

TELUS Component Library

Tabs

Multi-Platform Component | [Figma UI KIT](#)

Tab one	Tab two	Tab three	Tab four	Tab five
---------	---------	-----------	----------	----------

```
<Tabs
  items={ [
    { label: 'Tab one', href: '#one', id: 'one' },
    { label: 'Tab two', href: '#two', id: 'two' },
    { label: 'Tab three', href: '#three', id: 'three' },
    { label: 'Tab four', href: '#four', id: 'four' },
    { label: 'Tab five', href: '#five', id: 'five' }
  ] }
/>
```

Introduction

Use Tabs to display a horizontal menu of tabs that change the content shown below when pressed.

If there are too many tabs to fit the horizontal space, left and right buttons are shown.

[Follow the appropriate instructions](#) to add this component in to your app.

Guidance

- Use Tabs to show/hide a mutually exclusive content areas.
- If the tabs should link to a wholly different page with its own URL, use `NavigationBar` instead
- Recommend to use labels that are 35 characters or fewer
- Allow the use of hashes in the URL to automatically load a tab. Eg. <http://t.com#premium> should load the “Premium” tab
- Change the page URL to include the hash as tabs change

Accessibility

- If a user is using tabbed navigation and horizontal scrolling is required, tabbing through tabs scrolls them into view automatically
- Scrolling methods other than the scroll buttons may also be used, such as touchscreen swipes and touchpad or mousewheel scrolling
- Consider including an assertive focus effect inside content to move screenreader focus to the appropriate heading or content start when new content is rendered

Controlled usage

Pass `value` and `onChange` to control the tabs from a parent. This is usually the easiest way to render different content for different tabs:

- `value` should be the `id` property of the current tab in the `items` array
- `onChange` should be a function that updates the state that selects content and is passed to `value`
- To integrate with URL hashes on web so selections remain after returning to a page, choose the default state value based on the URL hash

Platform considerations

The component is available on both native platforms and web.

Touch swipes with momentum may be used to scroll through many tabs, in addition to the scroll buttons.

Keep in mind that URL hashes won't have any effect in an app.

Props

Name	Type	Platform	Default	Description
items	arrayOf	standard	[]	Array of `TabItem`

Name	Type	Platform	Default	Description
value	string	standard		`id` property of the current tab in the array
initialValue	string	standard		
onChange	func	standard		Callback for when selected tab changes
tokens	tokens	standard		System tokens provided in tokens for more details
itemTokens	custom	standard		Custom tokens for `TabItem`
scrollButtonTokens	custom	standard		Custom tokens for `HorizontalScrollButton`
variant	variant	standard		System variant provided in variants for more details

Tokens

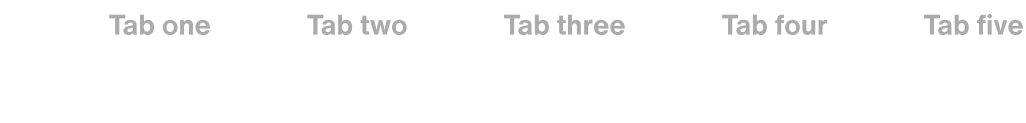
In exceptional circumstances, the following tokens can be passed to this component to override its default styles. **Do not do this unless absolutely necessary.** [Read more about overriding styles.](#)

► View Tokens

Variants

Inverse

- Use on dark backgrounds.



```
<div style={{ backgroundColor: 'black', padding: 20 }}>
  <Tabs
    variant={{ inverse: true }}
    items={[
      { label: 'Tab one', href: '#one', id: 'one' },
      { label: 'Tab two', href: '#two', id: 'two' },
      { label: 'Tab three', href: '#three', id: 'three' },
      { label: 'Tab four', href: '#four', id: 'four' },
      { label: 'Tab five', href: '#five', id: 'five' }
    ]}
  />
</div>
```

Feedback

- Spotted a problem with this component? Raise an [issue on GitHub](#)
- See any [existing issues](#) for this component
- Contact the team on slack in [#ds-support](#)