My Hotel Booking System: Documentation

Installation:

Virtual Environment: inside the directory pip install –

python -m virtualenv <nameOfEnv>

C:\Users\'Username'\venv\Scripts\activate.bat

pip install virtualenvwrapper-win

browse to the virtual environment root directory in the project or activate it as shown above, then

pip install –r requirements.txt

$ mysql -u root -p

mysql> CREATE USER 'is\_admin'@'localhost' IDENTIFIED BY 'is2020';

mysql> CREATE DATABASE ivorysand\_db;

mysql> GRANT ALL PRIVILEGES ON ivorysand\_db . \* TO 'is\_admin'@'localhost';

**Models: Database Representation**

Employee, Department, and Role

Customer, Rooms, Reservation Transactions, Customer Reviews

Each room needs information on:

Type of room (standard, double, deluxe, and suite)

# people who can stay in room

Textual description (depends on type of room, varies between one

hotel to another for same type of room)

Room number [unique to a room...obviously] (some are offered at

discount price

Discounted price for period of time at most once a year

Store discounts as %s of regular price

Must have start and end date

Floor number

Price per night (varies between one hotel to another for same type

of room)

room-images

***Database Design (Hersh Notes):*** *(aid in adding various functionalities to the website design)*

● Hotels

○ Must hold **at least 10 rooms** each

○ Diff breakfasts at each hotel

○ Diff services at each hotel

● Customer

○ ID, Name, Email, Phone Number

● Reservations

○ Invoice number - unique to the reservation number

○ Date reservation was made - use auto insert in the table

○ List of rooms reserved - allowed to reserve multiple rooms in the same

reservation

○ Credit Card Info

■ Card #

■ Name on Card

■ Expiration Date

■ Security Code

■ Billing Address

■ Type of Card

○ Check In / Check Out Date

■ # of days staying

○ Multiple room reservations done for a particular room in the same reservation

(diff dates of course)

○ Overlapping room reservations for same room (using room id) is NOT allowed

○ Breakfast Orders

■ Include multiple breakfast types

■ Multiple orders for each breakfast

■ Store # of orders for each breakfast type in each reservation

● Total # should not exceed # of people physically allowed in room

○ Service Orders

■ One or more services possible to add

○ Total Amount to Pay

■ Sum cost all breakfast orders

■ Sum cost all service orders

■ Sum cost different rooms in reservation

● Discounted room prices need to be taken into account

● Customer Reviews

○ Review ID

○ Name of Reviewer

○ 3 Types

■ Room Review

■ Service Review

■ Breakfast Review

○ 1 to 10 rating

○ Customer can add one or more review

○ Only allowed to write reviews for...

■ Hotels, Room Types, Breakfast Types, Service Types they had reserved

■ Either in present / past

User – Department - Role

Admin, Customer, Simple Reservation. 🡪 Login, Registration, Logout

Leave other employees first 🡪 to add later

Button: Make a reservation.

Admin – able to add employees.

AFTER MODELS

DATABASE INITIALISATION, MIGRATION, AND UPGRADE:

flask db init

flask db migrate

flask db upgrade

BLUEPRINTS:

Folders with \_\_init\_\_.py, forms.py, and views.py each: admin, auth, home, reservation, room...

UPDATE app/\_\_init\_\_.py to retgister blueprints created