

DJANGO REST FRAMEWORK

BUILDING BEAUTIFUL APIS

[Derek Stegeman](#) / [@derekstegelman](#)

DEREK STEGELMAN

Sr. Web Application Developer at Kansas State University

Python/Django for 5 years

[GitHub](#)

GOALS

- Understand the what and why of Web APIs
- Basics of REST
- Basics of building APIs with Django and DRF

WHAT IS AN API?

API

- Application Programming Interface
- Allows communication between components and systems.
- A web based API is simply one example of an API.

WHY DO I NEED AN API?

- Allow other websites/apps to interact with your data
- Build "microsites" with tools like AngularJS (front end frameworks)
- Build apps for Phones or other devices for your core product
- Web becomes a platform like Desktop, Phone, etc

WHAT EXACTLY IS A RESTFUL API?

REPRESENTATIONAL STATE TRANSFER (REST)

- High performance, scalable web services
- Focus on HTTP verbs
- Using the web as intended

HTTP VERBS/CRUD

Create: POST

Read: GET

Update: PUT or PATCH

Delete: DELETE

OKAY, SO...

WHAT'S SO AWESOME ABOUT REST?

RESTFUL APIS

It's simple, its just HTTP

Easy integration, just make a HTTP request

Easily predictable

Easy standard to get behind, has wide adoption

**CONCLUSION: THESE ARE BUILT INTO THE WEB,
LET'S USE THEM!**

LET'S BUILD AN API

(WITH DJANGO)

LOTS OF FRAMEWORKS TO CHOOSE FROM

- Piston (originally written for Bitbucket)
- Tastypie (old standard, good solid development, lacks easy customization)
- Django Rest Framework (accepted standard way in Django, easy to use, browseable API, Kickstarter funding)

DJANGO REST FRAMEWORK

Django REST framework is a powerful and flexible toolkit that makes it easy to build Web APIs.

WHY USE IT?

DJANGO REST FRAMEWORK

- Browseable API
- Authentication and Authorization Policies
- Serialization that supports both ORM and non-ORM data sources.
- Customizable/Extensible
- Extensive documentation, and great community support.
- Used and trusted by large companies such as Mozilla and Eventbrite.

BROWSEABLE API

WHAT AND WHY?

- A browseable API allows users to view and make calls to your API through a web interface
- Gathers buy in for your product
- Makes life easier for those consuming the API, and cuts down on support time for developers

BROWSEABLE API DEMO

AUTHENTICATION AND AUTHORIZATION

AUTHENTICATION

- Built in support for Token, Basic and Session.
- Community packages for Oauth, Oauth2, JSON Web Token, others

AUTHORIZATION

- Built in ORM based permissions (actions on Django models)
- Easy to build in custom permissions per view, etc

SERIALIZATION

- Built in serialization of Django/ORM objects
- Easy customization and extension of serializers

SERIALIZERS

```
class AreaSerializer(serializers.ModelSerializer):
    campus_organization = serializers.SerializerMethodField()

    def get_campus_organization(self, obj):
        return obj.student_organization

    class Meta:
        model = Area
        fields = ('name', 'slug', 'url', 'campus_organization')
```


VIEWS/DEMO/LIVE CODE

THANKS

QUESTIONS?