

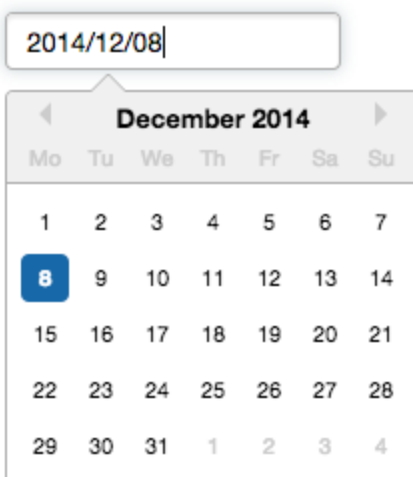
[Pro](#) [Teams](#) [Pricing](#) [Documentation](#)**npm**

Search

# react-datepicker TS

7.3.0 • Public • Published a month ago [Readme](#) [Code](#) Beta [5 Dependencies](#) [3,157 Dependents](#) [185 Versions](#)

## React Date Picker

npm package 7.3.0 Test suite failing codecov 97% downloads 9.3M/monthA simple and reusable Datepicker component for React ([Demo](#))

# Installation

---

The package can be installed via **npm**:

```
npm install react-datepicker --save
```

Or via **yarn**:

```
yarn add react-datepicker
```

You'll need to install React and PropTypes separately since those dependencies aren't included in the package. If you need to use a locale other than the default en-US, you'll also need to import that into your project from date-fns (see Localization section below). Below is a simple example of how to use the DatePicker in a React view. You will also need to require the CSS file from this package (or provide your own). The example below shows how to include the CSS from this package if your build system supports requiring CSS files (Webpack is one that does).

```
import React, { useState } from "react";
import DatePicker from "react-datepicker";

import "react-datepicker/dist/react-datepicker.css";

// CSS Modules, react-datepicker-cssmodules.css
// import 'react-datepicker/dist/react-datepicker-cssmodules.css';

const Example = () => {
  const [startDate, setStartDate] = useState(new Date());
  return (
    <DatePicker selected={startDate} onChange={(date) => setStartDate(
    );
  );
};
```

# Configuration

The most basic use of the DatePicker can be described with:

```
<DatePicker selected={startdate} onChange={(date) => setStartDate(date
```



You can use `onSelect` event handler which fires each time some calendar date has been selected

```
<DatePicker
  selected={date}
  onSelect={handleDateSelect} //when day is clicked
  onChange={handleDateChange} //only when value has changed
/>
```

`onClickOutside` handler may be useful to close datepicker in `inline` mode

See [here](#) for a full list of props that may be passed to the component. Examples are given on the [main website](#).

## Time picker

You can also include a time picker by adding the `showTimeSelect` prop

```
<DatePicker
  selected={date}
  onChange={handleDateChange}
  showTimeSelect
  dateFormat="Pp"
/>
```

Times will be displayed at 30-minute intervals by default (default configurable via `timeIntervals` prop)

More examples of how to use the time picker are given on the [main website](#)

## Localization

The date picker relies on **date-fns internationalization** to localize its display components. By default, the date picker will use the locale globally set, which is English. Provided are 3 helper methods to set the locale:

- **registerLocale** (string, object): loads an imported locale object from date-fns
- **setDefaultLocale** (string): sets a registered locale as the default for all datepicker instances
- **getDefaultLocale**: returns a string showing the currently set default locale

```
import { registerLocale, setDefaultLocale } from "react-datepicker";
import { es } from 'date-fns/locale/es';
registerLocale('es', es)
```

```
<DatePicker
  locale="es"
/>
```

Locales can be changed in the following way:

- **Globally** - `setDefaultLocale('es');`

## Compatibility

### React

We're always trying to stay compatible with the latest version of React. We can't support all older versions of React.

Latest compatible versions:

- React 16 or newer: React-datepicker v2.9.4 and newer
- React 15.5: React-datepicker v2.9.3
- React 15.4.1: needs React-datepicker v0.40.0, newer won't work (due to react-onclickoutside dependencies)

- React 0.14 or newer: All above React-datepicker v0.13.0
- React 0.13: React-datepicker v0.13.0
- pre React 0.13: React-datepicker v0.6.2

## Moment.js

Up until version 1.8.0, this package was using Moment.js. Starting v2.0.0, we switched to using `date-fns`, which uses native Date objects, to reduce the size of the package. If you're switching from 1.8.0 to 2.0.0 or higher, please see the updated example above or check out the [examples site](#) for up to date examples.

## Browser Support

The date picker is compatible with the latest versions of Chrome, Firefox, and IE10+.

Unfortunately, it is difficult to support legacy browsers while maintaining our ability to develop new features in the future. For IE9 support, it is known that the [classlist polyfill](#) is needed, but this may change or break at any point in the future.

## Local Development

---

The `main` branch contains the latest version of the Datepicker component.

To begin local development:

1. Run `yarn link` from project root
2. Run `cd docs-site && yarn link react-datepicker`
3. Run `yarn install` from project root
4. Run `yarn build` from project root
5. Run `yarn start` from project root

The last step starts documentation app as a simple webserver on <http://localhost:3000>.

You can run `yarn test` to execute the test suite and linters. To help you develop the component we've set up some tests that cover the basic functionality (can be found in `/tests`). Even though we're big fans of testing, this only covers a small piece of the component. We highly recommend you add tests when you're adding new functionality.

Please refer to `CONTRIBUTING.md` file for more details about getting set up.

## The examples

The examples are hosted within the docs folder and are ran in the simple app that loads the Datepicker. To extend the examples with a new example, you can simply duplicate one of the existing examples and change the unique properties of your example.

# Accessibility

---

## Keyboard support

- *Left*: Move to the previous day.
- *Right*: Move to the next day.
- *Up*: Move to the previous week.
- *Down*: Move to the next week.
- *PgUp*: Move to the previous month.
- *Shift+PgUp*: Move to the same day and month of the previous year. If that day does not exist, moves focus to the last day of the month.
- *PgDn*: Move to the next month.
- *Shift+PgDn*: Move to the same day and month of the next year. If that day does not exist, moves focus to the last day of the month.
- *Home*: Move to the first day (e.g Sunday) of the current week.
- *End*: Move to the last day (e.g. Saturday) of the current week.
- *Enter/Esc/Tab*: close the calendar. (Enter & Esc calls preventDefault)

## For month picker

---

- *Left*: Move to the previous month.
- *Right*: Move to the next month.
- *Enter*: Select date and close the calendar

# License

---

Copyright (c) 2014-2024 HackerOne Inc. and individual contributors. Licensed under MIT license, see **LICENSE** for the full license.


# Keywords

react   datepicker   calendar   date   react-component


## Install

```
> npm i react-datepicker
```

## Repository

 [github.com/Hacker0x01/react-datepicker](https://github.com/Hacker0x01/react-datepicker)

## Homepage

 [github.com/Hacker0x01/react-datepicker](https://github.com/Hacker0x01/react-datepicker)

## Weekly Downloads

1,571,530



Version

7.3.0

License

MIT

Unpacked Size

2.04 MB

Total Files

62

Issues

287

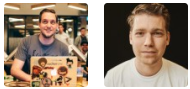
Pull Requests

12

## Last publish

a month ago

## Collaborators



>Try on RunKit

🚩Report malware



## Support

[Help](#)

[Advisories](#)

[Status](#)

[Contact npm](#)

## Company

[About](#)

[Blog](#)

[Press](#)



## Terms & Policies

[Policies](#)

[Terms of Use](#)

[Code of Conduct](#)

[Privacy](#)