

Project Name

- Quote Generator Service
-

Project Idea

A simple webservice that:

- Generates and retrieves random inspirational quotes.
- Allows adding new quotes to a local SQLite database.
- Filters out quotes containing swear or inappropriate words using a profanity filter.
- Exposes custom Prometheus metrics for monitoring.
- Uses Grafana dashboards for visualization.

The entire stack is containerized and runs locally using Docker Compose.

Team Members

- Tasmeem Mohsen Mahmoud
 - Omar Hussien AbdelRadi
 - Amira Hatem Anwar
 - Malak Yasser Mohamed Ali
 - Ahmed Mohamed Abdallah Ahmed Bakry
-

Work Plan

1. **Research & Analysis**
 - Audience personas
2. **Visual Identity**
 - Logo design
3. **Main Designs**
 - Poster
4. **Complementary Products**
5. **Review & Finalization**

6. Final Presentation

Roles & Responsibilities

- Tasneem Mohsen – (Docker Compose)
 - Amira Hatem – (Grafana)
 - Omar Huissein – (Prometheus)
 - Ahmed Bakry – (Docker & Documentation)
 - Malak Yasser – (Kubernetes)
-

Instructor

- Ahmed Gamil
-

Project Files

You can find the full project files here:

https://drive.google.com/drive/folders/1TkUkzulu4E4aL7a4uCEE_7vL0e1z4ly_?usp=sharing

License

This project is licensed under the GPL-3.0 license

- **KPIs (Key Performance Indicators)** – Metrics for project success (e.g., response time, system uptime, user adoption rate).

Quote Generation Count:

Counts the total number of quotes successfully generated and served to users.

Filtered Quotes Count:

Tracks how many quotes were rejected due to containing inappropriate or profane content.

Stakeholder Analysis

Below is the stakeholder analysis for the Quote Generator Service project:

Stakeholder	Role	Interest / Motivation	Influence / Impact
Project Manager	Oversees the project, manages timelines and risks	High interest in timely delivery and stability	High
Backend Developer	Implements the API, database, and filtering logic	Interested in clean, scalable, and maintainable code	High
Frontend Developer	Integrates API endpoints in frontend apps or dashboards	Needs stable API endpoints and clear documentation	Medium
End Users	Consumers of quotes via apps, websites, or bots	Want high-quality and appropriate quotes	High
Content Moderation Team	Reviews or flags inappropriate content if detected	Ensures content complies with brand policies	Medium
Marketing Team	Uses the quote API for promotional or engagement campaigns	Wants high uptime and reliable quote variety	Medium
System Administrator / DevOps Engineer	Maintains the Dockerized deployment and database	Ensures smooth deployment, scalability, and uptime	High

Database Design

The project uses SQLite as a lightweight embedded database. Below is the schema design:

Table: quotes

- id: INTEGER PRIMARY KEY AUTOINCREMENT
- text: TEXT (content of the quote)
- author: TEXT (person who said the quote)
- views: INTEGER DEFAULT 0 (number of times the quote was retrieved)
- created_at: DATETIME DEFAULT CURRENT_TIMESTAMP

The system ensures that all quotes added to the database are validated to exclude profanity and inappropriate words before storage. The database file is persisted using Docker volumes.

UI/UX Design



