Project title: Social Media Engagement Analytics

1. Introduction

This report documents the design and implementation of a backend system for social media analytics. The project's primary goal was to transform a flat, non-relational dataset of social media activities into a clean, normalized relational database. This new structure facilitates advanced data analysis, including user engagement tracking, post ranking, and real-time reporting, thereby providing meaningful insights from raw data.

2. Abstract

The Social Media Analytics Backend project successfully implemented a robust and scalable database solution using PostgreSQL. The core of the project involved designing a normalized schema with Users, Posts, Likes, and Comments tables to accurately represent social media interactions. A key component of the solution was a data cleaning and population script that converted summarized data (e.g., total likes as a count) into individual, relational records, simulating real-world user activity. The project deliverables include advanced SQL queries utilizing window functions for post ranking and a trigger for real-time like count updates, showcasing proficiency in database design and analytics.

3. Tools Used

Database Management System: PostgreSQL

Database Language: SQL (Structured Query Language)

4. Steps Involved in Building the Project

Schema Design: The first step was to design a normalized relational schema. This involved breaking down the original flat data into four distinct tables: Users, Posts, Likes, and Comments. Primary and foreign keys were defined to establish relationships between these tables, ensuring data integrity and preventing redundancy.

Data Cleaning and Population: A temporary table (RawData) was created to hold the un-cleaned dataset. Using a combination of INSERT, SELECT, and ON CONFLICT statements, data was migrated to the permanent tables. The key challenge was to convert the integer likes and comments counts from the raw data into individual rows in the Likes and Comments tables. This was achieved using the GENERATE_SERIES function, which dynamically created the required number of rows for each post.

Analytics and Reporting: Once the data was structured, several advanced analytics queries were written to extract meaningful insights:

A PostEngagement view was created to quickly query top-performing posts based on likes and comments.

Window functions were employed within a Common Table Expression (CTE) to calculate an engagement score and rank posts, demonstrating a key deliverable for advanced SQL.

A database trigger was created to automatically update a likes_count column on the Posts table whenever a new like is inserted, which is a common performance optimization technique.

A final report query was developed to provide a comprehensive view of a user's engagement, joining all four tables.

5. Conclusion

This project successfully demonstrates the principles of relational database design, data cleaning, and advanced SQL for analytics. By converting a simple dataset into a robust and scalable relational model, the system is capable of handling complex queries and providing valuable insights into social media engagement. The use of a database view, window functions, and triggers highlights a comprehensive understanding of database management best practices and the ability to deliver powerful, production-ready backend solutions.