

## To Do

- Review the Project Overview page and Problem Statement again more thoroughly.
- (If there were any updates,) Updated requirements clarification questions (and answers if given already).
- Create the scrum dashboard for task management.
- Create the code repository (e.g., Github, Bitbucket) and clone it to local. Include the repository link in the documentation.
- Use-case diagram, clearly illustrating system boundary and actors (including system actors).
- At least 6-7 use cases in brief form.

## Links

[Github](#) [Jira](#)

## Use Cases

### Table of Contents

1. [Create Hotel Clerk Account](#)
  2. [Make Reservation](#)
  3. [Process Check-In](#)
  4. [Search Available Room](#)
  5. [Cancel Reservation](#)
  6. [Modify Reservation](#)
- 

### 1. Create Hotel Clerk Account

#### Use Case Name:

Create Hotel Clerk Account

#### Actor

Admin

#### Preconditions

- Hotel system online and operational
- User is logged in as an Admin

#### Postconditions

- A new hotel clerk account is created
- Clerk account has given username and default password (or custom password)

## Main Success Scenario

1. Admin selects option to create hotel clerk account
2. System prompts admin to enter desired username and shows prefilled password for account
3. Admin enters username and optional different password
4. System validates input
5. System creates clerk account
6. System displays success message for created account

## Extensions

4a. Username already in use: 1. System detects username already in use (Ex: John\_Smith) 2. System displays error message and potential username replacement (Ex: John\_Smith1)

5a. Failure to create account: 1. Display error message of account creation failure 2. Reprompt user to try creating account again

## Special Requirements

- Create account in timely manner
  - Keep log of created accounts
  - Keep log of which admin created account
- 

## 2. Make Reservation

### Use Case Name:

Make Reservation

### Actor

Hotel Guest

### Preconditions

- The hotel system is functional and online
- The user is logged in to the system
- Room and reservation data exists in the database
- The user has searched for available rooms

### Postconditions

- A new reservation is created in the system
- The selected room is marked as reserved for the specified dates
- Guest information is recorded (name, address, credit card number, expiration date)
- Confirmation is displayed to the user

# Main Success Scenario

1. The user selects a room from the list of available rooms
2. The user enters the check-in and check-out dates
3. The user selects the rate type (standard, promotion, group, or non-refundable)
4. The user enters or confirms their personal information (name, address)
5. The user enters payment information (credit card number, expiration date)
6. The system validates all input data
7. The system verifies room availability for the selected dates
8. The system calculates the total cost based on quality level and rate type
9. The system creates the reservation and stores it in the database
10. The system displays reservation confirmation with details

## Extensions

3a. Corporate guest selected: 1. The user selects their corporation from the list 2. The system records the corporation for billing purposes 3. Continue from step 4

6a. Invalid input data: 1. The system displays an error message indicating the invalid fields 2. The user corrects the input 3. Continue from step 6

7a. Room is no longer available: 1. The system notifies the user that the room has been booked 2. The system redirects the user to search for available rooms 3. Use case ends

## Special Requirements

- Credit card information must be securely stored
  - Reservation must be atomic (all or nothing)
- 

## 3. Process Check-In

### Use Case Name:

Process Check-In

### Actor

Hotel Clerk

### Preconditions

- The hotel system is functional and online
- The clerk is logged in to the system
- The guest has an existing reservation for the current date
- At least one room matching the reservation criteria is available

## **Postconditions**

- The guest is checked in and assigned to a specific room
- The room status is updated to occupied
- The check-in date and time are recorded
- The guest can access hotel services (including the store)

## **Main Success Scenario**

1. The clerk searches for the guest's reservation by name or confirmation number
2. The system displays the reservation details
3. The clerk verifies the guest's identity
4. The clerk confirms the reservation details with the guest
5. The system displays available rooms matching the reservation
6. The clerk selects a room to assign to the guest
7. The system allocates the room to the guest
8. The system updates the room status to occupied
9. The system records the check-in timestamp
10. The clerk provides the room key/access information to the guest
11. The system displays check-in confirmation

## **Extensions**

1a. Reservation not found: 1. The clerk verifies guest information 2. The clerk offers to create a new reservation (see Make Reservation use case) 3. Use case ends or continues with new reservation

4a. Guest requests different room type: 1. The clerk searches for alternative available rooms 2. The system displays available alternatives with price differences 3. The guest selects a new room type 4. The system updates the reservation with new rate if applicable 5. Continue from step 5

6a. No rooms available matching reservation: 1. The system notifies the clerk of the situation 2. The clerk offers an upgrade or alternative room 3. The guest accepts or declines the alternative 4. If declined, the clerk processes a cancellation with no penalty 5. Use case ends or continues with alternative room

## **Special Requirements**

- Check-in must update room availability in real-time
- Guest must have an active reservation to access store purchasing

---

### **4. Search Available Room**

#### **Use Case Name:**

Search Available Room

#### **Actor**

Hotel Guest

## Preconditions

- The hotel system is functional and online
- Room and reservation data exists in the database

## Postconditions

- Available rooms are displayed to the user
- Data is not modified

## Main Success Scenario

1. The user selects the search option
2. The user enters their search criteria such as check in / out date, number of guests, number of beds, bed size, etc.
3. System validates input
4. System searches for rooms that match user criteria, if available
5. System displays list of available rooms that match user criteria, if available

## Extensions

## Special Requirements

---

### 5. Cancel Reservation

#### Use Case Name:

Cancel Reservation

## Actor

Hotel Guest

## Preconditions

- The hotel guest is logged into the system
- The guest has an existing reservation

## Postconditions

- The reservation is canceled only if cancellation is permitted
- If cancellation is permitted, any applicable cancellation penalty is recorded

- If cancellation is not permitted, the reservation remains unchanged

## Main Success Scenario

1. The guest selects the option to view reservations
2. The system displays the guest's reservations
3. The guest selects a reservation to cancel
4. The system checks the time remaining until the reservation's check-in date
5. The system determines that the cancellation request is more than the required time
6. The system displays the applicable cancellation policy and any penalty (if required)
7. The guest confirms the cancellation
8. The system cancels the reservation
9. The system displays a cancellation confirmation message

## Extensions

4a. Cancellation not allowed (within a specific time frame): 1. The system determines that the cancellation request is within X hours of the check-in time 2. The system displays a message explaining that cancellation is not permitted according to the policy 3. The reservation remains unchanged

## Special Requirements

- The system must enforce the X-hour cancellation policy exactly
  - Time comparisons must use the hotel's local time zone
  - All cancellation attempts must be logged for auditing and billing purposes
- 

## 6. Modify Reservation

### Use Case Name:

Modify Reservation

### Actor

Hotel Guest

### Preconditions

- The hotel guest is logged into the system
- The guest has an existing reservation
- The reservation has not yet started

### Postconditions

- The reservation is updated only if modification is permitted
- If modification is not permitted, the reservation remains unchanged

- Any change in price is recalculated and recorded

## Main Success Scenario

1. The guest selects the option to view their reservations
2. The system displays the guest's reservations
3. The guest selects a reservation to modify
4. The system displays the current reservation details
5. The guest enters the requested changes (e.g., dates or room type)
6. The system checks whether the modification request is more than X hours before the check-in time
7. The system checks room availability for the requested changes
8. The system recalculates the reservation cost, if applicable
9. The system displays the updated reservation details
10. The guest confirms the modification
11. The system updates the reservation
12. The system displays a modification confirmation message

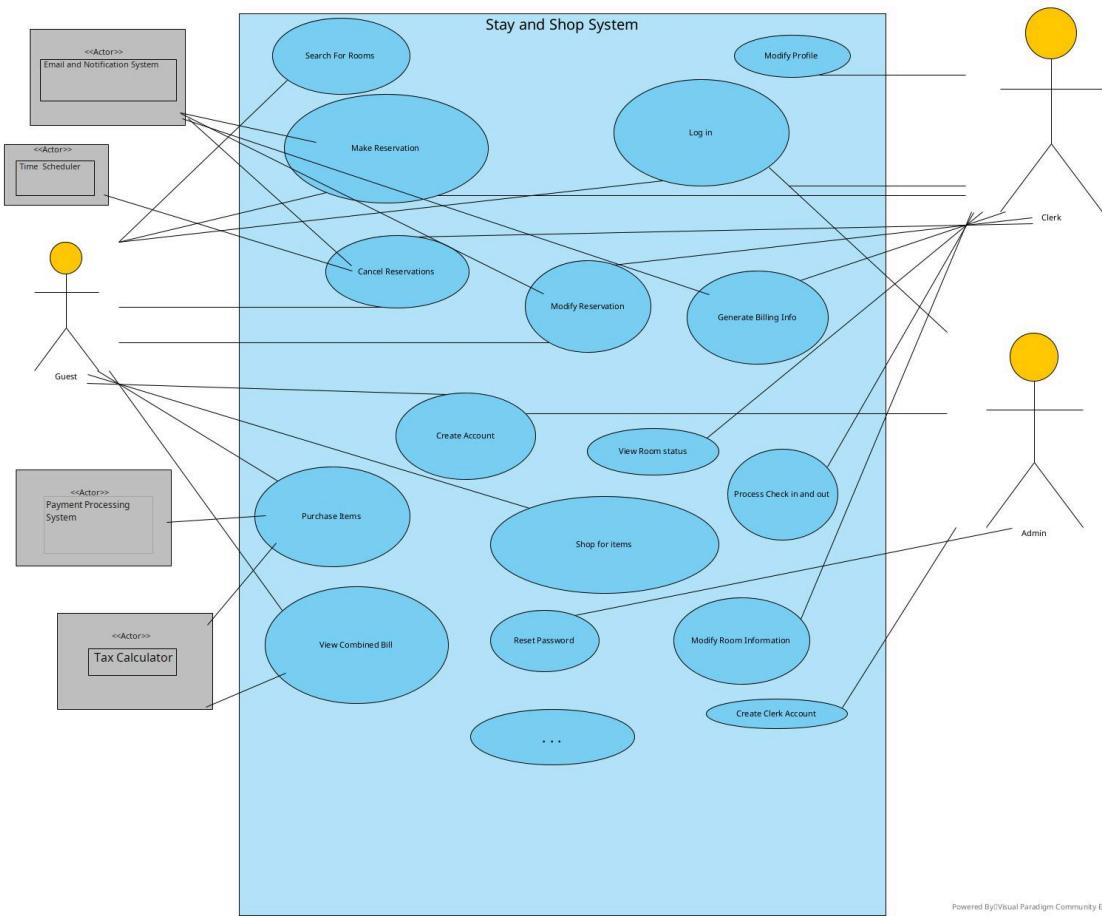
## Extensions

6a. Modification not allowed (within X hours of check-in): 1. The system determines that the modification request is within X hours of the check-in time 2. The system displays a message explaining that modifications are not permitted according to the policy

## Special Requirements

- The system must enforce the X-hour modification policy exactly
  - Availability checks must be consistent with current reservations
  - Price recalculation must follow hotel pricing rules
- 

## Use Case Diagram



Powered By:Visual Paradigm Community Edition