

Porque Viteest?

Vitest é uma estrutura de teste de unidade extremamente rápida, desenvolvida pela Vite. O Vitest visa se posicionar como o Test Runner de escolha para projetos Vite e como uma alternativa sólida mesmo para projetos que não usam Vite. O uso do Vite dev server para transformar seus arquivos durante o teste permite a criação de um executor simples que não precisa lidar com a complexidade de transformar arquivos de origem e pode se concentrar apenas em fornecer o melhor DX durante o teste. Um executor de teste que usa a mesma configuração do seu aplicativo (por meio de vite.config.js), compartilhando um pipeline de transformação comum durante o tempo de desenvolvimento, compilação e teste.





Por que Vitest ?

Vitest é uma estrutura de teste de unidade extremamente rápida, desenvolvida pela Vite. O Vitest visa se posicionar como o Test Runner de escolha para projetos Vite e como uma alternativa sólida mesmo para projetos que não usam Vite. O uso do Vite dev server para transformar seus arquivos durante o teste permite a criação de um executor simples que não precisa lidar com a complexidade de transformar arquivos de origem e pode se concentrar apenas em fornecer o melhor DX durante o teste. Um executor de teste que usa a mesma configuração do seu aplicativo (por meio de vite.config.js), compartilhando um pipeline de transformação comum durante o tempo de desenvolvimento, compilação e teste.



Features

- ✓ Vite's config, transformers, resolvers, and plugins.
- ✓ Use the same setup from your app to run the tests!
- ✓ Smart & instant watch mode, like HMR for tests!
- ✓ Component testing for Vue, React, Svelte, Lit and more
- ✓ Out-of-the-box TypeScript / JSX support
- ✓ ESM first, top level await
- ✓ Workers multi-threading via **Tinypool**
- ✓ Benchmarking support with **Tinybench**
- ✓ Filtering, timeouts, concurrent for suite and tests
- ✓ Jest-compatible Snapshot
- ✓ Chai built-in for assertions + Jest expect compatible APIs
- ✓ **Tinyspy** built-in for mocking
- ✓ **happy-dom** or **jsdom** for DOM mocking
- ✓ Code coverage via **c8** or **istanbul**
- ✓ Rust-like **in-source testing**