WordPress at the Edge

with

Next.js & the Jamstack

@colbyfayock

Colby Fayock

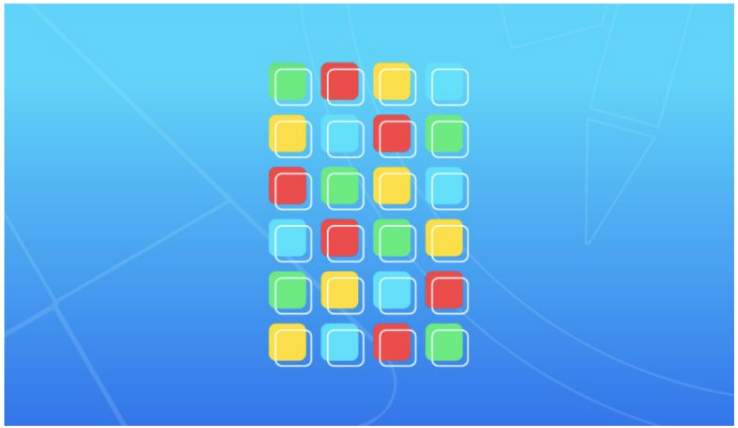
Developer Advocate

applitools

@colbyfayock

This Week in Apps: PayPal launches 'super app,' Twitter adds crypto tips, Apple won't take Fortnite back

Sarah Perez



Facebook's wearable glasses can succeed where Google Glass flopped

Ashish Bhatia

Aiming to create a gender-equitable startup landscape?

Priyanka Srinivas

When the biggest Disruption is to just sit down and focus

Natasha Mascarenhas

Extra Crunch roundup: Here's everything you missed at TechCrunch Disrupt 2021

Walter Thompson, Annie Saunders

The Latest

Singapore-based logistics startup Ninja Van secures $578M Series E

Kate Park
8:53 PM EDT • September 26, 2021

Ninja Van, a Singapore-based logistics startup, has closed a $578 million Series E round to support its operations infrastructure and technology systems growth. Investors include China's Alibaba Gr...

techcrunch.com

@colbyfayock

KIDS

STAR WARS

Search Star Wars

LOG IN // SIGN UP

NEWS + BLOG | VIDEO | FILMS | SERIES | INTERACTIVE | COMMUNITY | DATABANK | DISNEY+

LATEST FROM NEWS + BLOG

"IT'S A LOVE LETTER": THE PRODUCERS OF *STAR WARS: VISIONS* CELEBRATE THE UNIQUE ANIME COLLECTION

Executive producers James Waugh and Jacqui Lopez of Lucasfilm and producer Kanako Shirasaki pull back the curtain on creating the new Disney+ anime anthology series, streaming now.

BY: KRISTIN BAVER

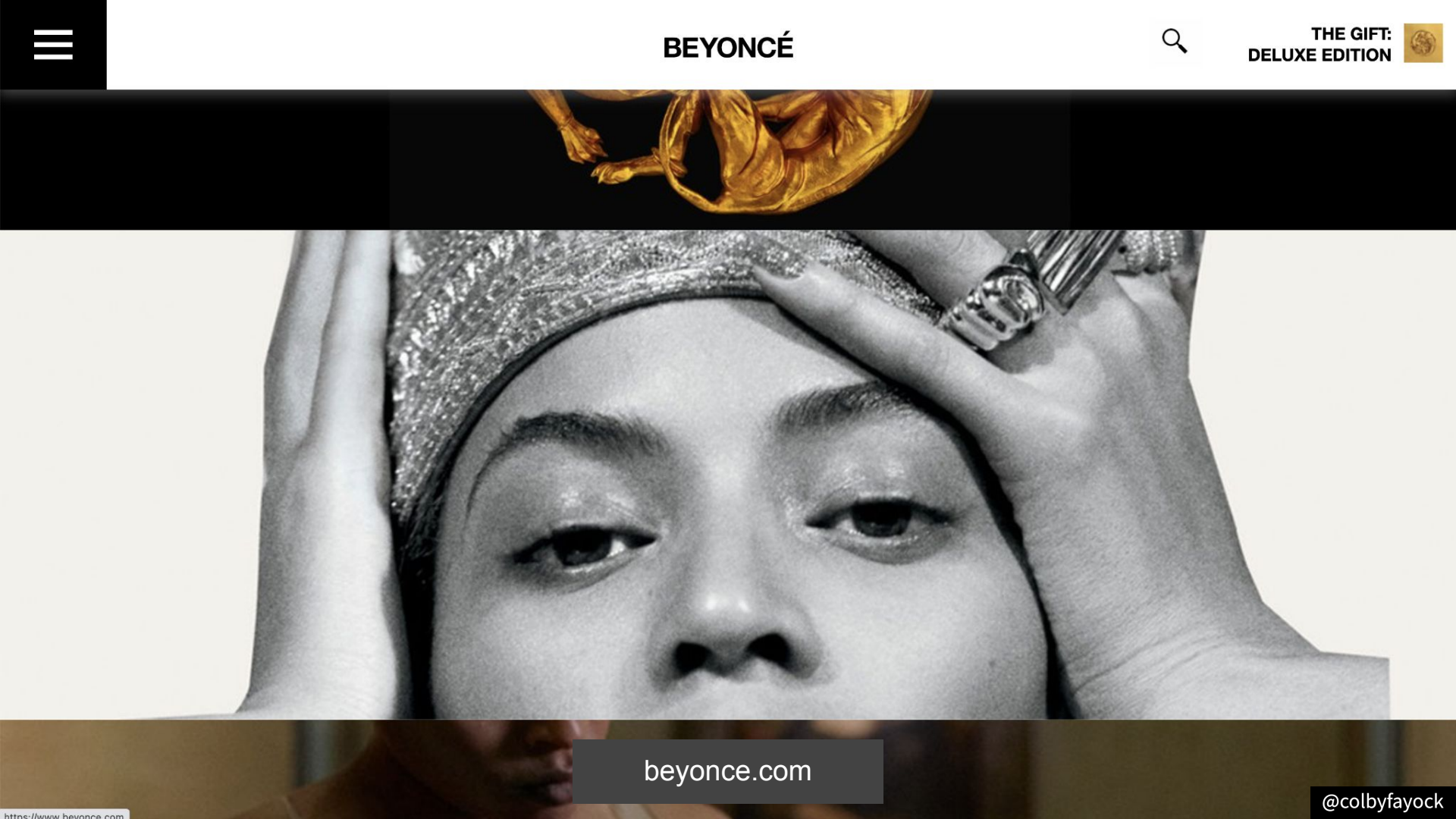
VISIONS // SEPTEMBER 24, 2021

starwars.com/blog

NEWS + BLOG CATEGORIES

- ALL
- BEHIND THE SCENES
- BOOKS + COMICS
- CHARACTERS + HISTORIES
- COLLECTING
- CREATIVITY
- DISNEY PARKS
- DISNEY+
- EVENTS
- FANS + COMMUNITY
- FILMS

@colbyfayock


BEYONCÉ
THE GIFT:
DELUXE EDITION
beyonce.com
@colbyfayock

WordPress

Dashboard
Jetpack
Posts
All Posts
Add New
Categories
Tags
Media
Pages
Comments
Products
CM Episodes
Courses
TMYC Episodes
Feedback
Appearance
Plugins 1
Users

Posts

Add New

All (23) | Published (23) | Cornerstone content (0)

Bulk actions | Apply | All dates | All Cate

Title	Author
How to Use the GitHub GraphQL API to Add Your Pinned Repositories In Next.js & React	Colby Fayock
How to Add a Dynamic Table of Contents to Static HTML in React with Rehype	Colby Fayock
How to Add Custom Dynamic Favicons in React & Next.js	Colby Fayock
How to Use Puppeteer to Automate Chrome in an API with Netlify Serverless Functions	Colby Fayock
How to Test Serverless Functions with Jest & Next.js	Colby Fayock

How to Use the GitHub GraphQL API to Add Your Pinned Repositories In Next.js & React

OSS projects can be hard work and when you're spending your time working on t
display those projects. While GitHub has a Pinned mechanism, you would need a
your own website. How can we instead use the GitHub GraphQL API to bring our

READ MORE

What is the GitHub GraphQL API?

In order to actually request data from GitHub we're going to take advantage of it'

If you're more familiar with the REST side of things, you likely could make request
GraphQL as our request mechanism gives us a lot more flexibility and options for
on complicated relationships.

The GitHub GraphQL API Explorer is a great way to see this in action. If you head
dynamically look through all of the data GitHub makes available through its API.

What are Pinned Repositories?

Pinned Repositories are GitHub's way to allow you to feature a few of your favorit

Document

Technologies > Content Management

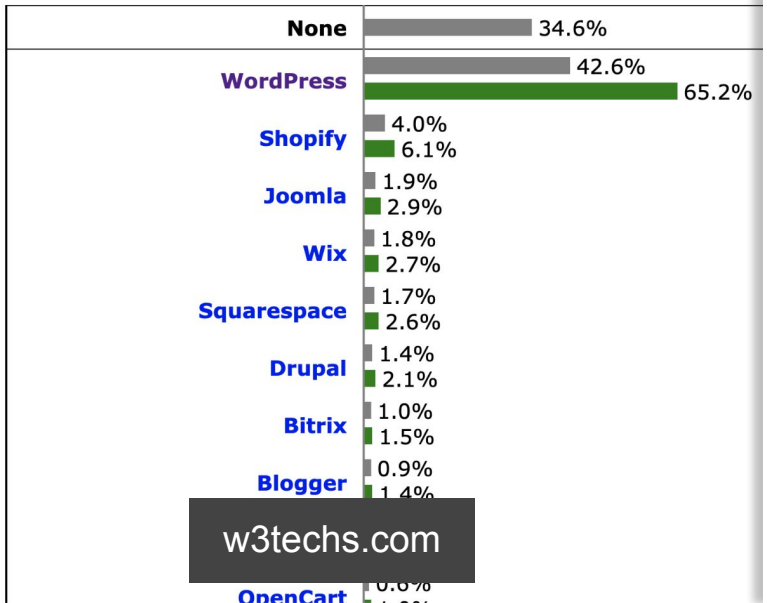
Usage statistics of content management systems

This diagram shows the percentages of websites using various content manageme
systems. See technologies overview for explanations on the methodologies used i
surveys. Our reports are updated daily.

Request an extensive content management systems market report.

Learn more

How to read the diagram:
34.6% of the websites use none of the content management systems that we mo
WordPress is used by 42.6% of all the websites, that is a content management sy
market share of 65.2%.

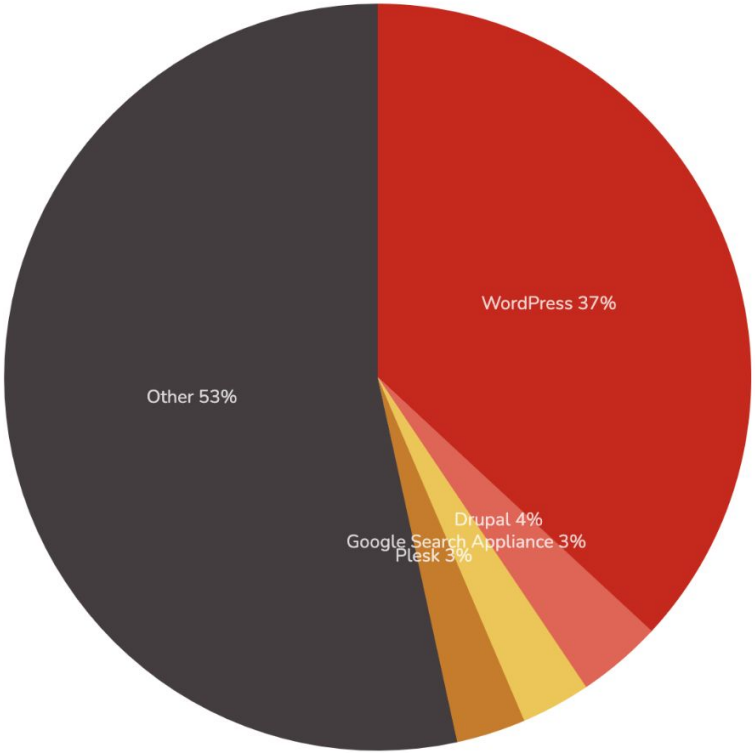


w3techs.com

Home / Trends / Content Management System

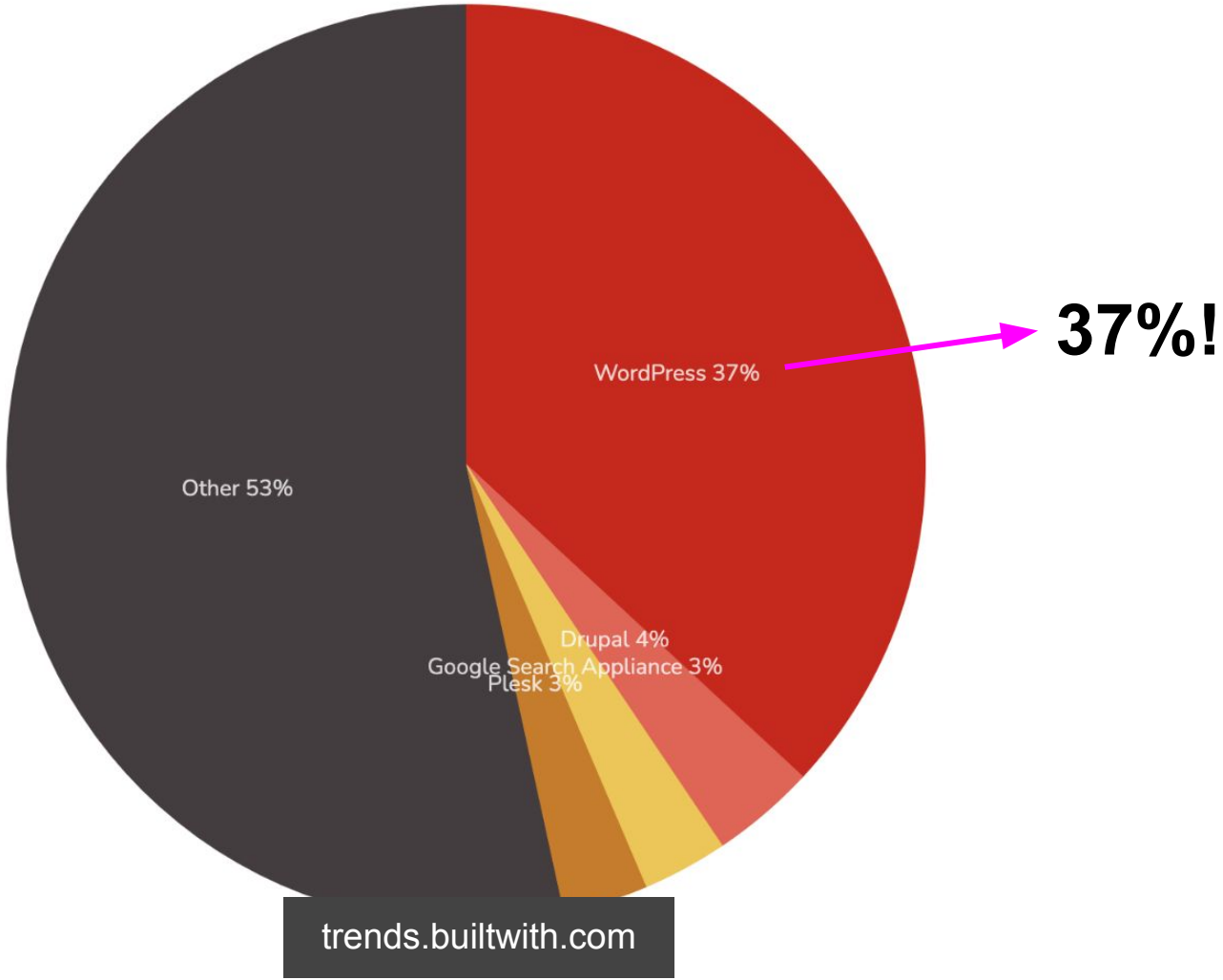
CMS Usage Distribution in the Top 1 Million Site

Distribution for websites using CMS technologies



trends.builtwith.com

Explai


WordPress 37%
Other 53%
Drupal 4%
Google Search Appliance 3%
Plesk 3%
37%!
trends.builtwith.com

WordPress

Performant

Cost Effective

Reliable

Ecosystem

UX

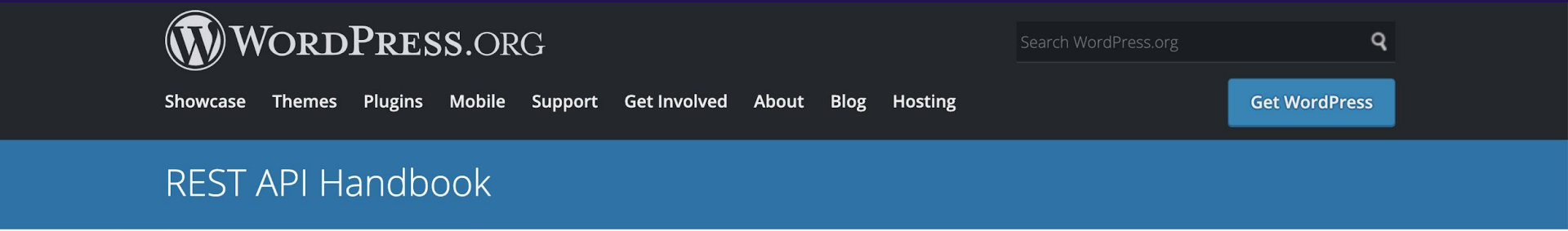
Familiar

Content Mesh

Has an API!

Headless WordPress

WordPress API



REST API Handbook

Edit

The WordPress REST API provides an interface for applications to interact with your WordPress site by sending and receiving data as JSON (JavaScript Object Notation) objects. It is the foundation of the WordPress Block Editor, and can likewise enable your theme, plugin or custom application to present new, powerful interfaces for managing and publishing your site content.

TOPICS

What Is A REST API?
Using the WordPress REST API
Next Steps

Using the WordPress REST API you can create a plugin to provide an entirely new admin experiences for WordPress, build a brand new interactive front-end experience, or bring your WordPress content into completely separate applications.

Not Secure | 54.145.245.231/wp-json


```
{
    name: "Space Jelly",
    description: "Cosmic web dev tutorials that will
    shock you with joy!",
    url: "http://54.145.245.231",
    home: "http://54.145.245.231",
    gmt_offset: "0",
    timezone_string: "",
  - namespaces: [
        "oembed/1.0",
        "jetpack/v4",
        "yoast/v1",
        "wpcom/v2",
        "wp/v2",
        "wp-site-health/v1"
    ],
    authentication: [ ],
  - routes: {
      - /: {
            namespace: "",
          - methods: [
                "GET"
            ],
          - endpoints: [
              - {
                  - methods: [
                        "GET"
                    ],
                  - args: {
```


+ - View source

localhost:3000

Space Jelly

Cosmic web dev tutorials that will shock you with joy!

@colbyfayock

Not Secure | 54.145.245.231/wp-json/wp/v2/posts

+ - View source


```
[
  - {
      id: 625,
      date: "2021-08-06T16:34:46",
      date_gmt: "2021-08-06T16:34:46",
    - guid: {
        rendered: "http://54.145.245.231/?p=625"
      },
      modified: "2021-08-06T16:34:46",
      modified_gmt: "2021-08-06T16:34:46",
      slug: "how-to-make-twitter-api-requests-with-nexta
      status: "publish",
      type: "post",
      link: "http://54.145.245.231/2021/08/06/how-to-mak
    - title: {
        rendered: "How to Make Twitter API Requests wi
      },
    - content: {
        rendered: " <p>NextAuth.js makes adding authen
        doesn&#8217;t mean we automatically have acces
        advantage of the active session to interact wi
        </h2> <p>NextAuth.js is an all-in-one solution
        <p>If you followed along with my previous tuto
        authenticate-next-js-apps-with-twitter-nextaut
        consumer-and-client-keys">How to Authenticate
        how we can use it to allow visitors to our app
        they&#8217;re who they say they are.</p> <bloc
        start off from that last tutorial, so if this
        idea to follow along there first!</p></blockqu
        </h2> <p>The cool thing about NextAuth.js is i
```


localhost:3000

Space Jelly

Cosmic web dev tutorials that will shock you with joy!

How to Make Twitter API Requests with NextAuth.js Session Tokens

NextAuth.js makes adding authentication with providers like Twitter easy, but that doesn't mean we automatically have access to use those providers' APIs. How can we take advantage of the active session to interact with APIs like Twitter's? What is NextAuth.js? NextAuth.js is an all-in-one solution for adding authentication to a Next.js application. If you followed [...]

How to Use Browser Event Listeners in React for Search and Autocomplete

@colbyfayock

WordPress + GraphQL

Docs Developer Reference Extensions Blog

Search...

GraphQL API for WordPress

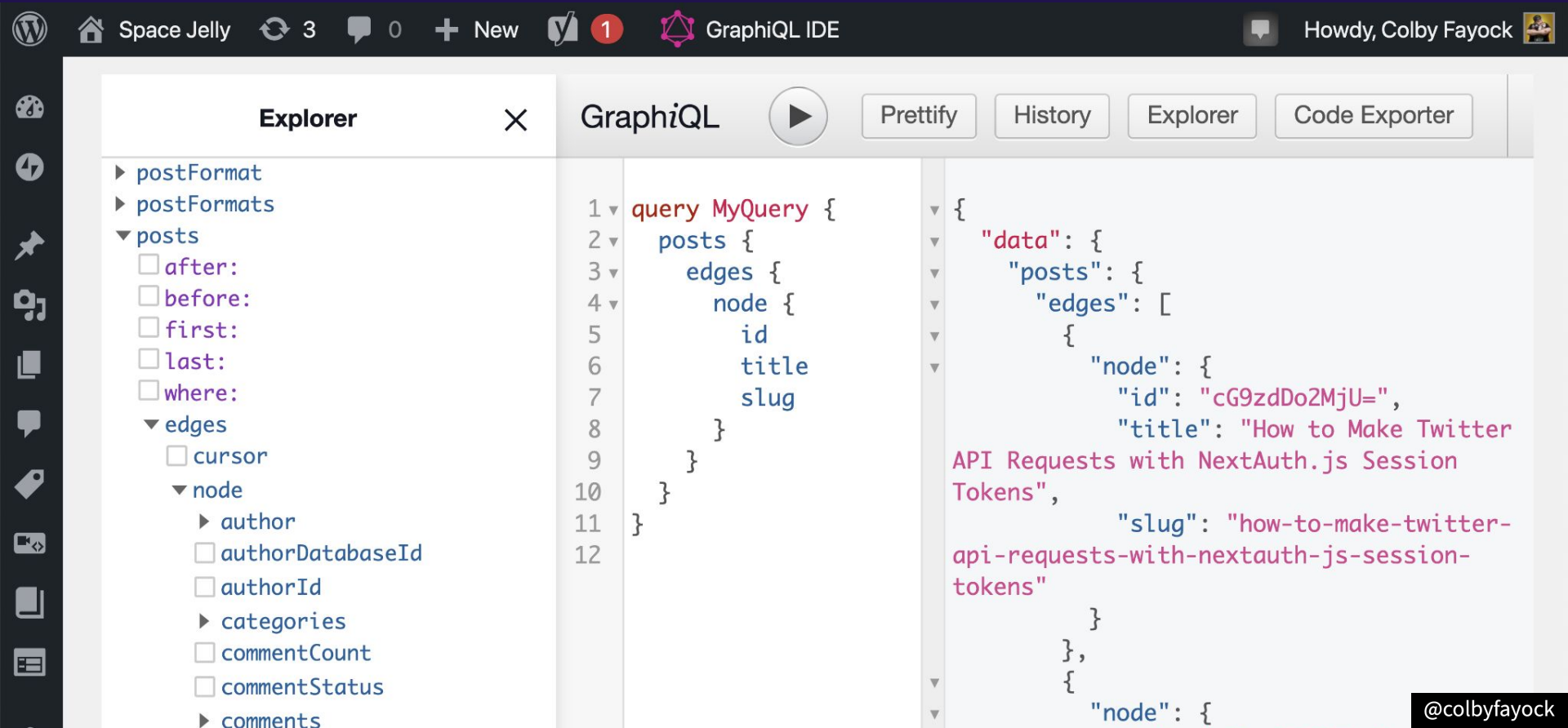
WPGraphQL is a free, open-source WordPress plugin that provides an extendable GraphQL schema and API for any WordPress site.

Get Started →

Download the Plugin

@colbyfayock

WPGraphQL

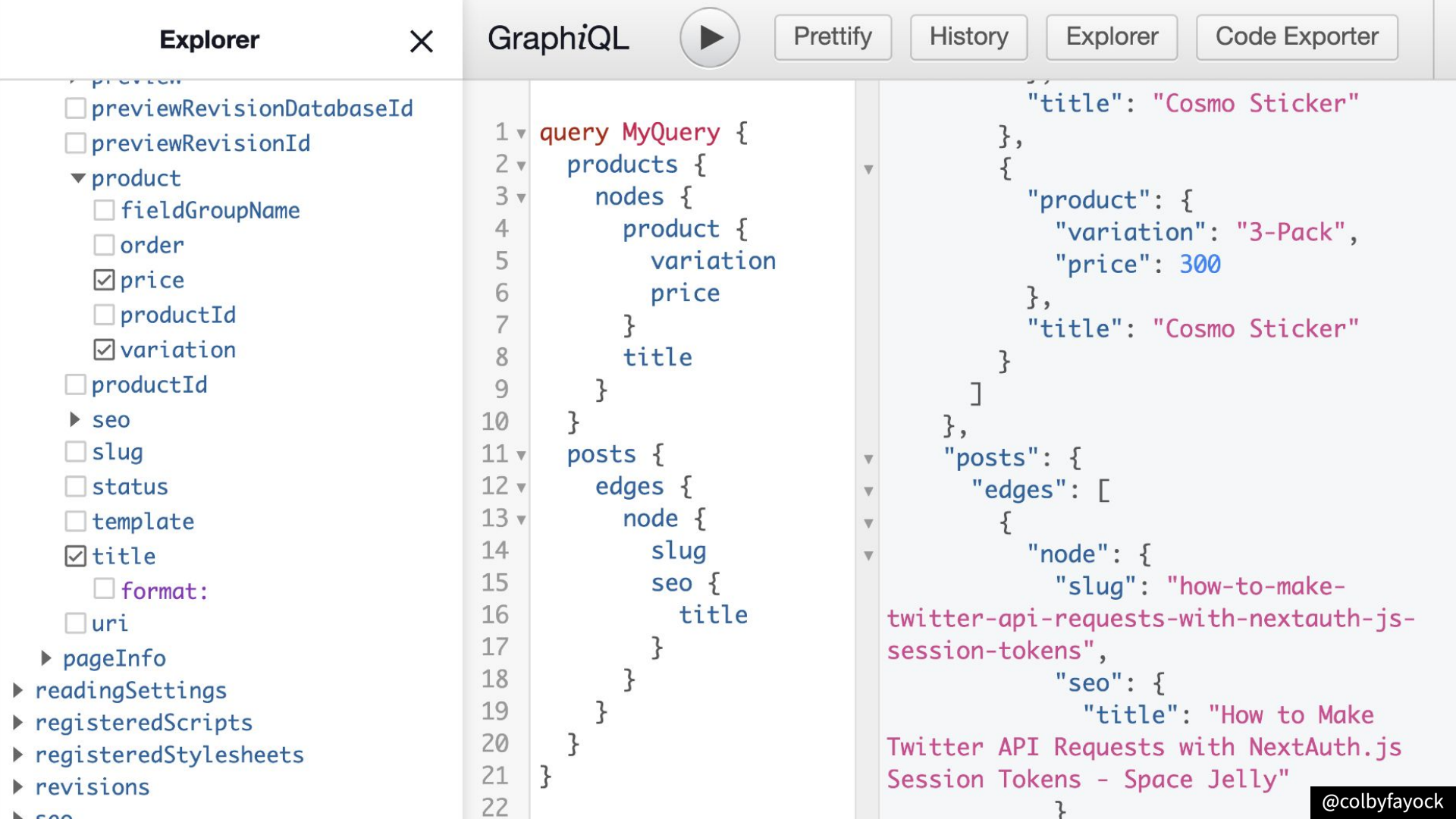

Space Jelly
3
0
New
1
GraphiQL IDE
Howdy, Colby Fayock
Explorer
GraphiQL
Prettify
History
Explorer
Code Exporter
postFormat
postFormats
posts
after:
before:
first:
last:
where:
edges
cursor
node
author
authorDatabaseId
authorId
categories
commentCount
commentStatus
comments
query MyQuery {
posts {
edges {
node {
id
title
slug
}
}
}
}
{
"data": {
"posts": {
"edges": [
{
"node": {
"id": "cG9zdDo2MjU=",
"title": "How to Make Twitter API Requests with NextAuth.js Session Tokens",
"slug": "how-to-make-twitter-api-requests-with-nextauth-js-session-tokens"
}
},
{
"node": {

```
query {
  generalSettings {
    title
    description
  }
  posts {
    edges {
      node {
        title
        slug
        excerpt
        categories {
          edges {
            node {
              name
            }
          }
        }
        author {
          node {
            name
          }
        }
```

```
{
  - data: {
      - generalSettings: {
            title: "Space Jelly",
            description: "Cosmic web dev tutorials that will shock you with joy!"
        },
      - posts: {
          - edges: [
              - {
                  - node: {
                        title: "How to Make Twitter API Requests with NextAuth.js Sessio
                        slug: "how-to-make-twitter-api-requests-with-nextauth-js-session
                        excerpt: ...,
                      - categories: {
                          - edges: [
                              - {
                                  - node: {
                                        name: "Authentication"
                                    }
                                },
                              + {…},
                              + {…},
                              + {…}
                            ]
                        },
                      - author: {
                          - node: {
                                name: "Colby Fayock"
                            }
                        }
```

@colbyfayock

WPGraphQL + Plugins


Explorer
GraphiQL
Prettify
History
Explorer
Code Exporter
previewRevisionDatabaseId
previewRevisionId
product
fieldGroupName
order
price
productId
variation
productId
seo
slug
status
template
title
format:
uri
pageInfo
readingSettings
registeredScripts
registeredStylesheets
revisions
query MyQuery {
products {
nodes {
product {
variation
price
}
title
}
}
posts {
edges {
node {
slug
seo {
title
}
}
}
}
}
"title": "Cosmo Sticker"
},
{
"product": {
"variation": "3-Pack",
"price": 300
},
"title": "Cosmo Sticker"
}
]
},
"posts": {
"edges": [
{
"node": {
"slug": "how-to-make-twitter-api-requests-with-nextauth-js-session-tokens",
"seo": {
"title": "How to Make Twitter API Requests with NextAuth.js Session Tokens - Space Jelly"
}
@colbyfayock

WPGraphQL + Plugins

WordPress at the Edge

Jamstack

Operational Excellence

Security

Reliability

Performance Efficiency

Cost Optimization

AWS Well-Architected

Performance

Plugins

My Favorites Beta Testing Developers cache

Showing results for: **cache**

LiteSpeed Cache

(1,169)

All-in-one unbeatable acceleration & PageSpeed improvement: caching, image/CSS/JS optimization...

LiteSpeed Technologies

1+ million active installations Tested with 5.5.1

W3 Total Cache

(4,573)

Search Engine (SEO) & Performance Optimization (WPO) via caching. Integrated caching: CDN, Page, Minify, Object, Fragment, Database support.

BoldGrid

1+ million active installations Tested with 5.5.1

WP Fastest Cache

(3,215)

The simplest and fastest WP Cache system

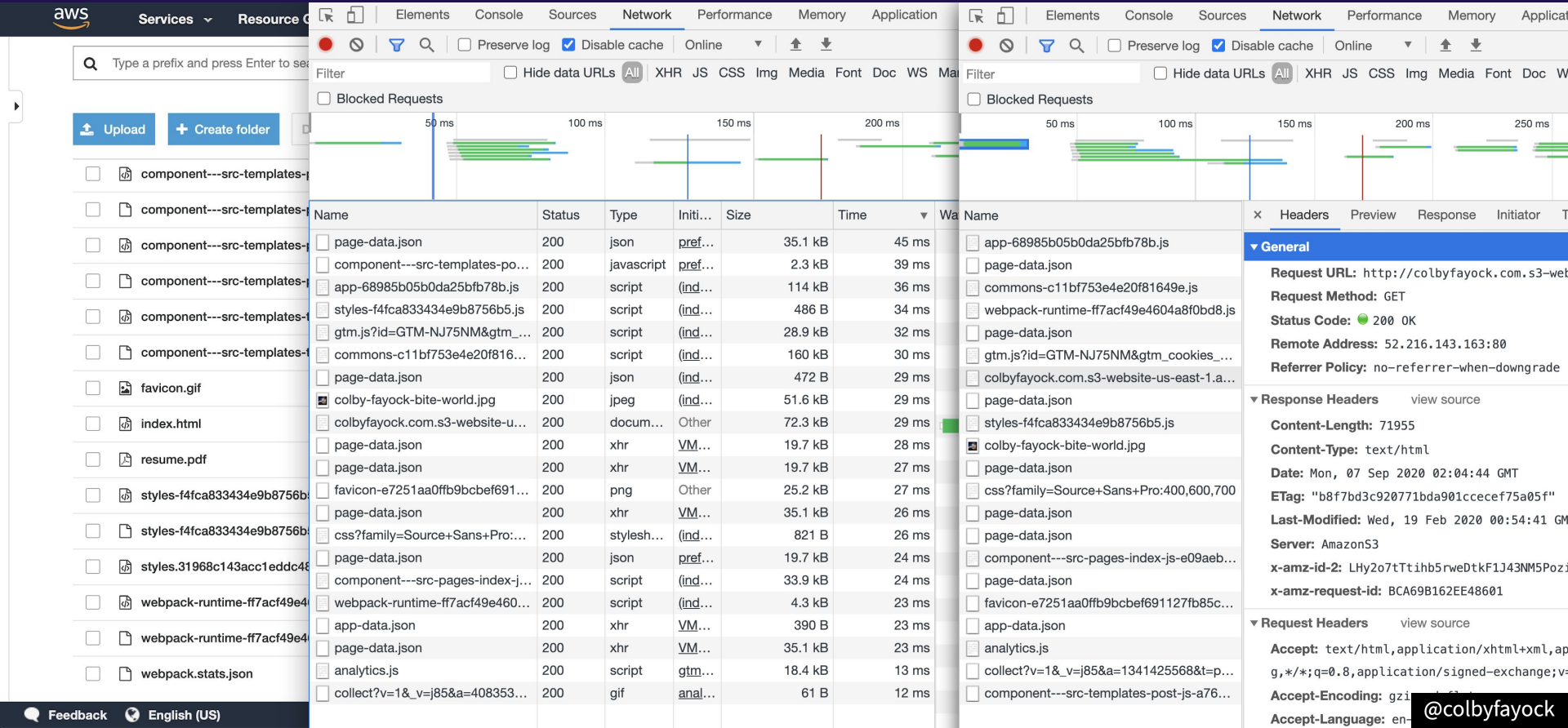
Emre Vona

WP-Optimize – Clean, Compress, Cache.

(1,187)

Makes your site fast and efficient. It cleans the database, compresses images and caches pages. Cached sites attract more traffic and users.

Performance



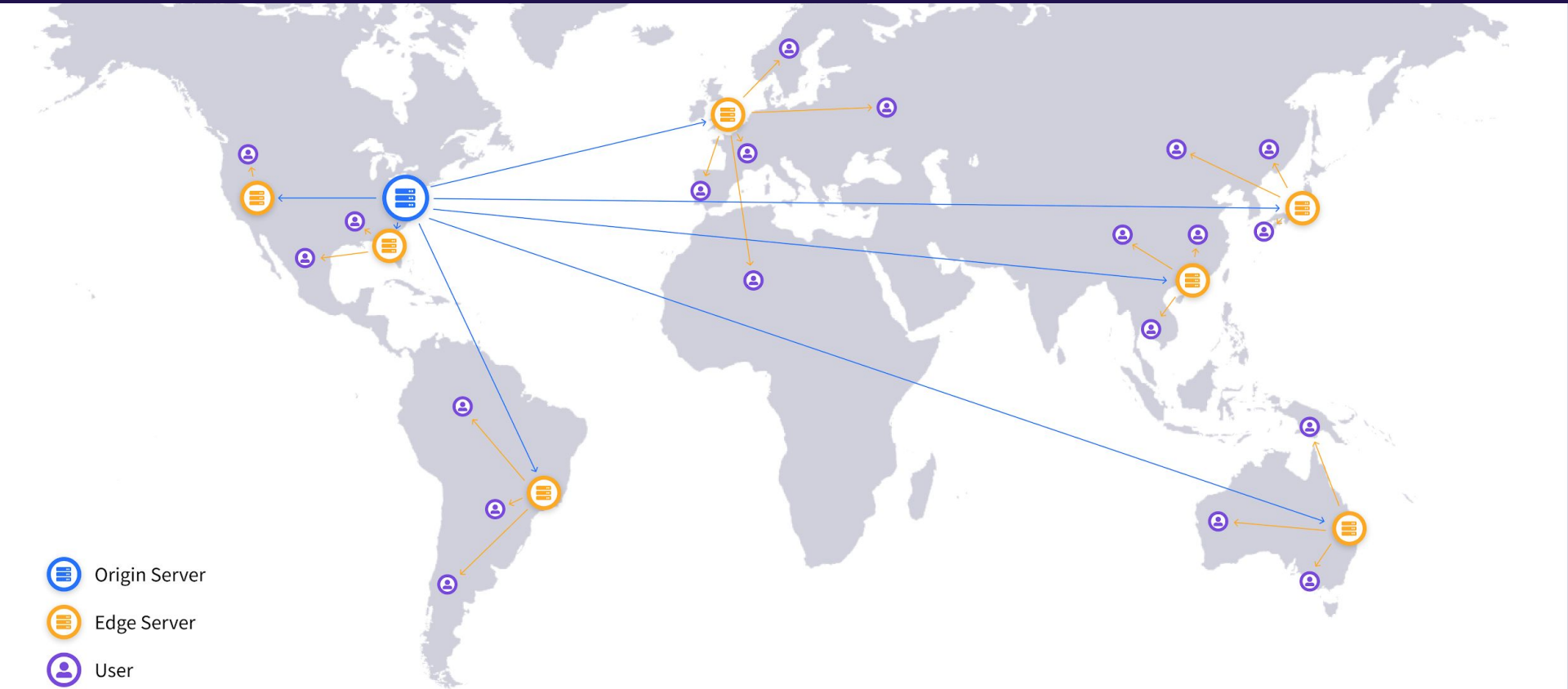
Reliability

5×× SERVER ERROR

503 SERVICE UNAVAILABLE

The server is currently unable to handle the request due to a temporary overload or scheduled maintenance, which will likely be alleviated after some delay.

Reliability



Reliability

We're having a really bad day.

The Unicorns have taken over. We're doing our best to get them under control and get GitHub back up and running.

Contact Support — GitHub Status — @githubstatus

Cost

Choose your instance plan

New! Check out our new 16 GB and 32 GB RAM bundles!

Sort by: Price per month | Memory | Processing | Storage | Transfer

First month free!					
$3.5 USD	$5 USD	$10 USD	$20 USD	$40 USD	Price per month
$3.50 USD	$5 USD	$10 USD	$20 USD	$40 USD	Price per month
512 MB	1 GB	2 GB	4 GB	8 GB	Memory
1 vCPU	1 vCPU	1 vCPU	2 vCPUs	2 vCPUs	Processing
20 GB SSD	40 GB SSD	60 GB SSD	80 GB SSD	160 GB SSD	Storage
1 TB	2 TB	3 TB	4 TB	5 TB	Transfer

@colbyfayock

Cost

FREE

then $20 / 100 GB
Measured as Bandwidth

$2.30

$0.023 / GB
Add $0.0004 / 1k GET Requests

$2.00 - $2.30

$0.020 - $0.023 / GB
(Depending on Region)
Add $0.0004 / 1k GET Requests

First 300gb / mo

Gatsby

SVELTE KIT

NEXT.JS

11ty

Gatsby

SVELTE KIT

NEXT.JS

11ty

Showcase Docs Live Analytics Commerce Enterprise Learn

The React Framework for Production

Next.js gives you the best developer experience with all the features you need for production: hybrid static & server rendering, TypeScript support, smart bundling, route pre-fetching, and more. No config needed.

Start Learning Documentation

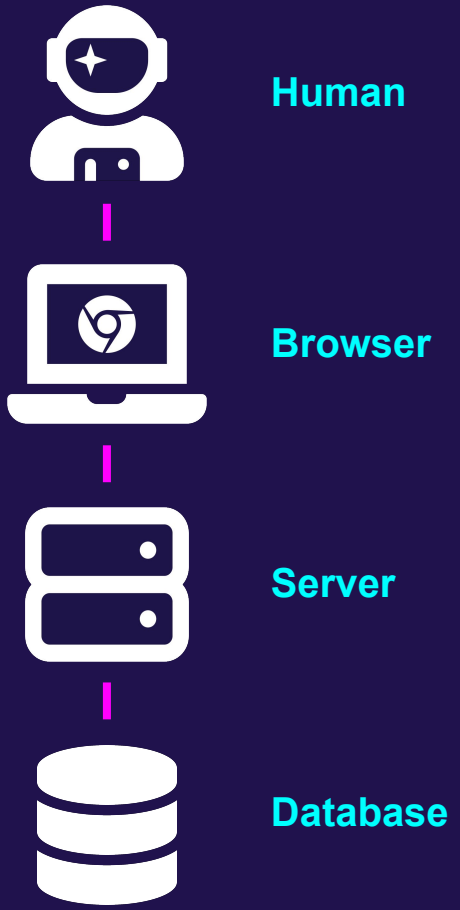
License: MIT GitHub

@colbyfayock

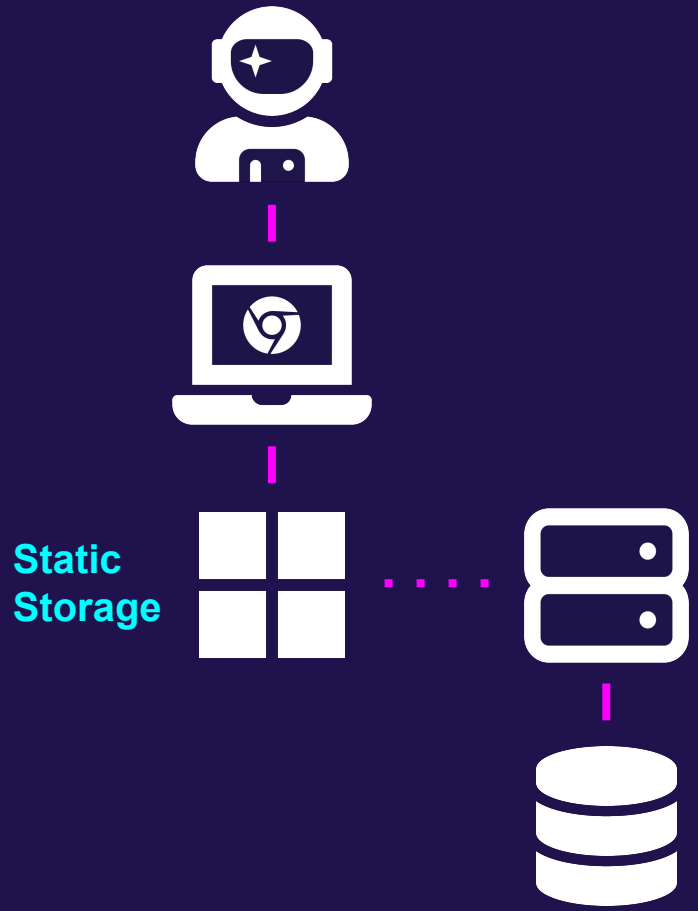
getServerSideProps

getStaticProps

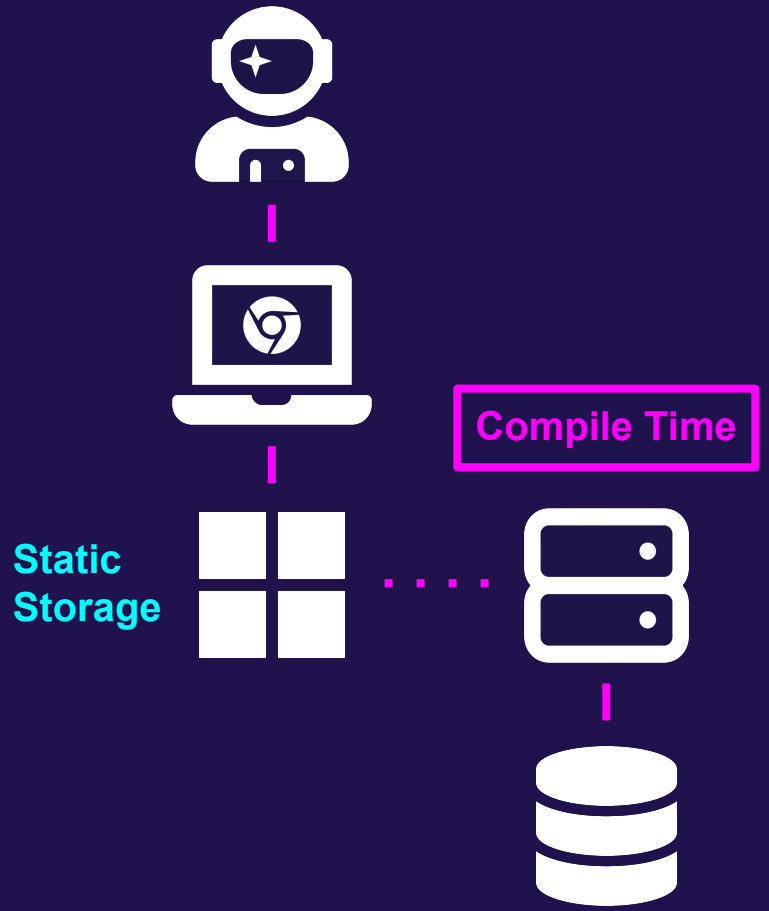
getServerSideProps



@colbyfayock

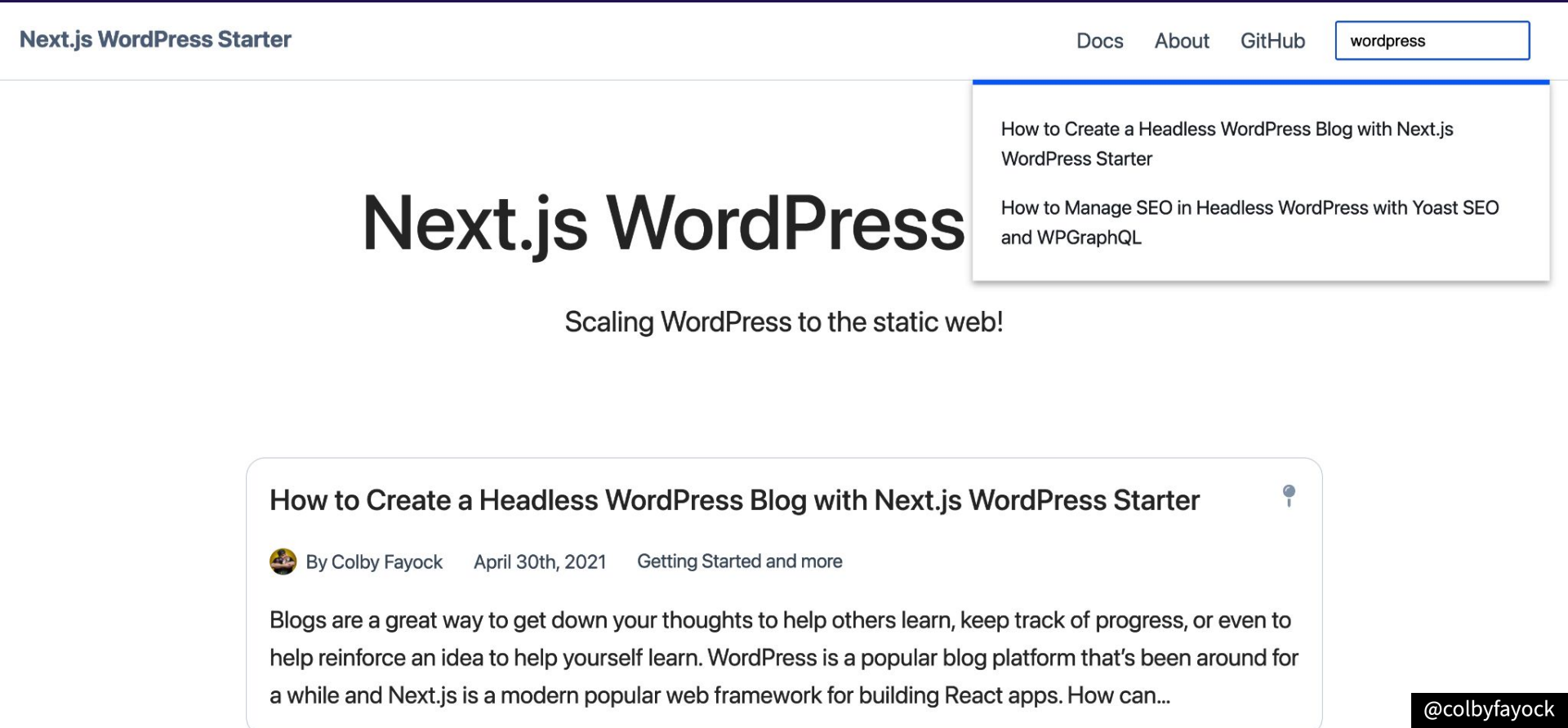


getStaticProps



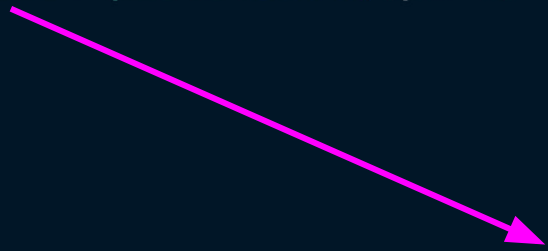
getStaticProps

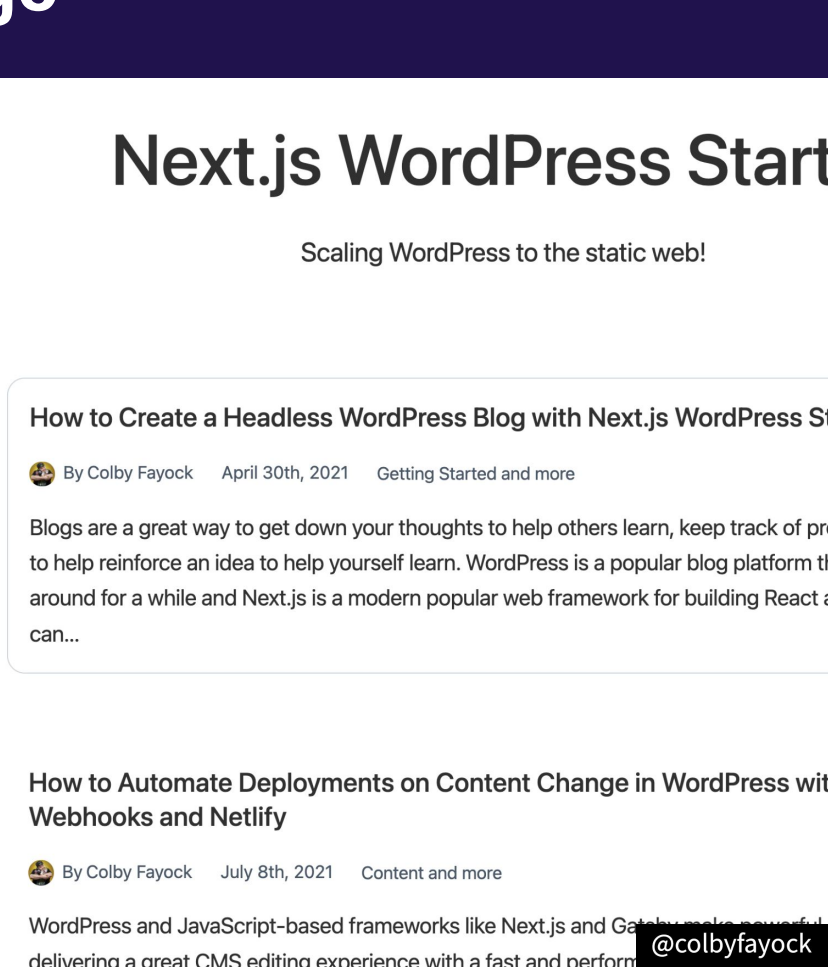
Next.js WordPress Starter



Homepage

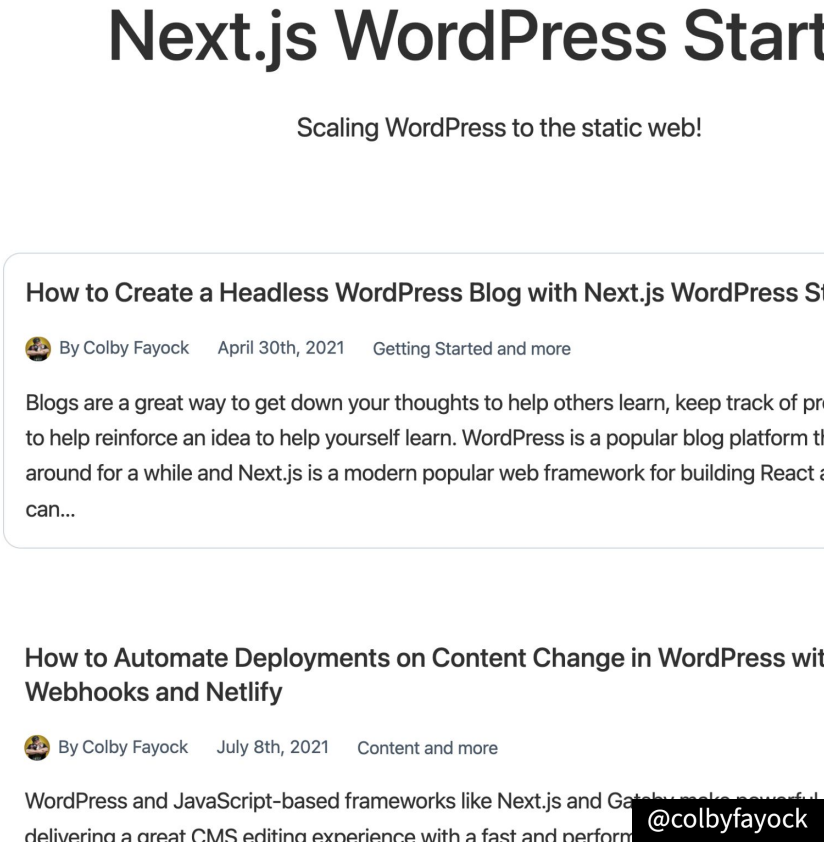
```
export async function getStaticProps() {
  const { posts } = await getAllPosts();
  return {
    props: {
      posts: posts.sort((post) => (post.i
    },
  };
}

export default function Home({ posts }) {
  const { metadata = {} } = useSite();
  const { title, description } = metadata;
```





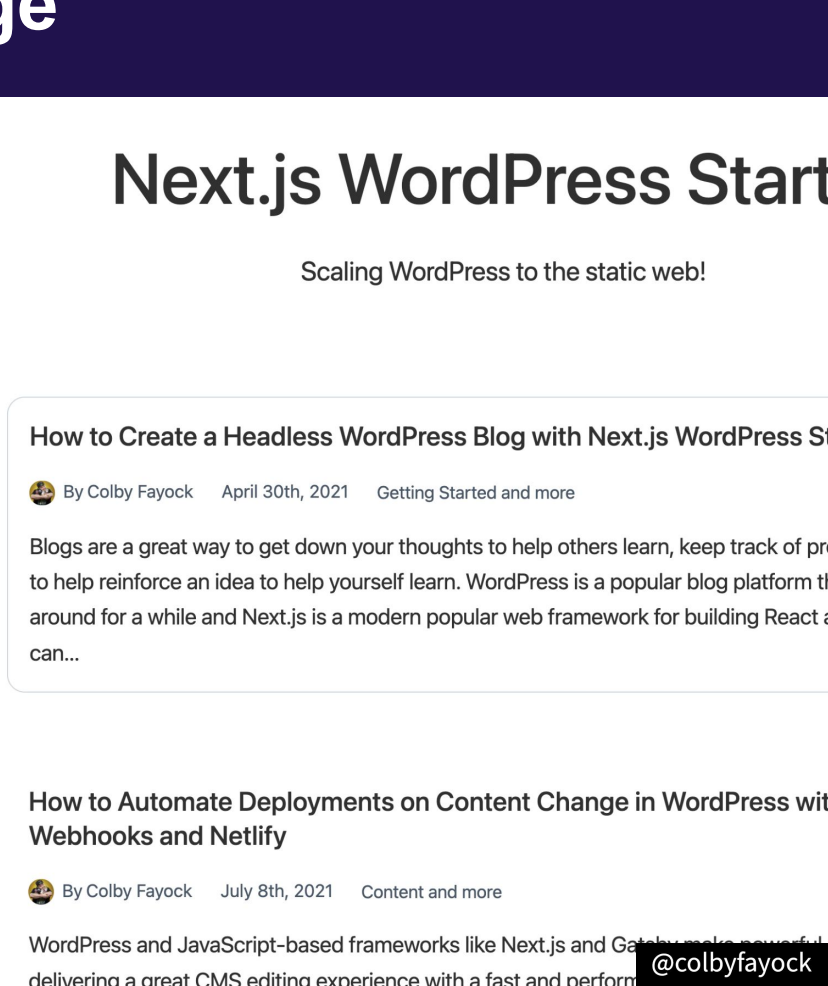
Homepage

```
export async function getStaticProps() {
  const { posts } = await getAllPosts();
  return {
    props: {
      posts: posts.sort((post) => (post.i
    },
  };
}

export default function Home({ posts }) {
  const { metadata = {} } = useSite();
  const { title, description } = metadata;
```



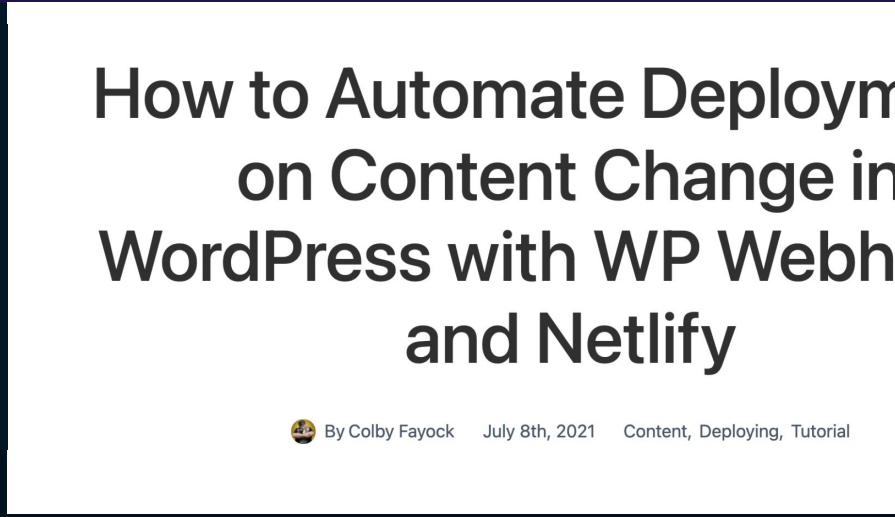
Homepage

```
export async function getStaticProps() {
  const { posts } = await getAllPosts();
  return {
    props: {
      posts: posts.sort((post) => (post.i
    },
  };
}

export default function Home({ posts }) {
  const { metadata = {} } = useSite();
  const { title, description } = metadata;
```



Post Page

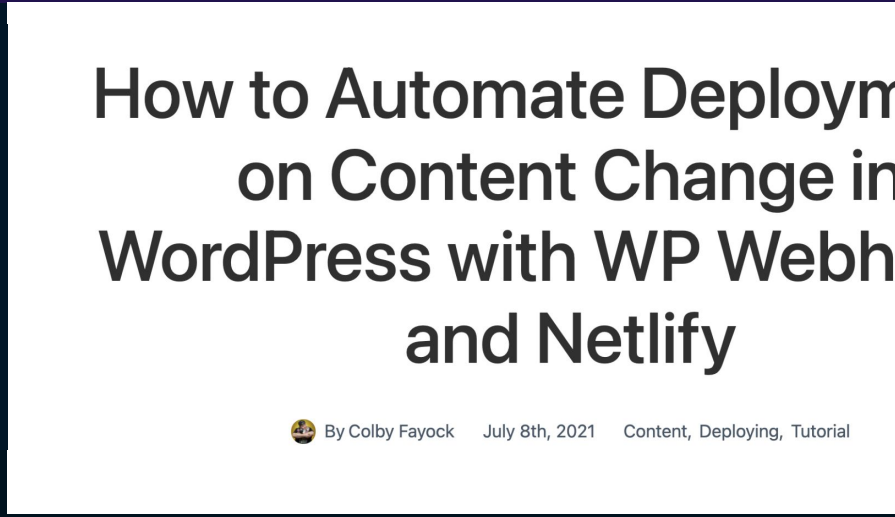
```
export async function getStaticPaths() {
  const { posts } = await getAllPosts();

  const paths = posts.map((post) => {
    const { slug } = post;
    return {
      params: { slug }
      ...
```




```
export async function getStaticProps({ params = {} }
  const { post } = await getPostBySlug(params?.slug)

  const socialImage = `/images/${params?.slug}.png`;
```

Post Page

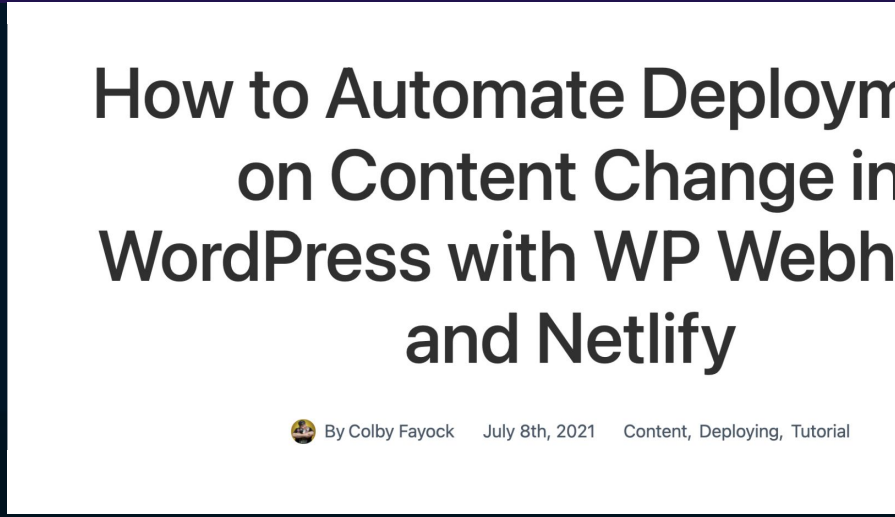
```
export async function getStaticPaths() {
  const { posts } = await getAllPosts();

  const paths = posts.map((post) => {
    const { slug } = post;
    return {
      params: { slug }
      ...
```




```
export async function getStaticProps({ params = {} }
  const { post } = await getPostBySlug(params?.slug)

  const socialImage = `/images/${params?.slug}.png`;
```

Post Page

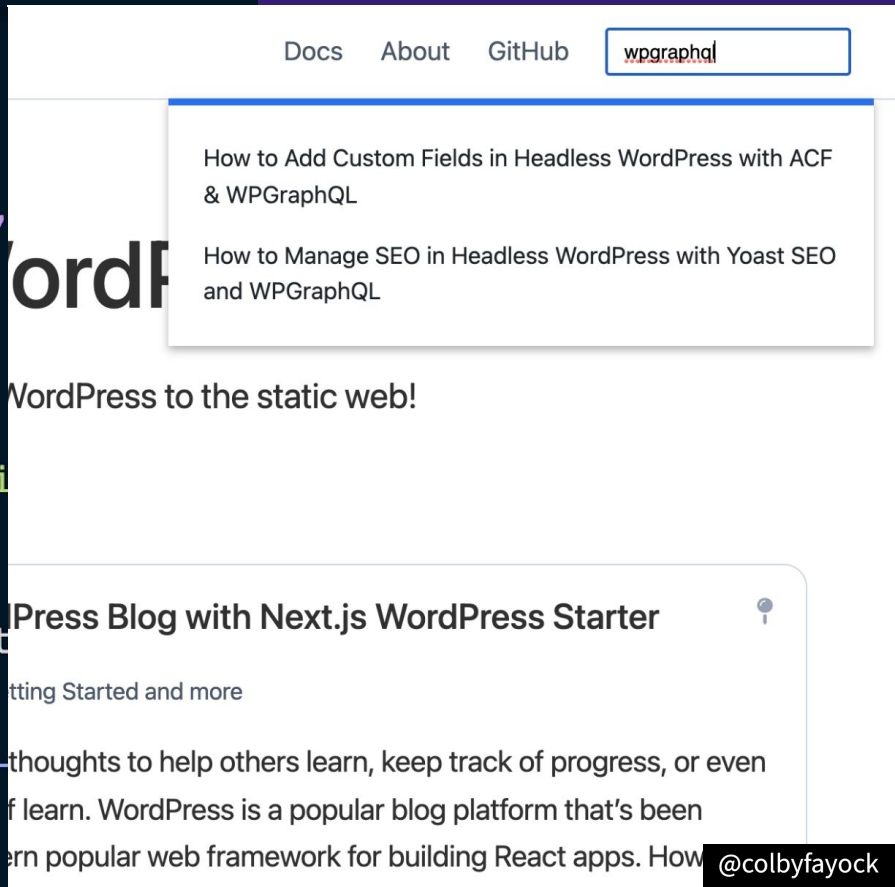
```
export async function getStaticPaths() {
  const { posts } = await getAllPosts();

  const paths = posts.map((post) => {
    const { slug } = post;
    return {
      params: { slug }
      ...
```




```
export async function getStaticProps({ params = {} }
  const { post } = await getPostBySlug(params?.slug)

  const socialImage = `/images/${params?.slug}.png`;
```

Clientside Search

```
async index(compilation, options) {
  const {
    host,
    outputDirectory = DEFAULT_OUTPUT_DIRECTORY,
    outputName = DEFAULT_OUTPUT_NAME
  } = options;

  if ( typeof host !== 'string' ) {
    throw new Error(`Failed to compile search i
  }

  const outputLocation = path.join(outputDirect

  const data = await fetch(`${host}${WORDPRESS_
  const posts = await data.json();
```



Clientside Search

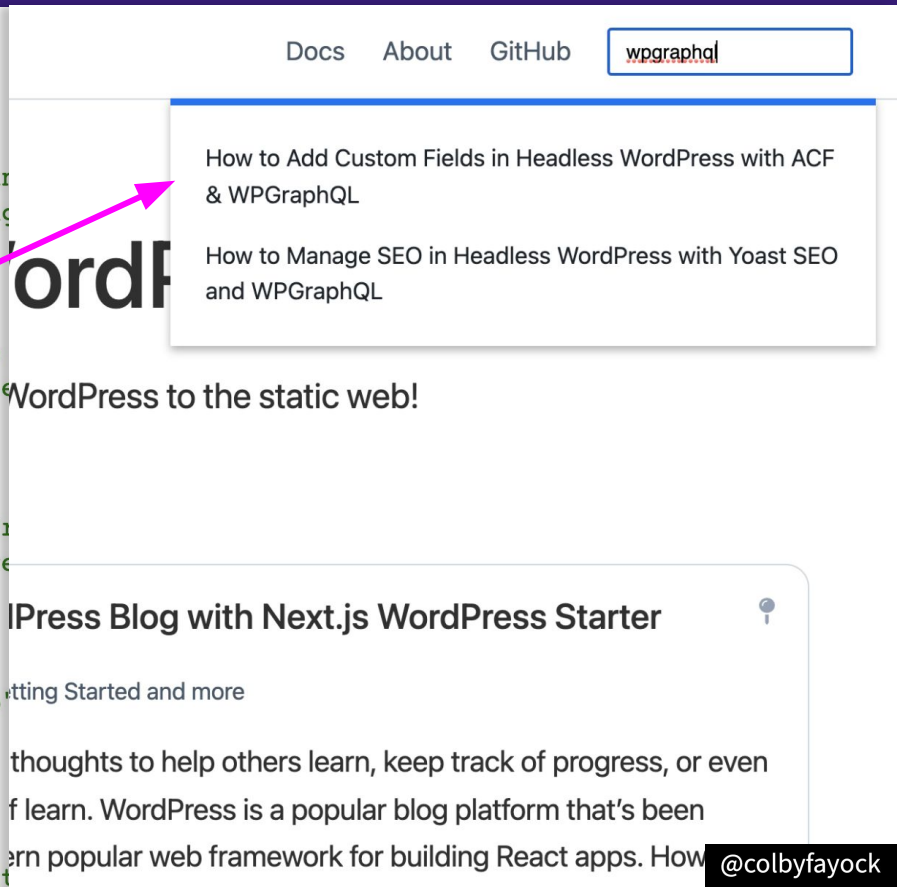
```
} = options;

if ( typeof host !== 'string' ) {
	throw new Error(`Failed to compile search i
}

const outputLocation = path.join(outputDirect

const data = await fetch(`${host}${WORDPRESS_
const posts = await data.json();

const index = posts.map((post = {}) => {
	return {
		title: post.title.rendered,
		slug: post.slug,
		date: post.date
```

```
{
   generated: 1631713626180,
 - posts: [
     - {
         title: "How to Automate Deployments on Con
         slug: "how-to-automate-deployments-on-cont
         date: "2021-07-08T15:51:30"
       },
     - {
         title: "How to improve your development ex
         slug: "how-to-improve-development-experien
         date: "2021-06-29T20:00:57"
       },
     - {
         title: "How to Add Custom Fields in Headle
         slug: "how-to-add-custom-fields-in-headles
         date: "2021-06-25T12:39:39"
       },
     - {
         title: "Exposing WordPress Data Without th
         slug: "exposing-the-wp-data-without-the-ma
         date: "2021-06-14T19:26:20"
       },
     - {
         title: "How to Manage SEO in He
```

@colbyfayock

Clientside Search


```
{
   generated: 1631713626180,
 - posts: [
     - {
         title: "How to Automate Deployments on Content Char
         slug: "how-to-automate-deployments-on-content-chang
         date: "2021-07-08T15:51:30"
       },
     - {
         title: "How to improve your development experience
         slug: "how-to-improve-development-experience-with-
         date: "2021-06-29T20:00:57"
       },
     - {
         title: "How to Add Custom Fields in Headless WordPr
         slug: "how-to-add-custom-fields-in-headless-wordpre
         date: "2021-06-25T12:39:39"
       },
     - {
         title: "Exposing WordPress Data Without the Markup"
         slug: "exposing-the-wp-data-without-the-markup",
         date: "2021-06-14T19:26:20"
       },
     - {
         title: "How to Manage SEO in Headless WordPress wit
```

@colbyfayock

Azure

Cloudflare
Pages

aws

netlify

render

Vercel

Azure

Cloudflare
Pages

aws

netlify

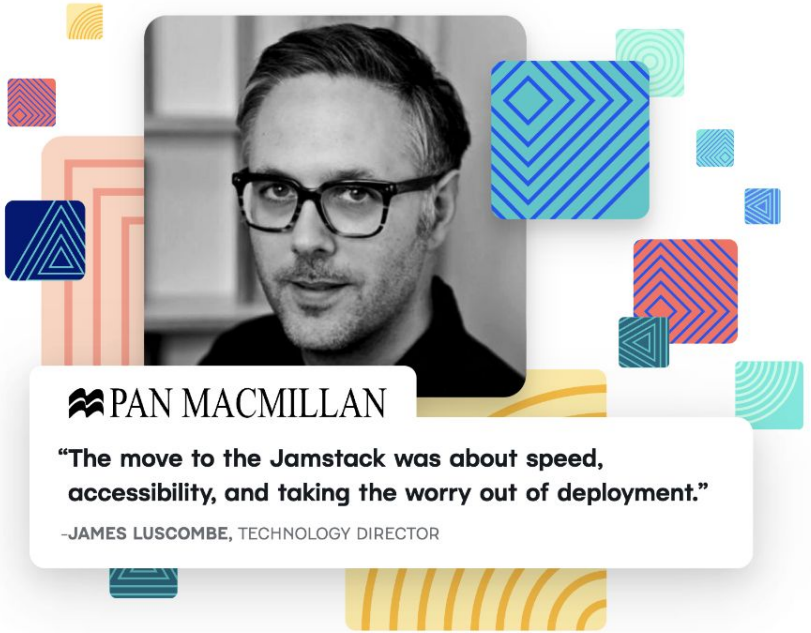
render

Vercel

Build, deploy, & scale modern web projects

Millions of developers use Netlify to ship faster, simplify their workflows, and scale effortlessly.

Get started for free Contact sales →



FIRST-CLASS SUPPORT FOR EVERY POPULAR FRAMEWORK

@colbyfayock

By Phil Hawksworth in Opinions & Insights • February 23, 2021

Terminology explained: Atomic and immutable deploys

There are terms and expressions used when talking about modern web development practices which can be confusing until explained.

Two such terms are *immutable deploys* and *atomic deploys*. Let's define each of those and call out why they are important and beneficial to an efficient and reliable development workflow.

By Phil Hawksworth in Opinions & Insights • February 23, 2021

Terminology explained: Atomic and immutable deploys

There are terms and expressions used when talking about modern web development practices which can be confusing until explained.

Two such terms are *immutable deploys* and *atomic deploys*. Let's define each of those and call out why they are important and beneficial to an efficient and reliable development workflow.

jamstack / jamstack.org Public

Code Issues 36 Pull requests 17 Discussions

Distributed Persistent Rendering (DPR) #5

cassidoo started this conversation in **RFCs**

cassidoo on Apr 14 Maintainer

Table of contents:

- Challenge
- Goal
- Proposal
- Usage scenarios
- Points of difference
- Exclusions and omissions

Challenge

The model of decoupling the generation of a site's assets from the time they are requ
but also introduces challenges when sites become very large.

As the Jamstack ecosystem has matured, tools and services have emerged bringing
productive workflows, but the practical ceiling for Jamstack sites often involves maint

By Matt Biilmann in News & Announcements • April 14, 2021

Distributed Persistent Rendering: A new Jamstack approach for faster builds

We'd like to share a new concept we've been exploring for the Jamstack, born out of our work with large enterprise dev teams deploying hundreds of thousands of pages at a time.

We wanted a way to make those deployments faster—without introducing unwanted complexity for Jamstack developers. Some solutions have come up before now, but they tend to lock you into a specific framework or break some of the most compelling reasons to use Jamstack in the first place. A Jamstack-focused approach was needed that would work across the ecosystem. After discussion both within Netlify and some members of the Jamstack community, we feel we've landed the right concept—something we've dubbed "Distributed

README.md

Essential Next.js Build Plugin

npm v3.9.0 license MIT

This build plugin is a utility for supporting Next.js on Netlify. To enable serv
framework-specific features in your Next.js application on Netlify, you will
app.

Table of Contents

- Installation and Configuration
- Docs
- Credits

Installation and Configuration

Basic build settings

If you're using a static site generator or build tool, we'll need these setting

Learn more in the docs ↗

Seems like this is a Next.js site. To enable key features of Next.js on Netlify, v
install the Essential Next.js Build Plugin. Learn more in the docs.

Base directory

Build command

yarn build

Publish directory

out

Show advanced

Deploy site

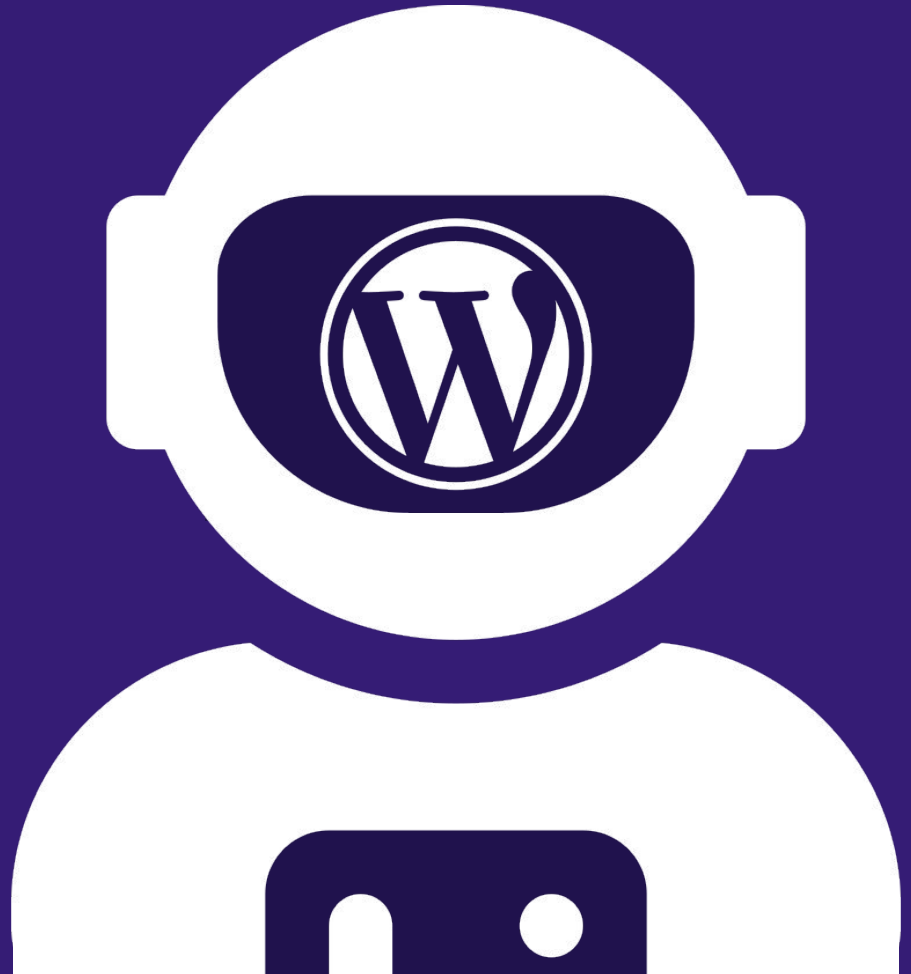
Demo

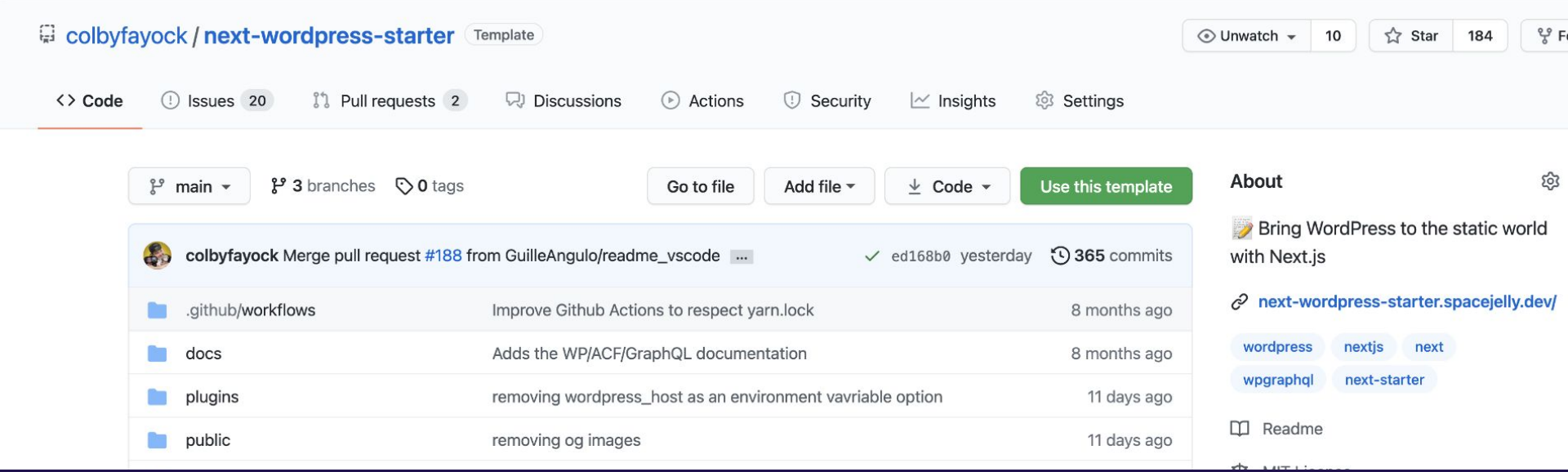
Demo

Install WPGraphQL
Query WordPress data
Bootstrap new Next.js app
Deploy to Netlify

Demo

- ✅ Install WPGraphQL
- ✅ Query WordPress data
- ✅ Bootstrap new Next.js app
- ✅ Deploy to Netlify





Ecommerce on the Jamstack

NEXT.JS

spacejelly.dev/levelupecommerce

Colby Fayock

Developer Advocate

applitools

@colbyfayock

Thank You

colbyfayock.com/jamstackconf