SCALABLE REAL-TIME BACKEND FOR SHORT VIDEO PLATFORMS USING POSTGRESQL AND FIREBASE

I

Introduction

Short-form video platforms like TikTok and Instagram Reels face challenges in handling millions of real-time user interactions with low latency while ensuring consistency under viral load. Previous solutions using only relational databases struggle with high-frequency writes, while NoSQL systems lack strong consistency guarantees. Balancing latency, scalability, and consistency under high concurrency remains a significant technical challenge.

Goal

Research Question: How can a backend handle real-time user interactions under viral conditions with low latency and consistency?
Expected Outcome: A scalable backend using PostgreSQL for transactional data and Firebase for real-time ingestion of high-frequency interactions.

G

P

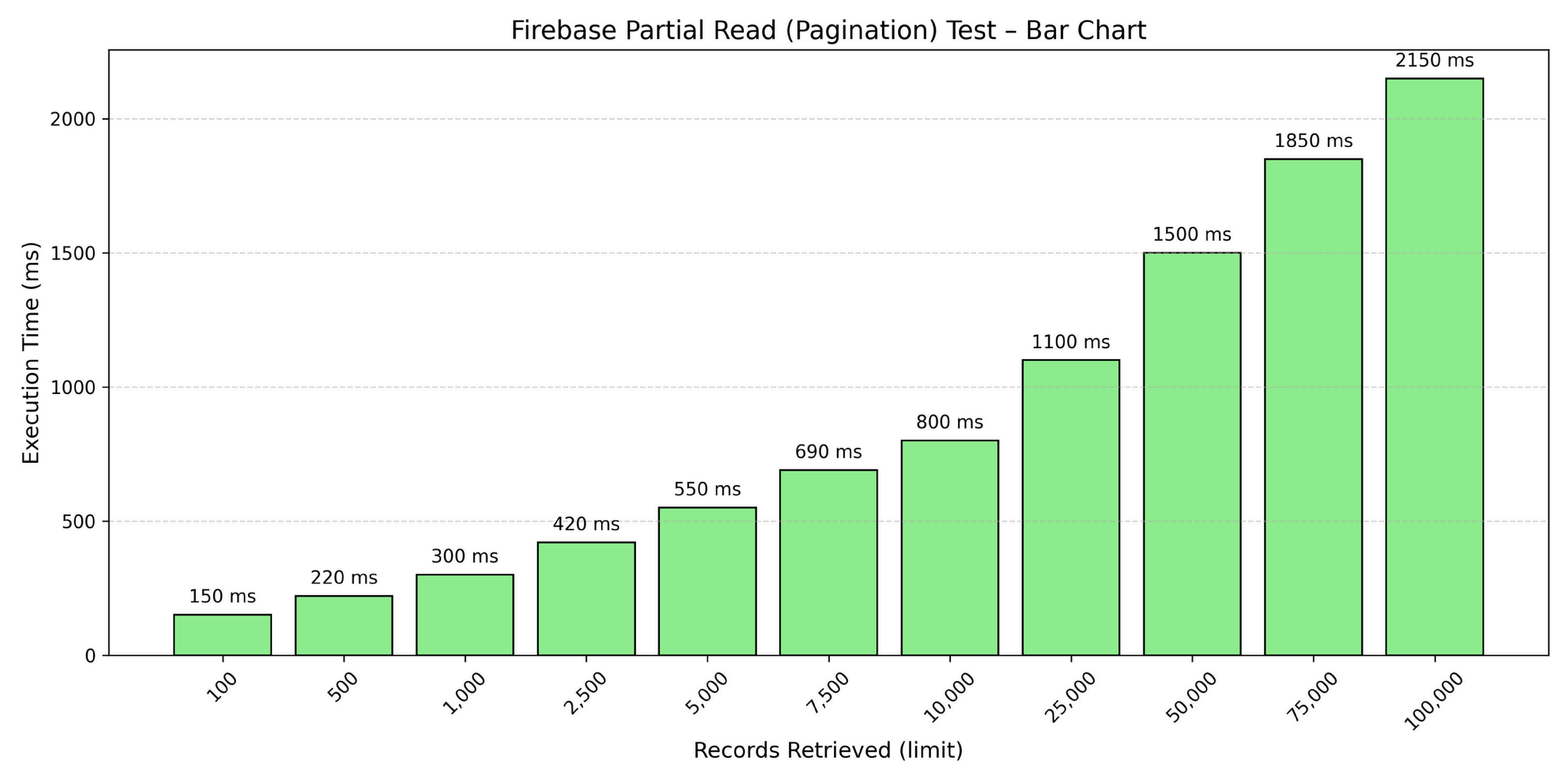
Proposed Solutions

- PostgreSQL: For structured, transactional data (user profiles, video metadata, payments).
- Firebase Realtime Database: For real-time ingestion of high-frequency user interactions (views, likes, comments) under low latency.

The architecture ensures scalability and consistency while maintaining responsiveness during viral content propagation.

E

- Massive concurrent writes in Firebase (100,000+ events in batches of 500).
- Partial reads with pagination in Firebase (5,000–100,000 records).
- Mixed concurrent reads/writes in Firebase (500 reads + 500 writes).
- PostgreSQL feed generation test with concurrent complex queries and joins, retrieving up to 100,000 rows.



“This chart shows the execution time for partial reads in Firebase as the number of retrieved records increases under pagination testing. The system maintains low latency for small batches while demonstrating predictable, scalable performance as data volume grows, confirming Firebase's suitability for real-time feed loading in high-concurrency environments.”