

# Laboratório 2 - Sistemas Distribuídos

João Victor Pacheco Sobral

1 de agosto de 2021

## 1 Atividade

1. **Funcionalidades da camada de interface com o usuário:** recebe do usuário o nome do arquivo e a palavra de busca e exibe na tela o resultado do processamento. O resultado do processamento poderá ser: (i) uma mensagem de erro indicando que o arquivo não foi encontrado; ou (ii) o número de ocorrências da palavra no arquivo. As mensagens vão ser editadas nesta camada.
2. **Funcionalidades da camada de processamento:** solicita o acesso ao arquivo texto. Se o arquivo for válido, realiza a busca pela palavra informada e prepara a resposta para ser devolvida para a camada de interface. Se o arquivo for inválido, responde com a mensagem de erro. O resultado vai ser computado e enviado em forma de texto, “Arquivo nao existe ou nao temos permissao de leitura” se o arquivo não existir/não tiver permissão de leitura ou o número de ocorrências da palavra.
3. **Funcionalidades da camada de acesso aos dados:** verifica se o arquivo existe em sua base. Se sim, devolve o seu conteúdo inteiro. Caso contrário, devolve uma mensagem de erro.

## 2 Atividade

**Proposta de arquitetura de sistema:**

1. **Lado cliente:** implementa a camada de interface com o usuário. O usuário poderá solicitar o processamento de uma ou mais buscas em uma única execução da aplicação: o programa espera pelo nome do arquivo e da palavra de busca, faz o processamento, retorna o resultado, e então aguarda um novo pedido de arquivo e palavra ou o comando de finalização.
2. **Lado servidor:** implementa a camada de processamento e a camada de acesso aos dados. Projete um servidor iterativo, isto é, que trata as requisições de um cliente de cada vez, em um único fluxo de execução. Terminada a interação com um cliente, ele poderá voltar a esperar por nova conexão. Dessa forma, o programa do servidor fica em loop infinito.

**Mensagens:**

Na camada de interface vão ser lidos duas entradas, o nome do arquivo e a palavra apenas.

Na camada de processamento (parte do cliente) vai ocorrer a codificação em `nome_do_arquivo\x00palavra`, o byte zero significa uma separação e foi escolhido porque nomes de arquivo não podem conter esse caractere.

Na camada de processamento (parte do servidor) vai ocorrer a decodificação, separando em duas partes `nome_do_arquivo` e `palavra`. O byte de separação vai ser o `\x00`.

Na camada de processamento vai ser onde as palavras serão contadas. Quando a entrada estiver correta vai ser retornado o número de ocorrências da palavra no arquivo. Quando estiver incorreto temos dois casos:

1. “Arquivo nao existe ou nao temos permissao de leitura” - Quando não existir ou o arquivo não tiver permissão de leitura para o usuário que roda o serviço.
2. “Mensagem incorreta” quando a mensagem estiver incorreta, ou seja, a decodificação não ocorreu como esperado.

A camada de acesso de dados joga uma exceção para camada de processamento no caso de problema com a leitura do arquivo.

**Implementação:**

O cliente vai poder enviar e receber quantas mensagens quiser, caso queira parar as consultas ele pode apertar `ctrl+d`. É pedido o nome do arquivo e depois a palavra separadamente e a parte de codificação é feita internamente.

O servidor vai tratar a mensagem recebida independente se foi enviada do cliente verdadeiro (`client.py`) e vai processar a entrada na camada de processamento. A camada de processamento vai acessar a camada de acesso de dados e resgatar a resposta requerida pelo usuário ou uma mensagem de erro.