



Desmond Nyamador

Alert Users to Errors with React Bootstrap

Desmond Nyamador

Oct 21, 2020 • 5 Min read • 462 Views

Oct 21, 2020 • 5 Min read • 462 Views

Web Development

Front End Web Dev...

Client-side Framew...

React

Introduction

3

- [Introduction](#)
- [React Bootstrap](#)
- [Set Up Development Environment](#)
- [Implementing the App](#)
- [Conclusion](#)

Introduction

Alerts are the ideal way to show feedback to a user when they either provide incorrect data or need to be informed of certain occurrences. In this guide, you'll use the React Bootstrap library to build an app that displays feedback based on the user's input.

- [Top ^](#)

React Bootstrap

Bootstrap is the most popular component-oriented frontend framework by far from the lovely people at Twitter. Unlike other frameworks like Tailwind CSS, which uses *utility classes*, Bootstrap has fully configurable UI elements that help you to build a full-blown app easily.

Set Up Development Environment

To get started, setup a React development environment by running the command below.

```
bash  
1      npx create-react-app <YOUR_APP_NAME> && cd <YOUR_APP_NAME>
```

Note: Replace <YOUR_APP_NAME> with your preferred app name.

You can also get a fully configured environment quickly by visiting <https://react.new> in your browser.

Next, add React Bootstrap to your dependencies by entering the following command in your terminal .

bash

```
1 npm install react-bootstrap bootstrap
```

There are two ways of importing components in React Bootstrap.

1. Importing individual components

bash

```
1 import Card from 'react-bootstrap/Card'
```

2. Destructuring

bash

```
1 import { Card } from 'react-bootstrap'
```

According to the documentation, the first method is advisable as "doing so pulls only the specific components that you use, which can significantly reduce the amount of code you end up sending to the client". (<https://react-bootstrap.github.io/getting-started/introduction>)

Implementing the App

This application prompts the user to enter a three-letter word and shows the user a success or error alert depending on the input.

First, add the application skeleton as shown below .

```
bash
1 import React, {useState} from "react";\
2
3 export default function App(){
4     const [value, setValue] = useState("");
5     const [isValid, setIsValid] = useState(false);
6
7     function handleSubmission(){}
8
9     return (
10     <div className="App">
11         <div>
12             <h1>Word Master</h1>
13             <p>Enter A Three Letter Word</p>
14             <input type="text" onChange={(e) => setValue(e.target.value)} value={value}>
15             <button onClick={handleSubmission}>Submit</button>
16         </div>
17     </div>
18 );
19 }
```

Next, implement the `handleSubmission` function to validate the input from the user, as shown below.

bash

```
import React, {useState} from "react";
import Alert from 'react-bootstrap/Alert';
import 'bootstrap/dist/css/bootstrap.min.css';

export default function App(){
    const [value, setValue] = useState("");
    const [isValid, setIsValid] = useState(false);

    function handleSubmission(){
        if (value.length > 3 || value.length < 3){
            setIsValid(false)
        }else{
            setIsValid(true)
        }
    }

    return (
        <div className="App">
        {isValid
            ? <Alert variant="success">Hurray! You're a genius.</Alert>
            : <Alert variant="danger">Oops! Try again</Alert>
        }
        <div>
            <h1>Word Master</h1>
            <p>Enter A Three Letter Word</p>
            <input type="text" onChange={(e) => setValue(e.target.value)} value={value}>
            <button onClick={handleSubmission}>Submit</button>
        </div>
        </div>
    );
}


```

```
29  
30  
31
```

Finally, add the control flow to display an error or success message using the React Bootstrap alert component.

bash

```
1 import React, {useState} from "react";  
2  
3 export default function App(){  
4     const [value, setValue] = useState("");  
5     const [isValid, setIsValid] = useState(false);  
6  
7     function handleSubmission(){  
8         if (value.length > 3 || value.length < 3){  
9             setIsValid(false)  
10        }else{  
11            setIsValid(true)  
12        }  
13    }  
14  
15    return (  
16        <div className="App">  
17            <div>  
18                <h1>Word Master</h1>  
19                <p>Enter A Three Letter Word</p>  
20                <input type="text" onChange={(e) => setValue(e.target.value)} value={  
21                    <button onClick={handleSubmission}>Submit</button>  
22                }>  
23            </div>  
24        </div>
```

```
24      );
25  }
```

Now enter a word and check out the feedback received from the world-class word master app you just built.

Conclusion

And that's a wrap! In this guide, you learned how to use the React Bootstrap alert component to display feedback to a user based on the length of the input. You should try as much as possible to implement this in many other ways to help you gain a better understanding. Read more on React Bootstrap [here](#). Stay safe!