

Vue.js



Katia Jaqueline Tartari SP3013251

O que é o VueJS

- Vue é um framework JavaScript de código-aberto, focado no desenvolvimento de interfaces de usuário e aplicativos de página única (SPAs).
- Esse framework conta com uma arquitetura que pode ser adotada de forma incremental, e que foca na renderização declarativa e na composição de componentes, ou seja, divide-se em diversas partes que podem ser facilmente encaixadas umas nas outras.
- Funcionalidades avançadas necessárias para aplicações complexas são oferecidas por meio de bibliotecas e pacotes de suporte mantidos oficialmente, sendo o Nuxt.js como uma das soluções mais populares.



História

- O Vue foi criado pelo desenvolvedor independente chinês Evan You, após trabalhar para o Google.
- O primeiro commit ao código-fonte do projeto foi em julho de 2013, e o framework foi lançado oficialmente no ano seguinte, em fevereiro de 2014.



Principais Características - Componentes

- Os componentes Vue estende elementos HTML básicos para encapsular código reutilizável.
- A um alto nível, os componentes são elementos personalizados aos quais o compilador do Vue associa determinados comportamentos.
- No Vue, um componente é essencialmente uma instância do Vue com opções predefinidas.



Principais Características - Modelos

- O Vue usa uma sintaxe de modelo baseada em HTML que permite vincular o DOM renderizado aos dados contidos na instância subjacente do Vue.
- Todos os modelos do Vue são HTML válido, e podem ser interpretados por navegadores compatíveis e analisadores HTML.
- Compilando os modelos, o Vue transforma-os em funções de renderização do DOM virtual.
- O Modelo de Objeto de Documento (ou "DOM") virtual permite ao Vue renderizar componentes internamente antes de atualizar o navegador.
- Combinado com o sistema de reatividade, consegue calcular o número mínimo de componentes a serem re-renderizados, e manipular minimamente o DOM quando o estado do aplicativo é alterado.



Principais Características -

Reatividade

- O sistema de reatividade do Vue utiliza objetos JavaScript simples e re-renderização otimizada.
- Cada componente controla suas próprias dependências reativas durante sua renderização, com o objetivo de que o sistema saiba exatamente quando e quais componentes renderizar novamente.



Principais Características -

Transições

- O Vue permite diversas maneiras de aplicar efeitos de transição quando itens são inseridos, atualizados ou removidos do DOM.
- Quando um elemento envolto em um componente de transição é inserido ou removido, o que acontece é:
- O Vue detecta automaticamente se o elemento-alvo tem transições ou animações CSS aplicadas; caso em que as classes de transição CSS serão adicionadas ou removidas nos momentos apropriados.
- Se o componente de transição fornecer ganchos JavaScript, esses ganchos serão chamados nos momentos apropriados.
- Se nenhuma transição/animação CSS for detectada e nenhum gancho JavaScript for fornecido, as operações do DOM para inserção e/ou remoção serão executadas imediatamente no próximo frame.



Principais Características - Encaminhamento

- O Vue oferece uma interface para alterar o que é exibido na página com base no caminho atual da URL - independentemente de como ela foi alterada.
- Além disso, o uso de encaminhamento front-end permite a transição intencional do caminho do navegador quando certos eventos ocorrem em botões ou links.
- O próprio Vue não vem com roteamento de hash front-end. Mas o pacote de código-aberto "vue-router" fornece uma API para atualizar a URL do aplicativo . Suporta ainda o mapeamento de rotas aninhadas para componentes aninhados e oferece controle de transição preciso.



Ferramentas Oficiais

- **Devtools** - extensão devtools do navegador para depuração de aplicativos Vue.js
- **Vue CLI** - Ferramentas padrão para desenvolvimento rápido com Vue.js
- **Vue Loader** - um loader webpack que permite a escrita de componentes Vue em um formato chamado SFC (Single-File Components)



Bibliotecas Oficiais

- **Vue Router** - O *router* oficial para o Vue.js
- **Vue CLI** - Ferramentas padrão para desenvolvimento rápido com Vue.js
- **Vuex** - Gerenciamento centralizado de estado inspirado em fluxo para Vue.js



Iniciar um projeto VueJS - CDN

- Usar o CDN (Content Delivery Network), que é uma rede de distribuição de conteúdo.
- Onde temos uma rede de servidores que armazenam cópias de códigos de outros sites na memória e depois devolvem esse conteúdo baseando-se na localização geográfica, para conectá-los ao servidor mais próximo e mais rápido, reduzindo o tempo de transferência de dados.
- Dependendo do caso, usar CDN talvez não seja a melhor opção, pois as vezes o servidor pode estar indisponível, afetando na sua produtividade.



Iniciar um projeto VueJS - NPM

- **Fazer a instalação como uma dependência do NPM e importá-la para o projeto, fazendo o seu uso normalmente.**
- **Sendo uma opção viável, pois há um controle maior sobre tudo o que é utilizado no projeto.**



Iniciar um projeto VueJS - Vue-CLI

- Usar o gerenciador de templates vue-cli.
- Sendo possível baixar e iniciar projetos com toda a configuração já pronta, facilitando todo o processo de construção do projeto.



Primeira Aplicação com VueJS - Hello World

- Arquivo HTML com a estrutura inicial do Vue.js:

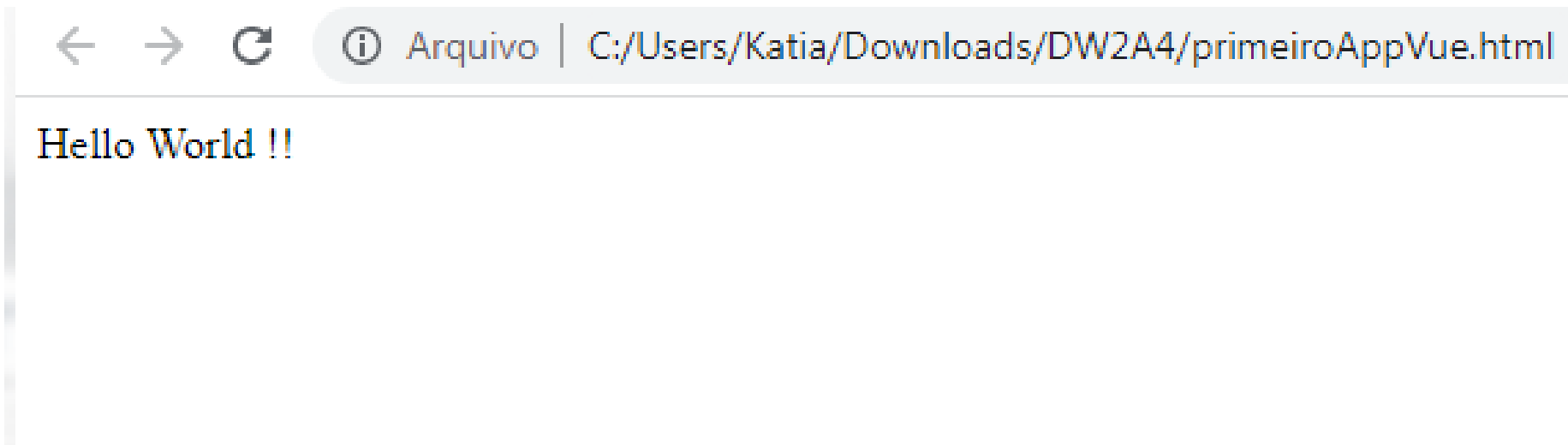
C: > Users > Katia > Downloads > DW2A4 > <> primeiroAppVue.html > ...

```
1  <html lang="en">
2
3    <div id="app">{{mensagem}}</div>
4
5    <script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"></script>
6
7    <script>
8
9      new Vue({
10        el: '#app',
11        data: {
12          mensagem: "Hello World !!"
13        }
14      });
15
16    </script>
17
18  </html>
19
```



Primeira Aplicação com VueJS - Hello World

- Até o momento não criei nada muito visual, tanto que se abrir o arquivo irá visualizar algo semelhante a pagina abaixo com a mensagem 'Hello World':



Primeira Aplicação com VueJS - Entendimento da estrutura

- Entendendo a estrutura, temos a mais enxuta página possível para entendimento. Há um elemento que irá renderizar a aplicação e nele estarão tudo o que for injetado na função `Vue()`.
- No exemplo, foi definido os dados no atributo `data()` e foi criado uma propriedade mensagem, para receber o valor a ser exibido.



Referências

- **Wikipédia, Vue.js. Disponível em: <<https://pt.wikipedia.org/wiki/Vue.js>> . Acesso em: 03 nov. 2021.**
- **Medium, Iniciando com Vue.js: criando um projeto do zero - Thaís Ribeiro. Disponível em: <<https://medium.com/@thais.ribeiro/iniciando-com-vue-js-criando-um-projeto-do-zero-b62e159ca1a7>>. Acesso em: 01 nov. 2021.**
- **FS Developer, Criando sua primeira aplicação com o Vue.js - Luiz Fernando Santana. Disponível em: <<https://www.fsdeveloper.com.br/blog/criando-sua-primeira-aplicacao-com-o-vue-js/>> .Acesso em: 01 nov. 2021.**