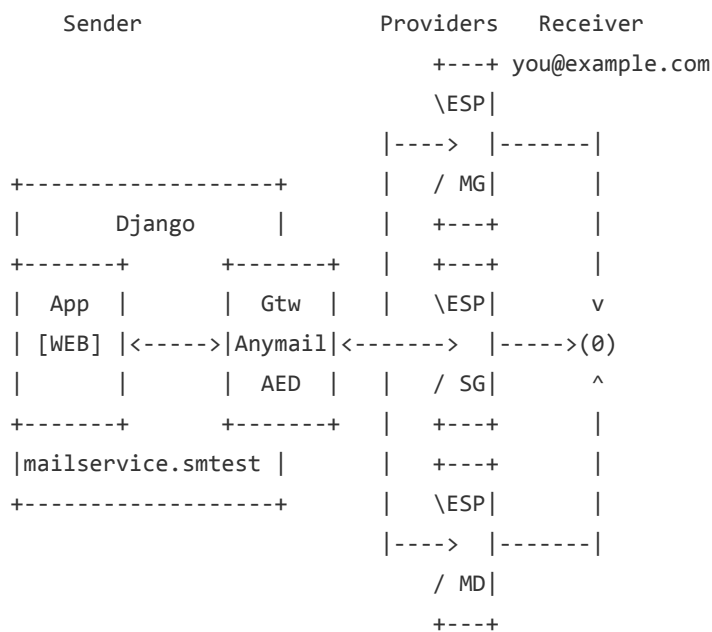


# smtest: Proof of Concept Automated Email Distribution for Email Service Providers (SendGrid, Mailgun, Mandrill)

smtest (send mail test) is a simple Python/Django service which is a thin wrapper around SendGrid, Mailgun and Mandrill APIs. It seamlessly transitions over from one provider to the other in case one stops responding. This project is Proof of Concept and implements the backend only.

## High Level Architecture Concept



## Requirement

If you have a computer that runs Python 2.7x then the application should run just fine on Windows, OS X and Linux. Then start from point 2.

It assumes that you are familiar with the terminal window (command prompt for Windows users) and know the basic command line file management functions of your operating system. If you don't, then I recommend that you learn how to create directories, copy files, etc. using the command line before continuing.

Familiarity with Python modules and packages is also recommended.

## Using the service

1. Install Python 2.7x
2. Install virtualenv: `pip install virtualenv`
3. Build test environment in local machine
  - `virtualenv devtest`

- `source devtest/bin/activate`
- 4. Install Django from PyPI: `pip install django`
- 5. Install Anymail from PyPI: `pip install django-anymail`
- 6. Get Project on Github: `https://github.com/mr700331/smttest.git`
- 7. Create ESP accounts on SendGrid [SG], Mailgun [MG] and Mandrill [MD] and get relevant information to configure `settings.py`
- 8. On ESP's account activate `Webhook` feature
- 9. Fill relevant information in `setting.py`
  - For Mailgun [MG]

code-block:: python

```
ALLOWED_HOSTS = [
    'X.X.X.X', # Public IP
    '127.0.0.1' # Localhost`
]
INSTALLED_APPS = [
    # ...
    "anymail",
    "mailservice",
    # ...
]

ANYMAIL = {
    # (exact settings here depend on your ESP...)
    "MAILGUN_API_KEY": "<your Mailgun key>",
    "MAILGUN_SENDER_DOMAIN": 'MG.example.com',
    "MAILGUN_API_URL": "https://api.mailgun.net/v3/",
}
EMAIL_BACKEND = "anymail.backends.mailgun.EmailBackend"

DEFAULT_FROM_EMAIL = "you@example.com"
```

- For SendGrid [SG]

code-block:: python

```
#...
ANYMAIL = {
    # (exact settings here depend on your ESP...)
    "SENDGRID_API_KEY": "<your SendGrid key>",
    "SENDGRID_GENERATE_MESSAGE_ID": 'True',
    "SENDGRID_API_URL": "https://api.sendgrid.com/v3/",
    "SENDGRID_MERGE_FIELD_FORMAT": "-{-}",
}
EMAIL_BACKEND = "anymail.backends.sendgrid.EmailBackend"
#EMAIL_BACKEND = "anymail.backends.mailgun.EmailBackend"

DEFAULT_FROM_EMAIL = "you@example.com"
```

.. \* For Mandrill [MD]

code-block:: python

```
#...
ANYMAIL = {
    # (exact settings here depend on your ESP...)
    "MANDRILL_API_KEY": "<your Mandrill key>",
    "MANDRILL_WEBHOOK_KEY": 'Webhook key',
    "MANDRILL_API_URL": "https://mandrillapp.com/api/1.0",
    "MANDRILL_WEBHOOK_URL": " ",
}
EMAIL_BACKEND = "anymail.backends.mandrill.EmailBackend"
#EMAIL_BACKEND = "anymail.backends.sendgrid.EmailBackend"
#EMAIL_BACKEND = "anymail.backends.mailgun.EmailBackend"

DEFAULT_FROM_EMAIL = "you@example.com"
```

.. \* Configuration Gateway [Gtw]

code-block:: python

```
#...
EMAIL_BACKEND = 'path.to.this.module.FirstWorkingEmailBackend'

FIRST_WORKING_EMAIL_BACKENDS = [
    'anymail.backends.mandrill.EmailBackend',
    'anymail.backends.sendgrid.EmailBackend',
    'anymail.backends.mailgun.EmailBackend',
]
#EMAIL_BACKEND = "anymail.backends.mandrill.EmailBackend"
#EMAIL_BACKEND = "anymail.backends.sendgrid.EmailBackend"
#EMAIL_BACKEND = "anymail.backends.mailgun.EmailBackend"

DEFAULT_FROM_EMAIL = "you@example.com"

FIRST_WORKING_EMAIL_CACHE_TIMEOUT = 60 * 60
```

1. Put email address in `mailsevice/view.py`

```
to="you@example.com"
```

2. Run server `./manage.py runserver`

## Limitation

---

Webhook activation on ESP can be restricted. This function can be paid.

In addition, testing requires access to public IP.

## Test

---

## 1. Test SendGrid ESP communication

Request URL: http://127.0.0.1:8000/  
 Request method: GET  
 Remote address: 127.0.0.1:8000  
 Status code: 200 OK  
 Version: HTTP/1.0

### \* Request Header

Cache-Control: max-age=0  
 Host: 127.0.0.1:8000  
 User-Agent: Mozilla/5.0 (X11; Ubuntu; Linux x86\_64; rv:54.0) Gecko/20100101 Firefox/54.0  
 Accept: text/html,application/xhtml+xml,application/xml;q=0.9,\*/\*;q=0.8  
 Accept-Language: en-US,en;q=0.5  
 Accept-Encoding: gzip, deflate  
 Connection: keep-alive  
 Upgrade-Insecure-Requests

### \* Response Header

Date: Mon, 18 Sep 2017 12:46:49 GMT  
 Server: WSGIServer/0.1 Python/2.7.12  
 X-Frame-Options: SAMEORIGIN  
 Content-Type: text/html; charset=utf-8  
 Content-Length: 63

### \* Result

Temat: Welcome  
 Data: Mon, 18 Sep 2017 14:46:50 +0000 (UTC)  
 Nadawca: webmaster@localhost  
 Adresat: mrakowski@zortel.pl

Welcome SNACC

### \* Trigger Bounce Event from link failure

Welcome SNACC  
 Return-Path: <bounces+6070716-f65b-mrakowski=zortel.pl@sendgrid.net>  
 Received: from o1.7qt.s2shared.sendgrid.net (167.89.106.76) (HELO o1.7qt.s2shared.sendgrid.net)  
 by domains005.home.pl (89.161.255.8) with SMTP (IdeaSmtpServer 0.82)  
 id 304e3a39596d979d; Mon, 18 Sep 2017 17:01:56 +0200  
 DKIM-Signature: v= a=rsa-sha1; c=relaxed/relaxed; d=sendgrid.net;  
 h=ntent-transfer-encoding:content-type;  
 from:mime-version:to:subject;  
 s=tpapi; bh=NaFAKF0UWiSn7VJ5CJSXETMVf6o=; b=os11CvIqmndtvETr5NfS8sbLEgIXLXc/rOmGpkzKRlOL3HADlrA1  
 Received: by filter0008p3mdw1.sendgrid.net with  
 SMTP id filter0008p3mdw1-16377-59BDFD-6D

```
2017-09-18 15:01:17.717881373 +0000 UTC
Received: from NjA3MDcxNg (ip.wnet.pl [193.34.178.69])
by ismtpd0003p1lon1.sendgrid.net (SG) with HTTP id up5g2NNfS4yXqPRCt-VQag ...
```

```
Mon, 18 Sep 2017 15:01:17.577 +0000 (UTC)
Content-Transfer-Encoding: quoted-printable
Content-Type: text/plain; charset=F-8
```

```
Date: Mon, 18 Sep 2017 15:01:36 +0000 (UTC)
From: webmaster@localhost
Mime-Version: 1.0
To: mrakowski@zortel.pl
Message-ID: <20170918150116.2320.35517@localhost>
Subject: Welcome
X-SG-EID:+VsPANuzAjRWA1m9ro ...
```

```
* Trigger Bounce Event from wrong address
```

```
Final-Recipient: rfc822; mm@zortel.pl
Action: failed
Status: 5.1.1
Remote-MTA: dns; zortel.pl. (89.161.255.8, the server for the domain zortel.pl.)
Diagnostic-Code: smtp; 550 5.1.1 User not found
Last-Attempt-Date: Mon, 18 Sep 2017 08:14:29 -0700 (PDT)
```

## 2. Test Mailgun ESP communication

Like the above

## 3. Test Mandrill ESP communication

## 4. Test Gateway

```
AnymailRequestsAPIError:
Sending a message to mrakowski@zortel.pl from webmaster@localhost
Mailgun API response 404: NOT FOUND
{
  "message": "Domain not found: localhost"
}
[18/Sep/2017 19:35:04] "GET /?to=mrakowski@zortel.pl HTTP/1.1" 500 103181

to=mrakowski@zortel.pl
status=<anymail.message.AnymailStatus object at 0x7fa335837850>
id=<20170918194551.3155.1701@localhost>
{u'mrakowski@zortel.pl': <anymail.message.AnymailRecipientStatus object at 0x7fa335837c90>}
[u'to']
====
[18/Sep/2017 19:45:52] "GET /?to=mrakowski@zortel.pl HTTP/1.1" 200 63
```

# Todo

---

1. Adding authentication
2. Better error messages if request payload is invalid (especially for the case when all required fields are not present)
3. Normalized tracking events
4. Adding support for attachments and custom headers
5. Adding installation tools or build package EGG, WHEEL
6. Adding unit tests for the mail package
7. Using a better tool than jsonschema for api validation. Jsonschema doesn't seem extendable.
8. Adding logging
9. Markdown Documentation
10. Test, Test ...