

2021
djangocon.eu



blackfire.io

Build, deploy and scale: Django, GraphQL and SPA

@dhilipsiva

2021
djangocon.eu

@dhilipsiva



blackfire.io

- Optimistic Nihilist
- Democratic Socialist
- I code for fun & profit
- I love Science, Python, FOSS & தமிழ்
- Dad of 2. Environmentalist. Story Teller. Gamer.
- Jack of all trades & Master of none
- A volunteer for progressives

2021
djangocon.eu



blackfire.io

I have no idea what I am talking about



- 11+ years of professional experience in python
- Built over 20 different products using django alone
- Scaled more than 5 products to serve millions of requests per second
- Built a Device Farm (iOS and Android)
- Involved in building a Bitcoin mining farm with Rpi + ASIC (aws for miners)

2021
djangocon.eu



blackfire.io

Profession

VP of Engineering @ Reckonsys

Hitoshikage @ Nitimis

Why GraphQL?



- Much more standardized than the fragmented RESTful implementations
- Out-of-box live & browsable documentation
- No back-and-forth communication between API developer and SPA developer
- Precise queries



Let's look at our (buggy) demo app
<https://stackopenflow.space>

It is a very silly StackOverflow
clone



- Python
- Django
- Graphene
- Django-graphql-jwt
- Django-rules
- postgres

- JavaScript
- React
- Relay
- easy-peasy

Graphene

- Custom field defs
- Synchronous
- Mature
- Status Quo
- Fantastic Django integration

Strawberry.rocks

- Dataclasses
- Asynchronous
- Not-so-mature
- Future (maybe)
- Very minimal Django integration

2021
djangocon.eu

Per-object perms



blackfire.io

- Choose a rule based framework (like django-rules) over a DB-based framework (guardian)
- Less DB calls = less latency
- No unnecessary migrations due to change in logic
- Easier to maintain

Why React?



- Plays really well with GraphQL (Facebook being a big proponent of both React and GraphQL)
- Can build native applications with react-native. So a front-end engineer can easily scale-up as a mobile application developer.
- Second only to vue.js in terms of community/popularity

Why Relay?



- Relay's APIs are cleaner compared to Apollo
- Hooks are 1st class citizens
- Out of box caching support
- Backed by Facebook (just like GraphQL and React) so React has excellent integration.
- Relay-compiler is fantastic at catching errors

2021 djangocon.eu Why easy-peasy?



blackfire.io

- Hooks are 1st class citizens
- Less boilerplate
- Pleasant APIs
- Ease of use

2021 djangocon.eu Project Layout



blackfire.io

- All other django apps are placed under the django root “stackopenflow” app (the module that contains settings.py, urls.py, wsgi.py, etc)
- Therefore, all imports are namespaced with “stackopenflow”

2021 djangocon.eu Typical Modules



blackfire.io

- `admin.py`
- `apps.py`
- `models.py`
- `tests.py`
- `views.py`



- `admin.py`
- `apps.py`
- `behaviours.py`
- `choices.py`
- `inputs.py`
- `models.py`
- `mutations.py`
- `queries.py`
- `tests.py`
- `types.py`

Let's walk through the source code Django & React

<https://github.com/djcone2021/stackopenflow-api>
<https://github.com/djcone2021/stackopenflow-app>

- <https://github.com/djcone2021/bigga>
- A docker-compose based deployment
- Can be configured to be used to docker-swarm with minimal changes
- Handy for local development too
- Suitable for cheap, non-production workloads
- Code walk through



- <https://aws.github.io/copilot-cli>
- ECS on AWS Fargate (serverless containers)
- Very reliable & scaleable, but vendor dependent
- Easy auto scaling configuration
- Cheaper & faster than Lambda (serverless functions)
- But be sure set-up along with tools like Terraform

2021 djangocon.eu SPA Deployment



blackfire.io

- S3 + CloudFront + ACM
- Cheaper than alternatives like Netlify
- Less latency
- Manual Setup:

https://github.com/reckonsys/bigga/blob/community/S3_FRONTEND_DEPLOYMENT.md

- Pyinvoke Script:

<https://github.com/reckonsys/bigga/blob/community/tasks.py>

2021
djangocon.eu

Thank You



blackfire.io

@dhilipsiva

dhilipsiva@pm.me

<https://dhilipsiva.com>

<https://t.me/dhilipsiva>