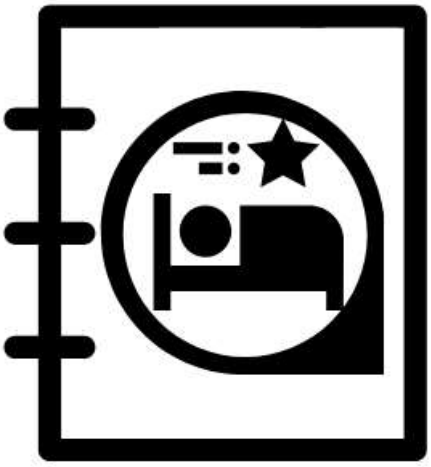
California State University Fullerton
CPSC 462

Object Oriented Software Design
Use Case Model – Annex 2
for the



Hotel Reservation System

Manage Reservations
Fully Dressed Use Case

Josh Ibad
Chief Software Architect
joshcibad@csu.fullerton.edu

Revision History:

Version	Date	Summary of Changes	Author
1.0	2021-10-18	• Initial Release	Josh Ibad
2.0	2021-11-15	Reviewed after obtaining feedback from management. Various revisions made: • Changed role to Chief Software Architect • Changed Use Case Scope from "System Use Case" to plainly "System" • Success guarantee modified to include expected data and sample expected data • Special requirements modified to have testable requirements • Frequence of occurence modified to include a specific measure Will be further reviewed and refined in subsequent iterations.	Josh Ibad

Table of Contents

1 Use Case Description

1.1 Use Case Title

Manage Reservations

1.2 Scope

System

1.3 Category

Greatest Risk

1.3.1 Risks addressed

1. Risk 2 - Security Breach: The public-facing interface for reservation management will be secured and tested, and will also built to write access to accomodations and hotel room listings.

1.4 Level

User Goal

1.5 Primary Actor(s)

- Guest

1.6 Stakeholders and Interests

1.6.1 Hotel CEO

Hotel CEO wants the reservation process to be managable and for it to be usable in the accomodation process in order to maintain high business operability. He wants the reservation process to be free of human error and operational risk.

1.6.2 Hotel's Private Equity Investor

Hotel Private Equity Investor wants guests to be convinced of marketable hotel room description and to provide business to the hotel by reserving a room.

1.6.3 Hotel Manager

Hotel Manager wants guests to see an updated list of hotel rooms along with their desired marketable features. Hotel Manager wants to convince guests to reserve a room and to gain their business.

1.6.4 Hotel Clerk

Hotel Clerk wants reservations made to be visible to them and to prevent conflicting accomodations in the future.

1.6.5 Guest

Guest wants to be able to reserve a room in through an easy-to-use and accessible interface. If possible, they want to be able to get a descriptive list of available hotel rooms and reserve a room from the comfort of their own homes through their own devices.

1.7 Preconditions (Entrance Criteria)

Guest account is configured with Guest role access. System is running and accessible from Guest's client device.

1.8 Success Guarantee (Exit State)

Guest reservation management actions (create/update/delete) are saved. Modifications are immediately reflected and blocks other conflicting reservations from being made. Expected data: Message of "Room reserved succesfully", along with total charges paid and account number of payment card. Example: "Room reserved successfully. Card 123 charhed $150.00"

1.9 Main Success Scenario

1. The Guest requests to connect to the system as a Guest, providing their username and password. The System responds with a session.
2. The Guest requests for a list of available hotel rooms at a given datetime range. The System responds with a list of hotel rooms available at the given datetime range.
3. The Guest requests to get information about a specific hotel room. The System responds with information about the specified hotel room, such as price, room type, bed type, bed count, and description.
4. The Guest requests to reserve the specified room at the given datetime range. The System responds with the total cost of reservation.
5. The Guest requests to make a payment, providing billing information such as cardholdder name, billing address, email, card type, card number, card expiration date, and CVV code. The System responds with a receipt of transaction and confirmation of reservation.
6. The Guest requests to terminate their session. The System responds with a receipt of whether or not the session was terminated.

1.10 Extensions (Alternate paths)

2-5a. Instead of creating a reservation, Guest wants to cancel an existing reservation

1. The Guest requests a list of upcoming reservations under their account. The System responds with a list of upcoming reservations.

2. The Guest requests for information about a specific reservation. The System responds with information about the specified reservation such as datetime range of reservation, price, room, and cancelation costs.

3. The Guest requests to cancel the same specified reservation. The System responds with a receipt of transaction and a confirmation of cancellation.

2-5a. Instead of creating a reservation, Guest wants to update an existing reservation

1. The Guest requests a list of upcoming reservations under their account. The System responds with a list of upcoming reservations.

2. The Guest requests for information about a specific reservation. The System responds with information about the specified reservation such as datetime range of reservation, price, room, and cancelation costs.

3. The Guest requests to update the reservation, providing a new datetime range. The System responds with a receipt denoting if the update was successfull.

1.11 Special Requirements

Reservation must be distinguished from accomodation. The interface should show different reservation management and accomodation management as different, distinct options or functionalities.

Interface must be intuitive and usable to non-technical Guests of various background. Non-technical personnel should be able to perform the use case with little to no instruction in under 5 minutes.

Guest Client interface must be secure to resist attacks from malicious actors since this interface will be public-facing. Client interface must not be vulnerable to injections and access controls can't be bypassed. Must be able to withstand a black-box penetration test from professional hackers for a day of assessment.

1.12 Technology and Data Variations List

1-4. The Hotel Reservation System must provide an intuitive graphical user interface and must support typical peripheral device inputs such as that of mouse, keyboard, and potentially touch screen. It must also be reachable remotely through a web interface where Guests will make reservations from.

1. Authentication is typed credentials. But within a two year, security may improve to include two factor authentication or OAuth.

1.13 Frequency of Occurrence

Accurate measure: Once per room every five days.

Estimate: 50 times every week

Rationale: Not every room gets booked and some people walk-in for their stay. However, reservations are also done very frequently.

1.14 Miscellaneous

The transaction feature can be improved upon and isolated into its own use case, which can be extended for use in providing other services. The Guest account feature can also be elaborated upon to improve user experience, and potentially support offering returning users rewards to promote business.

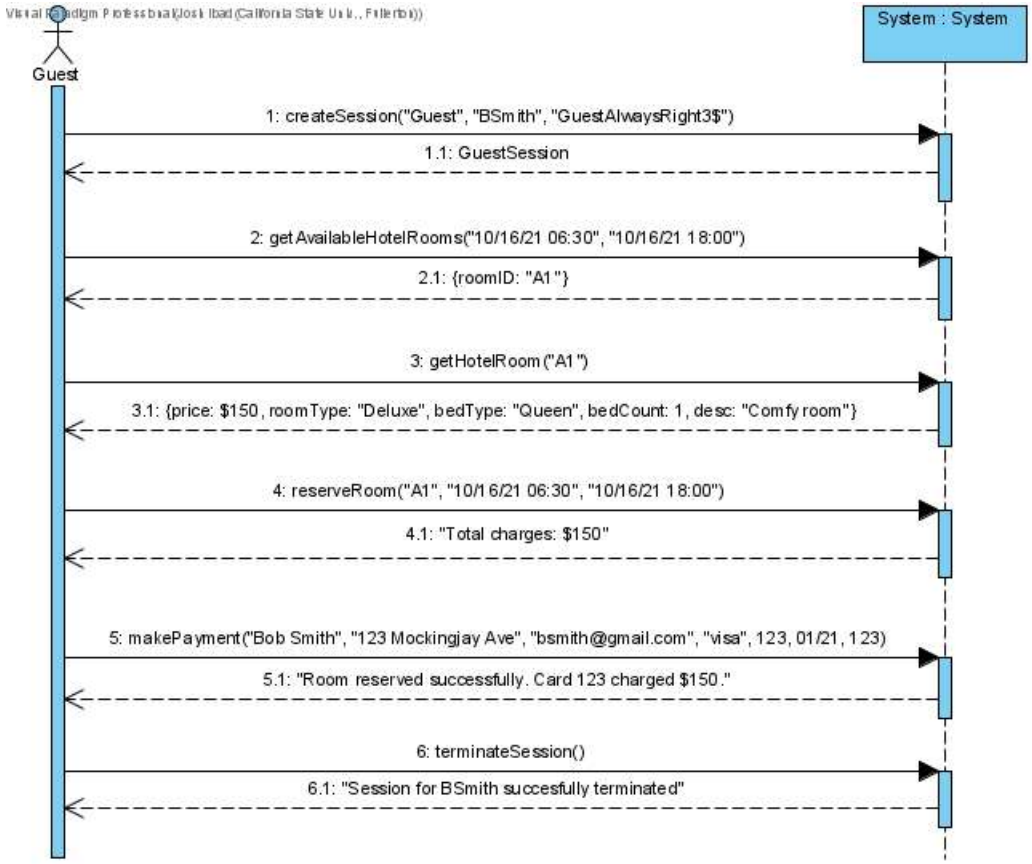
2 System Sequence Diagrams

2.1 Create Reservation

2.1.1 Scenario Description

The Guest requests to connect to the system as a Guest, providing his username "BSmith" and password "GuestAlwaysRight3$". The system responds with a session with guest role access, specifically with access to BSmith's account. Then, the Guest requests a list of available hotel rooms during the datetime range 10/16/21 06:30 to 10/16/21 18:00. The system responds with a list that is available at the time specified, replying with a single room "A1". The Guest requests for more information about room "A1". The system responds with information about the room, that it costs $150 per night, is a "Deluxe" room with 1 "Queen" bed, and is described as a "Comfy room". The Guest then requests to reserve room "A1" from 10/16/21 06:30 to 10/16/21 18:00. The system responds with the total charges, returning a message "Total charges: $150". The Guests then requests to make a payment under the cardholder name "Bob Smith", billing address "123 Mockingjay Ave", email address "bsmith@gmail.com", card type "Visa", card number 123, expiration date 01/21, and CVV code 123. The system responds with a receipt of transaction and confirmation of reservation, returning the message "Room reserved successfully. Card 123 charged $150." Finally, the Guest requests to terminate his connetion. The system responds with a receipt of the success of connection termination, returning the success message "Session for BSmith successfully terminated".

2.1.2 System Sequence Diagram



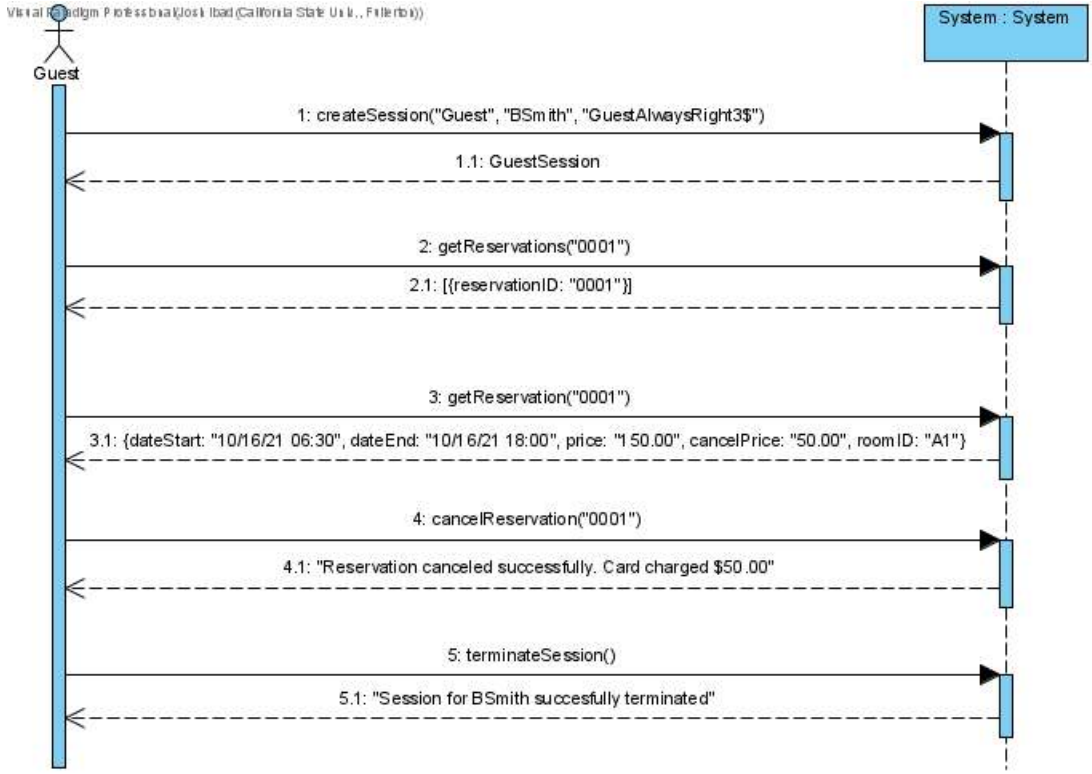
...

2.2 Cancel Reservation

2.2.1 Scenario Description

The Guest requests to connect to the system as a Guest, providing his username "BSmith" and password "GuestAlwaysRight3$". The system responds with a session with guest role access, specifically with access to BSmith's account. Then, the Guest requests a list of upcoming reservations under his account "0001". The system responds with a list of upcoming reservations: a single reservation "0001". The Guest requests more information about reservation 0001. The system replies with infomation about the reservation -- that it is set for 10/16/21 from 06:30 - 18:00, at room "A1", costing $150 per night, and costing $50 to cancel. The Guest then requests to cancel reservation 0001. The system responds with a receipt of cancellation and transaction, returning the message "Reservation canceled successfully. Card charged $50.00". Finally, the Guest requests to terminate his connetion. The system responds with a receipt of the success of connection termination, returning the success message "Session for BSmith successfully terminated".

2.2.2 System Sequence Diagram



...