

Flask-Dropzone

Release 1.5.1

Grey Li

该文档是吴依杰根据Flask-Dropzone官方文档整理, 文档所有权归原作者所有

通过Dropzone.js在Flask应用程序中上传文件

注意:这个扩展是为简单使用而建立的, 如果你需要更多的灵活性,请直接使用Dropzone.js

基本用法

安装

```
1 pip install flask-dropzone
```

初始化

初始化扩展

```
1 from flask_dropzone import Dropzone
2
3 app = Flask(__name__)
4 dropzone = Dropzone(app)
```

这个扩展还支持Flask application factory模式, 允许你创建一个Dropzone对象, 然后为一个app初始化它

```
1 dropzone = Dropzone()
2
3 def create_app(config):
4     app = Flask(__name__)
5     ...
6     dropzone.init_app(app)
7     ...
8     return app
```

包含Dropzone.js资源

除了自己管理和加载资源之外(推荐), 您还可以使用此方法来加载资源

```
1 <head>
2 {{ dropzone.load_css() }}
3 </head>
4 <body>
5 ...
6 {{ dropzone.load_js() }}
7 </body>
```

提示:有一个`dropzone.load()`方法是`dropzone.load_css()`和`dropzone.load_js()`的组合, 但是我们建议不要使用这个方法考虑页面加载性能. 另外, `dropzone.load()`将在不久的将来被删除.

你可以通过版本参数来指定Dropzone.js的版本, 默认值是5.2.0 另外,可以分别传递`css_url`和`js_url`来定制资源URL

Create a Drop Zone

使用`create()`创建一个Drop Zone, 并使用`config()`使配置生效

```
1 {{ dropzone.create(action='the_url_or_endpoint_which_handle_uploads') }}
2 ...
3 {{ dropzone.config() }}
4 </body>
```

记得编辑处理上传的URL或endpoint的`action`参数, 例如`dropzone.create(action='upload_view')`, `dropzone.create(action=url_for('upload_view'))`

美化美化

通过`dropzone.style()`方法根据您的偏好设置样式

```
1 <head>
2 {{ dropzone.load_css() }}
3 {{ dropzone.style('border: 2px dashed #0087F7; margin: 10%; min-height: 400px;') }}
4 </head>
```

用Flask保存上传结果

当文件被拖到drop zone里以后, 你可以在`request.files`里获取上传的文件, 只需传递upload input的`name`属性(默认为`file`)

```
1 import os
```

```

2 from flask import Flask, request
3 from flask_dropzone import Dropzone
4
5 app = Flask(__name__)
6
7 dropzone = Dropzone(app)
8
9 @app.route('/uploads', methods=['GET', 'POST'])
10 def upload():
11     if request.method == 'POST':
12         f = request.files.get('file')
13         f.save(os.path.join('the/path/to/save', f.filename))
14     return 'upload template'
15

```

配置

注册配置

除了DROPZONE_SERVE_LOCAL, 当您使用其他配置变量时, 您必须在模板中调用dropzone.config()来让Dropzone注册它们

```

1 <body>
2 ...
3 {{ dropzone.config() }}
4 </body>

```

注意: 在dropzone.load_js()或包含Dropzone.js的<script>之后再调用这个方法

可用配置

支持的配置选项列表如下:

名称[Name]	默认值 [Default Value]	信息[Info]
DROPZONE_SERVE_LOCAL	False	默认从CDN检索dropzone.js
DROPZONE_MAX_FILE_SIZE	3	最大允许文件大小. 单位:MB
DROPZONE_INPUT_NAME	file	<input>中的name属性 (i.e. <input type="file" name="file">)
DROPZONE_ALLOWED_FILE_CUSTOM	False	请参见下面的细节
DROPZONE_ALLOWED_FILE_TYPE	default	请参见下面的细节

DROPZONE_MAX_FILES	null	用户一次可以上传的最大文件数
DROPZONE_DEFAULT_MESSAGE	Drop files here to upload	拖拽区显示的信息，你可以在这里写入HTML（e.g. Drop files here Or <button type="button">Click to Upload</button>）
DROPZONE_INVALID_FILE_TYPE	You can't upload files of this type.	Error message
DROPZONE_FILE_TOO_BIG	File is too big {{ filesize }}, Max filesize:{{ maxFileSize }}MiB.	Error message
DROPZONE_SERVER_ERROR	Server error:{{ statusCode }}	Error message
DROPZONE_MAX_FILE_EXCEED	You csn't upload any more files.	Error message
DROPZONE_UPLOAD_MULTIPLE	False	是否在一个请求中发送多个文件
DROPZONE_PARALLEL_UPLOADS	2	当DROPZONE_UPLOAD_MULTIPLE设置为True时，每个请求将处理多少上传
DROPZONE_REDIRECT_VIEW	None	完成上载后要重定向的视图，如果你想传递一个URL，通常当你的视图接受变量时，您可以通过模板中的 redirect_url关键字来传递它： {{ dropzone.config(redirect_url=url_for('endpoint', foo=bar)) }}
DROPZONE_ENABLE_CSRF	False	Enable CSRF protect, see detail below
DROPZONE_TIMEOUT	None	取消上传请求的超时时间为毫秒，默认为30000(30秒)。 设置一个大的数字，如果你需要上传大文件

文件类型过滤器

只需设置DROPZONE_ALLOWED_FILE_TYPE即可

- default
- image
- audio

- video
- text
- app

```
1 app.config['DROPZONE_ALLOWED_FILE_TYPE'] = 'image'
```

如果你想自己设置允许的文件类型，你需要将DROPZONE_ALLOWED_FILE_CUSTOM设置为True, 然后将mime类型或文件扩展名添加到DROPZONE_ALLOWED_FILE_TYPE, 例如:

```
1 app.config['DROPZONE_ALLOWED_FILE_CUSTOM'] = True
2 app.config['DROPZONE_ALLOWED_FILE_TYPE'] = 'image/*, .pdf, .txt'
```

可以访问:<https://www.dropzonejs.com/> 来查看详细操作信息

自定义配置字符串

有时你可能需要更灵活, 你可以使用custom_init和custom_options来传递自定义JavaScript代码

```
1 {{ dropzone.config(custom_init='dz = this;document.getElementById("upload-
  btn").addEventListener("click", function handler(e)
  {dz.processQueue();});','custom_options='autoProcessQueue: false, addRemoveLinks: true,
  parallelUploads: 20,') }}
```

带有custom_init的代码将传递到init: function() {}中,这段代码将custom_options传递到Dropzone.options.myDropzone = {}中, 请参阅<https://www.dropzonejs.com/#configuration>文档中可用配置设置的完整列表.

覆盖全局配置

有时, 您可能希望对不同页面上的多个droparea使用不同的配置,在这种情况下,您可以直接将特定的关键字参数传递到dropzone.config()

关键字参数应该以这种方式映射相应的配置变量:

- DROPZONE_DEFAULT_MESSAGE:default_message
- DROPZONE_TIMEOUT:timeout
- DROPZONE_ALLOWED_FILE_TYPE:allowed_file_type

```
1 {{ dropzone.config(max_files=10, timeout=10000, default_message='Drop here!') }}
```

最后, 您传递的关键字参数将覆盖相应的配置

高级用法

平行的上传

如果将`DROPZONE_UPLOAD_MULTIPLE`设置为`True`, 则需要在单个请求中保存多个上传, 但是, 您不能使用`request.files.getlist('file')`来获得文件列表. 当您启用并行上传时, `Dropzone.js`会在每个文件后附加一个索引号, 例如: `file[2]`, `file[1]`, `file[0]`. 所以, 你必须像这样保存文件:

```
1 for key, f in request.files.items():
2     if key.startswith('file'):
3         f.save(os.path.join('the/path/to/save', f.filename))
```

下面是完整的例子

```
1 ...
2 app.config['DROPZONE_UPLOAD_MULTIPLE'] = True    # enable parallel upload
3 app.config['DROPZONE_PARALLEL_UPLOADS'] = 3     # handle 3 file per request
4
5 @app.route('/upload', methods=['GET', 'POST'])
6 def upload():
7     if request.method == 'POST':
8         for key, f in request.files.items():
9             if key.startswith('file'): f.save(os.path.join('the/path/to/save', f.filename))
10    return 'upload template'
```

CSRF保护

CSRF保护功能是由Flask-WTF的`CSRFProtect`扩展提供的, 所以你必须先安装Flask-WTF:

```
1 pip install flask-wtf
```

然后初始化`CSRFProtect`

```
1 from flask_wtf.csrf import CSRFProtect
2
3 app = Flask(__name__)
4 # the secret key used to generate CSRF token
5 app.config['SECRET_KEY'] = 'dev key'
6 ...
7 # enable CSRF protection
8 app.config['DROPZONE_ENABLE_CSRF'] = True
9
10 csrf = CSRFProtect(app)
```

请确保设置`SECRET_KEY`, 并将`DROPZONE_ENABLE_CSRF`设置为`True`. 现在所有的上传请求将被保护!

我们倾向于手动处理CSRF错误, 因为错误响应的主体将显示在文件缩略图下面的工具提示中.

```
1 from flask_wtf.csrf import CSRFProtect, CSRFError
2 ...
3 # handle CSRF error
4 @app.errorhandler(CSRFError)
5 def csrf_error(e):
6     return e.description, 400
```

这里我使用`e.description`作为错误消息, 它是由`CSRFProtect`提供的, 其中一个CSRF令牌丢失, 那么这个CSRF token就是违法的.

服务器端验证

虽然`Dropzone.js`可以处理上传的客户端验证, 但出于安全考虑, 你仍然需要设置服务器端验证. 做你通常做的事 (扩展检查, 大小检查等), 你唯一应该记住的是当有错误时返回纯文本错误消息作为响应体. 例如, 如果我们只想让用户上传扩展名为`.png`的文件, 我们可以这样验证:

```
1 @app.route('/', methods=['POST', 'GET'])
2 def upload():
3     if request.method == 'POST':
4         f = request.files.get('file')
5         if f.filename.split('.')[1] != 'png':
6             return 'PNG only!', 400      # return the error message, with a proper 4XX code
7         f.save(os.path.join('the/path/to/save', f.filename))
8     return render_template('index.html')
9
```

当您悬停上传文件的缩略图时, 将显示错误消息



如果您正在寻找关于特定函数、类或方法的信息, 文档的这一部分是为您准备的

API 参考

模版中的Dropzone

`class flask_dropzone._Dropzone`

```
static config(redirect_url=None,custom_init='',custom_options='',**kwargs)
```

功能:初始化dropzone配置

参数:

- `redirect_url`:上传完成后要重定向的URL
- `custom_init`:init中的自定义javascript代码: `function() {}`
- `custom_options`:在Dropzone.options自定义javascript代码. `myDropzone = {}`
- `**kwargs`:镜像配置变量, 小写且无前缀. 例如:`DROPZONE_UPLOAD_MULTIPLE`在这里变成了 `upload_multiple`

```
static create(action='',csrf=False,action_view='',**kwargs)
```

功能:用给定的action创建一个Dropzone表单

- version 1.4.2:添加csrf参数以启用csrf保护

- version 1.4.3:用action参数替换action_view, action_view已被弃用
- version 1.5.0:如果DROPZONE_IN_FORM=True, 创建而不是

参数:

- action:中的action属性, 传递处理上传的url
- csrf:启用或不启用CSRFprotect, 和DROPZONE_ENABLE_CSRF相同, 1.5.4版本后弃用
- action_view:处理post数据的视图, 1.4.2版本后弃用

```
static load(js_url='',css_url='',version='5.2.0')
```

功能:使用给定的版本加载Dropzone资源并初始化Dropzone配置

- 版本1.4.3中的变化:添加了js_url和css_url参数来传递自定义资源URL
- 在版本1.4.4中进行了更改:由于不灵活, 此方法被弃用. 现在它分为三种方法:
- 使用load_css()加载css资源
- 使用load_js()加载js资源
- 使用config()来配置Dropzone

参数:

- js_url:Dropzone.js的JavaScript url
- css_url:Dropzone.js的CSS url
- version:Dropzone.js的版本

```
static load_css(css_url=None,version='5.2.0')
```

功能:加载Dropzone的css资源与给定的版本

参数:

- css_url:Dropzone.js的CSS url
- version:Dropzone.js的版本

```
static load_js(js_url=None,version='5.2.0')
```

功能:加载Dropzone的js资源与给定的版本

参数:

- css_url:Dropzone.js的js url
- version:Dropzone.js的版本

```
static style(css)
```

功能:添加css到dropzone

参数:

- css:CSS code

Utils

```
flask_dropzone.utils.get_url(endpoint_or_url)
```

```
flask_dropzone.utils.random_filename(old_filename)
```