

# Relatório de Desenvolvimento - Projeto de Redes (Socket)

Discentes:

Marcelo Palmeira Falcão

Jose Milton de Moraes Neto

João Raphael Leão Guimarães Oliveira

Felipe Da Silva Araujo

## **Descrição da Aplicação**

A aplicação desenvolvida é um jogo de Jogo da Velha Multiplayer, onde dois jogadores podem se conectar à mesma partida por meio de uma rede local utilizando o protocolo TCP/IP. A comunicação entre os jogadores é realizada por meio de um servidor central, que gerencia as jogadas e a alternância entre os jogadores, além de verificar o status da partida, como a vitória ou empate.

## **Funcionalidades da Aplicação**

A aplicação é composta por três arquivos principais: cliente.py, servidor.py e rede.py. O cliente é responsável pela interface gráfica utilizando a biblioteca Pygame, enquanto o servidor lida com a comunicação entre os jogadores. A seguir, estão descritas as principais funcionalidades de cada parte:

- cliente.py: Este arquivo contém a lógica do jogo, incluindo a renderização do tabuleiro e das peças (X e O), o controle das jogadas dos jogadores, e a verificação da vitória. O jogador faz sua jogada clicando nas células do tabuleiro, e a jogada é enviada ao servidor, que a distribui para o outro jogador.

- servidor.py: O servidor gerencia as conexões de até dois jogadores e recebe as jogadas de cada um, alternando a vez entre os jogadores. Ele também verifica as condições de vitória e envia as atualizações do jogo para os clientes.

- rede.py: Este arquivo implementa a comunicação de rede entre o cliente e o servidor. Ele utiliza a biblioteca socket para enviar e receber dados via TCP, permitindo que os jogadores compartilhem suas jogadas em tempo real.

## **Protocolos Implementados**

O protocolo utilizado na comunicação entre o cliente e o servidor é o TCP/IP. O servidor aguarda conexões dos clientes e as aceita quando dois jogadores se conectam. Após a conexão, os dados (como jogadas e resultados) são trocados entre o cliente e o servidor utilizando sockets. O protocolo de comunicação é simples: o cliente envia a jogada e o servidor responde com a atualização do jogo, informando se a jogada foi válida e qual jogador deve jogar em seguida.

## **Melhorias Possíveis**

Embora a aplicação tenha sido concluída com sucesso, algumas melhorias poderiam ser implementadas:

1. Interface de Usuário: A interface gráfica poderia ser aprimorada, incluindo um menu de opções, escolha de cor das peças e animações durante a jogada.
2. Melhoria na Comunicação: Atualmente, a comunicação é simples, e não há controle de erros detalhado. Seria interessante implementar um sistema de reconexão ou detecção de desconexões inesperadas.
3. Armazenamento de Resultados: A aplicação poderia ser expandida para armazenar o histórico de partidas, permitindo que os jogadores acompanhassem suas vitórias e derrotas.

## **Dificuldades Encontradas**

Durante o desenvolvimento da aplicação, algumas dificuldades foram encontradas:

1. Sincronização entre os Jogadores: Garantir que o jogo estivesse sincronizado entre os dois jogadores foi um desafio, especialmente quando um jogador realizava uma jogada e o outro não via a atualização instantaneamente.
2. Tratamento de Erros de Rede: Como a comunicação entre os jogadores depende da rede, erros de conexão ou desconexões inesperadas causaram alguns problemas, que foram resolvidos com o tratamento de exceções no código.
3. Interface Gráfica: A implementação da interface com o Pygame foi desafiadora, especialmente para garantir que a interface estivesse fluida e sem falhas de renderização.