
django-email-services Documentation

Release 1.0.0

Yorgos Pagles

Jun 09, 2017

Contents

1	Installation	3
2	Development	5
3	Backends	7
3.1	Critsend	7
3.2	Postmark	7
3.3	Amazon SES	7

Email services for django providing various backends (future plans include respective statistics).

CHAPTER 1

Installation

From PyPI using pip:

```
pip install django-email-services
```

The email services package can also be installed from the repository:

```
pip install -e hg+http://bitbucket.org/pagles/django-email-services#egg=django-email-  
↪services
```

Add the 'email_services' application to your installed apps and set the following in your settings file:

```
# Replace BackendName with CritsendEmailBackend or PostmarkEmailBackend or  
↪AmazonSESBackend  
EMAIL_BACKEND = 'email_services.backends.BackendName'  
EMAIL_SERVICES_CLIENT_ID = 'your id'  
EMAIL_SERVICES_CLIENT_KEY = 'your key'
```


CHAPTER 2

Development

For the tests `django_nose` is used and coverage reports are generated. The aim is to keep the test coverage really high. To run the tests clone the repository, navigate to the `test_project` folder and (preferably after creating a virtual environment) run:

```
pip install -r requirements.txt
python manage.py test email_services
```

With the requirements installed from inside the `test_project` folder you can test the functionality:

```
python manage.py
```

Then in the python shell:

```
from django.conf import settings
from django.core.mail import send_mail
settings.EMAIL_BACKEND = 'the backend you'd like to use'
settings.EMAIL_SERVICES_CLIENT_ID = 'your user id depending on service'
settings.EMAIL_SERVICES_CLIENT_KEY = 'your key depending on service'
send_mail("subj", "body", "sender@example.com", ['recipient@example.com',
↪ "recipient@example.com"])
```


Critsend

Critsend exposes a SOAP API to interact with their service. The API works heavily with tags where we can store arbitrary information and then lookup statistics per tag. The Critsend service of django-email-services uses this feature each time an email is sent setting a tag to be the current site if the sites framework is installed.

The settings should be your username / password as EMAIL_SERVICES_CLIENT_ID / EMAIL_SERVICES_CLIENT_KEY respectively and 'email_services.backends.CritsendEmailBackend' as the EMAIL_BACKEND:

```
EMAIL_BACKEND = 'email_services.backends.CritsendEmailBackend'  
EMAIL_SERVICES_CLIENT_ID = 'your registration email with Critsend'  
EMAIL_SERVICES_CLIENT_KEY = 'your Critsend password'
```

Postmark

Postmark exposes a REST API to interact with their service.

The settings should be your api token as EMAIL_SERVICES_CLIENT_KEY and 'email_services.backends.PostmarkEmailBackend' as the EMAIL_BACKEND.

```
EMAIL_BACKEND = 'email_services.backends.PostmarkEmailBackend'  
EMAIL_SERVICES_CLIENT_KEY = 'your Postmark API key'
```

Amazon SES

For Amazon SES backend the boto library is used.

The settings should be your access id key / secret access key for AWS as `EMAIL_SERVICES_CLIENT_ID` / `EMAIL_SERVICES_CLIENT_KEY` respectively and `'email_services.backends.AmazonSESBackend'` as the `EMAIL_BACKEND`.

```
EMAIL_BACKEND = 'email_services.backends.PostmarkEmailBackend'  
EMAIL_SERVICES_CLIENT_KEY = 'your Postmark API key'
```