

**Московский государственный технический
университет им. Н.Э. Баумана**

Факультет «Информатика и системы управления»
Кафедра ИУ5 «Системы обработки информации и управления»

Курс «Разработка интернет приложений»

Отчет по лабораторной работе №6

«Разработка REST API с использованием Django REST Framework.»

Выполнил:
студент группы ИУ5-54Б
Низовцев Роман

Проверил:
преподаватель каф. ИУ5
Гапанюк Ю.Е.

Москва, 2021 г.

Задание:

1. С использованием Django REST Framework разработайте REST API для одной модели (одной таблицы базы данных).

Текст программы

Urls.py

```
from django.contrib import admin
from stocks import views as stock_views
from django.urls import include, path
from rest_framework import routers

router = routers.DefaultRouter()
router.register(r'os', stock_views.StockViewSet)

# Wire up our API using automatic URL routing.
# Additionally, we include login URLs for the browsable API.
urlpatterns = [
    path('', include(router.urls)),
    path('api-auth/', include('rest_framework.urls',
namespace='rest_framework'))),
    path('admin/', admin.site.urls),
```

Views.py

```
from rest_framework import viewsets
from stocks.serializers import StockSerializer
from stocks.models import Stock

class StockViewSet(viewsets.ModelViewSet):

    # queryset всех пользователей для фильтрации по дате последнего изменения
    queryset = Stock.objects.all().order_by('pk')
    serializer_class = StockSerializer # Сериализатор для модели
```

Serializers.py

```
from stocks.models import Stock
from rest_framework import serializers

class StockSerializer(serializers.ModelSerializer):
    class Meta:
        # Модель, которую мы сериализуем
        model = Stock
```

```
# Поля, которые мы сериализуем
fields = ["pk", "os name", "last version", "os descript", "src"]
```

Models.py

```
from django.db import models

class Stock(models.Model):
    os_name = models.CharField(max_length=50, verbose_name="Название ОС")
    last_version = models.DecimalField(max_digits=8, decimal_places=2,
    verbose_name="Номер версии")
    os_descript = models.CharField(max_length=100, verbose_name="Описание ОС")
    src = models.CharField(max_length=150, verbose_name="URL картинки")
```

Результат выполнения программы:

GET /os/

HTTP 200 OK
Allow: GET, POST, HEAD, OPTIONS
Content-Type: application/json
Vary: Accept

```
[
  {
    "pk": 14,
    "os_name": "Android",
    "last_version": "12.00",
    "os_descript": "Мобильная ОС от гугл",
    "src": "https://3dnews.ru/assets/external/illustrations/2011/10/22/618753/android-logo-white.png"
  },
  {
    "pk": 15,
    "os_name": "IOS",
    "last_version": "14.00",
    "os_descript": "Мобильная ОС от Apple",
    "src": "https://zillya.ua/sites/default/files/apple-ios-update_0.png"
  },
  {
    "pk": 16,
    "os_name": "Windows Mob",
    "last_version": "8.00",
    "os_descript": "Мобильная ОС от Microsoft",
    "src": "https://upload.wikimedia.org/wikipedia/ru/7/7b/Windows_Mobile_6_Series_logo.jpg"
  }
]
```