

# Programação para Internet

## Bacharelado em Ciência da Computação – FACOM – UFU

Prof. Flávio de Oliveira Silva, Ph.D

Para realizar as atividades abaixo é necessário utilizar as ferramentas incorporadas a um navegador. Para este laboratório vamos considerar a última versão do FIREFOX com o “Web Developer Tools”. Antes de iniciar o painel “Web Developer Tools” deve ser iniciado e a aba NET deve estar selecionada e habilitada.

O objetivo deste laboratório é permitir inspecionar os pedidos (requests) e respostas (responses) do protocolo HTTP. Para maiores informações veja sobre como isto é feito veja a figura no final deste documento. Em cada item será necessário obter os dados no formato “raw” exibidos pelo “Web Developer Tools”. Portanto é importante coletar estes dados com cuidado em cada item.

Todos os itens devem ser realizados de forma individual e também de forma individual, cada um deve preparar suas análises e respostas.

1. Abra o navegador e digite o endereço de um site qualquer. No painel Network do “Web Developer Tools” obtenha o conteúdo do REQUEST e do RESPONSE. Copie o conteúdo do request e response e pesquise em alguma referência sobre o HTTP (como em <https://developer.mozilla.org/en-US/docs/Web/HTTP>) o significado de cada header envolvido no pedido e na resposta. Inclua na sua resposta as fontes utilizadas para a sua pesquisa indicando sua localização na WEB.
2. Faça uma pesquisa no Google utilizando um termo qualquer, porém acrescente na caixa de busca a seguinte informação: **filetype:pdf**. Assim sua caixa de busca conterá: “**termo**” **filetype:pdf**  
A partir do primeiro link devolvido, obtenha seu conteúdo e inspecione o resultado. Qual o valor para o header “content-type”? O que este valor mesmo significa? Inclua em seu relatório tanto o **request**, quanto o **response** obtido nesta operação.
3. Repita o mesmo procedimento acima, porém adicione na caixa de busca a seguinte informação: **filetype:xls**. Assim sua caixa de busca conterá: “**termo**” **filetype:xls**  
A partir do primeiro link devolvido, obtenha seu conteúdo e inspecione o resultado. Qual o valor para o header “content-type”? O que este valor mesmo significa? Inclua em seu relatório tanto o **request**, quanto o **response** obtido nesta operação.
4. Digite no navegador o seguinte endereço: [www.ufu.br/flavio](http://www.ufu.br/flavio). Veja o que acontece e comente o resultado após inspecionar os dados no “Web Developer Tools”. Inclua em seu relatório tanto o **request**, quanto o **response** obtido nesta operação.
5. Inspeção o request e o response para os seguintes endereços: <http://www.portal.facom.ufu.br/> e <http://www.portal.facom.ufu.br:80>. Existe alguma diferença entre request e response em ambos os casos? Explique este comportamento. Inclua em seu relatório tanto o **request**, quanto o **response** obtido nesta operação.
6. Inspeção o resultado de um pedido no seguinte endereço: <http://www.facom.ufu.br:443/>. Explique o que ocorreu? Inclua em seu relatório tanto o **request**, quanto o **response** obtido nesta operação.

7. Acesse seguinte endereço: <http://www.portal.facom.ufu.br:8080/>. O que ocorreu? Explique o comportamento do “Web Developer Tools” nesta situação. Qual alteração seria necessária no servidor a fim de que o endereço devolvesse algum resultado?
8. Inspeção o que ocorre com os seguintes endereços: <http://pardal.dr.ufu.br/> e <http://pardal.dr.ufu.br:8080/>. O que ocorreu no primeiro endereço? E no segundo? Explique a diferença de comportamento. O que significa o número 8080 adicionado no endereço? Inclua em seu relatório tanto o **request**, quanto o **response** obtido em cada uma destas operações.
9. Acesse o seguinte endereço: [www.ufu.com.br](http://www.ufu.com.br). Inclua em seu relatório tanto o **request**, quanto o **response** obtido em cada uma destas operações. Explique o que aconteceu. Além disso, considerando o protocolo HTTP, explique o que seria necessário para que este endereço retornasse algum resultado?
10. Acesse o seguinte endereço: <https://mail.ufu.br/>. Informe um usuário e uma senha e tente “entrar”. Olhe para o método “POST” e inspecione o conteúdo do **Request**. Quais informações foram enviadas para o servidor? Você entende que há algum risco envolvido nesta operação? Em caso afirmativo, indique quais. Inclua em seu relatório tanto o **request**, quanto o **response** obtido nesta operação.
11. Prepare um documento, descrevendo de forma detalhada com todas as respostas solicitadas, identificando e separando cada uma delas. Envie sua resposta para a **Atividade do TEAMS – Laboratório 01** e também envie uma cópia para [flavio@ufu.br](mailto:flavio@ufu.br) com o seguinte título: [BCC-PI]Trabalho Protocolo HTTP. O mesmo deve ser enviado ainda hoje até o final da aula.

## Uso do Web Developer Tools do FIREFOX

A última versão do FIREFOX possui uma ferramenta chamada “Web Developer Tools”. Para habilitar o painel com a ferramenta, é necessário utilizar o menu geral do FIREFOX, na sequência a opção “More Tools” e na sequência a opção “Web Developer Tools”, conforme mostrado na figura 1.

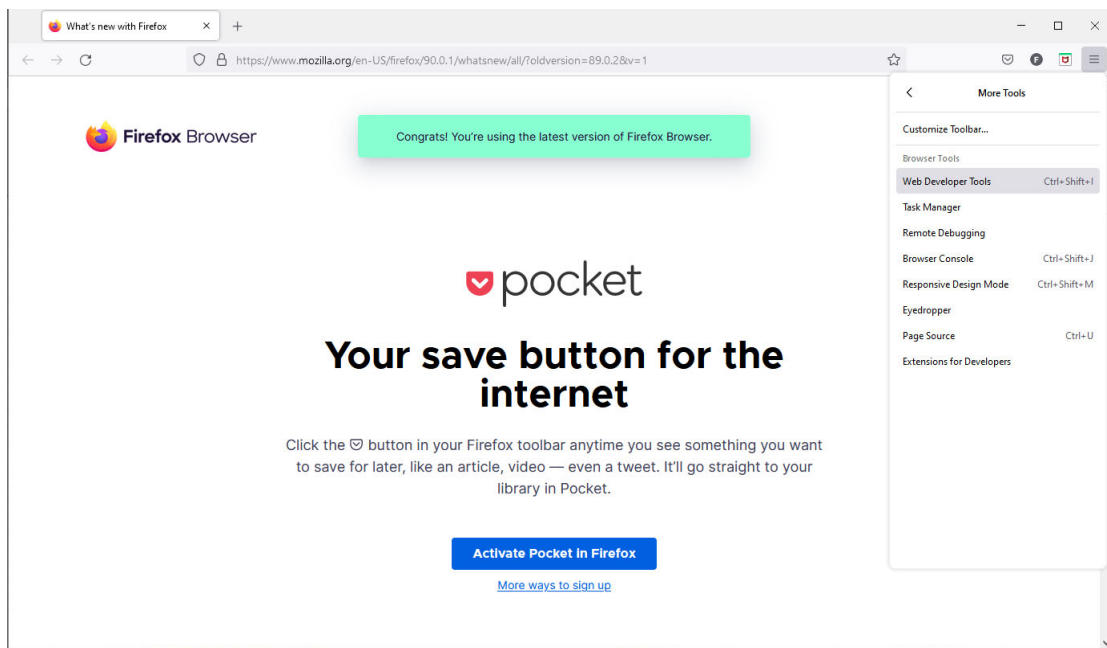


Figura 1 - Ativação do Web Developer Tools

Esta ferramenta permite monitorar o tráfego de pedidos e respostas do protocolo HTTP. Para visualizar os REQUESTS e RESPONSES é necessário clicar no painel “Network” conforme mostrado na Figura 2.

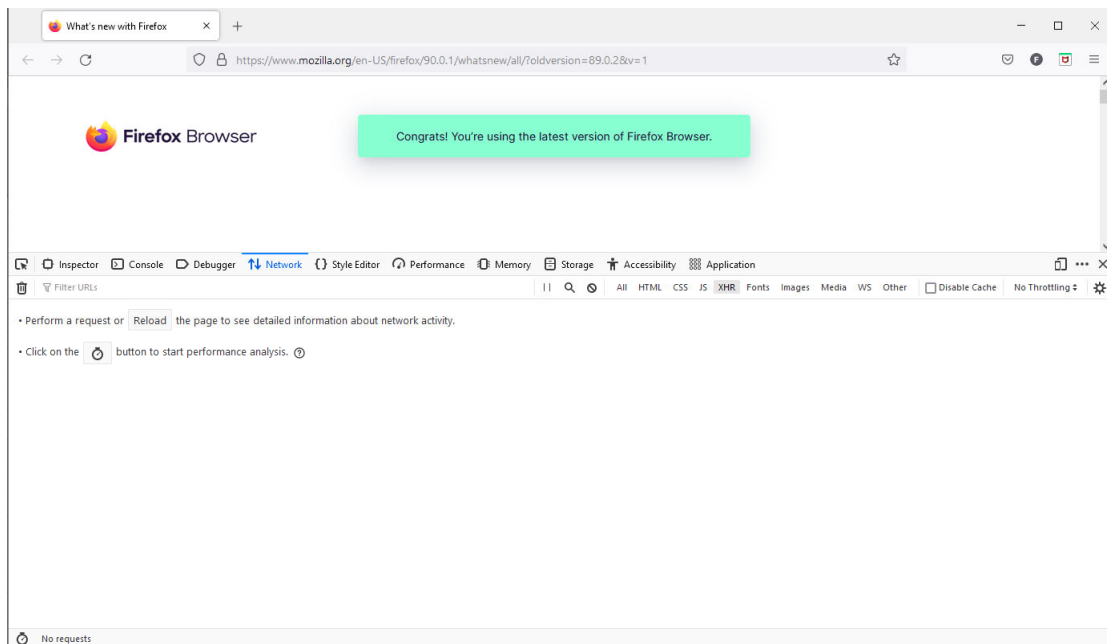


Figura 2 - Ativação monitoramento HTTP

Para ver o tráfego, basta informar um site e aguardar o resultado. Para visualizar o conteúdo, como mostrado na Figura 3, é necessário que a opção “All” esteja selecionada para visualizar todos os pedidos individuais necessários para compor uma página do site. Após selecionar um pedido qualquer é possível inspecionar todos os dados do pedido e da resposta, conforme a aba selecionada: Headers (permite inspecionar o cabeçalho da resposta - request headers – ou então o cabeçalho do pedido - request headers); Cookies; Request (informações associadas enviadas juntamente com o pedido) e Response (a resposta exibida em modo texto ou HTML). Para visualizar os dados como são trafegados, altere o modo de exibição para visualizar fonte “raw”

UFU | Universidade Federal de Uberlândia

www.ufu.br

**BRASIL** **CORONAVÍRUS (COVID-19)** Simplifique! Participe Acesso à informação Legislação Canais

Ir para o conteúdo Ir para o menu Ir para a busca Ir para o rodapé

ACESSIBILIDADE ALTO CONTRASTE

Universidade Federal de Uberlândia

MINISTÉRIO DA EDUCAÇÃO

COVID-19 Dados abertos Serviços Telefones Ouvidoria Imprensa Fale conosco

ESTUDANTE PROFESSOR TÉCNICO ADMINISTRATIVO COMUNIDADE

Graduação Pós-graduação

Inspector Console Debugger Network Style Editor Performance Memory Storage Accessibility Application

Filter URLs

Status	Method	Domain	File	Initiator	Type	Transferred	Size
200	GET	www.google.com	analytics.js	/75 [script]	js	cached	48.22 KB
200	POST	www.google.com	collected?v=1&_v=j&ip=1&a=42456394&t=pager&analytics.js=39 [...	analytics.js=39 [...	plain	638 B	2 B
200	GET	www.ufu.br	/	document	html	17.18 KB (cached)	97.71 KB
200	GET	www.ufu.br	js_vv1gU6LChHqbcSh4y1AwdKfDSQBIwt3GVGVUeuk: script	js	cached	113.58 KB	113.58 KB
200	GET	www.ufu.br	js_8kxotNk2y7YGF9QR9L9AMdnkQnN...ADBTW35Ie: script	js	cached	38.67 KB	38.67 KB
200	GET	www.ufu.br	js_BPT7FJUCS04zmV5uGp8FNQ6R0BOH6dCtG7yKbu: script	js	cached	11.93 KB	11.93 KB
200	GET	www.ufu.br	js_tH2RD0hTsvGQFMh6hE7E191JPBvVtEibC59vWaf4: script	js	cached	74.93 KB	74.93 KB
200	GET	www.ufu.br	js_h-5-dV-md39oe0e0D0D7F11-lafSp60UfTpZoO-8j: script	js	cached	66.08 KB	66.08 KB
200	GET	www.ufu.br	js_mBQqnv2jRTThvZVf9Cm1KFEC0RoI4UzREm5pW: script	js	cached	4.20 KB	4.20 KB

29 requests 1.50 MB / 158.30 KB transferred Finish: 2.48 s DOMContentLoaded: 275 ms load: 2.48 s

Filter Filter Headers Cookies Request Response Timings

GET http://www.ufu.br/

Transferred 17.18 KB (97.71 KB size)

Request Headers (424 B)

GET / undefined

Host: www.ufu.br

User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64; rv:90.0) Gecko/20100101 Firefox

Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/webp,\*/\*;q=0.8

Accept-Language: en-US,en;q=0.5

Accept-Encoding: gzip, deflate

Connection: keep-alive

Cookie: \_ga=GA1.2.1774420185.1622899551; hss\_js=1; \_gid=GA1.2.1835905227.1622874077; Upgrade-Insecure-Requests: 1

Raw

Figura 3 - Inspeção de Request Headers