Dispositivos Portáteis

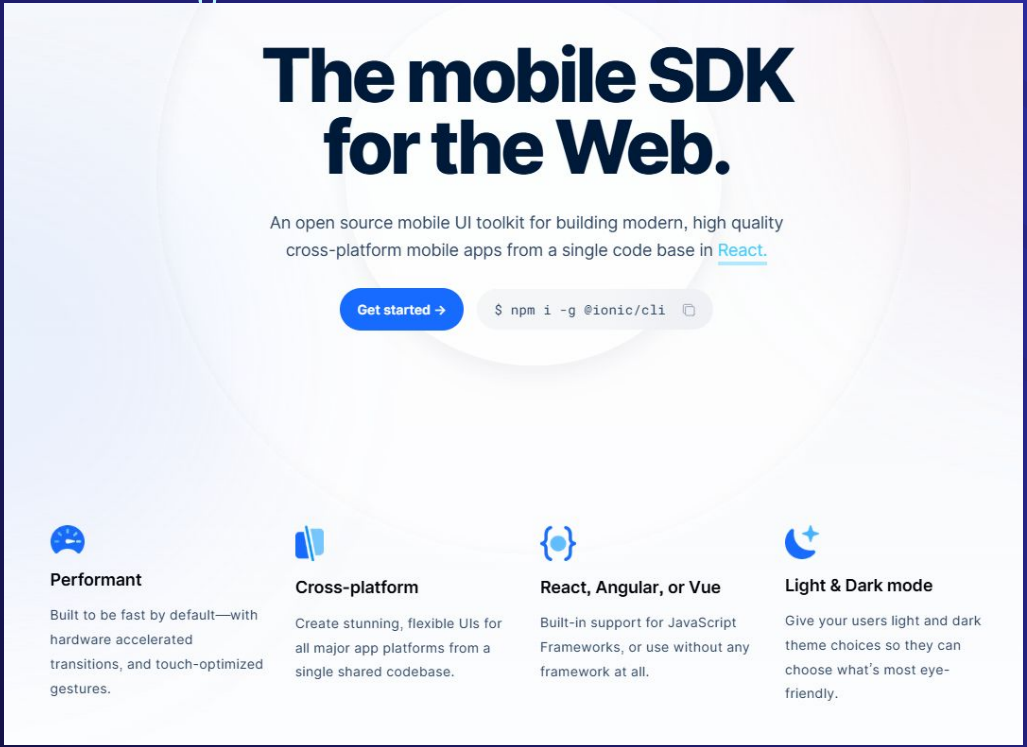
Genilson Medeiros Martins

MDB em Gerenciamento de Projetos - PUC Minas
Mestre em Tecnologias Estratégicas em Saúde - UEPB/NUTES
Engenheiro de Software / Líder Técnico - VIRTUS/UFCG
Graduado em Sistemas de Informação - UNIFACISA

ionic framework

ionic.

Desenvolvimento para mobile



The mobile SDK for the Web.

An open source mobile UI toolkit for building modern, high quality cross-platform mobile apps from a single code base in React.

```
$ npm i -g @ionic/cli
```

Performant

Built to be fast by default—with hardware accelerated transitions, and touch-optimized gestures.

Cross-platform

Create stunning, flexible UIs for all major app platforms from a single shared codebase.

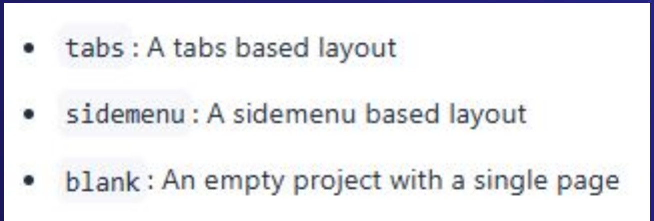
React, Angular, or Vue

Built-in support for JavaScript Frameworks, or use without any framework at all.

Light & Dark mode

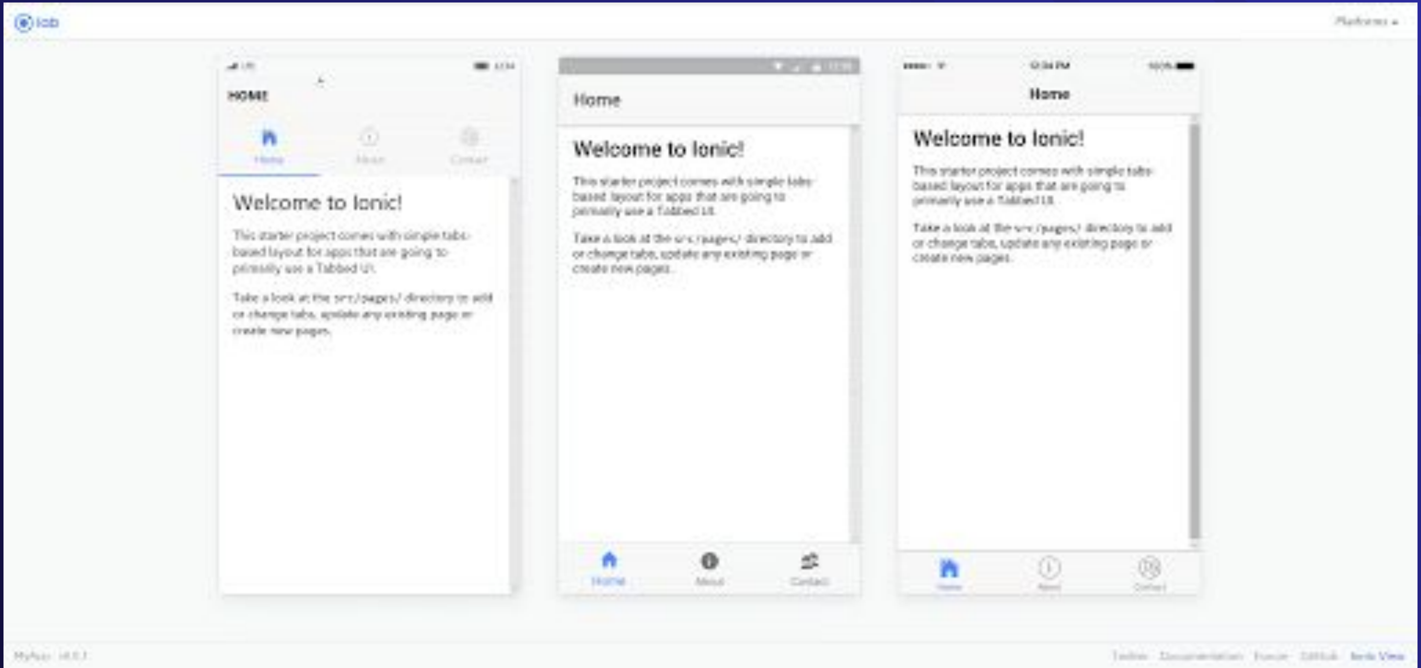
Give your users light and dark theme choices so they can choose what's most eye-friendly.

Templates do ionic



- `tabs` : A tabs based layout
- `sidemenu` : A sidemenu based layout
- `blank` : An empty project with a single page

Templates do ionic



Welcome to Ionic!

This starter project comes with simple tabs-based layout for apps that are going to primarily use a Tabbed UI.

Take a look at the src/pages/ directory to add or change tabs, update any existing page or create new pages.

Home

Welcome to Ionic!

This starter project comes with simple tabs-based layout for apps that are going to primarily use a Tabbed UI.

Take a look at the src/pages/ directory to add or change tabs, update any existing page or create new pages.

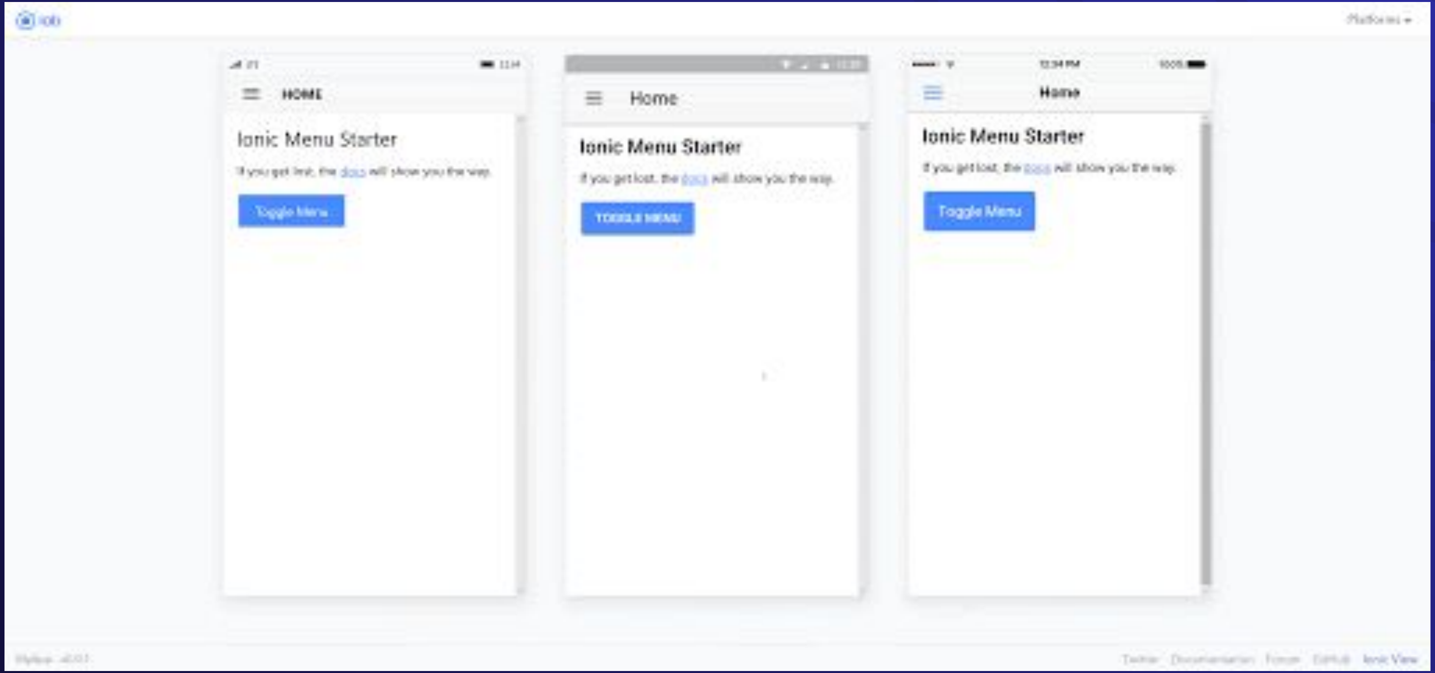
Home

Welcome to Ionic!

This starter project comes with simple tabs-based layout for apps that are going to primarily use a Tabbed UI.

Take a look at the src/pages/ directory to add or change tabs, update any existing page or create new pages.

Templates do ionic



HOME

Ionic Menu Starter

If you get lost, the docs will show you the way.

Toggle Menu

Home

Ionic Menu Starter

If you get lost, the docs will show you the way.

TOGGLE MENU

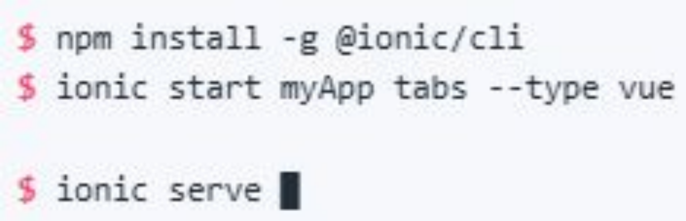
Home

Ionic Menu Starter

If you get lost, the docs will show you the way.

Toggle Menu

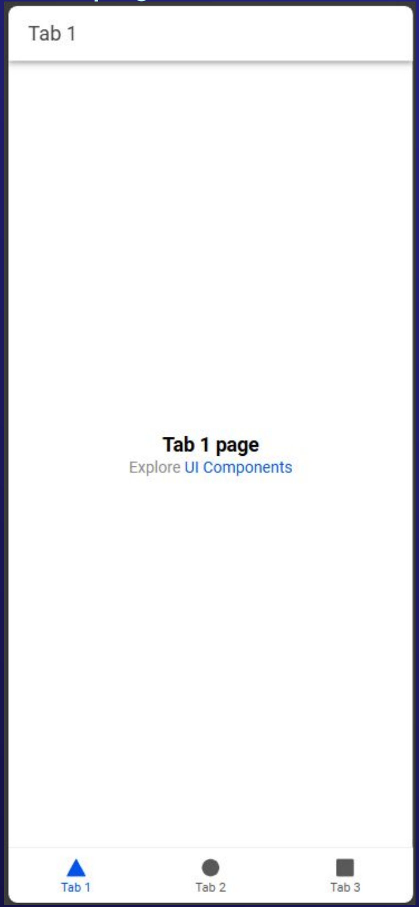
Primeiro projeto

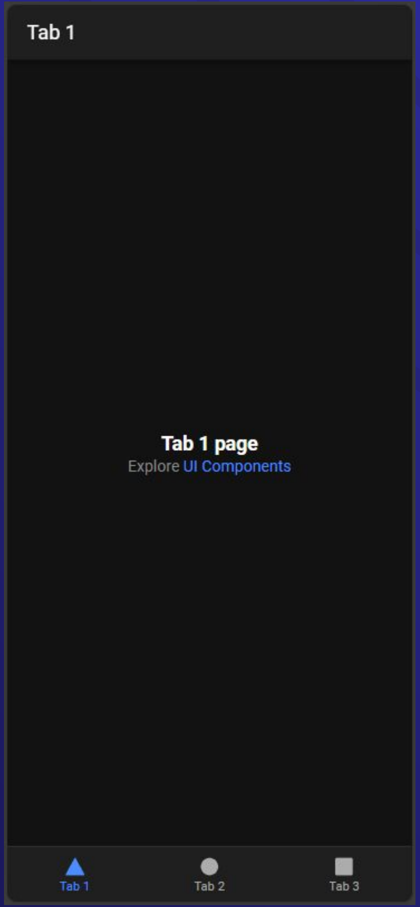


```
$ npm install -g @ionic/cli
$ ionic start myApp tabs --type vue

$ ionic serve
```

Themes

Light



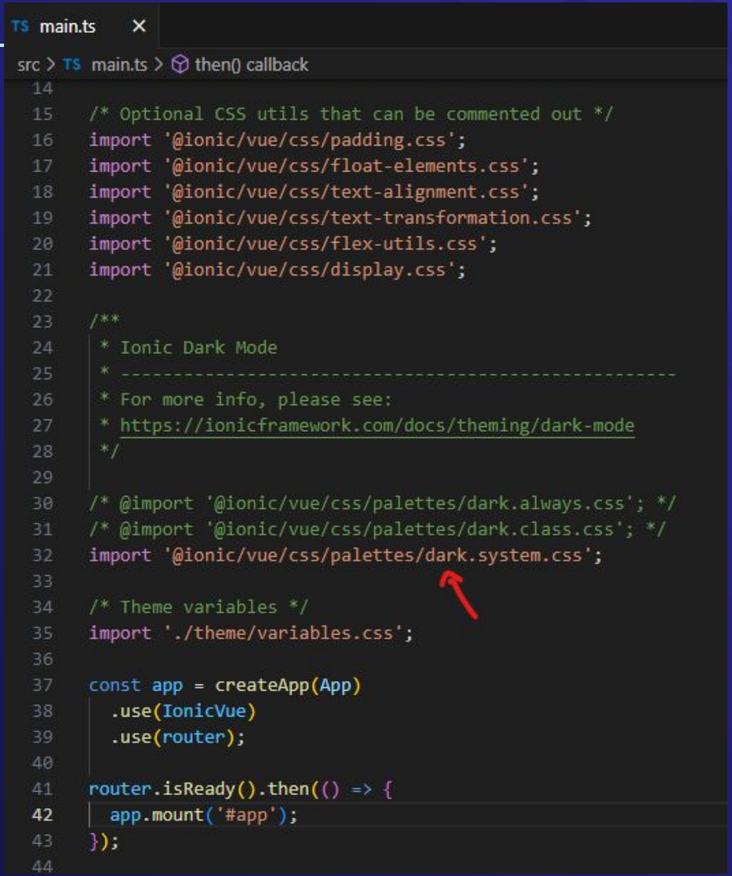


Dark

Themes

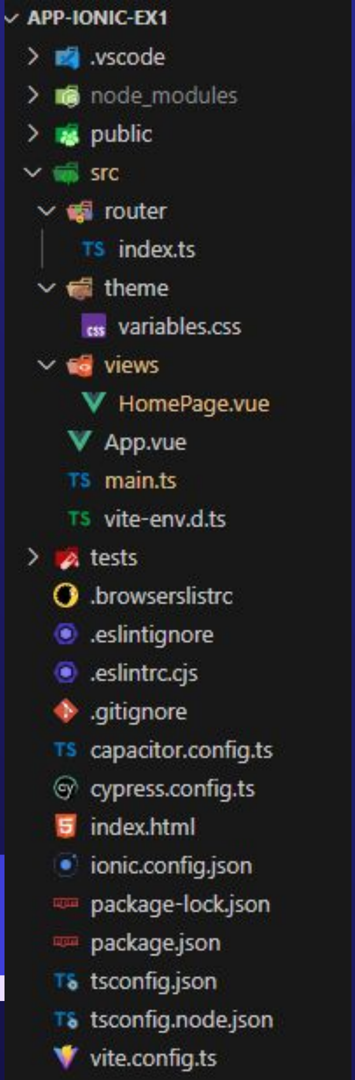
Dark: Mantém o css

Light: Remove o css



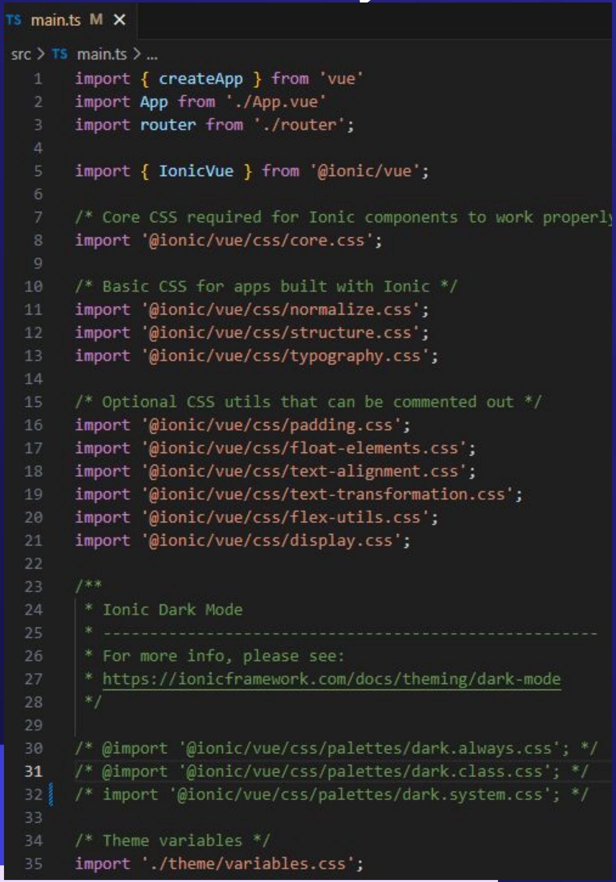
```
/* Optional CSS utils that can be commented out */
import '@ionic/vue/css/padding.css';
import '@ionic/vue/css/float-elements.css';
import '@ionic/vue/css/text-alignment.css';
import '@ionic/vue/css/text-transformation.css';
import '@ionic/vue/css/flex-utils.css';
import '@ionic/vue/css/display.css';

/**
 * Ionic Dark Mode
 * -----------------------------------------------------
 * For more info, please see:
 * https://ionicframework.com/docs/theming/dark-mode
 */

/* @import '@ionic/vue/css/palettes/dark.always.css'; */
/* @import '@ionic/vue/css/palettes/dark.class.css'; */
import '@ionic/vue/css/palettes/dark.system.css';

/* Theme variables */
import './theme/variables.css';

const app = createApp(App)
  .use(IonicVue)
  .use(router);

router.isReady().then(() => {
  app.mount('#app');
});

```

Estrutura

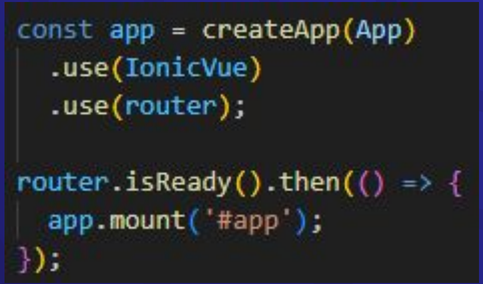


```
APP-IONIC-EX1
  > .vscode
  > node_modules
  > public
  v src
    v router
        index.ts
    v theme
        variables.css
    v views
        HomePage.vue
      App.vue
      main.ts
      vite-env.d.ts
  > tests
    .browserslistrc
    .eslintignore
    .eslintrc.cjs
    .gitignore
    capacitor.config.ts
    cypress.config.ts
    index.html
    ionic.config.json
    package-lock.json
    package.json
    tsconfig.json
    tsconfig.node.json
    vite.config.ts
```

Main.ts

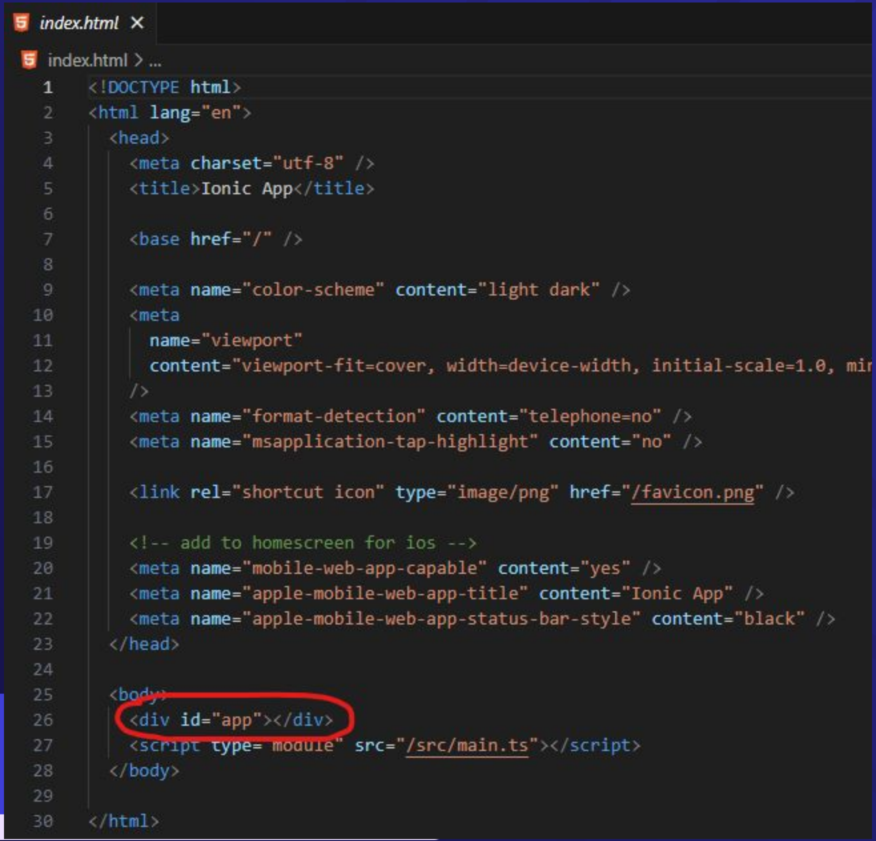


```
import { createApp } from 'vue'
import App from './App.vue'
import router from './router';

import { IonicVue } from '@ionic/vue';

/* Core CSS required for Ionic components to work properly
import '@ionic/vue/css/core.css';

/* Basic CSS for apps built with Ionic */
import '@ionic/vue/css/normalize.css';
import '@ionic/vue/css/structure.css';
import '@ionic/vue/css/typography.css';

/* Optional CSS utils that can be commented out */
import '@ionic/vue/css/padding.css';
import '@ionic/vue/css/float-elements.css';
import '@ionic/vue/css/text-alignment.css';
import '@ionic/vue/css/text-transformation.css';
import '@ionic/vue/css/flex-utils.css';
import '@ionic/vue/css/display.css';

/**
 * Ionic Dark Mode
 * -----------------------------------------------------
 * For more info, please see:
 * https://ionicframework.com/docs/theming/dark-mode
 */

/* @import '@ionic/vue/css/palettes/dark.always.css'; */
/* @import '@ionic/vue/css/palettes/dark.class.css'; */
/* import '@ionic/vue/css/palettes/dark.system.css'; */

/* Theme variables */
import './theme/variables.css';
```



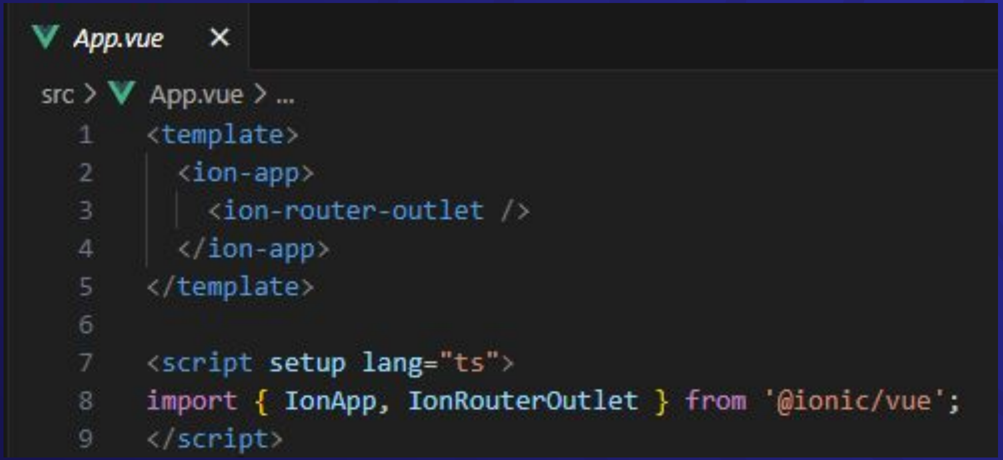
```
const app = createApp(App)
  .use(IonicVue)
  .use(router);

router.isReady().then(() => {
  app.mount('#app');
});
```

Main.ts



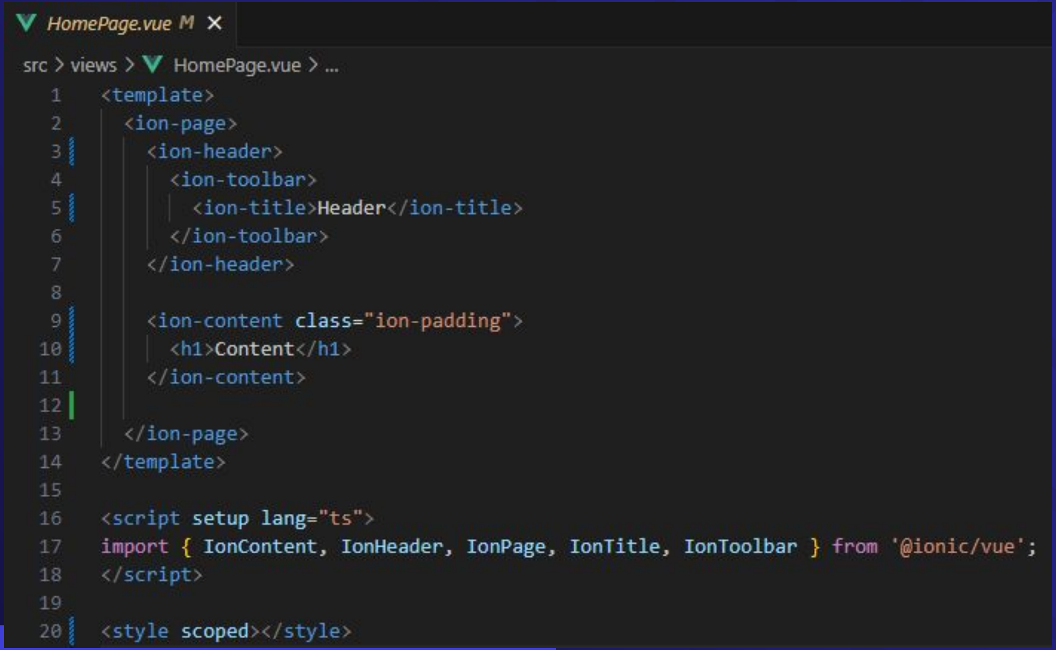
```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="utf-8" />
    <title>Ionic App</title>

    <base href="/" />

    <meta name="color-scheme" content="light dark" />
    <meta
      name="viewport"
      content="viewport-fit=cover, width=device-width, initial-scale=1.0, mi
    />
    <meta name="format-detection" content="telephone=no" />
    <meta name="msapplication-tap-highlight" content="no" />

    <link rel="shortcut icon" type="image/png" href="/favicon.png" />

    <!-- add to homescreen for ios -->
    <meta name="mobile-web-app-capable" content="yes" />
    <meta name="apple-mobile-web-app-title" content="Ionic App" />
    <meta name="apple-mobile-web-app-status-bar-style" content="black" />
  </head>

  <body>
    <div id="app"></div>
    <script type="module" src="/src/main.ts"></script>
  </body>

</html>
```

App.vue



```
<template>
  <ion-app>
    <ion-router-outlet />
  </ion-app>
</template>

<script setup lang="ts">
import { IonApp, IonRouterOutlet } from '@ionic/vue';
</script>
```

Home.vue

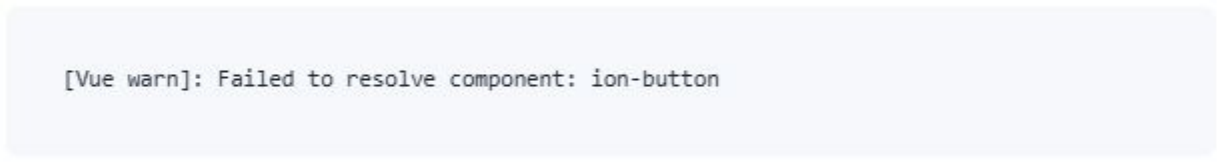


```
<template>
  <ion-page>
    <ion-header>
      <ion-toolbar>
        <ion-title>Header</ion-title>
      </ion-toolbar>
    </ion-header>

    <ion-content class="ion-padding">
      <h1>Content</h1>
    </ion-content>

  </ion-page>
</template>

<script setup lang="ts">
import { IonContent, IonHeader, IonPage, IonTitle, IonToolbar } from '@ionic/vue';
</script>

<style scoped></style>
```

DEMO.

Vamos testar nosso ambiente

Troubleshooting



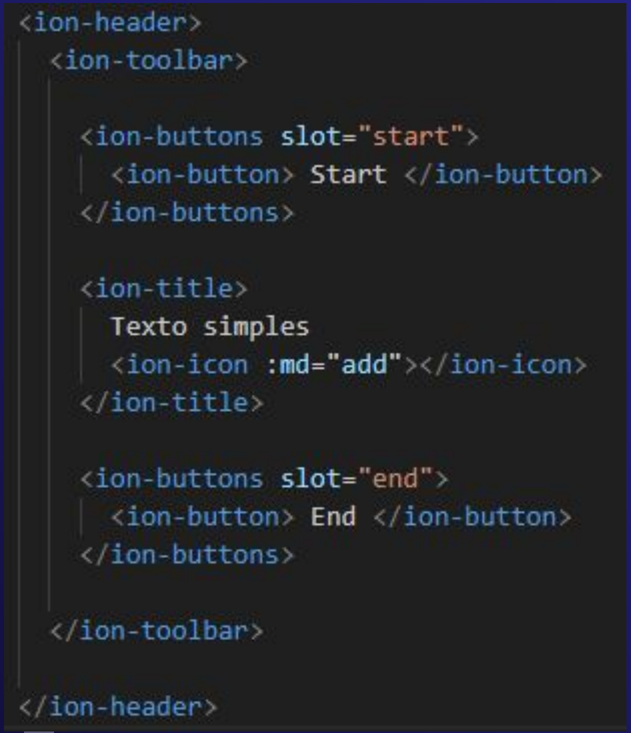
```
[Vue warn]: Failed to resolve component: ion-button
```

Se você vir esse aviso, é provável que você não tenha importado seu componente de @ionic/vue.

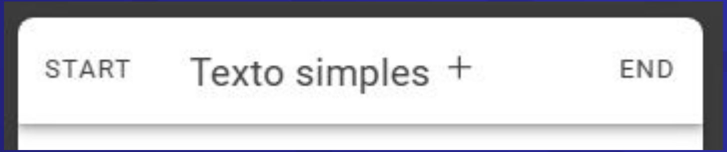
Por padrão, todos os componentes do Ionic Vue são registrados localmente, o que significa que você precisa importá-los sempre que quiser usá-los.

Sem importar o componente, você obterá apenas o Web Component subjacente, e recursos específicos do Vue, como o v-model, não funcionarão.

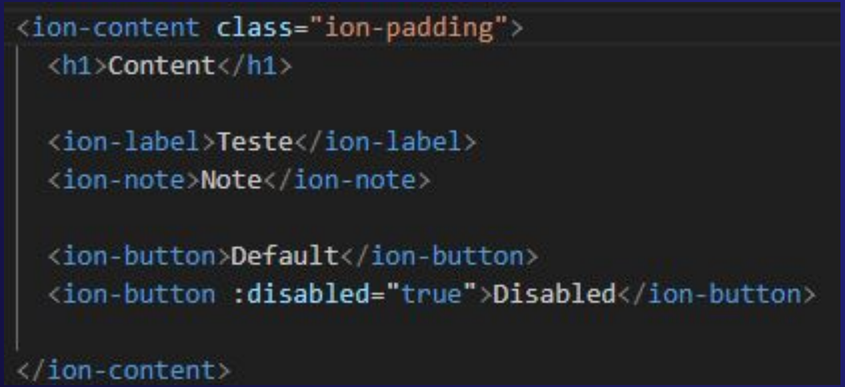
ion-header



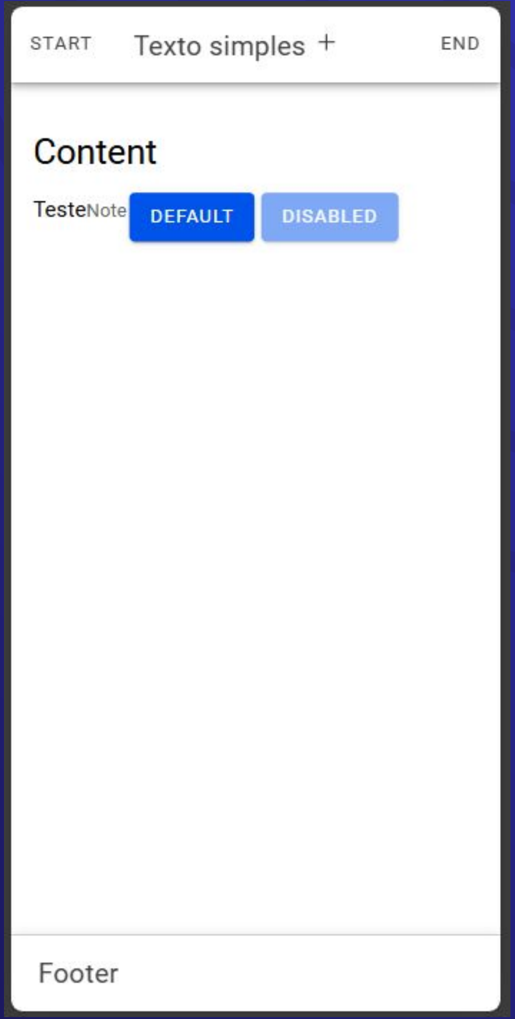
```
<ion-header>
  <ion-toolbar>

    <ion-buttons slot="start">
      <ion-button> Start </ion-button>
    </ion-buttons>

    <ion-title>
      Texto simples
      <ion-icon :md="add"></ion-icon>
    </ion-title>

    <ion-buttons slot="end">
      <ion-button> End </ion-button>
    </ion-buttons>

  </ion-toolbar>

</ion-header>
```



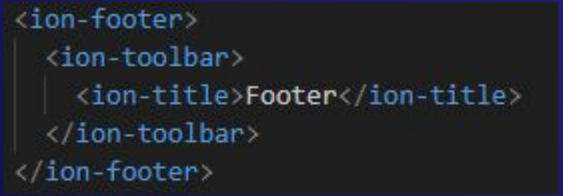
ion-content



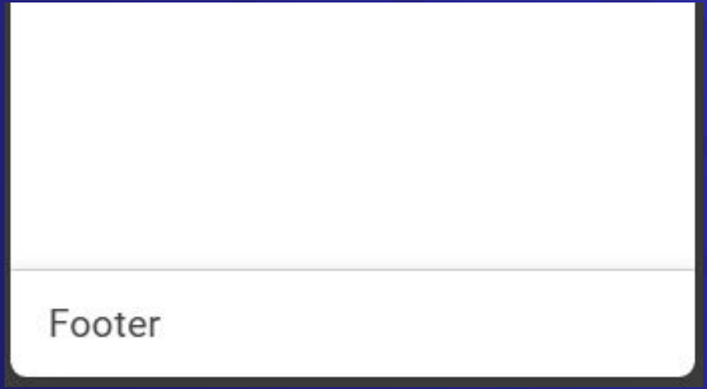
```
<ion-content class="ion-padding">
  <h1>Content</h1>

  <ion-label>Teste</ion-label>
  <ion-note>Note</ion-note>

  <ion-button>Default</ion-button>
  <ion-button :disabled="true">Disabled</ion-button>

</ion-content>
```



ion-footer



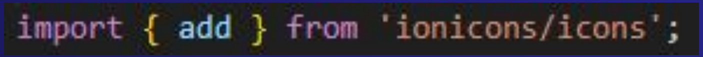
```
<ion-footer>
  <ion-toolbar>
    <ion-title>Footer</ion-title>
  </ion-toolbar>
</ion-footer>
```



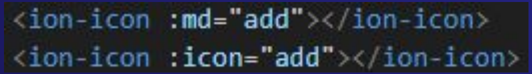
Footer

ion-icos

Ionicons: Premium Open Source Icon Pack for Ionic Framework



```
import { add } from 'ionicons/icons';
```

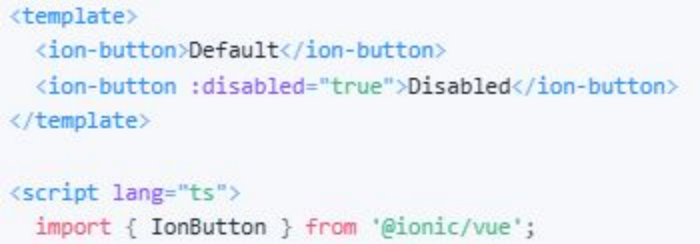


```
<ion-icon :md="add"></ion-icon>
<ion-icon :icon="add"></ion-icon>
```

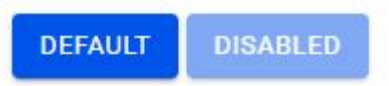
Principais componentes

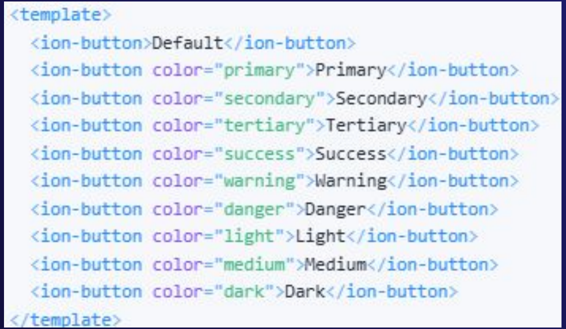
- API Index | Ionic Docs API Index for all API Custom Elements

Principais componentes

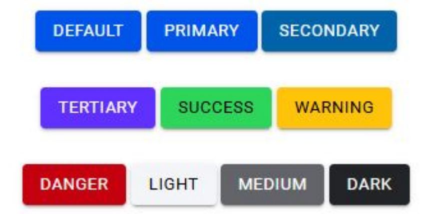


```
<template>
  <ion-button>Default</ion-button>
  <ion-button :disabled="true">Disabled</ion-button>
</template>

<script lang="ts">
  import { IonButton } from '@ionic/vue';
```

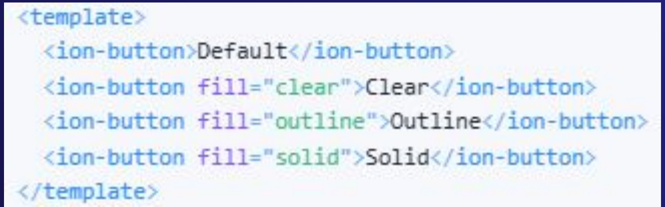




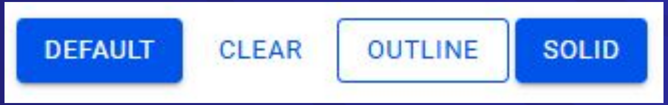
```
<template>
  <ion-button>Default</ion-button>
  <ion-button color="primary">Primary</ion-button>
  <ion-button color="secondary">Secondary</ion-button>
  <ion-button color="tertiary">Tertiary</ion-button>
  <ion-button color="success">Success</ion-button>
  <ion-button color="warning">Warning</ion-button>
  <ion-button color="danger">Danger</ion-button>
  <ion-button color="light">Light</ion-button>
  <ion-button color="medium">Medium</ion-button>
  <ion-button color="dark">Dark</ion-button>
</template>
```



Principais componentes

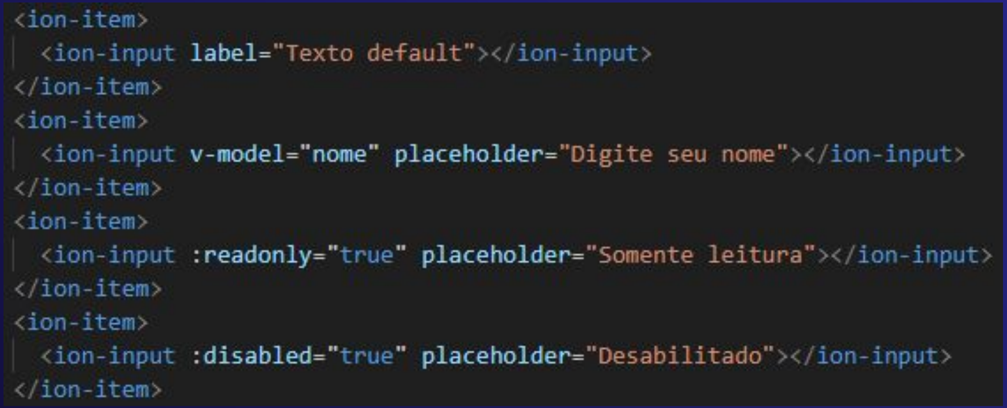


```
<template>
  <ion-button>Default</ion-button>
  <ion-button fill="clear">Clear</ion-button>
  <ion-button fill="outline">Outline</ion-button>
  <ion-button fill="solid">Solid</ion-button>
</template>
```

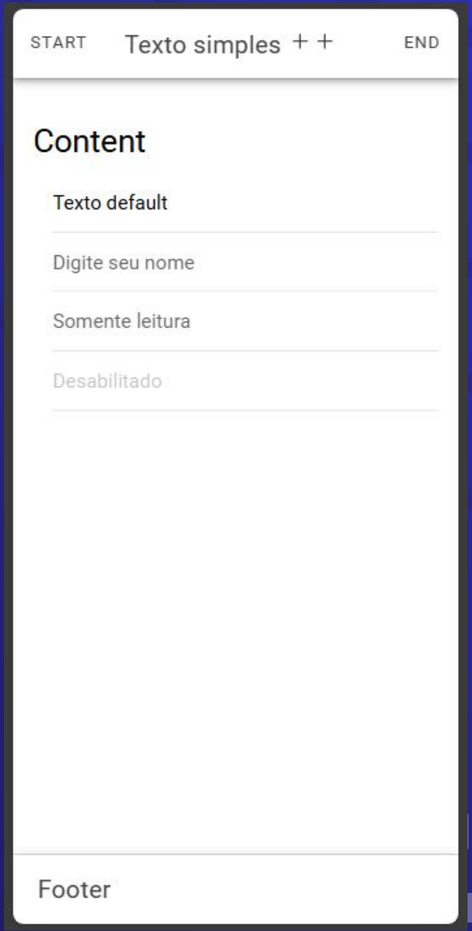


ion-button: Style Buttons with Custom CSS Properties

Principais componentes



```
<ion-item>
  <ion-input label="Texto default"></ion-input>
</ion-item>
<ion-item>
  <ion-input v-model="nome" placeholder="Digite seu nome"></ion-input>
</ion-item>
<ion-item>
  <ion-input :readonly="true" placeholder="Somente leitura"></ion-input>
</ion-item>
<ion-item>
  <ion-input :disabled="true" placeholder="Desabilitado"></ion-input>
</ion-item>
```



Texto simples + +

Content

Texto default

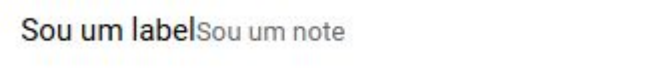
Digite seu nome

Somente leitura

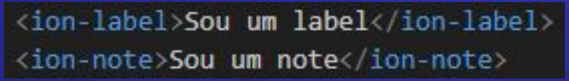
Desabilitado

Footer

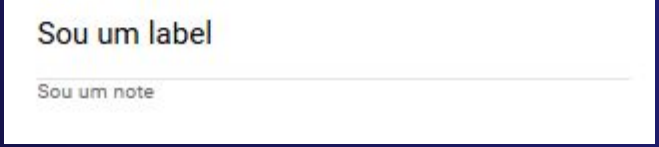
Principais componentes



Sou um labelSou um note

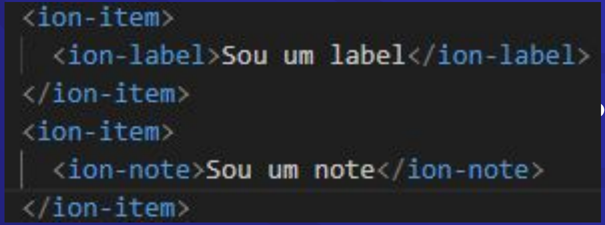


```
<ion-label>Sou um label</ion-label>
<ion-note>Sou um note</ion-note>
```

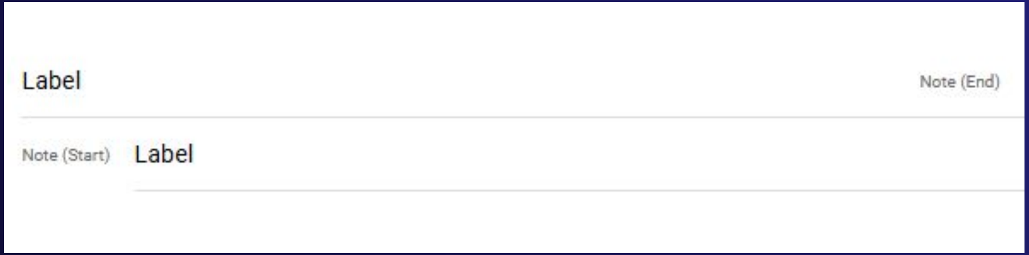


Sou um label

Sou um note

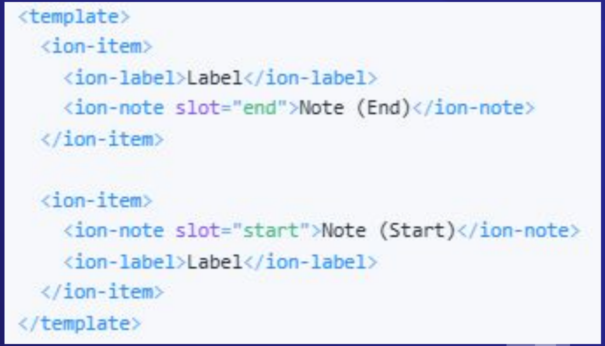


```
<ion-item>
  <ion-label>Sou um label</ion-label>
</ion-item>
<ion-item>
  <ion-note>Sou um note</ion-note>
</ion-item>
```



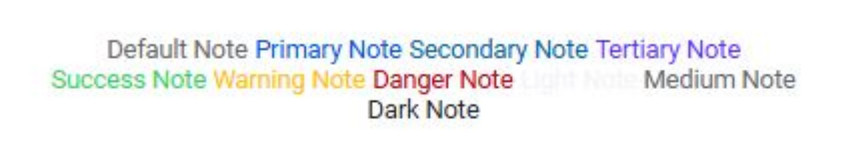
Label Note (End)

Note (Start) Label

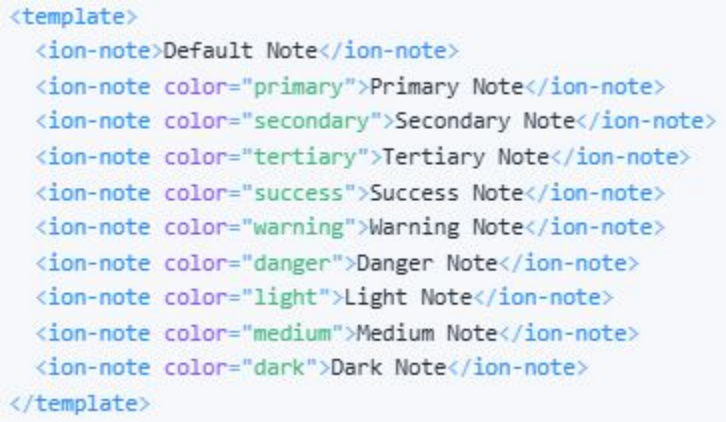


```
<template>
  <ion-item>
    <ion-label>Label</ion-label>
    <ion-note slot="end">Note (End)</ion-note>
  </ion-item>

  <ion-item>
    <ion-note slot="start">Note (Start)</ion-note>
    <ion-label>Label</ion-label>
  </ion-item>
</template>
```

Principais componentes



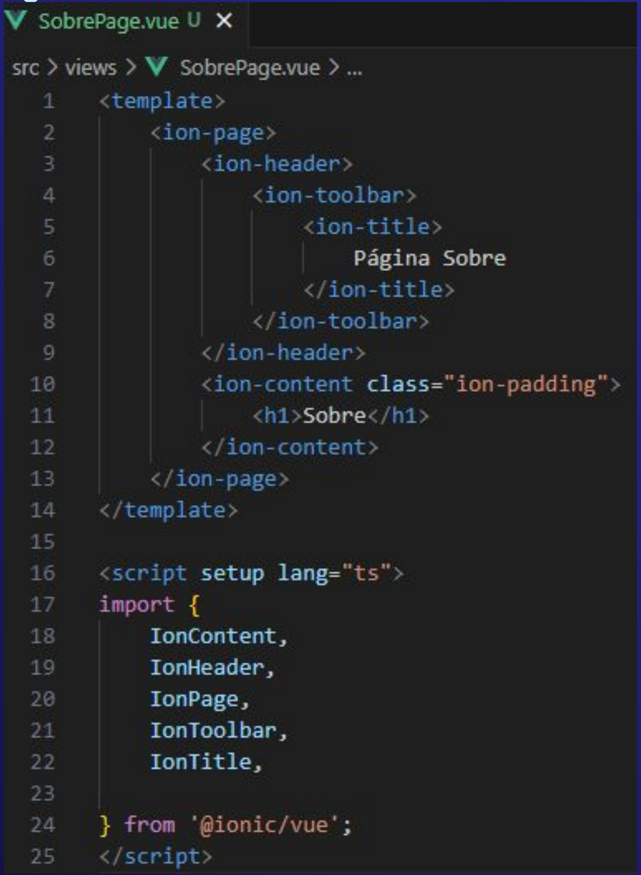
Default Note Primary Note Secondary Note Tertiary Note
Success Note Warning Note Danger Note Light Note Medium Note
Dark Note



```
<template>
  <ion-note>Default Note</ion-note>
  <ion-note color="primary">Primary Note</ion-note>
  <ion-note color="secondary">Secondary Note</ion-note>
  <ion-note color="tertiary">Tertiary Note</ion-note>
  <ion-note color="success">Success Note</ion-note>
  <ion-note color="warning">Warning Note</ion-note>
  <ion-note color="danger">Danger Note</ion-note>
  <ion-note color="light">Light Note</ion-note>
  <ion-note color="medium">Medium Note</ion-note>
  <ion-note color="dark">Dark Note</ion-note>
</template>
```

Navegação

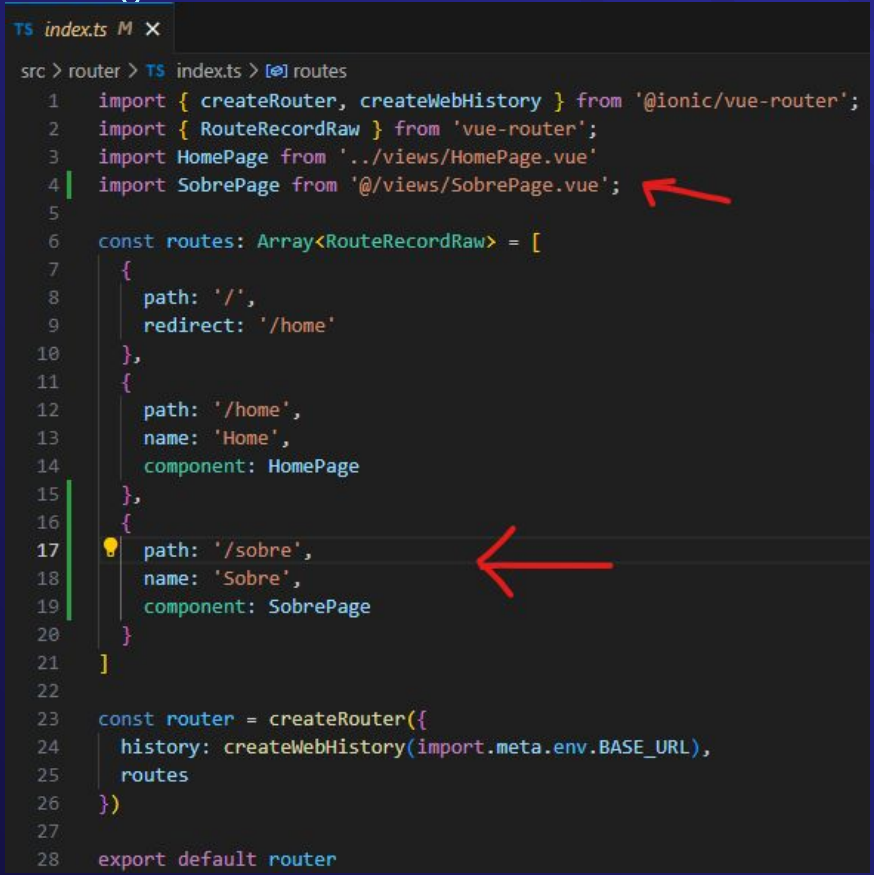
Criar a nova página



```
<template>
    <ion-page>
        <ion-header>
            <ion-toolbar>
                <ion-title>
                    Página Sobre
                </ion-title>
            </ion-toolbar>
        </ion-header>
        <ion-content class="ion-padding">
            <h1>Sobre</h1>
        </ion-content>
    </ion-page>
</template>

<script setup lang="ts">
import {
    IonContent,
    IonHeader,
    IonPage,
    IonToolbar,
    IonTitle,

} from '@ionic/vue';
</script>
```

Navegação

Adicionar a nova rota



```
import { createRouter, createWebHistory } from '@ionic/vue-router';
import { RouteRecordRaw } from 'vue-router';
import HomePage from '../views/HomePage.vue'
import SobrePage from '@/views/SobrePage.vue';

const routes: Array<RouteRecordRaw> = [
  {
    path: '/',
    redirect: '/home'
  },
  {
    path: '/home',
    name: 'Home',
    component: HomePage
  },
  {
    path: '/sobre',
    name: 'Sobre',
    component: SobrePage
  }
]

const router = createRouter({
  history: createWebHistory(import.meta.env.BASE_URL),
  routes
})

export default router
```

Navegação

Adicionar um botão para navegação na página Home

```
<ion-button router-link="/sobre" router-direction="back">Click Me</ion-button>
```

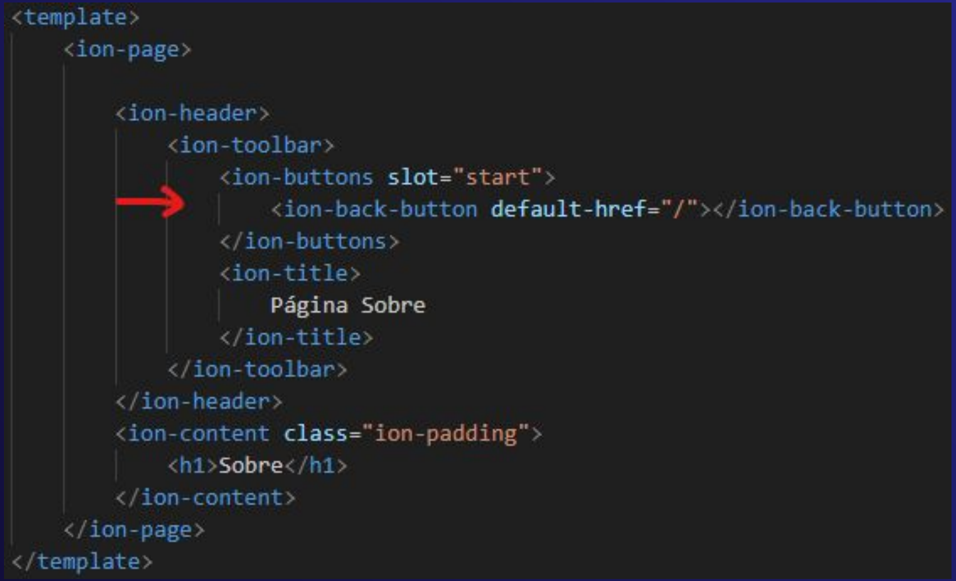
Página Sobre

Sobre

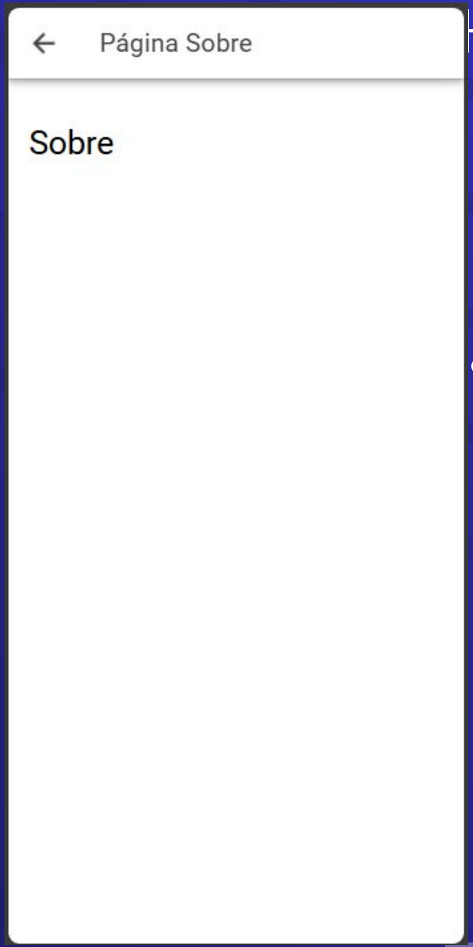
Ao clicar no botão o usuário será direcionado para a página sobre

Navegação

Para voltar para a página home, adicione o código a seguir:



```
<template>
    <ion-page>

        <ion-header>
            <ion-toolbar>
                <ion-buttons slot="start">
                    <ion-back-button default-href="/"></ion-back-button>
                </ion-buttons>
                <ion-title>
                    Página Sobre
                </ion-title>
            </ion-toolbar>
        </ion-header>
        <ion-content class="ion-padding">
            <h1>Sobre</h1>
        </ion-content>
    </ion-page>
</template>
```

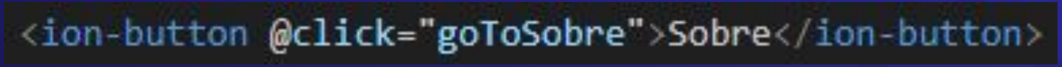


Página Sobre

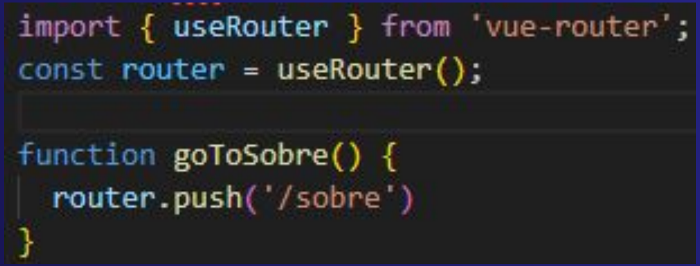
Sobre

Navegação

Também é possível navegar para outra página a partir de eventos



```html
<ion-button @click="goToSobre">Sobre</ion-button>
```



```javascript
import { useRouter } from 'vue-router';
const router = useRouter();

function goToSobre() {
  router.push('/sobre')
}
```