



Porto.Vue

Vue2Vite

A Developer Experience Sidequest 

Vue2Vite

A Developer Experience Sidequest 

Tiago Pereira

Vue2Vite

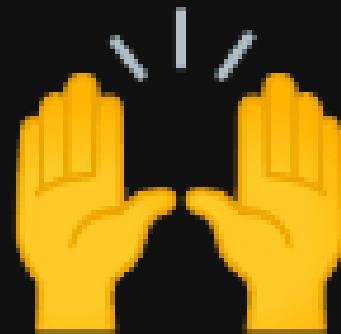
A Developer Experience Sidequest 

Tiago Pereira

Front End developer @ Infraspeak

First things first

Show of hands



Hands up if you are...

Hands up if you are...

- Using Vite in production

Hands up if you are...

- Using Vite in production
 - With Vue2

Hands up if you are...

- Using Vite in production
 - With Vue2
 - With Vue3

Hands up if you are...

- Using Vite in production
 - With Vue2
 - With Vue3
 - With something else 

and what about...

and what about...

- Having migrated a Vue2 app to Vite?

Good!

Good!

moving on...

The Problem

- Our app is built using Vue2
- That means it's built using webpack
- Which can become slow over time as complexity increases

The Problem

- Our app is built using Vue2
- That means it's built using webpack
- Which can become slow over time as complexity increases
- Really slow 

The Problem in numbers

npm run serve

```
DONE Compiled successfully in 90699ms
```

```
No type errors found
```

```
Version: typescript 4.0.5
```

```
Time: 33862ms
```

npm run serve

- 90 seconds on average
- Would be fine if this happened once per day

npm run serve

- 90 seconds on average
- Would be fine if this happened once per day

Have breakfast

npm run serve

- 90 seconds on average
- Would be fine if this happened once per day

Have breakfast

Drop the kid at kindergarten

npm run serve

- 90 seconds on average
- Would be fine if this happened once per day

Have breakfast

Drop the kid at kindergarten

npm run serve

npm run serve

- 90 seconds on average
- Would be fine if this happened once per day

Have breakfast

Drop the kid at kindergarten

npm run serve

Right?

Wrong

- Let's be pedantic
- 2022 has got 229 working days
 - Maybe not as many, second kid is due this year 
- Grand total of 5.725 hours wasted!
- That's almost a whole day's worth of work per year!

Wronger still

Let's get real and execute a simple code change
and watch the server reload

Simple change I

```
DONE Compiled successfully in 6503ms
```

```
Type checking in progress...
```

- 6 whole seconds
- But wait (no pun intended), it's still type checking!

Simple change II

```
DONE Compiled successfully in 6503ms
```

```
Type checking in progress...
```

```
App running at:
```

- Local: secret localhost
- Network: secret host

```
No type errors found
```

```
Version: typescript 4.0.5
```

```
Time: 27550ms
```

- 27 seconds
- Waiting times are starting to compound
- I'm also being mean: the view does not wait for type checking to finish

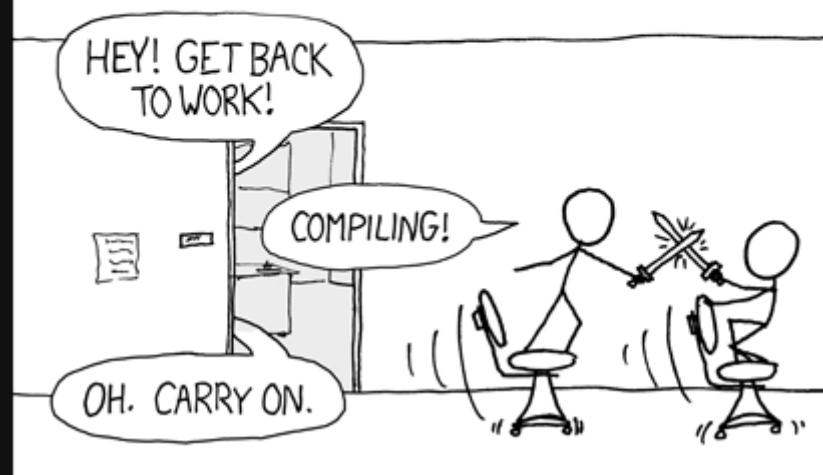
Time for some guesstimates

- 20 changes per hour
- 8 hours per day
- 6 seconds per change on average
- That adds up to 24 minutes waiting per day
- Remember those 229 working days?

Waste

- That's almost 92 hours wasted per year
- Not so pedantic now

THE #1 PROGRAMMER EXCUSE
FOR LEGITIMATELY SLACKING OFF:
"MY CODE'S COMPILING."



It gets worse

- We use a containerized development environment
- That means node also runs inside a container
- Works fine for 99% of devs at Infraspeak
- Except for me and another poor soul
- Neither of us is using vanilla Ubuntu
 - He uses Mint
 - And I btw use Ar... Manjaro.

It gets worse

- We use a containerized development environment
- That means node also runs inside a container
- Works fine for 99% of devs at Infraspeak
- Except for me and another poor soul
- Neither of us is using vanilla Ubuntu
 - He uses Mint
 - And I btw use Ar... Manjaro.
 - That's Arch Linux for dummies like me

Happy mode

```
sde
```

Infraspeak Developm 0* > stats

NAME	CPU %
web-core-api_infraspeak.consumers_1	1.00%
web-core-api_infraspeak.api.server_1	0.00%
web-core-api_infraspeak.direct.assets_1	0.00%
web-core-api_infraspeak.kafka.ui_1	0.07%
web-core-api_infraspeak.zookeeper_1	0.09%
web-core-api_infraspeak.postgres_1	0.00%
web-core-api_infraspeak.redis_1	0.11%
web-core-api_infraspeak.memcached_1	0.01%
web-core-api_infraspeak.api_1	0.00%
web-core-api_infraspeak.horizon_1	0.10%
web-core-api_infraspeak.kafka_1	0.51%
mailhog_mailhog_1	0.00%
traefik-traefik-1	0.00%
web-core-legacy_infraspeak.legacy.assets_1	0.00%
web-core-legacy_infraspeak.legacy_1	0.00%
web-core-legacy_infraspeak.legacy.server_1	0.00%
web-core-client_infraspeak.web-server_1	0.00%
web-core-client_infraspeak.web_1	286.94%

↑ 5h 24m 17s < 6.9 9.2 6.2 < 2022-05-18 < 15:15 < tpereira

```
/app/node_modules/cache-loader/dist/cjs.js??ref--0-0!/app/node_modules/vue-loader/index.js??vue-loader-options!/app/src/components/info/quote/InfoQuote.vue?vue&type=template&id=f80fc3ac& 2022-05-18T14:15:28.360201108Z <s> [webpack.Progress] 61% building 473/549 modules 76 active /app/node_modules/lottie-web-vue/dist/lottie-web-vue.esm.js 2022-05-18T14:15:28.361655591Z <s> [webpack.Progress] 61% building 474/549 modules 75 active /app/node_modules/lottie-web-vue/dist/lottie-web-vue.esm.js 2022-05-18T14:15:28.362812587Z <s> [webpack.Progress] 61% building 474/550 modules 76 active /app/node_modules/vue-loader/lib/loaders/pitcher.js??ref--4!/app/node_modules/cache-loader/dist/cjs.js??ref--0-0!/app/node_modules/vue-loader/lib/index.js??vue-loader-options!/app/node_modules/eslint-loader/index.js??ref--13-0!/app/src/components/common/search/IsSearch.vue?vue&type=template&id=1d391b85& 2022-05-18T14:15:28.363159509Z <s> [webpack.Progress] 61% building 475/550 modules 75 active /app/node_modules/lottie-web-vue/dist/lottie-web-vue.esm.js 2022-05-18T14:15:28.363701999Z <s> [webpack.Progress] 61% building 475/551 modules 76 active /app/node_modules/cache-loader/dist/cjs.js?{"cacheDirectory": "node_modules/.cache/vue-loader", "cacheIdentifier": "00042458-vue-loader-template"}!/app/node_modules/vue-loader/lib/loaders/templateLoader.js??vue-loader-options!/app/node_modules/cache-loader/dist/cjs.js??ref--0-0!/app/node_modules/vue-loader/index.js??vue-loader-options!/app/src/components/common/search/IsSearch.vue?vue&type=template&id=1d391b85&
```

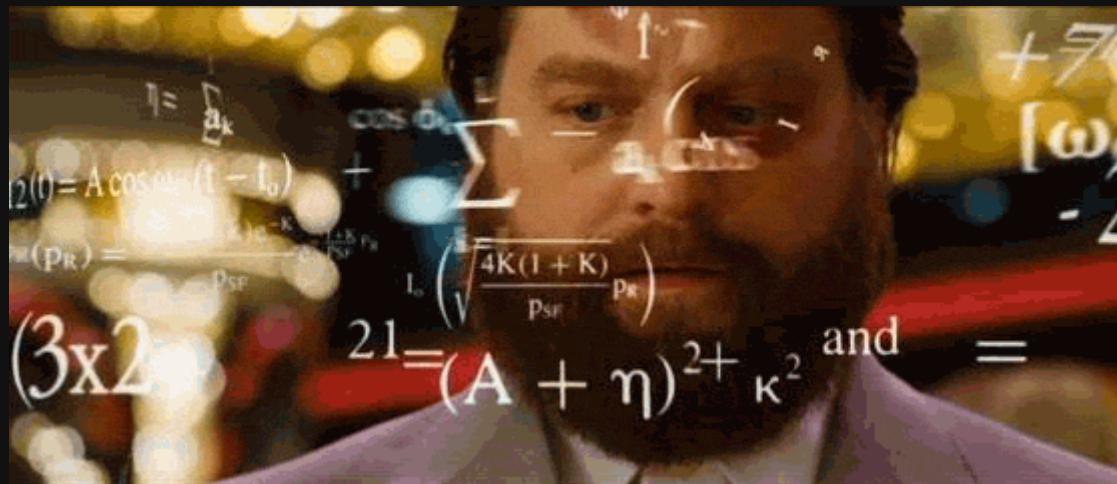
Berserk mode

```
sde
Infraspeak Developm 0* > stats
2022-05-18T14:15:44.675771456Z <s> [webpack.Progress] 67% building 1649/1719 modules 70 active /app/node_modules/faker/lib/locales/tr/phone_number/index.js
2022-05-18T14:15:44.676520446Z <s> [webpack.Progress] 67% building 1650/1719 modules 69 active /app/node_modules/faker/lib/locales/tr/phone_number/index.js
2022-05-18T14:15:44.678079624Z <s> [webpack.Progress] 67% building 1650/1720 modules 70 active /app/node_modules/faker/lib/locales/tr/lorem/index.js
2022-05-18T14:15:44.678490626Z <s> [webpack.Progress] 67% building 1651/1720 modules 69 active /app/node_modules/faker/lib/locales/tr/lorem/index.js
2022-05-18T14:15:44.678878837Z <s> [webpack.Progress] 67% building 1651/1721 modules 70 active /app/node_modules/faker/lib/locales/sv/address/index.js
2022-05-18T14:15:44.679143884Z <s> [webpack.Progress] 67% building 1651/1722 modules 71 active /app/node_modules/faker/lib/locales/sv/date/index.js
2022-05-18T14:15:44.680905940Z <s> [webpack.Progress] 67% building 1652/1722 modules 70 active /app/node_modules/faker/lib/locales/sv/date/index.js
2022-05-18T14:15:44.681245629Z <s> [webpack.Progress] 67% building 1653/1722 modules 69 active /app/node_modules/faker/lib/locales/sv/date/index.js
2022-05-18T14:15:44.681572071Z <s> [webpack.Progress] 67% building 1654/1722 modules 68 active /app/node_modules/faker/lib/locales/sv/date/index.js
2022-05-18T14:15:44.681886515Z <s> [webpack.Progress] 67% building 1655/1722 modules 67 active /app/node_modules/faker/lib/locales/sv/date/index.js
2022-05-18T14:15:44.682082087Z <s> [webpack.Progress] 67% building 1655/1723 modules 68 active /app/node_modules/faker/lib/locales/sv/team/index.js
↑ 5h 24m 55s < 24.1 12.7 7.5 < 2022-05-18 < 15:16 < tpereira
NAME                                     CPU %
web-core-api_infraspeak.consumers_1      137.36%
web-core-api_infraspeak.api.server_1       0.00%
web-core-api_infraspeak.direct.assets_1    38.06%
web-core-api_infraspeak.kafka.ui_1         87.08%
web-core-api_infraspeak.zookeeper_1        78.12%
web-core-api_infraspeak.postgres_1          45.07%
web-core-api_infraspeak.redis_1             39.00%
web-core-api_infraspeak.memcached_1         69.00%
web-core-api_infraspeak.api_1               47.19%
web-core-api_infraspeak.horizon_1           90.31%
web-core-api_infraspeak.kafka_1              150.14%
mailhog_mailhog_1                           0.00%
traefik-traefik-1                          90.33%
web-core-legacy_infraspeak.legacy.assets_1   42.40%
web-core-legacy_infraspeak.legacy_1           77.98%
web-core-legacy_infraspeak.legacy.server_1    0.00%
web-core-client_infraspeak.web-server_1      0.00%
web-core-client_infraspeak.web_1              97.15%
```

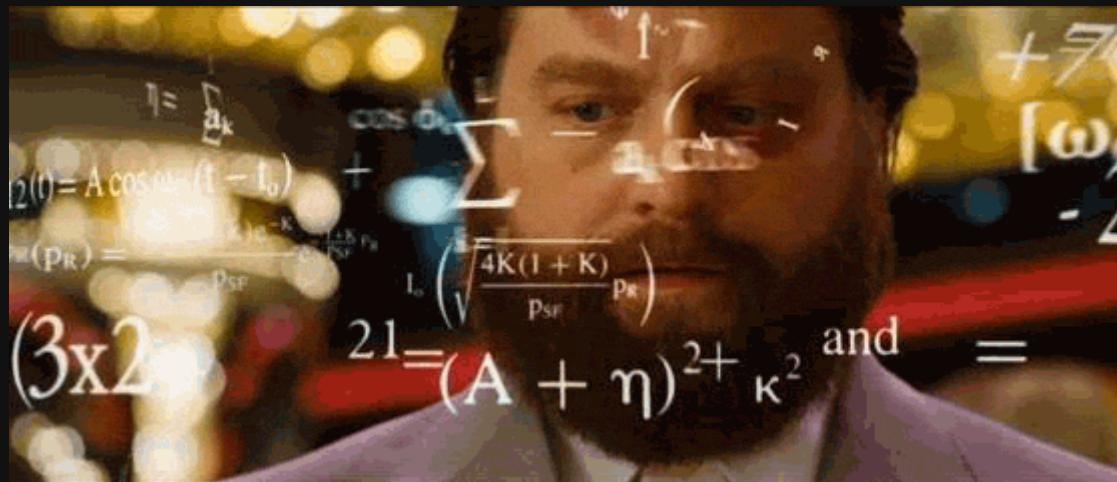
Not happy

- Need to restart the node container
- Which takes 90 seconds on average
- But now multiple times a day
- Math time again!

Just kidding



Just kidding



moving on...

We want happy devs

- This can mean different things to different people... developers
- It's not one size fits all
- If anything shortening the loop between writing code and seeing it happen is good

Defining DX

There is nothing more directly DX than the experience of typing code into code editing software and running it. That's what "coding" is and that's what developers do. It's no wonder that developers take that experience seriously and are constantly trying to improve it for themselves and their teams.

We could

- Split up the app into modules
- Give each dev a beefy workstation
- Use other tricks to speed things up

Or maybe...

- Vite is all the rage right?
- It's super fast and efficient
- Why not use it and ditch webpack?

Great timing!

Now I have something to present at PortoVue!



But wait...

- Could we even use Vite?
- From what I had heard it was only for Vue3
- We didn't want to do that (not right now anyway)
 - ElementUI
 - Class based components (prob not a concern anymore)

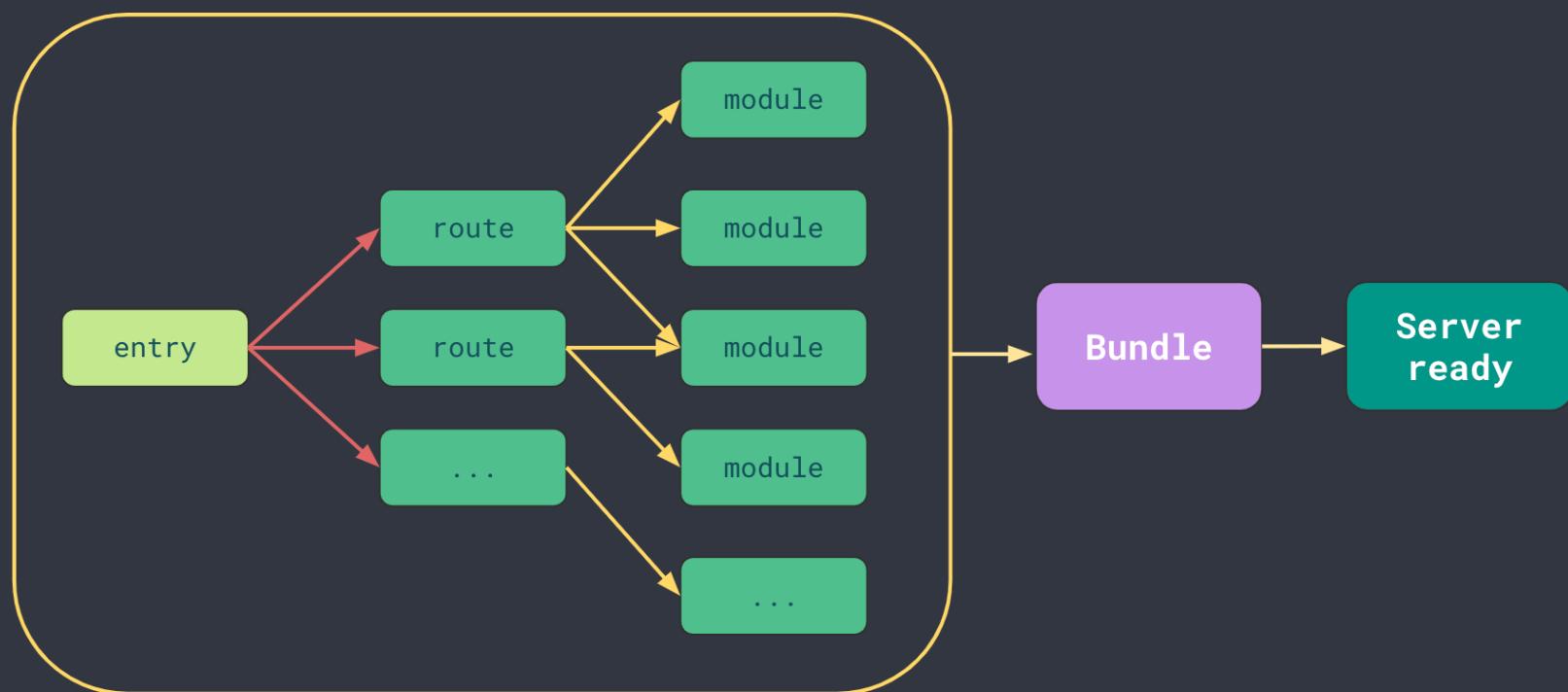
But how?

Let's see why Vite is so fast

Cool digrams snatched from [vitejs.dev](#) and [esbuild.github.io](#)

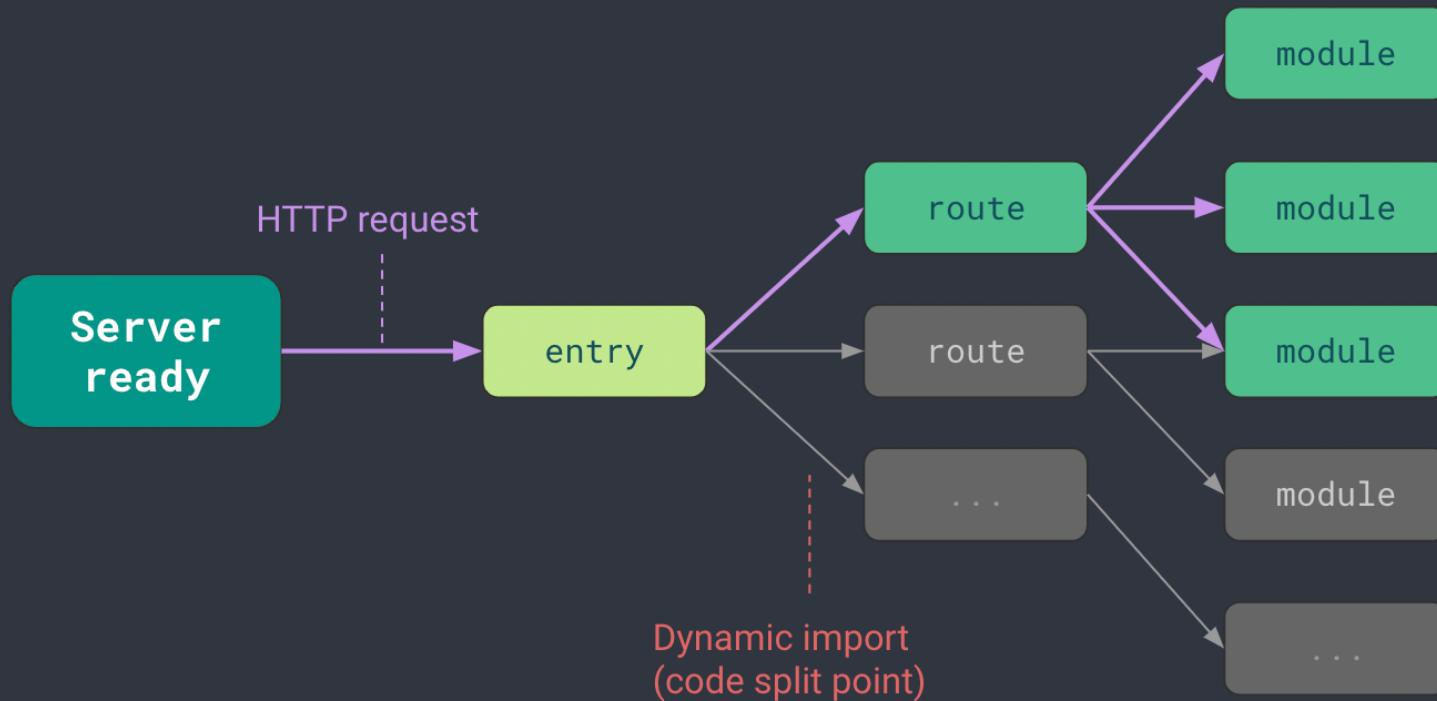
Classic Bundlers

Bundle based dev server



ESM Bundlers

Native ESM based dev server



Dependency Pre- Bundling

esbuild

An extremely fast JavaScript bundler



Above: the time to do a production bundle of 10 copies of the [three.js](#) library from scratch using default settings, including minification and source maps. More info [here](#).

Other tricks

- Route-based code splitting
- Native ESM on the browser
 - Effectively turning the browser into a bundler
 - dev server only
 - Bye bye IE (sorry not sorry)

This felt possible



BUT WEIRD!

Ready, set, Vite!

Ask the internet first

- Surely there must be a *simple* way of doing this
 - According to Daniel Kelly of Vue School there is: How to Migrate from Vue CLI to Vite
- Have guide, will follow! 

Update dependencies

```
1 ...
2   "@types/zxcvbn": "4.4.0",
3   "@typescript-eslint/eslint-plugin": "4.6.0",
4   "@typescript-eslint/parser": "4.2.0",
5 -  "@vue/cli-plugin-babel": "^4.5.15",
6 -  "@vue/cli-plugin-eslint": "3.8.0",
7 -  "@vue/cli-plugin-pwa": "4.4.6",
8 -  "@vue/cli-plugin-typescript": "^4.5.15",
9 -  "@vue/cli-plugin-unit-jest": "^4.5.15",
10 -  "@vue/cli-service": "4.4.6",
11 +  "@vitejs/plugin-vue": "2.3.1",
12   "@vue/eslint-config-standard": "5.1.2",
13   "@vue/eslint-config-typescript": "7.0.0",
14   "@vue/test-utils": "1.0.3",
15 ...
16   "postcss-cli": "9.1.0"
```

Update dependencies

```
13      "@vue/eslint-config-typescript": "7.0.0",
14      "@vue/test-utils": "1.0.3",
15      ...
16      "postcss-cli": "9.1.0",
17      "postcss-html": "1.3.0",
18      "purgecss-webpack-plugin": "4.1.3",
19 -      "sass": "1.45.1",
20 -      "sass-loader": "10.1.1",
21 +      "sass": "1.50.1",
22      "standard-version": "9.0.0",
23      "storybook-addon-themes": "6.1.0",
24      "stylelint": "14.1.0",
25      ...
26      "stylelint-scss": "4.0.1",
27      "ts-jest": "26.4.3",
```

Update dependencies

```
22   "standard-version": "9.0.0",
23   "storybook-addon-themes": "6.1.0",
24   "stylelint": "14.1.0",
25 ...
26   "stylelint-scss": "4.0.1",
27   "ts-jest": "26.4.3",
28   "typescript": "4.0.5",
29 +  "vite": "2.9.5",
30 +  "vite-plugin-vue2": "1.9.3",
31   "vue-cli-plugin-cordova": "2.4.1",
32   "vue-loader": "15.9.3",
33   "vue-template-compiler": "2.6.11",
34 +  "vue-tsc": "0.34.9",
35   "webpack-bundle-analyzer": "4.5.0"
36 },
37 "peerDependencies": {
```

Update dependencies

```
22      "standard-version": "9.0.0",
23      "storybook-addon-themes": "6.1.0",
24      "stylelint": "14.1.0",
25  ...
26      "stylelint-scss": "4.0.1",
27      "ts-jest": "26.4.3",
28      "typescript": "4.0.5",
29 +     "vite": "2.9.5",
30 +     "vite-plugin-vue2": "1.9.3",
31      "vue-cli-plugin-cordova": "2.4.1",
32      "vue-loader": "15.9.3",
33      "vue-template-compiler": "2.6.11",
34 +     "vue-tsc": "0.34.9",
35      "webpack-bundle-analyzer": "4.5.0"
36    },
37    "peerDependencies": {
```

Ignore old browsers

- Get rid of Babel
- Update eslint env from node to es2021

Add vite.config.js

- Forget about vue.config.js
- Mapping is not straightforward
- This was a bit of trial and error
- And uh... reading the docs! 😐

Move index.html

- With Vite index.html lives in the root folder
- Replace any webpack configs with hardcoded values

Update scripts

- Replace vue-cli-service commands with vite ones

Update env vars

- `import.meta.env.` takes the place of `process.env.`
- `VUE_APP_` prefix must be updated to `VITE_`

All done! Right?

*"Enjoy a Faster and More Seamless
Development Experience"*

All done! Right?

*"Enjoy a Faster and More Seamless
Development Experience"*

Right?!

npm run dev

```
vite v2.9.9 dev server running at:
```

```
> Local: http://localhost:3000/  
> Network: use `--host` to expose
```

```
ready in 832ms.
```

What? 832 miliseconds? 😱

Let's check the browser...

```
Unable to parse HTML; Illegal tag name. Use '&lt;' to print '<'.
  at {"file":"/index.html","line":22,"column":6}
20 |     <link rel="stylesheet" href="public/basic.css">
21 |
22 |     <% if(process.env.NODE_ENV === 'staging') { %>
|         ^
23 |         <script src="public/new-relic/new-relic-staging.js"></script>
24 |     <% } %>

at {"file":"/index.html","line":22,"column":6}
at handleParseError (/app/node_modules/vite/dist/node/chunks/dep-59dc6e00.js:34681:11)
at traverseHtml (/app/node_modules/vite/dist/node/chunks/dep-59dc6e00.js:34637:9)
at runMicrotasks (<anonymous>)
at processTicksAndRejections (internal/process/task_queues.js:97:5)
at async devHtmlHook (/app/node_modules/vite/dist/node/chunks/dep-59dc6e00.js:56216:5)
at async applyHtmlTransforms (/app/node_modules/vite/dist/node/chunks/dep-59dc6e00.js:35047:21)
at async viteIndexHtmlMiddleware (/app/node_modules/vite/dist/node/chunks/dep-59dc6e00.js:56286:28)
```

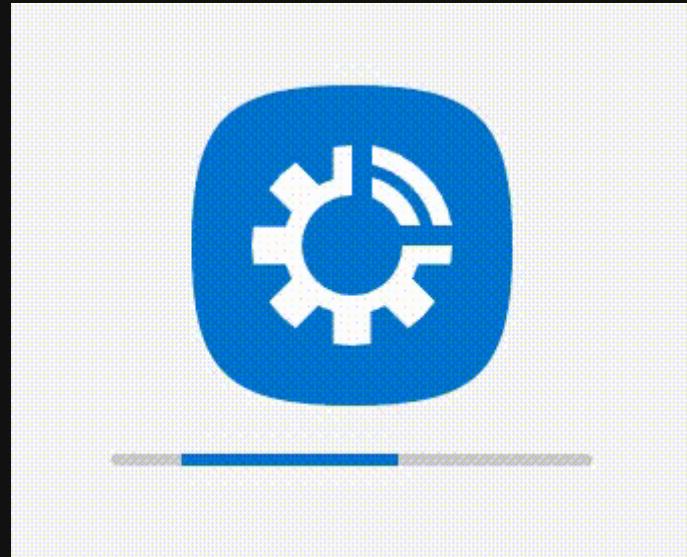
Click outside or fix the code to dismiss.

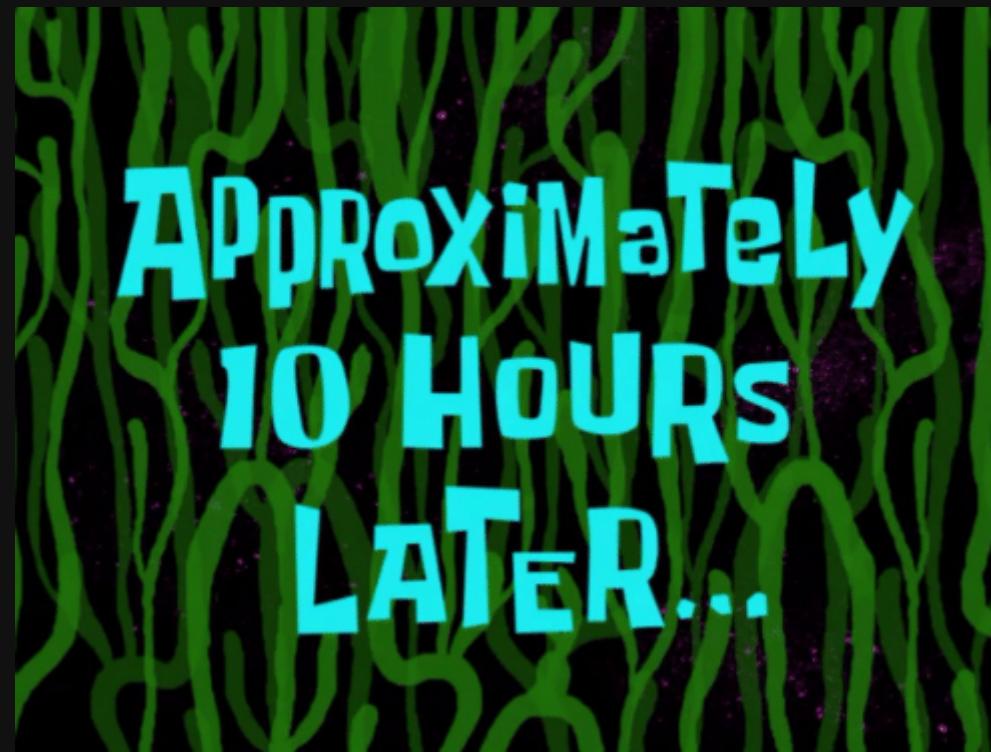
You can also disable this overlay by setting `server.hmr.overlay` to `false` in `vite.config.js`.

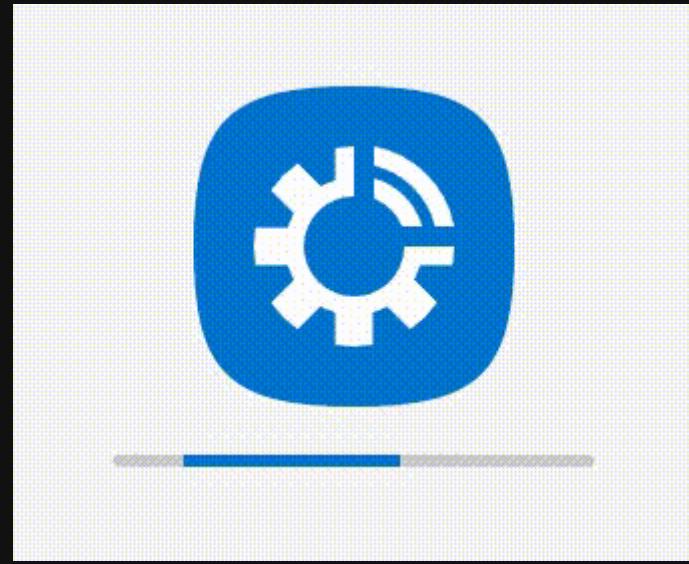
Comment out the errors



Let's check the browser...







Let's see what npm run build says...

```
vite v2.9.9 building for production...
✓ 2 modules transformed.
dist/index.html          9.38 KiB
dist/assets/index.f330a991.js  0.73 KiB / gzip: 0.41 KiB
```

- 1.570 seconds? 🔥
- 2 modules? 🤔

That's too fast...

- Even for Vite
- Vite wasn't seeing the whole picture
- `index.html` was using a `.js` loading script
- Swap the entry point with `src/main.ts`

C'mon Vite...

The following dependencies are imported but could not be resolved

```
element-ui/types/form (imported by ...)  
element-ui/types/message-box (imported by ...)  
element-ui/types/collapse (imported by ...)  
...  
element-ui/types/upload (imported by ...)  
element-ui/types/button (imported by ...)  
element-ui/types/form-item (imported by ...)
```

Are they installed?

What?

```
10 |     async changeLanguage(language) {  
11 |         const langCode = getLanguageConfiguration(language).ele  
12 |         await import(`element-ui/lib/locale/lang/${langCode}`).  
13 |             ^  
14 |             elementLocale.use(elementLang.default);  
15 |     }  
The above dynamic import cannot be analyzed by vite.
```

No no...

```
16 |     async changeLanguage(language) {  
17 |         const langCode = getLanguageConfiguration(language).mom  
18 |         await import(`moment/locale/${langCode}`).then(() => {  
19 |             ^  
20 |             moment.locale(langCode);  
});  
The above dynamic import cannot be analyzed by vite.
```

Seriously?

Undefined variable.

```
225 |     --ids-light-background: #{ids-light-background};  
|                                         ^^^^^^
```

Oh boy



Oh boy



Here we go...

So...

- Too many bugs 
- Just comment the errors
- Fix them later YOLO

Build it first

- Serve/fix loops would take forever
- A prod build should expose a decent amount of errors in one go
- Can also benchmark Vite builds against webpack

Language files

- Add .js extension

SASSy errors

- Comment all variable errors
- Will deal with this later
- Again... YOLO 😜

Element UI

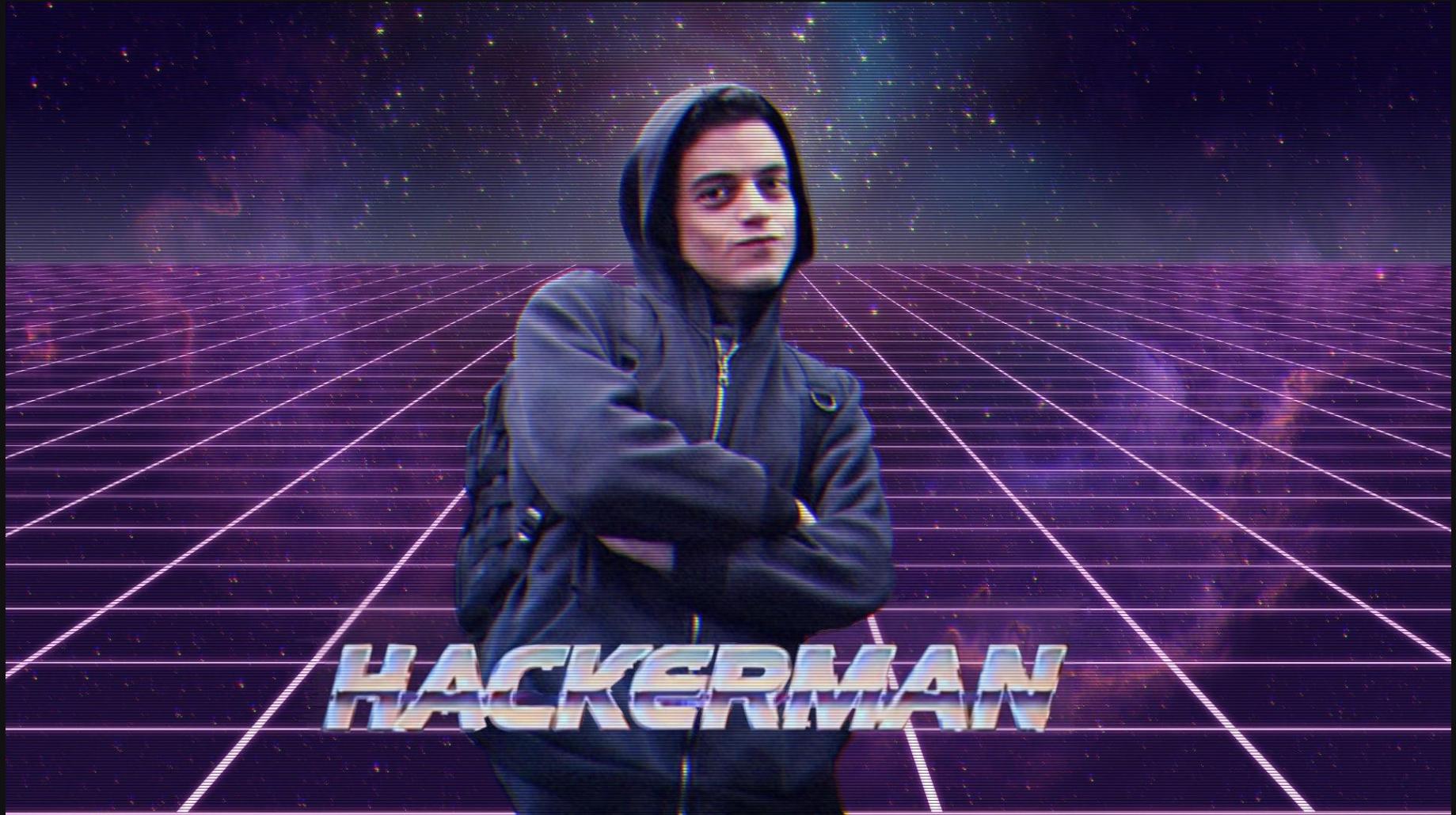
- CommonJS
- Hard dependency
- Only Element Plus supports Vue3

TypeScript type imports

import type

- Fix all ElementUI type imports

```
PATTERN="(import)( \{.*\} from 'element-ui/types/.*'$)"
for f in $(rg -l "$PATTERN")
do rg "$PATTERN" --passthru -r '$1 type$2' $f | sponge $f
done
```



That's it

- Everything seems ok
- Let's run a build

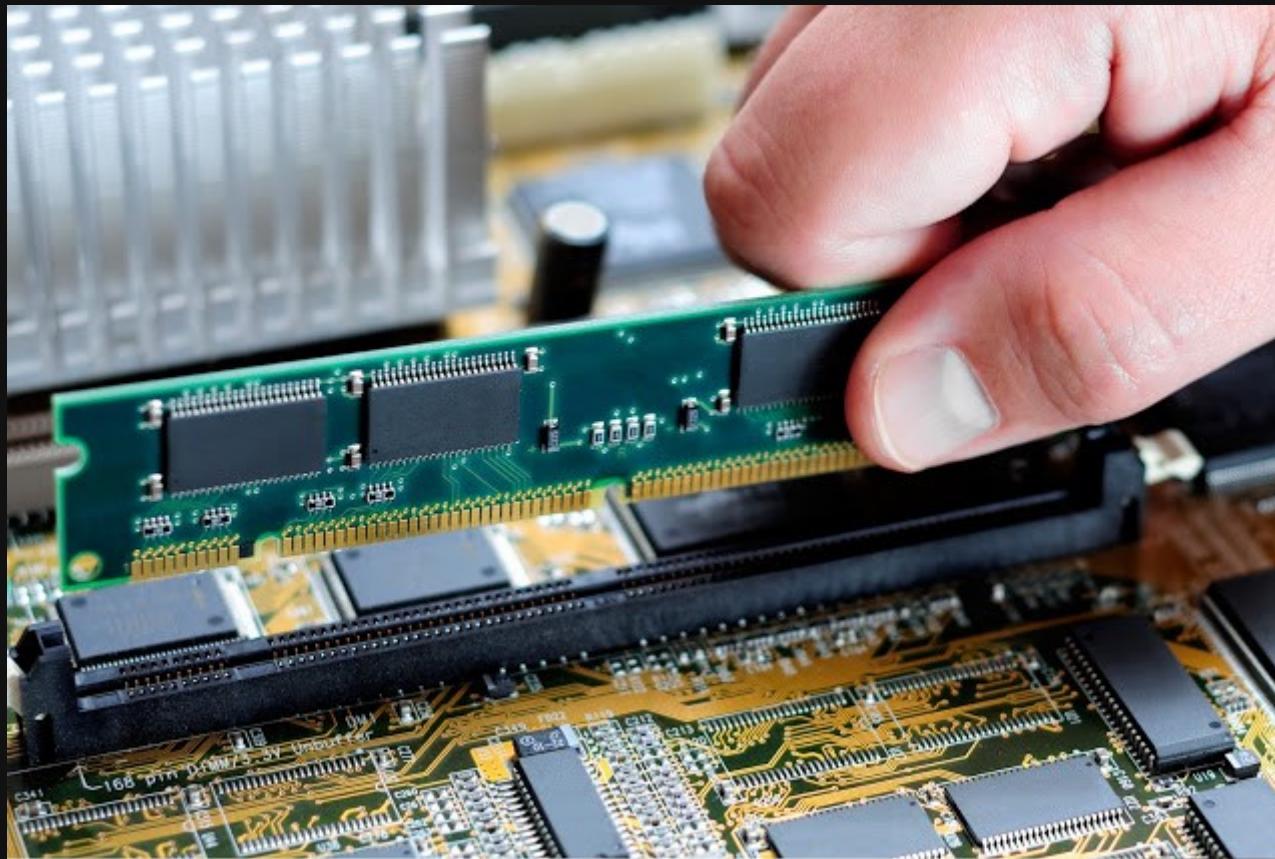
Almost!



===== JS stack trace =====

FATAL ERROR: JavaScript heap out of memory

Add more RAM

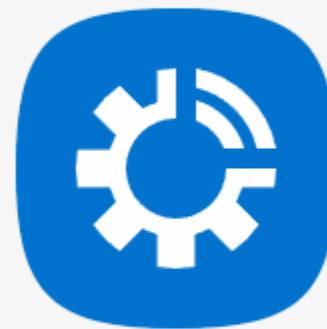


Building

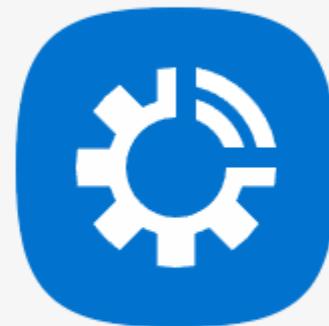
✓ 11492 modules transformed.

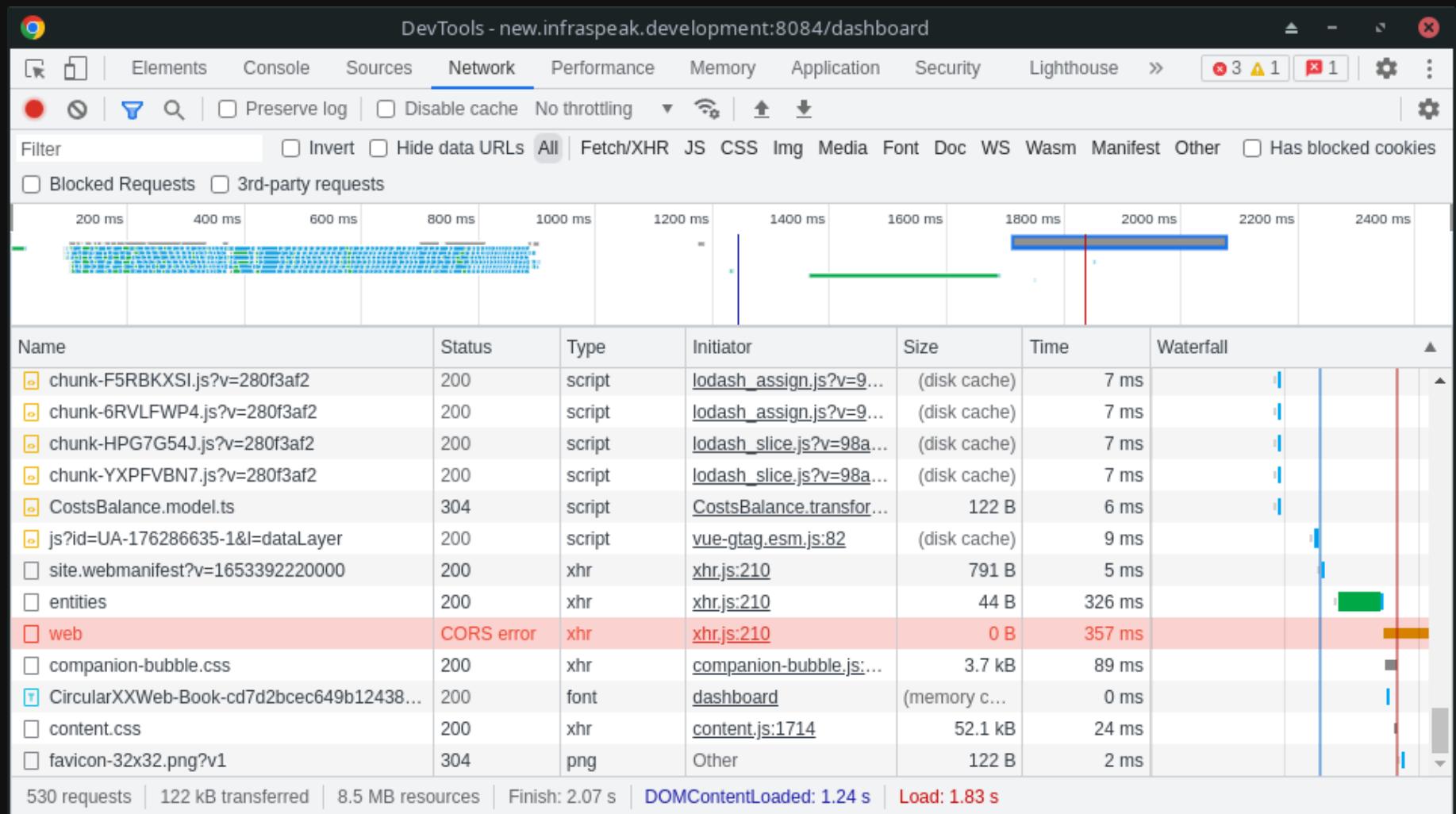
- 2:58 minutes? 🔥🔥🔥
- webpack takes 5:36
 - That's almost twice as fast: happier pipelines! 🚮
 - E.g.: First 6 months of 2020 totaled 255 pipeline hours that could be cut in half

Check the browser...



**TWO
HOURS
LATER....**





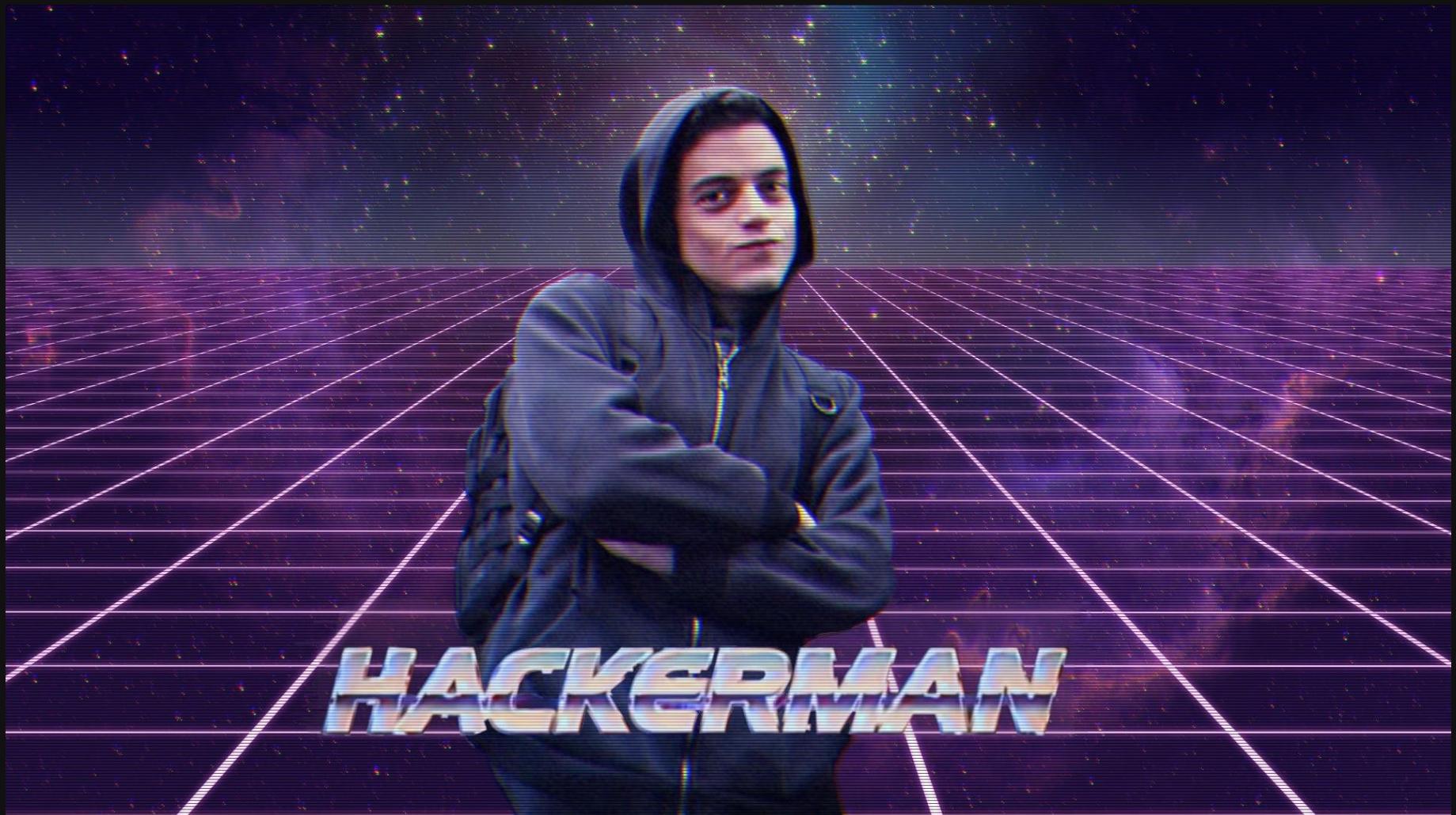
CORS error

Quick explanation

- CORS is a mechanism that allows restricted resources to be requested from a domain other than the one where the first resource was served
- The browser was attempting an XHR request to `api.infraspeak.development`
- But using a port the api was not expecting
- Thus the request is denied 💩

Know what?

Know what?



Fix the API

- This is a Vite demo
- I need this thing for the 25th
- Screw CORS

Fix the API

- This is a Vite demo
- I need this thing for the 25th
- Screw CORS
- YOLO



THIS IS A [REDACTED] SYSTEM

I KNOW THIS

imgflip.com

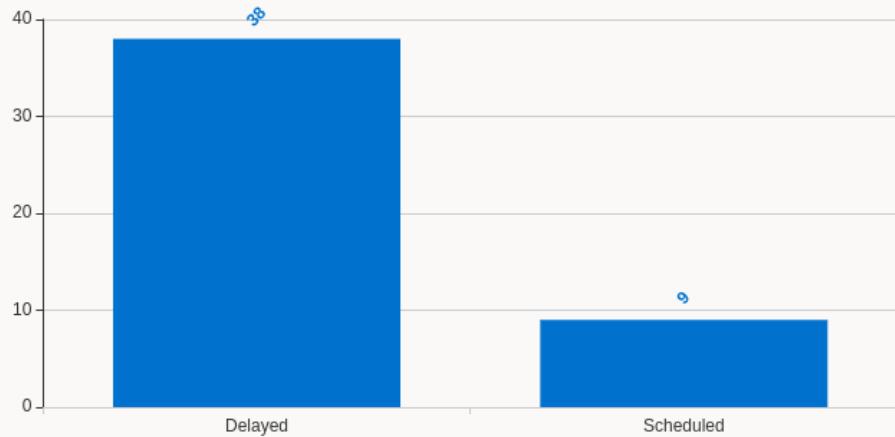
```
'allowed_origins' => ['*'],
```

Check the browser?

Check the browser?

[Hub](#)[+ Add widget](#) [Missing something? Make a Suggestion!](#)[Dashboard](#)[Assets](#)[Planned Jobs](#)[Work Orders](#)[Calendar](#)[Planner](#)[Requests](#)[Stock](#)[Sales](#)[Purchases](#)[Analytics](#)[Documents](#)[Utilities](#)[Housekeeper](#)[Maps](#)[Nº Total de Avarias \(This month\)](#)

0

[Nº de Trabalhos por Estado \(This month\)](#)[Nº Total de Trabalhos \(This m...\)](#)



I am invincible!

Let's click stuff

Let's click stuff

- Nothing happens...

Check DevTools

- ➊ ► **ReferenceError: require is not defined
at IsListCard.vue:50:1**



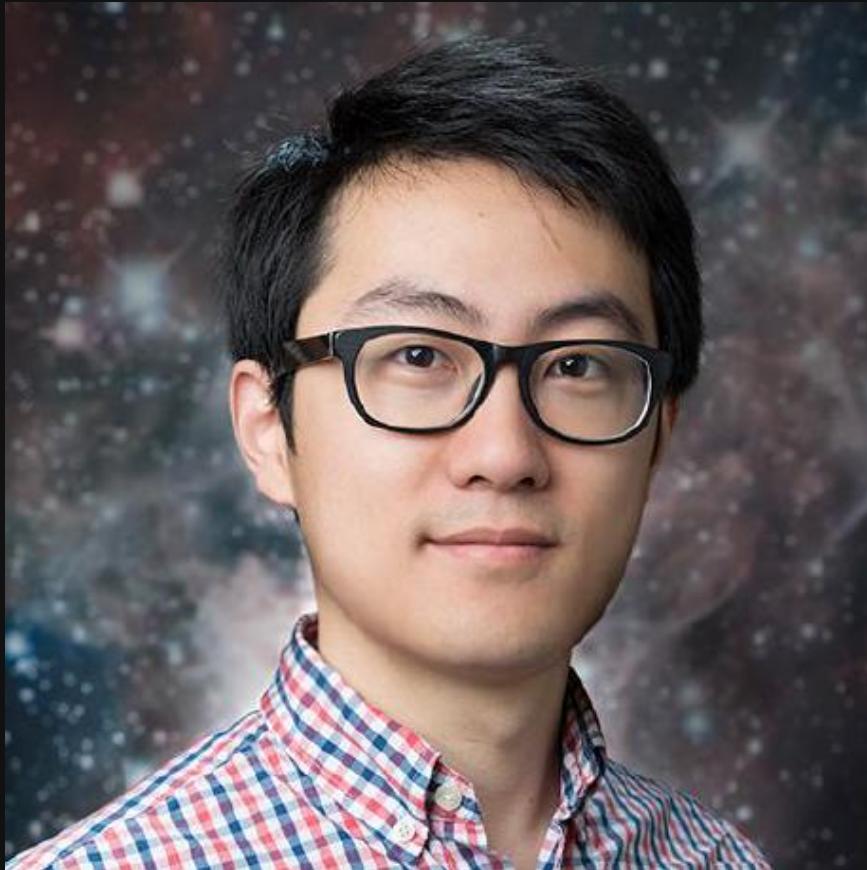
Ask the Internet why

 yyx990803 commented on 18 Jan 2021 • edited

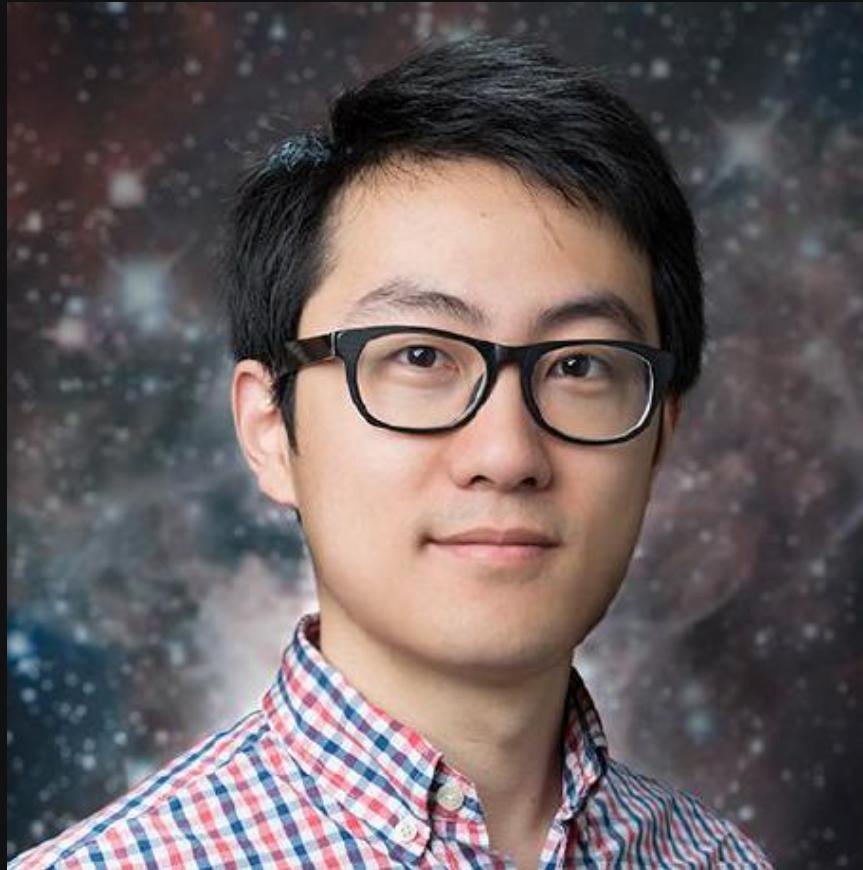
There should never be `require` in source code when using Vite. It's ESM only.

Uh... says who? 😞

Oh...



Oh...



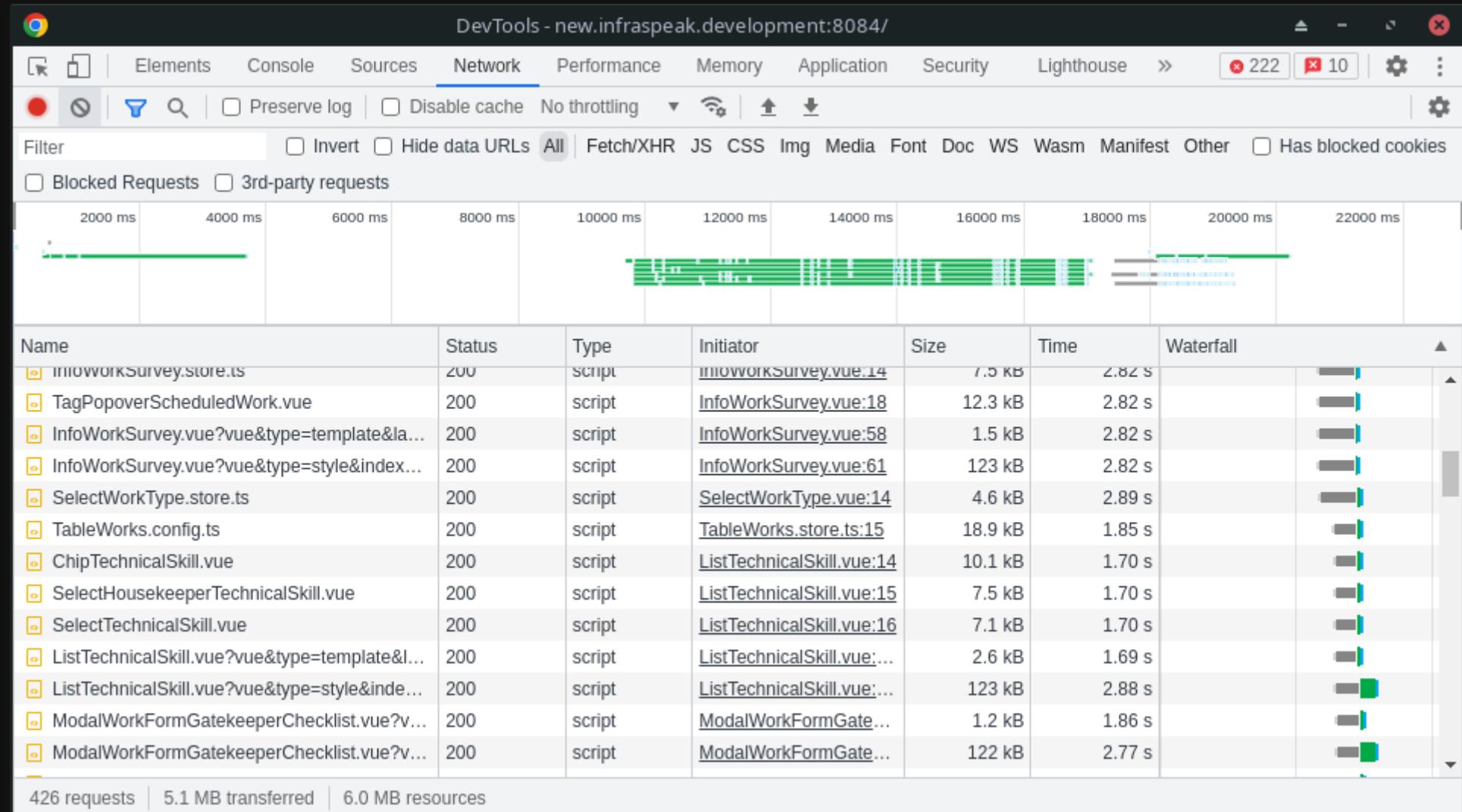
Ok, fine! 😠

Vite CommonJS Plugin

```
1 diff --git a/vite.config.js b/vite.config.js
2 index 7396c7914..0cf7e6944 100644
3 --- a/vite.config.js
4 +++ b/vite.config.js
5 @@ -1,11 +1,15 @@
6   import { defineConfig } from 'vite'
7   import { createVuePlugin as vue } from 'vite-plugin-vue2'
8 +import { viteCommonjs } from '@originjs/vite-plugin-commonjs'
9
10  const path = require('path')
11
12 // https://vitejs.dev/config/
13 export default defineConfig({
14 -   plugins: [vue()],
15 +   plugins: [
```

Vite CommonJS Plugin

```
5 @@ -1,11 +1,15 @@
6   import { defineConfig } from 'vite'
7   import { createVuePlugin as vue } from 'vite-plugin-vue2'
8 +import { viteCommonjs } from '@originjs/vite-plugin-commonjs'
9
10  const path = require('path')
11
12 // https://vitejs.dev/config/
13 export default defineConfig({
14 -   plugins: [vue()],
15 +   plugins: [
16 +     vue(),
17 +     viteCommonjs()
18 +   ],
19   resolve: {
```



- We're good now!

Poking around

- Doesn't feel *that* fast
 - That's because Vite builds on demand
 - Will require further tweaking
- Only slow when first visiting a view

Did DX improve?

- Let's see how fast HMR is

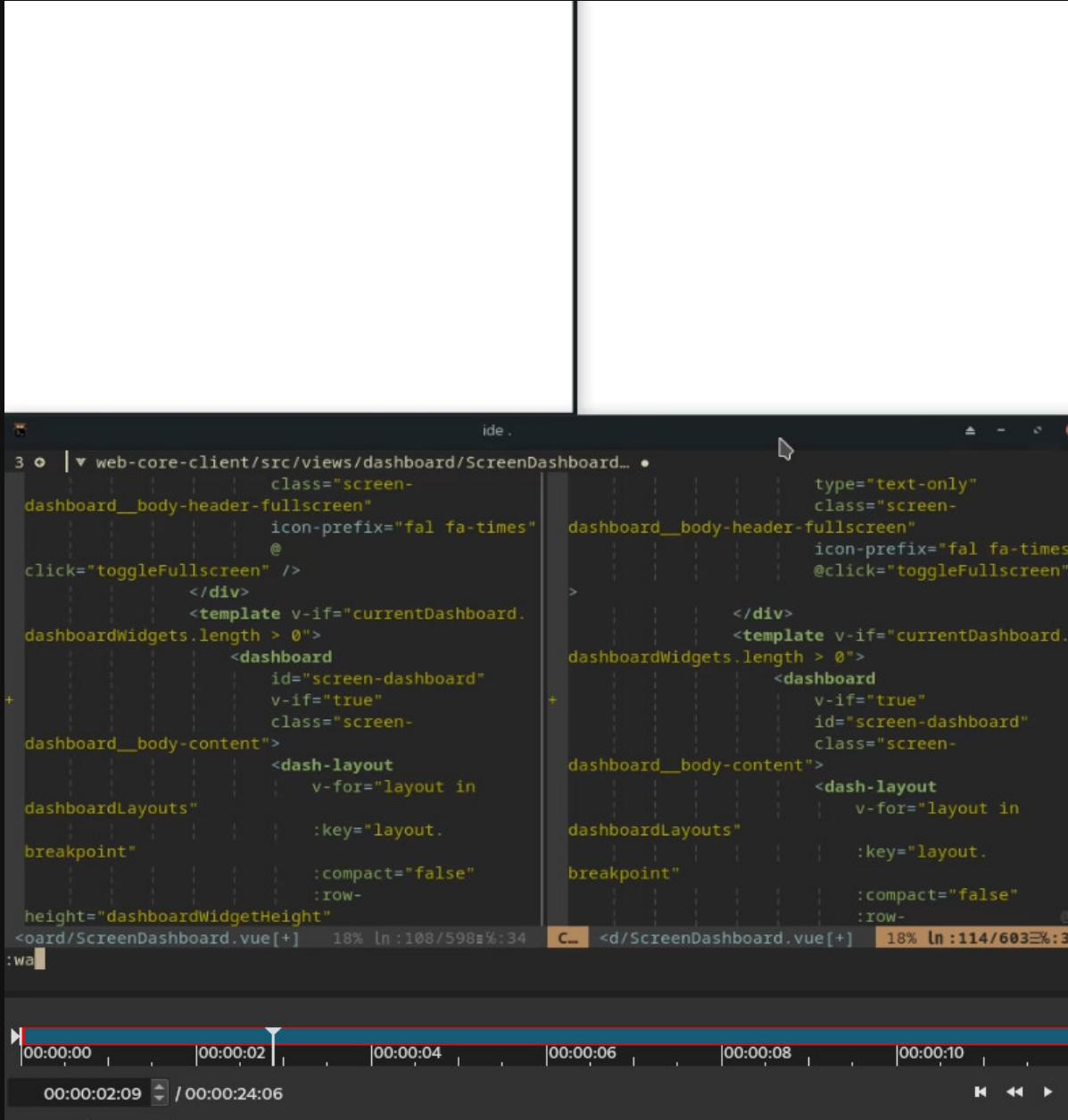
Simple benchmark

- We'll edit the same file over the same codebase
 - The original using webpack
 - The new and improved one using Vite
- Save both files at the same time
- Wait for HMR to kick in

Simple benchmark



CAN YOU
ENHANCE THAT



The screenshot shows a code editor with two panes displaying Vue.js component code. The code is identical in both panes, showing a template with a header and a content area containing a dashboard with layouts.

```
3 o | web-core-client/src/views/dashboard/ScreenDashboard.vue •
      class="screen-
      dashboard__body-header-fullscreen"
      icon-prefix="fal fa-times"
      @
      click="toggleFullscreen" />
      </div>
      <template v-if="currentDashboard.
      dashboardWidgets.length > 0">
      <dashboard
      id="screen-dashboard"
      v-if="true"
      class="screen-
      dashboard__body-content">
      <dash-layout
      v-for="layout in
      dashboardLayouts"
      :key="layout.
      breakpoint"
      :compact="false"
      :row-
      height="dashboardWidgetHeight"
      <div></div>
```

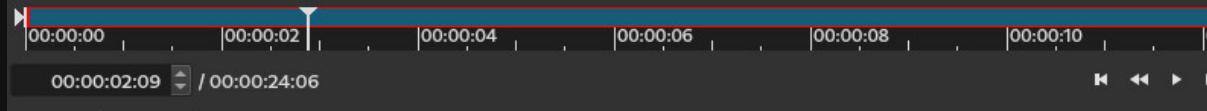
The code editor has tabs for 'ScreenDashboard.vue' and a file browser tab. The status bar at the bottom shows a timestamp of 00:00:24:06.



webpack

```
ide .  
3 o | v web-core-client/src/views/dashboard/ScreenDashboard... •  
      class="screen-  
dashboard__body-header-fullscreen"  
          icon-prefix="fal fa-times"  
          @  
click="toggleFullscreen" />  
      </div>  
      <template v-if="currentDashboard.  
dashboardWidgets.length > 0">  
      <dashboard  
          id="screen-dashboard"  
          v-if="true"  
          class="screen-  
dashboard__body-content">  
          <dash-layout  
              v-for="layout in  
dashboardLayouts"  
              :key="layout.  
breakpoint"  
              :compact="false"  
              :row-  
height="dashboardWidgetHeight"  
<oard/ScreenDashboard.vue[+] 18% ln :108/598±:34  
:wa
```

```
      type="text-only"  
      class="screen-  
dashboard__body-header-fullscreen"  
          icon-prefix="fal fa-times"  
          @click="toggleFullscreen" />  
      </div>  
      <template v-if="currentDashboard.  
dashboardWidgets.length > 0">  
      <dashboard  
          v-if="true"  
          id="screen-dashboard"  
          class="screen-  
dashboard__body-content">  
          <dash-layout  
              v-for="layout in  
dashboardLayouts"  
              :key="layout.  
breakpoint"  
              :compact="false"  
              :row-
```



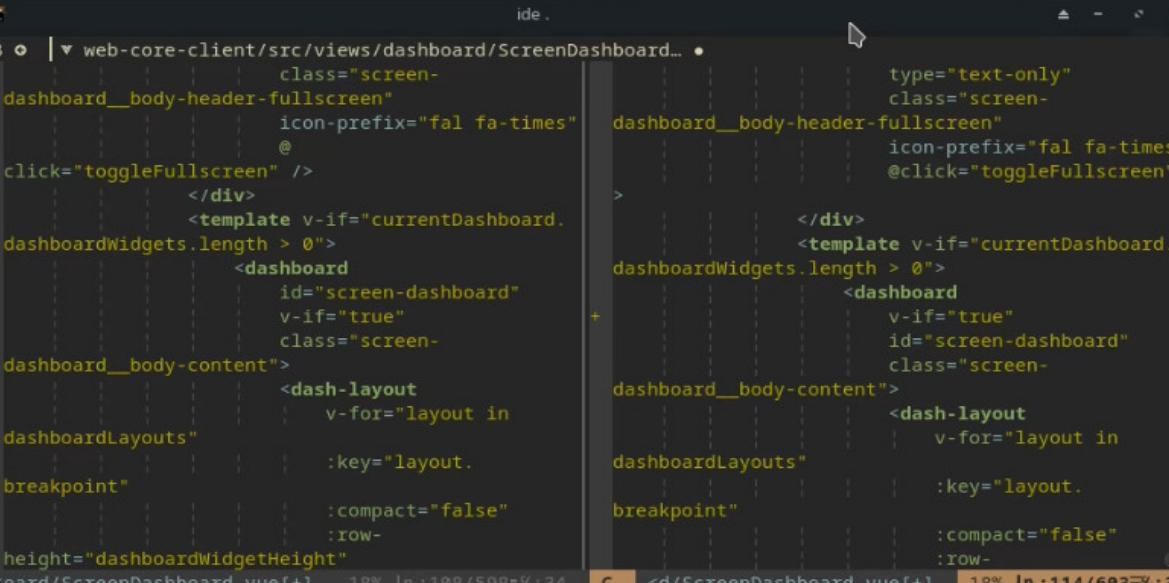


webpack



ide .

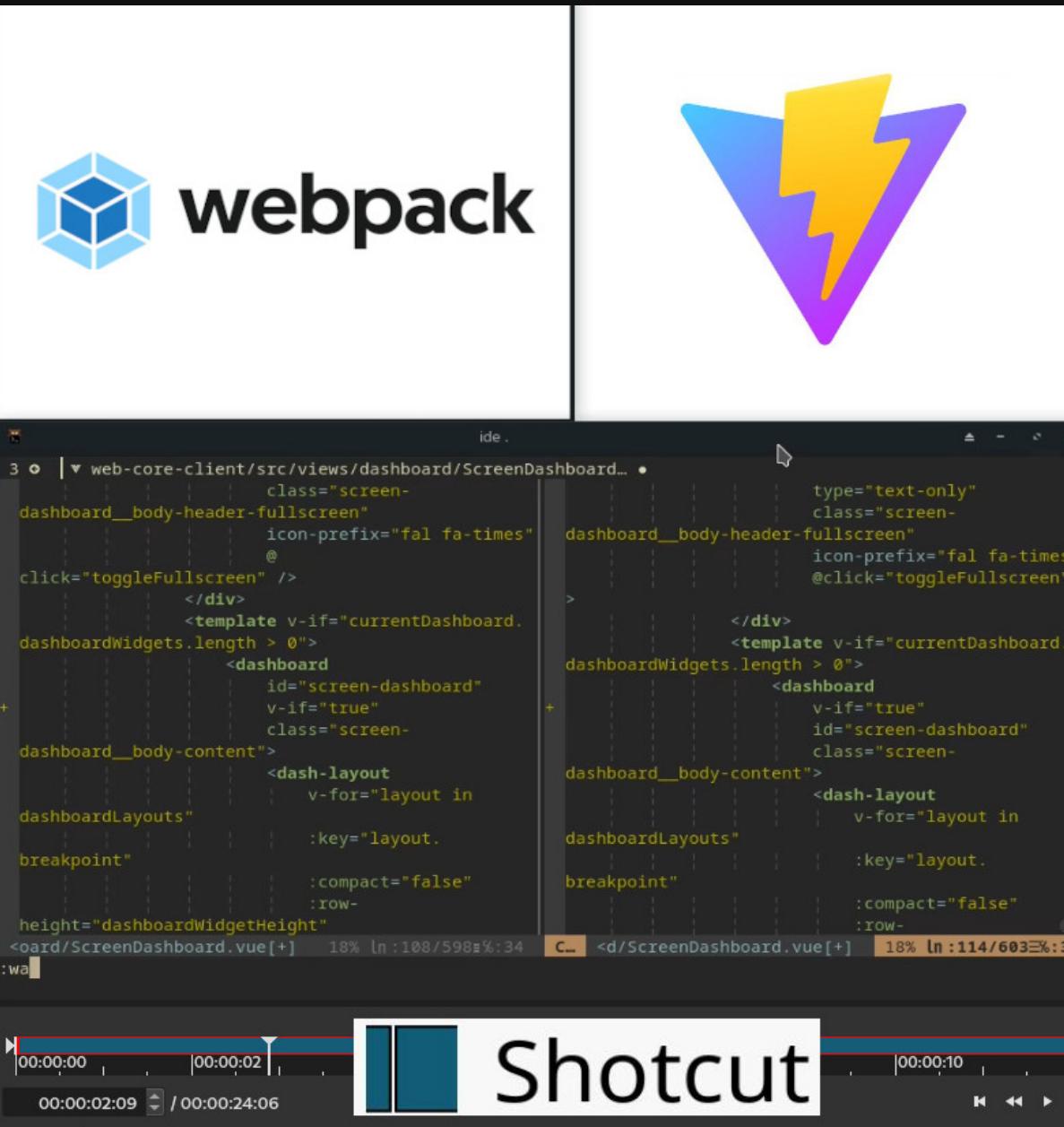
```
3 o | v web-core-client/src/views/dashboard/ScreenDashboard... •
      class="screen-
      dashboard__body-header-fullscreen"
          icon-prefix="fal fa-times"
          @
      click="toggleFullscreen" />
          </div>
          <template v-if="currentDashboard.
      dashboardWidgets.length > 0">
          <dashboard
              id="screen-dashboard"
              v-if="true"
              class="screen-
      dashboard__body-content">
              <dash-layout
                  v-for="layout in
      dashboardLayouts"
                  :key="layout.
      breakpoint"
                  :compact="false"
                  :row-
                  height="dashboardWidgetHeight"
<oard/ScreenDashboard.vue[+] 18% ln :108/598±:34
:wa
```

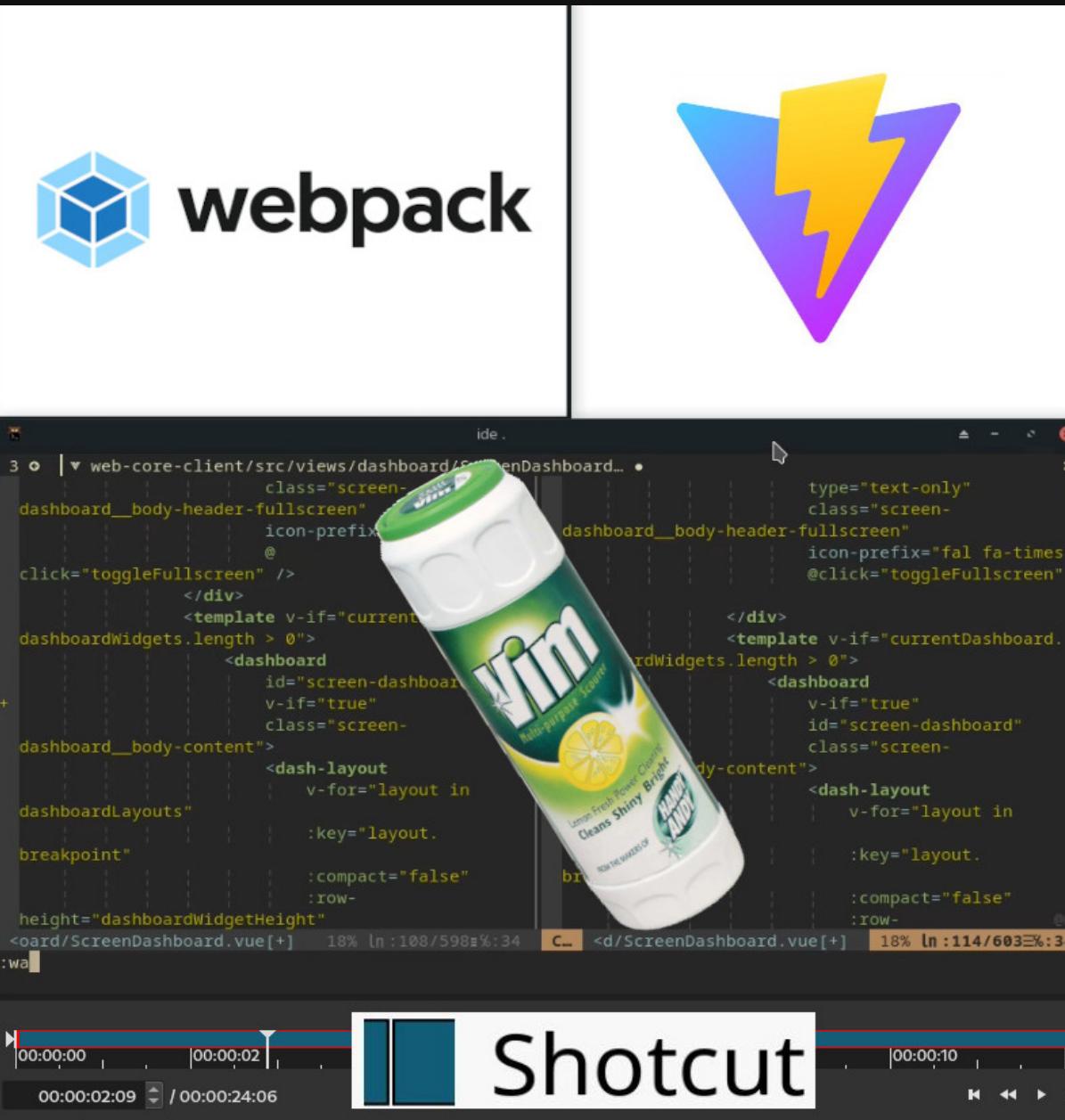


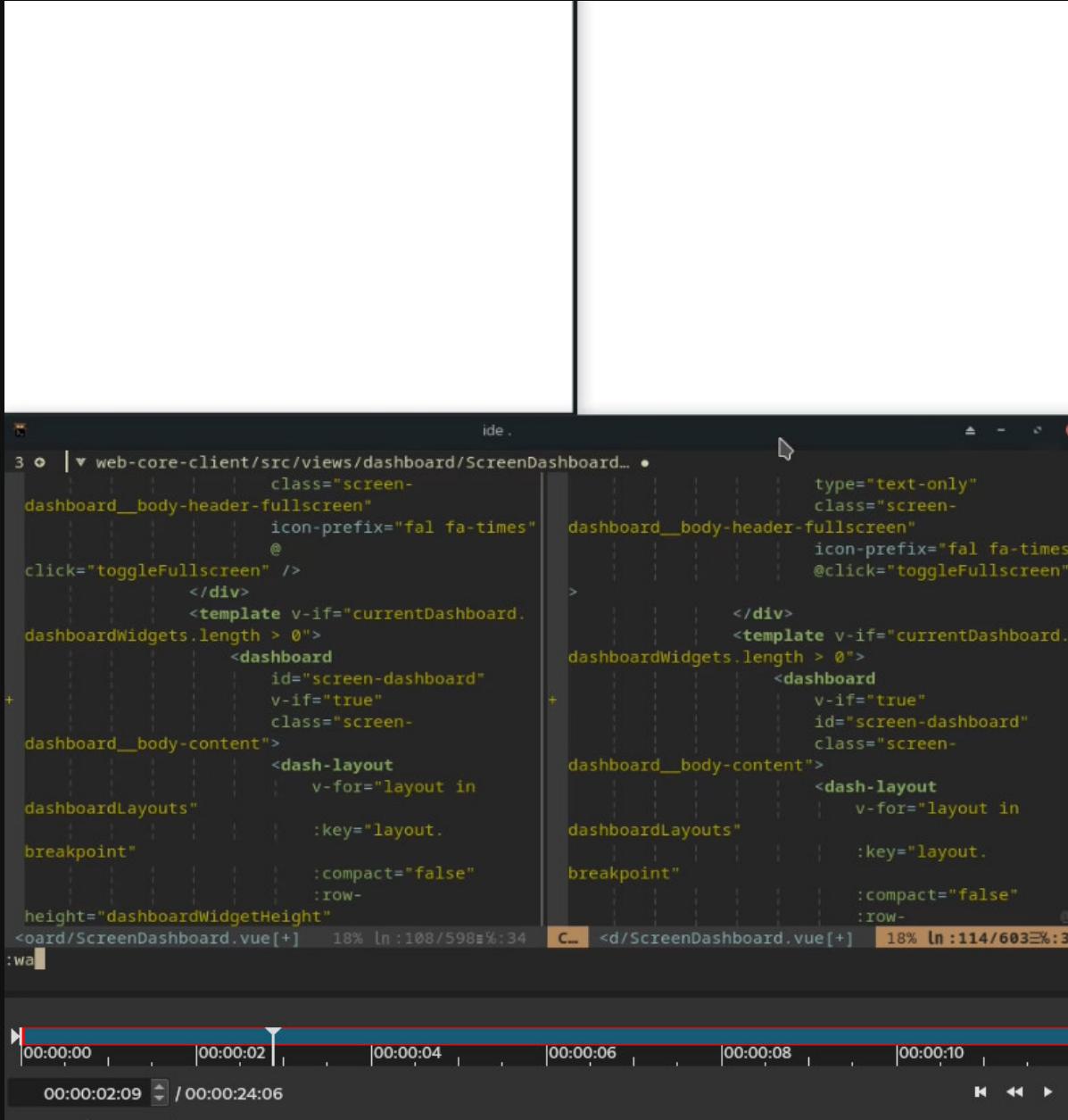
C... <d/ScreenDashboard.vue[+] 18% ln :114/603±:34

00:00:00 | 00:00:02 | 00:00:04 | 00:00:06 | 00:00:08 | 00:00:10

00:00:02:09 ▾ / 00:00:24:06







```
ide .  
3 o | v web-core-client/src/views/dashboard/ScreenDashboard... •  
      class="screen-  
dashboard__body-header-fullscreen"  
      icon-prefix="fal fa-times"  
      @  
click="toggleFullscreen" />  
      </div>  
      <template v-if="currentDashboard.  
dashboardWidgets.length > 0">  
      <dashboard  
          id="screen-dashboard"  
          v-if="true"  
          class="screen-  
dashboard__body-content">  
          +  
          <dash-layout  
              v-for="layout in  
dashboardLayouts"  
              :key="layout.  
breakpoint"  
              :compact="false"  
              :row-  
height="dashboardWidgetHeight"  
<oard/ScreenDashboard.vue[+] 18% ln :108/598±:34  
:wa
```

```
      type="text-only"  
      class="screen-  
dashboard__body-header-fullscreen"  
      icon-prefix="fal fa-times"  
      @click="toggleFullscreen"  
      >  
      </div>  
      <template v-if="currentDashboard.  
dashboardWidgets.length > 0">  
      <dashboard  
          v-if="true"  
          id="screen-dashboard"  
          class="screen-  
dashboard__body-content">  
          <dash-layout  
              v-for="layout in  
dashboardLayouts"  
              :key="layout.  
breakpoint"  
              :compact="false"  
              :row-
```

Nº Total de Trabalhos (redesign_sid...)

```
ide .
```

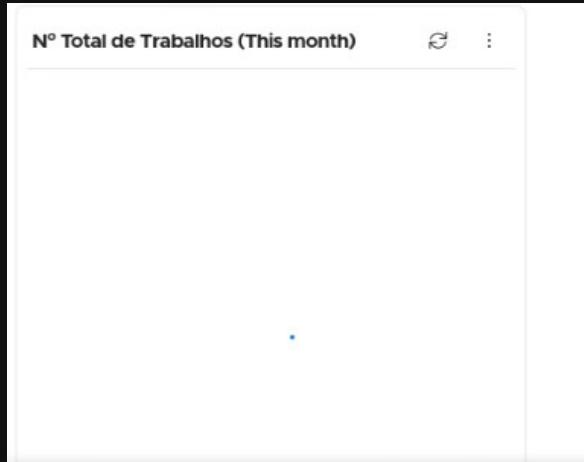
3 o | v web-core-client/src/views/dashboard/ScreenDashboard... x

```
      class="screen-
dashboard__body-header-fullscreen"
          icon-prefix="fal fa-times"
          @
click="toggleFullscreen" />
      </div>
      <template v-if="currentDashboard.
dashboardWidgets.length > 0">
      <dashboard
          id="screen-dashboard"
          v-if="true"
          class="screen-
dashboard__body-content">
          <dash-layout
              v-for="layout in
dashboardLayouts"
              :key="layout.
breakpoint"
              :compact="false"
              :row-
height="dashboardWidgetHeight"
<shboard/ScreenDashboard.vue 18% ln :108/598�:34
"~/Work/Infraspeak/repository/web/web-core-client/src/views/dashboard/ScreenDashboard.vue" 603L, 23440B witten
```

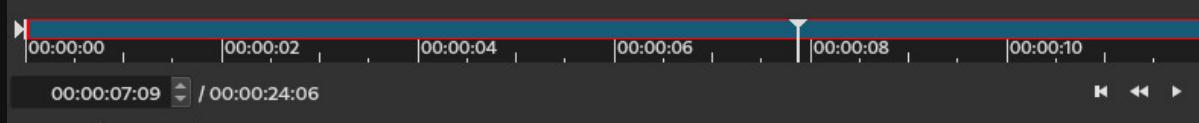
N... <hboard/ScreenDashboard.vue 18% ln :114/603�:34

00:00:00 | 00:00:02 | 00:00:04 | 00:00:06 | 00:00:08 | 00:00:10

00:00:03:00 ▲ / 00:00:24:06



```
ide .  
3 o | ▶ web-core-client/src/views/dashboard/ScreenDashboard... x  
      class="screen-  
      dashboard__body-header-fullscreen"  
      icon-prefix="fal fa-times"  
      @  
      click="toggleFullscreen" />  
      </div>  
      <template v-if="currentDashboard.  
      dashboardWidgets.length > 0">  
      <dashboard  
          id="screen-dashboard"  
          v-if="true"  
          class="screen-  
          dashboard__body-content">  
          <dash-layout  
              v-for="layout in  
              dashboardLayouts"  
              :key="layout.  
              breakpoint"  
              :compact="false"  
              :row-  
              height="dashboardWidgetHeight"  
<shboard/ScreenDashboard.vue 18% ln:108/598 El:34  
"~/Work/Infraspeak/repository/web/web-core-client/src/views/dashboard/ScreenDashboard.vue" 603L, 23440B written
```



Now THAT's fast

- A single frame after saving the files vite reloads the page (10fps)
- webpack takes almost 5 seconds to present the same change
- That's 50x faster. Effectively instantaneous! 

**Know what else we
could speed up?**

Storybook



Storybook with webpack

```
Storybook 6.4.21 for Vue started
3.27 min for preview

Local:          http://localhost:6006/
On your network: http://192.168.1.82:6006/

A new version (6.5.3) is available!
Upgrade now: npx sb@latest upgrade
Read full changelog: https://git.io/fhFYe
```

Storybook with Vite

```
Storybook 6.4.21 for Vue started
1.61 s for preview

Local:          http://localhost:6006/
On your network: http://192.168.1.82:6006/

A new version (6.5.3) is available!
Upgrade now: npx sb@latest upgrade

Read full changelog: https://git.io/fhFYe
```

Storybook with Vite

```
Storybook 6.4.21 for Vue started
1.61 s for preview

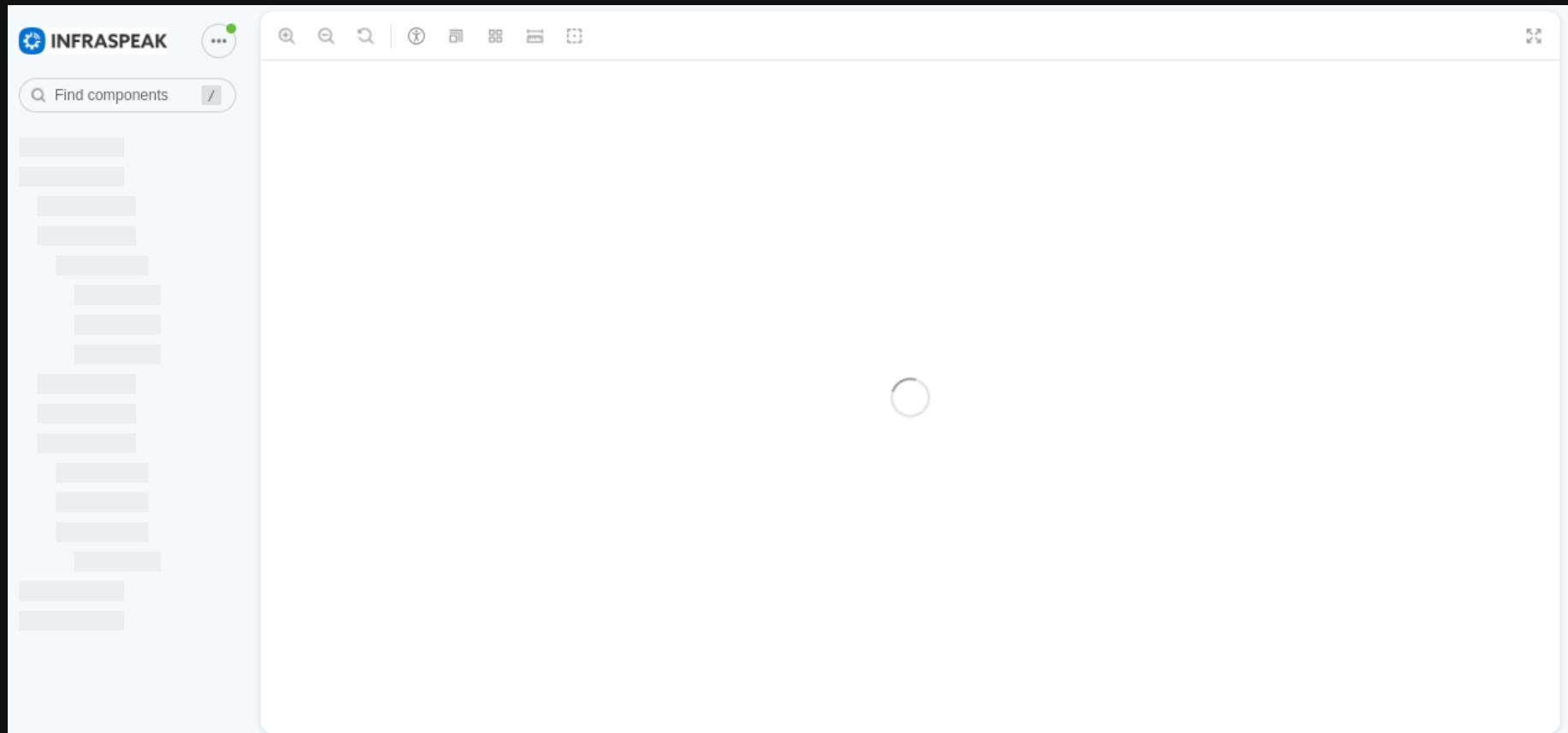
Local:          http://localhost:6006/
On your network: http://192.168.1.82:6006/

A new version (6.5.3) is available!
Upgrade now: npx sb@latest upgrade

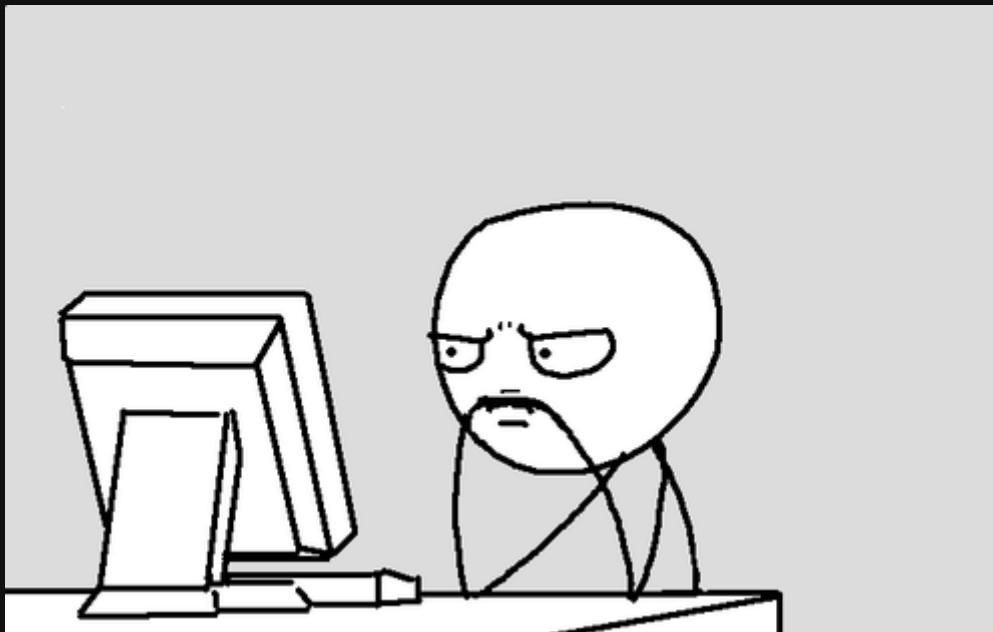
Read full changelog: https://git.io/fhFYe
```

- 1.61 seconds vs 3.27 minutes 🔥

Let's check our stories

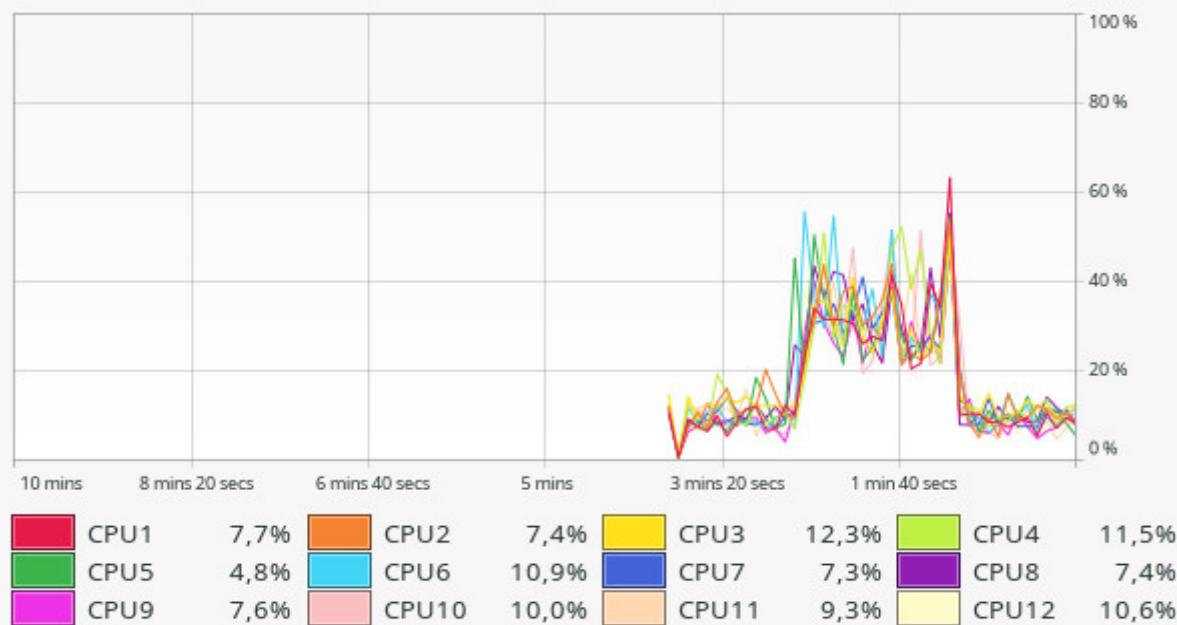


Something's not ri...



CTRL+C

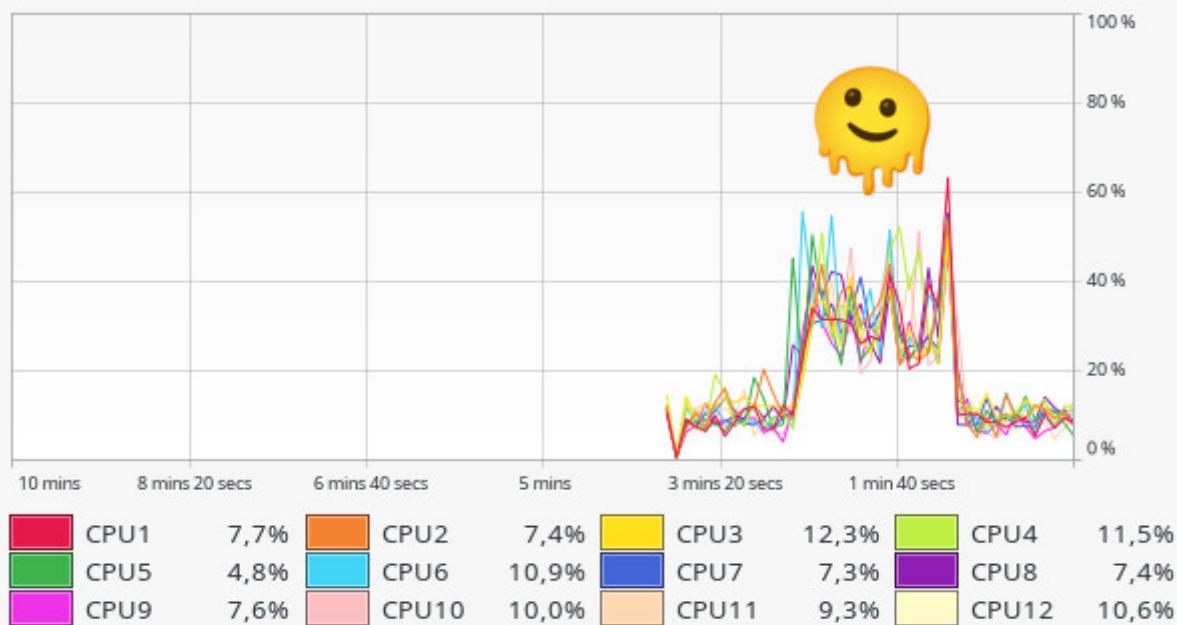
▼ CPU



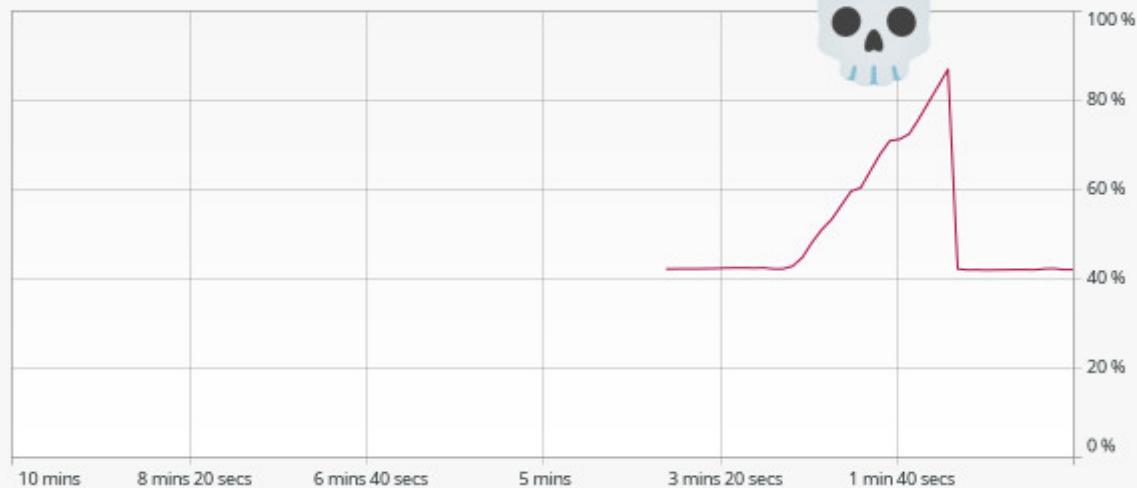
▼ Memory and Swap



▼ CPU



▼ Memory and Swap

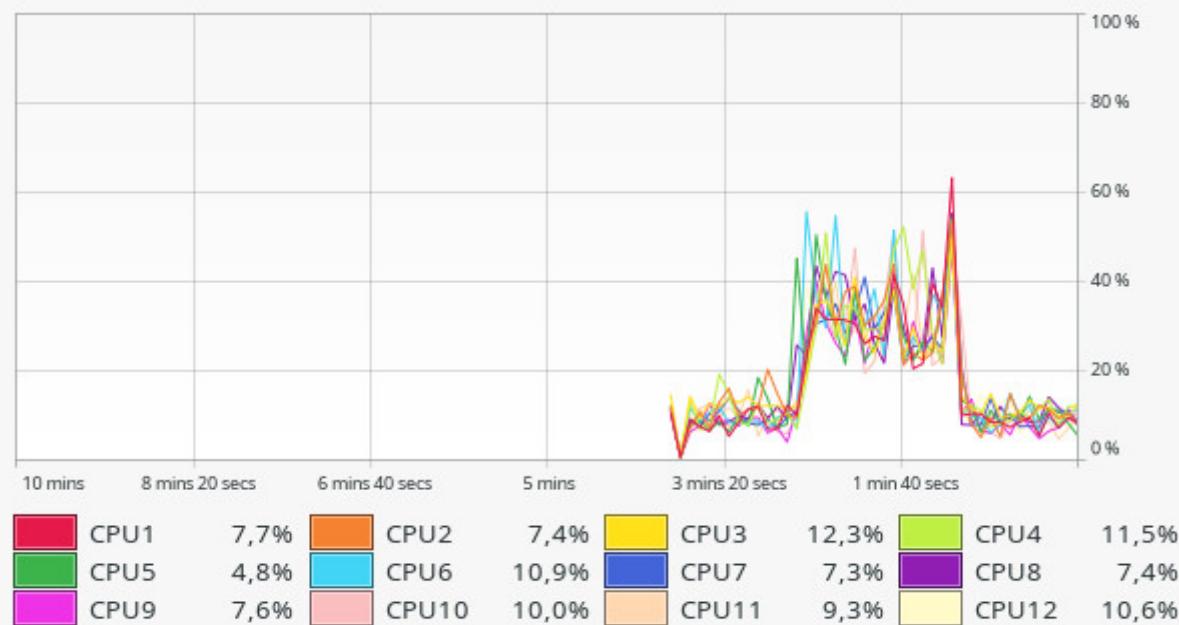


Memory
13,9 GB (41,4%) of 33,6 GB
Cache 4,5 GB



Swap
not available

▼ CPU



▼ Memory and Swap



Memory
13,9 GB (41,4%) of 33,6 GB
Cache 4,5 GB



Swap
not available

Maybe next time 😐

- Storybook will consume all the RAM in the universe
- Let's speed something else up shall we?



Up next



Up next

João Gonçalves



Up next

João Gonçalves

Someone who knows what he's doing!

Hope you enjoyed this

- Learning from someone else's mistakes is good
- But learning from your own is even better