

# **Rapport Projet Système de Gestion de Base de Donnée (SGBD)**

*DIT 1 INFO, 2024*

Auteurs :

*Papa Ibrahima Diop*

*Khadim Ka (Write correction)*

*André Michel Ibrahima Sarr*

## **1. Introduction**

Le présent rapport expose notre projet de gestion de réservation d'hôtels et de spectacles, réalisé en collaboration au sein de notre trinôme, selon les directives fournies. Axé sur la conception d'une base de données performante, ce projet vise à offrir une solution complète et efficace pour la gestion des réservations dans le domaine hôtelier et des spectacles.

Notre approche collaborative a permis une répartition équilibrée des tâches, exploitant les compétences individuelles de chaque membre. Cette méthodologie favorise une communication constante, assurant ainsi la cohérence globale du projet.

### **1.1 Objectifs du Projet**

L'objectif principal de cette initiative est de modéliser, implémenter et interagir avec une base de données adaptée à la gestion des réservations d'hôtels et de spectacles. Nous mettons en avant notre compréhension des concepts de bases de données en concevant un schéma adéquat, intégrant des contraintes d'intégrité, et en implémentant une interface utilisateur conviviale.

### **1.2 Méthodologie Adoptée**

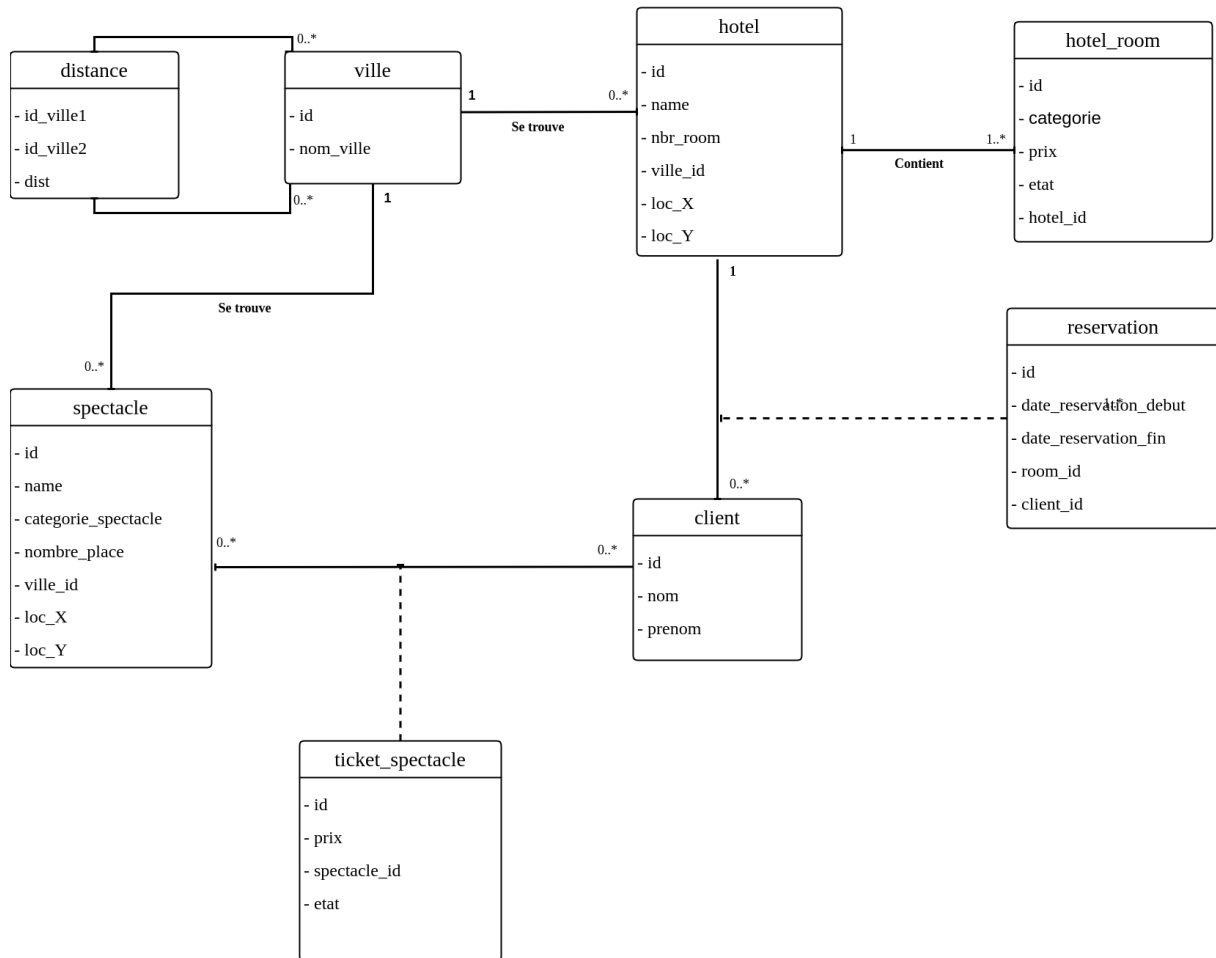
La réalisation du projet s'est effectuée dans un esprit de coopération, chaque membre du groupe contribuant de manière significative à différentes phases du processus. La répartition des tâches a été définie en fonction des compétences et des préférences de chacun, favorisant une productivité optimale. Des réunions régulières ont facilité la discussion des progrès, la résolution des problèmes et ont garanti la cohérence globale du projet.

## **2. Modélisation et Implémentation de la Base de Données (MySQL)**

### **2.1 Schéma MCD (Modèle Conceptuel de Données)**

Le modèle conceptuel de données (MCD) pour notre projet de gestion de réservation d'hôtels et de spectacles repose sur une représentation visuelle des entités clés et de leurs relations. Les principales entités incluent "client", "ville", "distance", "hotel", "hotel\_room", "reservation", "spectacle", et "ticket\_spectacle". Chaque entité est caractérisée par des attributs spécifiques, formant ainsi une représentation complète des données nécessaires à la gestion des réservations.

### Modèle Conceptuel de données :



## 2.2 Schéma MLD (Modèle Logique de Données)

Le schéma MLD de la base de données, basé sur le schéma MCD préexistant, est défini par les tables suivantes ; présenté de manière linéaire avec soulignement des clés primaires (PK) et indication des clés étrangères (FK) avec un '#' :

1. *Client* (id, nom, prenom)
2. *Ville* (id, nom\_ville )
3. *Distance* ( #id\_ville1, #id\_ville2, dist)
4. *Hotel* (id, name, nbr\_room, #ville\_id, loc\_X, loc\_Y )
5. *Hotel\_Room* (id, categorie, prix, etat, #hotel\_id )
6. *Reservation* (id, date\_reservation\_debut, date\_reservation\_fin, #room\_id, #client\_id )
7. *Spectacle* (id, name, nombre\_place, categorie\_spectacle, #ville\_id, loc\_X, loc\_Y, date\_spectacle)

8. *Ticket\_Spectacle* (id, prix, #spectacle\_id, #client\_id, etat )

Pour l'implémentation de ce modèle en base de données relationnelle, nous avons utilisé MySQL.

### **3. Implémentation de la Base de Données (MySQL)**

#### **3.1 Langage de Programmation Utilisé**

L'implémentation de la base de données a été réalisée en utilisant MySQL comme système de gestion de base de données (SGBD). Le langage SQL a été employé pour définir la structure des tables, les contraintes d'intégrité, et pour manipuler les données.

#### **3.2 Création des Tables et Insertion des Données**

La structure de la base de données a été définie en créant les tables nécessaires. Les tables telles que client, ville, distance, hotel, hotel\_room, reservation, spectacle, et ticket\_spectacle ont été créées en spécifiant les types de données des colonnes ainsi que les contraintes d'intégrité nécessaires. Les données ont ensuite été insérées à partir de fichiers CSV locaux, assurant une base de données peuplée.

#### **3.3 Contraintes d'Intégrité**

Des contraintes d'intégrité ont été mises en place pour garantir la cohérence des données dans la base de données MySQL. Ces contraintes comprennent des clés primaires et des clés étrangères qui établissent des liens entre les différentes tables (soulignées dans le modèle logique). L'utilisation de l'option ON DELETE CASCADE assure la synchronisation des modifications entre les tables liées, évitant ainsi des incohérences dans la base de données.

### **4.2 Fonctionnalités de l'Interface**

L'interface utilisateur comprend cinq pages principales, chacune correspondant aux requêtes spécifiques demandées dans le projet :

#### **1. Page de Recherche des Hôtels et Spectacles :**

- Permet aux utilisateurs de rechercher des hôtels et des spectacles dans une ville donnée.
- Filtres disponibles pour la catégorie de chambre, le type de spectacle, etc.

#### **2. Page de Consultation des Réservations :**

- Affiche les détails des réservations en fonction de critères tels que la date de début, la date de fin, et le client associé.

#### **3. Page de Modification des Réservations :**

- o Permet aux utilisateurs de mettre à jour les informations des réservations, comme les dates et les choix de chambres.

#### **4. Page de Recherche des Hôtels par Prix :**

- o Affiche l'hôtel (ou les hôtels) proposant la chambre la moins chère dans une ville à une date donnée.
5. **Page des Clients Dépensant Plus de 600 Euros :**
- o Affiche les clients ayant effectué des réservations pour un montant total d'au moins 600 Euros et qui ne dépensent jamais moins de 100 Euros par nuit d'hôtel.

### 4.3 Captures d'Écran

The screenshot shows the H\_S application interface. On the left, there is a sidebar with five buttons labeled 'Requête 1' through 'Requête 5'. 'Requête 1' is highlighted with a red border. The main area displays the text: 'La liste des hôtels qui ont des chambres disponibles dans un rayon de 5 km du lieu où se joue le spectacle XXX a la date dddddd'. Below this text, there is a form with a label 'Entrez l'id du spectacle' and a red 'Submit' button. At the top right, there is a search bar labeled 'Mysql request Syntaxe' and a magnifying glass icon. Above the main text, there are nine buttons labeled 'Extra 1' through 'Extra 9'.

The screenshot shows the H\_S application interface. On the left, there is a sidebar with five buttons labeled 'Requête 1' through 'Requête 5'. 'Requête 2' is highlighted with a red border. The main area displays the text: 'Les hôtels complets de la ville VVVV à la date dddd.'. Below this text, there is a form with two input fields: 'Entrez l'id de la ville' and 'mm/dd/yyyy' (with a calendar icon). A red 'Submit' button is to the right of the second input field. At the top right, there is a search bar labeled 'Mysql request Syntaxe' and a magnifying glass icon. Above the main text, there are nine buttons labeled 'Extra 1' through 'Extra 9'.

Requête 1

Requête 2

Requête 3

Requête 4

Requête 5

H\_S

Mysql request Syntax

Q

Extra 1

Extra 2

Extra 3

Extra 4

Extra 5

Extra 6

Extra 7

Extra 8

Extra 9

L'hôtel (les hôtels) qui a la chambre la moins chère disponible à la date dddd dans la ville VVVVVV.

Entrez l'id de la ville

mm/dd/yyyy

Submit

Requête 1

Requête 2

Requête 3

Requête 4

Requête 5

H\_S

Mysql request Syntax

Q

Extra 1

Extra 2

Extra 3

Extra 4

Extra 5

Extra 6

Extra 7

Extra 8

Extra 9

La ville proposant le séjour de trois jours le plus cher ; ici, séjour veut dire 2 nuits d'hôtel avec spectacles chaque soirée précédant les 2 nuitées, le tout dans la même ville 📍

Submit

# H\_S

Mysql request Syntax

Requête 1

Requête 2

Requête 3

Requête 4

Requête 5

Extra 1

Extra 2

Extra 3

Extra 4

Extra 5

Extra 6

Extra 7

Extra 8

Extra 9

les clients qui ont réservé pour au moins 600 Eur au total, et qui ne dépensent jamais moins de 100 Eur par nuit d'hôtel 📌

Submit

## 5. Conclusion

La réalisation de ce projet de gestion de réservations d'hôtels et de spectacles a abouti à la création d'une base de données robuste et d'une interface utilisateur basique mais intuitive. L'utilisation de MySQL pour la base de données, associée à Django, HTML et CSS pour l'interface, a permis de mettre en place un système fonctionnel répondant aux besoins spécifiques du projet.

La modélisation initiale, basée sur le modèle conceptuel de données (MCD), a été traduite avec succès en un modèle logique de données (MLD) implémenté dans MySQL. Les contraintes d'intégrité ont été soigneusement appliquées pour garantir la cohérence des données, et les requêtes SQL ont été développées pour répondre aux spécifications du projet.