
Testing Document

SWEN90007

Hotel Booking Management System

Team: LANS

In charge:

Mohammad Saood **Abbasi** <mohammadsaoo@student.unimelb.edu.au>

Arman **Arethna** <aarethna@student.unimelb.edu.au>

Navdeep **Beniwal** <nbeniwal@student.unimelb.edu.au>

Levi **McKenzie-Kirkbright** <levim@student.unimelb.edu.au>



SCHOOL OF
**COMPUTING &
INFORMATION
SYSTEMS**

Revision

Date	Version	Description	Author
19/10/22	01.00-D01	Initial draft	Saood, Arman
20/10/22	02.00-D01	Finalization	Saood
20/10/22	03.00-D01	Proof reading	Team

Contents

1. Introduction.....	7
1.1 Proposal.....	7
1.2 Target Users.....	7
2. Covered Requirements.....	7
2.1 Functional or Product Requirements.....	7
3. Functional Test Cases.....	7
3.1 TC1: Authorization.....	8
3.2 TC2: Create Rooms for a listed hotel	8
3.3 TC3: Create Rooms for a delisted hotel.....	9
3.4 TC4: View rooms of a hotel	10
3.5 TC5: View rooms of a cancelled hotel.....	10
3.6 TC6: Create a hotel group.....	11
3.7 TC7: Upgrade user to hotelier.....	12
3.8 TC8: Add hotelier to hotel group.....	12
3.9 TC9: Create a hotel.....	13
3.10 TC10: Customer Registration	14
4. Test Cases	15
4.1 TC11: Remove hotelier from a hotel group where the association does not exist.....	15
4.2 TC12: Create two hotels with the same name concurrently	16
4.3 TC13: Create two hotels with the same emails concurrently	16
4.4 TC14: Create a room for a hotel through a delisted hotelier.....	17
4.5 TC15: Create two rooms in the same hotel concurrently with the same room number.....	18
4.6 TC16: Modify booking through an unassigned hotelier	19
4.7 TC17: Two hoteliers concurrently try to cancel the same booking	19
4.8 TC18: A hotelier and a customer concurrently try to cancel the same booking.....	20
4.9 TC19: A customer tries to modify a booking that the hotelier cancels.....	21
4.10 TC20: A customer tries to create a booking in a hotel that a hotelier delist.....	22
4.11 TC21: Two customers try to create a booking in the same hotel for overlapping set of rooms for overlapping date range.....	23
4.12 TC22: Two customers try to concurrently cancel the same booking	23
4.13 TC23: Two customers try to update respective bookings with overlapping dates.....	24
4.14 TC24: Customer tries to change dates of a booking whose hotel is delisted.....	25
4.15 TC25: Customer tries changing the number of guests for a cancelled booking.....	26
4.16 TC26: Customer tries changing the number of guests for a delisted hotel.....	26

4.17 *TC27: Customer tries changing the number of guests through two instances concurrently ...27*

1. Introduction

This document is intended to outline the different testing strategies adopted by the team to test for sequential and concurrent functionalities.

1.1 Proposal

The purpose of this document is to define and present the test cases for project Hotel Booking Management System, Team LANS, covering the test cases for the system use cases.

1.2 Target Users

This document is mainly designed for those responsible for executing the test cases in this project [team members and SWEN90007 teaching team].

2. Covered Requirements

This section lists the system requirements covered in the test cases.

2.1 Functional or Product Requirements

Requirement Identifier	Requirement Name
TC1	Authorization
TC2	Create Rooms for a listed hotel
TC3	Create Rooms for a delisted hotel
TC4	View rooms of a hotel
TC5	View rooms of a cancelled hotel
TC6	Create a hotel group
TC7	Upgrade user to hotelier
TC8	Add hotelier to hotel group
TC9	Create a hotel
TC10	Customer Registration

3. Functional Test Cases

This section describes the test cases that cover the product requirements of the system.

3.1 TC1: Authorization

Test Type: General	Execution Type: Manual
Objective: Role Based Access Control: <i>A registered user should be able to sign in successfully on the application and perform functions relevant to his role.</i>	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> Use special chars, capital letters, digits, maximum allowed number of chars (username: 20, name: 55, email: 55), zero value (0, 00, 000) to fill email and password fields. 	
Pre-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> Perform tasks specific to your respective role 	
Notes: Landing Page corresponding to having visibility of the following fields in the top bar <ul style="list-style-type: none"> Customer: Search Admin: Admin Dashboard Hotelier: Hotelier Dashboard 	
Time constraint: Minimum: 20 min Maximum: 25 min	

3.2 TC2: Create Rooms for a listed hotel

Test Type: General	Execution Type: Manual
Objective: Verify the user is able to create rooms for a listed hotel	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> Login using hotelier credentials Click on hotelier's dashboard selection in the top bar. Click on hotel details for the hotel where we want to add rooms. Click on the New Room button in the side bar. 	

<ul style="list-style-type: none"> Input valid details for all fields in the room creation form Click on the submit button.
Pre-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> The user is presented with a success message and redirected to the hotel view.
Time constraint: Minimum: 20 min Maximum: 25 min

3.3 TC3: Create Rooms for a delisted hotel

Test Type: General	Execution Type: Manual
Objective: Verify the user is unable to create rooms for a delisted hotel	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> Login using hotelier credentials Click on hotelier's dashboard selection in the top bar. Click on hotel details for the hotel where we want to add rooms. Click on the New Room button in the side bar. Input valid details for all fields in the room creation form Login through a separate instance with admin credentials. Click on the admin dashboard option in the top bar Select the hotels tab in the dashboard Press the '-' option present at the end of the hotel field you want to delist. Proceed to click on the submit button in the hotelier's interface. Action should be unsuccessful 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> The user is presented with an unsuccessful message stating that the hotel does not exist 	

Time constraint:

Minimum: 20 min

Maximum: 25 min

3.4 TC4: View rooms of a hotel

Test Type: General	Execution Type: Manual
Objective: To view all the listed rooms available on the interface.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Open the application • The user should be presented with a list of all hotels on the system 	
Pre-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • The user should be able to successfully navigate through the list of all hotels present in the system 	
Time constraint: Minimum: 20 min Maximum: 25 min	

3.5 TC5: View rooms of a cancelled hotel

Test Type: General	Execution Type: Manual
Objective: To not view delisted hotels on the interface	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Open the application on two different instances. • With one instance observe a particular hotel in the list of hotels displayed on the landing page. 	

<ul style="list-style-type: none"> • With the other instance, login using admin credentials. • Proceed to the admin dashboard present in the top bar • Click on the hotels tab. • Delist a hotel by pressing the '-' button present on the right hand side of the row corresponding to the hotel being observed in the first instance. • Go back to the first instance and refresh the screen to reload the view of hotels. • The list of hotels should not include the hotel initially being observed by the user,
Pre-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • The user should be able to successfully navigate through the list of all hotels present in the system without the delisted hotel present
Time constraint: Minimum: 20 min Maximum: 25 min

3.6 TC6: Create a hotel group

Test Type: General	Execution Type: Manual
Objective: To verify the hotel group creation functionality of the system	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using admin credentials on two separate instances • Click on the admin dashboard in the top navigation bar in both instances • Click on hotel groups section in the dashboard from both instances • Both admins are logged in simultaneously from two different instances 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • Hotel group gets created by the first admin. • The second admin gets an error indicating hotel group name already exists 	

Time constraint:

Minimum: 20 min

Maximum: 25 min

3.7 TC7: Upgrade user to hotelier

Test Type: General	Execution Type: Manual
Objective: To verify that the admin is successfully able to elevate customer status to hotelier	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using admin credentials on two separate instances • Click on the admin dashboard in the top navigation bar in both instances • Click on users section in the dashboard from both instances. • Simultaneously select the make hotelier button for the same user through both the instances. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • Customer becomes a hotelier through the first request sent. A success message shows up on the screen and redirects the admin to the hotels screen. • An unsuccessful message shows up on the screen of the second admin and redirects the admin to the hotels screen. 	
Time constraint: Minimum: 20 min Maximum: 25 min	

3.8 TC8: Add hotelier to hotel group

Test Type: General	Execution Type: Manual
Objective: To verify that the hotelier gets successfully added to a hotel group	
Setup:	

<List the pre-conditions to carry through this test case >

- Login using admin credentials on two separate instances
- Click on the admin dashboard in the top navigation bar in both instances
- Click on hoteliers section in the dashboard from both instances
- Select the same hotelier to assign to a hotel group on both instances simultaneously by pressing the button available at the end of the row

Post-Conditions:

<List the conditions after the test execution >

- First request to assign hotelier is successful. The admin receives a success message and is then redirected to the hotels list
- The second request to assign hotelier is unsuccessful. The admin receives a failure message and is then redirected to the hotels list

Time constraint:

Minimum: 20 min

Maximum: 25 min

3.9 TC9: Create a hotel

Test Type:	Execution Type:
General	Manual
Objective:	
To verify the functionality to disable hoteliers not assigned to hotel groups from creating hotels	
Setup:	
<p><List the pre-conditions to carry through this test case ></p> <ul style="list-style-type: none"> • Login using admin credentials • Click on the admin dashboard in the top navigation bar • Click on hoteliers section in the dashboard from • Login to a second instance using the hoteliers credentials • Navigate to the hoteliers dashboard section using the navigation bar on the top • Click on the 'Create Hotel' button. • Fill in the form with appropriate details. • Switch to admin instance and remove the hotelier to hotel group association. • Try creating the hotel from the hotelier's side. 	
Post-Conditions:	
<p><List the conditions after the test execution ></p>	

- Admin is able to successfully remove the hotelier hotel group association
- The hotelier is shown an error message and gets redirected to an error state.

Time constraint:

Minimum: 20 min

Maximum: 25 min

Time constraint:

Minimum: 20 min

Maximum: 25 min

3.10 TC10: Customer Registration

Test Type: General	Execution Type: Manual
Objective: <i>To verify the functionality to register one user out of the two users trying to register onto the system using the same email address.</i>	
Setup: <ul style="list-style-type: none"> • Click on the Login button on top right corner of the home page of the application • Enter a valid email id and a valid password (that matches the criteria specified in the instructions on the screen) • Repeat the above set of instructions from another instance using the same email. • Click on continue from both the instances 	
Post-Conditions: <ul style="list-style-type: none"> • The first user successfully registers onto the system and is redirected back to the home page to login into the system. • The second user is unable to register and gets an error message. The user stays on the signup page. 	

Time constraint:

Minimum: 20 min

Maximum: 25 min

4. Test Cases

This section describes the general test cases that will be related to more than one requirement or assumption / constraint, avoiding data replication.

4.1 TC11: Remove hotelier from a hotel group where the association does not exist

Test Type: General	Execution Type: Manual
Objective: To remove the hotelier status from an account making them a normal customer of the application.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using admin credentials • Proceed to the admin dashboard in the top bar • Select the hoteliers tab in the dashboard. • Login using a different admin credentials from another instance. • Proceed to the admin dashboard in the top bar • Select the hoteliers tab in the dashboard. • Proceed to simultaneously through both instances remove the hotelier from his hotel group by pressing the button present at the right end of the row. • Only one should successfully be executed. 	
Pre-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • The first user is able to successfully remove a hotelier from his corresponding hotel group. • The second user gets redirected to a screen indicating there is an incorrect action that has taken place. They are the redirected back to the view of all hotels 	

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.2 TC12: Create two hotels with the same name concurrently

Test Type: General	Execution Type: Manual
Objective: To verify the functionality of the system not allowing creation of the same hotel through two accesses.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using hotelier credentials on two different instances. • Click on the hotelier dashboard in the top navigation bar. • Click on the Create hotel button in both the instances simultaneously. • Enter the same name in both the instances trying to create two hotels with the same name. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • The first hotelier is able to successfully create the hotel. They receive a success message and are redirected to the hotels page • The second hotelier is unable to create the hotel and receives an error message. They are further redirected to the hotels page. 	
Time constraint: Minimum: 20 min Maximum: 25 min	

4.3 TC13: Create two hotels with the same emails concurrently

Test Type: General	Execution Type: Manual
Objective: To verify the functionality of the system not allowing creation of hotels with the same email through two accesses.	

Setup:

<List the pre-conditions to carry through this test case >

- Login using hotelier credentials on two different instances.
- Click on the hotelier dashboard in the top navigation bar.
- Click on the Create hotel button in both the instances simultaneously.
- Enter the same email address in both the instances trying to create two hotels with the same name.

Post-Conditions:

<List the conditions after the test execution >

- The first hotelier is able to successfully create the hotel. They receive a success message and are redirected to the hotels page
- The second hotelier is unable to create the hotel and receives an error message. They are further redirected to the hotels page.

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.4 TC14: Create a room for a hotel through a delisted hotelier

Test Type: General	Execution Type: Manual
Objective: To verify the functionality of the system not allowing creation of a hotel through a hotelier that is not associated with the hotel group any more	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the admin credentials on one instance. • Navigate to the admin dashboard through the top navigation bar. • Click on the hoteliers tab. • Login using the hotelier credentials on another instance. • Navigate to the hotelier dashboard through the top navigation bar. • Select a hotel to add a room to. • Click on the new room tab. • Fill in the details for addition of a new room. • Through the admin instance, disassociate the hotelier from their hotel group through the button at the end of the row. 	

<ul style="list-style-type: none"> Through the hotelier instance, try creating the room.
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> The admin is able to successfully disassociate the hotelier from his hotelier group The hotelier is not able to create a room and receives an error message. They are further redirected to the hotels page.
Time constraint: Minimum: 20 min Maximum: 25 min

4.5 TC15: Create two rooms in the same hotel concurrently with the same room number

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit the user to create a room with a room number already existing in the database.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> Login using hotelier credentials on two different instances. Click on the hotelier dashboard in the top navigation bar. Click on the same hotel through both the instances simultaneously. Select the new room tab in both the instances. Create a new room through both such that the room number is the same. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> The first hotelier is able to successfully create the room. They receive a success message and are redirected to the hotels page The second hotelier is unable to create the room and receives an error message. They are further redirected to the hotels page. 	

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.6 TC16: Modify booking through an unassigned hotelier

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit hotelier to modify a booking if they are no longer associated with the hotel group.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the admin credentials on one instance. • Navigate to the admin dashboard through the top navigation bar. • Click on the hoteliers tab. • Login using the hotelier credentials on another instance. • Navigate to the hotelier dashboard through the top navigation bar. • Select a hotel to view the bookings on • Through the admin instance, disassociate the hotelier from their hotel group through the button at the end of the row. • Try clicking the 'Cancel Booking' button corresponding to any particular booking. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • The admin is able to successfully disassociate a hotelier from the hotel group • The hotelier is unable to proceed with booking modification. They receive an error message and are redirected to the hotels page 	
Time constraint: Minimum: 20 min Maximum: 25 min	

4.7 TC17: Two hoteliers concurrently try to cancel the same booking

Test Type: General	Execution Type: Manual
------------------------------	----------------------------------

Objective:

To verify the functionality to not permit multiple hoteliers to cancel the same booking at the same time.

Setup:

<List the pre-conditions to carry through this test case >

- Login using hotelier credentials on two different instances.
- Click on the hotelier dashboard in the top navigation bar.
- Click on the same hotel through both the instances simultaneously.
- Select the bookings tab through both instances
- Click on the 'Cancel Booking' button concurrently through both instances for the same booking.
-

Post-Conditions:

<List the conditions after the test execution >

- The first hotelier is able to successfully cancel the booking
- The second hotelier is unable to cancel the booking and gets an error message. They are then redirected to the hotels page.

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.8 TC18: A hotelier and a customer concurrently try to cancel the same booking

Test Type:	Execution Type:
General	Manual
Objective:	
To verify the functionality to not permit users cancel the same booking at the same time.	
Setup:	
<i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using hotelier credentials. • Click on the hotelier dashboard in the top navigation bar. • Click on the same hotel through both the instances simultaneously. • Select the bookings tab through both instances • Login using the customer credentials through a different instance. • Select the 'My Bookings' tab from the top navigation bar. 	

<ul style="list-style-type: none"> On a particular booking, click the 'Edit' button. Through the hotelier's instance navigate to a view of the same booking. Concurrently, press the 'Cancel Booking' button on both instances.
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> The first user to cancel the booking receives a success message and is then redirected to the hotels page. The second user is unable to cancel the booking and gets an error message. They are then redirected to the hotels page.
Time constraint: Minimum: 20 min Maximum: 25 min

4.9 TC19: A customer tries to modify a booking that the hotelier cancels

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit users cancel the same booking at the same time.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> Login using hotelier credentials. Click on the hotelier dashboard in the top navigation bar. Click on the same hotel through both the instances simultaneously. Select the bookings tab through both instances Login using the customer credentials through a different instance. Select the 'My Bookings' tab from the top navigation bar. On a particular booking, click the 'Edit' button. Through the hotelier's instance navigate to a view of the same booking. Concurrently, press the 'Cancel Booking' button on the hotelier instance and the 'Update Booking' button on the customer's instance with a change in dates. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> The first user to modify the booking receives a success message and is then redirected to the hotels page. The second user is unable to modify the booking and gets an error message. 	

They are then redirected to the hotels page.

4.10 TC20: A customer tries to create a booking in a hotel that a hotelier delist

Test Type: General	Execution Type: Manual
Objective: <i>To verify the functionality to create a booking in a hotel that the hotelier of the hotel is trying to delist</i>	
Setup: <ul style="list-style-type: none"> As a customer, click on the login button on the top right corner Enter your registered email and password in the login form. Click on continue button. Click on one of the hotels shown on the home page Select a room for booking, enter the guests details. As an admin, click on the login button on top right corner of the home page of the application Enter a valid email id and a valid password (that matches the criteria specified in the instructions on the screen) Click on continue button. Go to admin dashboard, and click on hotels tab. As an admin try to click on the delist button in front of the same hotel in which the user is trying to create a booking, as the customer try to click on the Book Rooms button. 	
Post-Conditions: <ul style="list-style-type: none"> The admin successfully delists the hotel with the hotel status being changed to Unlisted. The user gets an error message that something have went wrong during the booking process. 	
Time constraint: Minimum: 20 min Maximum: 25 min	

4.11 TC21: Two customers try to create a booking in the same hotel for overlapping set of rooms for overlapping date range

Test Type: General	Execution Type: Manual
Objective: <i>To verify the functionality to create a booking in a hotel where another customer is trying to create a booking for overlapping set of rooms for overlapping data range</i>	
Setup: <ul style="list-style-type: none"> As customer one, click on the login button on the top right corner Enter your registered email and password in the login form. Click on continue button. Click on one of the hotels shown on the home page Select a room for booking in a given date range, enter the guests details. As customer two, accessing the app from a different machine, click on the login button on the top right corner Enter your registered email and password in the login form. Click on continue button. Click on the same hotel that customer one has selected Search for rooms in a date range that overlaps with customer one's given date range. Select the set of rooms that overlaps with customer one rooms selection. Both try to click on the Book Rooms button after filling up guest details at the same time. 	
Post-Conditions: <ul style="list-style-type: none"> Customer one successfully creates a booking. Customer two gets an error message that something has went wrong during the booking process. 	
Time constraint: Minimum: 20 min Maximum: 25 min	

4.12 TC22: Two customers try to concurrently cancel the same booking

Test Type:	Execution Type:
-------------------	------------------------

General	Manual
Objective: To verify the functionality to not permit users to cancel the same boooking at the same time.	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the customer credentials through two instances. • Select the 'My Bookings' tab from the top navigation bar for both instances. • On the same booking, click the 'Edit' button for both instances. • Concurrently, press the 'Cancel Booking' button on both instances. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • The first user to cancel the booking receives a success message and is then redirected to the hotels page. • The second user is unable to cancel the booking and gets an error message. They are then redirected to the hotels page. 	
Time constraint: Minimum: 20 min Maximum: 25 min	

4.13 TC23: Two customers try to update respective bookings with overlapping dates

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit users to have overlapping dates	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the customer credentials through two instances. • Select the 'My Bookings' tab from the top navigation bar for both instances. • From both instances, select a booking with common rooms booked for different dates. • Concurrently, try cancelling the booking on the first instance while changing the dates to overlap on the second instance 	
Post-Conditions: <i><List the conditions after the test execution ></i>	

- If the cancel instance executes first, the second instance will successfully update booking dates.
- If the bookings modification executes first, the first instance will not be able to update the booking.

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.14 TC24: Customer tries to change dates of a booking whose hotel is delisted

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit users to make updates to bookings corresponding to delisted hotels	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the admin credentials on one instance. • Navigate to the admin dashboard in the navigation bar. • Select the hotels tab. • Select the hotel that you wish to delist. • Login through the customer credentials on a second instance. • Navigate to the bookings tab in the navigation bar. • Select a booking corresponding to the hotel about to be cancelled. • Modify dates of the booking while concurrently delisting the hotel 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • If the delist instance executes first, the hotel will be delisted and the booking modification made by the customer will not be possible. • If the update booking runs first, the booking will be successfully updated and the hotel will also get delisted successfully. 	

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.15 TC25: Customer tries changing the number of guests for a cancelled booking

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit users to make updates to bookings corresponding to cancelled bookings	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the customer credentials through two instances. • Select the 'My Bookings' tab from the top navigation bar for both instances. • Select the same booking from both instances. • Concurrently, try cancelling the booking on the first instance while changing the number of guests for the other. 	
Post-Conditions: <i><List the conditions after the test execution ></i> <ul style="list-style-type: none"> • If the cancellation instance executes first, the booking will be cancelled and the booking modification made by the customer will not be possible. • If the update booking runs first, the booking will be successfully updated and the booking will also get cancelled successfully. 	
Time constraint: Minimum: 20 min Maximum: 25 min	

4.16 TC26: Customer tries changing the number of guests for a delisted hotel

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit users to make updates to bookings corresponding to delisted hotels	

Setup:

<List the pre-conditions to carry through this test case >

- Login using the admin credentials on one instance.
- Navigate to the admin dashboard in the navigation bar.
- Select the hotels tab.
- Select the hotel that you wish to delist.
- Login through the customer credentials on a second instance.
- Navigate to the bookings tab in the navigation bar.
- Select a booking corresponding to the hotel about to be cancelled.
- Modify number of guests for the booking while concurrently delisting the hotel

Post-Conditions:

<List the conditions after the test execution >

- If the cancellation instance executes first, the booking will be cancelled and the booking modification made by the customer will not be possible.
- If the update booking runs first, the booking will be successfully updated and the booking will also get cancelled successfully.

Time constraint:

Minimum: 20 min

Maximum: 25 min

4.17 TC27: Customer tries changing the number of guests through two instances concurrently

Test Type: General	Execution Type: Manual
Objective: To verify the functionality to not permit users to make concurrent updates to	
Setup: <i><List the pre-conditions to carry through this test case ></i> <ul style="list-style-type: none"> • Login using the customer credentials through two instances. • Select the 'My Bookings' tab from the top navigation bar for both instances. • Select the same booking from both instances. • Concurrently, try cancelling the booking on the first instance while changing the number of guests for the other. 	
Post-Conditions:	

<List the conditions after the test execution >

- If the cancellation instance executes first, the booking will be cancelled and the booking modification made by the customer will not be possible.
- If the update booking runs first, the booking will be successfully updated and the booking will also get cancelled successfully.

Time constraint:

Minimum: 20 min

Maximum: 25 min