

# ESCOLA E FACULDADE DE TECNOLOGIA SENAI ROBERTO MANGE

## DESENVOLVIMENTO DE SISTEMAS

JULHO DE 2022 — SENAI-SP



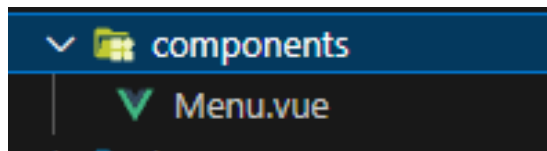
ESCOLA SENAI "ROBERTO MANGE"  
UM. RECONHECIDA NA INDÚSTRIA

**SENAI**  
SÃO PAULO

## CRIANDO COMPONENTES

- ✨ Componentes são uma eficiente maneira de reutilização de código, evitando repetições desnecessárias de código!
- 📌 Sempre que encontrar uma parte de sua aplicação que pode ou deve ser usada várias vezes, e que agrupa algumas responsabilidades definidas, é um dos indícios de que pode criar um componente para essa necessidade.

1) Para criar um componente no Nuxt, basta criar uma pasta chamada 'components', e criar os seus componentes com a primeira letra maiúscula.



2) Aqui neste exemplo criaremos um Menu visto que seu código não tende a mudar e se for usado em várias telas sem ser um componente irá ocasionar a repetição de código desnecessária.

```
Menu.vue
components > Menu.vue > ...
1  <template>
2    <div>
3      <NuxtLink to="/">Home</NuxtLink> <br>
4      <NuxtLink to="/trips">Trips</NuxtLink> <br>
5      <NuxtLink to="/booking">Booking</NuxtLink> <br>
6      <NuxtLink to="/login">Login</NuxtLink> <br>
7    </div>
8  </template>
```

- 📌 Note que é possível que um componente use outros componentes dentro de si, este é o caso do nosso Menu, onde usamos o componente padrão do Nuxt chamado NuxtLink, que basicamente serve para redirecionar para as páginas desejadas.



## CRIANDO COMPONENTES

3) Para usar o componente criado basta coloca-lo como uma tag html usando seu nome:

```
<template>
  <div>
    <Menu></Menu>
  </div>
</template>
```



## CRIANDO LAYOUTS

✨ Layouts também possuem o conceito de reutilização de código, porém em larga escala. Por exemplo, caso perceba que uma parte do código vai estar presente em todas ou quase todas as páginas, então é um exemplo que poderia criar um layout.

📌 A ideia do layout é fazer com que as páginas (que seria o conteúdo variável) esteja dentro dele.

📌 É possível utilizar mais de um layout por aplicação, mas para isso você tem que definir nas páginas qual o layout usado.

1) Primeiramente para habilitar o uso de layouts na aplicação devemos garantir que no App.vue esteja a tag NuxtLayout:

```
app.vue > ...  
1  <template>  
2    <div>  
3      <NuxtLayout>  
4        <NuxtPage />  
5      </NuxtLayout>  
6    </div>  
7  </template>
```

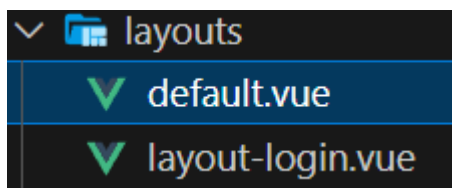
📌 Perceba que as páginas do Nuxt formada pelo NuxtPage estão dentro do NuxtLayout, tornando a ideia de que um Layout abrigará sempre uma página.

## CRIANDO LAYOUTS

2) Crie uma pasta chamada layouts: 

3) Por padrão o Nuxt utiliza “por baixo dos panos” um layout chamado default, então recomendo que crie o default.vue e coloque nele o código que deseja reaproveitar para a maior parte das páginas, pois a vantagem de utilizá-lo é que caso queira usá-lo em um página não é necessário fazer nenhum tipo de definição na mesma, pois o Nuxt já entende em cada página que se não for definido qual layout usar então ele utilizará o default.vue.

Sendo assim, criaremos neste exemplo dois layouts, um denominado default.vue para conter o logo da aplicação e o Menu, e um outro chamado layout-login.vue para ser usado nas páginas que não se deseja mostrar o Menu (exemplo na tela de login)



## CRIANDO LAYOUTS

4) Dentro do default.vue vamos utilizar o componente criado anteriormente de Menu e colocar a imagem logo da aplicação.

```
<template>
  <div>
    
4     <slot></slot>
5   </div>
6 </template>
```

6) Como este layout não é o default, então devemos informar nas páginas que queremos que o utilize para poder funcionar:

```
index.vue
pages > login > index.vue > ...
1 <script setup>
2   definePageMeta({
3     layout: "layout-login"
4   });
5 </script>
6
7 <template>
8   <div>
9     <h1>Login Page!</h1>
10  </div>
11 </template>
```