

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ
ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное
учреждение
высшего образования
«КУБАНСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
(ФГБОУ ВО «КубГУ»)

**Факультет компьютерных технологий и прикладной
математики
Кафедра информационных технологий**

ЛАБОРАТОРНАЯ РАБОТА №5

Работу выполнил _____ А. А. Иванов
(подпись)

Направление подготовки 02.03.03 Математическое обеспечение и
администрирование информационных систем

Направленность Технология проектирования программного
обеспечения

Руководитель _____ Н.Ю. Добровольская
(подпись)

Краснодар
2024

Тема: проектирования функциональной структуры программного продукта; объектно-ориентированный подход.

Цель: изучение объектно-ориентированного подхода программной инженерии для разработки и описания функциональности разрабатываемого программного обеспечения.

Задание

1. Проанализировать описание функционирования программной системы, разработанного при выполнении лабораторной работы №4, на предмет выявления набора абстракций предметной области проектируемой ПС. В качестве предварительных кандидатов в абстракции принять подлежащее, выделенные из текста анализируемого потока событий.
2. Разделить выделенные абстракции на 3 типа: абстракции сущности, абстракции поведения, абстракции интерфейсы. Результат представить в виде таблицы. Для каждой абстракции указать её класс согласно следующей классификации: Люди; Места; Предметы; Инструменты; Организации; Концепции; События; Показатели.
3. Проанализировать поведение выделенных абстракций. Выделить возможное поведение каждой абстракции в пределах функциональности проектируемой ПС, представленной моделью требований UML. Заполнить таблицу.
4. Построить диаграмму классов UML, указывая при этом лишь имена классов без указания свойств класса.
5. На основе анализа описания предметной области, разработанного при выполнении лабораторной работы №1,

выявить атрибуты и операции классов. Заполнить секции атрибутов и операций классов.

6. Выбрать в модели классов такой класс, который характеризуется наиболее частой сменой состояний, и построить для него диаграмму состояния.
7. На основе анализа функциональных моделей, разработанных при выполнении лабораторной работы №4, для каждого из базовых вариантов использования построить диаграмму деятельности. Для вариантов использования, с которыми связаны несколько действующих лиц, диаграмму деятельности построить в виде дорожек с привязкой к исполнителям конкретных операций алгоритма.
8. Для каждого варианта использования выделить список объектов участвующих во взаимодействии в этом прецеденте, заполнить таблицу.
9. Создать диаграммы последовательностей для перечисленных прецедентов (одну диаграмму для всех объектов).
10. Для наиболее сложных диаграмм последовательности создать кооперативные диаграммы и доработать их, если это необходимо.

1 Описание абстракций и их представление

Проектируемая программная система направлена на создание веб-портала, на котором пользователи могут ознакомиться с каталогом и записаться на курс(ы). Таким образом, основными абстракциями могут являться “Пользователь”, “Курс”, “Заявка записи на курс(ы)”, “Преподаватель”, “Уведомление о записи на курс(ы)”. Абстракция “Пользователь” является основной, так как все остальные (кроме “Преподавателя”) появляются в результате её действий.

Данные абстракции можно разбить на 3 группы и представить в виде таблицы 1.

№	Абстракция	Тип	Класс	Описание
1	Пользователь	Абстракция сущности	Люди	Основная абстракция, приводит к формированию остальных
2	Курс	Абстракция сущности	Предметы	Абстракция(и), из которой(ых) формируется абстракция “Заявка записи на курс(ы)”
3	Заявка записи на курс(ы)	Абстракция поведения	События	Абстракция, которая приводит к изменению базы данных
4	Преподаватель	Абстракция сущности	Люди	Абстракция, которая получает “Уведомление о

				записи на курс(ы)”
5	Уведомление о записи на курс(ы)	Абстракция интерфейса	Концепции	Абстракция, которая нужна для информирования абстракции “Преподаватель”

Таблица 1 – абстракции подсистемы

Так же можно сформировать возможное поведение каждой абстракции в проектируемой ПС и представить в виде таблицы 2.

№	Абстракция	Требование согласно модели UML	Описание поведения
1	Пользователь	Наличие персональных данных	Регистрация, внесение персональных данных, вход в систему, просмотр и запись на курсы
2	Курс	Наличие цены (плана обучения)	Добавление/удаление курсов
3	Заявка записи на курс(ы)	Содержит данные пользователя и выбранный(ые) курс(ы)	Создаётся автоматически, изменяет информацию в базе данных
4	Преподаватель	Наличие списка преподаваемых курсов	Просмотр списка курсов, оценивание слушателей(пользователей)
5	Уведомление о записи на курс(ы)	Содержит данные пользователя и выбранный(ые) курс(ы)	Создаётся автоматически и отправляется преподавателю

Таблица 2 – абстракции подсистемы и их поведение

2 Диаграмма классов UML

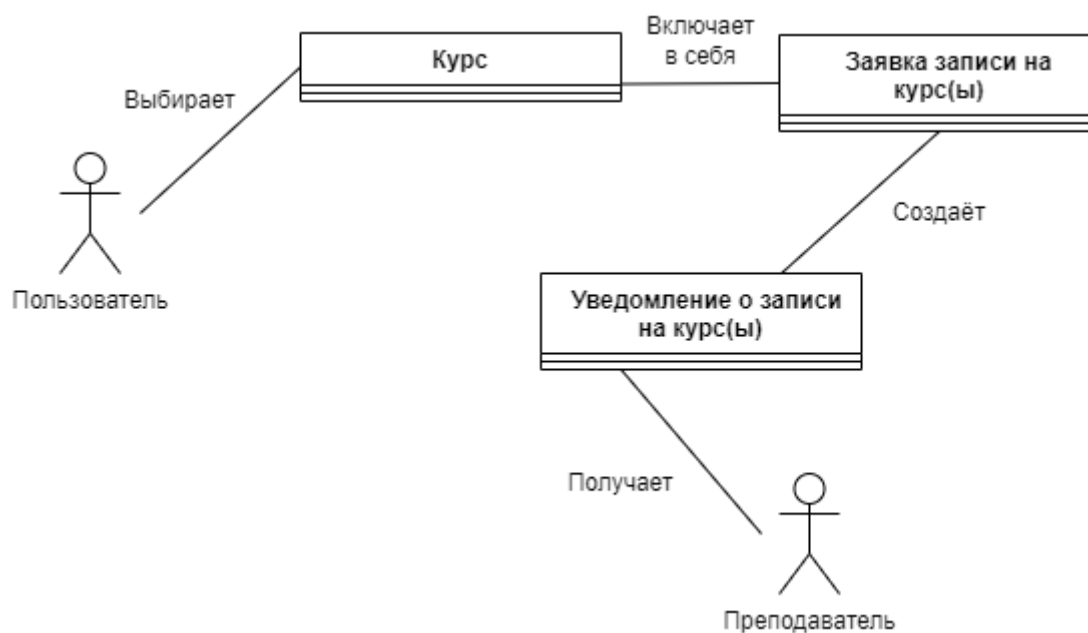


Рисунок 1 – диаграмма классов UML

Теперь можно попробовать выявить атрибуты и операции соответствующих классов, которые можно представить в виде таблицы 3.

Имя класса	Список атрибутов	Список операций
Курс	Название Стоимость Описание	Создать() Удалить()
Заявка записи на курс(ы)	Данные пользователя Статус	Создать() Отправить()
Уведомление записи на	Данные пользователя	Создать()

курс(ы)	Адрес получателя	Отправить()
---------	------------------	-------------

Таблица 3 – атрибуты и операции классов

Класс “Заявка записи на курс(ы)” является классом с наиболее частой сменой состояний, потому что на этот класс оказывает влияния наибольшее число внешних факторов, так что для него нужно построить диаграмму состояний.

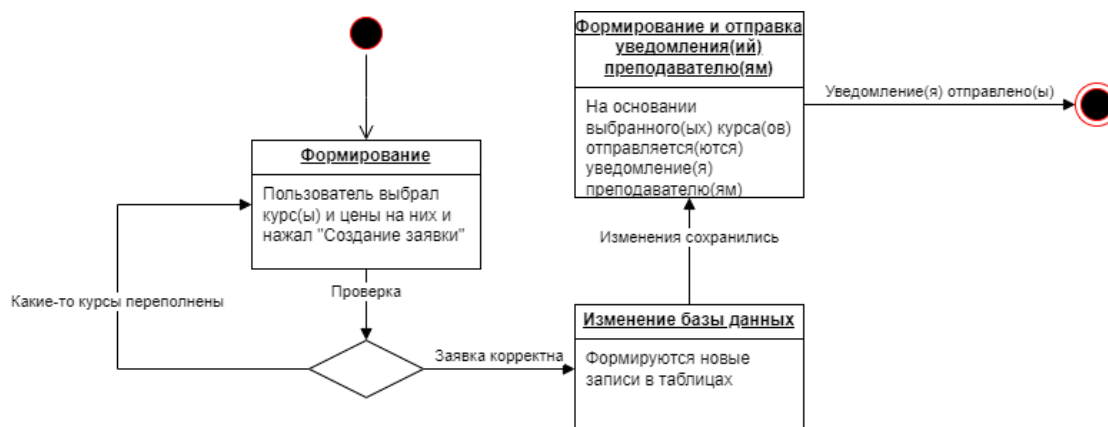


Рисунок 2 – диаграмма состояний класса “Заявка записи на курс(ы)”

3 Диаграмма деятельности

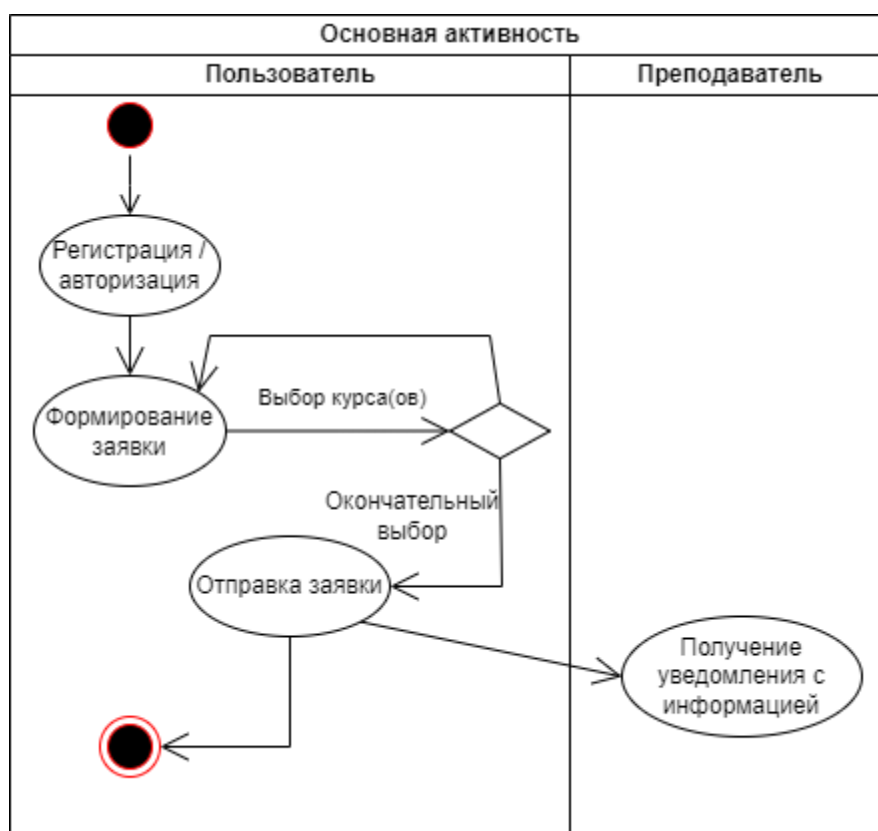


Рисунок 3 – диаграмма деятельности

В таблице 4 представлены объекты, участвующие в данном прецеденте, и их описание.

№	Прецедент	Объект	Описание объекта
1	Оформление и отправка заявки записи на курс(ы)	Пользователь	Инициатор данного прецедента
		Преподаватель	Участник прецедента, который не принимает активных действий

		Заявка записи на курс(ы)	Первый результат прецедента
		Уведомление о записи на курс(ы)	Второй результат прецедента

Таблица 4 – описание прецедента

Для рассмотрения жизненного цикла всего прецедента нужно составить диаграмму последовательности.

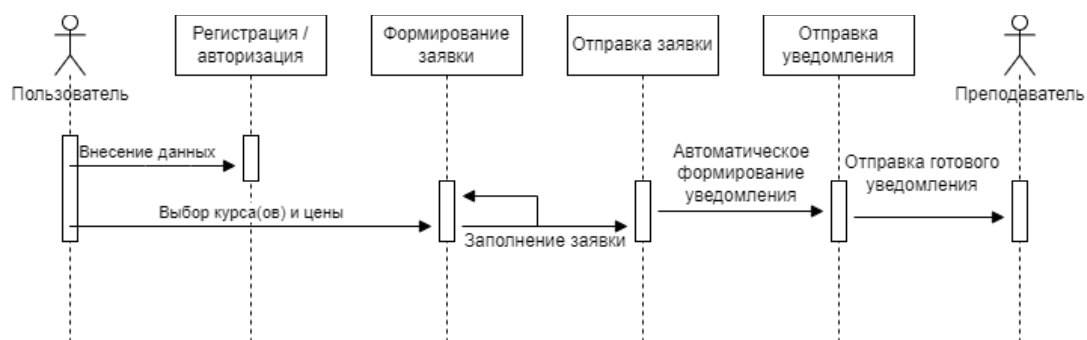


Рисунок 4 – диаграмма последовательности