

Software Design Specifications

STUDENT MANAGEMENT SYSTEM FAST FLYERS

Version: 001

Project Code	
Supervisor	Ms. Rubab Jaffar
Co Supervisor	TA
Project Team	Amna Mansoor: 22K-5159 Alizah Basit: 22K-5155 Hafsa Salman: 22K-5161
Submission Date	03 May, 2024

[Instructions]

- *No section of template should be deleted. You can write 'Not applicable' if a section is not applicable to your project. But all sections must exist in the final document.*
- *All comments/examples mentioned in square brackets ([]) are in the template for explanation purposes and must be replaced / removed in the final document.*
- *This 'Instruction' section should also be removed in the final document.*
- *MS-Word Reviewing feature must be used to get the document reviewed by PMs or supervisors.*

Document History

Version	Name of Person	Date	Description of change
01	Amna Mansoor	1/4/24	Document Created
02	Alizah Basit	4/4/24	Added Design Considerations and Design Strategy
03	Hafsa Salman	17/4/24	Added Activity Diagrams
04	Amna Mansoor	20/4/24	Added Class Diagram
05	Hafsa Salman	21/4/24	Added State Diagrams
06	Alizah Basit	21/4/24	Added Sequence Diagrams
07	Alizah Basit	1/5/24	Added Collaboration Diagram
08	Hafsa Salman	3/5/24	Added Component Diagram
09	Amna Mansoor	3/5/24	Added Deployment Diagram

Distribution List

Name	Role
Ms. Rubab Jaffar	Supervisor
TA	Co Supervisor

Document Sign-Off

Version	Sign-off Authority	Project Role	Signature	Sign-off Date

Document Information

Category	Information
Customer	FAST-NU
Project	Student Management System
Document	Software Design Specification
Document Version	1.0
Status	Draft
Author(s)	Amna Mansoor, Alizah Basit, Hafsa Salman
Approver(s)	
Issue Date	
Document Location	
Distribution	Advisor Project Coordinator's Office (through Advisor)

Definition of Terms, Acronyms and Abbreviations

Term	Description
ASP	Active Server Pages
DD	Design Specification
SDA	Software Design and Architecture
SMS	Student Management System

Term	Description

Table of Contents

1	Introduction	8
1.1	<i>Purpose of Document</i>	8
1.2	<i>Intended Audience</i>	8
1.3	<i>Document Convention</i>	8
1.4	<i>Project Overview</i>	8
1.5	<i>Scope</i>	9
2	Design Considerations	9
2.1	<i>Assumptions and Dependencies</i>	9
2.2	<i>Risks and Volatile Areas</i>	9
3	System Architecture	9
3.1	<i>System Level Architecture</i>	10
3.2	<i>Software Architecture</i>	11
4	Design Strategy	11
5	Detailed System Design	12
5.1	<i>Database Design</i>	12
5.1.1	ER Diagram	12
5.1.2	Data Dictionary	12
5.1.2.1	Data 1	13
5.2	<i>Application Design</i>	14
5.2.1	Class Diagram	14
5.2.2	Activity Diagram	14
5.2.3	Sequence Diagram	14
5.2.4	Collaboration Diagram	14
5.2.5	Sequence Diagram	14
6	References	15
7	Appendices	16

1 Introduction

1.1 Purpose of Document

The main purpose of this document is to illustrate the software design for the Student Management System (SMS). The document gives the scope as well as detailed description of system design. This document aims to provide an overview of this product including the system requirements and features. This document will provide all the necessary details to all the stakeholders and major users of the system.

1.2 Intended Audience

This document is designed for the Course Instructor for Software Design and Architecture (SDA): Miss Rubab Jaffar, and the developers of the System, stakeholders, administration and faculty of FAST NUCES, Karachi.

1.3 Document Convention

- Convention for Main Heading:

Font Face: Arial

Font Style: Bold and Italic

Font Size: 16

- Convention for Subheading:

Font Face: Arial

Font Style: Bold and Italic

Font Size: 12

- Convention for Body:

Font Face: Arial

Font Style: None

Font Size: 11

1.4 Project Overview

The aim for developing a student management system is to bring the administrative staff, teachers and students on the same platform. It will provide an efficient and comprehensive platform where administrative staff can offer courses and manually assign courses to teachers and students and do various other tasks. Teachers can register their courses that they want to teach in the following semester, upload and edit student's attendance and marks. Whereas, a student can view their attendance, marks and their grade. They can also register for the courses. The customer for the student management system is FAST NUCES. This project is a new product. It will be delivered in 9 weeks.

1.5 Scope

The project will deliver a fully functional student management system encompassing modules for course registration, attendance tracking, grade management, fee generation, and transcript generation.

The student management system will not include integration with external systems or advanced analytics features beyond the scope defined in the Software Design Specification.

2 Design Considerations

[This section describes many of the issues which need to be addressed or resolved before attempting to devise a complete design solution. In other words, this section is used to formally set the groundwork for the system design.]

2.1 Assumptions and Dependencies

- Hardware Compatibility: The system should be designed to be compatible with standard hardware specifications that support Windows OS.
- File System: System should have a stable file system with adequate read/write permissions for file based operations such as marking attendance, marks etc.
- Localization: While the current GUI supports English only, the design should be capable of accommodation for future localization without major problems arising.

2.2 Risks and Volatile Areas

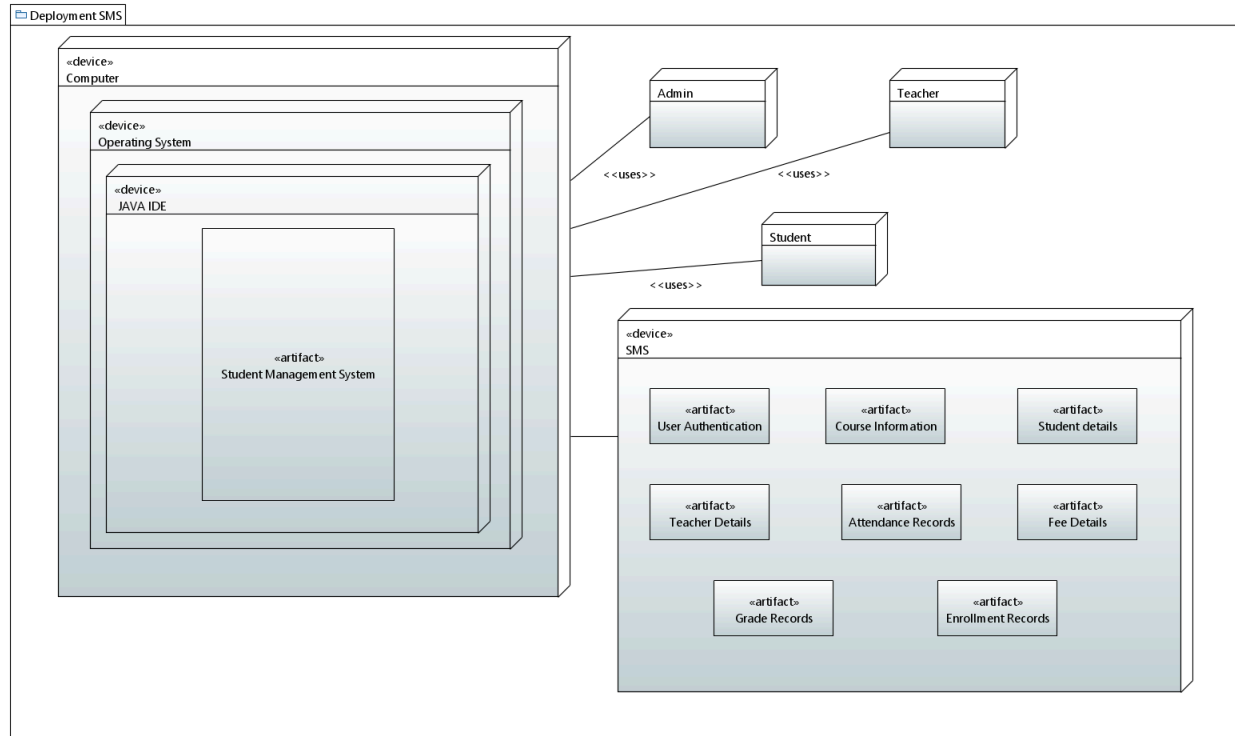
- Technological Evolution: Design should be flexible enough to be integrated with database systems if needed.
- Operating System Updates: Significant changes in the Windows Os could affect the SMS hence the design should follow best practices for OS compatibility.
- Security Vulnerabilities: As security threats evolve the system might need to update its authentication methods hence the design should allow easy updates to the security protocols.
- User Interface Trends: The system should have a separate layer for GUI, to allow the necessary redesign of UI according to the user trends.
- Documentation: There should be adequate and thorough documentation to ensure that future developers can understand and modify the system as needed.

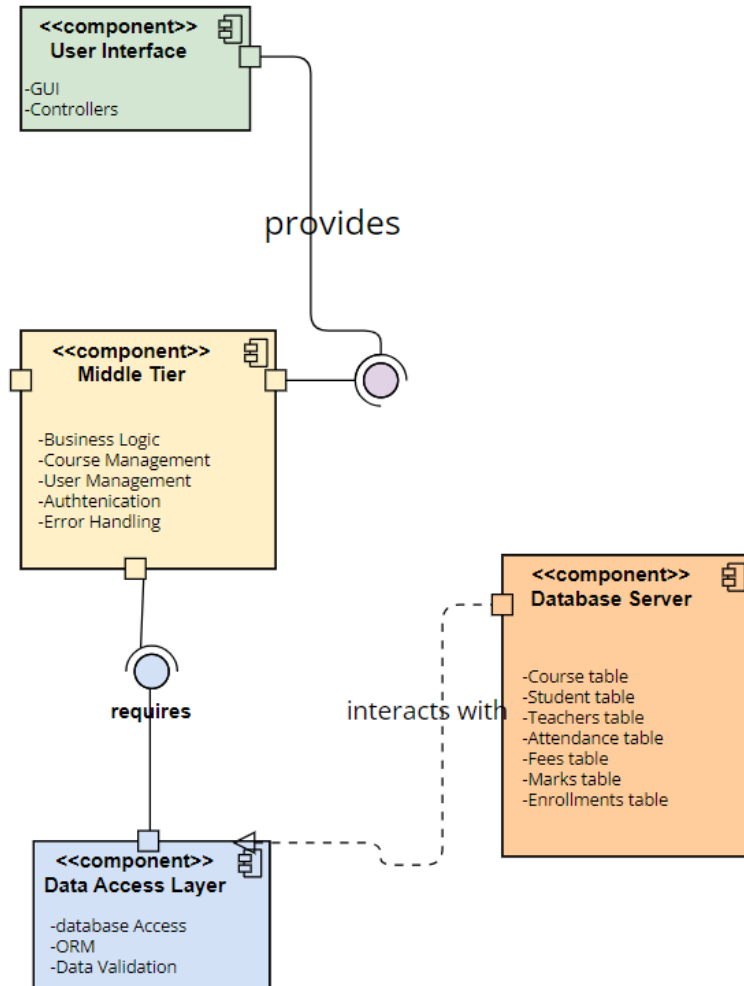
3 System Architecture

3.1 System Level Architecture

- User Interface: This layer encompasses the graphical user interface (GUI) that users interact with to perform various tasks such as adding courses, enrolling students, viewing teachers etc.
- Application Logic: The layer contains the logic and business rules of the LMS. It includes functionalities such as course management, student enrollment, teacher assignment, fee management, and attendance tracking.

- **Data Management:** This layer is responsible for managing the data within the system. It includes functionalities for storing and retrieving data related to courses, students, teachers, attendance records, and fees.
- **External Interfaces:** This layer deals with interactions with external systems or services, such as integration with external databases, authentication services, or payment gateways.
- **Error Handling:** Global design strategies for handling errors and exceptions throughout the system, ensuring robustness and reliability.





3.2 Software Architecture

- **User Interface Layer:** This layer provides the graphical interface for users to interact with the system. It includes components such as buttons, input fields, and displays information to the user. The user interface layer communicates with the middle tier to initiate actions based on user input and displays results back to the user.
- **Middle Tier:** The middle tier serves as the intermediary between the user interface layer and the data access layer. It contains the application logic and business rules of the system, handling tasks such as processing user requests, executing business logic, and coordinating interactions between different components. This layer interacts with both the user interface layer for receiving user input and the data access layer for accessing and manipulating data.
- **Data Access Layer:** The data access layer is responsible for interacting with the database to store and retrieve data. It includes components for executing database queries, retrieving data objects, and performing data manipulation operations. This layer abstracts the underlying database implementation from the middle tier, allowing for flexibility and scalability in the storage and retrieval of data.

4 Design Strategy

- Future System Enhancement:
 - System shall have a modular design that adheres to object oriented principles hence can easily integrate new features or modules without disrupting the existing system.
- System Reuse:
 - The system shall prioritize reusability to avoid redundancy.
 - The system shall use standard libraries (like JAVA Swing for GUI) to insure code reuse and consistency.
 - The system shall use encapsulation and abstraction techniques that enable the self-contained system components to be reused in different contexts without little to no modification.
- User Interface Paradigms:
 - System shall offer a user friendly interface suitable for varying levels of technical expertise.
 - System shall use GUI frameworks like JAVA Swing to insure a consistent user experience.
 - System shall be consistently improved in usability through user feedback and an iterative design process.
- Data Management:
 - The system shall implement robust data management strategies to ensure efficient storage, retrieval, and manipulation of information.
 - The system shall adopt a straightforward approach by utilizing simple file-based storage therefore ensuring the ease of implementation and maintenance.
 - The system shall employ secure data persistence mechanisms to safeguard data integrity and confidentiality.
- Concurrency and Synchronization:
 - To maintain data consistency and prevent race conditions during concurrent access, the system shall integrate concurrency control mechanisms.
 - Through the testing and validation of concurrency handling mechanisms shall ensure system reliability and robustness.

5 Detailed System Design

5.1 Database Design

N/A

5.1.1 ER Diagram

N/A

5.1.2 Data Dictionary

N/A

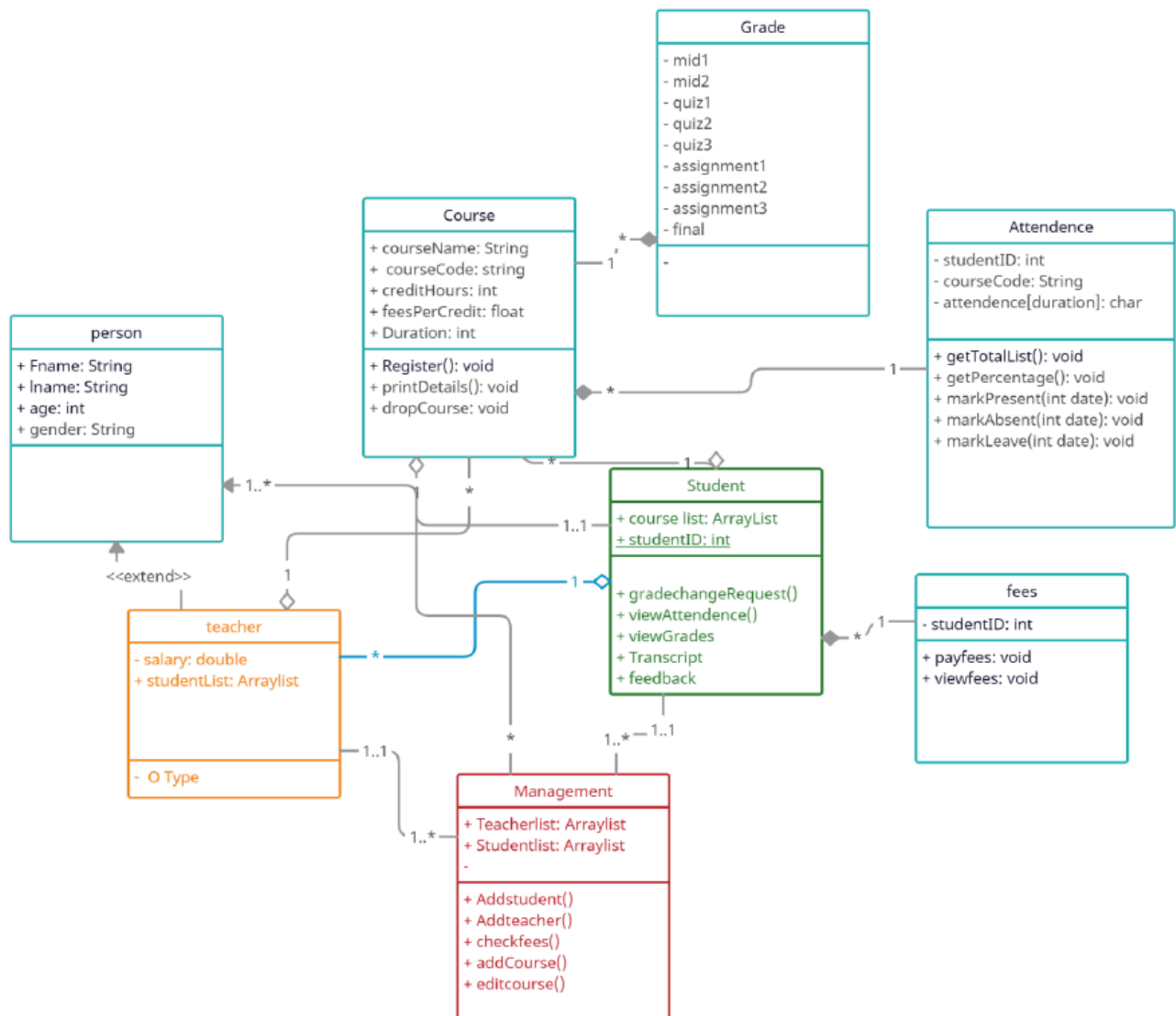
5.1.2.1 Data 1

N/A

< Data 1>						
Name		N/A				
Alias		N/A				
Where-used/how-used		N/A				
Content description		N/A				
Column Name	Description	Type	Length	Table	Default Value	Key Type
N/A	N/A	N/A	N/A	N/A	N/A	N/A
N/A	N/A	N/A	N/A	N/A	N/A	N/A

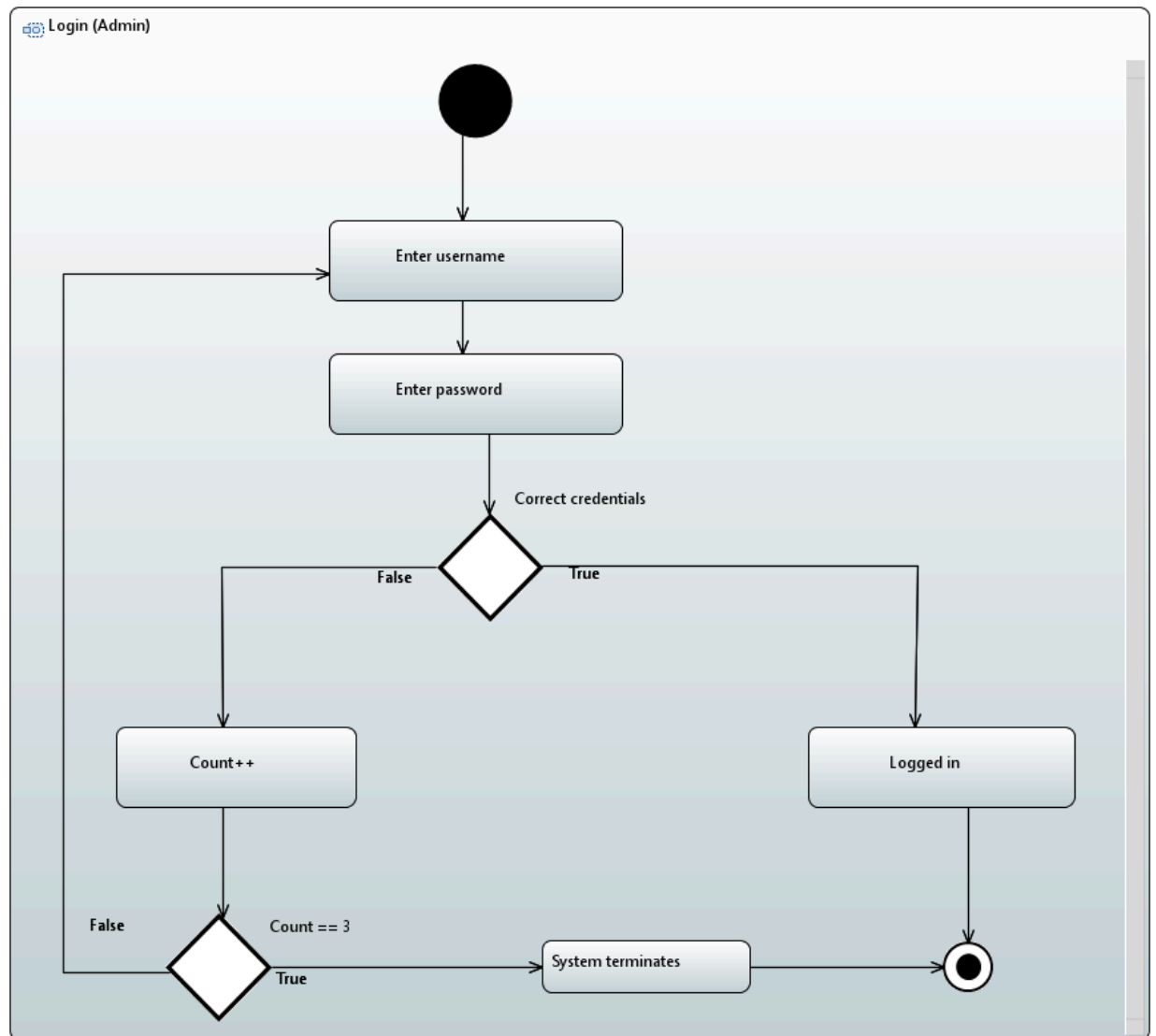
5.2 Application Design

5.2.1 Class Diagram

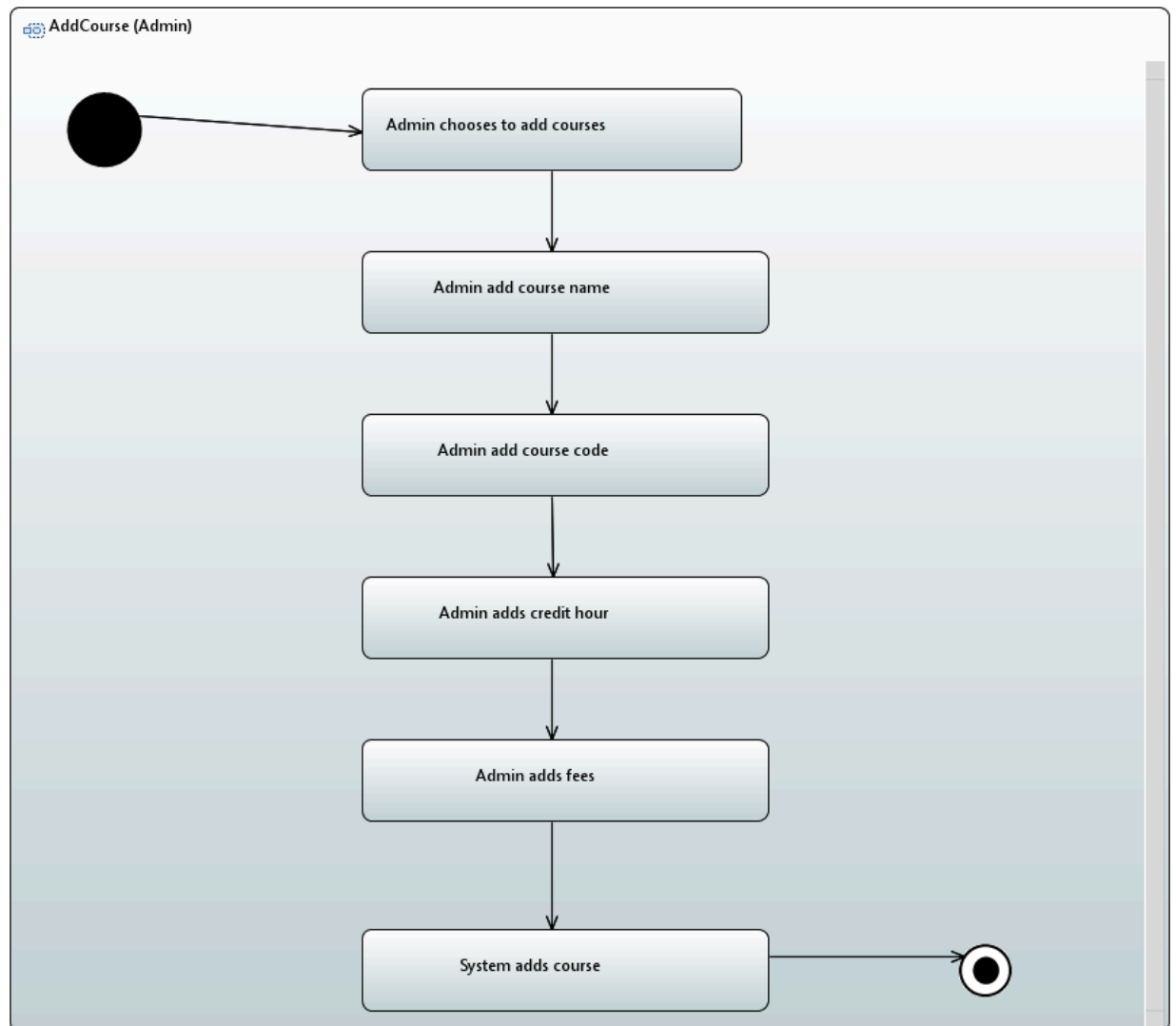


5.2.2 Activity Diagram

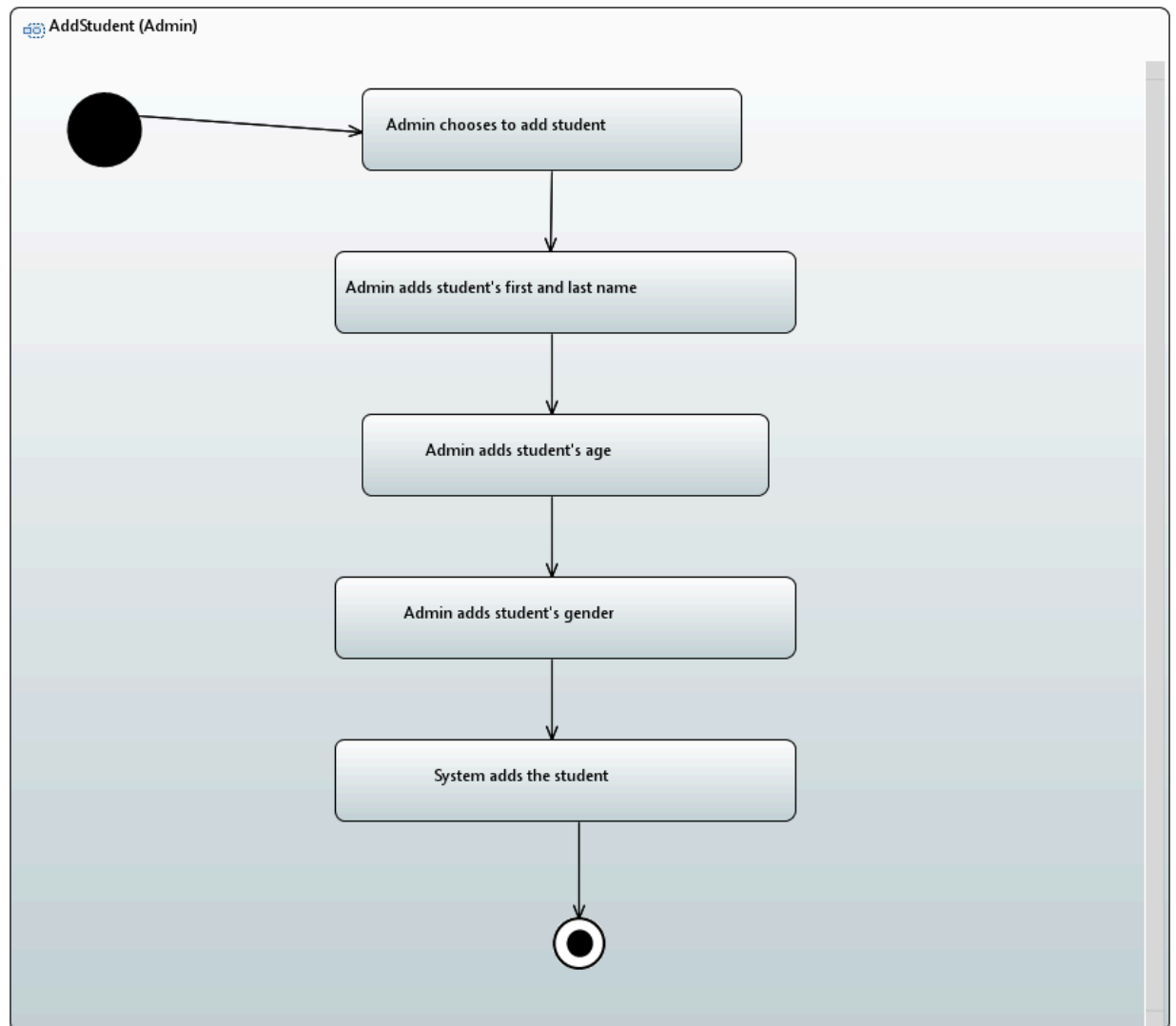
5.2.2.1 Login (Admin)



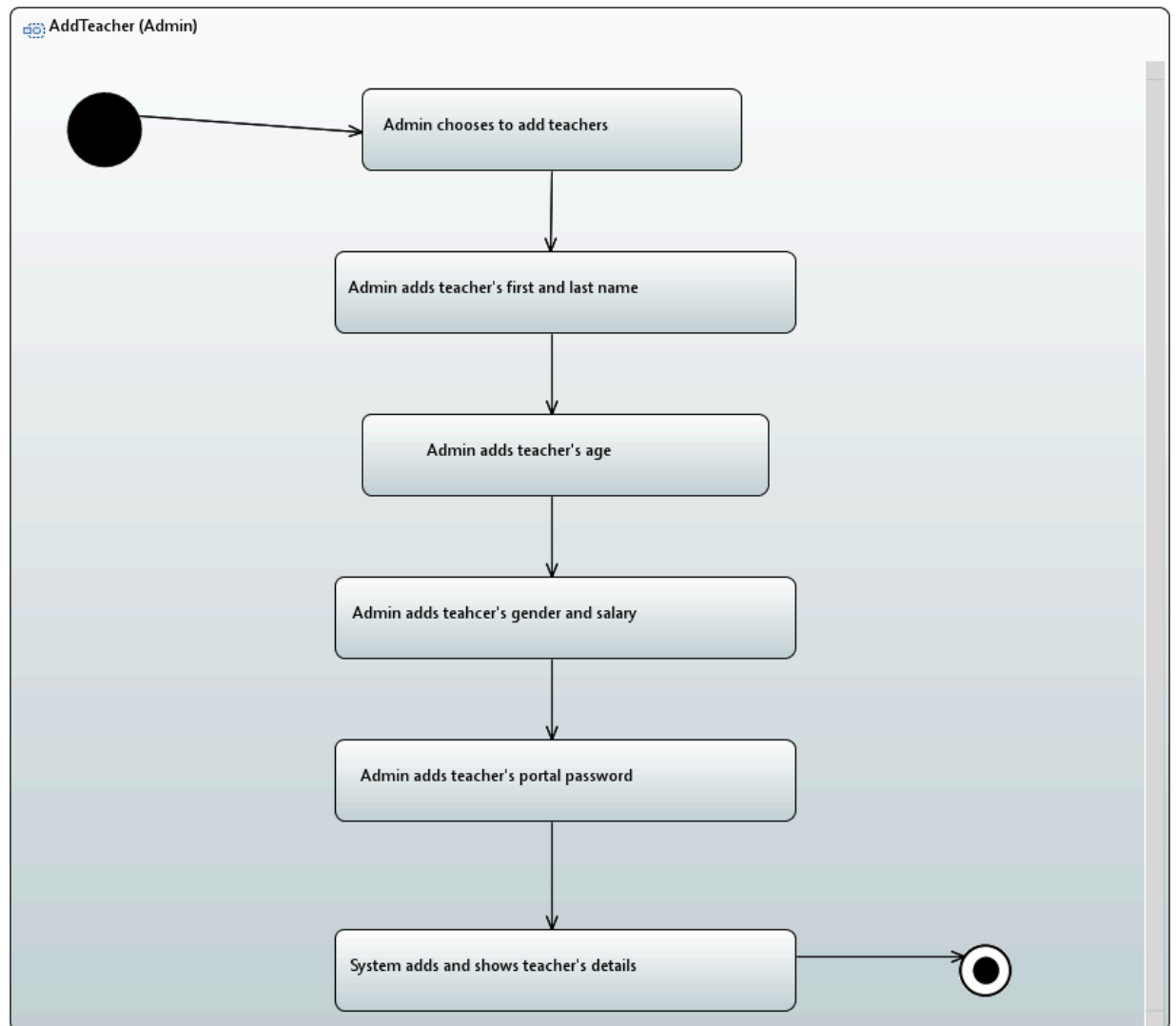
5.2.2.2 Add Course (Admin)



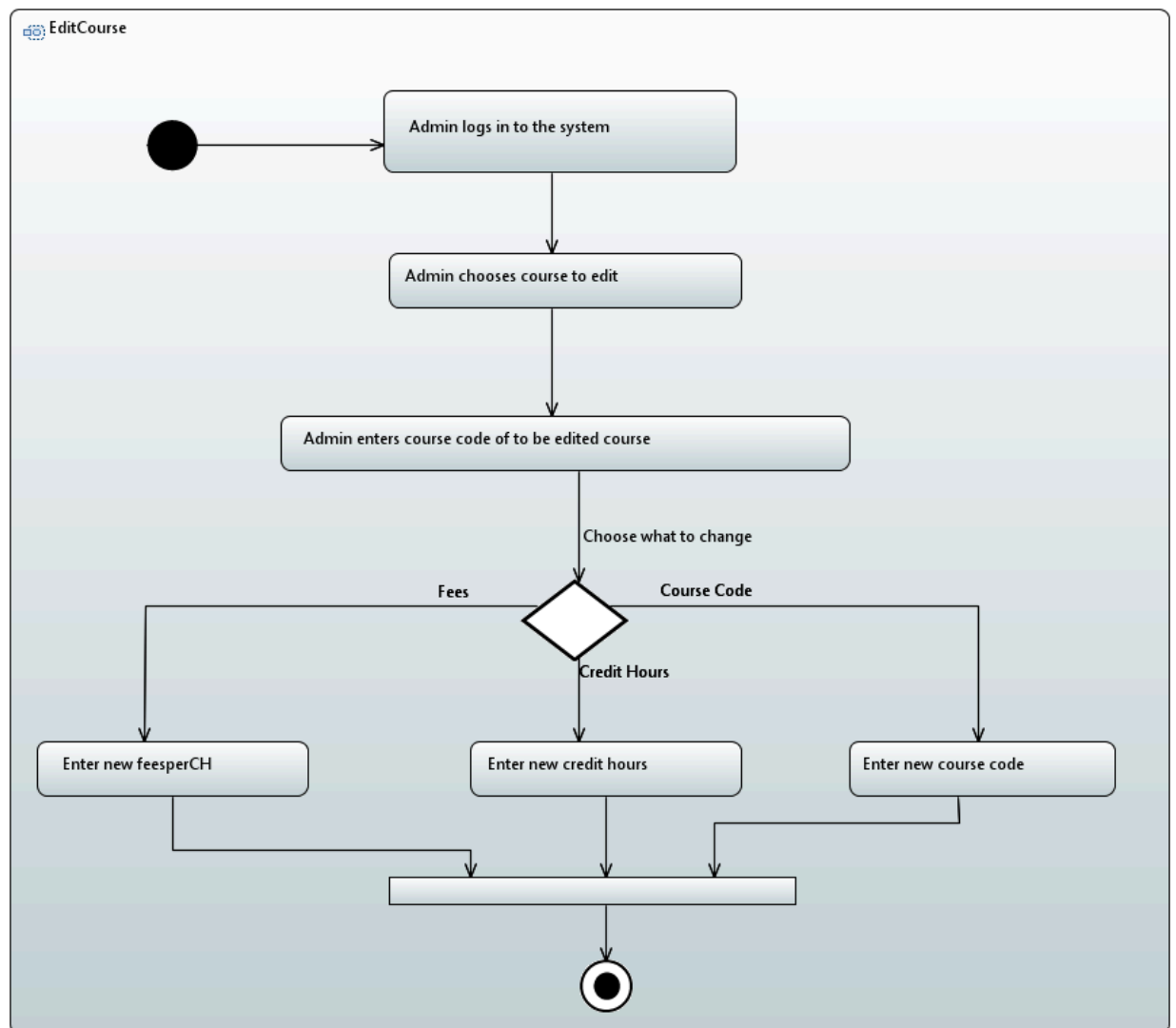
5.2.2.3 Add Student (Admin)



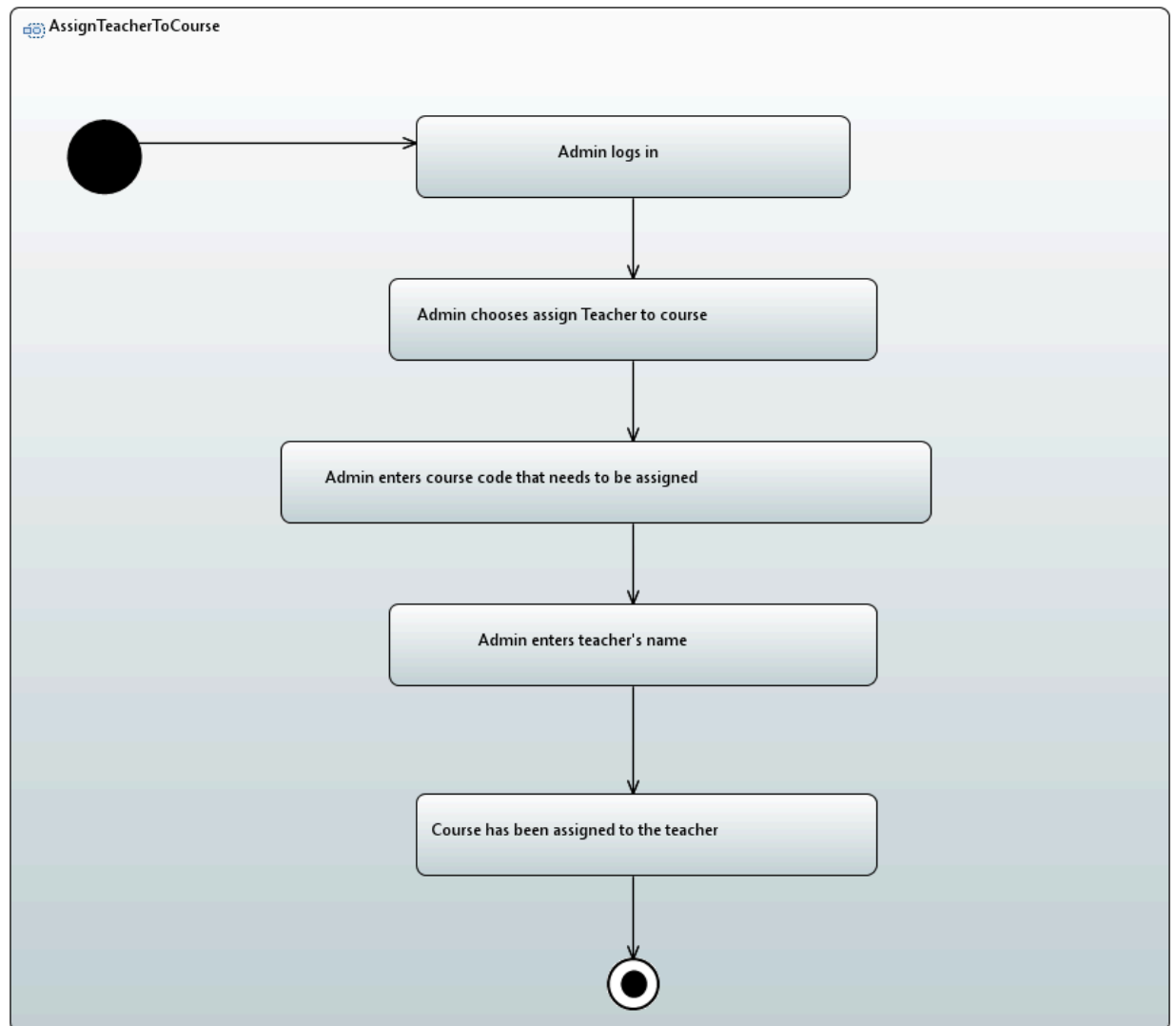
5.2.2.4 Add Teacher (Admin)



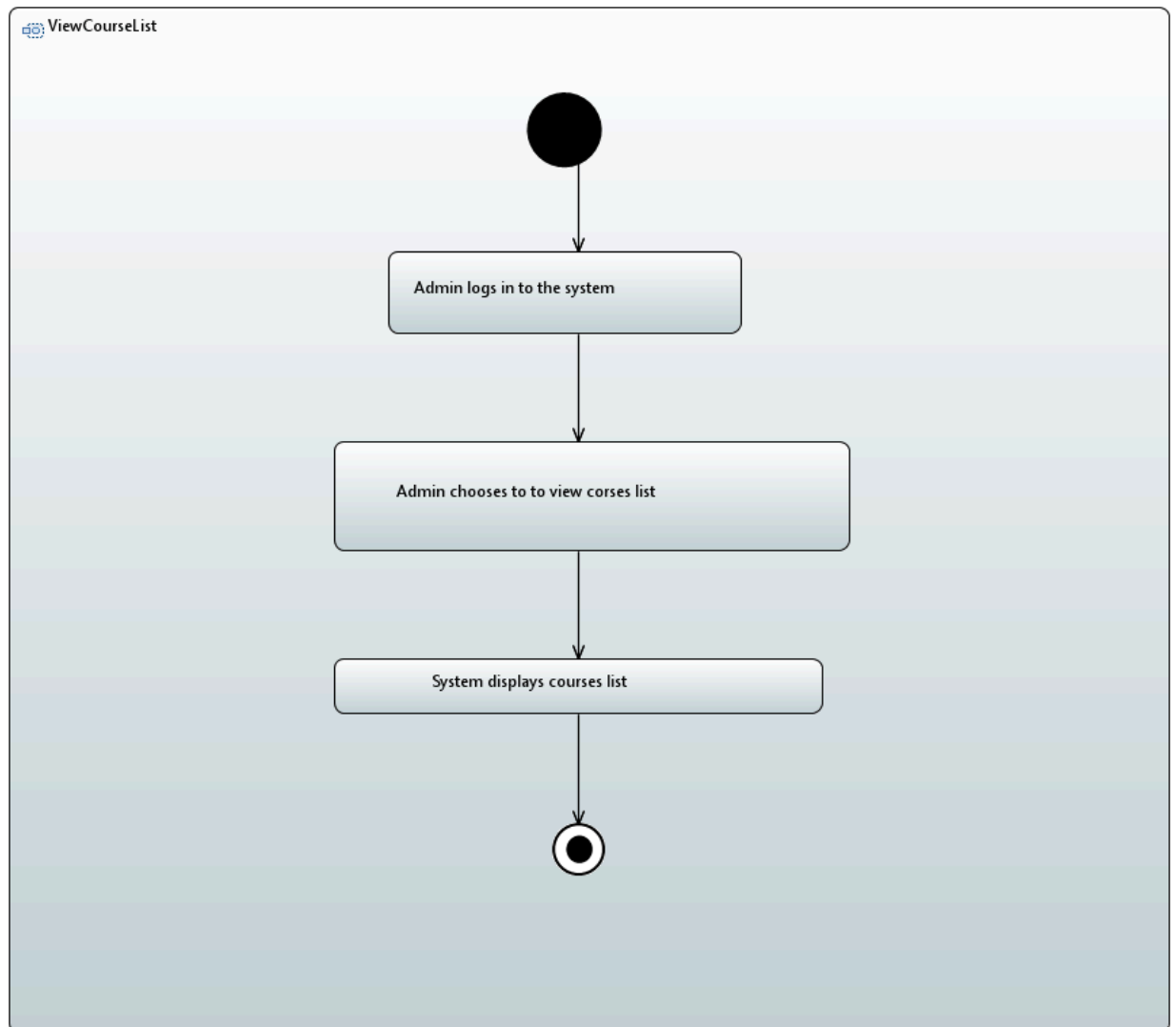
5.2.2.5 Edit Course (Admin)



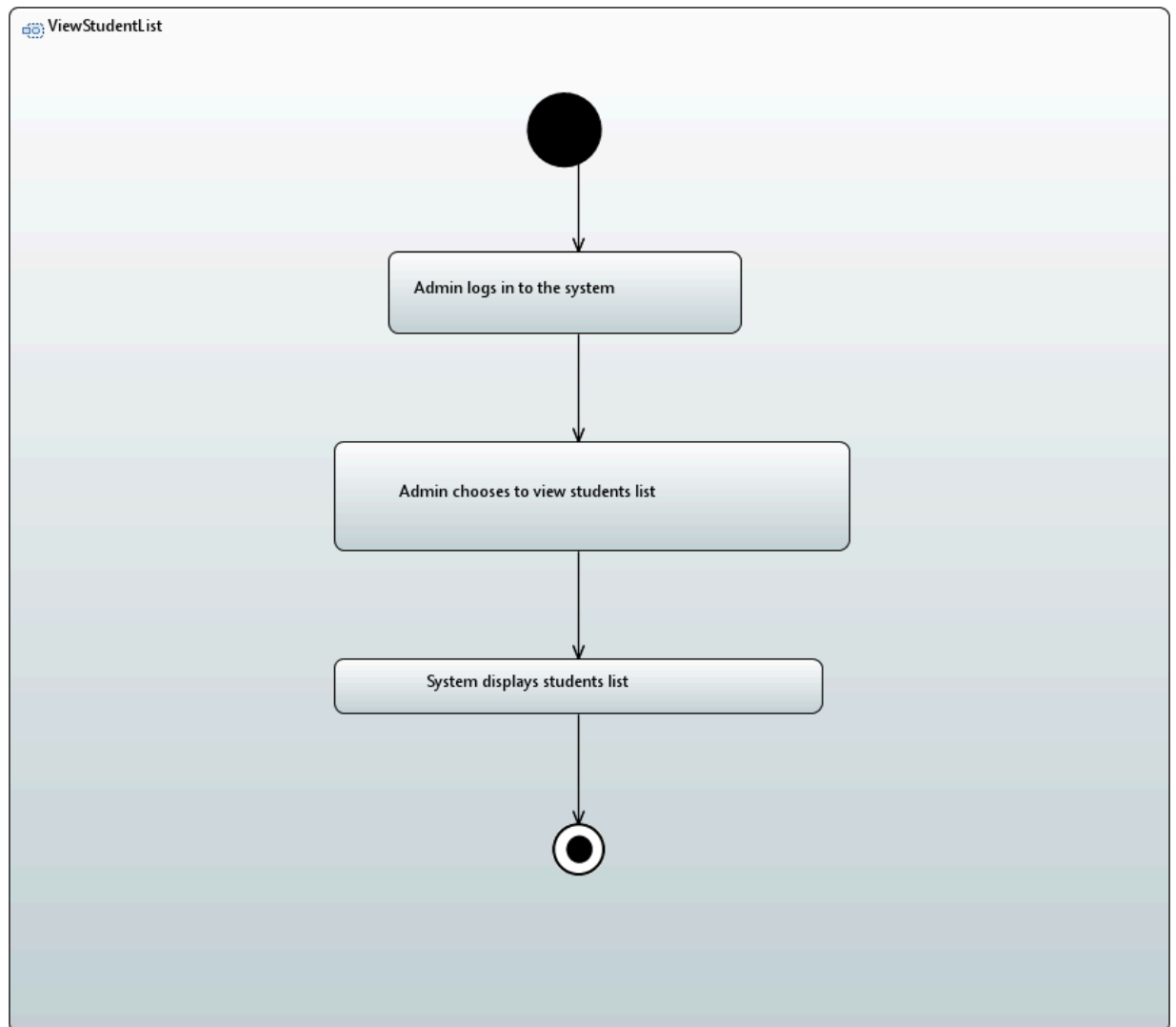
5.2.2.6 Assign Course to Teacher (Admin)



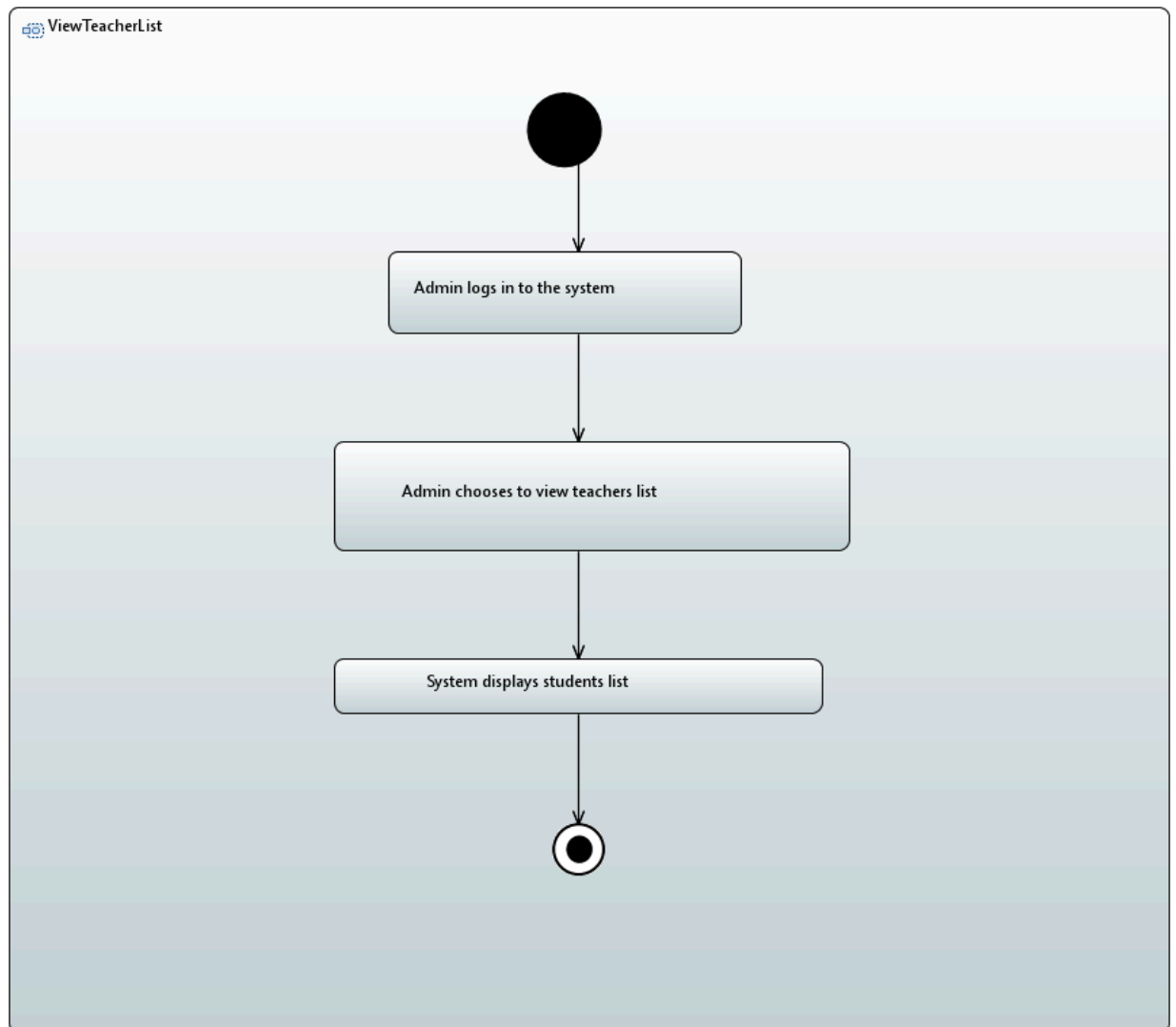
5.2.2.7 View Course List (Admin)



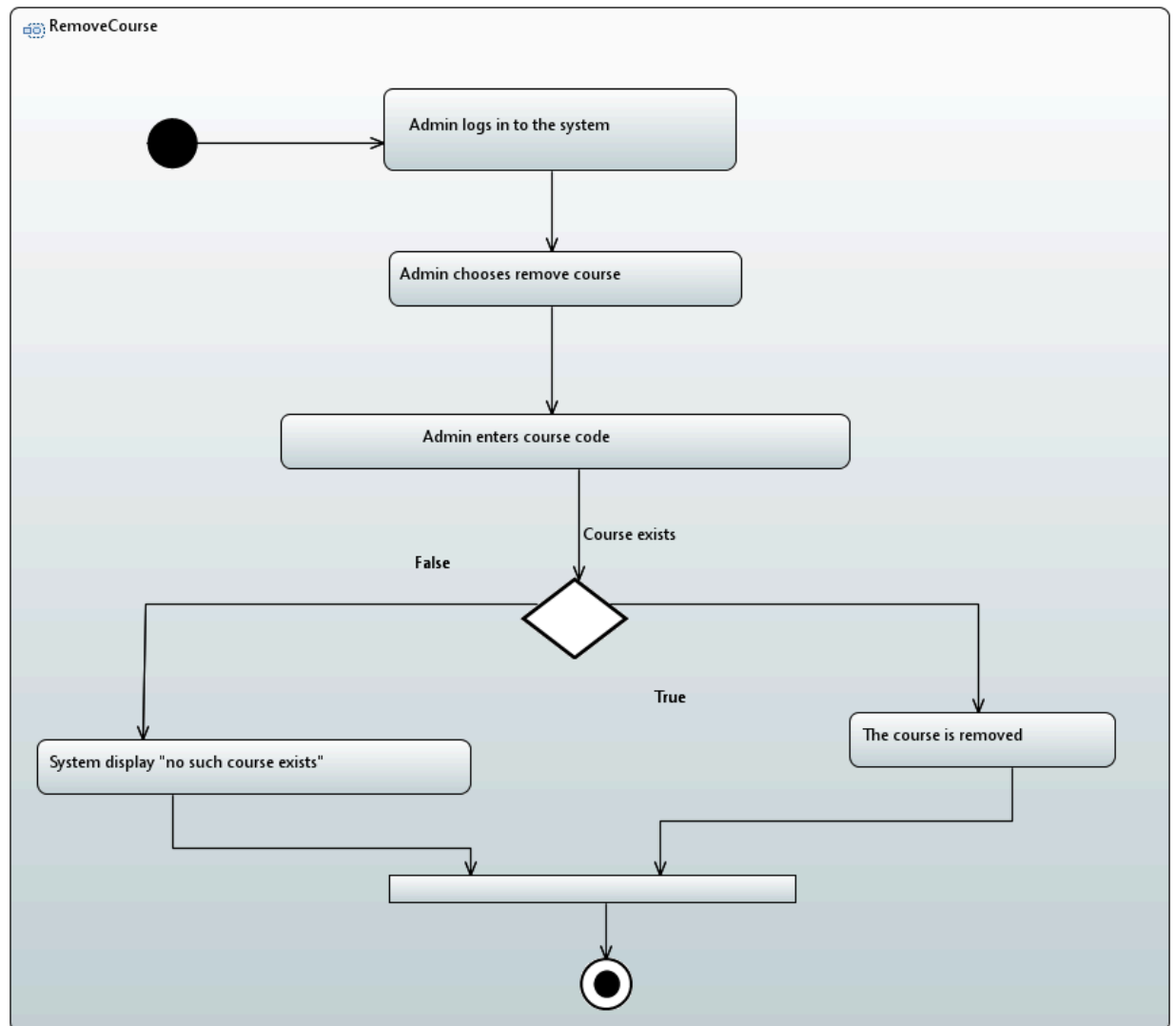
5.2.2.8 View Student List (Admin)



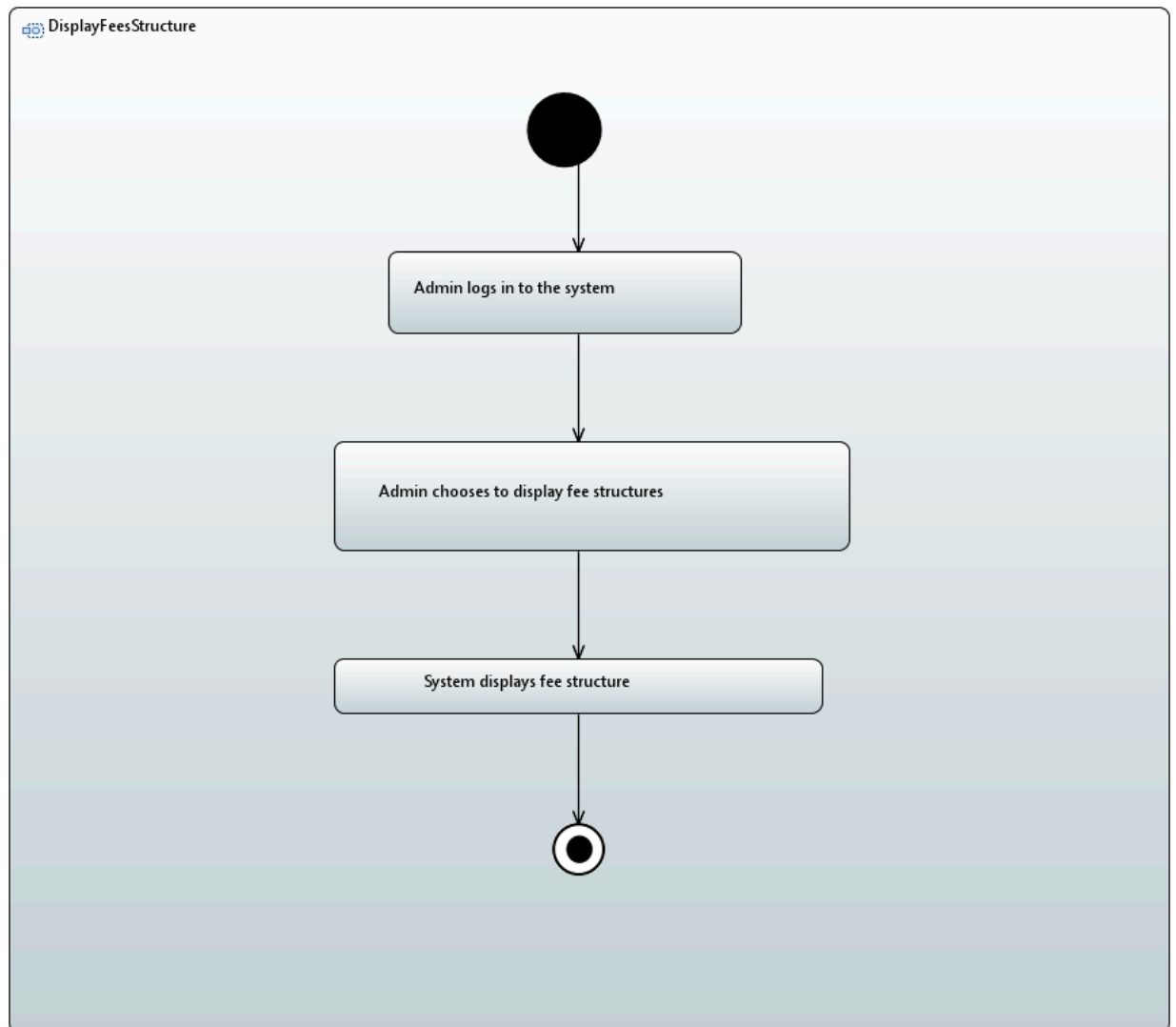
5.2.2.9 View Teacher List (Admin)



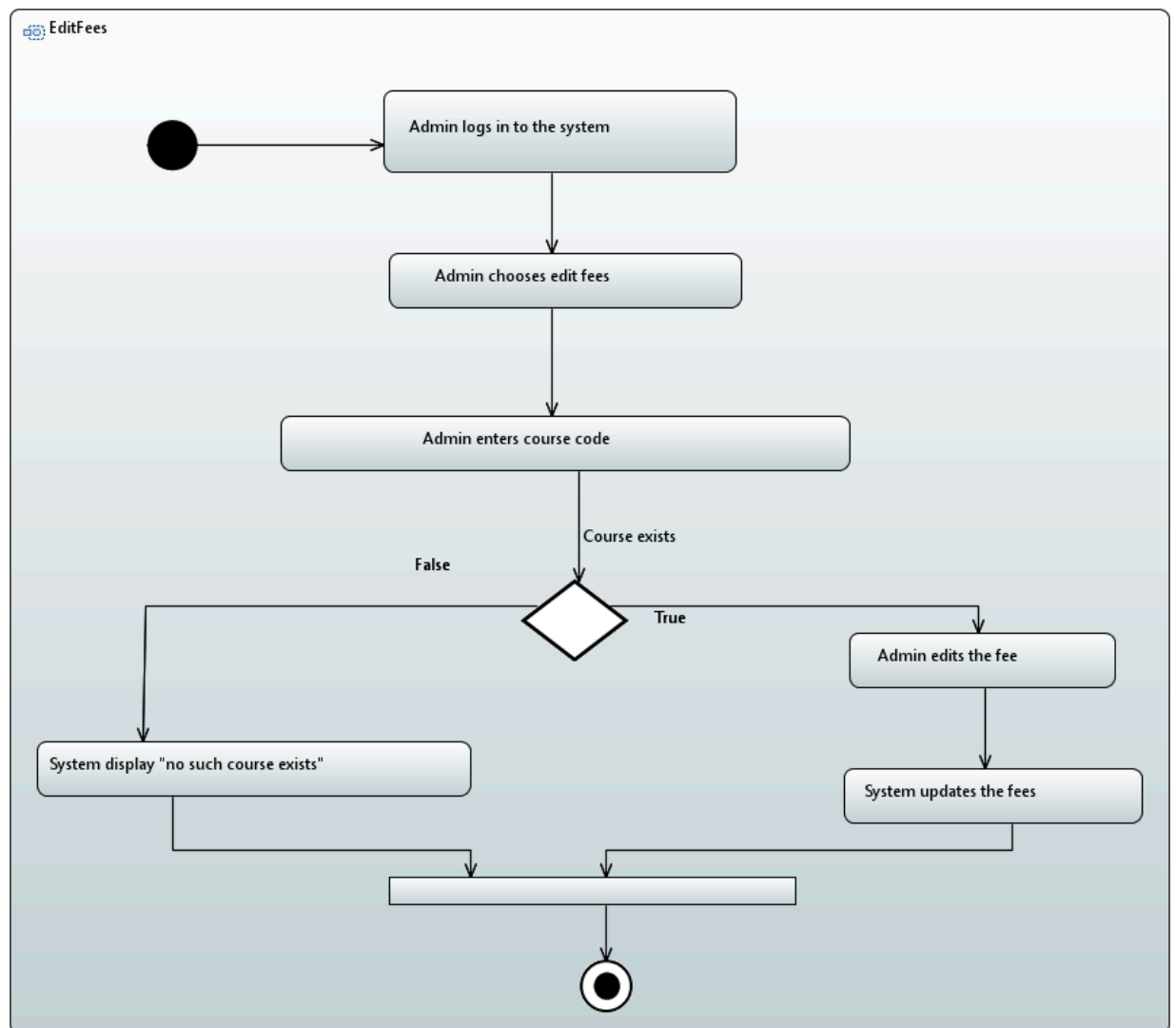
5.2.2.10 Remove Course (Admin)



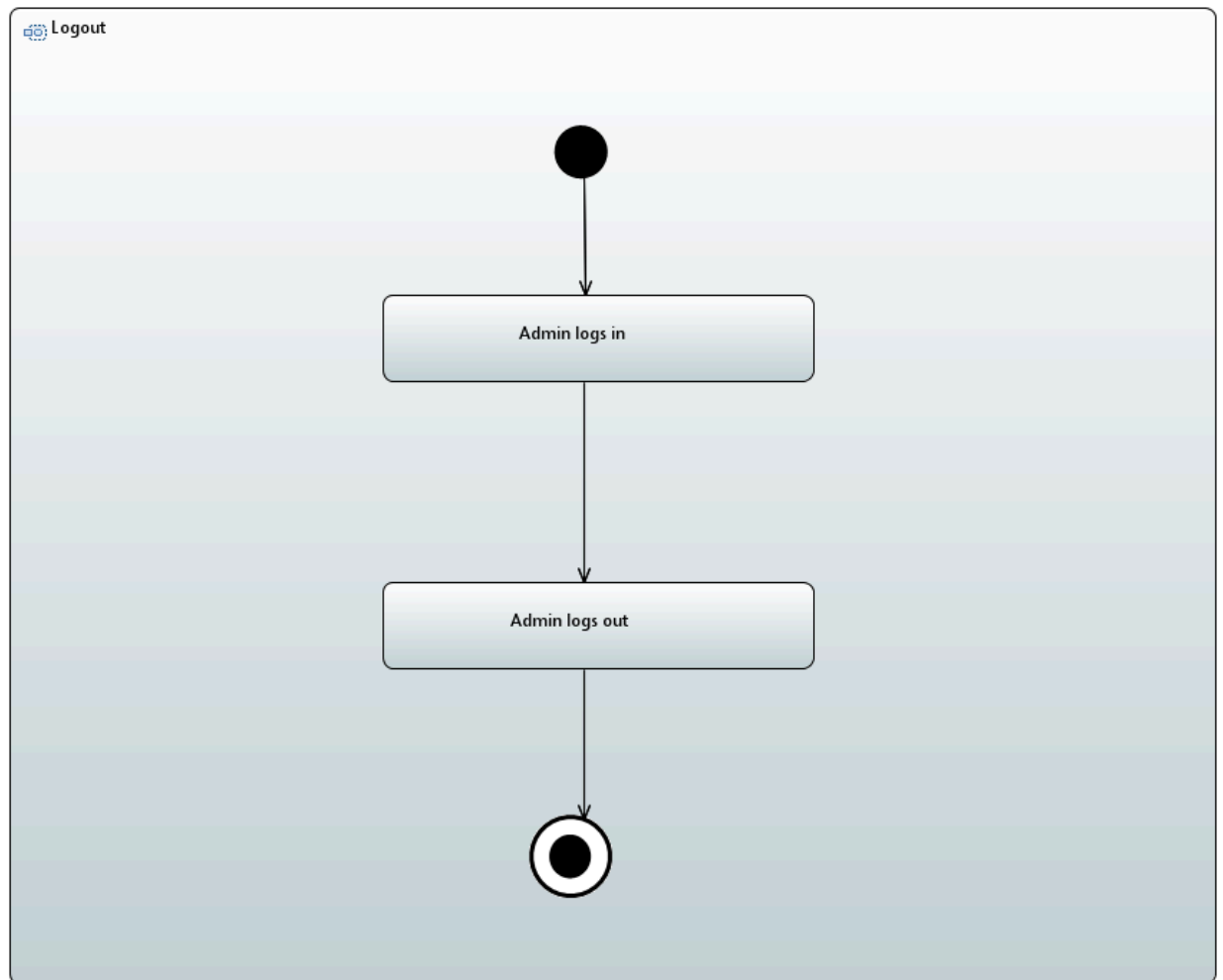
5.2.2.11 View Fee Structure



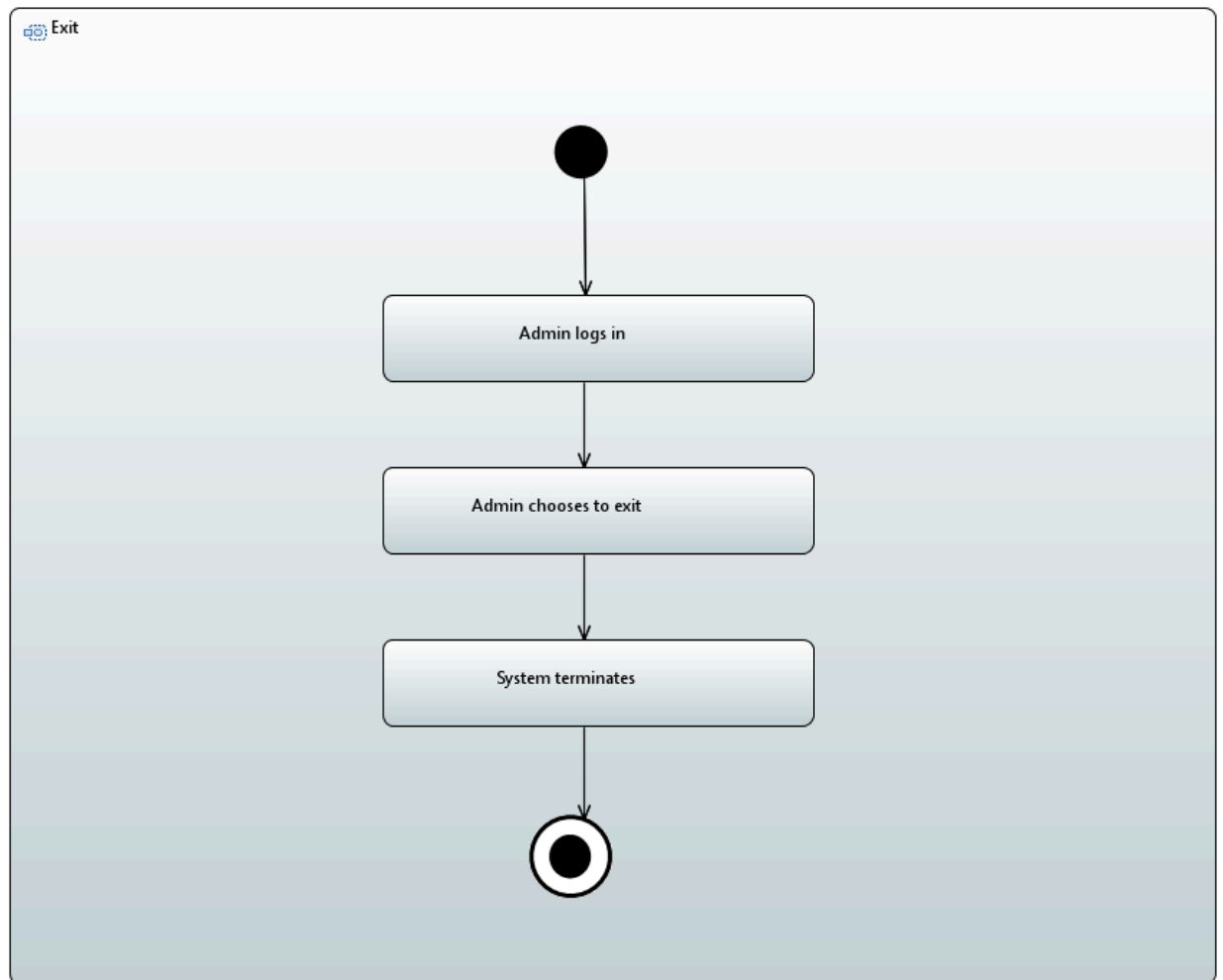
5.2.2.12 Edit Fees (Admin)



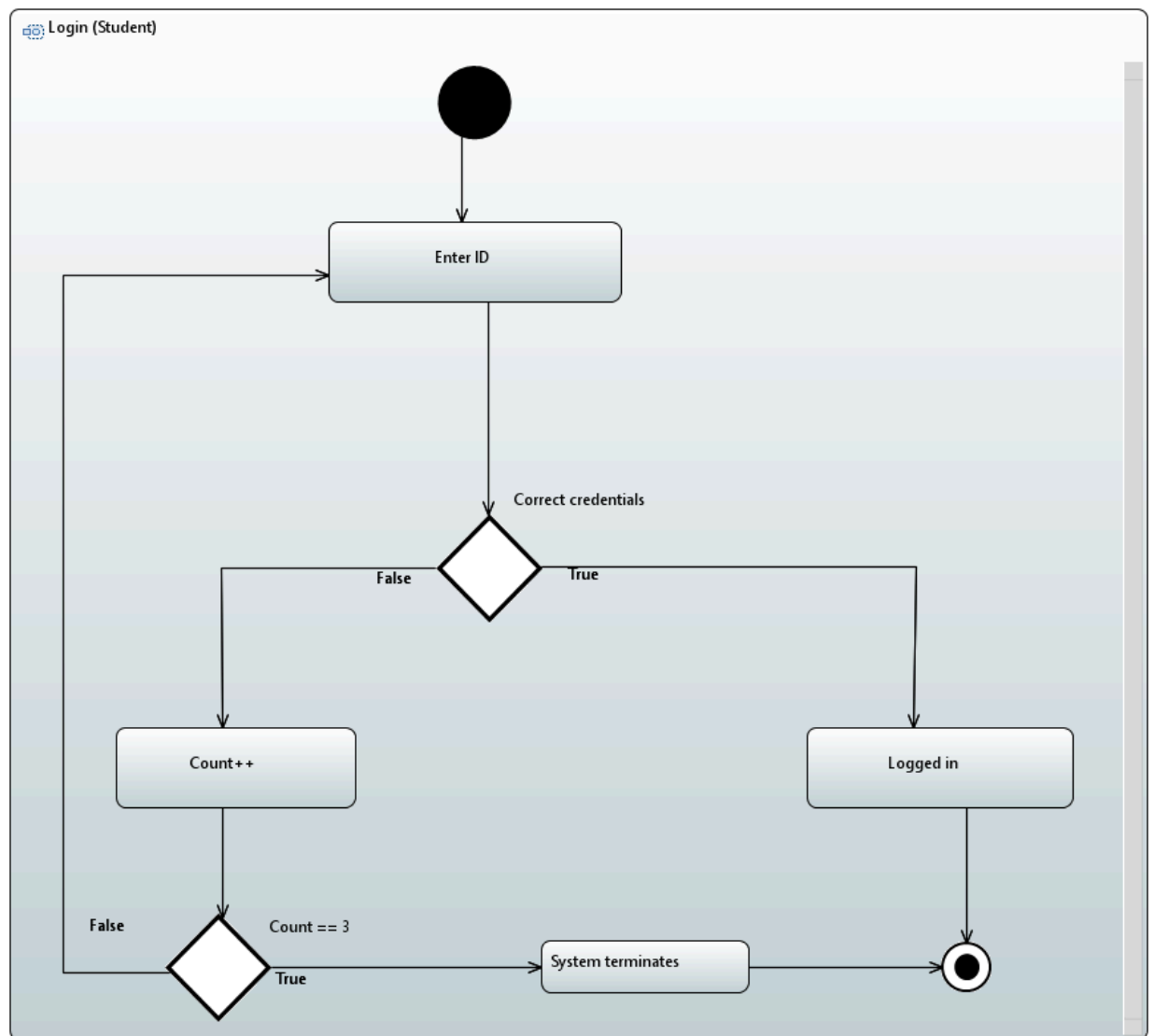
5.2.2.13 Log out (Admin)



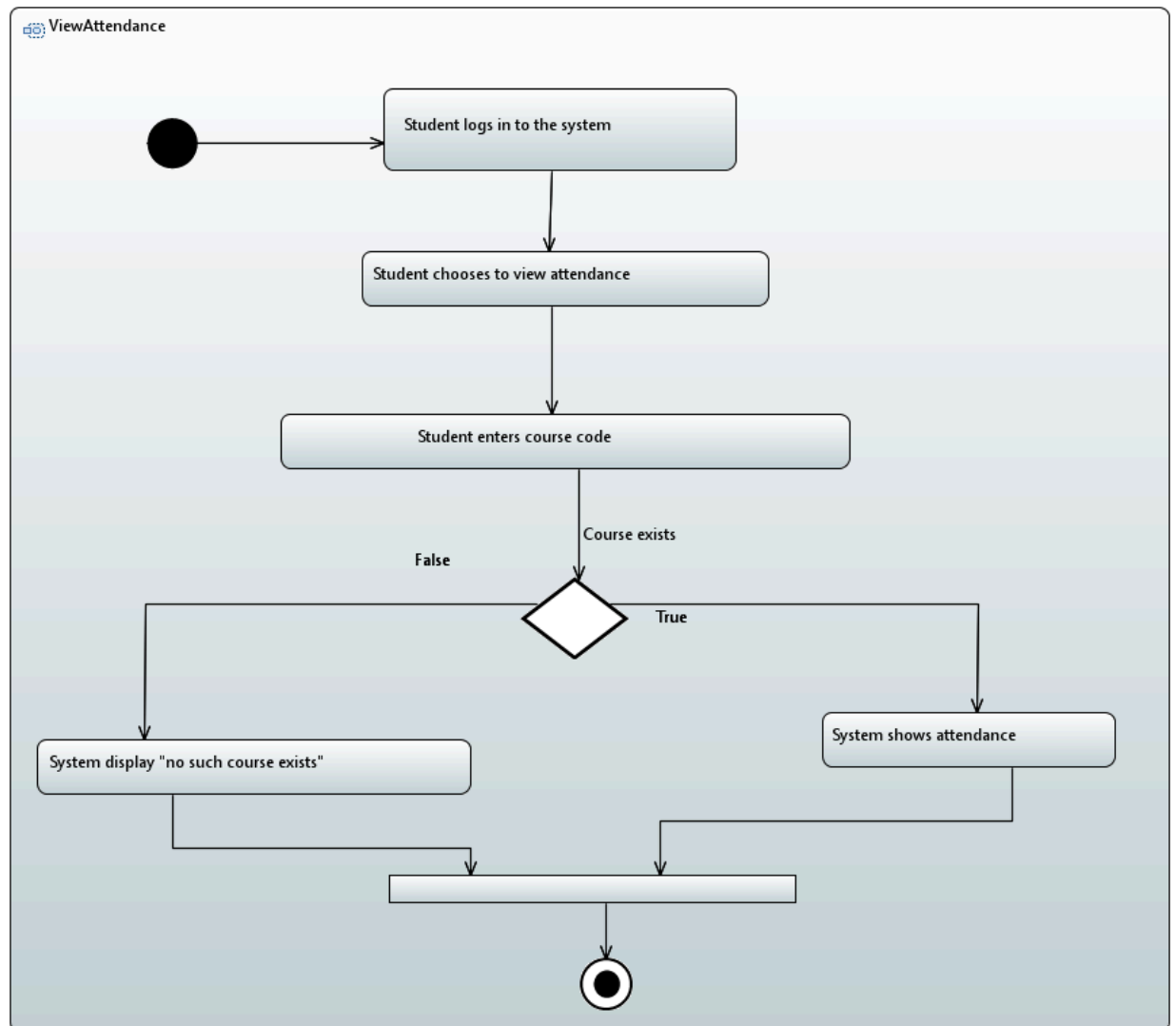
5.2.2.14 Exit (Admin)



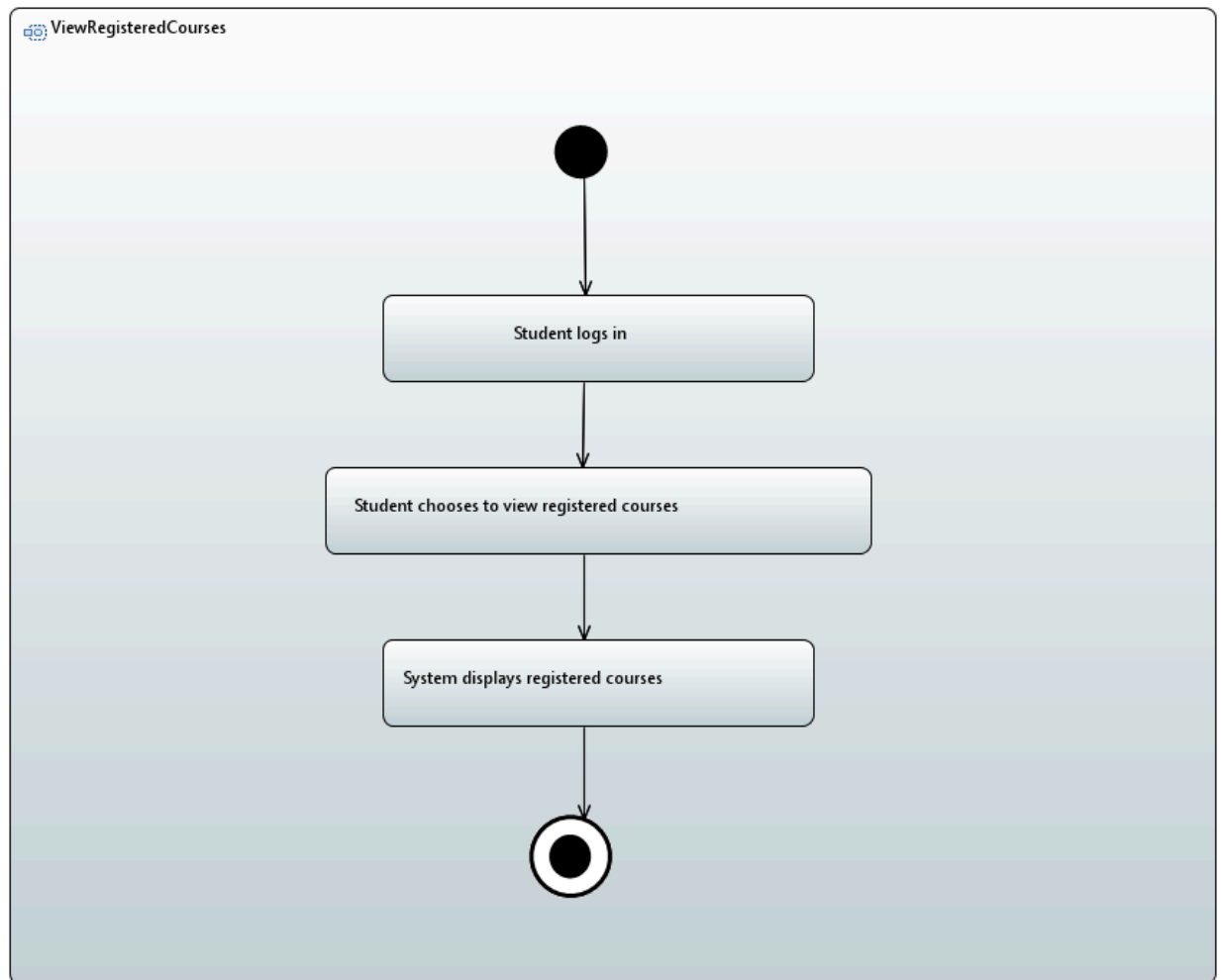
5.2.2.15 Login (Student)



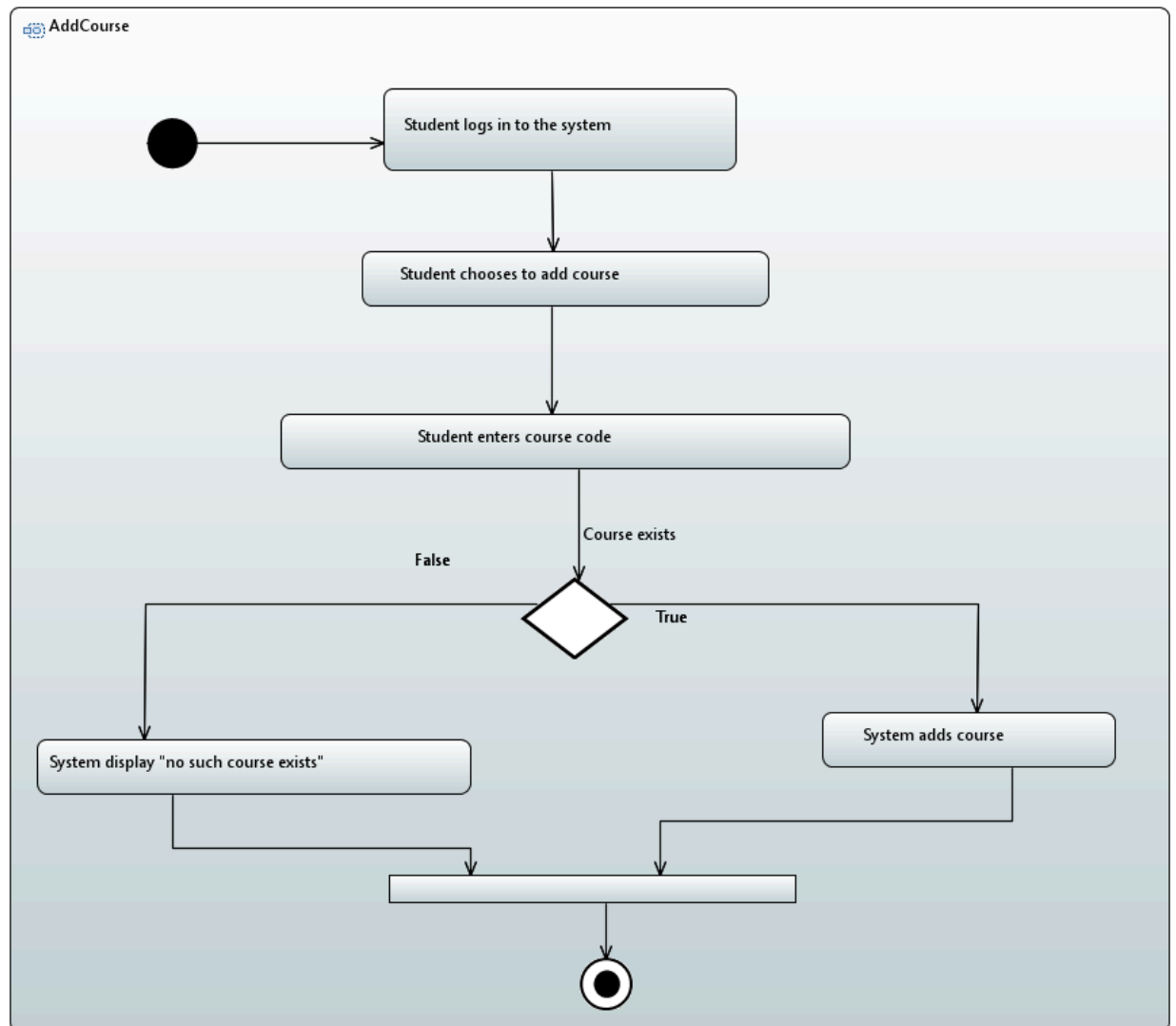
5.2.2.16 View Attendance (Student)



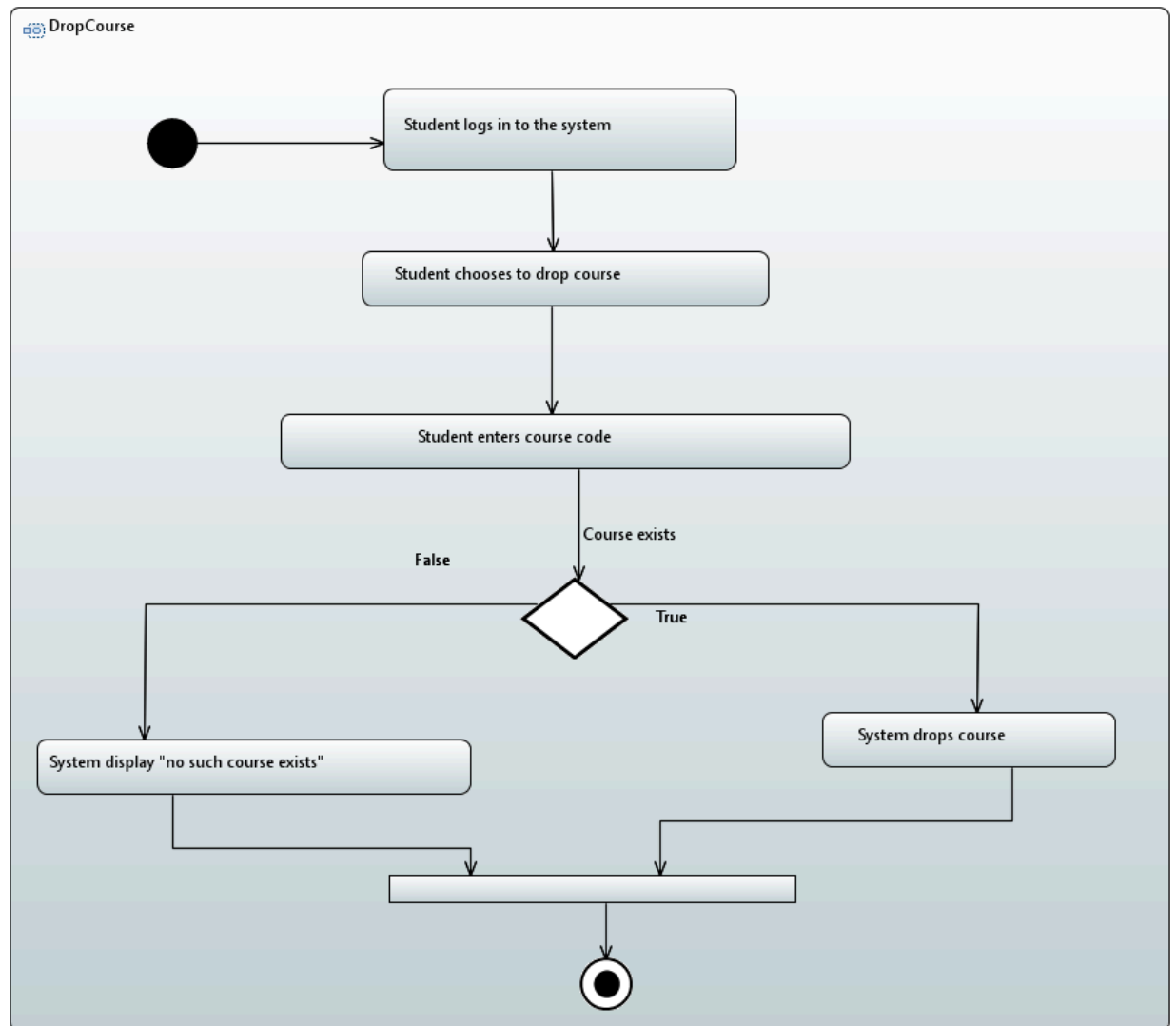
5.2.2.17 View Registered Courses (Student)



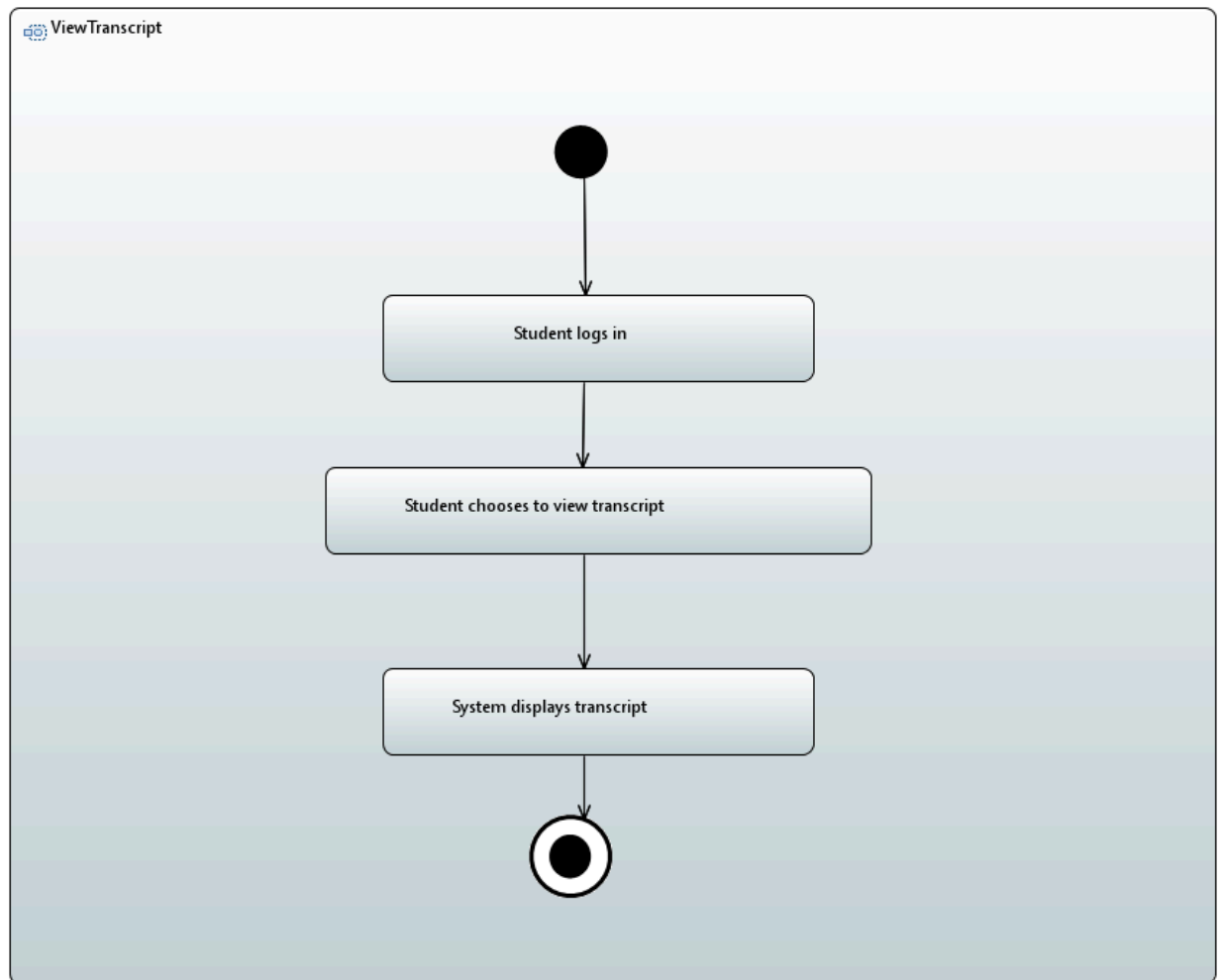
5.2.2.18 Add Course (Student)



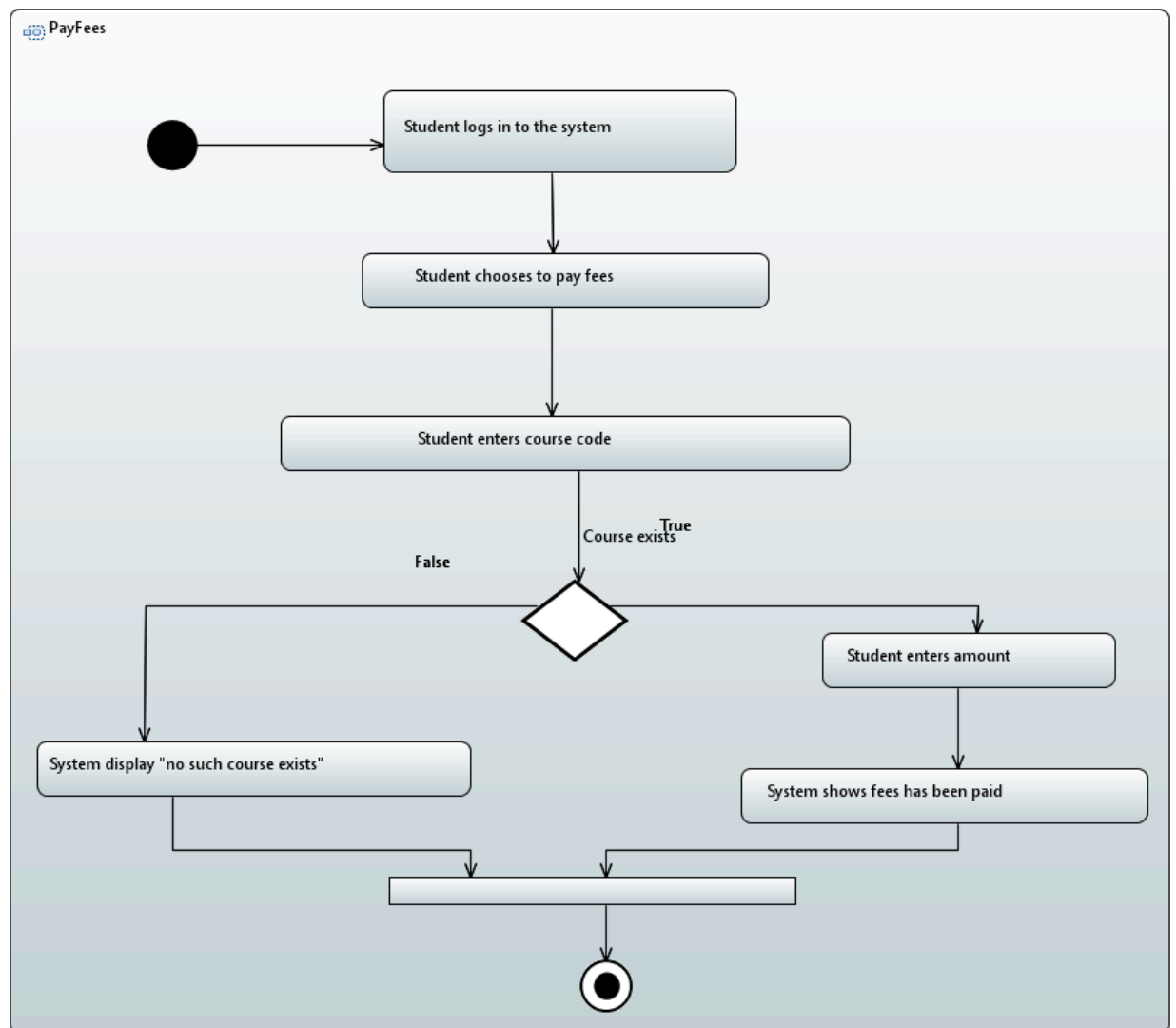
5.2.2.19 Drop Course (Student)



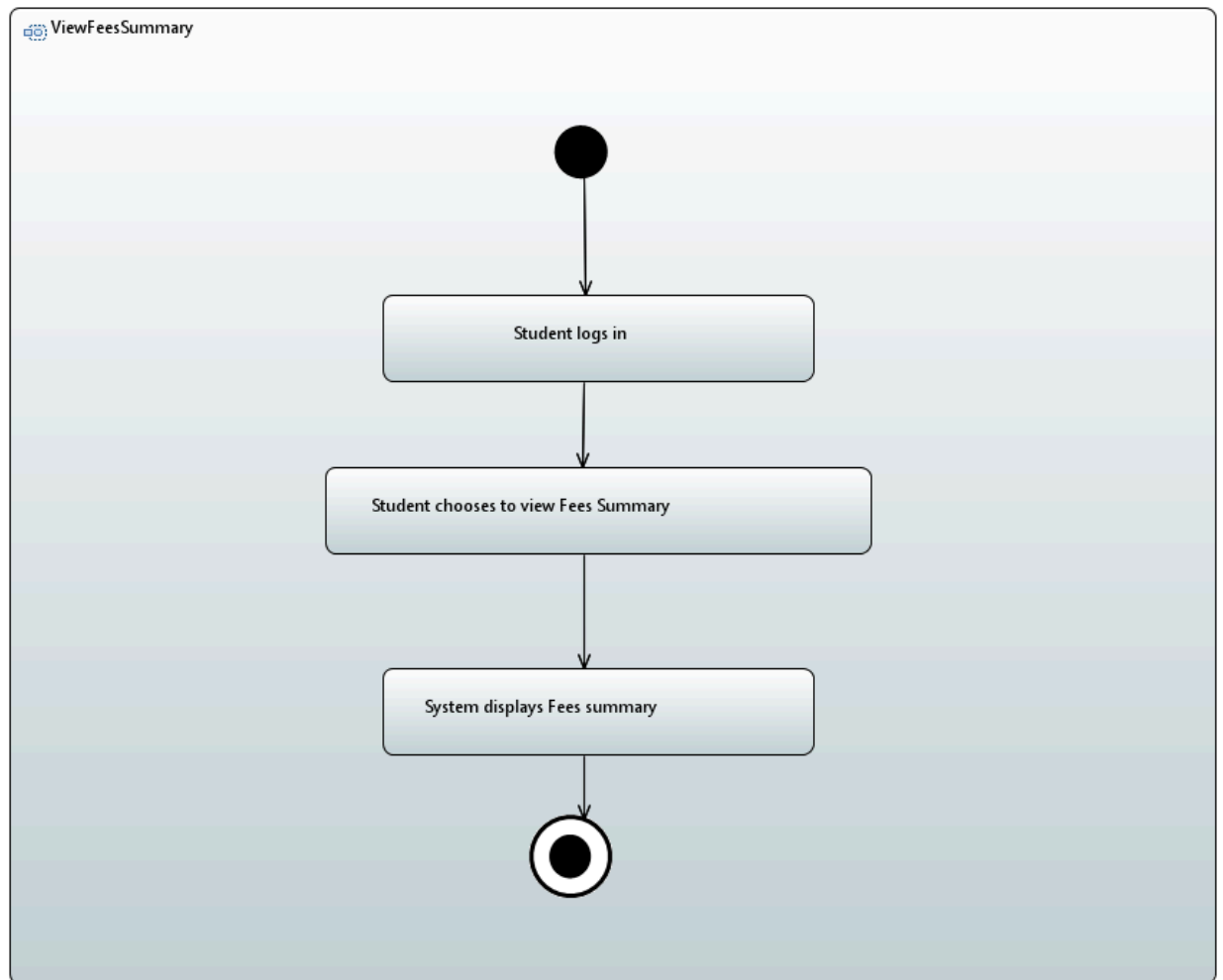
5.2.2.20 Transcript (Student)



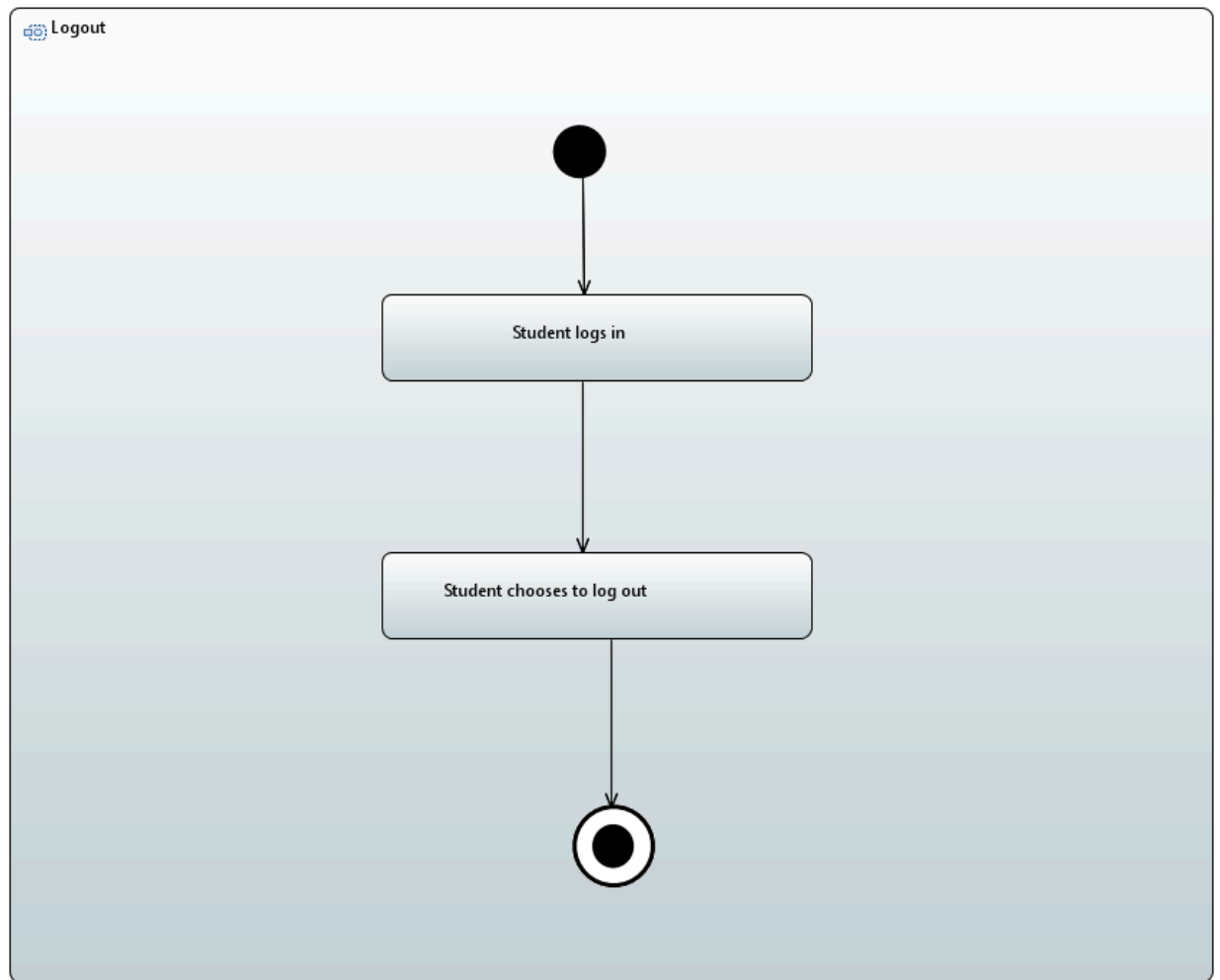
5.2.2.21 Pay Fees (Student)



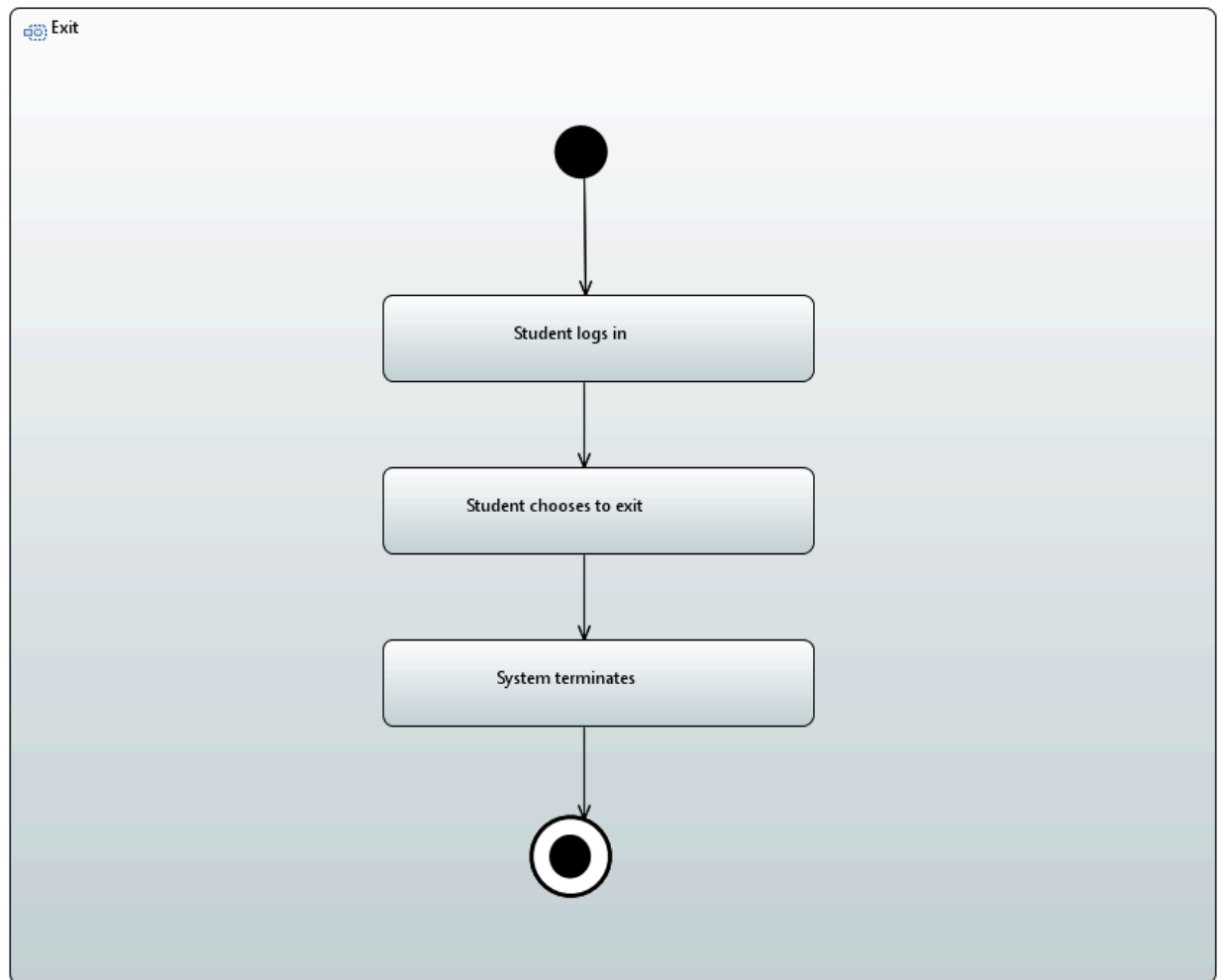
5.2.2.22 Fees Summary (Student)



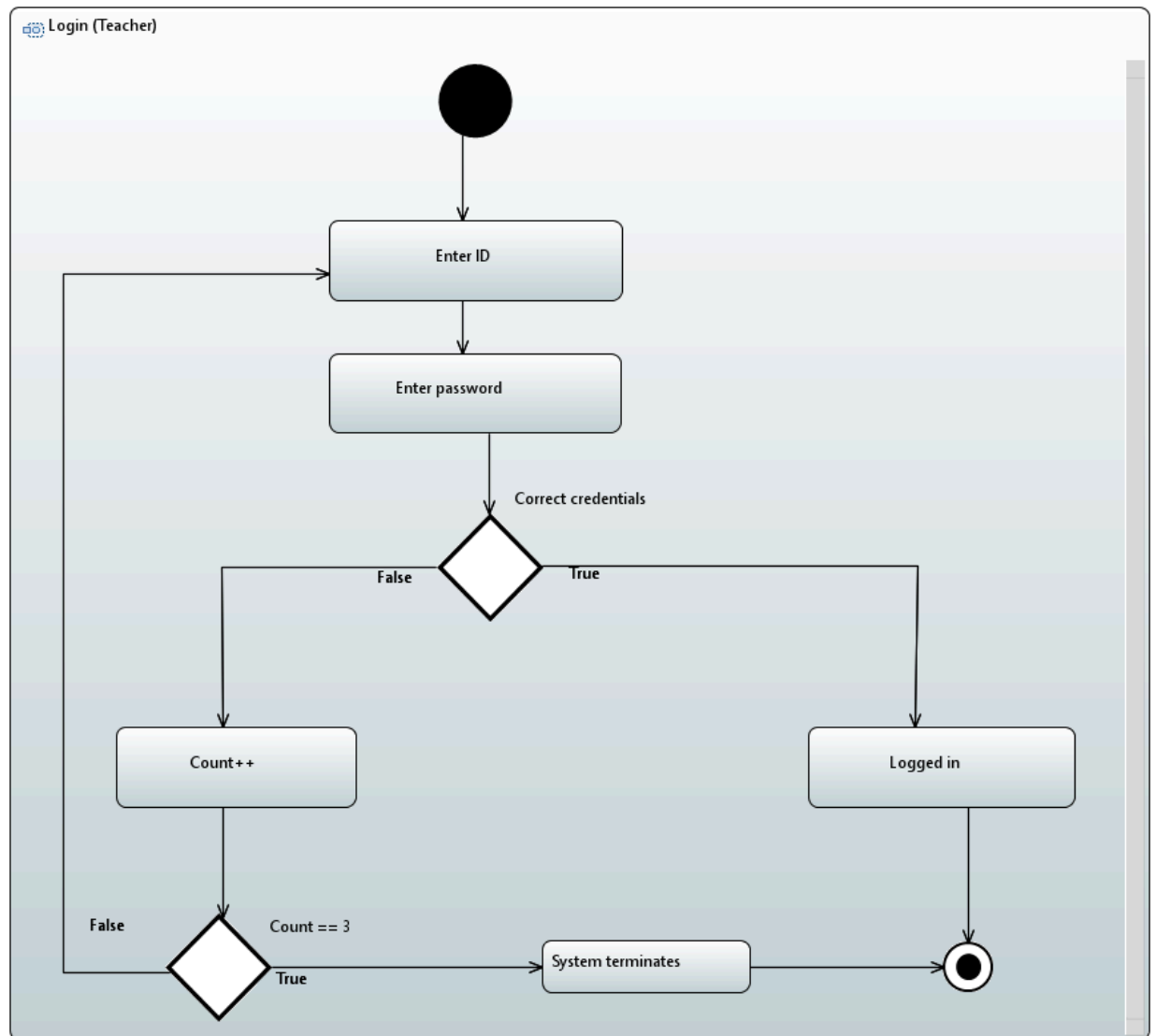
5.2.2.23 Log out (Student)



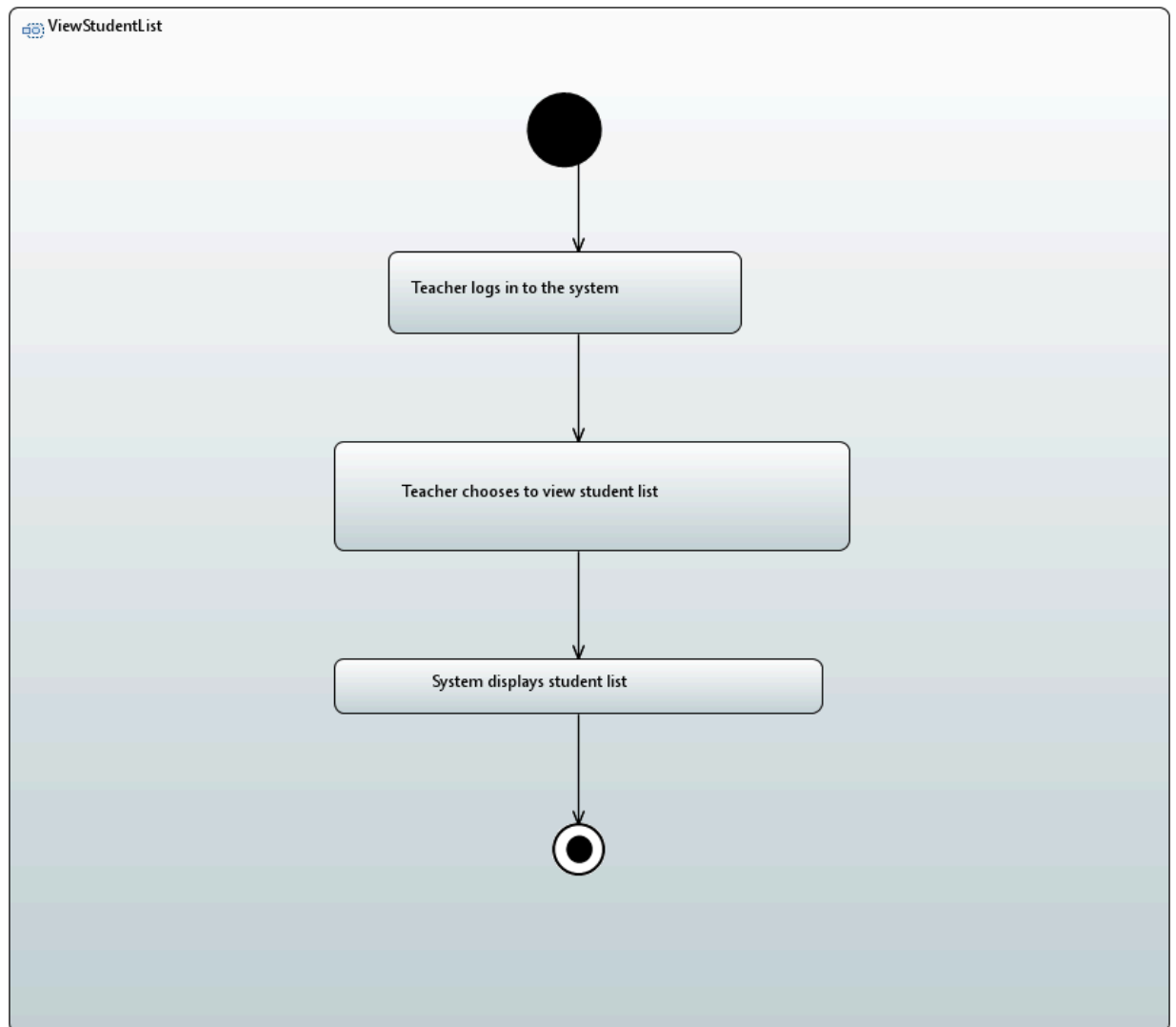
5.2.2.24 Exit (Student)



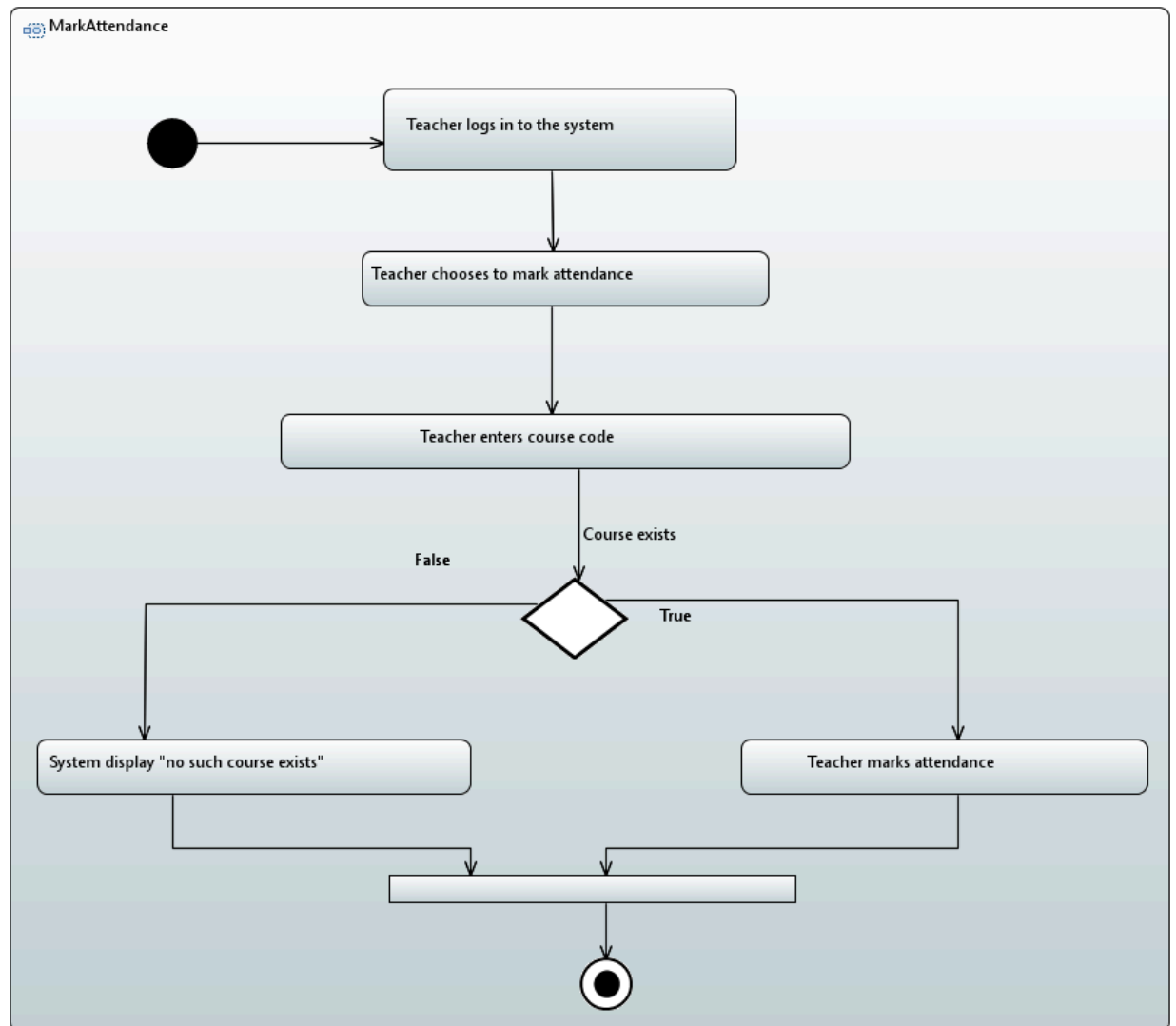
5.2.2.25 Login (Teacher)



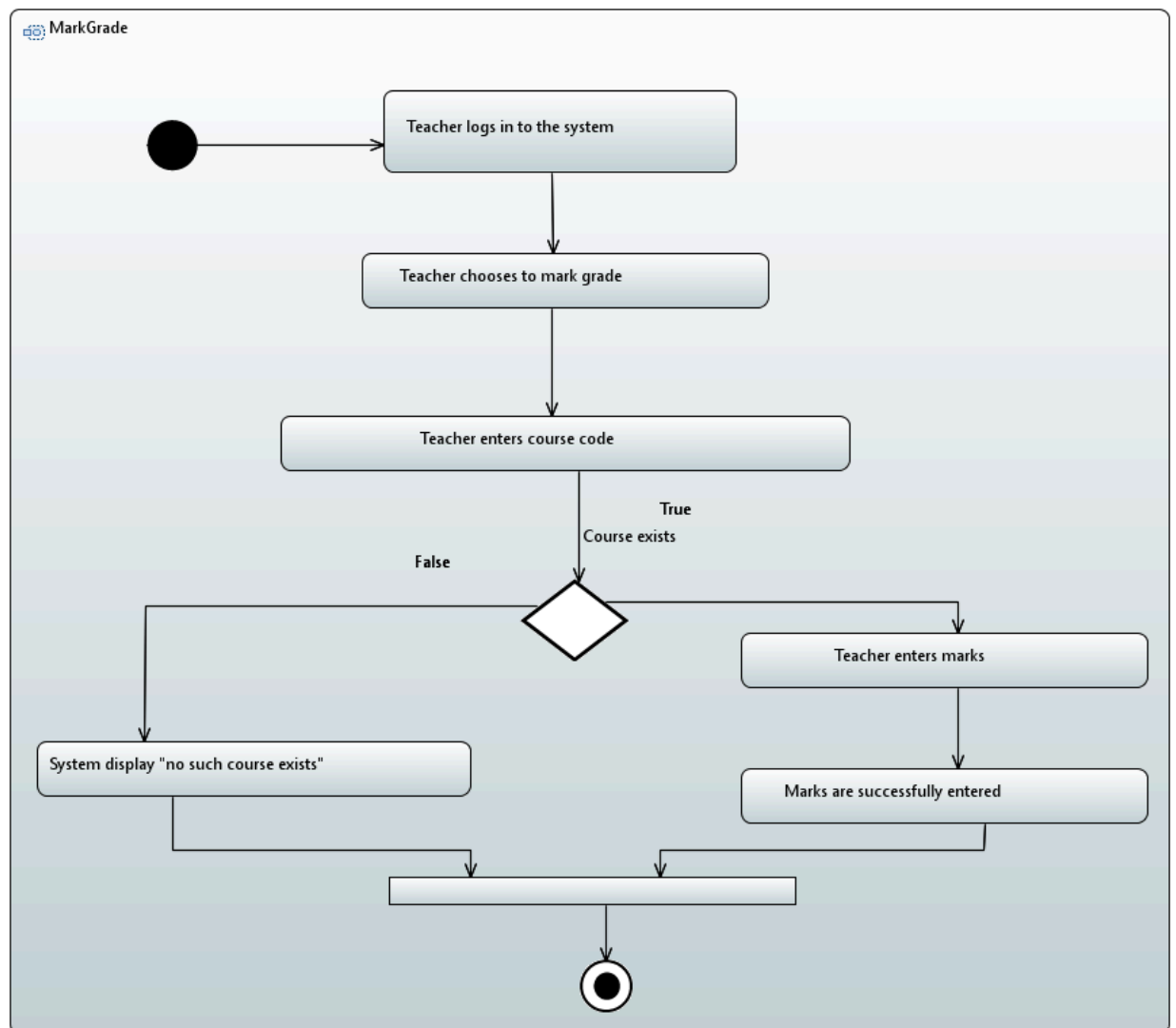
5.2.2.26 View Students List (Teacher)



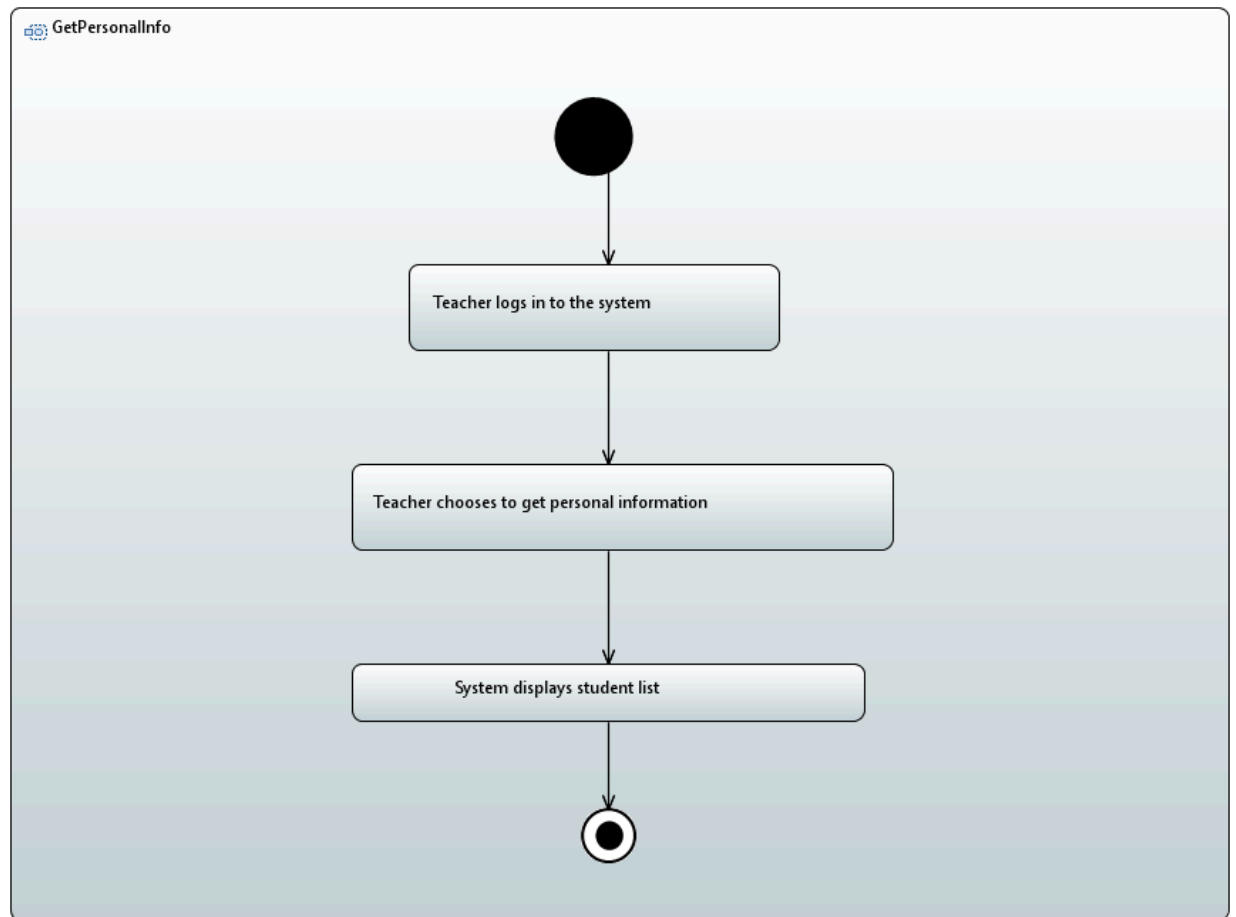
5.2.2.27 Mark Attendance (Teacher)



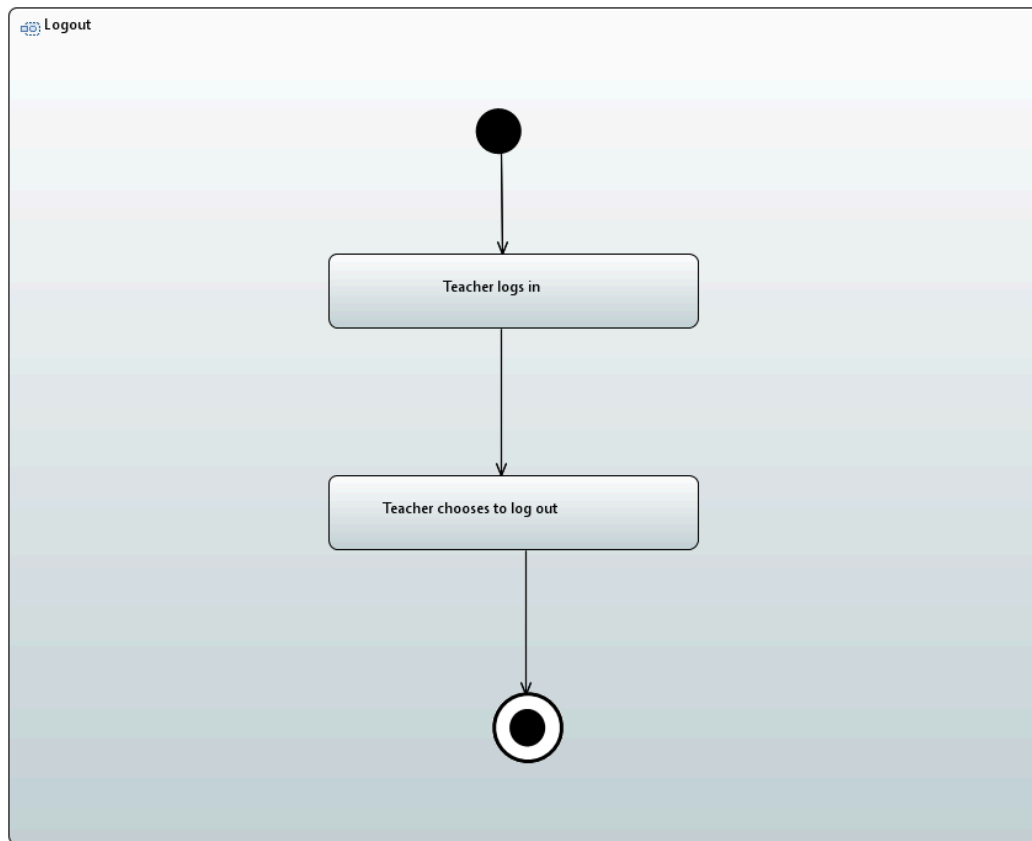
5.2.2.28 Mark Grade (Teacher)



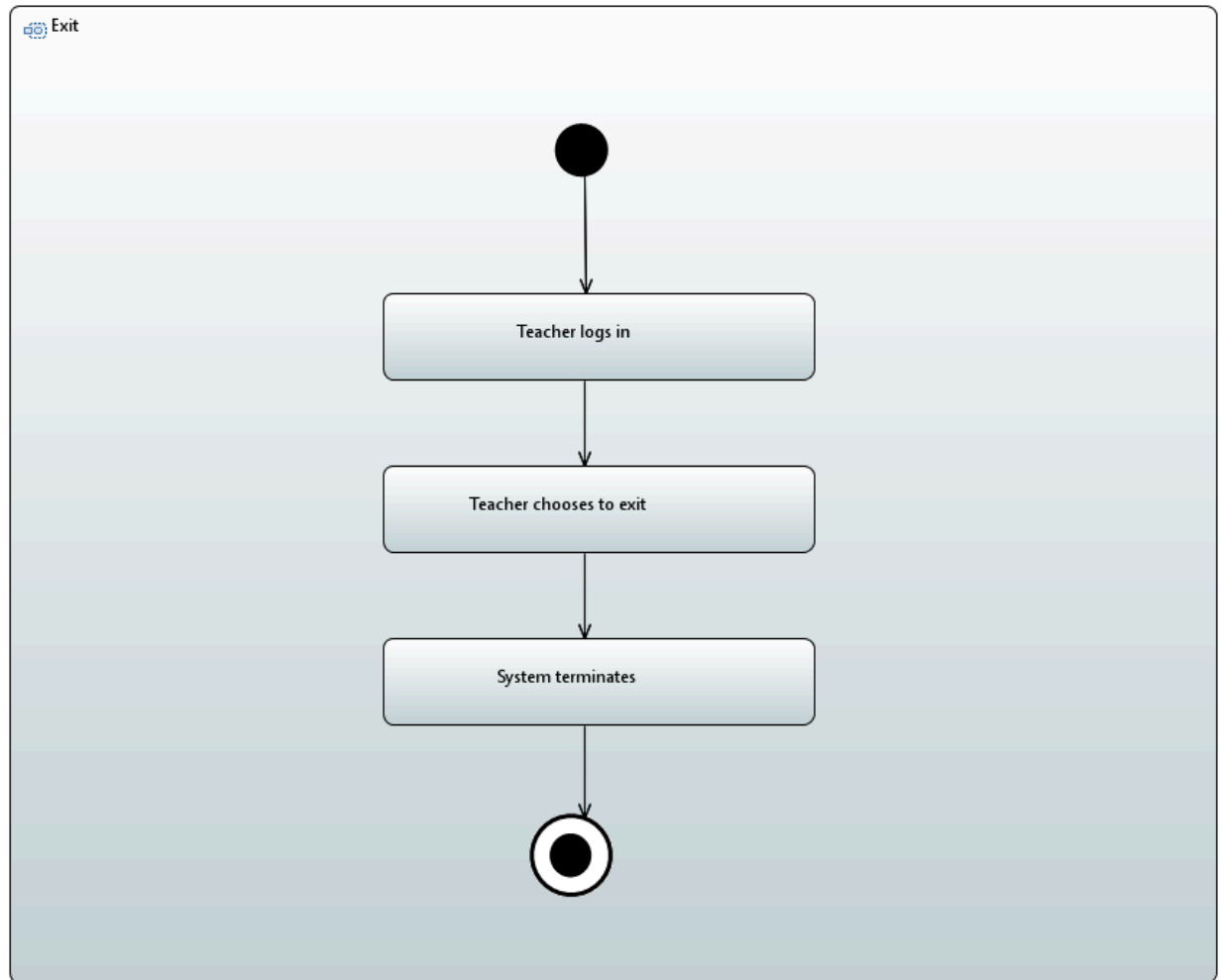
5.2.2.29 Get Personal Information (Teacher)



5.2.2.30 Log out (Teacher)



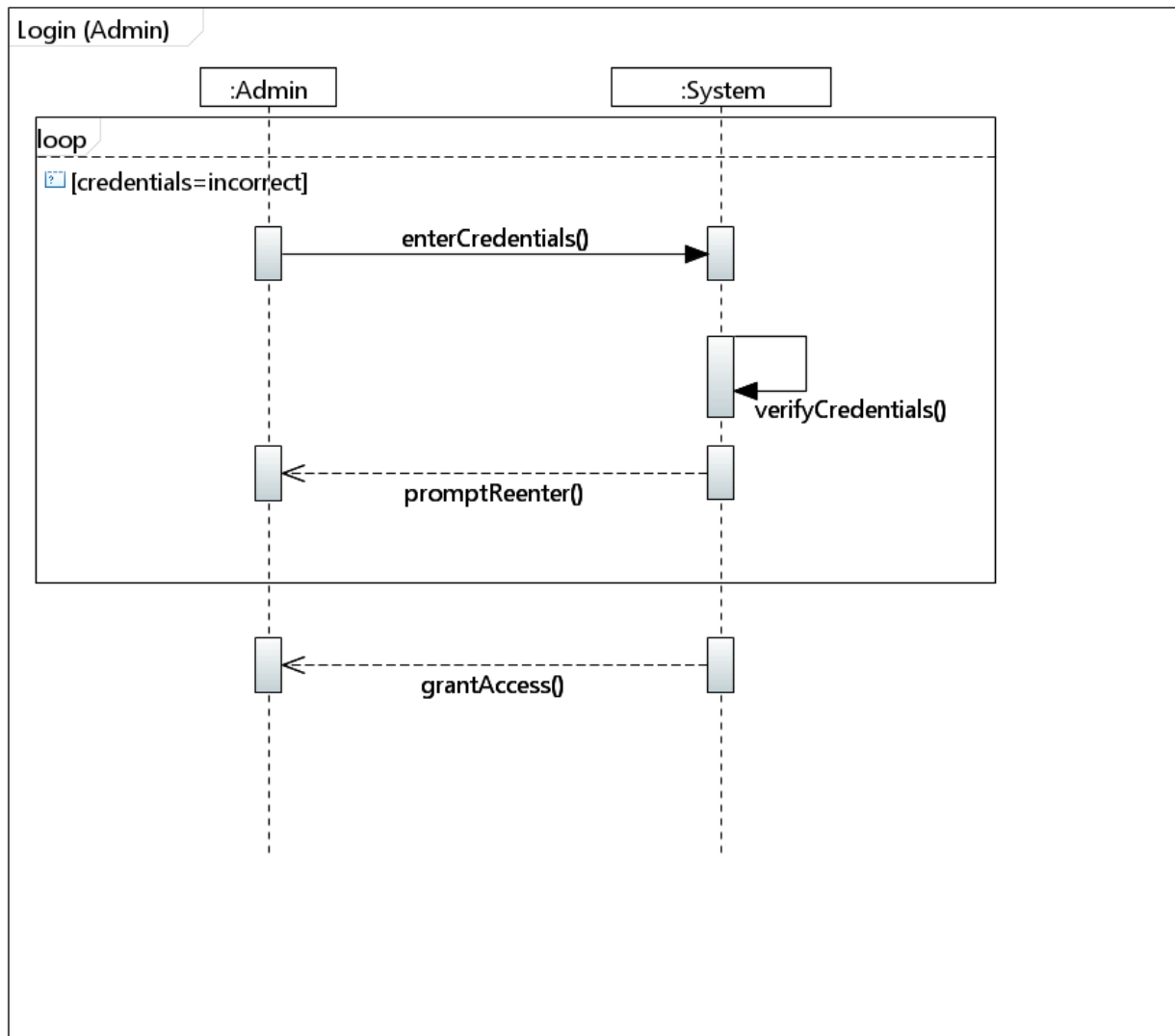
5.2.2.31 Exit (Teacher)



5.2.3 Sequence Diagram

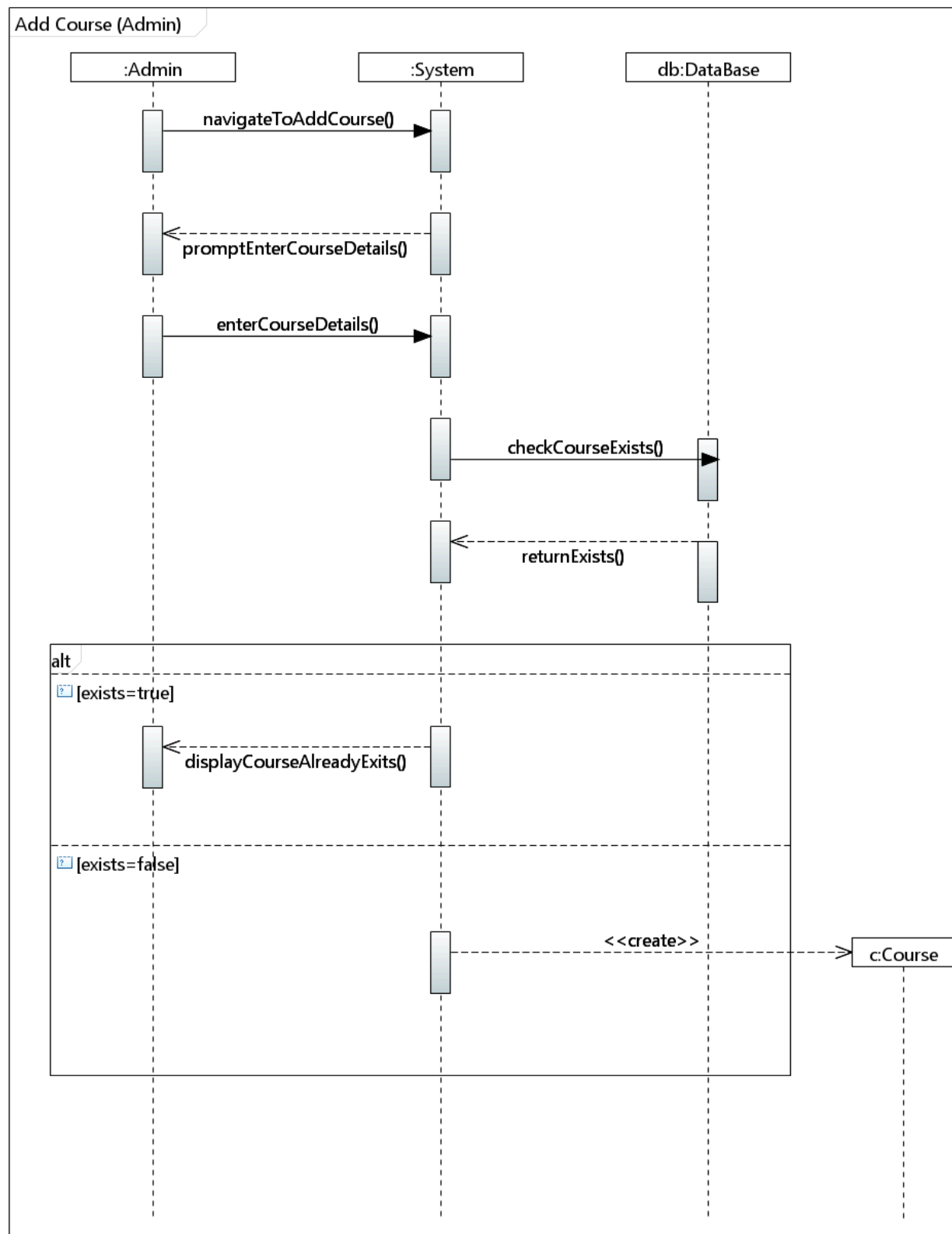
5.2.3.1 Login (Admin)

SMS-001: Log in



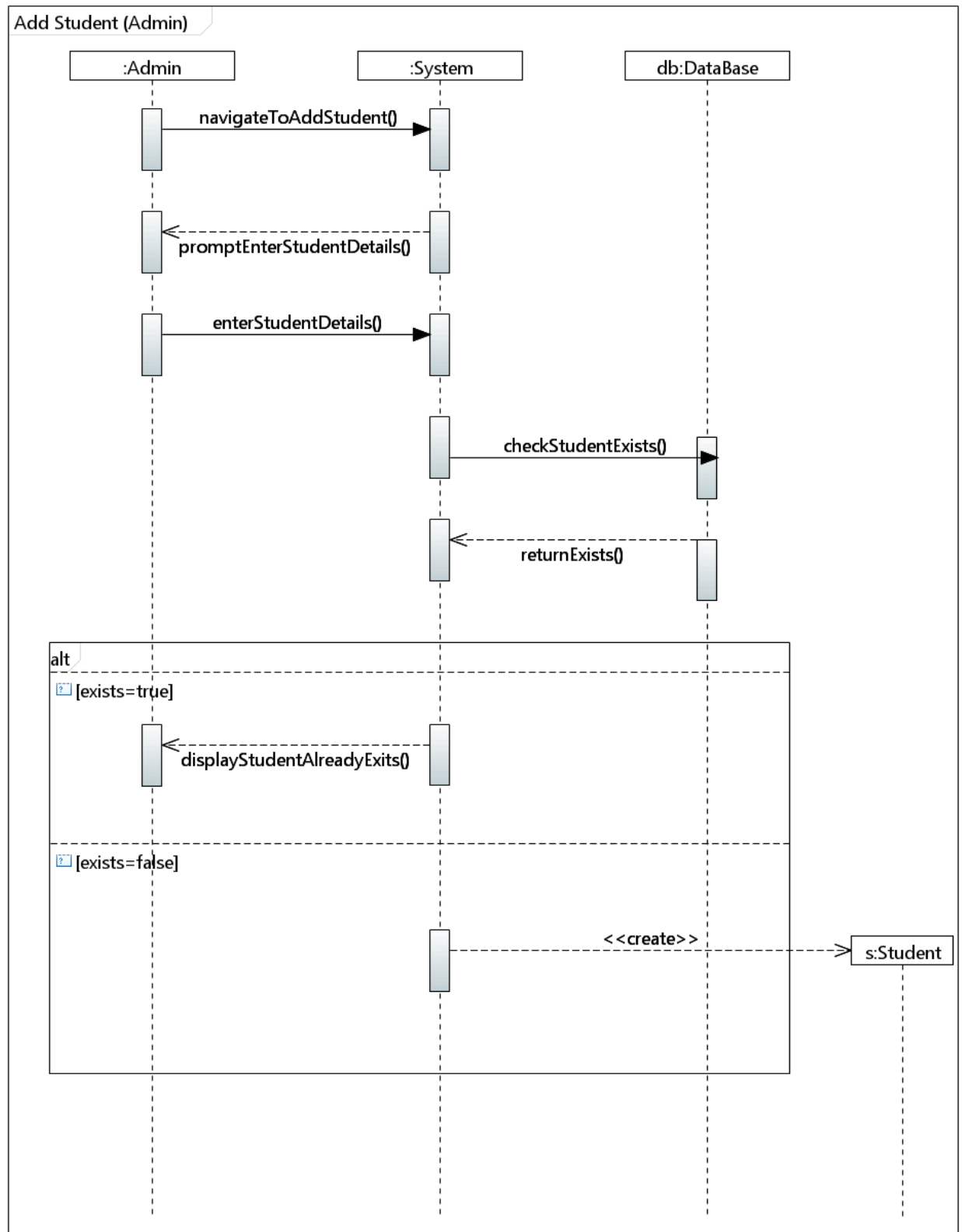
5.2.3.2 Add Course (Admin)

SMS-002: Add Course



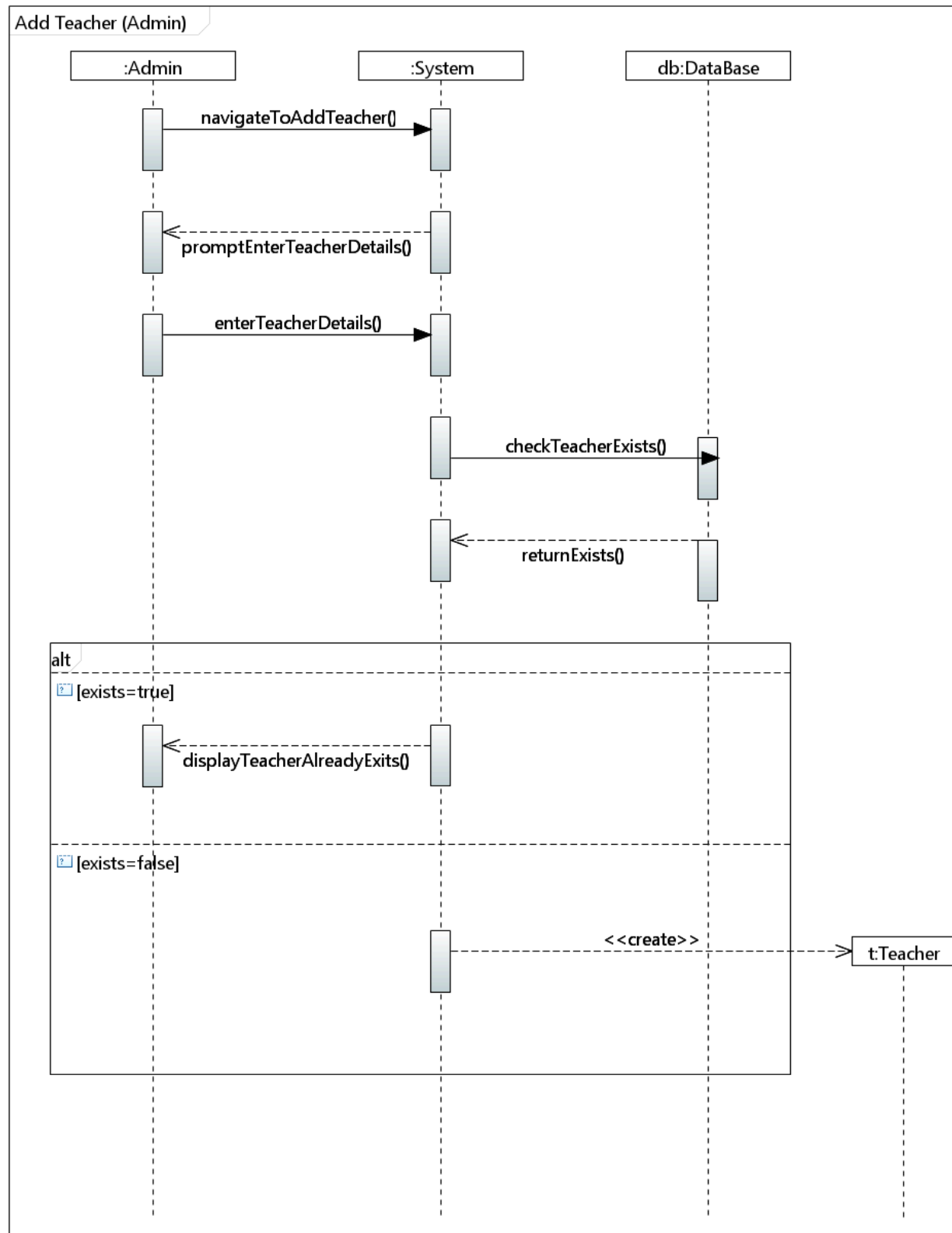
5.2.3.3 Add Student (Admin)

SMS-003: Add Student



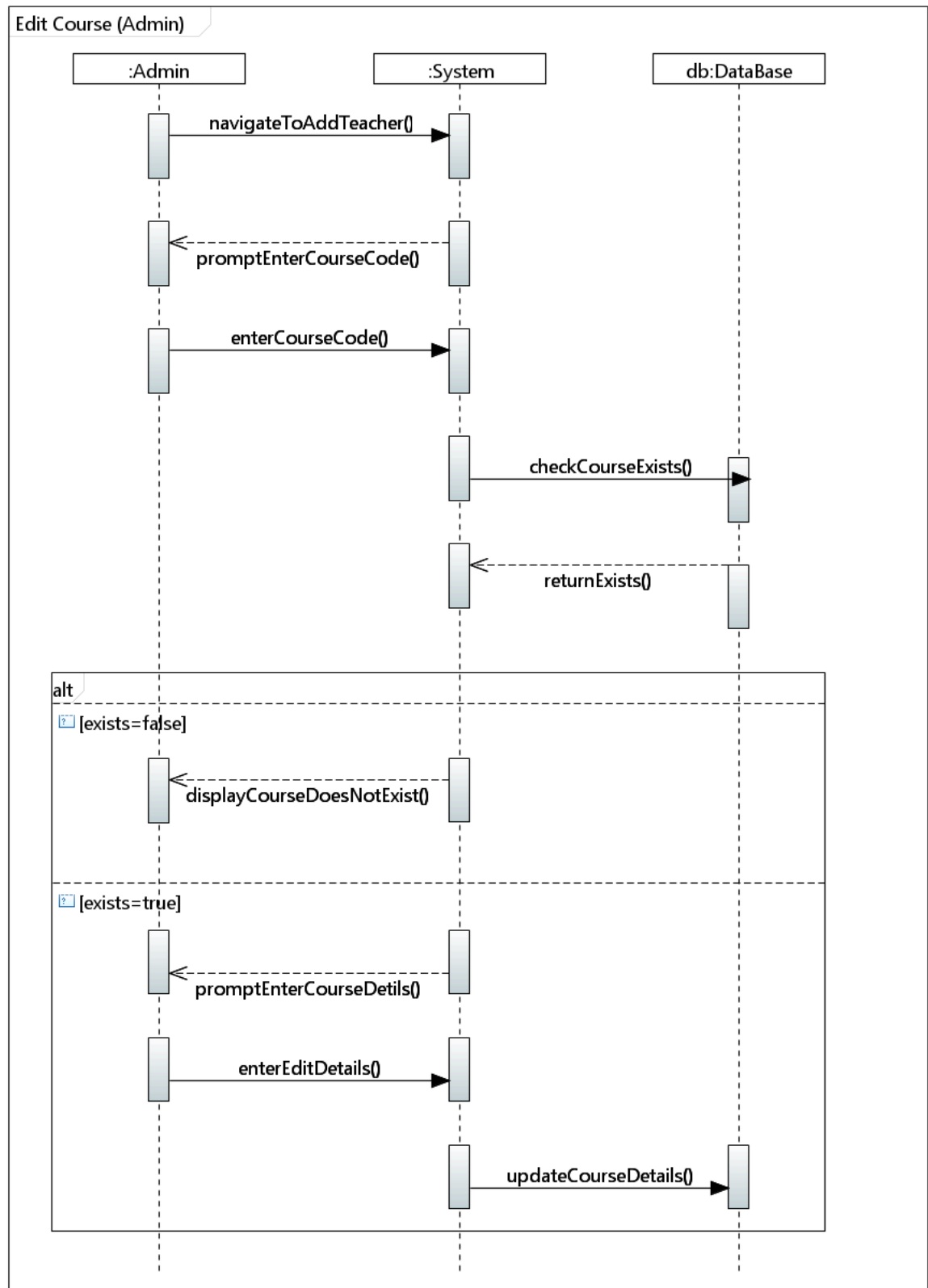
5.2.3.4 Add Teacher (Admin)

SMS-004: Add Teacher



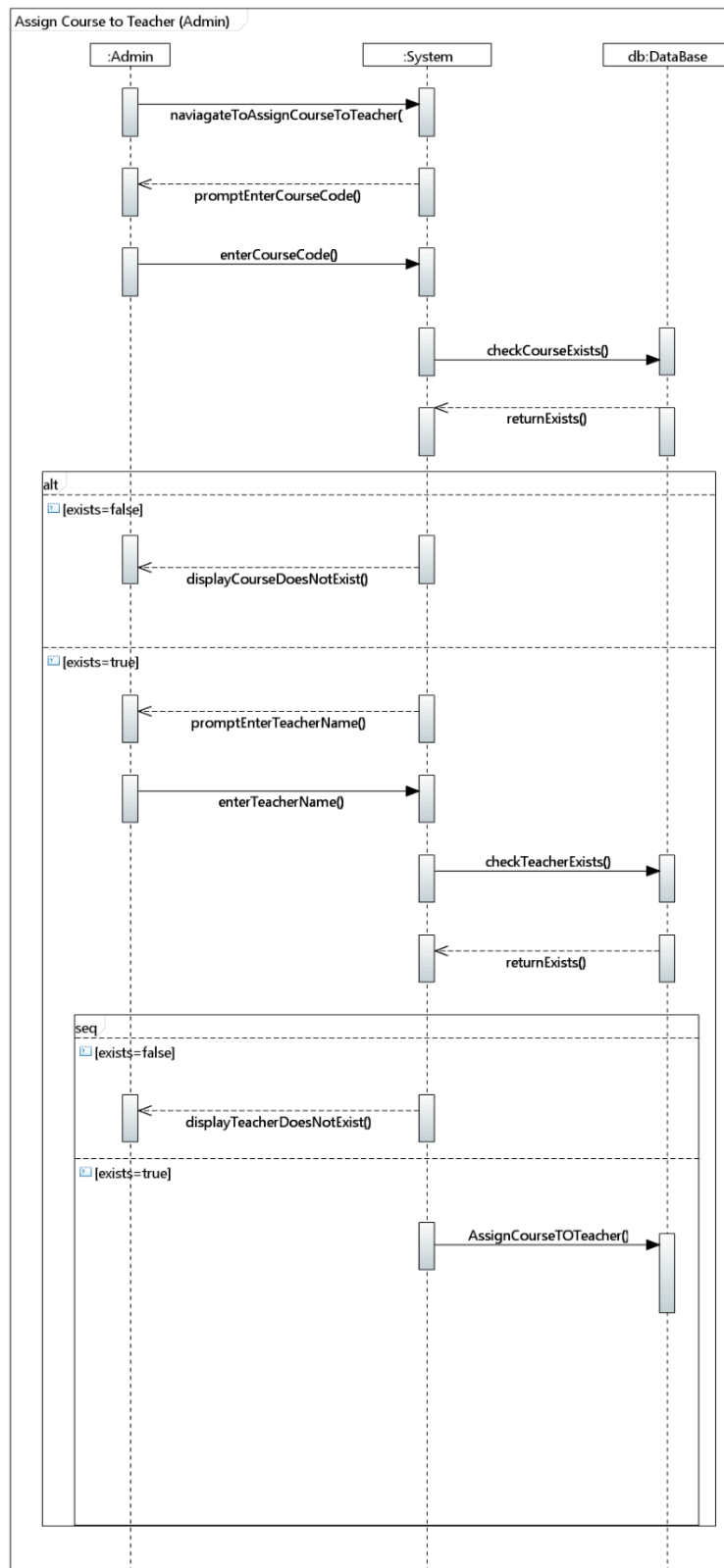
5.2.3.5 Edit Course (Admin)

SMS-005: Edit Course



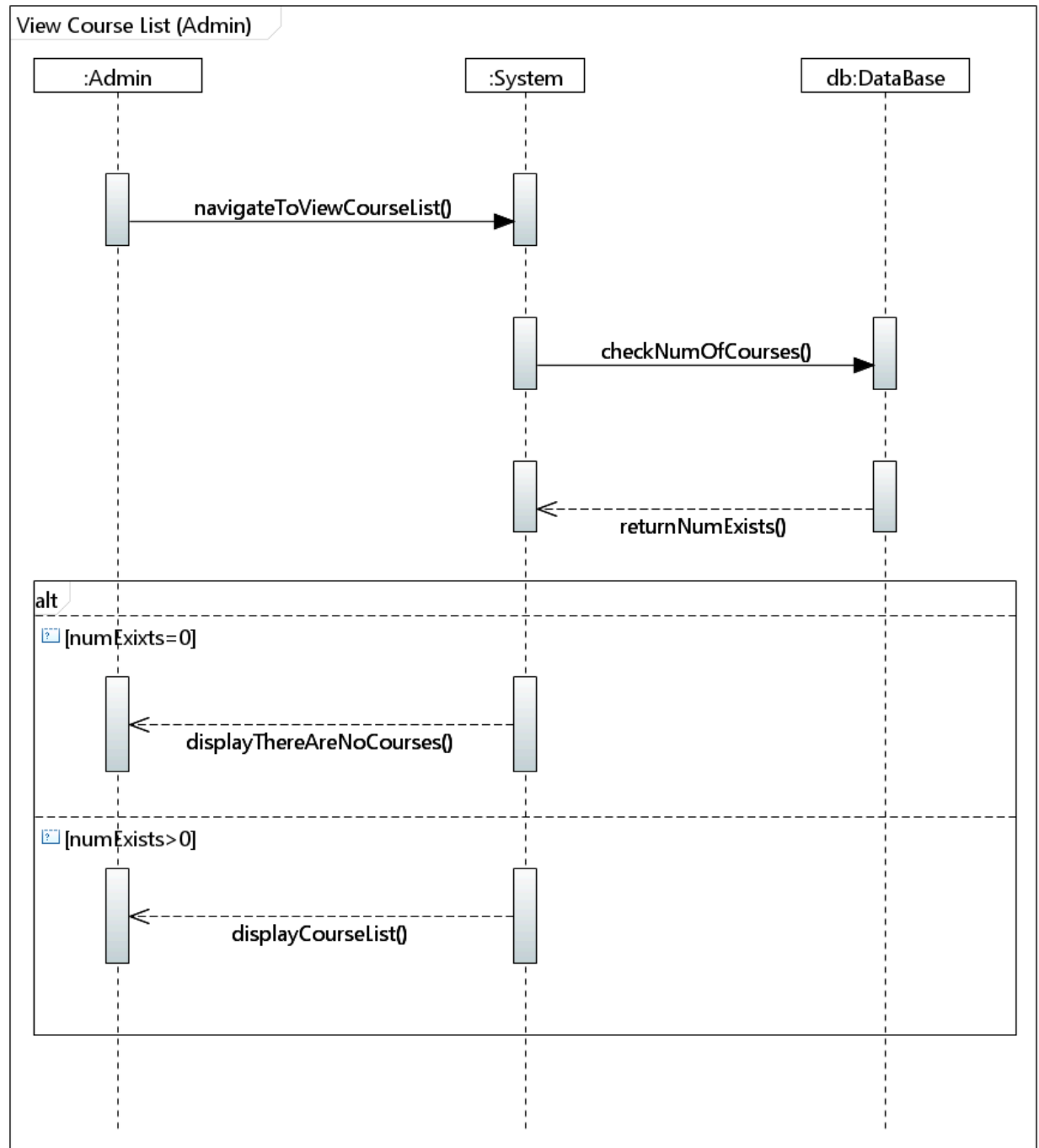
5.2.3.6 Assign Course to Teacher (Admin)

SMS-006: Assign Course to Teacher



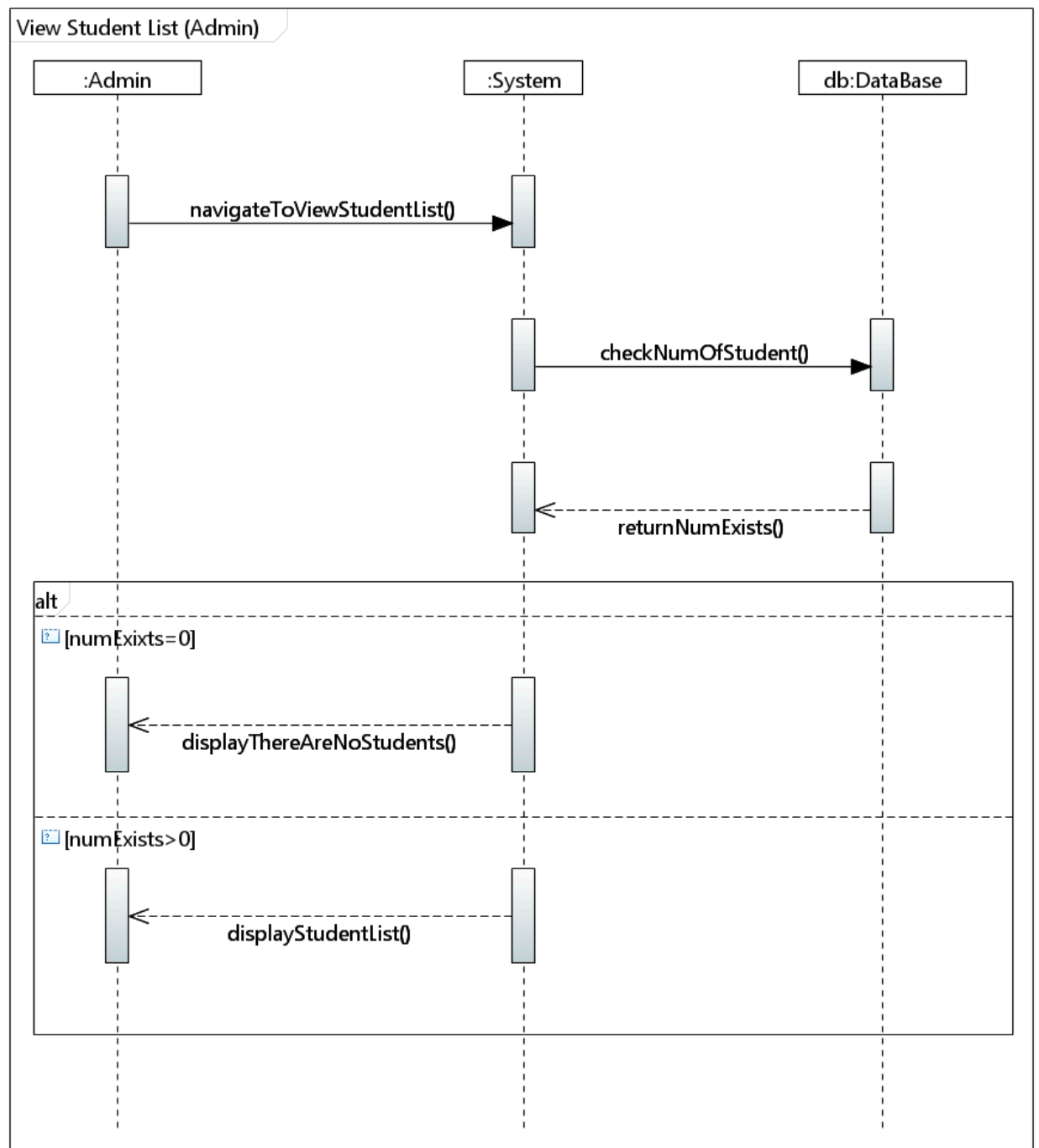
5.2.3.7 View Course List (Admin)

SMS-007: View Course List



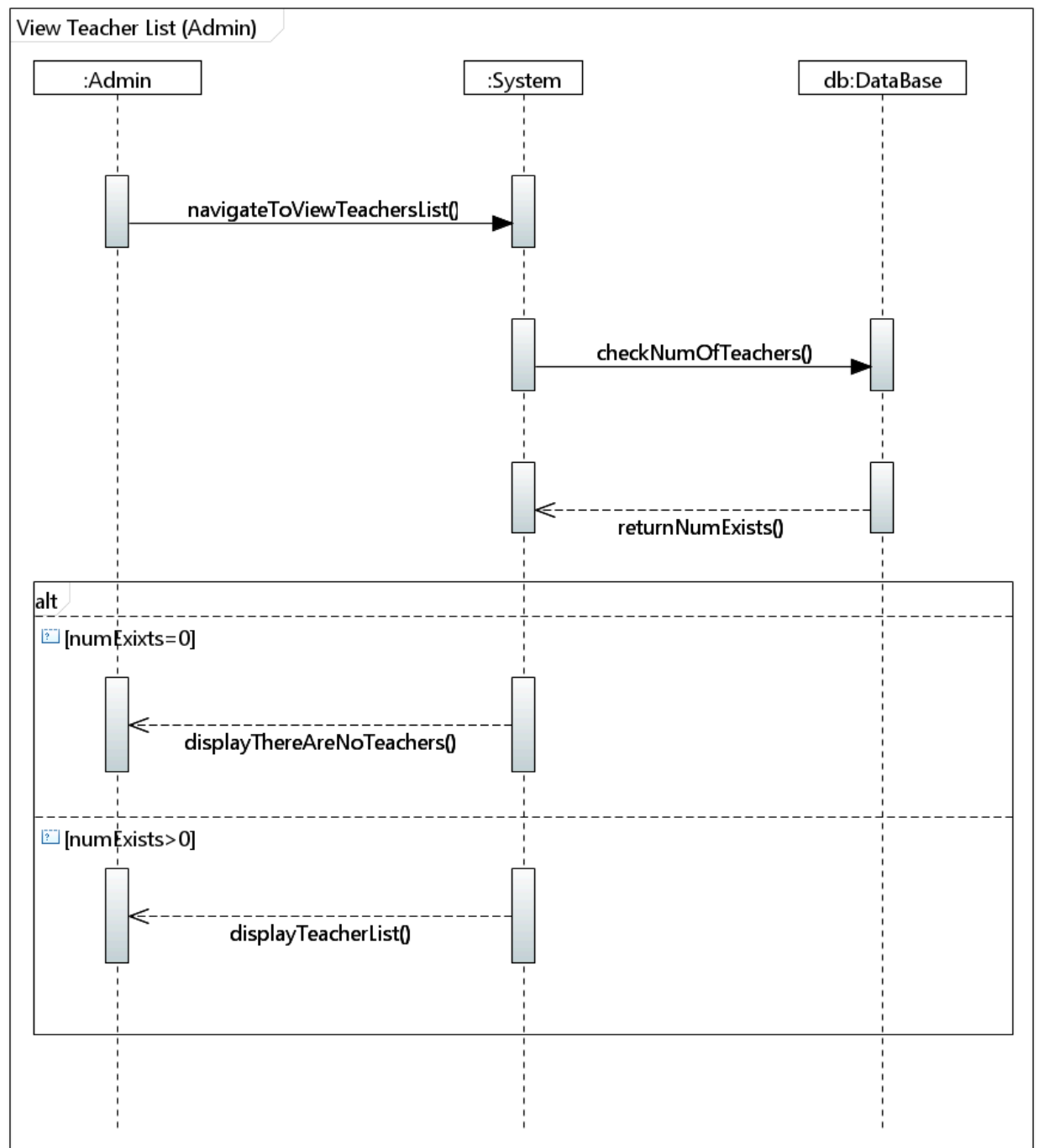
5.2.3.8 View Student List (Admin)

SMS-008: View Student List



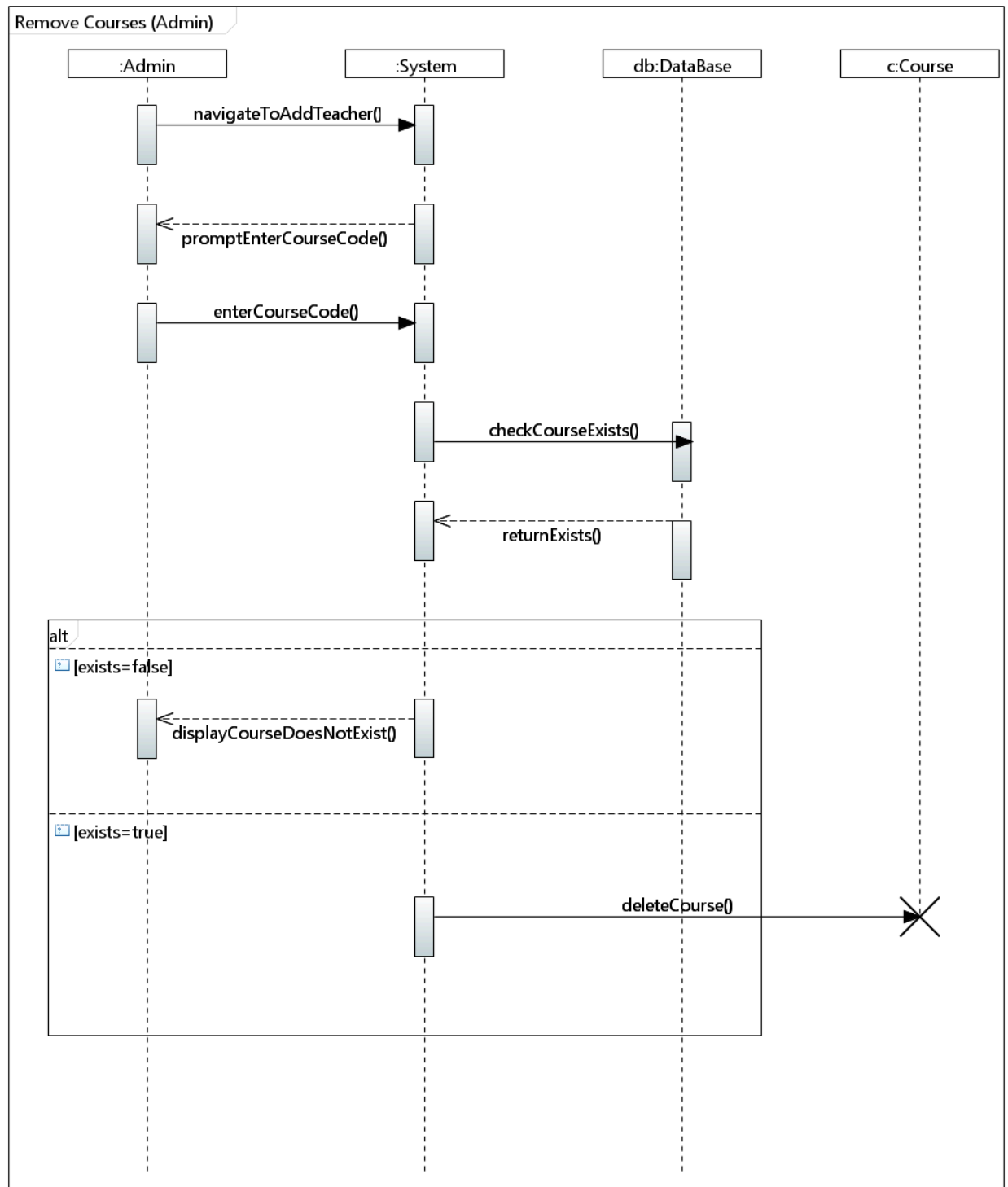
5.2.3.9 View Teacher List (Admin)

SMS-009: View Teacher List



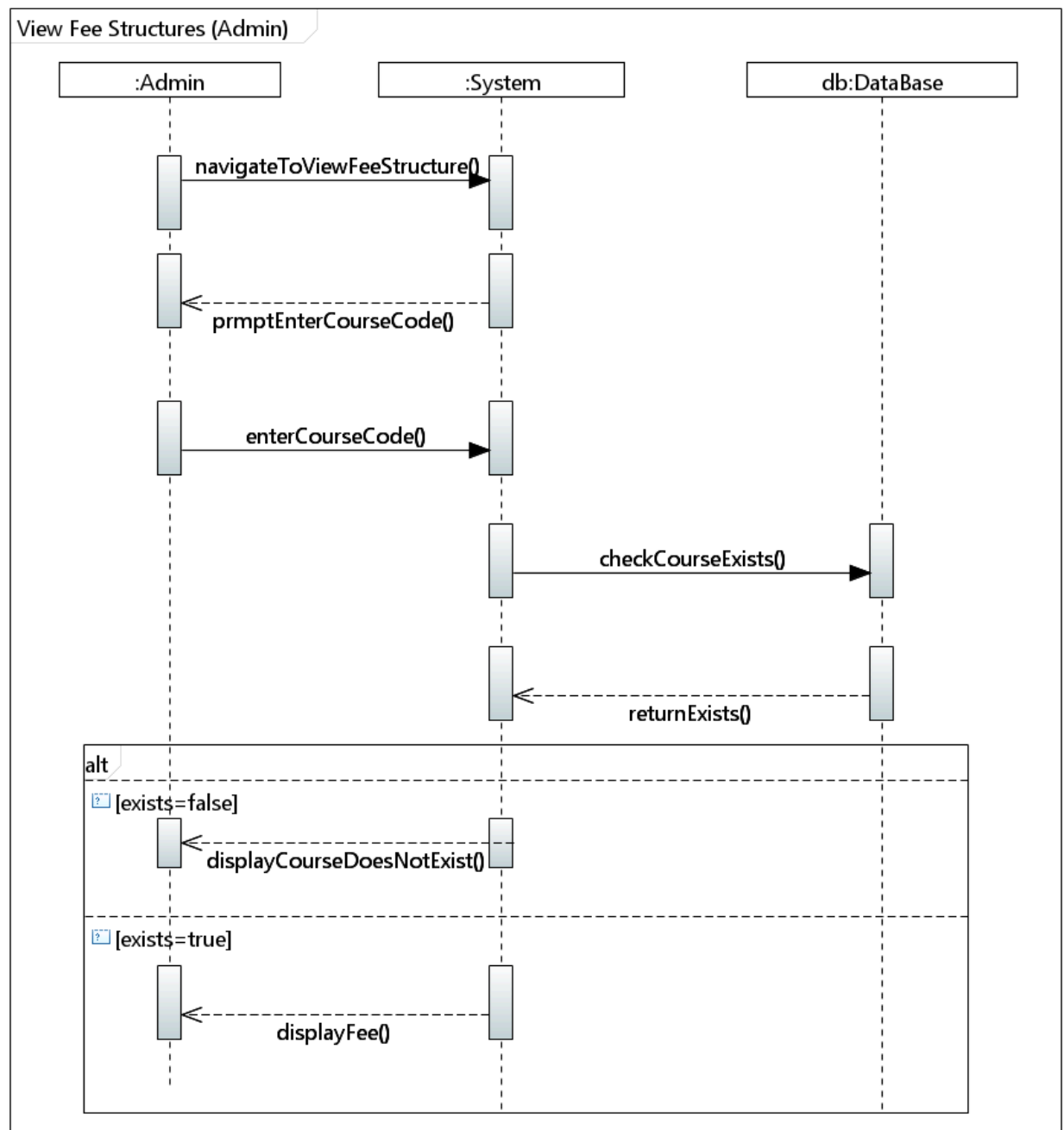
5.2.3.10 Remove Courses (Admin)

SMS-010: Remove Courses



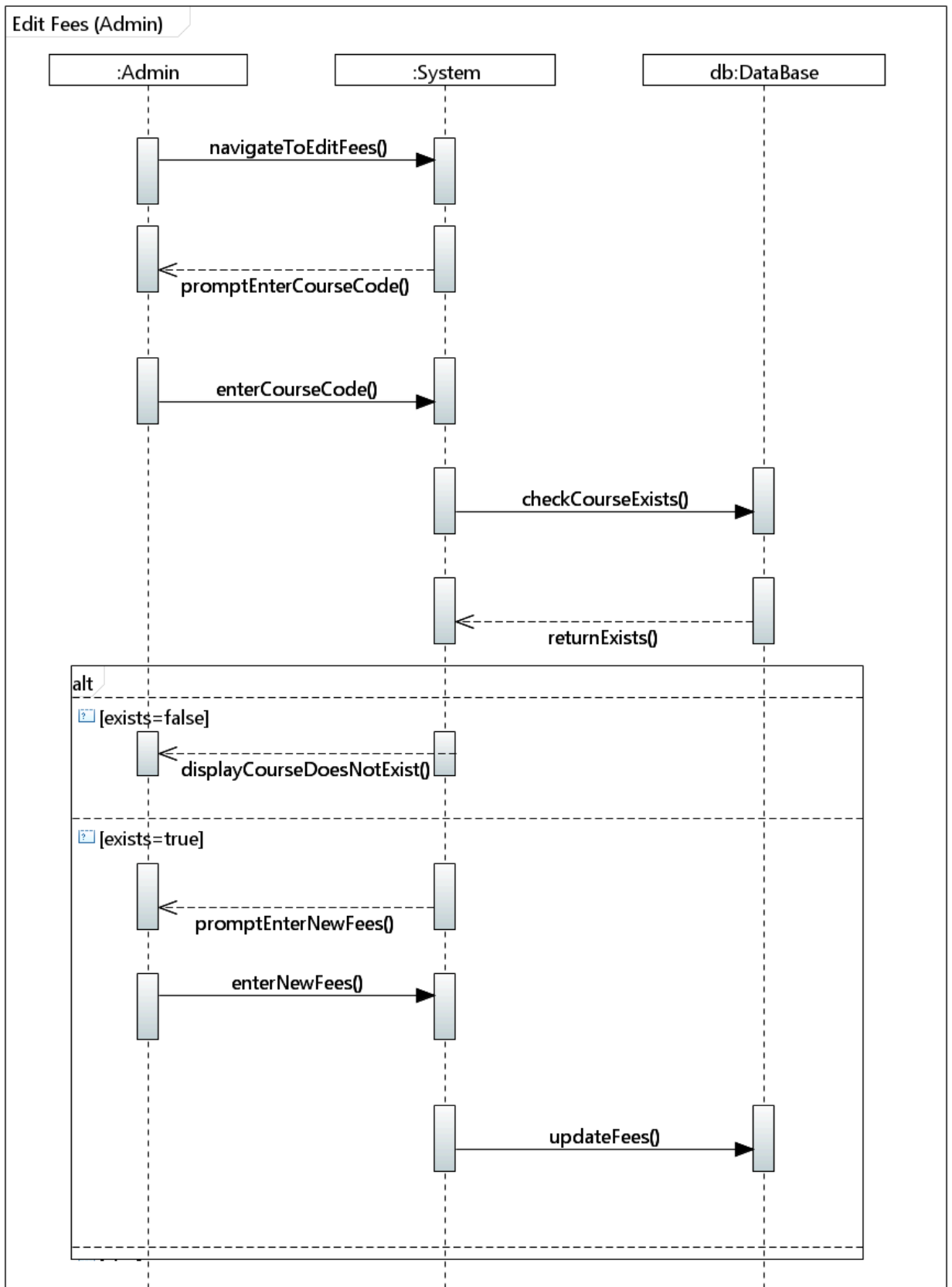
5.2.3.11 View Fee Structures (Admin)

SMS-011: View Fee Structures



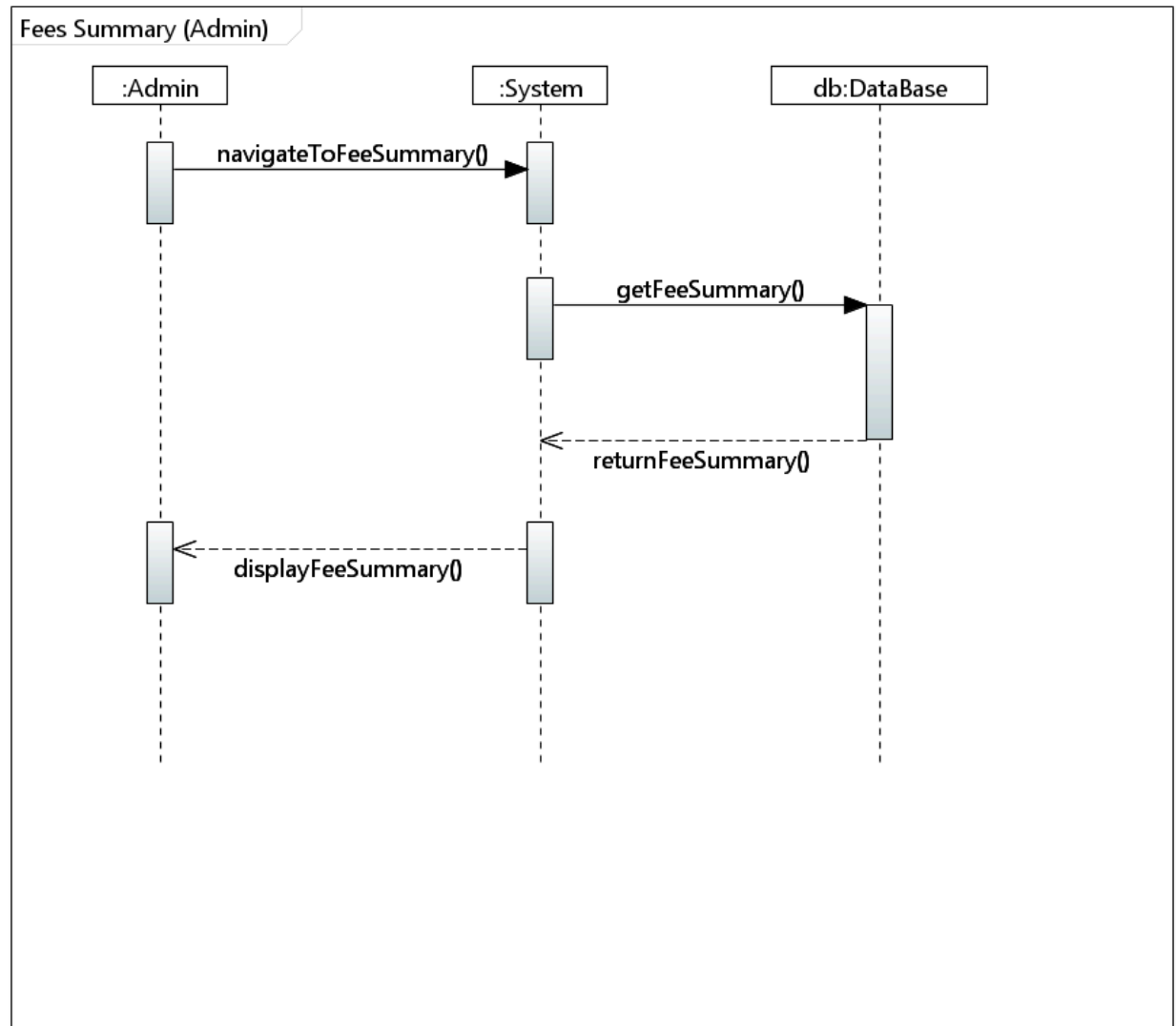
5.2.3.12 Edit Fees (Admin)

SMS-012: Edit Fees



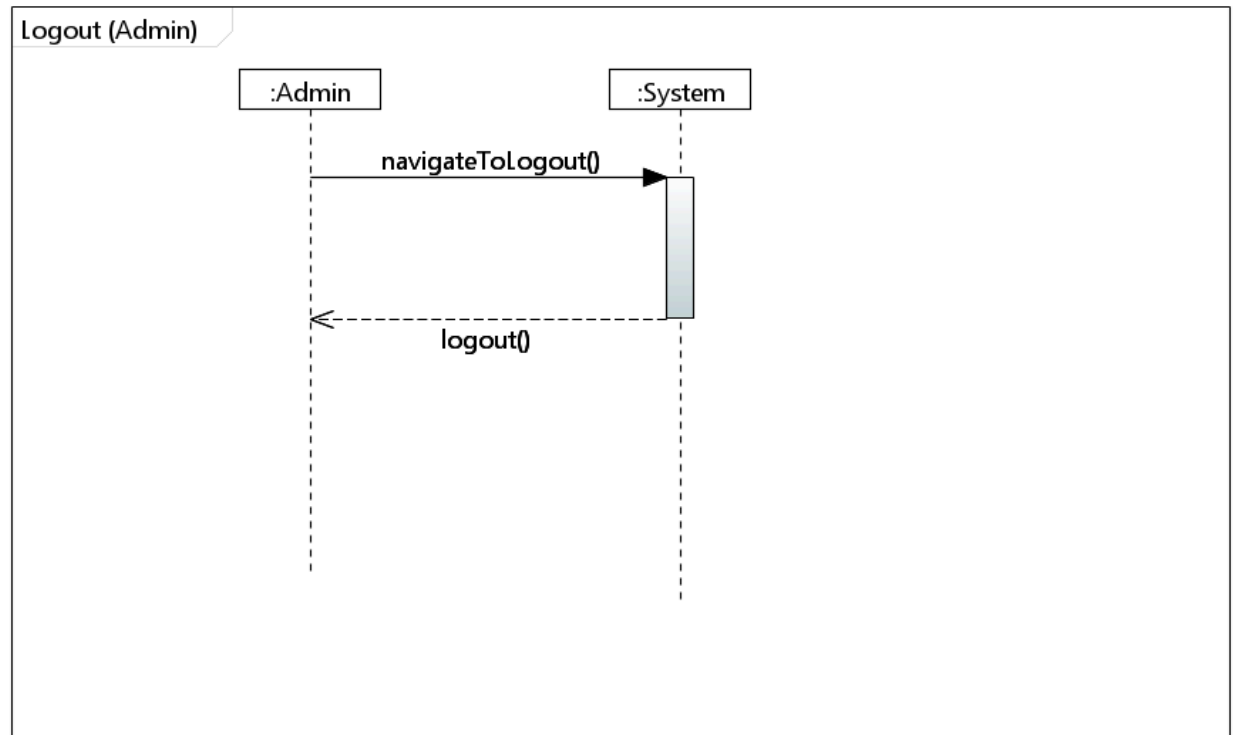
5.2.3.13 Fees Summary (Admin)

SMS-013: Fees Summary

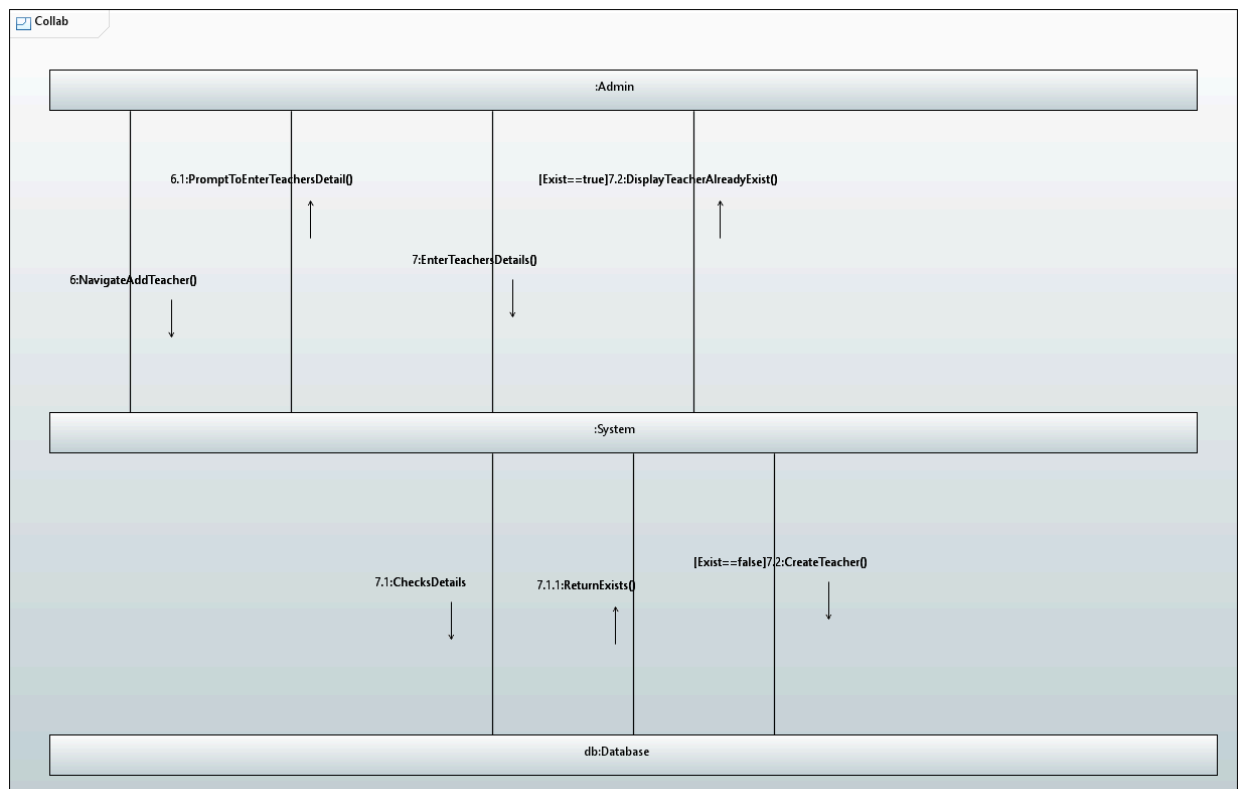
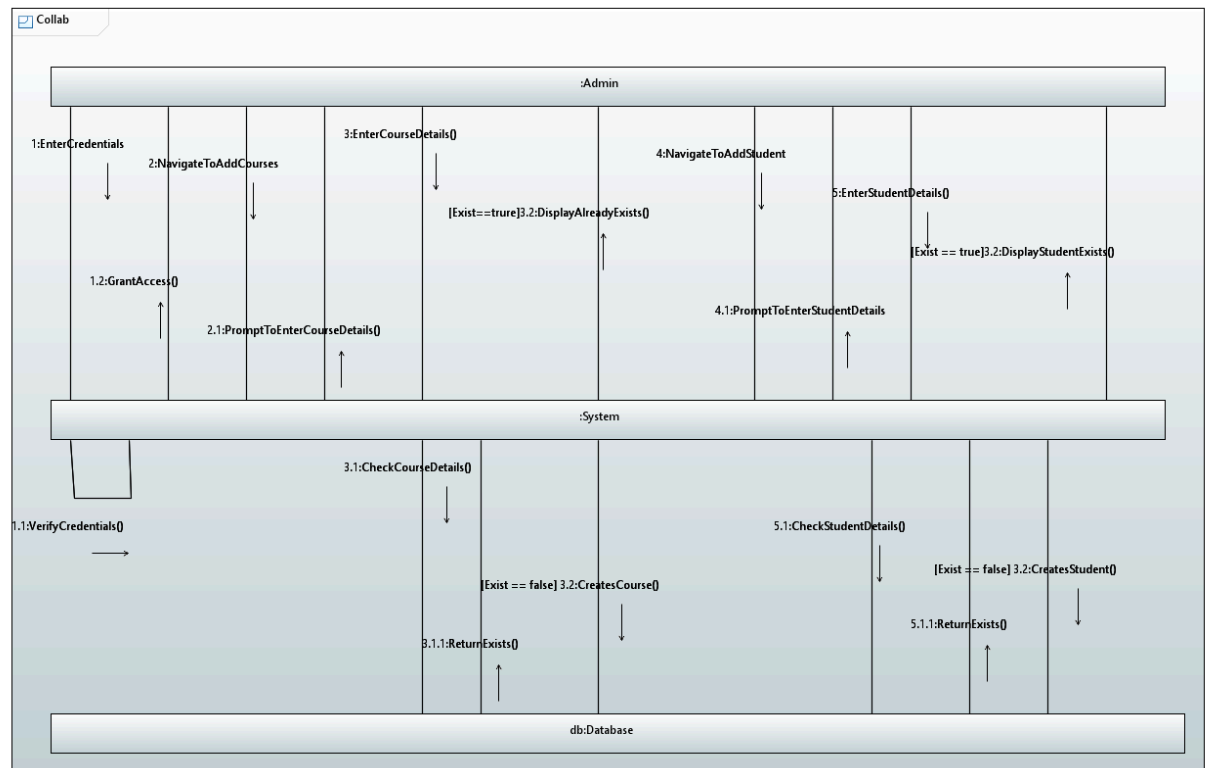


5.2.3.14 Logout (Admin)

SMS-014 : Logout

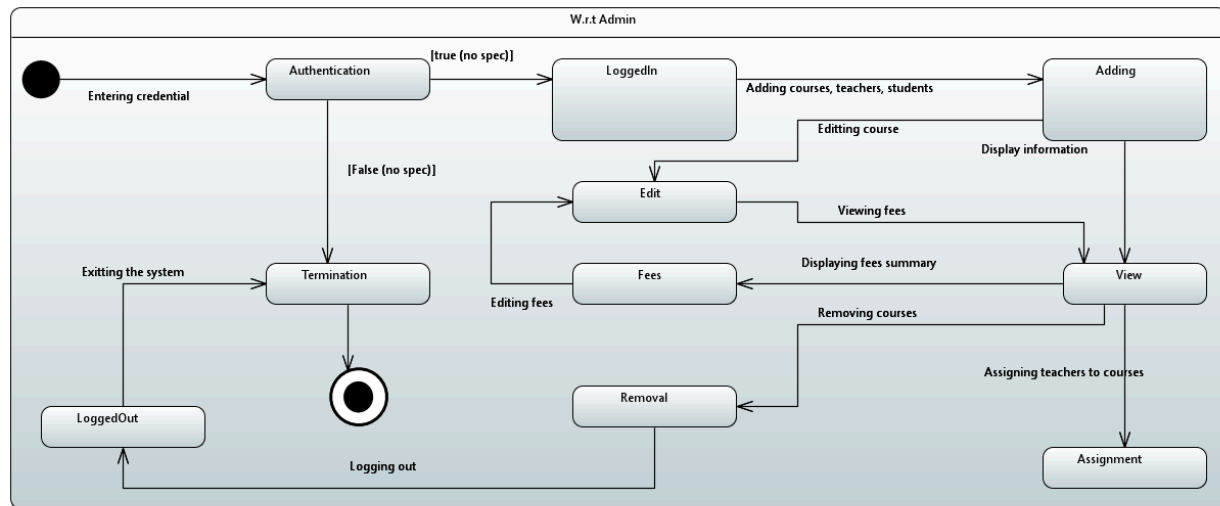


5.2.4 Collaboration Diagram

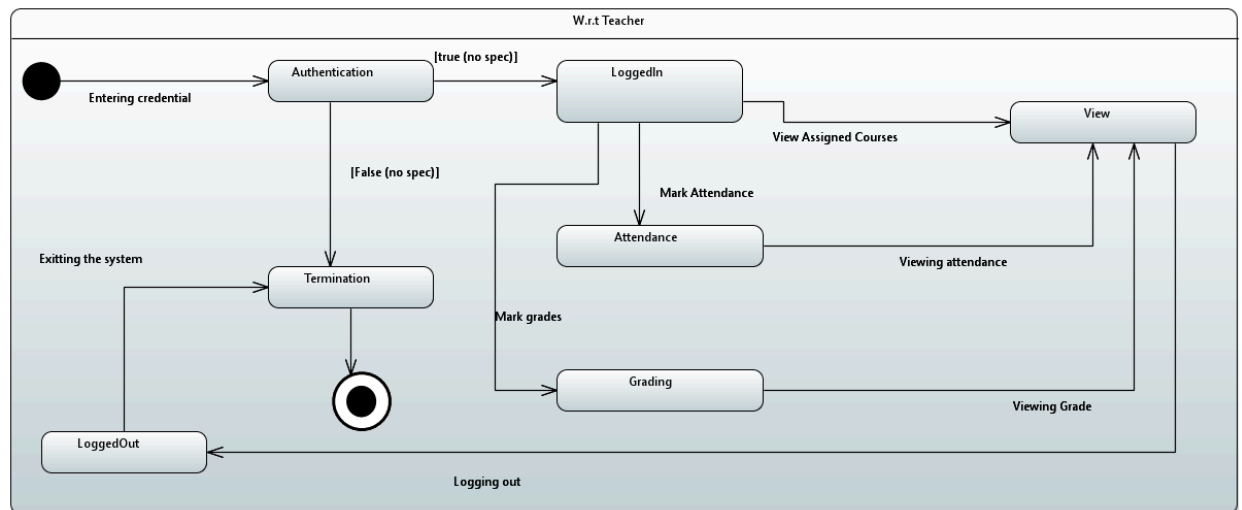


5.2.5 State Diagram

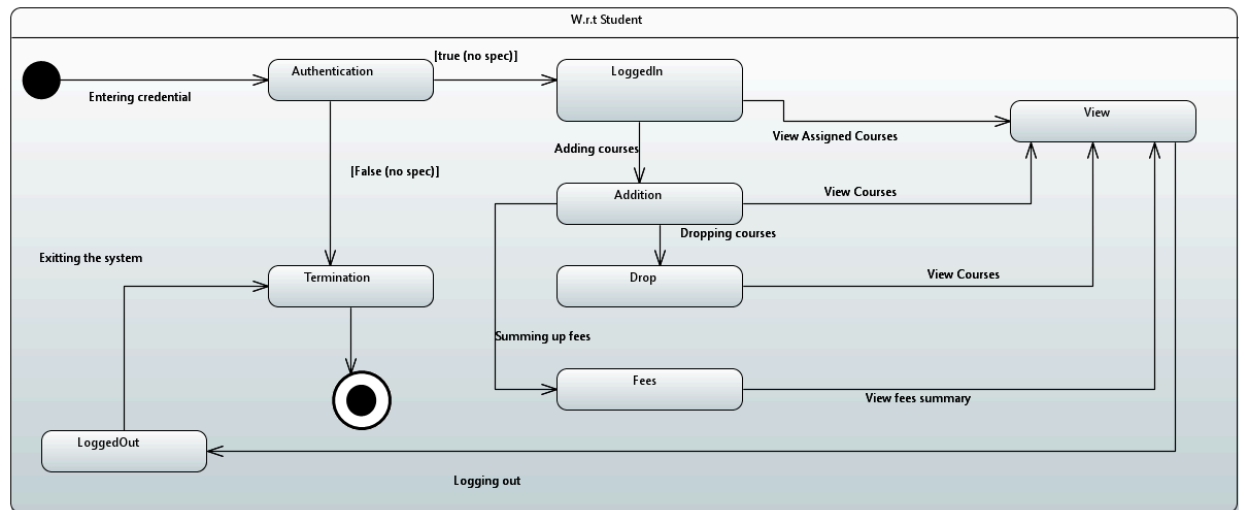
5.2.5.1 W.R.T Admin



5.2.5.2 W.R.T Teacher



5.2.5.3 W.R.T Student



6 References

N/A

Appendices

N/A