

Desenvolvimento Web III

Vue.JS

Introdução

Embora seja possível criar aplicativos Vue.js usando apenas JavaScript, a maioria dos desenvolvedores precisa de mais potência e flexibilidade.

Usando a CLI do Vue e componentes de arquivo único, você pode aprimorar a experiência de desenvolvimento.

Instalação

Para instalar a ferramenta vue-client devemos utilizar o comando abaixo:

```
npm install -g @vue/cli
```

Obs.: O computador deve ter o node instalado, com o gerenciador de pacotes(npm) instalado também.

Criando o projeto

- Para criar um projeto usando o Vue-CLI é necessário usar prompt de comando(terminal).

`vue create projeto-teste`

- Escolher a configuração do projeto e esperar a conclusão do processo
- Abrir a pasta do projeto

`cd projeto-teste`

- Rodar o servidor

`npm run serve`

Organização do projeto

```
> node_modules  
> public  
> src  
◆ .gitignore  
ℬ babel.config.js  
{ } jsconfig.json  
{ } package-lock.json  
{ } package.json  
ⓘ README.md  
JS vue.config.js
```

Arquivo .vue

```
1 <template>
2   <div id="app">
3     
4     <HelloWorld msg="Welcome to Your Vue.js App"/>
5   </div>
6 </template>
7
8 <script>
9   import HelloWorld from './components/HelloWorld.vue'
10
11   export default {
12     name: 'App',
13     components: {
14       HelloWorld
15     }
16   }
17 </script>
18
19 <style>
20 #app {
21   font-family: Avenir, Helvetica, Arial, sans-serif;
22   -webkit-font-smoothing: antialiased;
23   -moz-osx-font-smoothing: grayscale;
24   text-align: center;
25   color: #2c3e50;
26   margin-top: 60px;
27 }
28 </style>
29
```

Manuseando o arquivo .vue

```
1  <template>
2    <div id="app">
3      <p>{{ titulo }}</p>
4    </div>
5  </template>
6
7  <script>
8
9    export default {
10      name : 'App',
11      data: function() {
12        return {
13          titulo: 'Teste data usando vue'
14        }
15      }
16    }
17  </script>
18
19  <style>
20    #app {
21      background-color: #beige;
22    }
23  </style>
24
```

Fazendo o build

- Para gerar o arquivo de distribuição, devemos usar o seguinte comando:

```
npm run build
```


Adicionando plugins

- Os plugins podem ser adicionados facilmente nos projetos Vue.

`vue add vuetify`

Componente global

Para que seja possível compartilhar estados de componentes da nossa aplicação, devemos criar componentes globais que compartilham seus estados(Atributos).

```
Vue.component('contador', {  
  Template: `<div>  
<span>{{ contador }}</span></div>`,  
  data(){  
    return {  
      contador: 0  
    }  
  }  
})
```

Componente local

```
const contador = {  
  template: `<div>  
<span>{{ contador }}</span></div>`,  
  data(){  
    return {  
      contador: 0  
    }  
  }  
}
```

```
new Vue({  
  el: '#app1',  
  components: {  
    contador1: contador  
  }  
})
```