



# Golang Assignment | Chat Server

---

## Objective

To evaluate candidate's understanding of HTTP protocol and Go concurrency primitives such as `goroutine`, `channel` and `sync` pkg.

## Problem Statement

---

Implement a chat server in Go which fulfills the following requirements.

### New Connection

- When connecting to the server, server assigns a unique ID to each user.
- Users can configure their display name.

### Chat Rooms

- Server can have N number of chat rooms.
- After connecting to the server, users can fetch the list of chat rooms.
- Users can join an existing chat room or create a new one.

### Messages

After joining a chat room,

- Users can send broadcast message to everyone in the chat room.
- The member details of the chat room (user IDs and display names) will be accessible to everyone who has joined the chat room.
- Members can send private messages to each other using user ID.

### Protocol

- Use HTTP protocol only.
- For sending messages or updates to clients, use SSE.

### Auth

- No need to implement any auth mechanism.

### Persistence

- No need for persist anything. Implementation can be fully in-memory.

### Frontend

- No need to implement frontend. Backend implementation is sufficient.

