**[React Blueprint](https://www.geeksforgeeks.org/reactjs-blueprint-introduction/)** is a UI toolkit for building web applications with React.js, and DateInput2 is a component in that toolkit for working with dates. Localization in this context refers to the process of adapting the user interface of an application to support different languages and regions.

In the context of React Blueprint's DateInput2 component, localization means configuring the component to display dates in a format that is appropriate for a specific language or region. This can include customizing the date format, specifying the order of day, month, and year, and translating month and day names.

React Blueprint provides built-in support for localization, and DateInput2 can be easily localized using the **locale** prop. You can set the locale prop to a string that identifies the desired locale, and the component will automatically format the date accordingly.

**Syntax:**

import { DateInput2 } from "@blueprintjs/datetime2";

function MyComponent() {

return (

<DateInput2 locale="fr" />

);

}

**Creating a React Application and Installing the Module:**

**Step 1:** Run the npm command for creating a react app:

npx create-react-app appname

**Step 2:** Navigate into the folder using the cd command:

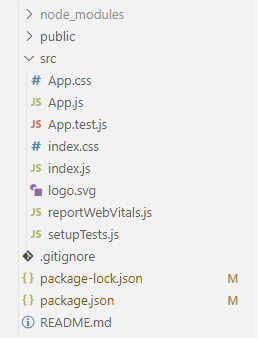
cd appname

**Step 3:** Install Blueprint Module with the following command:

npm install @blueprintjs/core

npm install @blueprintjs/datetime @blueprintjs/datetime2

**Project Structure:** The project folder structure will look approximate to the one provided below:



**Example 1:** This example shows how to use the DateInput2 component to display a date input field and update the state when the value changes.

Javascript

import { DateInput2 } from "@blueprintjs/datetime2";

import { useState } from "react"

const App = () => {

const [date, setDate] = useState(null);

const handleDateChange = (newDate) => {

setDate(newDate);

};

return (

<DateInput2

value={date}

onChange={handleDateChange}

locale="fr-FR"

/>

);

}

export default App;

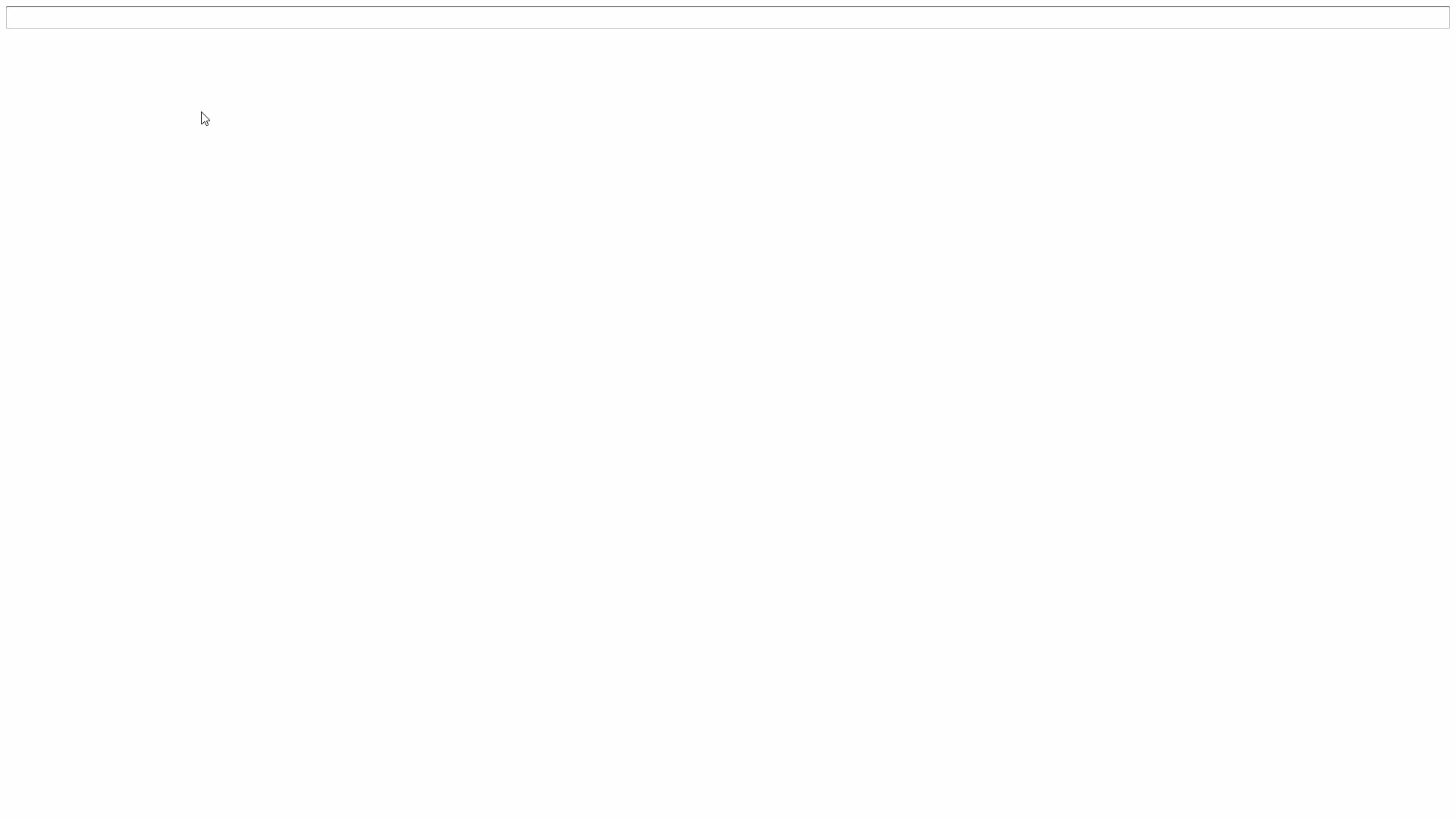
**Steps to run the application:**

**Step 1:** Run the application using the following command from the base/ root directory of your project.

npm start

**Step 2:** Navigate to ***http://localhost:3000/*** in any browser

**Output:**



**Example 2:** In this example, we set the locale prop to 'en-US', which specifies the US English locale. This will affect the formatting of the date and the ordering of the day, month, and year in the calendar widget.

Javascript

import { DateInput2 } from "@blueprintjs/datetime2";

import { useState } from "react"

const App = () => {

const [date, setDate] = useState(null);

const handleDateChange = (newDate) => {

setDate(newDate);

};

return (

<DateInput2

value={date}

onChange={handleDateChange}

locale="en-US"

/>

);

}

export default App;

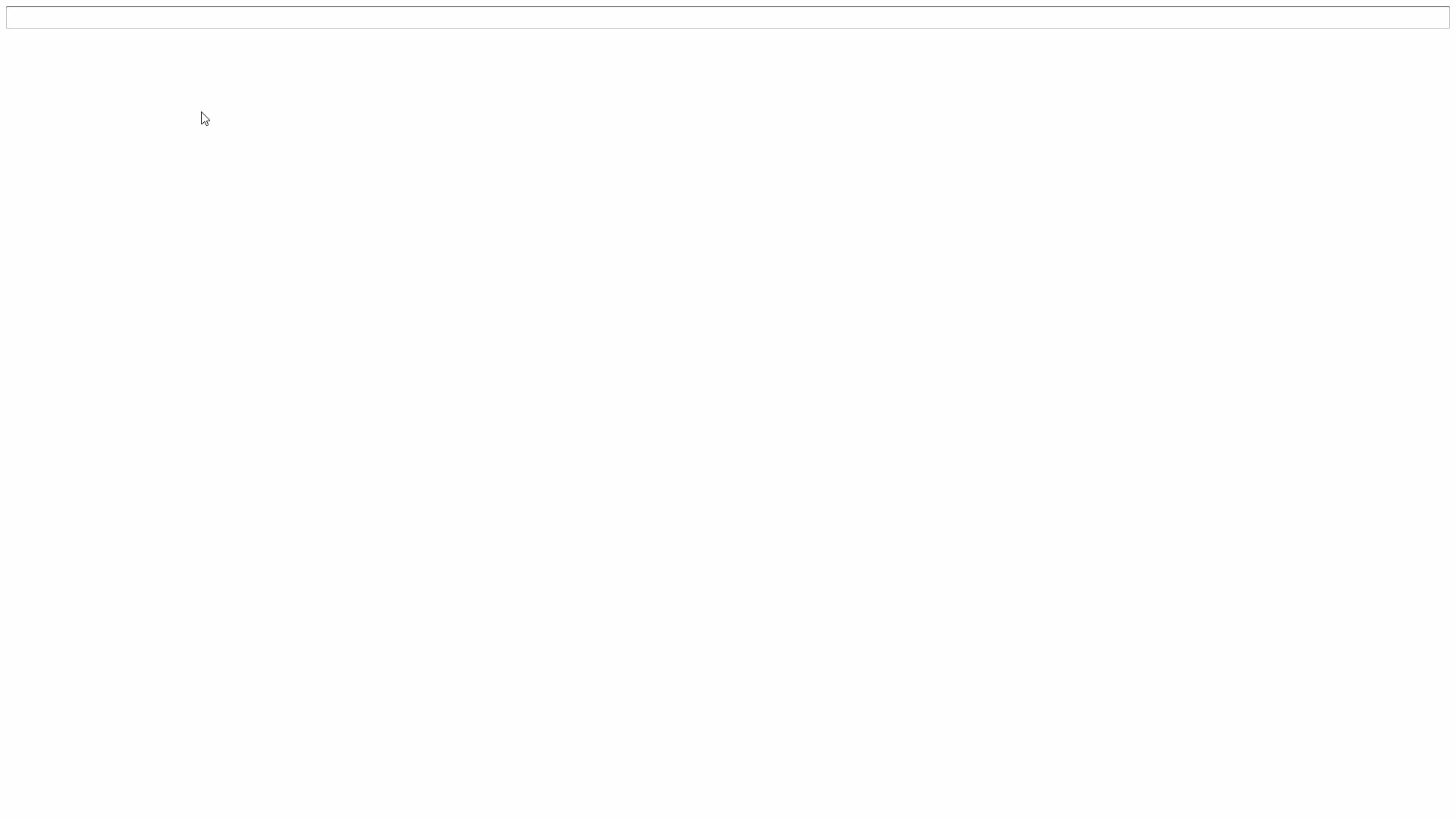
**Steps to run the application:**

**Step 1:** Run the following command in the terminal from the root of the project directory.

npm start

**Step 2:** Navigate to ***https://locahost:3000/*** *in any browser*

**Output:**



**Reference:** [https://blueprintjs.com/docs/#datetime-dateinput2](React.js%20Blueprint%20documentation:%20https:/blueprintjs.com/docs/#datetime-dateinput2)