



Kevin Shibu Chacko - 40241154

Richard Badir - 40249566

SOEN 342 Project Artifacts

SOEN 342

Section II

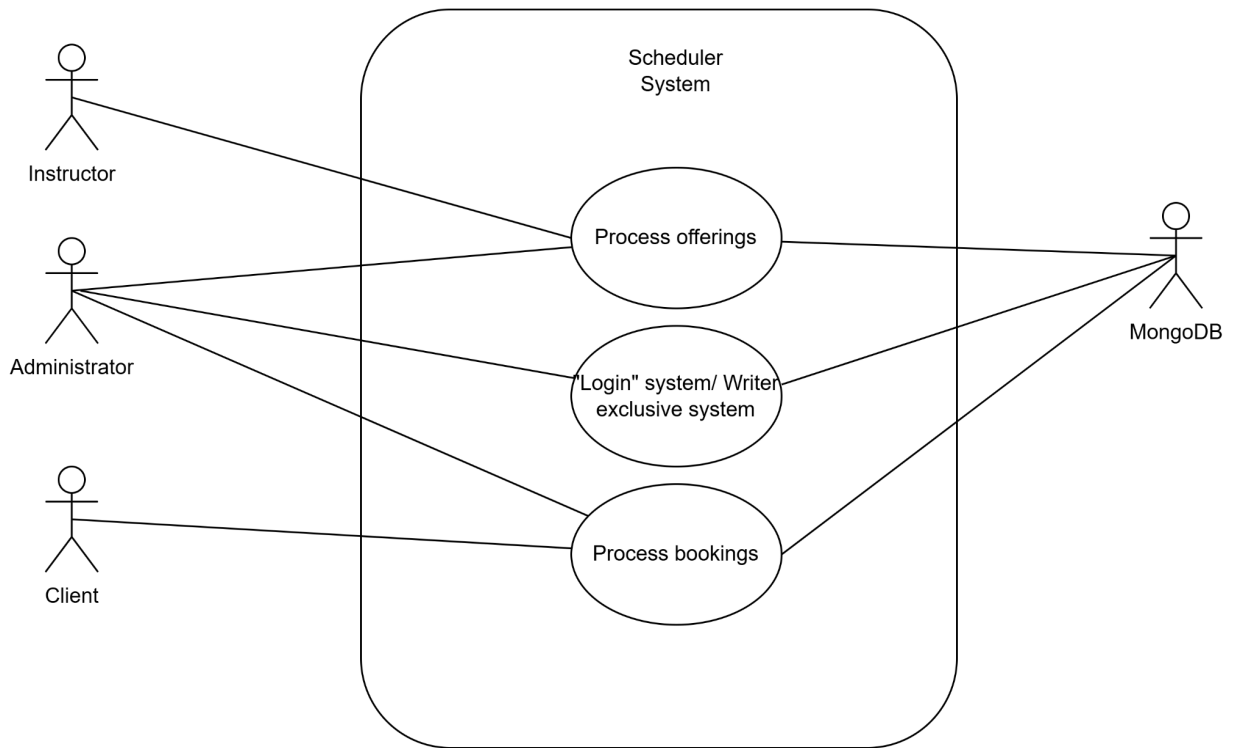
Constantino Constantinides

November 15th, 2024

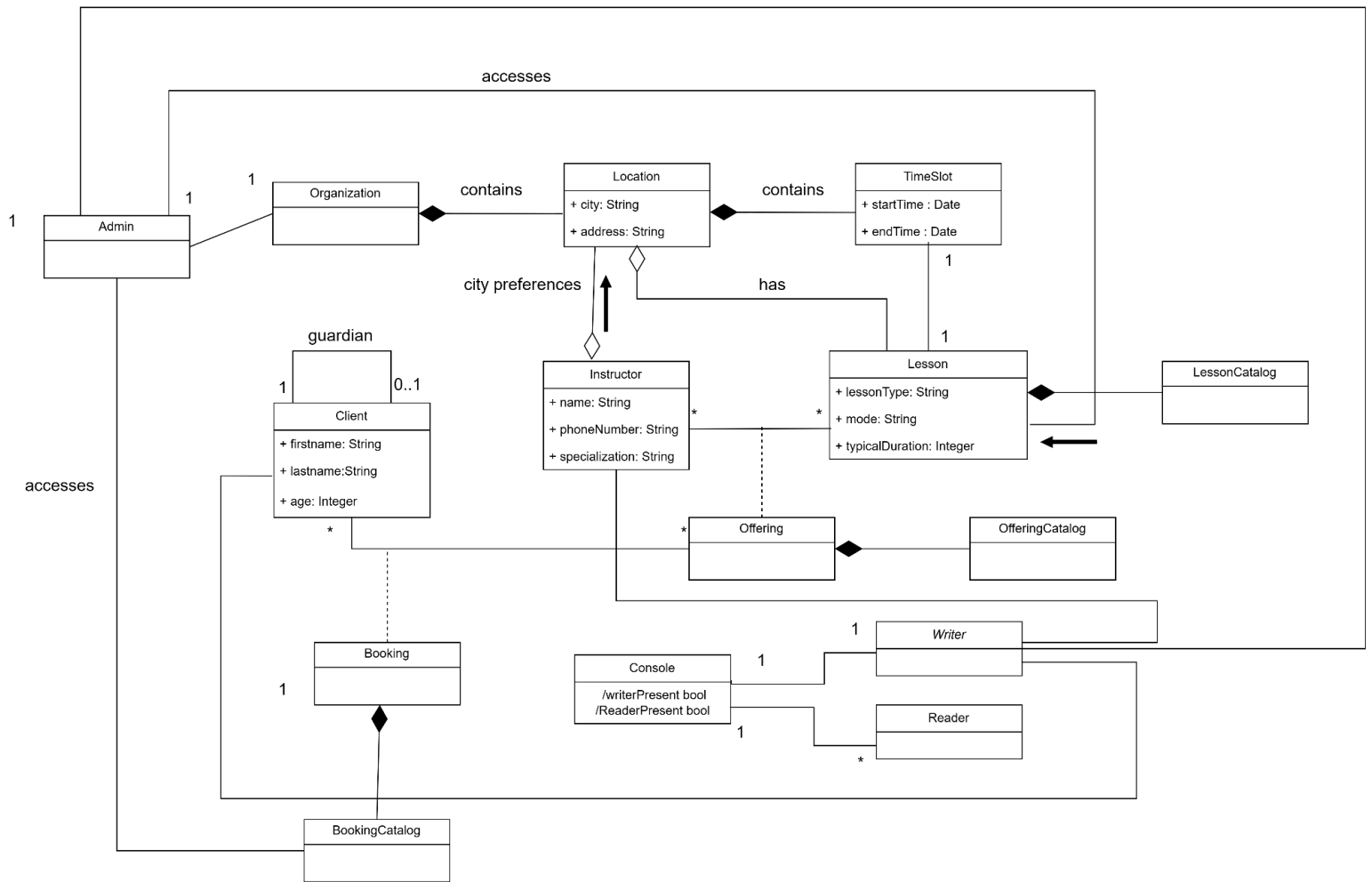
Table of Contents

Table of Contents.....	2
General Use Case:.....	2
Domain Model:.....	3
Success System Sequence Diagrams + System Operations:	
Use Case 1: Process Offerings.....	4
Use Case 2: Process Bookings.....	5
Failure System Sequence Diagrams:.....	6
Use Case 1: Process Offerings.....	6
Contract Operations:.....	8
Interaction Diagrams:.....	11
Relational Models:.....	18

General Use Case:

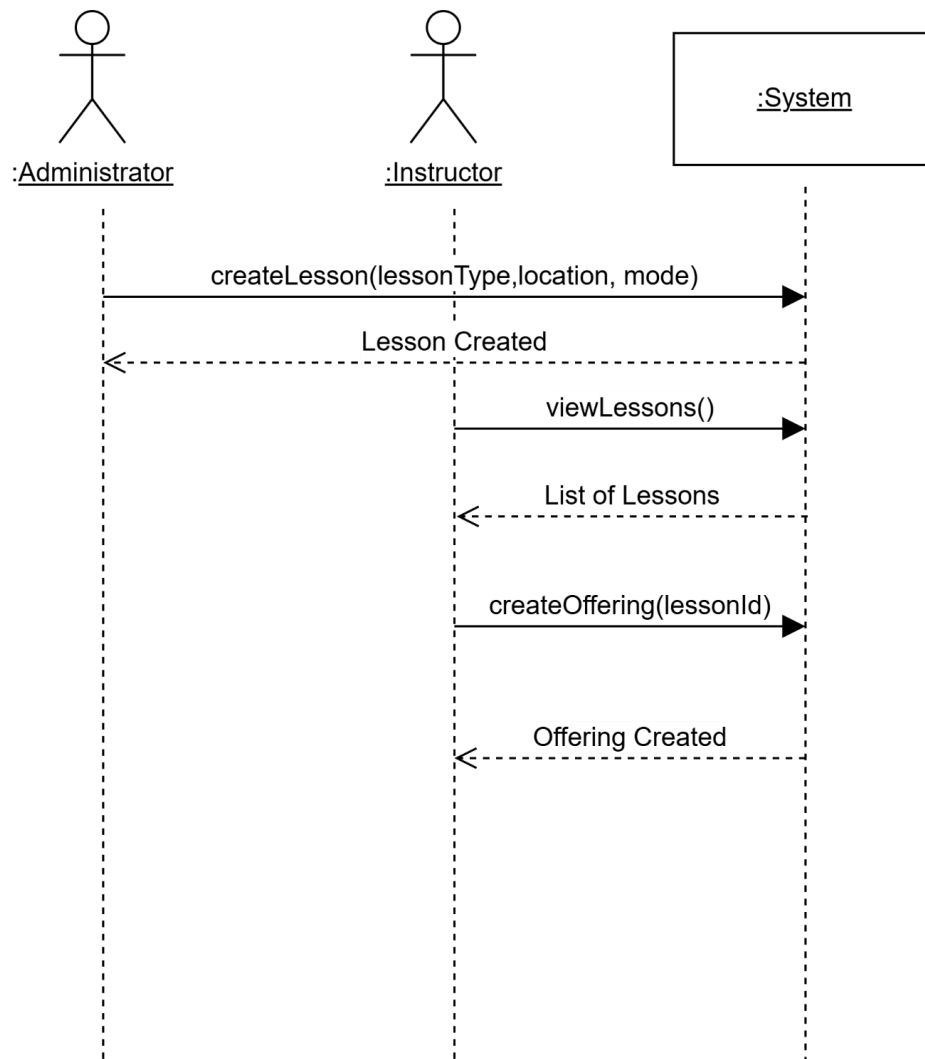


Domain Model:



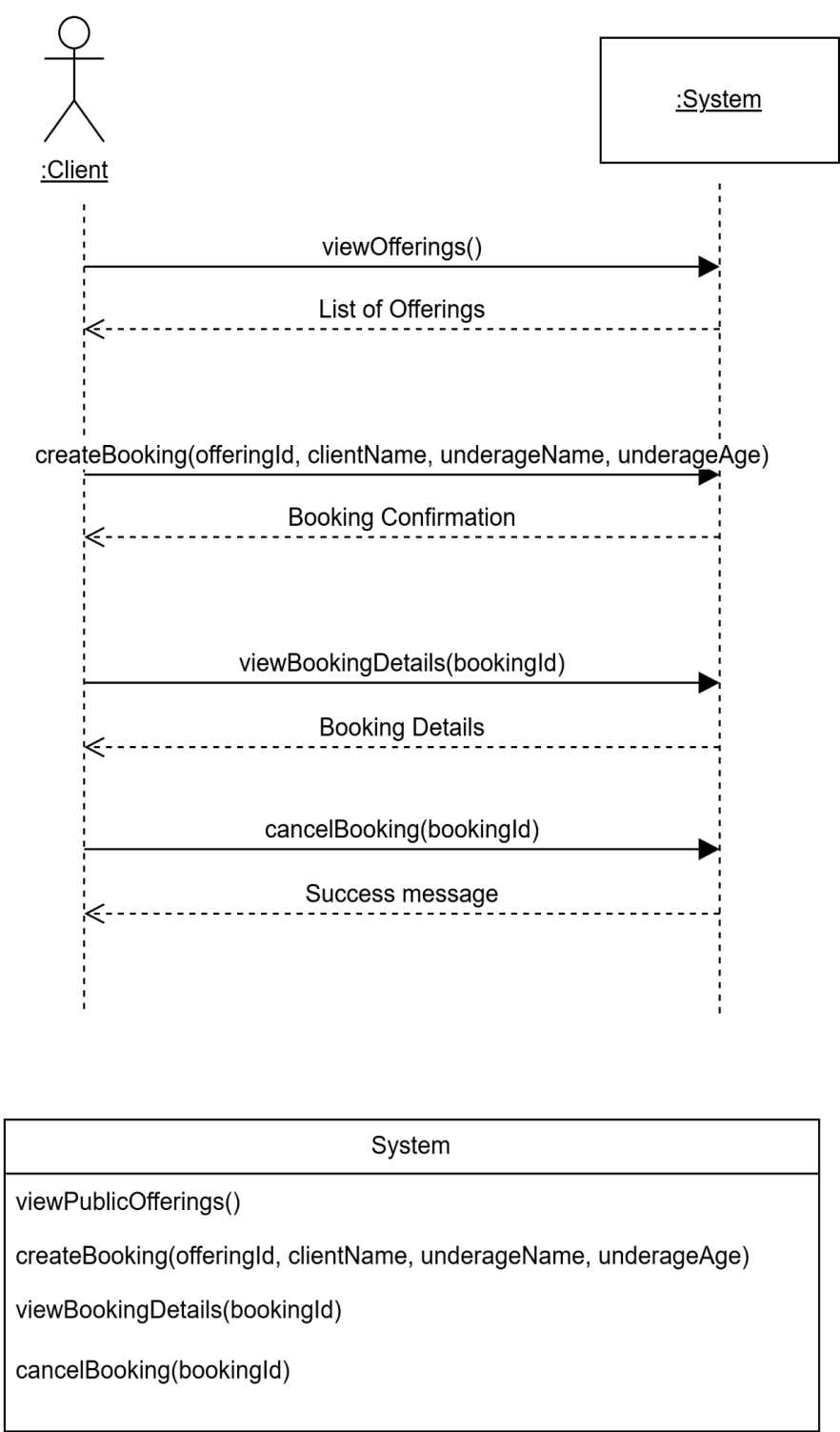
Success System Sequence Diagrams + System Operations:

Use Case 1: Process Offerings



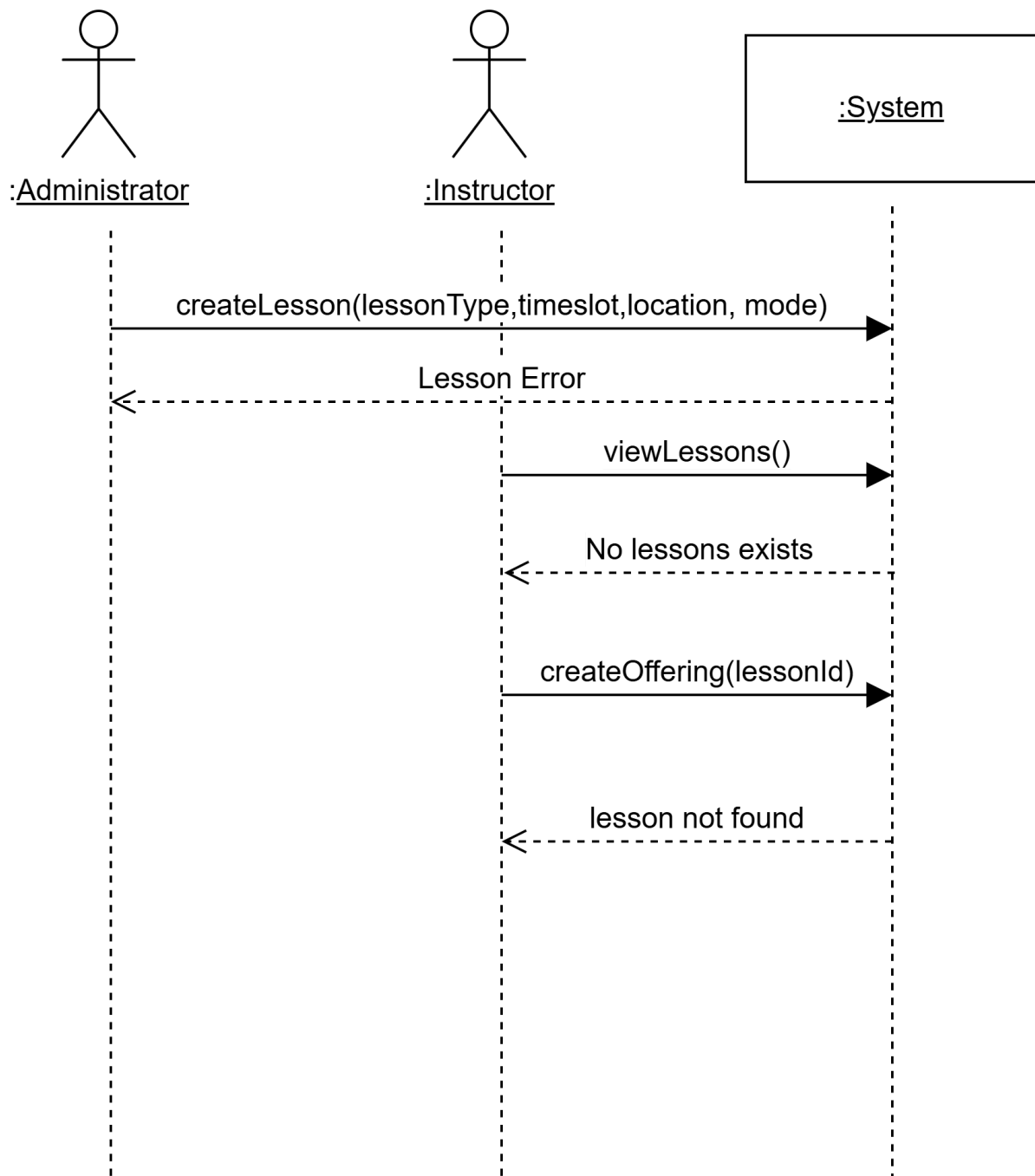
System
<code>createLesson(lessonType,location, mode)</code>
<code>viewLessons()</code>
<code>createOffering(lessonId)</code>

Use Case 2: Process Bookings

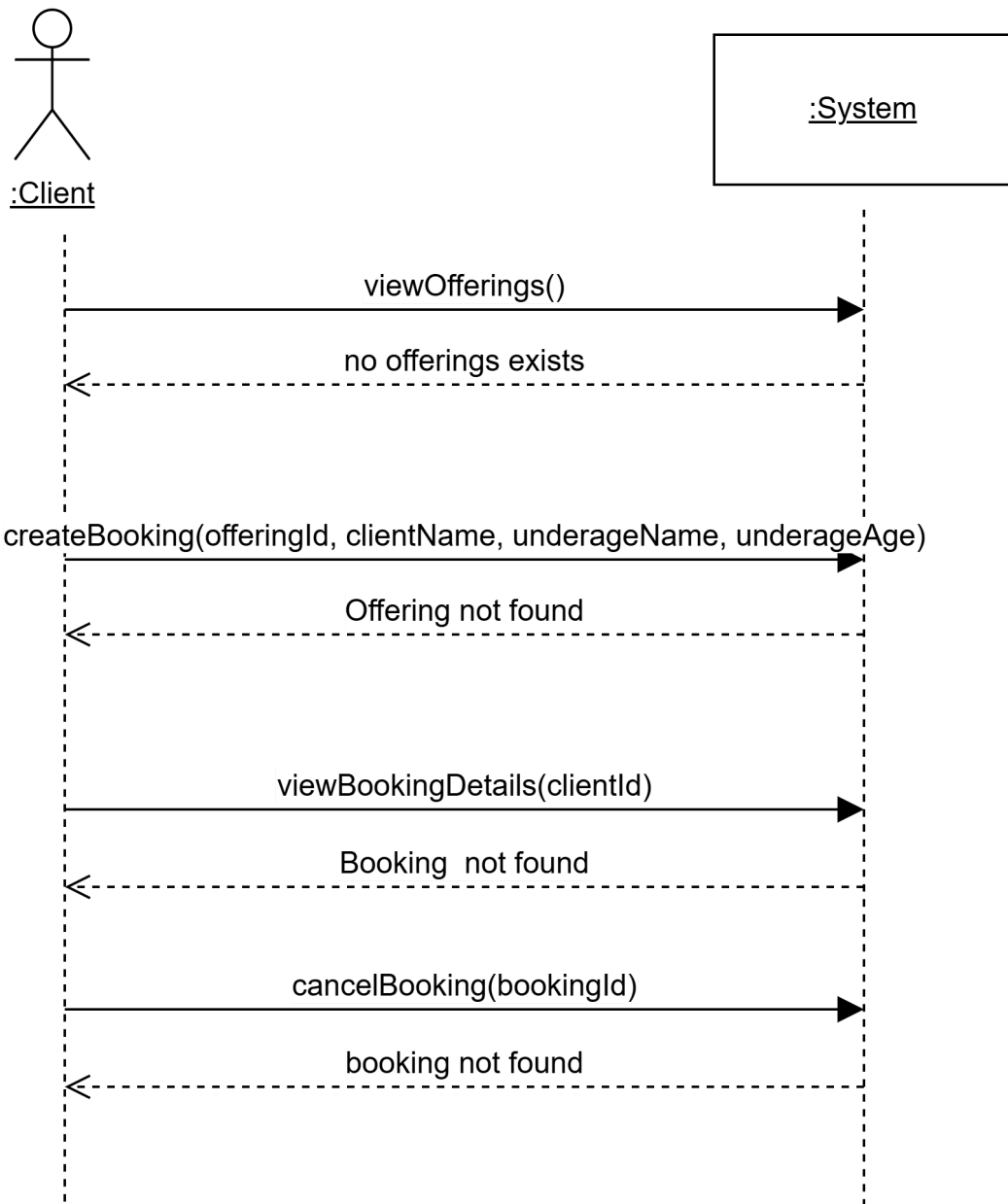


Failure System Sequence Diagrams:

Use Case 1: Process Offerings



Use Case 2: Process Bookings



Contract Operations:

CO1	<p>Contract: createLesson</p> <p>Operation: createLesson(lessonType: String, mode: String, Location: Location, TimeSlot: TimeSlot)</p> <p>Cross reference(s): Use Case Processing Offerings</p> <p>Preconditions:</p> <ol style="list-style-type: none">1. The operation is performed by administrator2. The lessonType is valid and exists in the system3. The mode is either 'private' or 'group'4. The specified location exists in the system and has available time slots <p>Postconditions:</p> <ol style="list-style-type: none">1. A new Lesson instance is created in the system (instance creation)2. the new Lessons' attributes are set: lessonType, timeslot, location, mode (attribute modification).3. A new lessonId is generated and assigned to new Lesson (attribute modification)4. An association is formed between the new Lesson and specified Location (Association formed)
CO2	<p>Contract: viewLessons</p> <p>Operation: viewLessons()</p> <p>Cross reference(s): Use Case Processing Offerings</p> <p>Preconditions:</p> <ol style="list-style-type: none">1. The instructor is logged in and authenticated <p>Postconditions:</p> <p>none</p>
CO3	<p>Contract: createOffering</p> <p>Operation: createOffering(lessonId: string)</p> <p>Cross reference(s): Use Case Processing Offerings</p> <p>Preconditions:</p> <ol style="list-style-type: none">1. The Instructor is logged in and authenticated2. The specified lessonId exists in the system3. The status of lesson is available (not taken by another Instructor)4. The Instructor's specialization match the lessons lesson type. <p>Postconditions:</p> <ol style="list-style-type: none">1. A new Offering instance is created in the system (instance creation)2. A new OfferingId is generated and assigned to new Offering (attribute modification)3. The Lessons status is changed from 'available' to 'taken'. (attribute modification)4. An Association is formed between Instructor and Lesson (Association formed)

CO4

Contract: viewOfferings

Operation: viewOfferings()

Cross reference(s): Use Case Processing Booking

Preconditions:

1. The Client is logged in and authenticated

Postconditions:

None

CO5

Contract: createBooking

Operation: createBooking(offeringId: String, clientName: String, underageName: String, underageAge: String)

Cross reference(s): Use Case Processing Booking

Preconditions:

1. The Client is logged in and authenticated
2. The specified offeringId exists in the system and is valid
3. The status of the Offering is available (not booked)
4. If underageName is provided, underageAge must also be provided
5. if underageAge is provided must be under 18.
6. If Offering mode is 'group', available spots must be greater than 0

Postconditions:

1. A new Booking instance is created in the system (instance creation)
2. The Booking's attributes are set: offeringId, clientName, underageAge (attribute modification)
3. A new BookingId is generated and assigned to new Booking (attribute modification)
4. The Booking's attribute status is set to 'active'
5. If Offering mode is 'private', the Offering's status is updated to 'booked' (attribute modification)
6. If Offering mode is 'group'
 - The Offering's available spots are decreased by 1 (attribute modification)
 - If available spots reach 0, the Offering's status is updated to 'booked' (attribute modification)
7. An association is formed between the Booking and Client (Association formed)
8. An association is formed between the Booking and the specified Offering (association formed)

CO6

Contract: viewBookingDetails

Operation: viewBookingDetails(clientId:String)

Cross reference(s): Use Case Processing Booking

Preconditions:

1. The Client is logged in and authenticated

Postconditions:

None

CO7

Contract: CancelBooking

Operation: cancelBooking(bookingId: String)

Cross reference(s): Use Case Processing Booking

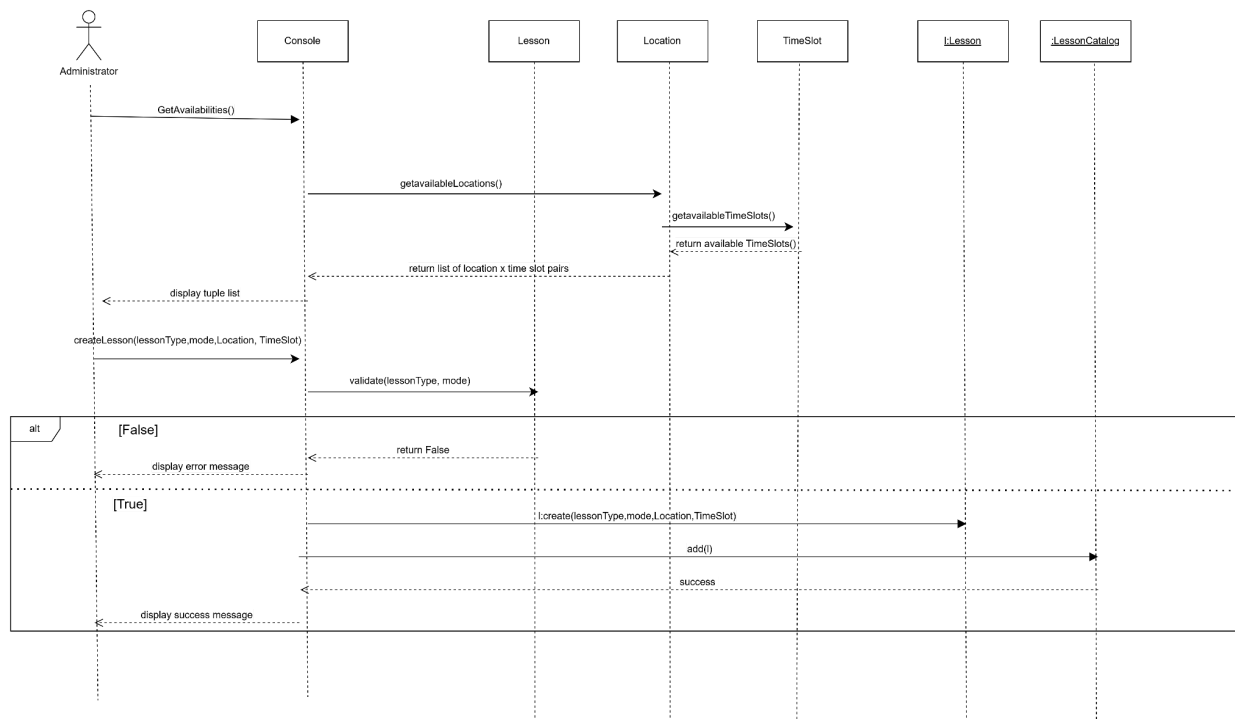
Preconditions:

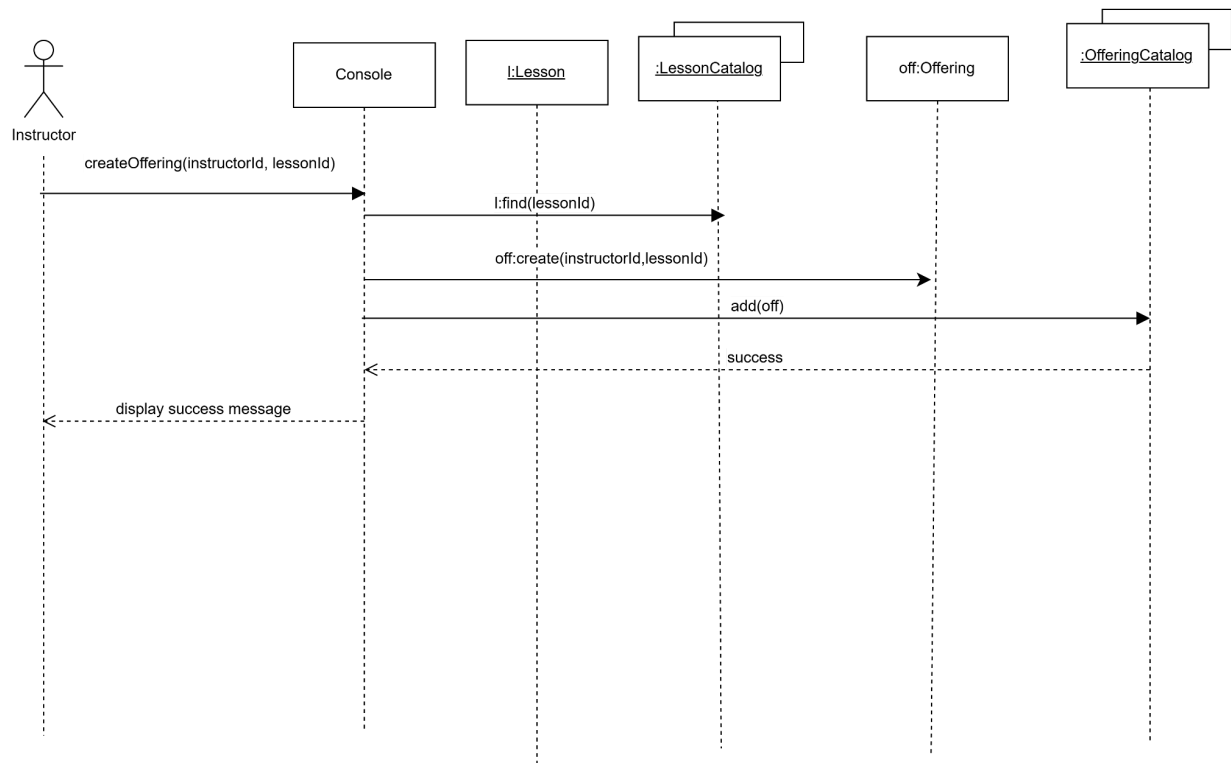
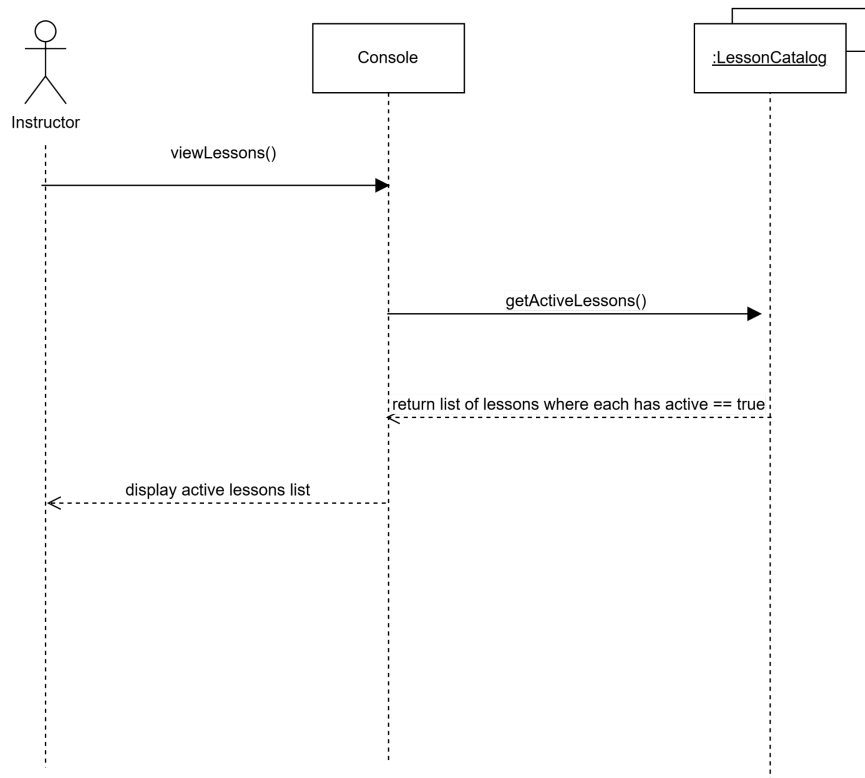
1. The Client is logged in and authenticated
2. The specified bookingId exists in the system
3. The Booking belongs to the requesting Client
4. The Booking status is 'active'

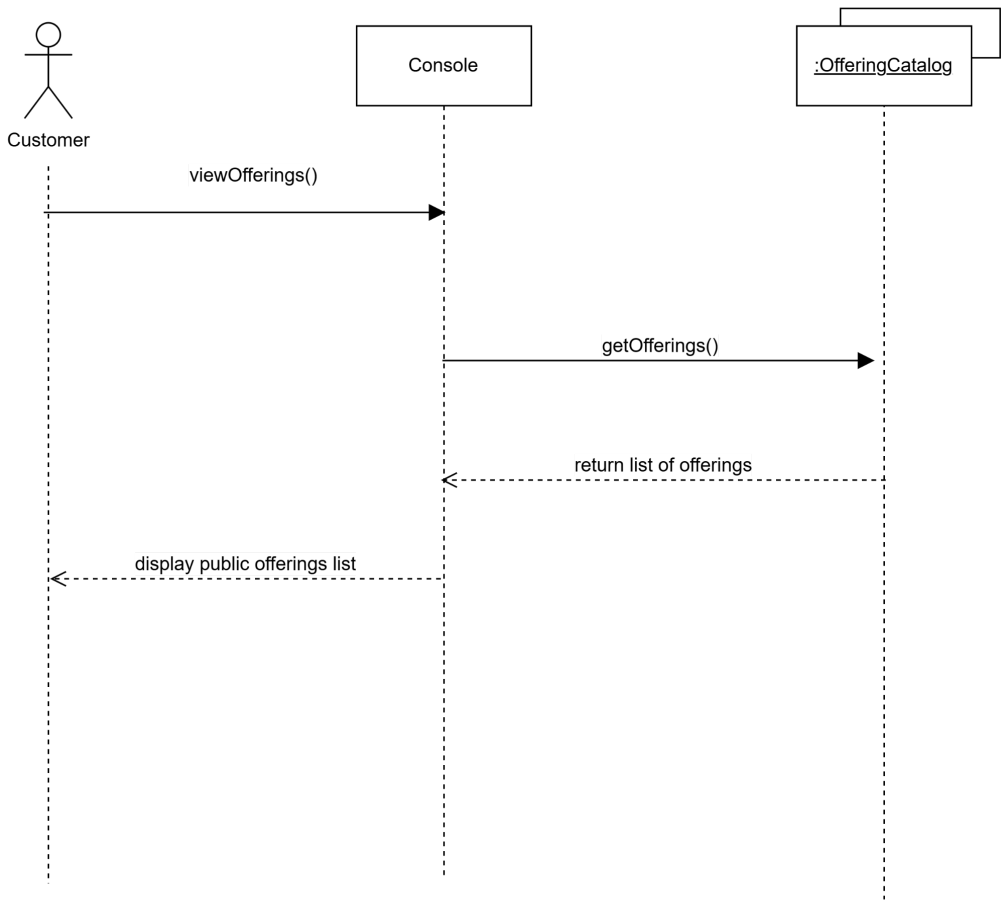
Postconditions:

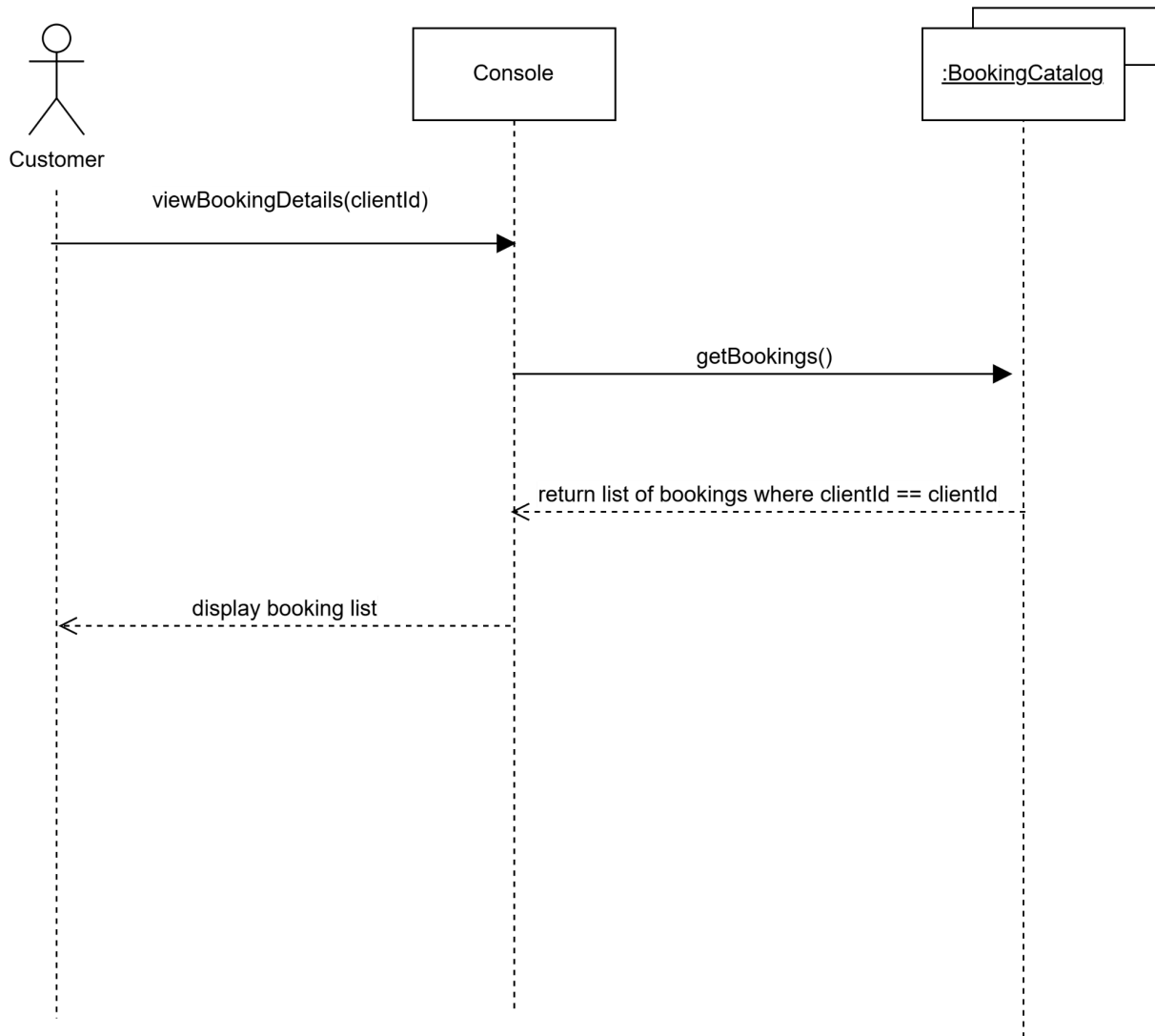
1. The Booking's status is updated to 'cancelled' (attribute modification)
2. if associated Offering mode is private, the Offering's status is updated to 'available' (attribute modification)
3. If associated Offering mode is 'group'
 - The Offering's available spots in increased by 1 (attribute modification)
 - The Offering's status is updated to 'available; if it was 'booked' (attribute modification)
4. Association between Booking and Client is removed (Association removed)
5. Association between Booking and Offering is removed (Association removed)
6. Booking instance is deleted (instance deletion)

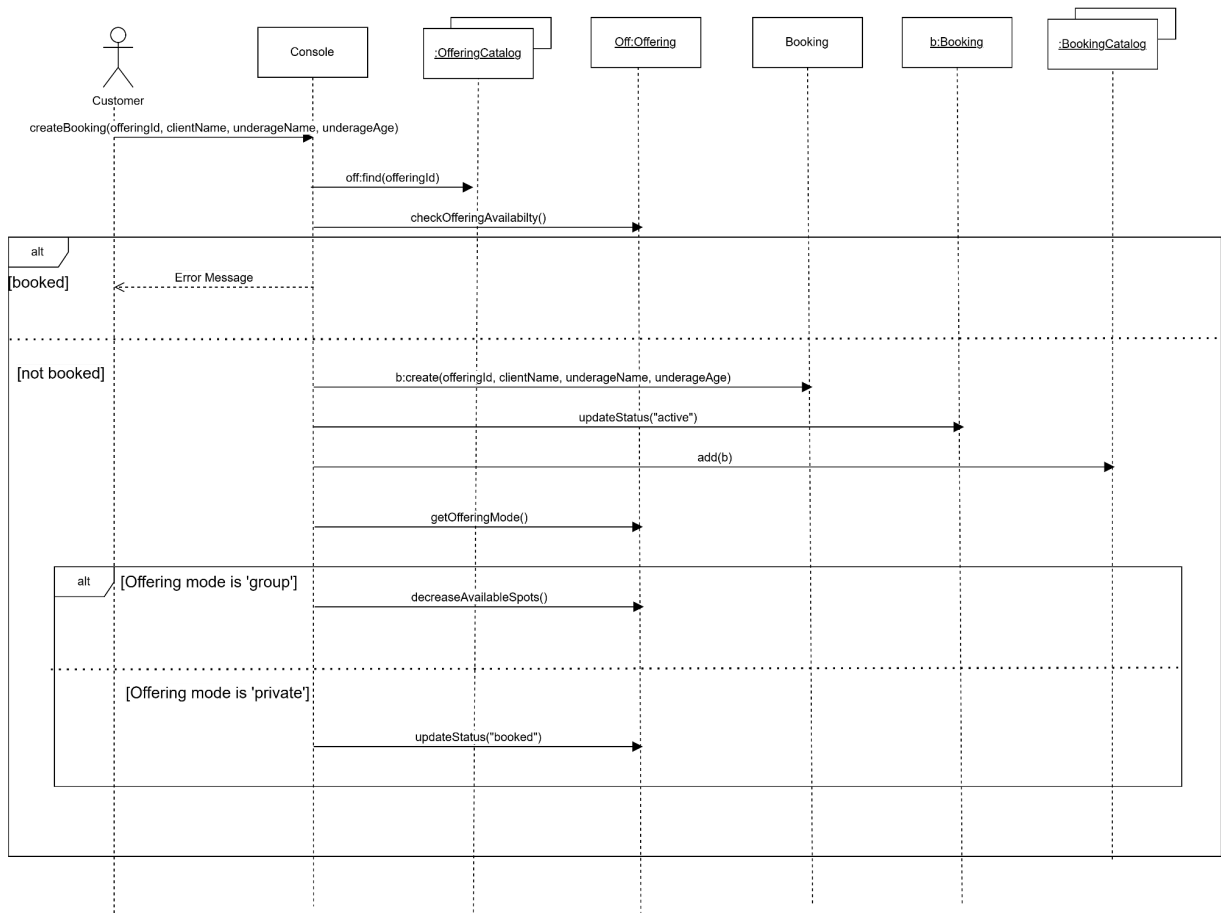
Interaction Diagrams:

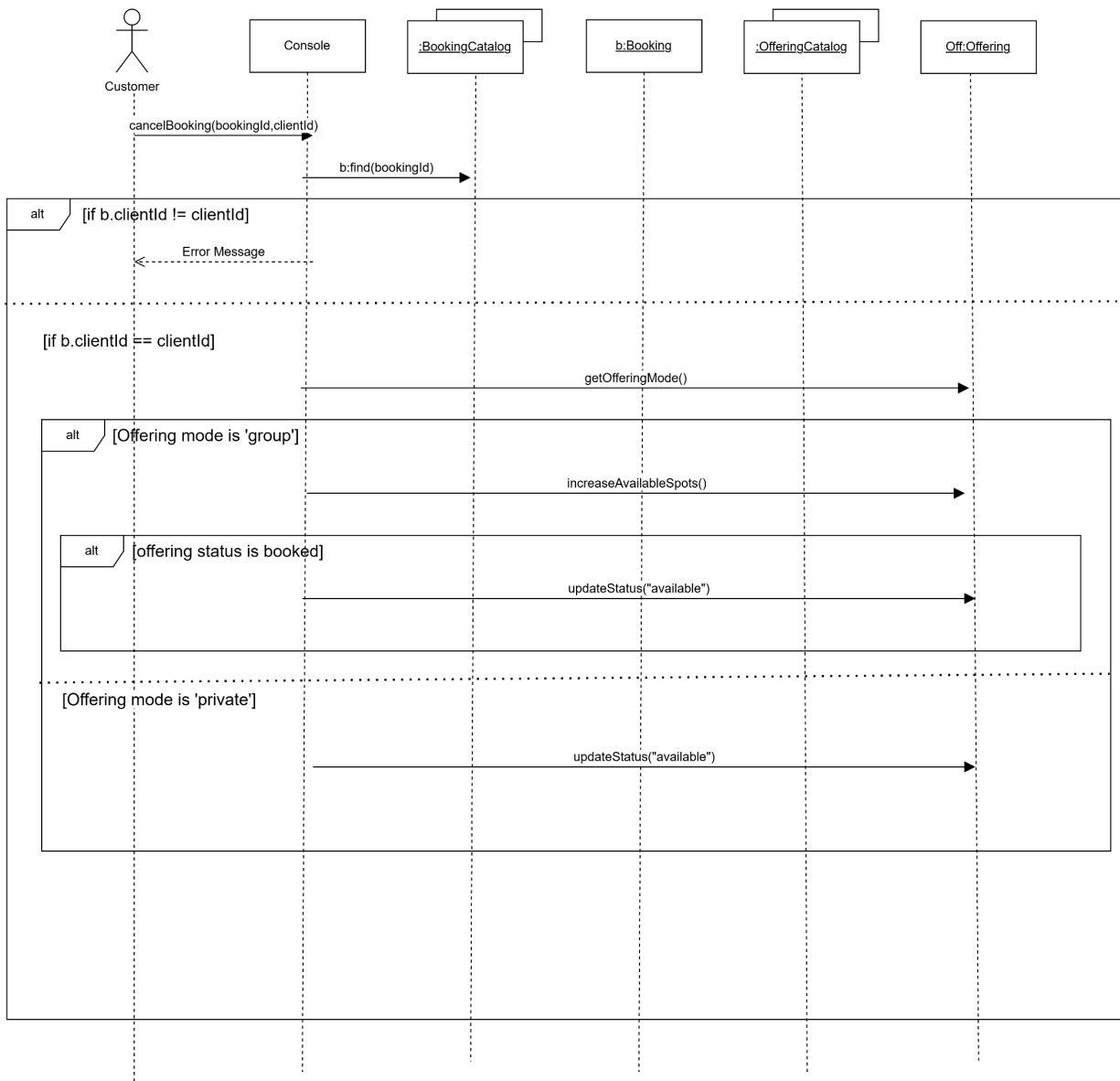


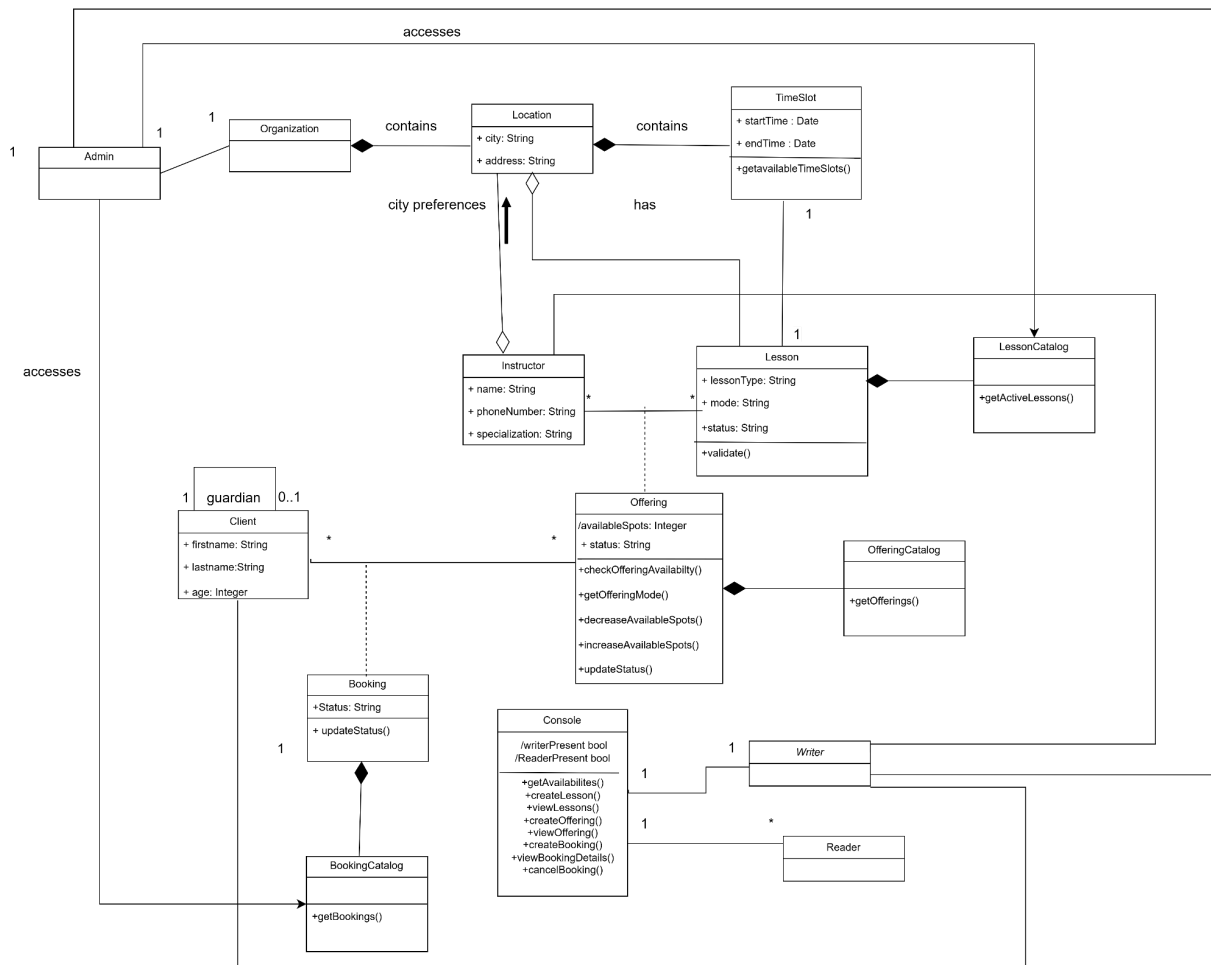












Relational Models:

Organizations
Id <<PK>> name

Locations
Id <<PK>> organizationId <<FK>> city address

TimeSlots
Id <<PK>> locationId <<FK>> startTime endTime

Instructors
Id <<PK>> name phoneNumber specialization

Lessons
Id <<PK>> timeslotId <<FK>> lessonType mode status

Offerings
Id <<PK>> instructorId <<FK>> lessonId <<FK>> availableSpots status

Clients
Id <<PK>> guardianId <FK> firstName lastName age

Bookings
Id <<PK>> clientId <<FK>> offeringId <<FK>> status

Admins
Id <<PK>> organizationId <<FK>>