

SYSTÈME DE GESTION HÔTELIÈRE

Rapport de projet

Cycle Ingénieur Analyse des Méga-données

Année académique 2024/2025

Encadré par :

AMAMOU Ahmed Réalisé par :

HANINI Marua HAMIM Aya

EL ACHABI Nada Ezzouhour

INDEX :

1. Introduction :

1.1 Contexte du projet………………………………………………………………………

* 1. Objectifs de l'application……………………………………………………………….
  2. Public cible……………………………………………………………………………..

1. Analyse des Besoins :

2.1 Fonctionnalités principales……………………………………………………………..

2.2 Cas d'utilisation…………………………………………………………………………

* 1. Diagrammes UML (cas d'utilisation, classes, séquence)………………………………

3. Interface Utilisateur

3.1 Maquettes des écrans…………………………………………………………………..

4. Base de Données

4.1 Schéma de la base de données………………………………………………………….

4.2 Tables et relations……………………………………………………………………….

5. Conclusion

5.1 Bilan du projet…………………………………………………………………………..

5.2 Perspectives d'évolution…………………………………………………………………

INTRODUCTION :

1.1 Contexte du projet

Le secteur de l'hôtellerie est en pleine croissance, et la gestion manuelle des réservations, des clients, des employés et des stocks devient de plus en plus complexe et sujette à des erreurs. Pour répondre à ces défis, ce projet vise à développer une  **application de gestion d'hôtel** en Java, utilisant les bibliothèques Swing et AWT pour l'interface graphique et SQL pour le stockage des données. Cette application centralise toutes les opérations de gestion sous le contrôle d'un \*administrateur unique\*, garantissant ainsi une gestion sécurisée et efficace des ressources de l'hôtel.

L'application permettra de gérer les réservations de chambres, l'ajout, la suppression et la modification des employés et des clients, ainsi que la mise à jour des informations en temps réel. En automatisant ces processus et en stockant les données de manière structurée dans une base de données SQL, l'application réduira les erreurs humaines, améliorera l'efficacité opérationnelle et offrira une meilleure expérience aux clients et au personnel.

1.2 Objectifs de l'application

Les objectifs principaux de cette application sont les suivants :

1. Centralisation de la gestion :

- Permettre à un seul administrateur de contrôler toutes les opérations de l'hôtel (réservations, gestion des clients, gestion des employés, gestion des stocks).

- Offrir une interface unique pour gérer les ajouts, modifications et suppressions.

2. Gestion des réservations :

- Faciliter la création, la modification et l'annulation des réservations de chambres.

- Afficher en temps réel la disponibilité des chambres.

3. Gestion des clients :

- Ajouter, modifier ou supprimer les informations des clients.

- Conserver un historique des séjours des clients pour un meilleur suivi.

4. Gestion des employés :

- Ajouter, modifier ou supprimer les informations des employés.

- Attribuer des rôles et des responsabilités aux employés.

6. Stockage des données :

- Utiliser une base de données SQL pour stocker de manière sécurisée et structurée toutes les informations relatives aux réservations, clients, employés et stocks.

- Assurer l'intégrité et la cohérence des données grâce aux fonctionnalités de SQL (tables, relations, contraintes).

7. Interface utilisateur intuitive :

- Utiliser Swing et AWT pour créer une interface graphique simple et conviviale.

- Permettre une navigation facile entre les différentes fonctionnalités.

1.3 Public cible

Le public cible de cette application est principalement composé des acteurs suivants :

1. Administrateur de l'hôtel :

- Rôle : Gérer toutes les opérations de l'hôtel via l'application.

- Besoins :

- Contrôler les réservations, les clients, les employés et les stocks.

- Avoir un accès complet pour ajouter, modifier ou supprimer des données.

- Générer des rapports pour suivre les performances de l'hôtel.

- Utiliser une base de données SQL pour stocker et récupérer les données de manière fiable.

2. Clients de l'hôtel (indirectement) :

- Rôle : Bénéficier d'un service de réservation et de gestion de séjour fluide.

- Besoins :

- Avoir des réservations précises et sans erreur.

- Pouvoir compter sur un système de gestion fiable pour leurs demandes spéciales.

ANALYSE DE BESOIN

2.1 Fonctionnalités principales

L'application de gestion d'hôtel offre les fonctionnalités suivantes :

Gestion des réservations :

* + Créer, modifier et annuler des réservations.
  + Associer une chambre à une réservation.
  + Vérifier la disponibilité des chambres.
  + Afficher les détails des réservations.

Gestion des clients :

* + Ajouter, modifier et supprimer des clients.
  + Consulter les informations des clients (nom, email, historique des réservations).

Gestion des chambres :

* + Ajouter, modifier et supprimer des chambres.
  + Afficher la liste des chambres disponibles.
  + Vérifier le statut des chambres (disponible, occupée, en nettoyage).

Gestion des employés :

* + Ajouter, modifier et supprimer des employés.
  + Mettre à jour les informations des employés.

Interface administrateur :

* + Contrôle centralisé des réservations, clients, chambres et employés.
  + Génération de rapports et confirmation des actions (ajout, modification, suppression).

Stockage des données :

* + Utilisation d'une base de données SQL pour stocker les informations des réservations, clients, chambres et employés.

2.2 Cas d'utilisation

Les principaux cas d'utilisation de l'application sont les suivants :

1. Réserver une chambre :
   * Le client ou l'administrateur vérifie la disponibilité des chambres.
   * Une chambre disponible est sélectionnée et associée à une réservation.
   * Les détails de la réservation sont confirmés et enregistrés.
2. Gérer les clients :
   * L'administrateur ajoute un nouveau client avec ses informations (nom, email).
   * L'administrateur peut modifier ou supprimer les informations d'un client existant.
3. Gérer les chambres :
   * L'administrateur ajoute une nouvelle chambre avec ses détails (numéro, type, prix).
   * L'administrateur peut modifier ou supprimer une chambre existante.
   * L'administrateur vérifie le statut des chambres (disponible, occupée, en nettoyage).
4. Gérer les employés :
   * L'administrateur ajoute un nouvel employé avec ses informations.
   * L'administrateur peut modifier ou supprimer les informations d'un employé existant.
5. Vérifier la disponibilité des chambres :
   * L'employé ou l'administrateur consulte la liste des chambres disponibles.
   * Le statut des chambres est mis à jour en temps réel.

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

2.3 Diagrammes UML

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

Diagramme de cas d'utilisation



Diagramme de classes

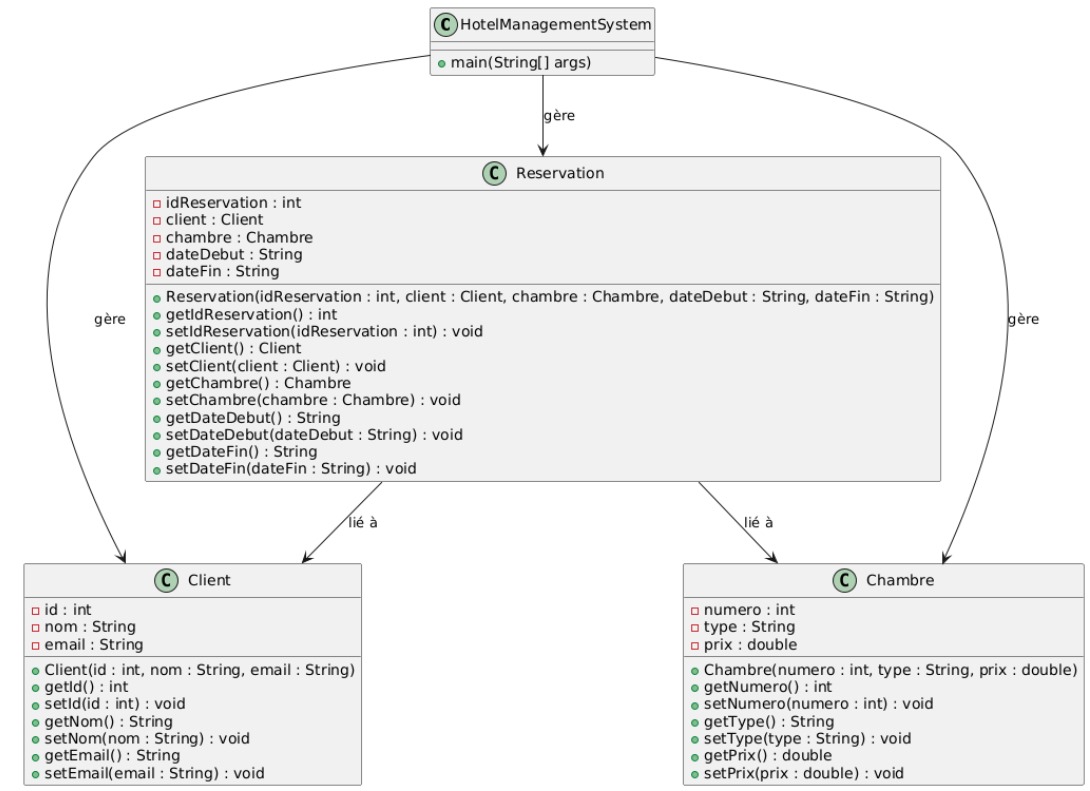
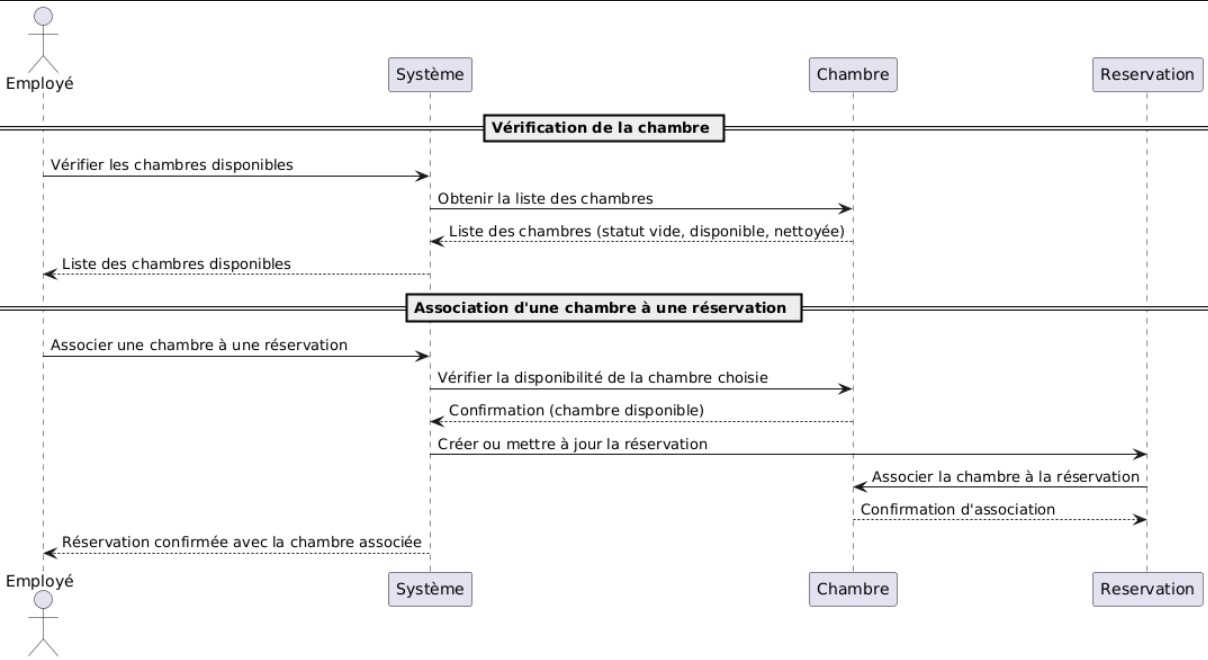
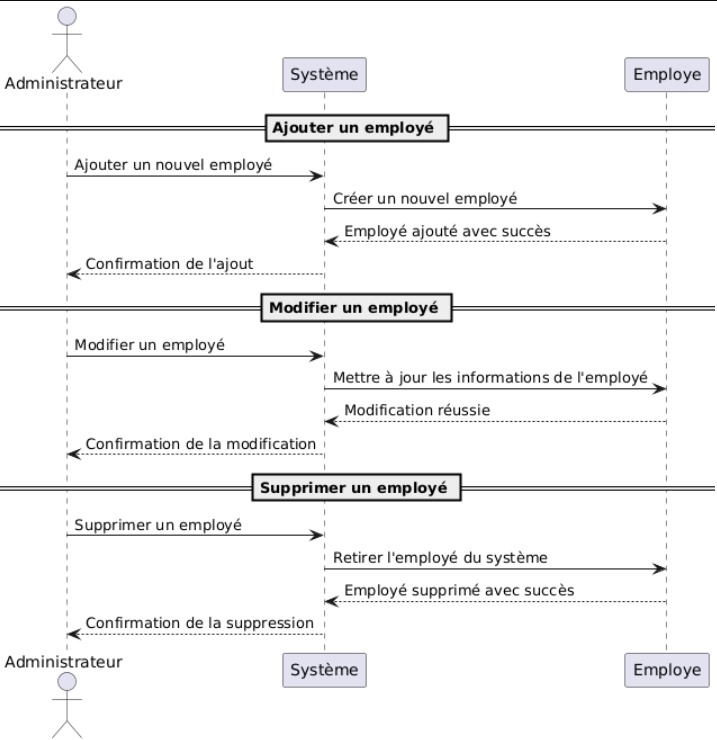


Diagramme de séquence





Cette section détaille les fonctionnalités principales, les cas d'utilisation et les diagrammes UML (cas d'utilisation, classes, séquence) de l'application de gestion d'hôtel. Les diagrammes UML illustrent les interactions entre les différents acteurs (administrateur, client, système) et les classes principales (Reservation, Client, Chambre). L'utilisation d'une base de données SQL garantit un stockage structuré et sécurisé des données.

INTERFACE UTILISATEUR

3.1 Maquettes des écrans

Les maquettes des écrans représentent les différentes interfaces de l'application. Voici une description des écrans principaux accompagnée de leurs captures d'écran :

3.1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

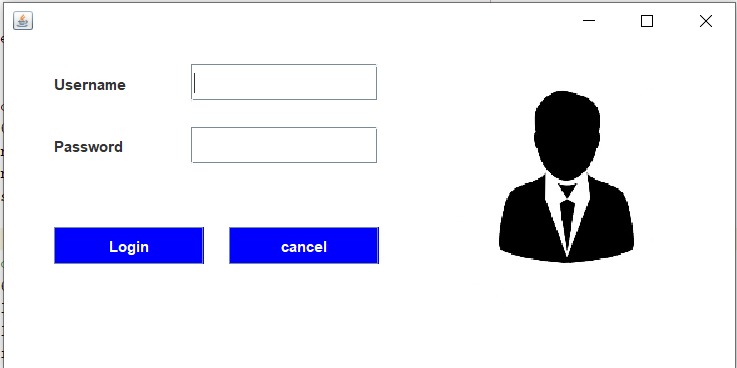
Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

Screenshot :



3.2 Tableau de bord (Dashboard)

Description :

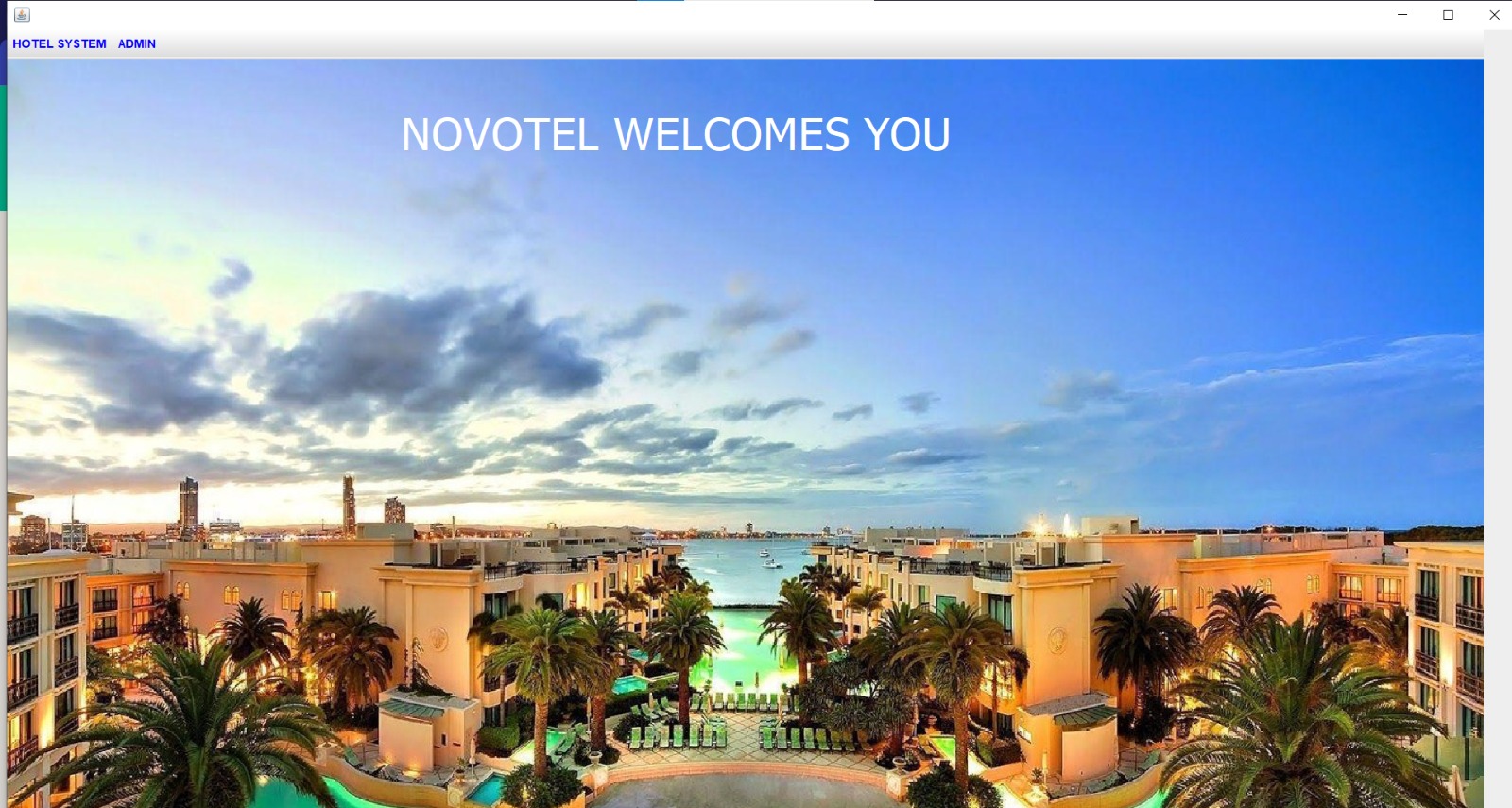
Cet écran affiche un aperçu des principales fonctionnalités de l'application (réservations, clients, chambres, employés).

Éléments clés :

Menu de navigation (réservations, clients, chambres, employés).

Statistiques rapides (nombre de réservations, chambres disponibles, etc.).

Screenshot



1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

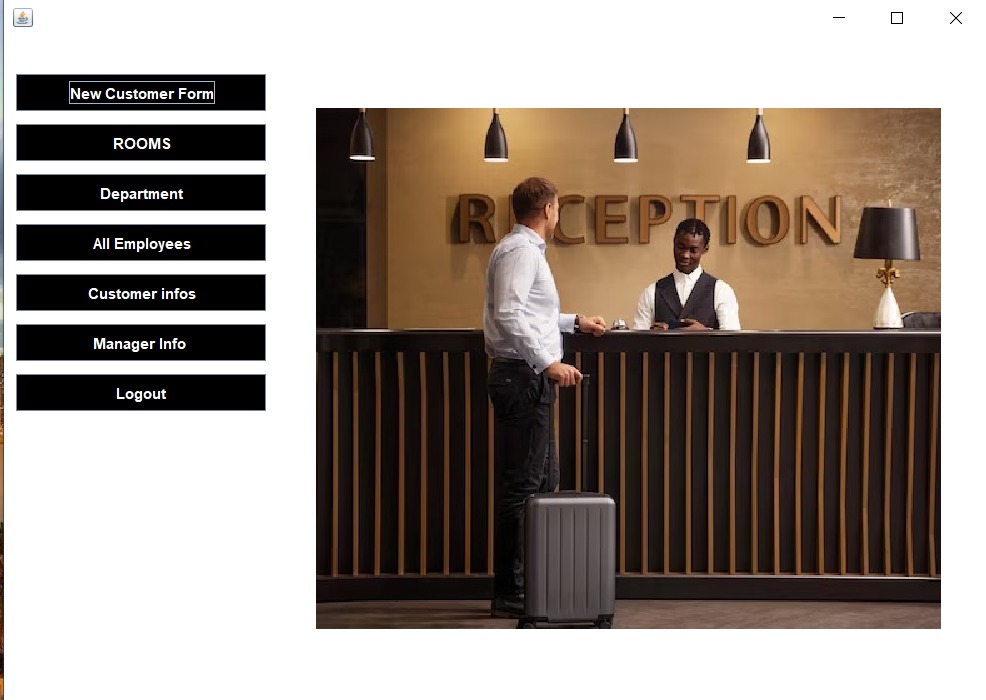
Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :



1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

3.3 Gestion des employés :

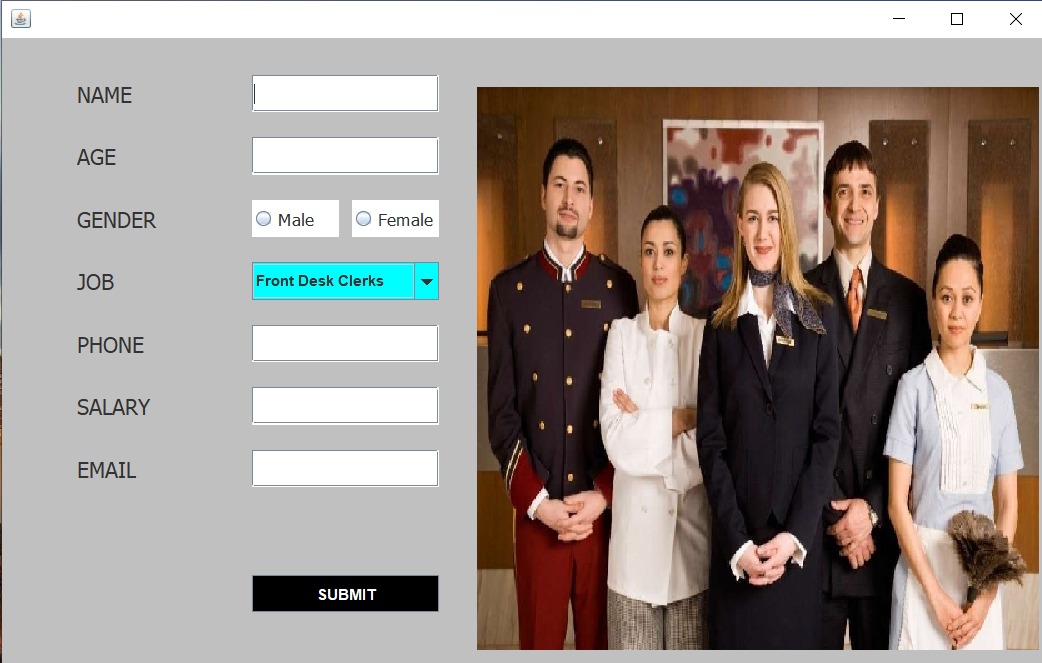
Description :

Cet écran permet de gérer les employés de l'hôtel (ajout, modification, suppression).

Éléments clés :

Liste des employés enregistrés.

Boutons pour ajouter, modifier ou supprimer un employé.

 Formulaire pour saisir les informations de l'employé.

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

3.4 . Gestion des chambres

Description :

Cet écran permet de gérer les chambres de l'hôtel (ajout, modification, suppression).

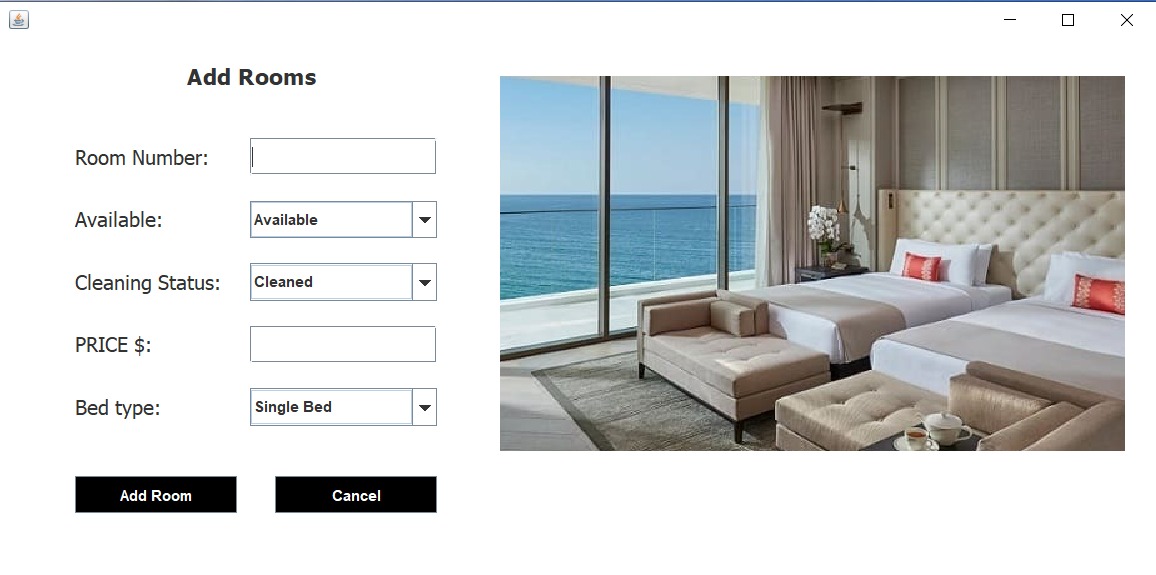
Éléments clés :

Liste des chambres avec leur statut (disponible, occupée, en nettoyage).

Boutons pour ajouter, modifier ou supprimer une chambre.

Formulaire pour saisir les détails de la chambre (numéro, type, prix).

Screenshot :



3.5. Gestion des clients

Description :

Cet écran permet d'ajouter, modifier ou supprimer des clients.

Éléments clés :

Liste des clients enregistrés.

Boutons pour ajouter, modifier ou supprimer un client.

Formulaire pour saisir les informations du client (nom, email).

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

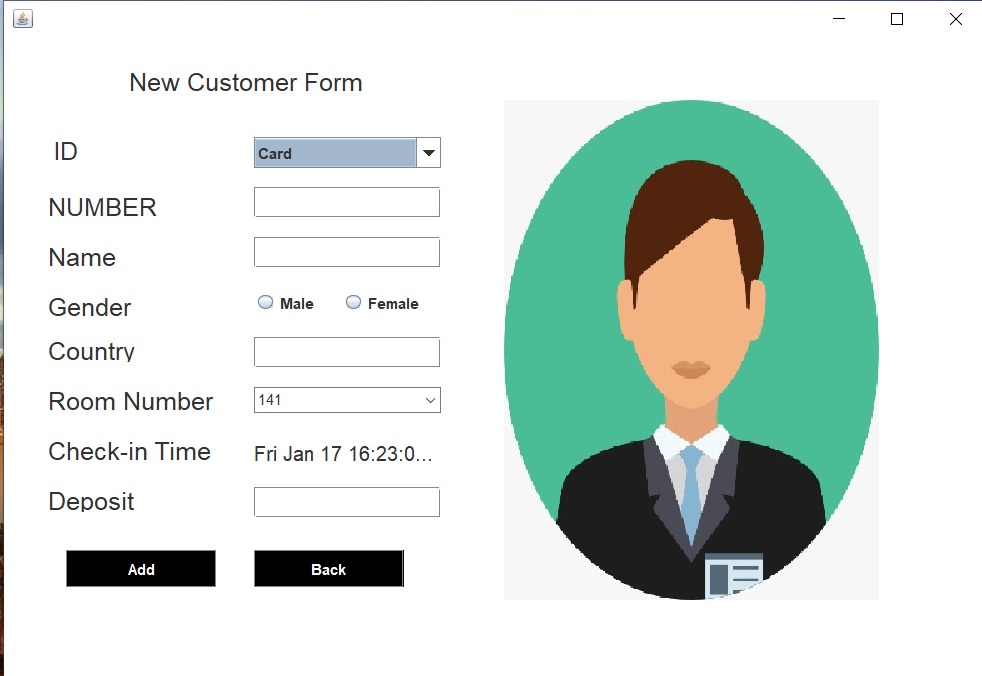
Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

Screenshot :



BASE DE DONNES

* 1. Schéma de la base de données

La base de données de l'application de gestion d'hôtel est conçue pour stocker de manière structurée les informations relatives aux réservations, clients, chambres et employés. Elle utilise un système de gestion de base de données relationnelle (SGBDR) comme MySQL Voici un aperçu du schéma de la base de données

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

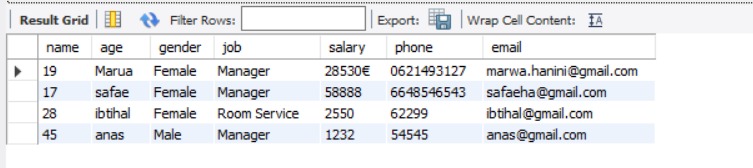
Champ pour saisir le mot de passe.

Bouton "Se connecter".

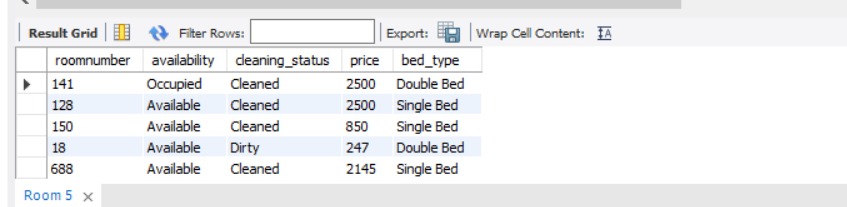
Screenshot :

Tables et relations

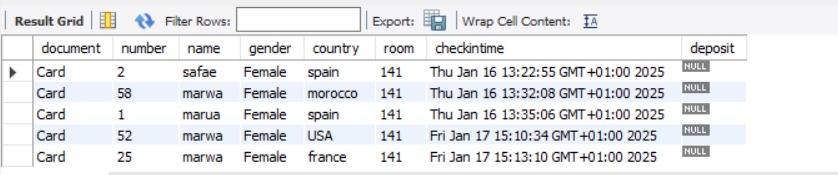
Employe :



Chambre :



Client :



1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

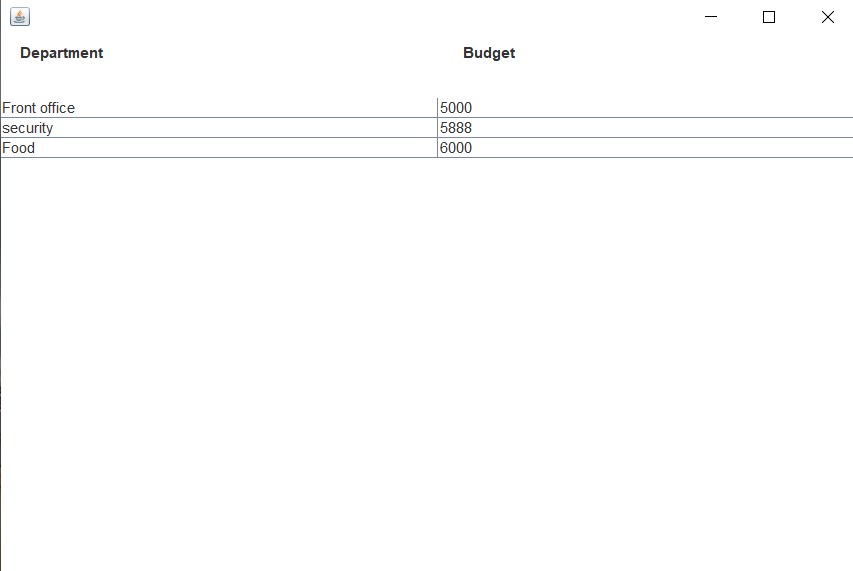
Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

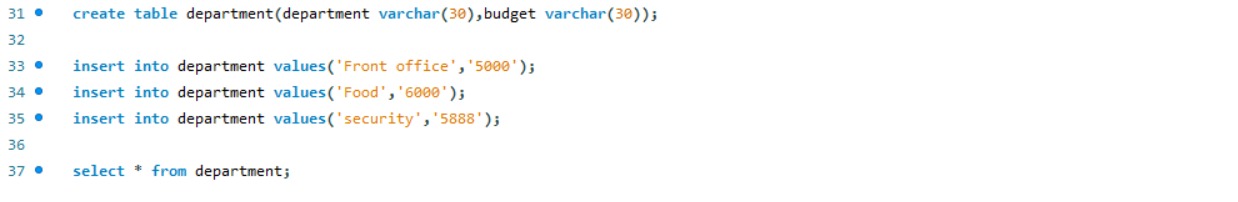
Departement :



LogOut :



Exemple de requêtes SQL :



1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

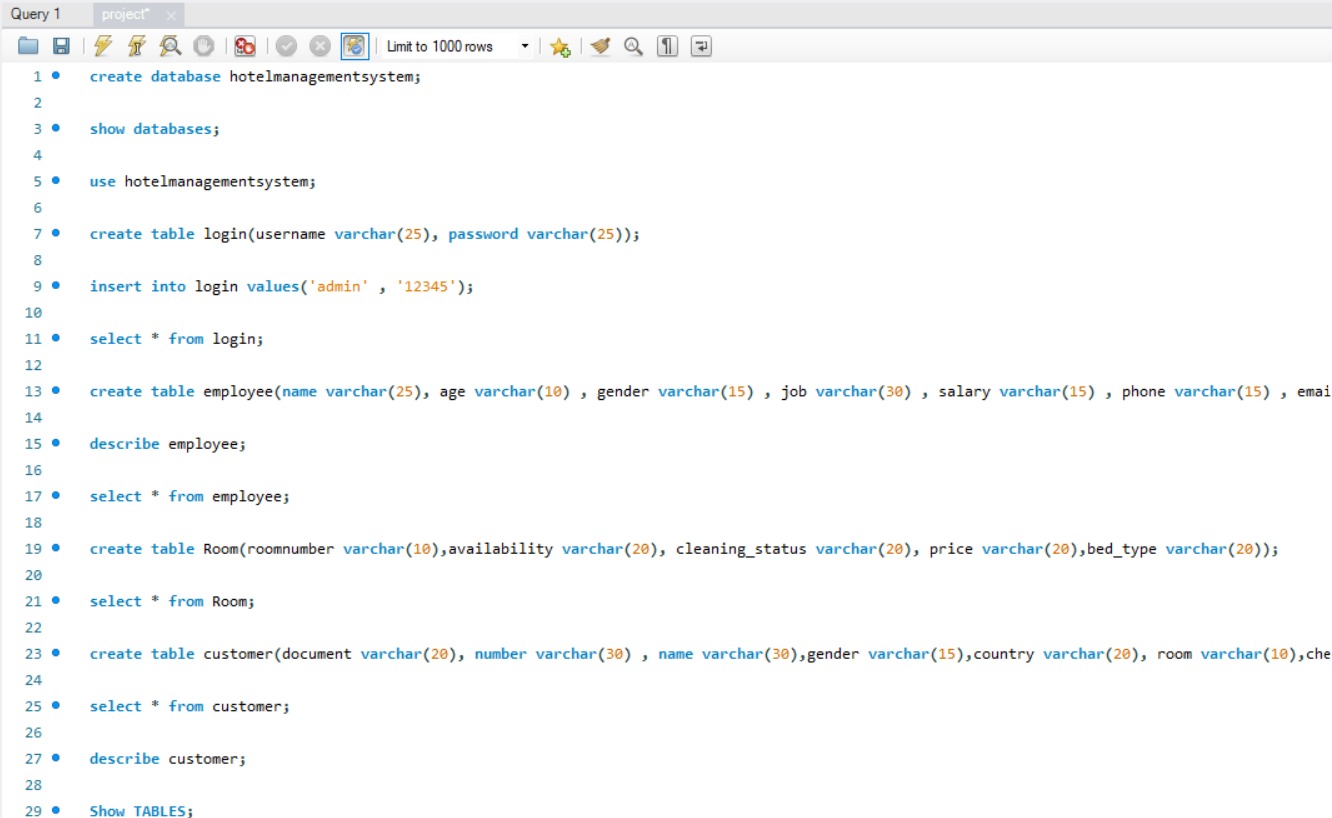
Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :



1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

CONCLUSION

5.1 Bilan du projet

Le projet de développement de l'application de gestion d'hôtel en Java, avec une interface graphique en Swing et AWT , et une base de données SQL, a été mené à bien avec succès. L'application répond aux objectifs initiaux en offrant une solution complète et centralisée pour la gestion des opérations d'un hôtel. Voici un bilan détaillé des réalisations :

1. Fonctionnalités clés implémentées :

- Gestion des réservations: L'application permet de créer, modifier et annuler des réservations, tout en associant les chambres aux clients de manière efficace.

- Gestion des clients : Les informations des clients (nom, email, historique des réservations) sont stockées et peuvent être mises à jour facilement.

- Gestion des chambres : Les chambres sont gérées avec précision (disponibilité, type, prix), et leur statut est mis à jour en temps réel.

- Gestion des employés : L'administrateur peut ajouter, modifier ou supprimer des employés, ce qui facilite la gestion du personnel.

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

2. Interface utilisateur intuitive :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

- L'utilisation de Swing et AWT a permis de créer une interface graphique simple et conviviale, adaptée aux besoins des utilisateurs finaux. La navigation entre les différentes fonctionnalités est fluide et intuitive.

1. Base de données robuste et sécurisée :

- La base de données SQL a été conçue pour garantir l'intégrité et la cohérence des données. Les relations entre les tables (Client, Chambre, Reservation, Employe) permettent une gestion efficace des informations.

1. Contrôle centralisé par l'administrateur :

- L'application est conçue pour être gérée par un seul administrateur, ce qui assure une gestion sécurisée et cohérente des données. Cela réduit les risques d'erreurs et de manipulations non autorisées.

5. Tests et validation :

- Des tests approfondis ont été réalisés pour valider le bon fonctionnement de l'application. Les retours des tests ont permis de corriger les bugs et d'améliorer la stabilité du système.

6.Documentation complète :

- Une documentation détaillée a été produite, incluant des diagrammes UML, des explications sur le code source et des guides d'utilisation pour les futurs utilisateurs.

5.2 Perspectives d'évolution :

Bien que l'application réponde aux besoins actuels, plusieurs améliorations et évolutions peuvent être envisagées pour renforcer ses fonctionnalités et son utilité. Voici quelques pistes d'évolution :

1. Modernisation de l'interface utilisateur :

- Remplacer Swing et AWT par des frameworks plus modernes comme JavaFx ou développer une interface web pour une expérience utilisateur plus riche et responsive.

1. Gestion des stocks et des services supplémentaires :

- Ajouter une fonctionnalité de gestion des stocks pour les produits de l'hôtel (articles de toilette, nourriture, boissons).

- Permettre la gestion des services supplémentaires (petit-déjeuner, spa, etc.) et leur facturation.

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :

1. Rapports et statistiques avancé :

- Intégrer des outils de reporting pour générer des rapports détaillés sur les réservations, les revenus, l'occupation des chambres, etc.

- Ajouter des graphiques et des tableaux de bord pour une visualisation plus claire des données.

1. Authentification multi-utilisateurs :

- Permettre à plusieurs utilisateurs (administrateurs, réceptionnistes) de se connecter avec des niveaux d'accès différents. Cela améliorerait la flexibilité de l'application.

1. Intégration de paiements en ligne :

- Ajouter une fonctionnalité de paiement en ligne pour permettre aux clients de payer directement via l'application, en utilisant des passerelles de paiement sécurisées.

1. Notifications et alertes :

- Implémenter un système de notifications pour informer les employés des nouvelles réservations, des chambres à nettoyer ou des stocks à réapprovisionner.

1. Application mobile :

- Développer une version mobile de l'application pour permettre aux clients de réserver des chambres et de gérer leurs réservations depuis leur smartphone. Cela améliorerait l'accessibilité et l'expérience utilisateur.

1. Amélioration de la base de données :

- Optimiser les requêtes SQL pour améliorer les performances de l'application.

- Ajouter des sauvegardes automatiques pour garantir la sécurité des données en cas de panne.

1. Intégration avec d'autres systèmes :

- Permettre l'intégration avec des systèmes externes, tels que des plateformes de réservation en ligne (Booking.com, Airbnb) ou des systèmes de gestion de relation client (CRM).

1. Support multilingue :

- Ajouter la prise en charge de plusieurs langues pour rendre l'application accessible à un public international.

11. Amélioration de la sécurité :

- Renforcer la sécurité de l'application en implémentant des mécanismes de chiffrement des données et en améliorant la gestion des mots de passe.

1. Écran de connexion (Login)

Description : Cet écran permet à l'administrateur de se connecter à l'application.

Éléments clés :

Champ pour saisir le nom d'utilisateur.

Champ pour saisir le mot de passe.

Bouton "Se connecter".

Screenshot :