

## **Projeto final de Redes de Computadores**

**Alunos:** Michael Miller Rodrigues Cardoso, Michel Thomas Gomes Lins

**Proposta:** Protocolo baseado em transferência de email com sockets e threads (Chat).

### **Principais funcionalidades da aplicação**

- Temos 5 arquivos python no projeto, um destinado ao servidor (que estabelece a conexão entre os clientes) e quatro clientes (que podem ser aumentado para N clientes, bastando duplicar o arquivo do primeiro);
- O servidor possui 3 funções:
  - `accept_connections` (configura o tratamento para os clientes conectados);
  - `server_broadcast` (transmite uma mensagem para todos os clientes);
  - `connect_client` (realiza a conexão de um único cliente ao chat).
- Os clientes têm as seguintes funcionalidades:
  - `receive_message` (recebe as mensagens dos outros clientes através do servidor);
  - `submit_name` (cadastra o nome de usuário para conectar-se ao servidor);
  - `submit` (envia uma mensagem/email para algum cliente);
  - `disconnect` (desconecta o cliente do chat);
  - `close_page` (tratamento para caso o cliente tente fechar a interface sem se desconectar);
- Explicando de forma resumida, o sistema funciona da seguinte maneira:
  - Um servidor socket fica disponível para a conexão dos clientes;
  - O cliente acessa a interface gráfica e cadastra seu nome de usuário;
  - Após o envio do nome de usuário, o cliente pode enviar mensagens;
  - O envio de email contém três campos: destinatário, assunto e mensagem, sendo o campo assunto opcional;
  - As mensagens enviadas são exibidas para o destinatário no seu inbox.

### **O que poderia ser implementado a mais**

- Interface mais elaborada;
- Permitir o envio de mensagens com texto mais longo;
- Clientes dinâmicos sem a necessidade de duplicação do arquivo.

## **Dificuldades encontradas no desenvolvimento**

- Tratar o envio das mensagens de um cliente para o outro através do servidor;
- Implementar a função de controle das mensagens do cliente;
- Aplicar o uso de threads no projeto.