1. **Introdução**

AJAX (abreviação de "Asynchronous JavaScript + XML ") é um *framework* de desenvolvimento *web* que usa tecnologias no lado do cliente para criar aplicações *web* assíncronas. Com o AJAX, os aplicativos da *web* podem enviar e recuperar dados de um servidor de forma assíncrona (em segundo plano), sem interferir na exibição e no comportamento da página existente. Ao separar a camada de comunicação da camada de apresentação, o AJAX permite que as páginas da *Web* e, por extensão, os aplicativos da *Web,* alterem dinamicamente o conteúdo sem a necessidade de recarregar a página inteira. Na prática, implementações modernas geralmente utilizam JSON em vez de XML.

O AJAX não é uma tecnologia única, mas um grupo de tecnologias. O HTML e CSS podem ser usados ​​em conjunto para marcar e estilizar informações. A página da *Web* pode ser modificada por JavaScript para exibir dinamicamente - e permitir que o usuário interaja com - as novas informações. O objeto XMLHttpRequest interno ou, desde 2017, a nova função "*fetch*()" no JavaScript é comumente usada para executar o AJAX em páginas da *Web,* permitindo que os sites carreguem conteúdo na tela sem atualizar a página. O AJAX não é uma nova tecnologia ou linguagem diferente, apenas tecnologias existentes usadas de forma diferente.

1. **História**

No início da década de 90, a maioria dos *sites* era baseada em páginas HTML completas. Cada ação do usuário exigia que uma nova página completa fosse carregada do servidor. Esse processo ineficiente refletiu na experiência do usuário. Cada vez que o navegador recarregava uma página devido a uma alteração parcial, todo o conteúdo precisava ser reenviado, mesmo que apenas algumas informações tivessem sido alteradas. Isso colocou carga adicional no servidor e transformou a largura de banda em um fator limitante no desempenho.

Em 1996, a tag <*iframe*> foi introduzida pelo Internet Explorer como um elemento que pode carregar ou buscar conteúdo de forma assíncrona. Em 1998, a equipe do *Microsoft Outlook Web Access* desenvolveu o conceito por trás do objeto de *script* XMLHttpRequest. Ele apareceu como XMLHTTP na segunda versão da biblioteca MSXML, que foi fornecida com o *Internet Explorer* 5.0 em março de 1999.

A funcionalidade do objeto XMLHTTP ActiveX no IE 5 foi implementada posteriormente pelo *Mozilla, Safari, Opera, Chrome*, e outros navegadores como o objeto JavaScript XMLHttpRequest. A *Microsoft* adotou o modelo XMLHttpRequest nativo a partir do *Internet Explorer 7.* A versão ActiveX ainda é suportada no IE, mas não no *Microsoft Edge*. A utilidade dessas solicitações HTTP de segundo plano e tecnologias assíncronas da *Web* permaneceram bastante obscuras até que aparecessem em aplicativos *online* de grande escala, como o *Outlook Web Access* (2000) e *Oddpost* (2002).

O *Google* fez uma ampla implantação do AJAX entre navegadores, compatível com os padrões, com o *Gmail* (2004) e o *Google Maps* (2005). Em outubro de 2004, *Kayak.com* versão beta pública, foi um dos primeiros *sites* de *e-commerce* que usaram AJAX em larga escala e isso aumentou o interesse no AJAX entre os desenvolvedores de programas da *web*.

O termo AJAX foi usado publicamente em 18 de fevereiro de 2005 por Jesse James Garrett em um artigo intitulado “AJAX: Uma nova abordagem para aplicativos da Web”, com base nas técnicas usadas nas páginas do *Google*. Em 5 de abril de 2006, o *World Wide Web Consortium* (W3C) lançou o primeiro rascunho de especificação para o objeto XMLHttpRequest, na tentativa de criar um padrão oficial da *Web*. O último rascunho do objeto XMLHttpRequest foi publicado em 6 de outubro de 2016.

1. **Exemplo de utilização**

Sabemos que o AJAX é uma forma de realizar uma requisição assíncrona a um servidor utilizando os objetos XMLHttpRequest. Nesse exemplo iremos utilizar uma biblioteca muito conhecida de Javascript denominada *jQuery*.

O *jQuery* é uma biblioteca de funções JavaScript que interage com o HTML, desenvolvida para simplificar os *scripts* interpretados no navegador do cliente (*client-side*). Usada por cerca de 74.4% dos 10 mil *sites* mais visitados do mundo, *jQuery* é a mais popular das bibliotecas JavaScript. Essa biblioteca de código aberto utiliza a licença MIT em seu código-fonte. A sintaxe do *jQuery* foi desenvolvida para tornar mais simples a navegação do documento HTML, a seleção de elementos DOM, criar animações, manipular eventos, desenvolver aplicações AJAX e criação de *plugins* sobre ela. Tais facilidades permitem aos desenvolvedores criarem camadas de abstração para interações de baixo nível de modo simplificado em aplicações *web* dinâmicas de grande complexidade.

**3.1 – Download dos recursos**

Antes de qualquer coisa, é necessário adicionar o *jQuery* ao seu código. Você poderá fazer de duas formas: 1) fazer *download* do código fonte e adicioná-lo localmente ao documento HTML; 2) Adicionar o *script* usando uma versão hospedada pelo *Google*.

O *download* do *jQuery* pode ser feito aqui: <https://jquery.com/download/>. As bibliotecas hospedadas pelo *Google* ficam aqui: https://developers.google.com/speed/libraries.

O *jQuery* pode ser adicionado usando a seguinte *tag* em seu html:

<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.4.1/jquery.min.js"></script>

**3.2 – Exemplo em Javascript**

Fazendo uma requisição assíncrona usando *jQuery:*

$.ajax('suaURL',

{

dataType: 'json', // Tipo de dados que irá ser recebido

success: function (data,status,xhr) { // função de sucesso

// faça o que você quiser

},

error: function (jqXhr, textStatus, errorMessage) { // função de erro

// faça o que você quiser

}

});

O *jQuery* permite ainda que você utilize um atalho para realizar requisições do tipo *GET* e *POST.*

$.get( "suaURL", function( data ) {

// faça o que você quiser

});

$.post( "suaURL", {data: “data”}, function( data ) {

// faça o que você quiser

});