

Instituto de

computação



Desenvolvimento Web HTTP

Prof. Leonardo Cruz.

leonardocruz@id.uff.br

Departamento de Computação, UFF

Objetivos

- Apresentar conceitos da Programação WEB usando a LP Java como exemplo

Ementa

- Introdução à linguagem Java.
- A API para acesso a bancos de dados relacionais.
- A API servlet (servlets como controladores de aplicações).
- Desenvolvimento de páginas jsp para a camada de interface

Programa

- HTML CSS e Javascripts.
- A API servlet (servlets como controladores de aplicações).
- Desenvolvimento de páginas jsp para a camada de interface

Esclarecimento

- Esse curso é baseado em materiais de diversos professores que disponibilizaram seus cursos na web.

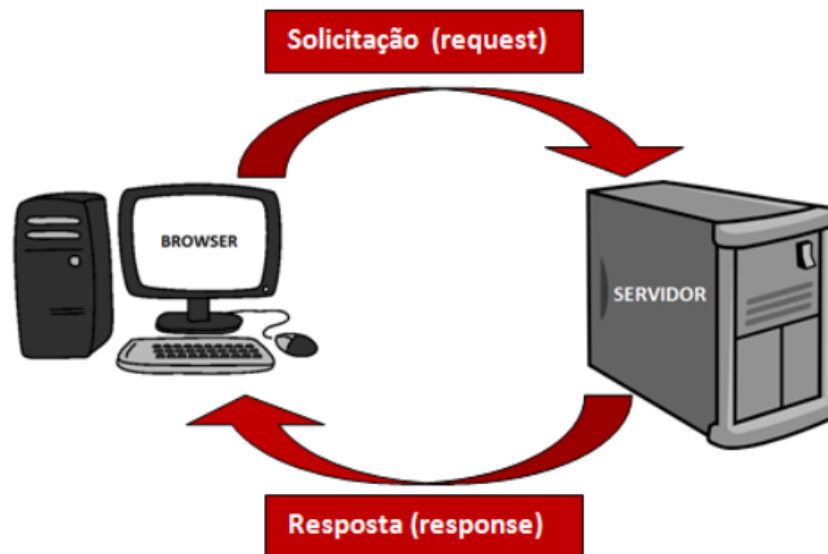
Desenvolvimento - tecnologias



Como funcionam as aplicações web

■ Servidor Web

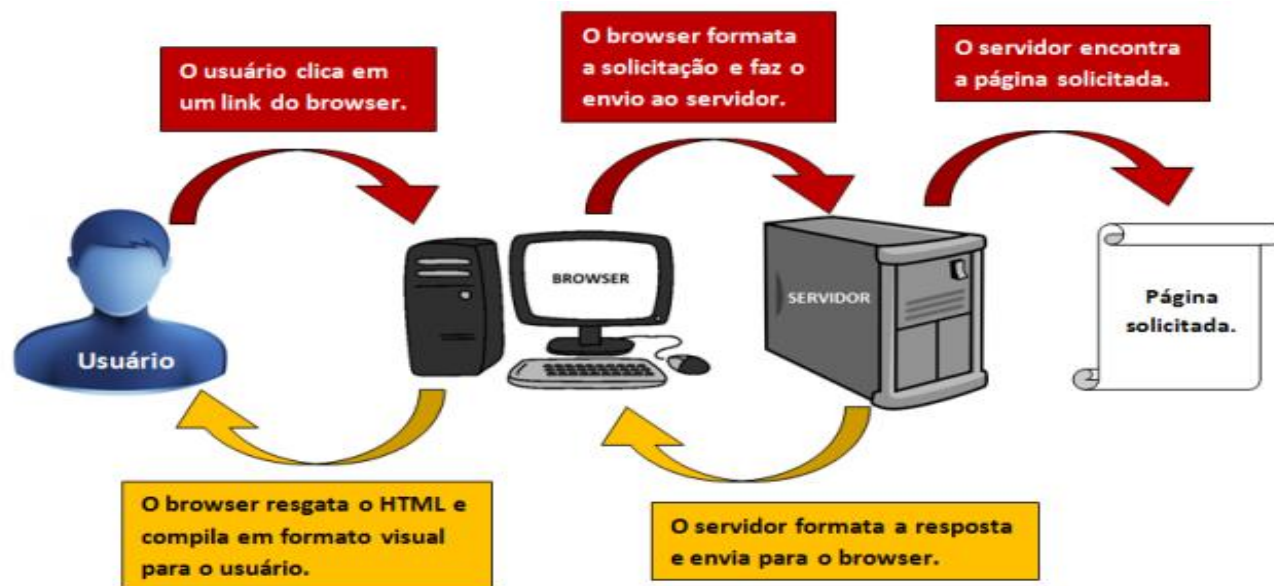
- A função do servidor web é receber uma solicitação (requisição) e devolver (resposta) algo para o cliente.



Como funcionam as aplicações web

■ Servidor Web

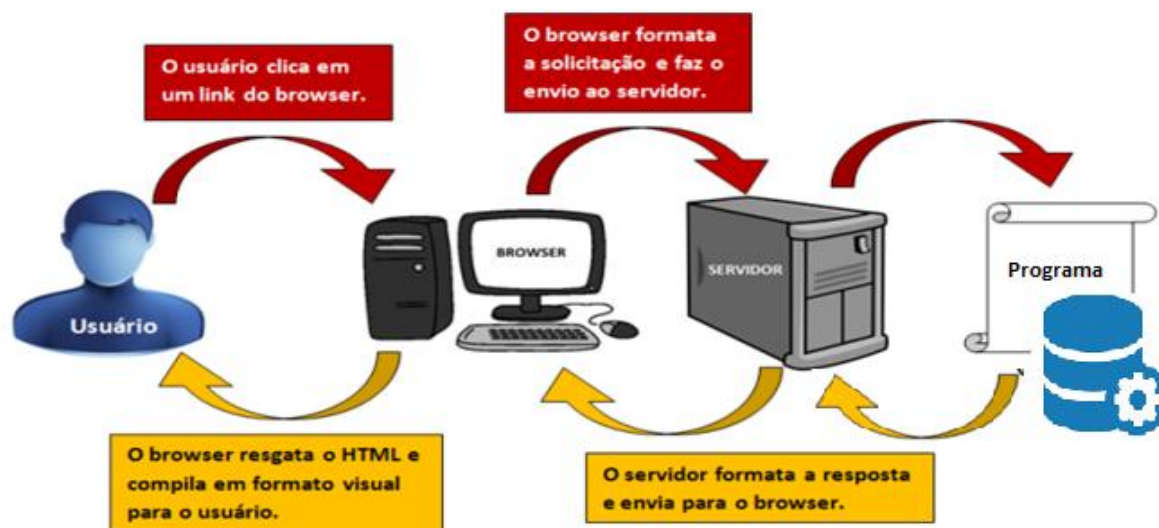
- **Páginas Estáticas**
- Construídas em HTML, exibem conteúdo fixo, onde a interação limita-se a links disponíveis na página.



Como funcionam as aplicações web

■ Servidor Web

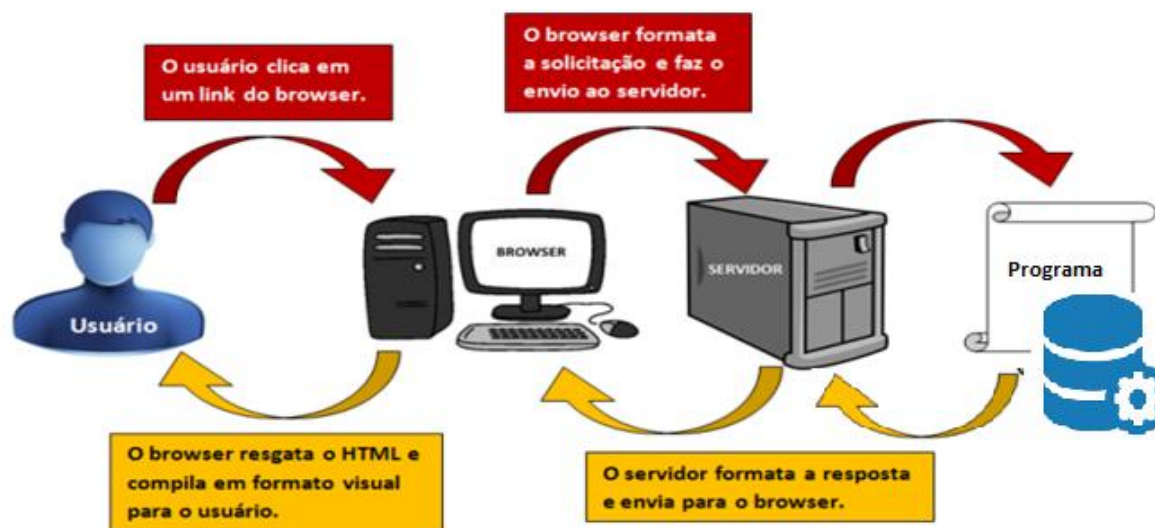
- O servidor em si tem alguns recursos, mas por algumas deficiências não consegue processar tudo sozinho como: criações de **páginas dinâmicas** e o **armazenamento de dados em um banco de dados**.



Como funcionam as aplicações web

■ Servidor Web

- **Páginas Dinâmicas** – para obter dados de BD é necessário o auxílio de uma outra aplicação



Protocolo HTTP

- É um protocolo que os clientes e os servidores usam para se comunicar. Essa comunicação é baseada em requisições (request) e respostas (responses).
- É um conjunto de regras de transmissão de dados que permitem que máquinas com diferentes configurações possam se comunicar com um mesmo “idioma”.

Protocolo HTTP

- O protocolo especificaverbos, cabeçalhos HTTP, etc.
- Quando o cliente solicita um recurso ele envia um pacote de dados com cabeçalhos (headers) a um **URL** (ou **URI**) e o destinatário (Provavelmente um Servidor) vai devolver uma resposta que pode ser um recurso ou outro cabeçalho.

Protocolo HTTP

- **URL - Uniform Resource Locator**
- Se refere ao HOST no qual se encontra algum recurso.
- Exemplo: **www.uff.br**

Protocolo HTTP

- **URN – Uniform Resource Name**
- É o nome do recurso que será acessado e também fará parte da URI.
- Exemplo: `home.html`

Protocolo HTTP

- **URI – Uniform Resource Identifier**
- A URI une o **Protocolo** (http://) a **localização do recurso** (URL – www.uff.br) e o **nome do recurso** (URN – home.html).
- `http://www.uff.br/home.html`

Protocolo HTTP

- esquema://**domínio**:**porta**/**caminho**/recurso?**query_string**#**fragmento**
- O **esquema** é o protocolo. Poderá ser HTTP, HTTPS, FTP etc.
- O **domínio** é o endereço da máquina: designa o servidor que disponibiliza o documento ou recurso solicitado.
- A **porta** é o ponto lógico no qual se pode executar a conexão com o servidor. (opcional)
- O **caminho** especifica o local (geralmente num sistema de arquivos) onde se encontra o **recurso**, dentro do servidor.
- A **query string** é um conjunto de um ou mais pares "pergunta-resposta" ou "parâmetro-argumento" (como por exemplo **nome=fulano**, em que **nome** pode ser, por exemplo, uma variável, e **fulano** é o valor (argumento) atribuído a **nome**).
- O **fragmento** é uma parte ou posição específica dentro do recurso. (opcional)

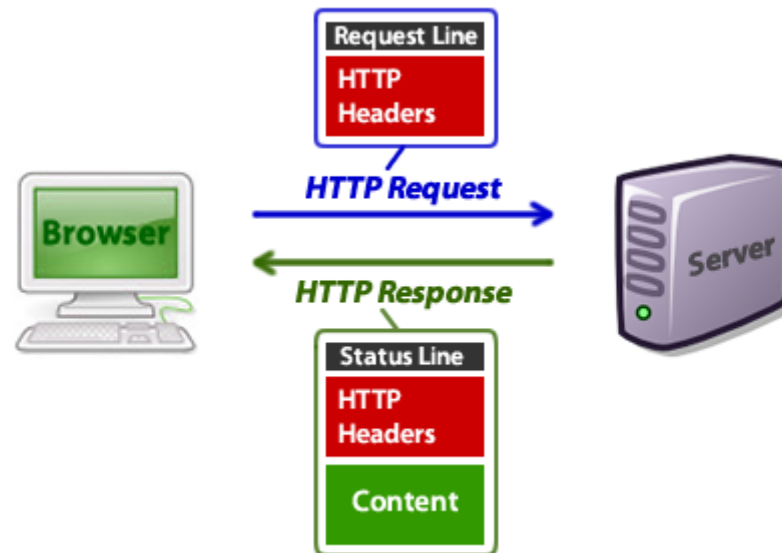
Protocolo HTTP

- Exemplo
- <http://www.w3.org/Addressing/URL/uri-spec.html>
- o protocolo é o HTTP, o servidor é designado por **www.w3.org** e o **recurso** (neste caso, o arquivo **uri-spec.html**) encontra-se em Addressing/URL/.
- A porta, omitida, recai sobre o padrão do protocolo (no caso, a porta **80**) e não há *query string* ou identificador de fragmento.

Protocolo HTTP

■ Conteúdo de uma solicitação

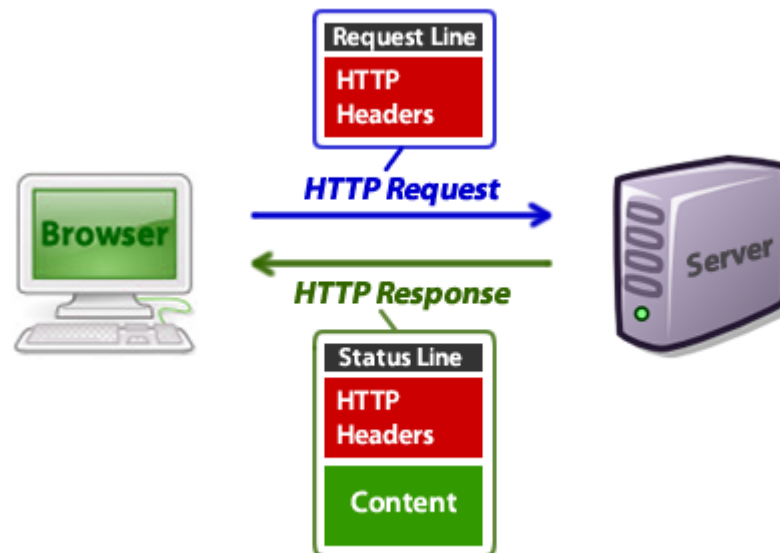
- Método HTTP;
- Página que será acessada;
- Parâmetros do formulário;



Protocolo HTTP

■ Conteúdo de uma resposta

- Código de status (informa se a solicitação foi realizada com sucesso ou não);
- Tipo de Conteúdo (HTML, figuras, textos, etc);



Protocolo HTTP

■ Conteúdo de uma solicitação

□ Métodos HTTP

- **GET** - Este é o método padrão, pois é a forma como o browser chama o servidor quando digita-se uma URL para que ele a recupere.
- **POST** - Contém um corpo nos quais seus parâmetros de solicitação já são codificados. O mais frequente uso desse método é na submissão de formulários.
- **HEAD** - Similar ao método GET, o servidor apenas retoma a linha de resposta e os cabeçalhos de resposta.

Protocolo HTTP

■ Conteúdo de uma solicitação

□ Métodos HTTP

- **PUT** - Esse método permite o envio de arquivos para o servidor Web.
- **DELETE** - Permite a exclusão de documentos dentro do servidor Web.
- **OPTIONS** - É possível fazer uma consulta de quais comandos estão disponíveis para um determinado usuário.
- **TRACE** - Permite depurar as requisições, devolvendo o cabeçalho de um documento.

Protocolo HTTP - exemplo

URL	Status	Domínio	Tamanho	Remote IP
GET www.uff.br	304 Not Modified	uff.br	24,4 KB	200.20.0.21:80
<div>Cabeçalhos Resposta HTML Cache Cookies</div>				
<div> <div>Cabeçalhos de Resposta</div> <div>ver fonte</div> <div> Cache-Control public, max-age=60 Date Thu, 27 Jul 2017 22:39:59 GMT Etag "1501193125-1" Expires Sun, 19 Nov 1978 05:00:00 GMT Server Apache/2.4.16 (Ubuntu) Vary Cookie,Accept-Encoding </div> </div>				
<div> <div>Cabeçalhos de Solicitação</div> <div>ver fonte</div> <div> Accept text/html,application/xhtml+xml,application/xml;q=0.9,*/*;q=0.8 Accept-Encoding gzip, deflate Accept-Language pt-BR,pt;q=0.8,en-US;q=0.5,en;q=0.3 Connection keep-alive Cookie has_js=1 DNT 1 Host www.uff.br If-None-Match "1501193125-1" Upgrade-Insecure-Requests 1 User-Agent Mozilla/5.0 (Windows NT 6.1; WOW64; rv:54.0) Gecko/20100101 Firefox/54.0 </div> </div>				
<div> <div>Response Headers From Cache</div> </div>				

Protocolo HTTP

■ Conteúdo de uma resposta

- **Código de status (informa se a solicitação foi realizada com sucesso ou não);**
- Tipo de Conteúdo (HTML, figuras, textos, etc);
- Conteúdo (HTML real, imagem, etc);

Protocolo HTTP

- **Conteúdo de uma resposta**

- **Código de status (informa se a solicitação foi realizada com sucesso ou não);**

- **200 (OK)** – Informa que a confirmação da requisição foi respondida com sucesso.
- **304 (NOT MODIFIED)** – Informa que os recursos que não foram modificados desde a última vez que foi feito um pedido. Isso ocorre por causa dos mecanismos de cache do browser.

Protocolo HTTP - exemplo

URL	Status	Domínio	Tamanho	Remote IP	Linha de tempo
+ GET www.uff.br	200 OK	uff.br	24,2 KB	200.20.0.21:80	4,35s
+ GET font-awesome.css	200 OK	netdna.bootstrapcdn.com	5,7 KB	94.31.29.55:80	1,31s
+ GET css?family=Open+Sans:800...	200 OK	fonts.googleapis.com	2,4 KB	216.58.222.106:80	1,2s
+ GET jquery.min.js?v=1.8.3	200 OK	uff.br	32,6 KB	200.20.0.21:80	79ms
+ GET jquery.once.js?v=1.2	200 OK	uff.br	1,0 KB	200.20.0.21:80	49ms
+ GET drupal.js?otrmcx	200 OK	uff.br	6,3 KB	200.20.0.21:80	85ms
+ GET jquery.ui.core.min.js?v=1.10.1	200 OK	uff.br	1,9 KB	200.20.0.21:80	50ms
+ GET jquery.ui.widget.min.js?v=1.1	200 OK	uff.br	2,4 KB	200.20.0.21:80	71ms
+ GET jquery.ui.mouse.min.js?v=1.1	200 OK	uff.br	927 B	200.20.0.21:80	51ms
+ GET jquery.ui.sortable.min.js?v=1	200 OK	uff.br	6,1 KB	200.20.0.21:80	85ms
+ GET jquery.cookie.js?v=67fb34f6a	200 OK	uff.br	1,3 KB	200.20.0.21:80	73ms
+ GET jquery.form.min.js?v=3.51.0	200 OK	uff.br	5,7 KB	200.20.0.21:80	84ms
+ GET ajax.js?v=7.56	200 OK	uff.br	7,6 KB	200.20.0.21:80	118ms

Protocolo HTTP

- **Conteúdo de uma resposta**

- **Código de status (informa se a solicitação foi realizada com sucesso ou não);**


- **401 (UNAUTHORIZED)** – Informa que o cliente não tem acesso autorizado para acessar a área requisitada. Ocorre muito em intranets de acesso privado que precisam ser acessadas com um usuário e senha.
- **403 (FORBIDDEN)** – Informa que o acesso à área requisitada falhou. Isso pode ocorrer em caso de acesso a áreas que exigem login e senha e não houve autorização para aquele usuário.

Protocolo HTTP

- **Conteúdo de uma resposta**
 - **Código de status (informa se a solicitação foi realizada com sucesso ou não);**
- **404 (NOT FOUND)** - Não encontrado. Ocorre quando o usuário tenta acessar uma área inexistente no endereço passado, por exemplo, páginas removidas ou recursos excluídos.

Protocolo HTTP - exemplo

- Tentativa de acessar www.uff.br/tempestade

 Limpar Persistente All HTML CSS JavaScript XHR Images Plugins Media Fonts						
URL	Status	Domínio	Tamanho	Remote IP	Linha de tempo	
+ GET tempestade	404 Not Found	uff.br	25,0 KB	200.20.0.21:80		
+ GET footable.js?v=0.1	200 OK	uff.br	8,2 KB	200.20.0.21:80		
+ GET autocomplete.js?v=7.56	200 OK	uff.br	2,6 KB	200.20.0.21:80		
+ GET caller.js?otrmcx	200 OK	uff.br	163 B	200.20.0.21:80		
+ GET footable.core.min.css?otrmcx	200 OK	uff.br	855 B	200.20.0.21:80		
+ GET css_injector_13.css?otrmcx	200 OK	uff.br	786 B	200.20.0.21:80		
+ GET xml.png	200 OK	uff.br	254 B	200.20.0.21:80		
+ GET pdf.png	200 OK	uff.br	529 B	200.20.0.21:80		

Protocolo HTTP

■ Conteúdo de uma resposta

- Código de status (informa se a solicitação foi realizada com sucesso ou não);
- **Tipo de Conteúdo (HTML, figuras, textos, etc);**

Protocolo HTTP

URL	Status	Domínio	Tamanho	Remote IP	Linha de tempo
GET www.uff.br	304 Not Modified	uff.br	24,4 KB	200.20.0.21:80	47ms
<div>Cabeçalhos Resposta HTML Cache Cookies</div> <pre> <!DOCTYPE html> <html lang="pt-br" dir="ltr" xmlns:content="http://purl.org/rss/1.0/modules/content/" xmlns:dc="http://purl.org/dc/terms/" xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:og="http://ogp.me/ns#" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:sioc="http://rdfs.org/sioc/ns#" xmlns:sioc:="http://rdfs.org/sioc/types#" xmlns:skos="http://www.w3.org/2004/02/skos/core#" xmlns:xsd="http://www.w3.org/2001/XMLSchema#" xmlns:og="http://opengraphprotocol.org/schema/> <head profile="http://www.w3.org/1999/xhtml/vocab"> <meta http-equiv="Content-Type" content="text/html; charset=utf-8" /> <meta name="HandheldFriendly" content="true" /> <meta name="viewport" content="width=device-width, initial-scale=1" /> <meta name="MobileOptimized" content="width" /> <link rel="alternate" type="application/rss+xml" title="Latest News" href="http://www.uff.br/?q=informativos.xml" /> <link rel="alternate" type="application/rss+xml" title="Universidade Federal Fluminense RSS" href="http://www.uff.br/?q=rss.xml" /> <title>Universidade Federal Fluminense</title> <style type="text/css" media="all"> @import url("http://www.uff.br/modules/system/system.base.css?otrmcx"); @import url("http://www.uff.br/modules/system/system.menus.css?otrmcx"); @import url("http://www.uff.br/modules/system/system.messages.css?otrmcx"); @import url("http://www.uff.br/modules/system/system.theme.css?otrmcx"); </style> </pre>					

Protocolo HTTP

URL	Status	Domínio	Tamanho	Remote IP	Linha de tempo
GET www.uff.br	304 Not Modified	uff.br	24,4 KB	200.20.0.21:80	47ms

Cabeçalhos

Resposta

HTML

Cache

Cookies

Assuntos

Setores

Unidades

Pessoas




uff


Universidade Federal Fluminense



Graduação



Pós-Graduação



Assuntos estudantis



"Lidei com papel praticamente a vida inteira, mas antes de me aposentar vou viver um novo tempo"