**Documentation of ICD-1O Codes Endpoint**

BACKEND LANGAUGE: Python

RDBMS: MYSQL

Framework: Flask

Deployment Container: Docker

ENDPOINT= <http://127.0.0.1:5000/>

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| ENDPOINST | REQUEST METHODS | PAYLOADS | RESPONSE CODE | PAYLOADS FORMAT | ACTION |
| **/api/v1/list** | GET | NONE | 200 | NONE | LIST ALL RECORDS |
| **/api/v1/create** | POST | PAYLOADS | 201 | JSON OBJECT | CREATE A NEW RECORD |
| **/api/v1/delete/{id}** | DELETE | NONE | 200 |  | DELETE A RECORD |
| **/api/v1/update/{id}** | PUT | PAYLOADS | 200 | JSON OBJECT | UPDATE A RECORD |
| **/api/v1/list/{id}** | GET | NONE | 200 |  | LIST RECORDS WITH A SPECIFIC ID |
| **/api/v1/list?start={0}&limit={20}** | GET | NONE | 200 |  | LIST RECORD IN A PAGINATED FORM |
|  |  |  |  |  |  |

**ENDPOINT PAYLOADS REQUEST BODY**

ENDPOINT: <http://127.0.0.1:5000/>api/v1/create

{

"category\_code": "A00", "diagnosis\_code": "1235",

"full\_icd\_code": "A001235", "abbreviated\_description": "Comma-ind anal dop",

"full\_description": "Comma-induced anal retention",

"category\_title": "Malignant neoplasm of anus and anal canal",

"icd\_type": "ICD-10”

}

ENDPOINT: <http://127.0.0.1:5000/>api/v1/update /1

{

"category\_code": "A00", "diagnosis\_code": "1235",

"full\_icd\_code": "A001235", "abbreviated\_description": "Comma-ind anal dop",

"full\_description": "Comma-induced anal retention",

"category\_title": "Malignant neoplasm of anus and anal canal",

"icd\_type": "ICD-10”

}

NB:

<http://127.0.0.1:5000/api/v1/delete/1>

This endpoint Flag’s a status to “inactive” without deleting the record or removing the entire record from the row of the schema

INCREMENTATION OF ICD-10, ICD-9 CODES PROBLEM

An Additional schema table was introduce to associate or link each ICD CODE to their respective ICD-10 or ICD-9 code .This method will help to track various code and to support changes.

For Data integrity and consistence ICD\_MIN\_LENE, ICD\_MAX\_LEN are used validate the submitted ICD CODE submitted by user.

e.g

TABLE NAME : icd\_records

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ID(PK) | ICD\_NAME | ICD\_MIN\_LEN | ICD\_MAX\_LEN |  |
| 1 | ICD-10 | 3 | 7 |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Id | Code\_category | Diagnosis\_code | Fullcode | Diagnosis | Icd\_record\_id |  |
| 1 | A00 | 123 | A00123 |  | 1 |  |

Deployment Instruction

Git clone

Local Deployment

1. Git clone repository to local machine

2. Change directory into the Project CD / product repo

4.Set up Local db(MSQL)

5. located a folder in the app directory called db.

6.Import the db.sql using any mysql client

3. Install venn env container to host app in a virtual container: Windows(python –m venv env ) Linux/MAC ( virtualenv venv ) pip install virtualenv

4. Activate the env folder: Windows command :(env\Scripts\Activate) linux/mac/UNIX(venv/bin/activate )

5. Install Flask(Python Frame Work) : pip install Flask

6. Run command python app.py to start the Flask app

7. To run a Test open a CMD /TERMINAL

8. Run command: python test.py to run the test.

9.Test the Endpoints with Postman