

Tauri with Vue 3 Project Structure

Tauri with Vue 3 Composition API and Pinia. Familiar Vue patterns for desktop development.

#tauri #rust #desktop #vue #pinia #vite

PNG

PDF

Copy

</> Prompt

Project Directory

my-tauri-app/

- package.json
- vite.config.ts
- tsconfig.json
- index.html
- > **src/** Vue 3 app
 - main.ts App bootstrap
 - App.vue Root component
 - style.css
 - > **components/**
 - TitleBar.vue
 - Sidebar.vue
 - FileList.vue
 - > **views/** Route views
 - HomeView.vue
 - SettingsView.vue
 - > **stores/** Pinia stores
 - app.ts
 - settings.ts
 - > **composables/** Vue composables
 - useTauri.ts Tauri commands
 - useWindow.ts Window controls
 - > **router/**
 - index.ts
 - public/**
 - > **src-tauri/**
 - Cargo.toml
 - tauri.conf.json
 - build.rs
 - > **src/**
 - main.rs
 - lib.rs
 - > **commands/**
 - mod.rs
 - app.rs
 - files.rs
 - > **state/**
 - mod.rs
 - app_state.rs
 - icons/**
 - capabilities/**

Why This Structure?

Vue 3 with Composition API brings reactive, organized code to Tauri. Pinia manages state with devtools support. Vue composables wrap Tauri commands for clean component code. Vue Router handles in-app navigation.

Key Directories

- src/composables/** - Reusable logic with Composition API
- src/stores/** - Pinia stores for global state
- src/views/** - Route-level components
- src/components/** - Reusable UI components

Getting Started

- npm create tauri-app@latest -- --template vue-ts
- cd my-tauri-app && npm install
- npm install pinia vue-router
- npm run tauri dev

Vue Composable

```
// src/composables/useTauri.ts
import { invoke } from '@tauri-apps/api/core';

export function useTauri() {
  const readFile = (path: string) =>
    invoke<string>('read_file', { path });

  const saveFile = (path: string, content: string) =>
    invoke('save_file', { path, content });

  return { readFile, saveFile };
}
```

When To Use This

- Teams with Vue experience
- Apps needing robust state management
- Complex UIs with many components
- Projects wanting strong devtools support

Trade-offs

- Bundle size** - Vue runtime larger than Svelte
- Setup complexity** - Router and Pinia need configuration
- TypeScript setup** - Vue TSC can be slower