
Newsify

Jun 19, 2020

1	Scraper	1
1.1	Scraper admin	1
1.2	Scraper models	1
1.3	Scraper serializers	1
1.4	Scraper services	2
1.5	Scraper tests	2
1.6	Scraper views	2
1.7	Scraper commands	2
2	Common	3
2.1	Common models	3
2.2	Common docs	3
3	Indices and tables	5

Scraper is the core app used to scrape the websites, parse the information and save it in its respective models.

1.1 Scraper admin

Make the posts modifiables in the admin

Just add the following code to add a simple model view:

```
from scraper.models import Post

admin.site.register(Post)
```

You can also check the following link for a better approach: <https://docs.djangoproject.com/en/dev/ref/contrib/admin/>

1.2 Scraper models

We are gonna find the Post model in here which is basically every post in the app the goal is to have a new post for every new news on each news platform

post_type is the way we can differentiate between news platforms posts

Every other field is well self-explanatory.

1.3 Scraper serializers

The PostSerializer is responsible to serialize the Post model. By now this gonna return every field.

1.4 Scraper services

We have the `ScraperManager` this is the core of the whole project due to its responsibility to scrape every news platform looking for the most recent post and save it into our DB

1.5 Scraper tests

TODO

1.6 Scraper views

`PostViewSet` will handle our `posts` methods logic, it has all the methods inherited from `ModelViewSet`.

1.7 Scraper commands

`scrape` command is gonna be used to scrape every post on the news platforms using the `ScraperManager` is gonna be used as a cronjob

Common app keeps every common folder and models

2.1 Common models

`BaseModel` is the base model, used for every other model it adds a `created_at` field to every model and change the way we set a unique identifier to every model instance

2.2 Common docs

Common also holds the documentation stuff, we use [ReDoc](#) for the `api` documentation and [Sphinx](#) for the backend app documentation.

- **ReDoc**
 - We used the [Django rest framework](#) example given to create the minimal `ReDoc` documentation.
 - We also use [this guide](#) to generate the OpenAPI schema.

CHAPTER 3

Indices and tables

- `genindex`
- `modindex`
- `search`