FPT UNIVERSITY

ASSIGNMENT

Java Web Application Development (PRJ301)

Team 1

HKCBooking

Instructor: Võ Thị Thanh Vân

Class: SE1612

Group members:

SE150276 - Phạm Vĩnh Tài

SE150271 - Phạm Phú Hưng

SE151508 - Lê Chí Cường

Contents

I. Case Study 5
1. Back ground: 5
2. Use-case Diagram 6
3. Class Diagram 7
4.1 Login User Account 8
4.2 Register User Account 9
4.3 Logout User Account 10
4.4 User Info 11
4.5 Change Password User Account 12
4. 6 Index Page 13
4. 7 Filter Page 14
4. 8 Room Detail Page 15
4. 9 Review room by authorization account 16
4. 10 Cart Page 17
4. 11 Remove Cart Item 18
4. 12 View History Page by authorization account 19
4. 13 Cancel History by authorization account 20
4. 14 Booking Orders by Admin account 21
4. 15 Add Room by Admin account 22
4. 16 Edit Room by Admin account 23
4. 17 Delete Room by Admin account 24
4. 18 Edit Room Detail by Admin account 25
4. 19 Add Room Type by Admin account 26
4. 20 View Users by Admin account 27
4.21 View User Histories by Admin account 28
4.22 Add Params Controller 29
4.23 Deployment Diagram 30
II. Database design 31
1. Database requirement: 31
2. Entity Relationship Diagram 32
3. Relationship diagram 33
III. Project Site Map and Explanation 34
1. Site Map 34
2. Explanation 35
2.1 Login Page 35
2.2 Register page 37

2.3 Filter page 39
2.4 User Information page 41
2.5 Change password page 43
2.6 Room detail page 45
2.7 Booking Order Management 46
2.8 Add room for management 48
2.9 Add room type 50
2.10 Edit room 52
2.11 Room list manager 53
2.12 User list 55
2.13 User history booking 56
IV. Conclusion 57
V. Reference 58

I. Case Study

1. Back ground:

HKCBooking is a page for hotel, homestay and resort owners to register for rooms to promote and promote their brands to readers. At the same time, it also provides online booking, paying fees right on the channel. For users, the site has up to 200+ accommodation for rent, all reviewed, detailed with amenities, number of rooms, location and cost.

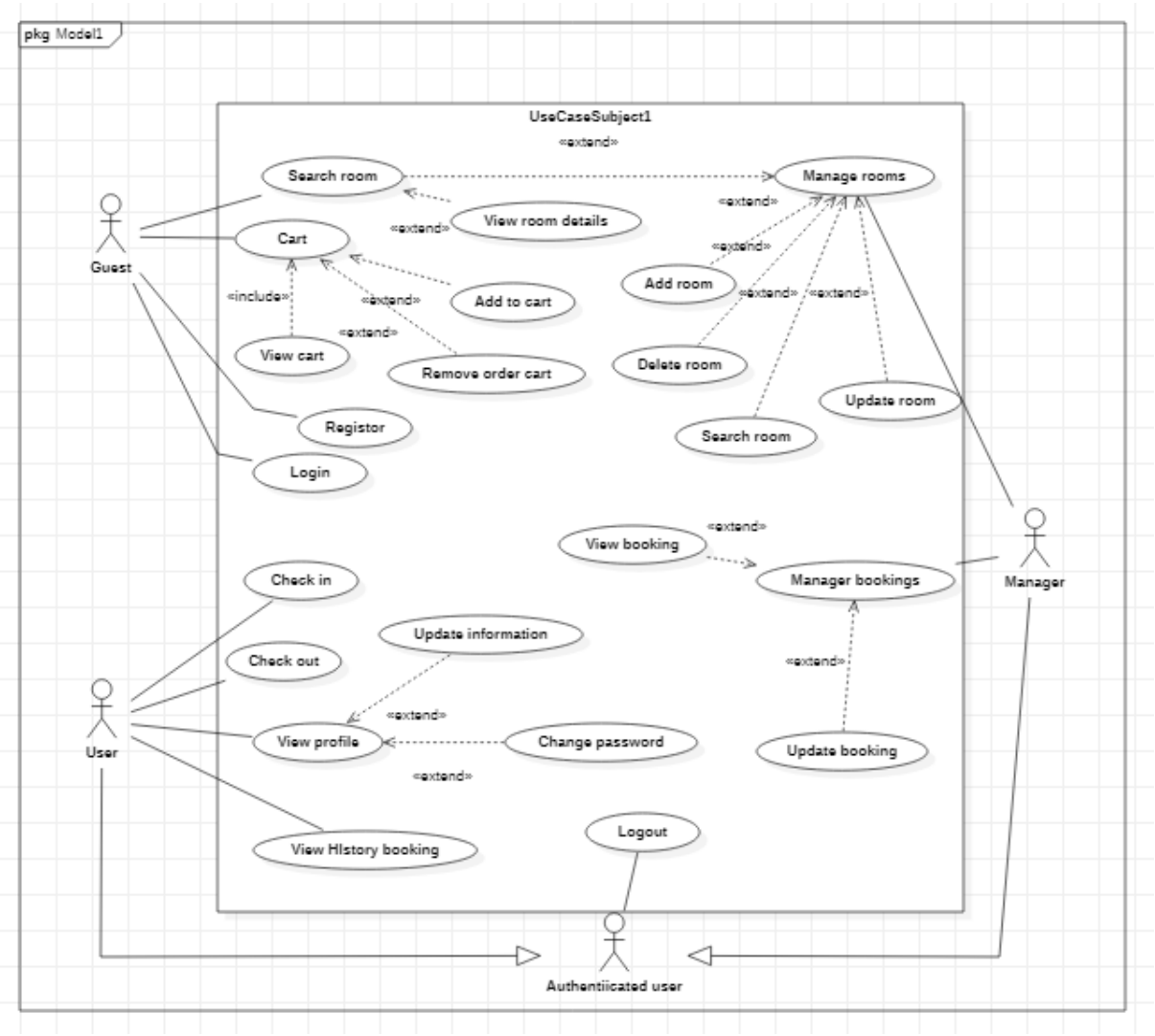
This hotel booking website is useful when there are more than 200 apartments in all places from Hanoi to Ho Chi Minh for your reference, comparison, booking in advance.this hotel chain needs to develop more strongly, the general manager has a Web Application Development Plan to manage all these web application online systems which need some conditions as follows:

A management system Manage staff and customer order system. It should follow the following business rules.

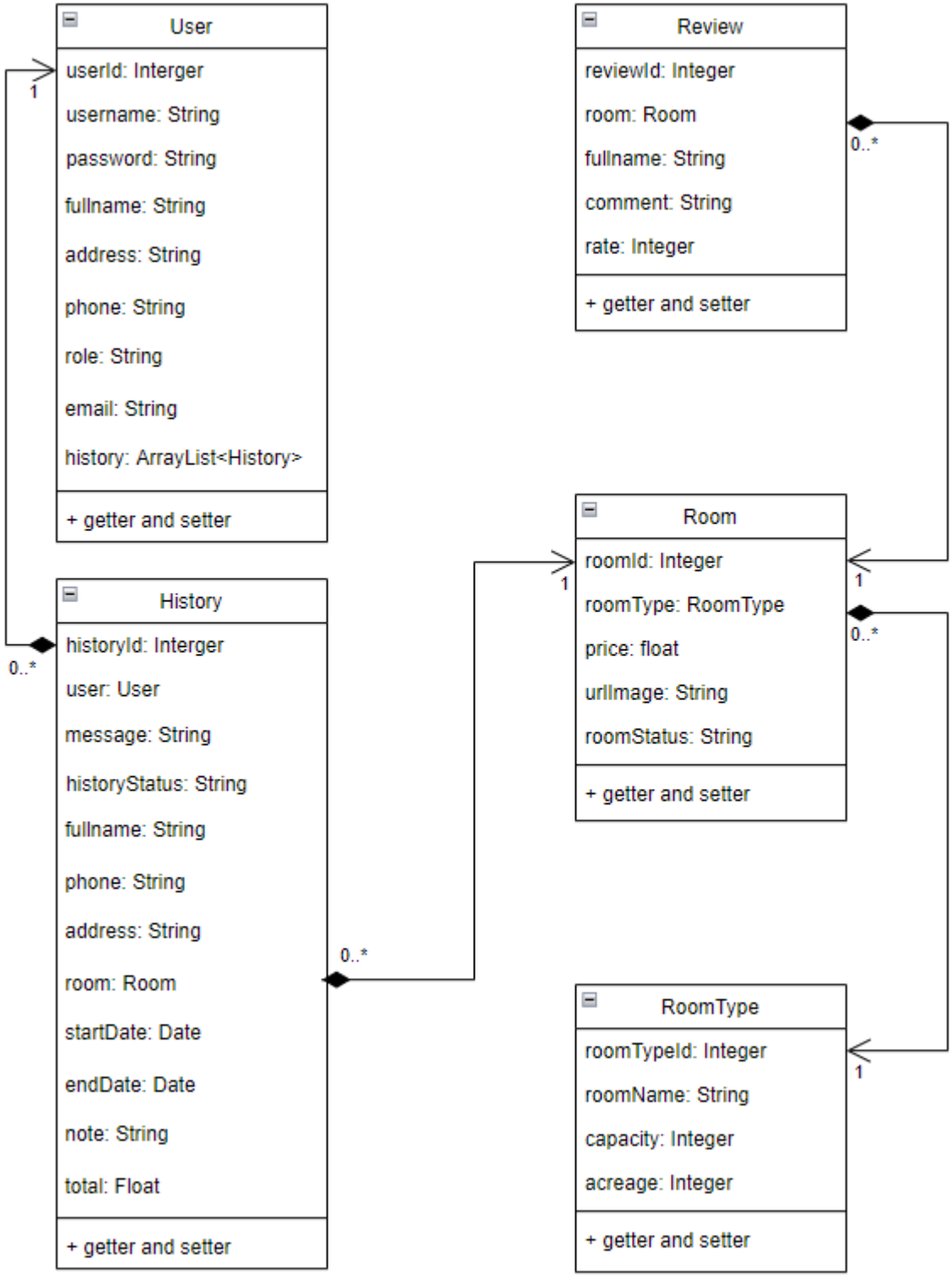
Only authenticated users have the ability to book a room. The ability to manage and Customer capabilities must be separate. For details, authenticated users know that customers can view detailed room information, reservation (include one or more rooms on another date). Customers can view their booking information and cancel in special situations, view booking history. Management capabilities include Add new rooms, update room information. View all customer orders and cancel or pay Pre-order

More special guests can see all the room lists, add favourite rooms to cart, but when booking a room, guests need to register or log in to the booking room.

2. Use-case Diagram



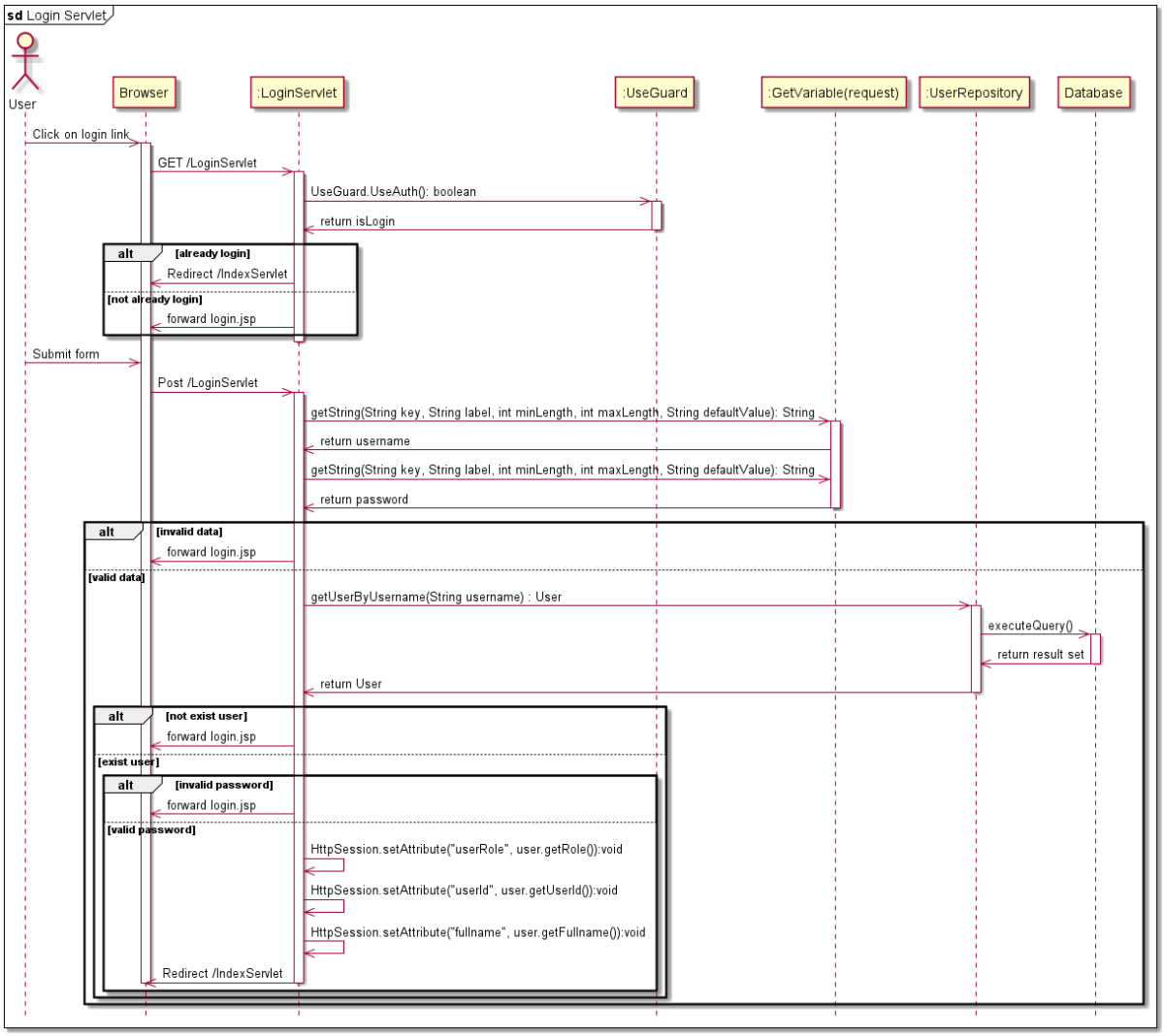
3. Class Diagram



4. Sequence Diagram

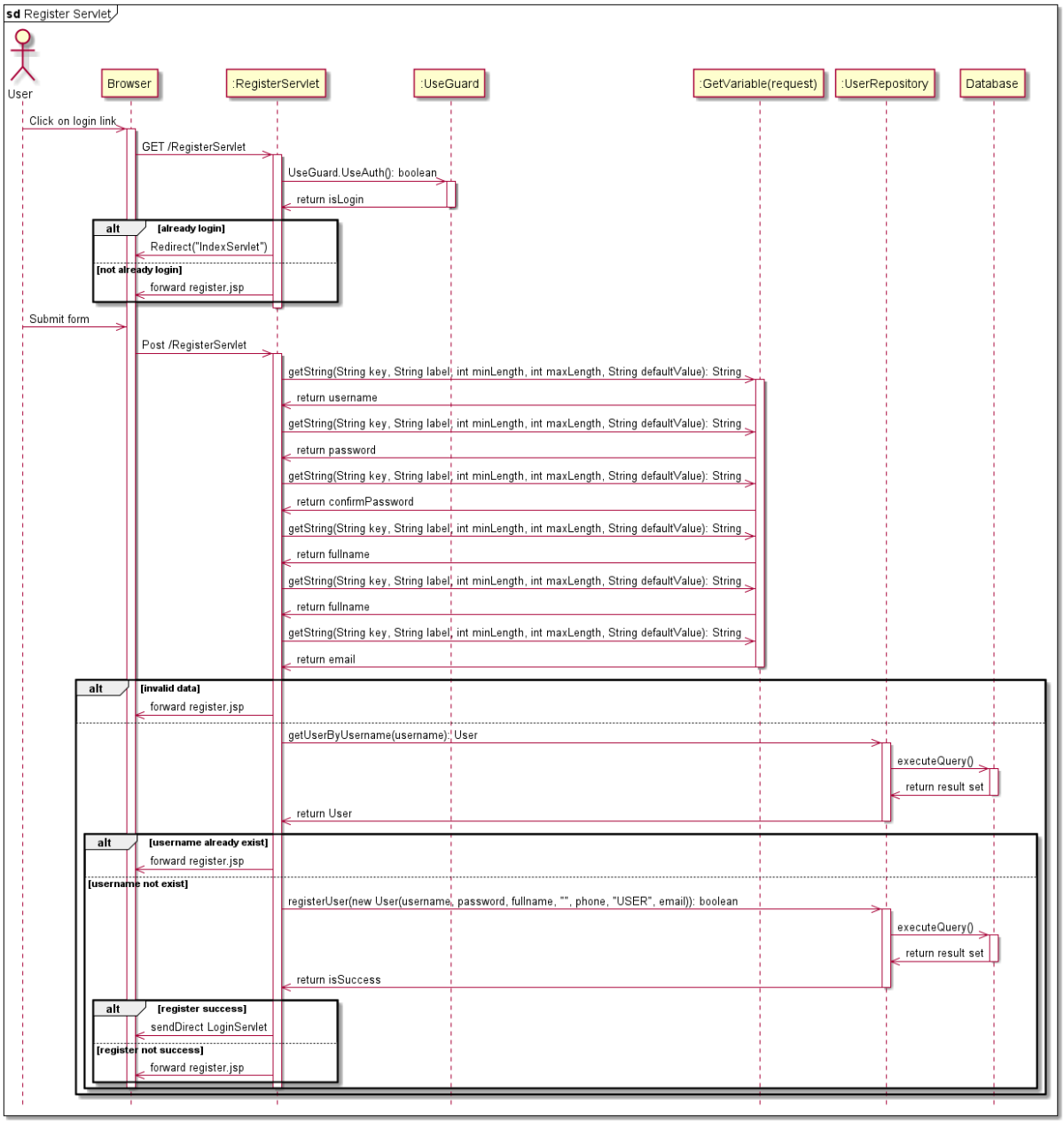
4.1 Login User Account

View full resolution: Here



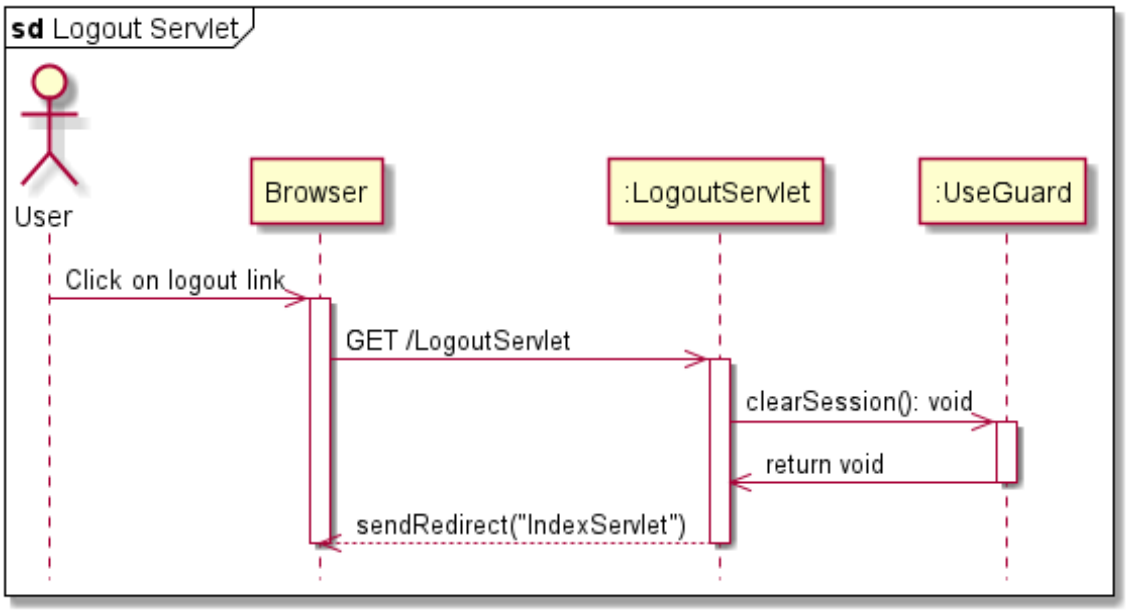
4.2 Register User Account

View full resolution: Here



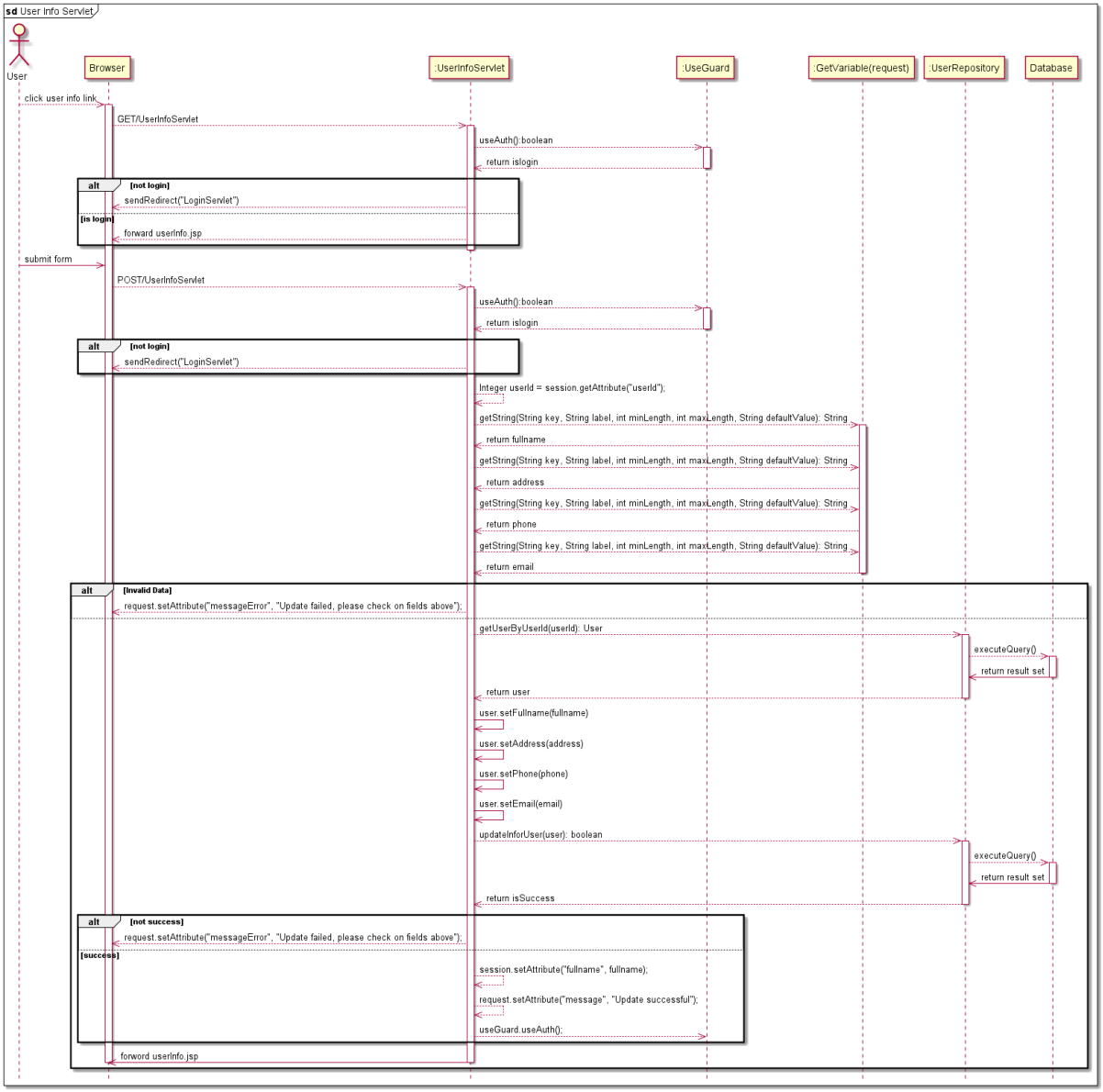
4.3 Logout User Account

View full resolution: Here



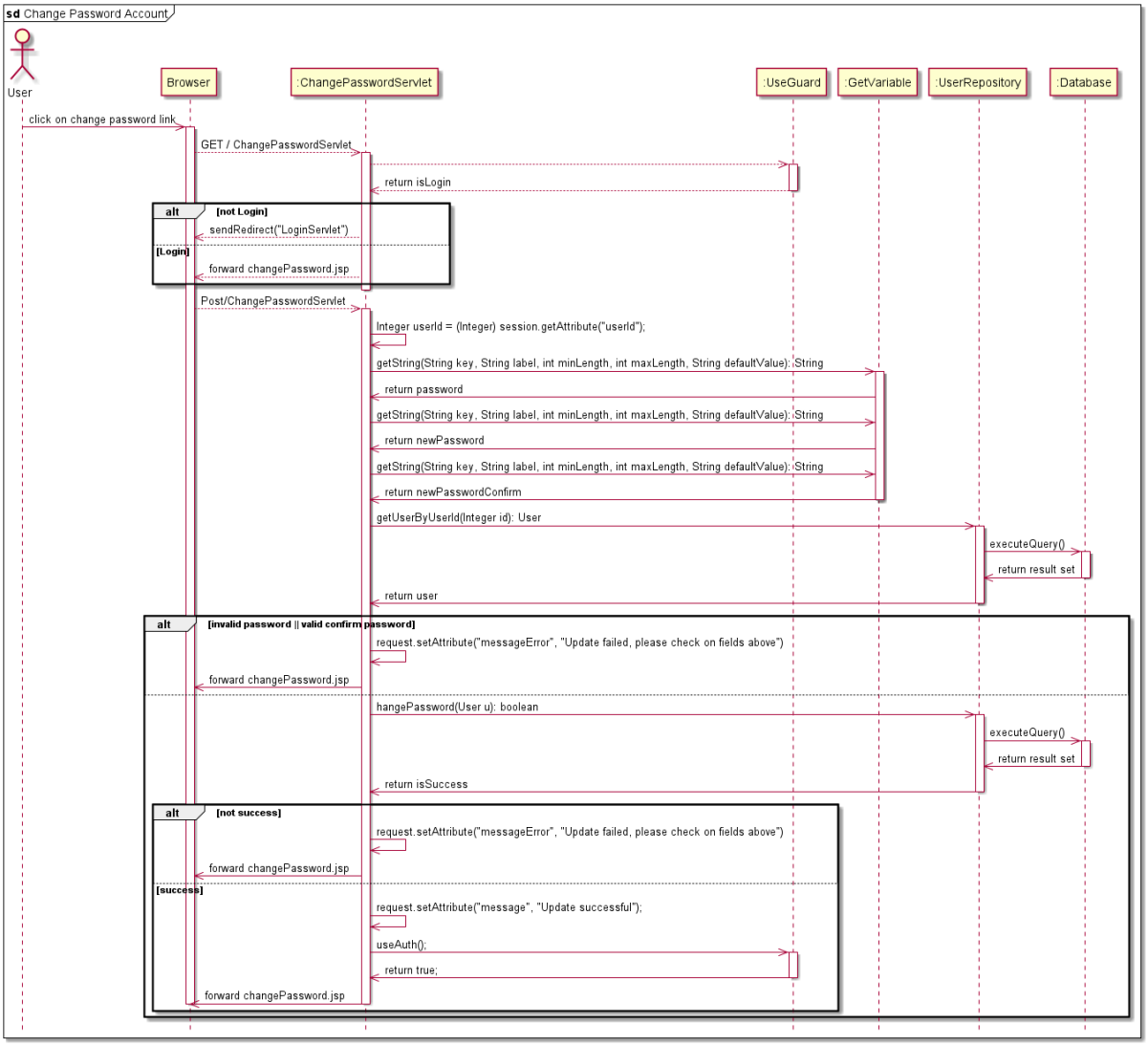
4.4 User Info

View full resolution: Here



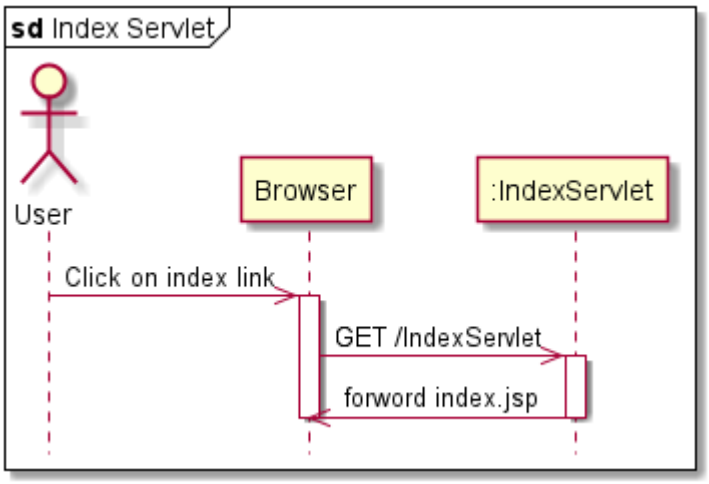
4.5 Change Password User Account

View full resolution: Here



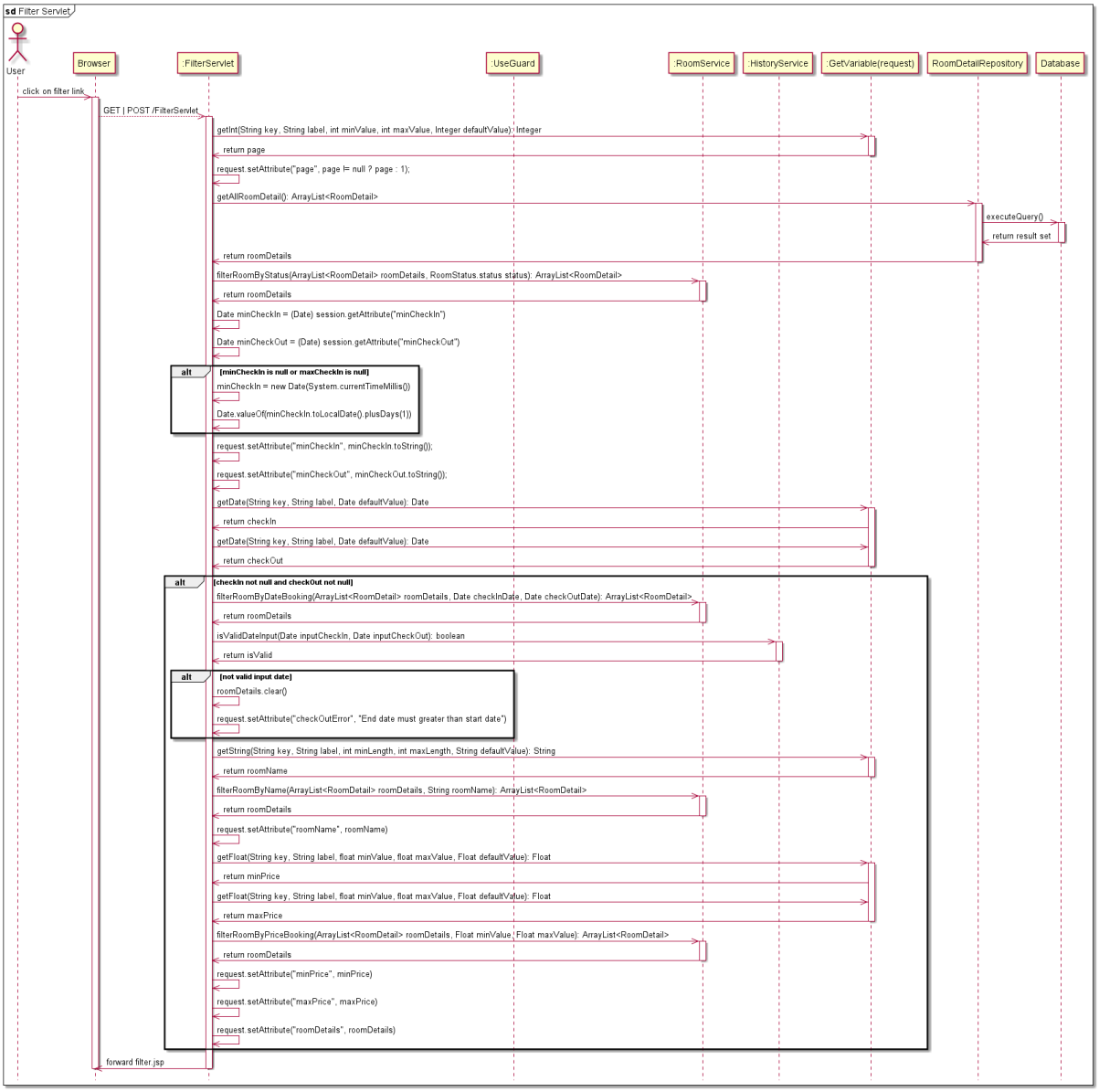
4. 6 Index Page

View full resolution: Here



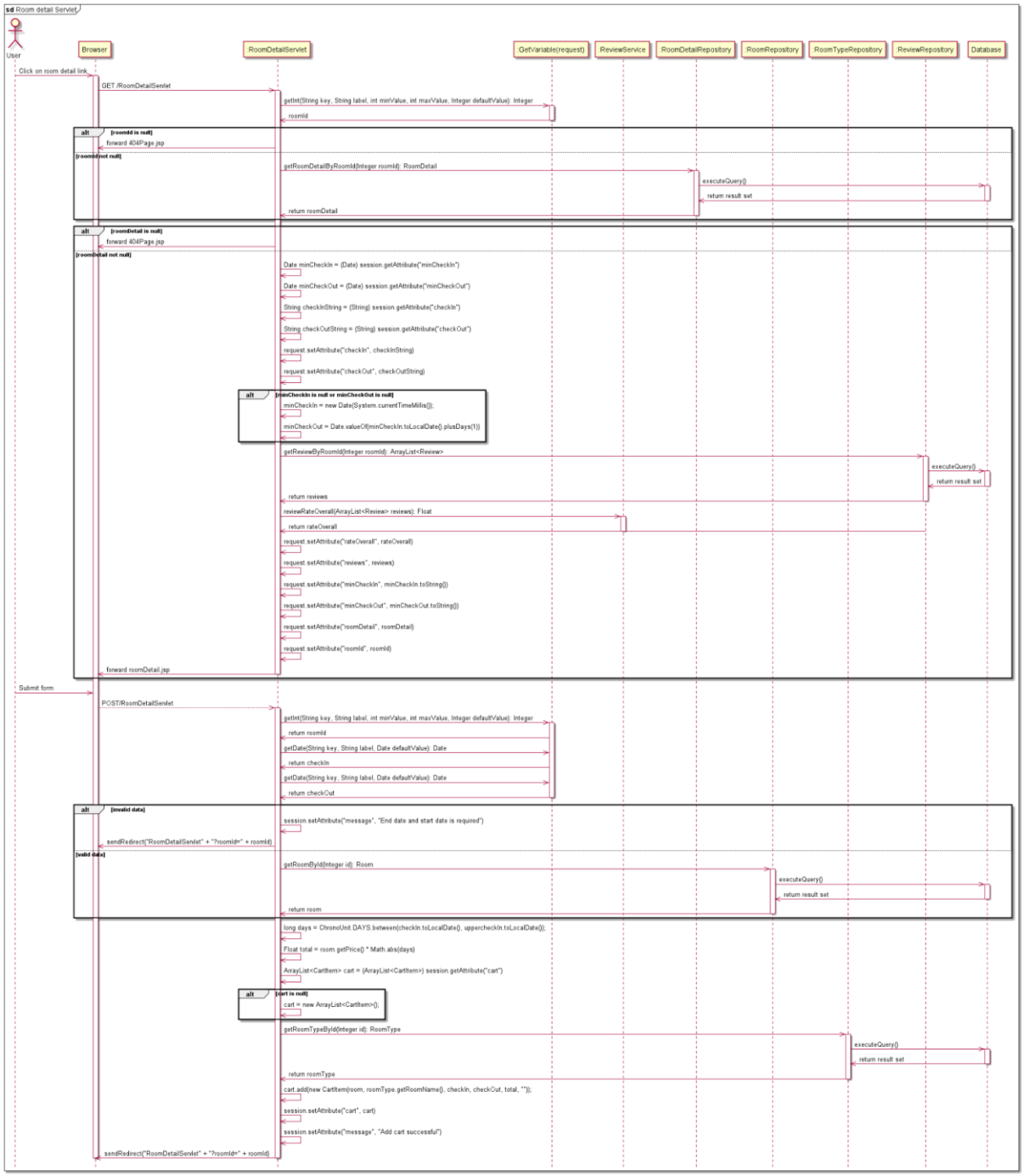
4. 7 Filter Page

View full resolution: Here



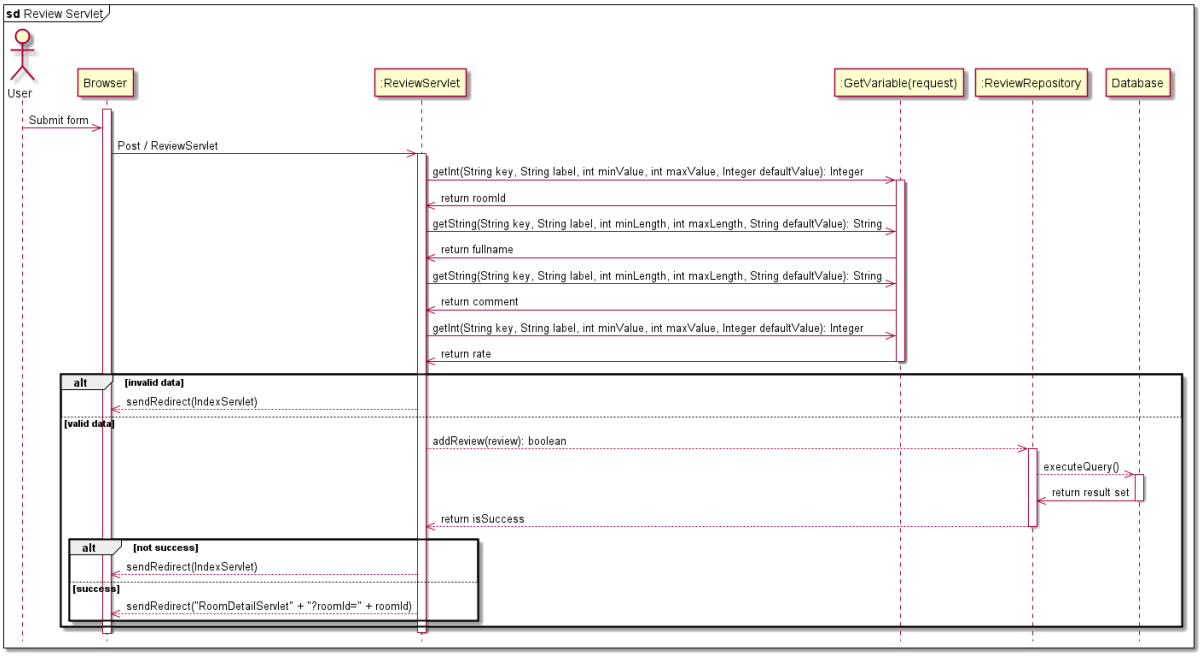
4. 8 Room Detail Page

View full resolution: Here



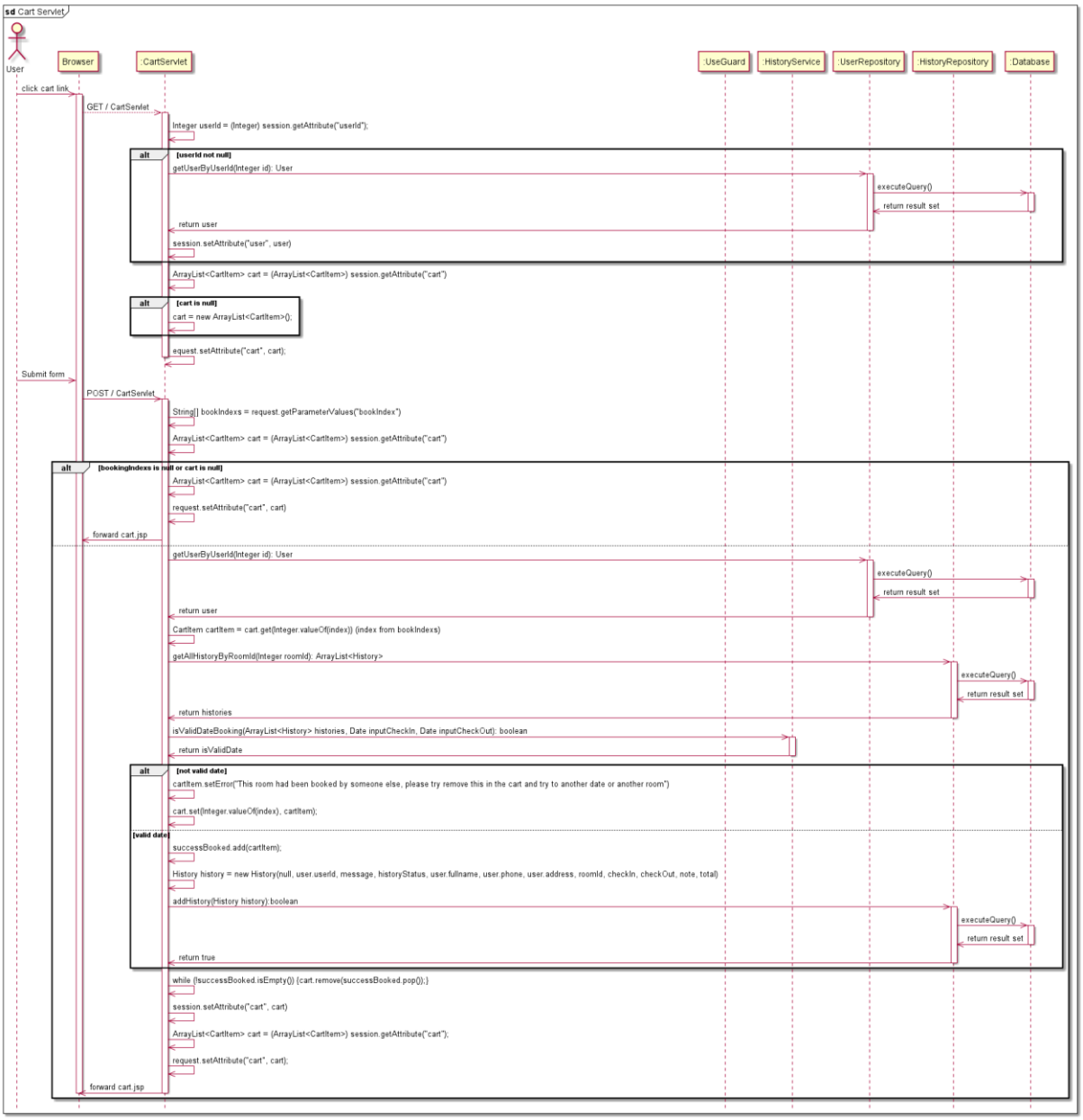
4. 9 Review room by authorization account

View full resolution: Here



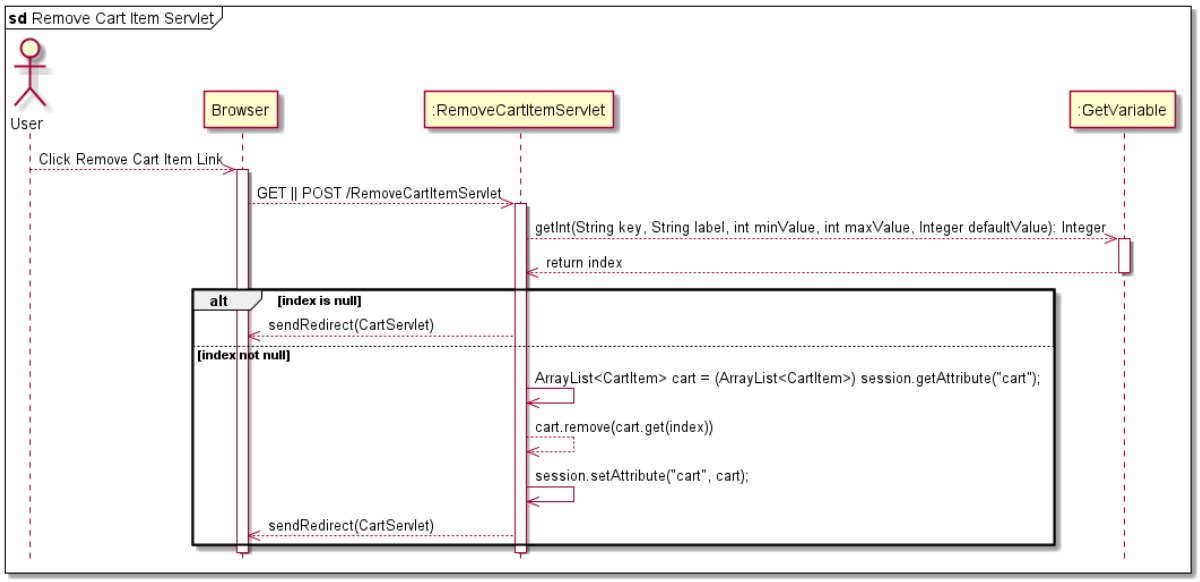
4. 10 Cart Page

View full resolution: Here



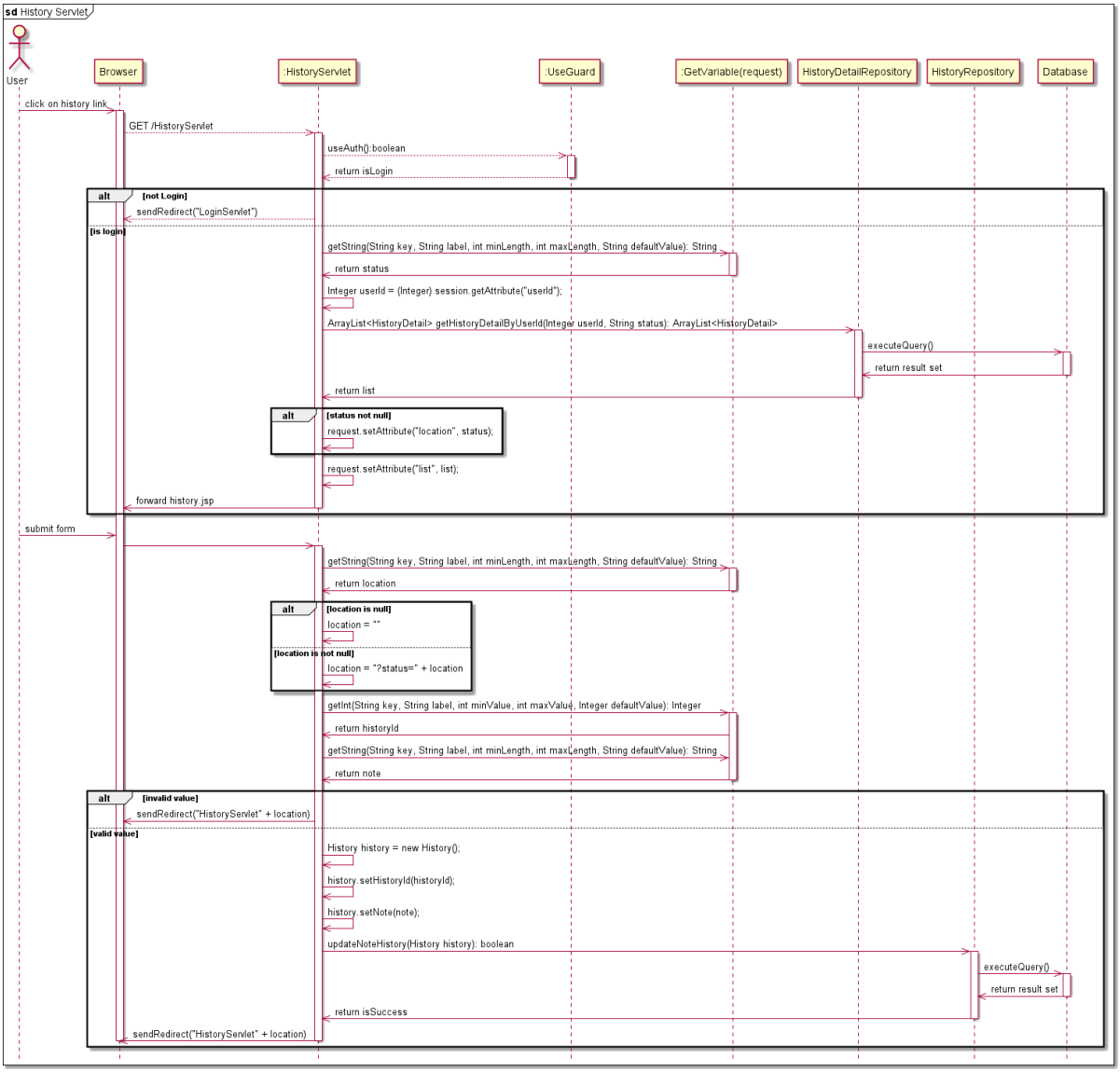
4. 11 Remove Cart Item

View full resolution: Here



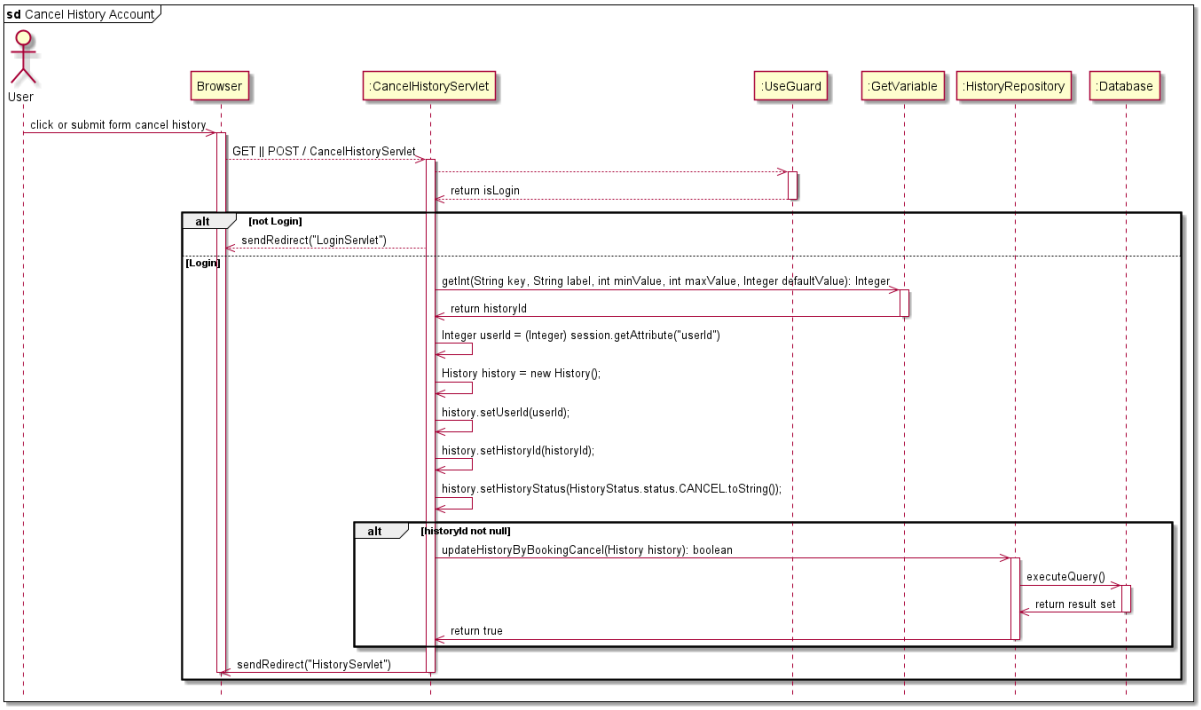
4. 12 View History Page by authorization account

View full resolution: Here



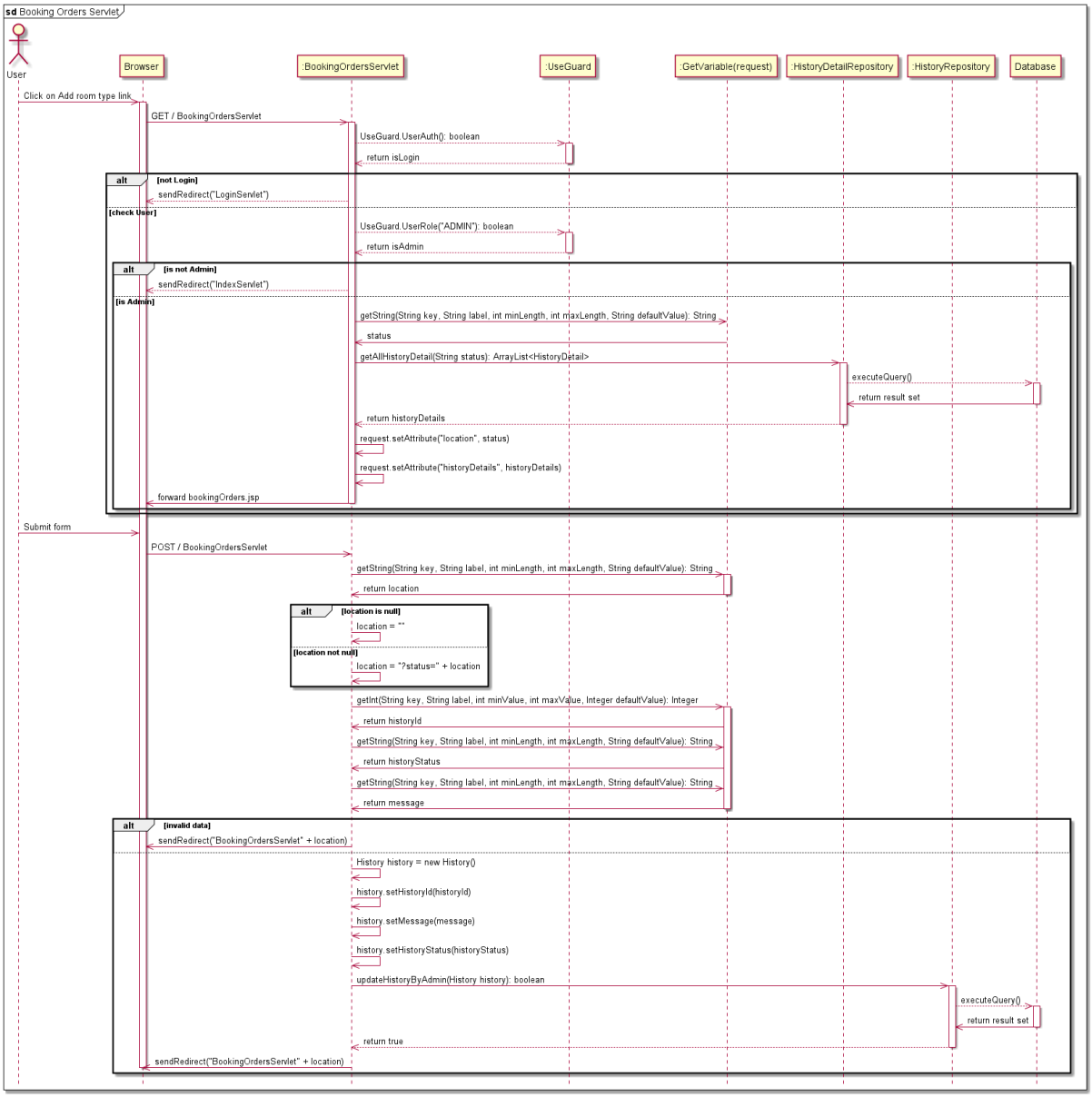
4. 13 Cancel History by authorization account

View full resolution: Here



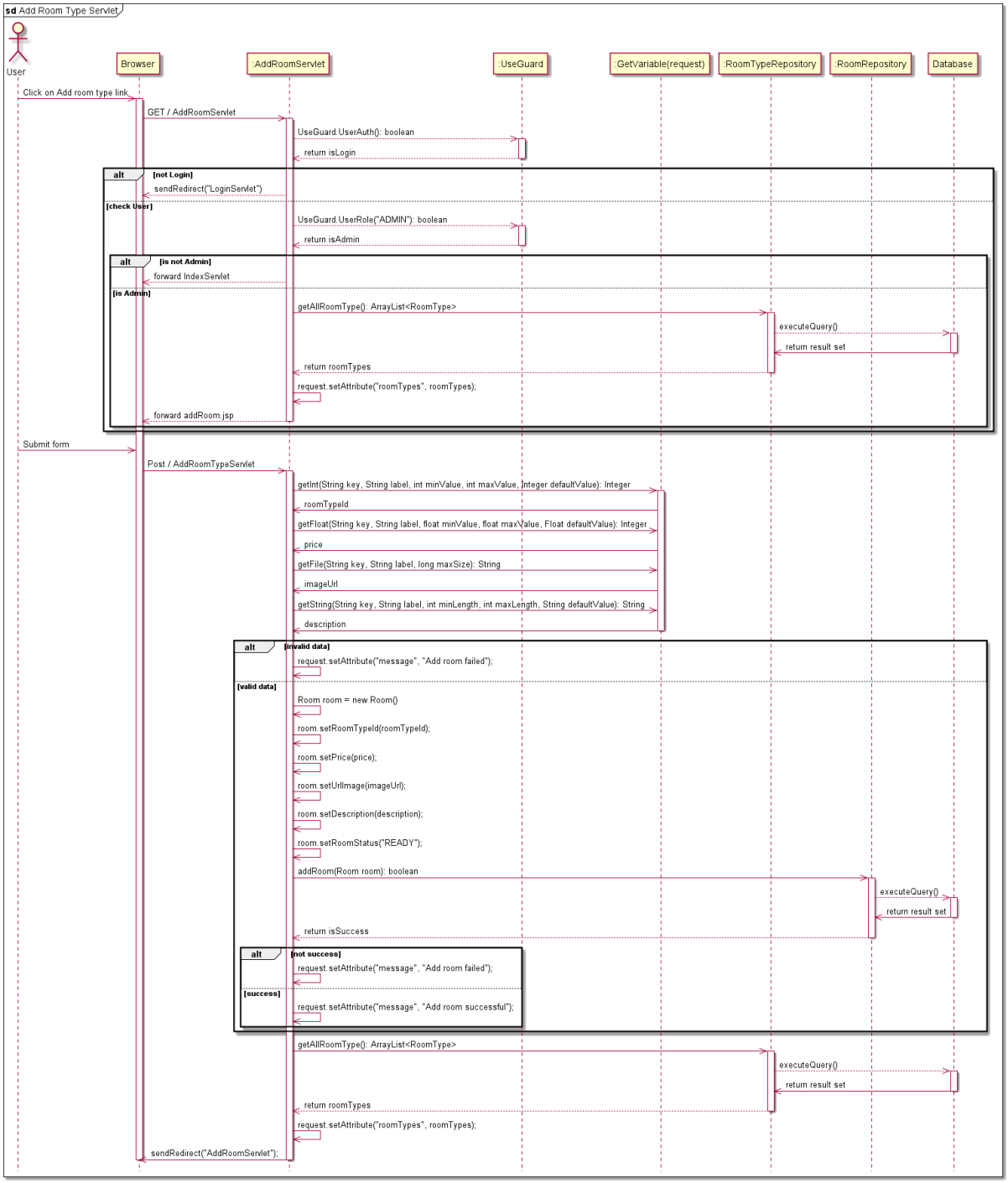
4. 14 Booking Orders by Admin account

View full resolution: Here



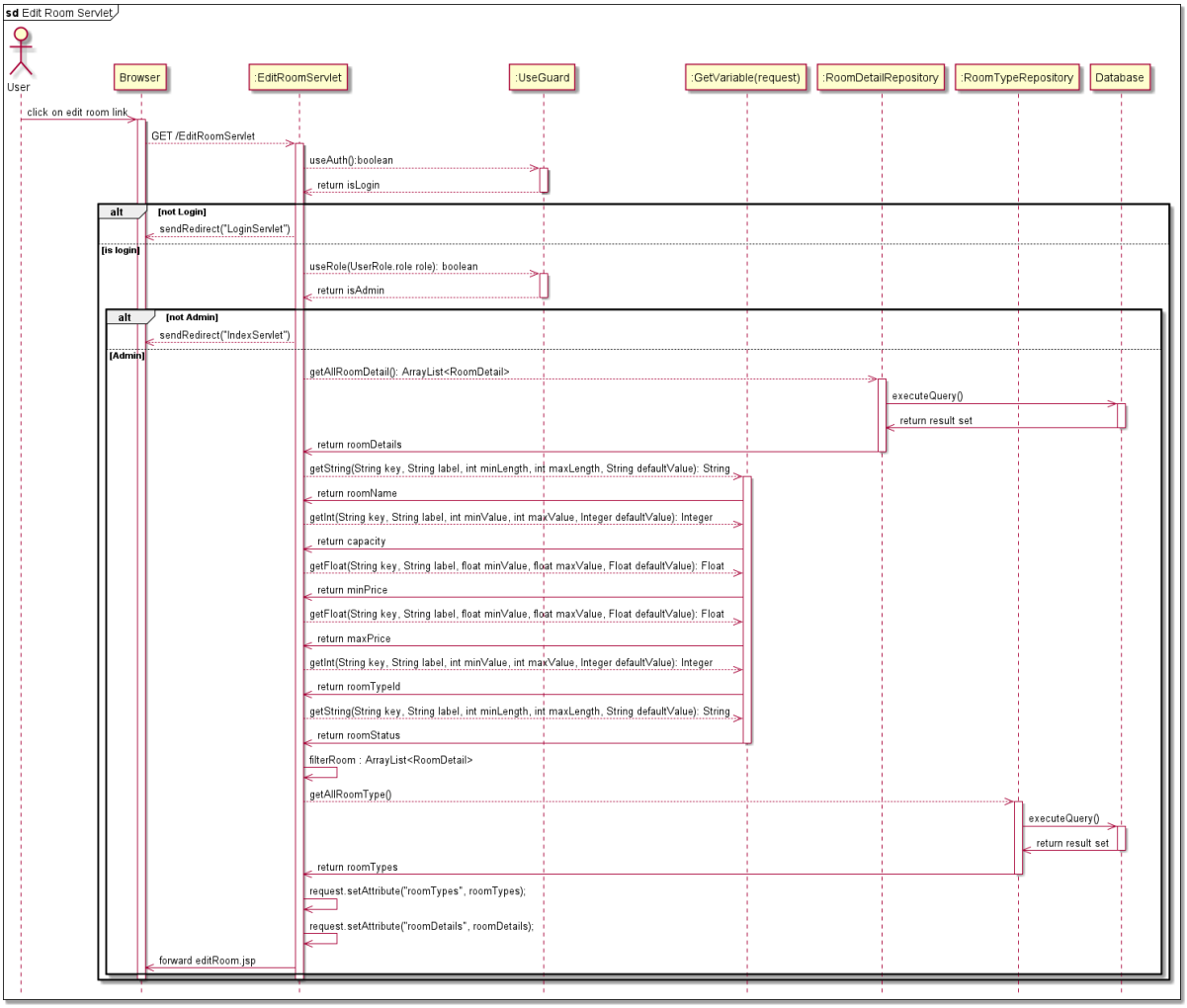
4. 15 Add Room by Admin account

View full resolution: Here



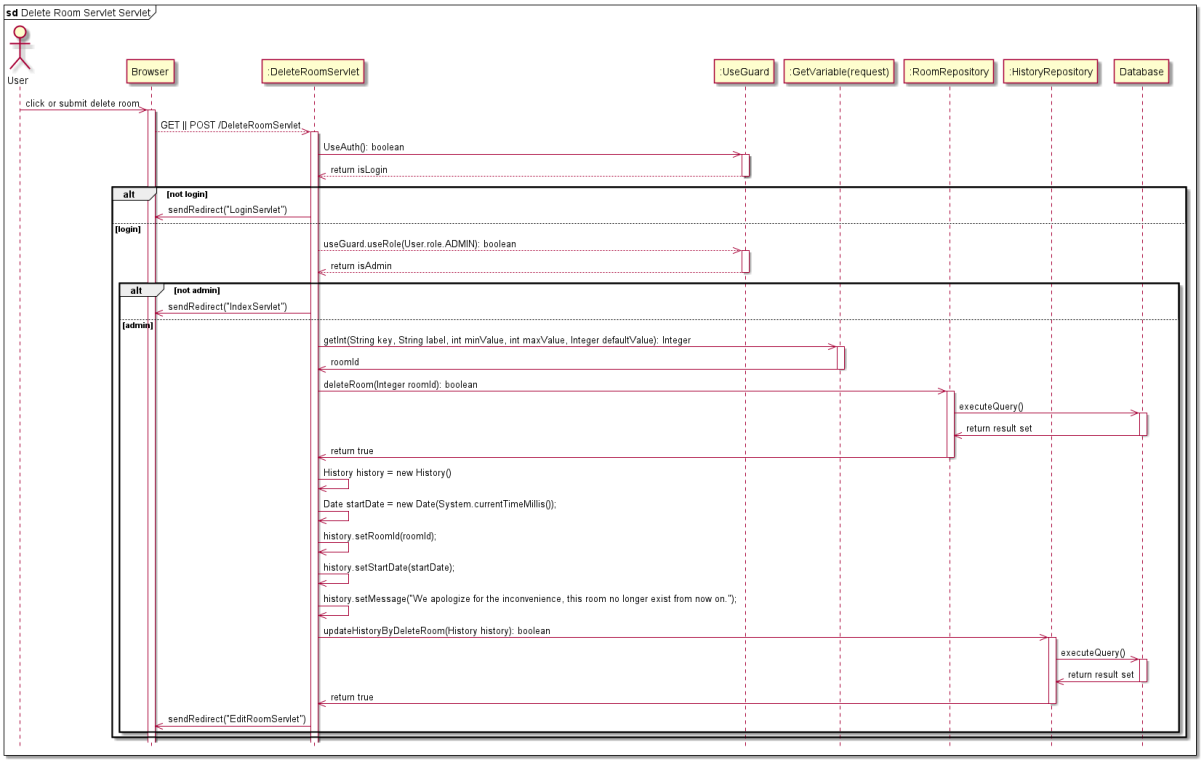
4. 16 Edit Room by Admin account

View full resolution: Here



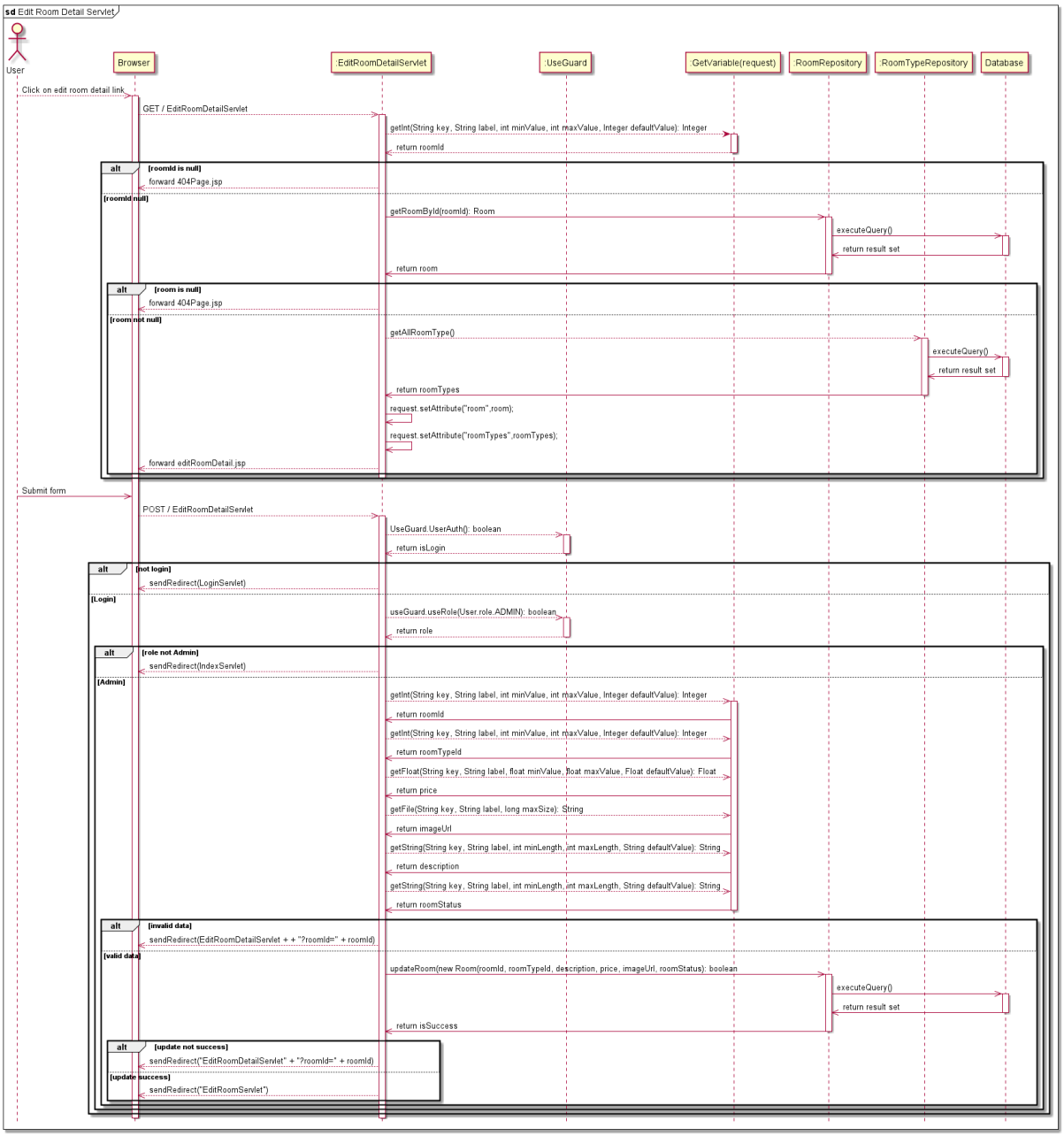
4. 17 Delete Room by Admin account

View full resolution: Here



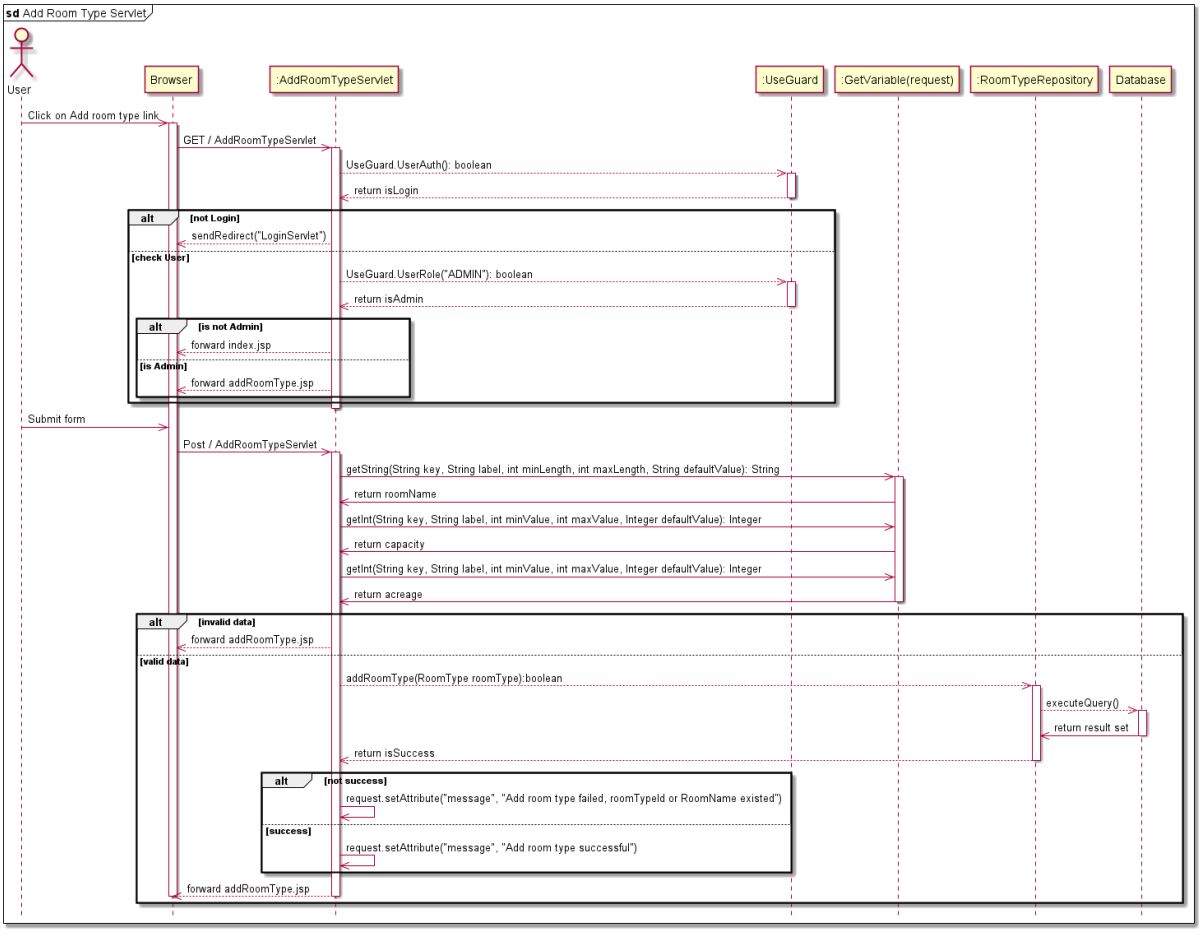
4. 18 Edit Room Detail by Admin account

View full resolution: Here



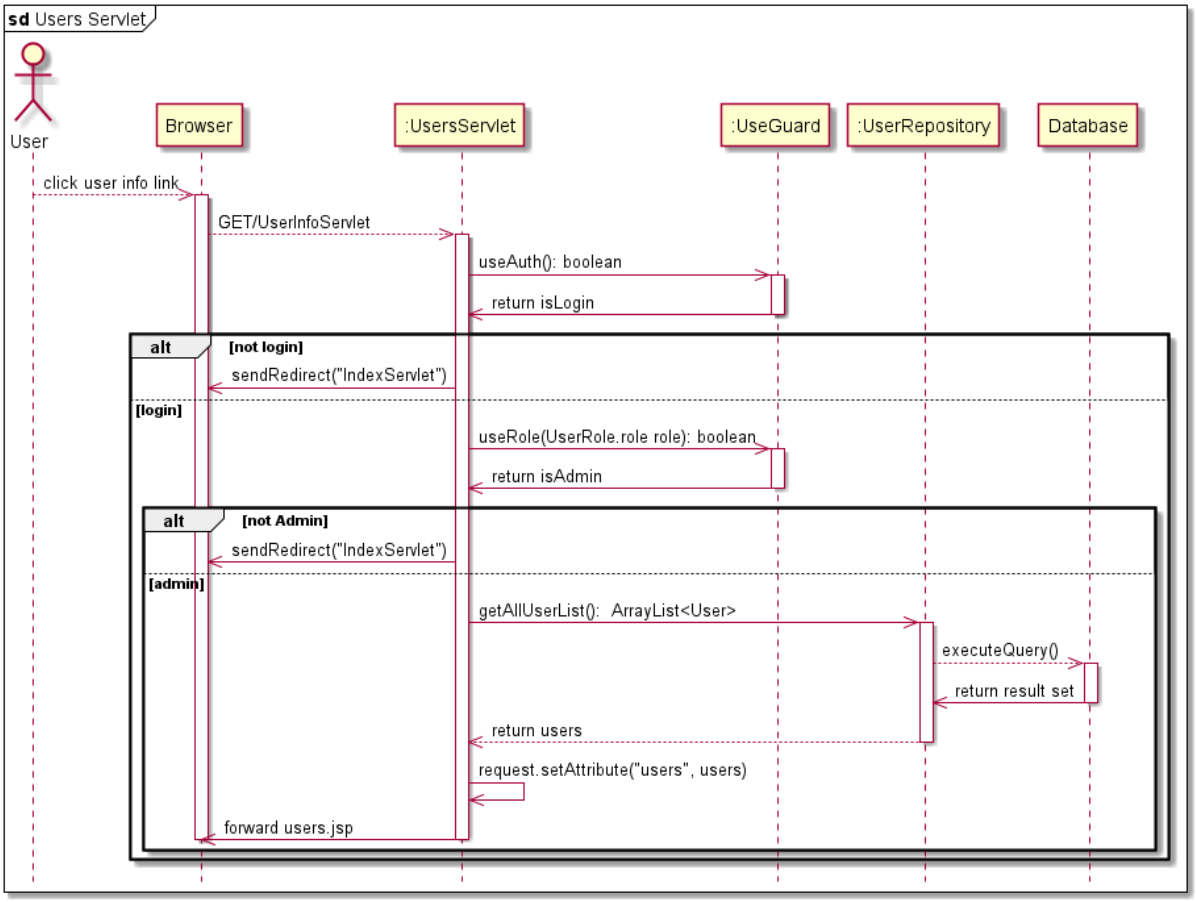
4. 19 Add Room Type by Admin account

View full resolution: Here



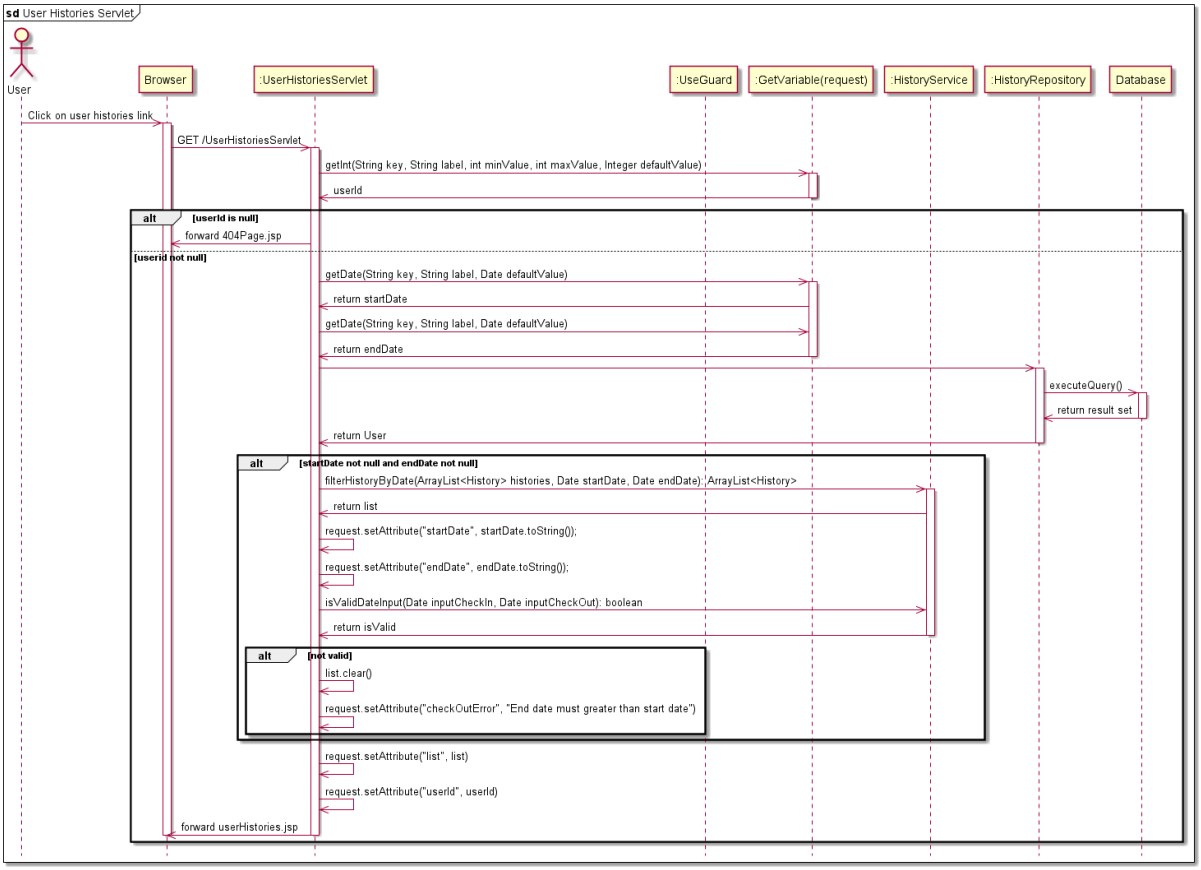
4. 20 View Users by Admin account

View full resolution: Here



4.21 View User Histories by Admin account

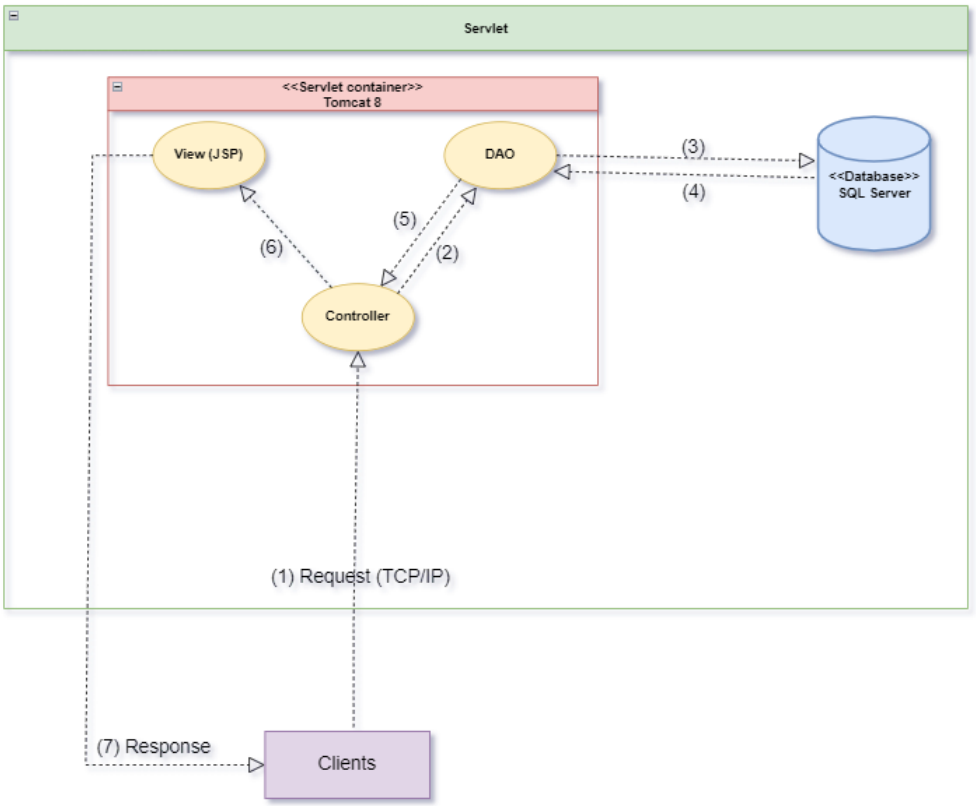
View full resolution: Here



4.22 Add Params Controller

View full resolution: Here

4.23 Deployment Diagram



New technology:

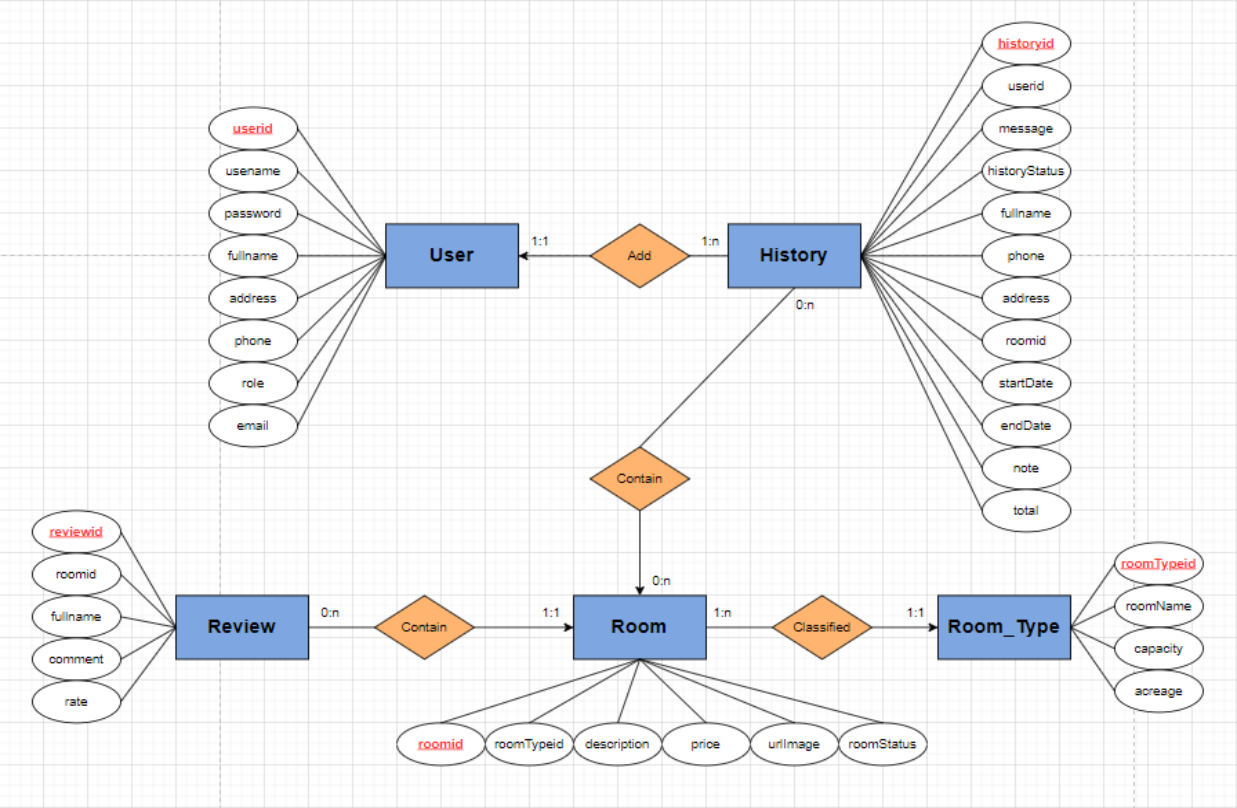
- Tailwind CSS: split every class property into small module classes. That means every class of CSS holds at least one style of CSS property. From that, developers can build UI rapidly without leaving HTML.

II. Database design

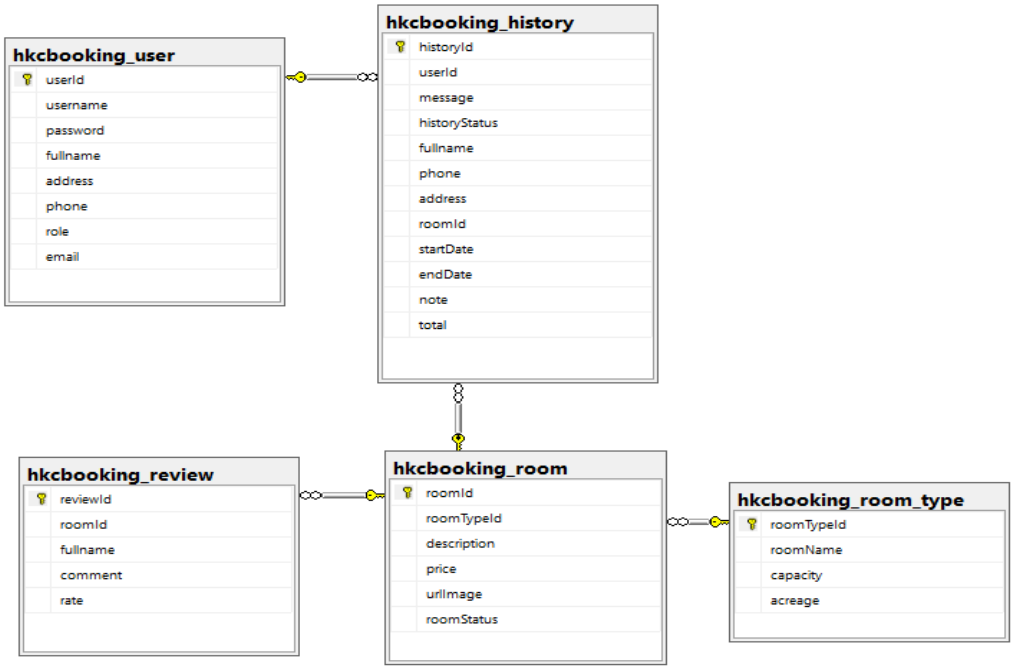
1. Database requirement:

- The hotel includes nearly 200 rooms, each **ROOMS** has room id for each room to identify, Room Type Id, description, price, urlImage, room status. The room id from 1 to 200 to mark for the number room in the hotel. We create Status (Ready, Maintain, Delete) to track the status of each room, convenient for customers to book rooms.
- Room types are varied. Each room is classified into **ROOM TYPES**. Each room type has a room type id, room name, capacity and area of each room in accordance with the choice of each customer.
- Each customer when creating an account to use our services, we will let **USERS** create username, password, fullname, address, phone, email for easy management. The role is either User or Admin. Admin can add a new room type.
- Users can book 1 room or many rooms depending on their usage requirements, at different times. And we also track the dates that customers book and check out to match other customers' orders. **HISTORY** has historyId, userId, message, historyStatus, fullname, phone, address, roomId, startDate, endDate, note, total will save User's booking information.
- Users can rate rooms after they check out. Each **REVIEW** has a reviewId, roomId full name, comment, rating. We track the creator of this review, the date it was created, and from there learn more to develop a better service.

2. Entity Relationship Diagram

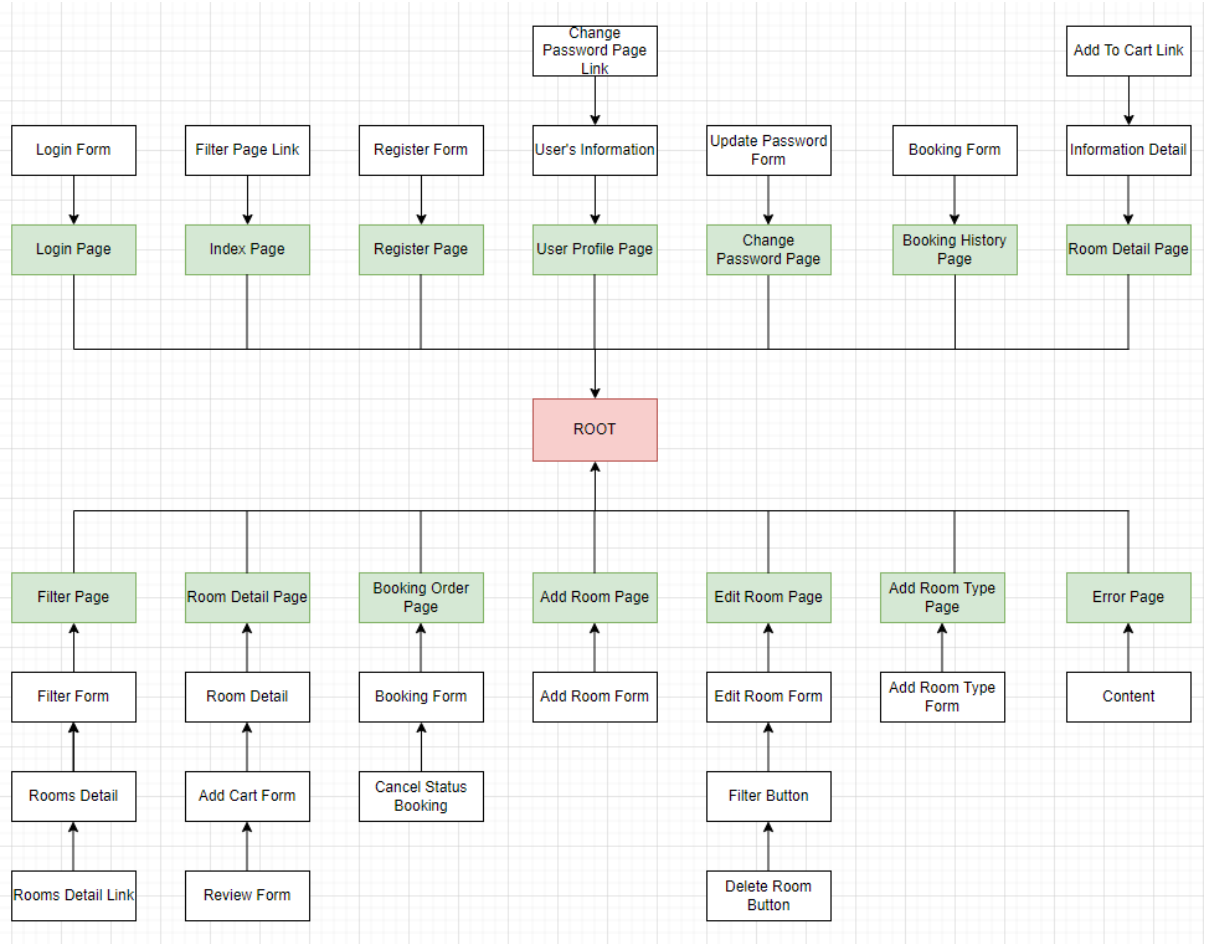


3. Relationship diagram



III. Project Site Map and Explanation

1. Site Map



2. Explanation

2.1 Login Page

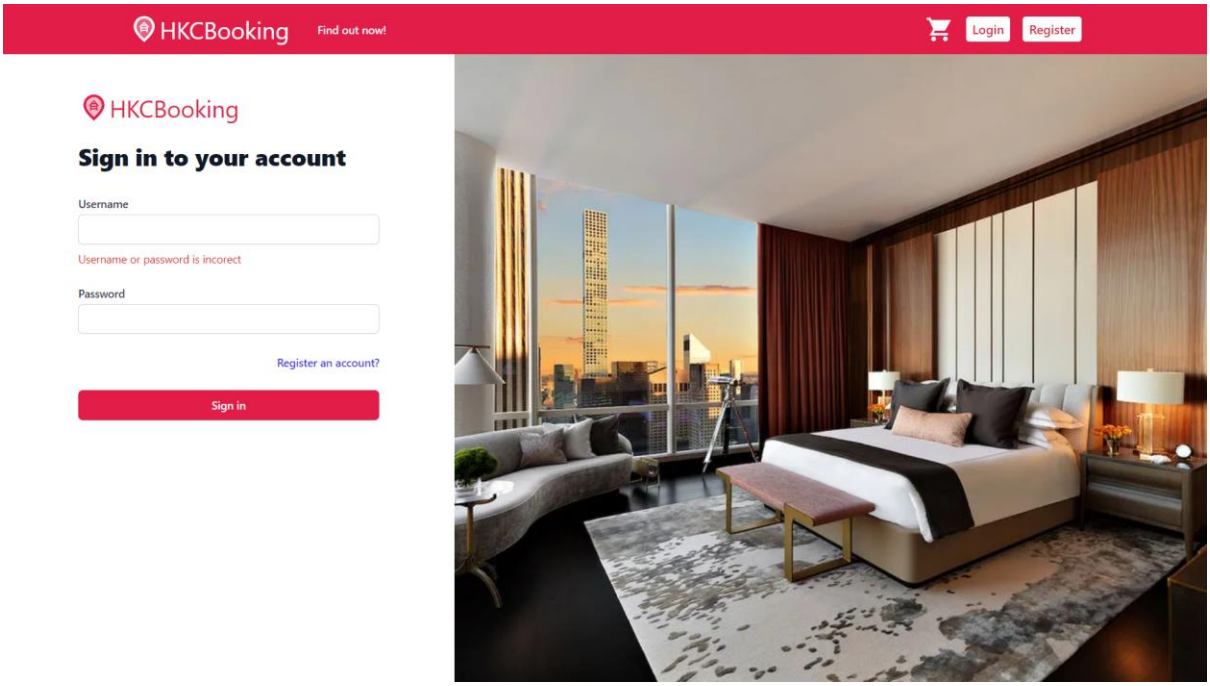
a. Description

This is the default website, users need to log in to act, if you don't have an account, you will register in the box next to it.

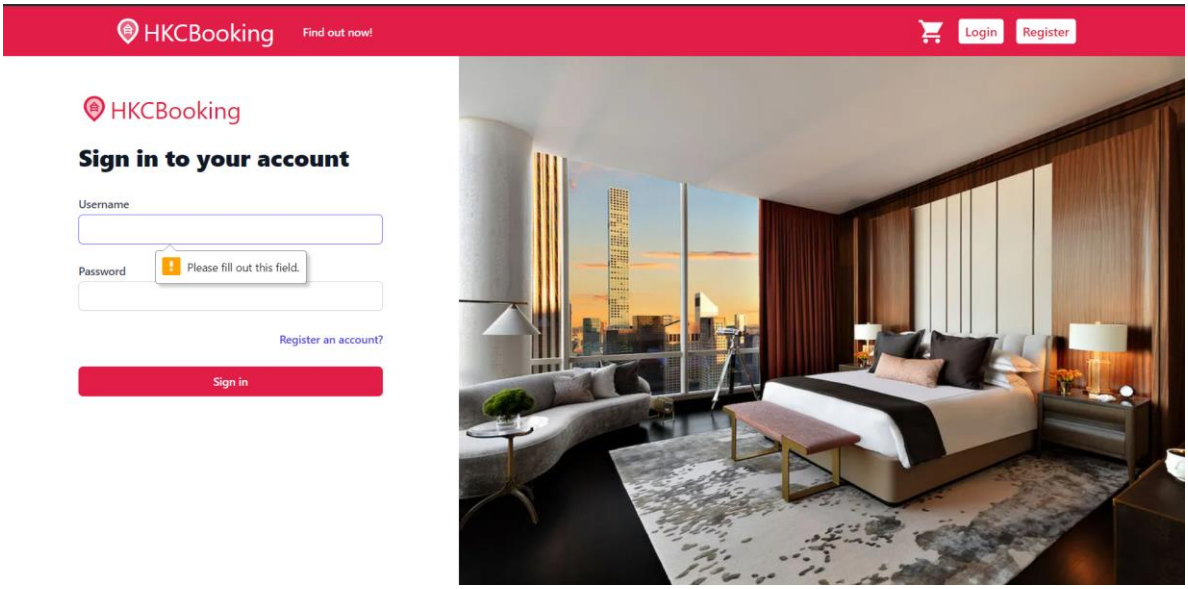
b. Validation

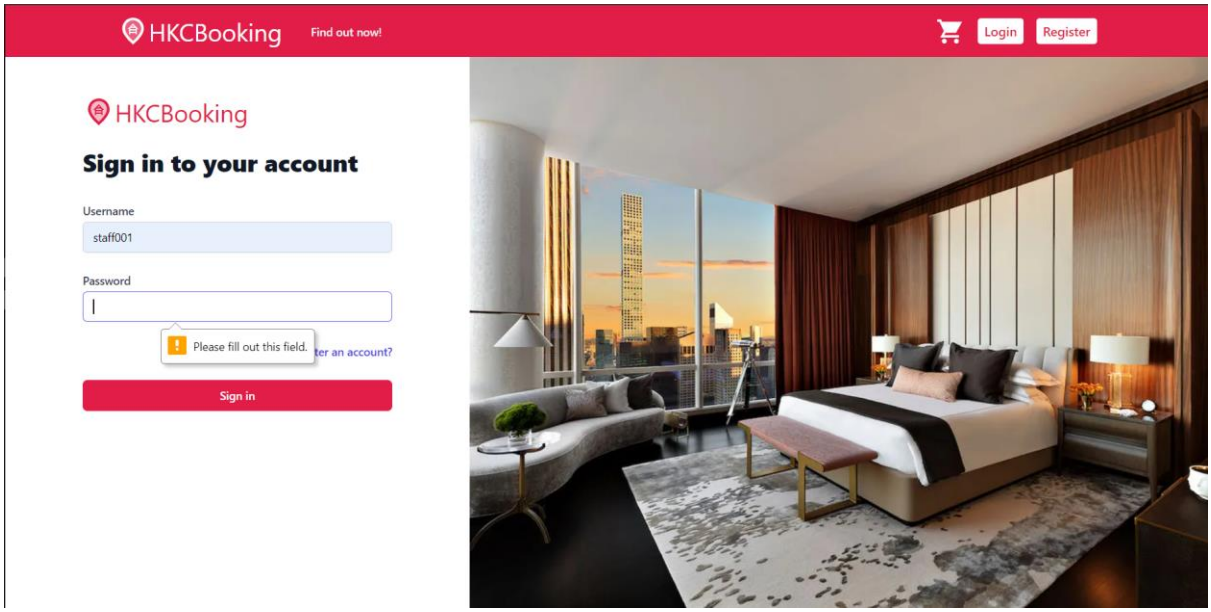
Login Form					
Field	**Type**	**Min**	**Max**	**isRequired**	**Description**
username	String	8	30	Not Null	
password	String	8	30	Not Null	

If the username and password do not match the username and password in the database, an error message will be displayed "Username or password is incorrect"

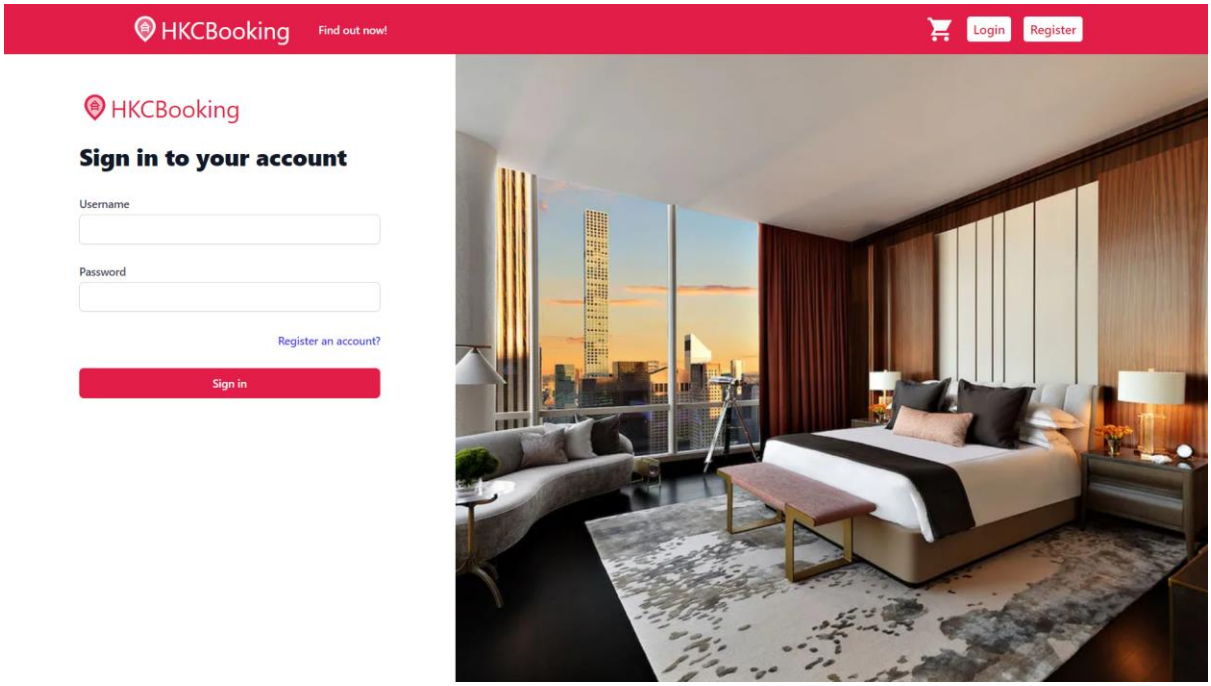


Username & password can't null





c. Accomplishment



2.2 Register page

a. Description

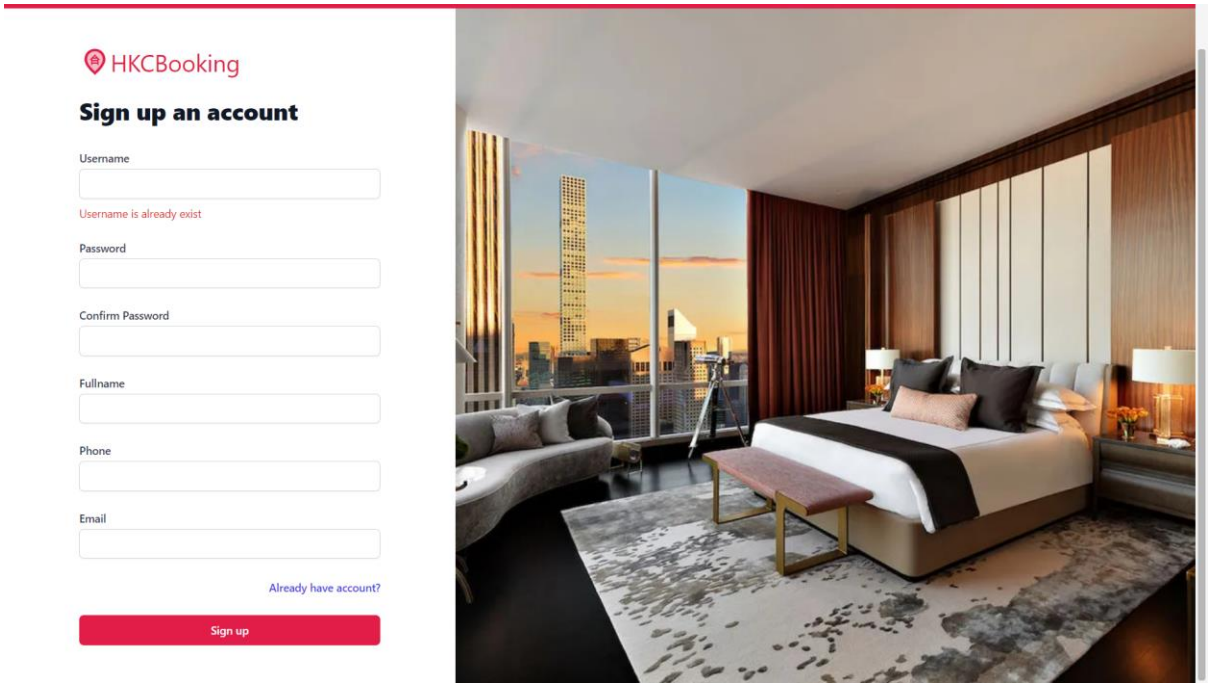
This is a page for those who do not have an account to log in, when creating an account, users will not be able to role-play, in order to ensure the safety of the website. All information entered by the user will be saved in the database

b. Validation

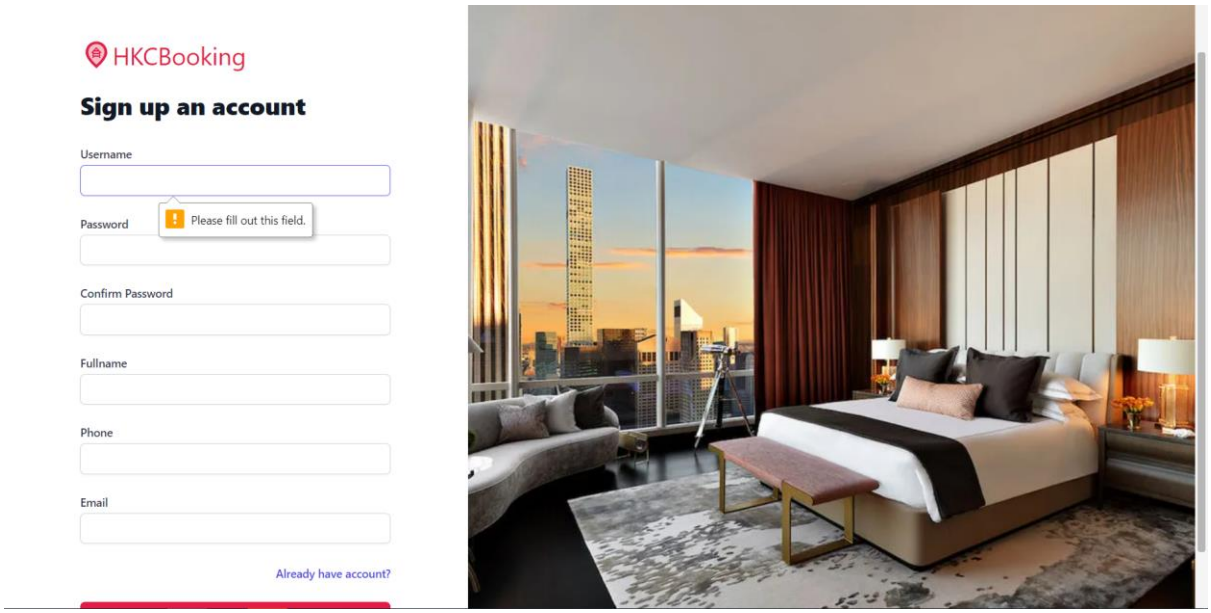
Register Form					
Field	**Type**	**Min**	**Max**	**isRequired**	**Description**
username	String	8	30	Not Null	Must unique
password	String	8	30	Not Null	
Confirm password	String	8	30	Not Null	Equal password
Full name	String	0	50	Not Null	
Phone	String	10	10	Not Null	Correct Number format in Viet Nam

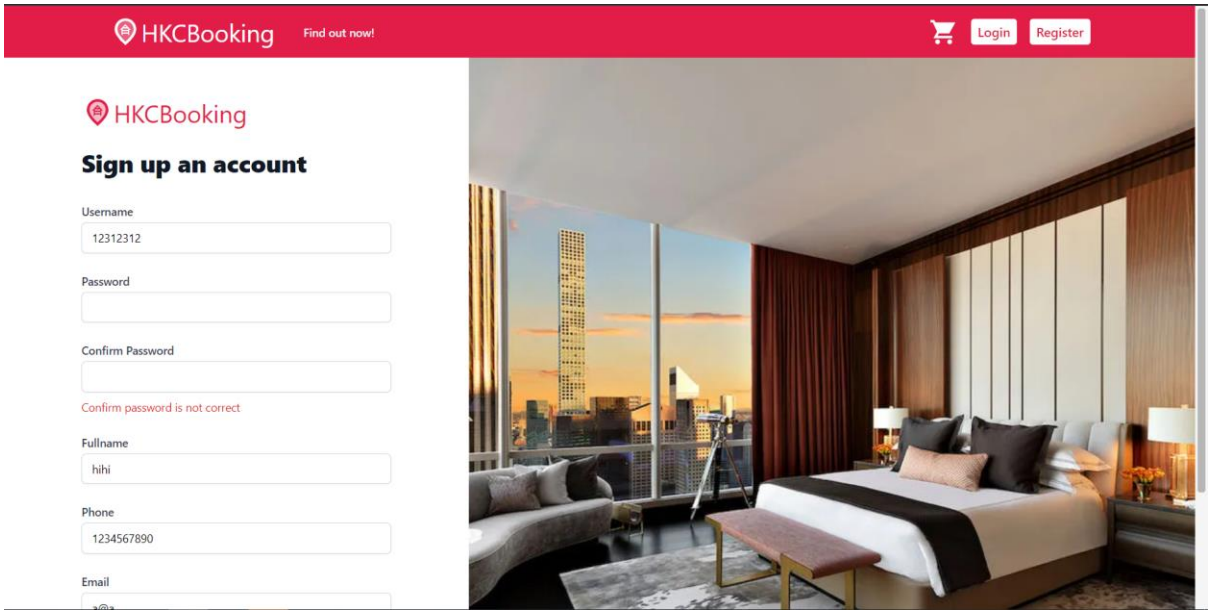
Email	String	0	50	Not Null	Correct Format Email

If username is already exist an error message will be displayed "Username is already exist"



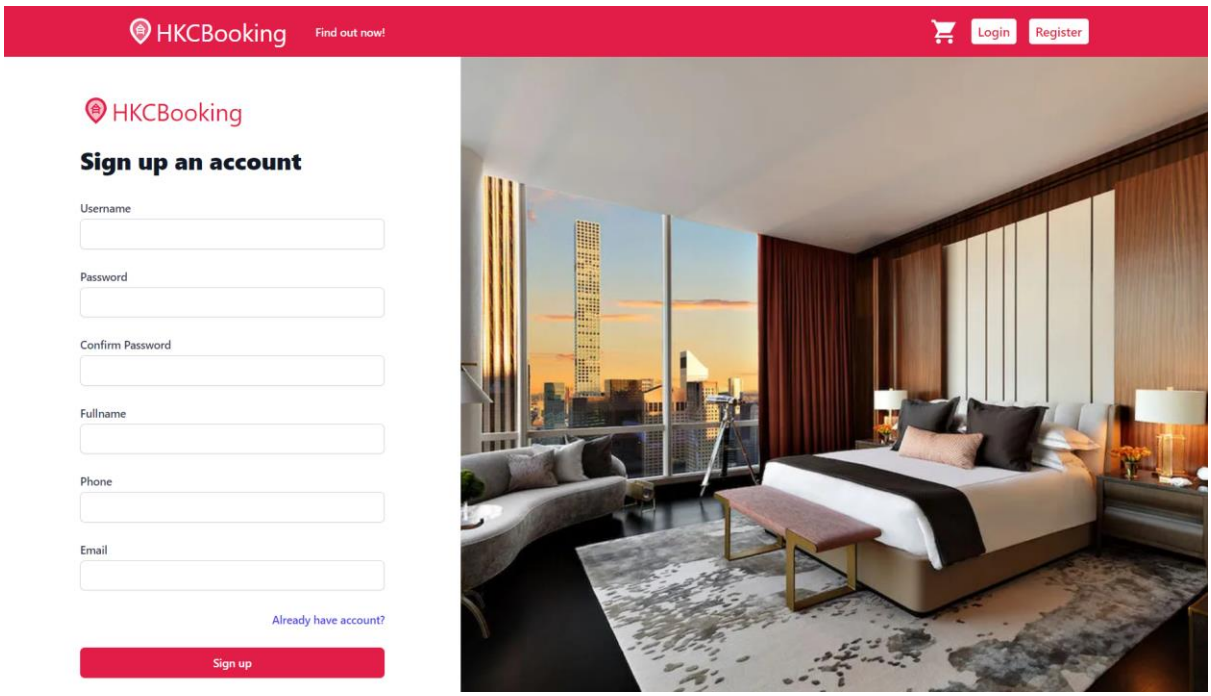
Username can't not null





Password is not correct with confirm password

c. Accomplishment



2.3 Filter page

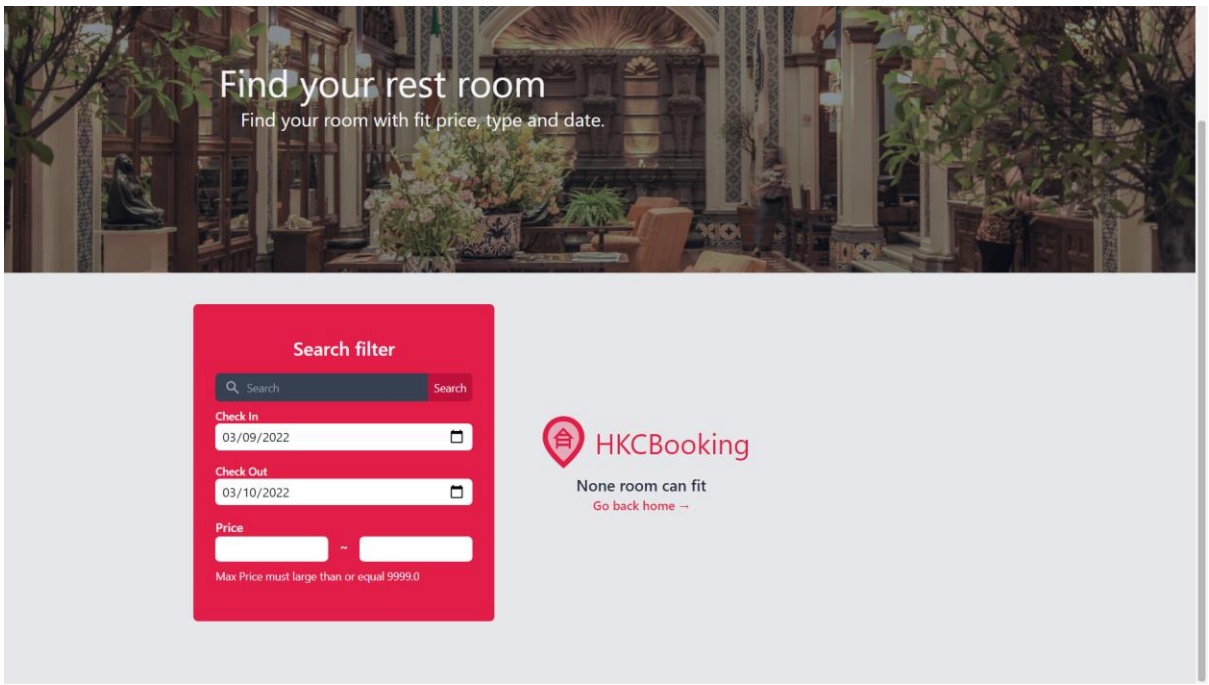
a. Description

This site helps users choose their favourite apartments and suits their personality

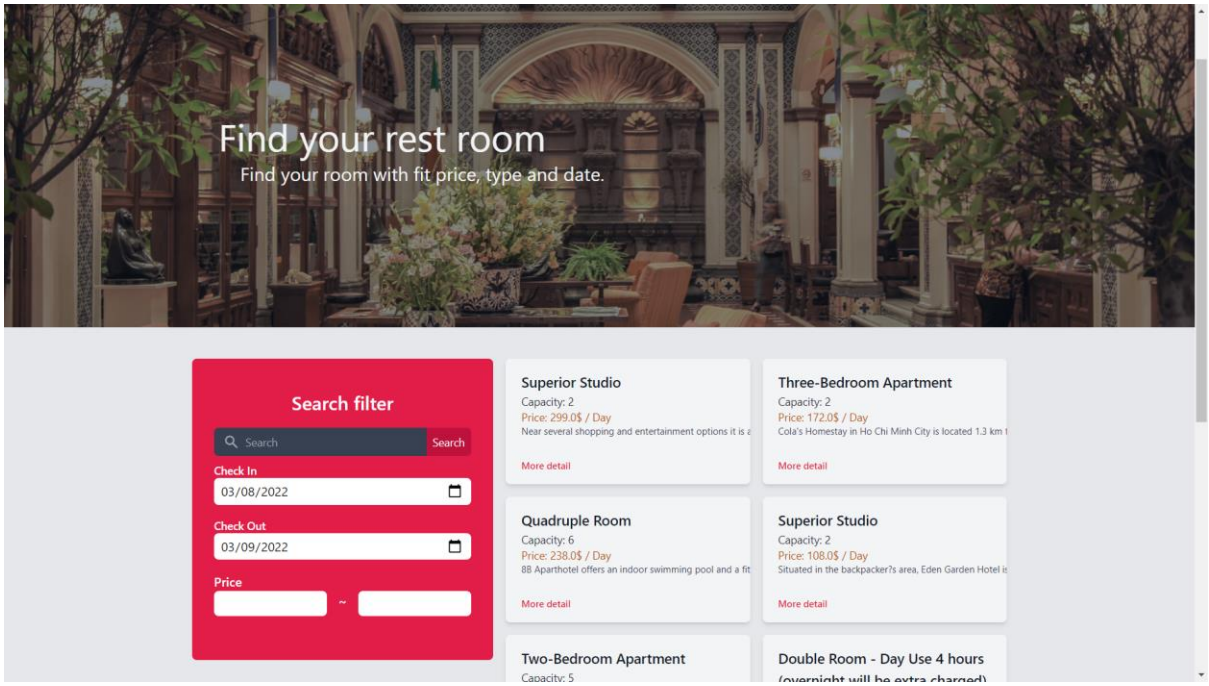
b. Validation

Filter Form					
Field	**Type**	**Min**	**Max**	**isRequired**	**Description**
roomName	String	0	200	No	
Checkin	Date			No	Default date is current day
Check out	Date			No	Must greater than current day
Lowest Price	Float	0	max	No	Default value 0
Highest Price	Float	0	max	No	Default value 9999

If lowest price greater than highest price an error message will be displayed "Max Price must large than or equal 9999.0"



c. Accomplishment



2.4 User Information page

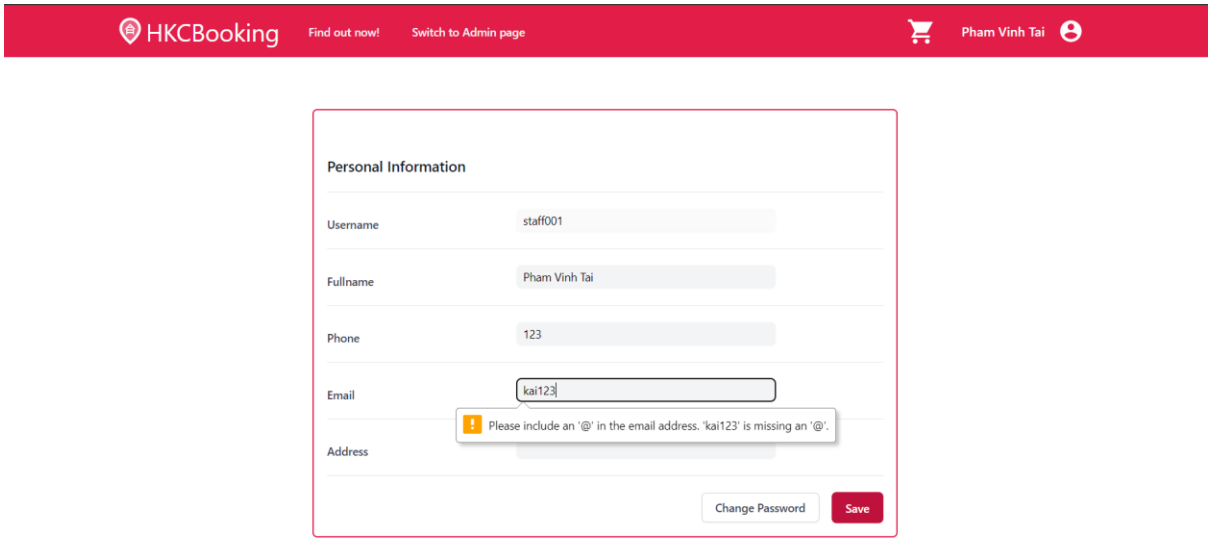
a. Description

This page will print out the user's information, the user can edit the above information except the password for safety. If you want to change your password, select the change password box below

b. Validation

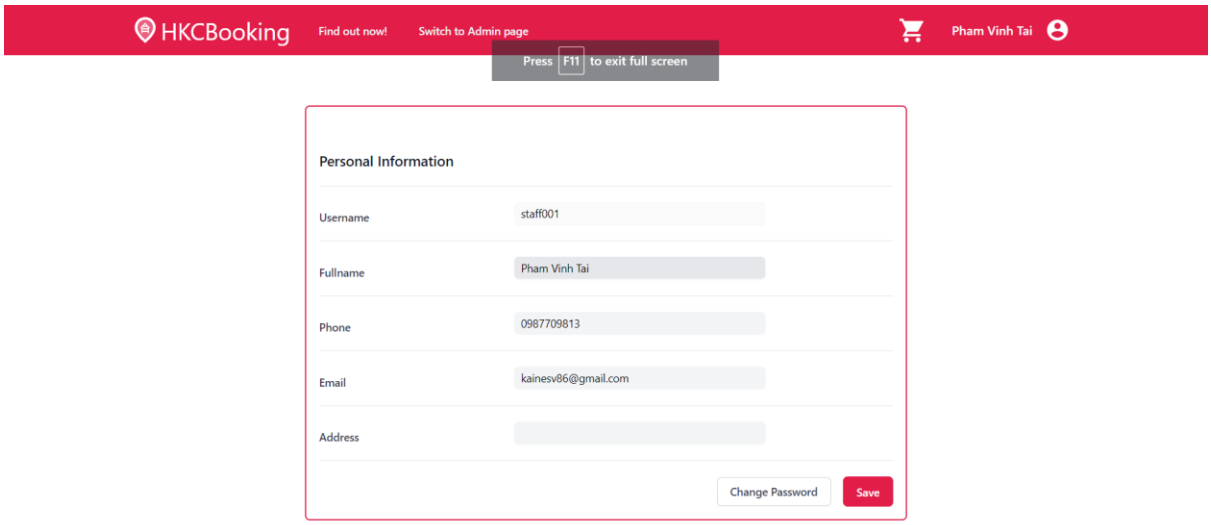
User information form					
Field	**Type**	**Min**	**Max**	**isRequired**	**Description**
Username	String	8	30	Not Null	unique
fullName	String	0	50	Not Null	
Phone	String	10	10	Not null	Correct Number format in Viet Nam
Email	String	0	100	Not null	Correct format email
Address	String	0	50		

The username part is for viewing only and cannot be edited



Wrong format will displayed error

c. Accomplishment



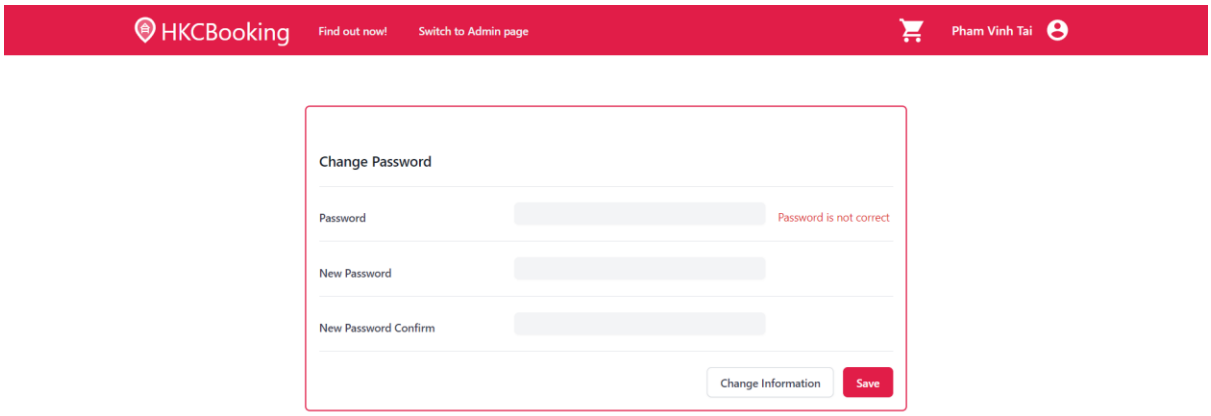
2.5 Change password page

a. Description

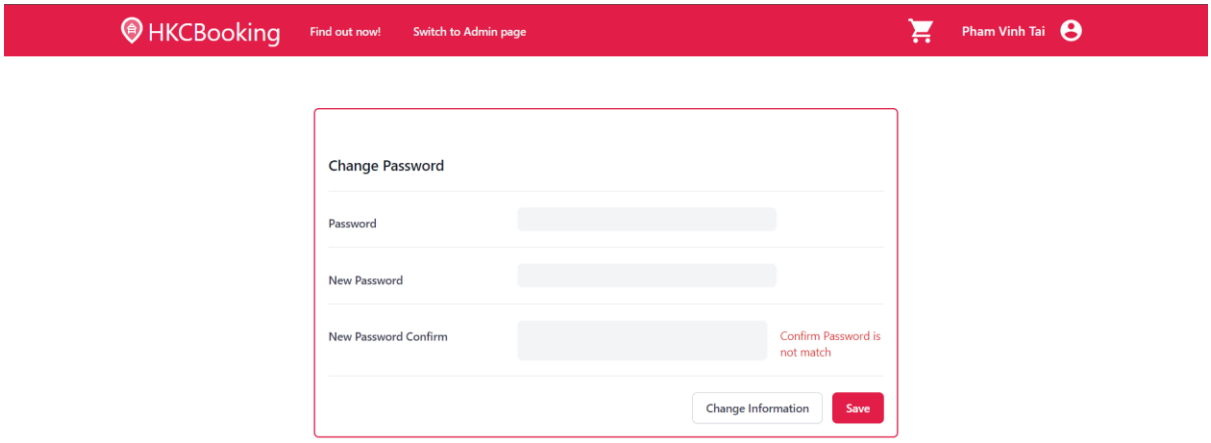
This page is used to change the password, if you want to change the password, then enter the old password and fill in the new password

b. Validation

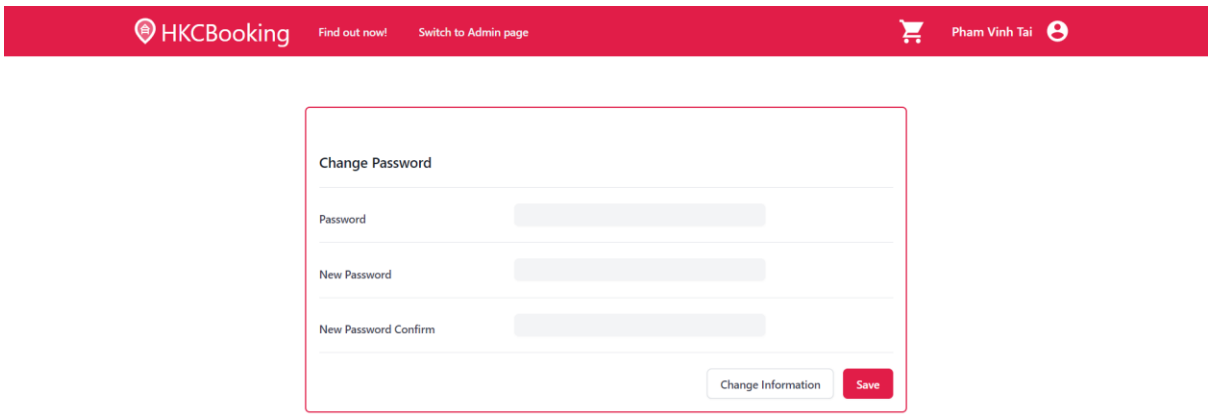
Change password form					
Field	**Type**	**Min**	**Max**	**isRequired**	**Description**
password	String	8	30	Not null	Same password of this user in database
New password	String	8	30	Not null	
New password confirm	String	8	30	Not null	Must equal new password



Confirm password not correct with new password



c.Accomplishment



2.6 Room detail page

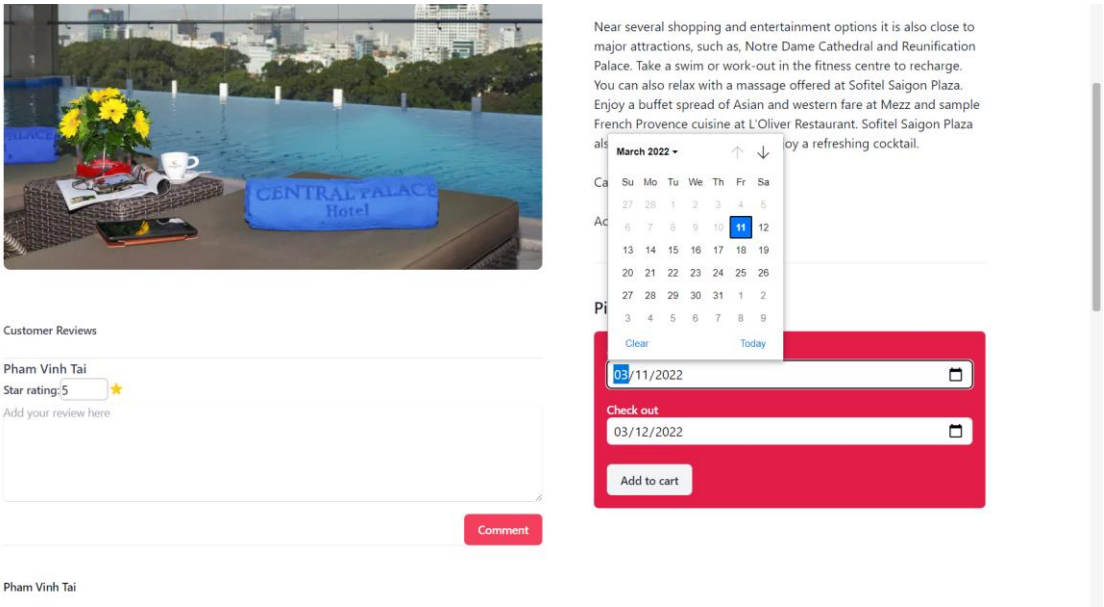
a. Description

This page users can see the description of the room and reviews

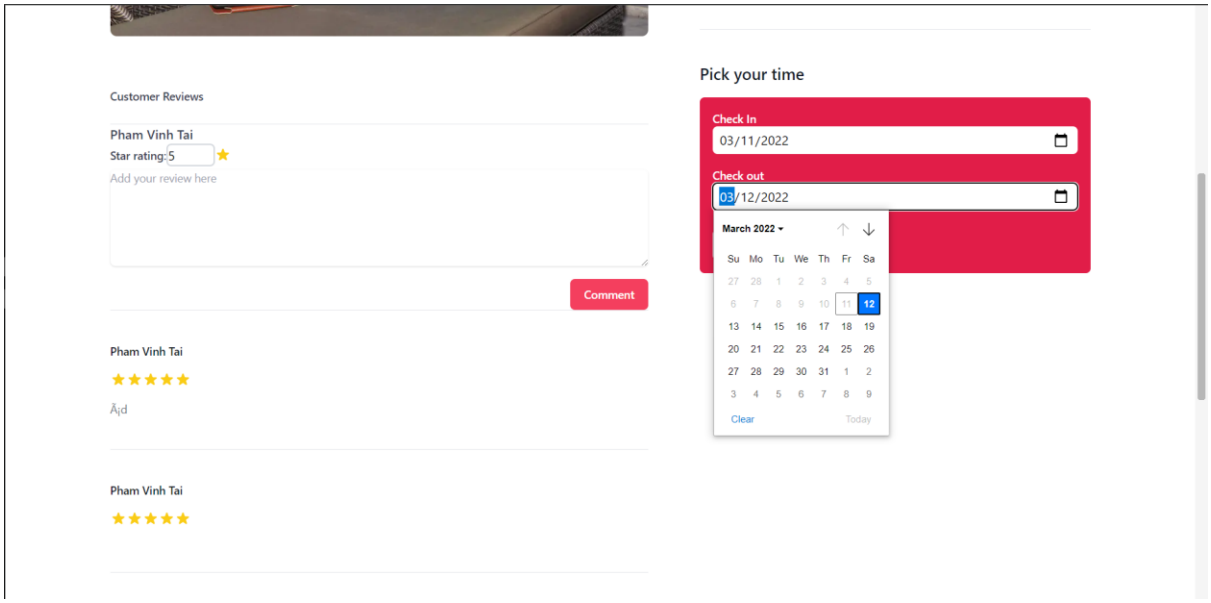
b. Validation

Review form					
Fiel	**Type**	**Min**	**Max**	**isRequired**	**Description**
rating	number	1	5		
review	String	0	1000		

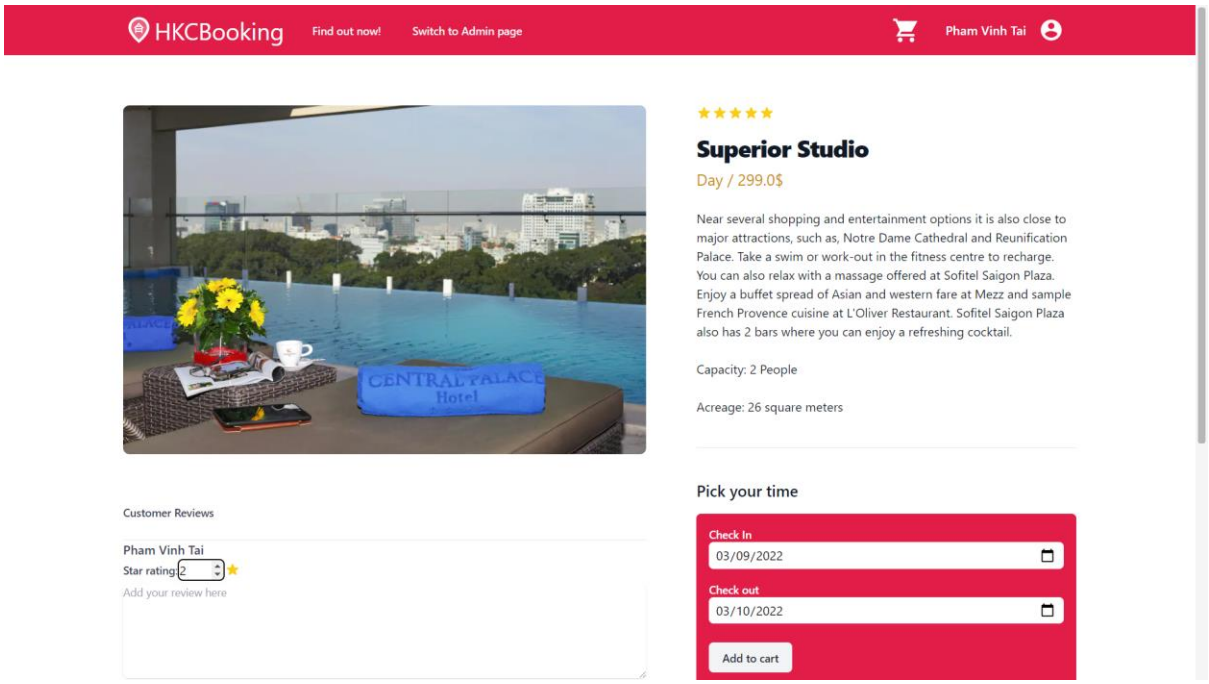
Check in date must be today or later



Check out later than check in



c.Accomplishment



2.7 Booking Order Management

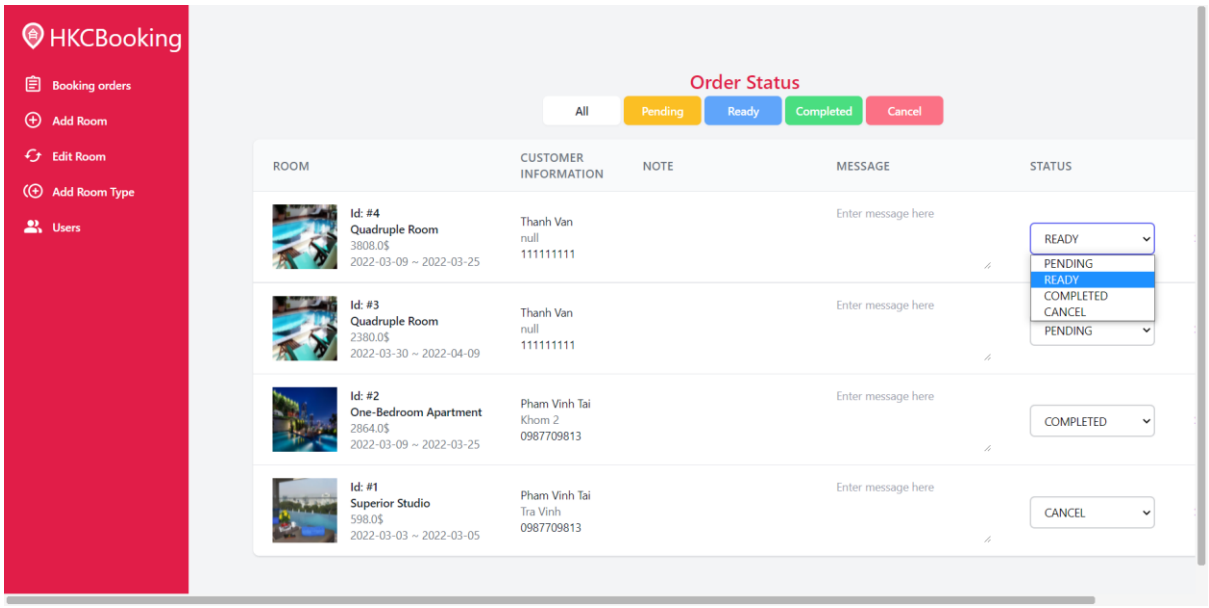
a. Description

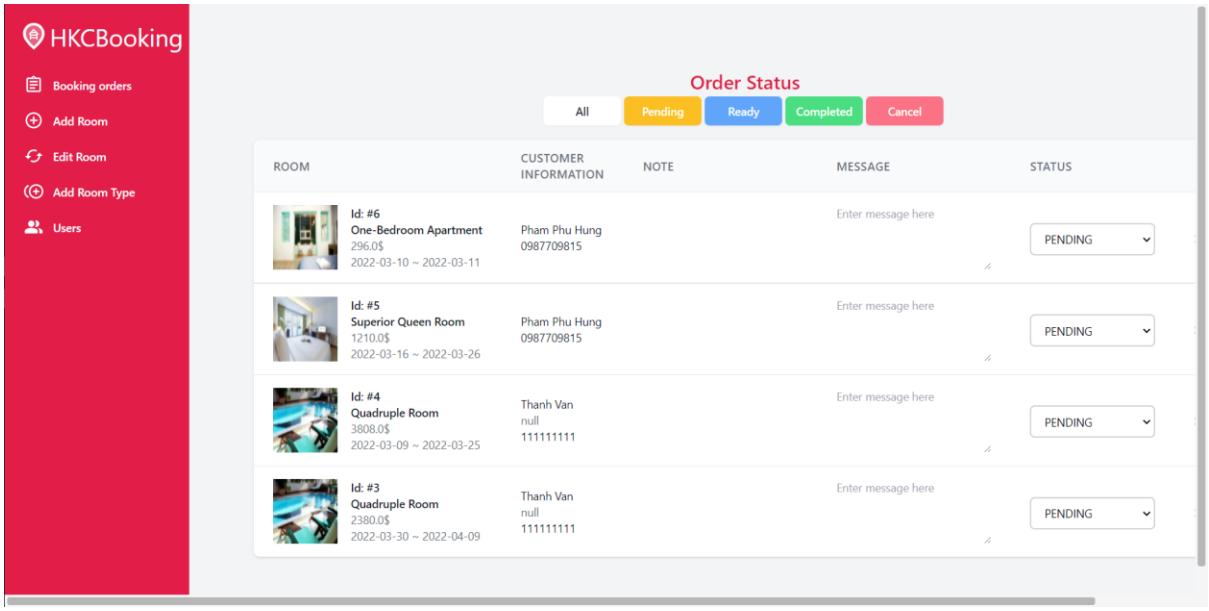
This page helps the administrator to know all the information about the rooms that have been booked and edit the status

b. Validation

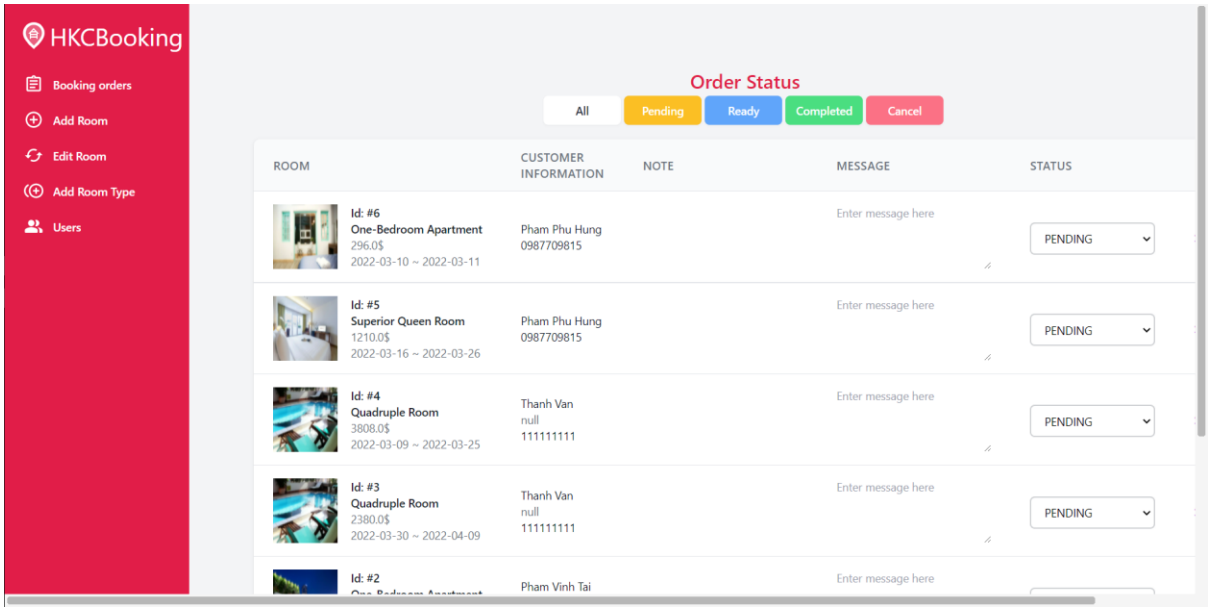
Booking order form					
Field	**Type**	**min**	**max**	**isRequired**	**Description**
message	String				
status	String			Not null	

Manager can change status, choose type status to search





c. Accomplishment



2.8 Add room for management

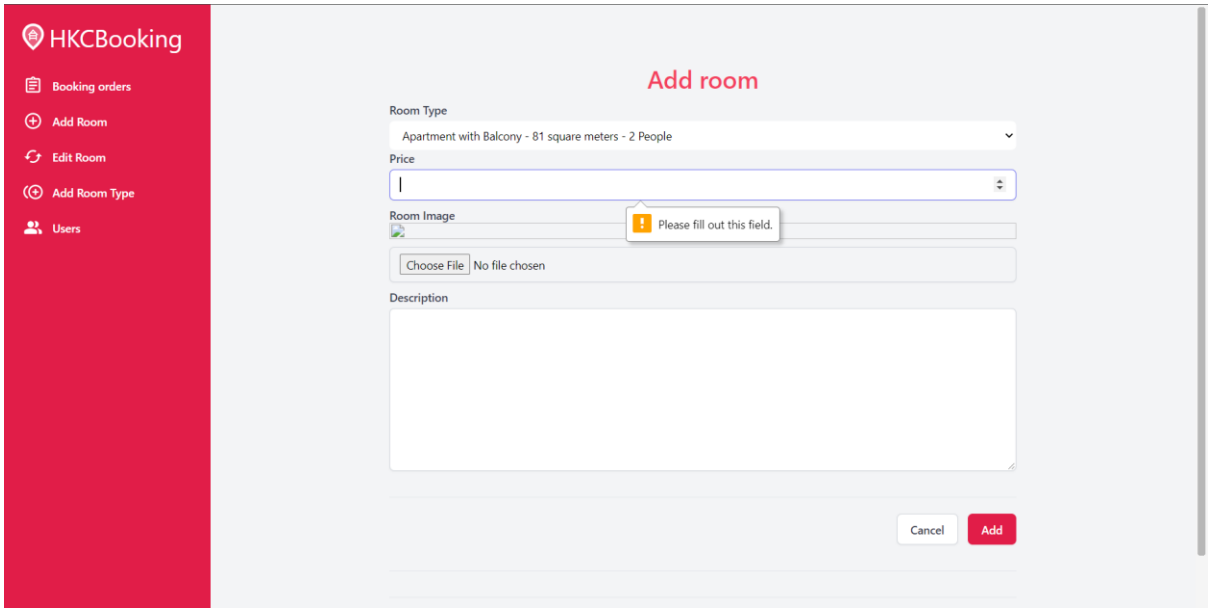
a. Description

When the hotel adds a new room with the same style as the existing ones

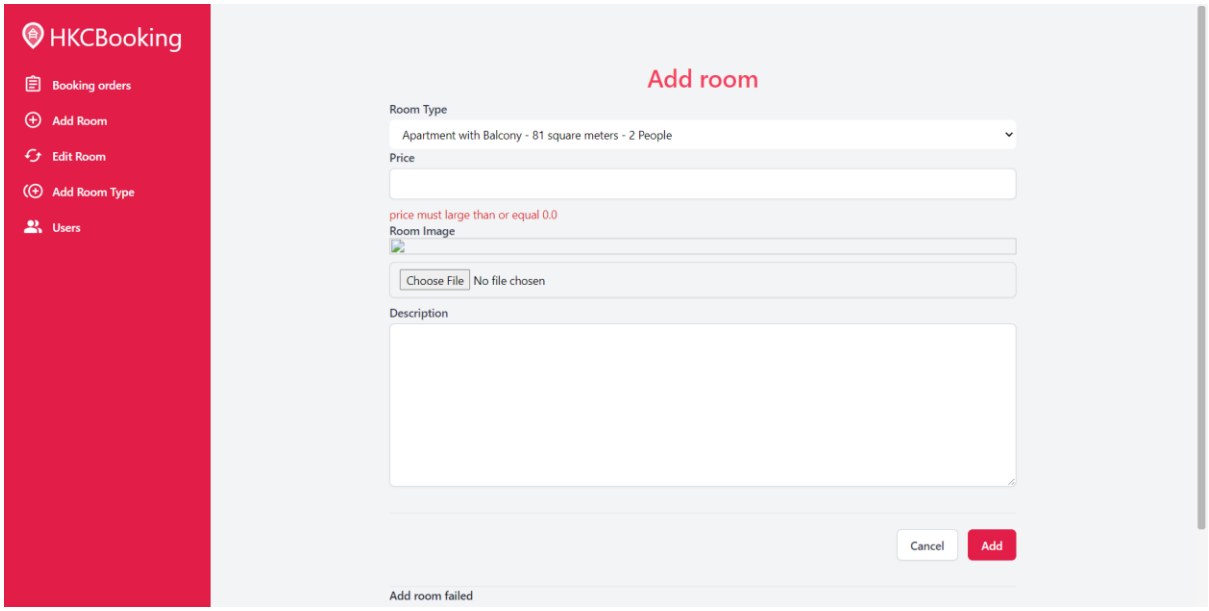
b. Validation

Add room form					
Field	**Type**	**min**	**max**	**isRequire**	**Description**
RoomType	String	0	200	Not null	
Price	Float			Not null	Greater than 0
Room image	String	0	255	Not null	
Description	String	0	500	Not null	

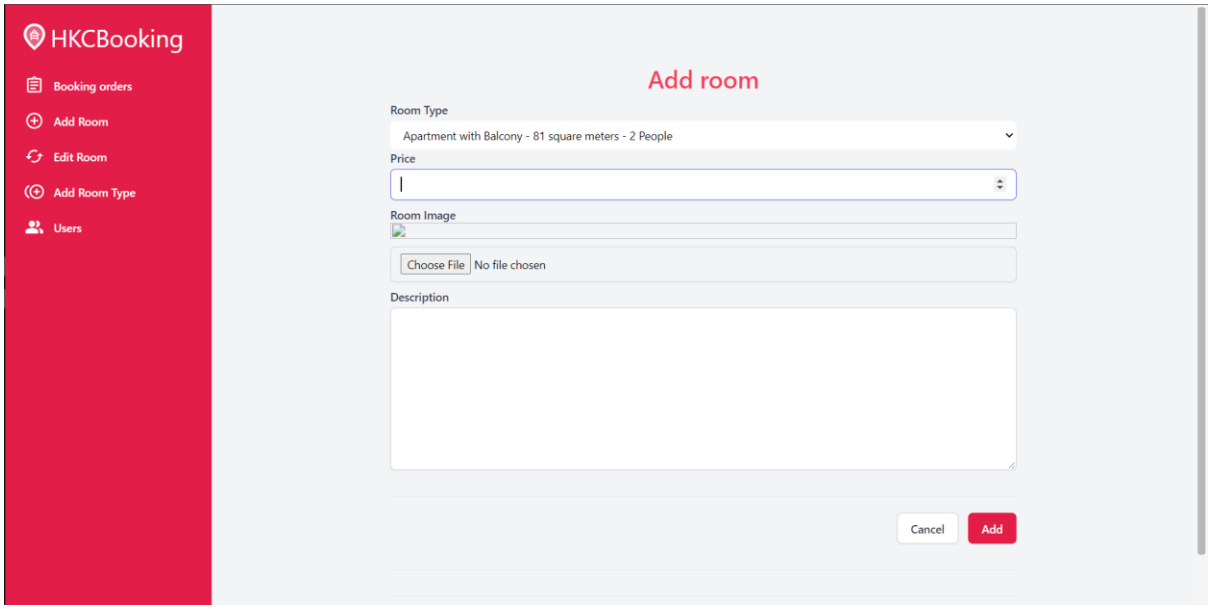
When price is empty



Price must large than 0



c. Accomplishment



2.9 Add room type

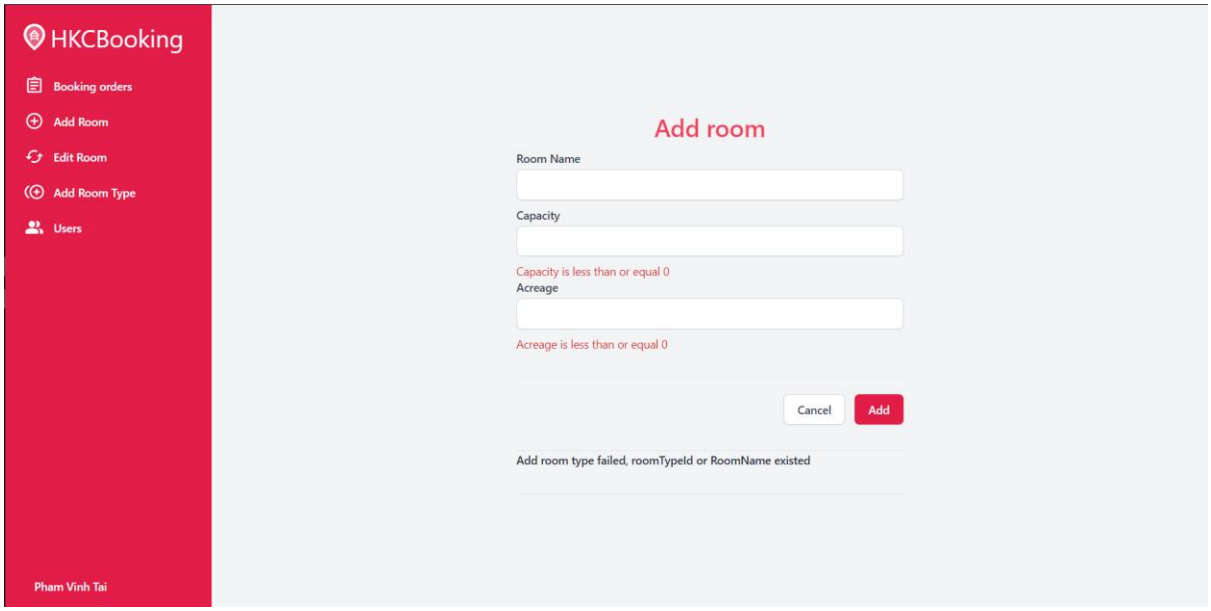
a. Description

When the hotel has a new room type added to the system

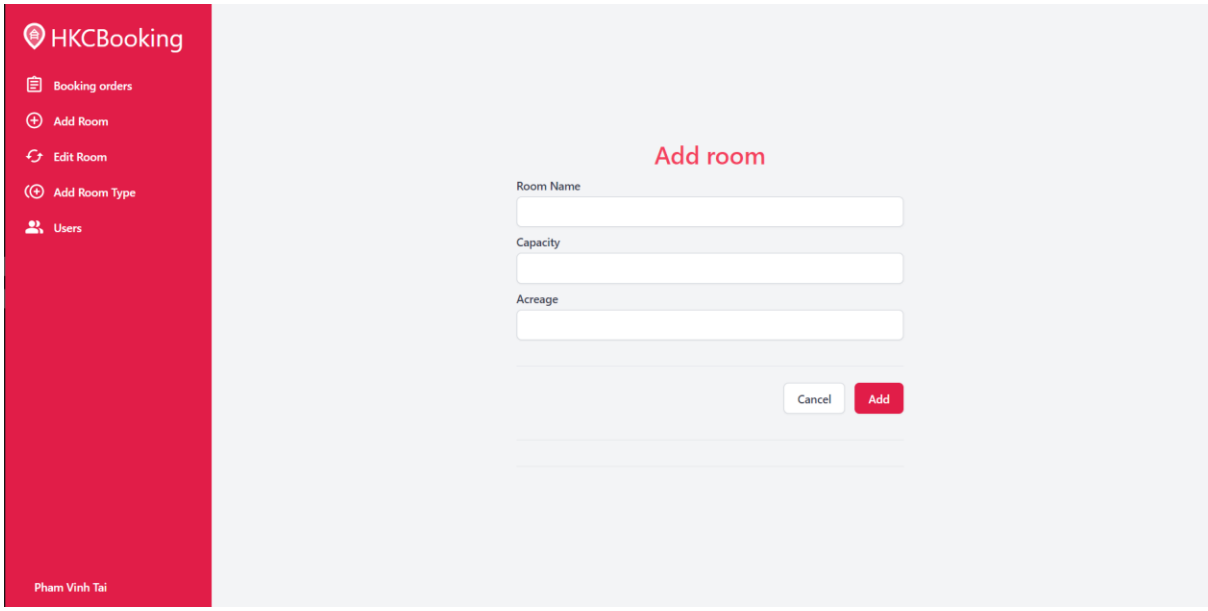
b. Validation

Add room Type form					
Field	**Type**	**min**	**max**	**isRequired**	**Description**
roomName	String	0	200	Not null	
capacity	int			Not null	
acreage	int			Not null	

Capacity and Acreage must large than 0



c.Accomplishment



2.10 Edit room

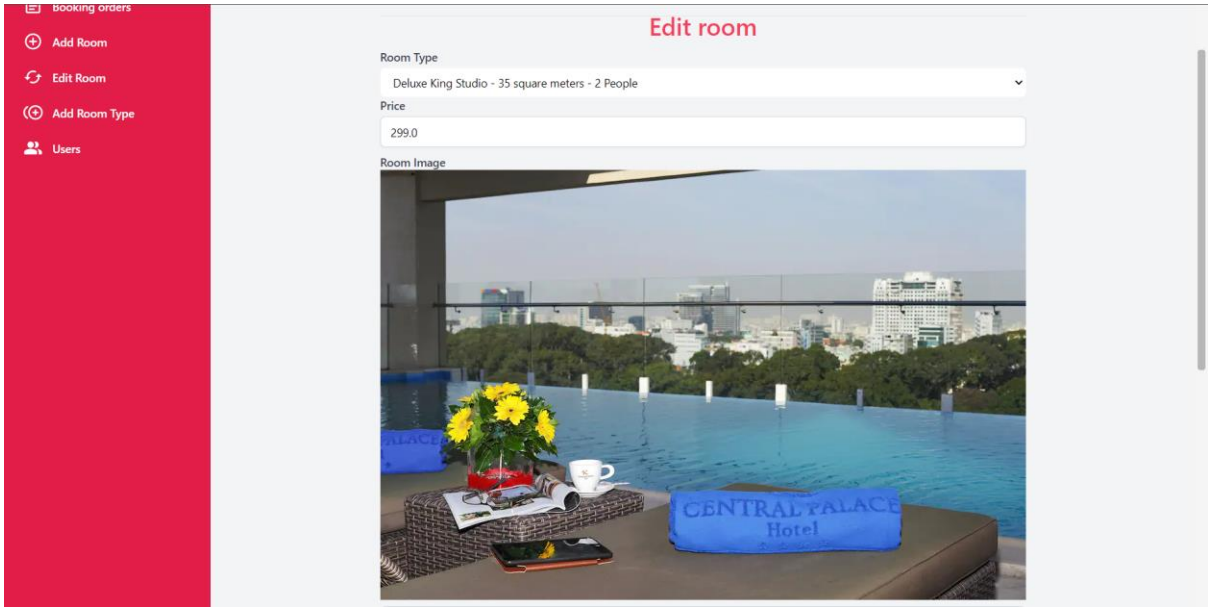
a.Description

This page helps the manager to edit all the information of the room

b.Validation

Edit Room Form					
Field	**Type**	**min**	**max**	**isRequired**	**Description**
Room type	String	0	200	Not null	
Price	float			Not null	
Room image	String	0	255	Not null	
description	String	0	500	Not null	
status	String				

c.Accomplishment



2.11 Room list manager

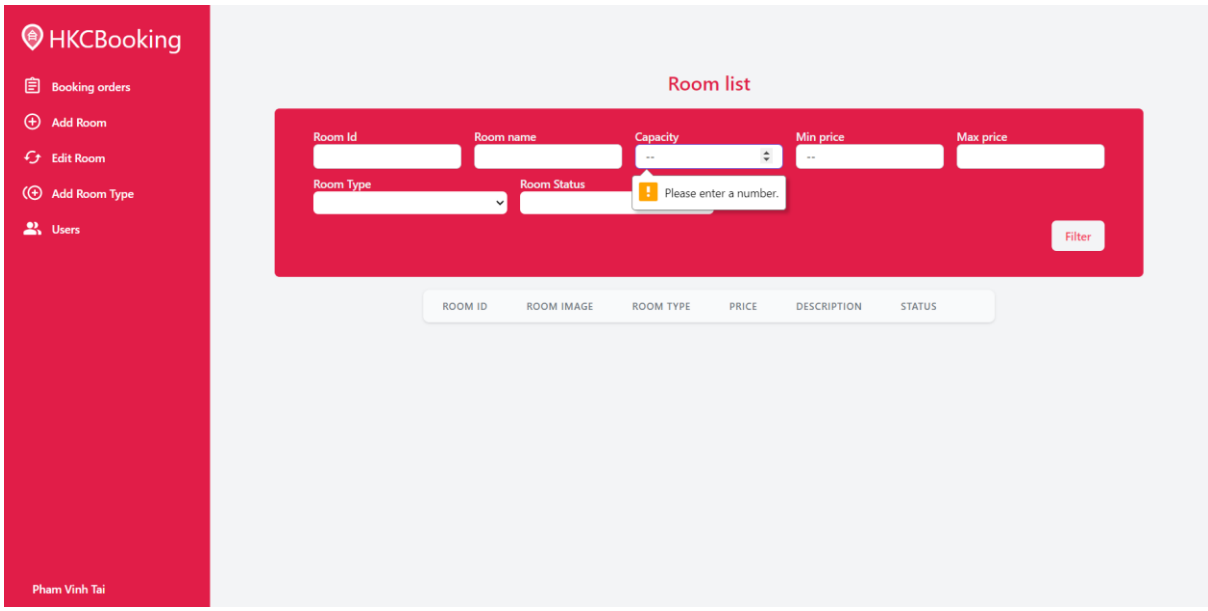
a.Description

This page helps management to find all rooms to edit or delete

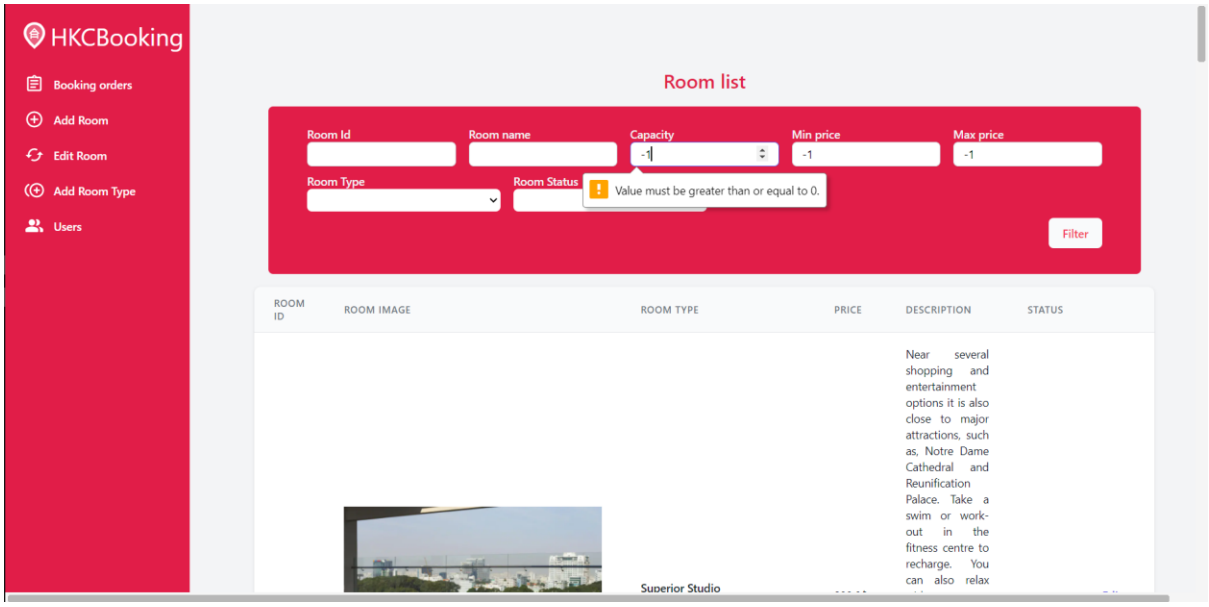
b.Validation

Room list form					
Field	**Type**	**min**	**max**	**isRequired**	**description**
Room id	int				
Room name	String	0	200		
Capacity	int			Must be a number	
Min price	float	0	max	Must be a number	Default value 0
Max price	float	0	max	Must be a number	Default value 9999
Room type	String	0	200		
Room status	String				

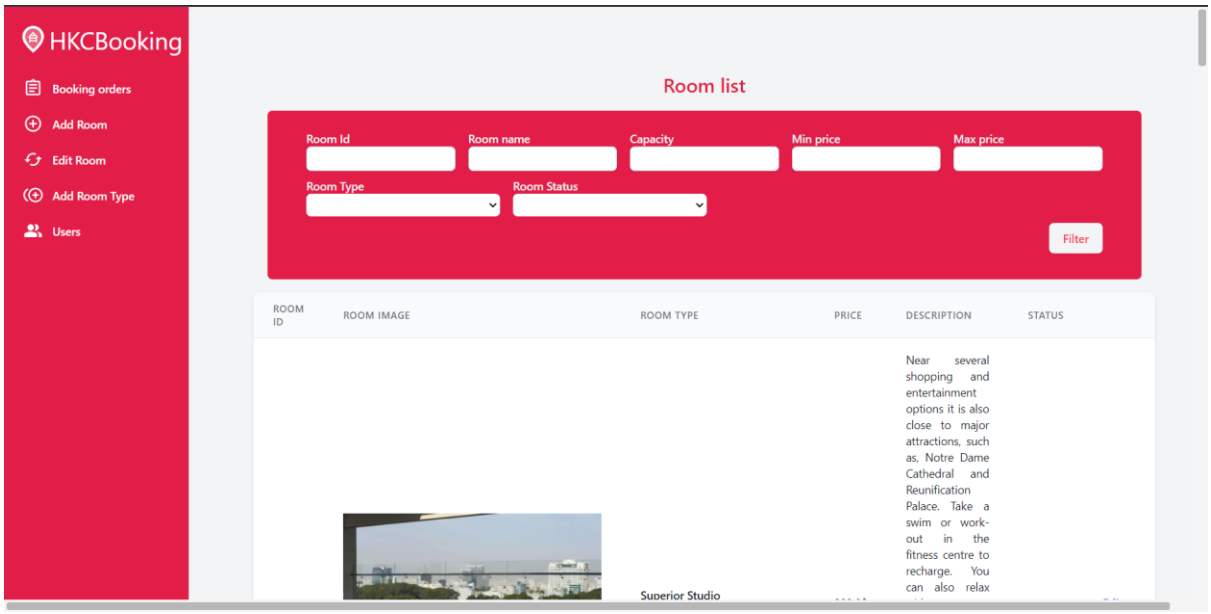
If min-max price and capacity not a number, page will displayed “please enter a number”



When number is not positive



c. Accomplishment

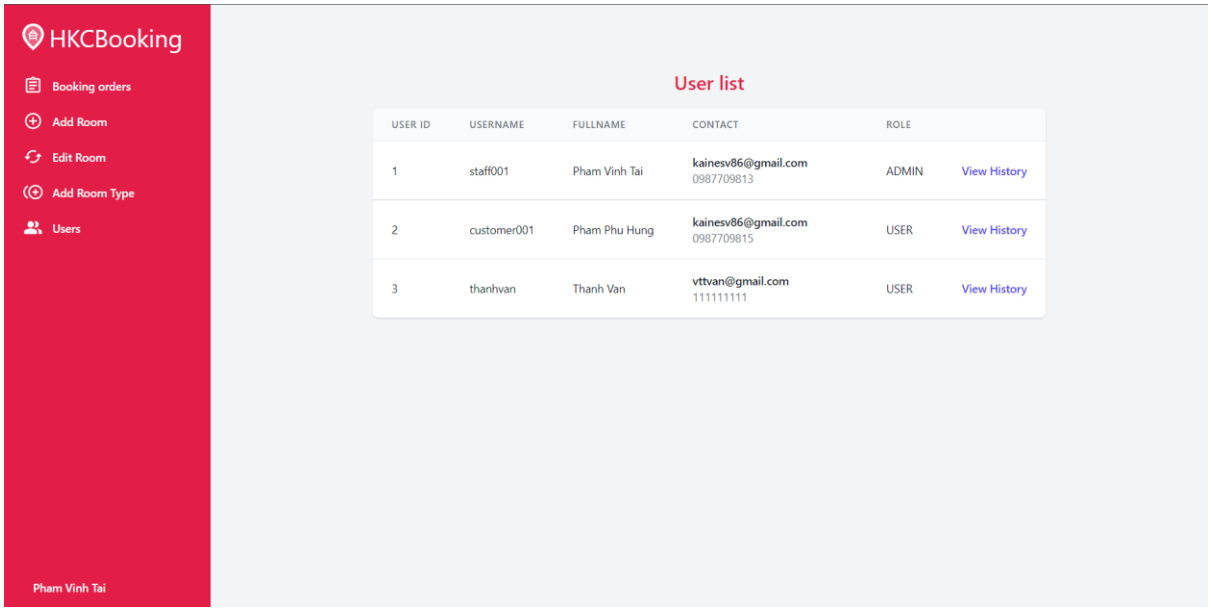


2.12 User list

a. Description

This page will show all users including admin to see everyone's role

b. Accomplishment



2.13 User history booking

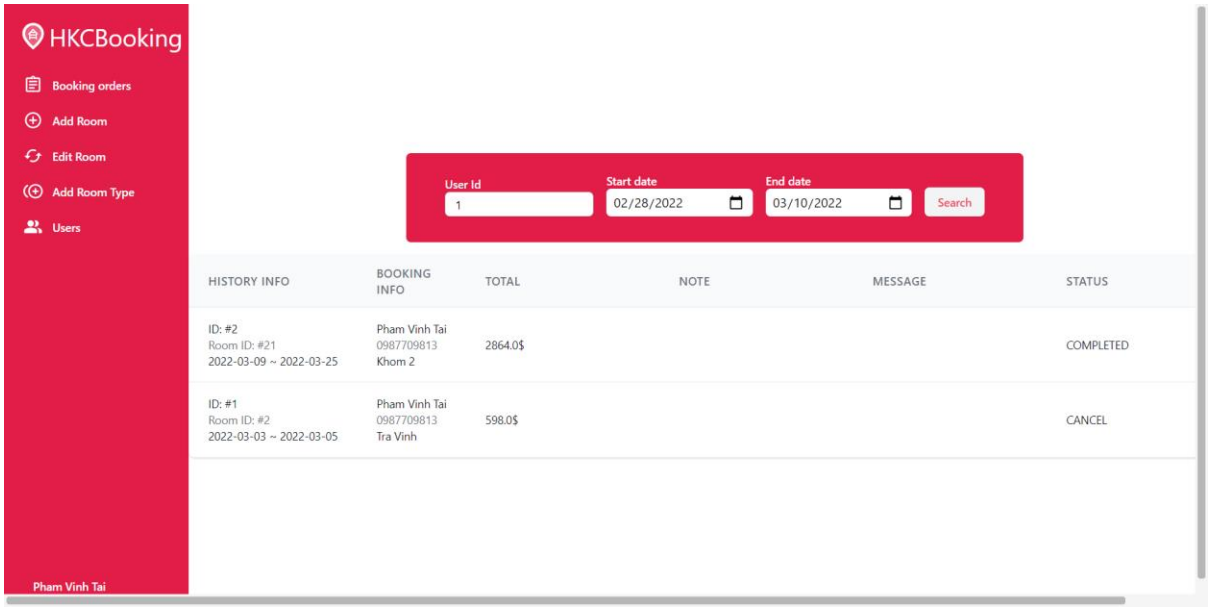
a. Description

This page is used to search users and check their booking history from start date to end date

b. Validation

User list form					
Field	**Type**	**Min**	**Max**	**isRequired**	**Description**
User id	int			Not null	
Start date	Date			Not null	
End date	Date			Not null	

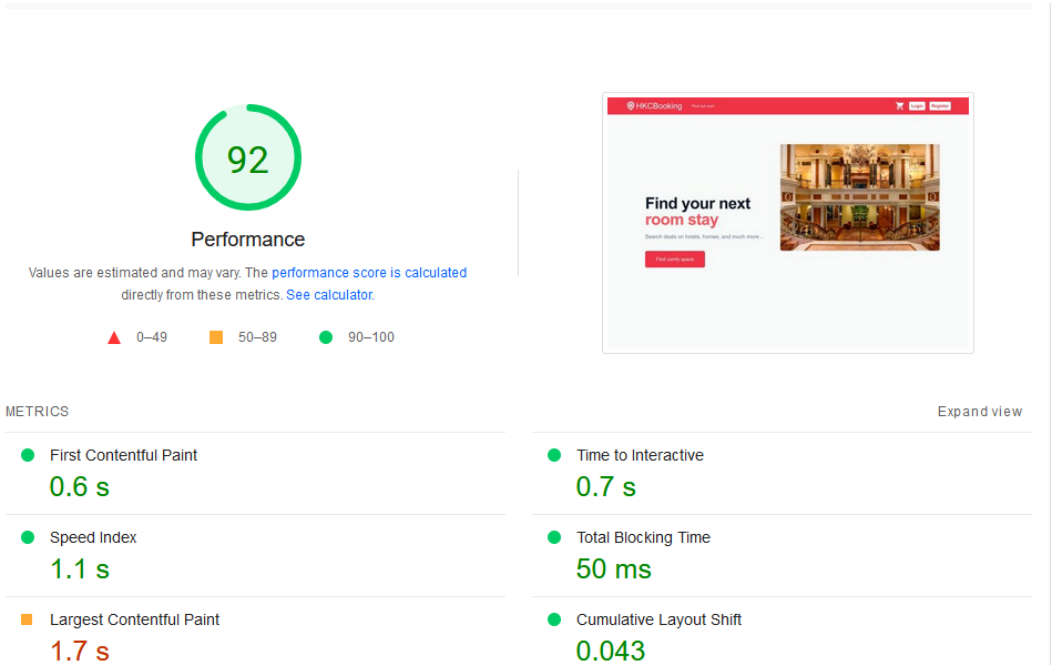
c.Accomplishment



IV. Conclusion

About website

Pros: In this application, we have to use Tailwind CDN to make it easier to skip up CSS to focus JSP and servlet controllers. Apply materials and in class and self-learn on the internet about SOLID, MVC model to design systems effectively and able to scalable and stable in future.



Cons: Focus more on Servlet and JSP than have to design by using CSS, we got some trouble about performance by Tailwind CDN so it loads a large amount of unnecessary CSS classes. Also, across platforms this website does not support too much about mobile devices.

Improvement: to be better off, We need to move on to other frameworks like Node JS and React JS where a large package ready to install, split to front-end and back-end is easy to manage and communicate by restful API.

About team

Pros: Apply scrum workflow by using Github, Google Meet... we try to use Github forks to make independent code and the scrum master order member to write code based on their tasks so every member has a chance to write code and learn experience from other code.

Cons: First time using Github forks instead branch do we cost unnecessary time to learn how to using, another one, member got limit experience about SOLID principles and clean code, so the code are too personal (code smell) and we tried to reduce it as possible and make them able to scalable and stable.

Improvement: We should try to more focus about a required of project, learning about necessary about basic object-oriented programming, SOLID principles, report an effective way by app like Notion instead Message to increase performance.

V. Reference

Tomcat: Apache Tomcat®

Tailwind: Tailwind CSS

Servlet: Servlet (Java EE 6)