

**САНКТ-ПЕТЕРБУРГСКИЙ НАЦИОНАЛЬНЫЙ
ИССЛЕДОВАТЕЛЬСКИЙ УНИВЕРСИТЕТ ИТМО**

Дисциплина: Бекенд разработка

Отчет

Лабораторная работа №3

Выполнил:

Золотов Павел

Группа:
К33401

Проверил:
Говоров А. И.

Санкт-Петербург

2021 г.

Задача

Реализация серверной части приложения средствами django и django rest framework в соответствии с заданием из текста работы.

Ход работы

Был выбран вариант с созданием сайта вакансий.

1) Созданные модели:

Индустрия компании

```
class Industry(models.Model):
    name = models.CharField(max_length=50)

    def __str__(self):
        return self.name
```

Регион компании:

```
class Region(models.Model):
    name = models.CharField(max_length=50)

    def __str__(self):
        return self.name
```

Компания:

```
class Company(models.Model):
    name = models.CharField(max_length=200)
    industry = models.ForeignKey(Industry, on_delete=models.CASCADE)
    regions = models.ManyToManyField(Region)
    rating = models.FloatField(default=0, validators=[MinValueValidator(0),
MaxValueValidator(5)])
    description = models.TextField(max_length=500)
    verified = models.BooleanField(default=False)

    def __str__(self):
        return self.name
```

Вакансия:

```
class Job(models.Model):
    title = models.CharField(max_length=100)
    description = models.TextField(max_length=2000)
    min_years_experience = models.IntegerField(default=0,
validators=[MinValueValidator(0), MaxValueValidator(10)])
    remote_available = models.BooleanField(default=False)
    open = models.BooleanField(default=True)
    company = models.ForeignKey(Company, on_delete=models.CASCADE)

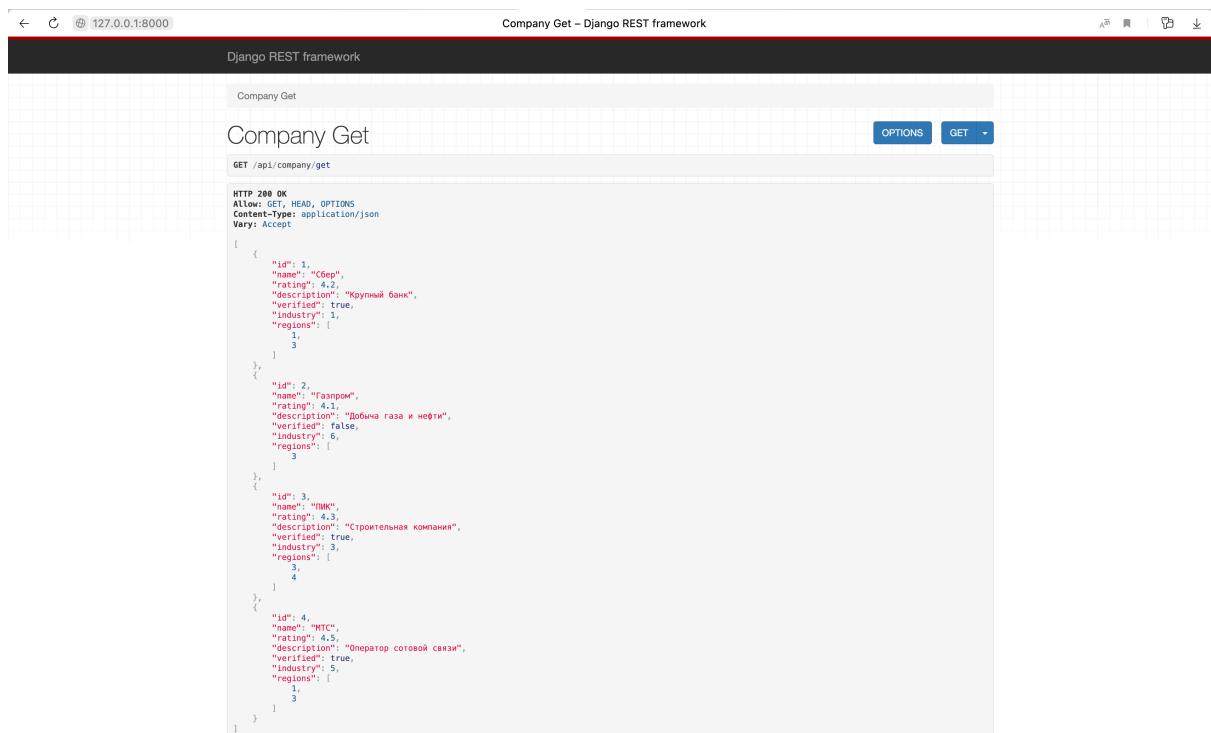
    def __str__(self):
        return "{} в {}".format(self.title, self.company.name)
```

Отклик на вакансию:

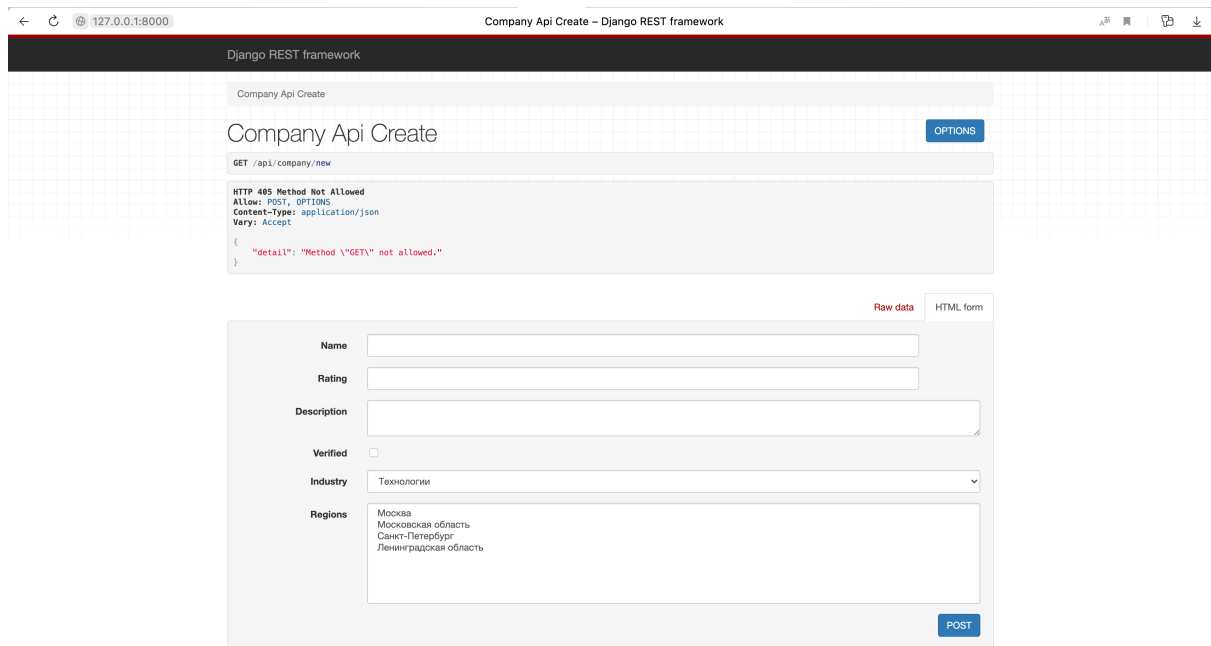
```
class JobResponse(models.Model):
    text = models.TextField(max_length=500)
    job = models.ForeignKey(Job, on_delete=models.CASCADE)
    cv = models.URLField()
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    created_at = models.DateTimeField()
```

2) Подключен и настроен DRF

Пример страницы просмотра данных о компаниях:



Создание новой компании:

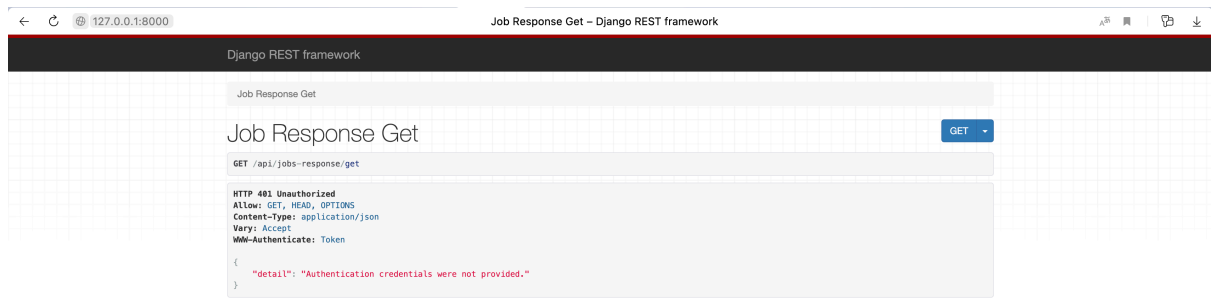


3) Настроена работа Djoser

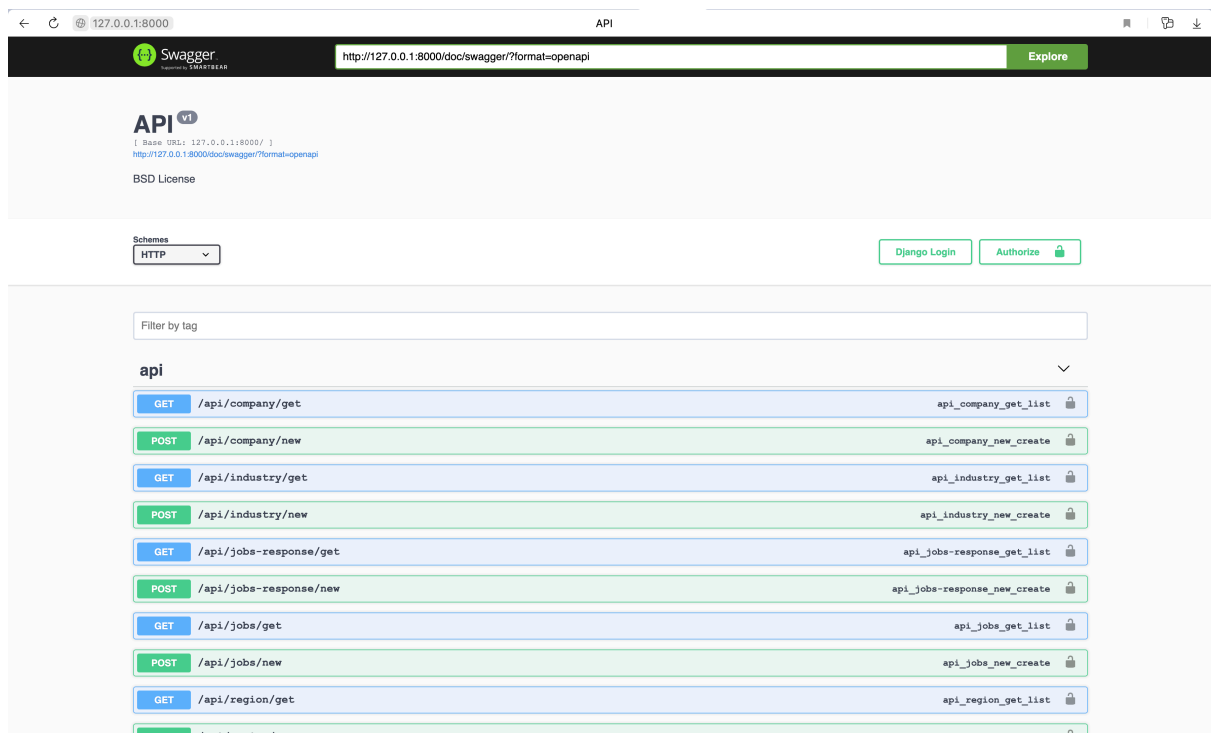
Эндпоинты для работы с пользователем:

auth		
POST	/auth/token/login/	auth_token_login_create
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

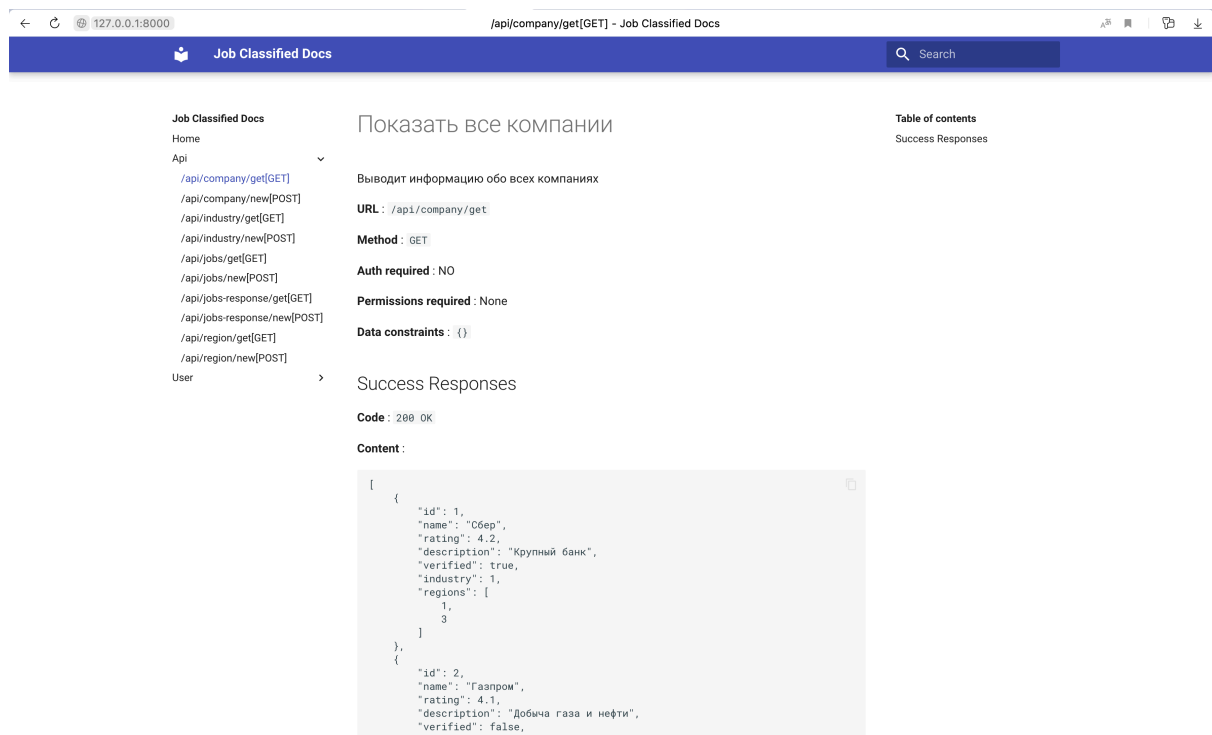
Пример страницы, требующей авторизацию пользователя:



4) Подключен Swagger



5) Добавлена документация на базе MkDocs



Вывод

В результате выполнения работы были получены начальные навыки работы с Django REST Framework, Djoser, а также навыки документирования с помощью Swagger и MkDocs.