

HOTEL RESERVATION SYSTEM

Phase I

Resubmission

SWE 621 – Spring 2013 Term Project

Project Team

Adarsh Ghanta

Matthias Mohammud

Shaeq Khan

Vividh S. Viswanatha

Table of Contents

#	Question/Topic	Page #
1	Correction to Phase 1	
a	Develop a Software System Context Class Model depicted on a class diagram showing how the system interfaces to the external environment. <ul style="list-style-type: none"> • Software System Context Class Model 	7
b	Develop a static model showing the entity classes in the system, attributes of the classes, and the relationships between them. <ul style="list-style-type: none"> • Static Model - Entity Classes 	8
c	Develop interaction diagrams (one for each use case), using either communication diagrams or sequence diagrams, depicting the sequence of interactions among the objects participating in each use case. Describe briefly the objects participating in each interaction diagram, identifying the object structuring criteria used. <ul style="list-style-type: none"> • Object Structuring Criteria <ul style="list-style-type: none"> → Hotel System External classes and Boundary classes → Hotel Client subsystem classes • Sequence Diagram (Interaction diagrams) <ul style="list-style-type: none"> → Make Reservation <ul style="list-style-type: none"> ▪ Client- Guaranteed, Valid Credit Card ▪ Client- Guaranteed, Invalid Credit Card ▪ Server → Change Reservation <ul style="list-style-type: none"> ▪ Client- Guaranteed, Valid Credit Card ▪ Client- Guaranteed, Invalid Credit Card ▪ Server → Cancel Reservation <ul style="list-style-type: none"> ▪ Client ▪ Server → Check-in Customer <ul style="list-style-type: none"> ▪ Client – Check-in ▪ Client – Check-in - Alternative 1 ▪ Client – Check-in - Alternative 2 ▪ Server → Check-out Customer <ul style="list-style-type: none"> ▪ Client- Credit Card payment 	9 10 11 12 12 14 15 16 17 18 19 20 21 22 23

▪ Client- Cash Payment	24
▪ Server	25
→ View Management Report	
▪ Client	26
▪ Server	27
→ Cancel No-Show Reservations	28
→ Bill Guaranteed No-show Reservations	29
d For state dependent use cases, show the state chart as well as the interaction diagram for each use case. In addition, show the alternative paths on both the interaction diagrams and state charts. Make sure that each state chart is consistent with the appropriate interaction diagram.	
• State Charts	
→ Make Reservation	30
→ Change Reservation	31
→ Cancel Reservation	32
→ Check-in Reservation	33
→ Check-out Reservation	34
→ View Management Report	35
e Develop message sequence descriptions, describing the object interactions on each interaction diagram depicted in (1c) above.	
• Sequence Description	
→ Make Reservation	
▪ Client	36
▪ Server	37
→ Change Reservation	
▪ Client	38
▪ Server	39
→ Cancel Reservation	
▪ Client	41
▪ Server	41
→ Check-in Customer	
▪ Client – Check-in	42
▪ Server	44
→ Check-out Customer	
▪ Client	45
▪ Server	46

→ View Management Report	
▪ Client	49
▪ Server	49
→ Cancel No-Show Reservations	50
→ Bill Guaranteed No-show Reservations	62
f Based on the state charts developed in 1(d), develop a complete state chart for each state dependent object.	
• Complete State Chart	63

Corrections to Phase 1

Page 2

The table of contents should include the corresponding page numbers

Corrected

Page 7

“Interacts with” instead of “Interacts”

“Communicates with” instead of “Communicates”

Corrected

Page 8

The entity “Calendar” should be associated with another entity class.

Corrected

Page 9

The software system “HotelSystem” is inconsistent with the “Hotel Reservation System” on p.7

Corrected

Page 10

Client Control in Customer Client should be different class from Client Control in Clerk Client

Corrected

Page 11

The subsystem “Hotel Service” is used. Is it a part of the software system “HotelSystem.”

What objects compose this subsystem; this is not documented anywhere although it seems to be assumed to be the server side objects.

Corrected

Page 17

Message 1.4 should be consistent with the diagram page 32 (stm ST Cancel Reservation)

Corrected

Page 19

Messages 1.8 and 1.14A.B.1 should be consistent with the diagram page 33 (stm ST Check-in)

Corrected

Page 24

Inconsistent messages with the diagram page 34 (stm ST Check-out) “room status

Updated/ print receipt"

Corrected

Page 26

Message 1.1 should be consistent with the diagram page 35 (stm ST View Management Reports)

Corrected

Page 30

Room Unavailable transition should start from Waiting for Room Availability state.

Customer details entered [guaranteed] transition should start from Waiting for customer details state.

State and transition should not have the same name: Waiting for customer choice.

Events "Customer details entered" and "customer choice". You seem to be using the condition as a parameter, not a condition.

Corrected

Page 33

The state "Print key/Information" should be "Printing key/Information."

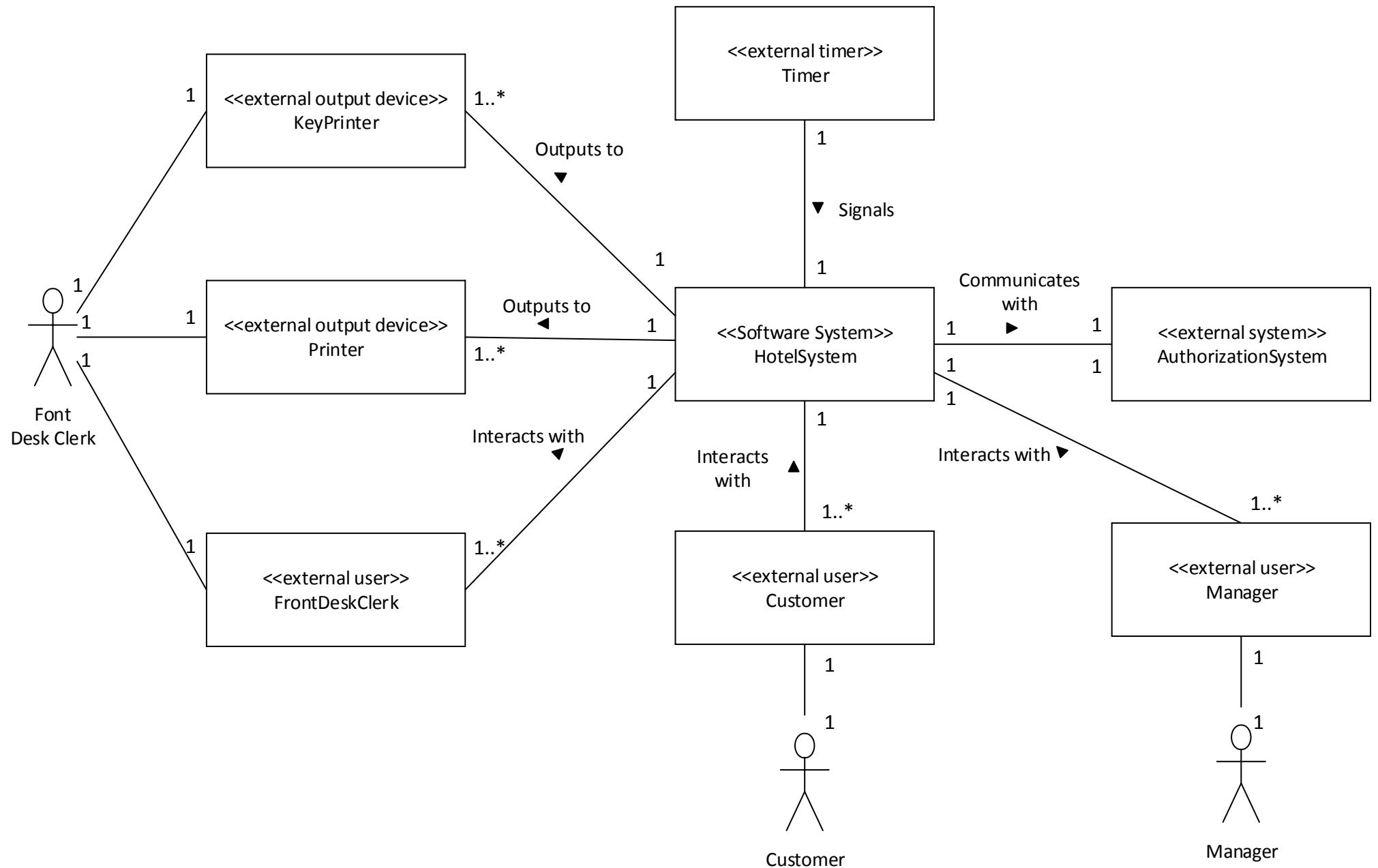
Transition "Checkin reservation selected [non guaranteed] should start from "Waiting for customer choice".

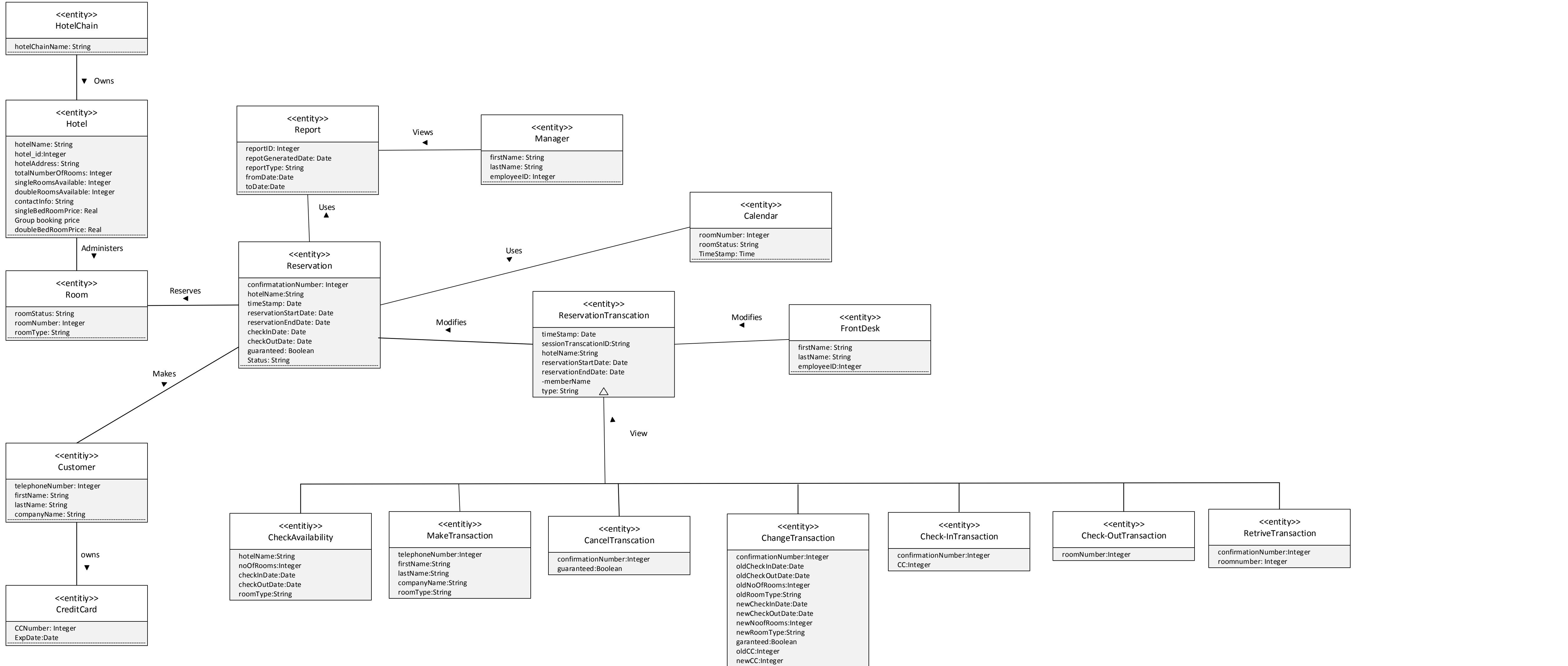
Corrected

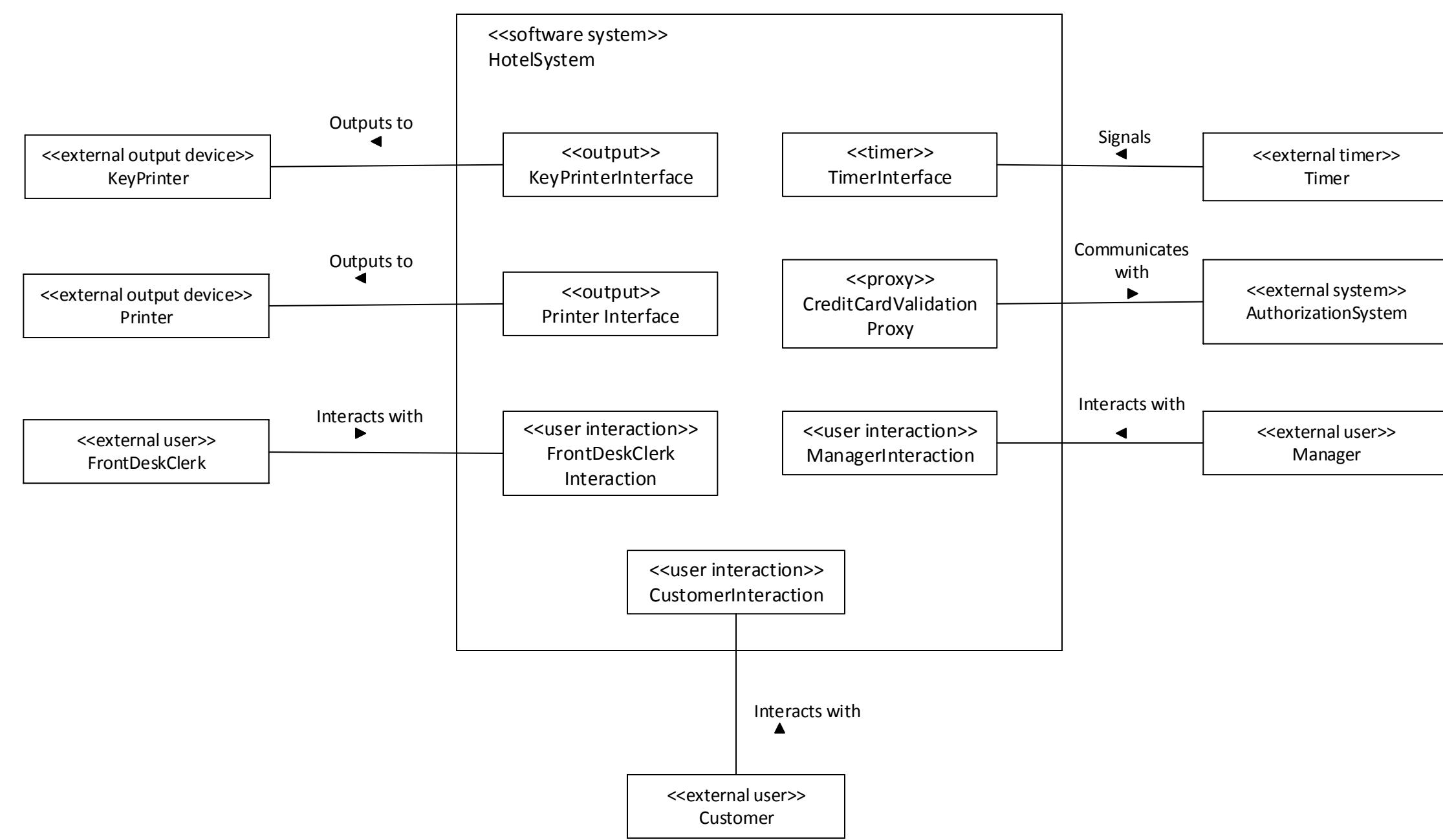
Page 34

The state "Print Receipt" should be "Printing Receipt"

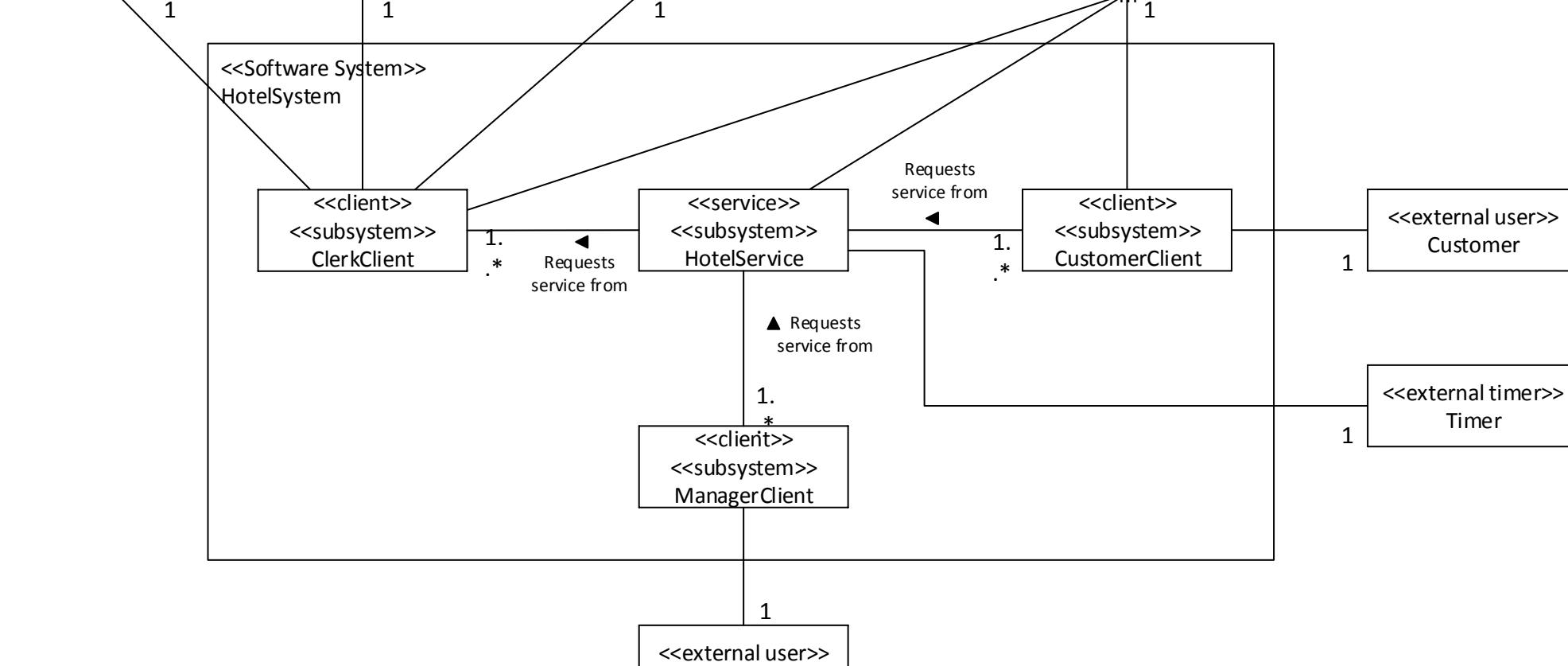
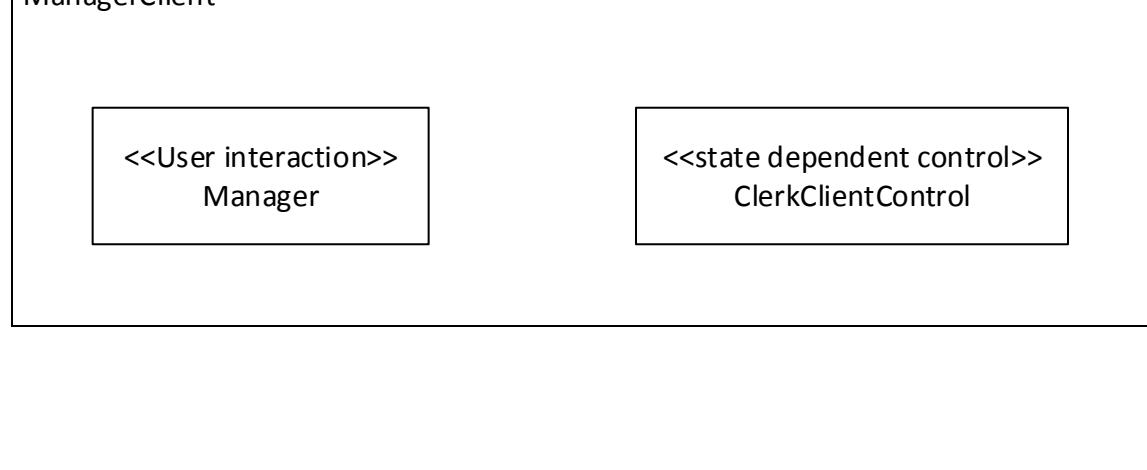
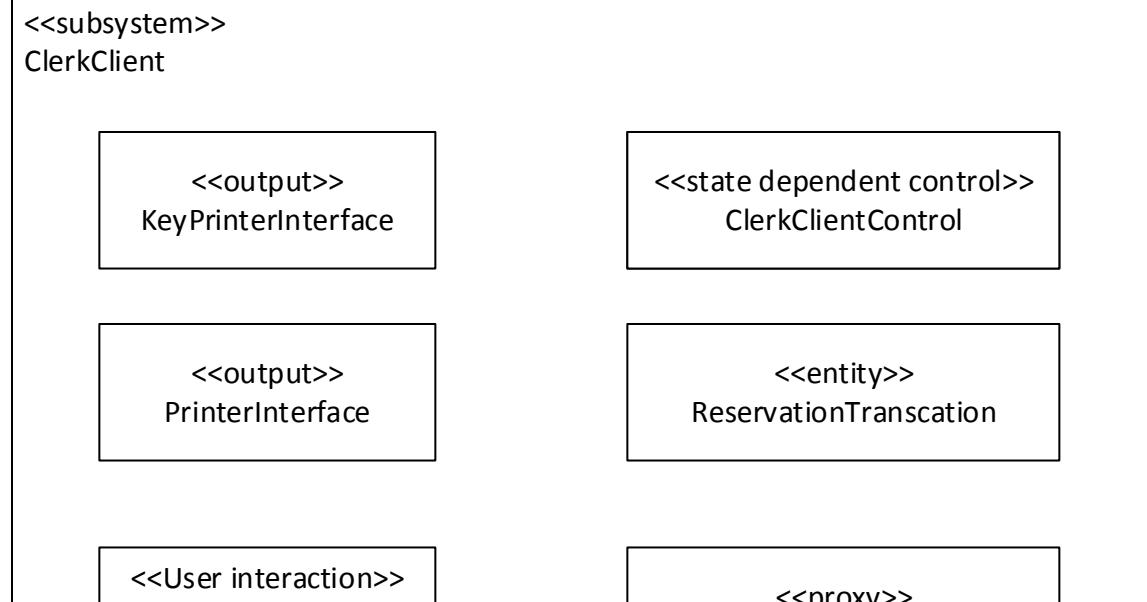
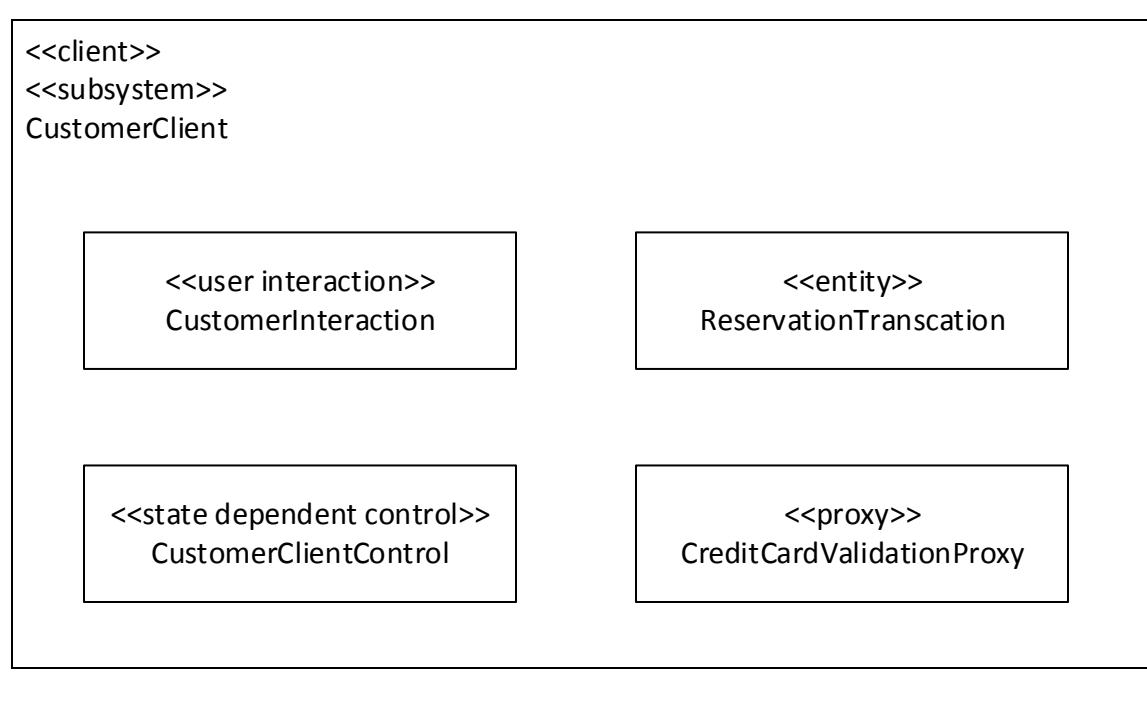
Messages 2.5, 2.6, 3.1, 3.5 on page 23 do not appear as events or actions on this statechart.



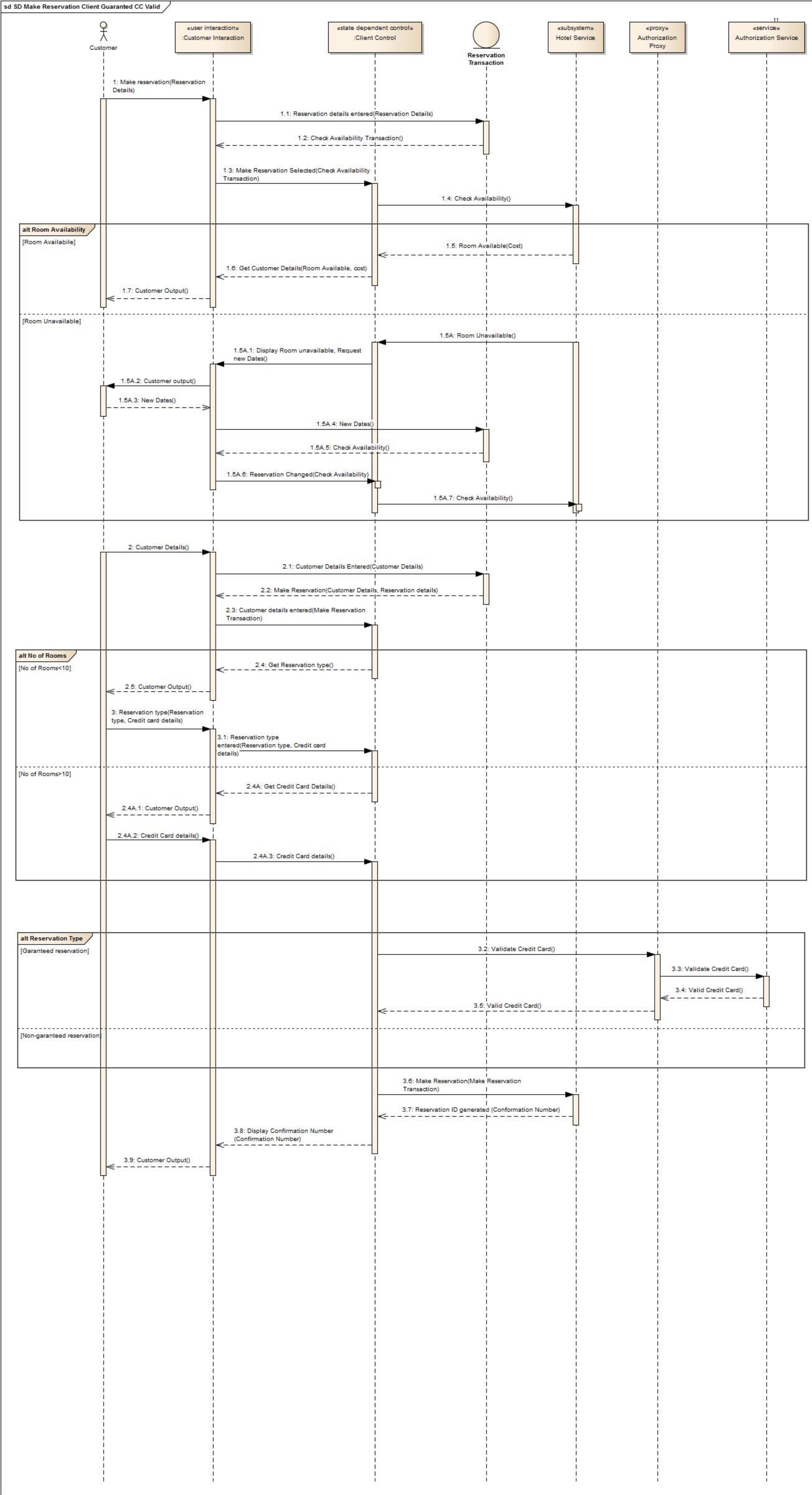


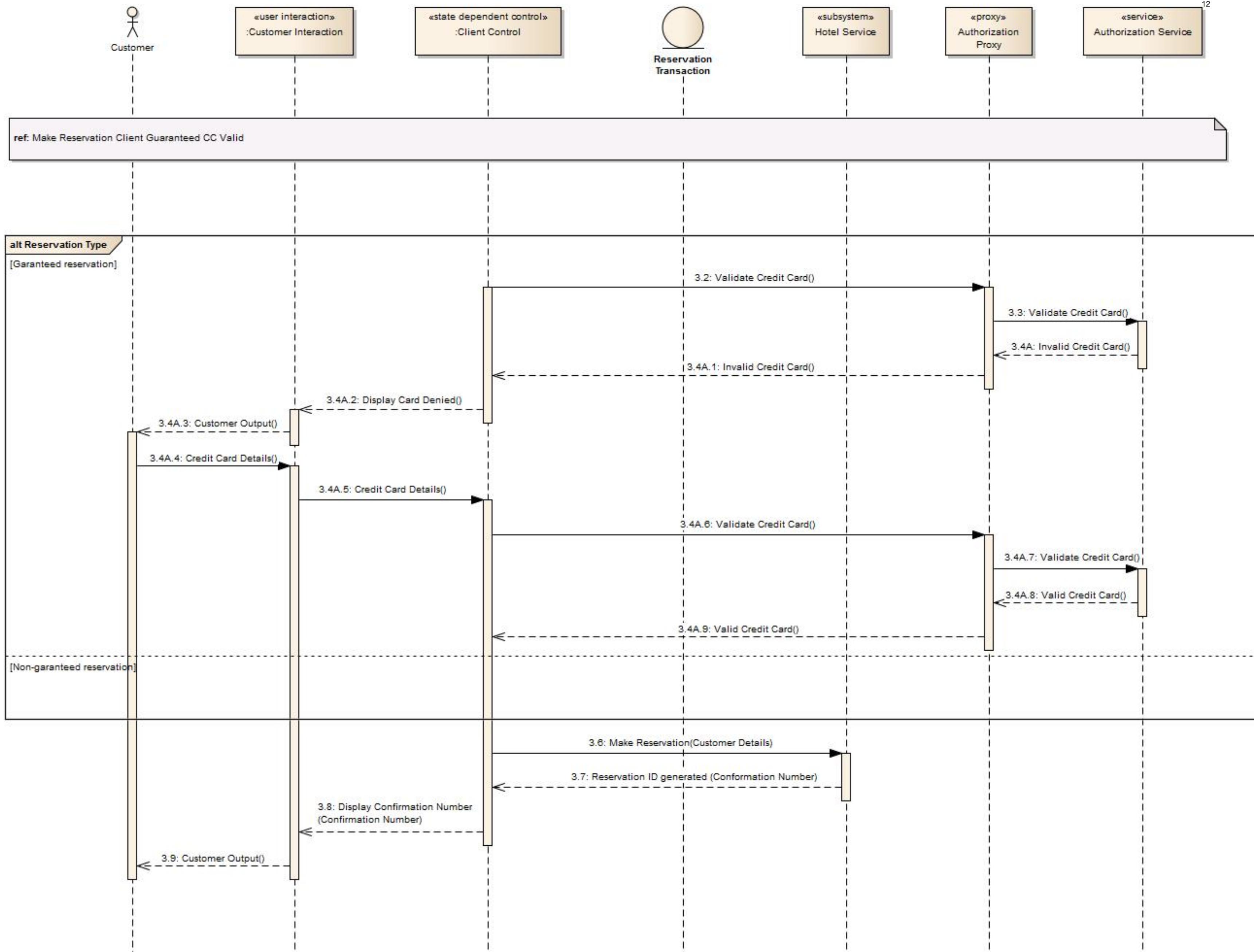


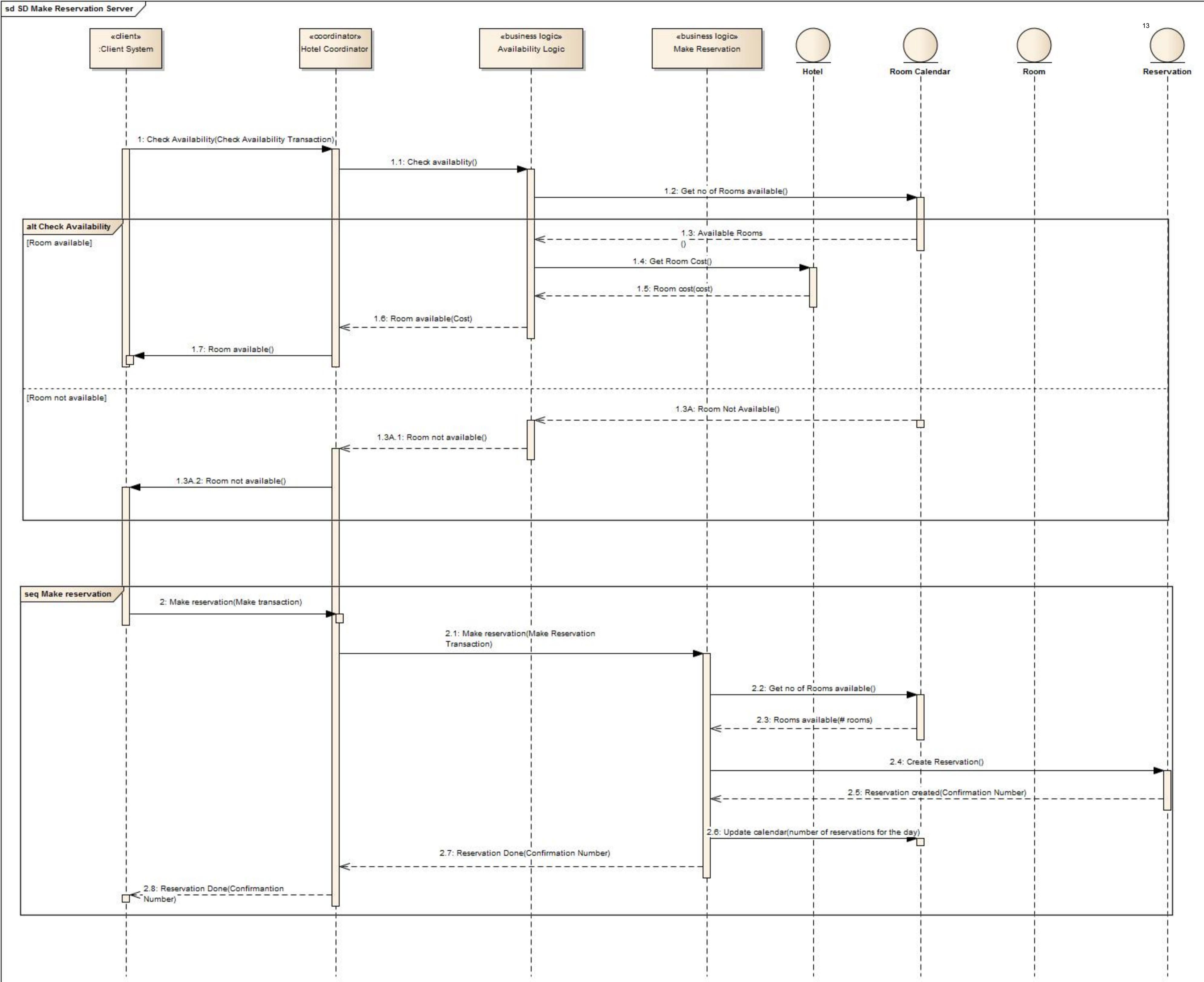
Hotel System External classes and Boundary classes

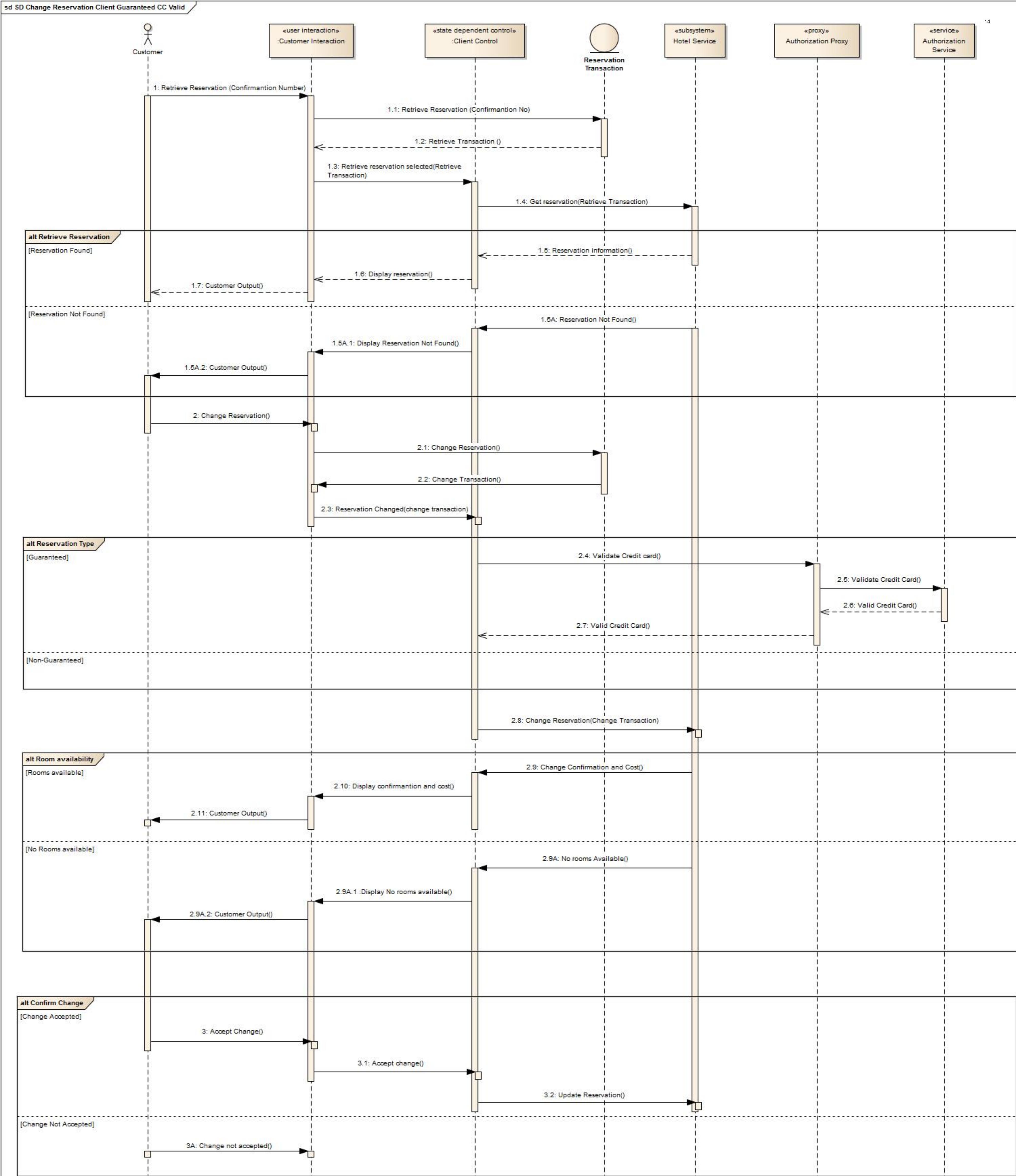


Hotel Client subsystem classes





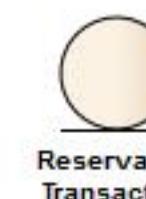






«user interaction»
:Customer Interaction

«state dependent control»
:Client Control

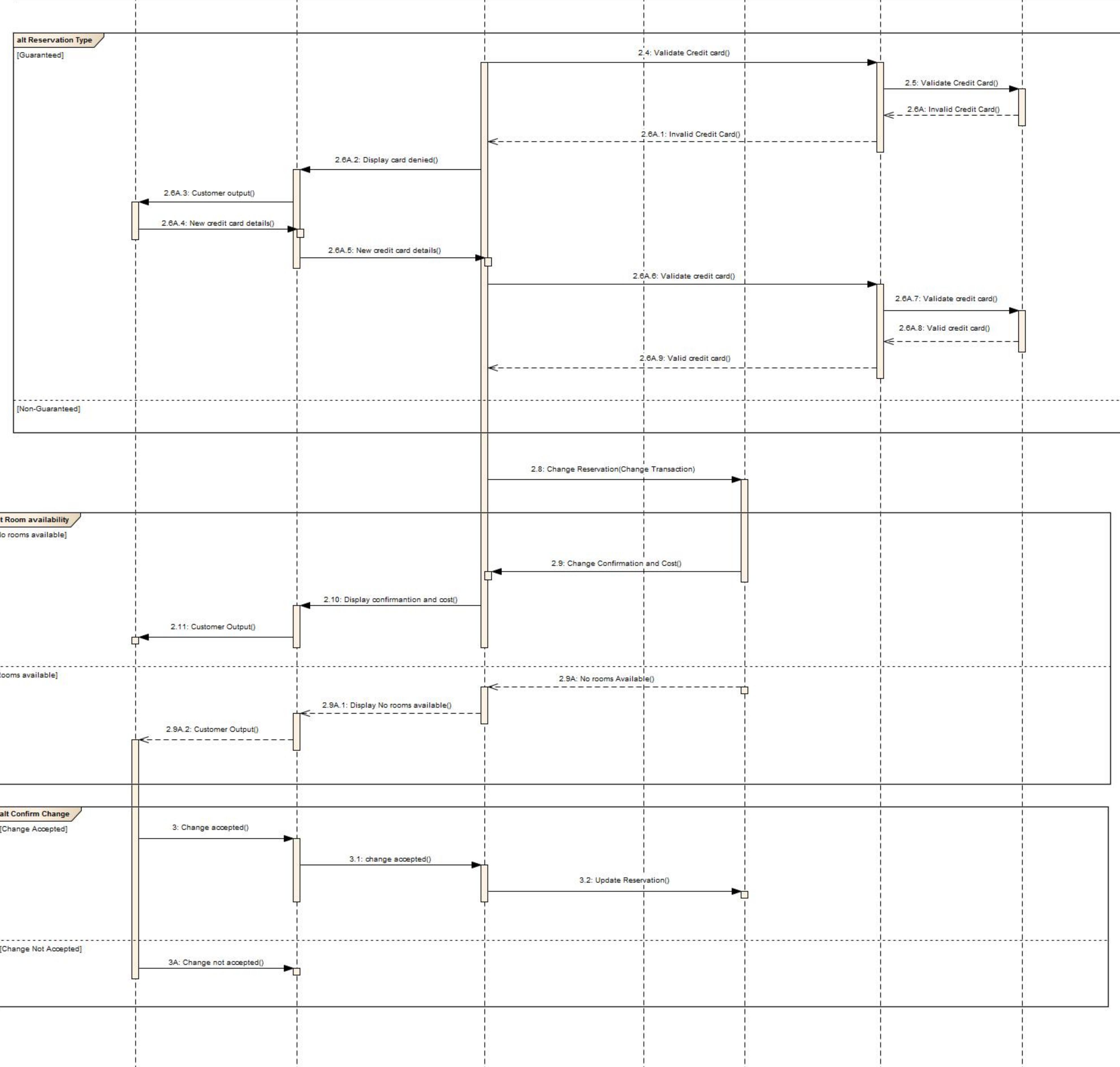


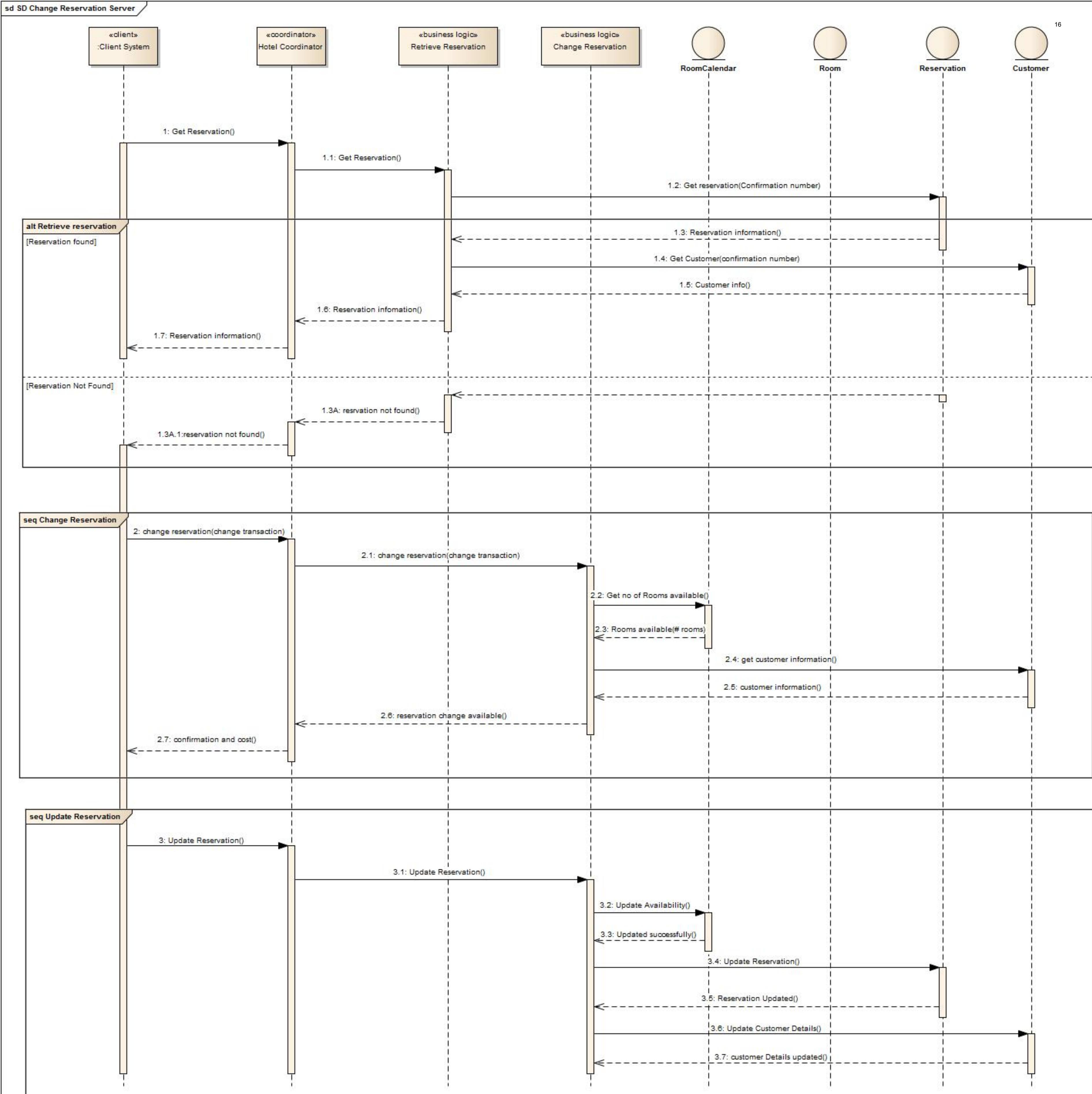
«subsystem»
Hotel Service

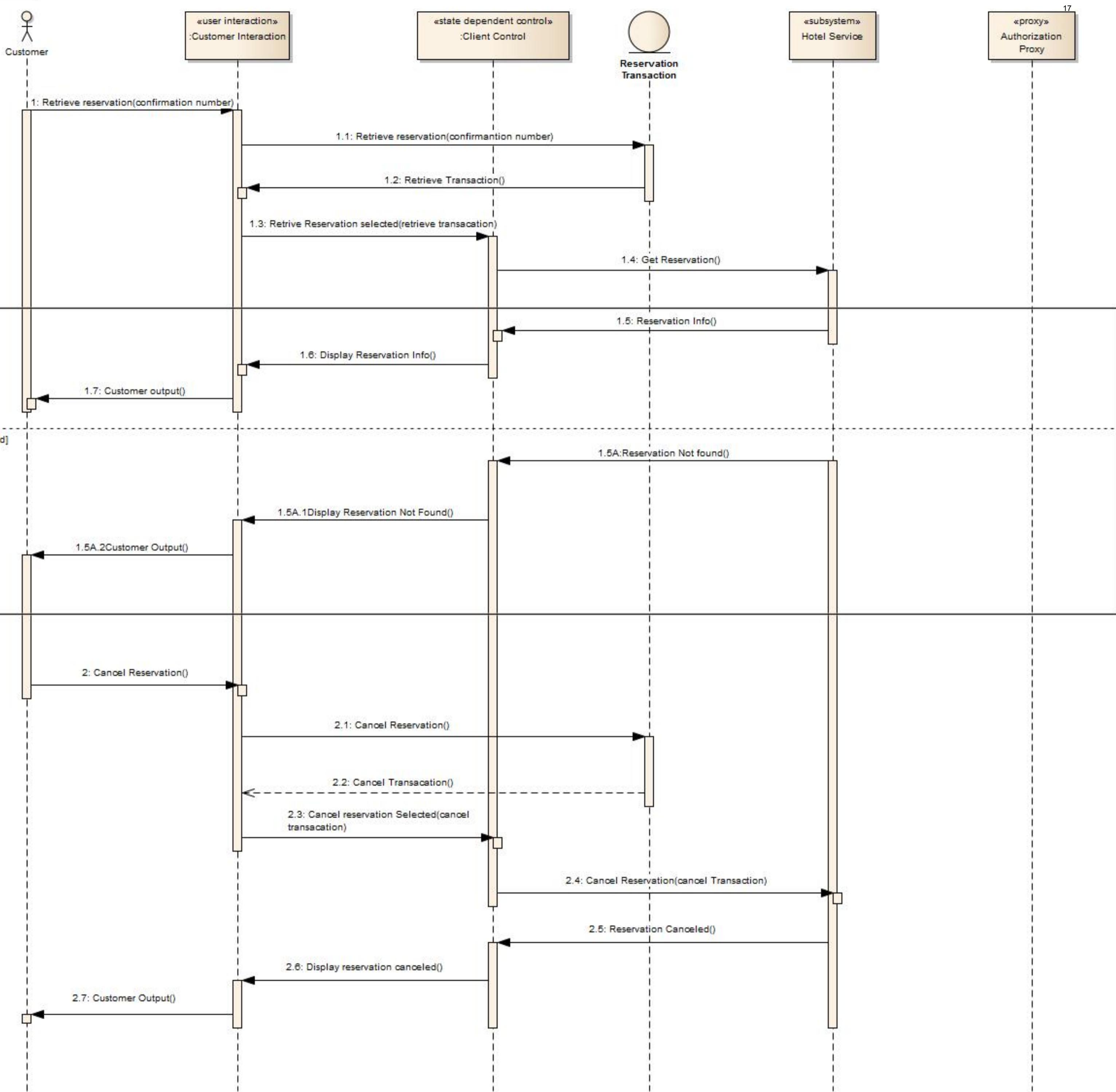
«proxy»
Authorization Proxy

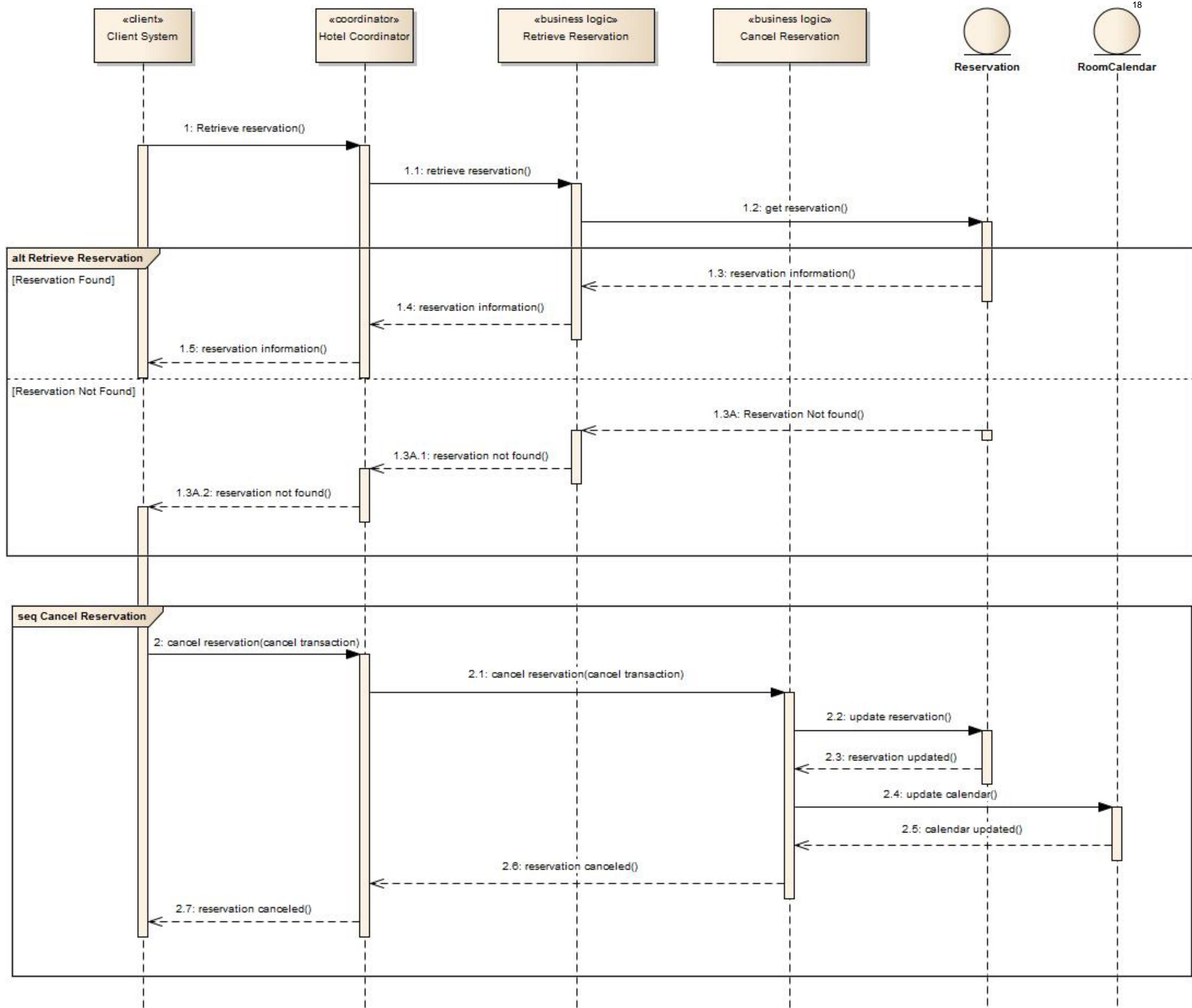
«service»
Authorization Service

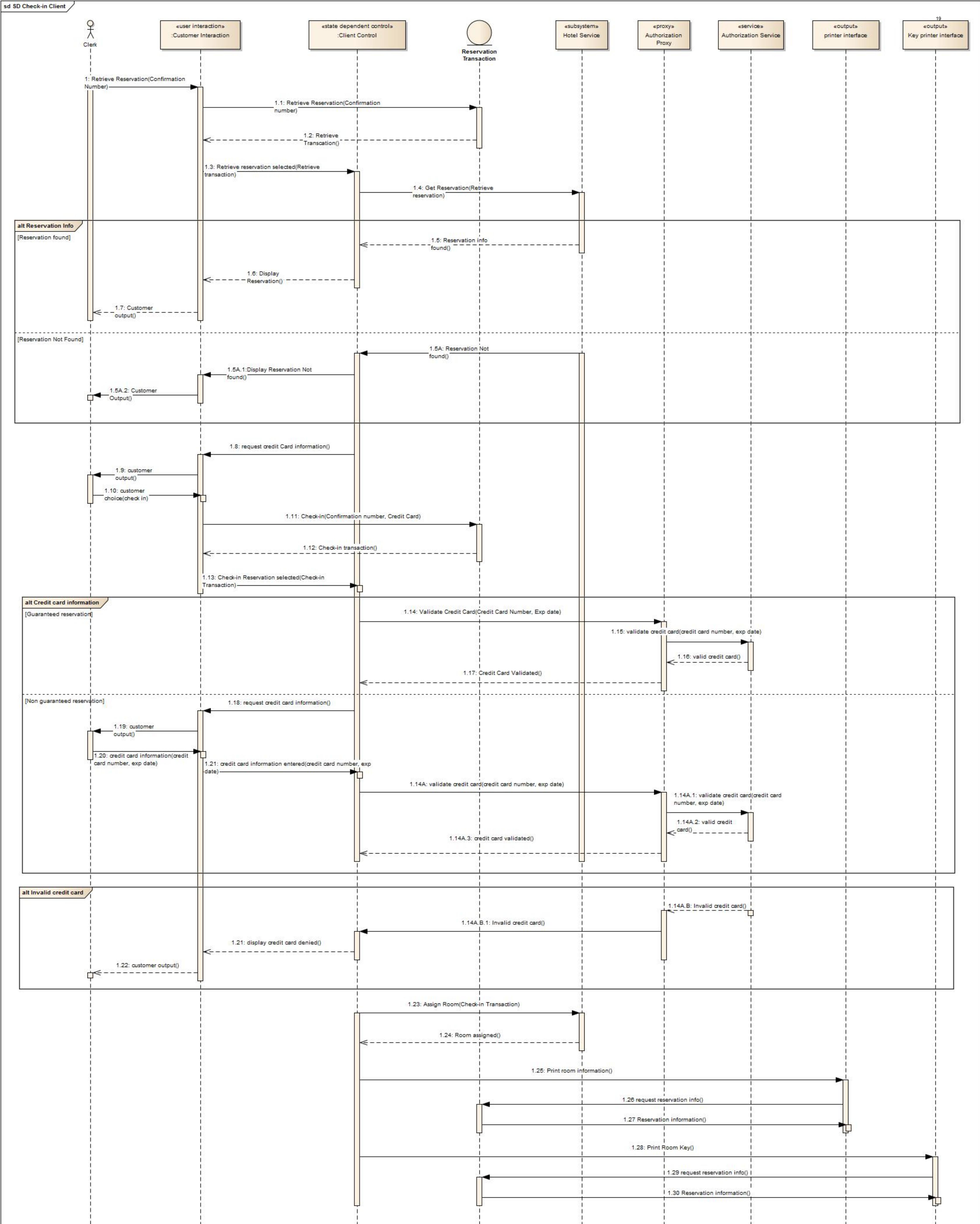
ref: Change Reservation Client Guaranteed CC Valid

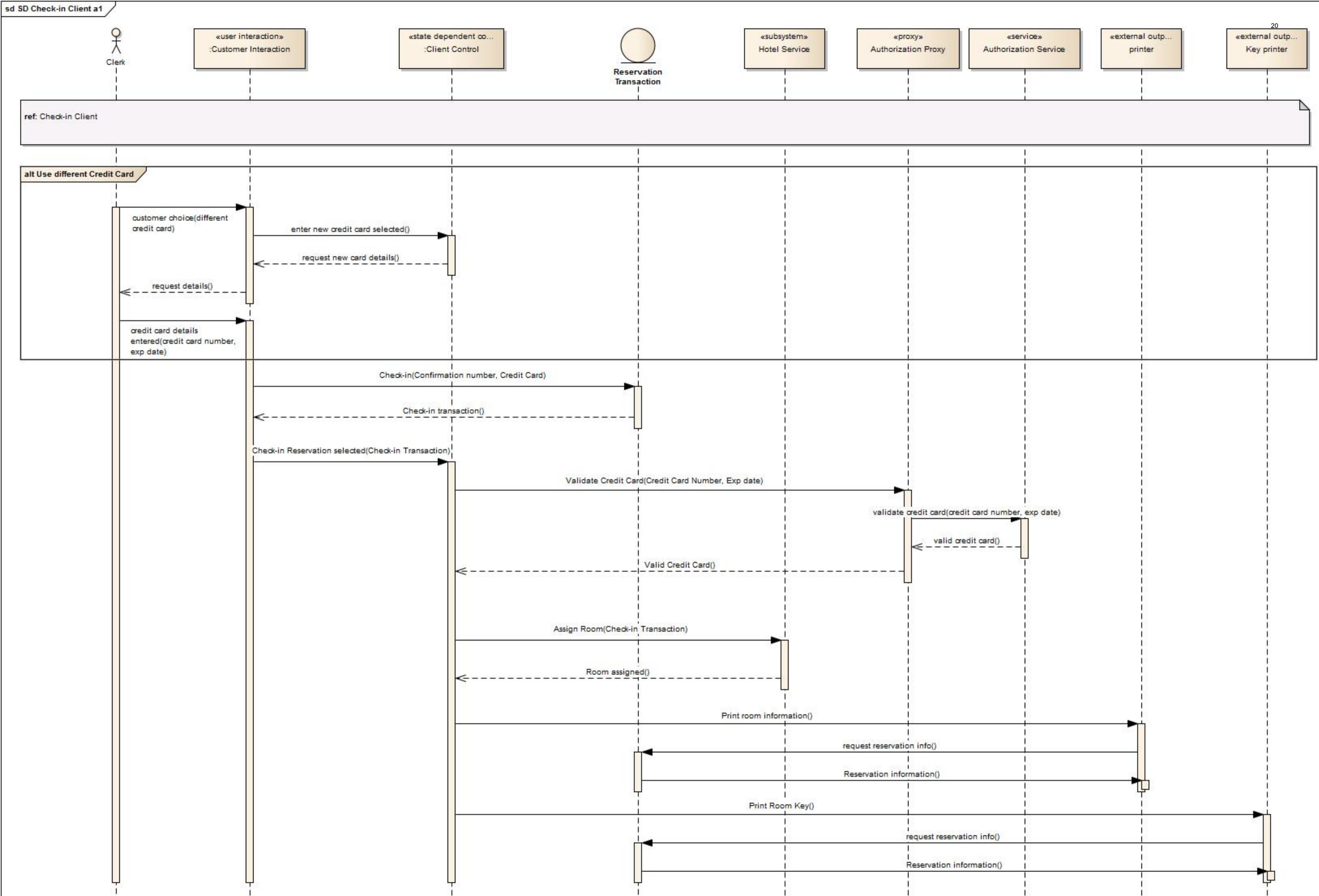


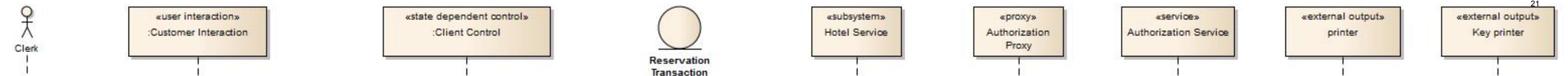








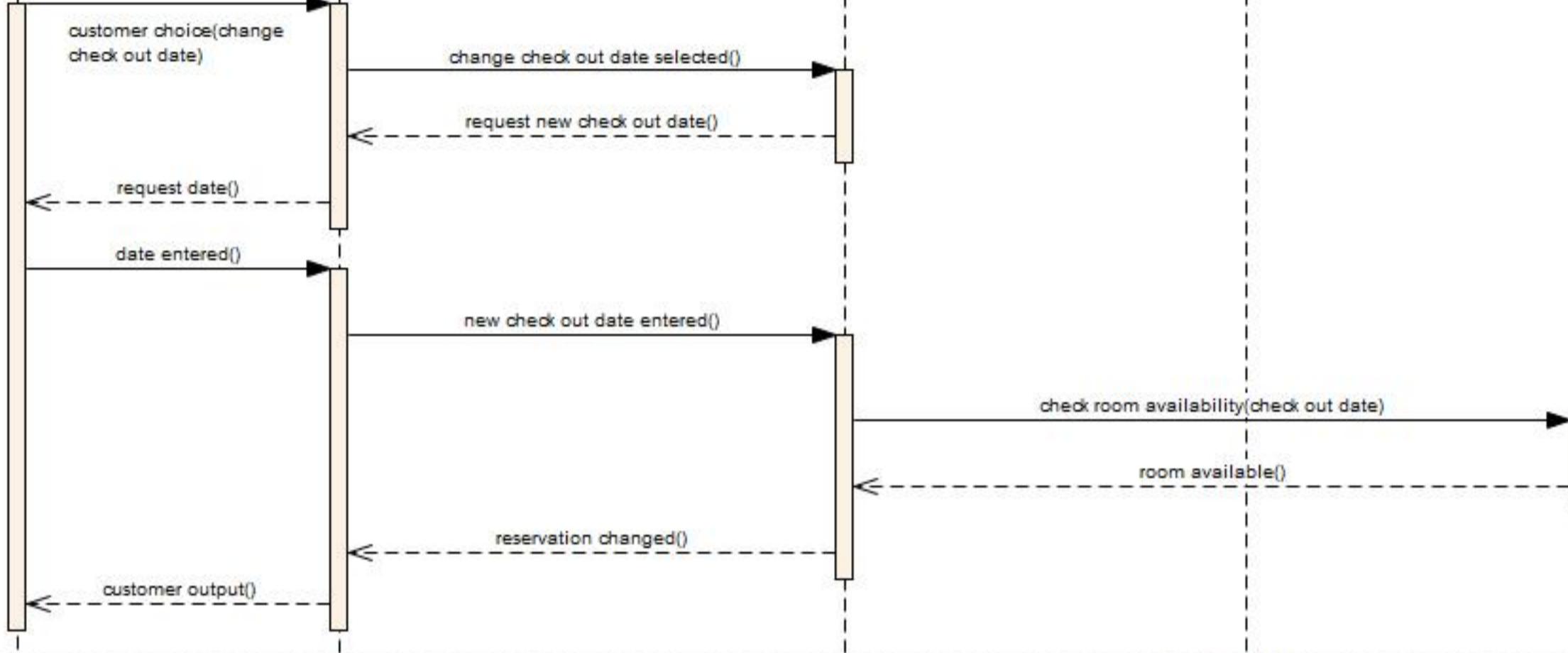




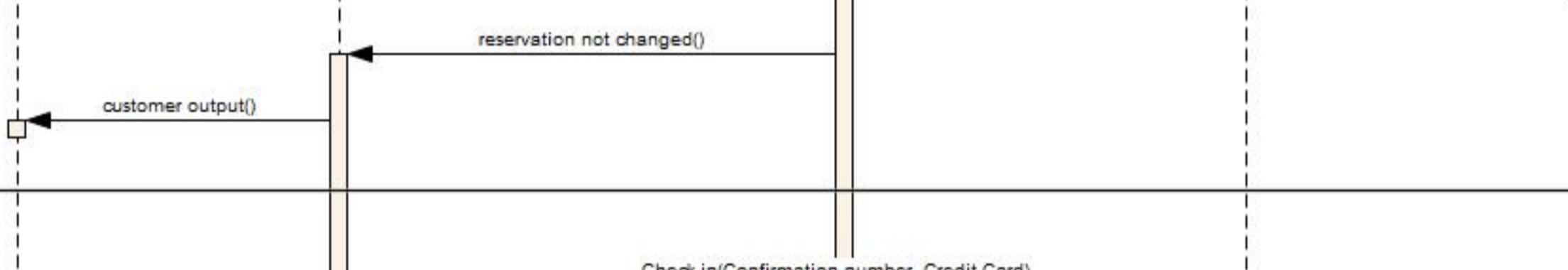
ref: Check-in Client

alt Change check out date

[Reservation changed]



[Reservation unchanged]



Check-in(Confirmation number, Credit Card)

Check-in transaction()

Check-in Reservation selected(Check-in Transaction)

Validate Credit Card(Credit Card Number, Exp date)

validate credit card(credit card number, exp date)

valid credit card()

Assign Room(Check-in Transaction)

Room assigned()

Print room information()

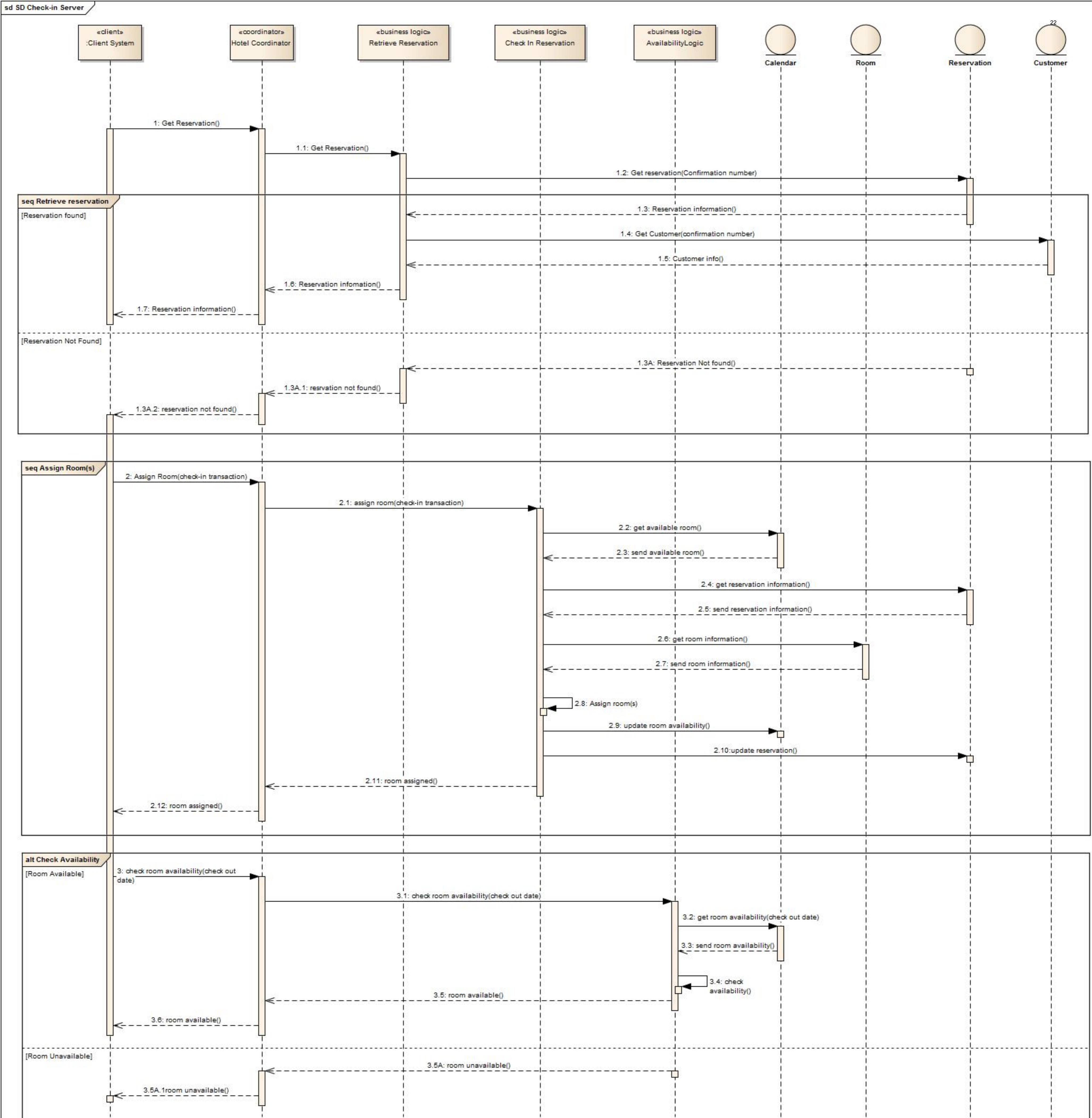
request reservation info()

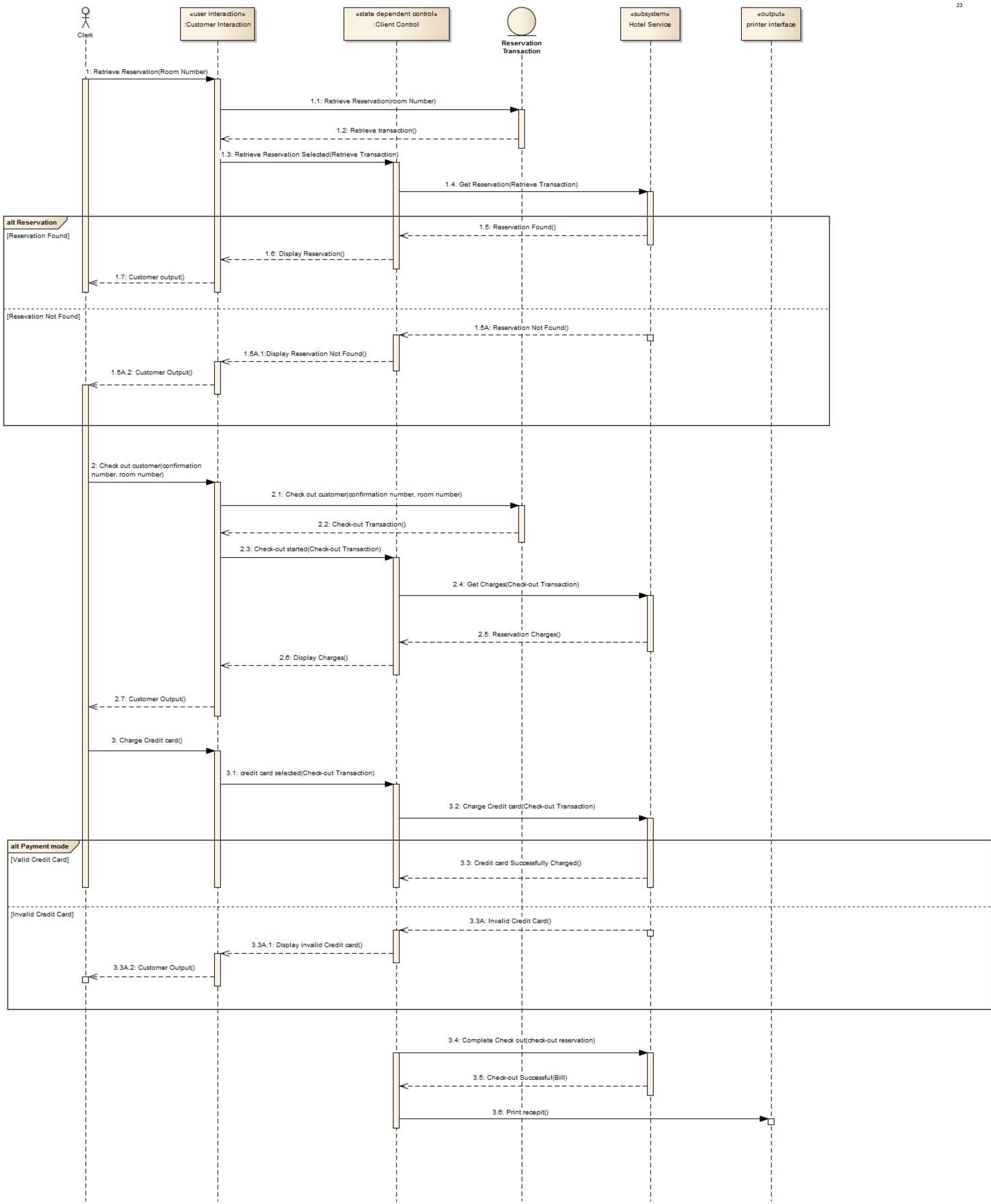
Reservation information()

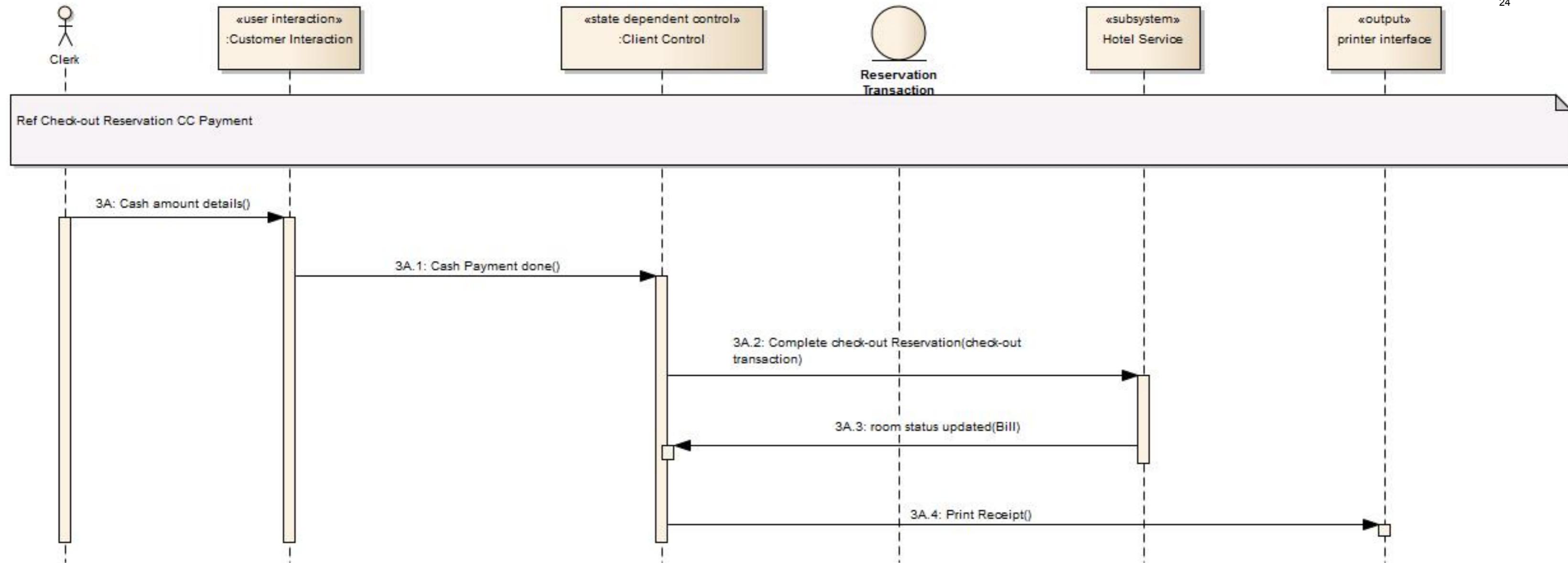
Print Room Key()

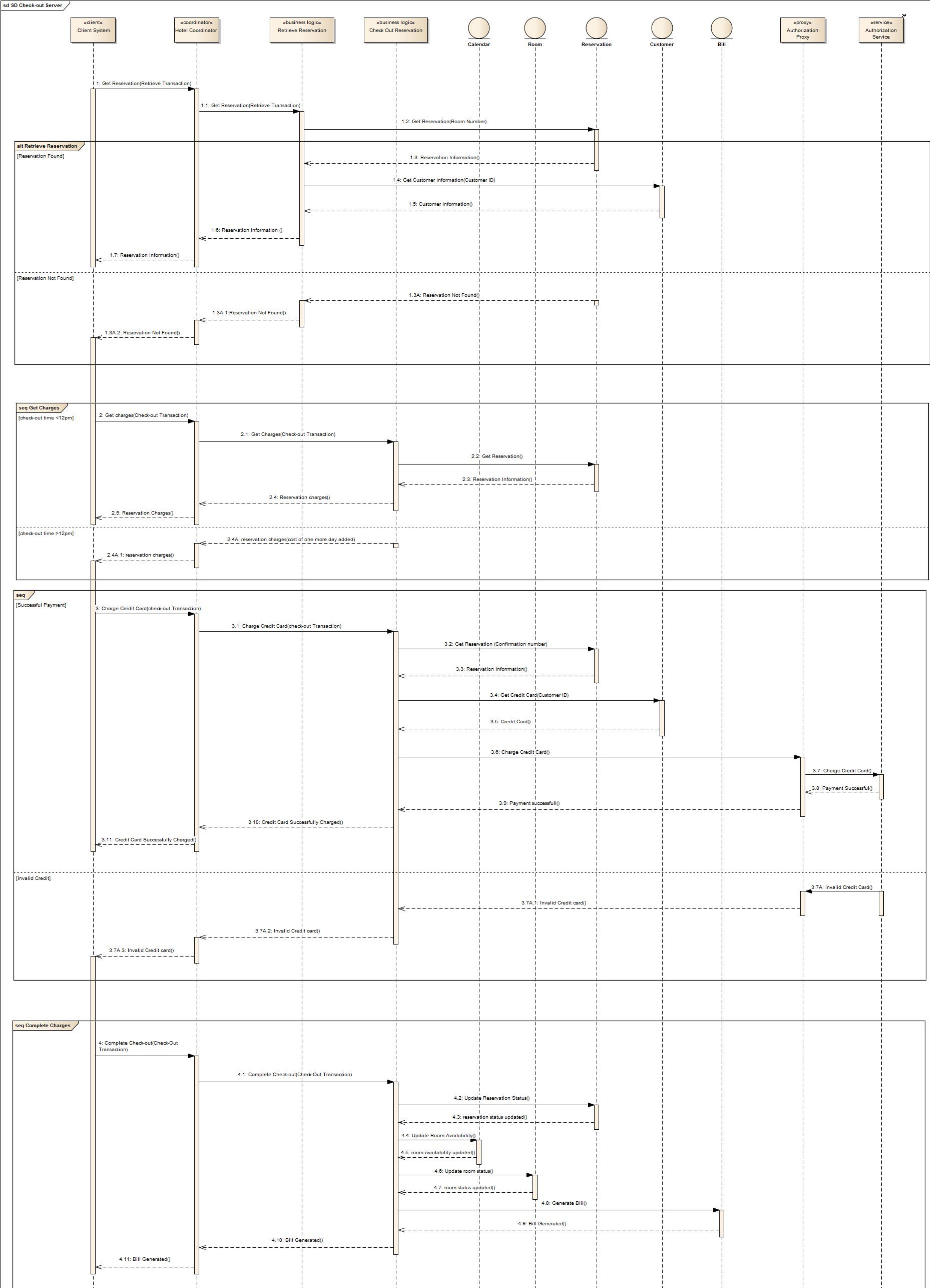
request reservation info()

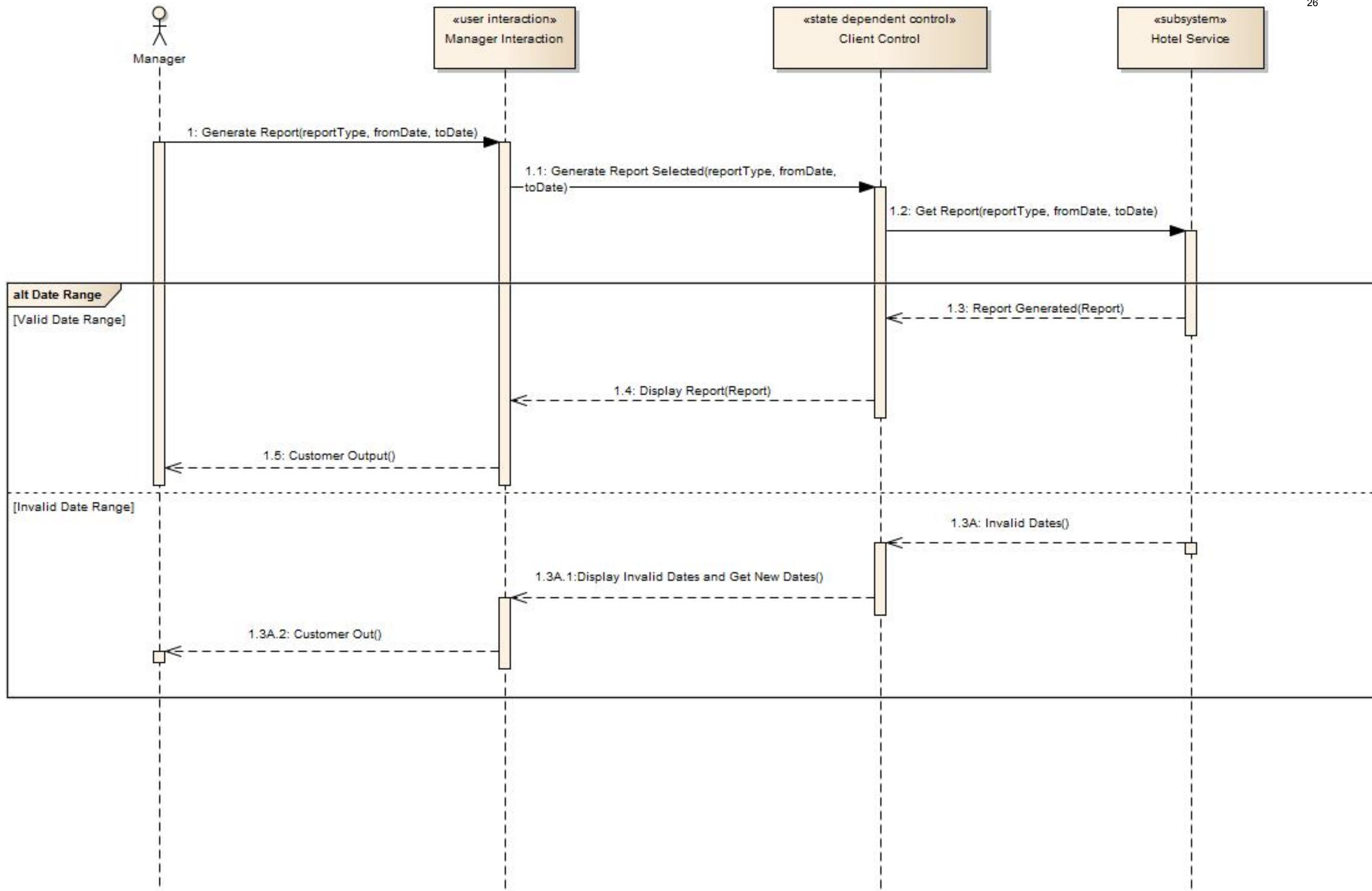
Reservation information()

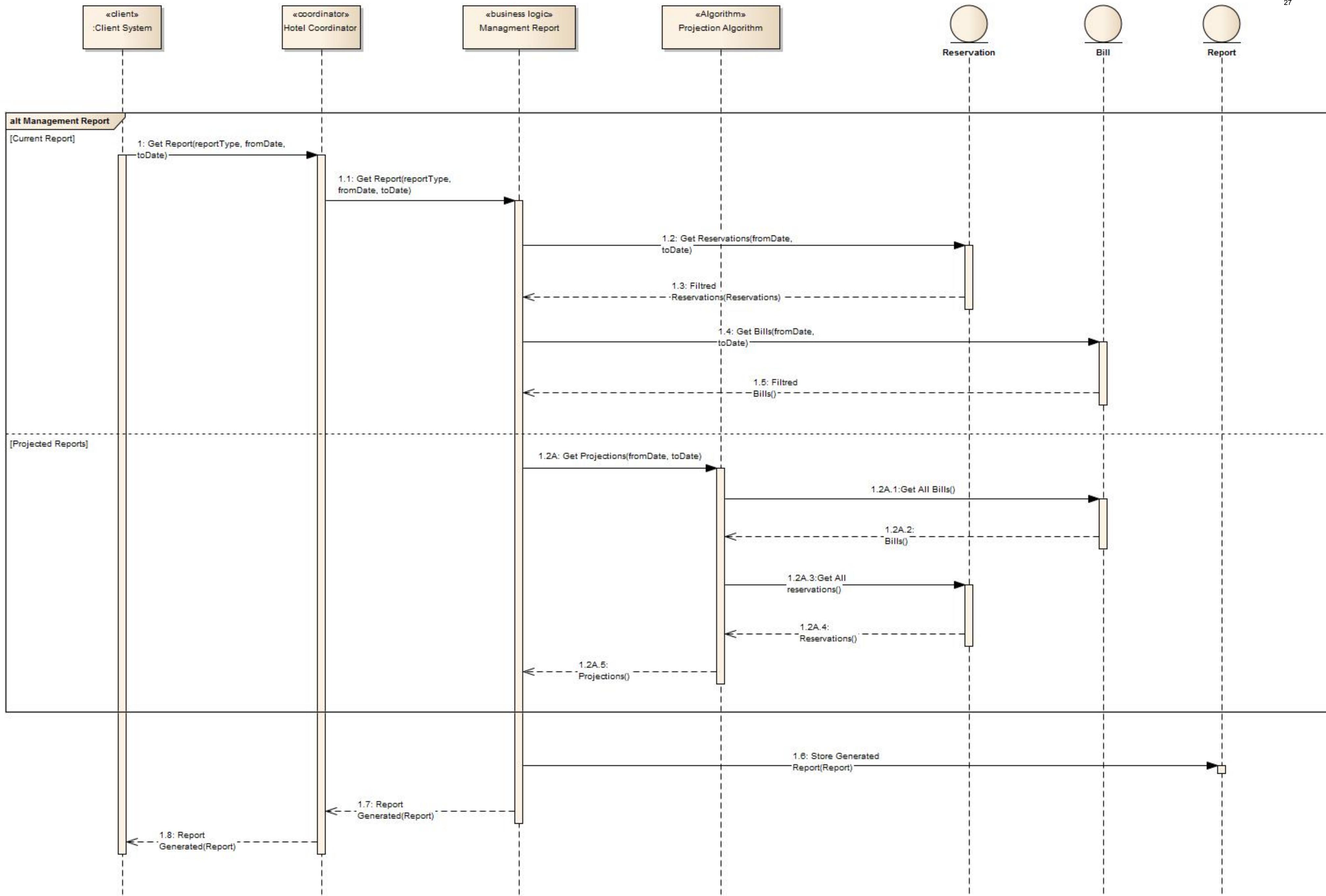


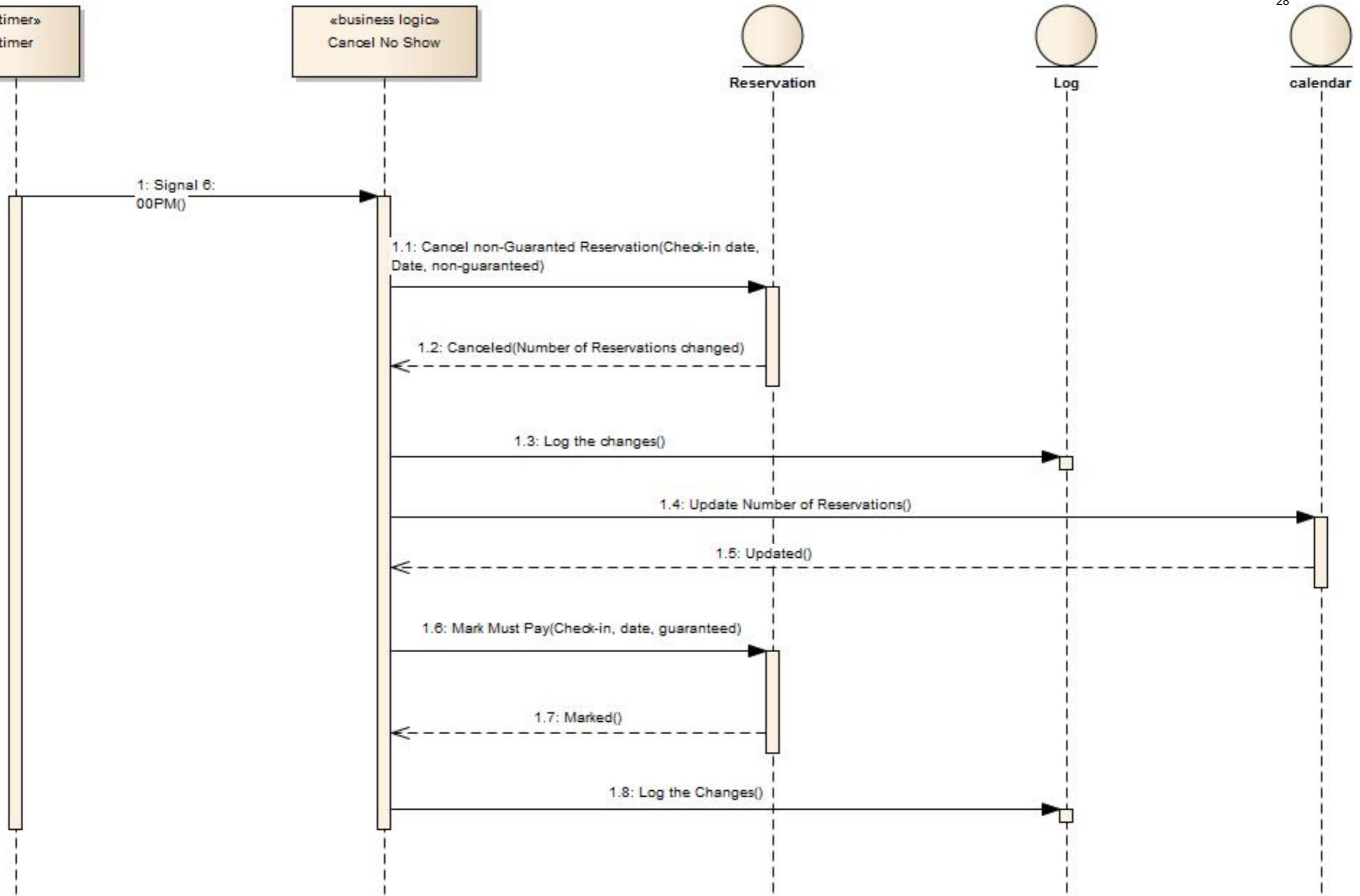


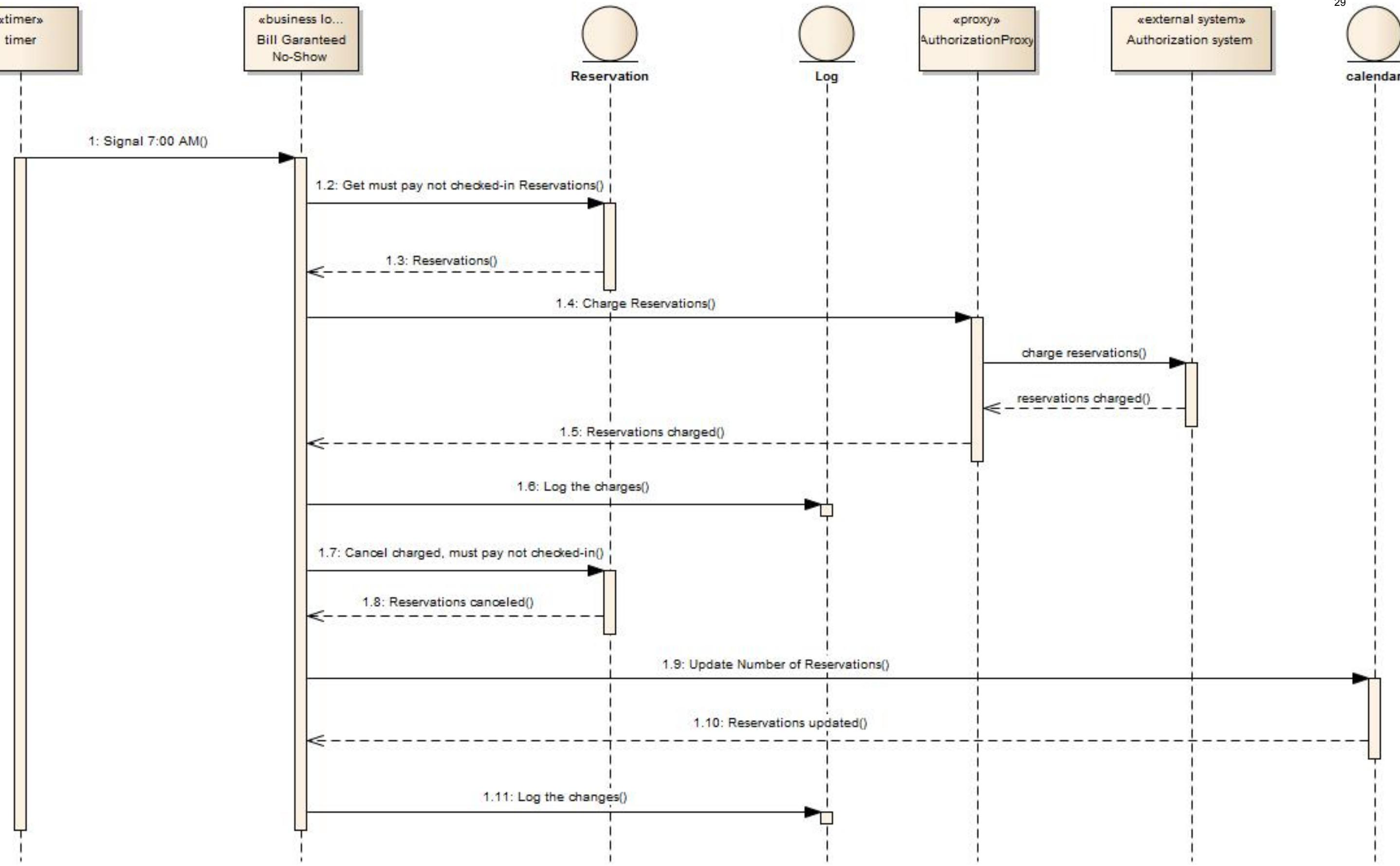


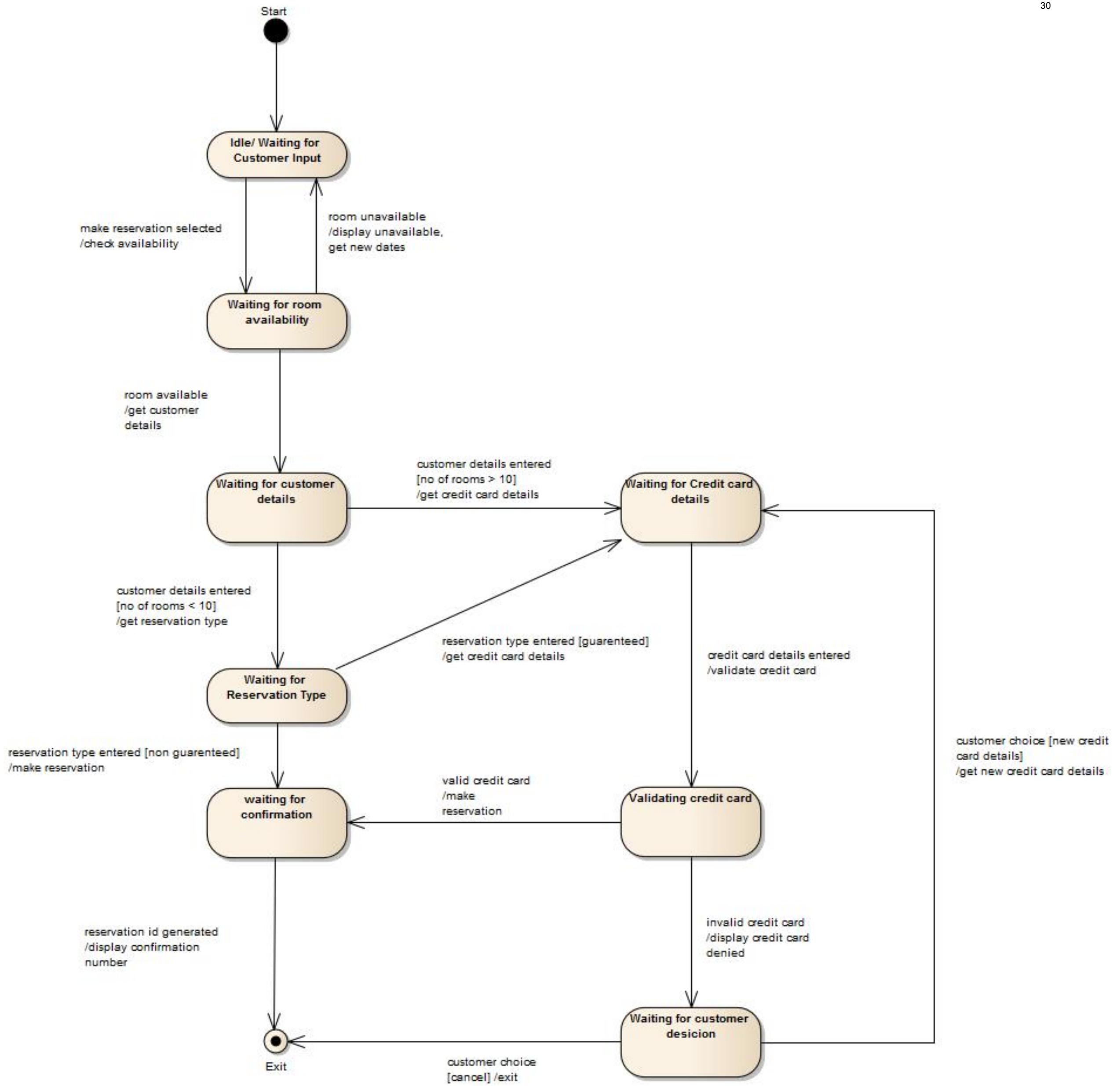


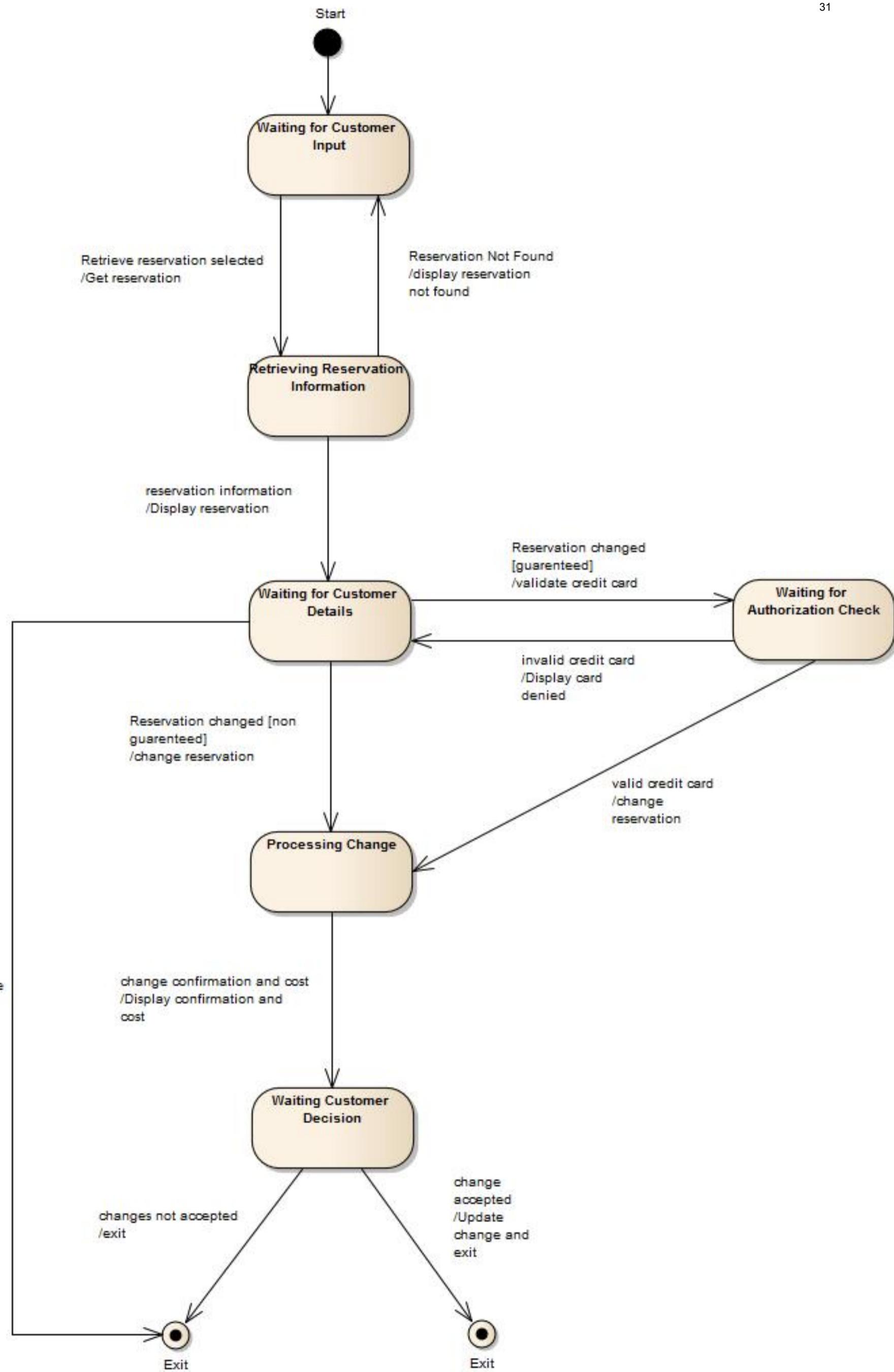


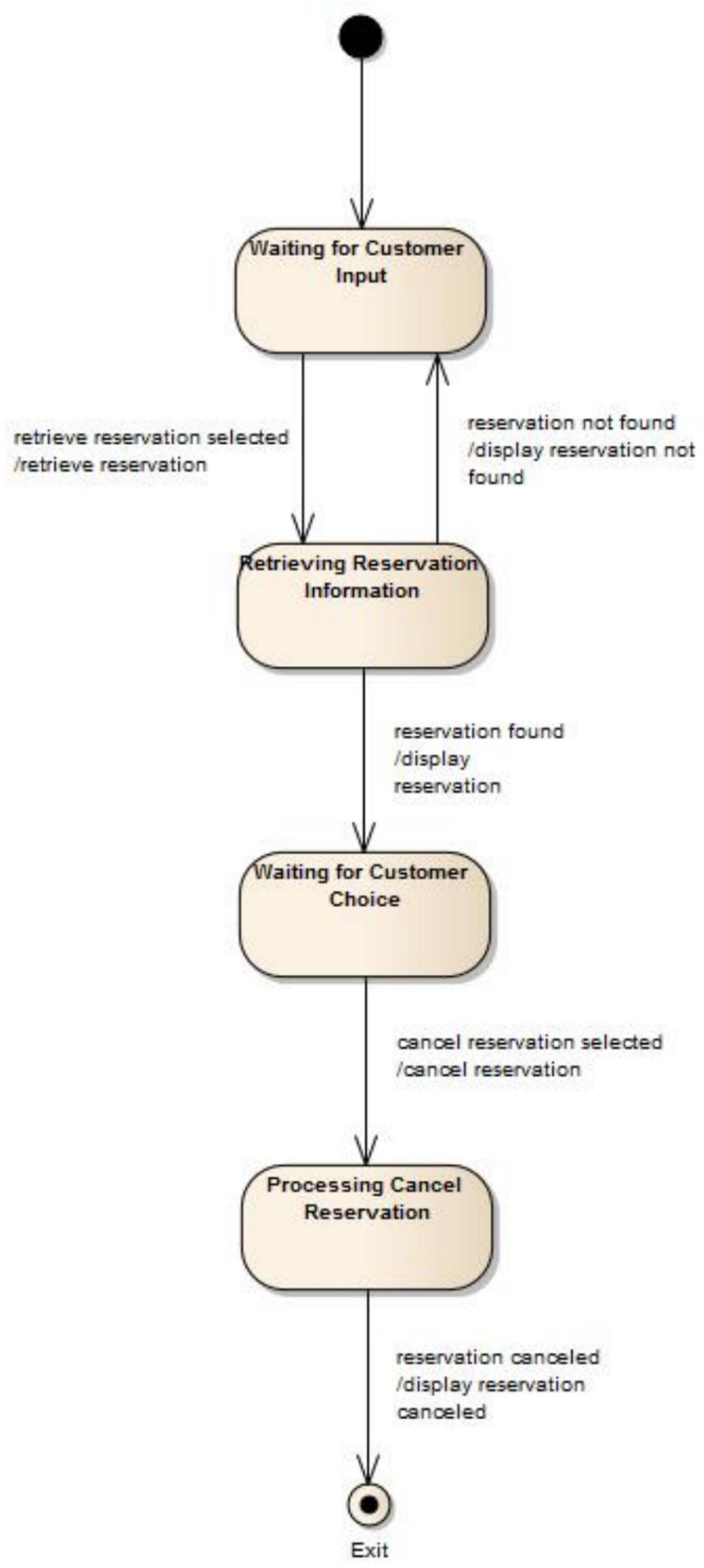


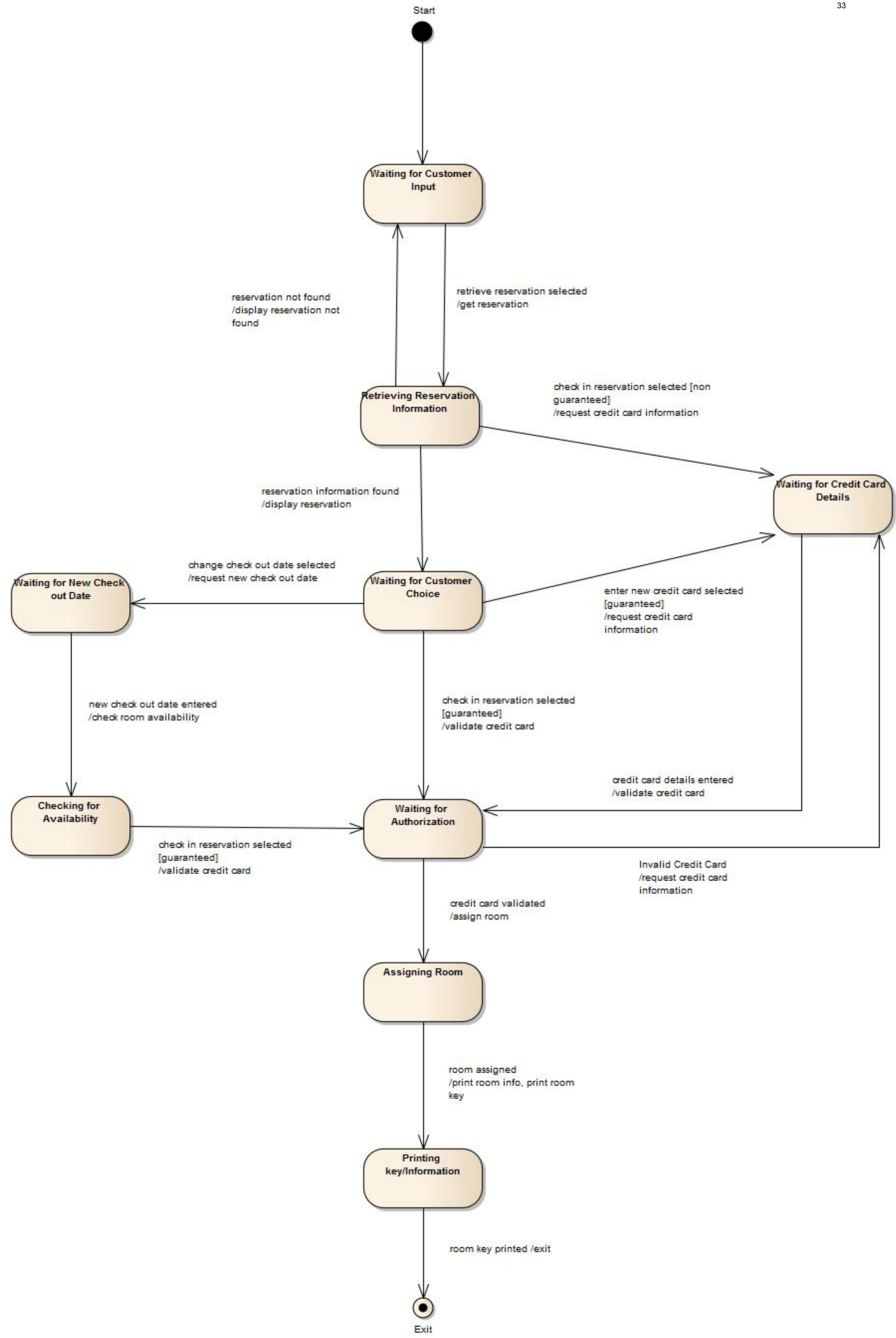


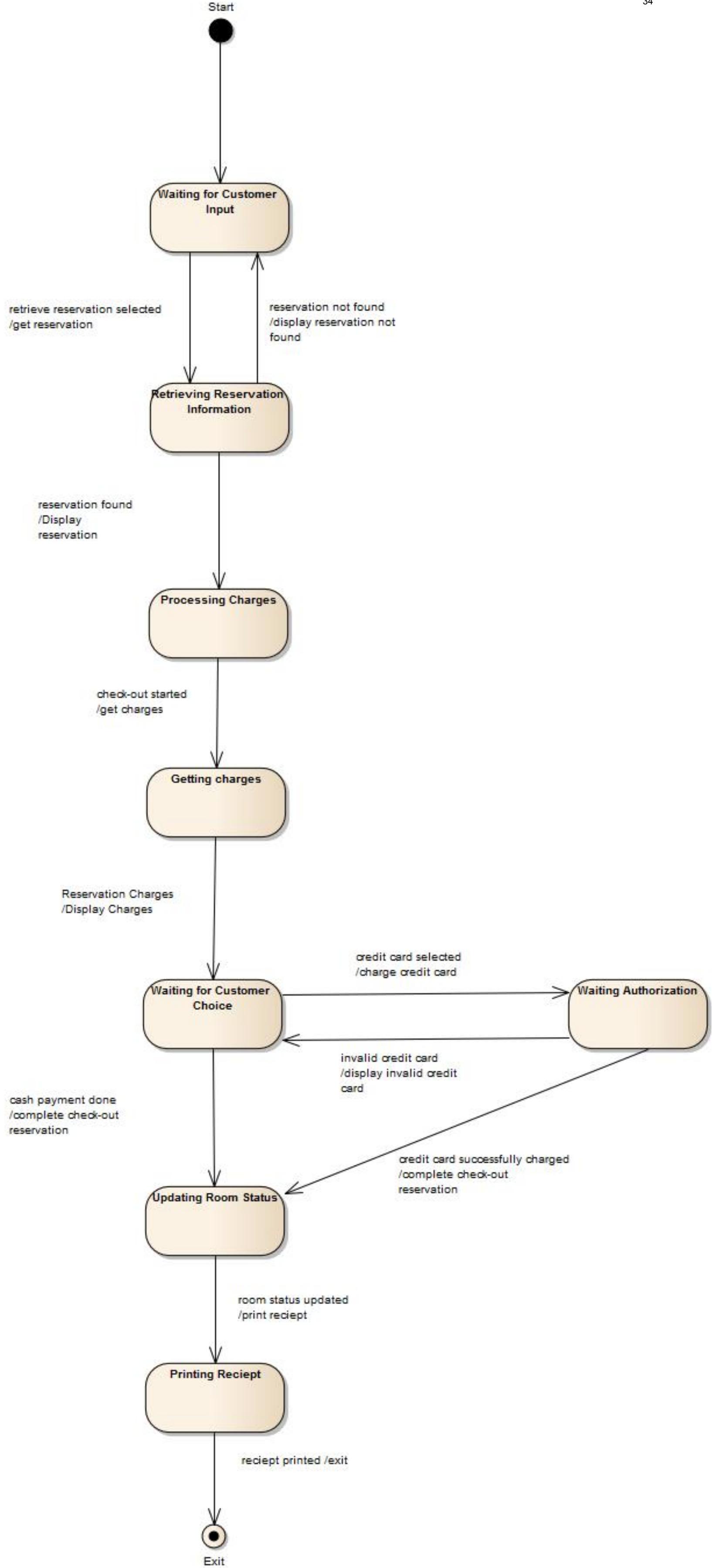


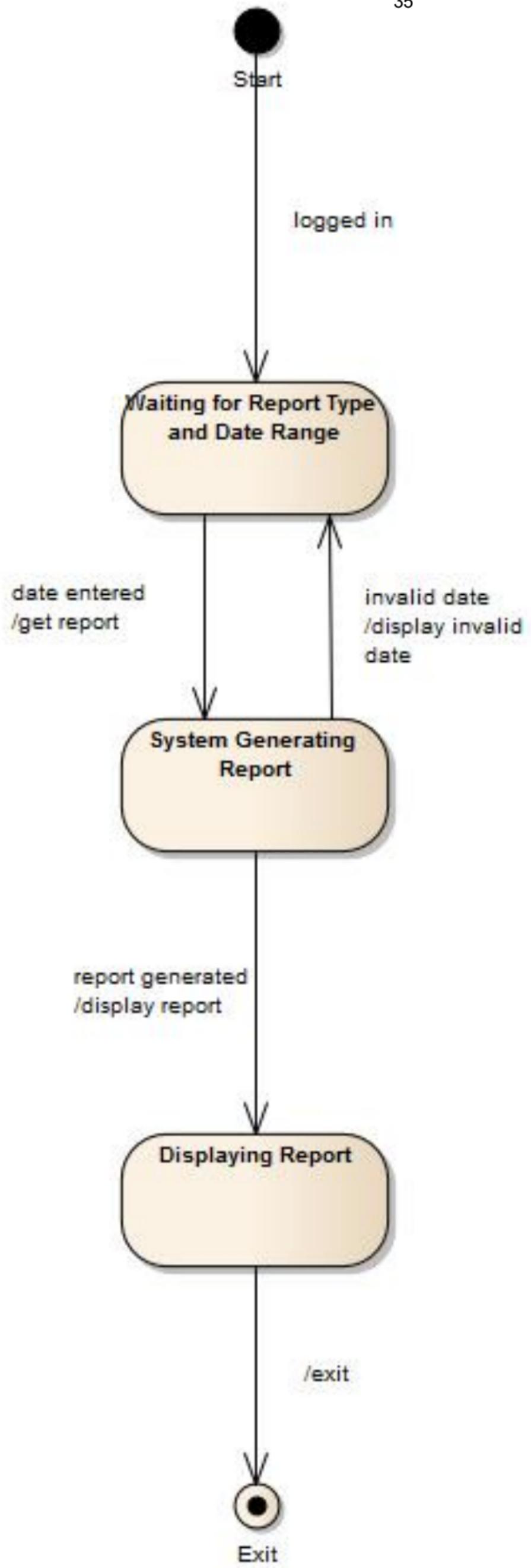












- 1:** Customer requests to make a reservation and enters the reservation details with include Hotel name, number of rooms, check-in date, check-out date, room type.
- 1.1:** The Customer Interaction object sends message reservation details entered to the Reservation Transaction entity object.
- 1.2:** The Reservation Transaction entity object responds to the Customer Interaction object that transaction has been created
- 1.3:** The Customer Interaction object sends a message Make Reservation selected with Check availability transaction to the Client control to check for Availability of Rooms, on selected check-in date, check-out date, room type and number of rooms.
- 1.4:** The client control sends a request to the Hotel service to check for availability of rooms.
- 1.5:** If rooms available, the Hotel service responds to the client control with a message confirming rooms are available and the total cost
- 1.6:** The client control then sends a message to the customer interaction object rooms are available, total cost and to get customer details
- 1.7:** The customer interaction object displays the message to the customer, and prompts the customer to enter the **customer details**
- 1.5A:** If rooms are not available, the Hotel service responds to the client control with a message rooms are not available
- 1.5A.1:** The client control then sends a message to the customer interaction object rooms are not available, and get new dates
- 1.5A.2:** The customer interaction object displays the message to the customer, and prompts the customer to enter new dates
- 1.5A.3:** Customer enters new check-in and check-out dates and sends the changed reservation details to Customer interaction object.
- 1.5A.4:** The Customer Interaction object sends a make reservation request to the Reservation Transaction object with the new date(s)
- 1.5A.5:** The Reservation Transaction object responds to the customer interaction object that transaction has been created
- 1.5A.6:** The Customer Interaction object sends a request to the client control to check for availability of rooms
- 1.5A.7:** On selected range of new dates, the client control sends a request to the Hotel service to check for availability of rooms
- 2:** Customer enters the require customer details like, first name, last name, company name(optional) and telephone number.
- 2.1:** Customer details entered messaged is sent to Reservation Transaction object with reservation details.
- 2.2:** Make reservation transaction is created and sent to the User interaction object.
- 2.3:** Customer details entered messaged is sent to Client control object with reservation details.
- 2.4:** If the number of rooms entered is less than 10, customer is prompted to choose reservation type by entering credit card details for guaranteed reservation.
- 2.4A:** If the number of rooms entered is greater than 10, customer is prompted to guarantee the reservation by providing credit card details
- 2.4A.1:** User interaction requests from Credit card details to guarantee the reservation.
- 2.4A.2:** Customer enters credit card details (credit card #, exp date, type)
- 2.4B.3:** The customer interaction object sends credit card details to the client control requesting to make the reservations guaranteed credit card details
-

2.5: customer chooses reservation type by entering the reservation type by entering credit card details for guaranteed reservation.

3: The customer enters credit card details is reservation has to be guaranteed else leaves empty.

3.1: Customer interaction forwards the details to client control.

3.2: If reservation type is guaranteed (i.e. if Credit card details are entered), the client control sends a request to the authorization proxy object to validate the credit card

3.3: The authorization proxy object sends a request to the authorization service to validate the credit card

3.4: The authorization service responds to the authorization proxy that credit card is validated

3.4A: The authorization service responds to the authorization proxy that credit card is invalid

3.4A.1: The proxy object responds to the client control object credit card is invalid.

3.4A.2: The client control sends credit denied and requests for new credit card display message.

3.4A.3: User interaction object display the message.

3.4A.4: Customer enters new credit card details

3.4A.5: New credit card details are forwarded to Client control.

3.4A.6: the client control sends a request to the authorization proxy object to validate the credit card

3.4A.7: The authorization proxy object sends a request to the authorization service to validate the credit card

3.4A.8: The authorization service responds to the authorization proxy that credit card is validated

3.4A.9: The proxy object responds to the client control object credit card is validated

3.5: The proxy object responds to the client control object credit card is validated

3.6: The client control object requests the hotel service object to make a reservation

3.7: The hotel service object makes the reservation and responds to the client control object with a confirmation number

3.8: The client control object sends the confirmation number to the customer interaction object

3.9: The client interaction object displays the confirmation number to the customer

- 1:** The client system object sends a request to the hotel coordinator object to check for room availability
- 1.1:** The hotel coordinator object sends a request to the business logic to check for room availability
- 1.2:** The business logic checks the calendar for room availability for the given dates in Availability transaction.
- 1.3:** If rooms are available, the calendar responds to the logic with the number of rooms that are available
- 1.4:** The logic requests the room cost from hotel entity object to compute for charges (cost)
- 1.5:** The entity object responds to the logic with the room cost
- 1.6:** The logic calculates responds to the client system that rooms are available and the total cost for rooms.
- 1.3A:** If rooms are not available, the calendar responds to the logic that rooms are not available
- 1.3A.1:** Availability logic object sends room not available message to Coordinator.
- 1.3A.2:** Hotel coordinator forwards the message to the client subsystem.
- 2:** Client subsystem sends Make reservation request with Make Transaction object to Hotel coordinator.
- 2.1:** Hotel Coordinator forwards the message to make reservation logic.
- 2.2:** The logic object sends a request to the room calendar object to check availability again.
- 2.3:** The room object responds to the logic object that rooms are available.
- 2.4:** Logic requests reservation entity to create a new reservation
- 2.5:** The Reservation entity returns the confirmation number of the new reservation created.
- 2.6:** The logic sends a request to the calendar object to update the number of reservations for that date and room availability.
- 2.7:** The logic responds to the hotel coordinator object with the confirmation number
- 2.8:** The hotel coordinator object forwards to the client system with the confirmation number, and reservation is done

1: Customer sends a request to the customer interaction object to change an existing reservation, enters confirmation number

1.1: The customer interaction object sends the confirmation number to the reservation transaction object to generate a transaction associated with the confirmation number

1.2: The reservation transaction object responds to the customer interaction object with the Retrieve transaction created.

1.3: The customer interaction object sends a request to the client control with message retrieve the reservation selected with Retrieve Reservation transaction.

1.4: The client sends a request to the hotel service to retrieve the reservation

1.5: If Reservation Found, The hotel service object sends the retrieved reservation information to the client control

1.6: The client control sends Display Reservation information message to user interaction.

1.7: customer interaction formats and displays the Reservation information to the Customer.

1.5A: If Reservation Not Found, The hotel service object sends a response to the client control object with reservation is not found message.

1.5A.1: The client control sends a response to the customer interaction object that reservation is not found.

1.5A.2: The customer interaction object displays reservation not found, and prompt the customer to enter the confirmation number again

2: The customer changes the reservation.

2.1: The customer interaction object sends a request to the reservation transaction object to of change Transaction.

2.2: The reservation transaction object responds with change transaction object to customer interaction.

2.3: The customer transaction object sends a request to the client control object with Reservation changed message to client control.

2.4: If Change transaction had GUARANTEED reservation the client control sends a request to the authorization proxy to validate the credit card

2.5: The authorization proxy sends a request to the authorization service to validate the credit card

2.6: The authorization service responds to the proxy with validated credit card.

2.6A: The authorization service responds to the authorization proxy that credit card is invalid.

2.6A.1: The proxy object responds to the client control object credit card is invalid.

2.6A.2: The client control sends credit denied and requests for new credit card display message.

2.6A.3: User interaction object display the message.

2.6A.4: Customer enters new credit card details

2.6A.5: New credit card details are forwarded to Client control.

2.6A.6: the client control sends a request to the authorization proxy object to validate the credit card

2.6A.7: The authorization proxy object sends a request to the authorization service to validate the credit card

2.6A.8: The authorization service responds to the authorization proxy that credit card is validated

2.6A.9: The proxy object responds to the client control object credit card is validated

2.7: The proxy responds to the client control with validated credit card.

2.8: The client control object sends a request to the hotel service object to change the reservation with change transaction.

2.9: if ROOMS are AVAILABLE, The hotel service object responds to the client control object with confirmation number and cost

2.10: The client control object to the customer interaction object to display confirmation # and cost

2.11: The customer interaction object displays the confirmation # and cost to the customer

2.9A: if NO ROOMS AVAILABLE, the hotel service object responds to the client control object with no rooms available message

2.9A.1: The client control object responds to the customer interaction object to display no rooms available

2.9A.2: The customer interaction object displays no rooms available to the customer

3: Customer accepts the changes

3.1: The customer interaction object sends a request to the client control object that change is accepted

3.2: The client control object sends a request to the hotel service to update the reservation

3A: if CHANGE NOT ACCEPTED, Customer selects no change accepted

-

- 1:** The client system object sends a request to the hotel coordinator object to retrieve reservation info with the confirmation number
- 1.1:** The hotel coordinator object sends a request to the retrieve reservation business logic object to retrieve reservation
- 1.2:** The retrieve reservation business logic object sends a request to the reservation object to get the reservation details.
- 1.3:** Reservation entity returns the Reservation details corresponding to the confirmation number.
- 1.3A:** If Reservation not found, the reservation object responds to the retrieve reservation business logic object with reservation not found message.
- 1.3A.1:** The retrieve reservation business logic object sends reservation not found to the hotel coordinator
- 1.3A.2:** The hotel coordinator object responds to the client system with the reservation not found
- 1.4:** Retrieve Reservation Logic request the customer details for the corresponding reservation number
- 1.5:** Customer entity returns the Customer details to the Retrieve Reservation object.
- 1.6:** Retrieve reservation returns the Reservation information to the hotel coordinator.
- 1.7:** Hotel coordinator forwards the reservation information to the client subsystem.
- 2:** Client subsystem sends a change reservation request with change transaction to the hotel coordinator.
- 2.1:** Hotel Coordinator forwards the change reservation request to Change Reservation business logic
- 2.2:** Change Reservation logic request the Room Calendar entity for number of Room available for the new selected check in and check out dates, number of room and room type.
- 2.3:** Room calendar entity returns the number of rooms available.
- 2.4:** Changes reservation also request for the Customer information from the customer entity.
- 2.5:** Customer entity returns the Customer information corresponding to the current reservation.
- 2.6:** Change reservation Logic simulates the changes and generates a cost for the new reservation and send it to hotel coordinator.
- 2.7:** Hotel coordinator forwards the message to client subsystem with confirmation of change availability and cost.
- 3:** Update Reservation change request is forward from Client system to hotel coordinator
- 3.1:** Hotel coordinator forwards the Message to Change Reservation business logic
- 3.2:** Change Reservation business logic updates the room availability in the room calendar entity.
- 3.3:** Room Calendar acknowledges about the successful update
- 3.4:** Change Reservation business logic the current reservation with the new changed details.
- 3.5:** Reservation entity acknowledges about the successful update.
- 3.6:** Change Reservation business logic updates the credit card details if it were changed in customer entity
- 3.7:** Customer entity acknowledges about the successful update.

- 1:** Customer sends a request to the customer interaction object to change an existing reservation, enters confirmation number
- 1.1:** The customer interaction object sends the confirmation number to the reservation transaction object to generate a transaction associated with the confirmation number
- 1.2:** The reservation transaction object responds to the customer interaction object with the Retrieve transaction created.
- 1.3:** The customer interaction object sends a request to the client control with message retrieve the reservation selected with Retrieve Reservation transaction.
- 1.4:** The client sends a request to the hotel service to retrieve the reservation
- 1.5:** If Reservation Found, The hotel service object sends the retrieved reservation information to the client control
- 1.6:** The client control sends Display Reservation information message to user interaction.
- 1.7:** customer interaction formats and displays the Reservation information to the Customer.
- 1.5A:** If Reservation Not Found, The hotel service object sends a response to the client control object with reservation is not found message.
- 1.5A.1:** The client control sends a response to the customer interaction object that reservation is not found.
- 1.5A.2:** The customer interaction object displays reservation not found, and prompt the customer to enter the confirmation number again
- 2:** Customer selects to cancel the reservation
- 2.1:** The customer interaction object sends a request to the reservation transaction object to get a cancel reservation transaction
- 2.2:** The reservation transaction object responds to the customer interaction object with the cancel reservation transaction
- 2.3:** The customer interaction object sends a request to the client control object to cancel the reservation
- 2.4:** The client control object sends a request to the hotel service object to cancel the reservation
- 2.5:** The hotel service object cancels the reservation and responds to the client control object that reservation is canceled
- 2.6:** The client control object sends a response to the customer interaction object reservation is canceled, and display information
- 2.7:** The customer interaction object displays reservation canceled to the customer

1: The client system object sends a request to the hotel coordinator object to retrieve reservation info with the confirmation number

1.1: The hotel coordinator object sends a request to the retrieve reservation business logic object to retrieve reservation

1.2: The retrieve reservation business logic object sends a request to the reservation object to get the reservation details.

1.3: Reservation entity returns the Reservation details corresponding to the confirmation number.

1.3A: If Reservation not found, the reservation object responds to the retrieve reservation business logic object with reservation not found message.

1.3A.1: The retrieve reservation business logic object sends reservation not found to the hotel coordinator

1.3A.2: The hotel coordinator object responds to the client system with the reservation not found

2: The client system sends a request to the hotel coordinator object to cancel the reservation

2.1: The hotel coordinator object sends a request to the cancel reservation business logic object to cancel the reservation

1: The client system object sends a request to the hotel coordinator object to retrieve reservation info with the confirmation number

1.1: The hotel coordinator object sends a request to the retrieve reservation business logic object to retrieve reservation

1.2: The retrieve reservation business logic object sends a request to the reservation object to get the reservation details.

1.3: Reservation entity returns the Reservation details corresponding to the confirmation number.

1.3A: If Reservation not found, the reservation object responds to the retrieve reservation business logic object with reservation not found message.

1.3A.1: The retrieve reservation business logic object sends reservation not found to the hotel coordinator

1.3A.2: The hotel coordinator object responds to the client system with the reservation not found

The cancel reservation business logic object cancels the reservation and sends a request to the reservation object to update reservation accordingly

2.3: The reservation object responds to the cancel reservation business logic object reservation is updated

2.4: The cancel reservation business logic object sends a request to the calendar object to update calendar accordingly

2.5: The calendar responds to the cancel reservation business logic object calendar is updated

2.6: The cancel reservation business logic object responds to the hotel coordinator object reservation is canceled

2.7: The hotel coordinator object responds to the client system reservation is canceled

- 1:** Customer sends a request to the customer interaction object to change an existing reservation, enters confirmation number
- 1.1:** The customer interaction object sends the confirmation number to the reservation transaction object to generate a transaction associated with the confirmation number
- 1.2:** The reservation transaction object responds to the customer interaction object with the Retrieve transaction created.
- 1.3:** The customer interaction object sends a request to the client control with message retrieve the reservation selected with Retrieve Reservation transaction.
- 1.4:** The client sends a request to the hotel service to retrieve the reservation
- 1.5:** If Reservation Found, The hotel service object sends the retrieved reservation information to the client control
- 1.6:** The client control sends Display Reservation information message to user interaction.
- 1.7:** customer interaction formats and displays the Reservation information to the Customer.
- 1.5A:** If Reservation Not Found, The hotel service object sends a response to the client control object with reservation is not found message.
- 1.5A.1:** The client control sends a response to the customer interaction object that reservation is not found.
- 1.5A.2:** The customer interaction object displays reservation not found, and prompt the customer to enter the confirmation number again
- 1.8: The client control issues a request to the customer interaction object to ask for user's choice
- 1.9: The customer interaction displays the message to the clerk
- 1.10: The clerk chooses to check in the person with the particular reservation information
- 1.10A: The clerk chooses to enter a different credit card for the Customer to check-in.
- 1.10A.1: The User Interaction informs the Client Control of the Clerk's choice.
- 1.10A.2: The Client Control issues a request to the Customer Interaction to get the details of the new credit card.
- 1.10A.3: The Client Control forwards this request to the Clerk.
- 1.10A.4: The Clerk enters the credit card number and the expiration date for the new card.
- 1.10B: The clerk chooses to enter a different check-out date for the Customer to check-in.
- 1.10B.1: The User Interaction informs the Client Control of the Clerk's choice.

1.10B.2: The Client Control issues a request to the Customer Interaction to get the details of the new check-out date.

1.10B.3: The Client Control forwards this request to the Clerk.

1.10B.4: The Clerk enters the new check out date.

1.10B.5: The customer interaction object passes the new check out date to the Client Control.

1.10B.6: The Client control sends a request to the Hotel Service to change the check out date for the particular reservation.

1.10B.7: RESERVATION CHANGED, Hotel Service sends a confirmation message back to the Client Control saying that the reservation has been changed and the check out date has been changed as requested.

1.10B.8: The Client Control forwards this message to the Customer Interaction.

1.10B.9: The Customer Interaction displays the message to the Clerk.

1.10B.7A: RESERVATION UNCHANGED; Hotel Service sends a message back to the Client Control that the reservation was not changed because of unavailability of dates.

1.10B.7A.1: The Client Control forwards this message to the Customer Interaction.

1.10B.7A.2: The Customer Interaction displays the message to the Clerk.

1.11: The customer interaction object requests credit card information associated with the confirmation number to the Reservation transaction entity.

1.12: The Reservation transaction entity sends the credit card information back to the customer interaction.

1.13: The customer interaction requests the Client Control to start the check in process for the customer

1.14: GUARANTEED RESERVATION, Client Control sends the information to the Authorization Proxy to get the information validated.

1.15: Authorization Proxy sends the credit card number and expiration date to the Authorization service.

1.16: The Authorization service authenticates the credit card information and sends a response to the Authorization Proxy.

1.17: The Authorization Proxy forwards the message to the Client Control.

1.18: NON GUARANTEED RESERVATION, The Client Control sends a request to the Customer Interaction to get information on the customer's credit card.

1.19: The Customer Interaction displays the request to the Clerk.

1.20: The Clerk enters the credit card number and expiration date.

1.21: The Customer Interaction sends this information to the Client Control.

1.14A: Client Control sends the information to the Authorization Proxy to get the information validated.

1.14A.1: Authorization Proxy sends the credit card number and expiration date to the Authorization service.

1.14A.2: The Authorization service authenticates the credit card information and sends a response to the Authorization Proxy.

1.14A.3: The Authorization Proxy forwards the message to the Client Control.

1.14A.B: INVALID CREDIT CARD

1.14A.B.1: The Authorization service informs that the credit card information is invalid to the Authorization Proxy.

1.21: The Authorization Proxy forwards the message to the Client Control.

1.21: Client Control issues a card denied message to the Customer Interaction.

1.22: Customer Interaction displays the card denied message to the Clerk.

1.23: Once the credit card is validated, the Client Control takes the reservation information and issues a message to the Hotel Service subsystem to assign a room to the customer with the mentioned reservation details.

1.24: The Hotel Service sends back a success message after it has assigned a room to the reservation.

1.25: The Client Control sends print command to the Printer interface.

1.26: Printer interface requests the reservation information

1.27: Reservation transaction sends the reservation info.

1.28: The Client Control sends print command to the Key Printer interface.

1.29: Key Printer interface requests the reservation information

1.30: The Client Control prints out the key card for the customer over the Key Printer.

1.10A: The clerk chooses to enter a different credit card for the Customer to check-in.

1.10A.1: The User Interaction informs the Client Control of the Clerk's choice.

1.10A.2: The Client Control issues a request to the Customer Interaction to get the details of the new credit card.

1.10A.3: The Client Control forwards this request to the Clerk.

1.10A.4: The Clerk enters the credit card number and the expiration date for the new card.

1.10B: The clerk chooses to enter a different check-out date for the Customer to check-in.

1.10B.1: The User Interaction informs the Client Control of the Clerk's choice.

1.10B.2: The Client Control issues a request to the Customer Interaction to get the details of the new check-out date.

1.10B.3: The Client Control forwards this request to the Clerk.

1.10B.4: The Clerk enters the new check out date.

1.10B.5: The customer interaction object passes the new check out date to the Client Control.

1.10B.6: The Client control sends a request to the Hotel Service to change the check out date for the particular reservation.

1.10B.7: RESERVATION CHANGED, Hotel Service sends a confirmation message back to the Client Control saying that the reservation has been changed and the check out date has been changed as requested.

1.10B.8: The Client Control forwards this message to the Customer Interaction.

1.10B.9: The Customer Interaction displays the message to the Clerk.

1.10B.7A: RESERVATION UNCHANGED; Hotel Service sends a message back to the Client Control that the reservation was not changed because of unavailability of dates.

1.10B.7A.1: The Client Control forwards this message to the Customer Interaction.

1.10B.7A.2: The Customer Interaction displays the message to the Clerk.

- 1:** The client system object sends a request to the hotel coordinator object to retrieve reservation info with the confirmation number
- 1.1:** The hotel coordinator object sends a request to the retrieve reservation business logic object to retrieve reservation
- 1.2:** The retrieve reservation business logic object sends a request to the reservation object to get the reservation details.
- 1.3:** Reservation entity returns the Reservation details corresponding to the confirmation number.
- 1.3A:** If Reservation not found, the reservation object responds to the retrieve reservation business logic object with reservation not found message.
- 1.3A.1:** The retrieve reservation business logic object sends reservation not found to the hotel coordinator
- 1.3A.2:** The hotel coordinator object responds to the client system with the reservation not found

Assign Room

- 2: Client System makes a request to the Hotel Coordinator to assign a room with a check-in transaction.
- 2.1: Hotel Coordinator forwards this request to the Check in Reservation business logic.
- 2.2: The Check in Reservation business logic requests the Calendar entity for current rooms available.
- 2.3: The Calendar Entity sends back the rooms available.
- 2.4: The Check in Reservation business logic requests the Reservation Entity for the reservation information for the reservation confirmation number.
- 2.5: The Reservation Entity sends back the reservation information.
- 2.6: The Check in Reservation business logic requests the Room Entity for the details of the room it has selected for the customer.
- 2.7: The Room entity sends back the room information.
- 2.8: The Check in Reservation business logic then assigns room(s) to the reservation.
- 2.9: The Check in Reservation business logic requests the Calendar Entity to update the availability of the rooms.
- 2.10: The Check in Reservation business logic requests the Reservation entity to update its information.
- 2.11: The Check in Reservation business logic informs the Hotel Coordinator that room(s) has successfully been assigned to the reservation.
- 2.12: The Hotel Coordinator informs the Client System.

3: Check Availability, The Client System requests the Hotel Coordinator to find if a room is available on the mentioned check-out date.

3.1: The Hotel Coordinator sends this request to the Check Availability Business Logic.

3.2: The Check Availability Business Logic sends a request to the Calendar Entity get rooms available on the check-out date.

3.3: The Calendar entity sends the rooms available to the Check Availability Business Logic.

3.4: Check Availability Business Logic uses the information to find the room available for the mentioned check-out date.

3.5: ROOM AVAILABLE, Check Availability Business Logic informs the Hotel Coordinator that the new check-out date for the room is available.

3.6: The Hotel Coordinator sends the room available message to the Client System.

3.5A: ROOM UNAVAILABLE, Check Availability Business Logic informs the Hotel Coordinator that the new check-out date for the room is unavailable.

3.5A.1: The Hotel Coordinator sends the room unavailable message to the Client System.

1: Customer sends a request to the customer interaction object to change an existing reservation, enters confirmation number

1.1: The customer interaction object sends the confirmation number to the reservation transaction object to generate a transaction associated with the confirmation number

1.2: The reservation transaction object responds to the customer interaction object with the Retrieve transaction created.

1.3: The customer interaction object sends a request to the client control with message retrieve the reservation selected with Retrieve Reservation transaction.

1.4: The client sends a request to the hotel service to retrieve the reservation

1.5: If Reservation Found, The hotel service object sends the retrieved reservation information to the client control

1.6: The client control sends Display Reservation information message to user interaction.

1.7: customer interaction formats and displays the Reservation information to the Customer.

1.5A: If Reservation Not Found, The hotel service object sends a response to the client control object with reservation is not found message.

1.5A.1: The client control sends a response to the customer interaction object that reservation is not found.

1.5A.2: The customer interaction object displays reservation not found, and prompt the customer to enter the confirmation number again

2: The Clerk enters the reservation confirmation number and the room number which is send to the Customer Interaction object.

2.1: The Customer Interaction object forwards this request to the Reservation Transaction to make Check-out Transaction and send it to the Server.

2.2: The Reservation Transaction makes a Check-out Transaction and sends it back to the Customer Interaction.

2.3: The Customer Interaction sends the Check-out Transaction to the Client Control to begin the check out process.

2.4: The Client Control sends a request to the Hotel Service to get the charges for the Check-out transaction.

2.5: The Hotel Service sends back the charges to the Client Control.

2.6: The Client Control forwards these charges to the Customer Interaction.

2.7: The Customer Interaction displays the charges to the Clerk.

3: The Clerk requests the system to charge the customer's credit card.

3.1: The Customer Interaction forwards this request to the Client Control with the Check-out Transaction information.

3.2: The Client Control sends the charges with the Check-out Transaction to the Hotel Service.

3.3: VALID CREDIT CARD, The Hotel Service sends a credit card successfully charged message to the Client Control.

3.3A: INVALID CREDIT CARD, The Hotel Service sends a credit card denied message to the Client Control.

3.3A.1: The Client Control forwards the credit card denied message to the Customer Interaction.

3.3A.2: The Customer Interaction displays the credit card denied message for the Clerk.

3.4: The Client Control sends a request to the Hotel Service to complete the check-out for the reservation.

3.5: The Hotel Service sends a success message back to the Client Control with the information about the Bill for the customer.

3.6: The Client Control prints the receipt with the using the Bill information for the customer.

3A: The Clerk enters the cash amount paid by the customer using the Customer Interaction.

3A.1: The Customer Interaction forwards the information that cash payment has been done to the Client Control.

3A.2: The Client Control sends a request to the Hotel Service to complete the check-out for the reservation.

3A.3: The Hotel Service sends a success message back to the Client Control with the information about the Bill for the customer.

3A.4: The Client Control prints the receipt with the using the Bill information for the customer.

- 1:** Customer sends a request to the customer interaction object to change an existing reservation, enters confirmation number
- 1.1:** The customer interaction object sends the confirmation number to the reservation transaction object to generate a transaction associated with the confirmation number
- 1.2:** The reservation transaction object responds to the customer interaction object with the Retrieve transaction created.
- 1.3:** The customer interaction object sends a request to the client control with message retrieve the reservation selected with Retrieve Reservation transaction.
- 1.4:** The client sends a request to the hotel service to retrieve the reservation
- 1.5:** If Reservation Found, The hotel service object sends the retrieved reservation information to the client control
- 1.6:** The client control sends Display Reservation information message to user interaction.
- 1.7:** customer interaction formats and displays the Reservation information to the Customer.
- 1.3A:** If Reservation Not Found, The hotel service object sends a response to the client control object with reservation is not found message.
- 1.3A.1:** The client control sends a response to the customer interaction object that reservation is not found.
- 1.3A.2:** The customer interaction object displays reservation not found, and prompt the customer to enter the confirmation number again
- 2: GET CHARGES, CHECK-OUT TIME < 12pm,** Client System requests the Hotel Coordinator to get the charges for a Check-out Transaction.
- 2.1: The Hotel Coordinator forwards check-out request to Check-out Reservation business logic.
- 2.2: The Check-out Reservation business logic requests the Reservation Entity for information for the confirmation number.
- 2.3: The Reservation Entity sends back the information to the Check-out Reservation business logic as requested.
- 2.4: The Check-out Reservation business logic computes the charges for the stay and sends the information to the Hotel Coordinator.
- 2.5: The Hotel Coordinator forwards the charges to the Client System.
- 2.4A: CHECK-OUT TIME >12pm,** The Check-out Reservation business logic computes the charges for the stay and adds a charge for one more day and then sends the information to the Hotel Coordinator.
- 2.4A.1: The Hotel Coordinator forwards the charges to the Client System.
- 3: CHARGE CREDIT CARD,** The Client System sends a request to the Hotel Coordinator to charge a customer credit card for a Check-out Transaction.

- 3.1: The Hotel Coordinator forwards the request to charge credit card to the Check-out Reservation business logic.
 - 3.2: The Check-out Reservation business logic requests the Reservation Entity for information for a particular confirmation number.
 - 3.3: The Reservation entity sends back the information to the Check-out Reservation business logic.
 - 3.4: The Check-out Reservation business logic requests the Customer Entity to get the information about the credit card.
 - 3.5: The Customer entity sends back the credit card information to the Check-out Reservation business logic.
 - 3.6: The Check-out Reservation business logic requests the Authorization Proxy to charge the customer credit card.
 - 3.7: The Authorization Proxy sends the credit card charge request to the Authorization Service.
 - 3.8: The Authorization Service sends back a payment successful message to the Authorization Proxy.
 - 3.9: The Authorization Proxy forwards the successful payment message to the Check-out Reservation business logic.
 - 3.10: The Check-out Reservation business logic sends the successful payment message to the Hotel Coordinator.
 - 3.11: The Hotel coordinator forwards the card successfully charges message to the Client System.
- 3.7A: INVALID CREDIT CARD,** the Authorization Service sends back a payment not successful message to the Authorization Proxy.
- 3.7A.1: The Authorization Proxy forwards the failed payment message to the Check-out Reservation business logic.
 - 3.7A.2: The Check-out Reservation business logic sends the failed payment message to the Hotel Coordinator.
 - 3.7A.3: The Hotel coordinator forwards the card invalid message to the Client System.

4: COMPLETE CHARGES, The Client System sends a request to the Hotel Coordinator to complete the Check-out with a Check-out Transaction.

- 4.1: The Hotel coordinator forwards the check-out request to the Check-out Reservation business logic.

- 4.2: The Check-out Reservation business logic requests the Reservation Entity to update the reservation information.
- 4.3: The Reservation Entity sends a reservation status updated message to the Check-out Reservation business logic.
- 4.4: The Check-out Reservation business logic requests the Calendar Entity to update the room availability.
- 4.5: The Calendar entity sends a room availability updated message to the Check-out Reservation business logic.
- 4.6: The Check-out Reservation business logic requests the Room Entity to update the room status.
- 4.7: The Room Entity sends a room status updated message back to the Check-out Reservation business logic.
- 4.8: The Check-out Reservation business logic requests the Bill Entity to generate a bill using the information gathered.
- 4.9: The Bill Entity sends back the bill generated to the Check-out Reservation business logic.
- 4.10: The Check-out Reservation business logic sends the generated bill to the Hotel coordinator.
- 4.11: The Hotel Coordinator forwards the bill generated to the Client system.

1: Manager requests the User Interaction object to generate a report of a particular type for a mentioned time frame.

1.1: The User Interaction object sends the request to generate a report to the Client Control.

1.2: The Client Control forwards the request to generate report to the Hotel Service.

1.3: The Hotel Service returns the report generated to the Client Control.

1.4: The Client Control forwards the generated report to the User Interaction object.

1.5: The User interaction object displays the generated report to the Manager.

1.3A: INVALID DATES, Hotel service sends a message to the Client Control that it was unable to generate a report for the requested date.

1.3A.1: The Client Control forwards the invalid date message to the User Interaction.

1.3A.2: The User Interaction displays the invalid date message to the Manager.

1: CURRENT REPORT, The Client System requests the Hotel Coordinator to generate a report between two given dates.

1.1: The Hotel Coordinator forwards the request to generate report to the Management Report business logic.

1.2: The Management Report business logic requests the Reservation Entity for reservations between the from and to dates.

1.3: The Reservation Entity sends back all reservations between the from and to date.

1.4: The Management Report business logic requests the Bill Entity for all bills generated between the from and to dates.

1.5: The Bill Entity sends back all the bills generated between the from and to dates.

1.2A: PROJECTED REPORT, The Management Report business logic requests the Projection algorithm to generate a report for time period between the from and to date.

1.2A.1: The Projection Algorithm requests the Bill Entity to send all the bills generated for the Hotel.

1.2A.2: The Bill entity sends the bills generated to the Projection Algorithm.

1.2A.3: The Projection Algorithm requests the Reservation Entity to get all the reservations made for the Hotel.

1.2A.4: The Reservation Entity sends back the reservations made to the Projection Algorithm.

1.2A.5 Projection Algorithm sends its projections to the Management Report business logic for it to generate the reports.

1.6: The Management Report business logic stores the generated reports with the Report Entity.

1.7: The Management Report business logic sends the report to the Hotel Coordinator.

1.8: The Hotel Coordinator forwards the generated report to the Client System.

- 1: Timer signals 6:00 pm to the cancel no show business logic to cancel non-guaranteed no show reservations
 - 1.1: The business logic sends a request to the reservation object to cancel the reservation (check_in_date, date, nonguaranteed)
 - 1.2: The reservation object responds to the logic that reservation is canceled (# of reservations changed, customer info)
 - 1.3: The business logic sends a request to the log object to log the changes (# of reservations, customer info)
 - 1.4: The business logic sends a request to the calendar object to update the room calendar (# of reservations)
 - 1.5: The Room Calendar acknowledges the change done.
 - 1.6: The business logic sends a request to the reservation object to mark 'Must pay' those who didn't check-in on time (check-in-date, date, guaranteed)
 - 1.7: The reservation object responds to the logic marked must pay
 - 1.8: The logic sends a request to the log object to log the transaction (customer info, marked must pay)

- 1:** Timer signals 7:00 am to bill-guaranteed no show business logic
- 1.2:** The logic sends a request to the reservations object to get those customers marked 'Must pay'
- 1.3:** The reservations object responds to the logic with reservations marked 'Must pay' (customer info, confirmation #)
- 1.4:** The logic sends a request to the Authorization system object to charge reservations marked 'Must pay' (credit card info, amount)
- 1.5:** The Authorization system object responds to the logic reservations charged (credit card info, amount charged)
- 1.6:** The logic object sends a request to the log object to log the changes (customer info, guaranteed, reservation canceled, amount charged)
- 1.7:** The logic object sends a request to the reservation object to cancel reservations marked 'Must pay' and charged (customer info, confirmation #)
- 1.8:** The reservation object cancels the reservations and responds to the logic object reservations canceled (customer info)
- 1.9:** The logic object sends a request to the calendar object to update the calendar (# of canceled reservations, date)
- 1.10:** The calendar object responds to the logic object reservations updated (date, # of reservations)
- 1.11:** The logic object sends a request to the log object to log the changes (customer info)

