TAURI

We’re all a little rusty…

Cofounder of ***Tauri***
CEO of ***CrabNebula***
Open Source ***Realist***

Daniel Thompson

Opinions (and bad jokes) are my own.

TAURI

What do you call someone that speaks 3 languages?

TAURI

What do you call someone that speaks 3 languages?

Polyglot

TAURI

What do you call someone that speaks 2 languages?

TAURI

What do you call someone that speaks 2 languages?

Not American

TAURI

What do you call someone that speaks 1 language?

TAURI

What do you call someone that speaks 1 language?

Fullstack Dev

TAURI

Don’t be offended, or take offense on other’s account, because even polyglots had, at one time, the need to start with one language.

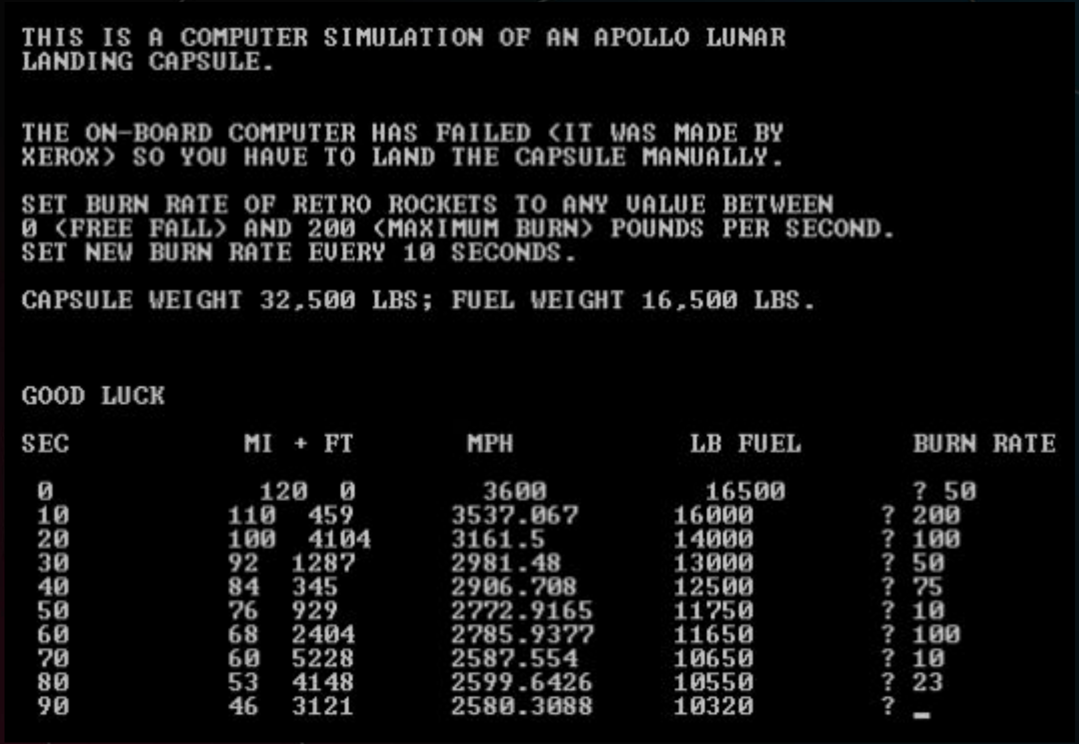
Life is a journey through time, so come with me.

Modern education succeeds in cementing the status quo and in encouraging __________, with some surprising results.

Let’s go for a romp.

TAURI

xx60’s Problem



```
THIS IS A COMPUTER SIMULATION OF AN APOLLO LUNAR
LANDING CAPSULE.


THE ON-BOARD COMPUTER HAS FAILED (IT WAS MADE BY
XEROX) SO YOU HAVE TO LAND THE CAPSULE MANUALLY.

SET BURN RATE OF RETRO ROCKETS TO ANY VALUE BETWEEN
0 (FREE FALL) AND 200 (MAXIMUM BURN) POUNDS PER SECOND.
SET NEW BURN RATE EVERY 10 SECONDS.

CAPSULE WEIGHT 32,500 LBS; FUEL WEIGHT 16,500 LBS.



GOOD LUCK

SEC           MI + FT       MPH           LB FUEL       BURN RATE

 0             120  0        3600          16500        ? 50
 10          110  459      3537.067      16000        ? 200
 20          100  4104     3161.5        14000        ? 100
 30          92  1287      2981.48       13000        ? 50
 40          84  345       2906.708      12500        ? 75
 50          76  929       2772.9165     11750        ? 10
 60          68  2404      2785.9377     11650        ? 100
 70          60  5228      2587.554      10650        ? 10
 80          53  4148      2599.6426     10550        ? 23
 90          46  3121      2580.3088     10320        ? 
```

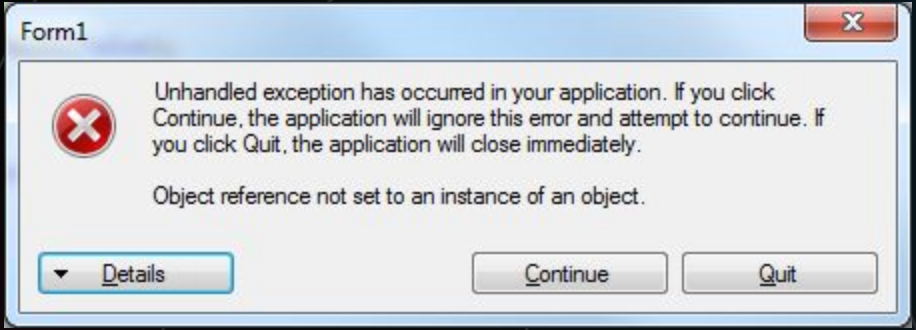
xx60’s Cheatcode

count to five

- Margaret Hamilton

TAURI

xx60’s Legacy

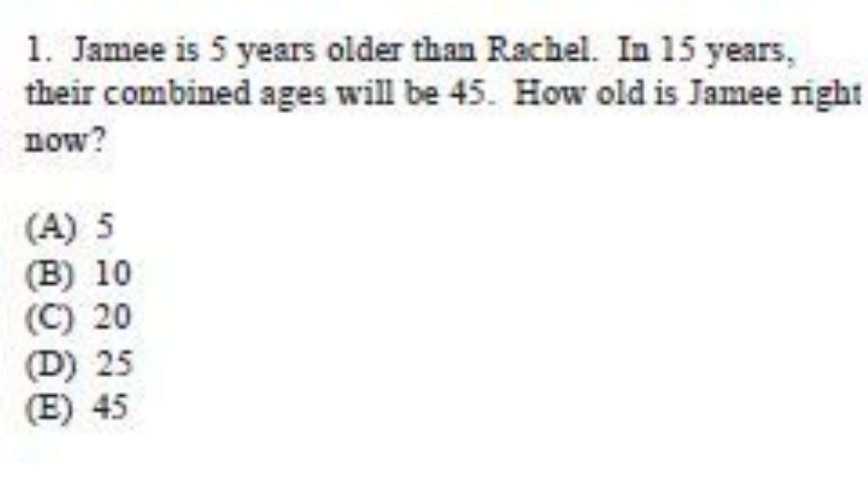


Unhandled exception has occurred in your application. If you click Continue, the application will ignore this error and attempt to continue. If you click Quit, the application will close immediately.

Object reference not set to an instance of an object.

TAURI

xx70's Problem

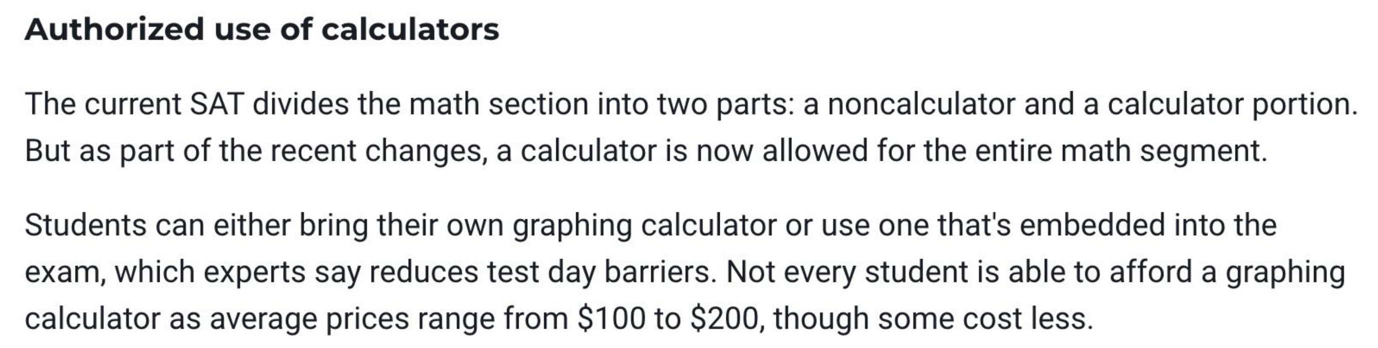


1. Jamee is 5 years older than Rachel. In 15 years, their combined ages will be 45. How old is Jamee right now?

(A) 5
(B) 10
(C) 20
(D) 25
(E) 45

xx70’s Cheatcode

xx70’s Legacy

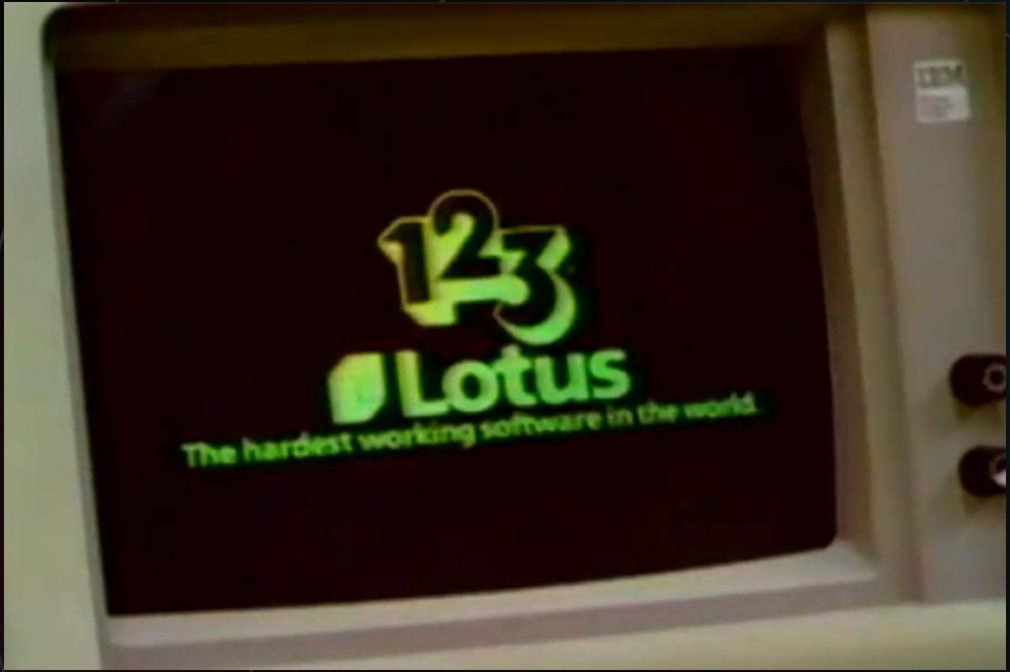


Authorized use of calculators

The current SAT divides the math section into two parts: a noncalculator and a calculator portion. But as part of the recent changes, a calculator is now allowed for the entire math segment.

Students can either bring their own graphing calculator or use one that's embedded into the exam, which experts say reduces test day barriers. Not every student is able to afford a graphing calculator as average prices range from $100 to $200, though some cost less.

xx80’s Problem



TAURI

xx80’s Cheatcode

FREE SOFTWARE
FOUNDATION

xx80’s Legacy

Microsoft

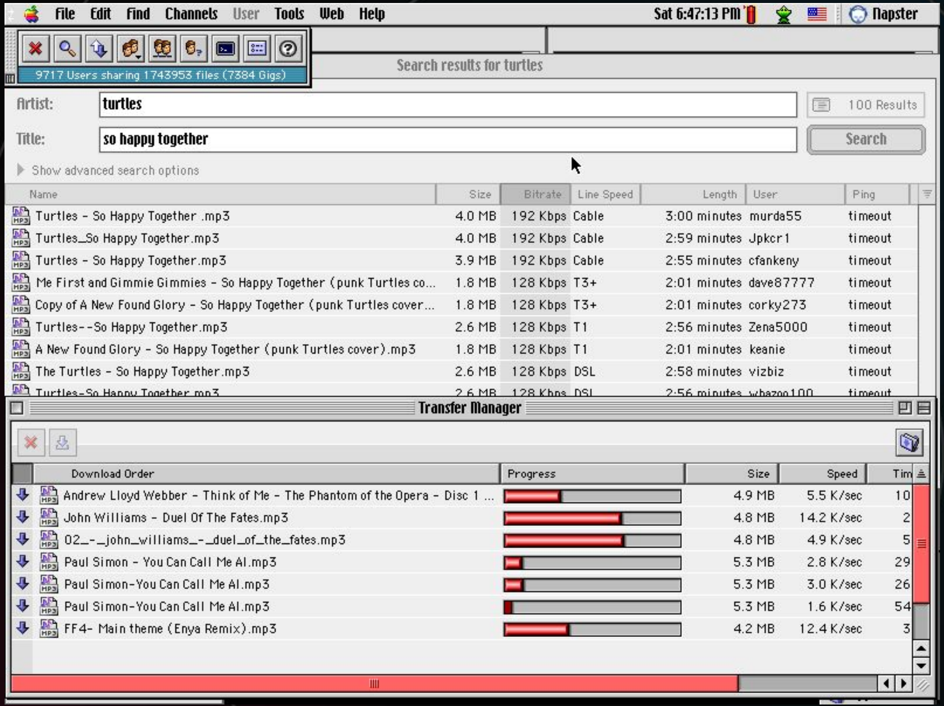
GitHub

npm

xx90's Problem

Chronicle / Chris Hardy, File

xx90's Cheatcode



9717 Users sharing 1743953 files (7384 Gigs)

Search results for turtles

Artist: turtles

Title: so happy together

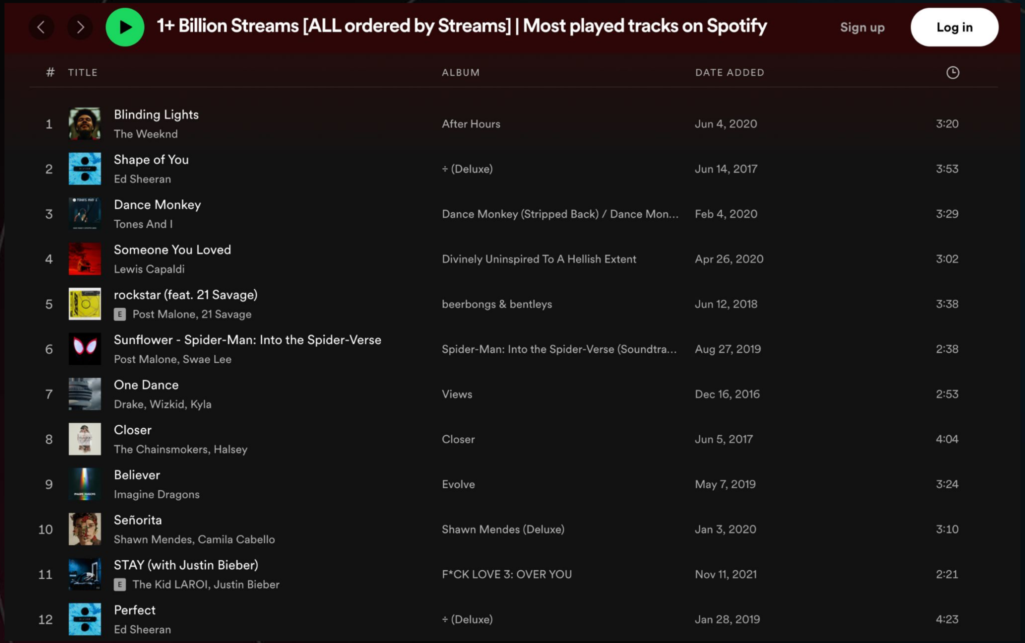
100 Results

Name	Size	Bitrate	Line Speed	Length	User	Ping
Turtles - So Happy Together .mp3	4.0 MB	192 Kbps	Cable	3:00 minutes	murda55	timeout
Turtles_So Happy Together.mp3	4.0 MB	192 Kbps	Cable	2:59 minutes	Jpkcr1	timeout
Turtles - So Happy Together.mp3	3.9 MB	192 Kbps	Cable	2:55 minutes	cfankeny	timeout
Me First and Gimmie Gimmies - So Happy Together (punk Turtles co...	1.8 MB	128 Kbps	T3+	2:01 minutes	dave87777	timeout
Copy of A New Found Glory - So Happy Together (punk Turtles cover...	1.8 MB	128 Kbps	T3+	2:01 minutes	corky273	timeout
Turtles--So Happy Together.mp3	2.6 MB	128 Kbps	T1	2:56 minutes	Zena5000	timeout
A New Found Glory - So Happy Together (punk Turtles cover).mp3	1.8 MB	128 Kbps	T1	2:01 minutes	keanie	timeout
The Turtles - So Happy Together.mp3	2.6 MB	128 Kbps	DSL	2:58 minutes	vizbiz	timeout
Turtles-So Happy Together.mp3	2.6 MB	128 Kbps	DSL	2:56 minutes	whazoo100	timeout

Transfer Manager

Download Order	Progress	Size	Speed	Tim
Andrew Lloyd Webber - Think of Me - The Phantom of the Opera - Disc 1 ...		4.9 MB	5.5 K/sec	10
John Williams - Duel Of The Fates.mp3		4.8 MB	14.2 K/sec	2
02_-_john_williams_-_duel_of_the_fates.mp3		4.8 MB	4.9 K/sec	5
Paul Simon - You Can Call Me Al.mp3		5.3 MB	2.8 K/sec	29
Paul Simon-You Can Call Me Al.mp3		5.3 MB	3.0 K/sec	26
Paul Simon-You Can Call Me Al.mp3		5.3 MB	1.6 K/sec	54
FF4- Main theme (Enya Remix).mp3		4.2 MB	12.4 K/sec	3

xx90's Legacy

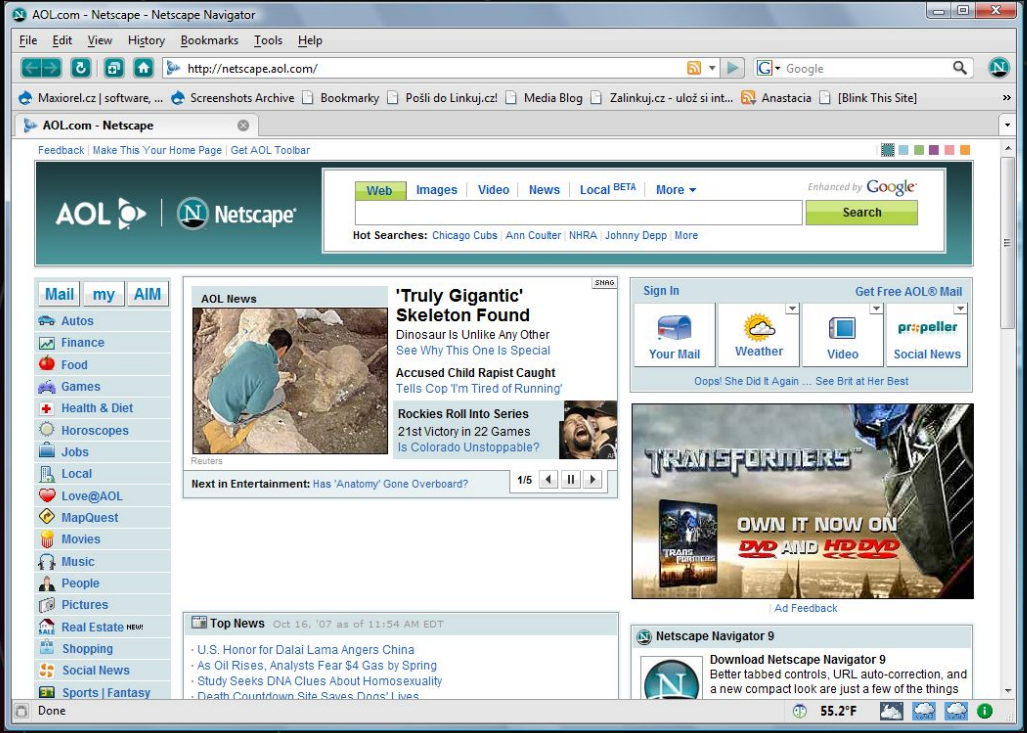


1+ Billion Streams [ALL ordered by Streams] | Most played tracks on Spotify

#	TITLE	ALBUM	DATE ADDED	🕒
1	Blinding Lights The Weeknd	After Hours	Jun 4, 2020	3:20
2	Shape of You Ed Sheeran	÷ (Deluxe)	Jun 14, 2017	3:53
3	Dance Monkey Tones And I	Dance Monkey (Stripped Back) / Dance Mon...	Feb 4, 2020	3:29
4	Someone You Loved Lewis Capaldi	Divinely Uninspired To A Hellish Extent	Apr 26, 2020	3:02
5	rockstar (feat. 21 Savage) E Post Malone, 21 Savage	beerbongs & bentleys	Jun 12, 2018	3:38
6	Sunflower - Spider-Man: Into the Spider-Verse Post Malone, Swae Lee	Spider-Man: Into the Spider-Verse (Soundtra...	Aug 27, 2019	2:38
7	One Dance Drake, Wizkid, Kyla	Views	Dec 16, 2016	2:53
8	Closer The Chainsmokers, Halsey	Closer	Jun 5, 2017	4:04
9	Believer Imagine Dragons	Evolve	May 7, 2019	3:24
10	Señorita Shawn Mendes, Camila Cabello	Shawn Mendes (Deluxe)	Jan 3, 2020	3:10
11	STAY (with Justin Bieber) E The Kid LAROI, Justin Bieber	F*CK LOVE 3: OVER YOU	Nov 11, 2021	2:21
12	Perfect Ed Sheeran	÷ (Deluxe)	Jan 28, 2019	4:23

TAURI

xx00’s Problem



Hot Searches: Chicago Cubs | Ann Coulter | NHRA | Johnny Depp | More

'Truly Gigantic' Skeleton Found

Dinosaur Is Unlike Any Other

See Why This One Is Special

Accused Child Rapist Caught

Tells Cop 'I'm Tired of Running'

Rockies Roll Into Series

21st Victory in 22 Games

Is Colorado Unstoppable?

Next in Entertainment: Has 'Anatomy' Gone Overboard?

Oops! She Did It Again ... See Brit at Her Best

Top News Oct 16, '07 as of 11:54 AM EDT

- U.S. Honor for Dalai Lama Angers China
- As Oil Rises, Analysts Fear $4 Gas by Spring
- Study Seeks DNA Clues About Homosexuality
- Death Countdown Site Saves Dogs' Lives

Download Netscape Navigator 9

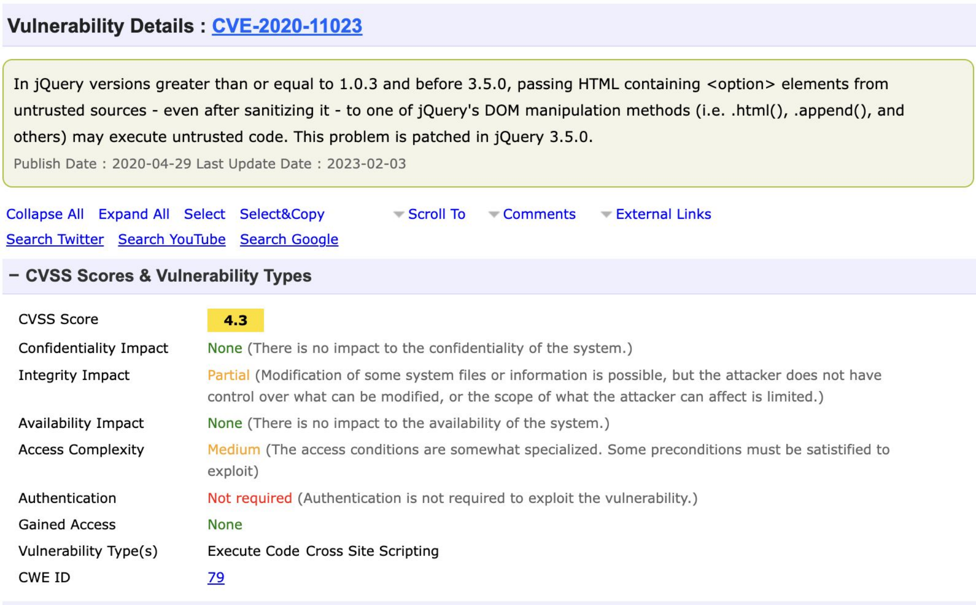
Better tabbed controls, URL auto-correction, and a new compact look are just a few of the things

xx00’s Cheatcode

jQuery

TAURI

xx00's Legacy



Vulnerability Details : CVE-2020-11023

In jQuery versions greater than or equal to 1.0.3 and before 3.5.0, passing HTML containing <option> elements from untrusted sources - even after sanitizing it - to one of jQuery's DOM manipulation methods (i.e. .html(), .append(), and others) may execute untrusted code. This problem is patched in jQuery 3.5.0.

Publish Date : 2020-04-29 Last Update Date : 2023-02-03

Collapse All Expand All Select Select&Copy
Scroll To
Comments
External Links
Search Twitter Search YouTube Search Google

− CVSS Scores & Vulnerability Types

CVSS Score	4.3
Confidentiality Impact	None (There is no impact to the confidentiality of the system.)
Integrity Impact	Partial (Modification of some system files or information is possible, but the attacker does not have control over what can be modified, or the scope of what the attacker can affect is limited.)
Availability Impact	None (There is no impact to the availability of the system.)
Access Complexity	Medium (The access conditions are somewhat specialized. Some preconditions must be satistified to exploit)
Authentication	Not required (Authentication is not required to exploit the vulnerability.)
Gained Access	None
Vulnerability Type(s)	Execute Code Cross Site Scripting
CWE ID	79

TAURI

xx10’s Problem

HTML, CSS, JS

xx10’s Cheatcode

TS

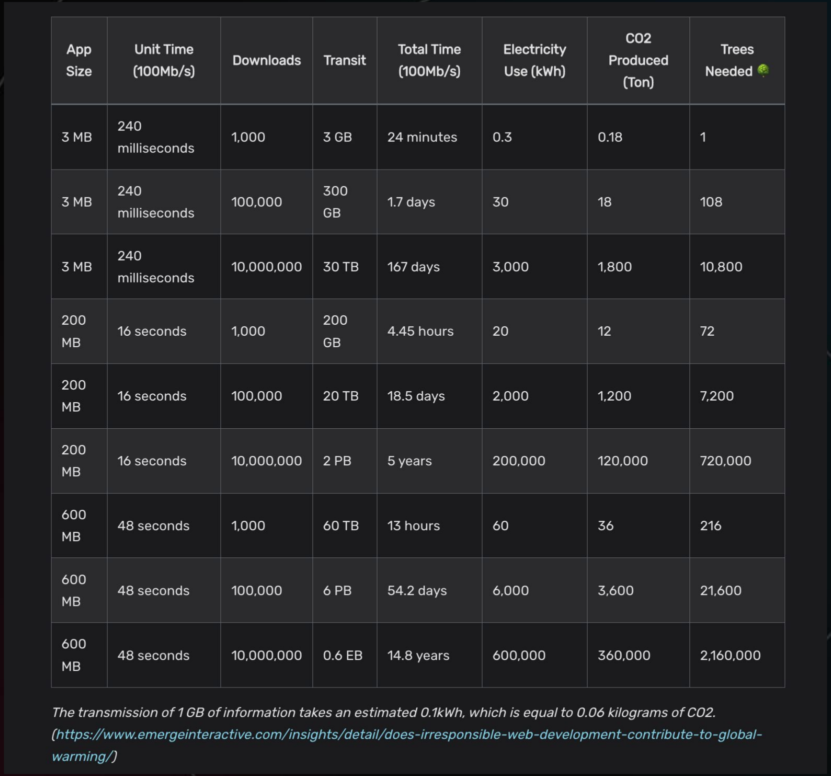
NativeScript

TAURI

BABEL

core-js

xx10’s Legacy



App Size	Unit Time (100Mb/s)	Downloads	Transit	Total Time (100Mb/s)	Electricity Use (kWh)	CO2 Produced (Ton)	Trees Needed 🌳
3 MB	240 milliseconds	1,000	3 GB	24 minutes	0.3	0.18	1
3 MB	240 milliseconds	100,000	300 GB	1.7 days	30	18	108
3 MB	240 milliseconds	10,000,000	30 TB	167 days	3,000	1,800	10,800
200 MB	16 seconds	1,000	200 GB	4.45 hours	20	12	72
200 MB	16 seconds	100,000	20 TB	18.5 days	2,000	1,200	7,200
200 MB	16 seconds	10,000,000	2 PB	5 years	200,000	120,000	720,000
600 MB	48 seconds	1,000	60 TB	13 hours	60	36	216
600 MB	48 seconds	100,000	6 PB	54.2 days	6,000	3,600	21,600
600 MB	48 seconds	10,000,000	0.6 EB	14.8 years	600,000	360,000	2,160,000

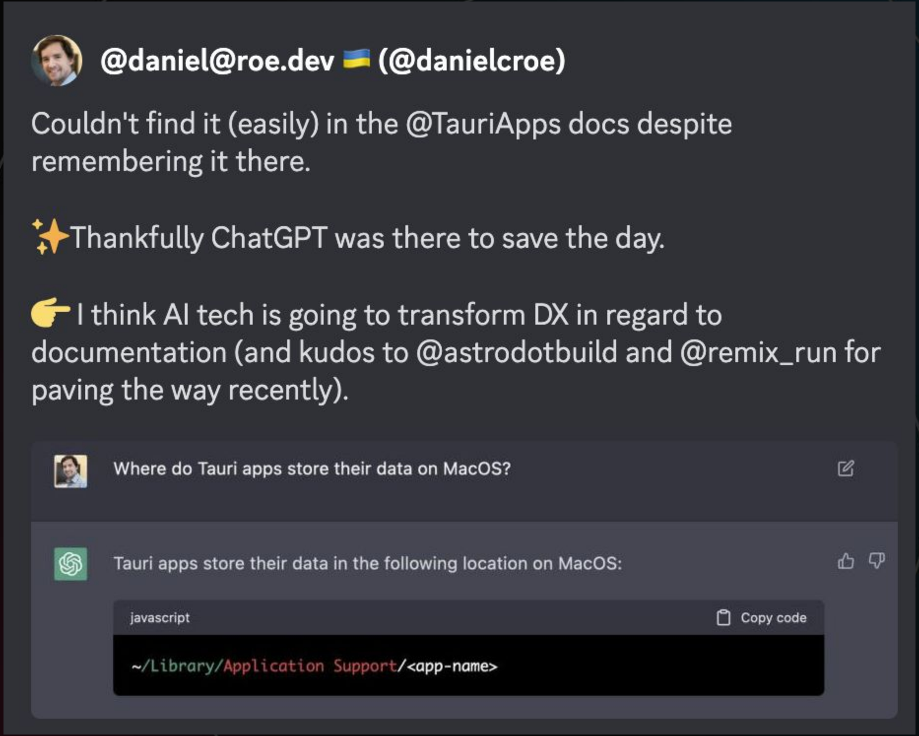
The transmission of 1 GB of information takes an estimated 0.1kWh, which is equal to 0.06 kilograms of CO2. (https://www.emergeinteractive.com/insights/detail/does-irresponsible-web-development-contribute-to-global-warming/)

xx20’s Problem

Remembering what it is that you forgot.

TAURI

xx20’s Cheatcode



@daniel@roe.dev 🇺🇦 (@danielcroe)

Couldn't find it (easily) in the @TauriApps docs despite remembering it there.

✨Thankfully ChatGPT was there to save the day.

👉I think AI tech is going to transform DX in regard to documentation (and kudos to @astrodotbuild and @remix_run for paving the way recently).

Where do Tauri apps store their data on MacOS?

Tauri apps store their data in the following location on MacOS:

```javascript
~/Library/Application Support/<app-name>
```

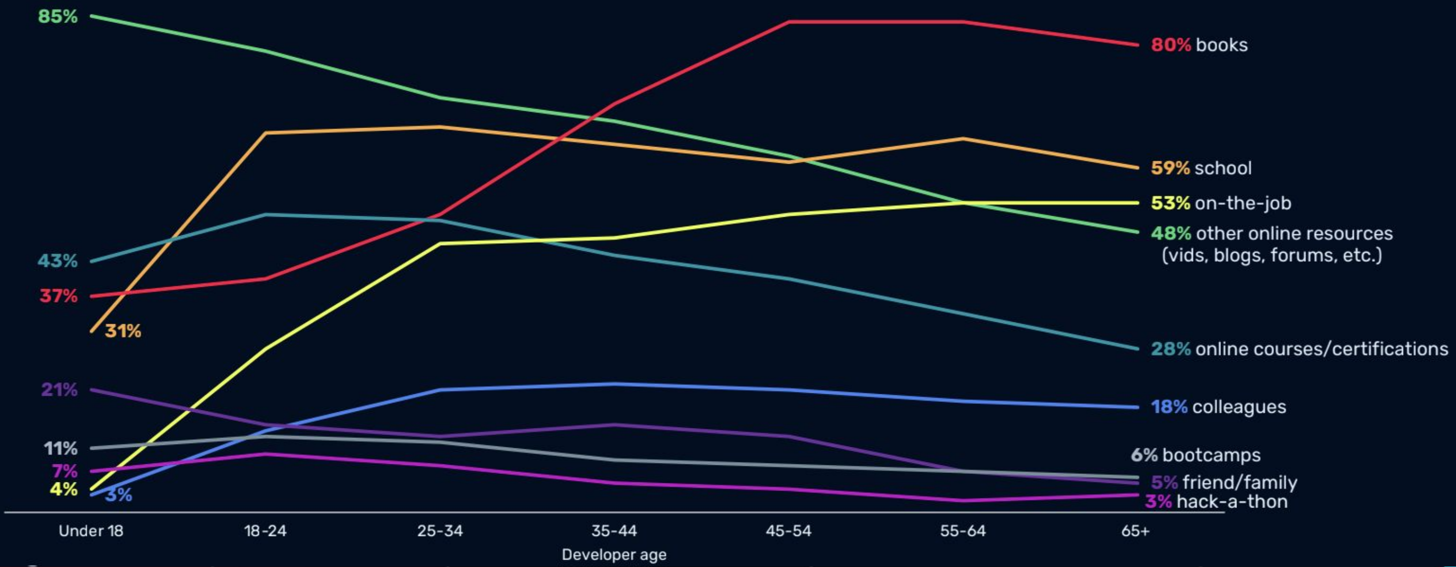
TAURI

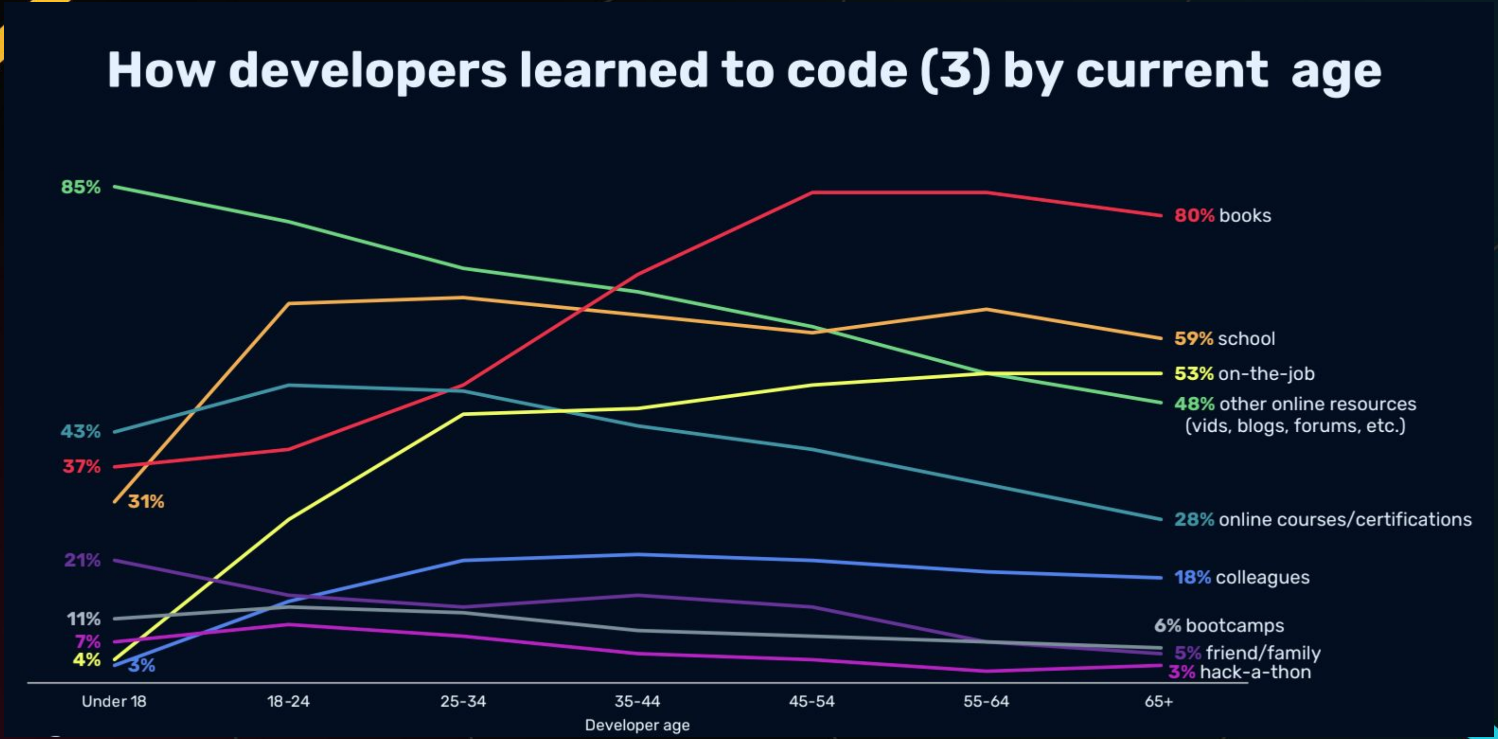
xx20's Legacy

> **“I never said it would be easy,
> I only said it would be worth it.”**
>
> Mae West

TAURI

How developers learned to code (3) by current age





How developers learned to code (3) by current age

242% **300%**

So let’s take a step back.

Younger developers haven’t had as much time to gain experience through learning - but this also shows a hidden trend in the world of software engineering.

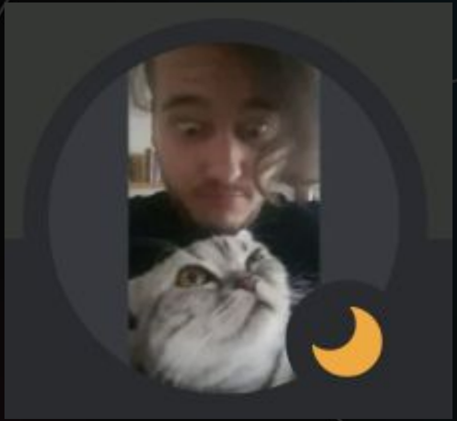
The ease with which software can be built has been streamlined with what is known as Developer Experience, which is kind of an ironic turn of phrase. Syntactic sugar and guardrails make things “easier”, while obscuring the underlying nature of code.

We’re all a little Rusty.

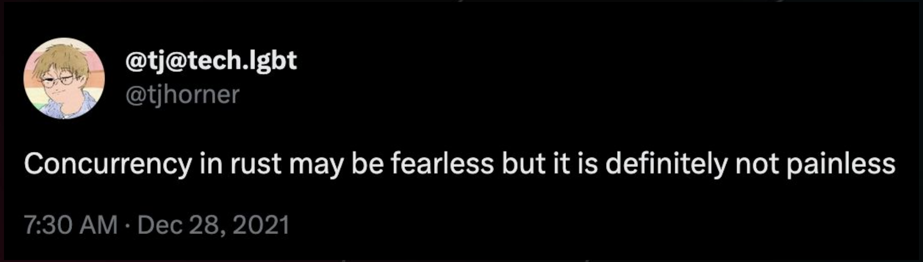
And asking for help from friends is something that a community does very well.

TAURI

We’re all a little Rusty.



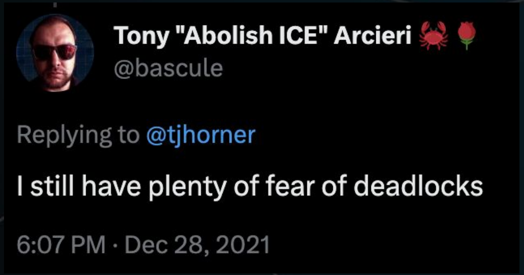
We’re all a little Rusty.



@tj@tech.lgbt
@tjhorner

Concurrency in rust may be fearless but it is definitely not painless

7:30 AM · Dec 28, 2021

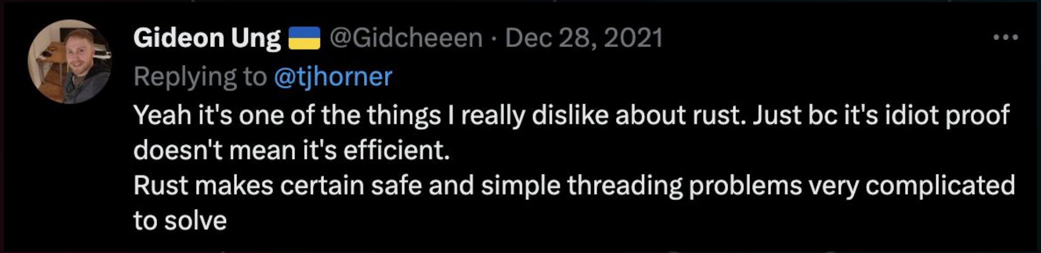


Tony "Abolish ICE" Arcieri 🦀🌹
@bascule

Replying to @tjhorner

I still have plenty of fear of deadlocks

6:07 PM · Dec 28, 2021



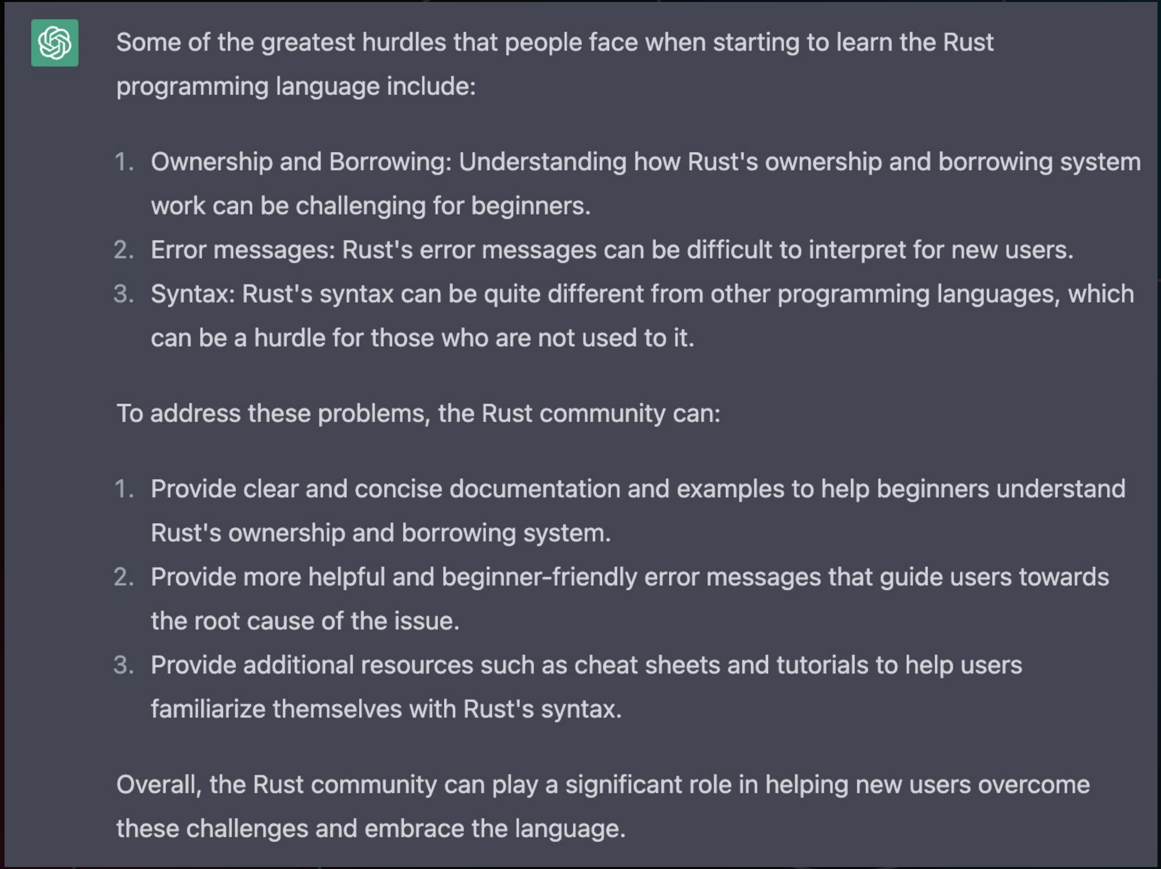
Gideon Ung 🇺🇦 @Gidcheeen · Dec 28, 2021 ···
Replying to @tjhorner

Yeah it's one of the things I really dislike about rust. Just bc it's idiot proof doesn't mean it's efficient.
Rust makes certain safe and simple threading problems very complicated to solve

TAURI

We’re all a little Rusty.

But thankfully synthetic knowledge has arrived, where you don’t have to out yourself as a noob…



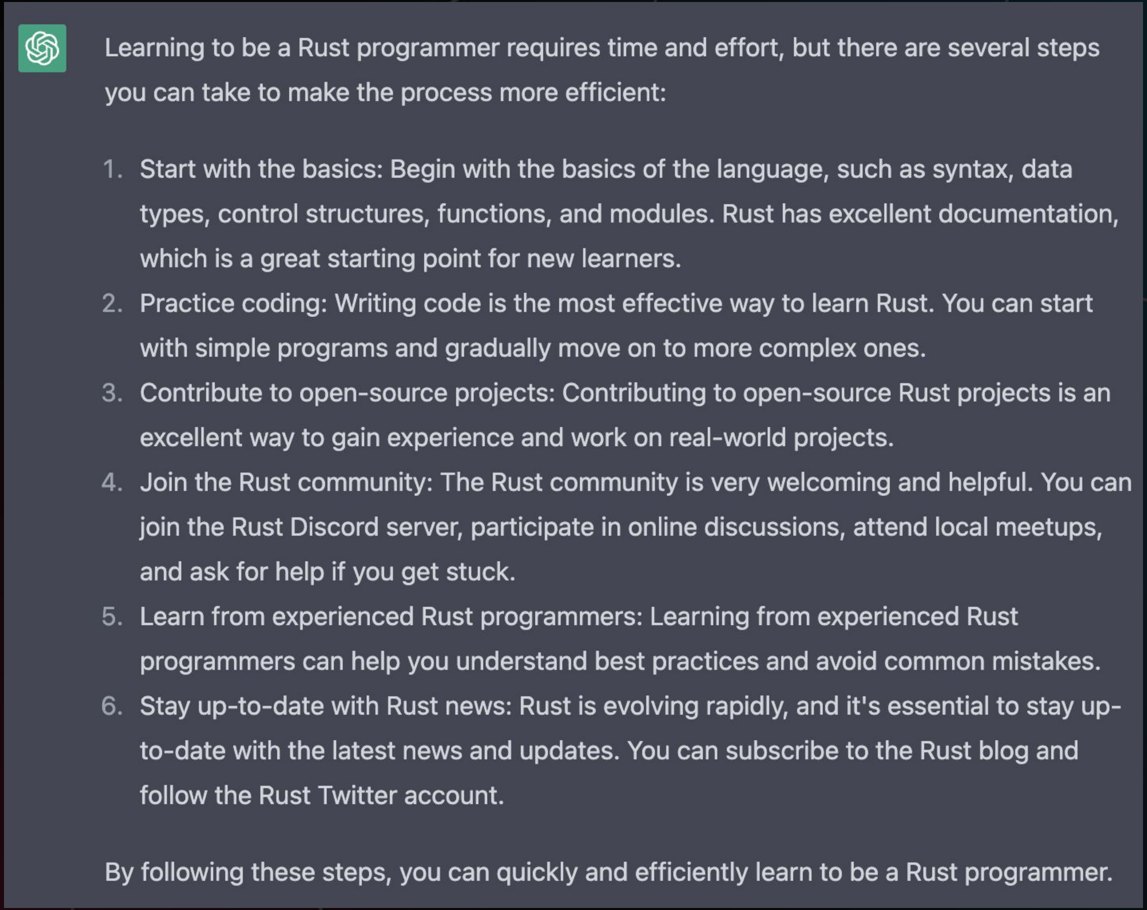
Some of the greatest hurdles that people face when starting to learn the Rust programming language include:

1. Ownership and Borrowing: Understanding how Rust's ownership and borrowing system work can be challenging for beginners.
2. Error messages: Rust's error messages can be difficult to interpret for new users.
3. Syntax: Rust's syntax can be quite different from other programming languages, which can be a hurdle for those who are not used to it.

To address these problems, the Rust community can:

1. Provide clear and concise documentation and examples to help beginners understand Rust's ownership and borrowing system.
2. Provide more helpful and beginner-friendly error messages that guide users towards the root cause of the issue.
3. Provide additional resources such as cheat sheets and tutorials to help users familiarize themselves with Rust's syntax.

Overall, the Rust community can play a significant role in helping new users overcome these challenges and embrace the language.



Learning to be a Rust programmer requires time and effort, but there are several steps you can take to make the process more efficient:

1. Start with the basics: Begin with the basics of the language, such as syntax, data types, control structures, functions, and modules. Rust has excellent documentation, which is a great starting point for new learners.
2. Practice coding: Writing code is the most effective way to learn Rust. You can start with simple programs and gradually move on to more complex ones.
3. Contribute to open-source projects: Contributing to open-source Rust projects is an excellent way to gain experience and work on real-world projects.
4. Join the Rust community: The Rust community is very welcoming and helpful. You can join the Rust Discord server, participate in online discussions, attend local meetups, and ask for help if you get stuck.
5. Learn from experienced Rust programmers: Learning from experienced Rust programmers can help you understand best practices and avoid common mistakes.
6. Stay up-to-date with Rust news: Rust is evolving rapidly, and it's essential to stay up-to-date with the latest news and updates. You can subscribe to the Rust blog and follow the Rust Twitter account.

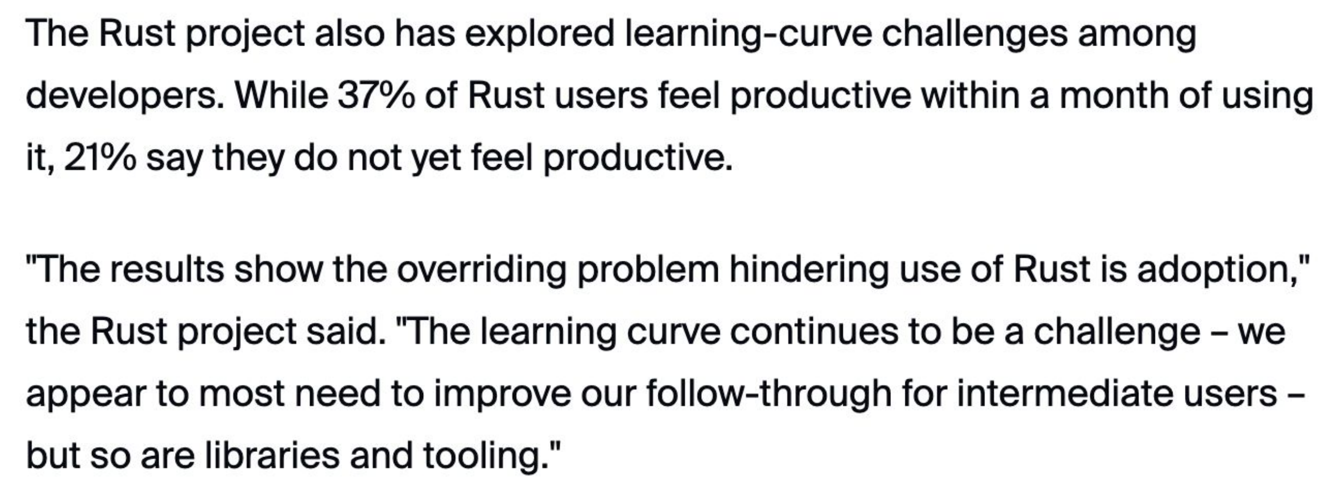
By following these steps, you can quickly and efficiently learn to be a Rust programmer.

TAURI

We know all this.

But, how can we as a community adapt to evolving ways of learning and working?

What will be our legacy?



The Rust project also has explored learning-curve challenges among developers. While 37% of Rust users feel productive within a month of using it, 21% say they do not yet feel productive.

"The results show the overriding problem hindering use of Rust is adoption," the Rust project said. "The learning curve continues to be a challenge – we appear to most need to improve our follow-through for intermediate users – but so are libraries and tooling."

https://www.zdnet.com/article/programming-language-rusts-adoption-problem-developers-reveal-why-more-arent-using-it/

Design your Org Like Software

Separate Concerns

Maintain Transparency

Empower Access

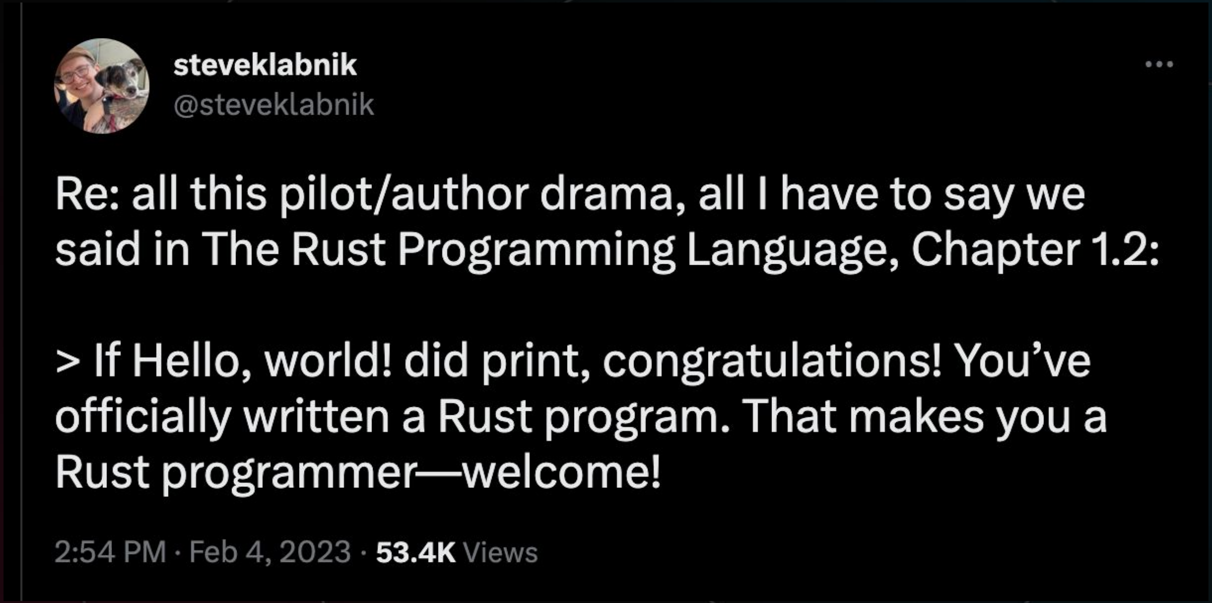
Stay Resilient

Reduce Bus Factor

Define Done == “rustable”

TAURI

Reframe Success



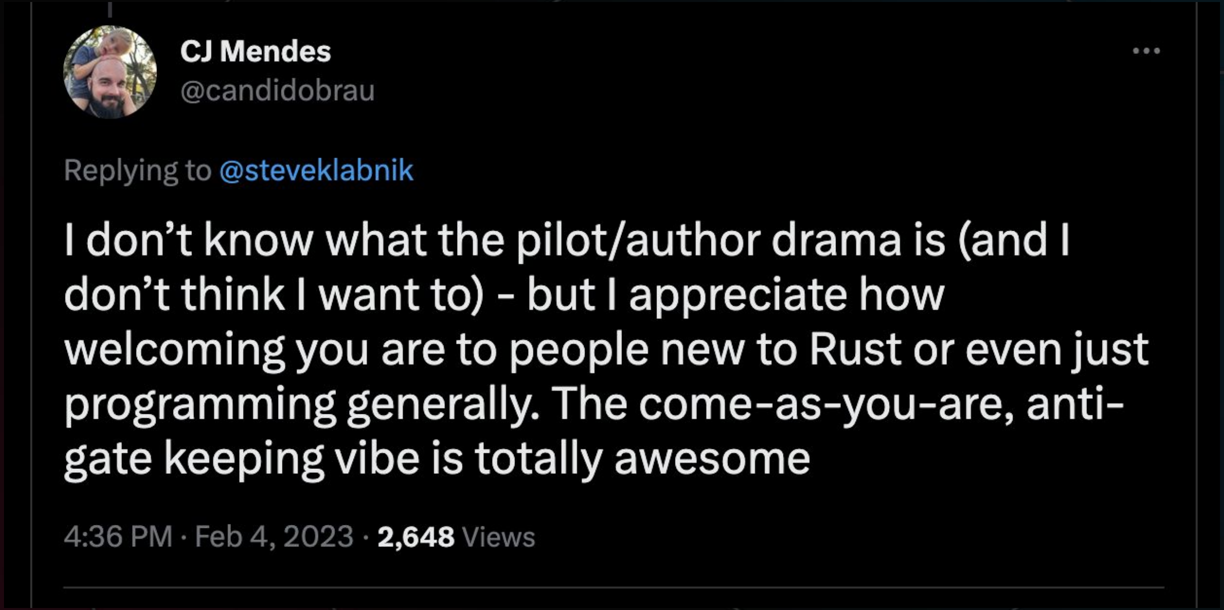
steveklabnik
@steveklabnik

Re: all this pilot/author drama, all I have to say we said in The Rust Programming Language, Chapter 1.2:

> If Hello, world! did print, congratulations! You’ve officially written a Rust program. That makes you a Rust programmer—welcome!

2:54 PM · Feb 4, 2023 · **53.4K** Views

Reframe Participation



CJ Mendes
@candidobrau

Replying to @steveklabnik

I don’t know what the pilot/author drama is (and I don’t think I want to) - but I appreciate how welcoming you are to people new to Rust or even just programming generally. The come-as-you-are, anti-gate keeping vibe is totally awesome

4:36 PM · Feb 4, 2023 · **2,648** Views

TAURI

Foster Engagement

Tauri’s working group:

- Open and accessible to all
- Inclusionary and Supportive
- Grows learners into teachers

TAURI

Get Involved

https://discord.gg/tauri

TAURI

Commons Conservancy

Our promise to open source.

https://dracc.commonsconservancy.org/0035/

TAURI

Open Collective

Donating makes you feel good.

https://opencollective.com/tauri

TAURI

Thank You

tauri.app

@TauriApps

denjell@tauri.app

TAURI

Problem Statement

1. Writing software is hard, even for the pros
2. Open source can instrumentalize the precariat
3. Set and forget has [security] implications
4. Beginners have to start somewhere
5. Traditional learning encourages

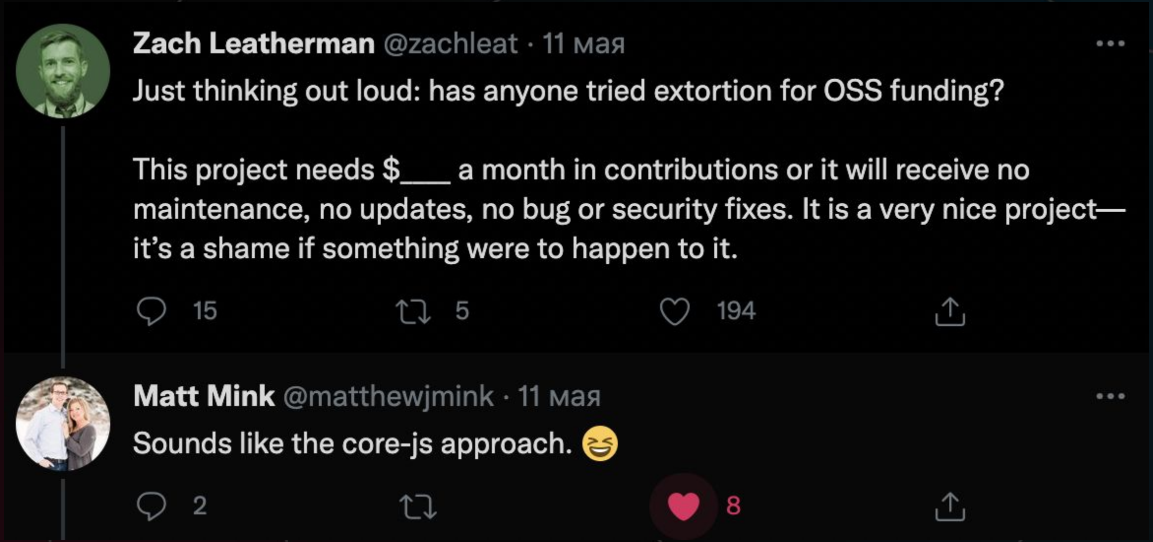
TAURI

Thanks

https://tauri.app

TAURI

Communal.



Zach Leatherman @zachleat · 11 мая

Just thinking out loud: has anyone tried extortion for OSS funding?

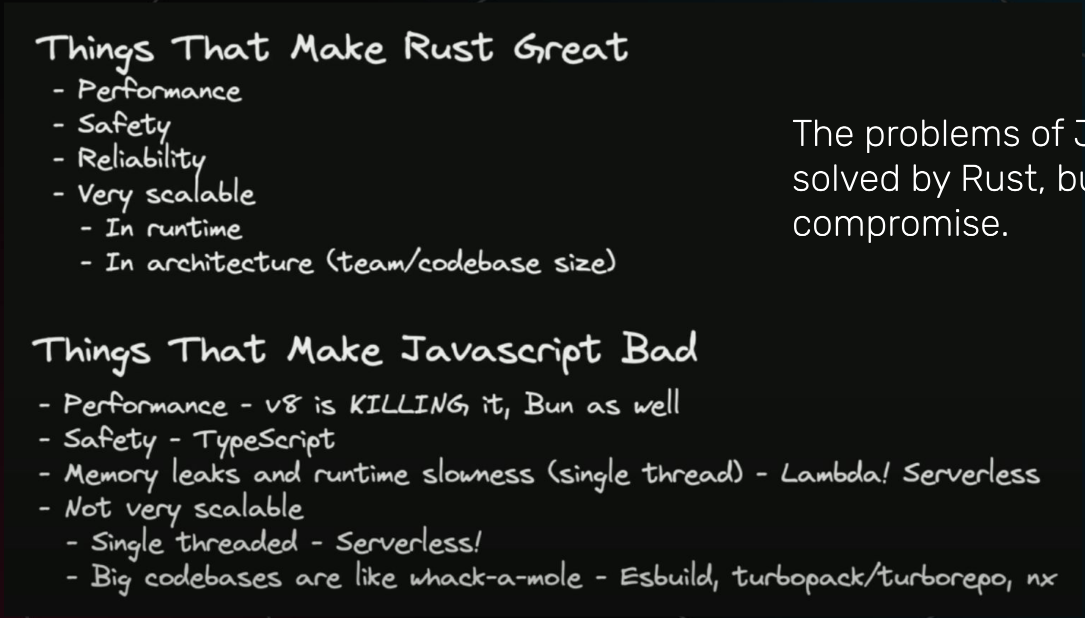
This project needs $___ a month in contributions or it will receive no maintenance, no updates, no bug or security fixes. It is a very nice project—it's a shame if something were to happen to it.

Matt Mink @matthewjmink · 11 мая

Sounds like the core-js approach. 😆

TAURI

Compromise.



Things That Make Rust Great

- Performance
- Safety
- Reliability
- Very scalable
  - In runtime
  - In architecture (team/codebase size)

Things That Make Javascript Bad

- Performance - v8 is KILLING it, Bun as well
- Safety - TypeScript
- Memory leaks and runtime slowness (single thread) - Lambda! Serverless
- Not very scalable
  - Single threaded - Serverless!
  - Big codebases are like whack-a-mole - Esbuild, turbopack/turborepo, nx

The problems of JavaScript are solved by Rust, but at great compromise.

TAURI

https://www.youtube.com/watch?v=kOFWIvNowXo

What is Tauri?

App Development Framework
and Global Community
For Desktop (now)
and Mobile Apps (alpha)

TAURI

Who uses Tauri?

Small teams, individuals

Many devs come from JS background

(Web, Electron, Capacitor)

Why use Tauri?

It is a Rust-based GUI available now.
It is security focussed.
To learn more about rust.

Why learn Rust?

Exciting, new, fun.
Friendly compiler.
Save energy.

DEEPTECH LIBRARY

Windows, Menus & System Trays

TAO

TAURI

INTRO

Every so often, something so fundamentally brilliant comes along that it challenges mainstream ideas about the right way to do things. The Rust language's adoption and its generated excitement align with this pattern.

Yet, many complain about how challenging it is to wrap their heads around the concepts that make the language and its tooling so robust. At Tauri, we have discovered that offering engineers low-friction pathways encourages experimentation, knowledge acquisition, and rapid mastery.

In this keynote, Tauri's co-founder Daniel Thompson will present key takeaways for enabling positive growth of the Rust community via empowerment and developer experience.

TAURI

DEEPTECH LIBRARY

Provides Webviews for Tauri

WRY

TAO

TAURI

Integrates Stack, Adds APIs and Build Tools

TAURI

WRY

TAO

Rust Features

Cross Platform

Memory Safety

High Performance

Amazing (and friendly) Compiler

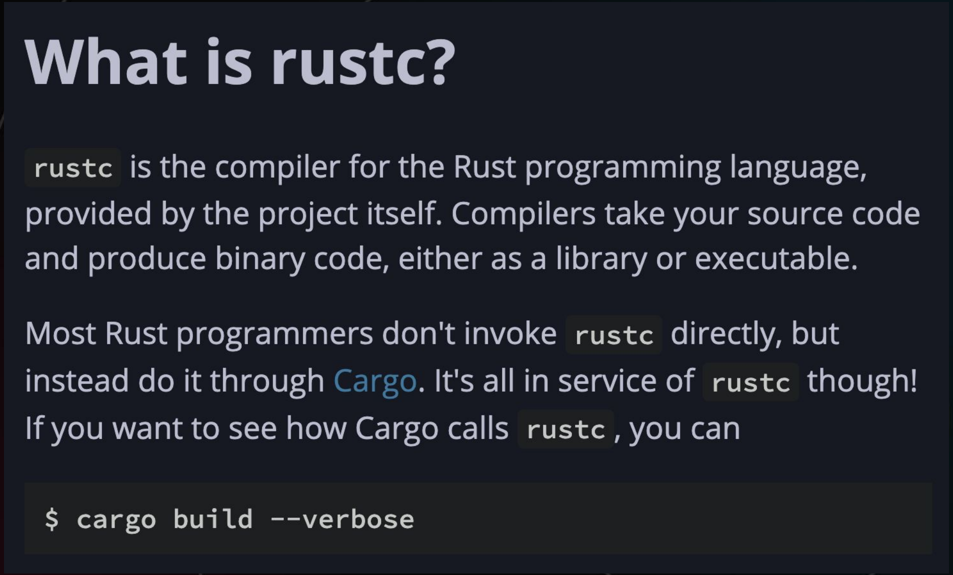
Built-in Documentation System

Quickly Growing Community

Ecosystem

rustc
cargo
rustup
crates.io
docs.rs

Ecosystem: rustc



What is rustc?

`rustc` is the compiler for the Rust programming language, provided by the project itself. Compilers take your source code and produce binary code, either as a library or executable.

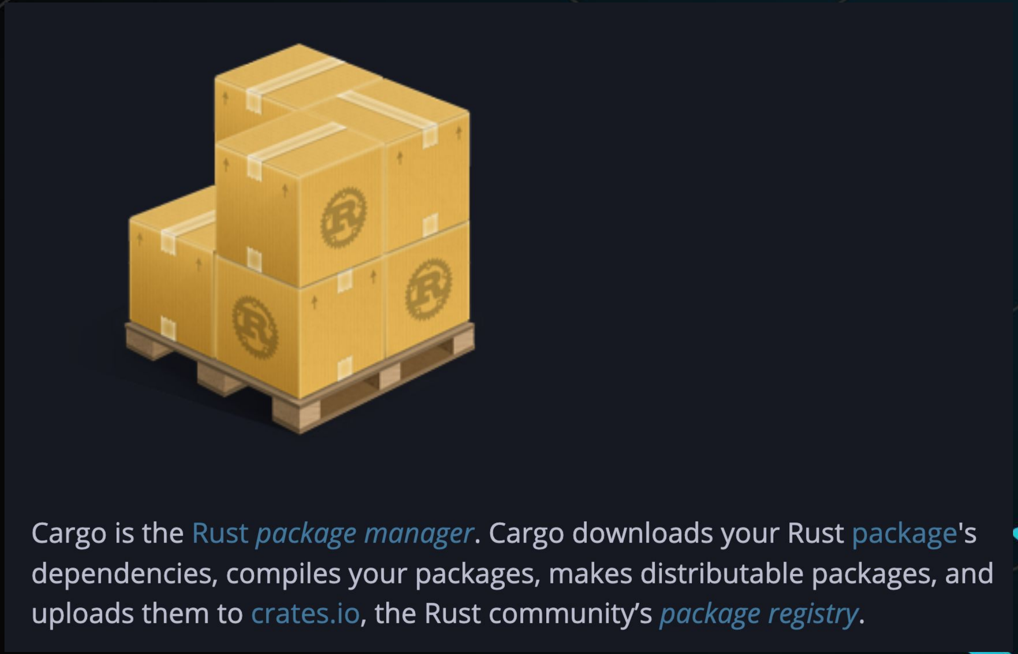
Most Rust programmers don't invoke `rustc` directly, but instead do it through Cargo. It's all in service of `rustc` though! If you want to see how Cargo calls `rustc`, you can

```
$ cargo build --verbose
```

Ecosystem: cargo

Cargo also provides for context aware “plugins”, like the tauri CLI:

```
cargo tauri build
```

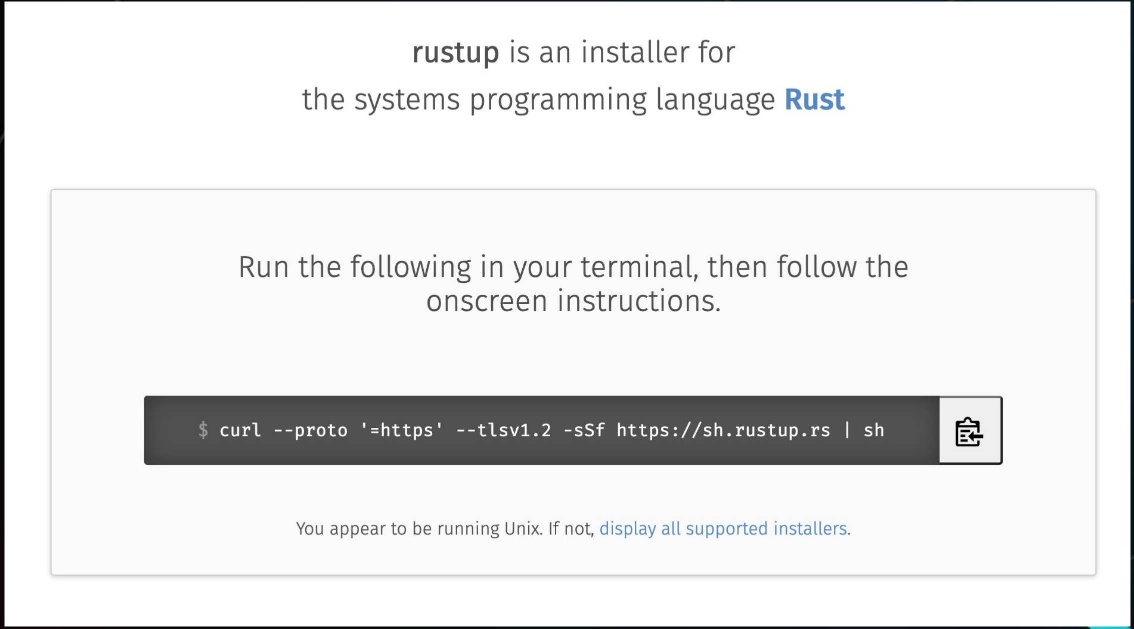


Cargo is the Rust *package manager*. Cargo downloads your Rust package's dependencies, compiles your packages, makes distributable packages, and uploads them to crates.io, the Rust community’s *package registry*.

TAURI

Ecosystem: rustup

- **rustc**
  Your compiler
- **triple**
  aarch64-apple-darwin
- **nightly**
  Living on the edge



rustup is an installer for
the systems programming language **Rust**

Run the following in your terminal, then follow the onscreen instructions.

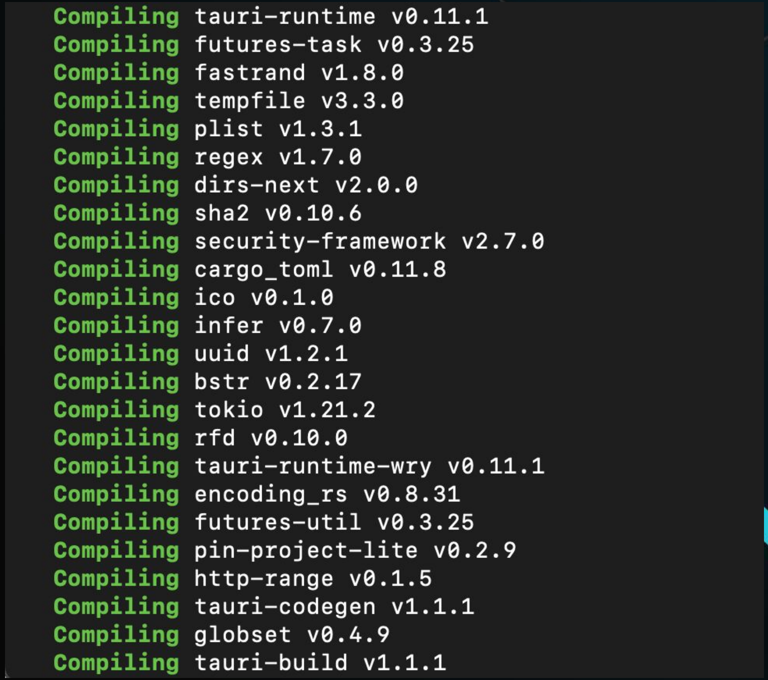
```
$ curl --proto '=https' --tlsv1.2 -sSf https://sh.rustup.rs | sh
```

You appear to be running Unix. If not, display all supported installers.

TAURI

Ecosystem: crates.io

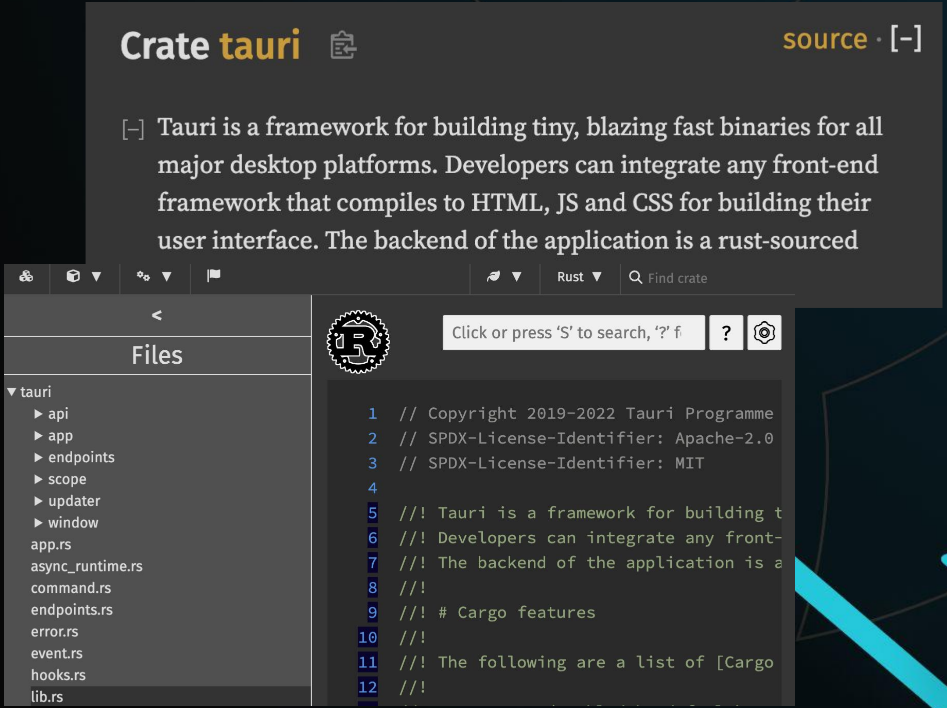
You don't have to publish a crate on crates.io - they can also be consumed via git or even from the local filesystem.



```
   Compiling tauri-runtime v0.11.1
   Compiling futures-task v0.3.25
   Compiling fastrand v1.8.0
   Compiling tempfile v3.3.0
   Compiling plist v1.3.1
   Compiling regex v1.7.0
   Compiling dirs-next v2.0.0
   Compiling sha2 v0.10.6
   Compiling security-framework v2.7.0
   Compiling cargo_toml v0.11.8
   Compiling ico v0.1.0
   Compiling infer v0.7.0
   Compiling uuid v1.2.1
   Compiling bstr v0.2.17
   Compiling tokio v1.21.2
   Compiling rfd v0.10.0
   Compiling tauri-runtime-wry v0.11.1
   Compiling encoding_rs v0.8.31
   Compiling futures-util v0.3.25
   Compiling pin-project-lite v0.2.9
   Compiling http-range v0.1.5
   Compiling tauri-codegen v1.1.1
   Compiling globset v0.4.9
   Compiling tauri-build v1.1.1
```

Ecosystem: docs.rs

Fun fact: Rust provides a number of ways to generate docs from comments (and the code itself).



Crate tauri

source · [−]

[−] Tauri is a framework for building tiny, blazing fast binaries for all major desktop platforms. Developers can integrate any front-end framework that compiles to HTML, JS and CSS for building their user interface. The backend of the application is a rust-sourced

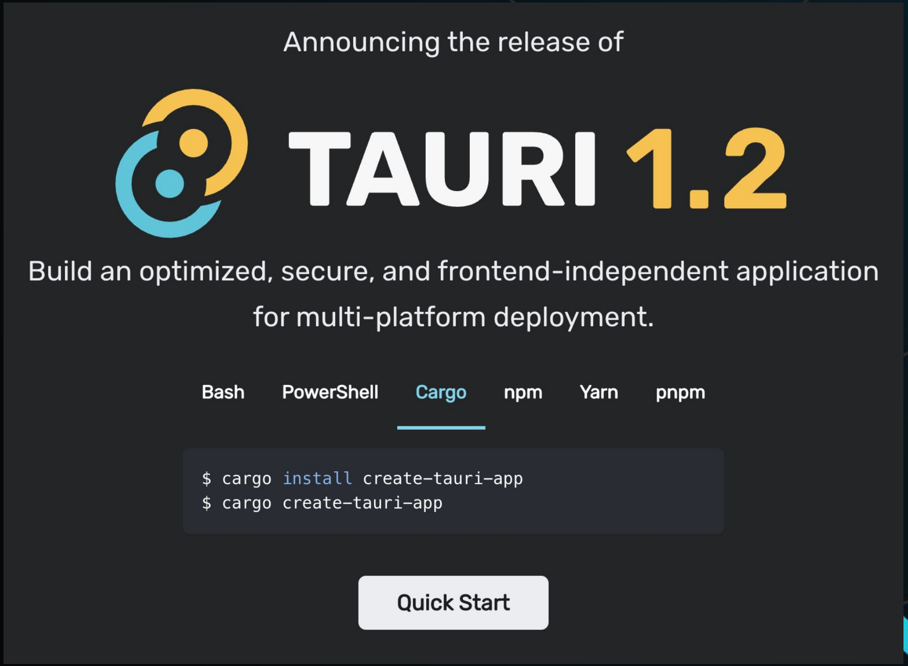
```
// Copyright 2019-2022 Tauri Programme
// SPDX-License-Identifier: Apache-2.0
// SPDX-License-Identifier: MIT

//! Tauri is a framework for building t
//! Developers can integrate any front-
//! The backend of the application is a
//!
//! # Cargo features
//!
//! The following are a list of [Cargo
//!
```

TAURI

Ecosystem: Tauri

Tauri wants to help you set up your system. Visit our page to do it right:

https://tauri.app/v1/guides/getting-started/prerequisites



Announcing the release of

TAURI 1.2

Build an optimized, secure, and frontend-independent application for multi-platform deployment.

```
$ cargo install create-tauri-app
$ cargo create-tauri-app
```

Rust Structure

./Cargo.toml

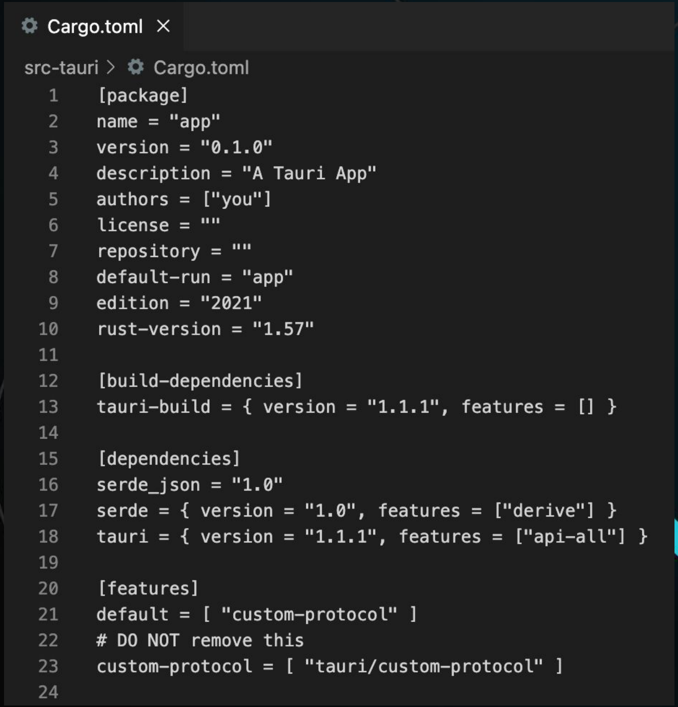
./src/main.rs

./build.rs

TAURI

./Cargo.toml

The Cargo.toml file defines the project, its dependencies and features.



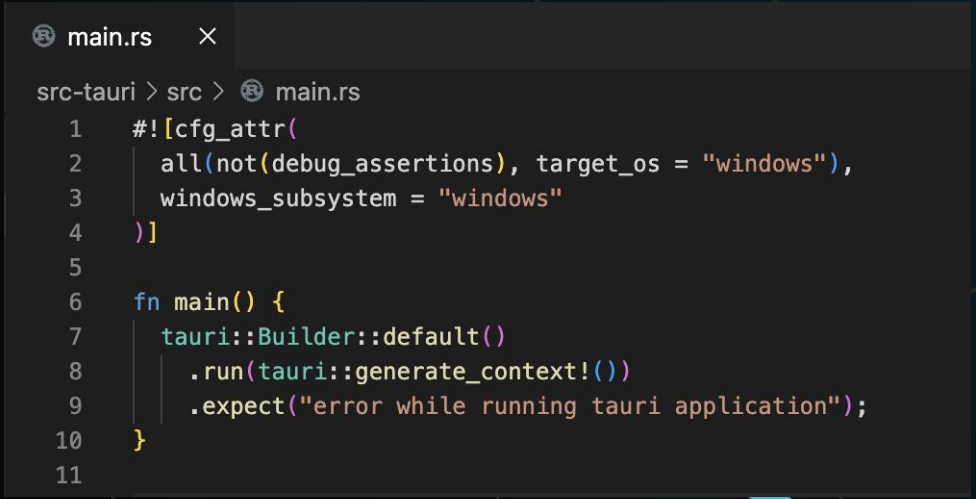
```toml
[package]
name = "app"
version = "0.1.0"
description = "A Tauri App"
authors = ["you"]
license = ""
repository = ""
default-run = "app"
edition = "2021"
rust-version = "1.57"

[build-dependencies]
tauri-build = { version = "1.1.1", features = [] }

[dependencies]
serde_json = "1.0"
serde = { version = "1.0", features = ["derive"] }
tauri = { version = "1.1.1", features = ["api-all"] }

[features]
default = [ "custom-protocol" ]
# DO NOT remove this
custom-protocol = [ "tauri/custom-protocol" ]

```

TAURI

./src/main.rs

The main.rs file sets up the main loop, and executes it.

Note that you can choose what code to conditionally compile.

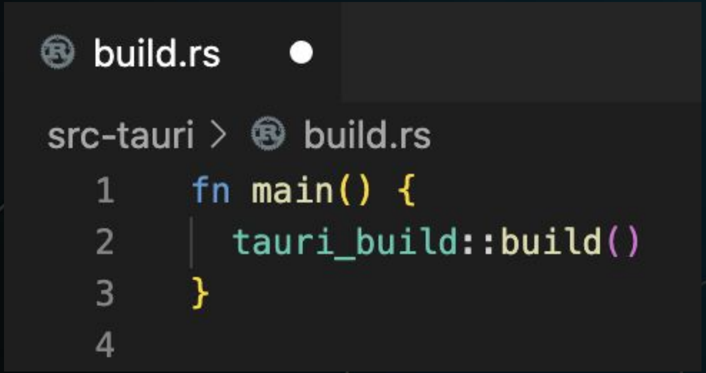


```rust
#![cfg_attr(
    all(not(debug_assertions), target_os = "windows"),
    windows_subsystem = "windows"
)]

fn main() {
    tauri::Builder::default()
        .run(tauri::generate_context!())
        .expect("error while running tauri application");
}

```

TAURI

./build.rs

The build.rs file is executed at build time, and allows for platform specific compilation and other tasks, like interacting with environment variables.

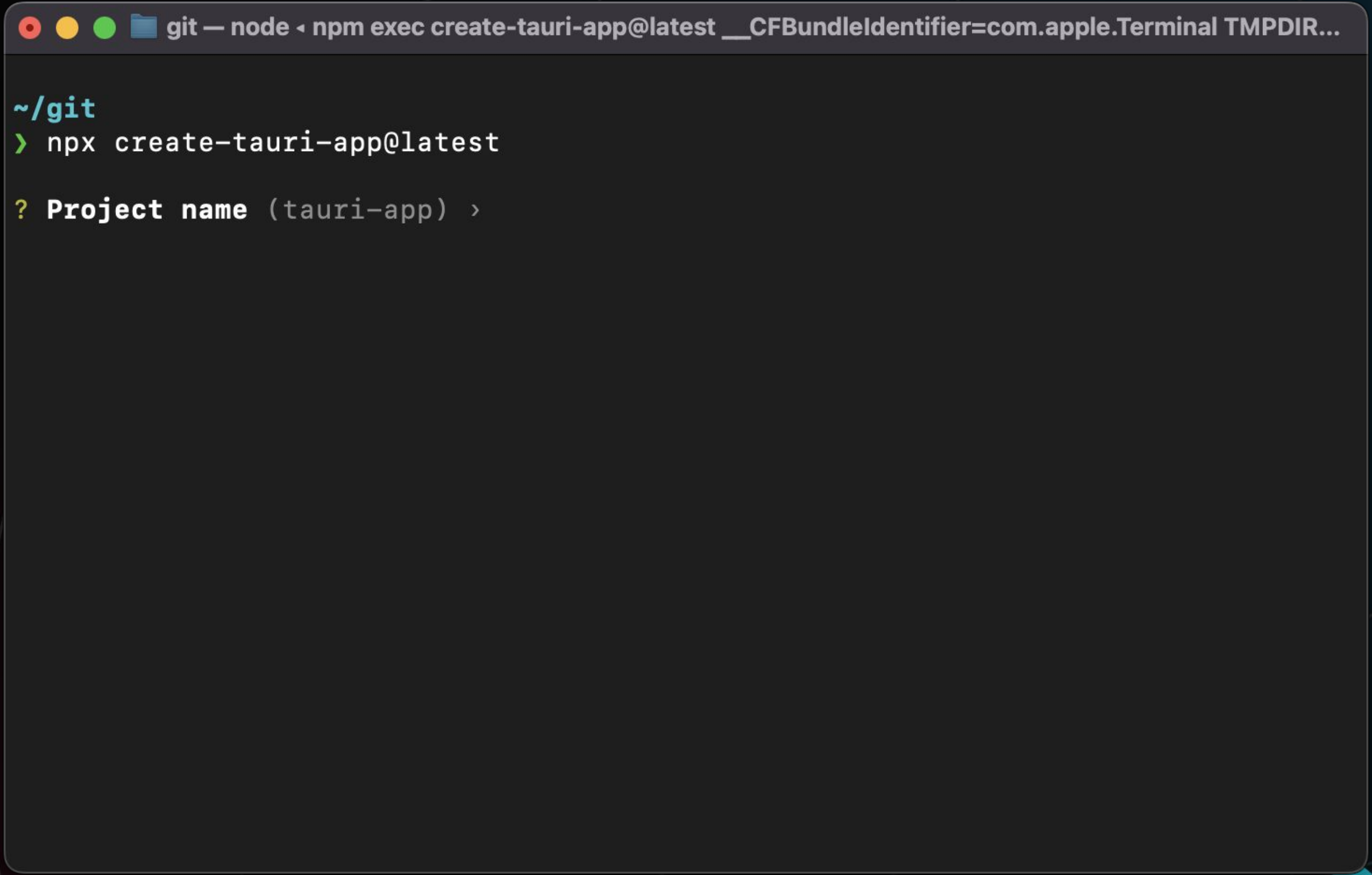


```rust
fn main() {
  tauri_build::build()
}
```

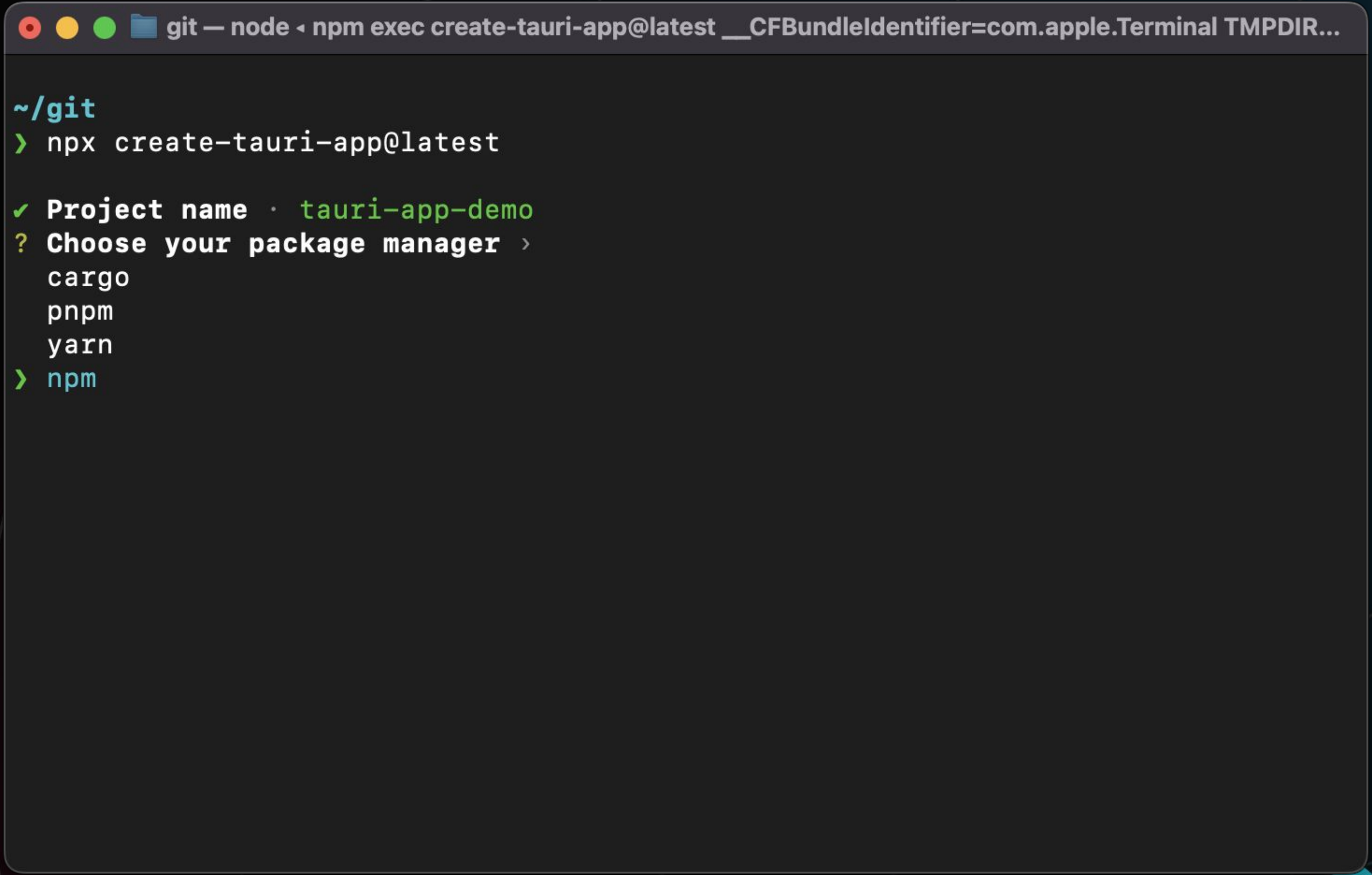
Tauri Helps You

```
> npx create-tauri-app@latest
> tauri dev
> tauri build
```

TAURI


git — node ◂ npm exec create-tauri-app@latest __CFBundleIdentifier=com.apple.Terminal TMPDIR...

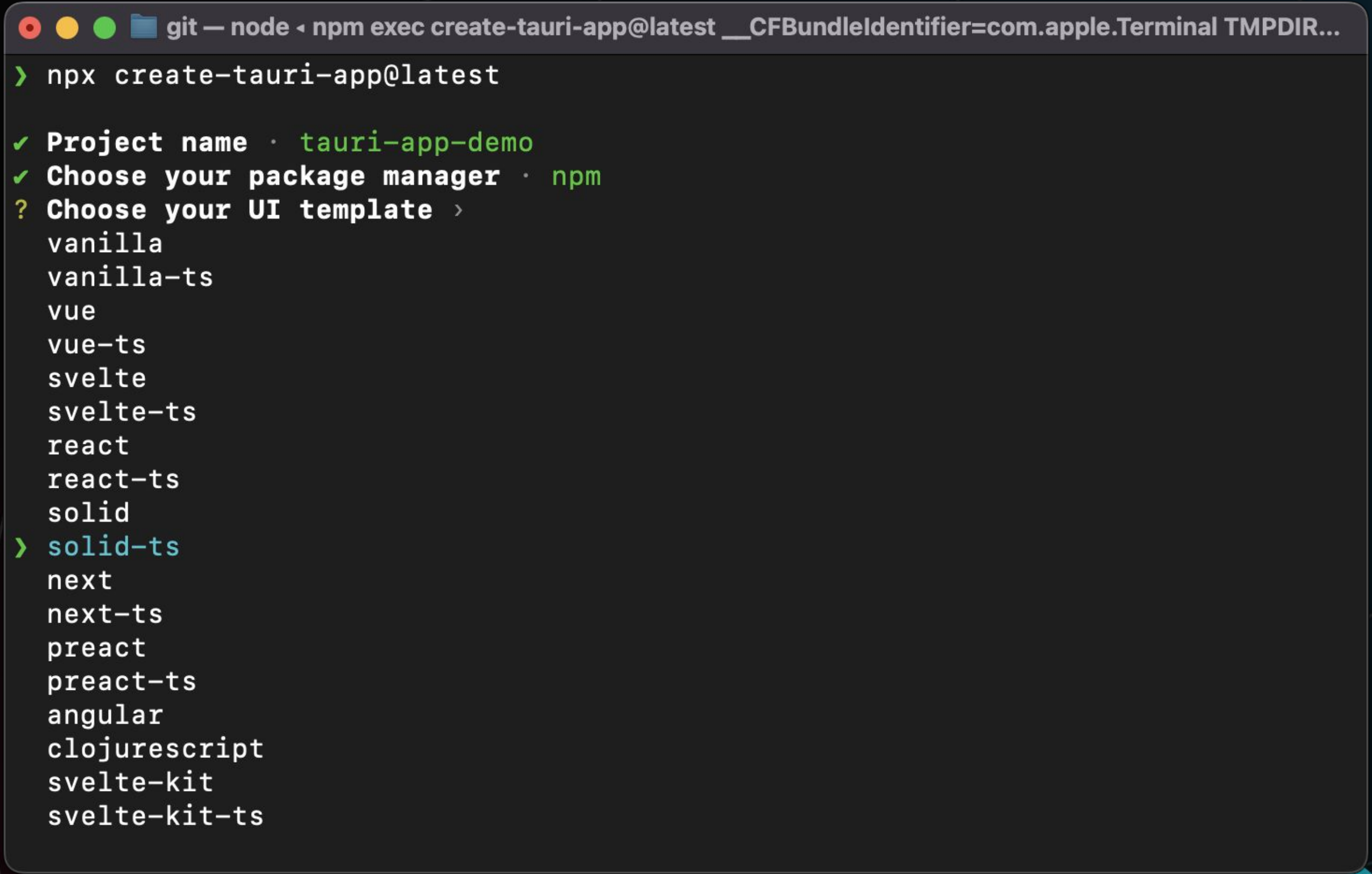
```
~/git
❯ npx create-tauri-app@latest

? Project name (tauri-app) ›
```



git — node ◂ npm exec create-tauri-app@latest __CFBundleIdentifier=com.apple.Terminal TMPDIR...

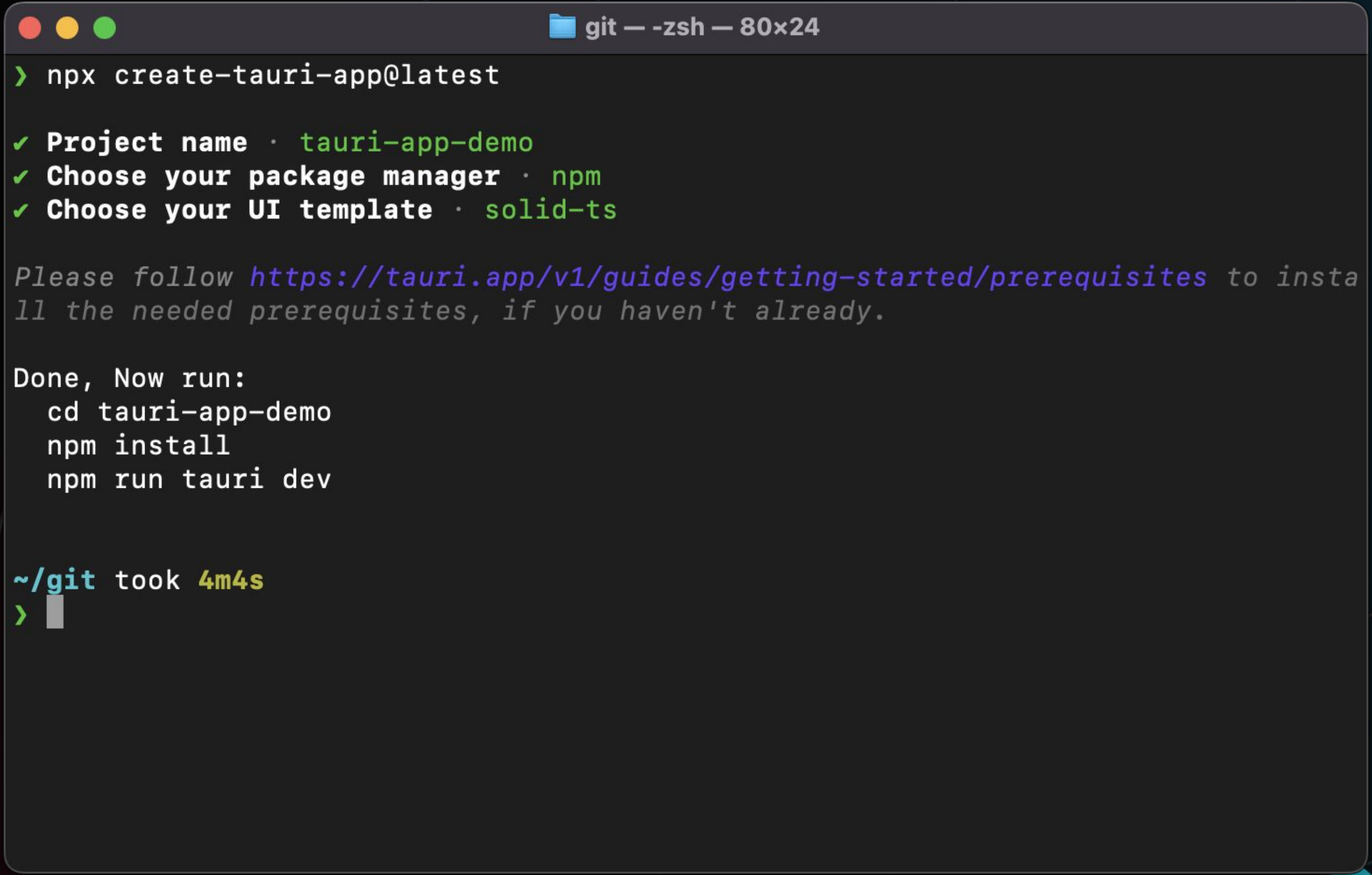
```
~/git
❯ npx create-tauri-app@latest

✔ Project name · tauri-app-demo
? Choose your package manager ›
  cargo
  pnpm
  yarn
❯ npm
```

TAURI


git — node ◂ npm exec create-tauri-app@latest __CFBundleIdentifier=com.apple.Terminal TMPDIR...

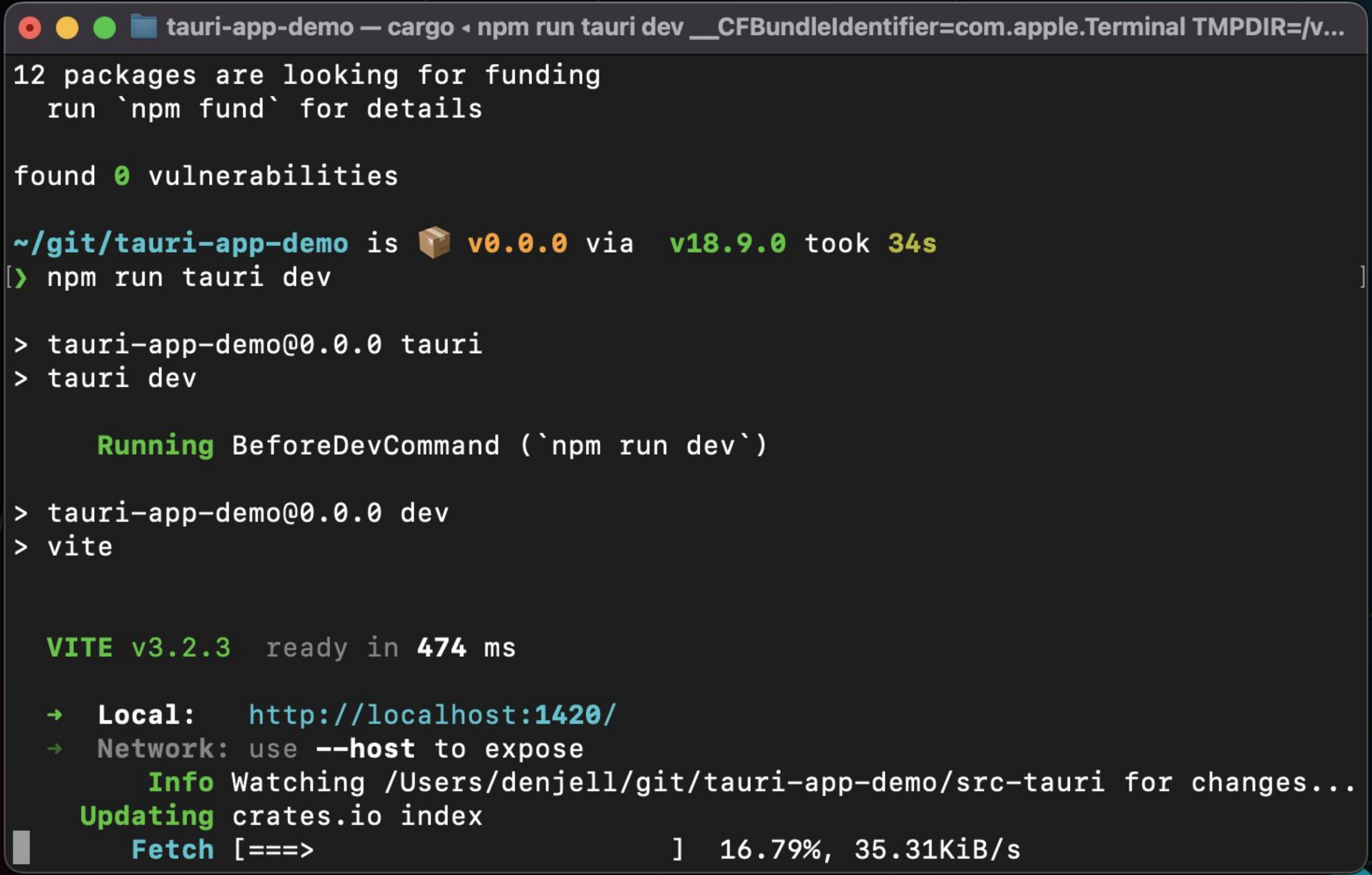
```
❯ npx create-tauri-app@latest

✔ Project name · tauri-app-demo
✔ Choose your package manager · npm
? Choose your UI template ›
  vanilla
  vanilla-ts
  vue
  vue-ts
  svelte
  svelte-ts
  react
  react-ts
  solid
❯ solid-ts
  next
  next-ts
  preact
  preact-ts
  angular
  clojurescript
  svelte-kit
  svelte-kit-ts
```

TAURI



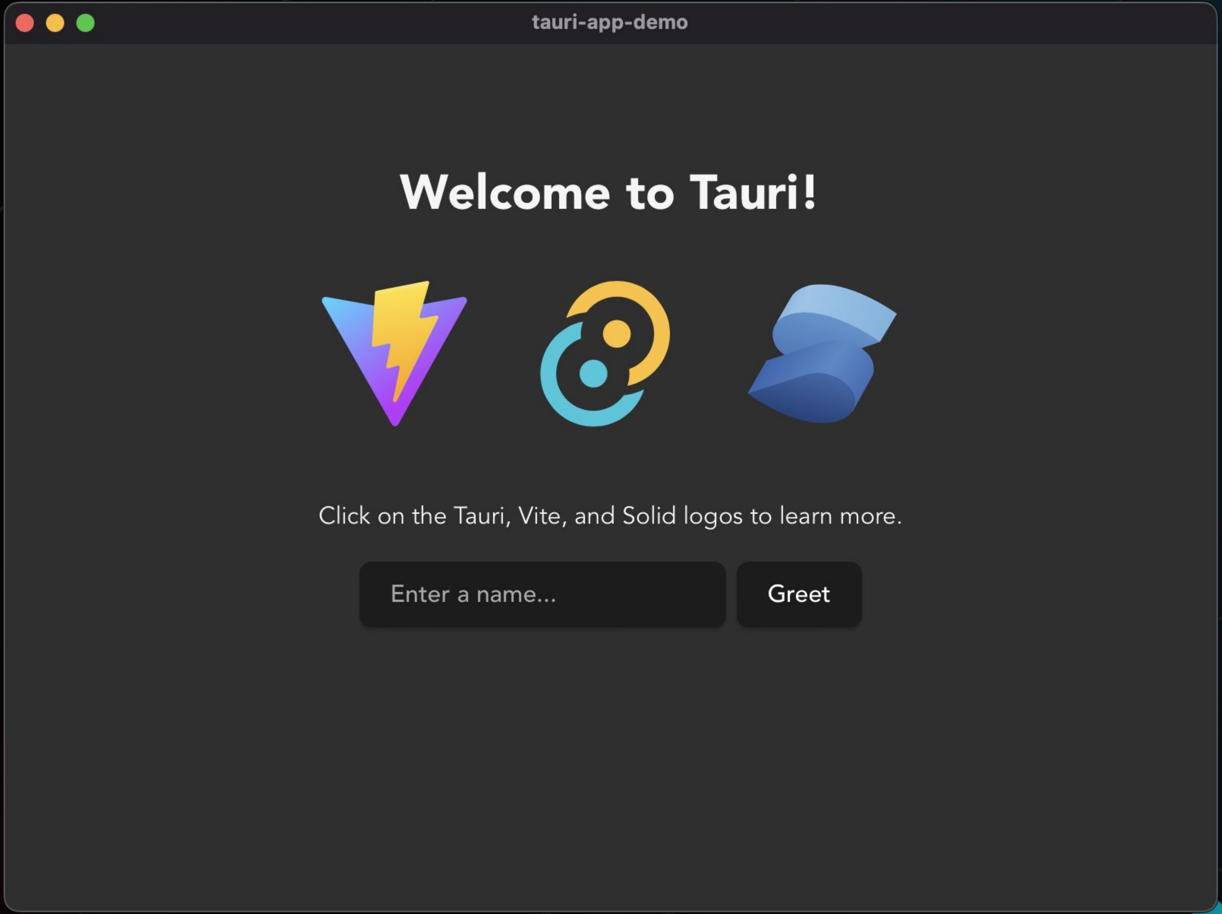
```
❯ npx create-tauri-app@latest

✔ Project name · tauri-app-demo
✔ Choose your package manager · npm
✔ Choose your UI template · solid-ts

Please follow https://tauri.app/v1/guides/getting-started/prerequisites to insta
ll the needed prerequisites, if you haven't already.

Done, Now run:
  cd tauri-app-demo
  npm install
  npm run tauri dev


~/git took 4m4s
❯ 
```

TAURI


tauri-app-demo — cargo ◂ npm run tauri dev __CFBundleIdentifier=com.apple.Terminal TMPDIR=/v...

```
12 packages are looking for funding
  run `npm fund` for details

found 0 vulnerabilities

~/git/tauri-app-demo is 📦 v0.0.0 via  v18.9.0 took 34s
❯ npm run tauri dev

> tauri-app-demo@0.0.0 tauri
> tauri dev

     Running BeforeDevCommand (`npm run dev`)

> tauri-app-demo@0.0.0 dev
> vite


  VITE v3.2.3  ready in 474 ms

  ➜  Local:   http://localhost:1420/
  ➜  Network: use --host to expose
        Info Watching /Users/denjell/git/tauri-app-demo/src-tauri for changes...
    Updating crates.io index
       Fetch [===>                    ]  16.79%, 35.31KiB/s
```

TAURI

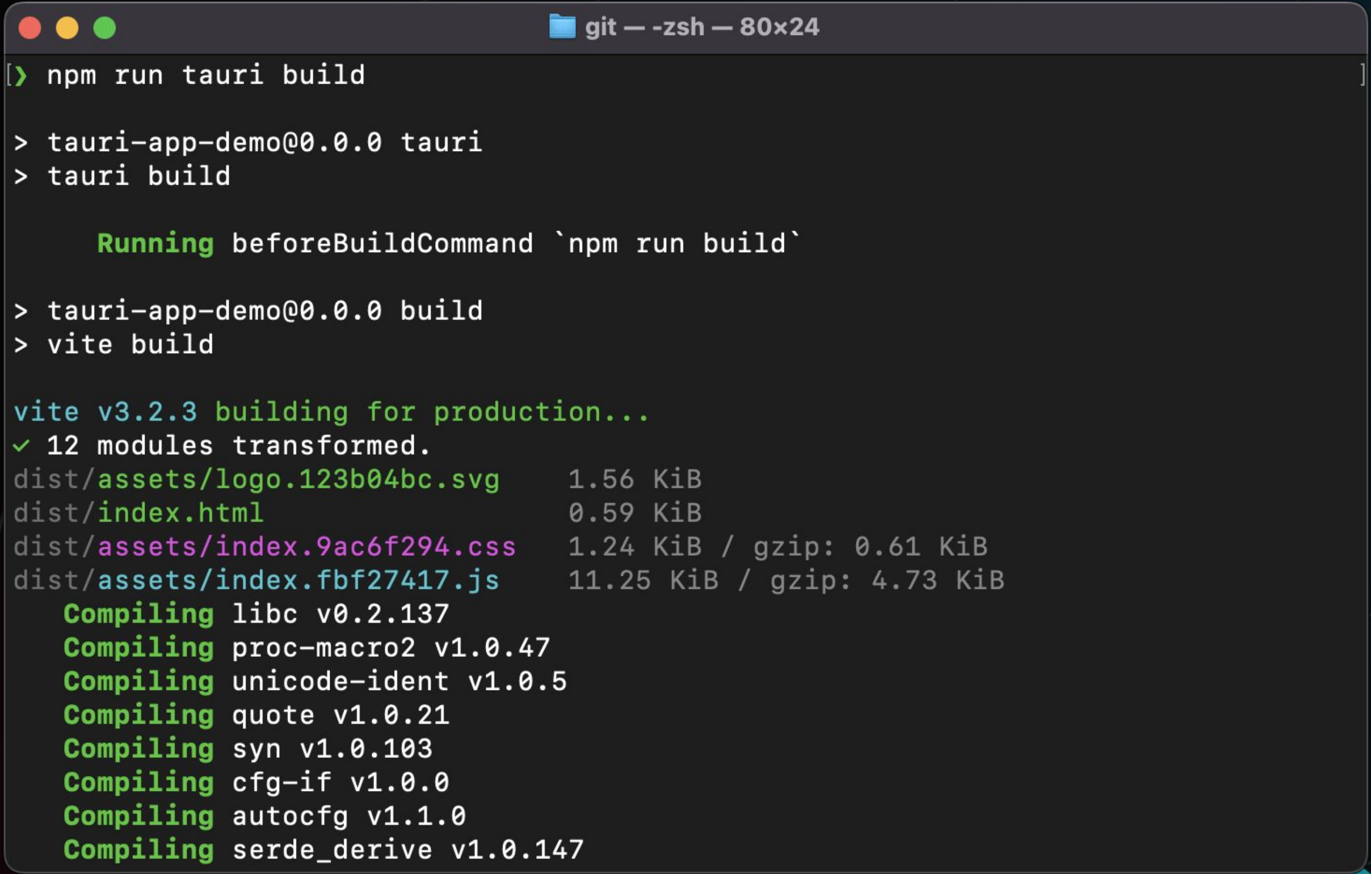


tauri-app-demo

Welcome to Tauri!

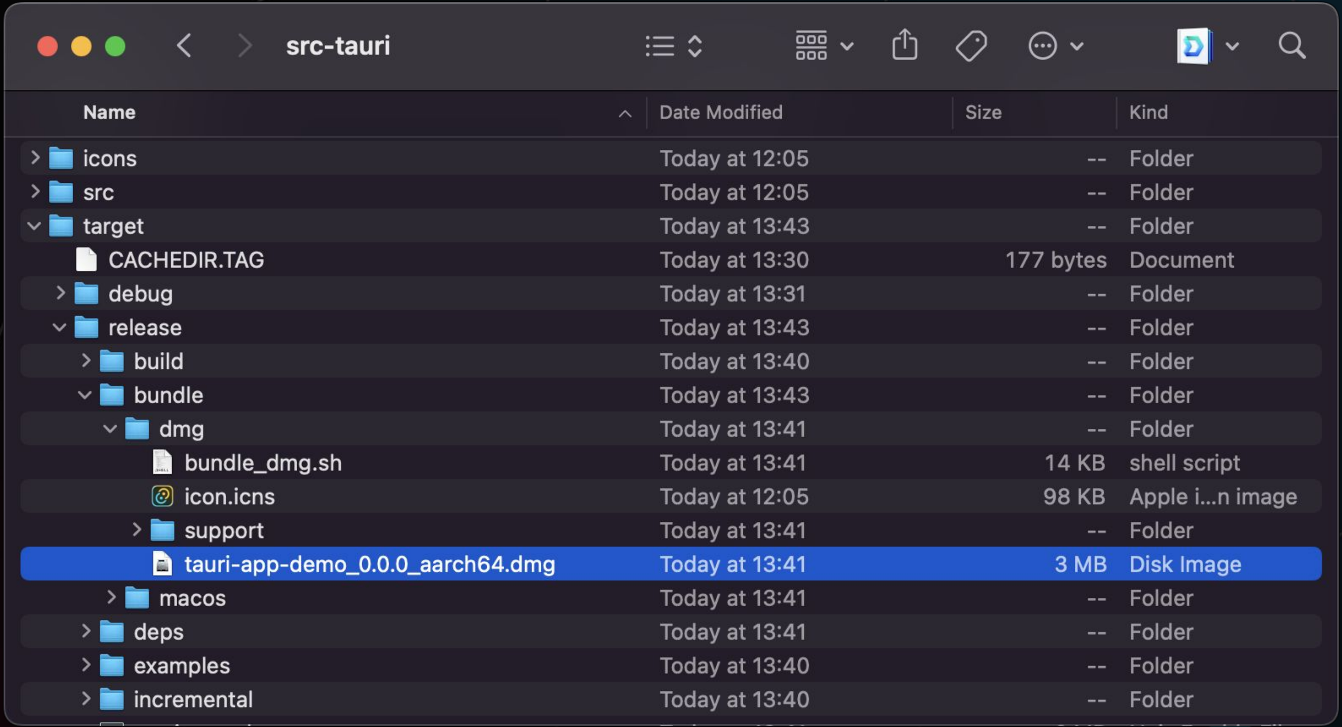
Click on the Tauri, Vite, and Solid logos to learn more.

TAURI



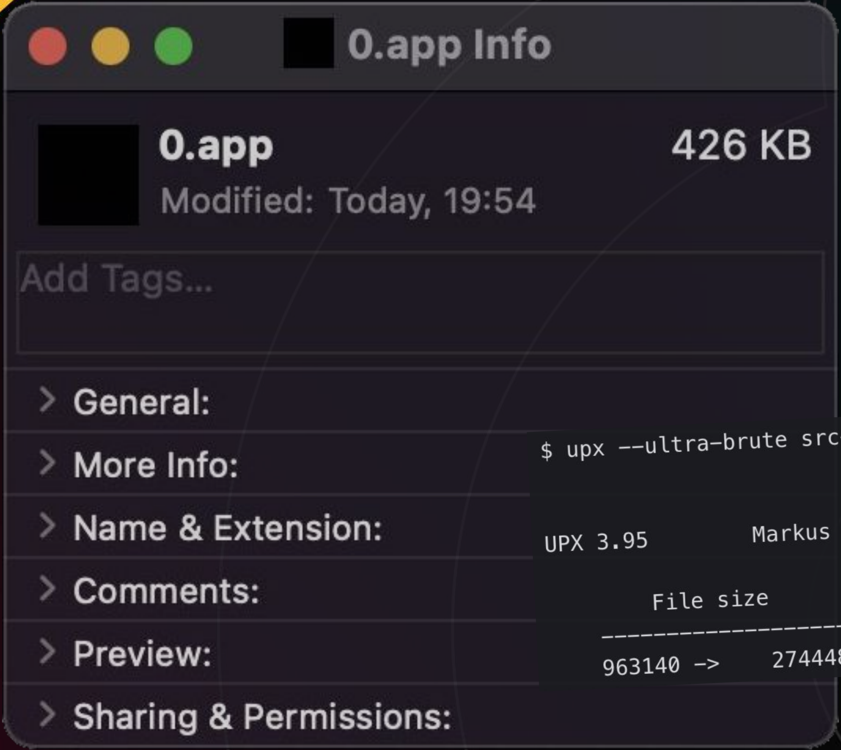
```
❯ npm run tauri build

> tauri-app-demo@0.0.0 tauri
> tauri build

     Running beforeBuildCommand `npm run build`

> tauri-app-demo@0.0.0 build
> vite build

vite v3.2.3 building for production...
✓ 12 modules transformed.
dist/assets/logo.123b04bc.svg    1.56 KiB
dist/index.html                  0.59 KiB
dist/assets/index.9ac6f294.css   1.24 KiB / gzip: 0.61 KiB
dist/assets/index.fbf27417.js    11.25 KiB / gzip: 4.73 KiB
   Compiling libc v0.2.137
   Compiling proc-macro2 v1.0.47
   Compiling unicode-ident v1.0.5
   Compiling quote v1.0.21
   Compiling syn v1.0.103
   Compiling cfg-if v1.0.0
   Compiling autocfg v1.1.0
   Compiling serde_derive v1.0.147
```

TAURI



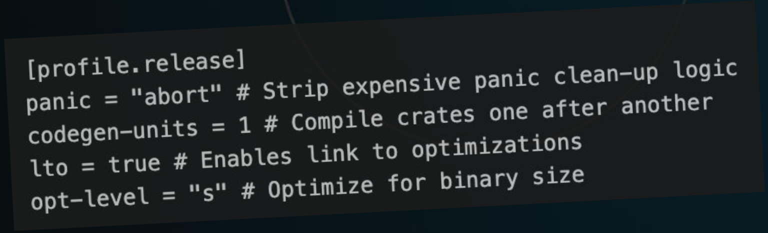
Name	Date Modified	Size	Kind
icons	Today at 12:05	--	Folder
src	Today at 12:05	--	Folder
target	Today at 13:43	--	Folder
CACHEDIR.TAG	Today at 13:30	177 bytes	Document
debug	Today at 13:31	--	Folder
release	Today at 13:43	--	Folder
build	Today at 13:40	--	Folder
bundle	Today at 13:43	--	Folder
dmg	Today at 13:41	--	Folder
bundle_dmg.sh	Today at 13:41	14 KB	shell script
icon.icns	Today at 12:05	98 KB	Apple i...n image
support	Today at 13:41	--	Folder
tauri-app-demo_0.0.0_aarch64.dmg	Today at 13:41	3 MB	Disk Image
macos	Today at 13:41	--	Folder
deps	Today at 13:41	--	Folder
examples	Today at 13:40	--	Folder
incremental	Today at 13:40	--	Folder

TAURI



0.app 426 KB

Modified: Today, 19:54

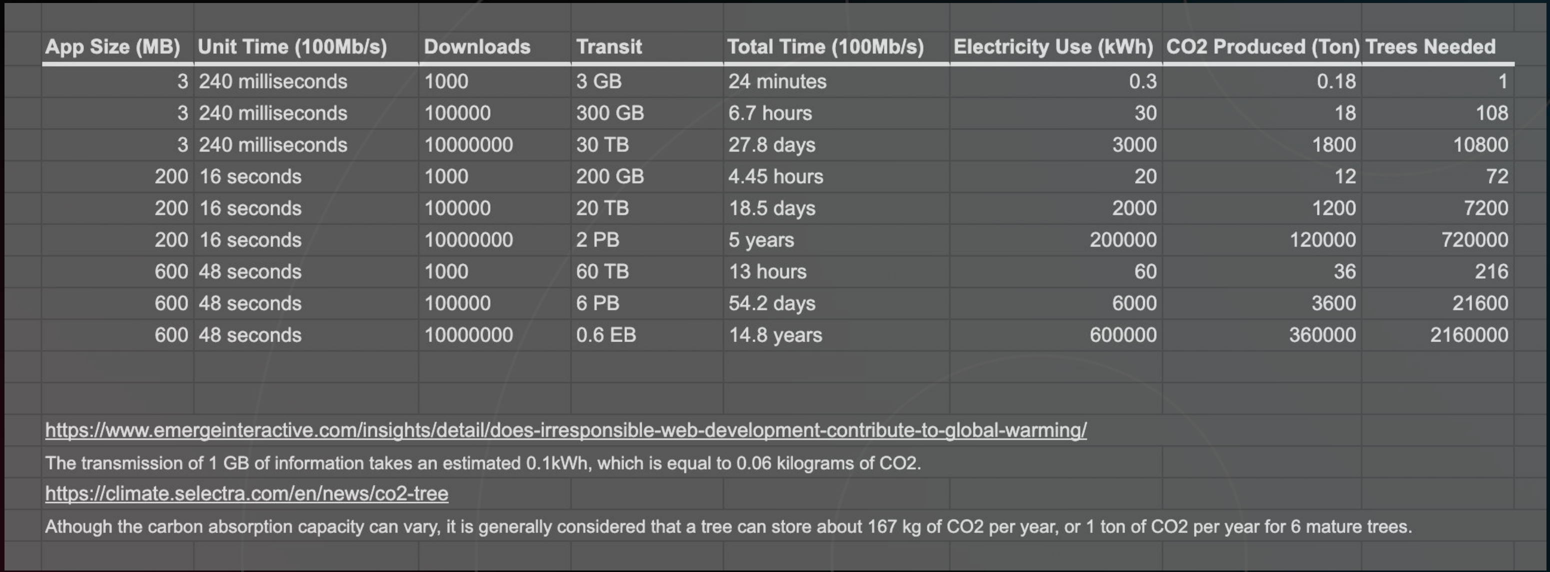


```
[profile.release]
panic = "abort" # Strip expensive panic clean-up logic
codegen-units = 1 # Compile crates one after another
lto = true # Enables link to optimizations
opt-level = "s" # Optimize for binary size
```

```
$ upx --ultra-brute src-tauri/target/release/bundle/macos/app.app/Contents/macOS/app
                       Ultimate Packer for eXecutables
                          Copyright (C) 1996 - 2018
UPX 3.95        Markus Oberhumer, Laszlo Molnar & John Reiser   Aug 26th 2018

        File size         Ratio      Format      Name
   --------------------   ------   -----------   -----------
    963140 ->    274448   28.50%   macho/amd64   app
```

TAURI

https://tauri.studio/docs/building/app-size



App Size (MB)	Unit Time (100Mb/s)	Downloads	Transit	Total Time (100Mb/s)	Electricity Use (kWh)	CO2 Produced (Ton)	Trees Needed
3	240 milliseconds	1000	3 GB	24 minutes	0.3	0.18	1
3	240 milliseconds	100000	300 GB	6.7 hours	30	18	108
3	240 milliseconds	10000000	30 TB	27.8 days	3000	1800	10800
200	16 seconds	1000	200 GB	4.45 hours	20	12	72
200	16 seconds	100000	20 TB	18.5 days	2000	1200	7200
200	16 seconds	10000000	2 PB	5 years	200000	120000	720000
600	48 seconds	1000	60 TB	13 hours	60	36	216
600	48 seconds	100000	6 PB	54.2 days	6000	3600	21600
600	48 seconds	10000000	0.6 EB	14.8 years	600000	360000	2160000

https://www.emergeinteractive.com/insights/detail/does-irresponsible-web-development-contribute-to-global-warming/

The transmission of 1 GB of information takes an estimated 0.1kWh, which is equal to 0.06 kilograms of CO2.

https://climate.selectra.com/en/news/co2-tree

Athough the carbon absorption capacity can vary, it is generally considered that a tree can store about 167 kg of CO2 per year, or 1 ton of CO2 per year for 6 mature trees.

Security

It’s not just a posture.

TAURI

Tauri Foundations
A practical introduction to the Tauri framework and adopting a security-first approach to app development
Daniel Thompson-Yvetot
Lucas Nogueira
TAURI
Tauri Foundations
From the founders of Tauri:
This, the first book in the educational series about Tauri Apps, is a practical introduction to the philosophy and architecture underpinning the development of next-generation, secure apps.
Packt

TAURI

https://www.packtpub.com/product/tauri-foundations/9781803241654

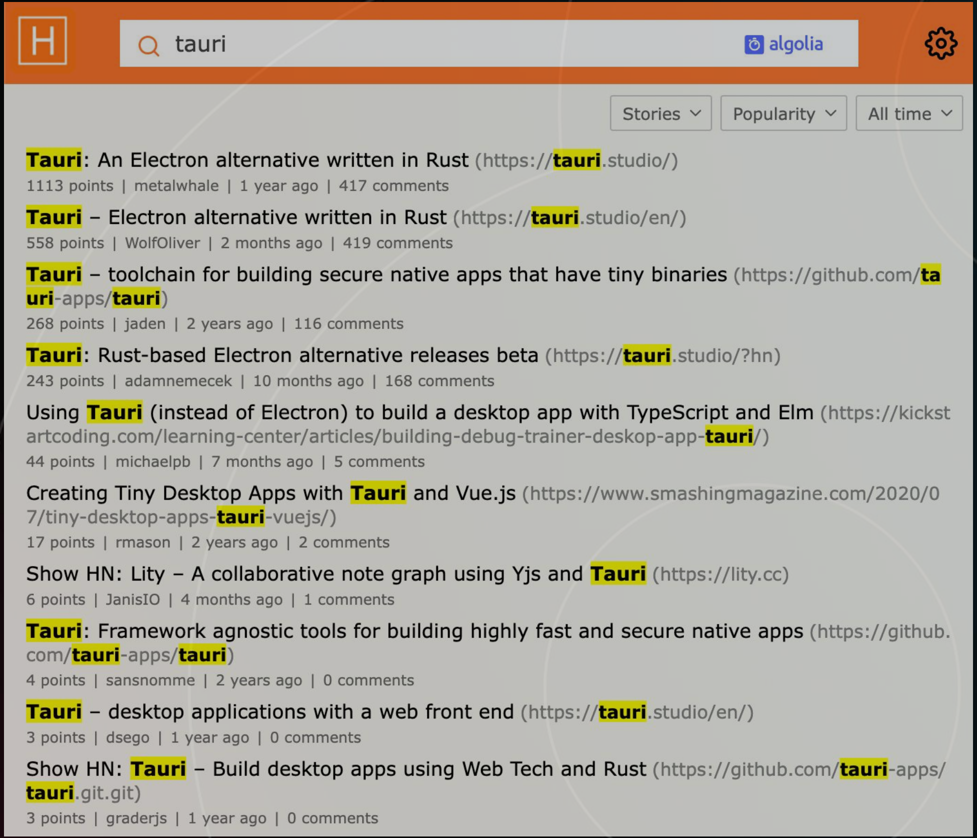
TAURI 1.0

TAURI

TAURI 1.0

Finally.

TAURI



Tauri: An Electron alternative written in Rust (https://**tauri**.studio/)
1113 points | metalwhale | 1 year ago | 417 comments

Tauri – Electron alternative written in Rust (https://**tauri**.studio/en/)
558 points | WolfOliver | 2 months ago | 419 comments

Tauri – toolchain for building secure native apps that have tiny binaries (https://github.com/**tauri**-apps/**tauri**)
268 points | jaden | 2 years ago | 116 comments

Tauri: Rust-based Electron alternative releases beta (https://**tauri**.studio/?hn)
243 points | adamnemecek | 10 months ago | 168 comments

Using **Tauri** (instead of Electron) to build a desktop app with TypeScript and Elm (https://kickstartcoding.com/learning-center/articles/building-debug-trainer-deskop-app-**tauri**/)
44 points | michaelpb | 7 months ago | 5 comments

Creating Tiny Desktop Apps with **Tauri** and Vue.js (https://www.smashingmagazine.com/2020/07/tiny-desktop-apps-**tauri**-vuejs/)
17 points | rmason | 2 years ago | 2 comments

Show HN: Lity – A collaborative note graph using Yjs and **Tauri** (https://lity.cc)
6 points | JanisIO | 4 months ago | 1 comments

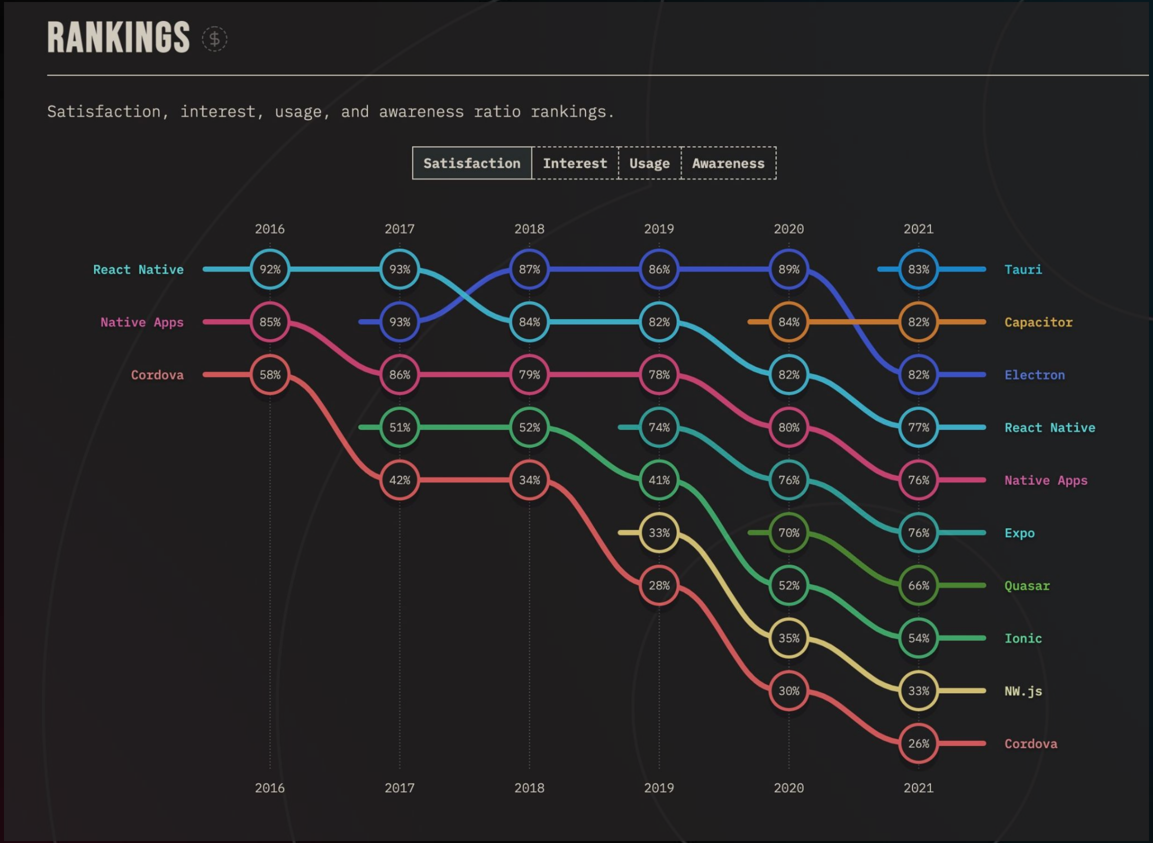
Tauri: Framework agnostic tools for building highly fast and secure native apps (https://github.com/**tauri**-apps/**tauri**)
4 points | sansnomme | 2 years ago | 0 comments

Tauri – desktop applications with a web front end (https://**tauri**.studio/en/)
3 points | dsego | 1 year ago | 0 comments

Show HN: **Tauri** – Build desktop apps using Web Tech and Rust (https://github.com/**tauri**-apps/**tauri**.git.git)
3 points | graderjs | 1 year ago | 0 comments

TAURI

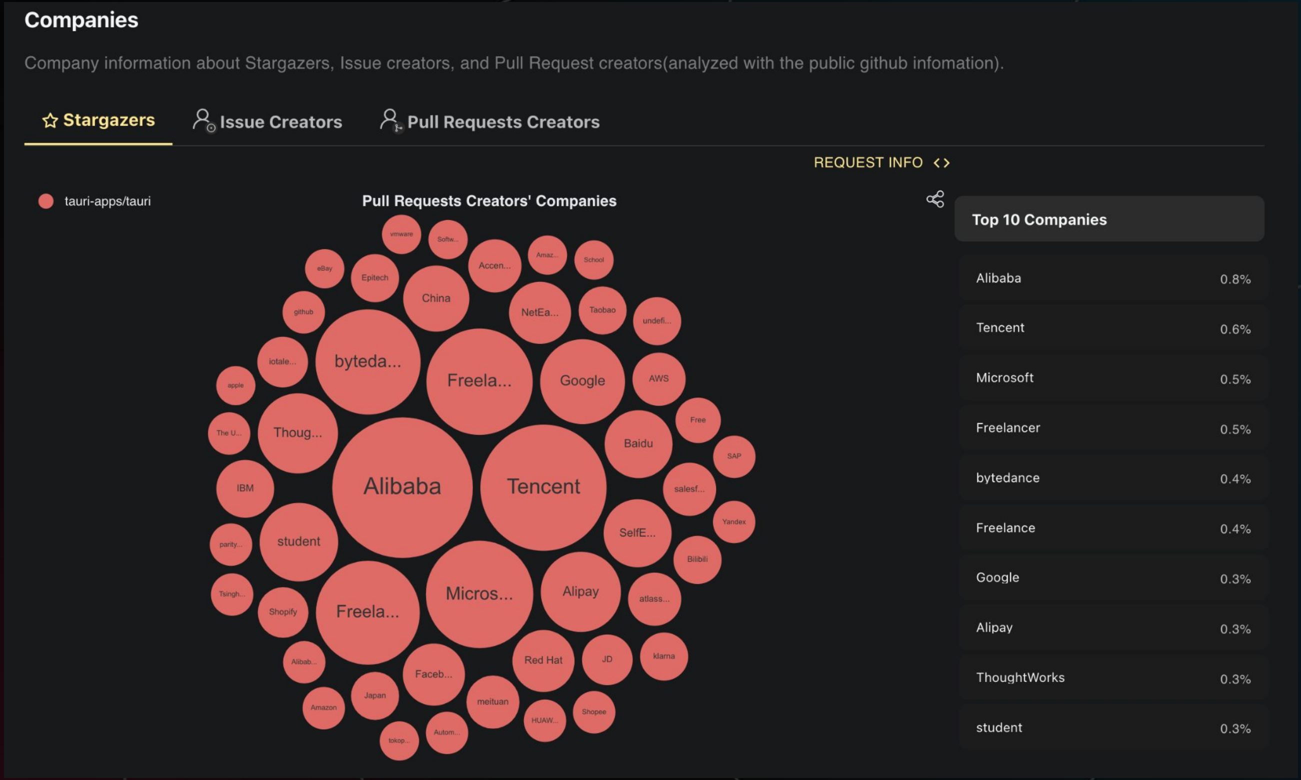
https://hn.algolia.com/?q=tauri



RANKINGS

Satisfaction, interest, usage, and awareness ratio rankings.

https://2021.stateofjs.com/en-US/libraries/mobile-desktop



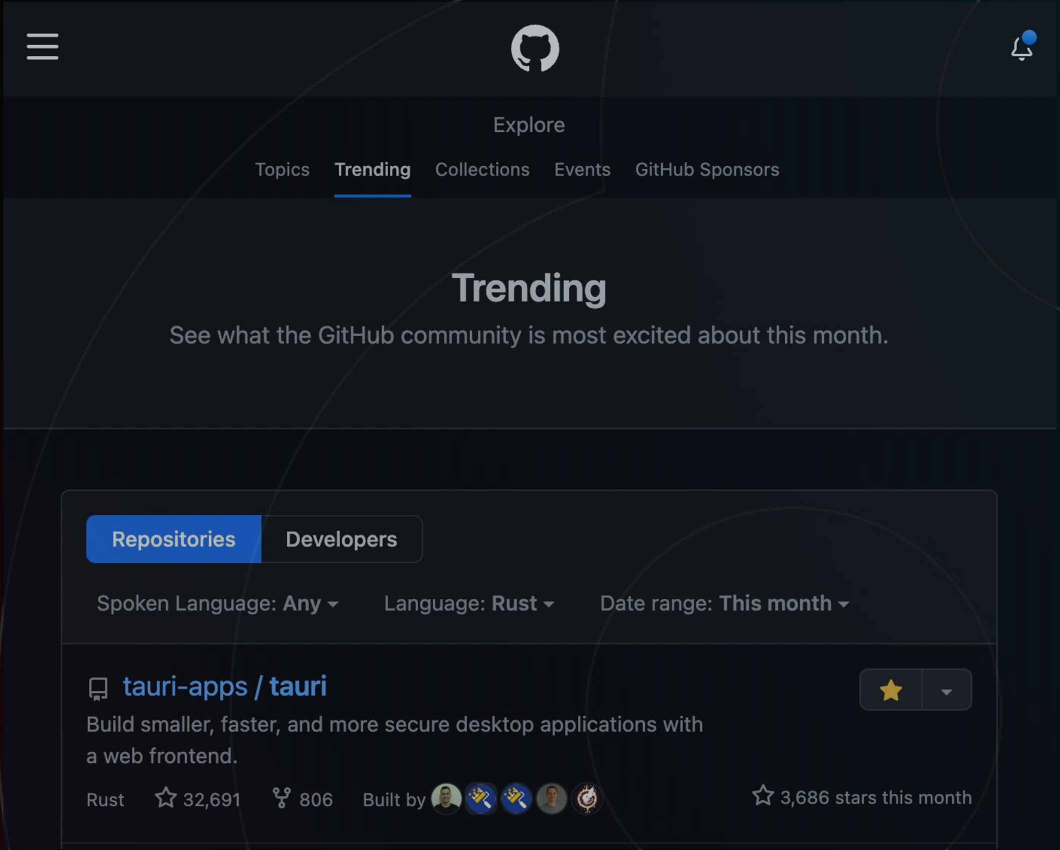
Companies

Company information about Stargazers, Issue creators, and Pull Request creators(analyzed with the public github infomation).

Top 10 Companies

Alibaba	0.8%
Tencent	0.6%
Microsoft	0.5%
Freelancer	0.5%
bytedance	0.4%
Freelance	0.4%
Google	0.3%
Alipay	0.3%
ThoughtWorks	0.3%
student	0.3%

https://ossinsight.io/analyze/tauri-apps/tauri



Trending

See what the GitHub community is most excited about this month.

Spoken Language: **Any**
Language: **Rust**
Date range: **This month**

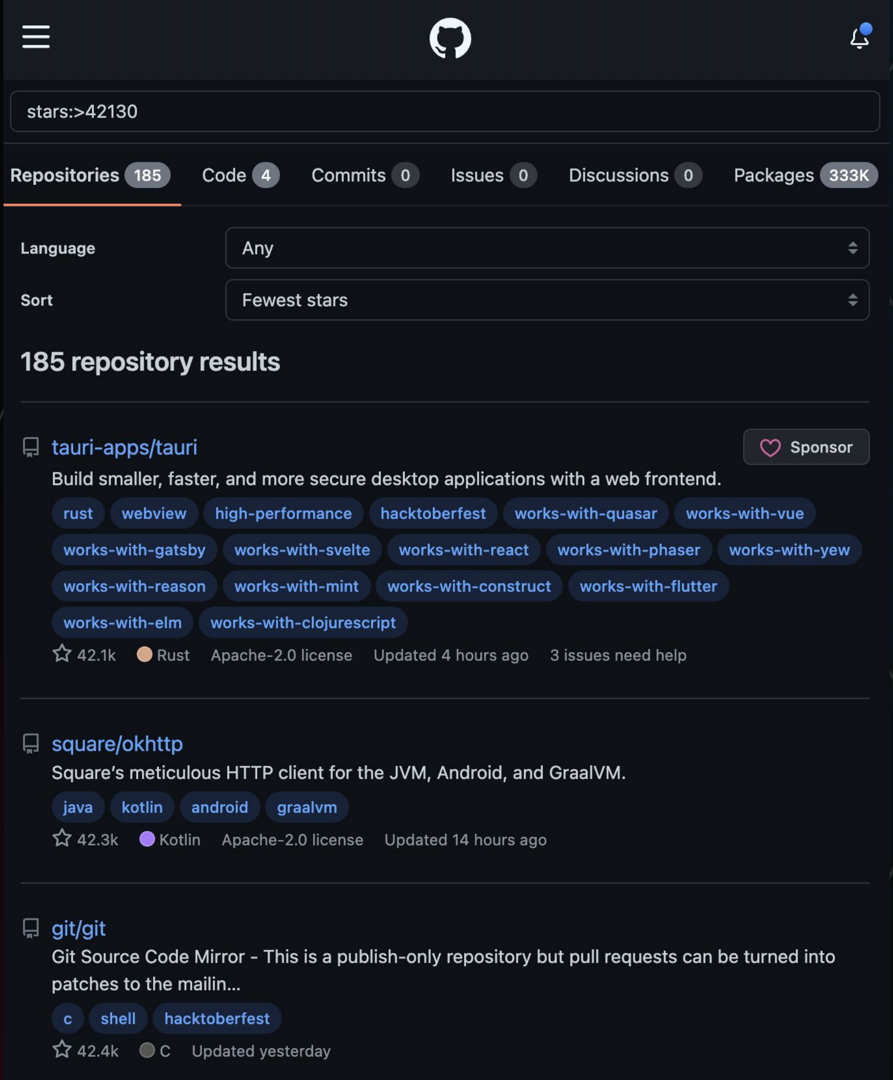
tauri-apps / **tauri**

Build smaller, faster, and more secure desktop applications with a web frontend.

Rust ☆ 32,691 806 Built by ☆ 3,686 stars this month

TAURI

https://github.com/trending/rust?since=monthly



stars:>42130

Repositories 185 | Code 4 | Commits 0 | Issues 0 | Discussions 0 | Packages 333K

Language: Any

Sort: Fewest stars

185 repository results

tauri-apps/tauri

Sponsor

Build smaller, faster, and more secure desktop applications with a web frontend.

rust webview high-performance hacktoberfest works-with-quasar works-with-vue works-with-gatsby works-with-svelte works-with-react works-with-phaser works-with-yew works-with-reason works-with-mint works-with-construct works-with-flutter works-with-elm works-with-clojurescript

☆ 42.1k ● Rust Apache-2.0 license Updated 4 hours ago 3 issues need help

square/okhttp

Square's meticulous HTTP client for the JVM, Android, and GraalVM.

java kotlin android graalvm

☆ 42.3k ● Kotlin Apache-2.0 license Updated 14 hours ago

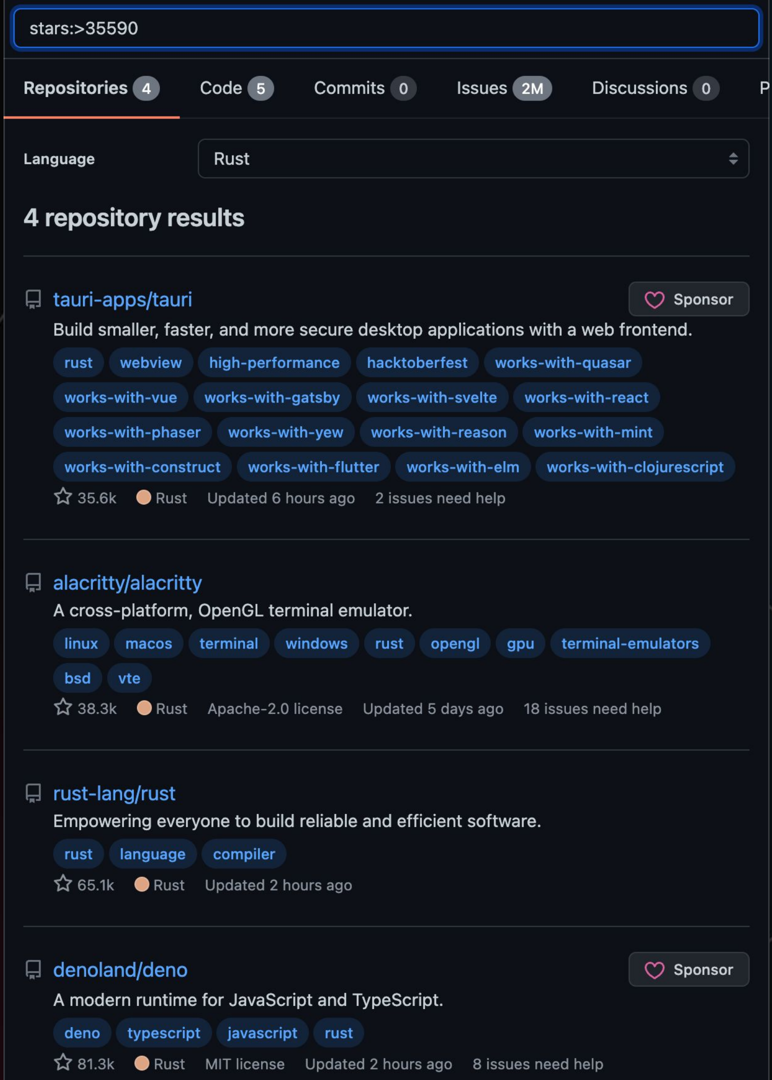
git/git

Git Source Code Mirror - This is a publish-only repository but pull requests can be turned into patches to the mailin...

c shell hacktoberfest

☆ 42.4k ● C Updated yesterday

TAURI



stars:>35590

Language: Rust

4 repository results

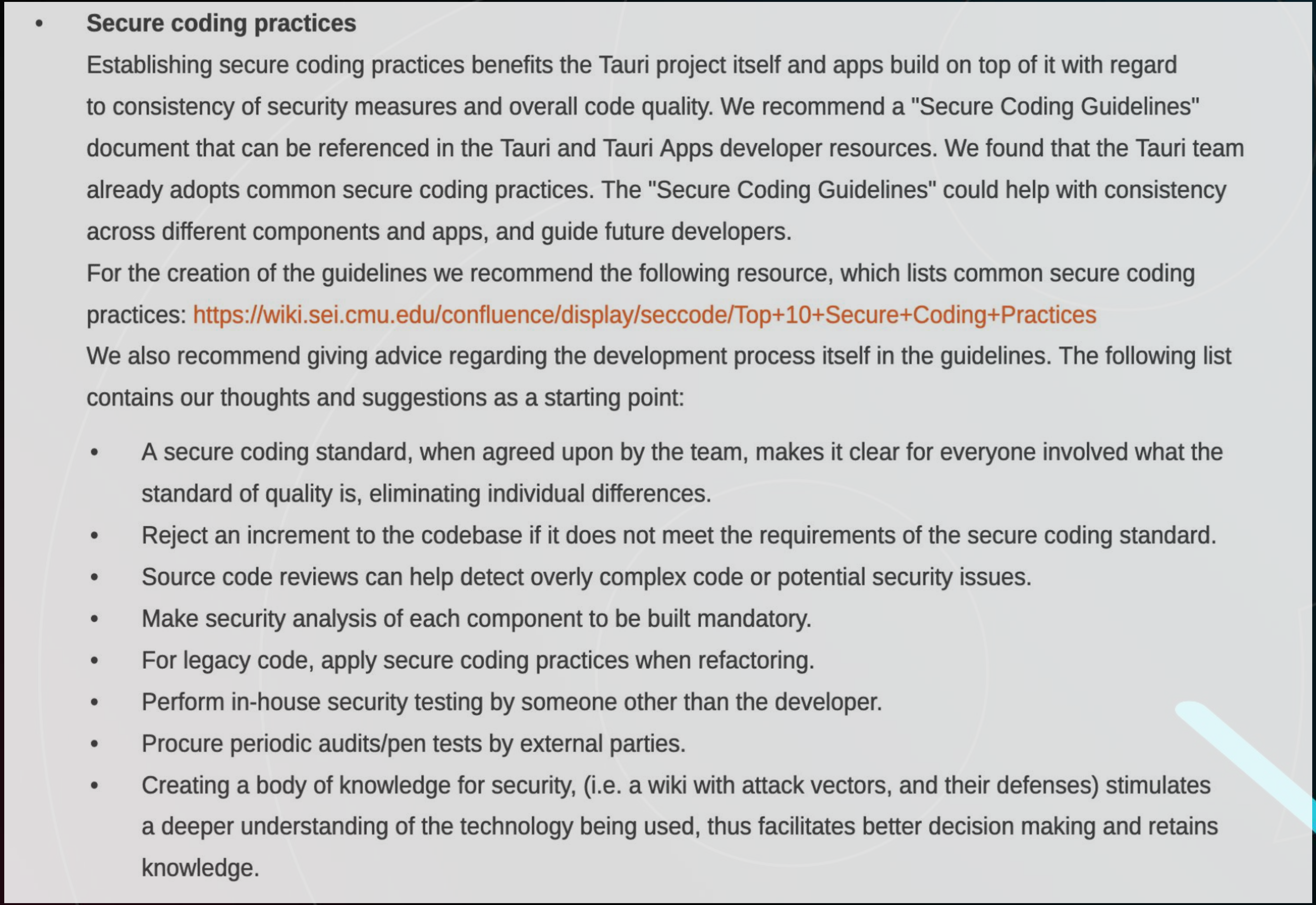
tauri-apps/tauri
Build smaller, faster, and more secure desktop applications with a web frontend.
rust · webview · high-performance · hacktoberfest · works-with-quasar · works-with-vue · works-with-gatsby · works-with-svelte · works-with-react · works-with-phaser · works-with-yew · works-with-reason · works-with-mint · works-with-construct · works-with-flutter · works-with-elm · works-with-clojurescript
☆ 35.6k · Rust · Updated 6 hours ago · 2 issues need help

alacritty/alacritty
A cross-platform, OpenGL terminal emulator.
linux · macos · terminal · windows · rust · opengl · gpu · terminal-emulators · bsd · vte
☆ 38.3k · Rust · Apache-2.0 license · Updated 5 days ago · 18 issues need help

rust-lang/rust
Empowering everyone to build reliable and efficient software.
rust · language · compiler
☆ 65.1k · Rust · Updated 2 hours ago

denoland/deno
A modern runtime for JavaScript and TypeScript.
deno · typescript · javascript · rust
☆ 81.3k · Rust · MIT license · Updated 2 hours ago · 8 issues need help

TAURI



- **Secure coding practices**

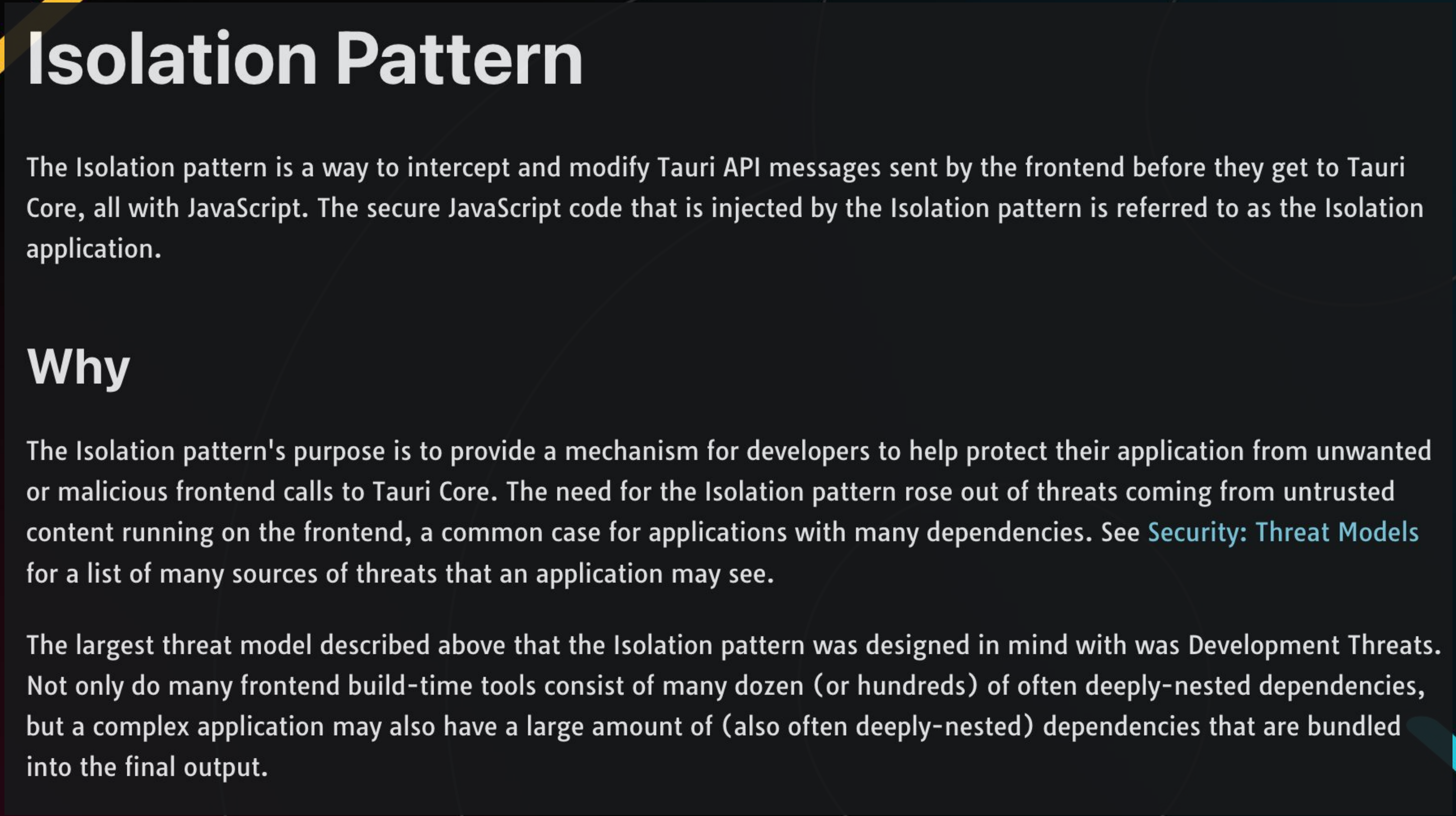
  Establishing secure coding practices benefits the Tauri project itself and apps build on top of it with regard to consistency of security measures and overall code quality. We recommend a "Secure Coding Guidelines" document that can be referenced in the Tauri and Tauri Apps developer resources. We found that the Tauri team already adopts common secure coding practices. The "Secure Coding Guidelines" could help with consistency across different components and apps, and guide future developers.

  For the creation of the guidelines we recommend the following resource, which lists common secure coding practices: https://wiki.sei.cmu.edu/confluence/display/seccode/Top+10+Secure+Coding+Practices

  We also recommend giving advice regarding the development process itself in the guidelines. The following list contains our thoughts and suggestions as a starting point:

  - A secure coding standard, when agreed upon by the team, makes it clear for everyone involved what the standard of quality is, eliminating individual differences.
  - Reject an increment to the codebase if it does not meet the requirements of the secure coding standard.
  - Source code reviews can help detect overly complex code or potential security issues.
  - Make security analysis of each component to be built mandatory.
  - For legacy code, apply secure coding practices when refactoring.
  - Perform in-house security testing by someone other than the developer.
  - Procure periodic audits/pen tests by external parties.
  - Creating a body of knowledge for security, (i.e. a wiki with attack vectors, and their defenses) stimulates a deeper understanding of the technology being used, thus facilitates better decision making and retains knowledge.

TAURI



Isolation Pattern

The Isolation pattern is a way to intercept and modify Tauri API messages sent by the frontend before they get to Tauri Core, all with JavaScript. The secure JavaScript code that is injected by the Isolation pattern is referred to as the Isolation application.

Why

The Isolation pattern's purpose is to provide a mechanism for developers to help protect their application from unwanted or malicious frontend calls to Tauri Core. The need for the Isolation pattern rose out of threats coming from untrusted content running on the frontend, a common case for applications with many dependencies. See Security: Threat Models for a list of many sources of threats that an application may see.

The largest threat model described above that the Isolation pattern was designed in mind with was Development Threats. Not only do many frontend build-time tools consist of many dozen (or hundreds) of often deeply-nested dependencies, but a complex application may also have a large amount of (also often deeply-nested) dependencies that are bundled into the final output.

TAURI

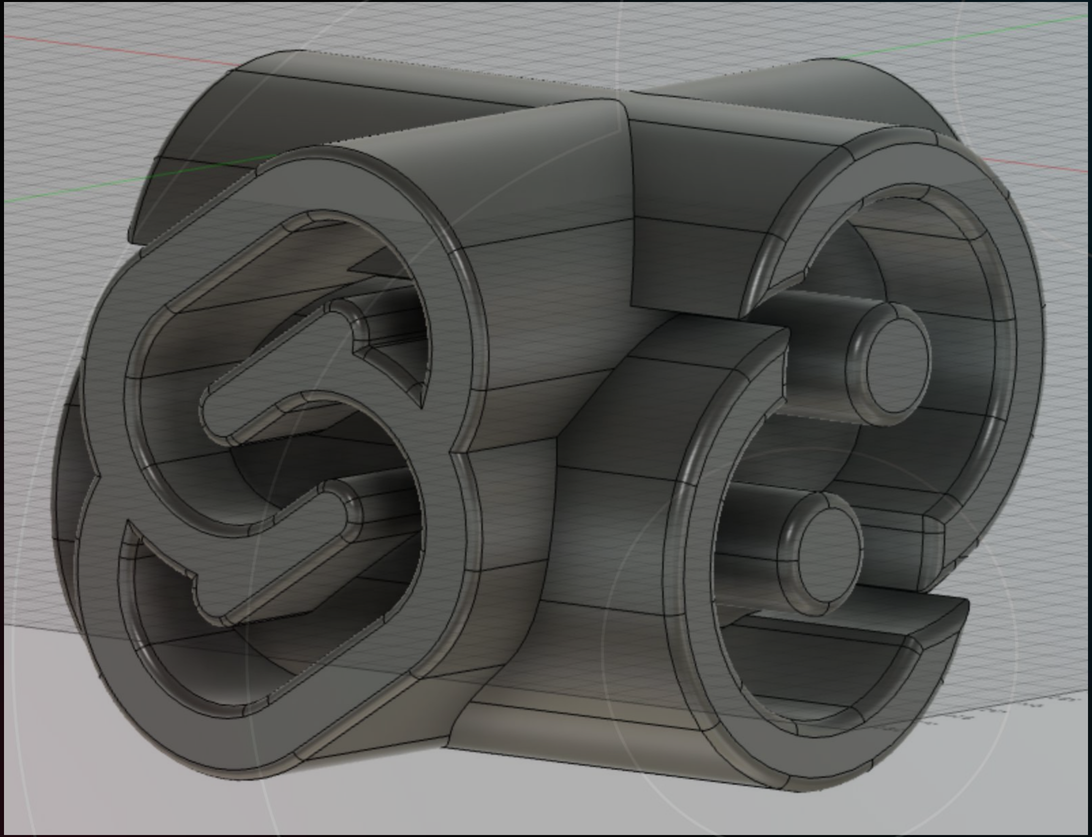
https://tauri.studio/docs/architecture/patterns/isolation

2022's Platform of Choice

TAURI

2022’s Platform of Choice

SVELTE



TAURI

TAURI