

# Relatório da Atividade 4: Sistema de Votação Distribuída

Pedro Spinosa 476520,  
Henricky Lima 475075

Nov 2023

## 1 Introdução

Este relatório descreve a implementação de uma aplicação distribuída que suporta um sistema de votações. A aplicação permite que eleitores votem em candidatos e administradores gerenciem a votação. A implementação envolve comunicação unicast com sockets TCP, comunicação multicast com sockets UDP, uso de protocol buffers para representação de dados e a criação de um servidor multi-threaded.

## 2 Metodologia

### 2.1 Servidor (server.py)

O servidor é responsável por gerenciar as conexões dos eleitores e administradores, permitindo que eles participem do processo de votação. A implementação do servidor segue os seguintes passos:

1. Inicialização do servidor e configuração dos parâmetros de host e porta.
2. Criação de um socket TCP para aguardar conexões de clientes.
3. Inicialização de uma thread separada para controlar o encerramento da votação.
4. Autenticação de eleitores por meio da classe `User`.
5. Abertura e encerramento da votação.
6. Comunicação com os clientes por meio de sockets TCP.
7. Adição e remoção de candidatos por administradores.
8. Envio de notas informativas aos eleitores.

9. Encerramento das conexões dos clientes após a conclusão da votação.

O servidor utiliza a classe `VotingManager` para gerenciar o processo de votação, manter controle de candidatos e calcular os resultados da votação.

## 2.2 Cliente (`client.py`)

O cliente é um programa simples que permite aos eleitores interagir com o servidor. A implementação do cliente segue os seguintes passos:

1. Inicialização do cliente e configuração dos parâmetros de host e porta do servidor.
2. Conexão com o servidor por meio de um socket TCP.
3. Recebimento de perguntas e mensagens do servidor.
4. Exibição das perguntas e mensagens para o usuário.
5. Envio de respostas ao servidor.
6. Encerramento da conexão com o servidor após a conclusão da interação.

O cliente utiliza a classe `Client` para gerenciar a comunicação com o servidor. Ele aguarda as instruções do servidor, exibe-as para o usuário e envia respostas de volta.

## 3 Resultados

A implementação dos módulos `server.py` e `client.py` expande o sistema de votação distribuída para permitir que os eleitores e administradores interajam com o sistema por meio de uma interface de linha de comando. O servidor lida com várias conexões de clientes de forma concorrente, fornecendo uma experiência de votação em tempo real.

## 4 Conclusão

A atividade 4 resultou na criação de um sistema de votação distribuída funcional. Os módulos `server.py` e `client.py` desempenham papéis essenciais na operação do sistema, permitindo que eleitores e administradores participem da votação. A implementação bem-sucedida demonstra o uso de sockets TCP e UDP, protocol buffers e concorrência para criar um sistema distribuído robusto.