

UNIVERSIDADE FEDERAL DE ALAGOAS - UFAL
INSTITUTO DE COMPUTAÇÃO - IC/UFAL
REDES DE COMPUTADORES

RELATÓRIO DE PROJETO - AB2

Alunos: Élisson Souza; José Eraldo dos Santos Neto

- **Descrição do Projeto:** A aplicação implementada foi um mensageiro via linha de comando, que busca simular a troca de mensagens dos aplicativos de comunicação mais populares, tais como o *WhatsApp* e *Telegram*. As funcionalidades incluem enviar mensagem para um usuário e enviar mensagens para vários usuários, que receberão o histórico de mensagens anteriores, à medida que estes vão se conectando ao servidor.
- **Instruções de Execução da aplicação:**
 - **Requisitos:**
 - i. Para a aplicação funcionar, é necessário que haja a linguagem Python 3 no seu computador. Para instalar, siga essas [instruções](#) de acordo com o seu sistema operacional.
 - ii. Conhecimento básico do terminal do sistema operacional, tais como escolher o diretório, selecionar pastas e executar aplicações. Caso não tenha familiaridade, confira os comandos do [Windows](#) e do [Linux](#).
 - **Como executar a aplicação:**
 - i. Realize o download do arquivo do projeto [aqui](#): clique em *code* e depois em *Download ZIP*. Após o download, extraia a pasta para um diretório de sua preferência.
 - ii. Abra o **CMD** [Windows] ou o **terminal** [Linux] e navegue até o diretório que você extraiu o arquivo. Em seguida, selecione a pasta 'Computer-Networks-main' e dentro dela, selecione a pasta 'src'. Aqui, você encontrará os arquivos 'client.py' e 'server.py'.
 - iii. [*Server side*]: Execute o seguinte comando: 'python .\server.py'. Você deverá ver a mensagem 'Socket is running.' Significa que você iniciou o lado do servidor com sucesso.
 - iv. [*Client side*]: **Repita a etapa ii** para abrir uma nova janela do CMD ou terminal. Em seguida, execute o seguinte comando: 'python .\client.py'. Você deverá ver a mensagem 'Please, type your name: '. Significa que você iniciou o lado do cliente com sucesso.
 - v. No aplicativo do cliente, você deve digitar o seu nome e apertar enter. Depois, só escrever as mensagens que desejar. Note que no aplicativo

do servidor será informado o endereço IP e a porta a cada nova conexão e mensagem enviada. **Caso deseje inserir outros usuários para simular uma conversa, repita os passos ii - iv.** Para encerrar a aplicação, basta fechar as janelas do CMD ou terminal.

- **Desenvolvimento:** O aplicativo foi desenvolvido utilizando a linguagem de programação Python, utilizando as funções *Socket* da biblioteca [socket](#), que está presente no arquivo *server.py* e *client.py*, que permite a conexão e a comunicação entre dois processos (*inter-process communication*) rodando na mesma máquina; e *Threads*, presente na biblioteca [threading](#) (*_thread*), e permite a execução de tarefas de forma assíncrona.
- **Maiores dificuldades:**
 - Implementar o servidor de tal forma que os usuários clientes pudessem receber os dados de maneira simultânea;
 - Conectar o servidor e clientes;
 - Permitir a execução de tarefas paralelas utilizando a lógica de threads;
 - Entender o funcionamento de sockets e a sua implementação para a aplicação.
- **Implementações futuras:**
 - Uma interface intuitiva para usuários que não possuam familiaridade com o terminal e também para uma melhor experiência;
 - A data e hora do envio da mensagem;
 - Como desafio, estender a aplicação utilizando o protocolo FTP para a transferência de arquivos e não apenas mensagens entre os computadores.