**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ   
РОССИЙСКОЙ ФЕДЕРАЦИИ**

**Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Университет «Дубна»**

**Институт системного анализа и управления**

**Кафедра системного анализа и управления**

**КУРСОВАЯ РАБОТА**

по дисциплине

**«Проектирование информационных систем»**

на тему:

**«Проектирование информационной системы   
для онлайн-школы «Школопутино»**

Выполнили:

студенты группы 3254

Смирнов Вадим Николаевич

Оценка:

Подпись:

Яковлев Артем Михайлович

Оценка:

Подпись:

Руководитель:

Ассистент, Дидоренко Алексей Викторович

Дата защиты:

Подпись:

Дубна, 2025

**Оглавление**

[Введение 4](#_Toc198120662)

[Анализ предметной области 5](#_Toc198120663)

[Постановка задачи 8](#_Toc198120664)

[Цель 8](#_Toc198120665)

[Исходные данные 8](#_Toc198120666)

[Априорные модельные представления 8](#_Toc198120667)

[Результат 9](#_Toc198120668)

[Критерии оценки результата 9](#_Toc198120669)

[Средства проектирования и реализации 9](#_Toc198120670)

[Требования к информационной системе 10](#_Toc198120671)

[Функциональные требования 10](#_Toc198120672)

[Нефункциональные требования 10](#_Toc198120673)

[Функциональная модель системы 12](#_Toc198120674)

[Диаграмма вариантов использования 12](#_Toc198120675)

[Детальное описание прецедентов 12](#_Toc198120676)

[Диаграммы деятельности 16](#_Toc198120677)

[Диаграммы взаимодействия 17](#_Toc198120678)

[Диаграммы состояний 19](#_Toc198120679)

[Диаграмма классов 20](#_Toc198120680)

[Модель данных 22](#_Toc198120681)

[Интерфейс информационной системы 26](#_Toc198120682)

[Программная реализация 29](#_Toc198120683)

[Серверная часть 29](#_Toc198120684)

[Клиентская часть 31](#_Toc198120685)

[Заключение 35](#_Toc198120686)

[Список использованной литературы 36](#_Toc198120687)

[Приложение 37](#_Toc198120688)

# Введение

В современном мире образование играет важнейшую роль, являясь значимой сферой социальной жизни. С каждым годом возрастает значимость интеллектуального труда, информационных ресурсов и скорости их обработки, что делает информационные технологии неотъемлемой частью образовательной среды.

**Информационные технологии (ИТ)** — это совокупность методов, производственных методов, производственных процессов и программно-технических средств, объединенных в технологическую цепочку, обеспечивающую сбор, хранение, обработку, распространение и отображение информации с целью снижения трудоемкости процессов использования информационных ресурсов, повышения их надежности и оперативности, рациональной организации часто повторяющихся информационных процессов [1].

Интеграция информационных технологий в сферу образования предоставляет новые возможности для учащихся и для преподавателей. Она позволяет получать доступ к учебным материалам из любой точки мира, выбирать преподавателя, подходящего по стилю и времени обучения, а также оптимизировать организацию занятий. Кроме того, современные ИТ-средства автоматизируют рутинные процессы: ведение журналов, проверку заданий и анализируют успеваемость. Всё это реализуется в рамках единой информационной системы.

**Информационная система (ИС)** — это совокупность методов и средств сбора, хранения, анализа, обработки, и передачи информации, методов управления процессами решения функциональных задач, а также информационными, материальными и денежными потоками в предметной области [1].

В данной курсовой работе рассматривается процесс проектирования и разработки информационной системы дистанционного обучения для онлайн-школы «Школопутино», направленной на подготовку учащихся к ОГЭ, ЕГЭ и дошкольному обучению. Система повысит качество образовательного процесса, упростит взаимодействие между участниками и обеспечит единое хранение данных.

# Анализ предметной области

Городская школа, для которой разрабатывается информационная система (ИС), специализируется на подготовке учащихся к экзаменам ОГЭ и ЕГЭ, а также на обучении дошкольников перед поступлением в первый класс. Занятия проводятся в онлайн- и в офлайн-форматах, но отсутствие единой системы управления образовательным процессом создает значительные неудобства для преподавателей, учеников и их родителей (см. рис. 1).

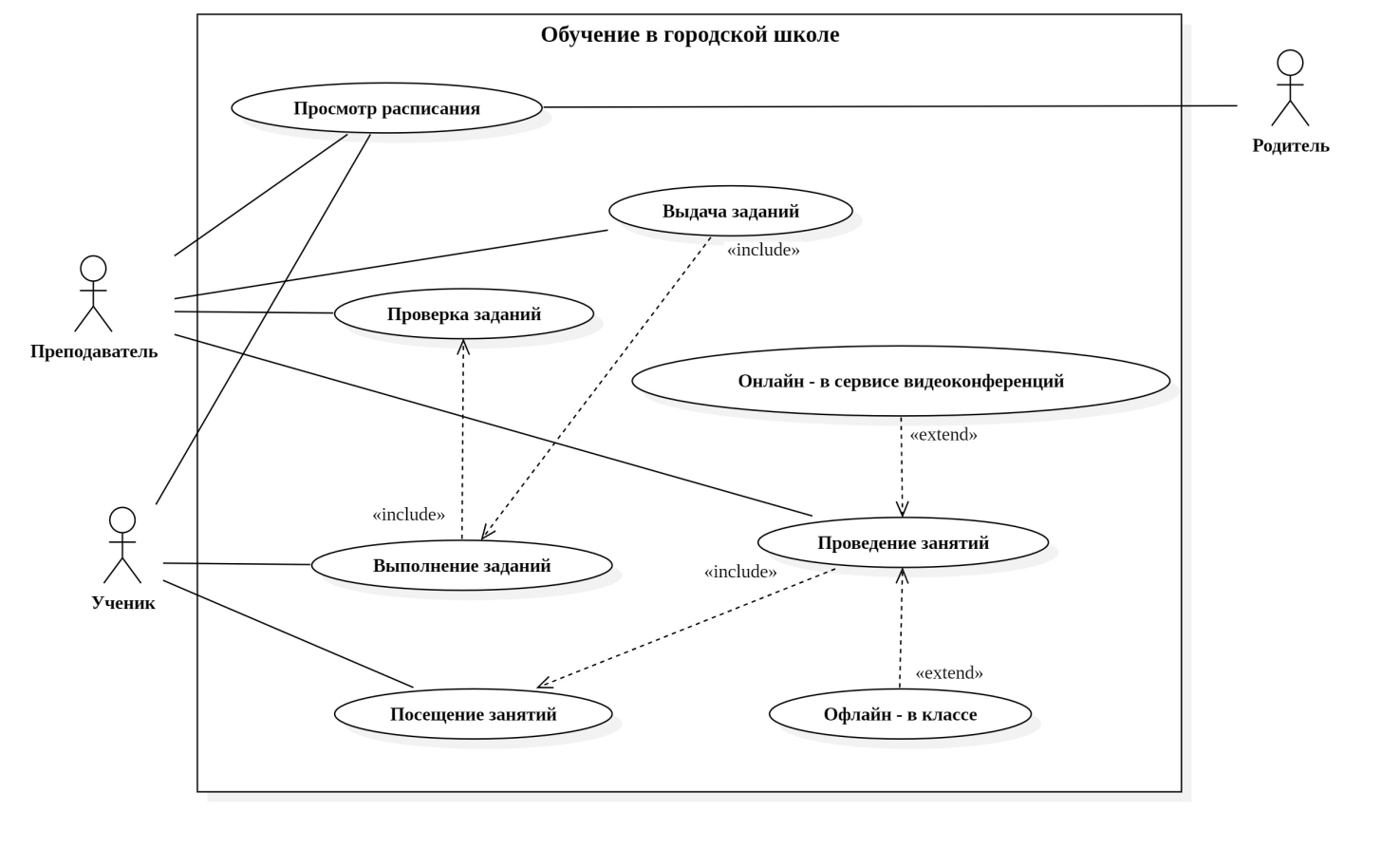


Рисунок 1. Диаграмма *USE CASE* «Как есть»

### Основные задачи школы

1. Организация образовательного процесса для различных возрастных категорий учащихся.
2. Проведение занятий в двух форматах: очном и дистанционном.
3. Оценка знаний учеников и их подготовка к государственным экзаменам.
4. Взаимодействие преподавателей, учеников и родителей для отслеживания прогресса.
5. Управление расписанием, доступом к учебным материалам и домашними заданиями.

### Основные процессы предметной области

1. Запись учащихся на курсы:

Родители или ученики выбирают подходящие курсы, регистрируются и получают доступ к материалам.

1. Формирование расписания:

Занятия распределяются по преподавателям, учитывая онлайн- и офлайн-форматы.

1. Проведение уроков:

Ученики участвуют в занятиях либо через видеоконференции, либо посещая занятия в классе.

1. Оценивание знаний:

Учителя проводят тестирования, проверяют домашние задания и фиксируют успеваемость.

1. Коммуникация между участниками:

Родители, ученики и преподаватели взаимодействуют через чаты, уведомления и электронные дневники.

### Недостатки и проблемы текущей системы

* Отсутствие единой платформы:

Сейчас все процессы разрознены: расписание, домашние задания и отчеты ведутся в разных местах (мессенджеры, таблицы, бумажные журналы).

* Контроля прогресс:

У родителей и учеников нет инструмента для отслеживания успеваемости и посещаемости.

* Разные виды коммуникации:

Информация о расписании, изменениях и заданиях передается через несколько платформ, что создает путаницу.

* Ограниченный доступ к материалам:

Ученикам неудобно получать и систематизировать учебные материалы, так как они хранятся в разных местах.

На рисунках 2 и 3 представлены диаграммы деятельности для прецедентов «Выдача заданий» и «Проверка заданий» в условиях отсутствия информационной системы. Эти диаграммы наглядно демонстрируют основные недостатки существующего процесса, такие как высокая трудозатратность, низкая скорость обработки информации и вероятность ошибок из-за человеческого фактора.

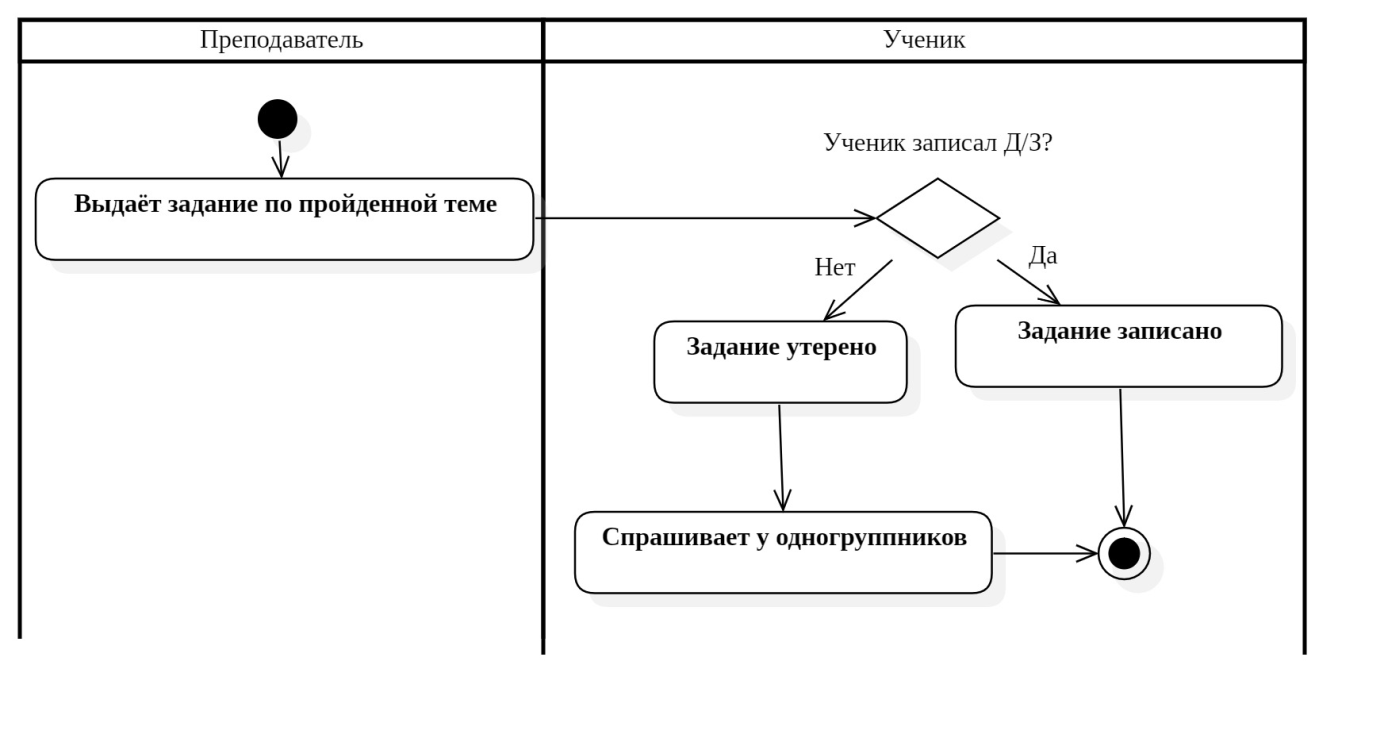


Рисунок 2. Диаграмма деятельности прецедента «Выдача заданий» вне ИС

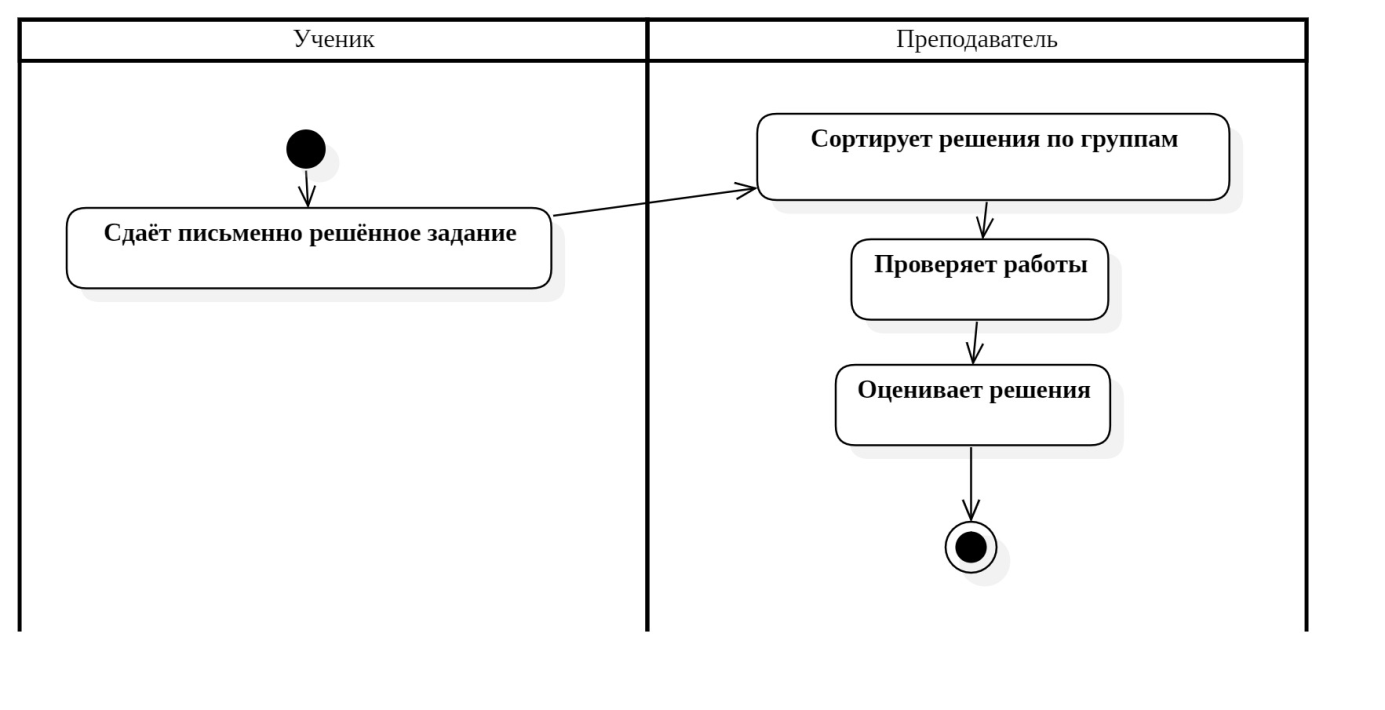


Рисунок 3. Диаграмма деятельности прецедента «Проверка заданий» вне ИС

### Вывод

Создание единой информационной системы позволит автоматизировать образовательные процессы, повысить удобство взаимодействия всех участников и улучшить качество обучения.

# Постановка задачи

## Цель

Целью данной работы является создание информационной системы для онлайн школы «Школопутино» по подготовке выпускников к экзаменам (ОГЭ и ЕГЭ) и дошкольников.

Система позволит:

* повысить качество и эффективность подготовки;
* автоматизировать процесс записи и формирования учебных групп;
* обеспечить удобное управление расписанием занятий;
* улучшить контроль над успеваемостью учащихся и доступ к учебным материалам;
* оптимизировать взаимодействие между учениками, родителями и преподавателями;
* автоматизированное составление расписания с учетом занятости преподавателей.

## Исходные данные

В качестве исходных данных для проектирования информационной системы выступают:

* требования к экзаменам (ОГЭ, ЕГЭ);
* регламент школы;
* государственные образовательные стандарты;
* учебные материалы.

## Априорные модельные представления

С проектируемой информационной системой будут работать следующие пользователи: преподаватели, ученики, родители.

1. Преподаватель:
   * проводит занятия;
   * оценивает знания учащихся;
   * загружает учебные материалы.
2. Ученик:
   * посещает занятия;
   * получает доступ к учебным материалам;
   * выполняет домашние задания;
   * проходит тестирования;
   * просматривает свои оценки и успеваемость.
3. Родитель:
   * просматривает успеваемость ребенка;
   * контролирует посещаемость занятий ребенком.

## Результат

Проект и частично реализованная информационная система для школы подготовки к экзаменам «Школопутино», которая будет обладать простым и удобным интерфейсом, централизованно хранить данные об учениках, преподавателях и курсах, автоматизировать процесс обучения, включая формирование расписания, проверку знаний и управление учебными материалами.

## Критерии оценки результата

Критериями оценки результата являются:

* Сокращение времени на администрирование учебного процесса на 40%.
* Экономия на печатных материалах и отчетности до 30% за счет перехода на цифровые учебные материалы и электронные документы.
* Повышение контроля успеваемости и посещаемости на 25% за счет отчетности для преподавателей и родителей.

## Средства проектирования и реализации

Для разработки информационной системы использованы следующие *CASE*-средства верхнего уровня: *Erwin Data Modeler r*7 и *StarUML*.

Для создания программной реализации будет использованы фреймворк языка программирования *python* – *Django* и фреймворк для реализации пользовательского интерфейса *Flutter*, а также система управления версиями *GitHub*. Хранение данных реализовано через СУБД *SQLite*.

# Требования к информационной системе

## Функциональные требования

Функциональные требования удобнее разделить по категориям лиц, которые будут взаимодействовать с информационной системой.

* Для преподавателя:
  + Добавление учебных материалов;
  + Изменение учебных материалов;
  + Удаление учебных материалов;
  + Проведение онлайн занятий;
  + Составление расписания;
  + Редактирование расписания;
  + Проверка заданий;
  + Оценка заданий.
* Для ученика:
  + Посещение онлайн занятий;
  + Просмотр учебных материалов;
  + Выполнение заданий;
  + Просмотр успеваемости.
* Для родителя:
  + Просмотр успеваемости ребёнка.
* Общие требования:
  + Авторизация пользователей в соответствие *CRUD*;
  + Просмотр расписания.

## Нефункциональные требования

### Требования к данным

В системе должны храниться следующие данные:

* о пользователях и их правах, которые регулируют действия и возможности пользователей (*id* пользователя, логин, пароль, имя, фамилия, отчество, номер телефона);
* данные о дисциплинах (*id* предмета, наименование предмета, описание);
* данные о занятиях (*id*, тема, дата);
* данные о заданиях (*id*, описание, дедлайн);
* данные об ответах на задания (*id*, статус, оценка);
* данные о расписании (*id*, время, дата).

### Требования к интерфейсу

Интерфейс системы должен быть интуитивно понятным, удобным и доступным для всех категорий пользователей: администратора, преподавателя, ученика и родителя.

Регистрация и вход должны быть реализованы с поддержкой аутентификации (логин, пароль) и возможностью восстановления доступа. Личный кабинет каждого пользователя должен содержать персональные данные.

Для учеников интерфейс должен обеспечивать удобный просмотр расписания, доступ к учебным материалам, тестам и домашним заданиям. Для преподавателей – удобную форму ведения электронного журнала, загрузки материалов и проверки работ. Родители должны иметь доступ к информации об успеваемости и посещаемости ребенка.

### Требования к разграничению доступа

Должна быть реализована система доступа, которая разграничивает функциональные возможности (*CRUD*) для каждого типа пользователей.

# Функциональная модель системы

Функциональная модель системы представлена в объектно-ориентированном подходе проектирования информационных систем. Ниже представлены диаграммы, описывающие работу системы.

## Диаграмма вариантов использования

**Диаграмма вариантов использования или диаграмма прецедентов** (*use* *case*) — это диаграмма, отражающая отношения между акторами и прецедентами и являющаяся составной частью модели прецедентов, позволяющей описать систему на концептуальном уровне.

Диаграмма прецедентов информационной системы представлена на рисунке 4. На ней изображено 3 актора (Преподаватель, Ученик и Родитель) и их варианты их взаимодействия с системой.

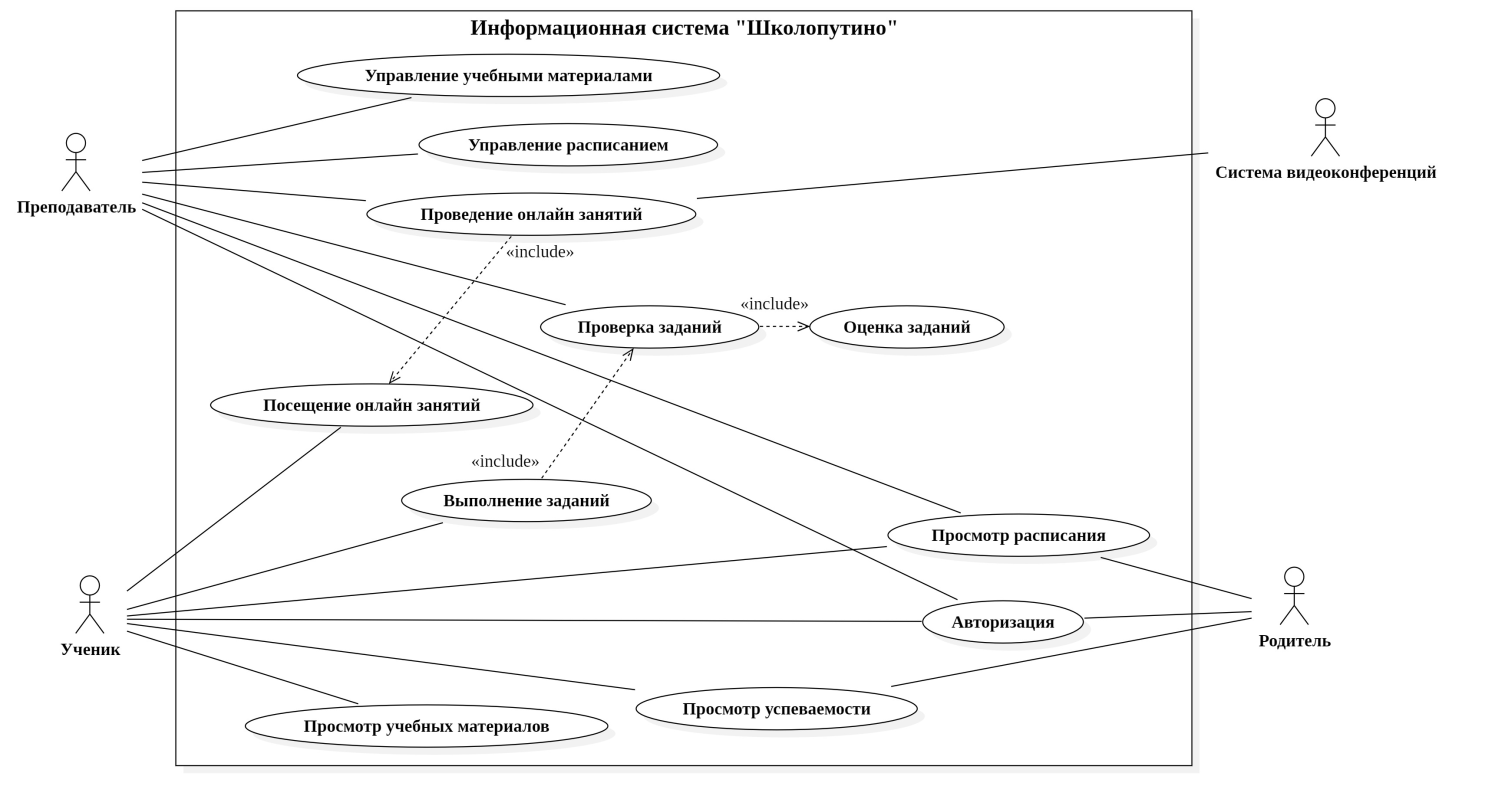


Рисунок 4. Диаграмма *USE CASE* «Как будет»

## Детальное описание прецедентов

### Вход в систему пользователя

Регистрация пользователей в системе осуществляется техническим специалистом школы через учётную запись администратора. После этого, пользователь может авторизоваться по выданному логину и паролю.

Табл. 1. Описание прецедента «Авторизация»

|  |  |
| --- | --- |
| Название прецедента | Авторизация |
| Основное действующее лицо | Пользователь (Ученик, Родитель, Преподаватель) |
| Цель | Войти в систему |
| Триггер | Необходимость авторизоваться |
| Результат | Пользователь вошел в систему |

### Просмотр расписания пользователем

Пользователь может перейти на страницу с расписанием и посмотреть его.

Табл. 2. Описание прецедента «Просмотр расписания»

|  |  |
| --- | --- |
| Название прецедента | Просмотр расписания |
| Основное действующее лицо | Пользователь (Ученик, Родитель, Преподаватель) |
| Цель | Узнать расписания занятий |
| Триггер | Переход на страничку с расписанием |
| Результат | Просмотр расписания занятий |

### Просмотр успеваемости ученика

Ученики и их родители могут просматривать успеваемость на странице журнала успеваемости.

Табл. 3. Описание прецедента «Просмотр успеваемости»

|  |  |
| --- | --- |
| Название прецедента | Просмотр успеваемости |
| Основное действующее лицо | Пользователь (Ученик, Родитель) |
| Цель | Узнать текущую успеваемость ученика |
| Триггер | Переход на страницу с журналом успеваемости |
| Результат | Просмотр успеваемости ученика по каждому из предметов |

### Просмотр учебных материалов учеником

Ученик получает доступ к учебным материалам по предметам на соответствующей странице.

Табл. 4. Описание прецедента «Просмотр учебных материалов»

|  |  |
| --- | --- |
| Название прецедента | Просмотр учебных материалов |
| Основное действующее лицо | Пользователь (Ученик) |
| Цель | Изучить учебные материалы по предмету |
| Триггер | Переход на страницу с учебными материалами |
| Результат | Получение учебных материалов преподавателя |

### Посещение онлайн занятий учеником

Ученик посещает онлайн занятия через систему видеоконфиренций на соответствующей странице.

Табл. 5. Описание прецедента «Посещение онлайн занятий»

|  |  |
| --- | --- |
| Название прецедента | Посещение онлайн занятий |
| Основное действующее лицо | Пользователь (Ученик) |
| Цель | Посетить онлайн занятие |
| Триггер | Переход на страницу занятия |
| Результат | Посещённое онлайн занятие |

### Выполнение заданий учеником

Ученик должен выполнять задания, которые выкладываются преподавателями.

Табл. 6. Описание прецедента «Выполнение заданий»

|  |  |
| --- | --- |
| Название прецедента | Выполнение заданий |
| Основное действующее лицо | Пользователь (Ученик) |
| Цель | Выполнить задание |
| Триггер | Переход на страницу с заданиями |
| Результат | Выполнение заданий, выложенных преподавателем |

### Управление учебными материалами преподавателем

Преподаватель может выкладывать, изменять и удалять учебные материалы по его предметам.

Табл. 7. Описание прецедента «Управление учебными материалами»

|  |  |
| --- | --- |
| Название прецедента | Управление учебными материалами |
| Основное действующее лицо | Пользователь (Преподаватель) |
| Цель | Поддержание актуальных учебных материалов |
| Триггер | Внесение изменений на странице с учебными материалами |
| Результат | Наличие актуальных учебных материалов по предметам |

### Управление расписанием

Преподаватель может вносить изменения в расписание.

Табл. 8. Описание прецедента «Управление расписанием»

|  |  |
| --- | --- |
| Название прецедента | Управление расписанием |
| Основное действующее лицо | Пользователь (Преподаватель) |
| Цель | Поддержание актуального расписания |
| Триггер | Внесение изменений на странице с расписанием |
| Результат | Наличие актуального расписания |

### Проведение онлайн занятия

Преподаватель проводят онлайн занятия через систему видеоконференций. Система автоматически фиксирует, кто из учеников посетил заниятие.

Табл. 9. Описание прецедента «Проведение онлайн занятия»

|  |  |
| --- | --- |
| Название прецедента | Проведение онлайн занятия |
| Основное действующее лицо | Пользователь (Преподаватель) |
| Цель | Проведение занятия |
| Триггер | Создание онлайн сессии |
| Результат | Видео проведённого онлайн занятия |

### Проверка заданий преподавателем

Преподаватель может добавлять тесты, которые проверяются автоматически, и задания, которые после решения нужно проверять вручную.

Табл. 10. Описание прецедента «Проверка заданий»

|  |  |
| --- | --- |
| Название прецедента | Проверка заданий |
| Основное действующее лицо | Пользователь (Преподаватель) |
| Цель | Проверить решённое учеником задание |
| Триггер | Ученик решил задание |
| Результат | Проверенное задание |

### Оценка заданий

Преподаватель оценивает проверенные задания и выставляет оценку, которая вносится в журнал успеваемости.

Табл. 11. Описание прецедента «Оценка заданий»

|  |  |
| --- | --- |
| Название прецедента | Оценка заданий |
| Основное действующее лицо | Пользователь (Преподаватель) |
| Цель | Оценить успеваемость ученика |
| Триггер | Задание проверено системой или преподавателем |
| Результат | Оцененное задание |

## Диаграммы деятельности

Диаграммы деятельности служат для моделирования последовательности действий, которые выполняются различными элементами, входящими в состав системы. [2] Эта диаграмма состоит из следующих элементов:

* начальный узел — изображается в виде закрашенного круга и обозначат начало процесса;
* действие — изображается в виде овала и обозначает определенное действие;
* поток управления — изображается в виде стрелки и обозначает последовательность действий;
* узел принятия решения — изображается в виде ромба и отражает принятие решения по выбору одного из маршрутов;
* точка разделения— изображается как толстая горизонтальная или вертикальная линия и обеспечивает разделение одного потока на несколько параллельных потоков;
* точка слияния— изображается как толстая горизонтальная или вертикальная линия и обеспечивает слияние нескольких параллельных потоков в один;
* конечный узел — изображается в виде закрашенного круга в круге и обозначает конец потока.

С помощью данной диаграммы были описаны следующие процессы:

1. проведение онлайн занятий (см. рис. 5);
2. выполнение заданий (см. рис. 6).

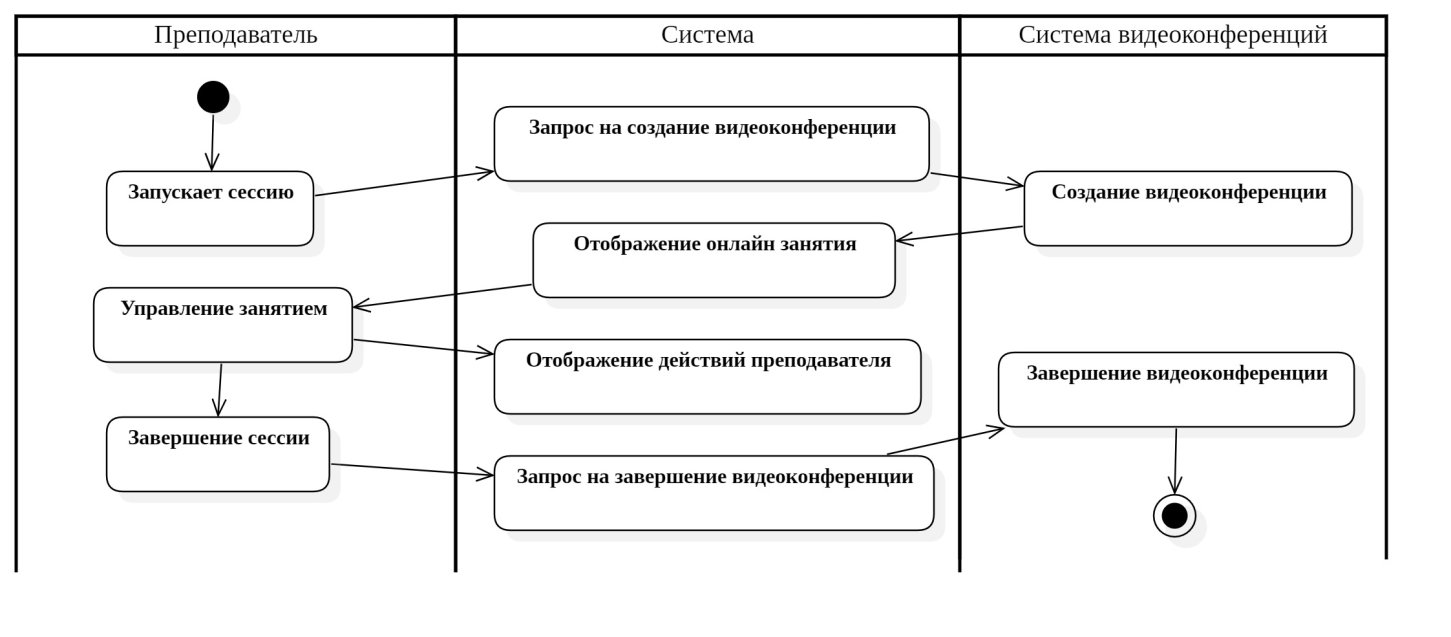


Рисунок 5. Диаграмма деятельности прецедента «Проведение онлайн занятия»

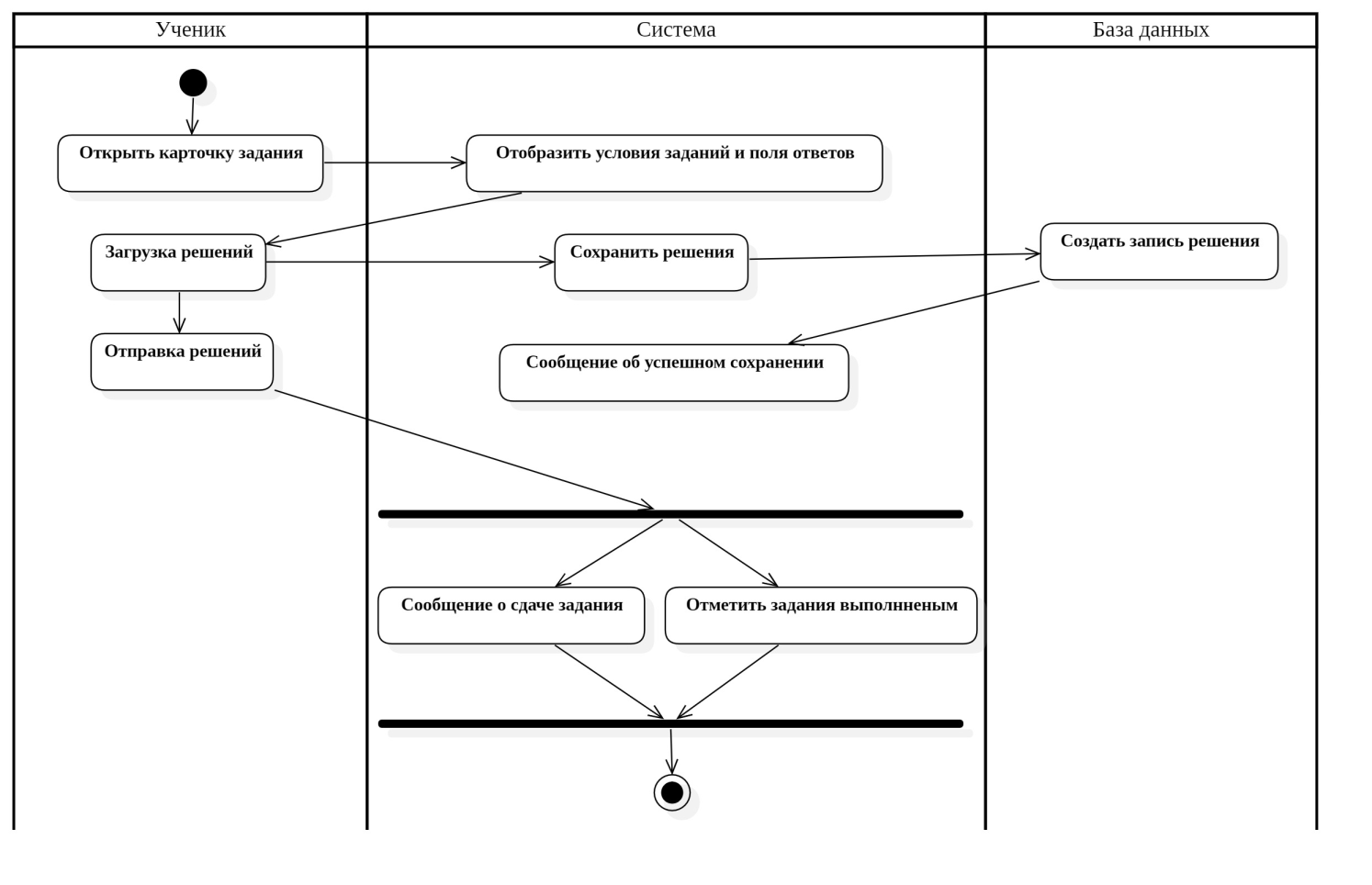


Рисунок 6. Диаграмма деятельности прецедента «Выполнение задания»

## Диаграммы взаимодействия

Диаграммы взаимодействия — это модели, описывающие поведение взаимодействующих групп объектов. На ней показаны объекты и те сообщениями, которыми обмениваются объекты между собой [1].

Существуют два вида диаграмм взаимодействия: диаграммы последовательности и кооперативные диаграммы.

Диаграммы последовательности описывают взаимодействие объектов с акцентом на последовательность взаимодействий во времени, логику и параллельность исполнения операций.

Диаграмма последовательности состоит из следующих ключевых элементов:

* линия жизни — вертикальная пунктирная линия на диаграмме последовательности, которая представляет существование объекта в течение определенного времени;
* фокус управления — специальный символ на диаграмме последовательности, указывающий период времени, в течение которого объект выполняет некоторое действие, находясь в активном состоянии;
* сообщение — спецификация передачи информации от одного элемента модели к другому с ожиданием выполнения определенных действий со стороны принимающего элемента;
* объект — нечто, содержащее в себе данные и поведение.

С помощью данной диаграммы были описаны следующие процессы:

1. оценка заданий (см. рис. 7);
2. просмотр успеваемости (см. рис. 8);
3. Просмотр учебных материалов (см. рис. 9).

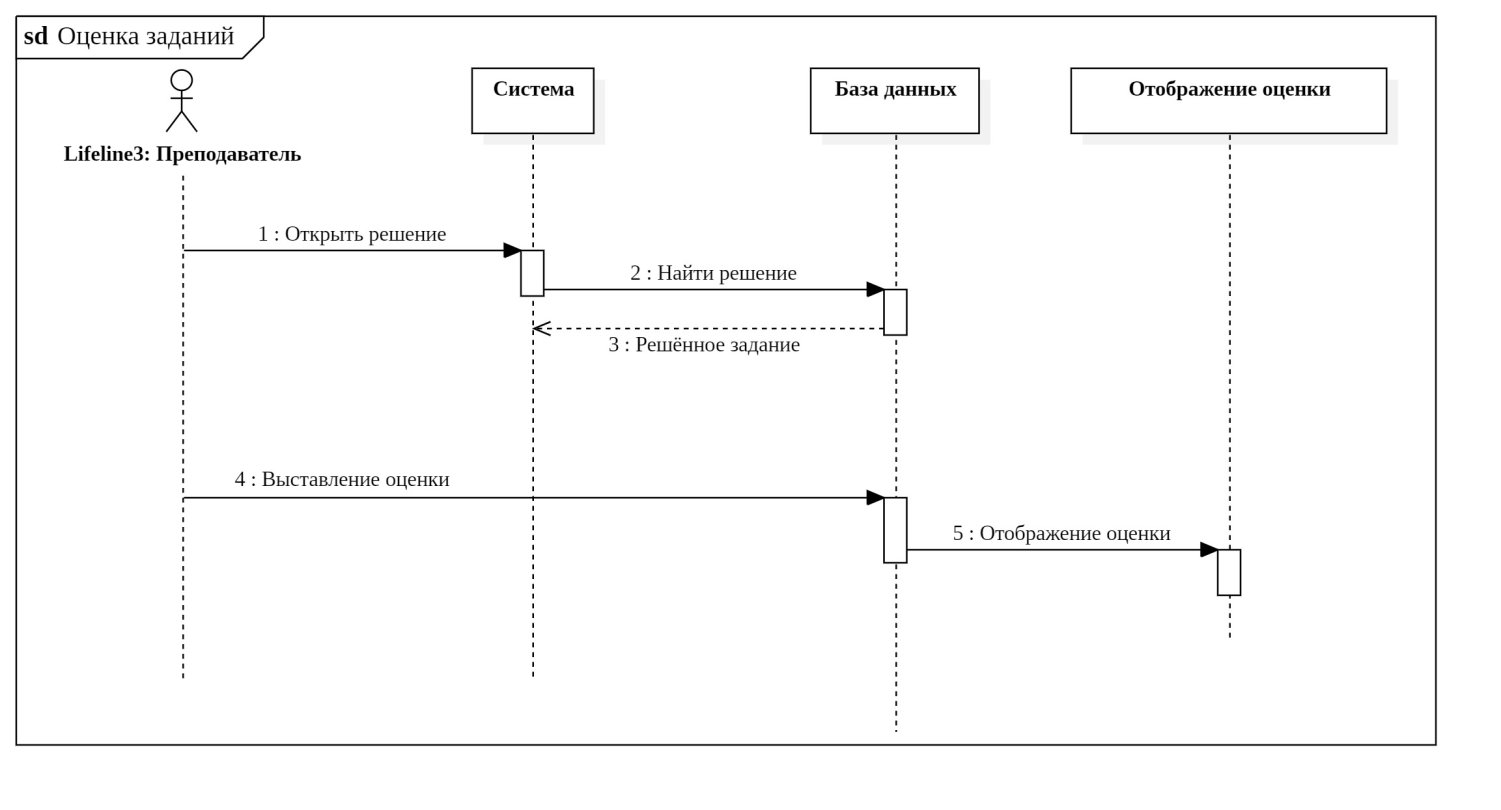


Рисунок 7. Диаграмма последовательности прецедента «Оценка заданий»

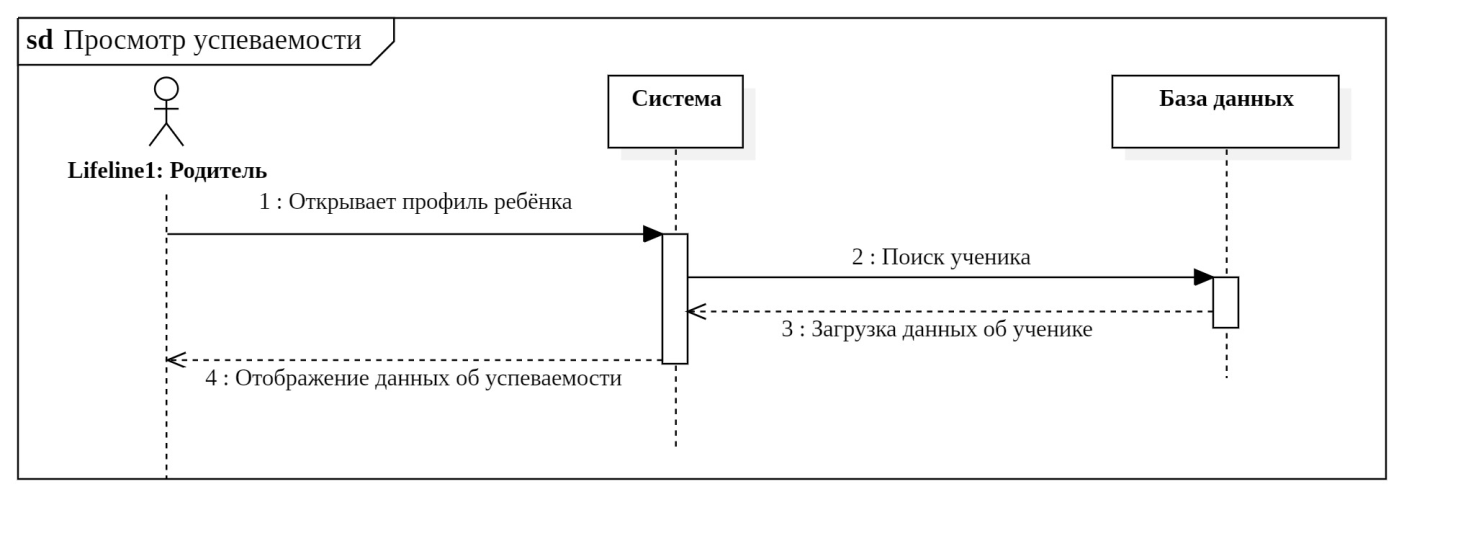


Рисунок 8. Диаграмма последовательности прецедента «Просмотр успеваемости»

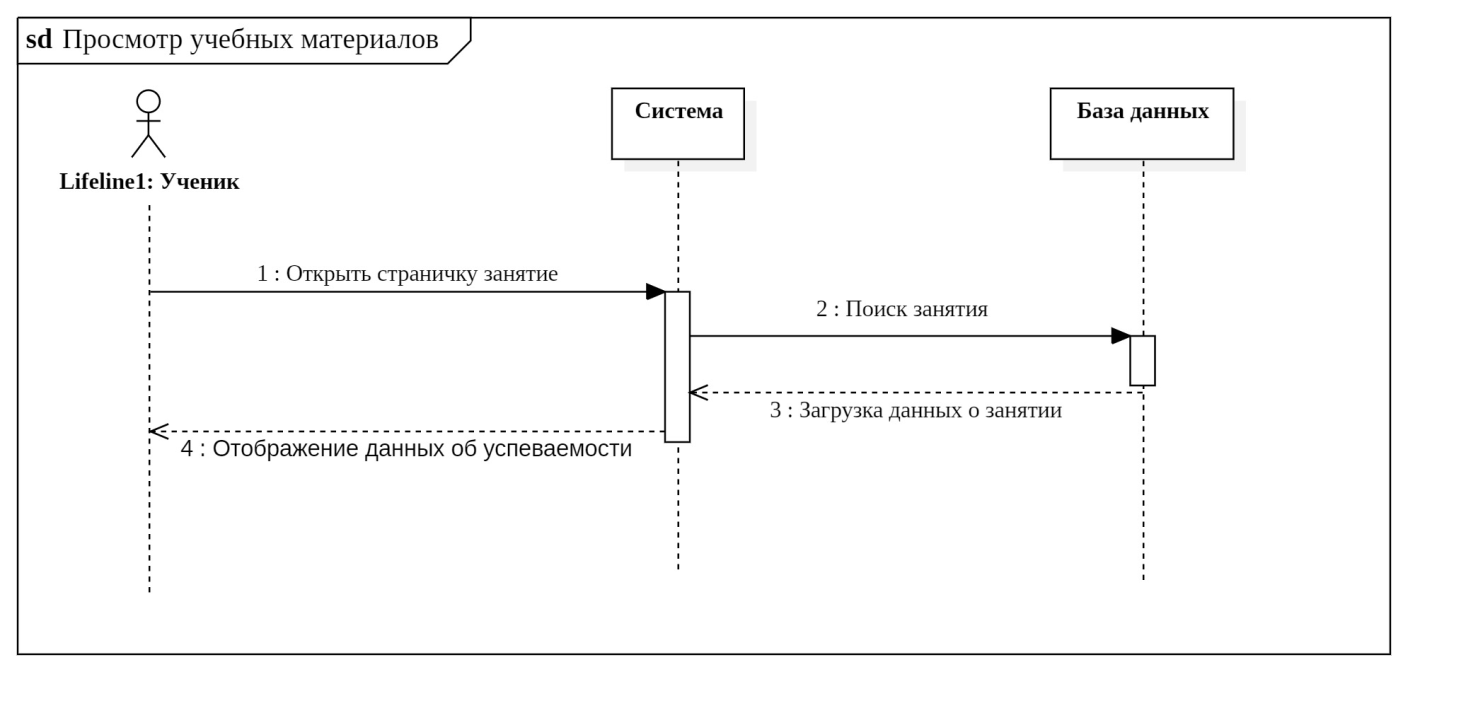


Рисунок 9. Диаграмма последовательности прецедента   
«Просмотр учебных материалов»

## Диаграммы состояний

Диаграммы состояний используются для описания поведения сложных систем. Они отражают динамику поведения единственного объекта и определяют типичные состояния, проходимые объектом, события, ведущие к изменению объекта, действия, которые будет выполнять объект после получения сообщений об изменении состояния, какие объекты создаются и какие уничтожаются (входные и выходные точки диаграммы) [1].

На рисунке 10 представлена диаграмма, на которой показаны состояния задание, при его выполнении учеником и проверки преподавателем.

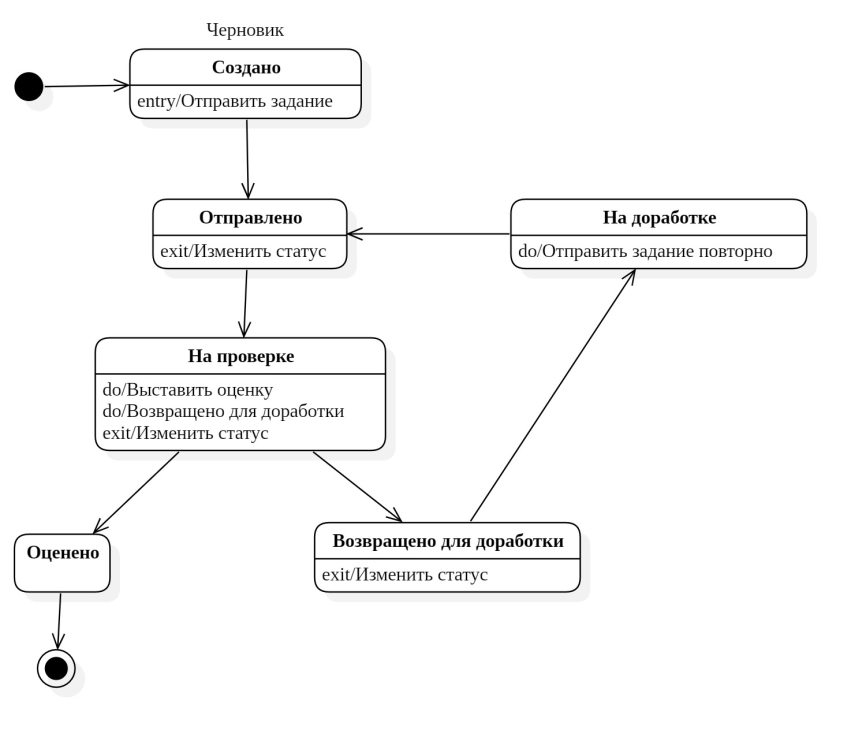


Рисунок 10. Диаграмма состояний объекта «Задание»

На рисунке 11 представлена диаграмма, на которой показаны состояния занятия, проводимого преподавателем.

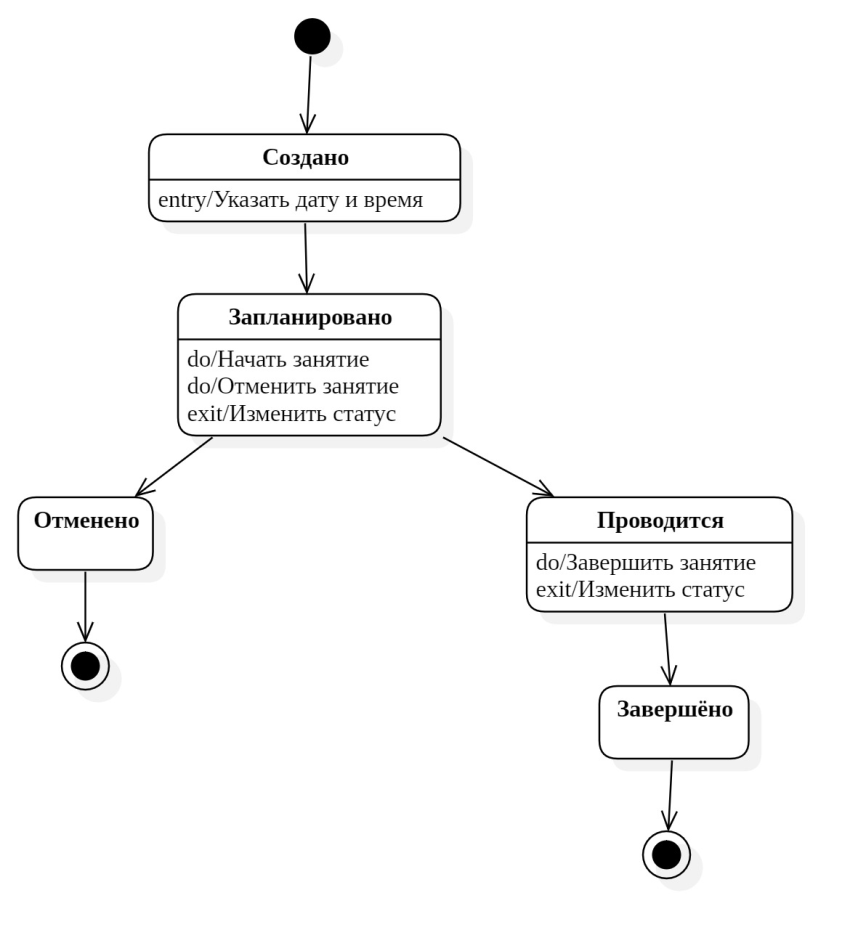


Рисунок 11. Диаграмма состояний объекта «Занятие»

## Диаграмма классов

Диаграмма классов предназначена для моделирования статической структуры классов системы и связей между ними. С помощью неё можно определить типы объектов системы, отобразить атрибуты, операции классов и ограничения, накладываемые на связи между объектами [1].

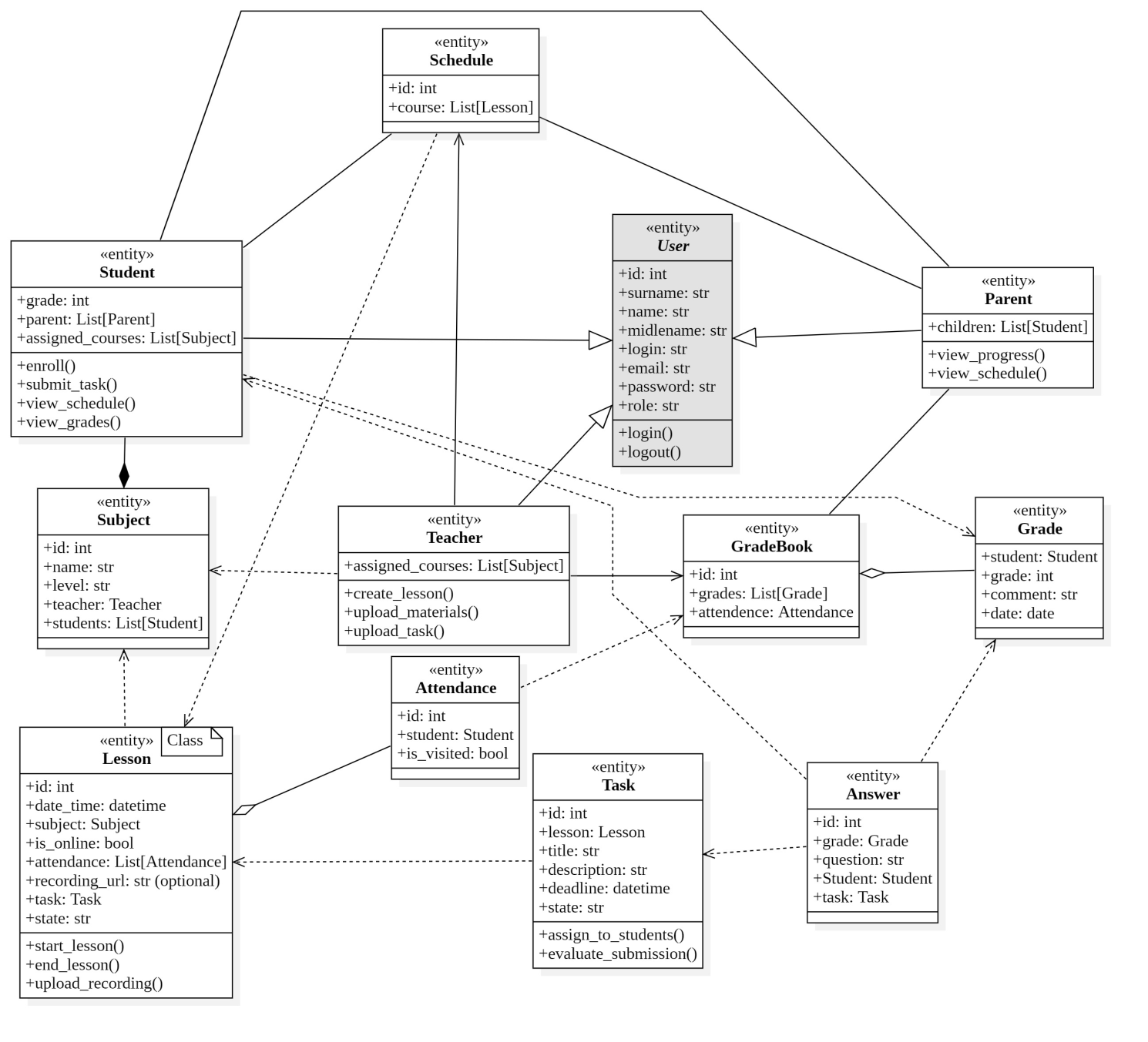


Рисунок 12. Диаграмма классов ИС

# Модель данных

### Логический уровень модели данных

Информация, используемая в ИС, хранится в базе данных. Логический уровень модель данных представлен на рисунке 13.

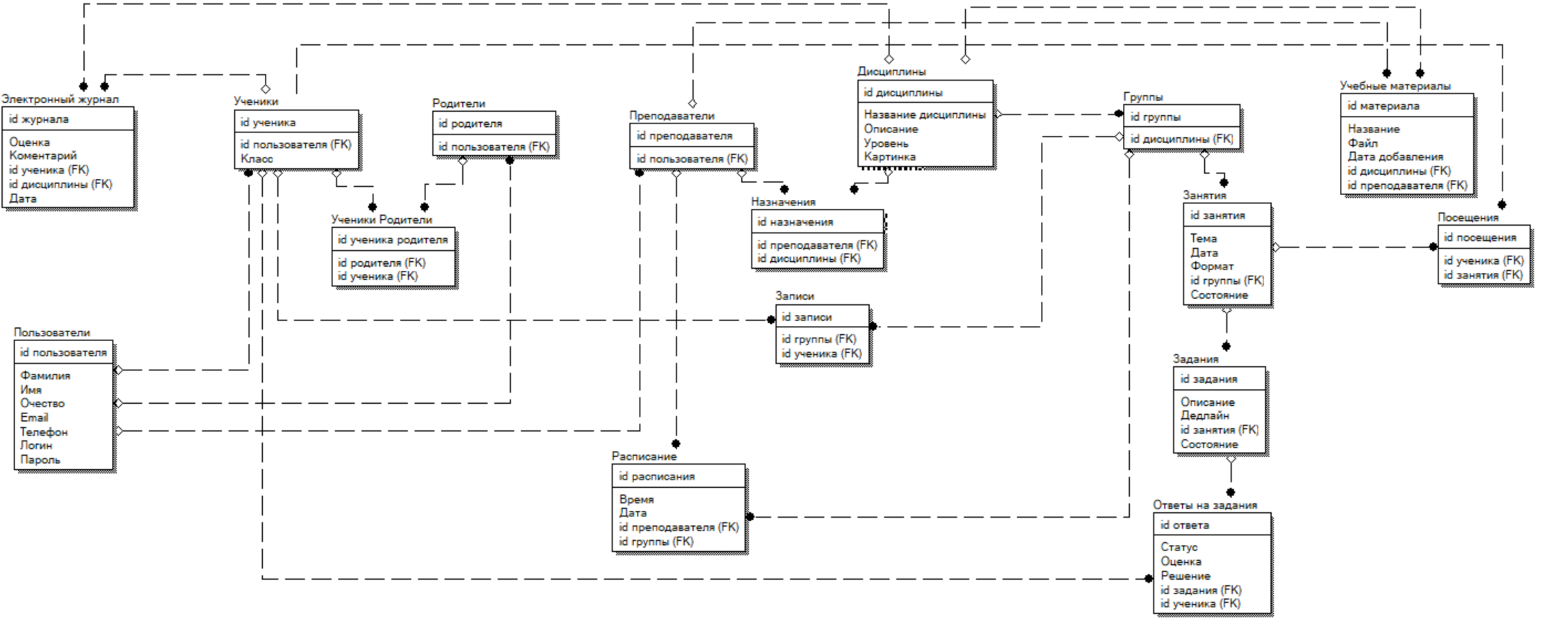


Рисунок 13. Логический уровень модели данных

### Физический уровень модели данных

Физический уровень представлен на рисунке 14. На нём изображены таблицы *SQLite*. Взаимодействие с базой данных происходит через *Django ORM*.

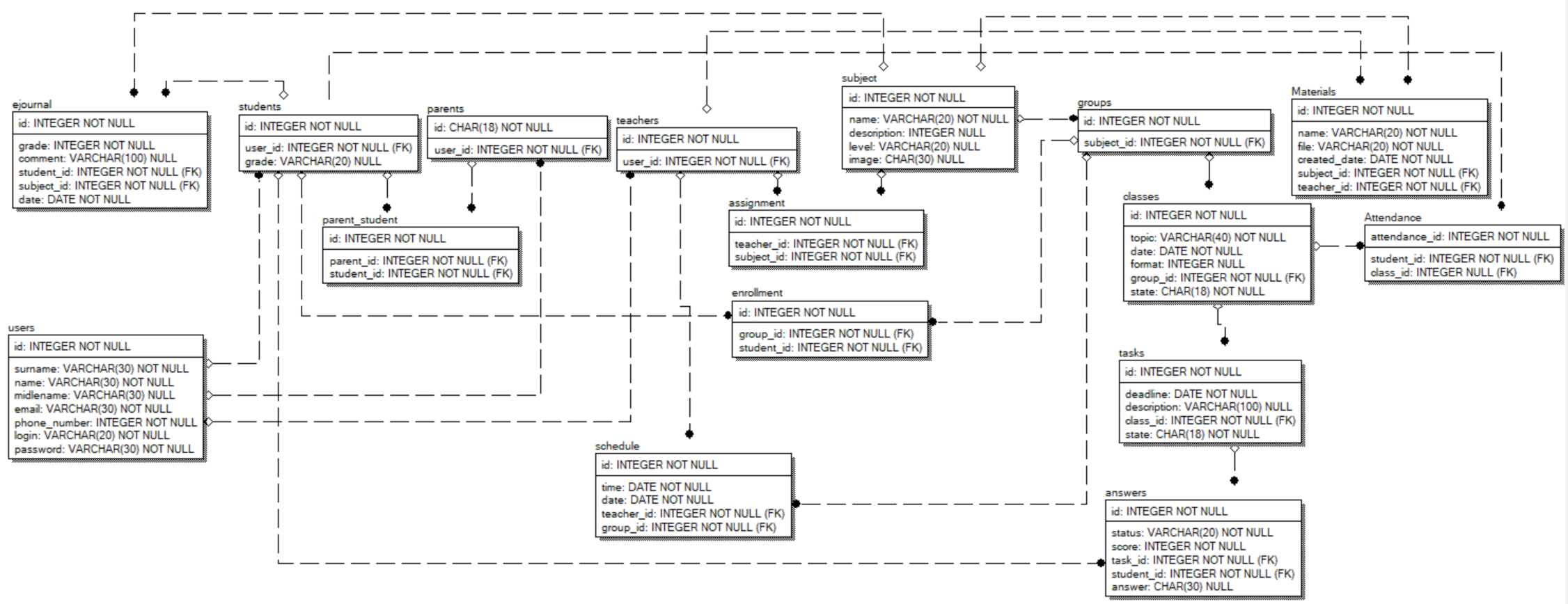


Рисунок 14. Физический уровень модели данных

# Интерфейс информационной системы

Форма просмотра оценок ученика (см. рис. 15) отображает список оценок, полученных пользователем в различных дисциплинах. Оценки отсортированы по предметам и датам.



Рисунок 15 .Форма просмотра оценок ученика

Форма просмотра онлайн занятия (см. рис. 16) отображает интерфейс проведения онлайн урока. В центральной части находится виртуальная доска, на которой преподаватель и ученики могут совместно писать и рисовать. Справа отображается список учеников, присоединившихся к занятию. В нижней части расположены кнопки управления.

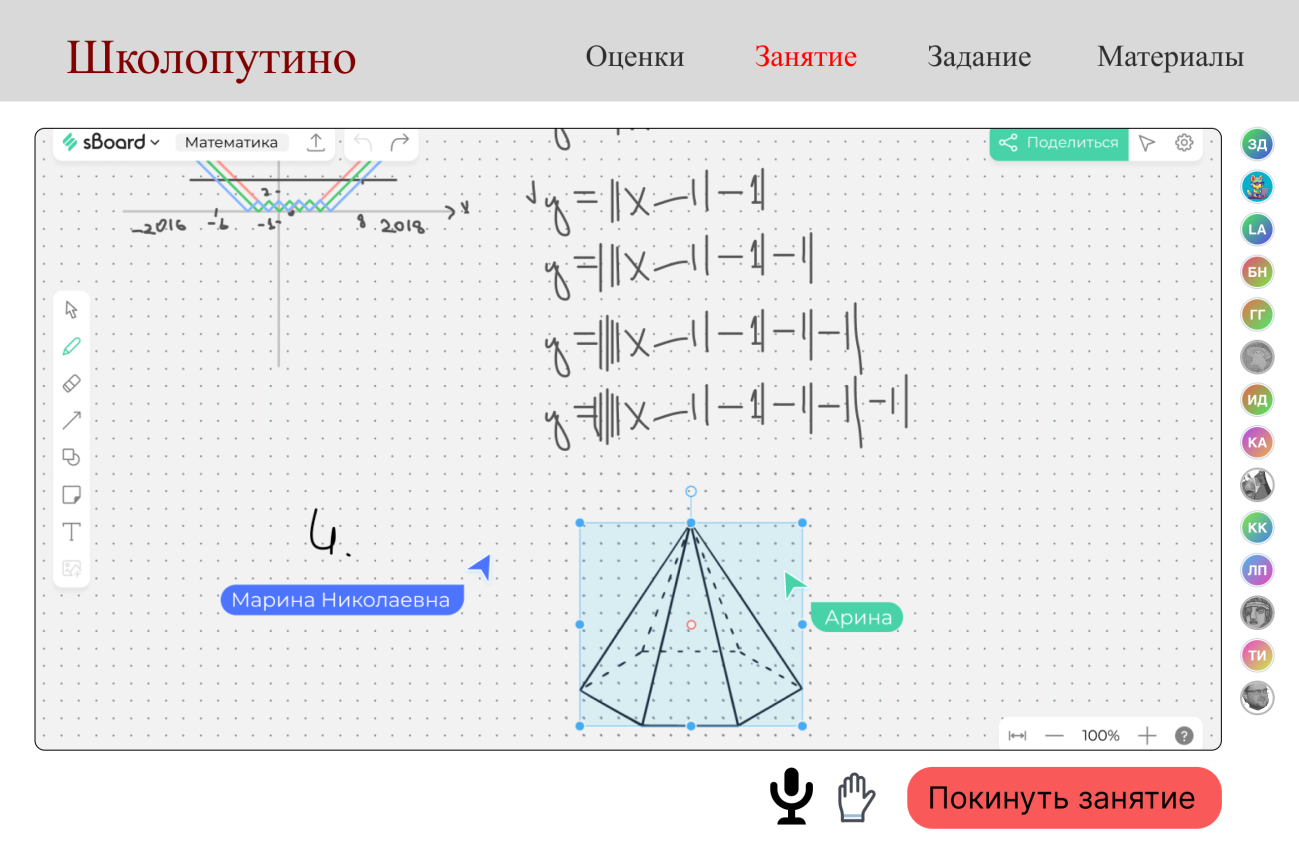


Рисунок 16. Форма просмотра онлайн занятия

Форма отправки задания (см. рис. 17) предназначена для прикрепления и отправки задания преподавателю. Форма содержит: название задания, условие задания, поле ввода текста решения и поле прикрепления файла.



Рисунок 17.Форма отправки задания

Форма просмотра учебных материалов (см. рис. 18) предназначена для доступа учащихся к дополнительным материалам. Форма содержит список предметов, на которые записан ученик.

.

Рисунок 18. Форма просмотра учебных материалов

Форма загрузки материалов преподавателем (см. рис. 19) предназначена для добавления методических материалов в систему. Форма содержит список добавленных материалов.



Рисунок 19. Форма загрузки материалов преподавателем

# Программная реализация

## Серверная часть

Серверная часть реализована на языке программирования *python* и фреймворком *Django* [3].

### *Django*

Проект был логически разделён на приложения (*Django Apps*):

* 1. *Users* – взаимодействие с пользователями системы;
  2. *Gradebook* – взаимодействие с успеваемостью;
  3. *Tasks* – взаимодействие с заданиями и ответами;
  4. *Schedule* – взаимодействие с расписанием, группами и дисциплинами;
  5. *Materials* – взаимодействие с учебными материалами;
  6. *Classes* – логика проведения онлайн занятий.

Были созданы модели базы данных через *Django ORM* (см. рис. 20).

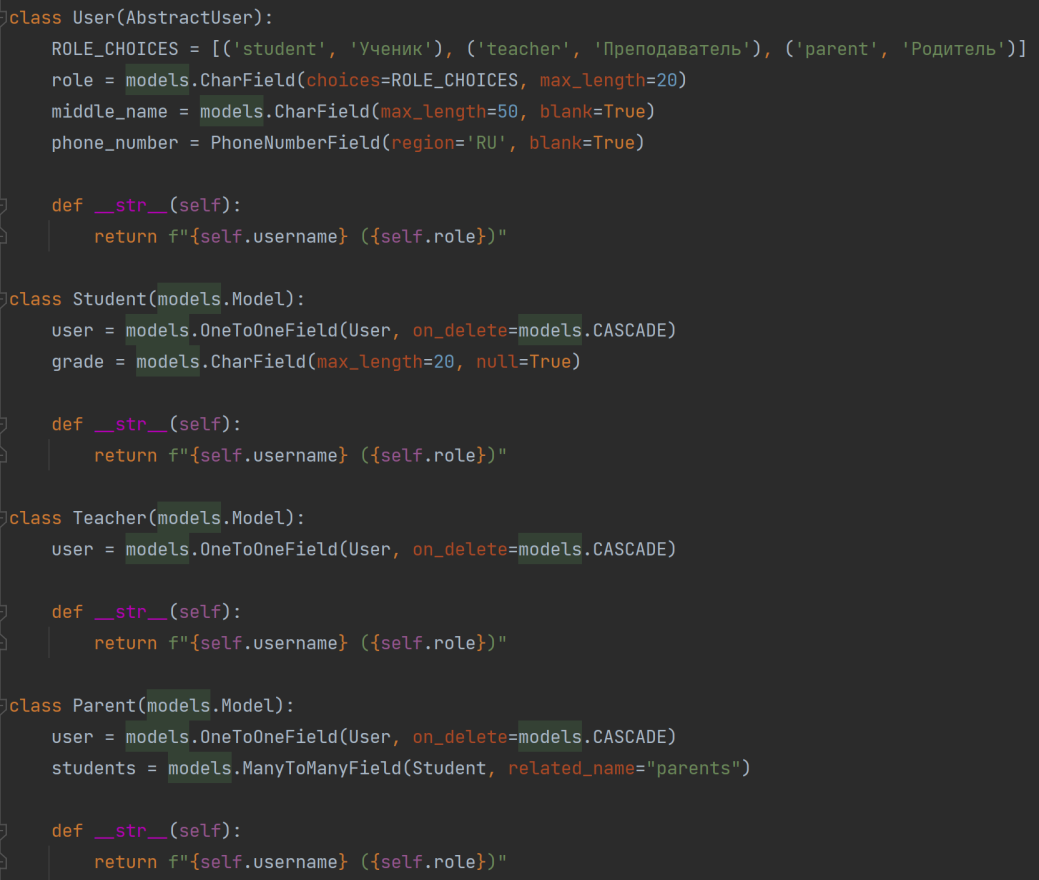


Рисунок 20. Классы модели пользователей в *Django ORM*

В приложениях, которые были реализованы, прописаны *View* классы и функции (см. рис. 21), которые управляют поведением запросов. Для корректной работы *API* запросов были написаны сериализаторы (*serializers*) для всех моделей БД.

***View* (представление)** — это компонент серверной части веб-приложения, который обрабатывает *HTTP*-запросы и возвращает *HTTP*-ответы. *View* выполняет логику приложения: получает данные из модели (или других источников), обрабатывает их и передаёт в шаблон или сериализует для *API*.



Рисунок 21. *View* классы для описания запросов в *Users*

### *API*

Взаимодействие между серверной и клиентской частью происходит через *API* (см. рис. 22), которое реализовано через *Django REST Framework* [4].



Рисунок 22. *API* ссылки

### *Swagger*

Для тестирования и документирования *API* была использована библиотека *Swagger* [5], которая автоматически отображает все существующие запросы, а также создаёт возможные параметры (*json*) для их выполнения (см. рис. 23).

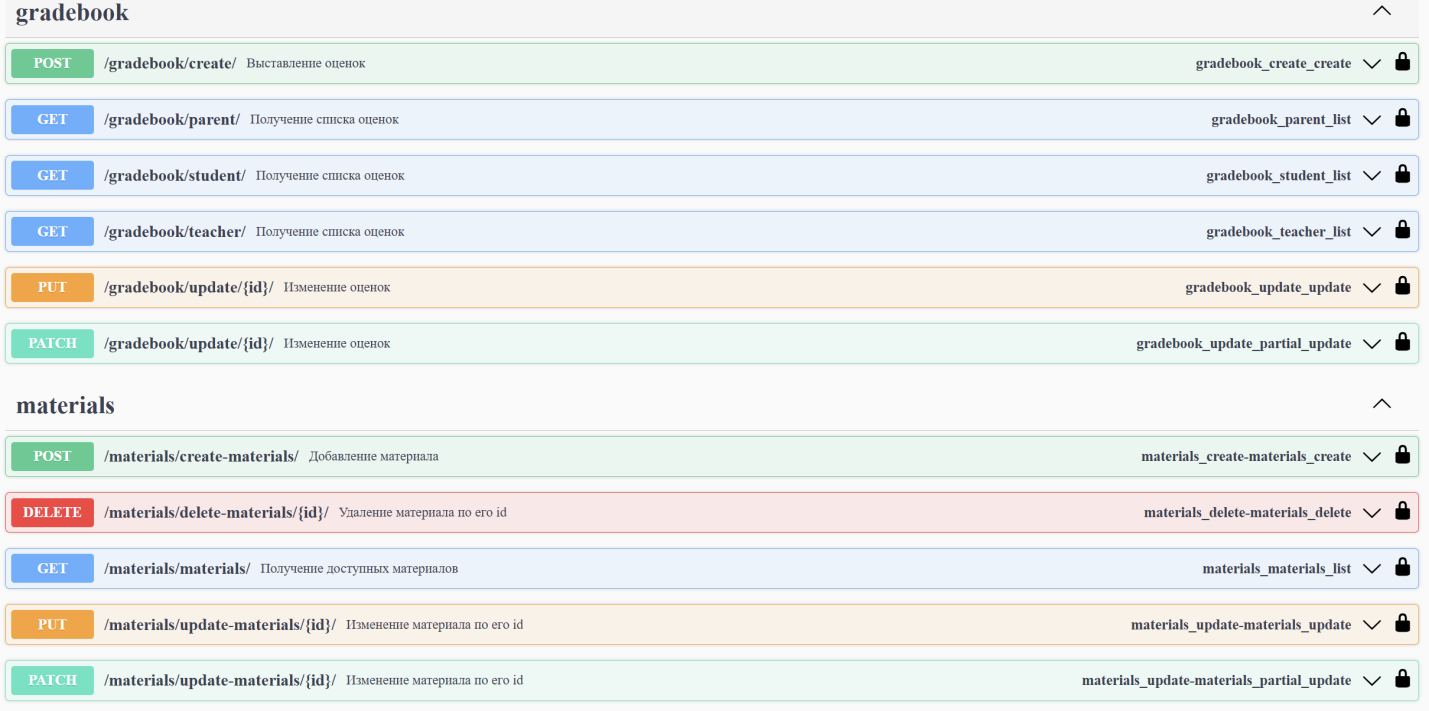


Рисунок 23. Интерфейс *Swagger*

## Клиентская часть

Клиентская часть реализована на языке *Dart*, используя фреймворк для разработки кроссплатформенных приложений *Flutter* [6].

Пользовательский интерфейс состоит из трёх основных разделов:

1. Страница «Журнал успеваемости»;
2. Страница «Методические материалы»;
3. Страница «Расписание занятий».

В зависимости от роли пользователя (ученик, преподаватель), функциональность адаптируется.

### Страница «Журнал успеваемости»

Изображение выглядит как текст, диаграмма, График, число

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 24. Интерфейс страницы «Журнал успеваемости»

Данный раздел предоставляет доступ к системе оценки успеваемости учащихся.

Преподаватель может:

* Просматривать список учащихся, привязанных к его учебным группам и предметам;
* Выбирать один из своих предметов и соответствующую группу
* Добавлять и удалять оценки по конкретным предметам и датам

Ученик может:

* Просматривать оценки по предметам.

При нажатии на иконку «+» в ячейке открывается модальное окно выбора оценки (см. рис).

Изображение выглядит как текст, снимок экрана, линия

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 25. Модальное окно выбора оценки

### Страница «Методические материалы»

Изображение выглядит как текст, снимок экрана

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 26. Интерфейс страницы «Методические материалы»

Этот раздел предназначен для обмена учебными материалами между преподавателями и учениками.

Преподаватель может:

* Загружать новые учебные материалы;
* Удалять загруженные материалы.

Ученик может:

* Просматривать загруженные материалы.

При нажатии на иконку «+» в правом верхнем углу открывается модальное окно добавление материала (см. рис. 27).

Изображение выглядит как текст, снимок экрана, программное обеспечение, Мультимедийное программное обеспечение

Контент, сгенерированный ИИ, может содержать ошибки.

Рисунок 27. Модальное окно добавления материала

### Страница «Расписание занятий»

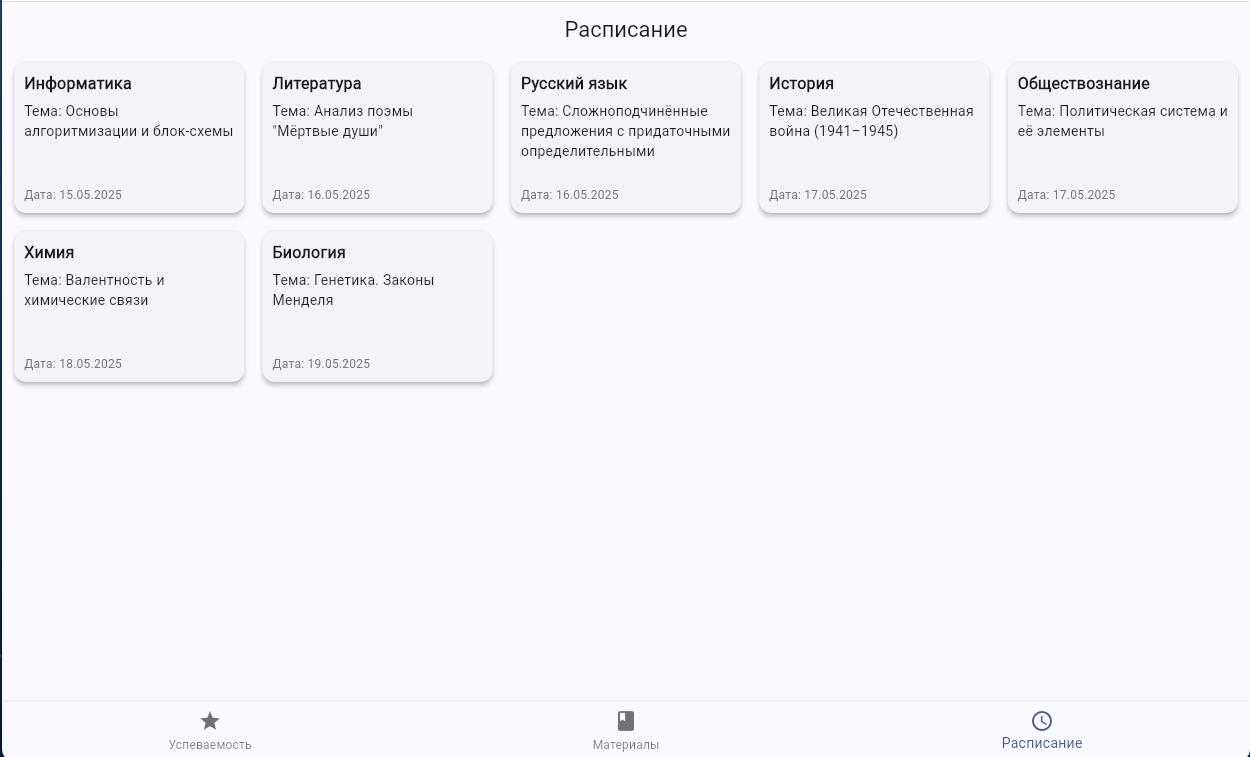


Рисунок 28. Интерфейс страницы «Расписание занятий»

Этот модуль содержит информацию о времени проведения уроков, преподавателях и предметах. На текущем этапе реализована версия расписания только для просмотра. Все категории пользователей — преподаватели, учащиеся и родители — имеют доступ только к просмотру актуального расписания занятий.

# Заключение

В ходе выполнения курсовой работы была спроектирована и частично реализована информационная система управления обучением для онлайн-школы подготовки к ОГЭ и ЕГЭ «Школопутино». Основанием для её разработки стали выявленные проблемы в организации учебного процесса, описанные в разделе «Описание предметной области». В рамках проектирования были разработаны ключевые диаграммы и модели: диаграммы вариантов использования, диаграммы последовательности, деятельности и состояний, а также диаграмма классов, логическая и физическая модели базы данных.

Для моделирования и анализа предметной области использовались программные инструменты *StarUML* и *Erwin Data Modeler*, позволяющие формализовать структуру и логику работы системы. Визуальный прототип пользовательского интерфейса был создан с помощью сервиса *Figma*.

Для реализации программной части информационной системы были использованы современные и широко применяемые технологии: фреймворк *Django* на языке *python* — для разработки серверной части, включая реализацию логики взаимодействия с базой данных и *API*, а также *Flutter* — кроссплатформенный фреймворк для создания клиентского интерфейса. Это позволило обеспечить удобную, масштабируемую архитектуру и потенциальную доступность системы как на мобильных устройствах, так и в браузере. В результате была получена частично реализованная информационная система.

Разработанная система призвана упростить выполнение повседневных задач преподавателей, таких как организация занятий, выдача и проверка заданий, отслеживание посещаемости и оценивание. Внедрение данной системы позволит автоматизировать рутинные процессы, повысить прозрачность и управляемость образовательной деятельности, а также создать единую цифровую среду для взаимодействия между преподавателями, учениками и родителями.

# Список использованной литературы

1. **Савватеева Т. П., Миловидова А. А., Кудрявцева Д. В.** *Технологии проектирования информационных систем. Учебное пособие.* Дубна: Университет «Дубна» : б.н., 2016. стр. 120.

2. **Заботина Н. Н.** *Проектирование информационных систем: учебное пособие.* Москва: ИНФРА-М : б.н., 2022. стр. 331.

3. *Django documentation.* *Django.* [В Интернете] https://www.djangoproject.com.

4. *Django REST framework*. [В Интернете] https://www.django-rest-framework.org/#.

5. *Yet another Swagger generator*. *drf-yasg.* [В Интернете] https://drf-yasg.readthedocs.io/en/stable/readme.html.

6. *Flutter*. Официальная документация. [В Интернете] https://docs.flutter.dev.

# Приложение

Рисунок 29. Отчёт проверки работы на плагиат