

Reddit Clone

API Structure

My implementation follows a clean architecture with:

1. Models package defining request structures
2. Handlers package implementing endpoint logic
3. Router package defining API routes

Request Models

Well-defined request structures for:

- User registration
- Post creation
- Voting system
- Comment management
- Subreddit operations
- Direct messaging

API Endpoints

The router implements RESTful endpoints:

User Operations

- POST /api/register
- POST /api/dm

Post Operations

- POST /api/posts
- POST /api/posts/{id}/vote

Comment Operations

- POST /api/posts/{postId}/comments
- GET /api/posts/{postId}/comments
- DELETE /api/posts/{postId}/comments/{commentId}

Subreddit Operations

- POST /api/subreddits
- POST /api/subreddits/{name}/join
- POST /api/subreddits/{name}/leave

Handler Implementation:

Each handler:

- Uses Proto.Actor for asynchronous message processing
- Implements proper error handling

- Returns appropriate HTTP status codes
- Uses JSON for request/response formatting
- Extracts authentication from headers

1. **Handler:**

- Handles user registration and sending direct messages
- Uses the actor system to process requests asynchronously
- Properly sets HTTP status codes and returns JSON responses

2. **Handler:**

- Manages creating, retrieving, and deleting comments
- Uses URL parameters to identify posts and comments
- Extracts username from request headers for authentication

3. **Handler:**

- Handles creating subreddits, joining, and leaving them
- Uses URL parameters for subreddit names
- Extracts username from request headers for user identification

4. **Handler:**

- Manages creating posts and voting on them
- Uses request body for post content and voting information
- Extracts username from request headers for authentication

All handlers follow a consistent pattern:

- They use the actor system to process requests asynchronously
- They handle JSON request parsing and error responses
- They set appropriate HTTP status codes and return JSON responses

The handlers appear to cover the core functionality of a Reddit-like system, including user management, content creation, voting, and subreddit interactions.

Endpoint Structure

The router properly defines endpoints for all core Reddit functionality:

User Operations

- POST /api/register - User registration
- POST /api/dm - Direct messaging

Post Operations

- POST /api/posts - Create new posts
- POST /api/posts/{id}/vote - Vote on posts

Comment Operations

- POST /api/posts/{postId}/comments - Create comments
- GET /api/posts/{postId}/comments - Retrieve comments
- DELETE /api/posts/{postId}/comments/{commentId} - Delete comments

Subreddit Operations

- POST /api/subreddits - Create subreddits
- POST /api/subreddits/{name}/join - Join subreddits
- POST /api/subreddits/{name}/leave - Leave subreddits

Implementation Details

- Uses gorilla/mux for routing
- Implements CORS middleware properly
- Handles all necessary HTTP methods (GET, POST, DELETE)
- Includes proper header configuration (Content-Type, Authorization, Username)
- Routes are properly mapped to their respective handlers

The router implementation successfully connects all the handler functionality (UserHandler, PostHandler, SubredditHandler, CommentHandler) to appropriate REST endpoints, providing a complete API interface for the Reddit clone functionality.

The engine implements four core actors that handle different aspects of the Reddit clone functionality:

UserActor

Manages user-specific functionality:

- Karma tracking and updates
- Subreddit memberships
- Direct message handling
- Personal feed management
- Subscription management
- Repost functionality

SubredditActor

Handles subreddit operations:

- Member management (join/leave)
- Post creation and storage
- Member tracking using a map

PostActor

Manages post-related operations:

- Comment management
- Voting system (upvotes/downvotes)
- Karma updates for voting

- Comment tree structure

Client:

This implementation:

1. Covers all required functionality categories
2. Includes proper error handling
3. Manages authentication via username header
4. Handles JSON request/response formatting
5. Provides methods for all specified message types
6. Supports both synchronous and asynchronous operations
7. Maintains proper REST API endpoint structure

Each method corresponds to the message types defined in your proto file and interacts with the appropriate REST API endpoints.

Instructions to Run the Reddit clone:

1. Ensure you have Go installed on your machine.
2. Navigate to the project directory.

```
reddit-clone/
├── engine/
│   ├── main.go
│   ├── actors/
│   │   ├── user_actor.go
│   │   ├── subreddit_actor.go
│   │   ├── comment_actor.go
│   │   └── post_actor.go
│   ├── messages/
│   │   ├── messages.pb
│   │   ├── messages.proto
│   │   └── messages.go
├── api/
│   ├── handlers/
│   │   ├── user_handler.go
│   │   ├── post_handler.go
│   │   ├── subreddit_handler.go
│   │   └── comment_handler.go
│   ├── models/
│   │   └── requests.go
```

```
|   └── router.go
|── simulator/
|   ├── main.go
|   └── simulator.go
|── metrics/
|   └── metrics.go
|── tests/
|   └── test_api.sh
|── go.sum
|── go.mod
```

3. Run the commands:

- **Run the engine:**
go run engine/main.go
- **In a separate terminal, run the client:**
go run client/main.go client/client.go
- **In a separate terminal, we can also run the simulator for stress testing:**
go run .\simulator\main.go .\simulator\simulator.go
- **To regenerate protobuf code by running the protoc compiler.:**
protoc --go_out=. --go_opt=paths=source_relative messages.proto

4. Observe the output in the console for simulated user interactions.

```
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Install the latest PowerShell for new features and improvements! https://aka.ms/PSWindows

PS C:\Users\shaik\OneDrive\Desktop\PONY\Project4\RedditClone\reddit-clone> go run engine/main.go
# reddit-clone/engine/actors
engine\actors\post_actor.go:48:23: cannot use karmaChange (variable of type int) as int32 value in struct literal
engine\actors\post_actor.go:54:23: cannot use -karmaChange (value of type int) as int32 value in struct literal
PS C:\Users\shaik\OneDrive\Desktop\PONY\Project4\RedditClone\reddit-clone> go run engine/main.go
9:36PM INF actor system started lib=Proto.Actor system=KPjXNW55wSoLsHzUEaLmMX id=KPjXNW55wSoLsHzUEaLmMX
9:36PM INF Starting remote with address lib=Proto.Actor system=KPjXNW55wSoLsHzUEaLmMX address=127.0.0.1:8080
9:36PM INF Started Activator lib=Proto.Actor system=KPjXNW55wSoLsHzUEaLmMX
9:36PM INF Started EndpointManager lib=Proto.Actor system=KPjXNW55wSoLsHzUEaLmMX
9:36PM INF Starting Proto.Actor server lib=Proto.Actor system=KPjXNW55wSoLsHzUEaLmMX address=127.0.0.1:8080
2024/12/13 21:36:24 Starting REST API server on :8081
```

```
PS C:\Users\shaik\OneDrive\Desktop\PONY\Project4\RedditClone\reddit-clone> go run client/main.go client/client.go
2024/12/13 22:29:32 Starting Reddit Clone Client...
2024/12/13 22:29:32 User registered: user1
2024/12/13 22:29:32 User registered: user3
2024/12/13 22:29:32 User registered: user4
2024/12/13 22:29:32 User registered: user7
2024/12/13 22:29:32 User registered: user6
2024/12/13 22:29:32 User registered: user5
2024/12/13 22:29:32 User registered: user8
2024/12/13 22:29:32 User registered: user2
2024/12/13 22:29:32 User registered: user0
2024/12/13 22:29:32 User registered: user9
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: Books
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: UF
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: Books
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: CISE
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: Gaming
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: Science
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: USA
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: CISE
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: Gaming
2024/12/13 22:29:32 2024/12/13 22:29:32 Created subreddit: Movies
2024/12/13 22:29:32 2024/12/13 22:29:32 User user3 joining subreddit: Books
2024/12/13 22:29:32 2024/12/13 22:29:32 User user4 joining subreddit: USA
2024/12/13 22:29:32 2024/12/13 22:29:32 User user7 joining subreddit: Gaming
2024/12/13 22:29:32 2024/12/13 22:29:32 User user2 joining subreddit: UF
2024/12/13 22:29:32 2024/12/13 22:29:32 User user0 joining subreddit: CISE
2024/12/13 22:29:32 2024/12/13 22:29:32 User user1 joining subreddit: DOSP
2024/12/13 22:29:32 2024/12/13 22:29:32 User user5 joining subreddit: Movies
2024/12/13 22:29:32 2024/12/13 22:29:32 User user6 joining subreddit: Gaming
2024/12/13 22:29:32 2024/12/13 22:29:32 User user9 joining subreddit: Movies
2024/12/13 22:29:32 2024/12/13 22:29:32 User user8 joining subreddit: UF
2024/12/13 22:29:32 2024/12/13 22:29:32 User user2 leaving subreddit: USA
2024/12/13 22:29:32 2024/12/13 22:29:32 User user1 leaving subreddit: CISE
2024/12/13 22:29:32 2024/12/13 22:29:32 User user3 leaving subreddit: Technology
2024/12/13 22:29:32 2024/12/13 22:29:32 User user4 leaving subreddit: Books
2024/12/13 22:29:32 2024/12/13 22:29:32 User user0 leaving subreddit: UF
```

```
2024/12/13 22:29:32 User user4 downvoting post post9
2024/12/13 22:29:32 User user4 will lose karma for downvoting post post9
2024/12/13 22:29:32 User user1 downvoting post post8
2024/12/13 22:29:32 User user1 will lose karma for downvoting post post8
2024/12/13 22:29:32 User user0 downvoting post post0
2024/12/13 22:29:32 User user0 will lose karma for downvoting post post0
2024/12/13 22:29:32 User user7 upvoting post post7
2024/12/13 22:29:32 User user2 upvoting post post2
2024/12/13 22:29:32 User user2 will gain karma for upvoting post post2
2024/12/13 22:29:32 User user9 downvoting post post8
2024/12/13 22:29:32 User user9 will lose karma for downvoting post post8
2024/12/13 22:29:32 User user8 downvoting post post0
2024/12/13 22:29:32 User user8 will lose karma for downvoting post post0
2024/12/13 22:29:32 User user7 will gain karma for upvoting post post7
2024/12/13 22:29:32 User user6 downvoting post post7
2024/12/13 22:29:32 User user6 will lose karma for downvoting post post7
2024/12/13 22:29:32 Current karma for user user1: 404
```

```
2024/12/13 22:29:32 Operations for user user9 completed in 20.0968ms
2024/12/13 22:29:32 User user8 retrieving feed
2024/12/13 22:29:32 Operations for user user8 completed in 20.0968ms
2024/12/13 22:29:32 User user1 retrieving feed
2024/12/13 22:29:32 User user2 retrieving feed
2024/12/13 22:29:32 Operations for user user2 completed in 20.0968ms
2024/12/13 22:29:32 Operations for user user1 completed in 20.0968ms
2024/12/13 22:29:32 User user5 reposting post post1
2024/12/13 22:29:32 User user6 retrieving feed
2024/12/13 22:29:32 Operations for user user6 completed in 20.6457ms
2024/12/13 22:29:32 User user5 retrieving feed
2024/12/13 22:29:32 Operations for user user5 completed in 20.6457ms
2024/12/13 22:29:32 --- Metrics Report ---
2024/12/13 22:29:32 Total Users: 10
2024/12/13 22:29:32 Total Posts: 10
2024/12/13 22:29:32 Total Comments: 10
2024/12/13 22:29:32 Total Votes: 20
2024/12/13 22:29:32 Subreddits Created: 10
2024/12/13 22:29:32 Subreddit Joins: 10
2024/12/13 22:29:32 Leave Subreddit Count: 10
2024/12/13 22:29:32 Send DM Count: 10
2024/12/13 22:29:32 Retrieve Feed Count: 10
2024/12/13 22:29:32 Subscribe Count: 0
2024/12/13 22:29:32 Repost Count: 10
2024/12/13 22:29:32 Total execution time: 6.3415ms
2024/12/13 22:29:32 Exiting gracefully...
```

Curl commands for API calls:

1. Register a new user:

```
curl -X POST http://localhost:8081/api/register \
-H "Content-Type: application/json" \
-d '{"username": "testuser1"}'
```

2. Create a subreddit:

```
curl -X POST http://localhost:8081/api/subreddits \
-H "Content-Type: application/json" \
-H "Username: testuser1" \
-d '{"name": "technology"}'
```

3. Join a subreddit:

```
curl -X POST http://localhost:8081/api/subreddits/technology/join \
-H "Content-Type: application/json" \
-H "Username: testuser1"
```

4. Leave a subreddit:

```
curl -X POST http://localhost:8081/api/subreddits/technology/leave \  
-H "Content-Type: application/json" \  
-H "Username: testuser1"
```

5. Create a post:

```
curl -X POST http://localhost:8081/api/posts \  
-H "Content-Type: application/json" \  
-H "Username: testuser1" \  
-d '{"title": "Test Post", "content": "Test content", "subreddit": "technology"}'
```

6. Add a comment:

```
curl -X POST http://localhost:8081/api/posts/post1/comments \  
-H "Content-Type: application/json" \  
-H "Username: testuser1" \  
-d '{"content": "Test comment"}'
```

7. Vote on a post:

```
curl -X POST http://localhost:8081/api/posts/post1/vote \  
-H "Content-Type: application/json" \  
-H "Username: testuser1" \  
-d '{"is_upvote": true}'
```

8. Get user's karma:

```
curl -X GET http://localhost:8081/api/users/testuser1/karma \  
-H "Username: testuser1"
```

9. Send a direct message:

```
curl -X POST http://localhost:8081/api/dm \  
-H "Content-Type: application/json" \  
-H "Username: testuser1" \  
-d '{"to_user": "testuser2", "message": "Hello!"}'
```

10. Get user's feed:

curl -X GET http://localhost:8081/api/feed \

-H "Username: testuser1"

```
shaik@Anisa MINGW64 ~/OneDrive/Desktop/PONY/Project4/RedditClone/reddit-clone (master)
$ curl -X POST http://localhost:8081/api/subreddits/technology/leave \
  -H "Content-Type: application/json" \
  -H "Username: testuser1"
{"message": "Successfully left subreddit"}

shaik@Anisa MINGW64 ~/OneDrive/Desktop/PONY/Project4/RedditClone/reddit-clone (master)
$ curl -X POST http://localhost:8081/api/subreddits/technology/join \
  -H "Content-Type: application/json" \
  -H "Username: testuser1"
{"message": "Successfully joined subreddit"}

shaik@Anisa MINGW64 ~/OneDrive/Desktop/PONY/Project4/RedditClone/reddit-clone (master)
$ curl -X POST http://localhost:8081/api/posts \
  -H "Content-Type: application/json" \
  -H "Username: testuser1" \
  -d '{"title": "Test Post", "content": "Test content", "subreddit": "technology"}'
{"message": "Post created successfully"}

shaik@Anisa MINGW64 ~/OneDrive/Desktop/PONY/Project4/RedditClone/reddit-clone (master)
$ curl -X POST http://localhost:8081/api/posts/post1/comments \
  -H "Content-Type: application/json" \
  -H "Username: testuser1" \
  -d '{"content": "Test comment"}'
{"message": "Comment created successfully"}

shaik@Anisa MINGW64 ~/OneDrive/Desktop/PONY/Project4/RedditClone/reddit-clone (master)
$ curl -X POST http://localhost:8081/api/posts/post1/vote \
  -H "Content-Type: application/json" \
  -H "Username: testuser1" \
  -d '{"is_upvote": true}'
{"message": "Vote recorded successfully"}
```