

PROFESSOR RESPONSÁVEL

Igor Moreira Félix - prof.igorfelix@usjt.br

OBJETIVO

Desenvolver um servidor web HTTP em Java, que atenda às requisições de métodos GET e HEAD. Este servidor deverá manusear as conexões dos clientes em *Threads* separadas. Além de apresentar relatórios variados ao administrador do servidor.

INTRODUÇÃO

No universo de hospedagem de conteúdos na Internet, existem três atores principais: o cliente, o servidor e a rede. O cliente acessa um endereço qualquer, através de um navegador, ou consome algum serviço, como um aplicativo de troca de mensagens. Enquanto a rede fica responsável por entregar as requisições do cliente ao servidor e vice-versa. O cliente, por sua vez, se empenha em receber, interpretar e responder os pedidos dos clientes. Todo esse funcionamento é resumido na Figura 1.

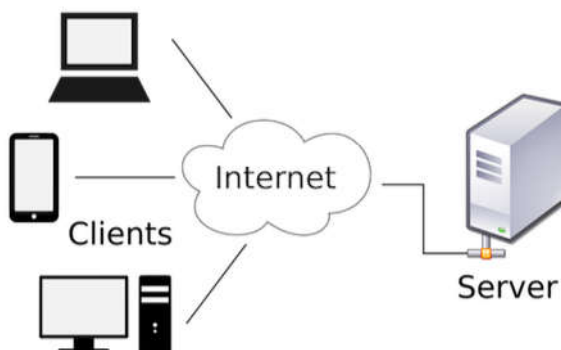


Figura 1: Esquema cliente-servidor

No servidor, são necessários softwares que recebem, interpretam e respondam corretamente às requisições dos clientes, entregando o conteúdo

solicitado. Existem diferentes softwares dessa categoria disponíveis no mercado, como o Apache HTTP Server, o NGINX, o Apache Tomcat e muitos outros.

Pensando nas tarefas do servidor, este projeto está sendo proposto. Sendo que a tarefa dos desenvolvedores é implementar um servidor web em Java, que atenda às requisições dos clientes, mais necessariamente, deverá tratar aquelas que forem enviadas via métodos GET e HEAD do protocolo HTTP.

O servidor desenvolvido deverá utilizar os recursos de Java Socket para permitir a comunicação com os clientes. Além disso, este sistema deverá obrigatoriamente trabalhar com múltiplas *Threads*, para proporcionar que o servidor atenda o maior número possível de clientes simultaneamente.

Inicialmente, uma interface gráfica deverá ser implementada, que é apresentada na Figura 2. Esta interface que ficará acessível no ambiente do servidor, permitirá que o administrador gerencie o sistema, tendo opções para iniciar, parar e reiniciar o serviço do servidor web.



Figura 2: Interface gráfica para gerenciar o servidor web

O servidor web deverá registrar em um banco de dados no MySQL todos os eventos que forem detectados tanto no gerenciamento do software, quanto as requisições recebidas dos clientes. As tabelas do banco de dados que deverão ser implementadas são apresentadas na Figura 3.

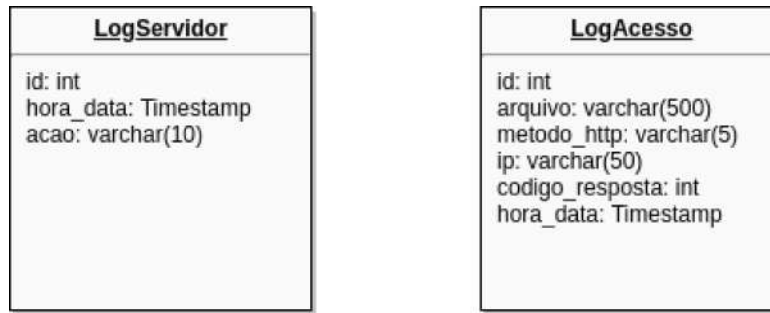


Figura 3: Diagrama das Entidades do Banco de Dados

Observa-se na Figura 3, a presença de duas entidades: LogServidor e LogAcesso. A tabela LogServidor deverá armazenar todos os eventos que acontecerem com relação ao gerenciamento do sistema, aquelas que estão disponíveis ao administrador, que foram apresentadas na Figura 2, sendo elas: inicialização, encerramento e reinicialização do software. Registrando para cada ação, o momento exato do fato.

Enquanto a tabela LogAcesso deverá armazenar todas as requisições que o software receber dos clientes. Nessa tabela, cada registro corresponderá a uma requisição recebida, anotando as seguintes informações: qual arquivo o usuário tentou acessar; qual método HTTP foi utilizado na requisição; qual foi o IP que originou a requisição; qual o código da resposta foi enviada ao cliente; e o momento exato do fato.

Todas as informações armazenadas no banco de dados servirão de fonte para a geração de uma série de relatórios para o administrador do sistema. Estes relatórios estarão acessíveis a partir de uma URL específica, como por exemplo: <http://meuservidor/relatorios>. Ao acessar este endereço o administrador deverá visualizar uma página similar à apresentada na Figura 4.

Alguns dos relatórios que deverão necessariamente ser apresentados são:

- arquivos mais acessados do servidor;
- horários do dia com maior número de requisições;
- IPs que mais enviam requisições;
- conteúdos que geraram erro 404;
- totais de requisições no dia e no mês;

- lista de IPs distintos que acessaram o servidor;
- lista de todos os eventos do servidor (inicialização, encerramento e reinicialização).

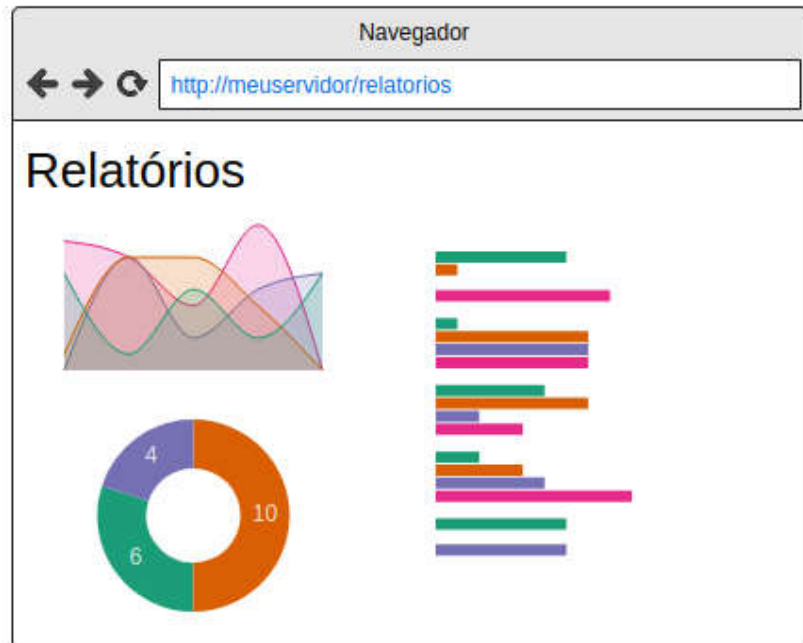


Figura 4: Página de relatórios

REQUISITOS DO PROJETO

Os requisitos do projeto englobam as condições mínimas para que o grupo esteja apto a apresentar seu trabalho à banca. Esses requisitos estão listados a seguir:

1. Interface gráfica com Java Swing para gerenciar o servidor web, tendo disponíveis as opções de iniciar, reiniciar e parar o serviço.
2. O servidor deverá atender obrigatoriamente aos métodos GET e HEAD do protocolo HTTP.

3. O software deverá tratar as requisições dos clientes em *Threads* separadas, a fim de atender o maior número de conexões simultâneas.
4. O servidor deverá permitir que o administrador configure as páginas de mensagem de erro das seguintes situações: 400 (requisição não compreendida); 404 (conteúdo não encontrado); e 405 (método não permitido).
5. O servidor deverá apresentar relatórios ao administrador, como: lista de últimos conteúdos acessados; conteúdos mais acessados; erros mais frequentes; etc.
6. Renderização correta do conteúdo HTML no navegador do cliente.

CRONOGRAMA

As entregas do projeto deverão seguir rigorosamente o cronograma apresentado abaixo, sendo inclusive critério para estar apto a ser apresentado à banca.

Atenção!

Cada entrega descrita no cronograma abaixo, deve ser inserida em uma subpasta da pasta compartilhada com o professor.

Exemplo:

Para a entrega 1, deve ser criada a sub-pasta de nome “entrega_1”. Em seguida, todos os arquivos relacionados a esta entrega, devem ser inseridos na pasta recém-criada

Tabela 1: Cronograma de entregas do projeto Servidor Web

#	Entrega	Meio	Data
1	Criação de uma pasta no Google Drive e compartilhá-la com o professor e com os demais membros do grupo.	Enviar o link por e-mail.	20/02/2019
2	Script SQL para criação da base. A base deve retratar exatamente aquilo contido no DER da Figura 2.	Postagem do arquivo.sql no Google Drive.	27/02/2019

3	Interface gráfica conforme Figura 1. Somente a interface gráfica, não sendo necessária a implementação das funcionalidades. Não é permitido o uso de ferramentas de geração de código (como Netbeans ou qualquer outra).	Postagem dos arquivos .java no Google Drive.	13/03/2019
4	Código em Java implementando Socket respondendo um simples texto "Hello World!" na tela do navegador do cliente.	Postagem dos arquivos .java no Google Drive.	27/03/2019
5	Código em Java implementando Socket devolvendo o arquivo buscado pelo cliente. Não é necessário implementar <i>Threads</i> .	Postagem dos arquivos .java no Google Drive.	03/04/2019
6	Código em Java implementando Socket devolvendo o arquivo buscado pelo cliente. Agora, deve ser implementado em <i>Threads</i> .	Postagem dos arquivos .java no Google Drive.	17/04/2019
7	Implementar a diferenciação dos tipos de métodos HTTP utilizados na requisição do cliente: GET e HEAD.	Postagem dos arquivos .java no Google Drive.	24/04/2019
8	Implementar o registro dos Logs de cada arquivo acessado pelo cliente.	Postagem dos arquivos .java no Google Drive.	08/05/2019
9	Implementar as mensagens de erros, deverá registrar nos Logs também.	Postagem dos arquivos .java no Google Drive.	15/05/2019
10	Implementar os relatórios para serem apresentados ao administrador do sistema.	Postagem dos arquivos .java no Google Drive.	22/05/2019
11	Integração entre a interface gráfica implementada na entrega 3 e o servidor.	Postagem dos arquivos .java no Google Drive.	29/05/2019
12	Aula de acompanhamento e plantão de dúvidas.	Sem entrega.	05/06/2019
13	Apresentações preliminares ao professor. 15 minutos para cada grupo. (Apresentação eliminatória, o professor	Apresentação em aula.	12/06/2019

	decide aqui se o grupo vai à banca ou não).		
14	Banca	Apresentação para a banca de professores.	19/06/2019

Observações importantes:

- a. Poderão ser acrescentados novos tópicos ao longo do projeto.
- b. Durante as aulas, o professor poderá repassar orientações sobre o desenvolvimento deste projeto que não constam neste documento. O aluno que não vier à aula e não obter estas informações, NÃO PODERÁ ALEGAR DESCONHECIMENTO das mesmas e requerer que sua nota não tenha prejuízo no momento da avaliação por não haver seguido as orientações que foram dadas em aula.
- c. Na apresentação do Projeto, cada aluno deverá participar e estar pronto para responder questionamentos. A nota pode ser atribuída individualmente, especialmente se ficar notório que não houve a mesma participação de todos na confecção do projeto.
- d. Formação dos grupos = mínimo de 5 e máximo 6 integrantes.
- e. A avaliação do Projeto Integrado, será a composição das notas abaixo:
 - i. Acompanhamento das entregas (com base no cronograma) = nota de 0 a 10
 - ii. Pré-banca (apresentação ao Professor de PI) = nota de 0 a 10
 - iii. Banca final (formada por professores das matérias) = nota de 0 a 10