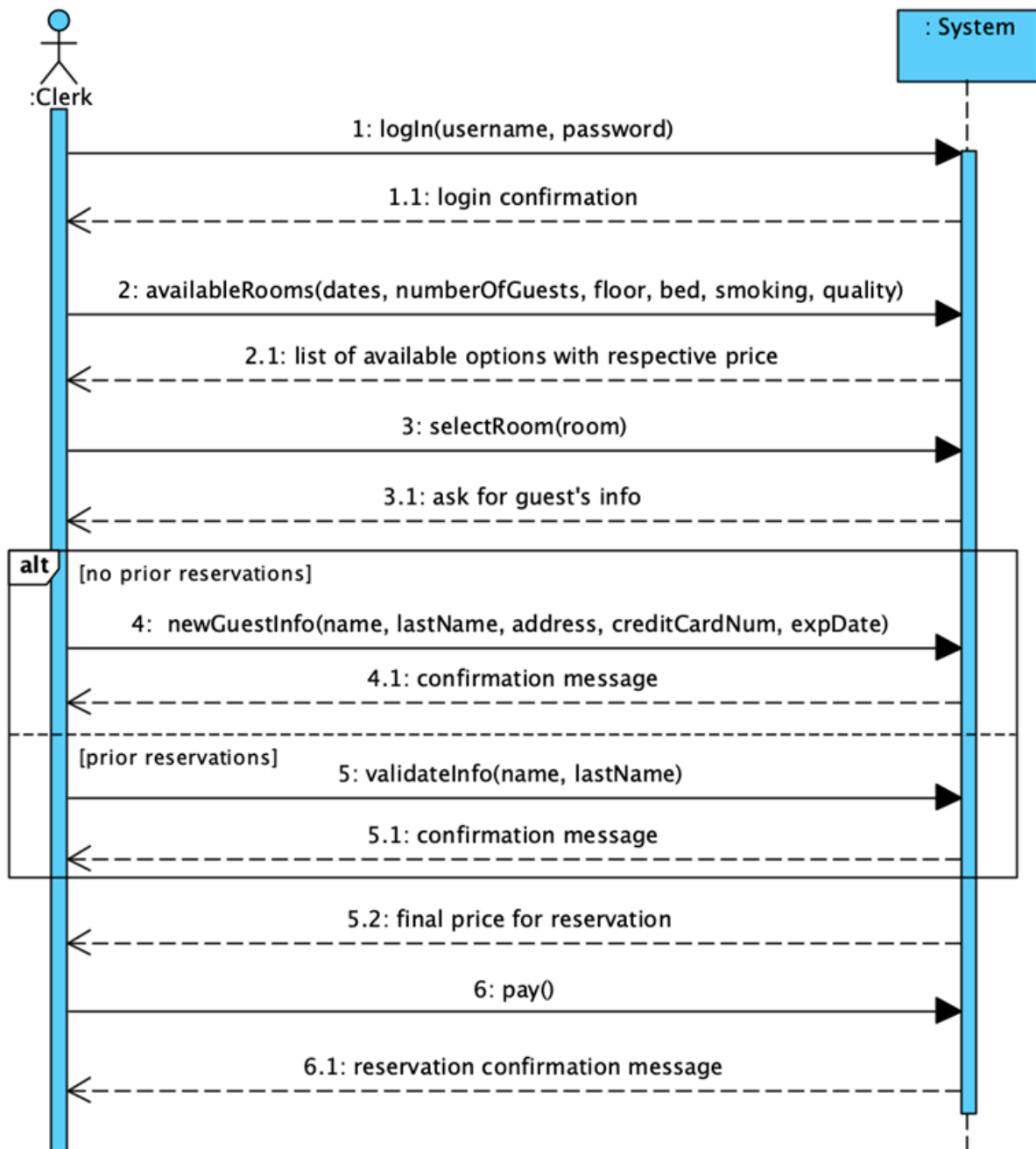


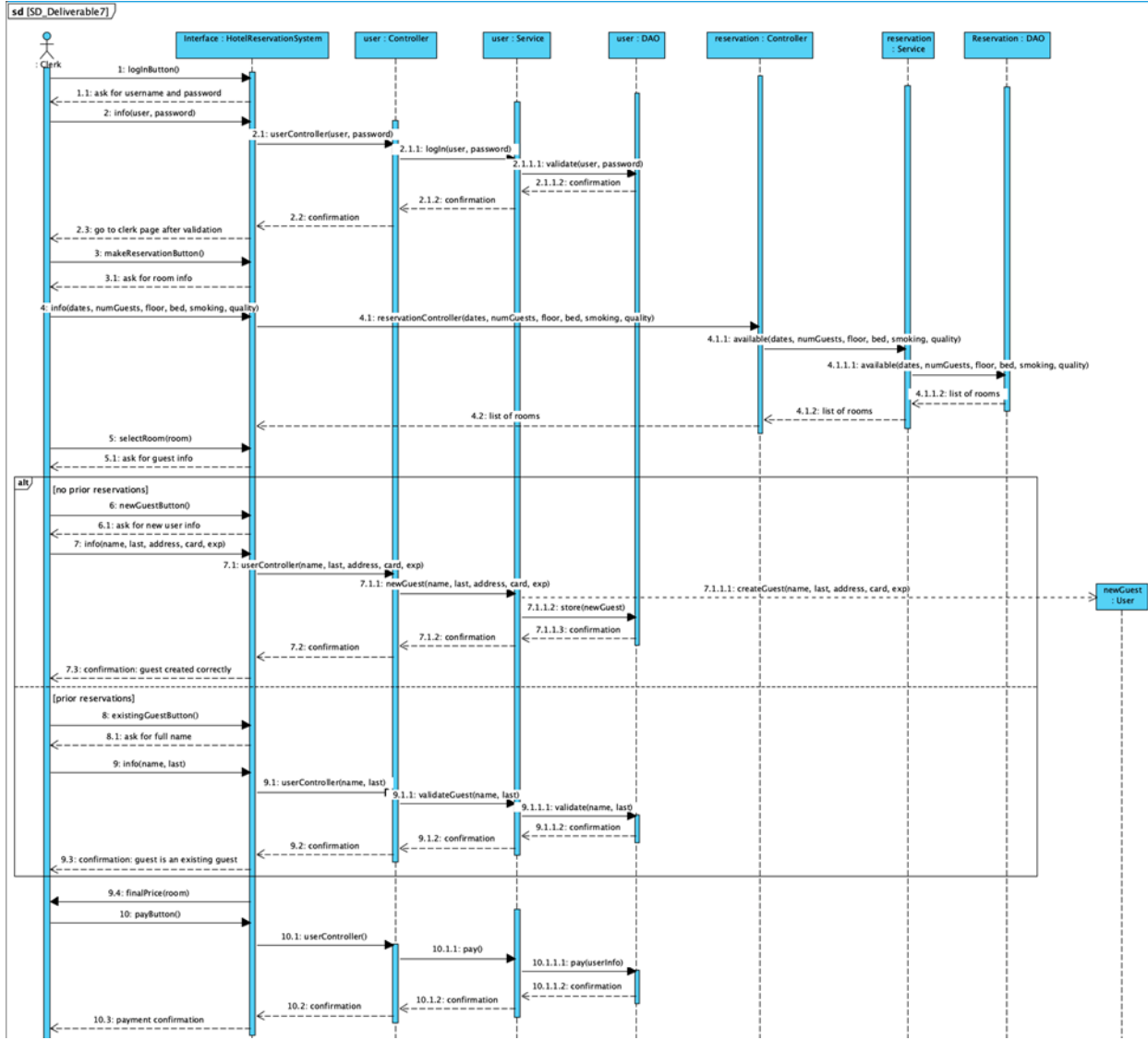
ID	UC – Guest Making a Reservation through Clerk
Scope	Hotel Reservation System
Level	User Goal
Stakeholders and Interests	<p>Guest</p> <ul style="list-style-type: none"> - Wants to make a reservation and receive relevant information about room options, pricing, and hotel rules. <p>Clerk</p> <ul style="list-style-type: none"> - Needs to assist the Guest in making a reservation, ensure the accuracy of the information provided, and follow up with the system's functionality. <p>Admin</p> <ul style="list-style-type: none"> - Assists the Clerk and Guest if needed.
Precondition	<ol style="list-style-type: none"> 1. Clerk has access to manual reservation documents, just in case the system fails. 2. Rooms are available for reservation.
Postcondition	<ol style="list-style-type: none"> 1. The reservation is confirmed in the system or stored manually for future entry.

Main Success Scenario	<ol style="list-style-type: none"> 1. Guest wants to make a reservation. 2. Clerk logs into the system. 3. Clerk asks the Guest for reservation details (dates, number of guests, floor preference, type of bed, smoking/non-smoking status, quality level). 4. Clerk checks room availability in the system. 5. Clerk provides the Guest with available room options and prices. 6. Guest chooses a room. 7. Clerk fills out the reservation information in the system, or looks up info if Guest has made a reservation in the past. 8. Clerk informs the Guest of the hotel rules and the final reservation price. 9. Guest agrees with the terms and successfully pays for the reservation. 10. Clerk confirms the reservation in the system. 11. System confirms and saves the reservation. 12. Clerk provides the Guest with confirmation details.
------------------------------	--

<p>Alternate Paths</p>	<p>*.a System unavailable</p> <ol style="list-style-type: none"> 1. Clerk will make a manual reservation. 2. Clerk updates the system with the reservation details once it is back online. <p>*.b System has no connection to network</p> <ol style="list-style-type: none"> 1. Clerk asks Admin to reset network <p>2.a Clerk fails to login</p> <ol style="list-style-type: none"> 1. Clerk either tries to login again or changes password <p>3.a Guest provides incomplete reservation details</p> <ol style="list-style-type: none"> 1. Clerk requests the missing information (e.g., missing date, number of guests, or special requests). 2. Once the Guest provides the missing details, the Clerk continues the reservation process. <p>9.a Guest does not agree with terms</p> <ol style="list-style-type: none"> 1. The Clerk discards the reservation and may suggest alternative room or pricing options if available. <p>9.b Payment declines</p> <ol style="list-style-type: none"> 1. Clerk informs the Guest that their credit card information was declined during the reservation process.
-------------------------------	---

	<p>2. Guest provides a different payment method, and the Clerk reattempts the payment.</p> <p>3. If successful, the reservation is confirmed. If unsuccessful, the reservation is discarded.</p>
--	--





M	System
	login(username, password) availableRooms(dates, numberOfGuests, floor, bed, smoking, quality) selectRoom(room) newGuestInfo(name, lastName, address, creditCardNum, expDate) validateInfo(name, lastName) pay()

Contract CO1: login(username, password)

Operation	logIn(username, password)
Cross References	<p>Guest Making a Reservation through Clerk</p> <p>Admin Creates a Clerk Account</p> <p>Corporate Guest Making a Reservation through Clerk</p>
Pre-conditions	<ul style="list-style-type: none"> ▫ The Clerk is not logged in and needs system access to assist the Guest.
Post-conditions	<ul style="list-style-type: none"> ▫ The Clerk is successfully logged into the system. (system state modification) ▫ A Clerk session is created and is associated with the Clerk instance. (association formed)

Contract CO2: availableRooms(dates, numberOfGuests, floor, bed, smoking, quality)

Operation	availableRooms(dates, numberOfGuests, floor, bed, smoking, quality)
Cross References	<p>Guest Making a Reservation through Clerk</p> <p>Corporate Guest Making a Reservation through Clerk</p>
Pre-conditions	<ul style="list-style-type: none"> ▫ The Clerk is logged in. ▫ The Guest has provided necessary reservation details (dates, number of guests, preferences)

Post-conditions	<ul style="list-style-type: none"> ▫ A list of available rooms is retrieved based on the Guest's preferences and date. (retrieval of data) ▫ The Clerk can view and present these options to the Guest.
------------------------	---

Contract CO3: selectRoom(room)

Operation	selectRoom(room)
Cross References	<p>Guest Making a Reservation through Clerk</p> <p>Corporate Guest Making a Reservation through Clerk</p>
Pre-conditions	<ul style="list-style-type: none"> ▫ The Guest has chosen a room from the available list provided by the Clerk.
Post-conditions	<ul style="list-style-type: none"> ▫ A Room instance room is associated with the Guest's reservation. (association formed) ▫ The room is reserved temporarily in the system to avoid double-booking. (system reservation)

Contract CO4: newGuestInfo(name, address, creditCardNum, expDate)

Operation	newGuestInfo(name, address, creditCardNum, expDate)
Cross References	<p>Guest Making a Reservation through Clerk</p> <p>Corporate Guest Making a Reservation through Clerk</p>
Pre-conditions	<ul style="list-style-type: none"> ▫ The Guest has provided their personal information and credit card details for the reservation. ▫ The Clerk is entering this information on behalf of the Guest.
Post-conditions	<ul style="list-style-type: none"> ▫ A new Guest instance guest is created and stored. (instance creation) ▫ guest.name, guest.address, guest.creditCardNum, guest.expDate are stored in the system. (attribute modification) ▫ guest is associated with the reservation. (association formed)

Contract CO5: validateInfo(name, address)

Operation	validateInfo(name, address)
Cross References	<p>Guest Making a Reservation through Clerk</p> <p>Corporate Guest Making a Reservation through Clerk</p>

Pre-conditions	<ul style="list-style-type: none"> ▫ The Guest's personal information (name, address, credit card) has been entered.
Post-conditions	<ul style="list-style-type: none"> ▫ The system validates the provided information for accuracy and completeness. (validation) ▫ The validated information is confirmed and associated with the Guest and reservation. (association formed) ▫ If validation fails, the Clerk is notified to correct the information.

Contract CO6: pay()

Operation	pay()
Cross References	Guest Making a Reservation through Clerk
Pre-conditions	<ul style="list-style-type: none"> ▫ The Guest's payment information has been validated.

Post-conditions	<ul style="list-style-type: none">▫ A Payment instance pay is created and stored. (instance creation)▫ pay is associated with the Guest's reservation. (association formed)▫ The amount is charged to the Guest's payment method, and a confirmation of payment is provided. (attribute modification)▫ If the payment fails, the Clerk is notified to reattempt or use a different method.
------------------------	---