

1. Servidor de Chat com Suporte a Múltiplas Salas

- Descrição: Implemente um servidor de chat onde os clientes podem criar e entrar em diferentes salas de chat. Cada sala deve ter seu próprio conjunto de clientes, e as mensagens enviadas em uma sala devem ser recebidas apenas pelos clientes dessa sala.
- Requisitos:
 - O servidor deve suportar múltiplas salas de chat simultâneas.
 - Os clientes devem poder listar as salas disponíveis e escolher em qual entrar.
 - Implementar comandos para criar, entrar e sair de salas.
 - As mensagens enviadas por um cliente em uma sala devem ser encaminhadas apenas para outros clientes na mesma sala.

2. Servidor de Arquivos Multithread com Download Paralelo

- Descrição: Implemente um servidor que permita aos clientes fazer o download de arquivos grandes. O servidor deve dividir o arquivo em partes e enviar cada parte em paralelo por diferentes threads.
- Requisitos:
 - O servidor deve aceitar pedidos de download de arquivos específicos.
 - Implementar uma forma de dividir o arquivo em partes e enviar essas partes em paralelo.
 - O cliente deve ser capaz de reconstituir o arquivo completo a partir das partes recebidas.
 - Implementar um mecanismo para verificar a integridade dos arquivos baixados (checksum, por exemplo).

3. Servidor de Jogo Multiplayer Básico

- Descrição: Desenvolva um servidor que permita a vários jogadores se conectarem e jogarem juntos um jogo simples, como um jogo de adivinhação de números, tic-tac-toe (jogo da velha) ou uma versão simplificada de batalha naval.
- Requisitos:
 - O servidor deve gerenciar o estado do jogo e coordenar as ações dos jogadores.
 - Implementar regras básicas do jogo escolhido.
 - Os jogadores devem poder ver o estado do jogo e realizar ações em tempo real.
 - Implementar um sistema de turnos, se necessário.

4. Servidor de Transferência de Mensagens com Fila de Prioridade

- Descrição: Implemente um servidor de mensagens que permita que as mensagens sejam enviadas com diferentes níveis de prioridade. O servidor deve processar e entregar as mensagens com base na prioridade delas.
- Requisitos:
 - Implementar um mecanismo para atribuir prioridades às mensagens (alta, média, baixa).

- O servidor deve processar as mensagens de alta prioridade antes das de média e baixa prioridade.
- Implementar um sistema de fila para armazenar e gerenciar as mensagens com suas respectivas prioridades.

6. Servidor de Streaming de Áudio Simples

- Descrição: Implemente um servidor que permita o streaming de áudio para vários clientes ao mesmo tempo. O áudio pode ser um arquivo pré-gravado ou um fluxo ao vivo.
- Requisitos:
 - O servidor deve ser capaz de enviar pacotes de áudio de forma contínua para os clientes.
 - Os clientes devem poder se conectar e começar a ouvir o áudio em tempo real.
 - Implementar controles básicos para pausar, continuar e parar o stream.
 - Tratar problemas de latência e sincronização entre os clientes.

7. Servidor de Autenticação e Autorização

- Descrição: Implemente um servidor que gerencie a autenticação de clientes e a autorização para acessar determinados recursos ou serviços.
- Requisitos:
 - Implementar um sistema de login com usuários e senhas.
 - O servidor deve manter um controle de sessões para os clientes autenticados.
 - Implementar diferentes níveis de acesso, onde apenas clientes autorizados possam acessar certos recursos ou comandos.
 - Tratar o armazenamento seguro das credenciais (por exemplo, usando hashing de senhas).

8. Servidor de Chat com Suporte a Criptografia

- Descrição: Desenvolva um servidor de chat onde todas as mensagens entre os clientes e o servidor sejam criptografadas.
- Requisitos:
 - Implementar criptografia simétrica ou assimétrica para proteger as mensagens.
 - Garantir que tanto o cliente quanto o servidor possam criptografar e descriptografar mensagens corretamente.
 - Implementar troca de chaves segura entre o cliente e o servidor.
 - Garantir que as mensagens criptografadas sejam seguras e não possam ser interceptadas ou lidas por terceiros.