**U-ASK Project Setup Guide**

**Project Overview**

U-ASK is a spatial-textual query processing system that supports POWER and Boolean Range algorithms for retrieving geospatial data from PostgreSQL and OpenSearch. This guide walks you through setting up the project from scratch, including database setup, indexing, and running the query algorithms.

**System Requirements**

Ensure your system meets the following requirements before proceeding:

* **Operating System:** Windows/Linux/MacOS
* **Python Version:** 3.11
* **PostgreSQL Version:** 13 or later (with PostGIS extension)
* **OpenSearch Version:** 2.x or later
* **Pip Package Manager**
* **VS Code or any Python IDE**

**Step 1: Install Dependencies Manually**

Before running the project, install the required Python libraries manually:

pip install psycopg2-binary

pip install geopy

pip install scikit-learn

pip install numpy

pip install concurrent.futures

pip install streamlit

pip install opensearch-py

pip install pandas

If using **PostGIS**, install it using:

CREATE EXTENSION postgis;

**Step 2: Set Up PostgreSQL Database**

1. Open PostgreSQL and create a new database:

**CREATE DATABASE tweet\_data**;

1. Switch to the new database:

**\c tweet\_data;**

1. Create the tweets table:

**CREATE TABLE tweets (**

**tweet\_id BIGINT PRIMARY KEY,**

**latitude DOUBLE PRECISION,**

**longitude DOUBLE PRECISION,**

**keywords TEXT[],**

**keyword\_weights FLOAT[]**

**);**

1. Ensure PostGIS extension is enabled:

**CREATE EXTENSION IF NOT EXISTS postgis;**

**Step 3: Load and Index Data**

**a. Process Tweets**

**We first process the raw tweet data to clean and structure it properly.**

python process\_tweets.py

**b. Clean Processed Tweets**

**We further clean the tweet data by removing unwanted characters, special symbols, and irrelevant data.**

python clean.py

**c. Indexing the Data**

**To enable fast querying and retrieval, we index the tweets into OpenSearch:**

python teq\_indexing.py

**c.Load Data into PostgreSQL**

**Once the tweets are processed and cleaned, we insert the cleaned tweet data into PostgreSQL using:**

python load\_indexed\_data.py

**Step 4: Run Query Algorithms**

**POWER Algorithm (Top-k Spatial-Textual Queries)**

To run the **POWER Algorithm** for query processing, execute:

python power\_algorithm.py

**Boolean Range Queries**

To run **Boolean Range Queries** with negative keyword filtering, execute:

python power\_boolean\_range.py

**RCA Algorithm**

To run **RCA Algorithm** execute:

python rca\_algorithm.py