

Rapport de projet base de données

Conception d'un système de réservation de chambres d'hôtel



Réalisé par :

- Da Cruz Mathis
- Zitouni Mélissa
- Pasbeau Arthur

Encadré par :

- Marc Lemaire
- Tao yuan Jen
- Tuyet tram Dang ngoc

Table des matières

Introduction	3
Membres de l'équipe	3
Contexte du projet	3
Contraintes	3
Le projet devra se baser sur une architecture associant :	3
Principaux choix techniques	3
Schéma d'architecture	4
Base de données	5
Dictionnaire de données	5
Schéma MCD	7
Schéma MLD	7
Jeu de données	9
DDL	12
Requêtes SQL et résultats	14
Select	14
Insert	18
Delete	19
Codes PHP	19
Code Java	24

Introduction

Membres de l'équipe

- Da Cruz Mathis
- Zitouni Mélissa
- Pasbeau Arthur

Contexte du projet

Le projet consiste en la création d'un système permettant la réservation de chambres dans plusieurs hôtels appartenant à une même entreprise et sa gestion via une application web. Différents acteurs verront le jour dans ce projet et agiront sur le système tels que :

Le client : Ce dernier pourra consulter les disponibilités dans l'hôtel, ajouter une chambre à une réservation, s'inscrire / se connecter, réserver (une ou plusieurs chambres selon la réservation) ou encore choisir une ou plusieurs activités.

A savoir qu'un client ne peut réserver qu'après s'être authentifié. Les données d'identification de ce dernier seront enregistrées dans la base de données telle que son prénom, nom, date de naissance, téléphone... Cette identification permettra ainsi de sauvegarder ses données sur le site et de garder ses préférences de réservation enregistrées.

L'utilisateur pourra renseigner ses préférences de réservation (prix, nb de personne/chambre, petit-déjeuner inclus ou pas, date d'arrivée et de départ) afin de lui trouver une chambre satisfaisant ses besoins. Ce dernier aura la possibilité de réserver 1 à plusieurs chambres au sein d'un même hôtel sachant que chaque chambre est différente (type de chambre, capacité, prix).

Contraintes

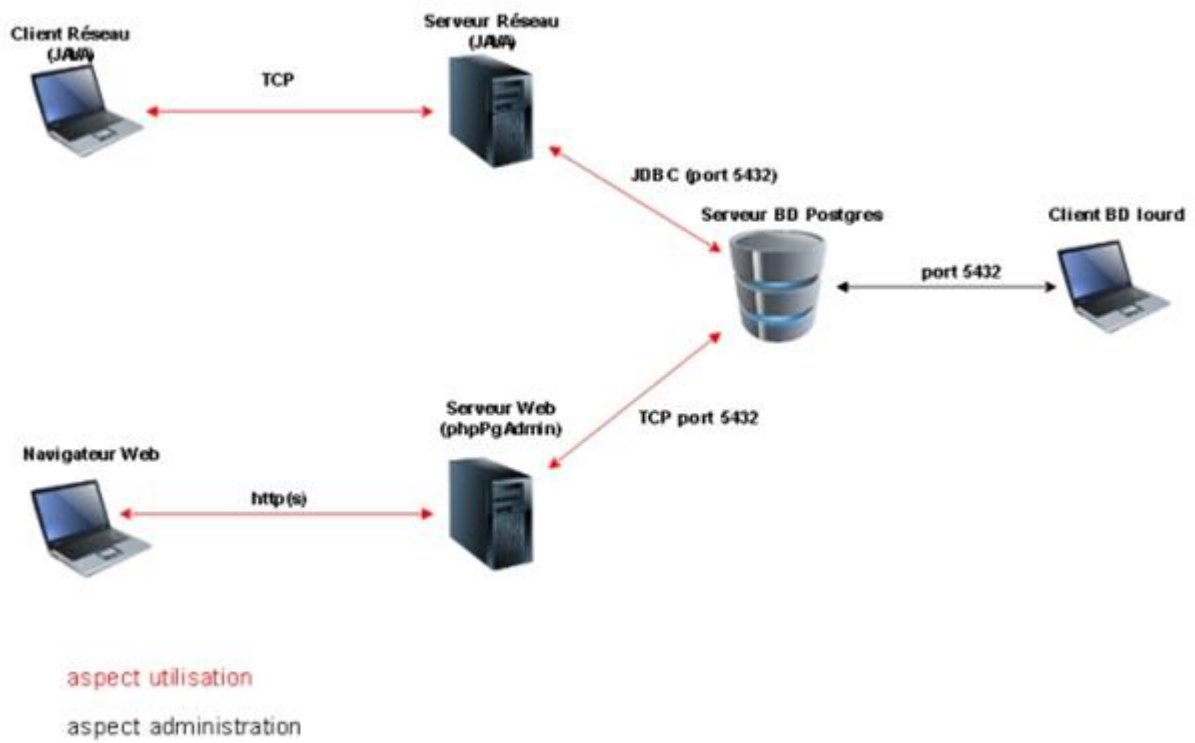
Le projet devra se baser sur une architecture associant :

- Un serveur de bases de données utilisant le SGBDR PostgreSQL
- Un serveur web avec une connexion en PHP vers le serveur de bases de données
- Un serveur réseau avec une connexion vers le serveur de base de données
- Un client réseau
- Un client web (navigateur)

Principaux choix techniques

Pour ce faire, nous privilégierons le langage java pour la programmation du client réseau ainsi que le serveur web qui établira une connexion avec la base de données via l'API JDBC. Quant au serveur web, on utilisera phpPgAdmin. Concernant la programmation réseau, TCP sera préféré à UDP car il est fiable et orienté connexion.

Schéma d'architecture



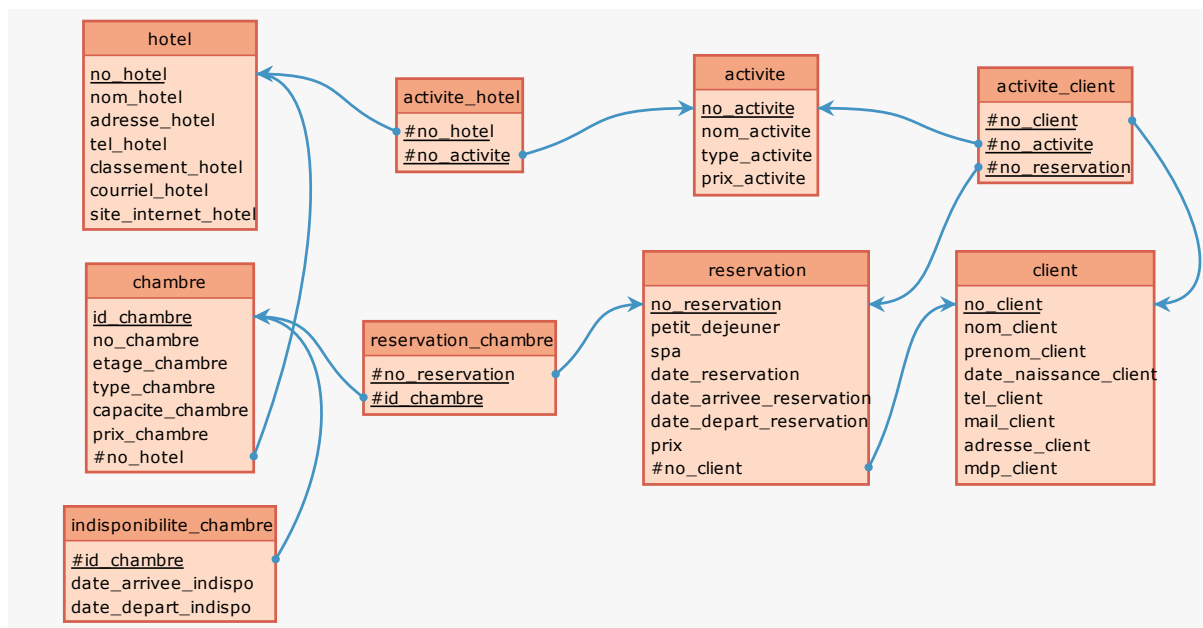
Base de données

Dictionnaire de données

Nom	Type	PK	Null/not null	Contrainte(s)	Description
no_client	INT (SERIAL)	PK	Not null		N° du client
nom_client	VARCHAR(50)	/	Not null		Nom du client
prenom_client	VARCHAR(50)	/	Not null		Prénom du client
date_naissance_client	DATE	/	Not null		Date de naissance du client
tel_client	CHAR(10)	/	Not null		N° tel du client
mail_client	VARCHAR(50)	/	Not null	LIKE ('%@%'), UNIQUE	Mail du client
adresse_client	VARCHAR(100)	/	Not null		Adresse du client
mdp_client	VARCHAR(32)	/	Not null		Mot de passe du client
no_activite	INT	PK	Not null		N° de l'activité
nom_activite	VARCHAR(20)	/	Not null		Nom de l'activité
type_activite	VARCHAR(20)	/	Not null		Type de l'activité
prix_activite	FLOAT	/	Not null		Prix de l'activité
no_hotel	INT	PK	Not null		N° de l'hôtel
nom_hotel	VARCHAR(100)	/	Not null		Nom de l'hôtel
adresse_hotel	VARCHAR(200)	/	Not null		Adresse de l'hôtel
tel_hotel	CHAR(10)	/	Not null		Numéro de tel hôtel
classement_hotel	VARCHAR(10)	/	Not null		Nombre d'étoiles
courriel_hotel	VARCHAR(100)	/	Not null		Adresse mail hôtel
site_internet_hotel	VARCHAR(200)	/	null si l'hôtel n'a pas de site		Site web hôtel

id_chambre	INT	PK	Not null		Identifiant de la chambre
no_chambre	INT	/	Not null		N° de la chambre
etage_chambre	INT	/	Not null		N° étage de la chambre
type_chambre	VARCHAR(20)	/	Not null		Type de la chambre
capacite_chambre	INT	/	Not null	<=4	Capacite d'accueil de la chambre
prix_chambre	FLOAT	/	Not null		Prix de la chambre en fonction du type et de la capacité
date_arrivee_indispo	DATE	/	Not null		Date de début de l'indisponibilité
date_depart_indispo	DATE	/	Not null		Date de fin de l'indisponibilité
no_reservation	INT (SERIAL)	PK	Not null		N° de la réservation
petit_dejeuner	BOOL	/	Not null	DEFAULT FALSE	Petit déjeuner inclus si vrai
spa	BOOL	/	Not null	DEFAULT FALSE	Spa inclus si vrai
date_reservation	DATE	/	Not null	Date réservation = date courante	Date de la réservation sur le site
date_arrivee_reservation	DATE	/	Not null	date_reservation <= date_arrivee_reservation <= date_depart_reservation	Date de début de séjour
date_depart_reservation	DATE	/	Not null	date_arrivee_reservation <= date_depart_reservation	Date de fin de séjour
prix	INT	/	Not null		Prix final réservation

Schéma MCD



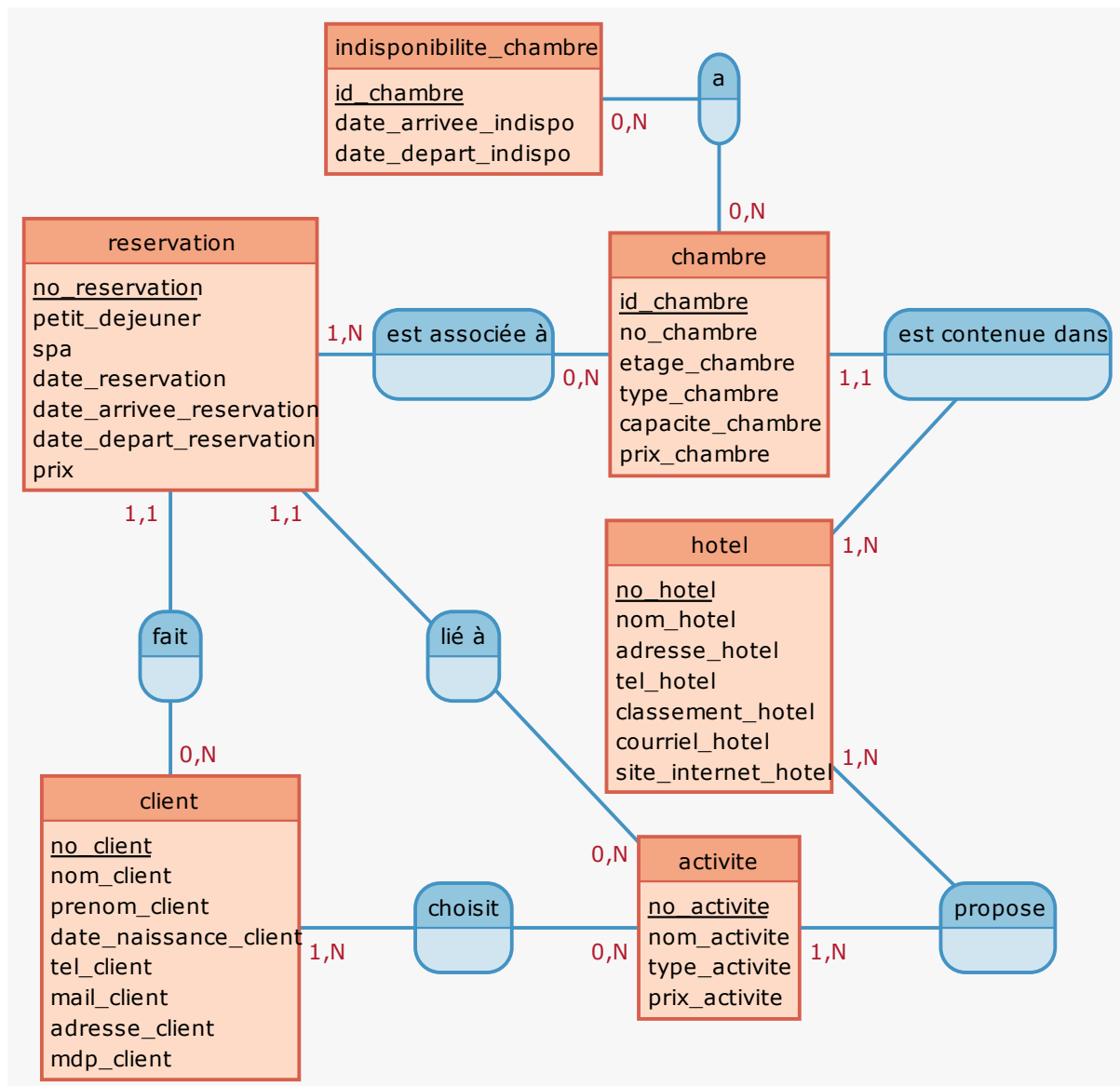
Une chambre peut être associée à plusieurs réservations à condition que l'intersection entre l'intervalle de temps de la réservation du client et les intervalles d'indisponibilité d'une chambre vaut l'ensemble vide.

Schéma MLD

```

hotel ( no_hotel, nom_hotel, adresse_hotel, tel_hotel, classement_hotel, courriel_hotel,
site_internet_hotel )
activite_hotel ( #no_hotel, #no_activite )
activite ( no_activite, nom_activite, type_activite, prix_activite )
activite_client ( #no_client, #no_activite, #no_reservation )
chambre ( id_chambre, no_chambre, etage_chambre, type_chambre, capacite_chambre,
prix_chambre, #no_hotel )
reservation_chambre ( #no_reservation, #id_chambre )
reservation ( no_reservation, petit_dejeuner, spa, date_reservation, date_arrivee_reservation,
date_depart_reservation, prix, #no_client )
client ( no_client, nom_client, prenom_client, date_naissance_client, tel_client, mail_client,
adresse_client, mdp_client )
indisponibilite_chambre ( #id_chambre, date_arrivee_indispo, date_depart_indispo )

```



Conditions

- Un hôtel propose entre 1 et N activités
- Une activité peut être proposé par 1 à N hôtels
- Une activité peut être choisi par 0 à N clients
- Une réservation est associée à un seul client et à une seule activité
- Une chambre peut être associé entre 0 et N intervalles de dates d'indisponibilité

Jeu de données

31 activités différentes sont proposées dans la table activité, avec pour chacune son type (artistique, culturelle, sportive et divers), son prix (variant entre 10 et 25).

no_activite	nom_activite	type_activite	prix_activite
0	football	sportive	17
1	danse	sportive	11
2	peinture	artistique	15
3	chant	artistique	22
4	cinema	culturelle	22
5	excursion	sportive	11
6	dessin	artistique	18

Figure 1 : Table activite_client

Lors d'une nouvelle réservation, l'activité choisie par le client est stockée dans la table activité_client

no_client	no_activite	no_reservation
2	0	3
1	0	8
1	0	9
4	0	10

Figure 2: Table activite

Chaque hôtel propose au moins une activité jusqu'à 3 maximum

no_hotel	no_activite
0	0
0	30
0	24
1	21
2	1
3	24
3	20

Figure 3: Table activite_hotel

Dans la table chambre, chaque chambre de chaque hôtel est répertoriée (environ 60 000), identifiée chacune par son ID, son no de chambre au sein de l'hôtel, l'étage à laquelle elle se situe (1-5), sa capacité d'accueil (0-4), son type (simple ou luxueuse), son prix (définis par les variables précédentes) et l'hôtel auquel elle appartient. Chaque hôtel possède entre 25 et 35 chambres.

id_chambre	no_chambre	etage_chambre	type_chambre	capacite_chambre	prix_chambre	no_hotel
0	1	1	simple	3	92	🔑0
1	2	1	simple	4	128	🔑0
2	3	1	simple	4	127	🔑0
3	4	1	simple	1	33	🔑0
4	5	1	simple	3	92	🔑0
5	6	1	simple	4	131	🔑0
6	7	2	simple	4	131	🔑0
7	8	2	luxe	4	229	🔑0
8	9	2	luxe	1	57	🔑0

Figure 4: Table chambre

Dans la table client sont répertoriés, les données fournis par l'utilisateur lors de son inscription

no_client	nom_client	prenom_client	date_naissance_client	tel_client	mail_client	adresse_client	mdp_client
1	Da Cruz	Mathis	2001-12-26	0781163276	mathisdacruz@gmail.com	39 Bis rue du docteur Vinaver, 78520, Limay	101247ae0db655b1a19313cdfdcef241
2	Lima	Catarina	1999-09-29	553456789	cat@gmail.com	Portugal, Chafé	557df39e62e16df76759670d8c695f91
3	Afatchawo	Junior	2003-01-03	0796581212	junior@gmail.com	78	90a8db953336c8dabbcf48b1592a8c06
4	Pasbeau	Arthur	2009-10-27	0695834828	jose@gmail.com	73 avenue babinks	81dc9bdb52d04dc20036dbd8313ed055
5	Araújo	Catarina	1999-09-29	937879457	catarina.l.araujo@gmail.com	Estrada Velha, Chafé	9773d3791945f97388acde09b18e4eca

Figure 5: Table client

Dans la table hôtel sont répertoriés plus de 2000 hôtels différents sur Paris et ses alentours, avec pour chacun son identifiant, son nom, adresse, no de téléphone, nombre d'étoiles, e-mail et site internet si renseigné

no_hotel	nom_hotel	adresse_hotel	tel_hotel	classement_hotel	courriel_hotel	site_internet_hotel
0	THE FIVE HÔTEL	3 RUE FLATTERS - 75005 - PARIS	143317421	3 étoiles	contact@thefivehotel.com	www.thefivehotel.com
1	HÔTEL TAMARIS	14 RUE DES MARAICHERS, 75020, PARIS	143728548	3 étoiles	tamarishotel@free.fr	www.hotel-tamaris.fr
2	HÔTEL SPLENDIDE ROYAL	18 rue du Cirque, 75008, PARIS	143871010	5 étoiles	welcome@splendideroyal.fr	www.splendideroyal.fr
3	L'OUEST HÔTEL	3 rue du Rocher, 75008, PARIS	143875749	3 étoiles	infos@ouesthotel.com	ouest-hotel-paris.com
4	HOTEL ROYAL SAINT MICHEL	3 boulevard Saint Michel, 75005, PARIS	144070606	4 étoiles	hotelroyalsaintmichel@wanadoo.fr	www.hotelroyalsaintmichel.com
5	HÔTEL DES BEAUX ARTS	2 rue Toussaint Ferron, 75013, PARIS	144242260	2 étoiles	infos@hotel-beaux-arts.fr	www.hotel-beaux-arts.fr
6	PARC HÔTEL PARIS	60 RUE BEAUNIER, 75014, PARIS	145407702	3 étoiles	parchotel-75@wanadoo.fr	www.parc-hotel-paris.com
7	HÔTEL LE RICHEMONT	17 rue Jean Coly, 75013, PARIS	145828484	3 étoiles	reservations@vacancel.com	www.vacancel.com

Figure 6: Table hotel

Lors de chaque réservation est stocké la durée d'indisponibilité de cette dernière dans la table indisponibilité_chambre

id_chambre	date_arrivee_indispo	date_depart_indispo
🔑266	2021-12-01	2021-12-08
🔑55460	2021-12-01	2021-12-02
🔑55460	2021-12-01	2021-12-09
🔑269	2021-11-30	2021-12-09

Figure 7: Table indisponibilite_chambre

Lors de chaque réservation, sont stockées les données communes de réservation de chaque hôtel dans la table reservation_chambre

no_reservation	petit_dejeuner	spa	date_reservation	date_arrivee_reservation	date_depart_reservation	prix	no_client
3	FALSE	FALSE	2021-11-30	2021-12-01	2021-12-08	227	2
8	FALSE	FALSE	2021-11-30	2021-12-01	2021-12-02	47	1
9	FALSE	FALSE	2021-11-30	2021-12-01	2021-12-09	257	1
10	FALSE	FALSE	2021-11-30	2021-11-30	2021-12-09	287	4

Figure 8: Table reservation

Chaque réservation est associée à une chambre

no_reservation	id_chambre
3	266
8	55460
9	55460
10	269

Figure 9: Table reservation_chambre

Calcul

- $\text{prix_chambre} = 30 * \text{capacité chambre} * (1 \text{ si type_chambre} = \text{simple}, 1.75 \text{ si type_chambre} = \text{luxé})$
- $\text{prix_chambre_final} = (0-10\%) * \text{prix_chambre} + \text{prix_chambre}$
- $\text{nb_jour} = \text{date_depart_reservation} - \text{date_arrivee_reservation}$
- $\text{prix_reservation} = (\text{prix_chambre_final} + (4 \text{ si petit_dejeuner})) * \text{nb_jour} + (12 \text{ si spa}) + \text{prix_activité}$

Note

L'insertion des données dans la base de données a été réalisé via l'import de fichiers CSV. Cette importation concerne les tables : hotel, activite, chambre, activite_hotel. Le reste des insertions se font via l'inscription de l'utilisateur sur le site web (client) et la réservation d'une chambre (reservation, reservation_chambre, indisponibilite_chambre, activite_client).

DDL

Création des tables de notre base de données.

```
1  CREATE TABLE client(  
2      no_client SERIAL PRIMARY KEY,  
3      nom_client VARCHAR(50) NOT NULL,  
4      prenom_client VARCHAR(50) NOT NULL,  
5      date_naissance_client DATE NOT NULL,  
6      tel_client VARCHAR(10) NOT NULL,  
7      mail_client VARCHAR(50) UNIQUE,  
8      adresse_client VARCHAR(100) NOT NULL,  
9      mdp_client VARCHAR(32) NOT NULL  
10 );  
11  
12 CREATE TABLE activite(  
13     no_activite INT PRIMARY KEY,  
14     nom_activite VARCHAR(20) NOT NULL,  
15     type_activite VARCHAR(20) NOT NULL,  
16     prix_activite FLOAT NOT NULL  
17 );  
18  
19 CREATE TABLE hotel(  
20     no_hotel INT PRIMARY KEY,  
21     nom_hotel VARCHAR(100) NOT NULL,  
22     adresse_hotel VARCHAR(200) NOT NULL,  
23     tel_hotel VARCHAR(10) NOT NULL,  
24     classement_hotel VARCHAR(10) NOT NULL,  
25     courriel_hotel VARCHAR(100) NOT NULL,  
26     site_internet_hotel VARCHAR(200) NULL  
27 );  
28
```

```
29 CREATE TABLE activite_hotel(  
30     no_hotel INT,  
31     no_activite INT,  
32  
33     CONSTRAINT fk_hotel  
34     FOREIGN KEY(no_hotel)  
35     REFERENCES hotel(no_hotel),  
36  
37     CONSTRAINT fk_activite  
38     FOREIGN KEY(no_activite)  
39     REFERENCES activite(no_activite)  
40 );  
41  
42 CREATE TABLE chambre(  
43     id_chambre INT PRIMARY KEY,  
44     no_chambre INT NOT NULL,  
45     etage_chambre INT NOT NULL,  
46     type_chambre VARCHAR(20) NOT NULL,  
47     capacite_chambre INT CHECK(capacite_chambre <= 4),  
48     prix_chambre FLOAT NOT NULL,  
49     no_hotel INT,  
50  
51     CONSTRAINT fk_hotel  
52     FOREIGN KEY(no_hotel)  
53     REFERENCES hotel(no_hotel)  
54 );  
55  
56 CREATE TABLE reservation(  
57     no_reservation SERIAL PRIMARY KEY,  
58     petit_dejeuner BOOLEAN NOT NULL DEFAULT FALSE,  
59     spa BOOLEAN NOT NULL DEFAULT FALSE,  
60     date_reservation DATE NOT NULL DEFAULT CURRENT_DATE,  
61     date_arrivee_reservation DATE NOT NULL CHECK(date_arrivee_reservation >= CURRENT_DATE),  
62     date_depart_reservation DATE NOT NULL CHECK(date_depart_reservation >= date_arrivee_reservation),  
63     prix INT NOT NULL,  
64     no_client INT,  
65  
66     CONSTRAINT fk_client  
67     FOREIGN KEY(no_client)  
68     REFERENCES client(no_client)  
69 );  
70  
71 CREATE TABLE reservation_chambre(  
72     no_reservation SERIAL,  
73     id_chambre INT,  
74  
75     CONSTRAINT fk_reservation  
76     FOREIGN KEY(no_reservation)  
77     REFERENCES reservation(no_reservation),  
78  
79     CONSTRAINT fk_chambre  
80     FOREIGN KEY(id_chambre)  
81     REFERENCES chambre(id_chambre)  
82 );  
83
```

```

84 CREATE TABLE activite_client(
85     no_client INT,
86     no_activite INT,
87     no_reservation SERIAL,
88
89     CONSTRAINT fk_client
90     FOREIGN KEY(no_client)
91     REFERENCES client(no_client),
92
93     CONSTRAINT fk_activite
94     FOREIGN KEY(no_activite)
95     REFERENCES activite(no_activite),
96
97     CONSTRAINT fk_reservation
98     FOREIGN KEY(no_reservation)
99     REFERENCES reservation(no_reservation)
100 );
101
102 CREATE TABLE indisponibilite_chambre (
103     id_chambre INT,
104     date_arrivee_indispo DATE NOT NULL,
105     date_depart_indispo DATE NOT NULL,
106
107     CONSTRAINT fk_chambre
108     FOREIGN KEY(id_chambre)
109     REFERENCES chambre(id_chambre)
110 );

```

Requêtes SQL et résultats

Select

```
1 SELECT *
2 FROM reservation
3 WHERE no_client='1';
```

















Data Output		Explain	Messages	Notifications					
	 no_reservation [PK] integer 	 petit_dejeuner boolean 	 spa boolean 	 date_reservation date 	 date_arrivee_reservation date 	 date_depart_reservation date 	 prix integer 	 no_client integer 	
1	32	false	false	2021-12-01	2021-12-02	2021-12-09	227	1	
2	33	false	false	2021-12-01	2021-12-02	2021-12-09	227	1	
3	34	false	false	2021-12-01	2021-12-15	2021-12-19	142	1	

Figure 10: Réservations d'un client

1	SELECT	nom_hotel
2	FROM	chambre
3	INNER JOIN	hotel ON chambre.no_hotel=hotel.no_hotel
4	WHERE	id_chambre='35026';



Data Output	Explain	Messages	Notifications
 nom_hotel character varying (100)			
1	HÔTEL IBIS STYLES PARIS NATION PORTE DE MONTREUIL		

Figure 11: Retourne l'hôtel grâce à l'ID d'une chambre

Compte le nombre de réservation effectué par un même client :

1	SELECT	COUNT(*)
2	FROM	reservation
3	WHERE	no_client='1';


Data Output	Explain	Messages	Notifications
 count bigint			
1	3		

Figure 12: Le nombre de réservation d'un client

1 SELECT *

2 FROM reservation

3 INNER JOIN reservation_chambre ON reservation.no_reservation=reservation_chambre.no_reservation

4 INNER JOIN chambre ON reservation_chambre.id_chambre=chambre.id_chambre

5 INNER JOIN hotel ON hotel.no_hotel=chambre.no_hotel












6 WHERE no_client='1';










Data Output

Explain

Messages

Notifications

 no_reservation integer	 petit_dejeuner boolean	 spa boolean	 date_reservation date	 date_arrivee_reservation date	 date_depart_reservation date	 prix integer	 no_client integer	 no_reservation integer	 id_chambre integer	
1	32	false	false	2021-12-01	2021-12-02	2021-12-09	227	1	32	266
2	33	false	false	2021-12-01	2021-12-02	2021-12-09	227	1	33	55460
3	34	false	false	2021-12-01	2021-12-15	2021-12-19	142	1	34	8851

 no_chambre integer	 etage_chambre integer	 type_chambre character varying (20)	 capacite_chambre integer	 prix_chambre double precision	 no_hotel integer	 no_hotel integer	 nom_hotel character varying (100)	 adresse_hotel character varying (200)	
21	2	simple		1	30	8	8	AMBIANCE HÔTEL	8 rue André Citroën, 94510, LA QUEUE-EN-BOIS
22	1	simple		1	30	1853	1853	HÔTEL MONTSOURIS ORLÉANS	15 rue Beaunier, 75014, PARIS
3	1	simple		1	30	296	296	HÔTEL JARRY	4 rue Jarry, 75010, PARIS







 nom_hotel character varying (100)	 adresse_hotel character varying (200)	 tel_hotel character varying (10)	 classement_hotel character varying (10)	 courriel_hotel character varying (100)	 site_internet_hotel character varying (200)
AMBIANCE HÔTEL	8 rue André Citroën, 94510, LA QUEUE-EN-BRIE	145940808	1 étoile	ambiancehotel@orange.fr	ambiance-hotel.net
HÔTEL MONTSOURIS ORLÉANS	15 rue Beaunier, 75014, PARIS	145405013	1 étoile	hotel.montsouris.orlean@club-internet.fr	[null]
HÔTEL JARRY	4 rue Jarry, 75010, PARIS	147707038	1 étoile	hoteljarry@gmail.com	www.hoteljarry.com

Figure 13: Affiche toutes les données relatives à la réservation d'une chambre d'un hôtel

Affiche les chambres indisponibles sur une durée donnée, ces chambres-là ne seront pas visibles lors des recherches des utilisateurs :

```

1 SELECT id_chambre
2 FROM indisponibilite_chambre
3 WHERE date_arrivee_indispo BETWEEN '02-12-2021' AND '09-12-2021'
4 OR date_depart_indispo BETWEEN '02-12-2021' AND '09-12-2021';

```

Data Output		Explain	Messages	Notifications
id_chambre	integer			
1	266			
2	55460			

Figure 14: Retourne l'ensemble des chambres indisponibles sur un intervalle donné


```

1 SELECT nom_hotel,prix_chambre,id_chambre,no_chambre,etage_chambre
2 FROM activite_hotel
3 INNER JOIN hotel ON activite_hotel.no_hotel=hotel.no_hotel
4 INNER JOIN activite ON activite_hotel.no_activite=activite.no_activite
5 INNER JOIN chambre ON activite_hotel.no_hotel=chambre.no_hotel
6 WHERE activite.nom_activite='football' AND type_chambre='luxe'
7 AND capacite_chambre='3' AND classement_hotel='3 étoiles'
8 GROUP BY nom_hotel,prix_chambre,id_chambre,no_chambre,etage_chambre
9 ORDER BY prix_chambre ASC;

```

	nom_hotel character varying (100)	prix_chambre double precision	id_chambre integer	no_chambre integer	etage_chambre integer
1	HÔTEL VANEAU SAINT-GERMAIN	158	5809	2	2
2	HÔTEL DAUMESNIL VINCENNES	158	30070	9	5
3	HÔTEL DES TUILERIES	160	22817	24	3
4	HÔTEL BEL AIR	161	26493	6	2
5	HÔTEL CARINA	161	34437	22	3
6	HÔTEL RESTAURANT KYRIAD TORCY	163	39021	14	2
7	HÔTEL IBIS BUDGET PARIS PORTE D'ITALIE EST	163	42881	29	4

Figure 15: Classe par prix ascendant le résultat de la recherche

```

1 select *
2 from client
3 where mail_client = 'mathisdaccruz@gmail.com'
4 and mdp_client = '101247ae0db655b1a19313cdfdcef241';

```

no_client [PK] integer	nom_client character varying (50)	prenom_client character varying (50)	date_naissance_client date	tel_client character varying (10)	mail_client character varying (50)	adresse_client character varying (100)	mdp_client character varying (50)
1	Da Cruz	Mathis	2001-12-26	0781163276	mathisdaccruz@gmail.com	39 Bis rue du Docteur Vinaver, 78520, Limay	101247aef

Figure 16: Vérifie les informations de connexions

Insert

```

1 insert into client(nom_client,prenom_client,date_naissance_client,tel_client,
2 mail_client,adresse_client,mdp_client)
3 values('lefebvre','lillian','1999-02-02','0132789564','lialian@gmail.com','Paris',
4 '9773d3791945f97388acde09b18e4eca');

```

Data Output Explain Messages Notifications

INSERT 0 1

Query returned successfully in 86 msec.

Figure 17: Inscription utilisateur

Afin de procéder à la réservation d'une chambre nous devons ajouter la ligne correspondant à :

1. L'activité qu'avait choisie le client pour son séjour dans la table activite_client
2. La liaison entre la réservation et la chambre dans reservation_chambre
3. La réservation dans reservation
4. Les dates d'indisponibilités de la chambre qui correspondent à la durée du séjour dans indisponibilite_chambre afin d'éviter que quelqu'un d'autre la réserve sur la même période

```

1 insert into reservation(petit_dejeuner,spa,date_reservation,date_arrivee_reservation,
2 date_depart_reservation,prix,no_client)
3 values('true','true','2021-12-01','2021-12-15',
4 '2021-12-17','67','1');
5 insert into reservation_chambre(id_chambre)
6 values('24525');
7 insert into activite_client(no_client,no_activite)
8 values('1','0');
9 insert into indisponibilite_chambre(id_chambre,date_arrivee_indispo,date_depart_indispo)
10 values('24525','2021-12-15','2021-12-17');

```

Data Output Explain Messages Notifications

INSERT 0 1

Query returned successfully in 93 msec.

Figure 18: Nouvelle reservation

Delete

Afin de procéder à l'annulation d'une réservation nous devons supprimer la ligne correspondant à :

1. L'activité qu'avait choisie le client pour son séjour dans la table `activite_client`
2. La liaison entre la réservation et la chambre dans `reservation_chambre`
3. La réservation dans `reservation`
4. Les dates d'indisponibilités de la chambre qui correspondaient à la durée du séjour dans `indisponibilite_chambre` afin de permettre que quelqu'un d'autre la réserve sur la même période

```

1 DELETE
2 FROM activite_client
3 WHERE no_reservation='32';
4 DELETE
5 FROM reservation_chambre
6 WHERE no_reservation='32';
7 DELETE
8 FROM reservation
9 WHERE no_reservation='32';
10 DELETE
11 FROM indisponibilite_chambre
12 WHERE id_chambre='266' AND date_arrivee_indispo='2021-12-02'
13 AND date_depart_indispo='2021-12-09';

```

Data Output Explain Messages Notifications

DELETE 1

Query returned successfully in 94 msec.

Figure 19: Annulation réservation

Codes PHP

Afin que le site internet puisse être lié à la base de données, nous avons ouvert un compte phpPgAdmin lié à Alwaysdata. En revanche, en localhost nous avons privilégié la solution PostgreSQL.

```

<?php
$host = "postgresql-dacruzmathis.alwaysdata.net";
$port = "5432";
$dbname = "dacruzmathis_mydatabase";
$user = "dacruzmathis";
$password = " ";
$connection_string = "host={$host} port={$port} dbname={$dbname} user={$user} password={$password} ";
$dbconn = pg_connect($connection_string);

```

Figure 20: Connexion Postgres

Lors de l'inscription, si les données de l'utilisateur ont pu être insérées dans la table client alors on autorise l'inscription (Inscription validée) et on est directement redirigé sur la page utilisateur. Le mot de passe entré par l'utilisateur est haché lors de l'insertion dans la table client via la fonction `md5`.

```

<h2 id='insc'>Inscrivez-vous ici</h2>
<?php
if(isset($_POST['submit'])&&!empty($_POST['submit'])) {
    if(isExistingMail($_POST['email'])) {
        echo "<p>Mail déjà existant !</p>";
    }
    else {
        $sql = "insert into client(nom_client, prenom_client, date_naissance_client, tel_client, mail_client, adresse_client, mdp_client)
        values('".$_POST['lastname']."','".$_POST['firstname']."','".$_POST['birth']."',
        ['".$_POST['mobno']."','".$_POST['email']."','".$_POST['adress']."','".$_POST['pwd']."')";
        $ret = pg_query($dbconn, $sql);
        if($ret) {
            echo "<p>Inscription validé !</p><a href='login.php'>Me connecter</a>";
            $_SESSION["mail"] = $_POST['email'];
            $_SESSION["is_connected"] = 1;
            header("Location: user.php");
        } else {
            echo "<p>Inscription non validé !</p>";
        }
    }
}
?>

```

Figure 21: Inscription

Lors de la connexion, si l'utilisateur entre un email et un mot de passe qui sont retrouvables dans la table client alors on autorise la connexion :

```

<h2 id='co'>Connectez-vous ici</h2>
<?php
if(isset($_POST['submit'])&&!empty($_POST['submit'])) {
    $hashpassword = md5($_POST['pwd']);
    $sql = "select * from client where mail_client = '".$_pg_escape_string($_POST['email'])."' and mdp_client = '".$hashpassword."'";
    $data = pg_query($dbconn, $sql);
    $login_check = pg_num_rows($data);
    if($login_check > 0) {
        echo "<p>Connecté avec succès !</p>";
        $_SESSION["mail"] = $_POST['email'];
        $_SESSION["is_connected"] = 1;
        header("Location: user.php");
    } else {
        echo "<p>Email ou mot de passe incorrect !</p>";
        $_SESSION["is_connected"] = 0;
    }
}
?>

```

Figure 22: Connexion

Lorsqu'on se déconnecte, la session est détruite et on est redirigé sur la page d'accueil

```

<?php
session_start();
session_destroy();
header("Location: index.php");
?>

```

Figure 23: Déconnexion

Lors d'une réservation, les critères choisis par l'utilisateur sont préservés dans l'élément `$_SESSION['critère']`. En effet, les éléments de filtrage sont saisis dans la page `user.php` puis le résultat est affiché dans la page `test.php` et la confirmation est affichée dans `user.php`. De plus, l'élément `SESSION['is_done']` permet d'empêcher le renvoi du formulaire lorsque la page est rafraîchie et donc d'éviter de se retrouver avec des réservations doublons. Enfin, le formulaire vérifie bien que tous les éléments sont présents afin d'envoyer le formulaire, et vérifie bien qu'il n'y a aucun conflit au niveau des dates (`dates_reservation <= date_arrivee <= date_depart`).

```

<?php
if(isset($_SESSION["is_done"])&&$_SESSION["is_done"]==1){
    $_SESSION["is_done"]=0;
    header('Location: user.php');
    exit();
}
if(isset($_SESSION["new_reserv"])&&$_SESSION["new_reserv"]==1){
    $_SESSION["new_reserv"] = 0;
}
if(isset($_POST["prix_final"])&&isset($_POST["id_chambre"])){
    $_SESSION["prix_final"] = $_POST['prix_final'];
    $_SESSION["id_chambre"] = $_POST['id_chambre'];
    $_SESSION["nom_hotel"] = getHotelById($_SESSION["id_chambre"]);
    $_SESSION["new_reserv"] = 1;
}
if(isset($_POST['arrivee'])&&isset($_POST['depart']))){
    $today = date("Y-m-d");
    if(strtotime($today)<=strtotime($_POST['arrivee'])&&strtotime($_POST['arrivee'])<=strtotime($_POST['depart'])){
        $_SESSION['spa']=$_POST['spa'];
        $_SESSION['dejeuner']=$_POST['dejeuner'];
        $_SESSION['arrivee']=$_POST['arrivee'];
        $_SESSION['depart']=$_POST['depart'];
        $_SESSION['today']=$today;
        $_SESSION['activite']=$_POST['activite'];
        $_SESSION['type_chambre']=$_POST['type_chambre'];
        $_SESSION['capacite']=$_POST['capacite'];
        $_SESSION['etoiles']=$_POST['etoiles'];
        $_SESSION['conflit_date']=0;
        header("Location: test.php");
    }
    else{
        $_SESSION['conflit_date']=1;
        header("Location: user.php#form");
    }
}
}
?>

```

Figure 24: Page utilisateur

Lorsque l'utilisateur valide sa réservation, les données relatives cette dernière vont devoir être insérée dans les tables liées à la réservation (réservation, reservation_chambre, activite_client, indisponibilite_chambre).

```

<h2 id='mine'>Mes reservation</h2>
<?php
if(isset($_SESSION["new_reserv"])&&$_SESSION["new_reserv"]==1){
    $today = date("Y-m-d");
    $_SESSION['today']=$today;

    $sql = "insert into reservation(petit_dejeuner,spa,date_reservation,date_arrivee_reservation,
    date_depart_reservation,prix,no_client)values('".$_SESSION["dejeuner"]."','".$_SESSION["spa"]."','".$_SESSION["today"]."',
    '".$_SESSION["arrivee"]."','".$_SESSION["depart"]."','".$_SESSION["prix_final"]."','".$_SESSION["nom_hotel"]."','".$_SESSION["no_client"]."')";
    $res = pg_query($dbconn, $sql);

    $sql2 = "insert into reservation_chambre(id_chambre)values('".$_SESSION["id_chambre"]."')";
    $res2 = pg_query($dbconn, $sql2);

    $sql3 = "insert into activite_client(no_client,no_activite)values('".$_SESSION["no_client"]."','".$_SESSION["no_activite"]."')";
    $res3 = pg_query($dbconn, $sql3);

    $sql4 = "insert into indisponibilite_chambre(id_chambre,date_arrivee_indispo,date_depart_indispo)values
    ('".$_SESSION["id_chambre"]."','".$_SESSION["arrivee"]."','".$_SESSION["depart"]."')";
    $res4 = pg_query($dbconn, $sql4);

    if($res&&$res2&&$res3&&$res4){
        echo "<p>Reservation validé !</p>";
        $_SESSION['is_done']=1;
    }
    else{
        echo "<p>Reservation non validé !</p>";
    }
}

```

Figure 25: Reservation

Affichage dans test.php du résultat de la recherche du client :

```

if(isset($_GET['prix'])){
    $array = minMax($activite,$type_chambre,$capacite,$etoiles,$_GET['prix']);
}
else if(isset($_GET['etage'])){
    $array = numEtage($activite,$type_chambre,$capacite,$etoiles,$_GET['etage']);
}
else{
    $query = "
        SELECT nom_hotel,prix_chambre,id_chambre,no_chambre,etage_chambre
        FROM activite_hotel
        INNER JOIN hotel ON activite_hotel.no_hotel=hotel.no_hotel
        INNER JOIN activite ON activite_hotel.no_activite=activite.no_activite
        INNER JOIN chambre ON activite_hotel.no_hotel=chambre.no_hotel
        WHERE activite.nom_activite='".$_.$activite.'" AND type_chambre='".$_.$type_chambre.'"
        AND capacite_chambre='".$_.$capacite.'" AND classement_hotel='".$_.$etoiles.'"
        GROUP BY nom_hotel,prix_chambre,id_chambre,no_chambre,etage_chambre
        ORDER BY prix_chambre ASC;
    ";

    $res = pg_query($dbconn, $query);
    $array = pg_fetch_all($res);
}

```

Figure 26: Affichage hotels

Lors de l'annulation d'une réservation, ce sont les données des 4 tables relatives a cette dernière qui doivent t'être supprimées.

```

function deleteRes($no_reservation,$id_chambre,$datearrivee_indispo,$datedepart_indispo){
    include 'include/dbconn.inc.php';
    $query = "DELETE
        FROM activite_client
        WHERE no_reservation='".$_.$no_reservation.'"';
    $res = pg_query($dbconn, $query);
    $query2 = "DELETE
        FROM reservation_chambre
        WHERE no_reservation='".$_.$no_reservation.'"';
    $res2 = pg_query($dbconn, $query2);
    $query3 = "DELETE
        FROM reservation
        WHERE no_reservation='".$_.$no_reservation.'"';
    $res3 = pg_query($dbconn, $query3);
    $query4 = "DELETE
        FROM indisponibilite_chambre
        WHERE id_chambre='".$_.$id_chambre.'" AND datearrivee_indispo='".$_.$datearrivee_indispo.'"
        AND datedepart_indispo='".$_.$datedepart_indispo.'"';
    $res4 = pg_query($dbconn, $query4);
    return $res&&$res2&&$res3&&$res4;
}

```

Figure 27: Suppression Réservation

Filtrage de l'affichage du résultat de la recherche de l'utilisateur en fonction du prix et via ORDER BY, la variable \$var vaut soit ASC (ascendant) soit DESC (descendant) :

```
function minMax($active,$type_chambre,$capacite,$etoiles,$var){
    include 'include/dbconn.inc.php';
    $query = "SELECT nom_hotel,prix_chambre,id_chambre,no_chambre,etage_chambre
    FROM active_hotel
    INNER JOIN hotel ON active_hotel.no_hotel=hotel.no_hotel
    INNER JOIN active ON active_hotel.no_active=active.no_active
    INNER JOIN chambre ON active_hotel.no_hotel=chambre.no_hotel
    WHERE active.nom_active='".$active."' AND type_chambre='".$type_chambre."' AND capacite_chambre='".$capacite."'
    AND classement_hotel='".$etoiles."'
    GROUP BY nom_hotel,prix_chambre,id_chambre,no_chambre,etage_chambre
    ORDER BY prix_chambre ".$var.";";
    $res = pg_query($dbconn, $query);
    $array = pg_fetch_all($res);
    return $array;
}
```

Figure 28: Filtrage par prix

Affiche toute les réservations effectuées par le client (user.php/mes_reservations) :

```
function getInforRes($no_client){
    include 'include/dbconn.inc.php';
    $query = "SELECT *
    FROM reservation
    INNER JOIN reservation_chambre ON reservation.no_reservation=reservation_chambre.no_reservation
    INNER JOIN chambre ON reservation_chambre.id_chambre=chambre.id_chambre
    INNER JOIN hotel ON hotel.no_hotel=chambre.no_hotel
    WHERE no_client='".$no_client."'";
    $res = pg_query($dbconn, $query);
    $array = pg_fetch_all($res);
    return $array;
}
```

Figure 29: Réservation d'un client X

Calcul du prix final de la chambre :

```
$prix=$array[$i]['prix_chambre'];
if($_SESSION['déjeuner']==1){
    $prix+=4;
}
$_SESSION['day']=nbDay($_SESSION['arrivee'], $_SESSION['depart']);
$prix_final=$prix;
$prix_final*=$_SESSION['day'];
$prix_final+=getPrixActive($_SESSION['active']);
if($_SESSION['spa']==1){
    $prix_final+=12;
}
```

Figure 30: Prix final chambre

Code Java

La connexion entre le serveur Java et la base de données PostgreSQL :

```
try {  
    // creation de la connexion à la BDD  
    DriverManager.setLoginTimeout(1);  
    connex = DriverManager.getConnection("jdbc:postgresql://localhost:5432/reservation_hotel", "postgres",  
        "mot de passe");  
} catch (SQLException e1) {  
    e1.printStackTrace();  
    System.err.println("connexion timed out");  
}
```

Figure 31: Connexion Java PostgreSQL