Distributed Master-Slave Database System with Web Interface

Overview:

This project implements a distributed database system using a master-slave architecture in Golang.

The system allows for SQL operations such as SELECT, INSERT, UPDATE, DELETE, and replication of s

commands (CREATE, DROP, ALTER, TRUNCATE) from the master to connected slaves. Additionally, a w

using Flask enables interactive query execution.

Components:

1. Master Node (Go):

- Listens for TCP connections on port 9090.
- Executes SQL queries against the master database.
- Handles connected slave nodes and broadcasts replication queries.
- Accepts commands via CLI or from the web interface.

2. Slave Node (Go):

- Connects to the master and listens for replication queries.
- Maintains a local slave database.
- Supports SELECT, INSERT, UPDATE, DELETE operations.
- Reports executed queries back to the master.

3. Web Interface (Flask):

- Allows users to enter SQL queries.
- Displays query results or errors in a formatted HTML view.
- Sends SQL to the master using TCP.

Features:

- Full MySQL command support from the master (CREATE, DROP, etc.).

- SELECT query support with formatted response display.

- Real-time replication to slave nodes.

- Responsive HTML interface with Bootstrap styling.

Use Cases:

- Educational demonstration of distributed systems.

- Lightweight SQL testing and control.

- Simple master-slave database experimentation.

Technologies:

- Golang (Go)

- MySQL

- Flask (Python)

- Bootstrap (HTML/CSS)