SOFTWARE PROJECT LAB-II
SOFTWARE REQUIREMENTS AND SPECIFICATION REPORT

Institute of Information Technology ,

Noakhali Science and Technology University

Supervised By:

Tasniya Ahmed

Assistant Professor,

Institute of Information Technology,

Noakhali Science and Technology University.

Mail:tasniya.iit@gmail.com

Presented By:

Team A Members:

Yeasin Arafat ASH2125013M)

Arpita Majumder (BFH2125015F)

Mohammad Junaid Hossain (MUH2125012M)

Adiba Ahmed Tiasha (BFH2025034F)

Date Of Submission:09/06/2024

Contents

1. Introduction

NSTU Academia is software that provides a powerful course registration system. We hope to simplify the educational process for students. With the help of an intuitive interface, students will be able to register for courses, check the availability of seats, and apply for a seat in the hall. Our goal is to provide a system that is not only effective but also fair and available to all students.

1.1 Project Scope:

The challenge of enrolling in classes is one that NSTU students must overcome. To pay the fees and turn in their applications, they had to rush to the academic building. Moreover, they must face difficulty in paying the fees. Thus, we have developed a fix. This work will become more dependable and hassle-free with our project. Students also must fight for a seat in the hall. Students can apply online for available seats, which will be displayed by our software. As a result, the student can do these things easily by a click away. With this software their academic work will be very easy and reliable. Furthermore, a student must go through academic pressure so this will be a relief for them if the task is done easily.

1.2 Project Scope:

The project scope is given below:

Course Registration Module: The project's main feature, the course registration module, enables online course registration for students. It will most significantly increase student convenience and efficiency by addressing the main complaint with existing registration procedures.

Real-Time Hall Seat Updates: One of the most important ways to help students with their accommodation-related issues is to give them access to real-time updates on available hall seats.

Fee Payment Integration: To simplify the financial side of course registration, it is imperative to integrate a secure payment gateway for online fee payments.

User Authentication and Access Control: To safeguard sensitive student data and uphold privacy and security, it is essential to have strong user authentication procedures and access control features.

Scalability and Future Improvements: To ensure the platform's long-term viability and relevance, it is imperative to design it with scalability and future improvements in mind.

So, these are the scope of our NSTU Academia.

1.3 Motivation

The motivation of the NSTUAcademia is pointed below:
Efficiency Boost: By cutting down on the time and effort students spend on administrative duties like hall seat assignments and course registration, our proposal will free up more time and energy for students to devote to their academic goals.

Convenience for Students: We hope to give students the flexibility to manage course registration and hall seat assignments online, whenever and from any location, eliminating the need to physically visit the academic building or wait in long lines.

Removal of Administrative Hassles: In order to provide a more streamlined and dependable experience for all students, our initiative aims to remove the annoyances related to conventional registration procedures, such as misplaced papers, system malfunctions, and last-minute modifications.

Enhanced Transparency: Real-time updates on available hall seats empower students with the information they need to make informed decisions, reducing uncertainty and anxiety surrounding accommodation arrangements.

Future-Proofing NSTU: By embracing digital solutions and streamlining administrative processes, our project lays the foundation for a more efficient and resilient academic infrastructure at NSTU, capable of accommodating future growth and technological advancements.

1.4 Glossary

This subsection contains definitions of all the terms and abbreviations used in the document.

i. MB – Megabytes

ii. UI – User Interface

iii. SRS – Software Requirement Specifications

iv. API – Application Program Interface

vii. HTML – Hyper Text Markup Language

1.5 References

IEEE. IEEE Std 830-1998 IEEE Recommended Practice for Software Requirements Specifications. IEEE Computer Society, 1998

1.6 Overview

Our project aims to revolutionize the academic experience at NSTU by providing a comprehensive online platform that streamlines course registration and hall seat allocation processes. Through an intuitive interface, students can easily navigate the system to register for courses, pay fees, and submit applications from anywhere, eliminating the need for long queues and paperwork. Additionally, real-time updates on available hall seats enable students to make informed decisions and apply for accommodation seamlessly. With a focus on reliability, efficiency, and convenience, our project aims to empower NSTU students to focus on their studies without the burden of administrative hassles.

2. Stakeholders and Characteristics

Stakeholders of NSTU academia is given below:

1. NSTU Students: The primary stakeholders who will directly benefit from the project's outcomes. They can easily login/signup, work on course registration, attendance tracking, payment processing, profile management, communication with faculty and staff. They are the high priority and high interest stakeholder.
2. Faculty: They are also high priority but low interest stakeholders. They can manage courses, input attendance, review and approve course selections.
3. Hall Administration: They manage hall seat allocations and approvals. They also Display and manage empty and occupied hall seats. Moreover, they review and approve hall seat applications, issue boarding cards, and process hall leaving certificates.
4. Third-party Service Providers: Providers of payment gateways, cloud hosting services, or software tools used in the system.
5. Developer: Responsible for developing, implementing, and maintaining the academic management system.

The characteristics of a house rental system will depend on the specific requirements and goals of the project.

However, some common characteristics that a course registration and hall seat allocation system might have include:

I. User-friendly interface: The system should be easy for students, teachers, and other stakeholders to use, with a clear and intuitive design.

II. Scalability: The system should be able to handle a large number of students, data, teachers data properly.

III. Security: The system should have robust security measures in place to protect the privacy and sensitive information of students, teachers, and other stakeholders.

IV. Integration with other software: The system may need to integrate with other software to do the work properly.

V. Customization: The system should be flexible and customizable to meet the specific needs and needs of the students, teachers and the hall administrations.

3. Design and Implementation Constrains

We have employed design and implementation constraints to ensure the success of this project. It also refers to a tool that allows developers and testers to inspect and interact with the application's user interface (UI) elements.

3.1 JavaScript, JSX and React.js

The visual layout of the components that a user could interact with in a website or technical product is referred to as user interface design, or UI design. In other terms, it is a website's visual design.

3.1.1 Programming Language

JavaScript is an ECMAScript-compliant high-level, frequently just-in-time compiled language. It has first- class functions, dynamic typing, and prototype-based object orientation. Its multi-paradigm, allowing you to program in event-driven, functional, or imperative styles. React is a front-end JavaScript toolkit for creating user interfaces using UI components that is free and open source. Meta and a community of individual developers and businesses support it. JavaScript XML is abbreviated as JSX. It's just a JavaScript syntactic extension. It allows us to create HTML directly in React (within JavaScript code). It is straightforward to generate a template in React using JSX, but it is not a simple template language; instead, it has all of JavaScript's capability.

It is faster than standard JavaScript because it optimizes when converting to standard JavaScript. Rather than dividing the markup and functionality in different files, React makes use of components.

3.1.2 CSS Framework

Cascading Style Sheets (CSS) is a language for specifying the appearance of a document written in a markup language like HTML. Along with HTML and JavaScript, CSS is a key component of the World Wide Web. Semantic UI is a website using UI component framework. Developers may use Semantic UI to create websites with quick and clear HTML, as well as a fully mobile responsive experience. Semantic UI offers a React- integrated version called Semantic UI React, which includes the following functionalities:

- JQuery Free.
- Declarative API.
- Augmentation.
- Shorthand Props.
- Subcomponents.
- Auto Controlled State

3.2 Server-Side Technology

Server-side development refers to the actions that take place behind the scenes when an application is used. It primarily focuses on databases, scripting, website architecture, backend logic, APIs, and Servers.

3.2.1 PHP

Using PHP for the backend of the NstuAcademia system provides several significant advantages. PHP is a mature and well-supported language specifically designed for web development, ensuring stability and reliability. It has a large community, which offers extensive documentation, tutorials, and third-party libraries, making development easier. PHP works seamlessly with databases like MySQL, essential for managing student, teacher, and administrative data in the system. Modern PHP frameworks such as Laravel and Symfony offer robust tools and features like routing, ORM, and security mechanisms, which accelerate development and ensure the application is secure and efficient. PHP's built-in security features help mitigate common web vulnerabilities, and its scalability and performance capabilities can handle high traffic loads. The language's compatibility with payment gateways like SSL Commerce is crucial for handling student fee payments. Additionally, PHP's wide hosting support simplifies deployment, and its open-source nature makes it a cost-effective choice. Overall, PHP's specialization in web development, combined with its extensive support and powerful frameworks, makes it an ideal choice for building and managing the backend of the NstuAcademia system.

3.2.2 Database Server

Using MySQL for the NstuAcademia system offers significant advantages, making it an ideal choice for managing the database needs of this educational platform. MySQL is a highly reliable, open-source relational database management system known for its robust performance and ease of use. It efficiently organizes data into tables and supports complex queries, essential for handling the extensive data related to students, teachers, courses, and administrative functions. MySQL ensures data integrity and reliability with its ACID compliance, which is crucial for maintaining accurate records of course registrations, attendance, and payments. Its compatibility with various programming languages, including PHP, facilitates seamless backend integration, enabling smooth interactions between the application and the database. Additionally, MySQL's support for multiple storage engines optimizes performance for different workloads, enhancing the system's versatility. Features like replication and clustering improve scalability and availability, ensuring continuous operation and data protection. MySQL's extensive community support and comprehensive documentation make it easier to address issues and implement best practices. Being open-source and free to use, MySQL is also cost-effective, reducing operational costs. Its widespread adoption ensures the availability of skilled developers for maintenance and development. Overall, MySQL's performance, reliability, scalability, and strong community support make it an excellent choice for the NstuAcademia system's database management.

4. Requirement Specification

All the requirements based on elicitation process are described in this section.

4.1 Functional Requirement

Functional requirements are those requirements that are used to illustrate the internal working nature of the system, the description of the system, and explanation of each subsystem. It consists of what task the systemshould perform, the processes involved, which data should the system hold and the interfaces with the use.

4.1.1 Create account and login to the system

FR-1	User log in or sign up to the system
Description	The system should allow users to create an account and log in to the system.

Stakeholders	Students, Teacher, Hall staffs	Priority	High

4.1.2 Profile Management

R-2	User update personal details such as name, batch, department, profile image, hall name, and residential status.		
Description	The system must allow users to manage their personal information.		
Stakeholders	Students	**Priority**	High

4.1.3 Course Registration

FR-3	Display available courses for the selected semester and allow students to select courses.		
Description	The system must allow students to register for courses for a selected semester.		
Stakeholders	Students, Teacher, Hall Staff	**Priority**	High

4.1.4 Select Semester

FR-4	Enable students to choose the semester for course registration.
Description	The system must allow students to select the semester for course registration.

Stakeholders	Students	**Priority**	High

4.1.5 Choose Courses

FR-5	Allow students to select courses from the list of available courses for the chosen semester.		
Description	The system must display a list of available courses for the selected semester.		
Stakeholders	Students	**Priority**	High

4.1.6 Attendance Check

FR-6	A Verify student attendance for each course session.		
Description	The system should allow students to check their attendance for registered courses.		
Stakeholders	Students, Teachers, Admin, Administration,	**Priority**	Medium

4.1.7 Attendance Input

FR-7	Record student attendance for each course session.		
Description	The system should allow teachers to input attendance records for students.		
Stakeholders	Students, Teacher	**Priority**	Medium

4.1.8 Course Selection Approval

FR-8	The system route course registration requests for approval and signature.		
Description	The system should allow teachers to approve students' course selections.		
Stakeholders	Teachers, Course Coordinator, Students	**Priority**	High

4.1.9 Approval and Signature

FR-9	Approve course selections and provide necessary signatures.		
Description	The system should allow for the approval and signature of course registration forms.		
Stakeholders	Students, Administration, Hall staff, Admin	**Priority**	Medium

4.1.10 Payment Processing

FR-10	Handle the transaction process through a secure payment gateway.		
Description	The system must provide a payment gateway for processing course registration fees.		
Stakeholders	Students, Admin	**Priority**	High

4.1.11 Receive Confirmation

FR-11	User can change the existing system language		
Description	The system must generate a payment confirmation receipt once the payment is successful.		
Stakeholders	Students, Admin	**Priority**	High

4.1.12 Communication System

FR-12	Generate and provide a payment confirmation receipt to the student upon successful transaction.		
Description	The system must provide a communication platform for students, teachers, and hall staff to interact.		
Stakeholders	Students, Teachers, Hall Staff	**Priority**	High

4.1.13 Attendance Management

FR-13	Enable teachers to record and manage student attendance.		
Description	The system must allow teachers to input attendance records for their allocated courses.		
Stakeholders	Faculty	**Priority**	High

4.1.14 Hall Seat Allocation

FR-14	Manage the process of allocating hall seats to students, including application, approval, and status updates.		
Description	The system must manage the hall seat allocation process, allowing students to apply for hall seats.		
Stakeholders	NSTU Students, Hall Staff	**Priority**	High

4.1.15 Display Available Seats

FR-15	Show real-time availability of hall seats for students to view and apply for.		
Description	The system must display available hall seats for students to view and apply for.		
Stakeholders	NSTU Students	**Priority**	High

4.1.16 Application for Hall Seat

FR-16	Show real-time availability of hall seats for students to view and apply for.		
Description	The system must allow students to apply for available hall seats.		
Stakeholders	Students	**Priority**	High

4.1.17 Approval & Boarding Card Issuance

FR-17	Verify hall seat applications and issue boarding cards to approved applicants.		
Description	The system must verify hall seat applications and issue boarding cards.		
Stakeholders	Hall Staff	**Priority**	High

4.1.18 Manage Empty Seat Display

FR-18	Update and maintain the display of empty hall seats in real-time		
Description	The system must update and display the status of hall seats in real-time.		
Stakeholders	Hall Staff	**Priority**	High

4.1.19 Certificate Submission

FR-19	Manage the process for students to submit leaving certificates when vacating hall seats.		
Description	The system must manage the certificate submission process for students vacating hall seats.		
Stakeholders	Students, Hall Staff	**Priority**	High

4.2 Data Requirement

The system should store student profiles, course information, attendance records, payment details, and hall seat allocation data. It should maintain a secure database with user authentication and authorization mechanisms to ensure data privacy and integrity.

4.2.1 Image Upload Size Limit

DR-1	Stakeholders must provide specific sized picture for the house		
Description	Users will upload profile pictures, course-related documents, and hall seat application materials. Images must be within a maximum size limit of 3 MB.		
Stakeholders	Students, Teacher, Hall Staff	**Priority**	High

4.2.2 Personal Information Storage

DR-2	Store and manage student personal information securely.		
Description	The system will store personal information such as name, email, batch, department, and residential details of users.		
Stakeholders	Students, Teacher, Hall Staff	**Priority**	**High**

4.2.3 Course Information Storage

DR-3	Store and manage course related data.

Description	Course-related data including course code, title, description, and instructor details will be stored in the system.		
Stakeholders	Teacher, System Staff	**Priority**	High

4.2.4 Attendance Records

DR-4	Store and manage course details and related information.		
Description	The system will record attendance data, including the number of classes held and attended by students for each course.		
Stakeholders	Teacher, System Staff	**Priority**	High

4.2.5 Payment Details

DR-5	Financial data will be stored in the system database.		
Description	Payment information such as transaction ID, amount, and payment status will be stored securely.		
Stakeholders	Students, Admin	**Priority**	High

4.2.6 Hall Seat Allocation Data

DR-6	Analytical report of ads and boosted ads
Description	Information regarding hall seat allocation, including available seats, applicant details, and boarding card issuance records, will be maintained.

Stakeholders	Hall Staff, Administration, Admin	**Priority**	High

4.3 Performance Requirement

It is important to maintain the performance of the software system. To ensure performance we maintain thesesteps

4.3.1 Faster Course Search and Load

PR-1	Load course details and images independently to enhance user experience.		
Description	Course search results should display within 1 second, and course details, including associated documents and images, should load within 5 seconds given good network speed.		
Stakeholders	Students, Teacher, Hall Staff, Admin	**Priority**	Medium

4.3.2 Quick Problem Management

PR-2	Provide a mechanism for users to report issues directly from the system.		
Description	The system should quickly detect and manage issues, ensuring minimal disruption to users by resolving problems within a short time frame.		
Stakeholders	Students, Teachers, Hall Staff	**Priority**	Medium

4.3.3 Efficient Payment Processing

PR-3	Provide real-time updates on the status of payment transactions.		
Description	Payment transactions should be processed within 3 seconds to provide immediate feedback to the user.		
Stakeholders	Students	**Priority**	High

4.3.4 Real-Time Attendance Updates

PR-4	Provide real-time updates on the status of Attendance.		
Description	Attendance records should update in real-time, with a maximum delay of 2 seconds between submission and database reflection.		
Stakeholders	Students	**Priority**	High

4.3.5 Rapid Hall Seat Allocation Updates

PR-5	Provide real-time updates on the status of Rapid Hall Seat Allocation process.		
Description	Hall seat availability status should update within 2 seconds after any change, ensuring real-time accuracy.		
Stakeholders	Students, Hall Staff	**Priority**	High

4.3.6 Responsive User Interface

PR-6	Ensure the system's interface adjusts seamlessly to various screen sizes and devices for optimal usability.		
Description	All user interface actions, such as clicks, navigations, and form submissions, should respond within 1 second to ensure a fluid user experience.		
Stakeholders	Students, Teacher, Hall Staff	**Priority**	High

4.4 Dependability Requirements

If NSTUacademia can provide availability, reliability, safety, security then this system will be dependable.

4.4.1 Availability and Reliability Requirements

This system will be available for 24 hours. Stakeholders can use the system anytime they want and cansee ads or rate a house and upload information which is very reliable for stakeholders.

4.4.2 Safety Requirements

This system will not contain any malware, and this will not harm any stakeholder’s device.

4.5 Maintainability and Supportability

4.5.1 Maintenance Requirements

MR-1	Make the code maintainable.		
Description	Code must be developed so that it can be modified later and will be readable.		
Stakeholders	Developers	**Priority**	High

4.5.2 Supportability Requirements

For accessing information, the system will use some authorization techniques to ensure that correct data isused by the correct user.

4.6 Security Requirements

Securing information is much more important for a system to get users dependability.
Here is some of them:

4.6.1 Access Requirements

For accessing information, the system will use some authorization techniques to ensure that correct data isused by the correct user.

4.6.2 Integrity Requirements

Integrity requirements refer to a security system which ensures an expectation of data quality. It also ensuresthat all data of the system would never be exposed to malicious modification or accidental destruction. For preventing anonymous access to user password, the system will use encryption technique called Hash Function for encrypting user password.

4.6.3 Privacy Requirements

Privacy requirements enhance to protect stakeholder's privacy. In this way, all data or a partial part of data are going to be disclosed according to system's privacy policy. To ensure privacy, the central database shouldbe protected by the anonymous. Users are permitted to get access to those data which are being associated bythem which can be ensured by the user log in system.

4.7 Usability and Human Integrity Requirements

This system will provide a more user-friendly environment.

4.7.1 Ease of Use Requirements

Our system will be easier to use for any type of person and they don't need any training to use the system.

4.7.2 Personalization and Internationalization Requirements

There are no personalization and internationalization requirements in our system.

4.8 Look and Feel Requirements

Look and feel requirements mainly refers to what the system will look like.

4.8.1 Appearance Requirements

AR-1	Text color and font		
Description	The user interface must be responsive and work seamlessly on various devices, including desktops, laptops, tablets, and smartphones.		
Stakeholders	NSTU Students, Teacher, Hall Staff	**Priority**	High

4.8.2 Responsive Design

AR-2	The user interface is responsive and works seamlessly on various devices.		
Description	All texts and description will be at a good font size so that users can understand what important and mandatory input fields will be kept red colored until user put correct information.		
Stakeholders	Students, Teacher, Hall Staff	**Priority**	High

4.9 Style Requirements

There are no style requirements in our system.

4.10 Operational and Environmental Requirements

Operational and environmental requirement refers to the capabilities, performance measurements, process, measurements of effectiveness, measurements of performance, measures of sustainability, measurements of
technical performance etc.

4.10.1 Expected Physical Requirements

There are no expected physical requirements in our system.

4.10.2 Requirements for Interfacing with Adjacent Systems

There are no requirements for interfacing with adjacent system for our project.

4.10.3 Release Requirements

There are no specific release requirements in our system.

4.11 Legal Requirements

Legal requirements normally refer to the terms and conditions or privacy policy of any organization.
The terms and condition of our application is that no third-party software or person allowed to engage to useour data for their business purpose.

5. Requirement Engineering Process

Process Requirements Engineering (RE) determines software requirements according to customer requirements or needs. Requirements engineering process includes requirements elicitation, needs modeling,requirements analysis, requirements assurance & validation, and requirements management.

5.1 Requirement Elicitation Techniques

Techniques Requirements elicitation is the practice of researching and finding system requirements for users,customers, and other stakeholders, also referred to as "requirement gathering". Requirement elicitation can be done by contacting participants directly or by doing some research, analysis and testing

5.1.1 Interviews

We hold discussions that can be held individually or with a small group of participants. They are an effectiveway to access services without spending a lot of time with participants because we meet with people to discuss only certain important requirements of this program. Negotiations are useful for obtaining individual requirements for members in organizing workshops where those members of the program come together to resolve any issues or conflicts. We mainly perform our interview based on specific criteria.

- Short description about goals and objectives
- Open-Ended Questions
- Active Listening
- Take Detailed Notes

5.1.2 Existing System Analysis

Existing systems can help to show how systems are currently operating or what they are what I should do. The system includes written information about current programs, business processes, needs specifications, and competitor research. Review once textual analysis can help Software Requirements Specification for To-let determine which performance should remain and functionality that isn't in use. After existing document in analysis, we found several problems with the existing system.

- Existing systems cannot perform category-wise price range. There is no shifting process on it.
- No cloud storage system is provided by the existing systems.

5.1.3 System Interface Analysis

The first thing to do is to identify which systems the system-to-be shall communicate with. It could be a serveron the Internet, a piece of software on the same host as the system-to-be, some hardware or something completely different.

5.1.4 Questionnaires

The questionnaire is a useful way to investigate styles, changes in attitudes and users' ideas, and user satisfaction with priorities and preferences. Our list of questions was as short as possible. The respondent may be tired or frustrated.

Had a basic reason for all the questions as well as group the topic areas together for the respondent to focus on. The main advantage of this survey responses was that they were collected in the usual way. Information was summarized by many people.

5.2 Requirement Validation

Requirement validation ensures that the requirements are correct and reflect the quality you want from this program. In the beginning, our requirements looked good but when we read them and tried to work with them,they came out having ambiguities and gaps.

5.2.1 Review the Requirements

Negative peer review, especially the type of rigorous review called evaluation, is unique among the highest quality software processes available. We had a team of reviewers representing different perspectives and carefully examined written needs, analysis models, and related information on disability.

5.2.2 Test the Requirements

The test creates another view of the requirements. We also performed writing tests regarding assurance of whether the expected performance was found or not. Getting tested by the user needs to document the expected product behavior under specified condition.

6. Use case diagram

Use case diagram comprises actors and use cases, where actors perform several cases or one. This also showswhich actors have access to which use case. Here is the Use Cass

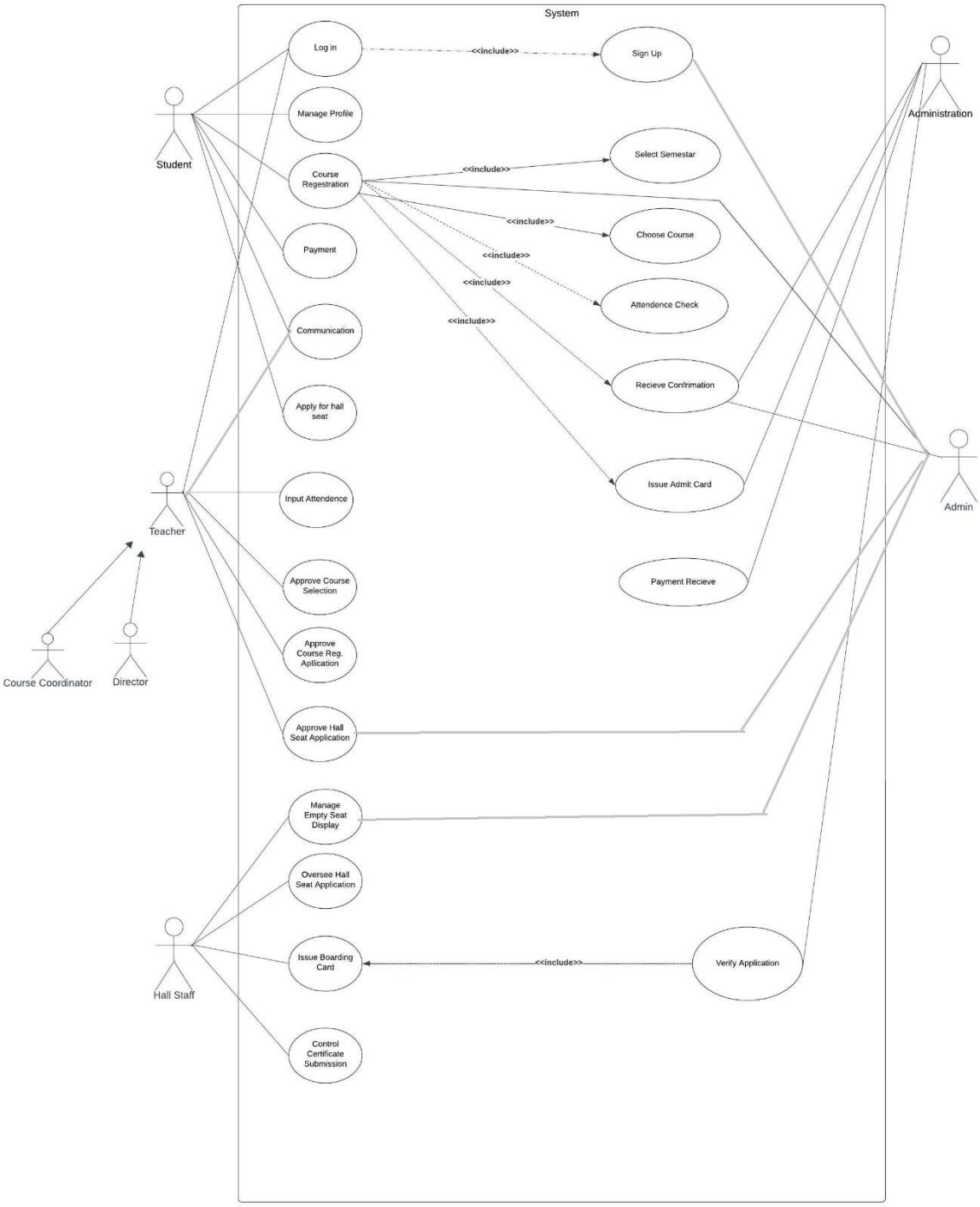

Fig: Use Case Diagram for NSTUAcademia

7. Use Case Description

All use cases from use case diagram are explained here.

Table 1 Log In:

Use Case 1	Log In	
Goal	Users want to log in to their accounts.	
Preconditions	Users must have a registered account in NstuAcademia.	
Success End Condition	User successfully logs in.	
Failed End Condition	Login fails due to incorrect credentials or system error.	
Primary Actors: **Secondary Actors:**	Students, Teachers, Hall Staffs Admin, Administration	
Trigger	User initiates the login process.	
Main Success Flows	**Step**	**Action**
	1	User navigates to the login page.
	2	User enters their username and password.
	3	System validates the credentials.
	4	System grants access to the user.
Alternative Flows	**Step**	**Branching Action**
	3a	User enters incorrect credentials.
	4a	System displays an error message and prompts the user to re-enter valid credentials.
Quality Requirements	**Step**	**Requirement**
	7	Login process must be secure and efficient.

Table 2: Sign Up

Use Case 2	Sign Up	
Goal	Users want to create a new account.	
Preconditions	User must come to our system	
Success End Condition	Users successfully create a new account.	
Failed End Condition	Users cannot create a new account.	
Primary Actors: **Secondary Actors:**	Students, Teachers, Hall Staffs Admin	
Trigger	Sign up request.	
Main Success Flows	**Step**	**Action**
	1	Users request to sign up.
	2	System displays the sign-up form.
	3	Users fill in the required information.
	4	System verifies the information.
	5	System creates a new account.
Alternative Flows	**Step**	**Branching Action**
	3a	Users enter invalid information.
	7a	System displays an error message and prompts users to correct the information.
Quality Requirements	**Step**	**Requirement**
	7	Users must confirm the code within 2 minutes. After this time the confirmation code will be invalid.

Table 3: Manage Profile

Use Case 3	Manage Profile	
Goal	Users can manage their personal profiles.	
Preconditions	User must be logged in.	
Success End Condition	Users' profiles are updated successfully.	
Failed End Condition	Users' profiles are not updated.	
Primary Actors: **Secondary Actors:**	Students, Teachers, Hall Staffs, Administration Admin	
Trigger	Profile update request	
Main Success Flows	**Step**	**Action**
	1	User logs into NstuAcademia.
	2	User navigates to the profile management section.
	3	User updates profile information (e.g., name, contact details, photo).
	4	System validates and saves the updated information.
	5	System confirms successful profile update to the user.
Alternative Flows	**Step**	**Branching Action**
	3a	User enters invalid data (e.g., invalid email format).
	4a	System displays an error message and prompts the user to correct the information.
Quality Requirements	**Step**	**Requirement**
	7	Profile updates must be reflected immediately in the user’s account

Table 4: Select Semester

Use Case 21	Select semester	
Goal	To enable students to select their desired semester for academic enrollment	
Preconditions	Students must be registered in the system. Admin and faculty must have updated the system with the available semesters.	
Success End Condition	Students must be registered in the system. Admin and faculty must have updated the system with the available semesters.	
Failed End Condition	Students are unable to select or enroll in a semester due to system errors or unavailability of options.	
Primary Actors: **Secondary Actors:**	Admin, Administration Teacher, Student	
Trigger		
Main Success Flows	**Step**	**Action**
	1	Student logs into the academic portal.
	2	Student navigates to the semester selection section.
	3	Student reviews the available semesters and selects one.
	4	System confirms the selection and enrolls the student in the chosen semester.
	5	Admin and faculty receive notification of the new enrollment.
Alternative Flows	**Step**	**Branching Action**
	3a	If the desired semester is not available, the student contacts admin or faculty for assistance.
	4a	If there is a system error during enrollment, the system alerts the student and admin for further action.
Quality Requirements	**Step**	**Requirement**
	6	The system must be accessible at all times for semester selection.

Table 5: Course Registration

Use Case 11	Course Registration	
Goal	Students want to register for courses for the upcoming term.	
Preconditions	Student must be logged into NstuAcademia during the course registration period.	
Success End Condition	Student successfully registers for desired courses.	
Failed End Condition	Registration fails due to system error or course availability.	
Primary Actors: **Secondary Actors:**	Students, Administration Admin	
Trigger	Course registration period begins.	
Main Success Flows	**Step**	**Action**
	1	Student logs into NstuAcademia.
	2	Student navigates to the course registration section.
	3	Student selects desired courses from available options.
	4	System checks course availability and prerequisites.
	5	System confirms successful registration to the student.
Alternative Flows	**Step**	**Branching Action**
	3a	Users enter invalid information.
	7a	System displays an error message and prompts users to correct the information.
Quality Requirements	**Step**	**Requirement**
	7	Profile updates must be reflected immediately in the user's account.

Table 6: Choose course

Use Case 9	Choose Courses	
Goal	To enable students to select courses for their academic term.	
Preconditions	Students must be registered in the system. Admin and faculty must have updated the system with the available courses.	
Success End Condition	Students successfully select their courses and are enrolled in them for the academic term.	
Failed End Condition	Students are unable to select or enroll in courses due to system errors or unavailability of options.	
Primary Actors: **Secondary Actors:**	Student Teacher, Admin	
Trigger	Students initiate the course selection process.	
Main Success Flows	**Step**	**Action**
	1	Student logs into the academic portal.
	2	Student navigates to the course selection section.
	3	Student reviews the available courses and selects the desired ones.
	4	System confirms the selections and enrolls the student in the chosen courses.
	5	Admin and faculty receive notification of the new course enrollments.
Alternative Flows	**Step**	**Branching Action**
	3a	If the desired course is not available, the student contacts admin or faculty for assistance.
	4a	If there is a system error during enrollment, the system alerts the student and admin for further action.
Quality Requirements	**Step**	**Requirement**
	6	The system must always be accessible for course selection.

Table 7: Approve course selection

Use Case 7	Approve Course Selection	
Goal	Academic staff (e.g., advisors, department heads) want to review and approve students' selected courses for the upcoming term.	
Preconditions	Academic staff must be logged into NstuAcademia. Students have submitted their course selections.	
Success End Condition	Selected courses are approved by academic staff.	
Failed End Condition	Course selections are rejected due to prerequisites, capacity, or other reasons.	
Primary Actors: **Secondary Actors:**	Faculty, Administration Admin, Student	
Trigger	Course selection submission by students.	
Main Success Flows	**Step**	**Action**
	1	Academic staff access the course selection approval section.
	2	Academic staff view students' selected courses
	3	Academic staff verify prerequisites, course availability, and capacity.
	4	Academic staff approve or reject course selections.
	5	System updates students' records accordingly.
Alternative Flows	**Step**	**Branching Action**
	3a	Academic staff identify missing prerequisites.
	4a	Academic staff reject specific courses and provide reasons.
Quality Requirements	**Step**	**Requirement**
	6	Real-time updates on attendance records.

Table 8: Input Attendance

Use Case 8	Input Attendance	
Goal	Faculty members want to record student attendance for classes.	
Preconditions	Faculty members must be logged into NstuAcademia.	
Success End Condition	Student attendance is accurately recorded.	
Failed End Condition	Attendance input fails due to system error or incorrect data.	
Primary Actors: **Secondary Actors:**	Teacher, Administration Admin, Student	
Trigger	Faculty member initiates attendance input.	
Main Success Flows	**Step**	**Action**
	1	Teacher selects the relevant class session.
	2	Teacher marks students as present, absent, or late.
	3	System validates and saves the attendance data.
	4	System checks course availability and prerequisites.
	5	System confirms successful attendance input.
Alternative Flows	**Step**	**Branching Action**
	2a	Teacher member encounters an issue (e.g., missing student, technical glitch).
	4a	System prompts faculty member to review and correct any discrepancies.
Quality Requirements	**Step**	**Requirement**
	6	Real-time updates on attendance records.

Table 9: Attendance check

Use Case 9	Attendance Check	
Goal	To accurately record and manage the attendance of students.	
Preconditions	Admin, administration, faculty, and teachers must be logged into the system with appropriate permissions. Students must have access to the attendance system.	
Success End Condition	Attendance is recorded accurately and is accessible for review and reporting.	
Failed End Condition	Attendance is not recorded, is inaccurately recorded, or is inaccessible.	
Primary Actors: **Secondary Actors:**	Admin, Administration, Teacher Student	
Trigger	The attendance recording process is initiated by faculty or teachers.	
Main Success Flows	**Step**	**Action**
	1	Student or teachers log into the attendance system.
	2	They select the class and session for which attendance needs to be recorded.
	3	Teacher marks their attendance
	4	The system records the attendance and updates the database.
	5	Admin and administration review attendance records for accuracy and completeness.
Alternative Flows	**Step**	**Branching Action**
	3a	If a student is unable to mark attendance, teachers can manually input the attendance.
	4a	If there is a discrepancy in attendance records, admin and administration take necessary actions to correct it.
Quality Requirements	**Step**	**Requirement**
	6	The system must ensure that attendance records are precise and reflect actual student presence.

Table 10: Approve course requirements Application

Use Case 10	Approve Course Registration Application	
Goal	Goal: Academic staff (e.g., advisors, department heads) want to review and approve students' selected courses for the upcoming term.	
Preconditions	Academic staff must be logged into NstuAcademia. Students have submitted their course selections.	
Success End Condition	Selected courses are approved by academic staff.	
Failed End Condition	Course selections are rejected due to prerequisites, capacity, or other reasons.	
Primary Actors: **Secondary Actors:**	Teacher, Administration Admin, Student	
Trigger	Course selection submission by students.	
Main Success Flows	**Step**	**Action**
	1	Academic staff access the course selection approval section.
	2	Academic staff view students' selected courses
	3	Academic staff verify prerequisites, course availability, and capacity.
	4	Academic staff approve or reject course selections.
	5	System updates students' records accordingly.
Alternative Flows	**Step**	**Branching Action**
	3a	Academic staff identify missing prerequisites.
	4a	Academic staff reject specific courses and provide reasons.
Quality Requirements	**Step**	**Requirement**
	6	Real-time updates on attendance records.

Table 11: Payment Receive

Use Case 11	Payment Received	
Goal	To accurately process and record payments made by students.	
Preconditions	Admin and administration must have access to the payment system. Teachers must be able to verify payment details if necessary. Students must have made a payment.	
Success End Condition	Payments are correctly processed, and records are updated to reflect the received payment.	
Failed End Condition	Payments are not processed, incorrectly recorded, or students are not acknowledged for their payment.	
Primary Actors: **Secondary Actors:**	Admin, Administration Teacher, Student	
Trigger	Student makes a payment.	
Main Success Flows	**Step**	**Action**
	1	Student completes the payment process through the online portal.
	2	System notifies the admin and administration of the received payment.
	3	Admin and administration verify the payment details and amount.
	4	System updates the student's financial records to show the payment has been received.
	5	Admin and administration confirm the payment receipt and notify the student.
Alternative Flows	**Step**	**Branching Action**
	3a	If the payment details are incorrect, admin and administration request the student to correct the information.
	4a	If there is a discrepancy in the amount received, admin and administration investigate and resolve the issue.
Quality Requirements	**Step**	**Requirement**
	6	The system must ensure that payment amounts and details are recorded correctly
	7	The system should process payments and update records without unnecessary delays.

Table 12: Payment

Use Case 12	Payment	
Goal	Users want to make payments for various services (e.g., fees, fines, purchases) through NstuAcademia.	
Preconditions	Student must be logged into NstuAcademia during the course registration period.	
Success End Condition	Payment is successfully processed.	
Failed End Condition	Payment fails due to insufficient funds, invalid payment details, or system error.	
Primary Actors: **Secondary Actors:**	Students, Administration, Hall Staffs Admin	
Trigger	User initiates a payment request.	
Main Success Flows	**Step**	**Action**
	1	User selects the payment option
	2	User enters payment details (e.g., amount, purpose, payment method).
	3	System validates payment details.
	4	System processes the payment transaction.
	5	System confirms successful payment to the user.
Alternative Flows	**Step**	**Branching Action**
	3a	User enters an invalid payment amount.
	4a	System displays an error message and prompts the user to correct the payment details.
Quality Requirements	**Step**	**Requirement**
	6	Payment confirmation must be sent immediately after successful transaction.

Table 13: Receive Confirmation

Use Case 13	Choose Courses	
Goal	To enable students to select courses for their academic term.	
Preconditions	Students must be registered in the system. Admin and faculty must have updated the system with the available courses.	
Success End Condition	Students successfully select their courses and are enrolled in them for the academic term.	
Failed End Condition	Students are unable to select or enroll in courses due to system errors or unavailability of options.	
Primary Actors: **Secondary Actors:**	Student Teacher, Admin	
Trigger	Students initiate the course selection process.	
Main Success Flows	**Step**	**Action**
	1	Student logs into the academic portal.
	2	Student navigates to the course selection section.
	3	Student reviews the available courses and selects the desired ones.
	4	System confirms the selections and enrolls the student in the chosen courses.
	5	Admin and faculty receive notification of the new course enrollments.
Alternative Flows	**Step**	**Branching Action**
	3a	If the desired course is not available, the student contacts admin or faculty for assistance.
	4a	If there is a system error during enrollment, the system alerts the student and admin for further action.
Quality Requirements	**Step**	**Requirement**
	6	The system must always be accessible for course selection.

Table 14: Issue Admit card

Use Case 14	Issue Admit Card	
Goal	To provide students with an Admit Card for their examinations.	
Preconditions	Students must have completed any prerequisites and be eligible for the examination. Admin and administration must have updated the system with the examination details. Teachers must have submitted any necessary approvals	
Success End Condition	Students receive an Admit Card with accurate examination details.	
Failed End Condition	Students do not receive an Admit Card or receive incorrect examination information.	
Primary Actors: **Secondary Actors:**	Admin, Administration Teacher, Student	
Trigger	Student requests an Admit Card.	
Main Success Flows	**Step**	**Action**
	1	Student logs into the examination portal.
	2	Student requests an Admit Card for the upcoming examination.
	3	System notifies the admin and administration of the request.
	4	Admin and administration review the student's eligibility and examination details.
	5	Admin and administration approve the request and issue the Admit Card.
	6	System generates the Admit Card and provides it to the student.
Alternative Flows	**Step**	**Branching Action**
	4a	If the student is not eligible for the examination, the admin and administration informs the student of the reasons.
	5a	If there is an error in generating the Admit Card, the system alerts the admin, and they take corrective action.
Quality Requirements	**Step**	**Requirement**
	7	The Admit Card must display the correct examination details and student information.

Table 15: Communication

Use Case 15	Communication	
Goal	Stakeholders want to effectively communicate requirements, changes, and progress related to NstuAcademia.	
Preconditions	Stakeholders (e.g., developers, project managers, users) are involved in the project.	
Success End Condition	Clear communication channels are established.	
Failed End Condition	Miscommunication leads to misunderstandings or delays.	
Primary Actors: **Secondary Actors:**	Students, Administration, Teacher Admin	
Trigger	Streamline and unify communication channels for efficient message management across platforms	
Main Success Flows	**Step**	**Action**
	1	Seamlessly integrates various communication channels (such as emails, messages, and notifications) into a centralized platform.
	2	Users can efficiently manage their messages, emails, and notifications from a single interface, allowing for quick responses and organized conversations.
	3	Users can customize their communication preferences and personalize their interface, enabling them to prioritize important conversations and filter out noise.
	4	The platform facilitates effective collaboration by enabling users to share files, collaborate on documents in real-time, schedule meetings, and assign tasks within the communication interface.
	5	System confirms successful registration to the student.
Alternative Flows	**Step**	**Branching Action**
	3a	Urgent changes require ad-hoc communication outside regular meetings
	7a	Stakeholders escalate critical issues through designated channels (e.g., urgent emails, direct calls).
Quality Requirements	**Step**	**Requirement**
	6	Communication must be timely, concise, and documented to ensure transparency and alignment among stakeholders.

Table 16: Apply for Hall Seat

Use Case 16	Apply for Hall Seat	
Goal	Students want to apply for a seat in the university hall of residence.	
Preconditions	Student must be eligible for hall accommodation (e.g., enrolled in a program, meeting criteria). Hall seat application period is open.	
Success End Condition	Student successfully secures a seat in the hall.	
Failed End Condition	No seats are available, or the application is rejected.	
Primary Actors: **Secondary Actors:**	Students, Administration Admin, Hall Staff	
Trigger	Hall seat application period begins.	
Main Success Flows	**Step**	**Action**
	1	Student logs into NstuAcademia.
	2	Student navigates to the hall seat application section.
	3	Student selects the desired hall and room type.
	4	Student submits the application.
	5	System confirms successful application and provides relevant details (e.g., room assignment, move-in date).
Alternative Flows	**Step**	**Branching Action**
	3a	Student selects a hall or room type that is already full.
	5a	System notifies the student of alternative options or places them on a waiting list.
Quality Requirements	**Step**	**Requirement**
	6	Real-time updates on seat availability during the application process.

Table 17: Approve hall application

Use Case 17	Approve hall application	
Goal	To manage and approve applications for hall reservations by students.	
Preconditions	Admin and administration must be logged into the system with appropriate permissions. Teachers must have submitted any necessary recommendations. Students must have completed the hall application form	
Success End Condition	Hall applications are reviewed, approved, and confirmed in the system.	
Failed End Condition	Hall applications are not approved due to incomplete information, lack of availability, or other criteria not being met.	
Primary Actors: **Secondary Actors:**	Admin, Administration Teacher, Student	
Trigger	Student submits a hall application form.	
Main Success Flows	**Step**	**Action**
	1	Student submits the hall application form through the system.
	2	System notifies admin and administration of the new application.
	3	Admin and administration review the application details and teacher recommendations.
	4	Admin and administration approve the hall application.
	5	System updates the hall reservation status and notifies the student.
Alternative Flows	**Step**	**Branching Action**
	3a	If the application is incomplete, admin and administration request additional information from the student.
	3a	If no halls are available, admin and administration inform the student and suggest alternative options.
Quality Requirements	**Step**	**Requirement**
	7	The system must ensure that all application details are correctly recorded and processed.

Table 18: Certificate submission (Boarding card)

Use Case 18	Certificate Submission (Boarding card)	
Goal	To manage and verify the submission of certificates by students.	
Preconditions	Admin and faculty must be logged into the system with appropriate permissions. Students must have access to the submission portal.	
Success End Condition	Certificates are submitted, verified, and recorded in the system accurately.	
Failed End Condition	Certificates are not submitted, cannot be verified, or are inaccurately recorded.	
Primary Actors: **Secondary Actors:**	Hall Saff, Student, Teacher Admin	
Trigger	Student initiates the certificate submission process.	
Main Success Flows	**Step**	**Action**
	1	Student logs into the certificate submission portal.
	2	Student uploads the required certificate(s).
	3	System notifies the admin and teacher of the new submission.
	4	Admin and teacher review and verify the certificate(s).
	5	System records the verified certificate(s) and updates the student's profile.
Alternative Flows	**Step**	**Branching Action**
	2a	If the student uploads an incorrect or incomplete certificate, the system alerts the student to resubmit.
	4a	If the admin or faculty cannot verify the certificate, they request additional information or a new submission from the student.
Quality Requirements	**Step**	**Requirement**
	6	The system must accurately record and reflect the status of certificate submissions.

Table 19: Empty seat display

Use Case 19	Manage Empty Seat Display	
Goal	To accurately display the status of empty seats for an event in a hall.	
Preconditions	Admin must be logged into the system with appropriate	
Success End Condition	The event and seat configuration must be set up in the system.	
Failed End Condition	The system displays incorrect information, leading to overbooking or customer dissatisfaction.	
Primary Actors: **Secondary Actors:**	Admin Administration, Student	
Trigger	Changes in seat reservation status.	
Main Success Flows	**Step**	**Action**
	1	Admin logs into the hall seat application management module
	2	Admin reviews the status of seat reservations.
	3	Admin updates the system with any changes in seat availability.
	4	System processes the updates and displays the current empty seats.
	5	Users view the updated empty seat display when selecting seats.
Alternative Flows	**Step**	**Branching Action**
	3a	If a seat becomes available, the admin immediately updates the system to reflect the change.
	4a	If there is a system error in displaying seat availability, the admin troubleshoots and resolves the issue.
Quality Requirements	**Step**	**Requirement**
	7	The system must display the correct number of empty seats available in real-time.
	8	Updates to seat availability must be reflected in the system without delay.

Table 20: Oversee Hall Application

Use Case 20	Issue Boarding Card	
Goal	To provide passengers with a boarding card for their flight.	
Preconditions	Passenger must have a confirmed flight booking and have checked in for them flight.	
Success End Condition	Passenger receives a boarding card with accurate flight and seat information.	
Failed End Condition	Passenger does not receive a boarding card or receives incorrect flight information.	
Primary Actors: **Secondary Actors:**	Student, Teacher, Hall Staff Admin, Administration	
Trigger	Initiates the boarding card issuance process.	
Main Success Flows	**Step**	**Action**
	1	Passenger approaches the check-in counter or uses an online check-in system.
	2	Check-in staff verifies the passenger's booking and identification.
	3	System generates the boarding card with the passenger's flight and seat details.
	4	Check-in staff provides the boarding card to the passenger.
	5	Hall staff confirms the details on the boarding card.
	6	System sends a confirmation message the user.
Alternative Flows	**Step**	**Branching Action**
	2a	If the passenger's booking cannot be verified, the check-in staff assists the passenger to resolve the issue.
	3a	If there is an error in generating the boarding card, the system alerts the check-in staff, who then take corrective action.
Quality Requirements	**Step**	**Requirement**
	7	The boarding card must display the correct flight and seat information.

Table 21: Issue Boarding Card

Use Case 21	Issue Boarding Card	
Goal	To provide passengers with a boarding card for their flight.	
Preconditions	Passenger must have a confirmed flight booking and have checked in for them flight.	
Success End Condition	Passenger receives a boarding card with accurate flight and seat information.	
Failed End Condition	Passenger does not receive a boarding card or receives incorrect flight information.	
Primary Actors: **Secondary Actors:**	Student, Teacher, Hall Staff Admin, Administration	
Trigger	Initiates the boarding card issuance process.	
Main Success Flows	**Step**	**Action**
	1	Passenger approaches the check-in counter or uses an online check-in system.
	2	Check-in staff verifies the passenger's booking and identification.
	3	System generates the boarding card with the passenger's flight and seat details.
	4	Check-in staff provides the boarding card to the passenger.
	5	Hall staff confirms the details on the boarding card.
	6	System sends a confirmation message the user.
Alternative Flows	**Step**	**Branching Action**
	2a	If the passenger's booking cannot be verified, the check-in staff assists the passenger to resolve the issue.
	3a	If there is an error in generating the boarding card, the system alerts the check-in staff, who then take corrective action.
Quality Requirements	**Step**	**Requirement**
	7	The boarding card must display the correct flight and seat information.

Table 22: Verify application

Use Case 22	Verify application	
Goal	To enable students to select courses for their academic term.	
Preconditions	Students must be registered in the system. Admin and faculty must have updated the system with the available courses.	
Success End Condition	Students successfully select their courses and are enrolled in them for the academic term.	
Failed End Condition	Students are unable to select or enroll in courses due to system errors or unavailability of options.	
Primary Actors: **Secondary Actors:**	Student Teacher, Admin	
Trigger	Students initiate the course selection process.	
Main Success Flows	**Step**	**Action**
	1	Student logs into the academic portal.
	2	Student navigates to the course selection section.
	3	Student reviews the available courses and selects the desired ones.
	4	System confirms the selections and enrolls the student in the chosen courses.
	5	Admin and faculty receive notification of the new course enrollments.
Alternative Flows	**Step**	**Branching Action**
	3a	If the desired course is not available, the student contacts admin or faculty for assistance.
	4a	If there is a system error during enrollment, the system alerts the student and admin for further action.
Quality Requirements	**Step**	**Requirement**
	6	The system must always be accessible for course selection.

8. Sequence Diagram

A sequence diagram in UML (Unified Modeling Language) is a type of interaction diagram that shows how objects interact with each other and in what order. It's a way of visualizing the sequence of messages exchanged between objects to carry out a function or a process.

8.1 LogIn

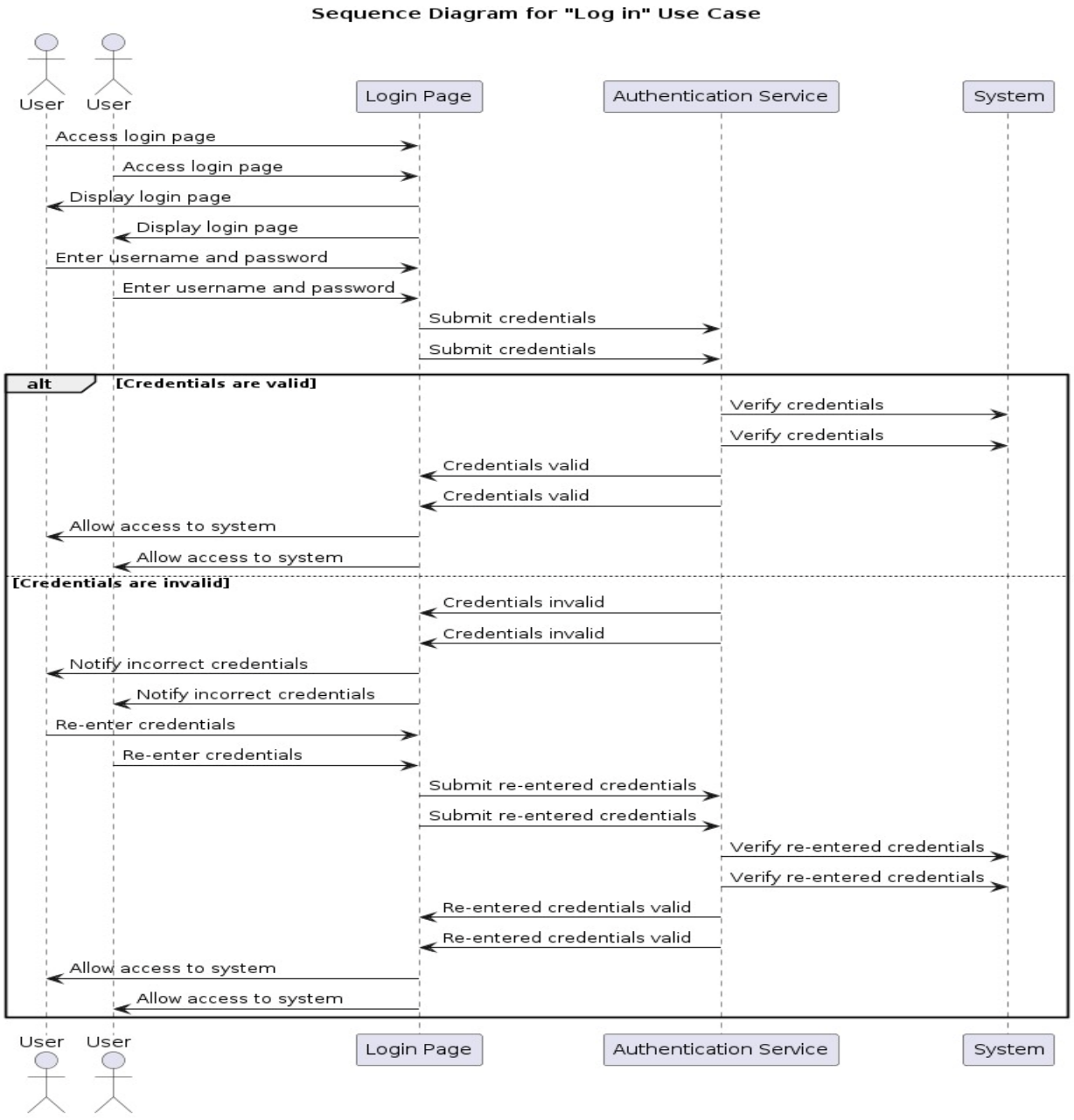

Fig-1: Sequence diagram for NSTUAcademia logIn

8.2 Sign Up

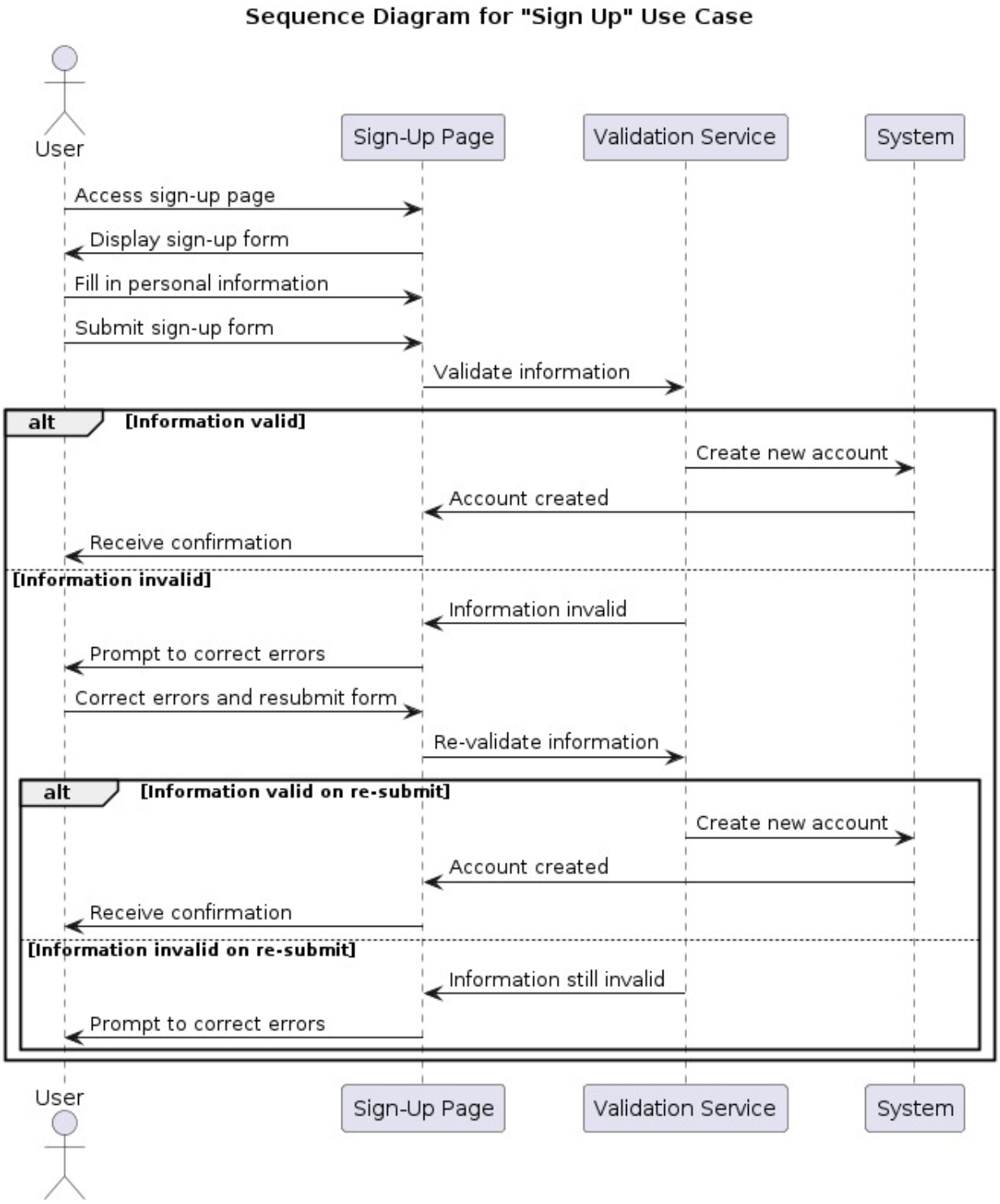

Fig-2: Sequence diagram for NSTUAcademia SignUp

8.3 Manage Profile:

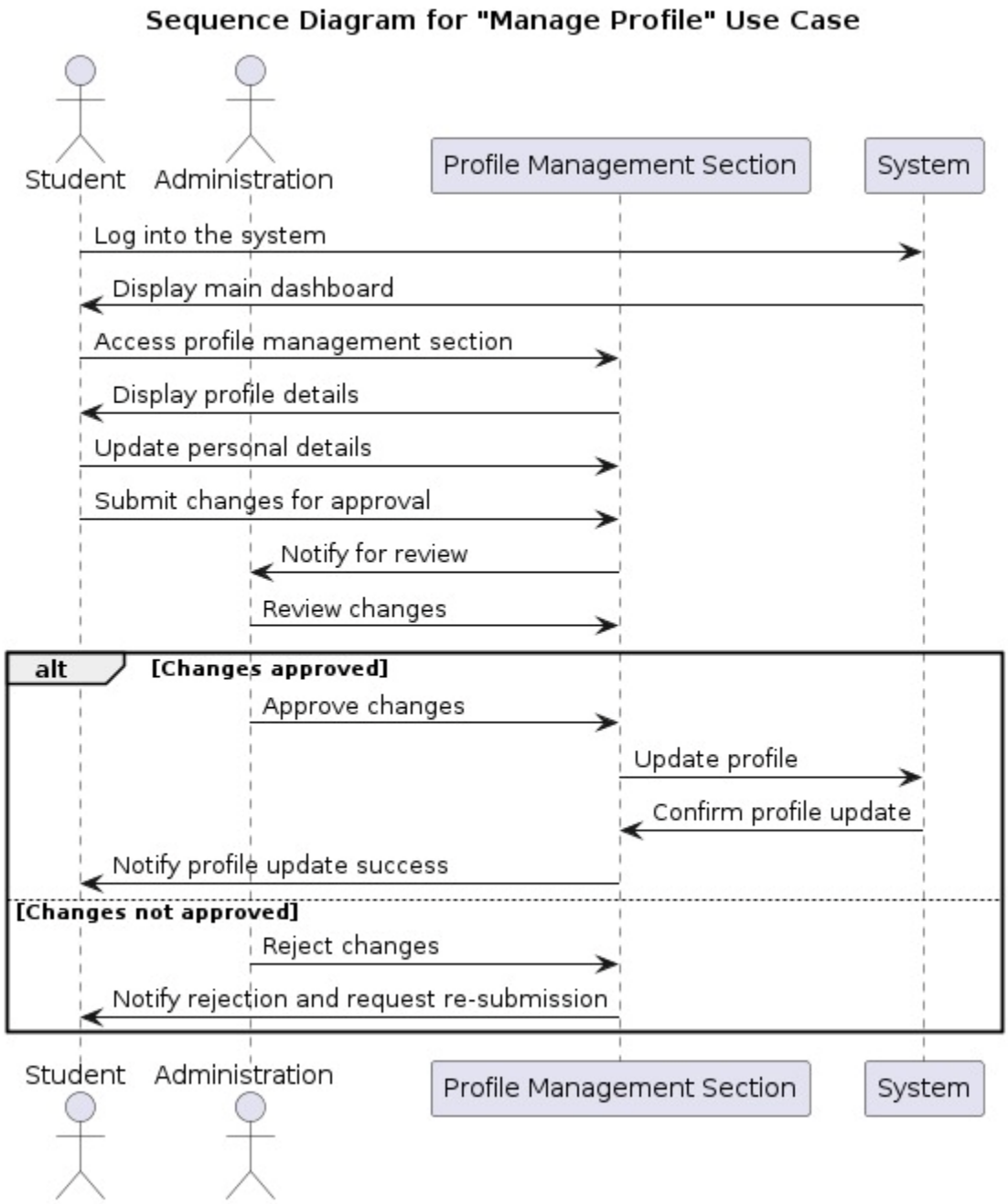

Fig-3: Sequence diagram for NSTUAcademia Manage Profile

8.4 Select Semester

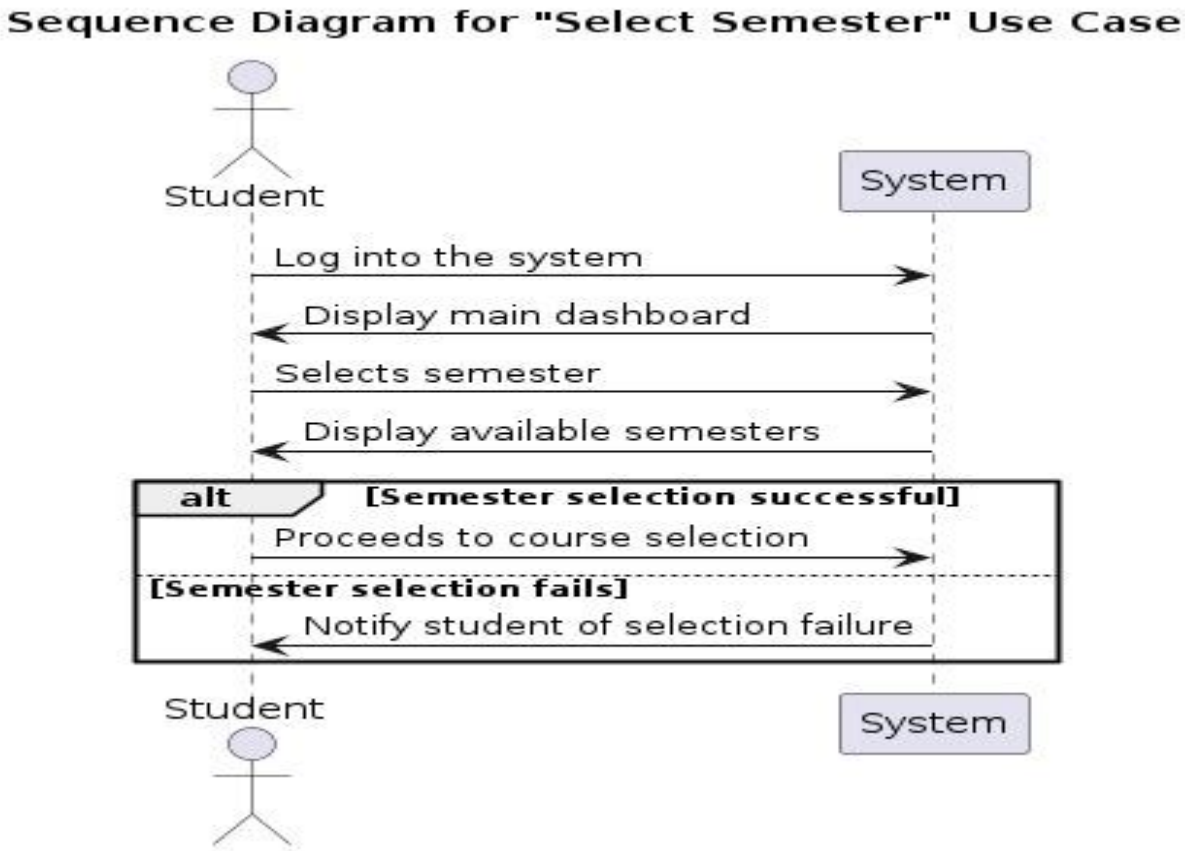

Fig-4: Sequence diagram for NSTUAcademia Select Semester

8.5 Course registration

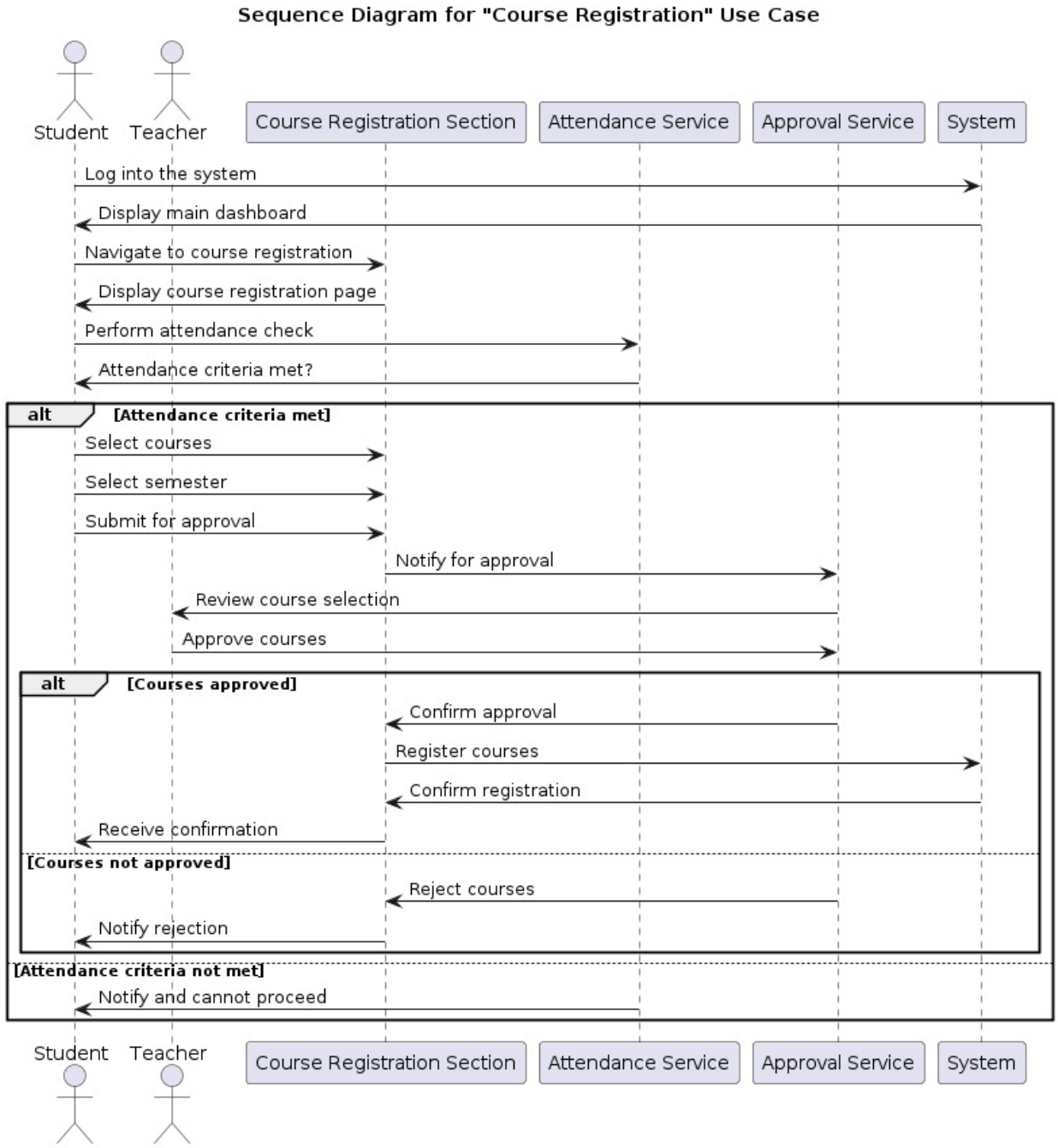

Fig-5: Sequence diagram for NSTUAcademia Course Registration

8.6 Choose Course

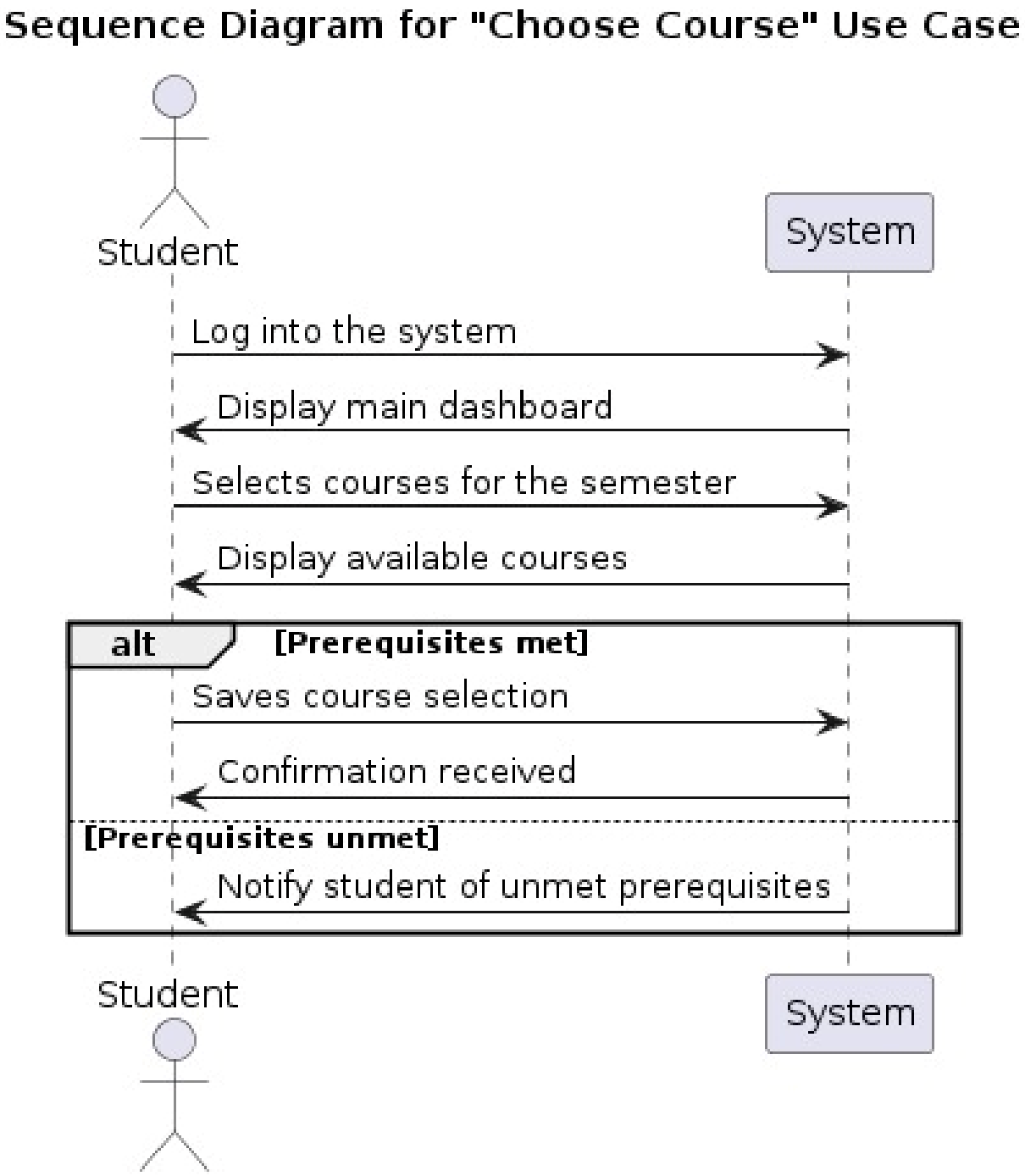

Fig-6: Sequence diagram for NSTUAcademia Choose Course

8.7 Approve Course Selection

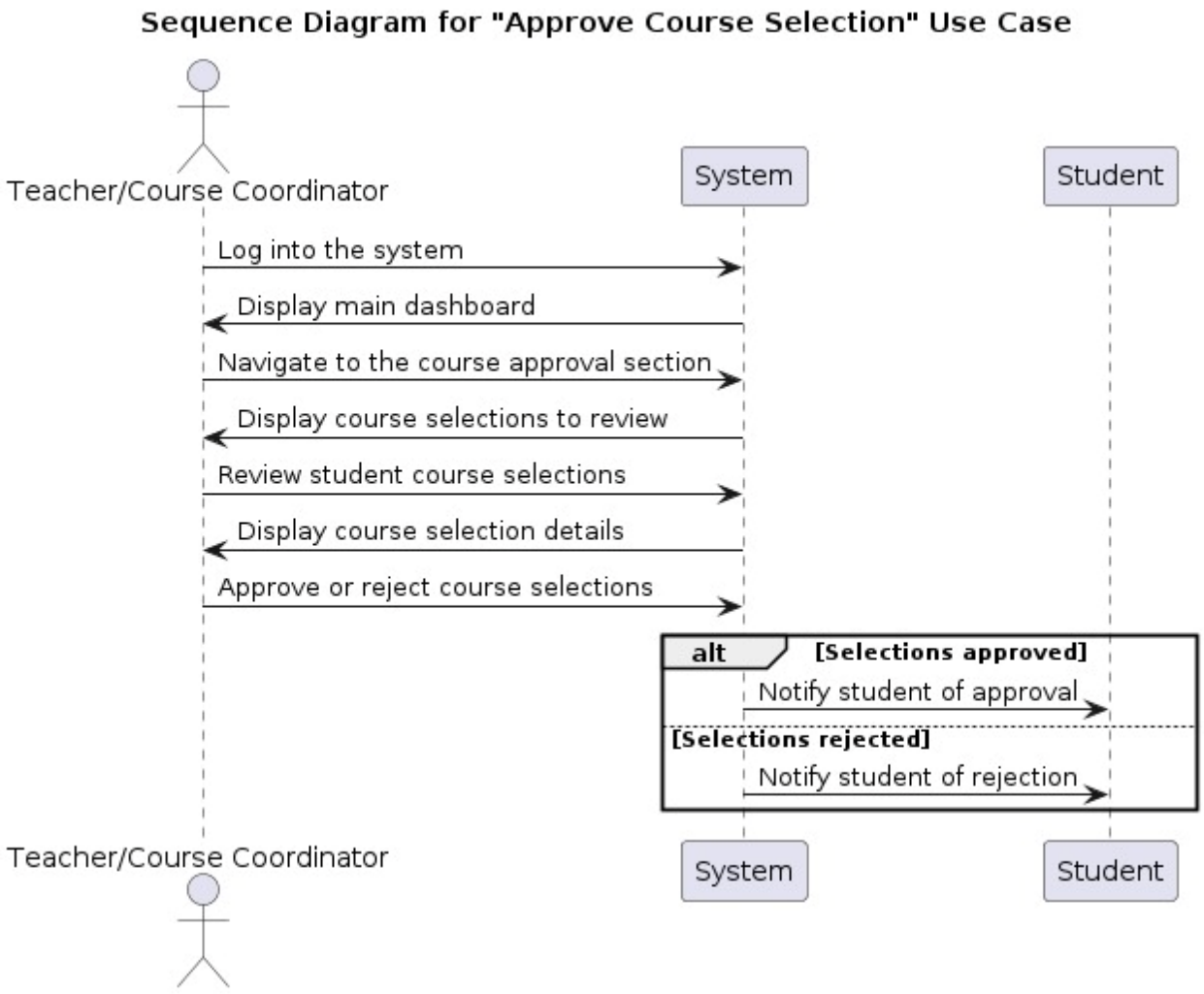

Fig-7: Sequence diagram for NSTUAcademia Approvr Course Selection

8.8 Input Attendance

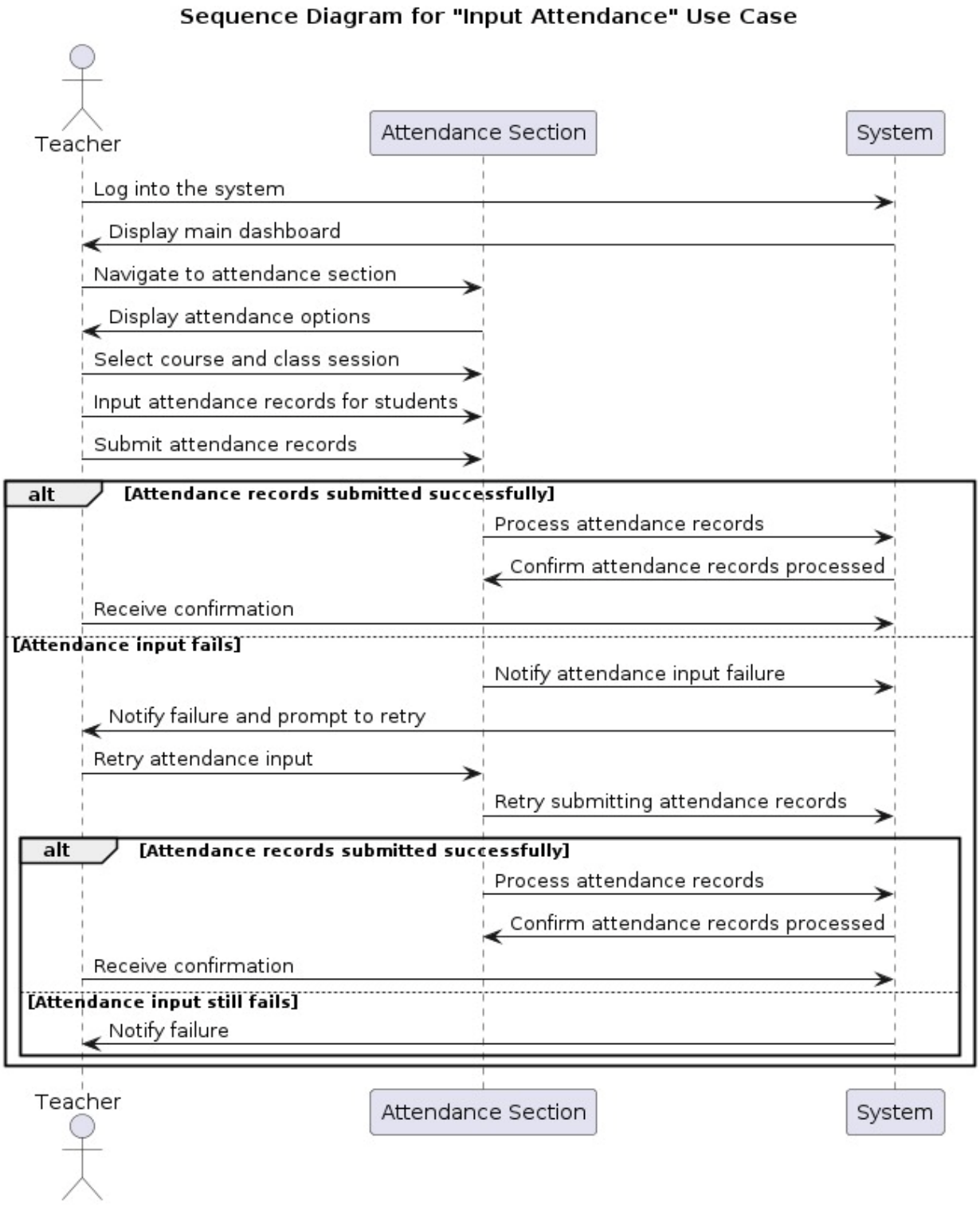

Fig-8: Sequence diagram for NSTUAcademia Input Attendence

8.9 Attendance Check

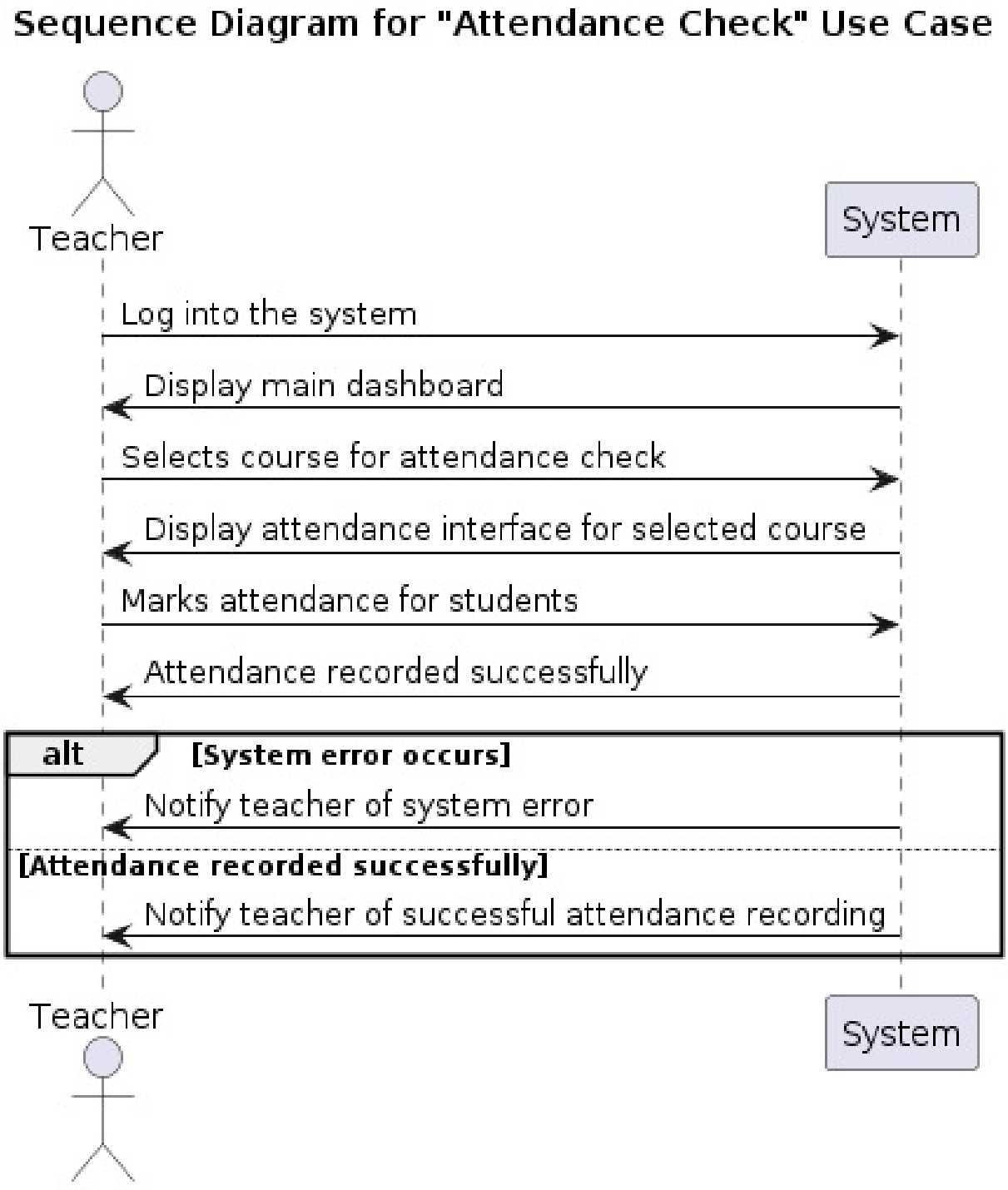

Fig-9: Sequence diagram for NSTUAcademia Attendence Check

8.10 Approve Course Registration Application

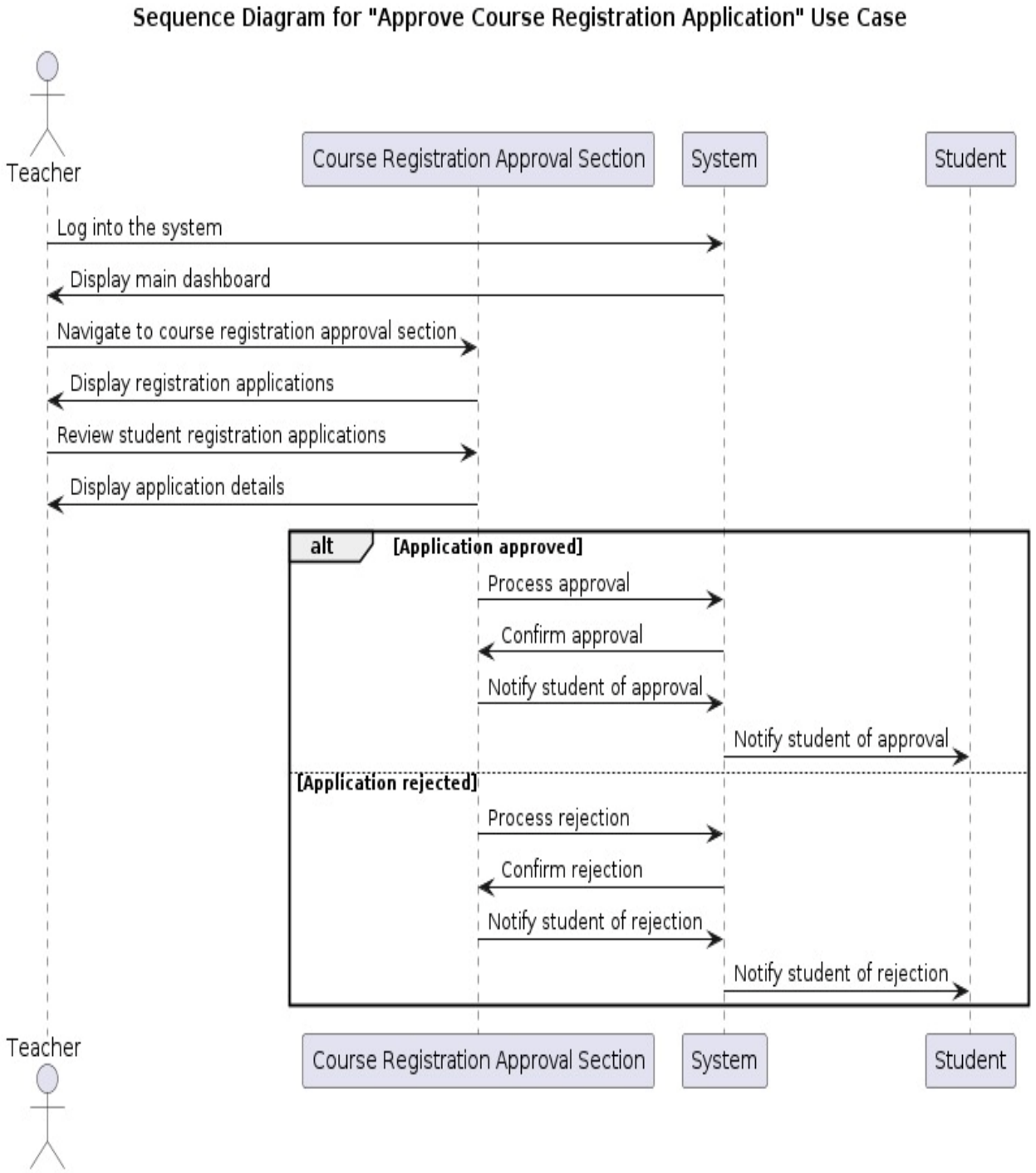

Fig-10: Sequence diagram for NSTUAcademia Approve Course Registration Application

8.11 Payment Receive

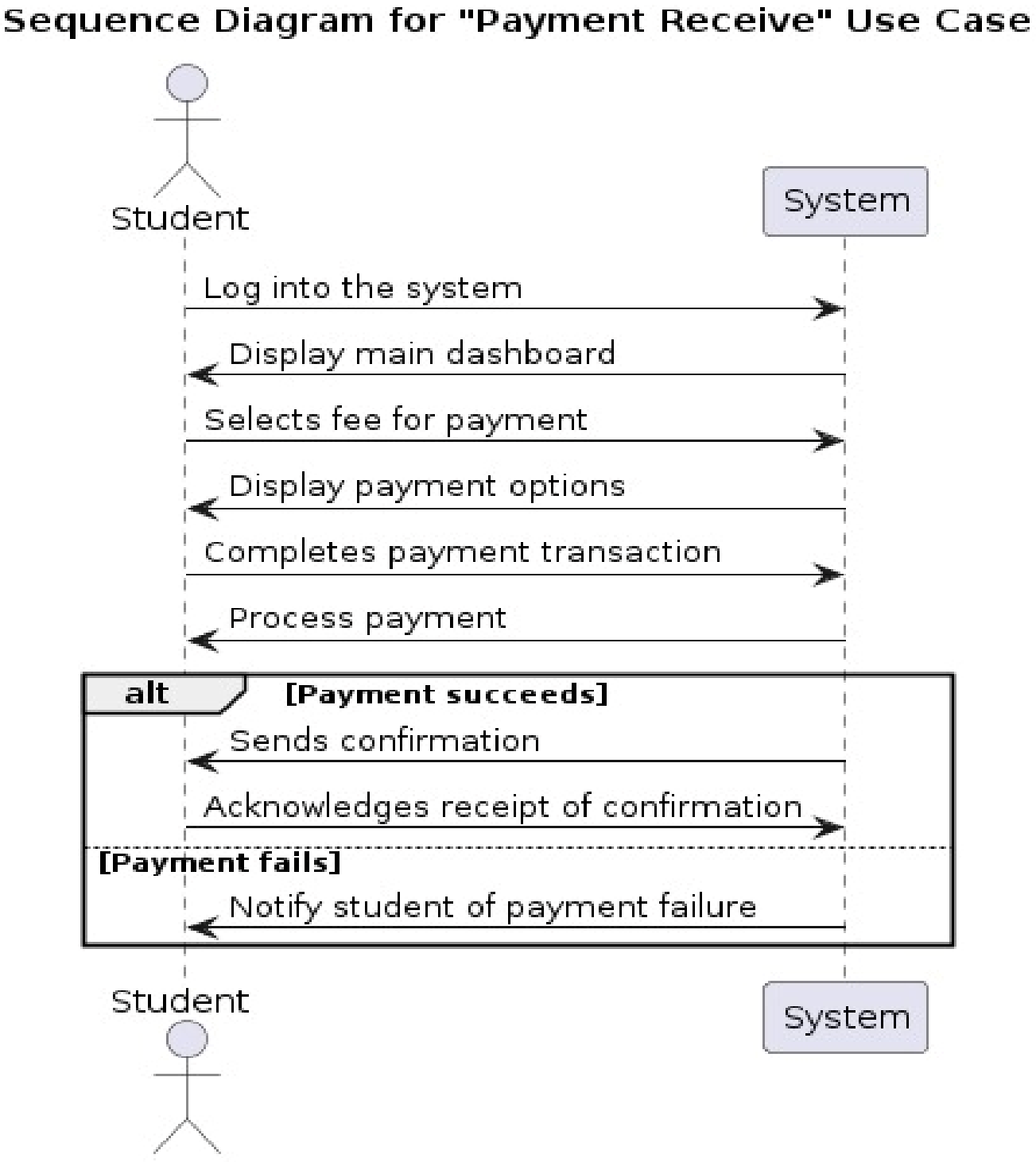

Fig-11: Sequence diagram for NSTUAcademia Payment Receive

8.12 Payment

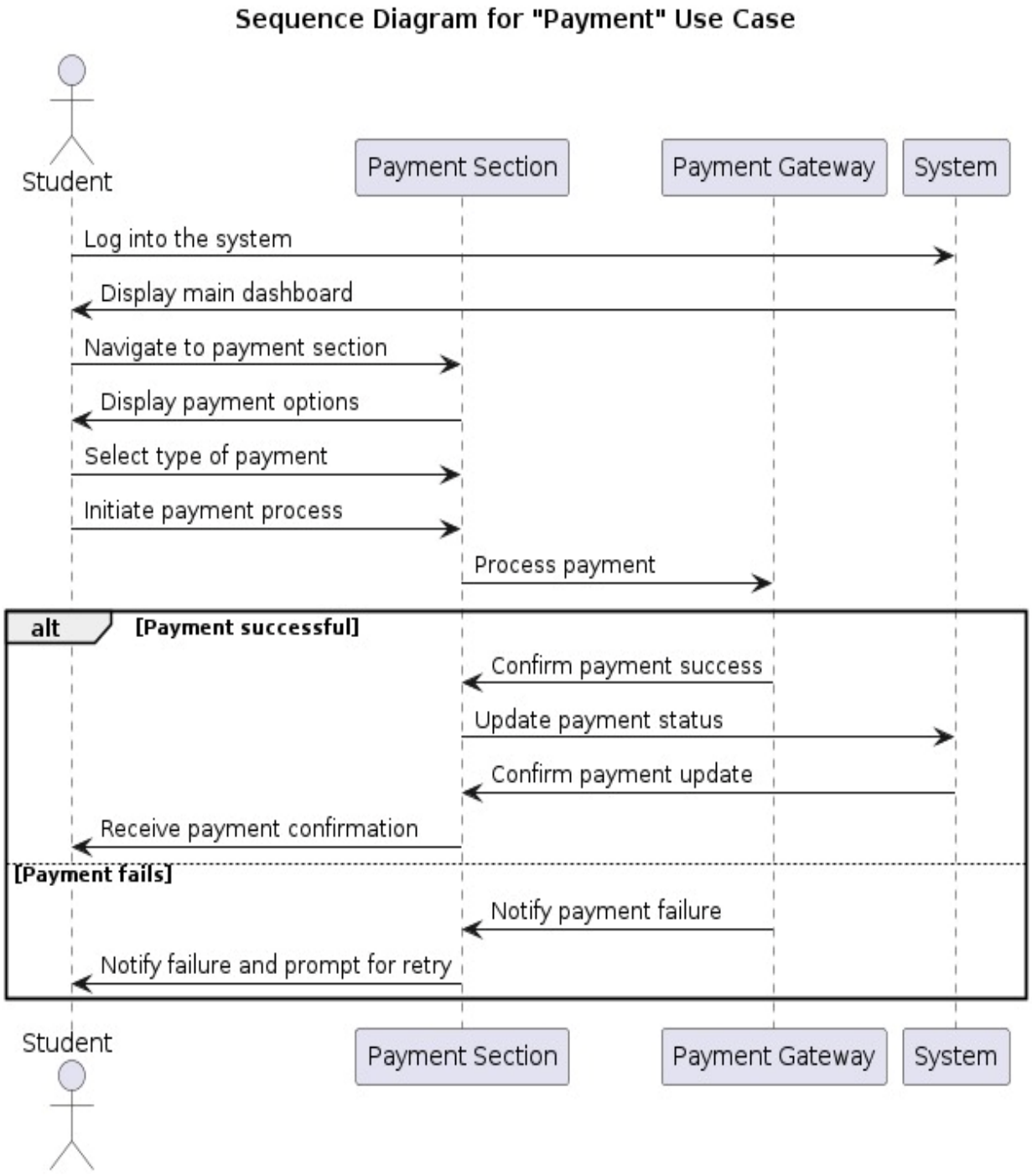

Fig-12: Sequence diagram for NSTUAcademia Payment

8.13 Receive Confirmation

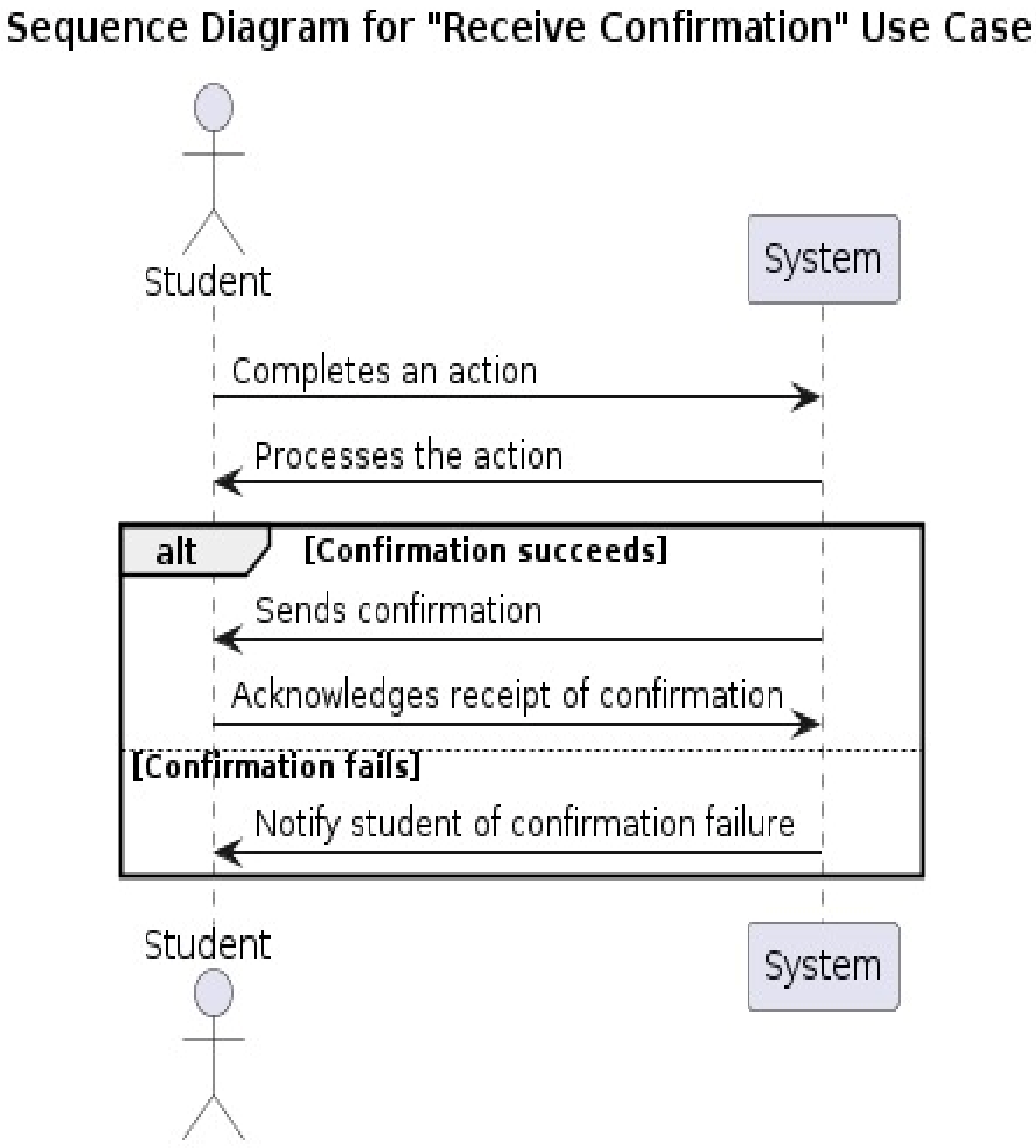

Fig-13: Sequence diagram for NSTUAcademia Receive Confirmation

8.14 Issue Admit Card

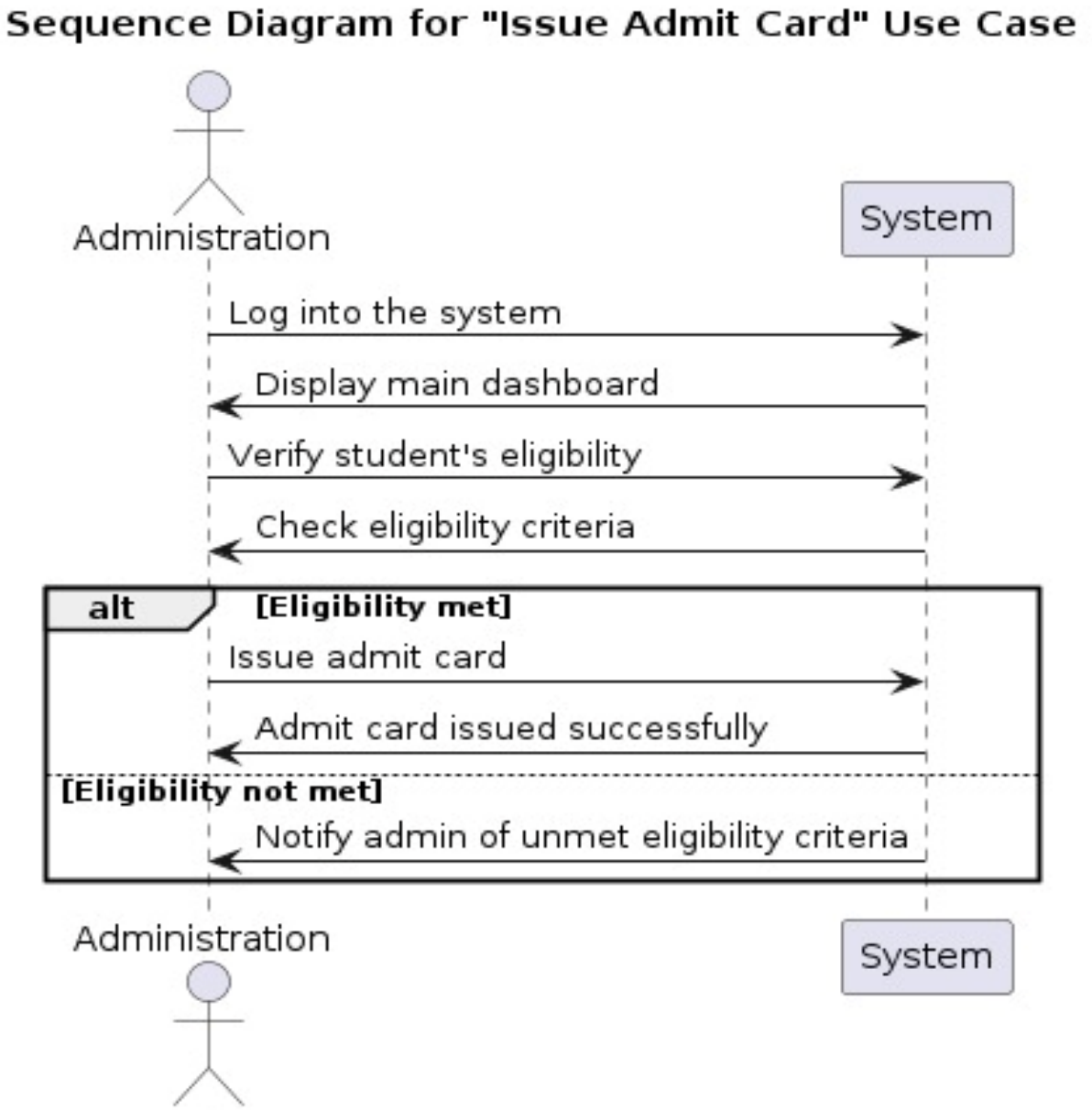

Fig-14: Sequence diagram for NSTUAcademia Issue Admit Card

8.15 Communication

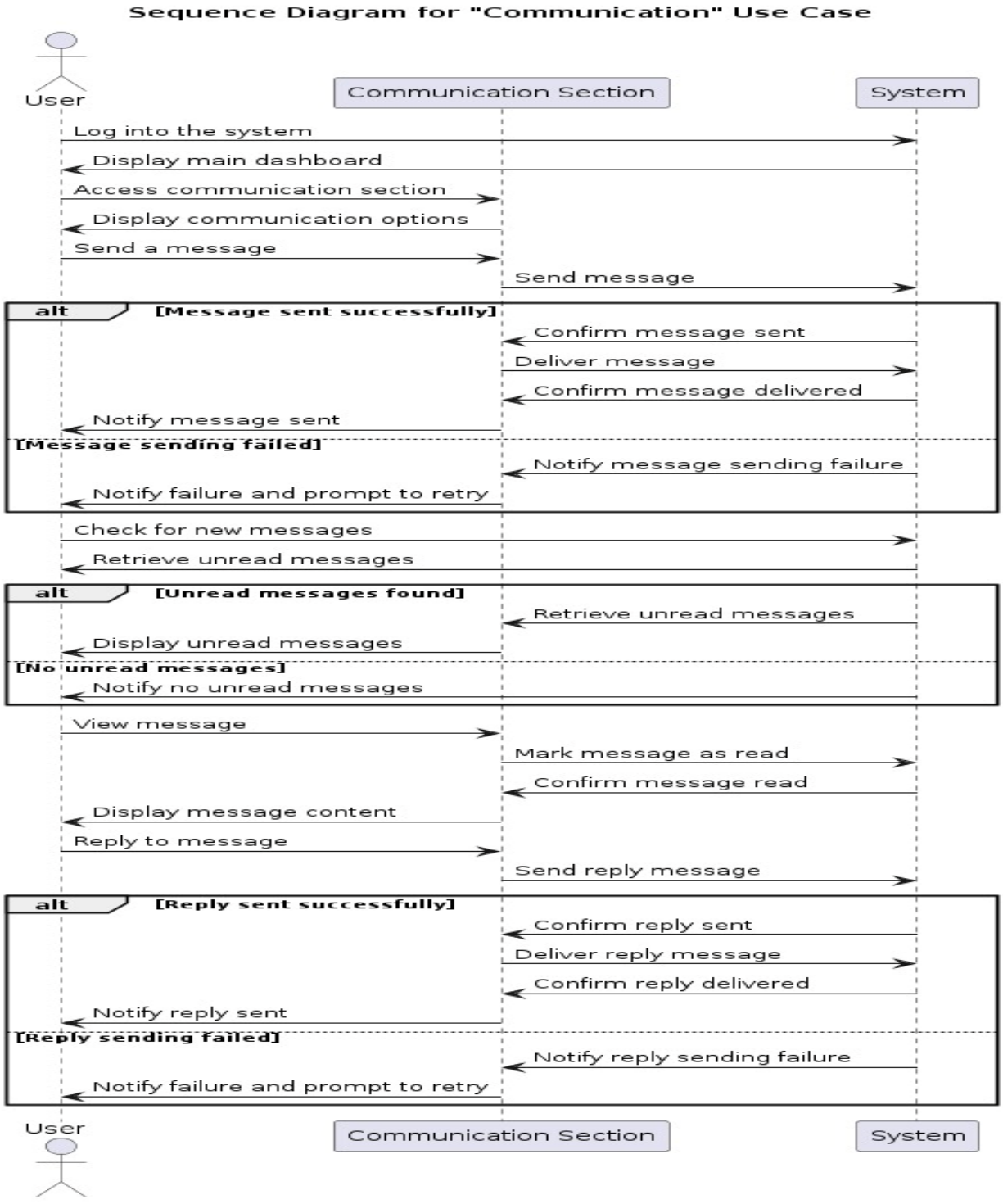

Fig-15: Sequence diagram for NSTUAcademia Communication

8.16 Hall Seat Apply

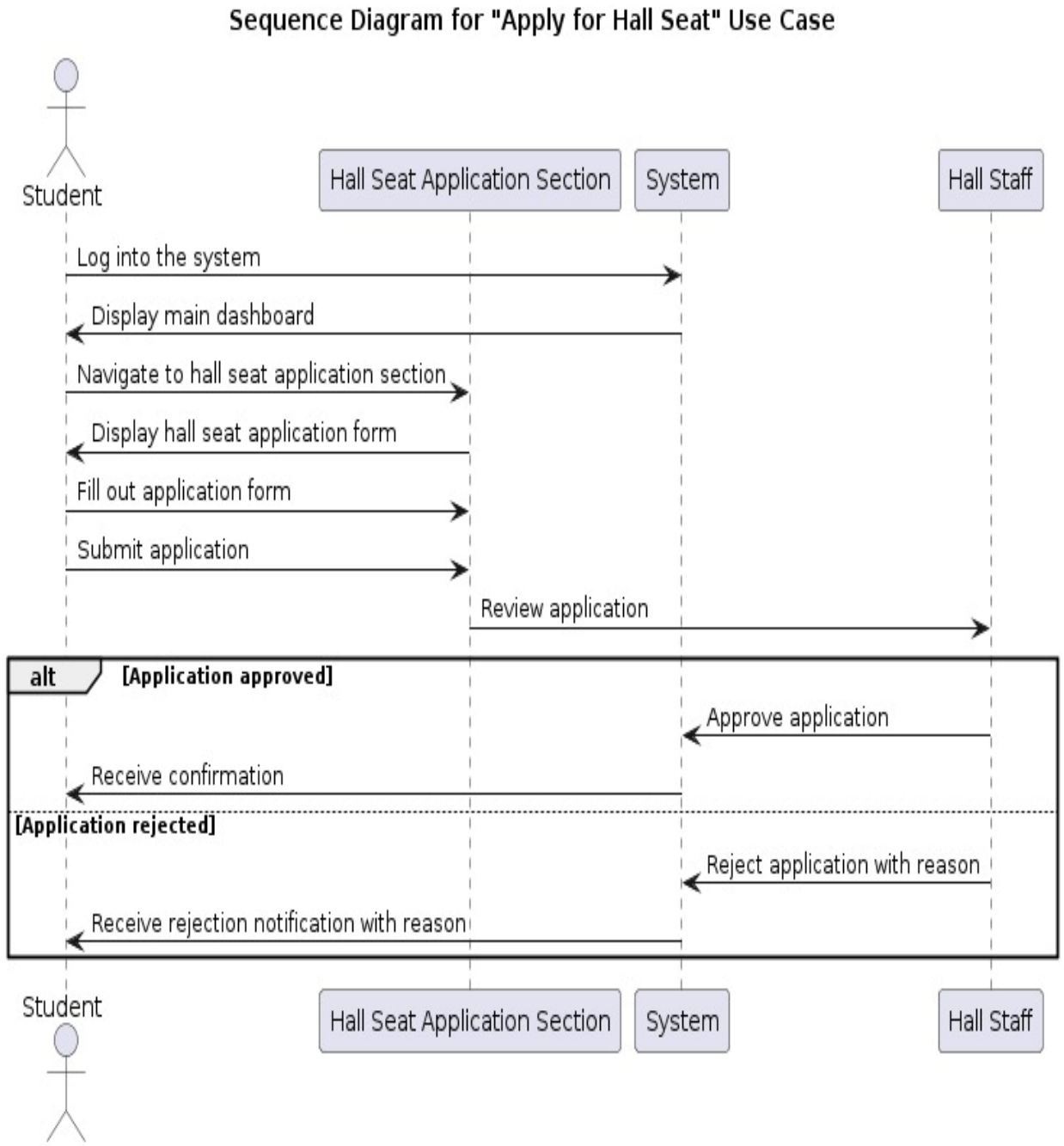

Fig-16: Sequence diagram for NSTUAcademia Hall Seat Apply

8.17 Approve Hall Seat Application

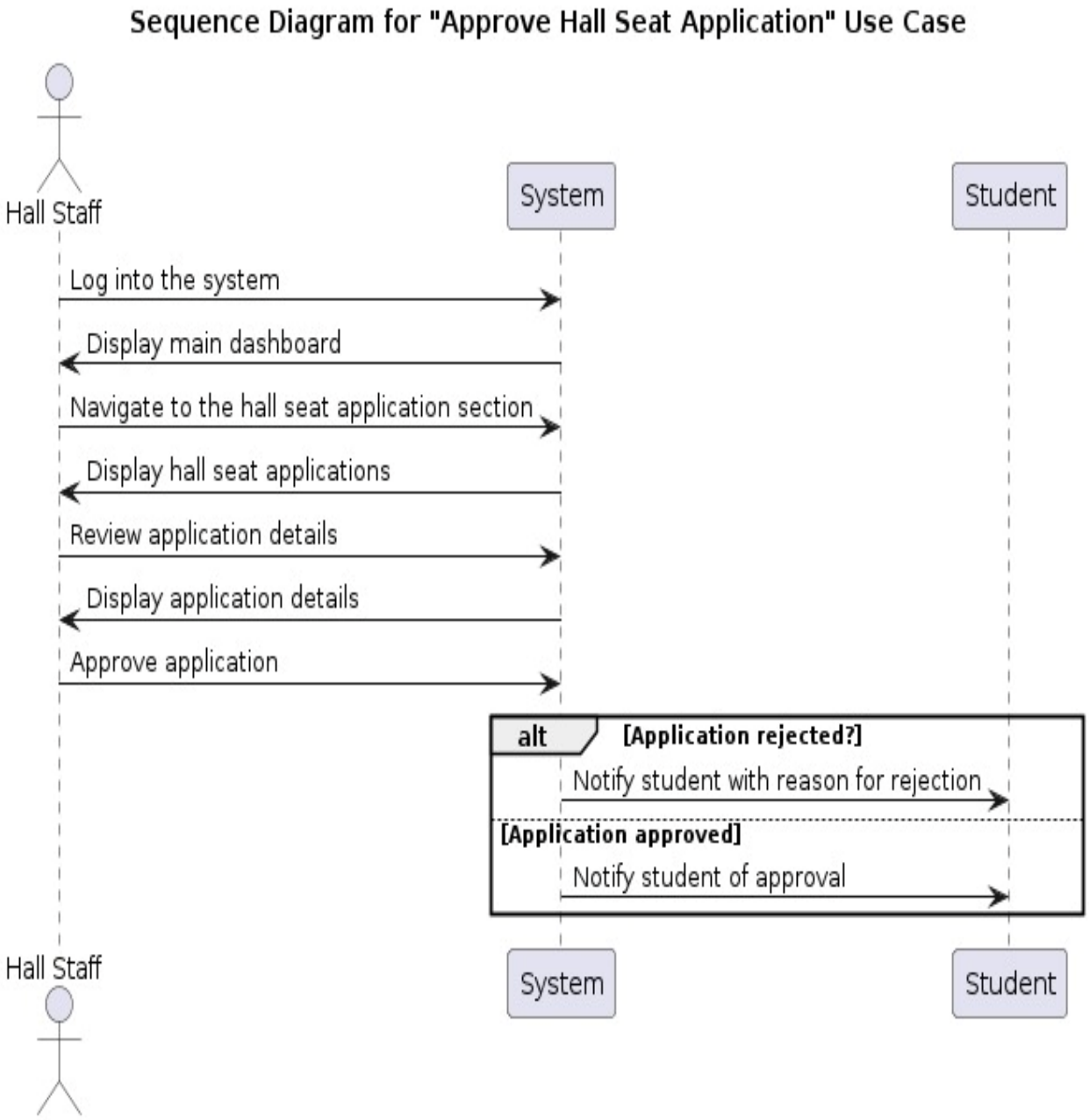

Fig-17: Sequence diagram for NSTUAcademia Approve Hall Seat Application

8.18 Certificate Submission

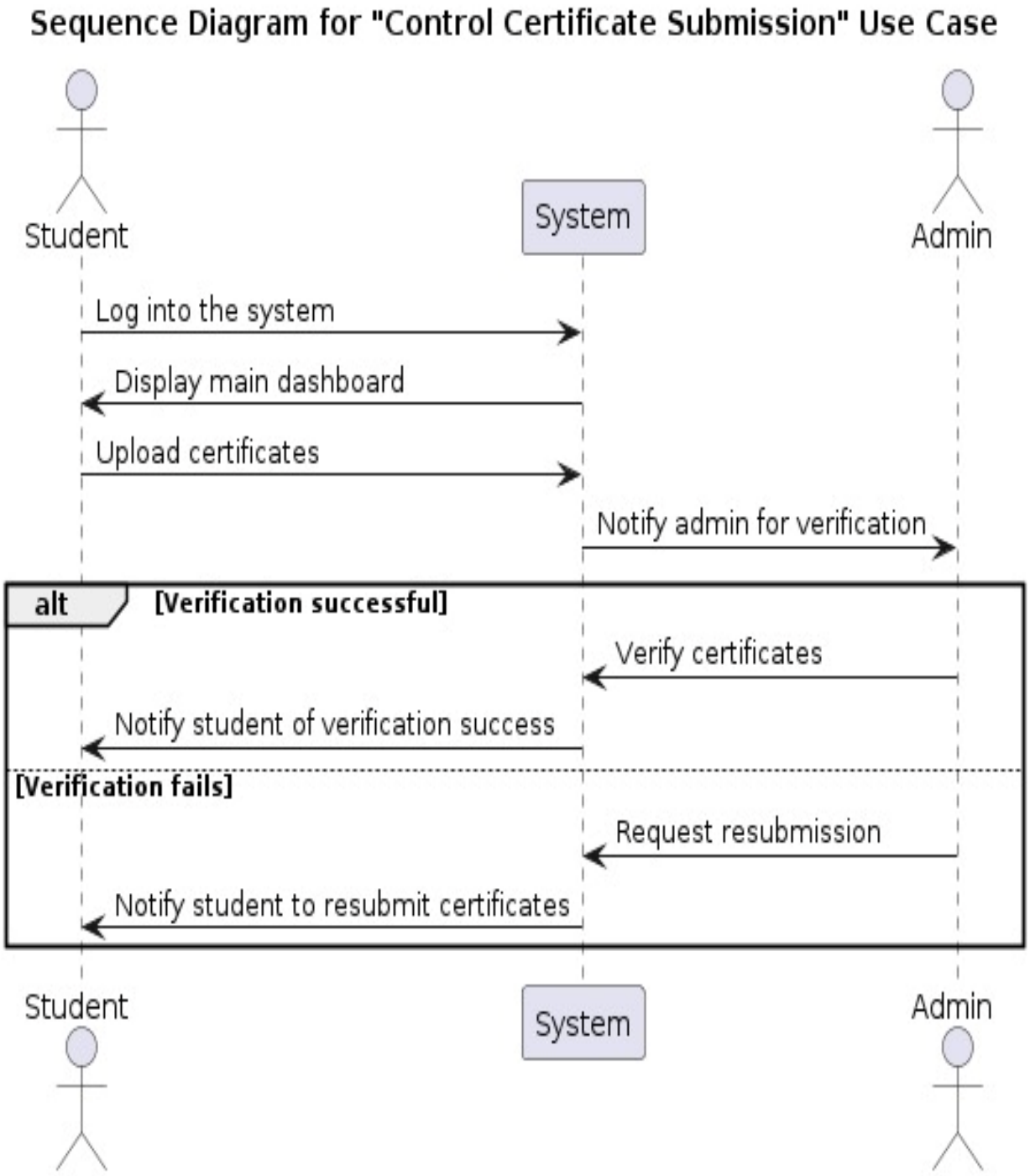

Fig-18: Sequence diagram for NSTUAcademia certificate submission

8.19 Empty Seat Display

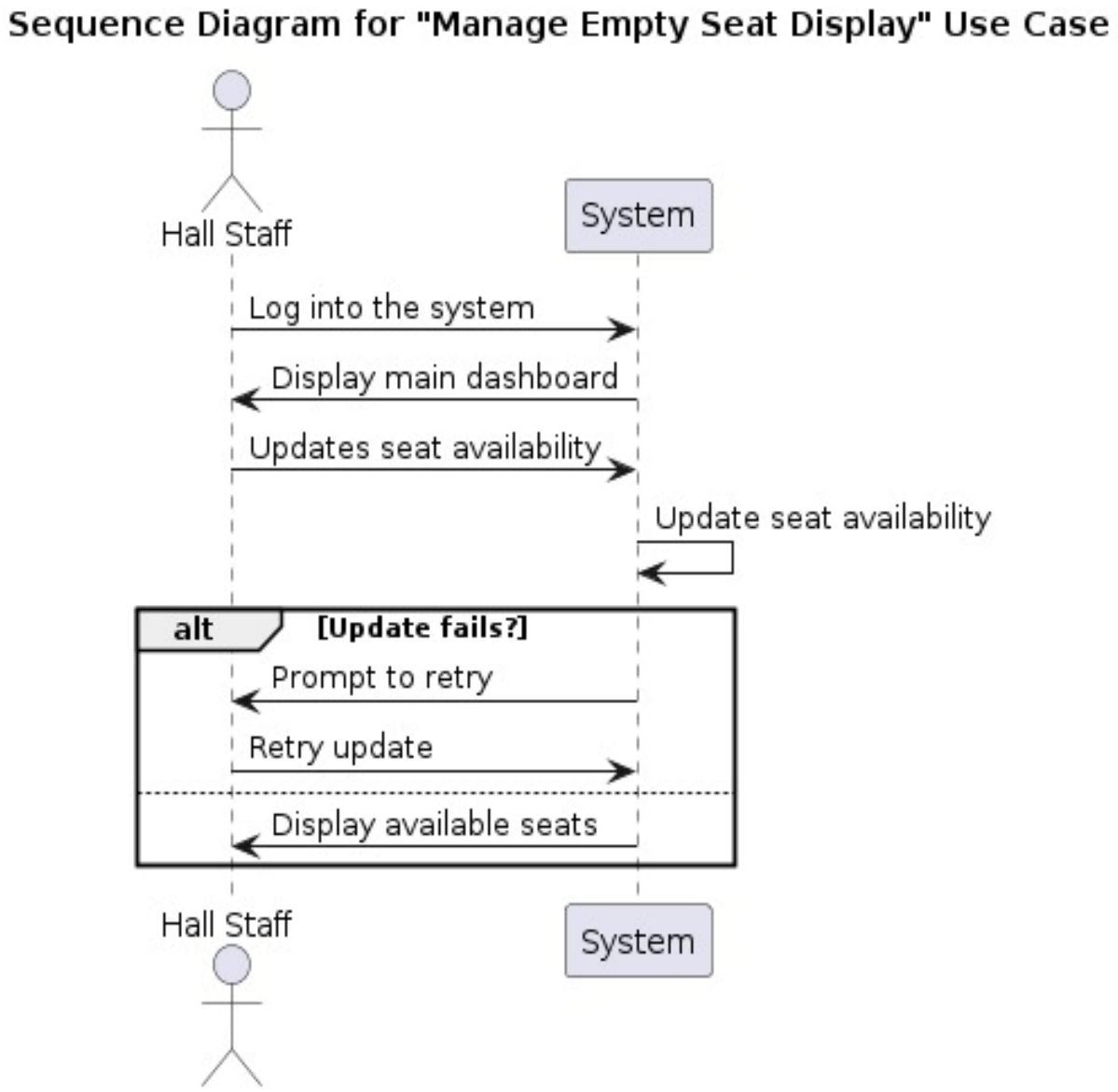

Fig-19: Sequence diagram for NSTUAcademia Empty Seat Display

8.20 Oversee Hall Seat Application

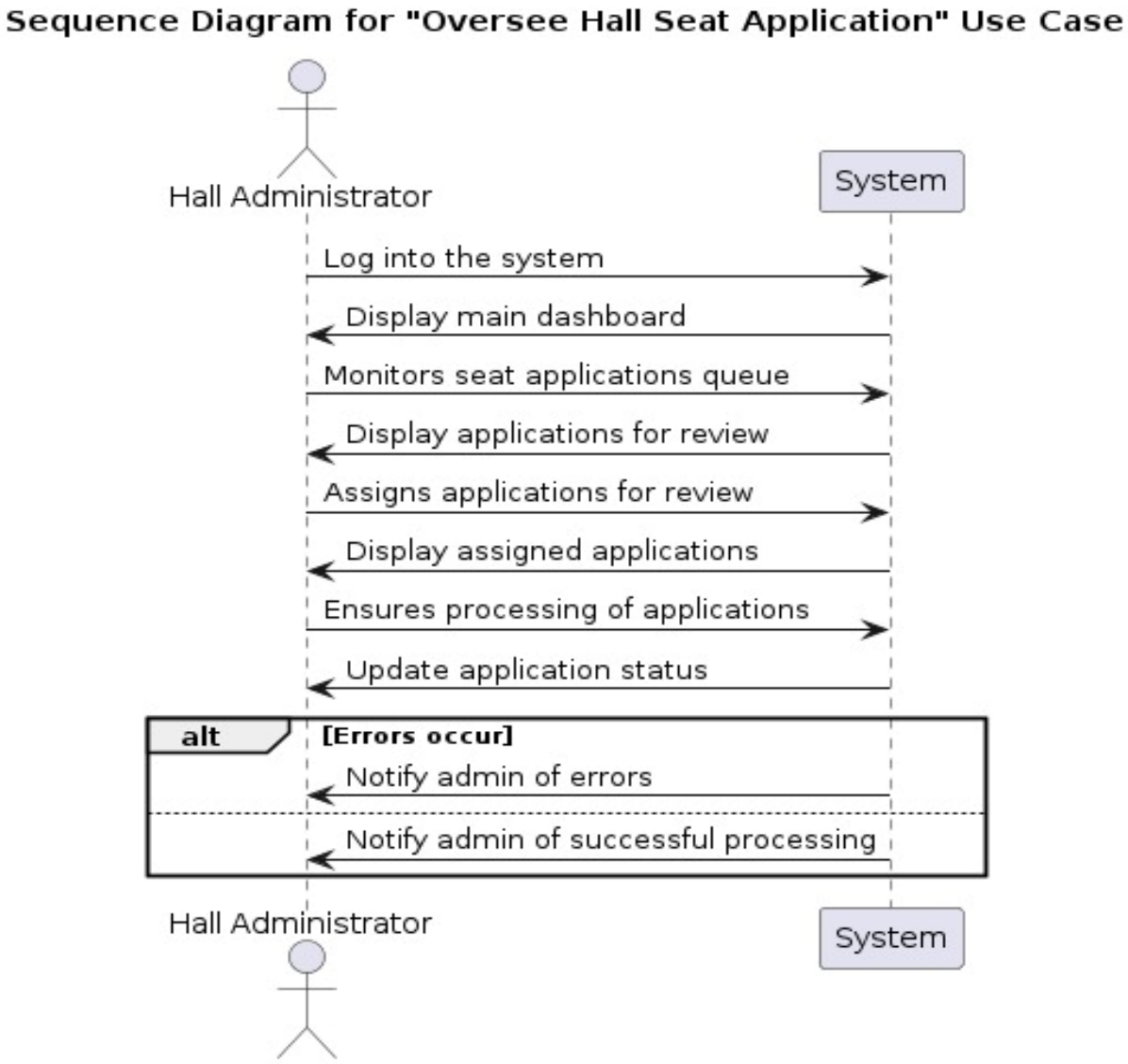

Fig-20: Sequence diagram for NSTUAcademia Oversee Hall Seat Application

8.21 Issue Boarding Card

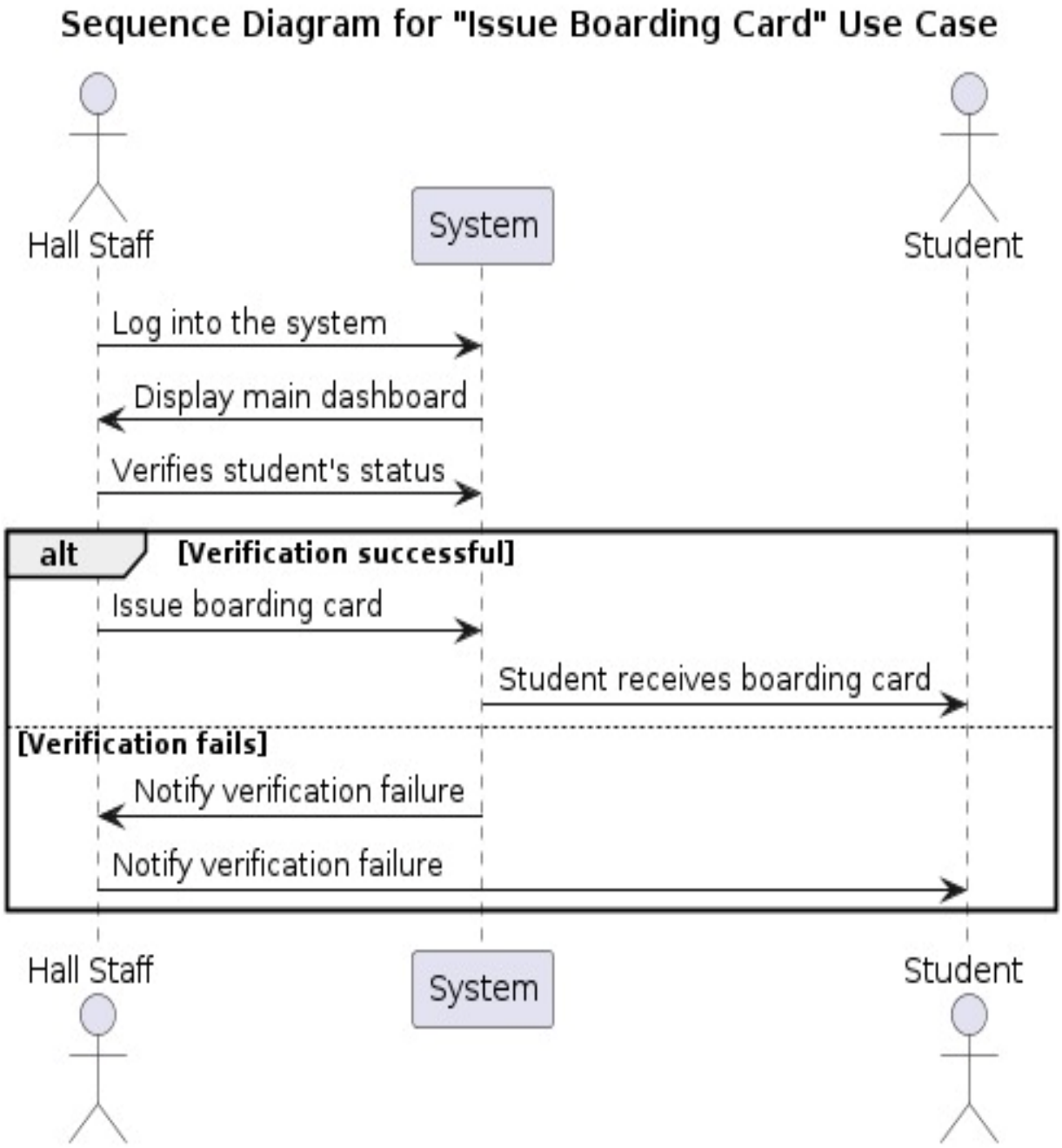

Fig-21: Sequence diagram for NSTUAcademia Issue Boarding Card

8.22 Verify Application

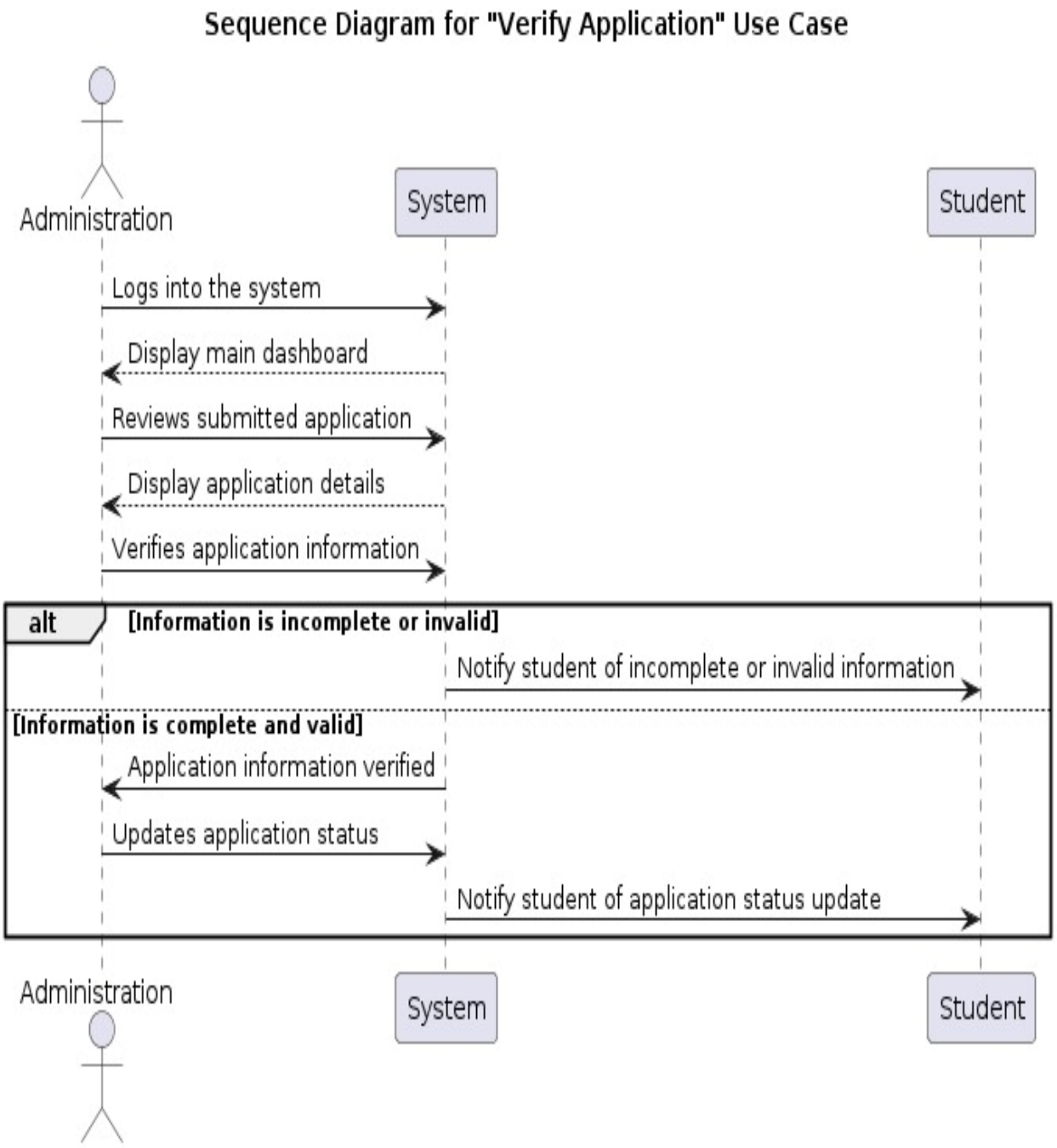

Fig-22: Sequence diagram for NSTUAcademia Verify Application

9. Activity and Swimlane Diagram

An activity diagram is a graphical representation of an executed set of procedural system activities and considered a state chart diagram variation. Activity diagrams describe parallel and conditional activities,use cases and system functions at a detailed level. Activity and swim lane diagram for NstuAcademia given below.

9.1 LogIn

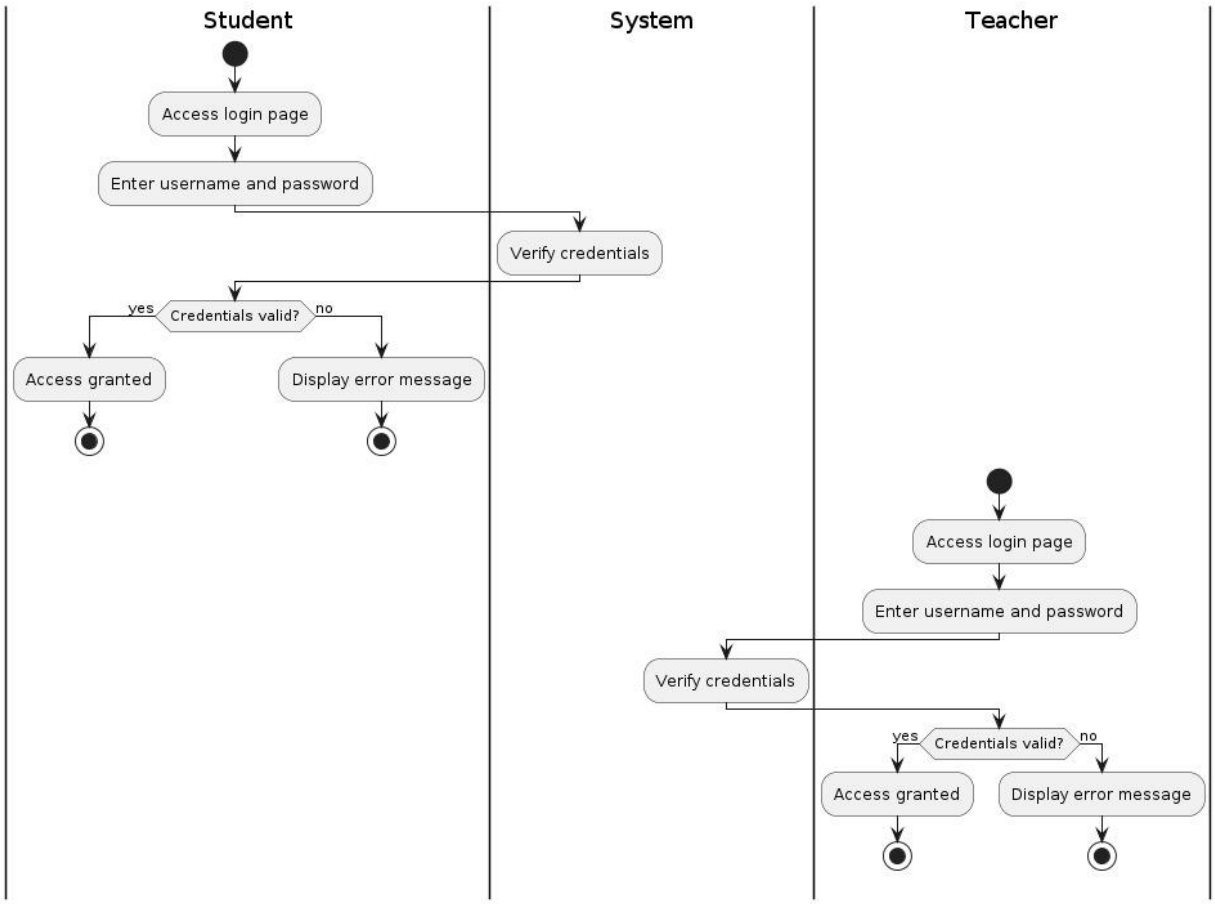

Fig-1: Activity and swim lane diagram for NSTUAcademia LogIn

9.2 SignUp

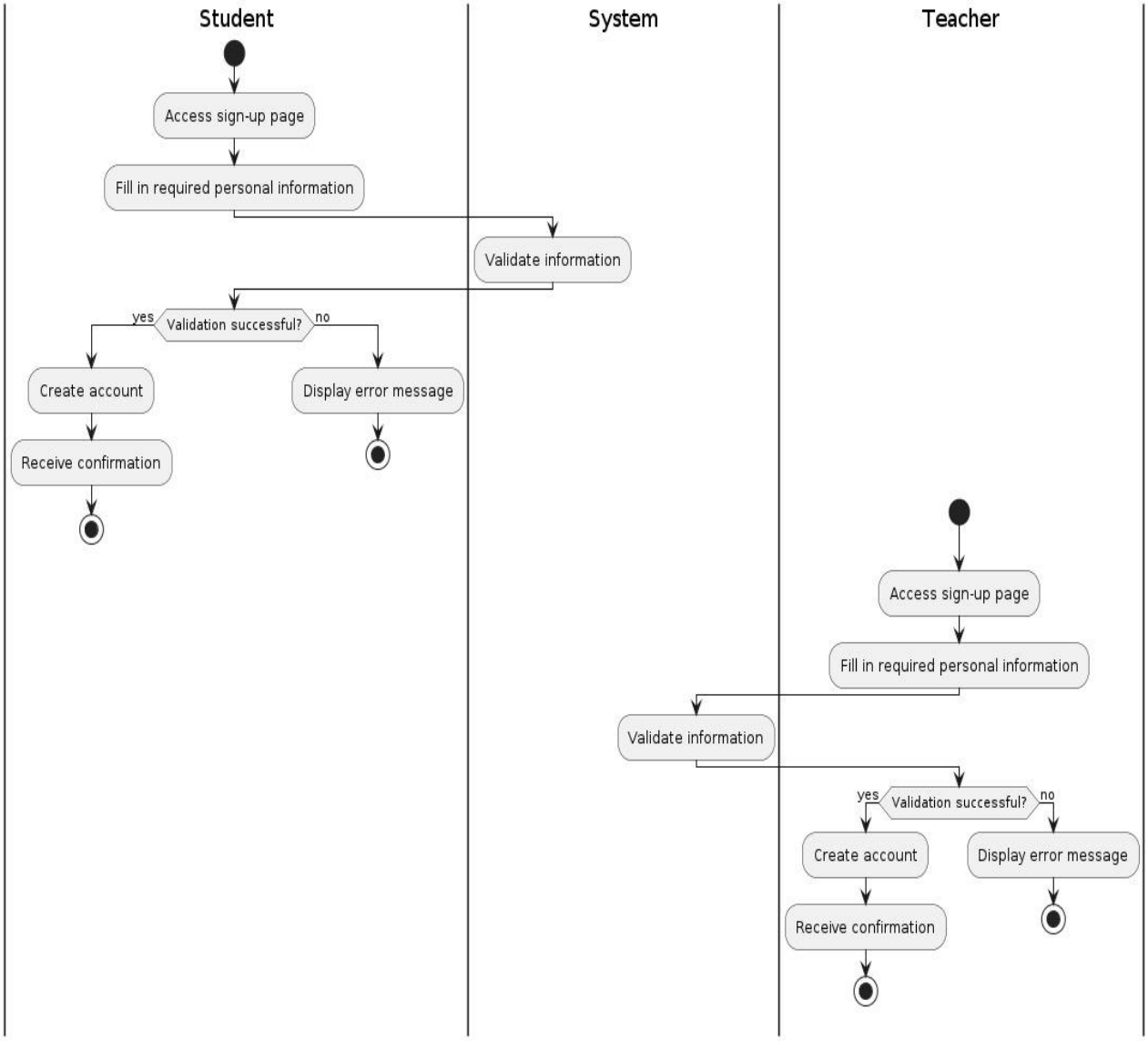

Fig-2: Activity and swim lane diagram for NSTUAcademia SignUp

9.3 Select Semester

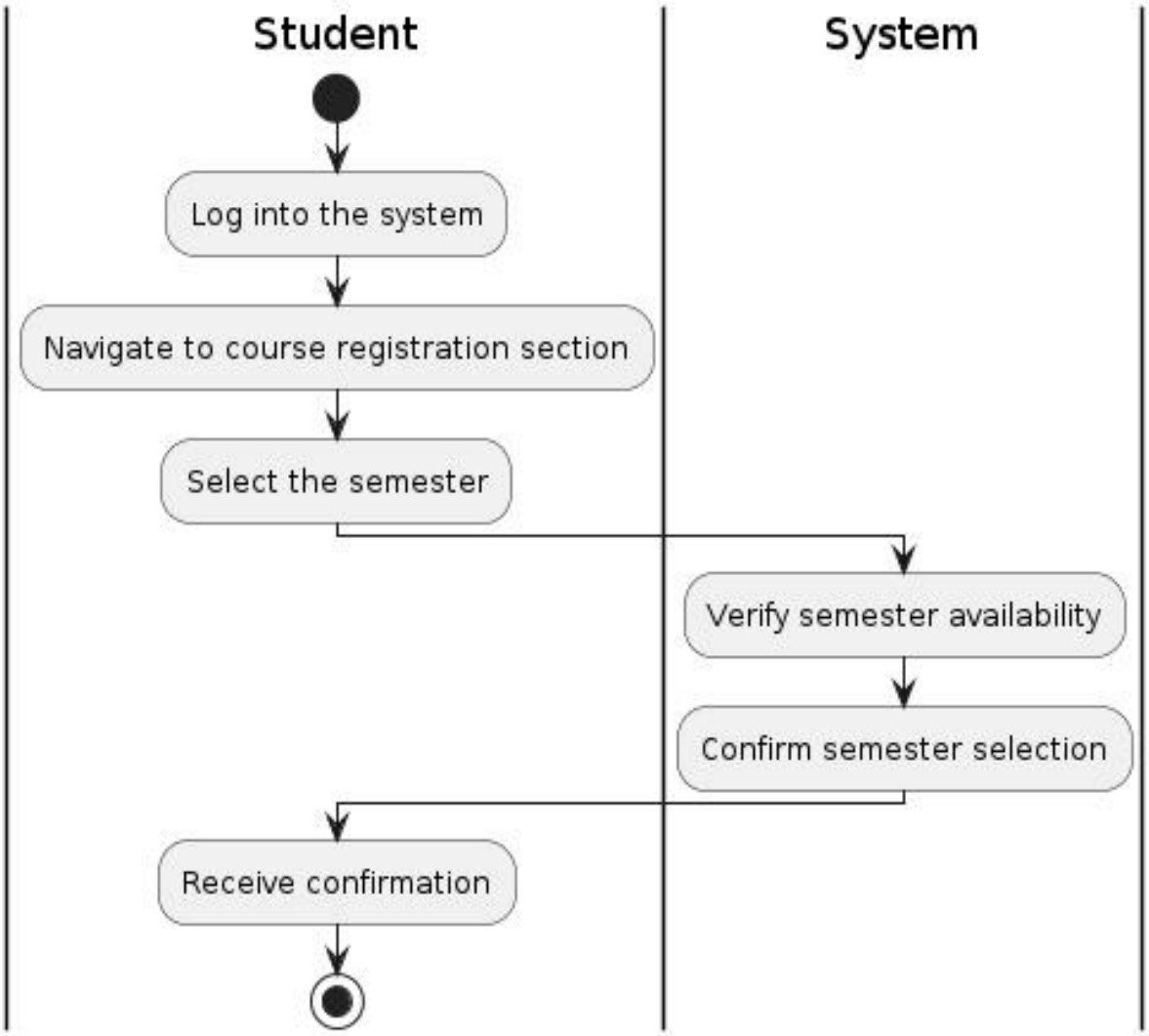

Fig-3: Activity and swim lane diagram for NSTUAcademia Select semester

9.4 Course Registration

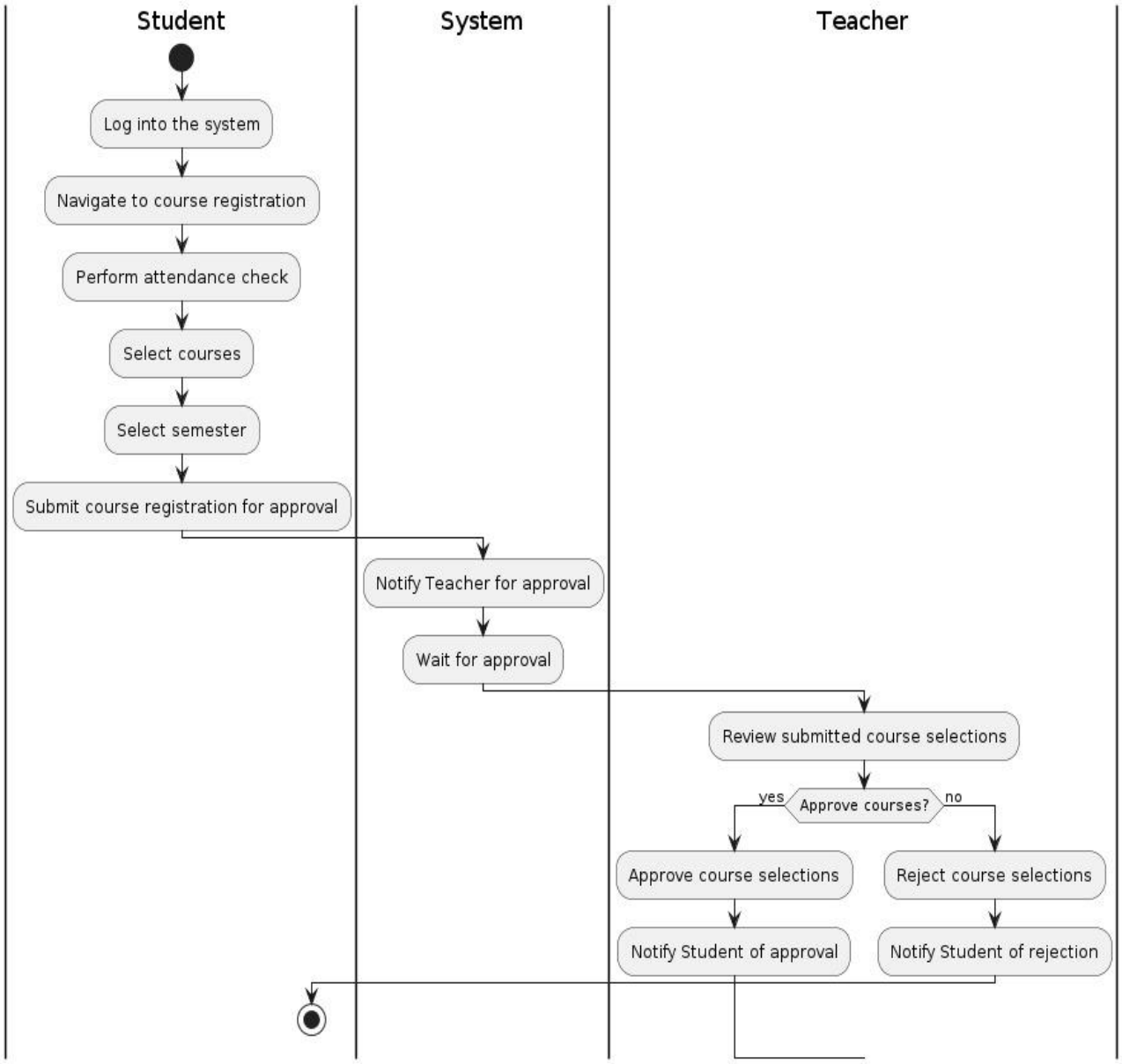

Fig-4: Activity and swim lane diagram for NSTUAcademia Course Registration

9.5 Choose Course

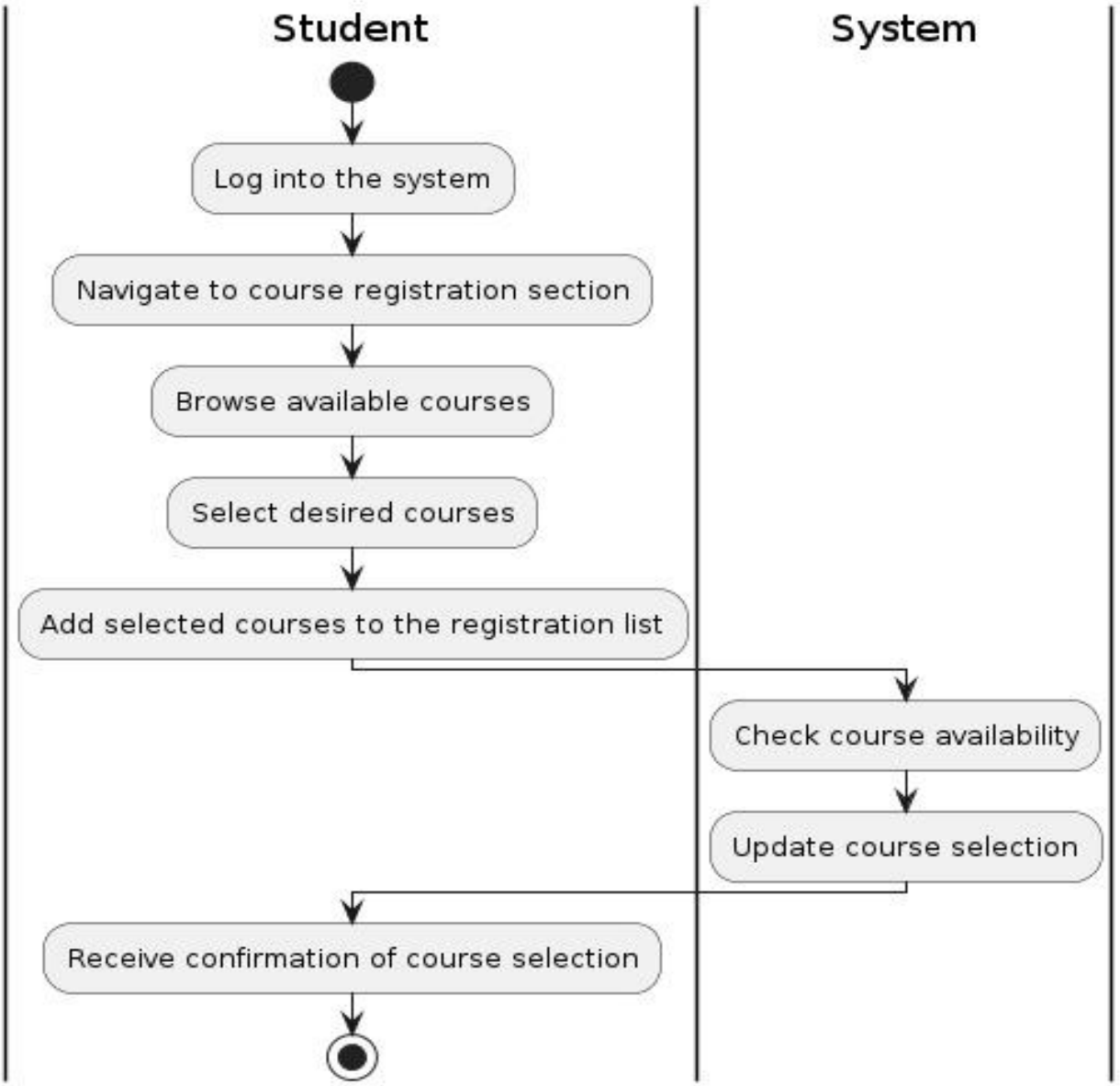

Fig-5: Activity and swim lane diagram for NSTUAcademia Choose course

9.6 Approve Course Selection

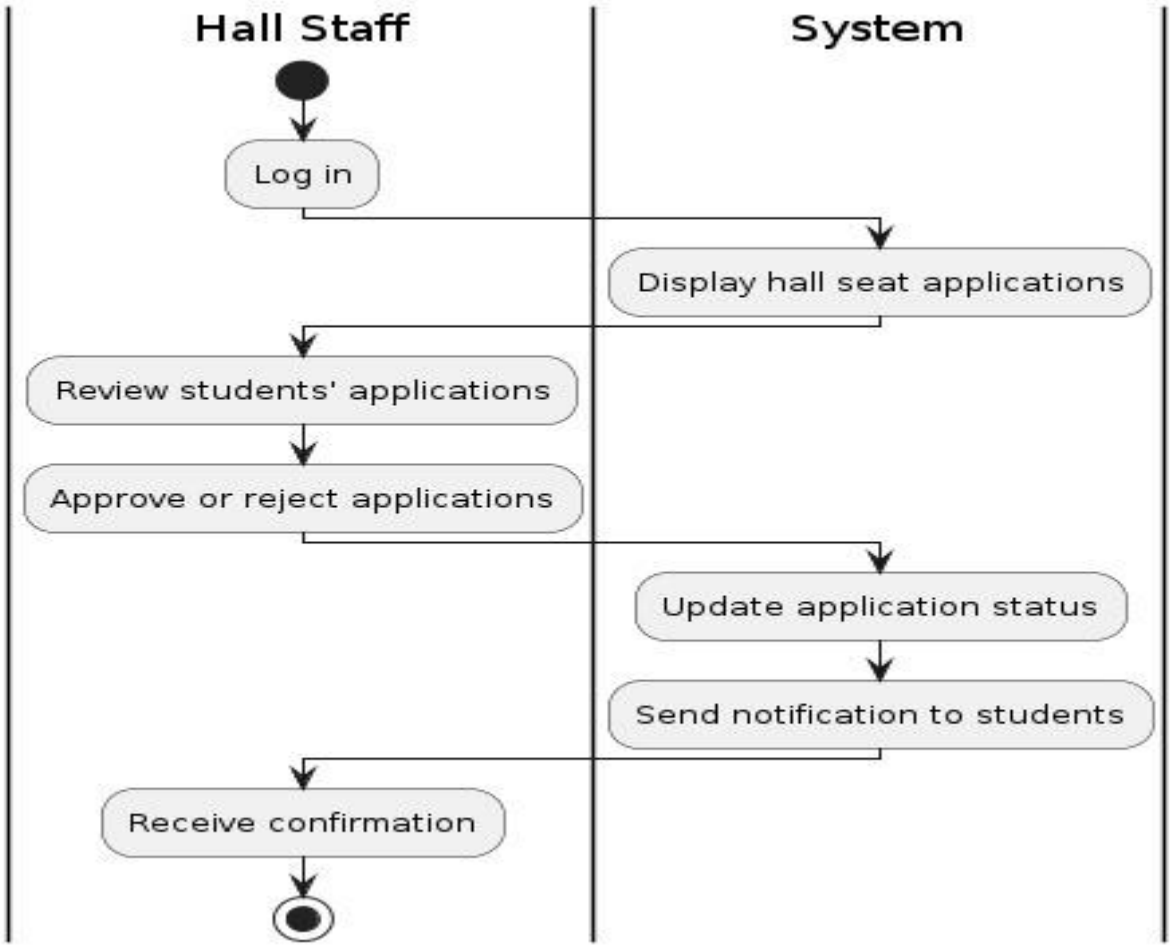

Fig-6: Activity and swim lane diagram for NSTUAcademia Approve Course Selection

9.7 Input Attendance

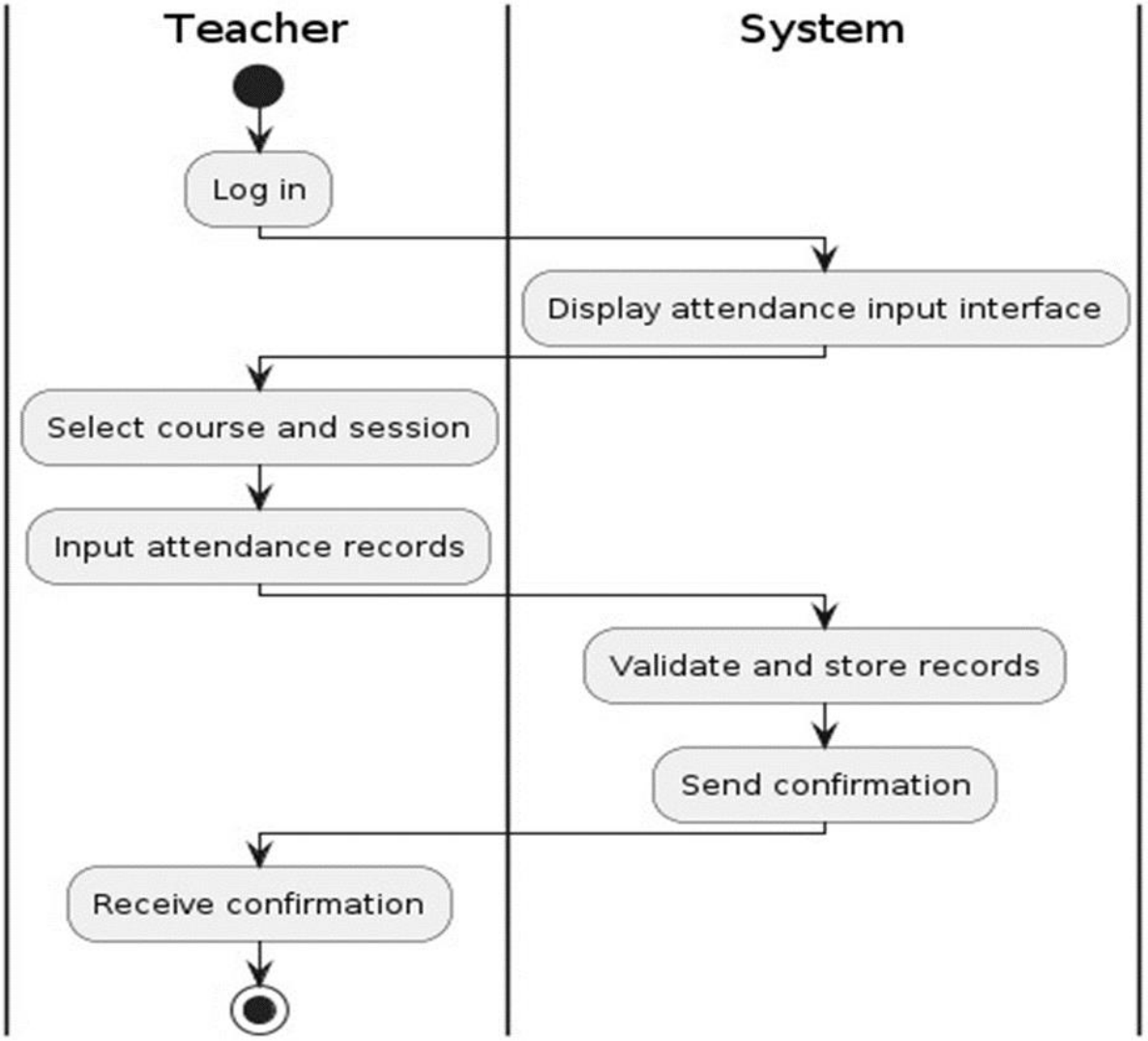

Fig-7: Activity and swim lane diagram for Antiacademic Input Attendance

9.8 Attendance Check

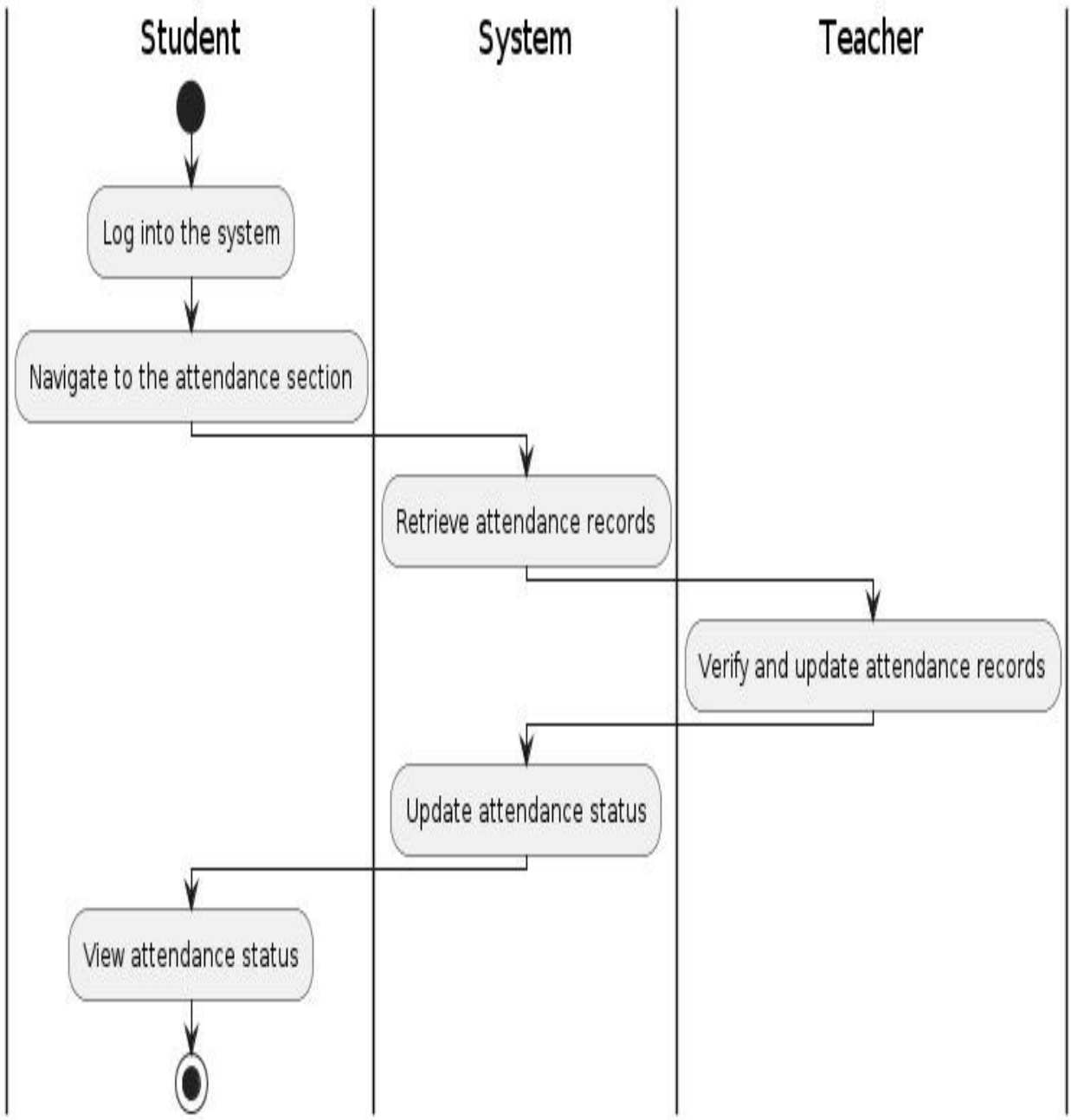

Fig-8: Activity and swim lane diagram for NSTUAcademia Attendance check

9.9 Approve Course Registration Application

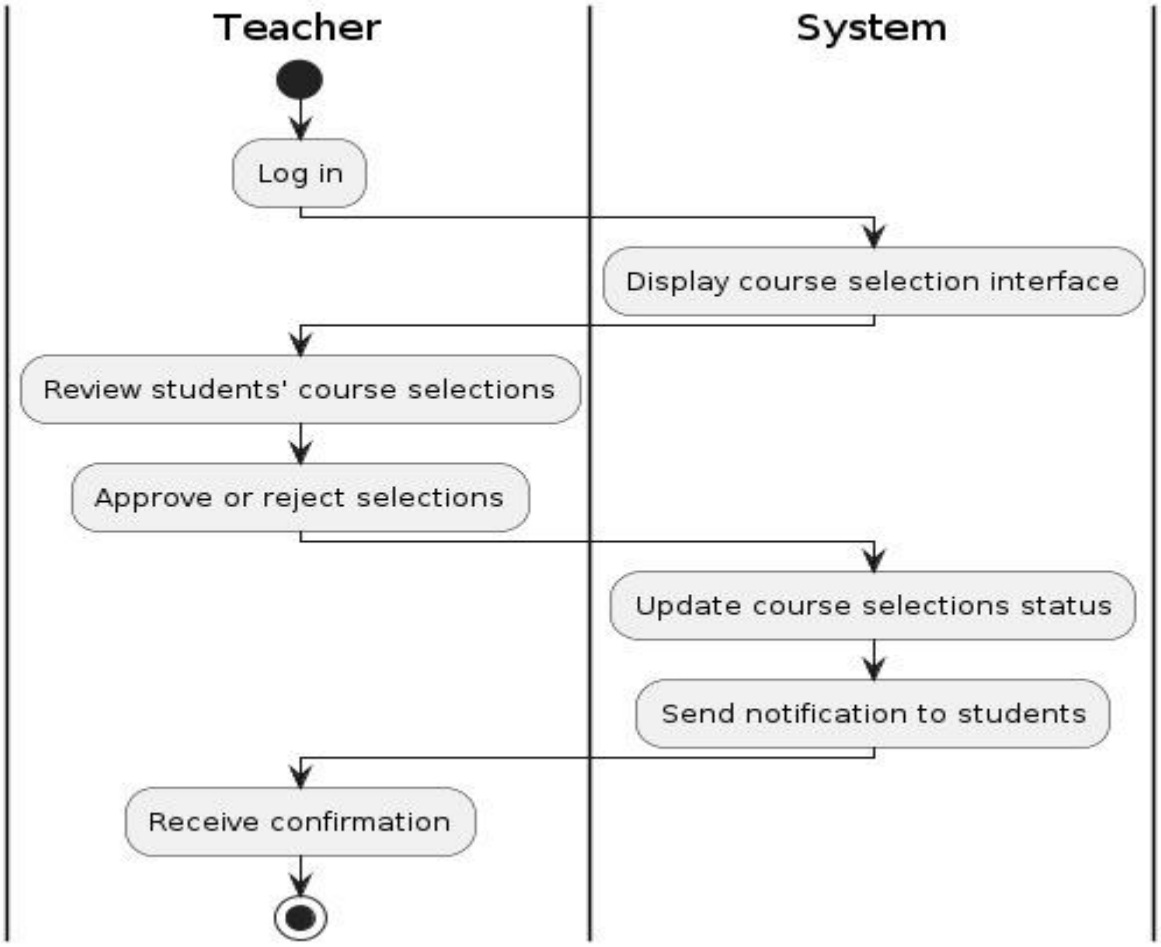

Fig-9: Activity and swim lane diagram for NSTUAcademia Approve course registration application

9.10 Payment Receive

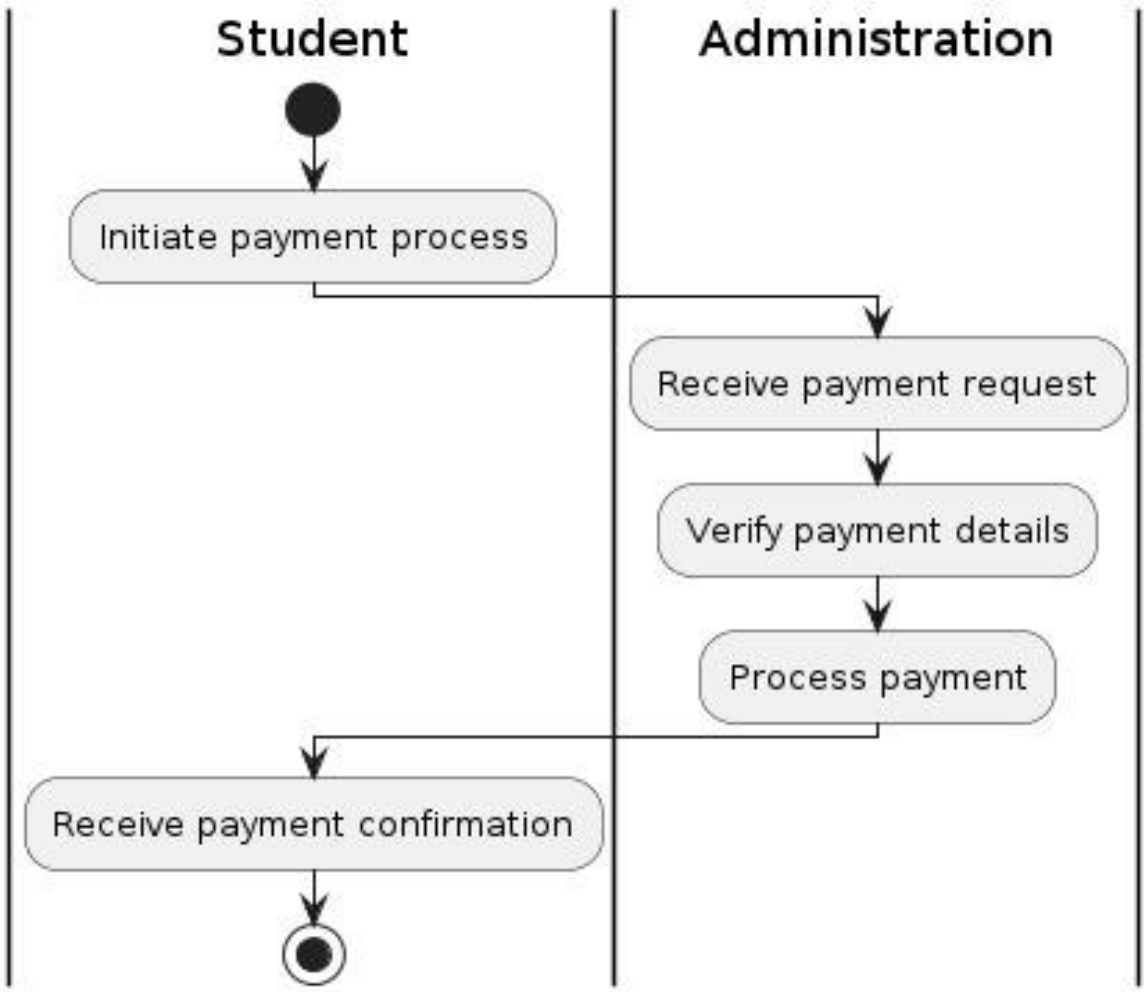

Fig-10: Activity and swim lane diagram for NSTUAcademia Payment Receive

9.11 Payment

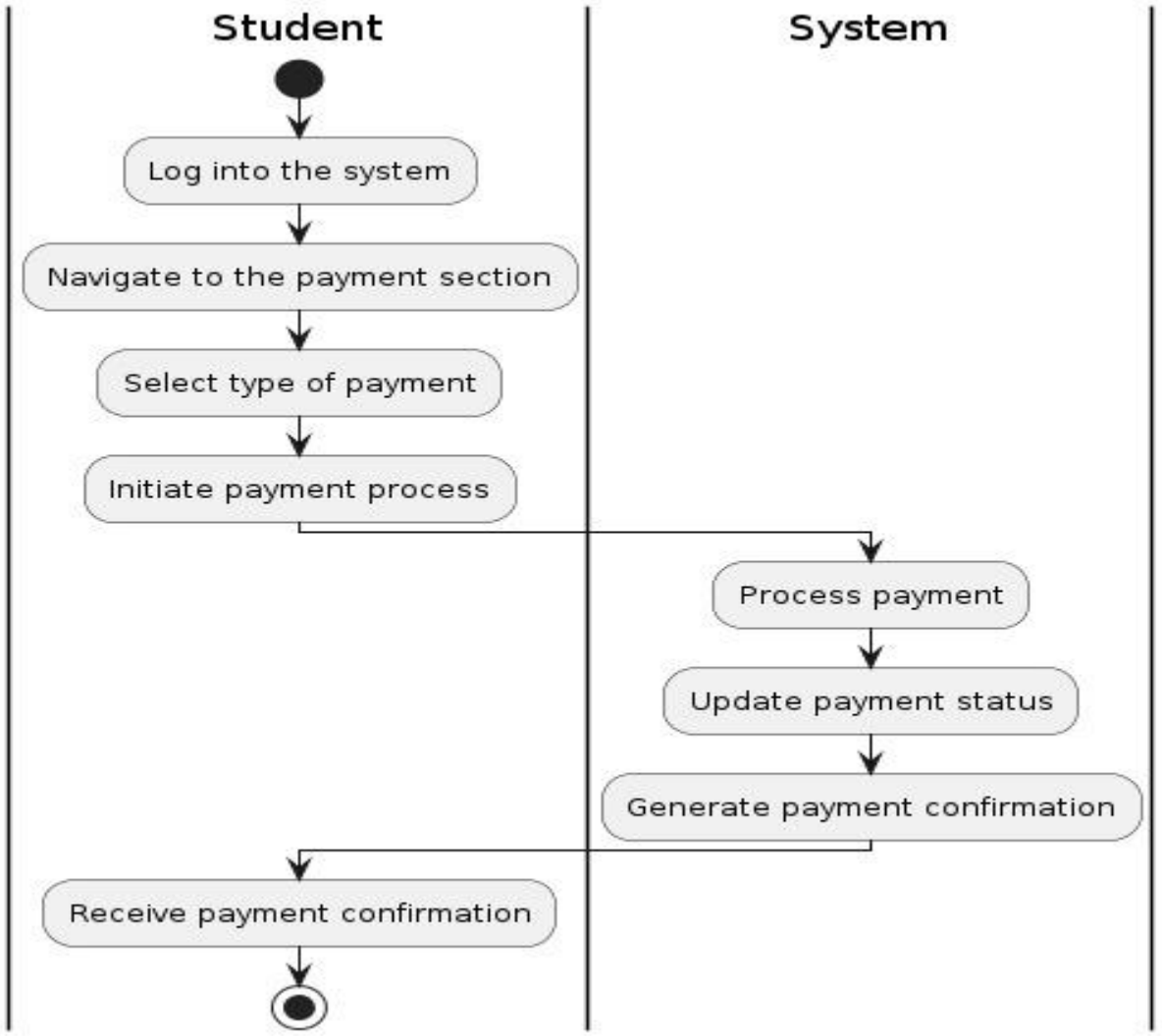

Fig-11: Activity and swim lane diagram for NSTUAcademia Payment

9.12 Receive Confirmation

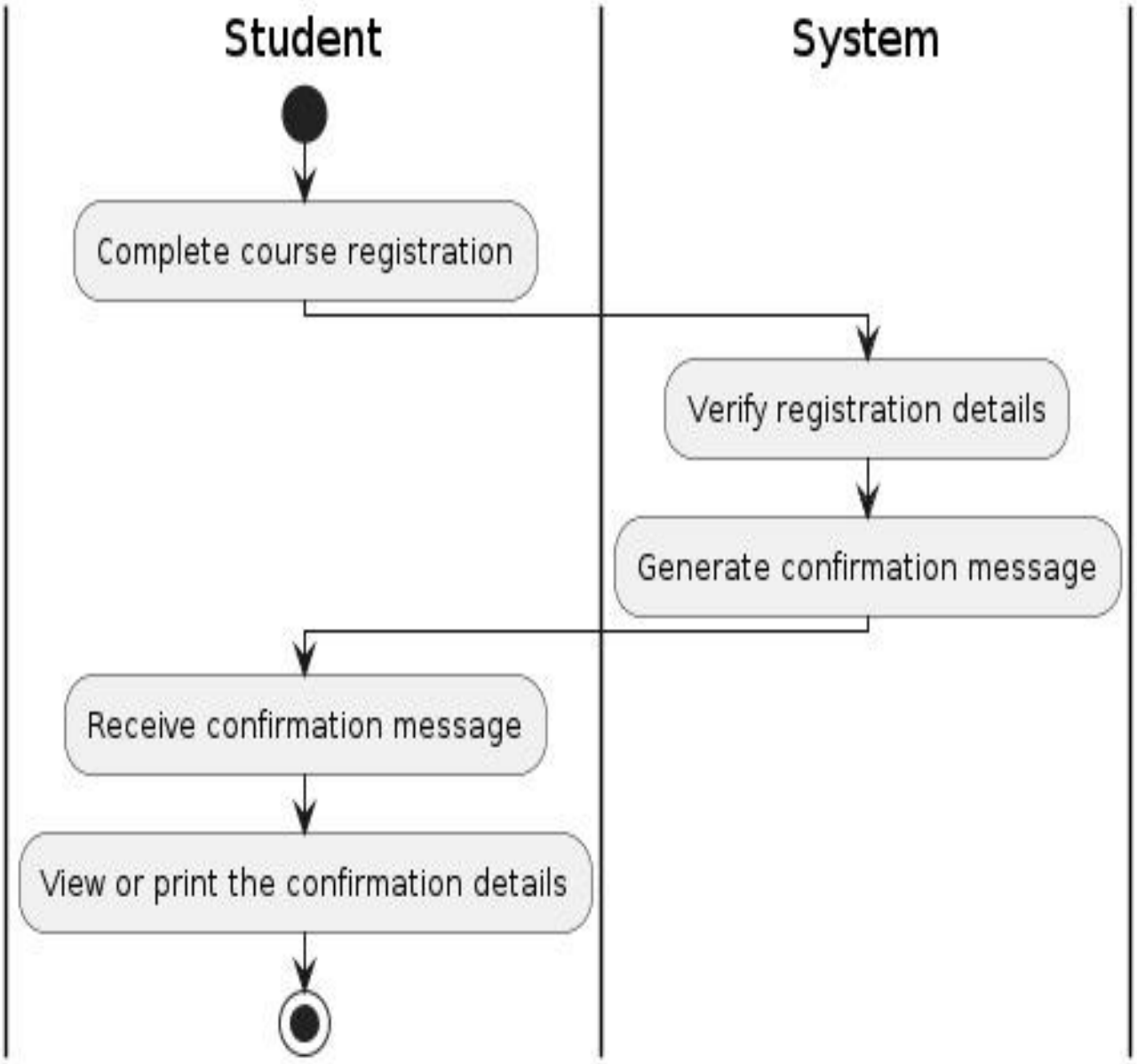

Fig-12: Activity and swim lane diagram for NSTUAcademia Receive confirmation

9.13 Issue Admit Card

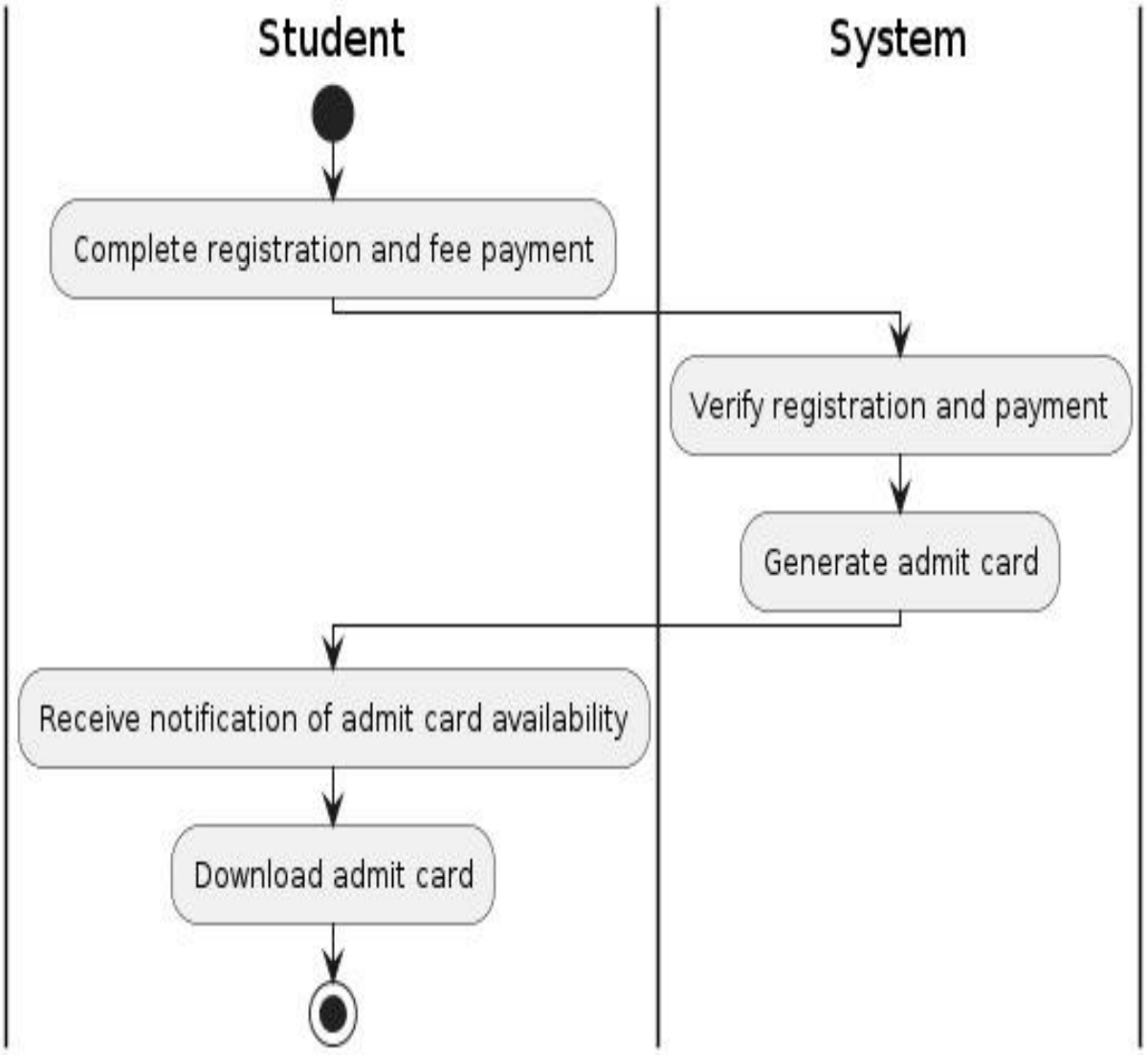

Fig-13: Activity and swim lane diagram for NSTUAcademia Issue admit card

9.14 Communication

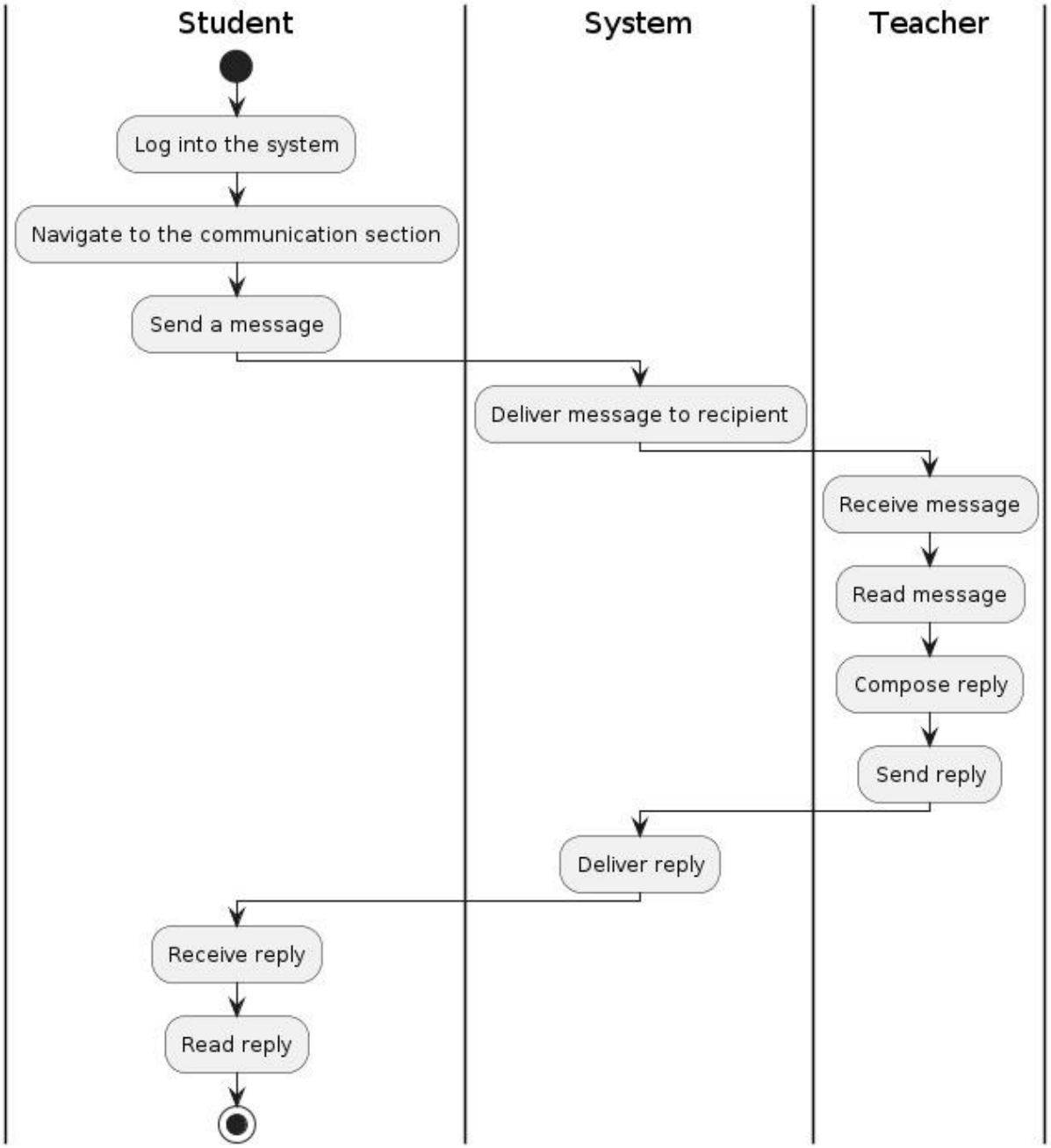

Fig-14: Activity and swim lane diagram for NSTUAcademia Communication

9.15 Apply Hall

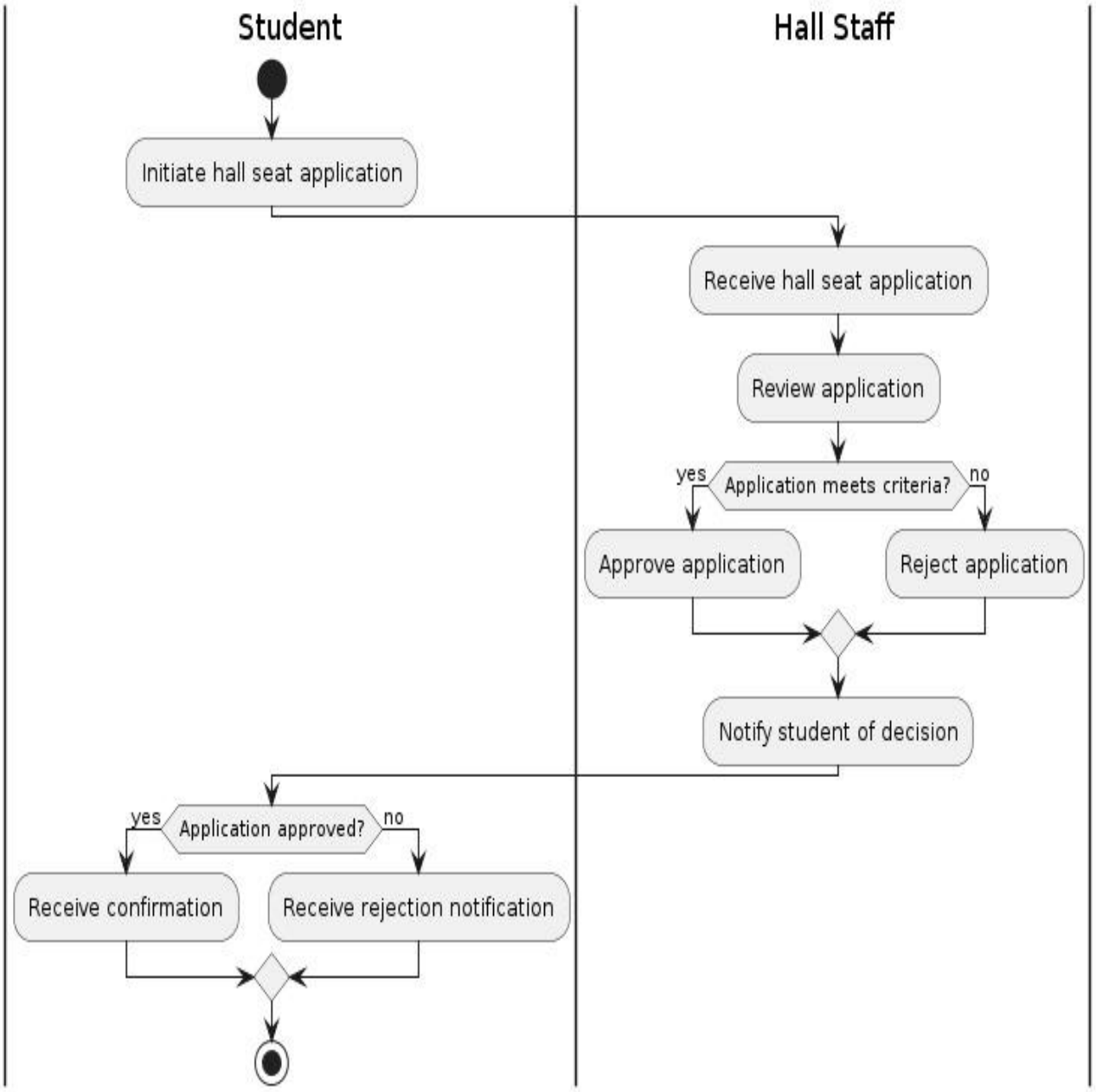

Fig-15: Activity and swim lane diagram for NSTUAcademia Apply Hall

9.16 Verify Application

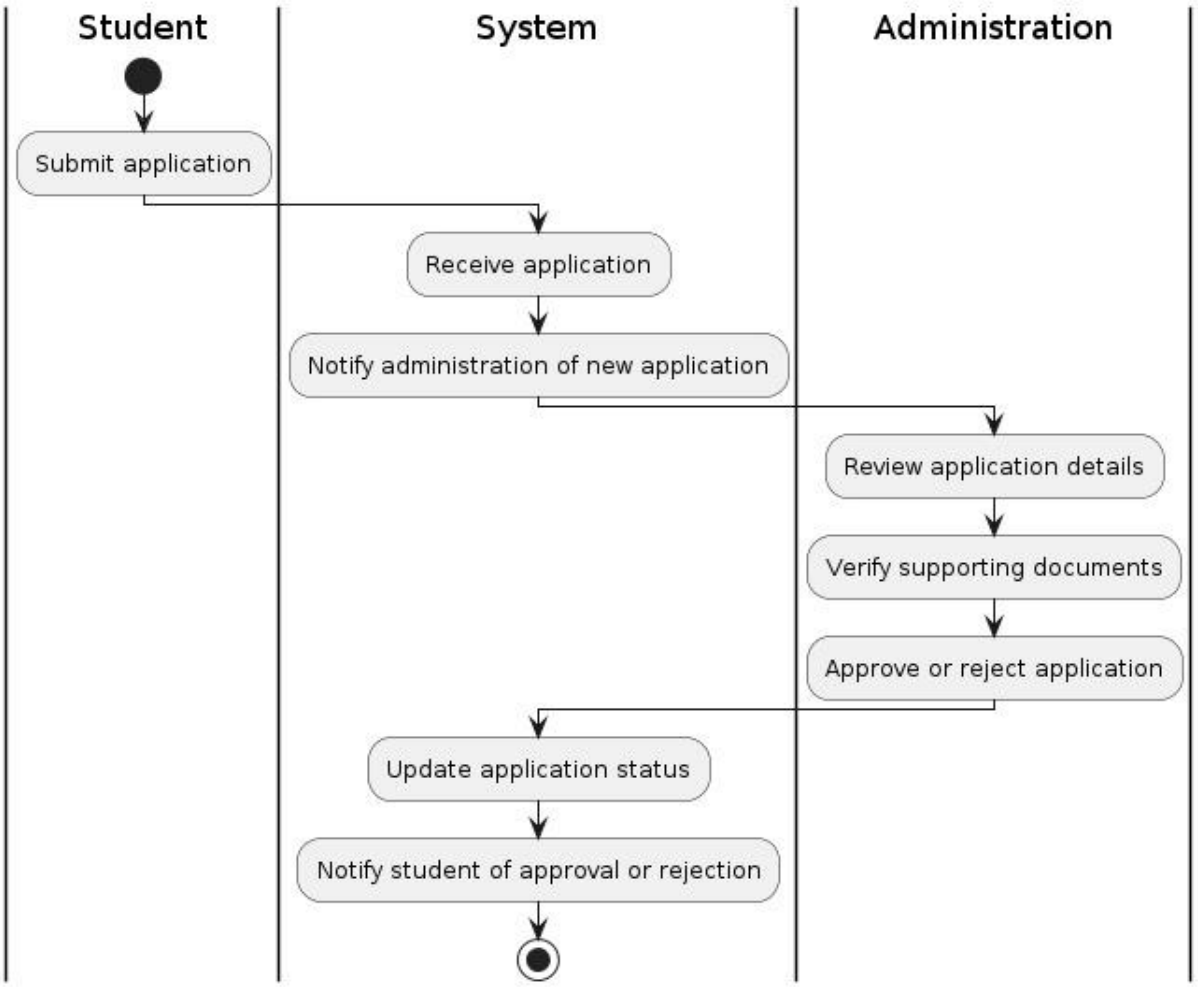

Fig-16: Activity and swim lane diagram for NSTUAcademia verify Application

9.17 Approve Hall Application

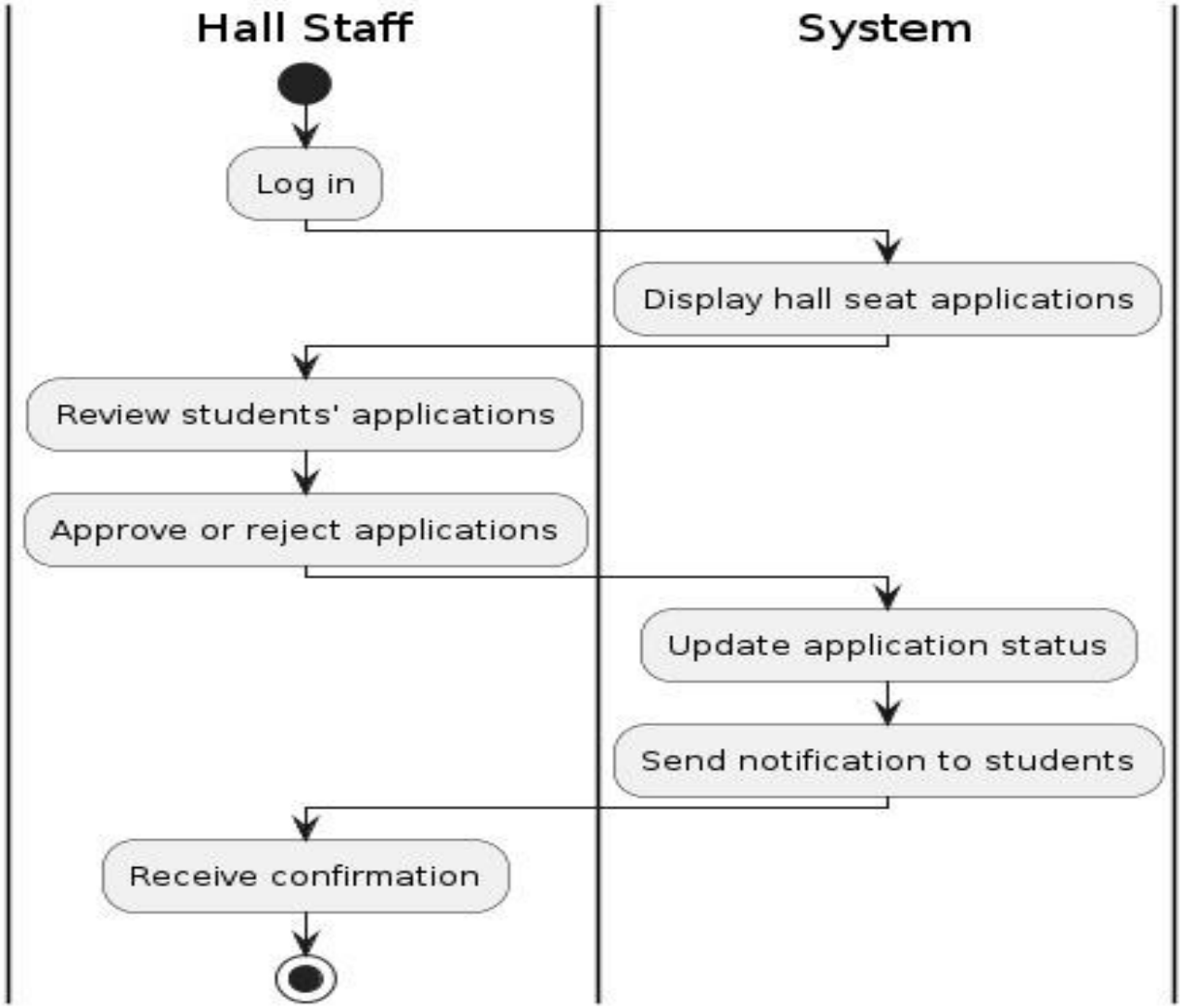

Fig-17: Activity and swim lane diagram for NSTUAcademia Approve Hall Application

9.18 Certificate Submission

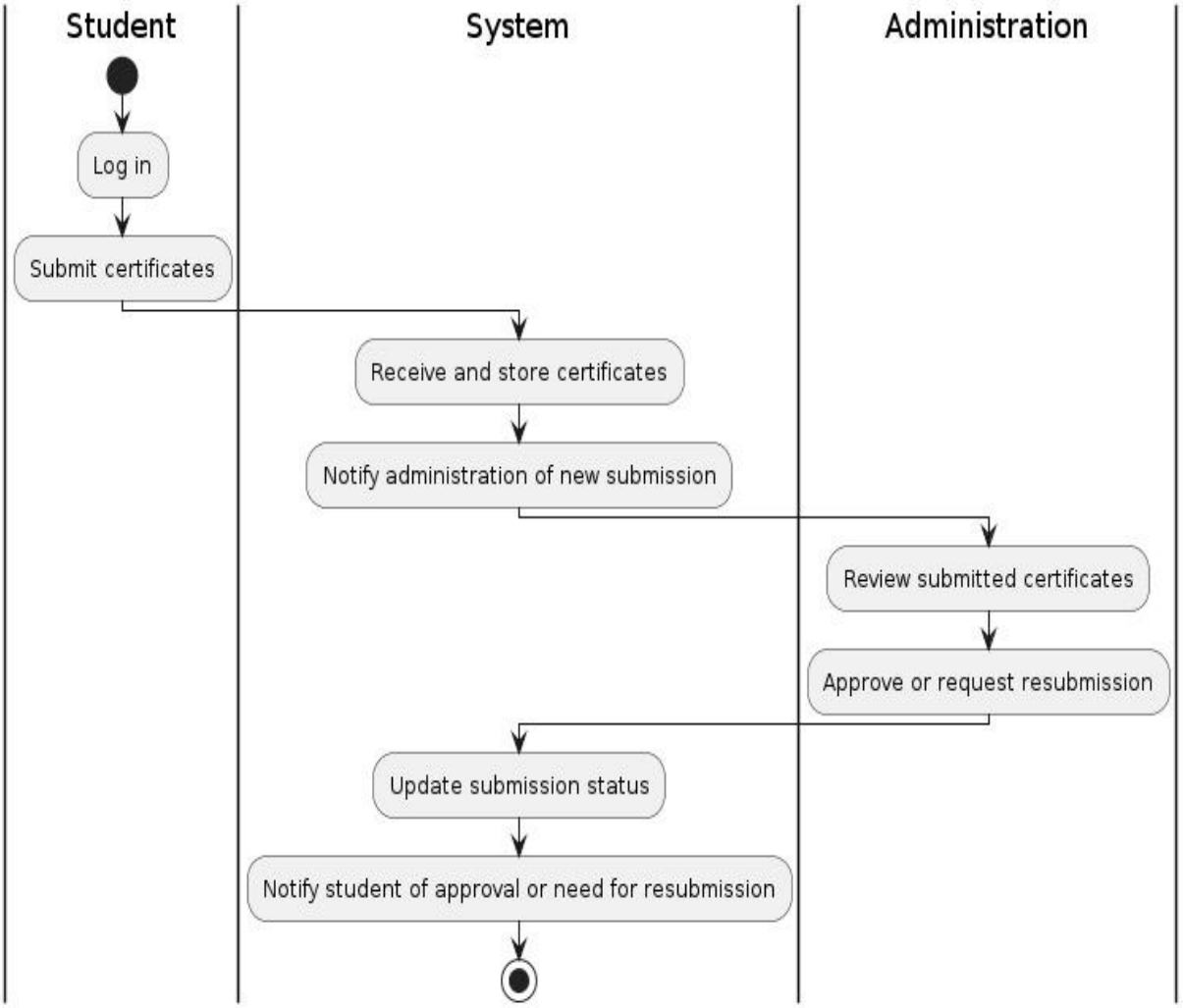

Fig-18: Activity and swim lane diagram for NSTUAcademia certificate submission

9.19 Empty Seat Display

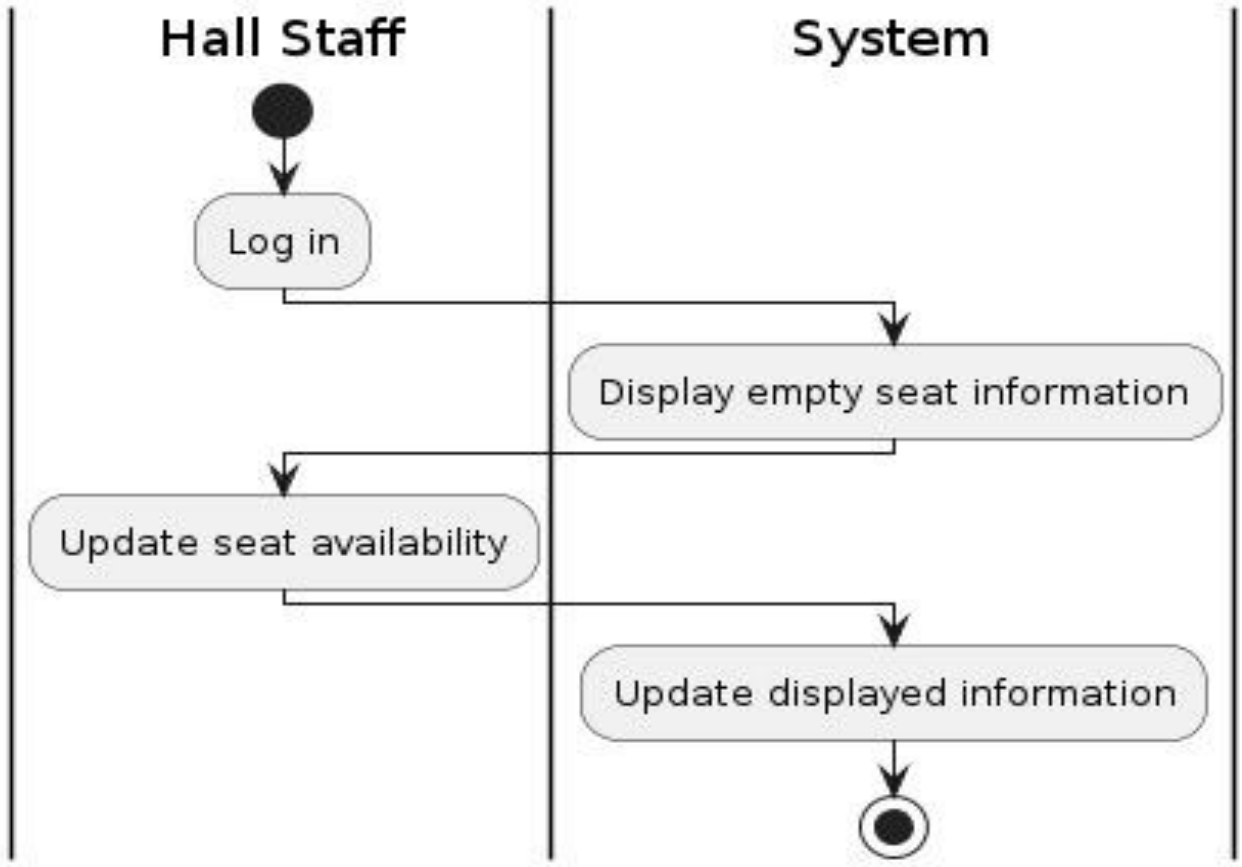

Fig-19: Activity and swim lane diagram for NSTUAcademia Empty seat Display

9.20 Oversee Hall Seat Application

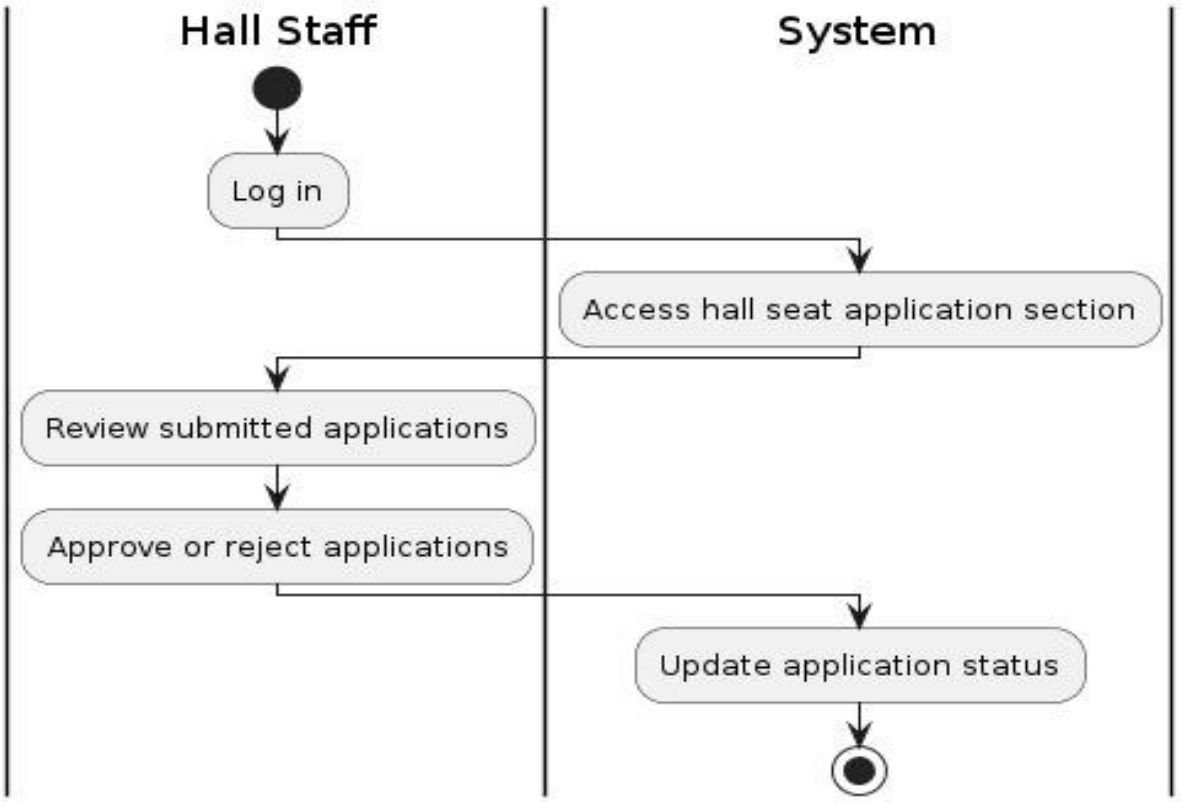

Fig-20: Activity and swim lane diagram for NSTUAcademia Oversee Hall Seat Application

9.21 Issue Boarding Card

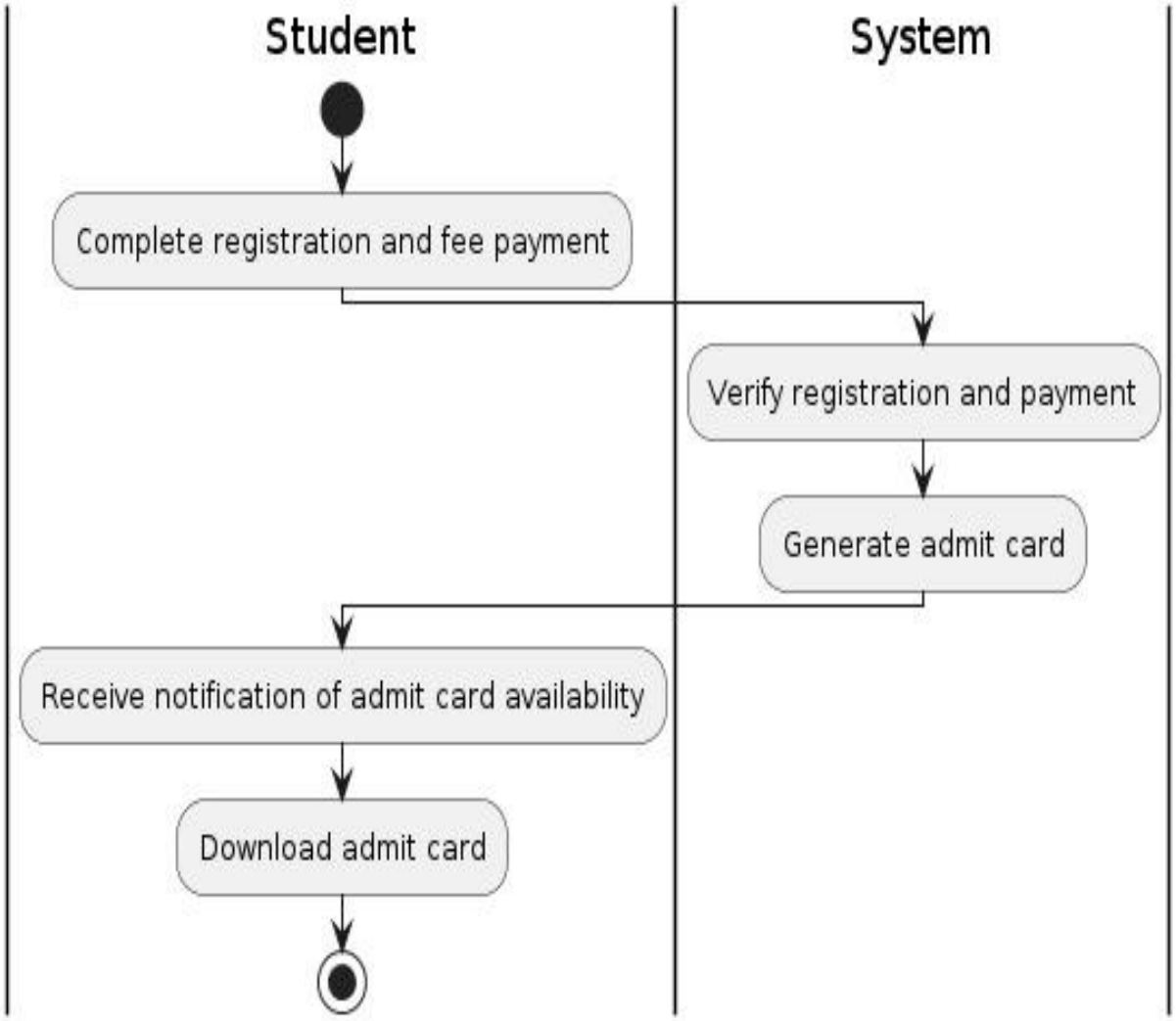

Fig-21: Activity and swim lane diagram for NSTUAcademia Issue Boarding Card

9.22 Verify Application

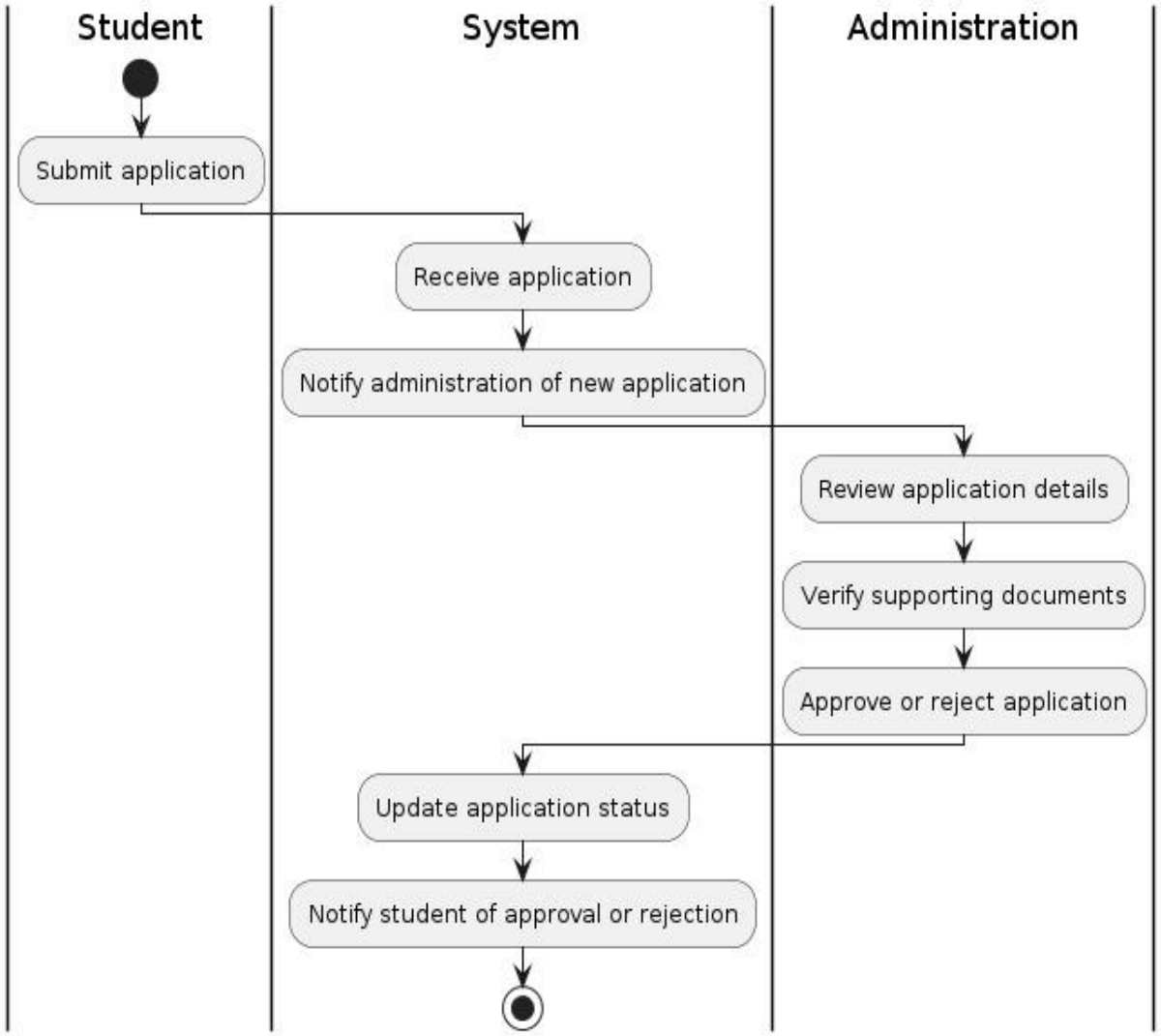

Fig-22: Activity and swim lane diagram for NSTUAcademia verify Application

10. Appendix

10.1 Prioritization of Requirements

We've prioritized the functional requirements of the NSTU Academia system by following the Three-level Scale technique.

10.1.1 Three-level Scale

When a Business Analyst categorizes the requirements in any ordering or ranking scale, it is subject to the analyst's understanding of the business. Many analysts suggest that this method has some drawbacks and advocate methods that have more than one scale.
The priority of the different sections in the diagram is indicated by the numbering system. The highest priority is given to important items, while the lowest priority is given to less urgent items.
High Priority: These requirements are critical and cannot be ignored. They pertain to compliance or contract obligations and must be addressed in the current release. Failure to implement these requirements can have negative consequences for the business.
Medium Priority: These requirements are important but not as pressing as high priority items. They should be addressed after completing the high priority items. Within this category, items on the right side of the dividing line are of higher priority.
Low Priority: These items are neither important nor urgent and can be addressed later, after completing the items in the first two categories.
The priority order should be followed by starting with the bottom-right corner of the high-priority section and working up and to the left.

10.1.2 Prioritization of the Requirements of NstuAvademia

FR1 – High priority: The system should allow users to create an account and log in. (Stakeholders: Students, Teacher, Hall Staff)
FR2 – High priority: The system must allow users to manage their personal information. (Stakeholders: Students)
FR3 – High priority: The system must allow students to register for courses for a selected semester. (Stakeholders: Students, Teacher, Hall Staff)

FR4 – High priority: The system must allow students to select the semester for course registration. (Stakeholders: Students)
FR5 – High priority: The system must display a list of available courses for the selected semester. (Stakeholders: Students)
FR6 – Medium priority: The system should allow students to check their attendance for registered courses. (Stakeholders: Students)
FR7 – Medium priority: The system should allow teachers to input attendance records for students. (Stakeholders: Students, Teacher)
FR8 – High priority: The system should allow teachers to approve students' course selections. (Stakeholders: Teachers, Course Coordinator, Students)
FR9 – Medium priority: The system should allow for the approval and signature of course registration forms. (Stakeholders: Students)
FR10 – High priority: The system must provide a payment gateway for processing course

registration fees. (Stakeholders: Students)
FR11 – High priority: The system must generate a payment confirmation receipt once the payment is successful. (Stakeholders: Students)
FR12 – High priority: The system must provide a communication platform for students, teachers, and hall staff to interact. (Stakeholders: Students, Teacher, Hall Staff)
FR13 – High priority: The system must allow teachers to input attendance records for their allocated courses. (Stakeholders: Faculty)
FR14 – High priority: The system must manage the hall seat allocation process, allowing students to apply for hall seats. (Stakeholders: Students, Hall Staff)
FR15 – High priority: The system must display available hall seats for students to view and apply for. (Stakeholders: Students)
FR16 – High priority: The system must allow students to apply for available hall seats. (Stakeholders: Students)
FR17 – High priority: The system must verify hall seat applications and issue boarding cards. (Stakeholders: Hall Staff)
FR18 – High priority: The system must update and display the status of hall seats in real-time. (Stakeholders: Hall Staff)
FR19 – High priority: The system must manage the certificate submission process for students vacating hall seats. (Stakeholders: Students, Hall Staff)
DR1 – High priority: Users will upload profile pictures, course-related documents, and hall seat application materials. Images must be within a maximum size limit of 3 MB. (Stakeholders: Students, Faculty, Hall Staff)
DR2 – High priority: The system will store personal information such as name, email, batch, department, and residential details of users. (Stakeholders: Students, Faculty, Hall Staff)
DR3 – High priority: Course-related data including course code, title, description, and instructor details will be stored in the system. (Stakeholders: Faculty, System Staff)
DR4 – High priority: The system will record attendance data, including the number of classes held and attended by students for each course. (Stakeholders: Faculty, System Staff)
DR5 – High priority: Payment information such as transaction ID, amount, and payment status will be stored securely. (Stakeholders: Students, System Staff)
DR6 – High priority: Information regarding hall seat allocation, including available seats, applicant details, and boarding card issuance records, will be maintained. (Stakeholders: Hall Staff, System Staff)
PR1 – Medium priority: Course search results should display within 1 second, and course details, including associated documents and images, should load within 5 seconds given good network speed. (Stakeholders: Students, Faculty, Hall Staff)
PR2 – Medium priority: The system should quickly detect and manage issues, ensuring minimal disruption to users by resolving problems within a short time frame. (Stakeholders: Students, Faculty, Hall Staff)
PR3 – High priority: Payment transactions should be processed within 3 seconds to provide immediate feedback to the user. (Stakeholders: Students)
PR4 – High priority: Attendance records should be updated in real-time, with a maximum delay of 2 seconds between submission and database reflection. (Stakeholders: Students)
PR5 – High priority: Hall seat availability status should update within 2 seconds after any change, ensuring real-time accuracy. (Stakeholders: Students, Hall Staff)
PR6 – High priority: All user interface actions, such as clicks, navigations, and form submissions,

should respond within 1 second to ensure a fluid user experience. (Stakeholders: Students, Faculty, Hall Staff)
MR1 – High priority: Code must be developed so that it can be modified later and will be readable. (Stakeholders: Developers)
AR1 – High priority: The user interface must be responsive and work seamlessly on various devices, including desktops, laptops, tablets, and smartphones. (Stakeholders: Students, Faculty, Hall Staff)