ГУАП

КАФЕДРА № 42

ОТЧЕТ   
ЗАЩИЩЕН С ОЦЕНКОЙ

ПРЕПОДАВАТЕЛЬ

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ассистент |  |  |  | Ю.В. Ветрова |
| должность, уч. степень, звание |  | подпись, дата |  | инициалы, фамилия |

|  |
| --- |
| ОТЧЕТ О ЛАБОРАТОРНОЙ РАБОТЕ №6  МОДЕЛИРОВАНИЕ С ПОМОЩЬЮ UML. ПОСТРОЕНИЕ ДИАГРАММ ВЗАИМОДЕЙСТВИЙ |
|  |
| по курсу: Архитектура информационных систем |

РАБОТУ ВЫПОЛНИЛ

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| СТУДЕНТ гр. № | 4321 |  |  |  | Г.В. Буренков |
|  |  |  | подпись, дата |  | инициалы, фамилия |

Санкт-Петербург 2025

Содержание

[1 Цель работы 5](#_Toc196223087)

[2 Краткое описание прецедента использования 7](#_Toc196223088)

[3 Перечень сформированных логических программных классов с пояснением их назначения 9](#_Toc196223089)

[4 Диаграмма взаимодействия действий для прецедента 10](#_Toc196223090)

[5 Вывод 12](#_Toc196223091)

# **1 Цель работы**

Целью выполнения лабораторной работы является закрепление навыков моделирования взаимодействий между объектами в информационной системе с применением диаграмм последовательностей UML. В процессе выполнения работы необходимо построить модель, отражающую логику взаимодействия различных компонентов системы в контексте выполнения одного из функциональных сценариев. Это позволяет более детально понять, каким образом осуществляется обмен сообщениями между пользователем, интерфейсом системы, серверной частью и базой данных, а также выявить ключевые точки обработки ошибок и альтернативных потоков управления. Работа направлена на развитие практических умений по формализации поведения системы на уровне взаимодействий, что способствует углубленному пониманию архитектуры информационных систем.

Вариант 1. Информационное система вуза.

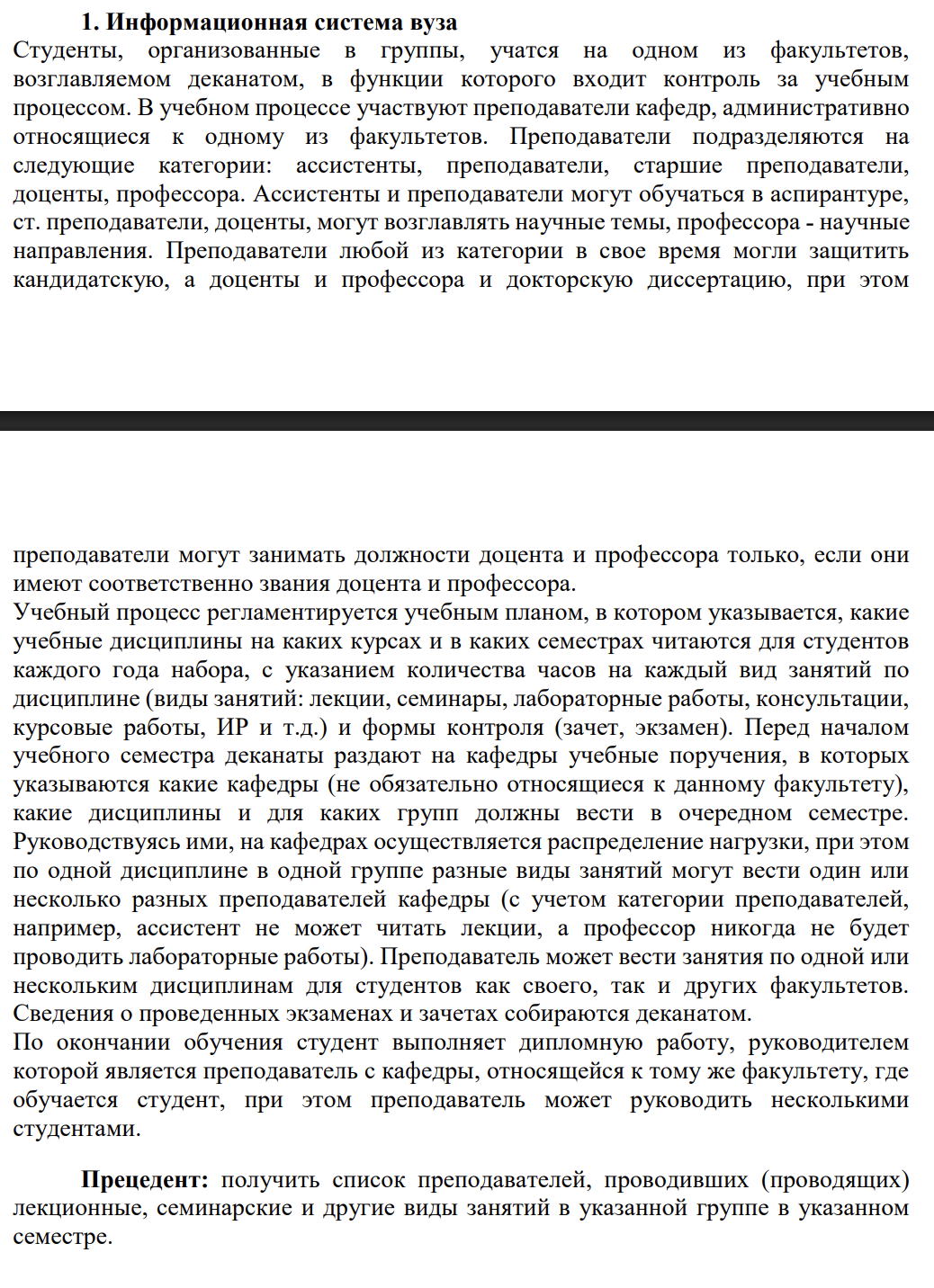


Рисунок 1 – Вариант задания

# **2 Краткое описание прецедента использования**

В данной работе используется тот же прецедент, что и в лабораторной работе №4, а именно "Получить список преподавателей". Суть данного прецедента заключается в том, что сотрудник деканата с помощью информационной системы вуза получает перечень преподавателей, проводивших или проводящих занятия в заданной учебной группе в конкретном семестре. Для этого пользователь авторизуется в системе, вводит параметры поиска — номер группы и семестр — и получает сформированный системой список с указанием дисциплин и типов занятий. Прецедент охватывает как основной, так и альтернативные сценарии, включая обработку ошибок ввода и ситуацию, когда по заданным параметрам отсутствуют данные в системе. На рисунке 2 изображена диаграмма прецедента, а на рисунке 3 изображено диаграмма изменение состояния.

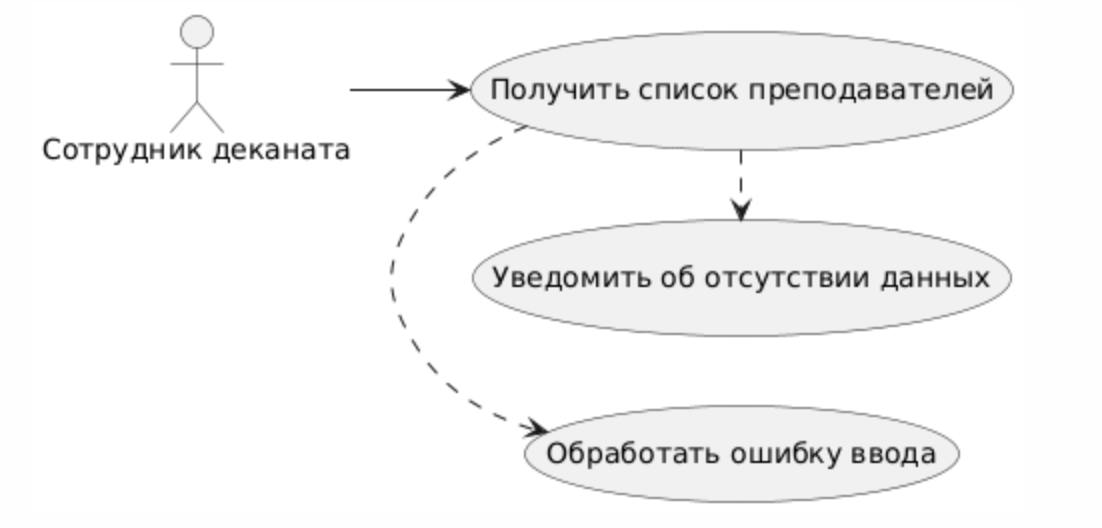


Рисунок 2 – Прецедент 4 лабораторной работы

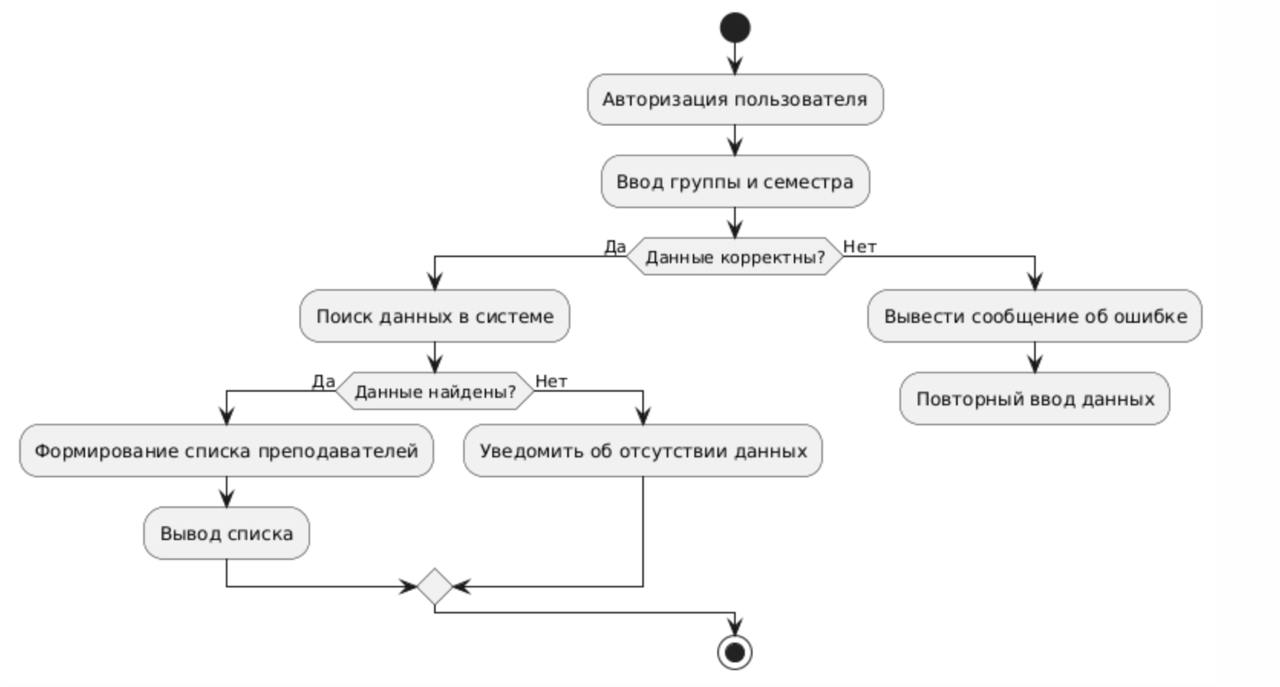


Рисунок 3 – Диаграмма состояния

# **3 Перечень сформированных логических программных классов с пояснением их назначения**

В разработанной информационной системе логически выделены несколько ключевых программных классов, каждый из которых выполняет свою конкретную роль в реализации функционала получения списка преподавателей. Класс, отвечающий за представление сотрудника деканата, инкапсулирует данные пользователя и методы для проверки его учетных данных, что обеспечивает начальный этап аутентификации. Отдельный класс, реализующий проверку подлинности и поддержку сессии, отвечает за авторизацию пользователя, передавая проверенные данные дальше по цепочке обработки. Логический класс, предназначенный для представления информации о преподавателях, содержит поля с данными о фамилиях, дисциплинах и типах занятий, что позволяет централизованно управлять данными преподавателей в системе. Класс, сфокусированный на обработке запросов по учебной группе и семестру, объединяет логику проверки вводимых параметров, инициирует поиск соответствующих данных и управляет прохождением основного сценария, а также альтернативных потоков, таких как обработка ошибок ввода или отсутствие данных. Кроме того, присутствует специализированный класс для организации взаимодействия с базой данных, который обеспечивает выполнение запросов и возврат результатов для дальнейшей обработки системой. Для управления взаимодействием с конечным пользователем используется класс, реализующий функции отображения информации и получения ввода, в том числе вывод сформированного списка преподавателей или сообщений об ошибках, что позволяет обеспечить корректное и своевременное взаимодействие между пользователем и остальными компонентами системы.

# **4 Диаграмма взаимодействия действий для прецедента**

Для визуализации взаимодействия объектов при реализации основного сценария прецедента «Получить список преподавателей» была построена диаграмма взаимодействия типа «диаграмма последовательности». На данной диаграмме отражено поэтапное взаимодействие между участниками процесса: сотрудником деканата, пользовательским интерфейсом, серверной логикой и базой данных. Диаграмма начинается с авторизации сотрудника в системе, включающей ввод учетных данных, их отправку на сервер и проверку в базе данных. После успешного входа пользователь вводит параметры запроса — номер группы и семестр — которые передаются через интерфейс к серверу, инициируя основной бизнес-процесс. Сервер обращается к базе данных с соответствующим запросом, получает результат и формирует список преподавателей. Далее результат возвращается через интерфейс пользователю, который получает на экран готовую информацию о преподавателях. Диаграмма также содержит развилки, отражающие возможные альтернативы: некорректный ввод параметров и отсутствие данных в базе. Такое поэтапное отображение помогает наглядно представить взаимодействие логических компонентов системы и обеспечивает более глубокое понимание механизма реализации прецедента на уровне архитектуры. На рисунок 4 изображена диаграмма деятельности.

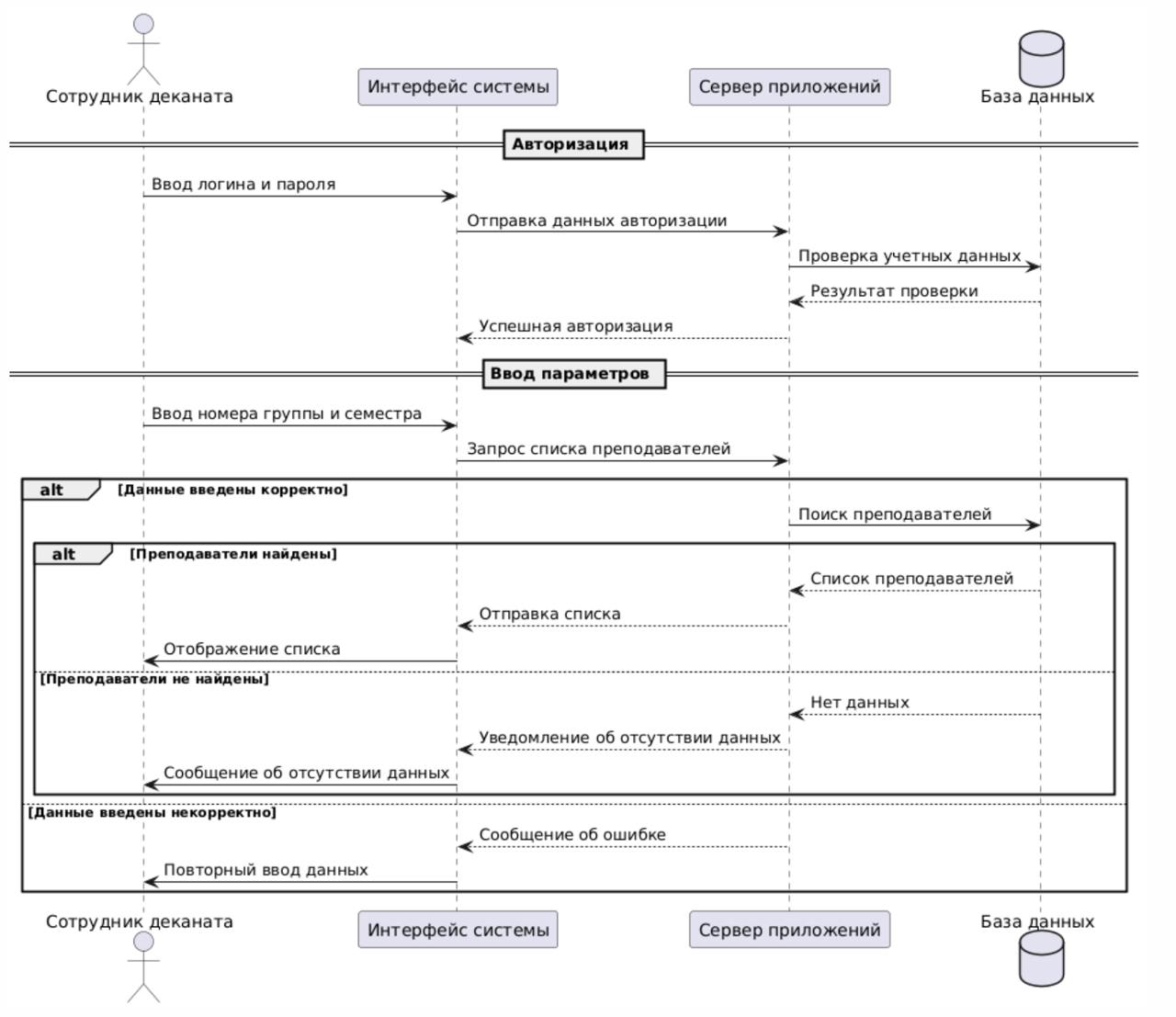


Рисунок 4 – Диаграмма деятельности

# **5 Вывод**

В ходе выполнения лабораторной работы была разработана диаграмма взаимодействия, отражающая реализацию основного сценария прецедента «Получить список преподавателей» в информационной системе вуза. Использование диаграммы последовательности позволило наглядно представить этапы взаимодействия между пользователем, интерфейсом, серверной логикой и базой данных. В процессе построения модели были выделены основные логические классы, реализующие ключевые функции системы: авторизацию, обработку запросов, работу с данными преподавателей и взаимодействие с базой данных. Разработка диаграммы позволила проследить весь путь запроса от момента ввода параметров до отображения результатов, а также учесть возможные отклонения от основного сценария. Выполнение работы способствовало углублению понимания архитектуры взаимодействий в информационных системах, а также закреплению навыков проектирования с использованием UML.