

Reference Sheet

1. External Libraries

There is no external library used in this Hotel Management System.

2. Brief walk through

The home page of this system is called homepage.php. You could either register for this website or just log in. Be aware that you have to register before you log in, and you will not be able to visit any webpages except the homepage, log in page and registration page without logging in.

If you want to create an account for the website, just click on the registration button and fill in the form on the page shown. The form accept various kinds of inputs but the website will check their validation at back-end. If there is any invalid input, user assistance message will be prompted to alert user.

After logging in, you could choose the service you want. Book one or more rooms or view all your reservations. When you choose to book a room, you will be lead into a page to specify your booking information, including check in date, check out date and room type. After that you will be shown a plane figure to choose your floor and your room in preference. Be attention that for privacy issue, rooms will only have two status, available (Book Now) and unavailable (Unavailable). A room could be unavailable for reasons that room type doesn't match or reserved by other consumers in that period. After choosing a room, the assisting message will ask users if they want to continue booking, if yes, jump to book specifications, if no, back to service page.

When you want to view all your reservations, click on 'view current bookings', then you get a page with all your bookings listed. Additionally, you could cancel a certain reservation by clicking on the cancel button under it.

After all your actions, click log out to write off your account.

For administration page, administrator logs in by normal user logging in page. The administrator username is 'admin', with the password of 'admin123456'. The settings for administrator id fixed, which means there will be no more account valid except this one for safety issue. All pages for staffs is protected from direct visit.

The administrator could view all reservations in sunny isle regardless of who booked them by clicking on 'View all reservations'. Finally, click on 'Log out' to log off.

3. Mandatory

For additional requirements, registered user is not a necessary requirement for checking in, therefore, there will be one more form called 'Non-user' to store information of people who checks in without registration. Because they have no usernames, the passport ID becomes the unique ID for identifying the non-users. The relationship of Users and Non-users is a many to many relationship, to clarify this relationship, I separated them into two one-to-many relationships and create a new assistant form called 'Relationship', the foreign key from 'User' is 'Username' and the one from 'Non-user' is 'passport ID'.

For reservations, the non-users should be mapped to reservations when checking in. This is a one-to-many relationship because a non-user could only check in for one room at a time and one room reservation could have multiple people to check in. The key for

To enable staffs to create administration account, a table for called 'Staff' was created. Multiple staffs manage multiple reservations. However, this many-to-many relationship is not that significant because a reservation will not have a closely relationship with a certain staff. Therefore, I didn't set any keys for staffs and reservations.

An additional table for Chef was set up to implement meal reservation system. Similar to the relationship between staffs and reservations, no relationship separation adopted or foreign key adopted.

SQL code

```
1 CREATE TABLE User (
2     username varchar(20),
3     password varchar(20),
4     passport varchar(20),
5     telephone int(11),
6     email varchar(25),
7 );
8
9 CREATE TABLE Non-user (
10     passport varchar(20),
11     realname varchar(20),
12     bookid int(3) NOT NULL UNIQUE,
13 );
14
15 CREATE TABLE Relationship (
16     username varchar(20),
17     passport varchar(20),
18 );
19
20 CREATE TABLE Reservation (
21     username varchar(20),
22     ckind date,
23     ckout date,
24     roomtype varchar(20),
25     roomnum varchar(20),
26     checkin bit,
27     bookid int(3) NOT NULL UNIQUE,
28 );
29
30 CREATE TABLE Chef (
31     name varchar(20),
32     dish varchar(50),
33     weekday1 varchar(10),
34     weekday2 varchar(10),
35     weekday3 varchar(10),
36 );
37
38 CREATE TABLE Meal (
39     username varchar(20),
40     chefname varchar(20),
41     dish varchar(50),
42     time datetime,
43 );
```

Entity Relationship Diagram

