

ECOLE SUPERIEURE PRIVEE DES TECHNOLOGIES DE L'INFORMATION ET DE  
MANAGEMENT DE NABEUL



## **Rapport**

**De mini Projet C #**

**Elaboré par**

**Yasmine Belhadj  
Imene Raboudi  
Ilef Khiari**

***Titre du rapport***

***« Conception et développement d'une application desktop de gestion  
d'hôtel »***

**Encadré par**

**Encadrante universitaire**

**Mme Taycir Bouasker**

# Remerciements

Au terme de ce projet nous tenons à remercier infiniment notre enseignante universitaire, **Mme Taycir Bouaskar** qui a gardé un œil attentif sur le déroulement du projet en donnant des remarques constructives.

Nous le remercions pour sa disponibilité et son aide précieuse et cela a été un plaisir de travailler sous sa directive.

Nous souhaitons profiter de cette occasion pour exprimer notre profonde gratitude envers l'ensemble des enseignants et du personnel de l'École Supérieure Privée des Technologies de l'information et de Management de Nabeul, notamment ceux de la section cycle ingénieur. Leur aide précieuse et leurs conseils avisés ont été d'une importance capitale tout au long du processus.

# Table des matières

<b>Introduction générale</b>	<b>1</b>
<b>Chpitre1 : Présentation du projet</b>	<b>1</b>
0.1 Contexte du projet . . . . .	2
0.2 Présentation de l'université . . . . .	2
0.3 Etude et critique de l'existant . . . . .	3
0.4 Solution proposée . . . . .	3
0.5 Etude conceptuelle . . . . .	3
0.5.1 Processus de travail : . . . . .	3
0.5.2 Langage de modélisation (UML) . . . . .	4
<b>Chpitre2 : Analyse et spécification des besoins</b>	<b>4</b>
0.6 Cadrage des besoins . . . . .	5
0.6.1 Identification des acteurs . . . . .	5
0.6.2 Identification des besoins . . . . .	5
0.6.2.1 Identification des besoins fonctionnels : . . . . .	5
0.6.2.2 Identification des besoins non fonctionnels : . . . . .	6
0.6.3 Diagramme de cas d'utilisation général . . . . .	6
0.6.4 Diagramme de classe . . . . .	7
0.7 Analyse des besoins . . . . .	7
0.7.1 Raffinement du Cas d'utilisation « Authentification » : . . . . .	8
0.7.2 Raffinement du Cas d'utilisation « Gestion des données des clients » : . . . . .	9
0.7.2.1 Raffinement du Cas d'utilisation « Ajouter Client » : . . . . .	9
0.7.2.2 Raffinement du Cas d'utilisation « Modifier Client » : . . . . .	10
0.7.2.3 Raffinement du Cas d'utilisation « Consulter Client » : . . . . .	11
0.7.2.4 Raffinement du Cas d'utilisation « Supprimer Client » : . . . . .	12
0.7.3 Raffinement du Cas d'utilisation « Saisir formulaire de satisfaction » : . . . . .	13
0.7.4 Raffinement du Cas d'utilisation « vérifier disponibilité » : . . . . .	14
0.8 Conclusion . . . . .	15
<b>Chpitre3 : Réalisation</b>	<b>15</b>
0.9 Environnement matériel . . . . .	16
0.10 Environnement logiciel . . . . .	17
0.10.1 Outils utilisées . . . . .	17
0.10.1.1 MySQL Server . . . . .	17
0.10.1.2 Microsoft Visual Studio . . . . .	17
0.10.1.3 Draw.io . . . . .	17
0.10.2 Langage de programmation . . . . .	18
0.11 Interfaces réalisées . . . . .	18

0.11.1	Interface de connexion . . . . .	18
0.11.2	Changement de mot de passe . . . . .	19
0.11.3	Interface d'accueil . . . . .	19
0.11.4	Interface Gestion des Clients . . . . .	20
0.11.5	Interface Gestion des Réservations . . . . .	20
0.11.6	Interface Gestion des Chambres . . . . .	21
0.11.7	Interface Avis Client . . . . .	21
0.12	Conclusion . . . . .	22
<b>Conclusion générale</b>		<b>22</b>

# Table des figures

1	Logo de l'ITBS . . . . .	2
2	Modèle de Cycle de Vie en Cascade . . . . .	4
3	UML . . . . .	4
4	Diagramme de cas d'utilisation général . . . . .	6
5	Diagramme de classe . . . . .	7
6	Diagramme de séquence d'authentification . . . . .	8
7	Description de cas d'utilisation "Gestion des données des clients" . . . . .	9
8	Diagramme de séquence ajouter client . . . . .	9
9	Diagramme de séquence modifier client . . . . .	10
10	Diagramme de séquence consulter client . . . . .	11
11	Diagramme de séquence supprimer client . . . . .	12
12	Diagramme de saisir formulaire de satisfaction . . . . .	13
13	Diagramme de vérifier disponibilité . . . . .	14
14	logo de MySQL . . . . .	17
15	logo de Microsoft Visual Studio . . . . .	17
16	logo de draw.io . . . . .	17
17	logo de C SHARP . . . . .	18
18	Interface Login Administrateur . . . . .	18
19	Interface Changer mot de passe . . . . .	19
20	Interface Accueil . . . . .	19
21	Interface Gestion des Clients . . . . .	20
22	Interface Gestion des Réservations . . . . .	20
23	Interface Gestion des Chambres . . . . .	21
24	Interface d'avis client . . . . .	21

# Liste des tableaux

1	Description de cas d'utilisation « Authentification » . . . . .	8
2	Description de cas d'utilisation « Ajouter client » . . . . .	10
3	Description de cas d'utilisation « Modifier client » . . . . .	11
4	Description de cas d'utilisation « Consulter client » . . . . .	12
5	Description de cas d'utilisation « Supprimer client » . . . . .	13
6	Description de cas d'utilisation « saisir formulaire de satisfaction » . . . . .	14
7	Description de cas d'utilisation « vérifier disponibilité » . . . . .	15
8	Environnement matériel . . . . .	16

# Introduction générale

L'industrie hôtelière est confrontée à des défis sans cesse croissants, avec des attentes client en constante évolution et des impératifs opérationnels de plus en plus complexes. La mise en œuvre de systèmes de gestion d'hôtel spécialisés devient essentielle pour relever ces défis et maintenir un niveau élevé de satisfaction client. Ce rapport se penche sur l'application de telles solutions de gestion dans un contexte hôtelier, en mettant l'accent sur leurs avantages, leurs défis potentiels et leur impact global sur les opérations.

L'environnement hôtelier contemporain est caractérisé par une concurrence intense et des attentes élevées des clients. Les établissements hôteliers cherchent activement des moyens d'améliorer leur efficacité opérationnelle tout en offrant des expériences exceptionnelles à leurs clients. C'est dans ce contexte que les systèmes de gestion d'hôtel prennent une importance cruciale.

Dans ce cadre, notre projet **''Conception et développement d'un système de gestion hôtelière''** exploite ces avancées pour répondre aux exigences numériques actuelles. Ce rapport expose les différentes phases de développement de notre projet, qui sont organisées en trois chapitres distincts. Le premier chapitre, intitulé « Présentation du projet », présente le contexte du projet. Dans le deuxième chapitre, intitulé « Analyse et spécification des besoins », nous détaillons l'analyse des besoins fonctionnels et non fonctionnels, ainsi que la partie conception. Le troisième chapitre « Réalisation », qui présente les différentes interfaces développées de notre application. Enfin, nous concluons ce rapport en proposant des perspectives futures pour le projet.

# Chapitre 1

## Présentation du projet

### Introduction

Dans ce premier chapitre, on va présenter notre projet d'une manière générale pour obtenir une compréhension approfondie. Cela nous permettra de planifier le parcours du projet et d'assurer une organisation efficace de son déroulement.

Ce chapitre va traiter les sections relatives au contexte du projet, l'étude et les critiques de l'existant, la solution proposée et l'étude conceptuelle.

### 0.1 Contexte du projet

Le présent projet s'intitule "Conception et développement d'une application de gestion d'hôtel". Il a été proposé comme mini-projet pour les étudiants de 1ère année du cycle ingénieur en matière de C sharp à l'École Supérieure Privée des Technologies de l'Information et de Management de Nabeul : IT Business School.

### 0.2 Présentation de l'université

L'It Business School (Figure 1.1), avec sa large gamme de programmes d'études et ses diverses activités de recherche appliquées, forme aussi bien les personnes en exercice, que la réserve de relevé. Elle a de ce fait l'avantage de rendre accessible les connaissances de priorité à toute échelle sociale et culturelle, et de servir la collectivité à la recherche de ses besoins.



FIGURE 1 – Logo de l'ITBS



## 0.3 Etude et critique de l'existant

De nos jours, les systèmes de gestion d'hôtel ont émergé comme des outils indispensables pour les propriétaires et les gestionnaires d'hôtels. Ces systèmes ont pour rôle la réduction de nombreuses tâches administratives et opérationnelles tels que le suivi des données des clients la gestion manuelle des réservations ... Alors quelle est la bonne solution pour aider les agents administratifs et faciliter les tâches .

## 0.4 Solution proposée

Notre projet de gestion automatisée des réservations de chambres représente une réponse novatrice aux défis opérationnels actuels. En établissant une base de données centralisée, notre solution vise à éliminer la structuration déficiente des données, simplifiant ainsi la gestion tout en automatisant les tâches répétitives. L'intégration de mises à jour dynamiques des disponibilités et une fonctionnalité de réservation rapide contribuent à offrir une expérience client optimale .

## 0.5 Etude conceptuelle

Dans ce chapitre nous commençons par une présentation des différents outils logiciels et les langages de modélisations utilisés. Ensuite nous détaillons les diagrammes des cas d'utilisation.

### 0.5.1 Processus de travail :

#### Modèle de Cycle de Vie en Cascade

Le modèle de cycle de vie en CASCADE a été élaboré en 1970 , permet de diviser le projet en sept phases distinctes sur le principe du non-retour.  
Ce modèle est basé sur l'hypothèse : dès le début, nous pouvons définir ce que nous pouvons réaliser pleinement (expressions des besoins).

Dans ce modèle le principe est très simple :

- L'étape peut commencer si non étape précédente est complètement terminée
- Chaque étape se termine à une date précise avec la production de documents ou programmes spécifiques.
- Les résultats sont déterminés sur la base des interactions entre les étapes, ils font l'objet d'un examen approfondi et l'étape suivante n'est franchie que s'ils sont satisfaisants.

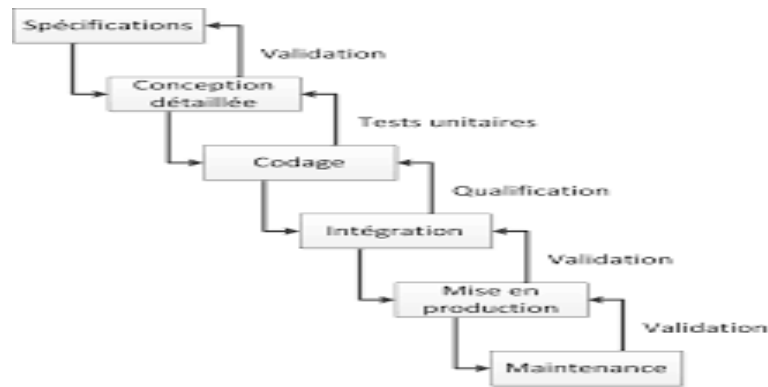


FIGURE 2 – Modèle de Cycle de Vie en Cascade

### 0.5.2 Langage de modélisation (UML)

Pour la conception de notre système nous avons adopté une méthode orientée objet. En effet cette dernière est une approche incontournable dans le cadre du développement des applications. Pour mieux présenter l'architecture de notre système, on va choisir le langage de modélisation le plus adopté UML :

L'UML est un langage de modélisation graphique basé sur des pictogrammes, qui permet de modéliser de manière très détaillée et indépendante des langages et des environnements, ce qui facilite la collaboration entre les différents acteurs impliqués dans un projet. Il permet également de faire des simulations avant de construire un système, ce qui peut aider à réduire les erreurs et à améliorer l'efficacité.

En utilisant l'UML, il est possible d'exprimer tous les aspects d'un système dans un seul modèle, y compris les aspects statiques, dynamiques, spécifiques, etc. Cela facilite la documentation et la compréhension du projet pour toutes les parties prenantes.



FIGURE 3 – UML

## Conclusion

A travers ce chapitre , nous avons présenté le cadre générale de notre projet . Puis, nous avons décrit la méthodologie de gestion de travail . Dans le prochain chapitre, nous aborderons l'analyse et la spécification des besoins.

# Chapitre 2

## Analyse et spécification des besoins

### Introduction

Ce chapitre présente une initiative d'effectuer ce projet. Nous commençons à identifier les acteurs, les exigences fonctionnelles et non-fonctionnelles du système. Nous terminons ce chapitre avec la présentation des différents diagrammes .

### 0.6 Cadrage des besoins

Dans cette section, nous identifions les acteurs de notre application et nous présentons en détails les exigences qui seront mises à la disposition de chaque acteur. Ces besoins seront classes en besoins fonctionnels et non fonctionnels.

#### 0.6.1 Identification des acteurs

**L'agent administrateur :** C'est le responsable du système d'administration ayant plus de privilèges d'accès aux données. Il est charge de la gestion des différents éléments qui seront affichés sur l'application. Il peut gérer, consulter, ajouter, modifier et supprimer les éléments importants qui sont affichés sur l'application.

#### 0.6.2 Identification des besoins

Dans cette section, nous procédons à la définition de toutes les exigences fonctionnelles et non fonctionnelles auxquelles notre site doit répondre.

##### 0.6.2.1 Identification des besoins fonctionnels :

— **Espace agent administratif :**

- Authentification
- Gestion des données des clients :l'agent administratif a la possibilité d'ajouter ,de modifier , de consulter et de supprimer un client.
- Gestion des réservations : l'agent administratif a la possibilité d'ajouter ,de modifier , de consulter et de supprimer une réservation.
- Gestion d'états de chambres

### 0.6.2.2 Identification des besoins non fonctionnels :

Il s'agit des besoins qui caractérisent le système. Ce sont des besoins en matière de performance

- **Temps de réponse** : Les utilisateurs s'attendent à ce que l'application réponde rapidement à leurs actions, que ce soit pour effectuer une réservation, rechercher des informations sur les chambres.
- **Ergonomie** : Il s'agit de concevoir une interface qui facilite la réservation des chambres, la gestion des clients et des transactions, en veillant à une navigation fluide pour le personnel de l'hôtel.
- **Convivialité** : L'interface utilisateur doit être conviviale, intuitive et accessible à un large éventail d'utilisateurs. S'assurer que le personnel peut être formé facilement à l'utilisation de l'application.
- **Évolutivité** : L'application doit être capable de gérer un nombre croissant d'utilisateurs et de données à mesure que l'entreprise se développe

### 0.6.3 Diagramme de cas d'utilisation général

Pour traduire les différents besoins et acteurs précédemment mentionnés, nous modélisons les différentes fonctions dans un diagramme de cas d'utilisation, voir la figure ci-dessous qui représente le diagramme de cas d'utilisation général de notre application.



FIGURE 4 – Diagramme de cas d'utilisation général

### 0.6.4 Diagramme de classe

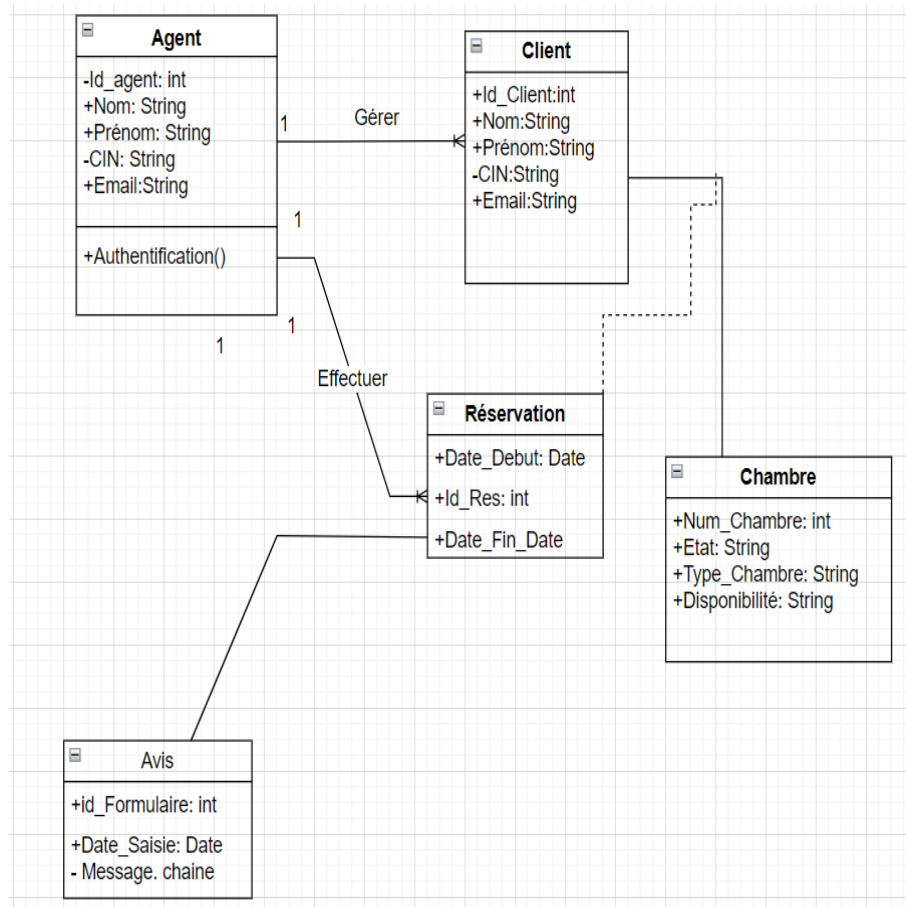


FIGURE 5 – Diagramme de classe

## 0.7 Analyse des besoins

Dans cette section, nous allons aborder la tâche la plus importante dans l'élaboration de ce travail, à savoir la tâche de conception. Nous détaillons chacun de ces modules conformément à la notation UML par la description de différents diagrammes.

### 0.7.1 Raffinement du Cas d'utilisation « Authentification » :

le diagramme de séquence

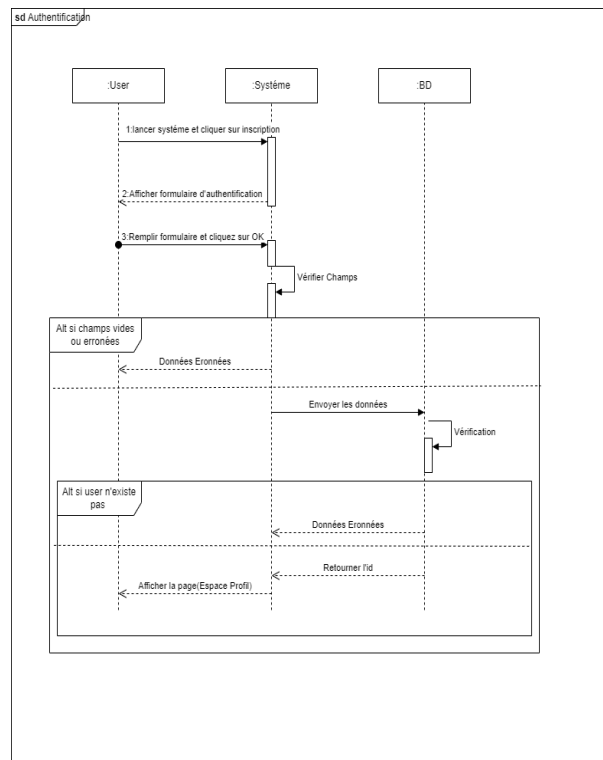


FIGURE 6 – Diagramme de séquence d'authentification

### Description de cas d'utilisation « Authentification »

<b>Cas d'utilisation</b>	-Authentification
<b>Acteur</b>	-Administrateur
<b>Précondition</b>	-L'administrateur accédé à l'application
<b>Post condition</b>	-L'administrateur est connecté à son compte
<b>Description de scénario</b>	-Le système affiche le formulaire d'authentification -L'admin saisie son login et son mot de passe -Le système donne l'accès à l'interface correspondante.
<b>Exception</b>	Le système affiche message d'erreur « login ou mot de passe incorrecte » : si les champs sont vides, les informations fournis sont incorrectes ou compte inexistant.

TABLE 1 – Description de cas d'utilisation « Authentification »

## 0.7.2 Raffinement du Cas d'utilisation « Gestion des données des clients » :

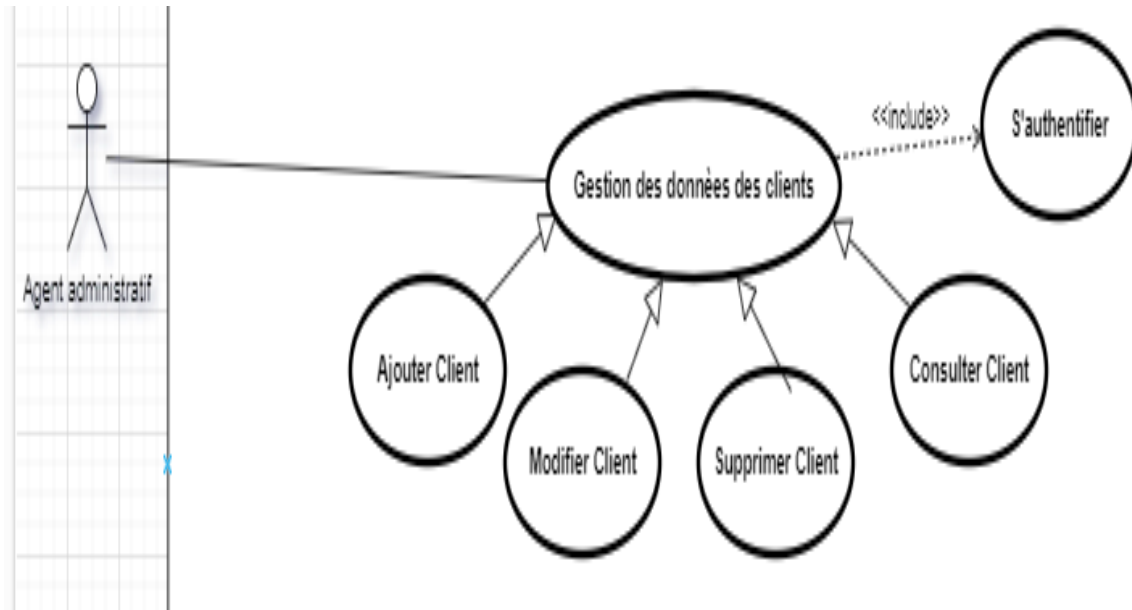


FIGURE 7 – Description de cas d'utilisation "Gestion des données des clients"

### 0.7.2.1 Raffinement du Cas d'utilisation « Ajouter Client » :

#### Le diagramme de séquence

Le diagramme de séquence permet d'obtenir une vision sur le déroulement de la tâche « Ajouter client ».

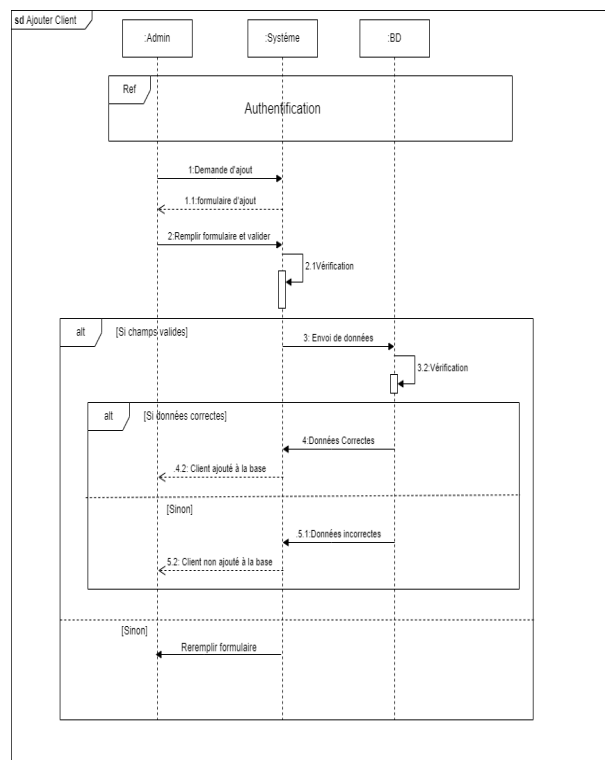


FIGURE 8 – Diagramme de séquence ajouter client

### Description de cas d'utilisation « Ajouter client »

<b>Cas d'utilisation</b>	Ajouter Client
<b>Acteur</b>	Agent Administrateur
<b>Pré-condition</b>	L'agent accède a l'application
<b>Post-condition</b>	L'admin est connecté
<b>Description de scénario</b>	Le système affiche formulaire d'ajout L'admin remplir le formulaire Le système donne l'accès à l'interface correspondante.
<b>Exception</b>	Le système affiche message d'erreur si les champs sont non valides

TABLE 2 – Description de cas d'utilisation « Ajouter client »

### 0.7.2.2 Raffinement du Cas d'utilisation « Modifier Client » :

le diagramme de séquence

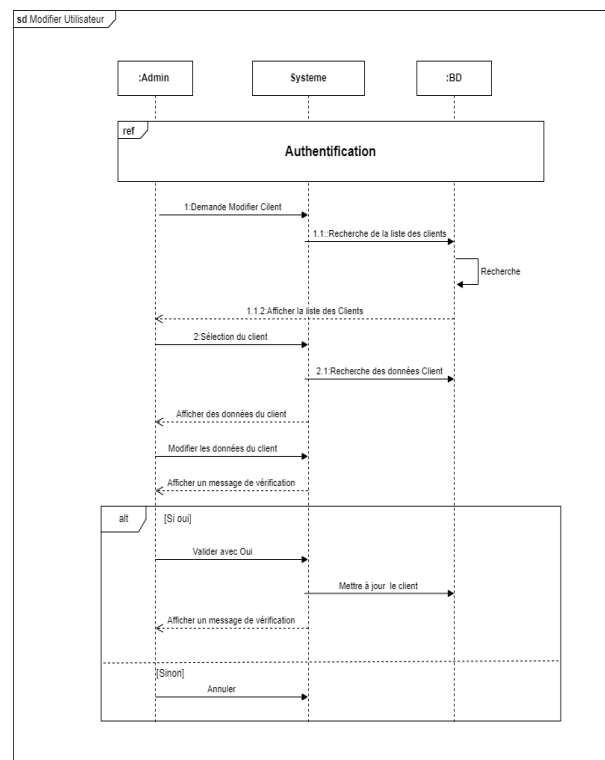


FIGURE 9 – Diagramme de séquence modifier client



### Description de cas d'utilisation « Modifier client »

<b>Cas d'utilisation</b>	Modifier Client
<b>Acteur</b>	Agent Administrateur
<b>Pré-condition</b>	L'agent accède à l'application
<b>Post-condition</b>	L'admin est connecté
<b>Description de scénario</b>	Le système affiche formulaire de modifier L'admin remplir le formulaire Le système donne l'accès à l'interface correspondante.
<b>Exception</b>	Le système affiche message d'erreur si les champs sont non valides

TABLE 3 – Description de cas d'utilisation « Modifier client »

### 0.7.2.3 Raffinement du Cas d'utilisation « Consulter Client » :

le diagramme de séquence

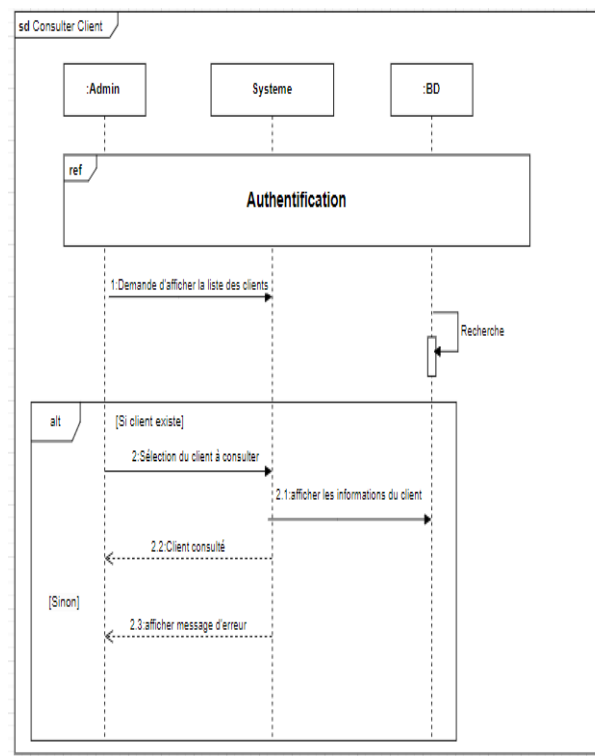


FIGURE 10 – Diagramme de séquence consulter client

### Description de cas d'utilisation « Consulter client »

<b>Cas d'utilisation</b>	Consulter Client
<b>Acteur</b>	Agent Administrateur
<b>Pré-condition</b>	L'agent accède a l'application
<b>Post-condition</b>	L'admin est connecté
<b>Description de scénario</b>	Le système affiche un formulaire de consultation de clients. L'administrateur sélectionne le client qu'il souhaite consulter. Le système affiche les informations détaillées du client .
<b>Exception</b>	Le système affiche message d'erreur si le client n'existe pas

TABLE 4 – Description de cas d'utilisation « Consulter client »

#### 0.7.2.4 Raffinement du Cas d'utilisation « Supprimer Client » :

le diagramme de séquence

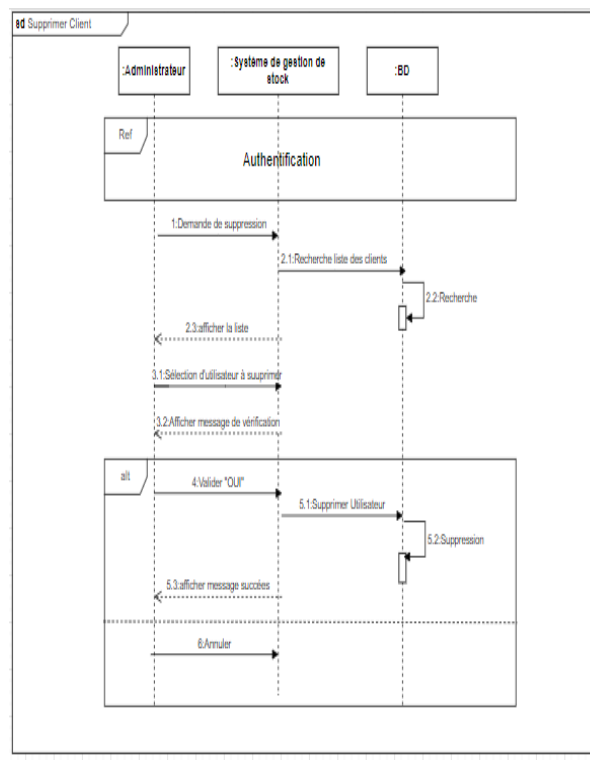


FIGURE 11 – Diagramme de séquence supprimer client

### Description de cas d'utilisation « Supprimer client »

<b>Cas d'utilisation</b>	Supprimer Client
<b>Acteur</b>	Agent Administrateur
<b>Pré-condition</b>	L'agent accède a l'application
<b>Post-condition</b>	L'admin est connecté
<b>Description de scénario</b>	<ul style="list-style-type: none"> <li>-Le système affiche la liste des clients</li> <li>- L'administrateur sélectionne le client qu'il souhaite supprimer.</li> <li>-L'application affiche une boîte de dialogue ou une confirmation de la suppression de l'offre</li> <li>- L'administrateur confirme la suppression de l'offre.</li> </ul>
<b>Exception</b>	Annulation si non confirmé

TABLE 5 – Description de cas d'utilisation « Supprimer client »

### 0.7.3 Raffinement du Cas d'utilisation « Saisir formulaire de satisfaction » : le diagramme de séquence

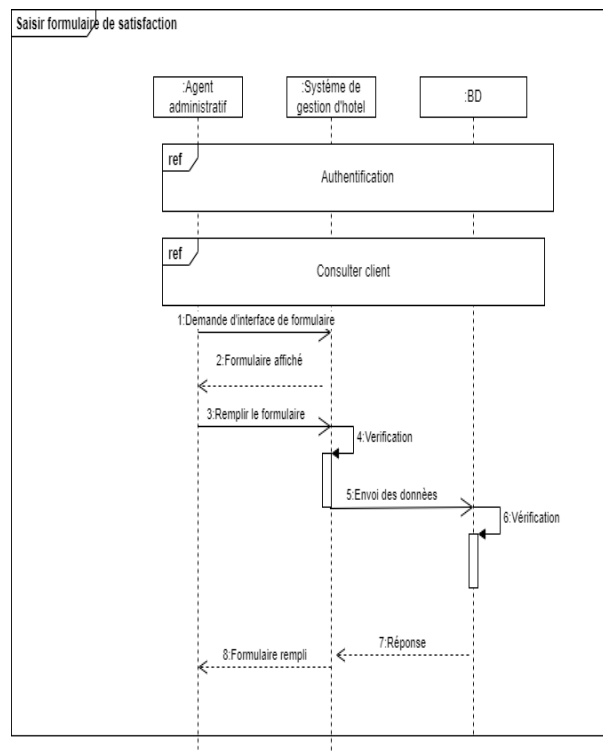


FIGURE 12 – Diagramme de saisir formulaire de satisfaction

### Description de cas d'utilisation « saisir formulaire de satisfaction »

<b>Cas d'utilisation</b>	Saisir formulaire de satisfaction
<b>Acteur</b>	Agent Administrateur
<b>Pré-condition</b>	Authentification préalable
<b>Post-condition</b>	L'agent administratif est connecté
<b>Description de scénario</b>	<ul style="list-style-type: none"> <li>1. L'agent se connecte au système de gestion à l'aide de ses identifiants.</li> <li>2. Le système affiche une option pour "Exprimer la satisfaction clientèle".</li> <li>3. L'agent sélectionne cette option.</li> <li>4. Le système affiche un formulaire de feedback ou d'évaluation avec des questions sur différents aspects du séjour, tels que la propreté, le service, la qualité de la chambre, la nourriture, etc.</li> <li>5. L'agent questionne le client afin de s'avoir son taux de satisfaction</li> <li>6. formulaire envoyé avec succès</li> </ul>
<b>Exception</b>	Client non satisfait

TABLE 6 – Description de cas d'utilisation «saisir formulaire de satisfaction »

### 0.7.4 Raffinement du Cas d'utilisation « vérifier disponibilité » : le diagramme de séquence

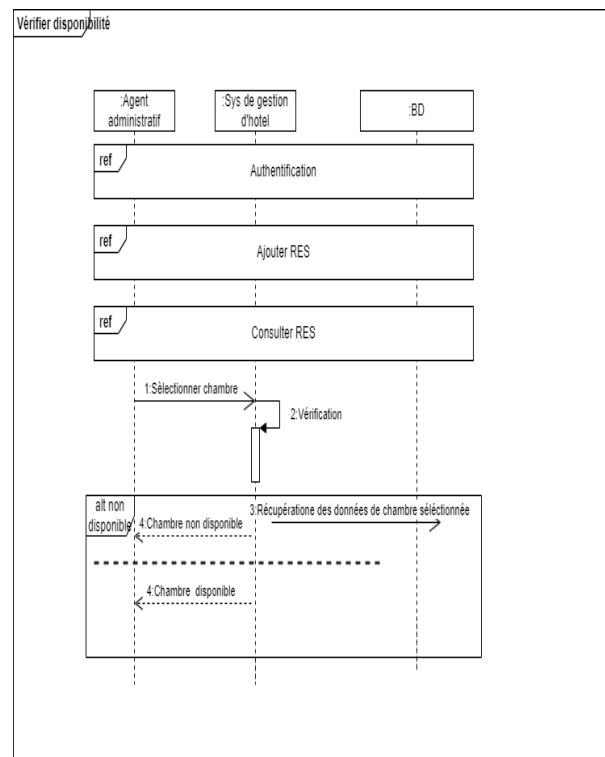


FIGURE 13 – Diagramme de vérifier disponibilité

### Description de cas d'utilisation « vérifier disponibilité »

<b>Cas d'utilisation</b>	Vérifier disponibilité
<b>Acteur</b>	Agent Administratif
<b>Pré-condition</b>	-Authentification préalable -Une réservation déjà ajoutée
<b>Post-condition</b>	Chambre vérifiée (disponible ou non)
<b>Description de scénario</b>	<ul style="list-style-type: none"> <li>• 1. L'agent se connecte au système de gestion à l'aide de ses identifiants.</li> <li>• 2. Le système affiche une option pour "Vérifier la disponibilité des chambres".</li> <li>• 3. L'agent sélectionne cette option.</li> <li>• 4. Le système effectue une vérification en temps réel de la disponibilité des chambres pour les dates spécifiées.</li> <li>• 5. Le système affiche les chambres disponibles</li> <li>• 6. L'agent peut informer le client de la disponibilité des chambres et procéder à une réservation si nécessaire.</li> </ul>
<b>Exception</b>	Si la réservation spécifiée ne peut pas être disponible, le système affiche un message : cette chambre n'est pas disponible

TABLE 7 – Description de cas d'utilisation «vérifier disponibilité »

## 0.8 Conclusion

En conclusion de ce chapitre, nous avons établi une vision claire des besoins de notre projet en réalisant un cadrage précis, identifiant les acteurs et détaillant les besoins fonctionnels et non fonctionnels. L'analyse approfondie a été menée à travers le raffinement des cas d'utilisation, soutenu par des diagrammes de séquence illustrant les interactions clés du système. Ces éléments fournissent une base solide pour une conception précise, préparant ainsi le terrain pour la phase suivante du développement axée sur la réalisation des fonctionnalités identifiées

# Chapitre 3

## Réalisation

### Introduction

Après avoir achevé l'étape de la conception, nous entamons dans ce chapitre la partie réalisation et implémentation dans laquelle nous nous assurons que le système est prêt à être exploité par les utilisateurs finaux. À la fin de ce chapitre, les objectifs doivent avoir atteints et le projet doit être fini.

### 0.9 Environnement matériel

C'est l'ensemble de matériels que nous avons utilisé pour arriver à développer notre application et qui sont généralement l'ensemble des trois ordinateurs portables.



Matériel	Image	Critères
HP		<ul style="list-style-type: none"><li>– Processeur : Intel Core i7 - 4 Cores 8 Threads 1.9 - 4.2 GHz</li><li>– Mémoire Vive : 8 Go - DDR4</li><li>– Disque stockage : HDD</li><li>– Carte graphique : Intégrée - Intel UHD Graphics 620</li><li>– Système d'exploitation installé : Microsoft Windows 11 Professionnel 64 bits</li></ul>
ASUS		<ul style="list-style-type: none"><li>– Processeur : AMD Ryzen 5 3500U</li><li>– Mémoire Vive : 8 Go</li><li>– Disque stockage : SSD 512 Go – Carte graphique : AMD Radeon Graphics</li><li>– Système d'exploitation installé : Microsoft Windows 10 Professionnel 64 bits</li></ul>

TABLE 8 – Environnement matériel

## 0.10 Environnement logiciel

### 0.10.1 Outils utilisés

Nous avons utilisé les outils suivants : MySQL Server , Microsoft Visual Studio , Draw.io

#### 0.10.1.1 MySQL Server

Pour gérer efficacement et en toute sécurité les données importantes de notre application hôtelière, nous utilisons **MySQL** comme système de gestion de base de données.



FIGURE 14 – logo de MySQL

#### 0.10.1.2 Microsoft Visual Studio

Microsoft Visual Studio est un environnement de développement intégré (EDI) utilisé pour créer des applications logicielles destinées aux plates-formes Windows, web, mobiles et cloud. Il offre une large gamme d'outils et de services aux développeurs de logiciels pour écrire, déboguer, tester et déployer des applications de manière efficace. Visual Studio prend en charge divers langages de programmation tels que C++, C, F, Visual Basic .NET, et bien d'autres. .

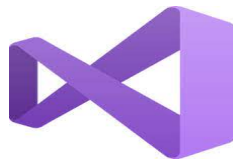


FIGURE 15 – logo de Microsoft Visual Studio

#### 0.10.1.3 Draw.io

Pour la modélisation efficace de notre projet, nous avons opté pour l'utilisation de Draw.io, un outil de conception graphique intuitif. Cette plateforme nous a permis de créer des diagrammes détaillés, favorisant une communication claire et une compréhension approfondie des différentes composantes du projet. L'intégration de Draw.io a grandement facilité la visualisation et la gestion de notre architecture, renforçant ainsi la cohérence et l'efficacité de notre démarche de développement



FIGURE 16 – logo de draw.io

### 0.10.2 Langage de programmation

Nous avons choisi **C sharp** comme langage de programmation principal pour créer notre application, car il offre une conception solide et une grande flexibilité grâce à son approche orientée objet.



FIGURE 17 – logo de C SHARP

En résumé, notre projet d'application de gestion d'hôtel repose sur une combinaison robuste de C, SQL Server et Visual Studio, assurant ainsi des performances fiables et une évolutivité à long terme.

## 0.11 Interfaces réalisées

Dans cette section, nous essayerons d'exposer des captures d'écran prises à partir des interfaces de notre solution et / ou de notre code source

### 0.11.1 Interface de connexion

La figure suivante présente l'interface de connexion d'un agent déjà inscrit : l'agent doit saisir son prénom ainsi que son mot de passe afin de se connecter.



FIGURE 18 – Interface Login Administrateur



### 0.11.2 Changement de mot de passe

Lorsqu'un agent souhaite changer le mot de passe, il doit cliquer sur "Changer mot de passe", ce qui ouvrira une interface permettant d'apporter les modifications nécessaires.

FIGURE 19 – Interface Changer mot de passe

### 0.11.3 Interface d'accueil

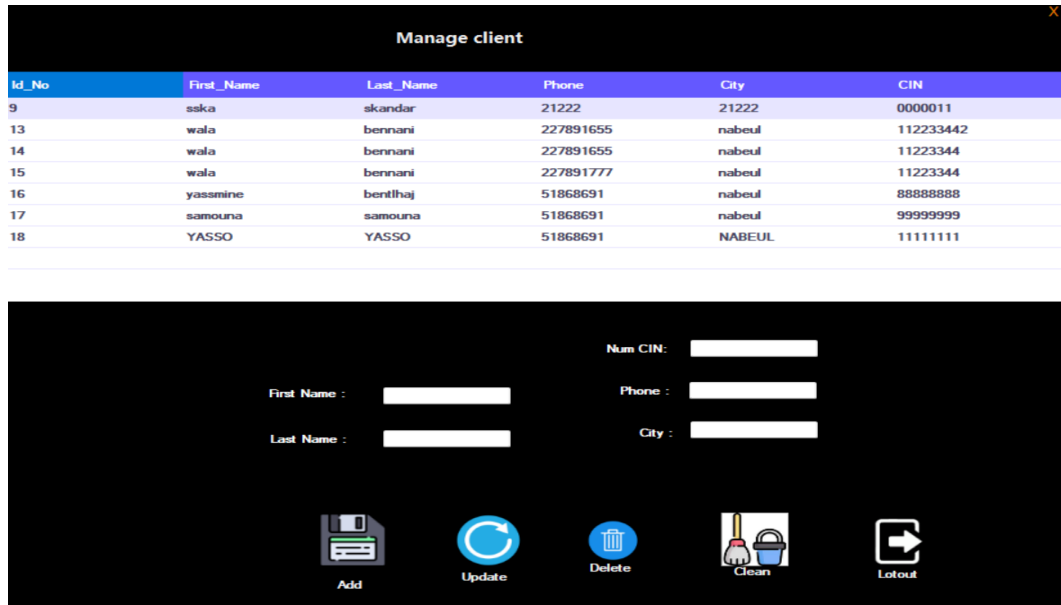
Interface d'accueil offre une navigation facile entre les différentes fonctionnalités de l'application, garantissant une expérience utilisateur cohérente sur l'ensemble des fonctionnalités réservées à l'agent.



FIGURE 20 – Interface Accueil

### 0.11.4 Interface Gestion des Clients

Dans cette section, nous dévoilerons l'interface de gestion des clients, offrant des fonctionnalités spécifiques pour la gestion des utilisateurs, permet à l'agent administrative d'ajouter, de modifier, de supprimer et de consulter des informations relatives aux clients.



**Manage client**

Id_No	First_Name	Last_Name	Phone	City	CIN
9	saka	skandar	21222	21222	0000011
13	wala	bennani	227891655	nabeul	112233442
14	wala	bennani	227891655	nabeul	11223344
15	wala	bennani	227891777	nabeul	11223344
16	yasmine	bentlhaj	51868691	nabeul	88888888
17	samouna	samouna	51868691	nabeul	99999999
18	YASSO	YASSO	51868691	NABEUL	11111111

Num CIN:

First Name :

Last Name :

Phone :

City :

FIGURE 21 – Interface Gestion des Clients

### 0.11.5 Interface Gestion des Réservations

Cette interface offre aux utilisateurs la possibilité de parcourir la liste des chambres disponibles avant de procéder à une réservation. En fournissant une vue détaillée qui concerne le processus de prise en charge des demandes de réservation des clients. Cela inclut la gestion des dates de séjour, la disponibilité des chambres, la tarification, et la coordination des réservations.



**Manage Reservation**

ReservationId	RoomNo	DateIn	DateOut	ClientId
1	12	06/12/2023	09/12/2023	17
2	13	15/12/2023	29/12/2023	17
3	16	05/12/2023	05/12/2023	18

Cin Client :

FIGURE 22 – Interface Gestion des Réservations

### 0.11.6 Interface Gestion des Chambres

Interface Gestion des Chambres concentre sur la disposition, la maintenance et la disponibilité des chambres au sein de l'établissement. Cela englobe la maintenance des chambres, le nettoyage, l'attribution des chambres aux clients, et la gestion des différents types de chambres

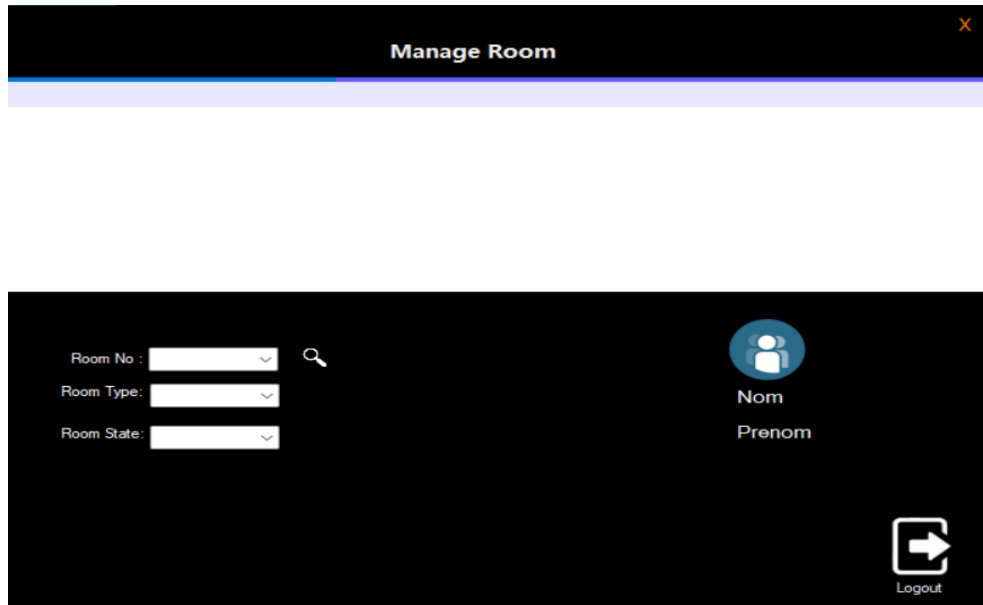


FIGURE 23 – Interface Gestion des Chambres

### 0.11.7 Interface Avis Client

L'interface Avis Client permet aux clients de partager leurs opinions, évaluations sur leur expérience dans l'établissement hôtelier en terme de service, propreté, confort, etc...



FIGURE 24 – Interface d'avis client

## **0.12 Conclusion**

Dans ce dernier chapitre, nous avons décrit brièvement les principales interfaces existantes dans notre plate forme en spécifiant l'environnement matériel, les outils et les langages de développement. En effet, nous avons achevé l'implémentation tout en respectant la conception élaborée.

# Conclusion générale

Ce projet, centré sur la gestion d'hôtel, a suivi une progression méthodique à travers ses différentes étapes. Le premier chapitre a posé les bases en identifiant les enjeux clés et en établissant une méthodologie robuste. Le deuxième chapitre a examiné de près les besoins, mettant en avant les acteurs et les actions qui ont guidé le développement de l'application. La phase de mise en œuvre, présentée dans le dernier chapitre, a détaillé l'environnement logiciel et les interfaces de l'application, démontrant ainsi l'efficacité de l'informatique comme outil essentiel dans la gestion quotidienne des réservations d'hôtel.

En conclusion, ce projet répond de manière pertinente à un besoin réel de gestion, soulignant la capacité de l'informatique à fournir des solutions efficaces. Les perspectives futures se tournent vers une amélioration continue de l'application pour rester adaptable aux besoins émergents, mettant en avant l'importance de l'innovation dans la gestion des espaces et des ressources hôtelières.