

NARRA

Landing Page

Developer Build Guide · React + Vite + shadcn/ui

Stack	React 18 + Vite 5 + TypeScript
UI	shadcn/ui (Radix UI + Tailwind CSS v3)
Fonts	DM Serif Display, DM Sans, DM Mono (Google Fonts)
Routing	React Router v6 (single-page)
Motion	Framer Motion (scroll reveal & hero animations)
Icons	Lucide React
Version	1.0.0 · February 2026

1. Project Setup

1.1 Scaffold with Vite

Run the following commands in your terminal to bootstrap the project:

```
npm create vite@latest narra-landing -- --template react-ts
cd narra-landing
npm install
```

1.2 Install Dependencies

```
# Tailwind CSS
npm install -D tailwindcss postcss autoprefixer
npx tailwindcss init -p

# shadcn/ui
npx shadcn-ui@latest init

# Animation & utilities
npm install framer-motion lucide-react clsx tailwind-merge

# Routing
npm install react-router-dom
```

1.3 Folder Structure

Organise your source directory as follows:

```
src/
```

```
components/  
  ui/          ← shadcn/ui primitives (auto-generated)  
  layout/      ← Nav, Footer  
  sections/    ← Hero, Features, HowItWorks, Metrics,  
                Testimonials, Pricing, CTA  
  shared/      ← EyebrowTag, SectionHeading, AnimatedCard  
lib/  
  utils.ts     ← cn() helper  
styles/  
  globals.css  ← CSS variables + Tailwind directives  
App.tsx  
main.tsx
```

2. Theme Configuration

2.1 CSS Variables (globals.css)

Define the design tokens in :root. These map directly to the original landing page variables:

```
@tailwind base;  
@tailwind components;  
@tailwind utilities;  
  
@import url('https://fonts.googleapis.com/css2?family=DM+Serif+Display:ital@0;1  
  &family=DM+Sans:wght@300;400;500;600;700  
  &family=DM+Mono:wght@400;500&display=swap');  
  
:root {  
  --bg:          #f5f3ef;  
  --bg2:         #edeae3;  
  --surface:     #ffffff;  
  --border:      #e2ddd6;  
  --teal:        #0e6b5e;  
  --teal-mid:    #12877a;  
  --teal-light:  #e6f3f1;  
  --gold:        #c8943a;  
  --gold-light:  #fdf5e8;  
  --ink:         #1a1814;  
  --ink2:        #3d3830;  
  --muted:       #7a7469;  
  --dark:        #1a1814;  
  --radius:      14px;  
  --radius-sm:   8px;  
}
```

2.2 Tailwind Config (tailwind.config.ts)

```
export default {  
  darkMode: ["class"],  
  content: ["../index.html", "./src/**/*.ts,tsx"],  
  theme: {  
    extend: {  
      colors: {  
        teal: { DEFAULT: "#0e6b5e", mid: "#12877a", light: "#e6f3f1" },  
        gold: { DEFAULT: "#c8943a", light: "#fdf5e8" },  
        ink: { DEFAULT: "#1a1814", 2: "#3d3830" },  
      },  
    },  
  },  
}
```

```
    muted: { DEFAULT: "#7a7469", 2: "#a09890" },
    bg:     { DEFAULT: "#f5f3ef", 2: "#edeae3" },
    dark:   { DEFAULT: "#1a1814", 2: "#252118" },
    border: "#e2ddd6",
  },
  fontFamily: {
    serif: ["DM Serif Display", "serif"],
    sans:  ["DM Sans", "sans-serif"],
    mono:  ["DM Mono", "monospace"],
  },
  borderRadius: {
    DEFAULT: "14px",
    sm: "8px",
    full: "9999px",
  },
},
},
plugins: [require("tailwindcss-animate")],
};
```

2.3 Color Token Reference

Token	Value	Usage
--teal	#0e6b5e	Primary brand, nav CTA, icon background, borders
--teal-mid	#12877a	CTA hover state
--teal-light	#e6f3f1	Teal tint backgrounds, badge fills
--gold	#c8943a	Eyebrow labels, accent highlights
--gold-light	#fdf5e8	Gold tint backgrounds
--ink	#1a1814	Primary text, headings
--muted	#7a7469	Secondary text, nav links
--bg	#f5f3ef	Page background
--bg2	#edeae3	Hover fills, alternate backgrounds
--dark	#1a1814	Hero section background
--border	#e2ddd6	Card borders, dividers

3. shadcn/ui Component Setup

3.1 Init Configuration

When running `npx shadcn-ui@latest init`, use these settings:

Style	Default
Base color	Neutral
CSS variables	Yes

TypeScript	Yes
Tailwind config	tailwind.config.ts
Components dir	src/components/ui
Utils file	src/lib/utils.ts

3.2 Install Required Components

```
npx shadcn-ui@latest add button
npx shadcn-ui@latest add card
npx shadcn-ui@latest add badge
npx shadcn-ui@latest add tabs
npx shadcn-ui@latest add separator
npx shadcn-ui@latest add navigation-menu
```

3.3 Override shadcn Tokens in globals.css

Map Narra's brand tokens into shadcn's CSS variable system inside @layer base:

```
@layer base {
  :root {
    --background: 245 241 235; /* --bg */
    --foreground: 26 24 20; /* --ink */
    --primary: 14 107 94; /* --teal */
    --primary-foreground: 255 255 255;
    --secondary: 237 234 227; /* --bg2 */
    --border: 226 221 214; /* --border */
    --ring: 14 107 94;
    --radius: 14px;
  }
}
```

4. Page Sections

4.1 Section Map

Section	Component / Element	Key Notes
Nav	<NavigationMenu>	Fixed, transparent → frosted glass on scroll. Logo left, links center, Sign In + CTA right.
Hero	Full-screen div	Dark bg (#1a1814). Mesh gradients via radial CSS. DM Serif Display headline. Two CTA buttons. Social proof bar.
Metrics	Grid of stat cards	3-up grid. Animated counter on scroll entry. Teal accent borders.
Features	<Tabs> + <Card>	Tabbed interface (Booking / Recall / Inbox). Left tab list, right content panel with icon + description.
How It Works	Numbered steps	3-step layout with connector lines. Icon circles with teal fill.

Section	Component / Element	Key Notes
Testimonials	<Card> carousel/grid	3 testimonials. Avatar initials with gradient bg. Star rating row.
Pricing	<Card> x3	3-tier grid. Center card dark (featured) with teal gradient. Feature checklist per tier.
CTA	Dark section	Full-width dark panel. Serif italic headline. Two CTA buttons.
Footer	Grid layout	4-column: Logo+desc, Product, Company, Legal. Bottom bar with badges.

4.2 Shared Components to Build

Create these reusable primitives in `src/components/shared/` before building sections:

- `EyebrowTag` — small uppercase pill label (teal or gold variant)
- `SectionHeading` — h2 in DM Serif Display with optional italic span
- `SectionSub` — muted paragraph body text, max-width constrained
- `AnimatedCard` — framer-motion wrapper with `fadeUp` + stagger delay
- `GradientButton` — primary teal CTA with glow box-shadow
- `GhostButton` — outline pill button
- `FeatIcon` — rounded square icon container in teal-light bg

5. Hero Section Details

5.1 Background Layers (z-index order)

Stack these layers inside a relative container:

1. Base: bg-dark (#1a1814) full-screen div
2. hero-grid: subtle 60px grid overlay, mask-image radial fade
3. hero-mesh: 3 radial-gradient layers (teal bottom, gold top-right, teal top-left)
4. hero-orb: 600px circle, pulse animation, positioned bottom-center
5. Content: z-index 1, centered flex column

5.2 Animations

Use Framer Motion for all hero entrance animations:

```
const fadeUp = {
  hidden: { opacity: 0, y: 28 },
  visible: (delay = 0) => ({
    opacity: 1, y: 0,
    transition: { duration: 0.7, ease: [0.16, 1, 0.3, 1], delay },
  }),
};

// Usage
<motion.div variants={fadeUp} custom={0.1} initial="hidden" animate="visible">
```

| *Stagger delays: eyebrow 0s, title 0.1s, subtitle 0.2s, buttons 0.3s, social proof 0.4s*

5.3 Headline Structure

The headline uses mixed text styles inline. Wrap each variant in a span:

- "Fill" words — white, normal weight
 - "Italic" words — DM Serif Display italic, -webkit-text-stroke outline style
 - "Underlined" word — white with gradient underline via ::after pseudo-element
-

6. Scroll Reveal System

6.1 Framer Motion Viewport Trigger

Replace the vanilla IntersectionObserver from the original HTML with Framer Motion's `whileInView`:

```
import { motion } from "framer-motion";

const reveal = {
  hidden: { opacity: 0, y: 28 },
  visible: { opacity: 1, y: 0,
    transition: { duration: 0.7, ease: [0.16, 1, 0.3, 1] } },
};

<motion.div
  variants={reveal}
  initial="hidden"
  whileInView="visible"
  viewport={{ once: true, margin: "-40px" }}
>
```

6.2 Staggered Children

```
const stagger = {
  visible: { transition: { staggerChildren: 0.1 } },
};

<motion.div variants={stagger} initial="hidden" whileInView="visible">
  {items.map(item => (
    <motion.div key={item.id} variants={reveal}>
      <Card>{item.content}</Card>
    </motion.div>
  ))}
</motion.div>
```

7. Navigation

7.1 Scroll Behaviour

The nav switches from transparent to a frosted-glass surface at 40px scroll depth:

```
const [scrolled, setScrolled] = useState(false);

useEffect(() => {
  const onScroll = () => setScrolled(window.scrollY > 40);
  window.addEventListener("scroll", onScroll);
  return () => window.removeEventListener("scroll", onScroll);
}, []);

// Apply to nav className:
cn(
  "fixed top-0 inset-x-0 z-50 h-[68px] flex items-center px-12",
  scrolled && "bg-bg/90 backdrop-blur-md border-b border-border shadow-sm"
)
```

7.2 Smooth Anchor Scrolling

```
const scrollTo = (id: string) => {
  document.getElementById(id)?.scrollIntoView({ behavior: "smooth", block: "start"
});
};
```

8. Pricing Section

8.1 Card Variants

Three cards share a base Card component but differ by variant prop:

default	White bg, --border stroke, --ink text
featured	Dark bg (#1a1814), teal/gold gradient top border, white text. Larger scale.

8.2 Feature List

Each tier renders a checklist. Use a boolean flag for included vs. excluded items:

```
interface Feature { label: string; included: boolean; }

<li className="flex items-center gap-2">
  {feature.included
    ? <Check className="text-teal w-4 h-4" />
    : <X className="text-muted w-4 h-4" />}
  <span className={feature.included ? "text-ink" : "text-muted"}>
    {feature.label}
  </span>
</li>
```

9. Performance Notes

9.1 Font Loading

Add the Google Fonts link tag to index.html with display=swap. Do not import via CSS @import in the component tree — it blocks rendering.

9.2 Image Optimisation

- Use WebP format for all screenshots and feature illustrations
- Lazy-load all below-the-fold images with loading="lazy"
- Provide explicit width and height attributes to prevent layout shift

9.3 Animation Performance

- Only animate transform and opacity — never width, height, or top/left
- Add will-change: transform to the hero orb element
- Use viewport={{ once: true }} on all scroll triggers to avoid re-triggering

10. Implementation Checklist

Phase 1 — Setup

- Scaffold Vite + React TS project
- Install Tailwind, shadcn/ui, Framer Motion, Lucide
- Configure tailwind.config.ts with Narra tokens
- Set globals.css with CSS variables and font imports
- Override shadcn token layer

Phase 2 — Shared Components

- EyebrowTag
- SectionHeading + SectionSub
- AnimatedCard (Framer Motion wrapper)
- GradientButton + GhostButton
- FeatIcon

Phase 3 — Layout

- Nav with scroll state + smooth anchors
- Footer grid

Phase 4 — Sections (top to bottom)

- Hero (mesh bg + animations)
- Metrics / Stats bar
- Features (Tabs)
- How It Works
- Testimonials
- Pricing (3-card grid)

- Final CTA

Phase 5 — QA

- Test scroll reveal on all sections
- Verify responsive layout at 375px, 768px, 1280px, 1440px
- Confirm font loading — no FOUT on DM Serif Display
- Validate color contrast ratios (WCAG AA)
- Lighthouse score targets: Performance 90+, Accessibility 95+

Narra AI Technologies · Made in Pampanga