



# Inside the Jamstack

The technology that gives us more reliable, fast, and secure sites.



# Tara Z. Manicsic

Netlify Sr. DevEx Engineer | GDE | Nerdy Mum



# What is the Jamstack architecture?

- Generating cacheable, static assets
- Deploying to a CDN (Content Delivery Network)
- Using client-side JS to utilize third-party APIs & server less functions
- Creating a better development experience through automated workflow



# What is the Jamstack architecture?

- Delivering content faster by serving pre-rendered HTML & CSS
- Putting our assets on edge nodes closest to the user requesting to load fast
- Dynamically grabbing data from APIs or by triggering serverless functions
- Decoupling and modularizing the development process & using git workflows for automation

one more time

# What is the Jamstack architecture?

- Pre-rendering & Static Site Generators
- Content Delivery Networks (CDNs)
- Dynamic data from APIs & Serverless Functions
- Git Workflow

# **What is the Jamstack?**



# WTF is the Jamstack? A goofy name for a great web architecture.

At a high level, the Jamstack is an architectural approach to web apps that focuses on:

- generating cacheable, static assets at build time whenever possible,
- deploying those assets to CDNs, and
- using client-side JavaScript to call third-party APIs and serverless functions for dynamic interactions and data.

For individual developers, the Jamstack lowers the barrier to entry, cuts down on the number of tools to learn, and provides modern tooling to build web apps.

For teams, the Jamstack helps enforce a more maintainable architecture and decreases iteration time, enabling faster delivery along with increased security and scalability.

DEV

Search...

Write a post



# jamuary

Follow

Posts

Feed

Week

Month

Year

Infinity

Latest



Divya

Jan 30

## Do I have to rebuild my entire JAMstack site for every new change?

#jamstack #jamuary

11 reactions

7 comments

2 min read

Save



Phil Hawksworth for Netlify

Feb 13

## The final full week of JAMuary for 2020

#jamstack #jamuary #netlify

9 reactions

6 min read

Save



Divya

Jan 28

## Can I create a rich content experience on the JAMstack?





## JAMstack Course - Build websites that are simpler, faster, and more secure

50,056 views • Mar 10, 2020

1.4K 17 SHARE SAVE ...



freeCodeCamp.org   
2.17M subscribers

JOIN

SUBSCRIBE



# Generating cacheable, static assets

# Pre-rendering & Static Site Generators (SSGs)

# Static Site Generators

Generates all the pages of a site once, when there are changes to the site. There are no moving parts in the deployed site, just generated content. Making it easier to cache, harder to hack, and get to the user faster.

**Every major framework and language has one.**

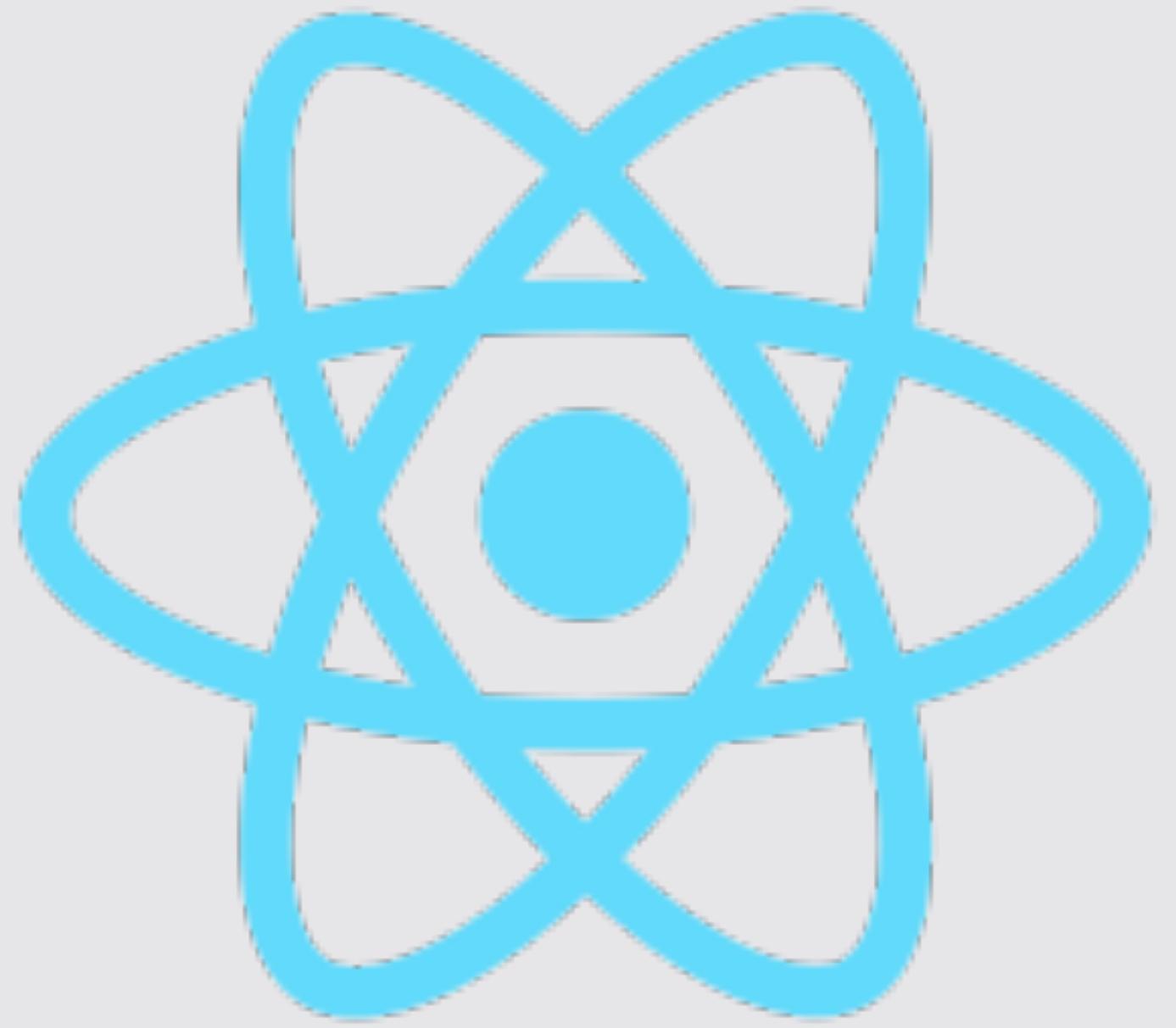
# Angular



Angular Universal

Scully

# React



@tzmanics

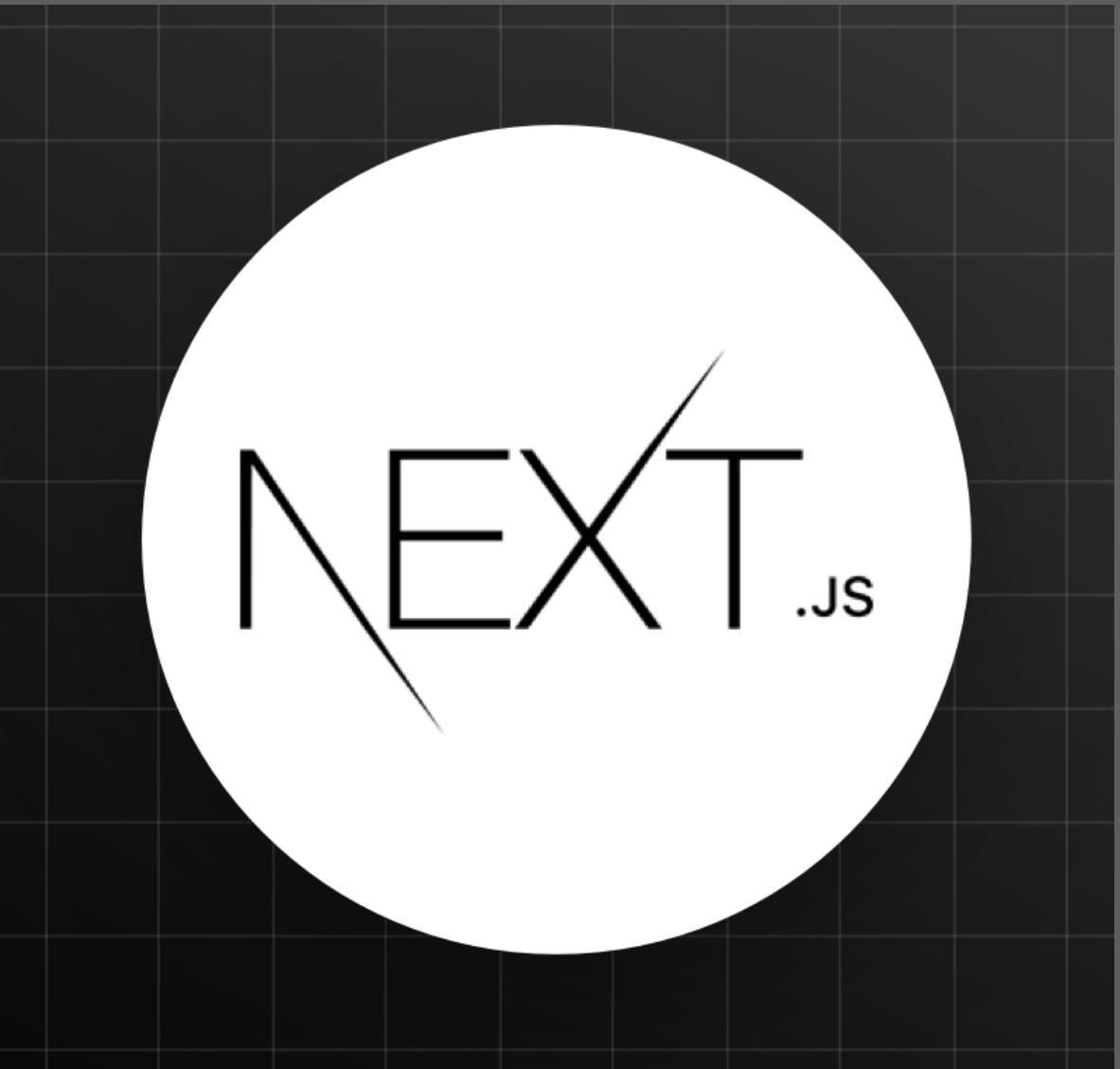
Netlify DevEx team

July, 2020

Gatsby



Next.js



netlify

# Vue



@tzmanics

Netlify DevEx team

July, 2020



Gridsome

 netlify

# Framework-free



*jekyll*  
A test tube containing red liquid, positioned next to the word "jekyll".

(Ruby)



Slate (Ruby)



SLATE

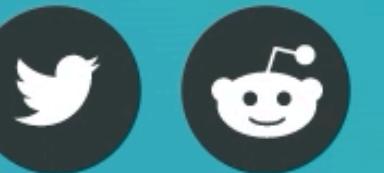
Hugo (GO)



Eleventy (JS)

# StaticGen

A List of Static Site Generators for Jamstack Sites



[About](#)   [Contribute](#)   [About Jamstack](#)   [Need a Static CMS?](#)

## Filter

Any Language



Any Template



Any License



## Sort

Stars (total)



## Next.js



**50538**

+2547



**621**

+109



**7821**

+669



N/A

## Gatsby



**45638**

+1419



**733**

-82



**8422**

+430



**57474**

+3945

## Hugo



**45329**

+1455



**534**

+18



**5155**

+151



**10568**

+524

A framework for statically-exported React apps (supports server side rendering)

Build blazing fast, modern apps and websites with React

A Fast and Flexible Static Site Generator.

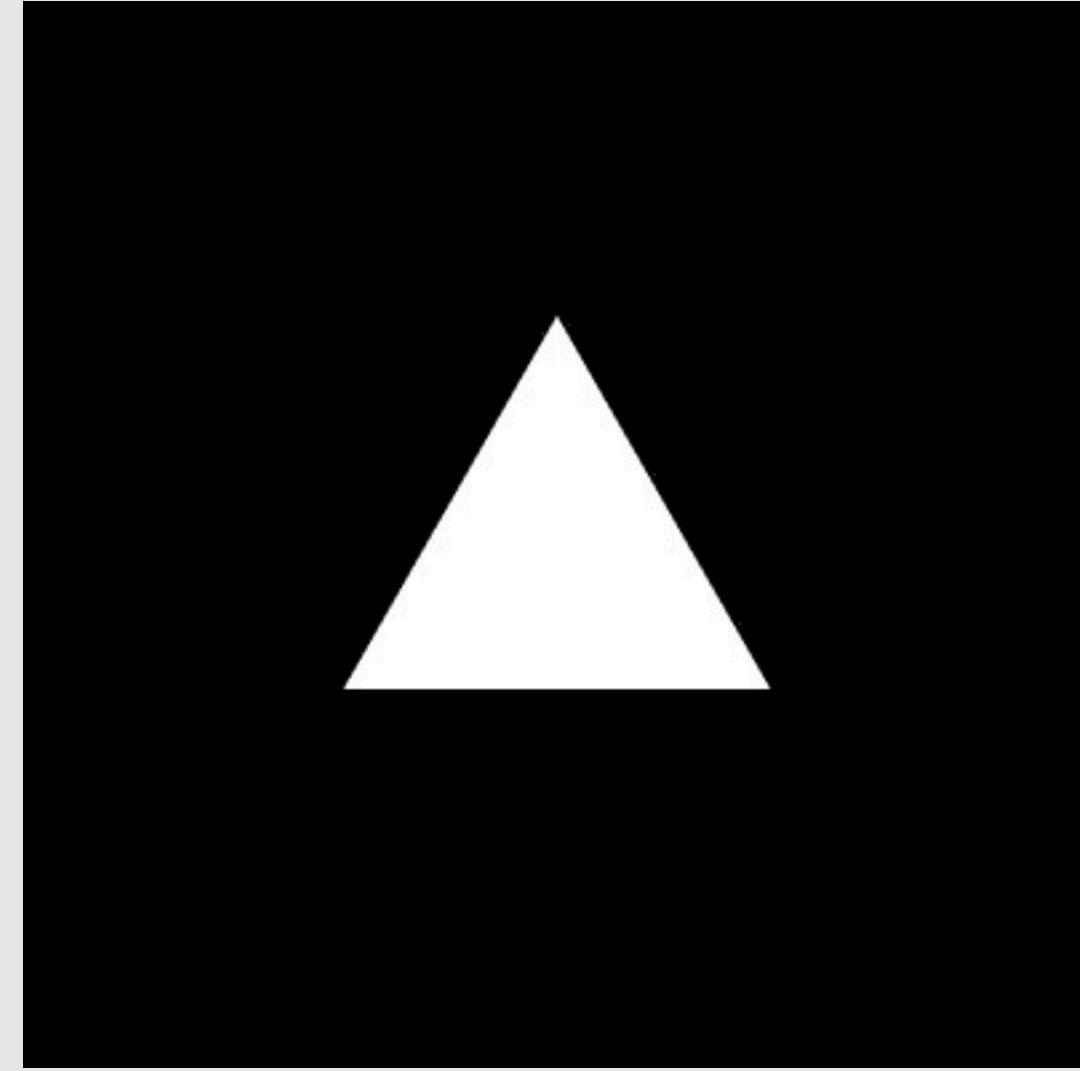


# Deploying to a Content Delivery Network (CDN)

# Content Delivery Networks

Geographically distributed edge nodes that hold the deliverable assets sent from the origin server and serve as close to the user as possible. Redirect knowledge allows them to send the deliverable to nearby nodes if there is problem, making them more reliable.

**Basically, a dumb server with redirect logic.**



@tzmanics

Netlify DevEx team

July, 2020

netlify



Platform ▾ Pricing Enterprise ▾ Jamstack ▾ Docs Blog



Contact sales

Log in

Sign up →

# Netlify Edge

A new type of application delivery network custom designed for blazing fast sites and modern workflows.





# What more can a CDN do on the JAMstack?



3

#jamstack #jamuary



Divya

Jan 25 · 2 min read

1



5

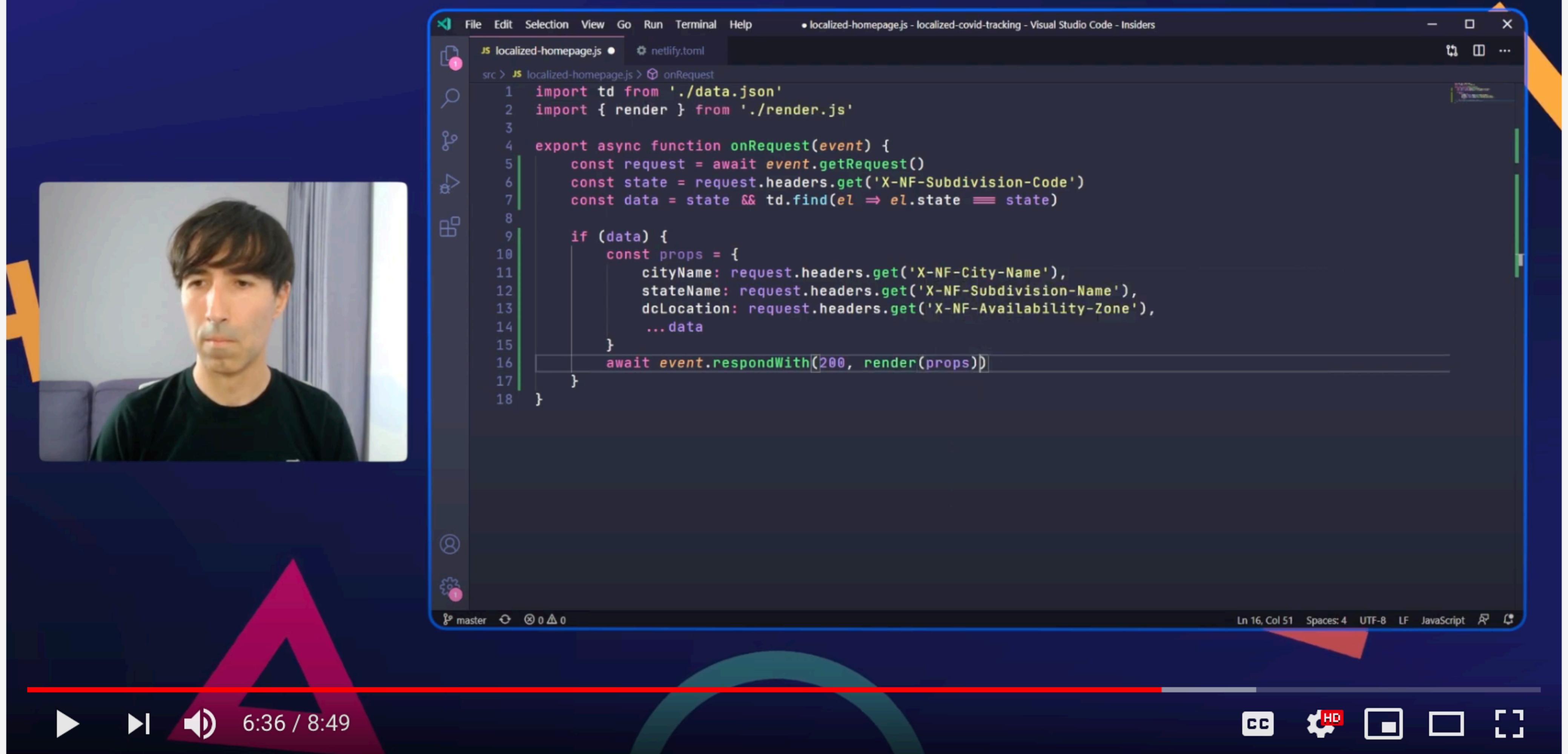
...

Compared to their predecessors, modern CDNs are not limited in their capacity and can do a lot more than serve static assets. The emergence of edge computing bolstered modern CDNs with the capability to handle crucial business logic like redirects and authentication at the edge. By forgoing the extra trip to the origin server, edge logic at the CDN layer brings the server ever closer to the user. With the increasingly blurred distinction between what an edge server can do and what an origin server can do, CDNs are increasingly being referred to as Application Delivery Networks (ADNs). With an ADN, logic once reserved for origin servers can now be implemented at the edge. Cache invalidation is one such example of this. In the context of an ADN, a new site build invalidates the old cache and replaces it with the updated content. Given the difficulty of managing the cache of a globally distributed network of edge servers, a CDN/ADN that updates its own cache quickly and reliably with every site build is incredibly convenient.

This is doubly the case since CDNs of the past used to rate limit the cache purge and

```
export async function onRequest(ev) {  
    await ev.respondWith(200, "Hello world", {})  
}
```

/fetch /hello 200! EdgeHandler=fetch-hello



## Netlify Edge Handlers Preview with David Calavera, Netlify CTO

# Dynamically grabbing data from APIs or by triggering serverless functions

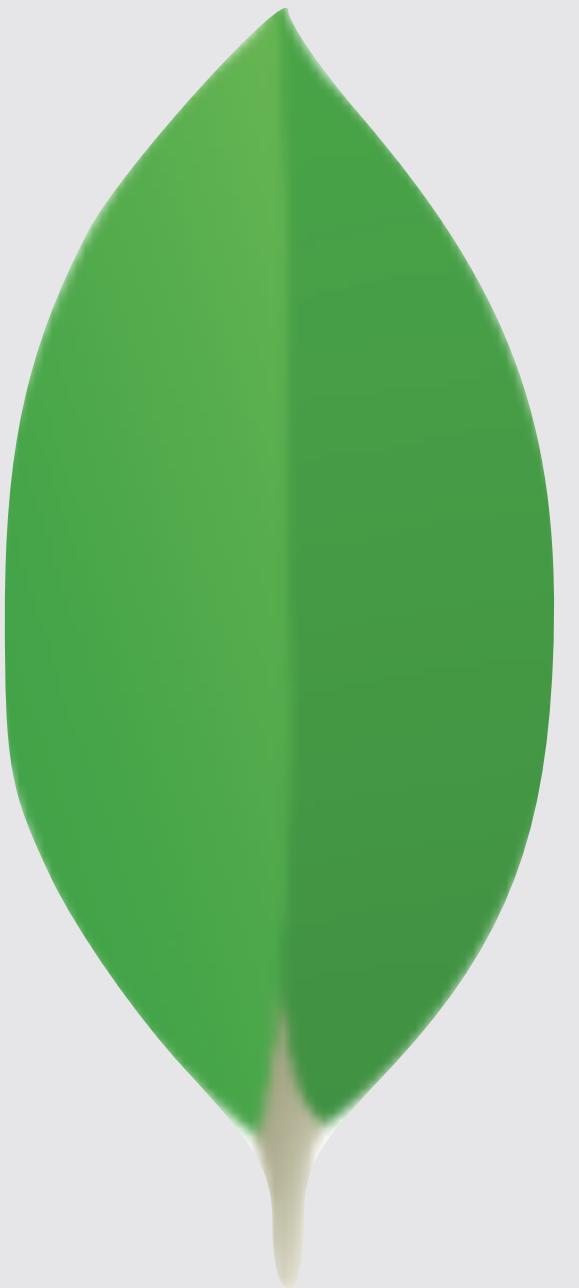
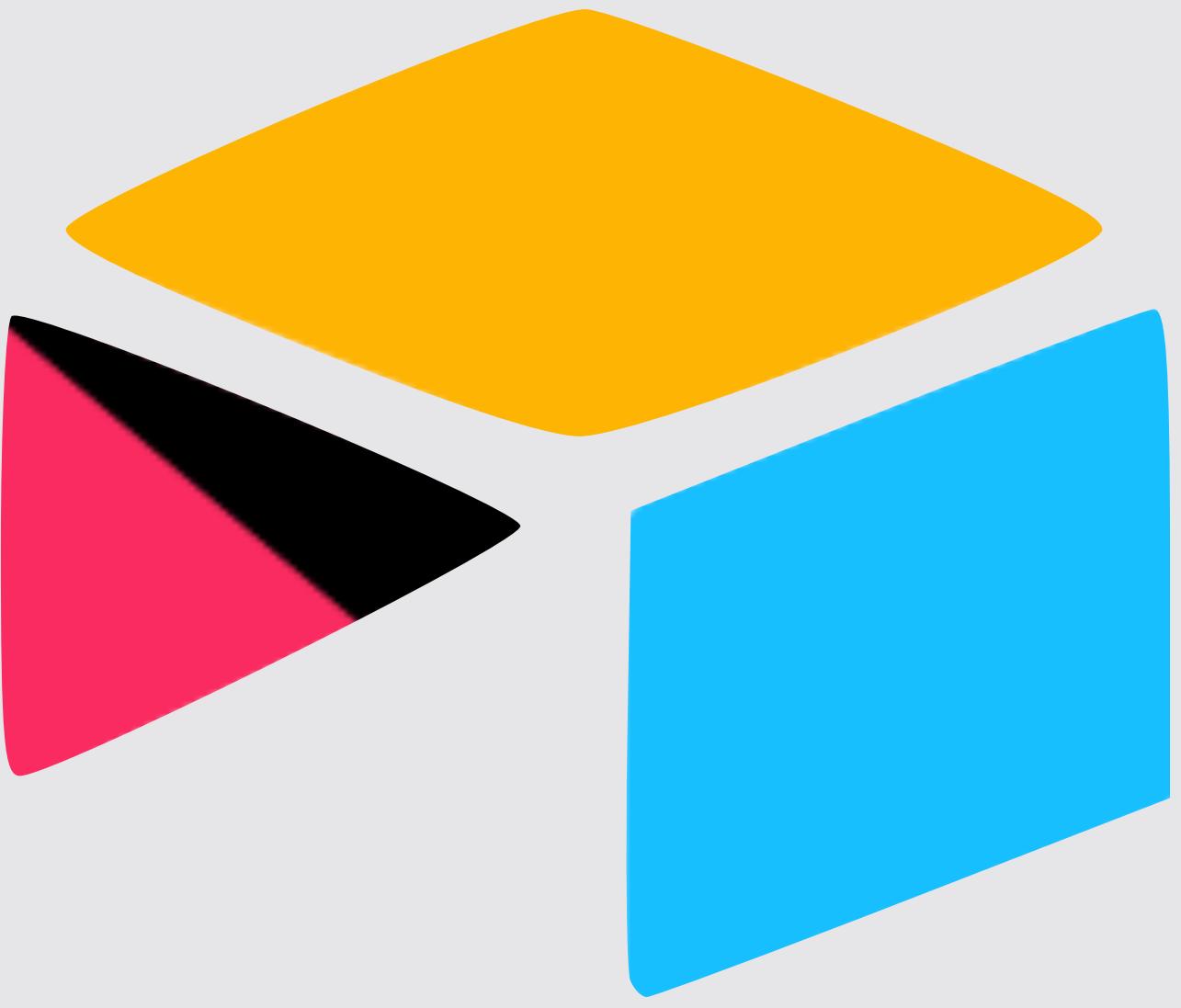
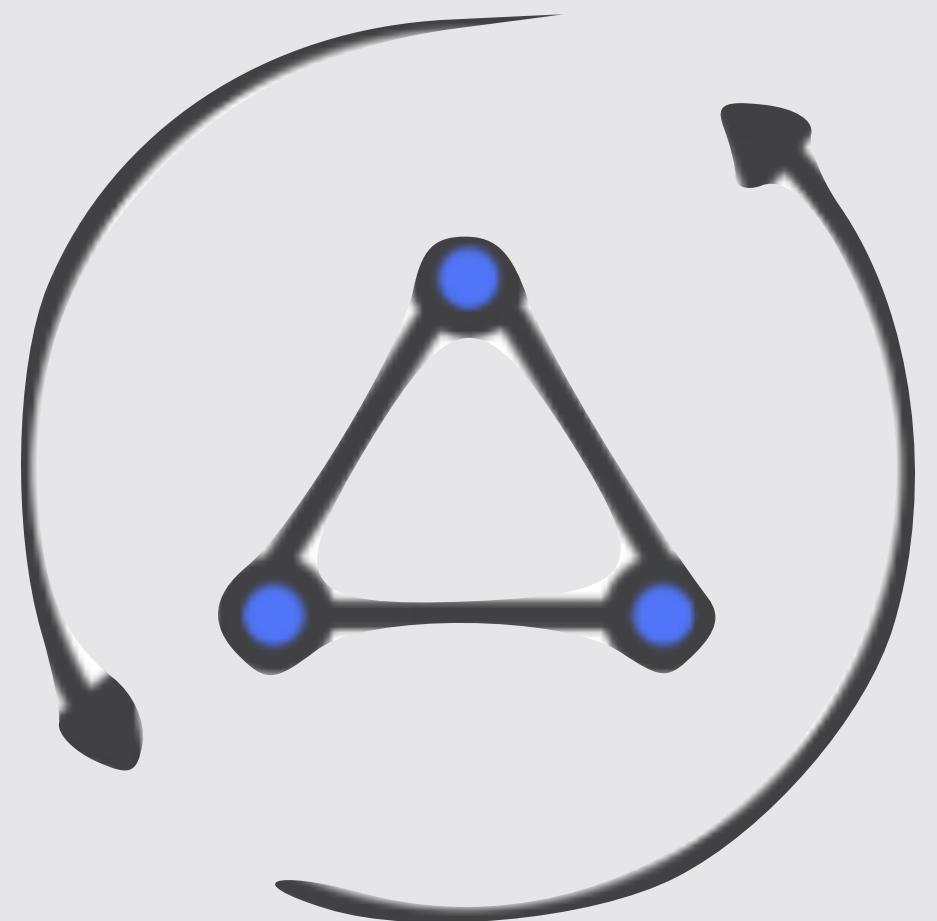
# APIs

# Application Programming Interface

A set of functions or procedures allowing access to data and features using requests.

**Third-party or build your own.**

# Data



@tzmanics

Netlify DevEx team

July, 2020

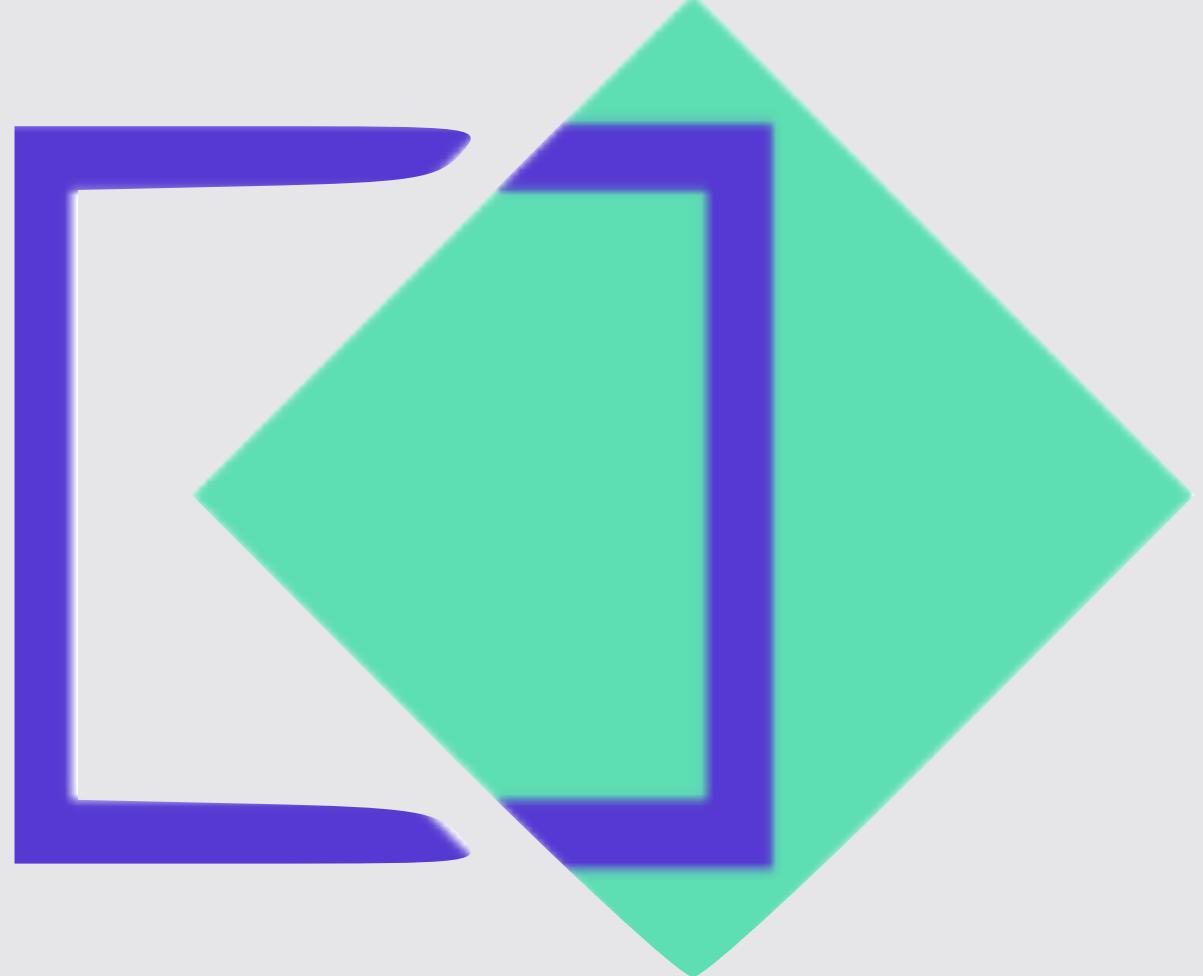
 netlify

# Headless CMS

# Headless CMS

Back-end only Content Management System making content accessible through an API.

**Detaching the UI from the API.**



# Auth



{okta}

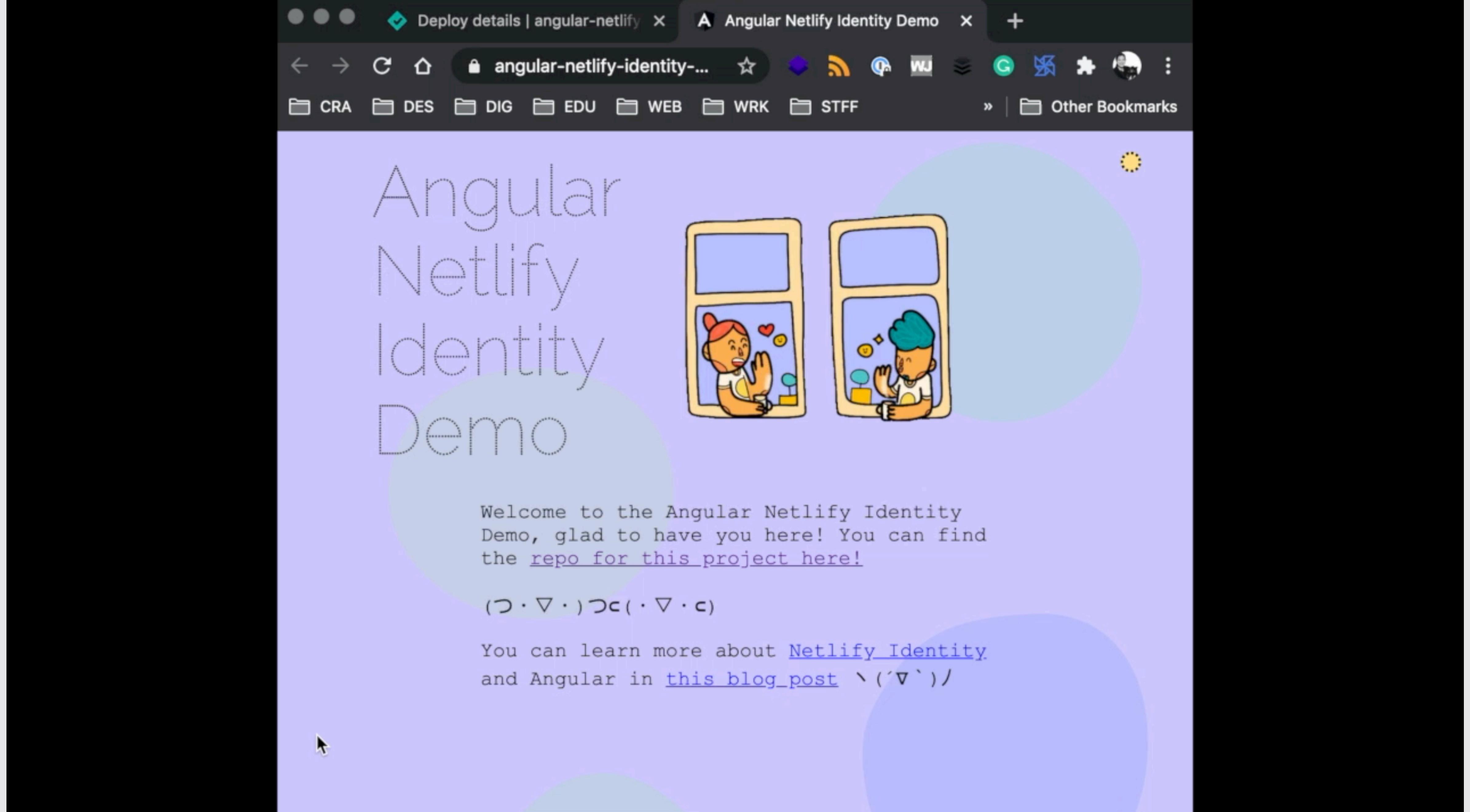


By [Tara Z. Manicsic](#) in [Guides & Tutorials](#) • July 7, 2020

# Authenticating Users with Angular and Netlify Identity

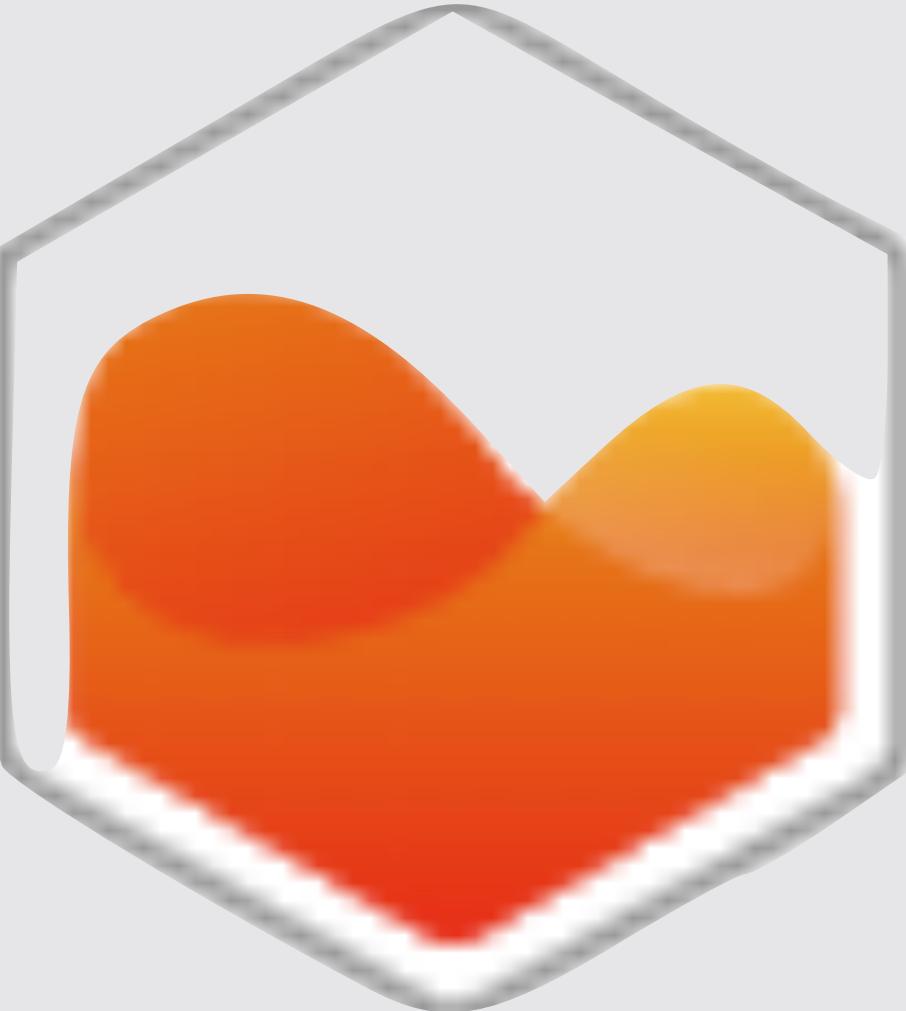
Setting up login and user permissions doesn't have to be hard. Breaking that security, that part should be hard. I've had the opposite of both, hard-to-code authentication and fragile security. So, I was stoked to work with our [Netlify Identity](#) features. Let me give you a little preview. In this post we'll:

- add Netlify Identity login to an Angular Project
- add external OAuth providers
- assign user roles



The screenshot shows a browser window with the title bar "Identity | angular-netlify-identity" and "Angular Netlify Identity Demo". The address bar displays "app.netlify.com/sites/an...". The bookmarks bar includes links to CRA, DES, DIG, EDU, WEB, WRK, STFF, and Other Bookmarks. The main content area features the Netlify logo and navigation links: Overview, Deploys, Plugins, Functions, Identity (which is bolded), Forms, Large Media, Split Testing, Analytics, and Settings. A central callout box highlights the "Identity Enterprise" feature, providing the API endpoint "https://angular-netlify-identity-demo.netlify.app/.netlify/identity" and a link to "Learn more about Identity in the docs". It also includes two buttons: "Settings and usage" and "Identity audit log".

# ECommerce & Payments



Tito



stripe

# Search



# Serverless Functions

# Serverless Functions

A single purpose function created by a developer that is then hosted, maintained, and executed by a cloud computing company

**JavaScript functions hosted and maintained by an infrastructure provider.**

# Azure Functions

More than just event-driven serverless compute



# AWS Lambda

Run code without thinking about servers. Pay only for the compute time you consume.

# Netlify Functions

A modern, scalable backend built into every Netlify account. No setup, servers, or ops required.

[Get started in seconds](#)

[Talk to an expert →](#)

## Full-powered sites—no servers required

Meet the modern backend. Infinitely scalable. Genuinely simple.

# Serverless made painless

Powered by AWS Lambda. Simplified by Netlify.

Write JavaScript or Go APIs to perform just about any backend task, scaling automatically as demand increases. Setting up AWS Lambdas on your own can be fairly complex—tutorials average 20 steps! With Netlify, simply set a folder and drop in your functions. Everything else is automatic.



## How Netlify Functions work

Write simple functions that automatically become APIs.

Netlify deploys the functions you write as full API endpoints and will even run them automatically in response to events (like a form submission or a user login). Functions receive request context or event data, and return data back to your frontend.

[Learn more in the docs →](#)

/functions/split-test-activated.js

```
exports.handler = async (event, context) => {
  return {
    statusCode: 200,
    body: "We are now split testing!"
  };
}
```



THE POWER OF **SERVERLESS**  
FOR FRONT-END DEVELOPERS★

# SERVICES

## THE MAJOR PROVIDERS

The biggest names in serverless, providing wide swaths of functionality.



### Amazon Web Services

[Lambda](#) (Cloud Functions), [S3](#) (File Storage), [Amplify](#) [Console](#) (Hosting with CI/CD and HTTPS), [AppSync](#) (Realtime GraphQL), and [Cognito](#) (Auth) are probably the most relevant things to front-end developers.



### Google Cloud Platform

More of a major infrastructure provider in vein of Amazon Web Services than a toolkit for building out an app like Firebase is.

## SERVICES

- Major Providers
- Cloud Functions
- Auth
- Forms
- Notifications
- Static Site Generators
- CMSs
- Hosting
- Media
- Ecommerce
- Payments
- Monitoring
- Realtime
- Data Storage
- Search

## RESOURCES

## IDEAS

# Creating a better development experience through automated workflow

# Git Workflow

# Git Workflow

Utilizing git repos and functionality like clone, forking, Pull Requests, merge checks and more to create an easily distributed, checked, and quick interaction development process.

**Using git to organize and check code from many devs.**



netlify bot commented 7 days ago

...

Deploy preview for www ready!

Built with commit [31684e1](#)

<https://deploy-preview-3254--www.netlify.app>

Add more commits by pushing to the [cms/blog/2020-07-04-creating-an-angular-jamstack-blog](#) branch on [netlify/netlify-www](#).



### Review required

At least 1 approving review is required by reviewers with write access. [Learn more.](#)

[Add your review](#)



### All checks have passed

1 neutral and 5 successful checks

[Hide all checks](#)



#### Pages changed - www

Completed in 2m — 17 new files uploaded

[Details](#)



#### Header rules - www

Successful in 2m — 2 header rules processed

[Details](#)



#### Mixed content - www

Successful in 2m — No mixed content detected

[Details](#)



#### Redirect rules - www

Successful in 2m — 187 redirect rules processed

[Details](#)



#### Required PR Labels — Has required label - type: content

[Details](#)



#### netlify/www/deploy-preview — Deploy preview ready!

[Details](#)



### Merging is blocked

Merging can be performed automatically with 1 approving review.

## Automated Builds

Because Jamstack markup is prebuilt, content changes won't go live until you run another build. Automating this process will save you lots of frustration. You can do this yourself with webhooks, or use a publishing platform that includes the service automatically.

## Atomic Deploys

As Jamstack projects grow really large, new changes might require re-deploying hundreds of files. Uploading these one at a time can cause inconsistent state while the process completes. You can avoid this with a system that lets you do “atomic deploys,” where no changes go live until all changed files have been uploaded.

## CONTENT TYPES

[Restaurants](#)[Users](#)

## PLUGINS

[Content Manager](#)[Content Type Builder](#)[Files Upload](#)[Roles & Permissions](#)

## GENERAL

[Plugins](#)[Marketplace](#)[Configurations](#)

MORE VIDEOS

Documentation



## CONTENT TYPES

[Permission \(from: users-permissions\)](#)[Restaurant](#)[Role \(from: users-permissions\)](#)[User \(from: users-permissions\)](#)[+ Add A Content Type](#)

## GROUP

[+ Add A Group](#)

## DOCUMENTATION

- Read more about Content Types in our [guide](#).

Restaurant 

Restaurant Listings

2 fields

[+ Add Another Field](#) [Name](#)

String

  [Description](#)

Rich text

 [+ Add Another Field](#) Delete

## CONTENT TYPES

▶ Restaurants

▶ Users

## PLUGINS

 Content Manager Content Type Builder Files Upload Roles & Permissions

## GENERAL

 Plugins Marketplace Configurations

## CONTENT TYPES

 Permission Restaurant Role User Category

+ Add A Content

## GROUP

+ Add A Group

## DOCUMENTATION

• Read more about

Category 

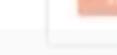
There is no description for this Content Type

CT

Category

x

## Select A Field For Your Content Type

 String Titles, names, paragraphs, list of names  Text Descriptions, text paragraphs, articles  Number Everything that is number  Boolean Yes or no, 1 or 0, true or false  Date Event date, opening hours  Email User's email...  Password User password...  JSON Data in JSON format  Media Images, videos, PDFs and other files  Relation Refers to a Content Type  Enumeration List of choices  Rich Text Formatting and creating text  Group Refers to a Group  Delete

master ▾

cinasia-backend / api / article / models / article.settings.json

Go to file

...



**tzmanics** Re-creates models from cinasia backend (#8) ...

Latest commit 8708883 12 days ago History

1 contributor

45 lines (45 sloc) | 764 Bytes

Raw

Blame



```
1  {
2    "kind": "collectionType",
3    "collectionName": "articles",
4    "info": {
5      "name": "Article"
6    },
7    "options": {
8      "increments": true,
9      "timestamps": true
10   },
11   "attributes": {
12     "title": {
13       "type": "string"
14     },
15     "mainImage": {
16       "model": "file",
17       "via": "related",
18       "allowedTypes": [
19         "images",
20         "files",
21         "videos"
22       ]
23     }
24   }
25 }
```

# Moar Technologies



netlify



# Bringing full-stack to the Jamstack

Do you love the Jamstack philosophy but need a database-backed web app? Want great developer experience and easy scaling? Redwood is here! Built on React, GraphQL, and Prisma, Redwood works with the components and development workflow you love, but with simple conventions and helpers to make your experience even better.

[Start the Tutorial](#)[Read the Docs](#)

# The Netlify Blog



Exploring the Jamstack, static sites, and the future of web development.

[Subscribe to our newsletter](#) to make sure you don't miss anything.

## All posts

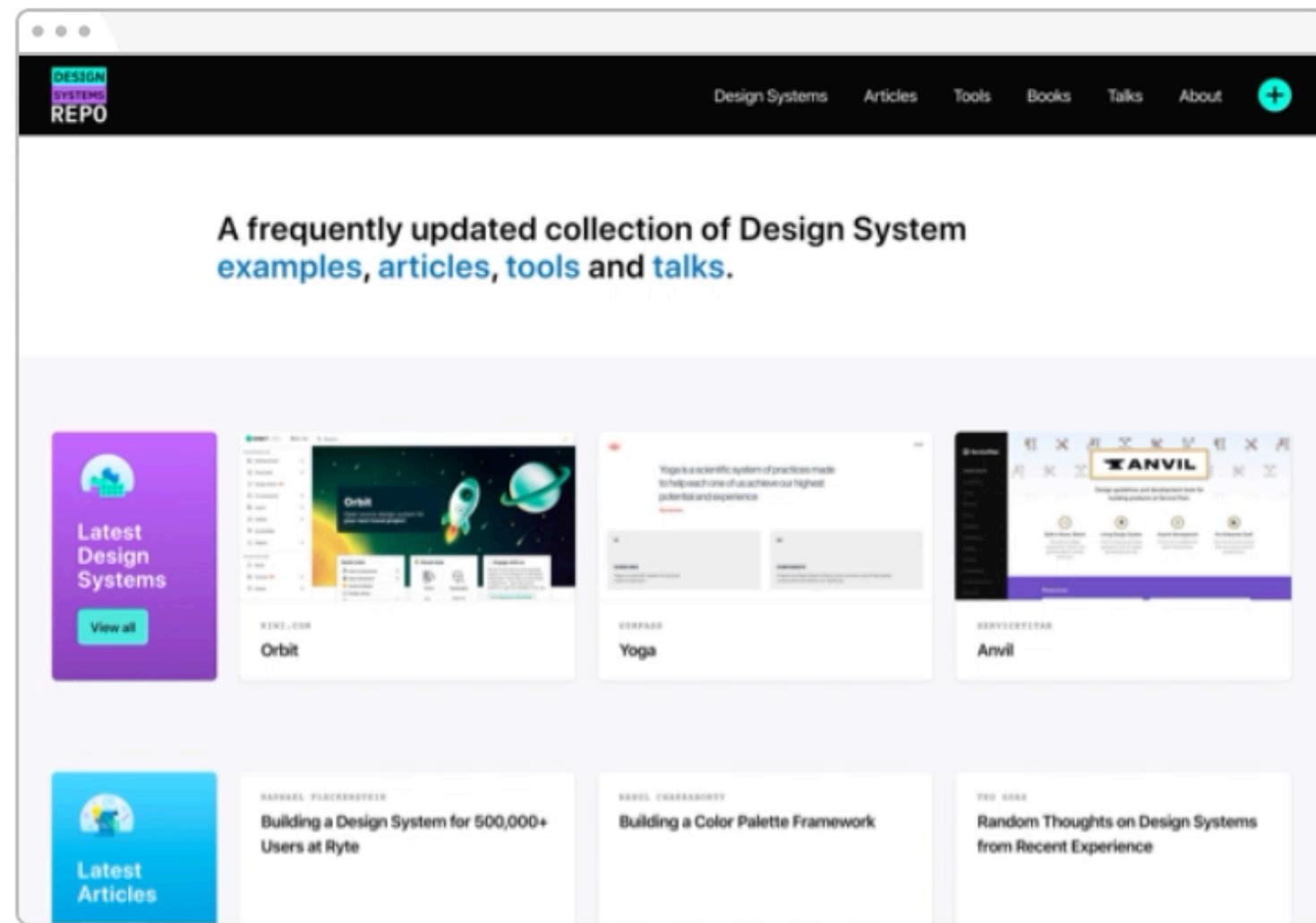
### Create a Fully-fledged Jamstack commerce store with Commerce.js and Netlify

Learn how to use Commerce.js and React with Next.js to create your own Jamstack eCommerce sites in this detailed tutorial with demonstrations and code samples.

#### NEWSLETTER

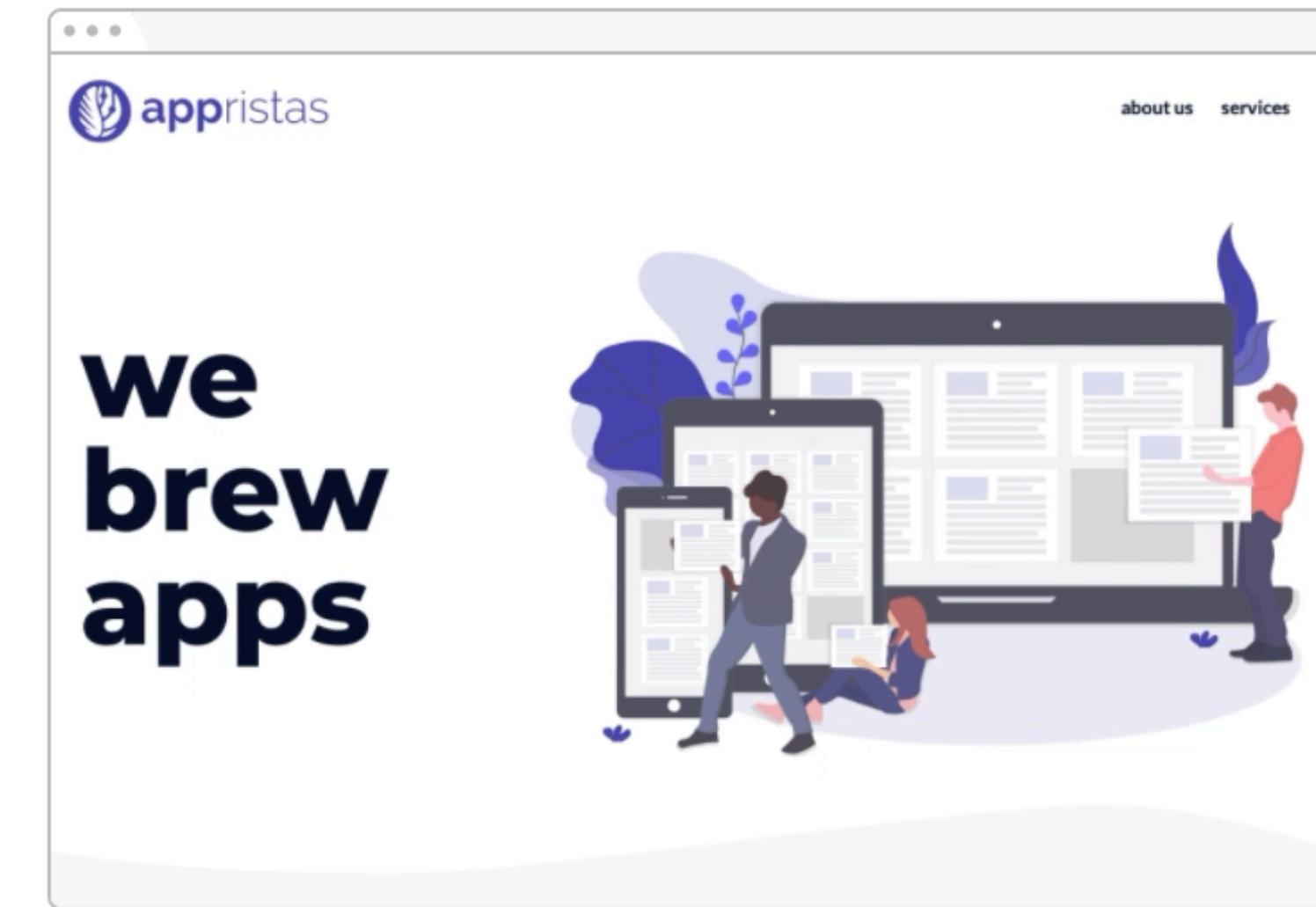
Subscribe to our newsletter to make sure you don't miss anything.

If you can use it on the web, you can probably build it with the Jamstack. Millions already do. Check out the Jamstacks of these examples below:



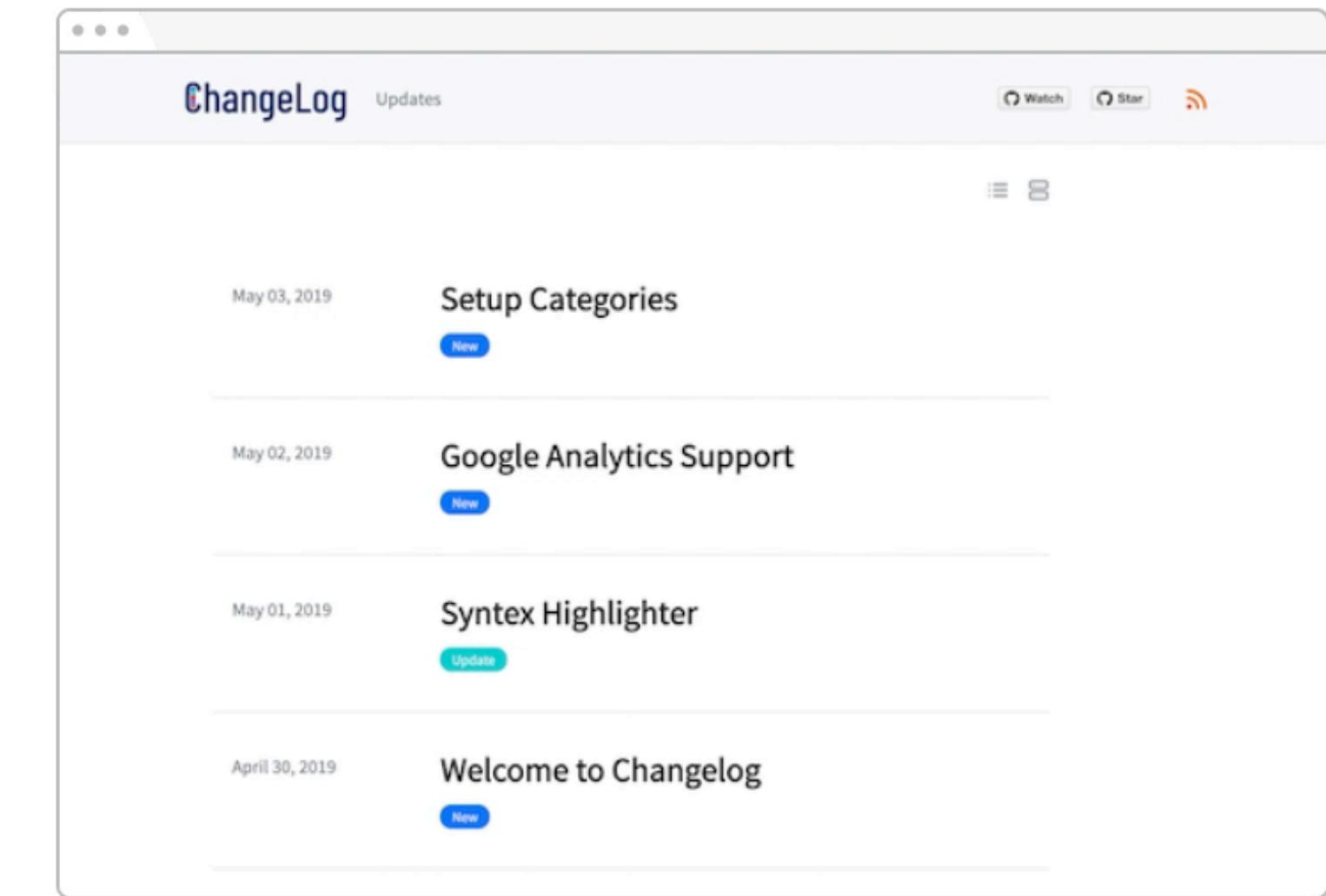
## Design Systems Repo

- Netlify
- Gridsome
- GraphQL
- Netlify CMS
- Markdown
- Github



## Appristas

- Firebase Hosting
- Create React App
- React Snap
- Webpack
- Gitlab
- Sass
- Custom API



## Changelog

- Netlify
- Jekyll
- Netlify CMS
- Disqus
- Bootstrap
- Sass

# Thank you

Learn more at [netlify.com](https://netlify.com)

Find me at [@tzmanics](https://twitter.com/tzmanics)

