Ex.No:6 Develop a web service using Django

Reading Materials:

REST (Representational State Transfer) is a standard architecture for building and communicating with web services. It typically mandates resources on the web are represented in a text format (like JSON, HTML, or XML) and can be accessed or modified by a predetermined set of operations. Given that we typically build REST APIs to leverage with HTTP instead of other protocols, these operations correspond to HTTP methods like GET, POST, or PUT.

An API (Application Programming Interface), as the name suggests, is an interface that defines the interaction between different software components. Web APIs define what requests can be made to a component (for example, an endpoint to get a list of shopping cart items), how to make them (for example, a GET request), and their expected responses.

Set up a project

First, create a structure for Django application;

```
$ mkdir myproject
$ cd myproject
```

Then, create a virtual environment to isolate package dependencies locally within the project directory:

```
$ python3 -m venv env
$ source env/bin/activate
```

On Windows, use the command env\Scripts\activate to activate Python virtual environment.

Install Django and the Django REST framework

Next, install the Django REST:

```
$ pip3 install djangorestframework
```

Instantiate a new Django project

```
$ django-admin startproject training .
$ cd training
$ django-admin startapp quickstart
```

Ex.No:6 **Develop a web service using Django**

Date : 27/09/24

Aim :-

To durlop a web dervice for Idea Forum Application

Procedure :-

1. Create a Function in Views py.file

```
from Django.views import view
from Django.http import json Response
import json
from.modules import sports
class IdeaForum(View):
def port(self.requests)
data = json.loads(requests.body.decode("utf-8"))
d_name = data.get(`sector_name')
d_ID = data.get('sector_likes')
sector_data={
 `sector_name'= d name,
 'sector_ID'= d-ID,
 'sector_likes' = d likes;}
```

```
sector_details = sectordetails .objects.create (* * sector
                                                 details)

data = {
"message" = { " New sector added to the sectors
                with id{ sector_details .id }"

        }

return   Sector Response (data . status = 201)
```

Step 2: In urls.py in IdeaForum directory,

```
from django .urls import path
from .views import IdeaForum
url pattens = [
path (" sector_details /", IdeaForum .as.view ()),
]
```

Step 3:- Run the application by utilizing mange.py file

Step 4: Enter the command

```
python manage.py runserver
```

Step 5: Output will be displayed

OUTPUT:-

```
Sectors

sector_name   :  Sports
sector_ID     :  5002
sector_ like  :  1286
```