

[Sign in](#)[Get started](#)**Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

8 React Application Deployment and Hosting Solutions for 2019

Useful solutions for deploying and hosting your React applications in the cloud.



Nwose Lotanna in Bits and Pieces

[Follow](#)

Dec 11, 2018 · 7 min read



Photo by John-Mark Smith on Unsplash

After working in your development environment and playing around with localhost, you most likely would need to build out your react application to become production ready and deploy it for everyone to experience the react goodness in their local machines too. In this article, we would look at a number of options to developers quickly pick from to deploy their react applications. Tip: Use open source tools like **Bit** (open source) to build faster with React components. Easily share and sync components across your projects and apps, and collaborate with your team over component collections. Give it a try.

[Sign in](#)[Get started](#)**bit Bits and Pieces**[WRITE](#)[COMPONENTS](#)[JAVASCRIPT](#)[WEBDEV](#)[REACT](#)**1. Firebase Hosting**

Firebase Hosting

Firebase Hosting provides fast and secure hosting for your web app and your static and dynamic content.

Firebase Hosting is production-grade web content hosting for developers. With a single command, you can quickly and easily deploy web apps and serve both static and dynamic content to a global content delivery network (CDN).

[GET STARTED](#)

Key capabilities

Firebase Hosting

Firebase Hosting is part of the many services Google's firebase offers like authentication, storage, cloud functions and even database. This is basically a static site hosting that would be perfect for any frontend react application, it also supports SSL, CDN and custom domains.

To use firebase hosting:

- Ensure you have a google account
- Login to Firebase Console
- Create a project
- Navigate to the hosting tab and click get started.

After this, you proceed to install firebase cli on your terminal with this line of command:

```
npm install -g firebase-tools
```

This would install firebase tools globally and you can now deploy using these commands sequentially in your terminal:

```
// cd to your project folder
firebase init
firebase deploy
```

Firebase would deploy your application to yourProjectName.firebaseio.com

2. Github Pages

GitHub Pages

Websites for you and your projects.

[Sign in](#)[Get started](#)**Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT



Github Pages

React applications by default when you build out for production gives you github pages as a great deployment option. Github pages is a deployment service by github that builds and deploys your react application straight from the application repository with a smooth git integration, it is such a breeze to use. When you run:

```
yarn build
```

React builds out your application for production and afterward shows you a suggestion if you choose to use github pages. Copy the homepage description given, it would look like this:

```
"homepage" : "https://yourgithubname.github.io/yourreponame",
```

Add it to your package.json file, then re-run the build command:

```
yarn build
```

Go to the package.json and add a new line under scripts

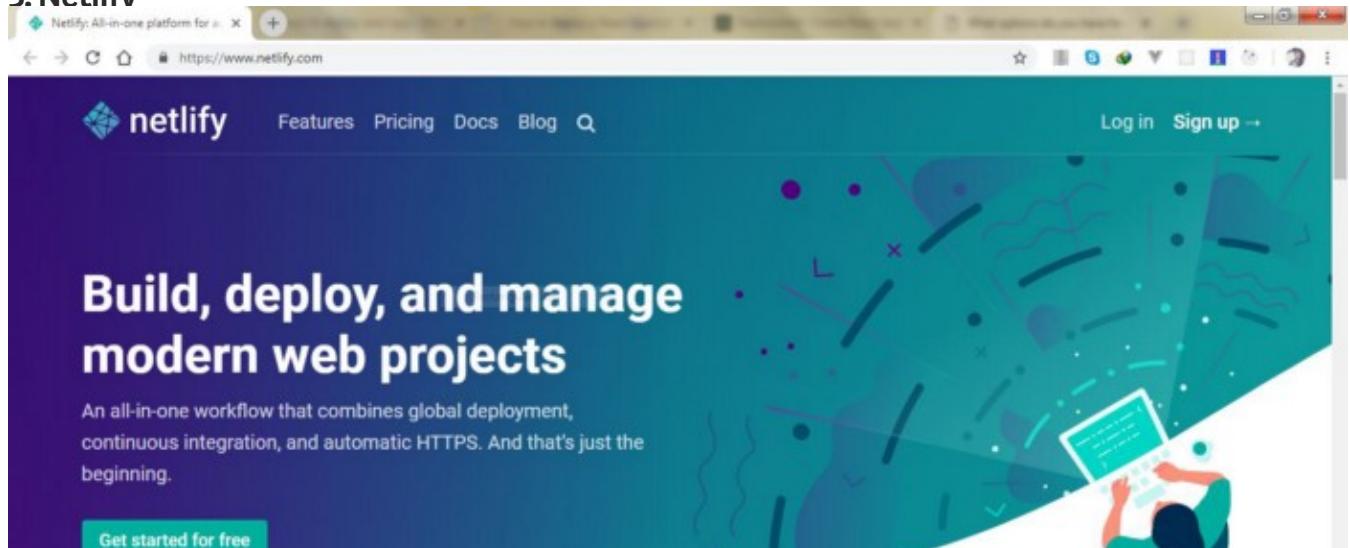
```
"deploy": "gh-pages -d build"
```

Then deploy with one line of command:

```
npm run deploy
```

Open the homepage on your browser and see the react application is live.

3. Netlify



[Sign in](#)[Get started](#)**Bits and Pieces**[WRITE](#)[COMPONENTS](#)[JAVASCRIPT](#)[WEBDEV](#)[REACT](#)

Deploying with Netlify is one of the the trendiest ways to put your react application out in the universe. Netlify provides a very stress-free and easy-to-use interface where you literally just click and select options and configure deployments and build your app from a development environment from a repository like github, you can even test deploys and also get support for A/B testing from the same user interface.

To get started:

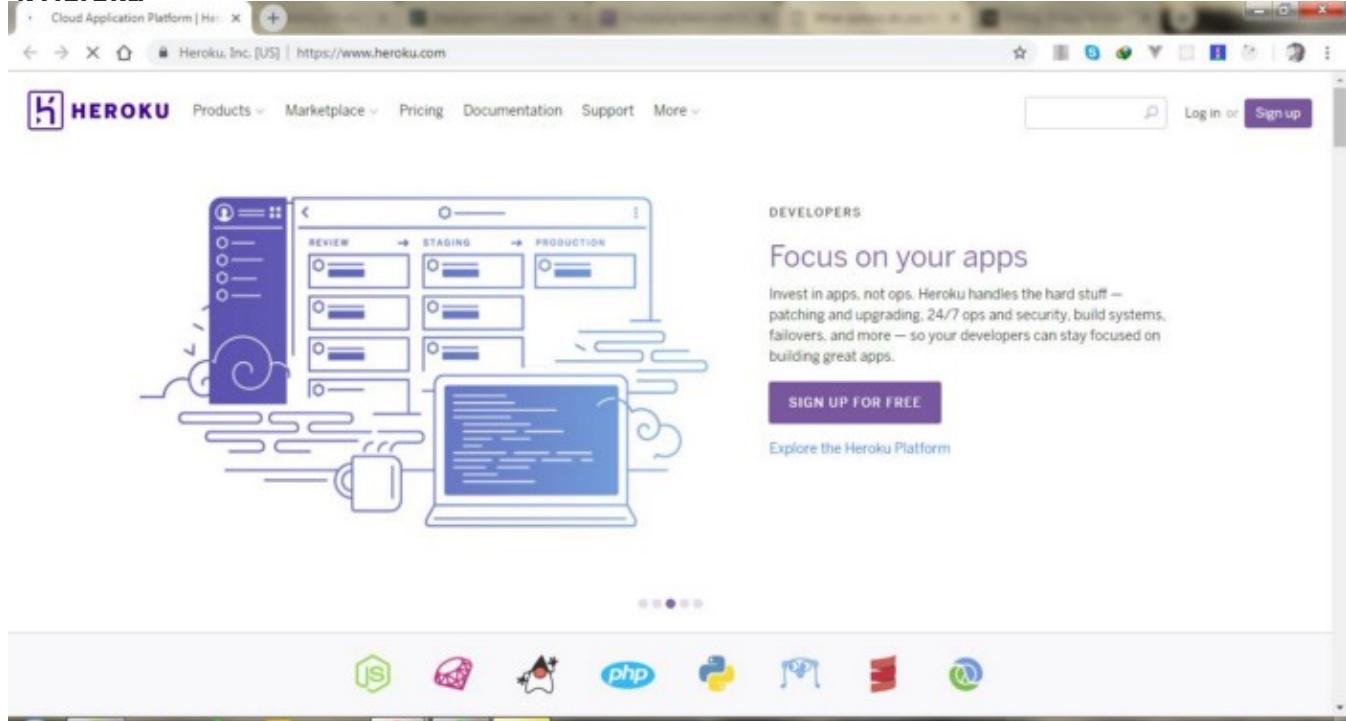
- create a netlify account
- login to the created account
- click new site from git on the top right corner.
- choose github (if that's where your project is)
- follow the prompt to add navigation and test deploy
- hit deploy site button

This can be achieved manually too in your terminal with a few commands

```
npm install netlify-cli -g
netlify deploy
```

And you would have a netlify.com project deployed successfully.

4. Heroku



Heroku

Heroku, an industry leading cloud-platform-as-a-service that handles a number of cloud based operations which includes building, managing and deployment of applications is a great place to start deploying your react application. If you wish to use create-react-app to build out your react application, here is an easy way to deploy it on heroku:

- create a heroku account
- login to the created account

You can now open your terminal and run the following commands below:

[Sign in](#)[Get started](#)**Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

```
git commit -m "react-create-app on Heroku"
git push heroku master
heroku open
```

After this, you have successfully deployed your react application, check your dashboard for the application link.

5. Now

The screenshot shows the Now website homepage. At the top, there's a navigation bar with links for ABOUT, BLOG, NOW, DNS, DOMAINS, CON, DOCS, PRICING, CHAT, LOGIN, and JOIN FREE. Below the header, the word "Now" is prominently displayed. To the right of "Now" are links for Intro, Features, and GitHub. The main heading is "Now — Global Serverless Deployments". Below the heading, a sub-headline reads "Now makes serverless application deployment easy. Don't spend time configuring the cloud. Just push your code." There are two buttons: "LEARN MORE" and "START WITH GITHUB". At the bottom, there are four sections with icons and descriptions: "Deploy in seconds" (Now builds and deploys with maximum parallelization), "Monorepo ready" (Your frontend, your assets and your APIs in one place), "Built-in CDN" (A global CDN in front of all static and dynamic responses), and "No code changes" (Our build process makes your apps serverless with no extra effort).

Now

Ziet's Now is a revolutionary serverless deployment service for various kinds of applications including react applications all the way to node applications. It provides a very robust support for the CLI also and it is smoothly integrated with github. To get started:

- install now desktop here (it ships with the cli tool)
- alternatively use npm:

```
npm install -g now
```

- Create a new account on now
- Login with the newly created account.

To deploy a react application, navigate to the root folder of your built project and run the now command in the terminal:

```
now --name your-project-name
```

A now.json file should appear in the root folder looking like this:

```
{
  "version": 1,
  "type": "static",
```

[Sign in](#)[Get started](#)

bit

Bits and Pieces

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

```

}
}
```

Follow the prompt and you will successfully deploy your application.

6. Surge

The screenshot shows the Surge website at <https://surge.sh>. The page has a clean, modern design with a green and white color scheme. At the top, there's a navigation bar with links to Pricing, Tour, Help, Blog, and @surge_sh. The main heading is "Static web publishing for Front-End Developers". Below the heading, a subtext explains: "Simple, single-command web publishing. Publish HTML, CSS, and JS for free, without leaving the command line.". Underneath, there are three prominent statistics: "3,356,998 deployments", "37.80 TB published", and "456,510 projects".

Surge

Surge is a very popular hosting service for static applications like react applications tastefully targeted at frontend web developers. The free plan supports custom domains, making it a really great option for deploying react applications. To get started:

- install surge using npm

```
npm install -g surge
```

- build out your react application

```
cd your-react-project
npm run build
```

- navigate to build folder

```
cd build
```

- run surge

```
surge
```

Follow the prompt and you will be given a surge url where your react application is live at.

7. AWS S3

The screenshot shows the AWS S3 console. The top navigation bar contains links for "Contact Sales", "Support", "English", and "My Account".

[Sign in](#)[Get started](#)

bit

Bits and Pieces

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

Object storage built to store and retrieve any amount of data from anywhere

[Get started with Amazon S3](#)[Request more information](#)

Amazon Simple Storage Service (Amazon S3) is an object storage service that offers industry-leading scalability, data availability, security, and performance. This means customers of all sizes and industries can use it to store and protect any amount of data for a range of use cases, such as websites, mobile applications, backup and restore, archive, enterprise applications, IoT devices, and big data analytics. Amazon S3 provides easy-to-use management features so you can organize your data and configure finely-tuned access controls to meet your specific business, organizational, and compliance requirements. Amazon S3 is designed for 99.999999999%.



Amazon S3

Amazon web services has a suite of cloud services to support every single part of software development cycle. They have a couple of services for deployment of web applications and S3 is one of the less complicated one to use. It is often said that a lot of amazon web services sacrificed simplicity of getting started for flexibility of usage. There is however a cli tool built by a community member for easily deploying your react application to the amazon s3 service called s3cmd. To get started:

- sign up to aws services
- login to the created account
- open an s3 bucket according to your preferable data point/center
- Download s3cmd here

Note: Amazon web services has an AWS CLI tool but we are using the s3cmd tool in this article

Alternatively, if you have homebrew, you can install with this line of command:

```
brew install s3cmd
```

You need the access keys of the bucket for the next step. To get the keys lookup this guide. When you have gotten the keys, run this line of command to use them:

```
s3cmd --configure
```

when prompted, paste in the keys you already got. We go ahead to list all our buckets with this command:

```
s3cmd ls
```

Now let us build out ur react application

```
yarn build
```

To synchronise the build folder with our s3 bucket, we run this:

[Sign in](#)[Get started](#)**Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

```
yarn build
s3cmd sync build/* s3://reactApp
```

To make it even a little easier, you can add these commands to your package.json file so that you can run it all like a yarn command

```
"build-deploy": "yarn build && s3cmd sync build/* s3://reactApp &&
echo 'Deployed Successfully' "
```

8. Roast

The screenshot shows the Roast website homepage. At the top, there's a navigation bar with links for PRICING, DOCS, SUPPORT, and SIGN IN. Below the navigation, the heading 'Modern Web Hosting' is displayed. A sub-copy below it reads: 'Host any HTML website or JS single-page app on our global network over HTTPS and HTTP2 in seconds... and get ultra-fast pagespeed, auto-generated unfurls, SEO reports, server-side rendering, and more'. A 'SIGN UP FREE' button is visible on the left. On the right, there's a dark background image of a smartphone displaying a webpage.

Roast

Roast is a content delivery network static web host you can use to deploy your react applications. It is built with speed and simplicity in mind with provisions for https and full-page server-side render. To get started:

- Create an account on roast here
- login with your new account.
- Install the Roast CLI tool

```
npm install -g roast
```

Then build out your react application:

```
yarn build
```

Add a _redirects file. See a guide here. When that is done, you can deploy your react application by running a line of command:

```
roast deploy
```

[Sign in](#)[Get started](#)**bit****Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

DigitalOcean — Cloud Computing, Simplicity at Scale

Providing developers and businesses a reliable, easy-to-use cloud computing platform of virtual servers (Droplets)...

www.digitalocean.com

Content Delivery Network (CDN) | Low Latency, High Transfer Speeds, Video Streaming | Amazon...

Amazon CloudFront is a fast content delivery network (CDN) service that securely delivers data, videos, applications...

aws.amazon.com

Conclusion

In this article we took a good look at some easy options react developers should know they have when it comes to deployment of their react applications. You can finally just take that new step of stepping out of the localhost comfort zone and deploy your application for the world to see too.

Happy Coding!

Stay in the Loop

Get chosen updates on the best posts shared in Bits and Pieces: JavaScript, React, WebDev and more. We won't spam.

 Email Sign up

• • •

Learn more

State of React State Management for 2019

Observing state management in the wild... and the Store is open for business!

[Sign in](#)[Get started](#)**Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

5 tools to speed the development of your React application, focusing on components.

blog.bitsrc.io

11 React UI Component Libraries you Should Know in 2019

11 React component libraries with great components for building your next app's UI interface in 2019.

blog.bitsrc.io

React

Web Hosting

Cloud Computing

JavaScript

App Development



914 claps



•••



WRITTEN BY

Nwose Lotanna[Follow](#)

Twitter:@victotana || Web Developer and Writer []
victotana@gmail.com]

**Bits and Pieces**[Follow](#)

Coding in the age of code components. Learn more @ bit.dev

[See responses \(3\)](#)

[Sign in](#)[Get started](#)**Bits and Pieces**

WRITE

COMPONENTS

JAVASCRIPT

WEBDEV

REACT

How to Become a Better Developer

Chris...
Jun 1...

1.5K



SOLID Principles every Developer Should Know

Chidu...
Oct 9,...

27K



11 Javascript Utility Libraries you Should Know in 2019

Jonath...
May 3...

17.4K

