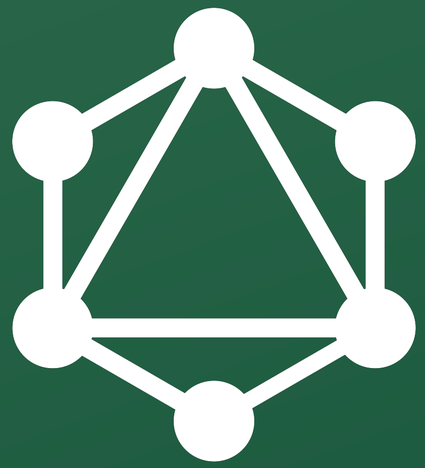


Goodbye REST

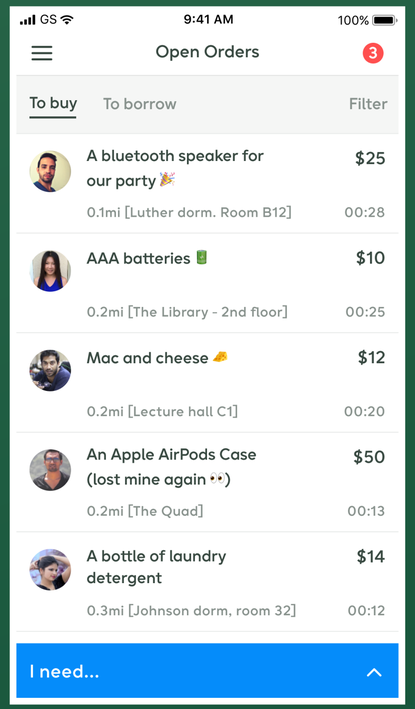
Building APIs with Django and GraphQL

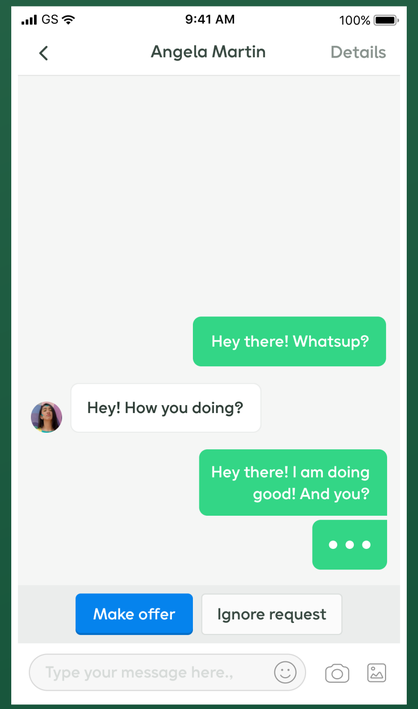
Jayden Windle

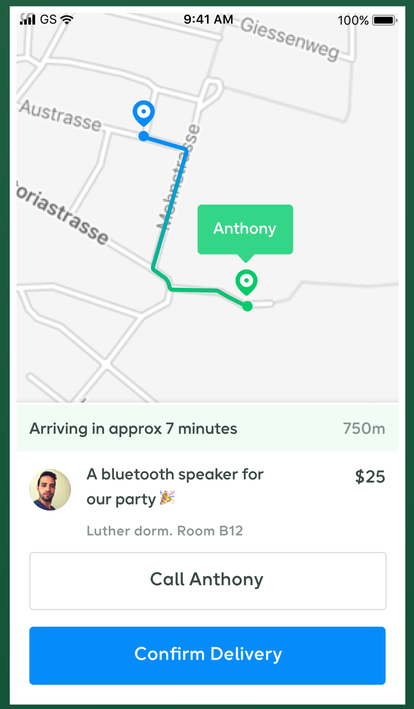
Lead engineer at Jetpack

@jaydenwindle

JETPACK


GS
9:41 AM
100%
Open Orders
3
To buy
To borrow
Filter
A bluetooth speaker for our party 🎉
$25
0.1mi [Luther dorm. Room B12]
00:28
AAA batteries 🔋
$10
0.2mi [The Library - 2nd floor]
00:25
Mac and cheese 🧀
$12
0.2mi [Lecture hall C1]
00:20
An Apple AirPods Case (lost mine again 👀)
$50
0.2mi [The Quad]
00:13
A bottle of laundry detergent
$14
0.3mi [Johnson dorm, room 32]
00:12
I need...


GS
9:41 AM
100%
Angela Martin
Details
Hey there! Whatsup?
Hey! How you doing?
Hey there! I am doing good! And you?
Make offer
Ignore request
Type your message here..


GS
9:41 AM
100%
Giessenweg
Austrasse
Anthony
Arriving in approx 7 minutes
750m
A bluetooth speaker for our party 🎉
$25
Luther dorm. Room B12
Call Anthony
Confirm Delivery

Disclaimer

Talk Outline

1. GraphQL 101 - key concepts
2. Build a simple GraphQL API
3. Pros / Cons
4. What's Next

What is GraphQL?

GraphQL is a query language, specification, and collection of tools, designed to operate over a single endpoint via HTTP, optimizing for performance and flexibility.

Source: https://philsturgeon.uk/api/2017/01/24/graphql-vs-rest-overview/

Key Features

- Query for only the data you need
- Easily query for multiple resources in a single request
- Great front end tooling for handling caching, loading / error states, and updates.

GraphQL Schema

- Types
- Queries
- Mutations
- *Subscriptions

Graphene

An Example App

```python
class Todo(models.Model):

    user = models.ForeignKey(User, on_delete=models.CASCADE)
    title = models.CharField(max_length=500)
    done = models.BooleanField(default=False)
    created_at = models.DateTimeField(auto_now_add=True)
```

```python
# example/settings.py
INSTALLED_APPS = [
    ...
    'graphene_django',
    'todo',
]

# example/urls.py
from graphene_django.views import GraphQLView

from example.schema import schema

urlpatterns = [
    ...
    path('graphql/', GraphQLView.as_view(graphiql=True, schema=schema)),
]

# example/schema.py
import graphene

class RootQuery(graphene.ObjectType):
    pass

schema = graphene.Schema(query=RootQuery)
```

Types

```python
from graphene_django.types import DjangoObjectType
from todo.models import Todo


class TodoType(DjangoObjectType):
    class Meta:
        model = Todo
```

Queries

```
query {
  allTodos {
    id
    title
    done
  }
}
```

```
{
  "data": {
    "allTodos": [
      {
        "id": "1",
        "title": "Finish Django Meetup Talk",
        "done": true
      },
    ]
  }
}
```

Mutations

```
mutation {
  createTodo(title:"Be awesome", done:true) {
    todo {
      id
      title
      done
      createdAt
    }
  }
}
```

Mutations

```
{
  "data": {
    "createTodo": {
      "todo": {
        "id": "6",
        "title": "Be awesome",
        "done": true,
        "createdAt": "2018-11-28T21:58:16.387141+00:00"
      }
    }
  }
}
```

Subscriptions

```
subscription {
  todoUpdated {
    todo {
      id
      title
      done
    }
  }
}
```

```
{
  "data": {
    "todoUpdated": {
      "todo": {
        "id": "6",
        "title": "Be awesome",
        "done": true,
      }
    }
  }
}
```

Subscriptions

No plug and play support yet 😢

GH Issue (#430)

Search or jump to... / Pull requests Issues Marketplace Explore

graphql-python / graphene

Watch 151 Star 3,905 Fork 413

Code Issues 87 Pull requests 15 Projects 0 Wiki Insights

Graphene real-time subscriptions and Apollo client graphql integration #430

New issue

Open **hballard** opened this issue on Mar 3, 2017 · 65 comments

hballard commented on Mar 3, 2017 • edited

Hello **@syrusakbary**.

Thanks for all your hard work on graphene and graphql-python. Awesome library!!

I posted this on #393 earlier this week...reposting here so it's easier to discover.

I implemented a port of the apollo graphql subscriptions modules (graphql-subscriptions and subscriptions-transport-ws) for graphene / python. They work w/ apollo-client.

It is here.

Same basic api as the Apollo modules. It is still very rough...but works so far, based on my limited internal testing. Uses redis-py, gevent-websockets, and syrusakbary/promises. I was going to add a simple example app, setup.py for easier install, and more info to the readme w/ the API, in the next few days. A brief example is below. Only works on python2 for now. My plan is to start working on tests as well. I figured I'd go ahead and share in this early stage in case anybody is interested...

I'm very new to open source, so any critiques or pull requests are welcome.

Assignees
No one assigned

Labels
None yet

Projects
None yet

Milestone
No milestone

Notifications
Subscribe
You're not receiving notifications from this thread.

Other Realtime Options

- graphql-python/graphql-ws
- eamigo86/graphene-django-subscriptions
- tricoder42's Implementation Gist
- Hybrid Websocket / GraphQL approach
- Polling / Refetch

File Uploads

lmcgartland/graphene-file-upload

Search or jump to... / Pull requests Issues Marketplace Explore

lmcgartland / graphene-file-upload

Watch 4 Star 59 Fork 5

Code Issues 3 Pull requests 0 Projects 0 Wiki Insights

Enhances Graphene Django GraphQL Server for intuitive file uploads via GraphQL mutations.

graphql graphene django

73 commits 1 branch 4 releases 5 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

lmcgartland Merge pull request #27 from davidroeca/linting ... Latest commit 326071d 9 days ago

examples	Added graphene-django requirement to example	5 months ago
graphene_file_upload	Lint files and resolve django issue	9 days ago
tests	use current fire emoji	3 months ago
.gitignore	removed merge conflicts	2 months ago
.pylintrc	Lint files and resolve django issue	9 days ago
.travis.yml	removed python 3.3 and flake8 tests	2 months ago
LICENSE	Initial Commit	8 months ago
MANIFEST.in	Initial Commit	8 months ago
README.md	add --develop flag to docs since it was removed from tox	19 days ago
README.rst	clarify how to recreate tox	3 months ago
requirements-tox.txt	add first test	3 months ago
setup.cfg	Initial Commit	8 months ago
setup.py	bump version	9 days ago
tox.ini	Lint files and resolve django issue	9 days ago

Extra Utilities

eamigo86/graphene-django-extras

Search or jump to... / Pull requests Issues Marketplace Explore

eamigo86 / graphene-django-extras

Watch 22 Star 145 Fork 37

Code Issues 21 Pull requests 1 Projects 0 Wiki Insights

Extras functionalities for Graphene-Django

graphene-django-extras django graphene-django graphql python

197 commits 1 branch 27 releases 17 contributors MIT

Branch: master New pull request Create new file Upload files Find file Clone or download

eamigo86 Deleted unused iso8601 package import to avoid error after update to ... Latest commit 20238fd 3 days ago

graphene_django_extras	Deleted unused iso8601 package import to avoid error after update to ...	3 days ago
testproject	Merge remote-tracking branch 'upstream/master'	4 months ago
.gitignore	Merge remote-tracking branch 'upstream/master'	4 months ago
LICENSE	Fixed License filename	a year ago
MANIFEST.in	Add Licence file, and update .md and .rst files	a year ago
README.md	Deleted unused iso8601 package import to avoid error after update to ...	3 months ago
README.rst	Deleted unused iso8601 package import to avoid error after update to ...	3 months ago
setup.cfg	Add files via upload	a year ago
setup.py	Deleted unused iso8601 package import to avoid error after update to ...	3 days ago

README.md

Graphene-Django-Extras pypi package 0.3.8

Pros

- Dramatically improves front end DX
- Flexible types allow for quick iteration
- Always up to date documentation
- Only sends needed data over the wire

Cons

- Graphene isn’t as mature as some other GraphQL implementations (JS, Ruby)
- Logging is different when using a single GraphQL endpoint
- REST is currently better at server-side caching*

What's Next?

- Graphene 3 Roadmap - great opportunity to get involved
- We're working on some cool projects internally related to Subscriptions and Apollo Engine support - will open source soon!

Questions?

https://github.com/jaydenwindle/goodbye-rest-talk

@jaydenwindle

An Example App

```
from rest_framework import routers

from todo.rest import views

router = routers.DefaultRouter()
router.register(r'todos', views.TodoViewSet)

urlpatterns = [
    path('admin/', admin.site.urls),
    path('api/', include(router.urls)),
]
```

An Example App

```
GET     /todos       => Returns all Todo items
GET     /todos/:id   => Returns single Todo item
POST    /todos       => Creates new Todo item
PUT     /todos/:id   => Updates a Todo item
DELETE  /todos/:id   => Deletes a Todo item
```

Queries

```
// GET /todos/

[
    {
        "id": 1,
        "title": "Some task",
        "done": false,
        "created_at": "2018-11-28T16:35:17.221721Z",
        "user": 1
    },
]
```

REST API problems we ran into

- Handling data on the front end was verbose
- Composing multiple resources is complex
- Keeping documentation up to date is hard

How GraphQL Solves them

- GraphQL has great front end tooling which handles lots of things out of the box (caching, error and loading states)
- GraphQL makes it easy to compose queries and fetch multiple types at once
- The interactive playground makes documentation easy to keep track of

GraphQL in Production

- We've been running GraphQL in production for ~3 months
- Haven't run into any performance bottlenecks vs our REST setup
- ~26% decrease in LOC that interact with our API

Queries

```
query {
    hello
}
```

```
"data": {
    "hello": "world",
}
```

Queries

```python
import graphene


class HelloWorldQuery(graphene.ObjectType):
    hello = graphene.String()

    def resolve_hello(self, info, **kwargs):
        return "World!"
```

Queries

```
/graphql?query=query%20%7B%0A%20%20hello%0A%7D
```