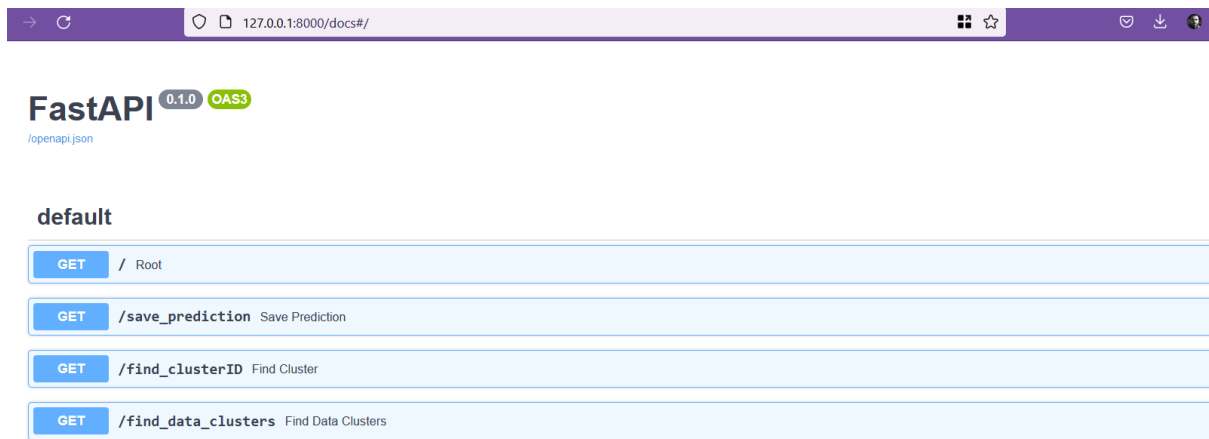
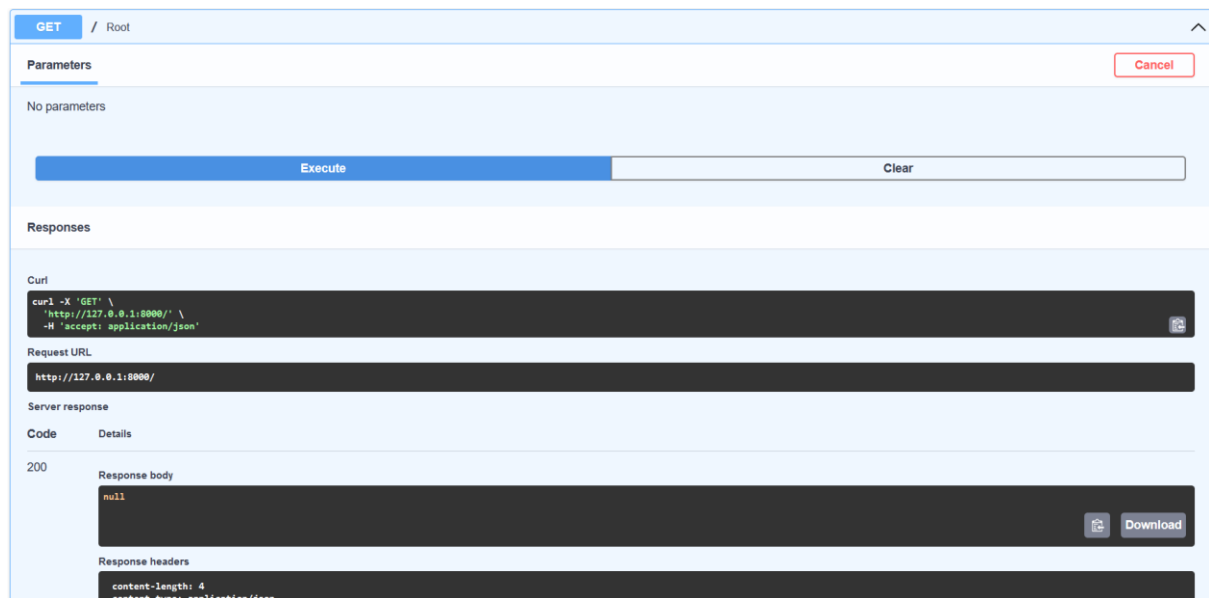


Front page:



Step 1: To generate coordinates and to cluster it



Snapshot of database after above step:

userClick.coordinates

5001

DOCUMENTSINDEXES

Documents

Aggregations

Schema

Explain Plan

Indexes

Validation

Filter

Type a query: { field: 'value' }

Reset

Find

More Options

ADD DATA

EXPORT COLLECTION

1 - 20 of 500

_id: ObjectId('63bb144e884376d666a94909')

x: 0.78

y: 0.01

_id: ObjectId('63bb144e884376d666a9490a')

x: 0.23

y: 0.96

_id: ObjectId('63bb144e884376d666a9490b')

x: 0.48

y: 0.56

_id: ObjectId('63bb144e884376d666a9490c')

x: 0.98

y: 0.97

_id: ObjectId('63bb144e884376d666a9490d')

x: 0.17

y: 0.81

Step 2: Saving the predicted cluster of input coordinates in the database

127.0.0.1:8000/docs#/default/save_prediction_save_prediction_get

GET /save_prediction Save Prediction

Parameters

Name	Description
x_coord * required (query)	<input type="text" value="0.78"/>
y_coord * required (query)	<input type="text" value="0.01"/>

Execute

Clear

Responses

Curl

curl -X 'GET' \ 'http://127.0.0.1:8000/save_prediction?x_coord=0.78&y_coord=0.01' \ -H 'accept: application/json'

Request URL

http://127.0.0.1:8000/save_prediction?x_coord=0.78&y_coord=0.01

Server response

Code	Details
200	<div><div>Response body</div><div>null</div></div>

Step 3: Finding the cluster ID of the given coordinates:

The screenshot shows a REST client interface with the following details:

- URL:** `127.0.0.1:8000/docs#/default/find_cluster_find_clusterID_get`
- Method:** GET
- Path:** `/find_clusterID`
- Parameters:**
 - Name:** `x_coord`, **Description:** (query), **Value:** `0.78`
 - Name:** `y_coord`, **Description:** (query), **Value:** `0.01`
- Buttons:** Execute, Clear
- Responses:**
 - Curl:**

```
curl -X 'GET' \
'http://127.0.0.1:8000/find_clusterID?x_coord=0.78&y_coord=0.01' \
-H 'accept: application/json'
```
 - Request URL:** `http://127.0.0.1:8000/find_clusterID?x_coord=0.78&y_coord=0.01`
 - Server response:**

Code	Details
200	<div>Response body</div> <pre>{}</pre>

Step 4: Find all the coordinates and it's cluster ID:

The screenshot shows a REST client interface with the following details:

- URL:** `127.0.0.1:8000/docs#/default/find_data_clusters_find_data_clusters_get`
- Method:** GET
- Path:** `/find_data_clusters`
- Parameters:** No parameters
- Buttons:** Execute, Clear
- Responses:**
 - Curl:**

```
curl -X 'GET' \
'http://127.0.0.1:8000/find_data_clusters' \
-H 'accept: application/json'
```
 - Request URL:** `http://127.0.0.1:8000/find_data_clusters`
 - Server response:**

Code	Details
200	<div>Response body</div> <pre>{ "x": { "0": 0.78, "1": 0.23, "2": 0.48, "3": 0.98, "4": 0.17, "5": 0.82, "6": 0.84, "7": 0.94, "8": 0.95, </pre>