Client
Documentation

Justin CHALEARD
2023/12/22

Summary

1. Introduction

Purpose

The purpose of this document is to provide comprehensive documentation for the client component of the chat application. It includes details about the system architecture, class diagram, client operations, error handling, security considerations, configuration, logging, and potential future enhancements.

Scope

This documentation covers the client-side implementation of the chat application, detailing the client's functionality, interactions with the server, and security measures.

Audience

This document is intended for developers, system administrators, and anyone involved in understanding, maintaining, or extending the functionality of the chat client.
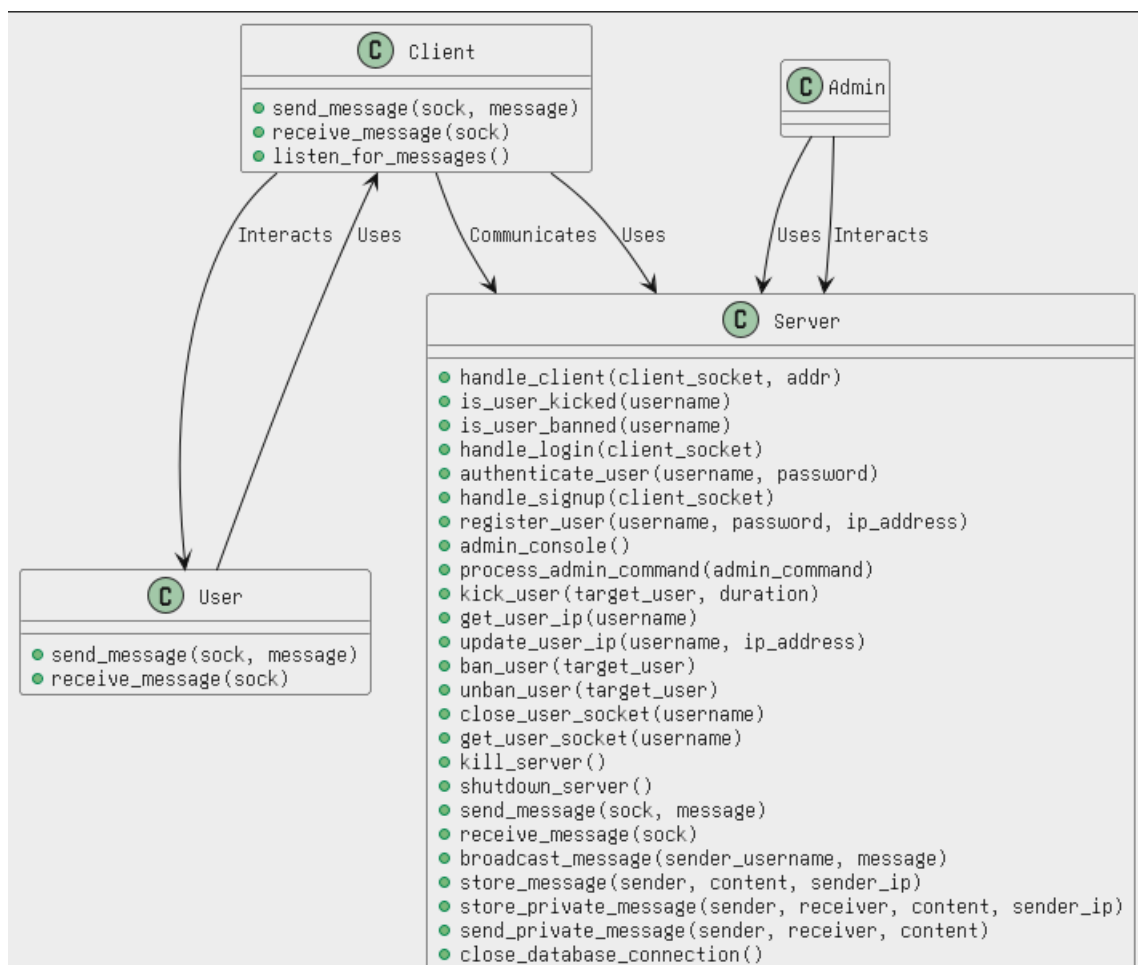
2. System Architecture

Overview

The chat client is part of a client-server architecture, connecting to a central server to send and receive messages. The client manages user authentication, message input, and display.

Components

- **Client:** The main client component responsible for connecting to the server, authenticating users, and handling message communication.

3. Class Diagram

4. Client Operations

Connecting to the Server

The client initiates a connection to the server upon startup and provides authentication details.

User Authentication

The client authenticates the user by sending a username and password to the server.

Sending Messages

The client sends messages to the server for broadcasting or private communication.

Receiving Messages

The client continuously listens for messages from the server and displays them to the user.

5. Client Configuration

Server Connection

The client's connection details, such as the server's host and port, can be configured in the code.

```python
# Example code snippet
SERVER_HOST = '127.0.0.1'
SERVER_PORT = 12345
```

6. Logging

Client Logs

The client logs important events, such as connection status, authentication, and message handling.

Error Logs

Error logs capture details about unexpected errors, helping in debugging and issue resolution.

7. Future Enhancements

- Implement additional chat features.
- Enhance user interface and user experience.

8. Conclusion

This documentation provides an in-depth understanding of the chat client's architecture, functionality, and key components. It serves as a valuable resource for developers and administrators involved in maintaining and extending the chat application.