Component Library

Components are grouped by feature under `src/components/`. Server components are the default. Only components that need browser state are marked `"use client"`.

UI primitives (`components/ui`)

Component	Purpose / API
`Button` / `buttonClasses(variant, size, extra?)`	Renders as a link (`href`) or a `<button>`. Variants: `primary` (brass; reserved for the strongest action), `secondary` (navy), `ghost`, `outline-light` (over photos). Sizes `sm` / `md` / `lg`, all at least 44px tall. Use `buttonClasses` to style plain `<a>` elements, such as FareHarbor or HighLevel links.
`Container`	Max-width page gutter wrapper.
`Card`, `CardBody`	Bordered card; optional `href` makes the whole card a link.
`Badge`	Variants: `featured`, `popular`, `new`, `neutral`, `onImage`.
`Tag`	Static pill label (event types on the Host page).
`FilterPill`	Toggle pill with `selected` state (event filters, list/calendar toggle).
`Accordion`	Accessible disclosure list; `items`, `headingLevel` (2/3/4).
`StickyTabNav`	Sticky in-page tab bar; `tabs: {id,label}[]` scroll to section IDs. Sections use `scroll-mt-32 lg:scroll-mt-36` so headings clear the sticky header.
`Breadcrumb`	`items: {label, href?}[]`.
`FortImage`	`"use client"` wrapper around `next/image`. Empty or null `src`, or a load error, swaps to `/images/placeholders/fort-fallback.svg` (or a custom `fallbackSrc`). **Use it for any image whose source may be missing** (CMS uploads, remote media).

Layout (`components/layout`)

Component	Notes

`Header`	Client component. Sticky; hamburger below `lg`, with Escape-to-close and body-scroll lock; Donate plus Buy Tickets (a direct FareHarbor link when `bookingUrl` is valid).
`Footer`	Org contact details, link columns, social icons, build-time copyright year.
`AlertBanner`	Client component. Dismissal is stored in `localStorage` under `fa-alert-dismissed:<hash(message)>`. Uses `useSyncExternalStore` with a server snapshot of "dismissed" to avoid a flash or hydration mismatch.
`PageHero`	Photo hero with breadcrumbs, eyebrow, title, and intro, used on most interior pages.
`Logo`	Site wordmark/logo.

Feature components

Folder	Components
`home/`	`HomeHero`, `QuickInfoStrip` (hours/directions/admission/parking, **hard-coded**), `ExploreFort` (client; desktop map with markers and autoplaying video, mobile full-screen vertical-video reel below 1024px, captions support), `UpcomingEvents` (client; horizontal scroller with dots), `AudienceRouter`, `VenueTeaser`, `FortressStory`
`visit/`	`TourCard`, `FareharborWidget` (renders a disabled "Online booking opens soon" button if the URL is invalid), `FareharborScript`
`events/`	`EventsExplorer` (client; filters and view toggle), `EventsCalendar`, `EventCard` (CTA priority: `externalUrl` → `bookingUrl` → "Learn more"), `FeaturedEventCard`, `EventsSidebar` (subscribe panel only when `NEXT_PUBLIC_EVENTS_SUBSCRIBE_URL` is valid), `EmailSignup`
`history/`	`HistoryExperience` (client; chapter ↔ hotspot sync, timeline strip, mobile Story/Map toggle), `RestorationTimeline`
`host/`	`VenueCard`, `EventInquiryForm`
`support/`	`DonateWidget` (link to the HighLevel form), `DafPanel`, `MembershipTiers`, `LeadForm` (generic webhook form), `SponsorGrid`

Conventions

- Keep data in `src/lib` and presentation in `src/components`. Components import typed data rather than defining copy inline, except for page-specific prose in `page.tsx` files.
- Client components receive build-time data as props from server components (for example, `EventsExplorer` gets events from `events/page.tsx`).
- Forms use `useId()` for label/input IDs, move focus to the confirmation message on success, and use `role="status"` / `role="alert"`.

Revision #1
Created 2026-09-28 14:25:57 UTC by David Prata
Updated 2026-09-28 14:25:57 UTC by David Prata