**Описание тестового задания**

# Требования к технологии

Необходимо разработать django приложение, которое будет предоставлять REST интерфейс, обеспечивающий авторизацию, доступ к read-only данным, CRUD операции со связными моделями данных. В качестве REST библиотеки необходимо использовать django-rest-framework.

В качестве БД с хранимыми данными проще всего использовать sqlite.

Разработанное API должно быть доступно для REST киентов по токену.

# 2. Макет приложения

По ссылке показано как должно потом выгядить веб-приложение, которое будет обращаться к API, разрабатываемое в рамках данного задания: <http://quadrogroup.ru/test/>

# 3. Тестовое задание

## Этап 1. Авторизация

Авторизация осуществляется REST запросом на URL:

[http://some\_domain/app/auth](http://test.quadrogroup.ru:4343/api/auth/)/login/

POST-ом передается username и password. В ответ приходит редирект, в котором в cookie выставлены sessionid и csrftoken. Для дальшейших запросов, они должны присутсвовать в cookie.

Logout для завершения сессии:

[http://some\_domain/app/auth](http://test.quadrogroup.ru:4343/api/auth/)/logout/

## Этап 2. Таблица численности населения по городам и странам

REST метод для получения данных - только для чтения:

[http://some\_domain/app](http://test.quadrogroup.ru:4343/api/auth/)[/api/stat/](http://test.quadrogroup.ru:4343/api/data/)

Структура полей для каждой записи:

* Id
* Country (string)
* City (string)
* Population (number)

## Этап 3. Редактор товаров и их категорий

Выгрузка и редактирование списка товаров. Необходимо обратить внимание, что связь между товарами и категориями - многие-ко-многим.

Список объектов из API выдаются в постраничном виде (по две записи по умолчанию).

### Структура данных:

#### Товар (Item):

* id
* name (string)
* categories (список Categories)
* value\_int (number - кол-во штук)
* value\_float (float - стоимость)

REST URL для CRUD операций:

[http://some\_domain/app](http://test.quadrogroup.ru:4343/api/auth/)[/api/item/](http://test.quadrogroup.ru:4343/hr/api/item/)

#### Категории (Category)

* id
* name (string)
* description (string)

REST URL для CRUD операций:

[http://some\_domain/app](http://test.quadrogroup.ru:4343/api/auth/)[/api/category/](http://test.quadrogroup.ru:4343/hr/api/category/)

# 4. Админ интерфейс

У приложения должен полноценно работать django admin интерфейс.

# 5. Проверка задания

Готовое приложение необходимо отправить в заархивированном виде на нашу почту hr@quadrogroup.ru.

# 6. Общие рекомендации и на что стоит обратить внимание

1. Читаемость кода
2. Структура приложения
3. Осмысленный и мотивированный выбор библиотек (если потребуются)