Команда Runtime Error

Хакатон 1.1

Состав команды

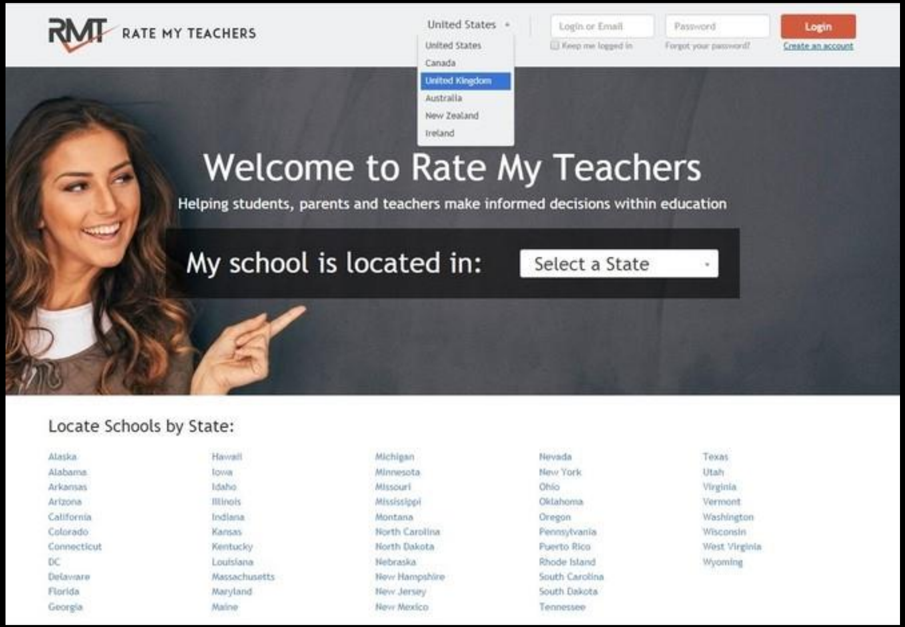
- Кулишов Егор (Frontend Developer)
- Инюцин Игнат (Team Lead)
- Михаил Леонтьев (Backend Developer)

Описание кейса

Составить web или мобильное приложение для автоматизированного составления рейтинга преподавателей(встроеная рейтинговая система, которая формируется путём голосования участников, с и без регистрации). В идеале - сайт или приложение, в котором есть страница с данными о преподавателях, их цитатами и др. доп. информацией, выстроен рейтинг популярности и особых навыков

Существующие решения

- **RateMyTeachers**



Welcome to Rate My Teachers

Helping students, parents and teachers make informed decisions within education

My school is located in:

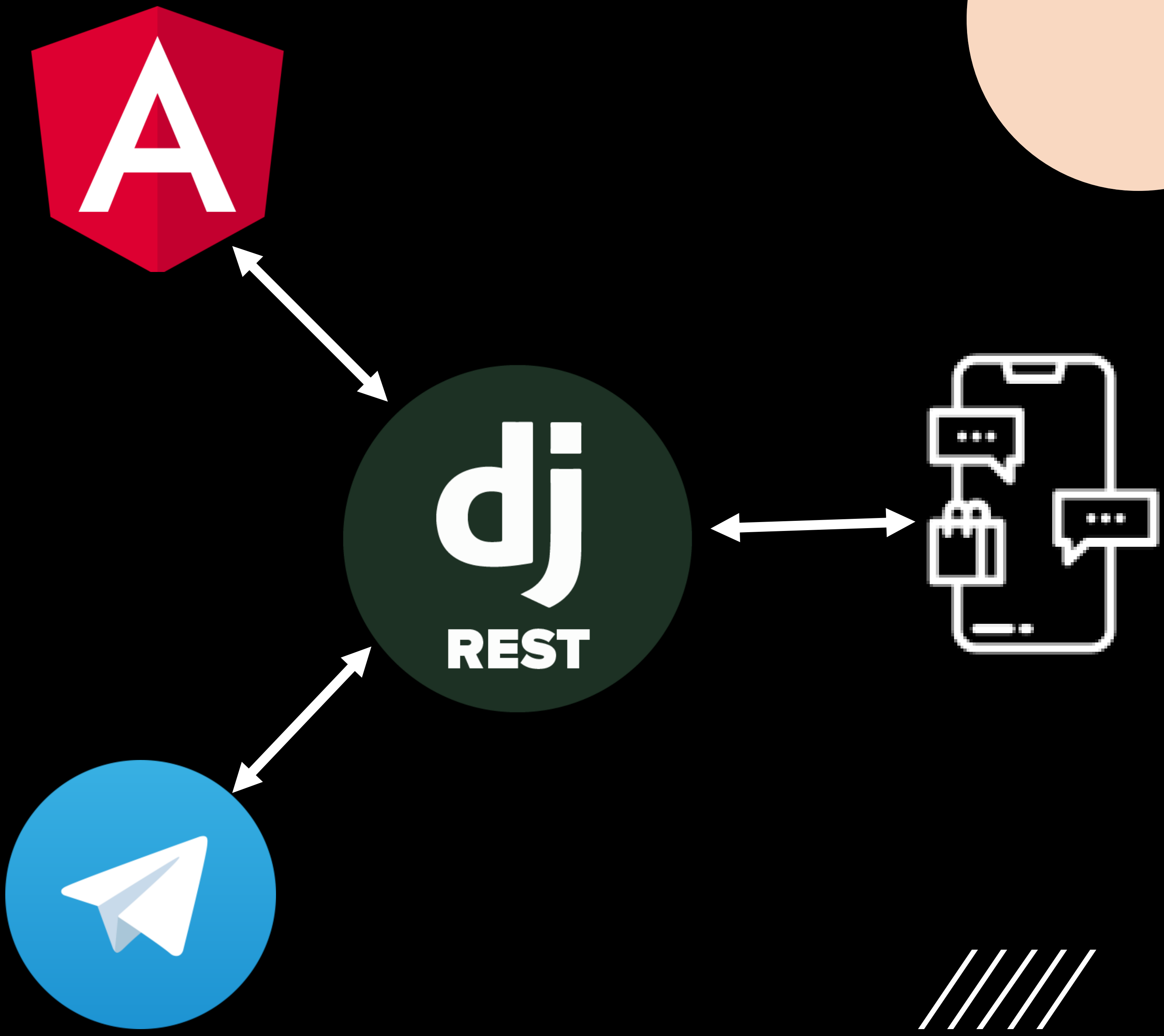
Locate Schools by State:

Alaska
Alabama
Arkansas
Arizona
California
Colorado
Connecticut
DC
Delaware
Florida
Georgia
Hawaii
Iowa
Idaho
Illinois
Indiana
Kansas
Kentucky
Louisiana
Massachusetts
Maryland
Maine
Michigan
Minnesota
Missouri
Mississippi
Montana
North Carolina
North Dakota
Nebraska
New Hampshire
New Jersey
New Mexico
Nevada
New York
Ohio
Oklahoma
Oregon
Pennsylvania
Puerto Rico
Rhode Island
South Carolina
South Dakota
Tennessee
Texas
Utah
Virginia
Vermont
Washington
Wisconsin
West Virginia
Wyoming

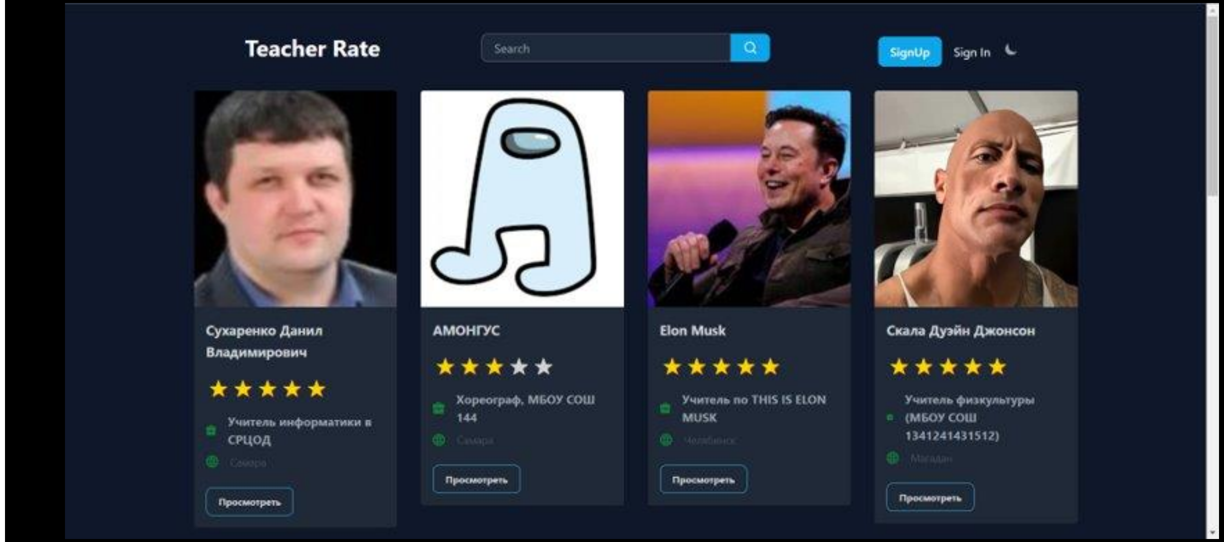
Стек технологий

- Django Rest Framework(backend)
- Angular 13(frontend)
- Tailwind

Преимущества Django Rest Framwork над Django



Наше решение



Сухаренко Данил Владимирович

★★★★★

Учитель информатики в СРЦОД

Самара

Просмотреть

АМОНГУС

★★★☆☆

Хореограф, МБОУ СОШ 144

Самара

Просмотреть

Elon Musk

★★★★★

Учитель по THIS IS ELON MUSK

Челябинск

Просмотреть

Скала Дуэйн Джонсон

★★★★★

Учитель физкультуры (МБОУ СОШ 1341241431512)

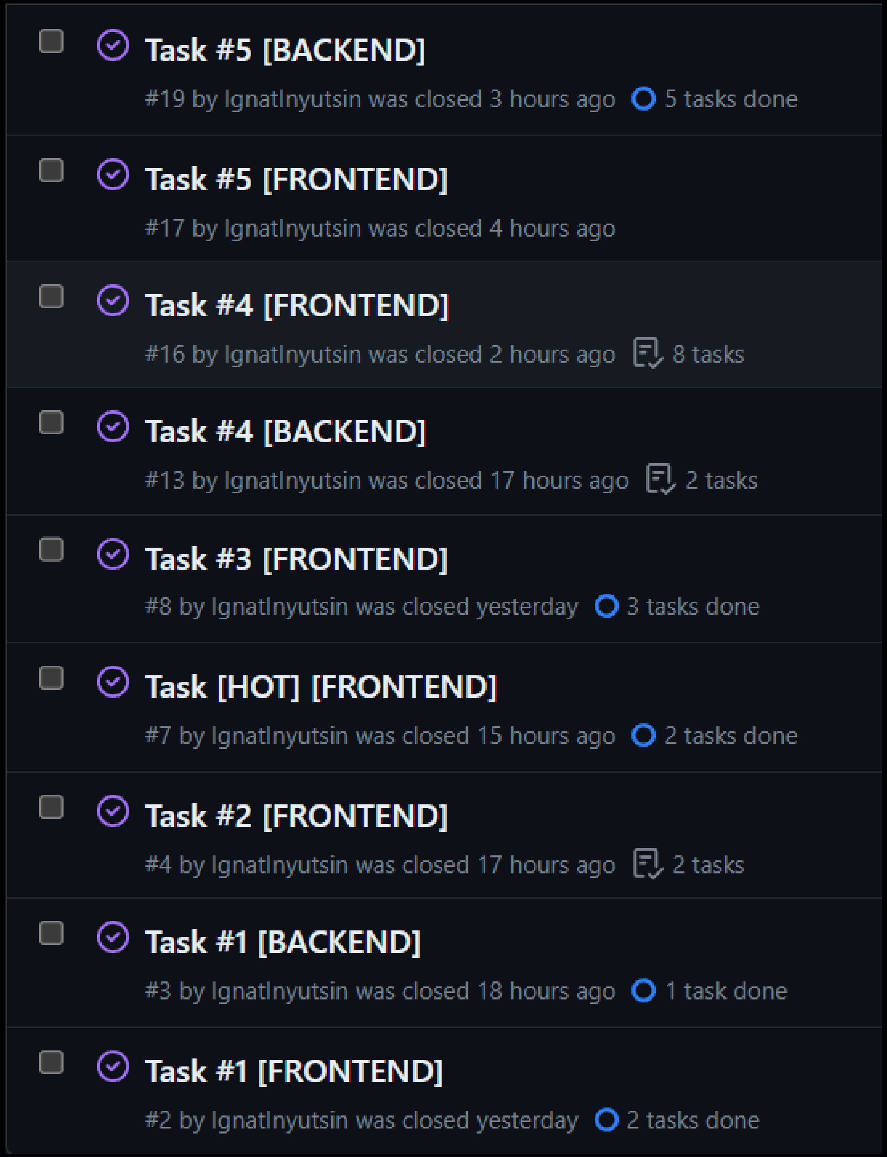
Магадан

Просмотреть

Преимущества

- Открытый исходный код
- Приятный интерфейс
- Удобная для дополнения архитектура приложения
- Документация

Доска задач



□ Task #5 [BACKEND]
#19 by IgnatInyutsin was closed 3 hours ago ○ 5 tasks done

□ Task #5 [FRONTEND]
#17 by IgnatInyutsin was closed 4 hours ago

□ Task #4 [FRONTEND]
#16 by IgnatInyutsin was closed 2 hours ago 8 tasks

□ Task #4 [BACKEND]
#13 by IgnatInyutsin was closed 17 hours ago 2 tasks

□ Task #3 [FRONTEND]
#8 by IgnatInyutsin was closed yesterday ○ 3 tasks done

□ Task [HOT] [FRONTEND]
#7 by IgnatInyutsin was closed 15 hours ago ○ 2 tasks done

□ Task #2 [FRONTEND]
#4 by IgnatInyutsin was closed 17 hours ago 2 tasks

□ Task #1 [BACKEND]
#3 by IgnatInyutsin was closed 18 hours ago ○ 1 task done

□ Task #1 [FRONTEND]
#2 by IgnatInyutsin was closed yesterday ○ 2 tasks done

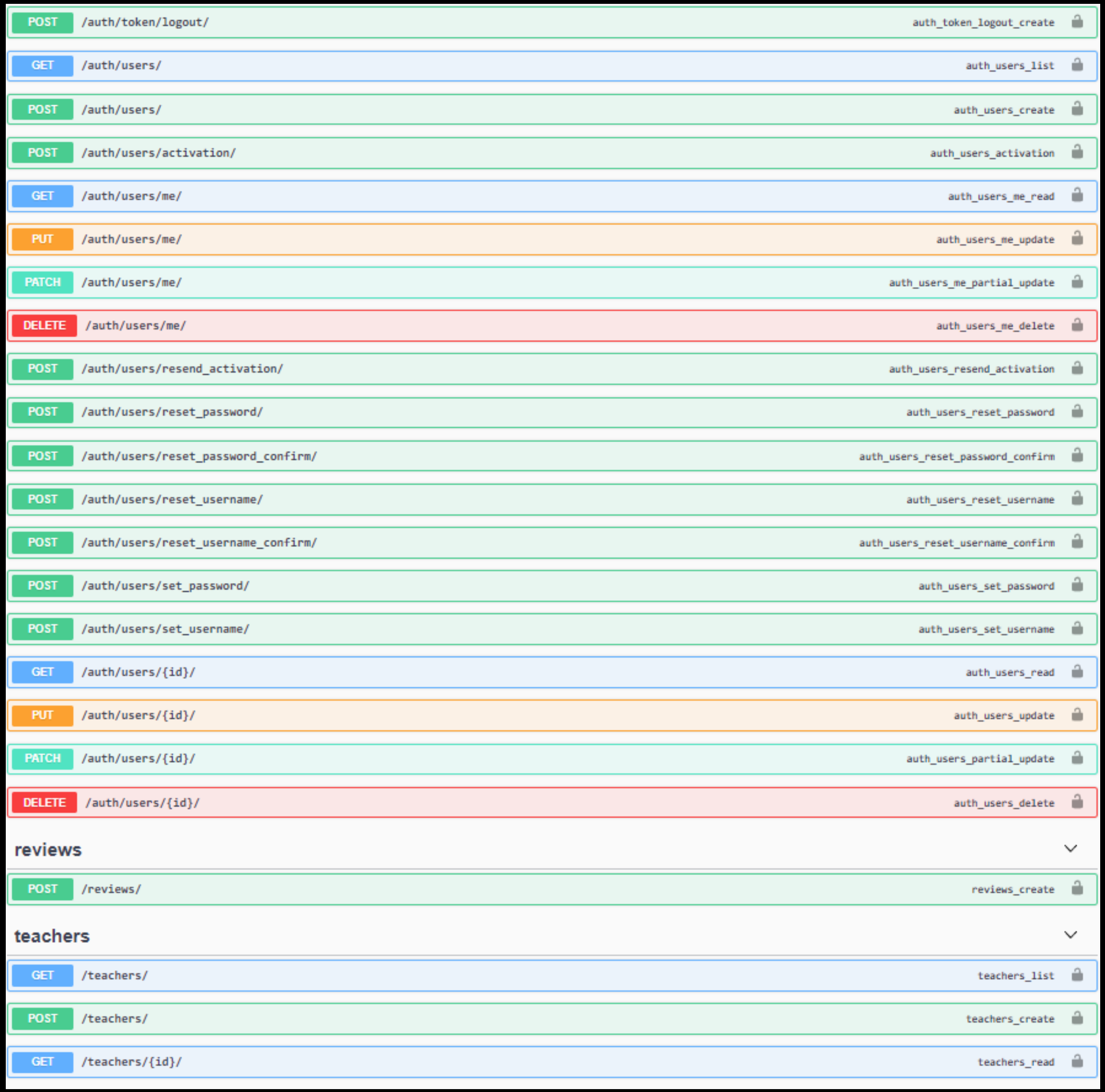
Функционал

- Регистрация аккаунта
- Вход в аккаунт
- Добавить учителя
- Найти учителя
- Посмотреть оценку и коментарии к учителю
- Поставить оценку и написать коментарий учителю

В планах

- Разделить учителей на секции
- Сделать поиск по секциям

Документация



POST	/auth/token/logout/	auth_token_logout_create
GET	/auth/users/	auth_users_list
POST	/auth/users/	auth_users_create
POST	/auth/users/activation/	auth_users_activation
GET	/auth/users/me/	auth_users_me_read
PUT	/auth/users/me/	auth_users_me_update
PATCH	/auth/users/me/	auth_users_me_partial_update
DELETE	/auth/users/me/	auth_users_me_delete
POST	/auth/users/resend_activation/	auth_users_resend_activation
POST	/auth/users/reset_password/	auth_users_reset_password
POST	/auth/users/reset_password_confirm/	auth_users_reset_password_confirm
POST	/auth/users/reset_username/	auth_users_reset_username
POST	/auth/users/reset_username_confirm/	auth_users_reset_username_confirm
POST	/auth/users/set_password/	auth_users_set_password
POST	/auth/users/set_username/	auth_users_set_username
GET	/auth/users/{id}/	auth_users_read
PUT	/auth/users/{id}/	auth_users_update
PATCH	/auth/users/{id}/	auth_users_partial_update
DELETE	/auth/users/{id}/	auth_users_delete

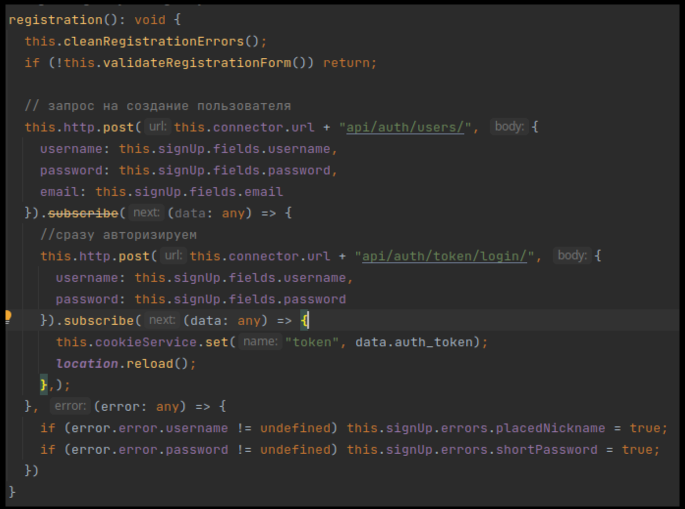
reviews

POST	/reviews/	reviews_create

teachers

GET	/teachers/	teachers_list
POST	/teachers/	teachers_create
GET	/teachers/{id}/	teachers_read

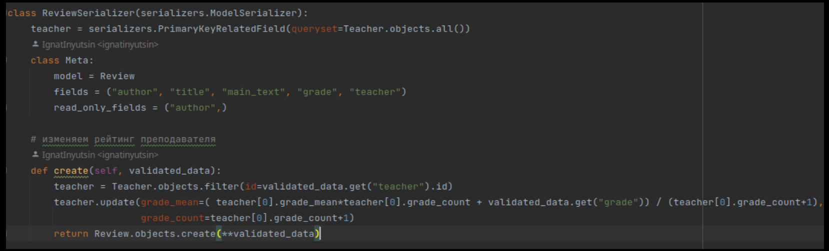
Примеры кода



```
registration(): void {
  this.cleanRegistrationErrors();
  if (!this.validateRegistrationForm()) return;

  // запрос на создание пользователя
  this.http.post( url: this.connector.url + "api/auth/users/",  body: {
    username: this.signUp.fields.username,
    password: this.signUp.fields.password,
    email: this.signUp.fields.email
  }).subscribe( next: (data: any) => {
    //сразу авторизируем
    this.http.post( url: this.connector.url + "api/auth/token/login/",  body: {
      username: this.signUp.fields.username,
      password: this.signUp.fields.password
    }).subscribe( next: (data: any) => {
      this.cookieService.set( name: "token", data.auth_token);
      location.reload();
    },);
  },  error: (error: any) => {
    if (error.error.username != undefined) this.signUp.errors.placedNickname = true;
    if (error.error.password != undefined) this.signUp.errors.shortPassword = true;
  })
}
```

Регистрация со стороны frontend



```
class ReviewSerializer(serializers.ModelSerializer):
    teacher = serializers.PrimaryKeyRelatedField(queryset=Teacher.objects.all())
    IgnatInyutsin <ignatinyutsin>
    class Meta:
        model = Review
        fields = ("author", "title", "main_text", "grade", "teacher")
        read_only_fields = ("author",)

    # изменяем рейтинг преподавателя
    IgnatInyutsin <ignatinyutsin>
    def create(self, validated_data):
        teacher = Teacher.objects.filter(id=validated_data.get("teacher").id)
        teacher.update(grade_mean=( teacher[0].grade_mean*teacher[0].grade_count + validated_data.get("grade")) / (teacher[0].grade_count+1),
                       grade_count=teacher[0].grade_count+1)
        return Review.objects.create(**validated_data)
```

Сериализатор для создания отзыва

Наш продукт

Выводы

Спасибо за внимание!