

Redes de Computadores

Trabalho Prático - 1

João Pedro Schio Ortega

09 de Outubro de 2025

1 Descrição

O objetivo do trabalho é implementar e experimentar a comunicação em redes a partir da construção e da análise de servidores.

2 Servidor de chat multi-usuário

Para exercitar o uso de sockets, decidi implementar um servidor de chat onde clientes vão poder trocar mensagens globais ou criar grupos de conversa privada.

3 Implementação do servidor

A linguagem escolhida para o servidor foi Rust, devido às suas garantias de segurança e ao alto desempenho.

3.1 Compilar Rust

Como Rust ainda é uma linguagem mais nichada vou deixar aqui um tutorial de como compilar o servidor.

A maneira mais fácil e que funcionará independente da plataforma é utilizando o script rustup que a fundação Rust disponibilizou [rustup](#).

```
curl --proto 'https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

Esse script vai identificar o seu sistema operacional e instalar a versão correspondente do sistema de compilação do rust.

Após isso, é preciso navegar para o diretório chat-server em um terminal e rodar os comandos:

```
cargo run # para rodar em debug
cargo run --release # para rodar em release
```

Ao rodar esses comandos, o cargo, (sistema de compilação do rust) começará a baixar as dependências do projeto. Para seguir as especificações do trabalho foram-se utilizadas apenas as crates (bibliotecas) tokio e serde-json. tokio serve para ter uma runtime assíncrona e serde-json serve para ter suporte a JSON.

3.2 Protocolo

O protocolo escolhido para o envio de mensagens foi TCP por conta de suas garantias de entrega

3.3 Api

Para manter o projeto o mais próximo ao hardware possível, não tentei implementar algo parecido com HTTP, utilizei apenas o envio de JSON via TCP.

3.4 As mensagens enviadas ao servidor

Todas as mensagens trocadas com o servidor são enviadas usando um arquivo JSON que contém os seguintes campos:

1. Tipo de mensagem
2. Usuário
3. Conteúdo

3.5 Tipos de mensagem

Os tipos de mensagem e seus comportamentos.

1. Ola
2. MensagemGlobal
3. MensagemPrivada
4. Tchau
5. GetMensagens

3.6 Comportamento das mensagens

Tipo	Argumentos	Comportamento
Ola	String (nome)	Adiciona o usuário à lista de conectados e envia uma mensagem de boas-vindas.
Tchau	String (nome)	Remove o usuário da lista de conectados e notifica os demais.
MensagemPrivada	[String] (usuários), String (conteúdo)	Envia uma mensagem privada apenas aos usuários especificados.
MensagemGlobal	String (conteúdo)	Envia uma mensagem a todos os usuários conectados.
GetMensagens	String (Usuário)	Recebe como resposta todas as mensagens globais e todas as mensagens privadas que o usuario faz parte