

Join GitHub today

Dismiss

GitHub is home to over 20 million developers working together to host and review code, manage projects, and build software together.

Sign up

Create React apps with no build configuration.

#react #zero-configuration #build-tools

1,449 commits

7 branches

164 releases

449 contributors

MIT

Branch: next

New pull request

Find file

Clone or download

viankakrisna committed with Timer Ensure workspaces key exists in parent package.json (#3989)		Latest commit 47d2d94 2 days ago
.github	Fix links to package (#3898)	17 days ago
packages	Ensure workspaces key exists in parent package.json (#3989)	2 days ago
tasks	Add support for yarn and lerna monorepos. (#3741)	7 days ago
.eslintignore	Fix internal linting setup and add missing headers (#1741)	11 months ago
.eslintrc	[Internal] Warn for missing brackets (#2151)	9 months ago
.gitignore	NPM version check for tip (#1193)	a year ago
.travis.yml	Add support for yarn and lerna monorepos. (#3741)	7 days ago
.yarnrc	Enable Yarn check files (#3769)	27 days ago
CHANGELOG-0.x.md	Fix links to package (#3898)	17 days ago
CHANGELOG.md	Fix links to package (#3898)	17 days ago
CODE_OF_CONDUCT.md	Add an explicit link to Code of Conduct (#3781)	26 days ago
CONTRIBUTING.md	Fix links to package (#3898)	17 days ago
LICENSE	BSD+Patents -> MIT (#3189)	5 months ago
README.md	Fix links to package (#3898)	17 days ago
appveyor.cleanup-cache.txt	Clear cache	22 days ago
appveyor.yml	Add support for yarn and lerna monorepos. (#3741)	7 days ago
lerna.json	Fix package repository name (#3900)	17 days ago
package.json	Add svg rendering for error example (#3825)	22 days ago
screencast-error.svg	Add svg rendering for error example (#3825)	22 days ago
screencast.svg	Update screencast to use npx (#3810)	23 days ago

README.md

Create React App

build passing

Create React apps with no build configuration.

- [Creating an App](#) – How to create a new app.
- [User Guide](#) – How to develop apps bootstrapped with Create React App.

Create React App works on macOS, Windows, and Linux.

If something doesn't work, please [file an issue](#).

## Quick Overview

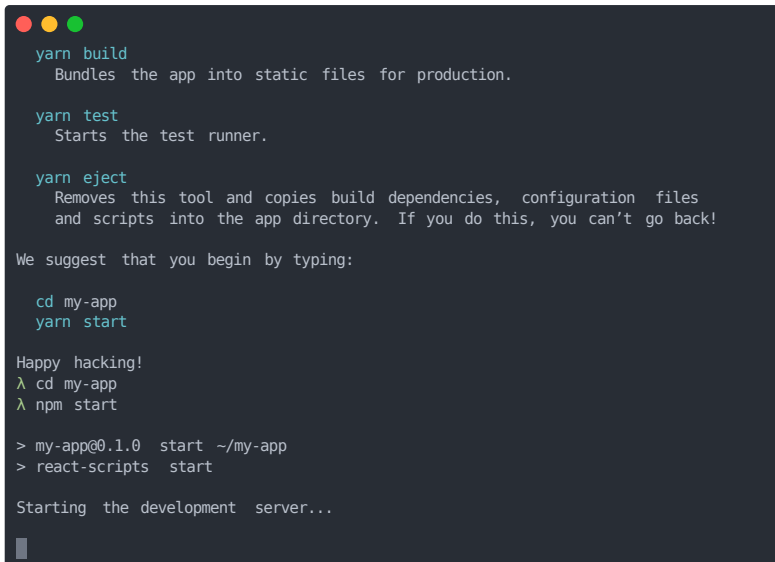
---

```
npx create-react-app my-app
cd my-app
npm start
```

(*npx* comes with *npm* 5.2+ and higher, see [instructions for older npm versions](#))

Then open <http://localhost:3000/> to see your app.

When you're ready to deploy to production, create a minified bundle with `npm run build`.



```
yarn build
  Bundles the app into static files for production.

yarn test
  Starts the test runner.

yarn eject
  Removes this tool and copies build dependencies, configuration files
  and scripts into the app directory. If you do this, you can't go back!

We suggest that you begin by typing:

  cd my-app
  yarn start

Happy hacking!
λ cd my-app
λ npm start

> my-app@0.1.0 start ~/my-app
> react-scripts start

Starting the development server...
```

## Get Started Immediately

You **don't** need to install or configure tools like Webpack or Babel.

They are preconfigured and hidden so that you can focus on the code.

Just create a project, and you're good to go.

## Creating an App

---

**You'll need to have Node  $\geq 6$  on your local development machine** (but it's not required on the server). You can use [nvm](#) (macOS/Linux) or [nvm-windows](#) to easily switch Node versions between different projects.

To create a new app, run a single command:

```
npx create-react-app my-app
```

(*npx* comes with *npm* 5.2+ and higher, see [instructions for older npm versions](#))

It will create a directory called `my-app` inside the current folder.

Inside that directory, it will generate the initial project structure and install the transitive dependencies:

```
my-app
├── README.md
├── node_modules
├── package.json
├── .gitignore
├── public
│   ├── favicon.ico
│   ├── index.html
│   └── manifest.json
└── src
    ├── App.css
    ├── App.js
    ├── App.test.js
    ├── index.css
    ├── index.js
    ├── logo.svg
    └── registerServiceWorker.js
```

No configuration or complicated folder structures, just the files you need to build your app.

Once the installation is done, you can open your project folder:

```
cd my-app
```

Inside the newly created project, you can run some built-in commands:

### **npm start Or yarn start**

Runs the app in development mode.

Open <http://localhost:3000> to view it in the browser.

The page will automatically reload if you make changes to the code.

You will see the build errors and lint warnings in the console.

A screenshot of a terminal window with a dark background. At the top, it says "Failed to compile." in red. Below that, it shows the file path "/src/App.js" and the error message "Syntax error: Unexpected token (8:7)". The error points to line 8 of the file. The code shown is a render function that returns a JSX element. On line 8, there is a closing tag for a div: "</div>". The error message indicates that the closing tag is not properly closed. The code continues with a header, an image, and a heading.

```
Failed to compile.

./src/App.js
Syntax error: Unexpected token (8:7)

   6 |   render() {
   7 |     return (
>  8 |       <<div className="App">
      |         ^
   9 |         <header className="App-header">
  10 |           <img src={logo} className="App-logo" alt="logo" />
  11 |           <h1 className="App-title">Welcome to React</h1>
```

### **npm test Or yarn test**

Runs the test watcher in an interactive mode.  
By default, runs tests related to files changed since the last commit.

[Read more about testing.](#)

### **npm run build or yarn build**

Builds the app for production to the `build` folder.  
It correctly bundles React in production mode and optimizes the build for the best performance.

The build is minified and the filenames include the hashes.  
By default, it also [includes a service worker](#) so that your app loads from local cache on future visits.

Your app is ready to be deployed.

## User Guide

---

The [User Guide](#) includes information on different topics, such as:

- [Updating to New Releases](#)
- [Folder Structure](#)
- [Available Scripts](#)
- [Supported Browsers](#)
- [Supported Language Features and Polyfills](#)
- [Syntax Highlighting in the Editor](#)
- [Displaying Lint Output in the Editor](#)
- [Formatting Code Automatically](#)
- [Debugging in the Editor](#)
- [Changing the Page `<title>`](#)
- [Installing a Dependency](#)
- [Importing a Component](#)
- [Code Splitting](#)
- [Adding a Stylesheet](#)
- [Post-Processing CSS](#)
- [Adding a CSS Preprocessor \(Sass, Less etc.\)](#)
- [Adding Images, Fonts, and Files](#)
- [Using the `public` Folder](#)
- [Using Global Variables](#)
- [Adding Bootstrap](#)
- [Adding Flow](#)
- [Adding a Router](#)
- [Adding Custom Environment Variables](#)
- [Can I Use Decorators?](#)
- [Fetching Data with AJAX Requests](#)
- [Integrating with an API Backend](#)
- [Proxying API Requests in Development](#)
- [Using HTTPS in Development](#)
- [Generating Dynamic `<meta>` Tags on the Server](#)
- [Pre-Rendering into Static HTML Files](#)
- [Running Tests](#)
- [Debugging Tests](#)
- [Developing Components in Isolation](#)

- [Publishing Components to npm](#)
- [Making a Progressive Web App](#)
- [Analyzing the Bundle Size](#)
- [Deployment](#)
- [Advanced Configuration](#)
- [Troubleshooting](#)

A copy of the user guide will be created as `README.md` in your project folder.

## How to Update to New Versions?

---

Please refer to the [User Guide](#) for this and other information.

## Philosophy

---

- **One Dependency:** There is just one build dependency. It uses Webpack, Babel, ESLint, and other amazing projects, but provides a cohesive curated experience on top of them.
- **No Configuration Required:** You don't need to configure anything. Reasonably good configuration of both development and production builds is handled for you so you can focus on writing code.
- **No Lock-In:** You can "eject" to a custom setup at any time. Run a single command, and all the configuration and build dependencies will be moved directly into your project, so you can pick up right where you left off.

## What's Included?

---

Your environment will have everything you need to build a modern single-page React app:

- React, JSX, ES6, and Flow syntax support.
- Language extras beyond ES6 like the object spread operator.
- Autoprefixed CSS, so you don't need `-webkit-` or other prefixes.
- A fast interactive unit test runner with built-in support for coverage reporting.
- A live development server that warns about common mistakes.
- A build script to bundle JS, CSS, and images for production, with hashes and sourcemaps.
- An offline-first [service worker](#) and a [web app manifest](#), meeting all the [Progressive Web App](#) criteria.
- Hassle-free updates for the above tools with a single dependency.

Check out [this guide](#) for an overview of how these tools fit together.

The tradeoff is that **these tools are preconfigured to work in a specific way**. If your project needs more customization, you can "eject" and customize it, but then you will need to maintain this configuration.

## Popular Alternatives

---

Create React App is a great fit for:

- **Learning React** in a comfortable and feature-rich development environment.
- **Starting new single-page React applications.**
- **Creating examples** with React for your libraries and components.

Here's a few common cases where you might want to try something else:

- If you want to **try React** without hundreds of transitive build tool dependencies, consider [using a single HTML file or an online sandbox instead](#).
-

If you need to **integrate React code with a server-side template framework** like Rails or Django, or if you're **not building a single-page app**, consider using [nwb](#), or [Neutrino](#) which are more flexible. For Rails specifically, you can use [Rails Webpacker](#).

- If you need to **publish a React component**, [nwb](#) can [also do this](#), as well as [Neutrino's react-components preset](#).
- If you want to do **server rendering** with React and Node.js, check out [Next.js](#) or [Razzle](#). Create React App is agnostic of the backend, and just produces static HTML/JS/CSS bundles.
- If your website is **mostly static** (for example, a portfolio or a blog), consider using [Gatsby](#) instead. Unlike Create React App, it pre-renders the website into HTML at the build time.
- If you want to use **TypeScript**, consider using [create-react-app-typescript](#).
- Finally, if you need **more customization**, check out [Neutrino](#) and its [React preset](#).

All of the above tools can work with little to no configuration.

If you prefer configuring the build yourself, [follow this guide](#).

## Contributing

---

We'd love to have your helping hand on `create-react-app` ! See [CONTRIBUTING.md](#) for more information on what we're looking for and how to get started.

## React Native

---

Looking for something similar, but for React Native?

Check out [Create React Native App](#).

## Acknowledgements

---

We are grateful to the authors of existing related projects for their ideas and collaboration:

- [@eanplatter](#)
- [@insin](#)
- [@mxstbr](#)