



**ECOLE MAROCAINE DES
SCIENCES DE L'INGENIEUR**
Membre de
HONORIS UNITED UNIVERSITIES

Application de Réservation d'Hôtel

Ingénierie Informatique et Réseaux

Année Universitaire : 2024/2025

Réalisée par :

CHAIMAE ELKABIL

HOUDA GMAIR

AICHA TAHIRi

Encadré par :

Ali IDMANSOUR

Table des matières :

Application de Réservation d'Hôtel	1
Table des matières :	2
Liste des acronymes	4
Remerciements :	5
Résumé :	6
Introduction Générale	1
Chapitre 1 : Analyse des besoins	2
1.Introduction	3
2. Diagramme de Cas d'Utilisation	4
.2.1. Acteurs Principaux	4
2.2. Cas d'Utilisation	4
2.3. Représentation Visuelle	4
3. Diagramme de Classes	6
3.1. Classes Principales	6
3.2. Relations	6
3.3. Représentation Visuelle	6
4.Outil de conception :	7
5.Conclusion :	8
Chapitre 2 : Etude Fonctionnelle et Technique	9
1. Introduction	10
2. Technologies utilisées	11
3. Architecture et Déploiement	13
4. Conclusion	14
Chapitre 3 : Réalisation de L'Application	15
1. Introduction	16
1. Partie Desktop :	17
a. Authentification et Gestion de Compte :	17
b. Gestion de la disponibilité des chambres :	17
c. Notification de réservation	19
2.Partie Web :	19
a. Page d'Accueil :	19

b.	Interface de Création de Compte Utilisateur	20
c.	Interface pour se connecter	20
d.	. Détails de la Chambre :	21
e.	Interface de reservation	21
f.	Notification par email :	22
g.	Confirmation de la réservation :	22
h.	Interface ou réservation peuvent être annuler par l admin :	23
3.	Conclusion	23
	Bibliographie :	24
	Lien GitHub :	24

Liste des acronymes

MVC : Modèle-Vue-Contrôleur

HTML : HyperText Markup Language

SQL : Structured Query Language

CSS : Cascading Style Sheets

JS : JavaScript

Remerciements :

Nous adressons nos plus sincères remerciements à

Ali IDMANSOUR, pour son accompagnement précieux tout au long de ce projet. Votre expertise remarquable, combinée à votre sens de l'écoute et à votre disponibilité, a été une source inestimable de motivation et d'apprentissage pour notre équipe.

Vous avez su transformer chaque défi en une opportunité de croissance, nous guidant avec patience et perspicacité à travers les étapes les plus complexes du développement. Votre engagement, votre passion et votre quête constante d'excellence ont inspiré notre travail et renforcé notre détermination à atteindre nos objectifs.

Nous tenons également à exprimer notre reconnaissance pour votre soutien indéfectible, vos précieux conseils et votre capacité à éclairer notre chemin avec bienveillance et rigueur.

Ce projet reflète non seulement nos efforts, mais aussi la richesse de votre contribution. Nous vous remercions profondément d'avoir cru en notre potentiel et d'avoir partagé avec nous votre savoir et votre expérience.

Merci du fond du cœur pour votre impact durable sur notre parcours.

Résumé :

Le projet **Mini Projet DotNet : Application de Réservation d'Hôtel** a pour objectif de développer une solution complète pour la gestion des réservations d'hôtel, en offrant des outils adaptés à la fois aux administrateurs et aux clients. La partie **Back-office**, dédiée aux administrateurs, permet de gérer les employés, les clients, les chambres, les disponibilités, les réservations et les paiements. Elle intègre des fonctionnalités avancées telles que l'exportation/importation de données, la génération de PDF, les notifications par e-mail, et des tableaux de bord pour une gestion optimale. La partie **Front-office**, accessible aux clients, leur permet de créer un compte, de réserver des chambres avec confirmation par e-mail et de générer des bons de réservation en PDF. Ce projet vise à offrir une gestion fluide et performante tout en garantissant une expérience utilisateur simple et agréable pour les clients.

Introduction Générale

La gestion des réservations dans les hôtels est une activité essentielle mais souvent complexe en raison du volume d'informations à traiter, de la variété des besoins des clients et de la nécessité de maintenir une synchronisation efficace entre les différents services de l'hôtel. Dans un contexte où la digitalisation transforme de nombreux secteurs, il devient indispensable pour les hôtels de disposer d'outils performants pour optimiser la gestion des réservations, améliorer l'expérience client et renforcer leur compétitivité.

L'objectif principal de ce projet est de développer une application informatique permettant de gérer efficacement les réservations d'un hôtel. L'application sera divisée en deux modules principaux :

Back-office : Un espace réservé aux administrateurs et au personnel, destiné à gérer les opérations internes telles que la gestion des utilisateurs, des clients, des chambres, des paiements, et des réservations.

Front-office : Une interface accessible aux clients leur permettant de réserver des chambres en ligne, de recevoir des confirmations par email et d'accéder à leurs bons de réservation.

Les systèmes traditionnels de gestion des réservations reposent souvent sur des procédures manuelles ou des outils limités, ce qui peut entraîner des erreurs humaines, des retards dans le traitement des demandes, et une expérience client dégradée. Ces limitations se traduisent par :

- Une surcharge de travail pour le personnel administratif.
- Des difficultés à assurer une disponibilité des chambres en temps réel.
- Un manque d'outils pour suivre les performances et prendre des décisions basées sur des données fiables.
- Une communication insuffisante avec les clients, notamment en matière de confirmation et suivi des réservations.

Ce projet vise à résoudre ces problèmes en proposant une solution centralisée, intuitive et automatisée, permettant aux hôtels d'optimiser leurs opérations tout en améliorant la satisfaction des clients.

Chapitre 1 : Analyse des besoins

1.Introduction

Dans tout projet logiciel, l'analyse des besoins est une étape clé qui conditionne le succès de la solution développée. Elle vise à identifier, organiser et structurer les attentes des utilisateurs ainsi que les contraintes techniques pour garantir que le produit final répond parfaitement aux objectifs du projet.

Dans le cadre de notre application de réservation d'hôtel, cette analyse s'inscrit dans un contexte où la digitalisation des processus devient indispensable pour améliorer l'efficacité opérationnelle et offrir une expérience utilisateur optimale. Notre solution doit non seulement permettre une gestion efficace des activités internes de l'hôtel, mais également proposer une interface conviviale pour les clients souhaitant réserver des chambres.

Ce chapitre a pour objectif de définir les besoins fonctionnels et non fonctionnels de l'application, en identifiant les fonctionnalités essentielles pour les administrateurs et les clients, tout en prenant en compte des aspects comme la sécurité, la performance et la maintenabilité. Il vise également à illustrer les interactions principales entre les utilisateurs et le système à travers des cas d'utilisation et des diagrammes UML, tout en structurant les entités principales via un diagramme de classes. L'analyse des besoins joue un rôle clé en réduisant les risques, en établissant une feuille de route claire et en garantissant l'alignement sur les objectifs du projet, posant ainsi les bases nécessaires pour la conception, le développement et la validation de l'application de réservation d'hôtel.

2. Diagramme de Cas d'Utilisation

Le diagramme de cas d'utilisation représente les interactions entre les différents acteurs (utilisateurs) et les fonctionnalités offertes par l'application.

2.1. Acteurs Principaux

1. **Administrateur**

- Gère les utilisateurs et les rôles.
- Gère les chambres et leurs types.
- Suit les réservations et les paiements.
- Génère des rapports et des statistiques.

2. **Client**

- Crée un compte et gère son profil.
- Recherche et réserve des chambres.
- Reçoit des notifications de confirmation.

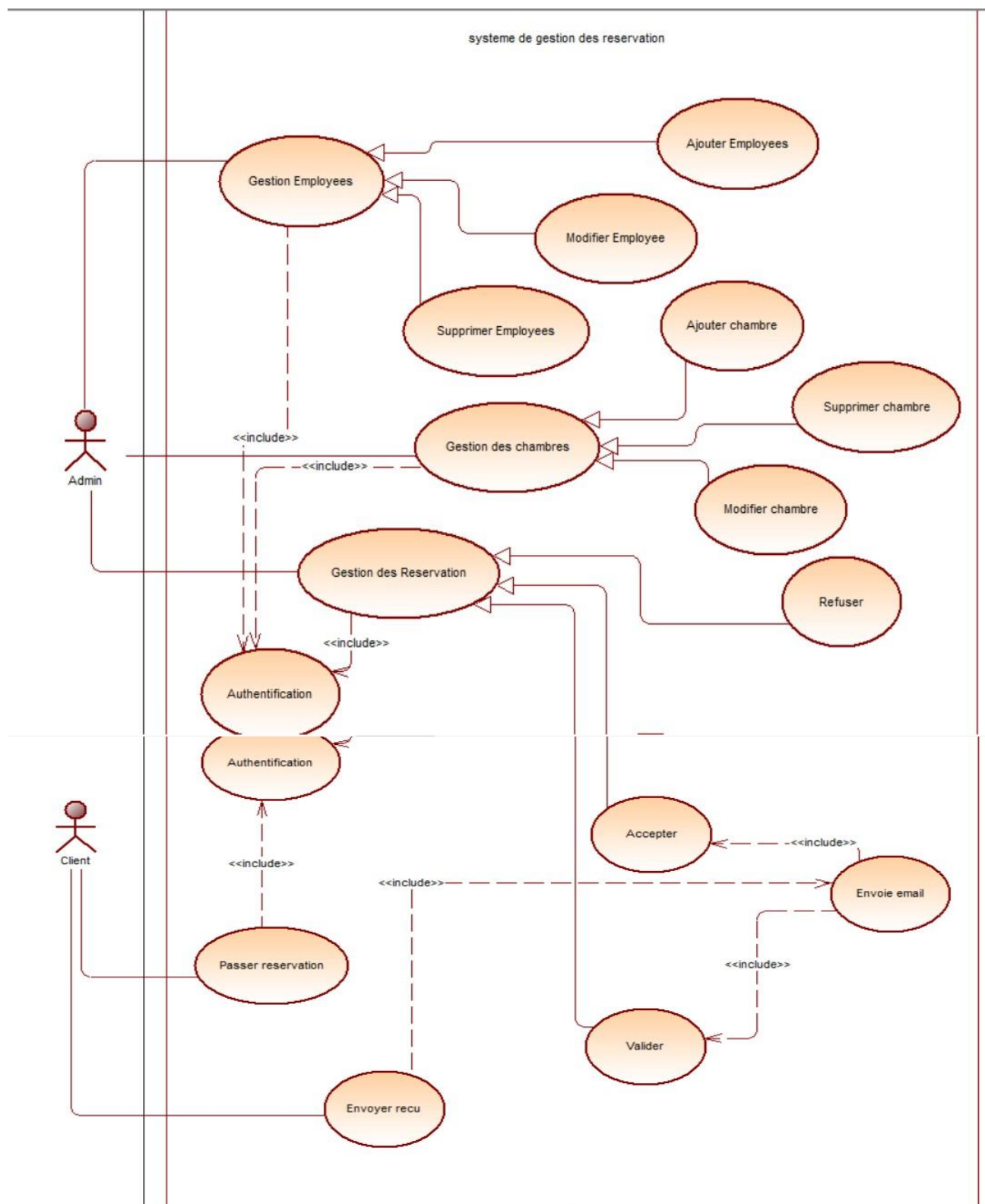
2.2. Cas d'Utilisation

Les cas d'utilisation principaux sont :

- **Back-office (Administrateur)**
 1. Gérer les utilisateurs (CRUD).
 2. Gérer les clients (CRUD).
 3. Gérer les chambres et leur disponibilité.
 4. Gérer les réservations et paiements.
 5. Exporter ou importer les données (Excel/CSV).
 6. Générer des rapports PDF et des tableaux de bord.
 7. Envoyer des notifications par email.
- **Front-office (Client)**
 1. Créer un compte.
 2. Rechercher et réserver des chambres.
 3. Recevoir une confirmation par email.

2.3. Représentation Visuelle

Le diagramme de cas d'utilisation peut être visualisé comme suit :



3. Diagramme de Classes

Le diagramme de classes modélise les entités principales du système, leurs attributs, méthodes, et relations.

3.1. Classes Principales

1. User

- **Attributs** : UserId, Nom, Email, MotDePasse, Rôle.
- **Méthodes** : CréerUtilisateur(), ModifierUtilisateur(), SupprimerUtilisateur().

2. Client

- **Attributs** : ClientId, Nom, Email, Téléphone, Adresse.
- **Méthodes** : CréerCompte(), ModifierProfil().

3. RoomType

- **Attributs** : TypeId, Nom, Description, Tarif.
- **Méthodes** : AjouterTypeChambre(), ModifierTypeChambre().

4. Room

- **Attributs** : RoomId, Numéro, TypeId, Disponibilité.
- **Méthodes** : MettreÀJourDisponibilité().

5. Reservation

- **Attributs** : ReservationId, DateDébut, DateFin, Statut, ClientId.
- **Méthodes** : CréerReservation(), ModifierReservation(), AnnulerReservation().

6. Payment

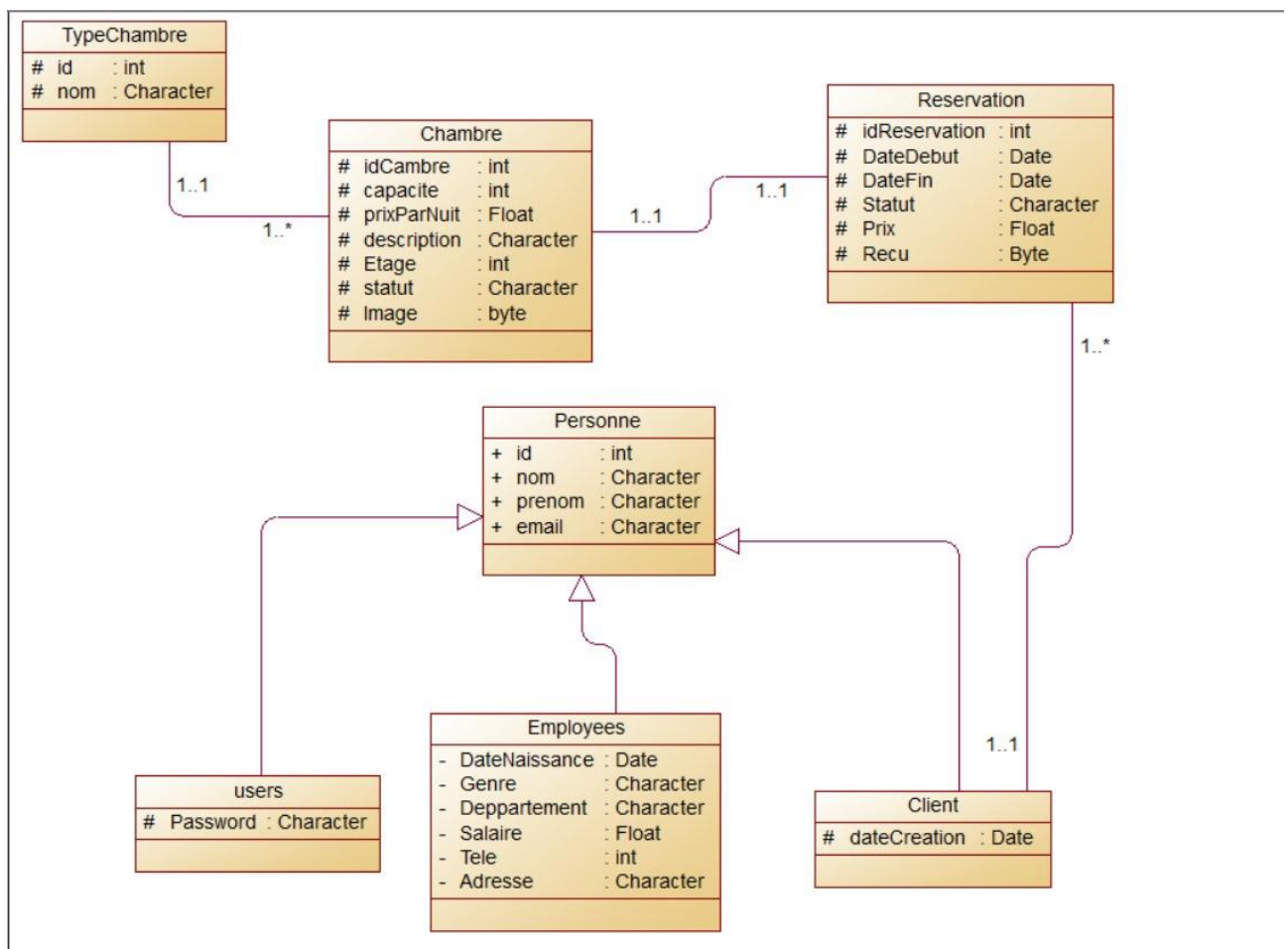
- **Attributs** : PaymentId, Montant, Date, Mode, ReservationId.
- **Méthodes** : EnregistrerPaieement().

3.2. Relations

- Une classe `Client` peut être liée à plusieurs `Réservations`.
- Une classe `Room` est associée à un `RoomType`.
- Une `Réservation` est associée à une ou plusieurs `Rooms` et un `Payment`.

3.3. Représentation Visuelle

Un exemple de diagramme de classes serait comme suit :



4.Outil de conception :



PowerAMC (Application Modeling and Construction), également connu sous le nom de **SAP PowerDesigner**, est un outil de modélisation puissant utilisé pour concevoir et documenter des bases de données, des systèmes d'information et des architectures logicielles. Développé à l'origine par

Sybase, puis acquis par SAP, il est largement utilisé dans les projets informatiques pour sa capacité à prendre en charge plusieurs types de modèles et à assurer une communication efficace entre les équipes de développement et les parties prenantes.

5.Conclusion :

Le développement du système de gestion d'hôtel repose sur des outils et concepts modernes, garantissant une conception claire et efficace. L'utilisation de diagrammes UML, réalisés avec PlantUML, permet de modéliser de manière structurée les interactions, les classes et les relations au sein du système. Cette approche a permis de définir précisément les fonctionnalités clés du système, d'identifier les relations entre les acteurs (administrateurs et clients) et les modules techniques, et de faciliter la communication entre les membres de l'équipe ainsi que les parties prenantes. Ainsi, la modélisation UML a jeté des bases solides pour le développement du projet, assurant une évolutivité et une maintenabilité optimales à long terme.

Chapitre 2 : Etude Fonctionnelle et Technique

1. Introduction

L'étude fonctionnelle et technique constitue une étape cruciale dans le développement d'un système informatique. Elle permet d'identifier les outils, les méthodes et les architectures les mieux adaptés pour répondre aux besoins exprimés lors de l'analyse des exigences. Dans ce chapitre, nous explorons les technologies utilisées, définissons l'architecture du système, et décrivons le processus de déploiement prévu pour assurer une mise en production optimale.

2. Technologies utilisées

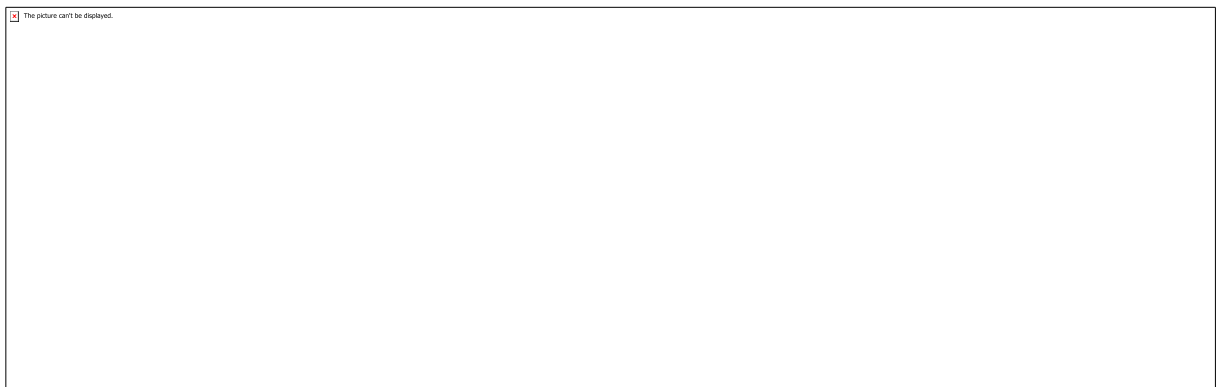
Ce projet repose sur des technologies modernes pour répondre aux besoins des différentes interfaces et garantir une expérience utilisateur optimale :

- **Frontend (interface utilisateur) :**
 - **Tailwind CSS** : Utilisé pour la conception rapide et responsive des interfaces web.



- **JavaScript et HTML/CSS** : Pour développer des interfaces dynamiques et interactives conformes au modèle MVC.

L'HTML permet l'affichage des informations de la page sur le navigateur, le CSS met en page son contenu et JavaScript intègre les animations et spécifie les interactions entre les éléments.



- **Backend et logique métier :**

WPF (Windows Presentation Foundation), un framework d'interface utilisateur permettant de créer des applications clientes de bureau riches en fonctionnalités. WPF offre un large éventail de possibilités, telles que la gestion des ressources, des contrôles, des graphiques, la liaison de données et la sécurité, tout en étant intégré dans le cadre de .NET. L'utilisation du langage **XAML** pour la programmation permet un modèle déclaratif, facilitant ainsi la création de l'interface.

Front-office (Web), il est basé sur **ASP.NET Core MVC**, un framework puissant pour la génération d'applications web et d'API utilisant le modèle de conception **Model-View-Controller (MVC)**, offrant ainsi une structure flexible et scalable pour le développement d'applications web interactives et performantes.

- **Base de données et persistance :**

MySQL est un logiciel qui permet de stocker, organiser, et gérer des données de manière structurée. Il offre un environnement pour la manipulation, la recherche, et la récupération des données [6].



- **La couche d'accès aux données : ADO.NET**

ADO.NET est un ensemble de classes qui exposent les services d'accès aux données pour les programmeurs .NET Framework. ADO.NET propose un large ensemble de composants pour la création d'applications distribuées avec partage de données. Partie intégrante du .NET Framework, il permet d'accéder à des données relationnelles, XML et d'application. ADO.NET répond à divers besoins en matière de développement, en permettant notamment de créer des clients de bases de données frontaux et des objets métier de couche intermédiaire utilisés par des applications, outils, langages ou navigateurs Internet. [3]

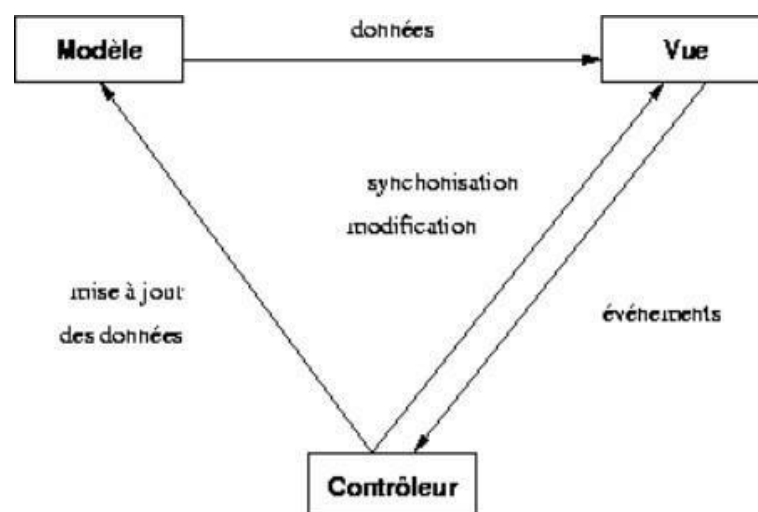


3. Architecture et Déploiement

Le projet repose sur l'architecture **MVC** (Modèle-Vue-Contrôleur), un modèle standard largement utilisé pour structurer les interfaces graphiques et les systèmes logiciels complexes. Cette architecture, bien qu'elle puisse introduire une certaine complexité initiale, offre une séparation claire des responsabilités, ce qui facilite la maintenance, l'évolutivité et la réutilisation des composants.

L'architecture MVC repose sur trois entités principales, qui jouent les rôles suivants :

- **Modèle (Model) :**
Le modèle représente la gestion des données de l'application. Il encapsule les données, garantit leur intégrité et fournit les mécanismes nécessaires pour leur manipulation, tels que l'insertion, la suppression et la modification. Dans le contexte de ce projet, le modèle est responsable de l'interaction avec la base de données SQL Server via ADO.NET ou Entity Framework. Il offre également des méthodes pour exposer les données aux autres couches, en permettant des vues partielles ou complètes selon les besoins.
- **Vue (View) :**
La vue est responsable de la présentation des données à l'utilisateur. Dans ce projet, les interfaces utilisateur sont développées avec **Tailwind CSS, HTML/CSS, et JavaScript** pour le front-office et avec **WPF** pour le back-office. La vue ne contient aucune logique métier, ce qui permet de séparer clairement l'apparence et le comportement du système.
- **Contrôleur (Controller) :**
Le contrôleur agit comme intermédiaire entre la vue et le modèle. Il gère les interactions des utilisateurs, traite les requêtes, applique les règles métier et met à jour les modèles ou les vues en conséquence. Dans ce projet, la couche métier et les contrôleurs sont implémentés dans **ASP.NET Core MVC**, centralisant la gestion des règles et des processus métier.



Déploiement

Le déploiement du système est structuré pour assurer un accès fiable et sécurisé :

- **Application web** : Déployée sur un serveur web ou cloud (Azure, AWS ou IIS), elle est accessible aux clients pour les opérations de réservation.
- **Application desktop** : Installée localement sur les postes administrateurs, elle offre une interface dédiée à la gestion interne.
- **Base de données centralisée** : Hébergée sur un serveur sécurisé, elle est accessible via des API ou des requêtes directes à partir des applications.

4. Conclusion

L'étude fonctionnelle et technique a permis de poser les bases solides pour le développement du projet en alignant les choix technologiques avec les besoins métiers. Les technologies choisies, telles que Tailwind CSS, JavaScript, WPF, et ASP.NET Core MVC, garantissent une performance optimale, une évolutivité et une facilité de maintenance. La séparation en couches de l'architecture assure une modularité et facilite les futures améliorations.

Chapitre 3 : Réalisation de L'Application

1. Introduction

Dans ce chapitre, nous allons présenter la réalisation de l'application de gestion hôtelière en détaillant les différentes étapes de son développement. Nous décrirons la conception des interfaces, les principales fonctionnalités implémentées, et l'organisation des différentes sections de l'application, en mettant un accent particulier sur l'optimisation de l'expérience utilisateur. Les interfaces du front-office, dédiées aux clients pour les réservations en ligne, et du back-office, destinées aux administrateurs pour la gestion des chambres, des clients et des réservations, seront explorées en détail. Ce chapitre mettra également en lumière les choix techniques effectués et les solutions adoptées pour garantir une application performante, sécurisée et facile à utiliser, répondant aux objectifs définis dans les étapes précédentes du projet.

2.Présentation graphique :

1. Partie Desktop :

a. Authentification et Gestion de Compte :

L'écran d'authentification permet aux administrateurs de se connecter à l'application pour accéder aux fonctionnalités du Back-office. Il comporte l'email et le mot de passe

A screenshot of a web browser window titled "Login". The address bar shows a URL starting with "http://localhost:". The page contains two input fields: one labeled "Email" and another labeled "Password". Below the "Password" field is a green button with the text "Login". The browser's developer tools are open at the top, displaying various icons like zoom, copy, paste, etc.

b. Gestion de la disponibilité des chambres :

La gestion de la disponibilité des chambres est une fonctionnalité clé dans la partie **Back-office** de l'application. Cet écran permet aux administrateurs de suivre et de mettre à jour l'état des chambres en temps réel pour garantir une gestion optimale des réservations et savoir le nombre

[illegible]

total des chambres et les chambres disponibles, occupée et sous maintenance et aussi modifier les informations de ces chambres

Ajouter une Chambre

Numéro Chambre:

Type de Chambre:

Capacité:

Prix par Nuit (en €):

Description:

Choisir Image

Étage:

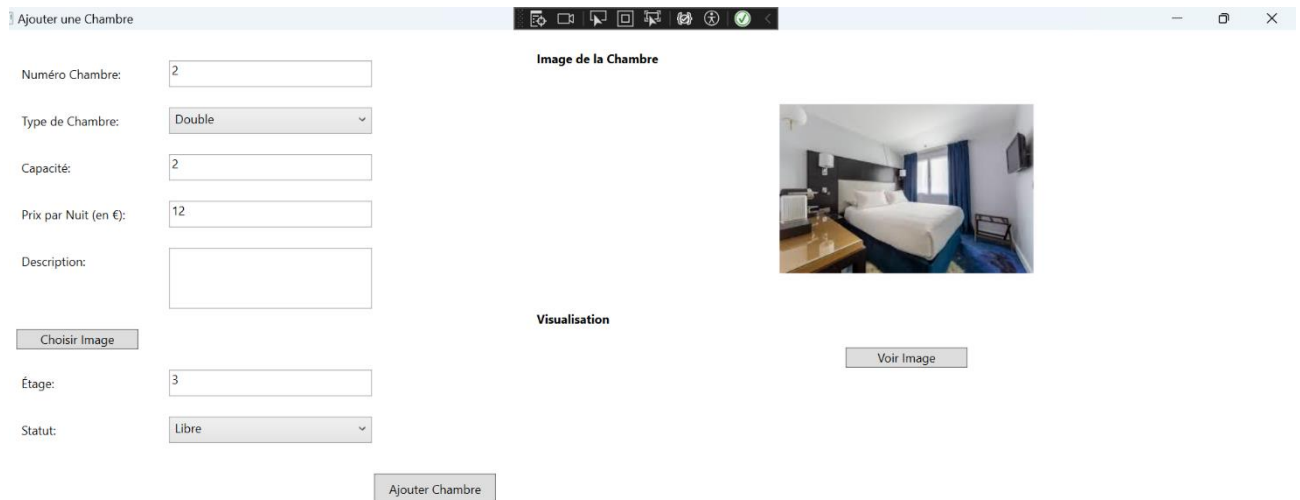
Statut:

Ajouter Chambre

Image de la Chambre

Visualisation

Voir Image




Enregistrement automatique

EmployeeList • Enregistré dans ce PC

Fichier Accueil Insertion Mise en page Formules Données Révision Affichage Automatiser Développeur Aide

Coller Police Alignement Nombre Mise en forme conditionnelle Mettre sous forme de tableau Styles de cellules Styles

Presse-papiers

Cellules Édition Compléments Analyse de données

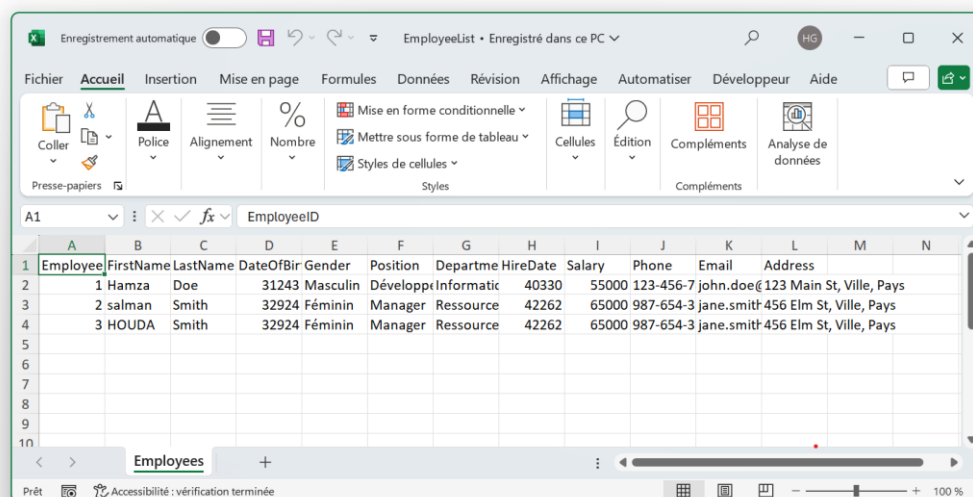
A1 EmployeeID

	A	B	C	D	E	F	G	H	I	J	K	L	M	N
	Employee	FirstName	LastName	DateOfBir	Gender	Position	Departme	HireDate	Salary	Phone	Email	Address		
1		1	Hamza	Doe	31243	Masculin	Développeur	Informatique	40330	55000	123-456-7	john.doe@123 Main St, Ville, Pays		
2		2	salman	Smith	32924	Féminin	Manager	Ressource	42262	65000	987-654-3	jane.smith@456 Elm St, Ville, Pays		
3		3	HOUDA	Smith	32924	Féminin	Manager	Ressource	42262	65000	987-654-3	jane.smith@456 Elm St, Ville, Pays		
4														
5														
6														
7														
8														
9														
10														

Employees

Prêt Accessibilité : vérification terminée

100 %



c. Notification de réservation

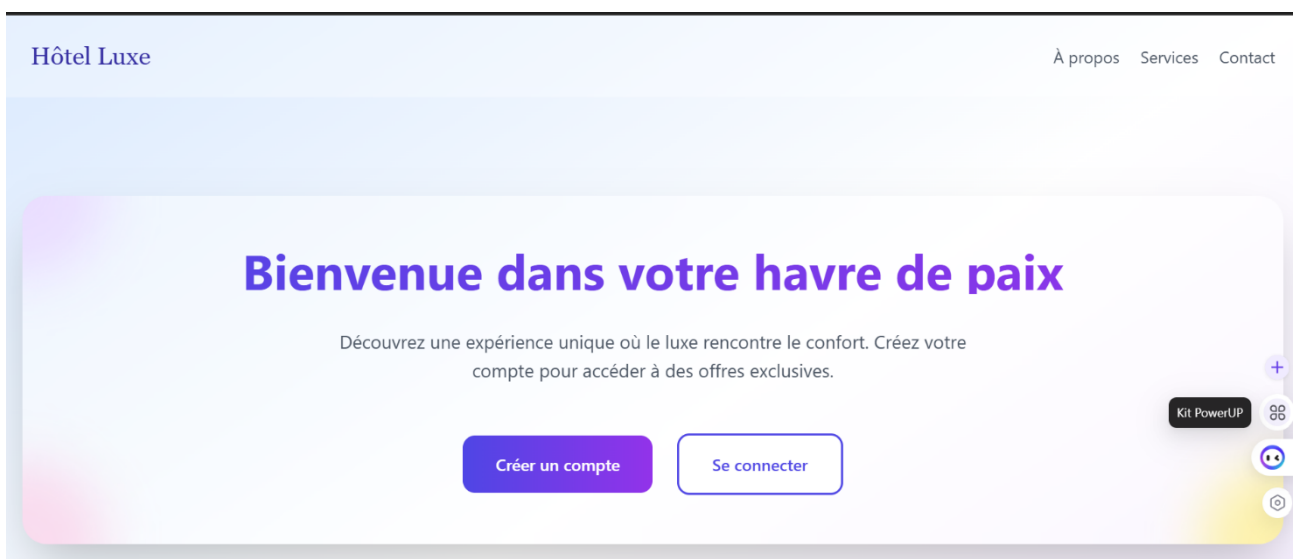
ID	Réservation	Numéro	Chambre	ID Client	Date Début	Date Fin	Statut	Date Création	Prix	Photo
0	0	0	Aucune nouvelle réservation aujourd'hui			04/01/2025	\$0.00	Accepter	Refuser	Valider

Actualiser

2.Partie Web :

a. Page d'Accueil :

La page **Home** constitue la vitrine principale de l'application Front-office, offrant une première impression engageante et professionnelle aux visiteurs. Elle met en avant les services et les atouts de l'hôtel tout en facilitant la navigation vers les autres sections de l'application.



b. Interface de Création de Compte Utilisateur

L'interface de création de compte permet aux utilisateurs de s'inscrire facilement sur l'application de réservation d'hôtel.

Hôtel Luxe

Créer un compte

Nom
Entrez votre nom

Prénom
Entrez votre prénom

Email
Entrez votre email

Mot de passe
Entrez un mot de passe

S'inscrire

c. Interface pour se connecter

L'interface permet à l'utilisateur d'entrer leur e-mail et mode passe

Hôtel Luxe

Se connecter

Email
gmairhouda148@gmail.com

Mot de passe
.....

Se connecter

Pas encore de compte ? [S'inscrire ici](#)

d. . Détails de la Chambre :

une interface dédiée à la présentation des détails spécifiques d'une chambre sélectionnée. Cette page est conçue pour offrir aux clients une vue complète des caractéristiques et des commodités de la chambre, renforçant ainsi leur décision de réservation

[illegible]

e. Interface de reservation

L'objectif est de permettre aux clients de réserver des chambres en ligne

Réserver la chambre 2

Date de début

jj/mm/aaaa

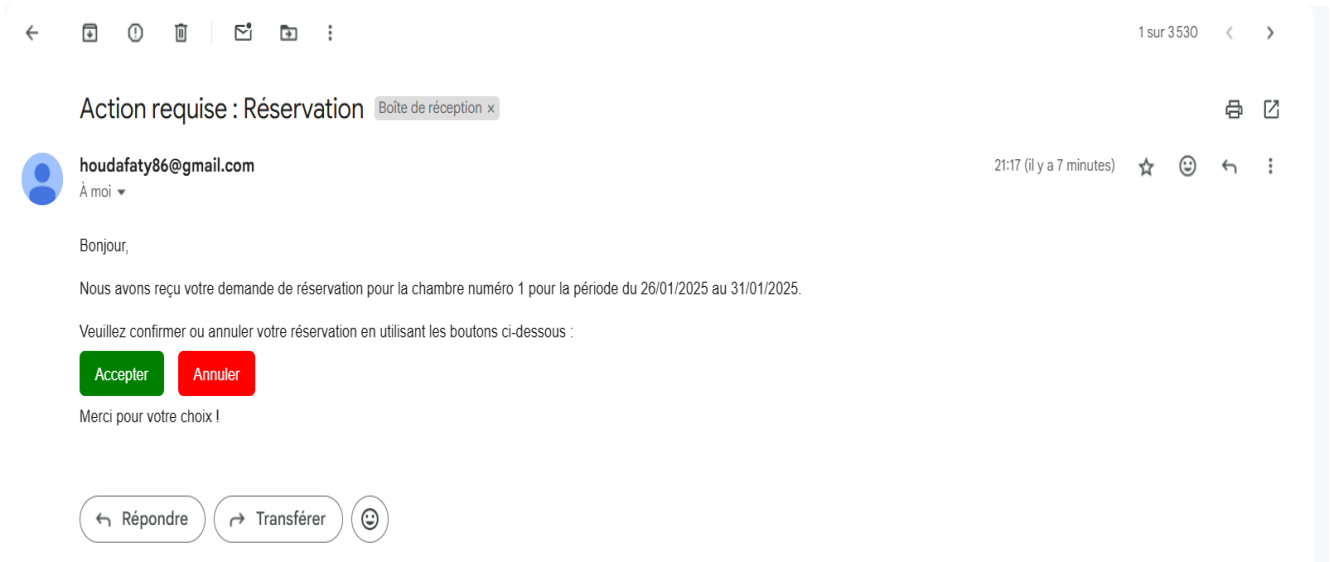
Date de fin

jj/mm/aaaa

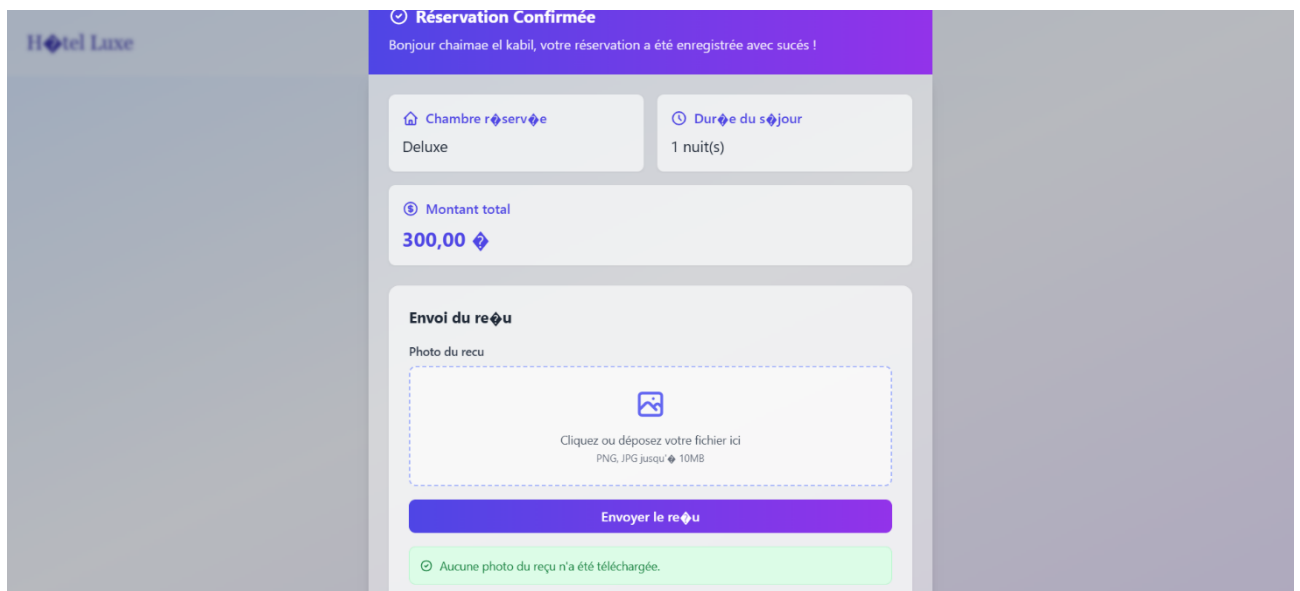
Confirmer la réservation

f. Notification par email :

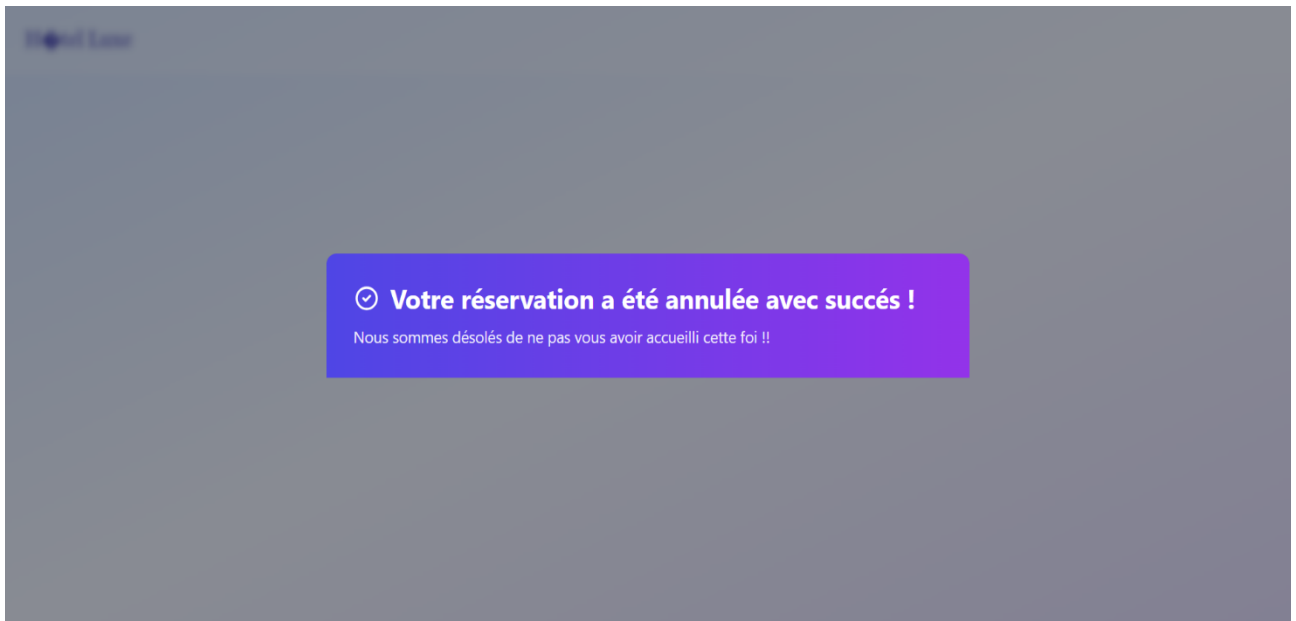
La fonctionnalité de **Notification par Email** permet d'automatiser l'envoi de messages électroniques Elle améliore la communication entre l'hôtel et ses clients en offrant liés aux opérations de réservation



g. Confirmation de la réservation :



h. Interface ou réservation peuvent être annuler par l'admin :



3. Conclusion

Ce chapitre a permis de mettre en lumière le développement concret de l'application et ses interfaces graphiques, traduisant les besoins identifiés en fonctionnalités opérationnelles. Les choix techniques et les interfaces développées assurent une expérience utilisateur fluide et une gestion efficace pour les administrateurs. Cette étape représente une réalisation majeure dans le processus de développement du projet, ouvrant la voie aux tests et validations à venir.

Bibliographie :

[1] [Qu'est-ce que SQL Server ? - SQL Server | Microsoft Learn](#)

[2] [Vue d'ensemble d'ASP.NET Core MVC | Microsoft Learn](#)

[3] [ADO.NET | Microsoft Learn](#)

[4] [Application Hello World avec WPF en C# - Visual Studio \(Windows\) | Microsoft Learn](#)

[5] [Vue d'ensemble d'ASP.NET Core MVC | Microsoft Learn](#)

Lien GitHub :

<https://github.com/Houda-11G/DotnetG6>