Distributed Streaming Platform Clone

Abstract

MovieVerse is a microservices-based movie recommendation platform that offers personalized content based on region-specific availability. Built with Flask, MySQL, Docker, Kong API Gateway, and Kubernetes, this system allows modular, scalable movie fetching with high security and performance. Movies are filtered into three regions — India, USA, and UK — and monitored using Prometheus.

1   System Overview

The system is built on microservices architecture comprising the following:

- **Flask API Service**: Handles requests and interacts with the MySQL database.

- **MySQL Database**: Stores metadata about movies, user preferences, and regions.

- **Kong API Gateway**: Manages user requests, routes to services, and provides security.

- **Prometheus**: Monitors services and visualizes metrics.

- **Kubernetes**: Orchestrates all Dockerized services for scalable deployment.

2   Architecture Design

2.1   Microservices and Components

- **Database (MySQL)**: Stores titles, languages, regions, and availability.

- **Data Access Layer (Flask)**: Pulls data in JSON and renders to frontend.

- **API Gateway (Kong)**: Filters traffic and ensures secure access.

- **Monitoring (Prometheus)**: Tracks performance and uptime.

- **Orchestration (Kubernetes)**: Manages container lifecycles and scaling.

Names from Database
Germany-Der mÃ¼de Tod
UK-A Matter of Life and Death
France, Italy-Le roi de coeur
France, Iran, USA-The Other Side of the Wind
India-Ankur
USA-Kiss Daddy Goodbye
USA-The Evil Dead
Hong Kong-Mo tai
India-Pestonjee
India-Ek Din Achanak
Chile-La Telenovela Errante
Iran-Leila
Canada-Joe Finds Grace
India-Aashirwad
Romania, Canada, USA-Teenage Space Vampires
USA-Against All Hope

Figure 1: Website from Dockerized MySQL Database

3 Region-Based Filtering

Kong routes incoming traffic based on region, resulting in 3 distinct endpoints:

- **India Service**: localhost:8050

- **USA Service**: localhost:8051

- **UK Service**: localhost:8052

  Each service is Dockerized and deployed via Kubernetes for load balancing and resilience.
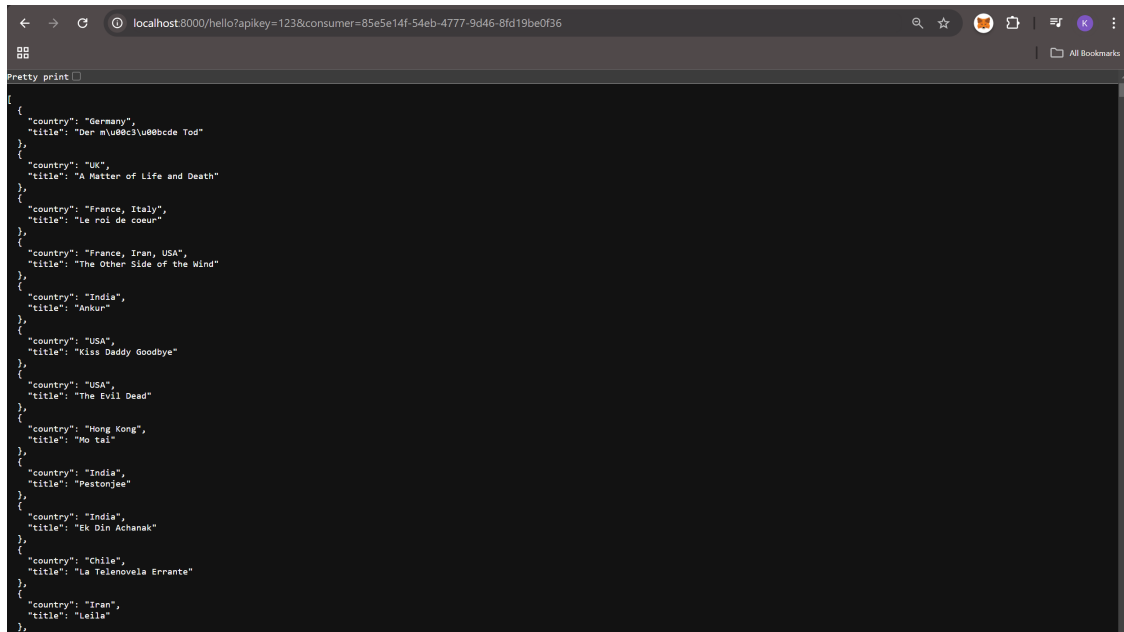
4 Kong API Gateway

Kong acts as the central access point for all user requests:

- **Routing and Load Balancing**: Directs traffic based on region.

- **Security**: Supports API keys and OAuth2.

- **Performance**: Uses caching, request limiting, and horizontal scaling.

- **Plugin Support**: Offers plugins for logging, monitoring, and transformations.

5 Monitoring and Scalability

Prometheus tracks metrics such as request rate and service health.

Figure 2: Movie Data via Kong API Gateway

6 Conclusion and Future Work

MovieVerse demonstrates how microservices, containerization, and orchestration can build a secure and region-aware recommendation platform. In the future, the system can be enhanced with:

- Circuit breakers for improved fault tolerance.

- Integration with external APIs for real-time movie updates.

- Advanced AI-based recommendation models.
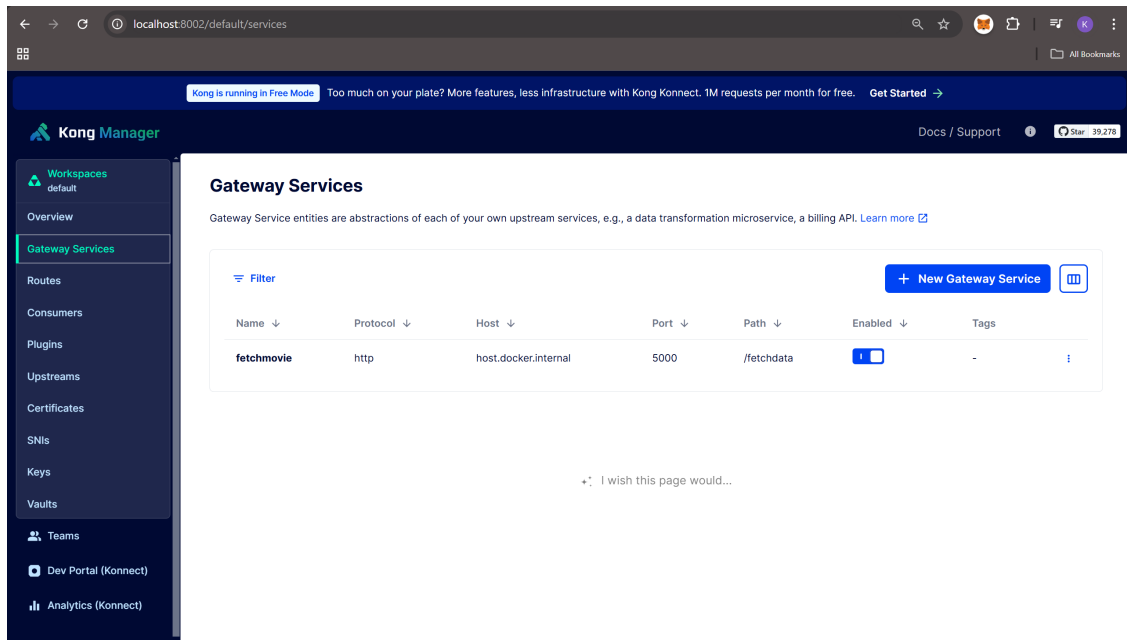
Figure 3: Filtered Movies for India
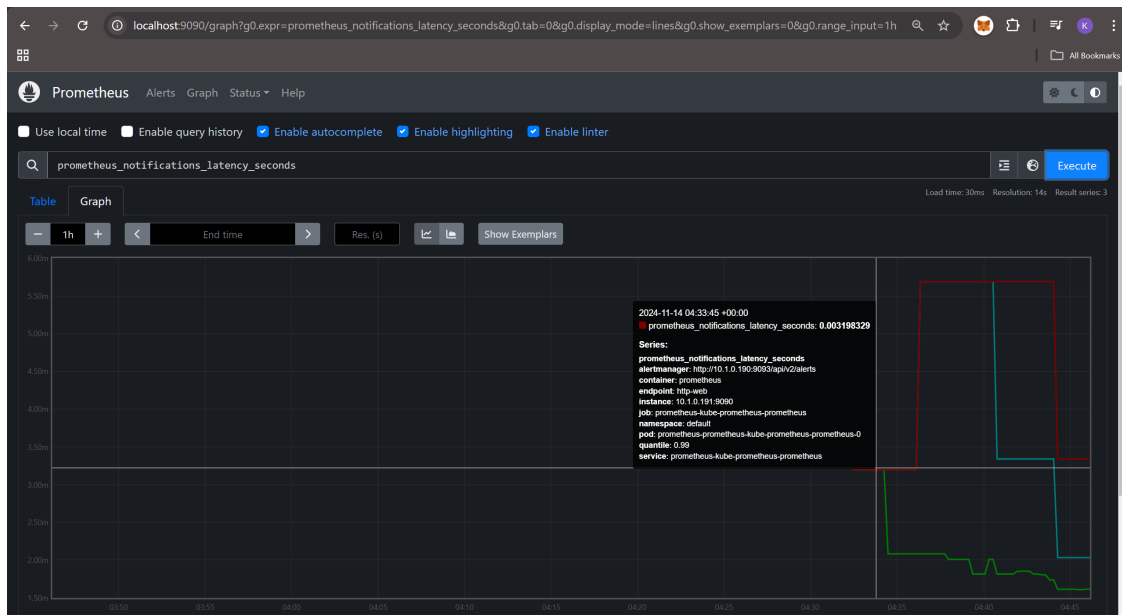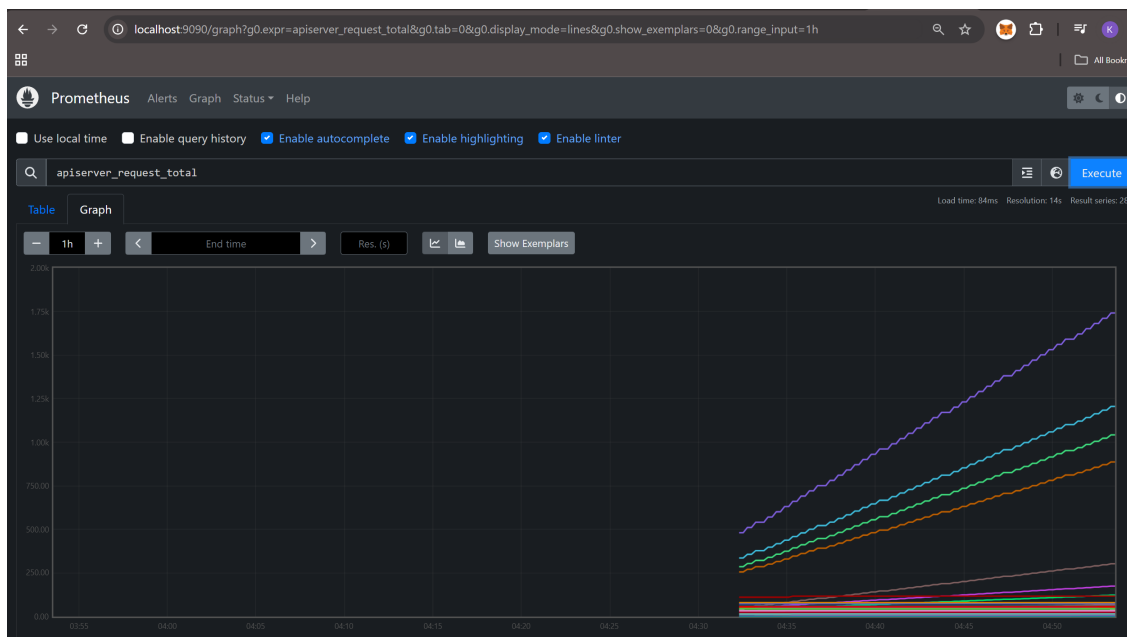

Figure 4: Kong API Gateway GUI

Figure 5: Prometheus Service Graph

Figure 6: API Request Count from Prometheus