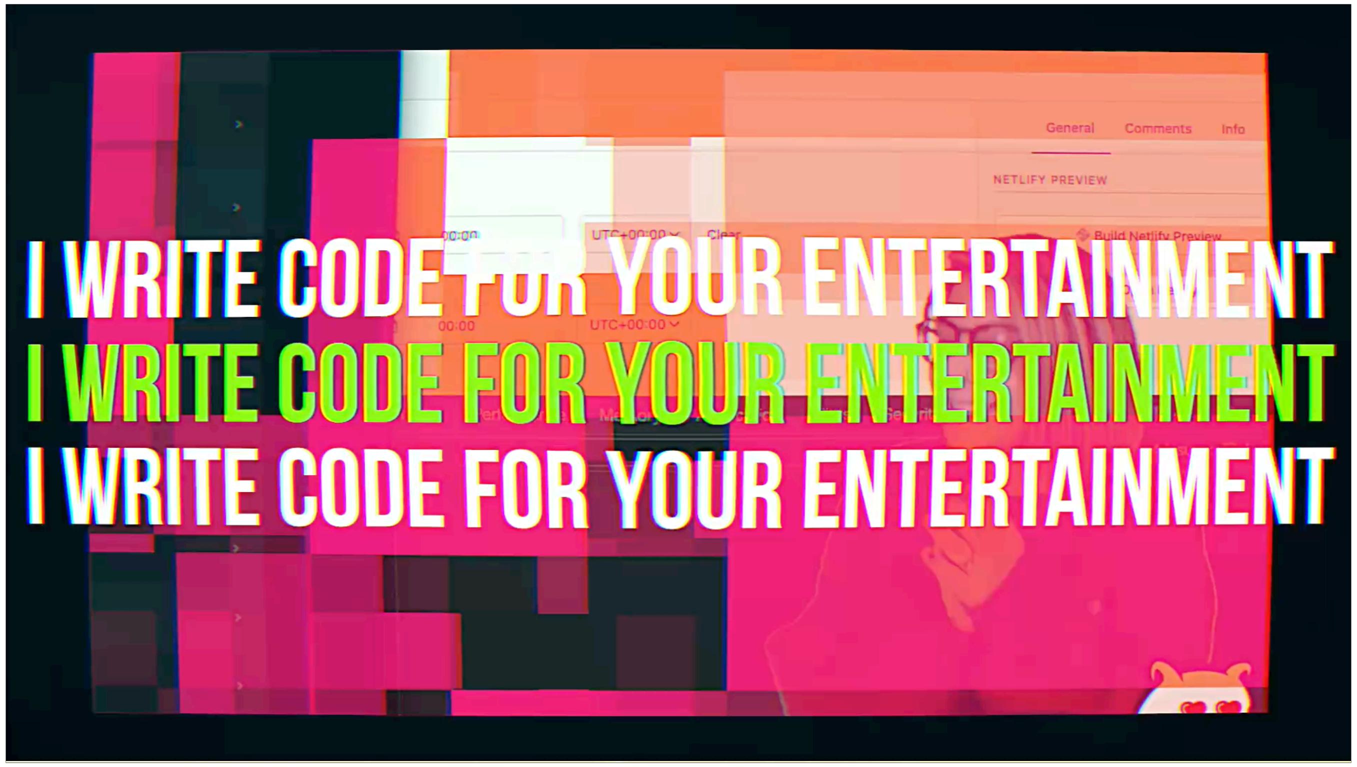
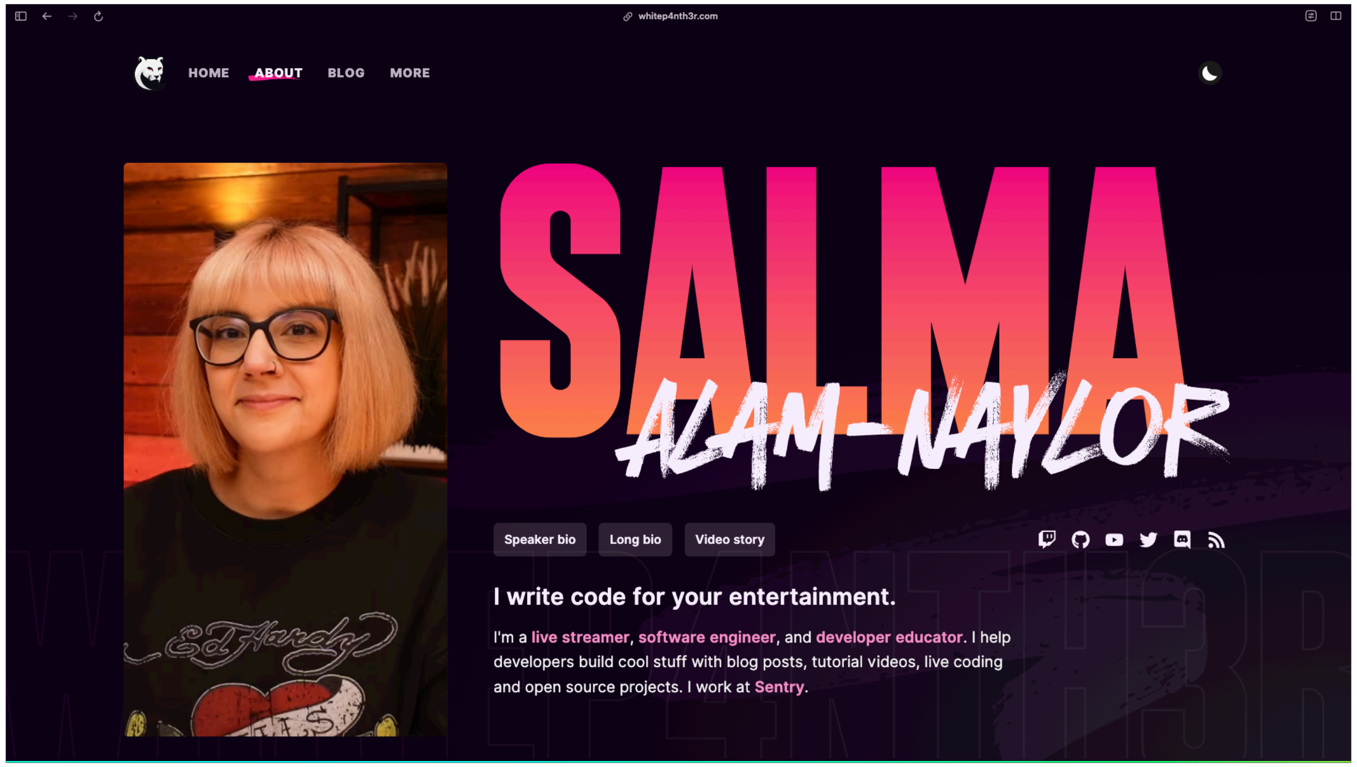
SALMA ALAM-NAYLOR

ENTERTAINMENT AS CODE

Hi I’m Salma and this is entertainment as code.


I WRITE CODE FOR YOUR ENTERTAINMENT
I WRITE CODE FOR YOUR ENTERTAINMENT
I WRITE CODE FOR YOUR ENTERTAINMENT
General
Comments
Info
NETLIFY PREVIEW
Build Netlify Preview
UTC+00:00
Clear



SALMA ALAM-NAYLOR

I write code for your entertainment.

I'm a live streamer, software engineer, and developer educator. I help developers build cool stuff with blog posts, tutorial videos, live coding and open source projects. I work at Sentry.

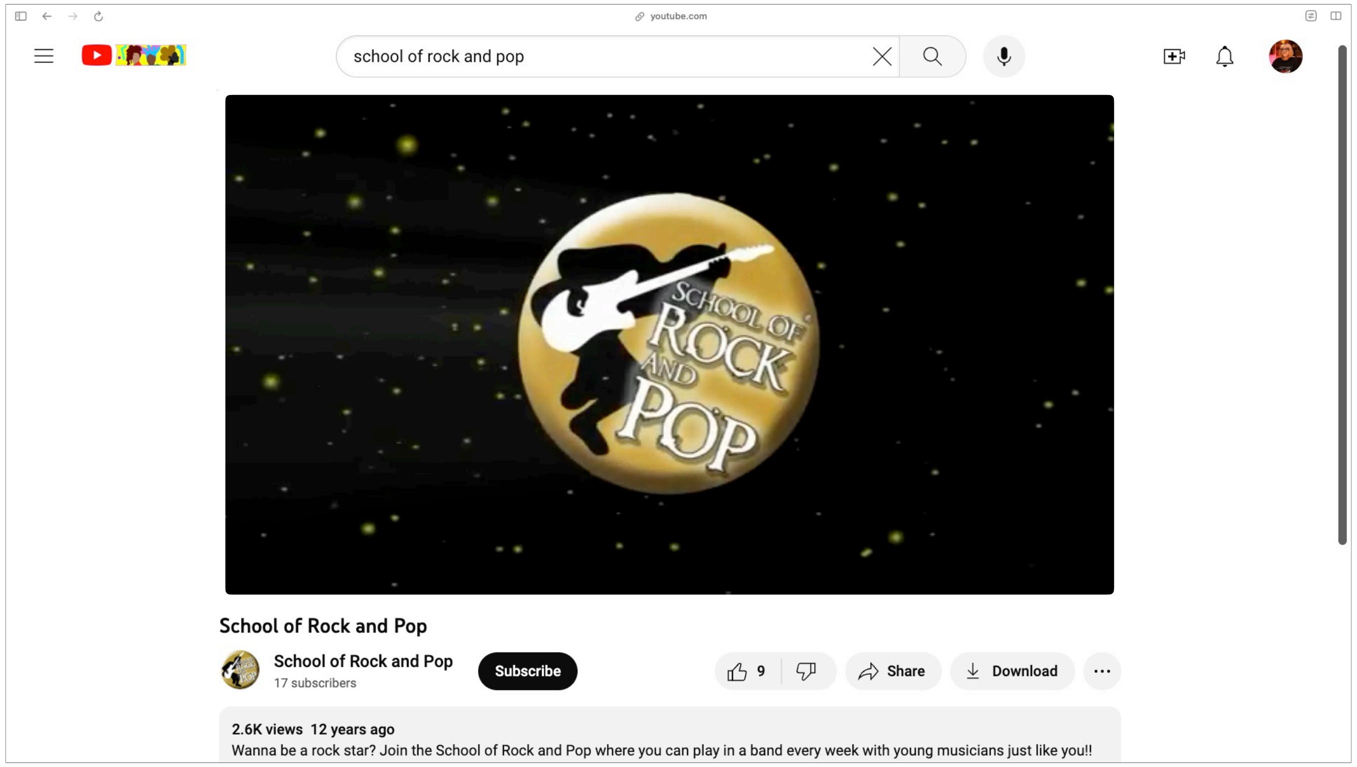
I write code for your entertainment
I’m a live streamer on twitch, software engineer and developer educator.
I work at Sentry
And I change my hair all the time.

Let’s travel back in time to 2008

HISTORY AND EARLY INFLUENCES

When I graduated from the RNCM in Manchester with a degree in composition
I wanted to be a film composer, for your entertainment
At the time I was in a folk band, we released an album, played weddings and festivals for your entertainment
And I was also making terrible websites, graphics, and album art for my musician friends

But as a musician, in order to pay the bills, you usually end up being a teacher



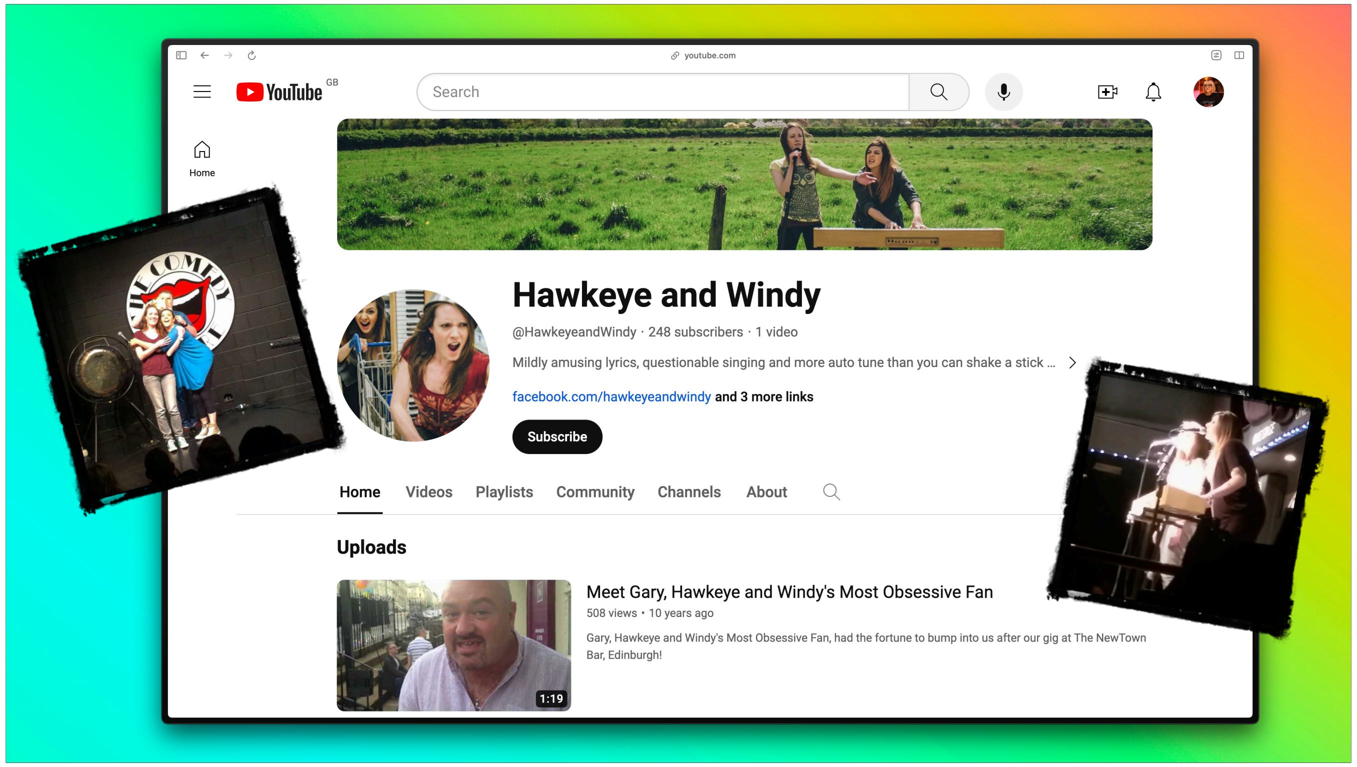
School of Rock and Pop

School of Rock and Pop
17 subscribers

2.6K views 12 years ago
Wanna be a rock star? Join the School of Rock and Pop where you can play in a band every week with young musicians just like you!!

2010
Set up in Manchester
I was teaching children how to be in rock bands… for your entertainment
Helping them put music into a real, tangible context, rather than practising alone in their rooms

I really liked this whole teaching thing… Qualified music teacher and worked in secondary schools and sixth forms for a few years



Hawkeye and Windy

@HawkeyeandWindy · 248 subscribers · 1 video

Mildly amusing lyrics, questionable singing and more auto tune than you can shake a stick ...

facebook.com/hawkeyeandwindy and 3 more links

Uploads

Meet Gary, Hawkeye and Windy's Most Obsessive Fan

508 views • 10 years ago

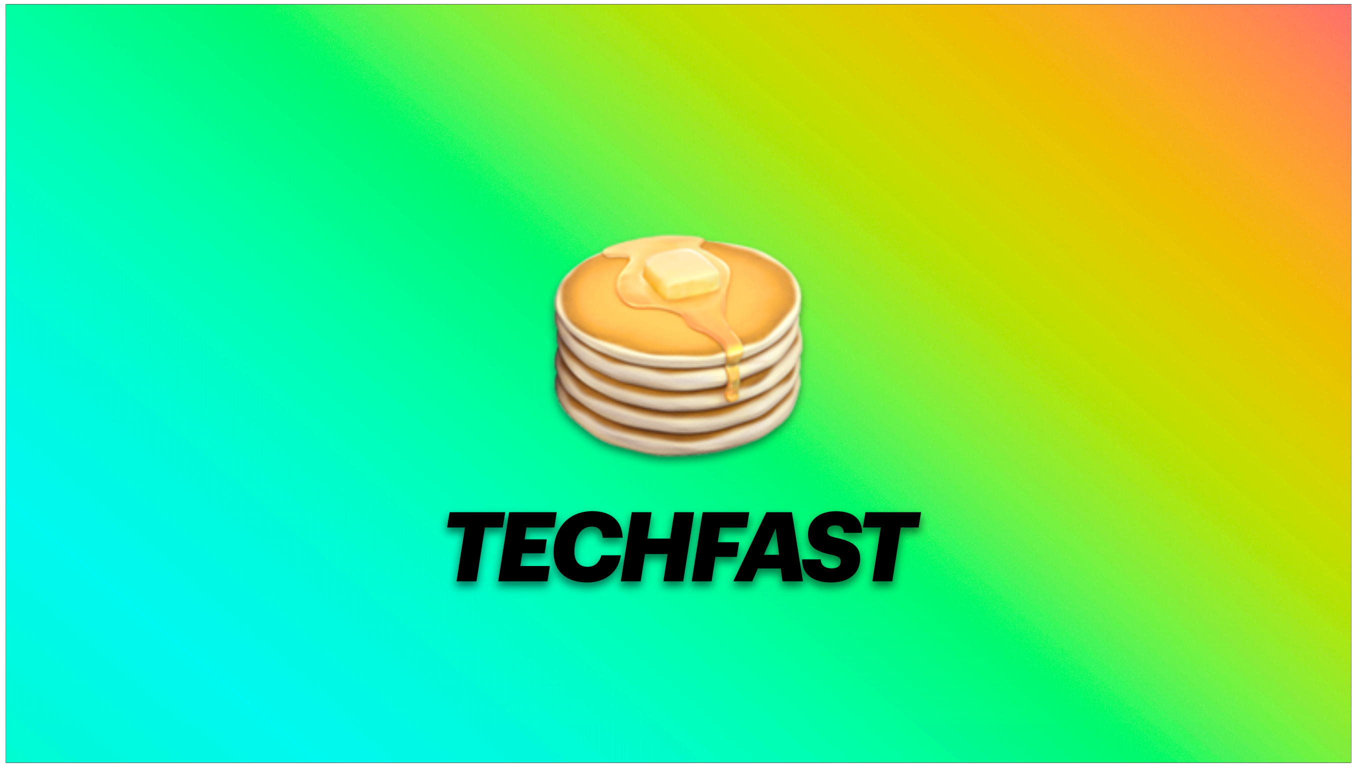
Gary, Hawkeye and Windy's Most Obsessive Fan, had the fortune to bump into us after our gig at The NewTown Bar, Edinburgh!

Around this time I was also a musical comedian for your entertainment
We played some good gigs, won some competitions, went viral on the internet…
You can’t watch our videos now..

And it’s funny because whenever I’m on stage with a mic, I automatically try to tell jokes

HAHA BUSINESS

No jokes, want to take a brief intermission and talk about a new business venture I’m working on
I love tech
And breakfast is the most important meal of the day
I want to merge the two and open a tech themed breakfast and brunch bar…

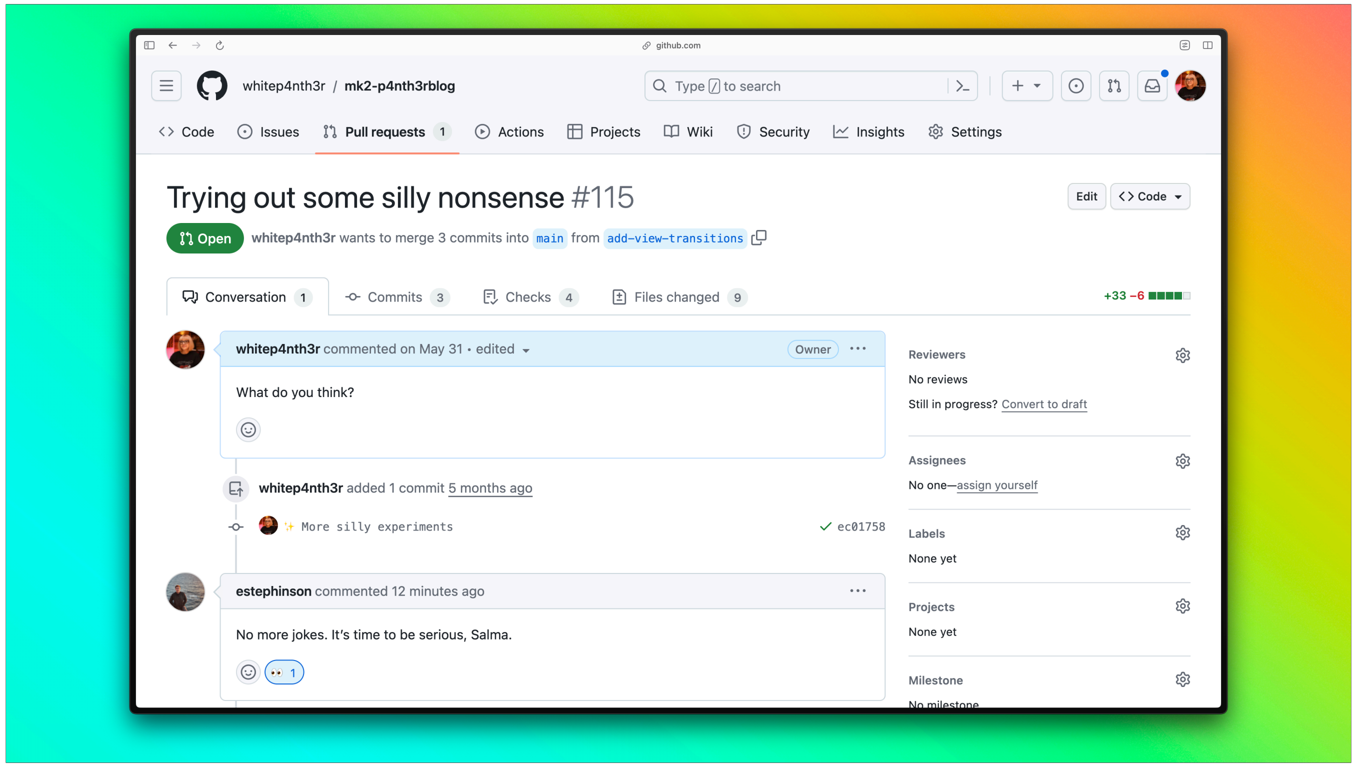


TECHFAST

Techfast, obviously
Full stack of pancakes, eggs and json, POP3 tarts, ChatGPTea (drink), infrastructure as toast, cereal numbers.
Served in little containers, micro servings. The service will be blazing fast.

If there are any investors in the audience today, I am trying to raise the chia seed round, so please talk to me later.

No more jokes. It’s time to be serious, Salma.



Trying out some silly nonsense #115

Open whitep4nth3r wants to merge 3 commits into `main` from `add-view-transitions`

whitep4nth3r commented on May 31 • edited

What do you think?

whitep4nth3r added 1 commit 5 months ago

`More silly experiments` ✓ ec01758

estephinson commented 12 minutes ago

No more jokes. It's time to be serious, Salma.

Which is often the feedback I get on my PRs when I'm trying out silly nonsense.

Let's fast forward to 2014

2014

MY FIRST JOB IN TECH

When I quit teaching and got my first job in tech
Don’t have time to tell the full story (it’s on my YouTube channel if you’re interested)
And this led

2014-2020
MANY MORE JOBS IN TECH

To many more jobs in tech, as a front end developer and tech lead at
Product agencies
Startups
Global e-commerce brands

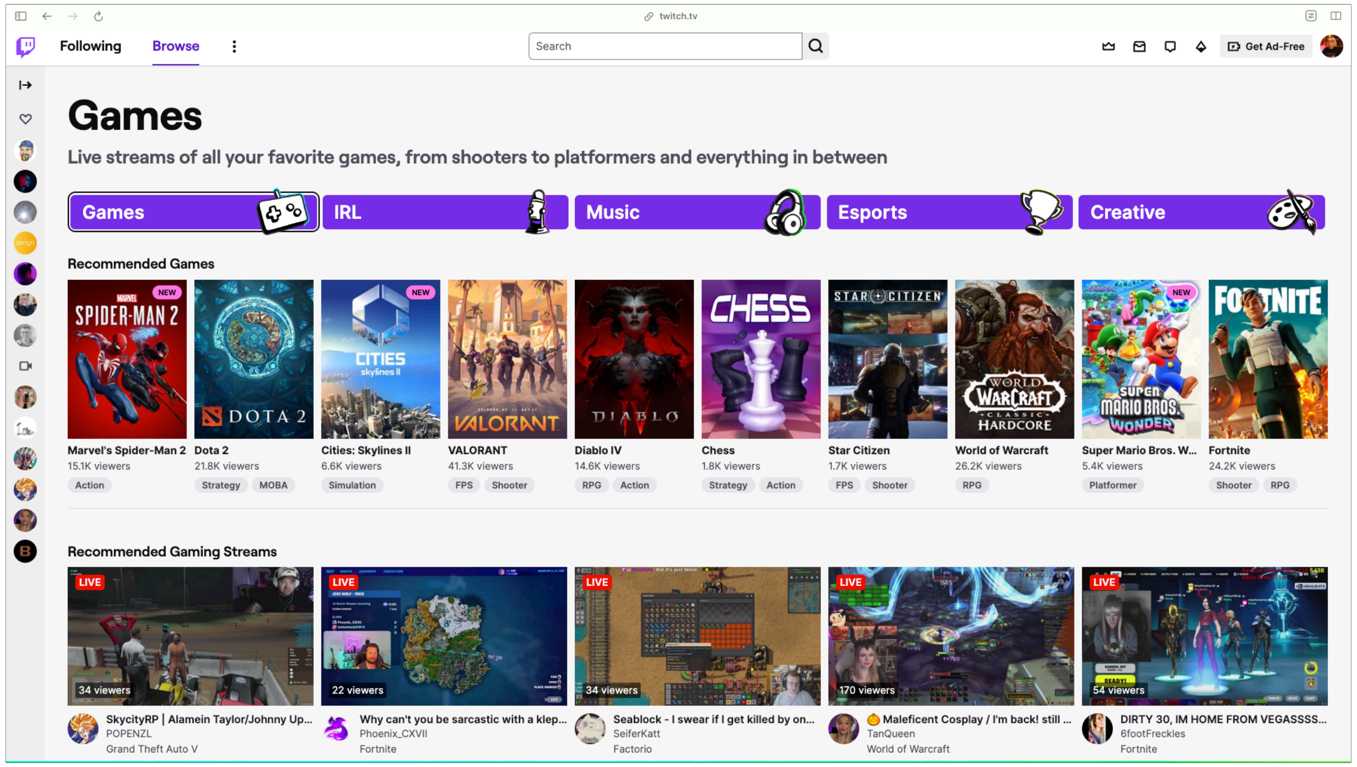
2020

EVERYTHING CHANGED

And in 2020, everything changed

The pandemic descended, everyone stayed inside
And found new ways to spend their time

Here’s where Twitch comes into my life



Games

Live streams of all your favorite games, from shooters to platformers and everything in between

Recommended Games

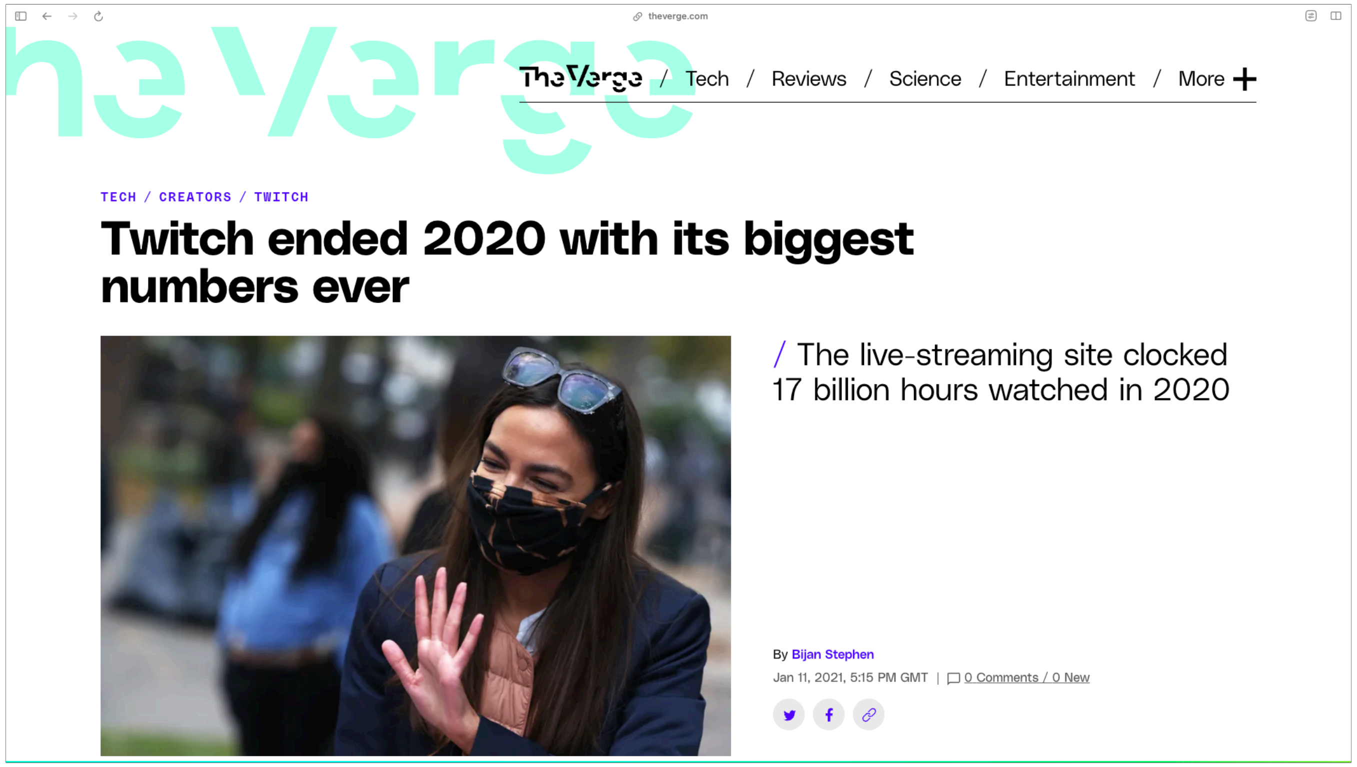
Marvel's Spider-Man 2 · 15.1K viewers · Action
Dota 2 · 21.8K viewers · Strategy · MOBA
Cities: Skylines II · 6.6K viewers · Simulation
VALORANT · 41.3K viewers · FPS · Shooter
Diablo IV · 14.6K viewers · RPG · Action
Chess · 1.8K viewers · Strategy · Action
Star Citizen · 1.7K viewers · FPS · Shooter
World of Warcraft · 26.2K viewers · RPG
Super Mario Bros. W... · 5.4K viewers · Platformer
Fortnite · 24.2K viewers · Shooter · RPG

Recommended Gaming Streams

LIVE · 34 viewers · SkycityRP | Alamein Taylor/Johnny Up... · POPENZL · Grand Theft Auto V
LIVE · 22 viewers · Why can't you be sarcastic with a klep... · Phoenix_CXVII · Fortnite
LIVE · 34 viewers · Seablock - I swear if I get killed by on... · SeiferKatt · Factorio
LIVE · 170 viewers · 🎃 Maleficent Cosplay / I'm back! still ... · TanQueen · World of Warcraft
LIVE · 54 viewers · DIRTY 30, IM HOME FROM VEGASSSS... · 6footFreckles · Fortnite

The main focus and the largest categories on live streaming platform Twitch have always been centred around gaming
As time went on, creators started streaming music, arts and crafts, talk shows, charity events, 24/7 live feeds of farm animals… anything you could probably ever dream of…

In 2020 for a short time I was glued to a behind the counter live stream of the comings and goings of a fish and chip shop


theverge.com
The Verge / Tech / Reviews / Science / Entertainment / More +

TECH / CREATORS / TWITCH

Twitch ended 2020 with its biggest numbers ever

/ The live-streaming site clocked 17 billion hours watched in 2020

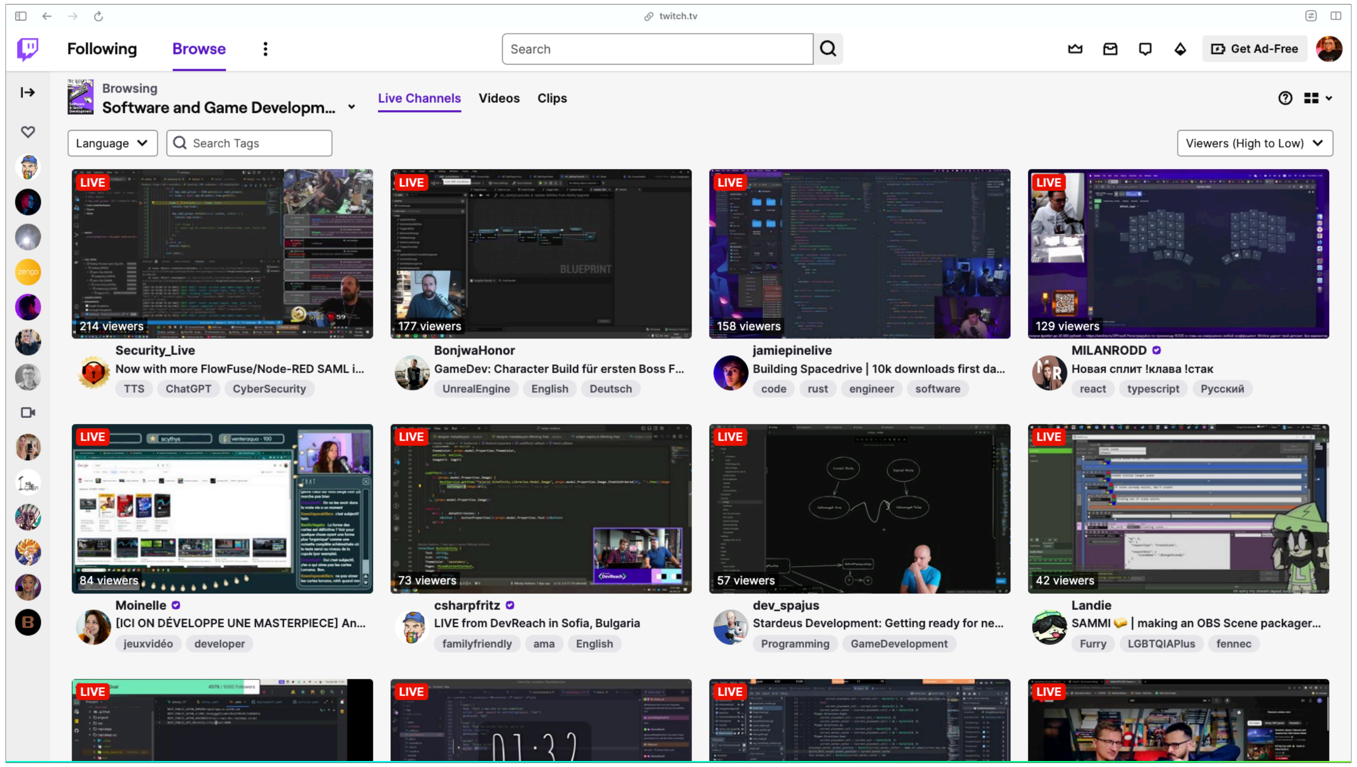
By Bijan Stephen

Jan 11, 2021, 5:15 PM GMT | 0 Comments / 0 New

Twitch ended 2020 with its biggest numbers ever

17 billion hours were spent on Twitch in 2020
Which was 83 percent higher than 2019

I was part of that
I started watching Twitch in the first lockdown of 2020
And the category that caught my attention was



214 viewers
Security_Live
Now with more FlowFuse/Node-RED SAML i...
TTS ChatGPT CyberSecurity

177 viewers
BonjwaHonor
GameDev: Character Build für ersten Boss F...
UnrealEngine English Deutsch

158 viewers
jamiepinelive
Building Spacedrive | 10k downloads first da...
code rust engineer software

129 viewers
MILANRODD
Новая сплит !клава !стак
react typescript Русский

84 viewers
Moinelle
[ICI ON DÉVELOPPE UNE MASTERPIECE] An...
jeuxvidéo developer

73 viewers
csharpfritz
LIVE from DevReach in Sofia, Bulgaria
familyfriendly ama English

57 viewers
dev_spajus
Stardeus Development: Getting ready for ne...
Programming GameDevelopment

42 viewers
Landie
SAMMI 🧈 | making an OBS Scene packager...
Furry LGBTQIAPlus fennec

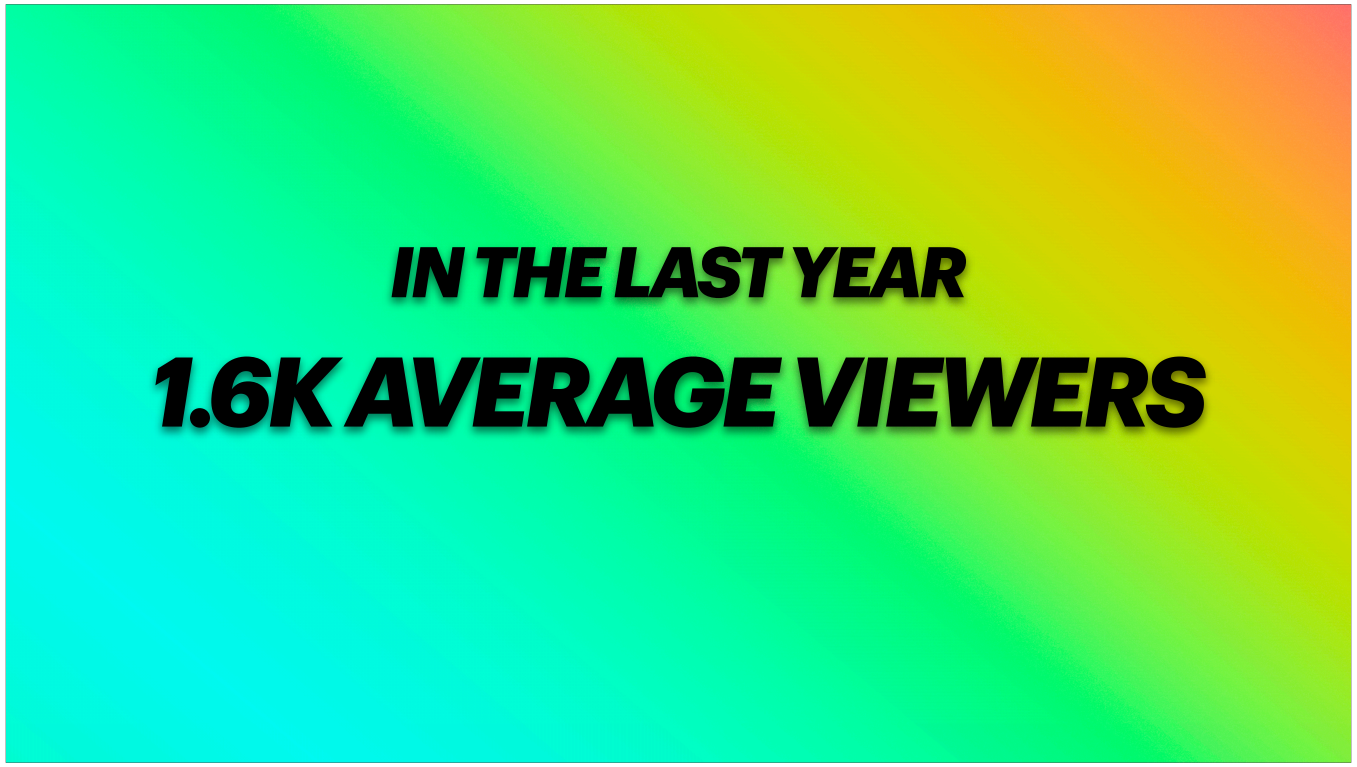
software and game development

People were writing code, live on stream, and people were watching

IN THE LAST YEAR

14M+ HOURS WATCHED

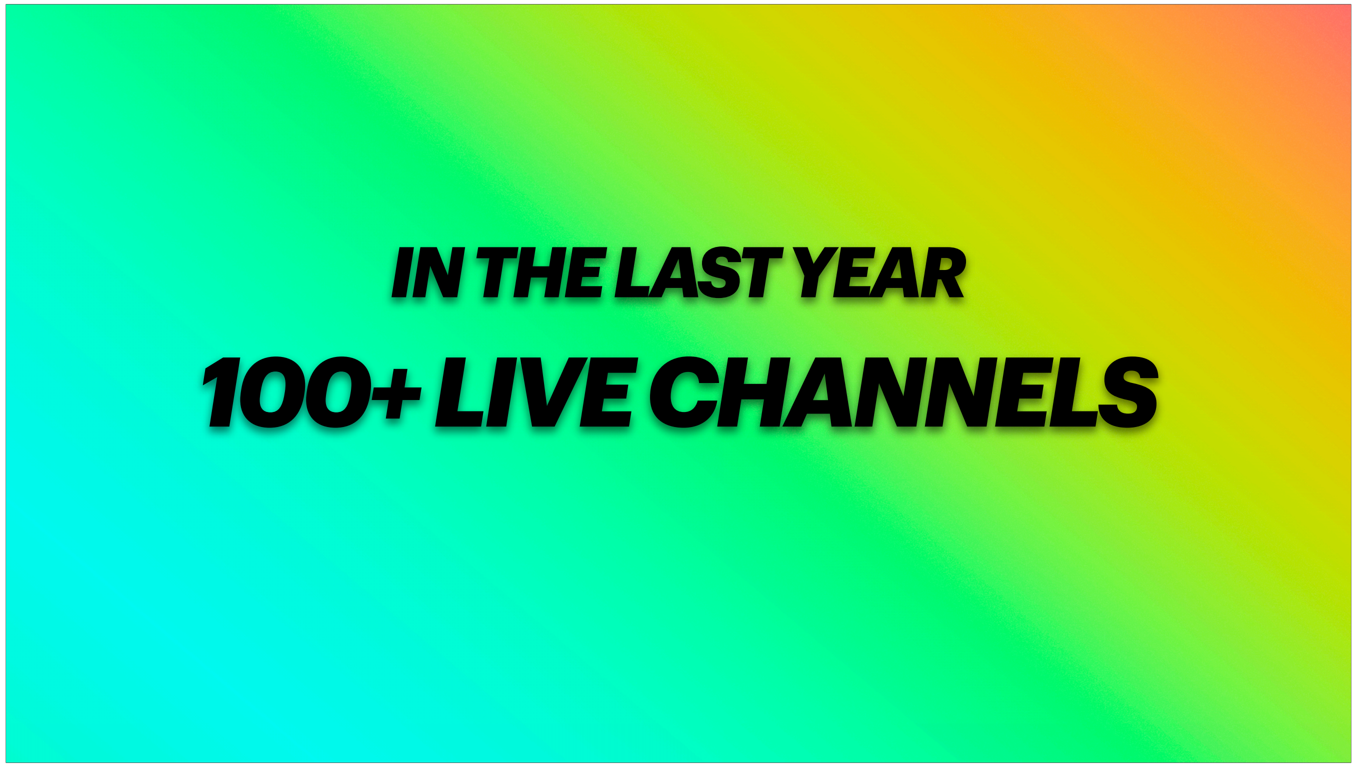
In the last year, over 14m hours of software and game development streams have been watched



IN THE LAST YEAR

1.6K AVERAGE VIEWERS

The category has around 1.6k average viewers at any one time



IN THE LAST YEAR

100+ LIVE CHANNELS

And there are always around 100 live software and game dev channels at any one time

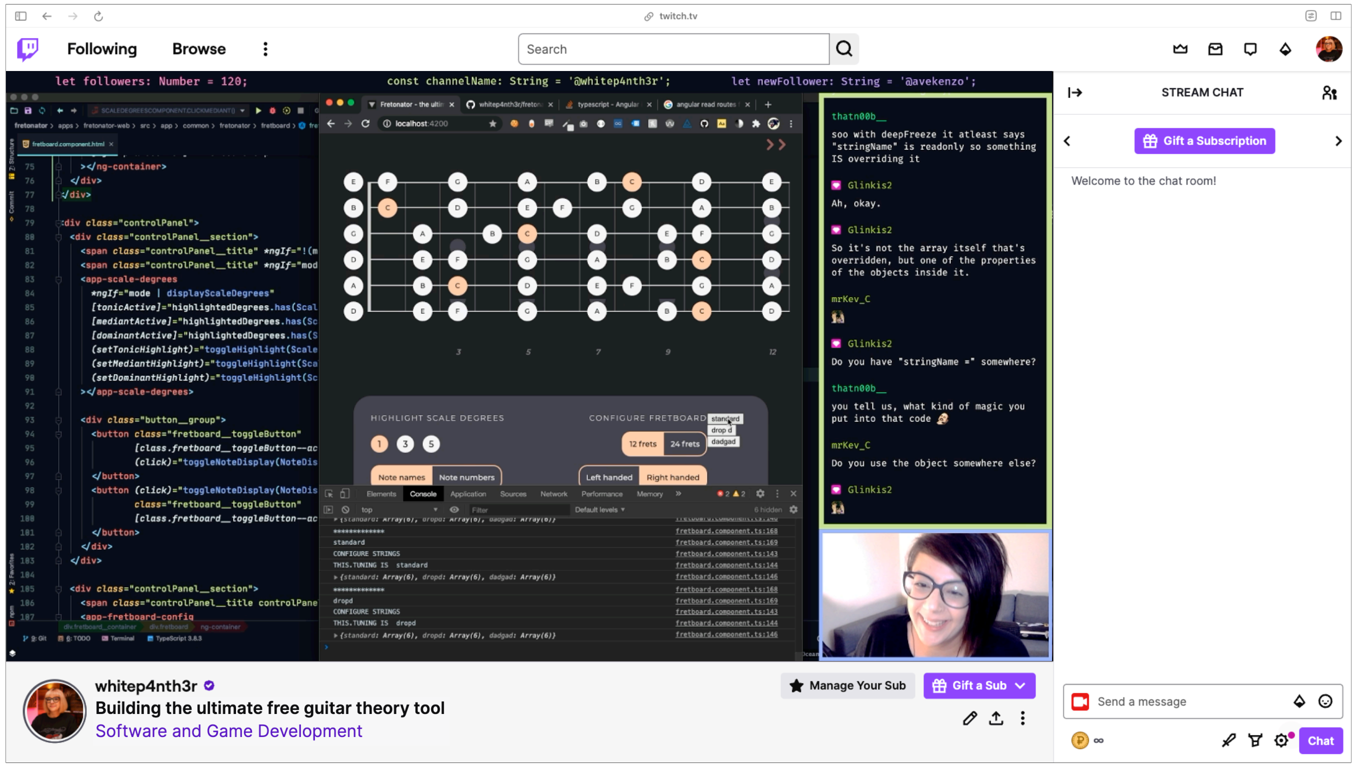
Small category compared to gaming… but with that comes big opportunity

THURSDAY 25TH JUNE 2020

On Thursday 25th June 2020, I went live on Twitch for the first time

Under the username whitep4nth3r
It was quite unintentional that panthers would become integral to everything…

Here’s a clip from a very early stream



```
let followers: Number = 120;        const channelName: String = '@whitep4nth3r';        let newFollower: String = '@avekenzo';
```

fretonator › apps › fretonator-web › src › app › common › fretonator › fretboard ›

fretboard.component.html

```
        ></ng-container>
      </div>
    </div>

    <div class="controlPanel">
      <div class="controlPanel__section">
        <span class="controlPanel__title" *ngIf="!(m
        <span class="controlPanel__title" *ngIf="mod
        <app-scale-degrees
          *ngIf="mode | displayScaleDegrees"
          [tonicActive]="highlightedDegrees.has(Scal
          [mediantActive]="highlightedDegrees.has(Sc
          [dominantActive]="highlightedDegrees.has(S
          (setTonicHighlight)="toggleHighlight(Scale
          (setMediantHighlight)="toggleHighlight(Sca
          (setDominantHighlight)="toggleHighlight(Sc
        ></app-scale-degrees>

        <div class="button__group">
          <button class="fretboard__toggleButton"
                  [class.fretboard__toggleButton--ac
                  (click)="toggleNoteDisplay(NoteDis
          </button>
          <button (click)="toggleNoteDisplay(NoteDis
                  class="fretboard__toggleButton"
                  [class.fretboard__toggleButton--ac
          </button>
        </div>
      </div>

      <div class="controlPanel__section">
        <span class="controlPanel__title controlPane
        <app-fretboard-config
```

```
▸{standard: Array(6), dropd: Array(6), dadgad: Array(6)}
*************                                              fretboard.component.ts:168
standard                                                   fretboard.component.ts:169
CONFIGURE STRINGS                                          fretboard.component.ts:143
THIS.TUNING IS  standard                                   fretboard.component.ts:144
▸{standard: Array(6), dropd: Array(6), dadgad: Array(6)}   fretboard.component.ts:146
*************                                              fretboard.component.ts:168
dropd                                                      fretboard.component.ts:169
CONFIGURE STRINGS                                          fretboard.component.ts:143
THIS.TUNING IS  dropd                                      fretboard.component.ts:144
▸{standard: Array(6), dropd: Array(6), dadgad: Array(6)}   fretboard.component.ts:146
```

thatn00b__
soo with deepFreeze it atleast says "stringName" is readonly so something IS overriding it

Glinkis2
Ah, okay.

Glinkis2
So it's not the array itself that's overridden, but one of the properties of the objects inside it.

mrKev_C

Glinkis2
Do you have "stringName =" somewhere?

thatn00b__
you tell us, what kind of magic you put into that code

mrKev_C
Do you use the object somewhere else?

Glinkis2

whitep4nth3r
Building the ultimate free guitar theory tool
Software and Game Development

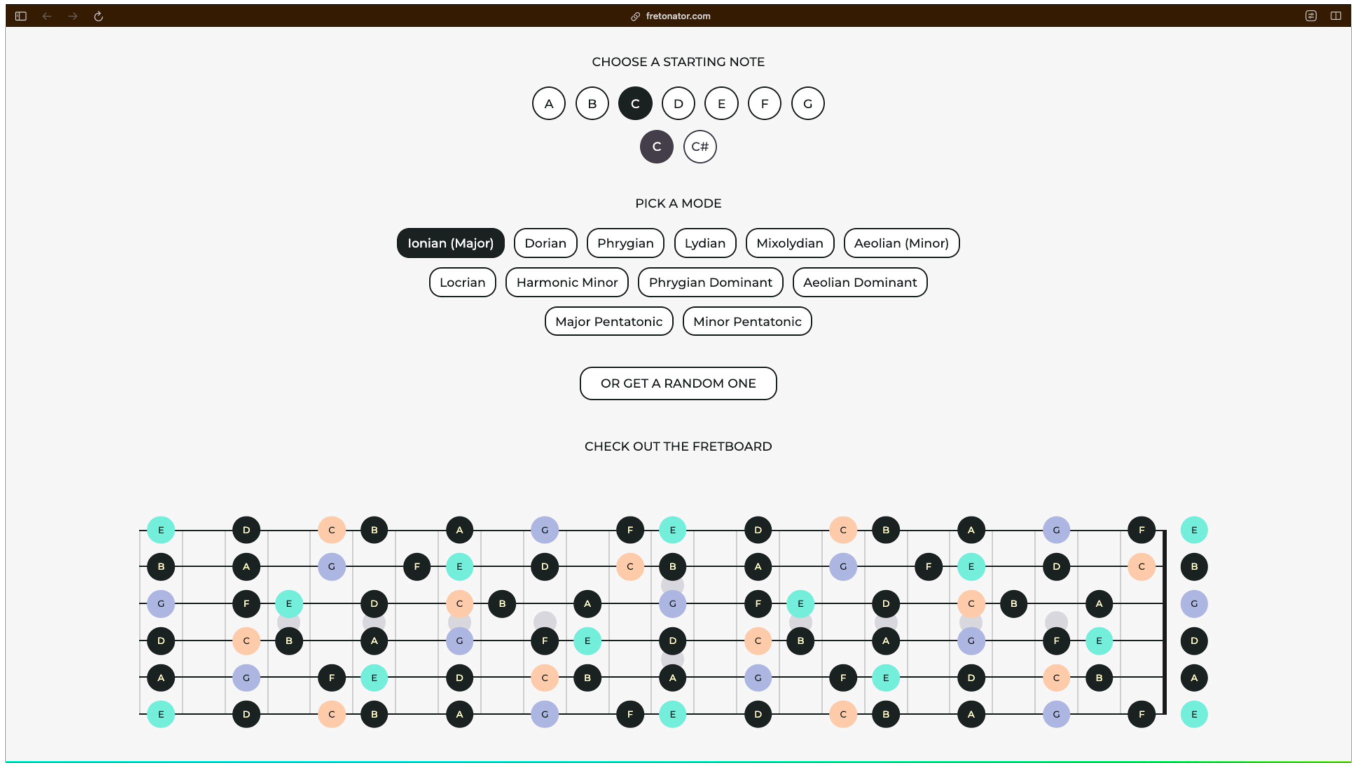
In this clip, I was building my first open source app — fretonator
Guitar app to learn scales and modes

BUT, WHY?

- Why did I do this? Other than boredom and nowhere to go

- the channels I saw were mainly backend dev, mainly men
- maybe I could fill a gap in the market
- I was building fretonator anyway — started building it in March of that year because I wanted to learn Angular
- for my husband — he wanted to learn scale and mode theory on the guitar, my traditional music teaching methods were failing him, he needed a way to learn interactively on his own

Let me show you a little bit about what I was building



CHOOSE A STARTING NOTE

A B [✓ C] D E F G

[✓ C] C#

PICK A MODE

[✓ Ionian (Major)] Dorian Phrygian Lydian Mixolydian Aeolian (Minor)

Locrian Harmonic Minor Phrygian Dominant Aeolian Dominant

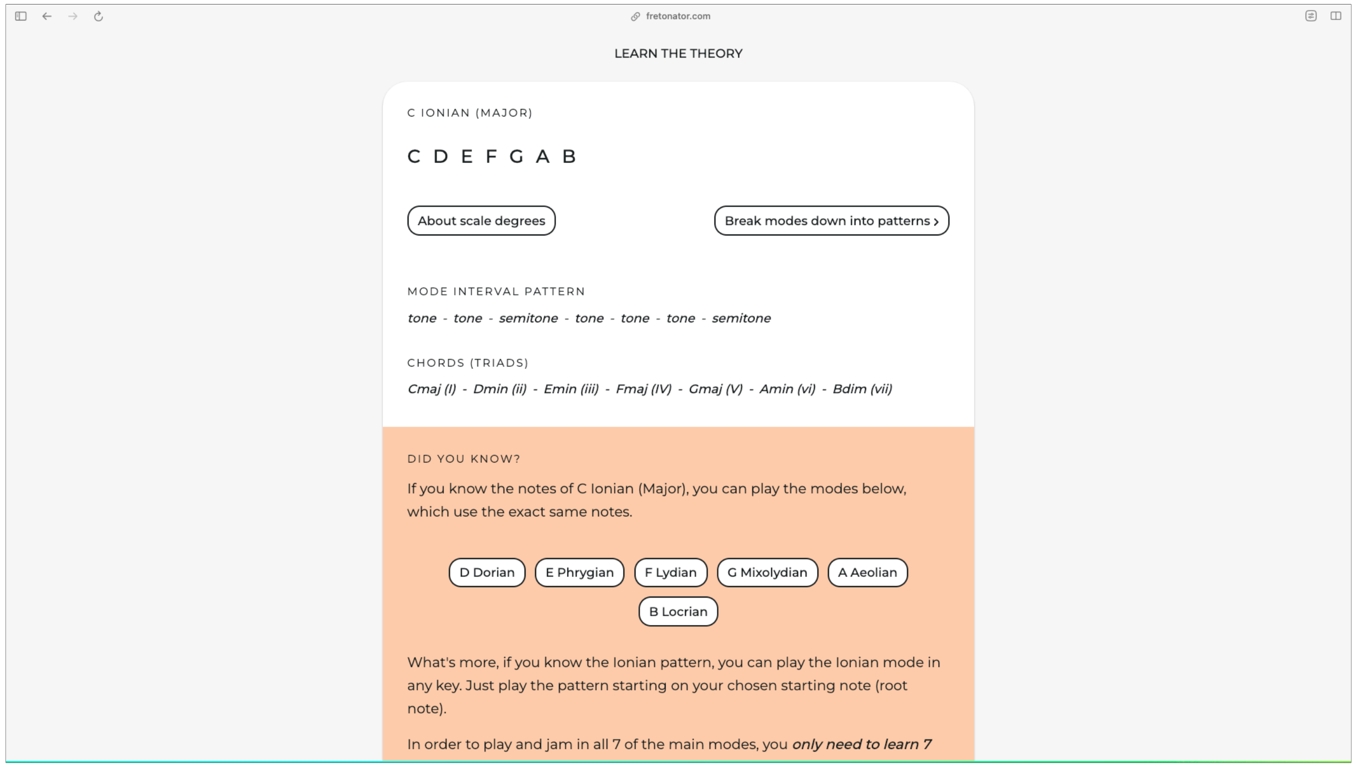
Major Pentatonic Minor Pentatonic

OR GET A RANDOM ONE

CHECK OUT THE FRETBOARD

E D C B A G F E D C B A G F E
B A G F E D C B A G F E D C B
G F E D C B A G F E D C B A G
D C B A G F E D C B A G F E D
A G F E D C B A G F E D C B A
E D C B A G F E D C B A G F E

Chose a starting note
Pick a mode/scale
Use the guitar fretboard to learn the scale



LEARN THE THEORY

C IONIAN (MAJOR)

C D E F G A B

MODE INTERVAL PATTERN

tone - tone - semitone - tone - tone - tone - semitone

CHORDS (TRIADS)

Cmaj (I) - Dmin (ii) - Emin (iii) - Fmaj (IV) - Gmaj (V) - Amin (vi) - Bdim (vii)

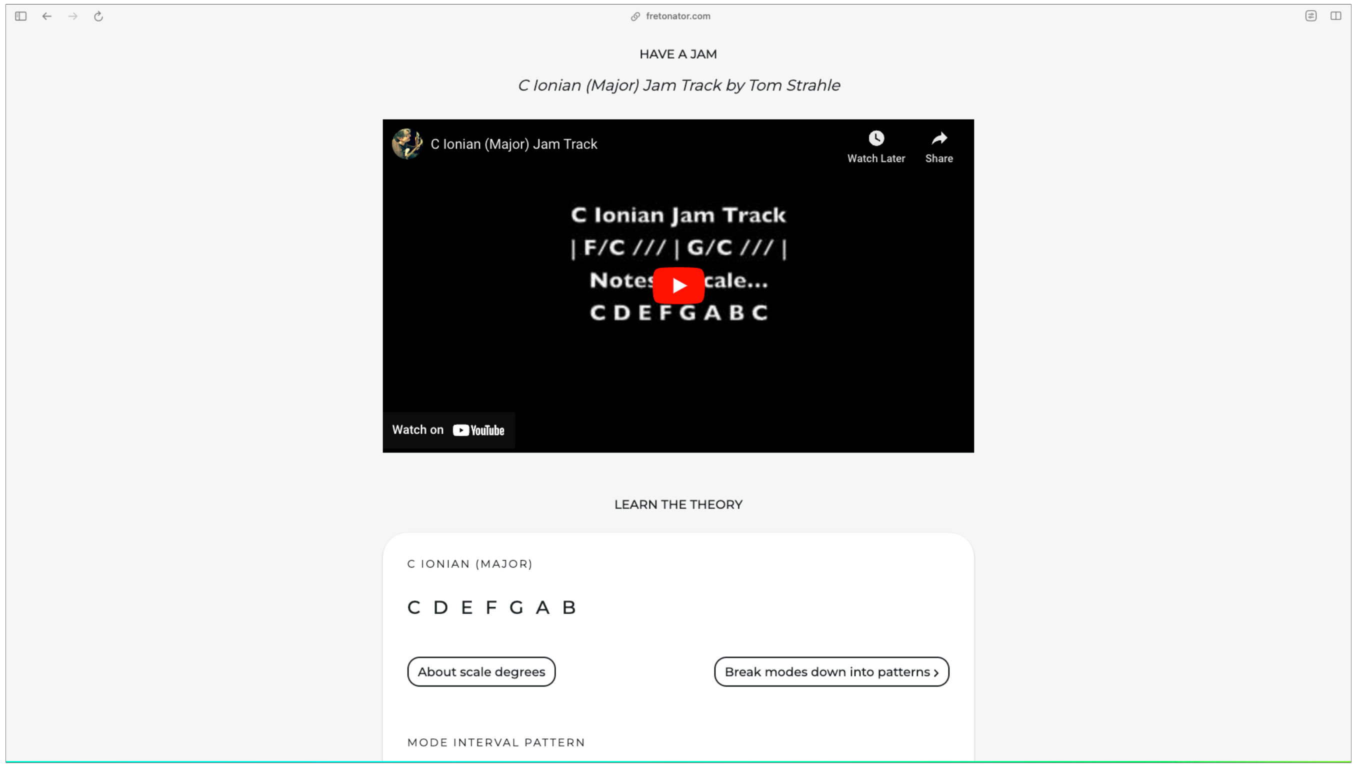
DID YOU KNOW?

If you know the notes of C Ionian (Major), you can play the modes below, which use the exact same notes.

What's more, if you know the Ionian pattern, you can play the Ionian mode in any key. Just play the pattern starting on your chosen starting note (root note).

In order to play and jam in all 7 of the main modes, you ***only need to learn 7***

You can dive deeper into the theory of modes and scales if that’s your thing



fretonator.com

HAVE A JAM

C Ionian (Major) Jam Track by Tom Strahle

LEARN THE THEORY

C IONIAN (MAJOR)

C D E F G A B

MODE INTERVAL PATTERN

But most importantly every scale for every note comes with a backing track for you to practise with
160 jam tracks for every note and scale combination were painstakingly sourced, for your entertainment

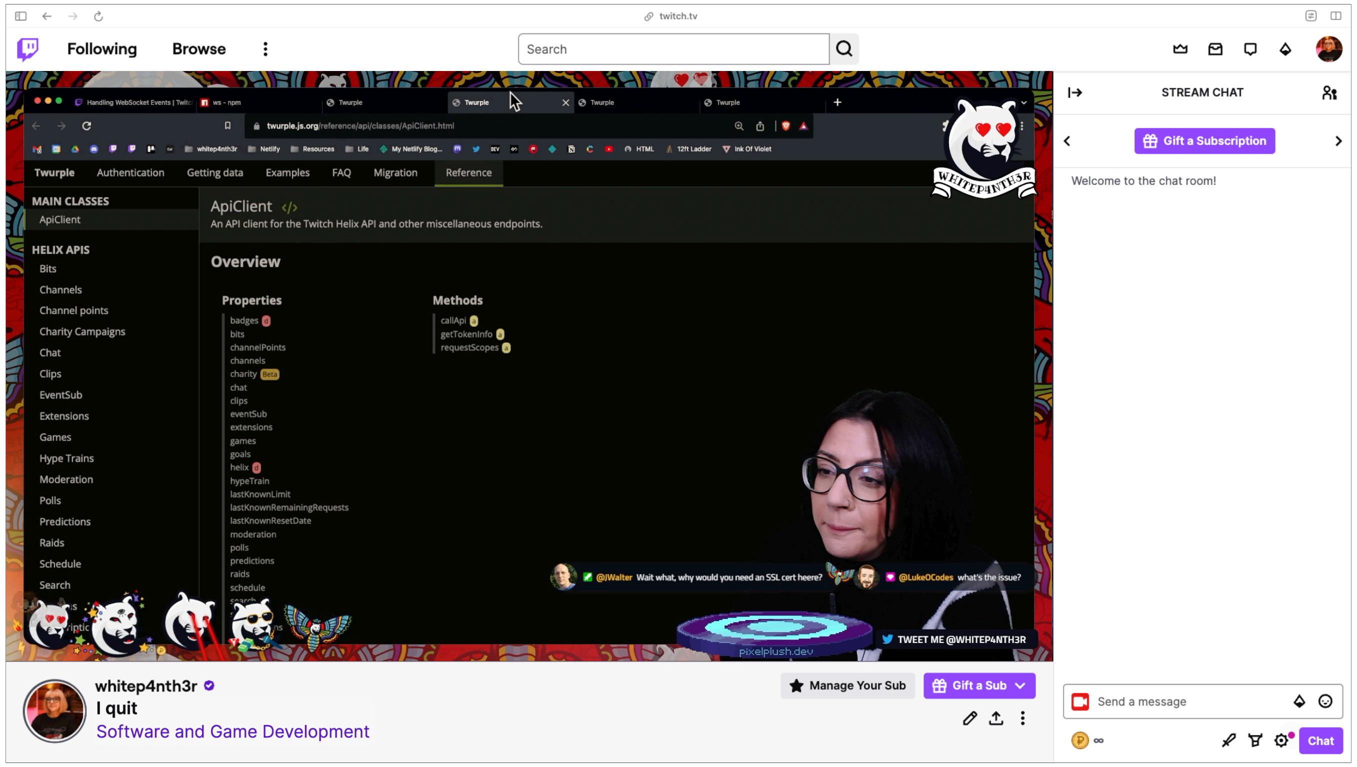
Rather than just playing the scale on your own, without any more musical context
The fretonator its about putting theory into practise in tangible, meaningful ways
Much like the mission of the school of rock and pop

And this concept of putting theory into practise, of making tangible things, has underpinned everything I do in tech and on my stream, for your entertainment

My original cheesy tagline… focus on building — which is representative of how I learn
Retired this tagline this year… Live laugh love… lol

Through live streaming on Twitch during the pandemic,
I was starting to discover that demonstrating and practically applying technical concepts — in front of a live audience — was making tech more accessible — more human, pretty much exactly like when you perform music with others

It was also like pair programming with 100 people
You know how camaraderie develops through tough experiences and especially through the inevitable PAIN and SUFFERING of software development? Like this?



@JWalter Wait what, why would you need an SSL cert heere?

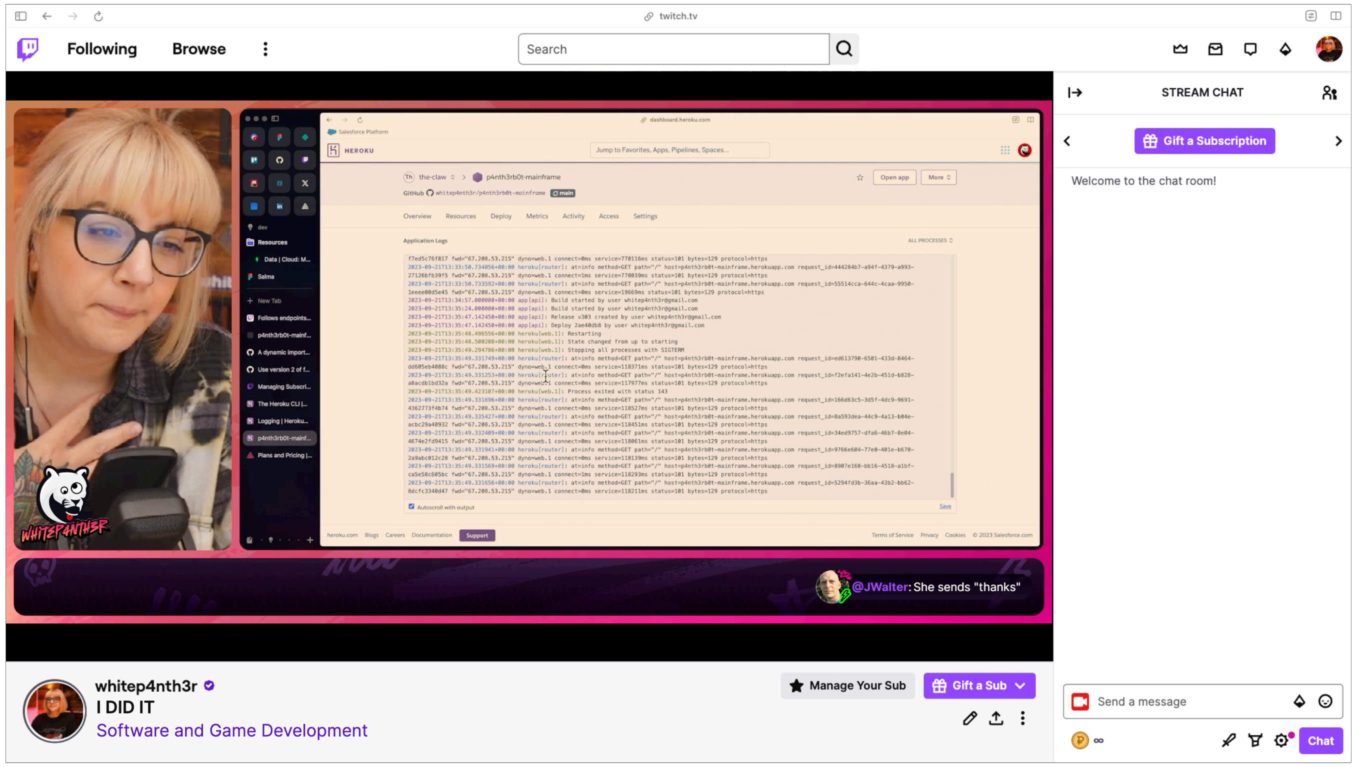
@LukeOCodes what's the issue?

I quit

Software and Game Development

[play]

And when you've problem-solved as a team, and your code finally works, it stimulates those good brain chemicals even more



```
f7ed5c76f817 fwd="67.208.53.215" dyno=web.1 connect=0ms service=770116ms status=101 bytes=129 protocol=https
2023-09-21T13:33:50.734056+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=444284b7-a94f-4379-a993-
27126bfb39f5 fwd="67.208.53.215" dyno=web.1 connect=1ms service=770039ms status=101 bytes=129 protocol=https
2023-09-21T13:33:50.733592+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=55514cca-644c-4caa-9950-
1eeee00d5e45 fwd="67.208.53.215" dyno=web.1 connect=0ms service=19669ms status=101 bytes=129 protocol=https
2023-09-21T13:34:57.000000+00:00 app[api]: Build started by user whitep4nth3r@gmail.com
2023-09-21T13:35:24.000000+00:00 app[api]: Build started by user whitep4nth3r@gmail.com
2023-09-21T13:35:47.142450+00:00 app[api]: Release v303 created by user whitep4nth3r@gmail.com
2023-09-21T13:35:47.142450+00:00 app[api]: Deploy 2ae40db8 by user whitep4nth3r@gmail.com
2023-09-21T13:35:48.496556+00:00 heroku[web.1]: Restarting
2023-09-21T13:35:48.500208+00:00 heroku[web.1]: State changed from up to starting
2023-09-21T13:35:49.294786+00:00 heroku[web.1]: Stopping all processes with SIGTERM
2023-09-21T13:35:49.331749+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=ed613790-6501-433d-8464-
dd605eb4088c fwd="67.208.53.215" dyno=web.1 connect=0ms service=118371ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.331253+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=f2efa141-4e2b-451d-b828-
a8acdb1bd32a fwd="67.208.53.215" dyno=web.1 connect=0ms service=117977ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.423107+00:00 heroku[web.1]: Process exited with status 143
2023-09-21T13:35:49.331696+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=166d63c5-3d5f-4dc9-9691-
4362773f4b74 fwd="67.208.53.215" dyno=web.1 connect=0ms service=118527ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.335427+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=8a593dea-44c9-4a13-b04e-
acbc29a40932 fwd="67.208.53.215" dyno=web.1 connect=0ms service=118451ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.332409+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=34ed9757-dfa6-46b7-8e04-
4674e2fd9415 fwd="67.208.53.215" dyno=web.1 connect=0ms service=118061ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.331941+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=9766e604-77e0-401e-b670-
2a9abc012c28 fwd="67.208.53.215" dyno=web.1 connect=0ms service=118139ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.331569+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=8907e168-bb16-4518-a1bf-
ca5e58c605bc fwd="67.208.53.215" dyno=web.1 connect=1ms service=118293ms status=101 bytes=129 protocol=https
2023-09-21T13:35:49.331656+00:00 heroku[router]: at=info method=GET path="/" host=p4nth3rb0t-mainframe.herokuapp.com request_id=5294fd3b-36aa-43b2-bb62-
8dcfc3340d47 fwd="67.208.53.215" dyno=web.1 connect=0ms service=118211ms status=101 bytes=129 protocol=https
```

@JWalter: She sends "thanks"

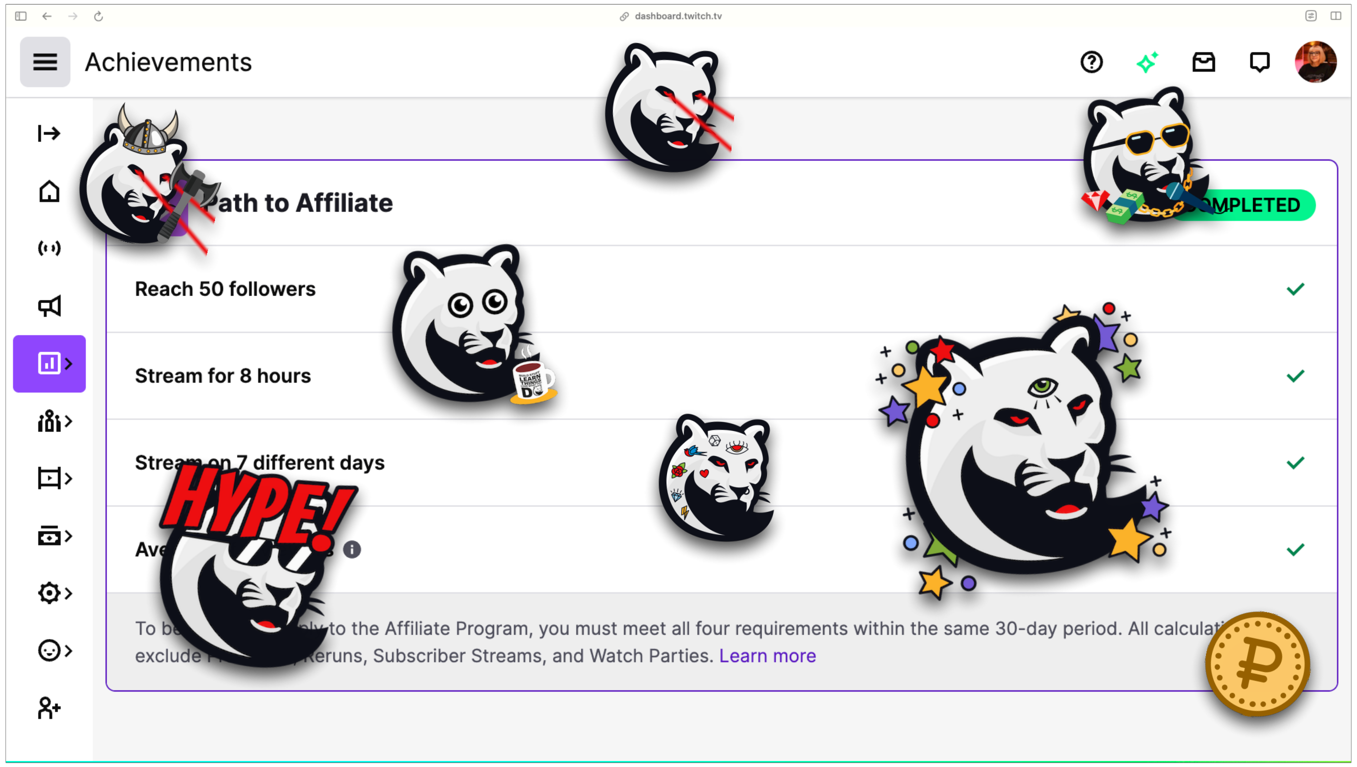
Welcome to the chat room!

whitep4nth3r
I DID IT
Software and Game Development

So as I was working out what this whole streaming thing was about

People were helping me, returning to the stream to see progress, pair programming with me, wanting to see me succeed

Without even setting out to do this, there was a community forming around me and my stream



Achievements

ath to Affiliate

MPLETED

Reach 50 followers ✓

Stream for 8 hours ✓

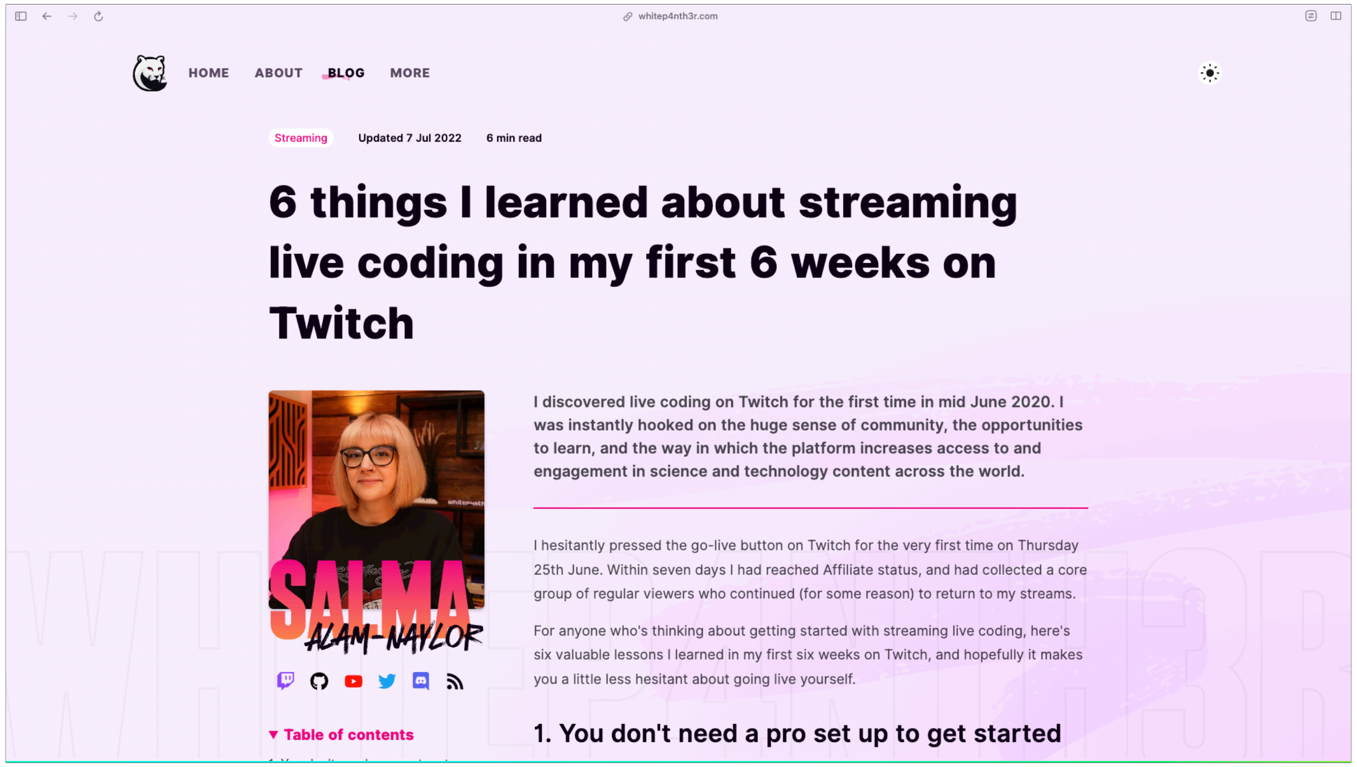
Stream on 7 different days ✓

Ave ✓

To be ly to the Affiliate Program, you must meet all four requirements within the same 30-day period. All calculati exclude eruns, Subscriber Streams, and Watch Parties. Learn more

And in just 7 days, I streamed enough and had enough average concurrent viewers to gain affiliate status on Twitch
Which meant I could make use of community engagement tools such as

[click] Custom emotes (there are those panthers), VIP status for viewers, channel point redemptions, and people could throw Twitch currency at me in the form of subs and cheers



Streaming Updated 7 Jul 2022 6 min read

6 things I learned about streaming live coding in my first 6 weeks on Twitch

I discovered live coding on Twitch for the first time in mid June 2020. I was instantly hooked on the huge sense of community, the opportunities to learn, and the way in which the platform increases access to and engagement in science and technology content across the world.

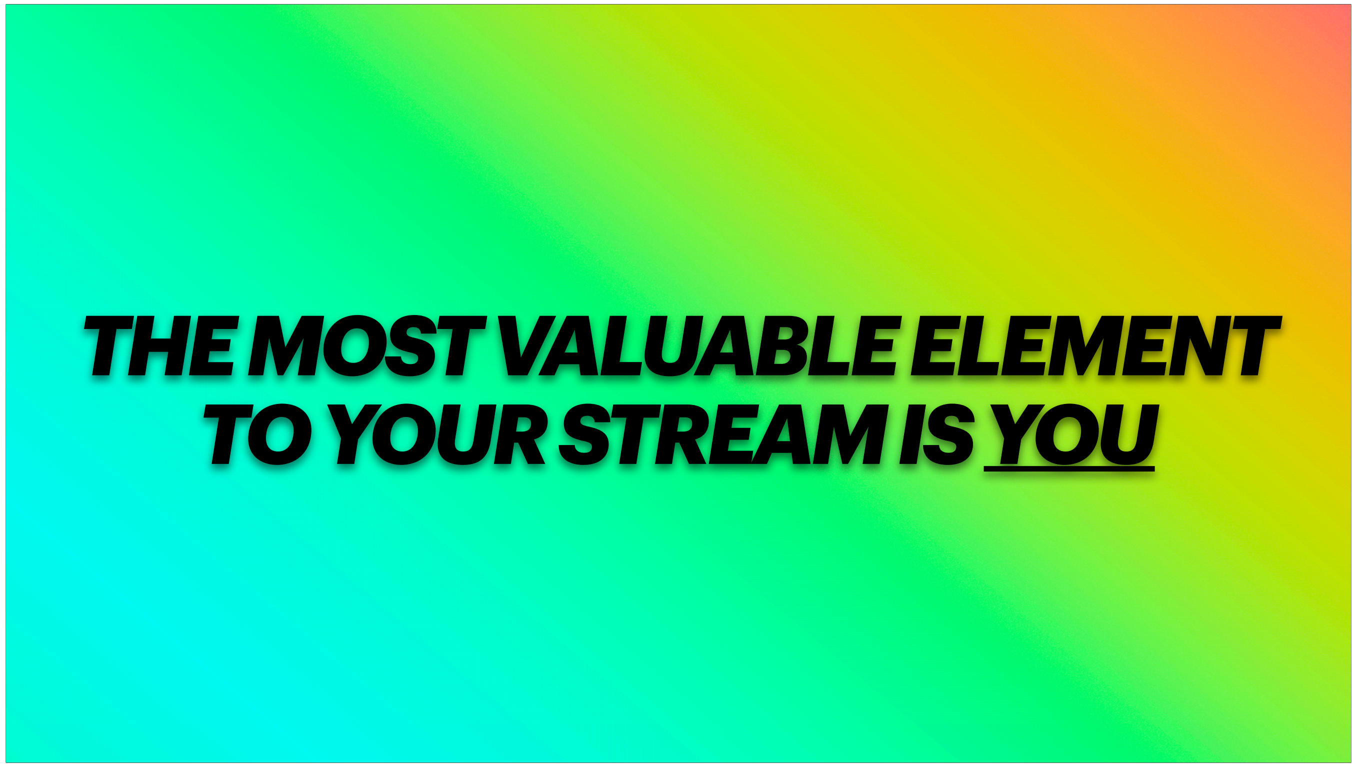
I hesitantly pressed the go-live button on Twitch for the very first time on Thursday 25th June. Within seven days I had reached Affiliate status, and had collected a core group of regular viewers who continued (for some reason) to return to my streams.

For anyone who's thinking about getting started with streaming live coding, here's six valuable lessons I learned in my first six weeks on Twitch, and hopefully it makes you a little less hesitant about going live yourself.

▼ Table of contents

1. You don't need a pro set up to get started

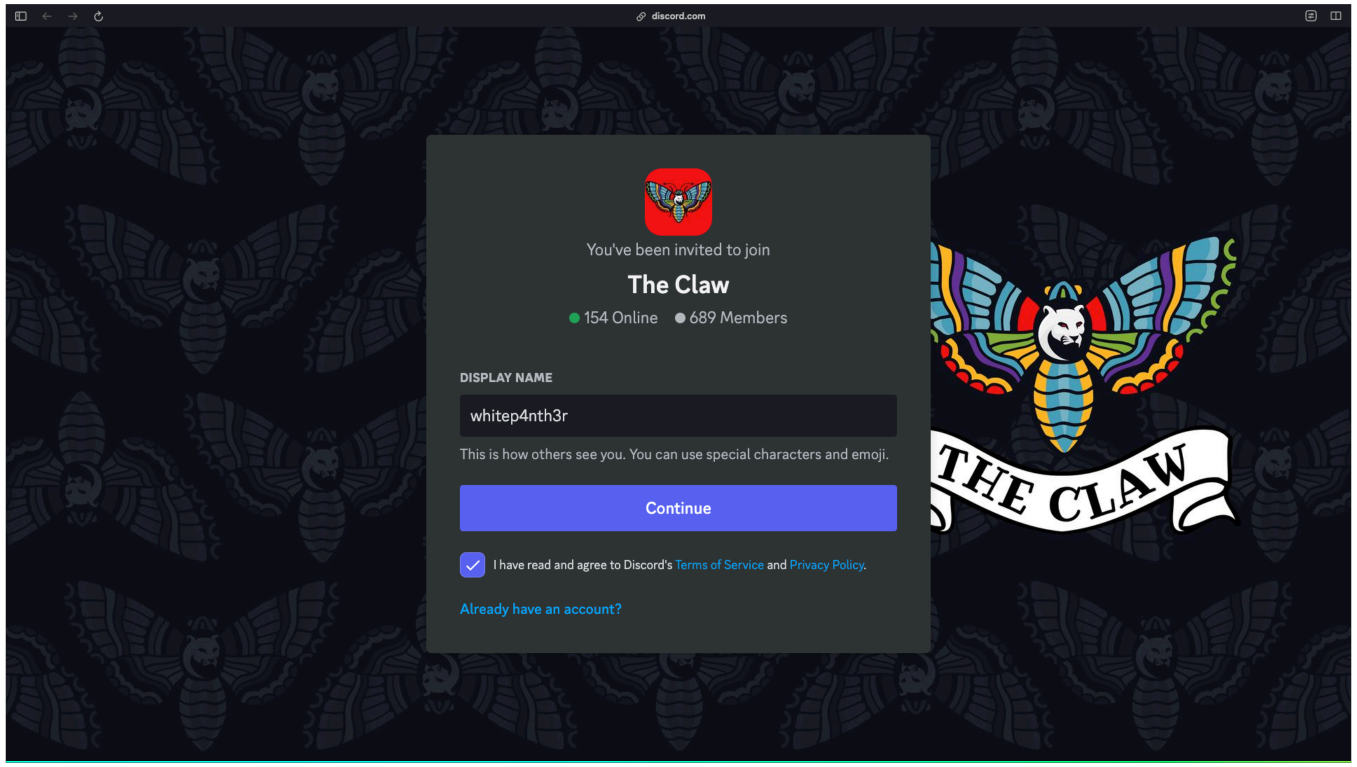
And I wrote about what I learned in those early days on my blog
And the most important thing I took away from this



THE MOST VALUABLE ELEMENT
TO YOUR STREAM IS YOU

The most valuable element to your stream is you

The code you are writing is only a small part of the experience. Viewers don’t actually watch you to learn how to code, that’s often a side effect — they come to hang out with you, to feel safe in the presence of people with similar interests, to be entertained.



You've been invited to join

The Claw

● 154 Online ● 689 Members

DISPLAY NAME

whitep4nth3r

This is how others see you. You can use special characters and emoji.

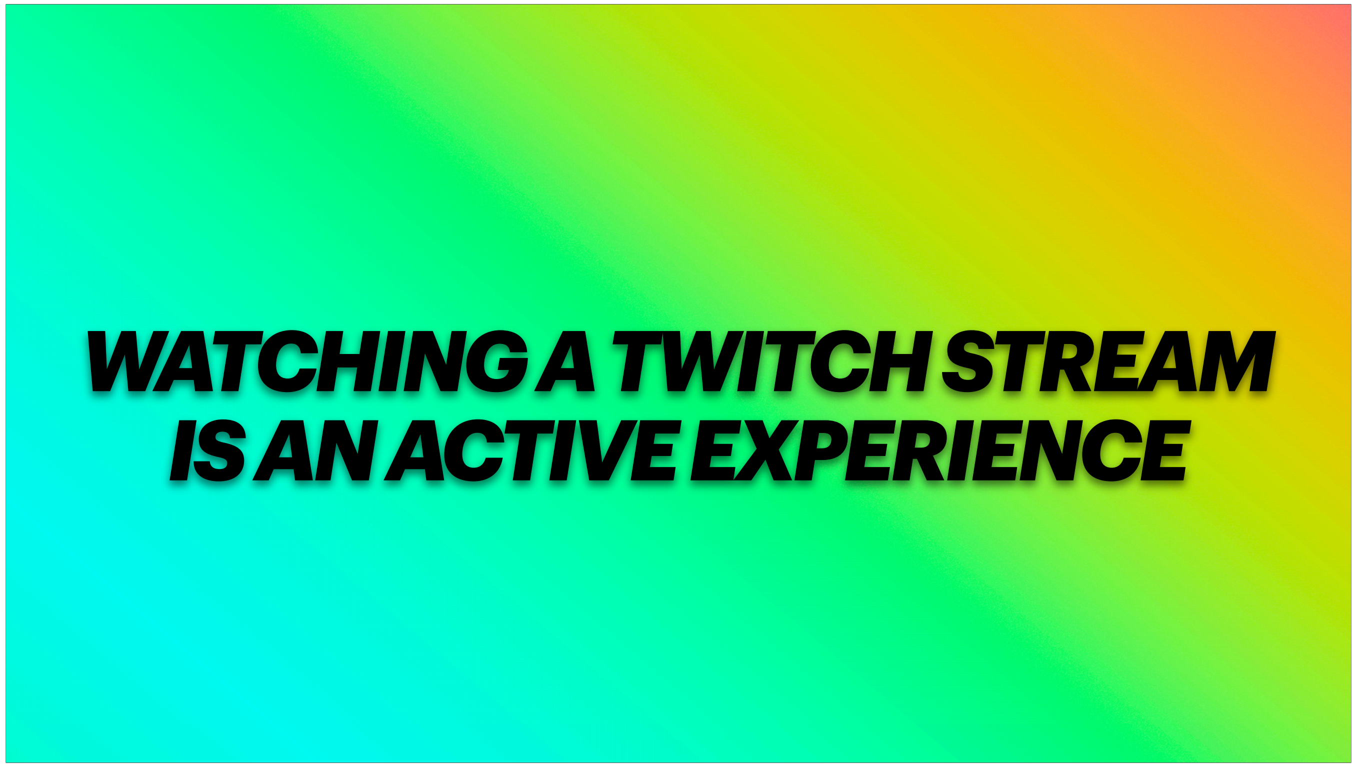
☑ I have read and agree to Discord's Terms of Service and Privacy Policy.

And to support the community that was forming…

I created the claw discord on 30th July 2020 (group of panthers lol)

People gathered here when I was offline, relationships formed even more

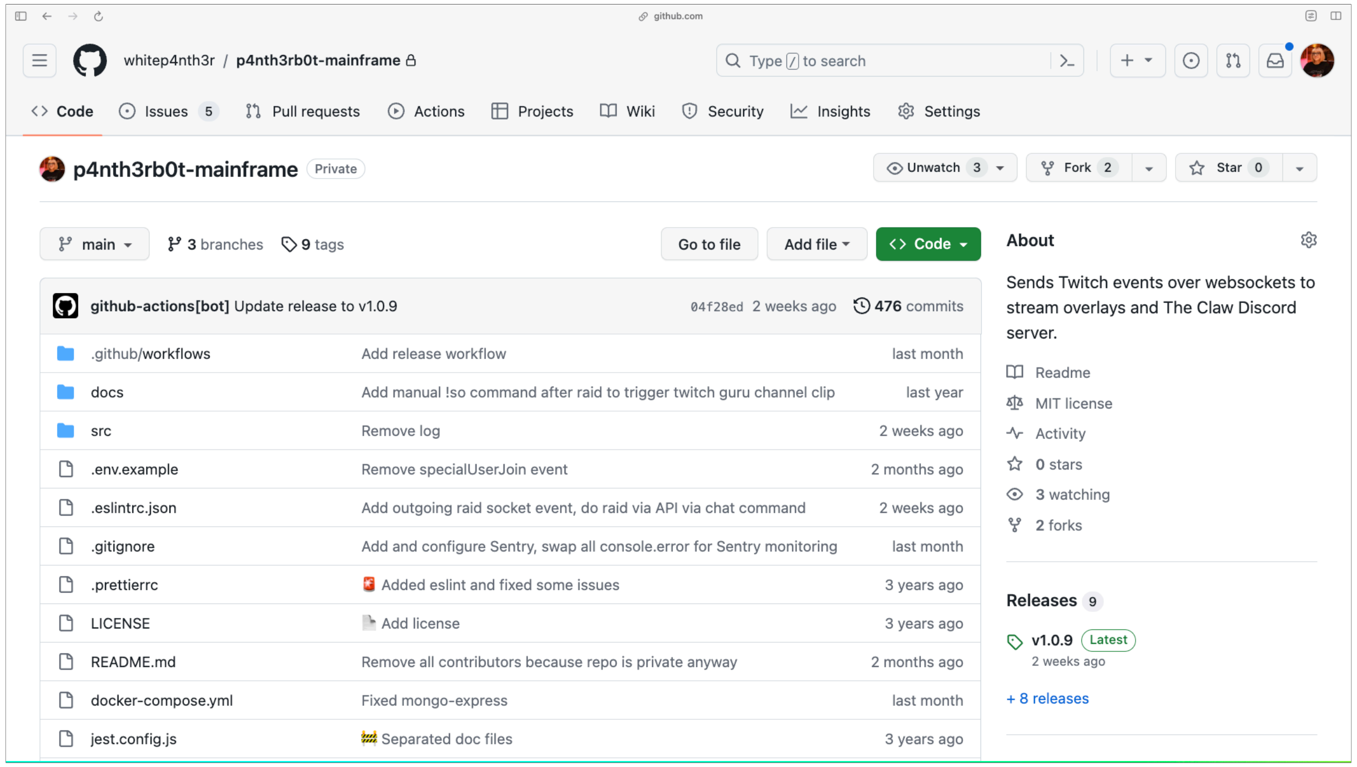
Co-working channel



WATCHING A TWITCH STREAM
IS AN ACTIVE EXPERIENCE

Twitch viewers are often not passive (like YouTube viewers), they’re active participants in the experience

So I started thinking about how could I build on this
And how I could bring people further into my stream to make them feel even more involved with what they were experiencing



p4nth3rb0t-mainframe

github-actions[bot] Update release to v1.0.9 04f28ed 2 weeks ago 476 commits

.github/workflows	Add release workflow	last month
docs	Add manual !so command after raid to trigger twitch guru channel clip	last year
src	Remove log	2 weeks ago
.env.example	Remove specialUserJoin event	2 months ago
.eslintrc.json	Add outgoing raid socket event, do raid via API via chat command	2 weeks ago
.gitignore	Add and configure Sentry, swap all console.error for Sentry monitoring	last month
.prettierrc	🚨 Added eslint and fixed some issues	3 years ago
LICENSE	📄 Add license	3 years ago
README.md	Remove all contributors because repo is private anyway	2 months ago
docker-compose.yml	Fixed mongo-express	last month
jest.config.js	🚧 Separated doc files	3 years ago

About

Sends Twitch events over websockets to stream overlays and The Claw Discord server.

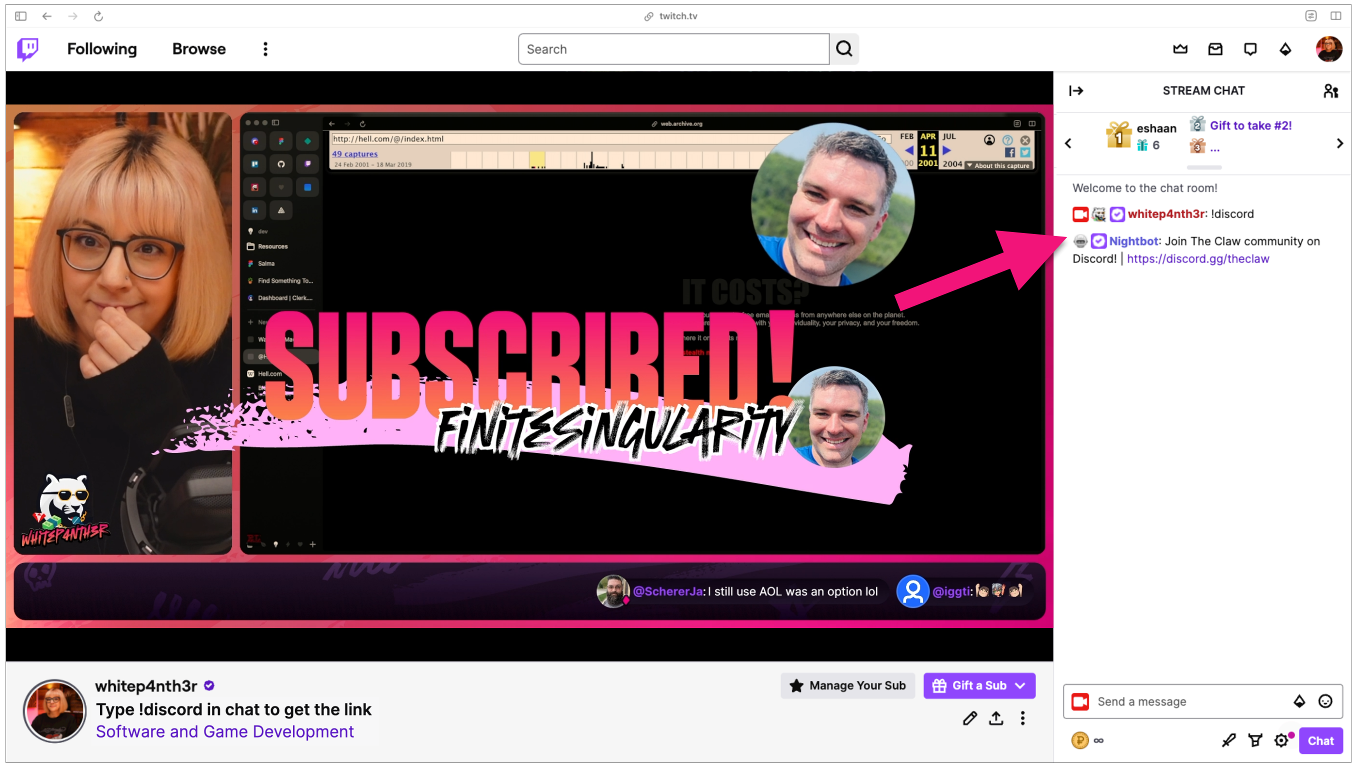
- Readme
- MIT license
- Activity
- 0 stars
- 3 watching
- 2 forks

Releases 9

v1.0.9 Latest
2 weeks ago

+ 8 releases

So I built a twitch bot
Application built using the Twitch API and other third party tools
To allow viewers to make things happen on screen via chat commands and channel point redemptions
To level-up their participation in the stream



eshaan 6 · Gift to take #2! · ...

Welcome to the chat room!

whitep4nth3r: !discord

Nightbot: Join The Claw community on Discord! | https://discord.gg/theclaw

@SchererJa: I still use AOL was an option lol

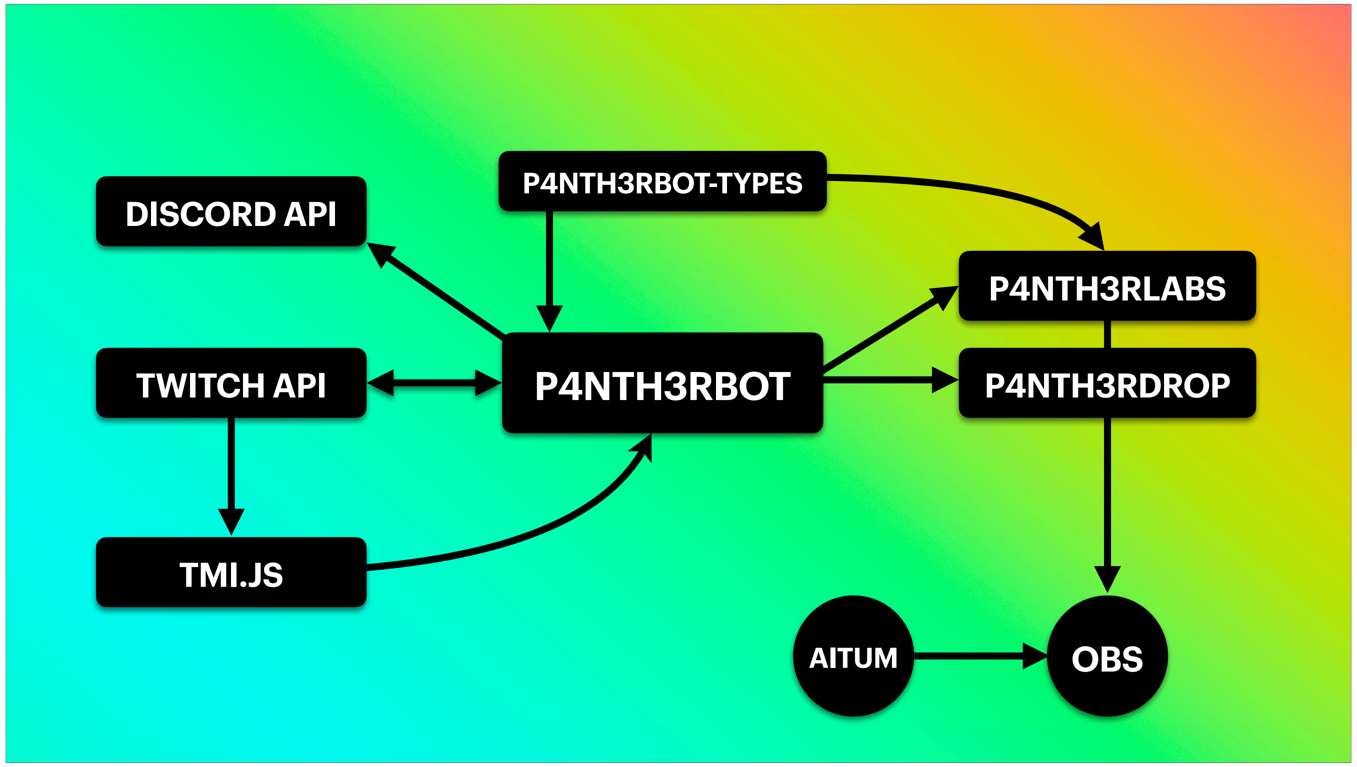
@iggti:

whitep4nth3r
Type !discord in chat to get the link
Software and Game Development

There are pre-existing tools to bring interactivity to your streams if you can't build your own, such as Nightbot

And making a Twitch bot is not a new innovative thing, lots of programming streamers have done this
But here I had the opportunity to build a twitch bot that reflected my personality, and the entertainment value of the stream I wanted to build on

Plus, building stuff for your stream, live on stream, with viewers, for viewers, is the best way to test functionality, crowdsource ideas, and again, make people feel part of the process, part of the product, and part of the stream



Here’s some probably very over-engineered architecture for your entertainment
This is everything I use to power my stream interactions

Express app, TypeScript, MongoDB, sends events over a web socket connection, it’s the backend
[click] Uses the Twitch API and a third party wrapper around the twitch api called tmi.js [click]
2 React apps, listening for web socket events
[click] There’s a types package that both the backend and frontend applications use
[click] It also uses the Discord API
[click] And all of this is brought together in OBS — open broadcaster software — which I use to stream, where each piece of functionality is added to my stream scenes as browser source URLs
I also use Aitum

Show you some of the key functionality of my twitch bot
Used to be open source… but got too bespoke, too silly, it’s private now
But I do have some core maintainers in my community who help out

ALERTS
(AND OTHER NONSENSE)

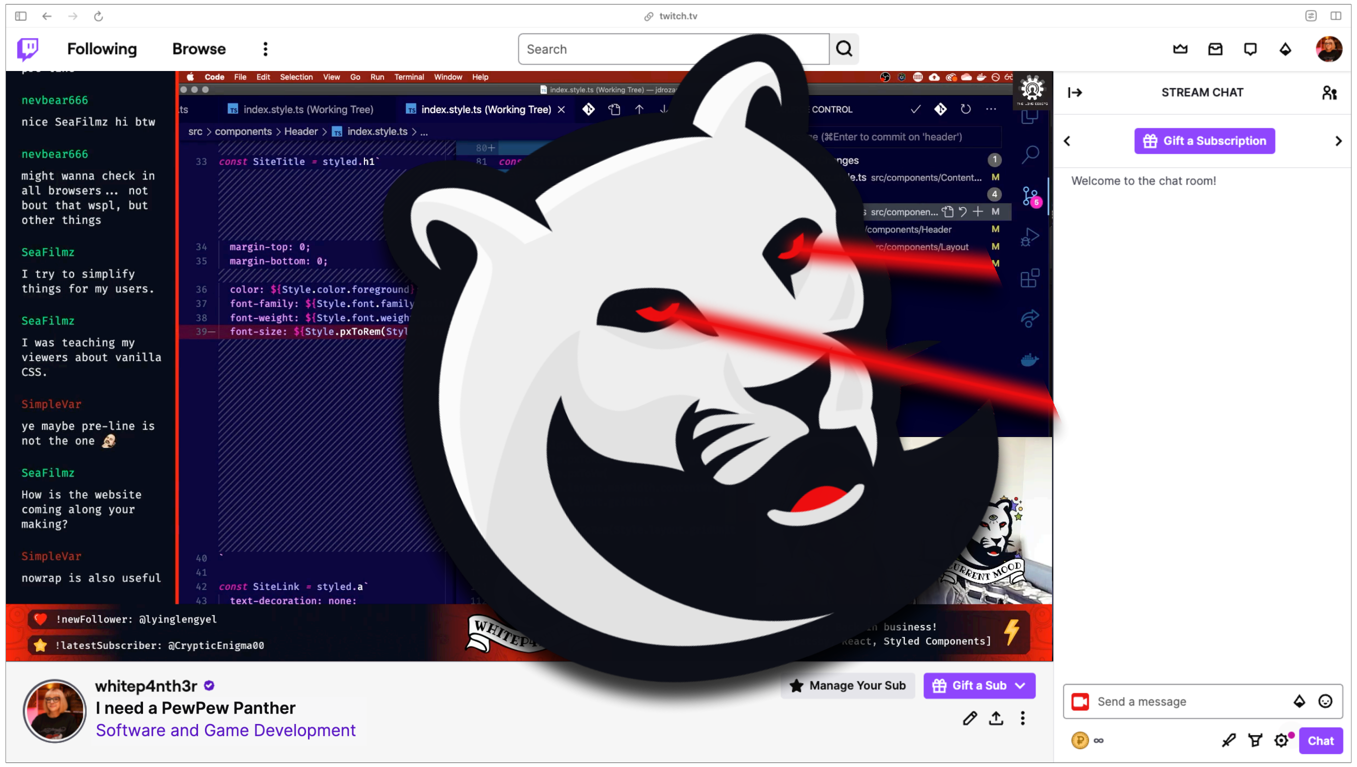
Follows, subscriptions, cheering — core parts of the Twitch experience

As a viewer myself, I like to sneak into streams and follow streamers just to see how much I become part of the stream when I press that follow button

If nothing happens, and the streamer doesn't notice me, I'll usually bounce right out

Because making your viewers feel welcome and part of something is key to getting them to stick around, especially in this smaller, more intimate category of software and game development

Here's how my alerts started out



nevbear666
nice SeaFilmz hi btw

nevbear666
might wanna check in all browsers ... not bout that wspl, but other things

SeaFilmz
I try to simplify things for my users.

SeaFilmz
I was teaching my viewers about vanilla CSS.

SimpleVar
ye maybe pre-line is not the one

SeaFilmz
How is the website coming along your making?

SimpleVar
nowrap is also useful

!newFollower: @lyinglengyel
!latestSubscriber: @CrypticEnigma00

whitep4nth3r
I need a PewPew Panther
Software and Game Development

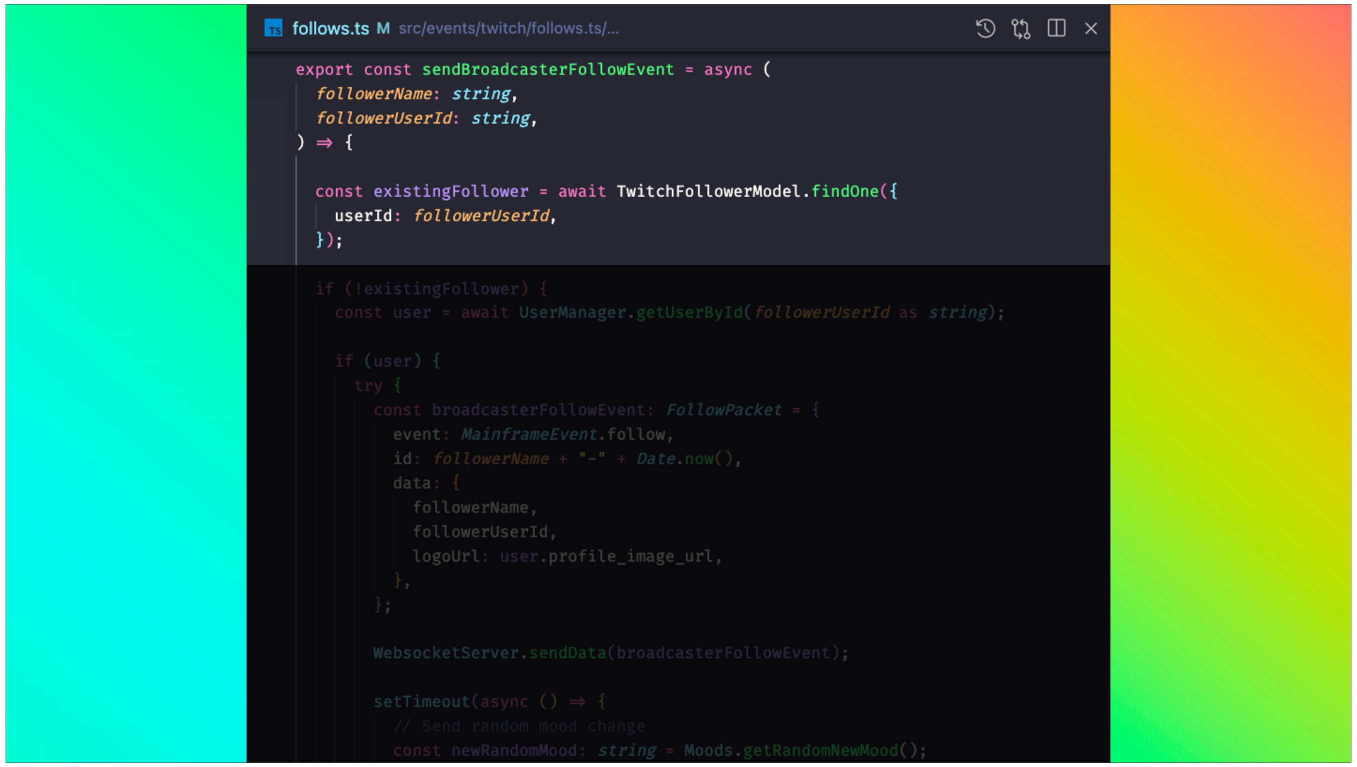
Welcome to the chat room!

Using Streamlabs alerts — another free streamer tool, added as a browser source to OBS.

And I did make a pew pew panther. [click] I think it made pewWTD feel very special.

My bot uses the EventSub API from twitch, which lets your application subscribe to events that happen on Twitch.

When an event occurs for one of my subscriptions, such as a follow, Twitch sends my app a notification.

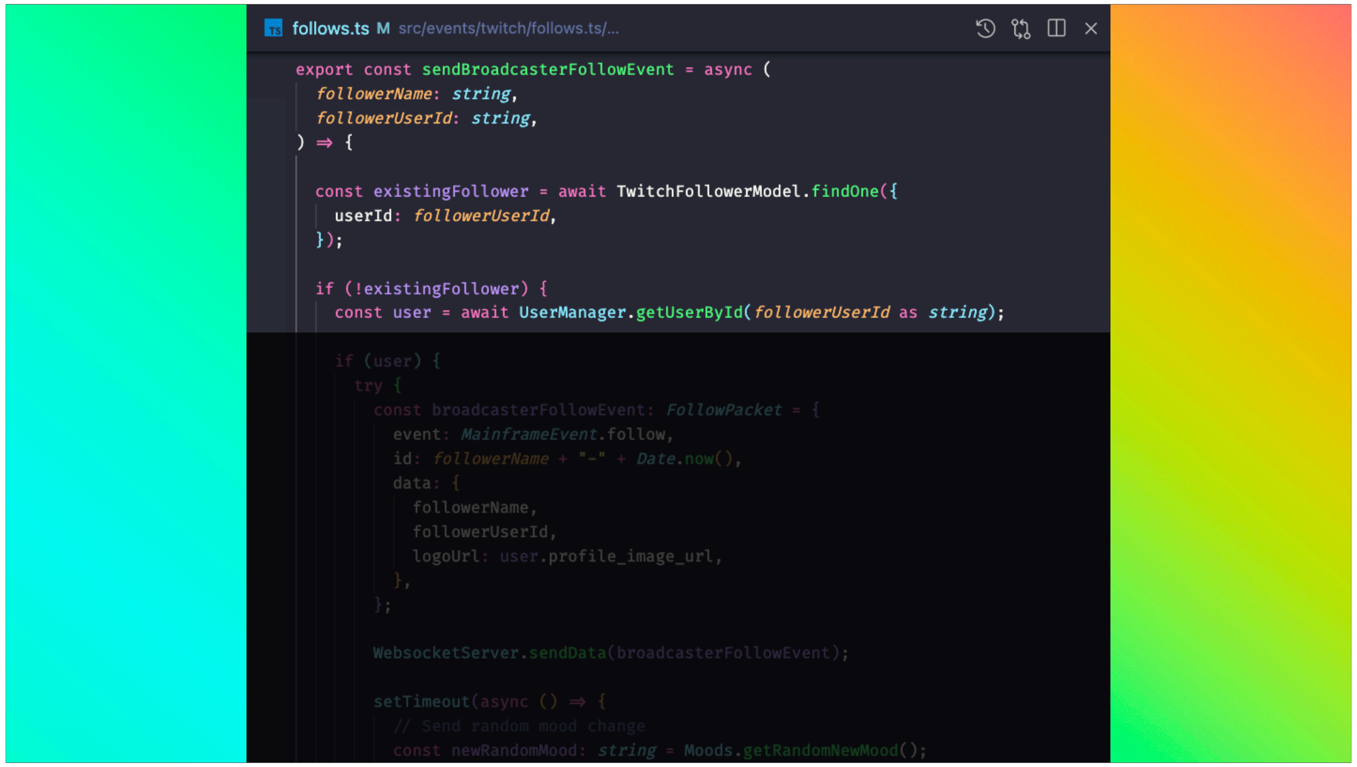


```
export const sendBroadcasterFollowEvent = async (
  followerName: string,
  followerUserId: string,
) ⇒ {

  const existingFollower = await TwitchFollowerModel.findOne({
    userId: followerUserId,
  });

  if (!existingFollower) {
    const user = await UserManager.getUserById(followerUserId as string);

    if (user) {
      try {
        const broadcasterFollowEvent: FollowPacket = {
          event: MainframeEvent.follow,
          id: followerName + "-" + Date.now(),
          data: {
            followerName,
            followerUserId,
            logoUrl: user.profile_image_url,
          },
        };

        WebsocketServer.sendData(broadcasterFollowEvent);

        setTimeout(async () ⇒ {
          // Send random mood change
          const newRandomMood: string = Moods.getRandomNewMood();
```

Here’s what happens when I receive a follow event.

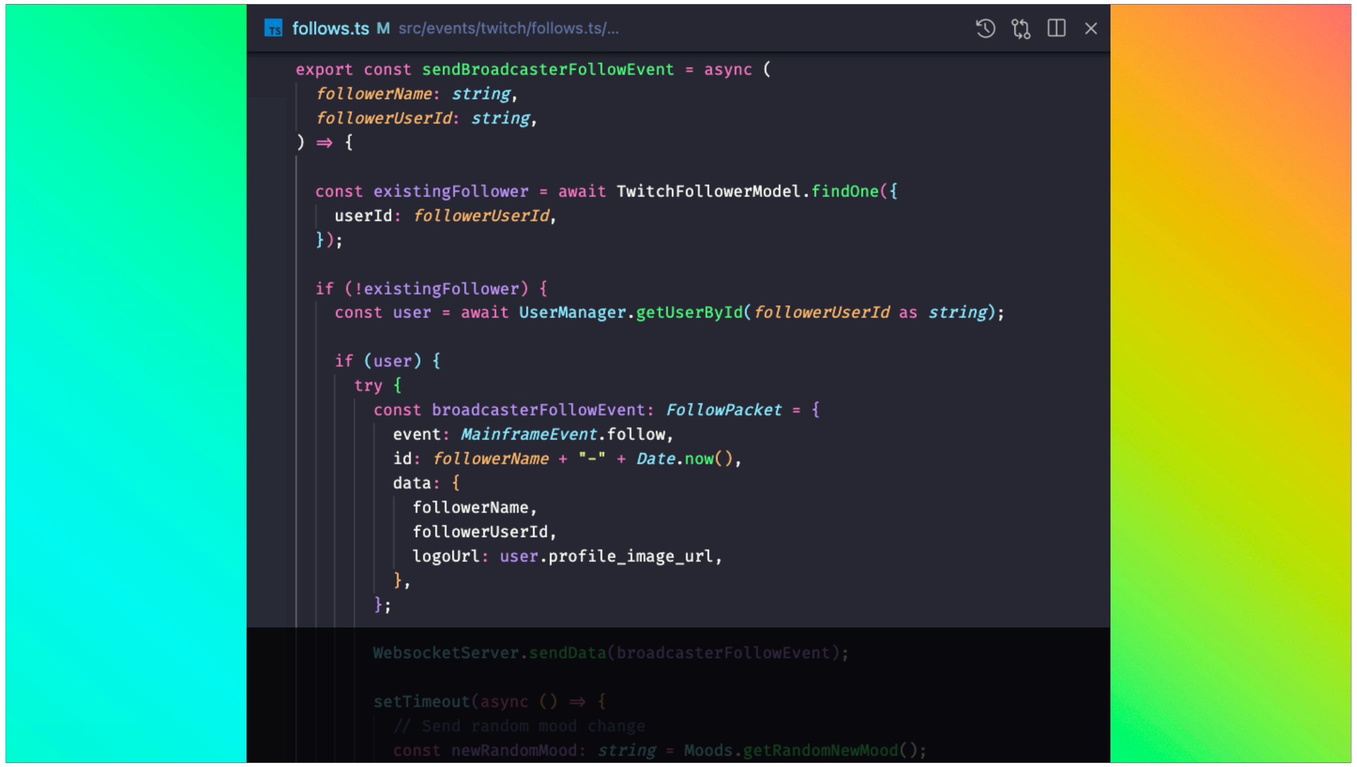
In order to protect against viewers spamming the alerts on my stream, we first check for an existing follower in the database.

If we don’t find one,



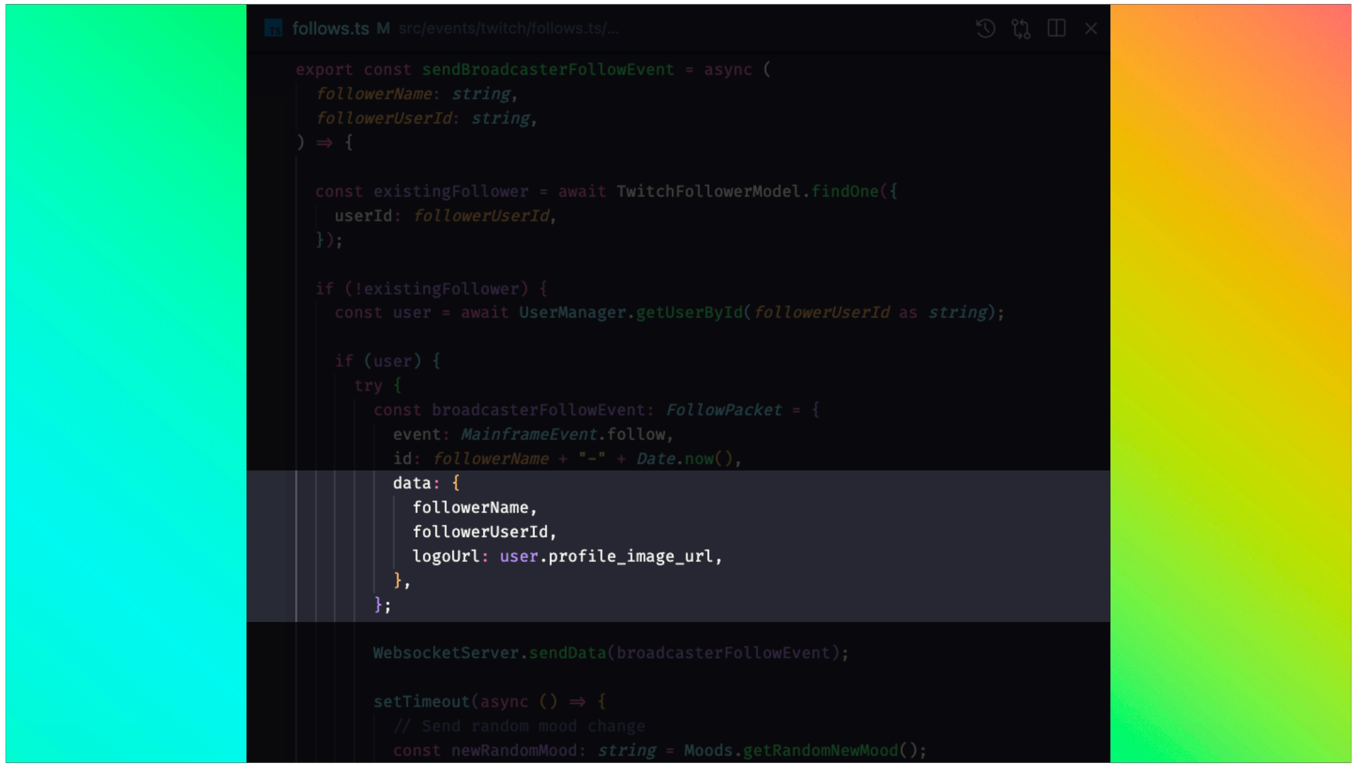
```typescript
export const sendBroadcasterFollowEvent = async (
  followerName: string,
  followerUserId: string,
) ⇒ {

  const existingFollower = await TwitchFollowerModel.findOne({
    userId: followerUserId,
  });

  if (!existingFollower) {
    const user = await UserManager.getUserById(followerUserId as string);

    if (user) {
      try {
        const broadcasterFollowEvent: FollowPacket = {
          event: MainframeEvent.follow,
          id: followerName + "-" + Date.now(),
          data: {
            followerName,
            followerUserId,
            logoUrl: user.profile_image_url,
          },
        };

        WebsocketServer.sendData(broadcasterFollowEvent);

        setTimeout(async () ⇒ {
          // Send random mood change
          const newRandomMood: string = Moods.getRandomNewMood();
```

We find the Twitch user



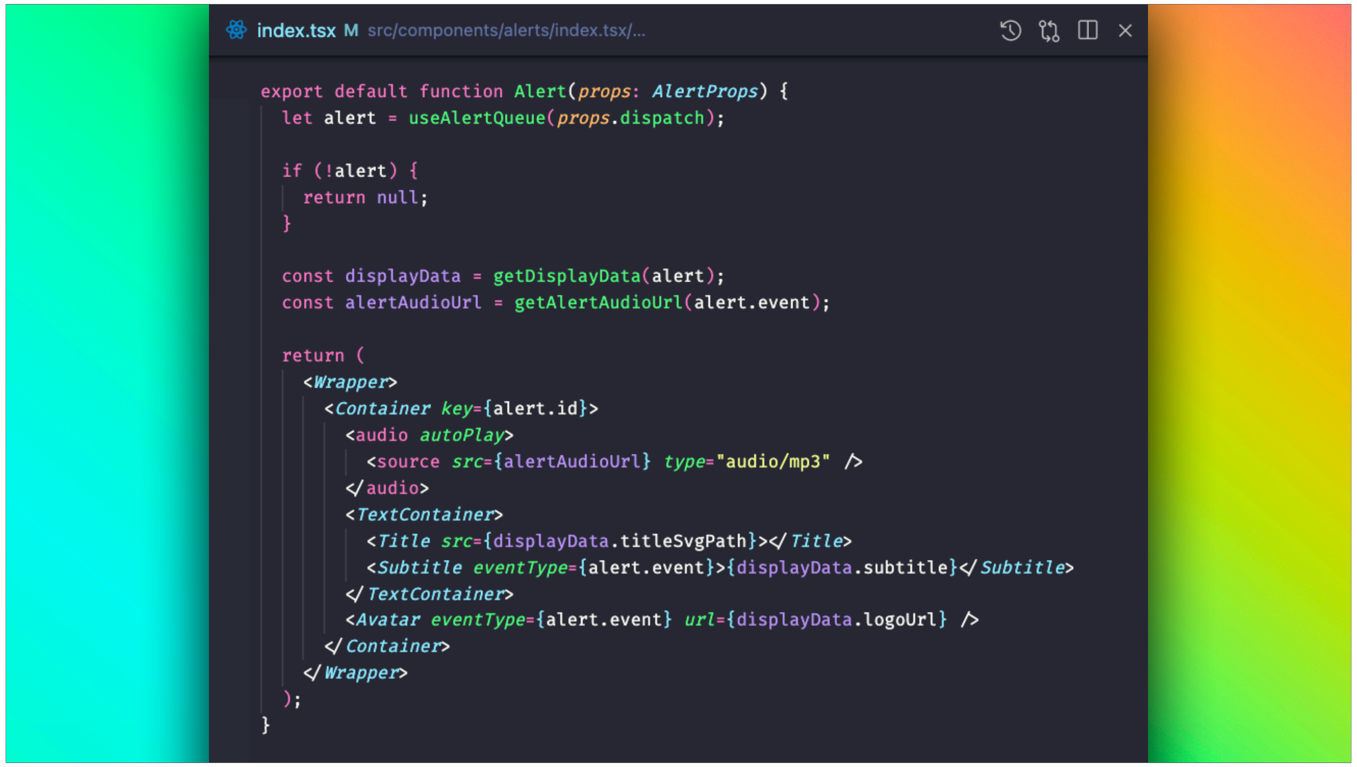
```
export const sendBroadcasterFollowEvent = async (
  followerName: string,
  followerUserId: string,
) ⇒ {

  const existingFollower = await TwitchFollowerModel.findOne({
    userId: followerUserId,
  });

  if (!existingFollower) {
    const user = await UserManager.getUserById(followerUserId as string);

    if (user) {
      try {
        const broadcasterFollowEvent: FollowPacket = {
          event: MainframeEvent.follow,
          id: followerName + "-" + Date.now(),
          data: {
            followerName,
            followerUserId,
            logoUrl: user.profile_image_url,
          },
        };

        WebsocketServer.sendData(broadcasterFollowEvent);

        setTimeout(async () ⇒ {
          // Send random mood change
          const newRandomMood: string = Moods.getRandomNewMood();
```

And send a follow event over the web socket connection



```
export const sendBroadcasterFollowEvent = async (
  followerName: string,
  followerUserId: string,
) => {

  const existingFollower = await TwitchFollowerModel.findOne({
    userId: followerUserId,
  });

  if (!existingFollower) {
    const user = await UserManager.getUserById(followerUserId as string);

    if (user) {
      try {
        const broadcasterFollowEvent: FollowPacket = {
          event: MainframeEvent.follow,
          id: followerName + "-" + Date.now(),
          data: {
            followerName,
            followerUserId,
            logoUrl: user.profile_image_url,
          },
        };

        WebsocketServer.sendData(broadcasterFollowEvent);

        setTimeout(async () => {
          // Send random mood change
          const newRandomMood: string = Moods.getRandomNewMood();
```

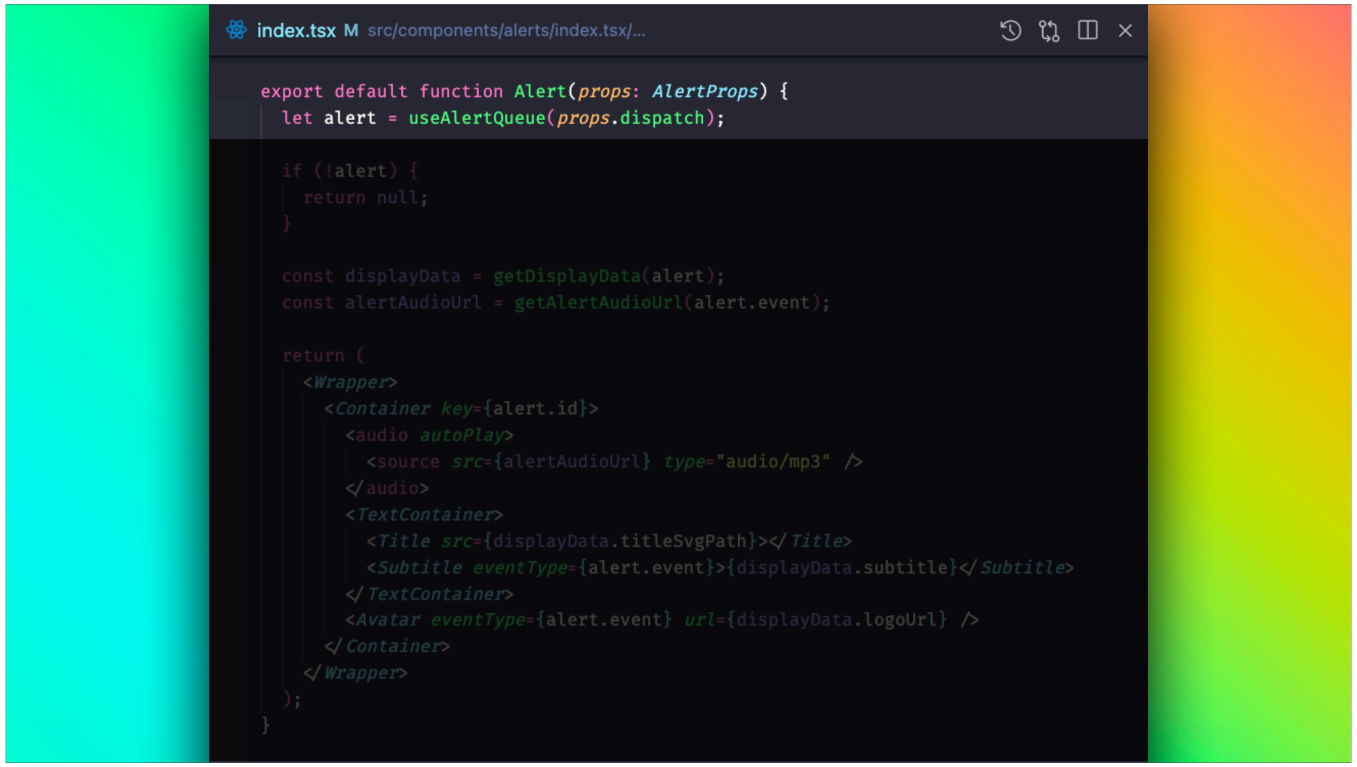
Most importantly the event contains the follower name and their profile image
Which is how we’ll put the viewer into the stream



```tsx
export default function Alert(props: AlertProps) {
  let alert = useAlertQueue(props.dispatch);

  if (!alert) {
    return null;
  }

  const displayData = getDisplayData(alert);
  const alertAudioUrl = getAlertAudioUrl(alert.event);

  return (
    <Wrapper>
      <Container key={alert.id}>
        <audio autoPlay>
          <source src={alertAudioUrl} type="audio/mp3" />
        </audio>
        <TextContainer>
          <Title src={displayData.titleSvgPath}></Title>
          <Subtitle eventType={alert.event}>{displayData.subtitle}</Subtitle>
        </TextContainer>
        <Avatar eventType={alert.event} url={displayData.logoUrl} />
      </Container>
    </Wrapper>
  );
}
```

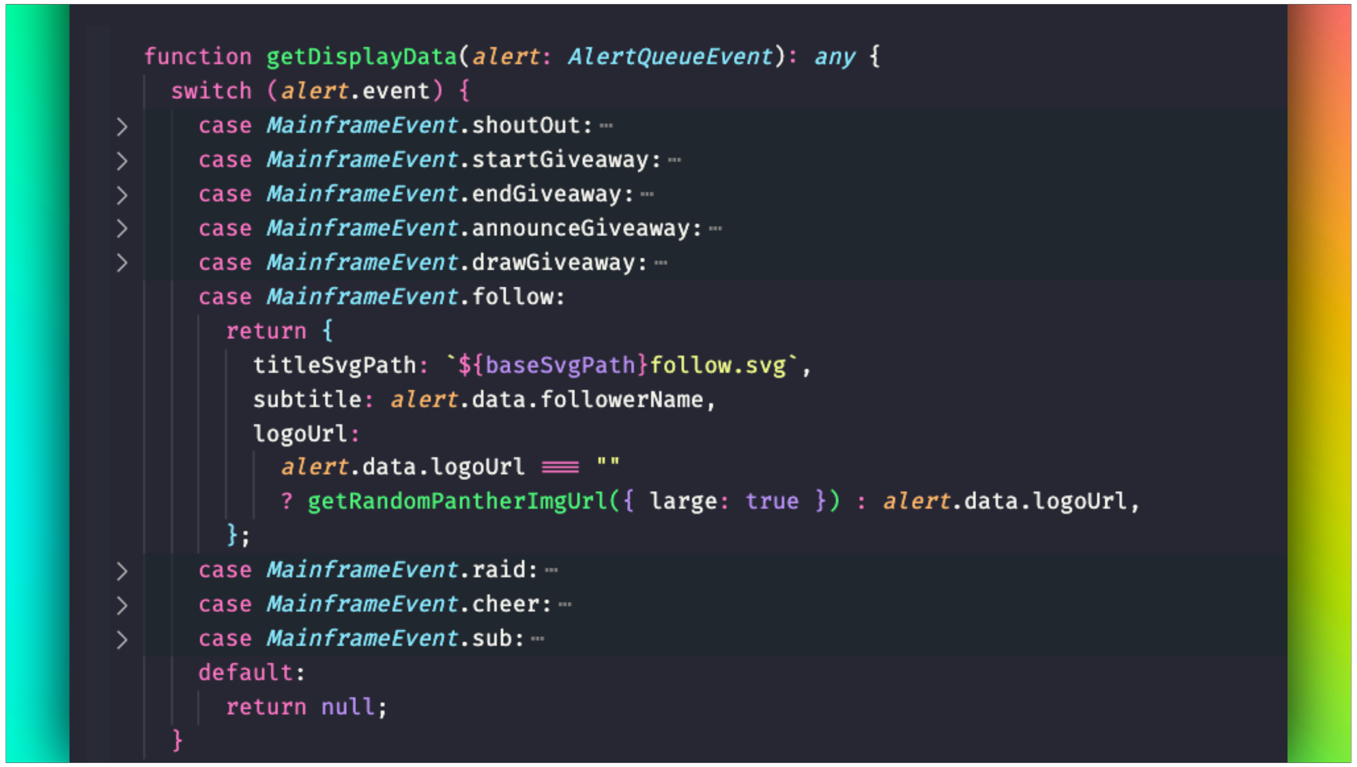
I have a number of front ends URLs which are listening for web socket events from the backend.

Here is a little React component that powers all the alerts



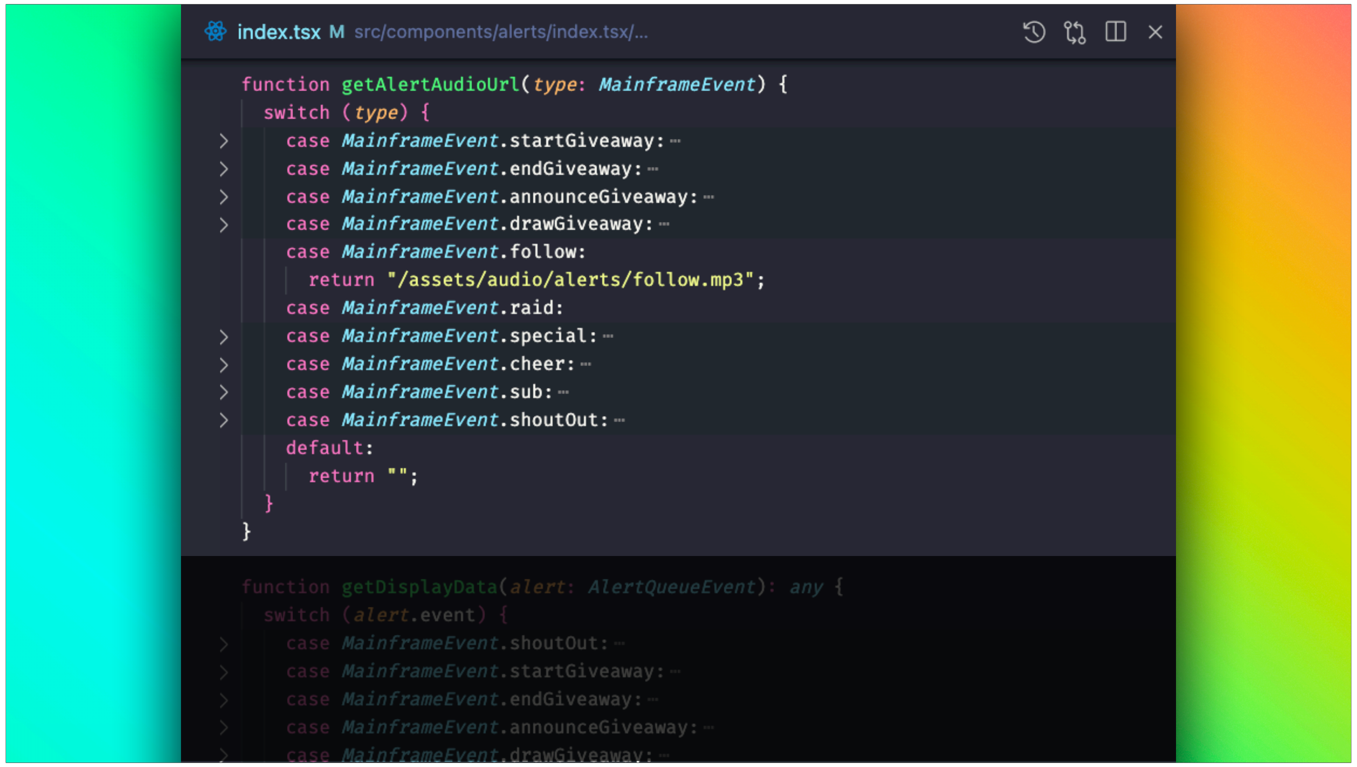
```tsx
export default function Alert(props: AlertProps) {
  let alert = useAlertQueue(props.dispatch);

  if (!alert) {
    return null;
  }

  const displayData = getDisplayData(alert);
  const alertAudioUrl = getAlertAudioUrl(alert.event);

  return (
    <Wrapper>
      <Container key={alert.id}>
        <audio autoPlay>
          <source src={alertAudioUrl} type="audio/mp3" />
        </audio>
        <TextContainer>
          <Title src={displayData.titleSvgPath}></Title>
          <Subtitle eventType={alert.event}>{displayData.subtitle}</Subtitle>
        </TextContainer>
        <Avatar eventType={alert.event} url={displayData.logoUrl} />
      </Container>
    </Wrapper>
  );
}
```

There’s a queuing system, because we want to account for lots of things happening at once, especially with a substantial amount of viewers



```
function getDisplayData(alert: AlertQueueEvent): any {
  switch (alert.event) {
    case MainframeEvent.shoutOut: …
    case MainframeEvent.startGiveaway: …
    case MainframeEvent.endGiveaway: …
    case MainframeEvent.announceGiveaway: …
    case MainframeEvent.drawGiveaway: …
    case MainframeEvent.follow:
      return {
        titleSvgPath: `${baseSvgPath}follow.svg`,
        subtitle: alert.data.followerName,
        logoUrl:
          alert.data.logoUrl === ""
          ? getRandomPantherImgUrl({ large: true }) : alert.data.logoUrl,
      };
    case MainframeEvent.raid: …
    case MainframeEvent.cheer: …
    case MainframeEvent.sub: …
    default:
      return null;
  }
```

And each type of alert shows a different image and subtitle depending on what’s sent



index.tsx M src/components/alerts/index.tsx/...

```
function getAlertAudioUrl(type: MainframeEvent) {
  switch (type) {
    case MainframeEvent.startGiveaway: ⋯
    case MainframeEvent.endGiveaway: ⋯
    case MainframeEvent.announceGiveaway: ⋯
    case MainframeEvent.drawGiveaway: ⋯
    case MainframeEvent.follow:
      return "/assets/audio/alerts/follow.mp3";
    case MainframeEvent.raid:
    case MainframeEvent.special: ⋯
    case MainframeEvent.cheer: ⋯
    case MainframeEvent.sub: ⋯
    case MainframeEvent.shoutOut: ⋯
    default:
      return "";
  }
}

function getDisplayData(alert: AlertQueueEvent): any {
  switch (alert.event) {
    case MainframeEvent.shoutOut: ⋯
    case MainframeEvent.startGiveaway: ⋯
    case MainframeEvent.endGiveaway: ⋯
    case MainframeEvent.announceGiveaway: ⋯
    case MainframeEvent.drawGiveaway: ⋯
```

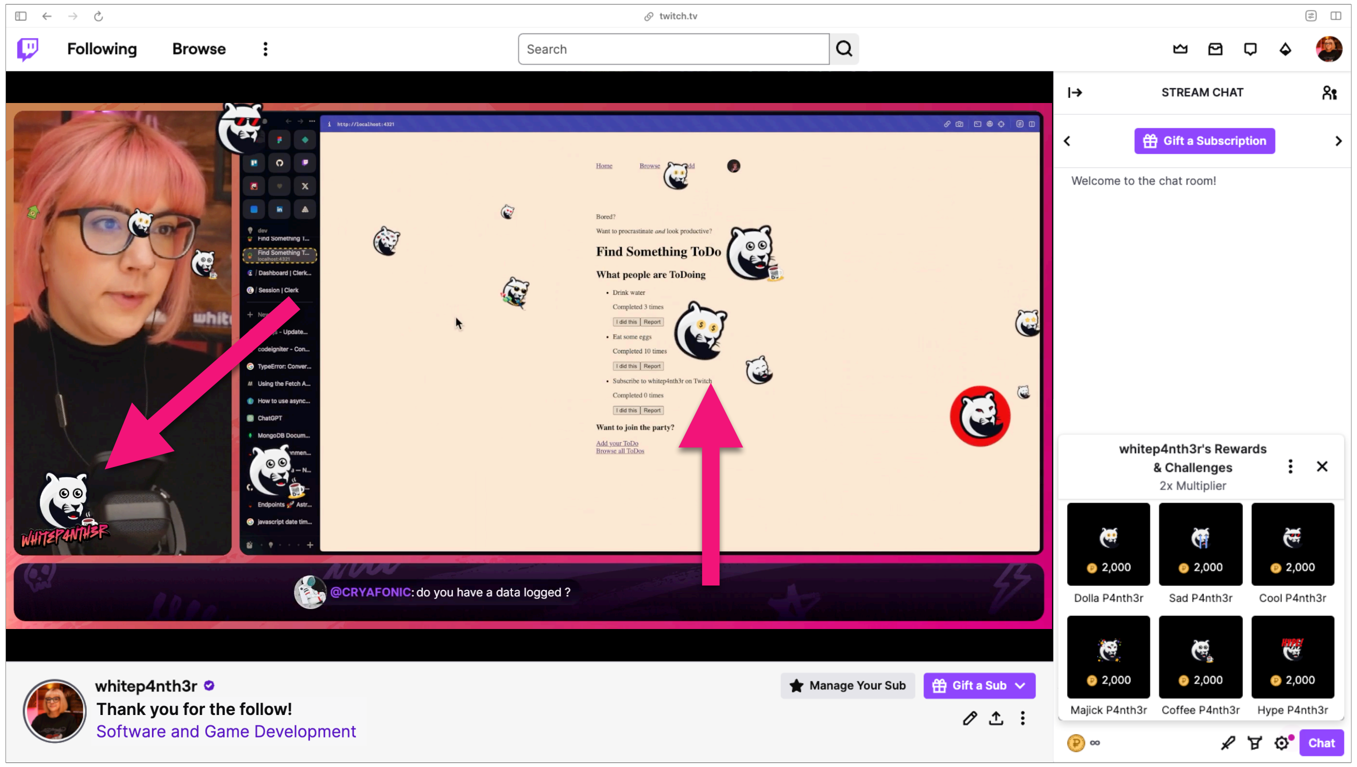
Each alert also plays a different alert sound, which both me and the viewers hear when I’m streaming.

All alert sounds are custom sounds written and recorded by me and my husband.

Funny story about custom sounds

When the repo was open source, I stored the mp3 files for alerts in my google drive, and used environment variables to access them when needed, so people wouldn’t steal the files from the repo. At random times the alert sounds would stop working, and it took us months to work out why. Turns out that too many alerts at any one time would DDoS the connection to Google Drive and deny the bot access. TLDR; my Twitch viewers DDoSsed Google.

And right now my follow alerts look like this



Welcome to the chat room!

Bored?

Want to procrastinate *and* look productive?

Find Something ToDo

What people are ToDoing

- Drink water

  Completed 3 times

- Eat some eggs

  Completed 10 times

- Subscribe to whitep4nth3r on Twitch

  Completed 0 times

Want to join the party?

Add your ToDo
Browse all ToDos

@CRYAFONIC: do you have a data logged ?

whitep4nth3r
Thank you for the follow!
Software and Game Development

whitep4nth3r's Rewards & Challenges
2x Multiplier

2,000 Dolla P4nth3r	2,000 Sad P4nth3r	2,000 Cool P4nth3r
2,000 Majick P4nth3r	2,000 Coffee P4nth3r	2,000 Hype P4nth3r

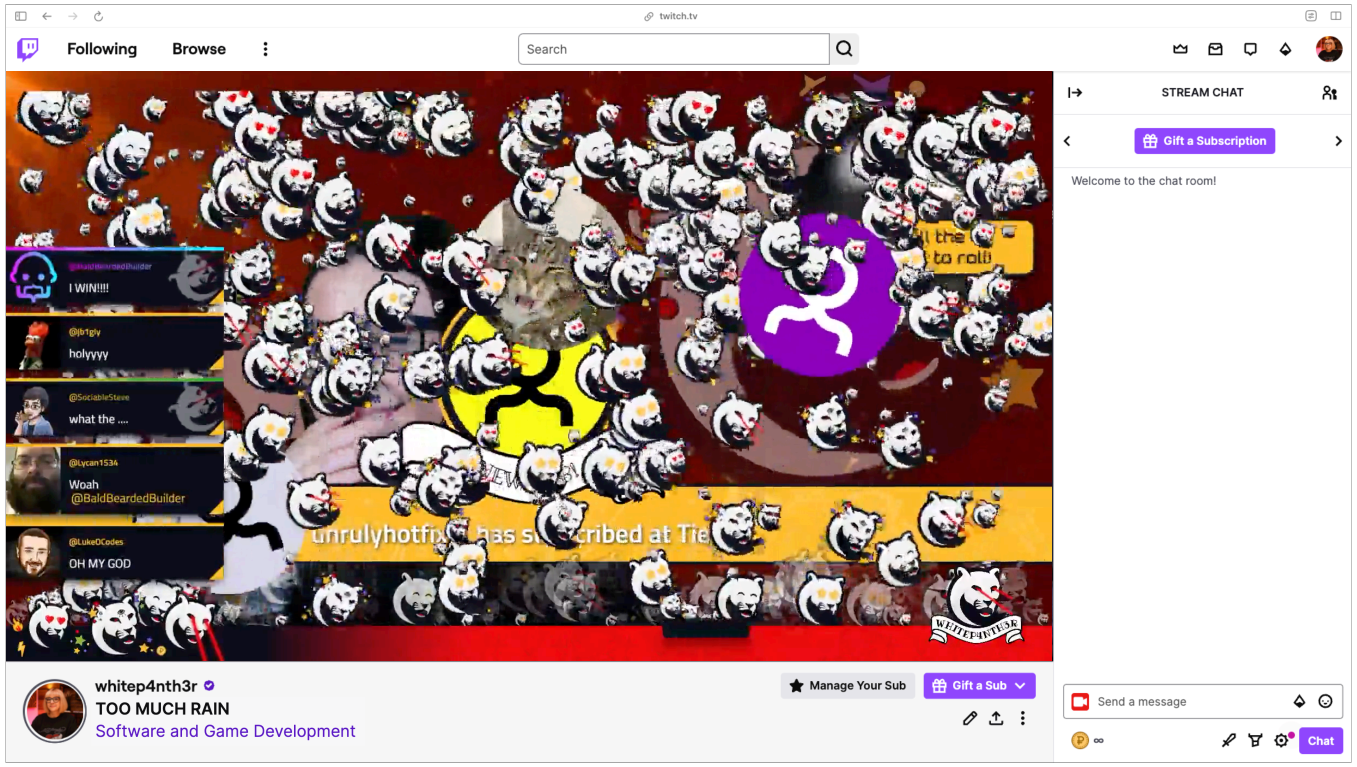
This is also when I accidentally coloured my hair to match my brand.

Also notice two other events that happened — one that changes the panther logo overlaid on my camera at random [click], and one that rains down my Twitch emotes [click] — all on separate browser sources in OBS, front ends listening to web socket events from the backend.

As you watch a stream you accumulate channel currency, and you can choose to spend that currency on fun stuff set up by the streamer. [click] So I enable viewers to trigger the logo change manually via channel point redemptions.

Viewers can also trigger the emote rain via a variety of different weather commands in chat, such as rain, hail, snow, blizzard etc

It has gone through a lot of iterations throughout the years… including needing to limit the amount of emotes that were dropped at any one time… because this was just ridiculous



@BaldBeardedBuilder
I WIN!!!!

@jb1gly
holyyyy

@SociableSteve
what the ....

@Lycan1534
Woah @BaldBeardedBuilder

@LukeOCodes
OH MY GOD

whitep4nth3r
TOO MUCH RAIN
Software and Game Development

Welcome to the chat room!

RIP my framerate

That happened when someone decided to spend stupid money on 50 gift subs to my channel all at once. It broke everything. They just kept coming.

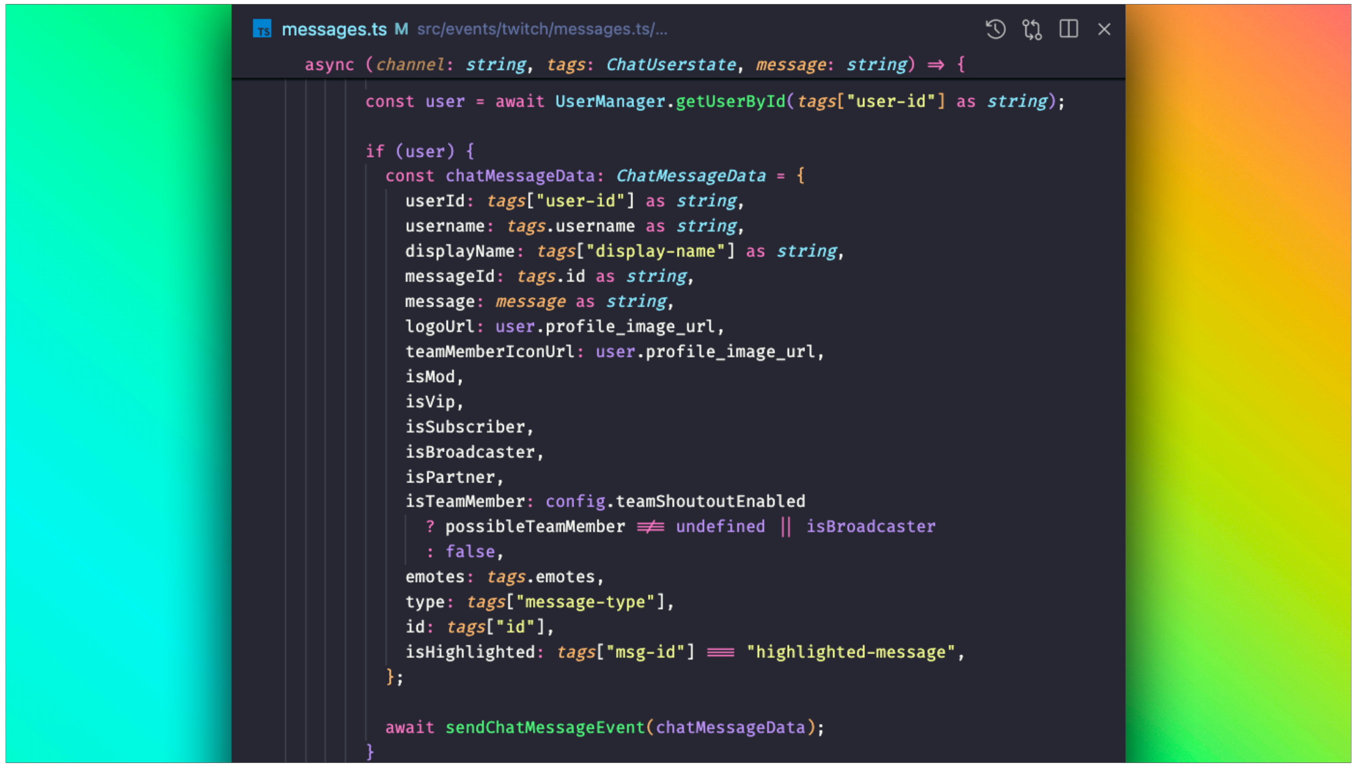
Yes, we heard that guitar riff 50 times

THAT’S ENTERTAINMENT LOL

But that’s entertainment
The act of writing the code to make that happen, live on Twitch, was entertainment
Viewers being able to make things happen on my stream, from anywhere in the world, is entertainment
And what sometimes happens unexpectedly as a result of writing that code, is also entertainment

CHAT

You’ve probably also noticed in my stream clips that I show chat messages on stream
Chat is another custom browser overlay
Powered by the central twitch bot
Which listens to message events from tmi.js
And sends events over the web socket connection

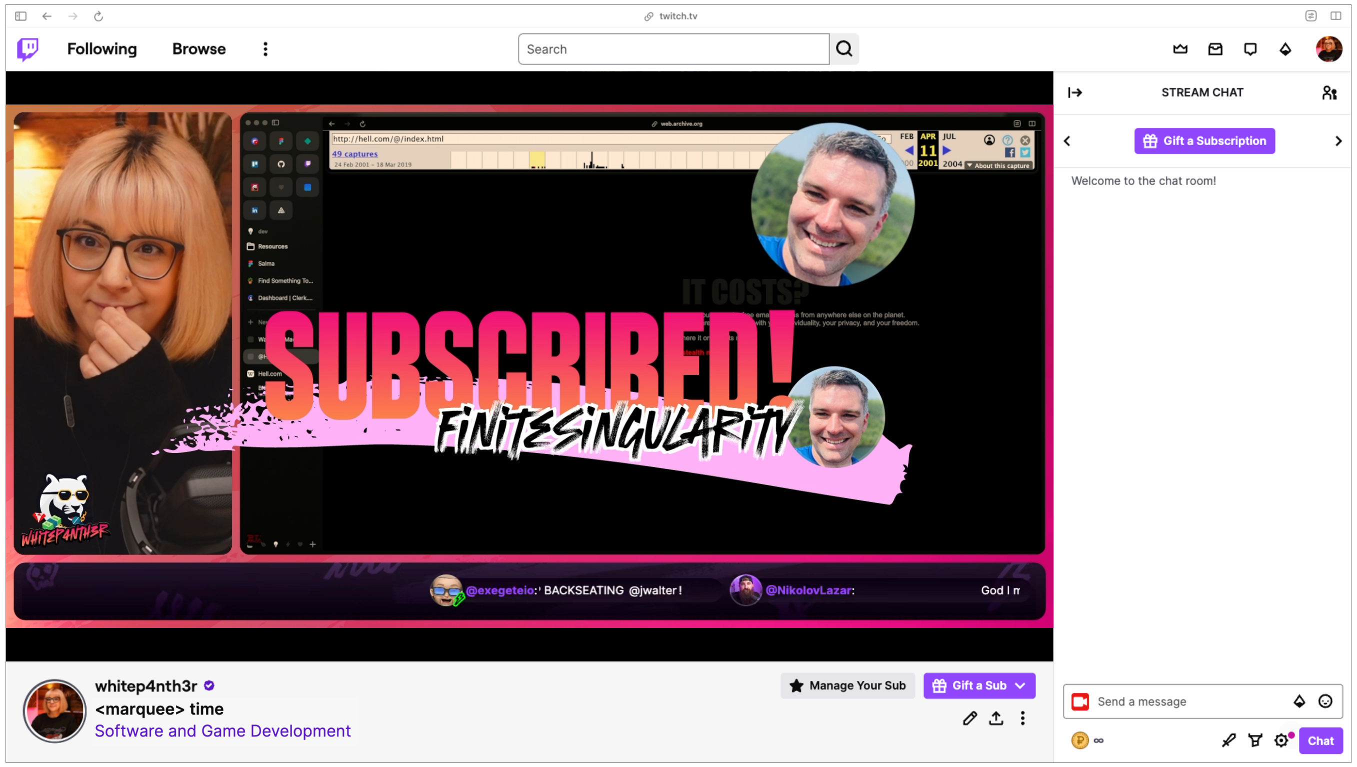


messages.ts M src/events/twitch/messages.ts/...

```
  async (channel: string, tags: ChatUserstate, message: string) => {
        const user = await UserManager.getUserById(tags["user-id"] as string);

        if (user) {
          const chatMessageData: ChatMessageData = {
            userId: tags["user-id"] as string,
            username: tags.username as string,
            displayName: tags["display-name"] as string,
            messageId: tags.id as string,
            message: message as string,
            logoUrl: user.profile_image_url,
            teamMemberIconUrl: user.profile_image_url,
            isMod,
            isVip,
            isSubscriber,
            isBroadcaster,
            isPartner,
            isTeamMember: config.teamShoutoutEnabled
              ? possibleTeamMember !== undefined || isBroadcaster
              : false,
            emotes: tags.emotes,
            type: tags["message-type"],
            id: tags["id"],
            isHighlighted: tags["msg-id"] === "highlighted-message",
          };

          await sendChatMessageEvent(chatMessageData);
        }
```

A lot of information is sent over the web socket to power the chat messages
Including vip/moderator/subscriber status

Viewers have been very eager to act as QA for chat, and suggest ideas for new features



Welcome to the chat room!

@exegeteio: BACKSEATING @jwalter!

@NikolovLazar:

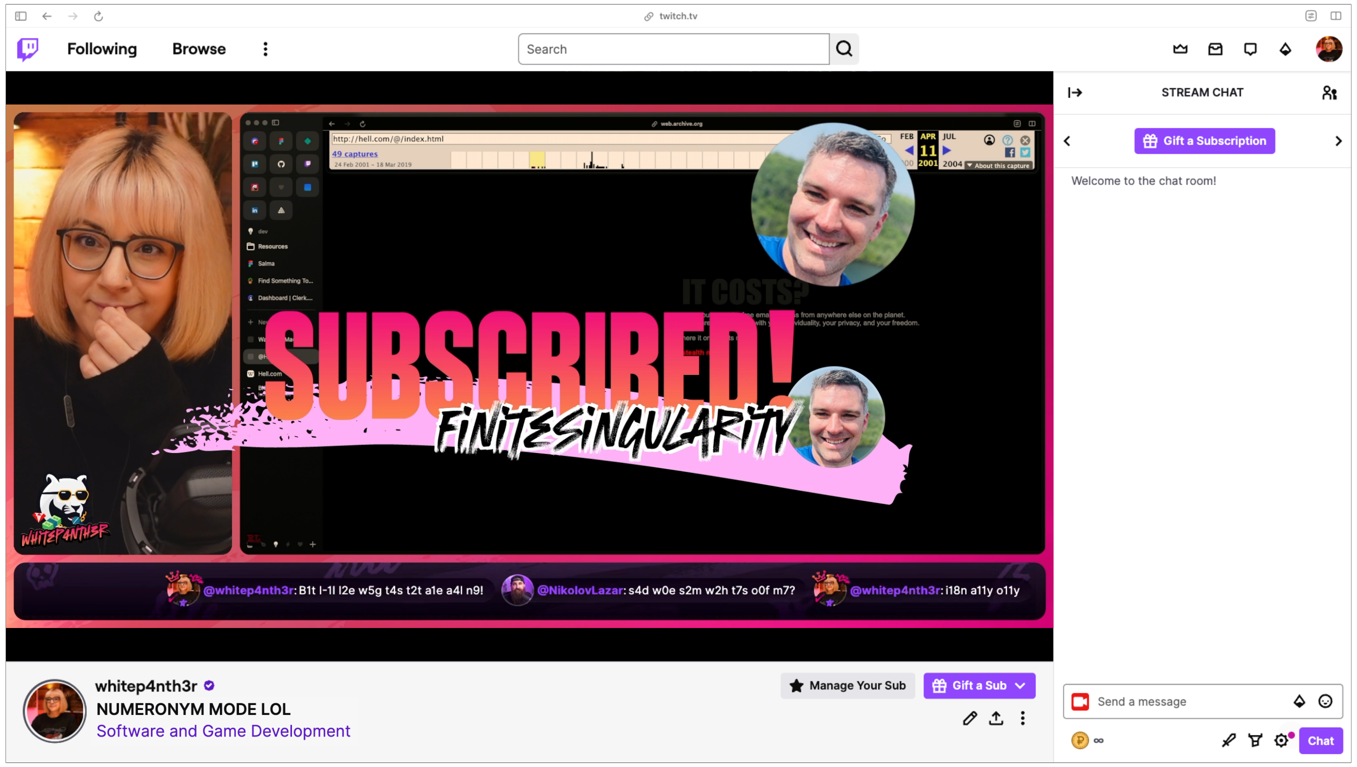
God I

whitep4nth3r

<marquee> time

Software and Game Development

Such as allowing the use of a marquee tag
An undocumented easter egg



Welcome to the chat room!

@whitep4nth3r: B1t l-1l l2e w5g t4s t2t a1e a4l n9!

@NikolovLazar: s4d w0e s2m w2h t7s o0f m7?

@whitep4nth3r: i18n a11y o11y

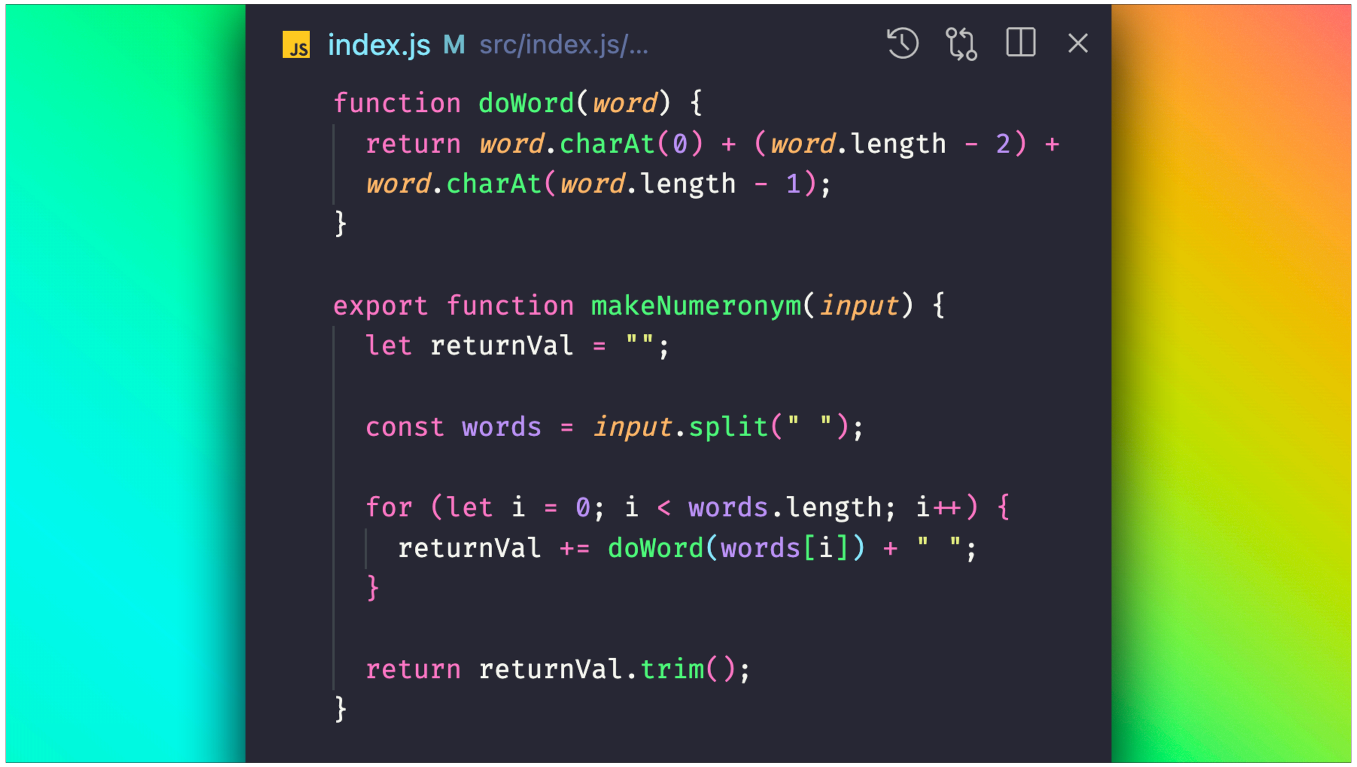
whitep4nth3r

NUMERONYM MODE LOL

Software and Game Development

And numeronym mode, which is a channel point redemption

A numeronym is a number-based word. Numeronyms are commonly used in tech used to shorten long words such as accessibility, internationalisation, Kubernetes, localisation, etc

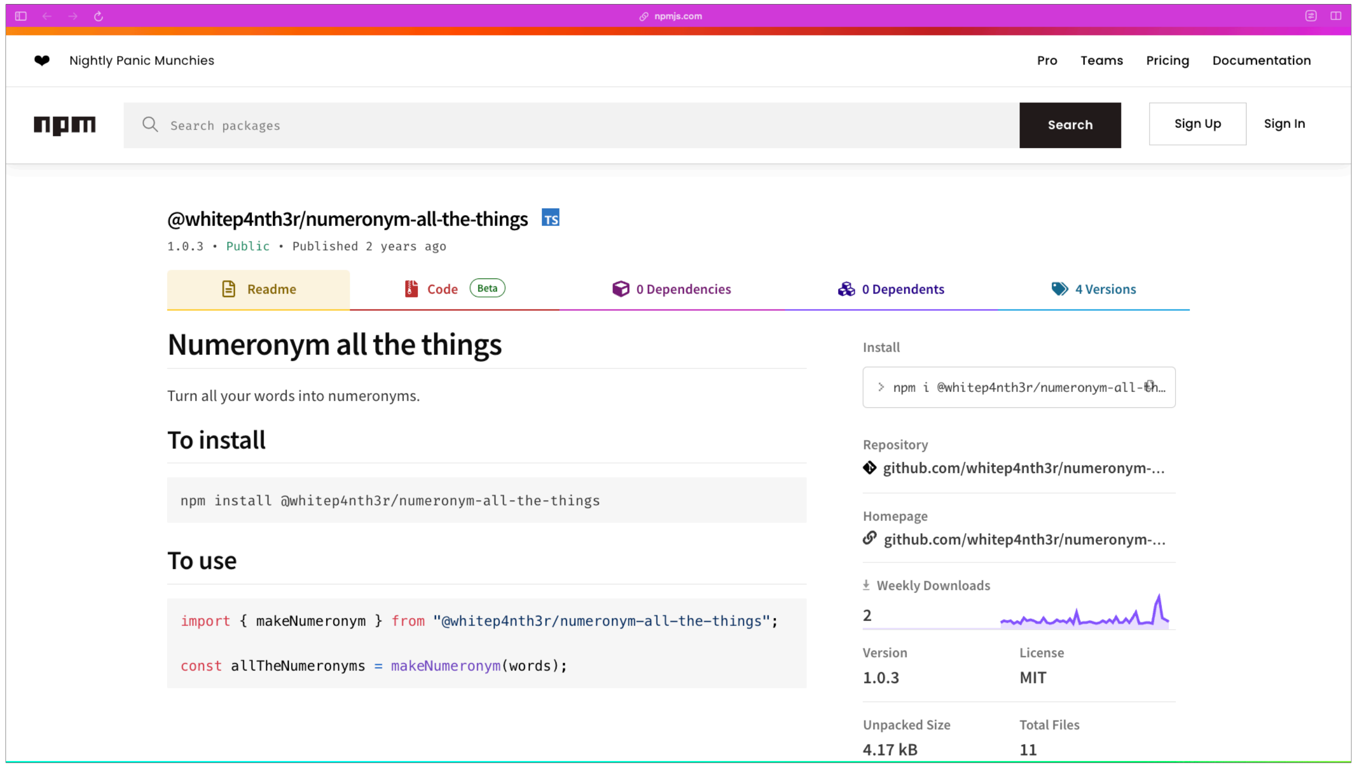


```
function doWord(word) {
  return word.charAt(0) + (word.length - 2) +
  word.charAt(word.length - 1);
}

export function makeNumeronym(input) {
  let returnVal = "";

  const words = input.split(" ");

  for (let i = 0; i < words.length; i++) {
    returnVal += doWord(words[i]) + " ";
  }

  return returnVal.trim();
}
```

Here is the code for numeronym mode

The make numeronym function splits the full message by space
And for each word, it returns the first character, the count of the characters in the middle, and the last character

And for some reason I published this to npm???
So I could use this in multiple places if I needed to



@whitep4nth3r/numeronym-all-the-things TS

1.0.3 • Public • Published 2 years ago

Numeronym all the things

Turn all your words into numeronyms.

To install

```
npm install @whitep4nth3r/numeronym-all-the-things
```

To use

```
import { makeNumeronym } from "@whitep4nth3r/numeronym-all-the-things";

const allTheNumeronyms = makeNumeronym(words);
```

Install

```
> npm i @whitep4nth3r/numeronym-all-th…
```

Repository
github.com/whitep4nth3r/numeronym-…

Homepage
github.com/whitep4nth3r/numeronym-…

Weekly Downloads
2

Version	License
1.0.3	MIT

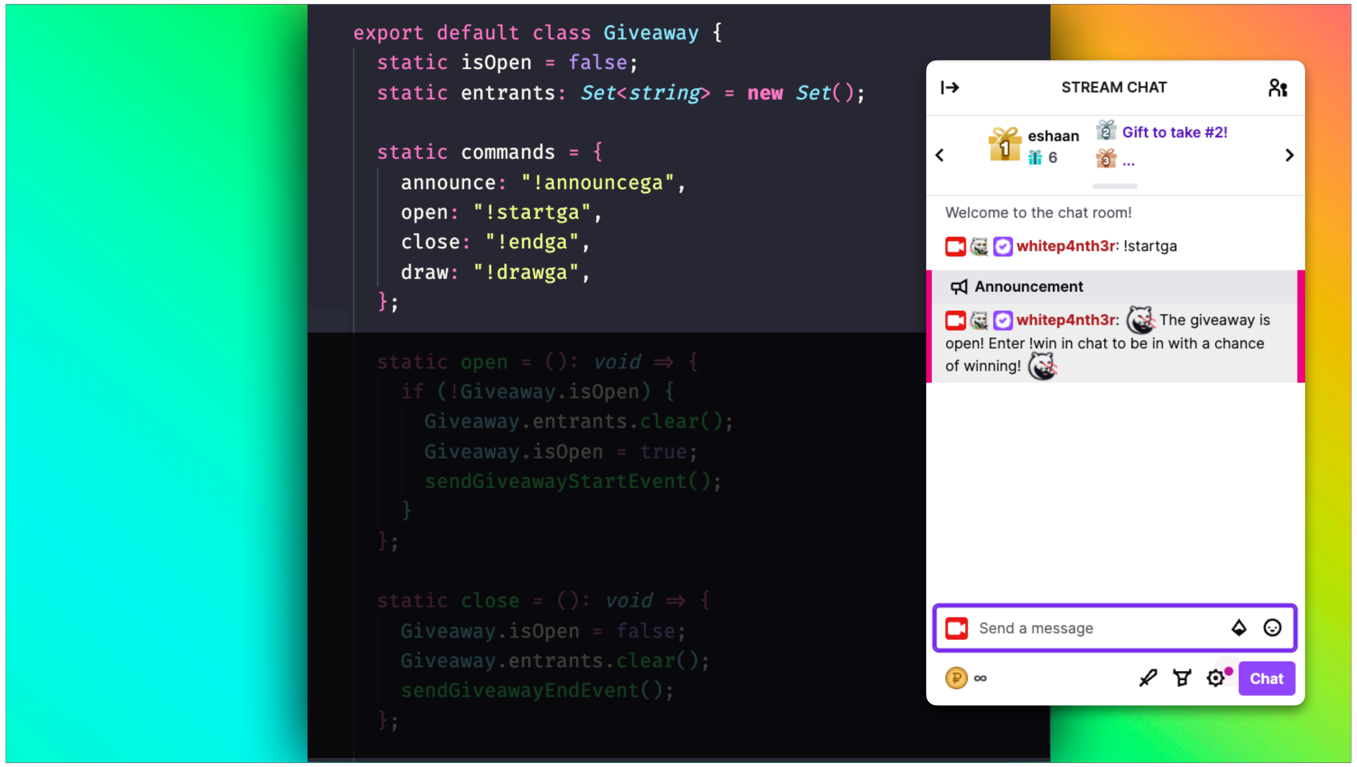
Unpacked Size	Total Files
4.17 kB	11

But people are downloading it???
So that’s fun.

But I’ve never used it any other projects.

GIVEAWAYS

I also built a giveaway mechanism into my twitch bot, live on stream of course



```
export default class Giveaway {
  static isOpen = false;
  static entrants: Set<string> = new Set();

  static commands = {
    announce: "!announcega",
    open: "!startga",
    close: "!endga",
    draw: "!drawga",
  };

  static open = (): void => {
    if (!Giveaway.isOpen) {
      Giveaway.entrants.clear();
      Giveaway.isOpen = true;
      sendGiveawayStartEvent();
    }
  };

  static close = (): void => {
    Giveaway.isOpen = false;
    Giveaway.entrants.clear();
    sendGiveawayEndEvent();
  };
```

Welcome to the chat room!

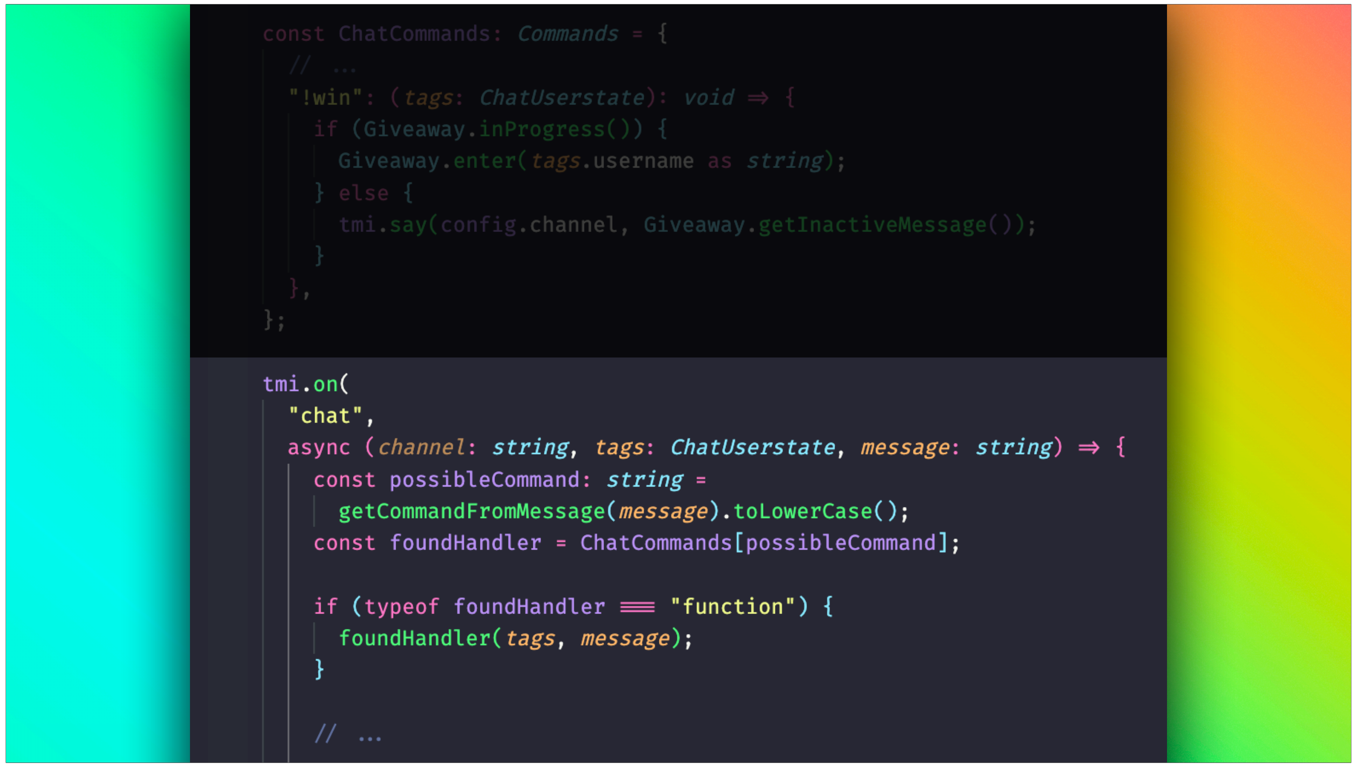
whitep4nth3r: !startga

Announcement

whitep4nth3r: The giveaway is open! Enter !win in chat to be in with a chance of winning!

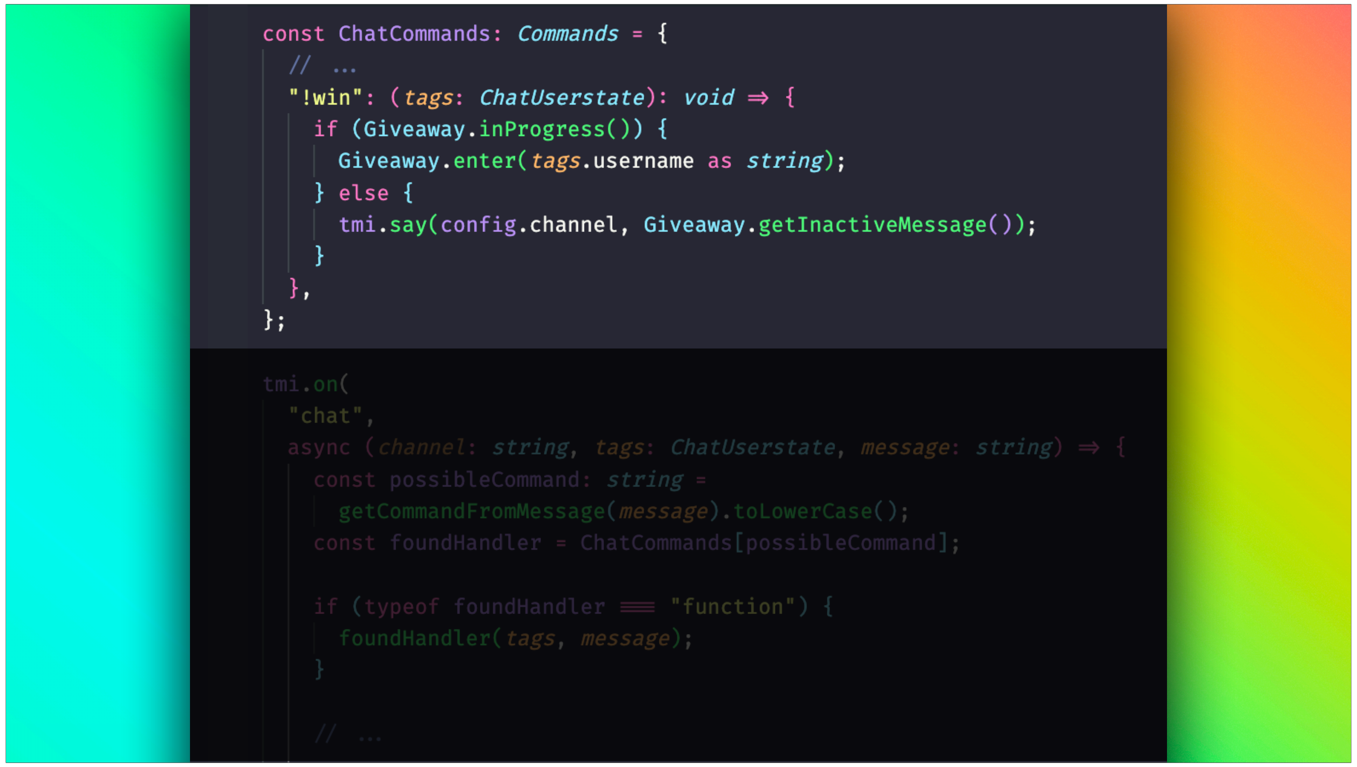
Giveaways are powered by typing commands in twitch chat [click] such as !start-ga

To enter, viewers type !win in chat



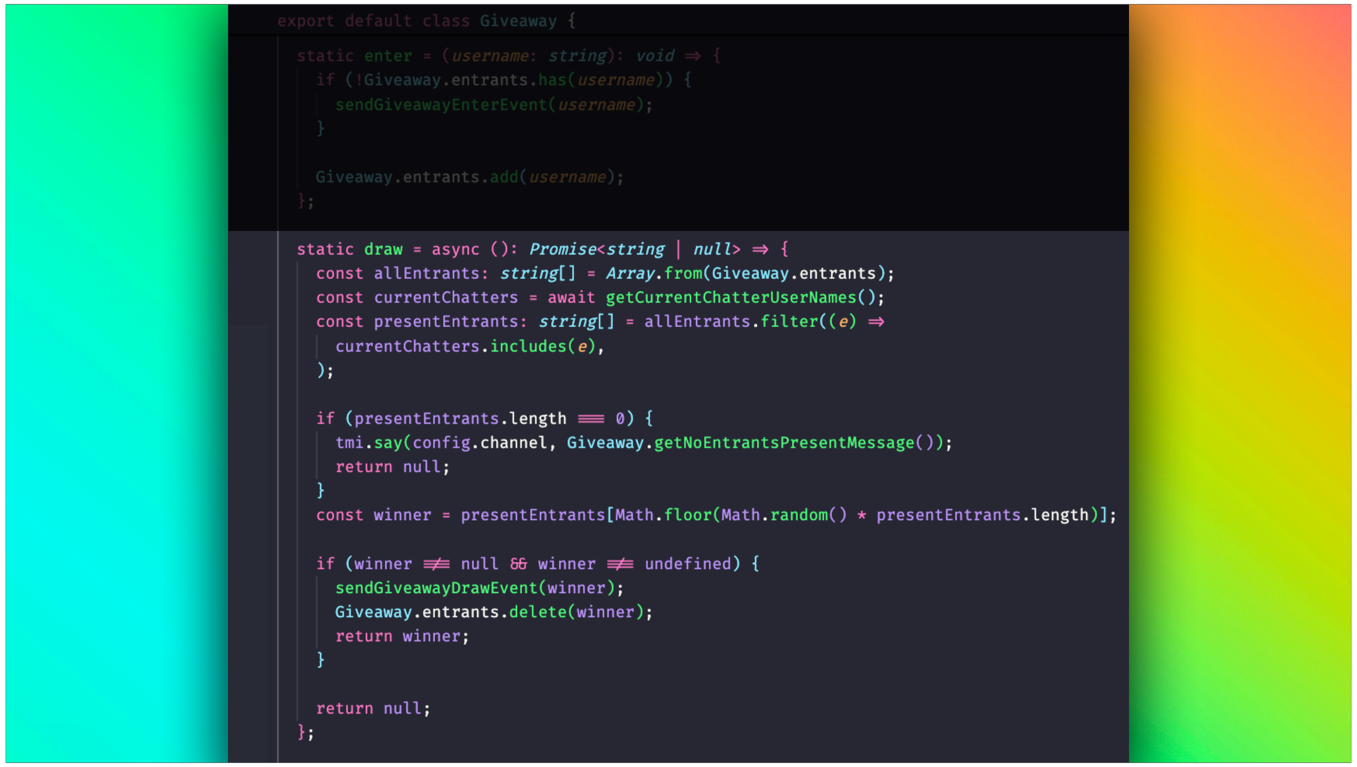
```
const ChatCommands: Commands = {
  // ...
  "!win": (tags: ChatUserstate): void => {
    if (Giveaway.inProgress()) {
      Giveaway.enter(tags.username as string);
    } else {
      tmi.say(config.channel, Giveaway.getInactiveMessage());
    }
  },
};

tmi.on(
  "chat",
  async (channel: string, tags: ChatUserstate, message: string) => {
    const possibleCommand: string =
      getCommandFromMessage(message).toLowerCase();
    const foundHandler = ChatCommands[possibleCommand];

    if (typeof foundHandler === "function") {
      foundHandler(tags, message);
    }

    // ...
```

tmi.js listens to chat, to find possible commands that the application knows about
If a command is found, then we run the handler for that command



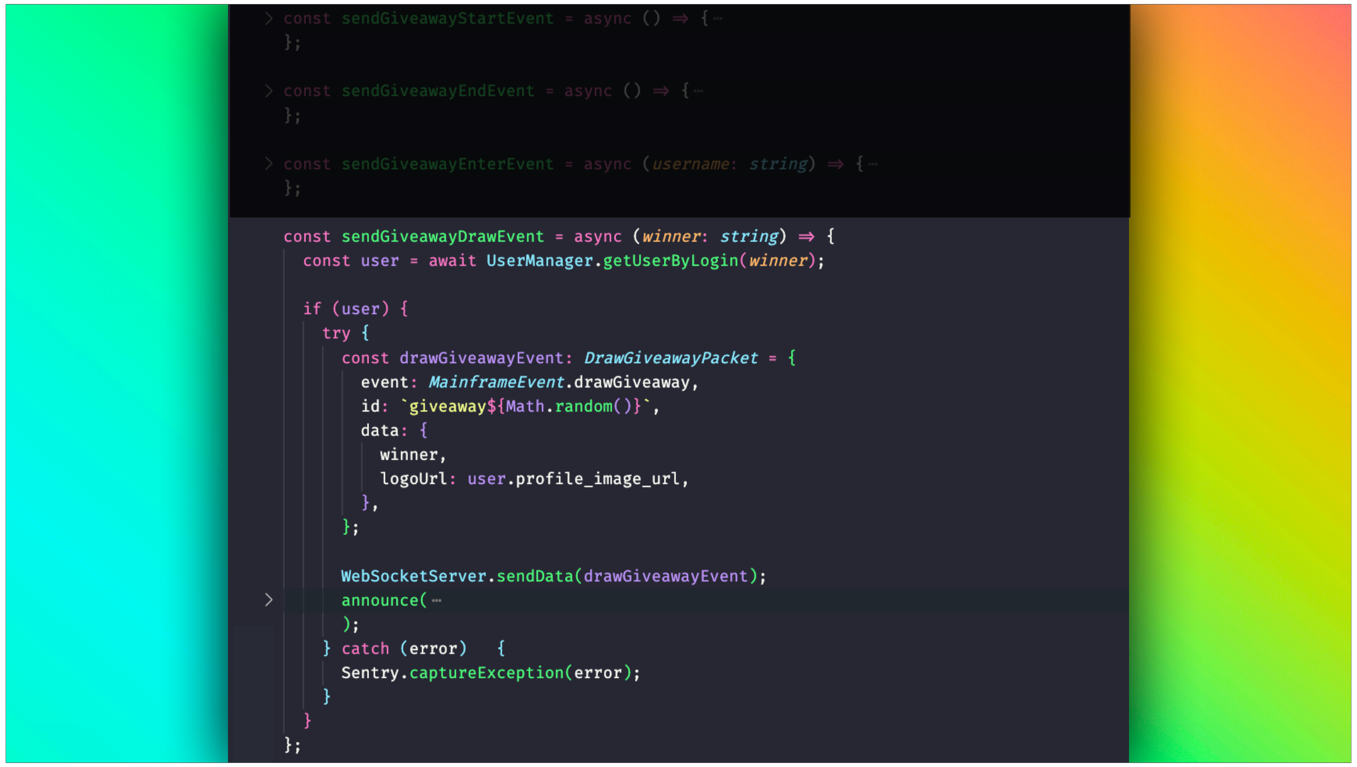
```
const ChatCommands: Commands = {
  // ...
  "!win": (tags: ChatUserstate): void => {
    if (Giveaway.inProgress()) {
      Giveaway.enter(tags.username as string);
    } else {
      tmi.say(config.channel, Giveaway.getInactiveMessage());
    }
  },
};

tmi.on(
  "chat",
  async (channel: string, tags: ChatUserstate, message: string) => {
    const possibleCommand: string =
      getCommandFromMessage(message).toLowerCase();
    const foundHandler = ChatCommands[possibleCommand];

    if (typeof foundHandler === "function") {
      foundHandler(tags, message);
    }

    // ...
```

if tmi.js reads !win in chat and a giveaway is in progress, we enter the viewer into the giveaway



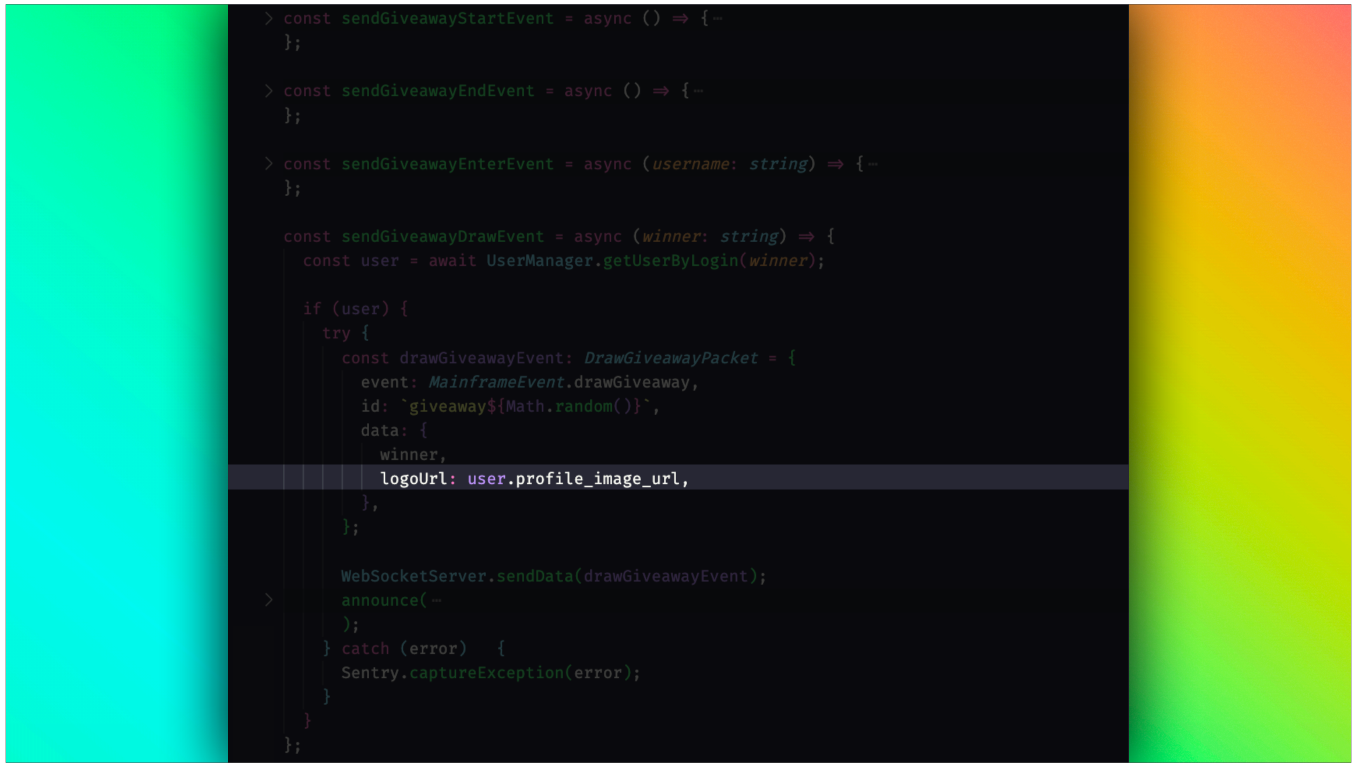
```typescript
export default class Giveaway {
  static enter = (username: string): void => {
    if (!Giveaway.entrants.has(username)) {
      sendGiveawayEnterEvent(username);
    }

    Giveaway.entrants.add(username);
  };

  static draw = async (): Promise<string | null> => {
    const allEntrants: string[] = Array.from(Giveaway.entrants);
    const currentChatters = await getCurrentChatterUserNames();
    const presentEntrants: string[] = allEntrants.filter((e) =>
      currentChatters.includes(e),
    );

    if (presentEntrants.length === 0) {
      tmi.say(config.channel, Giveaway.getNoEntrantsPresentMessage());
      return null;
    }
    const winner = presentEntrants[Math.floor(Math.random() * presentEntrants.length)];

    if (winner !== null && winner !== undefined) {
      sendGiveawayDrawEvent(winner);
      Giveaway.entrants.delete(winner);
      return winner;
    }

    return null;
  };
```

And then to draw, I type !drawga in chat
And a random winner is selected from entrants who are still present in the chat



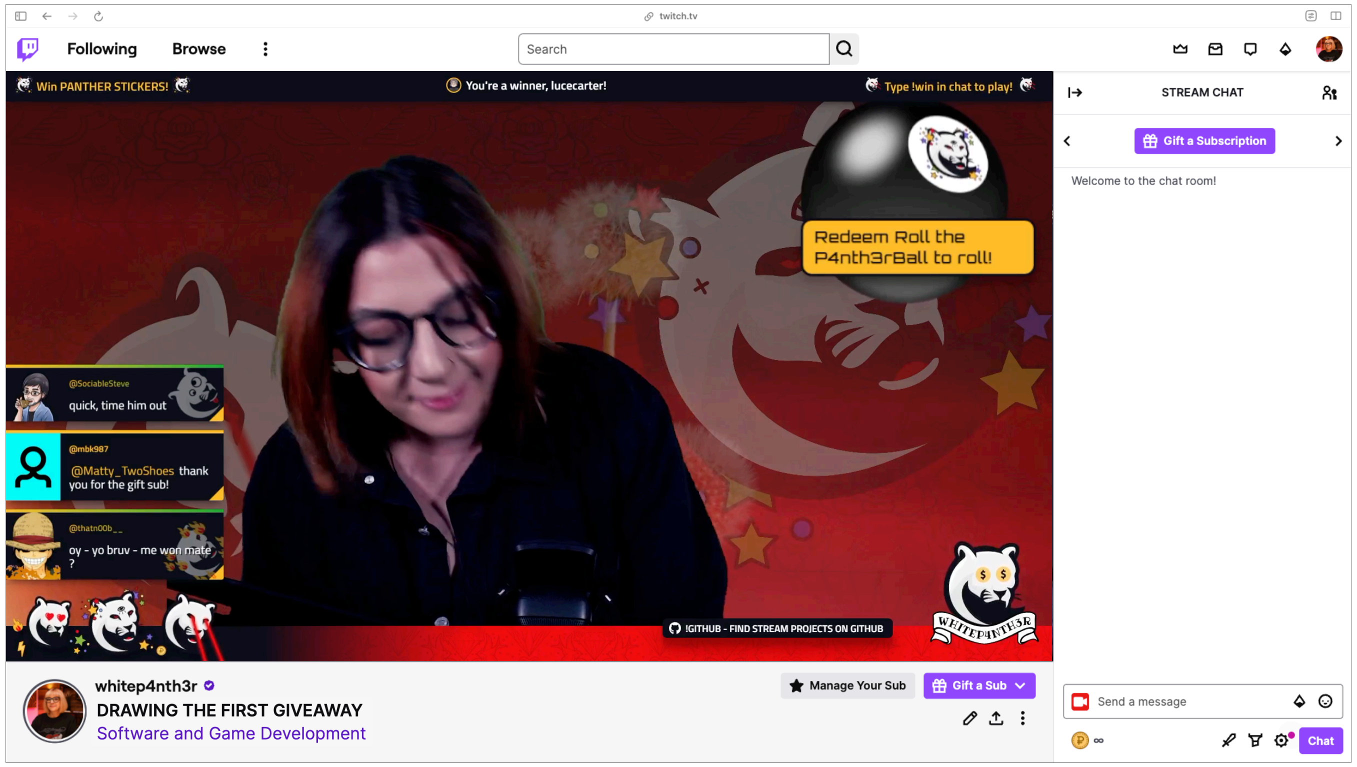
```
const sendGiveawayStartEvent = async () => {⋯
};

const sendGiveawayEndEvent = async () => {⋯
};

const sendGiveawayEnterEvent = async (username: string) => {⋯
};

const sendGiveawayDrawEvent = async (winner: string) => {
  const user = await UserManager.getUserByLogin(winner);

  if (user) {
    try {
      const drawGiveawayEvent: DrawGiveawayPacket = {
        event: MainframeEvent.drawGiveaway,
        id: `giveaway${Math.random()}`,
        data: {
          winner,
          logoUrl: user.profile_image_url,
        },
      };

      WebSocketServer.sendData(drawGiveawayEvent);
      announce(⋯
      );
    } catch (error)   {
      Sentry.captureException(error);
    }
  }
};
```

And then we send the event over the web socket to the browser source overlays listening in OBS



```
const sendGiveawayStartEvent = async () => {⋯
};

const sendGiveawayEndEvent = async () => {⋯
};

const sendGiveawayEnterEvent = async (username: string) => {⋯
};

const sendGiveawayDrawEvent = async (winner: string) => {
  const user = await UserManager.getUserByLogin(winner);

  if (user) {
    try {
      const drawGiveawayEvent: DrawGiveawayPacket = {
        event: MainframeEvent.drawGiveaway,
        id: `giveaway${Math.random()}`,
        data: {
          winner,
          logoUrl: user.profile_image_url,
        },
      };

      WebSocketServer.sendData(drawGiveawayEvent);
      announce(⋯
      );
    } catch (error)   {
      Sentry.captureException(error);
    }
  }
};
```

Again, the key thing here is to send the user’s profile image URL over the websocket
Here’s a clip of of the giveaway in action!



Win PANTHER STICKERS!

You're a winner, lucecarter!

Type !win in chat to play!

Redeem Roll the P4nth3rBall to roll!

@SociableSteve
quick, time him out

@mbk987
@Matty_TwoShoes thank you for the gift sub!

@thatn00b__
oy - yo bruv - me won mate ?

!GITHUB - FIND STREAM PROJECTS ON GITHUB

WHITEP4NTH3R

whitep4nth3r
DRAWING THE FIRST GIVEAWAY
Software and Game Development

Welcome to the chat room!

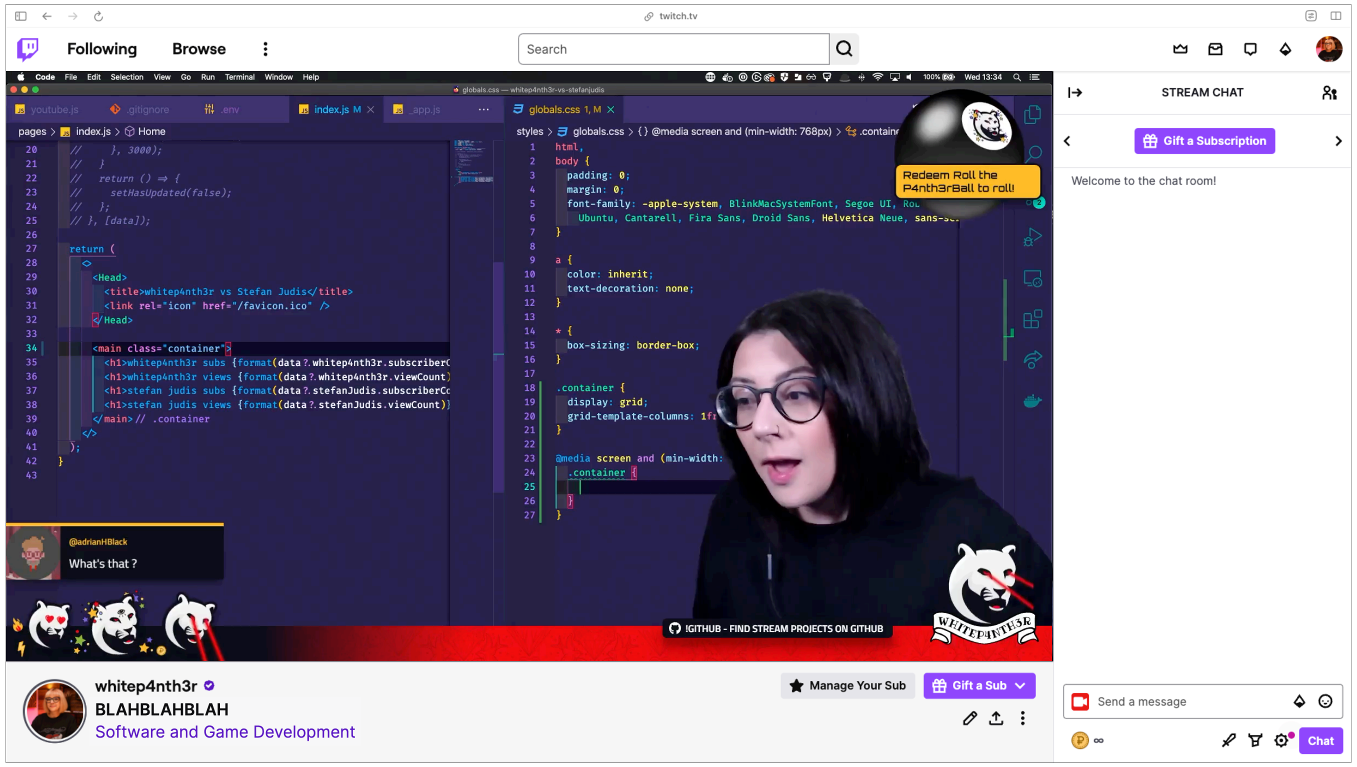
Stickers, merchandise, JetBrains licenses… postage costs for around the world were often extortionate

THE CAR

I want to tell you about the car

Backseating has always been a thing, kind of a meme

New viewers often come into the stream and tell me what to do — with no context about the problem we’re solving, and it’s the worst thing ever, like backseat driving



twitch.tv
Following
Browse
Search
Code File Edit Selection View Go Run Terminal Window Help
Wed 13:34
globals.css — whitep4nth3r-vs-stefanjudis
youtube.js
.gitignore
.env
index.js M
_app.js
globals.css 1, M
pages > index.js > Home
styles > globals.css > {} @media screen and (min-width: 768px) > .containe
Redeem Roll the P4nth3rBall to roll!
!GITHUB - FIND STREAM PROJECTS ON GITHUB
STREAM CHAT
Gift a Subscription
whitep4nth3r
BLAHBLAHBLAH
Software and Game Development
Manage Your Sub
Gift a Sub
Send a message
Chat

```
//     }, 3000);
//   }
//   return () => {
//     setHasUpdated(false);
//   };
// }, [data]);

return (
  <>
    <Head>
      <title>whitep4nth3r vs Stefan Judis</title>
      <link rel="icon" href="/favicon.ico" />
    </Head>

    <main class="container">
      <h1>whitep4nth3r subs {format(data?.whitep4nth3r.subscriberC
      <h1>whitep4nth3r views {format(data?.whitep4nth3r.viewCount)
      <h1>stefan judis subs {format(data?.stefanJudis.subscriberCo
      <h1>stefan judis views {format(data?.stefanJudis.viewCount)}
    </main>// .container
  </>
);
}
```

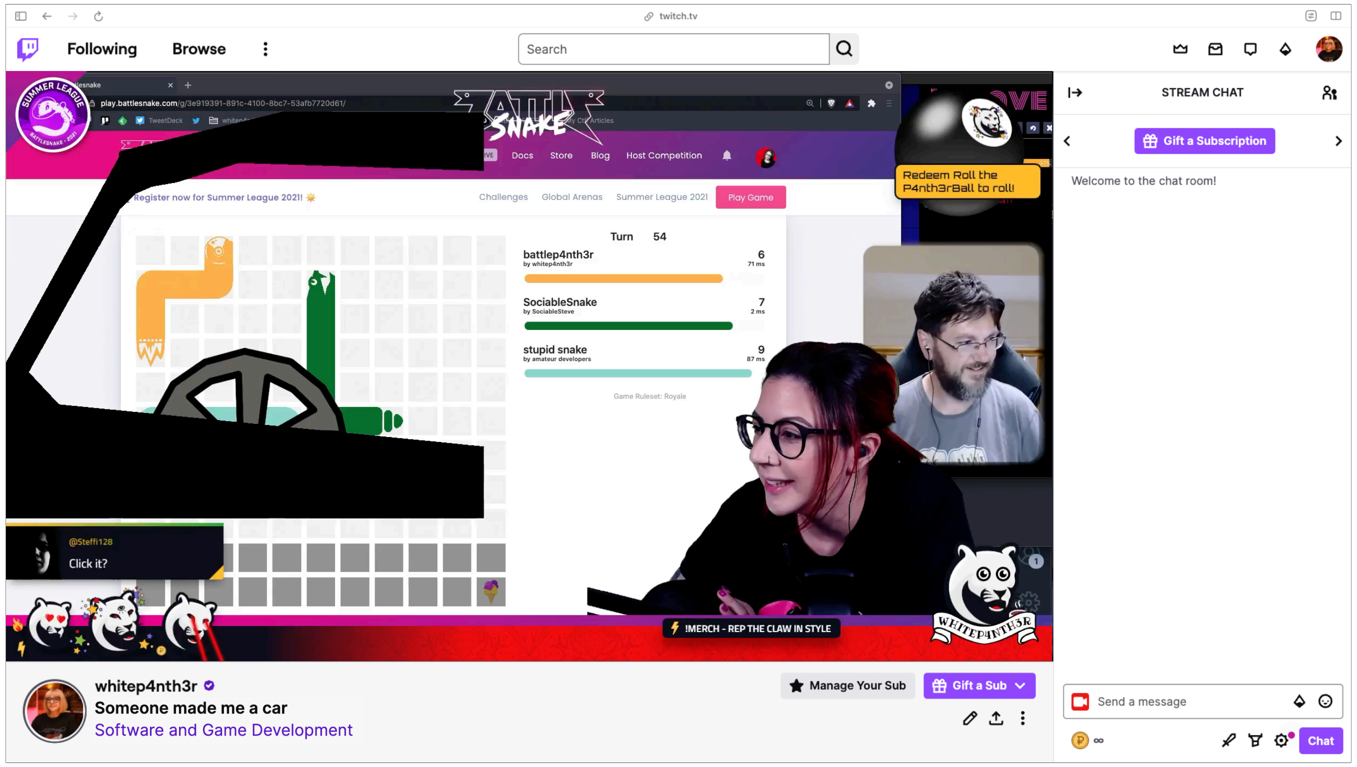
```
html,
body {
  padding: 0;
  margin: 0;
  font-family: -apple-system, BlinkMacSystemFont, Segoe UI, Rob
    Ubuntu, Cantarell, Fira Sans, Droid Sans, Helvetica Neue, sans-se
}

a {
  color: inherit;
  text-decoration: none;
}

* {
  box-sizing: border-box;
}

.container {
  display: grid;
  grid-template-columns: 1fr
}

@media screen and (min-width:
  .container {

  }
}
```

@adrianHBlack
What's that ?

Welcome to the chat room!

…

One of my viewers thought it would be a good idea to put backseaters in a literal car, and while I was streaming, sent me these very low fi drawn layers of a car that I added to the scene in OBS on the fly



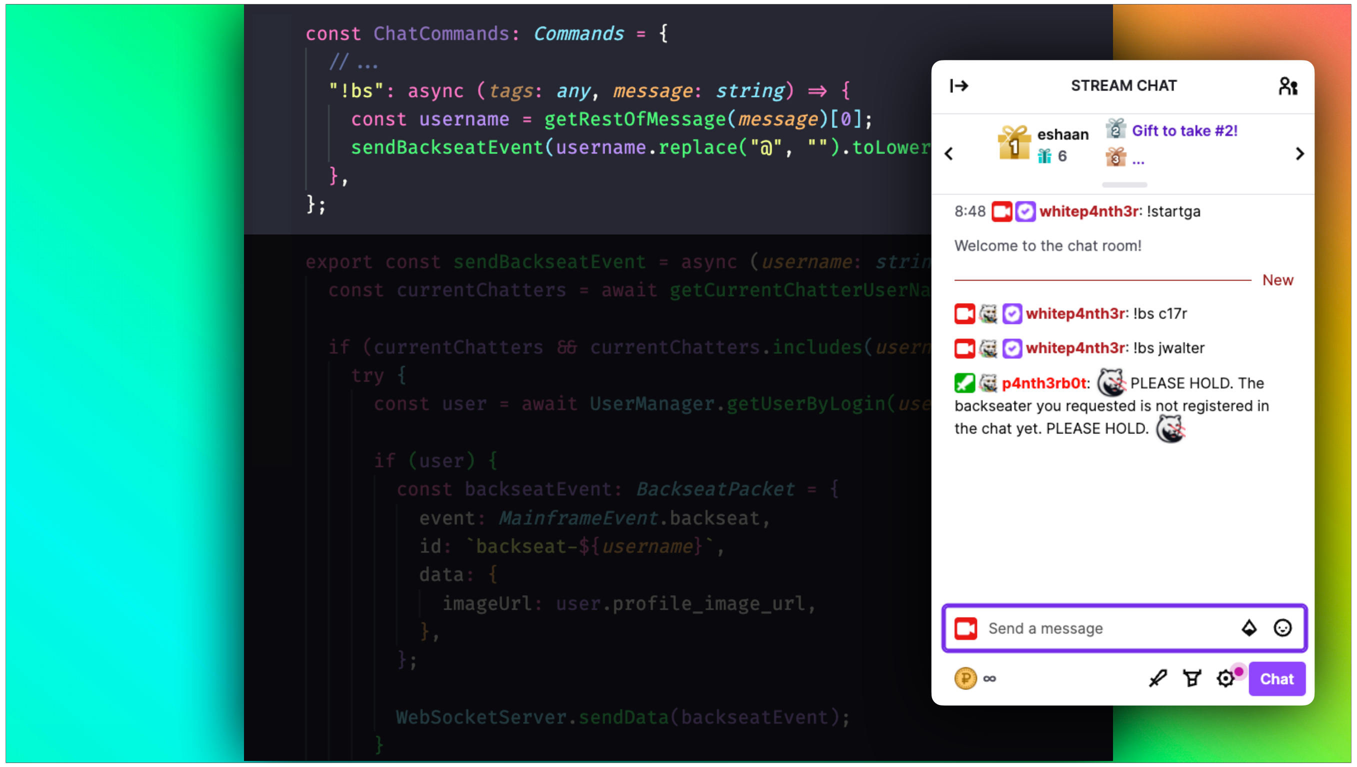
Welcome to the chat room!

@Steffi128
Click it?

whitep4nth3r
Someone made me a car
Software and Game Development

The car has evolved, of course, and is now a permanent fixture in my stream, and can be triggered by viewers

Here’s how it works



```
const ChatCommands: Commands = {
  // ...
  "!bs": async (tags: any, message: string) => {
    const username = getRestOfMessage(message)[0];
    sendBackseatEvent(username.replace("@", "").toLower
  },
};

export const sendBackseatEvent = async (username: strin
  const currentChatters = await getCurrentChatterUserNa

  if (currentChatters && currentChatters.includes(usern
    try {
      const user = await UserManager.getUserByLogin(use

      if (user) {
        const backseatEvent: BackseatPacket = {
          event: MainframeEvent.backseat,
          id: `backseat-${username}`,
          data: {
            imageUrl: user.profile_image_url,
          },
        };

        WebSocketServer.sendData(backseatEvent);
      }
```

8:48 whitep4nth3r: !startga

Welcome to the chat room!

whitep4nth3r: !bs c17r

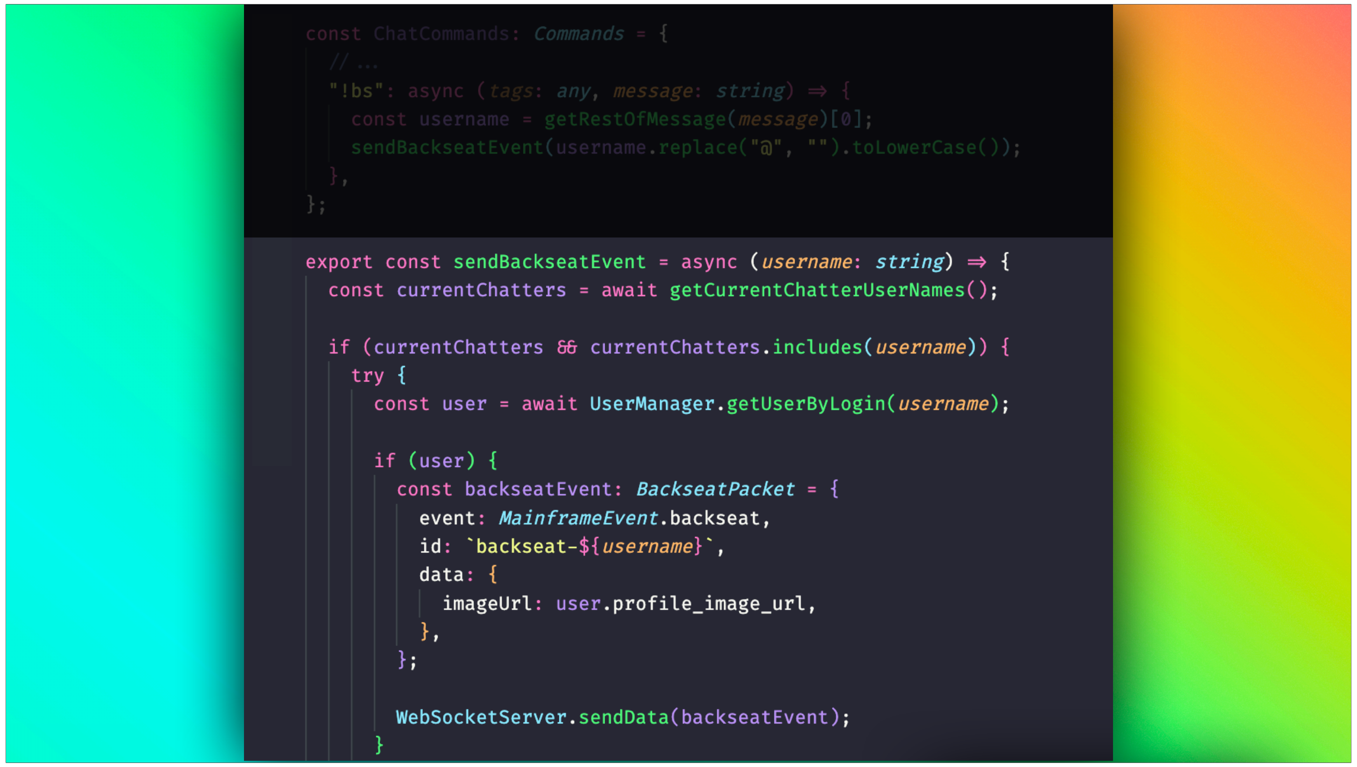
whitep4nth3r: !bs jwalter

p4nth3rb0t: PLEASE HOLD. The backseater you requested is not registered in the chat yet. PLEASE HOLD.

Again, tmi.js listens for a command in chat — !bs [click]

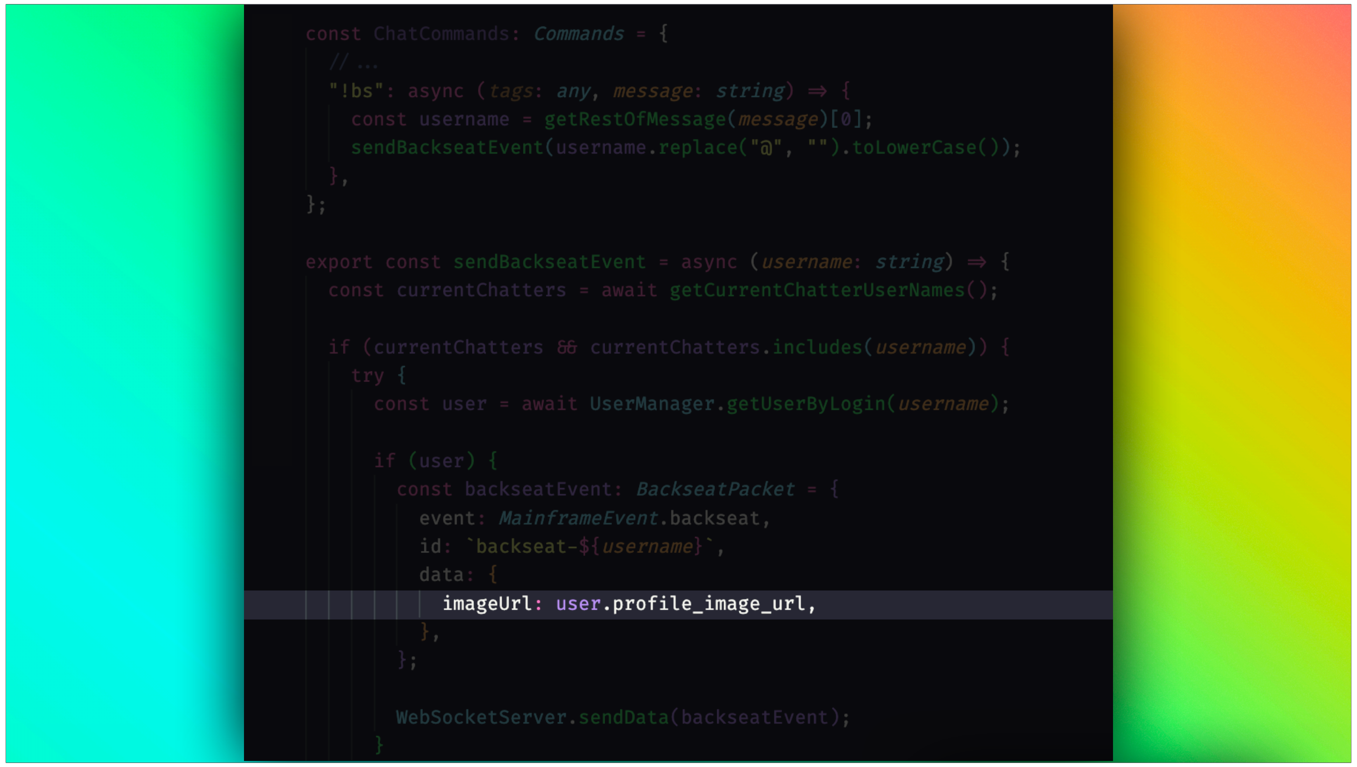
What bs stands for is open to interpretation here

And we grab the username from the message
We only put people in the backseat who are actually registered as viewing the stream
Otherwise that would be rude



```
const ChatCommands: Commands = {
  // ...
  "!bs": async (tags: any, message: string) => {
    const username = getRestOfMessage(message)[0];
    sendBackseatEvent(username.replace("@", "").toLowerCase());
  },
};

export const sendBackseatEvent = async (username: string) => {
  const currentChatters = await getCurrentChatterUserNames();

  if (currentChatters && currentChatters.includes(username)) {
    try {
      const user = await UserManager.getUserByLogin(username);

      if (user) {
        const backseatEvent: BackseatPacket = {
          event: MainframeEvent.backseat,
          id: `backseat-${username}`,
          data: {
            imageUrl: user.profile_image_url,
          },
        };

        WebSocketServer.sendData(backseatEvent);
      }
```

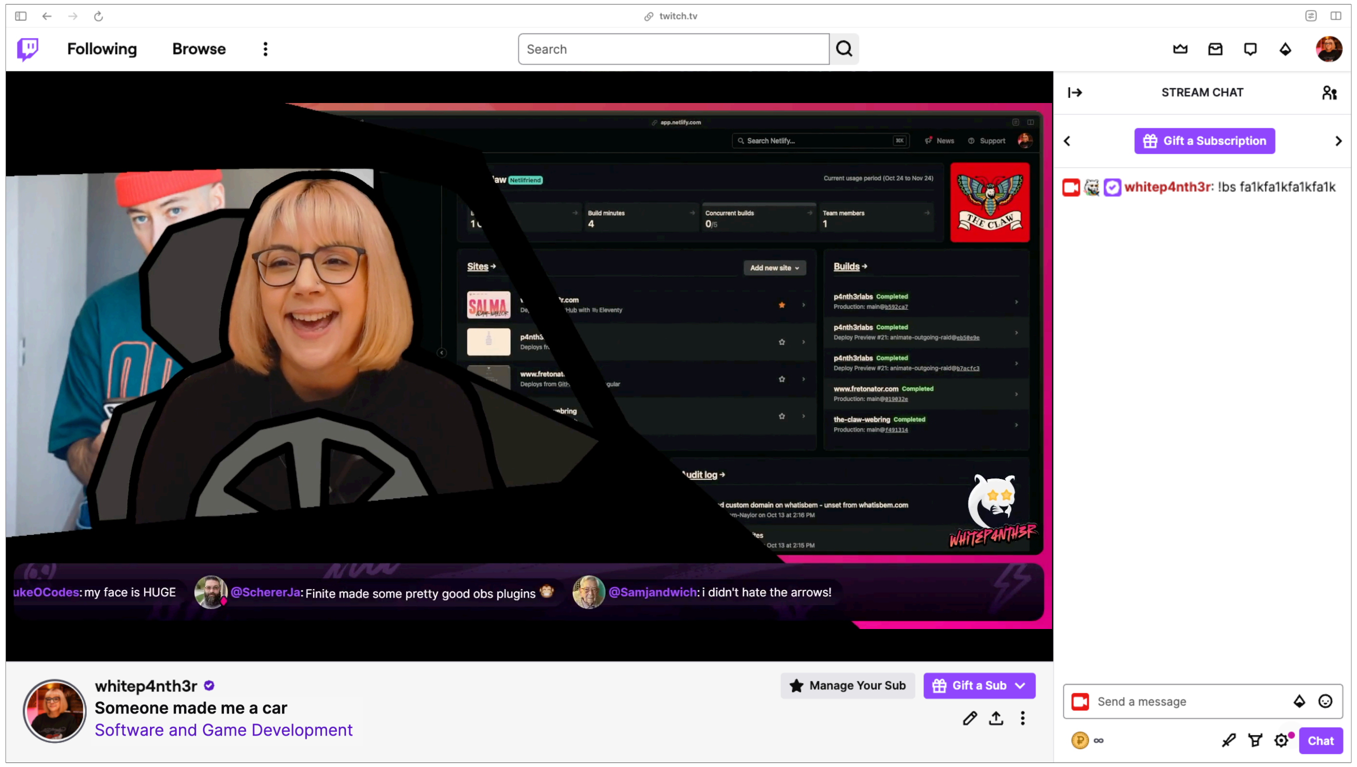
And then we send the event over the web socket.



```
const ChatCommands: Commands = {
  // ...
  "!bs": async (tags: any, message: string) ⇒ {
    const username = getRestOfMessage(message)[0];
    sendBackseatEvent(username.replace("@", "").toLowerCase());
  },
};

export const sendBackseatEvent = async (username: string) ⇒ {
  const currentChatters = await getCurrentChatterUserNames();

  if (currentChatters && currentChatters.includes(username)) {
    try {
      const user = await UserManager.getUserByLogin(username);

      if (user) {
        const backseatEvent: BackseatPacket = {
          event: MainframeEvent.backseat,
          id: `backseat-${username}`,
          data: {
            imageUrl: user.profile_image_url,
          },
        };

        WebSocketServer.sendData(backseatEvent);
      }
```

Again, making sure to send the profile image URL

The point of the car now is to be as offensively obtrusive as possible, like backseaters
And it looks like this



whitep4nth3r: !bs fa1kfa1kfa1kfa1k

ukeOCodes: my face is HUGE

@SchererJa: Finite made some pretty good obs plugins

@Samjandwich: i didn't hate the arrows!

whitep4nth3r

Someone made me a car

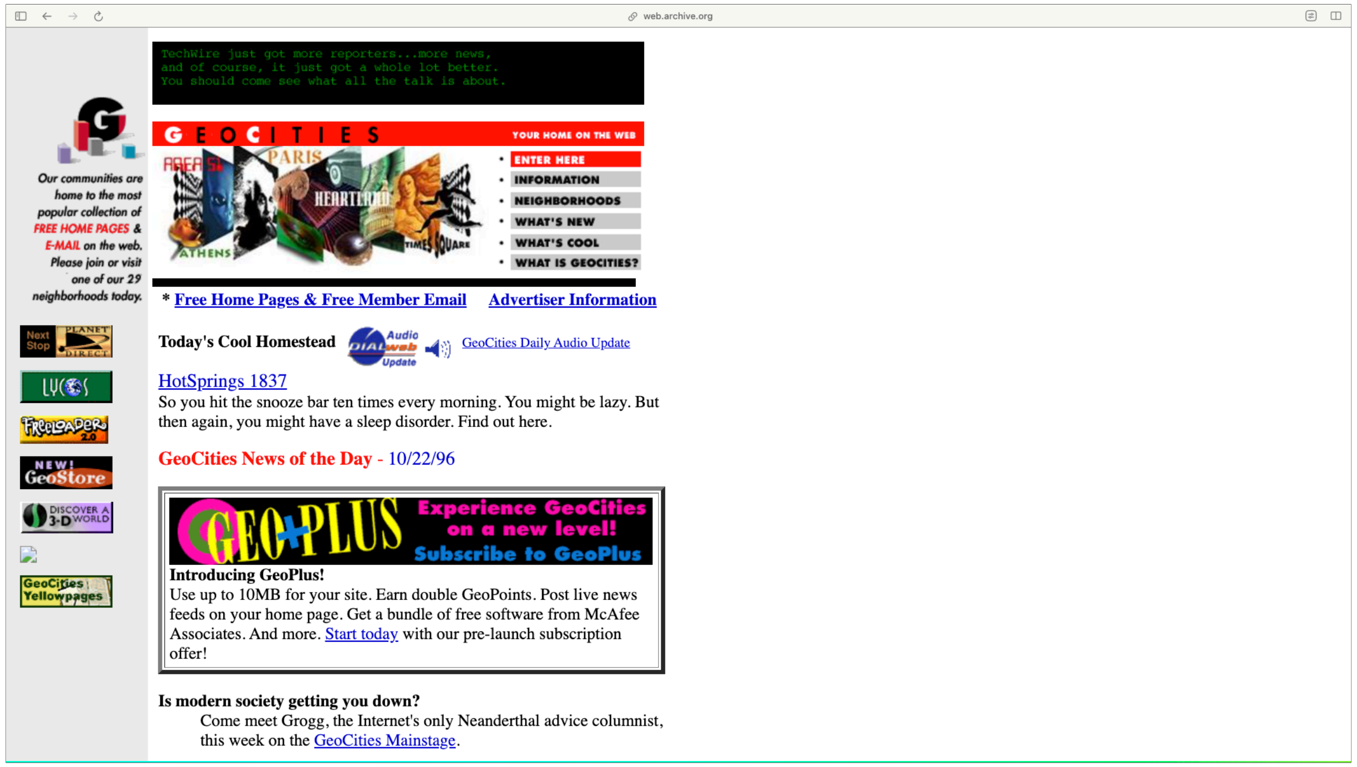
Software and Game Development

All sorts of magic involved in the car

Now I want to show you some silly things I've built on stream, how it brought people together, and what we learned along the way

I ❤️ GEOCITIES

Geocities close to my heart
The original website builder of the 90s
And in GCSE IT whilst people were making spreadsheets and text documents, I was making websites with geocities



TechWire just got more reporters...more news,
and of course, it just got a whole lot better.
You should come see what all the talk is about.

Our communities are home to the most popular collection of FREE HOME PAGES & E-MAIL on the web. Please join or visit one of our 29 neighborhoods today.

* Free Home Pages & Free Member Email   Advertiser Information

Today's Cool Homestead   GeoCities Daily Audio Update

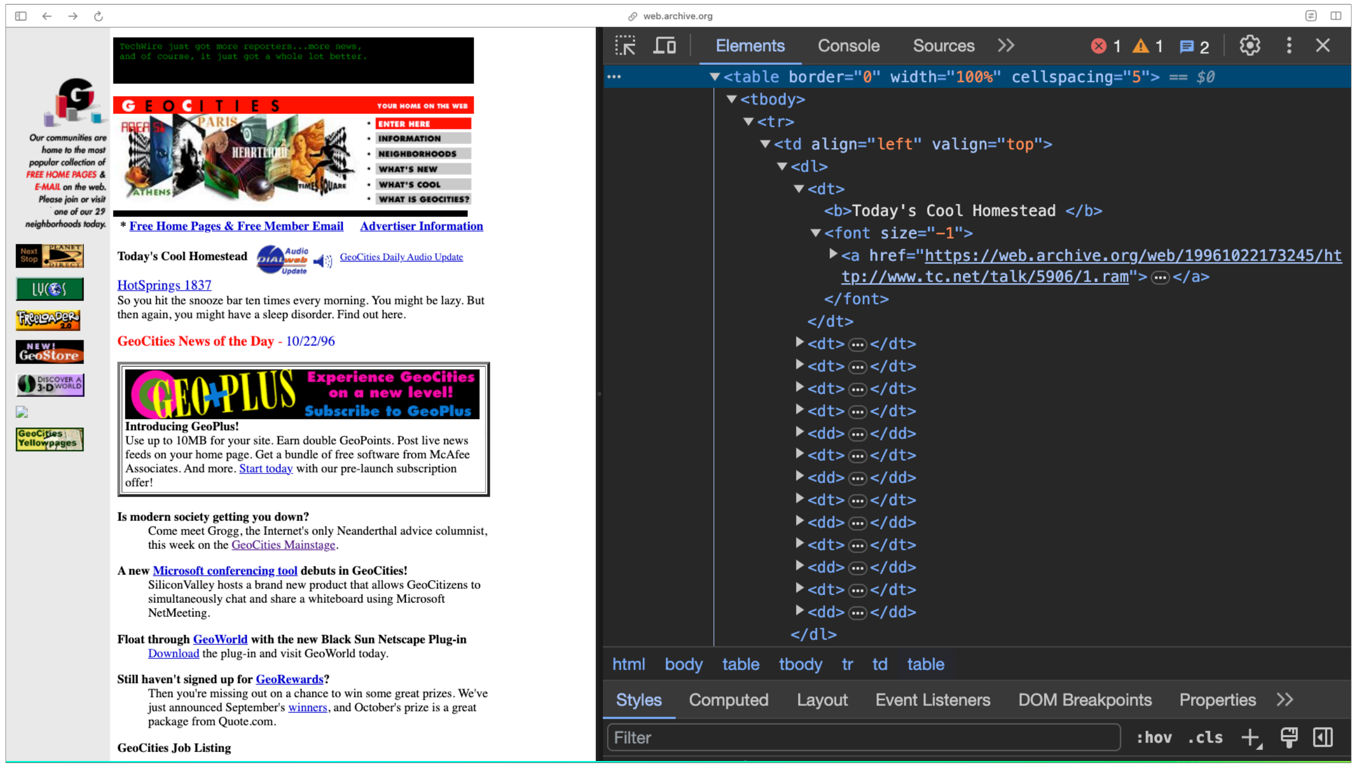
HotSprings 1837
So you hit the snooze bar ten times every morning. You might be lazy. But then again, you might have a sleep disorder. Find out here.

GeoCities News of the Day - 10/22/96

Introducing GeoPlus!
Use up to 10MB for your site. Earn double GeoPoints. Post live news feeds on your home page. Get a bundle of free software from McAfee Associates. And more. Start today with our pre-launch subscription offer!

Is modern society getting you down?
Come meet Grogg, the Internet's only Neanderthal advice columnist, this week on the GeoCities Mainstage.

First home page of Geocities from 22nd October, 1996
Left aligned



```
TechWire just got more reporters...more news,
and of course, it just got a whole lot better.
```

Our communities are home to the most popular collection of FREE HOME PAGES & E-MAIL on the web. Please join or visit one of our 29 neighborhoods today.

* Free Home Pages & Free Member Email   Advertiser Information

Today's Cool Homestead GeoCities Daily Audio Update

HotSprings 1837
So you hit the snooze bar ten times every morning. You might be lazy. But then again, you might have a sleep disorder. Find out here.

GeoCities News of the Day - 10/22/96

Introducing GeoPlus!
Use up to 10MB for your site. Earn double GeoPoints. Post live news feeds on your home page. Get a bundle of free software from McAfee Associates. And more. Start today with our pre-launch subscription offer!

Is modern society getting you down?
Come meet Grogg, the Internet's only Neanderthal advice columnist, this week on the GeoCities Mainstage.

A new Microsoft conferencing tool debuts in GeoCities!
SiliconValley hosts a brand new product that allows GeoCitizens to simultaneously chat and share a whiteboard using Microsoft NetMeeting.

Float through GeoWorld with the new Black Sun Netscape Plug-in
Download the plug-in and visit GeoWorld today.

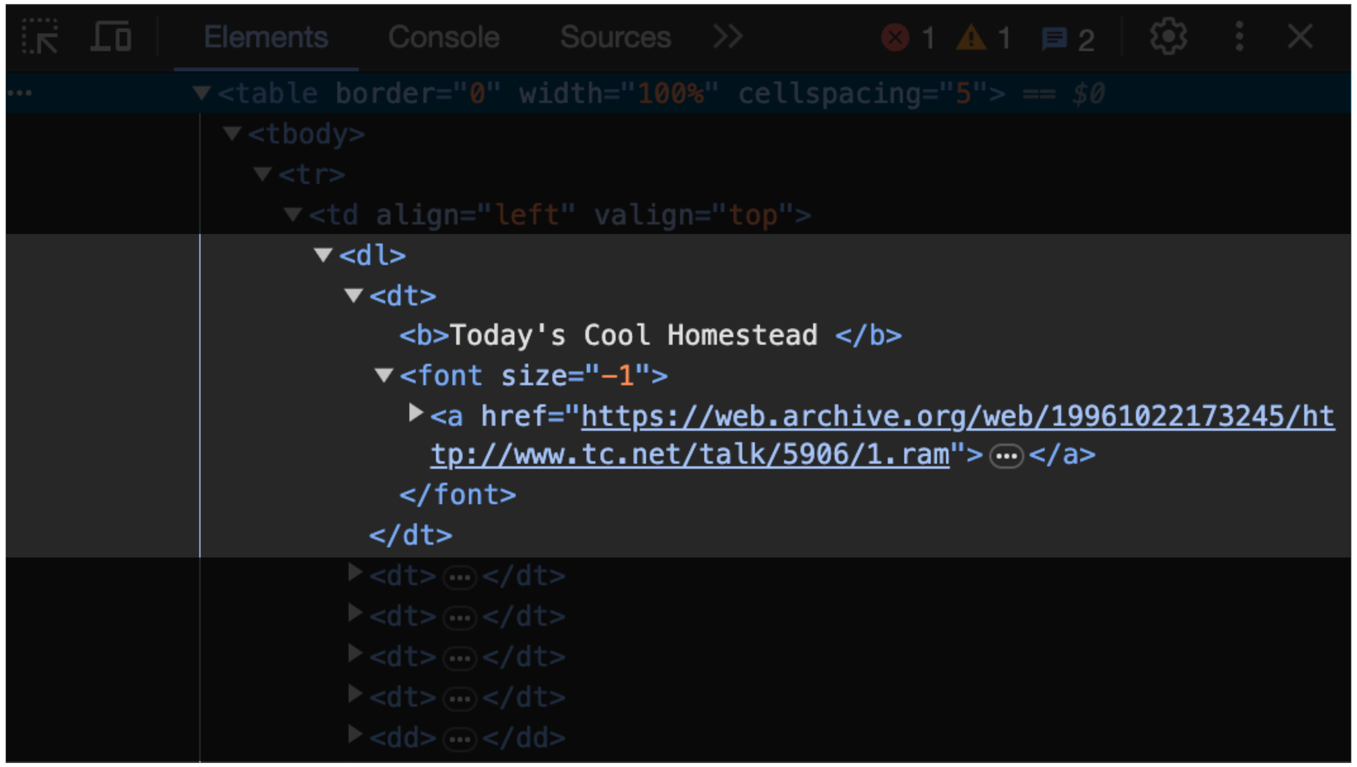
Still haven't signed up for GeoRewards?
Then you're missing out on a chance to win some great prizes. We've just announced September's winners, and October's prize is a great package from Quote.com.

GeoCities Job Listing

```
▼<table border="0" width="100%" cellspacing="5"> == $0
  ▼<tbody>
    ▼<tr>
      ▼<td align="left" valign="top">
        ▼<dl>
          ▼<dt>
              <b>Today's Cool Homestead </b>
            ▼<font size="-1">
              ▶<a href="https://web.archive.org/web/19961022173245/ht
                tp://www.tc.net/talk/5906/1.ram">…</a>
              </font>
            </dt>
          ▶<dt>…</dt>
          ▶<dt>…</dt>
          ▶<dt>…</dt>
          ▶<dt>…</dt>
          ▶<dd>…</dd>
          ▶<dt>…</dt>
          ▶<dd>…</dd>
          ▶<dt>…</dt>
          ▶<dd>…</dd>
          ▶<dt>…</dt>
          ▶<dd>…</dd>
          ▶<dt>…</dt>
          ▶<dd>…</dd>
          </dl>
```

Shout out to table-based layouts, the OG CSS grid

And also check out the semantic HTML



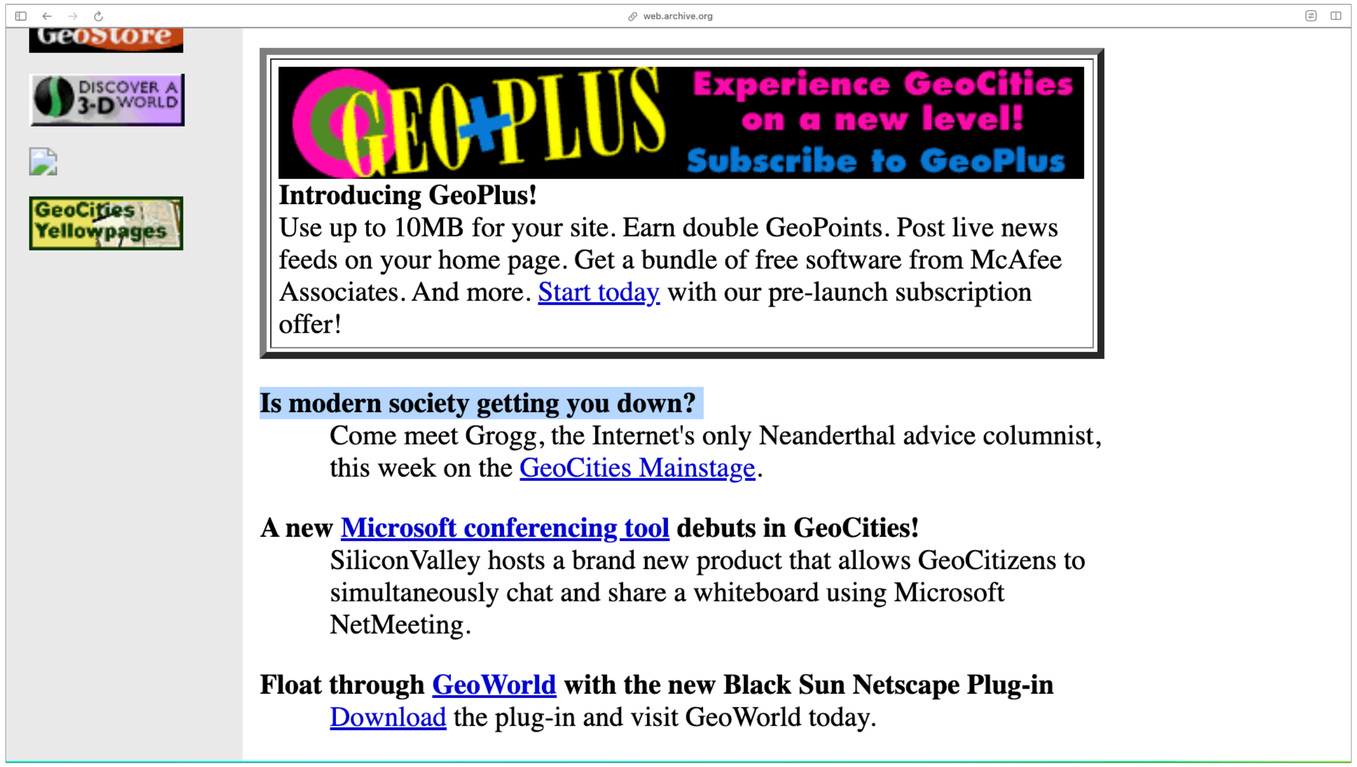
```
▼<table border="0" width="100%" cellspacing="5"> == $0
  ▼<tbody>
    ▼<tr>
      ▼<td align="left" valign="top">
        ▼<dl>
          ▼<dt>
              <b>Today's Cool Homestead </b>
            ▼<font size="-1">
              ▶<a href="https://web.archive.org/web/19961022173245/ht
                tp://www.tc.net/talk/5906/1.ram">…</a>
              </font>
            </dt>
          ▶<dt>…</dt>
          ▶<dt>…</dt>
          ▶<dt>…</dt>
          ▶<dt>…</dt>
          ▶<dd>…</dd>
```

<dl> description list
<dt> description term

Although… the description term isn't followed by a <dd> description definition element… sooo what's up with that Geocities!

Semantic HTML is a spicy topic isn't it. We're still fighting about what's semantic, what's not, who's getting it wrong. Who's going to call out who on Twitter today for attaching a click event to a div that should have been an anchor tag.

Nothing ever changes…



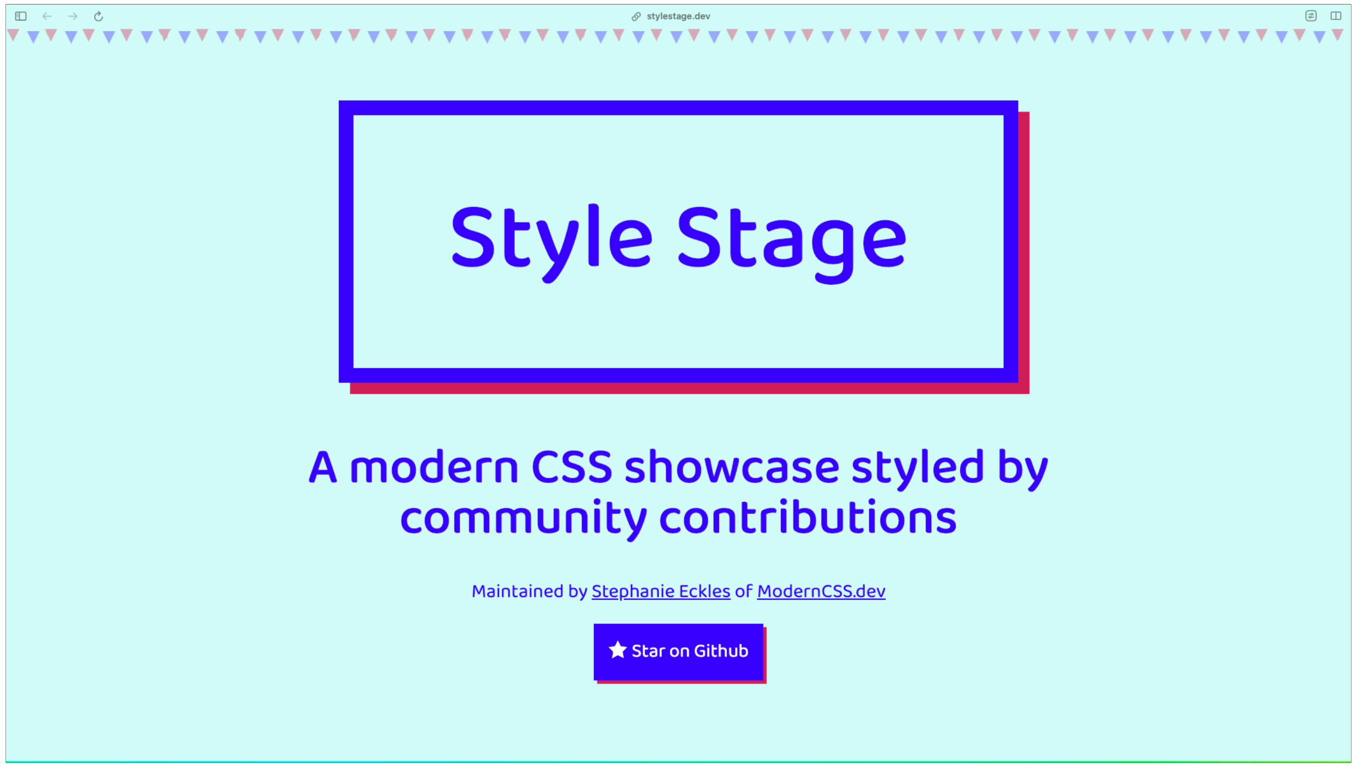
Introducing GeoPlus!
Use up to 10MB for your site. Earn double GeoPoints. Post live news feeds on your home page. Get a bundle of free software from McAfee Associates. And more. Start today with our pre-launch subscription offer!

Is modern society getting you down?
Come meet Grogg, the Internet's only Neanderthal advice columnist, this week on the GeoCities Mainstage.

A new Microsoft conferencing tool debuts in GeoCities!
SiliconValley hosts a brand new product that allows GeoCitizens to simultaneously chat and share a whiteboard using Microsoft NetMeeting.

Float through GeoWorld with the new Black Sun Netscape Plug-in
Download the plug-in and visit GeoWorld today.

Including how modern society is still getting us down.



stylestage.dev

Style Stage

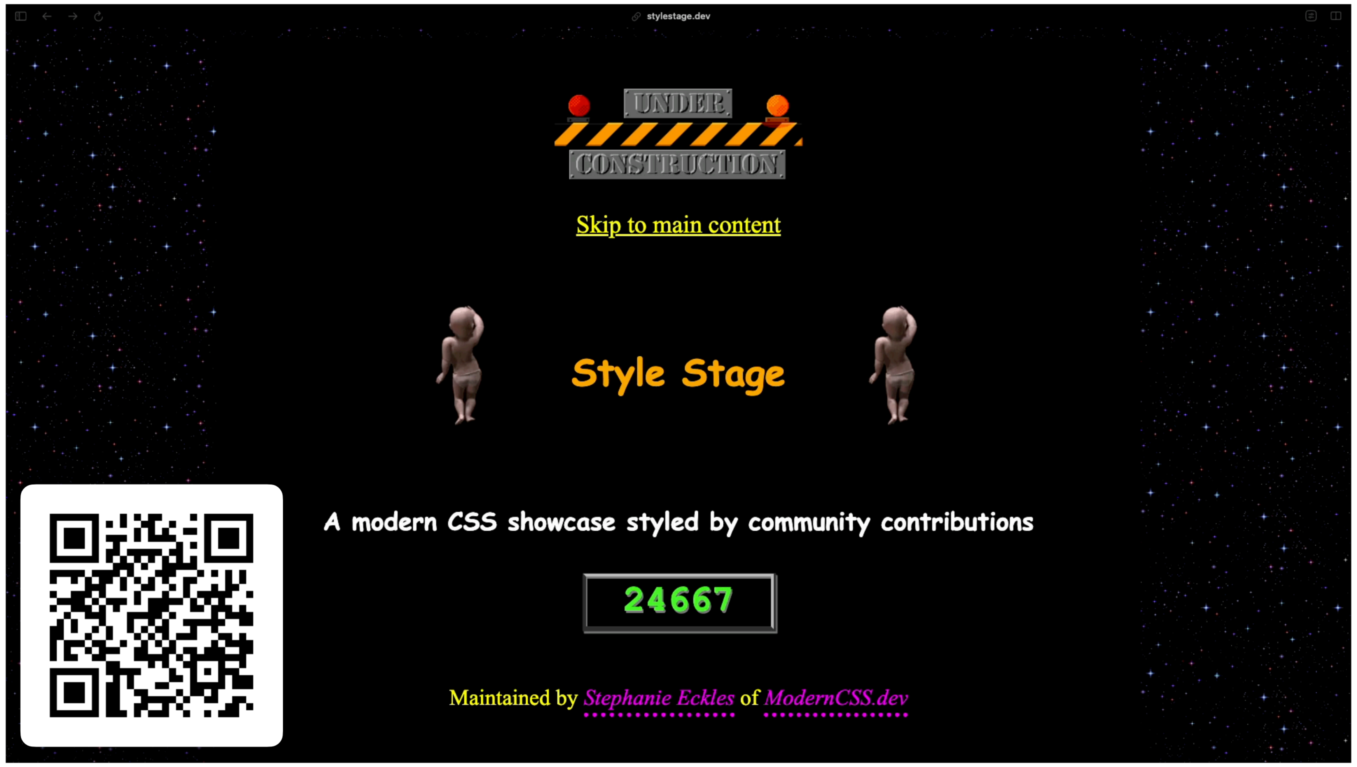
A modern CSS showcase styled by community contributions

Maintained by Stephanie Eckles of ModernCSS.dev

This is the style stage project.
Stylestage is all about showcasing the capabilities of modern CSS.

Your challenge is to completely change the look and feel of _this HTML page_ with CSS only. So it forces you to be really creative, and learn more about CSS in the process.

And I wanted to make style stage look like a Geocities website.



stylestage.dev

UNDER CONSTRUCTION

Skip to main content

Style Stage

A modern CSS showcase styled by community contributions

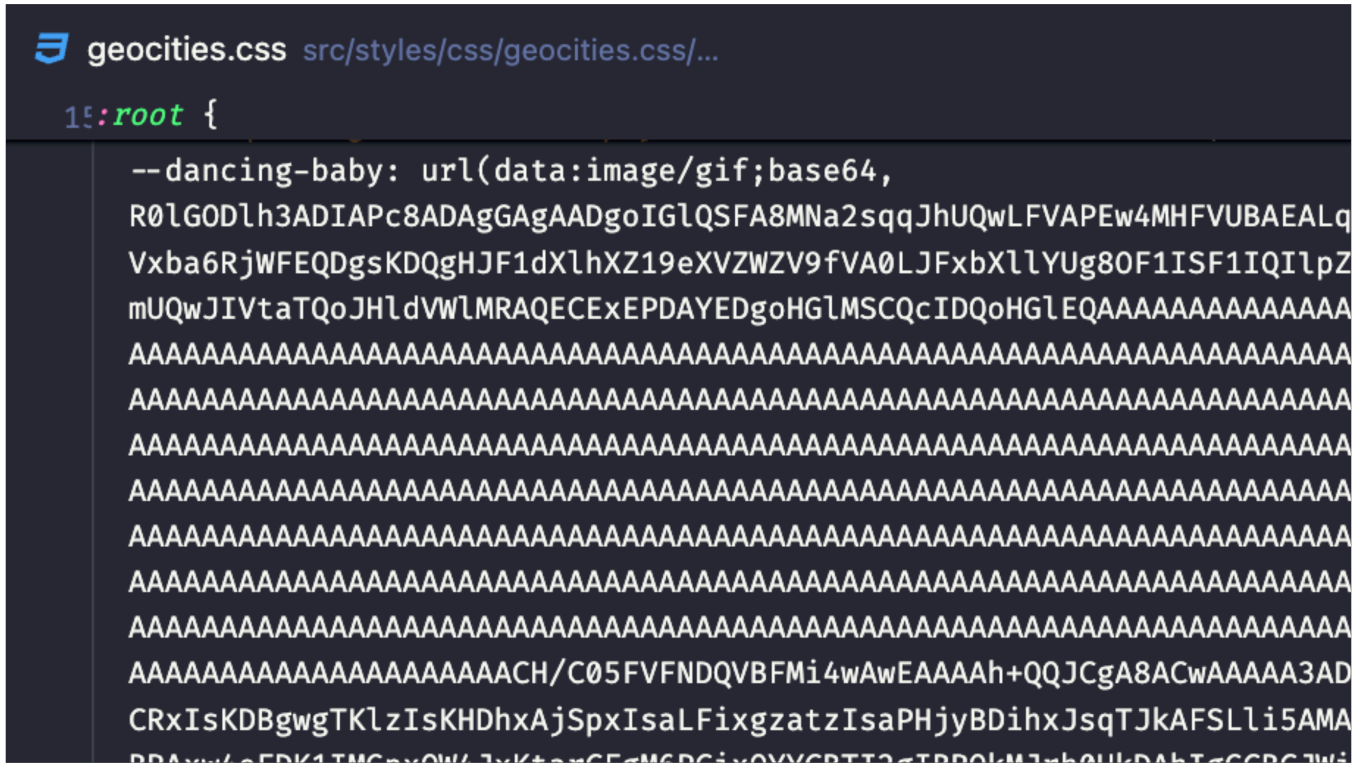
24667

Maintained by *Stephanie Eckles* of *ModernCSS.dev*

There’s a lot going on here.

[click] Scan the QR code if you want to check it out on your phone — it’s responsive! (Not very authentic but I couldn’t NOT make it responsive.)

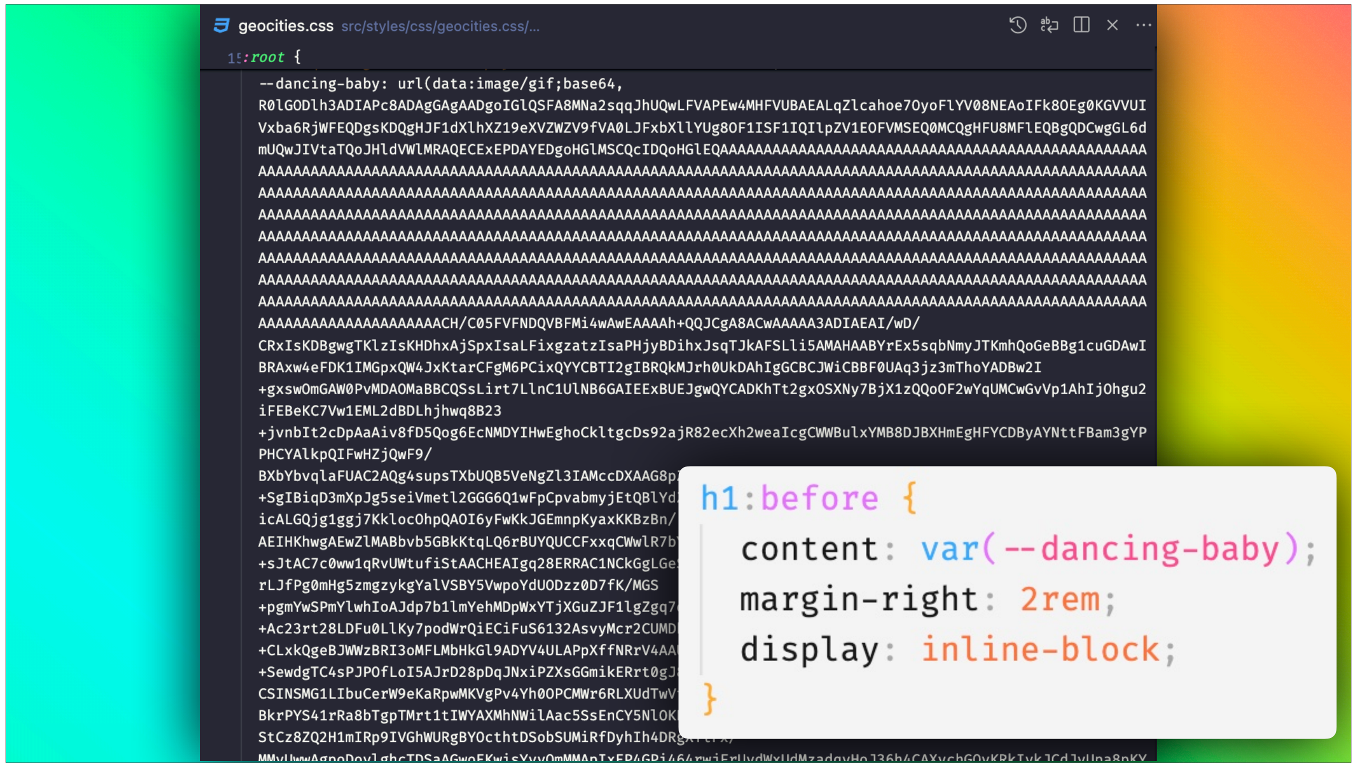
I know what you’re thinking.
Salma! How did you pull off getting those iconic dancing baby gifs in there without editing the HTML?
Well I’m glad you asked!



geocities.css src/styles/css/geocities.css/...

```
:root {
  --dancing-baby: url(data:image/gif;base64,
  R0lGODlh3ADIAPc8ADAgGAgAADgoIGlQSFA8MNa2sqqJhUQwLFVAPEw4MHFVUBAELq
  Vxba6RjWFEQDgsKDQgHJF1dXlhXZ19eXVZWZV9fVA0LJFxbXllYUg8OF1ISF1IQIlpZ
  mUQwJIVtaTQoJHldVWlMRAQECExEPDAYEDgoHGlMSCQcIDQoHGlEQAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAACH/C05FVFNDQVBFMi4wAwEAAAAh+QQJCgA8ACwAAAAA3AD
  CRxIsKDBgwgTKlzIsKHDhxAjSpxIsaLFixgzatzIsaPHjyBDihxJsqTJkAFSLli5AMA
```

Base 64 encoded data URLs.
Added as pseudo elements to existing HTML elements on the page.

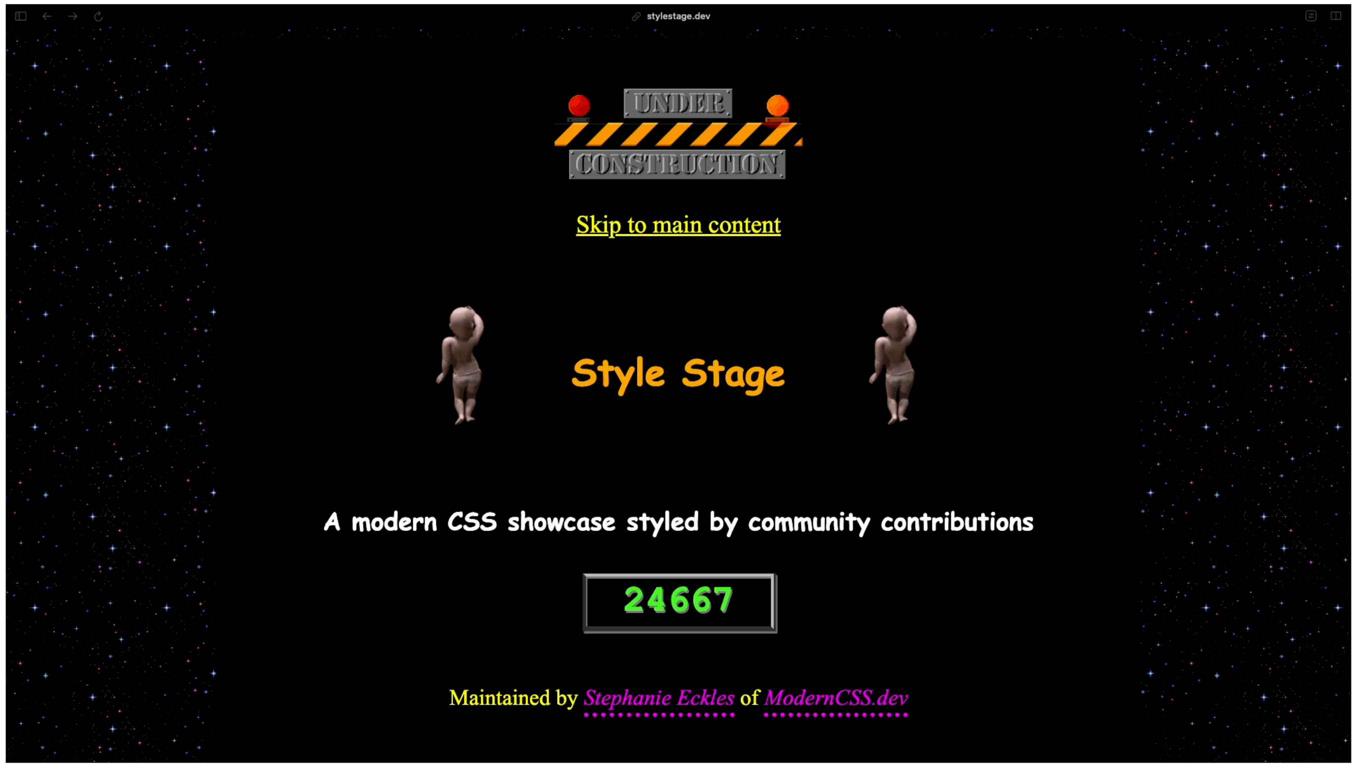


geocities.css src/styles/css/geocities.css/...

```css
:root {
  --dancing-baby: url(data:image/gif;base64,
  R0lGODlh3ADIAPc8ADAgGAgAADgoIGlQSFA8MNa2sqqJhUQwLFVAPEw4MHFVUBAEALqZlcahoe7OyoFlYV08NEAoIFk8OEg0KGVVUI
  Vxba6RjWFEQDgsKDQgHJF1dXlhXZ19eXVZWZV9fVA0LJFxbXllYUg8OF1ISF1IQIlpZV1EOFVMSEQ0MCQgHFU8MFlEQBgQDCwgGL6d
  mUQwJIVtaTQoJHldVWlMRAQECExEPDAYEDgoHGlMSCQcIDQoHGlEQAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAAA
  AAAAAAAAAAAAAAAAAAAAACH/C05FVFNDQVBFMi4wAwEAAAAh+QQJCgA8ACwAAAAA3ADIAEAI/wD/
  CRxIsKDBgwgTKlzIsKHDhxAjSpxIsaLFixgzatzIsaPHjyBDihxJsqTJkAFSLli5AMAHAABYrEx5sqbNmyJTKmhQoGeBBg1cuGDAwI
  BRAxw4eFDK1IMGpxQW4JxKtarCFgM6PCixQYYCBTI2gIBRQkMJrh0UkDAhIgGCBCJWiCBBF0UAq3jz3mThoYADBw2I
  +gxswOmGAW0PvMDAOMaBBCQSsLirt7LlnC1UlNB6GAIEExBUEJgwQYCADKhTt2gxOSXNy7BjX1zQQoOF2wYqUMCwGvVp1AhIjOhgu2
  iFEBeKC7Vw1EML2dBDLhjhwq8B23
  +jvnbIt2cDpAaAiv8fD5Qog6EcNMDYIHwEghoCkltgcDs92ajR82ecXh2weaIcgCWWBulxYMB8DJBXHmEgHFYCDByAYNttFBam3gYP
  PHCYAlkpQIFwHZjQwF9/
  BXbYbvqlaFUAC2AQg4supsTXbUQB5VeNgZl3IAMccDXAAG8p
  +SgIBiqD3mXpJg5seiVmetl2GGG6Q1wFpCpvabmyjEtQBlYd
  icALGQjg1ggj7KklocOhpQAOI6yFwKkJGEmnpKyaxKKBzBn/
  AEIHKhwgAEwZlMABbvb5GBkKtqLQ6rBUYQUCCFxxqCWwlR7b
  +sJtAC7c0ww1qRvUWtufiStAACHEAIgq28ERRAC1NCkGgLGe
  rLJfPg0mHg5zmgzykgYalVSBY5VwpoYdUODzz0D7fK/MGS
  +pgmYwSPmYlwhIoAJdp7b1lmYehMDpWxYTjXGuZJF1lgZgq7
  +Ac23rt28LDFu0LlKy7podWrQiECiFuS6132AsvyMcr2CUMD
  +CLxkQgeBJWWzBRI3oMFLMbHkGl9ADYV4ULAPpXffNRrV4AA
  +SewdgTC4sPJPOfLoI5AJrD28pDqJNxiPZXsGGmikERrt0gJ
  CSINSMG1LIbuCerW9eKaRpwMKVgPv4Yh0OPCMWr6RLXUdTwV
  BkrPYS41rRa8bTgpTMrt1tIWYAXMhNWilAac5SsEnCY5NlOK
  StCz8ZQ2H1mIRp9IVGhWURgBYOcthtDSobSUMiRfDyhIh4DRg
```

```css
h1:before {
  content: var(--dancing-baby);
  margin-right: 2rem;
  display: inline-block;
}
```

And assigned to CSS custom properties to be able to use those data urls easily [click] (because those strings are looonng)



Skip to main content

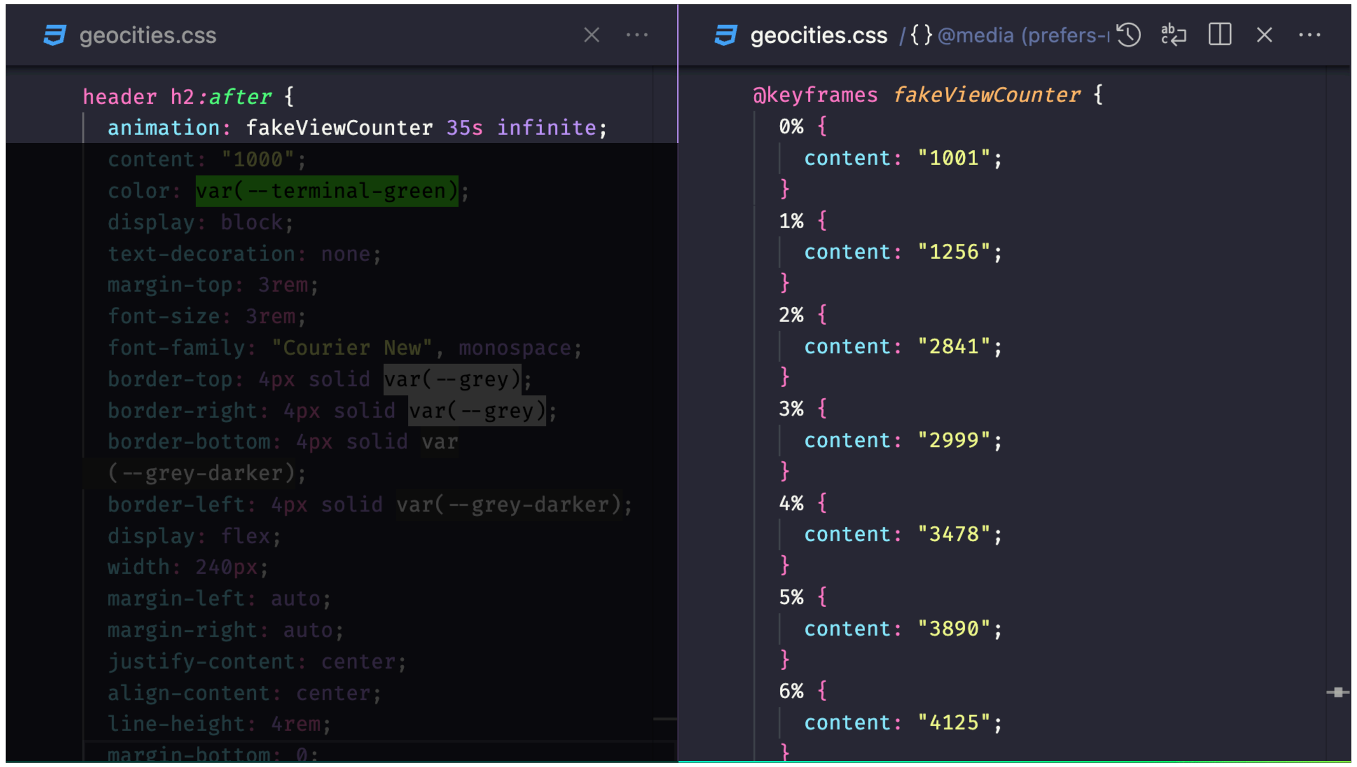
Style Stage

A modern CSS showcase styled by community contributions

24667

Maintained by *Stephanie Eckles* of *ModernCSS.dev*

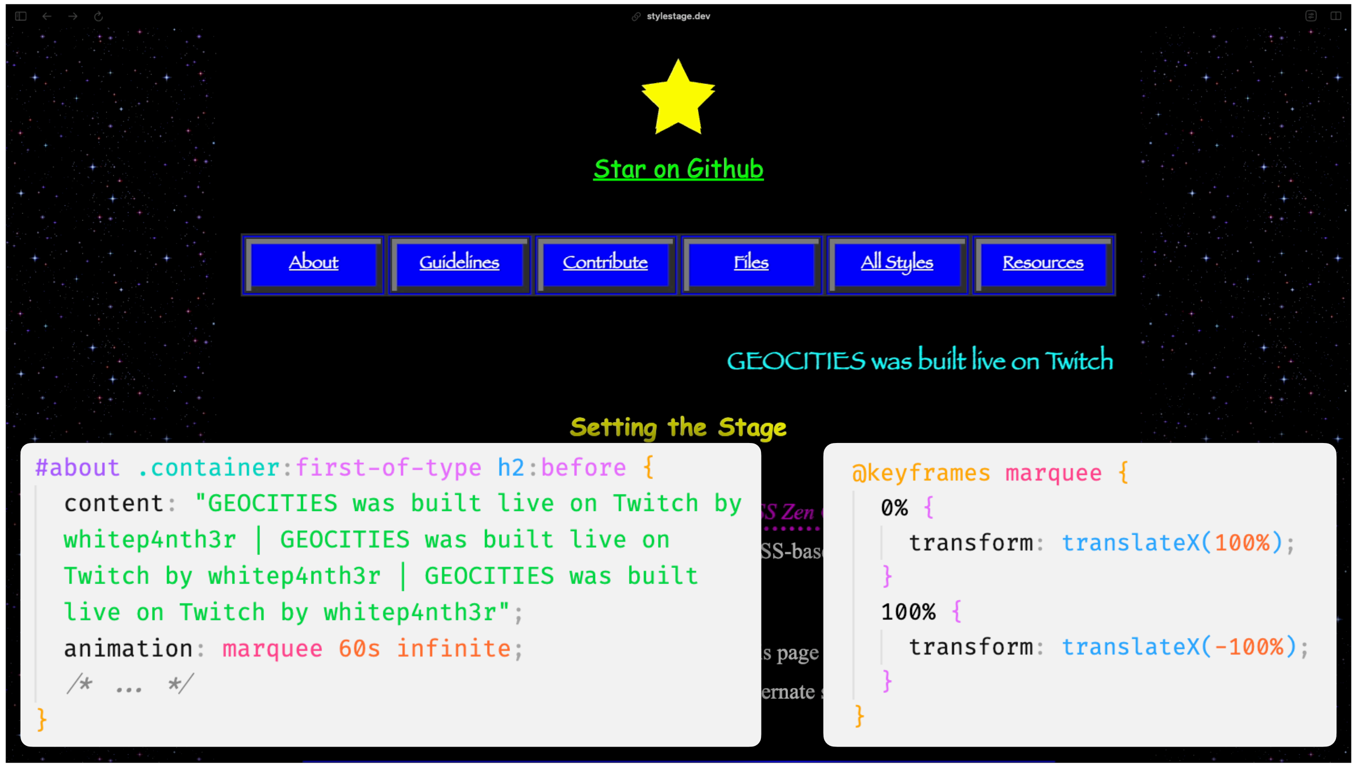
The (fake) visitor counter is also powered by a CSS animated pseudo element attached to an h2 element



```
header h2:after {
  animation: fakeViewCounter 35s infinite;
  content: "1000";
  color: var(--terminal-green);
  display: block;
  text-decoration: none;
  margin-top: 3rem;
  font-size: 3rem;
  font-family: "Courier New", monospace;
  border-top: 4px solid var(--grey);
  border-right: 4px solid var(--grey);
  border-bottom: 4px solid var
  (--grey-darker);
  border-left: 4px solid var(--grey-darker);
  display: flex;
  width: 240px;
  margin-left: auto;
  margin-right: auto;
  justify-content: center;
  align-content: center;
  line-height: 4rem;
  margin-bottom: 0;
```

```
@keyframes fakeViewCounter {
  0% {
    content: "1001";
  }
  1% {
    content: "1256";
  }
  2% {
    content: "2841";
  }
  3% {
    content: "2999";
  }
  4% {
    content: "3478";
  }
  5% {
    content: "3890";
  }
  6% {
    content: "4125";
  }
```

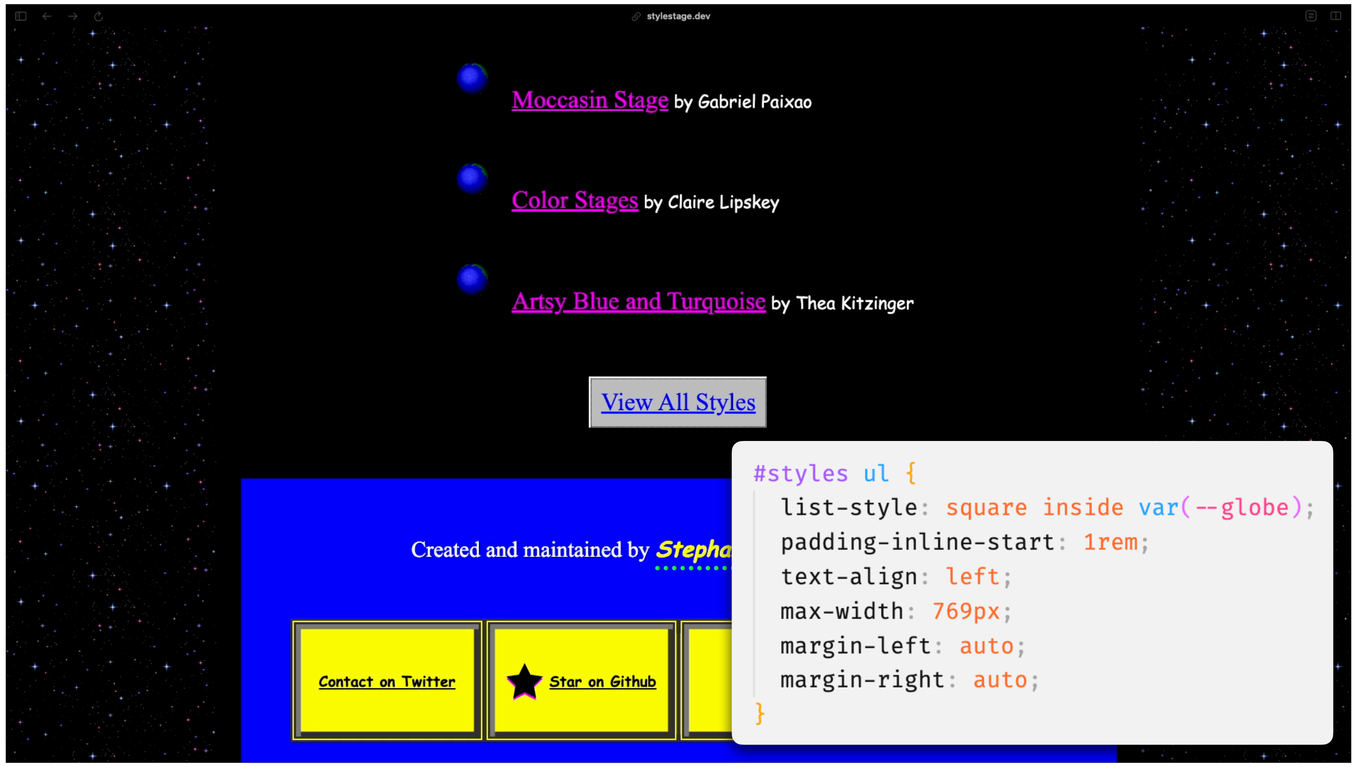
And yes, I did manually specify 100 keyframes, much to the frustration of my viewers as they were forced to watch me do it.



```css
#about .container:first-of-type h2:before {
  content: "GEOCITIES was built live on Twitch by
  whitep4nth3r | GEOCITIES was built live on
  Twitch by whitep4nth3r | GEOCITIES was built
  live on Twitch by whitep4nth3r";
  animation: marquee 60s infinite;
  /* ... */
}
```

```css
@keyframes marquee {
  0% {
    transform: translateX(100%);
  }
  100% {
    transform: translateX(-100%);
  }
}
```

Other highlights include this stunning marquee using the papyrus font (added via another convoluted pseudo element)
[click] Powered by a CSS animation using transform translate



Moccasin Stage by Gabriel Paixao

Color Stages by Claire Lipskey

Artsy Blue and Turquoise by Thea Kitzinger

Created and maintained by Stepha

```css
#styles ul {
  list-style: square inside var(--globe);
  padding-inline-start: 1rem;
  text-align: left;
  max-width: 769px;
  margin-left: auto;
  margin-right: auto;
}
```

And these globes

[click] Which are powered by the CSS list-style property and a custom property of a data URL of another base64 encoded gif

I LEARNED LOADS

(AND SO DID MY VIEWERS)

I learned loads and so did my viewers.
And we all bonded over our love for the web of yesterday

Speaking of 90s internet nostalgia…

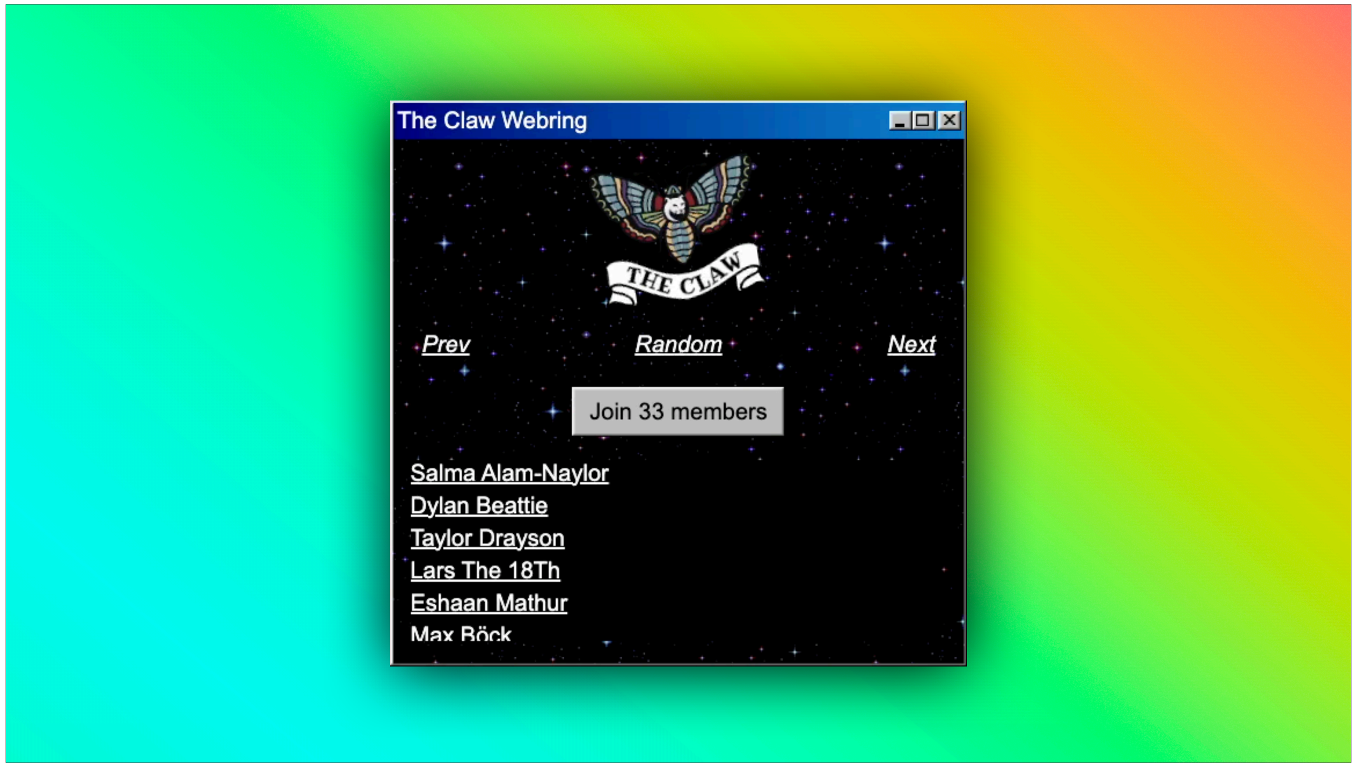
THE CLAW WEBRING

I made a webring!

A webring is a collection of websites linked together in a circular structure, usually organised around a specific theme such as technology.

They were popular in the 90s and early 2000s, particularly among amateur websites.

If you were a member of a webring, you'd display that on your website.



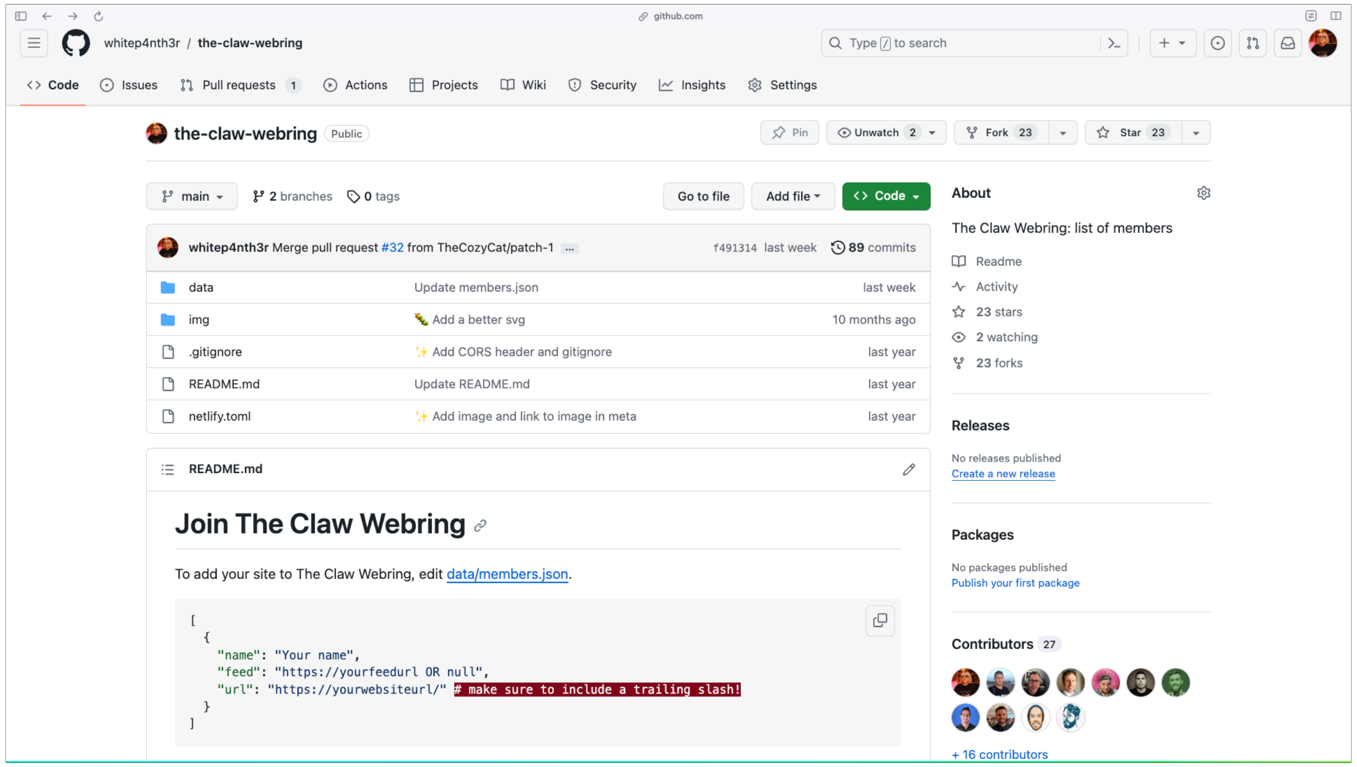
The Claw Webring

Salma Alam-Naylor
Dylan Beattie
Taylor Drayson
Lars The 18Th
Eshaan Mathur
Max Böck

The aim was to be as easy as possible to display the webring widget on any website built with any or no framework

and I wanted to make sure that updates — for example when new members joined the web ring — would be automatically pushed out to all sites without a redeploy.

So what you see here is a web component
Clicking join takes you to the GitHub repo, you can navigate back and forth between the ring, and you can click to go to a random site.



whitep4nth3r Merge pull request #32 from TheCozyCat/patch-1 … f491314 last week 89 commits

data	Update members.json	last week
img	Add a better svg	10 months ago
.gitignore	✨ Add CORS header and gitignore	last year
README.md	Update README.md	last year
netlify.toml	✨ Add image and link to image in meta	last year

README.md

Join The Claw Webring

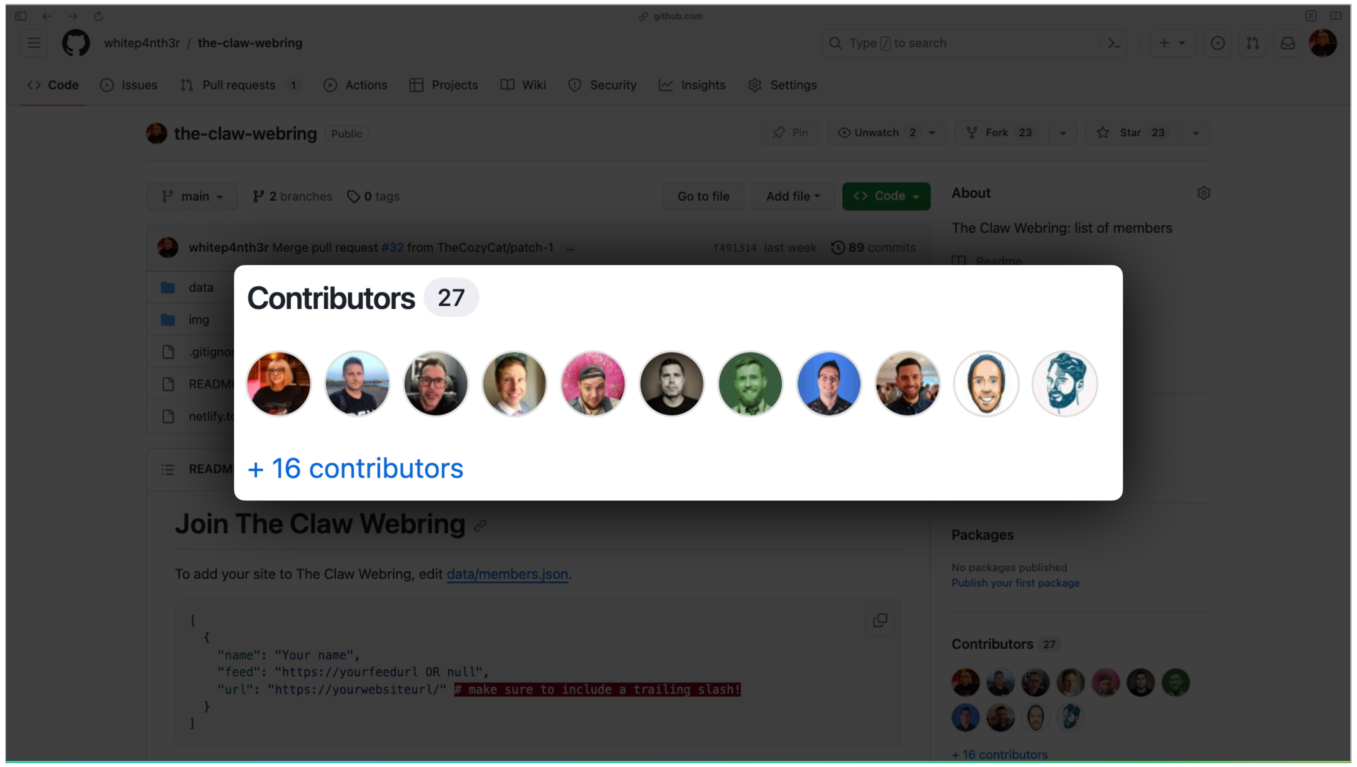
To add your site to The Claw Webring, edit data/members.json.

```
[
  {
    "name": "Your name",
    "feed": "https://yourfeedurl OR null",
    "url": "https://yourwebsiteurl/" # make sure to include a trailing slash!
  }
]
```

About

The Claw Webring: list of members

To add yourself to the webring
Open a PR to the claw webring repo
And add an entry to the members.json file with your name and site url



Contributors 27

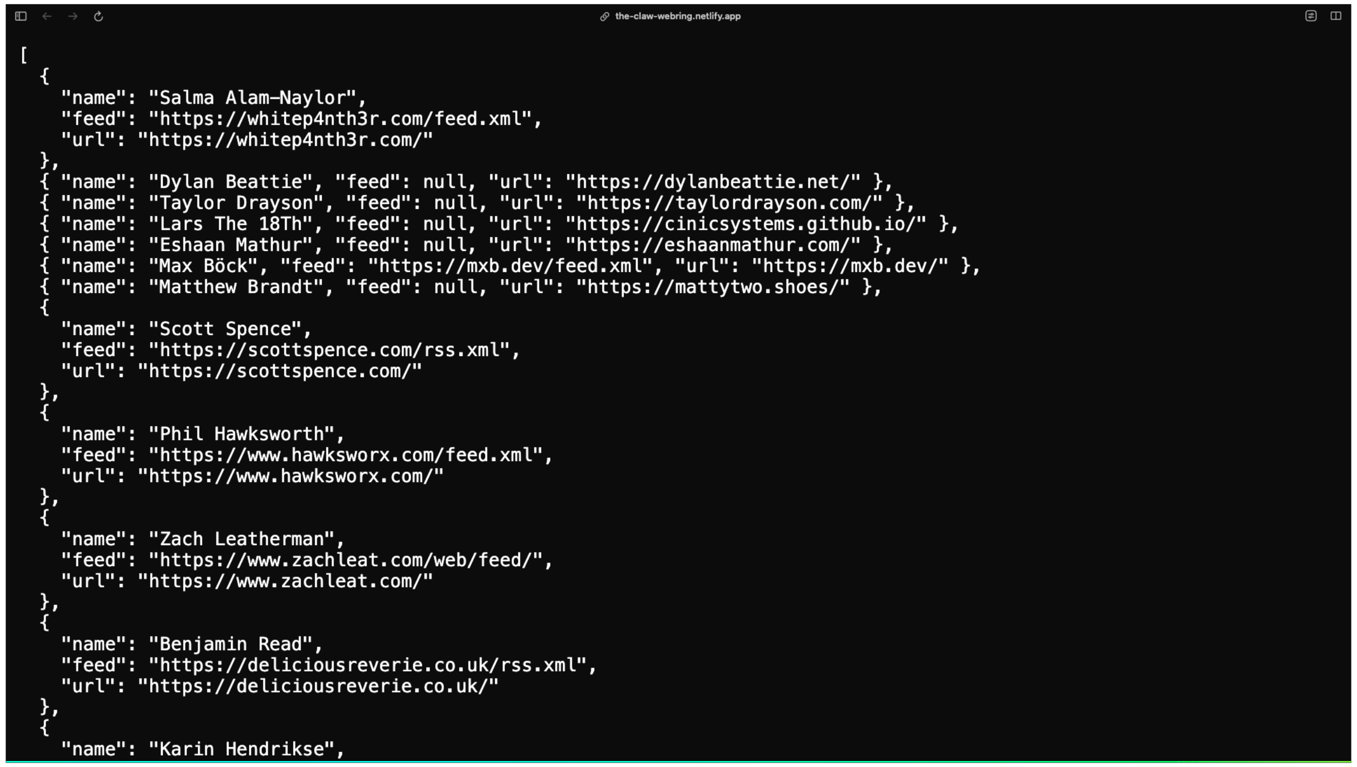
+ 16 contributors

Join The Claw Webring

To add your site to The Claw Webring, edit data/members.json.

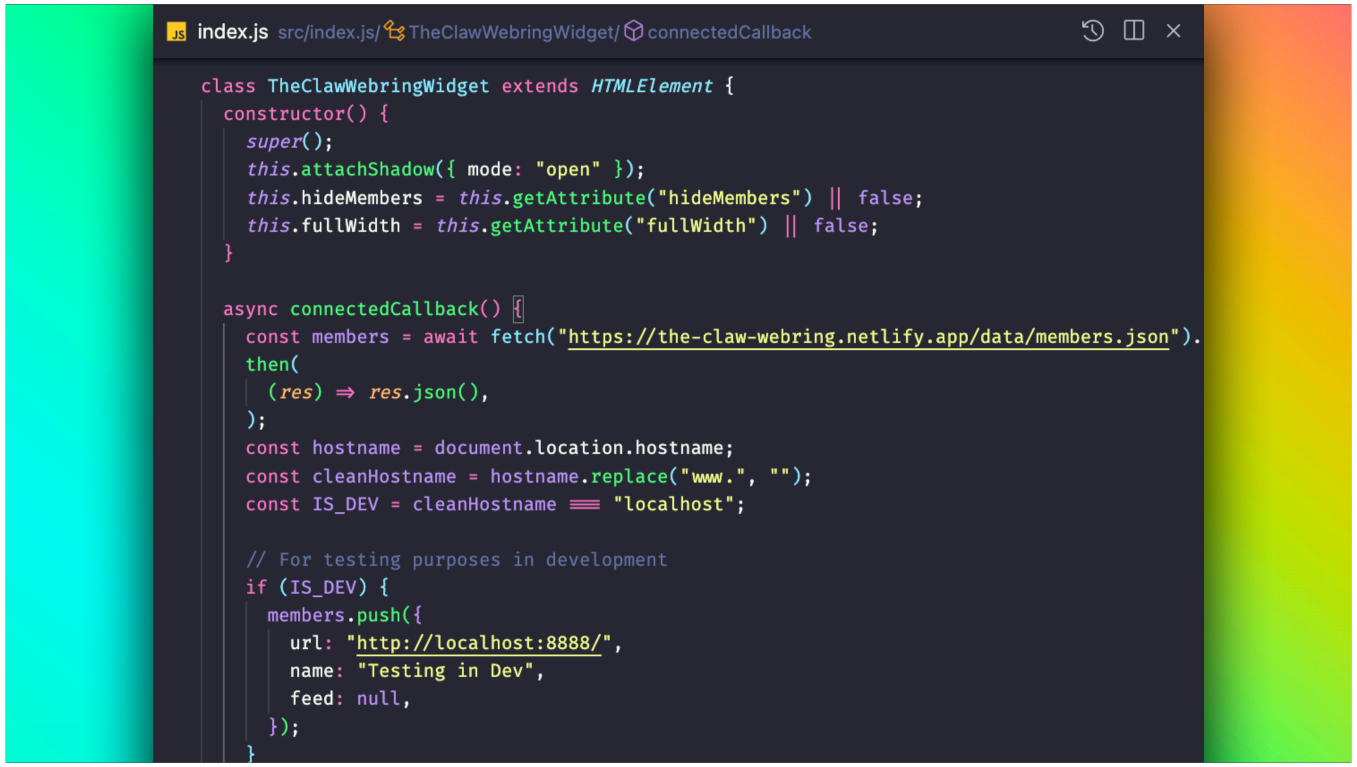
```
[
  {
    "name": "Your name",
    "feed": "https://yourfeedurl OR null",
    "url": "https://yourwebsiteurl/" # make sure to include a trailing slash!
  }
]
```

After the original release I had 26 more people add themselves to the webring (some of you are in the audience today!)
For some people this was their first open source contribution



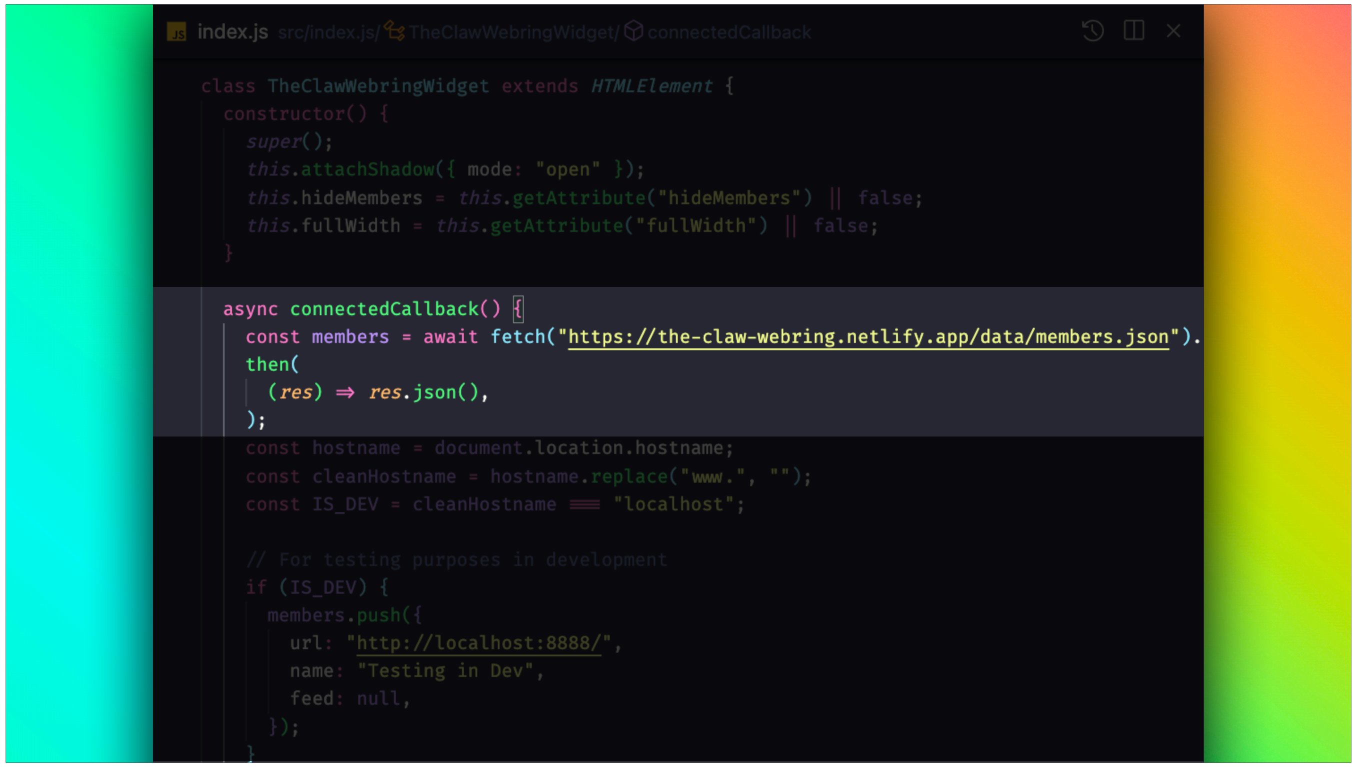
```
[
  {
    "name": "Salma Alam-Naylor",
    "feed": "https://whitep4nth3r.com/feed.xml",
    "url": "https://whitep4nth3r.com/"
  },
  { "name": "Dylan Beattie", "feed": null, "url": "https://dylanbeattie.net/" },
  { "name": "Taylor Drayson", "feed": null, "url": "https://taylordrayson.com/" },
  { "name": "Lars The 18Th", "feed": null, "url": "https://cinicsystems.github.io/" },
  { "name": "Eshaan Mathur", "feed": null, "url": "https://eshaanmathur.com/" },
  { "name": "Max Böck", "feed": "https://mxb.dev/feed.xml", "url": "https://mxb.dev/" },
  { "name": "Matthew Brandt", "feed": null, "url": "https://mattytwo.shoes/" },
  {
    "name": "Scott Spence",
    "feed": "https://scottspence.com/rss.xml",
    "url": "https://scottspence.com/"
  },
  {
    "name": "Phil Hawksworth",
    "feed": "https://www.hawksworx.com/feed.xml",
    "url": "https://www.hawksworx.com/"
  },
  {
    "name": "Zach Leatherman",
    "feed": "https://www.zachleat.com/web/feed/",
    "url": "https://www.zachleat.com/"
  },
  {
    "name": "Benjamin Read",
    "feed": "https://deliciousreverie.co.uk/rss.xml",
    "url": "https://deliciousreverie.co.uk/"
  },
  {
    "name": "Karin Hendrikse",
```

When your PR is merged, this will make your site data available on a public URL


index.js src/index.js/ TheClawWebringWidget/ connectedCallback

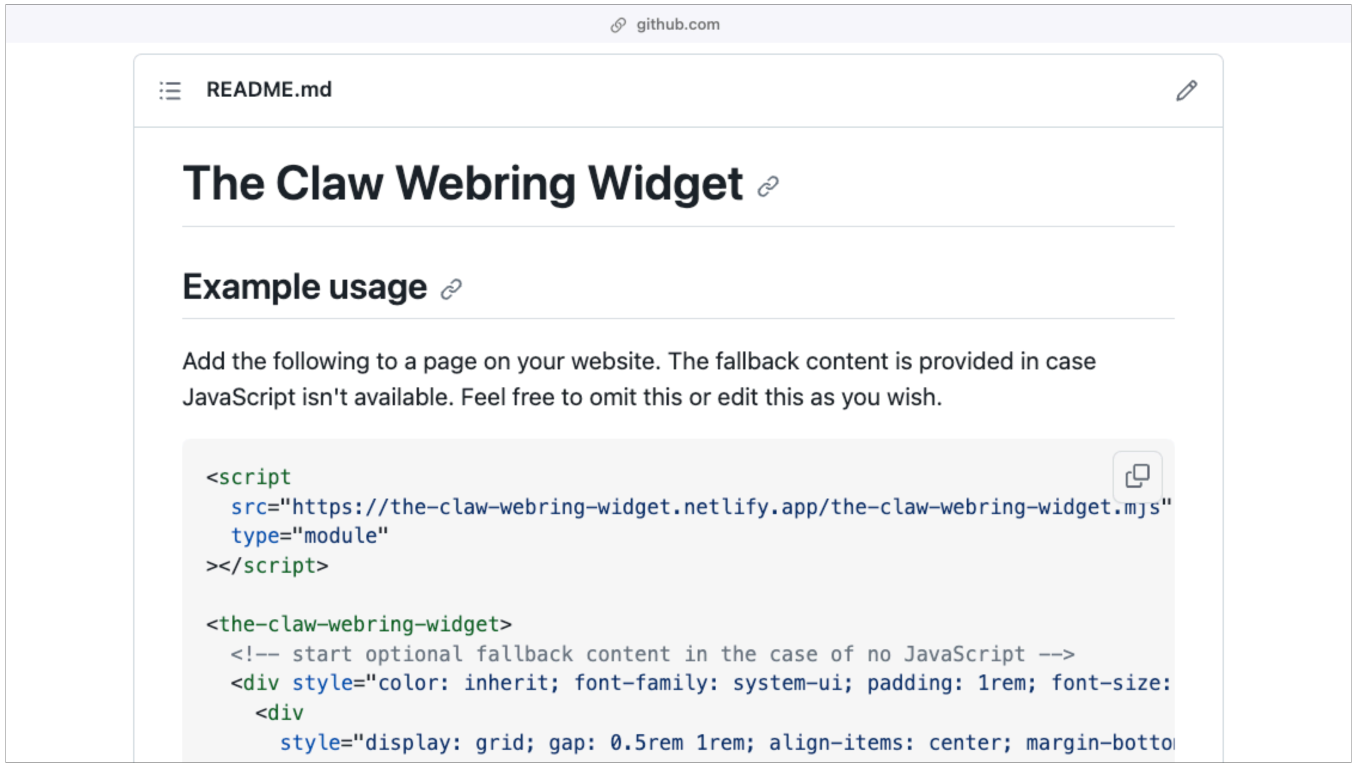
```
class TheClawWebringWidget extends HTMLElement {
  constructor() {
    super();
    this.attachShadow({ mode: "open" });
    this.hideMembers = this.getAttribute("hideMembers") || false;
    this.fullWidth = this.getAttribute("fullWidth") || false;
  }

  async connectedCallback() {
    const members = await fetch("https://the-claw-webring.netlify.app/data/members.json").
    then(
      (res) => res.json(),
    );
    const hostname = document.location.hostname;
    const cleanHostname = hostname.replace("www.", "");
    const IS_DEV = cleanHostname === "localhost";

    // For testing purposes in development
    if (IS_DEV) {
      members.push({
        url: "http://localhost:8888/",
        name: "Testing in Dev",
        feed: null,
      });
    }
```

Which is fetched by the web component at runtime


index.js src/index.js/ TheClawWebringWidget/ connectedCallback

```js
class TheClawWebringWidget extends HTMLElement {
  constructor() {
    super();
    this.attachShadow({ mode: "open" });
    this.hideMembers = this.getAttribute("hideMembers") || false;
    this.fullWidth = this.getAttribute("fullWidth") || false;
  }

  async connectedCallback() {
    const members = await fetch("https://the-claw-webring.netlify.app/data/members.json").
    then(
      (res) => res.json(),
    );
    const hostname = document.location.hostname;
    const cleanHostname = hostname.replace("www.", "");
    const IS_DEV = cleanHostname === "localhost";

    // For testing purposes in development
    if (IS_DEV) {
      members.push({
        url: "http://localhost:8888/",
        name: "Testing in Dev",
        feed: null,
      });
    }
```

Which means that every instance of the web component stays up to date



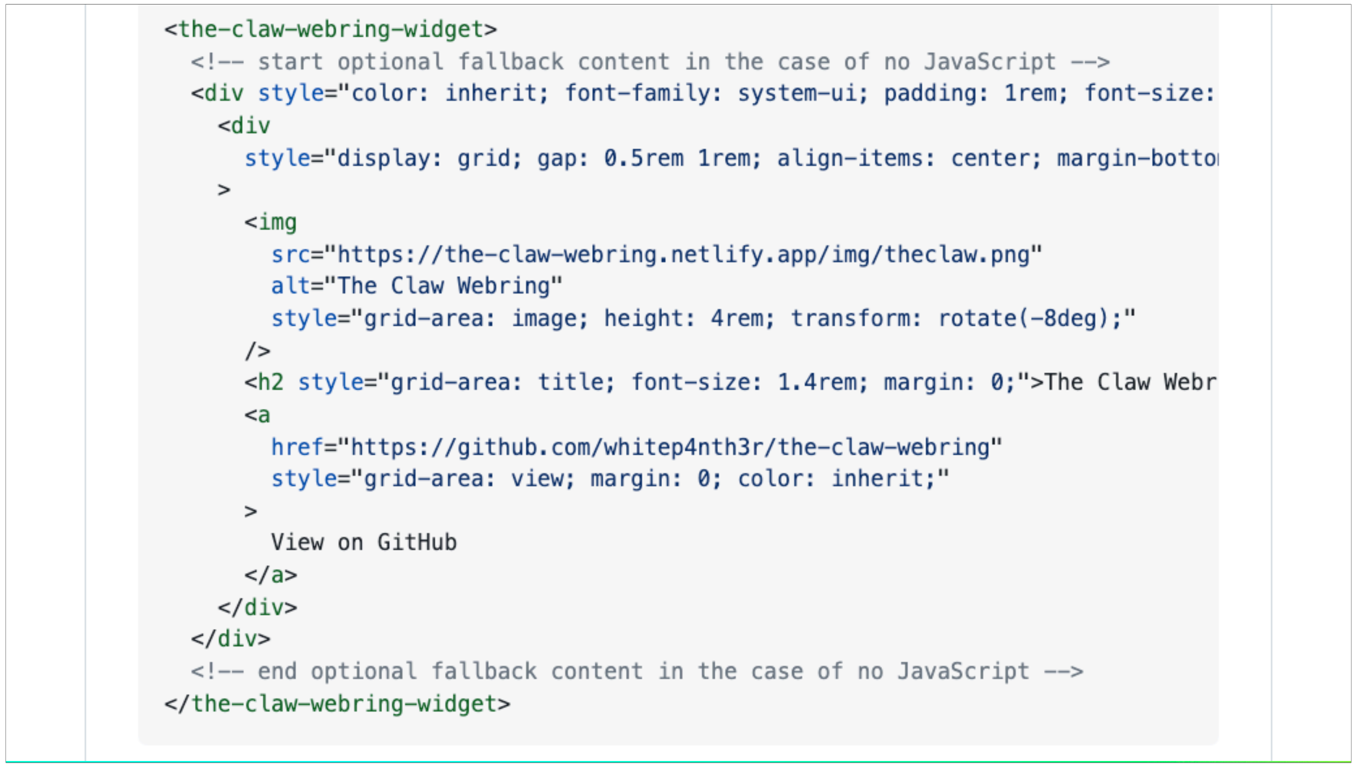
The Claw Webring Widget

Example usage

Add the following to a page on your website. The fallback content is provided in case JavaScript isn't available. Feel free to omit this or edit this as you wish.

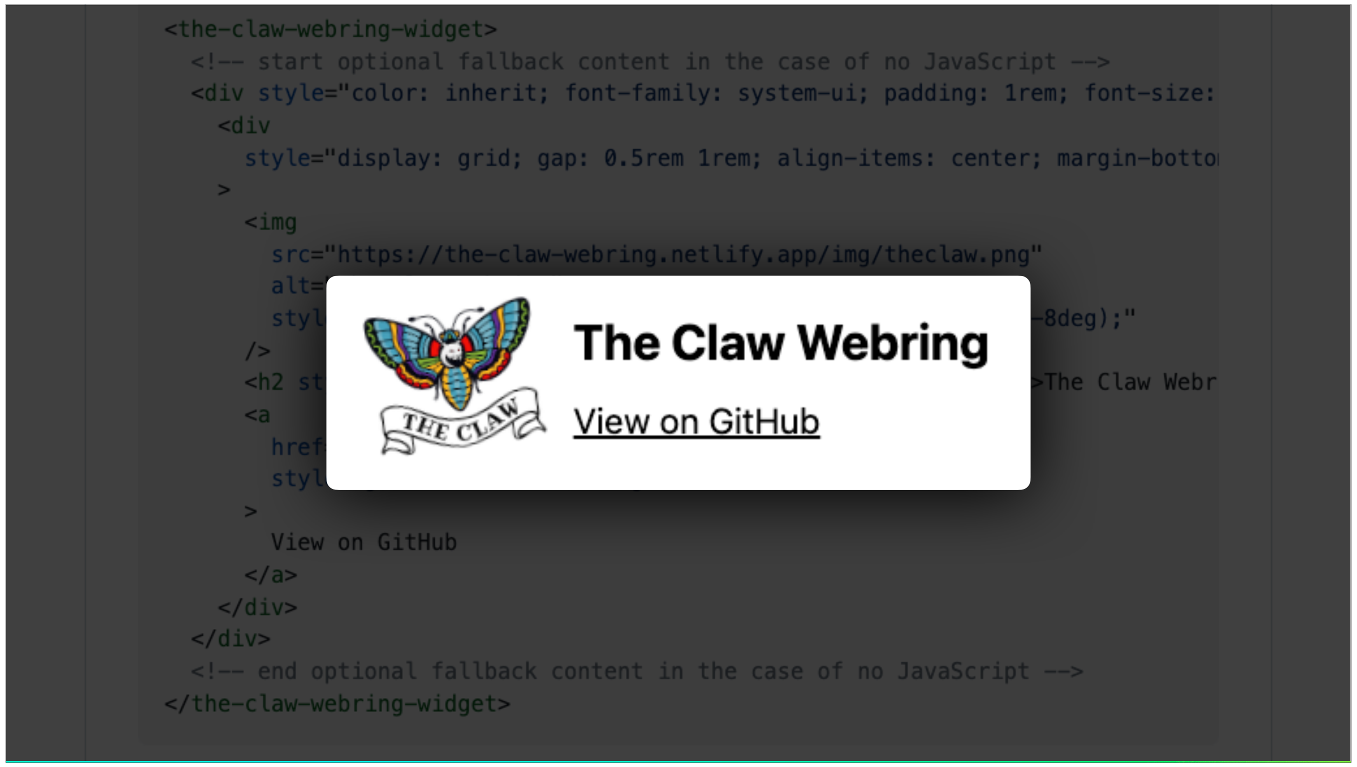
```
<script
  src="https://the-claw-webring-widget.netlify.app/the-claw-webring-widget.mjs"
  type="module"
></script>

<the-claw-webring-widget>
  <!-- start optional fallback content in the case of no JavaScript -->
  <div style="color: inherit; font-family: system-ui; padding: 1rem; font-size:
    <div
      style="display: grid; gap: 0.5rem 1rem; align-items: center; margin-botto
```

To show the webring widget on your site
Add the following script tag and custom element tag



```
<the-claw-webring-widget>
  <!-- start optional fallback content in the case of no JavaScript -->
  <div style="color: inherit; font-family: system-ui; padding: 1rem; font-size:
    <div
      style="display: grid; gap: 0.5rem 1rem; align-items: center; margin-botto
    >
      <img
        src="https://the-claw-webring.netlify.app/img/theclaw.png"
        alt="The Claw Webring"
        style="grid-area: image; height: 4rem; transform: rotate(-8deg);"
      />
      <h2 style="grid-area: title; font-size: 1.4rem; margin: 0;">The Claw Webr
      <a
        href="https://github.com/whitep4nth3r/the-claw-webring"
        style="grid-area: view; margin: 0; color: inherit;"
      >
        View on GitHub
      </a>
    </div>
  </div>
  <!-- end optional fallback content in the case of no JavaScript -->
</the-claw-webring-widget>
```

And you can include some optional fallback content in case JavaScript isn’t available or you don’t want to use JavaScript at all.



```
<the-claw-webring-widget>
  <!-- start optional fallback content in the case of no JavaScript -->
  <div style="color: inherit; font-family: system-ui; padding: 1rem; font-size:
    <div
      style="display: grid; gap: 0.5rem 1rem; align-items: center; margin-botto
    >
      <img
        src="https://the-claw-webring.netlify.app/img/theclaw.png"
        alt=
        styl                                                    -8deg);"
      />
      <h2 st                                                   >The Claw Webr
      <a
        href
        styl
      >
        View on GitHub
      </a>
    </div>
  </div>
  <!-- end optional fallback content in the case of no JavaScript -->
</the-claw-webring-widget>
```

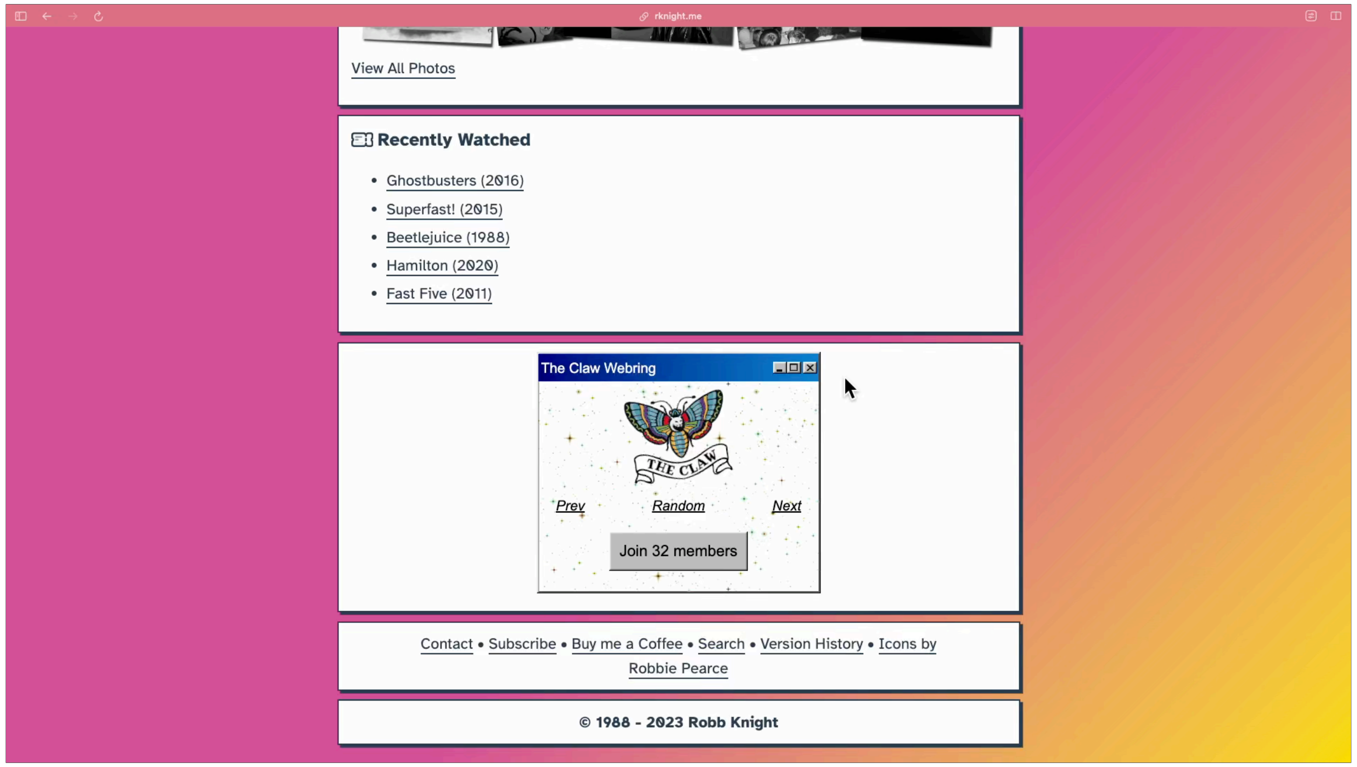
The Claw Webring

View on GitHub

Which makes the widget look like this

[click] PROGRESSIVE ENHANCEMENT

You can configure light and dark mode, whether or not to show the full scrollable member list



View All Photos

Recently Watched

- Ghostbusters (2016)
- Superfast! (2015)
- Beetlejuice (1988)
- Hamilton (2020)
- Fast Five (2011)

Contact • Subscribe • Buy me a Coffee • Search • Version History • Icons by Robbie Pearce

© 1988 - 2023 Robb Knight

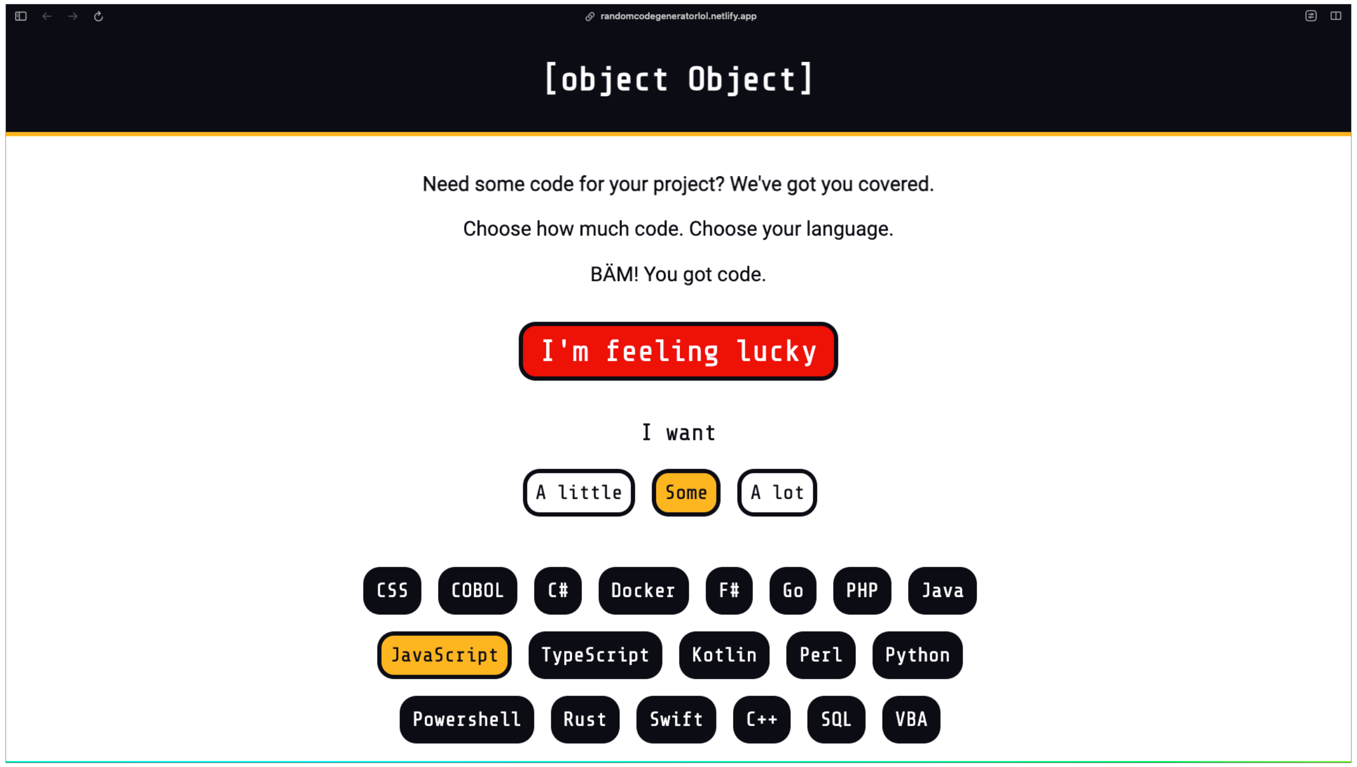
And there’s also a hidden easter egg, of course

And now for the biggest meme of my streaming career…

THE RANDOM CODE GENERATOR

I built a random code generator
And it was the funniest thing I ever did on stream

Now this is running the risk of NOT being funny today… and maybe you kind of had to be there
But we'll give it a try



randomcodegeneratorlol.netlify.app

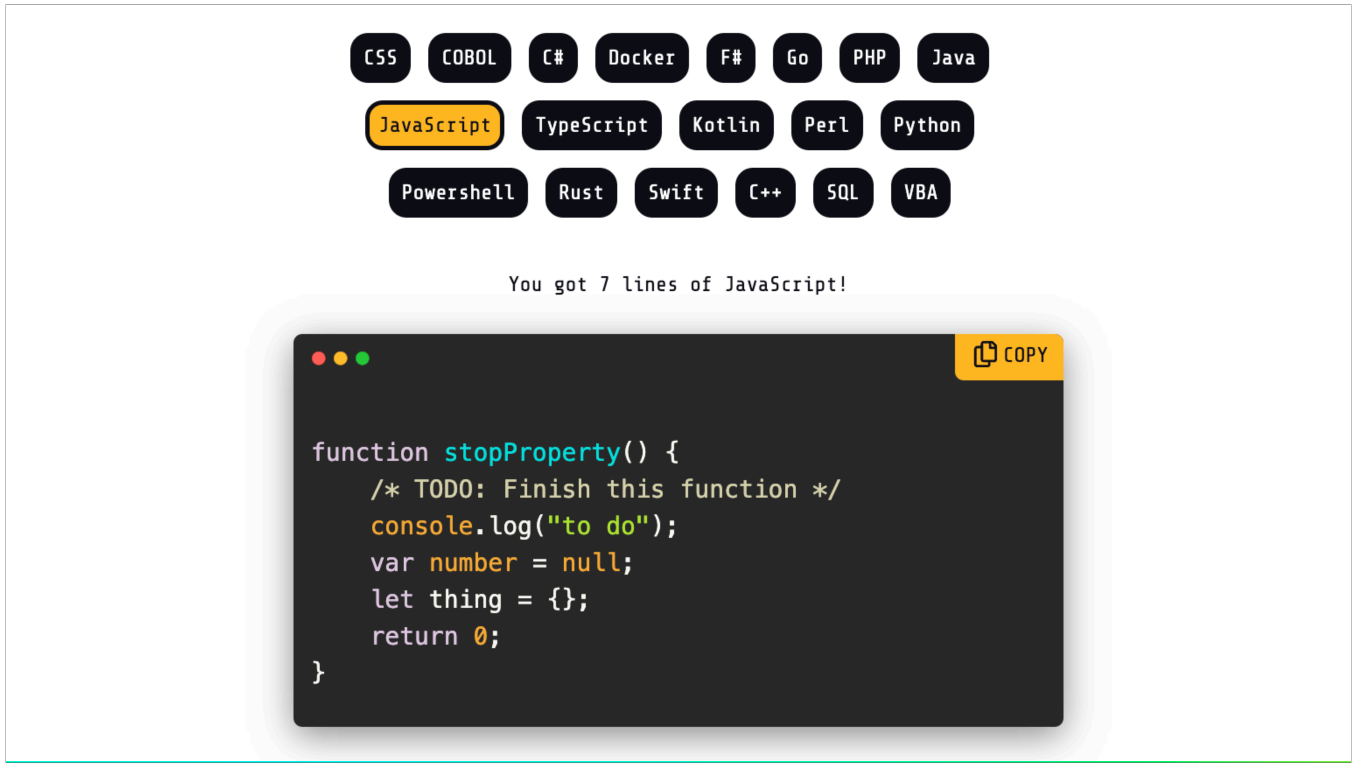
[object Object]

Need some code for your project? We've got you covered.

Choose how much code. Choose your language.

BÄM! You got code.

This is the random code generator
So, say I need some JavaScript…
I select the options, scroll down…



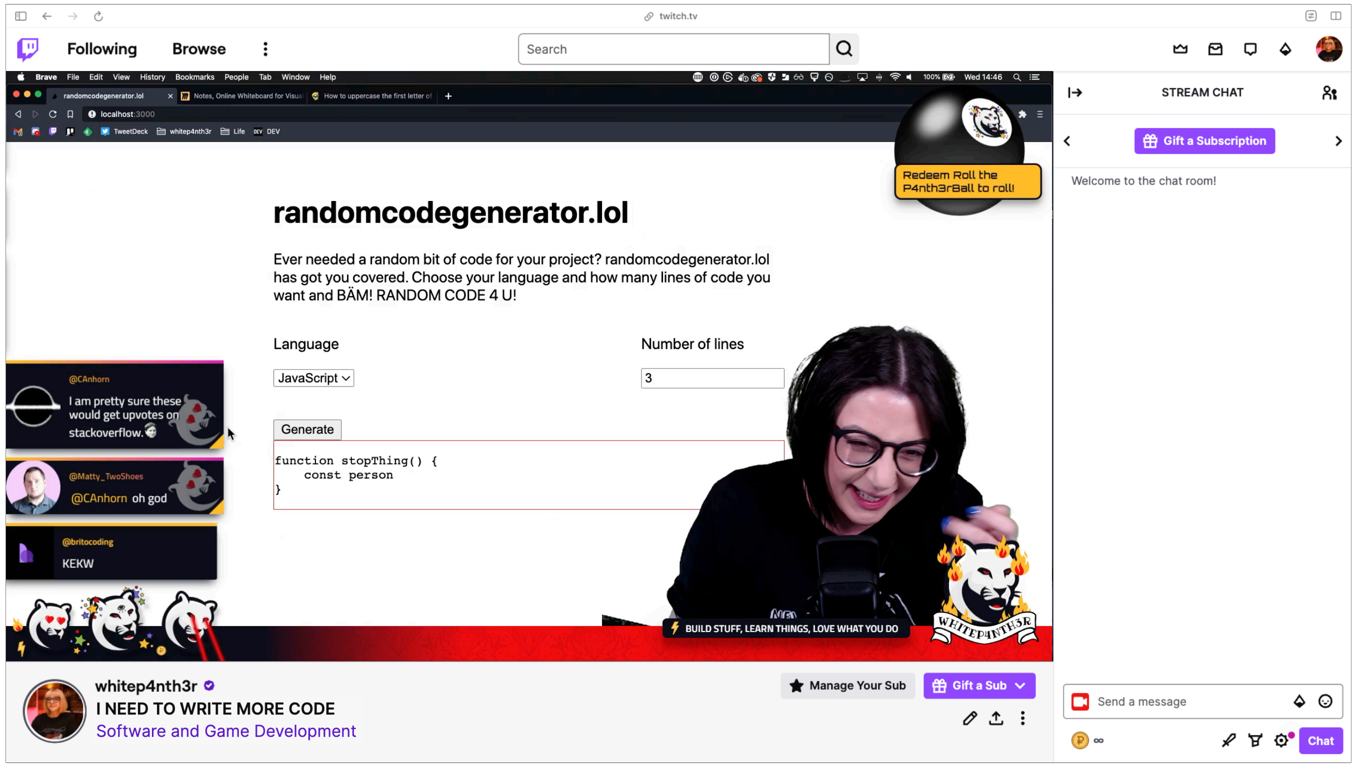
You got 7 lines of JavaScript!

```
function stopProperty() {
    /* TODO: Finish this function */
    console.log("to do");
    var number = null;
    let thing = {};
    return 0;
}
```

And bam, there it is.

Perfect JavaScript for you to copy and paste into your project.

This is how it started…



randomcodegenerator.lol

Ever needed a random bit of code for your project? randomcodegenerator.lol has got you covered. Choose your language and how many lines of code you want and BÄM! RANDOM CODE 4 U!

Language: JavaScript

Number of lines: 3

Generate

```
function stopThing() {
    const person
}
```

@CAnhorn: I am pretty sure these would get upvotes on stackoverflow.

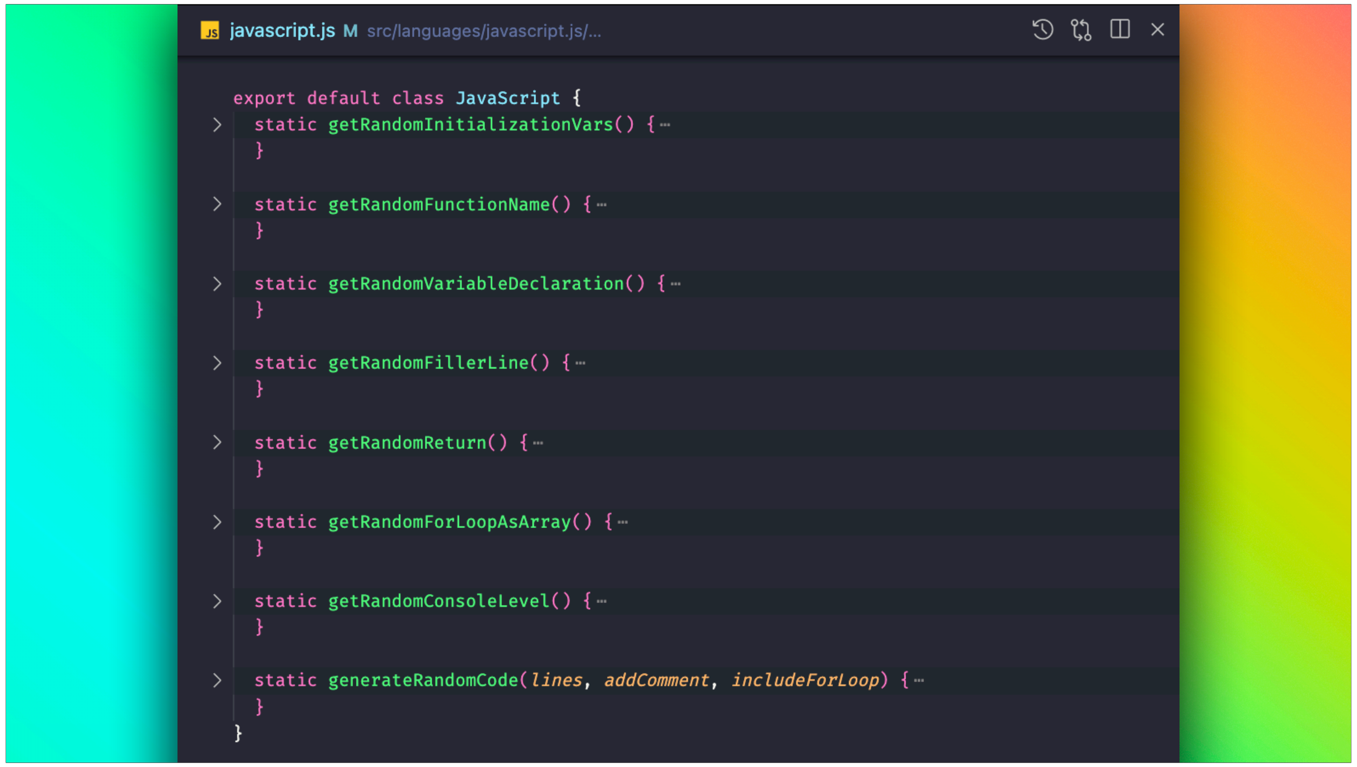
@Matty_TwoShoes: @CAnhorn oh god

@britocoding: KEKW

whitep4nth3r
I NEED TO WRITE MORE CODE
Software and Game Development

(Chat’s loving it)
(I’m dying)

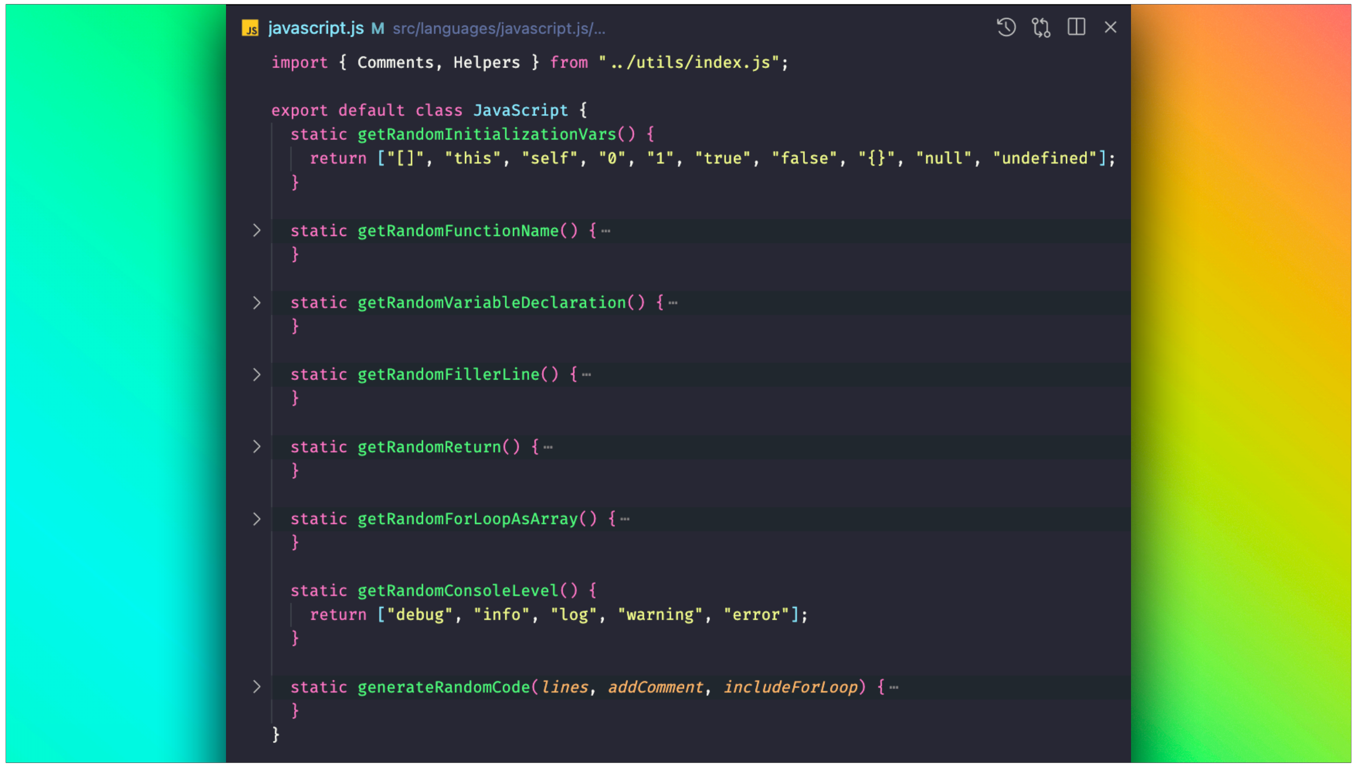
So how does this work… I don’t even’t know anymore



```
export default class JavaScript {
  static getRandomInitializationVars() {…
  }

  static getRandomFunctionName() {…
  }

  static getRandomVariableDeclaration() {…
  }

  static getRandomFillerLine() {…
  }

  static getRandomReturn() {…
  }

  static getRandomForLoopAsArray() {…
  }

  static getRandomConsoleLevel() {…
  }

  static generateRandomCode(lines, addComment, includeForLoop) {…
  }
}
```

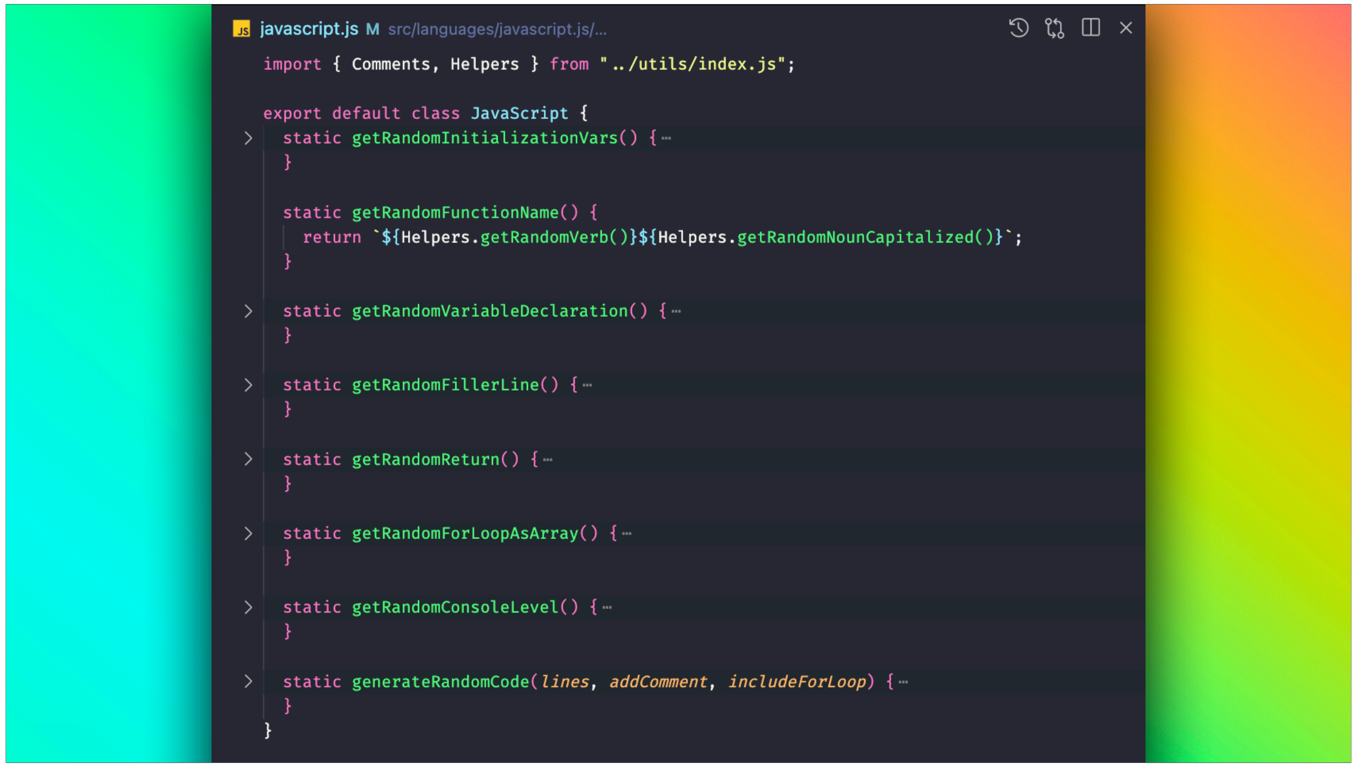
Here’s the JavaScript example…

A bunch of methods that return a “random” thing.



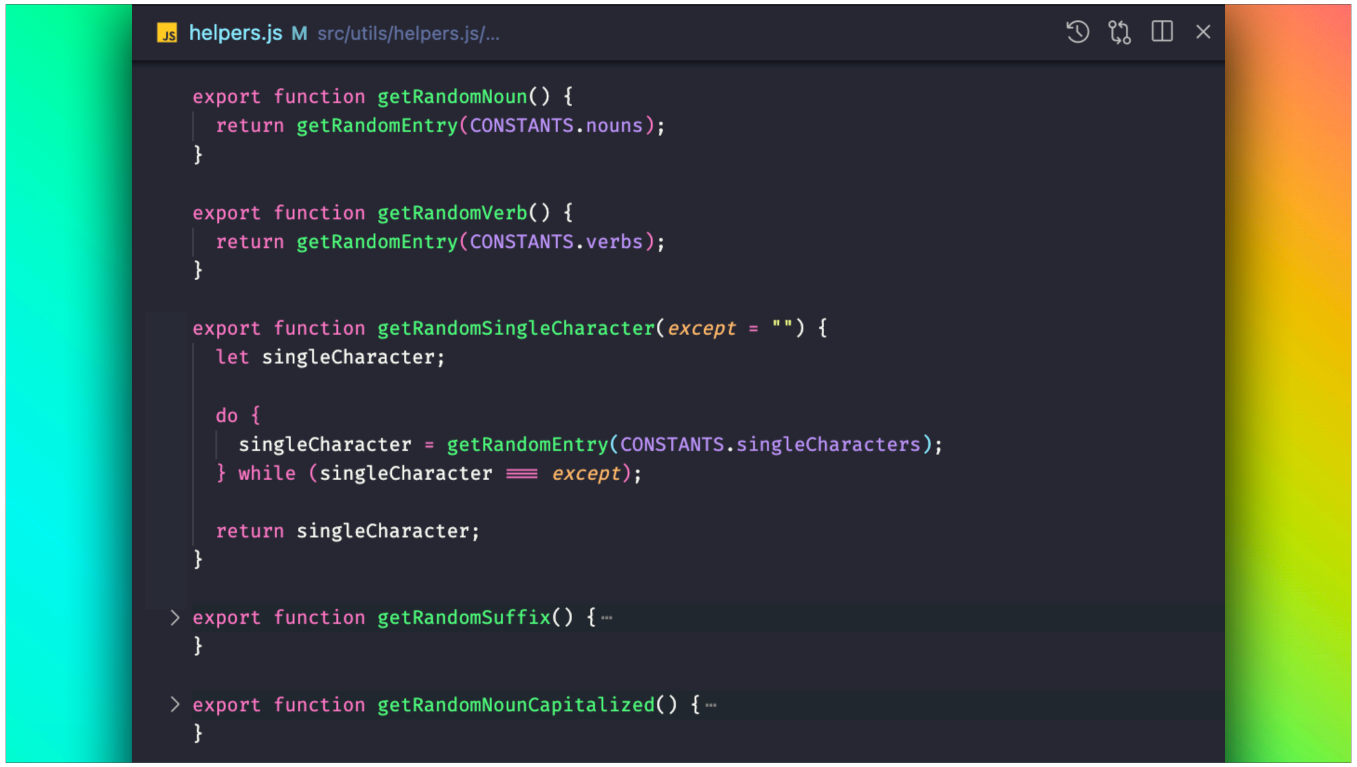
```
import { Comments, Helpers } from "../utils/index.js";

export default class JavaScript {
  static getRandomInitializationVars() {
    return ["[]", "this", "self", "0", "1", "true", "false", "{}", "null", "undefined"];
  }

  static getRandomFunctionName() {⋯
  }

  static getRandomVariableDeclaration() {⋯
  }

  static getRandomFillerLine() {⋯
  }

  static getRandomReturn() {⋯
  }

  static getRandomForLoopAsArray() {⋯
  }

  static getRandomConsoleLevel() {
    return ["debug", "info", "log", "warning", "error"];
  }

  static generateRandomCode(lines, addComment, includeForLoop) {⋯
  }
}
```

Some methods return an array which will be processed by a helper function to return a random value from that array



```
import { Comments, Helpers } from "../utils/index.js";

export default class JavaScript {
  static getRandomInitializationVars() {⋯
  }

  static getRandomFunctionName() {
    return `${Helpers.getRandomVerb()}${Helpers.getRandomNounCapitalized()}`;
  }

  static getRandomVariableDeclaration() {⋯
  }

  static getRandomFillerLine() {⋯
  }

  static getRandomReturn() {⋯
  }

  static getRandomForLoopAsArray() {⋯
  }

  static getRandomConsoleLevel() {⋯
  }

  static generateRandomCode(lines, addComment, includeForLoop) {⋯
  }
}
```

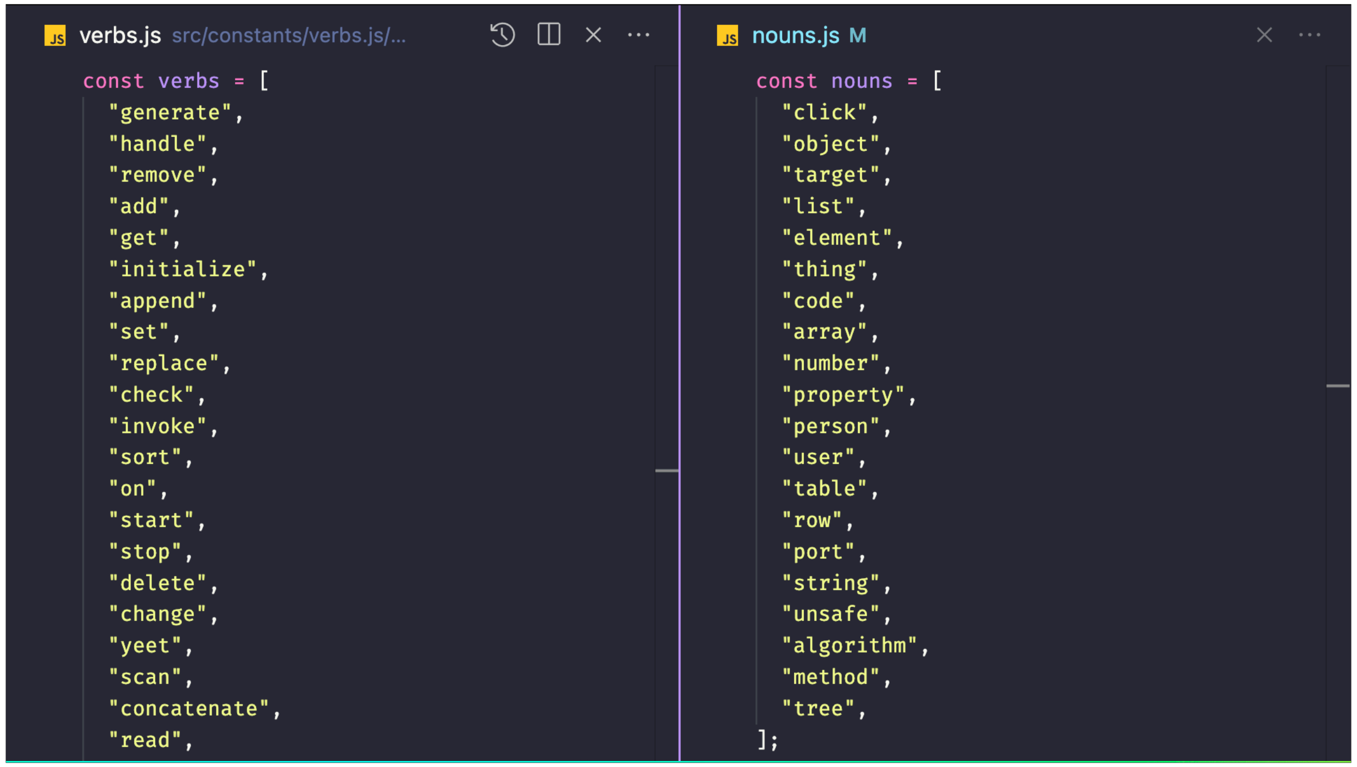
Some methods mash together values returned from helper functions



```
export function getRandomNoun() {
  return getRandomEntry(CONSTANTS.nouns);
}

export function getRandomVerb() {
  return getRandomEntry(CONSTANTS.verbs);
}

export function getRandomSingleCharacter(except = "") {
  let singleCharacter;

  do {
    singleCharacter = getRandomEntry(CONSTANTS.singleCharacters);
  } while (singleCharacter === except);

  return singleCharacter;
}

export function getRandomSuffix() {⋯
}

export function getRandomNounCapitalized() {⋯
}
```

Here are some fun helper functions

Get random noun
Get random verb
Get random single character (often used for nonsense variables)

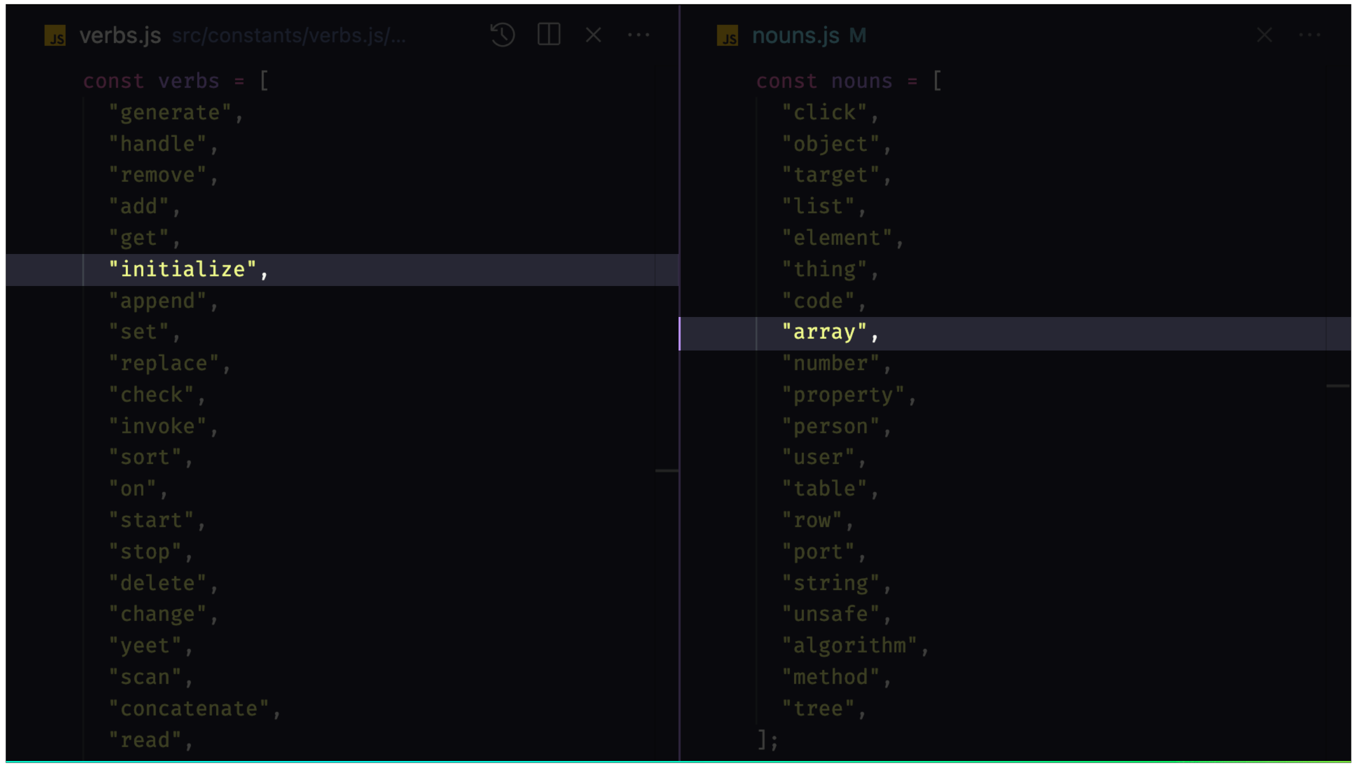


```
const verbs = [
  "generate",
  "handle",
  "remove",
  "add",
  "get",
  "initialize",
  "append",
  "set",
  "replace",
  "check",
  "invoke",
  "sort",
  "on",
  "start",
  "stop",
  "delete",
  "change",
  "yeet",
  "scan",
  "concatenate",
  "read",
```

```
const nouns = [
  "click",
  "object",
  "target",
  "list",
  "element",
  "thing",
  "code",
  "array",
  "number",
  "property",
  "person",
  "user",
  "table",
  "row",
  "port",
  "string",
  "unsafe",
  "algorithm",
  "method",
  "tree",
];
```

Here are the verbs and nouns arrays
When the random code is generated, we'll always start with a function name
Which mashes a verb and noun together

So for example



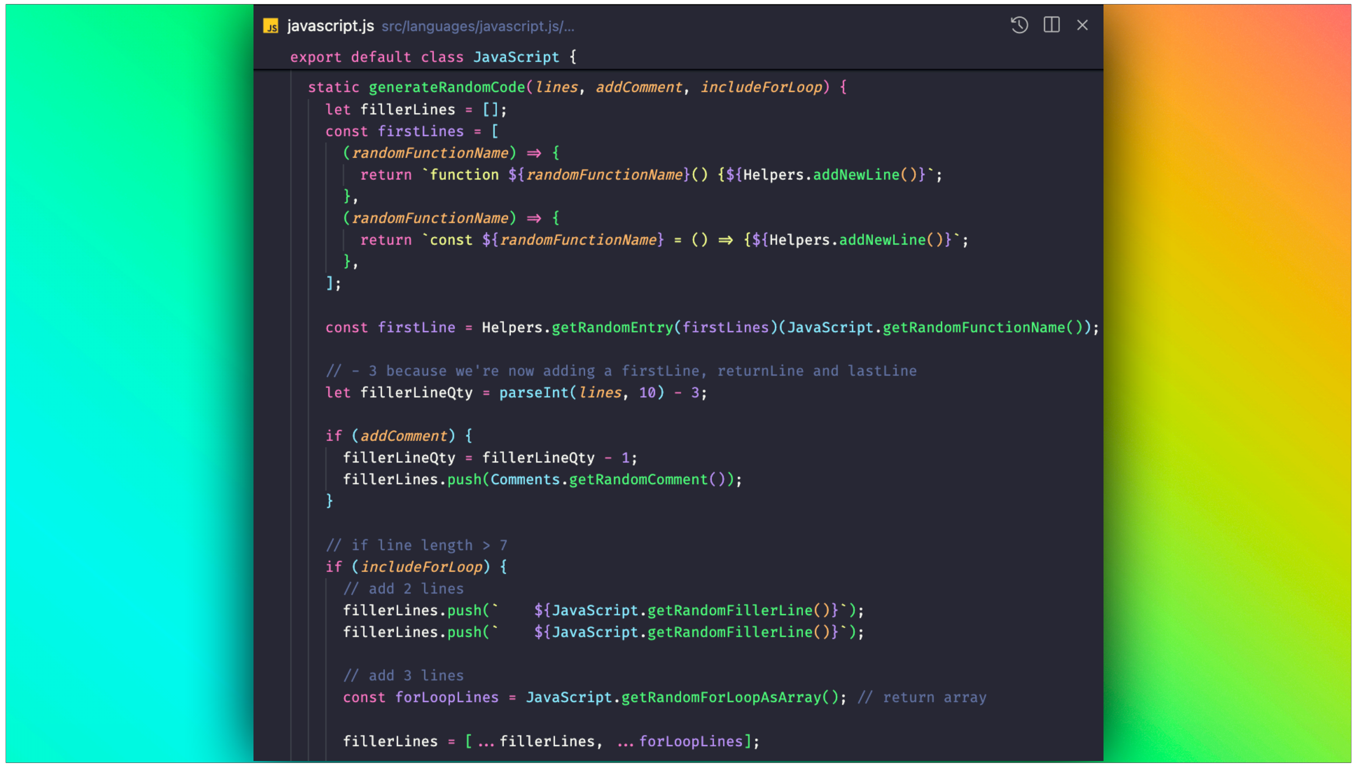
```javascript
const verbs = [
  "generate",
  "handle",
  "remove",
  "add",
  "get",
  "initialize",
  "append",
  "set",
  "replace",
  "check",
  "invoke",
  "sort",
  "on",
  "start",
  "stop",
  "delete",
  "change",
  "yeet",
  "scan",
  "concatenate",
  "read",
```

```javascript
const nouns = [
  "click",
  "object",
  "target",
  "list",
  "element",
  "thing",
  "code",
  "array",
  "number",
  "property",
  "person",
  "user",
  "table",
  "row",
  "port",
  "string",
  "unsafe",
  "algorithm",
  "method",
  "tree",
];
```

Initialise array

Writing code is easy, right? It’s just random words.

And it’s goes on pretty much like that.

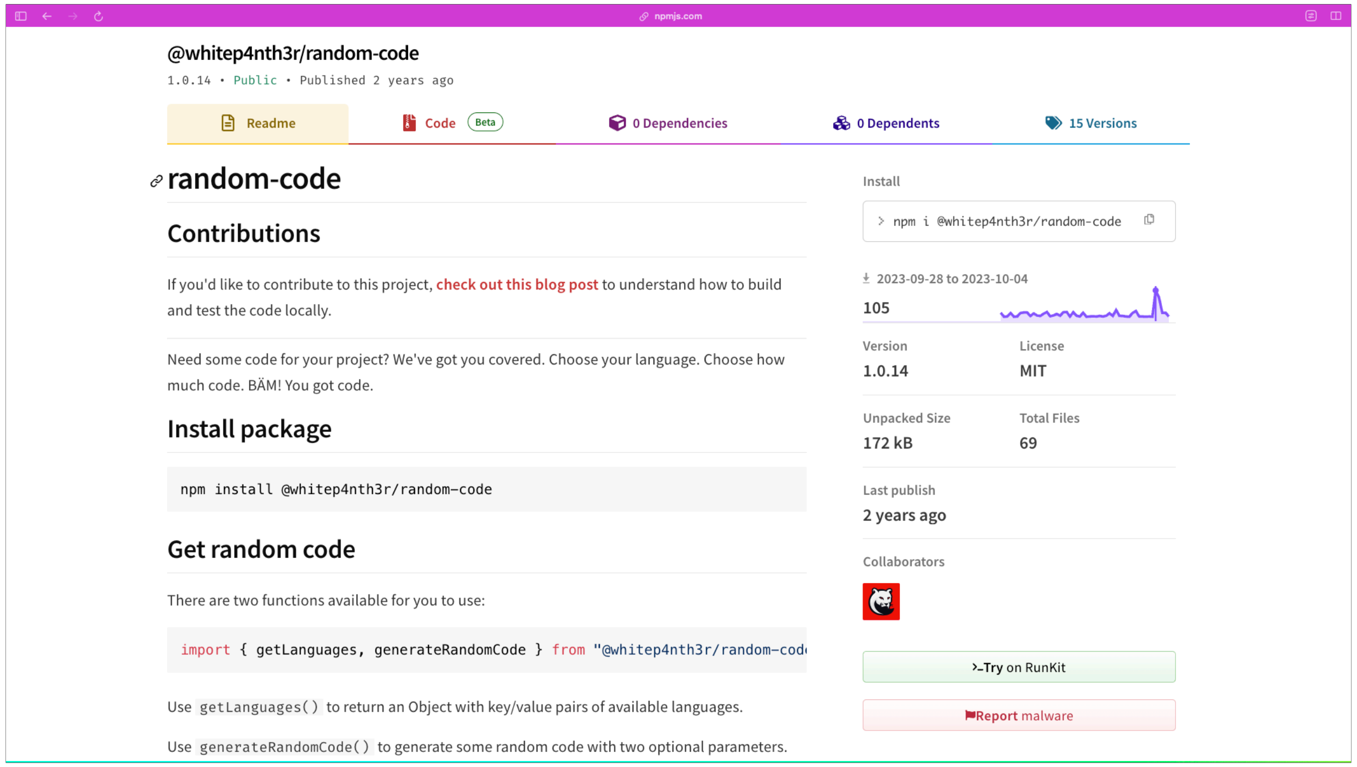


```
export default class JavaScript {
  static generateRandomCode(lines, addComment, includeForLoop) {
    let fillerLines = [];
    const firstLines = [
      (randomFunctionName) => {
        return `function ${randomFunctionName}() {${Helpers.addNewLine()}`;
      },
      (randomFunctionName) => {
        return `const ${randomFunctionName} = () => {${Helpers.addNewLine()}`;
      },
    ];

    const firstLine = Helpers.getRandomEntry(firstLines)(JavaScript.getRandomFunctionName());

    // - 3 because we're now adding a firstLine, returnLine and lastLine
    let fillerLineQty = parseInt(lines, 10) - 3;

    if (addComment) {
      fillerLineQty = fillerLineQty - 1;
      fillerLines.push(Comments.getRandomComment());
    }

    // if line length > 7
    if (includeForLoop) {
      // add 2 lines
      fillerLines.push(`    ${JavaScript.getRandomFillerLine()}`);
      fillerLines.push(`    ${JavaScript.getRandomFillerLine()}`);

      // add 3 lines
      const forLoopLines = JavaScript.getRandomForLoopAsArray(); // return array

      fillerLines = [ ...fillerLines,  ...forLoopLines];
```

I’m not going to explain this… it’s stupid

BUT

If you want to generate some random code, maybe show it on your website, “what I’m coding right now”, much like “what I’m listening to on Spotify right now”… you can!



@whitep4nth3r/random-code

1.0.14 • Public • Published 2 years ago

random-code

Contributions

If you'd like to contribute to this project, check out this blog post to understand how to build and test the code locally.

Need some code for your project? We've got you covered. Choose your language. Choose how much code. BÄM! You got code.

Install package

```
npm install @whitep4nth3r/random-code
```

Get random code

There are two functions available for you to use:

```
import { getLanguages, generateRandomCode } from "@whitep4nth3r/random-cod
```

Use `getLanguages()` to return an Object with key/value pairs of available languages.

Use `generateRandomCode()` to generate some random code with two optional parameters.

Install

```
> npm i @whitep4nth3r/random-code
```

2023-09-28 to 2023-10-04

105

Version	License
1.0.14	MIT

Unpacked Size	Total Files
172 kB	69

Last publish

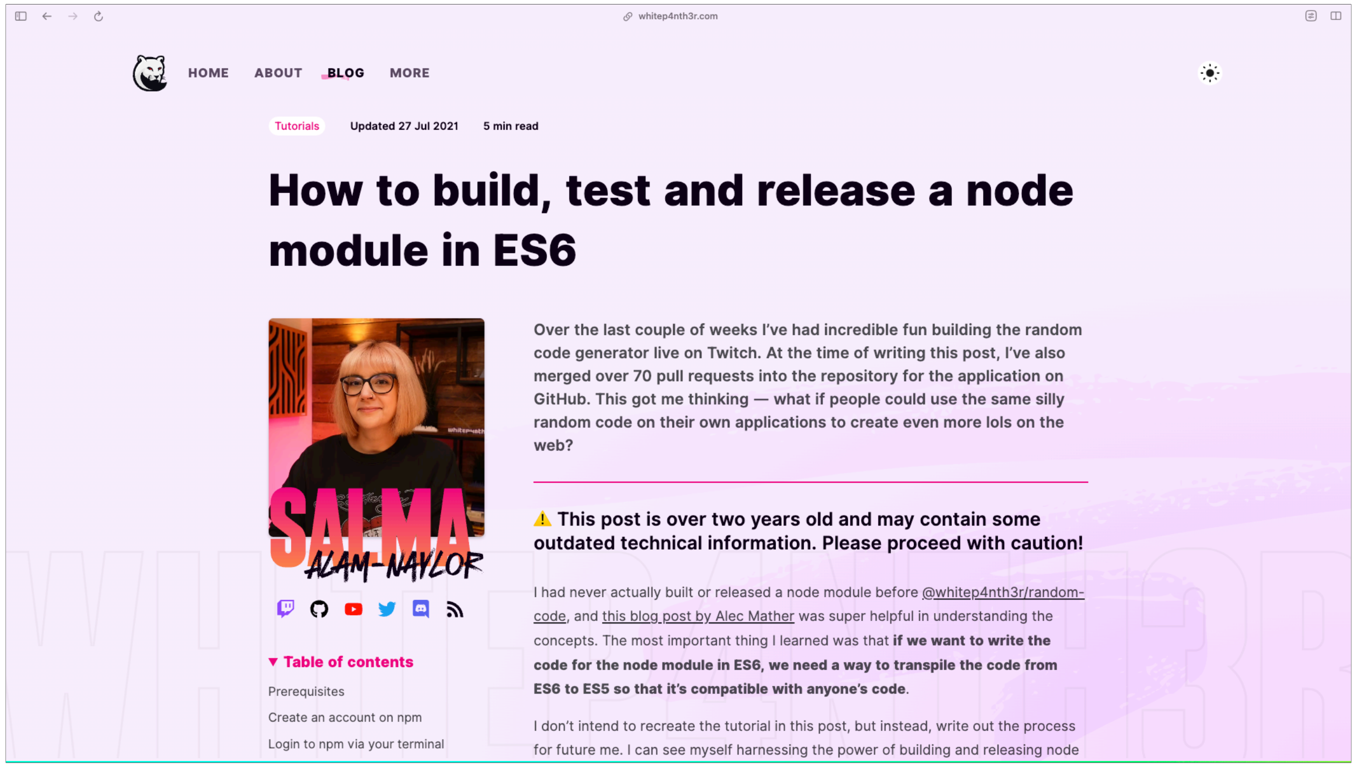
2 years ago

Collaborators

Because I published it to npm

Again, NO IDEA what people are downloading this???
There were 105 downloads in a week in September?

But this was actually the first package I published to npm
Live on stream, of course



Tutorials Updated 27 Jul 2021 5 min read

How to build, test and release a node module in ES6

Over the last couple of weeks I've had incredible fun building the random code generator live on Twitch. At the time of writing this post, I've also merged over 70 pull requests into the repository for the application on GitHub. This got me thinking — what if people could use the same silly random code on their own applications to create even more lols on the web?

⚠️ **This post is over two years old and may contain some outdated technical information. Please proceed with caution!**

I had never actually built or released a node module before @whitep4nth3r/random-code, and this blog post by Alec Mather was super helpful in understanding the concepts. The most important thing I learned was that **if we want to write the code for the node module in ES6, we need a way to transpile the code from ES6 to ES5 so that it's compatible with anyone's code**.

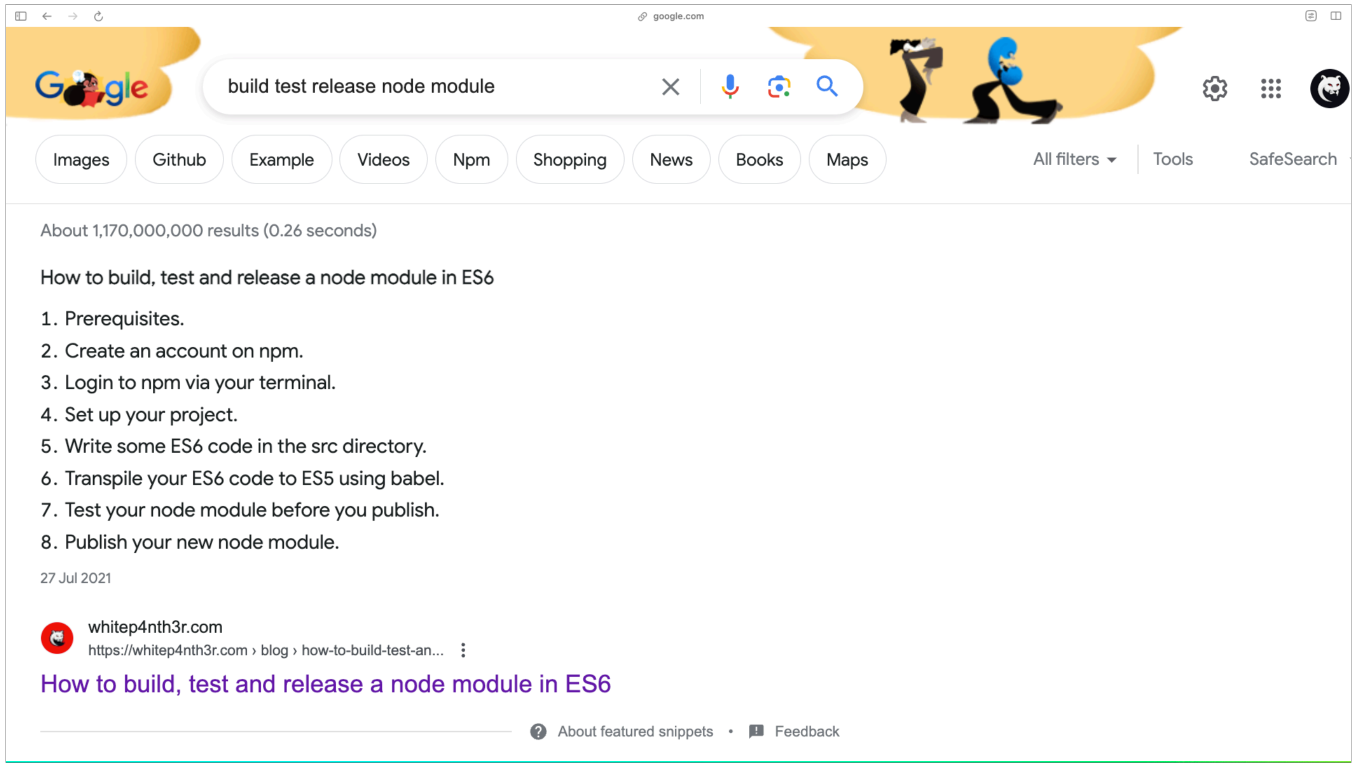
I don't intend to recreate the tutorial in this post, but instead, write out the process for future me. I can see myself harnessing the power of building and releasing node

▼ Table of contents

And I wrote about it

And the blog post was really popular

And if you search for "build test release node module" on google



About 1,170,000,000 results (0.26 seconds)

How to build, test and release a node module in ES6

1. Prerequisites.
2. Create an account on npm.
3. Login to npm via your terminal.
4. Set up your project.
5. Write some ES6 code in the src directory.
6. Transpile your ES6 code to ES5 using babel.
7. Test your node module before you publish.
8. Publish your new node module.

27 Jul 2021

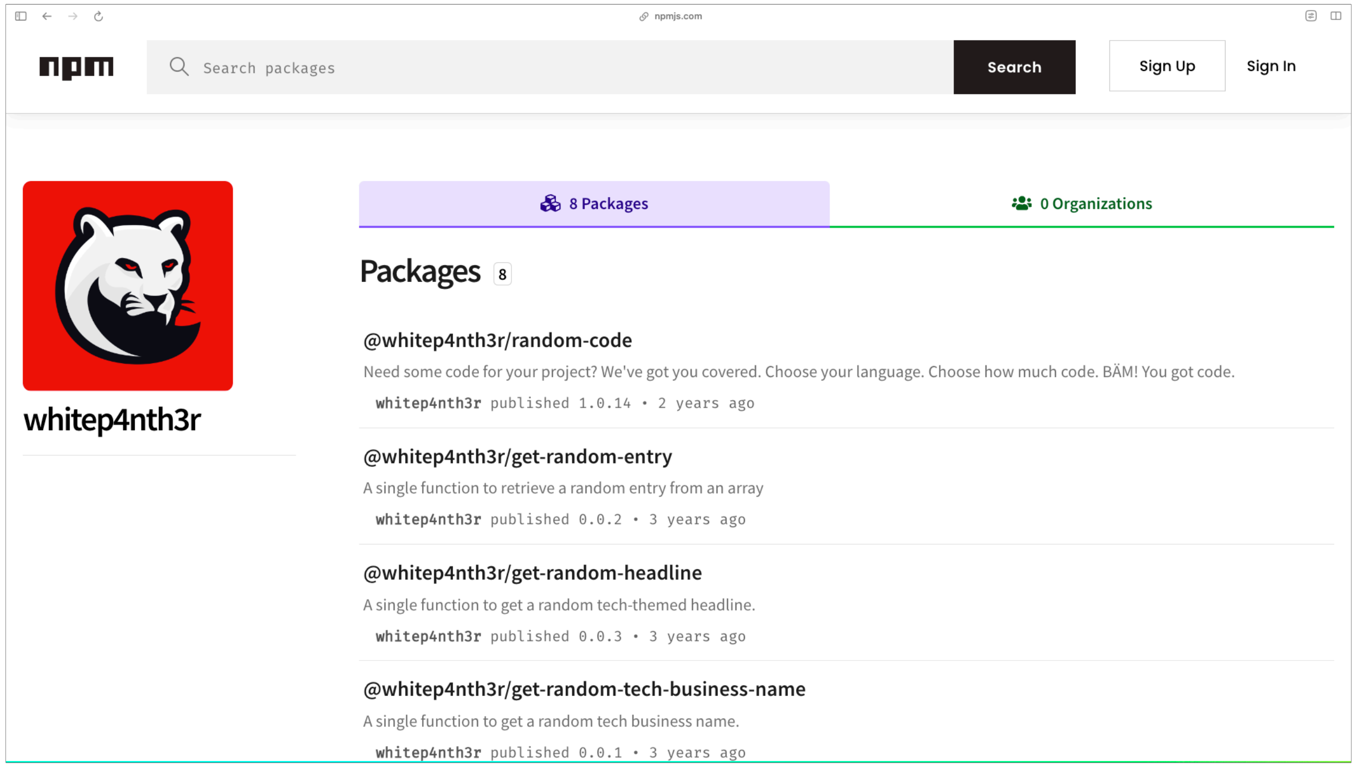
whitep4nth3r.com
https://whitep4nth3r.com › blog › how-to-build-test-an...

How to build, test and release a node module in ES6

It’s the top result with a featured snippet??

One of my biggest achievements

And it ALL CAME OUT OF WRITING STUPID POINTLESS SILLY NONSENSE CODE IN FRONT OF A LIVE AUDIENCE ON TWITCH???



whitep4nth3r

Packages 8

@whitep4nth3r/random-code

Need some code for your project? We've got you covered. Choose your language. Choose how much code. BÄM! You got code.

whitep4nth3r published 1.0.14 • 2 years ago

@whitep4nth3r/get-random-entry

A single function to retrieve a random entry from an array

whitep4nth3r published 0.0.2 • 3 years ago

@whitep4nth3r/get-random-headline

A single function to get a random tech-themed headline.

whitep4nth3r published 0.0.3 • 3 years ago

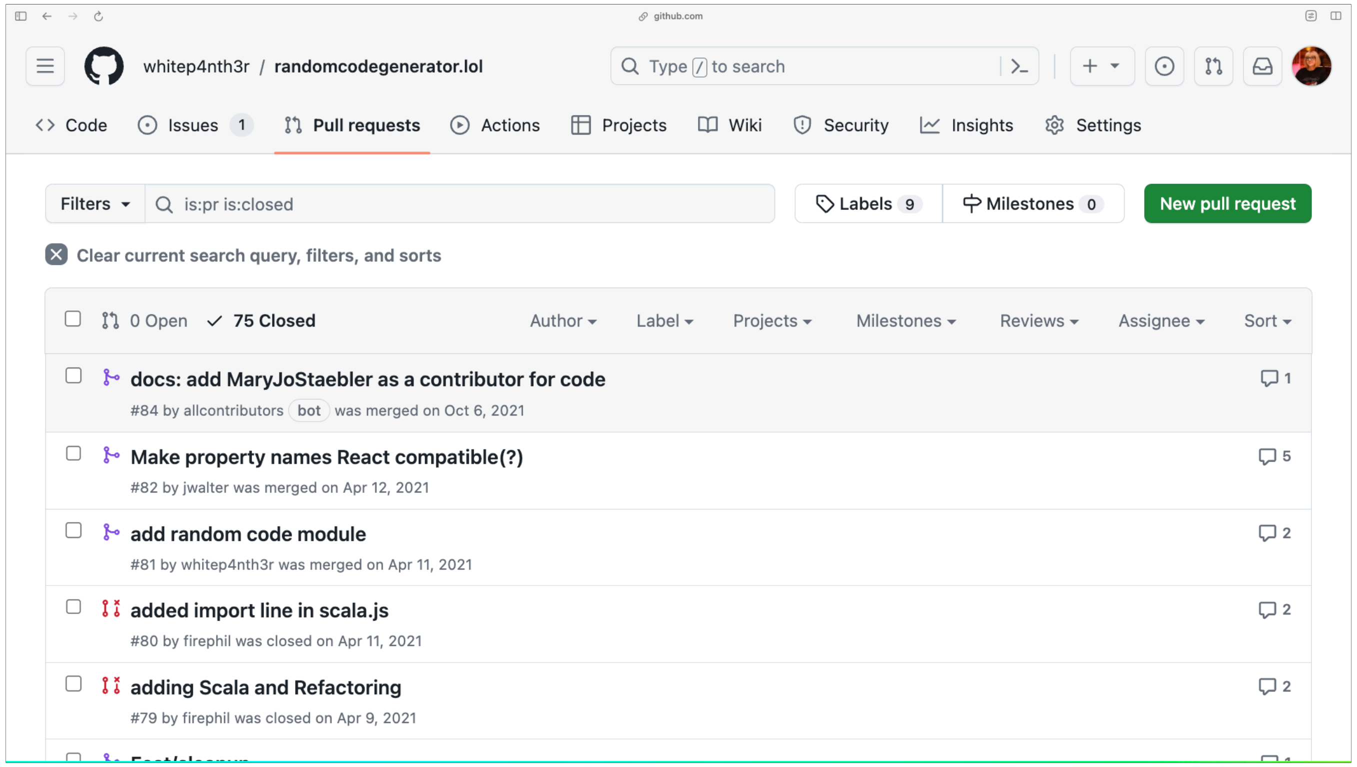
@whitep4nth3r/get-random-tech-business-name

A single function to get a random tech business name.

whitep4nth3r published 0.0.1 • 3 years ago

This led me to publish a few other nonsense get random thing npm modules

And I did start building a random website generator with get-random-headline and get-random-tech-business-name… but it wasn’t funny enough



is:pr is:closed

0 Open ✓ 75 Closed

□ docs: add MaryJoStaebler as a contributor for code — 1
#84 by allcontributors bot was merged on Oct 6, 2021

□ Make property names React compatible(?) — 5
#82 by jwalter was merged on Apr 12, 2021

□ add random code module — 2
#81 by whitep4nth3r was merged on Apr 11, 2021

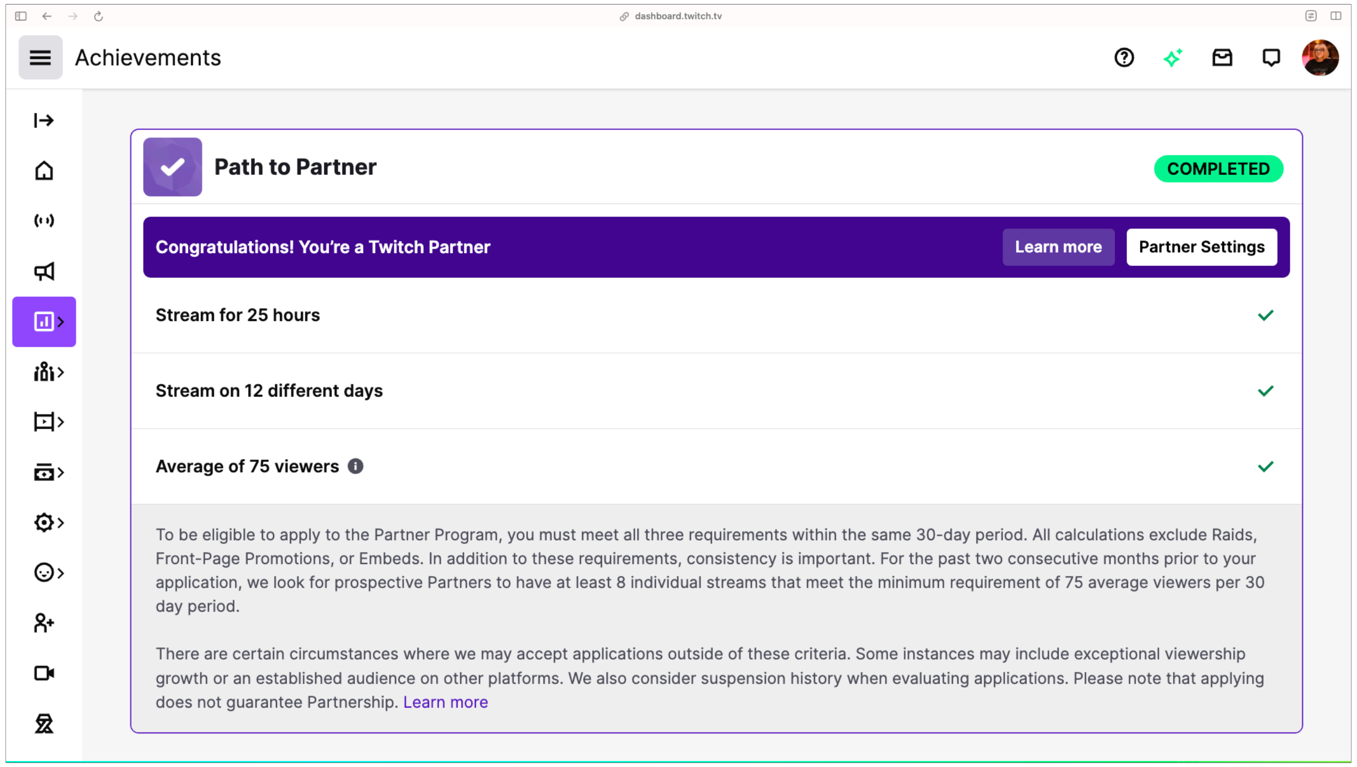
□ added import line in scala.js — 2
#80 by firephil was closed on Apr 11, 2021

□ adding Scala and Refactoring — 2
#79 by firephil was closed on Apr 9, 2021

I also received 75 pull requests to the original random code generator site before I turned it into an npm module. People LOVED adding their own favourite programming language to the catalogue of random code generators. And again, for some people, it was their first open source contribution.

AUGUST 2021

In August 2021, I achieved Twitch partner status.



Path to Partner

COMPLETED

Congratulations! You're a Twitch Partner

Stream for 25 hours ✓

Stream on 12 different days ✓

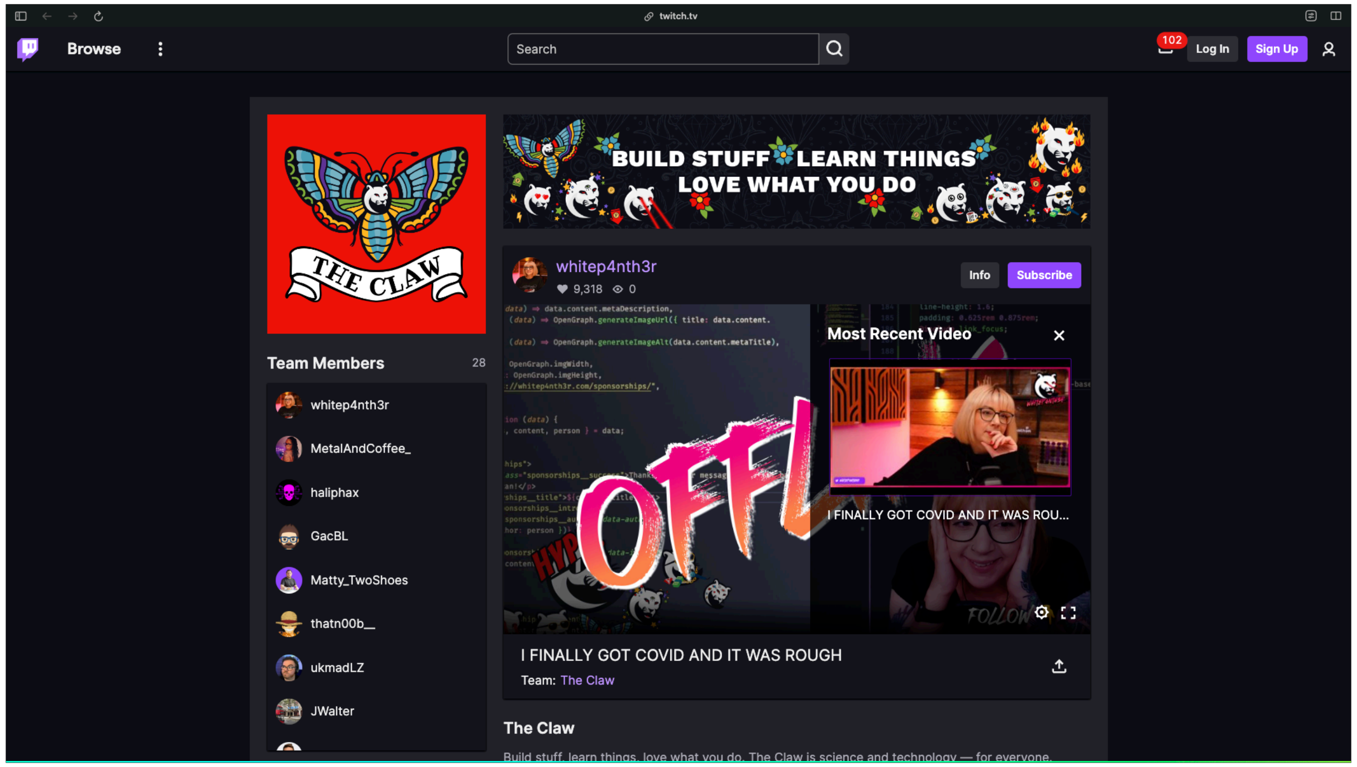
Average of 75 viewers ✓

To be eligible to apply to the Partner Program, you must meet all three requirements within the same 30-day period. All calculations exclude Raids, Front-Page Promotions, or Embeds. In addition to these requirements, consistency is important. For the past two consecutive months prior to your application, we look for prospective Partners to have at least 8 individual streams that meet the minimum requirement of 75 average viewers per 30 day period.

There are certain circumstances where we may accept applications outside of these criteria. Some instances may include exceptional viewership growth or an established audience on other platforms. We also consider suspension history when evaluating applications. Please note that applying does not guarantee Partnership. Learn more

I had streamed enough in 30 days and gained enough average viewers to be given the prestigious checkmark badge next to my username.

To be honest, it’s not that special, there are many many Twitch partners these days
BUT… being a Twitch partner meant I could form an official stream team on Twitch.



Team Members 28

- whitep4nth3r
- MetalAndCoffee_
- haliphax
- GacBL
- Matty_TwoShoes
- thatn00b__
- ukmadLZ
- JWalter

I FINALLY GOT COVID AND IT WAS ROU...

I FINALLY GOT COVID AND IT WAS ROUGH

Team: The Claw

The Claw

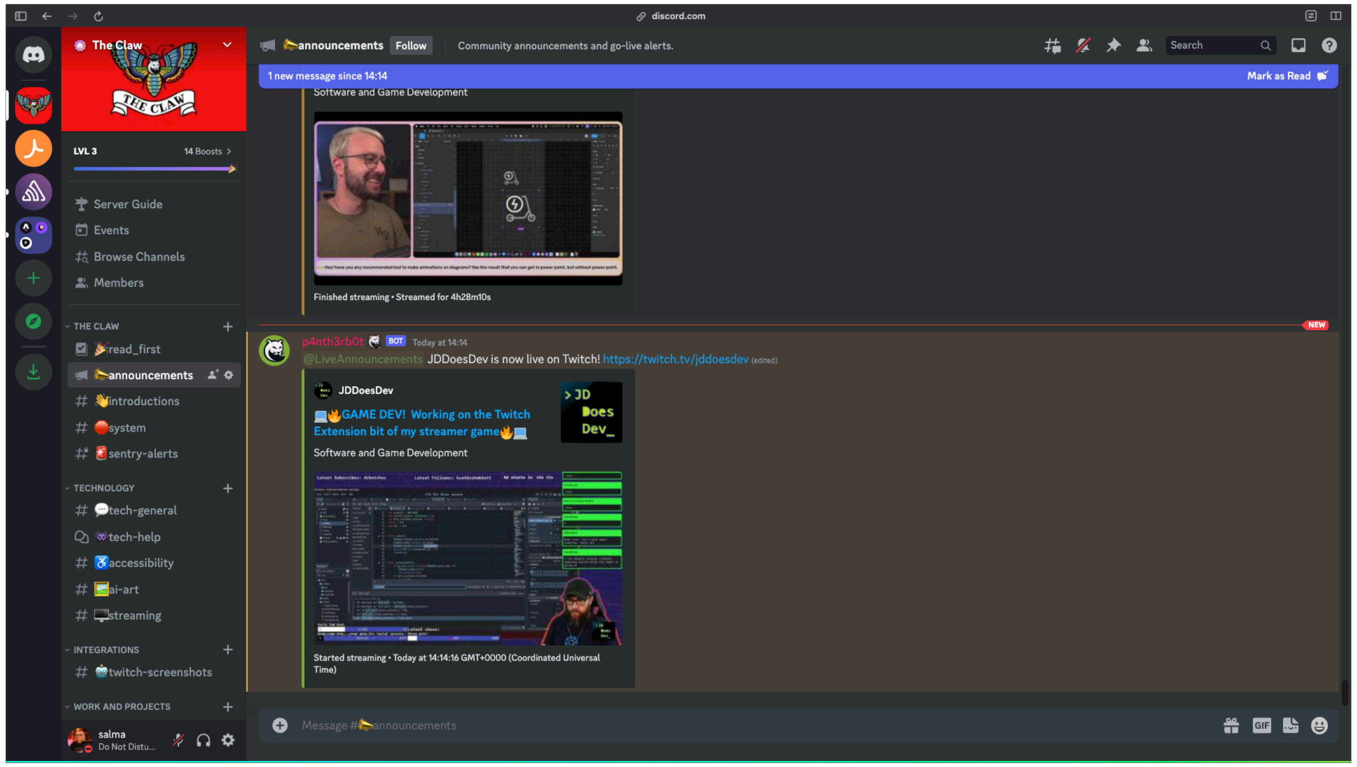
Build stuff, learn things, love what you do. The Claw is science and technology — for everyone.

The Claw

We’re currently at 28 members, located across multiple timezones, who stream software and game development in a variety of formats and programming languages, who are welcoming and inclusive to all

And yes, I need to update the live laugh love banner at the top

And of course, I used this opportunity to add some more functionality to my Twitch bot



Software and Game Development

Finished streaming • Streamed for 4h28m10s

p4nth3rb0t BOT Today at 14:14

@LiveAnnouncements JDDoesDev is now live on Twitch! https://twitch.tv/jddoesdev (edited)

JDDoesDev

💻🔥GAME DEV! Working on the Twitch Extension bit of my streamer game🔥💻

Software and Game Development

Started streaming • Today at 14:14:16 GMT+0000 (Coordinated Universal Time)

Including go-live announcements in Discord

Powered by p4nth3rb0t and the Twitch event sub API, when a stream team member goes live, an announcement is posted to Discord, alerting all members who have opted in to the go live announcement role

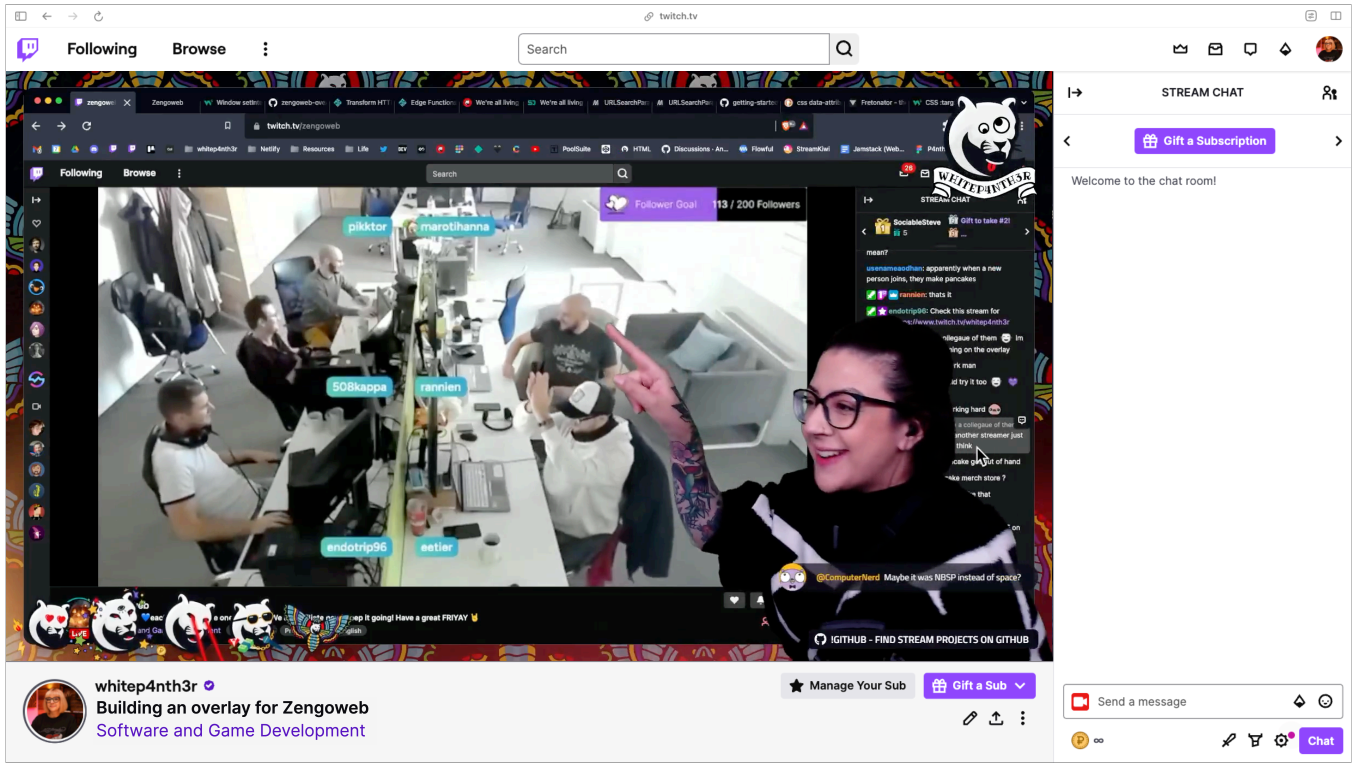
MISCELLANEOUS

There’s so much more I wanted to show you but there’s just not enough time.
Just want to wrap up by showing you some more miscellaneous nonsense.

GIFTS AS CODE

- Last year discovered Zengoweb
- a development agency in Hungary live streaming their work day from the office
- great to have on in the background, co working vibe
- but it was difficult to remember their twitch usernames and who was who in the office — so I decided to build something for them — another browser overlay for OBS to display each person’s Twitch username on the stream

- here’s the moment when it first went live



Welcome to the chat room!

whitep4nth3r

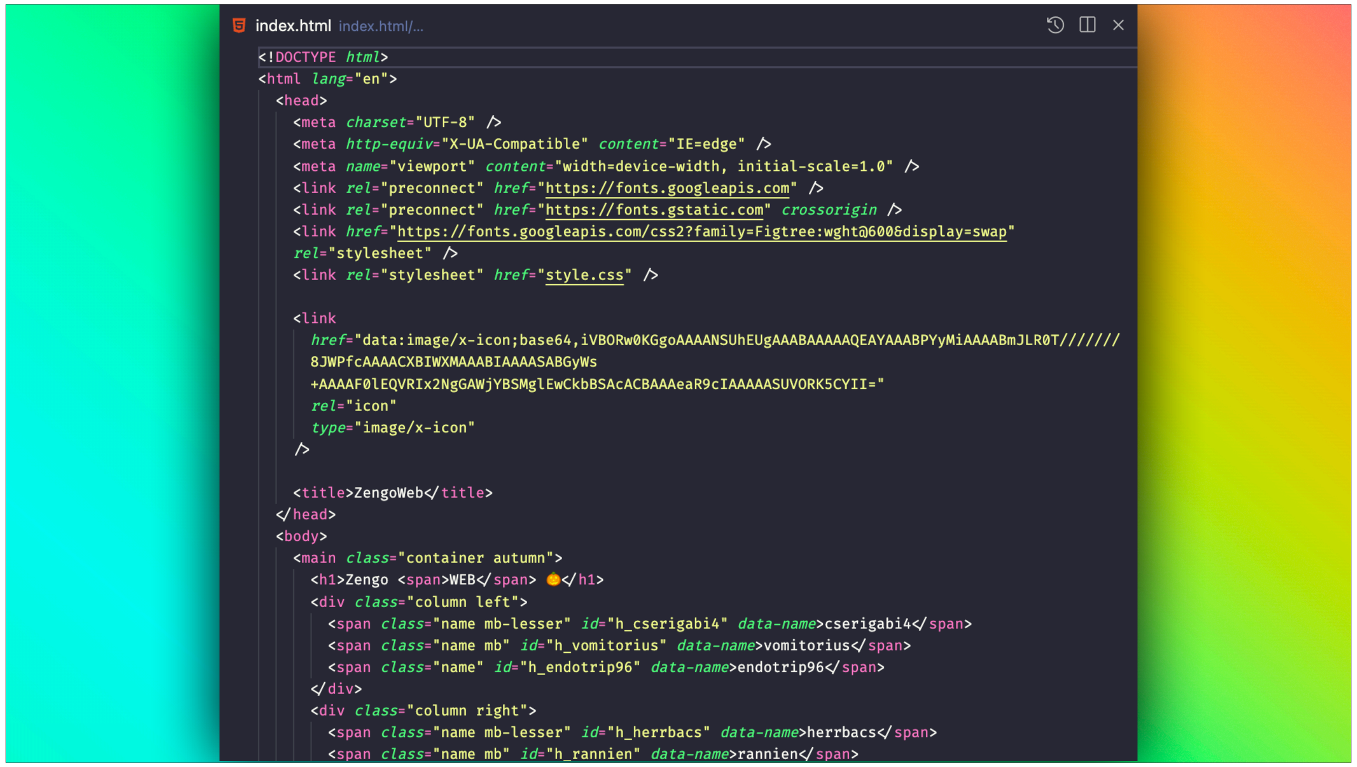
Building an overlay for Zengoweb

Software and Game Development

@ComputerNerd Maybe it was NBSP instead of space?

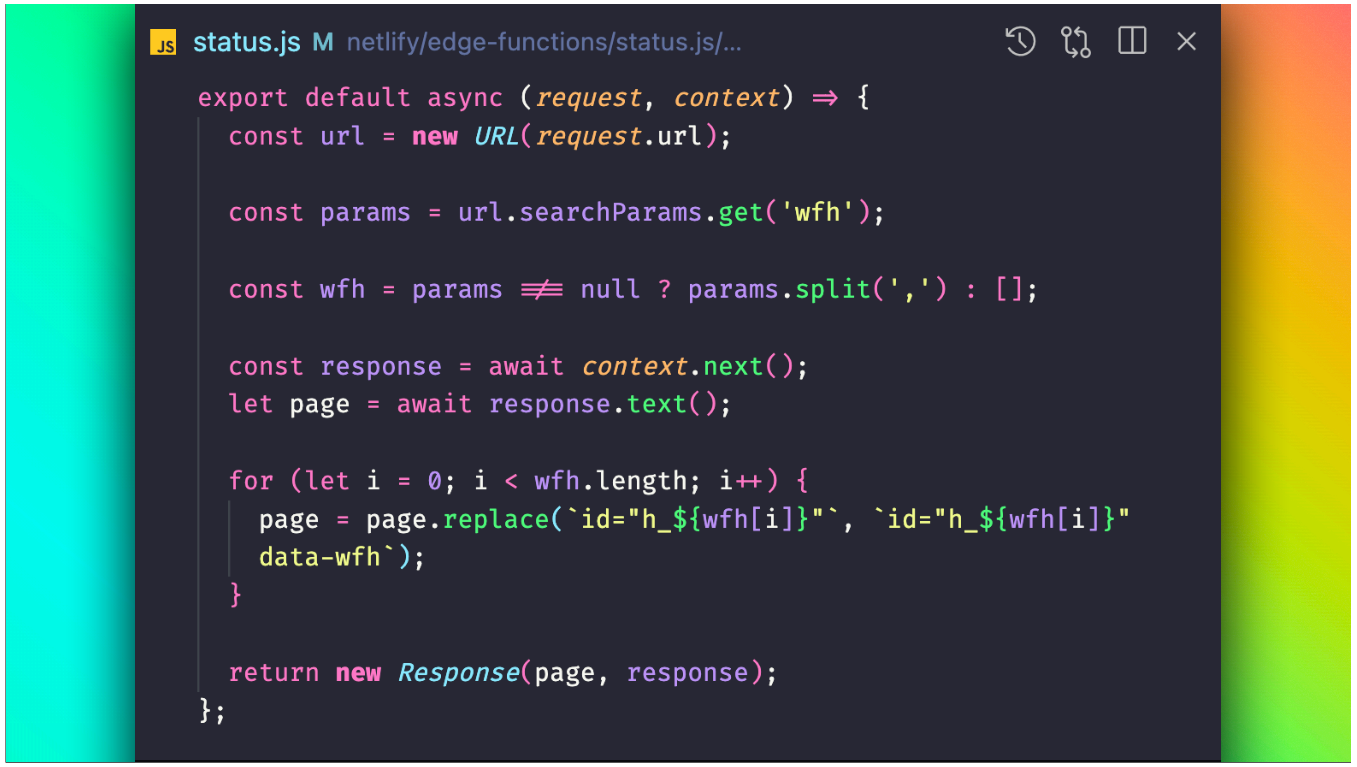
…

So how does it work?



```html
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <link rel="preconnect" href="https://fonts.googleapis.com" />
    <link rel="preconnect" href="https://fonts.gstatic.com" crossorigin />
    <link href="https://fonts.googleapis.com/css2?family=Figtree:wght@600&display=swap"
    rel="stylesheet" />
    <link rel="stylesheet" href="style.css" />

    <link
      href="data:image/x-icon;base64,iVBORw0KGgoAAAANSUhEUgAAABAAAAAQEAYAAABPYyMiAAAABmJLR0T///////
      8JWPfcAAAACXBIWXMAAABIAAAASABGyWs
      +AAAAF0lEQVRIx2NgGAWjYBSMglEwCkbBSAcACBAAAeaR9cIAAAAASUVORK5CYII="
      rel="icon"
      type="image/x-icon"
    />

    <title>ZengoWeb</title>
  </head>
  <body>
    <main class="container autumn">
      <h1>Zengo <span>WEB</span> 🎃</h1>
      <div class="column left">
        <span class="name mb-lesser" id="h_cserigabi4" data-name>cserigabi4</span>
        <span class="name mb" id="h_vomitorius" data-name>vomitorius</span>
        <span class="name" id="h_endotrip96" data-name>endotrip96</span>
      </div>
      <div class="column right">
        <span class="name mb-lesser" id="h_herrbacs" data-name>herrbacs</span>
        <span class="name mb" id="h_rannien" data-name>rannien</span>
```

Plain html page
It’s hosted on Netlify
and in order to give the zengoweb team some control of the overlay without needing to edit the code

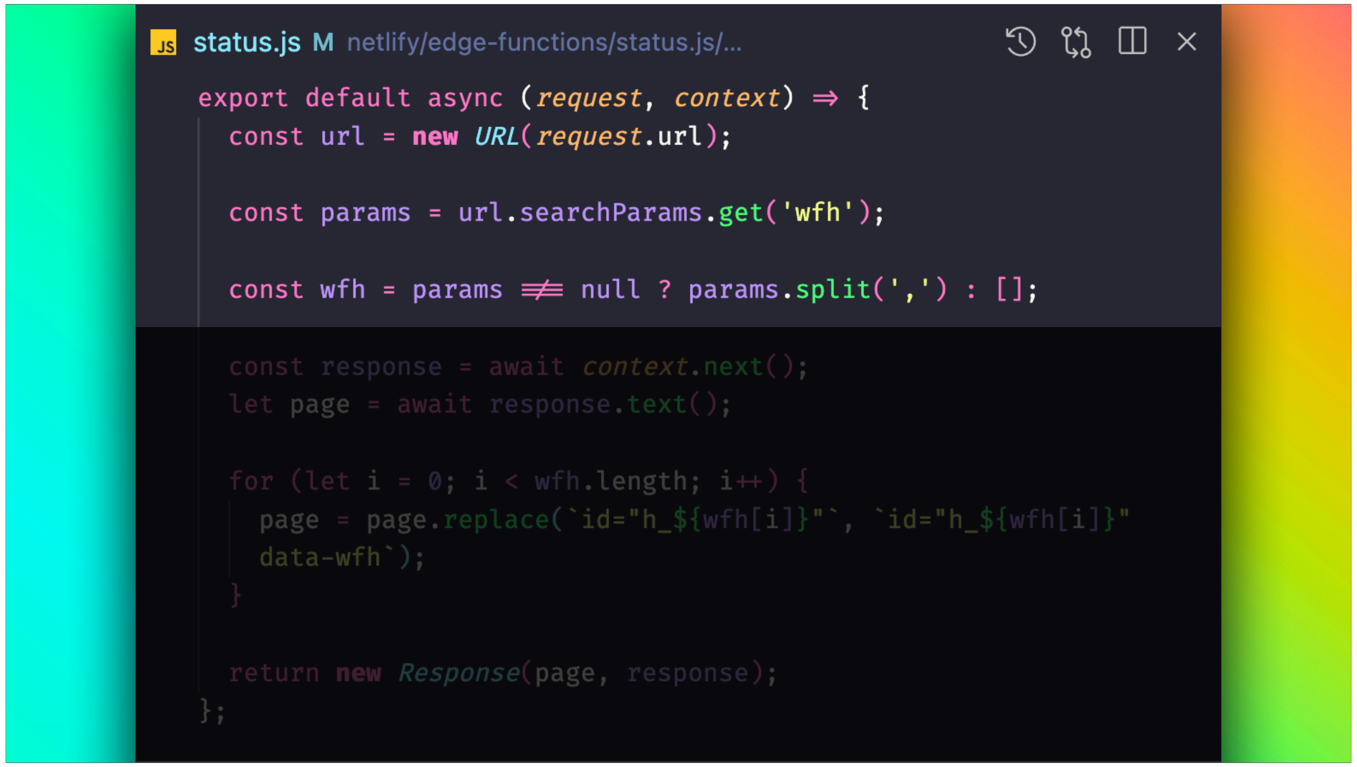


```
export default async (request, context) => {
  const url = new URL(request.url);

  const params = url.searchParams.get('wfh');

  const wfh = params !== null ? params.split(',') : [];

  const response = await context.next();
  let page = await response.text();

  for (let i = 0; i < wfh.length; i++) {
    page = page.replace(`id="h_${wfh[i]}"`, `id="h_${wfh[i]}"
    data-wfh`);
  }

  return new Response(page, response);
};
```

It also uses a Netlify edge function (serverless function that runs at the closest server location to the request)

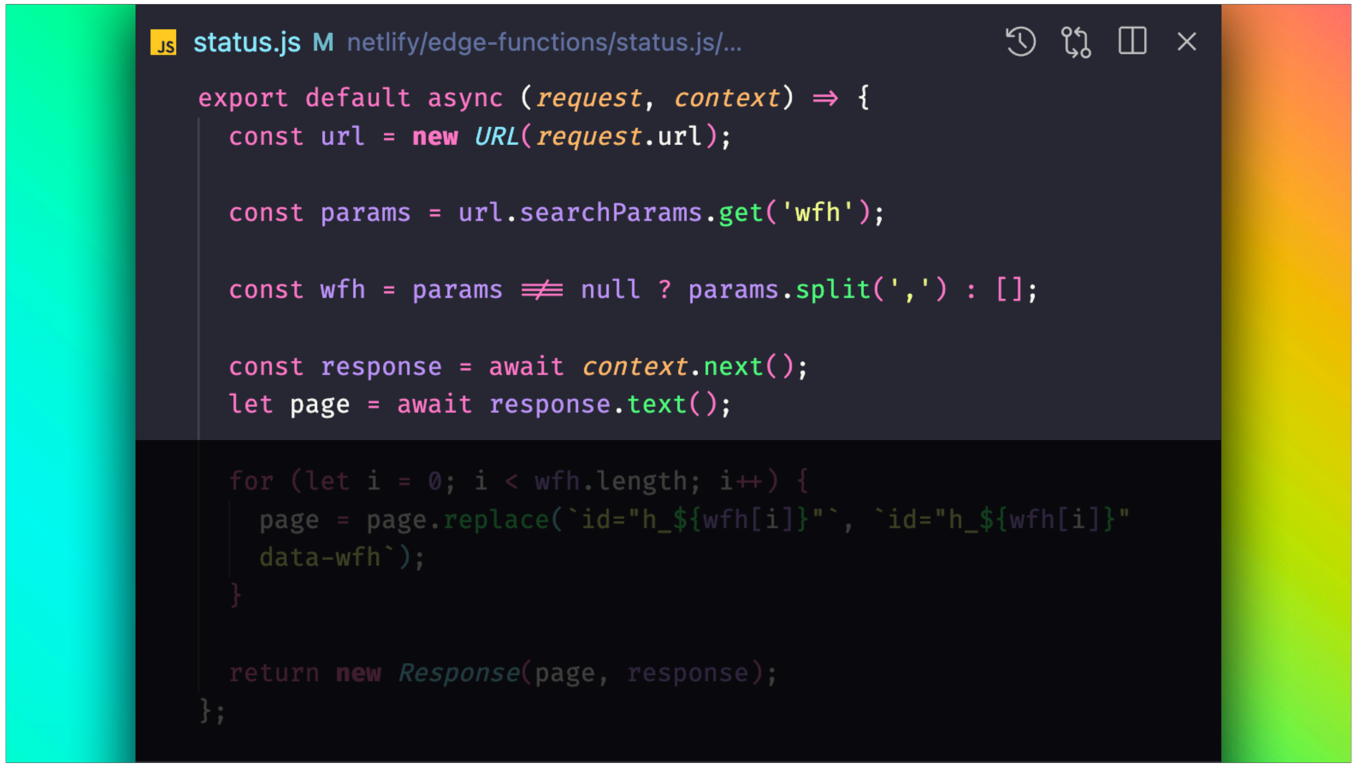
Which enables zengoweb to use a url query parameter to show team members as working from home whenever they need to

This works by intercepting the HTTP request for index.html



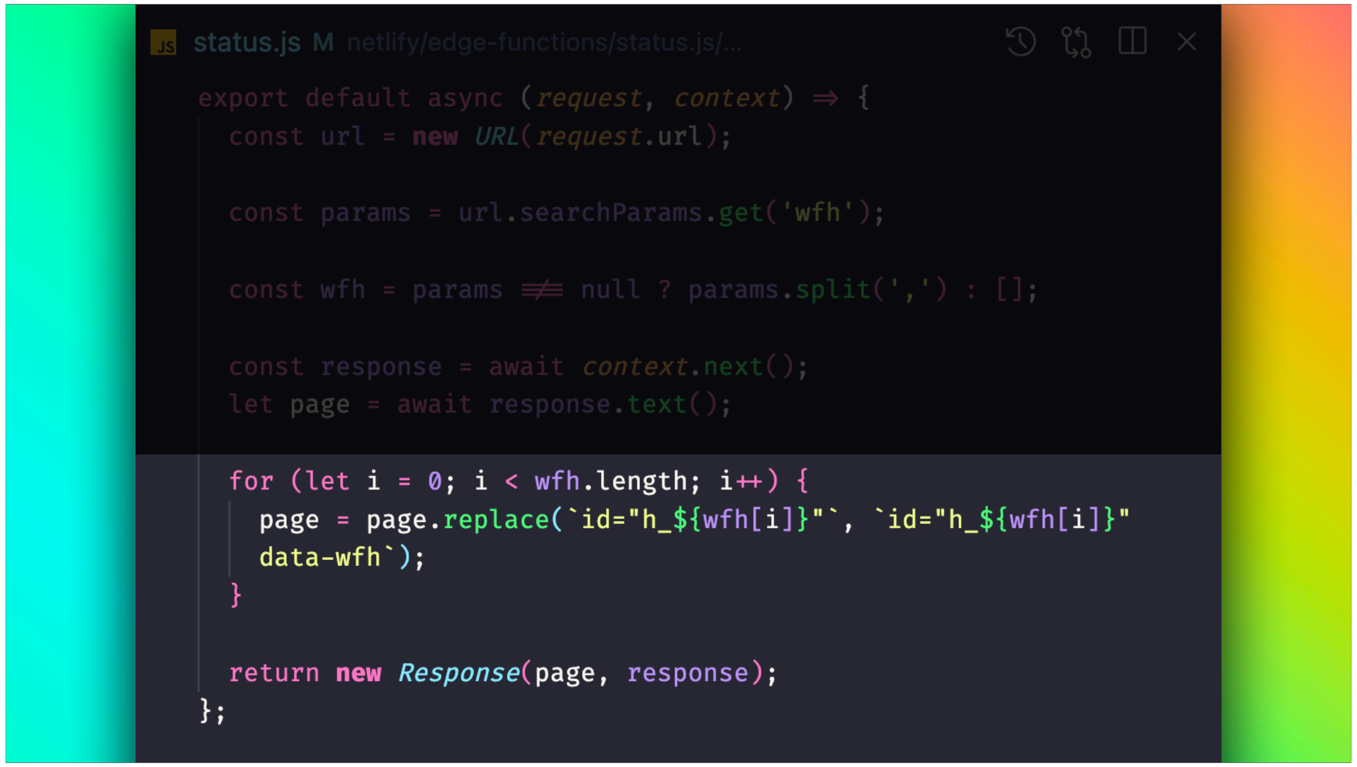
```js
export default async (request, context) => {
  const url = new URL(request.url);

  const params = url.searchParams.get('wfh');

  const wfh = params !== null ? params.split(',') : [];

  const response = await context.next();
  let page = await response.text();

  for (let i = 0; i < wfh.length; i++) {
    page = page.replace(`id="h_${wfh[i]}"`, `id="h_${wfh[i]}"
    data-wfh`);
  }

  return new Response(page, response);
};
```

Parsing the wfh URL parameter



```javascript
export default async (request, context) => {
  const url = new URL(request.url);

  const params = url.searchParams.get('wfh');

  const wfh = params !== null ? params.split(',') : [];

  const response = await context.next();
  let page = await response.text();

  for (let i = 0; i < wfh.length; i++) {
    page = page.replace(`id="h_${wfh[i]}"`, `id="h_${wfh[i]}"
    data-wfh`);
  }

  return new Response(page, response);
};
```

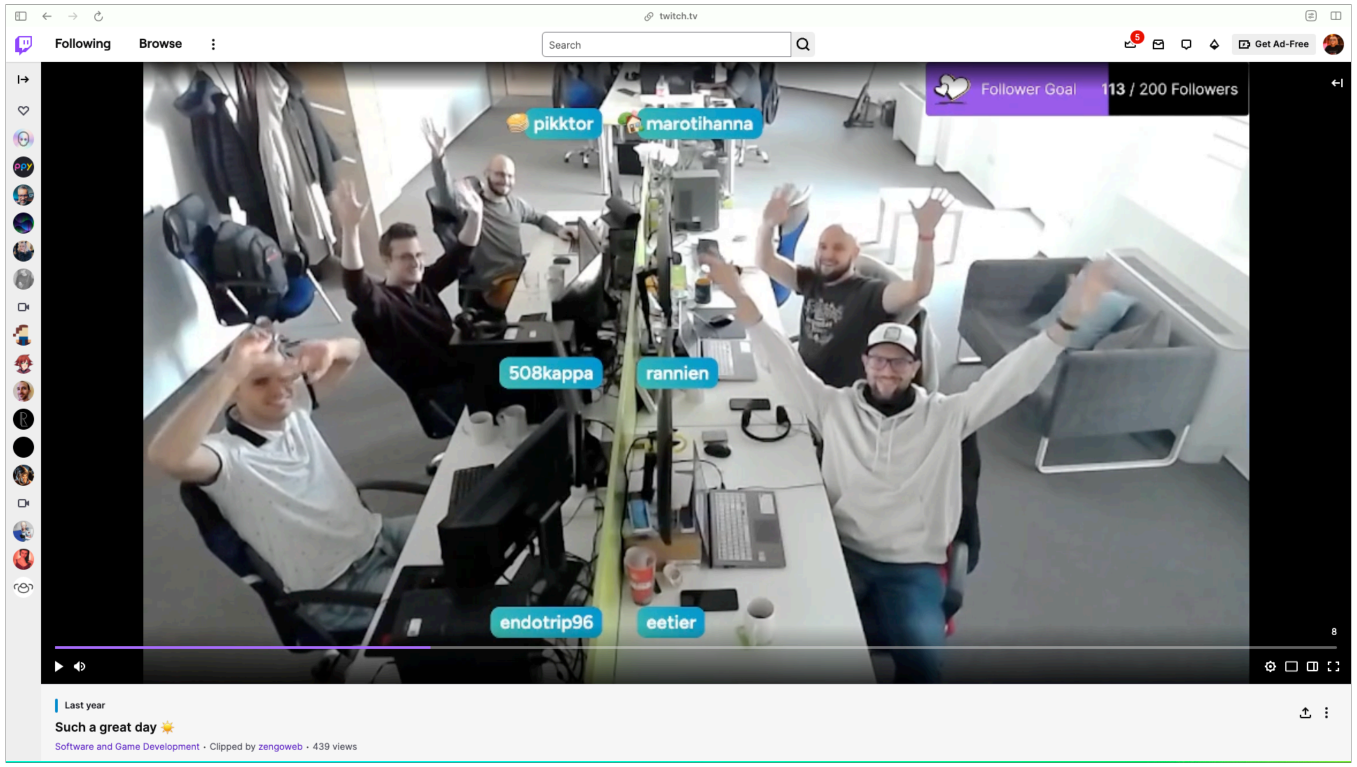
Grabbing the HTTP response of the initial request



```
export default async (request, context) => {
  const url = new URL(request.url);

  const params = url.searchParams.get('wfh');

  const wfh = params !== null ? params.split(',') : [];

  const response = await context.next();
  let page = await response.text();

  for (let i = 0; i < wfh.length; i++) {
    page = page.replace(`id="h_${wfh[i]}"`, `id="h_${wfh[i]}"
    data-wfh`);
  }

  return new Response(page, response);
};
```

Modifying the response accordingly, and returning it

The zengoweb office also has a tradition of eating pancakes



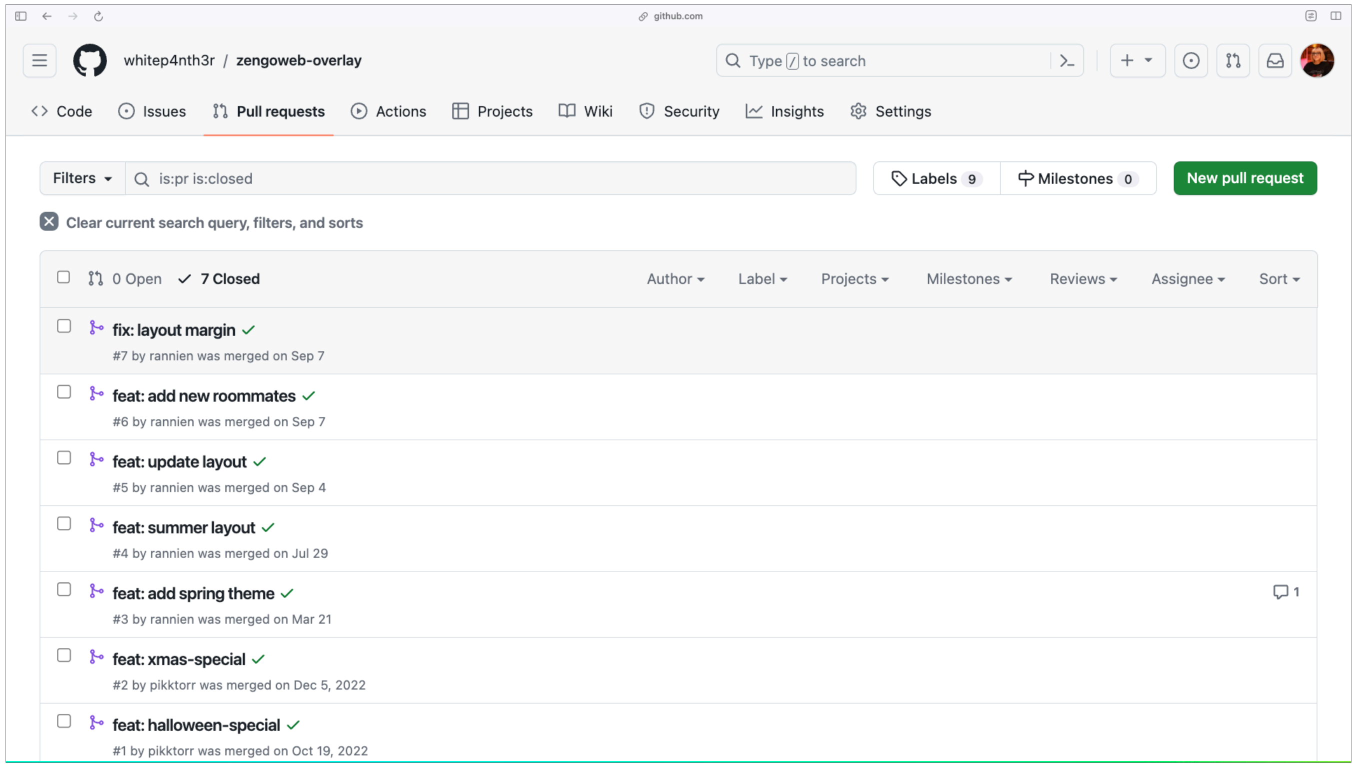
Last year

Such a great day ☀️

Software and Game Development · Clipped by zengoweb · 439 views

There’s also some client side JS that runs on page load that assigns a random team member a pancake emoji, so they know it’s their turn to make/purchase/source pancakes.

On the day of release, it was pikktor’s turn.



is:pr is:closed

☒ Clear current search query, filters, and sorts

□ 0 Open ✓ 7 Closed

□ fix: layout margin ✓
#7 by rannien was merged on Sep 7

□ feat: add new roommates ✓
#6 by rannien was merged on Sep 7

□ feat: update layout ✓
#5 by rannien was merged on Sep 4

□ feat: summer layout ✓
#4 by rannien was merged on Jul 29

□ feat: add spring theme ✓ 1
#3 by rannien was merged on Mar 21

□ feat: xmas-special ✓
#2 by pikktorr was merged on Dec 5, 2022

□ feat: halloween-special ✓
#1 by pikktorr was merged on Oct 19, 2022

And the team have also submitted quite a few PRs for new themes, improvements, so that’s fun.

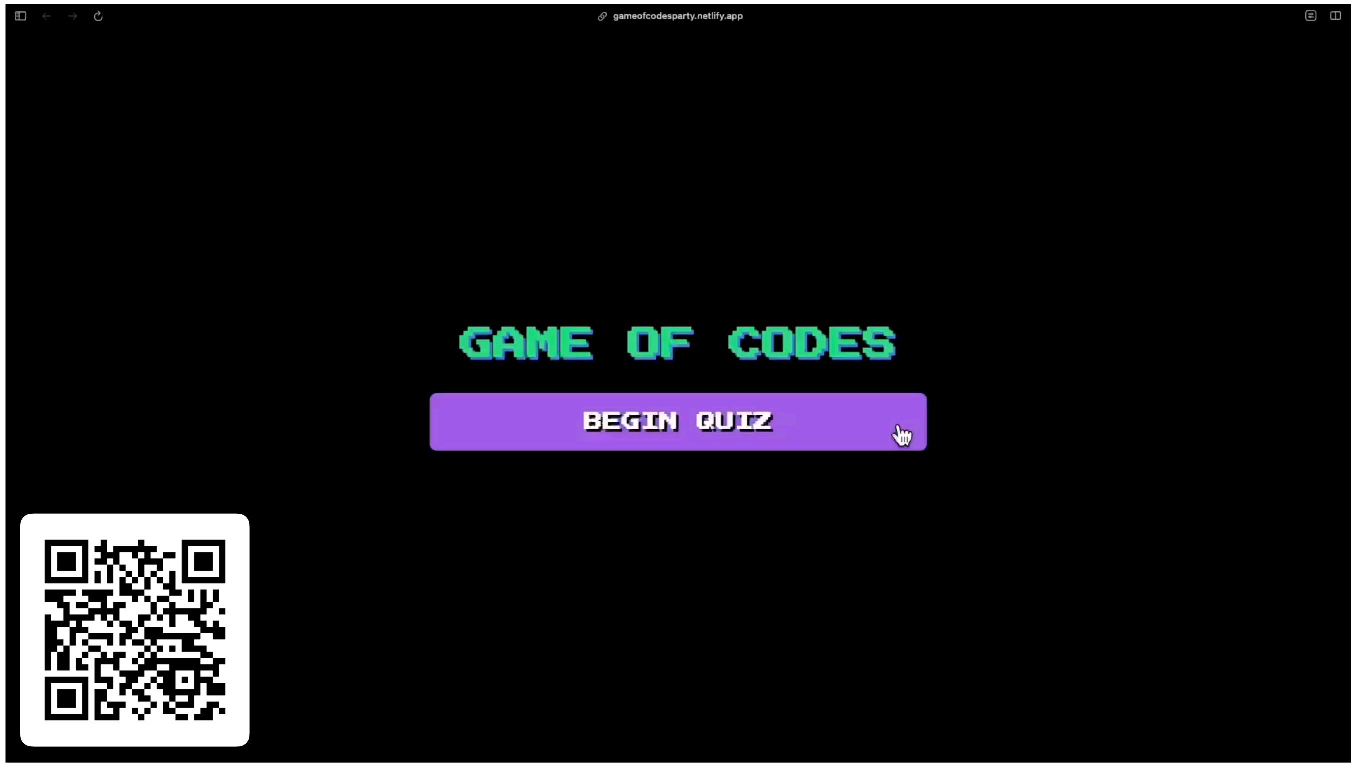
EDUCATION AS CODE

I’ve also experimented with creating some fun educational resources

One example of this is game of codes
An HTTP status code quiz

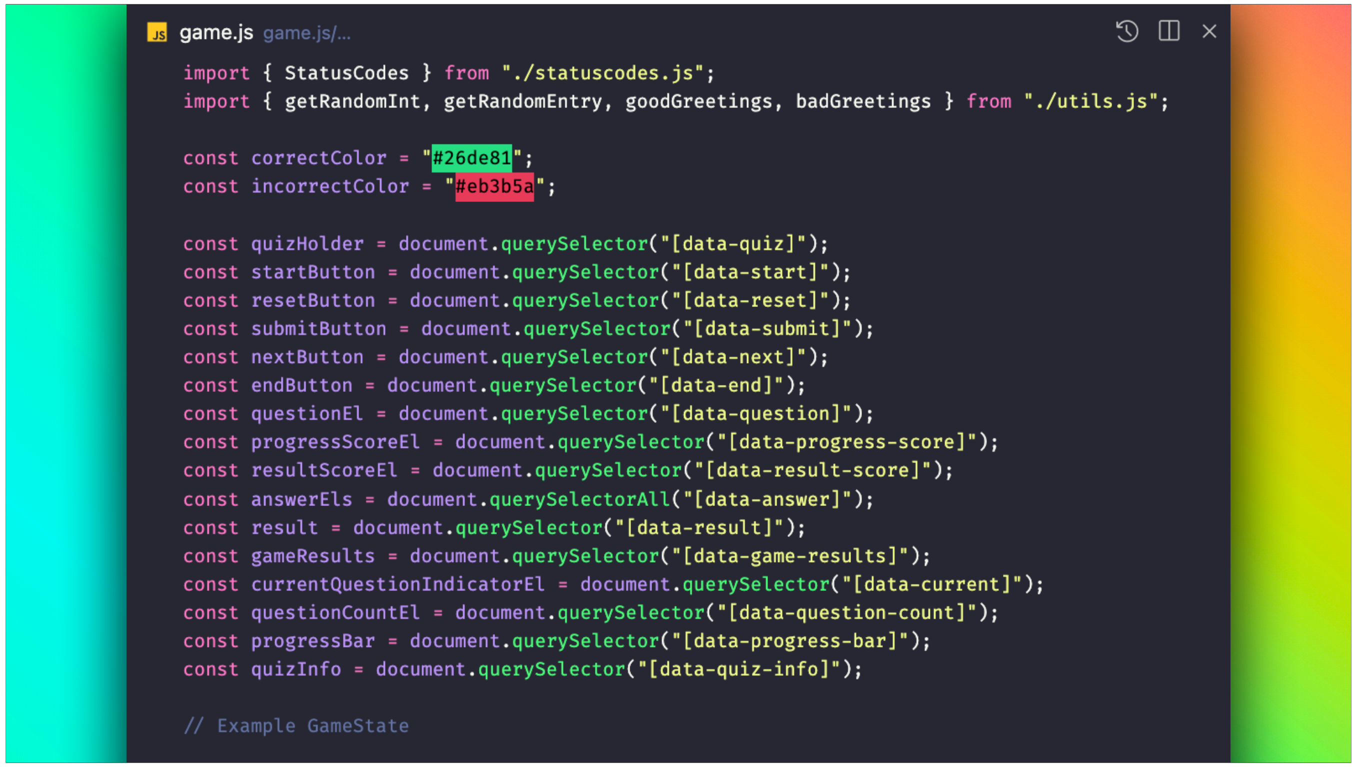
[click] here’s a QR code if you have internet and you want to play along or save it for later

Here’s a very sped up video of me completing the quiz…



This was one take
I didn't try it over and over again to get better results
But my score isn't bad!

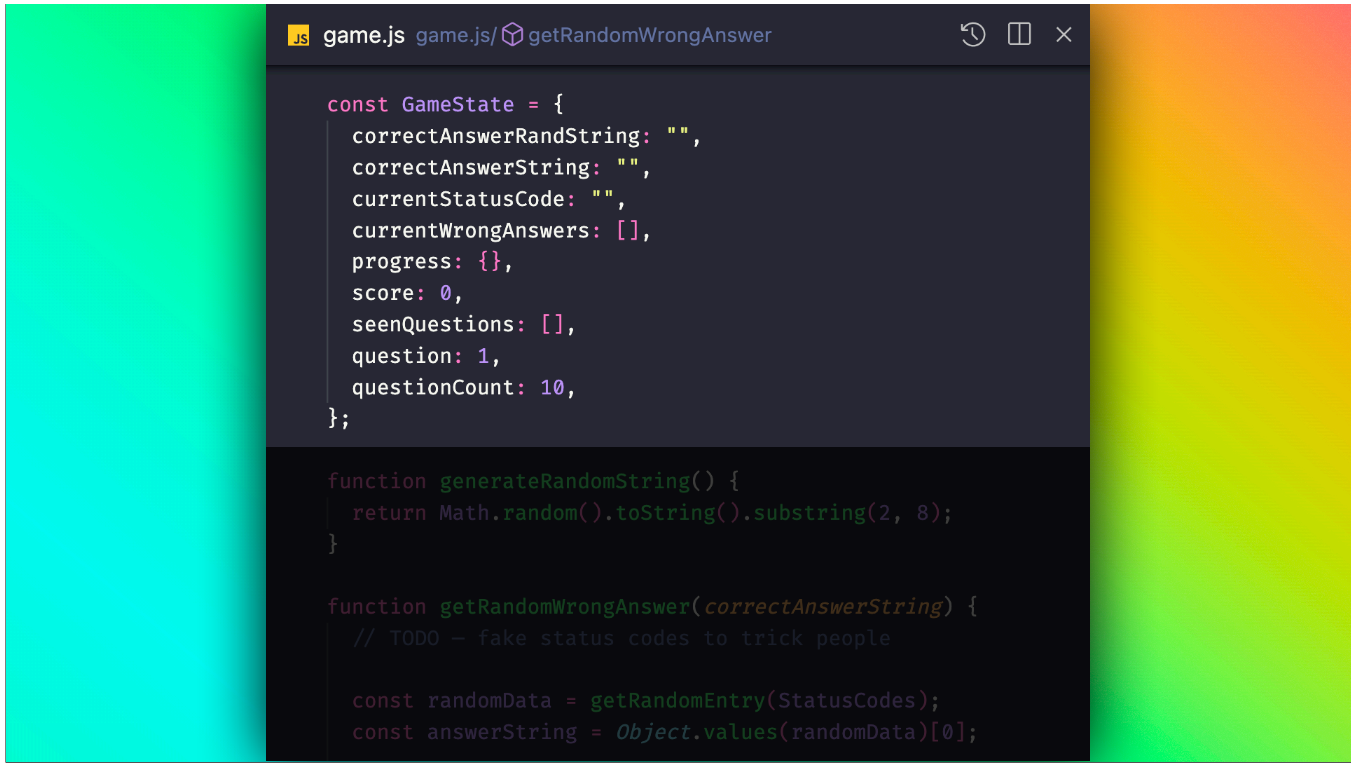
When the quiz is complete, it gives you a summary of what you got right, what you got wrong, and correct answers, so you can leaaarrrnn
And you can start over to try again



```
import { StatusCodes } from "./statuscodes.js";
import { getRandomInt, getRandomEntry, goodGreetings, badGreetings } from "./utils.js";

const correctColor = "#26de81";
const incorrectColor = "#eb3b5a";

const quizHolder = document.querySelector("[data-quiz]");
const startButton = document.querySelector("[data-start]");
const resetButton = document.querySelector("[data-reset]");
const submitButton = document.querySelector("[data-submit]");
const nextButton = document.querySelector("[data-next]");
const endButton = document.querySelector("[data-end]");
const questionEl = document.querySelector("[data-question]");
const progressScoreEl = document.querySelector("[data-progress-score]");
const resultScoreEl = document.querySelector("[data-result-score]");
const answerEls = document.querySelectorAll("[data-answer]");
const result = document.querySelector("[data-result]");
const gameResults = document.querySelector("[data-game-results]");
const currentQuestionIndicatorEl = document.querySelector("[data-current]");
const questionCountEl = document.querySelector("[data-question-count]");
const progressBar = document.querySelector("[data-progress-bar]");
const quizInfo = document.querySelector("[data-quiz-info]");

// Example GameState
```

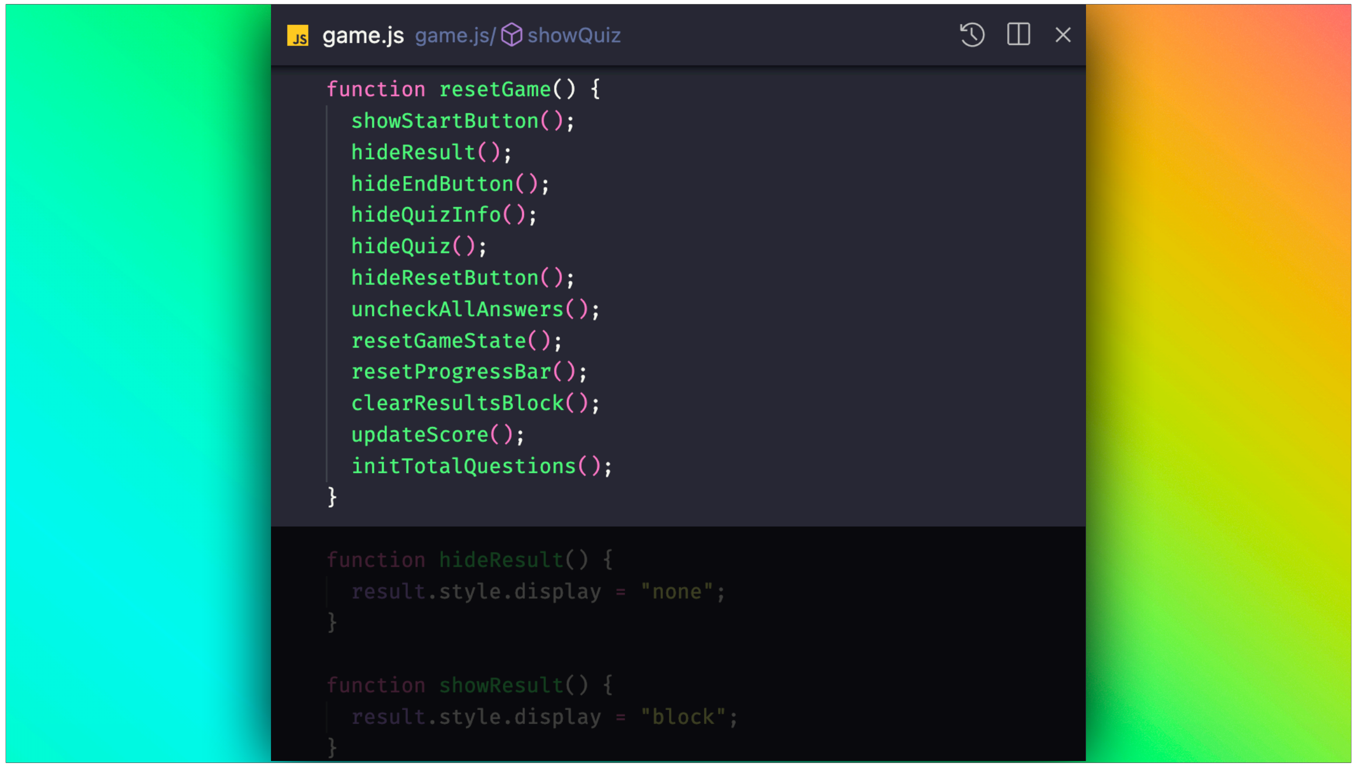
Game of codes is written in plain JavaScript
Shout out to document.querySelector

With game of codes, I also wanted to show what’s possible without a JavaScript framework or any dependencies



```javascript
const GameState = {
  correctAnswerRandString: "",
  correctAnswerString: "",
  currentStatusCode: "",
  currentWrongAnswers: [],
  progress: {},
  score: 0,
  seenQuestions: [],
  question: 1,
  questionCount: 10,
};

function generateRandomString() {
  return Math.random().toString().substring(2, 8);
}

function getRandomWrongAnswer(correctAnswerString) {
  // TODO — fake status codes to trick people

  const randomData = getRandomEntry(StatusCodes);
  const answerString = Object.values(randomData)[0];
```

Instead of using a state management library we’re using a good old JavaScript object to keep track of the game state

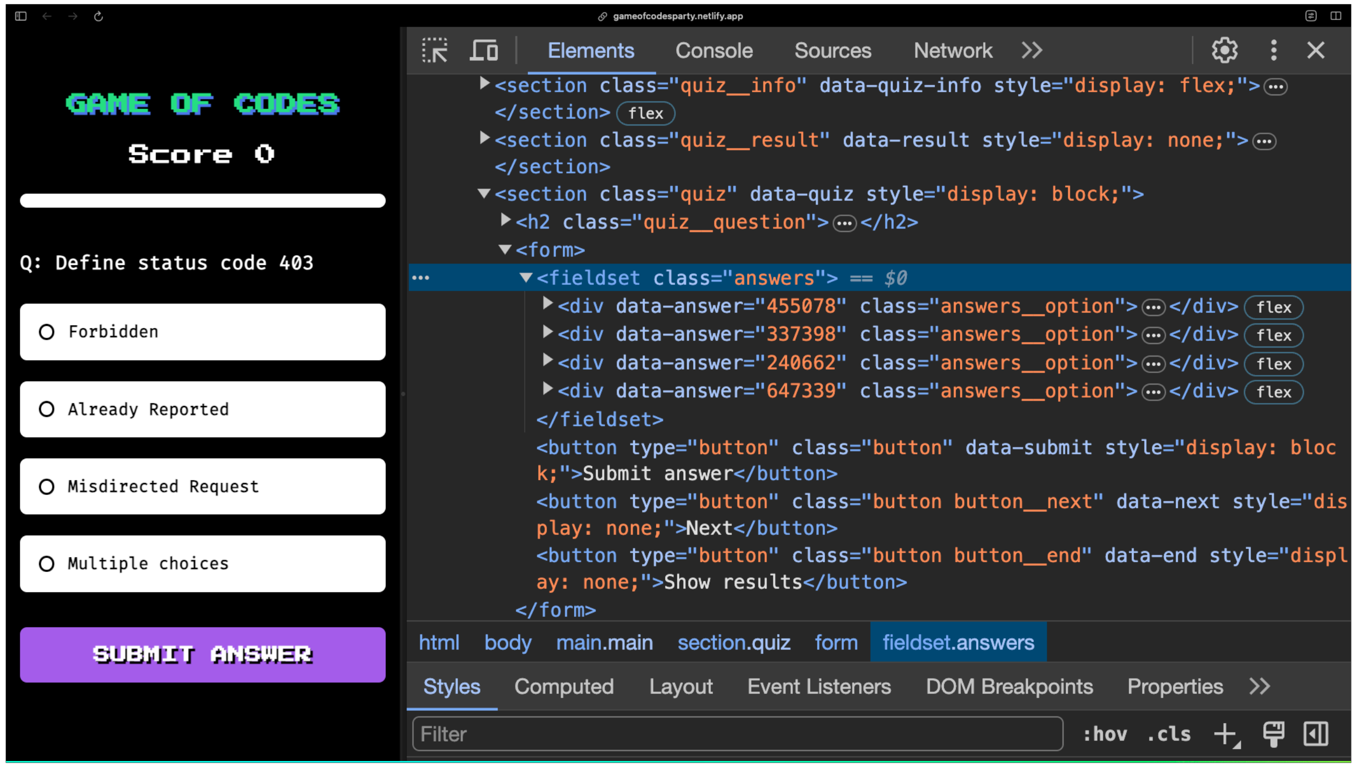


```
function resetGame() {
  showStartButton();
  hideResult();
  hideEndButton();
  hideQuizInfo();
  hideQuiz();
  hideResetButton();
  uncheckAllAnswers();
  resetGameState();
  resetProgressBar();
  clearResultsBlock();
  updateScore();
  initTotalQuestions();
}

function hideResult() {
  result.style.display = "none";
}

function showResult() {
  result.style.display = "block";
}
```

And the whole thing is written with a functional programming approach
So it's easily testable
(There are no tests lol)

And you might think, oh this is easy. Salma has used plain JavaScript. The HTML must have an identifier for the correct answer, so the JavaScript knows which one I have selected to determine the outcome. So I can just inspect the DOM to find the correct answer and get 10/10 every time. Surely Salma hasn't thought about this. Surely I can game the system.

I did think about this. Of course.



GAME OF CODES

Score 0

Q: Define status code 403

○ Forbidden

○ Already Reported

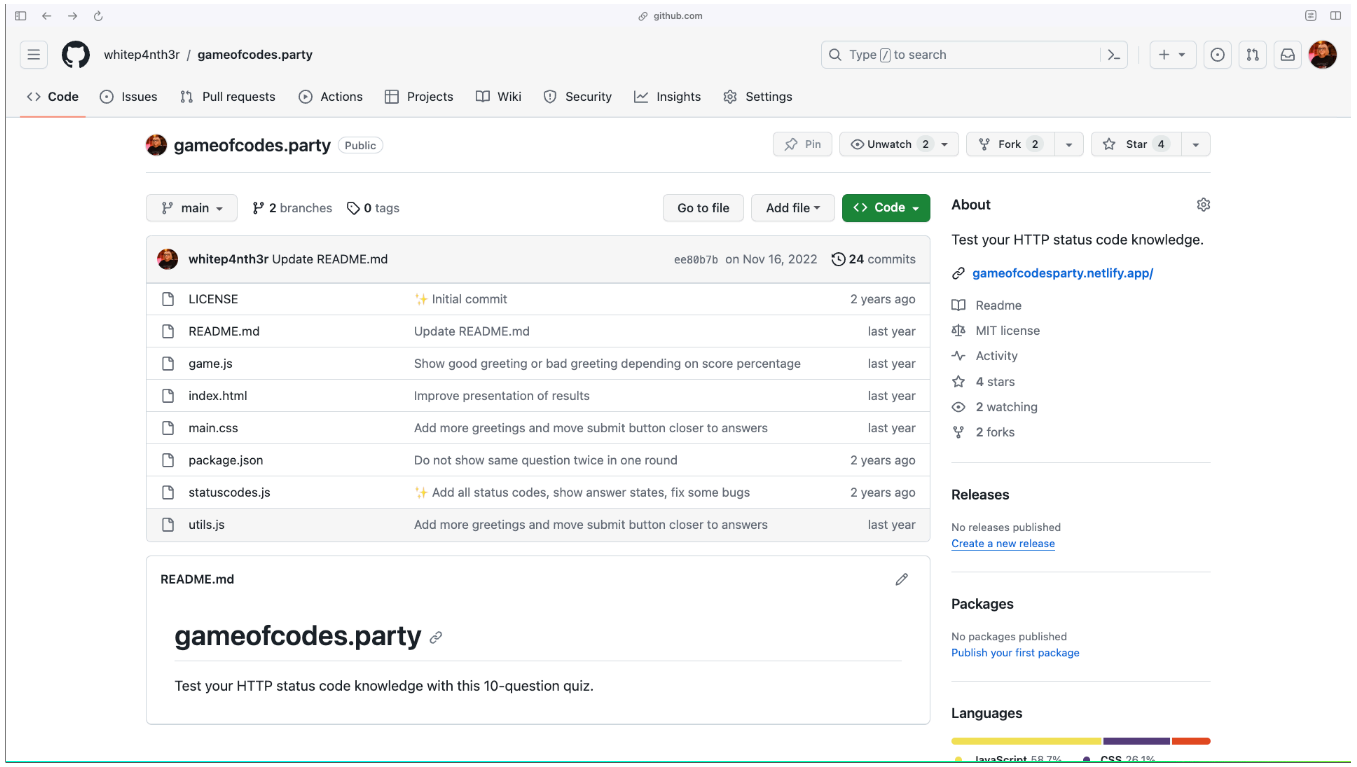
○ Misdirected Request

○ Multiple choices

SUBMIT ANSWER

```
▶<section class="quiz__info" data-quiz-info style="display: flex;">⋯
 </section> flex
▶<section class="quiz__result" data-result style="display: none;">⋯
 </section>
▼<section class="quiz" data-quiz style="display: block;">
  ▶<h2 class="quiz__question">⋯</h2>
  ▼<form>
    ▼<fieldset class="answers"> == $0
      ▶<div data-answer="455078" class="answers__option">⋯</div> flex
      ▶<div data-answer="337398" class="answers__option">⋯</div> flex
      ▶<div data-answer="240662" class="answers__option">⋯</div> flex
      ▶<div data-answer="647339" class="answers__option">⋯</div> flex
     </fieldset>
     <button type="button" class="button" data-submit style="display: bloc
     k;">Submit answer</button>
     <button type="button" class="button button__next" data-next style="dis
     play: none;">Next</button>
     <button type="button" class="button button__end" data-end style="displ
     ay: none;">Show results</button>
   </form>
```

Each answer generated is given a string of “random” numbers to identify it by. One of them is correct, the game state knows which one, but there’s no other way to know in the HTML. Smort.



whitep4nth3r Update README.md ee80b7b on Nov 16, 2022 24 commits

LICENSE	✨ Initial commit	2 years ago
README.md	Update README.md	last year
game.js	Show good greeting or bad greeting depending on score percentage	last year
index.html	Improve presentation of results	last year
main.css	Add more greetings and move submit button closer to answers	last year
package.json	Do not show same question twice in one round	2 years ago
statuscodes.js	✨ Add all status codes, show answer states, fix some bugs	2 years ago
utils.js	Add more greetings and move submit button closer to answers	last year

README.md

gameofcodes.party

Test your HTTP status code knowledge with this 10-question quiz.

About

Test your HTTP status code knowledge.

gameofcodesparty.netlify.app/

Readme
MIT license
Activity
4 stars
2 watching
2 forks

Releases

No releases published
Create a new release

Packages

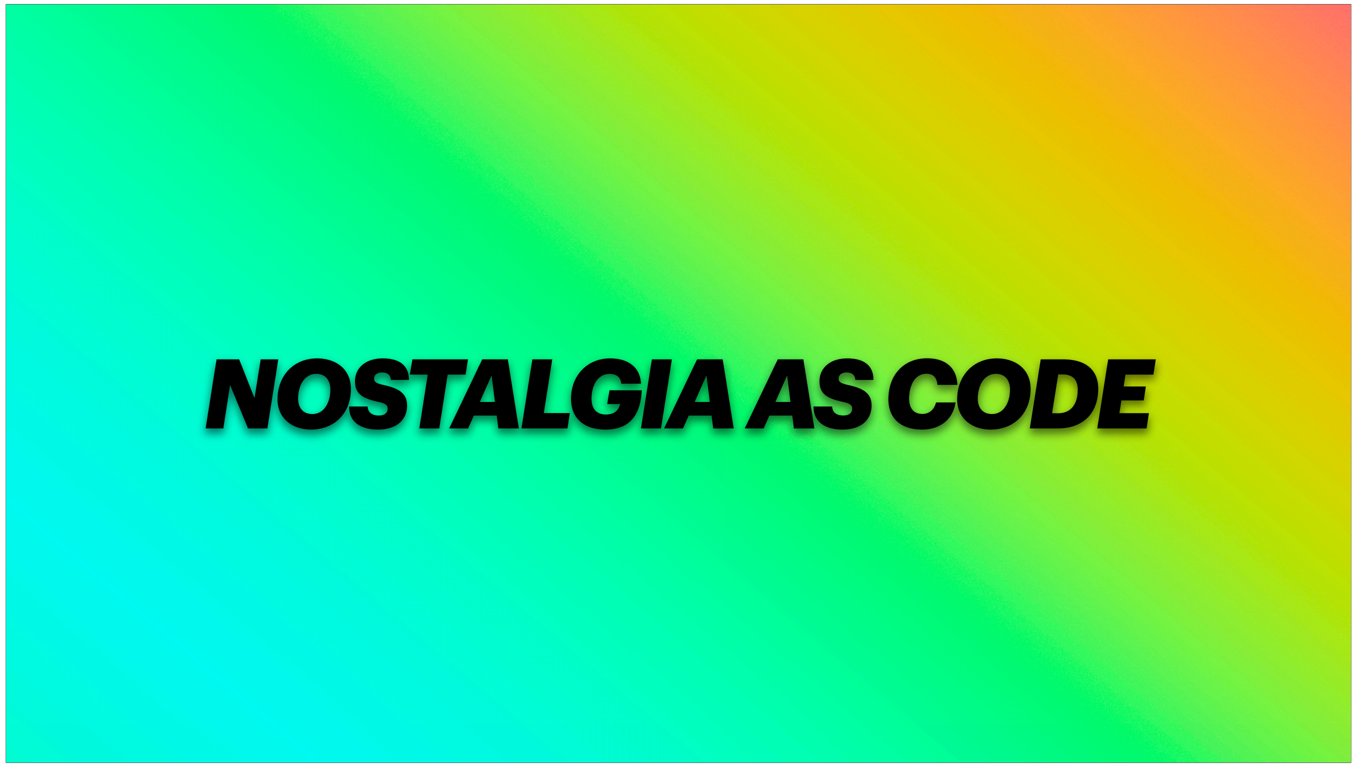
No packages published
Publish your first package

Languages

● JavaScript 58.7% ● CSS 26.1%

I did have bigger plans for this, authentication via GitHub, a leaderboard, competitions…

But at this point I was also learning that some projects don’t really need to be finished and can exist just as they are.

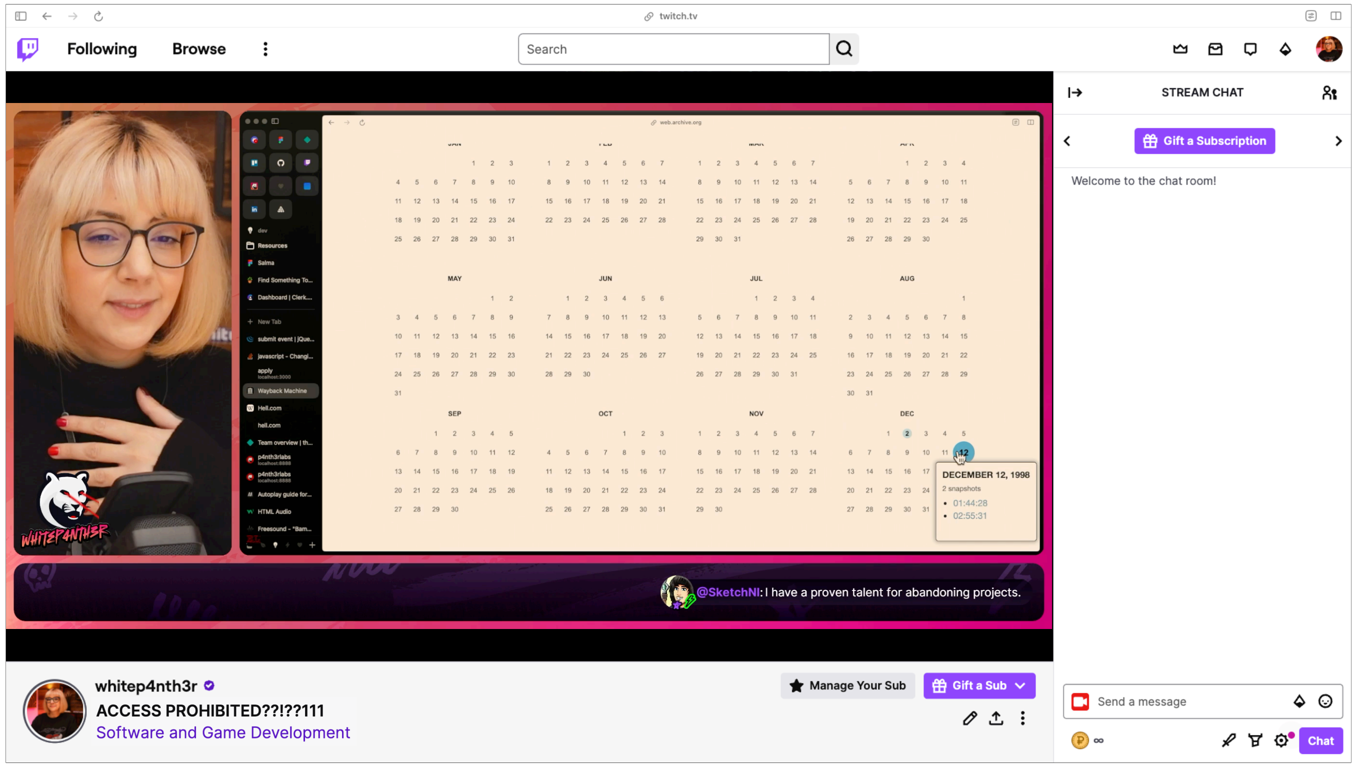


NOSTALGIA AS CODE

As well as sharing my love for geocities
I’ve been sharing my love for the most enchanting, intriguing website of all time

I’ve been recreating a website that has been burned into my brain since I first encountered it in 1999,
it’s a website that doesn’t exist anymore

hell.com



Welcome to the chat room!

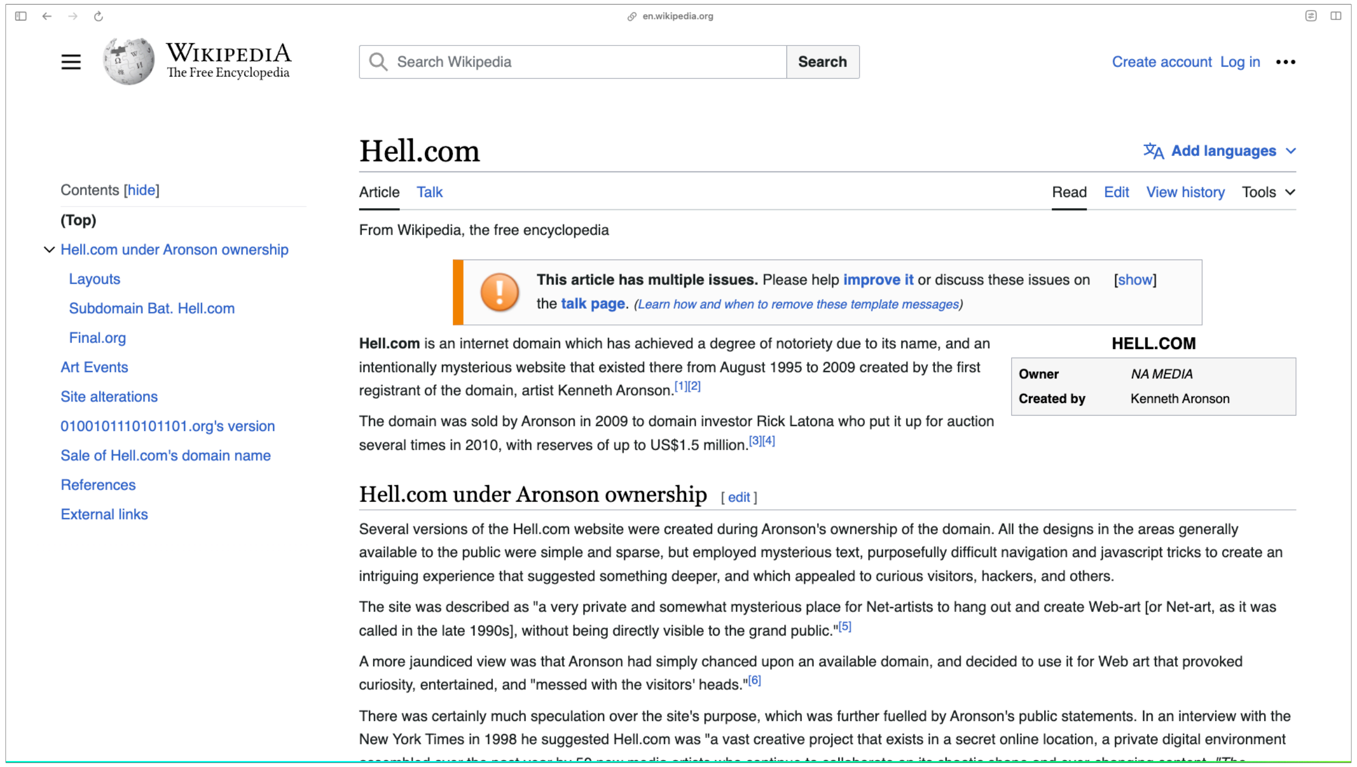
DECEMBER 12, 1998
2 snapshots
- 01:44:28
- 02:55:31

@SketchNl: I have a proven talent for abandoning projects.

whitep4nth3r
ACCESS PROHIBITED??!??111
Software and Game Development

It’s just google

When this happened to me, as a teenager… I believed it



Hell.com

From Wikipedia, the free encyclopedia

This article has multiple issues. Please help **improve it** or discuss these issues on the **talk page**. *(Learn how and when to remove these template messages)* [show]

HELL.COM

Owner	*NA MEDIA*
Created by	Kenneth Aronson

Hell.com is an internet domain which has achieved a degree of notoriety due to its name, and an intentionally mysterious website that existed there from August 1995 to 2009 created by the first registrant of the domain, artist Kenneth Aronson.[1][2]

The domain was sold by Aronson in 2009 to domain investor Rick Latona who put it up for auction several times in 2010, with reserves of up to US$1.5 million.[3][4]

Hell.com under Aronson ownership [edit]

Several versions of the Hell.com website were created during Aronson's ownership of the domain. All the designs in the areas generally available to the public were simple and sparse, but employed mysterious text, purposefully difficult navigation and javascript tricks to create an intriguing experience that suggested something deeper, and which appealed to curious visitors, hackers, and others.

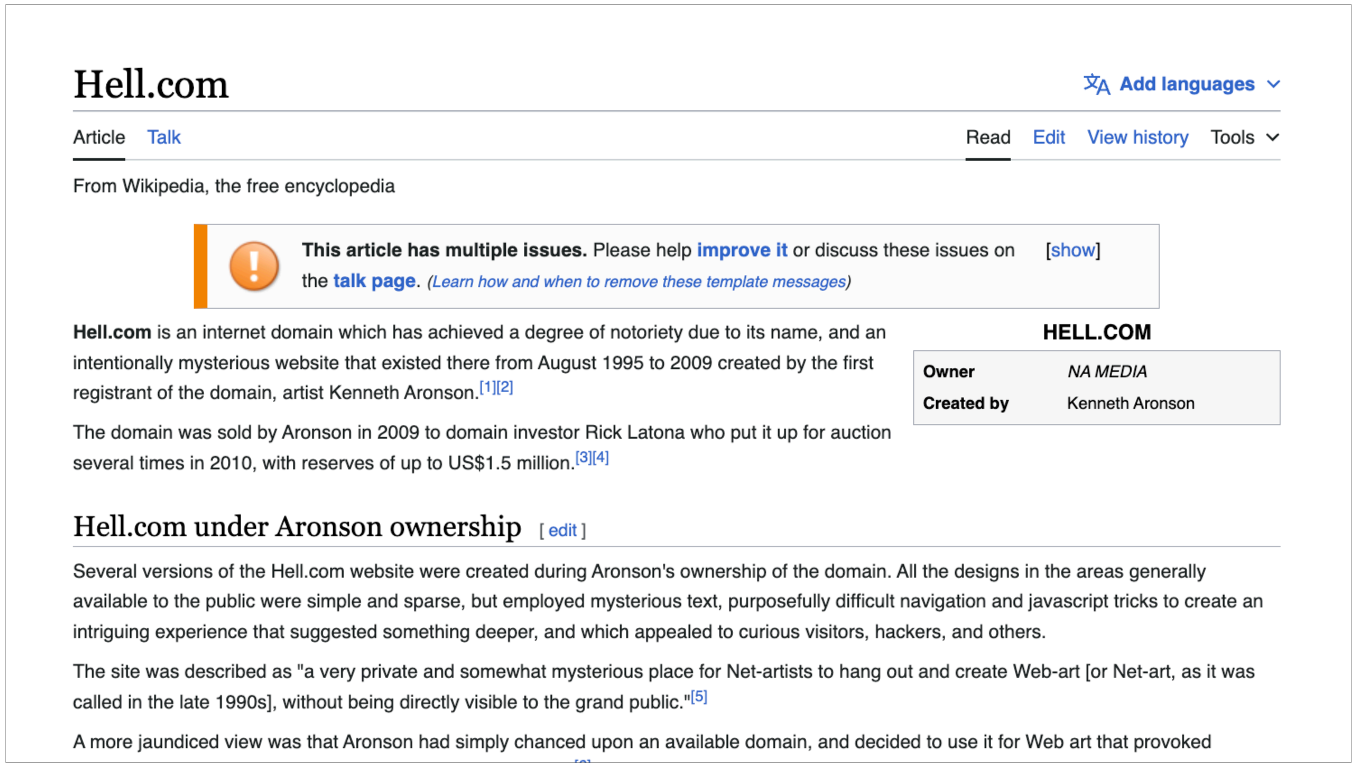
The site was described as "a very private and somewhat mysterious place for Net-artists to hang out and create Web-art [or Net-art, as it was called in the late 1990s], without being directly visible to the grand public."[5]

A more jaundiced view was that Aronson had simply chanced upon an available domain, and decided to use it for Web art that provoked curiosity, entertained, and "messed with the visitors' heads."[6]

There was certainly much speculation over the site's purpose, which was further fuelled by Aronson's public statements. In an interview with the New York Times in 1998 he suggested Hell.com was "a vast creative project that exists in a secret online location, a private digital environment

You can read up on all the lore on wikipedia

But the TLDR is that



Hell.com

From Wikipedia, the free encyclopedia

This article has multiple issues. Please help **improve it** or discuss these issues on the **talk page**. *(Learn how and when to remove these template messages)* [show]

Hell.com is an internet domain which has achieved a degree of notoriety due to its name, and an intentionally mysterious website that existed there from August 1995 to 2009 created by the first registrant of the domain, artist Kenneth Aronson.[1][2]

The domain was sold by Aronson in 2009 to domain investor Rick Latona who put it up for auction several times in 2010, with reserves of up to US$1.5 million.[3][4]

HELL.COM

Owner	*NA MEDIA*
Created by	Kenneth Aronson

Hell.com under Aronson ownership [edit]

Several versions of the Hell.com website were created during Aronson's ownership of the domain. All the designs in the areas generally available to the public were simple and sparse, but employed mysterious text, purposefully difficult navigation and javascript tricks to create an intriguing experience that suggested something deeper, and which appealed to curious visitors, hackers, and others.

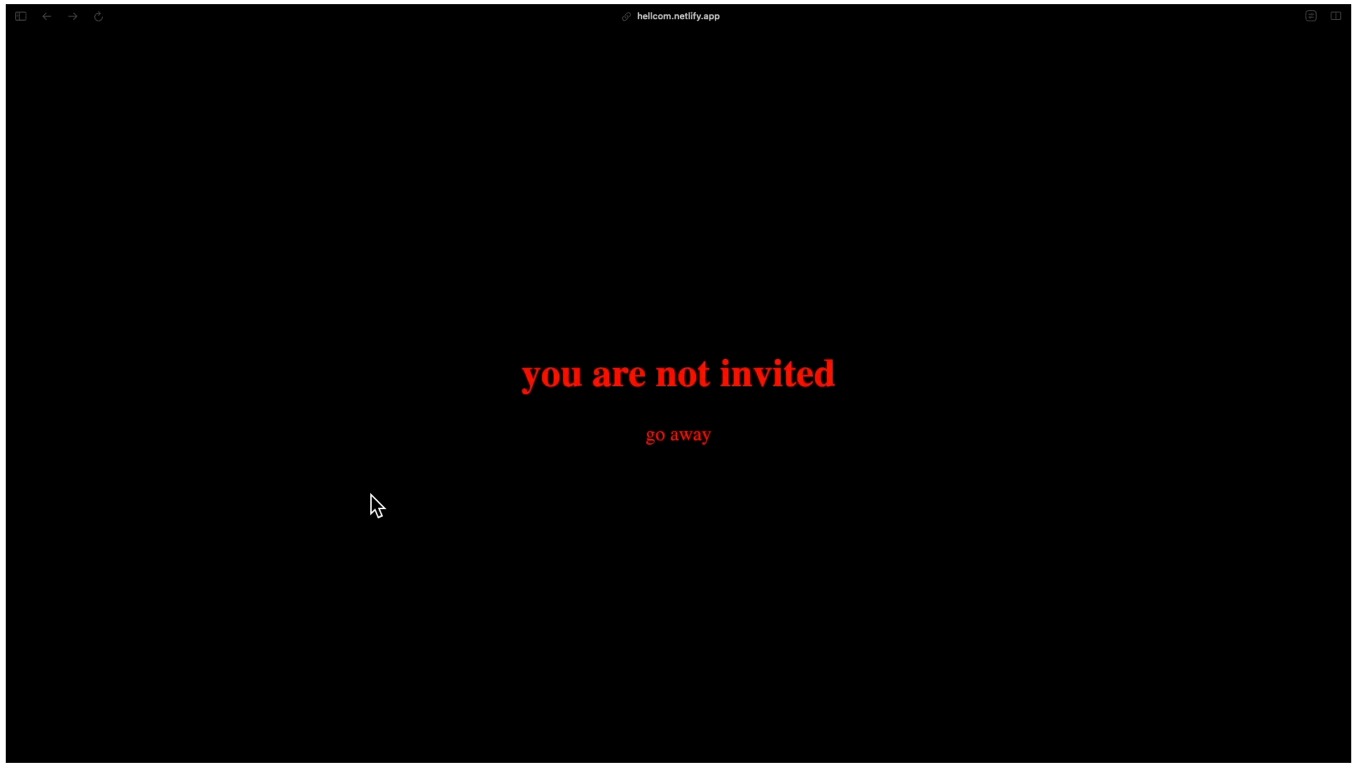
The site was described as "a very private and somewhat mysterious place for Net-artists to hang out and create Web-art [or Net-art, as it was called in the late 1990s], without being directly visible to the grand public."[5]

A more jaundiced view was that Aronson had simply chanced upon an available domain, and decided to use it for Web art that provoked

hell.com was an intentionally mysterious website that existed from 1995 to 2009

And I wanted to recreate that mystery on the modern web

Now there are too many easter eggs, too many things to share and not enough time
But I used a combination of the pages and user journeys described on wikipedia, and ones that I found on the wayback machine to recreate it



you are not invited

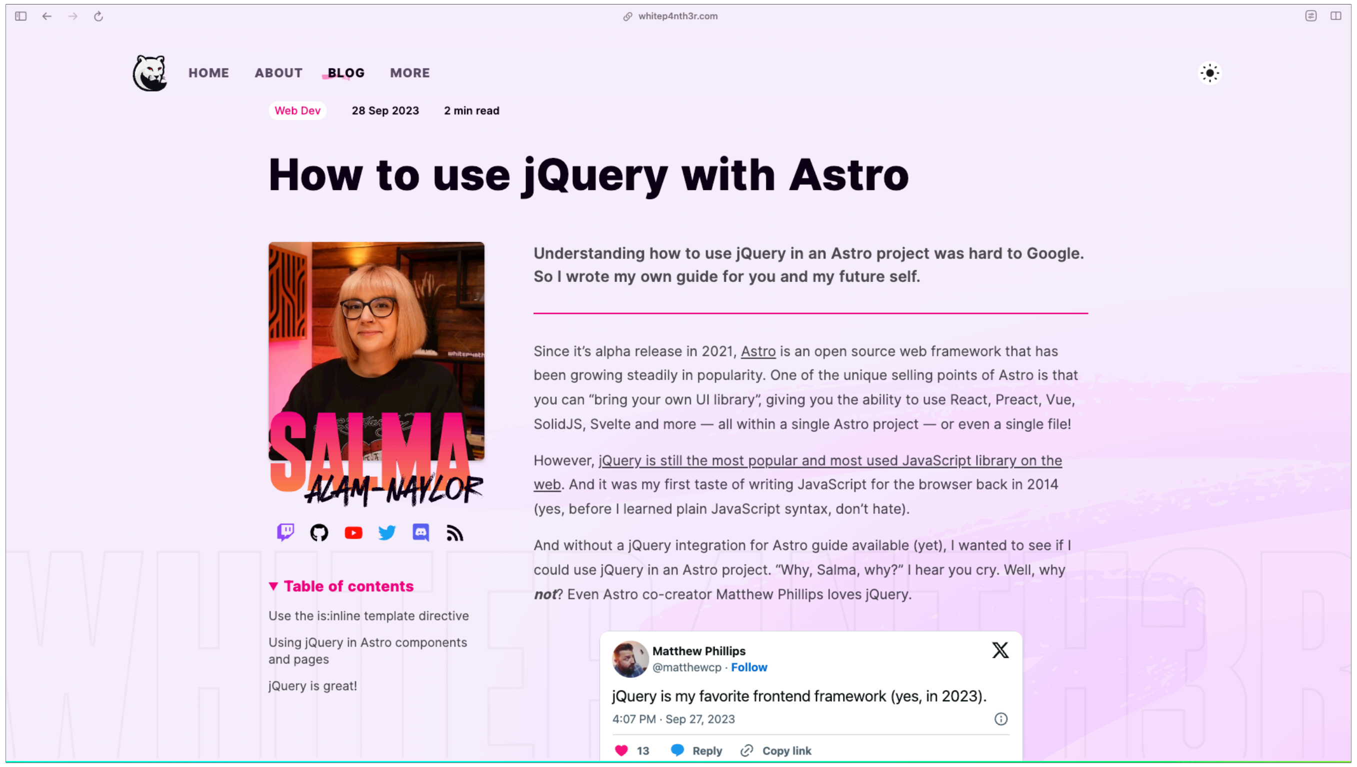
go away

Here’s just me randomly clicking around (even I don’t know what to click to get what part of the journey)

…

It’s built with Astro and jQuery — jQuery because I wanted to preserve that nostalgia of an earlier web (even though jQuery is still the most popular JavaScript library used on the web today)

And it was a little tricky to get jQuery working in Astro… so I wrote about it on my blog



Web Dev 28 Sep 2023 2 min read

How to use jQuery with Astro

Understanding how to use jQuery in an Astro project was hard to Google. So I wrote my own guide for you and my future self.

Since it's alpha release in 2021, Astro is an open source web framework that has been growing steadily in popularity. One of the unique selling points of Astro is that you can "bring your own UI library", giving you the ability to use React, Preact, Vue, SolidJS, Svelte and more — all within a single Astro project — or even a single file!

However, jQuery is still the most popular and most used JavaScript library on the web. And it was my first taste of writing JavaScript for the browser back in 2014 (yes, before I learned plain JavaScript syntax, don't hate).

And without a jQuery integration for Astro guide available (yet), I wanted to see if I could use jQuery in an Astro project. "Why, Salma, why?" I hear you cry. Well, why ***not***? Even Astro co-creator Matthew Phillips loves jQuery.

▼ Table of contents

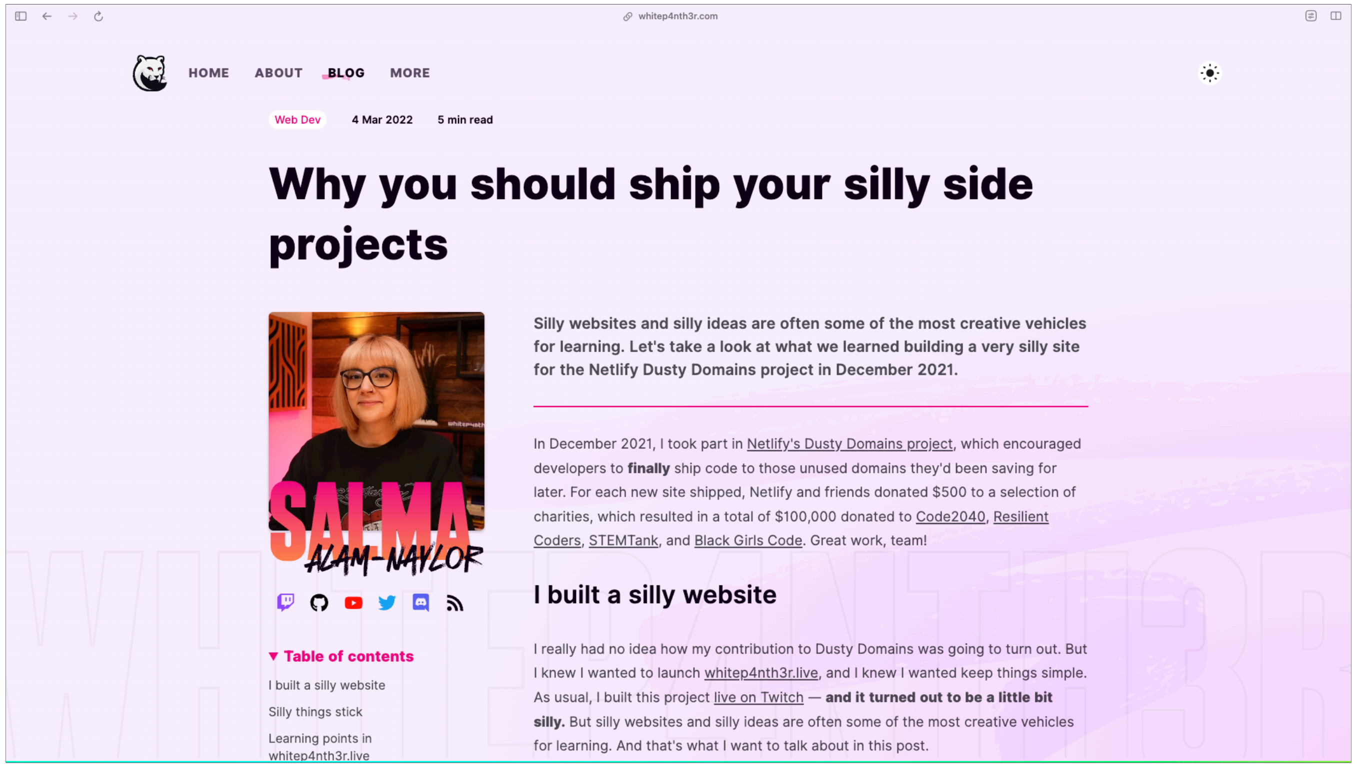
Matthew Phillips
@matthewcp · Follow

jQuery is my favorite frontend framework (yes, in 2023).

4:07 PM · Sep 27, 2023

If you like to blog, or if you're just thinking about it, the best things you can write about are things that were difficult, and problems you solved, chances are someone else has experienced this
You can help other people and also yourself when you forget how to do something

And if I have one more closing piece of advice for you



Web Dev 4 Mar 2022 5 min read

Why you should ship your silly side projects

Silly websites and silly ideas are often some of the most creative vehicles for learning. Let's take a look at what we learned building a very silly site for the Netlify Dusty Domains project in December 2021.

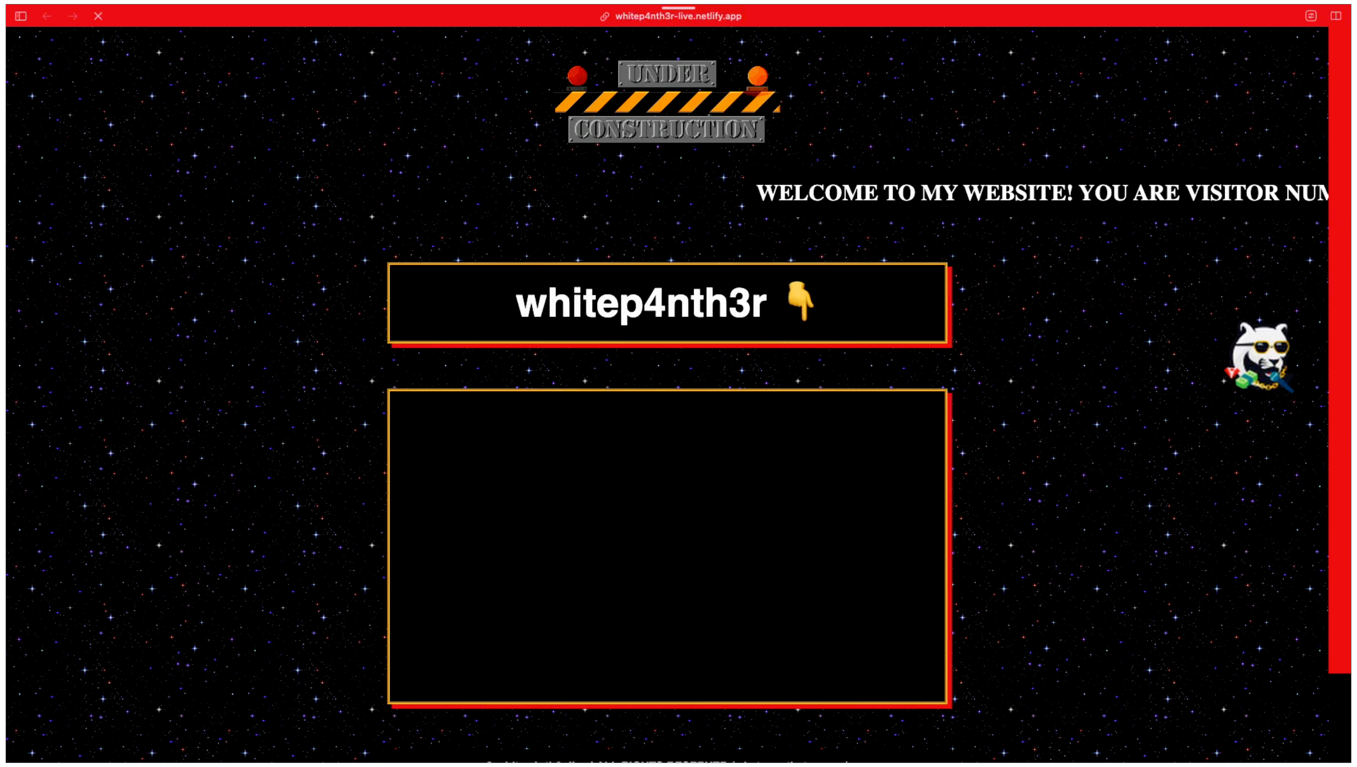
In December 2021, I took part in Netlify's Dusty Domains project, which encouraged developers to **finally** ship code to those unused domains they'd been saving for later. For each new site shipped, Netlify and friends donated $500 to a selection of charities, which resulted in a total of $100,000 donated to Code2040, Resilient Coders, STEMTank, and Black Girls Code. Great work, team!

I built a silly website

I really had no idea how my contribution to Dusty Domains was going to turn out. But I knew I wanted to launch whitep4nth3r.live, and I knew I wanted keep things simple. As usual, I built this project live on Twitch — **and it turned out to be a little bit silly.** But silly websites and silly ideas are often some of the most creative vehicles for learning. And that's what I want to talk about in this post.

▼ Table of contents

It’s to ship your silly side projects, have fun



WELCOME TO MY WEBSITE! YOU ARE VISITOR NUM

Like this one

Whitep4nth3r live
It just tells you whether I’m live on Twitch or not
It takes more inspiration from geocities, plays with CSS, iframes, the fake marquee makes another appearance

It’s nonsense
But I learned a lot from making this seemingly stupid and pointless website

Because silly things stick
I remember vividly in primary school, when we were learning about the effects of alcohol.
Instead of just standing at the front of the class and telling us how dangerous alcohol was — our teacher spent the lesson wobbling around the classroom, slurring his speech, and generally being silly, making us laugh. But we learned.

I’ve never forgotten this lesson, and I’ve always drawn on this approach for inspiration when I was a music teacher, and it’s one of the core driving principals of…

ENTERTAINMENT AS CODE

entertainment as code

SALMA

WHITEP4NTH3R

ALAM-NAYLOR

ENTERTAINMENT AS CODE

I’m Salma, I write code for your entertainment

Find me on the Internet everywhere as whitep4nth3r (white p 4 nth 3 r) [click]

And I look forward to entertaining you, with code, very soon

[redacted]