

Travelingwebsite App Summary

Repository snapshot summary generated 2026-02-06

What it is

A SvelteKit web app that presents a horizontally scrolling, section-based portfolio experience branded as Full Scope Media.

The repository is also configured as a packageable Svelte library, but the primary user-facing artifact here is the root route showcase page.

Who it's for

- Primary persona appears to be a traveling photographer or small creative studio owner (Ryan Champion / Full Scope Media) who needs a visual portfolio and booking-oriented landing page.

What it does

- Renders a full-viewport horizontal rail of sections (hero, work, journal, services, schedule, testimonials, about, contact).
- Uses a fixed top navigation bar with section buttons that programmatically jump to section indices.
- Presents static content blocks for travel chapters and client testimonials with responsive card layouts.
- Implements custom wheel-to-horizontal scrolling behavior for desktop navigation.
- Supports keyboard navigation (Arrow keys, PageUp/PageDown, Home/End) across sections.
- Includes a styled contact form UI shell (name, phone, email) with no submit handler connected to backend routes.
- Uses reduced-motion preference checks and scroll snapping for smoother, accessible navigation behavior.

How it works (repo evidence only)

- Route composition: 'src/routes/+page.svelte' defines page sections and static arrays ('navLinks', 'cards', 'testimonials') and nests everything inside 'HorizontalScroller'.
- Component behavior: 'src/lib/components/HorizontalScroller.svelte' owns the rail, maps wheel and keyboard events to horizontal movement, and uses 'IntersectionObserver' to track the visible section.
- State/data flow: 'src/lib/stores/scroller.ts' exposes 'scrollerHandle' and 'activeIndex'; the page reads 'scrollerHandle' to call 'scrollToIndex' when nav buttons are clicked.
- Styling stack: global Tailwind import plus custom CSS variables in 'src/app.css'; route-level styles in 'src/routes/+page.svelte'.
- Server/data layer: Drizzle + Postgres setup exists in 'src/lib/server/db/index.ts' and 'src/lib/server/db/schema.ts', but no API routes or UI wiring to that DB were found in this repo.

How to run (minimal)

- Install dependencies: 'npm install'.
- Start local dev server: 'npm run dev -- --open'.
- Optional quality check: 'npm run check'.
- Node version requirement: Not found in repo. Production deploy URL/instructions: Not found in repo.