REDDIT CLONE (Project 4.2)

By Shrey Gupta & Yash Bhalla

YouTube: https://youtu.be/g8zaPP_aeWc

This project implements a Reddit-like simulation using Go, demonstrating a scalable and efficient system for managing online communities and discussions. The application leverages Go's concurrency features, the actor model (using Proto Actor), and HTTP handling capabilities.

The system consists of several key components:
Main Application (main.go)

- Serves as the entry point for the application
- Initializes the random number generator
- Sets up goroutines for the server and client
- Creates an actor system and spawns the CommunityEngine actor
- Runs the simulation with configurable parameters
- Handles simulation termination and reports total time taken

Community Engine (engine.go)

- Core component handling business logic
- Maintains data structures for members, communities, messages, and threads
- Processes various message types using the Receive method
- Ensures thread-safety with locks when modifying shared data

Client Simulator (client.go)

- Simulates client interactions with the server
- Provides methods for user actions (e.g., registering, creating communities/threads)
- Sends HTTP requests to the server and logs interactions

Message Definitions (messages.go)

- Defines data structures and message types used throughout the application
- Includes structures for Member, Community, Thread, Reply, and PrivateMessage
- Defines message types for various actions

HTTP Server (server.go)

- Handles HTTP requests and interacts with the CommunityEngine
- Sets up routes for different actions
- Processes incoming requests and sends appropriate responses

Activity Simulator (simulator.go)

- Simulates user activity in the system
- Creates members, communities, and threads
- Generates random user actions (e.g., creating replies, casting votes)
- Uses a Zipf distribution to simulate content popularity
- Tracks and displays simulation metrics

Here are some key features implemented in the reddit clone:

- Concurrent processing using goroutines
- Actor model implementation for message handling
- HTTP server for client-server communication
- Realistic simulation of user activities and content popularity
- Metrics tracking for performance analysis

This Reddit-like simulation demonstrates the power of Go in creating scalable and efficient systems for managing online communities. By leveraging Go's concurrency model and the actor pattern, the project showcases a robust architecture capable of handling complex interactions in a social media-like environment.

How to run the code:
Install the required dependencies: go get github.com/asynkron/protoactor-go
Run the server: go run main.go server.go messages.go simulator.go engine.go
In a separate terminal, run the client: go run client.go

This is the simulation output: