**Министерство образования и науки Российской Федерации**

**федеральное государственное автономное образовательное учреждение высшего образования**

**“САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ ИССЛЕДОВАТЕЛЬСКИЙ**

**УНИВЕРСИТЕТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ,**

**МЕХАНИКИ И ОПТИКИ”**

Факультет ИКТ

Образовательная программа *09.03.02 Информационные системы и технологии*

Направление подготовки (специальность) Мобильные сетевые технологии

О Т Ч Е Т

о курсовой работе

Тема задания: РЕАЛИЗАЦИЯ WEB-СЕРВИСОВ СРЕДСТВАМИ Django REST framework,Vue.js, Muse-UI

Обучающийся Пахмурин Максим Андреевич, К3340

Преподаватель: Говоров Антон Игоревич

Курсовая сдана с оценкой \_\_\_\_

Подпись преподавателя:

\_\_\_\_\_\_\_\_\_\_\_\_\_\_

(подпись)

Дата \_\_\_\_

Санкт-Петербург

2020

#### СОДЕРЖАНИЕ

**ВВЕДЕНИЕ** 3

**1. ОПИСАНИЕ ПРЕДМЕТНОЙ ОБЛАСТИ** 4

**2. ОПИСАНИЕ СЕРВЕРНОЙ ЧАСТИ**  5

**3. ОПИСАНИЕ КЛИЕНТСКОЙ ЧАСТИ** 8

**ЗАКЛЮЧЕНИЕ** 14

**СПИСОК ЛИТЕРАТУРЫ** 15

# ВВЕДЕНИЕ

Цель: овладеть практическими навыками и умениями реализации web-сервисов

средствами Django REST framework, Vue.js, Muse-UI.

Оборудование: компьютерный класс.

Программное обеспечение: Python 3.6, Django REST framework, Vue.js, Muse-UI, PostgreSQL.

Практическое задание:

Реализовать сайт используя вышеуказанные технологии, в соответствии с практическим заданием.

Задание: создать программную систему, предназначенную для учебной части колледжа.

# Описание предметной области

Для реализации сайта были использованы следующие средства разработки: Django, Django rest framework, Vue.js, Muse UI, Postgresql.

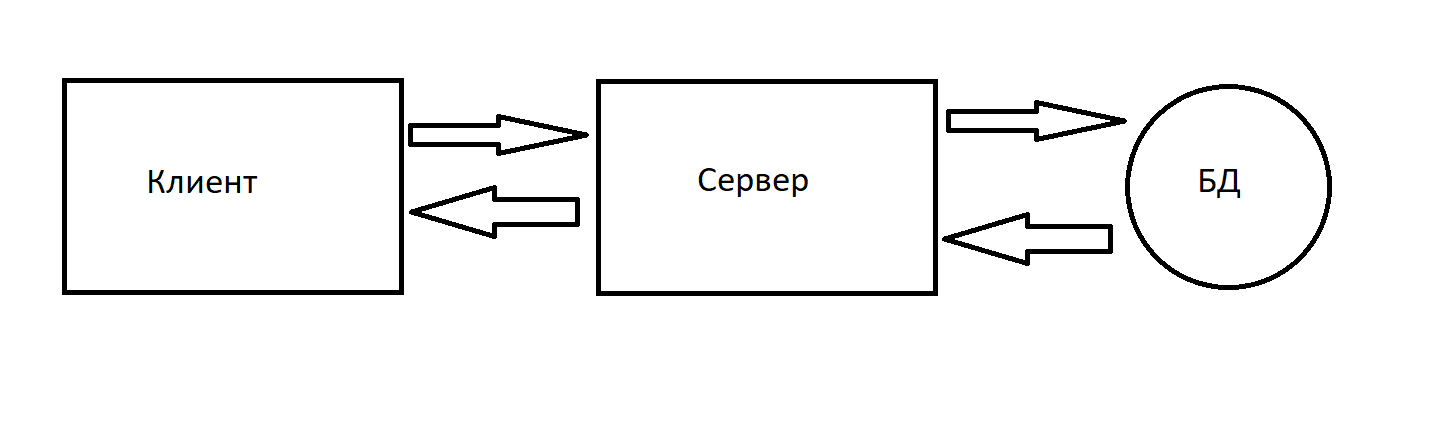
Django – свободный фреймворк для веб-приложений на языке Python, использующий шаблон проектирования MVC.

Django rest framework – это библиотека, которая работает со стандартными моделями Django для создания гибкого и мощного API для проекта.

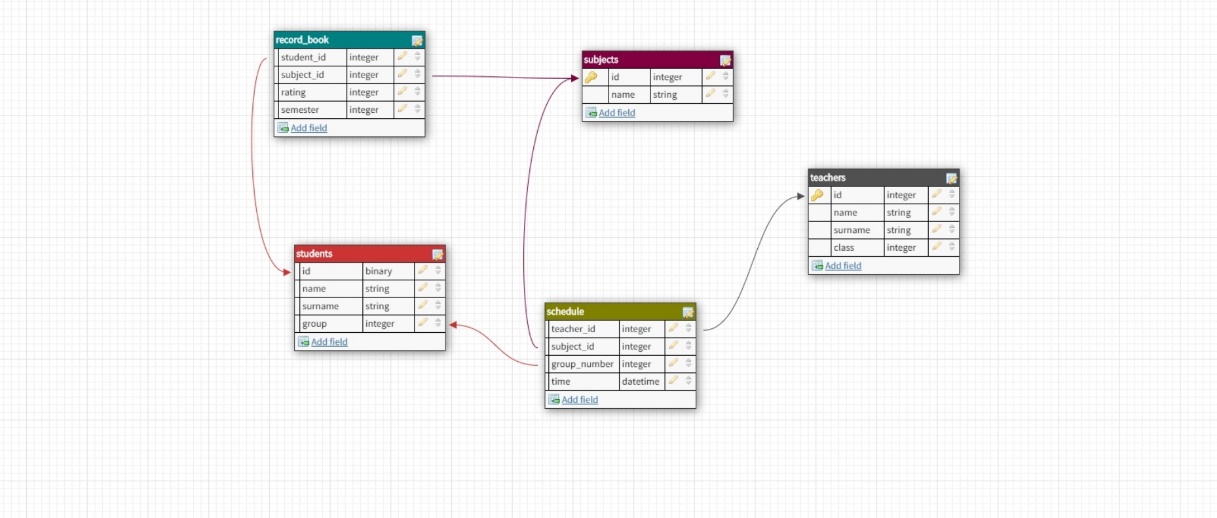
Vue.js – JavaScript-фреймворк с открытым исходным кодом для создания пользовательских интерфейсов. Легко интегрируется в проекты с использованием других JavaScript-библиотек. Может функционировать как веб-фреймворк для разработки одностраничных приложений в реактивном стиле.

Muse-UI – основанный на Vue 2.0 компонент для разработки красивых пользовательских интерфейсов.

Использована клиент-серверная архитектура – сетевая архитектура, в которой сетевая нагрузка распределена между серверными приложениями (поставщиками услуг) и клиентскими приложениями (заказчиками услуг).



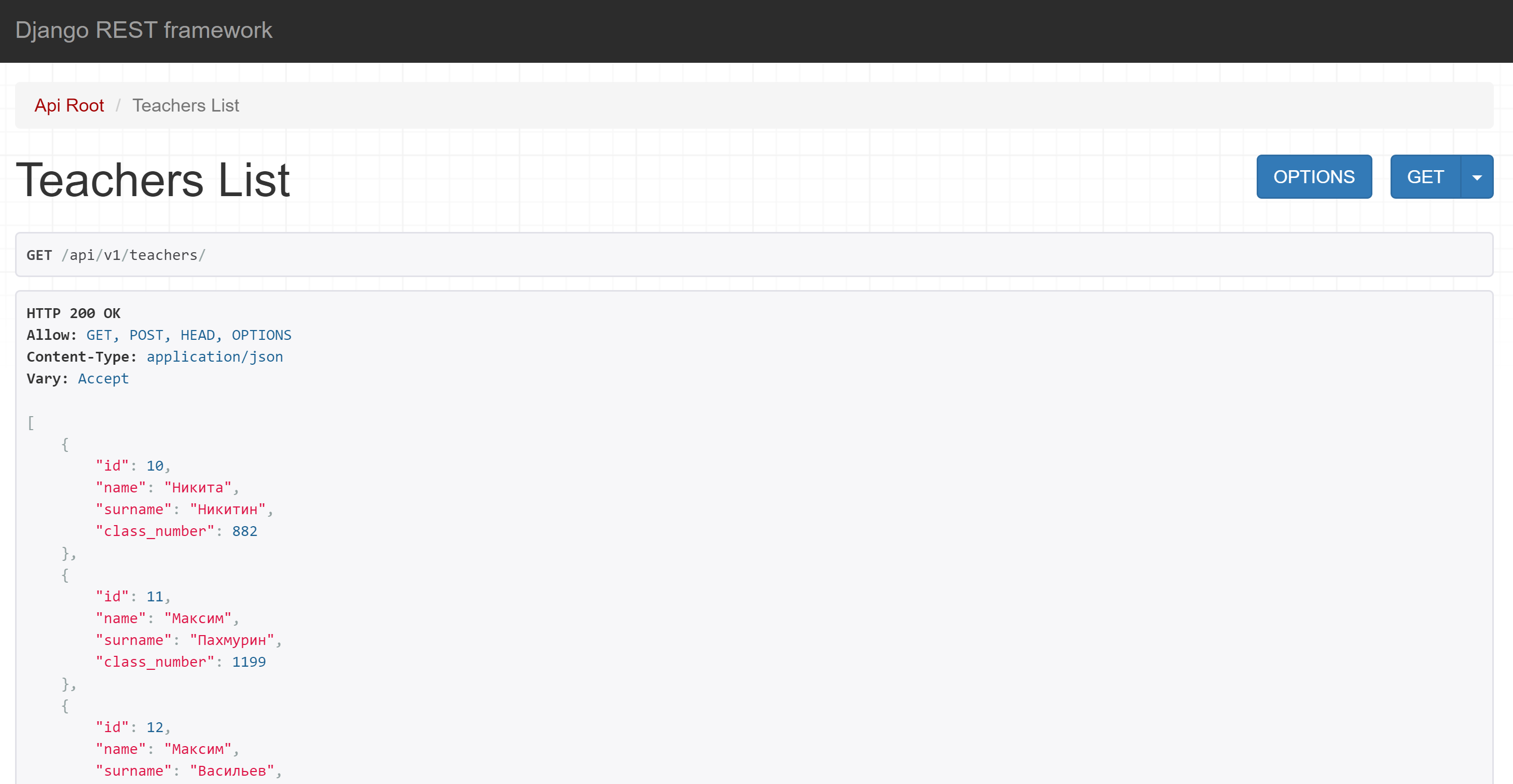
Модель базы данных:

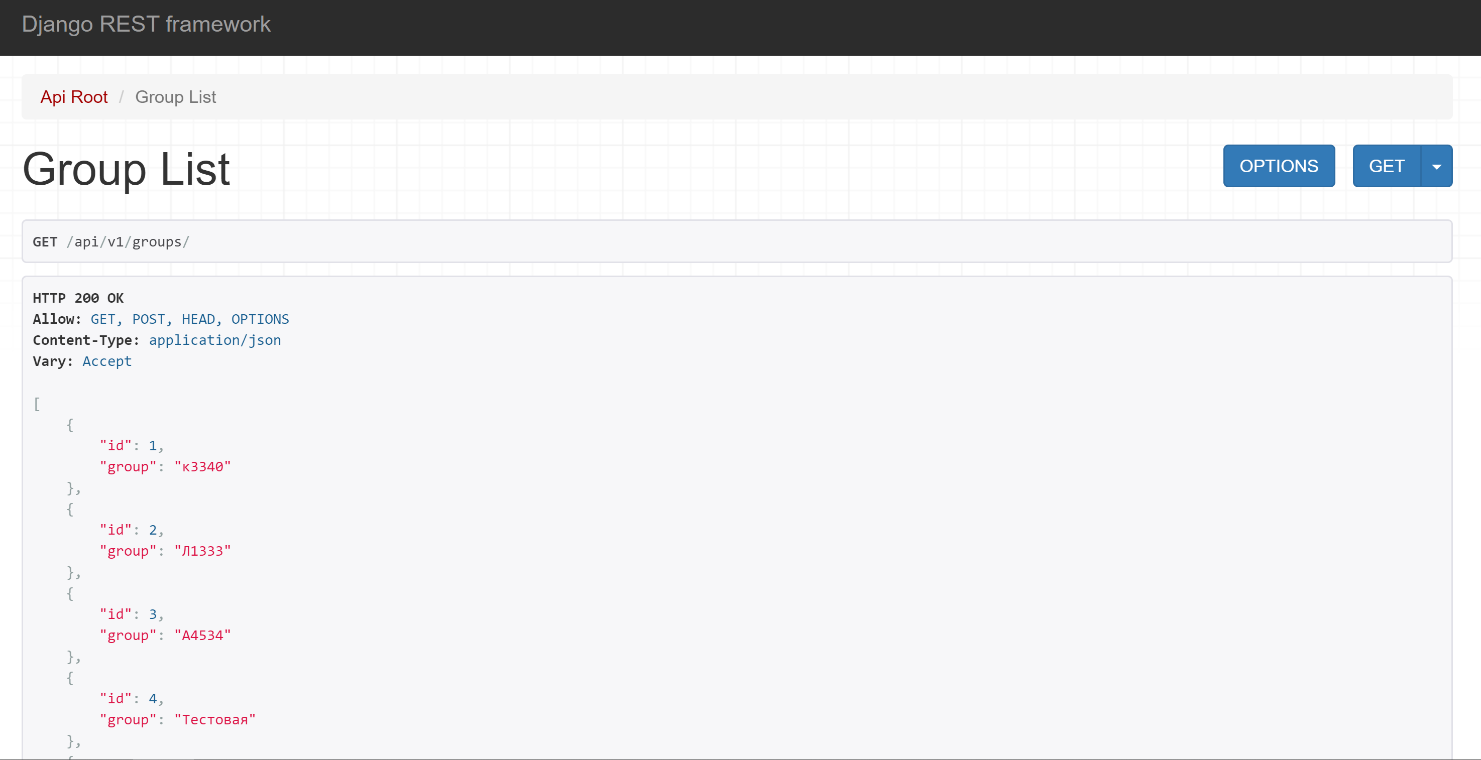


# Описание серверной части

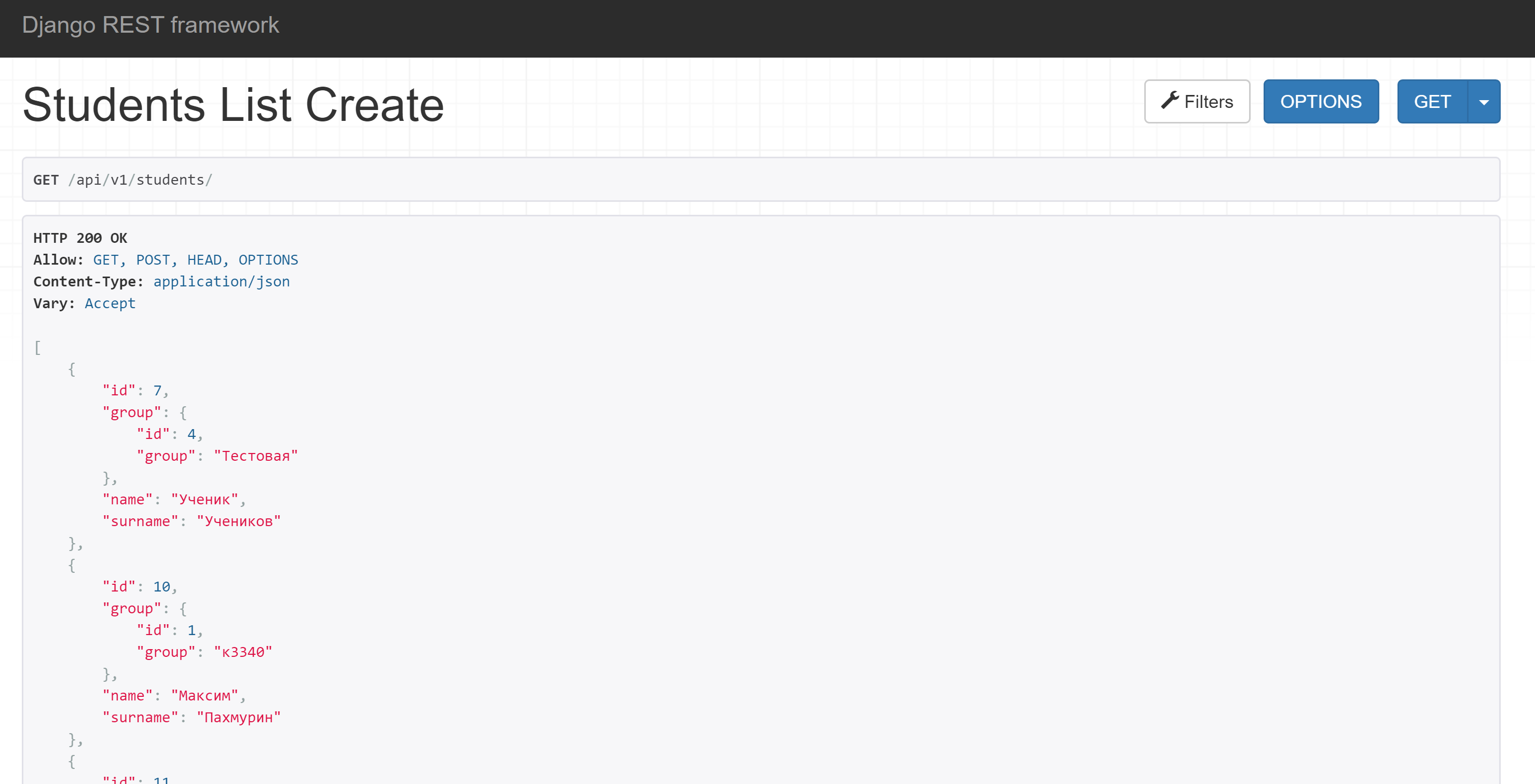
Rest (сокр. англ. Representational State Transfer, «передача состояния представления») — стиль построения архитектуры распределенного приложения. Данные в REST должны передаваться в виде небольшого количества стандартных форматов (например HTML, XML, JSON). Сетевой протокол, как и HTTP, должен поддерживать кэширование, не должен зависеть от сетевого слоя, не должен сохранять информацию о состоянии между парами «запрос-ответ». Утверждается, что такой подход обеспечивает масштабируемость системы и позволяет ей эволюционировать с новыми требованиями. Django REST framework — удобный инструмент для работы с rest основанный на идеологии фреймворка Django.

Json представление для интерфейса «Учителя»:

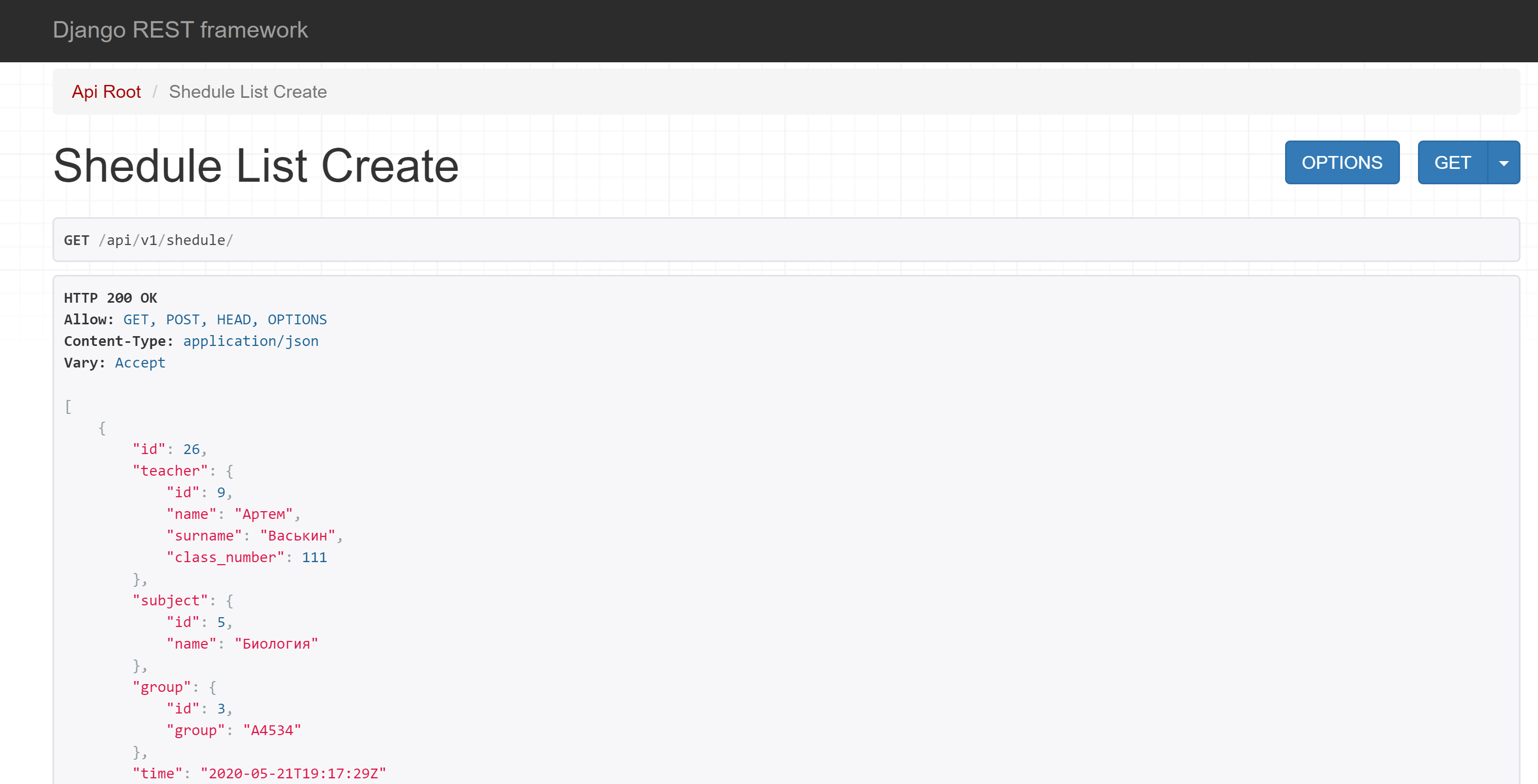


 Json представление для интерфейса «Группы»:

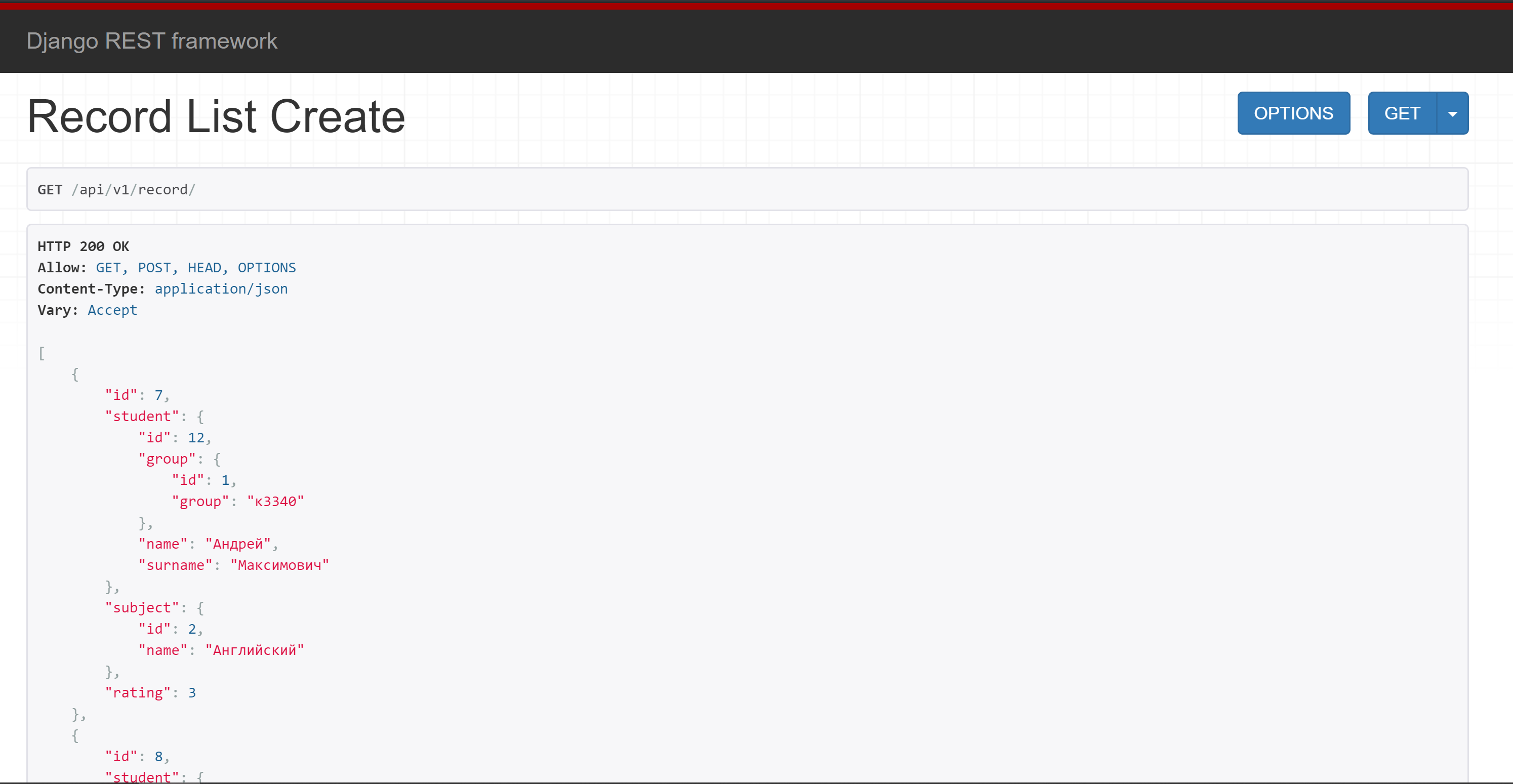
Json представление для интерфейса «Студенты»:



Json представление для интерфейса «Расписание»:

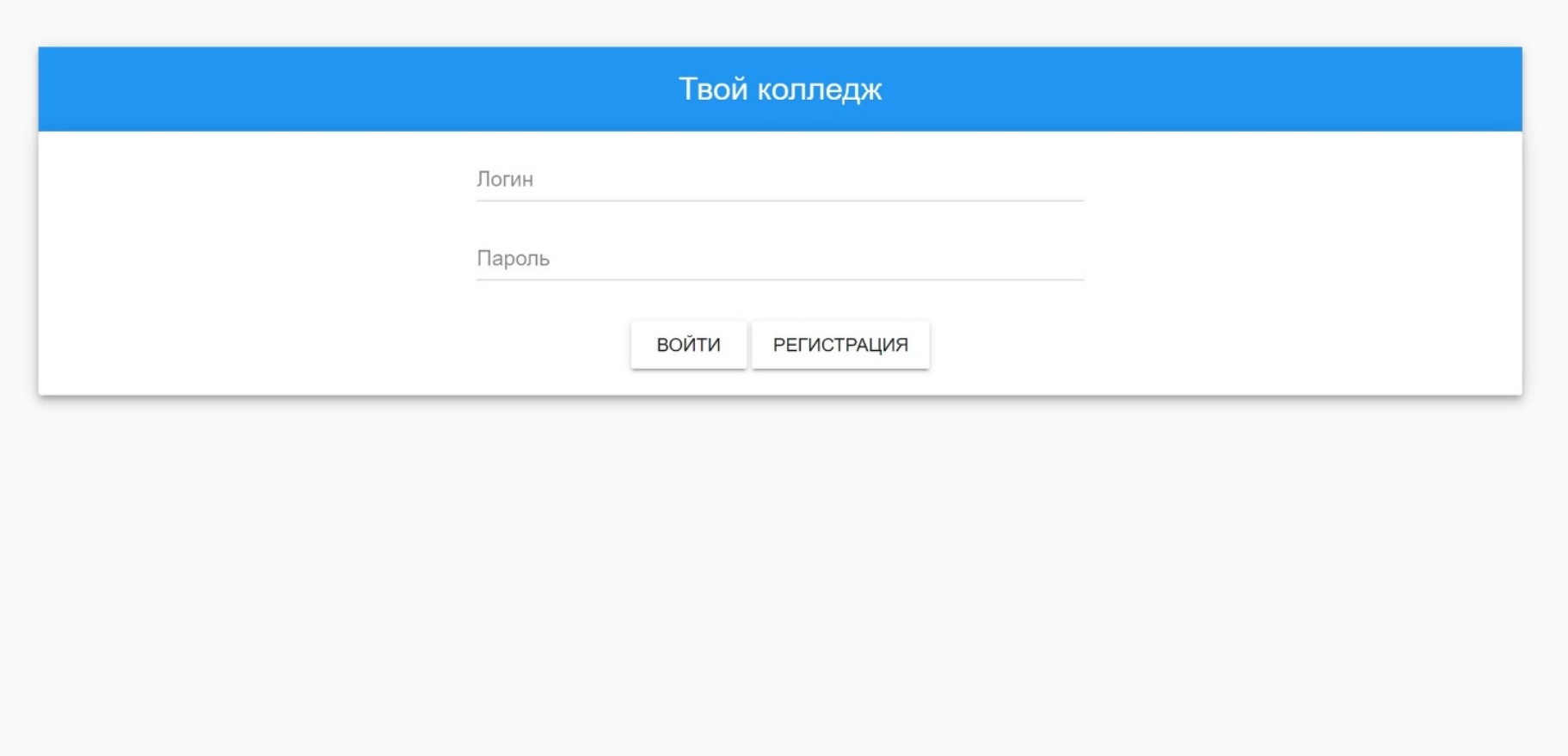


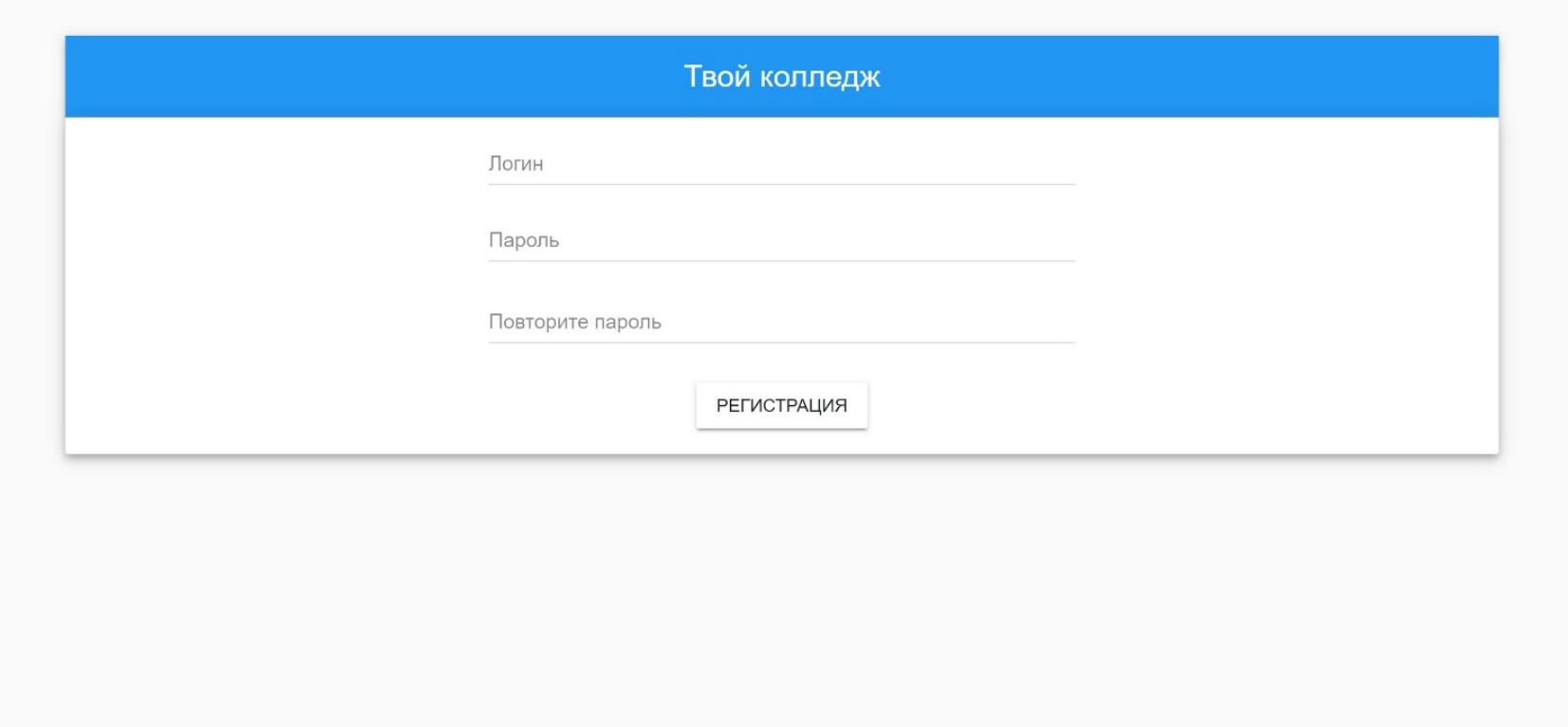
Json представление для интерфейса «Учебный журнал»:

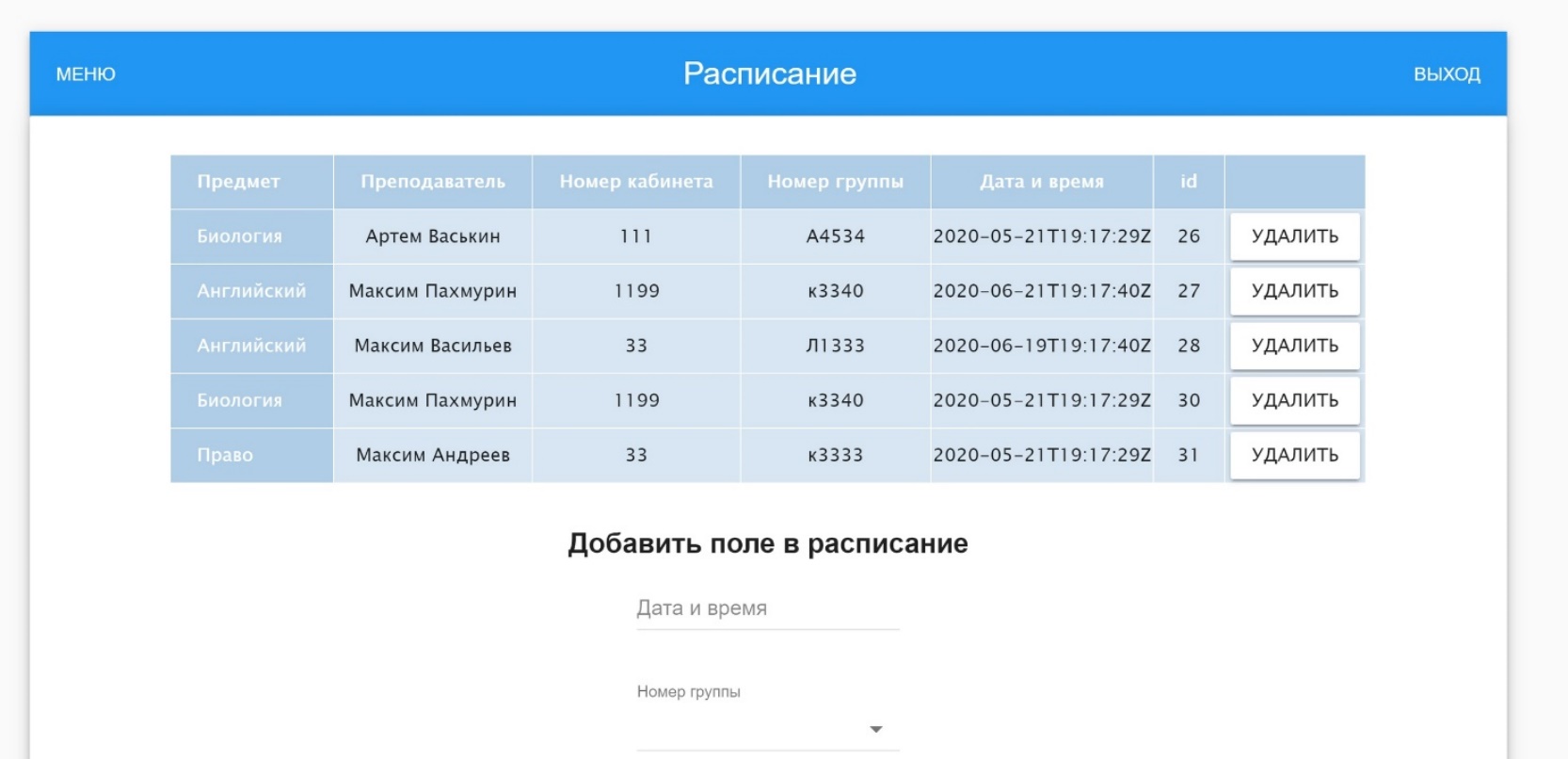


# Описание клиентской части

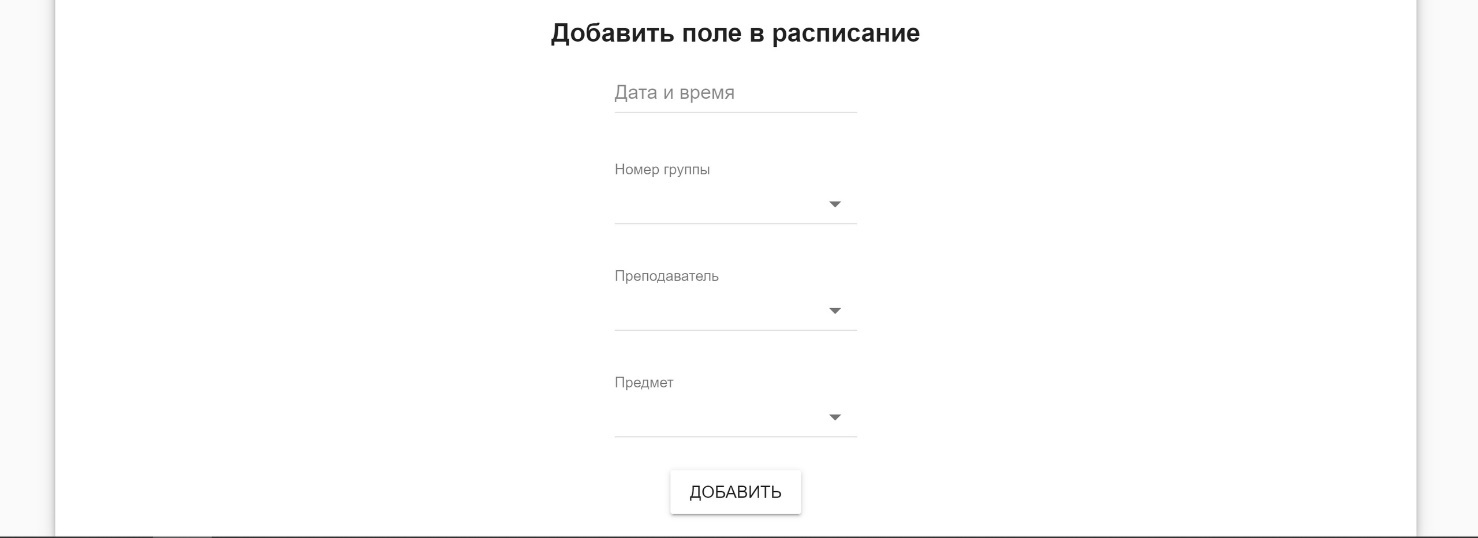
Vue (произносится /vjuː/, примерно как view) — это прогрессивный фреймворк для создания пользовательских интерфейсов. В отличие от фреймворков-монолитов, Vue создан пригодным для постепенного внедрения. Его ядро в первую очередь решает задачи уровня представления (view), что упрощает интеграцию с другими библиотеками и существующими проектами. С другой стороны, Vue полностью подходит и для создания сложных одностраничных приложений (SPA, Single-Page Applications), если использовать его совместно с современными инструментами и дополнительными библиотеками.

Форма входа:

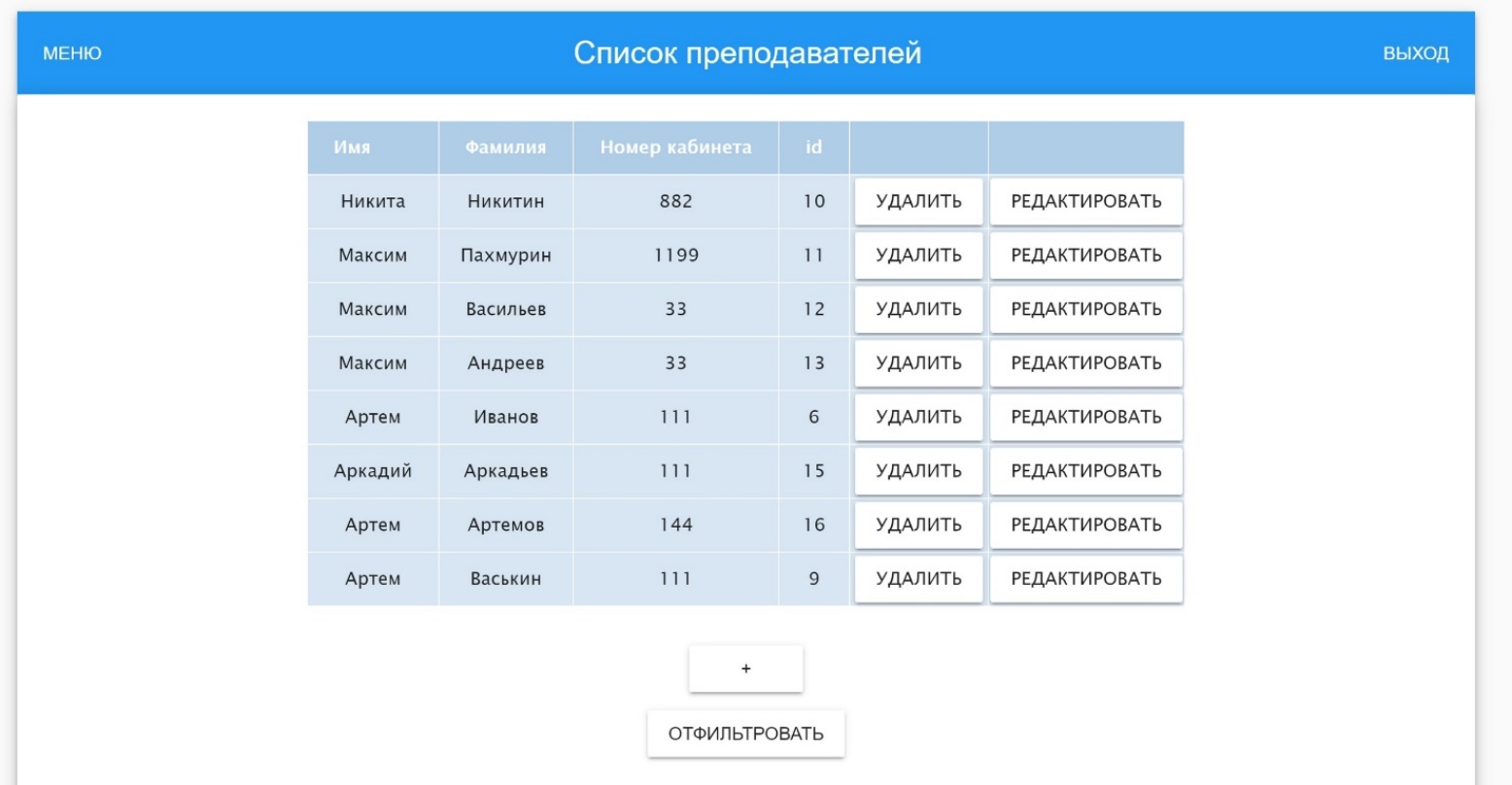
Форма регистрации:

Интерфейс «Расписание»:

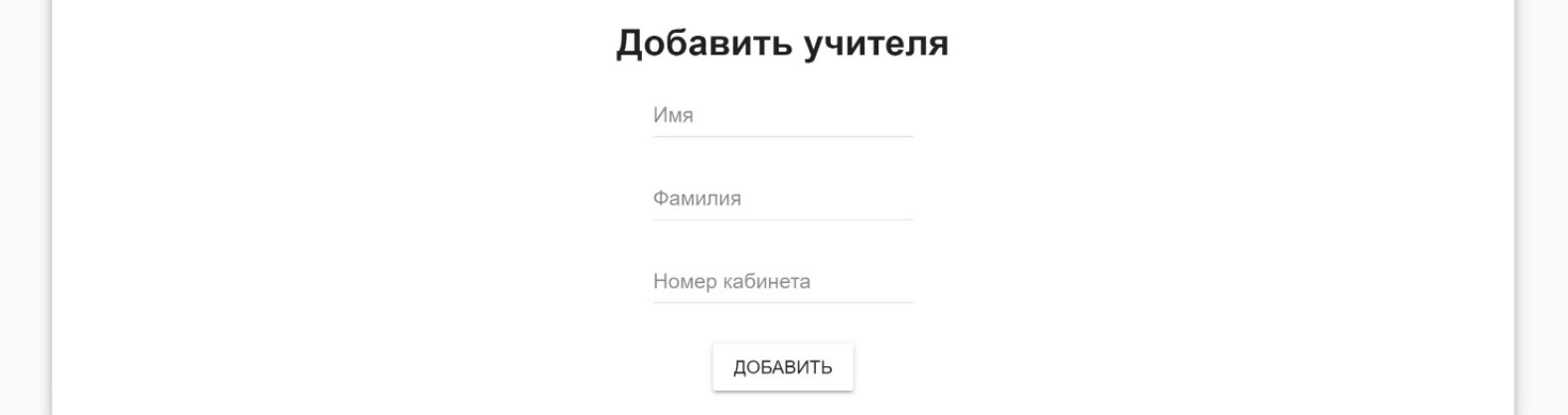
Интерфейс для добавление поля в расписание:



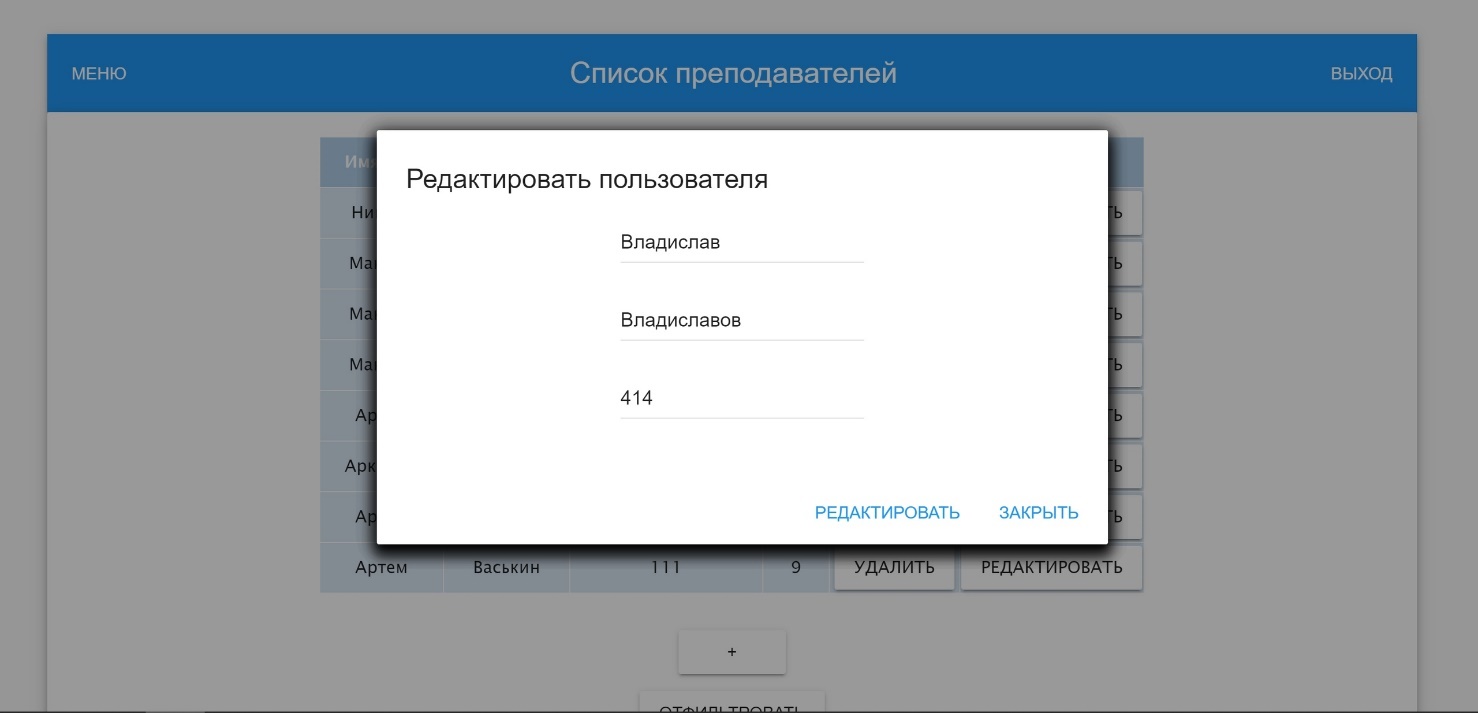
Интерфейс «Список преподавателей»:



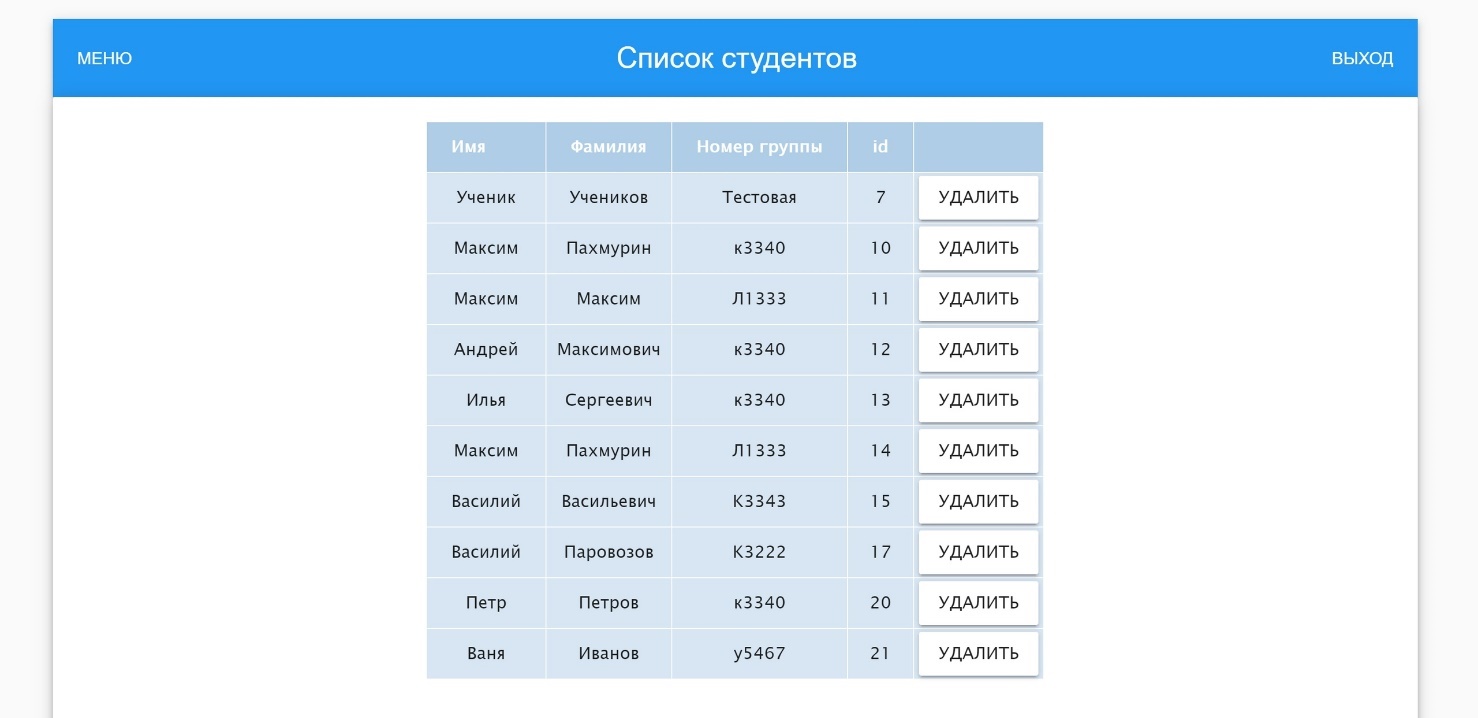
Интерфейс для добавление учителей:



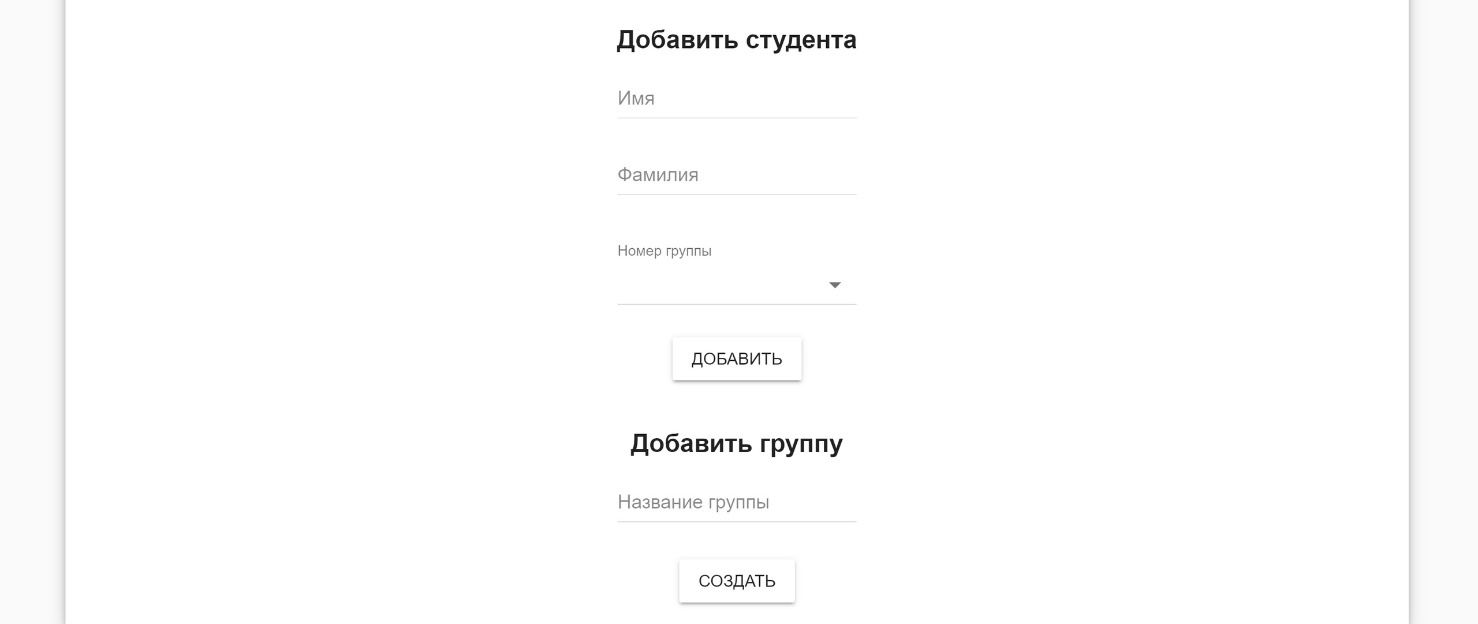
Интерфейс редактирования преподавателей:



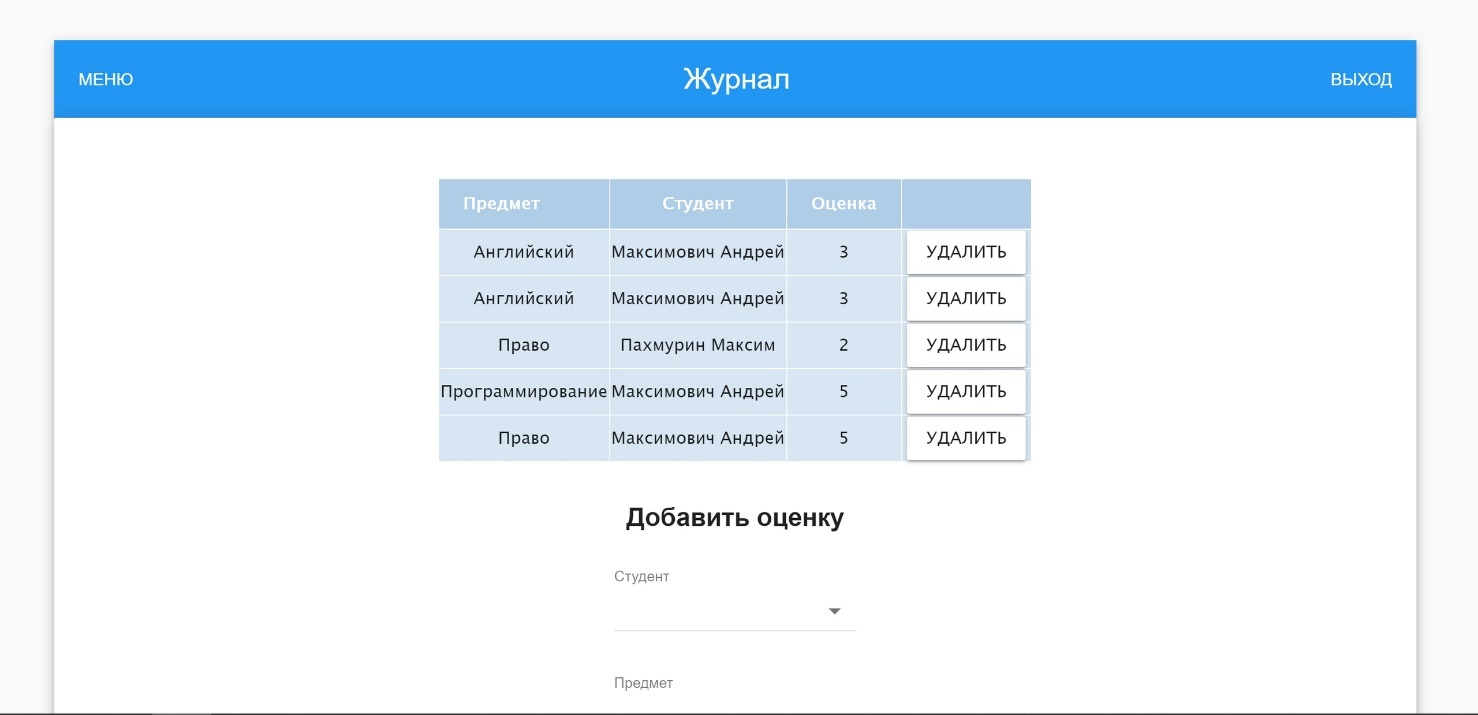
Интерфейс «список студентов»:



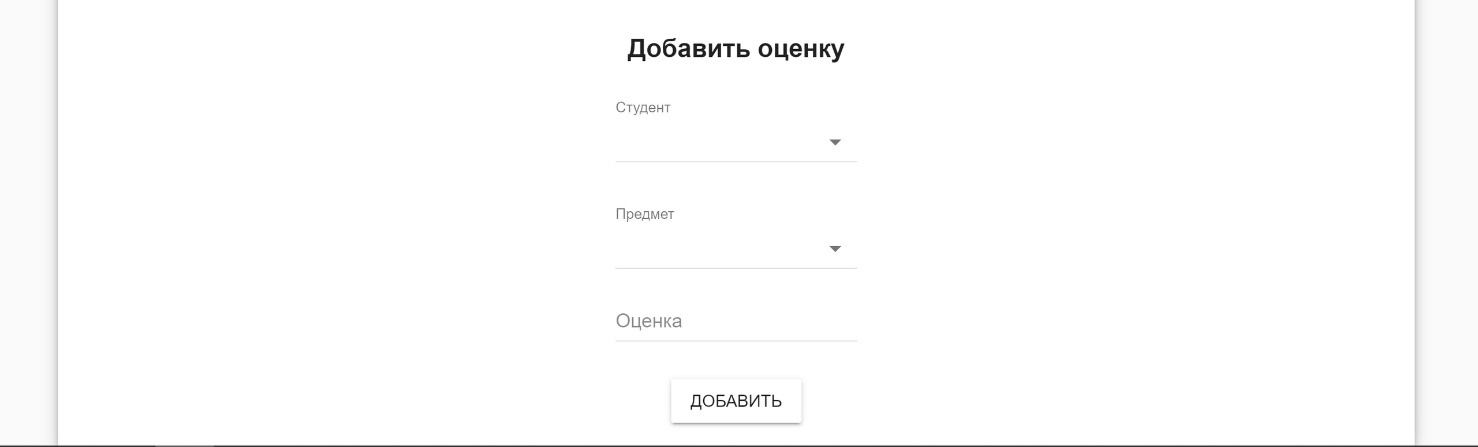
Интерфейс добавление студентов и групп:



Интерфейс «Журнал»:



Интерфейс для добавление оценки:



ЗАКЛЮЧЕНИЕ

Во время выполнения курсовой работы было изучено много нового и полезного, а также было приобретено несколько полезных навыков, а именно:

* Умение реализовывать клиент-серверные веб-сервисы.
* Умение правильно работать с документацией.
* Умение анализировать поставленную задачу и подбирать наиболее подходящие решения для неё (библиотеки, методы, и тд.)

# СПИСОК ЛИТЕРАТУРЫ

1. Документация Django: <https://django.fun/docs/django/ru/3.0/>;
2. Документация Vue.js <https://ru.vuejs.org/index.html>;
3. Документация Django rest framework: <https://www.django-rest-framework.org/>;