



MCAST

INTRODUCTION TO MOBILE APPLICATIONS DEVELOPMENT

Navigation

ITSFT-406-2001

Ing. James Attard

OVERVIEW OF NAVIGATION IN IONIC

- Efficient navigation is crucial for a seamless user experience.
- Ionic provides multiple navigation options like back buttons, side menus, and tabs.
- In this topic we'll explore each method to enhance your app's navigation.

Full Documentation:

<http://ionicframework.com/docs/api/nav>

ION-NAV

- We can use Ion-Nav to navigate between screens. Nav is a standalone component for loading arbitrary components and pushing new components on to the stack.
- We can place the `<IonNav>` component on our root page (ex. *Home.tsx*).
- Each page will then have an `<IonNavLink>` component to navigate us to the next screens.

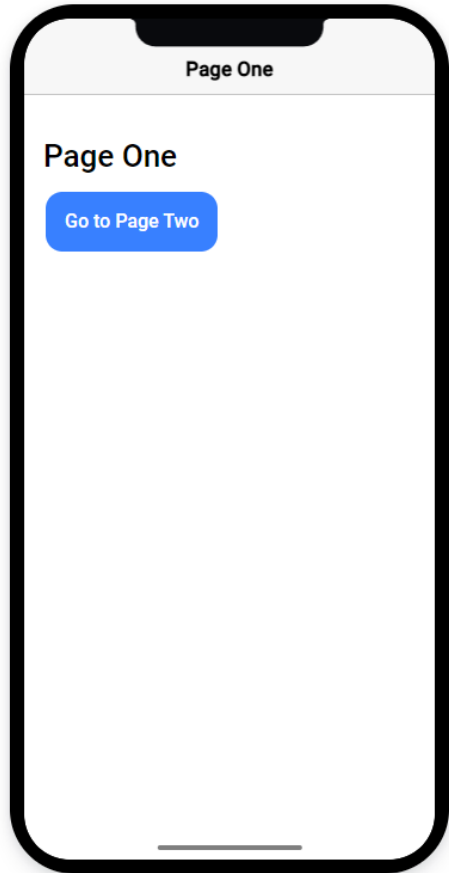
Read More:

<https://ionicframework.com/docs/api/nav>

ION-NAV



MCAST



```
TS src/main.tsx  TS src/page-one.tsx  TS src/page-two.tsx  TS

import React from 'react';
import { IonNav } from '@ionic/react';

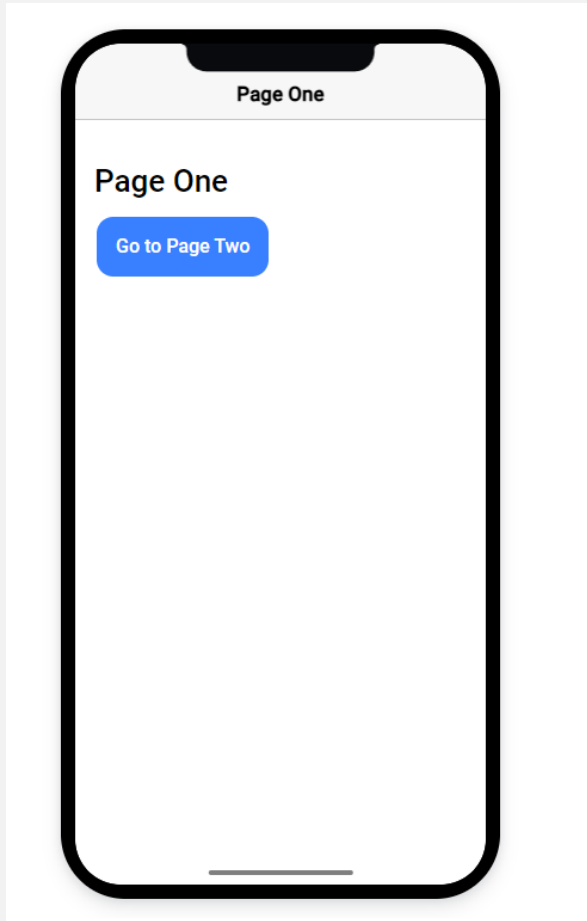
import PageOne from './page-one';

function Example() {
  return <IonNav root={() => <PageOne />></IonNav>;
}
export default Example;
```

ION-NAV



MCAST



TS src/main.tsx

TS src/page-one.tsx

TS src/page-two.tsx

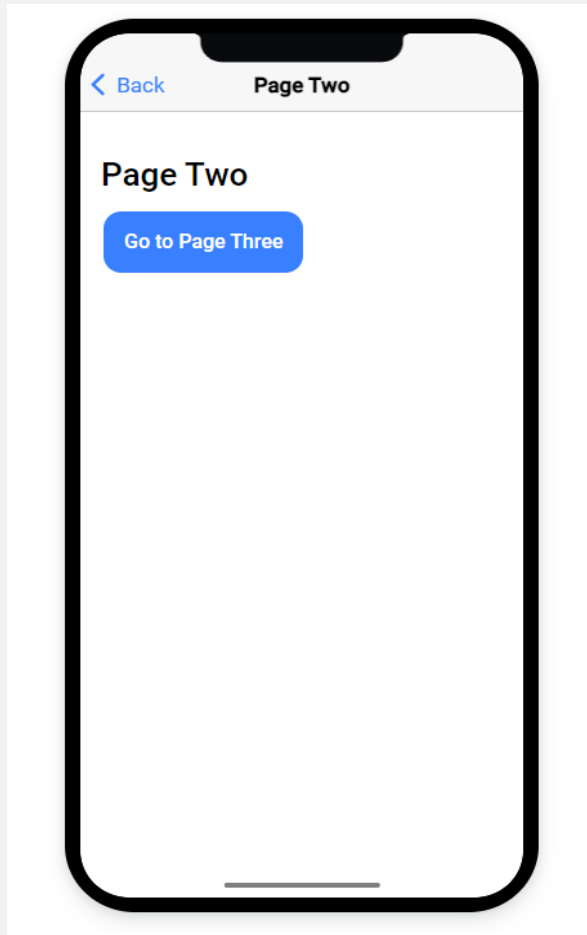
TS src/page-three.tsx

```
import React from 'react';
import { IonButton, IonHeader, IonContent, IonNavLink, IonToolbar, IonTitle } from '@ionic/react';
import PageTwo from './page-two';

function PageOne() {
  return (
    <>
      <IonHeader>
        <IonToolbar>
          <IonTitle>Page One</IonTitle>
        </IonToolbar>
      </IonHeader>
      <IonContent class="ion-padding">
        <h1>Page One</h1>
        <IonNavLink routerDirection="forward" component={() => <PageTwo />>
          <IonButton>Go to Page Two</IonButton>
        </IonNavLink>
      </IonContent>
    </>
  );
}

export default PageOne;
```

ION-NAV



```
TS src/main.tsx TS src/page-one.tsx TS src/page-two.tsx TS src/page-three.tsx

import React from 'react';
import {
  IonBackButton,
  IonButtons,
  IonButton,
  IonHeader,
  IonContent,
  IonNavLink,
  IonToolbar,
  IonTitle,
} from '@ionic/react';

import PageThree from './page-three';

function PageTwo() {
  return (
    <>
      <IonHeader>
        <IonToolbar>
          <IonButtons slot="start">
            <IonBackButton></IonBackButton>
          </IonButtons>
          <IonTitle>Page Two</IonTitle>
        </IonToolbar>
      </IonHeader>
      <IonContent class="ion-padding">
        <h1>Page Two</h1>
        <IonNavLink routerDirection="forward" component={() => <PageThree />>
          <IonButton>Go to Page Three</IonButton>
        </IonNavLink>
      </IonContent>
    </>
  );
}

export default PageTwo;
```



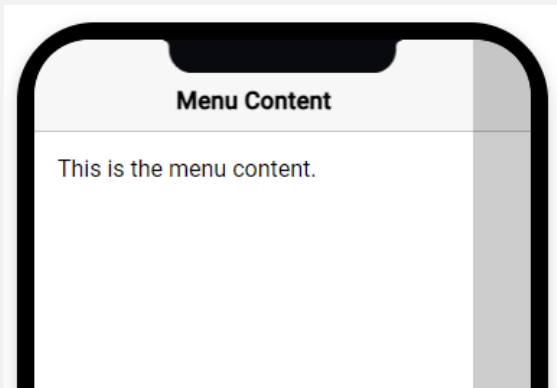
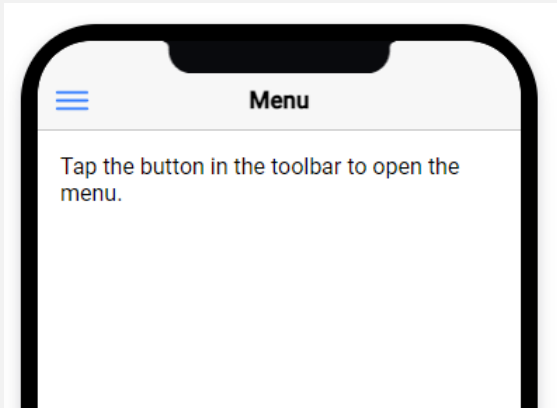
ION-MENU

- The menu component is a navigation drawer that slides in from the side of the current view.
- By default, it uses the start side, making it slide in from the left for LTR and right for RTL, but the side can be overridden.

Read More:

<https://ionicframework.com/docs/api/menu>

ION-MENU



```
import React from 'react';
import { IonButtons, IonContent, IonHeader, IonMenu, IonMenuButton, IonPage, IonTitle, Io
function Example() {
  return (
    <>
      <IonMenu contentId="main-content">
        <IonHeader>
          <IonToolbar>
            <IonTitle>Menu Content</IonTitle>
          </IonToolbar>
        </IonHeader>
        <IonContent className="ion-padding">This is the menu content.</IonContent>
      </IonMenu>
      <IonPage id="main-content">
        <IonHeader>
          <IonToolbar>
            <IonButtons slot="start">
              <IonMenuButton></IonMenuButton>
            </IonButtons>
            <IonTitle>Menu</IonTitle>
          </IonToolbar>
        </IonHeader>
        <IonContent className="ion-padding">Tap the button in the toolbar to open the men
      </IonPage>
    </>
  );
}
export default Example;
```

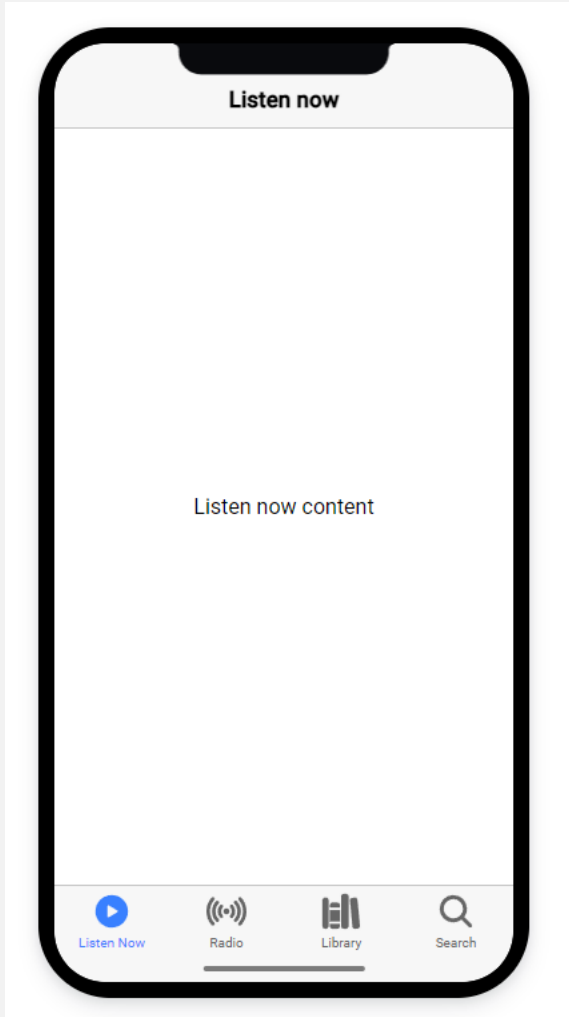

ION-TABS

- Tabs can be used with the Ionic router to implement tab-based navigation. The component is a container of individual Tab components.
- The tab bar and active tab will automatically resolve based on the url. This is the most common pattern for tabs navigation.

Read More:

<https://ionicframework.com/docs/api/tabs>

ION-TABS



```
import { IonTabs, IonTabBar, IonTabButton, IonIcon, IonLabel, IonRouterOutlet } from '@ionic/react';
import { IonReactRouter } from '@ionic/react-router';

import { Route, Redirect } from 'react-router';

import { playCircle, radio, library, search } from 'ionicons/icons';

import HomePage from './pages/HomePage';
import RadioPage from './pages/RadioPage';
import LibraryPage from './pages/LibraryPage';
import SearchPage from './pages/SearchPage';

function Example() {
  return (
    <IonReactRouter>
      <IonTabs>
        <IonRouterOutlet>
          <Redirect exact path="/" to="/home" />
          {/*
            Use the render method to reduce the number of renders your component will
            Use the component prop when your component depends on the RouterComponent
            */}
          <Route path="/home" render={() => <HomePage />} exact={true} />
          <Route path="/radio" render={() => <RadioPage />} exact={true} />
          <Route path="/library" render={() => <LibraryPage />} exact={true} />
          <Route path="/search" render={() => <SearchPage />} exact={true} />
        </IonRouterOutlet>

        <IonTabBar slot="bottom">
          <IonTabButton tab="home" href="/home">
            <IonIcon icon={playCircle} />
            <IonLabel>Listen now</IonLabel>
          </IonTabButton>

          <IonTabButton tab="radio" href="/radio">
            <IonIcon icon={radio} />
            <IonLabel>Radio</IonLabel>
          </IonTabButton>
        </IonTabBar>
      </IonTabs>
    </IonReactRouter>
  );
}
```

BEST PRACTICES

- **Keep it Simple:** Prioritize a clean and intuitive navigation flow.
- **Consistent Design:** Maintain a consistent look and feel.
- **User Feedback:** Provide feedback for successful actions.