**Email 1**

Assunto: Implementação XMLHttpRequest

Olá, equipe de desenvolvimento,

Por meio deste Email gostaria de detalhar como será feita a implementação do processamento assíncrono de dados utilizando o objeto **XMLHttpRequest**, farei um passo a passo para esclarecimento do processo para que todos possam se alinhar ao fluxo de desenvolvimento.

**Passo a passo da requisição assíncrona:**

1. **Ação do usuário**: O processo começa quando o usuário realiza uma ação, como clicar em um botão ou enviar um formulário. Essa ação será o **gatilho** para iniciar a requisição assíncrona.
2. **Criação do objeto XMLHttpRequest**: Após a ação do usuário, o código criará uma nova instância do objeto XMLHttpRequest:

var xhr = new XMLHttpRequest();

1. **Configuração da requisição**: Em seguida, configuramos o tipo de requisição (GET ou POST), a URL para onde ela será enviada, e se a operação será realizada de forma assíncrona:

xhr.open('GET', 'url-api-alvo', true);

1. **Monitoramento de mudanças de estado**: Utilizamos o evento onreadystatechange para acompanhar o progresso da requisição. Cada mudança de estado do readyState representa um estágio do ciclo de vida da requisição:

xhr.onreadystatechange = function() {

if (xhr.readyState === 4 && xhr.status === 200) {

// Aqui tratamos a resposta, uma vez que a requisição foi bem-sucedida

console.log(xhr.responseText);

}

};

* **readyState =** 0: Objeto foi criado.
* **readyState** = 1: Conexão foi estabelecida.
* **readyState =** 2: A requisição foi recebida.
* **readyState** = 3: A requisição está sendo processada.
* **readyState** = 4: A requisição foi concluída e a resposta está disponível.



1. **Envio da requisição**: Após configurar a requisição e definir como vamos tratar a resposta, enviamos a requisição ao servidor:

xhr.send();

1. **Recebendo a resposta**: Quando o readyState atingir 4 e o status for 200 (ou seja, a requisição foi concluída com sucesso), o retorno será tratado. A resposta do servidor, normalmente no formato JSON, será processada conforme o contexto da aplicação.

Texto

Descrição gerada automaticamente

Figura Resumo do codigo

Interface gráfica do usuário, Texto, Aplicativo, chat ou mensagem de texto

Descrição gerada automaticamente

Figura 2Diagrama do processo

**Considerações:**

* Durante o ciclo de vida da requisição, os dados são processados sem bloquear a interface do usuário, permitindo que ele continue utilizando o sistema enquanto a requisição é processada em segundo plano.
* A manipulação das respostas e erros deve ser sempre bem tratada, para garantir a robustez da aplicação.

Por favor, certifiquem-se de revisar o código em seus módulos para garantir que todos os fluxos de requisição estão configurados corretamente. Qualquer dúvida, estou à disposição.

Atenciosamente,  
Iago Machado