

Using React.js in a Laravel Application

August 29, 2022

[reactjs](#)

[laravel](#)



Using React in a Laravel Application

Have you ever wondered, “**Should I Learn Laravel or React.js?**”, “**Is using React in a Laravel application a good idea?**”, “**Can I** **integrate React with Laravel?**” Has any of these questions ever crossed your mind?

Yes! They’re all good questions but let’s take it one step at a time. In this tutorial, we will take you through whether **React Native and Laravel for future mobile development** can work wonders.

So let’s get started.

What is Laravel?

If you are artistically inclined, Laravel suits your tastes just right. Laravel is a free, open-source PHP web framework that is intended for the development of web applications following the model–view–controller architectural pattern.

- It’s all about clean code, wonderful functionality, simple, and elegant syntax that makes it very easy to use.
- All features are carefully considered and offered to developers so that you obtain an effortless and enjoyable development experience.

Install Laravel Project

To know how to integrate Laravel with React and install React in Laravel, we first need to install a brand new Laravel project.

```
composer create-project laravel/laravel --prefer-dist laravel-react-js
```

Get into the Laravel project.

```
cd laravel-react-js
```

Install laravel/ui:

Laravel/UI is an official package that offers a login and registration user interface scaffolding for React, Vue.js, and Svelte.

Run the below command to install the Laravel ui composer package.

```
composer require laravel/ui
```

Install React in Laravel

Now that we have installed the composer UI package, we have to run the below-given command to install React.

```
php artisan ui react
```

Install Required Packages

We need Node and NPM installed on our development machine.

We can check the Node and NPM installation using the below command.

```
# for node
```

```
node -v
```

```
# for npm
```

```
npm -v
```

The following command installs all the javascript packages which are required for our project.

```
function User() {  
  
    return (  
  
        <div className="container mt-5">  
  
            <div className="row justify-content-center">  
  
                <div className="col-md-8">  
  
                    <div className="card text-center">  
  
                        <div className="card-header"><h2>React Component in Laravel</h2></div>  
  
                        <div className="card-body">I'm tiny React component in Laravel app!</div>  
  
                    </div>  
  
                </div>  
  
            </div>  
  
        </div>  
  
    );  
  
}  
  
export default User;  
  
if (document.getElementById('user')) {  
  
    ReactDOM.render(<User />, document.getElementById('user'));  
  
}
```

We have to register the React component inside the resources/js/app.js file.

```
<head>
```

```
<meta charset="utf-8">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1">
```

```
<title>Laravel</title>
```

```
<!-- Styles -->
```

```
<link href="{{ asset('css/app.css') }}" rel="stylesheet">
```

```
</head>
```

```
<body>
```

```
<!-- React root DOM -->
```

```
<div id="user">
```

```
</div>
```

```
<!-- React JS -->
```

```
<script src="{{ asset('js/app.js') }}" defer></script>
```

```
</body>
```

```
</html>
```

Finally, we have to run the command to compile the JavaScript and CSS files for the Laravel and React.js

```
npm run watch
```

Our code has been compiled. Now run the command to view our app in the browser.

- React is declarative in nature which means it enables significant data changes that cause automatic updates to user interfaces.

Since React comes with a progressive functionality, there is no extra function that you need to perform to

The Downside:

- The JavaScript extension syntax or JSX in React.js is not preferred by many web developers due to its verbose process.
- There are new tools and documents that are frequently released with React, and they make it difficult to manage the codes are.
- The architecture of React undergoes plenty of frequent changes.

Conclusion

In conclusion, Laravel is a trending framework for a reason and React.js is one of the best frontend libraries. On the client side, the user experience is very good and we can use the power of JS in our project smoothly. Your project will run smoothly and the user will have a better experience while using your application.

If you need any more information on **how to integrate React with Laravel, Laravel 8 with React JS tutorial**, get in touch with [our experts](#) for professional guidance today!