Uma imagem contendo comida, desenho

Descrição gerada automaticamente

Faculdade de Tecnologia de Sorocaba

Tecnologia em Análise e Desenvolvimento de Sistemas

**PROGRAMAÇÃO WEB**

*Single Page* e *Multiple Page* *Applications*

Eduardo Steiner Hessel 0030481913009

SOROCABA

Fevereiro – 2021

# **1. Introdução**

Atualmente, a internet é praticamente essencial na vida de todos. Com o advento dos dispositivos móveis com acesso à rede e que permanecem sempre conectados a ela, as pessoas podem ter na palma de suas mãos, um aparelho capaz de entregar qualquer informação que elas desejarem. Porém, apesar da tecnologia ter evoluído muito, especialmente com o advento do 5G, a demanda por performance dos sites aos quais navegamos na internet é enorme, especialmente em um mundo onde se tem cada vez menos tempo para fazer qualquer coisa, portanto segundos podem fazer a diferença para que se conquiste um novo cliente ou um novo acesso.

Além disso, a experiência do usuário também é bastante importante, pois cada vez mais pessoas leigas em tecnologia acessam sites na internet, e a interface e a performance devem ser agradáveis para facilitar a entrada dessas pessoas no mundo virtual. O conceito de SPAs vem para ajudar a aprimorar essas áreas e trazer outros benefícios

# **2. Single Page Applications e Multiple Page Applications**

Uma *Single Page Application*, também chamada pela sigla SPA, é um aplicativo Web, ou um site da Web, que interage com o usuário reescrevendo dinamicamente a página atual em vez de carregar páginas novas inteiras de um servidor. O grande protagonista de uma SPA é o código Javascript executado no cliente. Toda a aplicação pode ser construída simplesmente manipulando o Document Object Model (DOM, modelo padronizado para representação com objetos em documentos HTML), de forma nativa, ou com o uso de bibliotecas e frameworks Javascript que auxiliam na construção da aplicação.

Estas bibliotecas e frameworks fornecem recursos para manipulação dinâmica do DOM, definição de design de tela, chamadas assíncronas ao servidor, organização do código JavaScript, dentre outros. Há alguns anos, essa arquitetura seria totalmente inviável por causa do pouco poder de processamento do lado do cliente. O poder de processamento do computador pessoal era limitado e a maturidade com o JavaScript ainda era bem pouca.

Estávamos em um cenário onde o poder de processamento estava mais concentrada nas mãos das empresas, ou seja, uma época em que era totalmente lógico todo o processamento ser feito pelo servidor e apenas o resultado ser entregue para o cliente. Apenas com a evolução conjunta do computador pessoal e as melhorias realizadas na linguagem JavaScript, o cenário mudou de figura e acabou atingindo diretamente aplicações web, desktop e até o mobile.

Nesse novo momento, chegamos em aplicações front-end totalmente desacopladas, fazendo o uso do poder do processamento ocioso do computador do cliente. Isso permitiu também um back-end centralizado, responsável por manter toda a lógica da aplicação, por fazer integrações, e servir dados para diferentes plataformas. Ou seja, temos hoje a possibilidade de desenvolver soluções especialistas em suas plataformas, com a regra de negócio centralizado em apenas um determinado ponto, com processamento menor e distribuído, por se aproveitar do processamento da máquina do cliente.

**2.1. Ciclo de vida de uma SPA**

Em um modelo SPA, o cliente faz uma requisição para o servidor de conteúdo (uma CDN — Content Delivery Network, por exemplo), que apenas serve o conteúdo. A página é retornada e posteriormente toda a comunicação posterior é feita via AJAX (JavaScript Assíncrono com XML), com servidor provendo partes ou até conteúdos inteiros da página. Assim, a página no cliente processa esses dados que foram fornecidos pelo servidor, neste caso, havendo uma interação bem maior no dispositivo do usuário.

Como toda aplicação, é necessário também a consulta de dados e integrações com diferentes serviços. Em uma SPA, um back-end a parte faz esse serviço. A comunicação é feita via requisição HTTP ou WebSocket, onde o back-end faz as integrações com banco de dados, apis internas ou outros serviços, e devolve essas informações em formato JSON para o cliente.

**2.2. Multiple Page Applications e vantagens e desvantagens em relação às SPAs**

As *Multiple Page Applications* é o método que se utiliza para construir aplicações Web desde o princípio. Nela uma linguagem de programação, como por exemplo o PHP, ASP.NET, JSP, entre outros, o cliente faz uma requisição ao servidor (necessariamente um servidor HTTP, como por exemplo o Apache, IIS, dentre outros), e retorna a primeira página desta aplicação, sendo posteriormente toda resposta do servidor uma página HTML totalmente já resolvida. Ou seja, a cada nova página selecionada em um menu, o navegador faria uma requisição ao servidor, que devolveria uma página HTML completa.

Resumidamente, a página era alimentada e processada pelo servidor (ou seja, um acoplamento maior do lado do servidor, não sendo tecnicamente viável a utilização por outras frentes), e retornada só o HTML da página para o cliente, havendo então a atualização da página.

As vantagens da abordagem SPA são várias, especialmente no quesito experiência do usuário e performance, além de facilitar o desenvolvimento de aplicações, desacoplando todas elas e tornando-as mais específicas à sua funcionalidade, ainda que dependam umas das outras. Comparando isso à dificuldade de manutenção das MPA, que acaba por ficar confusa por misturar muito back-end com front-end, as SPA ajudam e muito os desenvolvedores.

As SPAs possuem também algumas desvantagens quando comparadas às MPAs. A curva de aprendizado para novos desenvolvedores pode ser um pouco complexa de início quando comparada às décadas de informações que existem a respeito das aplicações de múltiplas páginas. A difícil integração com os motores de busca devido ao fato da aplicação ser renderizada totalmente pelo JavaScript, o que faz com que os robôs do SEO que varrem os documentos HTML procurando por informações e tags não as encontrem.

# **3. Conclusão**

As *Single Page Applications* são o futuro do desenvolvimento web, facilitando a vida de todos os lados, tanto de performance e experiência do usuário, tornando as aplicações web mais vivas e interativas, quanto à facilidade de criação e manutenção por parte dos desenvolvedores. Muitas empresas passaram a adotar essas tecnologias por conta dessas vantagens, apesar de algumas desvantagens ainda existirem. Assim que for resolvida a situação com os motores de busca, as aplicações de página única se tornarão padrão de mercado.

# **4. Referências bibliográficas**

YAMAMOTO JÚNIOR, Titao. Entendendo SPA — Aplicativo de Página Única. 2019. Disponível em: https://medium.com/@titaoyamamoto/entendendo-spa-ad69236ebd51. Acesso em: 17 fev. 2021.

CAVALCANTE, Johnny. Descomplicando SPA’s. 2019. Disponível em: https://medium.com/trainingcenter/descomplicando-spas-caa8f57bdbf3. Acesso em: 17 fev. 2021.

Aplicativo de página única. Disponível em: https://pt.wikipedia.org/wiki/Aplicativo\_de\_p%C3%A1gina\_%C3%BAnica. Acesso em: 17 fev. 2021.

Modelo de Objeto de Documentos. Disponível em: https://pt.wikipedia.org/wiki/Modelo\_de\_Objeto\_de\_Documentos. Acesso em: 17 fev. 2021.