

INFO0074 : Informatique pour sciences économiques et de gestion

Elise Vandomme & Eddy Heins

Description

HEC Hotel est un hôtel installé en Belgique. Il souhaite améliorer son efficacité en digitalisant ses services. Les transactions financières ne sont pas gérées dans ce projet.

En tant qu'étudiants en science économiques et de gestion à HEC Liège, vous et vos quatre amis décidez, à la fin de votre cursus universitaire, de créer votre propre société de consultance en digital. Au vu de vos compétences, vous avez remporté cet appel d'offre. Votre travail est de livrer une première version du système digital en HTML/CSS/PHP/MySQL.

On doit pouvoir se connecter en tant qu'administrateur ou en tant que client. Il faut donc proposer ce choix pour commencer afin d'accéder aux fonctionnalités ci-dessous.

Pour l'administrateur, les fonctionnalités du système sont les suivantes :

- permettre de s'authentifier (pour simplifier, le nom d'utilisateur et le mot de passe sont « admin »);
- une fois authentifié, on doit être dirigé vers la page de la liste des clients (menu « Client ») et le menu doit comporter 6 titres : « Client », « Type chambre », « Chambre », « Réservation », « Statistiques » et « Se déconnecter » ;
- le menu « Clients » doit permettre d'accéder uniquement en consultation à la liste des enregistrements de cette table. Toutes les informations de cette table de données doivent être affichées sous forme de tableau, en rassemblant l'adresse (rue, code postal et ville) en une seule colonne;
- le menu « Types de chambres » doit permettre d'accéder à la liste des enregistrements de cette table, ainsi que la possibilité d'en ajouter, modifier ou supprimer (uniquement s'il n'est pas encore affecté à une chambre). Pour l'ajout, l'id_type_chambre ne doit pas être encodé car auto-incrémenté et en cas de modification, il ne doit pas être modifiable.
- le menu « Chambres » doit permettre d'accéder à la liste des enregistrements de cette table, en affichant à côté d'id_type_chambre, les informations de toutes les autres colonnes de la table « type_chambre ». Via cette page, l'administrateur doit aussi pouvoir en ajouter ou modifier. Pour l'ajout et la modification, pour la clé étrangère (id_type_chambre), les valeurs possibles doivent provenir de la table mise en relation et apparaître sous forme de liste déroulante suffisamment complète (l'id ne suffit pas). Lorsqu'on ajoute une nouvelle chambre, on doit pouvoir choisir les valeurs de la clé étrangère via la liste déroulante et saisir le numéro de l'étage mais pas l'id_chambre car il est auto-incrémenté. Il ne vous est pas demandé d'être capable de supprimer une chambre.

- le menu « Réservations » doit permettre d'accéder en consultation uniquement à la liste des enregistrements de cette table, en affichant l'id réservation, les dates de réservation, l'id client, ainsi que le nom et prénom du client, l'id chambre, le nom du type de chambre (avec l'identifiant entre parenthèse, dans la même colonne) et le total à payer en fonction du prix par nuit par personne, du nombre de personne de la réservation et de la durée. De plus, il faut **pouvoir filtrer** cette liste sur base
 - o des clients : selon le choix fait dans le menu déroulant, le tableau doit contenir soit les réservations de tous les clients, soit les réservation d'un client en particulier;
 - o des réservations qui sont en cours à la date actuelle ou non : via une case à cocher.
- permettre d'avoir accès à un menu « Rapports » dans lequel se trouvent deux sous-menus. Le titre de chacun de ces deux sous-menus doit être explicite. Chacun doit permettre d'accéder à une page de statistique reprenant les fonctions SQL statistiques (AVG, SUM, ...), les jointures et les regroupements. Les deux statistiques ne peuvent pas être du même genre.
- permettre de se déconnecter via le menu du même nom.

Pour le client, les fonctionnalités du système sont les suivantes :

- permettre d'accéder à un menu de deux titres :
 - o « S'enregistrer » qui doit permettre de créer un nouveau compte où l'id_client ne doit pas être encodé car auto-incrémenté mais affiché une fois la création exécutée en utilisant « mysqli_insert_id(\$variable_connection) » qui permet de récupérer la dernière valeur auto-incrémentée ;
 - o « Se connecter » qui doit permettre de s'authentifier via son id_client et son mot de passe;
- une fois authentifié, on doit être redirigé vers la page « Mes réservations » et le menu doit afficher 3 titres : « Mes Réservations», « Compte » et « Se déconnecter » ;
- le menu « Mes réservations » doit permettre d'accéder à la liste des enregistrements de la table « reservation » mais uniquement celles du client connecté, ainsi que la possibilité d'en ajouter ou supprimer. Pour la liste des réservations du client connecté, il faut afficher toutes les colonnes de la table « reservation » ainsi que la capacité de la chambre et le prix total en fonction de la durée du séjour, du nombre de personnes et du prix par nuit par personne. Pour l'ajout, l'id_reservation ne doit pas être encodé car auto-incrémenté, ni l'id_client car c'est celui du client connecté. Le choix de la chambre doit se faire sur base d'une liste déroulante avec les informations utiles au choix de la chambre dont le prix. Même si ça devrait être le cas, cette liste ne doit pas être limitée aux chambres disponibles aux dates encodées.
- le menu « Compte » doit permettre de modifier les informations du compte client à l'exception de l'id_client qui est non modifiable;
- permettre de se déconnecter via le menu du même nom.

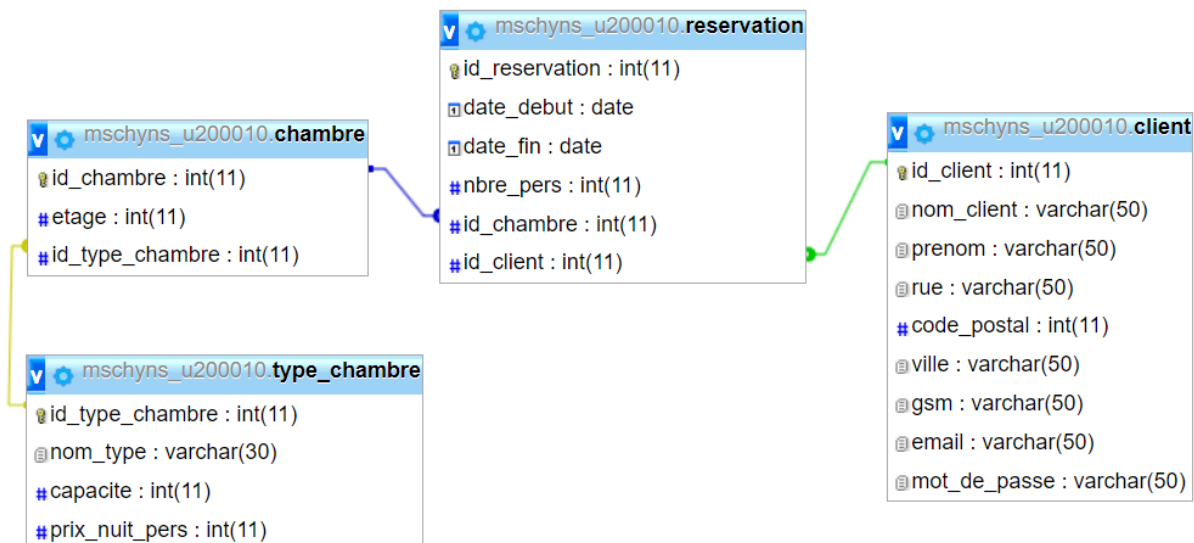
La structure de la base de données, simplifiée pour éviter trop de complexité, vous est fournie sous forme d'un fichier SQL nommé « hechotel.sql » (à importer dans votre base de données). Il contient les tables suivantes : *client*, *type_chambre*, *chambre* et *reservation*, ainsi que leurs attributs. La liste qui suit reprend les 4 tables et leurs attributs. La clé primaire (de type entier auto-incrémentée) y est soulignée et le nombre minimal d'enregistrements à encoder dans chacune des tables y est indiqué entre parenthèses :

- *client* (10 enregistrements) : id_client, nom_client, prenom, rue, code_postal (nous considérons que nous n'avons que des clients belges), ville, gsm, email et mot_de_passe;
- *type_chambre* (5 enregistrements) : id_type_chambre, nom_type, capacite, prix_nuit_pers;
- *chambre* (15 enregistrements) : id_chambre, etage, id_type_chambre;
- *reservation* (10 enregistrements) : id_reservation, date_debut, date_fin, nbre_pers, id_chambre et id_client ;

Les **contraintes** sont :

- Une *réservation* est associée à un seul *client* et un *client* peut avoir plusieurs *réservations*;
- Une *réservation* est associée à une seule *chambre* et une *chambre* peut être associée à plusieurs *réservations*;
- Une *chambre* est associée à un seul *type de chambre* et un *type de chambre* peut être associé à plusieurs *chambres*.

Voici le schéma de la base de données et il ne peut être modifié :



Délivrables

Ce projet doit être réalisé par groupe de **5 étudiants**. Les groupes portent le nom « groupeXX » où XX doit être remplacé par les deux chiffres représentant le numéro du groupe, exemple « groupe01 ».

Pour la réalisation du projet, vous devez vous servir de l'exemple « Cours-Prof » vu au cours et l'adapter. Toute autre manière de coder ne sera pas considérée et sera sanctionnée par un zéro pour le projet, car le but est de montrer que vous avez bien compris ce qui a été expliqué aux différentes séances du cours. Le dossier de départ sur le serveur doit être nommé avec le nom de votre groupe (exemple « groupe01 »). C'est ce dossier qui doit être zippé et déposé sur LOL@ avec des fichiers supplémentaires demandés ci-dessous.

Chaque fichier, à l'exception du fichier index.php, doit commencer par les deux chiffres du numéro du groupe. Par exemple : le fichier de la liste des chambres doit être nommé « 01list_chambre.php » pour le groupe 01.

Remarque pour le fichier « XXconfig.php » du projet, où XX doit être remplacé par les deux chiffres représentant le numéro du groupe :

Afin que vous travailliez par groupe sur la même base de données, choisissez le "s..." d'une personne du groupe, ainsi que son mot de passe (identique à celui de Filezilla) et écrivez dans le fichier « XXconfig.php ».

Le projet est à rendre pour le **12 mai 2025. Il s'agira d'un dossier compressé (zip/rar de maximum 20 Mo) nommé «groupeXX»** (où XX doit être remplacé par les deux chiffres représentant le numéro du groupe), **contenant :**

- Un fichier pdf contenant les matricules, noms et prénoms des membres du groupe, ainsi que l'identifiant et mot de passe pour se connecter à la base de données utilisée pour le projet (c'est-à-dire ceux utilisés dans le fichier « XXconfig.php ») ;
- L'exportation de votre base de données au format sql (**structure + enregistrements**) ;
- Les fichiers HTML, CSS et PHP.

IMPORTANT:

- **Vous devez vous servir de l'exemple « Cours-Prof » vu au cours sous peine d'être sanctionné d'un zéro dans le cas contraire. Il est interdit d'utiliser du code non vu au cours.**
- **A l'exception du fichier index.php, tous vos fichiers doivent commencer dans leur nom par le numéro du groupe sur deux chiffres (exemple : 01list_client.php pour le fichier reprenant la liste des clients pour le groupe01).**
- **Le travail doit avoir été déposé sur LOL@ par un seul membre du groupe au plus tard pour la limite du 12 mai 2025. Dans le cas contraire, la note est de 0 sur 5. Le dépôt**

sur LOL@ est accessible par tous les membres du même groupe et n'est plus modifiable une fois la date limite dépassée.

- **La note sur 5 est conservée pour la seconde session sans possibilité de l'améliorer.**