DJANGO REST BASICS

**Why is REST API USED ?**

REST or RESTful APIs were designed to take advantage of existing protocols.

While REST - or Representational State Transfer - can be used over nearly any protocol, when used for web APIs it typically takes advantage of HTTP.

One of the key advantages of REST APIs is that they provide a great deal of flexibility.

REST APIs let’s us send info back and forth between and interface and database.

What Is Django Rest Framework ?

Django Rest Framework (DRF) is a library which works with standard Django models to build a flexible and powerful API for your project.

To Be Explained Precisely :

DRF let’s us create RESTful API’s

A way to transfer information between an interface and a database in a simple way

We know now that REST APIs are important because they let us interact in a easy way with the database and we can use it with different platforms

But Why DRF and not Node.js,Golang,Rust etc.. ?

Well for starters DRF uses python and easy to learn

DRF ,as much as Django makes everything simple and easier

Example :

If we want an end-point(an HTTP address) that returns a list with each user

class UserList(generics.ListCreateAPIView):

queryset = User.objects.all()

serializer\_class = UserSerializer

That’s it . We created a class called User List that inherits methods from ListCreateAPIView.

As the name says ,we can list and create users.

If we ask for the URL(GET), you get the list of the users,

If we send (POST) information to that URL it will create a new user

3 lines of code.That’s all.

### Basic Architecture

A DRF API is composed of 3 layers: the serializer, the viewset, and the router.

* Serializer:
  + Converts the information stored in the database and defined by the Django Models into a format which is more easily transmitted via an API
* View-set:
  + Defines the functions (read, create, update, delete) which will be available via the API
* Router:
  + Defines the URLs which will provide access to each viewset