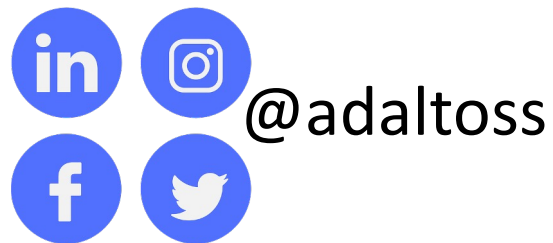


Desenvolvimento de Serviços e APIs - Aula04

Ajax

Prof. MSc. Adalto Selau Sparremerger

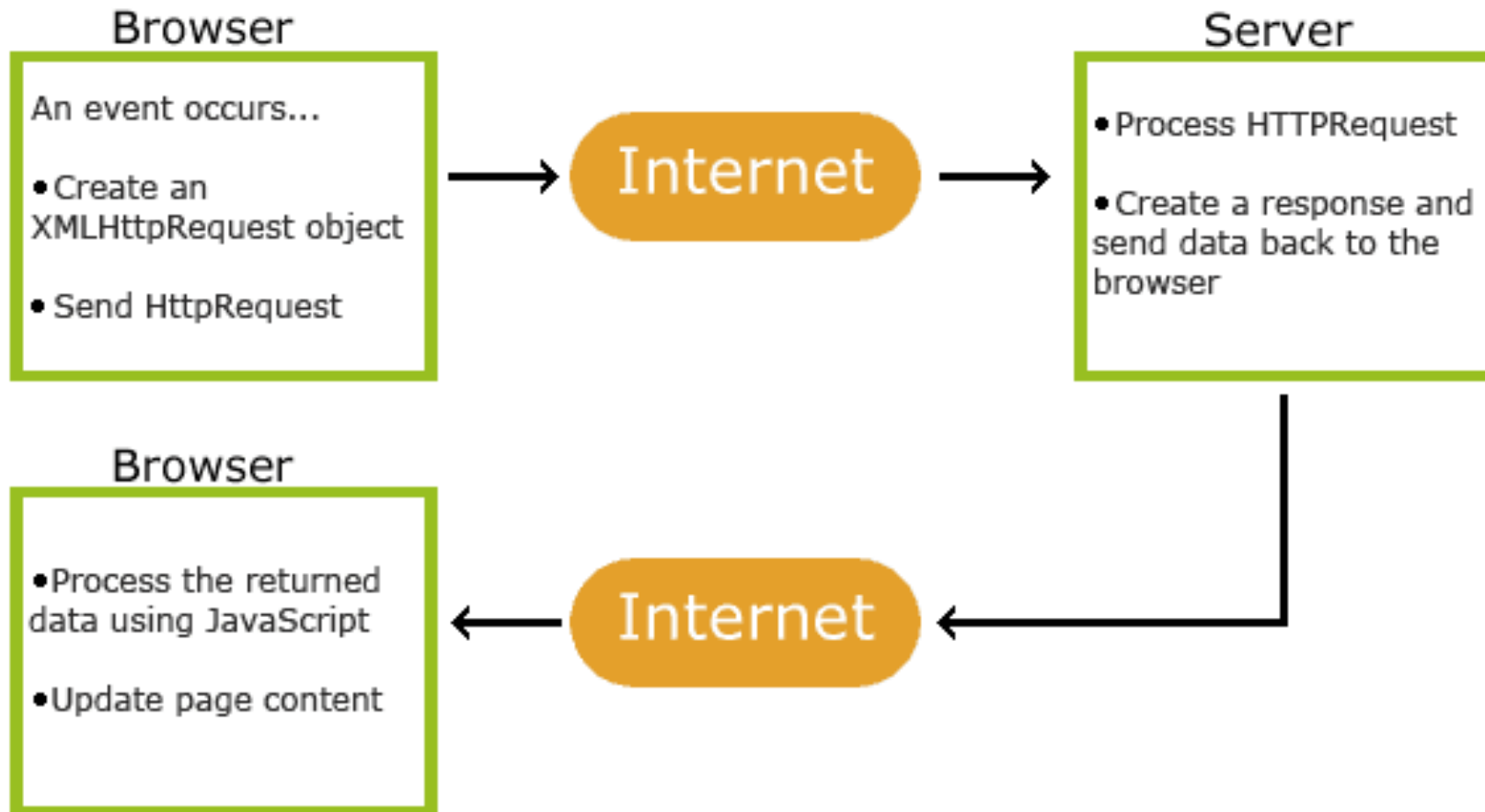
assparremerger@senacrs.com.br



Ajax

- **A**ynchronous **J**avaScript **A**nd **X**ML.
- Não é linguagem de programação
- É uma combinação de:
 - Um objeto **XMLHttpRequest** embutido no browser, para solicitar dados de um servidor
 - JavaScript e HTML DOM, para exibir ou usar os dados
- AJAX permite que páginas web, possam ser atualizadas de forma assíncrona, através da troca de dados com um servidor web nos bastidores.
- Isto significa que é possível atualizar partes de uma página web, sem recarregar a página inteira.

AJAX – Arquitetura



Fonte: W3Schols

[gtigub.com/assparremberger](https://github.com/assparremberger)

AJAX - Arquitetura

- 1. Um evento ocorre em uma página da web (a página é carregada, um botão é clicado)
- 2. Um objeto XMLHttpRequest é criado em JavaScript
- 3. O objeto XMLHttpRequest envia uma solicitação para um servidor web
- 4. O servidor processa o pedido
- 5. O servidor envia uma resposta de volta à página web
- 6. A resposta é lida pelo JavaScript
- 7. A acção adequada (como atualização da página) é executada por JavaScript

XMLHttpRequest

- O objeto XMLHttpRequest pode ser usado para trocar dados com um servidor web “por de trás dos panos” .
- Isto significa que é possível atualizar partes de uma página web, sem recarregar a página inteira.

```
var request = new XMLHttpRequest();
```

Acesso entre domínios

- Por razões de segurança, os navegadores atuais não permitem o acesso entre domínios.
- Isso significa que tanto a página web quanto o arquivo XML, devem ser localizadas no mesmo servidor.

AJAX Request

- Para enviar um pedido a um servidor, usamos os métodos **open()** e **send()** do objeto **XMLHttpRequest**:
- Ex:

```
xhttp.open("GET", "ajax_info.txt", true);  
xhttp.send();
```

AJAX - Server Response

- `readyState` apresenta o status do XMLHttpRequest.
- `onreadystatechange` define uma função a ser executada quando o `readyState` muda.
- `statusText` detém o status do objeto XMLHttpRequest.
- `onreadystatechange` é chamada cada vez que o `readyState` muda.
- Quando `readyState` é 4 e o status é de 200, a resposta está pronta

AJAX - Server Response

Propriedade	Descrição
onreadystatechange	Define uma função a ser chamada quando a propriedade readyState é alterada
readyState	Contém o status do XMLHttpRequest. 0: request não inicializado 1: conexão estabelecida com o servidor 2: solicitação recebida 3: processando a solicitação 4: solicitação concluída e resposta pronta
status	200: "OK" 403: "Forbidden" 404: "Page not found" Lista completa em: Http Messages Reference
statusText	Retorna o status-text (e.g. "OK" or "Not Found")

XMLHttpRequest.open()

Sintaxe

```
XMLHttpRequest.open(method, url)  
XMLHttpRequest.open(method, url, async)  
XMLHttpRequest.open(method, url, async, user)  
XMLHttpRequest.open(method, url, async, user, password)
```

Fonte: <https://developer.mozilla.org/pt-BR/docs/Web/API/XMLHttpRequest/open>

Exemplo

```
<div id="resultado"></div>

<button type="button" onclick="ler()">Ler Informações</button>

<script>
    function ler(){
        var xhttp = new XMLHttpRequest();
        xhttp.onreadystatechange = function() {
            if (this.readyState == 4 && this.status == 200) {
                document.getElementById("resultado").innerHTML = this.responseText;
            }
        };
        xhttp.open("GET", "informacoes.txt", true);
        xhttp.send();
    }
</script>
```

Exercício

- Construa uma função utilizando AJAX que busque no servidor o seu nome e seu e-mail, que apresenta estes dados ao usuário quando ele clica no botão VER CONTATO.