django-politico-slackchat-renderer Documentation

Release 0.2.14

POLITICO

May 22, 2018

Contents:


POLITICO

CHAPTER 1

Why this?

Pairs with django-slackchat-serializer, an app that serializes conversations within a Slack channel and fires webhooks whenever new messages are added.

This app responds to slackchat-serializer's webhooks and publishes the serialized data within a template. It maintains no data, itself, and therefore has no models, but is the repository of the templates used at POLITICO to render serialized SlackChats. It also includes the logic we use to publish SlackChats as static files to an AWS S3 bucket.

Note: We've open-sourced this project mostly to serve as an example of how we interact with django-slackchat-serializer.

That said, this project may not be wholly useful on its own to folks outside our team. In fact, a major part of the app is maintaining our branded templates. But if you're looking to follow our lead when rendering serialized slackchats, we encourage you to check out the source code in conjunction with these (admittedly sparse) docs.

CHAPTER 2

Quickstart

1. Install the app:

```
$ pip install django-politico-slackchat-renderer
```

2. Add the app to your installed apps and your project's urls.py.

```
# project/settings.py

INSTALLED_APPS = [
    # ...
    'chatrender',
]
```

```
# project/urls.py

urlpatterns = [
    # ...
    path('chatrender', include('chatrender.urls')),
]
```

3. Configure settings.

```
# project/settings.py

# AWS credentials and S3 bucket
CHATRENDER_AWS_ACCESS_KEY_ID = os.getenv('AWS_ACCESS_KEY_ID')
CHATRENDER_AWS_SECRET_ACCESS_KEY = os.getenv('AWS_SECRET_ACCESS_KEY')
CHATRENDER_AWS_S3_BUCKET = 'interactives.politico.com'

# Root path in bucket for publishing slackchats
CHATRENDER_AWS_S3_PUBLISH_PATH = '/interactives/slackchats/'

# The URL root the bucket is proxied to
CHATRENDER_AWS_CUSTOM_ORIGIN = 'https://www.politico.com/interactives/'
```

(continues on next page)

(continued from previous page)

```
# slackchat-serializers root API URL
CHATRENDER_SLACKCHAT_API_ENDPOINT = 'http://localhost:8000/slackchat/api/'
```

4. In django-slackchat-serializer, create a `ChatType` instance whose slug matches a renderer template name (cf. Developing) and configure a webhook to hit the events endpoint at `<chatrender>/endpoint/`.

CHAPTER 3

Workflow

In general, here's the workflow when the app responds to a webhook from slackchat-serializer.

1. slackchat-serializer hits the app's endpoint with the ID of the channel which was updated.
2. The app makes a GET request to the channel's API URL and retrieves the serialized slackchat.
3. The app uses the serialized `chat_type` to fetch the correct template files used to render the slackchat.
4. The app renders the template using the serialized slackchat as context.
5. The rendered slackchat is published to the S3 bucket at the location indicated by the serialized `publish_path`.

CHAPTER 4

Developing templates

As long as your template files follow a few naming conventions, slackchat-renderer will be able to find them and use them to render a serialized slackchat.

Create your `ChatType` instance in slackchat-serializer with a name which is a lowercase slug. You'll then use that slug to create the files that makeup your template.

By default, the render will look for each of these four template files when rendering a slackchat:

```
chatrender/
  templates/
    chatrender/
      <slug>/
        index.html
  static/
    chatrender/
      css/
        main-<slug>.css
      js/
        main-<slug>.js
        main-<slug>.js.map
```

This app includes our webpack-based bundler to help compile your static files, so if you're developing template static files in `staticapp` directory, you might organize your directory like this:

```
staticapp/
  src/
    js/
      main-<slug>.jsx
      <slug>/
        component.jsx
        # etc.
    scss/
      <slug>/
        styles.scss
```

(continues on next page)

(continued from previous page)

```
        _partial.scss
        # etc.
```

CHAPTER 5

Indices and tables

- genindex
- modindex
- search