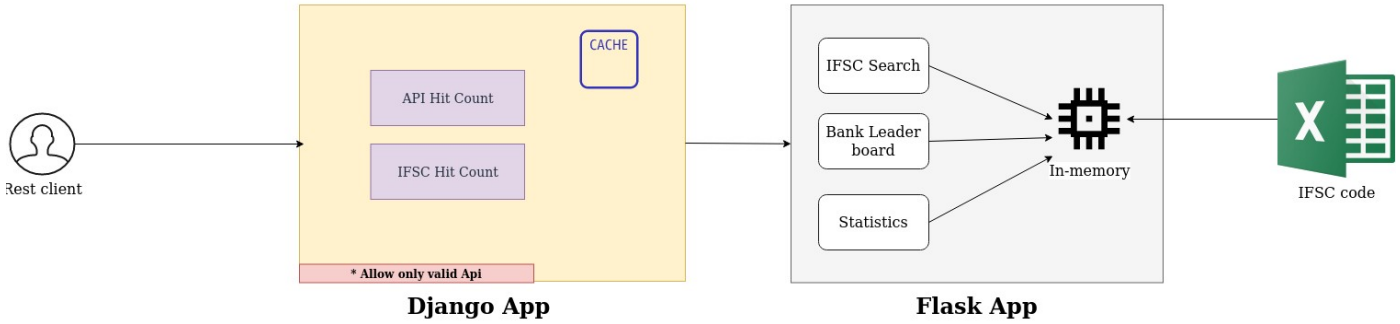
Group Data & Analytics – Backend Assignment

Scope:

Build 2 Web apps - Django and Flask based, where Flask will hold data IFSC data in-memory. We need to invoke the APIs from the REST Client (postman or CURL).

If you comfortable or familiar with Java, you can opt for Spring Boot. If any other server side scripting / language is chosen, then ensure that we have 2 different applications with REST interfacing as shown in the diagram below.

Block Diagram



Flask App

1. On start : load all the IFSC Data into the memory (Data Structure - Dict<IFSC, IFSCObject>) from the excel file.
2. IFSC Object - Create a class that has the following attributes (BANK,IFSC,MICR CODE,BRANCH,ADDRESS,STD CODE,CITY,DISTRICT,STATE)
3. Create another Data Structure [Bank LeaderBoard] - Dict <BANK, count_of_bank> and compute from the base data and load.
4. Create another Data Structure [Statistics] - List <IFSC, Timestamp> to hold the search history (only valid ifsc code to enter this list, and the timestamp of the request>
5. API Endpoints (return as JSON)

```
- IFSC Search :: input parameter: ifsc_code (mandatory)
        output -> object, handle null, empty, invalid data etc; ifsc not found, return 404 header
- Bank LeaderBoard :: input parameter -> (NONE); output -> Top 10 descending
  :: input parameter -> sortorder (DESC|ASC); default value DESC
  :: input parameter -> fetchcount (1..225); default value - 10
- Statistics :: input parameter -> (NONE); output -> ALL in the list ASC
  :: input parameter -> sortorder (DESC|ASC); default value ASC
  :: input parameter -> fetchcount (1..10000); default value - ALL (entire list)
```

Django App

1. Pass on all the valid requests to the flask instance; invalid requests should be filtered here
2. Create a Data Structure [Dict<IFSC, RESULT_JSON>] to cache the IFSC Search in this layer. If the

data is not found in the DS then request should be send to Flask and then the result in cached here and returned to the client.

3. Create a Data Structure [IFSC Hit Count] - Dict <IFSC, count_of_hits>. This is specific to IFSC API hit count by at the IFSC level.
4. Create a Data Structure [API Hit Count] - Dict <API_URL, count_of_hits>

Dependency

IFSC File – Google Drive Link
https://docs.google.com/spreadsheets/d/1149Ps2PCafaKjHHkUYw5PmeiY5skg3tY/edit?usp=sharing&ouid=105581454063004875005&rtpof=true&sd=true

Deliverables

- Code as zip archive or github url with README (include Python tests - unit tests)
- Documentation - setup virtual python env and start the servers
- Postman Collection / Curl URLs collection
- Deployment Diagram – AWS / Azure / OCI
- Document / Steps on Securing the applications on AWS, not generic one, with respect to the deployment diagram.