

Requisições

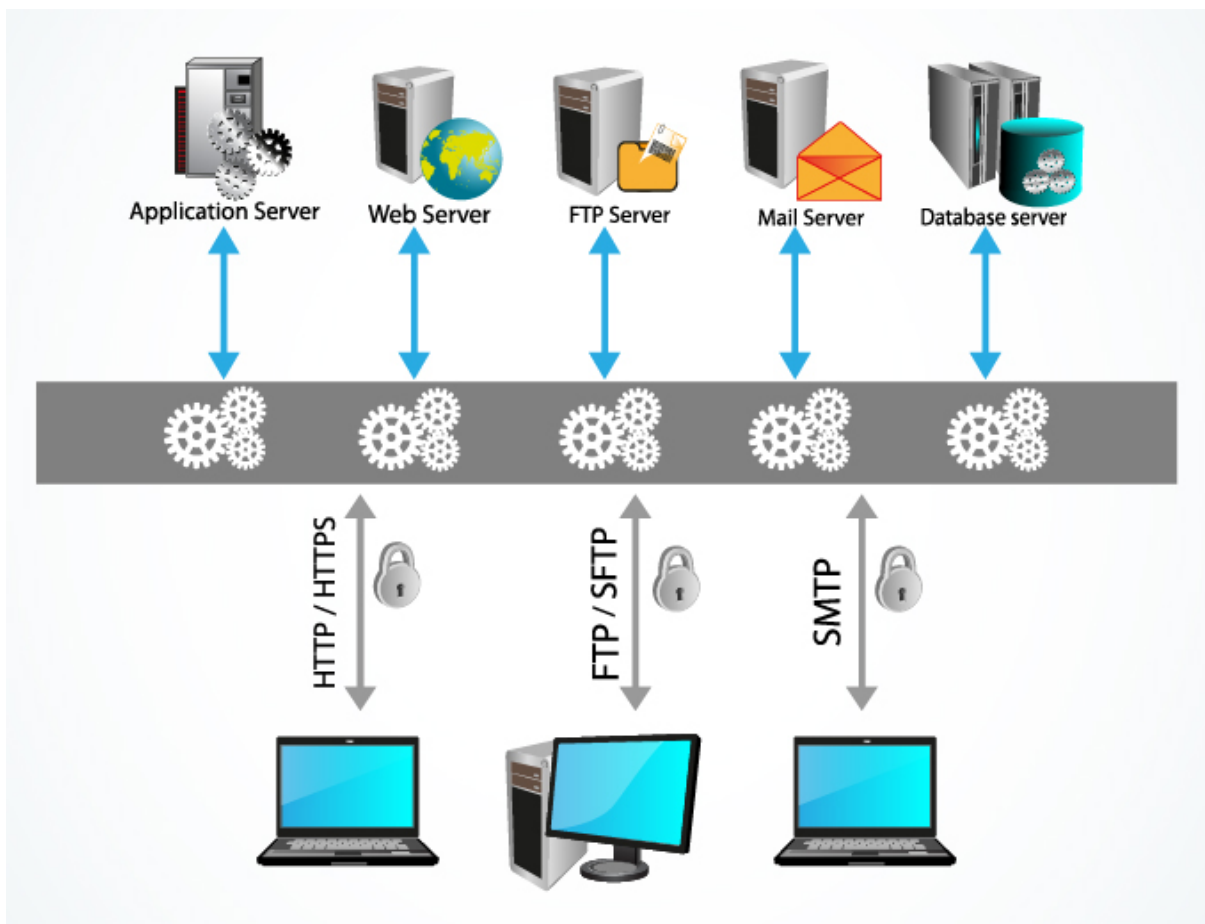
ESTUDAR REQUISIÇÕES EM JAVASCRIPT

AUTOR(A): PROF. DANIEL FERREIRA DE BARROS JUNIOR

Requisições

O caminho para o acesso a um conteúdo, ou página na internet, se dá de uma requisição do navegador, a um servidor remoto, indicado por um endereço web, ou seja, o seu navegador solicita um conteúdo a um servidor remoto. Caso ele encontre o conteúdo solicitado, o servidor devolve, entrega ao navegador (cliente), o conteúdo correspondente. No caso de não ser encontrado, algumas mensagens de erros podem ser entregues, de acordo com cada caso.

Nos casos de requisições HTTP, o conteúdo solicitado e entregue ao seu navegador geralmente fica a cargo de um servidor web, ou web server. Este servidor pode solicitar requisições internas a outros servidores, afim de buscar dados ou arquivos para completar o conteúdo solicitado. Depois que este arquivo ou conteúdo é enviado ao cliente, ou seja, ao computador que solicitou os dados, o servidor fica no aguardo de novas requisições.



Na imagem anterior, temos uma ilustração de diversos serviços, e não apenas os serviços web, disponibilizados pela internet como meio de comunicação. Cada requisição é controlada pelo servidor correspondente e transmitida de acordo com o seu protocolo de comunicação, no caso de conteúdo web o protocolo http/https.

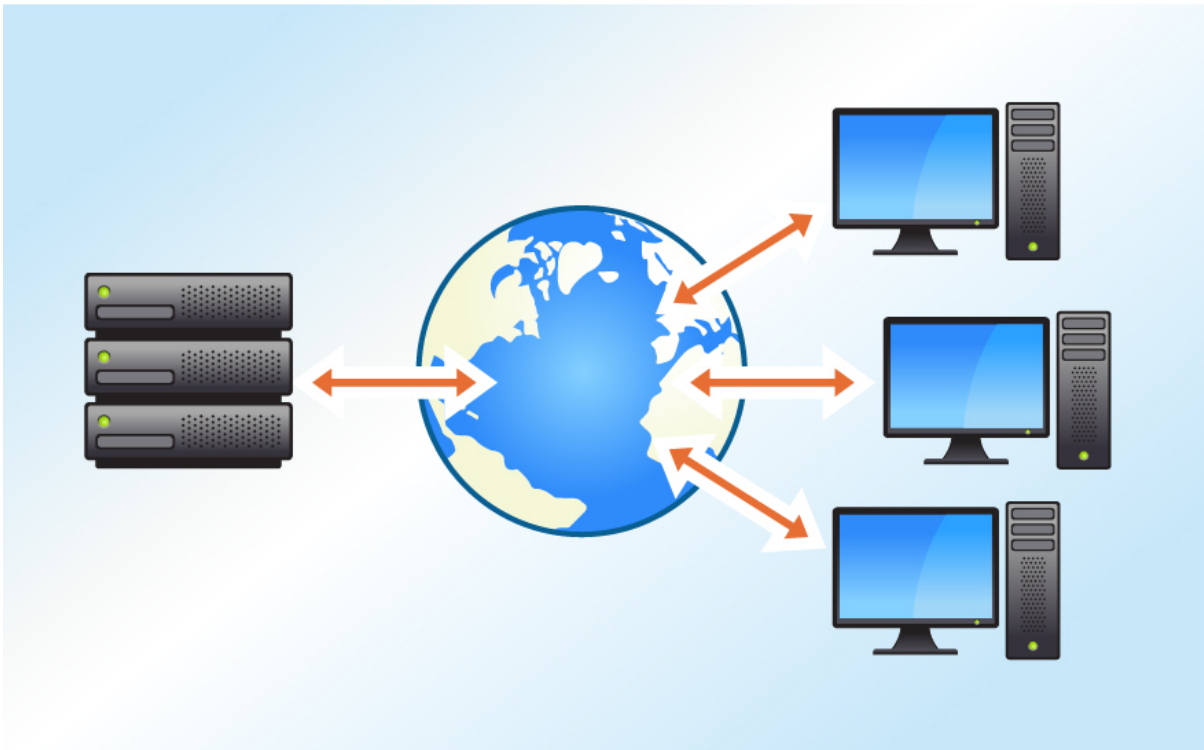
Requisição estática

As requisições web estáticas são conteúdos solicitados ao servidor e entregues por arquivos ou páginas predefinidas que não sofreram alteração de conteúdo depois de entregue. Por exemplo, a página institucional de uma empresa ou um conteúdo web, depois de entregue ao navegador, não irá ter novos dados ou elementos a não ser que seja dado um refresh no navegador, ou uma nova requisição ao servidor, recarregando a página em questão.

Este procedimento torna o processo de navegação mais lento uma vez que é necessário aguardar o novo carregamento da página inteira enviado pelo servidor.

Requisição dinâmica

As requisições web dinâmicas possibilitam que um conteúdo solicitado a um servidor obtenha novos dados ou elementos sem a necessidade de um novo carregamento do navegador. Em outras palavras, é possível obter mais conteúdo a página web sem a necessidade de atualização da página, o que torna o processo mais rápido e agradável ao usuário, principalmente devido ao menor tempo de resposta.



Requisição dinâmica síncrona

Com o uso do JavaScript é possível solicitar novas requisições de conteúdo, sem a necessidade de realizar o refresh da página, ou recarrega-la no navegador.

No entanto, o carregamento síncrono é sensível ao desempenho de resposta do servidor. Caso um servidor demore a responder ou entregar os dados o navegador do cliente continuará no aguardo do seu retorno, o que pode parecer um “travamento” da página se houver mais elementos a serem carregados na página solicitada.

Enquanto a requisição atual não for atendida e entregue ao cliente, mesmo que seja com erro, as demais requisições não serão respondidas ou entregues, o que pode acarretar maior tempo de espera ao usuário

Os tempos de resposta do servidor são sensíveis a toda a infraestrutura web, desde a velocidade de conexão de dados, captura e entrega de informação no servidor, tempo de resposta ou sobrecarga do servidor, latência da rede de dados, entre outros.

Requisição dinâmica assíncrona

O mesmo script JavaScript pode determinar o uso de uma requisição dinâmica de forma assíncrona.

Nesta forma, a requisição de certa maneira fragmenta as respostas e status de cada solicitação ao servidor e trata cada informação de uma forma específica, com o intuito de tratar, otimizar e até mesmo evitar o travamento no carregamento da página web, caso esteja bem codificado.

Caso ocorra mais de uma requisição ao servidor, a resposta pode ser dada em paralelo, ou assim que estiverem disponíveis ao cliente. Essa resposta em paralelo de requisições diminui o tempo de espera do usuário, uma vez que a disponibilidade da solicitação se dá em menos tempo.

Uma maneira de se trabalhar de forma assíncrona é com o uso de AJAX, que significa Asynchronous JavaScript e XML. O mesmo utiliza o objeto XMLHttpRequest para realizar a comunicação com o servidor a fim de tornar a comunicação assíncrona possível.

Veremos o AJAX e o XMLHttpRequest em detalhes no próximo tópico.

DICA:

Para saber mais, veja em:

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

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

https://www.w3schools.com/xml/ajax_xmlhttprequest_send.asp

(https://www.w3schools.com/xml/ajax_xmlhttprequest_send.asp)

Conclusão

As páginas web podem ser construídas de diferentes formas, com diferencial para as requisições aos servidores. Podemos ter requisições:

- Estáticas.
- Dinâmicas síncronas.
- Dinâmicas Assíncronas.

Avalie a correta necessidade e escopo da construção, de cada parte, de suas páginas HTML. Uma escolha inadequada pode trazer prejuízos ou muitos elogios de seus clientes!

ATIVIDADE

Sobre páginas estáticas, escolha uma alternativa:

- A. São páginas HTML que se alteram a cada consulta no banco de dados.
- B. São páginas HTML que se alteram a cada requisição
- C. São páginas HTML que se mantem a mesma a cada requisição.
- D. São páginas HTML que as vezes se alteram a cada requisição.

ATIVIDADE

Sobre a afirmação:

"É possível obter mais conteúdo a página web sem a necessidade de atualização da página?".

Qual alternativa corresponde a esta frase?

- A. Requisição estática
- B. Requisição dinâmica
- C. Requisição automática
- D. Requisição nula

ATIVIDADE

Complete a frase:

"o carregamento _____ é sensível ao desempenho de resposta do servidor. "

- A. síncrono
- B. assíncrono
- C. estático

D. rápido

REFERÊNCIA

MORRISON, M. Use a cabeça JavaScript. 5º Ed. Rio de Janeiro: Alta Books, 2012. 606 p.

OLIVIERO, C. A. J. Faça um site JavaScript orientado por projeto. 6º ed. São Paulo: Érica, 2010. 266 p.

ZAKAS, Nicholas C. JavaScript de alto desempenho. 8º Ed. São Paulo: Novatec, 2010. 245 p.

