

Projeto Sockets: Corrida de Pacotes

- Data de lançamento: 03/07/2025
- Data de entrega: 14/07/2022
- Equipe: individual ou em dupla.

1. Regras:

- Os projetos devem ser implementados em Python 3 (preferencialmente), Java, C ou C++;
 - Os projetos devem rodar obrigatoriamente no sistema operacional Linux.
- Frameworks ou bibliotecas que facilitam a manipulação com sockets não podem ser utilizados;
- O que deve ser entregue:
 - Os códigos desenvolvidos num “.zip”;
 - Um relatório no formato “.pdf”;
 - Captura do Wireshark;
- Deve ser elaborado um relatório contendo:
 - As respostas para as perguntas apresentadas neste documento.
 - Seção com os passos que devem ser seguidos para executar o código fonte de forma satisfatória;
 - Apresentar telas com a execução do projeto.
 - O relatório também deve conter prints das capturas do wireshark comprovando o devido funcionamento do projeto.
- Cópias acarretarão em nulidade dos projetos das equipes envolvidas;
 - Um algoritmo de controle de autoria será utilizado para verificar as cópias;

2. Etapa 1: Resumo da Aplicação Base (Quiz)

A aplicação a ser desenvolvida deve atender os seguintes requisitos:

- O servidor envia uma pergunta a todos os clientes. As perguntas devem ser configuradas previamente no código e deve ser escolhido como multipla-escolha entre as opções [A, B, C e D].
- Cada cliente responde. As perguntas devem ser configuradas previamente no código, criar diferentes respostas que devem ser enviadas por cada cliente.

- O primeiro cliente a responder corretamente ganha a maior pontuação (1,0), os demais que acertarem devem ter um decaimento de 0,1 até chegar em 0, com isso o cliente não pontua mais nada. Em caso de erro, o cliente também deve pontuar 0.
- O servidor atualiza e transmite o placar após cada rodada.
- Após 5 perguntas, o jogo encerra e o resultado final é exibido pelo servidor.

3. Etapa 2 – Implementação Cliente-Servidor (TCP vs UDP)

Objetivo: Comparar desempenho entre uso de TCP e UDP para envio/recebimento das respostas do quiz. Para isso deve ser implementado duas versões:

- A. Versão A: Utiliza TCP sockets
- B. Versão B: Utiliza UDP sockets

Perguntas etapa 2:

- 1) Qual o tempo médio entre a solicitação da pergunta e até o envio da resposta para servidor? Existe diferença entre os protocolos?
- 2) Qual a ordem das respostas (quem chegou primeiro), o mesmo cliente sempre chega primeiro?
- 3) Qual o efeito da concorrência com 2, 4 e 8 jogadores? o tempo médio entre a pergunta e resposta é o mesmo para ambos os protocolos?

4. Etapa 3 – Segurança com Autenticação (HTTP vs HTTPS)

Objetivo: Comparar desempenho entre uso de HTTP e HTTPS para envio/recebimento das respostas do quiz. Para isso deve ser implementado duas versões:

- C. Versão C: Utiliza HTTP sockets
- D. Versão D: Utiliza HTTPS sockets

Perguntas etapa 3:

- 1) Qual o tempo médio entre a solicitação da pergunta e até o envio da resposta para servidor? Existe diferença entre os protocolos? O HTTPs possui alguma desvantagem devido a camada de segurança para o cenário analisado?
- 2) Qual a ordem das respostas (quem chegou primeiro), o mesmo cliente sempre chega primeiro?
- 3) Qual o efeito da concorrência com 2, 4 e 8 jogadores? o tempo médio entre a pergunta e resposta é o mesmo para ambos os protocolos?
- 4) Qual a diferença entre utilizar o HTTP/HTTS na camada de aplicação ou apenas enviar os dados brutos realizados na etapa 2? Quais as vantagens e desvantagens observadas em cada um dos casos?