

## Tópicos Avançados em Redes de Computadores

### Projeto Servidor HTTP/1.1

#### Introdução

Neste projeto, deverá ser implementado um servidor HTTP/1.1, capaz de interpretar alguns comandos do protocolo HTTP recebidos por meio de solicitações de navegadores/clientes, e responder de forma apropriada a essas requisições.

Os comandos a serem implementados são: **GET, PUT, POST e DELETE.**

O servidor deverá ser capaz de receber e interpretar mensagens HTTPs vindas de um navegador. Além disso, deverá ser capaz de montar mensagens de resposta e responder apropriadamente (segundo o protocolo HTTP/1.1) ao navegador que realizou a solicitação.

O servidor implementado deve ser capaz de executar as seguintes ações:

- Responder a uma solicitação do cliente enviado um objeto solicitado (como uma página, por exemplo).

- Ser capaz de armazenar novos objetos no servidor que foram passados por uma solicitação do cliente.

- Ser capaz de atualizar recursos já existentes no servidor.

- Ser capaz de remover recursos existentes no servidor.

- Responder de forma apropriada se um objeto solicitado não existir no servidor.

- Responder de forma apropriada se um objeto solicitado foi movido para outro diretório do servidor.

Notem que, “recurso” diz respeito a páginas e/ou objetos existentes em uma página, como imagens, áudio, vídeo, etc.

O servidor deverá receber solicitações HTTP geradas por um navegador e responder de forma apropriada. Por exemplo, se uma página foi solicitada, o servidor deve responder corretamente ao cliente enviando o recurso solicitado, que deverá ser exibido no navegador.

#### Versão do Protocolo e RFC

O servidor HTTP deve ser implementado segundo a RFC 2616 (<https://tools.ietf.org/html/rfc2616>), que define o protocolo HTTP/1.1.

Notem que a implementação deve seguir à risca o que está disposto na RFC desse protocolo para interpretação das mensagens, formatação das respostas, etc.

## Entrega e correção

A data de entrega do projeto é contada a partir de um mês após a apresentação deste documento.

O trabalho será realizado pelos grupos já constituídos da disciplina.

O servidor deve ser escrito na linguagem Python, que será utilizada por mim para execução do servidor e correção.

Todas as correções serão feitas no sistema operacional Ubuntu.

Junto com o código do servidor, cada grupo deve enviar um documento em .pdf contendo instruções para executar o servidor e para executar as solicitações no navegador ou no terminal do ubuntu (por exemplo, com exemplos de solicitação de objetos que vocês criaram dentro do servidor, exemplo de uma solicitação para colocar um novo objeto dentro do servidor (nesse caso vocês devem prover o recurso a ser inserido no servidor), exemplo de modificação de um recurso já existente, remoção de um recurso já existente). Essa documentação é **OBRIGATÓRIA**.

A correção será realizada utilizando-se de um navegador (o Google Chrome) para geração das requisições. O terminal do Ubuntu também será utilizado para acessar o servidor por linha de comando quando necessário com os exemplos providos por vocês.

O servidor deve rodar na máquina local, ou seja, no localhost. Note que, o localhost será a máquina destino de cada solicitação HTTP. Como é um projeto, a porta a ser utilizada no socket do servidor não precisa ser necessariamente a porta do protocolo HTTP.

É **EXPRESSAMENTE PROIBIDO** utilizar-se de bibliotecas prontas para executar um servidor HTTP, como a biblioteca http do Python. O servidor deverá ser implementado do zero por vocês, por meio do recebimento de requisições, interpretação das mesmas e geração de respostas apropriadas. Também é **EXPRESSAMENTE PROIBIDO** utilizar-se de uma API REST para interpretação dos comandos HTTP. Caso esses casos ocorram, o projeto receberá automaticamente nota 0.

## Acompanhamento Semanal

Todos os grupos deverão apresentar semanalmente o progresso do projeto. Essas pequenas apresentações semanais receberão feedback do professor.

É obrigatório apresentar semanalmente o progresso do projeto. Como o projeto irá durar 1 mês, cada apresentação acarretará em 0,25 pontos para a composição do fator P, que é um fator de participação do projeto (veja abaixo a utilização desse fator na composição da nota).

**Composição da nota**

Documentação do servidor: 2,0

Cada comando implementado (conforme explicado acima): 2,0 pontos cada.

Assim, Nota = (Documentação + (GET, PUT, POST, DELETE)) \* P

Se for notado qualquer plágio, o trabalho será zerado.