







a

Anthony Fu

Core team member of  Vite  Vue and  Nuxt





Creator of  Vitest  Slidev  UnoCSS  Type Challenges  Elk

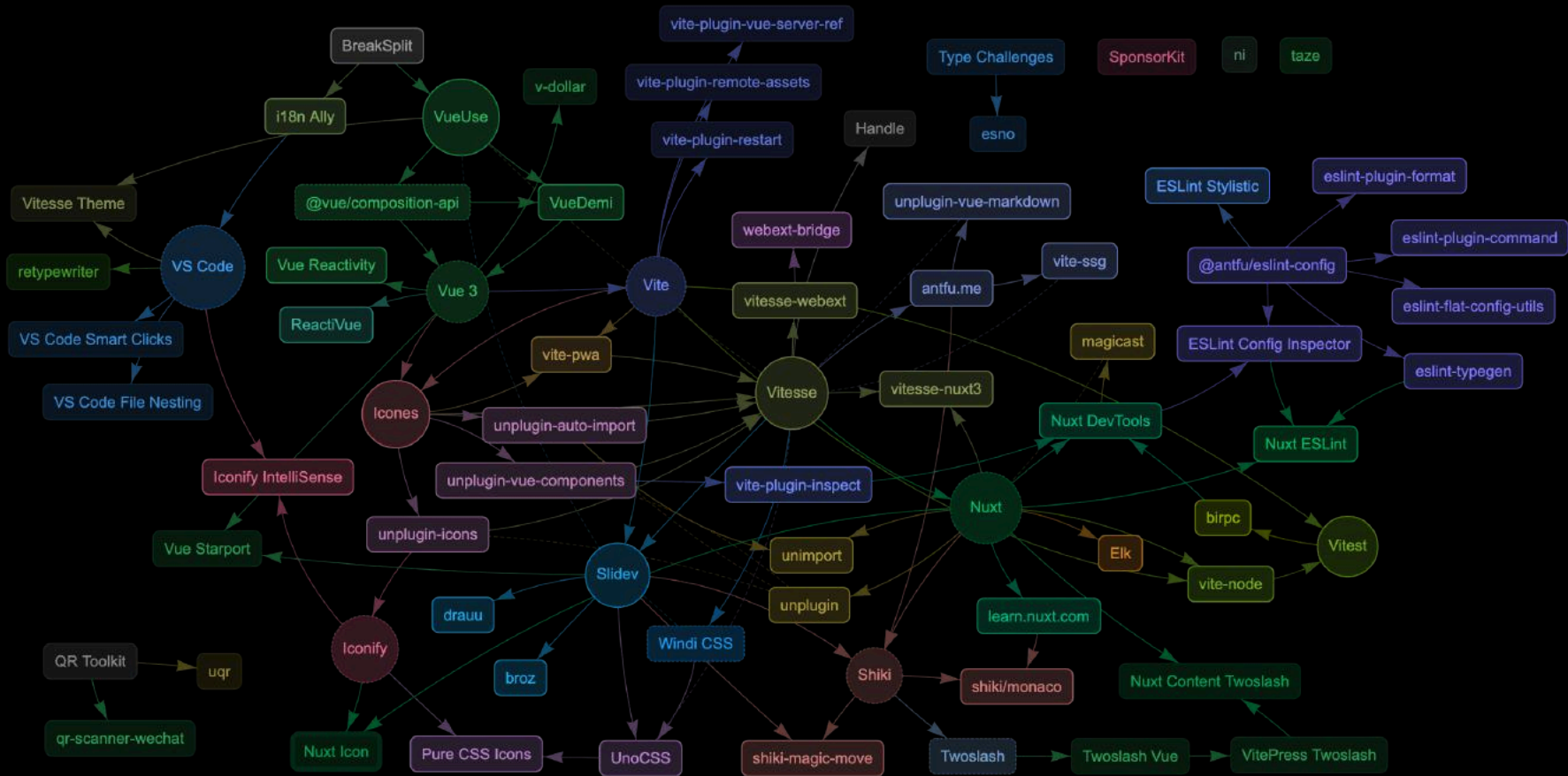
Maintainer of  ESLint Stylistic  Shiki  Twoslash

Working at  NuxtLabs /  Vercel

Base in  Tokyo



 antfu.me  antfu  antfu@webtoo.ls  antfu7



Make Tools People Love

From Anthony's Experience on Open Source

 SquiggleConf

Sep. 18th 2025

Tip 1

I Don't Know

We can't Know without Trying

Try & Error

Lower Your Expectations

Verify with the Community

Tip 2

Solve Your Own Problems

Build from your own pain points. Eat your own dogfood.

Build from your own pain points.

It can be a great way to come up with ideas and provide strong motivation

Demand

Start from your own needs, with a deeper understanding of the problem

Motivation

Solve your own problems, and maybe help others

Validation

Your needs are likely to be others' needs, validate your ideas through the community

Iterating

Refine the design with discussion and contributions from community




Tip 3


Progressive!

Reduce the friction.


Progressive On

 **Onboarding** Making it easy to understand and get started

Describe your tool clearly within one sentence

 **Features** Starts minimal and grows with users

Make your tool useful even in the simplest case

 **Extensible** Allows community to enhance the tool and the ecosystem

Make a plugin system for your tool

 **Breaking Changes** Provide middle stages for migration

Deprecate & Warning → Forward/Backward Compatibility → Remove

Tip 4

Extensible

Plugins system. Community driven.

All Software

should provide a Plugin System

Scripts vs Software

One-off

Tools

"Software is logic that adapts"

Benefits

of Having a Plugins System

- Allow community to build features without being blocked.
- Verify ideas without breaking changes to core.
- Better architecture.
- Maximizing the value of being a tool.

Tip 4.1

Make Core Feature Plugins

To verify you have a powerful plugin API.



Vite v0.x

Opinionated Dev Server for Vue



Vite v2.0+

Next Generation Frontend Tooling

Vue as a plugin

React as a plugin

Svelte as a plugin

Test runner for Vite

Meta-frameworks...

...

Tip 5

Work with the Community

Work with people, get them engaged, build together.



Build Your Own Identify

People need to know who is behind the project.



Work with Your Users

Listen to their feedback and iterate.



Maintains Good Feedback-loop for Contributors

Reply and review the PR in time. Keep them engaged. Build your team.



Collaborate with Other Projects/Authors/Maintainers

Extend the ecosystem by introducing new integrations.

Recap

Tip #1

I Don't Know

Try and Error; Verify with the users.

Tip #3

Progressive!

Reduce the friction.

Tip #5

Work with the Community

Work with people, get them engaged, build together.

Tip #2

Solve your own Problems

Build from your pain points. Eat your own dogfood.

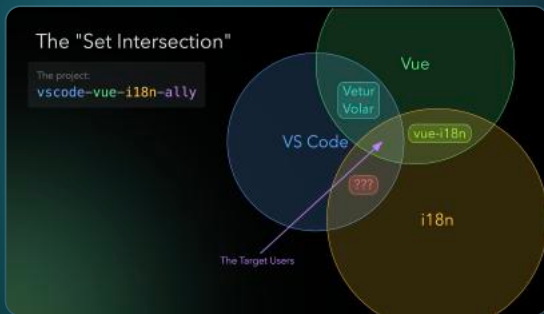
Tip #4

Extensible

Plugins system. Community driven.

Anthony's Road to Open Source

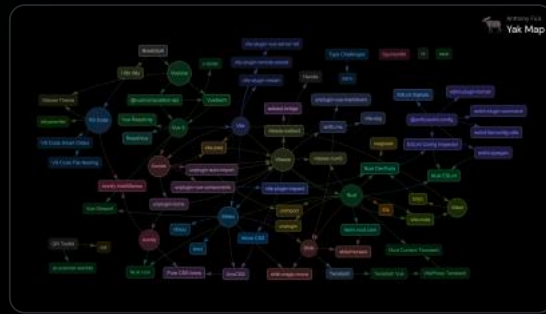
Part I The Set Theory



Part II The Progressive Path



Part III Yak Shaving



Recordings on antfu.me

To Make Tools that People Love
is to Put Love into the Tools We Make ❤️

Thank You!

Slides can be found on antfu.me