Bootcamp Training

Python

SummitWorks™
GLOBAL SOLUTION ARCHITECTS

Authorized & published by Summitworks Technologies Inc

Agenda

- web services – Restful API using DRF

SPA

- Modern web applications are increasingly built as Single-Page-Applications (SPAs) which feature a distinct frontend and backend.
- As opposed to the traditional monolith approach in Django itself–and other web framework like Ruby on Rails–SPAs require a backend API that can then be consumed by multiple frontends as needed.
- This approach works well when a company needs multiple frontend applications–mobile web, iOS, Android–to run off the same database.

Restful API

- An API (Application Programming Interface) provides an interface for developers to interact with an application's database. Instead of just giving someone full access to a database, an API sets up rules, permissions, and endpoints for various functionality: login, logout, reading a list of blogs, individual blog details, and so on.
- The traditional way to construct a web API is via REST (Representational State Transfer), a well-established architecture for how websites can communicate with one another. Since computers communicate via the web this means using the HTTP protocol which supports a number of common "methods" (also called "verbs") such as GET, PUT, POST, and DELETE.
- There are also a host of related access codes that indicate whether a request was successful (200), redirected (301), missing (404), or worse (500).

JSON

- It’s important to note that since an API is communicating with another computer the information being shared is not what would be sent for a standard web page. When your browser requests, for example, the Google homepage, it sends HTTP requests and receives HTTP responses with HTML, CSS, JavaScript, images, and so on.

- An API is different. Typically we’re only interested in the data from a database. This data is often transformed into JSON format to be efficiently transmitted about. The API will also have a series of well-defined rules for how a frontend client can interact with it via a REST architecture. To register a new user the frontend framework will need access an API endpoint called, for example, /api/register. This API endpoint contains both a specific URL route and its own set of permissions.

Setup - Django Rest Framework

- DRF takes care of the heavy lifting of transforming our database models into a RESTful API. There are two main steps to this process:
  - first a serializer is used to transform the data into JSON so it can be sent over the internet, then a View is used to define what data is sent.

Pip install djangorestframework

Setup - Django Rest Framework

```
INSTALLED_APPS = [
    'django.contrib.admin',
    'django.contrib.auth',
    'django.contrib.contenttypes',
    'django.contrib.sessions',
    'django.contrib.messages',
    'django.contrib.staticfiles',

    'rest_framework',
]
```

Setup - serializers

he serializer is used to convert our data into JSON format. That’s it

Setup – views

Just as Django has generic class based views, so too DRF has generic views we can use

Setup – endpoints

The final piece is urls. We need to create the url routes–known as endpoints in an API–where the data is available.

Any Queries