

Padrão de Projetos

DEVinHouse

Parcerias para desenvolver a sua carreira

SENAI

<LAB365>

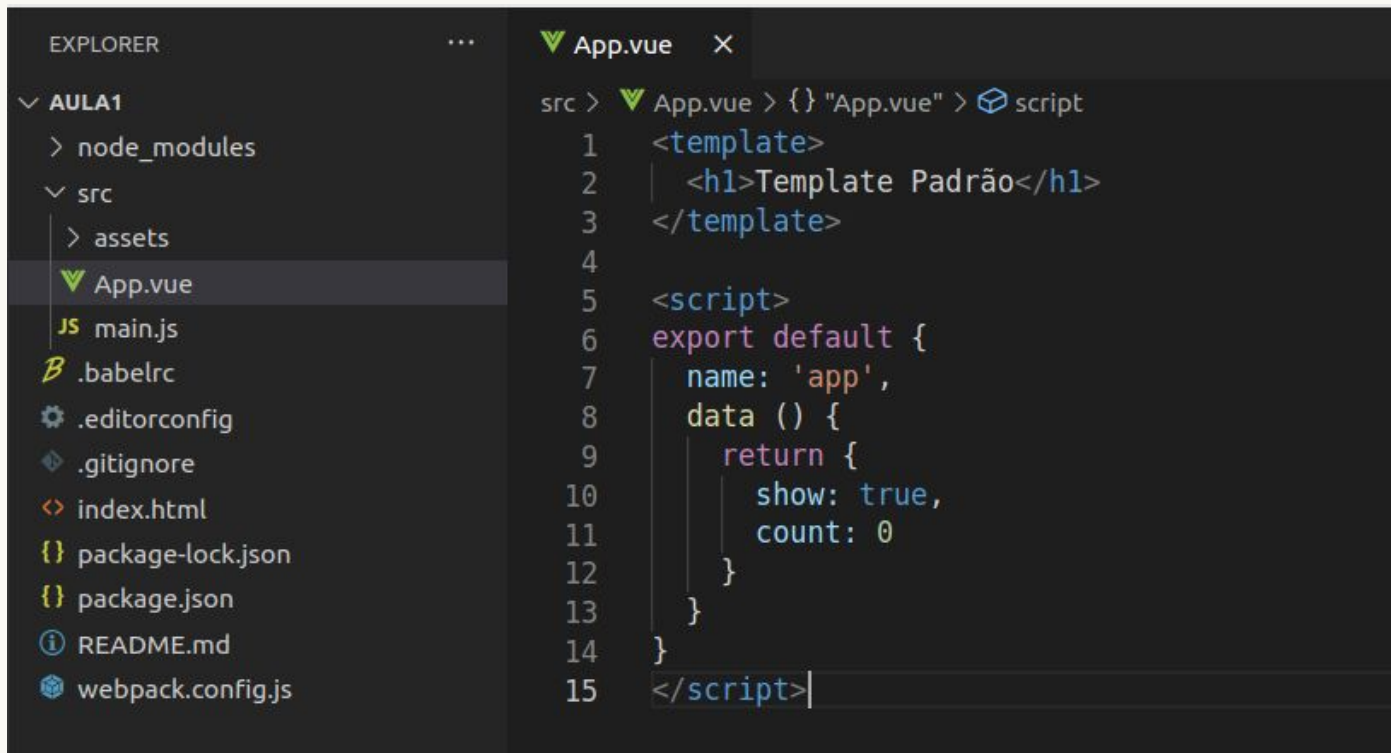
AGENDA DA SEMANA

- Transition
- Transition Group
- VueRouter
- Vuex e gerenciamento de estado
- **Padrão de projetos**

A estrutura básica de um projeto Vue, ao ser iniciada é composta pelos seguintes arquivos e diretórios:

- **node_modules:** Diretório com os módulos externos instalados de nossa aplicação.
- **public:** Arquivos de acesso público de nosso sistema (Algumas vezes o index poderá estar presente neste diretório).
- **src:** Diretório principal que se encontrará os códigos da aplicação.
- **index.html:** Página que irá renderizar ao abrir nosso sistema ao efetuar a abertura pelo path principal.
- **package.json:** Configuração do projeto e dependências.

Estrutura Inicial



```
EXPLORER
└─ AULA1
  └─ node_modules
  └─ src
    └─ assets
    └─ App.vue
    └─ main.js
    └─ .babelrc
    └─ .editorconfig
    └─ .gitignore
    └─ index.html
    └─ package-lock.json
    └─ package.json
    └─ README.md
    └─ webpack.config.js

App.vue
src > App.vue > {} "App.vue" > script
1  <template>
2  |   <h1>Template Padrão</h1>
3  </template>
4
5  <script>
6  export default {
7    name: 'app',
8    data () {
9      return {
10        show: true,
11        count: 0
12      }
13    }
14  }
15 </script>
```

Como dito anteriormente, o SRC ficará o código fonte da nossa aplicação. Nela se encontrará inicialmente:

- assets: Pasta de arquivos que usaremos na nossa aplicação como: imagens, folha de estilo e etc.
- App.vue: Componente inicial do Vue.
- main.js: Arquivo de configuração e inicialização do Vue.

Existem algumas formas indicadas para trabalharmos com o Vue, principalmente na separação dos componentes do sistema. Quando falamos de SPA podemos distinguir os componentes em duas categorias: Telas do sistema e componentes de uso comum. As telas são as páginas mapeadas no router, que será montada dependendo do acesso do usuário enquanto os componentes de uso comum são peças que utilizaremos nessas telas separadas em componentes para reaproveitarmos em nossos sistemas.

Padrão de Projeto

```
> node_modules
└─ src
  └─ assets
  └─ components
    └─ navbar
    └─ sidebar
    └─ table
      └─ MyTable.vue
  └─ views
    └─ customer
      └─ CustomerDetails.vue
      └─ CustomersList.vue
    └─ user
  └─ App.vue
  └─ JS main.js
  └─ .editorconfig
  └─ index.html
  └─ package.json
```

```
> node_modules
└─ src
  └─ assets
  └─ components
    └─ navbar
    └─ sidebar
    └─ table
      └─ MyTable.vue
  └─ pages
    └─ customer
      └─ CustomerDetails.vue
      └─ CustomersList.vue
    └─ user
  └─ App.vue
  └─ JS main.js
  └─ .editorconfig
  └─ index.html
  └─ package.json
```

```
> node_modules
└─ src
  └─ assets
  └─ components
    └─ pages
      └─ customer
        └─ CustomerDetails.vue
        └─ CustomersList.vue
      └─ user
    └─ shared
      └─ navbar
      └─ sidebar
    └─ table
      └─ MyTable.vue
  └─ App.vue
  └─ JS main.js
  └─ .editorconfig
  └─ index.html
  └─ package.json
```

Router

Como padrão, a utilização do Router é separada através de um novo diretório no src chamada "router" contendo primeiramente o index.js (que será responsável por exportar o objeto router), e arquivo de listagem de rotas:

```
✓ src
  > assets
  ✓ router
    JS index.js
    JS routes.js
```

```
import { createRouter, createWebHashHistory } from 'vue-router';
import routes from './routes';
```

```
const router = createRouter({
  routes,
  history: createWebHashHistory()
});
```

```
export default router;
```

```
export default [
  { path: '/', component: Home }
]
```


Assim como no router, para o VueX criaremos um novo diretório chamado “store” contendo o index.js seguido dos arquivos de exportação de módulos, actions, mutations e getters

```
✓ store
  ✓ modules
    JS cron.js
    JS actions.js
    JS index.js
    JS mutations.js
```

```
import { createStore } from 'vuex'
import actions from './actions';
import mutations from './mutations';
import cron from './modules/cron';

export default createStore({
  modules: {
    cron
  },
  actions,
  mutations
})
```



DEVinHouse

Parcerias para desenvolver a sua carreira

OBRIGADO!



<LAB365>