
django-sage-streaming Documentation

Release 0.1.0

SageTeam

Jul 18, 2021

CONTENTS

1	Documentation	3
1.1	Installation	3
1.2	Usage	3
1.3	Admin	4
1.4	Settings	4
1.5	Frontend	5
2	Issues	7
3	Indices and tables	9



This app supports the following combinations of Django and Python:

Django	Python
3.1	3.7, 3.8, 3.9
3.2	3.7, 3.8, 3.9

DOCUMENTATION

1.1 Installation

First install package

```
$ pip install django-sage-streaming
```

Then add `sage_stream` to `INSTALLED_APPS` in `settings.py`

```
INSTALLED_APPS = [  
    ...  
    'sage_stream',  
    ...  
]
```

Also make sure you have `rest_framework` in `INSTALLED_APPS`

```
INSTALLED_APPS = [  
    ...  
    'rest_framework',  
    ...  
]
```

1.2 Usage

`django-sage-streaming` provides two ways that you can stream videos over HTTP: 1. built-in API View: for using built-in view add `sage_stream.api.urls` to `urls.py`

```
from django.urls import path, include  
urlpatterns = [  
    ...  
    path('api/', include('sage_stream.api.urls')),  
    ...  
]
```

Now you can access the stream api here: `localhost:8000/api/stream?path=<you_video_path>`

The settings of built-in view is available here

2. create custom views: you can use stream functions in your application

```
from sage_stream.utils.stream_services import get_streaming_response

response = get_streaming_response(
    path=video_path, # path to video
    range_header=range_header, # range header extracted from request
    range_re=range_re, # range header regex pattern (default is available in sage_stream.
    ↪ settings.STREAM_RANGE_HEADER_REGEX_PATTERN)
    max_load_volume=max_load_volume, # the maximum volume of the response body
)
```

Other functions you can use

```
from sage_stream.utils.file_services import file_iterator # iterates in given file.
↪ chunk by chunk in generator mode
from sage_stream.utils.log_services import log_watch_request # creates WatchLog.
↪ instance with given data

# usage examples

# log
log_watch_request(video_path, request.user.is_authenticated, ip, request.user)

# file iterator
resp = StreamingHttpResponse(
    file_iterator(
        path,
        offset=first_byte,
        length=length
    ),
    status=206,
    content_type=content_type
)
```

1.3 Admin

django-sage-streaming also has logging system for watching requests

It is available in admin panel when you add sage_stream to INSTALLED_APPS

1.4 Settings

Here are the parameters that you can set from setting:

Parameter	Description
STREAM_DEFAULT_PERMISSION_CLASSES	permission_classes that used in built-in APIView. default: (AllowAny,)
STREAM_DEFAULT_VIDEO_PATH_URL_VARIABLE	url path variable key. default: path
STREAM_MAX_LOAD_VOLUME	maximum load video in each chunk(by MB). default: 8
STREAM_WATCH_LOG_ENABLED	is log watch requests enabled. default: True
STREAM_RANGE_HEADER_REGEX_PATTERN	range header regex pattern. default: r'bytes=(+)-()'

1.5 Frontend

Frontend implementation is so easy

you just need to add stream url as `src` in `video` tag

```
<video width="480" controls>
  <source src="http://localhost:8000/api/stream/?path=<path_to_video>" type="video/mp4
  ">
</video>
```

CHAPTER TWO

ISSUES

If you have questions or have trouble using the app please file a bug report at:

<https://github.com/sageteam-org/django-sage-streaming/issues>

INDICES AND TABLES

- search