

**Universidade Federal de Alagoas**  
**Redes de Computadores 2021.1**  
**Professor Leandro Sales**

**Alunos:** Emily Brito de Oliveira e Felipe Ferreira Vasconcelos

**Projeto Chat: Discord da Deepweb**

Implementação de sockets e multithreading utilizando a linguagem Python

### **Objetivo**

O objetivo da aplicação é funcionar como um chat **universal**, onde todos podem se conectar a um único “canal” e conversar com todos aqueles que estejam online naquele momento, numa ideia de “grupo de conversa”, onde todos que estiverem online irão visualizar as mensagens enviadas no grupo.

### **Principais funcionalidades**

- Cadastro e login de usuários
  - Ao acessar a aplicação, poderá ser feito o cadastro e o login, os dados são salvos em um arquivo chamado “usuarios.txt”.
- Múltiplos acessos simultâneos
- Log no servidor
  - Servidor tem log de informações como mensagens enviadas e usuários que entraram.
- Interface gráfica funcional
- Chat
  - Funcionalidade principal, chat entre os usuários online.

### **Possíveis implementações**

- Conversas particulares
  - Conversas privadas entre usuários, além do chat universal.
- Histórico de mensagens
- Sistema de cadastro mais seguro, com senhas.
- Mais opções envolvendo chat
  - Envio de mídia, seja áudio, imagens e etc

### **Dificuldades**

As principais dificuldades envolveram a implementação da interface gráfica funcional. Foi utilizada a biblioteca Tkinter, do Python, para a UI. Como nunca tínhamos usado o Tkinter antes, passamos pela curva de aprendizado que tomou um pouco de tempo e foi, inclusive, a razão do porquê não conseguimos implementar a funcionalidade de conversas diretas entre os clientes (não só as universais). Apesar de nunca termos usado, escolhemos o Tkinter em detrimento de outras ferramentas de GUI, pela comodidade de implementar algo já diretamente integrado com o código de cliente e da execução da interface também, por ser simplesmente a abertura de uma nova janela no sistema.

Outra dificuldade também foi manter o histórico de mensagens, um controle geral para todos os históricos, dado que foi feito “localmente”. Tivemos dificuldade em montar uma

solução abrangente, que funcionasse para todos os clientes, sem termos utilizar algo mais rudimentar como um arquivo textual que teria que ser atualizado a cada instante.

Uma terceira problemática foi a de lidar com exceções na conexão. Por exemplo, caso o servidor fosse encerrado e houvessem clientes online, não conseguimos entender o porquê de o cliente não receber a informação desse fechamento de conexão. Dessa forma, a aplicação não processava uma forma de avisar aos usuários a ausência do servidor e nem encerrava sua execução. Somente ao precisar enviar alguma mensagem do usuário, é que a aplicação percebia o erro no envio. Infelizmente esse problema não foi resolvido em tempo ábil para a entrega do projeto.

**Link para o repositório do projeto:**

<https://github.com/Emilybtoliveira/Projeto-Redes1>