

TELUS Component Library

DatePicker

Web Component | [Figma UI KIT](#)

DatePicker Enter a Date ?

DD / MM / YYYY

```
<DatePicker id="DatePicker" label="DatePicker" hint="Enter a Date" tooltip="This is a tooltip" />
```

Introduction

Use DatePicker to select a date on a calendar.

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

Guidance

- Use overlay (default) variant to display in a modal container
- Use overlay variant whenever possible as it is the most accessible solution; the input form field is type-accessible and optimized for mobile customers
- Includes an input form field to allow manual key-in of dates
- Opens the modal when the input form field receives focus
- Width expands 100% until max-width of 340px
- Enable selection of old dates by passing a `function` that returns a `boolean` in `isDayDisabled`

Uncontrolled component



DANGER

While `DatePicker` can be used as an uncontrolled component, using a ref to get the input's value behaves inconsistently between the default and

inline version of the component. The inline version forwards the ref to a hidden input, but due to the limitations of the `react-dates` library, the default `DatePicker` attaches the ref to the component's wrapper.

It's recommended to use the `onDateChange` prop to get the current value of the input even when using it as an uncontrolled component. If you must rely on a ref to get the value on the input, you will need to find the input element nested inside the wrapper.

DatePicker

DD / MM / YYYY

Current Value: None

```
function UncontrolledExample() {
  const [date, setDate] = useState()

  return (
    <>
      <DatePicker id="DatePicker" label="DatePicker"
onDateChange={setDate} />
      Current Value: {date ? date.format('YYYY-MM-DD') : 'None'}
    </>
  )
}
```

Controlled component

If it is required that the state of the `DatePicker` be controlled by the application or other external methods, `date` and `onDateChange` props must be passed to the `DatePicker`.

DatePicker

DD / MM / YYYY

```
function ControlledExample() {
  const [date, setDate] = useState()
```

```
return <DatePicker id="DatePicker" label="DatePicker" date={date} onChange={setDate} />
}
```

Accessibility

DatePicker has accessible controls and also uses the `label` prop to label the corresponding input.

Platform considerations

This component is currently only supported on web.

Date enablement

Date picker has the ability to enable or disable dates in the past by the use of a `function` that returns a `boolean` in `isDayDisabled`

DatePicker

DD / MM / YYYY

Current Value: None

```
function UncontrolledExample() {
  const [date, setDate] = useState()

  return (
    <>
      <DatePicker
        id="DatePicker"
        label="DatePicker"
        onChange={setDate}
        isDayDisabled={() => false}
      />
      Current Value: {date ? date.format('YYYY-MM-DD') : 'None'}
    </>
  )
}
```

Props

Name	Type	Platform	Default	Description
copy	'en' 'fr'	web only	'en'	Whether the English or French copy will be used (e.g. for accessibility labels).
hintPosition	'inline' 'below'	web only	'inline'	Position of the hint relative to label. Use `below` to display a larger hint below the label.
variant	variant	web only	{}	System variant prop, see variants for more details
date	custom	web only	undefined	A Moment instance representing the currently selected date, i.e. `moment()`
feedback	string	web only	undefined	A detailed description of validation error/success

Name	Type	Platform	Default	Description
				or additional instructions. Visual variant is determined based on the `validation` prop.
inline	bool	web only	false	A flag determining if the calendar picker is standalone or an input with overlay
isDayDisabled	func	web only	undefined	A function determining whether a given date should be disabled @param {Moment} date The date to optionally disable @returns {bool}
label	string	web only	undefined	The field label to be displayed above the calendar

Name	Type	Platform	Default	Description
hint	string	web only	undefined	A short description of the expected input.
tooltip	string	web only	undefined	Content of an optional `Tooltip`. If set, a tooltip button will be shown next to the label.
onDateChange	func	web only	() => {}	Event triggered every time a new date is clicked on @param {Moment} date The new date that was selected
validation	'error' 'success'	web only	undefined	Use to visually mark an input as valid or invalid.
disabled	bool	web only	false	Disable the input which will not open the calendar picker

Name	Type	Platform	Default	Description
tokens	tokens	web only	undefined	System tokens prop, see tokens for more details
prevTestID	string	web only	""	A unique identifier for the previous icon button located on the datepicker. This is for automation testing purposes. Will be added as a `data-testid-prev` attribute for example.
nextTestID	string	web only	""	A unique identifier for the next icon button located on the datepicker. This is for automation testing purposes. Will be added as a `data-testid-next` attribute for example.

Name	Type	Platform	Default	Description
id*	string	web only		A unique identifier.

Tokens

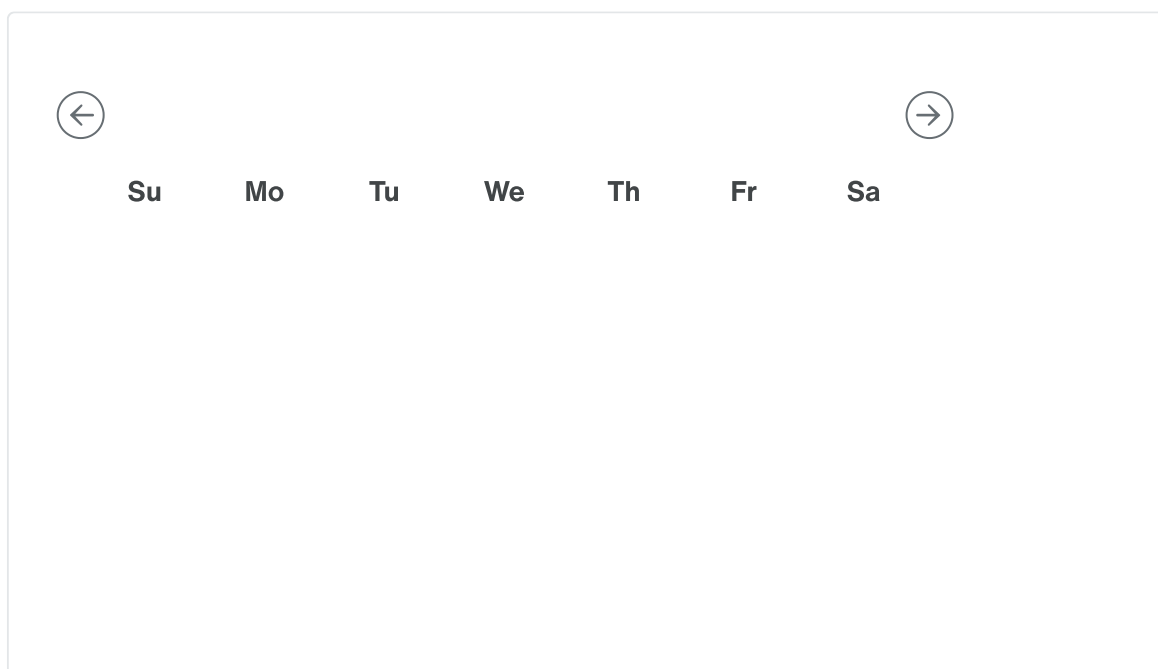
In exceptional circumstances, the following tokens can be passed to this component to override its default styles. **Do not do this unless absolutely necessary.** [Read more about overriding styles.](#)

► View Tokens

Variants

Inline

- Use Inline to display the DatePicker in a larger viewable area; allows the customer to quickly and easily date availability
- Does not include an input form field
- Optimized for keyboard interaction and tablet touch
- Recommended for viewports greater than or equal to 576px



October 2023

```
<DatePicker id="DatePicker" inline />
```

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](#)