**Alter Office Analytics - Documentation**

**Table of Contents**

1. Introduction
2. Prerequisites
3. Installation and Setup
4. Uploading Data to MongoDB
5. Running the FastAPI Server
6. API Endpoints
7. Data Processing and Segmentation
8. Conclusion

**1. Introduction**

This document describes the process of setting up and running a marketing analytics system using MongoDB and FastAPI. The system ingests user data from CSV files, processes unique user records, segments repeated email IDs into a separate cohort collection, and provides API endpoints to access this data.

**2. Prerequisites**

Ensure the following dependencies are installed before proceeding:

* Python 3.8+
* MongoDB (Installed and Running)
* FastAPI
* Uvicorn
* Pandas
* Pymongo

**3. Installation and Setup**

**Step 1: Install Required Libraries**

Run the following command to install all necessary dependencies:

pip install fastapi uvicorn pandas pymongo

**Step 2: Start MongoDB**

Ensure MongoDB is running on your system:

mongod --dbpath /path/to/mongodb/data

**4. Uploading Data to MongoDB**

**Step 1: Create upload\_csv.py**

Create a script upload\_csv.py with the following content to ingest CSV data into MongoDB:

import pandas as pd

from pymongo import MongoClient

from datetime import datetime

# MongoDB Connection

MONGO\_URI = "mongodb://localhost:27017/"

client = MongoClient(MONGO\_URI)

db = client["user\_database"]

users\_collection = db["users"]

unique\_collection = db["unique"]

cohort\_collection = db["cohort"]

# Function to Insert CSV Data into 'users' Collection

def insert\_csv\_to\_users(csv\_path):

df = pd.read\_csv(csv\_path)

records = df.to\_dict(orient="records")

formatted\_records = []

for record in records:

if "created\_at" in record and pd.notna(record["created\_at"]):

record["created\_at"] = datetime.strptime(str(record["created\_at"]), "%m/%d/%Y %H:%M")

if "interests" in record and pd.notna(record["interests"]):

record["interests"] = [interest.strip() for interest in record["interests"].split("|")]

formatted\_record = {"data": record}

formatted\_records.append(formatted\_record)

users\_collection.insert\_many(formatted\_records)

print("✅ Data inserted into 'users' collection")

if \_\_name\_\_ == "\_\_main\_\_":

csv\_path = "D:/alter\_office/sample\_user\_data.csv"

insert\_csv\_to\_users(csv\_path)

**Step 2: Run the script**

python upload\_csv.py

**5. Running the FastAPI Server**

Create server.py and add the following:

from fastapi import FastAPI, HTTPException, Query

from pymongo import MongoClient

from datetime import datetime

from typing import Optional, List, Dict

import math

app = FastAPI()

# MongoDB Connection

MONGO\_URI = "mongodb://localhost:27017/"

client = MongoClient(MONGO\_URI)

db = client["user\_database"]

users\_collection = db["users"]

unique\_collection = db["unique"]

cohort\_collection = db["cohort"]

@app.get("/api/health")

def health\_check():

return {"status": "OK"}

if \_\_name\_\_ == "\_\_main\_\_":

import uvicorn

uvicorn.run(app, host="0.0.0.0", port=8000)

**Start the Server**

uvicorn server:app –reload

**6. API Endpoints**

**1. Insert Data via API**

* **URL:** POST /api/ingest
* **Request Body:**

{

"data": {

"cookie": "cookie\_id",

"email": "user@example.com",

"phone\_number": "1234567890",

"created\_at": "03/14/2025 12:00",

"location": {"country": "India", "state": "Tamil Nadu", "city": "Chennai"},

"demographics": {"age": 30, "gender": "Male"},

"interests": ["Gaming", "Music"]

}

}

**2. Fetch User Data**

* **URL:** GET /api/user?cookie=cookie\_id
* **Response:** User details from the unique collection.

**3. Fetch Cohort Data**

* **URL:** GET /api/cohort/user?cookie=cookie\_id
* **Response:** Cohort user details.

**7. Data Processing and Segmentation**

* **Raw data** is first stored in the users collection.
* **Unique users** are identified based on cookie and stored in unique.
* **Duplicate email IDs** are stored separately in cohort for segmentation.
* **Real-time data updates** ensure accurate segmentation.

**8. Conclusion**

This documentation provides the full process of installing dependencies, uploading data, running the FastAPI server, and accessing user data through APIs. This ensures real-time marketing analytics using MongoDB.