

Mini-Projet d'Ingénierie de bases de données

Sujet Choisi: Sujet 7

Application JAVA de gestion de réservations de chambres d'Hôtel

Moula Mariem & Slaoui Yassine

Enseignante: FEKIH Tayssir

Faculté des Sciences de Tunis

Université de Tunis El Manar

Note de l'auteur

Premier paragraphe : La Répartition des tâches.

Deuxième paragraphe : Le Diagramme des cas d'utilisations.

Troisième paragraphe : Le Modèle Entité-Association.

Quatrième paragraphe : Les Imprimés écran.

Cinquième paragraphe : Divulgence et remerciements.

Description du Projet

Vu la complexité des activités d'un Hôtel, l'outil informatique est un élément qui lui facilite une bonne gestion pour une meilleure prise des décisions dans la vie.

Dans ce contexte, ce projet s'est focalisé sur une des problématiques de la gestion des Hôtel qui est la gestion des réservations des chambres car dans le domaine des Hôtels, la vitesse de traitement des réservations est fondamentale, c'est pour cela que l'automatisation et l'informatisation de la gestion d'un Hôtel est devenue indispensable.

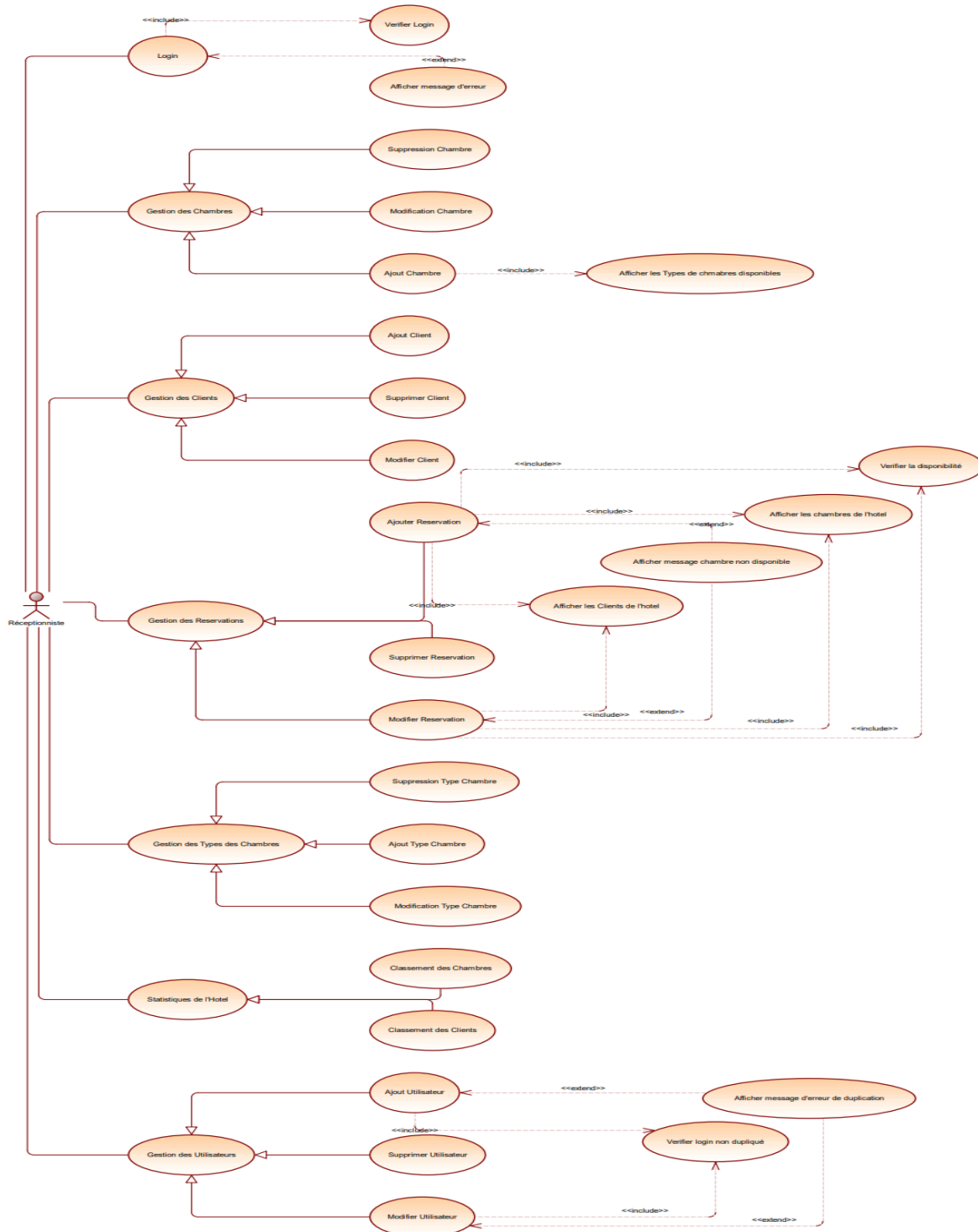
La répartition des tâches.

Moula Mariem : Frontend et lien

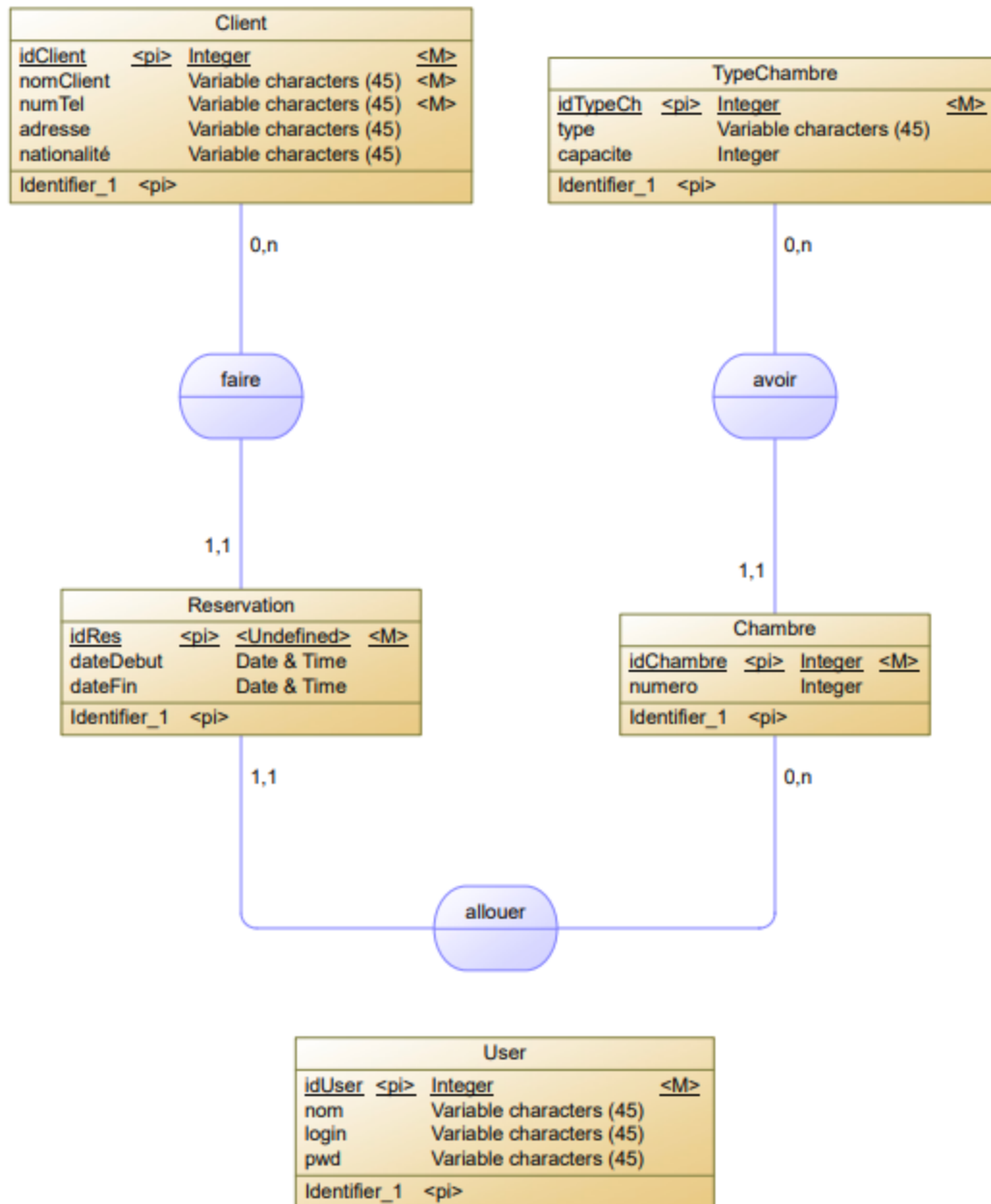
Slaoui Yassine : Backend et lien

Méthode

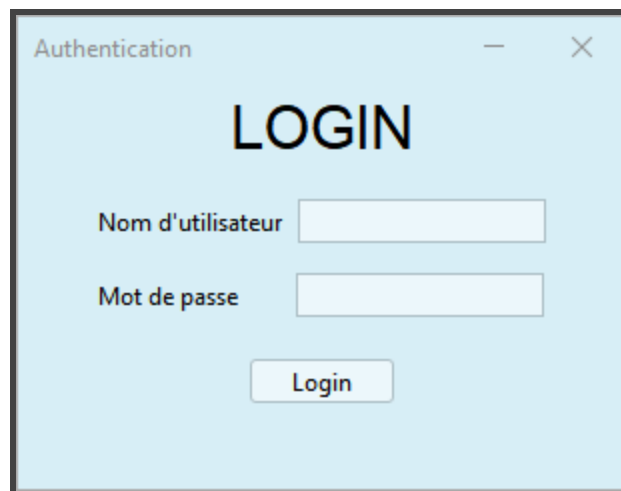
Le diagramme des Cas d'utilisation



Le Modèle Entité-Association



Les Imprimés écran



The screenshot shows a window titled "Authentication" with a light blue background. At the top center, the word "LOGIN" is displayed in large, bold, black capital letters. Below it, there are two input fields: the first is labeled "Nom d'utilisateur" and the second is labeled "Mot de passe". Both labels are in a smaller, regular black font. Below the input fields is a button labeled "Login" in a regular black font. The window has a standard title bar with a minus sign and a close button (X).

Interface d'authentification

Si aucun utilisateur n'est enregistré sur la base de données un utilisateur de départ se crée automatiquement pour pouvoir gérer l'hôtel provisoirement et/ou ajouter des utilisateurs a la base de données.



The screenshot shows a window titled "Espace de Gestion de l'Hotel" with a light blue background. At the top left, the word "Deconnexion" is visible. The main heading is "Bienvenue, Mariem" in large, bold, black capital letters. Below this, there is a section titled "BIENVENUE DANS L'ESPACE DE GESTION DE L'HOTEL!" followed by the text "Vous pouvez dans cet espace:". Below this text is a list of six items, each preceded by a hyphen: "-Gerer les Chambres", "-Gerer les Clients", "-Gerer les Reservations", "-Gerer les Types des Chambres de l'Hotel", "-Voir des statistiques sur les clients et les chambres", and "-Gerer les utilisateur de cette application". At the bottom of the window, there are six buttons labeled "Chambres", "Clients", "Reservations", "Types Chambres", "Statistiques", and "Utilisateurs". The window has a standard title bar with a minus sign and a close button (X).

Interface principale



Id	Numero	Type - Capacité	Etat Actuel (Le 14/05/2022)
1	1	Simple - 1	Disponible
2	2	Simple - 1	Disponible
3	3	Double - 2	Disponible
4	4	Double - 2	Reservée pour Maintenance, Renovation ou Autre
5	5	Triple - 3	Disponible
6	6	Triple - 3	Disponible
7	7	Quad - 4	Disponible
8	8	Quad - 4	Disponible
9	9	Suite - 7	Disponible
10	10	Suite - 7	Reservée par: Yassine, IdRes = 14

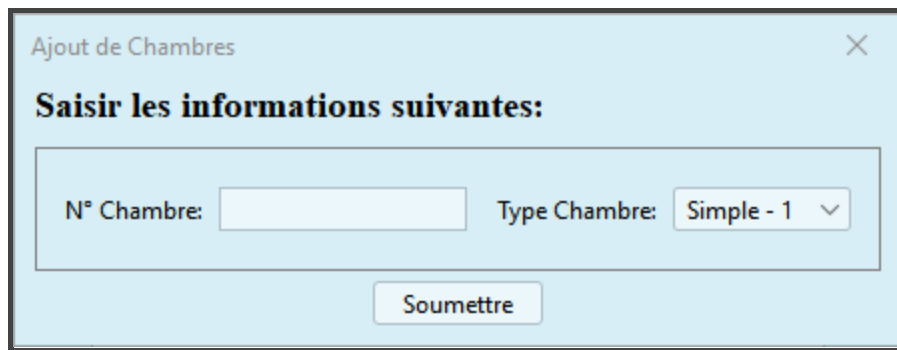
Ajouter Supprimer

Interface de gestion des chambres

On peut ici gérer les chambres, modifier les numéros, les types et voir si une chambre est réservée ou disponible ou en maintenance.

Si on modifie le numéro, un message d'erreur s'affiche si le numéro est dupliqué.

La suppression se fait après sélection des chambres à supprimer et clique sur le bouton supprimer.



Ajout de Chambres

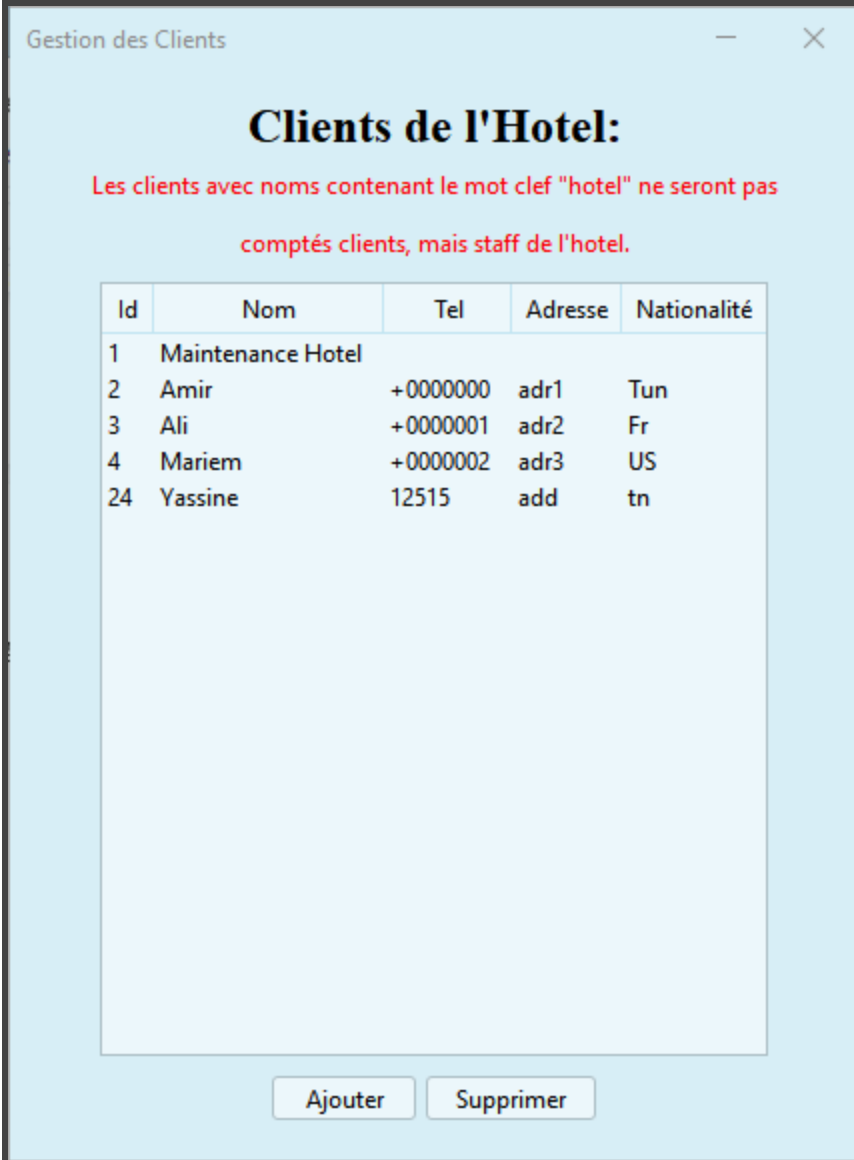
Saisir les informations suivantes:

N° Chambre: Type Chambre: Simple - 1 ▾

Soumettre

Interface d'ajout d'une chambre

Le numéro de chambre à ajouter ne doit pas exister déjà si c'est le cas un message d'erreur indiquant cela s'afficher.



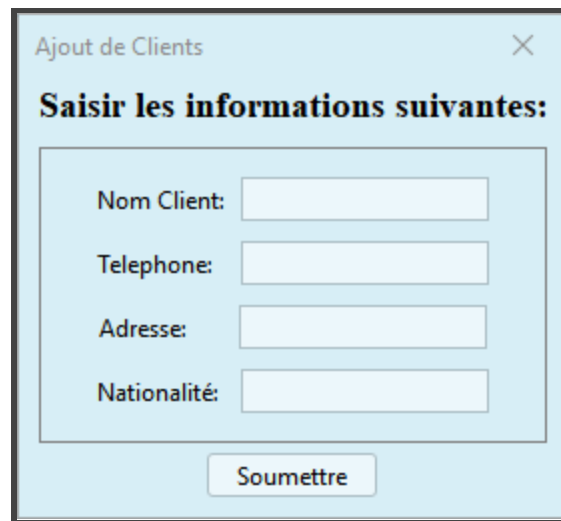
Id	Nom	Tel	Adresse	Nationalité
1	Maintenance Hotel			
2	Amir	+0000000	adr1	Tun
3	Ali	+0000001	adr2	Fr
4	Mariem	+0000002	adr3	US
24	Yassine	12515	add	tn

Interface de gestion des clients

On peut ici gérer les clients, modifier leurs données personnelles inscrites déjà, la seule contrainte est que le nom ne soit pas vide, ceci engendra un message d'erreur.

Les clients avec des noms contenant le mot clef « hôtel » seront considérés comme staff de l'hôtel et ne seront pas comptés dans les statistiques.

La suppression se fait après sélection des clients à supprimer et clique sur le bouton supprimer.



A screenshot of a web form titled "Ajout de Clients" with a close button (X) in the top right corner. Below the title is the instruction "Saisir les informations suivantes:". The form contains four input fields, each with a label to its left: "Nom Client:", "Telephone:", "Adresse:", and "Nationalité:". At the bottom right of the form is a button labeled "Soumettre".

Interface d'ajout de Clients

Ici les noms des clients à ajouter ne doivent pas être vides.

Gestion des Reservations

Reservations de l'Hotel:

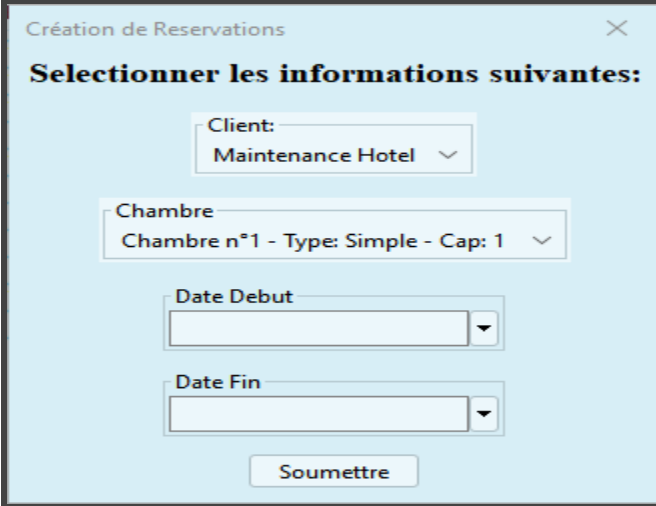
Id	Chambre - Type - Capacité	Client	Date Debut	Date Fin
1	Chambre n°4 - Type: Double - Cap: 2	Maintenance Hotel	May 13, 2022	May 17, 2022
2	Chambre n°4 - Type: Double - Cap: 2	Amir	Nov 8, 2010	Nov 12, 2010
3	Chambre n°4 - Type: Double - Cap: 2	Ali	Dec 10, 2010	Dec 12, 2010
4	Chambre n°8 - Type: Quad - Cap: 4	Mariem	Nov 15, 2010	Nov 19, 2010
5	Chambre n°4 - Type: Double - Cap: 2	Maintenance Hotel	Dec 18, 2010	Dec 31, 2010
6	Chambre n°7 - Type: Quad - Cap: 4	Ali	Sep 15, 2010	Oct 21, 2010
7	Chambre n°10 - Type: Suite - Cap: 7	Ali	Oct 25, 2010	Nov 6, 2010
8	Chambre n°9 - Type: Suite - Cap: 7	Mariem	Jul 16, 2010	Aug 23, 2010
11	Chambre n°2 - Type: Simple - Cap: 1	Amir	May 9, 2022	May 11, 2022
12	Chambre n°4 - Type: Double - Cap: 2	Ali	May 9, 2022	May 13, 2022
13	Chambre n°7 - Type: Quad - Cap: 4	Ali	May 18, 2022	May 31, 2022
14	Chambre n°10 - Type: Suite - Cap: 7	Yassine	May 10, 2022	May 18, 2022

Interface de gestion des réservations

On peut ici gérer les réservations, modifier la période de la réservation ou la chambre si possible (car la chambre peut ne pas être disponible), et on peut aussi modifier les clients qui louent la chambre à partir d'une liste de clients qui permet de choisir un client sans faire d'erreurs de nom ou id.

Si on tombe sur une période d'une chambre ou si elle n'est pas disponible un message d'erreur s'affiche et la date précédente se rétablit.

La suppression se fait après sélection des chambres à supprimer et clique sur le bouton supprimer.



Création de Reservations

Selectionner les informations suivantes:

Client:
Maintenance Hotel

Chambre
Chambre n°1 - Type: Simple - Cap: 1

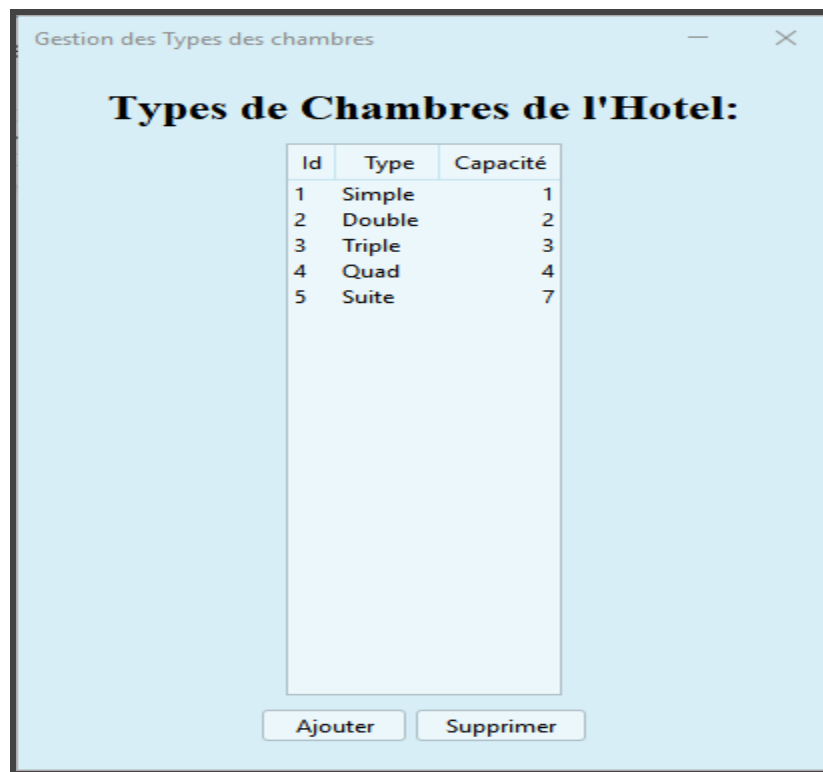
Date Debut

Date Fin

Soumettre

Interface de création de réservation

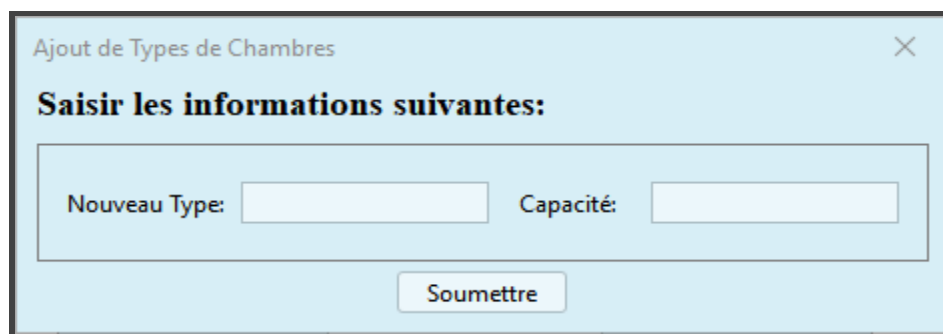
La chambre choisie doit être disponible dans l'intervalle choisi et la date de début doit être inférieure à la date de fin, sinon un message d'erreur s'affichera.



The screenshot shows a window titled "Gestion des Types des chambres". Inside, there is a section titled "Types de Chambres de l'Hotel:". Below this title is a table with three columns: "Id", "Type", and "Capacité". The table contains five rows of data. At the bottom of the window, there are two buttons: "Ajouter" and "Supprimer".

Id	Type	Capacité
1	Simple	1
2	Double	2
3	Triple	3
4	Quad	4
5	Suite	7

Interface de gestion des types de chambres



The screenshot shows a window titled "Ajout de Types de Chambres". Inside, there is a section titled "Saisir les informations suivantes:". Below this title, there are two input fields: "Nouveau Type:" and "Capacité:". At the bottom of the window, there is a button labeled "Soumettre".

Interface d'ajout de types de chambre



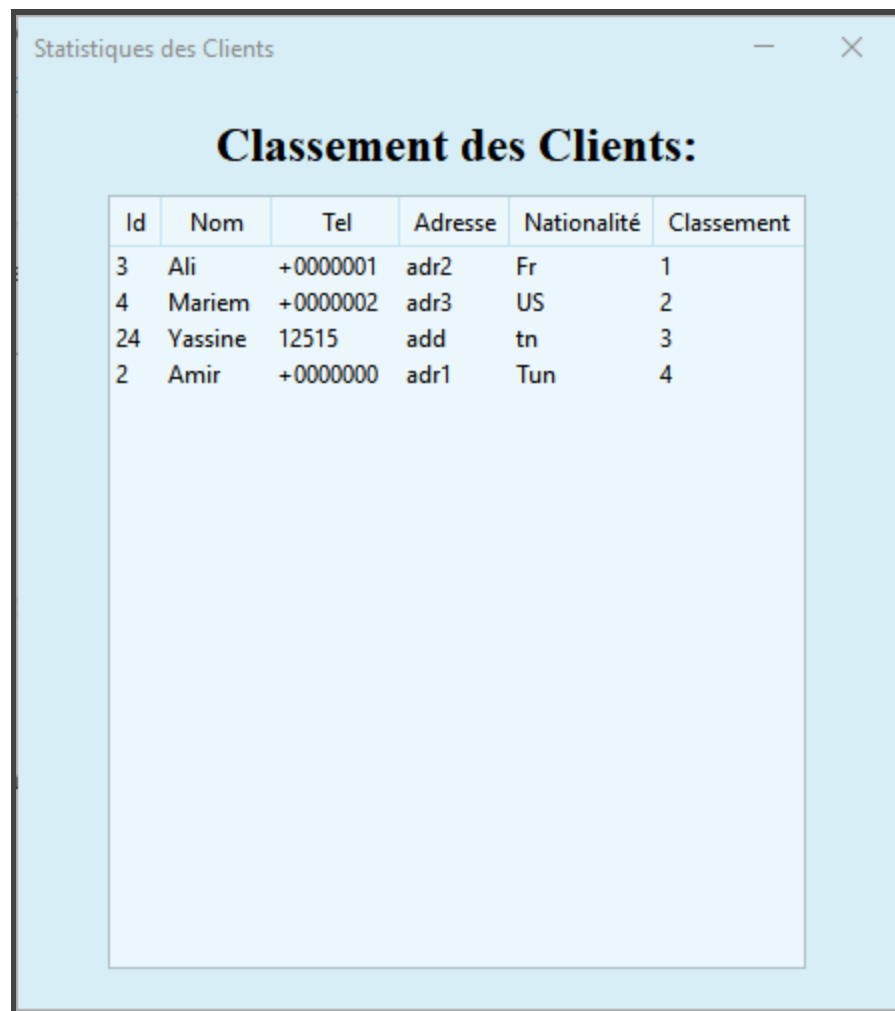
Interface des statistiques de l'hôtel

Statistiques des Chambres

Classement des Chambres:

Id	Numero	Type - Capacité	Classement
Chambre n°7 - Type: Quad - Cap: 4	7	Quad - 4	1
Chambre n°9 - Type: Suite - Cap: 7	9	Suite - 7	2
Chambre n°10 - Type: Suite - Cap: 7	10	Suite - 7	3
Chambre n°4 - Type: Double - Cap: 2	4	Double - 2	4
Chambre n°8 - Type: Quad - Cap: 4	8	Quad - 4	5
Chambre n°2 - Type: Simple - Cap: 1	2	Simple - 1	6

Interface des statistiques sur les chambres (les classements des chambres les plus demandées)



The screenshot shows a window titled "Statistiques des Clients" with a close button (X) in the top right corner. The window contains a table titled "Classement des Clients:". The table has six columns: Id, Nom, Tel, Adresse, Nationalité, and Classement. The data is as follows:

Id	Nom	Tel	Adresse	Nationalité	Classement
3	Ali	+0000001	adr2	Fr	1
4	Mariam	+0000002	adr3	US	2
24	Yassine	12515	add	tn	3
2	Amir	+0000000	adr1	Tun	4

Interface des statistiques des clients (les clients réguliers de l'hôtel)

Les Snippets de requêtes SQL

```
1 • CREATE DATABASE Hotel;
2 • USE HOTEL;
3
4 • CREATE TABLE Client(
5     idClient INT NOT NULL AUTO_INCREMENT,
6     nomClient VARCHAR(45) NOT NULL,
7     numTel VARCHAR(8),
8     adresse VARCHAR(45),
9     nationalite VARCHAR(45),
10    PRIMARY KEY (idClient));
11
12 • CREATE TABLE typeCh(
13     idType INT NOT NULL AUTO_INCREMENT,
14     typeCh VARCHAR(45) NOT NULL,
15     capacite INT,
16    PRIMARY KEY (idType));
17
18 • CREATE TABLE Chambre(
19     idChambre INT NOT NULL AUTO_INCREMENT,
20     numero INT UNIQUE NOT NULL,
21     idTypeCh INT,
22     FOREIGN KEY (idTypeCh) REFERENCES typeCh(idType),
23     PRIMARY KEY (idChambre));
24
25 • CREATE TABLE Reservation(
26     idRes INT NOT NULL AUTO_INCREMENT,
27     dateDebut DATETIME NOT NULL,
28     dateFin DATETIME NOT NULL,
29     idClient INT,
30     idChambre INT,
31     CONSTRAINT CHECK (dateDebut<dateFin),
32     FOREIGN KEY (idClient) REFERENCES Client(idClient),
33     FOREIGN KEY (idChambre) REFERENCES Chambre(idChambre),
34     PRIMARY KEY (idRes));
35
36 • CREATE TABLE User (
37     idUser INT NOT NULL AUTO_INCREMENT,
38     nom VARCHAR(45) NOT NULL,
39     login VARCHAR(45) UNIQUE NOT NULL,
40     pwd VARCHAR(45) NOT NULL,
41     PRIMARY KEY (idUsers));
```

Les Snippets de code JAVA

```
public static final String DB_URL = "jdbc:mysql://localhost:3306/hotel";  
public static final String USER = "root";  
public static final String PASS = "toor";
```

```
private void importFromDB() {  
    try (Connection connection = DriverManager.getConnection(DB_URL, USER, PASS);  
        Statement statement = connection.createStatement();) {  
        ResultSet rs = statement.executeQuery("SELECT * FROM Client");  
        while (rs.next()) {  
            clients.add(new Client(rs.getInt("idClient"), rs.getString("nomClient"),  
rs.getString("numTel"), rs.getString("adresse"), rs.getString("nationalite")));  
        }  
        rs = statement.executeQuery("SELECT * FROM TypeCh");  
        while (rs.next()) {  
            typeChambres.add(new TypeChambre(rs.getInt("idType"), rs.getString("typeCh"),  
rs.getInt("capacite")));  
        }  
        rs = statement.executeQuery("SELECT * FROM Chambre");  
        while (rs.next()) {  
            chambres.add(new Chambre(rs.getInt("idChambre"), rs.getInt("Numero"),  
findTypeChambre(rs.getInt("idTypeCh"))));  
        }  
        rs = statement.executeQuery("SELECT * FROM Reservation");  
        while (rs.next()) {  
            reservations.add(new Reservation(rs.getInt("idRes"), rs.getTimestamp("dateDebut"),  
rs.getTimestamp("dateFin"), findClient(rs.getInt("idClient")),  
findChambre(rs.getInt("idChambre"))));  
        }  
        rs = statement.executeQuery("SELECT * FROM User");  
        while (rs.next()) {  
            users.add(new User(rs.getInt(1), rs.getString(2), rs.getString(3), rs.getString(4)));  
        }  
    } catch (SQLException e) {  
        e.printStackTrace();  
    }  
}
```

```

static public int addClient(String aNom, String aTel, String aAdresse, String aNat) {
    try (Connection connection = DriverManager.getConnection(DB_URL, USER, PASS);
        PreparedStatement pStatement = connection.prepareStatement("INSERT INTO
Client(nomClient, numTel, adresse, nationalite) VALUES (?, ?, ?, ?);",
PreparedStatement.RETURN_GENERATED_KEYS)) {
        pStatement.setString(1, aNom);
        pStatement.setString(2, aTel);
        pStatement.setString(3, aAdresse);
        pStatement.setString(4, aNat);
        pStatement.executeUpdate();
        ResultSet rs = pStatement.getGeneratedKeys();
        if (rs.next()) {
            int autoId = rs.getInt(1);
            clients.add(new Client(autoId, aNom, aTel, aAdresse, aNat));
            return autoId;
        }
    } catch (Exception e) {
        return -1;
    }
    return 0;
}

```

```

static public boolean delClient(int aId) {
    try (Connection connection = DriverManager.getConnection(DB_URL, USER, PASS);
        PreparedStatement pStatement = connection.prepareStatement("DELETE FROM Client
WHERE idClient = ?;")) {
        Client cl = findClient(aId);
        for (int i = reservations.size() - 1; i >= 0; i--) {// on supprime toutes les
réservations
            Reservation res = reservations.get(i);// faites par ce client
            if (res.getClient().equals(cl))
                delReservation(res.getId());
        }
        pStatement.setInt(1, aId);
        pStatement.executeUpdate();
        clients.remove(cl);
        return true;
    } catch (Exception e) {
        return false;
    }
}

```

```
}
```

```

static public boolean updClient(int aId, String aNom, String aTel, String aAdresse, String
aNat) {
    try (Connection connection = DriverManager.getConnection(DB_URL, USER, PASS);
    PreparedStatement pStatement = connection.prepareStatement("UPDATE Client SET nomClient
= ?, numTel = ?, adresse = ?, nationalite = ? WHERE idClient = ?;")) {
        pStatement.setString(1, aNom);
        pStatement.setString(2, aTel);
        pStatement.setString(3, aAdresse);
        pStatement.setString(4, aNat);
        pStatement.setInt(5, aId);
        pStatement.executeUpdate();
        findClient(aId).update(aNom, aTel, aAdresse, aNat);
        return true;
    } catch (Exception e) {
        return false;
    }
}

```

```

try (Connection connection = DriverManager.getConnection(Launcher.DB_URL,
Launcher.USER, Launcher.PASS);
    Statement statement = connection.createStatement();) {
    ResultSet rs = statement.executeQuery("SELECT ch.idChambre, sum((r.dateFin -
r.dateDebut)) AS Periode\r\n"
        + "FROM Chambre ch, Reservation r, Client cl\r\n"
        + "WHERE ch.idChambre = r.idChambre\r\n"
        + "AND r.idClient = cl.idClient\r\n"
        + "AND UPPER(cl.nomClient) NOT LIKE '%HOTEL%'\r\n"
        + "GROUP BY ch.idChambre\r\n"
        + "ORDER BY periode DESC;");

    int d=0;
    while (rs.next()) {
        Chambre ch=Launcher.findChambre(rs.getInt(1));
        tModel.addRow(new Object[] { ch, ch.getNum(), ch.getType(), ++d});
    }
} catch (Exception e) {
    e.printStackTrace();
}

```

```
try (Connection connection = DriverManager.getConnection(Launcher.DB_URL,
Launcher.USER, Launcher.PASS);
    Statement statement = connection.createStatement();) {
    ResultSet rs = statement.executeQuery("SELECT cl.idClient, sum((r.dateFin -
r.dateDebut)) AS Periode\r\n"
        + "FROM Reservation r, Client cl\r\n"
        + "WHERE r.idClient = cl.idClient\r\n"
        + "AND UPPER(cl.nomClient) NOT LIKE '%HOTEL%'\r\n"
        + "GROUP BY cl.idClient\r\n"
        + "ORDER BY periode DESC;");

    int d=0;
    while (rs.next()) {
        Client cl=Launcher.findClient(rs.getInt(1));
        tModel.addRow(new Object[] {
cl.getId(),cl.getNom(),cl.getTel(),cl.getAdresse(),cl.getNat(), ++d});
    }
} catch (Exception e) {
    e.printStackTrace();
}
```

Divulcation et remerciements

En fin à ce projet nous remercions Allah qui nous a doté d'une grande volonté et qui nous a aidé et nous a donné le courage, la force et la patience d'accomplir ce travail.

On tient à remercier et exprimer notre profonde gratitude à notre enseignante pour les encouragements et les précieux conseils.