

Rapport de présentation

Fatimatou Zahra TALL

Mame Fatou WADE

Faty SOW

Adji Fatou Diop

Mareme Ndiaye

L'objectif de ce rapport est de présenter les résultats de cette analyse et proposer des recommandations visant à améliorer la gestion de l'hôtel. Dans ce rapport,nous allons examiner les différentes facettes de la gestion hôtelière,notamment la gestion des chambres,la gestion de réservations et la gestion des factures.

LA BASE DE DONNÉES

Une base de données MySQL réparti sur 3 nœuds en utilisant ProxySQL

- Téléchargement de MySQL

```
root@fatysow-VirtualBox:~# apt-cache policy mysql-server
mysql-server:
  Installé : 5.7.41-0ubuntu0.18.04.1
  Candidat : 5.7.41-0ubuntu0.18.04.1
  Table de version :
*** 5.7.41-0ubuntu0.18.04.1 500
      500 http://sn.archive.ubuntu.com/ubuntu bionic-updates/main amd64 Packages
      500 http://sn.archive.ubuntu.com/ubuntu bionic-updates/main i386 Packages
      500 http://security.ubuntu.com/ubuntu bionic-security/main amd64 Packages
      500 http://security.ubuntu.com/ubuntu bionic-security/main i386 Packages
      100 /var/lib/dpkg/status
  5.7.21-1ubuntu1 500
      500 http://sn.archive.ubuntu.com/ubuntu bionic/main amd64 Packages
      500 http://sn.archive.ubuntu.com/ubuntu bionic/main i386 Packages
root@fatysow-VirtualBox:~#
```

-Configuration de mysql

on édite le fichier *etc/mysql/mysql.conf.d/mysql.cnf*

```
GNU nano 2.9.3          /etc/mysql/mysql.conf.d/mysqld.cnf          Modifié

[mysqld_safe]
socket      = /var/run/mysqld/mysqld.sock
nice        = 0

[mysqld]
#
# * Basic Settings
#
user        = mysql
pid-file    = /var/run/mysqld/mysqld.pid
socket      = /var/run/mysqld/mysqld.sock
port        = 3306
basedir     = /usr
datadir     = /var/lib/mysql
tmpdir      = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 0.0.0.0
#
```

```
GNU nano 2.9.3          /etc/mysql/mysql.conf.d/mysqld.cnf          Modifié

[mysqld_safe]
socket      = /var/run/mysqld/mysqld.sock
nice        = 0

[mysqld]
#
# * Basic Settings
#
user        = mysql
pid-file    = /var/run/mysqld/mysqld.pid
socket      = /var/run/mysqld/mysqld.sock
port        = 3306
basedir     = /usr
datadir     = /var/lib/mysql
tmpdir      = /tmp
lc-messages-dir = /usr/share/mysql
skip-external-locking
#
# Instead of skip-networking the default is now to listen only on
# localhost which is more compatible and is not less secure.
bind-address        = 0.0.0.0
#
```

```
root@fatysow-VirtualBox:~# systemctl status mysql
● mysql.service - MySQL Community Server
  Loaded: loaded (/lib/systemd/system/mysql.service; enabled; vendor preset: e
  Active: active (running) since Mon 2023-03-20 17:59:03 GMT; 7s ago
    Process: 13240 ExecStart=/usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/
    Process: 13231 ExecStartPre=/usr/share/mysql/mysql-systemd-start pre (code=ex
  Main PID: 13242 (mysqld)
     Tasks: 29 (limit: 2318)
    CGroup: /system.slice/mysql.service
            └─13242 /usr/sbin/mysqld --daemonize --pid-file=/run/mysqld/mysqld.p

mar 20 17:58:59 fatysow-VirtualBox systemd[1]: Starting MySQL Community Server.
mar 20 17:59:03 fatysow-VirtualBox systemd[1]: Started MySQL Community Server.
lines 1-12/12 (END)
```

puis on se connecte sur mysql pour créer la base de donner Hotelerie les tables

```
Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE DATABASE Hotelerie;
ERROR 1007 (HY000): Can't create database 'Hotelerie'; database exists
mysql> use Hotelerie;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

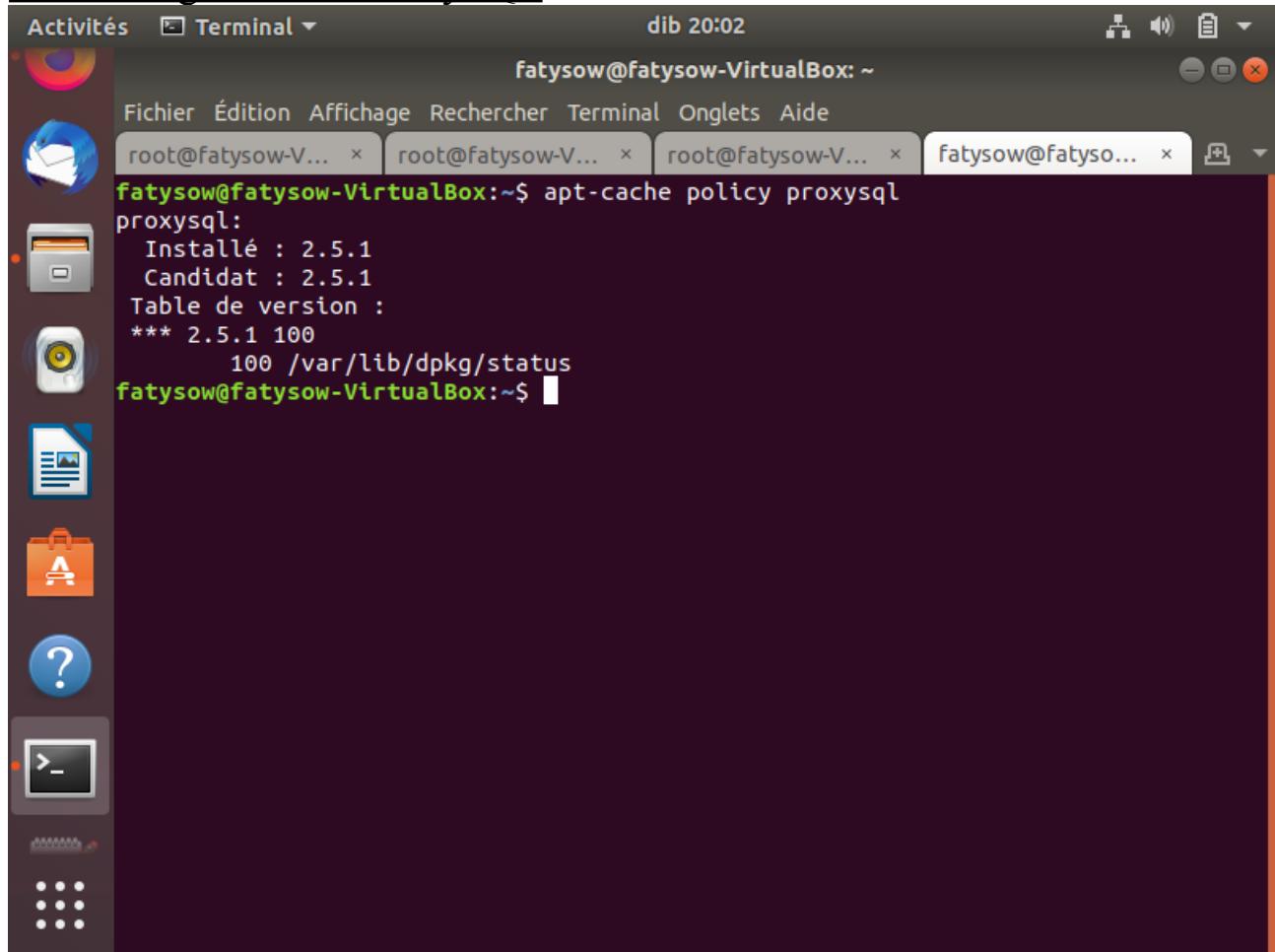
Database changed
mysql> show databases;
+-----+
| Database      |
+-----+
| information_schema |
| Hotelerie      |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0,00 sec)

mysql>
```

```
mysql> show tables;
+-----+
| Tables_in_Hotelerie |
+-----+
| Chambre           |
| Facture          |
| Reservation      |
| Statistique      |
| client           |
+-----+
5 rows in set (0,00 sec)

mysql>
```

- Téléchargement de ProxySQL



-Configuration de ProxySQL

A screenshot of a terminal window showing the configuration file for Proxysql. The file is named "/etc/proxysql.cnf" and is being edited with "GNU nano 2.9.3". The configuration includes the following sections:

```
datadir="/var/lib/proxysql"
errorlog="/var/lib/proxysql/proxysql.log"

admin_variables=
{
    admin_credentials="admin:admin;cluster_admin:cluster_admin_password"
#    mysql_ifaces="127.0.0.1:6032;/tmp/proxysql_admin.sock"
#    mysql_ifaces="0.0.0.0:6032"
#    refresh_interval=2000
#    debug=true
}

mysql_variables=
```

```
}
```

```
mysql_variables=
```

```
{
```

```
    read_timeout=5000
```

```
    write_timeout=5000
```

```
    connect_timeout_server=5000
```

```
    connect_timeout_server_max=10000
```

```
    threads=4
```

```
    max_connections=2048
```

```
    default_query_delay=0
```

```
    default_query_timeout=36000000
```

```
    have_compress=true
```

```
    poll_timeout=2000
```

```
#     interfaces="0.0.0.0:6033;/tmp/proxysql.sock"
```

```
    interfaces="0.0.0.0:6033"
```

```
    default_schema="information_schema"
```

```
    stacksize=1048576
```

```
    server_version="5.5.30"
```

```
# make sure to configure monitor username and password
```

```
# make sure to configure monitor username and password
```

```
# https://github.com/sysown/proxysql/wiki/Global-variables#mysql-monitor_user$
```

```
    monitor_username="monitor"
```

```
    monitor_password="monitor"
```

```
    monitor_history=600000
```

```
    monitor_connect_interval=60000
```

```
    monitor_ping_interval=10000
```

```
    monitor_read_only_interval=1500
```

```
    monitor_read_only_timeout=500
```

```
    ping_interval_server_msec=120000
```

```
    ping_timeout_server=500
```

```
    commands_stats=true
```

```
    sessions_sort=true
```

```
    connect_retries_on_failure=10
```

```
}
```

```
#     { address="127.0.0.2" , port=3306 , hostgroup=1 },
```

```
#     { address="127.0.0.3" , port=3306 , hostgroup=1 },
```

```
#     { address="127.0.0.4" , port=3306 , hostgroup=1 },
```

```
#     { address="/var/lib/mysql/mysql.sock" , port=0 , hostgroup=1 }
```

```
{ address="127.0.0.1" , port=3306 , hostgroup=10, max_connections=100},
```

```
{ address="127.0.0.1" , port=3307 , hostgroup=10, max_connections=100},
```

```
{ address="127.0.0.1" , port=3308 , hostgroup=10, max_connections=100}
```

```
)
```

```
mysql_users:
(
#      {
#          username = "username" # no default , required
#          password = "password" # default: ''
#          default_hostgroup = 0 # default: 0
#          active = 1           # default: 1
#      },
#      {
#          username = "root"
#          password = ""
#          default_hostgroup = 0
#          max_connections=1000
#          default_schema="test"
#          active = 1
#      },
#      {
#          username = "myuser" , password = "mypassword" , default_hostgroup = 10
#      }
)
```

Activités Terminal dib 20:11

root@fatysow-VirtualBox: ~

Fichier Édition Affichage Rechercher Terminal Onglets Aide

root@fatysow-V... x root@fatysow-V... x root@fatysow-V... x root@fatysow-Vi... x

GNU nano 2.9.3 /etc/proxysql.cnf Modifié

```
mysql_replication_hostgroups=
(
#      {
#          writer_hostgroup=30
#          reader_hostgroup=40
#          comment="test repl 1"
#      },
#      {
#          writer_hostgroup=50
#          reader_hostgroup=60
#          comment="test repl 2"
#      }
{writer_hostgroup=10, reader_hostgroup=20,comment="Hotelerie"}
)

# http://www.hyperrealm.com/libconfig/libconfig_manual.html#Configuration-File$
#
# Below is the BNF grammar for configuration files. Comments and include direc$
```

#

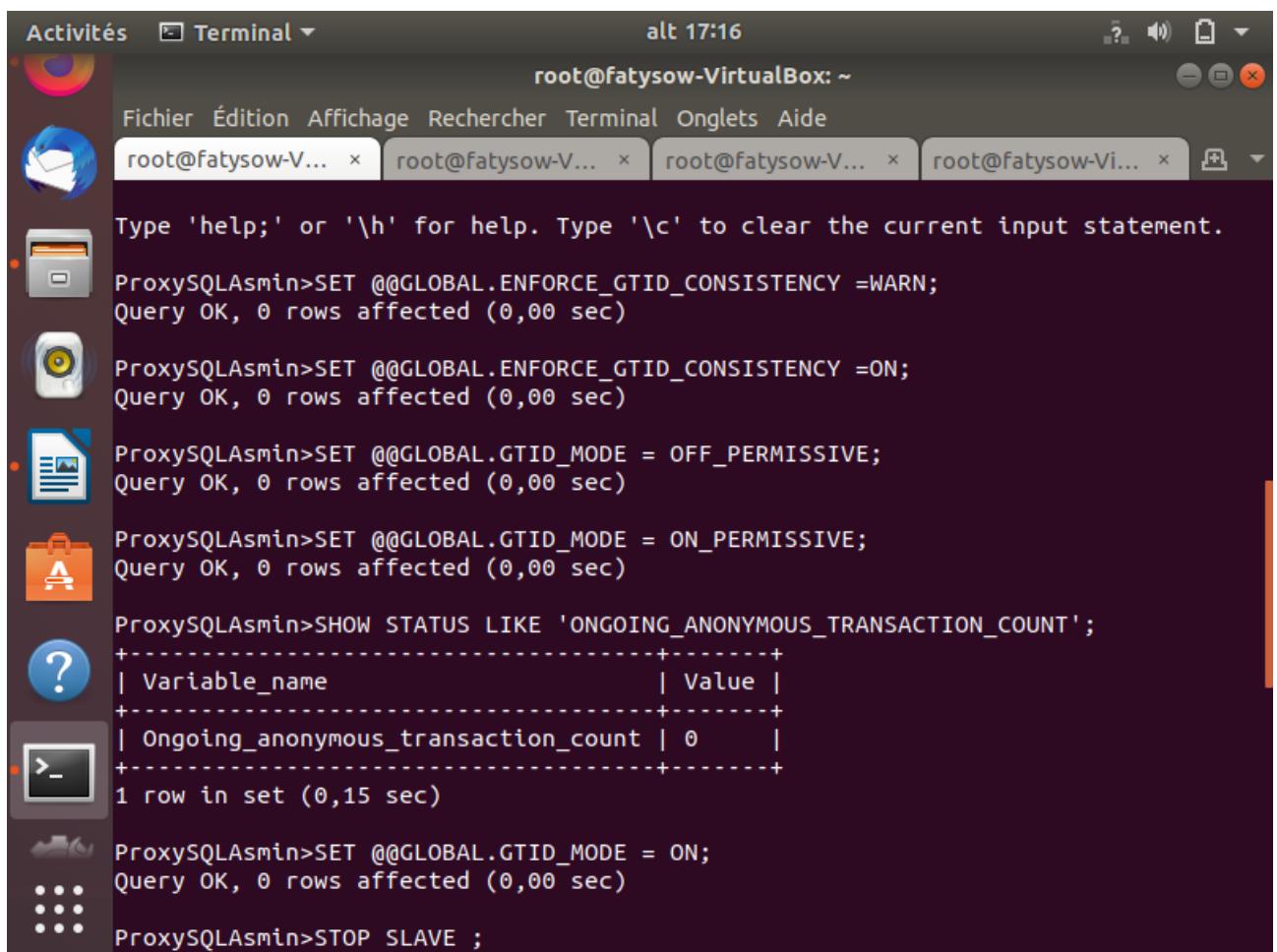
^G Aide ^O Écrire ^W Chercher ^K Couper ^J Justifier
^X Quitter ^R Lire fich. ^\ Remplacer ^U Coller ^T Orthograp.

```
root@fatysow-VirtualBox:~# systemctl status proxysql.service
● proxysql.service - High Performance Advanced Proxy for MySQL
   Loaded: loaded (/lib/systemd/system/proxysql.service; enabled; vendor preset: enabled)
   Active: active (running) since Thu 2023-03-16 22:16:23 GMT; 3 days ago
     Main PID: 943 (proxysql)
        Tasks: 25 (limit: 2318)
      CGroup: /system.slice/proxysql.service
              └─943 /usr/bin/proxysql --idle-threads -c /etc/proxysql.cnf
                ├─944 /usr/bin/proxysql --idle-threads -c /etc/proxysql.cnf

mar 16 22:16:19 fatysow-VirtualBox systemd[1]: Starting High Performance Advanced Proxy for MySQL...
mar 16 22:16:20 fatysow-VirtualBox proxysql[794]: 2023-03-16 22:16:20 [INFO] Usage of the proxy is now tracked by the MySQL Metrics API
mar 16 22:16:20 fatysow-VirtualBox proxysql[794]: 2023-03-16 22:16:20 [INFO] Configuration file: /etc/proxysql.cnf
mar 16 22:16:20 fatysow-VirtualBox proxysql[794]: 2023-03-16 22:16:20 [INFO] No configuration file found, using default values
mar 16 22:16:23 fatysow-VirtualBox systemd[1]: Started High Performance Advanced Proxy for MySQL.

lines 1-15/15 (END)
```

-Activation des transaction GTID en ligne



```
root@fatysow-VirtualBox: ~
Fichier Édition Affichage Rechercher Terminal Onglets Aide
root@fatysow-V... x root@fatysow-V... x root@fatysow-V... x root@fatysow-V... x
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

ProxySQLAsmin>SET @@GLOBAL.ENFORCE_GTID_CONSISTENCY =WARN;
Query OK, 0 rows affected (0,00 sec)

ProxySQLAsmin>SET @@GLOBAL.ENFORCE_GTID_CONSISTENCY =ON;
Query OK, 0 rows affected (0,00 sec)

ProxySQLAsmin>SET @@GLOBAL.GTID_MODE = OFF_PERMISSIVE;
Query OK, 0 rows affected (0,00 sec)

ProxySQLAsmin>SET @@GLOBAL.GTID_MODE = ON_PERMISSIVE;
Query OK, 0 rows affected (0,00 sec)

ProxySQLAsmin>SHOW STATUS LIKE 'ONGOING_ANONYMOUS_TRANSACTION_COUNT';
+-----+-----+
| Variable_name | Value |
+-----+-----+
| Ongoing_anonymous_transaction_count | 0 |
+-----+-----+
1 row in set (0,15 sec)

ProxySQLAsmin>SET @@GLOBAL.GTID_MODE = ON;
Query OK, 0 rows affected (0,00 sec)

ProxySQLAsmin>STOP SLAVE ;
```

-Puis on change le MASTER

```
CHANGE MASTER TO MASTER_HOST='127.0.0.1',MASTER_PORT=3306 ,MASTER_USER='replication' ,MASTER_PASSWORD= 'replication_password',MASTER_AUTO_POSITION= 1 ;
```

-Création des nœuds

on se connecte au serveur MYSQL

```
mysql -u root -p
```

```
Enter password:  
Welcome to the MySQL monitor. Commands end with ; or \g.  
Your MySQL connection id is 38609  
Server version: 5.7.41-0ubuntu0.18.04.1 (Ubuntu)  
  
Copyright (c) 2000, 2023, Oracle and/or its affiliates.  
  
Oracle is a registered trademark of Oracle Corporation and/or its  
affiliates. Other names may be trademarks of their respective  
owners.  
  
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.  
  
mysql> create user noeud1 identified by '1234';  
Query OK, 0 rows affected (0,61 sec)  
  
mysql> create user noeud2 identified by '1234';  
Query OK, 0 rows affected (0,00 sec)
```

```
mysql> create user 'MASTER'@'localhost' identified by '1234';  
Query OK, 0 rows affected (0,00 sec)  
  
mysql> grant all privileges on *.* to 'MASTER'@'localhost';  
Query OK, 0 rows affected (0,00 sec)  
  
mysql> grant all privileges on *.* to noeud1;  
Query OK, 0 rows affected (0,01 sec)
```

-créer une nouvelle utilisateur de réPLICATION dans le nœud maître

```
mysql> CREATE USER 'replication_user'@'localhost' IDENTIFIED BY '1234';  
Query OK, 0 rows affected (0,00 sec)  
  
mysql> GRANT REPLICATION SLAVE ON *.* TO 'replication_user'@'localhost';
```

-puis on exécute la commande suivante

```
SHOW MASTER STATUS\G
```

```

root@fatysow-VirtualBox:~# mysql -u MASTER -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 1641
Server version: 5.7.41-0ubuntu0.18.04.1-log (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> CREATE USER 'replication_user'@'localhost' IDENTIFIED BY '1234';
ERROR 1396 (HY000): Operation CREATE USER failed for 'replication_user'@'localhost'
mysql> SHOW MASTER STATUS\G
***** 1. row *****
    File: mysql-bin.000001
    Position: 419
    Binlog_Do_DB:
    Binlog_Ignore_DB:
    Executed_Gtid_Set:
1 row in set (0,00 sec)

```

-Ensuite on se connecte

mysql -u root -p

dans cette étape il faut que le serveur esclave se connecte au serveur indiquée
on met

STOP SLAVE ;

CHANGE MASTER TO MASTER_HOST='127.0.0.1'
,MASTER_USER='replication_user', MASTER-PASSWORD='1234',
MASTER_LOG_FILE='mysql-bin.000001', MASTER_LOG_POS='419';

```

mysql> STOP SLAVE;
Query OK, 0 rows affected, 1 warning (0,01 sec)

mysql> CHANGE MASTER TO MASTER_HOST='127.0.0.1', MASTER_USER='replication_user'
, MASTER_PASSWORD='1234', MASTER_LOG_FILE='mysql-bin.000001', MASTER_LOG_POS=41
9;
Query OK, 0 rows affected, 2 warnings (0,14 sec)

mysql> START SLAVE;
Query OK, 0 rows affected (0,01 sec)

mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| Hotelerie     |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0,02 sec)

```

-Vérification de la réPLICATION maître

```
mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| Hotelerie |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0,02 sec)
```

-vérification des nœuds

```
root@fatysow-VirtualBox:~# mysql -u noeud1 -p
Enter password:
Welcome to the MySQL monitor. Commands end with ; or \g.
Your MySQL connection id is 2948
Server version: 5.7.41-0ubuntu0.18.04.1-log (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW DATABASES;
+-----+
| Database |
+-----+
| information_schema |
| Hotelerie |
| mysql |
| performance_schema |
| sys |
+-----+
5 rows in set (0,00 sec)
```

```
root@fatysow-VirtualBox:~# mysql -u noeud2 -p
Enter password:
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3066
Server version: 5.7.41-0ubuntu0.18.04.1-log (Ubuntu)

Copyright (c) 2000, 2023, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> SHOW DATABASES;
+-----+
| Database      |
+-----+
| information_schema |
| Hotelerie      |
| mysql          |
| performance_schema |
| sys            |
+-----+
5 rows in set (0,00 sec)
```

Conclusion

On remarque que la base de données qu'on avait créé s'appelle Hôtellerie.
Le maître a été répliquée sur l'esclave .Donc la réPLICATION mysql maître-esclave fonctionne comme prévu . Les copies des fichiers de base de données seront répliquées sur le serveur esclave.

API

Une **API**, pour Application programming interface, est un programme permettant à deux applications distinctes de communiquer entre elles et d'échanger des données. Cela évite notamment de recréer et développer entièrement une application pour y ajouter ses informations.

J'ai eu à créer pour chaque table son CRUD(Create,Read,Update et Delete et eventuellement un get)qui permet aux applications d'ajouter supprimer ou même lister tous les éléments d'une table quelconque

```
Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use hotel;
Database changed
mysql> select *from client;
+-----+-----+-----+-----+-----+-----+
| nom   | prenom | telephone | email      | classeChambre | id |
+-----+-----+-----+-----+-----+-----+
| zahra  | tall   | 781705064 | zaratall@gma | standard      | 1  |
| oulimata | sene   | 781705064 | zaratall@gma | standard      | 2  |
| aissatou | tall   | 782505064 | fatiamrou@gma | economique    | 3  |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)

mysql> select *from chambre;
+-----+-----+-----+-----+-----+
| id   | numero | etage | disponibilite | TypeChambre | PrixParNuit |
+-----+-----+-----+-----+-----+
| 1    | 204   | 3     | oui           | economique | 25000       |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select *from chambre;
+-----+-----+-----+-----+-----+
| id   | numero | etage | disponibilite | TypeChambre | PrixParNuit |
+-----+-----+-----+-----+-----+
| 1    | 204   | 3     | oui           | economique | 25000       |
+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)

mysql> select *from reservation;
+-----+-----+-----+-----+-----+-----+-----+
| id   | nom    | prenom | telephone | classe   | chambre | DateChambre | Nuite | DateSortie |
+-----+-----+-----+-----+-----+-----+-----+
| 1    | zahra  | tall   | 78012220 | economise | 206    | 2023-06-05 | 5     | 2023-06-10 |
+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select *from statistique;
+-----+-----+-----+-----+-----+
| id   | NbreChambreReservee | NbreChambreOcupee | occupation | NomHotel      | AdressHotel |
+-----+-----+-----+-----+-----+
| 1    | 200                | 100              | 80%        | Radisson Blue | Dakar        |
+-----+-----+-----+-----+-----+
mysql> select *from facture;
+-----+-----+-----+
| id   | NumeroFacture | MontantTotal | ListeServices |
+-----+-----+-----+
| 1    | 10             | 90000          | dej,petit-dejeuner |
+-----+-----+-----+
1 row in set (0.00 sec)
```

GET localhost:9090/clients

Status: 200 OK Time: 161 ms Size: 460 B Save as Example

```

1 [
2   {
3     "id": 1,
4     "nom": "zahria",
5     "prenom": "tall",
6     "telephone": "781785864",
7     "email": "zaratal@gmail.com",
8     "classeChambre": "standard"
9   },
10  {
11    "id": 2,
12    "nom": "oulimata",
13    "prenom": "sene",
14    "telephone": "781795064",
15    "email": "zaratal@gmail.com",
16    "classeChambre": "standard"
17  },
18  {
19    "id": 3,
20    "nom": "aissatou",
21    "prenom": "tall",
22    "telephone": "782585864",
23    "email": "fatiamrou@gmail.com",
24    "classeChambre": "economique"
25  }
26 ]

```

localhost:9090/clients/1

Status: 200 OK Time: 18 ms Size: 231 B Save as Example

```

1 {"id":1,"nom":"zahria","prenom":"tall","telephone":"781785864","email":"zaratal@gmail.com","classeChambre":"standard"}
2

```

POST localhost:9090/clients/new

Status: 201 Created Time: 12 ms Size: 225 B Save as Example

```

1 {"nom":"fatou",
2  "prenom":"fall",
3  "telephone":"782585864",
4  "email":"fatima@gmail.com",
5  "classeChambre":"standard"}
6

```

Pretty Raw Preview Visualize JSON

```

1 [
2   {
3     "id": 4,
4     "nom": "fatou",
5     "prenom": "fall",
6     "telephone": "782585864",
7     "email": "fatima@gmail.com",
8     "classeChambre": "standard"
9   }
10 ]

```

```

log. Database changed
mysql> select *from client;
+-----+-----+-----+-----+-----+-----+
| nom   | prenom | telephone | email      | classeChambre | id |
+-----+-----+-----+-----+-----+-----+
| zahra | tall   | 781705064 | zaratall@gma | standard      | 1  |
| oulimata | sene   | 781705064 | zaratall@gma | standard      | 2  |
| aisssatou | tall   | 782505064 | fatiamrou@gma | economique   | 3  |
| fatou   | falll  | 782505064 | fatima@gma  | standard      | 4  |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

The screenshot shows a Postman request configuration. The method is set to PUT, the URL is `localhost:9090/clients/update/?id=1`, and the Body tab is selected with the JSON option chosen. The JSON payload is:

```

1 {"nom":"fatou",
2 "prenom":"fall",
3 "telephone":"782505064",
4 "email":"fatima@gma",
5 "classeChambre":"standard"
6 }

```

Below the request, the response status is shown as 200 OK with a time of 22 ms and a size of 75 B. The response body is empty.

```

mysql> select *from client;
+-----+-----+-----+-----+-----+
| nom   | prenom | telephone | email      | classeChambre | id |
+-----+-----+-----+-----+-----+
| zahra | tall   | 781705064 | zaratall@gma | standard      | 1  |
| oulimata | sene   | 781705064 | zaratall@gma | standard      | 2  |
| aiissatou | tall   | 782505064 | fatiamrou@gma | economique   | 3  |
| fatou   | falll  | 782505064 | fatima@gma  | standard      | 4  |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql> select *from client;
+-----+-----+-----+-----+-----+
| nom   | prenom | telephone | email      | classeChambre | id |
+-----+-----+-----+-----+-----+
| fatou   | tall   | 782505064 | fatima@gma  | standard      | 1  |
| oulimata | sene   | 781705064 | zaratall@gma | standard      | 2  |
| aiissatou | tall   | 782505064 | fatiamrou@gma | economique   | 3  |
| fatou   | falll  | 782505064 | fatima@gma  | standard      | 4  |
+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

```

DELETE <localhost:9090/clients/delete/?id=1> Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies Beautify

Body (none) form-data x-www-form-urlencoded raw binary GraphQL JSON

```

1 {"nom": "fatou",
2 "prenom": "tall",
3 "telephone": "782505064",
4 "email": "fatima@gma",
5 "classeChambre": "standard"
6 }
```

Body Cookies Headers (3) Test Results Status: 200 OK Time: 16 ms Size: 169 B Save as Example

Pretty Raw Preview Visualize JSON

```

1 {
2   "message": "Client deleted successfully",
3   "status": "success"
4 }
```

```
mysql> select *from client;
+-----+-----+-----+-----+-----+-----+
| nom      | prenom | telephone | email      | classeChambre | id |
+-----+-----+-----+-----+-----+-----+
| oulimata | sene   | 781705064 | zaratall@gma | standard      | 2  |
| aisssatou | tall   | 782505064 | fatiamrou@gma | economique    | 3  |
| fatou     | falll  | 782505064 | fatima@gma   | standard      | 4  |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.00 sec)
```

GET <localhost:9090/reservations> Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (3) Test Results Status: 200 OK Time: 8 ms Size: 272 B Save as Example

Pretty Raw Preview Visualize JSON

```

1 {
2   "id": 1,
3   "nom": "zahra",
4   "prenom": "tall",
5   "telephone": "78012220",
6   "classe": "economise",
7   "chambre": 266,
8   "DateChambre": "2023-06-05",
9   "Nuite": 5,
10  "DateSortie": "2023-06-10"
11 }
12 }
```

GET <localhost:9090/reservations/1> Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (3) Test Results Status: 200 OK Time: 7 ms Size: 279 B Save as Example

Pretty Raw Preview Visualize Text

```

1 {"id": 1, "nom": "zahra", "prenom": "tall", "telephone": "78012220", "classe": "economise", "chambre": 266, "DateChambre": "2023-06-05", "Nuite": 5, "DateSortie": "2023-06-10"}  
2 
```

POST <localhost:9090/reservations/new> Send

Params Authorization Headers (8) Body **JSON** Pre-request Script Tests Settings Cookies Beautify

Body: none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 [{"nom": "zahra", "prenom": "tall", "telephone": "78512220", "classe": "economisue", "chambre": 206, "DateChambre": "2023-06-05", "Nuite": 5, "DateSortie": "2023-06-10"}]
```

Body Cookies Headers (3) Test Results Status: 201 Created Time: 11 ms Size: 274 B Save as Example

Pretty Raw Preview Visualize JSON

```
1 {
  "id": 2,
  "nom": "zahra",
  "prenom": "tall",
  "telephone": "78512220",
  "classe": "economisue",
  "chambre": 206,
  "DateChambre": "2023-06-05",
  "Nuite": 5,
  "DateSortie": "2023-06-10"
}
```

PUT <localhost:9090/reservations/update/?id=1> Send

Params Authorization Headers (8) Body **JSON** Pre-request Script Tests Settings Cookies Beautify

Body: none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 [{"nom": "amina", "prenom": "tall", "telephone": "3389555", "classe": "standard", "chambre": 206, "DateChambre": "2023-06-05", "Nuite": 5, "DateSortie": "2023-06-10"}]
```

mysql> select *from reservation;

id	nom	prenom	telephone	classe	chambre	DateChambre	Nuite	DateSortie
1	amina	tall	3389555	standard	206	2023-06-05	5	2023-06-10
2	zahra	tall	78512220	economisue	206	2023-06-05	5	2023-06-10

2 rows in set (0.00 sec)

DELETE <localhost:9090/reservations/delete/?id=1> Send

Params Authorization Headers (8) Body **JSON** Pre-request Script Tests Settings

Body Cookies Headers (3) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
  "message": "Reservation deleted successfully",
  "status": "success"
}
```

mysql> select *from reservation;

id	nom	prenom	telephone	classe	chambre	DateChambre	Nuite	DateSortie
2	zahra	tall	78512220	economisue	206	2023-06-05	5	2023-06-10

1 row in set (0.00 sec)

DELETE localhost:9090/chambres Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (3) Test Results Status: 200 OK Time: 6 ms Size: 213 B Save as Example

Pretty Raw Preview Visualize JSON

```
1 [ {  
2   "id": 1,  
3   "numero": 204,  
4   "etage": 3,  
5   "disponibilite": "oui",  
6   "typeChambre": "economique",  
7   "PrixParNuit": 25000  
8 }  
9 ]  
10 [ ]  
11 [ ]  
12 [ ]  
13 [ ]  
14 [ ]  
15 [ ]  
16 [ ]
```

GET localhost:9090/chambres/1 Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings Cookies

Body Cookies Headers (3) Test Results Status: 200 OK Time: 5 ms Size: 220 B Save as Example

Pretty Raw Preview Visualize Text

```
1 { "id":0,"numero":204,"etage":3,"disponibilite":"oui","TypeChambre":"economique","PrixParNuit":25000}  
2
```

POST localhost:9090/chambres/new Send

Params Authorization Headers (8) Body Pre-request Script Tests Settings

Body form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {"numero":200,"etage":2,"disponibilite":"non","TypeChambre":"economique","PrixParNuit":25000}
```

Body Cookies Headers (3) Test Results Status: 200 OK

Pretty Raw Preview Visualize JSON

```
1 [ {  
2   "id": 2,  
3   "numero": 200,  
4   "etage": 2,  
5   "disponibilite": "non",  
6   "TypeChambre": "economique",  
7   "PrixParNuit": 25000  
8 }  
9 ]  
10 [ ]  
11 [ ]  
12 [ ]  
13 [ ]  
14 [ ]  
15 [ ]  
16 [ ]
```

```
mysql> select *from chambre;
+----+-----+-----+-----+-----+-----+
| id | numero | etage | disponibilite | TypeChambre | PrixParNuit |
+----+-----+-----+-----+-----+-----+
| 1  | 204   | 3    | oui          | economique  | 25000     |
| 2  | 200   | 2    | non         | economique  | 25000     |
+----+-----+-----+-----+-----+-----+
```

localhost:9090/chambres/update/?id=1

POST localhost:9090/chambres/update/?id=1

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON Beautify

1 {"numero":200,"etage":3,"disponibilite":"non","TypeChambre":"economique","PrixParNuit":25000}

Body Cookies Headers (2) Test Results

Pretty Raw Preview Visualize Text

Status: 200 OK Time: 11 ms Size: 75 B Save as Example

localhost:9090/chambres/delete/?id=1

DELETE localhost:9090/chambres/delete/?id=1

Params Authorization Headers (8) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

1 {"numero":200,"etage":3,"disponibilite":"non","TypeChambre":"economique","PrixParNuit":25000}

Body Cookies Headers (3) Test Results

Pretty Raw Preview Visualize JSON

1 "message": "Room deleted successfully",
2 "status": "success"

Status: 200

```
mysql> select *from chambre;
+----+-----+-----+-----+-----+-----+
| id | numero | etage | disponibilite | TypeChambre | PrixParNuit |
+----+-----+-----+-----+-----+-----+
| 2  | 200   | 2    | non         | economique  | 25000     |
+----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

```
mysql> select *from statistique;
+----+-----+-----+-----+-----+-----+
| id | NbreChambreReservee | NbreChambreLibre | TauxOccupation | NomHotel      | AdressHotel |
+----+-----+-----+-----+-----+-----+
| 1  |          200 |             10 |    98%        | Radisson Blue | Dakar       |
+----+-----+-----+-----+-----+-----+
```

APPLICATION DESKTOP

Python est un langage de programmation puissant et facile à apprendre. GTK est une bibliothèque de widgets pour l'interface graphique utilisateur. Elle offre des fonctionnalités pour créer des interfaces utilisateur conviviales.

D'abord j'ai développé l'application de référence en y ajoutant une interface d'authentification avec deux boutons de redirection qui donnent l'option de s'inscrire ou de se connecter.

Dans le cas où les données saisies sont fausses un message d'erreur s'affiche.

Maintenant après avoir vérifié si les données saisies existent déjà (c'est- à-dire que l'utilisateur s'est déjà inscrit) et que la connexion a réussie , une interface menu s'ouvre automatiquement :

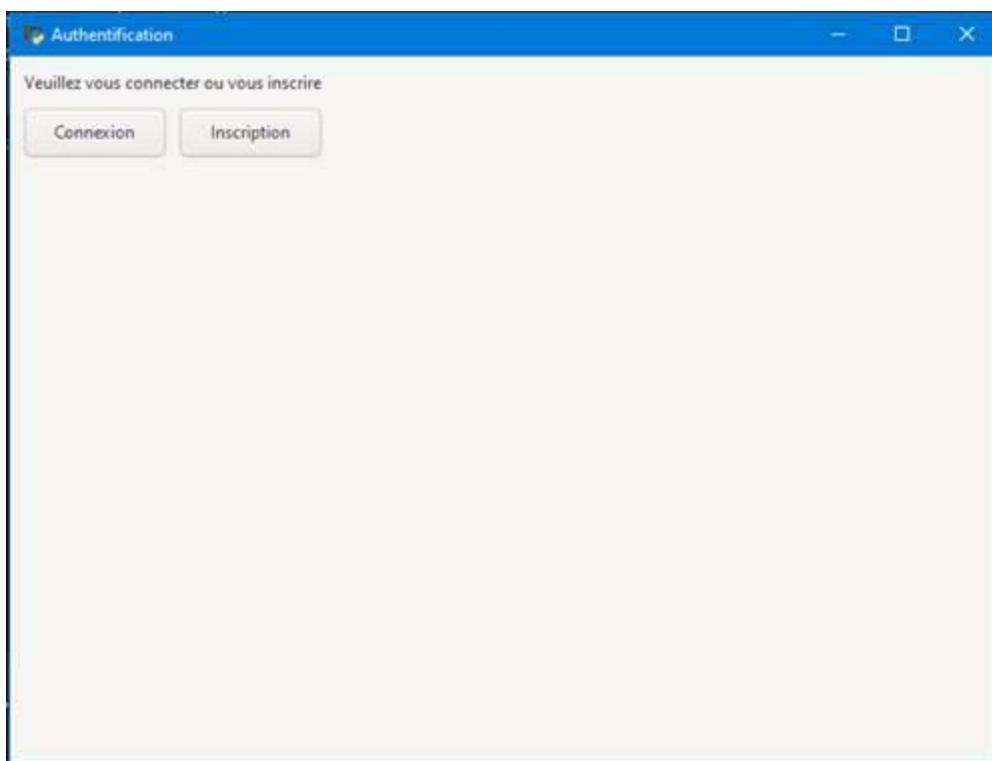
Gestion des réservations : Le menu Réservation est composé de sous menu ajouter réservation ,liste des réservations et annuler réservation.

Par exemple :si on clique le formulaire ajouter réservation une formulaire d'ajout de réservation avec des champs pour saisir les informations s'affiche :

Gestion des chambres :

Gestion des clients : Pour gérer les clients, nous avons créé un menu client contenant des sous menus qui redirige chacun vers une page . Nous avons également utilisé une base de données de la proxy SQL pour stocker les informations sur les clients. Les utilisateurs peuvent afficher les clients existants ainsi que leur état.

Le menu Édition comporte les sous menus de paramétrage de l'hôtel :modifier nom



 Inscription

INSCRIPTION

Informations

Nom	<input type="text"/>
Prénom	<input type="text"/>
Numéro	<input type="text"/>
Role	Gérant ▾
Email	<input type="text"/>
Mot de passe	<input type="text"/>
Confirmer votre mot de passe	<input type="text"/>

 Connexion

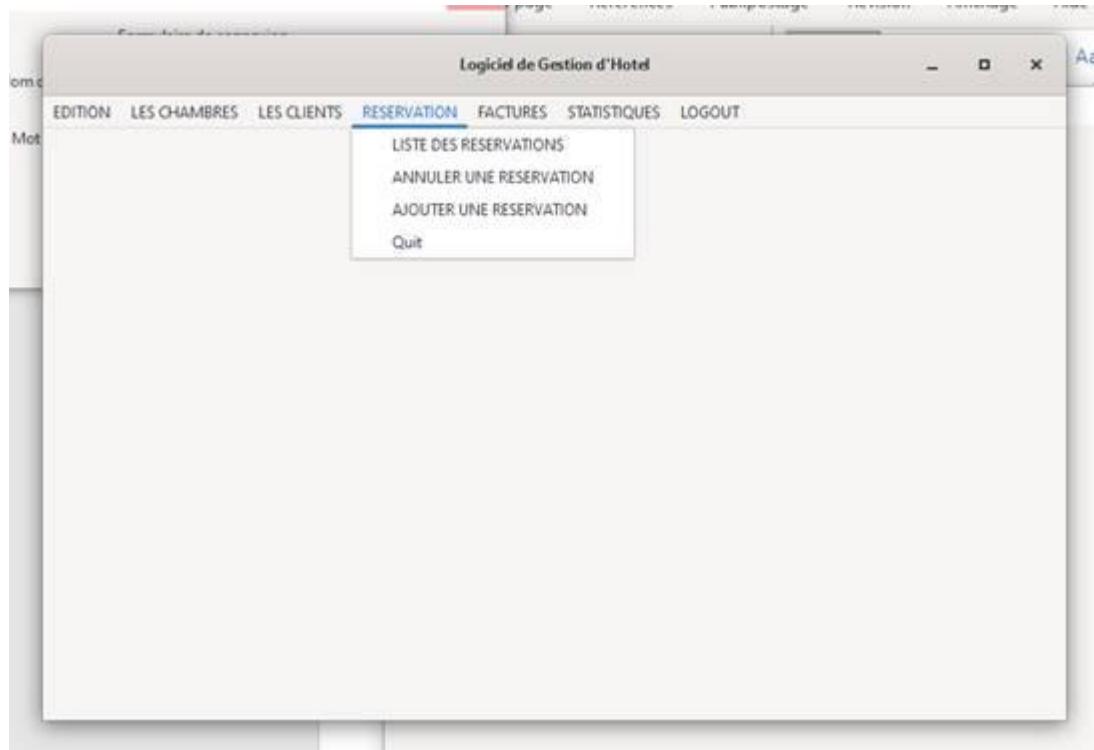
Formulaire de connexion

Nom d'utilisateur:

Mot de passe:

Logiciel de Gestion d'Hotel

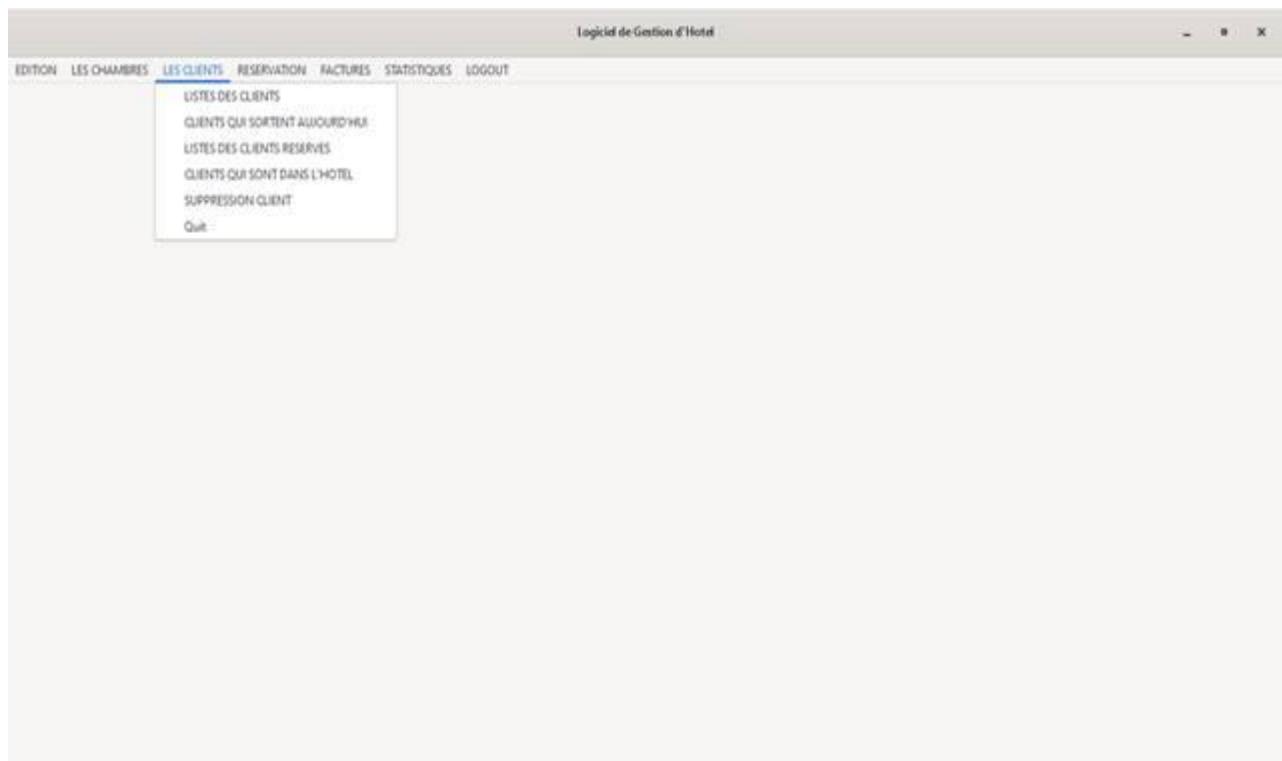
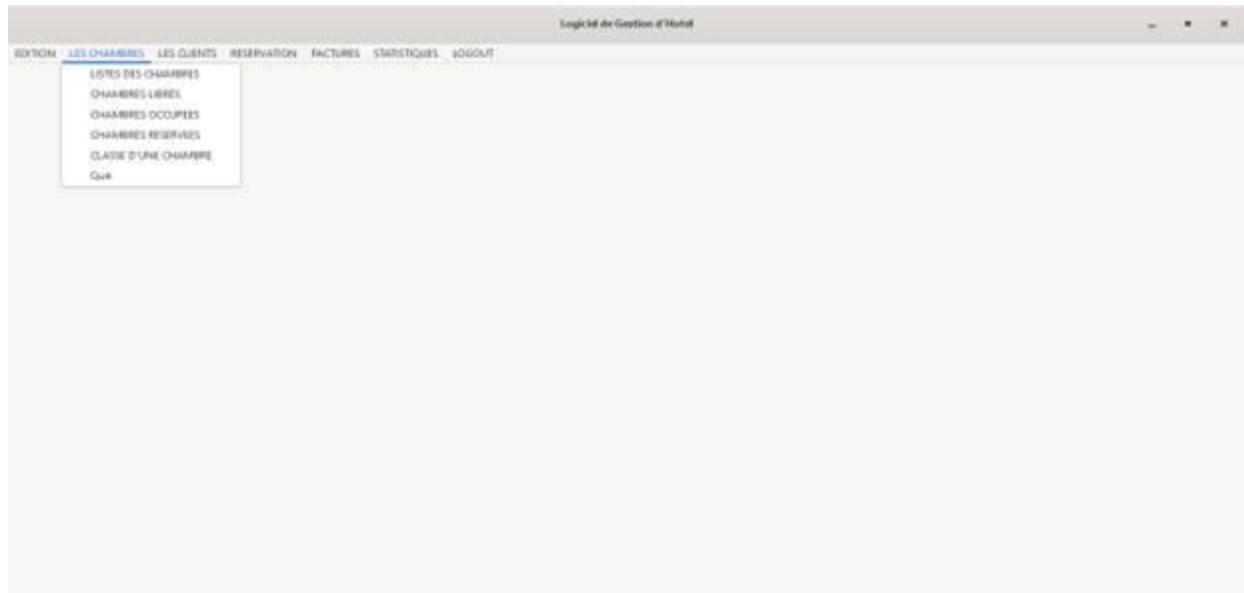
EDITION LES CHAMBRES LES CLIENTS RESERVATION FACTURES STATISTIQUES LOGOUT

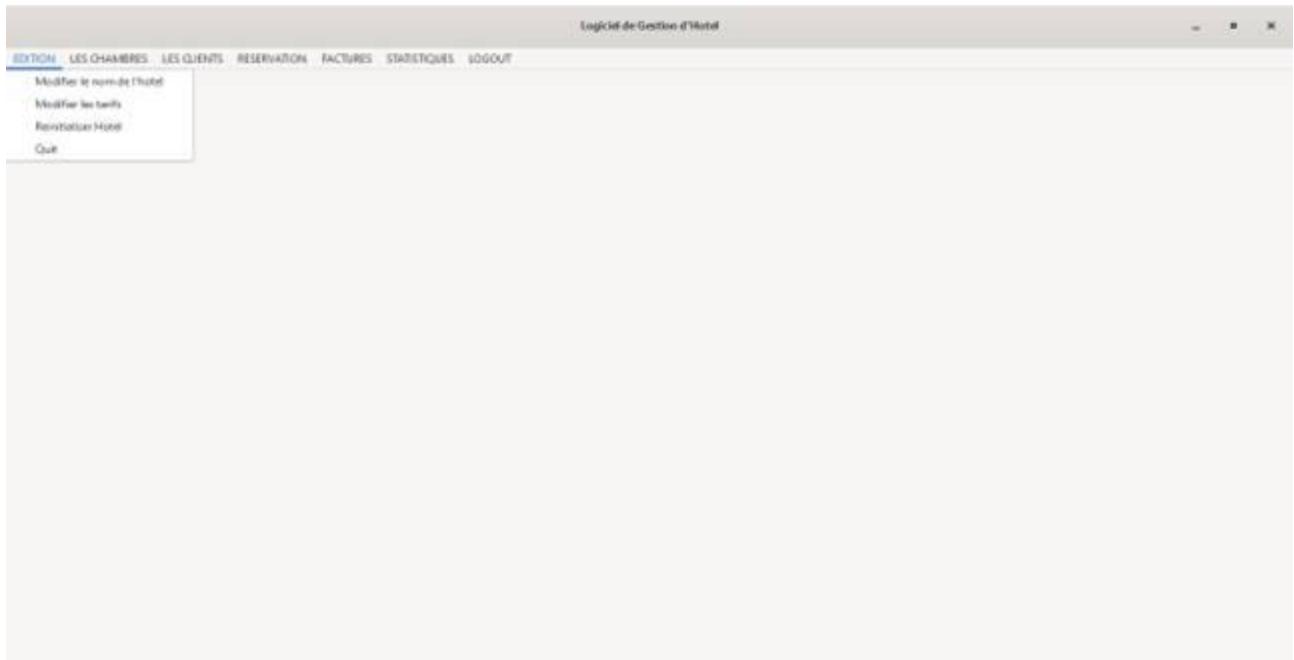


The screenshot shows a "Reservation Form" dialog box. It contains fields for inputting guest information and reservation details. The fields are labeled and have corresponding input boxes:

- ID: An input box containing a single vertical bar character.
- Nom: An empty input box.
- Prenom: An empty input box.
- telephone: An empty input box.
- Classe: An empty input box.
- Chambre: An empty input box.
- Nuitée: An empty input box.
- Date d'entrée: An empty input box.
- Date de sortie: An empty input box.

At the bottom of the form is a large "Submit" button.

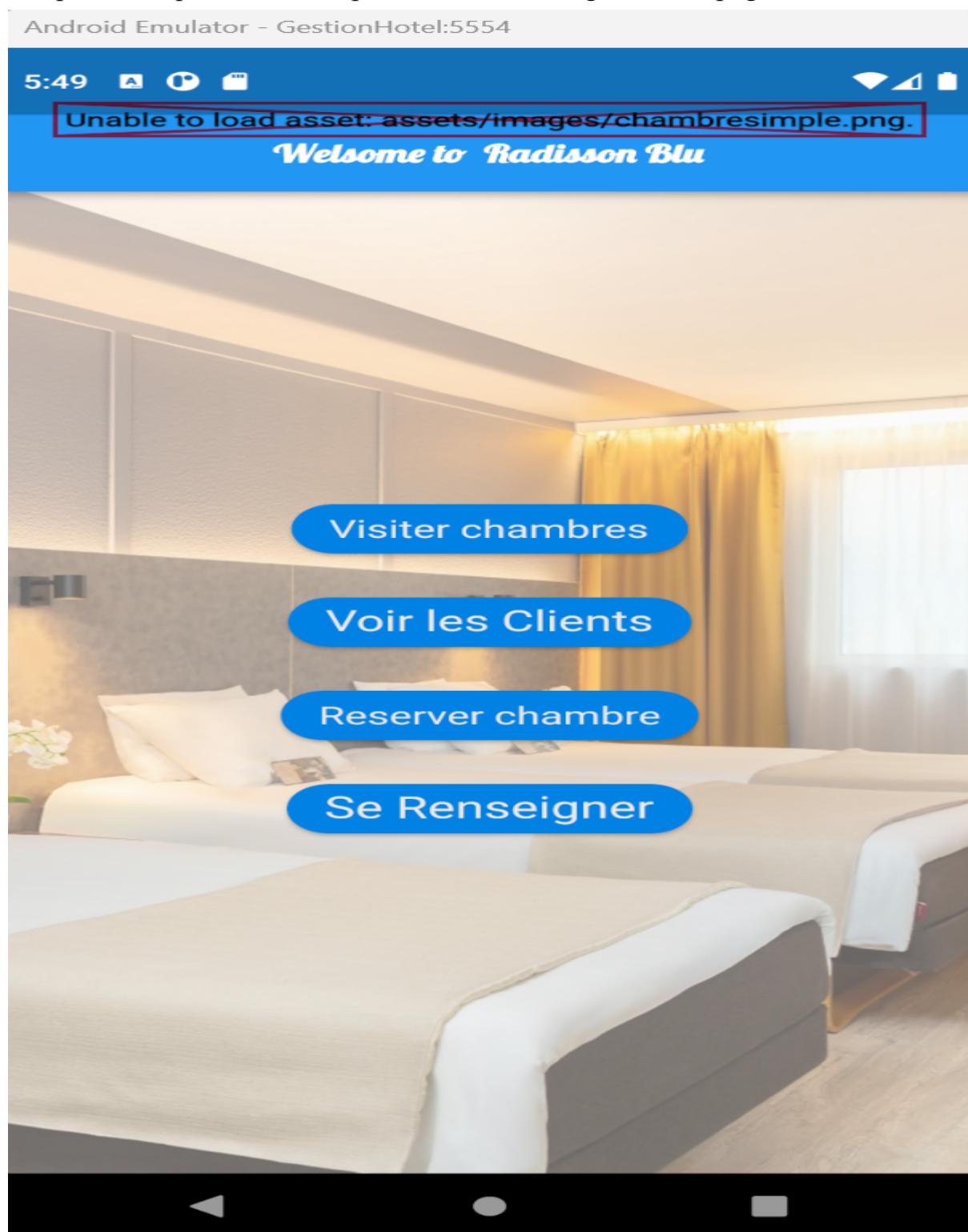




Rapport de présentation du développement mobile fait avec flutter

I. La page d'accueil

En début de lieu nous avons la page d'accueil de l'hôtel appelée Radisson Blu composée de quatre boutons pouvant nous rediriger vers la page concernée.



illustration

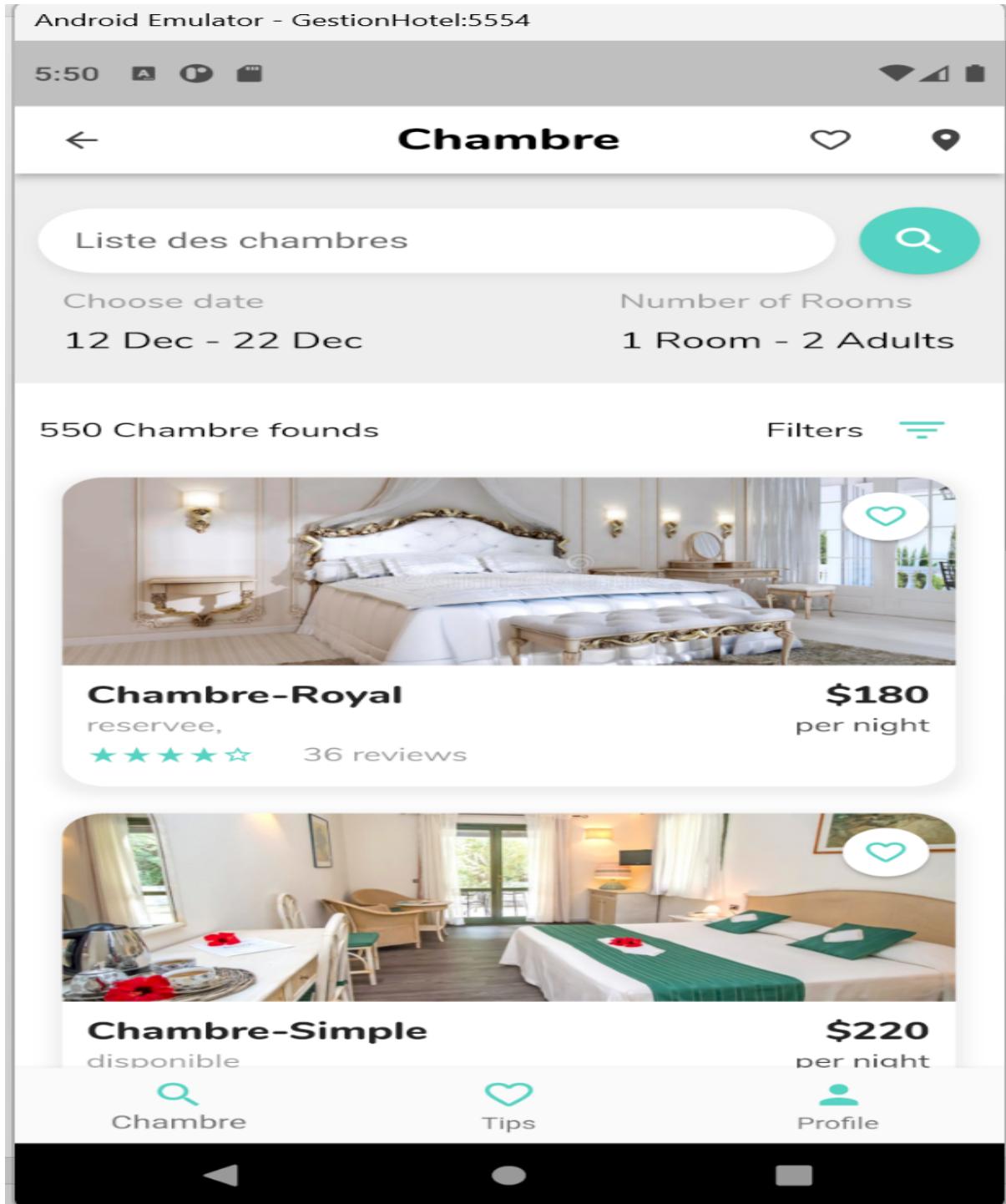
Pour que ces pages soient rediriger il nous a fallu les importer dans le fichier **main.dart**

```
--- main.dart X
lib > main.dart
● 1 import 'package:flutter/material.dart';
  2 import 'package:google_fonts/google_fonts.dart';
  3 import 'AuthentificationPage.dart';
  4 import 'ReservationPage.dart';
  5 import 'QuestionPage.dart';
  6
  7
  8
  9   const d_green = Color(0xFF54D3C2);
10   void main() {
11     WidgetsFlutterBinding.ensureInitialized();
12     runApp(MyApp());
13   }
14   class HomePage extends StatelessWidget {
15     @override
16     Widget build(BuildContext context) {
17       return WillPopScope(
18         onWillPop: () async{
19           return false;
20         },
21         child: Scaffold(
22           appBar: AppBar(
23             automaticallyImplyLeading: false,
24             title: Text('Accueil')
25           ),
26           body: Center(
27             child: ElevatedButton(
28               style: ElevatedButton.styleFrom(
29                 textStyle: TextStyle(fontSize: 21),
30                 primary: Color.fromARGB(255, 12, 130, 228),
31                 onPrimary: Color.fromARGB(255, 236, 233, 233),
32                 shape: RoundedRectangleBorder(
33                   borderRadius: BorderRadius.circular(20),
34                 ), // RoundedRectangleBorder
35               ),
36               onPressed: () {
37                 Navigator.push(
38                   context,
39                   MaterialPageRoute(
40                     builder: (context) => AuthentificationPage(),
41                   ), // MaterialPageRoute
42                 );
43               },
44               child: Text('Voir les Clients'), // ElevatedButton
45             ),
46             ),
47             child: ElevatedButton(
48               style: ElevatedButton.styleFrom(
49                 textStyle: TextStyle(fontSize: 18),
50                 primary: Color.fromARGB(255, 12, 130, 228),
51                 onPrimary: Color.fromARGB(255, 236, 233, 233),
52                 shape: RoundedRectangleBorder(
53                   borderRadius: BorderRadius.circular(20),
54                 ), // RoundedRectangleBorder
55               ),
56               onPressed: () {
57                 Navigator.push(
58                   context,
59                   MaterialPageRoute(
60                     builder: (context) => ReservationPage(),
61                   ), // MaterialPageRoute
62                 );
63               },
64             ),
65           )
66         ),
67       );
68     }
69   }
70 }
```

et les appelle dans une fonction **onPressed** situer dans le bouton de redirection.

```
ElevatedButton(
  style: ElevatedButton.styleFrom(
    textStyle: TextStyle(fontSize: 21),
    primary: Color.fromARGB(255, 12, 130, 228),
    onPrimary: Color.fromARGB(255, 236, 233, 233),
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(20),
    ), // RoundedRectangleBorder
  ),
  onPressed: () {
    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => AuthentificationPage(),
      ), // MaterialPageRoute
    );
  },
  child: Text('Voir les Clients'), // ElevatedButton
SizedBox(height: 20),
ElevatedButton(
  style: ElevatedButton.styleFrom(
    textStyle: TextStyle(fontSize: 18),
    primary: Color.fromARGB(255, 12, 130, 228),
    onPrimary: Color.fromARGB(255, 236, 233, 233),
    shape: RoundedRectangleBorder(
      borderRadius: BorderRadius.circular(20),
    ), // RoundedRectangleBorder
  ),
  onPressed: () {
    Navigator.push(
      context,
      MaterialPageRoute(
        builder: (context) => ReservationPage(),
      ), // MaterialPageRoute
    );
  },
)
```

II. la page visiter chambre



comme dit tantôt en cliquant sur les boutons ils nous redirigent vers la page concernée.c'est le cas pour le bouton "visiter chambre" qui nous montrent les différents types de chambres qui composent l'hôtel cette photo est la parfaite illustration.

Je tiens à préciser que pour pouvoir afficher les photos il m'a fallu créer un dossier appelé **assets** dans le document de mon projet (gestion hotel) dans ce dossier je crée un second appelé **images** et

dans image nous y mettons nos photos.

Nom	Modifié le	Type	Taille
.dart_tool	12/03/2023 21:56	Dossier de fichiers	
.idea	12/03/2023 21:56	Dossier de fichiers	
android	12/03/2023 21:56	Dossier de fichiers	
assets	13/03/2023 23:13	Dossier de fichiers	
build	16/03/2023 00:47	Dossier de fichiers	
ios	21/03/2023 17:40	Dossier de fichiers	
lib	21/03/2023 16:11	Dossier de fichiers	
linux	12/03/2023 21:56	Dossier de fichiers	
macos	12/03/2023 21:56	Dossier de fichiers	
test	13/03/2023 16:05	Dossier de fichiers	
web	12/03/2023 21:56	Dossier de fichiers	
windows	12/03/2023 21:56	Dossier de fichiers	
.flutter-plugins	21/03/2023 17:45	Fichier FLUTTER-PLU...	1 Ko
.flutter-plugins-dependencies	21/03/2023 17:45	Fichier FLUTTER-PLU...	3 Ko

Ensuite pour les appelés dans le code source j'ai éditer le fichier public.yaml en décommentant la ligne **#assets** et en y ajoutant le chemins de mon dossier images

```
! pubspec.yaml
  ! pubspec.yaml
  65 # For information on the generic Dart part of this file, see the
  66 # following page: https://dart.dev/tools/pub/pubspec
  67
  68 # The following section is specific to Flutter packages.
  69 flutter:
  70
  71     # The following line ensures that the Material Icons font is
  72     # included with your application, so that you can use the icons in
  73     # the material Icons class.
  74     uses-material-design: true
  75
  76     # To add assets to your application, add an assets section, Like this:
  77
  78     assets:
  79         - assets/images/
```

Remarque: il faut tenir compte de l'indentation des écritures pour que ca marche

toujours dans public.yaml il faut ajouter les extensions **google fonts** pour la police des écritures mais aussi **calendar_page** qui prend en compte les dates

```

# the latest version available on pub.dev. To see which dependencies have new
# versions available, run `flutter pub outdated`.
dependencies:
  flutter:
    | sdk: flutter
    http: ^0.13.4
    material:

  # The following adds the Cupertino Icons font to your application.
  # Use with the CupertinoIcons class for iOS style icons.
  cupertino_icons: ^1.0.2
  cloud_firestore: ^4.4.5

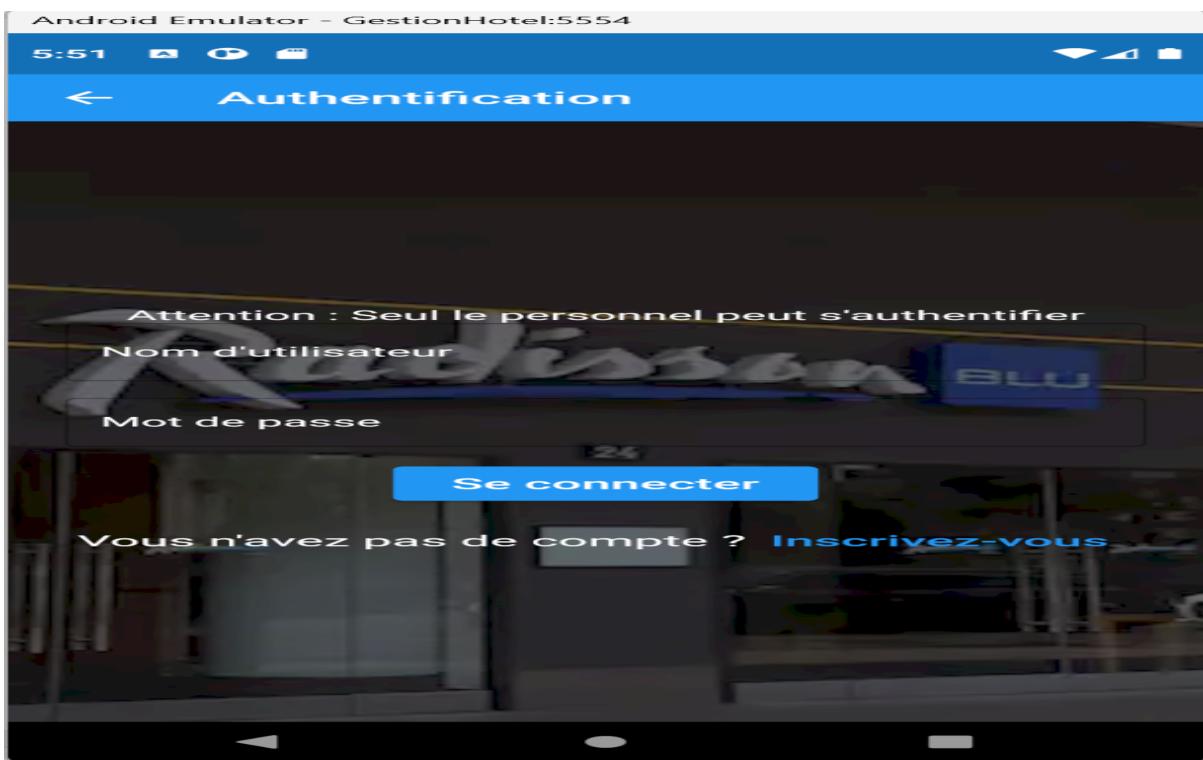
  google_fonts: ^2.1.0
  table_calendar: ^3.0.0

```

et vous remarquerez bien sûr que nous l'avons ajouté dans le fichier dependencies.

III. la page voir clients

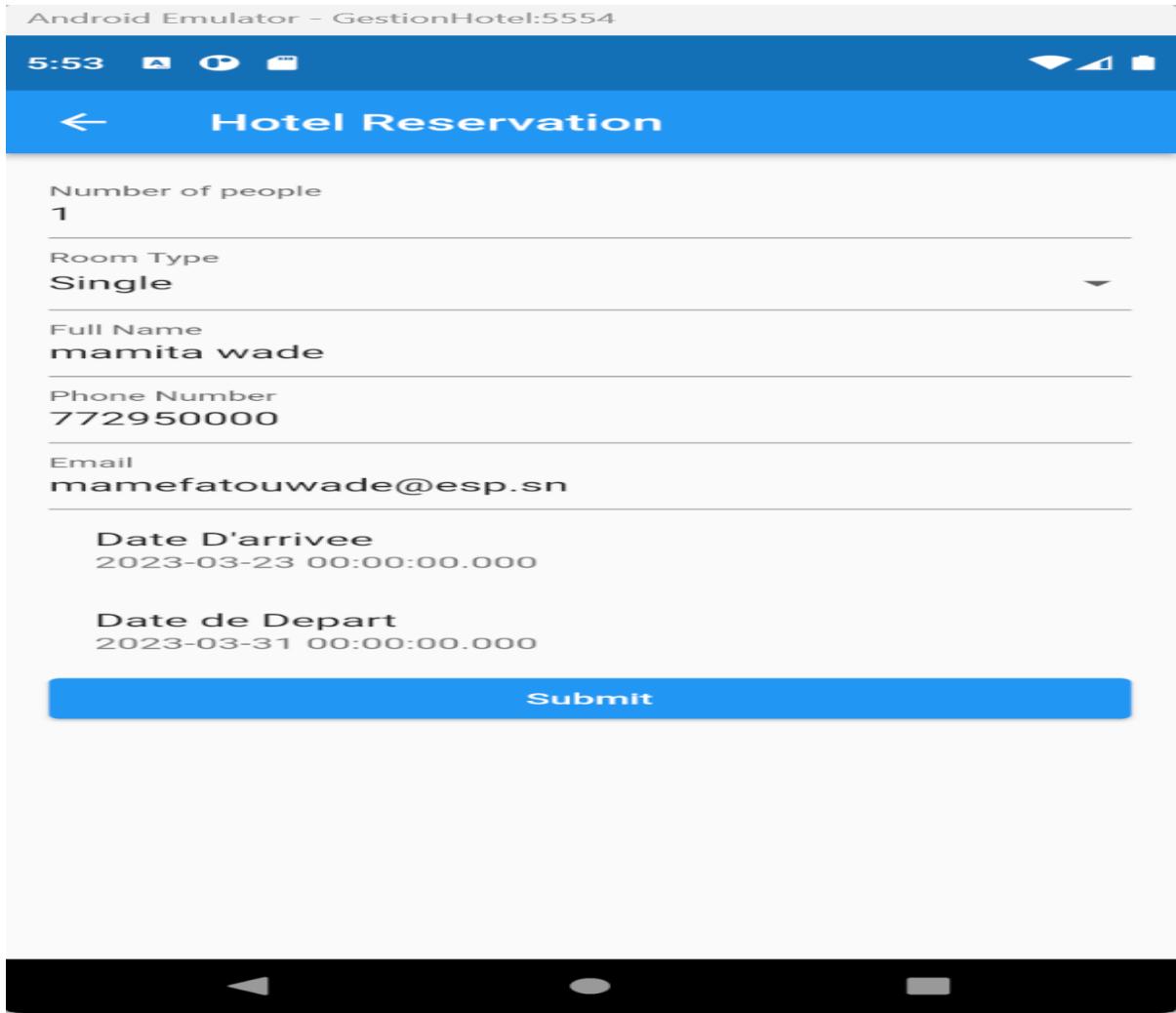
le bouton “voir clients” nous redirige vers cette page ci-dessous indique dans la photo.comme nous le constatons seul le personnel de l'hôtel a le pouvoir les différents clients qui séjournent dans l'hôtel



IV. la page de réservation

la page de réservation générée par le bouton “reserver chambre” qui permet de faire sa réservation en indiquant toutes les données demandées.

→ Si les données remplis sont correct la réservation c'est bien passée



→ Sinon un message d'erreur s'affiche à la partie où vous devez modifier comme le montre cette image

6:03

← Hotel Reservation

Number of people
4

The unique room can only accommodate 1 person

Room Type
Single

Full Name
mame fatou wade

Phone Number
782950135

Email
mamefatouwade@esp.sn

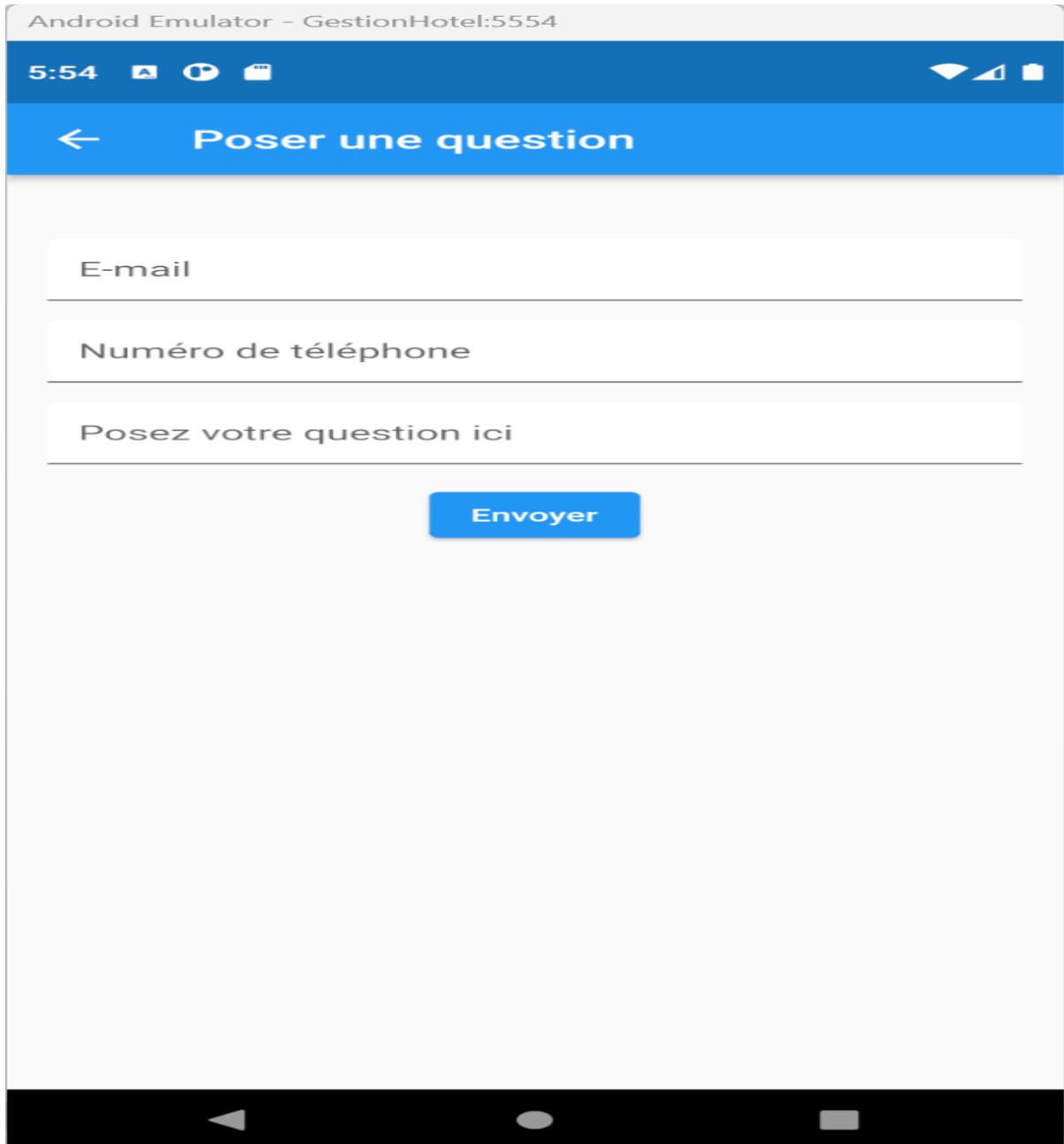
Date D'arrivee
2023-03-22 00:00:00.000

Date de Depart
2023-03-30 00:00:00.000

Submit

V. la page de services

et enfin la dernière partie pour poser n’importe quel questions au personnel qui vous répondra soit par mail ou par sms ou par appel.



Je précise que ces liens ci-dessous m'ont été bénéfiques. Ces personnes m'ont aidé à me situer et à découvrir le framework flutter. Je le recommanderai à tous ceux qui veulent se lancer dans le monde de la programmation.

[Graven](#)

[documentation flutter](#)

[INMANENCIIA](#)