Meetup Data Scraper Documentation

Release 0.1

Steffen Exler

Dec 19, 2019

Contents

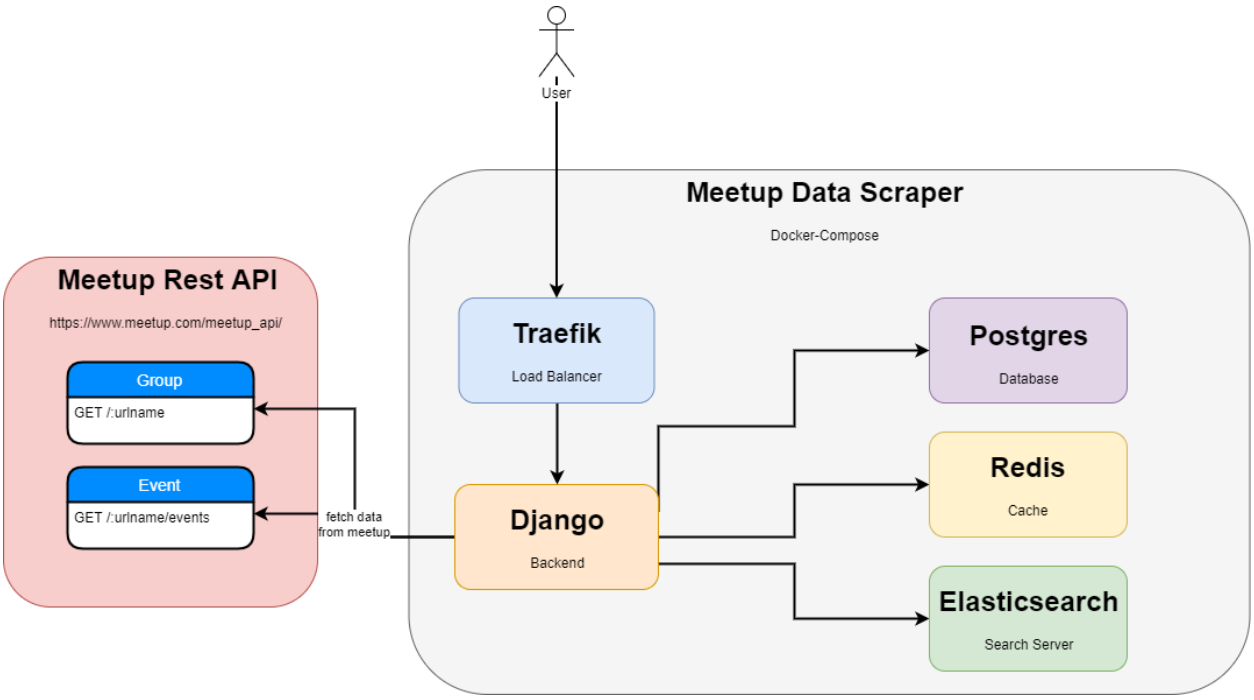

Table of Contents:

CHAPTER 1

Getting started

Note: These instructions assume familiarity with Docker and Docker Compose.

1.1 Development & Production Version

The Project comes with 2 different Docker-Compose files wich are for development `local.yml` and production `production.yml`.

The development version start the website in debug mode and bind the local path `./` to the django docker contaiers path `/app`.

For the production version, the docker container is build with the code inside of the container. Also the production version use redis as caching backend.

1.2 Quick install (Development Version)

Build the docker container.

```
$ docker-compose -f local.yml build
```

Create the sql tables or update the tables.

```
$ docker-compose -f local.yml run django python manage.py migrate
```

Create a new superuser account.

```
$ docker-compose -f local.yml run django python manage.py createsuperuser
```

Add Elasticsearch index.

```
$ docker-compose -f local.yml run django python manage.py update_index
```

Load the Meetup Sandbox Group with all events.

```
$ docker-compose -f local.yml run django python manage.py update_group --sandbox
```

Start the website.

```
$ docker-compose -f local.yml up
```

Now you can go to http://localhost:8000/ to visist your local site or to http://localhost:8000/admin/ to log in your admin panel.

1.3 Quick install (Production Version)

1.3.1 Settings

At first create the directory `./.envs/.production`

```
$ mkdir ./.envs\.production`
```

For Django container create a file `./.envs/.production/.django` wich should look like:

> **Warning:** Change DJANGO_SECRET_KEY & DJANGO_ADMIN_URL with your random strings.
>
> Don't share the DJANGO_SECRET_KEY with anybody!
>
> Share the DJANGO_ADMIN_URL only with the admins and moderators of the page! DJANGO_ADMIN_URL is the path for the admin panel, in this case it will be https://meetup-data-scraper.de/7qW3YfapGX9k3zNVftQm/

For Elasticsearch container create a file `./.envs/.production/.elasticsearch` wich should look like below. For further information how to setup Elasticsearch with enviroment vars got to https://www.elastic.co/guide/en/elasticsearch/reference/current/settings.html

For Postgres container create a file `./.envs/.production/.postgres` wich should look like:

1.3.2 Setup

Build the docker container.

```
$ docker-compose -f production.yml build
```

Create the sql tables or update the tables.

```
$ docker-compose -f production.yml run django python manage.py migrate
```

Add Elasticsearch index.

```
$ docker-compose -f production.yml run django python manage.py update_index
```

Create a new superuser account.

```
$ docker-compose -f production.yml run django python manage.py createsuperuser
```

Start the website.

```
$ docker-compose -f local.yml up -d
```

Note: For deployment instructions visit https://cookiecutter-django.readthedocs.io/en/latest/deployment-with-docker.html

!! There is no need to add a media storage (AWS S3 or GCP) for this project like it is described in cookiecutter-django docs !!

CHAPTER 2

Usage Guide

2.1 CLI

2.1.1 get_groups

Load mutiple groups from JSON files. The JSON files muss include a key & a URL name. To download use the rest api direkt or via the meetup website https://secure.meetup.com/meetup_api/console/?path=/find/groups

An example Rest API request for the first 200 german groups -> https://api.meetup.com/find/groups?&sign=true&photo-host=public&country=DE&page=200&offset=0&only=urlname

After you downloaded the json, put them into `./meetup_data_scraper`. When you download the JSON's in a another directory set the path via `--json_path /app/your-dir/`. When you run the command in docker, you need to set the path inside the docker container.

```
$ docker-compose -f local.yml run django python manage.py get_groups
```

Example JSON file in `./compose/local/django/meetup_groups/test-groups.json`

```
{
    "0": {
        "urlname": "Meetup-API-Testing"
    },
    "1": {
        "urlname": "None"
    },
    "2": {
        "urlname": "connectedawareness-berlin"
    }
}
```

2.1.2 update_group

Load a single group with all events from meetup rest api. When the group already exist in the database, it will just update the group and load new events to the group.

To set a group, use the param `--group_urlname GROUP_URLNAME`, for load the meetup sandbox group use:

```
$ docker-compose -f local.yml run django python manage.py update_group --group_
↪urlname Meetup-API-Testing
```

Or as a special case to load the sandbox group, add the param `--sandbox` without a value:

```
$ docker-compose -f local.yml run django python manage.py update_group --sandbox
```

2.1.3 update_groups

To get all new events from all groups in the database use:

```
$ docker-compose -f local.yml run django python manage.py update_groups
```

CHAPTER 3

Advanced topics

3.1 Changing Models

The models in `.meetup_data_scraper/meetup_scraper/models.py` are pages based on Wagtail CMS. For further information how to change the models read the docs from Wagtail Doc - Page Models

The hierarchy strukture of the page models is:

`HomePage` -> `GroupPage` -> `EventPage`

The default `HomePage` will automatically created on the first `migrate` process. The HomePage will only accept GroupPages as child and the GroupPages accept only EventPages.

CHAPTER 4

Troubleshooting

This page contains some advice about errors and problems commonly encountered during the development of Meetup Data Scraper.

4.1 max virtual memory areas vm.max_map_count [65530] likely too low, increase to at least [262144]

When using docker on some machines, you will need to manually extend the max virtual memory. For CentOS & Ubuntu use:

```
$ sudo sysctl -w vm.max_map_count=262144
```

4.2 Test faild

In some cases the tests can fail cause of a coruppted database. Try to reset your test database und retry the test.

CHAPTER 5

FAQ

5.1 What are the minimum hardware requirements?

To host it with docker you will need at leat a vServer with 2GB RAM, 10GB disk space & 1 CPU.

5.2 How to set the domain for a production site?

Change in `.envs/.production/.django` the value of `DJANGO_ALLOWED_HOSTS` to your domain. Also replace in `compose\production\traefik\traefik.toml` the entry `meetup-data-scraper.de` with your target domain.

CHAPTER 6

Indices & Tables

- genindex
- modindex
- search

Index

F