

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

Факультет «Информатика и управление»

Кафедра «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
Киреев А.А.
Подпись и дата:

Москва, 2021 г.

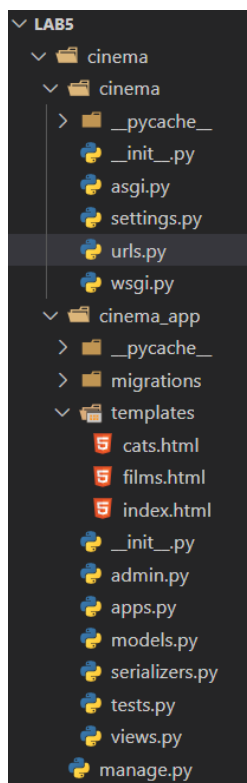
Цель лабораторной работы: изучение возможностей разработки REST API с использованием Django REST Framework.

Задание:

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

Общая структура веб-приложения:

Веб-приложение для разработки REST API основано на приложении из 5-ой лабораторной работы.



Текст программы.

Текст программы аналогичен 5-ой лабораторной, но с некоторыми дополнениями.

```
urls.py X
cinema > cinema > urls.py > ...
14 | 2. Add a URL to urlpatterns: path('blog/', include('blog.urls'))
15 | """
16 | from django.contrib import admin
17 | from django.urls import include, path
18 | from cinema_app.views import *
19 | from cinema_app import views as film_views
20 | from rest_framework import routers
21 |
22 | router = routers.DefaultRouter()
23 | router.register(r'films', film_views.FilmViewSet)
24 |
25 | urlpatterns = [
26 |     path('', include(router.urls)),
27 |     path('api-auth/', include('rest_framework.urls', namespace='rest_framework')),
28 |     path('', index, name="index"),
29 |     path('cats/<int:id>/', cats, name="cats"),
30 |     path('films/<int:id>/', films, name="films"),
31 |     path('admin/', admin.site.urls),
32 | ]
33 |
```

```
urls.py views.py X
cinema > cinema_app > views.py > FilmViewSet
1 | from django.shortcuts import render
2 | from django.http import HttpResponseRedirect
3 | from django.http import HttpResponseNotFound
4 | from cinema_app.models import *
5 | from rest_framework import viewsets
6 | from cinema_app.serializers import FilmSerializer
7 |
8 |
9 |
10 | def index(request):
11 |     cats = Category.objects.all()
12 |     return render(request, "index.html", {"cats": cats})
13 |
14 | def cats(request, id):
15 |     films = Films.objects.filter(genre=id)
16 |     return render(request, "cats.html", {"films": films})
17 |
18 | def films(request, id):
19 |     film = Films.objects.get(id=id)
20 |     return render(request, "films.html", {"film": film})
21 |
22 | class FilmViewSet(viewsets.ModelViewSet):
23 |     """
24 |     API endpoint, который позволяет просматривать и редактировать акции компаний
25 |     """
26 |     # queryset всех пользователей для фильтрации по дате последнего изменения
27 |     queryset = Films.objects.all().filter(id__lt = 100)
28 |     serializer_class = FilmSerializer # Сериализатор для модели
```

```
urls.py serializers.py X
cinema > cinema_app > serializers.py > ...
1 | from cinema_app.models import *
2 | from rest_framework import serializers
3 |
4 |
5 | class FilmSerializer(serializers.ModelSerializer):
6 |     class Meta:
7 |         # Модель, которую мы сериализуем
8 |         model = Films
9 |         # Поля, которые мы сериализуем
10 |         fields = ["id", "title_film", "year", "duration", "genre"]
```

Результат работы веб-приложения.

