une nouvelle demande de prise en charge.</p> <p><b>Scénario alternatif 2 :</b> Le scénario démarre au point 7 du scénario nominal</p> <p>7. L'utilisateur clique sur le bouton 'Valider' malgré la présence de champs vides.</p> <p>8. Le système affiche un message d'erreur et demande à l'utilisateur de remplir tous les champs.</p> <p>9. L'utilisateur remplit le formulaire et clique une nouvelle fois sur le bouton 'Valider'.</p>
<b>Post-condition</b>	Un nouveau bon de commande d'hôtel est enregistré dans la base de données et un document Word est généré et affiché à l'écran

TABLE 2.4 – Description textuelle du cas d'utilisation " Générer un bon de commande hôtel"

**Remarque :** pour la génération des bons de commande du transport aérien affrété (TAA) et du transport aérien commercial (TAC), la procédure est la même que celle présentée précédemment à l'exception que pour le TAC, l'option de génération automatique n'est pas disponible.

- Description textuelle du cas : Établir une facture hôtel

<b>Désignation</b>	<b>Établir une facture hôtel</b>
<b>Acteur principal</b>	Responsable de Relex
<b>Objectif</b>	Établir des factures d'hôtel
<b>Précondition</b>	L'utilisateur doit être authentifié
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur clique sur le lien 'Générer facture hôtel'.</li> <li>2. Le système affiche un formulaire demandant des informations sur la facture à générer.</li> <li>3. L'utilisateur remplit le formulaire et clique sur le bouton 'Valider'.</li> <li>4. Le système affiche un message de succès indiquant que la facture a été enregistrée avec succès.</li> <li>5. L'utilisateur est redirigé vers la page de la liste des factures d'hôtel déjà enregistrées.</li> </ol>
<b>Scénario alternatif</b>	<p>Scénario alternatif :</p> <p>Le scénario démarre au point 3 du scénario nominal.</p> <ol style="list-style-type: none"> <li>3. L'utilisateur clique sur le bouton 'Valider' malgré la présence de champs vides ou bien la facture existe déjà dans la base de données.</li> <li>4. Le système affiche un message d'erreur et demande à l'utilisateur de remplir tous les champs ou de modifier le champ numéro de facture si la facture existe déjà.</li> <li>5. L'utilisateur modifie les informations saisies et clique sur le bouton 'Valider'.</li> </ol>
<b>Post-condition</b>	Une nouvelle facture d'hôtel est enregistrée dans la base de données et la liste des factures d'hôtel déjà établies est affichée.

TABLE 2.5 – Description textuelle du cas d'utilisation " Établir une facture hôtel"

**Remarque** : pour l'établissement des factures du transport aérien commercial (TAC), la procédure est identique.

- Description textuelle du cas : Générer un rapport hôtel

<b>Désignation</b>	<b>Générer un rapport hôtel</b>
<b>Acteur principal</b>	Responsable de Relex
<b>Objectif</b>	Générer un rapport rapport hôtel.
<b>Précondition</b>	L'utilisateur doit être authentifié
<b>Scénario nominal</b>	<ol style="list-style-type: none"> <li>1. L'utilisateur clique sur le lien 'Générer rapport hôtel'.</li> <li>2. Le système affiche un formulaire demandant des informations (filtres) sur le rapport à générer.</li> <li>3. L'utilisateur choisit les filtres et clique sur le bouton 'Visualiser'.</li> <li>4. Le système affiche un rapport d'hôtel que l'utilisateur peut générer sous forme d'un document Word.</li> </ol>
<b>Scénario alternatif</b>	<p>Scénario alternatif :</p> <p>Le scénario démarre au point 3 du scénario nominal.</p> <ol style="list-style-type: none"> <li>3. L'utilisateur choisit des filtres qui ne donnent lieu à aucun rapport.</li> <li>4. Le système affiche un message d'erreur indiquant qu'il n'y a aucun résultat.</li> <li>5. L'utilisateur choisit une nouvelle fois des filtres et clique sur le bouton 'Visualiser'.</li> </ol>
<b>Post-condition</b>	Un rapport d'hôtel a été généré avec les filtres choisis, présentant le nombre de bons de commandes établis, chiffre d'affaires et d'autres informations pertinentes.

TABLE 2.6 – Description textuelle du cas d'utilisation "Générer un rapport hôtel"

**Remarque** : la génération des rapports du transport aérien commercial (TAC) suit la même procédure que la génération des rapports d'hôtel, à l'exception des filtres utilisés. Pour les rapports d'hôtel, les filtres comprennent la date, l'hôtel et le lieu (la date étant obligatoire, les autres sont facultatifs). En revanche, pour les rapports de TAC, les filtres utilisés sont la date, la structure et la compagnie (la date étant obligatoire, les autres sont facultatifs).

#### **2.4.2.4 Diagramme de séquence**

Il représente des échanges de messages entre objets, selon un point de vue temporel. Il permet ainsi de décrire les scénarios de chaque cas d'utilisation en mettant l'accent sur la chronologie des opérations en interaction avec les objets.

• Cas n° 1 : Générer un bon de commande hôtel automatiquement

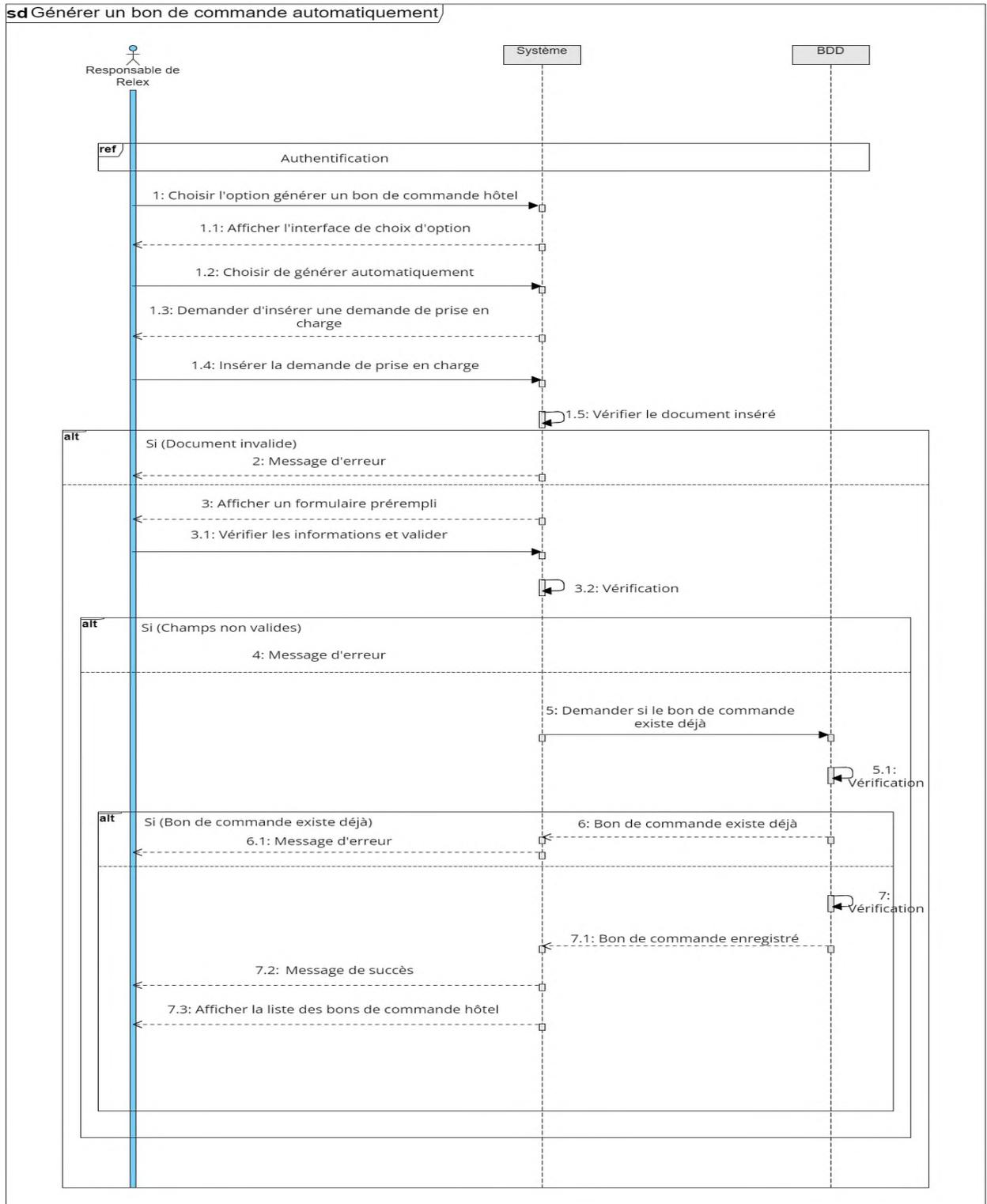


FIGURE 2.11 – Diagramme de séquence du cas d'utilisation " Générer un bon de commande hôtel automatiquement ".

• Cas n° 2 : Générer un bon de commande hôtel manuellement

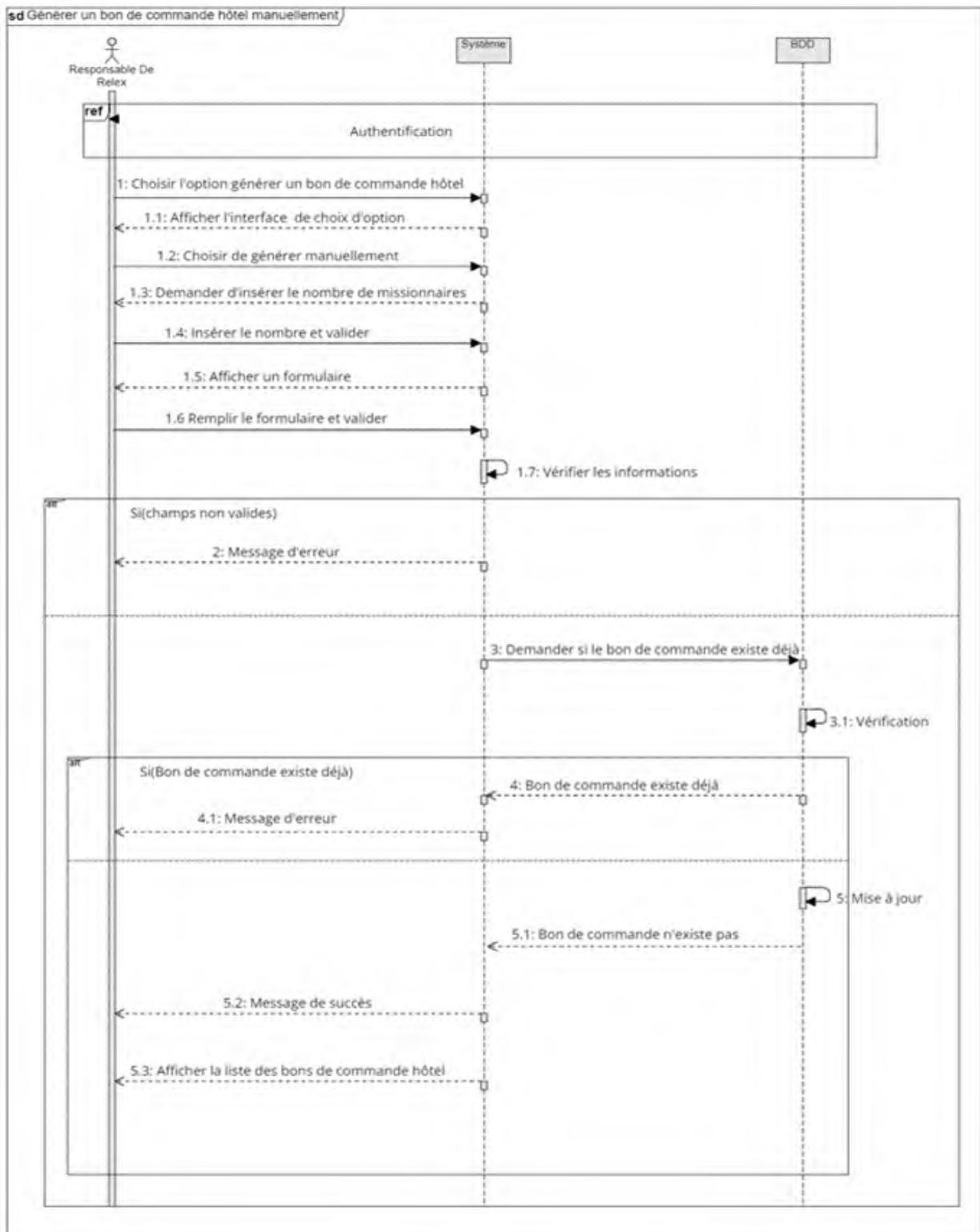


FIGURE 2.12 – Diagramme de séquence du cas d'utilisation " Générer un bon de commande hôtel manuellement ".

• Cas n° 3 : Établir une facture hôtel

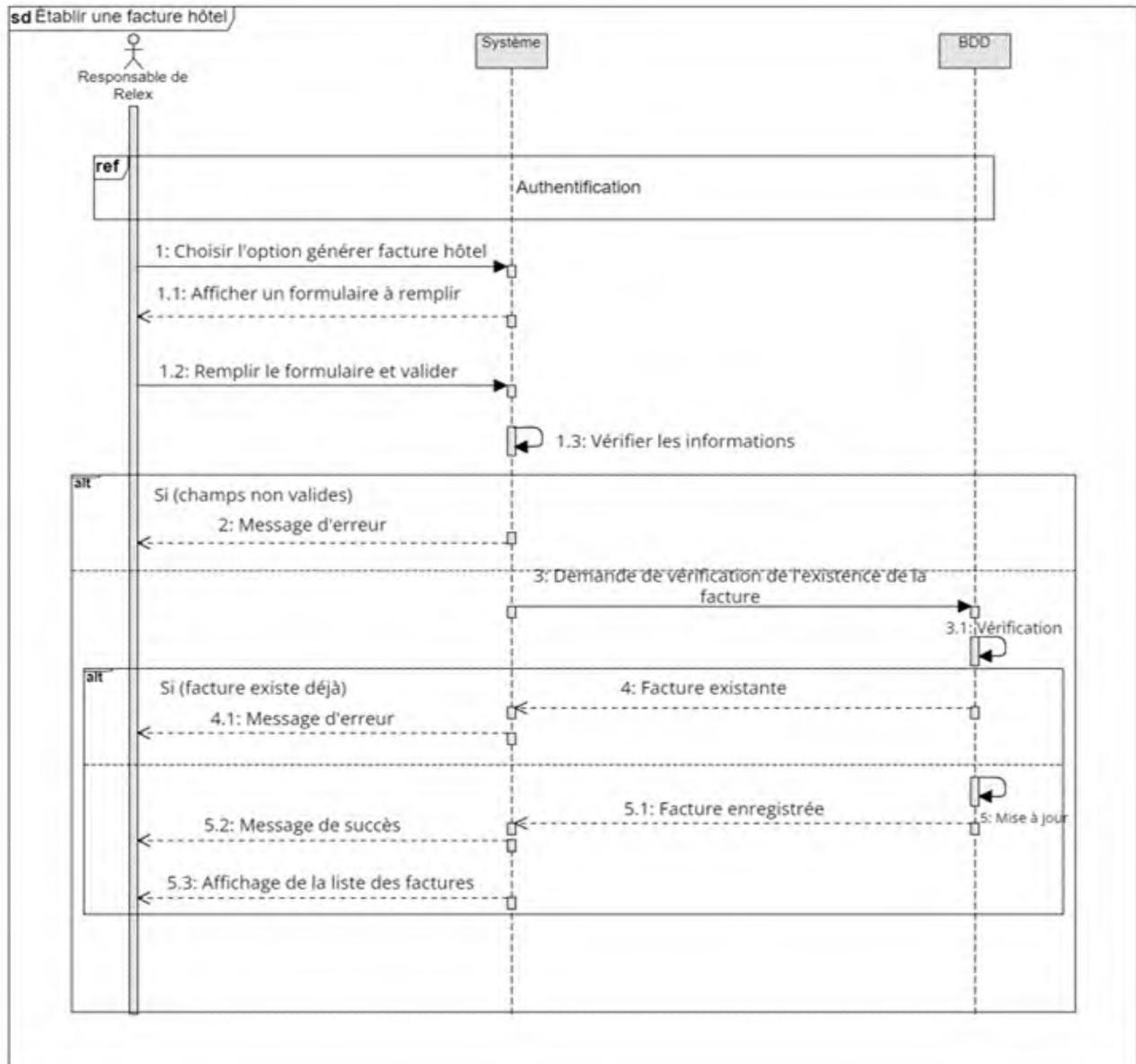


FIGURE 2.13 – Diagramme de séquence du cas d'utilisation " Établir une facture hôtel ".

• Cas n° 4 : Générer un rapport hôtel

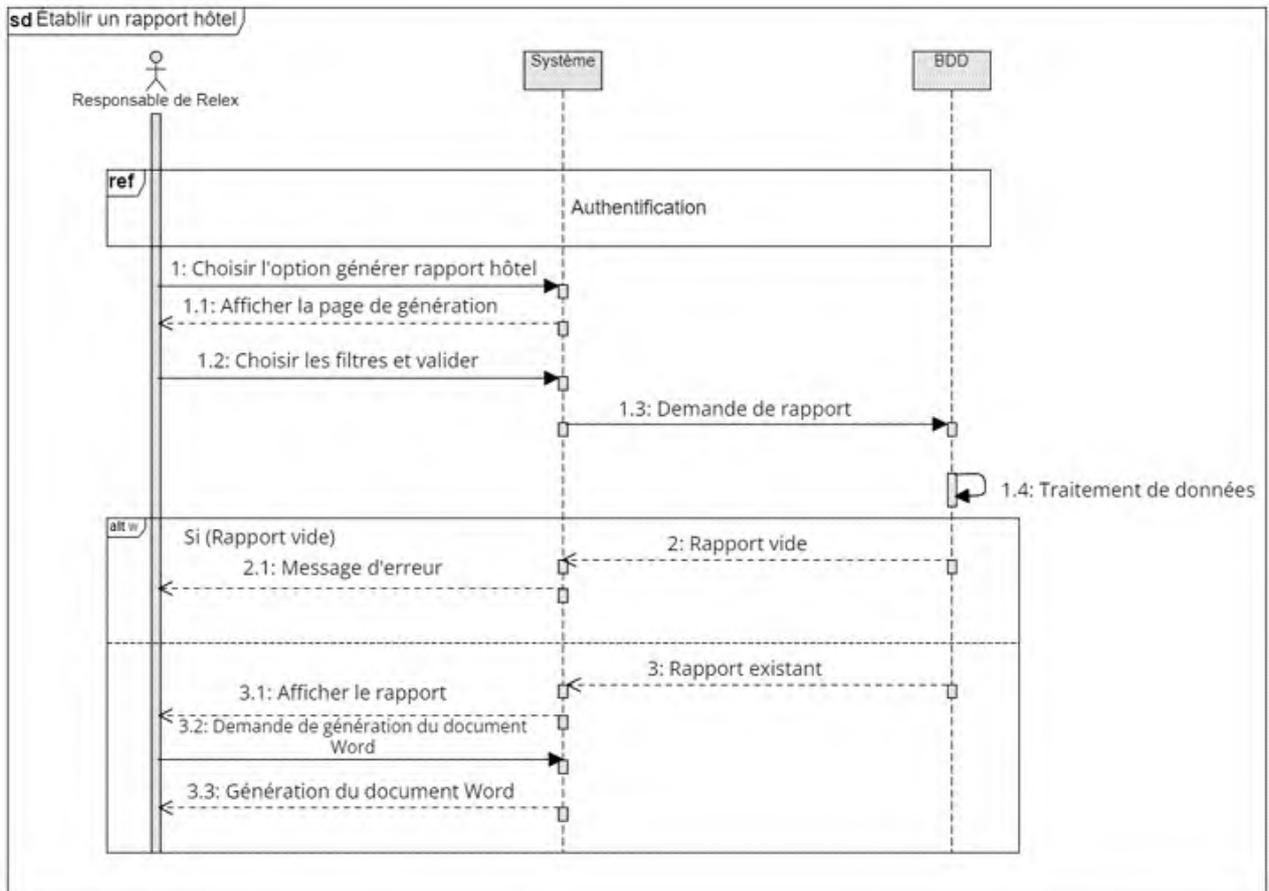


FIGURE 2.14 – Diagramme de séquence du cas d'utilisation " Générer un rapport hôtel ".

### 2.4.2.5 Dictionnaire de données

Voici le dictionnaire de données associé à notre diagramme de classes, présenté dans le tableau ci-dessous :

Classe	Attributs	Désignation	Type
ResponsableRelex	id_responsable_relex	Identifiant du responsable de Relex	Integer
	nom_responsable_relex	Nom du responsable de Relex	String
	prenom_responsable_relex	Prénom du responsable de Relex	String
	sexe_responsable_relex	Sexe (genre) du responsable de Relex	String
	email_responsable_relex	Email du responsable de Relex	String
	mdp_responsable_relex	Mot de passe du responsable de Relex	String
	pdp_responsable_relex	Photo de profil du responsable de Relex	BLOB
Etablissement	id_etablissement	Identifiant de l'établissement	Integer
	nom_etablissement	Nom de l'établissement	String
	type_etablissement	Type de l'établissement	String
	categorie_etablissement	Catégorie de l'établissement	String
	ville_etablissement	Ville de l'établissement	String
	telephone_etablissement	Téléphone de l'établissement	String
	fax_etablissement	Fax de l'établissement	String
	email_etablissement	E-mail de l'établissement	String
Missionnaire	id_missionnaire	Identifiant du missionnaire	Integer
	nom_missionnaire	Nom du missionnaire	String
	structure_missionnaire	Structure du missionnaire	String

BonDeCommande	id_BDC	Identifiant du bon de commande	Integer
	num_BDC	Numéro du bon de commande	Integer
	structure	Structure du bon de commande	String
	liste_personnes	Liste des missionnaires du bon de commande	String
	nbr_personnes	Nombre de missionnaires	Integer
	date_traitement	Date de génération du bon de commande	Date
	statut	Statut du bon de commande	String
BonDeCommandeHotel	nom_hotel	Nom de l'hôtel	String
	lieu_hotel	Lieu de l'hôtel	String
	date_deb_sejour	Date de début de séjour	Date
	date_fin_sejour	Date de fin de séjour	Date
	directeur	Directeur	String
	responsable	Responsable	String
	sous_directeur	Sous-Directeur	String
	signataire	Signataire du document	String
	hebergement	Hébergement	String
	restauration	Restauration	String
	demi_pension	Demi-pension	String
	pension_complete	Pension-complète	String
	type_chambre	Type de chambre	String
	autre_prestation	Autre prestation	String
path_DPC	Chemin du fichier (Demande de prise en charge)	String	

	path_BDC	Chemin du fichier (Bon de commande)	String
BonDeCommandeTAA	lieu_depart lieu_arrivee directeur responsable sous_directeur signataire num_vol	Lieu du départ Lieu d'arrivée Directeur Responsable Sous-Directeur Signataire Numéro du vol	String String String String String String Integer
	type_vol date_vol path_DATT path_BDC	Type du vol Date du vol Chemin du fichier (Demande achat titre de transport) Chemin du fichier (Bon de commande)	String Date String String
BonDeCommandeTAC	compagnie	Compagnie aérienne	String
Facture	id_facture num_facture date_facture montant_total nbr_personnes_factures date_traitement	Identifiant de la facture Numéro de la facture Date de la facture Montant total de la facture Nombre de missionnaires de la facture Date de génération de la facture	Integer String Date Float Integer Date
FactureHotel	fe nbr_nuites	Fiche d'enregistrement Nombre de nuitées	Integer Integer
FactureTAC	liste_personnes_facture	Liste des missionnaires de la facture	String

---

TABLE 2.7 – Dictionnaire de données

#### **2.4.2.6 Diagramme de classes**

Le diagramme de classes constitue l'un des pivots essentiels de la modélisation avec UML. En effet, ce diagramme fournit une représentation statique du système à développer. Cette représentation est centrée sur les concepts de classe et d'association. Chaque classe se décrit par les données et les traitements dont elle est responsable.

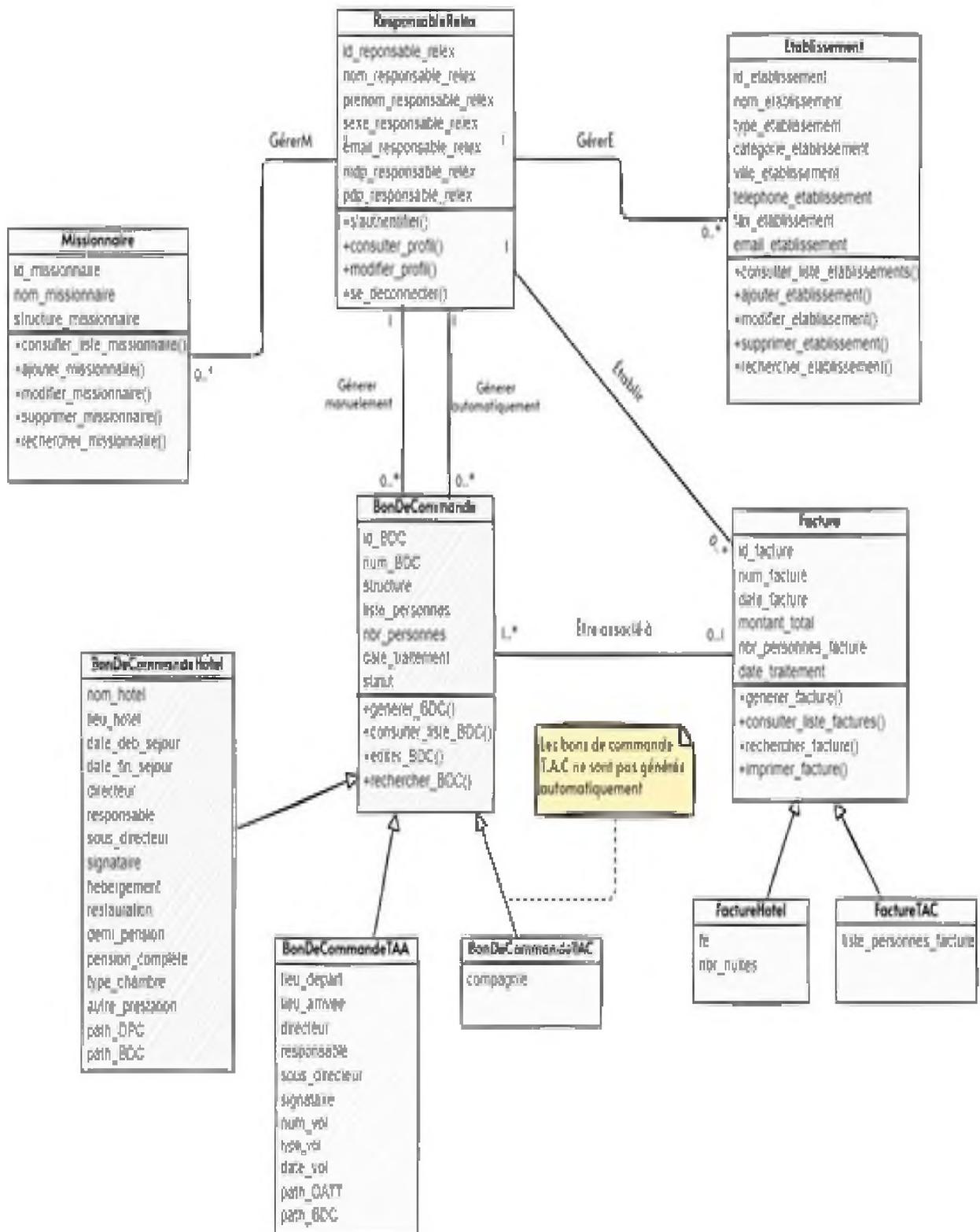


FIGURE 2.15 – Diagramme de classes.

### 2.4.2.7 Passage du diagramme de classes au modèle relationnel

Le passage du diagramme de classes au modèle relationnel est une étape cruciale dans le processus de conception d'une base de données. En utilisant des principes de modélisation relationnelle, le diagramme de classes est transformé en une représentation plus structurée et normalisée des données. Chaque classe devient une table dans le modèle relationnel, et chaque attribut devient une colonne. Les associations entre les classes sont exprimées à l'aide des clés étrangères, établissant ainsi des liens entre les différentes tables.

Ce passage suit des règles bien définies qui sont :

- La transformation des classes :
  - Une classe devient une table qui porte généralement le nom de la classe.
  - Les attributs de classe deviennent des colonnes dans la table correspondante.
  - L'identifiant unique de la classe devient une clé primaire de la table. La clé primaire garantit l'unicité des enregistrements de la table.
- La transformation des associations :
  - Une association un-à-plusieurs se traduit en ajoutant une clé étrangère dans la table qui est du côté « plusieurs ». Cette clé étrangère référence la clé primaire de la table qui est du côté « un ».
  - Une association plusieurs-à-plusieurs se traduit en ajoutant une nouvelle table. La clé primaire de cette nouvelle table sera composée des deux clés étrangères qui référencent les deux tables à associer.
  - Une association un-à-un se traduit en ajoutant une clé étrangère dans une des deux tables. Cette clé étrangère référence la clé primaire de l'autre table.

Il est important de noter que différentes approches peuvent être utilisées pour gérer l'héritage dans le modèle relationnel qui sont :

- Décomposition par distinction : Elle consiste à créer une table distincte pour chaque classe concrète impliquée dans l'héritage. Chaque table contient les attributs spécifiques à la classe correspondante, ainsi que les attributs hérités.

- Décomposition descendante (push-down) : Dans cette approche, les attributs d'une superclasse sont migrés vers ses sous-classes lors de la création des tables. Ainsi, seules les sous-classes deviennent des tables dans le modèle relationnel.
- Décomposition ascendante (push-up) : Contrairement à la décomposition descendante, la décomposition ascendante pousse les attributs spécifiques des sous-classes vers les superclasses. Ainsi, seules les superclasses deviennent des tables dans le modèle relationnel [8].

Après avoir appliqué les règles de passage au modèle relationnel, et en utilisant l'approche de décomposition "push down", nous avons obtenu le schéma suivant :

- (*id\_responsable\_relex, nom\_responsable\_relex, prenom\_responsable\_relex, sexe\_responsable\_relex, email\_responsable\_relex, mdp\_responsable\_relex, responsable\_relex*)
- Missionnaire (*id\_missionnaire, nom\_missionnaire, structure\_missionnaire*)
- Etablissement (*id\_etablissement, nom\_etablissement, type\_etablissement, categorie\_etablissement, ville\_etablissement, telephone\_etablissement, fax\_etablissement, email\_etablissement*)
- BonDeCommandeHotel (*id\_BDC\_hotel, num\_BDC, structure, liste\_personnes, nbr\_personnes, nom\_hotel, lieu\_hotel, date\_deb\_sejour, date\_fin\_sejour, directeur, responsable, sous\_directeur, signataire, hebergement, restauration, demi\_pension, pension\_complete, type\_chambre, autre\_prestation, path\_DPC, path\_BDC, date\_traitement, statut, #id\_facture\_hotel*)
- BonDeCommandeTAA (*id\_BDC\_aerien\_affrete, num\_BDC, structure, liste\_personnes, nbr\_personnes, lieu\_depart, lieu\_arrivee, directeur, responsable, sous\_directeur, signataire, num\_vol, type\_vol, date\_vol, path\_DATT, path\_BDC, date\_traitement, statut*)
- BonDeCommandeTAC (*id\_BDC\_aerien\_commercial, num\_BDC, structure, liste\_personnes, nbr\_personnes, compagnie, date\_traitement, statut, #id\_facture\_commercial*)
- FactureHotel (*id\_facture\_hotel, num\_facture, fe, date\_facture, montant\_total, nbr\_nuites, liste\_personnes\_facture, nbr\_personnes\_facture, date\_traitement*)
- FactureTAC (*id\_facture\_commercial, num\_facture, date\_facture, montant\_total, liste\_personnes\_facture, nbr\_personnes\_facture, date\_traitement*)

## **2.5 Conclusion**

Dans ce chapitre nous avons fourni une vision d'ensemble de notre travail, en combinant la méthodologie de conception SCRUM avec l'utilisation des différents diagrammes UML. Grâce à ces outils, nous avons pu analyser et concevoir l'application. Les diagrammes de cas d'utilisation et de séquence ont permis de modéliser la phase d'analyse, tandis que le diagramme de classes a été utilisé pour modéliser la structure et les relations entre les différentes classes qui composent le système. Nous passerons maintenant au chapitre suivant dédié à la mise en œuvre et à la réalisation de notre application.

# Chapitre 3

## Réalisation

### 3.1 Introduction

Nous entamons maintenant la phase de concrétisation de notre application, nous débuterons par présenter l'environnement de développement et les outils utilisés pour développer les différentes fonctionnalités de notre application. Nous aborderons également les Frameworks et packages spécifiques qui ont enrichi notre application avec des fonctionnalités supplémentaires. De plus, nous discuterons des outils utilisés pour le déploiement, notamment la génération de l'exécutable et la création du fichier d'installation, qui ont facilité la livraison de notre produit final. Enfin, nous finirons par vous présenter quelques interfaces de notre application.

### 3.2 Outils de développement

#### 3.2.1 Environnements de développement

##### 3.2.1.1 Visual Studio Code

Visual Studio Code (VS Code) est un éditeur de code multiplateforme, open source et extensible développé par Microsoft pour Windows, Linux et macOS supportant un très grand nombre de langages de programmation tels que C/C++, Java, PHP, Python, JavaScript, et bien d'autres. Grâce à des mises à jour régu-



lières, VS Code reste à jour avec les dernières avancées technologiques [9]

### 3.2.1.2 DB Browser for SQLite

DB Browser for SQLite (DB4S) est un outil visuel open source de haute qualité conçu pour créer, concevoir et manipuler des bases de données SQLite en utilisant une interface familière de type feuille de calcul. Il offre une interface conviviale et intuitive, permettant de travailler avec des bases de données sans nécessiter un serveur externe. Étant donné que nous avons utilisé SQLite, une base de données relationnelle utilisant le langage SQL, cet outil a été l'outil idéal pour naviguer et interagir avec notre base de données [10].



### 3.2.1.3 Visual Paradigm

Visual Paradigm est un outil de modélisation qui permet de créer des diagrammes UML et d'autres modèles visuels pour représenter les aspects d'un système informatique. Grâce à Visual Paradigm, nous avons pu définir l'architecture de notre application de manière claire, visualiser les relations entre les classes et les interactions entre les composants [11].



### 3.2.1.4 Adobe Illustrator

Adobe Illustrator est un logiciel de graphisme informatique produit en 1987 par Adobe Inc qui permet aux utilisateurs de créer des dessins, des designs et des mises en page sophistiqués. C'est l'une des nombreuses innovations d'Adobe qui ont révolutionné le design graphique. Dans notre cas, nous avons choisi de l'utiliser pour donner une touche personnelle à la création de notre logo [12].



## 3.2.2 Langages de programmation

### 3.2.2.1 HTML

HTML (HyperText Markup Language) est un langage de balisage utilisé pour structurer et présenter le contenu des pages web. Il utilise des balises pour définir la structure des éléments tels que les en-têtes, les paragraphes, les listes et les liens hypertextes [13].



### 3.2.2.2 CSS

CSS (Cascading Style Sheets) signifie feuilles de style en cascade, c'est un langage utilisé pour décrire le style des pages HTML, il décrit comment les éléments doivent être affichés (couleurs, tailles, espacements, ...etc) [14].



### 3.2.2.3 JavaScript

JavaScript est un langage de programmation utilisé pour ajouter des fonctionnalités interactives et dynamiques aux pages web. Il permet de manipuler le contenu HTML, de gérer les comportements des utilisateurs, d'effectuer des requêtes réseau et d'apporter des modifications visuelles en temps réel. JavaScript est exécuté côté client, directement dans le navigateur ou l'environnement d'exécution, ce qui en fait un outil essentiel pour le développement web moderne [15].



### 3.2.2.4 SQL

SQL (Structured Query Language) est un langage de programmation utilisé pour gérer et interagir avec des bases de données relationnelles. Il permet de stocker, manipuler et récupérer des données de manière structurée en utilisant des commandes telles que SELECT, INSERT, UPDATE et DELETE. Il permet également de créer des relations entre différentes tables à l'aide de clés primaires et étrangères, permettant ainsi de réaliser des opérations complexes telles que les jointures entre tables [16].



## 3.2.3 Environnement d'exécution et Frameworks

### 3.2.3.1 Node.js

Node.js est un environnement d'exécution côté serveur qui permet d'exécuter du code JavaScript en dehors d'un navigateur web. Il repose sur le moteur JavaScript V8 de Google Chrome et offre aux développeurs la possibilité de créer des applications web hautes performances et évolutives. Node.js est également connu pour son vaste écosystème de modules, disponibles via le gestionnaire de paquets npm. Ces modules offrent une multitude de fonctionnalités prêtes à l'emploi, ce qui facilite le développement d'applications web complexes [17].



### 3.2.3.2 Electron JS

Electron.js est un framework open source qui permet de créer des applications de bureau multiplateformes en utilisant des technologies web telles que HTML, CSS et JavaScript. Il utilise le moteur de rendu Chromium, le même que celui utilisé par Google Chrome, pour afficher l'interface utilisateur de l'application. L'avantage d'Electron.js réside dans le fait qu'il permet aux développeurs de créer des applications de bureau à partir des compétences qu'ils ont déjà acquises en développement web. Ils peuvent ainsi utiliser des langages familiers et des outils de développement pour créer des applications de bureau puissantes et interactives [18].



Pour développer notre application de bureau avec les technologies du web, on a suivi les étapes suivantes :

- Initialiser un projet Node.js en utilisant le gestionnaire de paquets « npm » avec la commande :  
`npm init -y`.
- Installer Electron avec la commande : `npm install electron`.
- Faire des ajustements nécessaires sur le fichier `package.json` présent à la racine de notre projet.
- Créer un script `main.js` avec un code JavaScript qui permet d'initialiser l'application Electron avec une fenêtre d'authentification.

De cette manière, nous pouvons programmer l'application en utilisant des technologies web telles

que HTML, CSS et JavaScript, offrant ainsi une approche familière pour le développement de l'interface utilisateur et des fonctionnalités de l'application de bureau.

### 3.2.3.3 Bootstrap

Bootstrap est un framework front-end de développement web populaire et largement utilisé. Il fournit un ensemble de styles prédéfinis, de composants interactifs et de scripts JavaScript pour aider les développeurs à créer rapidement et facilement des sites web réactifs et attrayants. Un autre avantage de bootstrap est sa documentation complète et ses exemples prêts à l'emploi [19].



## 3.2.4 Bibliothèques utilisées

### 3.2.4.1 SweetAlert2

SweetAlert2 est une bibliothèque JavaScript largement adoptée pour créer des boîtes de dialogue attrayantes et personnalisées. Nous l'avons utilisée pour afficher des messages de confirmation, des avertissements, des erreurs et des informations à l'utilisateur de notre application [20].

### 3.2.4.2 SQLite3

SQLite3 est une librairie pour Node.js qui permet d'interagir avec des bases de données SQLite. Elle fournit les fonctionnalités nécessaires pour manipuler des données dans une base de données SQLite à partir d'une application Node.js [21].

### 3.2.4.3 Font Awesome

Font Awesome est une bibliothèque d'icônes largement utilisée qui offre une grande variété d'icônes afin d'enrichir les interfaces et améliorer leur apparence visuelle. On l'a utilisée en installant Font awesome via la commande : `npm install @fortawesome/fontawesome-free`, puis en incluant le fichier CSS de Font Awesome dans notre fichier HTML. Pour afficher une icône, il suffit d'utiliser une balise comme celle-ci : `<i class = "fa-sharp fa-solid fa-house"></i>` [22].

#### **3.2.4.4 Mammoth**

Mammoth est une bibliothèque Node.js utilisée pour convertir des documents Word (.docx) en HTML. Elle fournit une solution simple et efficace pour extraire le contenu d'un document Word et le convertir en HTML structuré. Dans notre cas, on l'a utilisé simplement pour extraire le contenu textuel des documents Word insérés par l'utilisateur [23].

#### **3.2.4.5 Docxtemplater**

Docxtemplater est une bibliothèque Node.js utilisée pour générer des documents Word (.docx) à partir de modèles préexistants. Elle permet de remplir des modèles de documents Word avec des données dynamiques. On l'a utilisé pour générer des bons de commande à partir de nos modèles et les informations saisies par l'utilisateur [24].

#### **3.2.4.6 PizZip**

PizZip est une bibliothèque pour manipuler des fichiers .zip avec JavaScript. On l'a utilisé pour compresser notre modèle avant d'utiliser Docxtemplater pour générer le bon de commande [25].

### **3.2.5 Outils de déploiement**

Pour le déploiement de notre application, nous avons utilisé electron-packager pour générer l'exécutable et Inno Setup Compiler pour générer un fichier d'installation.

#### **3.2.5.1 Electron Packager**

Electron-packager est un outil de développement pour la plateforme Electron, qui permet de créer des exécutables pour différentes plateformes (Windows, macOS, Linux) à partir d'une application Electron

#### **3.2.5.2 Inno Setup Compiler**

Inno Setup Compiler est un outil de création d'installateurs pour les programmes informatiques. Il permet aux développeurs de regrouper leurs applications dans un package d'installation convivial et les

distribuer facilement aux utilisateurs. Son interface simple et intuitive en fait un choix populaire parmi les développeurs, facilitant ainsi la création d'installateurs personnalisés [26].

### **3.3 Présentation de quelques interfaces de l'application**

Avant de plonger dans la présentation des différentes interfaces de notre application, nous souhaitons présenter notre logo. Celui-ci est constitué des lettres "R" et "X" fusionnées, formant un design distinctif en référence à "RELEX". Sa couleur orange rappelle l'identité visuelle de Sonatrach, rappelant ainsi l'alignement de notre solution avec l'identité visuelle de l'organisation.



FIGURE 3.1 – Logo de notre application.

#### **3.3.1 L'interface d'authentification**

L'interface d'authentification a été conçue pour répondre à la demande de l'utilisateur en simplifiant l'expérience d'accès au compte, ne nécessitant que la saisie du mot de passe. Si le mot de passe saisi est incorrect, le système affiche un message d'erreur.

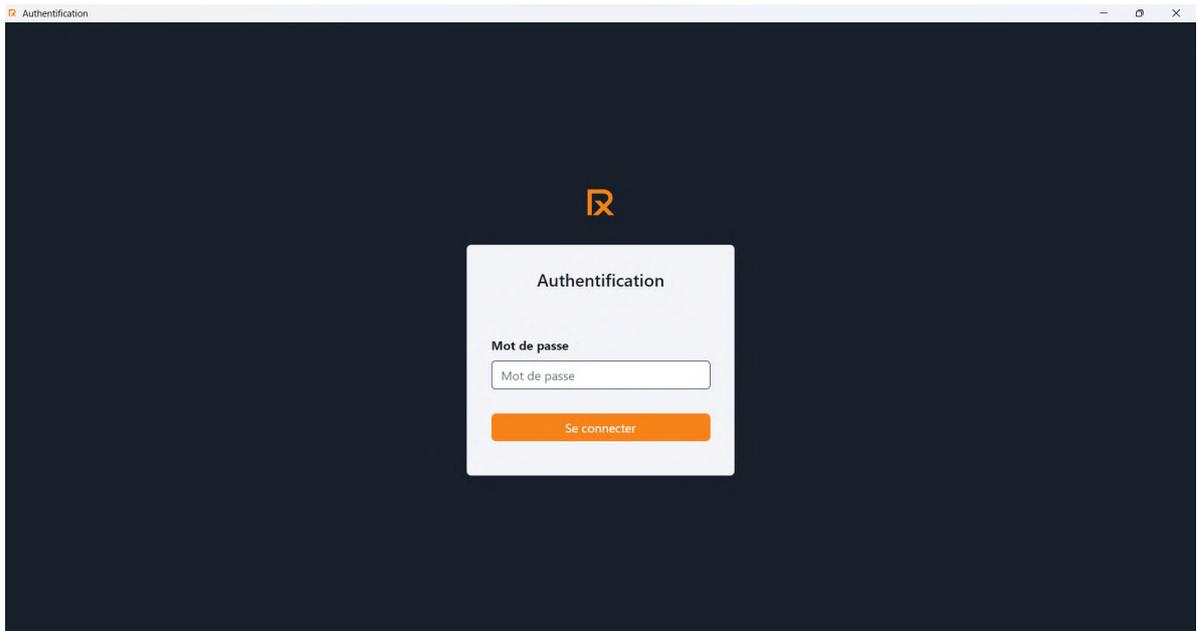


FIGURE 3.2 – L’interface d’authentification.

### 3.3.2 L’interface d’accueil utilisateur

Lorsque vous accédez à l’interface d’accueil, vous trouverez une barre de navigation latérale à gauche qui reste visible même lorsque vous explorez d’autres parties de l’application. Cette barre de navigation offre un ensemble de boutons avec des icônes représentatives pour les différentes fonctionnalités suivantes :

- Accueil
- Générer BDC pour les catégories hôtel, TAA et TAC
- Lister les bons de commande pour les catégories hôtel, TAA et TAC
- Générer facture pour les catégories hôtel et TAC
- Listing des factures pour les catégories hôtel et TAC
- Reporting pour les catégories hôtel et TAC
- Gestion des missionnaires
- Gestion des établissements
- Profil
- Déconnexion

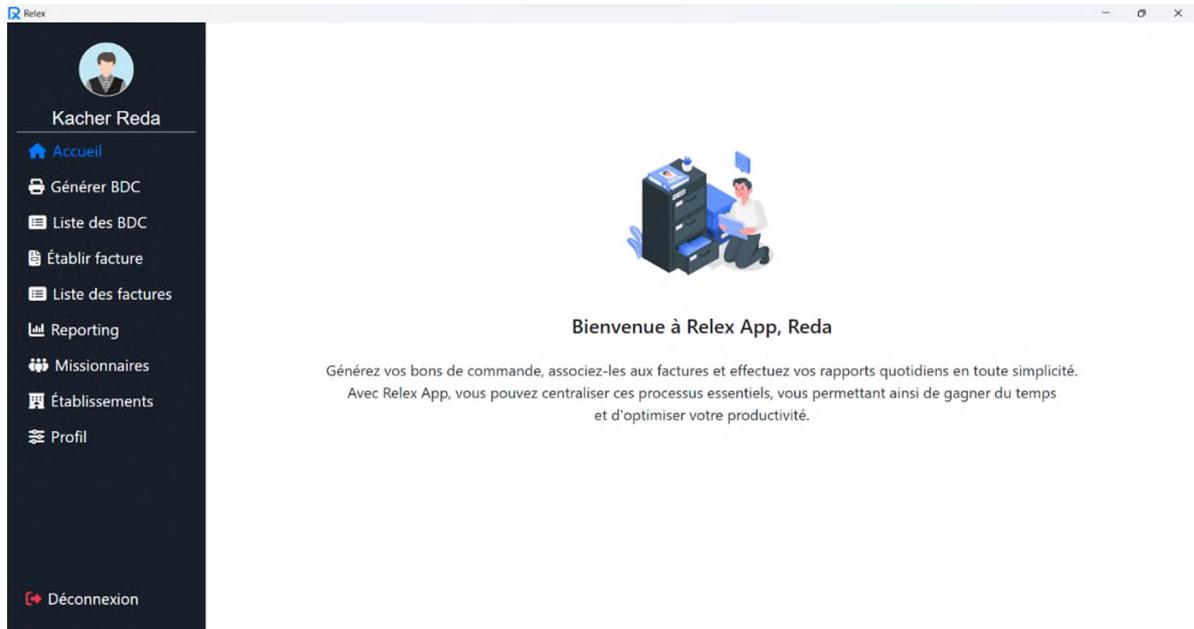


FIGURE 3.3 – L'interface d'accueil utilisateur.

#### **Remarque :**

Dans le but d'illustrer toutes les interfaces, nous avons décidé d'utiliser l'exemple du type "hôtel". Cependant, il est important de noter que toutes les fonctionnalités et interfaces sont disponibles pour les autres types à savoir TAA et TAC, offrant ainsi une expérience adaptée à divers scénarios d'utilisation.

### **3.3.3 L'interface de génération d'un bon de commande**

Quand l'utilisateur clique sur "Générer bon de commande" et sélectionne le type du bon de commande qu'il souhaite générer (Hôtel, TAA ou TAC), une interface dédiée s'affiche, offrant deux options : la génération automatique et la génération manuelle. Cette fonctionnalité permet à l'utilisateur de choisir la méthode qui correspond le mieux à ses besoins et préférences pour créer le bon de commande souhaité.



FIGURE 3.4 – Interface du choix de la méthode de génération du bon de commande.

### 3.3.3.1 L'interface de génération automatique d'un bon de commande

L'utilisateur insère une demande de prise en charge format Word et le système lui affiche automatiquement un formulaire pré-rempli d'informations récupérées à partir de la demande de prise en charge.

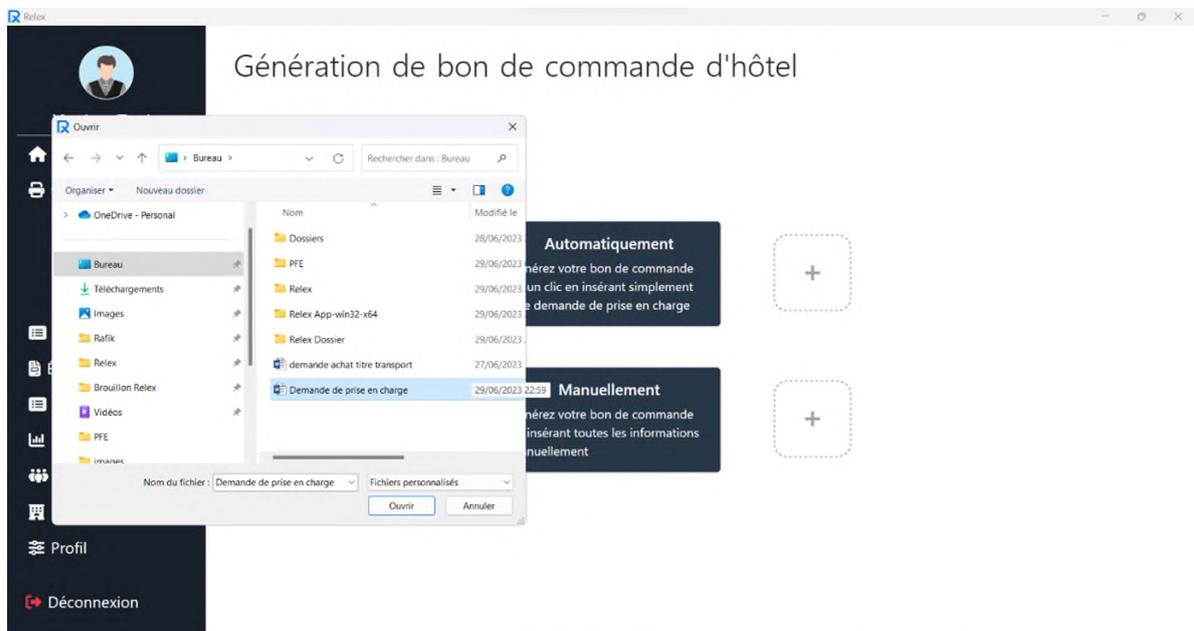


FIGURE 3.5 – L'interface de génération automatique d'un bon de commande.

L'utilisateur peut donc vérifier les informations et éventuellement les modifier avant de valider. Le

bon de commande au format Word s'affiche alors directement offrant à l'utilisateur la possibilité de l'imprimer.

### 3.3.3.2 L'interface de génération manuelle d'un bon de commande

Pour la génération manuelle, après que l'utilisateur ait choisi le nombre de missionnaires à prendre en charge, un formulaire est affiché et l'utilisateur doit remplir ses champs. En cas d'omission d'informations obligatoires ou de saisie de dates incohérentes (date de début du séjour supérieure à la date de fin du séjour), le système affiche des messages d'erreurs pour guider l'utilisateur et le conseiller à corriger les données saisies. De plus, l'utilisateur a la possibilité d'ajouter ou de supprimer des missionnaires selon les besoins.



FIGURE 3.6 – Interface du choix du nombre de missionnaires à prendre en charge.

FIGURE 3.7 – L’interface du formulaire de la génération manuelle d’un bon de commande.

Après avoir validé les informations du formulaire, le bon de commande au format Word s’ouvre automatiquement.

Voici le bon de commande généré dans les deux cas : automatiquement et manuellement.

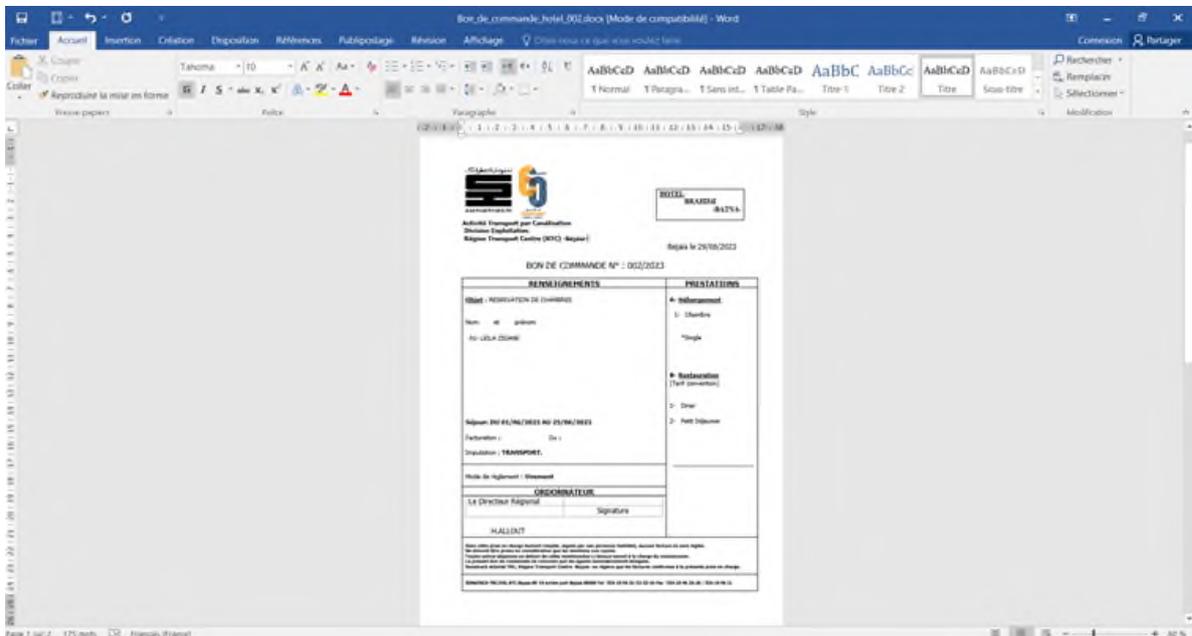


FIGURE 3.8 – Exemple d’un bon de commande d’hôtel généré.

Si le nombre de missionnaires pris en charge dans le bon de commande dépasse dix (10), une liste des missionnaires correspondants est jointe au bon de commande, comme illustré dans l'exemple suivant :

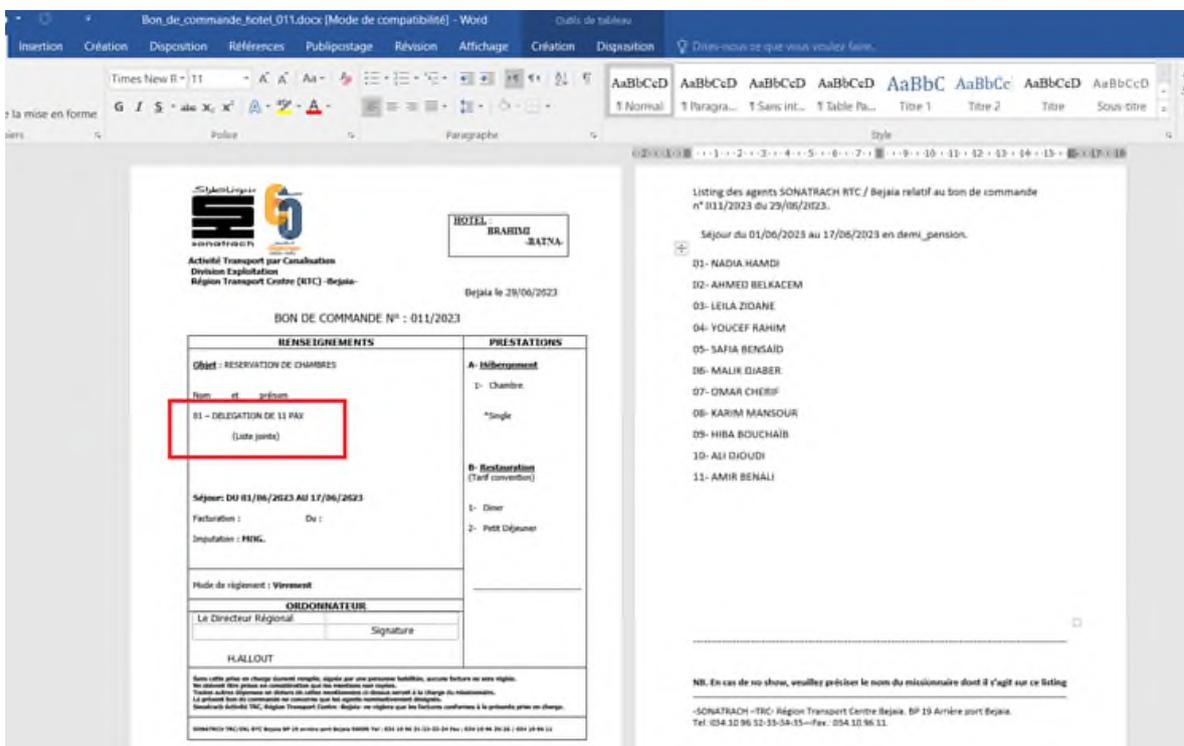


FIGURE 3.9 – Exemple d’une liste de missionnaires jointe au bon de commande.

**Remarque :** Tous les documents générés par l’application sont stockés de manière ordonnée dans un dossier créé automatiquement dans le bureau.

### 3.3.4 L'interface de la liste des bons de commande

Cette interface présente un tableau contenant les informations pertinentes de chaque bon de commande.

The screenshot shows a web application interface titled 'Bons de commande d'hôtel'. On the left is a dark sidebar menu with the user's name 'Kacher Reda' and various navigation options. The main content area features a table of orders and a legend for their statuses.

**Legend:**

- ✓ Bon de commande : VALIDE (2)
- ⊘ Bon de commande : ANNULÉ (1)
- 🔗 Bon de commande : ASSOCIÉ (7)

**Table:**

N°BDC	HOTEL	STRUCTURE	NBR_MISSIONNAIRES	DPC	BDC	STATUT	ÉDITION
010	BRAHIMI	ATR/ SCE TRANSPORT	1	📄	📄	✓	🔗   Éditer
009	HOTEL SAINT GEORGE	Centre informatique	1	📄	📄	🔗	🔗   Éditer
008	HOTEL EL KSAR	MOG	1	📄	📄	🔗	🔗   Éditer
007	EL AURASSI HOTEL	Transport	2	📄	📄	⊘	🔗   Éditer
006	BRAHIMI	Exploitation	1	📄	📄	🔗	🔗   Éditer
005	HOTEL OASIS	Centre Informatique	1	📄	📄	🔗	🔗   Éditer
004	HOTEL OASIS	Centre Informatique	1	📄	📄	✓	🔗   Éditer
003	EL AURASSI HOTEL	ATR	3	📄	📄	🔗	🔗   Éditer
002	BRAHIMI	Transport	1	📄	📄	🔗	🔗   Éditer
001	BRAHIMI	ATR	2	📄	📄	🔗	🔗   Éditer

FIGURE 3.10 – Liste des bons de commande.

L'utilisateur peut donc faire une recherche, ouvrir les documents Word associés à un bon de commande (Demande de prise en charge / Bon de commande) ou éditer un bon de commande.

### 3.3.5 L'interface d'édition d'un bon de commande

Dans cette interface, l'utilisateur a la possibilité de :

- Modifier les informations du bon de commande
- Changer son état, c'est-à-dire l'annuler ou le rétablir au cas où il est déjà annulé
- Supprimer le bon de commande

The screenshot shows the 'BON DE COMMANDE [HOTEL]' interface. The left sidebar is titled 'Kacher Reda' and lists various navigation options. The main content area is titled 'BON DE COMMANDE [HOTEL]' and contains the following fields and controls:

- Missionnaire 1:** SAMIRA AIT SAID
- Numéro bon de commande:** 004
- Date début de séjour:** 29/01/2023
- Date fin de séjour:** 08/06/2023
- Structure:** Centre Informatique
- Nom de l'hôtel:** HOTEL OASIS
- Lieu de l'hôtel:** BISKRA
- Ordonnateur:**
  - Directeur:** Le Directeur Régional
  - Responsable:** (empty)
  - P. NOM:** H.ALLOUT
- Prestations:**
  - Hébergement:**
  - Restauration:**
  - Demi-pension:**
  - Pension complète:**
- Type de chambre:** Single
- Autre:** (empty)

At the bottom of the form, there are four buttons: 'Valider', 'Supprimer', 'Annuler', and 'Retour'.

FIGURE 3.11 – L'interface d'édition d'un bon de commande.

### 3.3.6 L'interface d'établissement d'une facture

Pour l'établissement d'une facture, l'utilisateur doit sélectionner le ou les bons de commande (préalablement générés) à associer à la facture puis saisir les informations nécessaires pour compléter celle-ci.

Dans le cas où l'utilisateur choisirait deux bons de commande provenant de deux hôtels différents, un message d'erreur est affiché pour l'informer que les bons de commande doivent être associés à un seul et même hôtel.

The screenshot displays the 'FACTURE [HOTEL]' interface. On the left is a dark sidebar with the user profile 'Kacher Reda' and a list of navigation items: Accueil, Générer BDC, Liste des BDC, Établir facture, Hôtel, Aérien [CO], Liste des factures, Reporting, Missionnaires, Établissements, Profil, and Déconnexion. The main content area is titled 'FACTURE [HOTEL]' and includes a 'Bon de commande' dropdown menu with options 004, 007, and 010, and a 'Visualiser' button. Below this are input fields for 'Numéro de facture' (254), 'FE' (261026), 'Date de la facture' (16/06/2023), and 'Montant Total' (16598). There is also a 'Nombre de nuités' field with the value 26. A section titled 'Bon de commande N°1' lists two items: '1- MALIK DJABER' and '2- HIBA BOUCHAIB', each with a red minus sign icon. A 'Valider' button is at the bottom right.

FIGURE 3.12 – L'interface d'établissement d'une facture.

### 3.3.7 L'interface de la liste des factures

L'interface "Liste des factures" présente un tableau contenant les informations pertinentes de chaque facture.

La fonction de recherche permet à l'utilisateur de filtrer les factures en fonction de toutes les informations disponibles sur le tableau telles que : la structure, le montant, la date de traitement. Offrant ainsi une recherche précise et flexible.

ÉTABLISSEMENT	N°BDC	STRUCTURE	N°FACTURE	FE	DATE_FACTURE	MONTANT	DATE_TRAITEMENT	ÉDITION
HOTEL SAINT GEORGE	009	Centre informatique	06	606461	2023-06-10	68794 DA	2023-06-29 22:20:13	<a href="#">Éditer</a>
BRAHIMI	002,006	Transport	02	202120	2023-06-16	48751 DA	2023-06-29 22:15:02	<a href="#">Éditer</a>
HOTEL OASIS	005	Centre Informatique	04	404104	2023-06-22	98547 DA	2023-06-29 22:18:12	<a href="#">Éditer</a>
EL AURASSI HOTEL	003	ATR	03	303130	2023-06-23	9874 DA	2023-06-29 22:17:22	<a href="#">Éditer</a>
BRAHIMI	001	ATR	01	101011	2023-06-17	154986 DA	2023-06-29 22:13:33	<a href="#">Éditer</a>
HOTEL EL KSAR	008	MOG	05	505150	2023-06-25	467951 DA	2023-06-29 22:18:55	<a href="#">Éditer</a>

FIGURE 3.13 – l'interface de la liste des factures.

L'utilisateur peut par la suite générer un document Word contenant la liste des factures.

## Liste des factures

### Hôtel

Établissement	N° BDC	Structure	N° Facture	Date	FE	Nombre Nuitées	Nombre personnes	Montant
HOTEL SAINT GEORGE	009	Centre informatique	06	2023-06-10	60646 1	16	1	68794 DA
BRAHIMI	002, 006	Transport	02	2023-06-16	20212 0	45	2	48751 DA
HOTEL OASIS	005	Centre Informatique	04	2023-06-22	40410 4	13	1	98547 DA
EL AURASSI HOTEL	003	ATR	03	2023-06-23	30313 0	98	3	9874 DA
BRAHIMI	001	ATR	01	2023-06-17	10101 1	24	2	154986 DA
HOTEL EL KSAR	008	MOG	05	2023-06-25	50515 0	64	1	467951 DA
<b>TOTAL</b>	-	-	-	-	-	<b>260</b>	<b>10</b>	<b>848903DA</b>

FIGURE 3.14 – Exemple d’une liste des factures générée.

### 3.3.8 L’interface d’édition d’une facture

L'utilisateur a la possibilité de :

- Modifier les informations relatives à la facture
- Supprimer la facture, ce qui changera l'état du bon de commande automatiquement de 'Associé' à 'Valide'.

The screenshot displays the 'FACTURE [HOTEL]' editing interface. On the left is a dark sidebar with the user's name 'Kacher Reda' and a list of navigation items: Accueil, Générer BDC, Liste des BDC, Établir facture, Liste des factures, Hôtel, Aérien [CO], Reporting, Missionnaires, Établissements, Profil, and Déconnexion. The main content area is titled 'FACTURE [HOTEL]' and contains the following elements:

- Bon de commande:** A text field with '005' and a 'Visualiser' button.
- Facture Details:** A table-like structure with four columns: 'Numéro de facture' (04), 'FE' (404104), 'Date de la facture' (22/06/2023), and 'Montant Total' (98547).
- Nombre de nuitées:** A text field with '13'.
- Items:** A list with one item: '1- YOUCEF RAHIM' with a red minus icon.
- Actions:** Three buttons at the bottom: 'Valider' (blue), 'Supprimer' (red), and 'Retour' (red).

FIGURE 3.15 – L’interface d’édition d’une facture.

### 3.3.9 L'interface du reporting

Cette interface permet à l'utilisateur de générer des rapports de réalisations. Il doit spécifier la période, et il peut éventuellement filtrer par hôtel ou par lieu pour le reporting hôtel, par compagnie ou par structure pour le reporting TAC. De plus, il peut choisir d'inclure ou d'exclure les bons de commande annulés en cochant la case "BDC annulés". Cela offre une personnalisation des rapports à générer en fonction des besoins de l'utilisateur.

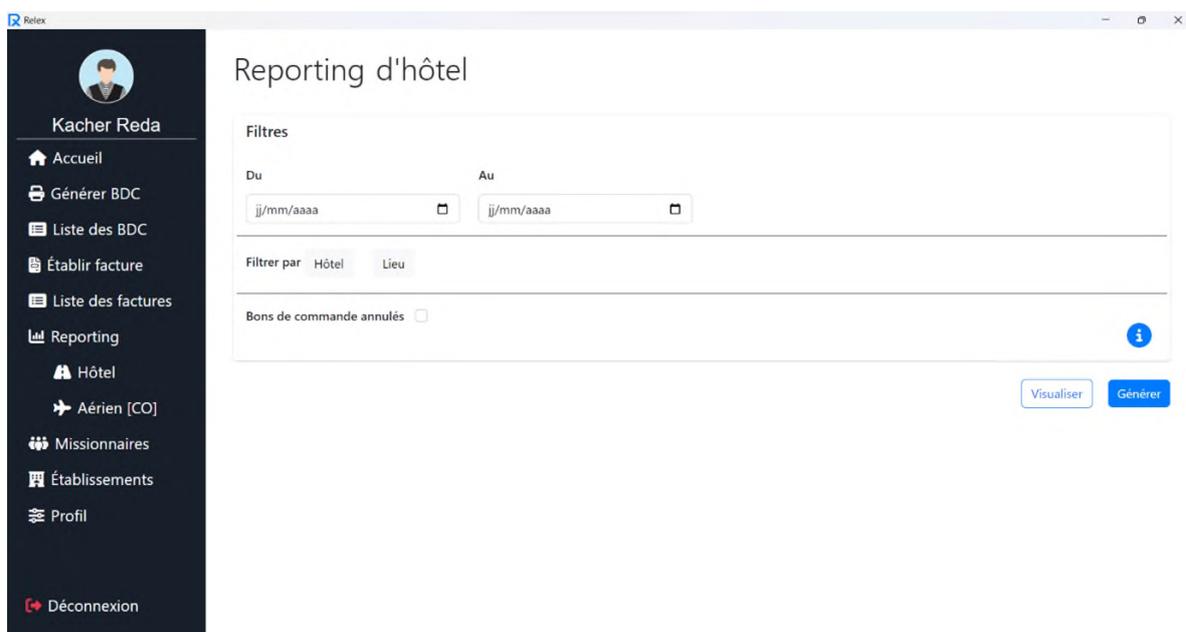


FIGURE 3.16 – L'interface du reporting.

Après avoir cliqué sur le bouton "Visualiser", les résultats du rapport s'affichent. L'utilisateur a la possibilité d'examiner les résultats directement dans l'interface. Cependant, s'il souhaite générer les résultats, il peut cliquer sur le bouton "Générer". Dans ce cas, un document Word contenant le rapport est généré, offrant la possibilité à l'utilisateur de l'imprimer.

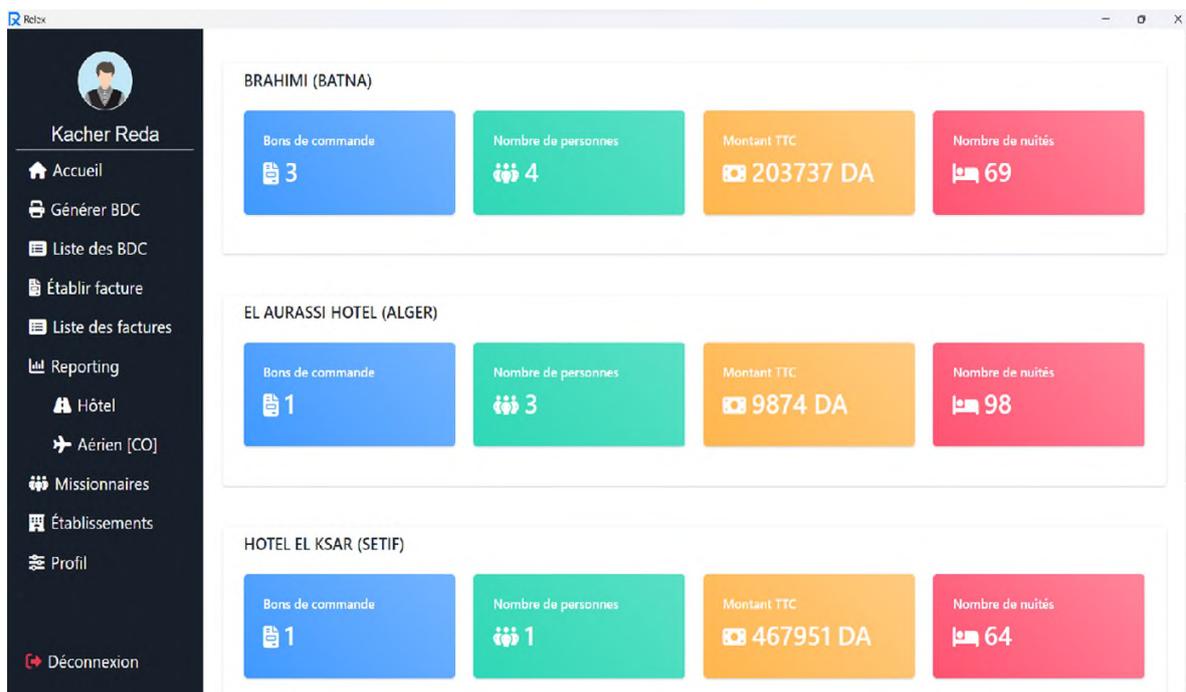


FIGURE 3.17 – Exemple de résultats d'un rapport.

### État des réalisations hôtelières

Période  
Du 29/01/2023 au 29/06/2023

Établissement	Localisation	Nombre de BDC placés	Nombre de personnes	Nombre de nuitées	Chiffre d'affaires
BRAHIMI	BATNA	3	4	69	203737 DA
HOTEL SAINT GEORGE	TIZI OUZOU	1	1	16	68794 DA
EL AURASSI HOTEL	ALGER	1	3	98	9874 DA
HOTEL EL KSAR	SETIF	1	1	64	467951 DA
HOTEL OASIS	BIKSRA	1	1	13	98547 DA
<b>Total</b>	-	<b>7</b>	<b>10</b>	<b>260</b>	<b>848903 DA</b>

FIGURE 3.18 – Exemple d'un document Word d'un rapport généré.

Pour faciliter la compréhension du reporting, nous avons veillé à ce que le titre et la période soient indiqués sur toutes les pages du document. Ainsi, quelle que soit la page où vous vous trouvez dans le rapport, vous pouvez facilement identifier le contenu correspondant au titre et à la période sélectionnée.

### 3.3.10 L'interface de gestion des établissements

Dans cette interface, l'utilisateur a accès à toutes les informations relatives à un établissement (Hôtel, compagnie aérienne ou restaurant). Il peut rechercher, ajouter, modifier, ou supprimer un établissement. De plus, il a la possibilité d'effectuer des recherches pour retrouver rapidement un missionnaire spécifique.



FIGURE 3.19 – L'interface de la liste des établissements.

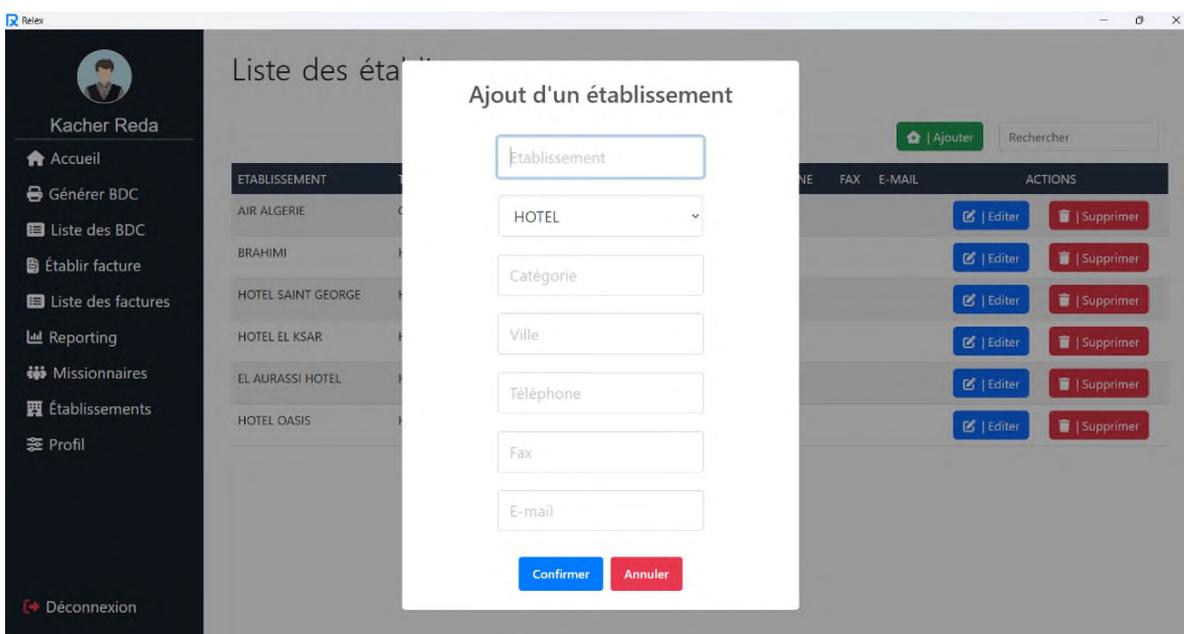


FIGURE 3.20 – L'interface d'ajout d'un établissement.

### 3.3.11 L'interface de gestion des missionnaires

Dans cette interface, l'utilisateur a accès à toutes les informations relatives aux missionnaires. Il peut rechercher, ajouter, modifier ou supprimer un missionnaire. De plus, il a la possibilité d'effectuer des recherches pour retrouver rapidement un missionnaire spécifique.

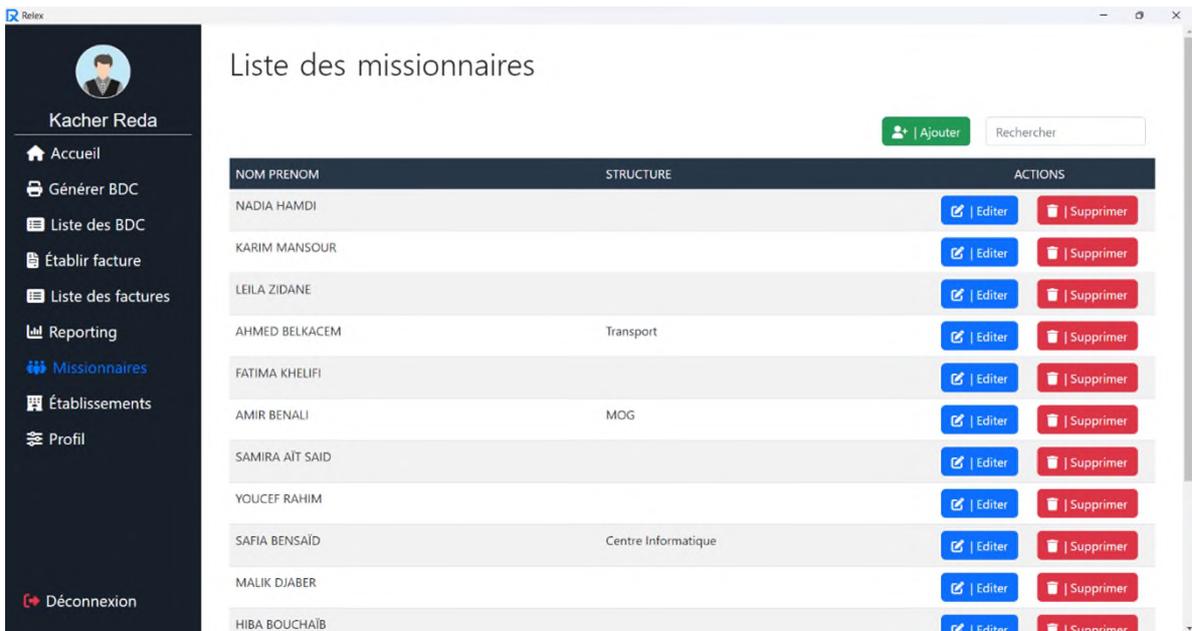


FIGURE 3.21 – L'interface de la liste des missionnaires.

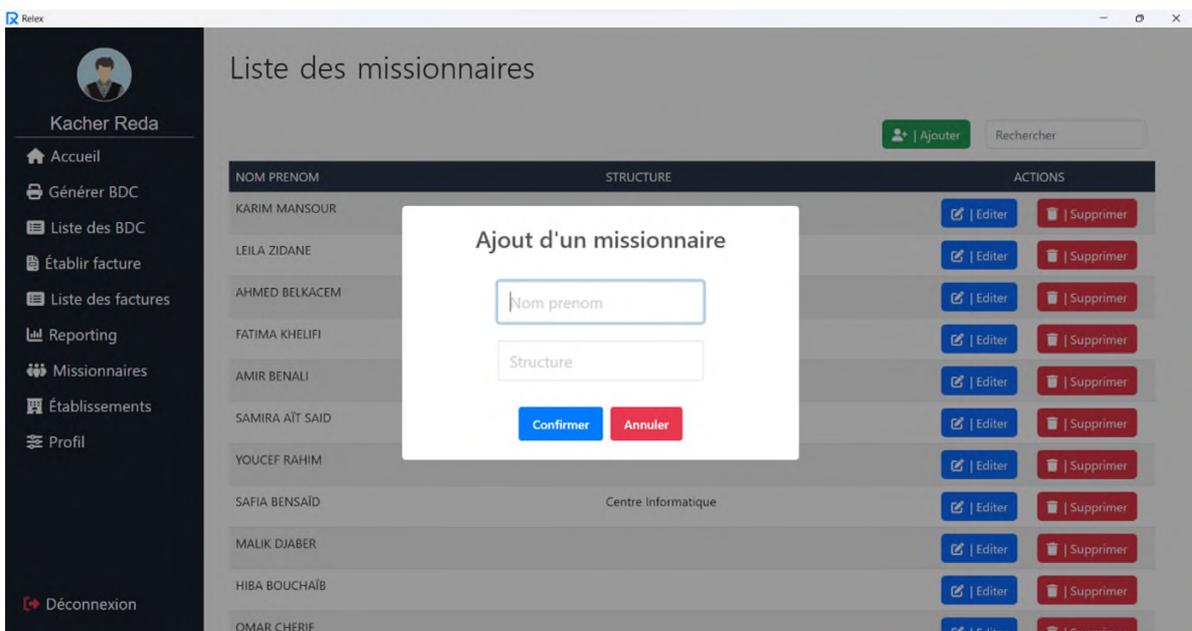


FIGURE 3.22 – L'interface d'ajout d'un missionnaire.

### 3.3.12 L'interface de gestion du profil utilisateur

L'interface de gestion du profil utilisateur permet à l'utilisateur de consulter et de modifier ses informations personnelles, telles que son nom, prénom, e-mail, mot de passe, sexe et photo de profil.

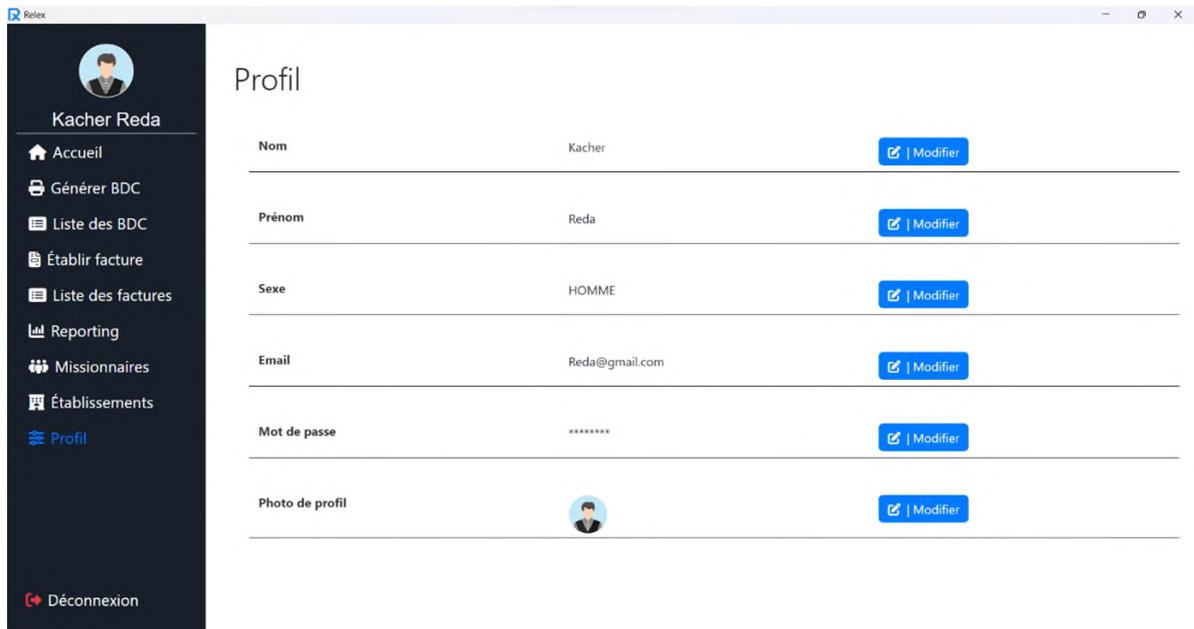


FIGURE 3.23 – L'interface de gestion du profil utilisateur.

## **3.4 Conclusion**

Dans ce chapitre, nous avons présenté la partie réalisation de notre application de gestion de la section RELEX de Sonatrach de Béjaïa. Nous avons présenté les outils de développement utilisés et avons illustré différentes interfaces de notre application, mettant en évidence son fonctionnement et ses scénarios applicatifs. Ce chapitre marque l'aboutissement de notre projet, où nous avons pu concrétiser notre solution grâce à la mise en œuvre d'une pile technologique adaptée.

# Conclusion générale

En guise de projet de fin de cycle Master, option Génie Logiciel, nous avons travaillé sur la conception et la réalisation d'une application desktop nommée Relex App. L'objectif de ce projet était d'aider le responsable de la section RELEX de l'entreprise Sontarach à automatiser ses tâches quotidiennes, telles que la génération de bons de commande, l'association de ces derniers à leurs factures et la génération de rapports personnalisés.

Pour ce faire, nous avons commencé dans le premier chapitre par introduire l'organisme d'accueil et réaliser une analyse approfondie de leur système existant. Nous avons identifié les problèmes spécifiques liés à la gestion quotidienne de la section RELEX et nous avons proposé des solutions adaptées pour y remédier. Dans le deuxième chapitre, nous avons établi une étude préliminaire visant à identifier les besoins fonctionnels et non fonctionnels de notre client ainsi que les différents acteurs qui réagissent avec le système à développer. À la suite de cette étude, nous avons exposé la méthode de développement adoptée, à savoir SCRUM. Ensuite, nous avons utilisé UML pour présenter la conception de notre application. Nous avons employé spécifiquement les diagrammes de cas d'utilisation, de séquence et de classes. À partir du diagramme de classes, nous avons obtenu le modèle relationnel qui a servi à la création de notre base de données. Dans le dernier chapitre, nous avons abordé la partie pratique en présentant les outils de développement et les langages de programmation utilisés. Après cela, nous avons illustré la plupart des interfaces de notre application à l'aide de captures d'écran, en fournissant des explications détaillées pour chacune d'entre elles. À la fin de ce projet, nous prévoyons de maintenir notre application ouverte à de futures améliorations. Étant donné qu'elle est basée sur des technologies web, nous envisageons de la transformer en une application web afin d'inclure un espace dédié aux structures internes de SONATRACH. Cela permettra l'envoi direct des documents au responsable RELEX via l'application. De plus, nous prévoyons de fournir un espace spécifique aux établissements externes,

afin qu'ils puissent recevoir les bons de commande et envoyer les factures directement via l'application. Pour mettre en œuvre ces fonctionnalités supplémentaires, nous envisageons d'implémenter l'application en utilisant Oracle Database. Ce SGBD est plus adapté en raison de ses qualités telles que sa haute performance, sa fiabilité, sa stabilité et sa sécurité avancée.

# Bibliographie

[1] Site de la Sonatrach, <https://sonatrach.com>, Consulté le 13/02/2023.

[2] Belmazouzi, Y. (2021). Développement et validation d'une approche de décision sociotechnique liée aux problèmes d'industrialisation en Algérie (Thèse de doctorat, Magister en Hygiène et Sécurité Industrielle, option : Gestion des risques). Institut d'Hygiène et Sécurité Industrielle (IHSI), Université Chahid Mostefa Ben Boulaid de Batna 2.

[3] Touati, O. (2007). Rapport au travail et dynamique de transformation du métier d'ingénieur dans le contexte algérien : le cas de l'entreprise Sonatrach (Thèse de doctorat). Université de Montréal, Montréal, Canada. École des Hautes Études en Sciences Sociales, Paris, France.

[4] La région de transport du centre RTC-SONATRACK-Bejaia. Document interne RTC Béjaia. Mai 2015.

[5] Guide de démarrage Scrum, <https://agiliste.fr/guide-de-demarrage-scrum/>, Consulté le 16/02/2023.

[6] Pete Deemer, Gabrielle Benfield, Craig Larman, (2021), The Scrum Primer.

[7] Définition | UML, <https://www.futura-sciences.com/tech/definitions/informatique-uml-3979>, Consulté le 16/03/2023

[8] Identifiez les éléments clés du modèle relationnel, <https://openclassrooms.com/fr/courses/6938711-modelisez-vos-bases-de-donnees/>. Consulté le 20/03/2023

[9] What is Visual Studio Code ? Microsoft's extensible code editor, <https://www.infoworld.com/article/3666488/what-is-visual-studio-code-microsofts-extensible-code-editor.html?>, Consulté le 21/03/2023.

- [10] [sqlitebrowser](https://github.com/sqlitebrowser/), <https://github.com/sqlitebrowser/>. Consulté le 22/03/2023.
- [11] Agile project tools to improve productivity, <https://www.visual-paradigm.com/>, Consulté le 26/03/2023.
- [12] Optimisez et accélérez votre apprentissage, <https://openclassrooms.com/fr/courses/2583216-creez-des-visuels-avec-illustrator/>. Consulté le 29/03/2023
- [13] INTRODUCTION À HTML, [https://microapp.com/contenus\\_propres/fiches\\_produits/extraits\\_livres/3189/extrait.pdf](https://microapp.com/contenus_propres/fiches_produits/extraits_livres/3189/extrait.pdf), Consulté le 27/03/2023
- [14] Apprendre à coder en HTML - CSS, [http://projet.eu.org/pedago/sin/ICN/2nde/1-html\\_css.pdf](http://projet.eu.org/pedago/sin/ICN/2nde/1-html_css.pdf), Consulté le 27/03/2023
- [15] Emmanuel Gutierrez,(Septembre 2008),JavaScript - Des fondamentaux aux concepts avancés.
- [16] Philippe Rigaux,(Février 2023),Cours de bases de données – Modèles et langages.
- [17] Luc Juggery,Thomas Castelly,(2014),autour-de-nodejs.
- [18] What Is Electron.js ?, <https://brainhub.eu/library/what-is-electron-js>, Consulté le 07/04/2023.
- [19] Maurice Chavelli,(2014),Prenez en main Bootstrap.
- [20] What is the SweetAlert ?, <https://www.javatpoint.com/sweetalert>, Consulté le 13/04/2023.
- [21] node-sqlite3, <https://www.npmjs.com/package/sqlite3>, Consulté le 13/04/2023.
- [22] FONT AWESOME, <https://github.com/FortAwesome/Font-Awesome>, Consulté le 15/04/2023.
- [23] Mammoth .docx to HTML converter for Java/JVM, <https://github.com/mwilliamson/java-mammoth>, Consulté le 22/04/2023.
- [24] DOCX Templates, <https://www.sdocs.com/knowledge-base/sdocs/understanding-template-formats/docx-templates/>, Consulté le 24/04/2023.

[25] PizZip, <https://yarnpkg.com/package/pizzip>, Consulté le 24/04/2023.

[26] Utiliser Inno Setup pour le déploiement de vos applications, <https://gkemayo.developpez.com/innosetup/creer-installateur-avec-inno-setup/>, Consulté le 27/04/2023.

# Annexes

الجمهورية الجزائرية الديمقراطية الشعبية  
وزارة الطاقة



رقم : 2022 / ATR / 420

## تكليف بمهمة

Nom & prénom : MOULOUDJ Nabil	الاسم و اللقب :
Fonction : CHAUFFEUR PL 1D	الوظيفة :
Itinéraire : Béjaia- Alger- Bordj Ménaiel- Béjaia	خط السير :
Objet : Transport de M. ADRAR Malek	موضوع المهمة :
Date Départ : 14 / 03/ 2023	تاريخ الذهاب المقرر :
Date Retour : 15/ 03/ 2023	تاريخ الرجوع المقرر :
Moyens de transport : Véhicule Sonatrach	وسيلة النقل :
	بجاية في :

Bejaia le : 15/ 03/ 2023

Le Sous-Directeur Technique

A. AZZI

ANNEXE 1 – L'ordre de mission, source : L'entreprise Sonatrach.

سوناتراش



sonatrach

Activité Transport par  
Canalisations  
Division Exploitation  
Région Transport Centre-

A Monsieur le Chef du Département Moyens Généraux

Structure émettrice ATR/ SCE TRANSPORT

DEMANDE DE PRISE EN CHARGE

N°: X /2022 DU: 15/11/2022

Réservation

Prolongation

NOM & PRENOM	ORDRE MISSION		CATEGORIES (E-M)-(C)-(CS)	Date de séjour	
	N°	DATE		DU	AU
MOUMINI CAROL	480	15/11/2022	E	15/11/2022	16/11/2022

DESIGNATION DES PRESTATIONS	NATURES	OBSERVATIONS
Hébergement	OUI	
Restauration	OUI	
Demi-pension	OUI	
Pension complète	NON	
Hôtel Souhaité :	Zaytuni (BATNA)	

\* Mettre OUI ou NON devant la prestation demandée

P/Chef de Département ATR,  
 Le Responsable Intérimaire,

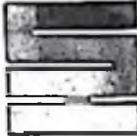
B.BOUNIF

NB :

- Demande à faire parvenir au département Moyens Généraux 72 Heures avant le départ du missionnaire ;
- Joindre obligatoirement une copie de l'ordre de mission ;
- Toute prestation non énumérée ci-dessus sera à la charge de l'intéressé.

ANNEXE 2 – Demande de prise en charge, source : L'entreprise Sonatrach.

سوناتراش



SONATRACH

SONATRACH  
ACTIVITE TRANSPORT PAR CANALISATIONS  
DIVISION EXPLOITATION  
REGION TRANSPORT CENTRE - BEJAIA

STRUCTURE EMETTRICE : R.H.U

A Monsieur

LE CHEF DEPARTEMENT MOG

### DEMANDE ACHAT TITRE DE TRANSPORT

N° : \_\_\_\_\_ DU

NOMS ET PRENOMS	FONCTIONS	OBSERVATIONS
1) RAHMOUNI née ZEMOURI Leila	Chargée d'Etudes RHU	

AERIEN <input checked="" type="checkbox"/>	MARITIME <input type="checkbox"/>
Type de vol : * régulier affrété	
Itinéraire : (Départ : (Béjaia - Alger - Oran - Alger - Béjaia)	
Date et heure de départ : Alger le 12/03/2023	
Date et heure de retour : Oran le 17/03/2023	
Dans le cas de non disponibilité de place réservation comme suit :	
Imputation : 40 20 10	

Réservée RELEX  
Numéro du titre de transport  
.....

Accusé de réception du bénéficiaire  
Reçu le \_\_\_\_\_

Le Responsable  
De la Structure

Le Sous-Directeur Administration et Finances,

Y. KRIM

NB : - Joindre obligatoirement une copie de l'ordre de mission  
- Restituer les billets non utilisés et les souches à l'issu de chaque mission





Passenger: Rahmouni Leila (ADT)  
Booking ref: SZ72R6  
Ticket number: 124 2425665435



Issuing office:  
AH BEJAIA LIBERTE CTO, 08 RUE DE LA  
LIBERTE, BEJAIA, Telephone: 034 21 13  
37  
Date: 02Mar2023

### ELECTRONIC TICKET RECEIPT

At check-in, you must provide valid travel documents. Check-in opens 03 hours before departure of flight.

Flight	Origin	Destination	Class	Departure	Arrival	Duration
ALGIERS HOUARI BOUMEDIENE Terminal: D	ORAN AHMED BEN BELLA	AH6182	14:20	15:35	13:35	01:15
Class: Y		Operated by: AIR ALGERIE Marketed by: AIR ALGERIE Booking status (1): OK		Duration: 01:15		
Baggage (4): 20K Fare basis: Y						
ORAN AHMED BEN BELLA	ALGIERS HOUARI BOUMEDIENE Terminal: D	AH6181	08:05	09:20	07:20	01:15
Class: Y		Operated by: AIR ALGERIE Marketed by: AIR ALGERIE Booking status (1): OK		Duration: 01:15		
Baggage (4): 20K Fare basis: Y						
ALGIERS HOUARI BOUMEDIENE Terminal: D	BEJAIA SOUMMAM ABANE RAMDANE	AH6052	13:00	13:30	12:15	00:30
Class: Y		Operated by: AIR ALGERIE Marketed by: AIR ALGERIE Booking status (1): OK		Duration: 00:30		
Baggage (4): 20K Fare basis: Y						

(1) OK = confirmed (2) NVB = Not valid before (3) NVA = Not valid after (4) Baggage: K = Kilo, PC=Piece. You may check-in the weight/  
number of baggage items listed above free of charge. Beyond this limit, you must pay an extra fee.

For more details check out on our website [www.airalgerie.dz](http://www.airalgerie.dz) our Baggage Policy and General Conditions of Carriage.

For a reservation, information or other, contact our Call center : Algeria : + 213 (0) 21 96 63 63, France : + 33 (1) 76 54 40 00.

#### PAYMENT DETAILS

Fare Calculation: ALG AH ORN3160.00AH ALG3160.00AH  
BJA1460.00DZD7780.00END  
Form of payment: INV  
Endorsements: AH ONLY FEES SHOULD APPLY

#### FARE DETAILS

Fare:	DZD 7780
Taxes:	DZD 200Z DZD 1500DZ DZD 1765YB DZD 11065
Total Amount:	DZD 500
Fee	OBT01 TKT FEES
Total OB Fees:	DZD 500



**SONATRACH**  
 Activité Transport par Canalisation  
 Division Exploitation  
 Région Transport Centre (RTC) -Bejaia-

**HOTEL :**  
**BRAHIMI**  
**-BATNA-**

Bejaia le 13/11/2022

BON DE COMMANDE N° : 1190/2022

RENSEIGNEMENTS	PRESTATIONS
Objet : RESERVATION DE CHAMBRES	A-Hébergement
Nom et prénom	1-Chambre
01- MOULOUDJ NABIL	*Single
Séjour : DU 14/11/2022 AU 15/11/2022	B-Restauration (Tarif convention)
Facturation : Du :	1- Dîner
Imputation : SBM / RTC Bejaia.	2- Petit Déjeuner
Mode de règlement : Virement	
<b>ORDONNATEUR</b>	
P/Le Directeur Régional, Le Responsable Intérimaire	Signature
H. ALLOUT	
<p>Sans cette prise en charge dûment remplie, signée par une personne habilitée, aucune facture ne sera réglée.          Ne doivent être prises en considération que les mentions non rayées.          Toutes autres dépenses en dehors de celles mentionnées ci-dessus seront à la charge du missionnaire.          Le présent bon de commande ne concerne que les agents nominativement désignés.          Sonatrach Activité TRC, Région Transport Centre - Bejaia- ne réglera que les factures conformes à la présente prise en charge.</p>	
<p>SONATRACH TRC/EXL RTC Bejaia BP 19 aéroport Bejaia 06000 Tel : 034 10 96 31-32-33-34 Fax : 034 10 96 25-26 / 034 10 96 11</p>	



**BON DE COMMANDE** Passages  
Frêt

N° COMPTE CLIENT : 171128

Noms Prénoms Psg	N° Billet / LTA	Parcours PAX	MONTANT
HOULO KARIM		Algérie/Alger	
ZEMOURI LILA			

Cachet Agence Emettrice.



Code Agent

Date :

8/11/2023

Signature et Cachet Client



ACTIVITE TRANSPORT PAR CANALISATIONS  
DIVISION EXPLOITATION  
REGION TRANSPORT CENTRE (RTC Bejaia)

**BON DE COMMANDE**  
**(Transport aérien)**  
**N° 02/2023 Du 22/02/2023**

Nous vous demandons d'assurer l'embarquement du personnel ci dessous :

BENEFICIAIRE	PARCOURS	VOL AFFRETE
01- ATROUCHE ABDELMALEK 02- MADDI KEMAL 03- MEZIANI TOUFIK	Bejaia → H.Messaoud	SF 1129 Du 07/03/2023 (Vol SH/EP)

**IMPUTATION : 181 70 40 RTC BEJAIA**

 P/Le Directeur Régional,  
Le Responsable Intérimaire  
**H. ALLOUT**  


SONATRACH-TRC- Direction Régionale Béjaia BP, 19 Arrière port Bejaia Tél : 034.10.96.31/32/33/34 – Fax : 034.10.96.25-26 , 034.10.96.11

## **Résumé**

Dans le but d'améliorer en continu les solutions informatiques au sein des organisations, nous avons effectué un stage au sein de la section RELEX de SONATRACH. Notre mission était d'identifier les problèmes liés au processus de gestion au sein de la section et de proposer une solution concrète pour l'améliorer. Nous avons examiné la procédure de travail, les processus en place ainsi que les phases d'établissement des tâches. Au cours de notre analyse, nous avons relevé des problèmes tels qu'une surcharge documentaire, des difficultés de gestion, des limitations matérielles et logicielles, ainsi que des contraintes liées au processus de reporting. Pour résoudre ces problématiques, nous avons proposé une solution qui consiste à développer une application desktop monoposte de gestion intégrant des fonctionnalités essentielles visant à accélérer le flux de travail, réduire les risques d'erreur et améliorer la productivité globale au sein de RELEX. Pour ce faire, nous avons adopté une approche de conception basée sur UML pour définir clairement les besoins et les interactions du système, tout en mettant en œuvre la méthodologie SCRUM pour une gestion agile du projet. Pour le développement, nous avons utilisé Node.js comme environnement d'exécution et Electron.js pour créer notre application desktop en utilisant des technologies web. De plus, nous avons employé le langage de balisage HTML, la feuille de style CSS, ainsi que le langage de programmation JavaScript.

**Mots clés : RELEX, SONATRACH, Node.js, Electron.js, HTML, CSS, JavaScript.**

## **Abstract**

In order to continuously improving IT solutions within organisations, we carried out an internship within SONATRACH's RELEX section. Our mission was to identify the problems linked to the management process within the section and to propose a concrete solution to improve it. We took a look at the working procedure, the processes in place and the task-setting phases. During our analysis, we identified problems such as document overload, management difficulties, hardware and software limitations, and constraints in the reporting process. To resolve these issues, we proposed a solution involving the development of a single-user desktop management application incorporating essential functionalities aimed at speeding up workflow, reducing the risk of errors and improve overall productivity within RELEX. To achieve this, we adopted a UML-based design approach to clearly define system requirements and interactions, while implementing the SCRUM methodology for agile project management. For development, we used Node.js as the runtime environment and Electron.js to develop our desktop application using web technologies. In addition, we used HTML markup language, CSS style sheet and JavaScript programming language.

**Key words : RELEX, SONATRACH, Node.js, Electron.js, HTML, CSS, JavaScript.**